TU Clausthal

Fakultät für Mathematik/Informatik und Maschinenbau
Institut für Software and Systems Engineering

Prädiktion einer langfristigen Fahrzeugzustandsänderung anhand virtueller datengetriebener Sensormodelle

Dissertation

zur Erlangung des Doktorgrades
der Ingenieurwissenschaften

vorgelegt von
Andreas Udo Sass
aus Gifhorn

genehmigt von der
Fakultät für Mathematik, Informatik und Maschinenbau
der Technischen Universität Clausthal

Tag der mündlichen Prüfung: 12.11.2021

Dissertation Technische Universität Clausthal, SSE-Dissertation 25, 2021

Dekan
Prof. Dr. rer. nat. Jörg P. Müller

Vorsitzender der Promotionskommission
Prof. Dr. Thorsten Grosch

Betreuer
Prof. Dr. Andreas Rausch

Gutachter
Prof. Dr.-Ing. Christian Rembe
Prof. Dr.-Ing. Volker von Holt

D 104

Für meine Familie.

Kurzfassung

Die immer weiterwachsende Digitalisierung in der Automobilindustrie ermöglicht eine vermehrte Nutzung und Analyse von Fahrzeug(flotten)daten. Die Nutzung dieser Flottendaten verspricht ein hohes Wertschöpfungspotenzial für zukünftige Mehrwertdienste. Dem Kunden können frühzeitig umfangreiche prädiktive Wartungs- und Reparaturinformationen mit Hilfe von datengetriebenen Analysemethoden bereitgestellt werden. In dieser Arbeit wird eine langfristige Fahrzeugzustandsänderung anhand virtueller datengetriebener Sensormodelle untersucht. Als Grundlage dafür werden dynamische CAN-Daten von unternehmensinternen Fahrzeugflotten verwendet. Der zugehörige aktuelle Stand der Wissenschaft wird im Rahmen der Problembeschreibung diskutiert.
Im weiteren Verlauf wird ein Konzept entworfen, welches die Schritte der Datenvorverarbeitung und des Data-Minings in Anlehnung an den Prozess der Knowledge Discovery in Databases (KDD) konkretisiert. Mit Hilfe geeigneter Vorverarbeitungen wie z.B. Clusterverfahren und Merkmalsextraktionen kann die Menge der Eingangsdaten reduziert werden. Im Rahmen dieser Vorverarbeitung werden die unterschiedlichen Signale unüberwacht gruppiert. Aus einer Gruppe verwechselbarer Signale wird ein Vertreter gewählt, der im weiteren Verlauf als einzigartiges Signal zur weiteren Analyse verwendet wird. Innerhalb der gesamten Zeitreihe werden Sequenzen auf Basis gewählter Diskretisierungsstufen erstellt. Aus diesen Sequenzen werden statistische Merkmale extrahiert und zur weiteren Verarbeitung genutzt. Unter Anwendung von Regressionsmethoden ist eine Extraktion relevanter Muster und Regeln aus den Daten möglich. Anhand eines konkreten Beispiels aus der Automobilindustrie wird dieses Vorgehen validiert. Dabei werden zwei Ansätze miteinander verglichen: Ein hybrider Expertenansatz und eine datengetriebene Hyperparameteroptimierung. Im Rahmen des hybriden Expertenansatzes wird zur Alterungsvorhersage auch spezifisches Wissen aus der Fachdomäne genutzt. Beim datengetriebenen Ansatz werden dagegen zahlreiche Einstellparameter der Vorverarbeitung und des Data-Minings mit Hilfe einer Optimierungsstrategie ausgewählt. Neben dem Vergleich der beiden Ansätze sind mit Hilfe von Einflussanalysen auch relevante Erkenntnisse für die Fachdomäne ableitbar. Diese Einflussanalysen können gezielt für zukünftige Fahrzeuggenerationen genutzt werden. Am Ende dieser Arbeit wird das hier beschriebene Vorgehen zusammengefasst, eine mögliche Übertragbarkeit diskutiert und ein Ausblick für zukünftige Forschungstätigkeiten gegeben.
Diese Arbeit kann dazu beitragen den steigenden Durchsatz digitaler Daten gezielt zu reduzieren. Es wird gezeigt, dass durch die Verwendung geeigneter Methoden des maschinellen Lernens die Eingangsdatenmenge um ein Vielfaches reduziert und gezielt für (Alterungs-) Vorhersagen genutzt werden kann.

Vorwort

Die vorliegende Arbeit entstand in der Zeit von Juni 2017 bis März 2021 in der Group Innovation der Volkswagen AG am Standort Wolfsburg. Viele unterschiedliche Menschen haben mich dabei auf meinem Weg begleitet und ich möchte diesen Personen an dieser Stelle danken.
Besonderer Dank gebührt meinem Doktorvater Prof. Dr. Andreas Rausch für die Betreuung und Begutachtung meiner Arbeit. Durch viele intensive Gespräche und lange Diskussionen konnte die Arbeit Schritt für Schritt geschärft werden. Dieser Austausch hat maßgeblich zum Gelingen dieser Arbeit beigetragen.
Weiterhin bedanke ich mich bei Prof. Dr.-Ing. Christian Rembe und Prof. Dr.-Ing. Volker von Holt für das Interesse an der Arbeit und die hilfreichen Diskussionen, sowie die Bereitschaft der Übernahme der Gutachten. Ich danke auch Herrn Prof. Dr. Thorsten Grosch für die Übernahme des Vorsitzes der Prüfungskommission.
Ich möchte mich für die Hilfsbereitschaft meiner Kollegen bedanken, die mich von Anfang an unterstützt haben. Mit Hilfe des Wissens und der Erfahrung der Kollegen konnte die Arbeit weiterentwickelt werden. Dazu zählen neben dem überfachlichen Austausch auch die tief-technischen Diskussionen. Ein besonders gutes Arbeitsklima hat diese wichtigen Diskussionen ermöglicht und gefördert.
Darüber hinaus möchte ich mich auch bei den Studenten bedanken, die im Rahmen dieser Dissertation Studien- und Masterarbeiten durchgeführt haben. Durch ihren Einsatz konnten Teilfragestellungen bearbeitet und meine Ideen dieser Arbeit weiter geschärft werden.
Abschließend danke ich meinen Eltern für die vorbehaltlose Unterstützung im gesamten Bildungsweg und für die vielen Korrekturen. Besonders großer Dank gilt auch meiner Frau, sie hat mir stets Freiräume ermöglicht und mich zeitgleich ermutigt meine Ziele zu erreichen. Außerdem bedanke ich mich bei den Eltern meiner Frau für das ausführliche Lesen und Korrigieren der Arbeit.

Im Rahmen der Promotionsphase habe ich Zwischenergebnisse auf nationalen und internationalen Fachkonferenzen veröffentlicht und konstruktives Feedback erhalten. Diese Teilstudien beleuchten verschiedene Teilaspekte der Arbeit und werden im Folgenden aufgelistet:

Artikel	Tangierte Kapitel
A. Sass, E. Esatbeyoglu und T. Fischer. „Monitoring of Powertrain Component Aging Using In-Vehicle Signals“. In: *Diagnose in Mechatronischen Fahrzeugsystemen XIII: Neue Verfahren für Test, Prüfung und Diagnose von E/E-Systemen im Kfz* (2019), S. 15–28	4.2, 5.2
A. Sass, E. Esatbeyoglu und T. Iwwerks. „Signal Pre-Selection for Monitoring and Prediction of Vehicle Powertrain Component Aging“. In: *Science & Technique* 18.6 (5. Dez. 2019), S. 519–524	4.3
A. Sass, E. Esatbeyoglu und T. Iwwerks. „Data-Driven Powertrain Component Aging Prediction Using In-Vehicle Signals“. In: *SOFSEM (Doctoral Student Research Forum)*. 2020, S. 109–119	5.4

Ergebnisse, Meinungen und Schlüsse dieser Dissertation sind nicht notwendigerweise die der Volkswagen Aktiengesellschaft.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Abkürzungsverzeichnis

AGR Abgasrückführung

APE Absolute Percentage Error

ARMA AutoRegressive-Moving Average

CAN Controller Area Network

CASH Combined Algorithm Selection and Hyperparameter Optimization Problem

CBM condition based monitoring

CRISP-DM Cross-industry standard process for data mining

CSMA/CR Carrier Sense Multiple Access/Collision Resolution

DFT Diskrete Fourier Transformation

DL Deep Learning

DTW Dynamic Time Warping

DWT Diskrete Wavelet Transformation

ECU elektronische Steuereinheit

FMI Fowlkes-Mallow Index

GUM Guide to the Expression of Uncertainty in Measurement

HMM Hidden Markov Model

KDD Knowledge Discovery in Databases

kNN k-Nearest Neighbor

LASSO Least Absolute Shrinkage and Selection Operator

MAE Mean Absolute Error

MAPE Mean Absolute Percentage Error

MdAPE Median Absolute Percentage Error

ME Mean Error

MPE Mean Percentage Error

MIL Malfunction Indicator Light

ML Maschinelle Lernen

MLR multiple lineare Regression

MSE Mean Squared Error

NN neuronale Netze

OBD On-Board-Diagnose

PAA Piecewise Aggregate Approximation

PCA Principal Component Analysis

RBF radiale Basisfunktion

ReLu rectified linear unit

RF Random Forest

RMSE Root Mean Square Error

RNN rekurrentes neuronales Netz

Rprop Resilient Backpropagation

RUL Remaining Useful Life

SAX Symbolic Aggregate approXimation

SMAC Sequential Model-based Algorithm Configuration

SOM Self Organizing Map

SSR sum of squared residuals

SST total sum of squares

SVM Support Vektor Maschine

SVR Support Vektor Regression

TPE Tree-Structured Parzen Estimator

TTCAN Time-Triggered-CAN

Symbolverzeichnis

A	effektive Querschnittsfläche
b_k	Regressionskoeffizient
C	Regularisierungsparameter (SVR)
C_k	Menge der Klassen (clustering)
Cov	Kovarianz
d	Distanz zwischen Datenpunkten, Diskretisierungsstufe
e	Residuum
J	Modellgüte
k	Anzahl an Nachbarn (kNN)
$\dot{m}$	Massenstrom
n_{pred}	Anzahl der prädizierten Werte
o	Teilmenge an Merkmalen
p	Distanzfunktion (kNN)
p_{post}	Druck nach der Drosselstelle
p_{pre}	Druck vor der Drosselstelle
q	Quartil
R	Korrelationskoeffizient, spezifische Gaskonstante
r_m	Massenstromrate
R^2	Bestimmtheitsmaß
s	Teilmenge an Signalen
T	Länge einer Reihe
T_{pre}	Temperatur vor der Drosselstelle
x	Einflussgröße
y	Zielgröße
α	ML-Methode
ε	Toleranzband
κ	Isentropenexponent des durchströmenden Gases
λ	Ausfallrate, Hyperparameter
μ	Erwartungswert
Π_{krit}	kritisches Druckverhältnis
Ψ_{max}	maximaler Durchflussbeiwert
s_x	Standardabweichung
σ^2	Varianz
$\bar{x}$	Mittelwert

1 Einleitung

Künstliche Intelligenz, Smart Home, Industrie 4.0 – Eine Auswahl spezieller Schlagwörter, die dem Überbegriff Digitalisierung zugeordnet werden können. Im ursprünglichen Gebrauch bezeichnet der Begriff Digitalisierung die digitale Speicherung und Anwendung analoger Messwerte. Mit der Einführung digitaler Zeitmessungen, Computern und des Internets können nicht nur immer mehr Daten gesammelt, sondern auch kabellos geteilt werden. Eine immer weiterwachsende Digitalisierung findet in allen Bereichen unseres Alltags statt, zum Beispiel durch größer werden Speichermedien oder die Nutzung von mobiler Funkverbindungen wie 4G oder 5G. Der weltweite private digitale Datentransfer über das Internet belief sich im Jahr 2017 auf rund 100 Exabyte pro Monat. Für das Jahr 2022 wird ein Anstieg auf rund 333 Exabyte pro Monat erwartet [Pol20]. Die kabellose Vernetzung hat nicht nur einen stetig steigenden Durchsatz[1] an Daten zur Folge, sondern lässt auch den Energiebedarf für die Digitalisierung weiter ansteigen. Allein in Deutschland betrug der Strombedarf von Servern und Rechenzentren im Jahr 2017 insgesamt 13,2 Mrd. kWh [Hin18]. Das entspricht einer Steigerung um 25 % im Vergleich zum Strombedarf im Jahr 2010 von insgesamt 10,5 Mrd. kWh. [Hin18]
Auch in der Automobilindustrie sind die Auswirkungen der Digitalisierung allgegenwärtig. Die steigende Anzahl an Steuergeräten führt zu einer größeren Komplexität und damit verbunden einem höheren Datenaustausch auf internen Protokollen [Wal06, S. 180]. Zeitgleich besteht auf Seiten des Herstellers ein größeres Interesse, dem Kunden durch das Angebot moderner Technologien einen Mehrwert zu bieten.
Die Digitalisierung in der Automobilindustrie lässt sich anhand eines Beispiels der Flottenverwaltung eines Logistikunternehmens veranschaulichen: Mit Hilfe der Digitalisierung und dem zunehmenden möglichen Austausch an Daten über das Internet kann die gesamte Flotte in Echtzeit überwacht und dem Logistikunternehmen somit ein Mehrwert gegeben werden. Denkbar ist dabei nicht nur die simultane Darstellung der aktuellen Fahrzeugposition auf einer Karte, sondern auch das simultane Sammeln von Zustandsinformationen über das Fahrzeug, den Fahrer und die Umwelt. Ein Großteil der Zustandsinformationen eines Fahrzeugs lässt sich direkt aus Sensordaten ableiten. Beispielhaft sei an dieser Stelle der Beladungszustand eines Fahrzeugs sowie eine Information über die Tankfüllung genannt.
Neben direkten univariaten Sensordaten können auch umfangreiche weitere Analysedaten abgeleitet werden, die nicht nur von einem, sondern zeitgleich von zahlreichen Sensoren abhängig sind. Jedes Fahrzeug kann zudem fahrerindividuell betrieben werden. Verschiedene Fahrzeuge sind somit unterschiedlichen Gegebenheiten ausgesetzt.
Die zuvor beschriebene Digitalisierung erhält zunehmend Einzug in die Automobilbranche. Es ist möglich, moderne Fahrzeuge untereinander sowie mittels Cloud-Speichern zu vernetzen. In diesem Zusammenhang sind Technologien wie Car2Car, V2X (engl. *vehicle to everything*) und 5G zu nennen.
Die zunehmende Vernetzung ermöglicht komplexere Datenauswertungen. Die Nutzung und

1 Ein Durchsatz beschreibt, wie viele Daten und Anweisungen in einer bestimmten Zeiteinheit verarbeitet werden können.

Analyse von Fahrzeug(flotten)daten verspricht ein hohes Wertschöpfungspotenzial für zukünftige Mehrwertdienste.
Prädiktive Wartungs- und Reparaturinformationen können frühzeitig mit Hilfe von datengetriebenen Analysemethoden bereitgestellt werden. Im Rahmen einer prädiktiven Instandhaltungsanalyse (engl. *predictive maintenance*) werden dem Kunden Informationen über den Zustand des Fahrzeugs oder einer Fahrzeugkomponente geliefert. Zur Modellierung und Bereitstellung dieser intelligenten Instandhaltungsstrategien ist auf Seiten des Fahrzeugherstellers ein hoher Aufwand im Entwicklungsprozess notwendig. Die unterschiedlichen Daten für eine solche Funktion müssen zunächst aufgezeichnet und gespeichert werden. Nach einer Datenvorverarbeitung und einer anschließenden Auswertung kann dem Kunden daraus eine Instandhaltungsstrategie erarbeitet werden.

1.1 Motivation

Mittels interner Fahrzeugnetzwerke (auch Bussysteme genannt) werden Informationen im Fahrzeug ausgetauscht. Diese Busse werden zur internen Kommunikation des Fahrzeugs genutzt, um unterschiedliche Anfragen der Steuergeräte mit digitalen Informationen zu versorgen. Aus diesem Grund stehen bereits ohne zusätzliche Einbaumaßnahmen zahlreiche Daten in Serienfahrzeugen zur Verfügung. Mit Hilfe dieser fahrzeuginternen Informationen lassen sich mittels Diagnosealgorithmen gezielt Fahrzeugsysteme überwachen. Ein Teil der Diagnosealgorithmen kann automatisiert im Fahrzeug selbst erfolgen. Diese Diagnosen werden in der Automobilindustrie unter dem Begriff On-Board-Diagnose (OBD) zusammengefasst. Gesetzliche Anforderungen verpflichten die Automobilhersteller dazu, abgasrelevante Teil(-systeme) im Fahrzeug zu überwachen und zu dokumentieren. So wird mit Hilfe des OBD-Systems der Katalysator über λ-Sonden überwacht [Wal06, S. 636]. Messwertüberschreitungen, beispielsweise der Kühlwassertemperatur, der Ansauglufttemperatur oder der Betriebsdauer nach Zurücksetzung des fahrzeuginternen Fehlerspeichers, können somit abgefragt werden [ZS14, S. 216]. Das Erkennen eines Fahrzeugfehlers wird dem Fahrer Aufleuchten der Motorkontrollleuchte (engl. *Malfunction Indicator Light*) angezeigt. Weiterhin können aufgezeichnete Fehlereinträge durch Speicherabfragen von Experten ausgelesen werden. In den gesetzlichen Normen sind die OBD-Diagnosemöglichkeiten fest definiert. Es werden nur Messwerte des eigenen Fahrzeuges betrachtet, ein Rückschluss, zum Beispiel im Rahmen eines Flottenmanagements, ist mit der OBD-Strategie nicht möglich.
Im weiteren Verlauf des Kapitels werden Strategien der Instandhaltung beschrieben, die über die im Fahrzeug implementierten Diagnosealgorithmen (OBD) hinaus gehen. Die Umsetzung solcher intelligenten Instandhaltungsstrategien ist meist nur durch einen hohen Wissensstand auf Seiten des Fahrzeugherstellers und durch Analyse großer Datenmengen möglich.

In der Literatur werden unterschiedliche Strategien der Instandhaltung definiert, diese sind nach [dFCO15]:

- korrigierende Instandhaltung (engl. *corrective maintenance*),
- präventive Instandhaltung (engl. *preventive maintenance*),
- prädiktive Instandhaltung (engl. *predictive maintenance*), und
- proaktive Instandhaltung (engl. *proactive maintenance*).

Korrigierende Instandhaltung Eine häufig verbreitete Form (vor allem in industriellen Anlagen) der Instandhaltung ist die *korrigierende* oder auch reaktive Instandhaltung. Systeme und Komponenten werden repariert, sobald ein Defekt auftritt. Werden industrielle Anlagen durch eine gemeinschaftliche Produktion betrieben, können Stillstandszeiten zu besonders hohen Kosten führen.

Präventive Instandhaltung Mittels der Strategie (*präventiver Instandhaltung*) wird versucht ungeplante korrigierende Instandhaltung zu vermeiden, indem basierend auf bekannten Intervallen (Teil-)Systeme präventiv ausgetauscht werden. In der Automobilbranche sind Service-Intervalle zeitbasiert (z.B. nach 2 Jahren) oder nach Festlegung einer Laufleitung (z.B. nach 60.000 km) definiert. Auch wenn bei Erreichen des Serviceintervalls noch kein Defekt vorliegt, werden einzelne (Teil-)Systeme getauscht. Bezogen auf die Gesamtlaufleistung eines Fahrzeugs fallen durch diese Strategie hohe Kosten an, wenn eine Komponente vermeintlich zu früh getauscht wird oder eine Komponente schon vor Ablauf des Serviceintervalls ausfällt.

Prädiktive Instandhaltung Im Gegenzug zur präventiven wird in der *prädiktiven* Instandhaltung das Alterungsverhalten einer (Fahrzeug-)Komponente vorhergesagt. Der Schwerpunkt dieser Strategie liegt auf der Komponentenfehler- und Alterungsprädiktion. Zur Prädiktionsberechnung wird ein mathematischer (rechnergestützter) Algorithmus genutzt. Dieser Algorithmus kann als Werkzeug verstanden werden, um einen geeigneten Zeitpunkt zum Austausch oder Wechsel einer Komponente vorherzusagen. Gegenüber der präventiven Instandhaltung können hierdurch Kosten gesenkt und der Zeitplan für Instandhaltungsmaßnahmen optimiert werden [Lee+14]. Prytz [Pry14] unterteilt die prädiktive Instandhaltung wiederum in drei Kategorien: Restnutzungsdauer (engl. *Remaining Useful Life*), Erkennung einer Abweichung (deviation detection) und überwachte Klassifizierung (supervised classification). In [TMZ12; Tet+10; Bed+13; Zhe+18] werden Anwendungsbeispiele zur Vorhersage einer Restnutzungsdauer vorgeschlagen. So zeigen beispielsweise die Autoren Zheng u. a. in ihrer Arbeit eine Vorhersage der Restnutzungsdauer verschiedener Flugzeugmaschinen [Zhe+18]. Kargupta u. a. präsentieren in ihrer Arbeit eine Möglichkeit der Implementierung von Abweichungserkennungen bezogen auf die prädiktive Instandhaltung [Kar+10]. Hierbei werden Datenstreams von Nutzfahrzeugen versendet und signifikante Änderungen

in einem Datenfenster untersucht. Instandhaltungsempfehlungen auf Basis von überwachtlernenden Klassifizierungen werden in [GP15; Pap+13; Cae+16] beschrieben.
Neben dem Ziel der prädiktiven Instandhaltung ist auch die Informationsquelle zu berücksichtigten. Im Fahrzeug werden für die Analyse notwendigen Eingangsdaten meist mittels Weiterverarbeitung von CAN-Bus verwendet (vgl. Kapitel 2.3.1). Darüber hinaus können zusätzliche Sensoren installiert werden. In der Studie von Lee u. a. wird aufgelistet, wie Vibrationsinformationen von zusätzlichen Sensoren von Lager, Zahnräder, Wellen, Pumpen, Wechselstromgeneratoren zur prädiktiven Instandhaltung genutzt werden können [Lee+14].

Proaktive Instandhaltung Während bei der prädiktiven Instandhaltung der Fokus auf den Symptomen liegt, wird in der Strategie der proaktiven Instandhaltung dagegen versucht, die Fehlerursachen eines Komponentenausfalls zu identifizieren. Ziel ist es, die Ursache eines möglichen drohenden Maschinenausfalls frühzeitig zu erkennen und Maßnahmen zu ergreifen, sodass im besten Fall kein Ausfall der Komponente eintritt. Für die Analyse der Fehlerursachen ist ein hoher messtechnischer und analytischer Aufwand notwendig. Die Arbeit der Autoren Xu, Wang und Xu zeigt, dass unterschiedliche Sensoren die Qualität der Vorhersagen von Alterungsprognosen von Komponenten beeinflussen können [XWX15]. Die Wahl der richtigen Sensorik ist entscheidend für die Vorhersagekraft einer proaktiven Instandhaltung.

1.2 Zielsetzung

Die in Kapitel 1.1 beschriebenen unterschiedlichen Strategien der Instandhaltung zeigen verschiedene Herangehensweisen zum Austausch einer (Fahrzeug-)Komponente auf. Wird eine Komponente zu früh getauscht, entstehen auf die Lebensdauer des Fahrzeugs bezogen vergleichsweise höhere Kosten, da die Komponente möglicherweise noch länger hätte betrieben werden können, unter der Voraussetzung, dass ein längerer Betrieb die Fahrsicherheit nicht negativ beeinflusst. Gleiches gilt für den reaktiven Fall: Wird eine Komponente erst nach deren Ausfall gewechselt, können höhere Kosten für Stillstandszeiten entstehen. Im Idealfall wird die Komponente dann getauscht, wenn sie gerade noch einwandfrei funktioniert, aber noch keine Einschränkungen im Nutzungsverhalten zeigt [Hod18, S. 138].
Eine Vielzahl fahrzeuginterner Messsignale bildet die Datengrundlage der vorliegenden Arbeit. Diese wurden mittels Datenaufzeichnungsgeräten von Serienfahrzeugen gespeichert. Ziel ist es, aus diesen Daten Modelle zu entwickeln, die eine datenbasierte Vorhersage von Fahrzeugzuständen ermöglichen. Im Speziellen handelt es sich bei diesen Zuständen um eine Alterung einer Fahrzeugkomponente. Dieses Modell zur datengetriebenen Prädiktion der Komponentenalterung wird im weiteren Verlauf auch als virtueller Sensor beschrieben. Dieser virtuelle Sensor hat die Aufgabe eine bestimmte langfristige Alterungscharakteristik

einer ausgewählten Fahrzeugkomponente geeignet abzubilden. Diesbezüglich stehen hochdynamische Daten des fahrzeuginternen Netzwerks in unterschiedlichen zeitlichen Auflösungen zur Verfügung. Weiterhin wird in dieser Arbeit geprüft, wie das zu erstellende Sensormodell hinsichtlich der Vorhersagequalität optimiert werden kann. Mit Hilfe aufgenommener Messdaten wird das Sensormodell validiert.

1.3 Aufbau der Arbeit

Die vorliegende Arbeit ist in sechs Kapitel aufgeteilt. In der Einleitung (vgl. Kapitel 1) wird der Leser in die zum Verständnis dieser Arbeit relevanten Themenbereiche eingeführt. Das nächste Kapitel (vgl. Kapitel 2) beschreibt die Grundlagen und Vorbetrachtungen, die für das weitere Verständnis der folgenden Kapitel notwendig sind. Im Rahmen der Problembeschreibung (vgl. Kapitel 3) wird das vorliegende Problem zunächst erläutert. Im Anschluss wird der Stand der Wissenschaft dargestellt und die Forschungsfragen definiert. Daran schließt das Kapitel zum Konzeptentwurf bezüglich der Problemlösung im Rahmen des dargestellten Kontextes (vgl. Kapitel 4) an. Die Konzeptvalidierung (vgl. Kapitel 5) stellt die Ergebnisse dieser Arbeit dar. Am Ende des Kapitels werden die Ergebnisse diskutiert und bewertet. Die vorliegende Arbeit schließt mit einer Zusammenfassung und einem Ausblick (vgl. Kapitel 6).
Die Abbildung 1.1 zeigt eine übersichtliche Darstellung zum Aufbau der Kapitel in einer leicht gekürzten Fassung. Die Abbildung zeigt dabei die einzelnen Kapitelüberschriften und

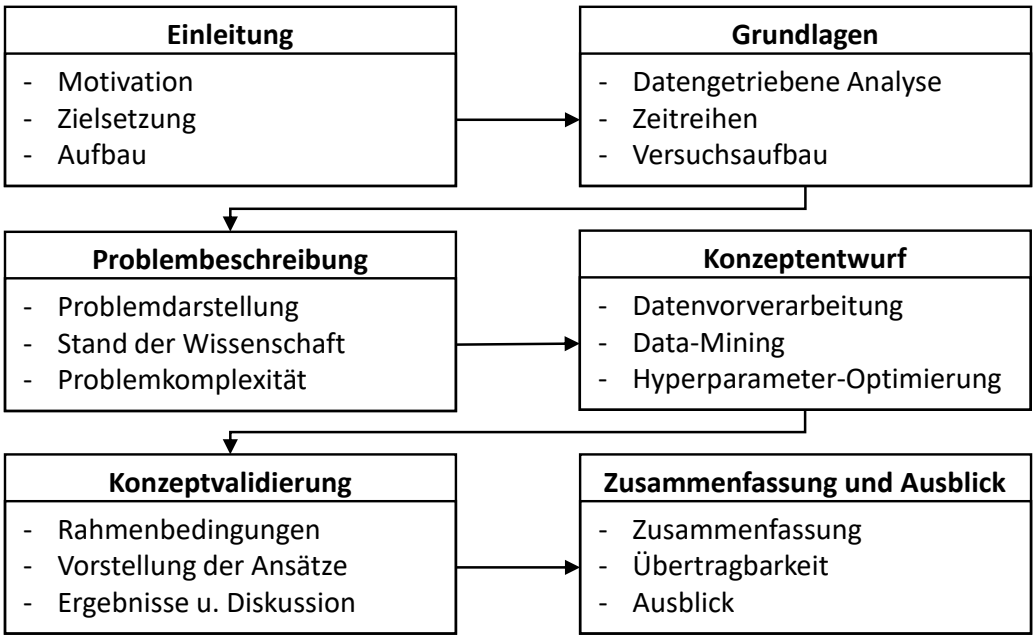


Abbildung 1.1: Überblick über den Aufbau der Arbeit

Auszüge aus den Unterkapiteln. Sie soll dem Leser eine Orientierung in dieser Arbeit geben.

2 Grundlagen und Vorbetrachtungen

In diesem Abschnitt werden Grundlagen und Vorbetrachtungen zur Erstellung eines virtuellen Sensormodells zur Bestimmung einer Alterungscharakteristik einer ausgewählten Fahrzeugkomponente vorgestellt. Eingangs werden in Kapitel 2.1 Grundbegriffe und Methoden des maschinellen Lernens erläutert. Der in dieser Arbeit analysierte Datenbestand liegt in Form von Zeitreihen vor, die in Kapitel 2.2 weiter spezifiziert werden. Der gesamte Versuchsaufbau ist in Kapitel 2.3 dargestellt.

2.1 Einführung in die datengetriebene Analyse

In dieser Arbeit werden mit Hilfe des *Maschinellen Lernens* (ML) datengetriebene Modelle gebildet. Die dafür notwendigen Begriffe werden in Kapitel 2.1.1 eingeführt. Das Maschinelle Lernen (ML) unterteilt sich wiederum u.a. in Methoden der Clusteranalyse (s. Kapitel 2.1.2) und Methoden des überwachten Lernens (s. Kapitel 2.1.3). Um die erstellten Modelle und Vorhersagen bewerten zu können, werden in Kapitel 2.1.4 relevante Kenngrößen des Zusammenhangs und Prognosegütemaße vorgestellt.

2.1.1 Terminologie

Wie bereits in Kapitel 1.1 beschrieben, handelt es sich bei der vorliegenden Komponentenalterung um eine *Black Box*, da nur die Alterung als gemessene Größe vorliegt, nicht aber interne Strukturen, mit deren Hilfe sich diese Alterung bestimmen ließe. So soll mit Hilfe der aufgezeichneten Daten eine *datengetriebene, rechnergestützte* Analyse stattfinden. Dem gegenüber steht die *physikalische* Modellierung. Hierbei wird vom Fachexperten der Domäne ein Systemmodell erstellt, meist auf Grundlage von physikalischen Zusammenhängen des Systems. Zur Erstellung dieses Modells wird ein umfassendes Verständnis des Systems benötigt. Soll dieses Modell zur Vorhersage einer Alterung genutzt werden, muss der Fachexperte zunächst die physikalischen Abhängigkeiten identifizieren und mathematisch modellieren. Danach ist das entwickelte Modell mit Messdaten zu validieren. Bei der Entwicklung eines physikalischen Modells entstehen hohe Kosten. Außerdem kann so ein physikalisches Modell nur für eine bestimmte Fahrzeugkomponente angewendet werden [Bro+00]. Weiterhin können auch beide Ansätze zu einem hybriden Modell kombiniert werden (vgl. „Diagnose und Prognose“).

Modell und Hyperparameter Ein künstliches System übernimmt dabei die Rolle des Fachexperten und lernt aus historischen Datenbeständen Muster und Gesetzmäßigkeiten. Dabei werden die Eingangsdaten genutzt, um diese im Rahmen des Problemkontextes mittels eines geeigneten Modells auf die Ausgangsdaten abzubilden. Das Modell beinhaltet diese

Gesetzmäßigkeiten, sodass die Ausgabe auch für zukünftige Eingangsdaten berechnet werden kann. Diese Erkenntnisse werden genutzt, um Vorhersagen für zukünftige Datensätze zu treffen.
Um das datengetriebene Modell rechnergestützt zu trainieren, werden Trainingsdaten (engl. *training set*) oder auch historische Datensätze benötigt. Mit Hilfe dieser Daten wird das Modell an die Randbedingungen und den zu lernenden Muster angepasst (engl. *fitted*). Hierbei werden die Modellparameter berechnet. In einem künstlichen neuronalen Netz sind dies die Gewichte der Verbindungen, in einer linearen Regression die Linearkombination der Regressionskoeffizienten. Ziel ist es, den Modellfehler (oder auch Vorhersagefehler) bezogen auf die gegebenen Trainingsdaten zu minimieren. Dieses erlernte Modell spiegelt allerdings ausschließlich den Zusammenhang der betrachteten Trainingsdaten und deren Ausgabe wider.
In einem weiteren Schritt wird die Generalisierungsfähigkeit des Modells überprüft. Hierzu werden Testdaten (engl. *test set*) genutzt, um auch hier die Zusammenhänge der Daten mit Hilfe des gelernten Modells zu identifizieren. Verfügt das Modell über eine Fähigkeit den Zusammenhang gut zu verallgemeinern, so wird von einer hohen Prädiktionsgüte gesprochen [Sam17]. Fällt die Prädiktionsgüte des gelernten Modells auf dem Testdatensatz (nicht gelernte Daten) deutlich geringer aus, so kann eine sogenannte Überanpassung (engl. *overfitting*) vorliegen. In diesem Fall kann das erstellte Modell so sensitiv auf die Daten reagiert haben, dass es selbst das Rauschen der Eingangsdaten angelernt hat. In einer multiplen Regression entstehen bei einer Überanpassung nicht relevante Regressoren.
Werden dagegen relevante Variablen modellseitig nicht betrachtet, wird von einer Unteranpassung (engl. *underfitting*) gesprochen [Bac+18, S. 94]. Es ist das Ziel, das Modell mit einer geeigneten Menge an Daten zu trainieren, sodass sich eine hohe Vorhersagequalität einstellt. Zeitgleich darf das Modell aber nicht überangepasst werden, sodass trotz der (eingeschränkten) Datengrundlage eine hohe Generalisierungsfähigkeit entsteht [MG16, S. 29]. Um dies zu überprüfen, kann die so genannte Kreuzvalidierung (engl. *cross-validation*) durchgeführt werden. Hierbei werden die Eingangsdaten gezielt in Trainings- und Testdaten aufgeteilt, um so das Modell besser bewerten zu können (vgl. Kapitel 4.4).
Soll ein erstelltes Modell zur Prognose (oder auch Prädiktion) von Werten genutzt werden, so wird es zunächst mit den wahren gemessenen Werten trainiert. Während des Validierungsprozesses kann mit Hilfe von Prognosegütemaßen (engl. *goodness-of-fit*) die Modellgüte bestimmt werden (vgl. Kapitel 2.1.4).
Weiterhin können Parameter auch außerhalb des Modells festgelegt werden. Diese *Hyperparameter* können vom Lernprozess nicht beeinflusst werden und unterscheiden sich deshalb von den Modellparametern. Die lineare Regression beinhaltet folgende Modellparameter: Den Achsenabschnitt β_0 und den Koeffizienten β_1 der Funktionsgleichung $y(x) = \beta_0 + \beta_1 x$. Hyperparameter sind bei einer linearen Regression nicht vorhanden. Dagegen beschreiben die Hyperparameter neuronaler Netze beispielsweise die Anzahl der Schichten und der Knoten der gewählten Netzstruktur. Im Gebiet der Hyperparameteroptimierung werden optimale Parameter im Rahmen des Problemkontextes gesucht, um so eine höhere Modellgüte zu erlangen [Pad+17]. Sind nun eine Vielzahl von Modellen erstellt, beschreibt die Modellselektion den Prozess der Auswahl des geeignetsten Modells [And19]. Wird ein Modell auf einen bestimmten Datensatz trainiert, so kann mit Hilfe der Hyperparameteroptimierung

eine optimale Hyperparameterkonfiguration bestimmt werden. Auch wenn dieses Modell mit dessen Hyperparameterkonfiguration beim gegebenen Datensatz eine zufriedenstellende Performance erzielt, werden diese Modellgüten nicht zwangsläufig auch bei anderen Datensätzen erzielt.
Im Validierungsprozess ist demnach neben der Wahl des Modells und dessen Hyperparametern auch die Wahl des Testdatensatzes entscheidend. Somit kann neben der Wahl der Modellhyperparameter auch bereits die Auswahl der Daten und dessen Vorverarbeitung als Hyperparameter angesehen werden [Sch+15].

Messunsicherheit und Prognosegüte Zu einem Messexperiment gehören gemessene Werte und wahre Werte. Eine zwischen diesen Werten liegende Messabweichung kann zum Beispiel durch menschliche Fehler, Messgeräteabweichungen oder unterschiedliche Messverfahren hervorgerufen werden. Der Leitfaden Guide to the Expression of Uncertainty in Measurement (GUM) liefert ein standardisiertes Vorgehen zur Bestimmung dieser Messunsicherheiten [BIP20]. Im Rahmen der datengetriebenen Modellerstellung werden Muster und Gesetzmäßigkeiten in vorliegenden Datensätzen künstlich gelernt. Auch hierbei ist mit Fehlern zwischen den wahren und prädizierten Werten auf Grund nicht idealer Modelle zu rechnen. Aus diesen Modellfehlern kann eine Prognosegüte bestimmt werden. Die in dieser Arbeit verwendeten Prognosegütemaße werden im Abschnitt 2.1.4 vorgestellt.

Diagnose und Prognose Unter einer *Prognose* wird die Vorhersage eines Zustandes oder die Vorhersage einer zukünftigen Entwicklung verstanden. Um eine Prognose zu erstellen, wird eine aktuelle *Diagnose* des Zustandes eines Systems oder einer Komponente benötigt. Im medizinischen Kontext prognostiziert der Arzt auf Basis einer Diagnose des Patienten und unter Zuhilfenahme bekannter Verläufe der identifizierten Krankheit den Gesundheitsverlauf des untersuchten Patienten [TMZ12].
Nach den Autoren Kim, An und Choi werden Prognoseansätze hinsichtlich der Nutzung der Informationen in

- physikalische,
- datengetriebene,
- und hybride

Ansätze unterteilt [KAC17]. Sowohl beim physikalischen Ansatz, als auch beim datengetriebenen Ansatz werden Daten zur Erstellung eines Modells benötigt. Zur Prognose des weiteren Verlaufs wird im physikalisch-basierten Ansatz auf ein physikalisches Modell gesetzt, wohingegen der datengetriebene Ansatz zur Extrapolation mathematische Funktionen nutzt. Ein physikalisches Modell benötigt bereits zu Beginn der Modellerstellung ein umfassendes Verständnis der Zusammenhänge. Prognosen mit einem physikalischen Modell sind bereits mit einer, im Vergleich zu datengetriebenen Ansätzen, geringen Datengrundlage möglich. Der datengetriebene Ansatz benötigt dagegen zahlreiche Daten zum Lernen des Modells, da das Modellwissen zunächst aus den zugrundeliegenden Daten extrahiert wird. Eine große Herausforderung stellt dabei die Menge der zu analysierenden Daten dar.

Liegt noch kein spezifisches Wissen zum analysierenden Systemverhalten vor, ist eine Abschätzung der benötigten Datenmenge für gute Prognosen nur bedingt möglich [KAC17]. Hybride Ansätze kombinieren dabei die Vorzüge des physikalischen und des datengetriebenen Ansatzes, um so die Vorhersagequalität zu maximieren [San+09].

Merkmal Zur Erstellung einer datengetriebenen Analyse werden (digitale) Informationen benötigt. Diese Informationen werden aus einzelnen Stichproben (oder auch Instanzen, engl. *samples*) gewonnen. Im Falle einer fortlaufenden Zeitreihe über mehrere Wochen könnte unter einer bestimmten Instanz die gesammelten Informationen eines bestimmten Tages verstanden werden. Diese Instanz wird wiederum durch ihre Attribute charakterisiert. Der Wert dieses spezifischen Attributs wird als Merkmal (engl. *feature*) bezeichnet. Es entsteht ein $m \times n$ Featurevektor, wobei m durch die Anzahl der Instanzen und n durch die Anzahl der Attribute ausgeprägt ist. Beispielweise seien für eine fortlaufende Zeitreihe im Attribut „Wetter“ folgende Merkmale (Featurevektor) gegeben: sonnig, bewölkt, neblig, regnerisch. Die Tabelle 2.1 zeigt die gesamte Zeitreihe mit den beiden Attribute „Wetter“ und „Tageshöchsttemperatur“. Nominale Attribute können Werte aus einer vordefinierten, endlichen Menge annehmen. Sie werden auch als kategoriale Attribute bezeichnet [WFH11, S. 39 ff.]. Dagegen sind numerische Attribute (z.B. die Tageshöchsttemperatur, vgl. Tabelle 2.1) Zahlen aus dem realen und ganzzahligen Wertebereich.

Tabelle 2.1: Tabellarische Darstellung einer Zeitreihe, bestehend aus vier Instanzen und mehreren Attributen

Sample	Datum	Wetter	Tageshöchsttemperatur
sample 1	07.05.2020	bewölkt	35°C
sample 2	14.08.2020	sonnig	28°C
sample 3	30.10.2020	neblig	12°C
sample 4	21.11.2020	regnerisch	15°C

Je größer die Menge an Merkmalen ist, desto nützlicher kann eine Selektion relevanter Merkmale für eine effiziente Erstellung eines datengetriebenen Modells sein. Im Rahmen der *Feature Selection* (oder auch *Feature Subset Selection*) können relevante Merkmale durch Filter- und Wrapper-Ansätze selektiert werden [Alp10, S. 138 f., GE03]. Dennoch bleibt zu beachten, dass einzelne Merkmale an Bedeutung gewinnen können, wenn diese in Kombination mit weiteren anderen Merkmalen auftreten [Dom12].

Überwachtes und unüberwachtes Lernen Weiterhin wird zwischen überwachten (engl. *supervised*) und unüberwachten (engl. *unsupervised*) Lernen unterschieden. Beim überwachten Lernen stehen neben den historischen Daten (Eingangsdaten) auch sogenannte *Label* zur Verfügung. Diese Label, auch als Zielwert bezeichnet (*target value*), können sowohl kategorial (z.B. „i.O.“ und „n.i.O.“) oder numerisch (metrisch) sein und sind den einzelnen Eingangsdaten zugeordnet. Für numerische Label kann mit Hilfe einer Regression

für neue Eingangsdaten ein Zielwert vorhergesagt werden. Stehen diese Label nicht zur Verfügung, handelt es sich um ein unüberwachtes Lernen. Dabei werden interne Strukturen in den historischen Daten gelernt. Diese Datensätze können so in unterschiedliche Kategorien eingeteilt werden (*clustering*).

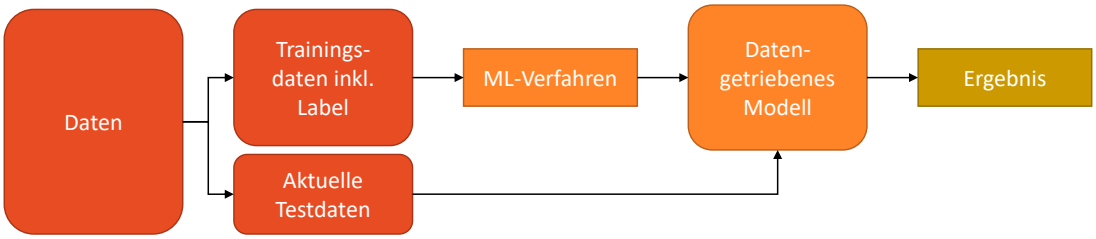


Abbildung 2.1: Schematische Darstellung des überwachten Lernens inkl. des Datenflusses

Clusteranalyse und Klassifikation Handelt es sich bei den Eingangsdaten um kategoriale (nominale) Feature, so werden mit Hilfe des Maschinellen Lernens innerhalb der Clusteranalyse neue Gruppen gebildet, die ähnliche Informationen enthalten. Dabei werden die Eingangsdaten in natürliche Gruppen (Klassen, engl. *cluster*) eingeteilt [WFH11, S. 138]. Darüber hinaus können mit Hilfe der Clusteranalyse Anomalien und versteckte Strukturen in den Eingangsdaten ermittelt werden. Ein weiterer Bedarf an Clusteranalysen besteht, wenn eine kompakte Beschreibung der Daten notwendig ist (Datenreduktion). Das repräsentativste Objekt eines Clusters kann so bestimmt werden, um die gesamte Klasse mit allen ihren Objekten zu repräsentieren [HBV01]. Dennoch kommt es vor, dass einzelne Messpunkte keiner Klasse zugeordnet werden können, diese werden auch Ausreißer genannt (engl. *outlier*).
Im Gegensatz zur Clusteranalyse wird mit Hilfe der Klassifikation eine Zuordnung der Daten in vorher fest definierten Klassen vorgenommen. Das Ziel der Klassifikation ist es, den Eingangsvektor einer der K diskreten Klassen $C_k(k = 1, ..., K)$ zuzuweisen.

Regressionsanalyse Bestehen sowohl die Eingangsdaten als auch die Zielgröße aus numerischen Werten, sind Regressionsanalysen anwendbar. Hierbei handelt es sich um ein Teilgebiet des überwachten maschinellen Lernen. Ziel ist es, eine reelle Funktion (*Regressionsfunktion*) zu ermitteln, die den Zusammenhang zwischen Zielwert und den Eingangswerten abbildet. Diese Funktion beschreibt den Zusammenhang für eine oder mehrere Variablen (*multivariates* Problem). Eine einfache Regression mit einer unabhängigen Variable sei beispielsweise die Vorhersage der Schuhgröße einer Person auf Basis ihrer Körpergröße.

Deep Learning Deep Learning (DL) oder auch tiefes Lernen, ist ein Teilgebiet des MLs, bei dem neuronale Netze (NN) benutzt werden. Diese sind sowohl zur Klassifikation, im

Rahmen der Clusteranalyse, für Regressionsanalysen oder auch für Vorhersagen von Prognosen verwendbar [Bac+18, S. 581 f.]. Der Wirkmechanismus ist angelehnt an neuronale Netze aus der Biologie.

Online- und Offline-Learning Im Bereich des maschinellen Lernen wird zwischen *Online-* und *Offline-Learning* differenziert. Beim Online-Learning handelt es sich um einen inkrementellen Lernprozess. Der Algorithmus lernt die Parameter (Modellanpassung) für jeden neu hinzugefügten Datensatz und passt diese kontinuierlich an. Je Datensequenz wird eine Vergütung oder ein Verlust zugewiesen. Dabei ist das Ziel, diese Summe der Vergütungen zu maximieren (oder die Verluste zu minimieren), um so das Modell ständig anzupassen und zu optimieren.
Der Prozess des *Offline-Learnings* betrachtet dagegen einmalig eine große Menge an Daten, auch Stapel (engl. *batch*) genannt. Diesbezüglich wird ein Modell angepasst. Dies Prozess wird *Batch Learning* genannt. Die Modellanpassungsfähigkeit ist nur beschränkt möglich, aus diesem Grund ist im Falle des Offline-Learnings ein umfassender Eingangsdatensatz umso wichtiger.

Verteilungen Neben eines unterschiedlichen Skalenniveaus der Eingangsdaten (kategorial oder metrisch) können diese auch unterschiedlich verteilt vorliegen. Kategoriale Daten lassen sich mit Hilfe von Balkendiagrammen darstellen, wohingegen metrische Daten meist durch Histogramme oder Boxplots visualisiert werden. Reale (Mess-)Daten unterliegen zufälligen Verteilungseffekten. Unter Beachtung des zentralen Grenzwertsatzes kann beim Vorliegen vieler kleiner zufälliger Effekte und einem großen Stichprobenumfang approximativ eine Normalverteilung angenommen werden [Bac+18, S. 91; Ger19, S. 250]. Hierbei folgt die stetige Zufallsvariable X einer Normalverteilung mit Erwartungswert μ und Varianz σ^2 der folgenden Dichtefunktion und hat einen glockenförmigen Verlauf:

$$f(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \qquad -\infty < x < \infty \tag{2.1}$$

Neben einer grafischen Darstellungsmöglichkeit, kann das Vorliegen einer Normalverteilung durch inferenzstatistische Testverfahren überprüft werden. Dazu steht beispielsweise der sogenannte Kolmogorov-Smirnov-Test zur Verfügung [Bac+18, S. 199].

Datenskalierung Stehen Daten von Signalen mit unterschiedlichen Einheiten, zum Beispiel eine Fahrzeuggeschwindigkeit in km/h und eine Motortemperatur in $°C$ zur Verfügung, kann es für eine bessere Vergleichbarkeit geeignet sein, diese Daten zu standardisieren [Bac+18, S. 73-74, S. 373-374; BCK12, S. 104-105; MS05, S. 45]. Dabei wird der Mittelwert $\bar{x}$ von den Datenpunkten X abgezogen und im Anschluss durch die Standardabweichung s_x geteilt. Für die neue Größe Z gilt nun folgendes: Mittelwert $\bar{z} = 0$ und Standardabweichung $s_z = 1$.

$$Z = \frac{X - \bar{x}}{s_x} \tag{2.2}$$

Die vorgestellte Standardisierung wird in der Literatur auch als *z-Transformation* bezeichnet. Neben einer Skalierung mit Hilfe des Mittelwertes und der Standardabweichung lassen sich die Daten auch über ihre Minimal- und Maximalwerte normalisieren. Bei dieser *Min-Max*-Skalierung wird ein gültiger Wertebereich festgelegt und im Anschluss werden die wahren Werten diesem neuen Gültigkeitsbereich (zum Beispiel zwischen Null und Eins) zugeordnet.

Datenskalierung Ein virtueller Sensor (engl. *soft sensor*) schätzt eine Messgröße mit Hilfe eines erstellten Modells. Das Ziel ist es, einen realen Sensor mit Hilfe des virtuellen Sensors zu ersetzen. Zum Lernen des Modells werden reale Messdaten verwendet. Diese Trainingsdaten können anwendungsabhängig über Labormessungen oder Erprobungen generiert werden. Mit Hilfe eines virtuellen Sensors sollen Muster und Gesetzmäßigkeiten zwischen diesen Mess- und Zielgrößen angelernt werden.

2.1.2 Methoden der Clusteranalyse

Die Clusteranalyse ist ein Teilgebiet des unüberwachten MLs. Nach den Autoren Jain, Murty und Flynn, sowie Aghabozorgi, Seyed Shirkhorshidi und Ying Wah lassen sich unterschiedliche Algorithmen in folgende Kategorien einteilen [JMF99; ASY15]:

- **Partitionierende Clusterverfahren** erstellen k Gruppen von n nicht gelabelten Objekten. Daraufhin ist in jeder Cluster-Gruppe mindestens ein Objekt zu finden. Die Anzahl der gefundenen Cluster wird hinsichtlich einer vorher definierten Kriteriumsfunktion optimiert. Ein partitionierendes Clusterverfahren ist beispielsweise der *k-Means* Algorithmus. Kerngedanke ist die Minimierung der Distanz (vgl. Kapitel 2.1.3) zwischen allen Objekten eines Clusters zum Cluster-Zentrum. Viele partitionierende Clusterverfahren benötigen als Initialisierungswert eine Clusteranzahl.
- **Hierarchische Clusterverfahren** ordnen jedem Objekt einen eigenen Cluster zu, welcher im Fortlauf zu einem größeren zusammengefasst wird (engl. *bottom-up*), oder die Objekte werden zunächst einem großen Cluster zugewiesen, der im Verlauf der Ausführung sukzessiv in weitere kleinere Cluster unterteilt wird (engl. *top-down*). Dabei entsteht ein Baum an Clustern, das sogenannte Dendrogram. Je nach Schnitthöhe des Dendrograms entsteht so eine bestimmte Anzahl an Clustern. Im Gegensatz zu partitionierende Clusterverfahren benötigen hierarchische Clusterverfahren keinen Initialisierungswert.
- In **dichtebasierten Clusterverfahren** werden Objekte, die räumlich dicht beieinander liegen gemeinsamen Gruppen zugeordnet. Ein Cluster wird erweitert, wenn neue Objekte sehr dicht an einem bekannten Cluster liegen (*DBSCAN* Algorithmus).
- **Netzbasierte Verfahren** quantisieren den Raum in eine endliche Anzahl an Zellen. Auf Basis dieser Zellen werden die Objekte räumlich voneinander getrennt. Beispielsweise seien hier die Algorithmen *STING* und *WaveCluster* genannt.

- **Modellbasierte Verfahren** teilen die Daten mit Hilfe erstellter Modelle in unterschiedliche Gruppen ein. *Gaussian Mixture Models* nutzen dazu statistische Modelle, wohingegen *Self-Organizing Maps (SOM)* neuronale Netze verwenden. Für eine effiziente Gruppierung der Eingangsdaten sind zahlreiche Parameter und Annahmen festzulegen. Außerdem ist eine Skalierbarkeit großer Datenmengen nur begrenzt möglich.
- **Kombinierte Verfahren** beschreiben ein Konzept, mehrere Clusteringverfahren miteinander zu verknüpfen, um so die Nachteile einzelner Clusterverfahren ausgleichen oder auch große Datenmengen gruppieren zu können, wie z.B. beim *Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH)* Algorithmus.

Weitere Clusterverfahren und Abwandlungen der genannten Konzepte beschreiben die Autoren Witten, Frank und Hall in [WFH11].
Teilweise können den genannten Clusteralgorithmen auch Initialisierungsparameter übergeben werden. Dazu zählen u.a. die Anzahl oder die Dichte der Klassen. Jedoch sei an dieser Stelle darauf hingewiesen, dass optimale Initialisierungsparameter nicht zwangsläufig ideale Partitionierungen finden. Clusteringverfahren definieren Gruppierungen von Objekten, die vorher nicht bekannt sind. Eine Überprüfung der Clusterlösungen ist deshalb stets zu evaluieren [RLR98; HBV01]. Die *Güte* eines Clusteralgorithmus definiert eine Größe zur Beurteilung der gefundenen Clusterlösung. Die Güte ist sowohl von einem *internen* als auch *externen Index* messbar.
Der *interne Index* beschreibt die Güte der gefundene Klassenstruktur ohne externe Informationen. Um so eine unüberwachte Validierung vornehmen zu können, werden die Kompaktheit, Separierung und Dichte der einzelnen Klassen zueinander betrachtet. Die Abbildung 2.2 zeigt beispielhaft diese drei genannten Validierungskonzepte. Der interne Index kann benutzt werden, um frühzeitig eine Anzahl der Klassen vorzuschlagen. Es existieren dabei eine Vielzahl an internen Indizes, wie z.B.: *Sum of Squared Error*, *Silhouetten-Koeffizient*, *Davies-Bouldin*-Index, *Calinski-Harabasz*-Index, *Dunn*-Index, *R-squared*-Index, *Hubert Levin*-Index, *Krzanowski-Lai*-Index und der *Hartigan*-Index [ASY15; WK18]. Dem-

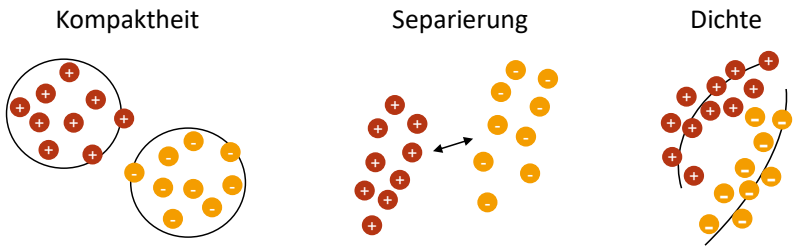


Abbildung 2.2: Darstellung verschiedener Konzepte für interne Clustervalidierungsindizes, in Anlehnung an [HKK05]

gegenüber wird der *externe Index* verwendet, um die Güte der Clusterlösung im Vergleich mit einer Referenzpartitionierung zu bestimmen. Zunächst sind vom Fachexperten die wahren Klassen für die Eingangsdatenmenge zu bestimmen. Es ist denkbar, dass sich einzelne Objekte mehreren Klassen zuordnen lassen. Im Rahmen der Evaluierung kann diese nicht explizite Klassenzuordnung einzelner Objekte zu einem entsprechenden Vorhersagefehler

führen. Dennoch kann der externe Index zur Beurteilung unterschiedlicher Clusterverfahren verwendet werden. Auch hierbei existiert eine Vielzahl an Methoden: *Cluster Similarity Measure*, *Fowlkes-Mallow Index (FMI)*, *Jaccard Score*, *Rand Index* oder *F-measure* [WK18, S. 175 f., HKK05].

2.1.3 Methoden des überwachten Lernens

Im Gegensatz zum unüberwachten Lernen steht beim überwachten Lernen eine Zielgröße zur Verfügung. Diese Zielgröße kann unter Zuhilfenahme der Eingangsdaten durch unterschiedliche Methoden bestimmt werden, die je nach Aufgabenstellung und Zusammensetzung der Eingangsdaten unterschiedlich gute Ergebnisse liefern. Die Literatur unterteilt die unterschiedlichen Methoden des überwachten Lernens in folgende Bereiche:

- Statistische Techniken (Lineare Regression)
- Support Vektor Maschinen (*SVM*)
- Lernen von Bayes-Netzen
- Entscheidungsbäume (engl. *decision trees*, RF)
- k-Nearest Neighbor-Methode (engl. k-Nearest Neighbor, *kNN*)
- Neuronale Netze (NN)

Wie bereits erwähnt, unterteilt sich das Gebiet des überwachten Lernens sowohl in die Klassifikation als auch Regression. Da in dieser Arbeit eine Alterungsvorhersage mit numerischen Werten vorgenommen werden soll, werden im weiteren Verlauf die Methoden des überwachten Lernens betrachtet, die sich für eine Regression eignen. Die Tabelle 2.2 stellt

Tabelle 2.2: Auflistung von Methoden des überwachten Lernens zur Möglichkeit der Klassifikation und Regression

Methode	**Klassifikation**	**Regression**
Statistische Techniken	-	X
Support Vektor Maschinen (SVMs)	X	X
Lernen von Bayes-Netzen	X	X
Entscheidungsbäume (RF)	X	(X)
k-Nearest Neighbor-Methoden (kNN)	X	X
Neuronale Netze (NN)	X	X
Assoziationsregeln zur Klassifikation	X	-

gegenüber, welche Methoden für eine Regression geeignet sind [Ert16, S. 258].
Im Folgenden werden die genannten Methoden des überwachten Lernen erläutert, die für eine Regression anwendbar sind.

Statistische Techniken Liegt ein linearer Zusammenhang der Eingangsdaten zum Zielwert vor, sind Modelle der Klasse der *linearen Regressionsmodelle* anwendbar. Die einfache lineare Regression ist das Grundmodell dieser Klasse (vgl. Gleichung 2.3) und berücksichtigt nur eine Eingangsvariable. Bei einer linearen Regression wird der Zielwert y unter Berücksichtigung des konstanten Gliedes b_0 und des Regressionskoeffizienten b_1 der Eingangsvariable x geschätzt. Dabei handelt es sich bei $\hat{y}$ nicht um beobachtete Werte, sondern um die aus dem Modell vorhergesagten. So gilt für eine einzelne Beobachtung i:

$$\hat{y}_i = b_0 + b_1 x_i \tag{2.3}$$

Die Punkte der Beobachtung folgen der linearen Regressionsgeraden normalerweise nicht ideal. Grund dafür sind zufällige Einflussgrößen und Messungenauigkeiten. Daraus folgt, dass sich beobachtete und geschätzte y-Werte unterscheiden. Diese Differenz wird in der Literatur als Residuum oder Residualgröße e bezeichnet [Bac+18, S. 68]. Es gilt dabei:

$$y = \hat{y} + e \tag{2.4}$$

Oft zeigen sich dagegen Zusammenhänge, die komplexer und nicht nur von einer einzigen Variable abhängig sind. Die multiple lineare Regression (MLR) (vgl. Gleichung 2.5) verwendet zur Vorhersage der Werte mehrere Eingangsvariablen.

$$\hat{y}(x_1, ..., x_N) = b_0 + \sum_{n=1}^{N} b_n x_n \tag{2.5}$$

Das Problem einer möglichen Überanpassung des Modells an die Eingangsdaten wird im Rahmen der Regularisierung gelöst. Dabei wird ein Strafterm zur Fehlerfunktion hinzugefügt. Über einen Regularisierungsparameter kann die Gewichtung dieses Strafterms angepasst werden [Bis06, S. 144]. Es wird zwischen der L_1-Regularisierung (auch als LASSO-Regularisierung bezeichnet, vgl. Gleichung 2.6) und der L_2-Regularisierung (auch als Ridge Regularisierung bezeichnet oder Kleinst-Quadrate-Kriterium, vgl. Gleichung 2.7) unterschieden [Bac+18, S. 70 f., QB17; Bis06, S. 145 f.]. Die L_1-Regularisierung ist gegeben durch:

$$\sum_{j=1}^{N} |e_j| \tag{2.6}$$

Die L_2-Regularisierung ist im Folgenden beschrieben:

$$\sum_{j=1}^{N} e_j^2 \tag{2.7}$$

Im Rahmen der L_1-Norm (Wurzel aus 2.6) werden die Abweichungen gleichgewichtig und unter Verwendung der L_2-Norm (Wurzel aus 2.7) quadratisch einbezogen [Bac+18, S. 450 f.]. Beide Varianten basieren auf dem Konzept der L_p-Norm (vgl. dazu [Bis06, S. 145 f.]), welche für $0 < p < \infty$ wie folgt definiert ist:

$$||x||_p = \left(\sum |e_j|^p\right)^{1/p} \tag{2.8}$$

Ein wesentlicher Aspekt einer MLR ist auch die *Multikollinearität*. Hierunter wird die direkte Abhängigkeit von statistischen Variablen untereinander verstanden. Wenn Variablen signifikant miteinander korrelieren, kann die fehlerfreie Interpretation der Koeffizienten beeinträchtigt werden [Sch12, S. 451ff]. Bei einer hohen Multikollinearität ist mit einer erhöhten Informationsredundanz zu rechnen. Das heißt aber auch umgekehrt, dass sich anhand dieser Informationen nicht mehr eindeutig auf die ursprünglichen Variablen rückschließen lässt [Bac+18, S. 99].

Support Vektor Maschinen Support Vektor Maschinen (SVMs) erstellen zur Trennung von verschiedenen Klassen eine Hyperebene. Sie werden bei Klassifikationsproblemen angewendet (vgl. Abbildung 2.3). Die Grundidee einer SVM ist die Unterteilung der Eingangsdaten in zwei Klassen. Die teilende Gerade wird als Hyperebene bezeichnet und darf keine Punkte der Eingangsdaten schneiden [Zha17]. Ziel ist es, dass die Hyperebene einen maximalen Abstand zu den Datenpunkten aufweist. Durch Veränderung der Kernelfunktion können SVMs auch nichtlineare Probleme lösen. Diesbezüglich wird die Eingangsmenge an Daten mit Hilfe der radiale Basisfunktion (RBF) in eine höhere Dimension transferiert, in der die Daten von einer Hyperebene separiert werden können. Die Support Vektor Regression (SVR) ist eine Erweiterung der SVM und kann komplexere Zusammenhänge als die lineare Regression abbilden. Ein großer Vorteil der SVR gegenüber einer linearen Regression liegt darin, dass das Ergebnis nicht von der Dimension der Eingangsdaten abhängig ist [Kle+17].

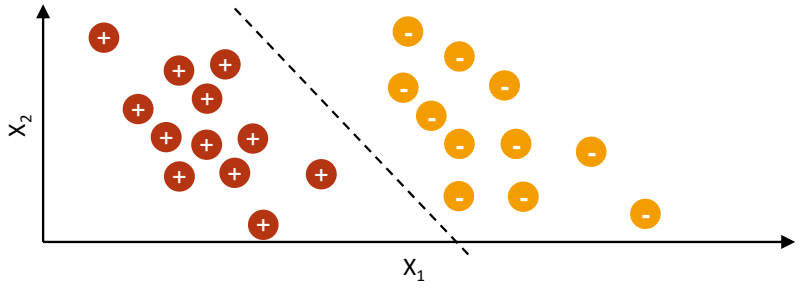


Abbildung 2.3: Darstellung der Trennung zweier Klassen mit Hilfe einer SVM, in Anlehnung an [Ert16, S. 299]

Bayes-Netze Das *bayessche Netz* ist ein gerichteter Graph mit Knoten und Kanten. Diese beschreiben die Zufallsvariablen und deren bedingte Abhängigkeiten. Jedem Knoten dieses Graphen ist eine Wahrscheinlichkeitsverteilung zugeordnet [WFH11, S. 261 ff.]. Mit Hilfe von Trainingsdaten kann die a priori Wahrscheinlichkeit der Klassen und die a posteriori Wahrscheinlichkeit berechnet werden. Mit Hilfe dieser Wahrscheinlichkeiten können so die Klassen von neuen Eingangsdaten prädiziert werden. Im Rahmen der bayesschen Regression wird nicht ein y-Wert vorhergesagt, wie bei der linearen Regression; stattdessen wird angenommen, dass es sich um eine Wahrscheinlichkeitsverteilung handelt. Bei der Anwendung der bayesschen Methoden wird vorausgesetzt, dass einzelne Merkmale der

Eingangsdaten statistisch unabhängig sind. In der Realität kann diese statistische Unabhängigkeit jedoch nicht immer gewährleistet sein. Dennoch zeigt die Literatur eine effektive Anwendbarkeit des Algorithmus [Ert16].

Entscheidungsbäume Diese Methode wird auch *Decision Tree* genannt und bildet Zusammenhänge von Entscheidungsregeln in Form eines gerichteten Baumes ab. Ein Entscheidungsbaum kann nicht nur als Zuweisung einer Klasse genutzt werden, sondern auch um numerische Werte zu prädizieren. Aus diesem Grund können Entscheidungsbäume auch zur Regression genutzt werden. Im Bereich des überwachten Lernens werden sie unter dem Namen *Random Forest* (*RF*) eingesetzt. Es werden mehrere unkorrelierte Entscheidungsbäume genutzt, um eine Klassifikation vorzunehmen. Dabei wird die Trainingsdatenmenge für einzelne Entscheidungsbäume randomisiert. Ein Entscheidungsbaum ist ein Graph mit Verbindungen und Knoten. Ein Knoten am Ende des Baumes wird als Blatt bezeichnet und gibt die Antwort auf eine Klassifikationsfrage. Der daraus entstehende Wald an Entscheidungsbäumen sammelt die Entscheidungen eines jeden Entscheidungsbaumes für ein Klassifikationsproblem. Die schlussendliche Klassifikation wird für die Klasse mit den meisten Befürwortern vorgenommen. Aufgrund der großen Anzahl an Entscheidungsbäumen und der Randomisierung der Eingangsdaten kann die Varianz der Ergebnisse reduziert werden [HTF17, S. 587]. Das Training des Modells eines Entscheidungsbaumes erfolgt mit Hilfe einer Teilmenge des Trainingsdatensatzes. Dabei teilt der Baum, ausgehend vom Wurzelknoten, diese Daten in weitere Kindteilmengen rekursiv auf. Ein vorher definiertes Kriterium lässt den Baum nicht weiter wachsen. Wird der letzte Knoten (Blattknoten) erreicht, so wird dort die jeweilige Systemantwort gespeichert. [HH17, S. 609 ff.]
Ein Baum kann dabei durch verschiedene Algorithmen erstellt werden. Ein bekannter Vertreter dieser Algorithmen ist der ID3-Algorithmus. Die Trainingsmengen werden dabei hinsichtlich ihrer Entropie aufgeteilt, sodass die Homogenität der Teilmengen möglichst hoch ist. Die ursprüngliche Form des Random Forest Algorithmus ist der Random Decision Forest [Tin95] von Tin Kam Ho. Hierbei werden zufällig gewählte Merkmale der Trainingsdaten einem Baum zugeordnet (Random Subspace Method) [SW17]. Später wurde dieses Verfahren von Breiman um den Bagging-Algorithmus (Bootstrap Aggregating) erweitert [Bre01]. Wie oben beschrieben, werden hierbei für jeden Baum eine zufällige Anzahl an Samples der Trainingsmenge genutzt.

k-Nearest Neighbor-Methoden Bei der *Nearest Neighbor*-Methode wird mit Hilfe von benachbarten Datenpunkten versucht Wissen aus den Eingangsdaten abzuleiten. Im Falle einer Klassifikation wird zur Vorhersage eines neuen Datenpunktes der nächste Nachbar betrachtet (engl. *Nearest Neighbor*). In der gelernten Datenmenge seien zahlreiche Punkte (Eingangsdaten) und deren zugehörigen Klassifikationen gespeichert. Soll nun für einen neuen Datenpunkt eine Vorhersage getroffen werden, wird der nächste Nachbar innerhalb dieser gelernten Datenmenge betrachtet. Um Überanpassungen und Fehlentscheidungen zu vermeiden, kann mit Hilfe von k nächsten Nachbarn (engl. *k-Nearest Neighbor*) ein Mehrheitsentscheid durchgeführt werden. Neben einer Klassifikation lassen sich k-Nearest

Neighbor-Methoden auch auf Regressionsprobleme anwenden, durch Berechnung des mittleren Prädiktionswertes in Bezug auf die Distanz des Punktes zu allen k Nachbarn. [Ert16, S. 206-210]
Nearest Neighbor-Methoden gehören zu den Methoden des faulen Lernens (engl. *Lazy Learning*). Hierbei ist die Phase der Prädiktion deutlich aufwändiger als die Lernphase. Während des Lernens werden die Daten lediglich gespeichert. Dagegen können die Verfahren des eifrigen Lernens (engl. *Eager Learning*) Prädiktionen sehr effizient durchführen. Allerdings besitzen diese einen deutlich aufwändigeren Lernprozess als die Methoden des Lazy Learnings [Ert16, S. 214].

Neuronale Netze Neben den genannten Methoden wie SVR, k-Nearest Neighbor (kNN) oder Bayes-Regressionen, können auch NN für Regressionsanalysen verwendet werden. Ein NN besteht aus einer Eingabeschicht (Input-Layer), einer (oder mehreren) mittleren Schichten (Hidden-Layer) und einer Ausgabeschicht (Output-Layer). Eine Schicht besteht wiederum aus mehreren Neuronen. Eingangssignale treffen auf ein Neuron, dort werden sie zunächst mit Hilfe einer Propagierungsfunktion zu einem Wert verdichtet. Im Anschluss erfolgt die Berechnung des tatsächlichen Ausgabewerts des Neurons über eine Aktivierungsfunktion [Roj93, S. 32 f.]. Um Nichtlinearitäten und nicht relevante Informationen modellseitig auszublenden, werden in einem NN Aktivierungsfunktionen verwendet. Hierbei werden einzelne Neuronen erst ab einem Grenzwert aktiviert. Typische Aktivierungsfunktionen sind unter anderem Sigmoid, rectified linear unit (ReLu) oder Tanh [MG16; Ert16, S. 269 f.]. Die Gewichte der Propagierungsfunktion und Parameter der Aktivierungsfunktion sind Teil des Lernprozesses. Diese werden solange verändert, bis die Zielgröße bestmöglich abgebildet wird. Die Lernrate beschreibt, wie die Parameter im nächsten Lernschritt verändert werden [Bac+18, S. 581 f.]. Das Training des NNs besteht aus einer Reihe von Daten, den sogenannten *Samples*. Die Chargengröße (engl. *batch size*) legt fest, nach wie vielen Daten-Samples die internen Modellparameter (zum Beispiel die Gewichte der Propagierungsfunktion) des Netzes aktualisiert werden. Ein kompletter Durchlauf wird als *Epoche* bezeichnet. Ein weiterer Parameter gibt an, nach wie vielen Epochen der Lernprozess spätestens abgeschlossen sein soll. Typischerweise werden die Gewichte in Abhängigkeit der eingestellten Aktivierungsfunktion randomisiert initialisiert. Dabei können geeignetere Initialwerte schneller zu Trainingserfolgen führen, als weniger geeignete Initialwerte.
Über eine Kostenfunktion wird der Fehler zwischen dem Zielwert und dem gelernten Wert bestimmt. Ziel ist die Minimierung dieser Kostenfunktion über mehrere Epochen. Die Anpassung der Lernrate trägt zur Minimierung der Kostenfunktion entscheidend bei. Bei Wahl einer zu großen Lernrate kann das gesamte Verfahren divergieren. Wird dagegen eine zu kleine Lernrate gewählt, kann eine Konvergenz möglicherweise erst nach einer Vielzahl von Iterationen erreicht werden. Üblicherweise wird über das Gradientenverfahren die Richtung der Anpassung der Gewichte gesteuert [LeC+12]. Beim SGD (engl. *stochastic gradient descent*) wird eine stochastische Schätzung der Anpassung der Gewichte vorgenommen. Neben den genannten Gradientenverfahren mit fester Lernrate können weitere Verfahren mit adaptiven Lernraten implementiert werden. Diesbezüglich stehen in der Literatur weitere Optimierungsfunktionen zur Verfügung: Resilient Backpropagation (Rprop), RMSProp, Adam, AdaGrad [Rud16].

Bei dem Optimierungsalgorithmus *Rprop* handelt es sich um ein iteratives Verfahren. Hierbei wird auch die Gewichtsänderung der vorherigen Iteration mit in die Berechnung einbezogen. Diese ist abhängig vom Vorzeichen des Gradienten. Ähnlich wie Rprop werden auch beim *RMSprop* vorherige Gewichte betrachtet. Dabei wird die Lernrate für das aktuelle Gewicht durch die durchschnittlichen vorherigen Gewichte geteilt. Der *Adam*-Optimierer ist ein weiteres Verfahren zur adaptiven Momentenanpassung. Dazu werden die Gewichte der ersten und zweiten Ordnung, als auch dessen Quadrate genutzt. *AdaGrad* passt die Lernrate entsprechend der Häufigkeit des Parameterauftretens an. Tritt ein Parameter häufiger auf, so wird eine geringere Anpassung durchgeführt. [Rud16; LeC+12]
Zur Vermeidung einer Überanpassung kann das neuronale Netz um ein *Dropout* erweitert werden. Dabei wird eine bestimmte Anzahl an Neuronen innerhalb der Schichten ausgeschaltet und für den weiteren Lernprozess nicht weiter betrachtet. Desweiteren können neuronale Netze auch über eine Rückkopplung erweitert werden, dies wird als rekurrentes neuronales Netz (RNN) bezeichnet. Ein solches Vorgehen erlaubt die Abbildung eines temporären dynamischen Modellverhaltens.

2.1.4 Kenngrößen des Zusammenhangs und Prognosegütemaße

In diesem Abschnitt werden Kenngrößen des Zusammenhangs und Prognosegütemaße vorgestellt. Statistische Variablen können dabei in einem unterschiedlichen Skalenniveau vorliegen: kategoriale (nominal, ordinal) oder metrische Merkmale. Diese statistischen Variablen können nun hinsichtlich ihres Zusammenhangs (vgl. *Korrelation und Bestimmtheitsmaß*, Kapitel 2.1.4) analysiert werden [Sch12]. Es werden nur die Kenngrößen vorgestellt, bei denen ein metrisches Skalenniveau vorliegt.
Im weiteren Verlauf soll nicht nur der Zusammenhang von einzelnen statistischen Größen betrachtet, sondern es sollen auch die Methoden des überwachten Lernens hinsichtlich der Prognose bewertet werden. Im Rahmen der Regression lassen sich die vorhergesagten Werte (Prädiktionswerte) mit den wahren Werten (Beobachtungswerte) vergleichen. Um diese Vorhersage bewerten zu können, werden in diesem Kapitel *Prognosegütemaße* (engl. *metric* oder *score*, vgl. Kapitel 2.1.4) eingeführt. Mit Hilfe dieser Metriken ist die Prädiktionsgüte bzw. Prognosegüte der verwendeten Methode bestimmbar. Neben einer reinen Bewertung der angewandten Methode können mit Hilfe dieses Maßes unterschiedliche Methoden miteinander verglichen werden.

Korrelation und Bestimmtheitsmaß

Der Zusammenhang von metrischen Variablen ist mit Hilfe der *Kovarianz* und der *Korrelation* bestimmbar. Die Kovarianz Cov_{xy} von zwei Variablen x und y wird wie folgt berechnet [Sch12, S. 92ff]:

$$Cov_{xy} = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y}) \tag{2.9}$$

Die Kovarianz ist eine Kenngröße zur Bestimmung des Zusammenhangs von x und y. Im Vergleich zur Kovarianz ist die Korrelation nicht von der Maßeinheit der Variable abhängig. Der Korrelationskoeffizient R von Bravais-Pearson beschreibt den linearen Zusammenhang der Variablen x und y. Er liegt zwischen den Werten -1 und 1. Je weiter sich der Wert von 0 entfernt, desto größer wird der lineare Zusammenhang. Ein positiver Wert beschreibt dabei einen positiven linearen Zusammenhang und ein negativer Wert einen negativen linearen Zusammenhang. Der Korrelationskoeffizient R von Bravais-Pearson wird aus der Kovarianz Cov_{xy} und den beiden Standardabweichungen s_x und s_y gebildet [Sch12, S. 95]:

$$R = \frac{Cov_{xy}}{s_x \cdot s_y} \tag{2.10}$$

In der Literatur wird der Betrag des Korrelationskoeffizienten R als Maß eines Zusammenhangs zwischen zwei numerischen Merkmalen angegeben. Die Autoren Schlittgen, sowie Cramer und Kamps sind sich einig, dass sich eine hohe Korrelation ab einem Wert von $R \geq 0.8$ einstellt [Sch12, S. 97; CK20, S. 111]. Je nach Anwendungsgebiet und vorliegender Daten kann sich die Grenze für einen signifikanten Zusammenhang leicht verschieben. Das *Bestimmtheitsmaß* R^2 (engl. *coefficient of determination*) beurteilt die Anpassungsgüte von Beobachtungswerten an eine Regressionsfunktion. Es ist ein Maß zur Beschreibung eines linearen Zusammenhangs von Prädiktions- und Beobachtungswerten. Der Wert des Bestimmtheitsmaßes ist dimensionslos und beträgt maximal Eins. Je geringer die Streuung ist, desto größer wird der Wert des Bestimmtheitsmaßes. Ein negativer Wert des Bestimmtheitsmaßes beschreibt dabei, dass eine empirische Verteilung der Variablen eine bessere Anpassung liefern würde als das aktuell verwendete Modell. Zur Berechnung wird die nicht erklärte Quadratsumme (engl. *sum of squared residuals (SSR)*) mit der totalen Quadratsumme (engl. *total sum of squares (SST)*) ins Verhältnis gesetzt [Bac+18, S. 77]. Wird die untersuchte Streuung der Daten durch eine lineare Regression erklärt, so entspricht das Bestimmtheitsmaß dem Quadrat des Bravais-Pearson-Korrelationskoeffizienten [Sch12, S. 108]. Das Bestimmtheitsmaß R^2 ist wie folgt definiert:

$$R^2 = 1 - \frac{SSR}{SST} = 1 - \frac{\sum_i^N (y_i - \hat{y}_i)^2}{\sum_i^N (y_i - \bar{y})^2} \tag{2.11}$$

Bei Anwendung von Regressionsfunktionen entstehen Prädiktionswerte, die den wahren Beobachtungswerten gegenüber stehen. In dem Kapitel 2.1.4 werden unterschiedliche Prognosegütemaße vorgestellt, die eine Metrik liefern, wie die Güte der Vorhersage einer Regression bestimmt werden kann.

Prognosegütemaße

Neben dem Bestimmtheitsmaß R^2 lässt sich die Güte einer Regressionprognose mit weiteren, in diesem Kapitel vorgestellten Metriken, bestimmen. Es werden *Grundformen*, *Mischformen* und *alternative* Prognosegütemaße nach [Alb+09, S. 548 ff.] vorgestellt.

- Die **Grundformen** der Prognosegütemaße sind beschrieben als: Der einfache (Mean Error (ME)), der absolute (Mean Absolute Error (MAE)) und der relative Prognosefehler (Mean Percentage Error (MPE)).

- Die **Mischformen** sind: Der durchschnittliche absolute prozentuale Prognosefehler (engl. Mean Absolute Percentage Error (MAPE)), der mittlere quadratische Fehler (Mean Squared Error (MSE)) und die Quadratwurzel des mittleren quadratischen Fehlers (Root Mean Square Error (RMSE)).
- Als **alternatives Prognosegütemaß** wird der Median Absolute Percentage Error (MdAPE) genannt.

Grundformen der Prognosegütemaße Der einfache Prognosefehler (ME) bestimmt die durchschnittliche Abweichung von prädizierten und beobachteten Werten. Negative und positive Abweichungen können sich aufheben, sodass $ME = 0$ nicht zwingend bedeutet, dass es sich um eine perfekte Prognose handelt. Dieser Nachteil wird mit Hilfe des MAE behoben, indem der Betrag der Abweichung gebildet wird. Der relative Prognosefehler MPE gibt dagegen eine relative Abweichung an und ist deshalb dimensionslos, im Vergleich zu ME und MAE. [Alb+09, S. 548 ff.]
Der mittlere absolute Fehler (engl. Mean Absolute Error, *MAE*) beschreibt eine größenabhängige Abweichung zwischen den prädizierten und beobachteten Werten. Dieser Fehler wird über alle beobachteten Instanzen gemittelt. Da es sich um eine absolute Angabe handelt, ist eine Aussage bezüglich der Abweichungsrichtung nicht möglich.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|\hat{y}_i - y_i| \tag{2.12}$$

Mischformen der Prognosegütemaße Die Mischformen der Prognosegütemaße beschreiben Abwandlungen der vorgestellten Grundformen. Der MAPE bestimmt den durchschnittlichen absoluten prozentualen Fehler von prognostizierten und beobachteten Werten. Er kombiniert die Vorteile von MAE und MPE. Dabei ist der MAPE größenunabhängig (vgl. MPE) und gleichzeitig lässt sich das Ergebnis eindeutig interpretieren (vgl. MAE).

$$MAPE = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{\hat{y}_i - y_i}{y_i}\right| \cdot 100\% \tag{2.13}$$

Der MSE bestimmt die quadratische Abweichung zwischen Prädiktions- und Beobachtungswert. Größere Abweichungen werden hierbei stärker gewichtet als kleinere Abweichungen. Durch Bildung des Quadrates können die Datenpunkte vorzeichenunabhängig miteinander verglichen werden, allerdings geht dadurch der Bezug zur Ausgangsgröße verloren. Der RMSE versucht dagegen diesen Nachteil zu beheben, indem die Quadratwurzel des MSE gebildet wird und so ein Bezug zur Ausgangsgröße gegeben ist. Ein kleiner Wert des RMSEs beschreibt dabei eine hohe Modellgüte. Der RMSE orientiert sich an der Größenordnung der Eingangsdaten und bietet dadurch eine hohe Vergleichbarkeit.

$$RMSE = \sqrt{MSE(y,\hat{y})} = \sqrt{\frac{1}{N}\sum_{i}^{N}(y_i - \hat{y}_i)^2} \tag{2.14}$$

Alternative Prognosegütemaße Die bisher vorgestellten Prognosegütemaße werden unter Zuhilfenahme des Mittelwertes berechnet. Der MdAPE wird mit Hilfe des Medians des absoluten prozentualen Fehlers (engl. *APE*) bestimmt. Nach den Autoren Armstrong und Collopy ist der MdAPE auf Grund der Verwendung des Medians robust gegenüber Ausreißern [AC92]. Für weitere Informationen zu diesen und weiteren Prognosegütemaßen sei auf folgende Literatur verwiesen: [Bar09], [MG16] und [HK06].

Nicht alle vorgestellten Metriken eignen sich für jede Problemstellung. Die Verwendung unterschiedlicher Metriken weisen unterschiedliche Vor- und Nachteile auf [Alb+09, S.556 f., Knö18]. Die folgenden Gütemaße sind dimensionslos: R2, MAPE, MdAPE. Dagegen besitzen folgende Maße die Größenordnung der Eingangsdaten: MSE, MAE und RMSE.
Der MAE ist weniger sensitiv gegenüber Ausreißern als der RMSE, da er zur Berechnung die absoluten Werten verwendet und diese nicht quadriert [Liu+17]. Der RMSE ist in den Studien ein besonders häufig verwendetes Gütemaß [Ric+11]. Der MdAPE ist auf Grund der Berechnung des Medians robust gegenüber Ausreißern [AC92]. Eine große Stärke des Prognosegütemaßes MAPE ist seine skalenunabhängige Darstellung und einfache Interpretierbarkeit. Dennoch können sehr kleine wahre Werte dazu führen, dass der MAPE die Prognosegüte verzerrt bewertet [HK06].

2.2 Zeitreihen

Die in dieser Arbeit untersuchten Daten sind Zeitreihen. Sie werden in unterschiedlichen Fahrzeugen mit Hilfe von Fahrzeugdatenloggern aufgezeichnet. In diesem Abschnitt werden die Grundlagen zu Zeitreihen und deren Analyse vorgestellt. Des Weiteren werden die Grundlagen der Steuergerätekommunikation innerhalb des Fahrzeugnetzwerkes (hier CAN) beschrieben.

2.2.1 Definition einer Zeitreihe

Unter einer *univariaten* Zeitreihe wird eine chronologische Reihe numerischer Größen verstanden [MT19]. Es wird also zu jedem Sample dieser Reihe sowohl die Zeitinformation t, als auch die Messgröße y erfasst. Die Länge T der Reihe gibt an, wie viele Samples diese Zeitreihe beinhaltet. Werden mehrere Messsignale aufgezeichnet, so wird dies als *multivariate* Zeitreihe bezeichnet. In der Abbildung 2.4 ist eine univariate Zeitreihe von den Messdaten gezeigt. Es werden Messwerte der Fahrzeuggeschwindigkeit zu unterschied-

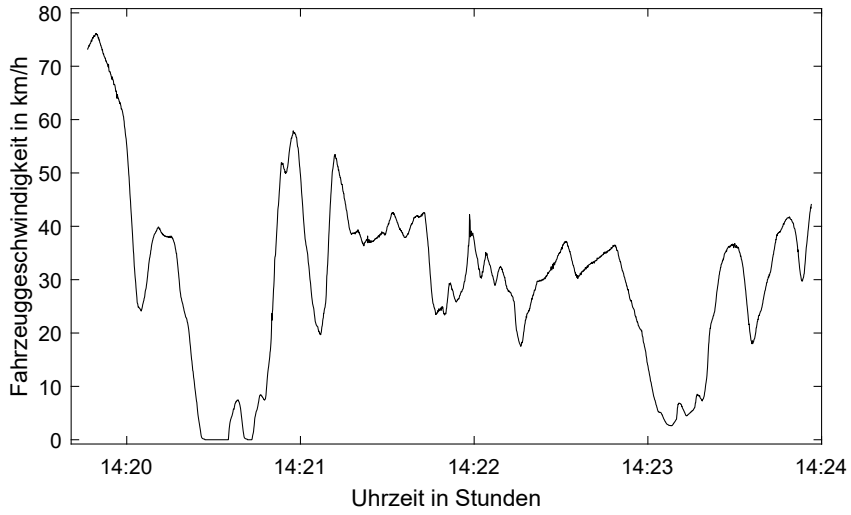


Abbildung 2.4: Darstellung der Fahrzeuggeschwindigkeit als Zeitreihe. Der dargestellte Ausschnitt zeigt etwa 4 Minuten Messdaten

lichen Zeitpunkten dargestellt. Die Zeitreihe beinhaltet 2500 Samples ($T = 2500$) bei einer Aufnahmezeit von etwas mehr als 4 Minuten.
In der Analyse univariater Zeitreihen wird die Zeitreihe in spezifische Komponenten zerlegt. Die Autoren Metz und Thome unterscheiden in [MT19] zwischen unterschiedlichen Verlaufsmustern dieser Komponenten: Trend, Zyklus, Saison, Strukturbrüchen und Ausreißern. Die Abbildung 2.5 zeigt diese Verlaufsmuster beispielhaft. Ausreißer werden die Datenpunkte genannte, die sich nicht in die Reihe der anderen Datenpunkte einsortieren lassen. Während der Zeitreihen-Zerlegung können sich einzelne Komponenten auch überlagern. Es

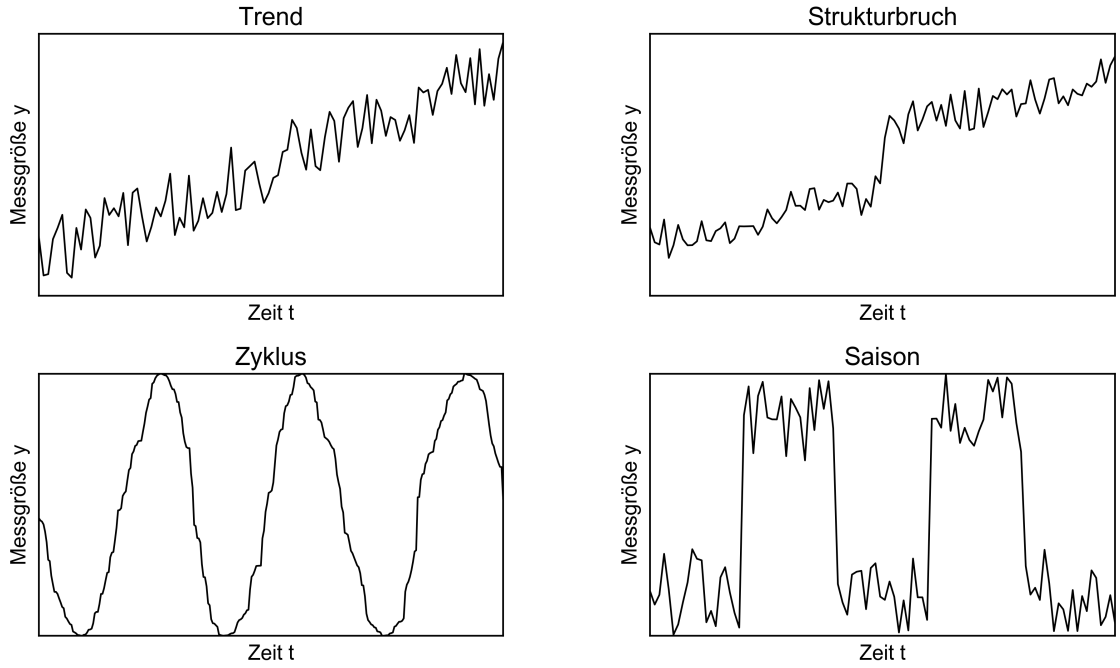


Abbildung 2.5: Darstellung unterschiedlicher Komponenten in Zeitreihen: Trend, Strukturbruch, Zyklus und Saison

wird zwischen kurzfristigen (*lokalen*) und globalen Änderungen differenziert, die die gesamte Zeitreihe betreffen. Lokale Trenderscheinungen treten kurzfristig auf und sind im Verlauf der gesamten Zeitreihe nicht wiederauffindbar. Eine nicht zweifelsfreie Zuweisung eines Einflusses wird der Restkomponente zugeordnet. Hierunter fallen auch die Einflüsse von zufälligen Ereignissen (z.B. Messungenauigkeiten). [MT19]
Für aussagekräftige Modellerstellungen ist eine möglichst vollständige Datenbasis erforderlich. Kauermann unterscheidet in [Kau19] zwischen dem Fehlen einzelner Zeitpunkte innerhalb der gesamten Zeitreihe und dem Fehlen ganzer Spalten. Eine unterschiedliche Konfiguration der verwendeten Datenaufzeichnungsgeräte führt zu einer unterschiedlichen Menge an aufgezeichneten Signalen eines Fahrzeugs. Neben Änderungen in der Datenmengenkonsistenz auf Grund von unterschiedlichen Konfigurationen können auch von Beginn an Variablen nicht aufgezeichnet werden, weil sie zum Beispiel nicht messbar sind. Wird ein komplexes datengetriebenes Modell erstellt, so ist dieses typischerweise nicht nur von einer sondern von mehreren Variablen abhängig. Es ist darauf zu achten, dass eine zu analysierende Variable (Zielgröße) von mehreren Einflussvariablen abhängig sein kann. Das Nichtvorhandensein von Einflussvariablen in der Datenmenge kann zu fehlerhaften Vorhersagen führen [Kau19].

Eigenschaften einer Zeitreihe Eine Zeitreihe lässt sich nicht nur durch die unterschiedlichen Verlaufsmuster, sondern auch mit Hilfe von statistischen Merkmalen charakterisieren. Dazu wird ein bestimmter Ausschnitt (Sequenz) der Zeitreihe betrachtet. Innerhalb dieser

Sequenz können statistische Merkmale wie z.B. Mittelwert, Maxima-Werte und Quantil-Werte berechnet werden (vgl. dazu auch [PHR16]). Die Abbildung 2.6 zeigt beispielhaft eine solche Zeitreihencharakterisierung für eine ausgewählte Sequenz.

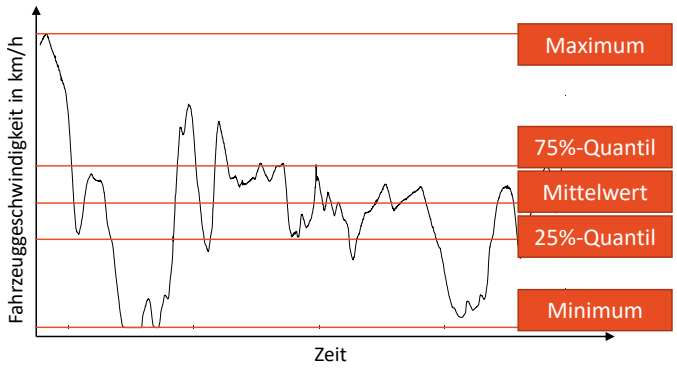


Abbildung 2.6: Charakterisierung einer Zeitreihe durch statistische Merkmale

Repräsentationen einer Zeitreihe Unterschiedliche Zeitreihen lassen sich in ihrer herkömmlichen Form nur bedingt miteinander vergleichen [Lin+03]. Sie können in unterschiedlichen zeitlichen Auflösungen vorliegen. Des Weiteren können sie kategoriale sowie numerische Werte beinhalten. Nach Lin u. a. werden Zeitreihenrepräsentationen auf einem höheren Abstraktionslevel in zwei Bereiche eingeteilt: *Datenadaptive* und *nicht datenadaptive* Repräsentationen [Lin+03]. Im Vergleich zu den nicht datenadaptiven Transformationen besitzen die datenadaptive Transformationen Parameter, die von den Eingangsdaten abhängig sind.
Zu den nicht datenadaptiven Transformationen zählen u. a. die Diskrete Fourier Transformation (DFT) und die Diskrete Wavelet Transformation (DWT). Unter Einbeziehung des Zeitbereichs kann auch die Piecewise Aggregate Approximation (PAA) den nicht datenadaptiven Transformationen zugeordnet werden. Dabei werden die Daten der Zeitreihe über Segmente konstanter Länge zusammengefasst. Die Approximation erfolgt über den Mittelwert. Die daraus resultierende Aneinanderreihung von Werten ergibt wiederum eine Zeitreihe der Mittelwerte.
Zu den datenadaptiven Repräsentationen zählen u.a. die Principal Component Analysis (PCA) und die Symbolic Aggregate approXimation (SAX). Die Hauptkomponentenanalyse (engl. *PCA*) bezeichnet die Extraktion der wesentlichen Komponenten aus höherdimensionalen Eingangsdaten. Durch Auswahl geeigneter Komponenten kann so eine Dimensionsreduzierung erzielt werden [Bac+18, S. 392; Ert16, S. 300]. Im Rahmen der SAX wird eine Dimensionsreduzierung erzielt, in dem die numerischen Werte einer Zeitreihe durch symbolische Werte diskretisiert werden [KLR04; Lin+03].

2.2.2 Dateninterpolation

Bei einer multivariaten Zeitreihe handelt es sich um eine mehrdimensionale Reihe an Informationen. Diese multivariate Zeitreihe besteht mindestens aus einer Zeitachse. Liegen dagegen mehrere Zeitachsen vor, die unterschiedliche Abtastungen beinhalten, können diese Daten im Rahmen einer multivariaten Dateninterpolation synchronisiert werden. Die Autoren Lepot, Aubin und Clemens unterteilen die deterministische Interpolation in Nearest-Neighbor Interpolation, Polynominterpolation und methoden-basierte Interpolation [LAC17].

- **Nearest-Neighbor Interpolation:** Innerhalb der vorliegenden Zeitreihe soll zu einem festgelegten Punkt ein neuer Wert eingefügt werden. Dieser neue Wert erhält im Rahmen dieser Methode den gleichen Wert wie sein nächster Nachbar.
- **Polynominterpolation:** Im Rahmen der Polynominterpolation werden bekannte Werte durch eine lineare Funktion ersetzt. Der neue Wert kann durch diese Funktion errechnet werden. Neben einer einfachen linearen Funktion, sind auch komplexere Funktionen denkbar: Zum Beispiel kubische oder spline Interpolationen.
- **Methoden-basierte Interpolation:** Bei dieser Interpolation werden zum Beispiel gewichtungsbasierte Funktionen oder Algorithmen aus der Frequenzanalyse verwendet. Für weitergehende Informationen zu methoden-basierten Interpolationen sei an dieser Stelle auf [LAC17] verwiesen.

Beim Vorhandensein von konstanten zeitlichen Abständen der einzelnen Werte wird dieses als *äquidistante* oder *synchrone* Zeitreihe bezeichnet. Auch multivariate Zeitreihen können äquidistant sein, dies wird auch als multivariate Äquidistanz bezeichnet.

2.2.3 Clustering von Zeitreihen

Die in dieser Arbeit vorliegenden Eingangsdaten sind Zeitreihen (vgl. Kapitel 2.2). In der Literatur lassen sich zahlreiche Anwendungsfälle zur Erstellung von Clustern bei Zeitreihen finden. Ziel dabei ist es immer, wertvolle Informationen aus den riesigen Datenmengen zu gewinnen. Da Zeitreihen meist in einer enormen Datengröße vorliegen, ist das Clustering dieser Datenmengen mit einem hohen Zeitaufwand verbunden. Die Autoren Aghabozorgi, Seyed Shirkhorshidi und Ying Wah beschreiben in [ASY15] die Taxonomie vom Clustering bei Zeitreihen. Sie differenzieren dabei die folgenden drei Bereiche:

- **Ganzheitliches Zeitreihen-Clustering** beschreibt dabei den Vorgang die Zeitreihe vollständig mit anderen Zeitreihen zu vergleichen. Die diskreten Objekte des Clusterings sind wiederum Zeitreihen.
- **Teilsequenz-Clustering** unterteilt eine Zeitreihe zunächst in kleinere Teilsequenzen. Diese Teilsequenzen sind Teil einer Langzeitreihe und werden miteinander verglichen.

- **Zeitpunkt-Clustering** ordnet den Feature-Vektor, der zu einem bestimmten Zeitpunkt aufgenommen wurde, einzelnen Gruppen zu. Die einzelnen Elemente einer Gruppe sind sich bezogen auf den Feature-Vektor ähnlich, wurden aber zu unabhängigen Zeitpunkten aufgenommen.

Aufgezeichnete Zeitreihen können allgemeines Rauschen, Messungenauigkeiten und Ausreißer enthalten. Außerdem können sie in unterschiedlichen Längen vorliegen. Zeitreihen ungleicher Länger verlangen einen komplexeren Prozess der Ähnlichkeitsbestimmung. Für die Bestimmung dieser Ähnlichkeit werden für metrische Variablen Distanzmaße genutzt und für nominale Variablen Ähnlichkeitsmaße.
Zur Bestimmung einer Ähnlichkeit zwischen zwei Objekten oder Klassen ist eine Berechnung der Distanz notwendig. Zur Ähnlichkeitsbestimmung von Zeitreihen aus Fahrzeugsignalen wird sich im Folgenden auf metrische Variablen beschränkt. Eine in der Literatur häufig angewandte Metrik zur Ähnlichkeitsbestimmung von zwei Zeitreihen X und Y mit n Messpunkten ist die euklidische Distanz d_e [Ert16, S. 245]:

$$d_e = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{2.15}$$

Die Generalisierung der euklidischen Distanz ist die *Minkowski* Distanz d_p. Für $p = 2$ ist es die bereits erwähnte euklidische Distanz und für $p = 1$ die Manhattan Distanz.

$$d_p = \left(\sum_{i=1}^{n} (x_i - y_i)^p \right)^{1/p} \tag{2.16}$$

Distanzen, die mit Hilfe der Minkowski Distanz gebildet werden, eignen sich nicht zur Identifikation von Ausreißern. Die berechnete Distanz kann eine verzerrte Wahrnehmung suggerieren. Außerdem ist die Berechnung empfindlich gegenüber kleinen Veränderungen auf der Zeitachse [LKB04]. Aus diesem Grund sind Erweiterung wie z.B. unterschiedliche Skalierungen, Verschiebungen und Normalisierungen einer Zeitreihe notwendig. Die *Mahalanobis* Distanz eignet sich zur Identifikation von Ausreißern besser [WK18, S. 22]. Sie bildet im zweidimensionalen Raum zu Punkten gleichen Abstands eine Ellipse, wohingegen die euklidische Distanz ein Kreis bildet.
Im Rahmen der *Mustererkennung* (engl. *pattern discovery*) werden Teilsequenzen in einer oder mehreren Zeitreihen nach bestimmten, wiederkehrenden Mustern durchsucht. Beispielsweise sei hier der Vergleich der Muster wiederkehrende Herzschläge genannt. Neben des Vergleichs von Mustern einer Zeitreihe können auch die gesamten Zeitreihen miteinander verglichen werden, wie zum Beispiel beim Verlauf von meteorologischen Wetterdaten über mehrere Jahre. Hierbei lassen sich drei Typen differenzieren: Kontur-, merkmals-, und modellbasierte Distanzmaße. Konturbasierte Distanzmaße bestimmen ähnliche Zeitreihen durch Stauchen und Strecken der gesamten Zeitreihe. So können mit Hilfe des Dynamic Time Warping (DTW) Algorithmus Zeitreihen unterschiedlicher Längen miteinander verglichen werden. Dennoch hat die Methode den Nachteil, dass auch verrauschte Datenpunkte zur Ähnlichkeitsbestimmung betrachtet werden und so das Ergebnis verfälschen können. Dagegen extrahieren merkmalsbasierte Distanzmaße Feature aus den Zeitreihen, um

mit Hilfe dieser Feature einzelne Zeitreihen miteinander zu vergleichen. Modellbasierte Distanzmaße wie z. B. AutoRegressive-Moving Average (ARMA) oder Hidden Markov Model (HMM) können auch genutzt werden, zeigen aber Skalierungsprobleme [LKB04]. [ASY15]
Die Anwendung von Clusterverfahren auf Zeitreihen findet in unterschiedlichen Bereichen statt. Im Bereich der *Anomaliedetektion* werden ungewöhnliche bzw. unerwartete Muster in einem gegebenen Datensatz untersucht. Dies zeigt auch ein Anwendungsbeispiel von Rousopoulou u. a., hier werden Akustikdaten von Industrieöfen verwendet, um Anomalien zu erkennen und vorherzusagen [Rou+19]. In dem Gebiet der bereits erwähnten Mustererkennung können wiederkehrende Muster gesucht und in Klassen eingeteilt werden. Zuletzt ermöglicht die *Prädiktion* dem Benutzer eine mögliche Einordnung des Datensatzes in vorher nicht bekannte Klassen vorherzusagen. [ASY15]

2.3 Versuchsaufbau

In diesem Abschnitt wird beschrieben, in welcher Form die Eingangsdaten zur Verfügung stehen und wie diese zur weiteren Analyse aufgenommen worden sind. Dazu wird zunächst die Ausstattung der Messfahrzeuge in Kapitel 2.3.2 vorgestellt und im Anschluss die Alterungscharakteristik der Messfahrzeuge in Kapitel 2.3.3 erläutert.

2.3.1 Einführung in die CAN-Bus-Technologie

In der Automobilindustrie werden interne Fahrzeug-Informationen, auch Zeitreihen, auf dem CAN-Bus ausgetauscht. Unterschiedliche elektronische Steuereinheiten (ECU) aktueller Fahrzeuge kommunizieren mit Hilfe des Controller Area Network (CAN)-Busses. Der CAN-Bus verbindet mit einem zweiadrigen Kabelstrang alle Steuergeräte miteinander. Eine Kommunikation und ein Datenaustausch wird so ermöglicht. Benötigt ein CAN-Bus-Teilnehmer ein Messsignal (zum Beispiel ein Temperatursignal), kann diese Information zur weiteren Verwendung mit Hilfe des CAN-Busses übermittelt werden. Eine komplexe Fahrfunktion wie etwa die Adaptive Cruise Control (ACC) benötigt zahlreiche Informationen von unterschiedlichen Steuergeräten. Im Fall einer solchen komplexen Fahrfunktion sind es unter anderem Nachrichten von Radar-, Motor-, Brems- und Getriebesteuergeräten, die auf dem CAN-Bus geschickt werden.
Der CAN-Bus verbindet mehrere gleichberechtigte Steuergeräte nach dem Multi-Master-Prinzip, das heißt die im Netzwerk verbundenen Steuergeräte sind grundsätzlich gleichberechtigt. Es gibt keinen Master, der in regelmäßigen Abständen Nachrichten von Bus-Teilnehmern einfordert oder steuert. Der CAN-Bus ist somit nicht zeitgesteuert. Die Zeitdauer zwischen Sendewunsch eines Steuergeräts und Empfang einer Nachricht ist nicht vorherzusehen. Ein Steuergerät kann eine Information erst dann übertragen, sobald der

CAN-Bus nicht mehr von anderen Teilnehmern blockiert ist. Dies führt zu einem nichtdeterministischen Verhalten des CAN-Busses. Unterschiedliche Nachrichtengruppen erhalten fest definierte Identifier, so lassen sich die Nachrichten eindeutig zuordnen und priorisieren. Liegt ein gleichzeitiger Buszugriff vor, löst das Carrier Sense Multiple Access/Collision Resolution (CSMA/CR)-Verfahren diese Kollision. Die logische *0* ist dominant und somit werden niedrigere Identifier höher priorisiert (vgl. Abbildung 2.7).
Für echtzeitfähige Bussysteme muss eine zeitliche Vorhersehbarkeit vorliegen. Dies ist beim CAN-Bus nur für die höher priorisierten Nachrichten möglich. Je mehr unterschiedliche Nachrichten und je höher die Auslastung ist, desto schwieriger ist eine zeitliche Vorhersehbarkeit [ZS14, S. 478 f.]. Ein deterministisches Verhalten wird mit Hilfe Time-Triggered-CAN (TTCAN) umgesetzt. Dabei werden Zeitfenster zur Verfügung gestellt, bei denen ein Steuergerät seine Botschaft kollisionsfrei senden kann [Bor14a, S. 123].
Ein CAN-Daten-Frame besteht aus mehreren Blöcken: Start of Frame (SOF), Arbitrierungsfeld, Kontrollfeld (CTRL), Datenfeld, Prüfsummenfeld, Bestätigungsfeld, End of Frame (EOF) und Intermission (IFS). Es wird zwischen 11-Bit-Identifier (engl. *base frame format*) und 29-Bit-Identifier (engl. *extended frame format*) unterschieden. Somit lassen sich im Datenfeld Informationen mit einer Länge von bis zu 64 Bit übertragen. Diese Feldlänge bietet genug Platz, um auch mehrere Sensorinformationen in einer CAN-Nachricht zu verschicken. Neben der Arbitrierung und der Priorisierung der Nachrichten wird die Gesamtüber-

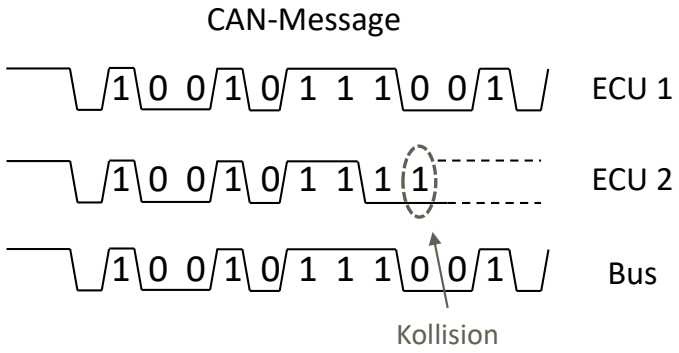


Abbildung 2.7: Arbitrierung von zwei CAN-Botschaften auf dem gleichen Bus bei unterschiedlichen Identifiern, nach [ZS14]

tragungsdauer einer Nachricht (Latenz) von weiteren Einflüssen beeinflusst. So sind Berechnungen einer Software notwendig, um eine gültige CAN-Nachricht in dem fest definierten Format vorzubereiten und auf Empfängerseite zu entpacken. Die Gesamtlatenz berechnet sich deshalb aus folgenden Verzögerungen: Softwareseitige Nachrichtenvorbereitung, Warten auf den freien Bus (Arbitrierung, Priorisierung), Übertragungsdauer und softwareseitige Nachrichtenentpackung. Als Jitter wird die zeitliche Schwankung dieser Gesamtlatenzzeit bezeichnet. [ZS14]
In der Automobilentwicklung werden diese CAN-Bus Informationen zur weiteren Analyse mit Hilfe von Datenloggern aufgezeichnet. Diese Geräte können so konfiguriert werden, dass bestimmte CAN-Bus Identifier aufgezeichnet und mit einem entsprechenden Zeitstempel versehen werden. In der Konfiguration wird auch festgelegt, mit welcher Abtastfrequenz diese Informationen aufgezeichnet werden. Das heißt, zum jeweiligen Abtastzeitpunkt wird

die zuletzt gültige CAN-Botschaft gelesen und gespeichert. Zwischen Auftreten einer Information und deren Speicherung mittels des Datenloggers kommt es zu Verzögerungen. Auf der einen Seite liegt das an den unterschiedlichen Prioritäten der Nachrichten und Auslastungen des CAN-Busses. Daraus resultiert die *CAN-Bus-Latenz*. Auf der anderen Seite wird das Prozessieren von Informationen verzögert, da der Datenlogger diese Daten empfängt und eigenständig verarbeitet (*Datenlogger-Latenz*). Die Zusammensetzung aus CAN-Bus-Latenz und Datenlogger-Latenz ergibt die *Gesamtlatenz*. Die Abbildung 2.8 stellt diese Latenzen vereinfacht dar. Für weitere Informationen zu Verzögerungen auf dem CAN-Bus und des-

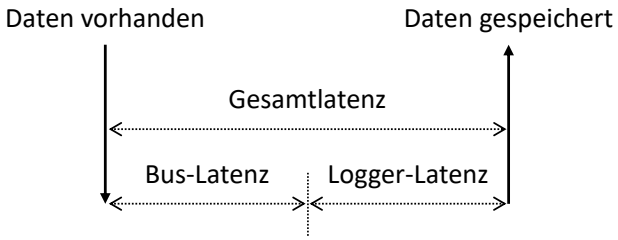


Abbildung 2.8: Vereinfachte Darstellung von Verzögerungen bei der Datenübertragung von CAN-Bus-Informationen bis zur Speicherung, in Anlehnung an [ZS14, S. 30]

sen Arbitrierung sei auf [ZS14] verwiesen. Die in dieser Arbeit vorliegenden Daten werden mit Hilfe dieser Datenlogger aufgezeichnet und anschließend in einer Umgebung außerhalb des Fahrzeugs analysiert. In dem Kapitel 2.2.2 werden weitere Verarbeitungsschritte vorgeschlagen, wie diese Informationen von unterschiedlichen Abtastfrequenzen und Latenzen synchronisiert werden können.

2.3.2 Ausstattung der Messfahrzeuge

Die in dieser Arbeit untersuchte Alterung(-sprädiktion) wird mit Hilfe der Daten von VW Nutzfahrzeugen durchgeführt. Es handelt sich dabei um Fahrzeuge des gleichen Fahrzeugtyps. Für jedes dieser untersuchten Fahrzeuge sind Messdaten von durchschnittlich über einem Jahr aufgezeichnet worden. Für ein einzelnes Fahrzeug können dabei hunderte von verschiedenen Signalen vorliegen. Darunter fallen auch mehrere Geschwindigkeitssignale, die an unterschiedlichen Positionen gemessen worden sind. Es liegen auch Informationen zu internen Zustands-, Kühler-, Druck-, Temperatur-, Spannungs- und Gasstreckenmodellsignalen vor. Des Weiteren werden auch Signale zur Fahrdynamik des Fahrzeugs aufgenommen.
Die Messdaten werden mit Hilfe von Fahrzeugdatenloggern aus Fahrzeugen gewonnen (vgl. dazu auch [ZS14, S. 460]). Mit Hilfe dieser Logger wird das Prozessieren von Informationen ohne zusätzlichen PC oder Laptop innerhalb des Fahrzeugs durchgeführt. Eine vorherige Logger-Konfiguration legt fest, welche CAN-Botschaften (s. Kapitel 2.3.1) mit welcher Auflösung aufgezeichnet werden. Diese Daten werden auf einem Medium gespeichert. Im Anschluss werden die Daten von dem Speichermedium manuell zur weiteren Analyse kopiert. Je nach Fahrbetrieb des Fahrzeugs, Dauer der Aufzeichnung und Größe des Speichermediums kann eine vollständige Aufnahme des gesamten CAN-Netzwerkes nicht gewährleistet sein.

Die vom Fahrzeugdatenlogger aufgezeichneten Informationen beinhalten neben den reinen Signalwerten auch zeitliche Informationen. Diese zeitliche Information entspricht die der vom Logger gesetzten Zeit. Zwischen dem eigentlichen Bereitstellen eines Messwertes (z.B. durch einen Sensor) und dem Speicherzeitpunkt entsteht eine Verzögerung. Neben der Latenz auf dem CAN-Bus, die durch Arbitrierung und durch mögliche Botschaftskollisionen auftreten kann, können weitere Verzögerungen innerhalb des Speichergangs des Loggers auftreten. Aus den genannten Gründen kann nicht zweifelsfrei der ursprüngliche Sendezeitpunkt bestimmt werden. Dennoch wird im Rahmen dieser Arbeit angenommen, dass es sich um Verzögerungen im unteren Millisekundenbereich handelt und auf Grund der über Wochen bis Monate stattfindenden Alterung keinen entscheidenden Einfluss haben.

2.3.3 Alterungscharakteristik

Die untersuchten Fahrzeuge weisen eine Alterung in einer bestimmten Komponente des Antriebs auf. Dazu werden die Fahrzeuge in unregelmäßigen Abständen in die Werkstatt geholt und einer Performance-Messung dieser Komponente unter fest definierten Bedingungen unterzogen. Eine solche Messung kann nur durch geschultes Personal erfolgen und erfordert einen hohen zeitlichen Aufwand.
Im Speziellen handelt es sich bei der zu analysierenden Komponente um einen Kühler innerhalb der Abgasrückführung (AGR). Durch Rückführung der Abgase (AGR) wird versucht die NOx-Emissionen zu senken. Bei Dieselfahrzeugen entzündet sich das Kraftstoff-Luft-Gemisch ohne Zündquelle selbst. Aufgrund der Luftverdichtung im Inneren des Zylinders steigt die Temperatur auf ein Wert, an dem sich der eingespritzte Dieselkraftstoff selbst entzündet. Die Emissionen eines Verbrennungsmotors sind das Ergebnis einer Wechselwirkung zwischen Schadstoffbildung und Schadstoffabbau in der Brennkammer und in der Abgasanlage. Während einer idealen Verbrennung entstehen im Wesentlichen Wasser (H_2O), Kohlenstoffdioxid (CO_2) und Schwefeldioxid (SO_2). Die Verbrennung ist jedoch aufgrund lokal schwankender Luftverhältnisse nicht ideal. Es können zusätzlich Stickoxide (NOx), Kohlenmonoxid (CO) und Kohlenwasserstoffe (HC) sowie Rußpartikel entstehen. Auch wenn der Dieselmotor einen hohen Wirkungsgrad unter den Verbrennungsmotoren hat, werden weitere Maßnahmen erforscht, um auch zukünftige Emissionsgrenzwerte der Gesetzgeber einzuhalten. Unter der AGR-Rate wird das Verhältnis vom zirkulierendem Abgasmassenstrom bezogen auf den eingebrachten Gesamtmassenstrom aus Luft und zurückgeführten Abgasen verstanden. Das AGR-Ventil kontrolliert dabei die Menge des zurückgeführten Abgases vom Motor in den Ansaugtrakt. Gekühltes Abgas kann eine weitere Reduzierung der Stickoxidemissionen ermöglichen [Lad+96; Zel+98]. Außerdem kann so die maximale Temperatur des Kraftstoff-Luft-Gemisches reduziert werden. Ein eingebauter Verteiler steuert, ob eine Kühlung des Abgases notwendig ist. So können Kohlenwasserstoff- und Kohlenstoffdioxidemissionen positiv beeinflusst werden [EKL03].
Allerdings fördert eine höhere AGR-Rate auch ein stärkeres Zusetzen (auch Versottung genannt) des AGR-Kühlers [Hoa+08; BML07]. Es handelt sich dabei um eine unerwünschte Ablagerung partikelbildender Substanzen an der Innenwand des Kühlers [Bra+15]. Diese Zusetzung des AGR-Kühlers ist ein komplexer Prozess und abhängig von verschiedenen Faktoren. Neben der Kondensation partikelbildender Substanzen kann auch der im Abgas

enthaltene Wasserdampf die Zusetzung erhöhen. Außerdem ist die Zusetzung beeinflusst vom Betriebszustand des Motors [Bra+15]. Aufgrund der Zunahme an Ablagerungen verändert sich auch die Oberfläche des Kühlers und damit verbunden ist auch eine geringere Kühlleistung. Des Weiteren wird der Querschnitt des Kühlers durch Ansammlung von Ablagerungen beeinflusst, sodass Massenströme in der Abgasrückführung nicht fehlerfrei modellierbar sind.
Diese Querschnittsflächenveränderung des Kühlers wird im weiteren Abschnitt analytisch betrachtet. Im Hochdruck-AGR-System der untersuchten Fahrzeuge ist ein AGR-Ventil eingebaut, welches wie eine Drosselklappe wirkt. Nach [NR00] stellt sich folgender Massenstrom $\dot{m}$ ein:

$$\dot{m} = \Psi_{max} \cdot \sqrt{\frac{2}{1-\Pi_{krit}} \cdot \frac{p_{pre}-p_{post}}{p_{pre}} - \frac{1}{(1-\Pi_{krit})^2} \cdot \left(\frac{p_{pre}-p_{post}}{p_{pre}}\right)^2} \cdot A \cdot p_{pre} \cdot \sqrt{\frac{2}{R \cdot T_{pre}}} \tag{2.17}$$

wobei
- Ψ_{max} : maximaler Durchflussbeiwert
- A : effektive Querschnittsfläche
- T_{pre} : Temperatur vor Drosselstelle
- p_{pre} : Druck vor Drosselstelle
- p_{post} : Druck nach Drosselstelle
- Π_{krit} : kritisches Druckverhältnis
- $\dot{m}$: Massenstrom durch Drossel
- R : spezifische Gaskonstante

Eine Umstellung nach der gesuchten Querschnittfläche A liefert [NR00]:

$$A = \frac{\dot{m}}{\Psi_{max} \cdot \sqrt{\frac{2}{1-\Pi_{krit}} \cdot \frac{p_{pre}-p_{post}}{p_{pre}} - \frac{1}{(1-\Pi_{krit})^2} \cdot \left(\frac{p_{pre}-p_{post}}{p_{pre}}\right)^2} \cdot p_{pre} \cdot \sqrt{\frac{2}{R \cdot T_{pre}}}} \tag{2.18}$$

Für den maximaler Durchflussbeiwert Ψ_{max} gilt unter Zuhilfenahme des Isentropenexponentens κ des durchströmenden Gases [NR00]:

$$\Psi_{max} = \left(\frac{2}{\kappa+1}\right)^{\frac{1}{\kappa-1}} \cdot \sqrt{\frac{\kappa}{\kappa+1}} \tag{2.19}$$

Da die Druckinformation vor und nach der Drosselstelle nicht zur Verfügung stehen, kann der analytische Weg an dieser Stelle nicht weiter betrachtet werden. Stattdessen ist ein alternativer Alterungswert des AGR-Kühlers eingeführt und in unregelmäßigen Zeitabständen messtechnisch erfasst worden. Die Massenstromrate r_m (vgl. Formel 2.20) beschreibt dabei einen Quotienten aus zwei Frischluftmassenströmen. Der erste Massenstrom wird beim vollständig geschlossenen AGR-Hochdruckventil und der zweite Massenstrom bei geöffneten AGR-Hochdruckventil gemessen[2]. Damit ein vergleichbarer Messwert entstehen kann, werden diese Messungen in den Werkstätten unter reproduzierbaren Umgebungsbedingungen

2 Der Motor wird bei einem vollständig geschlossenem AGR-Ventil ausschließlich mit Frischluft betrieben.

durchgeführt. Dazu zählt auch eine konstante Motordrehzahl und -moment. Im weiteren Verlauf wird diese gemessene Massenstromrate als Alterungswert des untersuchten AGR-Kühlers festgelegt.

$$r_m = \frac{\dot{m}_{HFM,o}}{\dot{m}_{HFM,c}} \tag{2.20}$$

wobei r_m : Massenstromrate, Wert für die Alterung des AGR-Kühlers
$\dot{m}_{HFM,o}$: Frischluftmassenstrom, AGR-Ventil geöffnet
$\dot{m}_{HFM,c}$: Frischluftmassenstrom, AGR-Ventil geschlossen

In folgender Abbildung 2.9 ist die Alterungscharakteristik eines Fahrzeuges über mehrere Monate aufgetragen. Es ist für jede Messung in der Werkstatt ein Wert zur Massenstromrate r_m vorhanden. Die Messungen fanden in unregelmäßigen Abständen statt. Auch wenn die Messung zu vorher festgelegten Bedingungen durchgeführt wurde, kann diese durch Umwelteinflüsse und Messungenauigkeiten verfälscht werden. Außerdem kann sich der gemessene Wert der Massenstromrate r_m auf Grund des zuvor gefahrenen Fahrverhaltens auch situativ leicht verbessern.

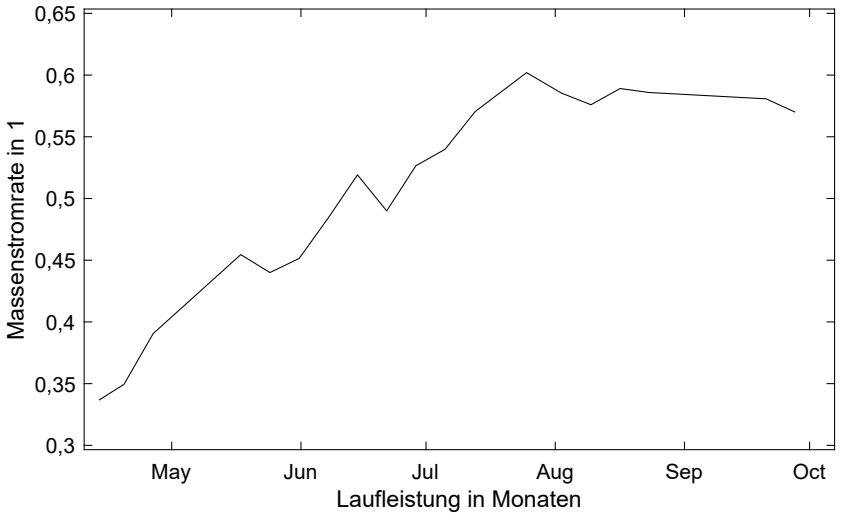


Abbildung 2.9: Beispielhafte Darstellung der Alterungscharakteristik eines Fahrzeuges hinsichtlich gemessener AGR-Kühlerperformancedaten

3 Problembeschreibung

Ein global agierender Automobilkonzern verkauft zahlreiche Fahrzeuge in unterschiedlichen Preissegmenten und Klassen. Diese Fahrzeuge werden von unterschiedlichen Kunden in unterschiedlichen Umgebungen in unterschiedlichen Fahrprofilen gefahren.
Weiterhin gewinnen vernetzte Funktionen im Automobilbereich zunehmend an Bedeutung. Neue Fahrzeuge erhalten nicht nur höhere Rechenleistungen, tauschen nicht nur immer größer werdende Datenmengen in internen Fahrzeug-Bussen aus [BS12a], sondern teilen diese Informationen vermehrt mit vernetzten Speicherarchitekturen wie z.B. in einer Rechnerwolke (engl. *Cloud*) [SG20; Sha+18]. Mit Hilfe dieser vernetzten Funktionen lassen sich Alterungsdiagnosen und -prognosen von Fahrzeugen und deren Fahrzeugkomponenten vornehmen [TMZ12; JLB06; SHM11]. Ziel eines Fahrzeugherstellers ist es, insbesondere die Fahrzeugausfälle zu vermeiden, die direkt beim Kunden auftreten.
Zur gesamtheitlichen Betrachtung der Lebenslaufkosten eines Fahrzeuges sind neben den Fixkosten, dazu zählen der Kaufpreis, die gesetzliche Abgaben und Kosten für Versicherungen, auch die variablen Kosten zu berücksichtigen [BS12b]. Die variablen Lebenslaufkosten teilen sich auf in Betriebskosten, Werkstattkosten und Entsorgungskosten. Die Zustände und Abnutzungserscheinungen einzelner Fahrzeugkomponenten werden im Rahmen einer Wartung (engl. *maintenance*) analysiert. Diese Abnutzungserscheinungen können durch Instandhaltungsmaßnahmen beseitigt werden. Dies führt zur Erhaltung eines funktionstüchtigen Betriebs. Ist eine Instandhaltungsmaßnahme aufgrund einer irreparablen Komponentenbeschädigung oder eines -defekts nicht möglich, muss ein Komponententausch durchgeführt werden. Ein ungeplanter Komponententausch führt zu vergleichsweise hohen variablen Kosten. Im Rahmen der prädiktiven Instandhaltung (engl. *predictive maintenance*) werden diese irreparablen Komponentenausfälle vorhergesagt, sodass sie vor Eintritt eines Ausfalls gewechselt werden können. Unter Einbeziehung der ganzheitlichen Kosten ist ein optimaler Zeitpunkt zum Tausch der Komponente berechenbar [Hod18, S. 138]. In der Literatur finden sich zahlreiche Untersuchungen zur Vermeidung von Komponentenausfällen unter Anwendung von Vorhersagen zu Handlungsempfehlungen [GP15; Pap+13; Cae+16]. Neben der Vorhersage von Handlungsempfehlungen wird in der Literatur auch die Vorhersage einer Restnutzungsdauer, des sogenannten Remaining Useful Life (RUL), untersucht [Mru19].
Im nächsten Kapitel 3.1 wird das Ausfallverhalten von Komponenten und die Restnutzungsdauer weiter beschrieben. Dabei wird zunächst das Ausfallverhalten in den allgemeinen Kontext einer Fahrzeugentwicklung einsortiert und im darauffolgenden Kapitel 3.2 das Problem allgemein dargestellt. Die im Zusammenhang mit der Problemstellung stehende Literatur wird im Stand der Technik in Kapitel 3.3 vorgestellt. In dem Kapitel 3.4 werden die bestehenden Lücken in der Literatur diskutiert und anschließend die daraus resultierenden Forschungsfragen definiert. In dem Abschnitt 3.5 folgt die Beschreibung der Anwendung auf ein konkretes Beispiel aus dem Automobilbereich. In den letzten beiden Abschnitten

werden die Forschungsfragen (vgl. Kapitel 3.6) und die Problemkomplexität (vgl. Kapitel 3.7) diskutiert.

3.1 Allgemeines Ausfallverhalten und Restnutzungsdauer von technischen Systemen

Im folgenden Abschnitt wird das Ausfallverhalten von Komponenten differenziert betrachtet und die Restnutzungsdauer beschrieben. Die Vorhersage eines Ausfallverhaltens ist ein Teilgebiet der Instandhaltung. Zu den weiteren Maßnahmen einer Instandhaltung gehören die Wartung, die Inspektion und die Instandsetzung [Sch10, S. 23 ff]. Neben einem konkreten Komponentenausfall ist auch die Alterung (oder Abnutzung) eine Komponente zu betrachten. Eine Komponente kann für ihren Einsatz auf Grund von Abnutzungserscheinungen oder Performanceeinbußen nicht mehr geeignet sein, obwohl diese noch nicht ausgefallen ist. Diese Restnutzungsdauer (engl. *RUL*) soll im weiteren Verlauf näher beschrieben werden.

Ausfallverhalten Es werden zwei unterschiedliche Arten eines Komponentenausfalls unterschieden. Apel differenziert in [Ape18, S. 152] zwischen einem plötzlichen Ausfall (*Sprungausfall*) und einem sich langfristig ankündigenden Ausfall (*Driftausfall*) einer Komponente (vgl. auch [SB14]). Diese Formen des Ausfalls sind in Abbildung 3.1 dargestellt. Neben

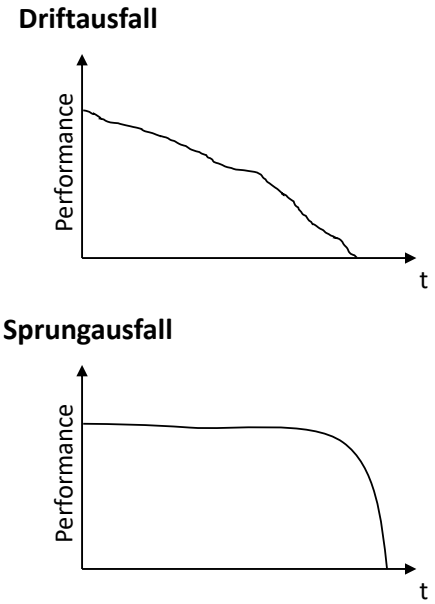


Abbildung 3.1: Schematische Darstellung unterschiedlicher Formen des zeitlichen Verlaufs von Komponentenausfällen, nach [Ape18, S. 152]

den Formen des zeitlichen Verlaufs von Komponentenausfällen wird auch die Ausfallrate

betrachtet. Die Ausfallrate λ von Komponenten wird durch Wahrscheinlichkeitsverteilungen beschrieben. Diese sind nach Sikorska, Hodkiewicz und Ma [SHM11]: Exponentielle Ausfallverteilung, Normalverteilung, logarithmische Normalverteilung, Gaußverteilung, Weibull-Verteilung und die Badewannenkurve. Die *Badewannenkurve* kombiniert drei dieser Verteilungen in einer zusammenfassenden Grafik (vgl. Abbildung 3.2) als eine Funktion der Zeit, da die Komponentenausfälle über die Lebensdauer nicht gleichmäßig verteilt sind [Bor14b]. Der erste Abschnitt der Badewannenkurve ist von frühzeitigen Ausfällen der Komponenten aufgrund von Designfehlern, fehlerhafter Produktion oder der Nutzung falscher Werkstoffe geprägt und wird auch als die sog. *Säuglingssterberate* bezeichnet. Der zweite Abschnitt beschreibt die zufälligen Fehler. Diese können im Betrieb unvorhergesehen und zu jeder Zeit auftreten. Der letzte Abschnitt beschreibt die Ausfälle auf Grund von Verschleiß- oder Ermüdungserscheinungen. Dies führt abnutzungsbedingt zu Ausfällen, zum Beispiel weil Werkstoffe ermüden, Dichtungen verspröden, oder Komponenten auf Grund von Fremdkörpereinschluss im weiteren Verlauf nicht die erforderliche initiale Leistung abrufen können. [Trz19]

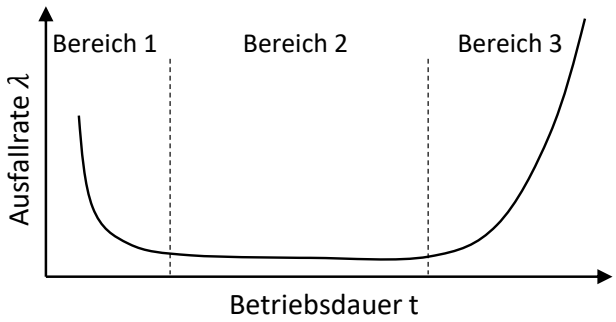


Abbildung 3.2: Ausfallverhalten von Komponenten (Badewannenkurve), nach [Trz19; Bor14b; BSS10; Hod18]

Bevor eine Ausfallwahrscheinlichkeit angegeben werden kann, sind die Alterungserscheinungen einzelner Komponenten zu analysieren. Der Autor Borgeest betrachtet in [Bor14b, S. 336 f.] die Alterung elektronischer Bauelemente und trennt dabei die Alterung von passiven Bauelementen (z.B. Schichtwiderstände), aktiven Bauelementen (z.B. Halbleiterbauelemente), elektromechanischen Bauelementen (Schalter und Taster), sowie die Alterung von Sensoren und Aktoren. Die durchschnittliche Lebensdauer der genannten Komponenten ist im Regelfall länger als die des Gesamtfahrzeugs. Dennoch kann sich die Lebensdauer bei belastenden Betriebszuständen (Betrieb bei hoher Temperatur oder zu hoher Feuchtigkeit) oder zufälligen Ausfällen verkürzen. [Bor14b]

Restnutzungsdauer Trotz gleicher Bauweise kann sich die Lebensdauer von identischen Komponenten auf Grund des kundenspezifischen Betriebs unterscheiden. Die Literatur nutzt dafür die Vorhersage der Restnutzungsdauer (RUL) einer Komponente. Die Vorhersage der

Restnutzungsdauer bezieht sich dabei auf Ermüdungserscheinungen, die dem dritten Abschnitt der Badewannenkurve zuweisbar sind. Die Abbildung 3.3 zeigt einen typischen Verlauf der Restnutzungsdauer einer Komponente über die Zeit nach Ermüdungserscheinungen. Hierbei werden die Auswirkungen einer initialen Fehlfunktion bis zum Ausfall der Komponente betrachtet [SHM11]. Ein Gesamtsystem besteht aus mehreren Teilsystemen. Die

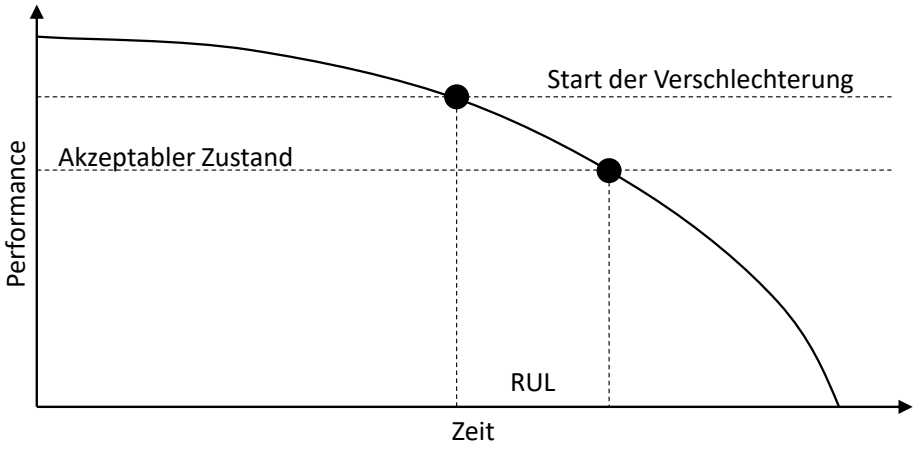


Abbildung 3.3: Restnutzungsdauer (RUL) einer Komponente über der Zeit, nach [SHM11] und [Mru19]

Abnutzung eines Teilsystems kann zu einem Ausfall des gesamten übergeordneten Systems führen. Bevor ein Ausfall des Gesamtsystems eintritt, können Funktionsstörungen auftreten. Im Fahrzeug lassen sich diese Schadensbilder nach den Autoren Schenk und Matyas in

- sicherheitsrelevante,
- umweltrelevante,
- funktionsrelevante

Störungen unterteilen [Sch10; Mat02]. Bei Betrachtung der gesamten Produktion ist diese Liste um betriebsrelevante und betriebsunabhängige Funktionsstörungen erweiterbar. Die sicherheitsrelevanten Fahrfunktionsstörungen fassen die Schadensbilder zusammen, die zu einer Beeinträchtigung der Fahrsicherheit führen können. Umweltrelevante Ereignisse zeigen sich z.B. durch Undichtigkeiten oder durch erhöhte Geräuschentwicklungen im Bereich des Fahrzeugmotors [Sch10; Mat02]. Beeinflusst das Schadensbild einer Teilkomponente weder die Sicherheit noch die Umwelt, so kann dennoch die Funktion des gesamten Systems beeinträchtigt sein. Die Alterung einer Komponente lässt sich typischerweise dem dritten Abschnitt der Badewannenkurve (vgl. Abbildung 3.2) zuordnen und betrifft damit den Bereich der nicht-zufälligen Fehler. Diese Abnutzungserscheinungen reduzieren zunächst die aktuelle Leistung der jeweiligen Komponente. Ein fehlerfreier Betrieb des gesamten Systems ist nur bis zu einem entsprechenden Abnutzungsgrad möglich [SHM11].
Die Restnutzungsdauer (engl. *RUL*) beschreibt einen Wert, wie lange noch eine Komponente ohne Funktionsstörung betrieben werden kann. Bei einer Wertüberschreitung ist ein fehlerfreier Betrieb nicht mehr gewährleistet und ein Ausfall der Komponente wahrscheinlich.

Meyer, Kimotho und Sextro zeigen am Beispiel eines Kugellagers, dass sich der Systemzustand anhand von Schwingungsdaten messen und beurteilen lässt [MKS15]. Mit Hilfe eines zugehörigen Systemmodells können so zukünftige Wartungshinweise algorithmisch entwickelt werden (engl. *condition based monitoring (CBM)*). Die Erstellung eines solchen physikalischen Modells erfordert ein umfassendes Verständnis des Systems [Mey+18, S. 199]. Lässt sich beispielsweise ein Systemmodell auf Grund einer zu hohen Komplexität (z.B. durch zu viele Einflussfaktoren) nicht erstellen, kann auch kein Wartungshinweis erfolgen. Im Gegensatz zu den physikbasierten Ansätzen benötigen datengetriebene (engl. *data-driven*) Methoden keine umfangreichen systemtechnischen Kenntnisse. Multivariate Aufzeichnungsdaten können unter Anwendung von Verfahren des maschinellen Lernens für Systemzustandsabbildungen oder Klassifikationen genutzt werden. Diese Verfahren können sowohl zur Diagnose als auch zur Prognose von Systemzuständen genutzt werden. Ziel einer Prognose ist es, eine Zustandsprädiktion auf Basis der Restnutzungsdauer durchzuführen. Handelt es sich bei diesem zu prädizierenden Wert um einen numerischen (metrischen) Wert, so sind Methoden der Regressionsanalyse anwendbar [TMZ12].
Im weiteren Verlauf werden die unterschiedlichen Algorithmen des überwachten maschinellen Lernens zur Vorhersage eines Abnutzungswertes genutzt. Das übergeordnete Ziel ist dabei eine Abschätzung der Restnutzungsdauer der untersuchten Komponente. Im nächsten Abschnitt 3.2 wird das Problem zur Vorhersage eines langfristigen Abnutzungswertes allgemein dargestellt.

3.2 Allgemeine Problemdarstellung

Der langfristige Abnutzungswert einer Komponente wird im weiteren Verlauf mit Hilfe hochaufgelöster Messsignale bestimmt, sodass ein *virtueller Sensor* (vgl. Kapitel 2.1.1) entsteht. Stehen die Messgrößen zur Bestimmung der Zielgröße in den Daten zur Verfügung, kann ein Modell zur virtuellen Erfassung des Sensorwertes erstellt werden. Die Erstellung eines Modells für den virtuellen Sensor kann auf Basis eines *physikalischen* oder *datengetriebenen Ansatzes* erfolgen (vgl. Kapitel 2.1).
Zur Erstellung eines physikalischen Modells kann das spezifische Wissen von Experten aus der Fachdomäne verwendet werden. Liegt ein umfassendes Verständnis einer Alterung vor, können die zur Verfügung stehenden Messgrößen in einen Zusammenhang gestellt werden. Die Komplexität dieses physikalischen Modells nimmt dabei zu, je mehr Messgrößen zu betrachten sind. Die Berücksichtigung aller physikalischen Einflüsse kann möglich sein, erhöht aber deutlich die Kosten und den Zeitaufwand. Dem physikbasierten Modell steht der datengetriebene Ansatz gegenüber (vgl. Kapitel 2.1.1). Hierbei können mit Hilfe multivariater Analysemethoden (vgl. Kapitel 2.1.3) auch ohne Zuhilfenahme von physikbasierten Expertenwissen Modelle erzeugt werden.
Die Eingangsdaten werden eine deutlich höhere Auflösung aufweisen als die gemessenen

Abnutzungswerte aus dem Labor. Eine erkennbare Veränderung der Messwerte einer langfristigen Abnutzungserscheinung ist erst nach Wochen beziehungsweise Monaten der Benutzung des zu analysierenden Systems sichtbar. Alterungsanstiege können je nach Betriebsverhalten in Situationen unterschiedlich auftreten. So kann die Komponente in manchen Situationen besonders schnell altern, oder in anderen Situationen in einer verminderten Ausprägung altern. Um eine gesamtheitliche Abbildung dieser Abnutzung zu gewährleisten, sollte nicht nur eine Teilmenge der Daten genutzt werden. Stattdessen sind die gesamten vorliegenden Daten für die Dauer der Abnutzung zu betrachten. Zur Modellerstellung ist der Unterschied der zeitlichen Auflösungen zwischen den Eingangsdaten und dem Abnutzungswert zu berücksichtigen.
Die Menge und Vielfalt der Eingangsdaten wirkt sich auf die Komplexität des zu erstellenden Modells aus. Die Generalisierbarkeit des Modells wird mit (unbekannten) Testdaten überprüft [Alp10, S. 39]. Dazu wird zunächst die Eingangsdatenmenge in Test- und Trainingsdaten getrennt [HKV19, S. 23; Bät17, S. 281]. Ist das erstellte Modell zu sehr an die Trainingsdaten angepasst und sind mit neuen Testdaten keine guten Ergebnisse erzielbar, liegt möglicherweise eine Überanpassung des Modells (engl. *overfitting*) vor [Bac+18, S. 94; Alp10, S. 39]. Dennoch ist eine Mindestkomplexität eines Modells erforderlich, damit relevante Muster und Regeln aus den Daten extrahiert werden können. [MG16]
Die Abbildung 3.4 visualisiert diesen Zielkonflikt [MG16, S. 29]. Es entsteht ein Sweetspot, bei dem das Modell komplex genug ist relevantes Wissen aus den Daten zu extrahieren, aber dennoch generalisierbar bleibt und auf neue Daten anwendbar ist [Alp10, S. 39]. Weiterhin

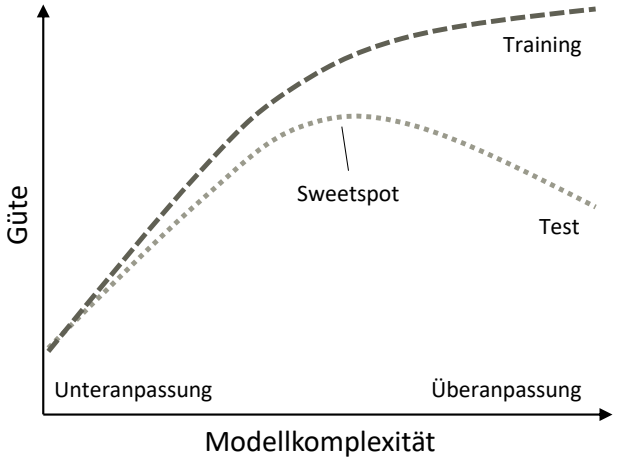


Abbildung 3.4: Zielkonflikt der Modellkomplexität unter Zuhilfenahme von Trainings- und Testdaten, nach [MG16, S. 29]

ist auch die Menge an Eingangsdaten zu berücksichtigten. Je komplexer die Eingangsdaten sind, desto größer ist die Herausforderung, relevante Muster und Regeln aus den Daten zu extrahieren. Eine Vorverarbeitung der hochdynamischen Eingangsdaten ist zur Bestimmung geeigneter Vorhersagen und Realisierung einer hohen Modellgeneralisierungsfähigkeit unvermeidlich. Mit Hilfe einer geeigneten Datenvorverarbeitung wird eine hohe Datenqualität für spätere Analysen bereitgestellt. Zu dieser Datenvorverarbeitung zählt sowohl die Bereinigung der Daten, Transformation von Daten, sowie die Datenreduktion. Die Datendiskretisierung ist eine Form der Datenreduktion. Die Methoden des Maschinellen Lernens können

mit Hilfe einer geeigneten Datenreduktion effektivere Modelle erstellen (vgl. Kapitel 2.1.1) [Liu+02; Dom12]. [TC19]
Neben der Datendiskretisierung ist auch die Wahl relevanter Merkmale nützlich und wird der Datenreduktion zugeordnet. Die Literatur zeigt, dass die Ergebnisse bei Anwendung einer Teilmenge von Merkmalen im Vergleich zur gesamten Datenmenge verbessert werden können [LLL17]. Im folgenden Abschnitt 3.3 werden weitere Arbeiten vorgestellt, die für die jeweiligen Data-Mining Aufgaben geeignete Merkmale und Signale auswählen.

3.3 Stand der Wissenschaft und Technik

Im folgenden Abschnitt wird der Stand der Wissenschaft und Technik vorgestellt. Wie bereits im Abschnitt 2.2.1 erläutert, lassen sich Zeitreihen durch Merkmale charakterisieren. Im weiteren Verlauf wird zwischen Signalen und deren Merkmalen unterschieden (weitere Definitionen dazu sind in Kapitel 2.1.1 beschrieben). Zunächst wird im Abschnitt 3.3.1 Literatur beschrieben, die unterschiedliche Merkmale aus Zeitreihen extrahiert und die Teilmenge für weitere Analysen nutzt. Im darauffolgenden Abschnitt 3.3.2 werden Arbeiten vorgestellt, die untersuchen, wie Signale aus der gesamten Datenmenge ausgewählt werden können, um mit dieser Teilmenge effizienter datengetriebene Modelle erzeugen zu können.

3.3.1 Merkmalsextraktion und -selektion

Zur Diskretisierung von Merkmalen in den hochaufgelösten Zeitreihen stehen unterschiedliche Methoden zur Auswahl. Im Rahmen der Datenanalyse wird zwischen der Extraktion und der Selektion von Merkmalen (*feature extraction* und *feature selection*) unterschieden [RB19]. Bei der Extraktion von Merkmalen werden neue, vorher nicht bekannte Merkmale aus den Daten bzw. Signalen generiert, meist durch statistische Berechnungen. Dagegen liegt die Hauptbedeutung bei der Selektion von Merkmalen nicht auf der Generierung dieser, sondern auf der Identifizierung von Merkmalen, die zur weiteren Verarbeitung betrachtet werden sollen. In diesem Fall wird davon ausgegangen, dass zahlreiche Merkmale bereits im Datensatz integriert sind oder vorher aus der Merkmalsextraktion generiert worden sind. Im Folgenden werden Studien vorgestellt, die sich in der methodischen Umsetzung auf die Selektion und Extraktion von Merkmalen fokussieren. Neben der Reduzierung der zu analysierenden Datenmengen zeigen diese Arbeiten auch, dass mit einer Teilmenge an Merkmalen bessere Ergebnisse zur Vorhersage oder Klassifizierung erzielbar sind, solange geeignete Merkmale dafür genutzt werden. Hui u. a. [Hui+17] beschreiben einen Ansatz zur Vorhersage von Maschinenfehler mit Hilfe von Vibrationssignalen. Im ersten Integrationsschritt wird das beste Merkmal genutzt. Weiterhin werden alle weitere Kombinationen getestet. Die beste Merkmalskombination aus dieser Iteration wird gewählt. Dies wird solange wiederholt, bis alle Merkmale selektiert worden sind. Zuletzt wird die Gruppe an Merkmalen ausgewählt, die die beste Klassifizierungsgüte und die geringste Anzahl an Merkmalen aufweisen kann. Der Ansatz zeigt eine Verbesserung der Klassifizierungsgüte von 74 % zu

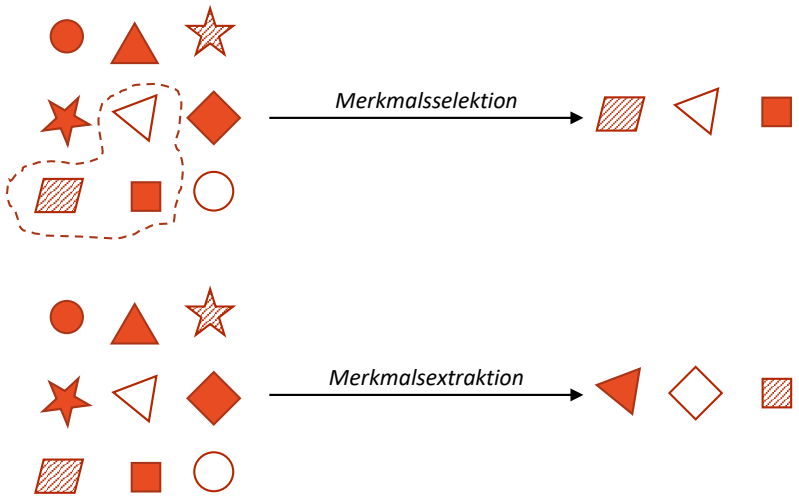


Abbildung 3.5: Schematische Darstellung der Merkmalsselektion und -extraktion, nach [RB19, S. 29]

81 % [Hui+17].
Zhang, Zhang und Xu zeigen in [ZZX16] einen Ansatz zur Auswahl von Merkmalen, um damit eine Verschlechterung von Kugellagern zu erkennen. Somit kann eine Vorhersage der verbleibenden Restnutzungsdauer (RUL) bestimmt werden. Es werden verschiedene statistische Merkmale im Zeit-, Frequenz und Zeit-Frequenz-Raum aus den Signalen generiert. Diese Merkmale sollen im Verlauf über die Lebenszeit mit der abfallenden Leistung korrelieren. Es werden folgende Gütekriterien betrachtet: Korrelation, Monotonie und Robustheit. Die Merkmale mit der höchsten Linearkombinationen von diesen Gütekriterien werden zur weiteren Betrachtung gewählt. Die Evaluation zeigt, dass die vom Algorithmus ausgewählten Signale einen mit der Zeit steigenden Verlauf aufweisen. Nicht relevante Merkmale zeigen ein solches Verhalten nicht.
Guo u. a. stellen in [Guo+00] ein System zum Erkennen von Problemen im Kontext der Fahrzeugdiagnose vor. In dieser Studie werden Segmente (Zeitreihen bestimmter Längen) bezüglich eines normalen oder abnormalen Verhaltens klassifiziert. Die Ergebnisse zeigen, dass zu 95-100 % eine korrekte Klassifizierung für abnormale Zustände erfolgt. Allerdings kann es auch vorkommen, dass normale Zustände als abnormale erkannt werden. Dies liegt zu einem großen Anteil an dem nicht komplett repräsentativen Trainingsdatensatz [Guo+00]. In der Studie [Cro+03] von Crossman u. a. wird die effektive Bildung relevanter Merkmale von Signalen und die entsprechende Fahrzeugfehlerzuordnung untersucht. Für die Durchführung von Fahrzeugdiagnosen im Motorsteuergerät ist ein Verständnis vom Signalverhalten und des zugehörigen Fehlers notwendig. Zunächst werden Sequenzen der Signale in Abschnitte auf Basis der Fahrdynamik einsortiert. Diese Bereiche können einem Beschleunigung-, Segel-, Bremsvorgang oder ähnlichen Situationen zugeordnet werden. Mit Hilfe der linearen Separierbarkeit wird überprüft, wie gut sich die einzelnen Klassen mit der selektierten Teilmenge an Merkmalen trennen lassen. Einzelne Segmente werden in normale und abnormale gegliedert, dies geschieht durch manuelle Prüfung von Experten (ähnlich zu [Guo+00]). Der Algorithmus liefert für diese Klassifikation unter Verwendung aller 48 Merkmale eine Klassifizierungsgüte (*accuracy*) von 61,9 %, bei Verwendung der automatisch ausgewählten

Merkmale sogar 83,9 %.
Weiterhin zeigen folgende Arbeiten, dass mit Hilfe von extrahierten Merkmalen aus einem einzigen zeitbasierten Signal (Drehmoment) prädiktive Instandhaltung möglich ist [Car+16; Car+18]. Auch hier wird das zeitbasierte Eingangssignal in vier wiederkehrende Segmente jeweils mit einer Länge von 1s eingeteilt, aus diesem wiederum fünf definierte Merkmale aus dem Zeitbereich extrahiert werden. Nach der Berechnung der Merkmale (*feature calculation*) erfolgt die Reduktion dieser (*feature reduction*). Bekannte Fehler werden mit Hilfe eines neuronalen Netzwerks klassifiziert und an ein Überwachungssystem weitergegeben, unbekannte Fehler werden durch eine Neuigkeitsdetektion (*novelty detection*) an das System weitergegeben. Die Ergebnisse zeigen, dass die sieben bekannten Fehlerszenarien von dem System erkannt werden. Mit Hilfe dieser Merkmalsreduktion konnten die Ergebnisse, sowohl für die Fehlerklassifizierung als auch für die Genauigkeit der Neuigkeitsdetektion, verbessert werden [Car+16; Car+18].
Liu u. a. stellen ein Methode zur Luftqualitätsvorhersage in Peking und weiteren umliegenden Städten mit Hilfe von täglichen Merkmalen in [Liu+17] vor. Als Datenquelle dienen sowohl Verschmutzungsdaten als auch Wetterinformationen. Als Methode verwenden sie dazu eine Support Vector Regression. Mit Hilfe einer Kreuzvalidierung werden die Ergebnisse evaluiert. Die Ergebnisse zeigen, dass die Vorhersage der Luftverschmutzung einen durchschnittlichen Fehler von 5 bis 9% aufweist.
In der Arbeit von Gardner u. a. werden Informationen aus Wartungstabellen analysiert und zukünftige Wartungsereignisse vorhergesagt [Gar+17]. Dabei wurden unterschiedliche Informationen zu einem Wartungsereignis aufgezeichnet. Insgesamt standen 25.000 Wartungsereignisse für die Analyse zur Verfügung, die im weiteren Verlauf noch eingekürzt wurden. Bei der Aneinanderreihung dieser Aufzeichnungen entstand eine Zeitreihe. Die Autoren haben in ihrer Arbeit nun ähnliche Fahrzeugtypen gruppiert und gezeigt, dass sie die Art des Wartungsereignisses diesen Fahrzeugen zuordnen und vorhersagen können.
Die Arbeit von Shafi u. a. stellt fest, wie Diagnosedaten von der OBD-Schnittstelle zur Fehleridentifikation in den Teilsystemen des Fahrzeugs (Kraftstoffsystem, Zündanlage, Abgassystem und Kühlsystem) anwendbar sind [Sha+18]. Relevante Muster werden durch folgende Klassifikatoren bestimmt: Entscheidungsbäume, SVM, kNN und Random Forest (RF). Die analysierten Daten beinhalten sogenannte diagnostische Fehlercodes von unterschiedlichen Sensoren, die im Anschluss auf einen Server übertragen wurden. Diese Daten wurden von 70 Fahrzeugen mit Hilfe von OBD-Scannern über Bluetooth aufgezeichnet. Im Anschluss wurden die Daten auf die wesentlichen Informationen über eine PCA reduziert. Nach dieser Datenvorverarbeitung und einer Klassifizierung konnten so relevante Muster in der Eingangsdatenmenge bestimmt werden. Die Arbeit zeigt, wie die Diagnosedaten anwendbar und in eine Fahrzeug/Cloud-Umgebung integrierbar sind, um entsprechende Defekte an Fahrzeug-Teilsystemen vorhersagen zu können.
In der Arbeit [Zhe+18] von Zheng u. a. stehen Daten der NASA von verschiedenen Flugzeugmotoren zur Verfügung. Diese Maschinen werden in mehreren Betriebszyklen betrieben. Mit Hilfe datengetriebener Methoden wird eine Vorhersage der Restnutzungsdauer durchgeführt. Zunächst sind die 21 vorliegenden Sensoren mit Hilfe statistischer Verfahren auf 14 Sensoren reduziert worden. Von den Autoren wird eine variable Fenstergröße zur

Aggregation der Daten vorgestellt. Die Merkmale aus den einzelnen Fenstern sind zur weiteren Verarbeitung zusammengefasst worden. Der Fehler der Vorhersage wird mittels des RMSEs bestimmt und ist durch verschiedene Lernalgorithmen evaluiert worden.

3.3.2 Signalauswahl

Neben der Merkmalsextraktion und -selektion muss auch die Selektion an relevanten Signalen betrachtet werden. Die Auswahl eines Signals steht in einer engen Beziehung zur Auswahl eines Merkmals, mit dem Unterschied, dass bei der Auswahl eines Signals die komplette Zeitreihe ausgewählt wird, wohingegen Merkmale einer Zeitreihe in Teilsequenzen gebildet werden. Im Folgenden werden Arbeiten von unterschiedlichen Autoren vorgestellt, die eine solche Auswahl an zeitbasierten Signalen vornehmen.
Prytz, Nowaczyk und Byttner implementieren in [PNB11] einen Ansatz zur Erkennung von Abhängigkeiten von fahrzeug-internen Signalen. Zu Beginn wird eine Signalbereinigung von externen Einflüssen durchgeführt. Dabei werden die Restfaktoren betrachtet, die nicht durch Linearkombination der externen Einflüsse beschrieben werden können. Interne Fahrzeugsignale sind abhängig, wenn sie sich besonders gut durch andere interne Fahrzeugsignale modellieren lassen. Als Gütemaß für dieses Modell wird der mittlere quadratische Fehler (MSE) verwendet. In diese Datenmenge werden bestimmte, fest definierte Fehlerbilder injiziert. So können mit Hilfe von linearer Regression, Support Vektor Maschine und Random Forest die Fehlerklassen bestimmt werden. Die Evaluierung zeigt, dass für diesen Anwendungsfall der Random Forest den geringsten Fehler aufweist.
Mrowca, Moser und Gunnemann zeigen in [MMG18] einen Ansatz zur Gruppierung von internen Fahrzeugsignalen, um so redundante Signale zu erkennen und damit die Buslast zu verringern. Ein weiteres Ziel ist die automatische Erkennung von funktionalen Zusammenhängen von Signalen für weitere Analysen. Die vorliegenden Signale werden zunächst bezüglich ihrer Art (numerische, kategorisch, etc.) sortiert. Aus diesen Signalen werden unterschiedliche Merkmale verwendet. Bei Verwendung von kategorischen Daten sind diese Merkmale beispielsweise durch die Anzahl an Änderungen, die Häufigkeiten bestimmter Werte oder Änderungen pro Zeitabschnitt beschreibbar. Für numerische Werte werden in der Arbeit eine Reihe von statistischen Merkmalen extrahiert, u.a. der Mittelwert, Varianz, Schiefe, etc. Ein durch Experten definierter Datensatz wurde genutzt, um unterschiedliche geeignete Teilmengen von Merkmalen zu identifizieren (Relevanzerkennung eines Merkmals). Mit Hilfe dieses Feature-Vektors wird über die euklidische Distanz die Ähnlichkeit bestimmt. Die Präzision gibt unter Einbeziehung der relevanten Merkmale die Güte der Clusterzuordnung an. Unter Verwendung relevanter Merkmale konnte sogar eine Verbesserung des Ergebnisses um bis zu 20 % erzielt werden.
Darüber hinaus werden in dieser Arbeit die folgenden Clusterverfahren miteinander verglichen: DBSCAN, Agglomerative, SOM, WaveCluster und der k-Means Algorithmus. Es liegt eine expertengenerierte Zuordnung der Gruppen vor. Die Autoren dieser Studie zeigen, dass der DBSCAN und der Wave-Cluster Algorithmus die besten Signalgruppierungen zu den gegebenen Eingangsdaten liefern. Zur Berechnung dieser Güte wurde der Silhouetten Index verwendet.

In einer weiteren Arbeit werden Gruppen bestimmter Fahrprofile anhand von Fahrsequenzen gebildet, statt Signalgruppen mit Hilfe von Clusterverfahren zu identifizieren [Fug+19]. Aus einer Vielzahl von CAN-Bus Signalen werden dazu die relevantesten acht zur Verhaltenserkennung ausgewählt. Dazu zählen u.a.: Geschwindigkeit, der Bremspedaldruck und die Beschleunigung. Aus diesen Signalen werden festgelegte Merkmale extrahiert, wie Mittelwert, Median, Standardabweichung, Maximum und die Steigung aus den Werten der gegebenen Sequenz des Signals. Diese werden innerhalb der gegebenen Sequenz über eine Minute gemittelt und zur weiteren Berechnung verwendet. Zur Klassifizierung des Fahrverhaltens wird der k-Means Algorithmus benutzt.
In der Arbeit von Calabrese, Campanella und Proverbio wird Stahlkorrosion über einen langen Zeitraum untersucht [CCP12]. Ziel dabei ist einer Verbesserung der Messaufzeichnung mit Hilfe einer Reduktion von Umgebungsrauschen. Die Eingangsdaten werden zunächst mit einem Hochpass-Filter geglättet, im Anschluss wurde das Rauschen durch eine Kombination von PCA und des k-Means Algorithmus entfernt. Zur Identifizierung der Güte werden folgende Güteindizes betrachtet: Silhouetten Index, der Dunn Index, der Davies-Bouldin Index und der Calinski-Harabasz Index. Abschließend werden die akustischen Signale mit Hilfe des Self Organizing Map (SOM) Algorithmus von den unerwünschten Geräuschsignalen separiert.
Xu, Wang und Xu wählen in ihrer Arbeit aus einer Menge an Sensoren diejenigen aus, die sich hinsichtlich von Kosten und Ergebnisqualität als besonders geeignet erweisen [XWX15]. Dazu stehen 13 unterschiedliche Sensoren zur Verfügung. Die Ergebnisse werden mit Hilfe von Daten eines Flugzeuggasturbinentriebwerks evaluiert. Neben charakteristischen Fehlerraten und Anschaffungskosten der Sensoren werden in deren Modell auch 9 verschiedene Ausfallereignisse in die Berechnung mit einbezogen. Die Ausfallereignisse sind dabei mit bestimmten Wahrscheinlichkeiten versehen worden.

3.4 Identifikation bestehender Lücken in der Literatur

In diesem Abschnitt werden bestehende Lücken in der Literatur identifiziert und darauf aufbauend Forschungsfragen festgelegt. Im vorherigen Abschnitt 3.3.1 und 3.3.2 wurden diverse Arbeiten vorgestellt, die Merkmale- und Signalselektionen vornehmen, um damit die datengetriebenen Analysen zu verbessern. In der Abbildung 3.6 wird diese Literatur hinsichtlich zweier Ausprägungen genauer charakterisiert. Die erste Ausprägung beschreibt die benutzte Eingangsdatenmenge und die zweite Ausprägung stellt den betrachteten Vorhersagehorizont der vorgestellten Arbeiten dar.

Datenmenge Diese Ausprägung beschreibt die Menge an Eingangsdaten, die zur Erstellung eines Modells benutzt werden. Im Rahmen der Zeitreihenanalyse sind hiermit die unterschiedlichen (physikalischen) Signale gemeint, die zum Beispiel durch unterschiedliche Sensoren aufgenommen werden.

Vorhersagehorizont Diese Ausprägung beschreibt den betrachteten Vorhersagehorizont in der jeweiligen Literatur. Ziel ist es, mit den verwendeten Eingangsdaten einen bestimmten Wert oder Zustand vorherzusagen. Diese Prognose hat dabei einen bestimmten zeitlichen Bezug, so kann ein kurzfristiges Ereignis in der Zukunft gemeint sein, oder eine langfristige Änderung eines Zustandes.

3.4.1 Vorstellung der Forschungslücke

Die in Kapitel 3.3 vorgestellte Literatur zur virtuellen Sensormodellentwicklung von Zustandsänderungen wird in der Abbildung 3.6 in einer Übersicht zusammenfassend dargestellt. Neben der Einsortierung der Literatur wird auch die Forschungslücke grob skizziert. Die vorgestellten Arbeiten zeigen unterschiedliche Ausprägungen hinsichtlich der Daten-

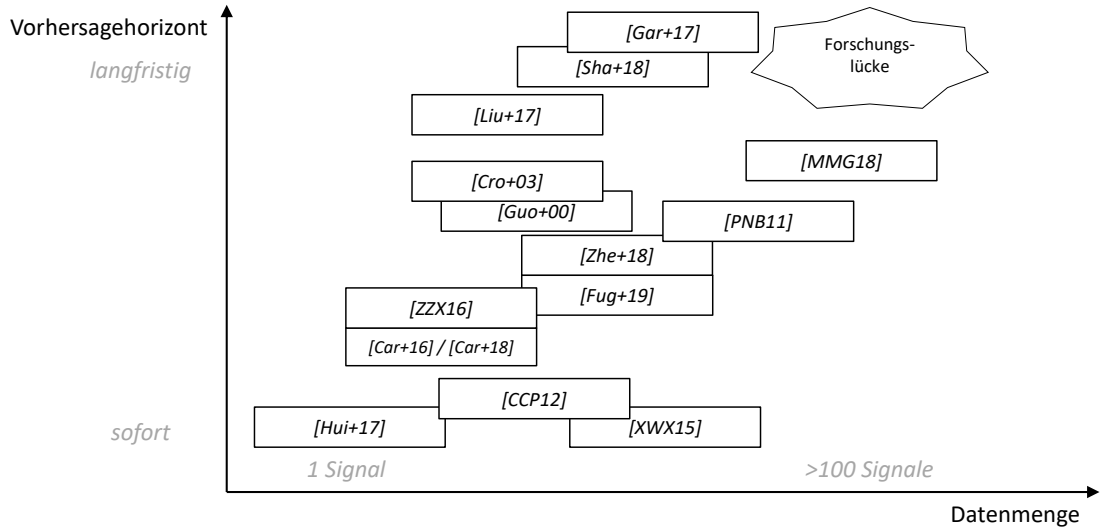


Abbildung 3.6: Darstellung der untersuchten Literatur zur Sensormodellentwicklung von Zustandsänderungen unter Einbezug der verwendeten Datenmenge und des Vorhersagehorizonts

reduktion und der Übertragbarkeit für andere Anwendungsfälle. Die Forschungslücke ist im Bereich der größeren Datenmenge und des längerfristigen Vorhersagehorizonts besonders ausgeprägt. Die virtuelle Sensormodellentwicklung zur Vorhersage einer langfristigen Zustandsänderung unter Zuhilfenahme von vielen Signalen befände sich in der Abbildung 3.6 am oberen rechten Rand. Es wird ein Konzept gesucht, mit dessen Hilfe ein langfristiger Abnutzungswert einer Komponente bestimmbar wäre, obwohl die multivariate Signalmenge im Vorfeld nicht zweifelsfrei eingegrenzt werden kann. Die Kombination von featurebasierten und signalbasierten Datenreduktionen (vgl. Kapitel 3.3.1 und 3.3.2) führt zu einer größeren Datenreduktion und wird nachfolgend weiter untersucht. Auch trotz dieser Datenreduktion sollte der Ansatz nicht nur für einen speziellen Anwendungsfall anwendbar sein, sondern zeitgleich auch für andere Beispiele übertragbar bleiben.
Neben der verwendeten Datenmenge und dem durch die Methode prognostizierten Vorhersagehorizont wird auch die Vorhersageeigenschaft betrachtet werden. Eine Einsortierung

der Daten und bestimmte Klassen, zum Beispiel „gesunde“ und „defekte“ Zustände kann im Rahmen des maschinellen Lernens durch eine Klassifikation erfolgen (vgl. Kapitel 2.1.3). Soll dagegen ein schleichender (alternder) Prozess beschrieben werden, eignen sich Regressionsmethoden des überwachten Lernens. Ein Großteil der vorgestellten Arbeiten in Abbildung 3.6 betrachten dabei Klassifikationen, so auch: [Hui+17; Guo+00; Cro+03; Car+16; Car+18; PNB11; MMG18; CCP12; Fug+19].

3.4.2 Zusammenfassung der Problemstellung

Ziel ist die Erstellung eines virtuellen Sensormodells zur Abbildung eines Alterungsverhaltens einer Komponente. Für die Erstellung eines physikalischen Modells entstünden hohe Kosten, da ein umfassendes Systemverständnis der zur untersuchenden Komponente benötigt werden würde. Außerdem bestünde die Gefahr, dass nicht alle Zusammenhänge identifiziert und dadurch auch nicht berücksichtigt werden könnten. Stattdessen soll ein datengetriebenes Modell zur Extraktion von Mustern und Regeln in hochaufgelösten Eingangsdaten mit Hilfe von Methoden des MLs erzeugt werden (vgl. Kapitel 2.1).
Da die in Kapitel 3.1 beschriebene allgemeine Restnutzungsdauer einer Komponente über einen langen Zeitraum zu beobachten ist und die Alterungserscheinung nicht (oder nicht vollständig) durch spezifisches Fachwissen erklärt werden kann, werden alle zur Verfügung stehenden Daten betrachtet. Aus diesem Grund werden die gesamtzeitlichen Zeitreihen der Messsignale analysiert. Damit relevante Muster und Regeln aus den Daten extrahiert werden können, wird die Menge und Variabilität der Eingangsdaten vor der Modellerstellung reduziert (vgl. Abbildung 3.4). Die Vorstellung der Literatur in Kapitel 3.3 hat gezeigt, dass es zahlreiche Ansätze zur Datenreduktion gibt. Diese Datenreduktion wurde in Merkmals- und Signalselektion aufgeteilt (vgl. Kapitel 3.3.1 und 3.3.2). Es ergibt sich ein Zielkonflikt für die Wahl der Diskretisierungsstufe der Merkmalsbildung. Auf der einen Seite soll eine präzise und zeitnahe Alterungsvorhersage ermöglicht werden, auf der anderen Seite muss eine Veränderung im Abnutzungswert in den aggregierten Daten wahrnehmbar sein. Es ergibt sich der Bedarf nach einer bewertenden Aussage, wie gut sich diese aus den hochdynamischen Eingangsdaten aggregierten Informationen zur Vorhersage von langfristigen Abnutzungserscheinungen nutzen lassen.
Ein weiteres Problem ist die *Übertragbarkeit* des Ansatzes auf andere Abnutzungserscheinungen. Es dürfen zunächst keine Signale von der datengetriebenen Analyse ausgeschlossen werden. Werden dagegen im weiteren Verlauf mit Hilfe eines Algorithmus relevante Signale in den Eingangsdaten identifiziert, können diese gesondert für eine mögliche Alterung betrachtet werden. Eine Übertragbarkeit auf andere Anwendungsfälle wäre damit gewährleistet.
Neben der reinen Datenaggregation ist auch die Vorhersagequalität im Sinne der Prädiktion zu untersuchen. Die Literatur schlägt dazu zahlreiche Werkzeuge vor (vgl. Kapitel 2.1.4). Dennoch besteht der Bedarf, diese Vorhersagequalität im Rahmen einer Komponentenabnutzung zu bewerten. Neben einer reinen Qualitätsbewertung ist auch die Fragestellung der *Erklärbarkeit* von Interesse. Ist die zu untersuchende Abnutzungserscheinung noch nicht vollständig charakterisiert, können datengetriebene Analysen und Auswertungen diesbezüglich hilfreiche Informationen für zukünftige Projekte liefern. Mit Hilfe solcher Analysen können

Zusammenhänge zwischen den Eingangsdaten und der Abnutzungserscheinung festgestellt werden. An dieser Stelle ist denkbar, dass sich bestimmte fahrzeuginterne Signale oder ausgewählte Methoden der Vorverarbeitung besonders gut für eine Alterungsschätzung eignen könnten. In dem Kapitel 3.6 werden aus den in diesem Kapitel skizzierten Problemen und den vorgestellten Lücken im Stand der Technik die Forschungsfragen abgeleitet.

3.5 Anwendung der Problematik auf eine Antriebskomponente im Fahrzeug

Das dargestellte abstrakte Problem soll im weiteren Verlauf auf eine bestimmte Komponente im Antrieb des Fahrzeuges angewendet werden. Dazu stehen neben den hochaufgelösten CAN-Bus Aufzeichnungen auch Alterungsinformationen einer Antriebskomponente von mehreren Fahrzeugen zur Verfügung (vgl. Kapitel 2.3.2). Diese Abnutzungserscheinung wird unregelmäßig mit hohem zeitlichen Aufwand in einer Werkstatt und fest definierten Bedingungen gemessen und in Kapitel 2.3.3 näher beschrieben. Dennoch unterliegen diese Messungen messtechnischen Qualitätsschwankungen. Ein datengetriebenes Modell soll diese aufwändige Messung ersetzen, um so diesen Alterungswert als virtuellen Sensor auf Basis der aufgezeichneten CAN-Bus Signale zu bestimmen. Der Alterungswert des AGR-Kühlers wird als Massenstromrate r_m definiert und wurde bereits in Formel 2.20 vorgestellt. Die in Werkstätten regelmäßig gemessene Massenstromrate r_m kann von Umwelteinflüssen oder Messungenauigkeiten verfälscht werden. Die Abnutzungserscheinung der Komponente kann sich auf Grund der Betriebsdauer, des individuellen Fahrstils oder der vorliegenden äußeren Witterungsbedingungen unterschiedlich verändern. Neben den Einflüssen des Fahrers und dessen Fahrprofil auf die Alterung, kann diese auch durch Softwareupdates oder nicht dokumentierten Hardwarewechsel geprägt sein. Außerdem werden die Massenstromraten auch in verschiedenen Werkstätten vermessen, was zu einer erneuten Messverfälschung führen kann. Weiterhin sind die Zeitinformationen der Messsignale kleinen Ungenauigkeiten bezüglich der Äquidistanz auf Grund unterschiedlicher Konfigurationen der Datenaufzeichnungsgeräte und des nichtdeterministischen Verhaltens des CAN-Busses (vgl. Kapitel 2.3.1) ausgesetzt.
Der wahre Alterungswert (gemessen in Werkstätten) wurde nur in unregelmäßigen Abständen messtechnisch erfasst. Die gespeicherten CAN-Daten liegen dagegen in einer hohen Aufzeichnungsfrequenz vor. In dieser Arbeit werden diese Unterschiede der Datenauflösung betrachtet und ein Konzept für die Datenaggregation vorgeschlagen. Im weiteren Verlauf wird das zu erstellende virtuelle Sensormodell zur Alterungsprädiktion hinsichtlich der Vorhersagequalität untersucht.

3.6 Forschungsfragen

Aus denen in diesem Kapitel identifizierten Defiziten werden die ersten Forschungsfragen (**RQ 1** und **RQ 2**) abgeleitet. Mit Hilfe des dargestellten speziellen Anwendungsfalls soll die in Kapitel 3.4.2 vorgestellte Problemzusammenfassung validiert werden und wird mit

RQ 3 adressiert.

RQ 1: Wie kann ein virtueller Sensor unter Berücksichtigung unterschiedlicher zeitlicher Auflösungen zwischen hochdynamischen Eingangsdaten und einer Zielgröße datenbasiert übertragbar modelliert werden?

RQ 2: Wie kann dieses datenbasierte Sensormodell hinsichtlich Genauigkeit und der genutzten Datenmenge optimiert werden?

RQ 3: Wie kann die Qualität dieses virtuellen Sensors anhand der AGR-Kühler-Alterung validiert werden?

3.7 Problemkomplexität

Das dargestellte Problem wird im weiteren Abschnitt hinsichtlich der Komplexität beschrieben. Zunächst wird in Kapitel 3.7.1 an die Problemkomplexität herangeführt und das in dieser Arbeit vorliegende Problem aus einer abstrakten Sicht betrachtet. Im weiteren Abschnitt (Kapitel 3.7.2) wird die Problemkomplexität in die Literatur einsortiert und weiter diskutiert. Zuletzt werden unterschiedliche Lösungsstrategien in Kapitel 3.7.3 vorgestellt.

3.7.1 Heranführung an die Problemkomplexität

Zur Prädiktion des Alterungswertes stehen zahlreiche Eingangsdaten X zur Verfügung. Eine unbekannte Funktion $f(X)$ ist in der Lage, diese Daten für die Alterung bestmöglich abzubilden. Die in dieser Arbeit analysierte Alterung ist modellseitig noch nicht vollständig erfasst. Relevante Signale zur Bestimmung der Alterung und das zugehörige Alterungsmodell dafür liegen nicht vor. Aus diesem Grund wird diese komplexe Funktion $f(X)$ mit einer einfachen Funktion durch datengetriebene Methoden des maschinellen Lernens angenähert, um so ein Ergebnis für die Alterung zu bestimmen.
Im weiteren Abschnitt werden die Rahmen der Problemkomplexität zu betrachtenden Anforderungen und Randbedingungen zur datengetriebenen Bestimmung dieser Alterung beschrieben. Dazu zählen die Datenauflösung und der Vorhersagehorizont, die Güte und Übertragbarkeit des Modells, sowie die Realisierbarkeit in einer Fahrzeug/Cloud-Umgebung.

Datenauflösung und Vorhersagehorizont Es steht eine große Menge an Eingangsdaten X zur Verfügung. Diese ist durch eine hohe Änderungsrate (die Auflösung der Werte ist hoch) und durch die Anzahl an aufgenommenen Signalen gekennzeichnet. Die Zielgröße verändert sich dagegen weniger dynamisch und ist nicht annähernd so hoch aufgelöst. Es handelt sich deshalb um einen langfristigen Vorhersagehorizont.

Güte und Übertragbarkeit des Modells Es wird ein virtuelles Sensormodell zur Alterungsprädiktion erstellt und mit gegebenen Daten angelernt (Trainingsdatensatz). Das Sensormodell ist sowohl von der benutzten Datenmenge als auch von dem genutzten Algorithmus zur Modellerstellung abhängig. Neben dem Trainingsdatensatz gibt es auch einen Testdatensatz, der nicht zum Lernen benutzt wird. Mit Hilfe dieses Testdatensatzes kann die Güte bestimmt werden, wie gut das Modell eine Vorhersage für eine unbekannte Datenmenge treffen kann. Für eine hohe Generalisierbarkeit sollte eine Überanpassung (engl. *overfitting*) vermieden werden. Sie entsteht bei hohen Modellsensitivitäten, die zu Anfälligkeiten gegenüber Rauschen führen können. Außerdem wird im zu erstellenden Konzept eines virtuellen Sensormodells eine theoretische Übertragbarkeit auf andere Abnutzungserscheinungen vorgesehen.

Realisierbarkeit in einer Fahrzeug/Cloud-Umgebung Auch wenn es nicht Fokus dieser Arbeit ist, sind auch Speicher- und Rechenbedarf zur Erstellung eines Modells relevante Einflussfaktoren im Rahmen der Fahrzeugentwicklung in der Automobilindustrie. Aus diesem Grund ist es wichtig, die für die Erstellung eines Modells genutzten Ressourcen zu betrachten.

3.7.2 Einsortierung

Im Rahmen der allgemeinen Komplexitätstheorie werden Probleme der Informatik in unterschiedliche Klassen eingeteilt. Probleme der P-Klasse wachsen im Rahmen ihrer Komplexität maximal polynomial. Dem gegenüber stehen die Probleme der NP-Klasse. Probleme der NP-Klasse lassen sich nur mit nichtdeterministischen Turingmaschinen mit polynomialen Zeitaufwand lösen. In der Informatik existieren bereits Beispielprobleme, die der Klasse der NP-vollständigen Probleme zugeordnet werden können. Lässt sich ein unbekanntes Problem auf ein Problem der Menge der NP-vollständigen Probleme reduzieren, so wird auch von der NP-Vollständigkeit des unbekannten Problems ausgegangen, solange $P \neq NP$ gilt [Sip13, S. 299ff].
Das in dieser Arbeit vorliegende Problem ist dadurch charakterisiert, dass es einzustellende Parameter der Vorverarbeitung und des Lernprozesses gibt. Zur Angleichung der unterschiedlichen Datenauflösungen und zur effizienteren Modellerstellung werden in den vorliegenden Datensätzen unterschiedliche Merkmale bestimmt (vgl. dazu auch die bereits vorgestellte Literatur in Kapitel 3.3.1 zum Thema Merkmalsextraktion und -selektion). Die geeignete Wahl der Merkmale bedingt sich wechselseitig mit der zu prädizierenden Alterung. Aus den vorliegenden Eingangsdaten werden im Rahmen der Vorverarbeitung die zu verwendenden Signale, die Wahl der Merkmalsoperatoren und die Diskretisierungsstufe festgelegt. Es wird die Modellgüte J gesucht, die unter Anwendung der ausgewählten Parameter der Vorverarbeitung und des Lernprozesses die langfristige Alterung bestmöglich prädiziert.
An dieser Stelle ist allein für die Einstellparameter der Vorverarbeitung eine unendliche Kombinationsmöglichkeit aus Signalen, Merkmalsoperatoren und Diskretisierungsstufen

möglich. Auch wenn für bestimmte Kombinationsmöglichkeiten eine sehr gute Modellgüte J erzielt werden kann, kann nicht gewährleistet werden, dass nicht eine bessere Lösung durch Veränderung einer der Hyperparameter gefunden werden kann. Es besteht eine wechselseitige Abhängigkeit zwischen den am Modelleingang eingespeisten Daten und der am Ausgang resultierenden Modellgüte.
In der Literatur ist das in dieser Arbeit vorliegende Problem im Rahmen der Hyperparameteroptimierung (engl. *hyperparameter optimization*) bekannt und wird als NP-vollständig beschrieben (vgl. [And19], [HKV19, S. 3]). Hyperparameter beschreiben einzustellende Modellparameter, die im Rahmen der Durchführung veränderbar sind. Der Parameterraum kann sehr komplex werden, da die einzustellenden Parameter sowohl kontinuierliche, kategoriale als auch konditionale Werte annehmen können (vgl. [HKV19, S. 4]). Der Prozess der Hyperparameteroptimierung ist durch folgende Formel 3.1 beschreibbar [And19]:

$$\lambda^* = \arg\min_{\lambda \in \Lambda} J(\lambda) \tag{3.1}$$

Dabei gilt, dass $J(\lambda)$ die zu optimierende Funktion ist, in diesem Fall die Güte des Modells. λ^* ist dabei die Konfiguration an Hyperparametern λ, die aus dem Raum Λ das beste Ergebnis bezogen auf die Validierungsdaten besitzt.
Sei α ein Algorithmus aus der Menge aller Algorithmen A und sei δ eine Vorverarbeitungskonfiguration aller Konfigurationen Δ. Neben den einzustellenden Modellparametern kann auch die Wahl des zugrundeliegenden Modells bzw. die Wahl des Lernalgorithmus α einen eigenen Hyperparameter darstellen, wenn gilt: $\alpha \in A$. Diese kombinierte Einstellmöglichkeit von Hyperparametern und Algorithmusauswahl wird in der Literatur auch als Combined Algorithm Selection and Hyperparameter Optimization Problem (CASH) bezeichnet (vgl. [HKV19; Feu+15]).
Neben der geeigneten Algorithmusauswahl α und dessen Einstellparameter λ können auch die Eingangsdaten durch unterschiedliche Verfahren δ vorverarbeitet werden, bevor sie zum Training in das Modell gegeben werden, wenn gilt: $\delta \in \Delta$. Wie auch Schilling u. a. in [Sch+15] zeigen, kann die Wahl und Konfiguration der Verfahren zur Vorverarbeitung der Daten auch durch Hyperparameter dargestellt werden. Das heißt, die gesamten Hyperparameter umfassen sowohl die Einstellparameter der Lernalgorithmen, die Wahl der Lernalgorithmen, als auch die entsprechenden Parameter der Datenvorverarbeitung. Ziel ist es, ein geeignetes Modell α mit entsprechenden Einstellparametern λ und eine geeignete Vorverarbeitung der Daten δ hinsichtlich einer optimalen Modellgüte J zu finden. Die Formel aus 3.1 kann nun in 3.2 weiter präzisiert werden:

$$\alpha^*, \lambda^*, \delta^* = \arg\min_{\alpha \in A, \lambda \in \Lambda, \delta \in \Delta} J(\alpha_\lambda, X_{\delta,train}, X_{\delta,valid}) \tag{3.2}$$

3.7.3 Lösungsmöglichkeiten

Im Folgenden werden heuristische Ansätze vorgestellt, die zur Optimierung eines einstellbaren Parameterraums eingesetzt werden können. Dazu zählen evolutionäre Verfahren, Rastersuche, Zufallssuche, die Bayessche Optimierung und gradientenbasierte Verfahren [And19;

HKV19]. Diese problemspezifische Heuristiken können ein Ergebnis für das vorliegende Problem in akzeptabler Zeit liefern, dieses entspricht aber nicht zwangsläufig dem globalen Optimum [VW16, S. 423]. Heuristiken versuchen annähernd optimale Lösungen eines Problems bei effizienterem Ressourceneinsatz zu finden [BD01; MDM11].

Evolutionäre Verfahren Die evolutionären Verfahren sind dadurch gekennzeichnet, dass zufällig ausgewählte initiale Konfigurationen der Hyperparameter verwendet und das Modell anschließend hinsichtlich eines ausgewählten Qualitätsmerkmals bewertet wird. Die schlechtesten Konfigurationen werden im weiteren Verlauf nicht mehr betrachtet und durch neue Konfigurationen modifiziert. Dies wird solange wiederholt, bis die eingestellte Mindestqualität erreicht ist.
Wicaksono und Afif zeigen in ihrer Arbeit, dass ein evolutionäres Verfahren zwar ähnliche Ergebnisse liefern kann wie eine Rastersuche, allerdings bei einem weitaus geringeren zeitlichen Aufwand für ausgewählte ML-Verfahren [WA18]. Dabei stand ein Datensatz mit 39797 Samples und 61 Merkmalen zur Verfügung.

Rastersuche Im Rahmen der Rastersuche (engl. *grid search*) wird zunächst eine Menge an Parametern festgelegt, die untersucht werden soll. Wenn nötig, werden hier für die Rastersuche Diskretisierungen der Hyperparameter vorgenommen. Auch hier wird das aus den unterschiedlichen Konfigurationen erstellte Modell hinsichtlich eines ausgewählten Qualitätsmerkmals bewertet. Typischerweise werden zur Validierung nicht die Trainingsdaten, sondern nicht angelernte Validierungsdaten verwendet. Im weiteren Verlauf wird über alle Konfigurationen iteriert und die entsprechende Vorhersagegüte jeder Konfiguration gespeichert.
Die Rastersuche wird typischerweise angewendet, wenn der Nutzer über ein vergleichsweise hohen Wissensstand der einzustellenden Hyperparameter verfügt. Die Rastersuche sollte jedoch vermieden werden, wenn zu viele Konfigurationsmöglichkeiten vorliegen [YZ20; And19].

Zufallssuche Die Zufallssuche (engl. *random search*) versucht die Nachteile der Rastersuche aufzuwiegen [YZ20]. Hierbei ist keine Diskretisierung für reelle Hyperparameter vorzunehmen und auch nicht alle möglichen Konfigurationen werden ausprobiert. Stattdessen wird eine zufällige Auswahl an Konfigurationen von Hyperparametern bestimmt und hinsichtlich der Modellqualität bewertet. Der Suchraum kann auch während eines Vorgangs ergänzt werden, falls erwünscht. Im Vergleich zur Rastersuche kann die Zufallssuche auch frühzeitig beendet werden, z.B. sobald eine Mindestqualität erreicht ist.
In vielen Fällen kann die Zufallssuche bessere Ergebnisse als die Rastersuche liefern, bei vergleichsweise rechenintensivem Aufwand [YZ20]. Liegt ein hochdimensionaler Suchraum vor, so kann der Berechnungsaufwand mit Hilfe der Zufallssuche gegenüber der Rastersuche sogar signifikant reduziert werden [And19; LJA16].

Bayessche Optimierung Im Rahmen der bayesschen Optimierung werden unterschiedliche Konfigurationen der Hyperparameter von dem Verfahren eigenständig bestimmt und optimiert. Im Vergleich zu den bisher vorgestellten Optimierungsverfahren werden bei der bayesschen Optimierung bereits berechnete Ergebnisse der Konfigurationen für zukünftige Selektion von Hyperparametern betrachtet [And19]. Hierbei werden unwichtigere und schlechtere Hyperparameterkonfigurationen vernachlässigt, wohingegen vielversprechende Konfigurationen weiter optimiert werden können. Um das zu erreichen, wird für jedes Ergebnis eine Wahrscheinlichkeitsverteilung berechnet. Diese Verteilung wird mit jeder neuen getesteten Konfiguration aktualisiert. Je mehr Konfigurationen bewertet werden, desto präziser wird der zukünftige Suchraum durch das Verfahren eingegrenzt [And19].
Ein spezifisches Wissen der Hyperparameterverteilung ist vom Nutzer nicht erforderlich. Aufgrund der Wahl vielversprechender Konfigurationen ist die bayesschen Optimierung recheneffizienter als die bisher vorgestellten Methoden. Eine besondere Herausforderung stellt dabei jedoch die Kombination aus numerischen und kategorialen Einstellparametern dar.

Gradientenbasierte Verfahren Es gibt Lernalgorithmen, die die eingestellten Hyperparameter hinsichtlich der Gradienten auswerten. Sofern ein solcher Lernalgorithmus angewendet wird, können geeignete Hyperparameter im Rahmen des gradientenbasierten Verfahrens bestimmt werden. Dabei werden die Hyperparameter hinsichtlich des höchsten Gradienten (dies wird auch als Verfahren des steilsten Abstiegs genannt) optimiert. Die Konvergenz zu einem globalen Optimum kann bei Vorhandensein mehrere lokaler Optima bei diesem Verfahren nicht garantiert werden [DKW18].

Die Autoren Macready und Wolpert beschreiben im „No-free-lunch“-Theorem, dass es keinen universellen Lernalgorithmus für alle Probleme gibt, der generell anderen Algorithmen überlegen ist. Je nach Anwendungsfall und Beschaffenheit der Daten haben unterschiedliche Lernalgorithmen Vor- und Nachteile. Datensätze können domänenspezifische Charakteristiken aufweisen, die sich mit bestimmten Algorithmen besser analysieren lassen. [MW97]
Zur Optimierung des Hyperparameter-Suchraums wurden unterschiedliche Ansätze vorgestellt. Unter Anwendung der Zufallssuche werden nur zufällig ausgewählte Konfigurationen an Hyperparametern verwendet. Bei der Rastersuche wird der Suchraum schon zu Beginn eingeschränkt, sodass beide Verfahren nicht notwendigerweise zum globalen Minimum konvergieren können. Die Anwendung gradientenbasierter Verfahren kann nicht auf beliebige Lernalgorithmen erfolgen. Im Sinne des CASH-Ansatzes (vgl. Kapitel 3.7.2) ist zu diesem Zeitpunkt allerdings noch kein Lernalgorithmus festgelegt. Aus diesem Grund eignen sich gradientenbasierte Verfahren nicht für das vorliegende Problem. Alle vorgestellten Ansätze zur Hyperparameteroptimierung profitieren, wenn einzelne Hyperparameter zur virtuellen Modellerstellung bereits im Vorhinein reduziert werden können.
Das vorliegende NP-vollständige Problem besitzt eine so hohe Komplexität, dass die Berechnung einer exakten Lösung des Optimierungsproblems einen unverhältnismäßigen rechenintensiven Aufwand bedeuten würde. Wenn eine Parameter-Konfiguration aus einem

Raum von 100 Eingangssignalen bestünde, so gäbe es allein 2^{100} mögliche Kombinationen, unterschiedliche Teilmengen von diesen Signalen zu bilden. Ein Algorithmus mit einer Rechendauer von 1ms pro Iteration würde allein für dieses Minimalbeispiel 10^{19} Jahre benötigen. Um Probleme dieser Art überhaupt lösen zu können, werden deshalb problemspezifische Heuristiken verwendet [BD01]. Außerdem bietet es sich an, die Komplexität des Problems bereits im Rahmen der Konzepterstellung zu reduzieren.

Das hier gezeigte Hyperparameter-Optimierungsproblem ist dadurch gekennzeichnet, dass es viele Parameter gibt, die es gilt zu optimieren. Wenn einzelne Parameter dieses Suchraums bereits vor der eigentlichen Optimierung eingeschränkt werden können, kann die gesamte Komplexität um ein Vielfaches reduziert werden. Sobald die Parameter festgestellt und vorausgewählt sind, kann eine Optimierung mit Hilfe der hier vorgestellten Ansätze erfolgen.
Im weiteren Verlauf sollen geeignete Hyperparameterausprägungen vorausgewählt und anschließend eine geeignete Konfiguration an Hyperparametern durch Anwendung des bayesschen Optimierungsansatzes durchgeführt werden. Mit Hilfe dieses Vorgehens können geeignete Hyperparameter in akzeptabler Zeit gefunden werden, die die Alterung mit Hilfe eines rechnergestützten Lernalgorithmus hinreichend genau vorhersagen können. Die einzelnen Parameter der Vorverarbeitung werden im Vorfeld diskretisiert und ausgewählt. Neben den Parametern der Vorverarbeitung sind auch die Parameter des Data-Minings zu bestimmen. Dieses kombinierte Vorgehen (vgl. CASH) ist durch einen großen Parameterraum charakterisiert [Feu+15]. Im Rahmen des Konzeptentwurfs werden Vorgehensweisen vorgeschlagen, wie einzelne Parameter der Vorverarbeitung schon bereits vor der eigentlichen Optimierung vorausgewählt werden können (vgl. Kapitel 4). Die Bestimmung eines exakten Problemoptimums ist nicht notwendig. Das gesamte Anwendungsproblem der Alterungsvorhersage ist mit Ungenauigkeiten behaftet, die sowohl von Messungenauigkeiten bei der Bestimmung der Komponenten-Performance (vgl. Kapitel 3.5), als auch durch interne Verzögerungen bei der Signalübertragung auftreten können. Deshalb ist an dieser Stelle davon auszugehen, dass schon am Eingang des Lernalgorithmus keine exakte Datenbasis zur Verfügung steht. Außerdem liegt nur eine begrenzte Menge an Signalen vor, die nicht den gesamten Informationsfluss im Fahrzeug abdecken. Für den Vergleich unterschiedlicher Parameter-Konfigurationen wird die Qualität der Alterungsvorhersage im weiteren Verlauf bewertet und untereinander verglichen. Eine für den Anwendungsfall ausreichende Genauigkeit der Alterungsvorhersage wird akzeptiert.
Unter Einbeziehung der Realisierbarkeit des in dieser Arbeit gezeigten Anwendungsfalls in einer Fahrzeug/Cloud-Umgebung zeigt sich außerdem, dass *eine* geeignete Lösung des Problems einer exakten Lösung vorgezogen wird. Ein solcher Konzeptentwurf für eine datengetriebene Alterungsvorhersage mit Hilfe von Fahrzeugdaten wird im folgenden Kapitel 4 vorgestellt.

4 Konzeptentwurf für eine datengetriebene Alterungsvorhersage

Im Folgenden wird das Konzept zur Erstellung eines virtuellen Sensors zur Abnutzungsbestimmung einer Komponente unter Analyse eines gegebenen Datenbestandes vorgestellt. Das allgemeine Vorgehen zur Extraktion relevanter Muster und Regeln mit Hilfe von Algorithmen aus gegebenen Daten wird in der Literatur als Knowledge Discovery in Databases (KDD) bezeichnet [FPS96; Bor97; SSG11; ES00a; CHT09] und beinhaltet folgende drei Arbeitsschritte: Datenaufnahme und -selektion, das Data-Mining und die Interpretation der Ergebnisse. Die Industrie hat diesen Ansatz weiter entwickelt und daraus den Standard-Prozess CRISP-DM formuliert [LC20; GB15]. Der Cross-industry standard process for data mining (CRISP-DM) beschreibt ein definiertes Vorgehen von Data-Mining-Experten und wird nach Azevedo und Santos als Implementierung des KDD-Prozesses verstanden [AS08].
Diese grundsätzlichen Schritte des KDD-Prozesses (bzw. des CRISP-DM-Prozesses) bilden die Struktur des weiteren Vorgehens für einen Konzeptentwurf einer datengetriebenen Alterungsvorhersage in dieser Arbeit. Hierbei werden auch erste konzeptuelle Einschränkungen getroffen, sodass die in Kapitel 3.7 vorgestellte Komplexität hinsichtlich des Anwendungsgebiets minimiert wird. Es werden unterschiedliche Parameter vorgestellt, die sowohl die Datenvorverarbeitung als auch das Data-Mining betreffen. Die kombinierte Festlegung der einzelnen Parameter wird im weiteren Verlauf zusammenfassend durch *Hyperparameter* beschrieben. Mit Hilfe der zur Verfügung stehenden Datenbestandes, den Hyperparametern und den Methoden des Data-Minings kann so ein virtueller Sensor zur Alterungsbestimmung modelliert und bewertet werden.
Innerhalb dieses Kapitels werden Ansätze zur Beantwortung der Forschungsfragen **RQ 1** und **RQ 2** vorgestellt. Im Rahmen der *Datenvorverarbeitung* (vgl. Kapitel 4.3) werden die unterschiedlichen zeitlichen Auflösungen der hochdynamischen Eingangsdaten und der Zielgröße zur weiteren Verarbeitung vorbereitet und Merkmale für weitere Analysen extrahiert. Dieses Vorgehen adressiert Lösungsansätze auf die Forschungsfrage **RQ 1**. Der

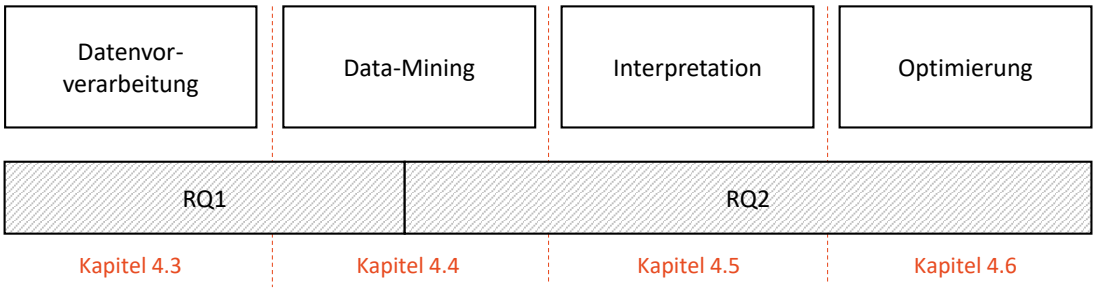


Abbildung 4.1: Darstellung der Forschungsfragen als strukturierendes Element in der Konzeptvorstellung

Abschnitt *Data-Mining* (vgl. Kapitel 4.4) beschreibt die Methoden des MLs. Sowohl die

Forschungsfrage **RQ 1** als auch **RQ 2** werden in diesem Abschnitt angesprochen. Weiterhin wird die Forschungsfrage **RQ 2** in den Abschnitten *Interpretation* (vgl. Kapitel 4.5) und *Hyperparameteroptimierung* (vgl. Kapitel 4.6) adressiert. Hier werden Vorgehensweisen vorgeschlagen, wie die Vorhersagequalität des Modells bestimmt wird und wie sie von den unterschiedlichen Methoden und den Parametern der Datenvorverarbeitung abhängig ist.
Die letzte Forschungsfrage **RQ 3** bezieht sich auf die Evaluation des Sensormodells und wird im anschließenden Kapitel 5 diskutiert. Die Abbildung 4.1 zeigt die Zugehörigkeit der Forschungsfragen **RQ 1** und **RQ 2** innerhalb der gesamten Konzeptvorstellung. Das datengetriebene Gesamtkonzept zur Erstellung eines virtuellen Sensors zur Alterungsbestimmung ist in der abstrakten Beschreibung abhängig vom Datenbestand, den gewählten Hyperparametern und den Methoden des Data-Minings.
Die mit Datenaufzeichnungsgeräten gespeicherten dynamischen Eingangsdaten sind unveränderbar. Die Hyperparameter sind im Rahmen der Untersuchung dagegen veränderbar und werden im weiteren Verlauf näher beschrieben. Zu den Einstellmöglichkeiten der Hyperparameter gehören die Parameter der Datenvorverarbeitung (Kapitel 4.3) und die Parameter des Data-Minings (Kapitel 4.4). Der gesamte abstrakte Suchraum wird in Kapitel 4.6 weiter beschrieben und in Kapitel 4.7 zusammengefasst. Hier wird auch vorgeschlagen, wie aus dieser Vielzahl an Kombinationsmöglichkeiten der Hyperparameter eine geeignete Lösung im Rahmen einer Optimierung gefunden werden kann.

4.1 Konzeptvorstellung

Im weiteren Verlauf wird zwischen den zur Verfügung stehenden *Daten* und dem daraus abgeleiteten *Wissen* getrennt. Borgelt unterscheidet in [Bor97] zwischen Daten und Wissen wie folgt: Die zu analysierenden Daten beziehen sich auf spezielle Einzelfälle und enthalten individuelle Eigenschaften. Mit Hilfe der zahlreich aufgenommenen Daten lassen sich ohne weitere Analysen keine Vorhersagen treffen. Zur Wissensgenerierung werden aus den Daten deshalb Muster und Strukturen extrahiert [HPK12]. Dieses Wissen ermöglicht nicht nur Rückschlüsse auf Einzelfälle, sondern auch Vorhersagen für Fallklassen [Bor97].
Folgende fünf Schritte werden im Rahmen des Konzepts durchgeführt:

- Datenaufnahme und -selektion (Kapitel 4.2)
- Datenvorverarbeitung (Kapitel 4.3)
- Data-Mining (Kapitel 4.4)
- Interpretation (Kapitel 4.5)
- Hyperparameteroptimierung (Kapitel 4.6)

Zunächst wird die Datenaufnahme und -selektion (vgl. Kapitel 4.2) erläutert. Neben der Sicherstellung der zur Weiterverarbeitung notwendigen Datenqualität werden die Daten außerdem von möglichen fehlerhaften Datenpunkten bereinigt. Im Abschnitt der Datenvorverarbeitung (vgl. Kapitel 4.3) werden unterschiedliche Vorgehensweisen vorgestellt. Dazu werden die Zeitreihen in Sequenzen auf Basis einer gewählten Diskretisierungsstufe aufgeteilt. Innerhalb dieser Sequenzen werden Merkmale gebildet, die im weiteren Verlauf von den Methoden des MLs genutzt werden, um datengetriebene Modelle zu erstellen.
Im nächsten Schritt, dem sogenannten *Data-Mining* (vgl. Kapitel 4.4), werden Zusammenhänge aus den Daten mit Hilfe unterschiedlicher Verfahren beschrieben. Die Eingangsdaten können speziellen Gruppen zugeordnet (engl. *clustering*) oder es werden mögliche Vorhersagen getroffen (engl. *prediction*). Dazu werden die Methoden des maschinellen Lernens verwendet, die bereits in Kapitel 2.1.1 vorgestellt worden sind. Mit Hilfe dieser Wissensgenerierung soll ein virtueller Sensor zur Abnutzungsbestimmung aus den Daten abgeleitet werden. Im Anschluss folgt die Interpretation der Ergebnisse. Darunter fallen neben einer visuellen Darstellung auch die Bestimmung der Vorhersagequalität und die Überprüfung der Generalisierungsfähigkeit der Methode (vgl. Kapitel 4.5). Dabei werden die Daten nach bestimmten Verfahren in Trainings- und Testdaten aufgeteilt, um so die Ergebnisse validieren zu können.
Die Parameter der Vorverarbeitung und des Data-Minings werden zusammenfassend durch Hyperparameter beschrieben. Unter Anwendung bestimmter Hyperparameter wird eine entsprechende Vorhersagegüte der Prädiktion erzielt. Aus diesem Grund entsteht eine Abhängigkeit zwischen der Wahl der Hyperparameter und dem daraus resultierendem Vorhersageergebnis, die im Vorhinein nicht festgestellt werden kann. Im Schritt der Hyperparameteroptimierung (vgl. Kapitel 4.6) wird ein Vorgehen vorgeschlagen, wie ein virtueller Sensor zur Alterungsbestimmung geeignet modelliert werden kann.
Zusammenfassend wird das gesamte Konzept in der Abbildung 4.2 dargestellt. Die einzelnen Kapitel 4.2 bis Kapitel 4.6 sind dort als übergeordnete Kästchen dargestellt. Im Rahmen der Ergebnisinterpretation wird auch die Vorhersagequalität bestimmt. Diese ist abhängig von den gewählten Hyperparametern, sodass an dieser Stelle eine Rückkopplung besteht. Die Abschnitte mit einstellbaren Hyperparametern (vgl. Kapitel 4.3 und Kapitel 4.4) sind in der Abbildung 4.2 durch eine graue Schraffur gekennzeichnet. Im weiteren Verlauf dieser Arbeit werden die Hyperparameter in zwei Teilmengen geteilt: Hyperparameter der Vorverarbeitung und Hyperparameter des Data-Minings.

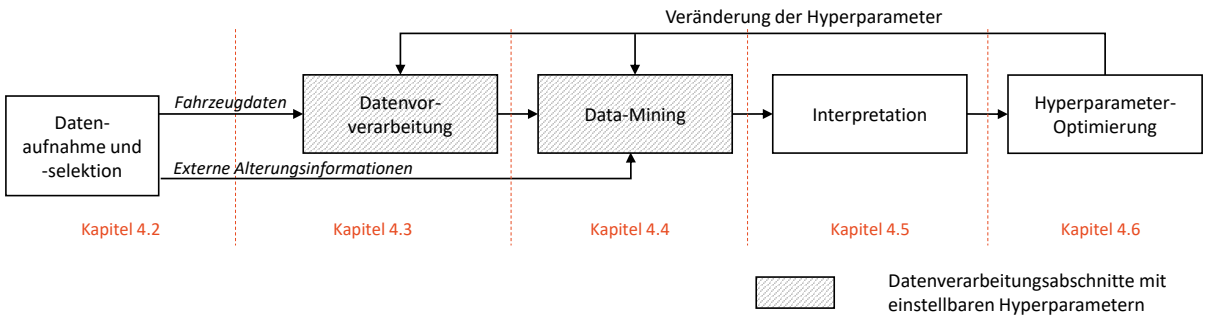


Abbildung 4.2: Schematische Darstellung des groben Konzeptentwurfs im Überblick

4.2 Datenaufnahme und -selektion

Die im weiteren Verlauf genutzte Datengrundlage wird mit Hilfe von Datenaufzeichnungsgeräten (hier Fahrzeugdatenloggern) von unterschiedlichen Systemen gewonnen. Weitere Ausführungen zum Versuchsaufbau werden in Kapitel 2.3 vorgestellt.
Da nur eine begrenzte Anzahl an Signalen (hier CAN-Botschaften oder Fahrzeugsignale) aufgezeichnet werden, sollte die Konfiguration des Datenaufzeichnungsgeräts für alle zu analysierenden Systeme identisch sein. Außerdem sollte darauf geachtet werden, dass die Signale, die sich im Umfeld der zu analysierenden Komponente befinden, in jedem Fall aufgezeichnet werden. Auf Grund der Konfiguration des Datenaufzeichnungsgeräts und der Begrenzung der Kapazität des Speichermediums ist eine daraus resultierende Datenselektion unvermeidbar.
Neben einer fehlerfreien Aufzeichnung der Signale sind für eine Erstellung eines virtuellen Sensormodells auch Informationen zum Grad der Abnutzung zu bestimmen. Diese Informationsbeschaffung erfordert meist einen hohen zeitlichen Aufwand und einen umfangreichen Wartungsbedarf. Es ist darauf zu achten, dass die Genauigkeit des zu erstellenden Modells

Tabelle 4.1: Unterschiedliche Bezeichnungen für die Einfluss- und Zielgrößen

x	**y**
Einflussgröße	Zielgröße
Feature	Label
Merkmale der Eingangsdaten	Performancegröße

steigt, wenn diese Performance-Informationen in einer hohen Auflösung zur Verfügung stehen. Die Tabelle 4.1 stellt unterschiedliche Bezeichnungen für die Einfluss- und Zielgröße dar, die zur Wissensextraktion angewendet werden.

Der aktuelle Abschnitt *Datenaufnahme und -selektion* teilt sich wiederum in den Teilbereich *Sicherstellung der Datenqualität*, *Datensynchronisierung* und die *Eliminierung redundanter Signale* auf. Im ersten Unterabschnitt werden die Herausforderungen zur Sicherstellung der Datenqualität beschrieben. Dies betrifft sowohl die internen Fahrzeugsignale der einzelnen Fahrzeuge als auch die extern aufgenommenen Performance-Informationen der zu untersuchenden Fahrzeugkomponenten. Neben der Sicherstellung der Datenqualität wird in einem weiteren Unterabschnitt die Datensynchronisierung erläutert. Für diesen Zweck werden die unveränderten Messdaten nach einem bestimmten Vorgehen synchronisiert. In einem weiteren Unterabschnitt wird die Eliminierung redundanter Signale vorgestellt. Eine Vielzahl von Signalen werden mit Hilfe des Datenaufzeichnungsgeräts gespeichert, die denselben Informationsgehalt liefern können. Aus diesem Grund wird ein Vorgehen vorgeschlagen, welches diese redundanten Signale erkennt und nur einen Repräsentanten einer Gruppe zur weiteren Verarbeitung auswählt.

Sicherstellung der Datenqualität Für eine gute Vorhersagequalität des Modells sowie dessen Generalisierungsfähigkeit (vgl. Kapitel 2.1.1) ist eine hohe Datenqualität erforderlich. Dies gilt nicht nur für die Daten, die mittels Fahrzeugdatenlogger aufgenommen werden, sondern auch für die manuell gemessenen Performance-Werte. Die Dateigrößen stehen dabei in einem direkten Zusammenhang zur festgelegten Auflösung, bzw. Aufzeichnungsfrequenz. Dennoch ist eine fortlaufende Aufzeichnung nicht immer gewährleistet, weil zum Beispiel die Konfiguration eines Fahrzeugdatenloggers im Verlauf des Fahrzeuglebens geändert wird. Das kann zur Folge haben, dass eine Auswahl an Signalen nur bis zu einem entsprechenden Zeitpunkt aufgezeichnet werden und ab diesem Zeitpunkt keine weiteren Daten liefern können. Die Abbildung 4.3 stellt die Auswirkungen einer Änderung der Fahrzeugdatenlogger-Konfiguration beispielhaft dar. Das Signal 1 wird fortlaufend dar-

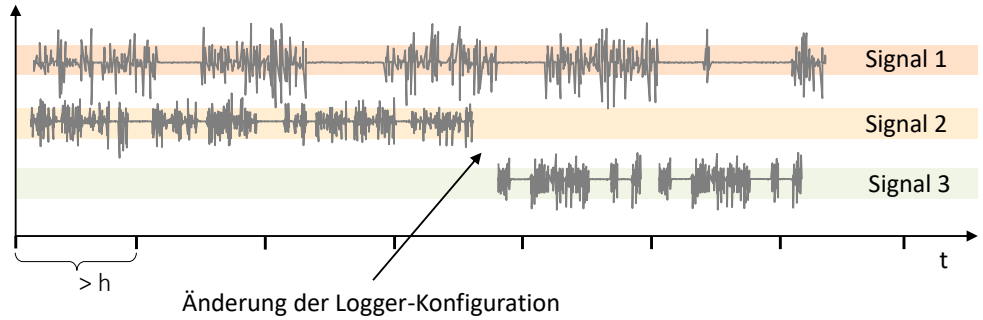


Abbildung 4.3: Darstellung unterschiedlicher Signale bei einer Änderung der Fahrzeugdatenlogger-Konfiguration

gestellt. Das Signal 2 wird nur bis zur Änderung der Konfiguration aufgezeichnet, wohingegen das Signal 3 erst ab diesem Zeitpunkt Aufzeichnungswerte aufweist. Im Rahmen der Analyse ist es nicht möglich, die Gründe oder Ursachen einer solchen Änderung zu identifizieren. Diese Konfigurationsänderungen sind in den Eingangsdaten enthalten und lassen sich nicht ändern. Das zu erstellende Konzept sieht aber vor, Signale mit einer zu geringen Aufzeichnungsdauer zu entfernen.
Auch ohne eine Änderung der Konfiguration oder Unterbrechung einer Aufzeichnung können Aufzeichnungsfehler auftreten oder fehlerhafte Daten übermittelt werden. Einige Fahrzeugsignale stammen von Sensoren, die im Fahrzeug verbaut sind. Diese können messtechnischen Schwankungen ausgeliefert sein und deshalb fehlerhafte Werte liefern [TL19]. Außerdem kann es zu Problemen bei der Übertragung von Informationen vom CAN-Bus zum Fahrzeugdatenlogger kommen. Weiterhin ist zu beachten, dass keine Aufzeichnungen gespeichert werden können, wenn das Speichermedium seine Kapazität erreicht hat.

Datensynchronisierung Neben der reinen Datenaufnahme und Sicherstellung einer ausreichenden Aufzeichnungsqualität ist eine Vorverarbeitung dieser Daten zur weiteren Analyse notwendig. Die aufgezeichneten Fahrzeugsignale besitzen auf Grund der nichtdeterministischen Kommunikation des CAN-Busses unterschiedliche zeitliche Auflösungen und demzufolge auch unterschiedliche Zeitstempel. Für weitere Analysen wird die gesamte Datengrundlage in ein einheitliches zeit-synchronisiertes Raster transferiert. Dadurch wird ein Datensatz erzeugt, der auch für weitere Analysen anwendbar ist. Der zeitliche Bezug ist

zwar nach wie vor vorhanden, wird aber nicht zwingend bei jeder Berechnung benötigt. Für weitergehende Betrachtungen eines Fachexperten können unter Anwendung eines synchronen Datensatzes hilfreiche Einflussanalysen erstellt werden. Folgendes Beispiel zeigt eine mögliche Einflussanalyse: *Stelle das Signal 2 an den Zeitpunkten grafisch dar, wenn Signal 1 einen Wert von über 90 überschreitet*. Des Weiteren können mit Hilfe dieser synchronisierten Datengrundlage die bereits in der Problemstellung (vgl. Kapitel 3.2) vorgestellten unterschiedlichen Diskretisierungsstufen direkt berechnet werden, ohne die ursprünglichen asynchronen Eingangsdaten erneut einzulesen. Für eine herkömmliche Interpolation werden bevorzugt numerische Werte benutzt, da diese auch Zwischenwerte annehmen können. Kategoriale Werte, wie zum Beispiel die Gangposition oder Statusbits, können dagegen keine Zwischenwerte annehmen. Allerdings soll das genannte Vorgehen nicht nur auf numerische Fahrzeugsignale beschränkt bleiben. Aus diesem Grund wird im Folgenden ein Vorgehen beschrieben, dass sowohl für numerische als auch kategoriale Fahrzeugsignale verwendet werden kann.
Die Abbildung 4.4 zeigt die eben eingeführte Synchronisierung der Messwerte. Dabei werden die Werte der einzelnen Signalwerte nicht interpoliert. Stattdessen wird der Messwert benutzt, der zum entsprechenden Zeitpunkt des vorgegebenen Rasters gültig ist. Zum Zeitpunkt der Vorbereitung der Eingangsdaten steht noch nicht fest, welches die zu verwendende Diskretisierungsstufe für eine effiziente Alterungsvorhersage ist. Auch bei kleinen Diskretisierungsstufen (entspricht einer hohen Auflösung) ist ein gültiger Signalwert bereitzustellen. Deshalb sollten die Eingangsdaten mindestens so hoch aufgelöst sein wie die spätere Diskretisierungsstufe. Mit diesen beiden Anforderungen entstehen nun äquidistante Messwerte von allen internen Fahrzeugsignalen.

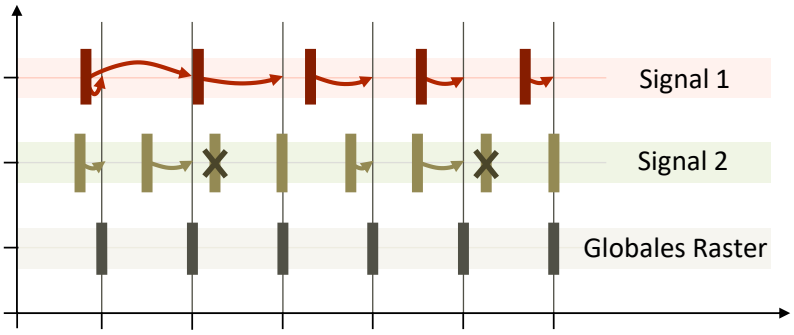


Abbildung 4.4: Darstellung der Synchronisierung der Messsignale unter Vorgabe eines globalen Rasters, in Anlehnung an [SEF19]

Mit Hilfe eines Referenzsignals können die Bereiche in den Zeitreihen identifiziert werden, in denen eine valide Aufzeichnung stattfand. Während der gesamten Aufzeichnung müssen gültige Werte dieses Referenzsignals vorhanden sein. Mit Hilfe dieses Referenzsignals können nun die zeitlichen Lücken ermittelt werden, in denen keine Aufzeichnung stattfand, zum Beispiel auf Grund von Stillstandszeiten des Fahrzeugs. Die einzelnen zeitlichen Bereiche einer validen Aufzeichnung werden miteinander verknüpft. Daraus entsteht eine Datenmenge, bei der alle Fahrzeugsignale dieselbe Anzahl an Werten aufweisen und zeitlich synchronisiert sind.

Ist ein Signal in der Datenbasis nicht nahezu vollständig vorhanden, wird dieses Signal für eine weitere Analyse nicht weiter betrachtet. In der zuvor dargestellten Abbildung 4.3 würden aus diesem Grund das Signal 2 und das Signal 3 für eine weitere Analyse entfernt werden. Zur Erstellung eines datengetriebenen Modells können nur Signale verwendet werden, die während des Betriebes des Fahrzeugs durchgängig zur Verfügung standen.

Eliminierung von redundanten Signalen Da es sich um ein datengetriebenes Konzept zur Vorhersage der Abnutzungserscheinung handeln soll, werden zunächst alle aufgezeichneten Signale in Betracht gezogen. Ein Fachexperte kennt die physikalischen Zusammenhänge eines Systems und weiß wohlmöglich, welche Signale zur Erstellung eines Modells relevant sein werden. Im Gegensatz dazu hat ein zu erstellendes datengetriebenes Modell kein Vorwissen und wird diese Signalauswahl eigenständig vornehmen.
Bei Betrachtung einer langfristigen physikalischen Alterung (oder auch *Driftausfall*, wie bereits in Kapitel 3.1 vorgestellt) liefern verwechselbare Signale keine zusätzlichen Informationen. Als Beispiel für diese verwechselbaren Signale seien an dieser Stelle *Geschwindigkeitssignale* erwähnt. Diese können am Fahrzeug an allen vier Rädern gemessen werden, so entstünden die Geschwindigkeitssignale *Fahrzeuggeschwindigkeit_VL (vorne links), Fahrzeuggeschwindigkeit_VR, Fahrzeuggeschwindigkeit_HL, Fahrzeuggeschwindigkeit_HR*. Es ist nicht das Ziel der Arbeit ähnliche Signale miteinander zu vergleichen. Stattdessen werden langfristige Änderungen der Signalverläufe genutzt, um eine Alterung vorherzusagen. Aus diesem Grund werden im Rahmen dieser Arbeit nicht die vier genannten Fahrzeugsignale genutzt, sondern nur ein (einzigartiger) Vertreter der Geschwindigkeit.
Statt einer Signalvermischung bzw. -kombination aus unterschiedlichen Cluster-Teilnehmern wird ein Repräsentant eines gefundenen Clusters ausgewählt, um so die Nachvollziehbarkeit des Modells zu gewährleisten. Aus diesem Grund werden diejenigen Signale aus der Eingangsdatenmenge entfernt, die nachweislich den gleichen Informationsgehalt aufweisen. Zur Identifikation dieser redundanten Signale wird ein Clustering-Algorithmus verwendet. Die resultierenden einzigartigen Signale werden den Methoden des Data-Minings zur Verfü-

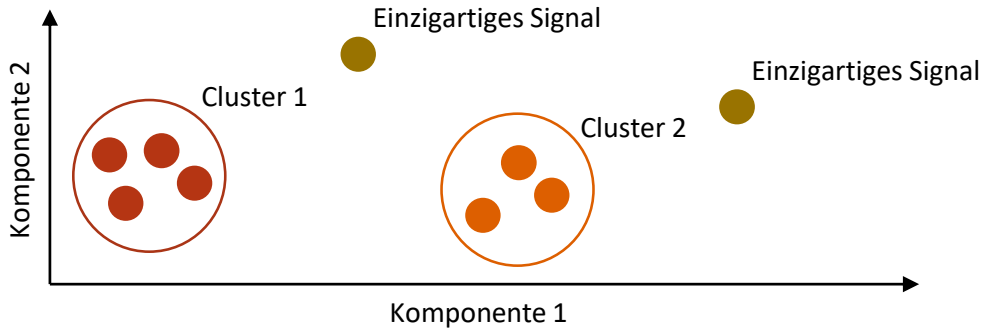


Abbildung 4.5: Darstellung zur Identifikation von einzigartigen und verwechselbaren Signalen zur Signalreduktion unter Anwendung der Clusteranalyse

gung gestellt. Im Rahmen der Hyperparameteroptimierung kann die Wahl der einzigartigen Signale weiter eingeschränkt werden. Eine Abhängigkeit der Abnutzungserscheinung von einer Teilmenge an einzigartigen Signalen ist denkbar.

4.3 Datenvorverarbeitung

Im Rahmen der Datenvorverarbeitung werden Verarbeitungsschritte beschrieben, wie die einzelnen Daten bei unterschiedlichen Auflösungen zu einer gemeinsamen Datenmenge vereint und zur weiteren Analyse vorverarbeitet werden können.
Wie bereits in Kapitel 3.2 erwähnt, ist es zur Erstellung eines effizienten Modells wichtig, dass Informationen in Form von ausgewählten Features zur Verfügung stehen ([Liu+02; TC19]). Ein datengetriebenes Modell kann dann effizient arbeiten, wenn in den vorliegenden Eingangsdaten auch die zur Alterung vermeintlich relevanten Informationen enthalten sind.
Diese Informationen können auf Ebene der gesamten Signalmenge abgeleitet werden. Sei die zu untersuchende Abnutzungserscheinung eine Ölalterung, so ließe sich diese in internen Temperatur- oder Drucksignalen erkennen. Wohingegen eine mögliche Alterung der Bremse in den Signalen der Bremsdauer oder des Bremsdrucks erfasst werden könnte.
Um ein möglichst effektives datengetriebenes Modell zu erstellen, werden aus den hochaufgelösten Zeitreihen der internen Fahrzeugsignale Merkmale berechnet. Diese Merkmale werden innerhalb von Sequenzen einer Diskretisierungsstufe gebildet. Mit dieser Datenqualität wird eine Modellgeneralisierungsfähigkeit ermöglicht. Dadurch wird eine Nutzung des datengetriebenen Modells eines Fahrzeugs auch für weitere Fahrzeuge ermöglicht, deren Daten nicht angelernt worden sind. Bei der Betrachtung eines einzelnen Signals kann die Wahl unterschiedlicher Merkmale unterschiedlich performante Vorhersage-Ergebnisse liefern. So kann ein Mittelwert des Geschwindigkeitssignals Auskunft über das Fahrprofil (z.B. privater Pendler oder gewerblicher Materialtransport) eines Fahrzeughalters geben. Mit Hilfe der Standardabweichung des Geschwindigkeitssignals kann dagegen Auskunft über ein mögliches dynamisches Fahrverhalten eines Fahrzeughalters (ein sportlicher oder ein komfortabler Fahrer) gegeben werden.
Im weiteren Verlauf wird zwischen der Wahl der Diskretisierungsstufe (vgl. Kapitel 4.3.1) und der Auswahl der Merkmale (vgl. Kapitel 4.3.2) differenziert. Die Diskretisierungsstufe gibt an, wie lang die Sequenz in der ursprünglichen Auflösung ist, in der die Merkmale berechnet werden. Hunderte bis tausende Datenpunkte innerhalb einer Sequenz werden dabei von wenigen festdefinierten Merkmalen repräsentiert. Die Wahl der für die Alterungsvorhersage benutzten Merkmale wird in Kapitel 4.3.2 vorgestellt. Dabei werden die ausgewählten Merkmale für jedes (Mess-)Signal berechnet.

4.3.1 Wahl der Diskretisierungsstufe

Die Wahl der Diskretisierungsstufe bestimmt, wie groß die ausgewählten Sequenzen sind, in denen die hochdynamischen Eingangsdaten aggregiert werden. Die Aggregation der Daten wird über die Merkmale der hochdynamischen Signale innerhalb einer Sequenz bestimmt. Aus den berechneten Merkmalen entsteht ein Merkmalsvektor. Dieser gehört zu einer bestimmten Sequenz des Signals. Alle Sequenzen enthalten dieselbe Anzahl an Messpunkten

bei einer festgelegten Diskretisierungsstufe. Die Beobachtung einer Sequenz ist charakterisiert durch einen festgelegten Beginn und ein entsprechendes Ende des Beobachtungspunktes. Direkt an das Ende einer vorherigen Sequenz schließt die neue Sequenz an, sodass die Daten lückenlos aggregiert werden. Wird eine feinere Diskretisierungsstufe gewählt, entstehen kürzere Sequenzen und schlussendlich auch eine größere Anzahl an Merkmalsvektoren bei gleichbleibenden Daten. Umgekehrt werden die Eingangsdaten durch weniger Merkmalsvektoren repräsentiert, wenn die Diskretisierungsstufen größer werden. Je nach

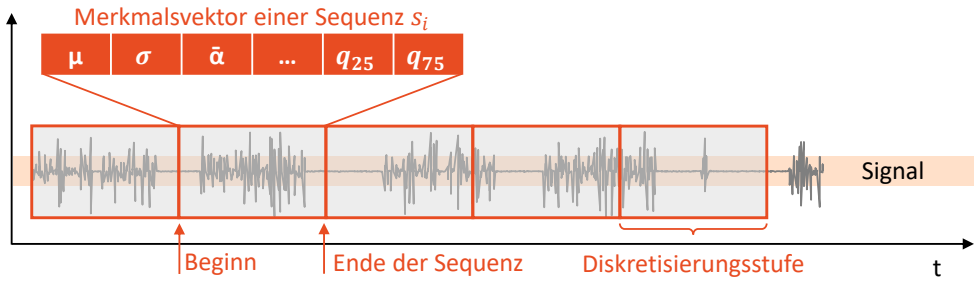


Abbildung 4.6: Darstellung zur Bestimmung von Merkmalsvektoren von Zeitreihen bei einer gegebenen Diskretisierungsstufe, in Anlehnung an [SEI19]

Fortschrittsgeschwindigkeit der Abnutzungserscheinung ist eine geringe oder hohe Diskretisierungsstufe vorstellbar. Bei einer langsamen Fortschrittsgeschwindigkeit der Alterung von beispielsweise Wochen oder Monaten ist eine besonders hohe Diskretisierungsstufe zu wählen. Wie bereits im Rahmen der Problembeschreibung in Kapitel 3.4.2 vorgestellt, ist es trotzdem das Ziel eine präzise und zeitnahe Alterungsvorhersage zu ermöglichen, obwohl die untersuchte Alterung nur über einen sehr großen Zeitraum beobachtet wurde. Aus diesem Grund werden unterschiedliche Diskretisierungsstufen zu Aggregation der Daten verwendet, um eine Alterung vorherzusagen. Die Ergebnisse der unterschiedlichen Diskretisierungsstufen werden im Rahmen der Hyperparameteroptimierung (vgl. Kapitel 4.6) in dieser Arbeit bewertet und ausgewählt. Mit dieser Auswertung kann die Wahl der Diskretisierungsstufe hinsichtlich der untersuchten Abnutzungserscheinung und der vorliegenden Daten optimiert werden.
Die zu untersuchende Abnutzungserscheinung zeigt messbare Änderungen erst nach Wochen oder Monaten. Aus den genannten Gründen werden die Diskretisierungsstufen auf einen Bereich von *10 Minuten* bis *150 Stunden* festgelegt.

4.3.2 Wahl der Merkmale

Eine gegebene Sequenz an Messinformationen kann durch unterschiedliche Merkmale charakterisiert werden. Im Rahmen dieser Arbeit werden zur Charakterisierung der metrischen Messsignale vor allem Merkmale der beschreibenden Statistik verwendet, so können diese Kenngrößen in die folgenden drei Kategorien eingeteilt werden (vgl. Cramer und Kamps in [CK20]). Zunächst beschreiben *Lagemaße* die Lage der einzelnen Daten im gesamten Wertebereich der betrachteten Daten. Hierzu zählt der Median, der Mittelwert und die Quantil-Werte. *Streuungsmaße* geben an, wie die betrachteten Daten um die Lageparameter streuen, d.h. wie gut die Daten durch die Lageparameter repräsentiert werden. Als Beispiel für

ein Streuungsmaß seien die Standardabweichung und der Interquartilsabstand genannt. Zuletzt charakterisieren *Formmaße* die Form der gegebenen Verteilung, an dieser Stelle sei die *Schiefe* genannt.
Neben den Merkmalen der beschreibenden Statistik werden weitere Größen ergänzt. Bislang wurden die Eingangssignale unverändert betrachtet. Wird dagegen aus dem Eingangssignal ein weiteres Signal berechnet, so wird ein *transformiertes Signal* (oder auch virtuelles Signal) erzeugt. Mit Hilfe eines solchen transformierten Signals kann die mittlere Steigung eines jeden Signals (engl. *slope*) innerhalb einer Sequenz berechnet werden. Darüber hinaus wurden in der vorgestellten Literatur auch Kenngrößen des Frequenzbereichs benutzt [MMG18]. Die zugrundeliegenden Daten für diese Kenngrößen werden von hochfrequenten Sensoren erfasst. Im vorliegenden Anwendungsfall werden die Daten nicht direkt vom Sensor abgenommen, sondern über ein Bussystem verschickt und im Anschluss mit einem Datenaufzeichnungsgerät gespeichert. Es ist mit unbestimmten Verzögerungen auf Grund der nicht-deterministischen und priorisierten Kommunikation des Bussystems zu rechnen. Somit ist eine unmittelbare Aufzeichnung der Sensordaten nicht möglich. Die Konfiguration des Datenaufzeichnungsgeräts legt einen fest definierten Zeitpunkt zum Prozessieren der Informationen fest. Im Rahmen der Signalanalyse im Frequenzbereich können nun Leck- und Aliaseffekte auftreten, die das zu analysierende Signal und dessen Kenngrößen im Frequenzbereich beeinflussen [Ger19, S. 259 f.]. Aus diesem Grund werden im Rahmen dieser Arbeit Kenngrößen des Frequenzbereichs nicht weiter betrachtet.
Zusammenfassend werden die folgenden Merkmale der beschreibenden Statistik und Signaltransformation benutzt:

- *Lagemaße* wie Median, der Mittel-, Minimum-, Maximum- und die Quantil-Werte;
- *Streuungsmaße* wie Standardabweichung und Interquartilsabstand;
- *Formmaße* wie die Schiefe;
- *Signaltransformierte* wie die mittlere Steigung.

In den vorgestellten Arbeiten in Kapitel 3.3 wurden auch Merkmale gebildet, um die Zeitreihen für eine weitere Verarbeitung zu charakterisieren. Dabei kommt es nicht selten vor, dass nur eine Teilmenge der genannten Merkmale genutzt wird. Die nachfolgende Tabelle 4.2 stellt die Merkmale hinsichtlich ihrer Verwendung in der Literatur zusammenfassend dar.
Aus den vorliegenden Eingangsdaten sollen relevante Muster und Regeln unter Zuhilfenahme von Merkmalen extrahiert werden. Das daraus generierte allgemeine Wissen ist abhängig von der Wahl der Merkmale. Intuitiv betrachtet könnte daraus abgeleitet werden, dass mehr Merkmale zu mehr Wissen führen würden. Dennoch kann es im Rahmen einer datengetriebenen Modellierung notwendig sein, einzelne Merkmale in den Informationen zu vernachlässigen. Das Vorhandensein von redundanten oder sehr ähnlichen Informationen (erklärenden Variablen) in der Datenmenge wird als *Multikollinearität* bezeichnet (vgl. Kapitel 2.1.3) [Sch12, S. 451 f.]. Bei Multikollinearität können die aus dem Modell erzeugten Prognosen ungenau sein.
Ein datengetriebenes Modell versucht Einflussfaktoren zu bestimmen, die die Zielgröße geeignet darstellen sollen. Diese Einflussfaktoren stehen aber nicht zwangsläufig in einer

Tabelle 4.2: Übersicht der verwendeten Merkmale zur Beschreibung von Sequenzen

Zuordnung	Merkmalsbezeichnung	Anwendung in
Lagemaße	Mean (Mittelwert), Median, Q_25 (1. Quartil), Q_75 (3. Quartil), Min (minimaler Wert), Max (maximaler Wert)	[ZZX16; Car+18; MMG18; Fug+19; Cro+03]
Streuungsmaße	Std (Standardabweichung), IQR (Interquartilsabstand)	[Cro+03; MMG18; Fug+19]
Formmaße	Skew (Quartilskoeffizient der Schiefe)	[Hui+17; ZZX16; Car+18; MMG18]
Signaltransformierte	Slope (Mittelwert der Steigung)	[MMG18]

kausalen Beziehung zur Zielgröße. Dieser Effekt wird auch als *Scheinkorrelation* beschrieben [Bac+18, S. 60; HK17, S. 252]. Eine hohe Korrelation beschreibt zwar eine statistische Beziehung zur Zielgröße, aber nicht deren Kausalität. Auf Grund von Scheinkorrelationen kann somit eine Kausalität nicht zweifelsfrei festgestellt werden. Je mehr Größen durch das Modell betrachtet werden, desto unwahrscheinlicher ist die Erkennung relevanter Einflussfaktoren. Außerdem ist das Fehlen relevanter Einflussgrößen zur Zielgrößenbestimmung denkbar.
Das Konzept sieht vor, dass Merkmale signal-individuell definiert werden. Aus diesem Grund werden im Rahmen der Hyperparameteroptimierung die Auswahl der Merkmale für jedes Signal individuell festgelegt, um so eine effiziente Lösung der Alterungsvorhersage zu erhalten.

Vergleichbarkeit durch Standardisierung Damit eine Vergleichbarkeit zwischen Merkmalen unterschiedlicher Signale gewährleistet ist, wird eine Standardisierung der Daten vorgenommen. Dadurch entfallen einheitenspezifische Ausprägungen und Signale lassen sich untereinander vergleichen [Bac+18, S. 73-74, S. 373-374; BCK12, S. 104-105; MS05, S. 45].

4.4 Data-Mining

In diesem Abschnitt sollen aus den Daten relevante Muster und Regeln extrahiert werden. Dieses allgemeine Wissen ist nach Ester und Sander statistisch gültig, bisher unbekannt und für die gegebene Anwendung potentiell nützlich [ES00a]. Zunächst werden die Daten der Beobachtungspunkte den einzeln gemessenen Performance-Informationen zugeordnet. Dabei besteht ein Beobachtungspunkt aus dem Merkmalsvektor einer Sequenz (vgl. dazu Kapitel 4.3 und in Abbildung 4.6). Ein Beobachtungspunkt besteht aus mehreren Variablen. In der Abbildung 4.7 wird die unterschiedliche Ausprägung der einzelnen Variablen

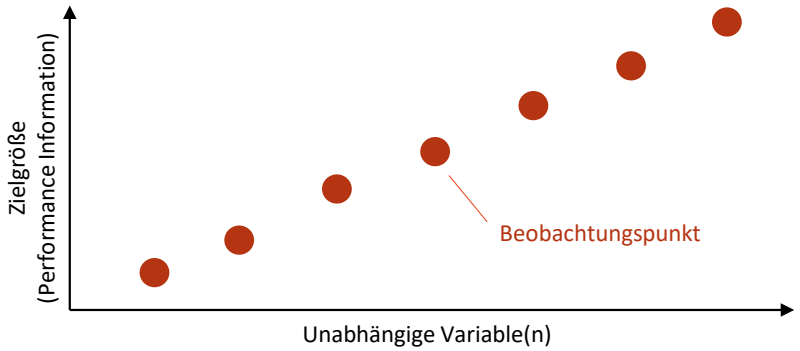


Abbildung 4.7: Zuordnung der Beobachtungspunkte zu einzelnen Performance-Werten

eines Beobachtungspunktes eindimensional dargestellt und der gemessenen Zielgröße (Performance-Informationen) zugeordnet. Der Merkmalsvektor des Beobachtungspunktes wird im Rahmen der Regressionsanalyse auch als Einflussgröße oder unabhängige Variable bezeichnet. Die Begriffe „unabhängig“ und „abhängig“ stellen dabei allerdings keine Kausalbeziehung dar, sondern nur eine vom Untersucher vermutete statistische Abhängigkeit [Bac+18, S. 58 f.].
Um aus diesen Beobachtungspunkten allgemeines Wissen abzuleiten, werden Methoden der Regressionsanalyse verwendet. Dazu wird die Trainingsdatenmenge genutzt, um ein datengetriebenes Modell zu erstellen. Ziel ist es unter Zuhilfenahme des angelernten Wissens (Modell) eine Vorhersage eines neuen Performance-Wertes für unbekannte Daten zu gewährleisten. Das zu erstellende Modell wird mit der Trainingsdatenmenge angelernt. Nur

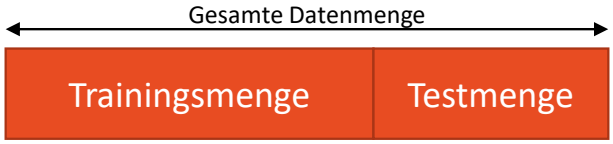


Abbildung 4.8: Schematische Darstellung des Prinzips der Aufteilung der gesamten Datenmenge in Trainings- und Testdaten

mit Hilfe der gemessenen Performance-Informationen (wahrer Wert der Abnutzung) kann das Modell validiert werden. Um dennoch eine Aussage bezüglich der Effizienz des erstellten Modells für unbekannte Daten treffen zu können ohne neue Performance-Werte zu erheben, wird die Datenmenge in Trainings- und Testmenge aufgeteilt. Die Testdatenmenge

wird während des Lernprozesses nicht betrachtet und kann für Vorhersagen und Qualitätsbeurteilungen genutzt werden. Die Abbildung 4.8 stellt die generelle Einteilung in Trainings- und Testdatenmenge beispielhaft dar.
Die Trainingsmenge, die zum Lernen des Modells benutzt wird, besteht aus dem Merkmalsvektor einer Beobachtung und dem zugehörigen Performance-Wert p. Dabei handelt es sich um eine messbare Größe. Dieser Performance-Wert wird im weiteren Verlauf der Arbeit durch einen virtuellen Sensor ersetzt. Die Abbildung 4.9 stellt die erzeugte Trai-

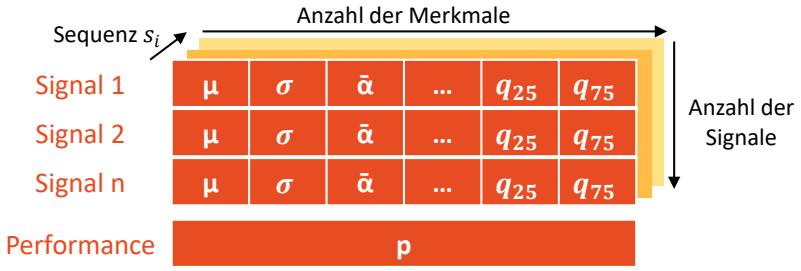


Abbildung 4.9: Darstellung der Merkmalsmatrix eines Beobachtungspunktes als Eingangsdatenmenge des zu lernenden Modells

ningsdatenmenge schematisch dar. Für die einzelnen Signale 1 bis n werden in der jeweiligen Sequenz s_i Merkmale berechnet. Weiterhin wird dieser Matrix der in diesem Zeitraum gemessene Performance-Wert zugeordnet.
Wie bereits in Kapitel 2.1.3 vorgestellt, können verschiedene Methoden des überwachten Lernens für eine Regression verwendet werden. Neben der Vorhersagegüte eines angewendeten Lernalgorithmus ist auch die Komplexität und Einsetzbarkeit der Lernalgorithmen zu betrachten. Außerdem werden weichere Faktoren wie die Interpretierbarkeit der Ergebnisse im weiteren Verlauf in ein gesamtheitliches Interpretationsbild mit einbezogen. In der Tabelle 4.3 wird die aus dem Stand der Wissenschaft (vgl. Kapitel 3.3) vorgestellte Literatur hinsichtlich der verwendeten Methoden dargestellt. Viele der vorgestellten Arbeiten ver-

Tabelle 4.3: Anwendung von Methoden des überwachten Lernens in der vorgestellten Literatur

ML-Methode des überwachten Lernens	Literatur
statistische Techniken (wie z.B. MLR)	[PNB11; Sha+18]
Support Vektor Maschinen (SVMs)	[Liu+17; PNB11; ZZX16; Hui+17; Car+16]
Lernen von Bayes-Netzen	[SSM10; Sha+18], ggf. [WGL14]
Entscheidungsbäume (RF)	[PNB11; Sha+18]
k-Nearest Neighbor-Methoden (kNN)	[Sha+18], ggf. [Fil+10]
neuronale Netze (NN)	[Guo+00; MMG18; CCP12; Fug+19]

wenden mehrere Methoden des überwachten Lernens (z.B. [PNB11]). Ob eine bestimmte Methode des überwachten Lernens für den jeweiligen Anwendungsfall geeignet ist, kann an

unterschiedlichen Ursachen liegen. Dazu zählt u.a. die Menge der zur Verfügung stehenden Lerndaten, die entsprechende Parametrisierung eines Algorithmus sowie die Beschaffenheit der Lerndaten.
In dieser Arbeit sollen mit Hilfe der vorgestellten Methoden Alterungsvorhersagen bestimmt werden. Im weiteren Verlauf dieser Arbeit werden diese Methoden genutzt und miteinander verglichen. Jeder einzelne Algorithmus hat dabei unterschiedliche Hyperparameter, die die Ergebnisse der Vorhersage beeinflussen können. Die einzelnen Parameter werden im Rahmen der Konzeptvalidierung (vgl. Kapitel 5) näher beschrieben und hinsichtlich geeigneter Parameter optimiert.

4.5 Interpretation und Bewertung

Der Abschnitt *Interpretation und Bewertung* beschreibt den letzten Schritt des zu Beginn des Kapitels vorgestellten KDD-Prozesses, bzw. den Bewertungsschritt des CRISP-DM-Prozesses. Die Ergebnisse aus dem vorherigen Abschnitt *Data-Mining* (vgl. Kapitel 4.4) werden in diesem Abschnitt visualisiert und bewertet. Unterschiedliche Maße zur Bewertung der Vorhersagegüte wurden bereits in Kapitel 2.1.4 vorgestellt und werden in diesem Abschnitt ausgewählt. Wie die Abbildung 4.10 skizziert, kann für jeden beobachteten Punkt ein prädizierter (vorhergesagter) Performance-Wert bestimmt werden. Typischerweise wird

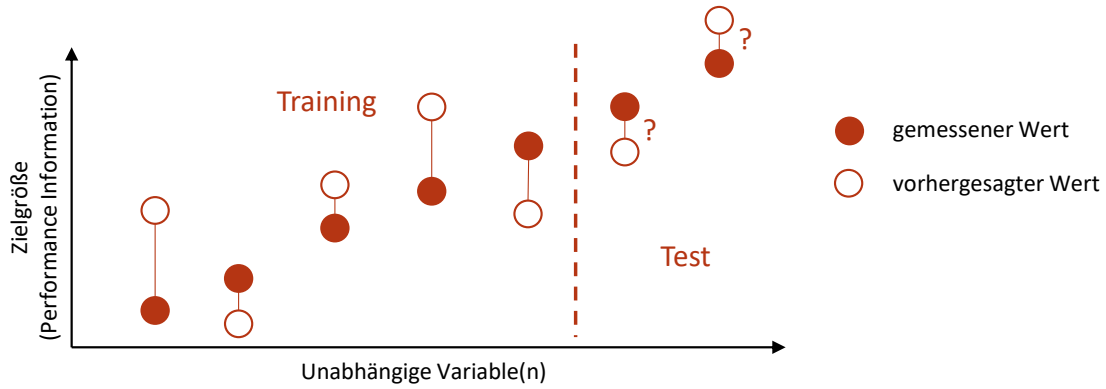


Abbildung 4.10: Darstellung und Vergleich zwischen gemessenen und vorhergesagten Performance-Werten im Rahmen der Regressionsanalyse

die Trainingsdatenmenge zum Training des Modells und die Testdatenmenge zur Bewertung des Modells genutzt.
Der vorhergesagte Performance-Wert kann nun mit einer geeigneten Metrik (engl. *scores*) und durch Zuhilfenahme des wahren (gemessenen) Wertes bewertet werden. Dabei galt vor allem der RMSE als geeignet, sofern die zu bewertenden Werte frei von Ausreißern sind (vgl. Kapitel 2.1.4). Der von einem geeigneten ML-Modell prädizierte Wert sollte zwangsläufig eine große Ähnlichkeit mit dem gemessenen Wert haben, andernfalls ist davon auszugehen, dass das angelernte Modell nicht geeignet ist. Aus diesem Grund sind keine größeren Ausreißer zu erwarten. Dennoch gilt zu beachten, dass der RMSE eine einheitenbehaftete Metrik ist.

Im Rahmen der Aufteilung in Trainings- und Testdatensätze kann die Generalisierungsfähigkeit überprüft werden. Die vorliegenden Eingangsdaten stammen von unterschiedlichen Fahrzeugen und können deshalb auf folgende zwei Arten hinsichtlich der Trainings- und Testmenge aufgeteilt werden:

- **Sequenzbasierte Aufteilung**: Sequenzen aller Fahrzeuge werden in einer Datenmenge vereint und daraus wird eine zufällige Aufteilung der einzelnen Sequenzen vorgenommen.
- **Fahrzeugbasierte Aufteilung**: Alle Sequenzen bleiben den einzelnen Fahrzeugen zugeordnet. Die gesamten Daten von bestimmten Fahrzeugen werden der Trainings- oder der Testdatenmenge zugeteilt.

Die zufällige *sequenzbasierte Aufteilung* ermöglicht ein hohes Maß an Objektivität, da keine Datensätze a-priori ausgewählt werden. Allerdings kann auf Grund der zufälligen Wahl der Sequenzen nicht gewährleistet werden, ob diese eine breite Masse an Fahrprofilen und Dynamik abdecken, wie sie für eine vollständige Eingangsdatengrundlage des Trainings bereitgestellt werden sollte. Werden die zufällig gewählten Sequenzen ungeschickt aufgeteilt, können so beispielsweise besonders viele gleiche Fahrprofile in der Trainingsdatenmenge enthalten sein. Daraus entstünde ein virtuelles Sensormodell, welches sehr an die Trainingsdatenmenge angepasst wäre. Um eine solche Überanpassung an die Trainingsdatenmenge auszuschließen, wird eine *fahrzeugbasierte Aufteilung* bevorzugt. Es wird davon ausgegangen, dass alle Fahrzeuge im Laufe der Nutzung dynamisch unterschiedlich bewegt worden sind. Dabei ist darauf zu achten, dass die Kombination aus Test- und Trainingsfahrzeugen nicht statisch ist. Für eine solche Aufteilung stehen Validierungsverfahren des Data-Minings zur Verfügung und werden im folgenden Abschnitt erläutert.
Die *fahrzeugbasierte Datenaufteilung* wird nun um ein Validierungsverfahren des Data-Minings erweitert. Bei dem *Kreuzvalidierungsverfahren* werden Trainings- und Testdaten nach einem bestimmten Verfahren aufgeteilt [Alp10, S. 486-488]. Üblicherweise geschieht diese Aufteilung mehrfach und die unterschiedlichen Ergebnisse jeden Durchlaufs (auch Experiment genannt) werden im Anschluss gemittelt. Bei einer k-fachen Kreuzvalidierung (engl. *k-fold cross-validation*) werden die Daten in k, meist gleichgroße, Unterteilmengen aufgeteilt.
Dieses Validierungsverfahren wird auch in der vorgestellten Literatur angewendet und bietet mehrere Vorteile [Fug+19; Car+18; Liu+17; Sha+18]. Mit Hilfe dieses Vorgehens kann eine Modellüberanpassung vermieden und ein deterministisches Verhalten ermöglicht werden. Eine Überanpassung liegt dann vor, wenn das Modell bei neuen Testdaten eine deutlich schlechtere Performance liefert als bei bereits gelernten Daten [SL19, S. 119 f.]. Dies kann zum Beispiel an einer zu geringen Trainingsdatenmenge liegen. In diesem Fall erkennt das Modell nicht die wahren Zusammenhänge in den Daten, sondern lernt diese „auswendig“.
In herkömmlichen Ansätzen werden Trainings- und Testdaten randomisiert erstellt. Dies kann zur Folge haben, dass zwischen einzelnen Durchläufen große Unterschiede in der Modellqualität im Testdurchlauf entstehen. Ein zufällig ausgewählter Trainingsdatensatz kann für den zu lernenden Zusammenhang nicht repräsentativ sein und eine verzerrte Abbildung erzeugen (Bias). Generell ist ein deterministisches Verhalten bei zufällig ausgewählten

Samples nicht gewährleistet [WFH11, S. 152 - 154].
Übergeordnet wird die Eingangsdatenmenge zunächst in Daten zum Finden der Hyperparameter und Evaluierungsdaten aufgeteilt (engl. *leave-one-out*) [Alp10, S. 486-488; Won15]. In einem weiteren Schritt wird nun die Kreuzvalidierung angewendet, um geeignete Hyperparameter festzulegen. Während der Kreuzvalidierung wird die Menge der Trainingsda-

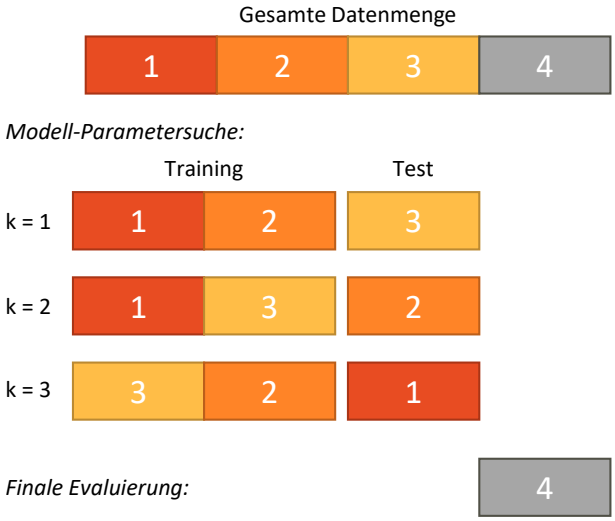


Abbildung 4.11: Schematische Darstellung der Aufteilung von Trainings- und Testdaten nach dem k-fachen Kreuzvalidierungsverfahren mit $k = 3$

ten in k Teilmengen aufgeteilt. Weiterhin werden $k-1$ Teilmengen für das Training innerhalb einer Iteration angewendet und die restliche Teilmenge an Daten wird zum Testen der Hyperparameter benutzt. Dieses Vorgehen wird k-fach wiederholt, bis alle Kombinationsmöglichkeiten ausprobiert sind (vgl. Abbildung 4.11). Für jeden einzelnen Durchlauf wird der Modellfehler berechnet. Der Gesamtfehler berechnet sich aus dem Mittel der einzelnen Prognosefehler. Mit Hilfe der Kreuzvalidierung kann eine ungeschickte Aufteilung an Trainings- und Testdaten vermieden werden. Weiterhin erlaubt dieses Vorgehen ein deterministisches Verhalten, auch wenn ein höherer Rechenaufwand entsteht. Das Ergebnis dieser Kreuzvalidierung liefert eine geeignete Konfiguration an Hyperparametern.
Für eine finale Evaluierung der gewählten Hyperparameter wird ein Modell aus allen zur Verfügung gestellten Trainingsdaten erstellt. Dieses Modell kann dann im Anschluss mit den bis dahin nicht betrachteten Daten final evaluiert werden.

4.6 Hyperparameteroptimierung

Das übergeordnete Konzept aus dem Abschnitt Konzeptvorstellung (vgl. Kapitel 4.1) sieht eine Rückkopplung vor, sodass die ausgewählten Hyperparameter hinsichtlich der Modellgüte anpassbar sind. Diese Hyperparameteroptimierung wird im folgenden Abschnitt konzeptionell vorgestellt.
Zur Lösung von Hyperparameteroptimierungsproblemen wird als Optimierungsverfahren die Bayessche Optimierung angewendet (vgl. Kapitel 3.7 und 3.7.3). Diese zeigt gegenüber

Zufallssuche und Rastersuche entscheidende Vorteile, da die einzustellenden Hyperparameter entsprechend einem probabilistischen Modell verändert werden. Dieses probabilistische Modell wird kontinuierlich aktualisiert. Dabei werden alle Informationen aus vorherigen gewählten Konfigurationen genutzt, um die nächste Konfiguration an Hyperparametern zu bestimmen. Zeitgleich wurde in der Literatur gezeigt, dass dieses Verfahren sogar schneller Ergebnisse erzielt als bei der Zufallssuche und Rastersuche [SLA12; Ber+11].
Zur Umsetzung der bayesschen Optimierung werden in der Literatur unterschiedliche Methoden vorgestellt [Egg+15]. Im weiteren Verlauf des Abschnittes werden folgende Implementierungen der bayesschen Optimierung miteinander verglichen: Spearmint, Sequential Model-based Algorithm Configuration (SMAC) und Tree-Structured Parzen Estimator (TPE). Die Spearmint-Methode bildet dabei eine unterliegende Wahrscheinlichkeitsverteilung mit Hilfe des gaußschen Prozesses ab [SLA12]. Die SMAC-Methode nutzt dafür eine Random Forest Struktur [HHL11]. Bei der TPE-Methode werden baumstrukturierte Kerndichteschätzer verwendet [Ber+11]. Die Tabelle 4.4 stellt die vorgestellten Implementierungen zur

Tabelle 4.4: Auflistung unterschiedlicher Methoden zur bayesschen Optimierung

Methode	**Erstveröffentlichung**	**Zitationen**[3]	**Charakteristik**
Spearmint	2012, von den Autoren Snoek, Larochelle und Adams [SLA12]	4319	Bayessche Optimierung / gaußscher Prozess
SMAC	2011, von den Autoren Hutter, Hoos und Leyton-Brown [HHL11]	1685	Bayessche Optimierung / Random Forest
TPE	2011, von den Autoren Bergstra u. a. [Ber+11]	2265	Bayessche Optimierung / baumstrukturierter Kerndichteschätzer

bayesschen Optimierung hinsichtlich der Erstveröffentlichung, der Anzahl der Zitationen und der Implementierungscharakteristik gegenüber.
Die vorgestellten Methoden sind in der Literatur vielfach zitiert (vgl. Tabelle 4.4). Zahlreiche Arbeiten in der Literatur vergleichen die Evaluationsergebnisse von Spearmint, SMAC und TPE, wie beispielsweise [Egg+13; Sno+14; Ili+17]. Die vorgestellten Methoden zeigen je nach Anwendungsfall unterschiedliche Vor- und Nachteile. So zeigt die Arbeit von Madrigal, Maurice und Lerasle, dass die Methoden von SMAC und TPE in den untersuchten Anwendungsfällen bessere Ergebnisse liefern als Spearmint [MML19]. Auch die Laufzeiten bei größeren Iterationen sind bei SMAC und TPE im Vergleich zu Spearmint vergleichsweise gering. Die TPE-Methode zeigt eine leicht bessere Robustheit gegenüber den anderen Methoden [MML19]. Die Autoren Wendt, Wuschnig und Lechner gehen darüber hinaus und bezeichnen die TPE-Methode als die beste Option für deren bayessches Optimierungsproblem auf Grund der schnellen und guten Ergebnisse in [WWL20].

3 Als Quelle für die Anzahl der Zitationen wurde die öffentliche Zitationsdatenbank von Google Scholar `https://scholar.google.de` verwendet. Das Abrufdatum war am 04.02.2021.

Eine unter Anwendung der bayesschen Optimierung gefundenen Lösung lässt dabei keine Rückschlüsse auf die generelle Anwendbarkeit der ML-Methode zu. Es können lediglich die Güten unterschiedlicher Lernalgorithmen und die Hyperparameter der Vorverarbeitung bezogen auf den in dieser Arbeit vorliegenden Anwendungsfall miteinander verglichen werden. Eine Verallgemeinerung und Rückschlüsse auf die Gesamtheit der Probleme sind nicht möglich (vgl. Vorstellung des „No-free-lunch"-Theorems in Kapitel 3.7.3). Die gefundenen Hyperparameter werden mit Hilfe eines Gütemaßes unter Anwendung des Kreuzvalidierungsverfahrens miteinander verglichen. Dieses Gütemaß ist nur ein Kriterium von vielen, die die Wahl einer Methode beeinflussen können (andere Kriterien sind zum Beispiel Betrachtung unterschiedlicher Laufzeiten oder Einfachheit der Methodenimplementierung [Alp10, S. 477 f.]).
Die Tabelle 4.5 zeigt den gesamten Suchraum an Hyperparametern für eine datengetriebene Vorhersage einer langfristig verändernden Zielgröße. Mit jedem hinzugefügten Signal erweitert sich auch der Suchraum, da für jedes Signal individuelle Merkmale festgelegt werden können. Ein Teil der dargestellten Hyperparameter wird dabei im Rahmen der Datenaufnahme und -selektion (vgl. Kapitel 4.2) festgelegt, ein weiterer Teil der Parameter stammt aus dem Abschnitt des Data-Minings (vgl. Kapitel 4.4). Die restlichen Hyperparameter werden im Rahmen der Konzeptvalidierung in Kapitel 5 vorgestellt.

Einstellparameter der Datenaufnahme und -selektion Es wurde gezeigt, dass sich Methoden des Maschinellen Lernens effektiver trainieren lassen, wenn die Daten bereits bereinigt, transformiert und ggf. reduziert werden. Im Folgenden werden drei Hyperparameter der Vorverarbeitung vorgestellt. Der erste Parameter beschreibt die anzuwendende Teilmenge an Signalen. Zunächst wurden verwechselbare Signale in den Daten durch das Clustering eliminiert. Die resultierende Menge an einzigartigen Signalen kann nun genutzt werden, um ein Modell zu erstellen. Darüber hinaus kann auch eine Teilmenge an Signalen geeignet sein, das zu erstellende Sensormodell effizienter zu erstellen. Das heißt, der erste Hyperparameter beschreibt die Wahl der *Teilmenge an Signalen*, die zur Erstellung des Modells genutzt werden soll. Weiterhin wurden unterschiedliche Diskretisierungsstufen der Sequenzen vorgestellt. Der zweite Hyperparameter beschreibt die Wahl der *Diskretisierungsstufe*. Bei einer zu feinen Diskretisierungsstufe lassen sich nicht mehr langfristige Alterungsprozesse abbilden, stattdessen werden nur kurzfristige fahrzeugdynamische Ereignisse in den Merkmalen abgebildet. Wohingegen eine zu grobe Diskretisierungsstufe zu wenige Vorhersagemöglichkeiten bietet und die Aussagekraft dieser Vorhersage abnehmen wird. Das letzte Parameter beschreibt die *Wahl der Merkmale*. Die im Abschnitt 4.3.2 vorgestellte Liste beinhaltet alle Merkmale, die im Rahmen dieser Arbeit angewendet werden. Dennoch kann die Liste der Merkmale reduziert werden, falls einzelne Merkmalsinformationen bereits durch zuvor ausgewählte Merkmale bereitgestellt worden sind und damit eine Multikollinearität vermieden werden kann.

Einstellparameter des Data-Minings Neben den Parametern der Vorverarbeitung werden auch die Methoden (oder auch *Regressoren*) des Maschinellen Lernens als Hyperparameter festgelegt. Dazu wurden bereits die in der Literatur verwendeten Methoden tabellarisch dargestellt (vgl. Kapitel 4.4). Die zu verwendende Methode kann im Hyperparameter *Auswahl einer ML-Methode* festgelegt werden. Zu jeder dieser Methoden gehören auch spezifische Einstellparameter, die neben der eigentlichen Regressorauswahl auch zu bestimmen sind und im Hyperparameter *Einstellparameter der ML-Methode* festgelegt werden. Eine detailliertere Vorstellung der spezifischen Einstellparameter der ML-Methoden wird im Rahmen der Konzeptvalidierung in Kapitel 5.3.2 vorgestellt.

Tabelle 4.5: Vorstellung der Hyperparameter und deren Ausprägungen für die datengetriebene Modellerstellung einer Abnutzungserscheinung, getrennt nach Hyperparametern der Vorverarbeitung und des Data-Minings

Hyperparameter der Vorverarbeitung	**Ausprägung / Eingrenzung**
Auswahl einer Teilmenge an Signalen s	$s \subseteq$ {Signal 1, Signal 2, ..., Signal n}
Diskretisierungsstufe d	$d \in$ {1 Stunde, 2 Stunden, ..., m Stunden}
Auswahl einer signalspezifischen Teilmenge an Merkmalen o	$o \subseteq$ {Mean, Median, Q_25, Q_75, Min, Max, Std, IQR, Skew, Slope}
Hyperparameter des Data-Minings	**Ausprägung / Eingrenzung**
Auswahl einer ML-Methode α	$\alpha \in$ {MLR, SVR, Bayes, RF, kNN, NN}
Einstellparameter der ML-Methode	*Einstellparameter sind entsprechend der ML-Methode auszuwählen, weitere Informationen in Kapitel 5.3.2*

4.7 Konzeptzusammenfassung

In diesem Abschnitt wird das Konzept für die Entwicklung eines virtuellen Sensormodells zur Bestimmung der Abnutzungserscheinung zusammengefasst. In dem Kapitel 4.2 wird beschrieben, wie die Daten für weitere Analysen aufgenommen und synchronisiert werden. Zur Dimensionsreduktion der Eingangsdaten werden Signalduplikate oder verwechselbare Signale vor der Modellerstellung entfernt, da sie keine zusätzlichen Informationen liefern. Folgende Arbeiten aus dem vorgestellten Stand der Wissenschaft und Technik verwenden zur Reduktion von Dimensionen eine PCA: [CCP12; Fug+19; MMG18]. Die unter Anwendung einer PCA erstellten Daten sind nur schwer interpretierbar. Aus diesem Grund wird ein Clusterverfahren, das die Vielzahl verwechselbarer Signale reduziert, angewendet. Dieses Vorgehen kann nicht nur die zu analysierende Datenmenge reduzieren, sondern ermöglicht zeitgleich eine Nachvollziehbarkeit der genutzten Signale. Desweiteren können die gefundenen Cluster und deren Repräsentanten auch für weitere Analysen verwendet werden. Eine erneute Berechnung einzigartiger Signale ist erst bei einer signifikanten Veränderung neuer

Eingangsdaten erforderlich, z.B. wenn neue Signale den Eingangsdaten hinzugefügt werden.

Anschließend wurde vorgestellt, wie Merkmale aus hochaufgelösten Signalen innerhalb von Sequenzen festgelegter Diskretisierungsstufen für Alterungsvorhersagen verwendbar sind (vgl. Kapitel 4.3). Die Wahl der Diskretisierungsstufe steht im Zielkonflikt mit einer präzisen und zeitnahen Alterungsvorhersage. In der vorgestellten Literatur wurden Fenstergrößen von wenigen Sekunden [Car+18; ZZX16], Minuten [Fug+19] und die Aggregation von Daten innerhalb eines Tages [Liu+17] diskutiert.
Die Wahl der Merkmalsmenge ist eng mit der Wahl der Diskretisierungsstufe verknüpft. Die einzelnen Merkmalswerte werden aus den Daten der Sequenzen einer Diskretisierungsstufe berechnet (vgl. Kapitel 4.3.2). In der Tabelle 4.2 sind dazu Merkmale aus folgenden Kategorien aufgelistet: Lagemaße, Streuungsmaße, Formmaße und Signaltransformierte, vgl. dazu folgende Arbeiten: [ZZX16; Car+18; MMG18; Fug+19; Cro+03; Hui+17; Car+18].

Neben den Parametern der Vorverarbeitung werden im Rahmen des Konzeptentwurfs auch die Parameter des Data-Minings variiert (vgl. Kapitel 4.4). Die verwendeten Lernalgorithmen des MLs sind von der vorgestellten Literatur abgeleitet und werden in der Tabelle 4.3 zusammengefasst. Im weiteren Verlauf dieser Arbeit sind zu den jeweiligen Lernalgorithmen weitere Einstellparameter dargestellt worden (vgl. Kapitel 5.3). In der betrachteten Literatur werden die vorgestellten Lernalgorithmen in unterschiedlichen Bereichen der Klassifikation (z.B. [PNB11; Sha+18]) und der Regression (z.B. [Zhe+18]) angewendet. Die Arbeiten zeigen, dass sich in den dargestellten Anwendungsfällen SVM, kNN und RF als geeignet erweisen [Sha+18]. In ausgewählten Anwendungsfällen ist der RF dem SVM überlegen [PNB11; Zhe+18].

Zuletzt wird eine Interpretation und Bewertung der Ergebnisse beschrieben (vgl. Kapitel 4.5). Das Konzept sieht eine Aufteilung von Trainings- und Testdaten nach dem Kreuzvalidierungsverfahren zur Vermeidung einer Überanpassung an die Trainingsdatenmenge vor.

Damit eine Optimierung in akzeptabler Zeit durchgeführt werden kann, werden einzelne Hyperparameter bereits im Vorhinein systematisch eingegrenzt. Diese systematische Eingrenzung ist nicht mit allen Hyperparametern möglich, da beispielsweise die Vorhersage der Zielgröße abhängig von der verwendeten ML-Methode ist. Aus diesem Grund sind in Kapitel 4.6 die einstellbaren Hyperparameter vorgestellt worden. Im Rahmen einer bayesschen Optimierung wird eine geeignete Konfiguration aus den vorgestellten Hyperparametern für das virtuelle Sensormodell gesucht, die die Alterungsvorhersage geeignet modellieren kann. Im nächsten Kapitel 5 wird die Umsetzung des Konzepts auf ein konkretes Beispiel validiert und die einzelnen Hyperparameter hinsichtlich einer geeigneten Vorhersagequalität optimiert.

5 Konzeptvalidierung am Beispiel einer Abgasrückführung-Kühlerversottung

In diesem Kapitel wird das zuvor beschriebene Konzept anhand eines Beispiels aus der Automobilbranche validiert. Zu Beginn werden allgemeine Rahmenbedingungen und Vorgehensweisen der Implementierung beschrieben (vgl. Abschnitt 5.1). Es wird auch auf den vorliegenden Datenbestand und die Datenmenge eingegangen, sowie das verwendete Prognosegütemaß vorgestellt. Außerdem werden in einem weiteren Unterkapitel zwei Hypothesen bezüglich der Vorhersageergebnisse aufgestellt.
Mit Hilfe des in Kapitel 4 vorgestellten Konzepts soll in diesem Kapitel die Vorhersage einer langfristigen Abnutzungserscheinung eines AGR-Kühlers validiert werden. Es werden zwei unterschiedliche Ansätze vorgestellt: Ein hybrider Expertenansatz und eine datengetriebene Hyperparameteroptimierung. Die beiden Ansätze werden über unterschiedliche Ausprägungen der Einstellparameter der Vorverarbeitung und des Data-Minings charakterisiert. Im Abschnitt 5.2 wird der hybride Expertenansatz beschrieben. Dieser Ansatz verfolgt eine intuitive Lösung mit Hilfe von spezifischen Wissen aus der Fachdomäne. Dem gegenüber steht die datengetriebene Hyperparameteroptimierung (vgl. Abschnitt 5.3). Hier liegt ein hochdimensionaler Raum an Einstellparametern der Vorverarbeitung und des Data-Minings vor. Mit Hilfe einer Optimierungsstrategie werden geeignete Einstellparameter ausgewählt. In beiden Ansätzen werden Methoden des MLs verwendet und für das in dieser Arbeit skizzierte Problem angewendet. Im Anschluss werden die Ergebnisse und Einflussfaktoren der datengetriebenen Hyperparameteroptimierung erläutert (vgl. Abschnitt 5.4).

5.1 Allgemeine Rahmenbedingungen

Der zur Verfügung stehende Datenbestand beinhaltet Messdaten vom CAN-Bus unterschiedlicher Fahrzeuge. Mit Hilfe von Fahrzeugdatenloggern werden diese Informationen aufgezeichnet. Diese Daten werden zunächst in *MATLAB* eingelesen und vorverarbeitet. Dazu liegen bereits Algorithmen für *MATLAB* vor, die das Dateiformat von Fahrzeugloggern (meist *MDF*, Measurement Data Format) in ein für *MATLAB* und *Python* interpretierbares Datenformat überführen. Sämtliche weitere Methoden der Vorverarbeitung und die darauf aufbauende Algorithmen des MLs werden in *Python* umgesetzt. Für die Anwendung der ML-Algorithmen in Python stehen zahlreiche Bibliotheken öffentlich zur Verfügung. Auch in den vorgestellten wissenschaftlichen Arbeiten werden Projektteilbereiche in Python programmiert [Gar+17; MMG18]. Im Folgenden werden die zwei bekannten Bibliotheken des MLs vorgestellt:

- **Scikit-learn**
  Die Bibliothek *Scikit-learn* bietet zahlreiche Methoden des überwachten und unüberwachten MLs an. Die einzelnen Methoden weisen unterschiedliche Parameter auf, deren Ausprägungen im Quellcode bestimmt werden. Für den Einstieg stehen Dokumentationen

zu den einzelnen Methoden, den Anpassungsmöglichkeiten und weiteren Beispielen zur Verfügung.

- **TensorFlow**
  Die Bibliothek *TensorFlow* ist spezialisiert auf die Implementierung künstlicher neuronaler Netze. Die internen Variablen werden als so genannte *Tensoren* abgebildet und mathematische Operation werden durch Graphen dargestellt.
  Im Rahmen dieser Arbeit wird die Bibliothek *Keras* in der Version 2.3.1 verwendet, die den Zugang zu der Funktionalität von *TensorFlow* ohne Restriktionen an die Verwendung eines bestimmten Backends auf einem höheren Level bietet.

Im weiteren Verlauf dieser Arbeit werden unterschiedliche Laufzeitverhalten für die Algorithmen angegeben. Diese sind unter Verwendung einer Workstation mit Windows 10, Intel Xeon 6134 @ 3.2 GHz, 8 Kernen bzw. 16 Threads, 192 GB Arbeitsspeicher und Python in der Version 3.5.2 entstanden.

5.1.1 Datengrundlage

Es liegt eine Datenbasis von hochaufgelösten CAN-Bus Aufzeichnungen von mehreren Fahrzeugen vor. Die folgende Tabelle 5.1 stellt die gesamte Datenmenge dar. Die nummerierten Fahrzeuge erhalten für eine bessere Lesbarkeit eine Kurzbezeichnung. In der zweiten Spalte ist die minimale und maximale Ausprägung der Zielgröße notiert, die im Rahmen der Performance-Messung durchgeführt wurden (vgl. Kapitel 2.3.3). Weiterhin sind die Anzahl der Signale aufgeführt, die im Rahmen der zur Verfügung gestellten Informationen aus den Rohdaten entnommen werden konnten. Aufgrund unterschiedlicher Fahrzeuglogger-Konfigurationen kann sich die Anzahl der Signale von Fahrzeug zu Fahrzeug unterscheiden. Es handelt sich dabei um die Anzahl der Signale, die über den gesamten Messzeitraum für das jeweilige Fahrzeug zur Verfügung stehen. Außerdem sind noch die Laufleistung und der betrachtete Zeitraum der analysierten Daten dargestellt. Der betrachtete Zeitraum umfasst dabei die Differenz vom ersten Tag bis zum letzten Tag, an dem Messdaten vorliegen. Die letzte Spalte kennzeichnet die verfügbare Datenmenge im unveränderten Datenformat des Fahrzeugdatenloggers (hier MDF). Die Tabelle 5.1 stellt die Eigenschaften der analysierten Daten dar. Die Zeitreihe „*Fahrzeuggeschwindigkeit*“ des Fahrzeugs *209* hat nach der Vorverarbeitung (also nach Synchronisation und Bereinigung fehlerhafter Datenpunkte) eine Anzahl von 65.785.953 Samples, d.h. mehr als 65 Mio. Datenpaare aus Zeitinformation und dem zugehörigen Messwert. In *MATLAB* belegt allein dieses einzige Signal in seinem äquidistanten Raster etwa 1 GB im Hauptspeicher.
In einer weiteren Tabelle wird der Zusammenhang von Diskretisierungsstufe und Anzahl der Samples an einem Beispielfahrzeug dargestellt (vgl. Tabelle 5.2). Die Tabelle zeigt eine Auswahl an Diskretisierungsstufen d (10 min, 1 Stunde, 6 Stunden und 15 Stunden) und die zugehörige Anzahl der Samples T. Die Anzahl der Samples ergibt sich aus der Gesamtmenge an verfügbaren Daten, abzüglich möglicher Aufzeichnungslücken, Fahrzeugstandzeiten und Bereinigung möglicher fehlerbehafteter Datenpunkten. Aus diesem Grund kann

Tabelle 5.1: Übersicht der Eigenschaften der analysierten Fahrzeugdaten

Bezeichnung	**Performance-Wert**	**Anz. der Signale**	**Laufleistung**	**Zeitraum**	**Daten**
Fzg. *219*	0,36 – 0,52	409	19903 km	211 Tage	42 GB
Fzg. *209*	0,36 – 0,87	371	29995 km	317 Tage	46 GB
Fzg. *618*	0,36 – 0,86	372	20996 km	259 Tage	34 GB
Fzg. *848*	0,36 – 0,78	413	17299 km	273 Tage	29 GB
Fzg. *537*	0,34 – 0,60	355	22527 km	167 Tage	56 GB
Fzg. *704*	0,35 – 0,87	376	18150 km	239 Tage	25 GB
Fzg. *382*	0,36 – 0,56	410	25946 km	188 Tage	67 GB

Tabelle 5.2: Übersicht zur Anzahl der Samples in ausgewählten Diskretisierungsstufen des Fahrzeugs *219*

Diskretisierungsstufe d	**Anzahl der Samples** T
10 min	3694
1 h	615
6 h	102
15 h	41

die Sample-Anzahl nicht direkt aus dem Gesamtzeitraum der Messerfassung berechnet werden.
Die Eingangsdatenmatrix zu Erstellung eines rechnergestützten Modells zur Alterungsbestimmung ist sowohl von der Anzahl an Samples T als auch von der zu verwendenden Anzahl an Attributen n bestimmt. Die Anzahl an Attributen n der Eingangsdaten setzt sich aus der Anzahl der Signale und der Wahl der Merkmale zusammen. Beispielsweise sei eine Eingangsdatenmatrix mit 5 Signalen und jeweils 7 statistische Merkmalen gegeben. Bei einer Sample-Anzahl von $T = 615$ entstünde daraus eine 615×35 Eingangsdatenmatrix.

5.1.2 Hypothesen

Das Sensormodell zur Bestimmung der Abnutzungserscheinung soll anhand eines Gütemaßes bewertet werden. Im Rahmen der Problembeschreibung und des anschließenden Konzeptentwurfs sind unterschiedliche Hyperparameter der Vorverarbeitung und des Data-Minings eingeführt worden, die die Vorhersagequalität beeinflussen. Zur Erstellung des Sensormodells sind zwei Ansätze denkbar. Bei dem einen wird das spezifische Wissen aus der Fachdomäne verwendet, bei dem anderen Ansatz werden die einzustellenden Parameter datengetrieben im Rahmen einer Hyperparameteroptimierung bestimmt. Es folgen die Aufstellung der beiden Hypothesen **H 1** und **H 2**:

H 1: Verbesserung der Vorhersageergebnisse durch Anwendung einer datengetriebenen Optimierung gegenüber dem Expertenwissen aus der Fachdomäne. Es wird vermutet, dass mit Hilfe des spezifischen Wissens aus der Fachdomäne zwar geeignete Vorhersagen getroffen möglich sind, diese aber mit Hilfe einer datengetriebenen Optimierung verbessert werden können. Ein Experte des Fachbereichs besitzt Kenntnisse über die physikalischen Zusammenhänge der Alterung. Mit Hilfe einer datengetriebenen Optimierung fließen weitere Einflussfaktoren in das Modell ein, die ein Fachexperte vermeintlich nicht betrachtet.

H 2: Beeinflussung der Vorhersageergebnisse durch Wahl der Diskretisierungsstufe. Weiterhin wird an dieser Stelle ein Einfluss zwischen Qualität der Vorhersage und der gewählten Diskretisierungsstufe vermutet. Die Wahl der Diskretisierungsstufe bestimmt, wie viele Rohdaten zu einer Merkmalsmatrix zusammengefasst werden und ist damit mit einer möglichen Datenreduktion verknüpft. Für die Vorhersage einer langfristigen Alterung ist die Datenvorverarbeitung so zu wählen, dass diese langfristigen Änderungen innerhalb der Merkmalsmatrix beobachtbar sind. Eine Beeinflussung durch die Wahl der Diskretisierungsstufe wird in beiden Ansätzen vermutet.

Im weiteren Verlauf sollen die aufgestellten Hypothesen überprüft werden. Dazu wird zunächst das Prognosegütemaß zur Beurteilung der Alterungsvorhersage in Kapitel 5.1.3 eingeführt. Im Anschluss wird ein hybrider Expertenansatz vorgestellt (vgl. Kapitel 5.2), der einen intuitiven Lösungsansatz bereitstellt, bei dem nur eine geringe Anzahl an möglichen Einstellparametern mit Hilfe des Wissens aus der Fachdomäne benutzt werden.
Daran anschließend folgt die Vorstellung der datengetriebenen Hyperparameteroptimierung (vgl. Kapitel 5.3). Diese weist im Vergleich zum hybriden Expertenansatz einen hochdimensionalen Raum an einstellbaren Hyperparametern auf. Der Parameterraum ist neben der Auswahl der ML-Methode und deren Einstellparametern auch durch die Wahl der Signale, der Wahl der Diskretisierungsstufe und der Wahl der Merkmale, die den Signalen zugeordneten werden, charakterisiert. Die Ergebnisse werden in Kapitel 5.4 erläutert. Dort werden auch Einflüsse der unterschiedlichen Hyperparameter auf die Vorhersagegüte dargestellt und diskutiert.

5.1.3 Einführung des Prognosegütemaßes

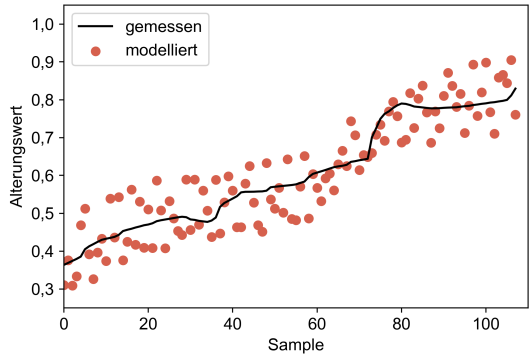
Im Rahmen der Validierung werden unterschiedliche Modellkonfigurationen miteinander verglichen. Zur Beurteilung der Vorhersagegüte eines Modells werden die vorhergesagten Alterungswerte mit den wahren gemessenen Werten verglichen. Eine Abweichung der Prognose wird als Ausreißer bezeichnet. Wie bereits in Kapitel 2.1.4 vorgestellt wurde, eignen sich unterschiedliche Prognosegütemaße für bestimmte Anwendungen besser [Alb+09, S.556 f., Knö18]. In dieser Arbeit wird ein Prognosegütemaß bevorzugt, welches sensitiv auf Ausreißer reagiert. Bezogen auf den vorliegenden Anwendungsfall führt eine vermeintlich zu hoch prognostizierte Alterung zu einer Warnmeldung. Diese könnte den Kunden zu

einem unnötigen Werkstattbesuch ermutigen, ohne dass eine Reparatur notwendig sei. Sowohl der RMSE als auch der MAE sind dimensionsbehaftet. Der MSE bildet die Einheit der Eingangsdaten quadratisch ab und kann deshalb schlechter interpretiert werden. Der RMSE bewertet Ausreißer sensibler als der MAE.
Die Autoren Armstrong und Collopy bewerten in [AC92] unterschiedliche Prognosegütemaße hinsichtlich verschiedener Kriterien. Sie zeigen, dass der RMSE, MAPE und der MdAPE gut zu einer Entscheidungsfindung eines möglichen Zusammenhangs beitragen können. Der RMSE wird auch in der vorgestellten Literatur als Gütemaß zur Beurteilung der Vorhersagequalität verwendet, z.B. in [Liu+17] oder [Zhe+18].
Aus den genannten Vorzügen wird der RMSE als Gütemaß zur Bewertung einer gewählten Konfiguration einer jeweiligen Iteration benutzt. Eine Konfiguration besteht dabei aus den gewählten Parametern der Vorverarbeitung und den Parametern des Data-Minings. In der finalen Darstellung der Ergebnisse wird für eine zusätzliche skalenunabhängige Darstellung auch der MAPE angegeben.

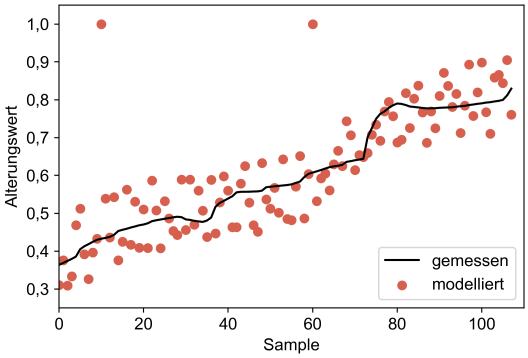
Im weiteren Verlauf wird anhand von Beispielen (vgl. Abbildung 5.1) dargestellt, wie sich der RMSE (vorgestellt in Kapitel 2.1.4) und der MAPE bei unterschiedlich vorhergesagten Alterungswerten voneinander unterscheiden. Die Abbildungen 5.1a-d zeigen synthetisch erzeugte Alterungswerte, welche mit den wahren Alterungswerten verglichen werden. Die ersten zwei Abbildungen 5.1a-b zeigen Beispiele auf, bei denen die Vorhersage im Zusammenhang mit den gemessenen Daten stehen. Die anderen Beispiele (vgl. Abbildungen 5.1c-d) zeigen, wie sich verrauschte Werte auf den RMSE und MAPE auswirken. Der RMSE ist aufgrund der Abhängigkeit von der zu untersuchenden Größe einheitenbehaftet. Ein niedriger Wert des RMSEs gibt eine hohe Ähnlichkeit von wahren und vorhergesagten Alterungswerten an. Der durchschnittliche absolute prozentuale Prognosefehler MAPE ist dimensionslos.

5.2 Vorstellung des hybriden Expertenansatzes

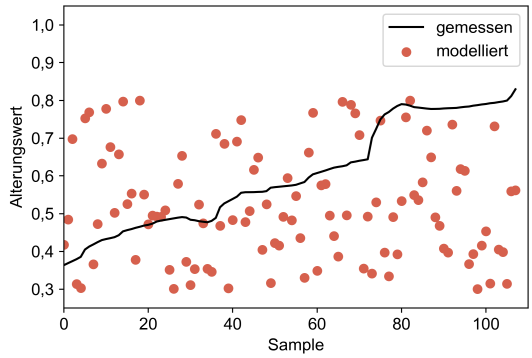
In diesem Abschnitt wird ein hybrider Expertenansatz zur Erstellung eines Alterungsmodells beschrieben. Dieser stellt eine Kombination aus intuitivem Vorgehen und spezifischen Wissen aus der Fachdomäne dar. Ein rein physikalischer Ansatz ist nicht vorhanden und kann deshalb nicht verwendet werden. Die Alterungsbestimmung der untersuchten Komponente wird bislang über externe Performance-Messungen in Werkstätten durchgeführt (vgl. Kapitel 2.3.3). Aus diesem Grund werden auch im hybriden Expertenansatz rechnergestützte Modelle mit Hilfe von ML-Methoden angewendet. Im weiteren Verlauf wird der in dieser Arbeit vorgestellte datengetriebene Optimierungsansatz (vgl. Abschnitt 5.3) mit diesem hybriden Expertenansatz verglichen.
Zunächst ist es denkbar, dass ein Expertenansatz aus einer analytischen Berechnung der Alterung besteht (vgl. Abschnitt 2.3.3). Eine solche Berechnung ist in diesem Fall nicht möglich. Die in dieser Arbeit untersuchten Fahrzeuge verfügen nicht über die ausreichende Ausstattung an Sensoren, sodass eine Bereitstellung entsprechender Größen an den notwendigen Messstellen nicht möglich ist.



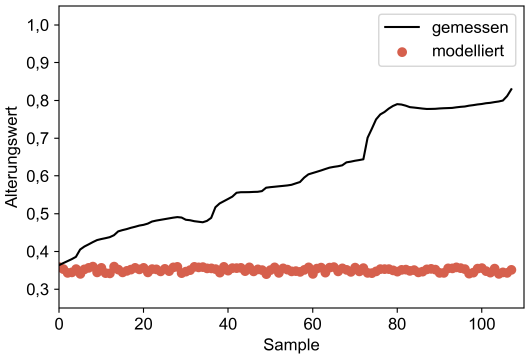
(a) Beispieldarstellung von gemessener Alterung und synthetisch erzeugter Alterungswerte, die um den wahren Wert verrauscht sind; RMSE = 0,0621; MAPE = 0,0926



(b) Beispieldarstellung von gemessener Alterung und synthetisch erzeugter Alterungswerte, die um den wahren Wert verrauscht sind inkl. Ausreißern; RMSE = 0,0907; MAPE = 0,1089



(c) Beispieldarstellung von gemessener Alterung und synthetisch erzeugter Alterungswerte, die mit der Alterung nicht im Zusammenhang stehen, sondern zufällig zwischen 0,3 und 0,8 verrauscht sind; RMSE = 0,2196; MAPE = 0,2934



(d) Beispieldarstellung von gemessener Alterung und synthetisch erzeugter Alterungswerte, die mit der Alterung nicht im Zusammenhang stehen, sondern um den Alterungswert von 0,35 verrauscht sind; RMSE = 0,2880; MAPE = 0,3845

Abbildung 5.1: In dieser Abbildung werden vier unterschiedliche Ausprägungen von Alterungsvorhersagen vorgestellt und zu jedem Beispiel wird der RMSE und der MAPE bestimmt

Statt einer analytischen Berechnung der Alterung wird an dieser Stelle ein hybrider Expertenansatz vorgestellt, der sowohl die datengetriebene Welt aus dem Bereich des MLs als auch das Fachwissen aus der Domäne vereint. Es wird eine dem Fachexperten zumutbare Menge an Konfigurationsmöglichkeiten zur Modellerstellung bereitgestellt. Dieses Vorgehen ist durch eine anwendungsorientierte Selektion von Eingangsdaten der Fachexperten und durch weitere statistischen Analysen charakterisiert. Weiterhin werden aus den bereits vorgestellten ML-Methoden diejenigen ausgewählt, die die Alterung unter gewählten Randbedingungen am besten abbilden können.

5.2.1 Hyperparameter der Datenvorverarbeitung

Die Datenvorverarbeitung des hybriden Expertenansatzes teilt sich wiederum in drei Teilbereiche. Zunächst werden Signale aus der Eingangsdatenmenge bestimmt. Im Anschluss wird eine geeignete Diskretisierungsstufe festgelegt. Zuletzt wird eine Auswahl relevanter Merkmale mit Hilfe von statistischen Analysen getroffen.
Die Fachexperten der Domäne haben Signale bestimmt, die auf Grund ihres physikalischen Verständnisses mit der in dieser Arbeit untersuchten Abnutzungserscheinung im Zusammenhang stehen sollten. Zu diesen Signalen zählen interne Modulationsgrößen wie z.B. AGR-Massenstrom, AGR-Gastemperaturen und Signale zu Ventilstellungen.
Die Festlegung einer Diskretisierungsstufe ist auch für den Fachexperten nur bedingt möglich. Aus diesem Grund wird im Expertenansatz durch unterschiedliche Diskretisierungsstufen iteriert, um so anhand dieser Ergebnisse eine geeignete Wahl der Diskretisierungsstufe festzulegen.
Zur Reduzierung der Merkmale in einer Datenmenge unterscheidet die Literatur zwischen einem *Filter-Ansatz* und einem *Wrapper-Ansatz* [Alp10, S. 138 f., GE03]. Der Filter-Ansatz berechnet die Relevanz der einzelnen Merkmale. Nur die ausgewählten relevanten Merkmale werden im Anschluss für das Training verwendet. Der Wrapper-Ansatz bezieht bei der Auswahl der Merkmale den entsprechenden Lernalgorithmus mit ein. Dabei wird durch mehrfaches Iterieren die Teilmenge an Merkmalen gesucht, die die beste Vorhersagequalität liefern. Bei diesem Vorgehen ist der Lernalgorithmus bereits zu Beginn festzulegen. Ausgewählte Arbeiten im Bereich der *Feature Selection* wurden bereits in Kapitel 3.3.1 vorgestellt.
Der Wrapper-Ansatz ist ein rechenintensives Vorgehen. Im Vergleich zum Wrapper-Ansatz ist der Filter-Ansatz weniger rechenintensiv. Aus diesem Grund werden im weiteren Vorgehen die Merkmale ausgewählt, die keine ähnlichen oder redundanten Informationen enthalten. Es werden zunächst alle in der Tabelle 4.2 vorgestellten Merkmale benutzt. Im weiteren Schritt werden die Korrelationen der einzelnen Merkmale untereinander betrachtet und ausgewertet. Korrelieren zwei Merkmale zueinander (vgl. Kapitel 2.1.4, hohe Korrelation bei $R \geq 0,8$), so wird im Sinne des *Filter-Ansatzes* eines der beiden Merkmale für die weitere Analyse nicht weiter betrachtet.
Es sei darauf hingewiesen, dass die Auswahl untereinander nicht korrelierender Merkmale keine Evidenz dafür liefert, dass diese Merkmale für eine zu untersuchende Alterungsvorhersage relevant sind. Dennoch bietet dieses Vorgehen eine Möglichkeit, die zu verwendenden

Merkmale für den hybriden Expertenansatz einzuschränken. Auf der anderen Seite zeigt eine hohe Korrelation von zwei Merkmalen, dass diese Merkmale innerhalb der betrachteten Daten ähnliche Informationen liefern. Im weiteren Verlauf werden die Korrelationen der Merkmale sämtlicher zur Verfügung stehender Daten der Fahrzeuge berechnet.
Die Abbildung 5.2 stellt die Korrelationswerte der unterschiedlichen Merkmale dar. Die Werte sind gemittelt über alle vorgestellten Fahrzeuge und beinhalten sämtliche Diskretisierungsstufen. Nun werden die Merkmale aus der Gesamtmenge an Merkmalen entfernt, die eine hohe Korrelation zu einem anderen Merkmal aufweisen. Die Merkmale *Median*, *Q_25* und *Q_75* zeigen sehr hohe Korrelationswerte zum *Mean*-Merkmal und werden aus diesem Grund nicht weiter betrachtet. Nach Entfernung von miteinander korrelierenden Merkmalen

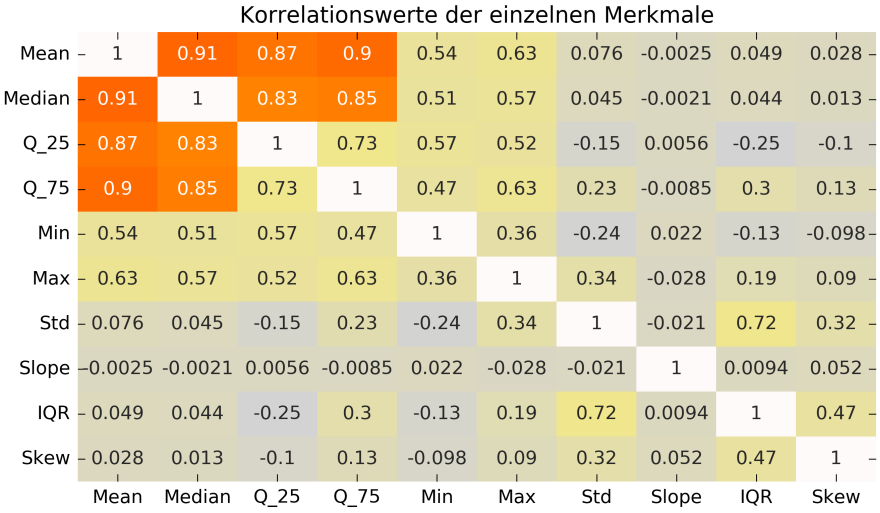


Abbildung 5.2: Darstellung der Korrelationswerte von unterschiedlichen Merkmalen unter Verwendung sämtlicher zur Verfügung stehender Fahrzeuge und Diskretisierungsstufen

ergibt sich folgende reduzierte Gesamtmenge an Merkmalen: *Mean*, *Min*, *Max*, *Std*, *IQR*, *Skew* und *Slope*.
In diesem Abschnitt wurden die Merkmale durch Korrelationsanalysen reduziert. Diese reduzierte Menge an Merkmalen wird im weiteren Verlauf für den hybriden Expertenansatz eingesetzt.

5.2.2 Hyperparameter des Data-Minings

In diesem Abschnitt werden die expertenbasierten Hyperparameter des Data-Minings vorgestellt. Die zur Auswahl stehenden Lernalgorithmen sind in der Tabelle 4.5 (vgl. Abschnitt 4.6) dargestellt.
Der im Rahmen des hybriden Expertenansatzes zu verwendende Lernalgorithmus soll im folgenden Abschnitt festgelegt werden. Dazu steht bereits eine Vorauswahl an Lernalgorithmen aus dem Abschnitt 4.4 zur Verfügung. Der hybride Expertenansatz sieht weiterhin vor,

das intuitiv die Standardimplementierung der Hyperparameter[4] der jeweiligen Lernalgorithmen verwendet wird.
Die Abbildung 5.3 stellt die Ergebnisse der in dieser Arbeit vorliegenden Alterungsvorhersage unterschiedlicher Lernalgorithmen dar. Dazu wurden unterschiedliche Diskretisierungsstufen (*10 Minuten* bis *150 Stunden*) verwendet (vgl. Kapitel 4.3.1). Außerdem wurden die bereits festgelegten Annahmen der Datenvorverarbeitung des hybriden Expertenansatzes mit einbezogen. Ein Wert der Vorhersagegüte (vgl. Abbildung 5.3) entspricht dabei dem gemittelten RMSE aus den unterschiedlichen Iterationen der verschiedenen Kombinationsmöglichkeiten von Test- und Trainingsdaten. Die einzelnen Boxplots beinhalten die Vorhersagegüten bei unterschiedlichen Diskretisierungsstufen. Es ist erkennbar, dass sich mit

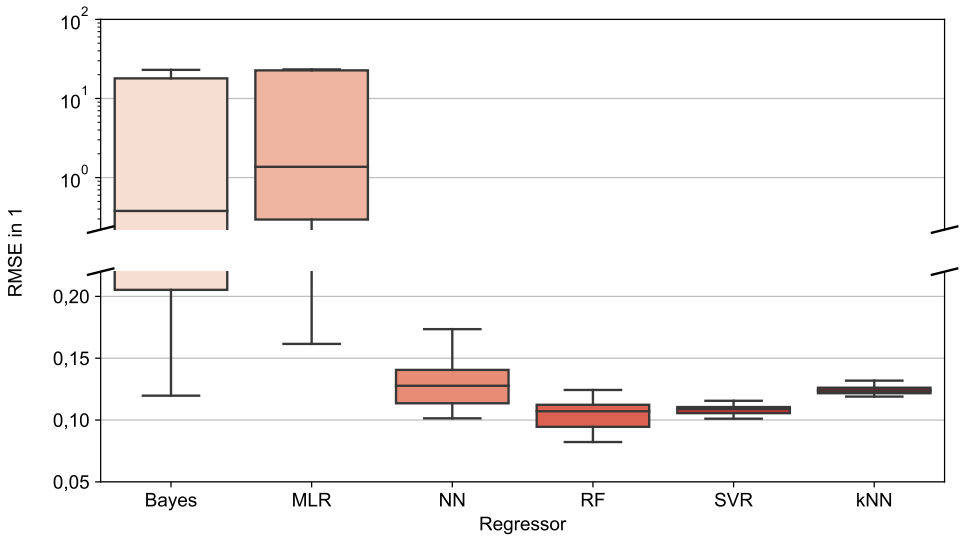


Abbildung 5.3: Darstellung der Güte der Alterungsvorhersage mit Hilfe des RMSEs als Boxplot unter Anwendung der von Fachexperten bestimmten Signalauswahl bei unterschiedlichen Regressoren und Diskretisierungsstufen, Nutzung aller zur Verfügung stehender Fahrzeugdaten nach dem Kreuzvalidierungsverfahren

Hilfe der MLR und des bayesschen Regressors bei diesem intuitiven Expertenansatz keine brauchbaren Vorhersageergebnisse erzielen lassen.
Dagegen liefern die SV-, kNN- und RF-Regressoren unter Anwendung der festgelegten Signalmenge Ergebnisse mit einer Vorhersagegüte von $RMSE < 0.2$. Der geringste RMSE stellt sich bei der Verwendung des RF-Regressors ein. Die Ergebnisse unter Anwendung des künstlichen NNs zeigen, dass die Vorhersagegüten in einem größeren Bereich schwanken und dadurch sehr von der Wahl der Diskretisierungsstufe abhängig sind. In der Abbildung 5.3 ist zu sehen, dass mit Hilfe des RF-Regressors im Vergleich aller Lernalgorithmen eine vergleichsweise gute Vorhersagegüte für das untersuchte Anwendungsbeispiel abgebildet werden kann.

4 Innerhalb der Scikit-learn Bibliothek gibt es für verschiedene Lernalgorithmen festgelegte Standard-Hyperparameter, die in der jeweiligen Dokumentation nachgelesen werden können. Link: `https://scikit-learn.org/stable/`

Die Tabelle 5.3 stellt den besten RMSE ($RMSE_{min}$) der unterschiedlichen Regressoren unter Anwendung der festgelegten Signalmenge dar. Außerdem wird zur jeder ML-Methode das durchschnittliche Laufzeitverhalten ($t_{Durchschnitt}$) zur Modellerstellung angegeben. Es

Tabelle 5.3: Tabellarische Darstellung der besten Vorhersagegüte und des Laufzeitverhaltens unter Anwendung unterschiedlicher Regressoren für den Expertenansatz (t: Laufzeitverhalten zur Modellerstellung)

ML-Methode	$RMSE_{min}$	$t_{Durchschnitt}$
Bayes	0,1197	0,0226s
MLR	0,1460	0,0172s
NN	0,1014	35,6718s
RF	0,0822	0,3425s
SVR	0,1011	0,2732s
kNN	0,1111	0,3673s

zeigt sich, dass der RF-Regressor nicht nur gute Vorhersageergebnisse für diese gewählte Konfiguration liefert, sondern auch eine vergleichsweise geringe durchschnittliche Laufzeit aufweist. Zwar liefert z.B. der Multi-lineare-Regressor deutlich schneller Ergebnisse, zeigt dabei aber keine besonders gute Vorhersagequalität. Auffallend ist, dass das gewählte neuronale Netz unter den gewählten Einstellparametern eine hohe Laufzeit aufweist.
Zusammenfassend kann an dieser Stelle festgestellt werden, dass der RF-Regressor intuitiv gute Ergebnisse bei vergleichsweise geringem Zeitaufwand liefert. Weiterhin ist der RF-Regressor in der Literatur durch seine Effizienz und Robustheit gekennzeichnet [HH17, S. 639]. Außerdem zeigen erste Vorarbeiten gute Ergebnisse unter Anwendung des RF-Regressors [SEI20]. Aus den genannten Gründen wird deshalb an dieser Stelle der RF-Regressor als Regressor des hybriden Expertenansatzes eingesetzt.
Die Abbildung 5.4 stellt die Ergebnisse des hybriden Expertenansatzes bei unterschiedlichen Diskretisierungsstufen unter Anwendung des RF-Regressors dar. Die eingestellte Konfiguration beinhaltet den RandomForest-Regressor mit Standard-Parametrisierung. Die dargestellte Abbildung 5.4 lässt sich in zwei Bereiche einteilen: Auf der linken Seite der Abbildung ist zu erkennen, dass die Prognosegüte mit abnehmendem Diskretisierungsgrad schlechter wird. An dieser Stelle kann die Alterung unter Anwendung der Standard-Parametrisierung des RandomForest-Regressors nicht mit einer hohen Güte abgebildet werden. Auf der rechten Seite der Abbildung 5.4 ist kein eindeutiger Trend ablesbar. Hier stellen sich unterschiedliche gute Prognosegüten ein, die bessere Ergebnisse erzielen als eine sehr feine Diskretisierungsstufe.
Neben der Laufzeit werden auch die zur Verfügung stehenden Samples aus den Testdatensätzen der verschiedenen Iterationen der Kreuzvalidierung betrachtet. Die nachfolgende Tabelle 5.4 stellt die besten Diskretisierungsstufen des Expertenansatzes unter Anwendung des RF-Regressors dar. Auch wenn das beste Ergebnis bei einer Diskretisierungsstufe von 40 Stunden vorliegt, soll an dieser Stelle darauf hingewiesen werden, dass bei einer geringen Diskretisierungsstufe mehr Samples zur Verfügung stehen. Eine zu hohe Diskretisierungsstufe hat den Nachteil, dass eine zu große Datenmenge aggregiert wird und dadurch keine

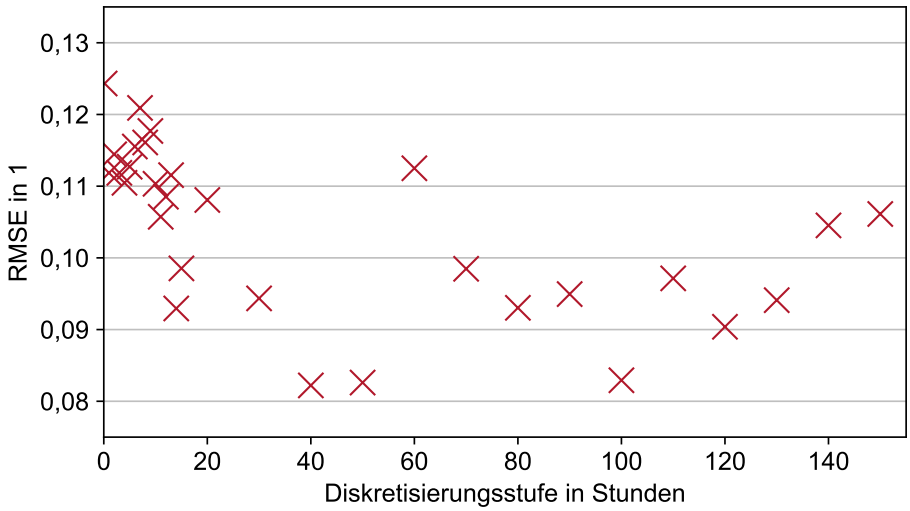


Abbildung 5.4: Darstellung der Prognosegüte der Alterungsvorhersage unter Anwendung des hybriden Expertenansatzes bei unterschiedlichen Diskretisierungsstufen und unter Anwendung des RF-Regressors, Nutzung aller zur Verfügung stehender Fahrzeugdaten nach dem Kreuzvalidierungsverfahren

Tabelle 5.4: Tabellarische Darstellung der fünf besten Diskretisierungsstufen des Expertenansatzes unter Anwendung des RF-Regressors, sortiert nach RMSE in aufsteigender Reihenfolge

	RMSE	Diskretisierungsstufe
1	0,0822	40 h
2	0,0826	50 h
3	0,0829	100 h
4	0,0904	120 h
5	0,0929	14 h

zeitnahe Vorhersage möglich ist.
Die Abbildung A.5 stellt die modellierten Alterungswerte der jeweils besten Konfiguration an Hyperparametern für die unterschiedlichen ML-Methoden unter Anwendung des hybriden Expertenansatzes dar. Die dargestellten modellierten Alterungswerte sind die Ergebnisse der Alterungsprädiktion des Expertenmodells auf Basis der Testdaten. Dennoch sind die Alterungscharakteristiken aller sieben Fahrzeuge zu sehen, da das erzeugte virtuelle Sensor-Modell nach dem Kreuzvalidierungsverfahren erzeugt worden ist und somit alle Kombinationen aus Trainings- und Test-Daten vereint worden sind. Es beste Vorhersagegüte stellt sich bei einer Diskretisierungsstufe von 40 Stunden ein (vgl. Abbildung A.5e). Phasenweise werden hier die Alterungswerte sehr genau vorhergesagt. Dennoch unterliegt die Vorhersage in einzelnen Bereichen Ungenauigkeiten. Außerdem zeigt die Abbildung A.5e, dass keine modellierten Werte vorliegen, die außerhalb des beobachteten Alterungsbereichs liegen. Die Abbildung A.6 stellt die modellierten Alterungswerte der jeweils besten Konfiguration an

Hyperparametern für die unterschiedlichen Diskretisierungsstufen unter Anwendung des hybriden Expertenansatzes dar.

5.3 Vorstellung der datengetriebenen Hyperparameteroptimierung

Im folgenden Abschnitt wird die datengetriebene Hyperparameteroptimierung beschrieben. Wie bereits im Rahmen des Konzepts in Kapitel 4 vorgestellt wurde, werden die einstellbaren Hyperparameter in *Hyperparameter der Datenvorverarbeitung* (vgl. Kapitel 5.3.1) und *Hyperparameter des Data-Minings* (vgl. Kapitel 5.3.2) eingeteilt.
Aus den im folgenden Abschnitt vorgestellten Hyperparametern der Datenvorverarbeitung und Hyperparametern des Data-Minings entsteht der Parameterraum zur datengetriebenen Hyperparameteroptimierung. Dieser hochdimensionale Parameterraum besteht zu Teilen aus numerischen und kategorialen Hyperparametern. In dem Kapitel 4.6 wurden bereits unterschiedliche Methoden der bayesschen Optimierung erläutert. Dabei zeigt sich, dass der baumstrukturierte Kerndichteschätzer (TPE) in vielen Anwendungsfällen der Literatur adäquate Ergebnisse bei vergleichsweise geringen Laufzeiten liefert. Zeitgleich eignet sich dieser auch für den Einsatz eines hochdimensionalen Parameterraums. Beim TPE wird der Suchraum baumartig angeordnet und mit Dichten angereichert. Nach jeder gewählten Konfiguration an Hyperparametern werden die entsprechenden Werte der Dichte aktualisiert, um das Modell der Abhängigkeit zwischen gewählter Konfiguration an Hyperparametern und der resultierenden Performance zu schärfen [Feu+15; Ber+11]. Implementierungen von TPE liegen in unterschiedlichen Programmiersprachen vor, beispielsweise in Python, Java oder *MATLAB* [MML19].
Die Bibliothek *Hyperopt* liefert eine Implementierung der TPE-Methode in Python [BYC13] und wird im weiteren Verlauf zur bayesschen Optimierung eingesetzt. *Hyperopt* ist ausführlich dokumentiert und erzielt vergleichsweise schnelle, so wie robuste Ergebnisse [WWL20]. Es kann ein komplexer Suchraum erstellt werden. Die Hyperparameter können reelle, diskrete oder kategoriale Dimensionen annehmen. Im Rahmen dieser Arbeit wird die Version 0.2.4 verwendet. Neben der genannten Implementierung gibt es auch weitere Implementierungen der TPE-Methode (wie z.B. *Optuna*) [Aki+19].
Neben den Hyperparametern der Vorverarbeitung und des Data-Minings ist auch die TPE-Methode zu konfigurieren. Die initialen Parameter der Optimierungsmethode können das finale Ergebnis beeinflussen. Dazu kann die Anzahl der randomisierten Start-Konfigurationen (*n_startup_jobs* = 250) des Suchraums festgelegt werden. Dieser Wert ist in der vorliegenden Implementierung standardmäßig auf 20 festgelegt und wird im weiteren Verlauf auf 250 erhöht, da erste Vorarbeiten vielversprechende Ergebnisse bei einer vergleichsweise hohen Anzahl initialer Konfigurationen gezeigt haben. Desweiteren kann auch die Anzahl der gesamten Iterationen bestimmt werden. Bereits in Kapitel 4.6 wurden Arbeiten gezeigt, die die Anzahl der Iterationen auf einen Wert zwischen 250 bis 2000 festgelegt haben [MML19; BYC13; Ili+17]. Dieser Parameter ist von den vorliegenden Daten und des Anwendungsfalls abhängig. Die Erstellung eines virtuellen Sensormodells findet offline außerhalb des Fahrzeugs statt. Im weiteren Verlauf wird bewusst eine vergleichsweise hohe Anzahl an Iterationen (*iterations* = 5000) festgelegt, da eine nachträgliche Erhöhung des Wertes während

des Durchlaufs nicht möglich ist. In einer zukünftigen Fahrzeug/Cloud-Umgebung ist eine Durchführung dieser Modellerstellung auf einem Rechencluster innerhalb einer Cloud denkbar. In diesem Fall könnte die Anzahl an Gesamtiterationen weiter erhöht werden, sofern es die vorliegenden Rechenkapazitäten der Cloud erlauben würden.

5.3.1 Hyperparameter der Datenvorverarbeitung

Die einstellbaren Hyperparameter der Datenvorverarbeitung werden wiederum in folgende drei Abschnitte aufgeteilt: Wahl der Signalmenge, Wahl der Diskretisierungsstufe und Wahl der Merkmalsmenge. Die nachfolgenden Abschnitte geben weitere detaillierte Informationen zu diesen Hyperparametern.

Wahl der Signalmenge

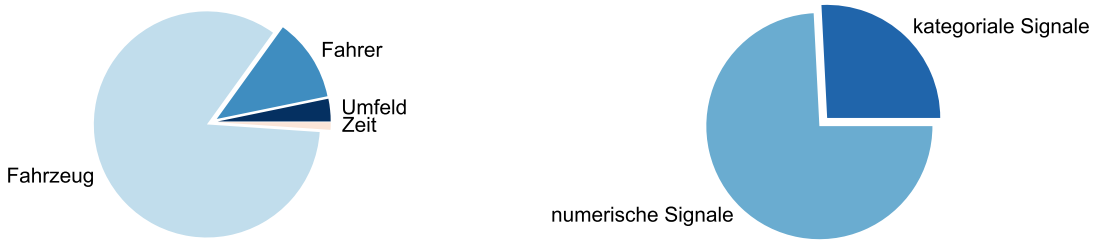
Für die Modellerstellung können sämtliche aufgenommene Signale in Betracht gezogen werden. Auf der einen Seite können Signale *einzigartig* sein, wenn diese in der Datenmenge nur einmal vorkommen und keine weiteren Signale mit den gleichen Informationen vorliegen. Auf der anderen Seite können Signale *verwechselbar* sein, wenn sich nahezu die gleichen Informationen in mehreren Signalen befinden. An dieser Stelle sei das Beispiel mit den Geschwindigkeiten genannt, die an unterschiedlichen Rädern aufgezeichnet werden können, es resultieren daraus vier Geschwindigkeitssignale, die alle nahezu die gleichen Informationen beinhalten.
Es wurde bereits gezeigt, dass die Effizienz eines Modells durch Nutzung redundanter Signale nicht gesteigert werden kann (vgl. Kapitel 3.2 und 4.3). Um die gesamt zu verarbeitende Datenmenge zu reduzieren und dem Modell einzigartige Informationen zur Verfügung zu stellen, sollen die Signale vom Algorithmus eigenständig in Cluster eingeteilt werden. Innerhalb eines Clusters soll sich eine Gruppe von verwechselbaren Signalen befinden. Daraus wird ein Vertreter dieser Signale gewählt und zur weiteren Verarbeitung genutzt. Der Algorithmus ist so zu entwerfen, dass eine Clustergruppe auch aus nur einem Teilnehmer bestehen darf. Dadurch ist gewährleistet, dass auch einzigartige Signale zwar einem Cluster zugeordnet werden, dieser Cluster wiederum aber keine weiteren Teilnehmer beinhaltet. In der Literatur wird dies auch als Erkennung von Ausreißern bezeichnet.
Das Vorgehen des Clusterings soll unüberwacht erfolgen, da es zu diesem Zeitpunkt keine Klassifizierung der Signale geben kann. Ein *interner Index* beschreibt die Güte der gefundene Klassenstruktur ohne zusätzliche Informationen und wurde bereits in Kapitel 2.1.2 vorgestellt.
Der SOM und der *k-Means* Algorithmus werden nicht weiter betrachtet, da diese Algorithmen keine Ausreißer in den Daten verarbeiten können. Stattdessen werden zunächst DBSCAN und Agglomerative Clustering als mögliche Algorithmen identifiziert. Die genannten Algorithmen sollen um ein drittes Verfahren (MeanShift Algorithmus) ergänzt werden. Dieser wird vor allem in der Bildverarbeitung eingesetzt und erzielt dort gute Ergebnisse [Car15]. Zum Zeitpunkt der Durchführung des Clustering-Algorithmus ist die ideale

Signalgruppenanzahl nicht bekannt. Eine geeignete Anzahl an Signalgruppen ist von dem zu implementierenden Clustering-Verfahren eigenständig festzustellen.
Ausgewählte Clustering-Verfahren können eigenständig hinsichtlich des besten Clustering-Ergebnisses optimieren. Dazu wird der Silhouetten-Koeffizient verwendet, dieser findet auch Anwendung in den bereits vorgestellten Arbeiten von Mrowca, Moser und Gunnemann [MMG18] und Calabrese, Campanella und Proverbio [CCP12]. Neben der Verwendung des Silhouetten-Koeffizienten haben die Arbeiten gemeinsam, dass sie zur Ähnlichkeitsbestimmung die euklidische Distanz verwenden. Auch in dieser Arbeit wird zur Bestimmung der Ähnlichkeit der einzelnen Signale die euklidische Distanz benutzt.
Unterschiedliche Parameter des Clusterings werden mit Hilfe des Silhouetten-Koeffizienten bewertet. Letztendlich wird die vielversprechendste Konfiguration ausgewählt und zur Identifikation von einzigartigen Signalen benutzt. Der Wert des Silhouetten-Koeffizienten kann zwischen -1 und 1 liegen, wobei das Clustering ab einem Wert von $0,5$ adäquate Strukturen erkennt [KR05, S. 87 f., ES00b]. Im Anhang befinden sich drei zugehörige Abbildungen, in denen der jeweilige Silhouettenverlauf dargestellt ist (vgl. Abbildung A.1, A.2 und A.3).
Für eine weitere Evaluierung wurden die wahren Klassenzuordnungen der einzelnen Signale vom Fachexperten durchgeführt und in einer Validierungstabelle gespeichert. Mit Hilfe eines *externen Index* kann so die Güte bestimmt werden, in diesem Fall wird der FMI angewendet. Die Tabelle 5.5 zeigt die unterschiedlichen Klassifizierungsgüten bei Verwendung

Tabelle 5.5: Vergleich unterschiedlicher Clustering-Methoden bzgl. Klassifizierungsgüte und Laufzeitverhalten zur Identifizierung einzigartiger Signale des Fahrzeugs *219*

	MeanShift	**DBSCAN**	**Agglomerative Clustering**
Klassifizierungsgüte	$93,5\%$	$93,8\%$	$91,2\%$
Laufzeitverhalten	134s	34s	16s

des FMIs der Cluster-Algorithmen und deren Laufzeitverhalten. Der DBSCAN-Ansatz zeigt gute Ergebnisse bezogen auf die Klassifizierungsgüte. Er weist ein vergleichsweise geringes Laufzeitverhalten auf und ist für das multivariate Zeitreihen-Clustering geeignet. Ein ähnliches Vorgehen wählen Chandrakala und Sekhar auch in [CS08]. Aus diesen Gründen wird im weiteren Verlauf dieser Arbeit für alle aufgezeichneten Fahrzeugdaten die Identifikation einzigartiger Signale mit Hilfe des DBSCAN Clustering-Verfahrens durchgeführt. Unter Anwendung dieses Clustering-Verfahrens resultieren 93 einzigartige Signale aus der Schnittmenge der Signale aller Fahrzeuge.
Im weiteren Verlauf werden die in dieser Arbeit verwendeten Eingangsdaten weiter beschrieben. Dazu werden unterschiedliche Signalverteilungen in der nachfolgenden Abbildung 5.5 gezeigt. Die Menge einzigartiger Signale wird dabei anhand ihrer Verwendung und ihrer Beschaffenheit charakterisiert. Die erste Abbildung 5.5a zeigt die Einsortierung der Signale anhand deren Verwendung innerhalb der Fahrer-Fahrzeug-Umwelt-Umgebung. Die zweite Abbildung 5.5b stellt die Signale anhand ihrer Beschaffenheit dar. Es ist zu sehen, dass ein Großteil der Signale für die Ansteuerung und Beobachtung des Fahrzeugs zuständig sind. Außerdem besteht die Mehrheit der Signale aus numerischen Signalen.



(a) Einsortierung anhand der Signalverwendung **(b)** Einsortierung anhand der Signalbeschaffenheit

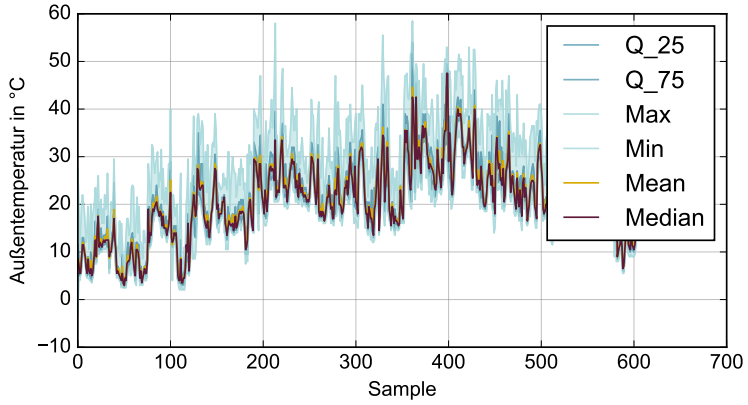
Abbildung 5.5: Darstellung der Verteilungen einzigartiger Signale zur Charakterisierung

Wahl der Diskretisierungsstufe

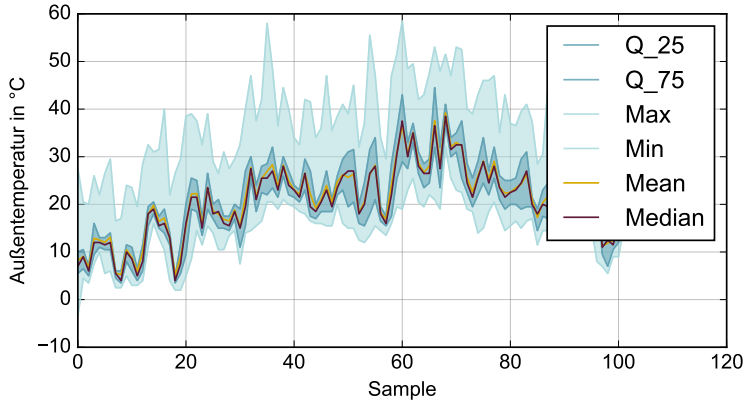
Die Diskretisierungsstufen bestimmen den Grad der Abtastung einer Zeitreihe. Die Heranführung an die Diskretisierungsstufe wurde bereits in Kapitel 4.3.1 gegeben und wird in diesem Abschnitt weiter konkretisiert. Aus einer feineren Diskretisierungsstufe resultieren mehr Datenpaare als im Vergleich zu einer gröberen Diskretisierungsstufe. Dadurch benötigen feinere Diskretisierungsstufen mehr Ressourcen im Hauptspeicher. Mit Hilfe der gewählten Diskretisierungsstufe werden hochdynamische Daten mit Hilfe von gewählten Merkmalen zu einer Sequenz aggregiert. Diese aggregierten Informationen werden zum effizienten Training der datengetriebenen Modelle genutzt.
Auf der einen Seite wird die Stufe der Diskretisierung nur so fein gewählt, wie gerade noch Analysen im Hauptspeicher beim gleichzeitigem Laden aller Signale möglich sind. Auf der anderen Seite wird die Diskretisierungsstufe dadurch begrenzt, dass dem Fahrzeugführer noch eine zeitnahe Alterungsprädiktion gewährleistet werden kann und ausreichend Datensamples für die Erstellung des Sensormodells zur Verfügung stehen. Es wird eine Diskretisierungsstufe zwischen *10 Minuten* und *150 Stunden* gewählt.

Wahl der Merkmalsmenge

Mit Hilfe von Merkmalen werden unter Anwendung der Diskretisierungsstufen die Eingangsdaten aggregiert. Eine erste umfassende Vorstellung der einzelnen Merkmale fand bereits in Kapitel 4.3.2 statt. Es wurde folgende Gesamtmenge an Merkmalen definiert: *Mean*, *Median*, *Q_25*, *Q_75*, *Min*, *Max*, *Std*, *IQR*, *Skew* und *Slope*.
Die Abbildung 5.6 zeigt mehrere Merkmale eines Fahrzeugs, aufgetragen über der Anzahl der Samples. Dabei ist die Abbildung in zwei Teilbereiche aufgeteilt: Die Abbildung 5.6a zeigt die Merkmale bei einer Diskretisierungsstufe von 1 Stunde, die Abbildung 5.6b stellt dagegen die Merkmale bei einer Diskretisierungsstufe von 6 Stunden dar.
Es ist zu sehen, dass bei einer feineren Diskretisierungsstufe der Detaillierungsgrad der Auflösung höher ist. Der gezeigte Außentemperaturverlauf zeigt eine höhere Dynamik in der



(a) Darstellung der Merkmale (*Mean*, *Median*, *Q_25*, *Q_75*, *Min* und *Max*) des Fahrzeugs *219*, Diskretisierungsstufe: 1 Stunde



(b) Darstellung der Merkmale (*Mean*, *Median*, *Q_25*, *Q_75*, *Min* und *Max*) des Fahrzeugs *219*, Diskretisierungsstufe: 6 Stunden

Abbildung 5.6: Darstellung mehrerer Merkmale bei unterschiedlichen Diskretisierungsstufen des Fahrzeugs *219*

feineren Diskretisierungsstufe, wohingegen die sichtbaren Änderungen der Temperaturamplituden der gröberen Diskretisierungsstufe deutlich geringer ausfallen.
Eine weitere Darstellung der Merkmale unter Anwendung einer Diskretisierungsstufe von 15 Stunden ist im Anhang in der Abbildung A.4 zu finden.

5.3.2 Hyperparameter des Data-Minings

Neben den Einstellparametern der Vorverarbeitung aus Kapitel 5.3.1 werden nun die Hyperparameter des Data-Minings vorgestellt. Diese setzen sich aus der Festlegung einer ML-Methode und deren zugehörigen Einstellparameter zusammen. Wie bereits in Kapitel 4.4 erläutert wurde, werden folgende Methoden des MLs angewendet: MLR, SVR, Bayes, RF, kNN und NN. Im weiteren Verlauf werden die Einstellparameter dieser Methoden näher erläutert und tabellarisch dargestellt.

Multiple lineare Regressor Die Implementierung der multiplen linearen Regression in Scikit-learn besitzt keine Parameter, die im Rahmen dieser Arbeit verändert werden.

Support Vector Regressor Die Implementierung der Support Vector Regression in Scikit-learn basiert auf der Support Vector Klassifikation. Der Ansatz beruht dabei auf der Separierung der Eingangsdaten. Mit Hilfe des Parameters ε wird ein Toleranz-Band festgelegt. Für alle abgebildeten Datenpunkte wird ein Fehler berechnet. Es gilt diesen Fehler zu minimieren, dabei ist C der Regularisierungsparameter innerhalb der Fehlerfunktion [Bis06, S. 340 f.]. Die Implementierung sieht vor, dass unterschiedliche Kernelfunktionen für lineare und nicht-lineare Anwendungen festgelegt werden können. Hierbei werden die Eingangsdaten in einen höherdimensionalen Raum transferiert. Zugehörig zu diesen Kernelfunktionen können weitere Parameter festgelegt werden, so kann bei einer Polynomfunktionen der Polynomgrad (*degree*) festgelegt werden. Außerdem kann die Anzahl der maximalen Iterationen (*max_iter*) festgelegt werden, damit der Lernalgorithmus in einer akzeptablen Zeit ein Ergebnis liefert. Die Tabelle 5.6 zeigt die Einstellparameter der Support Vector Regression und die in dieser Arbeit gewählten Ausprägungen.

Tabelle 5.6: Vorstellung der Einstellparameter des Support Vector Regressors

Einstellparameter des SVRs	Gewählte Ausprägungen
Kernel	linear, poly, rbf, sigmoid
Polynomgrad (*degree*)	2 bis 5
Regularisierungsparameter (C)	10^{-3} bis 10^{-1}
Toleranzband-Parameter (ε)	10^{-2} bis 10^{-1}
Toleranz-Parameter (*tol*)	default=10^{-3}
Maximale Iterationen (*max_iter*)	10000

Bayes Regressor Der bayessche Regressor gibt eine Wahrscheinlichkeitsverteilung als Vorhersage für zukünftige Ereignisse. Um die a-posteriori Verteilung zu bestimmen, kann die a-priori Gamma-Verteilung über die Parameter α und λ festgelegt werden. Es wird nun ein Modell trainiert, dass die Eingangsdaten inkl. einer Toleranz *tol* bestmöglich abbildet. Im Rahmen dieser Arbeit werden folgende Parameter angepasst: Maximale Anzahl an Iteration (*n_iter*) und der Toleranz-Parameter. Die Veränderung der α- und λ-Parameter zeigte in ersten Analysen keinen nennenswerten Veränderung und werden deshalb auf der Standard-Implementierung belassen. Die Tabelle 5.7 zeigt die Einstellparameter des bayesschen Regressors und die in dieser Arbeit gewählten Ausprägungen.

RandomForest Regressor Der RandomForest Regressor bildet Vorhersagen auf Basis von Entscheidungsregeln in Form von Bäumen ab. Die Komplexität des zu erstellenden Modells hängt u. a. von der Tiefe des Baumes ab. Dabei wird die Anzahl der Bäume mit Hilfe des Parameters *n_estimators* festgelegt. Die Anzahl an Samples, die mindestens für ein Blatt

Tabelle 5.7: Vorstellung der Einstellparameter des bayesschen Regressors

Einstellparameter des bayesschen Regressors	Gewählte Ausprägungen
Maximale Anzahl an Iterationen (*n_iter*)	100 bis 1000
Toleranz-Parameter (*tol*)	10^{-3} bis 10^{0}

benötigt werden, wird durch den Parameter (*min_samples_leaf*) beschrieben. Die maximale Tiefe eines Baumes (*max_depth*) und die maximale Anzahl an zu verwendeten Merkmalen (*max_features*) zeigten in ersten Analysen keine nennenswerten Veränderungen und werden deshalb auf der Standard-Implementierung belassen. Der Parameter *min_samples_split* gibt an, wie viele Samples mindestens benötigt werden, damit ein interner Knoten gespaltet werden darf. Weiterhin kann die maximale Anzahl an Blattknoten durch *max_leaf_nodes* angegeben werden. Die Tabelle 5.8 zeigt die Einstellparameter des RandomForest Regressors und die in dieser Arbeit gewählten Ausprägungen.

Tabelle 5.8: Vorstellung der Einstellparameter des RandomForest Regressors

Einstellparameter des RandomForest Regressors	Gewählte Ausprägungen
Anzahl der Bäume (*n_estimators*)	10, 20, ..., 250
min. Anzahl an Samples pro Ast (*min_samples_split*)	2
max. Prädiktoranzahl (*max_features*)	auto
min. Anzahl an Samples pro Blatt (*min_samples_leaf*)	5, 10, ..., 100
max. Teilungsebenenanzahl (*max_depth*)	unbegrenzt
maximale Anzahl an Blattknoten (*max_leaf_nodes*)	unbegrenzt

k-Nearest Neighbor Regressor Der KNN-Regressor lernt die Umgebung eines betrachteten Datenpunkts und versucht durch einen Mehrheitsentscheid der benachbarten Datenpunkte Vorhersagen zu treffen. Die Anzahl der betrachteten Nachbarn wird durch den Parameter *n_neighbors* angegeben. Der Parameter *weights* gibt an, wie die einzelnen Informationen der Nachbarn gewichtet werden sollen. Standardmäßig werden die Daten gleich gewichtet. Diese Gewichtung kann aber auch auf Basis der Distanz durchgeführt werden. Mit dem Parameter p kann zwischen der Berechnungsmethoden der Distanz nach der Minkowski-Metrik unterschieden werden, dabei steht $p = 1$ für die Manhattan Distanz und $p = 2$ für die euklidische Distanz. Die Tabelle 5.9 zeigt die Einstellparameter des KNN-Regressors und die in dieser Arbeit gewählten Ausprägungen.

Neuronale Netz Das neuronale Netz ist durch seine Eingangsschicht (engl. *input layer*) und Ausgangsschicht charakterisiert. Dazwischen liegen die versteckten Schichten (engl. *hidden layer*). Für alle drei Schichten können Aktivierungsfunktionen und die Anzahl der Neuronen festgelegt werden. Dabei ist zu beachten, dass die Eingangsdimension durch die

Tabelle 5.9: Vorstellung der Einstellparameter des kNN Regressors

Einstellparameter des kNN Regressors	Gewählte Ausprägungen
Anzahl der Nachbarn (*n_neighbors*)	2 bis 70
Distanzberechnung (*p*)	1 bis 5
Gewichtung (*weights*)	uniform, distance

Merkmale der Eingangsdaten festgelegt wird. Weiterhin wird am Ausgang nur ein Neuron mit einer linearen Aktivierungsfunktion verwendet, da es sich um eine Regressionsanalyse handelt (vgl. [Alp10, S. 246]). Neben den genannten Parametern wird auch die Anzahl der versteckten Schichten (*num_hiddenlayer*) angegeben. Die Batch-Größe gibt an, wie viele Samples zur Anpassung der Gewichte der Neuronen genutzt werden. Der Durchlauf eines vollständigen Trainingsdatensatzes wird als Epoche bezeichnet. Unterschiedliche Optimierungsalgorithmen (*optimizer*) bestimmen, wie die Gewichte im Lernprozess angepasst und aktualisiert werden. Dabei bestimmt die Lernrate, wie die einzelnen Gewichte im Laufe des Lernprozesses zur Minimierung des Fehlers variiert werden. Die Tabelle 5.10 zeigt die Einstellparameter des NNs und die in dieser Arbeit gewählten Ausprägungen.

Tabelle 5.10: Vorstellung der Einstellparameter des neuronalen Netzes

Einstellparameter des neuronalen Netzes	Gewählte Ausprägungen
Anzahl Neuronen am Eingang (*input_units*)	32, 64, 128, 256, 512
Aktivierungsfunktion am Eingang (*input_activation*)	sigmoid, relu, tanh
Dimension am Eingang (*input_dim*)	*datenabhängig*
Anzahl der versteckten Schichten (*num_hiddenlayer*)	2 bis 5
Anzahl Neuronen (*hidden_units*)	32, 64, 128, 256, 512
Aktivierungsfunktion (*hidden_activation*)	sigmoid, relu, tanh
Anzahl Neuronen am Ausgang (*output_unit*)	1
Aktivierungsfunktion am Ausgang (*outlayer_activation*)	linear
Optimierungsalgorithmus (*optimizer*)	Rprop, RMSProp, Adam, AdaGrad
Zahl der Epochen (*epochs*)	100, 200, ..., 1000
Batch-Größe (*batch_size*)	50, 100, 150, 250, 500

5.4 Vorstellung der Ergebnisse

Im weiteren Verlauf werden die Ergebnisse der beiden Ansätze vorgestellt. Dabei werden die zuvor genannten Einstellmöglichkeiten der Hyperparameter der Vorverarbeitung und des Data-Minings verwendet, um so mit Hilfe der Optimierung eine geeignete Konfiguration an Hyperparametern zu finden. In beiden Ansätze wird durch Anwendung des Kreuzvalidierungsverfahrens (vgl. Abschnitt 4.5) jede Konfiguration anhand des RMSEs bewertet. Im Rahmen der Hyperparameteroptimierung kann so nach mehreren Iterationen eine geeignete Konfiguration an Hyperparametern gefunden werden. Im weiteren Verlauf werden die Einflüsse der Parameter der Vorverarbeitung (vgl. Abschnitt 5.4.1) und des Data-Minings (vgl. Abschnitt 5.4.2) beschrieben.
Mit Hilfe der gefundenen Konfiguration an Hyperparametern wird im Abschnitt 5.5 eine finale Evaluierung durchgeführt, so wie es im Konzept in Kapitel 4.5 skizziert worden ist. Für diese Evaluierung wird ein Datensatz eines bis dahin unbekannten Fahrzeuges benutzt. Dieser Datensatz wurde auch nicht im Rahmen der Kreuzvalidierung benutzt.

Die Abbildung 5.7 stellt die Vorhersagegüten der einzelnen Modelle, die aus den jeweiligen Konfigurationen der datengetriebenen Hyperparameteroptimierung erzeugt wurden, dar. Als Vorhersagegüte wird der RMSE verwendet. Eine Konfiguration entspricht einer ausgewählten Hyperparameter-Einstellung aus den Parameter der Vorverarbeitung und des Data-Minings. Der gemittelte RMSE wird für jede Konfiguration durch die bereits beschriebene Kreuzvalidierung bestimmt. Im Verlauf der Iterationen ist zu sehen, dass der minimale

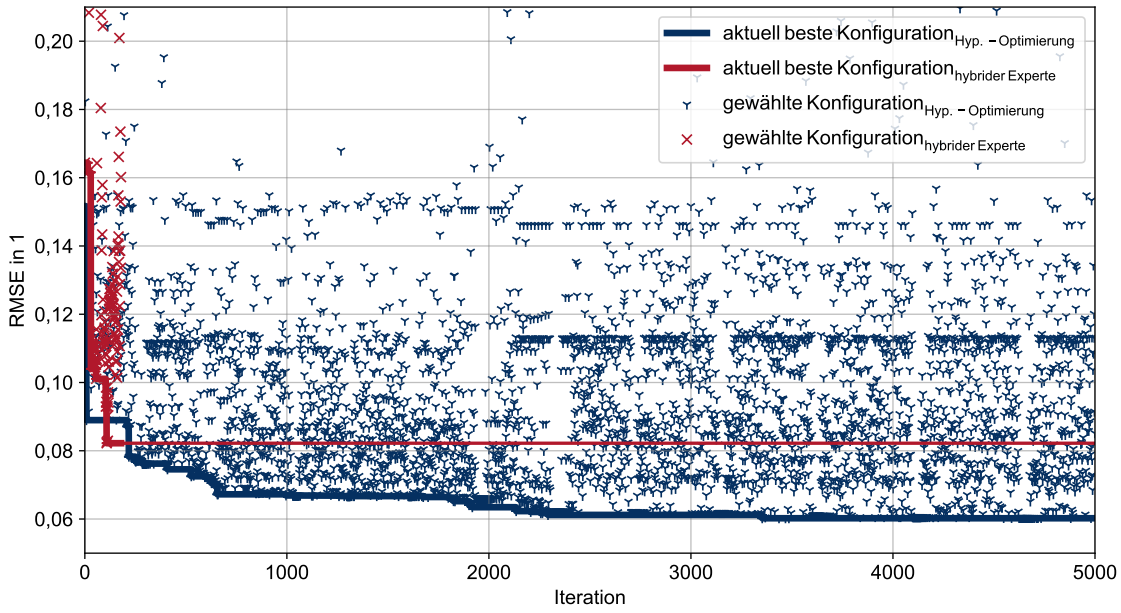


Abbildung 5.7: Darstellung der resultierenden Vorhersagegüten aus den einzelnen Konfigurationen der Hyperparameteroptimierung und des hybriden Expertenansatzes ohne Optimierung; dargestellt ist der gemittelte RMSE als Vorhersagegüte

RMSE stetig geringer wird. Die Aufgabe des Optimierungsalgorithmus ist die Bestimmung einer Konfiguration mit der höchsten Vorhersagegüte. Dafür werden im Verlauf unterschiedliche vielversprechende Hyperparameterkonfigurationen zur Modellerstellung der Alterung ausgewählt.
Die Tabelle 5.11 zeigt die minimale Vorhersagegüte und das Laufzeitverhalten der datengetriebenen Hyperparameteroptimierung. Zum besseren Verständnis ist neben dem minimalen mittleren quadratischen Fehler ($RMSE_{min}$) auch der minimale durchschnittliche absolute prozentuale Prognosefehler ($MAPE_{min}$) angegeben. Insgesamt benötigte der Optimierungs-

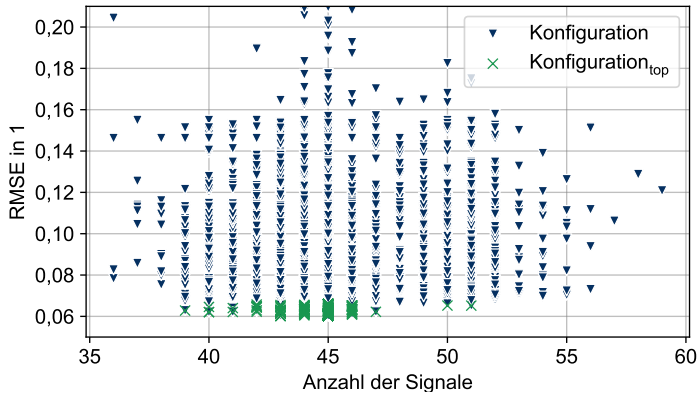
Tabelle 5.11: Tabellarische Darstellung der besten Vorhersagegüte und des gesamten Laufzeitverhaltens der datengetriebenen Hyperparameteroptimierung

$RMSE_{min}$	$MAPE_{min}$	Gesamtes Laufzeitverhalten
0,0602	8,35 %	108 Stunden

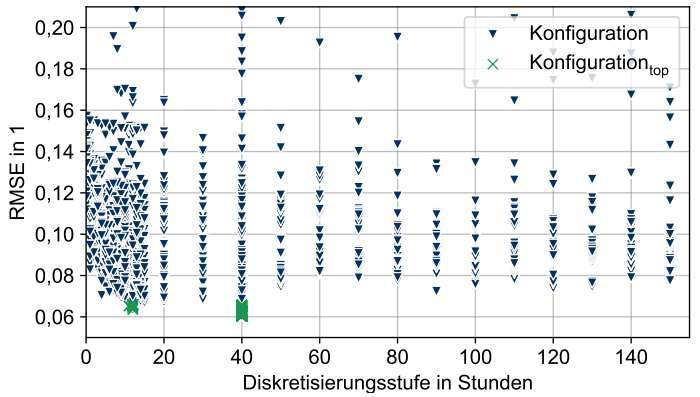
algorithmus unter Anwendung des Kreuzvalidierungsverfahrens bei 5000 Iterationen eine von der verwendeten Hardware abhängige Berechnungszeit von circa 108 Stunden. Diese Zeit beinhaltet sowohl das Laden der Eingangsdaten, die Vorverarbeitung der Eingangsdaten, sowie die Erstellung des datengetriebenen Modells als auch die Prädiktion der Vorhersagewerte. Im Rahmen des Optimierungsansatzes konnte eine Konfiguration an Hyperparametern ermittelt werden, bei denen sich ein minimaler Vorhersagefehler von $RMSE_{min} = 0,0620$ einstellt. Die zu dieser Konfiguration zugehörigen modellierten Alterungswerte sind in der Abbildung A.7e dargestellt. Außerdem zeigt die Abbildung A.7a-f auch für die anderen ML-Methoden die jeweils besten Prädiktion im Rahmen dieser Untersuchung.

5.4.1 Einfluss der Parameter der Vorverarbeitung

Im folgenden Abschnitt wird der Einfluss bestimmter Hyperparameter der Vorverarbeitung auf die Ergebnisse diskutiert. Dazu werden die Ergebnisse der zuvor vorgestellten datengetriebenen Hyperparameteroptimierung verwendet.
Die Abbildung 5.8a stellt die Ergebnisse der Optimierung der gewählten Anzahl an Signalen gegenüber. Vom Algorithmus werden unterschiedliche Konfigurationen an Hyperparametern gewählt, zu jeder dieser Konfiguration entsteht eine Vorhersagegüte (dargestellt als RMSE). Die besten 10% aller Konfigurationen ($Konfiguration_{top}$) werden in der Darstellung hervorgehoben und können mit den restlichen Konfigurationen verglichen werden. Es ist zu sehen, dass eine Anzahl an Signalen zwischen 39 und 51 vergleichsweise gute Ergebnisse erzielt. Eine größere und kleinere Anzahl an Signalen lässt die Vorhersagegüte im Rahmen dieser Untersuchung schlechter werden. Bereits frühere Vorarbeiten haben gezeigt, dass durch Auswahl unterschiedliche Signale unterschiedliche gute Vorhersageergebnisse prädiziert werden können [SEI19]. Die am häufigsten ausgewählten Signale im Rahmen der Hyperparameteroptimierung zeigen die Abbildungen A.9 und A.10 im Anhang.
Die Abbildung 5.8b stellt die Ergebnisse der Optimierung so dar, dass jedem Vorhersageergebnis eine entsprechende Diskretisierungsstufe zugeordnet wird. Dazu werden die Ergebnisse besonders hervorgehoben, die zu den 10% der besten Ergebnisse im Rahmen der



(a) Darstellung der Vorhersagegüten aus der Hyperparameteroptimierung unter Anwendung des gemittelten RMSEs als Vorhersagegüte bezogen auf die Anzahl der Signale; die besten 10% der Konfigurationen sind hervorgehoben



(b) Darstellung der Vorhersagegüten aus der Hyperparameteroptimierung unter Anwendung des gemittelten RMSEs als Vorhersagegüte bezogen auf die verwendete Diskretisierungsstufe; gezeigter Ausschnitt: Diskretisierungsstufe von 10 Minuten bis 150 Stunden; die besten 10% der Konfigurationen sind hervorgehoben

Abbildung 5.8: Darstellung der Vorhersagegüten aus der Hyperparameteroptimierung bei Veränderung der Signalmenge und der Diskretisierungsstufe

Optimierung gehören. Die beste Vorhersagegüte stellt sich im Rahmen dieser Untersuchung bei einer Diskretisierungsstufe von 40 Stunden ein. Es ist zu sehen, dass bei sehr feinen Diskretisierungsstufen die Vorhersagegüte unter den verwendeten Hyperparametern abnimmt. Auch eine sehr grobe Diskretisierungsstufe führt zu schlechteren Prognosen als Diskretisierungsstufen des mittleren Niveaus. Dennoch zeigt sich, dass sich bei Verwendung gröberer Diskretisierungsstufen vergleichsweise ähnlich gute Ergebnisse erzielen lassen. Erste Untersuchungen zu einem Einfluss der Diskretisierungsstufe auf das Vorhersageergebnis sind bereits in den Vorarbeiten [SEI20] veröffentlicht worden.
Auch die Wahl der Teilmenge an Merkmalen hat einen Einfluss auf die Vorhersageergebnisse. Dazu soll zunächst die Anzahl der pro Konfiguration und pro Signal verwendeten

Merkmale dargestellt werden. Die Abbildung 5.9 stellt dar, wie oft eine bestimmte Anzahl an Merkmalen in allen Konfigurationen auftritt. Weiterhin wird in dieser Abbildung zwischen den Ergebnissen von allen Durchläufen und zwischen denjenigen, die die besten 10% der Ergebnisse darstellen, unterschieden. Dabei fällt auf, dass vergleichsweise selten mehr

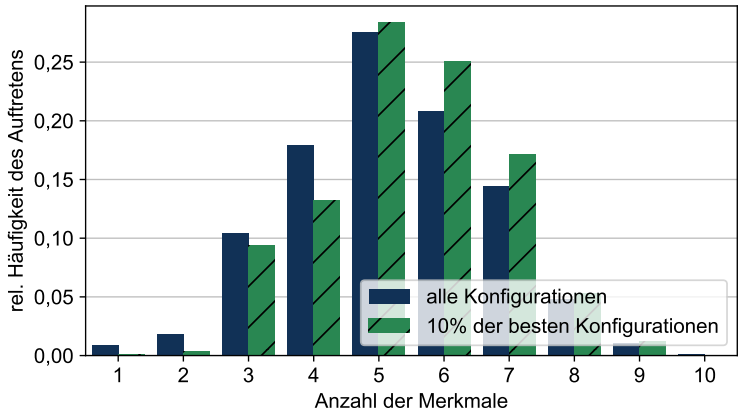


Abbildung 5.9: Darstellung der relativen Auftrittshäufigkeiten der Anzahl der Merkmale, die vom Optimierungsalgorithmus benutzt wurden; Vergleich zwischen allen Durchläufen und den Durchläufen der besten 10%

als sieben Merkmale verwendet werden. Eine eindeutige Verschiebung der Häufigkeitsverteilung der Anzahl der Merkmale ist mit Blick auf die besser performanten Konfigurationen nicht zu erkennen. So werden bei den 10% besten Konfigurationen zwar fünf Merkmale häufiger benutzt, aber die relative Auftrittshäufigkeit von vier Merkmalen nimmt ab. Weiterhin ist eine deutliche Verschiebung zugunsten von sechs bzw. sieben Merkmalen im Rahmen der besseren Konfigurationen zu erkennen.
Im Anhang zeigt die Abbildung A.11 die relative Verteilung der ausgewählten Merkmale im Verlauf aller Durchläufe der Optimierung. Die absolute Anzahl der Merkmale ist auch von der absoluten Anzahl der zu verwendenden Signale abhängig. Aus diesem Grund wird hier die relative Häufigkeit verwendet. Auch in dieser Abbildung sind die 10% der besten Konfigurationen hervorgehoben. Eine eindeutige Verschiebung der Häufigkeitsverteilung ist mit Blick auf den gewählten Merkmalen nicht zu erkennen.

5.4.2 Einfluss der Parameter des Data-Minings

Im folgenden Abschnitt wird der Einfluss aufgrund der Wahl der ML-Methode auf die Ergebnisse vorgestellt. Dazu zeigt die Abbildung 5.10 die Ergebnisse der Optimierung, wobei diese nach der Verwendung der ML-Methode sortiert sind. Erste Untersuchungen zu einem Einfluss der Wahl der ML-Methode auf das Vorhersageergebnis sind bereits in der Vorarbeit [SEI20] veröffentlicht worden. Es ist zu sehen, dass sowohl die MLR als auch der Bayes-Regressor vergleichsweise keine besonders hohe Vorhersagegüten zeigen. Die beste Vorhersagegüte ist mit Hilfe des RMSEs im Rahmen dieser Untersuchung bestimmt worden. Wie auch bereits im hybriden Expertenansatz gezeigt werden konnte (vgl. Kapitel 5.2.2), weisen

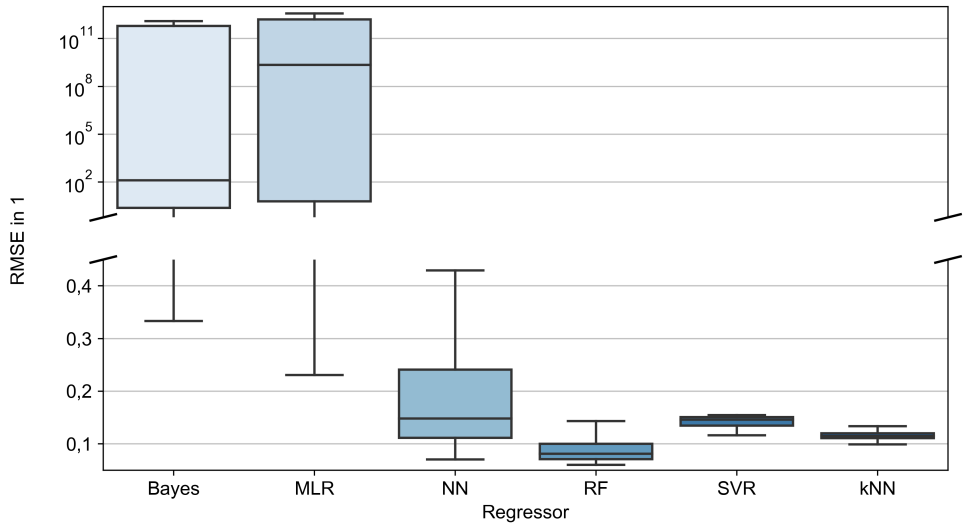


Abbildung 5.10: Darstellung der Vorhersagegüten aus der Hyperparameteroptimierung unter Anwendung des gemittelten RMSEs als Vorhersagegüte, sortiert nach der Verwendung der ML-Methode

der SVR- und der kNN-Regressor eine geringere Streuung der Ergebnisse auf. Die zuletzt genannten Regressoren reagieren weniger sensitiv auf eine Veränderung der Hyperparameter der Vorverarbeitung und des Data-Minings.
Die nachfolgende Abbildung 5.11 soll einen weiteren Einblick in die bereits gezeigten Ergebnisse liefern. Hier werden die Prognosegüten der einzelnen Konfigurationen aus der Hyperparameteroptimierung dargestellt. Die unterschiedlich gewählten Methoden des MLs werden dabei entsprechend abgebildet.

5.5 Diskussion und Potentialabschätzung

In den vorherigen Abschnitten sind die Ergebnisse der datengetriebenen Optimierung und deren Einflussfaktoren dargestellt. Im Rahmen der Optimierung sind unterschiedliche Hyperparameter eingestellt worden, die eine entsprechende Vorhersagegüte zur Folge haben. Die Hyperparameter, die sich im Rahmen des Kreuzvalidierungsverfahrens dieser Arbeit als geeignet erwiesen haben, sind in Tabelle 5.12 dargestellt. Dazu wurde der Optimierungsalgorithmus für einen Durchlauf mit 5000 Iterationen eingestellt. Der Verlauf der Vorhersagegüten der einzelnen Konfigurationen ist bereits in Abbildung 5.7 gezeigt worden. Im Rahmen der finalen Evaluierung (vgl. Abbildung 4.11 aus dem Kapitel 4.5) wird ein bis hierher vollständig unbekannter Datensatz zur Vorhersage der Alterungswerte benutzt. In der vorherigen datengetriebenen Optimierung ist eine Konfiguration an Hyperparametern gefunden worden, die unter Anwendung des Kreuzvalidierungsverfahrens die in dieser Arbeit vorliegenden Alterung am besten vorhersagen kann. Das daraus resultierende Modell

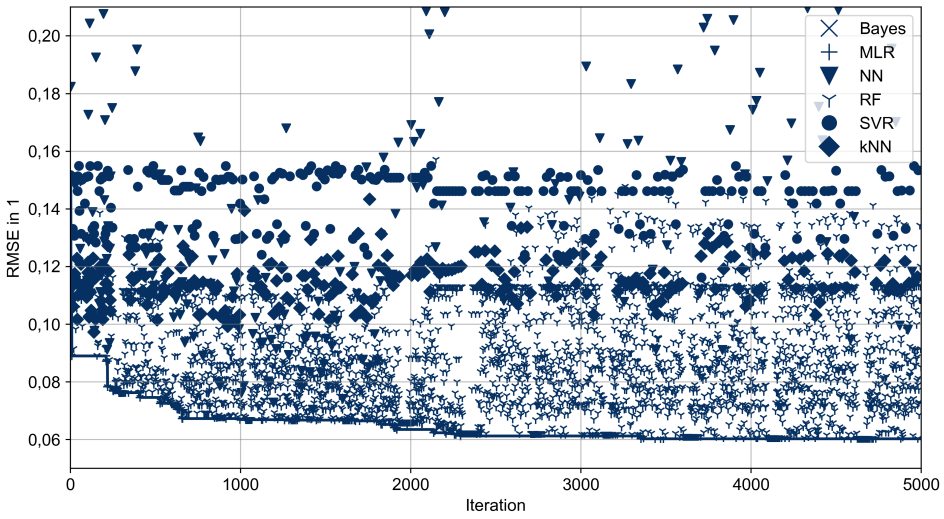


Abbildung 5.11: Darstellung der resultierenden Vorhersagegüten aus den einzelnen Konfigurationen der Hyperparameteroptimierung und des hybriden Expertenansatzes. dargestellt ist der gemittelte RMSE als Vorhersagegüte, gewählte Methoden des MLs sind entsprechend abgebildet

Tabelle 5.12: Auflistung der durch die datengetriebene Optimierung festgelegten besten Hyperparameter nach 5000 Iterationen unter Anwendung des Kreuzvalidierungsverfahrens

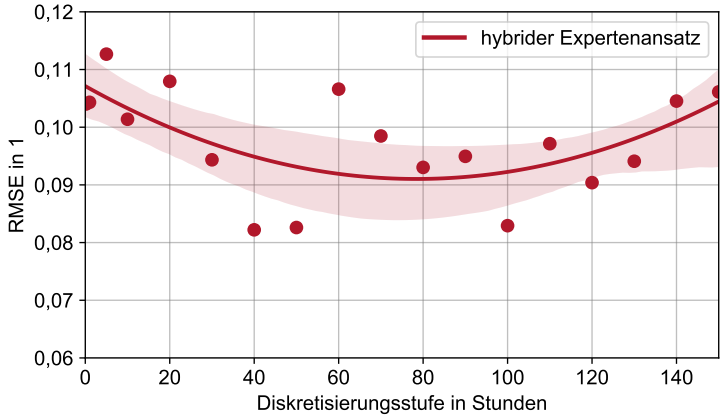
Hyperparameter der Vorverarbeitung	**Ausprägung**
Auswahl einer Teilmenge an Signalen	43 Signale
Diskretisierungsstufe	40 Stunden
Auswahl einer Teilmenge an Merkmalen	*signalindividuell*

Hyperparameter des Data-Minings	**Ausprägung**
Auswahl einer ML-Methode	RF
Einstellparameter der ML-Methode	*min_samples_split*: 2, *min_samples_leaf*: 5, *n_estimators*: 20, *max_features*: auto

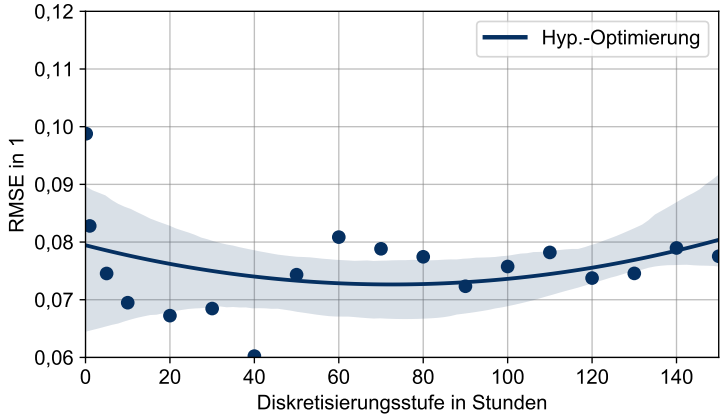
wird zur Alterungsvorhersage eines bislang unbekannten Fahrzeuges verwendet. Zum Training werden dabei sämtliche Daten der restlichen Fahrzeuge eingesetzt. Für die Vorhersage der Alterung bei dem vollständig unbekannten Datensatz stellt sich ein RMSE von 0,0639 (MAPE = 9,69 %) ein.

Die Tabelle 5.12 zeigt unter den gewählten Randbedingungen die beste Konfiguration an Hyperparametern für Vorhersage der untersuchten Alterung. Bei Betrachtung der Wahl der Teilmengen an Signalen stellt sich heraus, dass sich 43 Signale als geeignet erweisen. Das heißt

aber nicht, dass die hier untersuchte Alterung zwingend mit einer Auswahl von 43 Signalen erfolgen muss. Der hybride Expertenansatz hat bereits gezeigt, dass auch mit einer kleineren Menge an Signalen eine geeignete Vorhersage möglich ist. Mit Blick auf eine zukünftige mögliche Implementierung in einer Fahrzeug/Cloud-Umgebung ist kritisch zu hinterfragen, ob wirklich alle 43 Signale für eine Alterungsvorhersage zwingend erforderlich sind. Je nach zur Verfügung gestellter Menge und Variabilität an Eingangsdaten ist unter Einbeziehung der bisherigen Ergebnisse auch eine geeignete Vorhersage mit weniger Signalen denkbar. Die hier in Tabelle 5.12 dargestellten 43 Signale sind im Rahmen der in dieser Arbeit untersuchten Anwendungsfalls und unter genannten Randbedingungen entstanden.
Mit Hilfe der Optimierung kann für die Alterungsvorhersage eine beste Anzahl an Signalen nicht zweifelsfrei bestimmt werden kann. Im Rahmen der Hyperparameteroptimierung zeigen sich unterschiedliche Anzahlen an Signalen für die vorliegenden Alterungsvorhersage als geeignet. Dennoch können sich bei Betrachtung der relativen Häufigkeitsverteilung der verwendeten Signale über mehrere Iterationen wertvolle Hinweise ergeben, um so die Relevanz einzelner Signale für die untersuchte Alterungsvorhersage zu bewerten (vgl. im Anhang Abbildung A.9 und A.10). Diese Analyse kann den Entwicklern der Fachdomäne wichtige Informationen liefern und zu einem besseren Verständnis einer Komponentenalterung beitragen.
Die Ergebnisse aus der Tabelle 5.12 zeigen, dass eine Diskretisierungsstufe von 40 Stunden im Rahmen der Optimierung sich als geeignet erwiesen hat. Diese Beobachtung soll in einer weiteren Darstellung konkretisiert werden. In der Abbildung 5.12 ist der minimale Prognosefehler der beiden Ansätze bei unterschiedlichen Diskretisierungsstufen und der zugehörigen gemittelten Trend-Kennlinien dargestellt. Es ist jeweils die beste Vorhersagegüte einer jeweiligen Diskretisierungsstufe der beiden Ansätze dargestellt. Der Abbildung 5.12a des hybriden Expertenansatzes zeigt eine leichte Verschlechterung der Prognosegüte bei sehr feinen Diskretisierungsstufen. Eine größere Verschlechterung der Prognosegüte bei sehr feinen Diskretisierungsstufen ist dagegen in der Abbildung 5.12b zu erkennen.
Eine Erklärung kann darin begründet sein, dass sich in sehr feinen Diskretisierungsstufen die Fahrzeugzustände schnell ändern. Interne Fahrzeuggrößen reagieren sehr sensibel auf das Fahrverhalten des Fahrers. So scheint unter den genannten Rahmenbedingungen eine langfristige Alterung durch sehr feine Diskretisierungsstufen nicht besonders gut prognostizierbar zu sein. Dies kann auch durch eine veröffentlichte Vorarbeit [SEI20] bestätigt werden. Beide untersuchten Ansätze zeigen, dass gröbere Diskretisierungsstufen (etwa über 5h) bessere Ergebnisse liefern können. Wird die Diskretisierungsstufe zu grob (etwa über 120h), werden die Ergebnisse in beiden Ansätzen vergleichsweise schlechter.
Die im Rahmen der Konzeptvalidierung gezeigten Ergebnisse verwenden den RMSE als alleiniges Maß zur Beurteilung der Prognosegüte. Die Abbildung A.8 im Anhang zeigt unter Anwendung der jeweils besten Konfiguration an Hyperparametern im Rahmen des untersuchten Anwendungsbeispiels die gemessenen und modellierten Alterungswerte bei unterschiedlichen Diskretisierungsstufen. Das Prädiktionsergebnis mit dem besten RMSE ist in Abbildung A.8d dargestellt (n_{pred} = 155). Eine Veränderung der Diskretisierungsstufe führt zu anderen Prädiktionsergebnissen. Die Abbildung A.8c zeigt die zugehörigen Prädiktionsergebnisse der bestens Konfiguration bei einer Diskretisierungsstufe von 15 Stunden. Der für diese Diskretisierungsstufe zugehörige RMSE liegt bei 0,0669 und ist vergleichbar mit



(a) Darstellung der minimalen Prognosefehler der jeweiligen Diskretisierungsstufe des hybriden Expertenansatzes, zusätzlich ist eine Kurve der gemittelten Werte inkl. deren Konfidenzintervalle eingezeichnet



(b) Darstellung der minimalen Prognosefehler der jeweiligen Diskretisierungsstufe des datengetriebenen Optimierungsansatzes, zusätzlich ist eine Kurve der gemittelten Werte inkl. deren Konfidenzintervalle eingezeichnet

Abbildung 5.12: Darstellung der minimalen Prognosefehler der beiden Ansätze bei unterschiedlichen Diskretisierungsstufen

dem besten Ergebnis (RMSE_{min} = 0,0602). Bei dieser Diskretisierungsstufe von 15 Stunden liegen im Vergleich zur Konfiguration mit dem geringsten RMSE mehr als doppelt so viele Prädiktionswerte vor (n_{pred} = 420). Für zukünftige Anwendungen sollte das Maß der Prognosegüte mit Hilfe der Anzahl an Prädiktionswerten korrigiert werden. Dieses korrigierende Prognosegütemaß sollte die Anzahl an Prädiktionswerten so sensibel gewichten, dass eine sehr hohe Anzahl an Prädiktionswerten nicht zu einer verzerrten Darstellung der Güte führt.

Potentialabschätzung

Im folgenden Abschnitt werden zur Potentialabschätzung die beiden Ansätze, der hybride Expertenansatz und die datengetriebene Optimierung, miteinander verglichen. In den vorherigen Abschnitten konnte gezeigt werden, dass beide Ansätze grundsätzlich für eine Alterungsvorhersage geeignet sind. Die Tabelle 5.13 stellt den Vergleich beider Ansätze ta-

Tabelle 5.13: Vergleich der beiden Ansätze der Alterungsvorhersage zur Potentialabschätzung

	Ansatz 1	**Ansatz 2**
Beschreibung	hybrider Expertenansatz	datengetriebene Hyperparameteroptimierung
Genauigkeit	hoch	sehr hoch
$RMSE_{min}$	0,0822	0,0620
$MAPE_{min}$	11,69 %	8,35 %
Diskr.-$stufe_{min}$	40 h	40 h
Berechnungsaufwand	mäßig, ca. 4,6 h	hoch, ca. 108 h
Benötigtes Expertenwissen	hoch	mittel
Übertragbarkeit des Modells auf andere Anwendungsfälle	mittel	sehr hoch

bellarisch dar.
Zunächst werden die Vorhersageergebnisse im Rahmen der Kreuzvalidierung miteinander verglichen. Dabei erzielt der datengetriebene Ansatz einen minimalen RMSE im Rahmen der Kreuzvalidierung von 0,0602 bei einer Berechnungszeit von insgesamt circa 108 Stunden. Etwas schlechter ist der hybride Expertenansatz (RMSE = 0,0822) bei einer Berechnungszeit von circa 4,6 Stunden.
Für den Expertenansatz ist ein vergleichsweise großes Vorwissen über die zu untersuchende Alterung und Informationen über den Datenbestand notwendig. Für eine geeignete Auswahl relevanter Fahrzeugsignale müssen relevante Zusammenhänge bereits im Vorhinein bekannt sein. Ein Maß zur Beurteilung eines vollumfänglichen Ressourcenaufwandes von Wissen des Fachexperten und dessen zeitlichem Aufwand kann nicht bestimmt werden. Dies liegt darin begründet, dass ein Experte über die Jahre in der Entwicklungsphase auch notwendiges Wissen über die zu untersuchende Alterung ansammelt und dies zeitlich nicht erfasst werden kann.
Die Abbildung 5.13 zeigt die unterschiedlichen Prognosegüten bei verschiedenen Konfigurationen der beiden Ansätze als Boxplot-Darstellung ohne Ausreißer. Hierbei ist zu sehen, dass sich die oberen und unteren Quartile der Ergebnisdarstellung der Hyperparameteroptimierung auf einem niedrigeren Niveau als die Quartile des hybriden Expertenansatzes befinden. Auch der Median der Hyperparameteroptimierung liegt niedriger als der des hybriden Expertenansatzes. Dabei beinhaltet der hybride Expertenansatz 180 Iterationen und die Hyperparameteroptimierung 5000 Iterationen.
Dennoch sei an dieser Stelle darauf hingewiesen, dass einzelne Ausreißer-Konfigurationen an Hyperparametern des datengetriebenen Ansatzes auch schlechtere Modellgüten liefern

können, als in der Abbildung 5.13 gezeigt ist. Die hier gewählte Darstellung zeigt einen Ausschnitt als Boxplot-Darstellung. Aufgrund der großen Anzahl an Konfigurationen sind

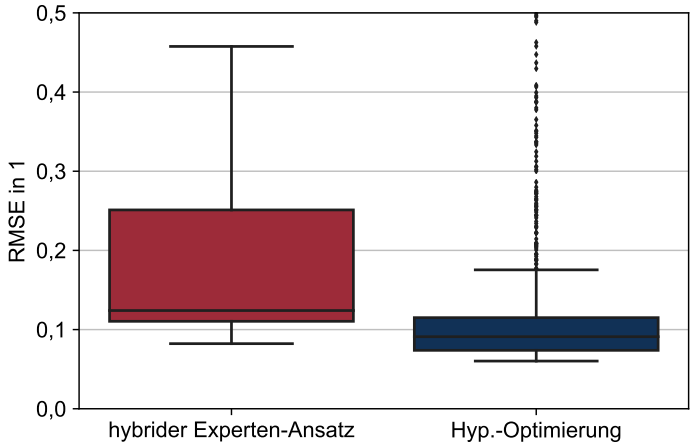


Abbildung 5.13: Darstellung der Vorhersagegüten der beiden Ansätze als Boxplot

im Rahmen der Hyperparameteroptimierung auch einzelne Ergebnisse zu erkennen, die deutlich schlechter als die Ergebnisse des hybriden Expertenansatzes sind (vgl. auch Abbildung 5.10). Allerdings sei an dieser Stelle angemerkt, dass letztendlich die Konfiguration an Hyperparameter gewählt werden soll, die den geringsten Prognosefehler aufweist. Beide vorgestellten Ansätze liefern Ergebnisse, die eine geeignete Vorhersage der untersuchten Alterung ermöglichen. Es ist festzustellen, dass der Optimierungsansatz durchschnittlich bessere Ergebnisse liefert als der hybride Expertenansatz, obwohl beim Optimierungsansatz ein Vielfaches an Iterationen durchgeführt worden sind (vgl. Abbildung 5.13). Außerdem ist zu sehen, dass nicht nur oberes und unteres Quartil des Optimierungsansatzes einen besseren RMSE aufweisen, sondern die Differenz der beiden Quartile auch im Vergleich zum hybriden Expertenansatz deutlich kleiner ist.
Neben diesen quantitativen Bewertungen der Prognosegüte sollen an dieser Stelle auch qualitative Bewertungen genannt sein. Ein großer Vorteil der datengetriebenen Optimierung ist die Identifizierung der Hyperparameter, die zu einer wesentlichen Verbesserung des Ergebnisses beitragen. Im Rahmen der Einflussanalyse kann dem Fachexperten mit Hilfe der Abbildung A.10 (s. Anhang) gezeigt werden, welche Signale in der Datenmenge einen entscheidenden Einfluss auf das Ergebnis hatten. Diese Einflussanalyse kann durch die Identifikation wichtiger Signale zu einem besseren Verständnis der Alterung in der Fachdomäne beitragen.
Mit Hilfe des gezeigten datengetriebenen Ansatzes kann ein besseres Verständnis einer von Experten der Fachdomäne nicht erklärbaren Alterung erzielt werden. Für ein solches Vorgehen sind regelmäßige bewertende Messungen der zu untersuchenden Zielgröße und die Aufnahme entsprechender Messgrößen notwendig.

In Bezug auf die Hypothese **H 1** konnte anhand des Beispiels der AGR-Kühlerversottung aus dem Bereich der Automobilindustrie gezeigt werden (vgl. Abbildung 5.7 und Tabelle 5.13), dass sich die Ergebnisse der Vorhersage der Fachexperten mit Hilfe einer datengetriebenen Hyperparameteroptimierung unter Anwendung der vorgestellten Methoden des MLs verbessern lässt. Im Rahmen der datengetriebenen Optimierung werden deutlich mehr Signale verwendet. Außerdem ist die Wahl der Merkmale nicht unveränderlich, sondern wird für jedes Signal individuell bestimmt. Aus den gezeigten Einflussanalysen der datengetriebenen Optimierung lassen sich daraus neben der Prognosegüte weitere für die Fachdomäne relevante Erkenntnisse ableiten.
Die in diesem Abschnitt gezeigten Einflussanalysen zeigen eine Beeinflussung durch die Wahl der Diskretisierungsstufe (vgl. Hypothese **H 2**, Kapitel 5.1.2). Die besten Vorhersagegüten des hybriden Expertenansatzes und der datengetriebenen Hyperparameteroptimierung stellen sich bei einer Diskretisierungsstufe von 40 Stunden für diesen Anwendungsfall ein. Es zeigt sich, dass sich die Wahl der Diskretisierungsstufe der in dieser Arbeit untersuchten Alterung bei den zur Verfügung stehenden Daten und den genannten Rahmenbedingungen nicht beliebig fein wählen lässt (vgl. Abbildung 5.4 und Abbildung 5.8). Dieser Zusammenhang ist auch in der Abbildung 5.12 beobachtbar.

Zusammenfassung der Konzeptvalidierung

Zusammenfassend wurde in diesem Kapitel der bereits zuvor vorgestellte Konzeptentwurf mittels eines konkreten Alterungsbeispiels aus der Automobilindustrie validiert. Mit Hilfe von Clusterverfahren konnte zunächst eine Reduktion der Signale vorgenommen werden. Im weiteren Verlauf wurden Merkmale der Signale anhand von unterschiedlichen Diskretisierungsstufen berechnet. Die Ergebnisse in diesem Kapitel zeigen, dass eine Veränderung der Diskretisierungsstufe eine Auswirkung auf die Vorhersagequalität hat. Sowohl im hybriden Expertenansatz als auch in der datengetriebenen Hyperparameteroptimierung konnte gezeigt werden, dass vor allem eine für den Vorhersagehorizont zu fein gewählte Diskretisierungsstufe vergleichsweise schlechtere Ergebnisse liefert. Eine besonders große Diskretisierungsstufe zeigte dagegen nur minimal schlechtere Ergebnisse als das Optimum. Dennoch hat eine gröbere Diskretisierungsstufe den Nachteil, dass eine Alterungsprädiktion erst nach einer größeren Zeitspanne möglich ist.
Die Ergebnisse zeigen, dass nicht alle vorgestellten Merkmale für jedes Signal notwendig sind. Es ist nicht auszuschließen, dass relevante Merkmale signalindividuell auftreten. Durchschnittlich wurde nur eine Teilmenge an Merkmalen zur Modellerstellung genutzt (vgl. Abbildung 5.9). Außerdem hat sich gezeigt, dass einige Merkmale, beispielsweise seien hier der Mittelwert (engl. *mean*) und Median genannt, ähnliche Informationen enthalten können (vgl. Abbildung 5.2). Eine hohe Korrelation zwischen zwei Merkmalen kann nur minimal zu einer Verbesserung des Modells beitragen. Allerdings kann im Vorhinein ohne ein umfassendes Verständnis der physikalischen Alterung kein Merkmal ausgeschlossen werden.
Die in dieser Arbeit vorgestellten Ergebnisse zeigen, dass sich nicht alle beschriebenen Lernalgorithmen unter den gegebenen Randbedingungen eignen, eine Vorhersage der in dieser Arbeit untersuchten Abnutzungserscheinung vorzunehmen. Also besonders geeignet erwies

sich der RF-Regressor zur Vorhersage der Abnutzungserscheinung unter den gegebenen Randbedingungen, sowohl im hybriden Expertenansatz als auch im Rahmen der datengetriebenen Hyperparameteroptimierung.

6 Zusammenfassung und Ausblick

In diesem Kapitel werden die in dieser Arbeit vorgestellten Beschreibungen, Konzepte und Validierungen zusammenfassend reflektiert. Weiterhin werden eine mögliche Übertragbarkeit des Konzepts im Rahmen einer Fahrzeug/Cloud-Umgebung und weitere Anwendungen vorgestellt (vgl. Kapitel 6.2). Im Ausblick werden (vgl. Kapitel 6.3) zukünftige Forschungsschwerpunkte und weiterführende Fragestellungen diskutiert.

6.1 Zusammenfassung

In dieser Arbeit wurde eine langfristige Fahrzeugzustandsänderung anhand virtueller datengetriebener Sensormodelle untersucht. Die Vorstellung notwendiger Grundlagen und Vorbetrachtungen erfolgte in Kapitel 2. Dabei wurde zwischen einer physikalischen und datengetriebenen Modellierung unterschieden (vgl. Kapitel 2.1.1). Weitere für die datengetriebene Modellierung notwendigen Begrifflichkeiten wurden beschrieben. Darauf aufbauend wurden Methoden des maschinellem Lernens (vgl. Kapitel 2.1.3) und Kenngrößen des Zusammenhangs, sowie Prognosegütemaße (vgl. Kapitel 2.1.4) erläutert.
Im weiteren Verlauf wurde in Kapitel 3 das Problem bezüglich des allgemeinen Ausfallverhaltens sowie der Restnutzungsdauer technischer Systeme und zwischen Drift- und Sprungausfall differenziert. Ein physikalisches Modell der in dieser Arbeit vorliegenden Komponentenzustandsänderung ist nicht vorhanden. Stattdessen wurden datengetriebene Ansätze mit Hilfe multivariater Analysemethoden zur Zustandsvorhersage eingesetzt (vgl. Kapitel 2.1.1 und 2.1.3).
Zur effizienten Modellbildung werden Merkmale extrahiert und selektiert (vgl. Kapitel 3.2). Verschiedene Arbeiten zum aktuellen Stand der Wissenschaft bezüglich Vorhersagen für technische Systeme auf Basis datengetriebener Ansätze wurden vorgestellt (vgl. Kapitel 3.3). Die in dieser Arbeit untersuchten Alterung ist durch einen langfristigen Vorhersagehorizont und eine Vielzahl verfügbarer Signale gekennzeichnet (vgl. Kapitel 3.4). Der Zielkonflikt zwischen einer zeitnahen Alterungsvorhersage und der wahrnehmbaren Veränderung des Alterungswertes ist in Kapitel 3.4.2 beschrieben worden. Weiterhin wurden konkrete Herausforderungen einer langfristigen Alterungsvorhersage beim Vorhandensein hochdynamischer Eingangsdaten charakterisiert und Lösungsmöglichkeiten dargelegt (vgl. Kapitel 3.7).
Aus dem in der Problembeschreibung vorgestellten Stand der Wissenschaft und Technik wurden im Konzeptentwurf (vgl. Kapitel 4) Verfahren zur Aggregation der Daten abgeleitet. Der Prozess der Knowledge Discovery in Databases (KDD) bildet das grundsätzliche Vorgehen zur Erstellung eines virtuellen Sensors zur Zustandsänderung einer Komponente. Im weiteren Verlauf folgte die Konkretisierung der Datenvorverarbeitung in Kapitel 4.3 und des Data-Minings im Rahmen des Konzeptentwurfs (vgl. Kapitel 4.4). Die Datenvorverarbeitung zeichnet sich durch die Wahl der Diskretisierungsstufe und die Wahl der Merkmale aus. Im Abschnitt des Data-Minings wurden konkrete Methoden des MLs aus den zuvor

zitierten Arbeiten vorgestellt. Diese einstellbaren Hyperparameter zur Erstellung eines virtuellen Sensormodells wurden am Ende des Kapitels zusammengefasst (vgl. Kapitel 4.7).
Im Anschluss wurde das beschriebene Konzept anhand eines Beispiels einer Abgasrückführung-Kühlerversottung aus dem Bereich der Automobilindustrie validiert. Es folgte eine Skizzierung zweier unterschiedlicher Ansätze mit anschließender Diskussion möglicher Vor- und Nachteile in Kapitel 5. Die analysierten hochdynamischen Zeitreihen stammen von unterschiedlichen Fahrzeugen und einem Aufzeichnungszeitraum über mehrere Monate (vgl. Kapitel 5.1). Neben diesen Daten standen auch Performance-Messungen der untersuchten Komponente zur Verfügung, die in unregelmäßigen Abständen in den Werkstätten gemessen wurden. Für die Gewährleistung einer effizienten Analyse wurden in dieser Arbeit verschiedene Verfahren zur Reduktion der Vielzahl an Eingangsdaten gezeigt. Mit Hilfe dieser vollumfänglichen datengetriebenen Analysen lassen sich auf den vorliegenden Anwendungsfall bezogen unter Anwendung geeigneter Einstellparameter vergleichsweise gute Vorhersagegüten erzielen (vgl. Abschnitt 5.3). Doch neben diesen quantitativen Bewertungen sind vor allem im Bereich der Fachdomäne auch qualitative Analysen gewünscht (vgl. Kapitel 5.4). Die in dieser Arbeit gezeigten Einflussanalysen der Signale liefern hierzu hilfreiche Informationen, die zu einem besseren Verständnis einer unbekannten Alterung einer Komponente führen (vgl. Kapitel 5.5).

In dieser Arbeit wurden drei Forschungsfragen (**RQ 1** bis **RQ 3**) aufgestellt (vgl. Kapitel 3.6) und in unterschiedlichen Kapiteln adressiert:

- **RQ 1** „*Wie kann ein virtueller Sensor unter Berücksichtigung unterschiedlicher zeitlicher Auflösungen zwischen hochdynamischen Eingangsdaten und einer Zielgröße datenbasiert übertragbar modelliert werden?*“ wird in Kapitel 4.3 und zu Teilen in Kapitel 4.4 aufgegriffen und adressiert die unterschiedlichen Auflösungen zwischen den Eingangsdaten und den untersuchten langfristigen Komponentenzustandsänderungen. Nach entsprechender Datenvorverarbeitung werden innerhalb gewählter Diskretisierungsstufen statistische Merkmale von den hochdynamischen Eingangsdaten extrahiert. Zusammen mit den Daten der Zielgröße werden unter Zuhilfenahme der Methoden des Data-Minings virtuelle Sensormodelle erstellt.
- **RQ 2** „*Wie kann dieses datenbasierte Sensormodell hinsichtlich Genauigkeit und der genutzten Datenmenge optimiert werden?*“ wird in den Kapiteln 3.7.3, 4.4, 4.5 und 4.6 aufgegriffen und spiegelt die Fragestellung nach einer möglichen Optimierung wider. Im Rahmen der Problemkomplexität werden entsprechende Lösungsmöglichkeiten dieser Optimierungsaufgabe diskutiert (vgl. Kapitel 3.7.3). Im weiteren Verlauf wurden unterschiedliche Implementierungen der bayesschen Optimierung verglichen und die einstellbaren Hyperparameter der Vorverarbeitung und des Data-Minings benannt (vgl. Kapitel 4.5). In Kapitel 5.4 wird diese Hyperparameteroptimierung mit einem hybriden Expertenansatz verglichen.
- **RQ 3** „*Wie kann die Qualität dieses virtuellen Sensors anhand der AGR-Kühler-Alterung validiert werden?*“ reflektiert die Validierung eines virtuellen Sensormodells anhand eines konkreten Anwendungsfalls aus der Automobilindustrie. Die Validierung der Ergebnisse wird in Kapitel 5 diskutiert. Es wird ein hybrider Expertenansatz (vgl. Kapitel 5.2) und der

Ansatz der datengetriebenen Hyperparameteroptimierung (vgl. Kapitel 5.3) vorgestellt. Im weiteren Verlauf wird mit Hilfe von qualitativen Einflussanalysen die Qualität des virtuellen Sensors validiert (vgl. Kapitel 5.4).

6.2 Übertragbarkeit des Konzepts

In Kapitel 4 wird das Konzept zur Erstellung eines virtuellen Sensormodells zur langfristigen Komponentenzustandsbestimmung dargestellt. Mit Hilfe eines Beispiels aus der Automobilindustrie wird dieses Konzept unter Anwendung von hochdynamischen CAN-Daten validiert. Die Ergebnisse aus Kapitel 5 zeigen, dass langfristige Komponentenzustandsänderungen auch für Fahrzeuge vorhergesagt werden können, deren Messdaten nicht Teil der gelernten Eingangsdaten waren. Die in dieser Arbeit erstellten Modelle ermöglichen demnach auch Komponentenzustandsvorhersagen für Fahrzeuge, deren Komponentenzustände nicht aufwendig in Werkstätten vermessen worden sind. Aus diesen Erkenntnissen ist eine Anwendung der erstellten Modelle auch für unbekannte (Kunden-) Fahrzeuge denkbar. So können bereits im frühen Entwicklungsprozess aufwendigen Komponentenzustandsmessungen an einzelnen Fahrzeugprototypen durchgeführt werden, um diese Daten und Modelle für eine größere Anzahl (unbekannter) Fahrzeuge nutzen zu können. Die Abbildung 6.1 zeigt eine solche Übertragbarkeit konzeptionell.
Die Fahrzeuge der Klasse F0 (vgl. Abbildung 6.1) zeichnen hochdynamische Daten auf und unterliegen regelmäßigen aufwendigen Komponentenzustandsmessungen. Mit Hilfe der Da-

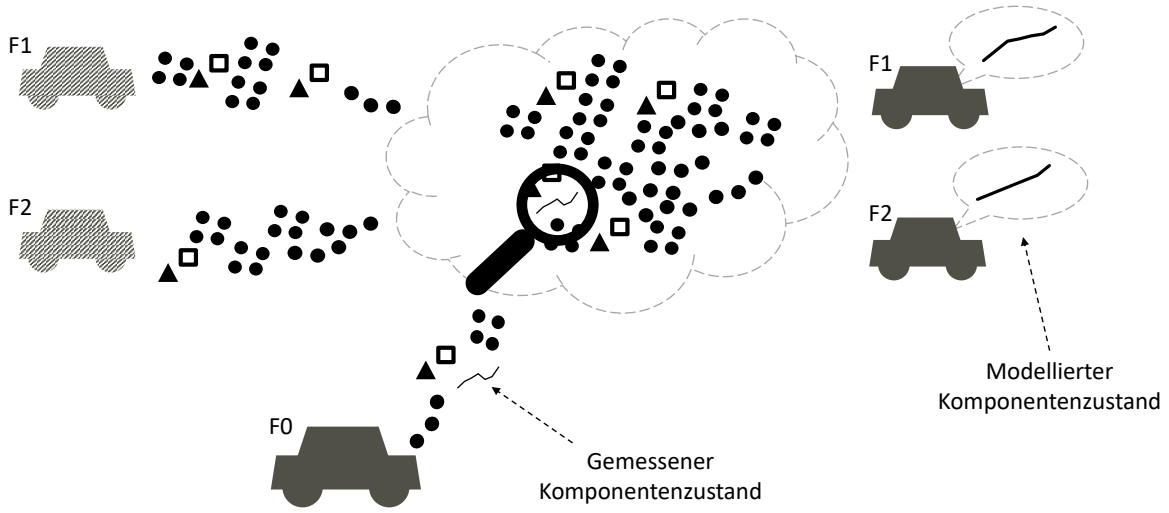


Abbildung 6.1: Konzeptionelle Darstellung einer virtuellen Sensormodellerstellung und einer möglichen Übertragbarkeit auf unbekannte Fahrzeuge

ten der Fahrzeuge der Klasse F0, den Informationen über die Komponentenzustandsänderung und ML-Algorithmen wird so ein virtuelles Sensormodell erstellt, welches die Komponentenzustandsänderung für eine gegebene Datenmenge vorhersagt. Liegen von unbekannten Fahrzeugen auch solche hochdynamischen Daten (z.b. CAN-Busdaten) vor, kann mit Hilfe der erstellten virtuellen Sensormodelle der Komponentenzustand für Fahrzeuge der

Klassen F1 und F2 (vgl. Abbildung 6.1) bestimmt werden. An dieser Stelle sei erwähnt, dass eine solche Anwendbarkeit des Modells auf unbekannte Fahrzeuge von der Menge und der Qualität der aufgezeichneten Eingangsdaten und Komponentenzustandsinformationen abhängt.
Dieses Konzept ist nicht auf Fahrzeuge und deren Komponenten beschränkt. Es können auch diverse technische Systeme aus anderen Industriebereichen für eine solche Komponentenzustandsbestimmung in Betracht gezogen werden. Beispielsweise sei hier die langfristige Verformung eines Pressstempels in einer großen Industrieanlage oder die langfristige Alterung von Lithium-Ionen-Akkus in elektrisch betriebenen Kraftfahrzeugen genannt.

Neben einer Übertragbarkeit auf andere Fahrzeuge, Komponenten oder technische Systeme ist auch eine Übertragbarkeit auf andere zeitliche Horizonte denkbar. Der in dieser Arbeit betrachtete zeitliche Horizont umfasst eine Ausdehnung über mehrere Monate. Innerhalb einer Diskretisierungsstufe werden statistische Merkmale von (physikalischen) Signalen gebildet. Diese Diskretisierungsstufe kann zielgerichtet angepasst werden. Aus diesem Grund ist das in Kapitel 4 beschriebene Konzept auch auf Zustandsänderungsvorhersagen eines anderen zeitlichen Horizonts anwendbar.

Grundsätzlich ist darauf zu achten, dass genügend Informationen über die zu analysierende Abnutzung und auch über die Testobjekte zur Verfügung stehen. Die Qualität der Daten spielt eine entscheidende Rolle bei der Erstellung des virtuellen Sensormodells. Zeitgleich heißt das aber nicht, dass das alleinige Erzeugen von Daten zur Erstellung geeigneter Modelle notwendig ist. Im Gegenteil können, wie bereits in Kapitel 4.3.2 beschrieben, sehr ähnliche Informationen auch zu einer ineffizienten Modellerstellung führen. In dieser Arbeit wird mit einem unüberwachten Ansatz ein Vorgehen erläutert, mit dem redundante Signale identifiziert und bestimmten Klassen zugeordnet werden können. Dennoch gibt dieser Ansatz keine Garantie, alle Duplikate in den Signalen zu identifizieren und den richtigen Klassen zuzuordnen. Aus diesem Grund ist der Aufwand in der Vorbereitung für eine solche Analyse einer Komponentenzustandsänderung nicht zu vernachlässigen. Eine gezielte Vorauswahl von Informationen und Signalen kann zu einer effizienteren datengetriebenen Analyse führen.

Aus der Komponentenzustandsvorhersage lässt sich die Restnutzungsdauer (engl. RUL) bzw. die Lebensdauer einer Komponente ableiten (vgl. Abschnitt 3.1). So kann dem Fachexperten der Domäne oder sogar dem Fahrzeugführer zukünftig die Lebensdauer der untersuchten Komponente vorhergesagt werden. Aufbauend darauf sind Hinweise bei größeren Komponentenzustandsänderungen beispielweise im Multimedia-Display denkbar. Der Fahrzeugführer kann daraufhin sein Fahrverhalten anpassen und zu einer möglichen längeren Lebensdauer der untersuchten Komponente beitragen.

6.3 Ausblick

Die in Kapitel 5.4 dargestellten Ergebnisse wurden mit Datensätzen erzielt, die im Rahmen des Fahrzeugentwicklungsprozesses aufgenommen wurden. Die in dieser Arbeit verwendeten Alterungsinformationen werden in Werkstätten in unregelmäßigen Abständen gemessen.

Zukünftig können regelmäßige Komponentenzustandsmessungen in kurzen Zeitintervallen nicht nur eine hohe Datenqualität bieten, sondern ermöglichen auch gezielte Ursachenuntersuchungen möglicher Einflussfaktoren bezogen auf die Zustandsänderungen. Auch für zukünftige Fahrzeuggenerationen können datengetriebene Einflussfaktorenanalysen von Bedeutung sein. Die Entwicklung zukünftiger Fahrzeuge kann von den Ergebnissen und Erfahrungen datengetriebener Analysen vorheriger Fahrzeuggenerationen profitieren. Durch die Reduktion der Anzahl der Prototypen in der frühen Phase der Produktentstehung können Kosten und Ressourcen eingespart und durch den frühen Einsatz datengetriebener Analysemethoden kann Entwicklungszeit verkürzt werden.
Des Weiteren können zukünftig auch vernetzte, automatisierte Prüfstände genutzt werden, um solche aufwendigen Zustandsänderungen an ausgewählten (Fahrzeug-)Komponenten zu messen. An einem Prüfstand können einzelne technische Komponenten gezielt und reproduzierbar auf Veränderungen untersucht werden. Auch hier ist eine hochdynamische Messdatenerfassung möglich. Ein weiterer Vorteil ist, dass ein Prüfstand eine Komponente gezielt mit einstellbaren dynamischen Zuständen belasten kann. Ein weiterer Vorteil ist Möglichkeit einer dynamischen Komponentenbelastung. Sie kann dadurch vergleichsweise intensiveren Abnutzungserscheinungen unterliegen, als es im normalen Fahrbetrieb im Straßenverkehr möglich wäre. Die an einem Prüfstand erhobenen Daten können zukünftig auch mit denen aus dem realen Fahrbetrieb im Straßenverkehr verglichen und für datengetriebene Analysen betrachtet werden.

Das in dieser Arbeit dargestellte Konzept sieht eine Alterszustandsprädiktion einer ausgewählten Fahrzeugkomponente unter Zuhilfenahme von vorliegenden Daten vor. Die untersuchten Fahrzeuge weisen den gleichen Fahrzeugtyp auf. Allerdings führt derselbe Fahrzeugtyp nicht zwangsläufig zu einem identischen Komponenteneinbau im Fahrzeug. Dies ist zum Einen durch unterschiedliche Ausstattungsmerkmale gegeben, die wiederum unterschiedliche technische Komponenten und Steuergeräte im Fahrzeug bedingen. Zum Anderen werden einzelne Komponenten möglicherweise von unterschiedlichen Lieferanten bzw. Herstellern gefertigt. Diese Komponenten können beispielsweise Strukturteile der Karosserie, Leuchtmittel in den Scheinwerfern oder Motorsteuergeräte sein. Neben der Hardware können auch softwareseitige Unterschiede in verschiedenen Fahrzeugen bestehen. Allein eine unterschiedliche Installationshandhabung von Updates können abweichende Software-Versionen in den einzelnen Fahrzeugen auftreten.
Es ist nicht auszuschließen, dass sich verschiedene Soft- und Hardware-Versionen unterschiedlich auf eine zu untersuchende Fahrzeugzustandsänderung auswirken können. An dieser Stelle ist eine Dokumentation dieser Versionen für eine weitere mögliche Einflussanalyse empfehlenswert. Bei einer ausreichenden Menge an Daten und Fahrzeugen ist einer Modellerstellung der Fahrzeugzustandsänderung nicht nur für jedes Fahrzeug, sondern sogar für jede Soft- und Hardware-Kombination eines Fahrzeugtyps denkbar. Stünden darüber hinaus ausreichend erstellte Modelle zur Verfügung, könnten für eine weiterführende Analyse die Änderungen zwischen den einzelnen Modellen betrachtet werden.

Die Arbeit zeigt, dass mit dem hier vorgestellten Konzept ein virtuelles Sensormodell zur Alterungsbestimmung erstellt werden kann. Neben einer quantitativen Bewertung der Prognosegüte der unterschiedlichen Modelle ist auch eine qualitative Bewertung der Ergebnisse

möglich. So konnten im Zusammenhang mit der untersuchten Alterung relevante Signale ermittelt werden. Im Rahmen des Produktentstehungsprozesses eines Fahrzeugs können solche Relevanzanalysen zu einer zielgerichteten Analyse einer Komponentenzustandsänderung beitragen und in der Entwicklung für zukünftige Fahrzeugentwicklungen genutzt werden.
Weiterhin zeigt die Arbeit, dass unter Einbezug des Standes der Wissenschaft, geeignete Methoden aus dem Bereich des Data-Minings, ausreichend zur Verfügung gestellter Daten und entsprechender Vorverarbeitung der Daten, datengetriebene Modelle zur Vorhersage von physikalisch nicht vollumfänglich verstandenen Komponentenzustandsänderungen anwendbar sind. Der intuitive Gedanke, dass eine feinere Diskretisierungsstufe zu besseren Ergebnissen führt, konnte nicht bestätigt werden (vgl. Kapitel 5.5). Dagegen ist eine Abhängigkeit der Diskretisierungsstufe von dem untersuchten Alterungshorizont, der Vorverarbeitung der Daten und der eingesetzten Methoden des maschinellen Lernens vorstellbar.

Diese Arbeit trägt dazu bei, den eingangs erwähnten steigenden Durchsatz digitaler Daten zielorientiert zu reduzieren und mit geeigneten Methoden hilfreiche Analysen für Entwickler und Kunden zu liefern. Es wird hier ein Konzept zur Prädiktion einer langfristigen Zustandsänderung aufgezeigt. Mit Hilfe geeigneter Methoden (Signalclustering, Auswahl von Teilmengen einzigartiger Signale, Auswahl von Merkmalen bei gegebenen Diskretisierungsstufen) kann die Eingangsdatenmenge um ein Vielfaches reduziert werden. Es ist nicht nur eine Frage der Güte der Vorhersage, sondern auch eine Frage der verbrauchten Ressourcen und der Wiederverwendbarkeit der Ergebnisse. Letztendlich werden zukünftige Arbeiten zeigen, wie der eingangs erwähnte steigende Durchsatz digitaler Daten für zielorientierte Analysen genutzt werden kann und wie diese Analysen und Ergebnisse in den (Fahrzeug-) Entwicklungsprozess integriert werden.

Anhang

A.1 Vorstellung der Hyperparameter

A.1.1 Wahl der Signalmenge

Clustering mit unterschiedlichen Ansätzen

Im Folgenden werden Ergebnisse des Clusterings mit drei unterschiedlichen Ansätzen gezeigt: MeanShift-, DBSCAN- und der Agglomerative-Ansatz. Die Abbildungen A.1, A.2 und A.3) zeigen den Verlauf des Silhouetten-Koeffizienten bei unterschiedlichen Einstellparametern, unter Verwendung aller zur Verfügung stehender Merkmale des Fahrzeugs *219* bei einer festgelegten Diskretisierungsstufe von einer Stunde. Es sind jeweils die Werte des Silhouetten-Koeffizienten und die zugehörigen Clusteranzahlen dargestellt. Vom Algorithmus wird eigenständig die Konfiguration gewählt, die das Maximum des Silhouetten-Koeffizient repräsentiert (gekennzeichnet durch eine rote Umrandung).

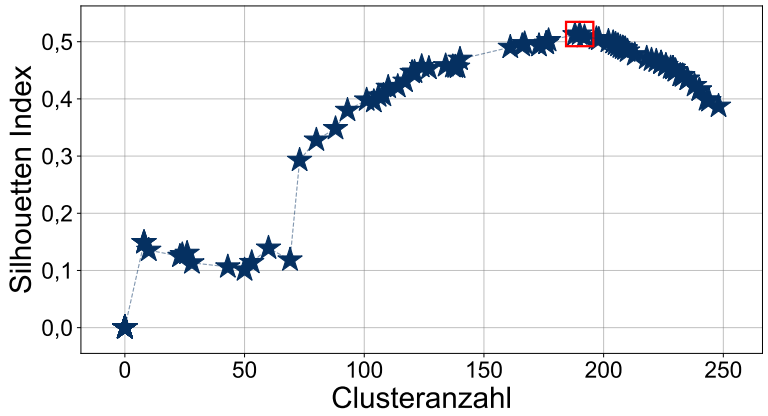


Abbildung A.1: Darstellung des Silhouettenverlaufs des MeanShift-Ansatzes bei unterschiedlichen Einstellparametern unter Verwendung der Daten vom Fahrzeugs *219*

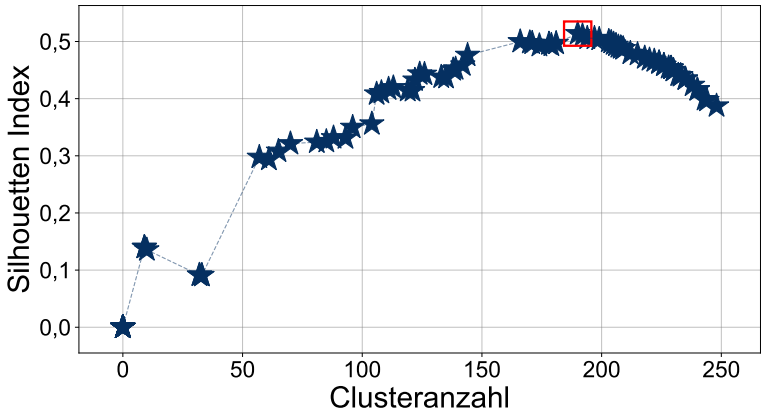


Abbildung A.2: Darstellung des Silhouettenverlaufs des DBSCAN-Ansatzes bei unterschiedlichen Einstellparametern unter Verwendung der Daten vom Fahrzeugs *219*

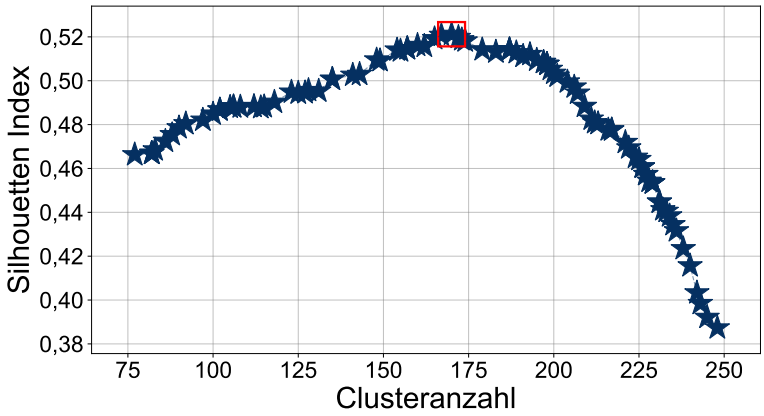


Abbildung A.3: Darstellung des Silhouettenverlaufs des Agglomerative-Ansatzes bei unterschiedlichen Einstellparametern unter Verwendung der Daten vom Fahrzeugs *219*

A.1.2 Wahl der Diskretisierungsstufe

Die Abbildung A.4 zeigt mehrere Merkmale des Fahrzeugs *219*, aufgetragen über der Anzahl der Samples. Dabei sind in der Abbildung die Merkmale *Mean*, *Median*, *Q_25*, *Q_75*, *Min* und *Max* bei einer Diskretisierungsstufe von 15 Stunden dargestellt.

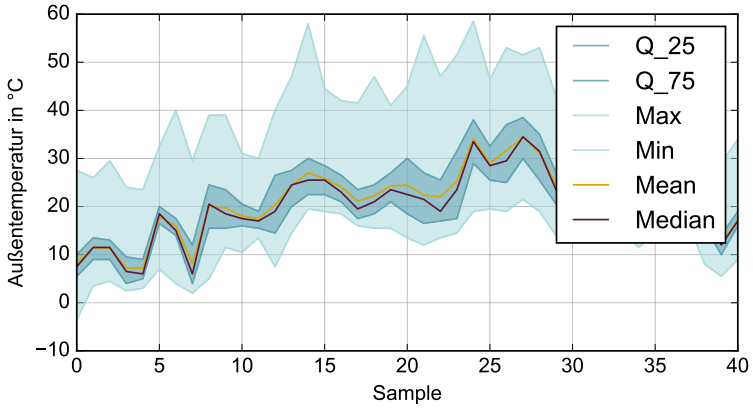
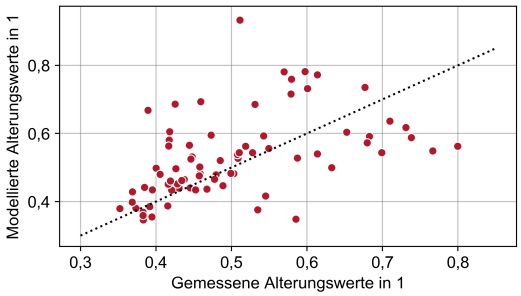


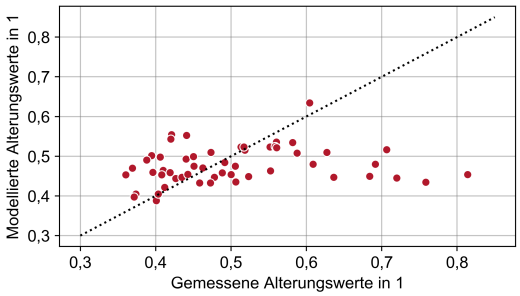
Abbildung A.4: Darstellung der Merkmale (*Mean*, *Median*, *Q_25*, *Q_75*, *Min* und *Max*) des Fahrzeugs *219*, Diskretisierungsstufe: 15 Stunden

A.2 Hybrider Expertenansatz

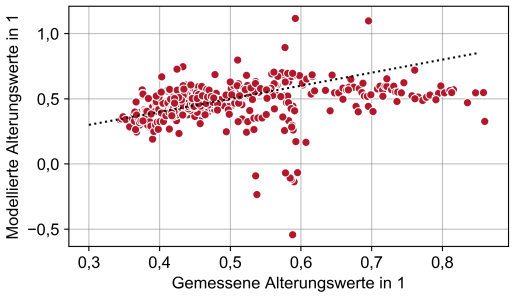
Die Abbildungen A.5 und A.6 stellen die gemessenen und modellierten Alterungswerte unter Anwendung des Expertenansatzes dar. Die modellierten Alterungswerte entsprechen der Modellprädiktion des Expertenmodells bei gegebenen Testdaten.



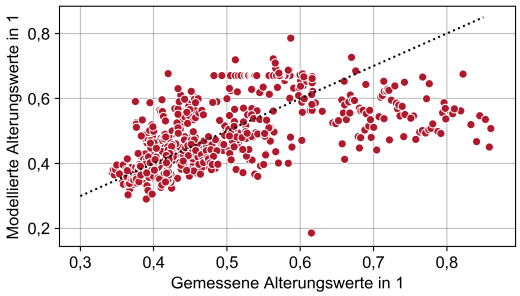
(a) ML-Methode = Bayes; Diskr.-stufe = 80h; RMSE = 0,1197; MAPE = 17,02%, n_{pred} = 75



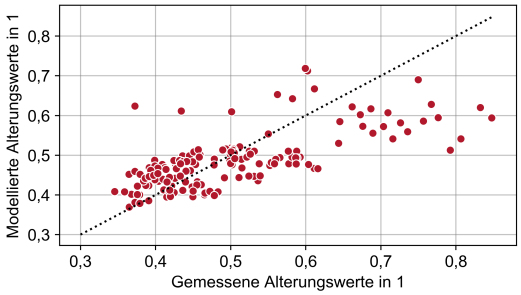
(b) ML-Methode = kNN; Diskr.-stufe = 110h; RMSE = 0,1111; MAPE = 14,10%, n_{pred} = 54



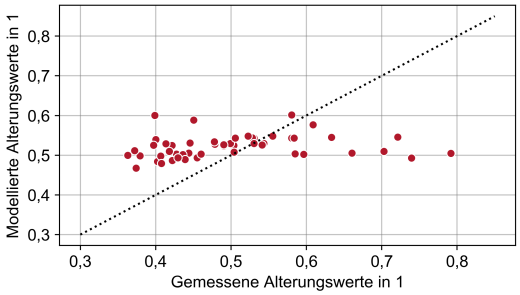
(c) ML-Methode = MLR; Diskr.-stufe = 15h; RMSE = 0,1616; MAPE = 18,54%, n_{pred} = 420



(d) ML-Methode = NN; Diskr.-stufe = 10h; RMSE = 0,1014; MAPE = 14,05%, n_{pred} = 630

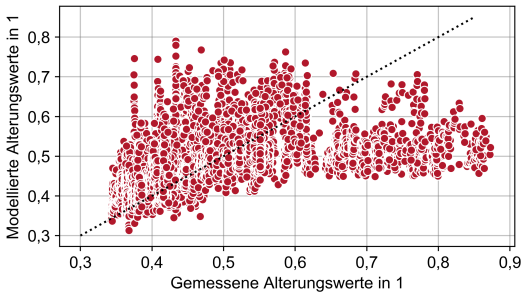


(e) ML-Methode = RF; Diskr.-stufe = 40h; RMSE = 0,0822; MAPE = 11,69%, n_{pred} = 155

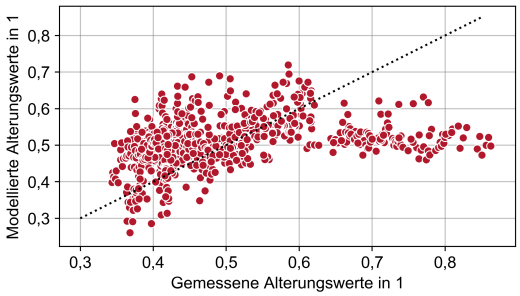


(f) ML-Methode = SVR; Diskr.-stufe = 120h; RMSE = 0,1011; MAPE = 16,25%, n_{pred} = 51

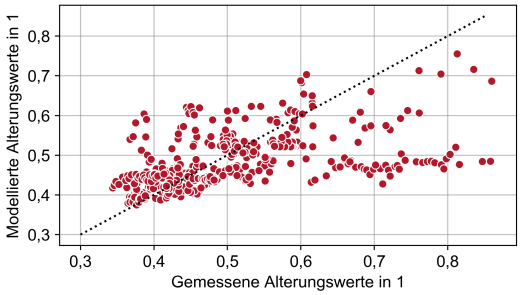
Abbildung A.5: Darstellung der modellierten Alterungswerte der jeweils besten Konfiguration für die unterschiedlichen ML-Methoden unter Anwendung des hybriden Expertenansatzes, gewählte ML-Methoden: Bayes **(a)**, kNN **(b)**, MLR **(c)**, NN **(d)**, RF **(e)** und SVR **(f)**



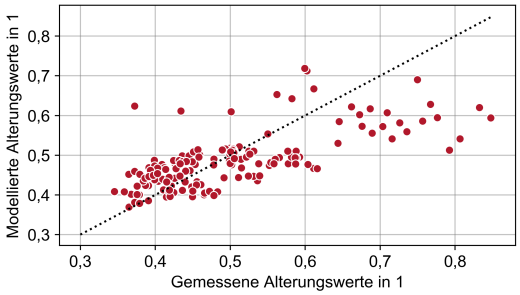
(a) Diskr.-stufe = 1h; ML-Methode = SVR; RMSE = 0,1043; MAPE = 13,78%, n_{pred} = 6344



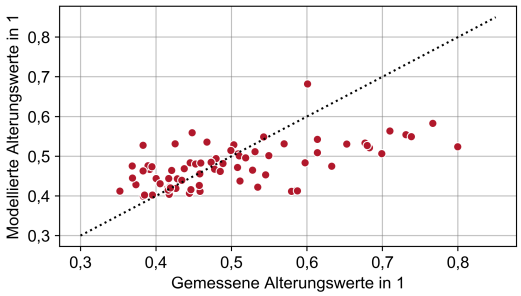
(b) Diskr.-stufe = 8h; ML-Methode = SVR; RMSE = 0,1085; MAPE = 16,21%, n_{pred} = 790



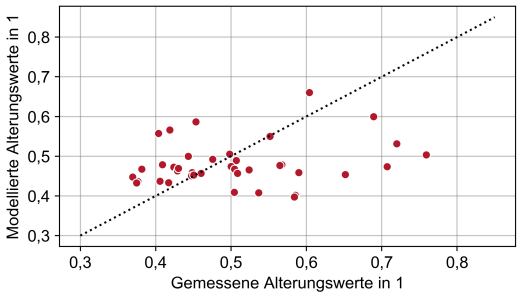
(c) Diskr.-stufe = 15h; ML-Methode = RF; RMSE = 0,0985; MAPE = 12,16%, n_{pred} = 420



(d) Diskr.-stufe = 40h; ML-Methode = RF; RMSE = 0,0822; MAPE = 11,69%, n_{pred} = 155



(e) Diskr.-stufe = 80h; ML-Methode = RF; RMSE = 0,0930; MAPE = 12,68%, n_{pred} = 75

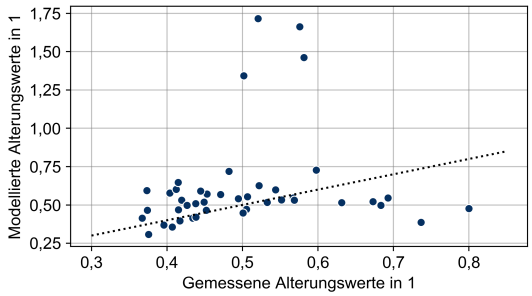


(f) Diskr.-stufe = 150h; ML-Methode = RF; RMSE = 0,1061; MAPE = 15,39%, n_{pred} = 39

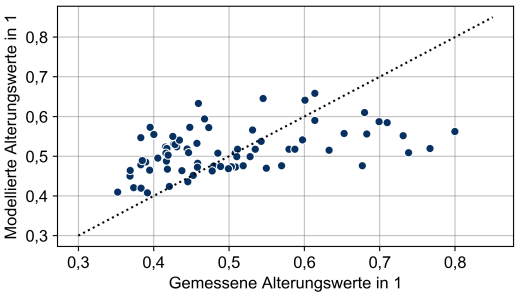
Abbildung A.6: Darstellung der modellierten Alterungswerte der jeweils besten Konfiguration für die unterschiedlichen Diskretisierungsstufen unter Anwendung des hybriden Expertenansatzes, gewählte Diskr.-stufen: 1 Stunde **(a)**, 8 Stunden **(b)**, 15 Stunden **(c)**, 40 Stunden **(d)**, 80 Stunden **(e)** und 150 Stunden **(f)**

A.3 Datengetriebene Hyperparameteroptimierung

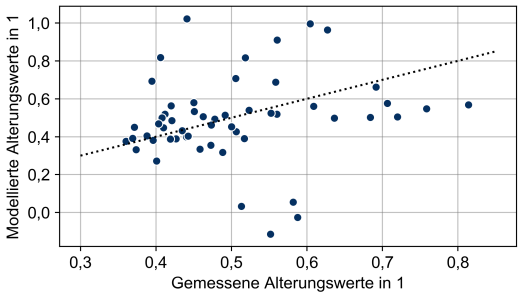
Die Abbildungen A.7 und A.8 stellen die gemessenen und modellierten Alterungswerte unter Anwendung des Optimierungsansatzes dar. Die modellierten Alterungswerte entsprechen der Modellprädiktion des Optimierungsansatzes bei gegebenen Testdaten.



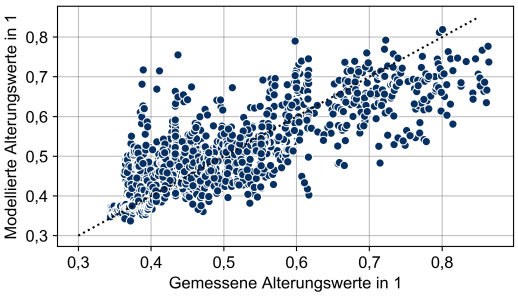
(a) ML-Methode = Bayes; Diskr.-stufe = 140h; RMSE = 0,3332; MAPE = 35,61%, n_{pred} = 43



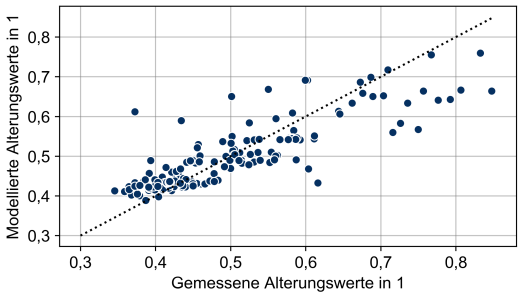
(b) ML-Methode = kNN; Diskr.-stufe = 80h; RMSE = 0,0972; MAPE = 15,73%, n_{pred} = 75



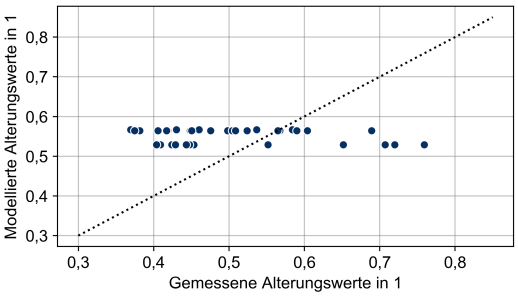
(c) ML-Methode = MLR; Diskr.-stufe = 110h; RMSE = 0,2308; MAPE = 30,49%, n_{pred} = 54



(d) ML-Methode = NN; Diskr.-stufe = 4h; RMSE = 0,0703; MAPE = 10,39%, n_{pred} = 1582

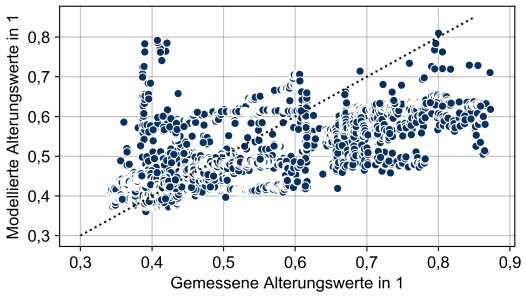


(e) ML-Methode = RF; Diskr.-stufe = 40h; RMSE = 0,0602; MAPE = 8,35%, n_{pred} = 155

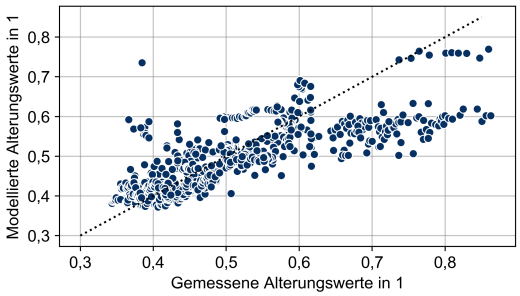


(f) ML-Methode = SVR; Diskr.-stufe = 150h; RMSE = 0,1162; MAPE = 21,16%, n_{pred} = 39

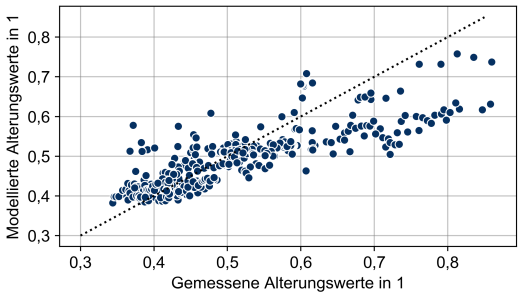
Abbildung A.7: Darstellung der modellierten Alterungswerte der jeweils besten Konfiguration für die unterschiedlichen ML-Methoden unter Anwendung des Hyperparameteroptimierungsansatzes, gewählte ML-Methoden: Bayes **(a)**, kNN **(b)**, MLR **(c)**, NN **(d)**, RF **(e)** und SVR **(f)**



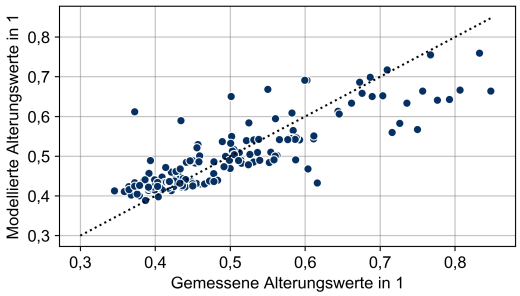
(a) Diskr.-stufe = 1h; ML-Methode = RF; RMSE = 0,0828; MAPE = 11,27%; n_{pred} = 6344



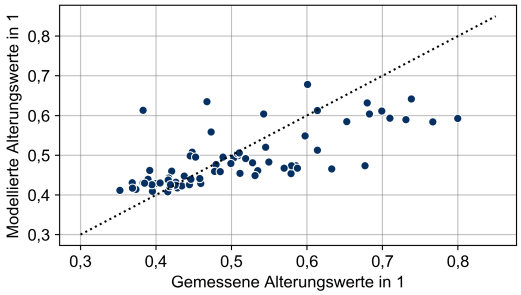
(b) Diskr.-stufe = 8h; ML-Methode = RF; RMSE = 0,0711; MAPE = 9,14%; n_{pred} = 790



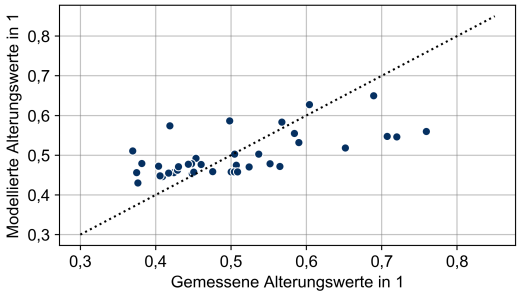
(c) Diskr.-stufe = 15h; ML-Methode = RF; RMSE = 0,0669; MAPE = 8,74%; n_{pred} = 420



(d) Diskr.-stufe = 40h; ML-Methode = RF; RMSE = 0,0602; MAPE = 8,35%; n_{pred} = 155



(e) Diskr.-stufe = 80h; ML-Methode = RF; RMSE = 0,0774; MAPE = 10,59%; n_{pred} = 75



(f) Diskr.-stufe = 150h; ML-Methode = RF; RMSE = 0,0775; MAPE = 11,93%; n_{pred} = 39

Abbildung A.8: Darstellung der modellierten Alterungswerte der jeweils besten Konfiguration für die unterschiedlichen Diskretisierungsstufen unter Anwendung des Hyperparameteroptimierungsansatzes, gewählte Diskr.-stufen: 1 Stunde **(a)**, 8 Stunden **(b)**, 15 Stunden **(c)**, 40 Stunden **(d)**, 80 Stunden **(e)** und 150 Stunden **(f)**

A.3.1 Einfluss der Signale

Die Abbildung A.9 zeigt die relative Häufigkeit der einzelnen Signale, die vom Optimierungsalgorithmus benutzt wurden. Die Abbildung A.10 zeigt die am häufigsten ausgewählten Signale, die vom Optimierungsalgorithmus benutzt wurden.

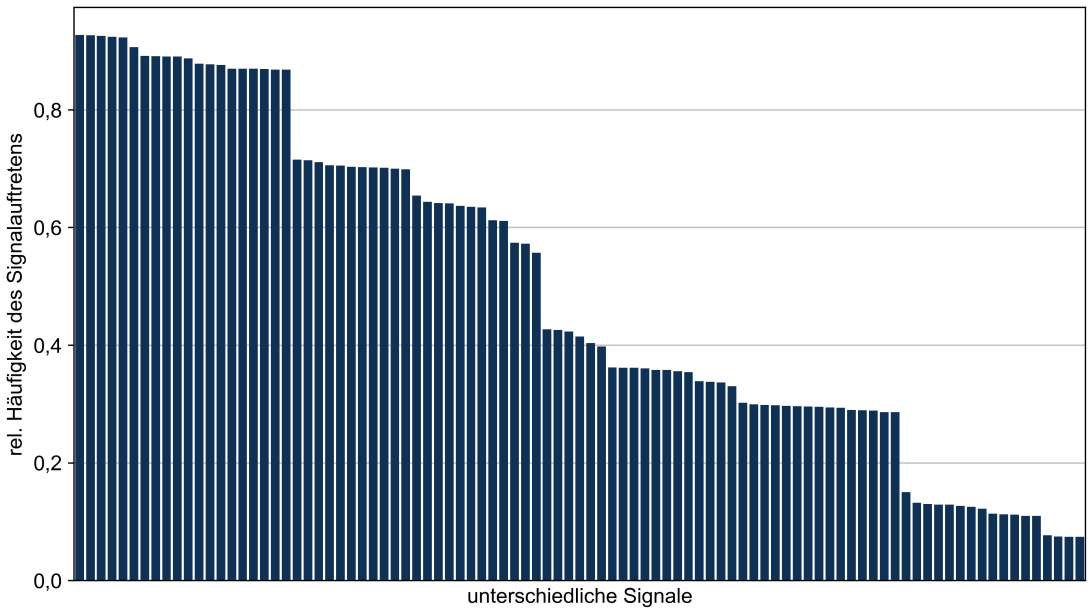


Abbildung A.9: Darstellung der relativen Häufigkeiten des Auftretens der unterschiedlichen Signale im Rahmen der datengetriebenen Optimierung

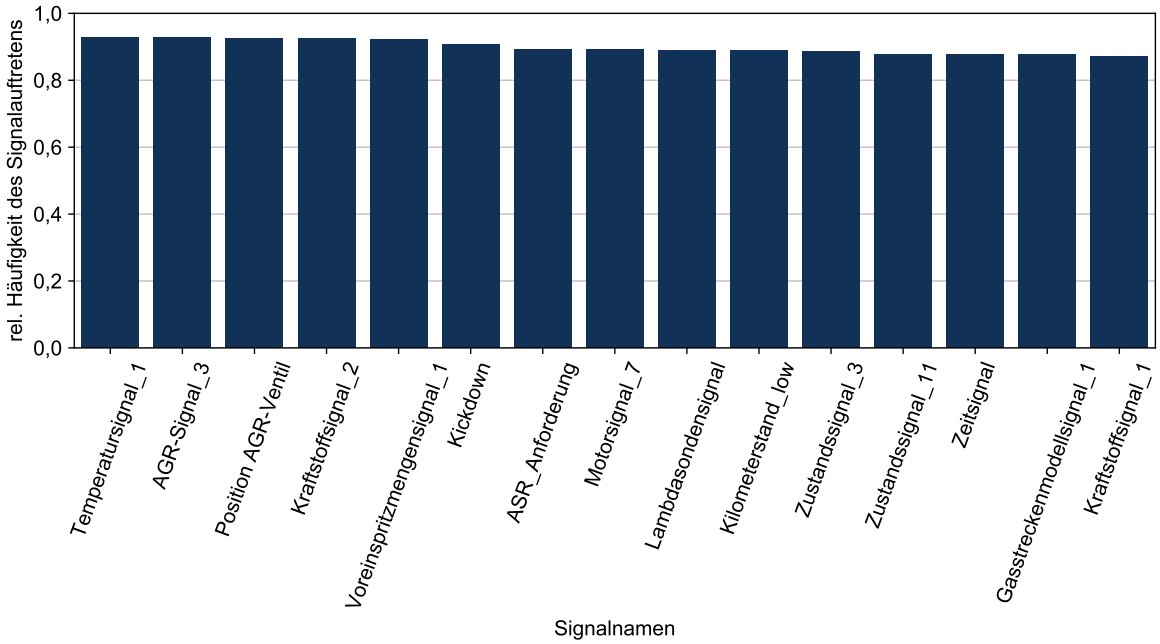


Abbildung A.10: Darstellung der relativen Häufigkeiten der ausgewählten Signale, die vom Optimierungsalgorithmus am häufigsten benutzt wurden

A.3.2 Einfluss der Merkmale

Die Abbildung A.11 zeigt die relative Verteilung der ausgewählten Merkmale im Verlauf aller Durchläufe der Optimierung.

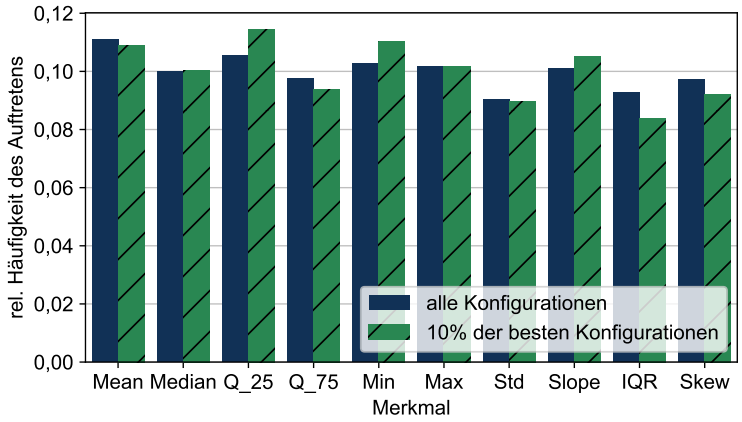


Abbildung A.11: Darstellung der relativen Auftrittshäufigkeiten der verwendeten Merkmale, die vom Optimierungsalgorithmus benutzt wurden; Vergleich zwischen allen Durchläufen und den Durchläufen der besten 10%

Literatur

[SEF19] A. Sass, E. Esatbeyoglu und T. Fischer. „Monitoring of Powertrain Component Aging Using In-Vehicle Signals“. In: *Diagnose in Mechatronischen Fahrzeugsystemen XIII: Neue Verfahren für Test, Prüfung und Diagnose von E/E-Systemen im Kfz* (2019), S. 15–28.

[SEI19] A. Sass, E. Esatbeyoglu und T. Iwwerks. „Signal Pre-Selection for Monitoring and Prediction of Vehicle Powertrain Component Aging“. In: *Science & Technique* 18.6 (5. Dez. 2019), S. 519–524.

[SEI20] A. Sass, E. Esatbeyoglu und T. Iwwerks. „Data-Driven Powertrain Component Aging Prediction Using In-Vehicle Signals“. In: *SOFSEM (Doctoral Student Research Forum)*. 2020, S. 109–119.

[Pol20] A. Poleshova. *Datenvolumen des privaten und geschäftlichen IP-Traffics weltweit in den Jahren 2014 bis 2017 sowie eine Prognose bis 2022*. 1. Apr. 2020. URL: `https://de.statista.com/statistik/daten/studie/266885/umfrage/prognose-zum-datenvolumen-des-privaten-und-geschaeftlichen-ip-traffics-weltweit/` (besucht am 02. 10. 2020).

[Hin18] Ralph Hintemann. „Boom führt zu deutlich steigendem Energiebedarf der Rechenzentren in Deutschland im Jahr 2017“. In: *Borderstep Institut für Innovation und Nachhaltigkeit, Berlin* Borderstep Institut (2018).

[Wal06] Henning Wallentowitz, Hrsg. *Handbuch Kraftfahrzeugelektronik: Grundlagen, Komponenten, Systeme, Anwendungen ; mit zahlreichen Tabellen*. 1. Aufl. ATZ-MTZ-Fachbuch. Wiesbaden: Vieweg, 2006. 716 S.

[ZS14] Werner Zimmermann und Ralf Schmidgall. *Bussysteme in der Fahrzeugtechnik: Protokolle, Standards und Softwarearchitektur ; mit 103 Tabellen*. 5., aktualisierte und erw. Aufl. ATZ/MTZ-Fachbuch. Wiesbaden: Springer Vieweg, 2014. 507 S.

[dFCO15] Haroldo de Faria, João Gabriel Spir Costa und Jose Luis Mejia Olivas. „A Review of Monitoring Methods for Predictive Maintenance of Electric Power Transformers Based on Dissolved Gas Analysis“. In: *Renewable and Sustainable Energy Reviews* 46 (Juni 2015), S. 201–209.

[Lee+14] Jay Lee u. a. „Prognostics and Health Management Design for Rotary Machinery Systems—Reviews, Methodology and Applications“. In: *Mechanical Systems and Signal Processing* 42.1-2 (Jan. 2014), S. 314–334.

[Pry14] Rune Prytz. „Machine Learning Methods for Vehicle Predictive Maintenance Using Off-Board and on-Board Data“. Halmstad: Halmstad University, 2014.

[TMZ12] D.A. Tobon-Mejia, K. Medjaher und N. Zerhouni. „CNC Machine Tool's Wear Diagnostic and Prognostic by Using Dynamic Bayesian Networks“. In: *Mechanical Systems and Signal Processing* 28 (Apr. 2012), S. 167–182.

[Tet+10] R. Teti u. a. „Advanced Monitoring of Machining Operations“. In: *CIRP Annals* 59.2 (2010), S. 717–739.

[Bed+13] Inigo Bediaga u. a. „Ball Bearing Damage Detection Using Traditional Signal Processing Algorithms“. In: *IEEE Instrumentation & Measurement Magazine* 16.2 (Apr. 2013), S. 20–25.

[Zhe+18] Caifeng Zheng u. a. „A Data-driven Approach for Remaining Useful Life Prediction of Aircraft Engines“. In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. 2018 21st International Conference on Intelligent Transportation Systems (ITSC). Maui, HI: IEEE, Nov. 2018, S. 184–189.

[Kar+10] Hillol Kargupta u. a. „Minefleet: The Vehicle Data Stream Mining System for Ubiquitous Environments“. In: *Ubiquitous Knowledge Discovery*. Springer, 2010, S. 235–254.

[GP15] Deepam Goyal und B.S. Pabla. „Condition Based Maintenance of Machine Tools—A Review“. In: *CIRP Journal of Manufacturing Science and Technology* 10 (Aug. 2015), S. 24–35.

[Pap+13] A. Papacharalampopoulos u. a. „Acoustic Emission Signal Through Turning Tools: A Computational Study“. In: *Procedia CIRP* 8 (2013), S. 426–431.

[Cae+16] Wahyu Caesarendra u. a. „Acoustic Emission-Based Condition Monitoring Methods: Review and Application for Low Speed Slew Bearing“. In: *Mechanical Systems and Signal Processing* 72–73 (Mai 2016), S. 134–159.

[XWX15] Jiuping Xu, Yusheng Wang und Lei Xu. „PHM-Oriented Sensor Optimization Selection Based on Multiobjective Model for Aircraft Engines“. In: *IEEE Sensors Journal* 15.9 (Sep. 2015), S. 4836–4844.

[Hod18] Wilhelm Hodapp. „Die Bedeutung einer zustandsorientierten Instandhaltung: Einsatz und Nutzen in der Investitionsgüterindustrie“. In: *Betriebliche Instandhaltung*. Hrsg. von Jens Reichel, Gerhard Müller und Jean Haeffs. Berlin, Heidelberg: Springer Berlin Heidelberg, 2018, S. 135–152.

[Bro+00] T. Brotherton u. a. „Prognosis of Faults in Gas Turbine Engines“. In: *2000 IEEE Aerospace Conference. Proceedings (Cat. No.00TH8484)*. 2000 IEEE Aerospace Conference Proceedings. Bd. 6. Big Sky, MT, USA: IEEE, 2000, S. 163–171.

[Sam17] Claude Sammut. „Generalization“. In: *Encyclopedia of Machine Learning and Data Mining*. Hrsg. von Claude Sammut und Geoffrey I. Webb. Boston, MA: Springer US, 2017, S. 556–556.

[Bac+18] Klaus Backhaus u. a. *Multivariate Analysemethoden: eine anwendungsorientierte Einführung*. 15., vollständig überarbeitete Auflage. Berlin Heidelberg: Springer Gabler, 2018. 625 S.

[MG16] Andreas Christian Müller und Sarah Guido. *Introduction to Machine Learning with Python: A Guide for Data Scientists*. First edition. Python/Machine Learning. Beijing: O’Reilly, 2016.

[Pad+17] Luis Carlos Padierna u. a. „Hyper-Parameter Tuning for Support Vector Machines by Estimation of Distribution Algorithms“. In: *Nature-Inspired Design of Hybrid Intelligent Systems*. Hrsg. von Patricia Melin, Oscar Castillo und Janusz Kacprzyk. Bd. 667. Cham: Springer International Publishing, 2017, S. 787–800.

[And19] Răzvan Andonie. „Hyperparameter Optimization in Learning Systems“. In: *Journal of Membrane Computing* 1.4 (Dez. 2019), S. 279–291.

[Sch+15] Nicolas Schilling u. a. „Hyperparameter Optimization with Factorized Multilayer Perceptrons“. In: *Machine Learning and Knowledge Discovery in Databases*. Hrsg. von Annalisa Appice u. a. Bd. 9285. Cham: Springer International Publishing, 2015, S. 87–103.

[BIP20] BIPM. *GUM: Guide to the Expression of Uncertainty in Measurement*. 2020. URL: `https://www.bipm.org/en/publications/guides/gum.html` (besucht am 23. 02. 2021).

[KAC17] Nam-Ho Kim, Dawn An und Joo-Ho Choi. „Introduction“. In: *Prognostics and Health Management of Engineering Systems*. Cham: Springer International Publishing, 2017, S. 1–24.

[San+09] Chaitanya Sankavaram u. a. „Model-Based and Data-Driven Prognosis of Automotive and Electronic Systems“. In: *2009 IEEE International Conference on Automation Science and Engineering*. 2009 IEEE International Conference on Automation Science and Engineering (CASE 2009). Bangalore, India: IEEE, Aug. 2009, S. 96–101.

[WFH11] I. H. Witten, Eibe Frank und Mark A. Hall. *Data Mining: Practical Machine Learning Tools and Techniques*. 3rd ed. Morgan Kaufmann Series in Data Management Systems. Burlington, MA: Morgan Kaufmann, 2011. 629 S.

[Alp10] Ethem Alpaydin. *Introduction to Machine Learning*. 2nd ed. Adaptive Computation and Machine Learning. Cambridge, Mass: MIT Press, 2010. 537 S.

[GE03] Isabelle Guyon und André Elisseeff. „An Introduction to Variable and Feature Selection“. In: *Journal of machine learning research* 3 (Mar 2003), S. 1157–1182.

[Dom12] Pedro Domingos. „A Few Useful Things to Know about Machine Learning“. In: *Communications of the ACM* 55.10 (Okt. 2012), S. 78–87.

[HBV01] Maria Halkidi, Yannis Batistakis und Michalis Vazirgiannis. „On Clustering Validation Techniques“. In: *Journal of intelligent information systems* 17.2-3 (2001), S. 107–145.

[Ger19] Stefan Gerlach. „Daten- und Signalanalyse“. In: *Computerphysik*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2019, S. 245–268.

[BCK12] Marco Burkschat, Erhard Cramer und Udo Kamps. *Beschreibende Statistik*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012.

[MS05] Karl C. Mosler und Friedrich Schmid. *Beschreibende Statistik und Wirtschaftsstatistik: mit 2 Tabellen*. 2., verb. Aufl. Springer-Lehrbuch. Berlin: Springer, 2005. 252 S.

[JMF99] A. K. Jain, M. N. Murty und P. J. Flynn. „Data Clustering: A Review“. In: *ACM Computing Surveys* 31.3 (1. Sep. 1999), S. 264–323.

[ASY15] Saeed Aghabozorgi, Ali Seyed Shirkhorshidi und Teh Ying Wah. „Time-Series Clustering – A Decade Review“. In: *Information Systems* 53 (Okt. 2015), S. 16–38.

[RLR98] M. Ramze Rezaee, B.P.F. Lelieveldt und J.H.C. Reiber. „A New Cluster Validity Index for the Fuzzy C-Mean“. In: *Pattern Recognition Letters* 19.3-4 (März 1998), S. 237–246.

[WK18] Slawomir Wierzchoń und Mieczyslaw Kłopotek. *Modern Algorithms of Cluster Analysis*. Bd. 34. Studies in Big Data. Cham: Springer International Publishing, 2018.

[HKK05] J. Handl, J. Knowles und D. B. Kell. „Computational Cluster Validation in Post-Genomic Data Analysis“. In: *Bioinformatics* 21.15 (1. Aug. 2005), S. 3201–3212.

[Ert16] Wolfgang Ertel. *Grundkurs Künstliche Intelligenz: eine praxisorientierte Einführung*. 4., überarbeitete Auflage. Computational Intelligence. Wiesbaden: Springer Vieweg, 2016. 385 S.

[Bis06] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Information Science and Statistics. New York: Springer, 2006. 738 S.

[QB17] Novi Quadrianto und Wray L. Buntine. „Linear Regression“. In: *Encyclopedia of Machine Learning and Data Mining*. Hrsg. von Claude Sammut und Geoffrey I. Webb. Boston, MA: Springer US, 2017, S. 747–750.

[Sch12] Rainer Schlittgen. *Einführung in die Statistik: Analyse und Modellierung von Daten*. München: Oldenbourg; 2012.

[Zha17] Xinhua Zhang. „Support Vector Machines“. In: *Encyclopedia of Machine Learning and Data Mining*. Hrsg. von Claude Sammut und Geoffrey I. Webb. Boston, MA: Springer US, 2017, S. 1214–1220.

[Kle+17] Tania Kleynhans u. a. „Predicting Top-of-Atmosphere Thermal Radiance Using MERRA-2 Atmospheric Data with Deep Learning“. In: *Remote Sensing* 9.11 (7. Nov. 2017).

[HTF17] Trevor Hastie, Robert Tibshirani und Jerome H. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Second edition, corrected at 12th printing. Springer Series in Statistics. New York, NY: Springer, 2017.

[HH17] Ronny Hänsch und Olaf Hellwich. „Random Forests“. In: *Photogrammetrie und Fernerkundung*. Hrsg. von Christian Heipke. Berlin, Heidelberg: Springer Berlin Heidelberg, 2017, S. 603–643.

[Tin95] Tin Kam Ho. „Random Decision Forests“. In: *Proceedings of 3rd International Conference on Document Analysis and Recognition*. 3rd International Conference on Document Analysis and Recognition. Bd. 1. Montreal, Que., Canada: IEEE Comput. Soc. Press, 1995, S. 278–282.

[SW17] „Random Subspace Method“. In: *Encyclopedia of Machine Learning and Data Mining*. Hrsg. von Claude Sammut und Geoffrey I. Webb. Boston, MA: Springer US, 2017, S. 1055–1055.

[Bre01] Leo Breiman. „Random Forests“. In: *Machine Learning* 45.1 (1. Okt. 2001), S. 5–32.

[Roj93] Raúl Rojas. *Theorie der neuronalen Netze*. Springer-Lehrbuch. Berlin, Heidelberg: Springer Berlin Heidelberg, 1993.

[LeC+12] Yann A. LeCun u. a. „Efficient BackProp“. In: *Neural Networks: Tricks of the Trade: Second Edition*. Hrsg. von Grégoire Montavon, Geneviève B. Orr und Klaus-Robert Müller. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, S. 9–48.

[Rud16] Sebastian Ruder. „An Overview of Gradient Descent Optimization Algorithms“. In: *CoRR* abs/1609.04747 (2016). arXiv: 1609.04747.

[CK20] Erhard Cramer und Udo Kamps. „Beschreibende Statistik“. In: *Grundlagen der Wahrscheinlichkeitsrechnung und Statistik*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2020, S. 1–159.

[Alb+09] Sönke Albers u. a. *Methodik der empirischen forschung*. 3., überarbeitete und erweiterte Auflage. Springer eBook collection: Business and economics. Wiesbaden: Gabler Verlag, 2009.

[AC92] J.Scott Armstrong und Fred Collopy. „Error Measures for Generalizing about Forecasting Methods: Empirical Comparisons“. In: *International Journal of Forecasting* 8.1 (Juni 1992), S. 69–80.

[Bar09] Christian Barrot. „Prognosegütemaße“. In: *Methodik der empirischen Forschung*. Hrsg. von Sönke Albers u. a. Wiesbaden: Gabler Verlag, 2009, S. 547–560.

[HK06] Rob J. Hyndman und Anne B. Koehler. „Another Look at Measures of Forecast Accuracy“. In: *International Journal of Forecasting* 22.4 (Okt. 2006), S. 679–688.

[Knö18] Patrick Knöfel. *Energiebilanzmodellierung zur Ableitung der Evapotranspiration - Beispielregion Khorezm*. Unter Mitarb. von Julius-Maximilians-Universität Würzburg. Würzburger geographische Arbeiten Band 120. Würzburg: Würzburg University Press, 2018. 247 S.

[Liu+17] Bing-Chun Liu u. a. „Urban Air Quality Forecasting Based on Multi-Dimensional Collaborative Support Vector Regression (SVR): A Case Study of Beijing-Tianjin-Shijiazhuang“. In: *PLOS ONE* 12.7 (14. Juli 2017). Hrsg. von Chon-Lin Lee.

[Ric+11] Katja Richter u. a. „Goodness-of-Fit Measures: What Do They Tell about Vegetation Variable Retrieval Performance from Earth Observation Data“. In: SPIE Remote Sensing. Hrsg. von Christopher M. U. Neale und Antonino Maltese. Prague, Czech Republic, 6. Okt. 2011.

[MT19] Rainer Metz und Helmut Thome. „Zeitreihenanalyse“. In: *Handbuch Methoden der empirischen Sozialforschung*. Hrsg. von Nina Baur und Jörg Blasius. Wiesbaden: Springer Fachmedien Wiesbaden, 2019, S. 1451–1465.

[Kau19] Göran Kauermann. „Data Science – Einige Gedanken aus Sicht eines Statistikers“. In: *Informatik Spektrum* (13. Nov. 2019).

[PHR16] Henrik Peters, Falk Howar und Andreas Rausch. „Towards Inferring Environment Models for Control Functions from Recorded Signal Data“. In: *2016 IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering (SANER)*. Suita, Osaka, Japan: IEEE, März 2016, S. 1–4.

[Lin+03] Jessica Lin u. a. „A Symbolic Representation of Time Series, with Implications for Streaming Algorithms“. In: *Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*. DMKD '03. New York, NY, USA: Association for Computing Machinery, 2003, S. 2–11.

[KLR04] Eamonn Keogh, Stefano Lonardi und Chotirat Ann Ratanamahatana. „Towards Parameter-Free Data Mining“. In: *Proceedings of the 2004 ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '04*. The 2004 ACM SIGKDD International Conference. Seattle, WA, USA: ACM Press, 2004, S. 206.

[LAC17] Mathieu Lepot, Jean-Baptiste Aubin und François Clemens. „Interpolation in Time Series: An Introductive Overview of Existing Methods, Their Performance Criteria and Uncertainty Assessment“. In: *Water* 9.10 (17. Okt. 2017), S. 796.

[LKB04] Mark Last, Abraham Kandel und Horst Bunke, Hrsg. *Data Mining in Time Series Databases*. Series in Machine Perception and Artificial Intelligence v.57. New Jersey; London: World Scientific, 2004. 192 S.

[Rou+19] Vaia Rousopoulou u. a. „Data Analytics Towards Predictive Maintenance for Industrial Ovens: A Case Study Based on Data Analysis of Various Sensors Data“. In: *Advanced Information Systems Engineering Workshops*. Hrsg. von Henderik A. Proper und Janis Stirna. Bd. 349. Cham: Springer International Publishing, 2019, S. 83–94.

[Bor14a] Kai Borgeest. „Datenkommunikation im Fahrzeug“. In: *Elektronik in der Fahrzeugtechnik*. Wiesbaden: Springer Fachmedien Wiesbaden, 2014, S. 89–134.

[Lad+96] Nicos Ladommatos u. a. „The Effect of Exhaust Gas Recirculation on Combustion and NOx Emissions in a High-Speed Direct-injection Diesel Engine“. In: International Congress & Exposition. 1. Feb. 1996.

[Zel+98] Paul Zelenka u. a. „Cooled EGR - A Key Technology for Future Efficient HD Diesels“. In: International Congress & Exposition. 23. Feb. 1998.

[EKL03] Jochen Eitel, Wolfgang Kramer und Rainer Lutz. „Abgasrückführung: Neue Abgaskühler reduzieren Emissionen von Dieselmotoren“. In: *ATZ - Automobiltechnische Zeitschrift* 105.9 (Sep. 2003), S. 856–859.

[Hoa+08] John Hoard u. a. „Diesel EGR Cooler Fouling“. In: *SAE International Journal of Engines* 1.1 (6. Okt. 2008), S. 1234–1250.

[BML07] Y. Bravo, F. Moreno und O. Longo. „Improved Characterization of Fouling in Cooled EGR Systems“. In: SAE World Congress & Exhibition. 16. Apr. 2007.

[Bra+15] Yolanda Bravo u. a. „Untersuchung Der Ablagerungsbildung Bei AGR-Kühlern“. In: *MTZ-Motortechnische Zeitschrift* 76.5 (2015), S. 36–41.

[NR00] Hans-Georg Nitzke und Thorsten Rebohl. „Simulation und Realisierung von Abgasrückführkonzepten für Dieselmotoren“. Braunschweig: Verl. Mainz; / Zugl.: Braunschweig, Techn. Univ., Diss., 2000.

[BS12a] Hans-Hermann Braess und Ulrich Seiffert. „Elektrik/Elektronik/Software“. In: *Vieweg handbuch kraftfahrzeugtechnik*. Hrsg. von Hans-Hermann Braess und Ulrich Seiffert. Wiesbaden: Vieweg+Teubner Verlag, 2012, S. 644–762.

[SG20] Martin Schleicher und Sorin Mihai Grigorescu. „Wie neuronale Netze die Entwicklung von Automobilsoftware verändern“. In: *ATZelektronik* 15.1 (1. Jan. 2020), S. 26–34.

[Sha+18] Uferah Shafi u. a. „Vehicle Remote Health Monitoring and Prognostic Maintenance System“. In: *Journal of Advanced Transportation* 2018 (2018), S. 1–10.

[JLB06] Andrew K.S. Jardine, Daming Lin und Dragan Banjevic. „A Review on Machinery Diagnostics and Prognostics Implementing Condition-Based Maintenance“. In: *Mechanical Systems and Signal Processing* 20.7 (Okt. 2006), S. 1483–1510.

[SHM11] J.Z. Sikorska, M. Hodkiewicz und L. Ma. „Prognostic Modelling Options for Remaining Useful Life Estimation by Industry“. In: *Mechanical Systems and Signal Processing* 25.5 (Juli 2011), S. 1803–1836.

[BS12b] Hans-Hermann Braess und Ulrich Seiffert. „Produktentstehungsprozess“. In: *Vieweg handbuch kraftfahrzeugtechnik*. Hrsg. von Hans-Hermann Braess und Ulrich Seiffert. Wiesbaden: Vieweg+Teubner Verlag, 2012, S. 881–948.

[Mru19] Beata Mrugalska. „Remaining Useful Life as Prognostic Approach: A Review“. In: *Human Systems Engineering and Design*. Hrsg. von Tareq Ahram, Waldemar Karwowski und Redha Taiar. Bd. 876. Cham: Springer International Publishing, 2019, S. 689–695.

[Sch10] Michael Schenk, Hrsg. *Instandhaltung technischer Systeme: Methoden und Werkzeuge zur Gewährleistung eines sicheren und wirtschaftlichen Anlagenbetriebs*. Berlin: Springer, 2010. 328 S.

[Ape18] Harald Apel. *Instandhaltungs- und servicemanagement: Systeme mit industrie 4.0*. München: Hanser, 2018.

[SB14] René Schenkendorf und Thomas Böhm. „Aspekte einer datengetriebenen, zustandsabhängigen Instandhaltung“. In: *EI - Der Eisenbahningenieur (Tetzlaff Verlag)* 14.11 (2014), S. 14–18.

[Bor14b] Kai Borgeest. „Sicherheit und Zuverlässigkeit“. In: *Elektronik in der Fahrzeugtechnik*. Wiesbaden: Springer Fachmedien Wiesbaden, 2014, S. 331–353.

[Trz19] Michael Trzesniowski. „Fahrzeugkonzept und Entwurf Vehicle Concept and Draft Design“. In: *Gesamtfahrzeug*. Wiesbaden: Springer Fachmedien Wiesbaden, 2019, S. 19–95.

[BSS10] Shaiju M. Belsus, Gopi Sankar und Amol Sharma. „Vehicle Reliability Estimation Model for Concept Vehicle Target Setting and Identification of Critical Parameters Influencing System Reliability“. In: Small Engine Technology Conference & Exposition. 28. Sep. 2010.

[Mat02] Kurt Matyas. „Ganzheitliche Optimierung durch individuelle Instandhaltungsstrategien“. In: *Industrie Management* 18.2 (2002), S. 13–16.

[MKS15] Tobias Meyer, James Kuria Kimotho und Walter Sextro. „Anforderungen an Condition-Monitoring-Verfahren zur Nutzung im zuverläsigkeitsgeregelten Betrieb adaptiver Systeme“. In: *27. Tagung Technische Zuverlässigkeit (TTZ 2015)-Entwicklung und Betrieb zuverlässiger Produkte* 2260 (2015).

[Mey+18] Tobias Meyer u. a. „Steigerung der Verlässlichkeit technischer Systeme“. In: *Steigerung der Intelligenz mechatronischer Systeme*. Hrsg. von Ansgar Trächtler und Jürgen Gausemeier. Berlin, Heidelberg: Springer Berlin Heidelberg, 2018, S. 193–213.

[HKV19] Frank Hutter, Lars Kotthoff und Joaquin Vanschoren, Hrsg. *Automated Machine Learning: Methods, Systems, Challenges*. The Springer Series on Challenges in Machine Learning. Cham: Springer International Publishing, 2019.

[Bät17] Daniel Bättig. *Angewandte Datenanalyse*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2017.

[Liu+02] Huan Liu u. a. „Discretization: An Enabling Technique“. In: *Data mining and knowledge discovery* 6.4 (2002), S. 393–423.

[TC19] Chih-Fong Tsai und Yu-Chi Chen. „The Optimal Combination of Feature Selection and Data Discretization: An Empirical Study“. In: *Information Sciences* 505 (Dez. 2019), S. 282–293.

[LLL17] Yun Li, Tao Li und Huan Liu. „Recent Advances in Feature Selection and Its Applications“. In: *Knowledge and Information Systems* 53.3 (Dez. 2017), S. 551–577.

[RB19] Beatriz Remeseiro und Veronica Bolon-Canedo. „A Review of Feature Selection Methods in Medical Applications“. In: *Computers in Biology and Medicine* 112 (Sep. 2019), S. 103375.

[Hui+17] Kar Hoou Hui u. a. „An Improved Wrapper-Based Feature Selection Method for Machinery Fault Diagnosis“. In: *PLOS ONE* 12.12 (20. Dez. 2017). Hrsg. von Quan Zou, e0189143.

[ZZX16] Bin Zhang, Lijun Zhang und Jinwu Xu. „Degradation Feature Selection for Remaining Useful Life Prediction of Rolling Element Bearings“. In: *Quality and Reliability Engineering International* 32.2 (März 2016), S. 547–554.

[Guo+00] Hong Guo u. a. „Automotive Signal Diagnostics Using Wavelets and Machine Learning“. In: *IEEE transactions on vehicular technology* 49.5 (2000), S. 1650–1662.

[Cro+03] J.A. Crossman u. a. „Automotive Signal Fault Diagnostics. I. Signal Fault Analysis, Signal Segmentation, Feature Extraction and Quasi-Optimal Feature Selection“. In: *IEEE Transactions on Vehicular Technology* 52.4 (Juli 2003), S. 1063–1075.

[Car+16] Jesus A. Carino u. a. „Enhanced Industrial Machinery Condition Monitoring Methodology Based on Novelty Detection and Multi-Modal Analysis“. In: *IEEE Access* 4 (2016), S. 7594–7604.

[Car+18] Jesus A. Carino u. a. „Fault Detection and Identification Methodology Under an Incremental Learning Framework Applied to Industrial Machinery“. In: *IEEE Access* 6 (2018), S. 49755–49766.

[Gar+17] Josh Gardner u. a. „Driving with Data: Modeling and Forecasting Vehicle Fleet Maintenance in Detroit“. In: *CoRR* abs/1710.06839 (2017). arXiv: 1710.06839.

[PNB11] Rune Prytz, Sławomir Nowaczyk und Stefan Byttner. „Towards Relation Discovery for Diagnostics“. In: *Proceedings of the First International Workshop on Data Mining for Service and Maintenance - KDD4Service '11*. The First International Workshop. San Diego, California: ACM Press, 2011, S. 23–27.

[MMG18] Artur Mrowca, Barbara Moser und Stephan Gunnemann. „Discovering Groups of Signals in In-Vehicle Network Traces for Redundancy Detection and Functional Grouping“. In: (2018), S. 16.

[Fug+19] Umberto Fugiglando u. a. „Driving Behavior Analysis through CAN Bus Data in an Uncontrolled Environment“. In: *IEEE Transactions on Intelligent Transportation Systems* 20.2 (Feb. 2019), S. 737–748.

[CCP12] L. Calabrese, G. Campanella und E. Proverbio. „Noise Removal by Cluster Analysis after Long Time AE Corrosion Monitoring of Steel Reinforcement in Concrete“. In: *Construction and Building Materials* 34 (Sep. 2012), S. 362–371.

[Sip13] Michael Sipser. *Introduction to the Theory of Computation*. 3rd edition, international edition. Australia: Cengage Learning, 2013.

[Feu+15] Matthias Feurer u. a. „Efficient and Robust Automated Machine Learning“. In: *Advances in Neural Information Processing Systems 28*. Hrsg. von C. Cortes u. a. Curran Associates, Inc., 2015, S. 2962–2970.

[VW16] Gottfried Vossen und Kurt-Ulrich Witt. *Grundkurs Theoretische Informatik*. Wiesbaden: Springer Fachmedien Wiesbaden, 2016.

[BD01] J. Bins und B.A. Draper. „Feature Selection from Huge Feature Sets“. In: *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*. Eighth IEEE International Conference on Computer Vision. Bd. 2. Vancouver, BC, Canada: IEEE Comput. Soc, 2001, S. 159–165.

[MDM11] Robert May, Graeme Dandy und Holger Maier. „Review of Input Variable Selection Methods for Artificial Neural Networks“. In: *Artificial Neural Networks - Methodological Advances and Biomedical Applications*. Hrsg. von Kenji Suzuki. InTech, 11. Apr. 2011.

[WA18] Ananto Setyo Wicaksono und Ahmad Afif. „Hyper Parameter Optimization Using Genetic Algorithm on Machine Learning Methods for Online News Popularity Prediction“. In: *International Journal of Advanced Computer Science and Applications* 9.12 (2018).

[YZ20] Tong Yu und Hong Zhu. *Hyper-Parameter Optimization: A Review of Algorithms and Applications*. 12. März 2020. arXiv: 2003.05689 [cs, stat]. URL: http://arxiv.org/abs/2003.05689 (besucht am 04. 09. 2020).

[LJA16] J. Lemley, F. Jagodzinski und R. Andonie. „Big Holes in Big Data: A Monte Carlo Algorithm for Detecting Large Hyper-Rectangles in High Dimensional Data“. In: *2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC)*. Bd. 1. Juni 2016, S. 563–571.

[DKW18] Odeh Dababneh, Timoleon Kipouros und James Whidborne. „Application of an Efficient Gradient-Based Optimization Strategy for Aircraft Wing Structures“. In: *Aerospace* 5.1 (4. Jan. 2018).

[MW97] W.G. Macready und D.H. Wolpert. „No Free Lunch Theorems for Optimization“. In: *IEEE transactions on evolutionary computation*. IEEE Transactions on Evolutionary Computation 1.1 (1997), S. 67–82.

[FPS96] Usama Fayyad, Gregory Piatetsky-Shapiro und Padhraic Smyth. „From Data Mining to Knowledge Discovery in Databases“. In: *AI Magazine* 17.3 (März 1996), S. 37.

[Bor97] Christian Borgelt. „Einführung in Datenanalyse Und Data Mining Mit Intelligenten Technologien“. In: *Seminar zu Anwendungen in Datenanalyse und Data Mining*. Bergholz-Rehbrücke (1997).

[SSG11] Hemlata Sahu, Shalini Shrma und Seema Gondhalakar. „A Brief Overview on Data Mining Survey“. In: *International Journal of Computer Technology and Electronics Engineering (IJCTEE)* 1.3 (2011), S. 114–121.

[ES00a] Martin Ester und Jörg Sander. „Einleitung“. In: *Knowledge Discovery in Databases: Techniken Und Anwendungen*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, S. 1–13.

[CHT09] A. K. Choudhary, J. A. Harding und M. K. Tiwari. „Data Mining in Manufacturing: A Review Based on the Kind of Knowledge“. In: *Journal of Intelligent Manufacturing* 20.5 (Okt. 2009), S. 501–521.

[LC20] Uwe Lämmel und Jürgen Cleve. *Data Mining*. Berlin: De Gruyter; 2020.

[GB15] T. Göpfert und A. Breiter. „Knowledge Discovery in Big Data: Herausforderungen Durch Big Data Im Prozess Der Wissensgewinnung Am Beispiel Des CRISP-DM“. In: *GI-Jahrestagung*. 2015.

[AS08] A. Azevedo und M. Santos. „KDD, SEMMA and CRISP-DM: A Parallel Overview“. In: *IADIS European Conf. Data Mining*. 2008.

[HPK12] Jiawei Han, Jian Pei und Micheline Kamber. *Data Mining: Concepts and Techniques*. Amsterdam: Elsevier; 2012.

[TL19] Tiedo Tinga und Richard Loendersloot. „Physical Model-Based Prognostics and Health Monitoring to Enable Predictive Maintenance“. In: *Predictive Maintenance in Dynamic Systems: Advanced Methods, Decision Support Tools and Real-World Applications*. Hrsg. von Edwin Lughofer und Moamar Sayed-Mouchaweh. Cham: Springer International Publishing, 2019, S. 313–353.

[HK17] Andreas Handl und Torben Kuhlenkasper. *Multivariate Analysemethoden*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2017.

[SSM10] Niranjan Subrahmanya, Yung C. Shin und Peter H. Meckl. „A Bayesian Machine Learning Method for Sensor Selection and Fusion with Application to On-Board Fault Diagnostics“. In: *Mechanical Systems and Signal Processing* 24.1 (Jan. 2010), S. 182–192.

[WGL14] Wu He, Gongjun Yan und Li Da Xu. „Developing Vehicular Data Cloud Services in the IoT Environment“. In: *IEEE Transactions on Industrial Informatics* 10.2 (Mai 2014), S. 1587–1595.

[Fil+10] Dimitar P. Filev u. a. „An Industrial Strength Novelty Detection Framework for Autonomous Equipment Monitoring and Diagnostics“. In: *IEEE Transactions on Industrial Informatics* 6.4 (Nov. 2010), S. 767–779.

[SL19] Sigurd Schacht und Carsten Lanquillon, Hrsg. *Blockchain und maschinelles Lernen: Wie das maschinelle Lernen und die Distributed-Ledger-Technologie voneinander profitieren*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2019.

[Won15] Tzu-Tsung Wong. „Performance Evaluation of Classification Algorithms by K-Fold and Leave-One-out Cross Validation“. In: *Pattern Recognition* 48.9 (Sep. 2015), S. 2839–2846.

[SLA12] Jasper Snoek, Hugo Larochelle und Ryan P Adams. „Practical Bayesian Optimization of Machine Learning Algorithms“. In: *Advances in Neural Information Processing Systems 25*. Hrsg. von F. Pereira u. a. Curran Associates, Inc., 2012, S. 2951–2959.

[Ber+11] James S Bergstra u. a. „Algorithms for Hyper-Parameter Optimization“. In: *Advances in Neural Information Processing Systems*. 2011, S. 2546–2554.

[Egg+15] Katharina Eggensperger u. a. „Efficient Benchmarking of Hyperparameter Optimizers via Surrogates“. In: *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*. AAAI'15. Austin, Texas: AAAI Press, 2015, S. 1114–1120.

[HHL11] Frank Hutter, Holger H. Hoos und Kevin Leyton-Brown. „Sequential Model-Based Optimization for General Algorithm Configuration“. In: *Proceedings of the 5th International Conference on Learning and Intelligent Optimization*. LION'05. Berlin, Heidelberg: Springer-Verlag, 2011, S. 507–523.

[Egg+13] Katharina Eggensperger u. a. „Towards an Empirical Foundation for Assessing Bayesian Optimization of Hyperparameters“. In: *NIPS workshop on Bayesian Optimization in Theory and Practice. Vol. 10* (2013).

[Sno+14] Jasper Snoek u. a. „Input Warping for Bayesian Optimization of Non-Stationary Functions“. In: *Proceedings of the 31st International Conference on Machine Learning*. Hrsg. von Eric P. Xing und Tony Jebara. Bd. 32. Proceedings of Machine Learning Research 2. Bejing, China: PMLR, 22.–24. Juni 2014, S. 1674–1682.

[Ili+17] Ilija Ilievski u. a. *Efficient Hyperparameter Optimization of Deep Learning Algorithms Using Deterministic RBF Surrogates*. 20. Jan. 2017. arXiv: 1607.08316 [cs, stat]. URL: http://arxiv.org/abs/1607.08316 (besucht am 22. 12. 2020).

[MML19] Francisco Madrigal, Camille Maurice und Frédéric Lerasle. „Hyper-Parameter Optimization Tools Comparison for Multiple Object Tracking Applications“. In: *Machine Vision and Applications* 30.2 (März 2019), S. 269–289.

[WWL20] Alexander Wendt, Marco Wuschnig und Martin Lechner. „Speeding up Common Hyperparameter Optimization Methods by a Two-Phase-Search“. In: *IECON 2020 the 46th Annual Conference of the IEEE Industrial Electronics Society*. 2020, S. 517–522.

[BYC13] James Bergstra, Dan Yamins und David D. Cox. „Making a Science of Model Search: Hyperparameter Optimization in Hundreds of Dimensions for Vision Architectures“. In: *TProc. of the 30th International Conference on Machine Learning (ICML 2013)* (Juni 2013).

[Aki+19] Takuya Akiba u. a. „Optuna: A next-Generation Hyperparameter Optimization Framework“. In: *CoRR* abs/1907.10902 (2019). arXiv: 1907.10902.

[Car15] Miguel Á Carreira-Perpiñán. *A Review of Mean-Shift Algorithms for Clustering*. 2. März 2015. arXiv: 1503.00687 [cs, stat]. URL: http://arxiv.org/abs/1503.00687 (besucht am 26. 08. 2020).

[KR05] Leonard Kaufman und Peter J. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis, Wiley Series in Probability and Mathematical Statistics, Wiley-Interscience Paperback Series*. Hoboken, N.J.: Wiley, 2005.

[ES00b] Martin Ester und Jörg Sander. „Clustering“. In: *Knowledge discovery in databases: Techniken und anwendungen*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, S. 45–105.

[CS08] S. Chandrakala und C. Chandra Sekhar. „A Density Based Method for Multivariate Time Series Clustering in Kernel Feature Space“. In: *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*. 2008 IEEE International Joint Conference on Neural Networks (IJCNN 2008 - Hong Kong). Hong Kong, China: IEEE, Juni 2008, S. 1885–1890.