Methodisches Vorgehen zur Realisierung von maschinellen Lernprojekten im Mittelstand

Dissertation
zur Erlangung des Grades
des Doktors der Ingenieurswissenschaften
der Naturwissenschaftlich-Technischen Fakultät
der Universität des Saarlandes

von
Christopher Schnur, M.Sc. & M.Eng.

Saarbrücken
2024

Tag des Kolloquiums:	14.04.2025
Dekan:	Prof. Dr.-Ing. Dirk Bähre
Berichterstatter:	Prof. Dr. rer. nat. Andreas Schütze
	Prof. Dr.-Ing. Rainer Müller
Vorsitz:	Prof. Dr.-Ing. Stefan Seelecke
Akad. Mitarbeiter:	Dr.-Ing. Oliver Maurer

"Wer hohe Türme bauen will, muss lange beim Fundament verweilen."

Anton Bruckner

Zur besseren Lesbarkeit wird in dieser Dissertation das generische Maskulinum verwendet.

Zusammenfassung

Im Rahmen dieser Arbeit wurde ein methodisches Vorgehen zur Realisierung von maschinellen Lernprojekten im Mittelstand entwickelt. Oftmals sind zeitliche, finanzielle und personelle Ressourcen hier stark beschränkt, wodurch sich ein systematischer Wettbewerbsnachteil bzw. Entwicklungsstau ergibt. Um es unerfahrenen Anwendern dennoch zu ermöglichen Datenanalyseprojekte mittels maschinellem Lernen (ML) durchzuführen, wird ein Konzept eines persönlichen Informationsassistenten (PIA) entwickelt. Kernelement ist eine Checkliste, die den Anwender ganzheitlich bei einem maschinellen Lernprojekt begleitet und folgende Themengebiete abdeckt: Vorbereitung und Projektplanung, Mess- und Datenplanung, Datenaufnahme, Datenprüfung und -bereinigung, Datenauswertung und Modellbildung sowie den Projektabschluss. Neben einer Einführung in die Thematik bietet die Checkliste Tipps, Hinweise und weiterführende Literatur. Der PIA besteht zudem aus zwei weiteren Modulen, der Zugänglichkeit von Daten und Wissen und der Datenanalyse. Für beide Module wurden Konzepte aus Wissenschaft und Forschung adaptiert und in einen industrienahen Kontext gebracht. Um die Anwender bei der Modellbildung mittels ML zu unterstützen, wurde eine Toolbox in den PIA integriert, welche automatisiert komplementäre Algorithmen testet und validiert. Die entwickelte Methodik wurde in zwei industrienahen Anwendungsfällen validiert.

Abstract

In this work, a methodological approach was developed which enables unexperienced workers in small and medium-sized enterprises (SME) to perform machine learning projects. Due to the limited resources in SMEs in time, staff and money available, a systematic competitive disadvantage is observed. To enable especially inexperienced users to perform a data analysis project with machine learning, the concept of a Personal Information Assistant (PIA) is introduced. Core element of PIA is a checklist, which comprehensively guides the user through a machine learning project covering the topics preparation and project planning, measurement and data planning, data acquisition, data check and cleansing, data evaluation and modeling as well as project completion. After a short introduction into each topic, the checklist offers tips, remarks and further literature. Besides the checklist, PIA consists of the two additional modules accessibility of data and knowledge and data analysis. For both modules, concepts from research have been adapted into an industrial context. To support users in building a machine learning model, a toolbox is integrated in the assistant which automatically tests and validates complementary algorithms. The developed methodology was validated in two industry relevant use cases.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Quellcodeverzeichnis

Abkürzungsverzeichnis

ADC	Analog-to-digital converter
ALA	Adaptive Lineare Approximation
BDW	Beste Daubechies Wavelets
BFC	Beste Fourier Koeffizienten
BMWi	Bundesministerium für Wirtschaft und Energie
BPFI	Innenring-Überrollfrequenz
CNN	Convolutional Neural Networks
cRIO	CompactRIO
CRISP-DM	Cross-Industry Standard Process for Data Mining
CWRU	Case Western Reserve University
DCMI	The Dublin Core™Metadata Initiative
DIN	Deutsche Institut für Normung
DoE	Design of Experiments
ERP	Enterprise Resource Planning
FAIR	Findable, Accessible, Interoperable, Re-usable
FPGA	Field Programmable Gate Array
FV	Fehlervermeidung
GUI	grafische Benutzeroberfläche
GUM	Guide to the Expression of Uncertainty in Measurement
I4.0	Industry 4.0
ID	Identifikationsnummer
IQ	Informationsqualität
ISO	Internationale Organisation für Normung
KDD	Knowledge Discovery in Databases
KI	künstliche Intelligenz
knn	k-nächste Nachbarn
LDA	Lineare Diskriminanzanalyse
LOGOCV	Leave-One-Group-Out-Cross-Validation
m4i	Metadata4Ing
MessMo	Messtechnisch gestützte Montage

ML	maschinelles Lernen
ML-Montage-Checkliste	Checkliste Mess- und Datenplanung für das maschinelle Lernen in der Montage
ML-Toolbox	Automatisierte ML-Toolbox für zyklische Daten
MuD	Mess- und Datenplanung
NASA-BD	NASA Bearing Dataset
NFDI	Nationale Forschungsdateninfrastruktur
NFDI4Ing	Nationale Forschungsdateninfrastruktur für die Ingenieurwissenschaften
NN	Neuronale Netze
PC	Hauptkomponente
PCA	Principal Component Analysis
PIA	Persönlicher Informationsassistent
PoE	Power over Ethernet
PU	Paderborn University
RAID	Redundant Array of Independent Disks
RFE	Recursive Feature Elimination
RFESVM	Recursive Feature Elimination Support Vector Machine
SM	Statistische Momente
SVM	Support Vector Machine
TDMS	Technical Data Management Streaming
UA-ML-Toolbox	Uncertainty-Aware Automated Machine Learning Toolbox
UTC	Universal Time Coordinated
UWD	Ursache-Wirkungs-Diagramm
VProSaar	Verteilte Produktion für die saarländische Automotivindustrie: Nachhaltig, Vernetzt, Resilient
W3C	World Wide Web Consortium
WaMo	Wandelbares Montagesystem
ZeMA	Zentrum für Mechatronik und Automatisierungstechnik gGmbH

1 Einleitung und Motivation

Im Rahmen der vierten industriellen Revolution bieten künstliche Intelligenz (KI) und maschinelles Lernen (ML) Unternehmen ein großes Potenzial zur Effizienzsteigerung [1]. Typische Aufgaben sind hier bspw. die Zustandsüberwachung (engl. Condition Monitoring), welche anhand von Daten den Zustand von Betriebsmitteln oder Anlagen überwacht, die vorausschauende Wartung (engl. Predictive Maintenance), welche sich mit der datengetriebenen Optimierung des Wartungszeitpunkts befasst und die Qualitätsvorhersage von Produkten [2]. Um robuste KI- bzw. ML-Modelle zu erzeugen, müssen die zugrundeliegenden Daten eine hohe Qualität aufweisen und in ausreichender Quantität vorhanden sein. Dies ist in industriellen Anwendungen nicht immer gegeben. So sehen nach einer Studie der IDG Research Services [3] mit 343 befragten Unternehmen 34,1 % davon die mangelnde Datenqualität als eine der größten Hürden für die Anwendung von ML.

Der Lehrstuhl für Messtechnik der Universität des Saarlandes beschäftigt sich seit mehr als 10 Jahren mit der Anwendung von ML-Algorithmen im industriellen Kontext. In Kooperation mit dem Zentrum für Mechatronik und Automatisierungstechnik gGmbH (ZeMA) wurden über diesen Zeitraum zahlreiche Forschungsprojekte und Direktaufträge mit Partnern aus dem Mittelstand und Großkonzernen durchgeführt. Oftmals war innerhalb dieser Projekte die Datenqualität bereits bestehender Datensätze für die Anwendung von ML-Algorithmen nicht ausreichend oder musste durch hohen manuellen Aufwand aufbereitet werden. Exemplarisch für die gegenwärtige Situation in der Industrie war das Forschungsprojekt Messtechnisch gestützte Montage (MessMo), welches vom Europäischen Fonds für regionale Entwicklung (EFRE) unter dem Projektkennzeichen 14.2.1.4–2017/5 gefördert wurde. Ziele des Projektes waren u.a. die ML-gestützte Prozessüberwachung, die Qualitätsvorhersage von Produkten und die Zustandsüberwachung anhand von drei Anwendungsszenarien innerhalb der Produktion eines Großkonzerns. Abbildung 1.1 zeigt eine anonymisierte Nachbildung eines kritischen Prüfprozesses aus einem der drei Anwendungsfälle.

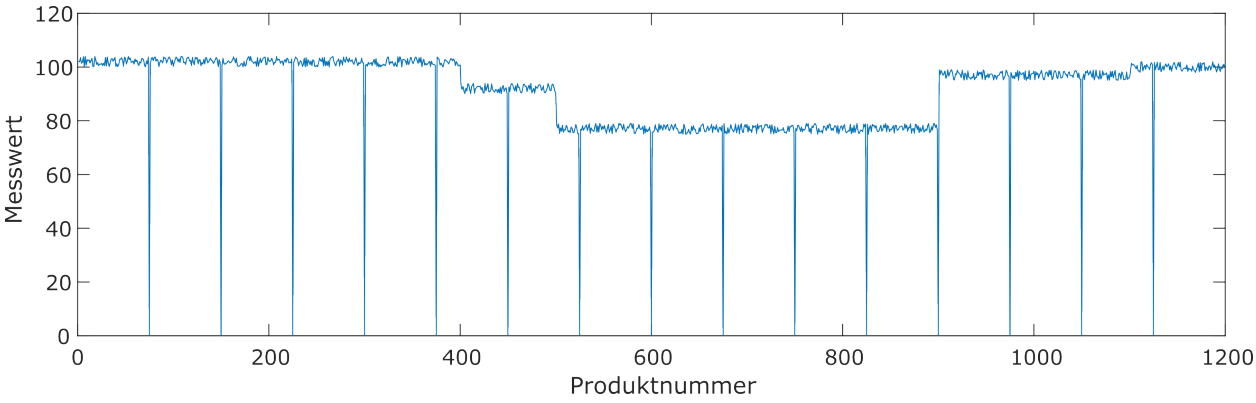


Abbildung 1.1: Anonymisierte Nachbildung eines realen kritischen Prozesses, adaptiert aus [4].

Die dargestellten Messwerte weisen statt eines konstanten Wertes mehrere Mittelwertverschiebungen auf. Weiterhin tritt in regelmäßigen Abständen systematisch der Messwert 0 auf. Aufgrund fehlender Metadaten und Ansprechpartner konnten die Ursachen für die Mittelwertverschiebungen und die systematischen Messungen des Messwertes 0 erst kurz vor Ende der Projektlaufzeit und durch hohen manuellen Aufwand identifiziert werden. Dieses Beispiel war stellvertretend für die niedrige Qualität der im Projekt vorhandenen Daten. So wurde als Projektergebnis abgeleitet, dass der Ablauf eines (ML-)Projekts ganzheitlich betrachtet werden und ein Fokus auf der Mess- und Datenplanung liegen muss. Der hohe manuelle Aufwand in der Aufbereitung der Daten in Kombination mit der niedrigen Datenqualität erschweren die Anwendung von KI- bzw. ML-Algorithmen und bieten ein signifikantes Verbesserungspotenzial. Weiterhin entstehen Daten in der Industrie primär als Nebenprodukt von Steuerungs-, Regelungs- oder Prüfprozessen, während in der Forschung oftmals die Gewinnung hochqualitativer Daten das eigentliche Ziel ist.

Dieser Umstand hat Initiativen wie z.B. die Nationale Forschungsdateninfrastruktur (NFDI) [5] oder Konzepte wie die FAIR-Prinzipien [6] hervorgebracht (siehe Abschnitt 2.1), welche den Fokus auf die Aufzeichnung und Gestaltung hochwertiger Daten in der Forschung legen. Erwähnenswert ist an dieser Stelle die Nationale Forschungsdateninfrastruktur für die Ingenieurwissenschaften (NFDI4Ing), ein Konsortium der NFDI, welches speziell für die Ingenieurwissenschaften eingerichtet wurde. Vereinzelt existieren auch bereits Ansätze für die Industrie, wie bspw. der Cross-Industry Standard Process for Data Mining (CRISP-DM) [7] (siehe Abschnitt 2.3.1), welche die Datenqualität und die Wissensgewinnung in einem industriellen Kontext behandeln, jedoch sind diese oftmals sehr umfangreich und spezifisch und somit für unerfahrene Anwender nur schwer und unter hohem Aufwand umzusetzen. Auch

die erwähnten Forschungsansätze, wie bspw. die FAIR-Prinzipien, sind nicht ohne Anpassungen für die industrielle Anwendung geeignet.

1.1 Ziel der Dissertation

Die oben genannten Gegebenheiten finden sich in ausgeprägter Form in mittelständischen Unternehmen wieder, da hier üblicherweise keine dezidierten Abteilungen für bspw. die Anwendung und Entwicklung von KI- und ML-Algorithmen zur Verfügung stehen und auch personelle und finanzielle Ressourcen beschränkt sind. Übergeordnetes Ziel dieser Dissertation ist daher die Beantwortung der Forschungsfrage:
Wie können Unternehmen im Mittelstand maschinelle Lernprojekte realisieren, ohne auf spezialisierte ML-Fachkräfte angewiesen zu sein?
Um ein ML-Projekt erfolgreich umzusetzen, sind eine hohe Datenqualität und der Einsatz geeigneter ML-Methoden essenziell. Gemäß der Ergebnisse des Projekts MessMo wird der Fokus auf die Mess- und Datenplanung gelegt, um eine hohe Datenqualität sicherzustellen. Abgeleitet von diesen Faktoren und dem übergeordneten Ziel ergeben sich daraus folgende Forschungsfragen:

- **Forschungsfrage 1:** Wie können aktuelle Forschungskonzepte zur Generierung hochwertiger Daten in die Industrie übertragen werden?
- **Forschungsfrage 2:** Wie können unerfahrene industrielle Anwender befähigt werden, hochwertige Daten aufzuzeichnen?
- **Forschungsfrage 3:** Wie können unerfahrene Anwender unterstützt werden, eine Datenanalyse mittels ML durchzuführen?

Zur Beantwortung der Forschungsfragen wurde im Rahmen dieser Dissertation eine Checkliste mit zugehörigem Ablaufplan entwickelt, die Anwender ganzheitlich bei der Bearbeitung eines ML-Projektes unterstützt. Die Checkliste dient als zentrales Element in dem anschließend entwickelten Konzept eines persönlichen Informationsassistenten (PIA). PIA fungiert dabei als Plattform, die dem Anwender neben der integrierten Checkliste Zugang zu Daten und Wissen bietet und eine automatisierte ML-Toolbox zur Datenanalyse [8] enthält. Abbildung 1.2 visualisiert das Vorgehen der Dissertation.

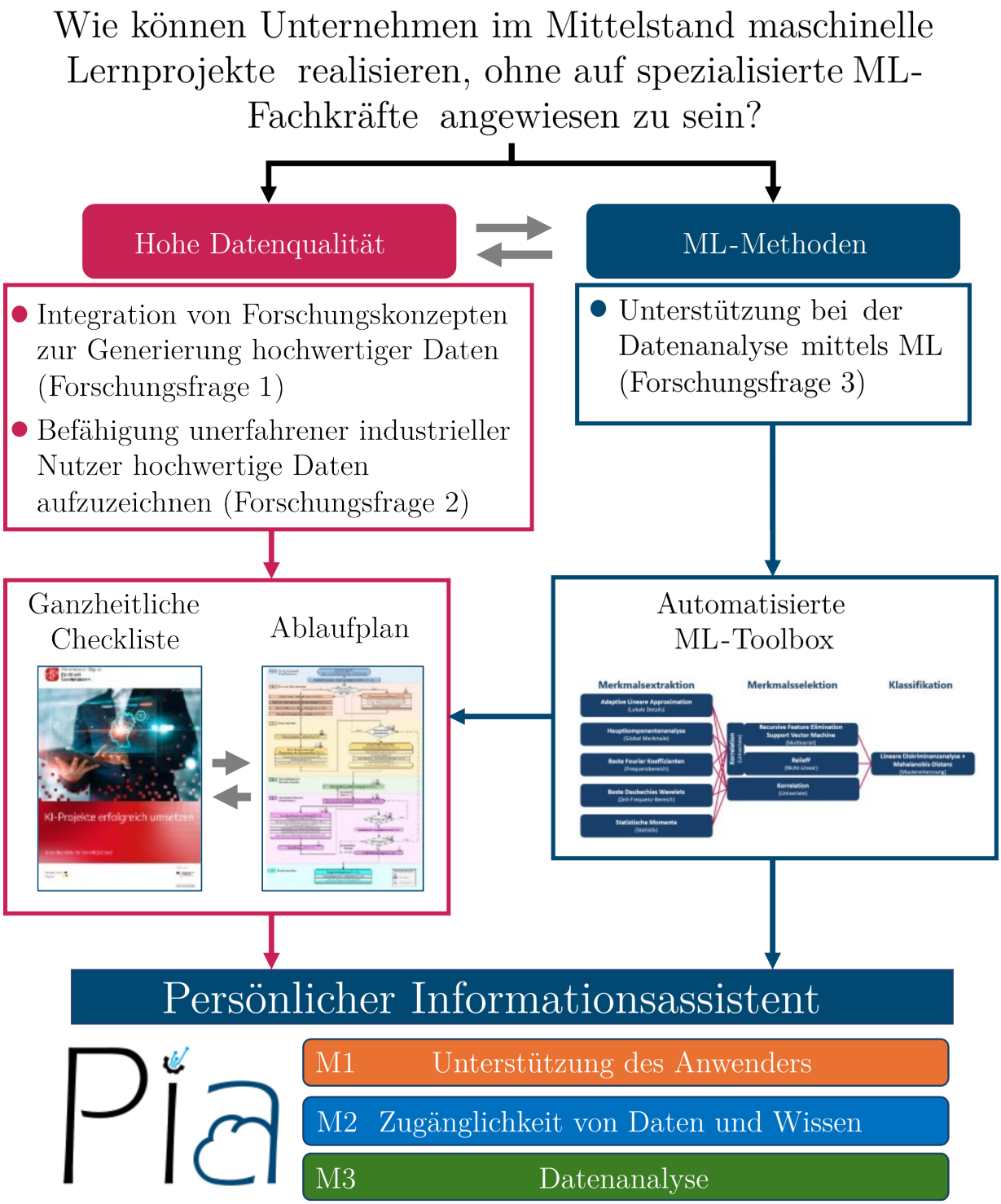


Abbildung 1.2: Grafischer Überblick der Dissertation.

1.2 Aufbau der Dissertation

Zunächst werden die notwendigen Grundlagen mit dem Stand der Technik in Kapitel 2 vorgestellt. Da für die Durchführung eines ML-Projektes ein breites Spektrum an Wissen aus verschiedensten (Forschungs-)Bereichen benötigt wird, ist Kapitel 2 entsprechend divers gestaltet und kann lediglich eine Übersicht über

die jeweiligen Bereiche abbilden. Im nachfolgenden Kapitel 3 wird eine Checkliste entwickelt, welche unerfahrene Anwender bei der Bearbeitung eines ML-Projektes unterstützt und ihnen ermöglicht, hochqualitative Daten aufzuzeichnen. Anschließend wird in Kapitel 4 ein Konzept für einen Persönlichen Informationsassistenten (PIA) vorgestellt, welcher die entwickelte Checkliste aufgreift und diese auf einen ganzheitlichen Ansatz mit graphischer Benutzeroberfläche erweitert. Die Validierung der vorgestellten Methodik erfolgt anhand von zwei Anwendungsfällen in Kapitel 5. Der erste Anwendungsfall behandelt die Erzeugung eines robusten ML-Modells zur Erkennung von Innenringschäden an Zylinderrollenlagern. Dazu wurde am Lehrstuhl für Messtechnik der Universität des Saarlandes ein spezieller Prüfstand entwickelt. Der zweite Anwendungsfall behandelt die Erzeugung eines robusten ML-Modells zur Erkennung von Beschädigungen an einem Schrauberbit an einer Montageanlage. Bei der Montageanlage handelt es sich um das Wandelbare Montagesystem (WaMo) des Lehrstuhls für Montagesysteme am ZeMA. Das WaMo bietet den Vorteil einer industrienahen Erprobung der vorgestellten Methodik, ohne eine Beeinträchtigung einer Anlage im Betrieb hervorzurufen. Nach Abschluss von Kapitel 3, Kapitel 4 und Kapitel 5 wird jeweils ein Zwischenfazit gezogen. Ein übergeordnetes Fazit der vorgestellten Methodik folgt in Kapitel 6. Abschließend erfolgt in Kapitel 7 eine Zusammenfassung der vorgestellten Methodik und ein Ausblick auf zukünftige Arbeiten.

1.3 Grenzen der Arbeit

Aufgrund der Vielfalt anfallender Daten in der Industrie wird der Fokus im Rahmen dieser Dissertation auf zeitdiskrete Daten gelegt. Diese stammen üblicherweise von Sensoren, welche zu äquidistanten Zeitpunkten Messwerte aufzeichnen, wie z.B. Temperatur- oder Beschleunigungssensoren. Grundsätzlich sind die vorgestellten Lösungsansätze auch für andere Datenarten, wie z.B. Bilddaten gültig, jedoch bedarf es hier mehrerer Anpassungen, bspw. bei der Algorithmenauswahl.

Diese Arbeit bedient sich verschiedenster (Forschungs-)Bereiche und generiert einen Mehrwert durch deren Kombination. Die entsprechenden Grundlagen und der aktuelle Stand der Technik wurden sorgfältig und nach bestem Wissen des Autors für diese Dissertation aufgearbeitet. Dennoch stellen die jeweiligen Abschnitte lediglich einen Überblick der behandelten Themengebiete dar. Hier wurde ein Trade-off zwischen Verständlichkeit und Vollständigkeit eingegangen mit dem Ziel, unerfahrene Anwender zur Durchführung eines ML-Projektes zu befähigen und dabei ein breites Spektrum

an Bereichen abzudecken. Um Anwender nicht zu überfordern, werden zudem gängige Methoden zur Durchführung der ML-Projekte vorgeschlagen, welche sich am Lehrstuhl für Messtechnik für die Bearbeitung industrieller Projekte bewährt haben. Erfahrene Anwender können die ausgewählten Methoden auch durch bspw. branchenspezifische Methoden und Ansätze austauschen bzw. ergänzen.

Im forschungslastigen Anwendungsfall 1 erfolgt die Anwendung der entwickelten Methodik durch den Autor dieser Dissertation. In dem industrienahen Anwendungsfall 2 erfolgt die Anwendung der Methodik durch ein arbeitsgruppen-übergreifendes Team und der Autor begleitet und unterstützt lediglich das ML-Projekt. Die Anwendung der Methodik im Mittelstand als dritter Anwendungsfall wurde angestrebt, konnte jedoch nicht innerhalb dieser Dissertation behandelt werden. Obwohl die vorgestellten Lösungsansätze bei mittelständischen Unternehmen Anklang fanden und bpsw. auch mehrere Exemplare der Checkliste zur Verfügung gestellt werden konnten, waren die angefragten Unternehmen aus dem Mittelstand aus Gründen des Datenschutzes, der Bewahrung von Geschäftsgeheimnissen oder mangelnder zeitlicher und monetärer Ressourcen vorerst nicht bereit, ein ML-Projekt mit einem externen Partner durchzuführen.

2 Grundlagen und Stand der Technik

Im Kapitel Grundlagen und Stand der Technik werden die im Rahmen dieser Dissertation verwendeten Methoden und Begrifflichkeiten eingeführt. Durch die Vielzahl der behandelten Themengebiete wird auf eine tiefgehende Beschreibung dieser zugunsten von Klarheit und Übersichtlichkeit verzichtet. Spezifische technologische Grundlagen der Anwendungsszenarien sind in Kapitel 5 nachgelagert.

2.1 Daten in der Industrie

In der heutigen Zeit spielen Daten nicht mehr nur in der Wissenschaft, sondern auch zunehmend in der Industrie eine immer größere Rolle und werden sogar als das neue Gold bezeichnet [9, 10]. Lange Zeit wurden Daten in der Industrie primär zu Steuerungs- und Regelungsaufgaben bzw. zur Qualitätssicherung und -kontrolle genutzt. Im Rahmen der vierten industriellen Revolution, auch Industrie 4.0 (I4.0) genannt, steigt die Relevanz von Daten jedoch zunehmend. Speziell durch die rasche Weiterentwicklung der Leistungsfähigkeit von Computern und den Fortschritt in der künstlichen Intelligenz (KI) und des maschinellen Lernens (ML) steigt das Potenzial der Nutzung von Daten über die ursprünglichen Steuerungs- und Regelungsaufgaben hinaus, hin zu u.a. Condition Monitoring (deutsch: Zustandsüberwachung) und Predictive Maintenance (deutsch: Vorausschauende Wartung). Einerseits bietet sich durch diesen Fortschritt die Chance der Kostenoptimierung und Qualitätsverbesserung, den häufig primären Zielen der Industrie; andererseits entstehen dadurch diverse Herausforderungen. Denn im Gegensatz zur Forschung, bei der oftmals die Aufzeichnung von Daten das primäre Ziel ist und deren Daten dadurch typischerweise eine hohe Datenqualität aufweisen, sind industrielle Daten, speziell von Bestandsanlagen im sog. Brownfield, häufig lediglich ein Nebenprodukt der oben genannten Steuerungs- und Regelungsaufgaben. Der fehlende Fokus auf die Daten führt tendenziell zu einer niedrigeren Datenqualität bzw. einem geringen Informationsgehalt der Daten. Dies geht häufig mit fehlenden Metadaten („Daten über Daten“) bzw. einer unzureichenden Annotation (Kennzeichnung von Daten) einher. Weiterhin ist in der Industrie oftmals zwar eine hohe Datenmenge

vorhanden, aber diese stark einseitig ausgeprägt. So enthalten z.B. die an einer Produktionslinie mit geringer Ausschussquote aufgezeichneten Daten einen hohen Anteil „Gut-Daten“ und nur einen geringen Anteil „Schlecht-Daten“, was für das Modell eine Hürde zur Erkennung von Ausschuss darstellen kann.

In einer von IDG Research Services im Jahr 2019 durchgeführten Studie [3] (im Folgenden IDG-Studie) mit 343 befragten Unternehmen sehen 34,1 % der Unternehmen die mangelnde Datenqualität als größte Hürde für die Anwendung von ML. Ursache hierfür sind diverse Herausforderungen, welche in der Industrie auftreten. So definiert [11] im Kontext der Produktion 19 Herausforderungen nach Wilhelm und gliedert diese in die vier Kategorien produktionsspezifische Herausforderungen, datenspezifische Herausforderungen, analytische Herausforderungen und softwarespezifische Herausforderungen ein. Tabelle 2.1 zeigt eine Auflistung der Herausforderungen nach Wilhelm.

Tabelle 2.1: Herausforderungen für die Qualitätskontrolle in der Produktion nach Wilhelm [11].

Nr.	**Herausforderungen für die Qualitätskontrolle**
	Produktionsspezifische Herausforderungen
P1.	Heterogene Produktfamilien und Betriebsmittel
P2.	Vielfältige Fehlertypen
P3.	Ungleichmäßige Datenverteilungen
P4.	Nicht-lineare Produktionsprozesse und -schleifen
P5.	Konzeptverschiebungen
P6.	Kostensensitives Modellieren
	Datenspezifische Herausforderungen
D1.	Komplexe Datenfusion von multimodalen Datenquellen
D2.	Fehlende Eindeutigkeit und Zuordenbarkeit
D3.	Unsynchronisierte Zeitstempel entlang der Produktionskette
D4	Doppelte und ungültige Messwerte
D5.	Fehlende Spezifikationen und Hintergründe von Daten
D6.	Unzureichende Datenaufnahme
D7.	Nicht dokumentierte Systemänderungen
	Analytische Herausforderungen
A1.	Fehlendes Vertrauen in ML-Modelle und analytische Ergebnisse
A2.	Fehlende Rückverfolgbarkeit in Analyseergebnisse und ihren Ursachen
A3.	Verwaltung zahlreicher maschineller Lernmodelle
A4.	Feature Engineering und Auswahl von Modellen und ihren Parametern
	Softwarespezifische Herausforderungen
S1.	Fehlende moderne Datenschnittstellen
S2.	Programmierung individueller Lösungen

Eine detaillierte Beschreibung der jeweiligen Herausforderungen nach Wilhelm findet sich in Anhang A.1.1 in Tabelle A.1 - Tabelle A.4.

Um die Nutzbarkeit von Daten in der Industrie sicherzustellen, wurden bereits mehrere Initiativen auf nationaler und internationaler Ebene gestartet. Auf internationaler Ebene sind bspw. die FAIR (Findable, Accessible, Interoperable, Re-usable) Prinzipien [6] erwähnenswert, welche die Auffindbarkeit, Zugänglichkeit, Interoperabilität und Wiederverwendbarkeit von Daten sicherstellen sollen. Die FAIR-Prinzipien werden in Abschnitt 2.2 im Detail vorgestellt. Auf nationaler Ebene wurde z.B. 2019 von der Deutschen Forschungsgemeinschaft (DFG) die Initiative Nationale Forschungsdateninfrastruktur (NFDI) [5] gestartet, welche sich auf die Forschungsbereiche der Geisteswissenschaften, Ingenieurwissenschaften, Lebenswissenschaften, Naturwissenschaften und Sozialwissenschaften fokussiert hat, mit dem Ziel, Datenbestände für das deutsche Wissenschaftssystem systematisch zu erschließen und zu vernetzen. Besonders relevant für die Industrie ist der Forschungsbereich Nationale Forschungsdateninfrastruktur für die Ingenieurwissenschaften (NFDI4Ing), welcher u.a. an Methoden zur Steigerung der FAIRness von Daten und Metadaten sowie deren Maschinenlesbarkeit und -interpretierbarkeit forscht. Um die bisherigen Ergebnisse der NFDI4Ing in die Industrie zu transferieren, wurde im Jahr 2023 die Sektion Industry Engagement [12] eingerichtet.

Weitere Ansätze, KI in die Industrie zu transferieren, werden nachfolgend beschrieben:

- Der *Crashkurs KI im Unternehmen* [13] vermittelt Grundlagen der KI, gängige Methoden und Abläufe im KI-Projektmanagement, sowie grundlegende Kompetenzen, Ergebnisse von KI-Projekten zu beurteilen. Allerdings bietet dieser nur eine begrenzte Unterstützung bei der konkreten Umsetzung von KI-Projekten.

- *Der Leitfaden für qualitativ hochwertige Daten und Metadaten* [14] fokussiert die Qualität von Daten und Metadaten und stellt verschiedene Qualitätsdimensionen und Schemata zu deren Bewertung vor. Jedoch wird die Anwendung der Konzepte im Gesamtkontext eines KI-Projektes nicht ausreichend betrachtet.

- Der *Leitfaden Künstliche Intelligenz zur Umsetzung von Industrie 4.0 im Mittelstand* [15] untersucht den Einsatz von KI in kleinen und mittleren Unternehmen und bietet grundlegende Vorgehensweisen und Anwendungsbeispiele. Dabei bleibt der Leitfaden eher auf einer strategischen Ebene und bietet weniger tiefgehende Anleitungen für die Praxis.

- Der Leitfaden *Künstliche Intelligenz – Potenziale und Umsetzungen im Mittelstand* [16] bietet einen praxisnahen Überblick über die Potenziale von KI, erläutert zentrale Begriffe und veranschaulicht deren Anwendung anhand konkreter Beispiele für mittelständische Unternehmen. Jedoch bietet der Leitfaden keine detaillierten Handlungsempfehlungen zur praktischen Durchführung.

Die beschriebenen Ansätze können Unternehmen wertvolle Grundlagen und Informationen bieten. Gerade in mittelständischen Unternehmen fehlen jedoch häufig die Ressourcen, um sich intensiv mit der Literatur auseinanderzusetzen. Daher besteht die Notwendigkeit eines Ansatzes, der KI-Projekte ganzheitlich betrachtet und dem Anwender übersichtlich strukturierte sowie praxisnahe Handlungsempfehlungen liefert.

2.2 FAIR Data

Mit dem Ziel, die Dateninfrastruktur in der Forschung und Industrie zu verbessern, wurden im Jahr 2016 die FAIR-Prinzipien für Datenmanagement und Datenadministration in der Wissenschaft (Originaltitel: The FAIR Guiding Principles for Scientific Data Management and Stewardship) [6] vorgestellt. Das Akronym **FAIR** setzt sich hierbei aus den vier Kategorien **F**indable (auffindbar), **A**ccessible (zugänglich), **I**nteroperable (interoperabel) und **R**e-usable (wiederverwendbar) zusammen, welche nachfolgend näher erläutert werden.

Um Daten bzw. Metadaten gemäß den FAIR-Prinzipien auffindbar (F) zu gestalten, werden vier Bedingungen definiert [6]:

- **F1**: Daten müssen global einzigartig und dauerhaft identifizierbar sein, z.B. über eine eindeutige ID.
- **F2**: Daten müssen durch Metadaten hinreichend beschrieben bzw. annotiert werden.
- **F3**: Die Metadaten müssen die eindeutige ID der Daten enthalten.
- **F4**: Die Daten bzw. Metadaten müssen (maschinell) durchsuchbar sein.

Die Zugänglichkeit (A) der Daten wird durch nachfolgende zwei Bedingungen sichergestellt [6]:

- **A1**: Daten müssen über ein standardisiertes Protokoll anhand ihrer ID abrufbar sein. Das genutzte Protokoll muss hierbei offen, frei verfügbar und universell

einsetzbar sein (**A1-1**), gleichzeitig aber auch die Möglichkeit bieten, die Daten durch eine Authentifizierung bzw. Autorisierung zu schützen (**A1-2**).

- **A2**: Die Zugänglichkeit der Metadaten muss gewährleistet sein, auch wenn die zugehörigen Daten nicht länger verfügbar sind.

Der insbesondere für die Industrie relevante Punkt A1-2 impliziert, dass Daten sowohl zugänglich als auch sicher sein können, da sich die Zugänglichkeit auch auf ausgewählte Kreise innerhalb eines Unternehmens beziehen kann.

Interoperabilität (I) beschreibt im Kontext der Daten die Fähigkeit, diese im Idealfall so bereitzustellen, dass sie ohne zusätzliche Anpassungen von mehreren Systemen, Benutzern usw. genutzt werden können. Die FAIR-Prinzipien definieren dazu folgende drei Bedingungen [6]:

- **I1**: Die Wissensrepräsentation der Daten und Metadaten muss in einer formalen, weithin akzeptierten und breit zugänglichen Sprache erfolgen.
- **I2**: Die verwendeten Vokabularien müssen den FAIR-Prinzipien entsprechen.
- **I3**: Daten und Metadaten enthalten geeignete Referenzen auf andere Daten und Metadaten.

Zur Gewährleistung der Wiederverwendbarkeit (R) müssen Daten hinreichend mit zutreffenden Eigenschaften beschrieben werden (**R1**). Dazu müssen die Daten und Metadaten [6]:

- **R1.1**: Unter einer klaren und zugänglichen Nutzerlizenz veröffentlicht werden.
- **R1.2**: Eindeutig und detailliert mit ihrer Herkunft verbunden sein.
- **R1.3**: Gültigen Standards der jeweiligen Branche bzw. Community entsprechen.

In der Wissenschaft sind die FAIR-Prinzipien laut einer Studie aus dem Jahr 2021 bereits zwei von drei Wissenschaftlern bekannt und jeder vierte ist mit den Prinzipien vertraut [17]. Um Anwender bei der Nutzung bzw. Umsetzung der FAIR-Prinzipien zu unterstützen, veröffentlicht [18] im Rahmen des FAIRplus Projektes das FAIR Cookbook, welches mehr als 80 sog. Rezepte zu den FAIR-Prinzipien (Stand 2024) enthält. Eine quantitative Beurteilung der FAIRness von Daten wurde 2020 von der RDA FAIR Data Maturity Model Working Group mit dem sog. FAIR Data Maturity Model vorgestellt [19]. Das FAIR Data Maturity Model definiert für jedes der vier Prinzipien diverse Indikatoren mit den drei Relevanzgraden notwendig, wichtig

und nützlich. Jeder Indikator wird von der evaluierenden Person mit den Stufen 0 - Nicht zutreffend bis 4 - Vollständig umgesetzt bewertet und anschließend das Ergebnis in einem Netzdiagramm dargestellt. Eine Übersicht der Indikatoren inklusive Beschreibung findet sich in Anhang A.1.2. In einer Fallstudie zeigt [20] ein praxisnahes Beispiel zur Umsetzung der FAIR-Prinzipien und der Anwendung des FAIR Data Maturity Model anhand eines Prüfstandes für elektromechanische Zylinder.

2.3 Methoden für die Mess- und Datenplanung

2.3.1 Cross-Industry Standard Process for Data Mining

Data Mining (deutsch: Förderung von Wissen) bezeichnet den Prozess, Wissen aus Daten zu generieren und wird als Teil der Knowledge Discovery in Databases (KDD), der Wissensentdeckung in Datenbanken, gesehen [21, 22]. Allgemein kann der KDD-Prozess in die neun folgenden Schritte eingeteilt werden [22]: 1. Aufbau von domänenbezogenem Wissen, 2. Auswahl von Daten, 3. Bereinigung und 4. Vorverarbeitung der Daten, 5. Datenreduktion und 6. Projektion, 7. Modellauswahl, 8. Data Mining und 9. Interpretation der Ergebnisse.

Im industriellen Kontext werden die abgewandelten Methoden SEMMA (**S**ample, **E**xplore, **M**odify, **M**odel, **A**ssess), und Cross-Industry Standard Process for Data Mining (CRISP-DM) als die bekanntesten Methoden für die praktische Anwendung gesehen [23]. Beide Methoden führen den Anwender durch den KDD-Prozess, jedoch ist der CRISP-DM ganzheitlicher [23]. In der Studie *A survey of data mining and knowledge discovery process models and methodologies* [24] wurden gängige Methoden verglichen und der CRISP-DM als die gängigste Methode in der Industrie identifiziert.

Mit dem CRISP-DM [7] wurde im Jahr 1996 ein industrieübergreifender Standard für das Data Mining entwickelt. Im Referenzmodell wird ein Data Mining Projekt in sechs Phasen unterteilt:

- **Phase 1**: Geschäftsverständnis (englisch: Business understanding)
- **Phase 2**: Datenverständnis (englisch: Data understanding)
- **Phase 3**: Datenaufbereitung (englisch: Data preparation)
- **Phase 4**: Modellbildung (englisch: Modeling)
- **Phase 5**: Evaluation
- **Phase 6**: Anwendung (englisch: Deployment)

Abbildung 2.1 zeigt, dass die Phasen zwar voneinander abhängig sind, aber nicht zwangsweise sequentiell ablaufen.

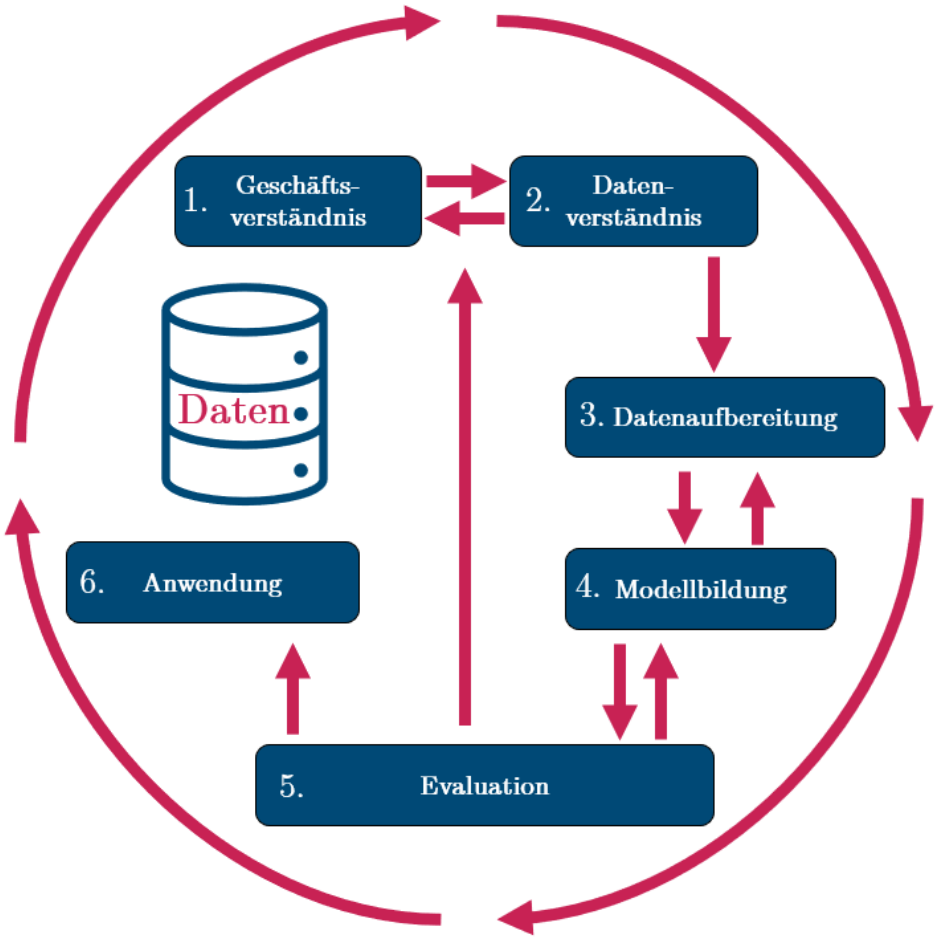


Abbildung 2.1: Die sechs Phasen des CRISP-DM Referenzmodells nach [7].

Das Ziel der ersten Phase ist der Aufbau eines grundlegenden **Geschäftsverständnisses** (1). Hier definiert der CRISP-DM die vier Aufgaben: 1.1 Bestimmung der Geschäftsziele, 1.2 Beurteilung der Situation, 1.3 Zielsetzung des Data Mining Projektes und 1.4 Erstellung eines Projektplans.

In der zweiten Phase **Datenverständnis** (2) werden zunächst erste Daten gesammelt (2.1), beschrieben (2.2) und analysiert (2.3). Weiterhin wird die Qualität der Daten überprüft (2.4). Die Phasen 1 und 2 können sich gegenseitig beeinflussen, da z.B. für die Beurteilung der aktuellen Situation (1.2) und der konkreten Zielsetzung (1.3) die erste Analyse relevanter Daten (2.3) und deren Qualität (2.4) relevant ist.

In der anschließenden Phase der **Datenaufbereitung** (3) werden Daten ausgewählt (3.1) und bereinigt (3.2). Diese Daten werden dann konstruiert (3.3), wobei dadurch neue Attribute aus bestehenden Daten abgeleitet, Werte transformiert oder gänzlich neue Datensätze erzeugt werden, um zusätzliche Informationen für die Modellierung bereitzustellen. Anschließend werden die Daten integriert (3.4) und formatiert (3.5).

Die **Modellbildung** (4) befasst sich mit der Auswahl von Modellierungstechniken (4.1), dem Entwurf eines Testdesigns (4.2) sowie der Erstellung (4.3) und Bewertung (4.4) des Modells. Anzumerken ist, dass sich die Unterphasen der Datenaufbereitung (3) und Modellbildung (4) gegenseitig beeinflussen können. So können bspw. gewisse Modellierungstechniken eine Anpassung in der Datenkonstruktion (3.3) und Datenformatierung (3.5) erfordern.

Nach der Erstellung eines Modells muss dieses in der Phase **Evaluation** (5) evaluiert werden. Der CRISP-DM gliedert diese Phase in die drei Schritte Auswertung der Ergebnisse (5.1), Überprüfung der Prozesse (5.2) und Festlegung der nächsten Schritte (5.3). Phase 5 kann hierbei einerseits zu den Phasen 1 und 2, als auch zu Phase 4 weitere Erkenntnisse liefern.

Um das evaluierte Modell in die **Anwendung** (6) zu transferieren, muss zunächst der Einsatz konkret geplant (6.1) und regelmäßige Prüfintervalle zur Überwachung und Wartung des Modelles festgelegt werden (6.2). Letztlich erfolgt die Erstellung eines Abschlussberichtes (6.3) und eine abschließende Bewertung des Data Mining Projektes (6.4). Der gesamte Prozess des CRISP-DM kann wiederholt werden, wobei Ergebnisse, Erfahrungen und Erkenntnisse des ersten Durchlaufes wiederverwertet werden können [7].

2.3.2 Statistische Checkliste

In ihrer Dissertation [25] stellt Mende die *Statistische Checkliste* zur Datenaufbereitung vor, welche innerhalb einer Merkmalentstehungs- und -wechselwirkungsanalyse bei der strukturierten Bearbeitung von Daten unterstützen soll. Die *Statistische Checkliste* orientiert sich dabei an den fünf Phasen des DMAIC-Zykluses (aus dem Englischen von **D**efine, **M**easure, **A**nalyse, **I**mprove und **C**ontrol): Definieren, Messen, Analysieren, Verbessern und Steuern [26]. Zu den Phasen Definieren, Messen und Analysieren formuliert Mende [25] 88 Checkpunkte aus den Erfahrungen des Projektes Messtechnisch gestützte Montage (MessMo) und bietet zusätzliche Notabene (deutsch: Wohlgemerkt) zur Unterstützung der Anwender. Abbildung 2.2 zeigt einen Auszug der Statistischen Checkliste in der Definieren-Phase [25].

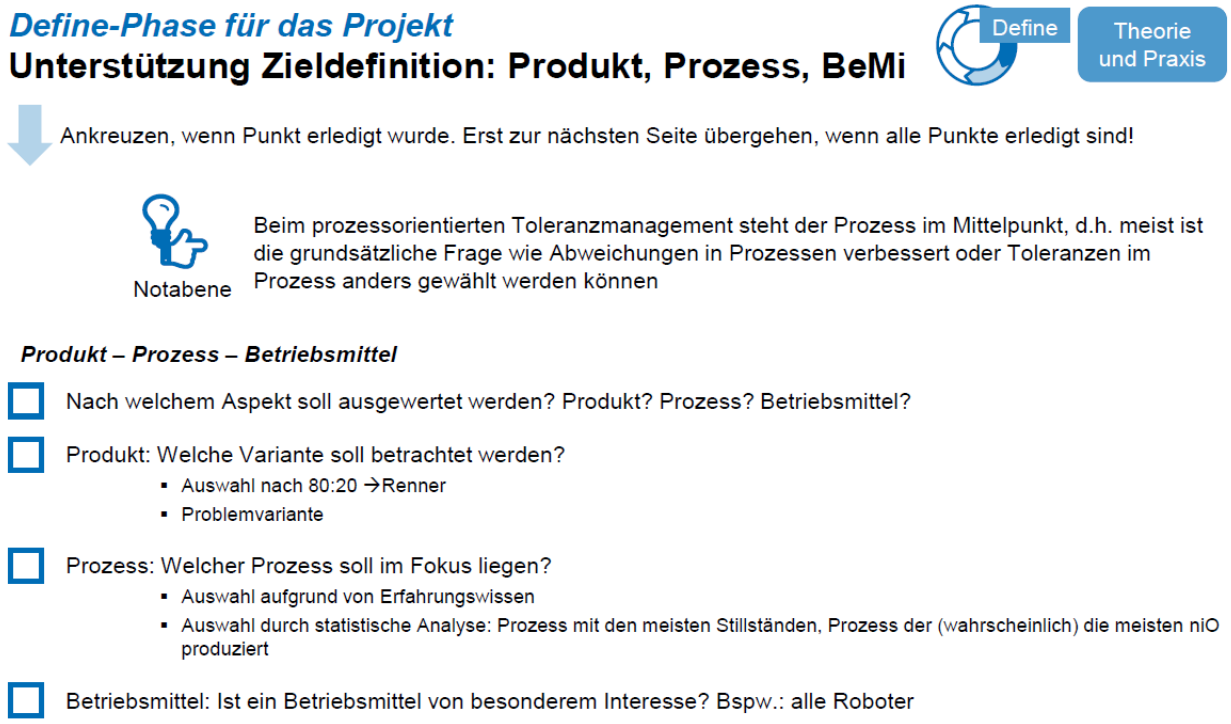


Abbildung 2.2: Auszug aus der Statistischen Checkliste [25].

Aufgrund der positiven Resonanz der Projektpartner im Projekt MessMo (vgl. [27]) wird die Struktur der statistischen Checkliste in Kapitel 3 aufgegriffen, maßgeblich erweitert und verallgemeinert.

2.3.3 Ursache-Wirkungs-Diagramm

Daten in einem industriellen Kontext sind häufig zahlreichen, gleichzeitig wirkenden und sich gegenseitig beeinflussenden Einfluss- und Störgrößen ausgesetzt (vgl. Tabelle 2.1). Zur Generierung von Wissen über kausale Zusammenhänge zwischen Eingangsgrößen (Einfluss- und Störgrößen) und Ausgangsgrößen existieren diverse Methoden, wie z.B. das Ursache-Wirkungs-Diagramm (UWD) [28], das Kausalschleifendiagramm [29] oder der gerichtete azyklische Graph [30]. Eine Übersicht über weitere Methoden und deren Vergleich untereinander zeigt [31]. Die wohl bekannteste Form des UWD wurde von Kaoru Ishikawa im Jahr 1943 entwickelt, weswegen das UWD auch unter dem Namen Ishikawa-Diagramm bekannt ist [32]. Abbildung 2.3 zeigt ein generisches UWD mit einem zu untersuchenden Qualitätsmerkmal (blau), welches von vier Hauptfaktoren (pink) beeinflusst wird.

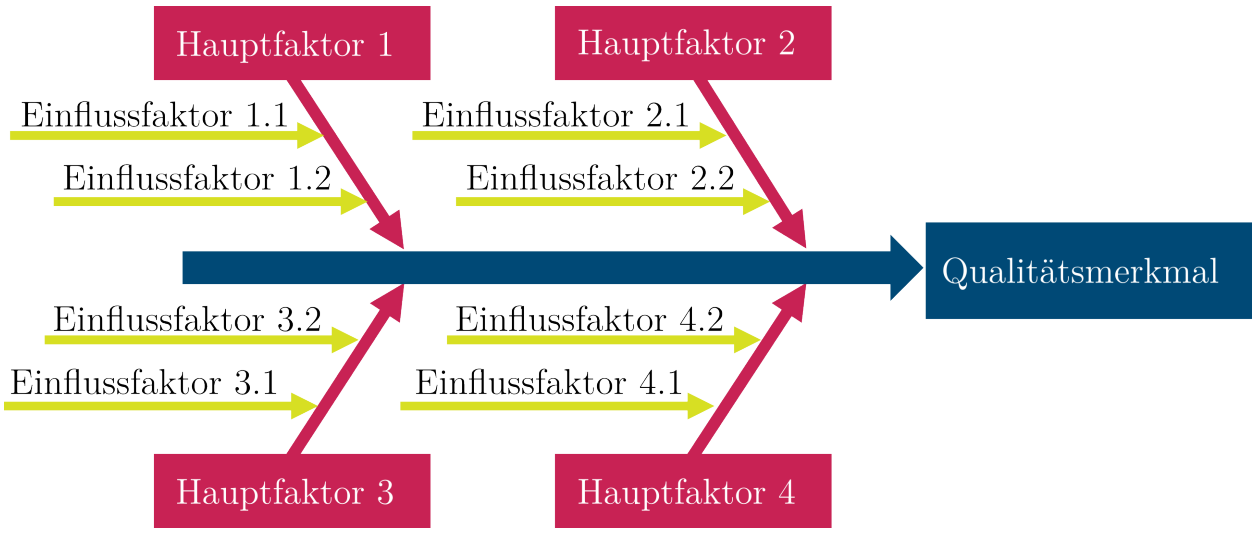


Abbildung 2.3: Prinzipieller Aufbau eines UWD nach [28].

Jeder Hauptfaktor enthält zudem jeweils zwei weitere Einflussfaktoren. In seiner Publikation *Guide to Quality Control* definiert Ishikawa fünf grundlegende Schritte zur Erstellung eines UWD [28]:

1. Auswahl des zu untersuchenden Qualitätsmerkmals (Abbildung 2.3, blau).
2. Zeichnen eines auf das Qualitätsmerkmal deutenden Pfeiles (Abbildung 2.3, blau).
3. Bestimmung der Hauptfaktoren, welche das Qualitätsmerkmal beeinflussen (Abbildung 2.3, pink).
4. Bestimmung von Einflussfaktoren auf die Hauptfaktoren (Abbildung 2.3, gelb). Jeder Einflussfaktor kann ebenfalls weitere Untereinflussfaktoren enthalten.
5. Überprüfung der Diagramms auf Sinnhaftigkeit und Vollständigkeit.

Eine verbreitete Methode für eine Auswahl der Kategorien der Hauptfaktoren ist die 5M-Methode [33], mit den fünf Hauptfaktoren **M**ensch, **M**aschine, **M**aterial, **M**ilieu, und **M**ethode. Trotz der Zugänglichkeit und Nutzerfreundlichkeit kann das UWD bei komplexen Systemen unübersichtlich werden. Weitere Nachteile sind die fehlende Analyse von Eintrittswahrscheinlichkeiten der Einflussfaktoren und die fehlende Darstellung der Beziehung bzw. Beeinflussung der Faktoren untereinander [31].

Um den Nachteil der fehlenden Eintrittswahrscheinlichkeiten auszugleichen, kann z.B. eine Pareto-Analyse (auch ABC-Analyse) mit dem UWD kombiniert werden [33]. Eine Auflistung solcher Abwandlungen des UWD wie bspw. dem Reverse Fishbone Diagram, in welchem die Auswirkungen von Änderungen untersucht werden, findet sich in [34].

2.3.4 Versuchsplanung

Speziell die Industrie strebt nach stetiger Verbesserung und Optimierung ihrer Produkte, Prozesse oder Anlagen. Die reine Analyse in der Retrospektive von ohnehin anfallenden Daten ist aufgrund komplexer Wirkungszusammenhänge schwierig bzw. häufig unzureichend [35]. Durch zusätzliche Versuchsreihen können hier neue Erkenntnisse solcher Wirkungszusammenhänge gewonnen und dadurch Produkte, Prozesse und (ML-)Modelle verbessert werden.

Die statistische Versuchsplanung, im Englischen Design of Experiments (DoE), befasst sich mit der strukturierten und effizienten Planung von Versuchsreihen zur Gewinnung statistischer Erkenntnisse [36]. In einem generischen System bzw. Prozess (Abbildung 2.4 a) wird aus einem Input ein Output erzeugt.

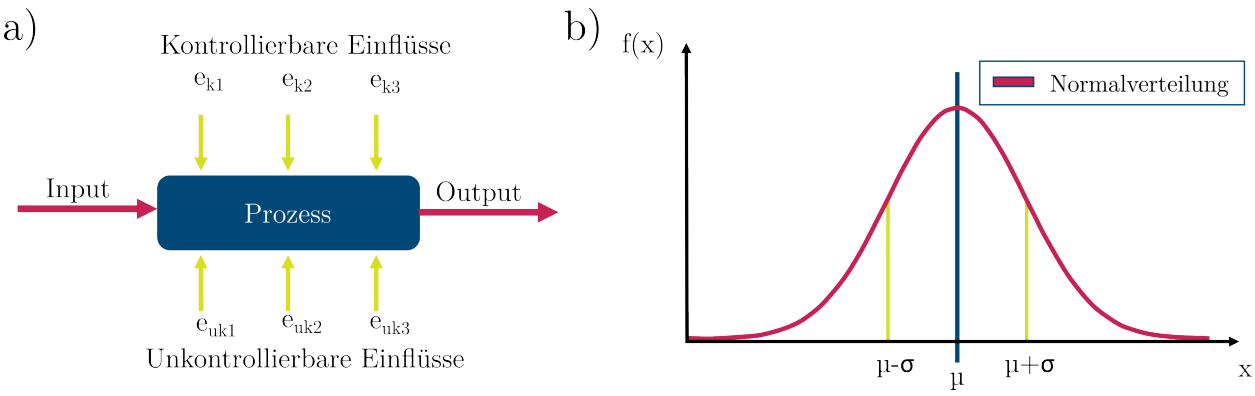


Abbildung 2.4: **a)** Modell eines Prozesses mit einem Input und Output (rot), sowie kontrollierbaren und unkontrollierbaren Einflussgrößen (gelb) nach [37]. **b)** Normalverteilung mit Erwartungswert μ (blau) und einfacher Standardabweichung 1σ (gelb) [38].

Dabei wird der Output von kontrollierbaren Einflüssen e_k, wie bspw. Prozessparametern, und unkontrollierbaren Einflüssen (Störgrößen) e_{uk}, wie z.B. Temperaturschwankungen, beeinflusst [37]. Diese Einflüsse bewirken eine Abweichung des realen Outputs zum erwarteten Output (Erwartungswert μ). Der Output kann z.B. das Maß der Länge eines bearbeiteten Bauteils sein. Durch die oben genannten Einflüsse wird das resultierende Maß streuen und nicht exakt dem Erwartungswert μ entsprechen. Ist die Verteilungsfunktion bekannt, kann abgeschätzt werden, mit welcher Wahrscheinlichkeit die Maße in einem gewissen Bereich der Verteilungsfunktion liegen. Abbildung 2.4 zeigt eine Normalverteilung mit Erwartungswert μ und der Standardabweichung σ. Die einfache Standardabweichung 1σ definiert, dass bei normalverteilten Daten 68,3 % der Daten innerhalb des Bereiches μ-σ und μ+σ liegen [35]. Eine Übersicht über weitere gängige Verteilungsfunktionen listet [39] auf. Die Normalverteilung ist in der Industrie eine der relevantesten Verteilungen

[32]. Grundlage hierfür bildet der zentrale Grenzwertsatz, welcher besagt, dass sich eine Verteilung mit zunehmender Anzahl an Einflussgrößen einer Normalverteilung annähert [40].

Im Kontext der Normalverteilung in der Industrie ist die Six Sigma Strategie [41] erwähnenswert, welche versucht, ein Qualitätsniveau (z.B. fehlerfrei produzierte Bauteile) von 6σ (99,999.66 %) zu erreichen. In der deutschen Industrie erreichen produzierende Unternehmen bei ihren Produkten durchschnittlich ein Qualitätsniveau von $3{,}8\sigma$ (99 %) [41].

Um die Einflussgrößen (e_k, e_{uk}) auf die Daten zu identifizieren, kann bpsw. das UWD (vgl. Abschnitt 2.3.3) verwendet werden. Die so identifizierten Einflussgrößen können anschließend, nach Möglichkeit kontrolliert im Rahmen eines DoE, während der Datenaufnahme variiert werden. Weiterhin kann das klassische DoE in die drei sequentiellen Phasen qualitative Systembeschreibung, Sichtung der Faktoren und Detailuntersuchung aufgeteilt werden [36]. Die qualitative Systembeschreibung befasst sich mit der Ermittlung der Systemgrenzen, der Bestimmung von Qualitätsmerkmalen, Einflussgrößen und Zielbereichen sowie der Ermittlung von deren Messbarkeit. In der zweiten Phase (Sichtung der Faktoren) findet neben der Faktorauswahl und Stufenfestlegung auch die Erstellung eines Screening-Plans, dessen Versuchsdurchführung und ein anschließendes Beschreibungsmodell statt. In der anschließenden Detailuntersuchung werden die Faktoren aus Phase 2 angepasst, ein Modell und ein Versuchsplan erstellt und damit ein weiterer Versuch durchgeführt, der ebenfalls ein Beschreibungsmodell erhält und anschließend optimiert wird.

2.3.5 Lessons Learned

Das Project Management Institute empfiehlt nach Abschluss eines Projektes die Formulierung von sog. Lessons Learned und definiert diese als im Projektverlauf gesammeltes Wissen über aufgetretene Umstände und deren Lösung zur Steigerung der Performanz in zukünftigen Projekten [42]. Dabei kann dieses Wissen bspw. von Projektergebnissen, Fachexperten, interdisziplinärer Zusammenarbeit oder der angewandten Forschung stammen [43]. Die Erstellung bzw. Formulierung von Lessons Learned erfolgt nach den fünf aufeinanderfolgenden Phasen [44]:

- **Phase 1**: **Identifizierung** von potentiellen Lessons Learned z.B. in einem dezidierten Meeting.
- **Phase 2**: Ausführliche **Dokumentation** der potentiellen Lessons Learned.

- **Phase 3**: **Analyse** des Potenzials der dokumentierten Lessons Learned.
- **Phase 4**: **Speicherung** der Lessons Learned in einem Repositorium.
- **Phase 5**: **Nutzung** der Lessons Learned in zukünftigen Projekten. Lessons Learned welche sich nicht bestätigen, werden aus dem Speicher entfernt.

Zu jeder der oben genannten Phasen bietet [44] eine detaillierte Beschreibung sowie Vorlagen zum Download. Das U.S. Department of Energy empfiehlt zudem in seinem Standard DOE-STD-7501-99, dass jede einzelne Lesson Learned die folgenden Kriterien erfüllen sollte [45]:

- Die Lesson Learned enthält eine klare Beschreibung sowie die Umstände, unter denen sie gewonnen wurde.
- Die Vorteile der Lesson Learned und das Potenzial für deren Anwendung in zukünftigen Projekten wird aufgezeigt.
- Die Lesson Learned enthält Kontaktdaten bzw. einen Ansprechpartner für potentielle Rückfragen oder zusätzliche Informationen.
- Der Lesson Learned wurden beschreibende Stichworte zur Erhöhung der Auffindbarkeit zugeordnet.

Jede neu generierte Lesson Learned wird zunächst als Lesson Learned Hypothese bezeichnet [43]. Um aus einer Lesson Learned Hypothese eine sog. High-Quality Lesson Learned abzuleiten, muss diese mehrfach in verschiedenen Projekten bzw. Anwendungsfällen gültig sein.

2.4 Datenqualität

Die Qualität aufgezeichneter Daten spielt in der späteren Datenanalyse und Modellbildung eine entscheidende Rolle. Durch qualitativ minderwertige Daten können keine robusten und zuverlässigen ML-Modelle erzeugt werden. Dementsprechend ist, neben der Quantität der Daten, eine hohe Qualität essenziell. Zudem kann eine niedrige Datenqualität gemäß der 1:10:100-Regel von [46] hohe Kosten verursachen. Gemäß dieser Regel stehen jedem 1 $ für die Prävention einer schlechten Datenqualität, 10 $ für die Korrektur schlechter Daten und 100 $ für die resultierenden Folgen einer schlechten Datenqualität gegenüber.

In der Industrie wird, abgesehen von diversen Abteilungen wie z.B. der Marketing-abteilung, der Großteil der Daten vorrangig durch den Einsatz von Sensoren und über

Steuerungs- bzw. Regelungseinheiten gewonnen. Daher wird nachfolgend kurz auf Sensoren und aus ihnen resultierende Fehlerquellen eingegangen. Ein Sensor ist dabei als ein Bauteil, welches eine physikalische oder chemische Größe quantitativ erfasst und in eine elektrische Ausgangsgröße umwandelt, definiert [47]. Bei der Erfassung von Messgrößen durch einen Sensor können diverse Fehler auftreten. Nachfolgend werden für Sensordaten typische Fehler aufgelistet [48]:

- **Ausreißer**: Messwerte, welche im Betrag verhältnismäßig stark von den übrigen Messwerten abweichen.
- **Drift**: Verschiebung der Verteilung der Daten über die Zeit, z.B. hervorgerufen durch Verschleiß.
- **Konstante Werte**: Messwerte, deren Betrag über mehrere Messungen hinweg unverändert bleiben, bspw. durch einen Sensorfehler oder Fehler bei der Datenübertragung.
- **Nullwerte**: Messwerte, die konstant den Betrag 0 annehmen und u.a. durch defekte Sensoren oder Fehler bei der Kommunikation hervorgerufen werden können.
- **Offset**: Eine konstante positive oder negative Abweichung der Messwerte, welche z.B. durch fehlende oder falsche Kalibrierung entstehen kann.
- **Rauschen**: Rauschen bezeichnet eine betragsmäßig wechselnde Abweichung des Messwertes zum realen Wert. Prinzipiell ist jeder Sensor durch Messunsicherheiten rauschbehaftet.
- **Unvollständige Daten**: Fehlende Messwerte, welche bpsw. durch den Ausfall eines Sensors oder der Kommunikation entstehen.

2.4.1 Messunsicherheit

Jede Messung einer Messgröße, z.B. durch einen Sensor, kann den tatsächlichen Wert der Messgröße nur bis zu einem bestimmten Grad bzw. einer bestimmten Genauigkeit abbilden und ist mit einer gewissen Messunsicherheit verbunden [49]. Ein Messwert kann daher nur unter Angabe einer zugehörigen Messunsicherheit als komplett angesehen werden [50].

Im Internationalen Wörterbuch der Metrologie wird die Messunsicherheit als nicht negativer Parameter beschrieben, welcher die Streuung quantitativer Werte um eine

Messgröße basierend auf gegebenen Informationen angibt [51]. Messunsicherheiten können durch zufällige Effekte wie z.B. Schwankungen der Temperatur (Umgebungseffekte) oder durch systematische Fehler wie Nullpunktabweichungen der Messgeräte entstehen und treten sowohl im Messwert als auch im zugehörigen Zeitstempel auf [49, 52]. In [52] werden die potentiellen Ursachen von Messunsicherheiten in die Haupteinflussgruppen Messstrategie, Messgerät, Messobjekt, Mensch und Umgebung eingeteilt.

Der Guide to the Expression of Uncertainty in Measurement (GUM), der auch die Grundlage der Norm ISO/IEC GUIDE 98-3:2008 bildet, ist ein anerkannter Leitfaden zur Angabe von Messunsicherheiten [49]. Dieser Leitfaden wird durch die Erweiterungen GUM-S1 [53] und GUM-S2 [54] ergänzt. GUM-S1 ermöglicht die Berechnung der Messunsicherheitsfortpflanzung mithilfe der Monte-Carlo-Methode, während GUM-S2 die Methoden auf eine beliebige Anzahl von Ausgangsgrößen erweitert. Das Dokument GUM-1:2023 [55] stellt eine Einführung in die Betrachtung von Messunsicherheiten dar und verweist auf die oben aufgeführten Teile der GUM-Serie.

Die Fortpflanzung von Messunsicherheiten im Kontext des maschinellen Lernens wird in [52] näher untersucht. Weiterhin erweitert [52] die in Abschnitt 2.6.3 vorgestellte Automatisierte ML-Toolbox für zyklische Daten (ML-Toolbox) um die Betrachtung der Messunsicherheit nach GUM, zu der Uncertainty-Aware Automated Machine Learning Toolbox (UA-ML-Toolbox).

2.4.2 Metadaten

Neben den eigentlichen Messdaten sind auch qualitativ hochwertige Metadaten essenziell. Metadaten beschreiben und enthalten idealerweise alle relevanten Informationen und Hintergründe zum entsprechenden Datensatz und bilden die Grundlage für dessen Interpretierbarkeit und Verwertbarkeit. Allgemein können Metadaten in drei Metadatentypen eingeteilt werden [56]:

- **Administrative Metadaten**: Administrative Metadaten enthalten rechtliche Informationen wie z.B. Nutzungsbedingungen oder Zugriffsrechte.
- **Deskriptive Metadaten**: Deskriptive Metadaten sind eine textbasierte Beschreibung der Daten selbst.
- **Strukturelle Metadaten**: Strukturelle Metadaten beschreiben die Struktur bzw. den Aufbau der Daten.

Zur Vereinheitlichung und Sicherstellung der Maschinenlesbarkeit von Metadaten wurden bereits zahlreiche Standards entworfen, z.B. durch das World Wide Web

Consortium (W3C) [57], die The Dublin Core™Metadata Initiative (DCMI) [58] oder die FAIRsharing Community [59].

Metadaten können auch unter Verwendung von Taxonomien, Ontologien und Semantiken beschrieben werden. Dadurch ergibt sich der Vorteil, dass Informationen z.B. über die Beziehung von Merkmalen untereinander vorhanden sind. In der Forschungslandschaft existieren bereits diverse spezifische Ontologien wie z.B. die Semantic Sensor Network Ontologie [60] oder die Dublin Core Ontologie [61], welche jeweils Teilgebiete einer bestimmten Domäne abdecken. So kann die Semantic Sensor Network Ontologie zur Beschreibung von Sensoren und deren Messgrößen verwendet werden, während die Dublin Core Ontologie eine effiziente Organisation und standardisierte Beschreibung von Ressourcen wie Dokumenten und Medien ermöglicht.

Eine allgemeingültige, domänenübergreifende Ontologie existiert derzeit nicht, weswegen einzelne, bereits bestehende Ansätze auch kombiniert werden können, um spezifische Problemstellungen zu lösen [62]. Im Projekt NFDI4Ing wurde die Ontologie Metadata4Ing (m4i) [63] mit dem Ziel entwickelt, einen Rahmen für die ganzheitliche semantische Beschreibung von (Forschungs-)Daten der Ingenieurwissenschaften zu schaffen. Dabei nutzt, erweitert und kombiniert die m4i Ontologie bereits bestehende Ontologien, statt neue Ontologien zu schaffen [63].

2.4.3 Bewertung der Datenqualität

Die Erfassung und Bewertung der Datenqualität ist eine wichtige Maßnahme der Datenaufnahme, da hochwertige Daten die Grundlage für robuste ML-Modelle bilden. In [64] wird die Datenqualität dabei als „Grad, in dem ein Satz inhärenter Merkmale eines Datenprodukts Anforderungen erfüllt“ definiert. Speziell im industriellen Kontext werden die Anforderungen individuell von Unternehmen definiert und folgen keinem festen Bewertungsschema bzw. sind subjektiv. In der Literatur wurden bereits diverse qualitative und quantitative Ansätze wie z.B. die 15 IQ-Dimensionen [65] oder die vier Datenqualitätsmetriken nach Heinrich [66] entwickelt, welche nachfolgend vorgestellt werden.

2.4.3.1 Die 15 IQ-Dimensionen

Im Jahr 1992 wurden im Rahmen einer Studie [65] die 15 Informationsqualität (IQ)-Dimensionen definiert und in die vier IQ-Kategorien systemunterstützt, inhärent, darstellungsbezogen und zweckabhängig eingeteilt. Abbildung 2.5 zeigt die 15 IQ-

Dimensionen ihren IQ-Kategorien zugeordnet. Der jeweilig zugehörige Untersuchungsgegenstand der vier IQ-Kategorien wird am äußeren Rand dargestellt [67].

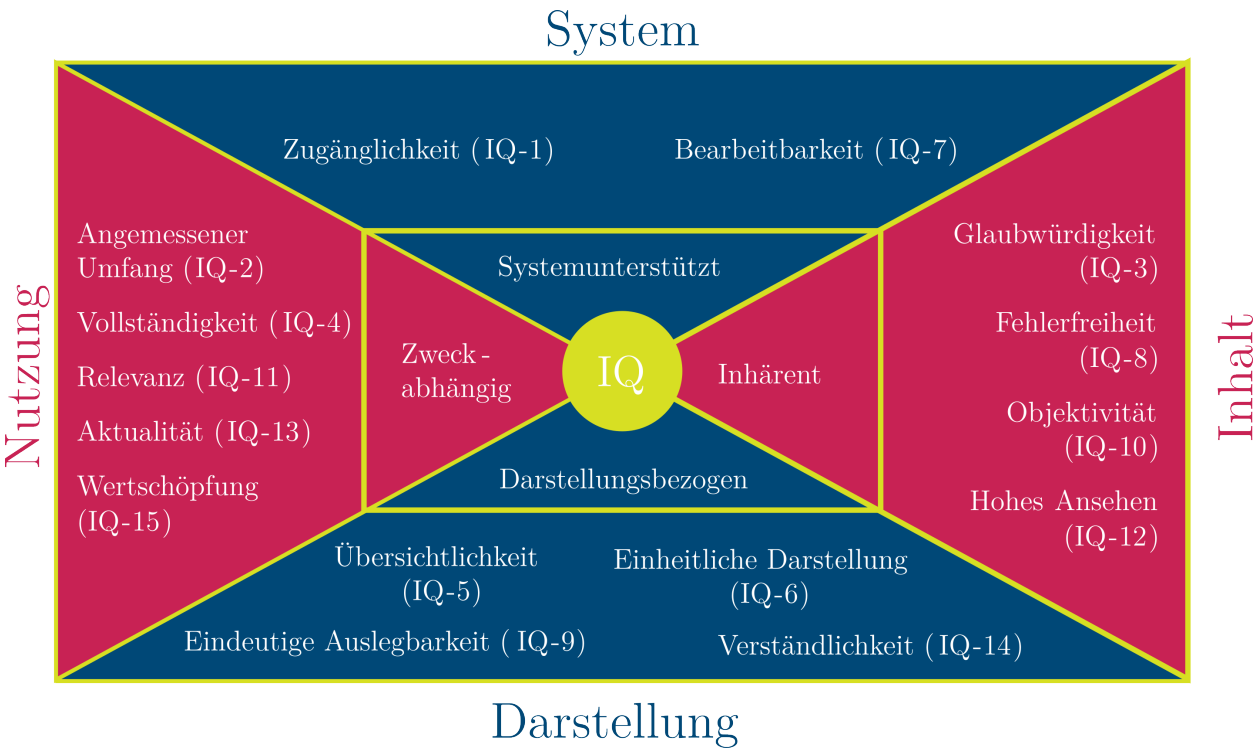


Abbildung 2.5: Darstellung der 15 IQ-Dimensionen und der vier IQ-Kategorien nach [67].

Nachfolgend werden die einzelnen IQ-Dimensionen aufgelistet und näher beschrieben [67]:

- **IQ-1 Zugänglichkeit**: Die Zugänglichkeit von Informationen ist durch einfache, direkte Verfahren sicherzustellen.
- **IQ-2 Angemessener Umfang**: Der Umfang der Informationen ist für die jeweiligen Anforderungen angemessen.
- **IQ-3 Glaubwürdigkeit**: Die Informationen wurden durch entsprechenden Aufwand bzw. unter hohen Qualitätsstandards gewonnen.
- **IQ-4 Vollständigkeit**: Informationen sind in den zu betrachtenden Zeitabschnitten vollständig.
- **IQ-5 Übersichtlichkeit**: Die Informationen sind in einer leicht verständlichen und übersichtlichen Form dargestellt.
- **IQ-6 Einheitliche Darstellung**: Die Informationen sind in ihrer Struktur und Beschaffenheit einheitlich abgebildet.

- **IQ-7 Bearbeitbarkeit**: Die Informationen sind für unterschiedliche Zwecke ohne hohen Aufwand bearbeitbar.
- **IQ-8 Fehlerfreiheit**: Die Realität wird durch die gesammelten Informationen abgebildet.
- **IQ-9 Eindeutige Auslegbarkeit**: Die Informationen sind so beschaffen, dass sie nicht auf unterschiedliche Arten interpretiert werden können.
- **IQ-10 Objektivität**: Die Informationen wurden objektiv betrachtet und sind somit sachlich und wertfrei.
- **IQ-11 Relevanz**: Es werden lediglich die Informationen gesammelt, welche für den Anwender bzw. den Anwendungszweck relevant sind.
- **IQ-12 Hohes Ansehen**: Die Vertrauenswürdigkeit der beteiligten Systeme und die Kompetenz der beteiligten Parteien während der Informationsgewinnung verleihen den Daten ein hohes Ansehen.
- **IQ-13 Aktualität**: Die Informationen bilden die aktuellen Eigenschaften des Anwendungsfalls ab.
- **IQ-14 Verständlichkeit**: Die Informationen können vom Anwender verstanden und eingesetzt werden.
- **IQ-15 Wertschöpfung**: Durch die Informationen kann eine Wertsteigerung erzielt werden.

Eine detaillierte Beschreibung der IQ-Dimensionen inklusive Beispiel findet sich in [67].

2.4.3.2 Datenqualitätsmetriken nach Heinrich

Zur quantitativen Bestimmung der Datenqualität entwickelte [66], basierend auf [64], Metriken für die vier Datenqualitäts-Dimensionen Vollständigkeit, Fehlerfreiheit, Konsistenz und Aktualität. Nachfolgend werden die aufgezählten Datenqualitäts-Dimensionen vorgestellt und näher erläutert, dabei wurden die Gleichungen 2.1-2.10 aus [66] übernommen bzw. adaptiert.

Um die Datenqualität in der Dimension **Vollständigkeit** zu bewerten, werden in [66] sukzessiv die drei Ebenen Merkmalsebene, Messebene und Datenbankebene betrachtet. Dabei wird unter der Vollständigkeit die Eigenschaft eines Merkmals

im Informationssystem verstanden, welche semantisch vom Wert NULL abweicht, wobei NULL weder ein erforderlicher, noch definierter Attributwert ist, sondern als Platzhalter für das Nicht-Befüllen eines Attributwertes steht. Daraus ergibt sich für die Vollständigkeit auf Merkmalsebene $Q_{Vollst.}(w)$, wobei w einen Merkmalswert darstellt:

$$Q_{Vollst.}(w) := \begin{cases} 0 & \text{falls } w = NULL \text{ oder } w \text{ zu } NULL \text{ semantisch äquivalent} \\ 1 & \text{sonst} \end{cases} \tag{2.1}$$

Auf der Ebene einer Messung M mit den Merkmalen $M.A$ ($M.A_1, M.A_2, ..., M.A_{|A|}$) und der Mächtigkeit $|A|$, ergibt sich die Vollständigkeit $Q_{Vollst.}(M)$ durch

$$Q_{Vollst.}\ (M) := \frac{\sum_{i=1}^{|A|} Q_{Vollst.}(M.A_i) g_i}{\sum_{i=1}^{|A|} g_i}. \tag{2.2}$$

Dabei beschreibt Gleichung (2.2) ein durch g_i ($g_i \in [0;1]$) gewichtetes arithmetisches Mittel der einzelnen Vollständigkeiten auf Merkmalsebene (Gleichung (2.1)). Das Gewicht g_i ist hierbei ein Mittel zur Gewichtung der Relevanz bzw. Wichtigkeit einzelner Merkmale. Auf Basis der sich aus Gleichung (2.2) ergebenden, einzelnen Vollständigkeiten einer Messung M_j über eine Anzahl Relationen R ($j = 1, 2, ..., |M|$) ergibt sich die Vollständigkeit $Q_{Vollst.}(R)$ durch

$$Q_{Vollst.}\ (R) := \frac{\sum_{j=1}^{|M|} Q_{Vollst.}(M_j)}{|M|}. \tag{2.3}$$

Die Vollständigkeit auf Datenbankebene $Q_{Vollst.}(D)$ setzt sich aus der Summe der Vollständigkeit der einzelnen Relationen R_k ($k = 1, 2, ..., |R|$) mit ihrer relativen Wichtigkeit g_k ($g_k \in [0;1]$) zusammen. Es gilt:

$$Q_{Vollst.}\ (D) := \frac{\sum_{k=1}^{|R|} Q_{Vollst.}(R_k) g_k}{\sum_{k=1}^{|R|} g_k}. \tag{2.4}$$

Die Qualität der **Fehlerfreiheit** $Q_{Fehl.}(w_I, w_R)$ auf Merkmalsebene, welche die Übereinstimmung der Merkmalswerte im Informationssystem w_I (Soll-Wert) mit den realen Werten w_R (Ist-Wert) angibt, kann durch

$$Q_{Fehl.}(w_I, w_R) := 1 - d(w_I, w_R) \tag{2.5}$$

bestimmt werden. Die Bewertung der Übereinstimmung von w_I mit w_R erfolgt durch das Distanzmaß $d(w_I, w_R)$. Je nach Beschaffenheit der Daten können verschiedene

Ansätze zur Bestimmung von $d(w_I, w_R)$ gewählt werden. Aus den von Heinrich vorgestellten Ansätzen wird nachfolgend ein Abstandsmaß speziell für numerische Merkmalswerte beschrieben. Es gilt

$$d(w_I, w_R) := \left(\frac{|w_I - w_R|}{max\{|w_I|, |w_R|\}} \right)^{\alpha}. \tag{2.6}$$

Durch den Parameter α ($\alpha \in \mathbb{R}^+$) kann die Toleranz gegenüber Abweichungen von w_I zu w_R eingestellt werden. Um die Fehlerfreiheit $Q_{Fehl.}(M)$ auf Messebene zu berechnen, gilt

$$Q_{Fehl.}(M) := \frac{1}{n} \sum_{i=1}^{n} Q_{Fehl.}(w_{I,i}, w_{R,i}). \tag{2.7}$$

Die Berechnung der Fehlerfreiheit auf Datenbankebene $Q_{Fehl.}(D)$ erfolgt analog zu Gleichung (2.7), durch Aggregation mittels Mittelwertbildung der Fehlerfreiheit auf Messebene.

Zur Bestimmung der Datenqualitäts-Dimension **Konsistenz** $Q_{Kons.}$, bzw. der Widerspruchsfreiheit, werden zunächst Regeln r_s innerhalb einer Regelmenge $\Re$ definiert ($r_i \in \Re$ mit $i = 1, 2, ..., |\Re|$). Auf der Merkmalsebene gilt für einen Merkmalswert w:

$$r_i(w) := \begin{cases} 0 & \text{falls } w \text{ der Konsistenzregel } r_i \text{ entspricht} \\ 1 & \text{sonst} \end{cases} \tag{2.8}$$

Daraus lässt sich die Qualität durch

$$Q_{Kons.}(w, \Re) := \prod_{i=1}^{|\Re|} (1 - r_i(w)) \tag{2.9}$$

bestimmen. Analog erfolgt die Bestimmung auf Mess- und Datenbankebene durch Aggregation mittels Mittelwertbildung der Konsistenz auf Merkmals- bzw. Messebene.

Die Metrik zur Bestimmung der Datenqualitäts-Dimension **Aktualität** stützt sich auf die Annahme einer zugrundeliegenden Wahrscheinlichkeitsverteilung, in diesem Falle einer Exponentialverteilung, mit der die Aktualität eines Datenbestandes beschrieben werden kann. Die Wahrscheinlichkeit $Q_{Akt.}(w, A)$, mit der ein Merkmal A und dem entsprechenden Merkmalswert w noch aktuell ist, wird durch

$$Q_{Akt.}(w, A) := e^{-\text{Verfall}(A) \cdot \text{Alter}(w, A)} \tag{2.10}$$

beschrieben, wobei Verfall(A) die angenommene Verfallsrate und Alter(w, A) das Alter des Merkmalswertes w seit Aufzeichnungszeitpunkt angeben.

2.5 Datenaufbereitung

Vor der eigentlichen Datenanalyse bzw. der Modellbildung ist in der Regel eine Aufbereitung der Daten erforderlich, welche grundlegend auf folgenden Schritten basiert:

1. Zusammenführen der Daten
2. Bereinigen der Daten
3. Anreichern der Daten
4. Normalisieren der Daten

Abhängig von der Beschaffenheit der Daten sind nicht alle der genannten Schritte zwingend erforderlich. Je nach herangezogener Literatur kann auch die Zuordnung, Reihenfolge und der Umfang der oben genannten Schritte variieren. So werden die in Abschnitt 2.6.2.2 vorgestellte Merkmalsextraktion und die in Abschnitt 2.6.2.3 vorgestellte Merkmalsselektion in mancher Literatur ebenfalls der Datenaufbereitung zugeschrieben [68, 69].

Unter dem **Zusammenführen der Daten** wird das Vereinigen von Daten aus mehreren Datenquellen zu einer einheitlichen Datenbasis verstanden [68]. Neben einer einfacheren Handhabbarkeit der Daten können durch die sog. Informationsfusion von bspw. mehreren Sensoren eines Sensornetzwerks auch neue und präzisere Informationen und Erkenntnisse über physikalische Wirkzusammenhänge gewonnen werden [70]. Speziell in der industriellen Anwendung fallen oftmals viele heterogene Daten an, wie z.B. an einer Produktionslinie mit mehreren Stationen, Prozessen und Betriebsmitteln (vgl. Herausforderung D1 nach Wilhelm in Tabelle 2.1). Dieser Umstand kann den notwendigen Aufwand der Datenintegration signifikant erhöhen [11].

Nach der Zusammenführung der Daten erfolgt das **Bereinigen der Daten** von u.a. Ausreißern, falschen, fehlerhaften oder stark verrauschten Messungen [68]. Auch die Korrektur von Drift oder Rekalibrierungen kann dem Schritt Datenbereinigung zugeordnet werden [25]. Die in Abschnitt 2.6.1 vorgestellten Methoden zur Visualisierung der Daten können ebenfalls Rückschlüsse auf Fehler in den Daten liefern.

Durch das **Anreichern der Daten** wird die Datenbasis ergänzt bzw. vervollständigt. Dies bezieht sich einerseits auf fehlende bzw. unvollständige Metadaten, andererseits auch auf die eigentlichen Messdaten. Hier können bspw. fehlende oder fehlerhafte Daten interpoliert werden.

Abschließend kann eine **Normalisierung der Daten** erfolgen, z.B. durch die Min-Max-Normalisierung, die Z-Score-Normalisierung oder die Dezimal-Skalierung, um Verzerrungen durch unterschiedliche Skalenumfänge zu vermeiden und die Daten aus mehreren Datenquellen vergleichen zu können [68].

2.6 Datenauswertung und Modellbildung

2.6.1 Methoden Visualisierung

Die Visualisierung von Daten ist eine wichtige Methode der Datenanalyse und besonders nützlich für die Bereinigung und Interpretation von Daten [71]. Die nachfolgenden Methoden wurden anhand der gesammelten Erfahrungen aus Forschungs- und Industrieprojekten am Lehrstuhl für Messtechnik zusammengestellt. Eine Übersicht über weitere gängige Visualisierungsmethoden findet sich in [72].

2.6.1.1 Histogramme

In Histogrammen werden kontinuierliche Daten X in Klassen der Anzahl m (Histogramm-Intervalle) eingeteilt und graphisch dargestellt, wodurch die Verteilung der Daten visualisiert werden kann [72, 73]. Dabei gilt für äquidistante Histogramm-Intervalle m mit den Klassengrenzen ξ [73]:

$$[\xi_1, \xi_2), [\xi_2, \xi_3), ..., [\xi_{m-1}, \xi_m), [\xi_m, \xi_{m+1}], \tag{2.11}$$

wobei

$$\xi_1 = min(X), \xi_{m+1} = max(X). \tag{2.12}$$

Für die jeweiligen Häufigkeitswerte $h_k(X)$ der Werteanzahl pro Intervall gilt

$$h_k(X) = \mid \xi \in X \mid \xi_k \leq \xi < \xi_{k+1} \mid, \quad k = 1, ..., m. \tag{2.13}$$

Die Intervallbreite Δx ergibt sich aus

$$\Delta x = \frac{max(X) - min(X)}{m}. \tag{2.14}$$

Die Anzahl m der Histogramm-Intervalle bzw. der Intervallbreite kann beliebig gewählt werden. Diese kann jedoch auch anhand der Anzahl von Werten n abgeschätzt werden, z.B. durch die Sturges-Regel [74]

$$\Delta x = 1 + \log_2(n), \tag{2.15}$$

oder die Scott-Regel [75]

$$\Delta x = \frac{3{,}5 \cdot \sigma}{\sqrt[3]{n}}. \tag{2.16}$$

Eine schematische Darstellung eines Histogramms einer Normalverteilung ist in Abbildung 2.6 a) dargestellt. Durch die Anwendung von Histogrammen können speziell die in industriellen Daten häufig vorkommenden ungleichmäßigen Datenverteilungen (Herausforderung P3 nach Wilhelm in Tabelle 2.1) wie z.B. bi- bzw. multimodale Verteilungen (Abbildung 2.6 b) schnell erkannt werden.

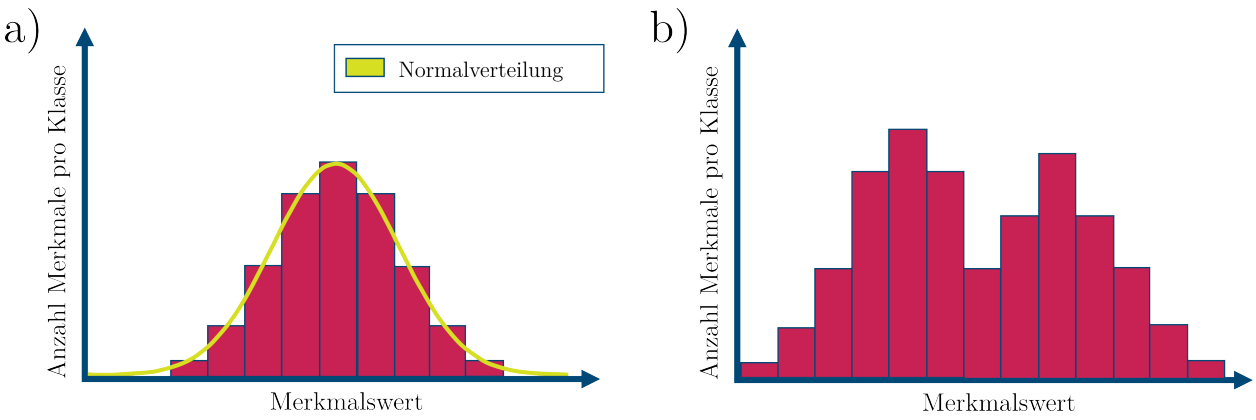


Abbildung 2.6: **a)** Histogramm einer Normalverteilung nach [72]. **b)** Histogramm mit bimodaler Verteilung.

Weiterführende Informationen zur Berechnung von Histogrammen wie z.B. Histogramme mit ungleicher Intervallbreite finden sich in [73].

2.6.1.2 Boxplot-Diagramm

Das Boxplot-Diagramm besteht aus einem oder mehreren sog. Boxplots (deutsch: Kastengrafik), welche jeweils Informationen über die Streuung und Verteilung von Daten liefern [72]. Abbildung 2.7 zeigt beispielhaft zwei Boxplot-Diagramme.

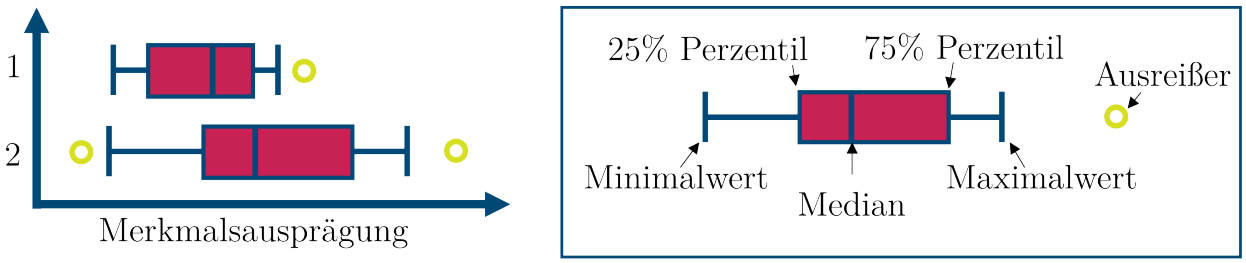


Abbildung 2.7: Darstellung zweier Boxplot-Diagramme nach [72].

Neben dem Minimal- und Maximalwert sowie dem 25 % und 75 % Perzentil wird auch der Median der zugrundeliegenden Daten dargestellt. Dabei werden der Minimal- und Maximalwert üblicherweise durch die äußersten Datenpunkte definiert, die innerhalb des 1,5-fachen Interquartilsabstands (entspricht ungefähr $\pm 2{,}7\,\sigma$ bzw. 99,3 % der normalverteilten Daten) um die Quartile liegen, während Werte außerhalb dieses Bereichs als Ausreißer dargestellt werden können (Abbildung 2.7, gelbe Kreise) [76].

Boxplot-Diagramme können Aufschluss über u.a. asymmetrische und unregelmäßig geformte Datenverteilungen geben und eignen sich auch zum Vergleich von Datensätzen untereinander (Abbildung 2.7 a) [72].

2.6.1.3 Hauptkomponentenanalyse

Die Hauptkomponentenanalyse (englisch: Principal Component Analysis (PCA)) ist ein Algorithmus zur Visualisierung von hochdimensionalen Daten und eignet sich potentiell zur Reduktion der Dimensionalität eines Datensatzes bei bestmöglichem Erhalt der Varianz der Daten [77]. Als strukturentdeckende Methode können durch die PCA zugrundeliegende Muster und Strukturen in den Daten identifiziert werden, ohne auf annotierte Daten angewiesen zu sein (vgl. unüberwachtes Lernen in Abschnitt 2.6.2) [78]. Der PCA liegt eine Lineartransformation zugrunde, welche aus n orthogonalen, nach erklärter Varianz absteigend sortierten Hauptkomponenten (PCs) besteht. Die resultierende Anzahl der PCs wird durch das Minimum aus Anzahl der Variablen $n_{Variablen}$ oder Messungen $n_{Messungen}$ festgelegt. Die Dimensionsreduktion erfolgt anschließend durch das Entfernen von PCs mit geringer erklärter Varianz aus der Menge aller PCs. Abbildung 2.8 a) zeigt eine Messreihe bei der jeweils zwei Merkmalswerte pro Messung erfasst wurden.

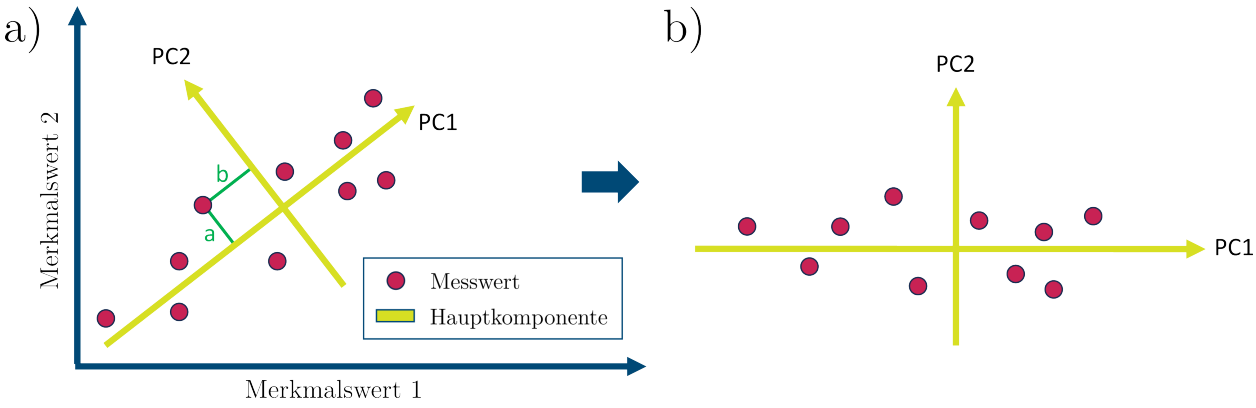


Abbildung 2.8: **a)** Im Merkmalsraum eingezeichnete PCs und **b**) transformierte Darstellung.

Zur Bestimmung der ersten PC wird der Datensatz um seinen Mittelwert zentriert und anschließend der quadrierte Abstand a (Abbildung 2.8a, grün) der Messwerte zur Ursprungsgeraden minimiert. Die Bestimmung der weiteren PCs erfolgt analog, wobei die PCs orthogonal zueinander stehen. Eine detaillierte mathematische Beschreibung der PCA findet sich in [79].

Die durch die PCA resultierenden PCs können auch als extrahierte Merkmale (vgl. Merkmalsextraktion in Abschnitt 2.6.2.2) verwendet werden. Dabei extrahiert die PCA Informationen über die allgemeine Zyklusform zeit-kontinuierlicher Daten mit geringem Approximationsfehler und Rauschunterdrückung [8].

2.6.1.4 Lineare Diskriminanzanalyse

Die Lineare Diskriminanzanalyse (LDA) ist ein Klassifikator, welcher die Separierbarkeit zwischen zwei Klassen maximiert und gleichzeitig die Streuung innerhalb der Klassen minimiert [80]. Weiterhin kann die LDA auch für Mehrklassenprobleme angewendet werden, indem sie den Merkmalsraum auf einen $(c-1)$-dimensionalen Unterraum reduziert, wobei c die Anzahl der Klassen ist [81]. Durch diese Dimensionsreduktion eignet sich die LDA zur Visualisierung von hochdimensionalen Daten in niedrigeren Dimensionen. Anhand der annotierten Daten können (vgl. überwachtes Lernen in Abschnitt 2.6.2) so auch zugrunde liegende Muster und Strukturen erkannt und überprüft werden [82].

Zur Durchführung der LDA gilt es die Kriteriumsfunktion $J(W)$

$$J(W) = \frac{W^\intercal S_B W}{W^\intercal S_W W} \tag{2.17}$$

zu maximieren, wobei S_W die Streu-Matrix innerhalb der Klassen, S_B die Streu-Matrix zwischen den Klassen und W den Projektionsvektor beschreibt [83, 84]. Eine detaillierte mathematische Beschreibung der LDA findet sich in [81].

2.6.1.5 Quasi-statisches Signal

Beim quasi-statischen Signal handelt es sich um die graphische Darstellung eines ausgewählten Messwertes in einem Messzyklus über mehrere Messzyklen hinweg [85]. Abbildung 2.9a zeigt zwei ausgewählte Messzyklen (Messzyklus 50, rot; Messzyklus 100, gelb) mit je sechs Messwerten.

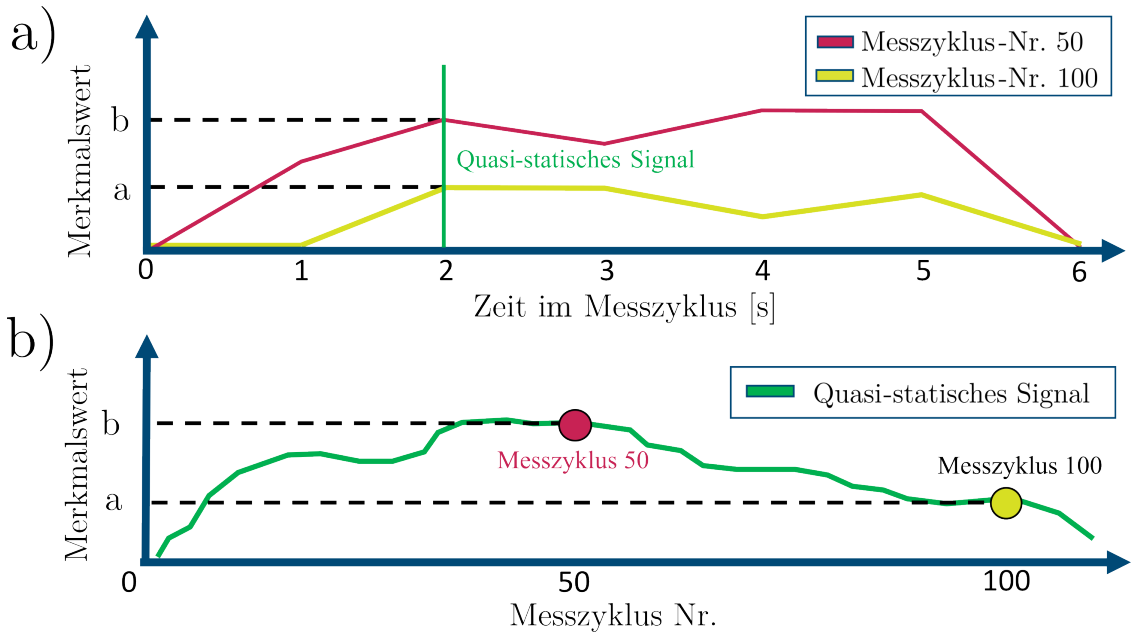


Abbildung 2.9: **a)** Signal von zwei Messzyklen mit je sechs Messwerten.
b) Quasi-statisches Signal des Messwertes 2.

Zur Erzeugung des quasi-statischen Signals für den Messwert 2 werden die jeweiligen Messwerte an Position 2 aus jedem Messzyklus aneinandergereiht. Dabei wird der Messwert aus Messzyklus 50 mit dem Merkmalswert a und der Messwert aus Messzyklus 100 mit dem Merkmalswert b verknüpft. Abbildung 2.9 zeigt das so resultierende quasi-statische Signal.

2.6.2 Künstliche Intelligenz und Methoden des maschinellen Lernens

Das Forschungsgebiet der künstlichen Intelligenz (KI) ist aus dem heutigen Alltag nicht mehr wegzudenken und versucht, Algorithmen Eigenschaften der menschlichen Intelligenz wie z.B. logisches Denken, das Treffen von Entscheidungen oder die

Kommunikation in natürlicher Sprache, anzutrainieren bzw. imitieren zu lassen [86]. Um dies zu erreichen, müssen im ersten Schritt sog. Modelle erstellt und anhand von Daten trainiert werden. Modelle bestehen aus einzelnen oder einer Kombination von Algorithmen. Das Training bezeichnet hierbei den Prozess, anhand von Daten den Algorithmus so anzulernen, dass dieser Muster und Zusammenhänge in den Daten erkennen kann. Um zu gewährleisten, dass die Modelle für neue Daten gültig sind, müssen diese validiert werden. Auf die Validierung wird dezidiert in Abschnitt 2.6.2.5 eingegangen. Die resultierende Leistungsfähigkeit der Modelle hängt stark von der Qualität der zugrundeliegenden Daten ab. Eine zu geringe Datenqualität kann zu unzuverlässigen Modellen führen, weswegen die Mess- und Datenplanung (MuD) zur Erzeugung hochqualitativer Daten eine entscheidende Rolle spielt. Daher sehen industrielle Unternehmen laut der IDG-Studie [3] eine mangelnde Datenqualität als eine der größten Herausforderungen für die Anwendung von KI. Als Ursachen der niedrigen Datenqualität werden in der Studie u.a. Herausforderungen bei der Datenaufbereitung, sowie eine fehlende Zugänglichkeit dieser bemängelt.

Allgemein wird zwischen zwei Stufen der KI unterschieden: Der schwachen künstlichen Intelligenz, welche sich auf einzelne Anwendungsfelder beschränkt, und der starken künstlichen Intelligenz, welche Anwendungsfelder-übergreifend (universell) agieren kann [87]. In der Praxis finden heutzutage ausschließlich schwache KI Anwendung. Typische Beispiele sind z.B. Chatbots, Sprachassistenten oder die Spam-Erkennung von E-Mails [87].

Im industriellen Kontext ist insbesondere das maschinelle Lernen (ML) von Relevanz. Dieses Teilgebiet der KI befasst sich mit der Idee, Maschinen anhand von Daten lernen zu lassen, um so neu aufgezeichnete Daten interpretieren zu können, und daraus Schlussfolgerungen für Handlungen zu ziehen oder Handlungsempfehlungen abzuleiten. Gängige Anwendungen sind hier z.B. die Zustandsüberwachung (engl. Condition Monitoring), die vorausschauende Wartung (engl. Predictive Maintenance) und die Qualitätsvorhersage von Produkten [87]:

- Die **Zustandsüberwachung** befasst sich mit der Überwachung des Zustands von bspw. Maschinen und Anlagen und kann auf Fehler oder Defekte hinweisen.
- Die **vorausschauende Wartung** verfolgt das Ziel, den Wartungszeitpunkt bpsw. einer Anlage auf Basis aufgezeichneter Daten zu optimieren, sodass die Lebensdauer von Verschleißteilen bestmöglich ausgeschöpft wird, ohne Stillstände zu riskieren.

- Die **Qualitätsvorhersage von Produkten** versucht die Qualität von Produkten schon vor einer dezidierten Prüfung vorherzusagen.

Allgemein kann das ML zunächst in die drei Arten unüberwachtes Lernen, überwachtes Lernen und bestärkendes Lernen unterteilt werden [87]:

- **Unüberwachtes Lernen**: Der Algorithmus kennt bzw. nutzt keine Zielgrößen, die den Daten zugeordnet sind. Daher eignen sich unüberwachte Lernverfahren, um versteckte Muster bzw. Cluster in den Daten zu identifizieren. Beispiele für unüberwachte Algorithmen sind u.a. die Hauptkomponentenanalyse (Abschnitt 2.6.1.3) oder Clustering-Verfahren wie z.B. das k-means Clustering, welches Datenpunkte in die vorgegebene Cluster-Anzahl k einteilt [88].
- **Überwachtes Lernen**: Der Algorithmus benötigt annotierte Daten und versucht, anhand der Eingangsdaten auf die entsprechenden Zielgrößen zu schließen. Beispiele für überwachte Algorithmen sind u.a. Lineare Diskriminanzanalyse (Abschnitt 2.6.1.4) oder Neuronale Netze [89].
- **Bestärkendes Lernen**: Der Algorithmus folgt vorgegebenen Regeln und wird durch Feedback in Form von Belohnungen bei richtigem Verhalten und Bestrafungen beim falschem Verhalten trainiert. Das bestärkende Lernen kann jedoch aufgrund der vergleichsweise hohen Komplexität und des hohen benötigten Ressourcenaufwandes unerfahrene Nutzer überfordern und wird daher im Rahmen dieser Dissertation nicht behandelt [87].

Da die Kategorien Unüberwachtes Lernen und Überwachtes Lernen nicht trennscharf sind, findet sich in der Literatur auch die Zwischenkategorie Semi-überwachtes Lernen, bei der ein Teil des Lernprozesses mit annotierten Daten durchgeführt wird, welcher anschließend mit den unannotierten Daten fortgesetzt wird [87]. Eine Übersicht der genannten Kategorien findet sich in Abbildung 2.10.

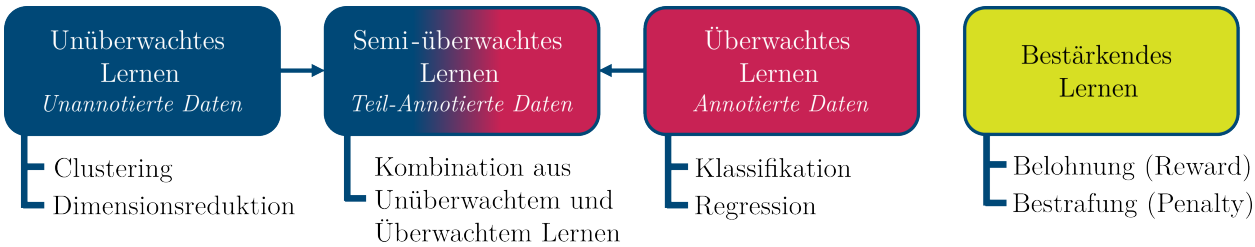


Abbildung 2.10: Übersicht der drei Hauptkategorien des maschinellen Lernens Unüberwachtes Lernen, Überwachtes Lernen und Bestärkendes Lernen, sowie der Zwischenkategorie Semi-überwachtes Lernen. Abbildung adaptiert aus [52] und ergänzt mittels [72, 87].

Eine Sonderstellung im Bereich KI und ML haben die sog. künstlichen neuronalen Netze (NN), welche insbesondere bekannt durch Chatbots wie ChatGPT [90] oder die Erkennung bzw. Interpretation von Bildern [91] wurden. Bei NN handelt es sich um Modelle, welche versuchen, die im Gehirn befindlichen neuronalen Netze künstlich abzubilden, um so komplexe Probleme bzw. Aufgabenstellungen lösen zu können. Ein einfaches NN besteht hierbei aus drei Schichten [89]:

- **Eingabe-Schicht**: Die Eingabe-Schicht nimmt die Eingangswerte, wie z.B. Daten aus Messungen, in das Modell auf.
- **Versteckte Schicht**: In der versteckten Schicht werden die Daten durch mathematische Operationen verarbeitet.
- **Ausgabe-Schicht**: Die Ausgabe-Schicht bringt die in der versteckten Schicht verarbeiteten Daten in eine für den Anwender verständliche Form zurück.

Durch die stetig steigende Rechenleistung können immer komplexere und leistungsfähigere NN mit einer Vielzahl an versteckten Schichten, berechnet werden, dem sog. Deep Learning [89]. Um jedoch zuverlässige Modelle mit NN zu erzeugen, bedarf es einer großen Menge hochqualitativer Daten. In industriellen Anwendungen ist i.d.R. zwar eine große Menge an Daten vorhanden, jedoch sind diese tendenziell einseitig verteilt (vgl. Herausforderung P3 nach Wilhelm in Tabelle 2.1), wodurch die Nutzung von NN erschwert wird. Weiterhin sind NN häufig eine Black-Box, da die Entscheidungsfindung der Modelle nicht oder bestenfalls schwer nachvollziehbar ist [89]. Aufgrund der hohen erforderlichen Rechenleistung und der schweren Interpretierbarkeit für unerfahrene Anwender werden in dieser Arbeit NN nicht weiter ausgeführt.

Einen alternativen Ansatz zu NN bietet die Bildung eines Modells mit einer Merkmalsextraktion und Merkmalsselektion und einer anschließenden Klassifikation bzw. Regression. Hierbei werden zunächst Merkmale aus den Daten extrahiert und anschließend relevante Merkmale ausgewählt, welche zur Lösung des vorgegebenen Problems beitragen. Je nach gewählten Algorithmen kann so ein Modell erstellt werden, welches eine ähnliche Performanz wie NN bietet, jedoch eine höhere Interpretierbarkeit aufweist und weniger Rechenleistung benötigt [92].

2.6.2.1 Klassifikation und Regression

Bei der Bildung von ML-Modellen erfolgt im ersten Schritt die Definition eines Lernproblems. Hier wird zwischen einem Klassifikationsproblem und einem Regressionsproblem unterschieden. Je nach Anwendungsszenario kann zusätzlich eine Merkmalsextraktion

und Merkmalsselektion durchgeführt werden, um die Modellkomplexität zu reduzieren [93]. Bei Klassifikationsproblemen versucht das ML-Modell neuen Daten kategorische Größen bzw. Klassen zuzuweisen. Ein typisches Klassifikationsproblem ist z.B. die Zustandsbewertung (unbeschädigt/beschädigt). Gängige Algorithmen zur Lösung eines Klassifikationsproblems sind u.a. die k-nächste-Nachbarn-Klassifikation und Support Vector Machines [94]. Bei Regressionsproblemen hingegen existieren keine diskreten Gruppen bzw. Klassen, sondern die Wertzuweisungen neuer Daten erfolgen kontinuierlich, i.d.R. durch numerische Werte, wie z.B. bei einer Temperaturskala [87]. Typische Beispiele für Algorithmen, die zur Lösung eines Regressionsproblems verwendet werden, sind die lineare Regression und die logistische Regression [87]. Eine Überführung von Regressionsproblemen in Klassifikationsprobleme ist ebenfalls möglich. So wird in [8] bspw. der Verschleiß (1 % - 100 % verbleibende Lebensdauer) von Komponenten eines hydraulischen Prüfstandes als Klassifikationsproblem (100 Klassen zu je 1 % Lebensdauer) definiert, um zu prüfen, ob das verwendete ML-Modell Daten, welche nicht im Training enthalten waren, in eine korrekte Reihenfolge bringen kann.

Die in Abschnitt 2.6.3 vorgestellte automatisierte ML-Toolbox für zyklische Daten (ML-Toolbox) behandelt primär Klassifikationsprobleme und nutzt eine Kombination aus der LDA mit dem Mahalanobis-Distanzmaß. Für die Mahalanobis-Distanz [95] $s_{Mahalanobis}$ neuer Daten (wie z.B. Messungen) d zu einer im Diskriminanzraum bestehenden Klasse mit dem Mittelwert μ gilt

$$s_{Mahalanobis}(d) = \sqrt{(d-\mu)^\top S^{-1}(d-\mu)}, \tag{2.18}$$

mit der Kovarianzmatrix S einer Gruppe. Neue Daten werden anhand der geringsten Distanz zu einer Klassenmitte klassifiziert [95].

Eine Sonderform der Lernprobleme stellt die Anomalieerkennung dar, welche in Abschnitt 2.6.2.4 näher betrachtet wird. Während bei Klassifikations- und Regressionsproblemen das Modell so trainiert wird, dass bekannte Muster erkannt oder vorhergesagt werden, versucht das Modell bei der Anomalieerkennung den Normalzustand zu erlernen und Abweichungen von diesem als Anomalien zu erkennen. Somit kann die Anomalieerkennung eine sinnvolle Ergänzung zu einem Klassifikations- oder Regressionsmodell sein, um Abweichungen vom Normalzustand zu erkennen [96].

2.6.2.2 Merkmalsextraktion

Die Merkmalsextraktion ist ein Verfahren, in welchem neue Merkmale aus bestehenden Daten generiert bzw. extrahiert werden. Die Merkmalsextraktion kann manuell, durch das Einbringen von Fachwissen, oder automatisiert durch Algorithmen erfolgen [84, 97]. Insbesondere für industrielle Datensätze kann die Extraktion von Merkmalen, z.B. Merkmalen aus dem Frequenzbereich von Beschleunigungssensoren, die Performanz der ML-Modelle steigern [8, 98]. Ziele der Merkmalsextraktion sind u.a. [93]:

- Die Generierung neuer bzw. signifikanterer Merkmale
- Die Dimensionsreduktion des Datensatzes
- Die hinreichend genaue Abbildung des Datensatzes mit möglichst wenigen Merkmalen.

In Zusammenhang mit der erwähnten Dimensionsreduktion steht der sog. Fluch der Dimensionalität (engl. Curse of Dimensionality). Dieser besagt, dass der Merkmalsraum exponentiell mit einer zunehmenden Anzahl von Variablen (Dimensionen) wächst [99]. Dies hat zur Folge, dass eine große Menge an Daten benötigt wird, um den Merkmalsraum zu füllen. Ist diese große Datenmenge nicht gegeben, kann das Modell zur Überanpassung tendieren [100]. Die Extraktion wie auch die nachfolgende vorgestellte Selektion von Merkmalen können die Dimensionalität der Daten reduzieren und so dem Fluch der Dimensionalität entgegenwirken [84].

2.6.2.3 Merkmalsselektion

Die Merkmalsselektion befasst sich mit der Auswahl von relevanten Merkmalen, einerseits zur Dimensionsreduktion (z.B. durch die Entfernung redundanter bzw. irrelevanter Merkmale) und andererseits zur Verbesserung der Performanz (z.B. hinsichtlich der Vorhersagequalität) des ML-Modells [93]. Die Auswahl kann dabei sowohl händisch, bspw. durch Domänenexperten mit entsprechendem Fachwissen, als auch automatisch durch Algorithmen erfolgen. Bei der automatischen Selektion mittels Algorithmen wird zwischen drei Methoden unterschieden [93, 97]:

- **Filter-Methoden**: Merkmalsselektion mittels der Auswertung statistischer Analysen und anschließender Ordnung nach individueller Relevanz.
- **Wrapper-Methoden**: Merkmalsselektion anhand der Performanz verschiedener Merkmalssets in einem kreuzvalidierten Szenario und dem sukzessiven Entfernen irrelevanter Merkmale.

- **Embedded-Methoden**: Merkmalsselektion ähnlich der Wrapper-Methoden, allerdings erfolgt die Selektion als integrierter Teil der Modellbildung (Klassifikation/Regression).

2.6.2.4 Anomalieerkennung

Die Anomalieerkennung (engl. Anomaly Detection oder Novelty Detection) befasst sich mit der Identifikation von Anomalien in Daten, welche nicht einem im Voraus definierten Normal entsprechen [101]. Beispiele für Anomalien sind u.a. Ausreißer, Defekte oder signifikante Verschiebungen der Verteilung der Daten. Die Anomalieerkennung kann als überwachtes, semi-überwachtes und unüberwachtes Modell trainiert werden und in folgende Methodengruppen eingeteilt werden [96]:

- Probabilistische Methoden
- Distanzbasierte Methoden
- Domänenbasierte Methoden
- Rekonstruktionsmethoden
- Informationstheoretische Methoden

Abbildung 2.11 zeigt eine schematische Darstellung einer distanzbasierten Anomalieerkennung mit definiertem Grenzwert.

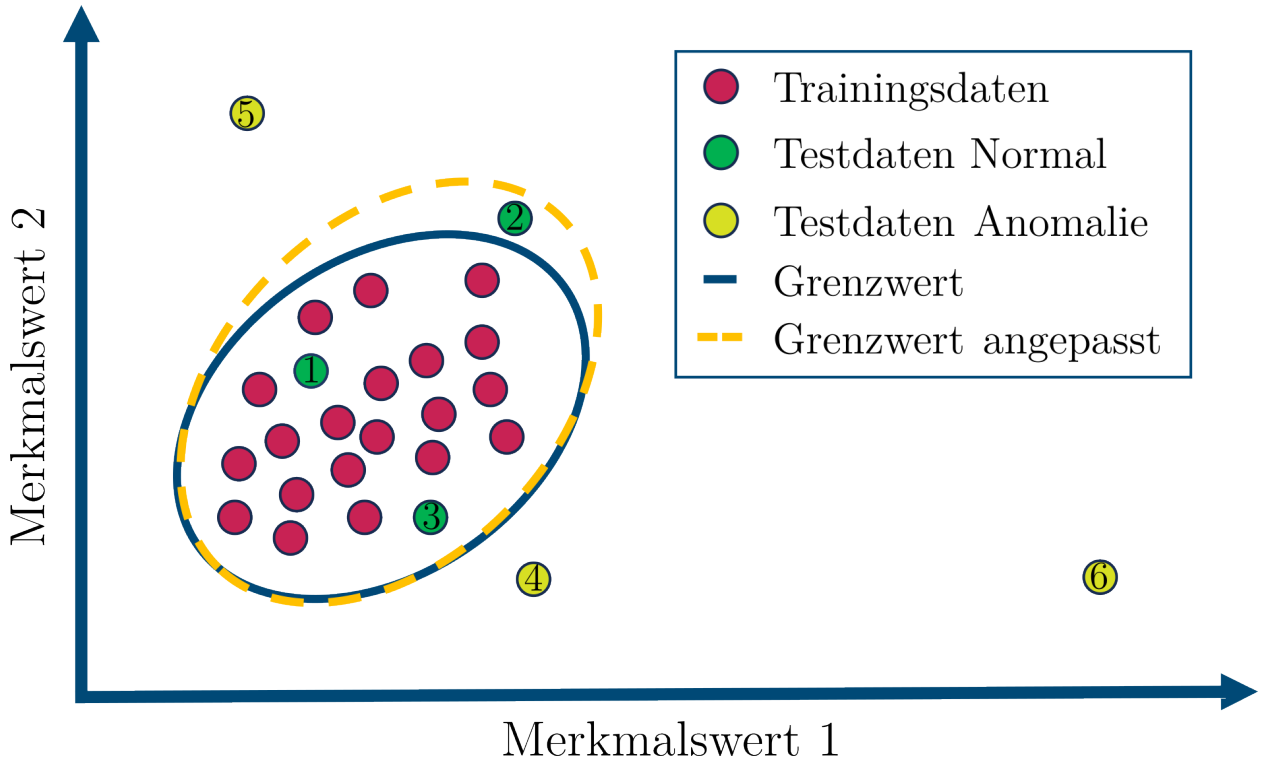


Abbildung 2.11: Schematische Darstellung einer Anomalieerkennung.

Neue Messungen (Testdaten), die außerhalb des anhand der Trainingsdaten definierten Grenzwerts (blaue Linie) liegen, werden zunächst als Anomalien gekennzeichnet. Beispiele hierfür sind die Messungen 2 sowie 4 bis 6. Ob diese Messungen tatsächlich als Anomalien einzustufen sind, hängt von einer nachträglichen Untersuchung ab, bei der fach- bzw. domänenspezifisches Wissen herangezogen wird. Falls eine Messung wie bspw. Messung 2 fälschlicherweise als Anomalie klassifiziert wurde, kann der Grenzwert entsprechend angepasst werden (gelbe gestrichelte Linie).

Insbesondere industrielle Daten sind zahlreichen Stör- und Einflussgrößen ausgesetzt (vgl. Tabelle 2.1), weswegen die Anomalieerkennung eine sinnvolle Ergänzung zum eigentlichen ML-Modell sein kann. In Studie [102] wird folgender Workflow zur Erkennung bekannter und unbekannter Anlagen- und Sensorfehler mit der ML-Toolbox (Abschnitt 2.6.3) vorgestellt:

1. Merkmalsextraktion und -selektion durch die ML-Toolbox.
2. Standardisierung von Merkmalen zur Kompensation von Skaleneffekten.
3. Trainieren eines Modells zur Anomalieerkennung.
4. Verwenden eines Histogramms zur Identifikation von Anomalien.
5. Auswahl eines Grenzwertes, ab welchem ein Messwert als Anomalie gekennzeichnet wird.
6. Verwenden von Progressionsplots zur Korrelation von Anomalien mit physikalischen Ereignissen.

2.6.2.5 Validierung

Die Validierung ist ein essenzieller Teil des maschinellen Lernens und wird zur Überprüfung der Vorhersagequalität eines ML-Modells genutzt [72]. Oftmals wird hierzu eine Teilmenge des Datensatzes gebildet, die für das Training des Modells verwendet wird, während weitere Teilmengen für spätere Test- und Optimierungszwecke zurückgehalten werden. Im Rahmen dieser Dissertation werden folgende Terminologien verwendet:

- **Validieren**: Das Testen und Anpassen von Hyperparametern zur Verbesserung des ML-Modells mit Daten, welche nicht im Training enthalten waren (Validierungsdaten).

- **Testen**: Das Testen eines validierten ML-Modells zur Bestimmung der Performanz mit Daten, die weder im Training, noch in der Validierung enthalten waren (Testdaten).

Die Aufteilung in Trainings-, Validierungs- und Testdaten ist variabel, wobei die Teilmenge der Trainingsdaten i.d.R. die größte Teilmenge repräsentiert. Eine gängige Aufteilung ist z.B. 70 % Trainingsdaten, 15 % Validierungsdaten und 15 % Testdaten [103].

Durch die Aufteilung in drei Teilmengen ergeben sich folgende Ziele für die jeweiligen Prozesse Training, Validierung und Test [69]:

- **Training**: Befähigung des Modells, Zusammenhänge der Daten und ihren Zielgrößen zu finden.
- **Validierung**: Prüfung des Modells unter Variation der Hyperparameter hinsichtlich seiner Robustheit gegenüber Störgrößen wie bspw. Rauschen und anschließende Auswahl der Hyperparameter, welche den geringsten Validierungsfehler erzeugen.
- **Test**: Prüfung des validierten Modells zur Überprüfung hinsichtlich dessen Robustheit gegenüber unbekannten Daten.

Das übergeordnete Ziel der Validierung ist sicherzustellen, dass das ML-Modell die Realität ausreichend abbildet, um zuverlässige Ergebnisse zu erzeugen. Hier wird zwischen der Überanpassung (engl. Overfitting) und der Unteranpassung (engl. Underfitting) unterschieden. Ein Modell mit Überanpassung passt sich zu stark an die Trainingsdaten an, z.B. indem es auch Rauschmuster erlernt, und kann daher nicht ausreichend generalisieren [104]. Modelle mit einer Unteranpassung hingegen können die Realität nicht hinreichend abbilden [105]. Durch die Validierung kann somit überprüft werden, inwiefern ein Modell die Realität abbilden kann.

Im Rahmen dieser Dissertation werden die k-fache Kreuzvalidierung und die Leave-One-Group-Out-Cross-Validation (LOGOCV) verwendet [103]. Die k-fache Kreuzvalidierung unterteilt den Datensatz in k gleichgroße Teilmengen, auch Folds genannt. Bei bspw. einer 10-fachen Kreuzvalidierung ($k = 10$) wird der Datensatz in 10 Folds unterteilt. Anschließend wird mit neun dieser Folds das Modell trainiert und der verbleibende Fold zur Validierung verwendet. Dieser Prozess wird zehn mal wiederholt, sodass jeder Fold einmal zur Validierung genutzt wurde. Ähnlich zur k-fachen Kreuzvalidierung wird der Datensatz bei der gruppenbasierten Kreuzvalidierung ebenfalls in Folds unterteilt. Die Anzahl der Folds entspricht dabei der Anzahl der

vorher festgelegten Gruppen, in welche die Daten eingeteilt werden. Anzumerken ist jedoch, dass die LOGOCV bei rein kategorischen Klassifizierungen nicht verwendet werden kann. Weiterhin muss bei der Auswahl der Gruppen darauf geachtet werden, dass ML-Modelle i.d.R. nicht für die Extrapolation geeignet sind und daher Grenzwertgruppen in den Trainingsdaten verbleiben sollten [106].

Abbildung 2.12 zeigt beispielhaft die Unterteilung eines Datensatzes in vier Folds.

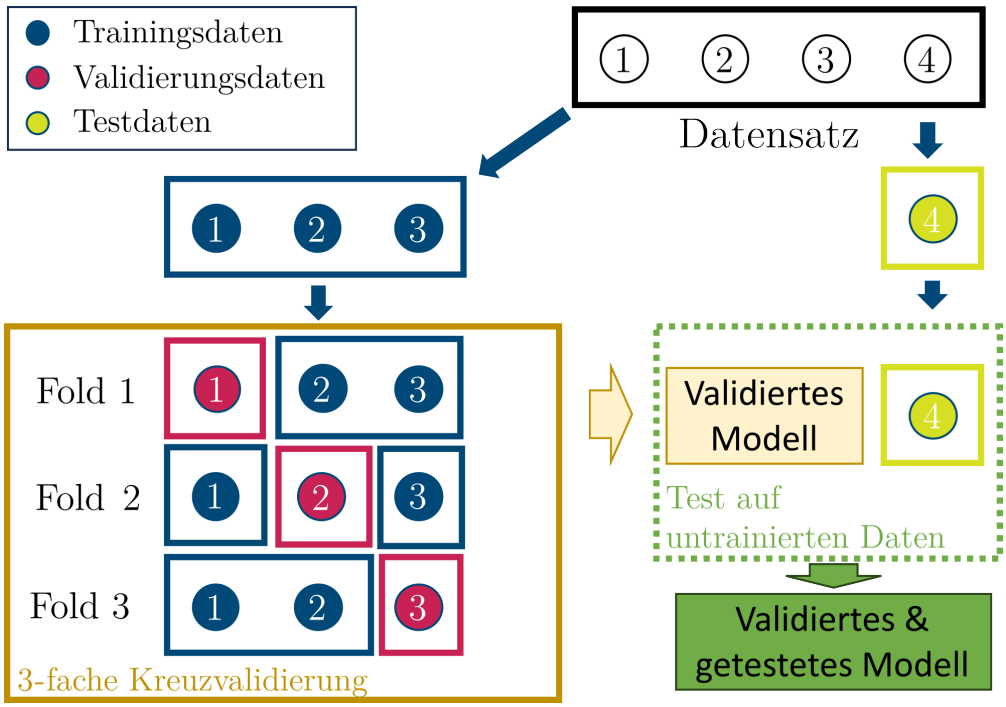


Abbildung 2.12: Schematische Darstellung einer 3-fachen Kreuzvalidierung mit anschließendem Test eines ML-Modells.

Drei der Folds dienen zum Trainieren und Validieren eines Modells, während der vierte Fold zum Testen des Modells auf bisher ungesehenen Daten verwendet wird. Alternativ kann die Validierung ebenfalls als sog. verschachtelte Kreuzvalidierung durchgeführt werden, bei der jeder Fold einmal zum Testen des Modells verwendet wird, um ein erweitertes Modellbewertungskriterium zu erhalten [103]. Für binäre Klassifikationsprobleme und Algorithmen mit wenigen Hyperparametern zeigt sich jedoch, dass die verschachtelte Kreuzvalidierung in vielen praktischen Anwendungen nicht erforderlich ist [107]. Eine Übersicht über gängige Validierungsmethoden findet sich in [103].

2.6.3 Automatisierte Toolbox für maschinelles Lernen

Die große Anzahl verfügbarer komplexer Algorithmen des ML kann, insbesondere für neue bzw. unerfahrene Anwender, abschreckend und überfordernd wirken. Speziell

Unternehmen im Mittelstand haben häufig keine dezidierten Abteilungen, die sich mit der Datenanalyse mittels KI beschäftigen. In der IDG-Studie [3] gaben 31,1 % der befragten Unternehmen an, mit der Wahl eines ML-Modells überfordert zu sein, wobei der Anteil in kleinen Unternehmen (< 1000 Mitarbeiter) auf 39 % ansteigt. Eine potentielle Lösung können sog. KI-Toolboxen sein, welche eine Art Baukasten mit KI-Algorithmen darstellen und anwenderfreundlich gestaltet sind. Populäre Beispiele sind hier u.a. die Low-Code-Entwicklungsplattform KNIME (Konstanz Information Miner) [108], die digitale Datenplattform Power BI [109], oder die Python Datenanalyse Bibliothek Pandas [110]. Im Rahmen dieser Dissertation wird nachfolgend zunächst die auf Github (`https://github.com/ZeMA-gGmbH/LMT-ML-Toolbox`) veröffentlichte Basisversion der automatisierten Toolbox für maschinelles Lernen (ML-Toolbox) [111] vorgestellt und deren Auswahl begründet. Die ML-Toolbox basiert auf MATLAB® und kombiniert die fünf Merkmalsextraktionsmethoden.

- Adaptive Lineare Approximation (ALA)
- Hauptkomponentenanalyse (PCA)
- Beste Fourier-Koeffizienten (BFC)
- Beste Daubechies Wavelet-Koeffizienten (BDW)
- Statistische Momente (SM)

mit den drei Merkmalsselektionsmethoden:

- Recursive Feature Elimination Support Vector Machine (RFESVM)
- ReliefF
- Pearson-Korrelation

und führt anschließend eine Klassifikation mittels linearer Diskriminanzanalyse und Mahalanobis Distanz durch. Durch diese Algorithmenkombination entstehen 15 Pfade (Kombinationen aus jeder Merkmalsextraktionsmethode mit jeder Merkmalsselektionsmethode) durch die Toolbox (vgl. Abbildung 2.13), wovon der beste Pfad durch eine interne 10-fache Kreuzvalidierung auf Basis des geringsten resultierenden Validierungsfehlers ermittelt wird. Dabei wird nach jeder Merkmalsextraktionsmethode, sollte die extrahierte Anzahl an Merkmalen 500 übersteigen, eine Vorselektion mittels Pearson-Korrelation auf 500 Merkmale durchgeführt, um Rechenleistung einzusparen. Darüber hinaus bestimmt die interne

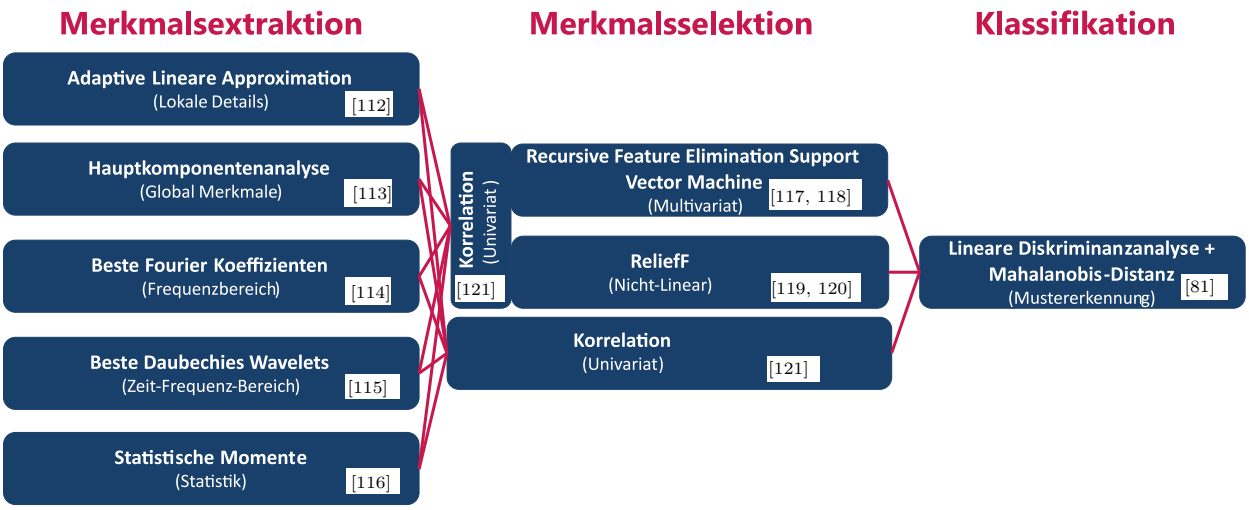


Abbildung 2.13: Automatisierte ML-Toolbox zur Klassifikation und den verwendeten Algorithmen, adaptiert aus [122].

`NumFeatRanking()`-Funktion der ML-Toolbox eine geeignete Anzahl selektierter Merkmale zur Minimierung des Validierungsfehlers [8]. Durch die komplementären Algorithmen konnte die ML-Toolbox bereits gute Resultate in einem breiten Anwendungsfeld [97, 123] und speziell im industriellen Kontext [8, 84, 124] erzielen. In den Studien [92, 124] wird gezeigt, dass klassische FESR/C-Ansätze (Merkmalsextraktion, Merkmalsselektion und Regression/Klassifikation; engl. feature extraction, selection and regression/classification) im Vergleich zu NN hinsichtlich des Validierungsfehlers eine vergleichbare Performanz aufweisen können. Gleichzeitig bieten sie jedoch eine geringere Modellkomplexität und eine höhere Interpretierbarkeit. Eine detaillierte Beschreibung der Funktionsweise der aufgelisteten Algorithmen findet sich in Anhang A.1.3.

Durch die Low-Code-Anwendungsmöglichkeit der ML-Toolbox eignet sich die Toolbox auch für Benutzer mit wenig Programmiererfahrung. Listing 2.1 zeigt den notwendigen Programmieraufwand zur Anwendung der kompletten Toolbox.

```
1 addPaths; %Ordner und Unterordner dem Pfad hinzufuegen
2 load dataset.mat %Laden des Datensatzes
3 %Objekt erstellen:
4 fulltoolbox = Factory.FullToolboxMultisens();
5 fulltoolbox.train(data,target); %Trainieren des Models
6 prediction = fulltoolbox.apply(data); %Modellanwendung
```

Listing 2.1: Quellcode zur Anwendung der kompletten Toolbox [125].

Einzelne Algorithmenkombinationen können mittels des SimpleTrainingStack()-Objekts (vgl. Listing 2.2) aufgerufen werden.

```
1 obj = SimpleTrainingStack(...
2 {@MultisensorExtractor,@Pearson,@NumFeatRanking,...
3 @LDAMahalClassifier},...
4 {{@BFCExtractor},{500},{@RFESVM},{}});
```

Listing 2.2: Quellcode zur Anwendung einer bestimmten Algorithmenkombination [111, 125].

Im obigen Beispiel wurden die Merkmalsextraktionsmethode Beste Fourier Koeffizienten (BFC), Pearson-Korrelation mit 500 Merkmalen als Vorselektor, die `NumFeatRanking()`-Funktion mit dem Merkmalsselektor Recursive Feature Elimination Support Vector Machine (RFESVM) und der LDA mit Mahalanobis-Distanzmaß als Klassifikator verwendet.

Neben der ML-Toolbox existiert auch die UA-ML-Toolbox [126], welche die Fortpflanzung von Messunsicherheiten nach GUM durch die Algorithmenkombinationen berücksichtigt. Aufgrund des deutlich höheren Rechenaufwands wird die UA-ML-Toolbox jedoch nur demonstrativ und ergänzend zur ML-Toolbox in den Anwendungsszenarien eingesetzt.

3 Entwicklung einer Checkliste zur Durchführung von KI-Projekten im Mittelstand

Im Forschungsprojekt Messtechnisch gestützte Montage (MessMo) wurde die Notwendigkeit einer umfassenden Mess- und Datenplanung als essenzielle Grundlage für eine hohe Datenqualität und die Anwendung von maschinellem Lernen (ML) im industriellen Kontext identifiziert [127]. Im Rahmen dieser Dissertation wurde daher in mehreren Iterationen eine Checkliste entwickelt, welche die Mess- und Datenplanung aufgreift und ein ML-Projekt ganzheitlich betrachtet. Dabei orientiert sich die Checkliste an der Struktur der *Statistischen Checkliste* [25], erweitert diese und baut auf dieser auf. In einer ersten Iteration der Checkliste wurde in Kooperation mit dem Lehrstuhl für Montagesysteme der Universität des Saarlandes die *Checkliste Mess- und Datenplanung für das maschinelle Lernen in der Montage* (ML-Montage-Checkliste) [128] entwickelt, um den in der Montage aufgetretenen Problemen entgegenzuwirken. In der folgenden Iteration wurde die ML-Montage-Checkliste, unter Berücksichtigung der initial gestellten Forschungsfragen erneut grundlegend überarbeitet, erweitert und der Fokus auf die Nutzung durch unerfahrene Anwender im Mittelstand gerichtet. Sie bildet darüber hinaus das Kernelement des in Kapitel 4 vorgestellten Persönlichen Informationsassistenten (PIA).

Im Rahmen des Mittelstand-Digital Zentrum Saarbrücken wurde die resultierende Checkliste *KI-Projekte erfolgreich umsetzen: Eine Checkliste für den Mittelstand* (im Folgenden Checkliste) auf der Plattform Zenodo publiziert [129] und mittelständischen Unternehmen zur Verfügung gestellt.

Während der Erprobung der Checkliste (vgl. Kapitel 5) konnten mehrere Verbesserungspotenziale identifiziert werden. Diese werden nachfolgend gekennzeichnet und im Anhang in Tabelle A.10 näher beschrieben.

3.1 Ausgangspunkt

Um unerfahrene Anwender bei der Durchführung einer grundlegenden Datenanalyse mittels ML zu unterstützen (Forschungsfrage 3), gilt es zunächst, die Bedarfe der Unternehmen zu identifizieren. So benötigen laut der IDG-Studie [3] diese verstärkt Unterstützung bei den in Tabelle 3.1 dargestellten Teilschritten eines ML-Projektes.

Tabelle 3.1: Teilschritte eines ML-Projektes, bei denen Unternehmen Unterstützung benötigen [3].

Teilschritt	**Anteil** [%]
Datenauswahl	36,2 %
Modellauswahl	34,7 %
Datenbereinigung	32,1 %
Datenverständnis	31,8 %
Datenreduktion	29,2 %
Definition von Use-Cases	26,5 %
Datenanalyse	21,3 %
Interpretation der Ergebnisse	18,1 %
Andere Teilschritte	2,0 %

Die Studie verdeutlicht, dass Unternehmen in mehreren Teilschritten des ML Unterstützung benötigen (Forschungsfrage 3). Obwohl Schritte wie die Datenaufnahme nicht explizit erwähnt wurden, deuten die Ergebnisse des Projektes MessMo darauf hin, dass auch hier ein Bedarf besteht. Um diesen Bedarfen gerecht zu werden, ist die Entwicklung eines Lösungsansatzes unerlässlich, insbesondere da industrielle Daten aufgrund zahlreicher spezifischer Herausforderungen häufig eine mangelhafte bzw. niedrige Datenqualität besitzen (vgl. Herausforderungen nach Wilhelm in Tabelle 2.1) [11]. Die Checkliste berücksichtigt daher auch die Mess- und Datenplanung (MuD), um Anwender dazu zu befähigen, hochwertige Daten aufzuzeichnen (Forschungsfrage 2). Andere bereits existierende Ansätze (vgl. Abschnitt 2.1) besitzen einen tendenziell hohen Umfang oder verfolgen keinen ganzheitlichen Ansatz. Speziell Unternehmen im Mittelstand haben zudem oftmals nur begrenzte personelle und zeitliche Ressourcen, weswegen ein hoher Umfang und eine hohe Komplexität abschreckend wirken können.

Checklisten bieten das Potenzial, sicherzustellen, dass für den Erfolg kritische Schritte eines (ML-)Projektes korrekt bearbeitet werden und sind ein Leitfaden zur Orientierung für unerfahrene Anwender. Dabei können sie Fehler reduzieren und das

Ergebnis eines Vorgangs verbessern, wenn sie den Anwender nicht überfordern [130]. Die im Nachfolgenden vorgestellte Checkliste stellt einen ganzheitlichen Ansatz für die Durchführung von ML-Projekten dar, der den Anwender von der Projektvorbereitung bis zum Projektabschluss begleitet und dabei die in Tabelle 3.1 vorgestellten Bedarfe berücksichtigt. Um aktuelle Konzepte der Forschung zur Generierung hochwertiger Daten wie z.B. die FAIR-Prinzipien zu berücksichtigen, werden diese ebenfalls in Form von Checkpunkten in die Checkliste integriert (Forschungsfrage 1).

3.2 Anpassung des CRISP-DM-Modells

Das Referenzmodell Cross-Industry Standard Process for Data Mining (CRISP-DM) gilt als industrieübergreifender Standard für das Data Mining (vgl. Abschnitt 2.3.1). Zwar bildet der CRISP-DM die Prozesse im Data Mining weitgehend allgemeingültig und umfangreich ab, jedoch fehlen spezifische Anweisungen und wichtige Bereiche wie z.B. die Organisation konkreter Projekte [24]. Speziell für Anwender, welche wenig oder keine Erfahrung im Data Mining besitzen, kann der Umfang und die Komplexität überfordernd wirken. Der angepasste CRISP-DM (Abbildung 3.1) greift die Phasen des ursprünglichen CRISP-DM auf und passt diese an.

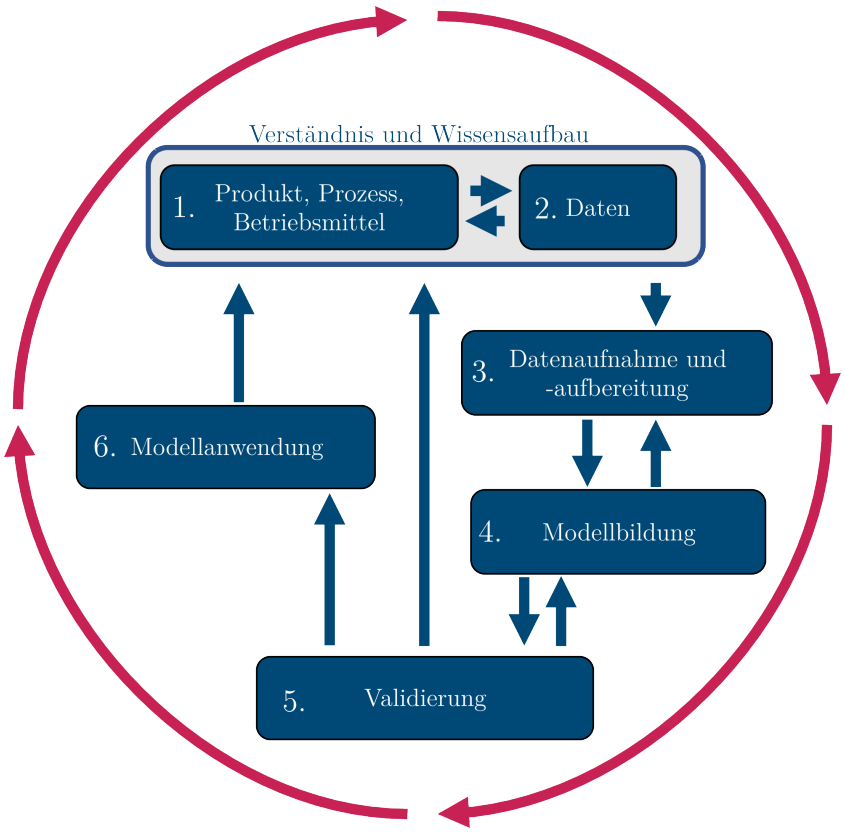


Abbildung 3.1: Angepasster CRISP-DM, adaptiert aus [131].

Die angepassten Phasen sind: Verständnis und Wissensaufbau über Produkt, Prozess, Betriebsmittel (Phase 1) und Daten (Phase 2), die Datenaufnahme und -aufbereitung (Phase 3), die Modellbildung (Phase 4), die Validierung des Modells (Phase 5) und die Modellanwendung (Phase 6) [131].
Hinsichtlich des ursprünglichen CRISP-DM wurden zur Erhöhung der Verständlichkeit und Interpretierbarkeit weiterhin folgende Änderungen durchgeführt:

- Spezifizierung des Geschäftsverständnis zu Verständnis und Wissensaufbau über Produkt, Prozess, Betriebsmittel (Phase 1) und Verdeutlichung der Wechselwirkung mit den Daten (Phase 2). Durch die Spezifizierung und Gruppierung der ersten zwei Phasen kann der Anwender unmittelbar und ohne weiterführende Texte erkennen, dass zum Aufbau von Verständnis und Wissen sowohl die relevanten Produkte, Prozesse und Betriebsmittel als auch die Daten betrachten werden müssen.
- Erweiterung des Punktes Datenaufbereitung zu Datenaufnahme und -aufbereitung (Phase 3).
- Austausch der Evaluation durch die im ML-Kontext übliche Bezeichnung Validierung (Phase 5).
- Spezifizierung der Anwendung zu Modellanwendung (Phase 6), um dem Anwender zu verdeutlichen, dass sich Phase 6 auf die Anwendung des Modells bezieht.

Der angepasste CRISP-DM dient als Grundlage für die nachfolgende Checkliste sowie den Ablaufplan und ist implizit in diesen enthalten.

3.3 Aufbau und Struktur der Checkliste

Die *Statistische Checkliste* [25] stellt den Ausgangspunkt für die entwickelte Checkliste dar und unterstützt die Anwender bei der strukturierten Bearbeitung von Daten bei der Merkmalentstehungs- und -wechselwirkungsanalyse und basiert auf dem DMAIC-Zyklus. Um die *Statistische Checkliste* außerhalb einer Merkmalentstehungs- und -wechselwirkungsanalyse und im Kontext der Montage nutzen zu können, wurde diese in Kooperation mit dem Lehrstuhl für Montagesysteme der Universität des Saarlandes zunächst zur Checkliste *Mess- und Datenplanung für das maschinelle Lernen in der Montage* (ML-Montage-Checkliste) weiterentwickelt. Zu den Änderungen zählen u.a. die Anpassung der Struktur vom DMAIC-Zyklus (vgl. Abschnitt 2.3.2) auf

den allgemeingültigeren CRISP-DM (vgl. Abschnitt 3.2) und die Erweiterung und Ergänzung um produktionsspezifische Inhalte.

Neben wesentlichen inhaltlichen Änderungen und Erweiterungen wurde die im Rahmen dieser Dissertation entwickelte Checkliste auch in der Nutzerfreundlichkeit deutlich gesteigert, u.a. durch die Einbindung eines Ablaufplans. Sie besteht aus den folgenden Kapiteln:

- Vorwort und Vorgehensweise
- Vorbereitung und Projektplanung (Abschnitt 3.4.1)
- Mess- und Datenplanung (Abschnitt 3.4.2)
- Datenaufnahme (Abschnitt 3.4.3)
- Datenprüfung und Datenbereinigung (Abschnitt 3.4.4)
- Datenauswertung und Modellbildung (Abschnitt 3.4.5)
- Projektabschluss (Abschnitt 3.4.6)
- Fazit, Abkürzungsverzeichnis, Glossar und Literaturverzeichnis (Abschnitt 3.4.7)

Jedes der Kapitel leitet zunächst mit einem kurzen Abschnitt in die entsprechende Thematik ein und besitzt einen Buchstaben als Identifikator, wie z.B.: Vorbereitung und Projektplanung (A; Verbesserungspotenzial Nr. 1). Anschließend werden themenspezifische Checkpunkte für jedes Kapitel aufgelistet, welche in Muss-Checkpunkte und Best-Practice-Checkpunkte untergliedert sind. Ein Muss-Checkpunkt ist dabei ein Checkpunkt, dessen Erfüllung zum Projekterfolg erforderlich ist, während die Erfüllung eines Best-Practice-Checkpunkts optional ist, aber empfohlen wird. Jeder Checkpunkt erhält weiterhin eine Nummer, wie z.B.: „A1. Checkliste gelesen und verstanden“, um Anwendern die Orientierung innerhalb der Checkliste zu erleichtern. Am Ende jeder Seite befindet sich eine Abfrage der Vollständigkeit zur Sicherstellung, dass alle Muss-Checkpunkte bearbeitet wurden. Neben den Checkpunkten erhalten Anwender der Checkliste zusätzliche Tipps und Hinweise, welche sie bei der Bearbeitung der Checkliste unterstützen. Abbildung 3.2 zeigt den resultierenden Aufbau und die Struktur der Checkliste.

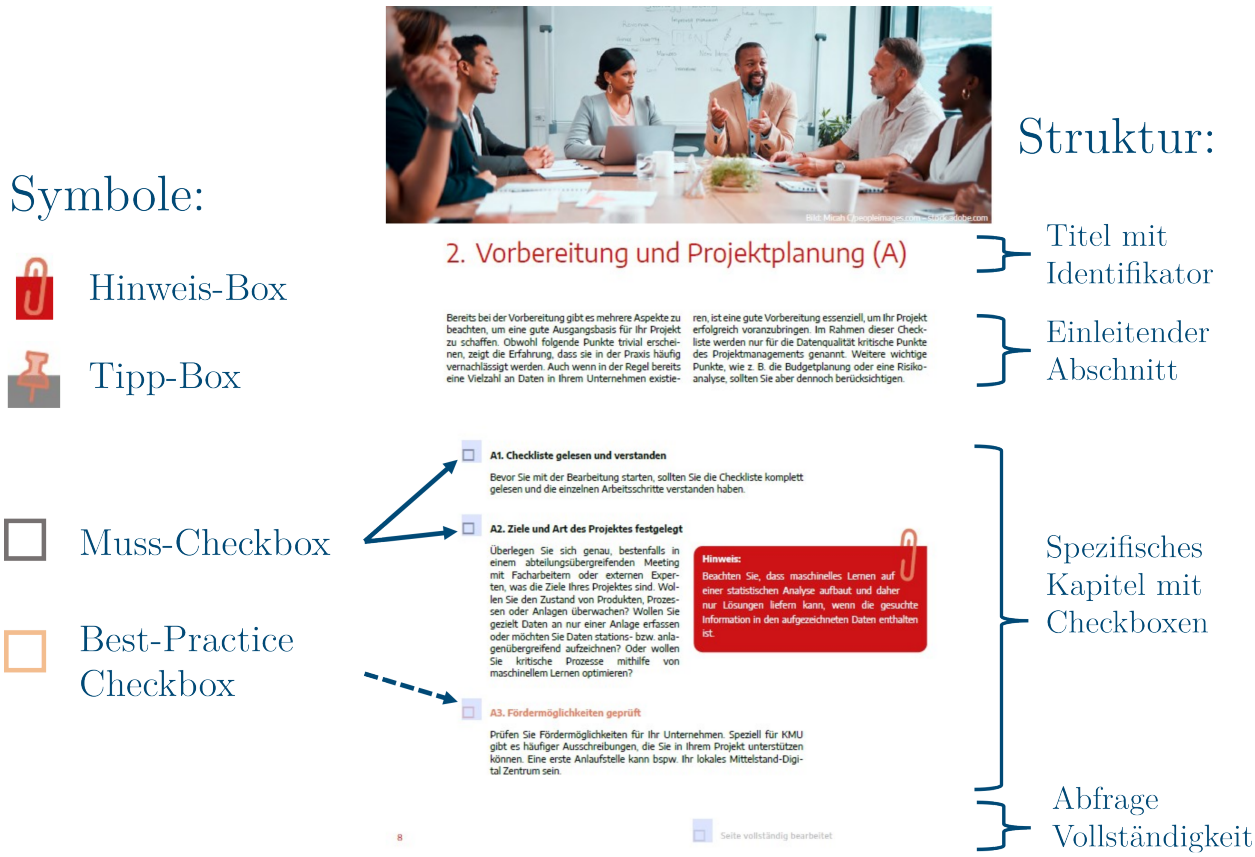


Abbildung 3.2: Aufbau und Struktur der Checkliste [129].

Bei der graphischen Gestaltung der Checkliste wurde der Fokus auf die Nutzerfreundlichkeit und Übersichtlichkeit gelegt, was durch die strukturierte Darstellung und farblich hervorgehobenen Hinweisen bzw. Tipps unterstützt wird. Durch die kompakte Beschreibung der einzelnen Checkpunkte ist der Anwender in der Lage, die erforderlichen Schritte schnell zu erfassen. Die ansprechende und übersichtliche Aufbereitung der Themengebiete erleichtert Anwendern die effiziente Nutzung der Checkliste in der Praxis. Abbildung 3.3 zeigt exemplarisch eine Seite der Checkliste aus dem Kapitel Mess- und Datenplanung. Die vollständige Checkliste kann kostenlos unter `https://zenodo.org/records/10069539` abgerufen werden.

3.5 Datenablage

☐ **B26. Vorhandene Datenerfassungssysteme eingebunden**

Versuchen Sie, alle (relevanten) erfassten Daten vorhandener Datenerfassungssysteme, bestenfalls automatisiert und zentral in Ihrer Datenablage einzubinden.

Hinweis:
Erfahrungsgemäß sind die Schnittstellen vom Sensor zum Server nicht so schnell implementiert wie gewünscht/gedacht. Typische Beispiele wären u.a. das Durchlaufen interner Freigabeprozesse, VPN-Verbindungen oder Firewalls. Planen Sie hier ggf. zusätzliche Zeit ein.

☐ **B27. Speicherbedarf abgeschätzt**

Schätzen Sie den benötigten Speicherbedarf ab und planen Sie eine zusätzliche Sicherheitspuffer ein. Beachten Sie hier auch, dass unterschiedliche Datenformate (B12) einen unterschiedlichen Speicherbedarf haben.

☐ **B28. Geeignete Plattform für Datenablage gewählt**

Wählen Sie, ggf. unter Berücksichtigung von Firmenrichtlinien und dem voraussichtlichen Speicherplatzbedarf, eine geeignete Plattform zur Datenablage. Beispiele wären: Festplatte, Server oder Cloud. Beachten Sie bei sensiblen Daten die jeweils gültigen Sicherheitsstandards (insbesondere bei Servern außerhalb der EU).

☐ **B29. Zugriffsrechte definiert**

Definieren Sie wer (z. B. welche Abteilung, Mitarbeiter etc.) Zugriff auf Ihre abgelegten Daten erhält.

☐ **B30. Datensicherung ausgewählt**

Um den Verlust von Daten zu verhindern, sollten Sie Ihre Daten mit einer geeigneten Strategie speichern (bspw. der 3-2-1-Back-up Regel [13] in Kombination mit RAID-Systemen).

3.6 Manuelle Datenquellen einbinden

☐ **B31. Menschlichen Einfluss reduziert**

Reduzieren Sie den menschlichen Einfluss auf die Daten so weit wie möglich, z. B. durch:

- Einheitliche Benamung anhand eines Glossars
- Dropdownlisten
- Fehlerkategorien
- Standard-Fehlercodes

Tipp:
Die Nutzung von Standards und Dropdownlisten erleichtern die Maschinenlesbarkeit manuell erfasster Daten.

☐ Seite vollständig bearbeitet

Abbildung 3.3: Darstellung von Seite 17 der Checkliste.

3.4 KI-Projekte erfolgreich umsetzen: Eine Checkliste für den Mittelstand

In den nachfolgenden Kapiteln werden die jeweiligen Checkpunkte zu den sechs Kapiteln der Checkliste beschrieben. Die Checkpunkte werden zur besseren Lesbarkeit weitestgehend als Fließtext beschrieben und können durch die entsprechende Nummerierung zugeordnet werden. So entspricht z.B. (A1) dem Checkpunkt A1. Manche der Checkpunkte dienen speziell der Fehlervermeidung (FV), deren Ziel die Vermeidung von Flüchtigkeitsfehlern ist (Verbesserungspotenzial Nr. 2). Sie werden durch die entsprechende Abkürzung gekennzeichnet, wie z.B. (A1, FV).

3.4.1 Vorbereitung und Projektplanung

Im Abschnitt Vorbereitung und Projektplanung (A) wird der Anwender zunächst aufgefordert, die Checkliste komplett zu lesen (A1, FV), um so einen Überblick über die bevorstehenden Aufgaben zu erhalten und den Prozess der Durchführung eines KI-Projektes ganzheitlich nachvollziehen zu können. Weiterhin definiert der Anwender zunächst ein Ziel (A2), wie die Zustandsüberwachung einer Anlage und legt die entsprechenden Randbedingungen des KI-Projekts fest. Der finanzielle Aspekt ist bei der Durchführung von KI-Projekten nicht zu vernachlässigen, wird von der Checkliste jedoch nicht abgedeckt, da der Fokus nicht auf der Finanzierung von Projekten liegt. Allerdings wird aus Gründen der Vollständigkeit der Anwender auf die Prüfung von Fördermöglichkeiten zur Finanzierung des KI-Projektes hingewiesen (A3, optional). Zu den oben genannten Randbedingungen zählen:

- **Identifikation relevanter Facharbeiter/Fachexperten** (A4): Das frühe Identifizieren und Einbeziehen von Facharbeitern und Fachexperten kann wesentlich zur Mess- und Datenplanung (MuD) beitragen, da diese häufig spezifisches Wissen besitzen, welches hilfreich für die Datenanalyse ist.

- **Hinzuziehen externer Berater** (A5, optional): Sollten notwendige Kompetenzen firmenintern nicht vorhanden sein, besteht die Möglichkeit, dass Unternehmen externe Berater hinzuziehen.

- **Prüfung des Archivs auf ähnliche Projekte** (A6): Durch die Prüfung des Archivs auf ähnliche Projekte kann aus den Erfahrungen vorangegangener Projekte gelernt und ggf. Synergien genutzt werden. Sollte noch kein Archiv

vergangener ML- bzw. Datenanalyse-Projekte vorhanden sein, so muss dieses angelegt und über die Zeit aufgebaut werden.

- **Erstellung einer Übersicht möglicher Produkte, Prozesse und Anlagen** (A7): In diesem Schritt kann der Anwender eine Vorauswahl relevanter Produkte, Prozesse und Anlagen für ein ML-Projekt treffen, wie Problemstationen mit hoher Ausschussquote.

Auf Basis der Checkpunkte A1-A7 muss eine erste Aufwandsabschätzung (A8) erfolgen. Nicht jedes der in A7 aufgelisteten Probleme eignet sich für ein KI-Projekt, da die zur Verfügung stehenden Ressourcen oftmals begrenzt sind. Sollte der abgeschätzte Aufwand zu hoch sein, kann auch ein Betrachtungsfokus gewählt werden (A9, optional). So kann bspw. der Fokus statt auf einer ganzen Montagelinie zunächst auf einem kritischen Prozess innerhalb dieser liegen.

Anschließend wird ein Zeitplan (A10) erstellt und die entsprechenden Arbeitspakete müssen abgeleitet und aufgeteilt werden (A11, optional). Essenziell ist auch die Bestimmung einer verantwortlichen Person (A12), welche die Koordination übernimmt und das Projekt vorantreibt. Um den Widerstand der Mitarbeiter (im Sinne des Veränderungsmanagements [132]) so gering wie möglich zu halten, ist eine Aufklärung der Mitarbeiter (A13) über das KI-Projekt und seine Ziele erforderlich. Wenn die Mitarbeiter ebenfalls die Ziele des Projektes kennen und dessen Nutzen bzw. Vorteile verstehen, werden diese das Projekt nicht bewusst ausbremsen. Weiterhin erfolgt die Erstellung eines Lastenhefts (A14, Verbesserungspotenzial Nr. 8) in dem die konkreten Anforderungen des KI-Projektes definiert werden.

Der Großteil der genannten Checkpunkte der Vorbereitung und Projektplanung stammt aus dem klassischen Projektmanagement und ist Anwendern häufig bereits bekannt. Die Bearbeitungsreihenfolge der obigen Checkpunkte kann jedoch variieren, so wird z.B. der Zeitplan (A10) i.d.R. nach dem Lastenheft (A14) erstellt (siehe Diskussion in Kapitel 6). Insbesondere unerfahrenen Anwendern können diese jedoch eine Orientierung bieten, da sowohl allgemeine als auch spezifische Anforderungen für KI-Projekte berücksichtigt werden.

3.4.2 Mess- und Datenplanung

Um hochwertige Daten aufzuzeichnen, welche die Grundlage für die nachfolgende Bildung von Lernmodellen darstellen, ist die Mess- und Datenplanung (MuD) essenziell. Das Kapitel *Mess- und Datenplanung (B)* gliedert sich daher in die sechs Unterkapitel:

- **Prozesswissen aufbauen und nutzen** (Abschnitt 3.4.2.1): Der Aufbau und die Nutzung von Prozesswissen ist ein wichtiger Schritt für die Planung und Aufzeichnung von Daten. So können bspw. durch ein hohes Verständnis über Prozesse relevante Stör- und Einflussgrößen identifiziert werden.
- **Normen und Standards nutzen** (Abschnitt 3.4.2.2): Je nach Branche und Unternehmen gelten ggf. Normen oder es sind bereits interne oder externe Standards vorhanden.
- **Messunsicherheiten** (Abschnitt 3.4.2.3): Da jede Messung die Realität nur bis zu einem gewissen Grad abbilden kann, muss die Messunsicherheit z.B. durch Kalibrierung der Sensoren quantifiziert, und deren Fortpflanzung durch die Algorithmen ermittelt werden.
- **Aufbau der Daten** (Abschnitt 3.4.2.4): In diesem Abschnitt werden die Struktur, das Format und die Metadaten der Daten festgelegt.
- **Datenablage** (Abschnitt 3.4.2.5): Die Datenablage befasst sich mit der Speicherung und der Verwaltung der Daten.
- **Manuelle Datenquellen einbinden** (Abschnitt 3.4.2.6): Auch manuelle Daten wie Informationen über Stillstände aus einem Schichtbuch können wertvolle Informationen für die spätere Datenanalyse darstellen.

Innerhalb der jeweiligen Kapitel werden die wissenschaftlichen Konzepte FAIR-Data, Ursache-Wirkungs-Diagramm (UWD), Versuchsplanung, Messunsicherheit und Metadaten eingebunden. Dabei werden diese Konzepte teilweise auch aufgeteilt, um deren Bearbeitung durch den Nutzer zu jeweils geeigneten Zeitpunkten sicherzustellen.

3.4.2.1 Aufbau von Prozesswissen

Um Wissen über den ausgewählten Prozess aufzubauen, muss der Anwender zunächst die bereits existierenden Messwerte aus aufgezeichneten Daten identifizieren (B1) und deren Relevanz, z.B. für die eigene und andere Abteilungen, festlegen. Bestenfalls werden zusätzlich Qualitätsanforderungen recherchiert (B2, optional) und dokumentiert. Durch die Checkpunkte B1 und B2 kann die Ausgangssituation bestimmt werden. In Zusammenarbeit mit den ausgewählten Facharbeitern bzw. Fachexperten wird anschließend ein Ursache-Wirkungs-Diagramm (UWD) erstellt (B2), um zunächst die Einflussgrößen, welche die Daten beeinflussen, zu bestimmen. Nachfolgend können anhand des UWD die relevantesten Störgrößen identifiziert und

deren Einfluss durch Expertenwissen abgeschätzt werden (B4, Verbesserungspotenzial Nr. 9). In komplexen Industrieanlagen treten oftmals viele Störgrößen gleichzeitig auf, wodurch die Bestimmung des Einflusses der Störgrößen nicht trivial ist. Daher empfiehlt es sich, den Einfluss von Störgrößen nach Möglichkeit zu minimieren (B5, optional). So kann z.B. die Erfahrung von Arbeitern eine Störgröße darstellen, welche durch entsprechende Schulungen minimiert werden kann. Dabei ist sicherzustellen, dass die Minimierung der Störgrößen nicht dazu führt, dass die Daten die Realität nicht mehr ausreichend abbilden. Enthalten die Daten in der späteren Anwendung Störgrößen, die im Training nicht ausreichend berücksichtigt wurden, kann das Modell seine Gültigkeit verlieren. Dies könnte der Fall sein, wenn die Datenaufzeichnung in einem klimatisierten Raum erfolgt, die Anlage aber in einer Halle steht, welche nicht klimatisiert werden kann.

Die Nutzung zusätzlicher Sensorik (B6, optional) kann es dem Anwender ermöglichen, Daten gezielter und an relevanteren Positionen zu erfassen, um so zusätzliches Verständnis über die Wirkzusammenhänge aufzubauen und ggf. die Performanz des Lernmodells zu verbessern. Dies gilt insbesondere für Prozesse, Anlagen und Produkte, welche über keine modernen Schnittstellen verfügen und deren Daten so nicht zugänglich sind. Anschließend wird eine Übersicht aller am betrachteten Prozess (bzw. Produkt oder Anlage) eingesetzten Sensoren inklusive deren Metadaten erstellt (B7). Als Empfehlung werden dem Anwender die folgenden Metadaten aufgelistet: Sensorbezeichnung, Sensorart, IP-Adresse, Hersteller, Messgröße mit der entsprechenden Einheit, Messbereich, Abtastrate bzw. Auslesemethode, Bandbreite, Auflösung, Messunsicherheit und das Ausgangssignal (analog bzw. digital, Kommunikation). Je nach konkretem Anwendungsfall können auch andere Metadaten relevant sein, wobei die oben ausgeführten Elemente einen Ausgangspunkt darstellen. Eine Vorlage einer Sensorübersicht wird dem Anwender am Ende der Checkliste zur Verfügung gestellt.

Nach Abschluss des Abschnitts *Aufbau von Prozesswissen* kann der Anwender die Wirkzusammenhänge von Einflussgrößen und Störgrößen auf die Daten verstehen und zuordnen. Das Verständnis über diese Größen sowie die Festlegung ihrer Grenzen ist essenziell für die Datenaufnahme und Modellbildung, da Modelle i.d.R. nur interpolieren, aber nicht extrapolieren können [106] (Verbesserungspotenzial Nr. 3).

3.4.2.2 Normen und Standards

In großen Industrieunternehmen ist die Festlegung und Anwendung interner Standards, Leitlinien oder Normen eine gängige Praxis und ggf. auch Pflicht. Auch im Mittelstand

können solche Standards existieren bzw. Normen relevant sein. Daher empfiehlt die Checkliste, zunächst die internen Standards zu recherchieren (B8, optional). Sollten entsprechende Standards gefunden werden, so sind diese bevorzugt zu nutzen. Weiterhin müssen auch externe Standards und Normen recherchiert und berücksichtigt werden (B9). So existieren z.B. auf nationaler Ebene das Deutsche Institut für Normung (DIN) und auf internationaler Ebene die Internationale Organisation für Normung (ISO), welche Normen und Standards erstellen. Für den Anwender ist es wichtig, über diese Normen informiert zu sein und diese ggf. auch zu berücksichtigen.

3.4.2.3 Messunsicherheiten

Ein zentraler Aspekt der MuD ist die Berücksichtigung von Messunsicherheiten. Während in Checkpunkt B7 die Messunsicherheit als Bestandteil der Metadatenübersicht empfohlen wird, liegt der Fokus dieses Abschnitts auf der expliziten Ermittlung der Messunsicherheiten durch den Anwender. Die Angabe der Messunsicherheit gehört zu einer vollständigen Beschreibung eines Messergebnisses zwingend dazu. Insbesondere in kleineren Unternehmen kann das notwendige Wissen bzw. ein generelles Bewusstsein über Messunsicherheiten fehlen. Daher wird der Anwender in diesem Abschnitt der Checkliste zunächst auf die Thematik der Messunsicherheiten und deren Relevanz für die Leistungsfähigkeit im ML hingewiesen. Außerdem erhält der Anwender Referenzen zu weiterführender Fachliteratur (u.a. [49, 53, 54]) und bekommt als Tipp die Uncertainty-Aware Automated Machine Learning Toolbox (UA-ML-Toolbox) vorgestellt, welche die Fortpflanzung von Messunsicherheiten durch die Lernalgorithmen berücksichtigt und eine Abwandlung der Automatisierten ML-Toolbox für zyklische Daten (ML-Toolbox) ist.

Als ersten Handlungsschritt muss der Anwender daher die Quellen der Messunsicherheiten in der Messkette bestimmen (B10). Anschließend wird die Unsicherheit der eingesetzten Messmittel, z.B. im Kalibrierzertifikat, identifiziert (B11). Sollte kein Kalibrierzertifikat vorhanden sein, kann diese auch grob mittels des Sensordatenblattes geschätzt werden. Im Kontext der MuD ist die Messunsicherheit zunächst abgeschlossen (Verbesserungspotenzial Nr. 5) und wird wieder in der Modellbildung (Abschnitt 3.4.5.3) aufgegriffen.

3.4.2.4 Aufbau der Daten

Im nächsten Schritt legt der Anwender den Aufbau der Daten fest. An dieser Stelle werden die FAIR (Findable, Accessible, Interoperable, Re-usable)-Prinzipien in die Checkliste eingebunden.

Im ersten Schritt wird, ggf. in Rücksprache mit den Prozess- und Datenanalyse-experten ein passendes Dateiformat ausgewählt (B12), welches idealerweise nicht proprietär ist und einfach um Metadaten ergänzt werden kann. Dies ist insbesondere für die spätere Zugänglichkeit relevant. Anschließend wird die Struktur der Daten festgelegt (B13). Dazu gehört auch die Festlegung einer geeigneten Dateigröße (Trade-off zwischen vielen kleinen Dateien und einer großen Datei) und die Prüfung der Kompatibilität mit bereits verwendeten Datenbanken. Durch eine eindeutige Benennung der Daten (B14) sind diese leichter aufzufinden und zu handhaben. Je nach Komplexität der Anlage bzw. des Vorhabens kann eine beschreibende Benennung der Daten (B15, optional) in Form strukturierter Namensbestandteile wie z.B. Datum_Uhrzeit_Produkt_Station_Sensor, die Auffindbarkeit und Handhabbarkeit durch Menschen weiter steigern. Hier gilt es zu berücksichtigen, dass lange Dateinamen bzw. Dateipfade z.B. durch eine Zeichenbegrenzung des Betriebssystems zu Problemen in der Auswertung führen können. Die konkrete Struktur sollte daher zweck- und systemgerecht gewählt werden. Empfehlenswert ist außerdem die Herstellung einer eindeutigen Beziehung der Daten zu den Produkten, Prozessen oder Anlagen (B16, optional), z.B. über den beschreibenden Dateinamen oder durch eine eindeutige Identifikationsnummer (ID). Essenziell für die Interoperabilität und Wiederverwertbarkeit ist die Auswahl der aufzuzeichnenden Metadaten (B17) und deren verständliche, bestenfalls maschinenlesbare Beschreibung (B18). Um maschinenlesbare Metadaten zu erzeugen, können Terminologien und Ontologien verwendet werden.

Da eine händische bzw. nachträgliche Annotation von Metadaten ein potenzielles Risiko für Fehler bietet, wird die Implementierung einer automatisierten Annotation empfohlen (B19, optional). Die nachfolgenden Schritte beziehen sich zum Teil auf Checkpunkt B17, werden aber aufgrund ihrer Relevanz und zur Fehlervermeidung explizit behandelt:

- **Festlegung eines einheitlichen Formates der Zeitstempel sowie deren Dokumentation** (B20, FV): Je nach Region können unterschiedliche Darstellungen von Zeit und Datum gelten, wie z.B. Monat/Tag/Jahr (USA),

Tag/Monat/Jahr (Deutschland), oder aber verschiedene Zeitzonen. Um Missverständnissen vorzubeugen, muss die Darstellungsart ersichtlich sein.

- **Klärung und Dokumentation von Bezugssystemen** (B21, optional und FV): Messgrößen können in unterschiedlichen Bezugssystemen dargestellt werden. Diese Bezugssysteme können z.B. unterschiedliche Nullpunkte, Referenzen, Koordinatensystemen oder Einheiten aufweisen. So kann bspw. eine Temperatur in Grad Celsius, Fahrenheit oder Kelvin angegeben werden. Ohne die Kenntnis des verwendeten Bezugssystems kann es zu einer Fehlinterpretation der Daten kommen und die Interoperabilität der Daten ist nicht gewährleistet.

- **Kennzeichnung von Referenzfahrten und Testmessungen** (B22, FV): Da interne Prüfprozesse wie z.B. Referenzfahrten und Testmessungen sich ggf. von den im Normalbetrieb aufgezeichneten Daten unterscheiden, gilt es diese entsprechend zu kennzeichnen (Verbesserungspotenzial Nr. 10). Ist dies nicht der Fall, können Modelle ggf. nicht genug generalisieren und ihre Vorhersage basierend auf falschen Annahmen treffen.

- **Speicherung von Grenzwerten** (B23, optional): Im industriellen Umfeld werden oftmals Grenzwerte für Prozesse festgelegt, welche die Grenzen des Prozesses im Normalbetrieb markieren. Das Über- bzw. Unterschreiten der Grenzwerte kann auf Verschleiß oder eine Fehlfunktion hindeuten. Die Aufnahme der Grenzwerte in die Metadaten kann bei der Analyse helfen, die Messwerte zu interpretieren bzw. in einen Kontext zu setzen.

- **Klare Definition von Messwerten und Kennzahlen** (B24, FV): Insbesondere bei Kennzahlen existieren unterschiedliche Formeln zu deren Berechnung. So kann bspw. der in der Industrie häufig genutzte Indikator Overall Equipment Effectiveness (deutsch: Gesamt-Anlagen Effektivität) auf mehrere Arten berechnet werden [133]. Die Nachvollziehbarkeit der Berechnung von Messwerten oder Kennzahlen muss daher gewährleistet sein.

- **Sicherung von nicht-digitalem Wissen bzw. Digitalisierung von Daten** (B25, optional): Für ältere Bestandsanlagen ohne moderne Datenschnittstellen (vgl. Herausforderung S1 nach Wilhelm in Tabelle 2.1) wird häufig ein handschriftliches Schichtbuch zur Dokumentation von Systemänderungen geführt (vgl. Herausforderung D5 nach Wilhelm in Tabelle 2.1). Die Einträge in solchen Schichtbüchern können wertvolle Informationen über Wirkzusammenhänge bei Problemen liefern. So können Modifikationen an einer Anlage die Daten und

deren Qualität beeinflussen. Daher ist es für eine bessere Zugänglichkeit dieser Informationen vorteilhaft, sie in digitaler Form verfügbar zu machen.

Nachdem der Aufbau und die Struktur der Daten anhand der obigen Checkpunkte ausgewählt wurden, gilt es, diese im folgenden Abschnitt in eine Datenablage einzubinden.

3.4.2.5 Datenablage

Im ersten Schritt müssen die bereits vorhandenen, relevanten Datenerfassungssysteme bzw. Datenquellen in die Datenablage eingebunden bzw. dort zusammengeführt werden (B26). Dieser Schritt kann aufgrund z.B. fehlender Schnittstellen (vgl. Herausforderung S1 nach Wilhelm in Tabelle 2.1) von Bestandsanlagen, der komplexen Datenfusion heterogener Datenquellen (Herausforderungen D1 und P1 nach Wilhelm in Tabelle 2.1) oder auch interner (Freigabe-)Prozesse unter Umständen zeitaufwendig sein. Anhand der zusammengeführten Datenquellen muss anschließend eine Abschätzung des Speicherbedarfs für die Daten des ML-Projektes erfolgen (B27). Zu berücksichtigen ist hier auch eine Sicherheitsmarge für z.B. die Daten zusätzlich angebrachter Sensorik oder zusätzlichen Messungen. Anhand des zuvor definierten Aufbaus der Daten und der Schritte B26-B27 muss anschließend eine geeignete Plattform für die Datenablage gewählt werden (B28). Die Auswahl der Datenablage sollte dabei unter Berücksichtigung der Firmenrichtlinien und unter Absprache der beteiligten Abteilungen erfolgen. Anschließend können zusätzlich Zugriffsrechte für die Datenablage definiert werden (B29, optional).

Weiterhin sollte, um dem Verlust von Daten vorzubeugen, ein Konzept zur Datensicherung integriert werden (B30, optional). Dies kann z.B. durch ein RAID (Redundant Array of Independent Disks) realisiert werden, welches Daten redundant auf einem Festplattenverbund speichert [134].

3.4.2.6 Manuelle Datenerfassung

Neben einer automatisierten Erfassung von Daten können diese auch manuell erfasst werden. So sind z.B. Handarbeitsplätze oft nicht mit einer automatischen Datenerfassung ausgestattet. Stattdessen dokumentiert der Arbeiter Mess- oder Prüfwerte, aber auch Ereignisse wie Maschinenstillstände manuell. Die digitale Erfassung dieser Messdaten, z.B. durch ein Tablet mit einer benutzerfreundlichen Bedienoberfläche, ist jedoch relevant, da diese Informationen Erkenntnisse über

Wirkzusammenhänge enthalten können und somit zur Datenanalyse beitragen. Hier können die folgenden Checkpunkte den Anwender unterstützen:

- **Reduzierung des menschlichen Einflusses** (B31): Bei der Erfassung manueller Daten muss zunächst der menschliche Einfluss, z.B. durch die Einführung eines Glossars, Standard-Fehlercodes oder Fehlerkategorien reduziert werden. So können diese auch maschinenlesbar gestaltet werden.
- **Einführung digitaler Schichtbücher** (B32, optional): Falls vorhanden, sollten analoge Schichtbücher durch digitale Schichtbücher ersetzt werden. Dies gibt dem Datenanalysten die Möglichkeit, z.B. bei Unstimmigkeiten in den Daten auf leicht auffindbare und zugängliche Art das Schichtbuch auf besondere Ereignisse zu prüfen.
- **Sicherstellung einer durchgängigen Speicherung von Zustandsänderungen** (B33): Die durchgängige Speicherung von Zustandsänderungen umfasst insbesondere Änderungen, die von Arbeitern durchgeführt werden, wie z.B. der Austausch von Betriebsmitteln oder die Anpassung von Parametern einer Anlage. So wird sichergestellt, dass diese Änderungen auch im Nachgang nachvollzogen und den Daten zugeordnet werden können.
- **Einbindung eines Kommentarfeldes** (B34, optional): Zur Abdeckung unvorhergesehener Ereignisse kann ein Kommentarfeld implementiert werden, welches die Eingabe von Fließtext zur Fehlerbeschreibung ermöglicht.
- **Einbindung von Bilddaten** (B35, optional): Bilder können als zusätzliche Datenquelle zur Interpretation von Fehlerzuständen oder Defekten dienen.
- **Automatisierte Zuordnung manuell erfasster Daten** (B36, optional): Die automatisierte Zuordnung der manuell erfassten Daten zu den entsprechenden automatisiert erfassten Daten reduziert die Wahrscheinlichkeit einer fehlerhaften Zuordnung durch einen Mitarbeiter und erleichtert die Auswertung der Daten. Um die Daten zuordnen zu können, eignen sich z.B. die Verwendung eindeutiger IDs oder (synchrone) Zeitstempel.

Nach Abschluss des Abschnitts *Manuelle Datenerfassung* ist die MuD vorerst abgeschlossen, sodass die Vorbereitungen für die Datenaufnahme beginnen können. Der Übergang von der MuD zur Datenaufnahme verläuft dabei fließend, da z.B. nach der Durchführung erster Testmessungen auch Anpassungen in der MuD erforderlich werden können.

3.4.3 Datenaufnahme

Der Abschnitt Datenaufnahme (C) gliedert sich in die zwei Abschnitte *Erste Testmessung* und *Überprüfung der Datenqualität und Langzeitdatenaufnahme*. Die Kapitel *Mess- und Datenplanung* und *Datenaufnahme* sind eng miteinander verbunden. So können nach den ersten Testmessungen neue Erkenntnisse gewonnen oder Probleme identifiziert werden, welche eine Anpassung der MuD erforderlich machen.

Vor der ersten Testmessung müssen zunächst sämtliche notwendigen hard- und softwareseitigen Vorbereitungen abgeschlossen werden (C1). Dieser Checkpunkt dient primär der Fehlervermeidung, da im Rahmen der MuD idealerweise alle notwendigen Vorbereitungen abgeschlossen wurden.

3.4.3.1 Erste Testmessung und Überprüfung der Datenqualität

Sobald sichergestellt wurde, dass die hard- und softwareseitigen Vorbereitungen abgeschlossen wurden, kann die Durchführung einer ersten Testmessung (C2) unter Real-Bedingungen erfolgen. Anhand dieser Testmessung kann der Anwender die folgenden Checkpunkte überprüfen:

- **Überprüfung der Datenstruktur** (C3): Zunächst muss der Anwender überprüfen, ob die tatsächliche Datenstruktur der in Abschnitt 3.4.2 festgelegten Struktur entspricht. Dies erfolgt bestenfalls in einem abteilungsübergreifenden Team mit den beteiligten Fachexperten und Datenanalysten, um die Daten auf Verständlichkeit und Vollständigkeit zu überprüfen.

- **Funktionsprüfung der Sensoren** (C4): Weiterhin muss die ordnungsgemäße Funktion der Sensoren überprüft werden. Hier empfiehlt sich ebenfalls die Zusammenarbeit mit Fachexperten und eine erste Visualisierung der Daten, z.B. durch Plotten der Messwerte über die Zeit. In den visualisierten Daten können u.a. fehlerhafte Messwerte oder Quantisierungsstufen leicht erkannt werden.

- **Synchronität der Datenerfassungssysteme** (C5): Beim Einsatz mehrerer Datenerfassungssysteme muss vom Anwender geprüft werden, ob die Synchronität unter den Systemen gewährleistet ist. Insbesondere bei hohen Stückzahlen und niedriger Taktzeit steigen die Anforderungen an die Synchronität, da die Differenz der jeweiligen Zeitstempel nicht die Taktzeit überschreiten darf [25]. Die Zuordnung der Daten kann über eine eindeutige Produkt-ID erfolgen; Zeitstempel dienen hierbei als redundante Absicherung (vgl. B16).

- **Zuordenbarkeit der Daten** (C6): Die Überprüfung der Zuordenbarkeit der Daten zu den jeweiligen Prozessen, Produkten und Anlagen ist essenziell für die spätere Analyse der Daten, insbesondere um Prozess- oder anlagenübergreifende Wirkzusammenhänge zu identifizieren.
- **Überprüfung der Datenqualität** (C7): Anhand der Testmessung muss eine erste Überprüfung der Datenqualität erfolgen. Sollte weder in den firmeninternen Richtlinien noch in Normen o.ä. spezifische Vorgaben enthalten sein, können z.B. die in Abschnitt 2.4 vorgestellten 15 IQ-Dimensionen oder die vier Datenqualitätsmetriken nach Heinrich (Abschnitt 2.4.3.2) als Orientierung zur Auswahl relevanter Kriterien verwendet werden. Auch eine erste Überprüfung der FAIRness der Daten kann in diesem Schritt erfolgen.
- **Implementierung einer automatisierten Prüfung** (C8, optional und FV): Durch die fortlaufende Prüfung der Daten und deren Qualität während der Langzeitdatenaufnahme können Abweichungen oder Unregelmäßigkeiten frühzeitig erkannt werden. Auf diese Weise wird verhindert, dass diese die Qualität der gesamten Daten negativ beeinflussen wird.

3.4.3.2 Langzeitdatenaufnahme

Vor dem Start der Langzeitdatenaufnahme durch den Anwender müssen Prüfintervalle zur Überprüfung der Daten analog zu Schritt C2-C7 festgelegt werden (C9, Verbesserungspotenzial Nr. 11). Anschließend folgt der Start der Langzeitdatenaufnahme (C10).

Im Rahmen der festgelegten Prüfintervalle sollte auch eine Überprüfung der Datenverteilung erfolgen (C11, optional). Die Überprüfung kann hierbei visuell, z.B. mittels Histogrammen, oder rechnerisch [135] durchgeführt werden. So können die in der Industrie üblichen, ungleichmäßigen Datenverteilungen (vgl. Herausforderung P3 nach Wilhelm in Tabelle 2.1) frühzeitig identifiziert werden. Weiterhin kann innerhalb der Prüfintervalle eine erste Datenanalyse durchgeführt werden (C12, optional), um das Potenzial der Datengrundlage unter Einbindung von Fachexperten abzuschätzen.

3.4.4 Datenprüfung und Datenbereinigung

Ziel der Kapitel A bis C der Checkliste ist die Akquise von hochwertigen, gut strukturierten und verwertbaren Daten. In realen Anwendungen, insbesondere bei komplexen Systemen und Sachverhalten, können während der Messungen Fehler und

Fehlfunktionen auftreten. Diese müssen nicht zwangsweise menschlicher Natur sein, sondern können auch bpsw. im Datenerfassungssystem entstehen. Daher müssen die aufgezeichneten Daten geprüft und ggf. bereinigt werden. Anfänglich empfiehlt sich die erneute Prüfung der Datenstruktur (D1, optional und FV) und, falls erforderlich, die Ergänzung fehlender Informationen (D2, optional und FV), wie z.B. Zielgrößen oder Metadaten. Sollten mehrere Datenquellen existieren und diese nicht oder nicht vollständig in die Datenablage eingebunden sein, müssen diese über die ID zusammengeführt werden (D3, FV). Die Checkpunkte D1-D3 dienen primär der Fehlervermeidung und sind bei gründlicher Durchführung der MuD ggf. obsolet.

Im Nachfolgenden werden die eigentlichen Schritte zur Prüfung und Bereinigung der Daten aufgelistet:

- **Identifikation blinder Flecken** (D4, optional): Eine gezielte Prüfung der Daten auf blinde Flecken kann Lücken in der Datenbasis aufzeigen, welche ggf. durch Erfahrungswissen und physikalische Zusammenhänge geschlossen werden können. Als blinde Flecken werden Bereiche bezeichnet, in denen Daten nicht aufgenommen werden bzw. nicht aufgenommen werden können (Verbesserungspotenzial Nr. 12) [25].

- **Bereinigung der Daten** (D5): Anschließend müssen die Daten von bspw. doppelten oder unvollständigen Messungen und offensichtlichen und begründbaren Fehlern bereinigt werden.

- **Anpassung von Referenzen und Nullpunkten** (D6, optional): Falls nicht bereits in der MuD geschehen, müssen Referenzen und Nullpunkte angepasst bzw. ineinander überführt werden, um die Interoperabilität der Daten zu gewährleisten (Verbesserungspotenzial Nr. 13).

- **Separierung von Referenzfahrten und Testmessungen** (D7): Wurden im Rahmen der Langzeitdatenaufnahme auch Referenzfahrten und Testmessungen aufgezeichnet, müssen diese separiert werden.

- **Entfernung von Ausreißern** (D8): Sollten während der Messung Ausreißer aufgetreten sein, müssen diese zunächst geprüft und ggf. aus den Daten entfernt werden, wenn diese physikalisch nicht plausibel sind und/oder die Verteilung der Daten signifikant beeinflussen (Verbesserungspotenzial Nr. 14).

- **Überprüfung der Einheiten** (D9): Die Einheiten der aufgezeichneten Daten müssen geprüft und, falls erforderlich, vereinheitlicht werden. Fehlerhafte,

unvollständige oder falsche Einheiten bieten ein großes Risiko in der Datenanalyse. Ursache für diese Fehlerquelle kann der Einsatz eines Messsystems sein, dessen Hersteller in einer Region mit anderem gängigen Einheitensystem ist.

- **Ausgleich von Datendrift** (D10): Schließlich sollte geprüft werden, ob die Daten driften (sich über die Zeit verschieben) und dieser ggf. ausgeglichen werden. Neben Prozessen und Betriebsmitteln können auch Sensoren driften. Eine visuelle Möglichkeit, Datendrift zu erkennen, ist z.B. das quasi-statische Signal.

Nach Abschluss der Schritte D1-D10 wird die Prüfung und Bereinigung der Daten nachvollziehbar und ggf. mit zusätzlichen Lessons Learned dokumentiert (D11).

3.4.5 Datenauswertung und Modellbildung

Das Kapitel Datenauswertung und Modellbildung (E) unterstützt den Anwender bei einer ersten Datenanalyse mittels maschinellem Lernen (Forschungsfrage 3). Dabei werden die nachfolgenden vier Themengebiete behandelt:

- **Datenvisualisierung und Datenverständnis** (Abschnitt 3.4.5.1): Im ersten Schritt werden die Daten durch verschiedene Visualisierungsmethoden dargestellt, um so ein tieferes Verständnis über die Daten aufzubauen. Sie gestaltet sich als iterativer Prozess mit der Datenprüfung und Datenbereinigung, da z.B. durch die Visualisierung der Daten Ausreißer erkannt werden können, welche anschließend bereinigt werden.

- **Auswahl von maschinellen Lernalgorithmen** (Abschnitt 3.4.5.2): In diesem Abschnitt erfolgt die Auswahl potentiell geeigneter ML-Algorithmen sowie eines geeigneten Validierungsszenarios.

- **Modellbildung** (Abschnitt 3.4.5.3): Anhand der Performanz der potentiellen Algorithmen im Validierungsszenario werden die Algorithmen für das finale Modell bestimmt und dieses trainiert.

- **Modellanwendung** (Abschnitt 3.4.5.4): Das trainierte Modell wird zur Anwendung auf das System aufgespielt und in regelmäßigen Abschnitten kontrolliert.

Neben den notwendigen Checkpunkten werden dem Anwender hilfreiche Methoden vorgestellt, die ihn bei der Datenanalyse unterstützen. Zudem wird die Verwendung der ML-Toolbox empfohlen, da diese in zahlreichen domänenübergreifenden Projekten [97,

123] sowie speziell im industriellen Kontext [8, 84, 124] nachweislich gute Ergebnisse erzielen konnte.

3.4.5.1 Datenvisualisierung und Datenverständnis

Die Visualisierung von Daten ist ein gängiges Werkzeug bei der Datenanalyse und trägt zum Aufbau von Datenverständnis bei (Verbesserungspotenzial Nr. 6). Als Grundlage für die Visualisierung muss der Anwender zunächst eine geeignete Zeitskala bzw. einen Betrachtungszeitraum definieren (E1, FV). Abhängig von dem Zeitraum der Messungen können unterschiedliche Betrachtungszeiträume existieren bzw. sinnvoll sein, wie bspw. die Betrachtung der Daten über Chargen, Schichten, Tage oder Wochen hinweg. Die nachfolgend aufgelisteten Methoden bzw. Schritte werden dem Anwender empfohlen, da diese einerseits einen hohen Praxisbezug haben und andererseits bewährte Methoden für die Auswertung industrieller Daten darstellen:

- **Darstellung von Signalverläufen im Zeitreihendiagramm** (E2): Durch Visualisierung von Signalverläufen im Zeitreihendiagramm kann der Anwender zeitliche Wirkzusammenhänge erfassen und Messungen untereinander vergleichen. Weiterhin können zusätzlich Zielgrößen im Diagramm dargestellt werden, wodurch ggf. Wirkzusammenhänge erkannt werden können.
- **Betrachtung quasi-statischer Signale** (E3, optional): Durch die Darstellung quasi-statischer Signale können z.B. Verschiebungen bzw. Drifts in den Daten periodischer Prozesse erkannt werden.
- **Darstellung von Histogrammen** (E4): Histogramme bieten dem Anwender die Möglichkeit schnell Rückschlüsse über die Verteilung zu ziehen.
- **Darstellung von Boxplot-Diagrammen** (E5, optional): Mittels Boxplot-Diagrammen können mehrere Lage- und Streumaße von Daten übersichtlich dargestellt werden. Weiterhin eignen sich Boxplot-Diagramme zum Vergleich der Lage- und Streumaße verschiedener Daten. So können die Daten bspw. nach Charge eingeteilt und verglichen werden, ob die Lage- und Streumaße mit der Charge variieren.
- **Durchführung einer Hauptkomponentenanalyse** (E6, optional): Die Hauptkomponentenanalyse kann, in Kombination mit der Einfärbung nach Stör- und Zielgrößen, dazu genutzt werden, die größten Einflüsse auf die Daten zu ermitteln und Cluster zu identifizieren.

Die Reihenfolge der Bearbeitung der aufgelisteten Schritte und Methoden kann variieren. Anhand der Ergebnisse der Visualisierung der aufgelisteten Methoden (E2–E6) kann der Anwender gezielt Fehlerbilder untersuchen (E7, optional). Dabei können auch zusätzliche Erkenntnisse gewonnen werden, indem die Visualisierungsmethoden nebeneinander betrachtet bzw. verglichen werden.

3.4.5.2 Auswahl maschineller Lernalgorithmen

Die Auswahl von geeigneten ML-Algorithmen kann insbesondere unerfahrene Anwender aufgrund des breiten Spektrums und der hohen Anzahl verfügbarer Algorithmen vor eine große Herausforderung stellen. Die in diesem Abschnitt vorgestellten Schritte dienen dem Anwender daher als Orientierung und Leitfaden bei der Auswahl geeigneter Algorithmen. Zunächst werden dem Anwender nachfolgende, vorbereitende Schritte aufgezeigt:

- **Prüfung des Stands der Technik** (E8, optional): Zunächst wird dem Anwender empfohlen, den Stand der Technik, z.B. durch eine Literaturrecherche oder die Teilnahme an Fortbildungen oder Fachkonferenzen, zu prüfen. Ggf. existieren bereits bewährte Ansätze oder aktuelle Forschungskonzepte für den ausgewählten oder einen ähnlichen Use-Case, an denen sich der Anwender orientieren kann.
- **Prüfung der verfügbaren Rechenleistung** (E9): Weiterhin muss eine Überprüfung der verfügbaren Rechenleistung und des notwendigen Speicherbedarfs erfolgen. Da unerfahrene Anwender den notwendigen Rechenaufwand zum Trainieren eines Modells leicht unterschätzen können, empfiehlt sich die Absprache mit Fachexperten.
- **Definition des Lernproblems** (E10): Anschließend erfolgt die Definition des Lernproblems. Je nach Anwendungsfall wird, speziell im industriellen Kontext, zwischen den zwei Lernproblemen Klassifikation und Regression unterschieden. Die frühzeitige Definition des Lernproblems ist entscheidend für die Auswahl der ML-Algorithmen, da sich Algorithmen ggf. nur für bestimmte Lernprobleme eignen.
- **Definition eines passenden und realistischen Validierungsszenarios** (E11): Die Wichtigkeit der Validierung bzw. die Auswahl eines passenden Validierungsszenarios kann von unerfahrenen Anwendern leicht unterschätzt werden, obwohl dieses essenziell für die Glaubwürdigkeit und Zuverlässigkeit von ML-Modellen ist. So kann durch die Auswahl eines zu trivialen und unrealistischen

Szenarios zwar eine hohe Performanz erzielt werden, jedoch kann diese durch die Überanpassung des Modells auf die Trainingsdaten entstehen und das ML-Modell verliert seine Gültigkeit in der realen Anwendung. Häufig werden die in der Praxis angewendeten Algorithmen nur unzureichend bzw. nicht realistisch validiert [136], wodurch die Generalisierbarkeit nicht ausreichend erfasst wird und deren Einsatz in der Praxis kritisch ist. Empfehlenswert ist eine Leave-One-Group-Out-Cross-Validation (LOGOCV), bei der die Gruppeneinteilung z.B. anhand einer relevanten Störgröße erfolgt [92]. Eine Übersicht über weitere gängige Validierungsszenarien findet sich in [103].

Je nach Art und Beschaffenheit der aufgezeichneten Daten können die nachfolgenden, optionalen Schritte angewandt werden, um die Performanz des Modells zu steigern:

- **Auswahl einer Datenvorverarbeitung** (E12, optional): Eine gezielte Vorverarbeitung der Daten, z.B. durch die Einbringung von Fachwissen, kann die Performanz der Algorithmen verbessern [137]. So können bspw. Beschleunigungsmessungen durch die Anwendung von Filtern auf relevante Frequenzbereiche begrenzt werden, um Rauschen und andere Störfrequenzen zu unterdrücken.

- **Auswahl von Algorithmen zur Merkmalsextraktion** (E13, optional): Insbesondere bei industriellen Daten können durch die Einbringung von Fachwissen relevante Merkmale extrahiert werden. Weiterhin kann durch die Merkmalsextraktion auch eine Dimensionsreduktion erfolgen (vgl. Abschnitt 2.6.2.2).

- **Auswahl von Algorithmen zur Merkmalsselektion** (E14, optional): Die Betrachtung von mehreren Sensoren, Prozessen oder Anlagen in einem ML-Projekt kann zu einer hohen Dimensionalität der Daten führen. Zur Vorbeugung von Überanpassung (Fluch der Dimensionalität) und der Schonung von Ressourcen können durch die Merkmalsextraktion die Daten auf die relevantesten Merkmale reduziert werden.

Die Auswahl einer geeigneten Datenvorverarbeitung und von geeigneten Algorithmen der Merkmalsextraktion und -selektion hängt stark von den Daten und ihren zugrundeliegenden physikalischen Eigenschaften bzw. Zusammenhängen der Daten ab. Hier empfiehlt es sich, die entsprechenden Fachexperten in den Auswahlprozess einzubeziehen. Der Anwender wird in diesem Kontext zudem auf die Existenz von frei verfügbaren Toolboxen, wie z.B. die vorgestellte ML-Toolbox hingewiesen, welche

verschiedene Kombinationen von komplementären Methoden der Merkmalsextraktion und -selektion abdeckt.

Entsprechend des gewählten Lernproblems erfolgt im Anschluss die Auswahl der Algorithmen zur Klassifikation bzw. Regression (E15). Das (Sonder-)Lernproblem Anomalie-Erkennung wird in Checkpunkt E16 aufgegriffen. Empfehlenswert ist zunächst die Auswahl von Algorithmen mit geringer Komplexität, wie z.B. eine Lineare Diskriminanzanalyse (LDA) in Kombination mit einem k-nächste-Nachbarn-Klassifikator bei Klassifikationsproblemen. Nichtlineare Verfahren wie künstliche neuronale Netze benötigen in der Regel eine große Datenbasis und sind meist Black-Box-Methoden, bei denen das Ergebnis bzw. die Entscheidungsfindung kaum interpretierbar und nachvollziehbar ist (vgl. Abschnitt 2.6.2 und Abschnitt 2.1, Herausforderungen A1-A2 nach Wilhelm in Tabelle 2.1). Daher ist im Rahmen der Checkliste die Anwendung neuronaler Netze für unerfahrene Anwender nicht empfohlen.

Aufgrund von komplexen Prozessen und Anlagen (vgl. Herausforderungen P1-P5 nach Wilhelm in Tabelle 2.1), Wechselwirkungen der Störgrößen, Interessenkonflikten (Daten als Mittel zum Zweck) und aus Gründen der Wirtschaftlichkeit sind oftmals nicht alle Störgrößen und Fehlerszenarien in hinreichendem Maße in den Daten abgebildet. Daher empfiehlt sich die Implementierung einer zusätzlichen Anomalie-Erkennung (E16, optional). Für die Anwendung der Anomalieerkennung (Abschnitt 2.6.2.4) wird zunächst der Normalzustand bzw. -betrieb einer Anlage oder eines Prozesses erfasst und ein Grenzwert festgelegt. Sobald der Grenzwert überschritten und somit eine Anomalie festgestellt wird, kann bspw. eine Warnmeldung an den Arbeiter erfolgen. Bei Anomalien muss es sich nicht zwangsweise um Beschädigungen handeln, sondern diese können auch durch Störgrößen wie Temperaturschwankungen o.ä. ausgelöst werden. Handelt es sich bei den Anomalien um Beschädigungen, können diese entsprechend annotiert und im nächsten Training des ML-Modells inkludiert werden [138]. Werden keine Beschädigungen festgestellt, kann der Grenzwert des Modells angepasst werden.

3.4.5.3 Modellbildung

Nachdem geeignete ML-Algorithmen, sowie ein geeignetes Validierungsszenario ausgewählt wurden, wird das ML-Modell gebildet. Hierzu werden zunächst anhand ausgewählter Algorithmen ML-Modelle erzeugt. Anhand der aufgezeichneten Daten und dem gewählten Validierungsszenario wird anschließend die Performanz ermittelt und verglichen (E17). Bei Klassifikationsproblemen kann hierzu z.B. der Prozentsatz korrekt klassifizierter Validierungs- oder Testdaten verwendet werden und bei

Regressionsproblemen der mittlere quadratische Fehler von Validierungs- oder Testdaten [139]. Entsprechend der Performanz der Modelle werden die Algorithmen für die Modellbildung ausgewählt (E18). Sollte die Performanz nicht den im Kapitel *Vorbereitung und Projektplanung* festgelegten Anforderungen entsprechen, können als erste Maßnahme komplexere Algorithmen in Betracht gezogen werden (Schritt E13-E15). Erzielen die weiteren Algorithmen keine Steigerung der Performanz, wird dem Anwender empfohlen, die Datenqualität zu prüfen und die Datenmenge zu erhöhen, um so komplexere Algorithmen anwenden zu können.

Ist die erzielte Performanz ausreichend, muss der Anwender die Unsicherheit der Vorhersage des ML-Modells bestimmen (E19, Verbesserungspotenzial Nr. 15 und 16). Die Fortpflanzung der Unsicherheit der Vorhersage kann automatisiert mittels der UA-ML-Toolbox erfolgen, oder manuell z.B. mittels Guide to the Expression of Uncertainty in Measurement (GUM) [55]. Entspricht die resultierende Performanz inklusive Betrachtung der Messunsicherheit ebenfalls den Anforderungen, wird das finale Modell mit allen Daten trainiert. Das trainierte Modell kann anschließend zur Anwendung auf das System aufgespielt werden (E20, optional und FV).

Essenziell ist die anfängliche Kontrolle des ML-Modells in der Anwendung (E21). Wurde das ML-Modell z.B. mit einem ungeeigneten Validierungsszenario gebildet, kann eine Überanpassung eintreten und das ML-Modell im Normalbetrieb ungültig sein. Dann muss die Modellbildung angepasst oder wiederholt werden. Ist das ML-Modell gültig, startet die eigentliche Modellanwendung.

3.4.5.4 Modellanwendung

Während der Anwendung des ML-Modells muss eine regelmäßige Gültigkeitsprüfung eingeplant werden (E22). In dieser wird überprüft, ob die Sensoren sinnvolle und plausible Daten aufzeichnen und das ML-Modell weiterhin gültig ist. Im Gegensatz zu einer ggf. vorgesehenen Maschinenfähigkeitsuntersuchung wird hierbei gezielt auf lokale Veränderungen im Betrieb geachtet. Außerdem muss das regelmäßige Nachtrainieren des ML-Modells eingeplant werden (E23), um so auch aktuelle Daten im Trainingsprozess zu inkludieren und somit ggf. Drift-Effekte auszugleichen. Jegliche Hardware- und Software-Änderungen müssen dokumentiert werden (E24). Insbesondere bei komplexen Anlagen oder Prozessen bzw. ML-Modellen können selbst kleine Änderungen an Hardware und Software die Datenmuster signifikant beeinflussen und ein Neutrainieren des ML-Modells erforderlich machen. Bei Änderungen am ML-Modell empfiehlt sich eine zusätzliche Versionierung. So können auch im Nachgang Änderungen an den Modellen nachvollzogen werden.

3.4.6 Projektabschluss

Der Projektabschluss ist ein nicht zu vernachlässigender Schritt in der Bearbeitung eines ML-Projektes. Von den gesammelten Erfahrungen und Erkenntnissen können zukünftige (ML-)Projekte profitieren, unabhängig vom Erfolg des Projektes. Auch aus fehlgeschlagenen ML-Projekten können wertvolle Erkenntnisse gewonnen werden, insbesondere wenn die Gründe für den Fehlschlag ermittelt wurden.

Im ersten Schritt des Projektabschlusses wird dem Anwender empfohlen, die Projektergebnisse mit dem ursprünglichen Ziel des ML-Projektes abzugleichen und Abweichungen zu dokumentieren (F1, optional und Verbesserungspotenzial Nr. 17). Üblicherweise treten in einem Projekt Fehler und Probleme auf. Daher wird dem Anwender empfohlen, Lessons Learned zu formulieren (F2, optional). Durch die formulierten Lessons Learned können diese Fehler und Probleme in zukünftigen Projekten vermieden werden. Weiterhin muss eine kurze und übersichtliche Abschlusspräsentation erstellt werden (F3). In dieser müssen die Ziele, die Vorgehensweise und die Erfolge bzw. Grenzen des ML-Projektes klar benannt werden, wodurch das Projekt z.B. durch Dritte schnell erfasst und nachvollzogen werden kann. Eine detaillierte Dokumentation erfolgt in einem Abschlussbericht (F4). Dieser sollte so umfassend wie nötig, aber so kurz wie möglich sein.

Gemäß den FAIR-Prinzipien müssen alle Unterlagen, Dokumente, Programme und ggf. Daten auffindbar, zugänglich, interoperabel und wiederverwendbar abgelegt werden (F5).

3.4.7 Abschließende Kapitel

Die Checkliste umfasst weiterhin die folgenden abschließenden Kapitel:

- **Fazit**: Im Fazit wird der Anwender darauf aufmerksam gemacht, dass die Durchführung eines ML-Projektes ein zeitaufwendiger und iterativer Prozess ist, welcher oftmals nicht zu einem perfekten Ergebnis führt, sondern stetig verbessert werden kann. Weiterhin werden die Grenzen der Checkliste aufgeführt und aufgezeigt, dass diese keine alleinstehende Musterlösung ist, sondern als Hilfestellung für unerfahrene Anwender dient (Verbesserungspotenzial Nr. 7).
- **Abkürzungsverzeichnis**: Das Abkürzungsverzeichnis enthält allen verwendeten Abkürzungen.
- **Glossar**: Im Glossar werden zum Verständnis der Checkliste essenzielle Begrifflichkeiten erklärt.

- **Literaturverzeichnis**: Das Literaturverzeichnis enthält Verweise auf weiterführende Literatur und bietet dem Anwender einen niedrigschwelligen Einstieg in die jeweiligen Themengebiete. Die Verständlichkeit und Zugänglichkeit hatten bei der Auswahl der Literatur Priorität vor dem wissenschaftlichem Tiefgang.
- **Übersicht der Sensorik**: Dem Anwender wird eine beispielhafte Vorlage zur Gestaltung einer Sensorübersicht präsentiert. Diese enthält wichtige Informationen über die Sensoren und bildet für den Anwender eine gute Ausgangsbasis.

3.5 Ablaufplan zur Durchführung von KI-Projekten im Mittelstand

Die vorgestellte Checkliste kann, aufgrund des schriftlichen Formats für Anwender den Eindruck vermitteln, dass die jeweiligen Checkpunkte sequenziell und einmalig abgearbeitet werden müssen. Daher wurde der Ablaufplan zur Durchführung von KI-Projekten im Mittelstand (im Folgenden Ablaufplan) als komplementäres Dokument zur Checkliste entwickelt. Dieser dient dem Anwender zur schnellen Orientierung innerhalb des ML-Projektes und ermöglicht eine visuelle Darstellung von Zusammenhängen wie z.B. iterativen Prozessen oder Schleifen. Der Ablaufplan ist analog zu den Checklisten-Kapiteln A-F in sechs Abschnitte unterteilt. Jeder Abschnitt entspricht somit einem Kapitel der Checkliste und hebt sich zur besseren Unterscheidung farblich von den anderen Abschnitten ab.

Aufgrund der hohen Anzahl einzelner Checkpunkte wurden diese zugunsten der Übersichtlichkeit gruppiert und lediglich die relevantesten Punkte der Checkliste hervorgehoben. Darüber hinaus werden bei den jeweiligen Checkpunkten bzw. Checkpunktabschnitten hilfreiche Personen (Fachexperte und Dateningenieur) durch ein entsprechendes Symbol indiziert.

Der Ablaufplan orientiert sich an dem grundlegenden Aufbau des Ablaufplans für einen ganzheitlichen, wissensgetriebenen ML-Prozess in der Produktion [127]. Im Rahmen dieser Dissertation wurde er optisch und inhaltlich grundlegend überarbeitet und auf die entwickelte Checkliste angepasst (Verbesserungspotenzial Nr. 18). Abbildung 3.4 zeigt den so resultierenden Ablaufplan.

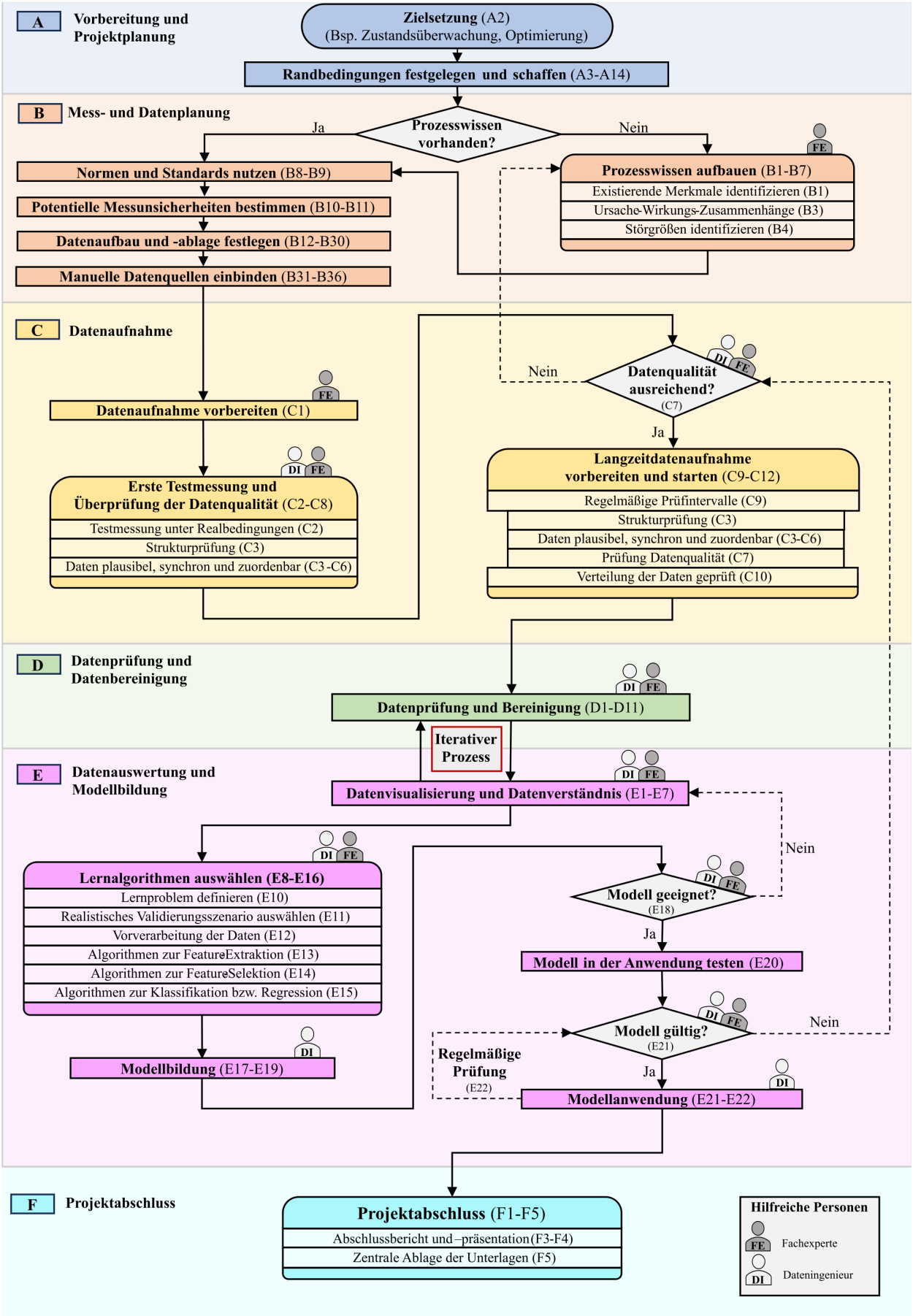


Abbildung 3.4: Ablaufplan zur Durchführung von KI-Projekten im Mittelstand [129].

3.6 Diskussion und Zwischenfazit

Die Checkliste *KI-Projekte erfolgreich umsetzen: Eine Checkliste für den Mittelstand* wurde als Unterstützung für unerfahrene Anwender mittelständischer Unternehmen entwickelt. Dabei bietet die Checkliste Anwendern einen kompakten Überblick über die Themengebiete eines ML-Projektes und zeigt deren wesentlichen Schritte in Form von Checkpunkten auf. Hier musste ein Trade-off zwischen Vollständigkeit und Handhabbarkeit bzw. Nutzerfreundlichkeit eingegangen werden. Um den reduzierten wissenschaftlichen Tiefgang auszugleichen, wird innerhalb der Checkliste auf weiterführende Literatur verwiesen.

Hinsichtlich der initial gestellten Forschungsfragen kann folgendes Zwischenfazit gezogen werden:

- **Forschungsfrage 1**: Innerhalb der Checkliste wurden die Forschungskonzepte CRISP-DM, UWD, FAIR-Prinzipien, Messunsicherheiten und Methoden zur Bewertung der Datenqualität in den Ablauf eines ML-Projektes integriert. So verwendet der Anwender die Konzepte bei der Bearbeitung der Checkliste automatisch.

- **Forschungsfrage 2**: Um unerfahrene industrielle Anwender dazu zu befähigen, hochwertige Daten aufzuzeichnen, wird die MuD als essenzielle Grundlage innerhalb der Checkliste ausführlich behandelt. Hier werden dem Anwender die notwendigen Schritte aufgezeigt, sowie Metriken zur Bewertung der Datenqualität vorgeschlagen.

- **Forschungsfrage 3**: Die Checkliste unterstützt Anwender bei der Durchführung einer Datenanalyse mittels ML, indem alle notwendigen Schritte von der Datenvisualisierung hin zur Algorithmenauswahl, Modellbildung und Validierung aufgegriffen und behandelt werden. Zudem unterstützen weiterführende Literatur, zusätzliche Hinweise und Tipps den Anwender. Die empfohlene ML-Toolbox bietet dem Anwender weiterhin die Möglichkeit, eine erste Datenanalyse mit ML durchzuführen, ohne dabei tiefgreifende Kenntnisse über die Algorithmen zu besitzen.

Obwohl die Checkliste vorrangig für bereits bestehende Maschinen und Anlagen (Brownfield) entwickelt wurde, kann sie auch als Orientierung für die Neuanschaffung von Maschinen und Anlagen (Greenfield) genutzt werden. Durch ihren allgemeingültigen Aufbau kann sie zudem auch Anwender aus Unternehmen unterstützen, welche nicht dem Mittelstand zuzuordnen sind.

4 PIA - Konzept eines persönlichen Informationsassistenten

Kapitel 4 befasst sich mit der Konzept eines persönlichen Informationsassistenten (PIA), welcher den Anwender ganzheitlich bei der Bearbeitung eines Datenanalyseprojektes mittels maschinellem Lernen (ML) unterstützt und dabei die vorgestellte Checkliste als wesentliches Element integriert.

Im Rahmen der Dissertation wurde PIA in der Publikation *PIA - A Concept for a Personal Information Assistant for Data Analysis and Machine Learning of Time-Continuous Data in Industrial Applications* [122] veröffentlicht.

4.1 Motivation und Anforderungen

Neben dem Bedarf einer sorgfältigen Mess- und Datenplanung wurde auch die Notwendigkeit einer digitalen, ganzheitlichen Unterstützung des Anwenders bei der Durchführung eines ML-Projektes im Projekt Messtechnisch gestützte Montage (MessMo) identifiziert. In diesem Kontext wurden in Zusammenarbeit mit industriellen Partnern folgende Bedarfe formuliert:

- **B-1**: Unterstützung des Nutzers bei der Durchführung eines ML-Projektes.
- **B-2**: Unterstützung des Nutzers bei der Aufzeichnung von Daten.
- **B-3**: Zugänglichkeit zu domänenspezifischem (Experten-)Wissen.
- **B-4**: Zugänglichkeit zu allen notwendigen Daten und Metadaten.
- **B-5**: Bereitstellung eines Tools zur Datenanalyse.

Diese Bedarfe decken sich mit den in Tabelle 3.1 ermittelten Bedarfen der IDG-Studie [3]. Zusätzlich ergab eine Untersuchung [1] mit 49 Unternehmen im Jahr 2021, dass 19 % der Unternehmen fehlendes technisches Wissen als Hemmnisse bei der Datennutzung sehen.

Um einen Lösungsansatz für diese Herausforderungen zu erstellen, wurde im Rahmen des Projektes *iTecPro – Erforschung und Entwicklung von innovativen Prozessen und Technologien für die Produktion der Zukunft* das Konzept eines persönlichen Informationsassistenten (PIA) entwickelt.

4.2 Konzeption und Aufbau

Das Konzept PIA stellt eine Software dar, welche Mitarbeiter im Mittelstand ganzheitlich bei der Durchführung von ML-Projekten unterstützt. Anhand der ermittelten Bedarfe B-1 bis B-5 wurden für PIA drei grundlegende Module definiert:

- **M1 - Zugänglichkeit von Daten und Wissen** (Abschnitt 4.2.1): Durch einen erleichterten Zugriff auf Daten und Wissen mittels PIA werden in M1 die Bedarfe B-3 und B-4 abgedeckt.
- **M2 - Unterstützung bzw. Begleitung des Anwenders bei der Bearbeitung eines ML-Projektes** (Abschnitt 4.2.2): Die Integration der in Kapitel 3 vorgestellten Checkliste als digitales Element in PIA deckt die Bedarfe B-1 und B-2 ab. Die Implementierung der Checkliste in PIA erhöht zudem ihre Nutzerfreundlichkeit und ist die logische Konsequenz ihrer Anwendung im Rahmen von Industrie 4.0 (I4.0).
- **M3 - Unterstützung bzw. Begleitung bei der Datenanalyse** (Abschnitt 4.2.3): Zur Unterstützung bei der Datenanalyse wird eine ML-Toolbox in PIA integriert. In Kombination mit der Checkliste aus M2 kann so Bedarf B-5 abgedeckt werden.

Abbildung 4.1 visualisiert das Konzept von PIA und zeigt zudem, welche Module bei den jeweiligen Schritten eines ML-Projektes unterstützen.

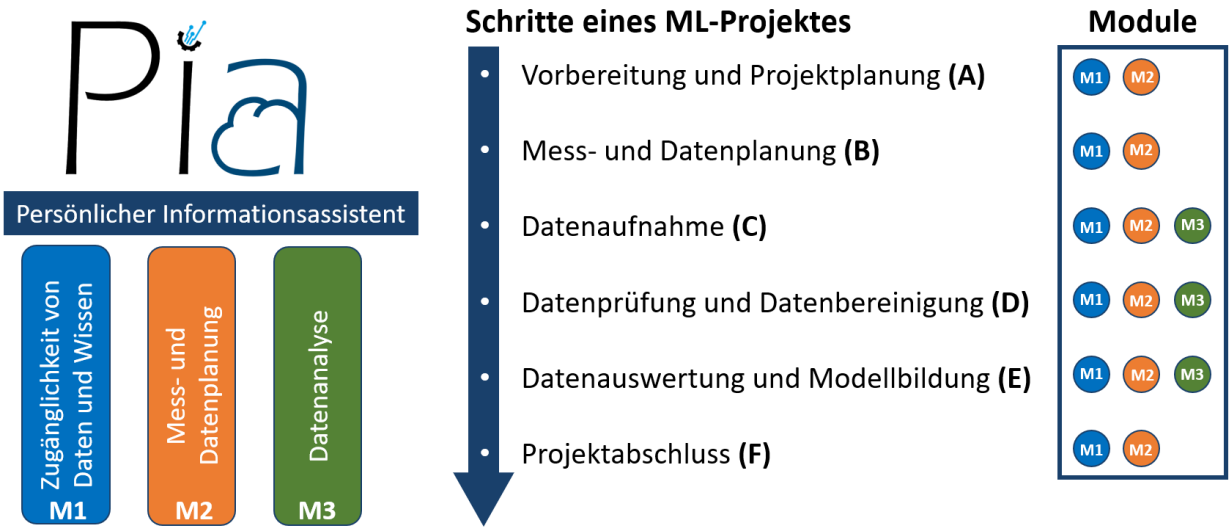


Abbildung 4.1: Konzept PIA mit seinen drei Modulen (M1-M3) und deren Beteiligung an den jeweiligen Schritten eines ML-Projektes, adaptiert von [122].

Durch die Verknüpfung von Daten und Wissen kann die FAIRness der Daten gesteigert werden. Weiterhin können in PIA zusätzliche Informationen bspw. über das Ursache-Wirkungs-Diagramm (UWD) oder Messunsicherheiten niederschwellig zur Verfügung gestellt werden. Dies trägt u.a. dazu bei, dass Konzepte der Forschung in die Industrie übertragen werden (Forschungsfrage 1). Die Integration der Checkliste und deren Verknüpfung mit einer Wissensdatenbank befähigt industrielle Anwender, hochwertige Daten aufzuzeichnen (Forschungsfrage 2). Weiterhin können durch Integration einer ML-Toolbox industrielle Anwender auch bei der Datenanalyse unterstützt werden (Forschungsfrage 3).

Bereits bestehende Ansätze oder Lösungen in Unternehmen, wie z.B. ein Wissensmanagementsystem (Teilbereich von M1) oder Tools für die Datenanalyse (M3), können mit den vorgeschlagenen Ansätzen ausgetauscht oder ergänzt werden. Dies bietet Unternehmen die Möglichkeit, trotz bereits bestehender Systeme PIA in ihr Unternehmen zu integrieren.

Im Nachfolgenden werden die jeweiligen Module näher erläutert. Zur Unterstützung von mittelständischen Unternehmen, welche ggf. keinerlei bestehende Systeme besitzen, werden zudem konkrete Vorschläge für die Gestaltung der Module gegeben.

4.2.1 Modul 1: Zugänglichkeit von Daten und Wissen

In Modul 1 liegt der Fokus auf der Zugänglichkeit von Daten, da Formen des Wissensmanagements in der Industrie bereits verbreitet sind. Dennoch wird das

Konzept der Lessons Learned als ergänzende Methode für zugängliches Wissen aufgegriffen.

Die Zugänglichkeit von industriellen Daten ist aufgrund diverser Herausforderungen, wie bspw. fehlender moderner Datenschnittstellen bei Bestandsanlagen oder der komplexen Datenfusion multimodaler Datenquellen, oftmals erschwert [11]. Im Sinne der FAIR-Prinzipien sollten Daten jedoch auffindbar, zugänglich, interoperabel und wiederverwendbar sein. Durch die Anwendung der Checkliste (M2) wird sichergestellt, dass die Daten einerseits den FAIR-Prinzipien entsprechen und andererseits bereits in einer strukturierten Form in einer Datenbank hinterlegt wurden. Um die Zugänglichkeit dieser Daten und Metadaten in PIA zu gewährleisten, muss die Datenbank entsprechend mit PIA verknüpft werden.

Bei der Analyse von Daten und deren Wirkzusammenhängen spielen Wissen und Berufserfahrung, insbesondere bei komplexen Anlagen, Produkten und Betriebsmitteln, eine entscheidende Rolle. Dieses (Experten-)Wissen ist oft implizit und somit schwer zugänglich und erfordert gezielte Maßnahmen zu dessen Erfassung. Abhilfe kann ein Wissensmanagement-System schaffen, in welchem spezifisches Wissen, z.B. über Anlagen, Produkte und Betriebsmittel, durch Nutzer systematisch und strukturiert abgelegt werden kann. Da viele Unternehmen bereits über ein Wissensmanagementsystem oder eine Vorstufe davon verfügen, werden für den PIA keine spezifischen Vorgaben abgeleitet. Ein bereits bestehendes Wissensmanagement-System kann und sollte in PIA integriert werden. In Abschnitt 4.3 wird die beispielhafte Integration eines Wissensmanagementsystems in PIA dargestellt.

Zusätzlich empfiehlt sich die Integration von Lessons Learned, um zukünftige Projekte mit den Erfahrungen vergangener Projekte zu verbessern. Abbildung 4.2 zeigt das Vorgehen bei der Erfassung einer Lesson Learned und deren Weiterentwicklung zu einer High-Quality Lesson Learned.

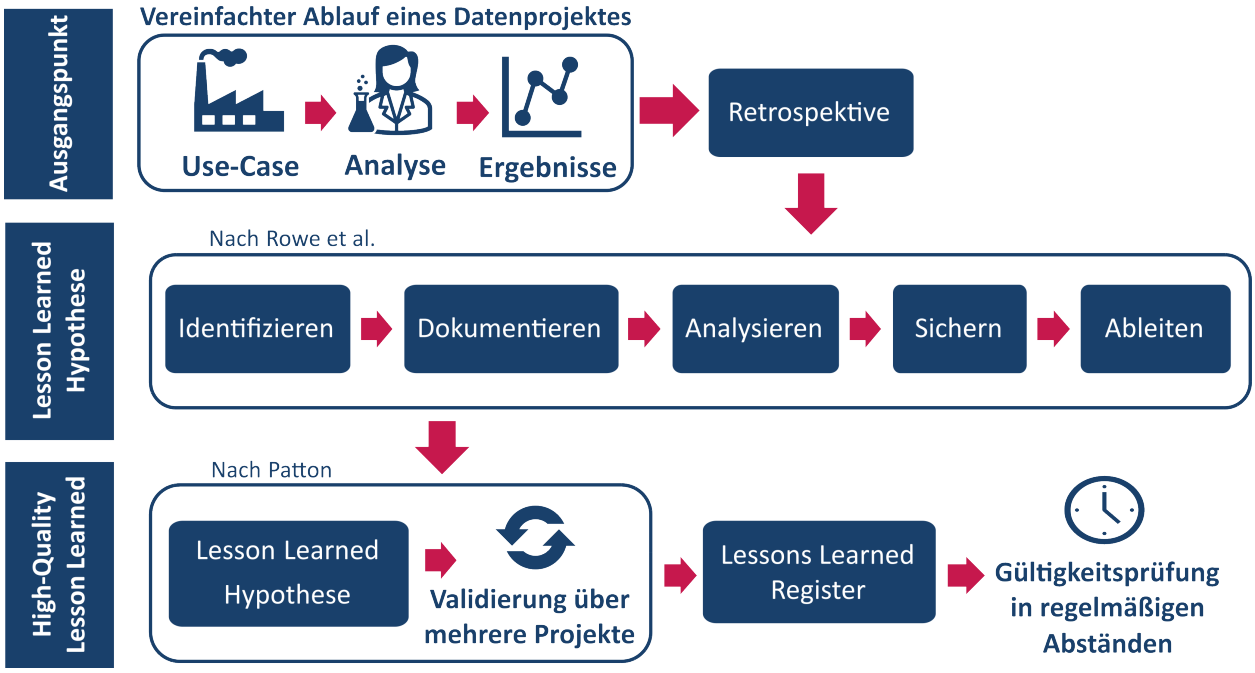


Abbildung 4.2: Vorgehen bei der Erstellung einer High-Quality Lesson Learned, adaptiert von [122].

Ausgangspunkt bildet ein klassisches Daten- oder ML-Projekt, welches in der Retrospektive auf Verbesserungspotential untersucht wird. Dabei erfolgt im ersten Schritt die Formulierung einer Lesson Learned Hypothese mittels der fünf Schritte identifizieren, dokumentieren, analysieren, sichern und ableiten [44]. Jede Lesson Learned Hypothese ist anfangs nur für das bereits abgeschlossene Projekt gültig.

Behält diese Lesson Learned Hypothese über mehrere Projekte hinweg ihre Gültigkeit, kann diese als High-Quality Lesson Learned angesehen werden [43]. In regelmäßigen Abschnitten wird die Gültigkeit der Lesson Learned überprüft, sodass lediglich gültige Lessons Learned im Register enthalten sind. Ein solches Lessons Learned Register wurde beispielhaft in PIA implementiert (siehe Abschnitt 4.3).

Somit gewährleistet Modul 1 die Zugänglichkeit von Daten durch die Verknüpfung einer Datenbank mit hochqualitativen Daten und Metadaten (M2). Die Zugänglichkeit von Wissen wird durch die Verknüpfung des PIA mit einem Wissensmanagementsystem und die zusätzliche Integration eines Lessons Learned Registers sichergestellt.

4.2.2 Modul 2: Unterstützung des Anwender

Daten bilden die Grundlage eines ML-Projektes und ein ML-Algorithmus kann nur eine ausreichende Robustheit aufweisen, wenn die Messdaten alle zugrundeliegenden physikalischen oder chemischen Effekte der relevanten Stör- und Steuergrößen in angemessenem Umfang enthalten. In Kapitel 3 wurde eine Checkliste vorgestellt, die

den Anwender ganzheitlich bei der Durchführung eines ML-Projektes und insbesondere bei der Mess- und Datenplanung (MuD) unterstützt. Obwohl sich diese bereits durch Übersichtlichkeit und Nutzerfreundlichkeit auszeichnet, ist sie primär eine alleinstehende Lösung. Durch die Integration der Checkliste in PIA kann diese niederschwellig genutzt werden und es ergeben sich u.a. folgende Vorteile:

- **Einheitliches und ganzheitliches System**: Die Checkliste befindet sich in der gleichen Umgebung wie das (Experten-)Wissen und die Daten (M1), sowie die Werkzeuge der Datenanalyse (M3). Somit kann der Anwender effizient, insbesondere bei iterativen Prozessen der Datenaufzeichnung (Forschungsfrage 2) und der Datenanalyse (Forschungsfrage 3) unterstützt werden.
- **Gesteigerte Zugänglichkeit**: Innerhalb von PIA ist die Checkliste für Nutzer leicht zugänglich. Weiterhin bietet die digitale Einbindung, bpsw. bei kurzfristigem Ausfall des bearbeitenden Mitarbeiters, Kollegen die Möglichkeit, den Projektfortschritt nachzuvollziehen und dieses fortzusetzen.
- **Integration relevanter Daten**: Innerhalb von PIA können Wissen, Daten und Metadaten direkt mit den jeweiligen Checkpunkten verknüpft werden. So kann z.B. ein erstelltes UWD (Checkpunkt B2) direkt beim Checkpunkt hinterlegt werden und ist auch nach Projektende auffindbar und zugänglich.
- **Verknüpfung relevanter Projekte**: Innerhalb von PIA können, z.B. anhand von Schlagwörtern, ähnliche, bereits existierende Projekte miteinander verknüpft werden. Dies gilt nicht nur für abgeschlossene Projekte, sondern ggf. auch für parallel laufende Projekte, bei denen so Synergien genutzt werden können.

Sollte bereits eine interne Strategie zur Abwicklung von ML-Projekten im Unternehmen existieren, kann auch diese statt der Checkliste verwendet werden. Empfehlenswert ist jedoch die Anpassung der Checkliste auf die spezifischen Bedürfnisse des Unternehmens und die Integration bereits existierender Richtlinien bzw. von Normen und Standards.

4.2.3 Modul 3: Datenanalyse

Die Analyse industrieller Daten ist oftmals spezifisch, kann auf unterschiedlichste Arten erfolgen und hängt vom Anwendungsfall, den zur Verfügung stehenden Ressourcen und dem Wissen bzw. der Erfahrung des Analysten ab. Um Anwender ohne Vorkenntnisse bei der Datenanalyse zu unterstützen (Forschungsfrage 3), ist eine strukturierte

Vorgehensweise hilfreich. Hierfür bieten sich die Kapitel D (Datenprüfung und Datenbereinigung) und E (Datenauswertung und Modellbildung) der in M2 integrierten Checkliste an.

Weiterhin kann die Vielzahl der verfügbaren Algorithmen für eine KI- bzw. ML-gestützte Datenanalyse unerfahrene Anwender vor eine Herausforderung stellen. Oftmals sind verfügbare Algorithmen, speziell aus dem Forschungskontext, spezifisch auf einzelne Anwendungsfälle zugeschnitten und nicht ausreichend getestet [140, 141]. Daher wird deren Anwendung für unerfahrene Anwender nicht empfohlen. Ein möglicher Lösungsansatz sind Toolboxen, welche ein Spektrum verschiedener Algorithmen abdecken und bereits in industriellen Applikationen verwendet wurden. Die vorgestellte Automatisierte ML-Toolbox für zyklische Daten (ML-Toolbox) kann durch ihre komplementären Algorithmen ein breites Spektrum zeitdiskreter Daten abdecken. Zudem ist sie durch ihre Automatisierbarkeit und Low-Code Anwendbarkeit auch für unerfahrene Anwender geeignet. Erfahrenere Anwender können darüber hinaus Parameter optimieren oder auch Algorithmen ergänzen oder austauschen.

Somit bildet die ML-Toolbox eine geeignete Grundlage für die Datenanalyse in M3 und kann in Kombination mit der Checkliste unerfahrene Anwender dazu befähigen, eine erste Datenanalyse durchzuführen (Forschungsfrage 3).

4.3 Implementierung eines Softwaredemonstrators

Um die Vorteile von PIA zu veranschaulichen und dessen Nutzen in den Anwendungsfällen (Kapitel 5) zu evaluieren, ist die Erstellung eines Softwaredemonstrators erforderlich. Nachfolgend wird ein möglicher Aufbau von PIA vorgestellt. Anschließend erfolgt in Abschnitt 4.3.2 die Konzeption und Umsetzung einer grafischen Benutzeroberfläche für den Demonstrator.

4.3.1 Aufbau und Struktur

Um die Zugänglichkeit von PIA zu gewährleisten, ist eine Implementierung als Webanwendung auf einem firmeninternen Server empfehlenswert. Dies ermöglicht einen einfachen und ortsunabhängigen Zugriff im Intranet von PCs, Tablets und mobilen Endgeräten. Für den Softwaredemonstrator wurde daher mit Hilfe des Hypervisors VirtualBox (Oracle Corporation) [142] ein virtueller Server mit dem Betriebssystem Ubuntu 22.04.4 LTS aufgesetzt, um einen Server zu simulieren. Bei Ubuntu handelt es sich um eine leistungsfähige und flexible Linux-Distribution, welche für den Einsatz

auf Servern geeignet ist [143]. Alternativ kann PIA auch auf anderen Betriebssystemen, wie z.B. Microsoft Windows oder Apple macOS aufgesetzt werden.

Nachfolgend wird die Programmierung und Implementierung der grafischen Benutzeroberfläche von PIA näher betrachtet. Die Webanwendung für PIA wurde unter Anwendung des weit verbreiteten open-source Frameworks Angular 13.3.4 [144] entwickelt. Dieser besitzt eine hohe Modularität und ermöglicht eine effiziente und schnelle Entwicklung des Softwaredemonstrators. Eine Übersicht der genutzten Umgebungen und Bibliotheken mit ihrer Bezugsquelle ist in Tabelle 4.1 gelistet.

Tabelle 4.1: Übersicht der in PIA verwendeten Pakte und Bibliotheken [122].

ID	**Paket (E)**	**Quelle**
E1	Angular CLI	`https://www.angular.io/`
E2	Node JS	`https://www.nodejs.org/en/`
E3	Node Package Manager	`https://www.npmjs.com/`
ID	**Bibliothek (L)**	**Quelle**
L1	Angular Forms	`https://www.npmjs.com/package/@angular/forms`
L2	Angular Material	`https://www.material.angular.io/`
L3	Bootstrap	`https://www.npmjs.com/package/bootstrap`
L4	Charts js	`https://www.npmjs.com/package/chart.js`
L5	Flex Layout	`https://www.npmjs.com/package/flex-layout`

Aufgrund des hierarchischen Aufbaus von Angular wurden die Funktionalitäten von PIA in einzelne Komponenten (K) aufgeteilt und separat programmiert. Somit können Komponenten modular wiederverwendet und neue Komponenten einfach hinzugefügt werden. Die so entwickelten Komponenten werden nachfolgend aufgelistet:

- **App Komponente** (K1): Die *App Komponente* der Anwendung wird in der Datei app.module.ts definiert und, zum Start der Anwendung, zu main.ts gebootstrapped. K1 dient als Container für alle anderen Komponenten der Anwendung.
- **Menü** (K2): Die Komponente *Menü* erlaubt mittels Buttons die Navigation durch PIA.
- **Anlage** (K3): Die Komponente *Anlage* enthält die Struktur einer angelegten Anlage, idealerweise der digitale Zwilling [145] der Anlage, und erlaubt die Navigation durch die eingebetteten Komponenten mittels Buttons.

- **Anlagen Details** (K4): Die Komponente *Anlagen Details* enthält die Informationen über eine angelegte Anlage oder Maschine in Form eines Arrays von Objekten. Jedes Objekt enthält Eigenschaften, welche die Anlage oder Maschine, deren Prozesse und ggf. deren Stationen beschreiben.

- **Checkliste** (K5): Die Komponente *Checkliste* stellt die Implementierung der Checkliste (M2) in PIA dar. Sie erlaubt die Navigation durch die jeweiligen Checkpunkte und deren digitales Abhaken. Weiterhin sind die entsprechenden Zusatzbeschreibungen, Tipps, Hinweise und relevanten Abschnitte des Ablaufplans in ihr enthalten. Außerdem enthält die Komponente eine Funktion, um den Checkpunkten Kommentare, Lessons Learned oder Dateien hinzuzufügen.

- **Graph** (K6): Die Komponente *Graph* erlaubt das Plotten von Graphen mittels der Chart.js-Bibliothek (L4). Dies erlaubt dem Anwender, z.B. eine erste Visualisierung von Daten.

- **Wissensablage** (K7): Die Komponente *Wissensablage* enthält das Lessons Learned Register und ermöglicht die Implementierung einer Baumansicht für die Darstellung von domänenspezifischem Wissen über bpsw. Anlagen, Prozesse oder Betriebsmittel.

- **Info** (K8): Die Komponente *Info* ist die Implementierung einer sog. Karte, welche spezifische Informationen über bspw. Prozesse innerhalb einer Anlage enthält.

- **Navigation** (K9): Die Komponente *Navigation* entspricht dem Header von PIA, welcher das Logo und den Namen der Anwendung enthält.

Abbildung 4.3 zeigt einen zusammenfassenden Überblick über den Aufbau und die Struktur des Softwaredemonstrators für den PIA. Die Umsetzung in mittelständischen Unternehmen erfordert ggf. Anpassungen hinsichtlich der IT-Infrastruktur und Softwarebereitstellung.

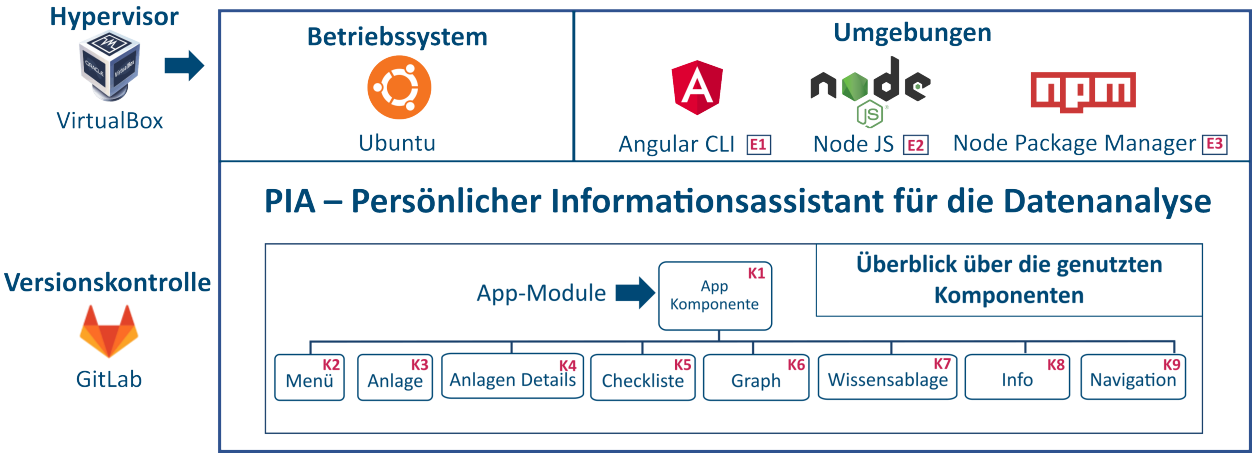


Abbildung 4.3: Aufbau und verwendete Umgebungen in PIA, adaptiert von [122].

4.3.2 Grafische Benutzeroberfläche

Die Implementierung einer grafische Benutzeroberfläche (GUI) erhöht die Anwenderfreundlichkeit von PIA. Abbildung 4.4 zeigt die Startseite der GUI mit dem Logo im Anzeigefeld (Abbildung 4.4, blau) und der Menüleiste (Abbildung 4.4, rot) mit den vier Menüpunkten:

- **Anwendungsfälle**: In diesem Menüpunkt befinden sich alle relevanten Daten und Metadaten des ML-Projektes nach Anwendungsfällen gruppiert.
- **Wissensdatenbank**: In der Wissensdatenbank befindet sich das Lessons Learned Register und das Expertenwissen der Anwendungsfälle.
- **Checkliste**: Dieser Menüpunkt enthält die digitale Version der Checkliste und des Ablaufplans. Jedem Anwendungsfall wird eine eigene Checkliste zugeordnet, in welcher der aktuelle Bearbeitungsstand mit Kommentaren und Anhängen abgerufen werden kann.
- **Datenanalyse**: Dieser Menüpunkt enthält die ML-Toolbox zur Datenanalyse in den Anwendungsfällen (vgl. Modul 3 in Abschnitt 4.3.2.3).

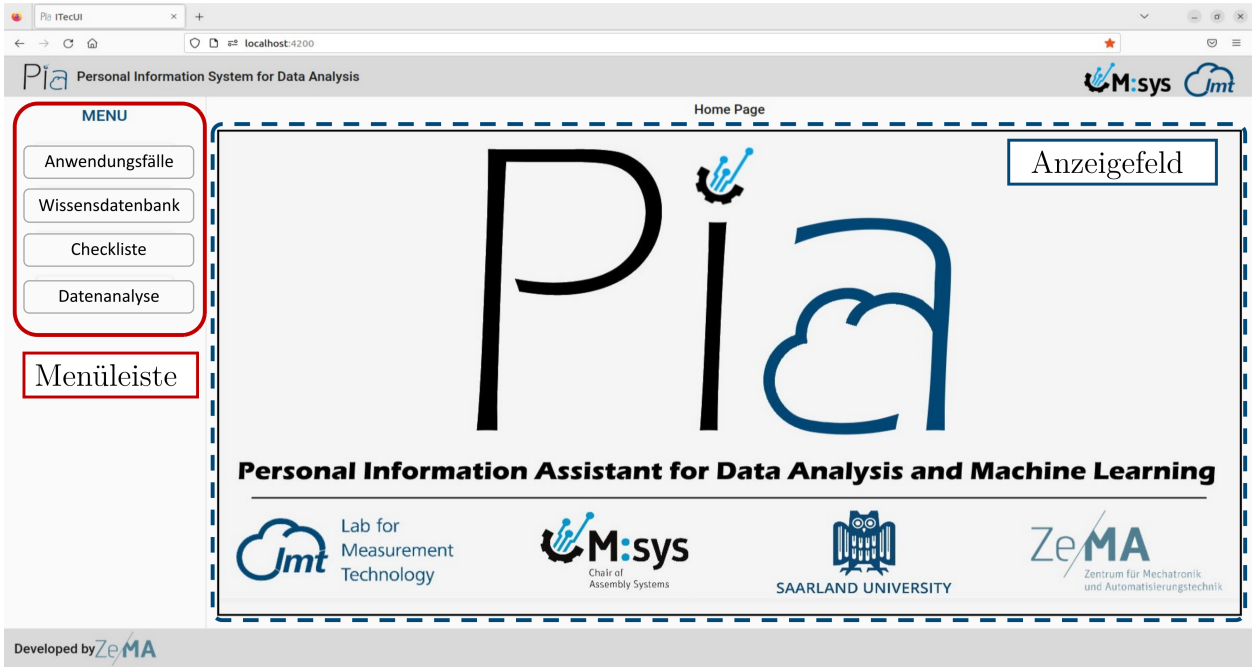


Abbildung 4.4: Startseite von PIA, mit Anzeigefeld (blauer Kasten) und der Menüleiste mit den vier Menüpunkten (roter Kasten): Anlage, Wissensdatenbank, Checkliste und Datenanalyse. Adaptiert und übersetzt aus [122].

Teilweise besitzen die Menüpunkte untereinander eine direkte Verlinkung. So kann z.B. innerhalb eines Anwendungsfalles im Menüpunkt *Anwendungsfälle* auf Expertenwissen in der *Wissensdatenbank* zugegriffen werden. Nachfolgend wird die Umsetzung der einzelnen Module näher beschrieben.

4.3.2.1 Modul 1: Zugänglichkeit von Daten und Wissen

Modul 1 umfasst in der Implementierung die zwei Menüpunkte *Anwendungsfälle* für die Zugänglichkeit von Daten und Metadaten und *Wissensdatenbank* für die Zugänglichkeit von Expertenwissen.

Zunächst wird der Menüpunkt *Anwendungsfälle* näher erläutert. Durch Klicken auf den Anwendungsfälle-Button erhält der Anwender eine Übersicht über die vorhandenen Anwendungsfälle bzw. ML-Projekte. Wird ein Anwendungsfall angewählt, werden weiterführende Informationen wie z.B. eine Kurzbeschreibung und eine Abbildung angezeigt (Abbildung 4.5). Dadurch erhält der Anwender einen schnellen Überblick über die Anwendungsfälle und kann zudem auch effizient durch aktuelle und bereits abgeschlossene Anwendungsfälle navigieren.

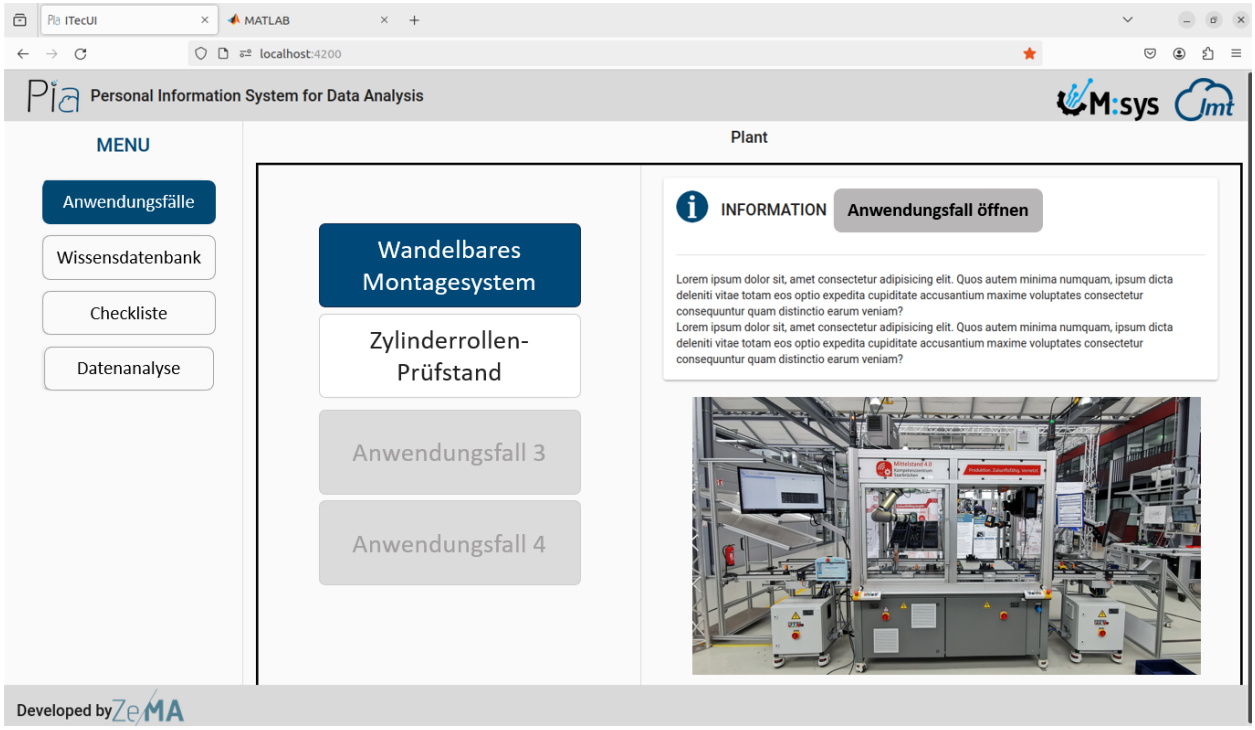


Abbildung 4.5: Übersicht der Anwendungsfälle mit Kurzbeschreibung und Abbildung.

Innerhalb eines Anwendungsfalles erfolgt eine weitere Einteilung nach Konfigurationen. Eine Konfiguration kann auch bspw. eine (Prozess-)Station darstellen und besteht aus folgenden Menüpunkten:

- **Produkt**: Der Menüpunkt *Produkt* enthält relevante Informationen bzw. Metadaten über das Produkt, wie z.B. Produktvarianten, technische Daten und technische Zeichnungen.
- **Betriebsmittel**: Der Menüpunkt *Betriebsmittel* enthält eine Übersicht über die eingesetzten Betriebsmittel. In der nachfolgenden Ebene finden sich relevante Informationen bzw. Metadaten über das jeweilige Betriebsmittel.
- **Messdaten**: Der Menüpunkt *Messdaten*, enthält Messdaten die abgerufen und visualisiert werden können.
- **Video** (optional): Der Menüpunkt *Video* enthält ein Video z.B. eines Prozessablaufes. Dies erlaubt dem Anwender, den Prozess und die dort entstehenden Daten besser verstehen und interpretieren zu können.
- **Sensoren**: Der Menüpunkt *Sensoren* enthält eine Übersicht mit allen verwendeten Sensoren und ihren Metadaten, wie z.B. dem Sensortyp, seiner Position, Abtastrate usw. (vgl. Übersicht Sensoren in Anhang A.2.2).

- **Maschinenlesbare Metadaten**: Der Menüpunkt *Maschinenlesbare Metadaten* enthält alle relevanten Metadaten, z.B. nach der Metadata4Ing (m4i) Ontologie, in einem maschinenlesbaren Format. Der PIA ermöglicht dem Anwender zudem die Visualisierung der Relationen der Metadaten untereinander als graphische Ontologie-Darstellung (vgl. Anhang, Abbildung A.5) [146].
- **Schichtbuch**: Der Menüpunkt *Schichtbuch* enthält ein digitales Schichtbuch.

Jedes der aufgelisteten Elemente wurde, sofern möglich, modular angelegt. Daher können einzelne Module flexibel hinzugefügt, ausgetauscht oder entfernt werden. Sollte ein angelegtes Projekt keine Produkte enthalten, kann dieses Modul entfernt werden.

Der nachfolgende Code-Ausschnitt zeigt beispielhaft die Implementierung einer Station_1 mit dem Prozess_XY und dem Betriebsmittel Roboter_XY in PIA:

```
1  const Station = [
2    {
3      id: '1',
4      title: 'Station_1',
5      img: "Pfad/Station_1/Abbildung_Station_1.jpg",
6      process_1: 'Prozess_XY',
7      resource_1:[
8        {id: '1',
9        name: 'Roboter_XY',
10       technicalName: 'Prozessname',
11       img:"Pfad/Station_1/Prozess_1/Roboter_XY/
           Abbildung_Roboter_XY.jpg",
12       technicalData:"Pfad/Station_1/Prozess_1/Roboter_XY/
           Datenblaetter/Roboter_Technische_Details.pdf",
13       manual:"Pfad/Station_1/Prozess_1/Roboter_XY/
           Datenblaetter/Roboter_Anleitung.pdf",
14       technicalDrawing:"Pfad/Station_1/Prozess_1/
           Roboter_XY/Datenblaetter/
           Roboter_Technische_Zeichnung.pdf"
15       }]
16   }]
```

Listing 4.1: Beispielcode für eine Station. Adaptiert und übersetzt aus [122].

Die entsprechenden Metadaten, wie z.B. Abbildungen oder technische Zeichnungen, werden durch das Hinterlegen ihres Speicherortes (Datenbank) verknüpft.

Abbildung 4.6 zeigt Ausschnitte einer beispielhaften Navigation von einer Station hin zu den spezifischen Informationen eines Roboters durch die resultierende GUI in PIA.

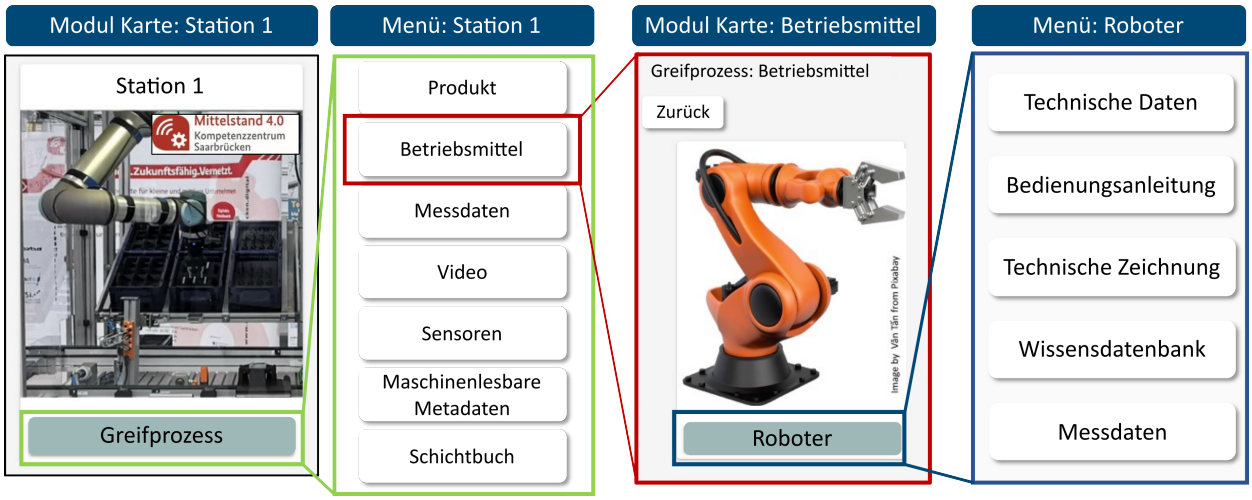


Abbildung 4.6: Beispielhafte Navigation durch die Wissensdatenbank eines Greifprozesses (Anwendungsfall 2). Adaptiert von [122].

Im ersten Schritt wählt der Anwender in einem Anwendungsfall die entsprechende Station Greifprozess aus. Nun erhält der Anwender die verfügbaren Menüpunkte der Station (Abbildung 4.6, grün). Durch das Aufrufen des Menüpunktes *Betriebsmittel* (Abbildung 4.6, rot) werden dem Anwender alle vorhandenen Betriebsmittel aufgelistet und er kann das Betriebsmittel *Roboter* auswählen. Nun kann der Anwender auf alle hinterlegten Informationen des Roboters (Abbildung 4.6, blau) wie die Bedienungsanleitung oder eine technische Zeichnung zugreifen.

Weiterhin sind die Metadaten in PIA maschinenlesbar hinterlegt. Im Rahmen des Softwaredemonstrators wird die m4i Ontologie im JSON-Format verwendet, da diese eine ganzheitliche und semantische Beschreibung der Daten ermöglicht. Die Verknüpfung der Metadaten mit einer Ontologie steigert die FAIRness (vgl. Abschnitt 2.2), insbesondere in den folgenden Punkten:

- **FAIR-F2**: Die Metadaten sind hinreichend beschrieben, da sie dem Standard einer Ontologie folgen.
- **FAIR-F4**: Die maschinelle Durchsuchbarkeit ist gewährleistet.
- **FAIR-I1**: Das Verständnis der Metadaten ist durch die beschriebenen Relationen der Metadaten untereinander gesteigert.

- **FAIR-I2**: Die verwendeten Vokabulare stimmen mit den FAIR-Prinzipien überein.
- **FAIR-I3**: Die Metadaten enthalten geeignete und hochwertige Referenzen auf andere Daten und Metadaten.
- **FAIR-R1.2**: Die Metadaten sind eindeutig und detailliert mit ihrer Herkunft verbunden.
- **FAIR-R1.3**: Die Metadaten entsprechen mindestens den gültigen Standards der Branche.

Neben der Zugänglichkeit der Daten und Metadaten spielt auch der Zugang zu (Experten-)Wissen eine entscheidende Rolle für die Datenanalyse bzw. die Durchführung eines ML-Projektes. Im Softwaredemonstrator PIA wurde das Lessons Learned Register zur Abdeckung von Erfahrungswissen und eine Wissensablage zur Integration von domänenspezifischem Wissen implementiert.

Abbildung 4.7 zeigt die exemplarische Umsetzung des beschriebenen Lessons Learned Registers.

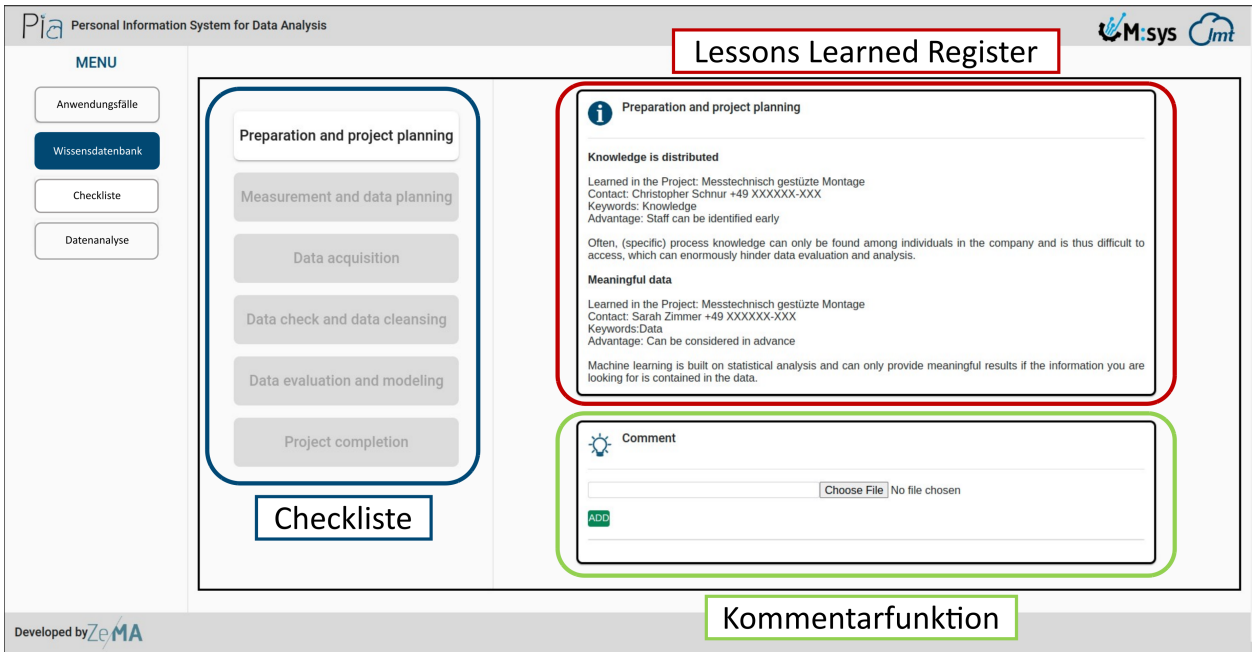


Abbildung 4.7: Implementierung des Lessons Learned Register: Kapitel der Checkliste (blau), Lessons Learned (rot) und Kommentarbox (grün). Adaptiert aus [122].

Die hinterlegten High-Quality Lessons Learned wurden zur besseren Übersicht nach den Schritten eines ML-Projektes (vgl. Abbildung 4.1) gruppiert. Weiterhin kann der Anwender die Lessons Learned mit Kommentaren versehen, sollten diese z.B. ihre

Gültigkeit verlieren oder ergänzt werden müssen. Die erwähnten Lessons Learned Hypothesen werden in Modul 2 im Rahmen der Checkliste eingetragen und werden erst in das Lessons Learned Register übertragen, sobald diese sich mehrfach in Projekten bestätigen.

Die Ablage des domänenspezifischen Wissens in der Wissensablage wurde beispielhaft als Baumstruktur implementiert, anhand derer sich der Anwender von der jeweiligen Obergruppe hin zum spezifischen Objekt navigieren kann. Abbildung 4.8 zeigt einen Ausschnitt der Wissensablage von Anwendungsfall 2 (Abschnitt 5.3), in der domänenspezifisches Wissen für Montageprozesse hinterlegt wurde. Der Anwender kann über die entsprechende Operation, z.B. Fügen, durch den Baum navigieren, bis er zum entsprechenden Betriebsmittel, z.B. einem Schrauber, gelangt und dort (domänenspezifische) Informationen über diesen erhält.

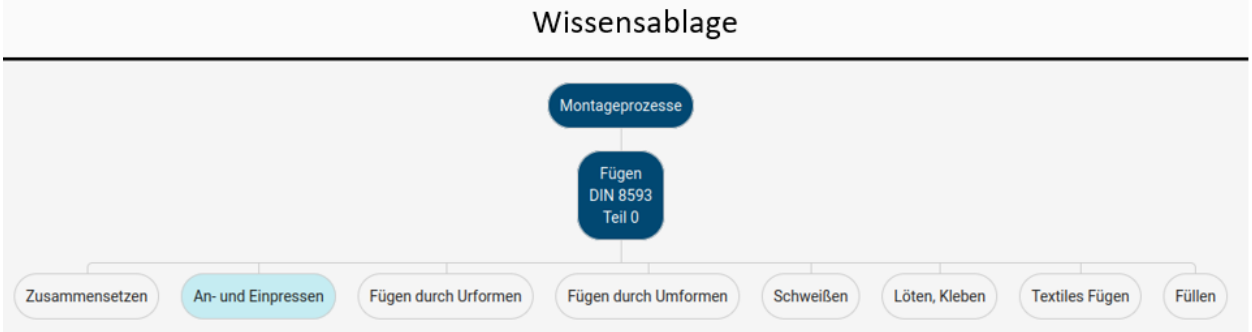


Abbildung 4.8: Implementierung der Wissensablage als Baumstruktur am Beispiel von Montageprozessen.

4.3.2.2 Modul 2: Unterstützung des Anwenders

Die Implementierung von M2 stützt sich auf die in Kapitel 3 vorgestellte Checkliste. Für jedes ML-Projekt bzw. jeden Anwendungsfall existiert in PIA eine dedizierte Checkliste. Dies ermöglicht die Bearbeitung mehrerer ML-Projekte gleichzeitig. Nachdem ein Anwendungsfall erstellt wurde, befindet sich auf der linken Seite (Abbildung 4.9, blauer Kasten) eine Übersicht mit den Checkpunkten, eingeteilt in die Kapitel A bis F. Innerhalb der GUI kann der Anwender bearbeitete Checkpunkte abhaken, wodurch der Projektfortschritt dokumentiert wird. Sind alle Unterpunkte eines Kapitels abgehakt, wird das Kapitel automatisch als bearbeitet markiert. Analog zur Checkliste besitzt jeder Checkpunkt Info-, Tipp- und/oder Hinweisboxen (Abbildung 4.9, roter Kasten), die den Anwender bei der Bearbeitung unterstützen.

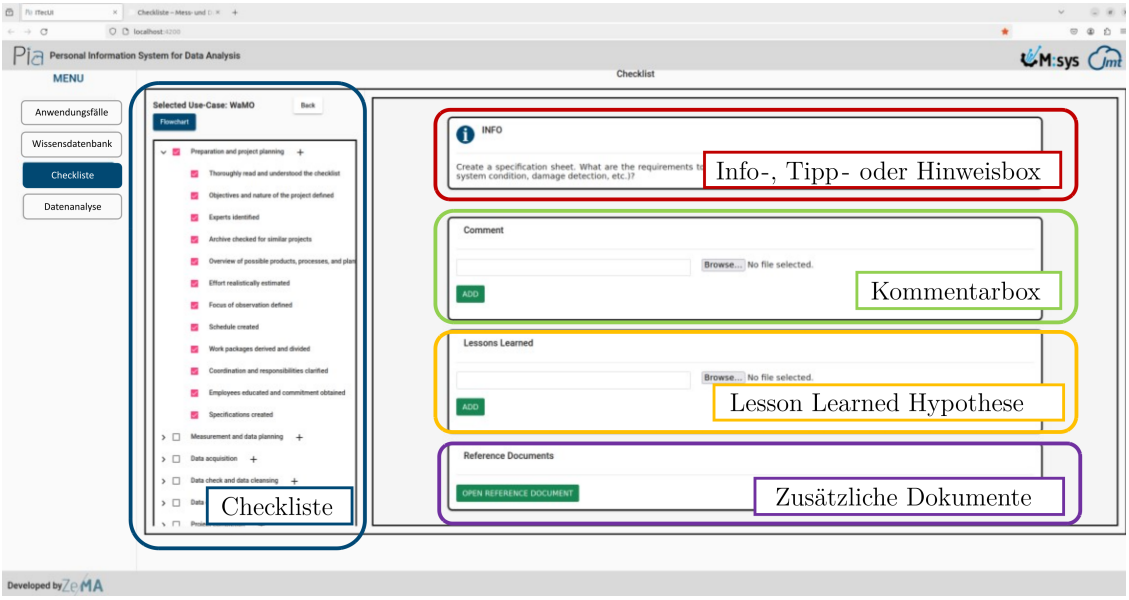


Abbildung 4.9: Implementierung der Checkliste in PIA mit den Feldern: Checkliste (blau), Info-, Tipp oder Hinweisbox (rot), Kommentarbox (grün), Lesson Learned Hypothese (gelb) und der Funktion zusätzliche Dokumente anzuhängen (lila).

Im Vergleich zu der alleinstehenden Variante der Checkliste bietet die Integration in PIA folgende Vorteile:

- **Kommentarfunktion**: Durch eine Kommentarfunktion kann der Anwender Notizen machen oder Besonderheiten festhalten (Abbildung 4.9, grüner Kasten).
- **Lesson Learned Hypothese**: Wird während oder nach der Bearbeitung eines Checkpunktes eine Lesson Learned abgeleitet, kann diese als Lesson Learned Hypothese hinzugefügt werden (Abbildung 4.9, gelber Kasten).
- **Zusätzliche Dokumente**: Zu jedem Checkpunkt können zusätzliche Dateien beigefügt werden (Abbildung 4.9, lila Kasten). Dies können z.B. weiterführende Informationen sein, wie z.B. in Checkpunkt B3 ein UWD.
- **Fortschrittsanzeige**: Durch digitales Abhaken der Checkpunkte kann der Projektfortschritt erfasst werden.
- **Zugänglichkeit:** Durch die hohe Zugänglichkeit in PIA kann von weiteren berechtigten Mitarbeitern auf die Checkliste zugegriffen werden. Dies hat insbesondere den Vorteil, dass das ML-Projekt auch zugänglich bleibt, wenn bspw. ein Mitarbeiter aufgrund längerer Abwesenheit ausfällt.

Weiterhin wurde der Ablaufplan (vgl. Abschnitt 3.5) in PIA integriert (Abbildung 4.10).

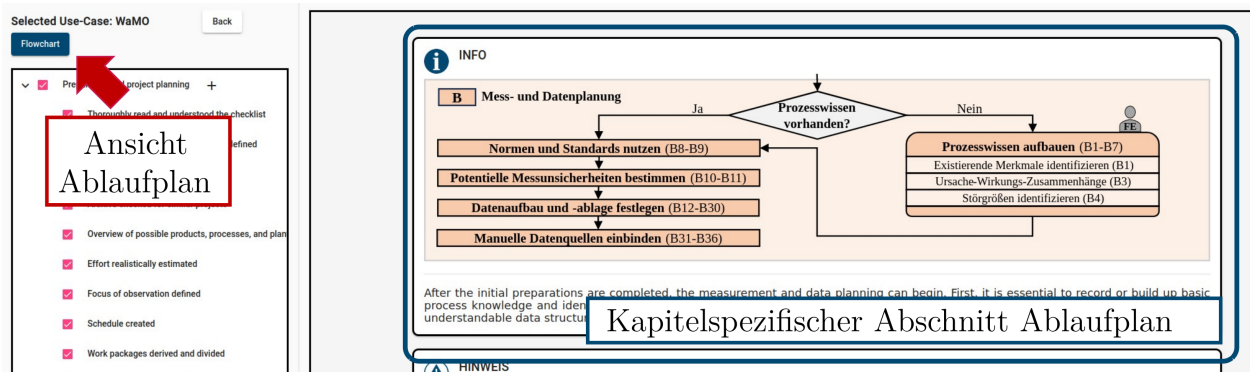


Abbildung 4.10: Implementierung des Ablaufplan in PIA.

Der vollständige Ablaufplan kann über einen Button oberhalb der Übersicht der Checkliste (Abbildung 4.10, roter Pfeil) abgerufen werden. Zudem wurden die kapitelspezifischen Abschnitte des Ablaufplans dem jeweiligen Schritt zugeordnet (Abbildung 4.10, blauer Kasten). Dies ermöglicht dem Anwender die Orientierung im jeweiligen Schritt des ML-Projektes und über den gesamten Projektverlauf hinweg.

4.3.2.3 Modul 3: Datenanalyse

Im Softwaredemonstrator PIA wird M3 durch die Kombination der Checkliste (M2) mit der ML-Toolbox realisiert. Die Checkliste bietet dem Anwender eine systematische Unterstützung bei den zentralen Schritten einer Datenanalyse. So wird der Anwender bei der Datenprüfung und -bereinigung (Kapitel D) und bei der Datenauswertung und Modellbildung (Kapitel E) unterstützt. Der Bedarf eines Tools zur Datenanalyse (B-5) wird in PIA durch die Integration der ML-Toolbox gedeckt. Diese wird durch die Verknüpfung von PIA mit MathWorks® MATLAB Online [147] realisiert. Die Verwendung von MATLAB Online erlaubt eine schnelle und stabile Integration der ML-Toolbox in PIA, da diese in MATLAB entwickelt wurde.

Durch Anklicken des Menüpunkts Datenanalyse (siehe Abbildung 4.4) in PIA kann der Anwender MATLAB Online aufrufen. Abbildung 4.11 zeigt das sich anschließend öffnende Interface der ML-Toolbox. Auf der linken Seite (Abbildung 4.11, blaue Box) befindet sich die Toolbox mit den implementierten Algorithmen. Im mittleren Bereich befindet sich der Coding Bereich (Abbildung 4.11, grüne Box), in welchem der Anwender z.B. die ML-Toolbox oder aber einzelne Algorithmen anwenden kann. Auf der rechten Seite werden die Daten und Ergebnisse visualisiert (Abbildung 4.11, rote Box).

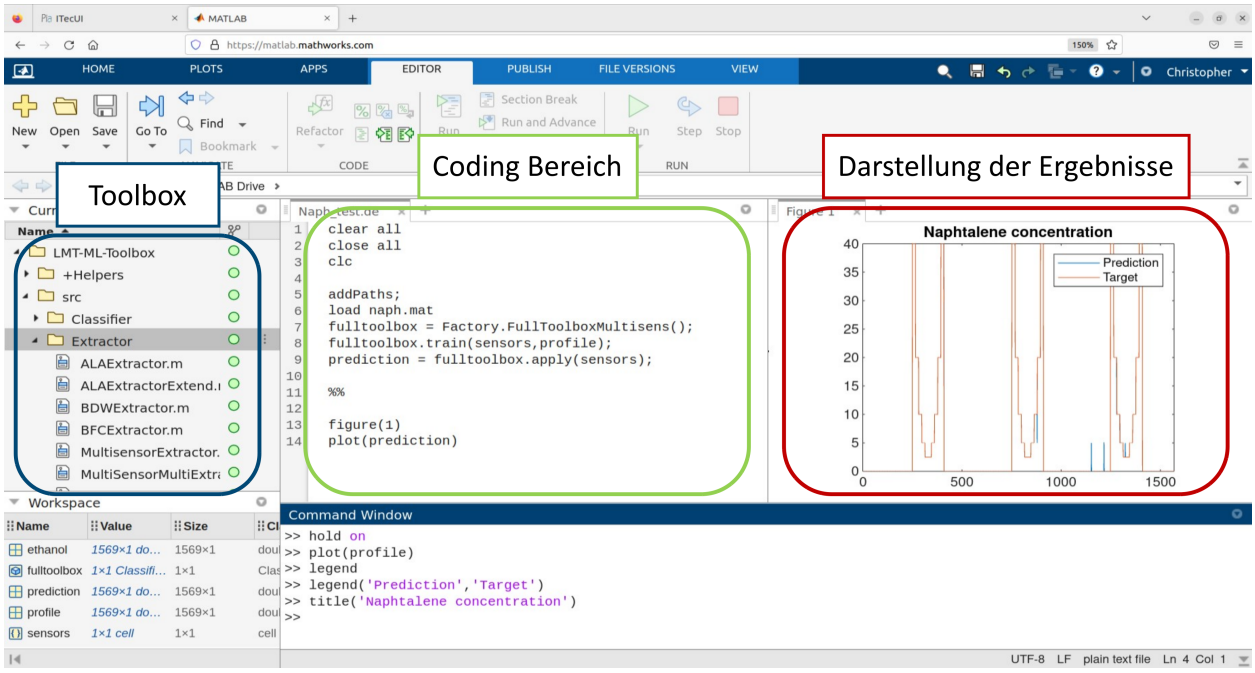


Abbildung 4.11: Implementierung der ML-Toolbox in PIA mit ihrer Struktur (blau), dem Coding Bereich (grün) und der Darstellung der Ergebnisse (rot). Adaptiert aus [122].

4.4 Diskussion und Zwischenfazit

PIA kann Anwender bei der Bearbeitung eines ML-Projektes ganzheitlich unterstützen und deckt dabei die geforderten Bedarfe B-1 bis B-5 ab. Hierbei stützt sich PIA auf die drei Module Zugänglichkeit von Daten und Wissen (M1), Unterstützung des Anwenders (M2) und Datenanalyse (M3).

Hinsichtlich der initial gestellten Forschungsfragen gilt:

- **Forschungsfrage 1**: Innerhalb der in Kapitel 3 vorgestellten Checkliste wurden bereits aktuelle Forschungskonzepte aufgegriffen und integriert. Die Implementierung der Checkliste in PIA kann die Nutzererfahrung für den Anwender weiterhin steigern, indem nötiges Fachwissen in die Wissensablage (M1) integriert wird und somit leichter zugänglich ist.

- **Forschungsfrage 2**: Durch die Verwendung einer einheitlichen Plattform, in der einerseits die notwendigen Schritte einer gezielten MuD integriert und andererseits Daten und Wissen zugänglich sind, werden unerfahrene industrielle Anwender bei der Aufzeichnung von Daten mit einer hohen Datenqualität unterstützt.

- **Forschungsfrage 3**: Die Verknüpfung der Zugänglichkeit von (Fach-)Wissen (M1), einer schrittweisen Unterstützung bei der Durchführung (M2) und einer

integrierten Toolbox (M3) befähigen unerfahrene industrielle Anwender zur Durchführung einer ersten Datenanalyse.

Der vorgestellte Softwaredemonstrator kann nicht als fertige Software angesehen werden. Vielmehr soll der Softwaredemonstrator Anwender und Unternehmen dazu anregen, einen ganzheitlichen Ansatz bei der Bearbeitung von ML-Projekten zu verfolgen. PIA dient hier als Framework, an dem sich Anwender und Unternehmen orientieren können. Aufgrund des breiten Spektrums an industriellen Anwendungen und verschiedenen Ausgangssituationen, wie z.B. unterschiedliche Wissensstände, finanzielle und personelle Kapazitäten der Unternehmen, kann für PIA keine allgemeingültige Variante entwickelt werden. Einschränkungen des vorgestellten Softwaredemonstrators sind:

- **Angular**: Die Nutzung von Angular erlaubte die zeit-effiziente Implementierung eines Softwaredemonstrators. Jedoch existieren für den spezifischen Anwendungsfall ggf. geeignetere Lösungen, insbesondere hinsichtlich Robustheit und Sicherheit.
- **Komponenten**: Die vorgestellte Struktur der Komponenten erlaubte ebenfalls eine zeit-effiziente Implementierung und sollte hinsichtlich Nomenklatur und Aufbau verbessert werden.
- **Fehlendes Backend**: Der Softwaredemonstrator ist lediglich als Frontend implementiert und nicht mit einem Backend gekoppelt. Die integrierten Inhalte, wie bspw. Metadaten, sind derzeit statisch hinterlegt. Daher können u.a. Fortschritte und Kommentare nicht dauerhaft gespeichert werden. Die Implementierung eines Backend und die Verknüpfung mit einer Datenbank sind für den Einsatz in der Industrie notwendig.
- **Manuell eingetragene Metadaten**: Die Metadaten von Anlagen und Betriebsmitteln wurden manuell eingetragen. Idealerweise würden z.B. Hersteller mit ihren Produkten eine maschinenlesbare Datei mitliefern, welche bereits alle notwendigen Metadaten gemäß bestehender Ontologien enthält.
- **Implementierung ML-Toolbox**: Die ML-Toolbox wurde in PIA mit MathWorks® MATLAB Online verknüpft, was voraussetzt, dass die Unternehmen eine entsprechende Lizenz besitzen oder erwerben.

- **Eingeschränkte Daten**: Die vorgestellte Variante von PIA fokussiert sich auf zeitdiskrete Daten. In der Industrie können jedoch auch andere Datenarten wie z.B. Bilddaten anfallen. Um diese muss PIA erweitert werden.
- **Reduzierte Nutzerfreundlichkeit**: Im Softwaredemonstrator wurde weitgehend auf eine hohe Modularität geachtet. Um z.B. eine neue Station hinzuzufügen, kann der Code mit geänderten Metadaten wiederverwendet werden. Praxistauglicher wäre hier die Implementierung eines Buttons welcher automatisch eine Station anlegt, die durch den Nutzer in einem Interface mit relevanten Informationen ergänzt wird.
- **Fehlende Schnittstellen**: PIA wurde als stand-alone Variante vorgestellt. Sinnvoll wäre die Schaffung von Schnittstellen zu externer, bereits im Unternehmen verwendeter Software wie z.B. Enterprise Resource Planning (ERP)-Systemen.

Innerhalb der Publikation [122] wurde eine Vorgänger-Version des Softwaredemonstrators auf der Plattform GitHub als open source veröffentlicht [148].

5 Beispielhafte Erprobung

Im nachfolgenden Kapitel wird die zuvor vorgestellte Methodik des persönlichen Informationsassistenten (PIA) mit der implementierten Checkliste anhand von zwei Anwendungsfällen beispielhaft demonstriert. Zunächst wird hierzu ein Messkoffer zur Datenaufzeichnung in den Anwendungsfällen motiviert und entwickelt.

5.1 Messkoffer für flexible Feldmessungen

5.1.1 Motivation

Neue Maschinen und Anlagen sind häufig mit hohen Investitionskosten verbunden und benötigen eine mehrjährige Verwendungsdauer, bis sich deren Anschaffung amortisiert [149]. Dies führt dazu, dass in der Industrie überwiegend Bestandsanlagen vorhanden sind. Diese Anlagen verfügen tendenziell nur über wenige Sensoren, welche primär zu Steuerungs- und Regelungszwecken dienen und nur bedingt für die Anwendung von maschinellem Lernen (ML) geeignet sind. Abhilfe kann das sog. Retrofitting, das Nachrüsten neuer Technologien an bestehenden Maschinen und Anlagen, schaffen [150]. Auch das Retrofitting von Sensoren und Messtechnik kann hohe Investitionen erfordern. Ein möglicher Lösungsansatz ist die Anschaffung eines wiederverwendbaren Messsystems, mit dem erste Messreihen zur Potenzialanalyse aufgezeichnet werden können, das die folgenden Kriterien erfüllt:

- **Flexibel**: Das Messsystem muss flexibel an verschiedenen Anlagen und Maschinen einsetzbar sein und den Einsatz unterschiedlicher Sensoren ermöglichen.
- **Autark**: Das Messsystem muss eine autarke Datenerfassungseinheit bilden, welche nicht zwangsweise mit der Anlage oder Maschine verbunden werden muss. Dies erlaubt eine schnelle Installation des Messsystems, ohne die bestehende Anlage zu verändern, und vermeidet Herausforderungen hinsichtlich der Schnittstellen (Herausforderung S1 nach Wilhelm in Tabelle 2.1).
- **Kompakt**: Das Messsystem muss kompakt und portabel sein.

- **Robust**: Das Messsystem muss für den Betrieb in der Anwendungsumgebung geeignet sein und bis zu einem gewissen Grad vor Schmutz und Beschädigungen geschützt sein.
- **Anwendbar**: Das Messsystem muss den Anwender dazu befähigen, ohne hohen technischen Aufwand erste Daten aufzuzeichnen und zu analysieren (Forschungsfrage 2 und 3).

Der Messkoffer wird als Datenerfassungssystem für Anwendungsfall 1 (Abschnitt 5.2) und Anwendungsfall 2 (Abschnitt 5.3) verwendet.

5.1.2 Konzept und Aufbau

Um die oben genannten Anforderungen an den Messkoffer zu erfüllen, wurde das CompactRIO (cRIO) System der Firma National Instruments verwendet. Das cRIO System ist ein echtzeitfähiger Embedded-Controller mit integriertem Field Programmable Gate Array (FPGA), welcher über ein Stecksystem mit verschiedensten Messmodulen ausgestattet werden kann. Der Messkoffer besitzt einen integrierten Switch, welcher mit dem cRIO System und ggf. Sensoren verbunden wird. Der Switch ist über den Ethernet-Eingang an der Seite des Messkoffers erreichbar, welcher auch zur Kommunikation mit einem PC oder Laptop dient. Nachfolgend werden die ausgewählten Komponenten der Firma National Instruments sowie zusätzliche Komponenten des Messkoffers aufgelistet und kurz erläutert:

- **NI cRio 9040** [151]: Der Embedded-Controller besitzt vier Slots für Messmodule und kann durch LabVIEW (National Instruments) programmiert werden.
- **NI 9232** [152]: Das Schall- und Schwingungsmessmodul mit einem 24-Bit Analog-to-digital converter (ADC) besitzt drei Kanäle und kann diese mit je 102,4 kS/s abtasten. An dieses Messmodul können z.B. Beschleunigungssensoren angeschlossen werden. Damit sind Vibrationsmessungen möglich, welche eine gängige Methode zur Bewertung technischer Systeme darstellen [98].
- **NI 9215** [153]: Das universale Spannungseingangsmodul mit einem 16-Bit ADC besitzt vier Kanäle mit kann diese mit je 100 kS/s abtasten. Dieses Messmodul erlaubt den Anschluss einer Vielzahl verschiedener Sensoren, welche durch Spannungsmessung ausgelesen werden können.
- **TRENDnet TI-PE50**: Der Power over Ethernet (PoE) Switch erlaubt den Anschluss von Sensoren, welche einen Ethernet-Anschluss und PoE benötigen.

- **Puls CS5**: Das 24V Netzteil dient der Stromversorgung des NI cRio 9040 und des TRENDnet TI-PE50. Außerdem können zusätzliche Sensoren über Reihenklemmen mit Strom versorgt werden.
- **Toshiba DTB320**: Zur Speicherung der Daten wird eine externe Festplatte mit 2 TB Speicherkapazität verwendet. Diese kann bei Bedarf durch eine Festplatte mit einer höheren Kapazität ausgetauscht werden.

Die aufgelisteten Komponenten wurden anhand der bereits bestehenden Infrastruktur am Lehrstuhl für Messtechnik ausgewählt und erfüllen die gestellten Anforderungen.

Abbildung 5.1 (a) zeigt den zusammengebauten Messkoffer mit den aufgelisteten Komponenten.

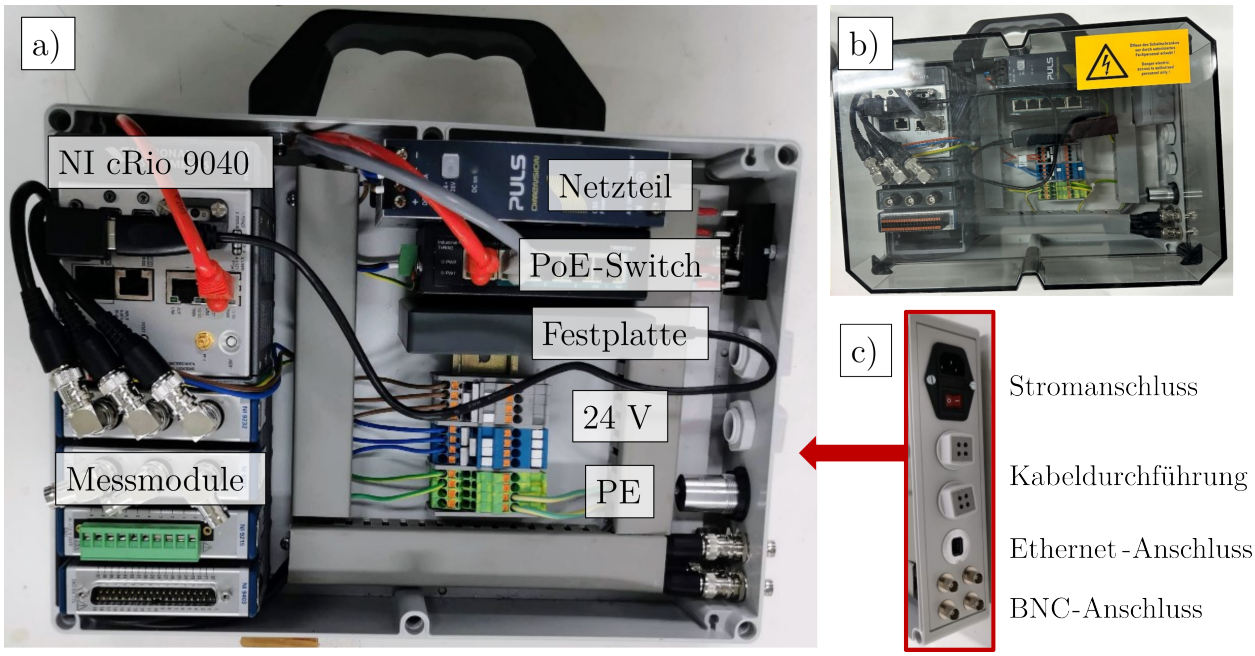


Abbildung 5.1: **a)** Darstellung des offenen Messkoffers mit seinen Komponenten, **b)** des geschlossenen Messkoffers und **c)** den seitlichen Anschlüssen.

Um die empfindliche Messtechnik zu schützen, wird ein glasfaserverstärktes Polycarbonatgehäuse der Firma Rittal verwendet (Abbildung 5.1, b). Der Messkoffer wird über einen seitlichen Anschluss (Abbildung 5.1, c) mit Strom versorgt und besitzt weiterhin einen Ethernet-Anschluss, vier BNC-Anschlüsse (gängiger Anschluss für Beschleunigungssensoren) sowie acht Kabeldurchführungen zum Anschluss der Sensorik.

Die GUI des Messkoffers wurde mittels der grafischen Programmierumgebung LabVIEW der Firma National Instruments umgesetzt und besteht aus den Voreinstellungen (Abbildung 5.2) und der Messoberfläche (Abbildung 5.3). In den Voreinstellungen (Abbildung 5.2) kann der Nutzer über ein Dropdown-Menü die an

den Messkoffer angeschlossenen Sensoren sowie die gewünschte Abtastrate und die Anzahl aufzuzeichnender Werte auswählen.

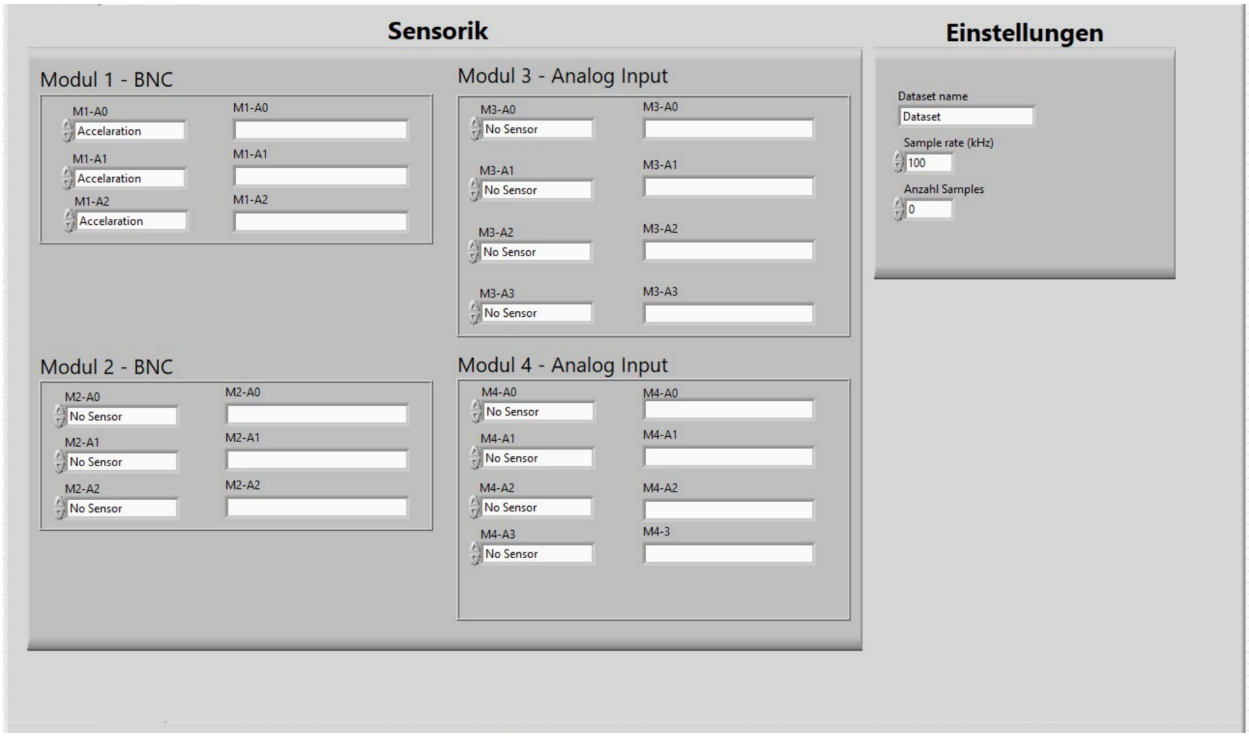


Abbildung 5.2: Darstellung der Voreinstellungen im User Interface des Messkoffers.

Anschließend kann in der Messoberfläche (Abbildung 5.3) die Messung gestartet werden.

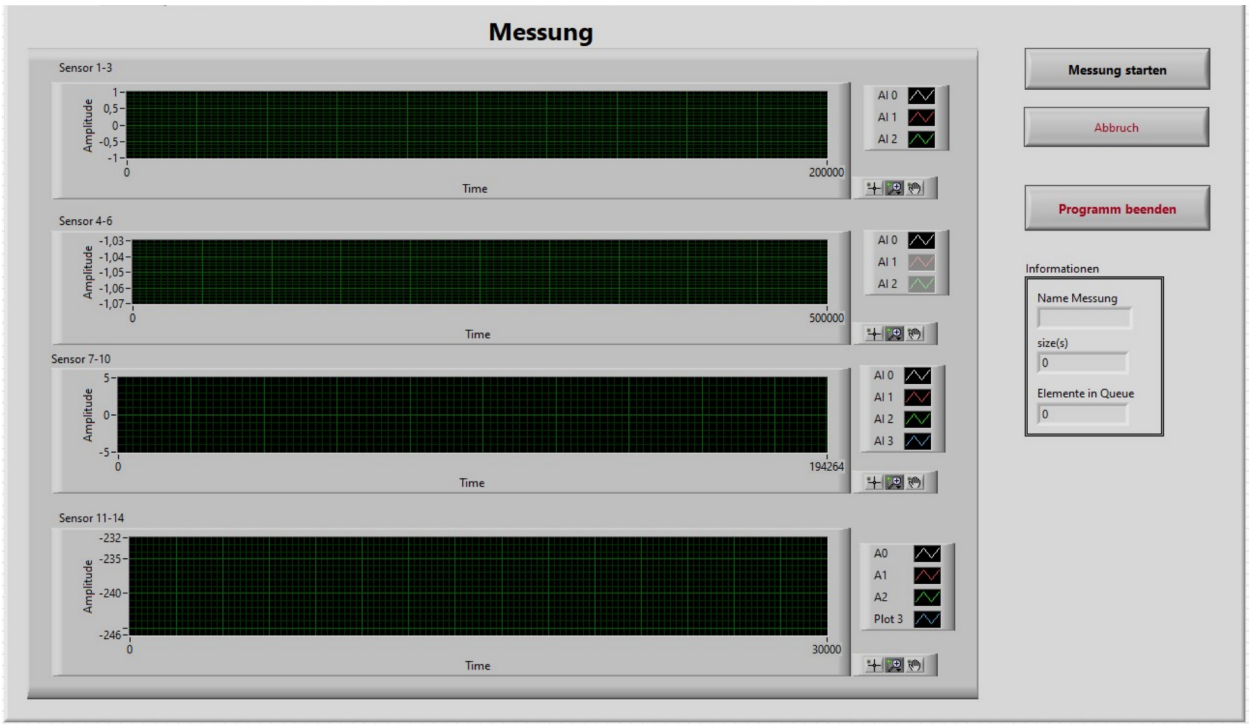


Abbildung 5.3: Darstellung der Messoberfläche im User Interface des Messkoffers.

Während der Messung werden dem Nutzer die gemessenen Werte live angezeigt bzw. geplottet. Dies ermöglicht dem Nutzer, die Messung zu überwachen und eine ordnungsgemäße Funktion der Sensoren sicherzustellen. Nach Abschluss einer Messung wird diese im Technical Data Management Streaming (TDMS)-Format auf der externen Festplatte des Messkoffers gespeichert. Das TDMS-Format ist ein von National Instruments entwickeltes und für LabVIEW und cRIO optimiertes Datenformat, welches sich für große Datenströme eignet und auch von externer Software (wie z.B. MATLAB) gelesen werden kann [154]. Weiterhin wird vom Messkoffer automatisch der Zeitstempel einer internen Clock erfasst und in der TDMS-Datei gespeichert.

Der vorgestellte Messkoffer erfüllt die in Abschnitt 5.1 definierten Anforderungen und bildet ein flexibles, autarkes, kompaktes und robustes Messsystem zur Potenzialanalyse im industriellen Einsatz. Durch die GUI und den Plug & Play-Ansatz wird es unerfahrenen Anwendern ermöglicht, niederschwellig Daten für ein ML-Projekt aufzuzeichnen (Forschungsfrage 1 und 2).

5.2 Anwendungsfall 1: Zylinderrollenlager

Wälzlager sind ein weit verbreitetes Maschinenelement und finden vielfach Anwendung in der Industrie. Sie werden zur Reduzierung der Reibung und zur Abstützung radialer und axialer Kräfte verwendet [155]. Bei Zylinderrollenlagern handelt es sich um eine Bauart der Wälzlager, welche hohe Kräfte in radialer Richtung aufnehmen können [98]. Trotz einer allgemein hohen Zuverlässigkeit können Lager vorzeitig ausfallen. Die Schadens- und Ausfallmechanismen von Wälzlagern sind gut erforscht und es existieren bereits zahlreiche Ansätze, Beschädigungen zu erkennen, z.B. durch die Analyse von entstehenden Schwingungen (Köperschall und Luftschall) [156].

Im Kontext von I4.0 und der vorausschauenden Wartung wird zunehmend auf die datenbasierte Auswertung mittels ML gesetzt, um Schäden frühzeitig zu erkennen und ungeplante Ausfälle zu verhindern. Um solche ML-Modelle zu untersuchen, existieren mehrere öffentlich zugängliche und etablierte Datensätze wie z.B.:

- Case Western Reserve University (CWRU) [157]: Der Datensatz besteht aus künstlich beschädigten Lagern in mehreren Schadensgrößen. Mit Hilfe von magnetisch befestigten Beschleunigungssensoren wurden Daten unter drei Laststufen aufgezeichnet.

- Paderborn University (PU) [158]: Der Datensatz besteht aus 26 beschädigten Lagern und sechs unbeschädigten Lagern, welche in vier Szenarien getestet wurden.

Teilweise handelt es sich um künstliche, teilweise um reale Beschädigungen am Innen- und Außenring. Während der Messungen wurden neben Schwingungen zudem die Rotationsgeschwindigkeit, Last, Drehmoment und die Temperatur gemessen.

- NASA Bearing Dataset (NASA-BD) [159]: Der Datensatz besteht aus drei Experimenten, bei denen jeweils vier Lager unter einer konstanten Last bis zum Erreichen der Verschleißgrenze belastet und dabei mit Beschleunigungssensoren vermessen wurden.

Jeder der aufgelisteten Datensätze variiert bis zu einem gewissen Grad Umgebungseinflüsse, wie z.B. Rotationsgeschwindigkeit oder Last. Diese sind im Sinne der MuD als unvollständig zu betrachten, da keine ganzheitliche Betrachtung der existierenden Störgrößen vorliegt oder dokumentiert wurde. Die Berücksichtigung der Störgrößen ist jedoch essenziell für die Erzeugung von robusten Modellen. Daraus ergibt sich die Notwendigkeit, einen Datensatz aufzuzeichnen, welcher die gezielte und kontrollierte Einbringung von Störgrößen beinhaltet. Ziel des ersten Anwendungsfalls ist daher einerseits die Aufzeichnung eines Datensatzes, welcher gängige Störgrößen berücksichtigt (Forschungsaspekt), und andererseits die Durchführung eines ML-Projektes unter Verwendung von PIA und der Checkliste zur Schadenserkennung am Innenring von Zylinderrollenlagern (Evaluation der Methodik).

Die nachfolgenden Abschnitte sind anhand der Kapitel der Checkliste gegliedert und behandeln die Bearbeitung des ML-Projekts aus der Perspektive des Anwenders. Da die Anwendung der Checkliste in der Forschung nicht deren primäres Einsatzgebiet ist, entfallen industriespezifische Checkpunkte oder werden im Forschungskontext interpretiert. Dies verdeutlicht die universelle Anwendbarkeit der Checkliste, auch über die Verwendung im Mittelstand hinaus.

Der Bau des Zylinderrollenlager-Prüfstandes und die Aufzeichnung der Daten wurden durch die Projekte *Mittelstand 4.0-Kompetenzzentrum Saarbrücken* des Bundesministerium für Wirtschaft und Energie (BMWi) und *Verteilte Produktion für die saarländische Automotivindustrie: Nachhaltig, Vernetzt, Resilient* (VProSaar) gefördert.

Alle Darstellungen der Implementierung in PIA befinden sich aus Gründen der Übersichtlichkeit und Leserlichkeit in Anhang A.2.1.

5.2.1 Vorbereitung und Projektplanung

Zunächst wurde durch den Anwender in PIA ein neues Projekt (siehe Anhang A.2.1, Abbildung A.1) erstellt und anschließend gemäß Checkpunkt A1 die integrierte Checkliste (siehe Anhang A.2.1, Abbildung A.2) gelesen. Im nächsten Schritt wurden die Ziele des ML-Projektes festgelegt (Checkpunkt A2):

- **AF1-Ziel 1**: Ermittlung gängiger Einfluss- und Störgrößen auf den Verschleiß von Zylinderrollenlagern.
- **AF1-Ziel 2**: Aufzeichnung eines Datensatzes mit gezielter Variation der ermittelten Einfluss- und Störgrößen unter variierenden Rotationsgeschwindigkeiten und Laststufen.
- **AF1-Ziel 3**: Erzeugung eines gegenüber Positionsänderungen robusten ML-Modelles zur Erkennung von Innenringenschäden an Zylinderrollenlagern unter den Einflussgrößen Rotationsgeschwindigkeit und Last.

Hinsichtlich AF1-Ziel 3 erfolgte im Rahmen dieser Untersuchung zunächst eine Einschränkung auf eine konstante Rotationsgeschwindigkeit. Abbildung 5.4 zeigt den für die Untersuchungen verwendeten Zylinderrollenlager-Prüfstand, welcher in PIA als Use-Case integriert wurde (siehe Anhang A.2.1, Abbildung A.3).

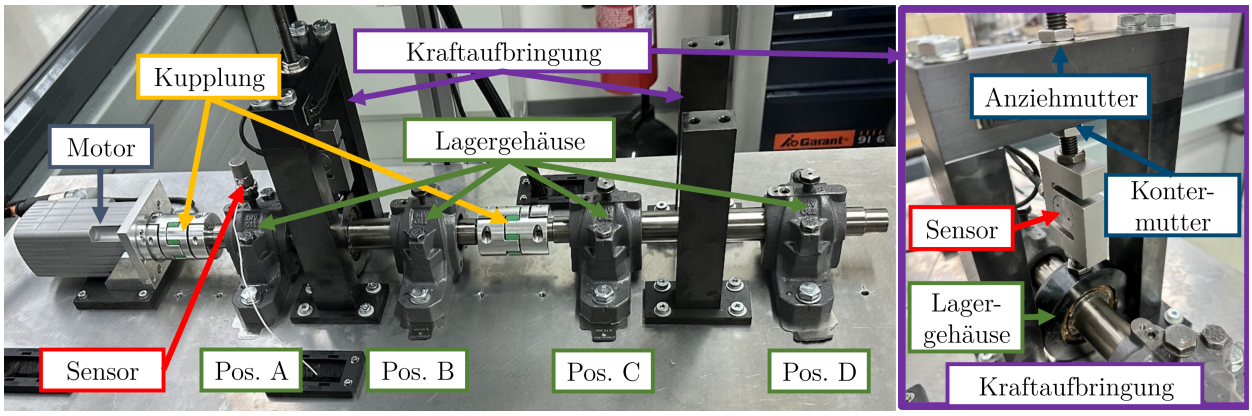


Abbildung 5.4: Darstellung des Zylinderrollenlager-Prüfstands mit seinen Komponenten.

Abbildung 5.4 zeigt den Zylinderrollenlager-Prüfstand. Er besteht aus einem Motor (grau), welcher über zwei Kupplungen (gelb) zwei Wellen antreibt. Über eine Kraftaufbringung (violett) können die in den Lagergehäusen (grün) verbauten Zylinderrollenlager belastet werden. Eine Übersicht der verbauten mechanischen Komponenten sowie des Datenerfassungssystems findet sich in Tabelle 5.1.

Tabelle 5.1: Übersicht der mechanischen Komponenten (I) und der Komponenten des Datenerfassungssystem (II).

Komponente	**Modell**	**Hersteller**
	I. Mechanischer Aufbau	
Motor	EMMS-AS-70S-LS-RSB	Festo
Motorcontroller	CMMP-AS-C2-3A-M3	Festo
Kupplung	GWE 5106-24-11-25	Ringfeder
Loslager	NU206-E-XL-TVP2	Schaeffler Technologies
Festlager	1206-TVH	Schaeffler Technologies
Lager Kraftaufbringung	NU207-E-XL-TVP2	Schaeffler Technologies
	II. Data Acquisition System	
Beschleunigungssensor	3233a	Dytran Instruments
Kraftsensor	K-25	Lorenz Messtechnik
Embedded Controller	cRIO 9040	National Instruments
Schwingungsmessmodul	NI-9232	National Instruments
Spannungseingangsmodule	NI-9215	National Instruments

Die aufgelisteten Betriebsmittel wurden anschließend mit den zugehörigen Dokumenten, wie z.B. technischen Daten (siehe Anhang, Abbildung A.4), und deren Metadaten gemäß der m4i Ontologie (siehe Anhang A.2.1, Abbildung A.5) in PIA integriert. Weiterhin wurde ein Video, welches den Aufbau und die Funktionsweise des Prüfstandes zeigt, in PIA hinterlegt (siehe Anhang A.2.1, Abbildung A.6).

Die Finanzierung bzw. Förderung (Checkpunkt A3) des ML-Projektes erfolgt über das eingangs erwähnte Forschungsprojekt Verteilte Produktion für die saarländische Automotivindustrie: Nachhaltig, Vernetzt, Resilient (VProSaar). Obwohl sich innerhalb des Projektkonsortiums ML-Experten befinden, welche in der Datenanalyse unterstützen könnten, werden die Checkpunkte A4 und A5 übersprungen. So kann die Methodik in Anwendungsszenario 1 zunächst unter erschwerten Bedingungen, wie sie in mittelständischen Unternehmen auftreten können, getestet werden. Gemäß Checkpunkt A6 wurden ähnliche Projekte der Vergangenheit geprüft. Dies umfasste die Ergebnisse einer Vorstudie [160] über die Schadenserkennung (beschädigt/unbeschädigt) eines Zylinderrollenlagers des Typs NU206-E-XL-TVP2 unter variierender Einbauposition. Die Einbauposition der Zylinderrollenlager wurde in der Studie, basierend auf den Ergebnissen einer Hauptkomponentenanalyse, als relevante Störgröße identifiziert

(Checkpunkt B4). Abbildung 5.5 zeigt ein Zylinderrollenlager des Typs NU206-E-XL-TVP2 mit seinen einzelnen Komponenten.

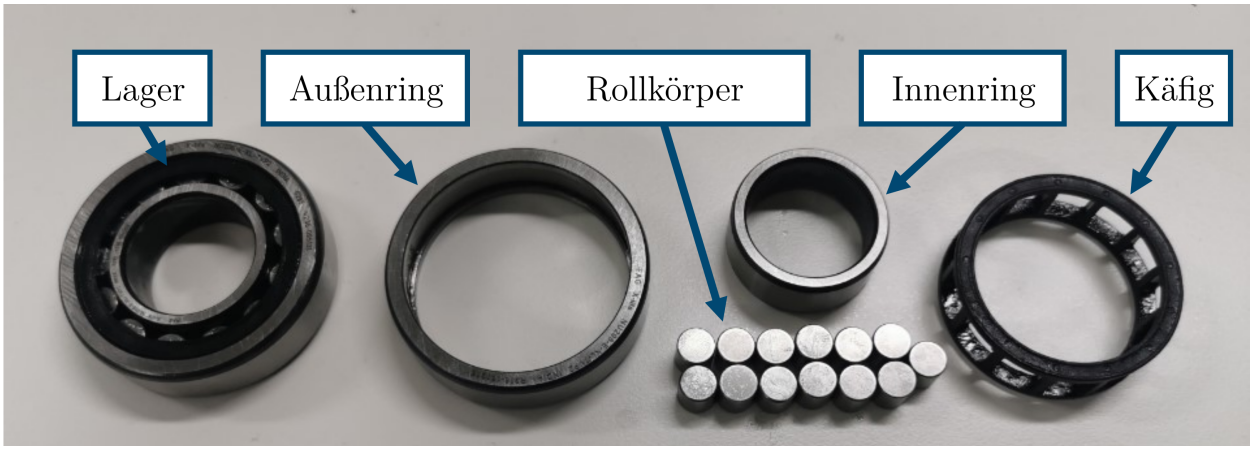
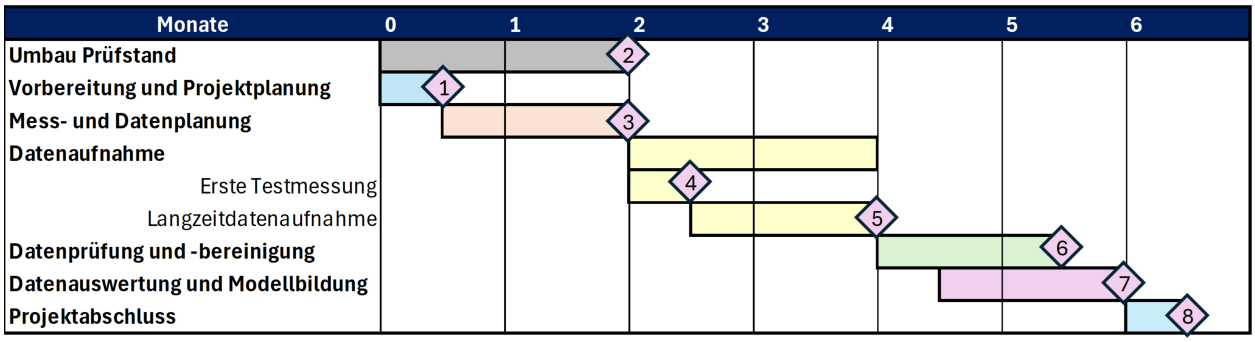


Abbildung 5.5: Darstellung eines Zylinderrollenlager des Typs NU206-E-XL-TVP2 mit seinen Komponenten Außenring, Rollkörper, Innenring und Käfig.

Innerhalb des ML-Projektes soll der Fokus zunächst auf dem Innenring liegen und erst in nachfolgenden Studien um weitere Komponenten erweitert werden, z.B. dem Außenring, dem Käfig oder den Rollkörpern (Checkpunkt A7). Im Nachfolgenden wurde der Aufwand des ML-Projektes abgeschätzt (Checkpunkte A8) und ein Betrachtungsfokus definiert (Checkpunkt A9). Analog zur Vorstudie wurde der Fokus auf Zylinderrollenlager des Typs NU206-E-XL-TVP2 gelegt, welche am Innenring künstlich beschädigt wurden. Anschließend erfolgte die Erstellung eines Zeitplans mit Meilensteinen (Checkpunkt A10), welcher in Abbildung 5.6 dargestellt ist. Neben den jeweiligen Schritten eines ML-Projektes wurde zusätzliche Zeit für einen Umbau des Prüfstandes eingeplant.



Meileinsteine #

1. Vorbereitung und Projektplanung abgeschlossen
2. Prüfstand umgebaut und Einsatzbereit
3. Mess- und Datenplanung abgeschlossen
4. Erfolgreich erste Testmessungen durchgeführt und Langzeitdatenaufnahme gestartet
5. Messkampagne abgeschlossen
6. Daten geprüft und bereinigt
7. Robustes ML-Modell erzeugt
8. Projekt abgeschlossen

Abbildung 5.6: Zeitplan mit Meilensteinen für Anwendungsszenario 1.

Mit Ausnahme der Datenerfassung führte der Autor dieser Dissertation die Bearbeitung der Checkliste durch und übernahm die Koordination des ML-Projektes (Checkpunkt A11 und A12). Die Erfassung der Daten wurde durch zwei Hilfswissenschaftler durchgeführt, welche zuvor über die Ziele und Hintergründe des ML-Projektes aufgeklärt wurden (Checkpunkt A13). Abschließend wurden folgende Anforderungen im Lastenheft, ergänzend zu den Zielen **AF1-Ziel 1 & 2** definiert (Checkpunkt A14):

- **AF1-LH 1**: Die Daten müssen in einer hohen Qualität vorliegen und gemäß der FAIR-Prinzipien aufgezeichnet werden.
- **AF1-LH 2**: Während der Messkampagne soll die Rotationsgeschwindigkeit aufgrund des geringen technischen Aufwandes variiert werden.

5.2.2 Mess- und Datenplanung

Nachdem die Vorbereitung und Projektplanung abgeschlossen wurde, konnte die Mess- und Datenplanung erfolgen. Hier entfallen die Punkte B1 und B2 (Anlagenmerkmale, Qualitätsanforderungen), B8 (firmeninterne Standards), B16 (prozess- bzw. anlagenübergreifende Beziehungen), B23 (Prozessgrenzwerte) und B32 (digitale Schichtbücher), da diese primär für den Einsatz in Unternehmen relevant sind und nicht auf den gewählten Anwendungsfall zutreffen.

5.2.2.1 Aufbau von Prozesswissen

Kernelement der MuD und des Aufbaus von Prozesswissen ist die Erstellung eines UWD in Checkpunkt B3. Abbildung 5.7 zeigt das für die Zylinderrollenlager erstellte UWD, welches zur Identifikation relevanter Einflussgrößen auf das Messergebnis diente.

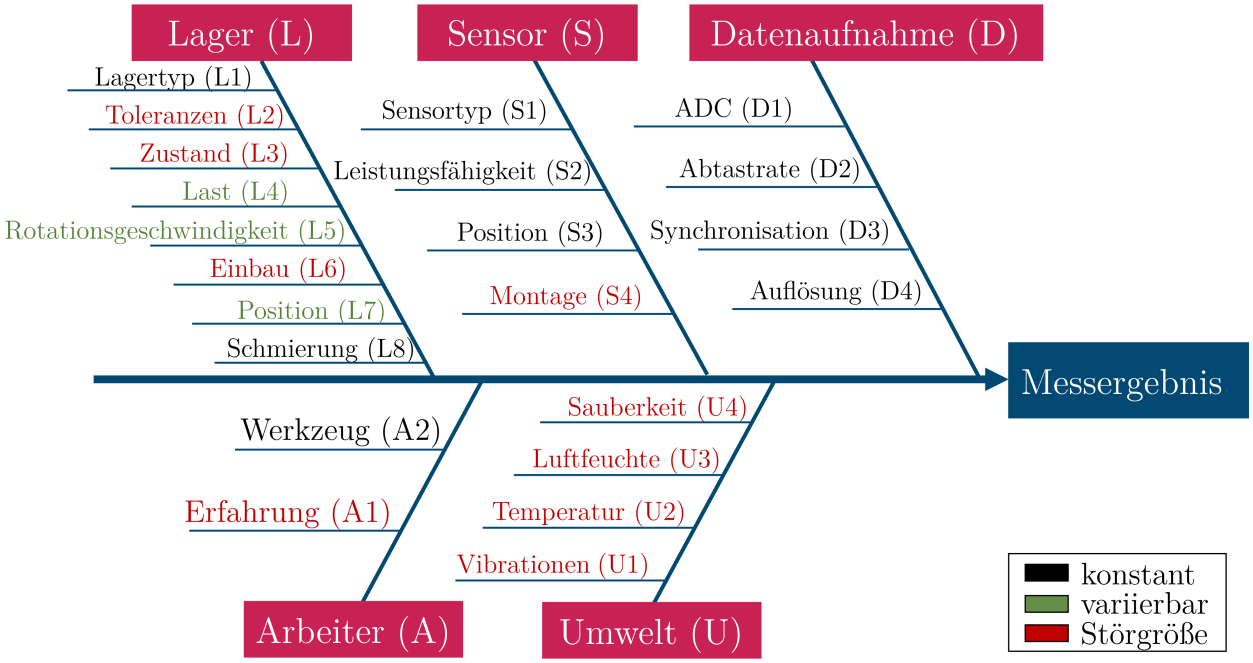


Abbildung 5.7: Ursache-Wirkungs-Diagramm für die Einflussgrößen des Zylinderrollen-Prüfstandes auf das Messergebnis. Adaptiert und erweitert aus [160]

Die jeweiligen Einflussfaktoren wurden in die Hauptfaktoren Lager (L), Sensor (S), Datenaufnahme (D), (Projekt-)Arbeiter (A) und Umwelt (U) eingegliedert. Diese Eingliederung weicht von der gängigen Eingliederung (Mensch, Maschine, Material, Milieu und Methode) zugunsten einer spezifischeren Anwendung des UWD und gleichzeitig höheren Übersichtlichkeit ab. Jeder Einflussfaktor wurde zudem in die drei Kategorien konstant, variierbar und Störgröße eingeteilt (Checkpunkt B4):

- **Konstant** (schwarz): Konstante Einflussfaktoren bleiben während der Datenaufzeichnung konstant bzw. wurden konstant gehalten (Checkpunkt B5). Dies betrifft den Lagertyp (L1), die Schmierung (L8), den Sensortyp (S1), dessen Leistungsfähigkeit (S2) und Postion (S3), die Datenaufnahme mit ADC (D1), Abtastrate (D2), Synchronisation (D3) und Auflösung (D4), sowie das eingesetzte Werkzeug zum Montieren bzw. Demontieren (A2).

- **Variierbar** (grün): Variierbare Einflussfaktoren werden kontrolliert während der Messungen verändert. Hierzu zählen die auf das Lager wirkende Last (L4), die Rotationsgeschwindigkeit (L5) und die Position des Lagers (L7).
- **Störgröße** (rot): Störgrößen können nur unter verhältnismäßig hohem technischen Aufwand konstant gehalten oder kontrolliert werden. Als Störgrößen wurden die (Hersteller-)Toleranzen (L2), der Zustand (L3) und Einbau des Lagers (L6), die Montage des Sensors (S4), die Erfahrung der (Projekt-)Arbeiter (A1), Vibrationen (U1), die Temperatur (U2), die Luftfeuchte (U3) und die Sauberkeit (U4) eingestuft. Störgrößen können ggf. auch als konstant oder variierbar angesehen werden. So bleiben z.B. U2 und U3 in klimatisierten Räumen nahezu konstant.

Gemäß Abschnitt 2.3.4 können Einflussgrößen weiterhin in kontrollierbare (konstante und variable Einflussgrößen) und unkontrollierbare Einflüsse (Störgrößen) unterteilt werden. Insbesondere die unkontrollierbaren Einflüsse können die Robustheit und Transferierbarkeit von ML-Modellen negativ beeinflussen und sollten daher minimiert werden (Checkpunkt B5) oder in der Datengrundlage hinreichend enthalten sein.

Anhand der ermittelten Einflussfaktoren wurde ein Versuchsplan erstellt. Dieser sieht die Vermessung von vier Lagern (Berücksichtigung L2) an vier Positionen (Berücksichtigung L7) in den drei Zuständen unbeschädigt, kleiner Schaden und großer Schaden (Berücksichtigung L3) vor. Jede Konfiguration wurde drei Mal in sog. Runs vermessen. Ein Run entspricht dabei dem sequentiellen Einbau an allen vier Positionen (Berücksichtigung L6 und S4). Weiterhin wurde jede Konfiguration in vier Laststufen (Berücksichtigung L4) und sechs Rotationsgeschwindigkeitsstufen (Berücksichtigung L5) vermessen. Die Last- bzw. Rotationsgeschwindigkeitsstufen wurden unter Berücksichtigung der limitierenden Bauteile (Festlager/Kupplung) bestimmt. Ein Ausschnitt des Versuchsplans ist in Anhang A.2.3 dargestellt.

Aufgrund der Variation der Last (L4) wurde ein zusätzlicher Kraftsensor in den Messaufbau integriert (Checkpunkt B6). Weiterhin wurde die Rotationsgeschwindigkeit (L5) über den Motorcontroller ausgelesen. Eine vollständige Übersicht findet sich in Anhang A.2.2 und deren Implementierung in PIA in Anhang A.2.1, Abbildung A.7. Weiterhin wird über ein externes Hygro-Thermometer die Temperatur (U2) und Luftfeuchte (U3) vermessen.

5.2.2.2 Normen und Standards

Im nächsten Schritt wurden Standards und Normen recherchiert (Checkpunkt B9). Relevante Beispiele hierzu sind:

- **Condition Monitoring Praxis** [98]: Handbuch, mit praxisorientiertem Expertenwissen zur Schwingungs- und Zustandsüberwachung von Maschinen und Anlagen.
- **ISO 20816-1:2016-11** [161]: Internationale Norm, mit einer allgemeinen Anleitung über die Messung und Bewertung von mechanischen Schwingungen.
- **BS ISO 7919-3+A1:2009-03-31** [162]: Internationale Norm für die Bewertung mechanischer Schwingungen an rotierenden Wellen bzw. gekoppelter industrieller Maschinen.

Eine Übersicht weiterer relevanter Normen findet sich in [98]. Im Rahmen des ersten Anwendungsfalls wurden die Empfehlungen des Handbuchs Condition Monitoring Praxis umgesetzt [98]. Diese beinhalten u.a. die Art der Befestigung des Beschleunigungssensors, die Messdauer und die Richtung der Lastaufbringung.

5.2.2.3 Messunsicherheiten

Die Bestimmung der Quellen von Messunsicherheiten (Checkpunkt B10) konnte durch die bereits erstellte und in PIA integrierte Sensorübersicht (vgl. Anhang A.2.2) erfolgen. Als relevante Quelle der Messunsicherheit wurde der Beschleunigungssensor identifiziert, da dieser das primäre Messsignal für die Datenanalyse aufzeichnet. Tabelle 5.2 zeigt die Messunsicherheit des Beschleunigungssensors (Checkpunkt B11) gemäß Kalibrierungszertifikat.

Tabelle 5.2: Messunsicherheiten des Beschleunigungssensors aus dem Kalibrierungszertifikat.

Sensor	**Unsicherheit** [%]
X-Achse	2.3
Y-Achse	3.6
Z-Achse	2.7

Am Kraftsensor, Motorcontroller und Hygro-Thermometer treten ebenfalls Messunsicherheiten auf, jedoch wurden diese lediglich zu Steuerungs- und Regelungszwecken

der Störgrößen Last und Rotationsgeschwindigkeit bzw. zur Überwachung der Raumtemperatur und Luftfeuchte verwendet und sind daher von untergeordneter Priorität.

5.2.2.4 Aufbau der Daten

Das für die Datenerfassung verwendete cRIO-System des Messkoffers zeichnet Daten standardmäßig im TDMS-Format auf. Da dieses sich für große Datenströme eignet, von einer Vielzahl von Programmen eingelesen (wie z.B. MS Excel oder MATLAB), sowie um Metadaten ergänzt werden kann, wurde es als Datenformat beibehalten (Checkpunkt B12). Dabei wurde für jede Prüfstands-Konfiguration eine Datei erstellt (Checkpunkt B13). Jede dieser Dateien wurde nach der Syntax Schadensposition_-LagerNr._Beschädigung_Lauf_Position_Kraft_Geschwindigkeit_Arbeiter.tdms, z.B. Inner-Ring_B10_DNoD_R1_PA_F0_S706_W1.tdms, eindeutig (Checkpunkt B14) und beschreibend (Checkpunkt B15) benannt. Neben den ausgewählten Metadaten (Checkpunkt 17 und 18), welche zusätzlich im Dateinamen integriert sind und automatisch annotiert werden (Checkpunkt B19), wird manuell ein Foto der aktuellen Prüfstands-Konfiguration aufgenommen (Checkpunkt B25). Anhand der Fotos der jeweiligen Prüfstands-Konfiguration konnten im Nachgang u.a. Montagefehler des Sensors, der Kupplung sowie beider Wellen identifiziert werden. Abbildung 5.8 zeigt eine Prüfstand-Konfiguration, bei der die Kupplung außermittig (nach rechts verschoben, rote Kästen) montiert wurde.

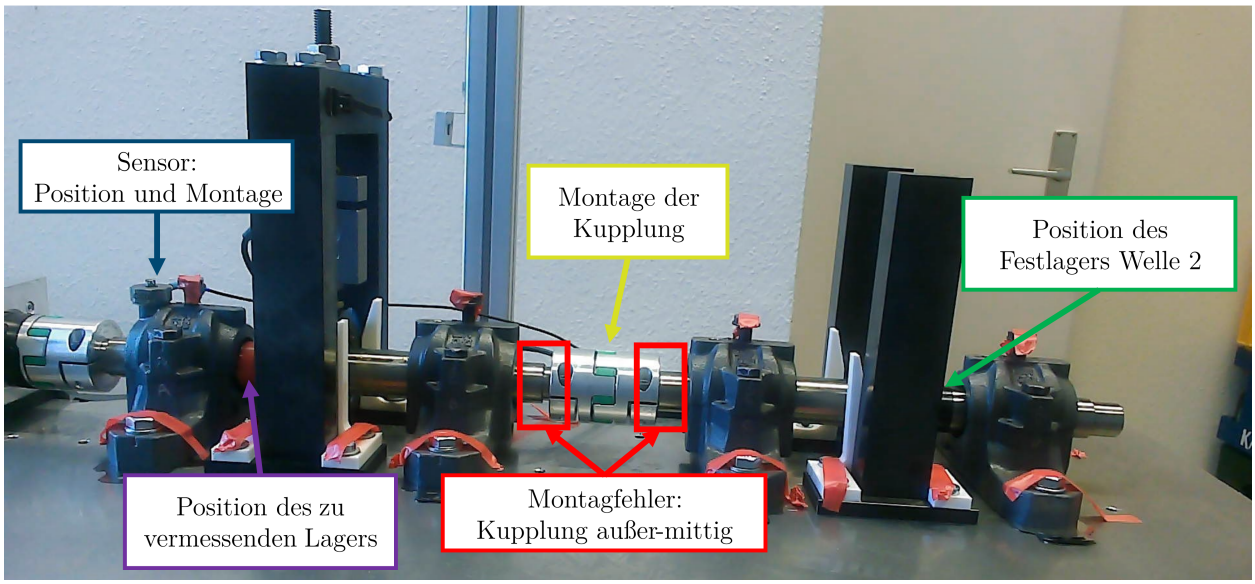


Abbildung 5.8: Foto einer Prüfstand-Konfiguration mit fehlerhaft montierter Kupplung.

Jede TDMS-Datei enthält zudem pro Kanal einen Zeitstempel im Format DD-MMM-YYYY HH:MM:SS der Zeitzone Universal Time Coordinated (UTC) (Checkpunkt B20/B21). Interne Prüfprozesse wurden von den Projektmitarbeitern entsprechend gekennzeichnet und anschließend separat gespeichert (Checkpunkt B22).

Für die nachfolgende Datenanalyse bzw. Interpretation der Ergebnisse wurde die Innenring-Überrollfrequenz (BPFI) als relevante Kennzahl identifiziert (Checkpunkt B24) [98]. Es gilt

$$\mathrm{BPFI} = \frac{1}{2} \cdot f_n \cdot z \left[1 + \frac{D_w}{D_T} \cdot \cos(\alpha)\right] \tag{5.1}$$

mit der Drehfrequenz f_n, der Wälzkörperanzahl (pro Reihe) z, dem Wälzkörperdurchmesser D_w, dem Teilkreisdurchmesser D_T und dem Druckwinkel α [98]. Tabelle 5.3 zeigt die zur Berechnung erforderlichen geometrischen Details des Zylinderrollenlagers vom Typ NU206-E-XL-TVP2 [163].

Tabelle 5.3: Geometrisch relevante Details eine Zylinderrollenlagers des Typs NU206-E-XL-TVP2 zur Berechnung der Überrollfrequenz am Innenring [163].

Variable	z	D_w	D_T	α
Wert	13	9 mm	46,5 mm	0°

5.2.2.5 Datenablage

Neben dem Messkoffer wurde ein externes Hygro-Thermometer zur Überwachung der Temperatur und Luftfeuchtigkeit verwendet. Das Zusammenführen der beiden Systeme (Checkpunkt B26) erfolgte über den Zeitstempel, da die Daten des Hygro-Thermometers nicht über den Messkoffer abgegriffen werden konnten. Anhand der Erfahrungen der Daten der Vorstudie [160] konnte der voraussichtliche Speicherbedarf auf ca. 73 GB geschätzt werden (Checkpunkt B27). Dieser Speicherbedarf ergibt sich aus den geplanten 2592 Messungen mit einer konstanten Dateigröße von 23,5 MB sowie einem 20 % Puffer.

Die Dateiablage erfolgte auf dem Server des Lehrstuhls für Messtechnik (Checkpunkt B28) mit Zugriff für alle wissenschaftlichen Projektmitarbeiter (Checkpunkt B29). Neben dem RAID-gesicherten Server wurden die Daten zusätzlich in der Cloud und auf einer lokalen Festplatte gesichert (Checkpunkt B30).

5.2.2.6 Einbindung manueller Datenquellen

Manuelle Datenquellen sind in der Messkampagne der Versuchsplan, die Fotos der Prüfstands-Konfiguration (Checkpunkt B35) und die daraus abgeleiteten Metadaten, wie z.B. Montagefehler. Um den menschlichen Einfluss auf die Daten zu reduzieren, wurden die im UWD ermittelten Einflussgrößen Erfahrung (A1) und Werkzeug (A2) näher betrachtet. Diese Einflussgrößen wurden durch die Verwendung von bereits voreingestellten Drehmomentschlüsseln zur Montage der Kupplungen, des Beschleunigungssensors und der Lagerdeckel reduziert. Weiterhin wurden alle nicht relevanten Verschraubungen abgeklebt (vgl. Abbildung 5.8) und die Hilfswissenschaftler im Umbau des Prüfstandes geschult.

Im Forschungskontext ist die Verwendung von (digitalen) Laborbüchern wie z.B. das eLabFTW [164] anstelle von (digitalen) Schichtbüchern üblich (Checkpunkt B32). Da zum Zeitpunkt der Messkampagne weder ein digitales Schichtbuch noch ein digitales Laborbuch in der bestehenden Infrastruktur implementiert war, wurde aus zeitlichen Gründen auf eine Einführung verzichtet und stattdessen der Versuchsplan verwendet, um Auffälligkeiten bzw. Unstimmigkeiten zu dokumentieren. Die Kommentarspalte des Versuchsplans erlaubt den Hilfswissenschaftlern Eingriffe, Änderungen (Checkpunkt B33) und weitere Auffälligkeiten (Checkpunkt B34) zu dokumentieren. Anhand der eindeutigen Zuordenbarkeit können diese im Nachgang den entsprechenden Messdaten zugeordnet werden (Checkpunkt B36). Ein Ausschnitt des Versuchsplans findet sich in Anhang A.2.3.

5.2.3 Datenaufnahme

Nachdem die am Messkoffer angeschlossenen Sensoren am Prüfstand installiert und die Datenaufnahme getestet wurden, sind die notwendigen Vorbereitungen für eine Datenaufnahme abgeschlossen (Checkpunkt C1).

5.2.3.1 Erste Testmessung und Überprüfung der Datenqualität

Im Anschluss an Checkpunkt C1 konnten erste Testmessungen erfolgen (Checkpunkt C2). Hierzu wurde ein unbeschädigtes Lager an den Positionen A bis D unter zwei Last- und sechs Geschwindigkeitsstufen vermessen. So konnten einerseits die Datenstruktur überprüft (Checkpunkt C3) und andererseits die Hilfswissenschaftler bei den Umbaumaßnahmen am Prüfstand angeleitet werden. Die Messdauer betrug 60 s mit einer Abtastrate von 20 kHz. Während der Testmessungen erlaubte die Live-Ansicht des Messkoffers die Überprüfung einer ordnungsgemäßen Funktion der

Sensoren (Checkpunkt C4). Weiterhin wurden die Synchronität der in der TDMS-Datei gespeicherten Zeitstempel (Checkpunkt C5) und die Zuordenbarkeit der Daten (Checkpunkt C6) erfolgreich geprüft.

Eine erste Beurteilung der Datenqualität (Checkpunkt C7) der Testdaten mittels der 15 IQ-Dimensionen (vgl. Abschnitt 2.4.3.1) findet sich in Tabelle 5.4.

Tabelle 5.4: Erste Bewertung der Datenqualität mittels der 15 IQ-Dimensionen in positiv und negativ (nach [67]).

Nr.	**IQ-Dimension**	**Bewertung**	**Anmerkung**
IQ-1	Zugänglichkeit	positiv	Unkompliziert einlesbar
IQ-2	Angemessener Umfang	positiv	Hinsichtlich Messdauer und Metadaten
IQ-3	Glaubwürdigkeit	positiv	Verwendung hochwertiger Messtechnik
IQ-4	Vollständigkeit	positiv	Hinsichtlich der Messreihen pro Sensor
IQ-5	**Übersichtlichkeit**	**negativ**	**Struktur der Metadaten unübersichtlich**
IQ-6	Einheitliche Darstellung	positiv	Daten und Metadaten einheitlich
IQ-7	Bearbeitbarkeit	positiv	Nach dem Einlesen leicht bearbeitbar
IQ-8	Fehlerfreiheit	positiv	Unter Berücksichtigung einer verbleibenden Messunsicherheit
IQ-9	**Eindeutige Auslegbarkeit**	**negativ**	**Die Messdaten von Kraft und Geschwindigkeit liegen in der Einheit Volt vor**
IQ-10	Objektivität	positiv	Unbearbeitete Rohdaten.
IQ-11	Relevanz	positiv	Relevant zur Beurteilung der Datenqualität
IQ-12	Hohes Ansehen	positiv	Datenerfassungssystem von NI kann als vertrauenswürdig eingestuft werden
IQ-13	Aktualität	positiv	Daten aktuell, da Aufzeichnung unmittelbar vor Messkampagne
IQ-14	**Verständlichkeit**	**negativ**	**Metadaten unverständlich**
IQ-15	Wertschöpfung	positiv	Erlaubt die Einschätzung der Daten vor Start der Messungen

Anhand der ersten Beurteilung der Datenqualität konnten Verbesserungspotenziale in den Dimensionen Übersichtlichkeit (IQ-5), Eindeutigen Auslegbarkeit (IQ-9) und Verständlichkeit (IQ-14) aufgedeckt werden. Zunächst wurde zur Verbesserung von IQ-9 in das Skript zum Einlesen der Daten die Umwandlung der Spannungsmesswerte der Last in [N] und Rotationsgeschwindigkeit in [1/min] integriert. Zur Verbesserung von IQ-5 und IQ-14 wurden die Messungen um eine Metadaten-Datei ergänzt. Diese enthält den jeweiligen Dateinamen der Messung, deren Dateipfad und Zeitstempel, die Lagernummer, den Schadensfall, den Run, die Position, die Last [N], die Geschwindigkeit [1/min], den Messtag, die Batch-Nr., die Temperatur [°C] und rel. Luftfeuchte [%] sowie die Schadensdimensionen [mm] und den messenden Projektmitarbeiter.

Da die anschließende Messkampagne lediglich einmalig durchgeführt wurde, wurde auf die Implementierung einer automatischen Überprüfung der Datenqualität verzichtet (Checkpunkt C8) und stattdessen eine erneute Überprüfung der Datenqualität nach der Datenprüfung und Bereinigung bzw. Datenvisualisierung durchgeführt.

5.2.3.2 Langzeitdatenaufnahme

Vor Start der Langzeitdatenaufnahme wurde zunächst die regelmäßige Prüfung der Vollständigkeit (IQ-4) und Fehlerfreiheit (IQ-8) nach jedem Messtag (vgl. Versuchsplan Anhang A.2.3) durch die Hilfswissenschaftler und nach jedem Lager durch den Verantwortlichen festgelegt (Checkpunkt C9). Anschließend wurde die Messkampagne gestartet (Checkpunkt C10). Aufgrund der kurzen Dauer der Messkampagne entfällt die regelmäßige Prüfung der Datenverteilung (Checkpunkt C11).

Während der Langzeitdatenaufnahme erwies sich insbesondere die regelmäßige Prüfung als sinnvoll, da so nachfolgende Fehler frühzeitig aufgedeckt wurden:

- **Ausfall des Hygro-Thermometers**: Nach Messtag 1 wurde festgestellt, dass das Hygro-Thermometers ausgefallen war. Der Fehler konnte frühzeitig behoben werden. Aufgrund der niedrigen Relevanz des Hygro-Thermometers (keine starken Schwankungen der Temperatur und Luftfeuchte im Messraum) wurden die Messungen nicht wiederholt.

- **Fehlerhafte Sensormontage des Beschleunigungssensors**: Nach Messtag 1 wurde außerdem festgestellt, dass ein Teil der Messungen mit invertierter Ausrichtung der Z-Achse erfolgten. Hier konnte der entsprechende Hilfswissenschaftler nachgeschult und der Fehler in zukünftigen Messungen vermieden werden.

- **Änderung der Versuchsreihenfolge**: Die Prüfung der Zeitstempel hinsichtlich IQ-8 ergab Unstimmigkeiten. Eine nähere Analyse ergab, dass Hilfswissenschaftler 2 statt der ursprünglichen Vorgabe (Lauf 1: Pos. A-D, Lauf 2: Pos. A-D, Lauf 3: Pos. A-D) die Messreihenfolge zu Pos. A: Lauf 1-3, Pos. B: Lauf 1-3, Pos. C: Lauf 1-3, Pos. D: Lauf 1-3 geändert hat, um die Anzahl notwendiger Prüfstandsumbaumaßnahmen zu reduzieren. Obwohl die Daten nach IQ-4 vollständig sind, sind diese nach IQ-8 nicht fehlerfrei, da die Störgröße der Montage des Lagers (L6) nicht mehr in den Daten enthalten ist. Die fehlerhaften Messungen wurden an einem neuen Lager wiederholt.

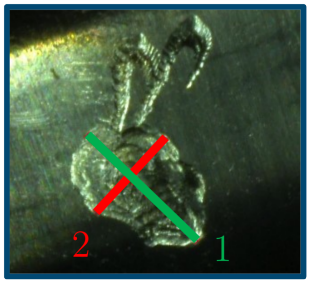
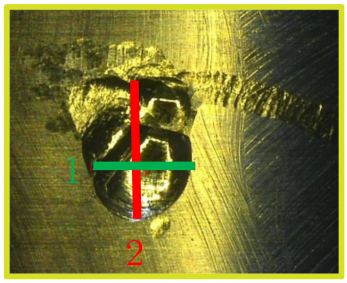
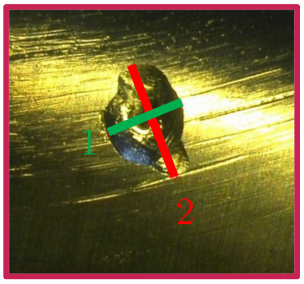
5.2.4 Datenprüfung und Datenbereinigung

Nach Abschluss der Messkampagne wurde zunächst die Datenstruktur erneut überprüft (Checkpunkt D1) und keine Auffälligkeiten festgestellt. Anschließend wurde der Datensatz um folgende Metadaten ergänzt (Checkpunkt D2):

- **Montagefehler**: Anhand der zusätzlich aufgezeichneten Bilder konnten Fehler bei der Montage des Beschleunigungssensors, der mittleren Kupplung (links-zentriert, rechts-zentriert und Winkelversatz zur vorderen Kupplung) und der zweiten Welle (Lagerpositionen vertauscht) identifiziert werden. Die Montagefehler wurden in der Metadaten-Datei entsprechend annotiert.

- **Schadensdimensionen**: Mittels Auflichtmikroskopie wurden die Maße der künstlichen Beschädigung der Innenringe vermessen (vgl. Abbildung 5.9). Die Schadensdimensionen wurden ebenfalls in der Metadaten-Datei ergänzt und die Aufnahmen des Auflichtmikroskopes dem Datensatz beigefügt.

a)

	Lager 10	Lager 20	Lager 30
			
1	3.29 mm	2.22 mm	1.69 mm
2	2.20 mm	2.94 mm	2.55 mm

b)

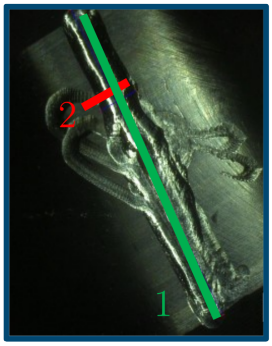
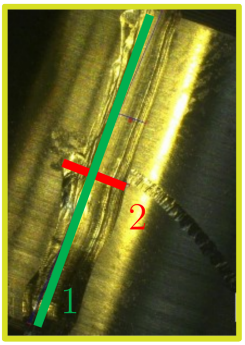
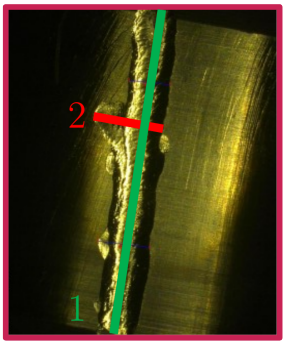
			
1	8.08 mm	11.96 mm	12.03 mm
2	1.03 mm	2.35 mm	2.51 mm

Abbildung 5.9: Innenring mit Abmaßen **a)** der kleinen und **b)** großen Beschädigung.

Weiterhin umfasste die Bereinigung des Datensatzes die nachfolgenden Schritte:

- **Zusammenführung der Daten** (Checkpunkt D3): Die Daten des Hygro-Thermometers wurden mit den Daten des Messkoffers über den Zeitstempel zusammengeführt.
- **Identifikation blinder Flecken** (Checkpunkt D4): Als blinder Fleck in der Messkampagne konnte der Montageprozess der Hilfswissenschaftler identifiziert werden, da dieser nicht messtechnisch erfasst werden konnte. Jedoch erlaubten die Bildaufnahmen der Prüfstands-Konfigurationen die Identifikation der Montagefehler (s.o.).
- **Bereinigung des Datensatzes** (Checkpunkt D5): Im nächsten Schritt wurde der Datensatz von fehlerhaften Messungen wie z.B. Mehrfach-Messungen der gleichen Prüfstand-Konfiguration bereinigt.
- **Anpassung der Bezugssysteme** (Checkpunkt D6): Eine Anpassung der Bezugssysteme war aufgrund der getroffenen Maßnahmen in Checkpunkt B21 nicht erforderlich.

- **Separierung von Testmessungen und Entfernung von Ausreißern** (Checkpunkt D7/D8): Unter Verwendung des Notizfelds im Versuchsplan konnten die Testmessungen separiert und Ausreißer entfernt werden.
- **Prüfung der Einheiten** (Checkpunkt D9): Die Prüfung der Einheiten auf Konsistenz und Richtigkeit ergab keine Auffälligkeiten.
- **Ausgleich von Sensordrift** (Checkpunkt D10): Aufgrund der vergleichsweise kurzen Dauer der Messkampagne wurden keine Daten- oder Sensordrifts erwartet und konnten auch nicht identifiziert werden.

Für die abschließende Beurteilung der Datenqualität wurden zunächst erneut die 15 IQ-Dimensionen betrachtet. Tabelle 5.5 zeigt Veränderungen in den Dimensionen im Vergleich zu Tabelle 5.4.

Tabelle 5.5: Aktualisierte Bewertung der Datenqualität.

Nr.	**IQ-Dimension**	**Bewertung**	**Anmerkung**
IQ-5	Übersichtlichkeit	positiv	Durch angepasste Struktur der Metadaten.
IQ-9	Eindeutige Auslegbarkeit	positiv	Durch Integration der Einheiten in die Messdaten-Datei.
IQ-14	Verständlichkeit	positiv	Durch Anpassung der Darstellung der Metadaten.

Die zuvor negativ bewerteten Dimensionen IQ-5, IQ-9 und IQ-14 konnten durch die Anpassungen nach den ersten Testmessungen nun positiv bewertet werden.

Weiterhin wurde die FAIRness der Daten anhand der FAIR-Indikatoren [165] beurteilt und in Abbildung 5.10 dargestellt.

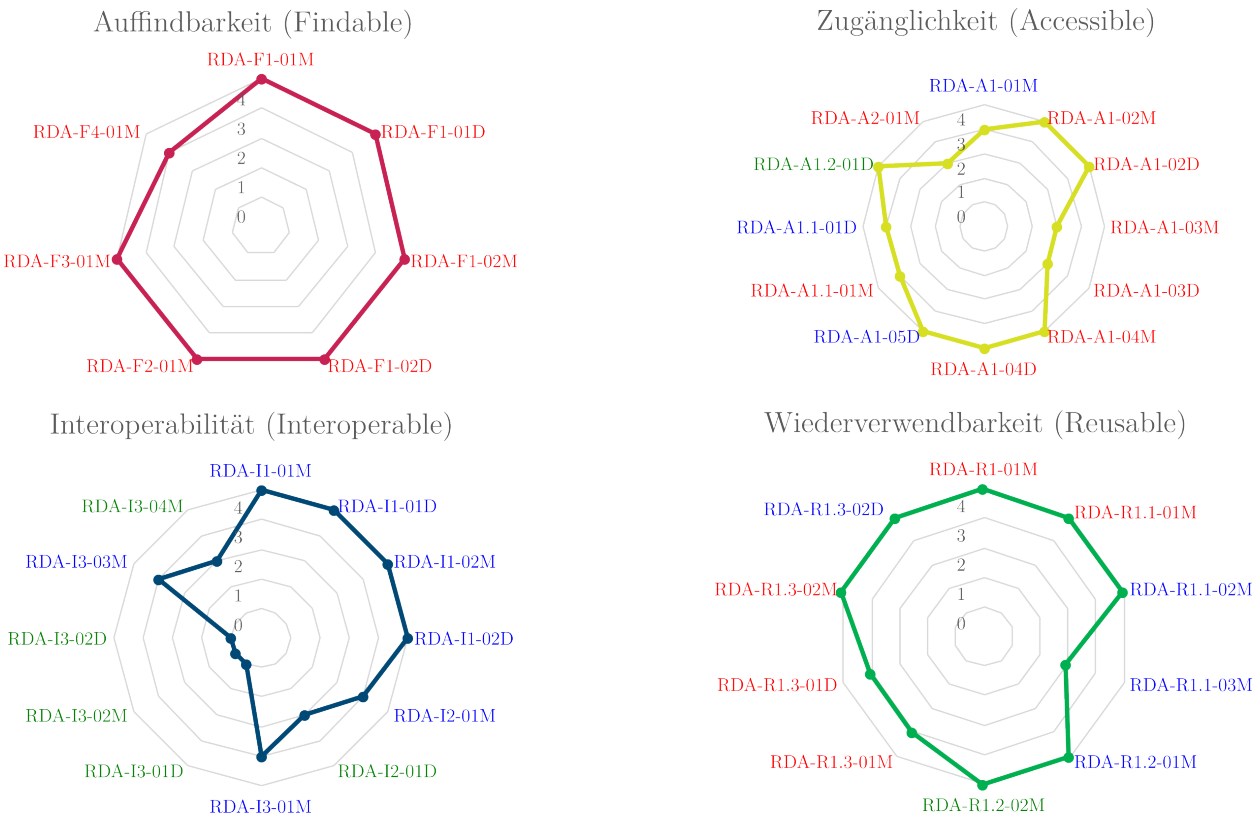


Abbildung 5.10: FAIRness der Daten nach [165], dargestellt als Spiderplots, mit den jeweiligen FAIR-Indikatoren (vgl. Anhang A.1.2 und deren Prioritäten essentiell (rot), wichtig (blau) und nützlich (grün).

Insbesondere in den Dimensionen Interoperabilität und Zugänglichkeit zeigen sich Optimierungspotenziale. Jedoch ist anzumerken, dass die niedrig bewerteten Indikatoren größtenteils einer niedrigen Priorität zugeordnet sind und die essentiellen Indikatoren zu 87,5 % erfüllt sind. Eine Beschreibung der Indikatoren findet sich in Anhang A.1.2.

Letztlich erfolgt die quantitative Beurteilung der Datenqualität nach Heinrich [66] (vgl. Abschnitt 2.4.3.2) in Tabelle 5.6.

Tabelle 5.6: Bewertung der Datenqualität nach Heinrich et al. [66].

Dimension	%	**Anmerkung**
Vollständigkeit	99,96 %	Von geplanten 2592 Messungen wurden 2591 durchgeführt.
Fehlerfreiheit	96,18 %	Fehlerfreiheit der Rotationsgeschwindigkeit als einziger geregelter Parameter.
Konsistenz	85,37 %	Gemäß der definierten Regel: Messwerte des Beschleunigungssensors sind innerhalb des gültigen Messbereiches.
Aktualität	100 %	Primär relevant für Daueranwendung des Modells bzw. Anlagen mit voranschreitendem Verschleiß.

Hier zeigt die Dimension *Konsistenz* mit 85,18 % ein deutliches Optimierungspotenzial. Gemäß der Konsistenzregel sind Messwerte des Beschleunigungssensors inkonsistent, sobald ein Messwert den gültigen Messbereich über- bzw. unterschreitet. Dies deutet auf einen zu geringen Messbereich des Beschleunigungssensors hin. Weiterhin reduzierten Spannungseinbrüche des Messmoduls die Fehlerfreiheit auf 96,18 %.

Abschließend wurden die durchgeführten Schritte der Datenprüfung und -bereinigung dokumentiert (Checkpunkt D11).

5.2.5 Datenauswertung und Modellbildung

Die nachfolgende Datenauswertung und Modellbildung erfolgte mit der in PIA integrierten MathWorks® MATLAB Online Version der ML-Toolbox bzw. Uncertainty-Aware Automated Machine Learning Toolbox (UA-ML-Toolbox). Da für den vollständigen Datensatz kein zufriedenstellendes Ergebnis erzielt werden konnte, wurde zur Veranschaulichung der Methodik und Vorgehensweise ein reduzierter Datensatz verwendet:

- **Konstante Rotationsgeschwindigkeit**: Die Daten enthalten nur Messungen mit einer Rotationsgeschwindigkeit 969 1/min.
- **Entfernung lastfreier Zustand**: Zylinderrollenlager benötigen laut Herstellerangaben eine Mindestbelastung, um Beschädigungen durch Schlupf zu vermeiden [166]. Daher werden die lastfreien Messungen nicht berücksichtigt.

- **Entfernung großer Schaden**: Da ein großer Lagerschaden im Messsignal deutlich erkennbar ist (siehe Abschnitt 5.2.5.1), wird dieser Schadensfall für die Modellbildung nicht berücksichtigt.

5.2.5.1 Datenvisualisierung und Datenverständnis

Nachdem die Datenprüfung und Datenbereinigung abgeschlossen wurden, konnte mit der Visualisierung der Daten sowie dem Aufbau von weiterem Datenverständnis begonnen werden. Im ersten Schritt wurde hierzu ein Zeitabschnitt von 150'000 Samples in der Mitte einer jeden Messung zur Visualisierung gewählt (Checkpunkt E1) und diese anschließend als Zeitreihendiagramm geplottet (Checkpunkt E2). Die Anzahl der Samples wurde gemäß [98] so gewählt, dass bei der niedrigsten Rotationsgeschwindigkeit mindestens zehn volle Umdrehungen des Innenrings in den Messdaten enthalten waren. Abbildung 5.11 zeigt drei beispielhafte Messungen eines Lagers ohne Beschädigung, mit einer kleinen Beschädigung und einer großen Beschädigung des Innenrings, jeweils bei konstanter Rotationsgeschwindigkeit und Last.

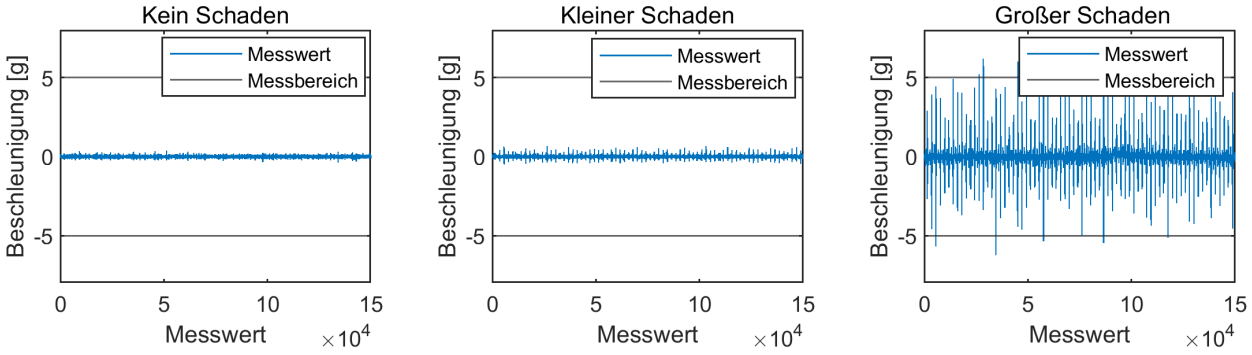


Abbildung 5.11: Darstellung des Messsignals mit den drei Schadenszuständen kein Schaden, kleiner Schaden und großer Schaden.

Durch die Visualisierung der Messsignale mit ihrem Messbereich sind erste Überschreitungen des Messbereichs bei einer großen Beschädigung erkennbar.

Eine weitere Auffälligkeit ist in Abbildung 5.12 a) zu erkennen, welche Messsignale des Kraftsensors und des Motorcontrollers zeigen.

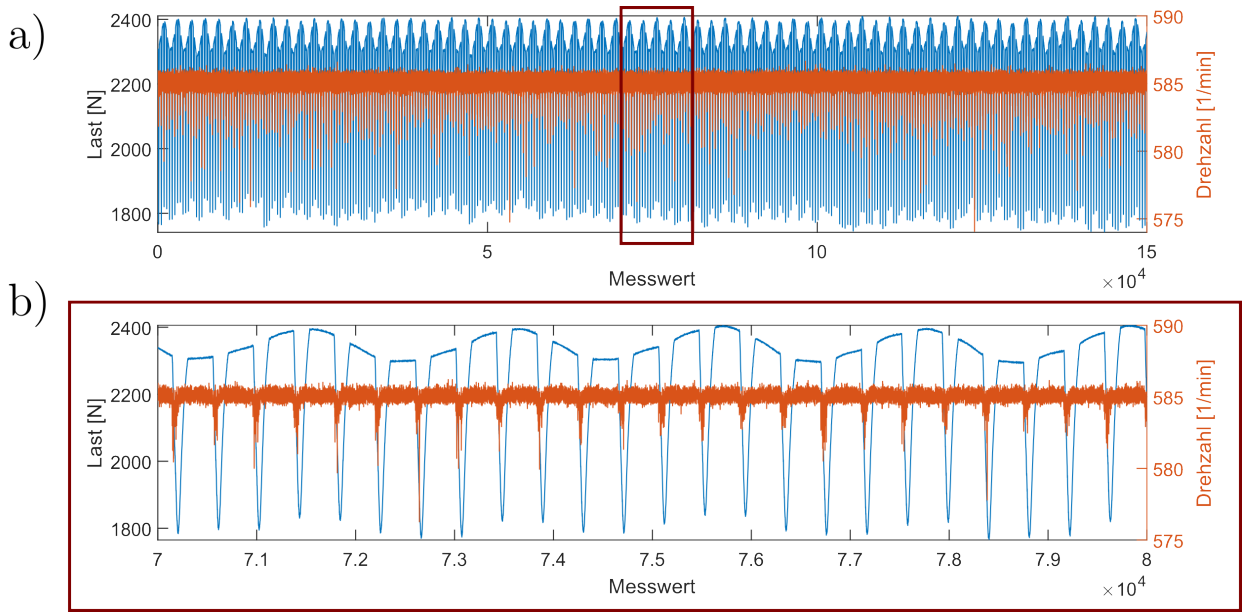


Abbildung 5.12: **a)** Darstellung des Messsignals des Last (blau) und der Rotationsgeschwindigkeit (orange). **b)** Ausschnitt der Messsignale in welchem systematische (Spannungs-)Einbrüche zu erkennen sind.

Hier bricht systematisch und zeitgleich das Messsignal der Last und Rotationsgeschwindigkeit ein (vgl. Abbildung 5.12, b), was auf einen Defekt im Messmodul hindeuten könnte. Dies verdeutlicht die Relevanz einer kontinuierlichen Aufzeichnung (vgl. Tipp in Kapitel 4.1 der Checkliste), da z.B. die Speicherung von Mittelwerten hier einerseits den Fehler verschleiern und andererseits die Mittelwerte durch die Ausreißer systematisch zu niedrig ausfallen. Weiterhin reduzieren die betroffenen Messungen die Datenqualität hinsichtlich IQ-3 (Glaubwürdigkeit) und IQ-8 (Fehlerfreiheit).

Im nächsten Schritt erfolgte die Analyse der quasi-statischen Signale (Checkpunkt E3). Das Plotten des quasi-statischen Signals eignet sich primär für Messsignale mit definierten Bezugspunkten, welche bei den Messungen mit dem Beschleunigungssensor aufgrund der unbekannten Winkelposition der Welle nicht gegeben sind. Allerdings kann z.B. das quasi-statische Signal der Mittelwerte der Messungen geplottet werden. Abbildung 5.13 a) zeigt das quasi-statische Signal der berechneten Mittelwerte (blau) und die Schadensstufe (orange) der Messungen.

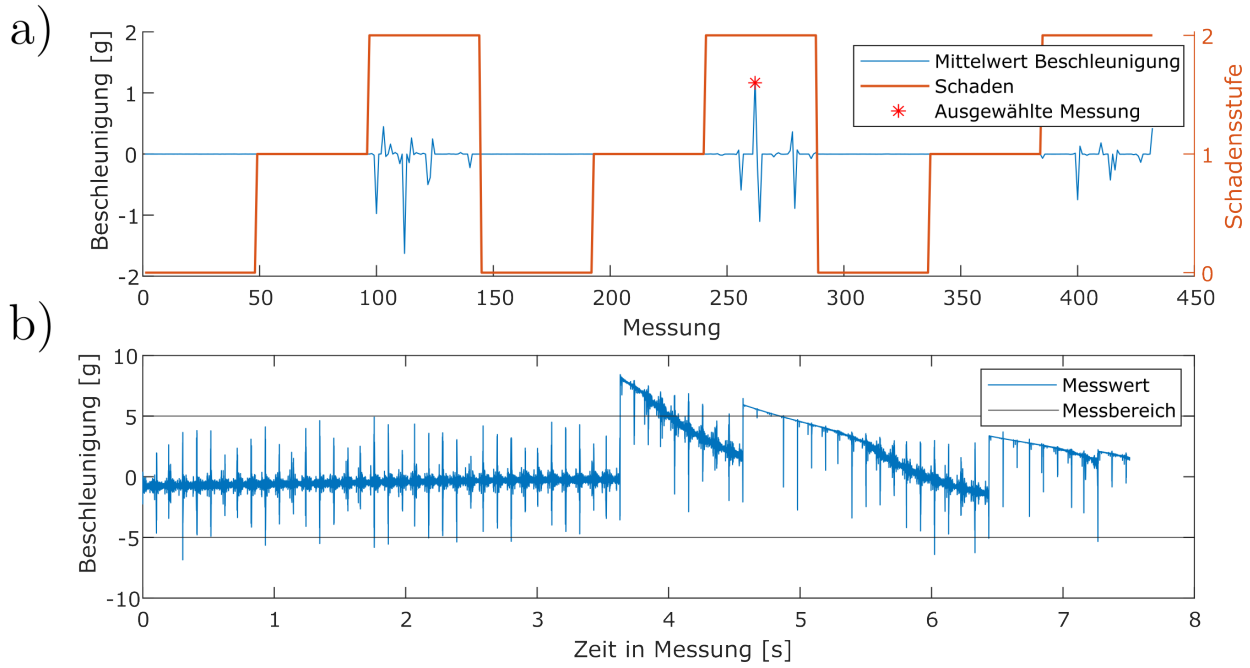


Abbildung 5.13: **a)** Quasi-statisches Signal der Mittelwerte (pro Messung) des Beschleunigungssensors. **b)** Ausgewählte Messung aus a), welche Anomalien aufweisen.

Hier zeigen sich Auffälligkeiten bei mehreren Messungen mit einer großen Beschädigung des Innenrings, da der Mittelwert stark von Null abweicht. In Abbildung 5.13 b) wird eine der betroffenen Messungen dargestellt. Die Verschiebung des Mittelwertes wird durch eine signifikante Überschreitung des Messbereichs ausgelöst und resultiert in einem Einlauf-ähnlichem Verhalten.

Die Checkpunkte E4 (Histogramm) und E5 (Boxplot-Diagramm) sind für den primären Einsatz in der Industrie relevant, da sie dem Anwender eine schnelle Beurteilung der Verteilung und einen Vergleich von bspw. Prozessen erlauben. Sie wurden aus Gründen der Übersichtlichkeit in Anhang A.2.4 ausgelagert. Abbildung A.11 zeigt die Histogramme der in Abbildung 5.11 dargestellten Messsignale. Die einzelnen Messwerte sind normalverteilt und weisen keine bi- oder multimodale Verteilungen auf. Die in Abbildung A.12 dargestellten Boxplot-Diagramme, welche ebenfalls aus den Messsignalen in Abbildung 5.11 abgeleitet wurden, deuten auf eine größer werdende Streuung der Messwerte bei steigender Schadensdimension hin.

Anschließend erfolgte eine Hauptkomponentenanalyse (PCA) der Daten des Beschleunigungssensors (Checkpunkt E6). Da auf den Rohdaten keine sichtbaren Cluster der Ziel- bzw. Störgrößen erkennbar waren, wurde die Merkmalsextraktionsmethode Beste Fourier Koeffizienten (BFC) der PCA vorgeschaltet. Abbildung 5.14 zeigt jeweils die ersten zwei Hauptkomponenten der PCA, welche nach **a)** Position, **b)** Lager und **c)** Lauf eingefärbt wurden.

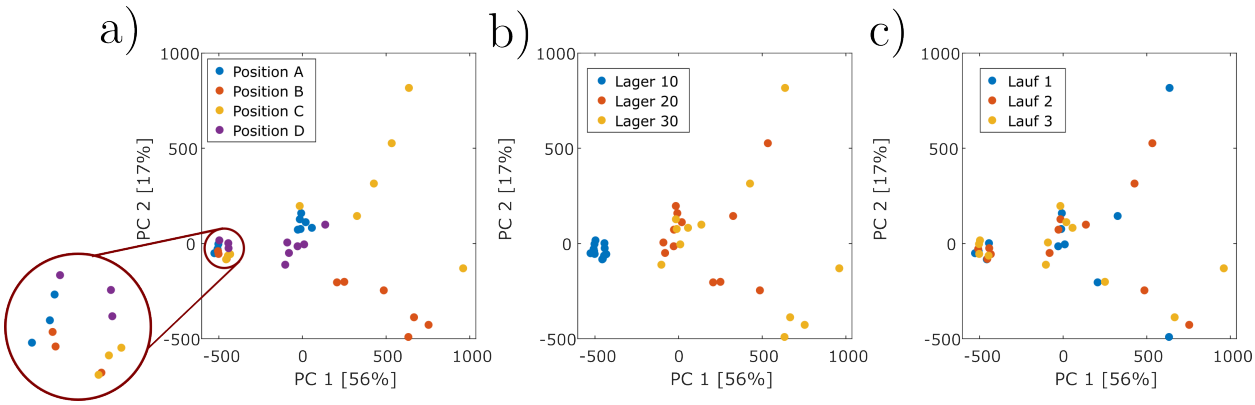


Abbildung 5.14: PCA der Merkmale des BFC-Extraktors eingefärbt nach **a)** Position, **b)** Lager und **c)** Lauf.

Insbesondere Abbildung 5.14 a) verdeutlicht den Einfluss der verschiedenen Positionen und die Relevanz der Berücksichtigung von Störgrößen in den Messdaten. Mögliche Ursachen der Clusterbildung könnten positionsspezifische Frequenzen wie z.B. Eigenfrequenzen oder positionsrelevante Störgrößen wie z.B. Schwingungen vom Antriebsmotor sein (Checkpunkt E7).

5.2.5.2 Auswahl maschineller Lernalgorithmen

Vor der Anwendung der ML-Toolbox wurde zunächst der Stand der Technik hinsichtlich der Analyse von Wälzlagerschäden mittels ML überprüft (Checkpunkt E8). Als gängige Praxis gilt die Vorverarbeitung der Beschleunigungssignale durch Filterung mit anschließender Hüllkurvenbildung [98, 167]. Grundlage hierfür ist die Demodulierung von hochfrequenten Resonanzen, die durch entstehende Impulse beim Überrollen von Schäden erzeugt werden, wodurch die Schadensfrequenzen sichtbar werden [168]. Im Kontext des ML bezieht sich ein Großteil der Literatur auf die oben genannten Datensätze CWRU, PU und NASA-BD unter Anwendung von neuronalen Netzen (NN). Dies verdeutlicht ein Review-Paper [169] in dem 32 wissenschaftliche Publikationen des CWRU-Datensatzes mit verschiedenen Deep Learning Ansätzen analysiert werden. Nach einer Studie [92] können traditionelle ML-Methoden bei den PU und CWRU Daten im Leave-One-Group-Out-Cross-Validation (LOGOCV)-Szenario nahezu gleichwertige und teilweise bessere Resultate erreichen als die getesteten NN-Ansätze Convolutional Neural Networks (CNN), ResNet [170], WaveNet [171] und Multi-Layer Perceptron [172], bei tendenziell höherer Interpretierbarkeit und geringerer erforderlicher Rechenleistung. Für die Berechnungen im Anwendungsfall wurde ein Notebook mit einem Intel Core i7-10610U CPU mit 1.8 GHz und 4 Kernen verwendet (Checkpunkt E9).

Im nächsten Schritt wurde das Lernproblem (Checkpunkt E10) definiert. Analog zu den Erkenntnissen der Vorstudie [160] wurde das Klassifikationsproblem Lager gesund/Lager beschädigt ausgewählt. Weiterhin wurde die LOGOCV mit einer Gruppeneinteilung nach Einbauposition des Lagers als Validierungsszenario festgelegt (Checkpunkt E11). Abbildung 5.15 zeigt das schematische Vorgehen im gewählten Szenario. Zunächst wurden die Daten von Pos. D als Testdaten festgelegt. Anschließend erfolgte die Unterteilung der Daten in drei Folds, welche jeweils zwei Positionen im Training und eine dritte Position zur Validierung enthalten.

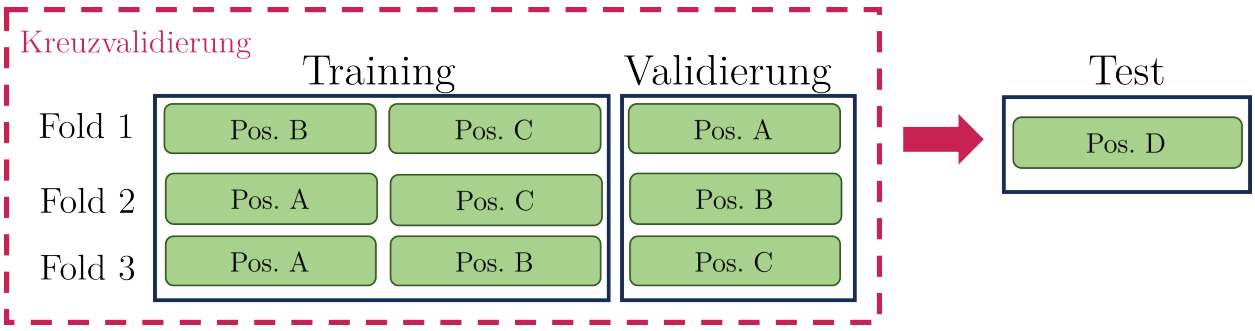


Abbildung 5.15: Darstellung des LOGOCV mit dem systematischen Auslassen einer Position (Pos.A bis Pos.C) in einer Kreuzvalidierung und dem anschließenden Test von Pos. D.

Gemäß der Literaturrecherche wurden die Daten vor der Analyse mit einem Bandpassfilter gefiltert und anschließend die Hüllkurve gebildet. Abbildung 5.16 a) zeigt einen Ausschnitt des Messsignals des beschädigten Lagers 10 in Lauf 1 an Pos. A unter Laststufe 3 mit der Rotationsgeschwindigkeit 969 1/min. Deutlich erkennbar sind Überrollungen der Beschädigung zwischen 0,15 s bis 0,16 s und 0,215 s bis 0,225 s. Weiterhin werden das gefilterte Signal (orange) und die mittels Hilbert-Transformation [173] gebildete Hüllkurve (gelb) visualisiert. Durch die Vorverarbeitung der Daten werden, verglichen mit der unbeschädigten Messung (gleiche Prüfstandskonfiguration), im Frequenzspektrum in Abbildung 5.16 b) die für einen Innenringschaden (blau) charakteristische BPFI sowie die Drehfrequenz f_n als Seitenbänder um die BPFI sichtbar [98].

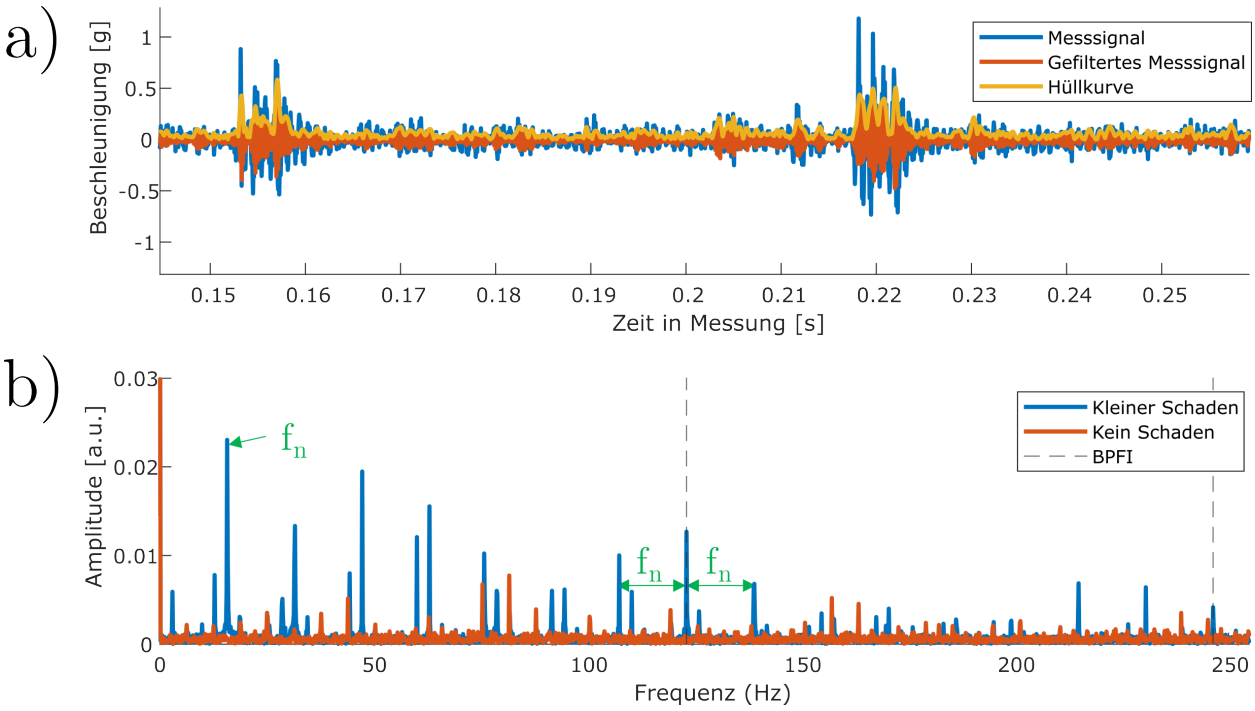


Abbildung 5.16: **a)** Darstellung des Messsignals des beschädigten Lager 10, sowie des gefilterten Signals (orange) und dessen Hüllkurve (gelb). **b)** FFT des Hüllkurvensignals des unbeschädigten Lagers (orange, gleiche Prüfstandskonfiguration) und des beschädigten Lagers (blau) mit der BPFI und der Drehfrequenz f_n.

Im nächsten Schritt erfolgt die Anwendung der ML-Toolbox. Diese besitzt eine komplementäre Auswahl aus den Merkmalsextraktoren Adaptive Lineare Approximation (ALA), PCA, BFC, Beste Daubechies Wavelets (BDW) und Statistische Momente (SM) (Checkpunkt E13) sowie den Merkmalsselektoren Recursive Feature Elimination Support Vector Machine (RFESVM), ReliefF und der Pearson-Korrelation (Checkpunkt E14) (vgl. Abschnitt 3.4.5.2). Die Klassifikation erfolgt mittels Lineare Diskriminanzanalyse (LDA) und Mahalanobis-Distanz (Checkpunkt E15). Zur ergänzenden Anomalieerkennung wurde die in der ML-Toolbox implementierte Methode k-nächste Nachbarn (knn) [174] ausgewählt (Checkpunkt E16).

5.2.5.3 Modellbildung

Ein Vorteil der ML-Toolbox ist die automatisierte Auswertung der Performanz der jeweiligen Algorithmenkombination und die Auswahl der Kombination mit der höchsten Performanz (Checkpunkt E17 und E18). Tabelle 5.7 zeigt den resultierenden Validierungsfehler mit der LOGOCV. Die Algorithmenkombination BFC als Merkmalsextraktionsmethode und Pearson-Korrelation als Merkmalsselektionsmethode erreichte mit einem Validierungsfehler von 16 % die höchste Genauigkeit.

Tabelle 5.7: Ergebnisse der ML-Toolbox mit LOGOCV.

Merkmalsextraktor	**Merkmalselektor**	**Validierungsfehler** [%]
ALA	RFESVM	41 %
PCA		50 %
BFC		22 %
BDW		48 %
SM		23 %
ALA	RELIEFF	47 %
PCA		50 %
BFC		38 %
BDW		45 %
SM		25 %
ALA	**Pearson**	45 %
PCA		50 %
BFC		**16 %**
BDW		40 %
SM		26 %

Jedoch sind die Ergebnisse der ML-Toolbox ohne Betrachtung der Messunsicherheit und der Fortpflanzung durch die Algorithmen als unvollständig zu betrachten (vgl. Abschnitt 2.4.1). Daher wurde die Unsicherheit der ausgewählten Algorithmenkombination mit Hilfe der UA-ML-Toolbox bestimmt. Zur Reduzierung des Rechenaufwands wurde die Messunsicherheit für Fold 1 ermittelt. Weiterhin wurden die Messungen gemäß dem Nyquist-Shannon-Theorem [175] mittels Downsampling um Faktor 20 reduziert und so gekürzt, dass ein Messsignal 10 Umdrehungen umfasst. Abbildung 5.17 zeigt die resultierende Unsicherheitsbetrachtung mit der Zielgröße (grün gestrichelt), der Vorhersage des Algorithmus (blau) und der Vorhersage unter Berücksichtigung der Messunsicherheit (orange).

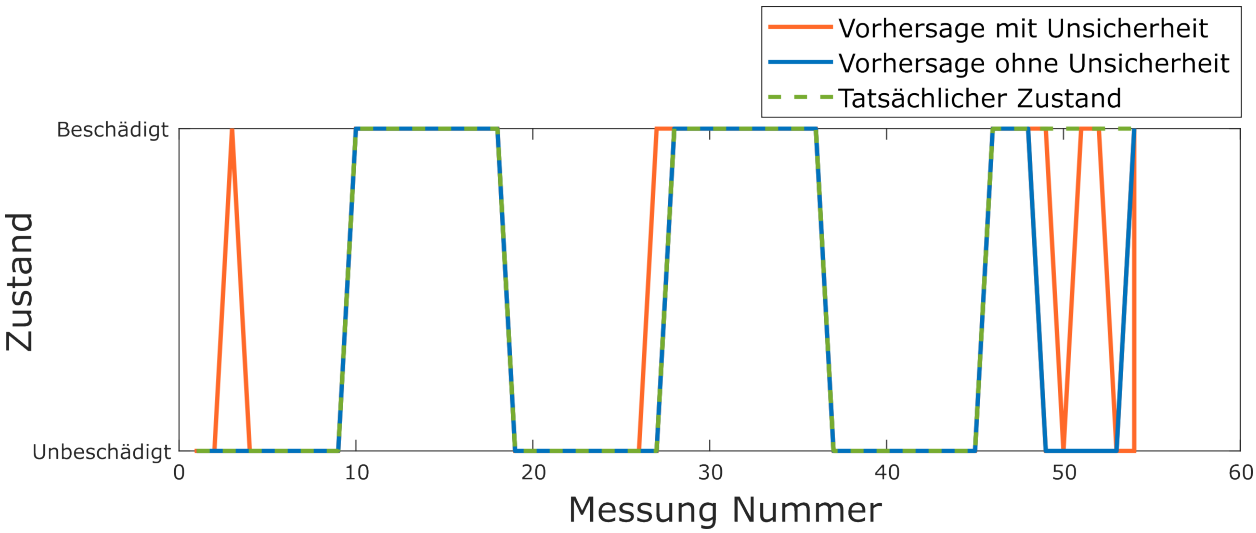


Abbildung 5.17: Beispielhafte Unsicherheitsbetrachtung der Vorhersage von Fold 1 der geresampleten Daten.

Während der Validierungsfehler ohne Betrachtung der Messunsicherheit bei 9,2 % liegt, kann dieser unter Berücksichtigung der Messunsicherheit auf bis zu 11,1 % ansteigen. Daraus ergibt sich eine Abweichung von 1,9 %. Dies deutet auf einen begrenzten Einfluss der Messunsicherheit auf die Vorhersagequalität des ML-Modells hin. Dennoch ist die Quantifizierung des Einflusses der Messunsicherheit essentiell, um die Vorhersagequalität fundiert beurteilen zu können.

Da die Messunsicherheit nicht den dominierenden Fehler darstellte, wurden anschließend die von der ML-Toolbox gewählten Algorithmen angepasst, um den Modellfehler zu reduzieren. In der Standardeinstellung extrahiert der BFC-Extraktor der ML-Toolbox 10 % der Frequenzen mit den höchsten Amplituden und der zugehörigen Phase. Für die Verbesserung wurde zunächst der gesamte Frequenzbereich betrachtet, da relevante Frequenzen nicht zwangsweise mit einer hohen Amplitude auftreten. Anschließend wurde im Merkmalsselektor die interne 10-fache Kreuzvalidierung durch die LOGOCV ausgetauscht. Dadurch ist das interne Szenario realistischer und berücksichtigt unbekannte Positionen. In der abschließenden Bewertung des Modells mit den Testdaten der Pos. D konnte mit dem verbesserten Modell (Training mit Pos. A bis Pos. C) der Validierungsfehler auf 4,3 % gesenkt werden.

Das Modell wurde anschließend mit allen vorhandenen Daten (Pos. A bis Pos. D) trainiert (Checkpunkt E20). Bei der nächsten Versuchsreihe muss es erneut auf seine Gültigkeit überprüft werden (Checkpunkt E21).

Um auch unbekannte Beschädigungen abzudecken und mittels ML zu detektieren, wurde zudem eine Anomalieerkennung implementiert. Hier wurde mit der knn-

Anomalieerkennung der ML-Toolbox unter einer 50 % Holdout-Validierung ein Modell auf Basis der Daten unbeschädigter Lager erstellt. Der Grenzwert, ab welcher eine Messung als Anomalie gekennzeichnet wird, wird automatisiert von der ML-Toolbox auf Basis einer hinterlegten Metrik ermittelt und ist über den Quellcode nachvollziehbar. Anschließend wurde das Modell mit den verbleibenden 50 % der Daten der unbeschädigten Lager sowie allen Daten der beschädigten Lager getestet. Abbildung 5.18 zeigt die Testergebnisse des Anomalieerkennung-Modells ohne (orange Punkte) und mit Beschädigung (blaue Punkte).

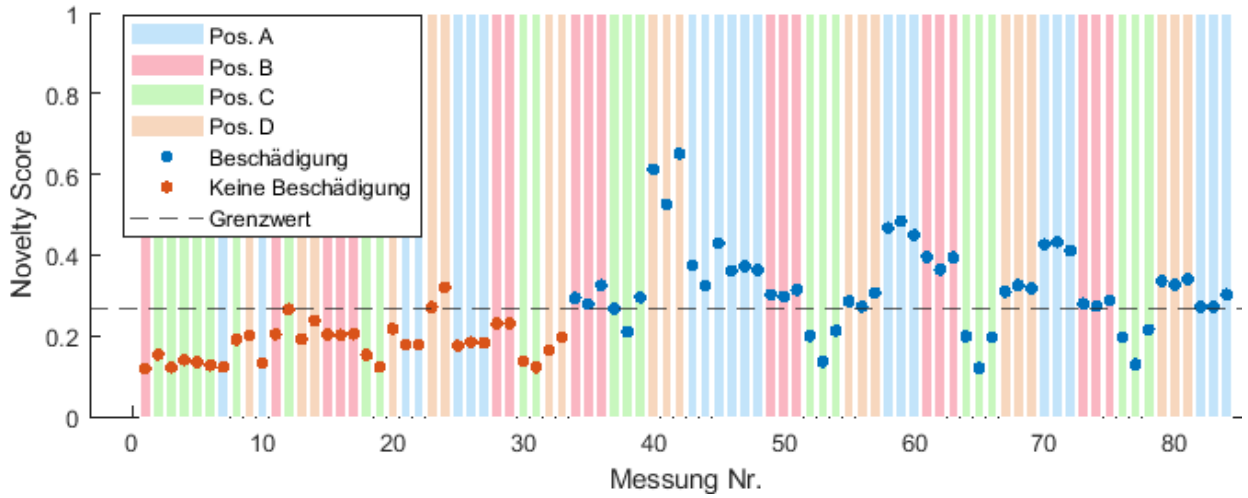


Abbildung 5.18: Scatterplot der Anomalieerkennung eingefärbt nach Positionen.

Insgesamt erzielte das Modell eine Genauigkeit von 84 %. Der Großteil der fälschlicherweise als unbeschädigt klassifizierten Messungen stammt von Pos. C (grüner Hintergrund), was darauf hindeutet, dass die Datengrundlage an dieser Position die Realität ggf. nicht hinreichend abbildet.

Das nachfolgende Checklisten-Kapitel Modellanwendung mit den Checkpunkten Regelmäßige Gültigkeitsprüfung des Modells (E22), Regelmäßiges Nachtrainieren des Modells (E23) und Dokumentation der Hard- und Softwareänderungen (E24) beziehen sich primär auf die dauerhafte Anwendung des Modells in der Industrie und konnten im vorliegenden Anwendungsfall nicht evaluiert werden.

5.2.6 Projektabschluss

Nach Abschluss der Modellbildung wurden im Projektabschluss zunächst die Projektziele mit den erreichten Ergebnissen verglichen (Checkpunkt F1):

- **AF1-Ziel 1**: Durch die Anwendung des UWD in Kombination mit der Visualisierungsmethode PCA konnten gängige Stör- und Einflussgrößen identifiziert werden.
- **AF1-Ziel 2**: Abgeleitet vom UWD konnte ein umfangreicher Datensatz aufgezeichnet werden, welcher auch weiterführende Analysen mit variierender Rotationsgeschwindigkeit ermöglicht.
- **AF1-Ziel 3**: Mit Hilfe der ML-Toolbox konnte ein ML-Modell erzeugt werden, welches einen Testfehler von 4,3 % an einer Position erreichte, welche nicht im Trainings- und Validierungsprozess enthalten war.

Somit konnten alle Ziele des ML-Projektes erreicht werden. Anschließend wurde die Einhaltung der im Lastenheft festgelegten Lasten überprüft:

- **AF1-LH 1**: Die Datenqualität des maßgeblichen Beschleunigungssensors ist gemäß den 15 IQ-Dimensionen und den Datenqualitätsmetriken nach Heinrich [66] als hoch zu bewerten. Die Daten des Kraftsensors und Motorcontrollers (Rotationsgeschwindigkeit) weisen hingegen durch die systematischen Einbrüche eine niedrigere Datenqualität auf. Entsprechend des FAIR Data Maturity Model [165] ist der Datensatz zudem als FAIR zu bewerten. Anzumerken ist jedoch eine potentielle Steigerung der FAIRness durch die Verbesserung von Indikatoren der Kategorie *Nützlich*.
- **AF1-LH 2**: Durch die Ansteuerung des Motorcontrollers konnte die Rotationsgeschwindigkeit automatisiert variiert werden.

Auch die Lasten konnten bei der Durchführung des ML-Projektes eingehalten werden. Trotz des Erreichens der Projektziele und des Einhaltens der Lasten konnten nachfolgende Lessons Learned Hypothesen (Checkpunkt F2) formuliert werden:

- **AF1-LL1**: Für eine fehlerfreie und konsistente Ausführung der Tätigkeiten erwies sich eine umfangreiche Schulung mit mehrfacher Wiederholung unter Aufsicht als notwendig. Eine einmalige Einweisung führte zu Unsicherheiten und Abweichungen im Ablauf.
- **AF1-LL2**: Die Überprüfung der Datenqualität muss anfänglich in einer höheren Frequenz erfolgen oder automatisiert werden.

AF1-LL1 bezieht sich hierbei auf die entstandenen Montagefehler der Hilfswissenschaftler und zielt auf deren zukünftige Vermeidung ab. Ziel hinter AF1-LL2 ist

die Vermeidung bzw. frühzeitige Erkennung von Fehlern in Daten, wie z.B. die systematischen Einbrüche der Rotationsgeschwindigkeit und Lastmessungen. Diese hätten durch das Integrieren von Grenzwerten leicht und schnell erkannt werden können. Die Lessons Learned Hypothesen wurden in der Wissensdatenbank von PIA hinterlegt (siehe Anhang A.2.1, Abbildung A.8).

Die Erstellung der Abschlusspräsentation (Checkpunkt F3) bzw. des Abschlussberichtes (Checkpunkt F4) erfolgt innerhalb der Erstellung dieser für das Projekt VProSaar. Abschließend wurden alle Unterlagen zentral und zugänglich hinterlegt (Checkpunkt F5).

5.2.7 Diskussion und Zwischenfazit

Nach Abschluss des Anwendungsszenarios Schadenserkennung an Zylinderrollenlagern kann ein erstes positives Fazit über die entwickelte Methodik der Checkliste und PIA gezogen werden. Trotz des primären Fokus für die Anwendung im Mittelstand eigneten sich sowohl die Checkliste als auch PIA für die Anwendung in einem wissenschaftlichen Kontext. Hinsichtlich der initial gestellten Forschungsfragen konnten durch die Integration der vorgestellten Forschungskonzepte in die Checkliste hochwertige Daten generiert (Forschungsfrage 1) und der Anwender befähigt werden, diese aufzuzeichnen (Forschungsfrage 2). Weiterhin wurde der Anwender durch die schrittweise Vorgehensweise der Checkliste und die Verwendung der ML-Toolbox dazu befähigt, eine erste Datenanalyse mittels ML erfolgreich durchzuführen (Forschungsfrage 3). Jedoch ist anzumerken, dass der Anwender bereits Erfahrung mit der Bearbeitung von ML-Projekten hatte, wodurch keine Unklarheiten bei der Bearbeitung der Checkliste auftraten. Weiterhin konnten bei der Bearbeitung des Anwendungsfalls folgende Verbesserungspotenziale identifiziert werden:

- **Muss-Checkpunkte**: Einige der Muss-Checkpunkte sind lediglich in der Industrie als Muss-Checkpunkte zu kategorisieren. Als Beispiel seien hier die Checkpunkte im Abschnitt 6.4 Modellanwendung angeführt. Während dieser Abschnitt essentiell für die sichere Anwendung von ML-Modellen bspw. in einer Produktion ist, stellen in der Forschung die aufgezeichneten Daten und die Realisierbarkeit einer grundlegenden Schadenserkennung bereits die primären Ergebnisse des ML-Projektes dar und es erfolgt keine weitergehende Anwendung des ML-Modells.
- **Nummerierung der Checkpunkte**: Die Nummerierung der Checkpunkte bietet dem Anwender eine Orientierung innerhalb der Checkliste, suggeriert aber

gleichzeitig eine lineare Reihenfolge der Checkpunkte. Je nach Gegebenheiten kann eine abweichende Reihenfolge der Checkpunkte jedoch sinnvoll sein. Als Gegenmaßnahme wurden im Ablaufplan bereits Checkpunkte in Gruppen zusammengefasst und auf iterative Prozesse hingewiesen.

- **Fehlendes Backend**: Aufgrund des fehlenden Backends von PIA existierte keine direkte Schnittstelle zu einer Datenablage sodass Fortschritte nicht gespeichert werden konnten.

Die Integration des Anwendungsfalles in den Softwaredemonstrator PIA zeigte dennoch ein großes Potenzial des entwickelten Konzeptes. Hervorzuheben ist hier z.B. die Verfügbarkeit der Daten und Metadaten sowie die einfache Handhabung der Checkliste mit Kommentar- und Anhangfunktion.

5.3 Anwendungsfall 2: Wandelbares Montagesystem

Nachdem die Methodik der vorgestellten Checkliste und PIA erfolgreich in Anwendungsfall 1 in einem internen Datenanalyseprojekt durch den Autor dieser Dissertation angewendet wurde, wird im nächsten Schritt die Methodik in einem forschungsgruppenübergreifenden, industrienahen Datenanalyseprojekt evaluiert. Hierzu wird eine Machbarkeitsstudie zur Schadenserkennung von Schrauberbits an der Schraubstation eines wandelbaren Montagesystems (WaMo) durchgeführt.

Die Gliederung und der Umfang dieses Kapitels weichen zugunsten einer erhöhten Lesbarkeit und der Reduzierung redundanter Inhalte von Anwendungsfall 1 ab.

Die Ergebnisse der Machbarkeitsstudie zur Schadenserkennung von Schrauberbits wurden in [176] veröffentlicht.

5.3.1 Motivation und Problemstellung

Bestandsanlagen im Brownfield stellen einen Großteil der Anlagen in der deutschen Produktionslandschaft dar. Insbesondere an älteren Anlagen ist die Datenaufnahme häufig, z.B. aufgrund der geringen Konnektivität, mit erhöhtem Aufwand und Kosten verbunden [1].

Das in Anwendungsfall 2 betrachtete WaMo repräsentiert bis zu einem gewissen Grad eine solche Bestandsanlage im Brownfield eines Unternehmens. Somit kann die Methodik von PIA und der Checkliste industrienah und ressourcenschonend (u.a. kein Stillstand von Produktionsanlagen) am WaMo getestet werden. Im Rahmen des Projektes VProSaar soll an dieser Anlage überprüft werden, ob Beschädigungen

eines Schrauberbits unter Verwendung eines zusätzlichen Beschleunigungssensors frühzeitig erkannt werden können. Solche Beschädigungen bzw. Verschleiß an Schrauberbits können zu ungeplanten Stillständen führen und somit die Effizienz der Prozesse reduzieren. Um die Machbarkeit dieses Vorhabens zu bewerten, wird zunächst eine Machbarkeitsstudie in Form eines ML-Projektes durchgeführt. Aufgrund der proprietären Software des in der Schraubstation verwendeten Schraubers konnte nicht auf die dort anfallenden Daten zugegriffen werden. Ein zusätzlicher Beschleunigungssensor in Kombination mit dem Messkoffer bietet das Potenzial, eine Schadenserkennung des Schrauberbits als Retrofit nachzurüsten, ohne Änderungen an der bestehenden Anlage durchzuführen.

5.3.2 Beschreibung der Anlage

Das WaMo wurde vom Lehrstuhl für Montagesysteme der Universität des Saarlandes entwickelt und ist eine modulare Montageline, in der die einzelnen Module flexibel ausgetauscht und neu angeordnet werden können [177]. Abbildung 5.19 a) zeigt die Konfiguration des WaMo zum Zeitpunkt des ML-Projektes mit zwei Stationen.

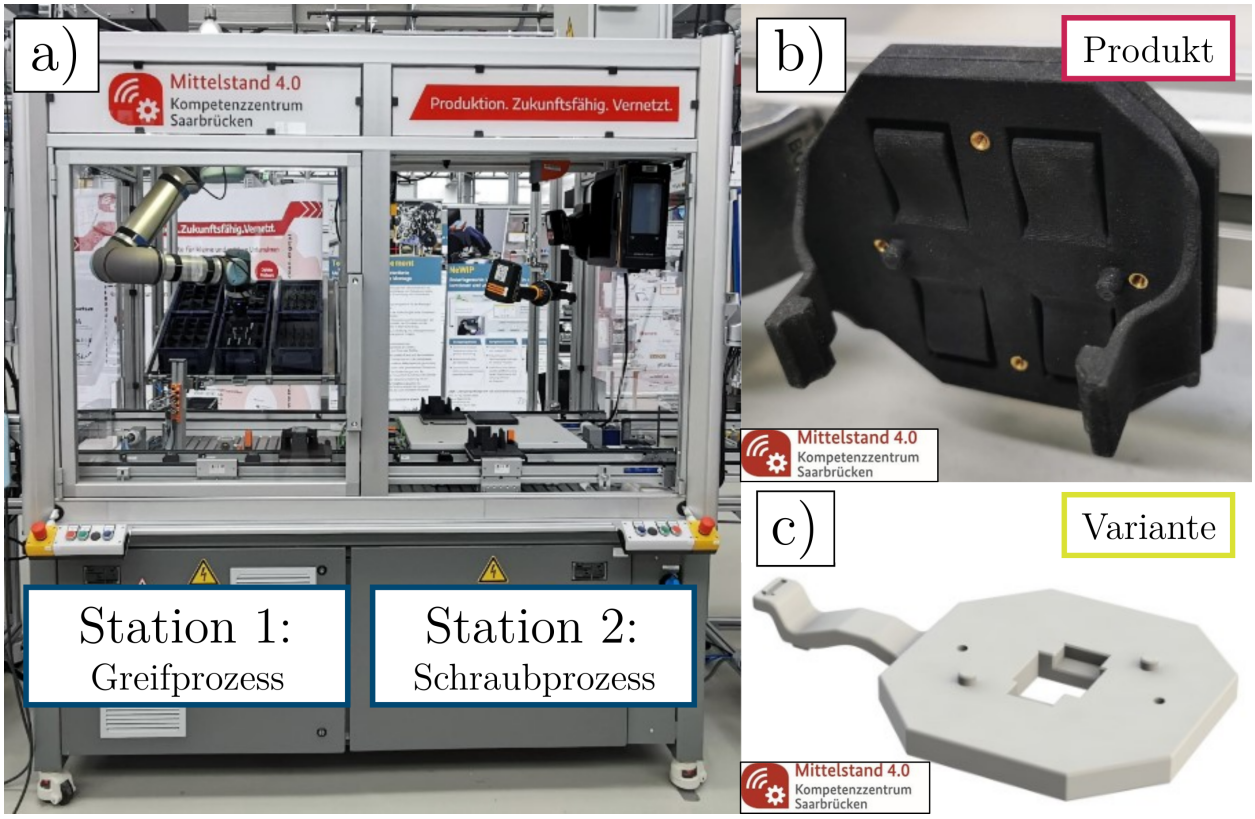


Abbildung 5.19: Darstellung des Wandelbaren Montagesystems mit **a)** den jeweiligen Stationen, **b)** dem gefertigten Produkt und **c)** einer Produktvariante, adaptiert von [122].

Die erste Station besteht aus einem Greifprozess, in welchem ein Roboter automatisiert zwei Komponenten des zu fertigenden Halters (Abbildung 5.19 b) aus einem Magazin entnimmt und auf einem Werkstückträger platziert. Neben der in Abbildung 5.19 b) gezeigten Variante, existiert eine weitere mögliche Variante des Halters, dargestellt in Abbildung 5.19 c). Nachdem das zusammengesetzte Produkt durch den Werkstückträger zur Schraubstation (Station 2) transportiert wurde, werden die Einzelteile durch einen Projektmitarbeiter händisch mittels Elektroschrauber (Atlas Tensor STB Modell ETVSTB34-30-10W) verschraubt. Gängige Praxis bei der Verschraubung von Bauteilen in der Produktion ist die Verwendung von drehwinkelgesteuerten bzw. drehmomentgesteuerten elektrischen Schraubern, wie dem oben genannten Atlas Tensor STB [178]. Dieser ist jedoch mit proprietärer Software ausgestattet und erlaubt dem Anwender keinen Zugriff auf die anfallenden Daten (Herausforderung S2 nach Wilhelm in Tabelle 2.1).

Um keine Produkte während der Machbarkeitsstudie zu beschädigen, erfolgten die Verschraubungen in einer speziell für die Messungen angefertigten Gewindeplatte (Abbildung 5.20), welche in die Schraubstation montiert werden kann.

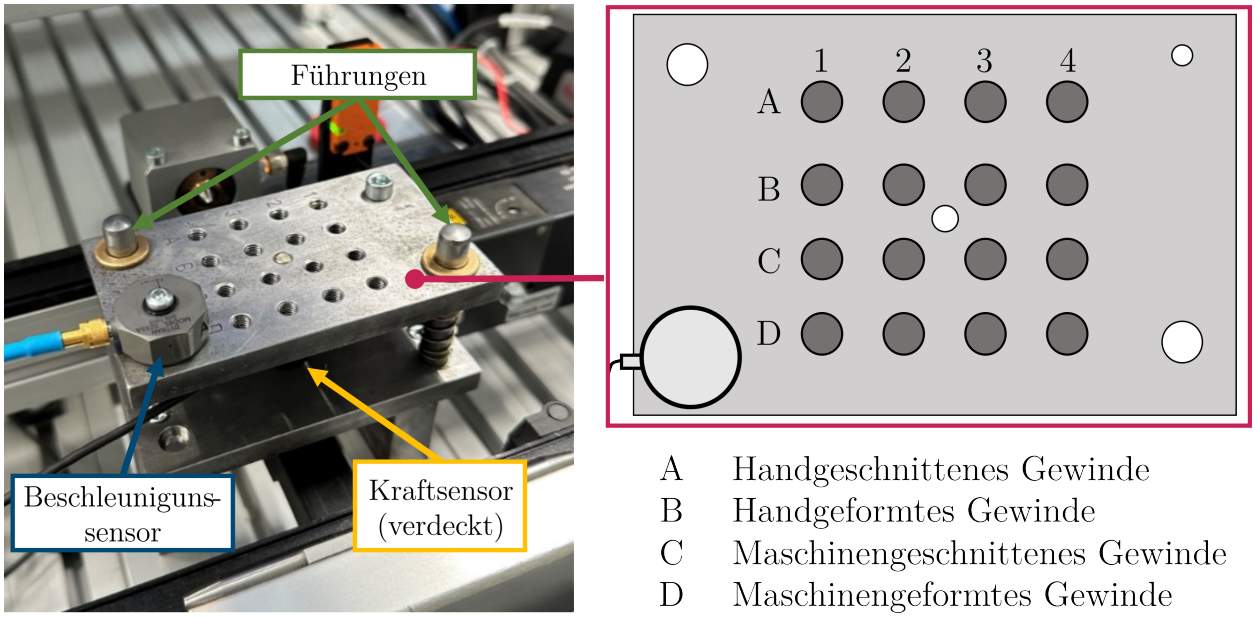


Abbildung 5.20: Darstellung der Gewindeplatte mit dem zusätzlichen Beschleunigungssensor und der schematischen Darstellung der einzelnen Gewinde.

Die Platte besitzt 16 Gewinde, welche durch die vier unterschiedlichen Varianten handgeschnitten (A), handgeformt (B), maschinengeschnitten (C) und maschinengeformt (D) hergestellt wurden. Jede Variante wurde zudem vier Mal (1-4) hergestellt. So kann das spätere ML-Modell auf die Robustheit gegenüber der Herstellungsart (A-D) inklusive Fertigungstoleranzen (Wiederholung 1-4) überprüft werden. Um die

vom Projektmitarbeiter verwendete Anpresskraft zu erfassen, wurde ein zusätzlicher Kraftsensor (gelb) zwischen Gewindeplatte und Halterung montiert, wobei zwei Führungen (grün) ein Kippen der Platte verhindern. Der Beschleunigungssensor (blau) wurde fest mit der Platte verschraubt. Sowohl Kraftsensor, als auch Beschleunigungssensor wurden an den Messkoffer angeschlossen und mit 20 kHz abgetastet.

5.3.3 Implementierung in PIA

Im ersten Schritt der Implementierung des WaMo in PIA erfolgte die Erstellung eines Use-Cases (siehe Abbildung A.13) mit den beiden Stationen Greifstation und Schraubstation (siehe Abbildung A.14). Damit einhergehend wurden angelegt:

- **Produkt**: Das gefertigte Produkt wurde mit seinen Varianten A und B in PIA implementiert (siehe Abbildung A.15).
- **Betriebsmittel**: Die Betriebsmittel der beiden Stationen wurden mit ihren zugehörigen Dokumenten in PIA hinterlegt (siehe Abbildung A.16).
- **Maschinenlesbare Metadaten**: Die vorhandenen Metadaten der Schraubstation wurden exemplarisch gemäß der m4i Ontologie in PIA hinterlegt (siehe Abbildung A.17).
- **Video**: Ein Video, welches den Ablauf des Greif- und Schraubprozesses veranschaulicht (siehe Screenshot in Anhang, Abbildung A.18), wurde in PIA hinterlegt.
- **Sensoren**: Die verwendeten Sensoren des Messkoffers wurden mit ihren Metadaten in PIA hinterlegt (siehe Abbildung A.19).

Weiterhin wurde im Menüpunkt Checkliste ein neues ML-Projekt angelegt (siehe Abbildung A.20), welches im nachfolgenden Abschnitt bearbeitet wird.

Analog zu Anwendungsfall 1 sind die Abbildungen des zweiten Anwendungsfalls in PIA zur besseren Lesbarkeit in Anhang A.3.1 nachgelagert.

5.3.4 Anwendung der Checkliste

Die Anwendung der Checkliste wurde in Anwendungsfall 1 ausführlich beschrieben. Da der Autor dieser Dissertation in Anwendungsfall 2 lediglich in beratender und unterstützender Funktion tätig war, werden im nachfolgenden Abschnitt nur die

zur Evaluierung der Methodik relevanten Checkpunkte behandelt. So werden z.B. Checkpunkte zur Fehlervermeidung, wie *Checkpunkt A1: Checkliste gelesen und verstanden*, nicht aufgeführt.

5.3.4.1 Vorbereitung und Projektplanung

Im ersten Schritt der Vorbereitung und Projektplanung wurde gemäß Checkpunkt A2 die Zielsetzung des ML-Projektes wie folgt festgelegt:

- **AF2-Ziel 1**: Identifikation von Stör- und Einflussgrößen am Schraubprozess des WaMo.
- **AF2-Ziel 2**: Überprüfung der Machbarkeit einer Schadenserkennung eines künstlich beschädigten Schrauberbits basierend auf Messungen eines Beschleunigungssensors.
- **AF2-Ziel 3**: Erzeugung eines ersten robusten ML-Modells, welches eine Beschädigung am Schrauberbit unter Verwendung eines Beschleunigungssensors erkennen kann.

Sowohl die Auswahl der Schraubstation der WaMo als auch die weiteren Randbedingungen (Checkpunkte A3-A14), wie z.B. der Zeitplan (Checkpunkt A10) und Verantwortlichkeiten innerhalb des ML-Projektes (Checkpunkt A12) wurden durch das übergeordnete Projekt VProSaar vorgegeben.

Erwähnenswert ist die gruppenübergreifende Zusammenarbeit des Lehrstuhls für Montagesysteme als Fachexperten für die Montageanlage und deren Prozesse und des Lehrstuhls für Messtechnik als Fachexperten für Datenerfassung und deren Auswertung mittels ML.

5.3.4.2 Mess- und Datenplanung

In der Mess- und Datenplanung (MuD) wurde zunächst Prozesswissen aufgebaut (Checkpunkt B1-B7). Kernelement ist die Erstellung eines Ursache-Wirkungs-Diagramms (UWD) in Checkpunkt B3 und die anschließende Identifizierung von Störgrößen in Checkpunkt B5. Abbildung 5.21 zeigt das resultierende UWD.

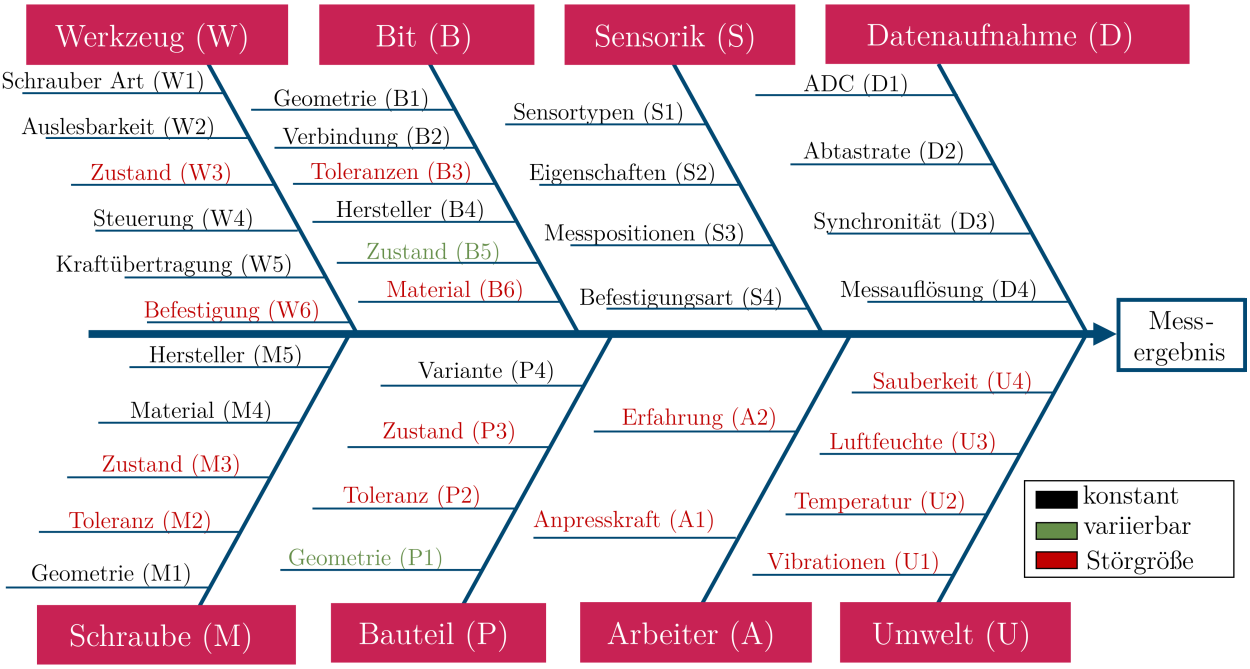


Abbildung 5.21: Ursache-Wirkungs-Diagramm für die Einflussgrößen des Zylinderrollen-Prüfstandes auf das Messergebnis.

Analog zu dem erstellten UWD in Anwendungsfall 1 wurde die Eingliederung von den 5M zu den Gruppen Werkzeug (W), (Schrauber-)Bit (B), Sensorik (S), Datenaufnahme (D), Schraube (M), Bauteil (P), (Projekt-)Arbeiter (A) und Umwelt (U) abgeändert und die Einflussfaktoren in die drei Kategorien konstant, variierbar und Störgröße unterteilt:

- **Konstant**: Da die Messungen über einen kurzen Zeitraum erfolgen, bleibt der Zustand des Werkzeuges (W3) nahezu unverändert und wird als konstant angenommen. Weiterhin erfolgten die Messungen durch einen einzelnen Projektmitarbeiter unter Laborbedingungen, wodurch auch die Erfahrung des Projektmitarbeiter (A2) und die Umwelteinflüsse U1-U4 als konstant betrachtet werden können. Jede Messung wurde mit einer neuen Schraube durchgeführt, wodurch deren Zustand als durchgehend neuwertig angesehen werden kann (M3). Zudem werden durch die verschiedenen Schrauben geometrische Toleranzen (M2) in den Daten abgedeckt. Hinsichtlich des Schrauberbits wird im Rahmen der ersten Messkampagne lediglich ein Schrauberbit betrachtet, wodurch die Störgrößen Toleranz (B3) und Material (B6) in der Machbarkeitsstudie vernachlässigt werden können.

- **Variierbar**: Als variierbare Einflussgrößen wurden die Bauteilgeometrie (P1) und der Zustand des Schrauberbit (B5) eingruppiert. Die Bauteilgeometrie wird

aufgrund der vier Herstellungsarten (A-D) an vier Positionen (1-4) und der Zustand des Schrauberbits aufgrund der künstlichen Beschädigung als variierbar angenommen.

- **Störgröße**: Als relevante Störgrößen wurden der Werkzeugzustand (W3) und dessen Befestigung (W6), die Toleranzen (B3) und das Material (B6) des Schrauberbits, der Zustand (M3) und die Toleranzen (M2) der Schraube und des Bauteils (P2, P3), die eingetragene Anpresskraft (A1) durch den Projektmitarbeiter und dessen Erfahrung (A2), sowie die Umwelteinflüsse bzw. Umgebungseinflüsse wie Vibrationen (U1), die Temperatur (U2), Luftfeuchte (U3) und die Sauberkeit des Arbeitsplatzes (U4) identifiziert. Aufgrund der kurzen Dauer der Messkampagne sind W3, W6, P2, P3 und A2 nahezu konstant. Die Umgebungseinflüsse U2 bis U4 sind zu vernachlässigen, da die Messkampagne unter Laborbedigungen in einem klimatisierten Raum durchgeführt wurden.

Neben einem Beschleunigungssensor (Dytran 3233A [179]) wurde zur Erfassung der Störgröße Anpresskraft (A1) ein zusätzlicher Kraftsensor (EMSYST EMS 20 [180]) als zusätzliche Sensorik in den Messaufbau integriert (Checkpunkt B6). Die Messungen wurden mit dem in Abschnitt 5.1 beschriebenen Messkoffer durchgeführt. Dies ermöglichte die Überprüfung der Machbarkeit, ohne eine Veränderung der existierenden Infrastruktur bzw. der aktuellen Konfiguration des WaMo.

Weiterhin wurden die nachfolgenden Schritte der MuD durchgeführt:

- **Normen und Standards** (Checkpunkte B8-B9): Für die Verwendung von Beschleunigungssensoren zur Schadenserkennung von Schrauberbits existieren nach Wissen des Autors zum Zeitpunkt dieser Dissertation keine dezidierten Normen oder Standards. In der Literatur werden bereits Beschleunigungssensoren und Clustering-Methoden genutzt, um die Klemmkraft von Schraubverbindungen abzuschätzen [181]. In einer weiteren Studie [182] werden die durch Reibung bzw. den Stick-Slip-Effekt entstehenden Vibrationen untersucht, um die erforderliche Klemmkraft präziser zu erreichen. Dies zeigt, dass bereits erste Studien im Kontext von Schraubprozessen erfolgreich durchgeführt werden konnten.

- **Messunsicherheiten** (Checkpunkte B10-B11): Der verwendete Beschleunigungssensor wurde bereits in Anwendungsfall 1 eingesetzt. Die entsprechenden Messunsicherheiten finden sich in Tabelle 5.2.

- **Datenaufbau** (Checkpunkte B12-B25): Der Aufbau und die Struktur der Daten wurde analog zu Anwendungsfall 1 gestaltet. Die aufgezeichneten Metadaten (Checkpunkt B17) sind in Tabelle 5.8 aufgelistet.

Tabelle 5.8: Aufgezeichnete Metadaten in Anwendungsfall 2.

Name	**Wert**	**Beschreibung**
Lauf	1-5	Anzahl der Wiederholungen einer Verschraubung.
Zustand	0, 1	Zustand des Schrauberbits, wobei 0 einen unbeschädigten und 1 einen beschädigten Schrauberbit darstellt.
Erfolg	0, 1	Erfolg einer Verschraubung, wobei 0 eine erfolgreiche und 1 eine fehlgeschlagene Verschraubung darstellt.
Messdauer	0-11 s	Dauer der Messung in Sekunden.
Zeitstempel	DD-MM-YYYY HH:MM:SS	Zeitstempel des Messkoffers bei Start der Messung.
Gewinde-Nr.	1-4	Nummer des Gewindes.
Herstellungs-varianten	1-4	Herstellungsvarianten, analog zu der Bezeichnung A-D (s.o.).

- **Datenablage** (Checkpunkte B26-B30): Die Ablage der Daten erfolgte auf dem Redundant Array of Independent Disks (RAID)-gesicherten Server des Lehrstuhls für Messtechnik mit Zugriffsrechten für die entsprechenden Projektmitglieder.
- **Manuelle Datenquellen** (Checkpunkte B31-B36): Als manuelle Datenquellen lagen die Metadaten Herstellungsart des Gewindes, Lauf, Zustand des Schrauberbits und Erfolg einer Verschraubung vor. Diese wurden aufgrund der geringen Anzahl durchgeführter Messungen händisch in den Datensatz eingetragen.

5.3.4.3 Datenaufnahme

Vor Beginn der Messkampagne wurde eine erste Testmessung mit einem unbeschädigten und einem beschädigten Schrauberbit durchgeführt (Checkpunkt C2). Zudem wurde in dieser Testmessung untersucht, in welchem Ausmaß die künstliche Beschädigung vorliegen muss, um ein Überdrehen des Schrauberbits hervorzurufen. Abbildung 5.22 zeigt den unbeschädigten (blau) und beschädigten Schrauberbit (rot), wie sie in der Messkampagne verwendet wurden.

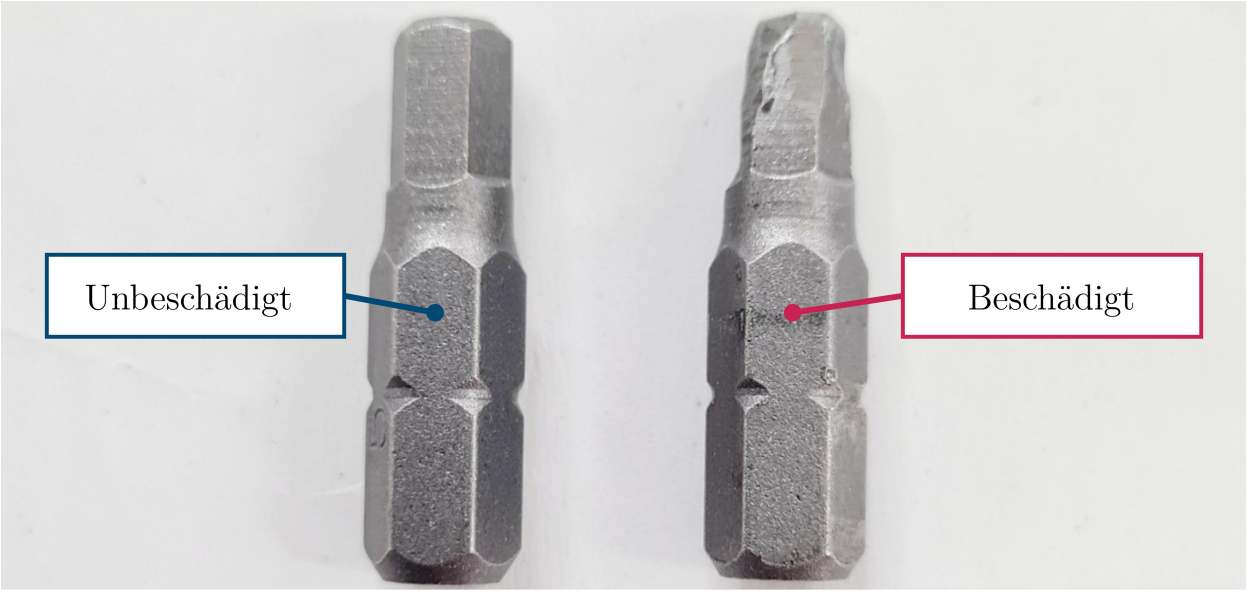


Abbildung 5.22: Darstellung eines unbeschädigten (blau) und eines beschädigten Schrauberbits (rot).

Anschließend wurde die ordnungsgemäße Funktion der Sensoren sowie die Struktur der erfassten Daten erfolgreich überprüft (Checkpunkte C3-C8). Danach wurde die Messkampagne von einem Projektmitarbeiter gestartet (Checkpunkt C10) und durchgeführt.

5.3.4.4 Datenprüfung und Datenbereinigung

Nach Abschluss der Datenaufnahme wurde im ersten Schritt die Qualität der Daten gemäß den 15 IQ-Dimensionen geprüft. Auffällig war die negativ bewertete Dimension IQ-8, dargestellt in Tabelle 5.9.

Tabelle 5.9: Auszug der Bewertung der Datenqualität mittels der 15 IQ-Dimensionen in positiv und negativ (nach Hildebrand [67]) aus Anhang, Tabelle A.9.

Nr.	**IQ-Dimension**	**Bewertung**	**Anmerkung**
IQ-8	Fehlerfreiheit	negativ	Fehlerhafte Messungen des Kraftsensors.

Diese bezieht sich primär auf die Daten des Kraftsensors. Hier bemerkte der Projektmitarbeiter während der Messungen, dass die Schrauben an den Positionen B2, B3, C2 und C3 beim Einschrauben den Kraftsensor berühren und somit die Messung verfälschen. Dies hat zur Folge, dass alle nachfolgenden Messungen am Kraftsensor für die Datenanalyse nicht herangezogen werden können. Für nachfolgende

Messkampagnen wurden kürzere Schrauben verwendet. Eine vollständige Bewertung der IQ-Dimensionen findet sich im Anhang, Tabelle A.9.

Im nächsten Schritt erfolgte die Bewertung der Datenqualität nach Heinrich [66], siehe Tabelle 5.10.

Tabelle 5.10: Bewertung der Datenqualität nach Heinrich et al. [66].

Dimension	**%**	**Anmerkung**
Vollständigkeit	99,37 %	Von geplanten 160 Messungen wurden 159 durchgeführt.
Fehlerfreiheit	/	Keine vorgegebenen Größen.
Konsistenz	62 %	Gemäß der definierten Regel: Messwerte des Beschleunigungssensors sind innerhalb des gültigen Messbereiches.
Aktualität	100 %	Primär relevant für Daueranwendung des Modells bzw. Anlagen mit voranschreitendem Verschleiß.

Auffällig sind hier die Dimensionen *Fehlerfreiheit* und *Konsistenz*. Die *Fehlerfreiheit* konnte nicht bewertet werden, da keine Referenzwerte erfasst wurden, wie bspw. bei der Regelung der Rotationsgeschwindigkeit in Anwendungsfall 1. Weiterhin deutet die Dimension der *Konsistenz* auf eine hohe Inkonsistenz der Daten des Beschleunigungssensors hin. Eine Messung wurde als inkonsistent bewertet, wenn ein Messwert des Beschleunigungssensors den gültigen Messbereich von $\pm 5\,\mathrm{g}$ über- bzw. unterschreitet. Ursache hierfür waren einerseits entstehende Impulse beim Überdrehen des Schrauberbits und andererseits ein hartes Aufsetzen des Schraubers auf die Schraube durch den Projektmitarbeiter. Diese können bei Bedarf durch Zuschneiden des Messsignals entfernt werden.

Die Beurteilung der FAIRness der Daten (Abbildung 5.23) zeigt ebenfalls einen Optimierungsbedarf, insbesondere in der Dimension der Auffindbarkeit.

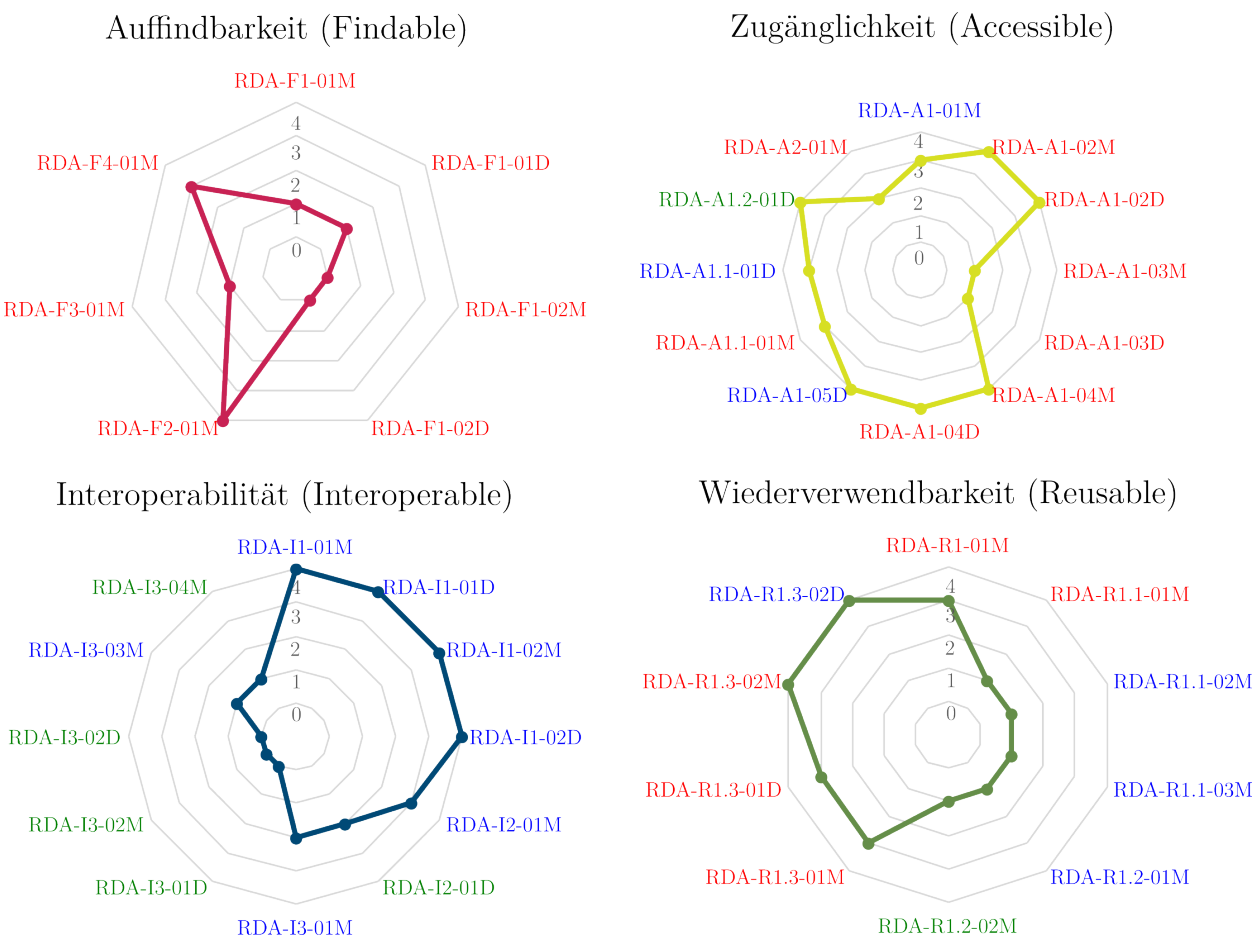


Abbildung 5.23: FAIRness der Daten nach [165] mit den jeweiligen FAIR-Indikatoren, dargestellt als Spiderplots.

Da es sich bei der Messkampagne in erster Linie um eine Machbarkeitsstudie handelt, wurde lediglich auf eine interne Auffindbarkeit geachtet. Dies stellt die Ursache für die geringe Bewertung der Auffindbarkeit dar, da mehrere der Indikatoren eine global eindeutige Zuordnung der Daten und Metadaten fordern, wie z.B. RDA-F1-02M und RDA-F1-02D (siehe Tabelle A.5).

Die Prüfung und Bereinigung der Daten (D1-D11) fokussierte sich auf die Bereinigung fehlerhafter Messungen und die Korrektur von Metadaten (Checkpunkt D5). So wurde z.B. die Reihenfolge bei zwei Messungen vertauscht.

5.3.4.5 Datenauswertung und Modellbildung

Im ersten Schritt der Datenauswertung und Modellbildung erfolgten die Datenvisualisierung und der weitere Aufbau von Datenverständnis (Checkpunkte E1-E7). Abbildung 5.24 zeigt beispielhaft eine Messung des Beschleunigungssensors (Z-Achse) einer Verschraubung mit unbeschädigtem (blau) und beschädigtem (orange) Schrauberbit, sowie den gültigen Messbereich des Beschleunigungssensors (±5 g, grau).

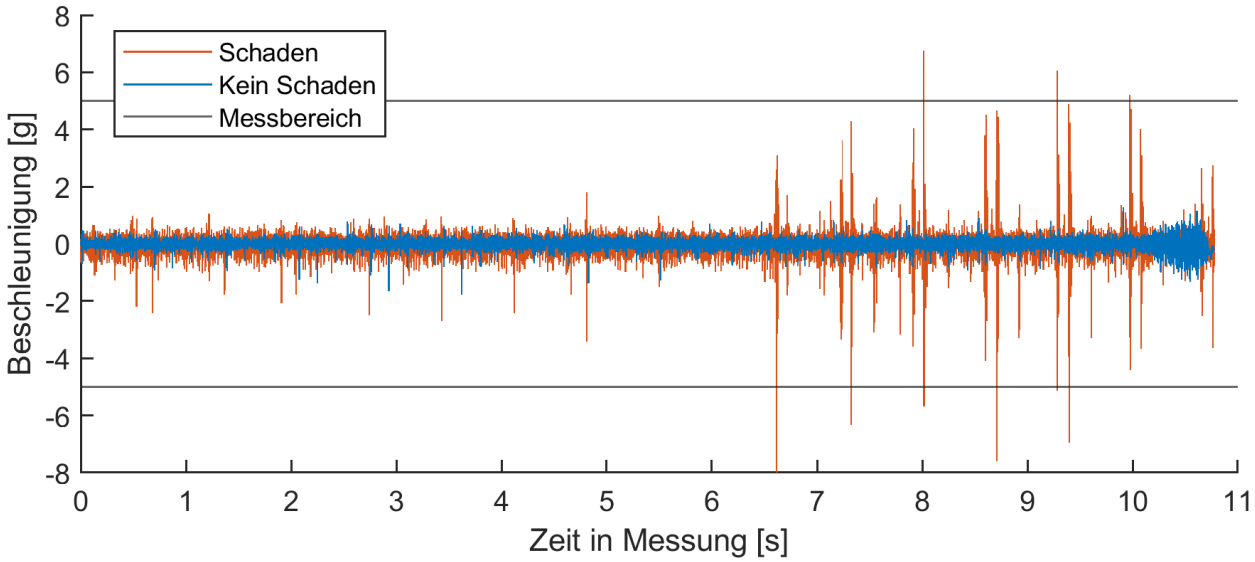


Abbildung 5.24: Signal des Beschleunigungssensors bei einem Schraubvorgang mit unbeschädigtem (blau) und beschädigtem (orange) Schrauberbit.

Im Messsignal des beschädigten Schrauberbits sind deutliche Impulse zu erkennen, welche durch das Überdrehen des Schrauberbits entstehen. Wie in der Dimension Konsistenz nach Heinrich angedeutet, überschreitet das Messsignal bei Verwendung eines beschädigten Schrauberbits den Messbereich. Daher muss in nachfolgenden Messkampagnen ein Beschleunigungssensor mit größerem Messbereich verwendet werden. Im Rahmen der Machbarkeitsstudie können die Messungen dennoch verwendet werden, da sie die Tendenzen der entstehenden Schwingungen korrekt wiedergeben und lediglich die gemessenen Absolutwerte bei Überschreitung des Messbereichs ungültig sind.

Anschließend wurden die Histogramme der Messungen aus Abbildung 5.24 erstellt und in Abbildung 5.25 visualisiert (Checkpunkt E4).

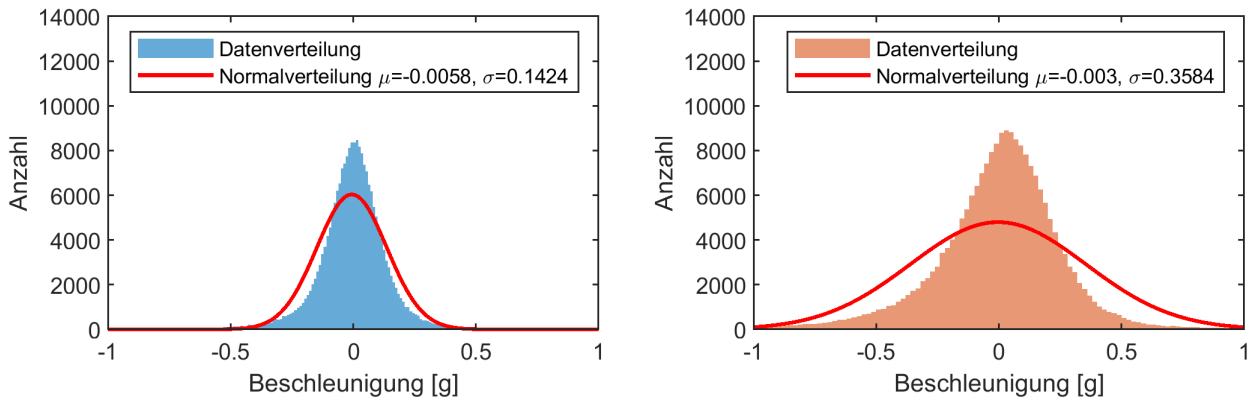


Abbildung 5.25: Histogramm der Messungen aus Abbildung 5.24 und der zugehörigen angepassten Normalverteilung.

Zum Vergleich wurde außerdem eine angepasste Normalverteilung (rot) dargestellt. Sowohl die Messung mit unbeschädigtem Schrauberbit als auch mit beschädigtem Schrauberbit weisen eine leichte Linksschiefe mit positivem Kurtosiswert auf, wobei diese bei einer Beschädigung stärker ausgeprägt ist.

Weiterhin wurde eine Hauptkomponentenanalyse (PCA) der unbeschädigten Daten durchgeführt (Checkpunkt E6). Da diese keine Clusterbildungen aufzeigte, wurde eine weitere PCA mit den extrahierten Merkmalen des BFC-Extraktors der ML-Toolbox durchgeführt. Abbildung 5.26 zeigt die Ergebnisse der PCA, eingefärbt nach a) Gewindenummer, b) Herstellungsvariante und c) Lauf.

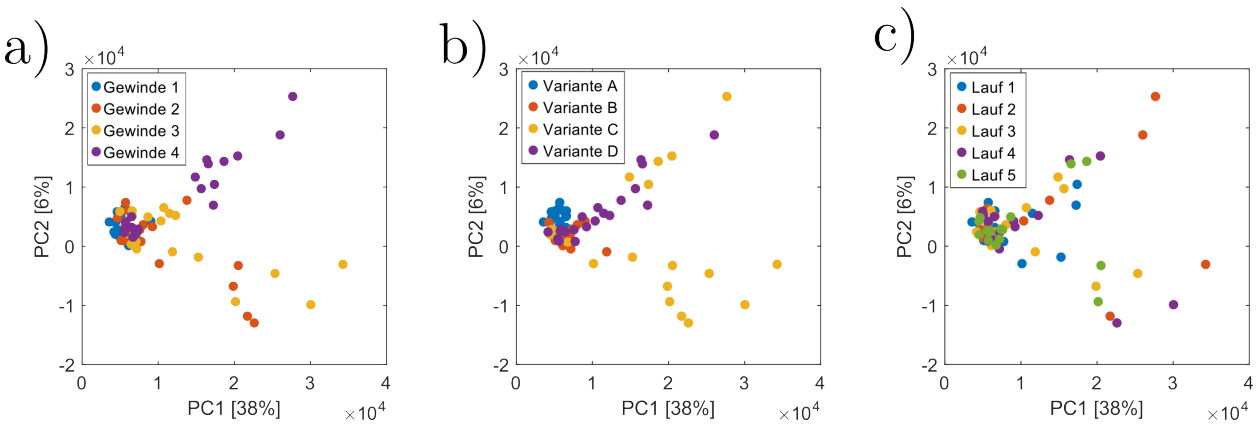


Abbildung 5.26: PCA der Merkmale des BFC-Extraktors, eingefärbt nach **a)** Gewindenummer, **b)** Herstellungsvariante und **c)** Lauf.

Während sich keine klaren Cluster bei der Einfärbung nach Lauf (c) zeigen, lassen sich bei der Gewindenummer und der Herstellungsvariante Cluster erkennen. Diese Cluster sind in den dargestellten Dimensionen nicht trennscharf, aber verdeutlichen dennoch einen Einfluss der Fertigungstoleranz bei der mehrfachen Herstellung eines Gewindes und deren Herstellungsart auf die Daten. Jedoch ist zu beachten, dass die Gewindenummer und die Herstellungsvariante in einer systematischen Reihenfolge 1-4 bzw. A-D angeordnet wurden und nicht randomisiert sind. Dadurch kann die Position der Gewinde ebenfalls eine Störgröße darstellen, welche nicht im UWD berücksichtigt wurde.

Die Betrachtung des quasi-statischen Signals (Checkpunkt E3) und der Boxplot-Diagramme (Checkpunkt E5) zeigte keine Auffälligkeiten und findet sich nachgelagert im Anhang in Abbildung A.21 und Abbildung A.22.

Im nächsten Schritt erfolgte die Auswahl von ML-Algorithmen (Checkpunkte E8-E11). Analog zu Anwendungsfall 1 wurde die Modellbildung unter Verwendung eines Notebooks mit einem Intel Core i7-10610U CPU mit 1.8 GHz und 4 Kernen durchgeführt. Das Lernproblem wurde, gemäß AF2-Ziel 2, auf das

Klassifizierungsproblem Schrauberzustand unbeschädigt/beschädigt festgelegt. Die Validierung des ML-Modells erfolgte mit einer Leave-One-Group-Out-Cross-Validation (LOGOCV). Hierzu wurden die Daten zunächst nach Herstellungsvariante unterteilt und die maschinengeformten Daten als Testdaten ausgewählt. In zukünftigen Auswertungen sollten auch die anderen Herstellungsvarianten als Testdaten verwendet werden, um zusätzliche Erkenntnisse über deren Einfluss auf das Modell zu gewinnen, z.B. in einer verschachtelten LOGOCV.

Die Modellbildung erfolgte anhand einer Kreuzvalidierung mit den drei Varianten handgeschnitten, handgeformt und maschinengeschnitten. Abbildung 5.27 zeigt die Vorgehensweise der LOGOCV.

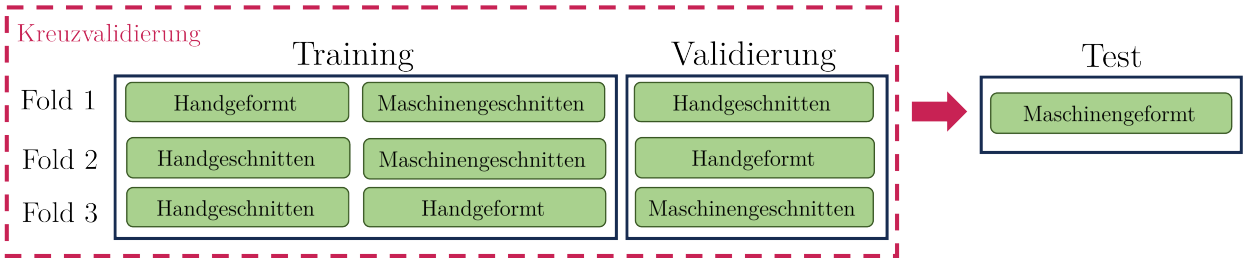


Abbildung 5.27: Darstellung des LOGOCV mit dem systematischen auslassen einer Herstellungsvariante (A-C) in einer Kreuzvalidierung und dem anschließenden Test von Variante D.

In Anwendungsfall 2 wurde ebenfalls die ML-Toolbox bzw. UA-ML-Toolbox verwendet, um eine geeignete Algorithmenkombination aus Merkmalsextraktor und Merkmalsselektor für einen LDA-Klassifikator zu finden (Checkpunkt E12-E15). Tabelle 5.11 zeigt die resultierenden Validierungsfehler der Algorithmenkombinationen der ML-Toolbox (Checkpunkt E17).

Tabelle 5.11: Ergebnisse der ML-Toolbox mit LOGOCV.

Merkmalsextraktor	**Merkmalselektor**	**Validierungsfehler** [%]
ALA		44 %
PCA		50 %
BFC	RFESVM	20 %
BDW		42 %
SM		11 %
ALA		43 %
PCA		50 %
BFC	RELIEFF	14 %
BDW		49 %
SM		12 %
ALA		43 %
PCA		50 %
BFC	**Pearson**	**6 %**
BDW		45 %
SM		19 %

Den niedrigsten Validierungsfehler von 6 % erreichte die Kombination BFC als Merkmalsextraktionsmethode und Pearson Korrelation als Merkmalsselektionsmethode (Checkpunkt E18).

Um die benötigte Rechenleistung bei der Ermittlung der Messunsicherheit (Checkpunkt E19) zu reduzieren, wurde ein Messfenster von 5 s betrachtet, welches zusätzlich mittels Downsampling um den Faktor 500 reduziert wurde. Abbildung 5.28 zeigt die Ergebnisse des ML-Modells unter Berücksichtigung der Messunsicherheit des ersten Folds. In diesem wurde das Modell mit den handgeformten und maschinengeschnittenen Daten trainiert und den handgeschnittenen Daten getestet.

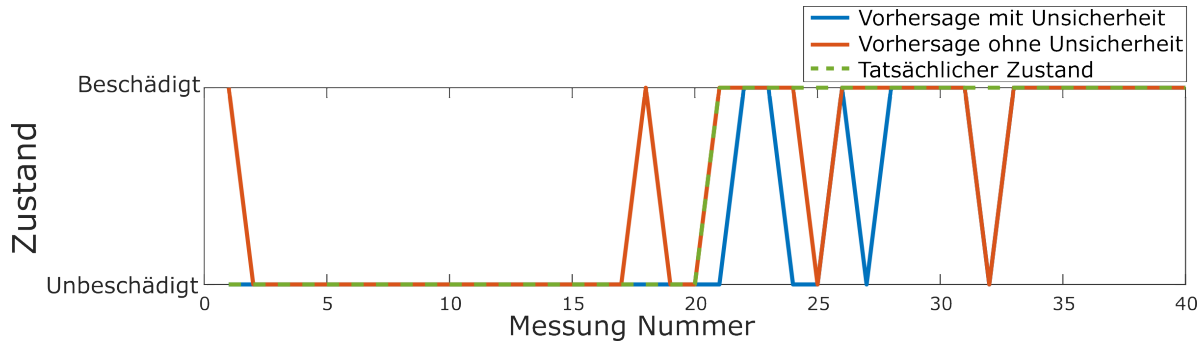


Abbildung 5.28: Darstellung der Unsicherheitsbetrachtung des ersten Folds.

Während der Validierungsfehler ohne Betrachtung der Messunsicherheit bei 10 % liegt, kann dieser unter Berücksichtigung der Messunsicherheit auf 12,5 % ansteigen.

Anschließend wurde das ML-Modell mit den drei Folds Handgeschnitten, Handgeformt und Maschinengeschnitten trainiert und erzielte dabei einen Trainingsfehler von 3 %. Die Anwendung des ML-Modells auf die maschinengeformten Testdaten resultierte in einem Testfehler von 8 %, was auf ein leichtes Overfitting hindeutet.

Das finale Modell wurde mit allen Daten der vier Folds gebildet, um alle möglichen Herstellungsvarianten im Modell abzudecken. Hier ergibt sich unter einer 10-fachen Kreuzvalidierung ein Trainingsfehler von 4 %.

Zusätzlich zum Klassifikationsmodell wurde eine Anomalieerkennung (Checkpunkt E16) zur Erkennung von Abweichungen zum Normalzustand implementiert. Hierzu wurde auf den Daten des unbeschädigten Schrauberbits mit der knn-Anomalieerkennung der ML-Toolbox ein Modell trainiert und mittels Holdout-Validierung überprüft. Abbildung 5.29 zeigt den resultierenden Scatterplot der Messungen des unbeschädigten (blau) und beschädigten (orange) Schrauberbits.

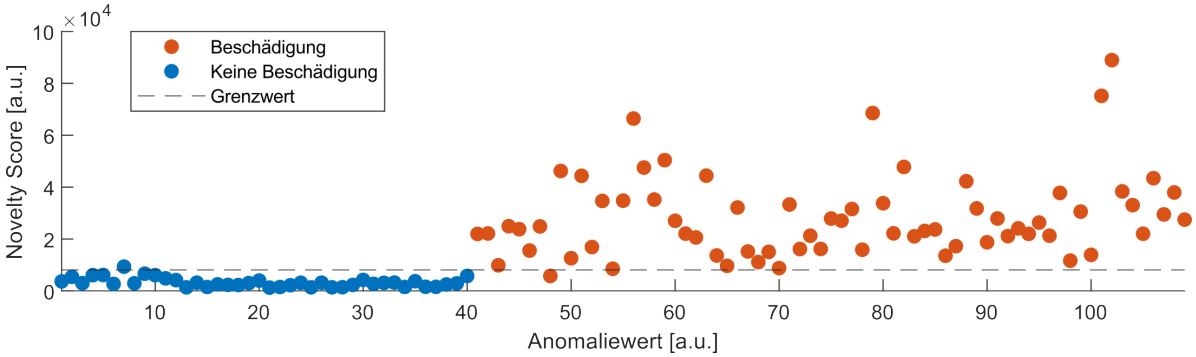


Abbildung 5.29: Scatterplot der Anomalieerkennung mit unbeschädigten (blau) und beschädigten (orange) Messungen.

Das Anomalieerkennungsmodell klassifiziert Messungen als beschädigt, sobald diese die von der ML-Toolbox automatisch bestimmten Grenzwert überschreiten. Dabei erzielte das Modell eine Genauigkeit von 98 %.

Da es sich bei diesem ML-Projekt um eine Machbarkeitsstudie handelte, entfallen die Checkpunkte der Modellanwendung (Checkpunkt E22-E24).

5.3.4.6 Projektabschluss

Abschließend erfolgte der Projektabschluss (Checkpunkt F1-F5). Hierzu wurden zunächst die Projektergebnisse mit den Zielen AF2-Ziel 1-3 abgeglichen:

- **AF2-Ziel 1**: Durch die Anwendung des UWD konnten relevante Stör- und Einflussgrößen für Schraubprozesse ermittelt werden.

- **AF2-Ziel 2**: Die Machbarkeit der Schadensdetektion eines Schrauberbits unter Verwendung eines Beschleunigungssensors konnte erfolgreich nachgewiesen werden.
- **AF2-Ziel 3**: Durch die Verwendung der ML-Toolbox konnte ein ML-Modell erzeugt werden, welches eine hohe Robustheit gegenüber vier verschiedenen Herstellungsvarianten besitzt.

Somit konnten alle vorgegebenen Ziele erreicht werden. Während der Bearbeitung der Machbarkeitsstudie ergab sich zudem folgende Lesson Learned Hypothese (AF2-LL 1): Die Positionen der Gewinde bzw. des Beschleunigungssensors auf der Platte können eine Störgröße darstellen und sollten zukünftig, z.B. durch eine randomisierte Anordnung, berücksichtigt werden. Die weiteren Projektergebnisse wurden in einer Zwischenpräsentation dem Projektkonsortium von VProSaar vorgestellt und entsprechend dokumentiert.

5.3.5 Diskussion und Zwischenfazit

Auch im Anwendungsfall WaMo konnte ein positives Fazit über die Checkliste und PIA gezogen werden. So wurden alle Ziele der Machbarkeitsstudie erreicht und die Ergebnisse können als Grundlage für weitere Messkampagnen genutzt werden. Während der Betreuung der Machbarkeitsstudie konnten jedoch weitere Verbesserungspotenziale der Checkliste bzw. PIA identifiziert werden:

- **Konfigurationen in PIA**: In der aktuellen Implementierung des Software-Demonstrators PIA erfolgt die Eingliederung von Anlagen und Maschinen nach Konfigurationen. Dies könnte bei vielen kleineren Anpassungen bspw. an einer Produktionsanlage zu einer hohen Anzahl an Konfigurationen führen, wodurch PIA unübersichtlich werden könnte. Weiterhin könnte die zusätzliche Implementierung einer Versionierung hilfreich sein.
- **Bearbeitung von Checkpunkten**: Teilweise wurden Checkpunkte von den Projektmitarbeiter nur auf Nachfrage bearbeitet. Dies war insbesondere der Fall, wenn Checkpunkte mit einem erhöhten Arbeitsaufwand verbunden waren und/oder kein unmittelbarer Nutzen für die Projektmitarbeiter ersichtlich war. Hier ist eine intensivere Aufklärung erforderlich.
- **Zusätzliche Inhalte in PIA**: Die Integration von Anleitungen inkl. Beispielen in PIA könnte die Nutzererfahrung verbessern. So erforderte bspw. die Erstellung

des UWD und die Auswertung mittels ML-Toolbox trotz weiterführenden Literaturempfehlungen eine intensivere Betreuung. Mit Hilfe der Anleitungen kann ein Anwender eigenständiger arbeiten und das ML-Projekt effizienter durchführen.

- **FAIR-Indikatoren**: Hinsichtlich der Bewertung der resultierenden FAIRness der Daten zeigten diese ein erhebliches Verbesserungspotenzial in der Dimension der Auffindbarkeit. Hier gilt es jedoch zu berücksichtigen, dass nicht jeder Indikator für die Anwendung im Rahmen einer Machbarkeitsstudie geeignet ist, wie die Erstellung global eindeutiger IDs. Hier können Indikatoren im Kontext des Unternehmens interpretiert werden, wie z.B. durch die Erstellung unternehmensweit eindeutiger IDs anstelle global eindeutiger IDs.

Trotz der oben genannten Punkte stellten die Checkliste und PIA eine wesentliche Unterstützung bei der Bearbeitung der Machbarkeitsstudie dar. So konnten durch die vorgeschlagenen Methoden der Datenvisualisierung die Messposition des Sensors und die Position der Gewinde als weitere mögliche Störgrößen herausgearbeitet werden. Für nachfolgende Messkampagnen kann der Anwender diese berücksichtigen und z.B. die Anordnung der Gewinde (A1-D4) auf der Gewindeplatte randomisieren und den Sensor an mehreren Positionen montieren.

Hinsichtlich der initial gestellten Forschungsfragen konnte der Anwender durch PIA bzw. die integrierte Checkliste dazu befähigt werden, Forschungskonzepte zur Aufzeichnung hochqualitativer Daten anzuwenden und diese aufzuzeichnen (Forschungsfragen 1 und 2). Weiterhin wurde gezeigt, dass die Methodik auch für industrienahe Anwendungsfälle wie das WaMo geeignet ist und Anwender mit geringer Erfahrung in der Bearbeitung von ML-Projekten dazu befähigt werden, Daten mittels ML auszuwerten (Forschungsfrage 3).

6 Fazit

Nachdem in den einzelnen Kapiteln bereits ein Zwischenfazit gezogen wurde, werden nachfolgend die Resultate der Methodik gesamtheitlich diskutiert. Zunächst werden hierzu die initial gestellten Forschungsfragen aufgegriffen und beantwortet:

- **Forschungsfrage 1**: Wie können aktuelle Forschungskonzepte zur Generierung hochwertiger Daten in die Industrie übertragen werden?
  Um aktuelle Forschungskonzepte zur Generierung hochwertiger Daten in die Industrie zu übertragen, wurden ausgewählte Konzepte zunächst in Einzelschritte aufgeteilt, als Checkpunkte formuliert und anschließend in eine praxisorientierte Checkliste integriert. Durch die systematische Aufteilung der Konzepte und deren Zuordnung zu den jeweiligen Schritten eines ML-Projekts wendet der Anwender diese automatisch an, ohne vorher vertieftes Wissen über die Konzepte aufbauen zu müssen. Die Implementierung der Checkliste in den persönlichen Informationsassistenten (PIA) steigert zudem die Nutzerfreundlichkeit, indem Fachwissen (Modul M1) gut zugänglich bereit gestellt wird und gewonnenes Wissen leicht verknüpft werden kann. Somit wurden durch die Checkliste bzw. den PIA erfolgreich mehrere Forschungskonzepte zur Generierung hochwertiger Daten in einen industriellen Kontext übertragen.

- **Forschungsfrage 2**: Wie können unerfahrene industrielle Anwender befähigt werden, hochwertige Daten aufzuzeichnen?
  Unerfahrene industrielle Anwender können durch strukturierte Anleitungen und einen Fokus auf die Mess- und Datenplanung (MuD) dazu befähigt werden, hochwertige Daten aufzuzeichnen. Diese strukturierte Anleitung bietet die entwickelte Checkliste, indem sie einerseits die notwendigen Schritte auflistet und andererseits den Anwender dazu befähigt, die Qualität der Daten anhand vorgeschlagener Metriken zu bewerten. Auch hier steigert die Implementierung der Checkliste in PIA die Nutzerfreundlichkeit, indem der Anwender bspw. Messungen visualisieren kann.

- **Forschungsfrage 3**: Wie können unerfahrene Anwender unterstützt werden, eine Datenanalyse mittels ML durchzuführen?
  Das Konzept des PIA kann durch die Kombination aus zugänglichen Daten und Wissen (Modul M1), einer schrittweisen Unterstützung (Modul M2) sowie einer integrierten, automatisierten ML-Toolbox (Modul M3) unerfahrene industrielle Anwender bei einer Datenanalyse unterstützen. Hierbei führt die Checkliste (Modul M2) den Anwender durch die notwendigen Schritte der Datenanalyse mittels ML, angefangen bei der Datenprüfung und -bereinigung über die Visualisierung bis hin zur Modellbildung und der Validierung des Modells. Die integrierte ML-Toolbox bietet dem Anwender eine komplementäre Auswahl verschiedener ML-Algorithmen und wertet diese automatisiert aus. Sie erfordert dabei keine tiefgreifenden Programmierkenntnisse oder detailliertes Wissen über die implementierten Algorithmen.

Die vorgestellte Checkliste stellt ein essenzielles Element der vorgestellten Methodik und dem PIA dar. Insbesondere verdeutlichte sich in den beiden Anwendungsfällen, dass die Checkliste bereits ausgereifter ist als der PIA. Dies liegt darin begründet, dass die Checkliste als eigenständiges Dokument bereits einem breiten Publikum präsentiert werden konnte und so in iterativen Schritten regelmäßig Feedback aus Wissenschaft und Industrie eingearbeitet werden konnte. Das Konzept des PIA hingegen konnte im Rahmen dieser Arbeit lediglich als rudimentärer Software-Demonstrator implementiert werden und zeigt einen hohen Entwicklungsbedarf, z.B. durch das fehlende Backend, die manuell einzutragenden Metadaten, fehlende Schnittstellen und die Notwendigkeit einer höheren Nutzerfreundlichkeit. Denkbar wäre auch die Implementierung eines unterstützenden Chatbots wie z.B. ChatGPT [183]. Dieser bietet, trotz aktuell limitierender Faktoren wie z.B. fehlendem domänen-spezifischem Wissen und den daraus resultierenden fehlerhaften Antworten [184], bereits ein hohes Potenzial für zahlreiche Branchen [185]. Weiterhin könnte ChatGPT auch den Prozess der Datenanalyse, z.B. durch Unterstützung bei der Modellbildung und Interpretation von Code und Daten, vereinfachen [90]. Im Rahmen der Dissertation wurde ChatGPT in der aktuellen Version GPT-4 aufgrund der aktuell mangelhaften Zuverlässigkeit, Korrektheit und einem Bias in den Antworten (vgl. [90, 184]) für den PIA nicht in Betracht gezogen.

Die implementierte ML-Toolbox des Lehrstuhls für Messtechnik stellt eine einsteigerfreundliche Option dar, um eine Datenanalyse mit ML durchzuführen. Bei besonders komplexen ML-Problemen kann die ML-Toolbox an ihre Grenzen stoßen und ggf. keine zufriedenstellenden Modelle erzeugen. Jedoch konnte diese bereits in

einem breiten Anwendungsspektrum komplexe Probleme lösen [141], wodurch sie für den Großteil der industriellen Anwendungsfälle geeignet ist. Weiterhin erfordert die ML-Toolbox im Vergleich zu tiefen neuronalen Netzen geringere Hardwareressourcen in der Modellbildung [141]. Es gilt zu berücksichtigen, dass Lernprobleme möglicherweise nicht oder nicht auf Basis aufgezeichneten Daten lösbar sind.

Die Anwendung der Checkliste und des PIA verdeutlichen die Flexibilität und Wirksamkeit der vorgestellten Methodik im wissenschaftlichen und industrienahen Kontext. In beiden Anwendungsfällen konnte das ML-Projekt erfolgreich abgeschlossen werden und Daten mit hoher Qualität aufgezeichnet werden. Obwohl Fehler im ML-Projekt nicht vollständig verhindert werden konnten, sind diese jedoch durch kontrollierende Checkpunkte zeitnah von den Projektmitarbeitern entdeckt worden und entsprechende Gegenmaßnahmen konnten eingeleitet werden. Weiterhin entstanden bei der Bearbeitung der Anwendungsfälle mehrere Fehler durch Projektmitarbeiter, insbesondere bei arbeitsaufwändigen Tätigkeiten oder Tätigkeiten, deren Nutzen nicht direkt ersichtlich war. Diese hätten durch die strikte Befolgung der Checkliste vermieden werden können. Hier könnte eine gezielte Berücksichtigung von Aspekten des Change Managements Abhilfe schaffen, indem Projektmitarbeiter stärker in den (Veränderungs-)Prozess eingebunden werden [132].

Die vorgestellten Methoden zur Beurteilung der Datenqualität decken in der Wissenschaft etablierte Standards wie bpsw. FAIR-Data ab. Allerdings sind nicht alle der vorgestellten Methoden und Metriken optimal für den unmittelbaren industriellen Einsatz, wie bspw. die FAIR-Indikatoren. Sie können Anwendern jedoch als Orientierungshilfe für relevante Metriken dienen, oder von diesen entsprechend angepasst werden. Um eine belastbare Aussage über die Methodik treffen zu können, sollte die Anwendung der Methodik im nächsten Schritt in einem ML-Projekt eines mittelständischen Unternehmens erfolgen.

Die übergeordnete Frage, wie Unternehmen im Mittelstand maschinelle Lernprojekte ohne spezialisierte ML-Fachkräfte realisieren können, wurde durch die Entwicklung der Checkliste und PIA beantwortet. Durch die Integration aktueller Forschungskonzepte in die Checkliste sowie deren ganzheitlich unterstützender Ansatz befähigen Anwender, eine Datenanalyse mit ML durchzuführen. Die Einbettung der Checkliste in PIA steigert die Nutzerfreundlichkeit und gewährt dem Anwender zudem Zugang zu Fachwissen und der automatisierten ML-Toolbox. Dies ermöglicht Anwendern im Mittelstand durch die Verwendung von PIA ein ML-Projekt zu realisieren, ohne auf spezialisierte Fachexperten angewiesen zu sein.

7 Zusammenfassung und Ausblick

In der vorliegenden Arbeit wurde ein persönlicher Informationsassistent (PIA) entwickelt, mit dessen Hilfe mittelständische Unternehmen ein Datenanalyseprojekt mit maschinellem Lernen (ML) durchführen können, ohne auf spezialisierte ML-Fachkräfte angewiesen zu sein.

Kern der vorgestellten Methodik ist die Checkliste, welche Anwender von der Vorbereitung bis zum Projektabschluss begleitet. Dabei wird der Anwender in die Thematik der jeweiligen Kapitel der Checkliste durch einen einleitenden Abschnitt eingeführt und erhält anschließend eine Übersicht von Muss-Checkpunkten, deren Bearbeitung zwingend erforderlich ist, und Best-Practice-Checkpunkten, deren Bearbeitung empfohlen wird, aber nicht zwangsweise erfolgen muss. In diese Checkpunkte wurden aktuelle Forschungskonzepte wie z.B. die FAIR-Prinzipien und Messunsicherheitsbetrachtungen schrittweise integriert. Dies ermöglicht die Anwendung der Konzepte, ohne tiefgreifendes Wissen vorauszusetzen. Die Checkliste legt dabei den Fokus auf die Mess- und Datenplanung, um Anwender zu befähigen, hochwertige Daten aufzuzeichnen. Neben den Checkpunkten bietet die Checkliste Tipps, Hinweise und weiterführende Literatur, die den Anwender zusätzlich unterstützen. Der ergänzende Ablaufplan bietet dem Anwender Orientierung bei der Durchführung eines ML-Projektes und verdeutlicht u.a. iterative Abläufe und Schleifen.

Weiterhin wurde das Konzept PIA eingeführt und in Form eines webbasierten Softwaredemonstrators umgesetzt. PIA stützt sich auf die drei Module:

- **Zugänglichkeit von Daten und Wissen**: Die Zugänglichkeit von Daten und deren Metadaten sowie von (Fach-)Wissen ermöglicht dem Anwender eine effiziente Bearbeitung des ML-Projektes.
- **Unterstützung des Anwenders**: Der Anwender wird bei der Bearbeitung seines ML-Projektes in allen Schritten unterstützt. Im vorgestellten Softwaredemonstrator wurde hierzu die Checkliste und der Ablaufplan in PIA implementiert.

- **Datenanalyse**: Der Anwender wird bei der Datenanalyse durch die in PIA integrierte *Automatisierte ML-Toolbox für zyklische Daten* (ML-Toolbox) des Lehrstuhls für Messtechnik unterstützt. Diese wählt basierend auf einer Kreuzvalidierung automatisiert die Algorithmenkombination mit dem geringsten Validierungsfehler aus fünf komplementären Merkmalsextraktions- und drei Merkmalsselektionsmethoden aus. Anschließend erfolgt eine Klassifikation mit einer Lineare Diskriminanzanalyse (LDA) mit Mahalanobis-Distanzmaß. Diese Kombination konnte ihre breite Nutzbarkeit bereits in verschiedensten Anwendungsszenarien zeigen [141].

PIA stellt somit ein Konzept dar, welches den Anwender ganzheitlich bei der Bearbeitung eines ML-Projekts begleitet und unterstützt.

Anschließend wurde die vorgestellte Methodik in zwei Anwendungsfällen evaluiert. Der erste Anwendungsfall befasste sich mit der Schadenserkennung an Zylinderrollenlagern an einem Prüfstand des Lehrstuhls für Messtechnik und wurde primär vom Autor dieser Dissertation bearbeitet. Ziel war die Erzeugung eines ML-Modells, welches robust gegenüber gängigen Störgrößen und insbesondere der Einbauposition der Zylinderrollenlager ist. Trotz des wissenschaftlichen Charakters des Anwendungsfalles konnte für die vorgestellte Methodik ein erstes positives Fazit gezogen werden. So konnte ein robustes ML-Modell erzeugt werden, welches Lagerschäden unabhängig von der Einbauposition erkennt. Weiterhin konnten verschiedene Probleme durch die Bearbeitung der Checkpunkte aufgedeckt werden, wie bspw. Fehler in der Datenaufzeichnung. Dies verdeutlicht die universelle Anwendbarkeit der vorgestellten Methodik, wenngleich leichte Anpassungen bei bspw. Muss-Checkpunkten der Checkliste erforderlich waren.

Im zweiten Anwendungsfall erfolgte die Anwendung der Methodik im Rahmen eines ML-Projekts an der Schraubstation des wandelbaren Montagesystems (WaMo), einer Fertigungslinie mit zwei Stationen am Zentrum für Mechatronik und Automatisierungstechnik gGmbH (ZeMA). Ziel war die Überprüfung der Machbarkeit einer Schadenserkennung des Schrauberbits in der Schraubstation unter Verwendung eines Beschleunigungssensors. Auch hier soll das ML-Modell robust gegenüber gängigen Störgrößen sein. Dieser industrienahe Anwendungsfall wurde in Kooperation mit dem Lehrstuhl für Montagesysteme der Universität des Saarlandes im Projekt *Verteilte Produktion für die saarländische Automotivindustrie: Nachhaltig, Vernetzt, Resilient* (VProSaar) bearbeitet. Hier war der Autor dieser Dissertation lediglich in unterstützender und beratender Funktion tätig. So konnte die Methodik realitätsnah durch Anwender ohne ML-Expertise angewendet und anschließend durch den Autor

evaluiert werden. Auch im zweiten Anwendungsfall konnte durch die vorgestellte Methodik die Machbarkeitsstudie erfolgreich durchgeführt, ein robustes ML-Modell erzeugt und vertieftes Wissen gewonnen werden.

Bei der Bearbeitung der Anwendungsfälle zeigten sich jedoch mehrere Verbesserungspotenziale, welche gleichzeitig den Ausblick auf zukünftige Arbeiten geben:

- **Unterstützung in PIA**: Trotz weiterführender Literatur erforderten einige Checkpunkte die Unterstützung des Autors. Hier ist eine tiefergreifende Unterstützung der Anwender, z.B. in Form kurzer Tutorials mit Beispielen, erforderlich.

- **Fehlendes Backend im Softwaredemonstrator**: Durch das fehlende Backend des Softwaredemonstrators PIA konnte dessen Potenzial nicht voll ausgeschöpft werden. So war u.a. die Integration der Informationen in PIA auf Code-Ebene erforderlich und das Speichern des Fortschritts in der Checkliste nicht möglich.

- **Aufbau der Checkliste**: Die Checkliste wurde von den Anwendern sequentiell abgearbeitet. Obwohl der Ablaufplan andeutet, dass innerhalb gewisser Checkpunkte-Gruppen, wie z.B. Visualisierung der Daten, die Checkpunkte nicht zwangsweise sequentiell abzuarbeiten sind, halten sich die Anwender an die vorgegebene Reihenfolge. Je nach Gegebenheiten des Anwendungsfalles kann jedoch die Bearbeitungsreihenfolge abweichen. In einer folgenden Iteration der Checkliste muss dem Anwender klar ersichtlich sein, welche Checkpunkte sequentiell abgearbeitet werden müssen und welche parallel bearbeitet werden können. Dies kann bpsw. durch eine visuelle Hierarchie-Ebene mit Aufzählungszeichen erfolgen. Weiterhin gilt es die Reihenfolge der Checkpunkte zu überprüfen und ggf. zu optimieren.

- **Aufwendige Checkpunkte**: Der Arbeitsaufwand der Checkpunkte kann stark variieren. Checkpunkte, die von den Anwendern als aufwendig empfunden werden oder deren Nutzen nicht unmittelbar erkennbar ist, werden tendenziell mit einer geringeren Sorgfalt ausgeführt. Hier gilt es, entsprechende Gegenmaßnahmen in der Checkliste bzw. in PIA zu ergreifen, wie z.B. eine erweiterte Aufklärung oder die Aufteilung von Checkpunkten.

- **Lösbare Probleme**: Die Checkliste setzt voraus, dass ein Problem immer lösbar ist. Hier muss auch darauf eingegangen werden, dass Probleme auftreten können, die mit den verfügbaren Daten und Modellen nicht lösbar sind.

Aufgrund der genannten Verbesserungspotenziale sollte die Checkliste in einer weiteren Iteration angepasst werden. Weitere Verbesserungspotenziale finden sich im Anhang in Tabelle A.10.

Nachdem die Methodik zuerst in einem wissenschaftlichen und anschließend in einem industrienahen Anwendungsfall evaluiert wurde, gilt es, diese in einem echten Anwendungsfall eines mittelständischen Unternehmens anzuwenden und zu evaluieren. Hier erfolgten bereits erste Schritte durch die Publikation der Checkliste [129] und PIA [122]. Weiterhin wurde insbesondere die Checkliste im Rahmen des Projektes *Mittelstand-Digital Zentrum Saarbrücken* mittelständischen Unternehmen zur Verfügung gestellt und konnte in durchgeführten Sprechstunden positives Feedback erzeugen. Die Bereitschaft der mittelständischen Unternehmen, die Methodik in einem Projekt zu evaluieren, war im Rahmen dieser Dissertation nicht gegeben. Gründe hierfür waren u.a. der Schutz von Daten und Informationen, die aktuell schlechte wirtschaftliche Lage oder der Mangel an personellen und finanziellen Mitteln.

Zusammenfassend kann die vorgestellte Methodik der Checkliste und PIA Anwender im Mittelstand dazu befähigen, ML-Projekte ohne spezialisierte ML-Fachkräfte durchzuführen, dadurch effizient Verbesserungen zu erzielen und ihre Wettbewerbsfähigkeit zu steigern.

Literaturverzeichnis

[1] Philipp Gönnheimer, Markus Netzer, Carolin Lange, Roman Dörflinger, Judith Armbruster und Jürgen Fleischer. „Datenaufnahme und -verarbeitung in der Brownfield-Produktion“. *Zeitschrift für wirtschaftlichen Fabrikbetrieb* 117.5 (2022), S. 317–320. DOI: 10.1515/zwf-2022-1062.

[2] Ravi Kant und Hema Gurung, Hrsg. *Industry 4.0: Concepts, Processes and Systems*. Boca Raton, FL, USA: CRC Press, 2023. ISBN: 978-1-032-15949-2.

[3] IDG. *Studie Machine Learning / Deep Learning 2019*. 2019. URL: https://www.lufthansa-industry-solutions.com/de-en/studien/idg-study-machine-learning-2019 (besucht am 21. 10. 2024).

[4] Christopher Schnur, Steffen Klein, Tizian Schneider, Andreas Schütze, Andreas Blum und Ralf Müller. „Mess- und Datenplanung für Modelle des maschinellen Lernens an Bestandsanlagen“. *Proceedings of the 16. Dresdner Sensor-Symposium, 5-7 December 2022*. Dresden, Germany: AMA Service GmbH, 2022. DOI: 10.5162/16dss2022/P47.

[5] Bundesministerium für Bildung und Forschung. *Nationale Forschungsdateninfrastruktur*. 2023. URL: https://www.bmbf.de/bmbf/de/forschung/das-wissenschaftssystem/nationale-forschungsdateninfrastruktur/nationale-forschungsdateninfrastruktur_node.html (besucht am 21. 10. 2024).

[6] Mark D. Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak und Niklas Blomberg. „The FAIR Guiding Principles for scientific data management and stewardship“. *Scientific Data* 3.1 (2016). DOI: 10.1038/sdata.2016.18.

[7] Peter Chapman, Janet Clinton, Randy Kerber, Tom Khabaza, Thomas P. Reinartz, Colin Shearer und Richard Wirth. *CRISP-DM 1.0: Step-by-step data mining guide*. 2000. URL: https://www.kde.cs.uni-kassel.de/wp-content/uploads/lehre/ws2012-13/kdd/files/CRISPWP-0800.pdf (besucht am 21. 10. 2024).

[8] Tizian Schneider, Nikolai Helwig und Andreas Schütze. „Industrial condition monitoring with smart sensors using automated feature extraction and selection“. *Measurement Science and Technology* 29.9 (2018), S. 094002. DOI: 10.1088/1361-6501/aad1d4.

[9] Herfurth und Partner. *Industrie 4.0 Eckpunkte*. 2016. URL: https://www.herfurth.de/wp-content/uploads/2021/03/industrie-4-0-eckpunkte-indy4-2-aufl-2016.pdf (besucht am 21. 10. 2024).

[10] Uwe Küppers. „Die Zukunft der Produktion liegt in den Daten der Vergangenheit“. *Digitale Welt* 5.3 (2021), 72—74. DOI: 10.1007/s42354-021-0381-1.

[11] Yannick Wilhelm, Ulf Schreier, Peter Reimann, Bernhard Mitschang und Holger Ziekow. „Data Science Approaches to Quality Control in Manufacturing: A Review of Problems, Challenges and Architecture“. *Service-Oriented Computing*. Springer International Publishing, 2020. DOI: 10.1007/978-3-030-64846-6_4.

[12] Nathalie Hartl. *Industry Engagement: Speaker elected and section concept published | NFDI*. 2023. URL: https://www.nfdi.de/industry-engagement-speaker-elected-and-section-concept-published/?lang=en (besucht am 21. 10. 2024).

[13] Stephan Matzka. *Crashkurs KI im Unternehmen: Alles, was Sie über Data Science wissen müssen*. Freiburg, Deutschland: Haufe Group, 2021. ISBN: 978-3-648-14920-1.

[14] Lina Bruns, Benjamin Dittwald und Fritz Meiners. *Leitfaden für qualitativ hochwertige Daten und Metadaten*. 2019. URL: https://cdn0.scrvt.com/fokus/e472f1bf447f370f/32c99a36d8b3/NQDM_Leitfaden-f-r-qualitativ-hochwertige-Daten-und-Metadaten_2019.pdf (besucht am 21. 10. 2024).

[15] Joachim Metternich, Tobias Biegel, Beatriz Bretones Cassoli, Felix Hoffmann, Nicolas Jourdan, Jannik Rosemeyer, Patrick Stanula und Amina Ziegenbein. *Künstliche Intelligenz zur Umsetzung von Industrie 4.0 im Mittelstand: Expertise des Forschungsbeirats der Plattform Industrie 4.0*. München, Deutschland: acatech – Deutsche Akademie der Technikwissenschaften, 2021.

[16] Markus Röhler und Sajedeh Haghi. *Leitfaden Künstliche Intelligenz – Potenziale und Umsetzungen im Mittelstand*. München, Deutschland: VDMA Bayern, 2020.

[17] Natasha Simons et al. *The State of Open Data 2021*. Techn. Ber. Digital Science, 2021. DOI: 10.6084/m9.figshare.17061347.v1.

[18] Philippe Rocca-Serra et al. *The FAIR Cookbook*. Version 0.1.0. 2022. URL: https://github.com/FAIRplus/the-fair-cookbook/ (besucht am 21. 10. 2024).

[19] FAIR Data Maturity Model Working Group. *FAIR Data Maturity Model. Specification and Guidelines*. 2020. DOI: 10.15497/rda00050.

[20] Tanja Dorst, Maximilian Gruber, Anupam P. Vedurmudi, Daniel Hutzschenreuter, Sascha Eichstädt und Andreas Schütze. „A case study on providing FAIR and metrologically traceable data sets“. *Acta IMEKO* 12.1 (2023). DOI: 10.21014/actaimeko.v12i1.1401.

[21] Jiawei Han, Jian Pei und Hanghang Tong. *Data Mining: Concepts and Techniques*. San Francisco, CA, USA: Morgan Kaufmann, 2022. ISBN: 978-0-12-811760-6.

[22] Usama Fayyad, Gregory Piatetsky-Shapiro und Padhraic Smyth. „From data mining to knowledge discovery in databases“. *AI magazine* 17.3 (1996), S. 37–54. DOI: 10.1609/aimag.v17i3.1230.

[23] Ana Azevedo und Manuel Filipe Santos. „KDD, SEMMA and CRISP-DM: A Parallel Overview“. *Proceedings of the IADIS European Conference on Data Mining, 24-26 July 2008*. Amsterdam, Netherlands: IADIS, 2008, S. 182–185. ISBN: 978-972-8924-63-8.

[24] Gonzalo Mariscal, Óscar Marbán und Covadonga Fernández. „A survey of data mining and knowledge discovery process models and methodologies“. *The Knowledge Engineering Review* 25.2 (2010), S. 137–166. DOI: 10.1017/S0269888910000032.

[25] Leonie Mende. „Merkmalentstehungs- und -wechselwirkungsanalyse (MEWA) für das prozessorientierte Toleranzmanagement in der Montage“. Dissertation. Universität des Saarlandes, Naturwissenschaftlich-Technische Fakultät, 2020. DOI: 10.22028/D291-32507.

[26] Jörg Niemann, Benedikt Reich und Carsten Stöhr. *Lean Six Sigma: Methoden zur Produktionsoptimierung*. Heidelberg, Deutschland: Springer-Verlag, 2021. ISBN: 978-3-662-63007-5.

[27] Staatskanzlei Saarland. *Künstliche Intelligenz und Industrie 4.0: Digitale Technologien für die Zukunft*. 2020. URL: https://www.saarland.de/DE/presse-informationen/medienservice/pressearchiv/stk/stk-medieninfo-archive/2020/Q4_2020/pm_2020-12-11-Ki-Industrie4-0 (besucht am 21. 10. 2024).

[28] Kaoru Ishikawa, Hrsg. *Guide to Quality Control*. Industrial Engineering & Technology. Tokyo, Japan: Asian Productivity Organization, 1976. ISBN: 978-92-833-1035-8.

[29] Pete Barbrook-Johnson und Alexandra S. Penn. *Systems Mapping: How to build and use causal models of systems*. Cham, Switzerland: Springer International Publishing, 2022. ISBN: 978-3-031-01919-7.

[30] K. Thulasiraman und M. N. S. Swamy. *Graphs: Theory and Algorithms*. New York, NY, USA: John Wiley & Sons, 1992. ISBN: 978-8-126-54958-0.

[31] B. Vogel-Heuser, V. Karaseva, J. Folmer und I. Kirchen. „Operator Knowledge Inclusion in Data-Mining Approaches for Product Quality Assurance using Cause-Effect Graphs“. *IFAC-PapersOnLine* 50.1 (2017), S. 1358–1365. DOI: 10.1016/j.ifacol.2017.08.233.

[32] Dale H Besterfield, Carol Besterfield-Michna, Glen H Besterfield, Mary Besterfield-Sacre, Hemant Urdhwareshe und Rashmi Urdhwareshe. *Total Quality Management Revised Edition: For Anna University*. New Delhi, India: Pearson Education India, 2012. ISBN: 978-8-131-76496-1.

[33] Holger Brüggemann und Peik Bremer. *Grundlagen Qualitätsmanagement: von den Werkzeugen über Methoden zum TQM*. Wiesbaden, Germany: Springer Vieweg, 2015. ISBN: 978-3-658-09220-7.

[34] Nancy R. Tague. *The Quality Toolbox*. 2nd. Milwaukee, WI, USA: ASQ Quality Press, 2005. ISBN: 978-0-87389-639-9.

[35] Wilhelm Kleppmann. *Versuchsplanung: Produkte und Prozesse optimieren*. 10. Auflage. München, Deutschland: Carl Hanser Verlag, 2020. ISBN: 978-3-446-46146-8.

[36] Karl Siebertz, David van Bebber und Thomas Hochkirchen. *Statistische Versuchsplanung: Design of Experiments (DoE)*. Berlin, Deutschland: Springer Vieweg, 2017. ISBN: 978-3-662-55742-6.

[37] Bradley Jones und Douglas C. Montgomery. *Design of Experiments: A Modern Approach*. Hoboken, NJ, USA: Wiley, 2020. ISBN: 978-1-119-74601-0.

[38] Guido Walz, Hrsg. *Lexikon der Mathematik: Band 4*. Berlin, Deutschland: Springer-Verlag, 2017. ISBN: 978-3-662-53500-4.

[39] Uwe Reinert, Herbert Blaschke und Uwe Brockstieger. *Technische Statistik in der Qualitätssicherung: Grundlagen für Produktions- und Verfahrenstechnik*. Berlin, Deutschland: Springer-Verlag, 1999. ISBN: 978-3-540-64107-0.

[40] Lothar Papula. *Mathematik für Ingenieure und Naturwissenschaftler. Band 3: Vektoranalysis, Wahrscheinlichkeitsrechnung, mathematische Statistik, Fehler- und Ausgleichsrechnung*. 7. Auflage. Berlin, Deutschland: Springer Vieweg, 2016. ISBN: 978-3-658-11924-9.

[41] Armin Töpfer, Hrsg. *Six Sigma: Konzeption und Erfolgsbeispiele für praktizierte Null-Fehler-Qualität*. Berlin, Deutschland: Springer-Verlag, 2007. ISBN: 978-3-540-48591-9.

[42] Project Management Institute. *A Guide to the Project Management Body of Knowledge (PMBOK Guide)*. 7th. Newtown Square, PA, USA: Project Management Institute, 2021. ISBN: 978-1-628-25664-2.

[43] Michael Quinn Patton. „Evaluation, Knowledge Management, Best Practices, and High Quality Lessons Learned“. *American Journal of Evaluation* 22.3 (2001), S. 329–336. DOI: 10.1177/109821400102200307.

[44] Sandra F. Rowe. „Lessons Learned: Taking It to the Next Level“. *PMI Global Congress 2007—EMEA, 14-16 May 2007*. Budapest, Hungary: Project Management Institute, 2007.

[45] US Department of Energy. *DOE-STD-7501-99 The DOE Corporate Lessons Learned Program*. 1999. URL: https://www.standards.doe.gov/standards-documents/7000/7501-astd-1999/@@images/file (besucht am 21. 10. 2024).

[46] George H. Labovitz, Yu Sang Chang und Victor Rosansky. *Making Quality Work: A Leadership Guide for the Results-Driven Manager*. New York, NY, USA: HarperBusiness, 1993. ISBN: 978-0-88730-582-5.

[47] Ekbert Hering und Gert Schönfelder, Hrsg. *Sensoren in Wissenschaft und Technik*. Wiesbaden, Germany: Springer Fachmedien, 2018. ISBN: 978-3-658-12561-5.

[48] Hui Yie Teh, Andreas W. Kempa-Liehr und Kevin I-Kai Wang. „Sensor data quality: a systematic review“. *Journal of Big Data* 7.1 (2020). DOI: 10.1186/s40537-020-0285-1.

[49] BIPM, IEC, IFCC, ILAC, ISO, IUPAC, IUPAP und OIML. *Guide to the expression of uncertainty in measurement — Part 1: Introduction*. 2008. URL: https://www.bipm.org/documents/20126/2071204/JCGM_100_2008_E.pdf/cb0ef43f-baa5-11cf-3f85-4dcd86f77bd6 (besucht am 21. 10. 2024).

[50] T. Dorst, T. Schneider, S. Eichstädt und A. Schütze. „Influence of measurement uncertainty on machine learning results demonstrated for a smart gas sensor“. *Journal of Sensors and Sensor Systems* 12.1 (2023), S. 45–60. DOI: 10.5194/jsss-12-45-2023.

[51] BIPM, IEC, IFCC, ILAC, ISO, IUPAC, IUPAP und OIML. *JCGM 200: International Vocabulary of Metrology - Basic and General Concepts and Associated Terms*. 2012. URL: https://www.bipm.org/documents/20126/2071204/JCGM_200_2012.pdf/f0e1ad45-d337-bbeb-53a6-15fe649d0ff1 (besucht am 21. 10. 2024).

[52] Tanja Dorst. „Measurement uncertainty in machine learning – uncertainty propagation and influence on performance“. Dissertation. Universität des Saarlandes, Naturwissenschaftlich-Technische Fakultät, 2023. DOI: 10.22028/D291-40173.

[53] BIPM, IEC, IFCC, ILAC, ISO, IUPAC, IUPAP und OIML. *JCGM 101: Evaluation of measurement data – Supplement 1 to the "Guide to the expression of uncertainty in measurement" – Propagation of distributions using a Monte Carlo method*. 2008. URL: https://www.bipm.org/documents/20126/2071204/JCGM_101_2008_E.pdf/325dcaad-c15a-407c-1105-8b7f322d651c (besucht am 21. 10. 2024).

[54] BIPM, IEC, IFCC, ILAC, ISO, IUPAC, IUPAP und OIML. *JCGM 102: Evaluation of measurement data – Supplement 2 to the "Guide to the expression of uncertainty in measurement" – Extension to any number of output quantities*. 2011. URL: https://www.bipm.org/documents/20126/2071204/JCGM_102_2011_E.pdf/6a3281aa-1397-d703-d7a1-a8d58c9bf2a5 (besucht am 21. 10. 2024).

[55] BIPM, IEC, IFCC, ILAC, ISO, IUPAC, IUPAP und OIML. *Guide to the expression of uncertainty in measurement — Part 1: Introduction*. Techn. Ber. 2023. DOI: 10.59161/JCGMGUM-1-2023.

[56] Makx Dekkers, Michiel De Keyzer, Nikolaos Loutas und Stijn Goedertier. *Einführung in das Metadaten-Management*. 2013. URL: https://data.europa.eu/sites/default/files/d2.1.2_training_module_1.4_introduction_to_metadata_management_de_edp.pdf (besucht am 21. 10. 2024).

[57] Terrence Brooks. „World wide web consortium (W3C)“. *Encyclopedia of library and information sciences*. CRC Press, 2009. ISBN: 978-0-203-75763-5.

[58] Stuart L. Weibel und Traugott Koch. „The Dublin core metadata initiative“. *D-lib magazine* 6.12 (2000). DOI: 10.1045/december2000-weibel.

[59] Susanna-Assunta Sansone, Peter McQuilton, Philippe Rocca-Serra, Alejandra Gonzalez-Beltran, Massimiliano Izzo, Allyson L. Lister, Milo Thurston und the FAIRsharing Community. „FAIRsharing as a community approach to standards, repositories and policies“. *Nature Biotechnology* 37.4 (2019), 358—367. DOI: 10.1038/s41587-019-0080-8.

[60] W3C. *Semantic Sensor Network Ontology*. Techn. Ber. 2021. URL: https://www.w3.org/TR/vocab-ssn/ (besucht am 21. 10. 2024).

[61] DCMI Usage Board. *DCMI Metadata Terms*. Techn. Ber. Dublin Core Metadata Initiative, 2020. URL: http://dublincore.org/specifications/dublin-core/dcmi-terms/2020-01-20/ (besucht am 21. 10. 2024).

[62] Fabiano B. Ruy, Cássio C. Reginato, Victor A. Santos, Ricardo A. Falbo und Giancarlo Guizzardi. „Ontology Engineering by Combining Ontology Patterns“. *Conceptual Modeling*. Springer International Publishing, 2015. ISBN: 978-3-319-25264-3.

[63] Susanne Arndt et al. *Metadata4Ing: An ontology for describing the generation of research data within a scientific activity*. 2022. DOI: 10.5281/zenodo.7706017.

[64] Holger Hinrichs. „Datenqualitätsmanagement in data warehouse-systemen“. Dissertation. Universität Oldenburg, Fakultät II - Informatik, Wirtschafts- und Rechtswissenschaften, 2002.

[65] Richard Y. Wang und Diane M. Strong. „Beyond Accuracy: What Data Quality Means to Data Consumers“. *Journal of Management Information Systems* 12.4 (1996), S. 5–33. DOI: 10.1080/07421222.1996.11518099.

[66] Bernd Heinrich und Mathias Klier. „Datenqualitätsmetriken für ein ökonomisch orientiertes Qualitätsmanagement“. *Daten- und Informationsqualität: Auf dem Weg zur Information Excellence*. Vieweg+Teubner, 2011. DOI: 10.1007/978-3-8348-9953-8_3.

[67] Knut Hildebrand, Marcus Gebauer und Michael Mielke, Hrsg. *Daten- und Informationsqualität: Die Grundlage der Digitalisierung*. Wiesbaden, Germany: Springer Fachmedien, 2021. ISBN: 978-3-658-30990-9.

[68] Salvador García, Julián Luengo und Francisco Herrera. *Data Preprocessing in Data Mining*. Bd. 72. Cham, Switzerland: Springer, 2015. ISBN: 978-3-319-10247-4.

[69] Dorian Pyle. *Data Preparation for Data Mining*. San Francisco, CA, USA: Morgan Kaufmann Publishers, 1999. ISBN: 978-1-55860-529-9.

[70] Heinrich Ruser und Fernando León. „Informationsfusion – Eine Übersicht“. *tm - Technisches Messen* 74 (2007), S. 93–102. DOI: 10.1524/teme.2007.74.3.93.

[71] Antony Unwin. „Why is data visualization important? what is important in data visualization?“ *Harvard Data Science Review* 2.1 (2020). DOI: 10.1162/99608f92.8ae4d525.

[72] Stefan Papp et al. *Handbuch Data Science und KI: Mit AI, Datenanalyse und Machine Learning Wert aus Daten generieren*. 2. Auflage. München, Deutschland: Hanser, 2022. ISBN: 978-3-446-46947-1.

[73] Thomas A. Runkler. *Data Mining: Modelle und Algorithmen intelligenter Datenanalyse*. Wiesbaden, Deutschland: Springer Fachmedien, 2015. ISBN: 978-3-8348-1694-8.

[74] Herbert A. Sturges. „The Choice of a Class Interval“. *Journal of the American Statistical Association* 21.153 (1926), 65—66. DOI: 10.1080/01621459.1926.10502161.

[75] DAVID W. SCOTT. „On optimal and data-based histograms“. *Biometrika* 66.3 (1979), 605—610. DOI: 10.1093/biomet/66.3.605.

[76] Neil C Schwertman, Margaret Ann Owens und Robiah Adnan. „A simple more general boxplot method for identifying outliers“. *Computational Statistics and Data Analysis* 47.1 (2004), S. 165–174. DOI: https://doi.org/10.1016/j.csda.2003.10.012.

[77] Markus Ringnér. „What is principal component analysis?“ *Nature Biotechnology* 26.3 (2008), 303—304. DOI: 10.1038/nbt0308-303.

[78] I. T. Jolliffe. *Principal Component Analysis*. Springer Series in Statistics. New York, NY, USA: Springer, 2002. ISBN: 9780387954424.

[79] Rasmus Bro und Age K. Smilde. „Principal component analysis“. *Anal. Methods* 6.9 (2014), S. 2812–2831. DOI: 10.1039/C3AY41907J.

[80] Ronald Aylmer Fisher. „The use of multiple measurements in taxonomic problems“. *Annals of Eugenics* 7.2 (1936), S. 179–188. DOI: 10.1111/j.1469-1809.1936.tb02137.x.

[81] Richard O. Duda, Peter E. Hart und David G. Stork. *Pattern Classification*. 2. Aufl. Hoboken, NJ, USA: John Wiley & Sons, 2001. ISBN: 978-0-471-05669-0.

[82] Aleix M. Martínez und Avinash C. Kak. „PCA versus LDA“. *IEEE Trans. Pattern Anal. Mach. Intell.* 23 (2001), S. 228–233. DOI: 10.1109/34.908974.

[83] Dijun Luo, Chris Ding und Heng Huang. „Linear Discriminant Analysis: New Formulations and Overfit Analysis“. *Proceedings of the AAAI Conference on Artificial Intelligence* 25.1 (2011), S. 417–422. DOI: 10.1609/aaai.v25i1.7926.

[84] Nikolai J. Helwig. „Zustandsbewertung industrieller Prozesse mittels multivariater Sensordatenanalyse am Beispiel hydraulischer und elektromechanischer Antriebssysteme“. Dissertation. Universität des Saarlandes, Naturwissenschaftlich-Technische Fakultät, 2018. DOI: 10.22028/D291-27896.

[85] M. Bastuck, T. Baur und A. Schütze. „DAV^{3}E – a MATLAB toolbox for multivariate sensor data evaluation“. *Journal of Sensors and Sensor Systems* 7.2 (2018), S. 489–506. DOI: 10.5194/jsss-7-489-2018.

[86] Günther Görz, Ute Schmid und Tanya Braun, Hrsg. *Handbuch der Künstlichen Intelligenz*. Berlin, Germany: De Gruyter Oldenbourg, 2021. ISBN: 978-3-11-065994-8.

[87] Jörg Frochte. *Maschinelles Lernen: Grundlagen und Algorithmen in Python*. Munich, Germany: Carl Hanser Verlag GmbH & Co. KG, 2019. ISBN: 978-3-446-45291-6.

[88] Kristina P. Sinaga und Miin-Shen Yang. „Unsupervised K-Means Clustering Algorithm“. *IEEE Access* 8 (2020), S. 80716–80727. DOI: 10.1109/ACCESS.2020.2988796.

[89] Sandhya Samarasinghe. *Neural Networks for Applied Sciences and Engineering: From Fundamentals to Complex Pattern Recognition*. New York, NY, USA: Auerbach Publications, 2006. ISBN: 978-0-4291-1578-3.

[90] Hossein Hassani und Emmanuel Sirmal Silva. „The Role of ChatGPT in Data Science: How AI-Assisted Conversational Interfaces Are Revolutionizing the Field“. *Big Data and Cognitive Computing* 7.62 (2023). DOI: 10.3390/bdcc7020062.

[91] E. R. Davies und M. Turk. *Advanced Methods and Deep Learning in Computer Vision*. Computer Vision and Pattern Recognition. Cambridge, MA, USA: Academic Press, 2021. ISBN: 978-0-128-22149-5.

[92] Payman Goodarzi, Andreas Schütze und Tizian Schneider. *Comparing AutoML and Deep Learning Methods for Condition Monitoring using Realistic Validation Scenarios*. 2023. DOI: 10.48550/arXiv.2308.14632.

[93] Huan Liu und Hiroshi Motoda, Hrsg. *Feature Extraction, Construction and Selection: A Data Mining Perspective*. Bd. 453. The Springer International Series in Engineering and Computer Science. New York, NY, USA: Springer-Verlag, 1998. ISBN: 978-1-4613-7720-1.

[94] Kamran Kowsari, Kiana Jafari Meimandi, Mojtaba Heidarysafa, Sanjana Mendu, Laura Barnes und Donald Brown. „Text classification algorithms: A Survey“. *Information* 10.4 (2019). DOI: 10.3390/info10040150.

[95] Geoffrey J. McLachlan. „Mahalanobis distance“. *Resonance* 4.6 (1999), 20—26. DOI: 10.1007/BF02834632.

[96] Marco A.F. Pimentel, David A. Clifton, Lei Clifton und Lionel Tarassenko. „A review of novelty detection“. *Signal Processing* 99 (2014), S. 215–249. DOI: 10.1016/j.sigpro.2013.12.026.

[97] Tizian Schneider. „Methoden der automatisierten Merkmalextraktion und -selektion von Sensorsignalen“. Masterarbeit. Saarbrücken, Deutschland: Lehrstuhl für Messtechnik, Universität des Saarlandes, 2015.

[98] Schaeffler Monitoring Services GmbH. *Condition Monitoring Praxis: Handbuch zur Schwingungs-Zustandsüberwachung von Maschinen und Anlagen*. 1. Auflage. Mainz, Deutschland: Vereinigte Fachverlage, 2019. ISBN: 978-3-7830-0419-9.

[99] Michel Verleysen und Damien François. „The Curse of Dimensionality in Data Mining and Time Series Prediction“. *Computational Intelligence and Bioinspired Systems: 8th International Work-Conference on Artificial Neural Networks (IWANN 2005), 8-10 June 2005*. Barcelona, Spain: Springer-Verlag, 2005, S. 758–770. ISBN: 978-3-540-32106-4.

[100] Naomi Altman und Martin Krzywinski. „The curse(s) of dimensionality“. *Nature Methods* 15.6 (2018), S. 399–400. DOI: 10.1038/s41592-018-0019-x.

[101] Markos Markou und Sameer Singh. „Novelty detection: a review—part 1: statistical approaches“. *Signal Processing* 83.12 (2003), S. 2481–2497. DOI: 10.1016/j.sigpro.2003.07.018.

[102] Tizian Schneider, Steffen Klein und Andreas Schütze. „Machine learning in industrial measurement technology for detection of known and unknown faults of equipment and sensors“. *tm - Technisches Messen* 86.11 (2019), 706—718. DOI: 10.1515/teme-2019-0086.

[103] Farhad Maleki, Nikesh Muthukrishnan, Katie Ovens, Caroline Reinhold und Reza Forghani. „Machine Learning Algorithm Validation: From Essentials to Advanced Applications and Implications for Regulatory Certification and Deployment“. *Neuroimaging Clinics of North America* 30.4 (2020), S. 433–445. DOI: 10.1016/j.nic.2020.08.004.

[104] Haider Kalaf Jabbar und Rafiqul Zaman Khan. „Methods to Avoid Over-Fitting and Under-Fitting in Supervised Machine Learning“. *Proceedings of the International Conference on Computer Science, Communication and Instrumentation Devices*. Research Publishing Services, 2014. DOI: 10.3850/978-981-09-5247-1_017.

[105] Pádraig Cunningham und Sarah Jane Delany. „Underestimation Bias and Underfitting in Machine Learning“. *Trustworthy AI - Integrating Learning, Optimization and Reasoning*. Hrsg. von Fredrik Heintz, Michela Milano und Barry O'Sullivan. Cham: Springer International Publishing, 2021, S. 20–31. ISBN: 978-3-030-73959-1.

[106] Yuxuan Wang und Ross D. King. „Extrapolation is not the same as interpolation“. *Machine Learning* 113 (2024), 8205—8232. DOI: 10.1007/s10994-024-06591-2.

[107] Jacques Wainer und Gavin Cawley. „Nested cross-validation when selecting classifiers is overzealous for most practical applications“. *Expert Systems with Applications* 182 (2021), S. 115222. DOI: https://doi.org/10.1016/j.eswa.2021.115222.

[108] Michael R. Berthold, Nicolas Cebron, Fabian Dill, Thomas R. Gabriel, Tobias Kötter, Thorsten Meinl, Peter Ohl, Kilian Thiel und Bernd Wiswedel. „KNIME - the Konstanz Information Miner: Version 2.0 and Beyond“. *SIGKDD Explor. Newsl.* 11.1 (2009), 26—31. DOI: 10.1145/1656274.1656280.

[109] Santosh Chandwani. *Creating Machine Learning models in Power BI*. 2019. URL: https://powerbi.microsoft.com/de-de/blog/creating-machine-learning-models-in-power-bi/ (besucht am 21. 10. 2024).

[110] The pandas development team. *pandas-dev/pandas: Pandas*. Version v2.0.3. 2023. DOI: 10.5281/zenodo.8092754.

[111] Tanja Dorst, Yannick Robin, Tizian Schneider und Andreas Schütze. „Automated ML Toolbox for Cyclic Sensor Data“. *MSMM 2021 - Mathematical and Statistical Methods for Metrology*. 2021. URL: http://www.msmm2021.

polito.it/content/download/245/1127/file/MSMM2021_Booklet_c.pdf (besucht am 21. 10. 2024).

[112] Robert Thomas Olszewski. „Generalized feature extraction for structural pattern recognition in time-series data“. Dissertation. Carnegie Mellon University, 2001. ISBN: 978-0-493-53871-6.

[113] Svante Wold, Kim Esbensen und Paul Geladi. „Principal component analysis“. *Chemometrics and Intelligent Laboratory Systems* 2.1-3 (1987), S. 37–52. DOI: 10.1016/0169-7439(87)80084-9.

[114] Fabian Mörchen. *Time series feature extraction for data mining using DWT and DFT*. Techn. Ber. Department of Mathematics und Computer Science, 2003.

[115] Ingrid Daubechies. *Ten Lectures on Wavelets*. Philadelphia, PA, USA: Society for Industrial und Applied Mathematics, 1992. ISBN: 978-0-89871-274-2.

[116] Athanasios Papoulis und S. Unnikrishna Pillai. *Probability, Random Variables, and Stochastic Processes*. 4th. Boston, MA, USA: McGraw-Hill, 2001. ISBN: 978-0-071-12256-6.

[117] Isabelle Guyon und André Elisseeff. „An introduction to variable and feature selection“. *Journal of Machine Learning Research* 3 (2003), 1157–1182.

[118] Alain Rakotomamonjy. „Variable Selection Using SVM-based Criteria“. *Journal of Machine Learning Research* 3.7 (2003), 1357—1370. DOI: 10.1162/153244303322753706.

[119] Marko Robnik-Šikonja und Igor Kononenko. „Theoretical and Empirical Analysis of ReliefF and RReliefF“. *Machine Learning* 53.1 (2003), S. 23–69. DOI: 10.1023/A:1025667309714.

[120] Igor Kononenko und Se June Hong. „Attribute selection for modelling“. *Future Generation Computer Systems* 13.2 (1997), S. 181–195. DOI: 10.1016/S0167-739X(97)81974-7.

[121] Jacob Benesty, Jingdong Chen, Yiteng Huang und Israel Cohen. „Pearson correlation coefficient“. *Noise Reduction in Speech Processing*. Springer Verlag, 2009. DOI: 10.1007/978-3-642-00296-0_5.

[122] Christopher Schnur, Tanja Dorst, Kapil Sajjan Deshmukh, Sarah Zimmer, Philipp Litzenburger, Tizian Schneider, Lennard Margies, Rainer Müller und Andreas Schütze. „PIA - A Concept for a Personal Information Assistant for Data Analysis and Machine Learning of Time-Continuous Data in Industrial Applications“. *ing.grid* 1.2 (2023). DOI: 10.48694/inggrid.3827.

[123] Christopher Schnur, Payman Goodarzi, Yevgeniya Lugovtsova, Jannis Bulling, Jens Prager, Kilian Tschöke, Jochen Moll, Andreas Schütze und Tizian Schneider. „Towards Interpretable Machine Learning for Automated Damage Detection Based on Ultrasonic Guided Waves“. *Sensors* 22.1 (2022). DOI: 10.3390/s22010406.

[124] Payman Goodarzi, Steffen Klein, Andreas Schütze und Tizian Schneider. „Comparing Different Feature Extraction Methods in Condition Monitoring Applications“. *Proceedings of the 2023 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), 22-25 May 2023*. Kuala Lumpur, Malaysia: IEEE, 2023, S. 1–6. DOI: 10.1109/I2MTC53148.2023.10176106.

[125] *LMT-ML-Toolbox*. 2021. URL: https://github.com/ZeMA-gGmbH/LMT-ML-Toolbox (besucht am 21. 10. 2024).

[126] Tanja Dorst, Tizian Schneider, Sascha Eichstädt und Andreas Schütze. „Uncertainty-aware automated machine learning toolbox“. *tm - Technisches Messen* 90.3 (2023), S. 141–153. DOI: 10.1515/teme-2022-0042.

[127] Anne Blum, Yannick Wilhelm, Steffen Klein, Christopher Schnur, Peter Reimann, Rainer Müller und Andreas Schütze. „Ganzheitlicher Ablaufplan für wissensgetriebene Projekte des maschinellen Lernens in der Produktion“. *tm - Technisches Messen* 89.5 (2022), S. 363–383. DOI: 10.1515/teme-2022-0027.

[128] Christopher Schnur, Steffen Klein und Anne Blum. *Checkliste – Mess- und Datenplanung für das maschinelle Lernen in der Montage*. Version 7. 2022. DOI: 10.5281/zenodo.6943476.

[129] Mittelstand-Digital Zentrum Saarbrücken, Christopher Schnur, Daniela Schmidt, Daniel Becker und Anne Blum. *KI-Projekte erfolgreich umsetzen: Eine Checkliste für den Mittelstand*. 2024. DOI: 10.5281/zenodo.10069539.

[130] Brigette M. Hales und Peter J. Pronovost. „The checklist—a tool for error management and performance improvement“. *Journal of Critical Care* 21.3 (2006), S. 231–235. DOI: 10.1016/j.jcrc.2006.06.002.

[131] Christopher Schnur, Steffen Klein, Anne Blum, Andreas Schütze und Tizian Schneider. „Steigerung der Datenqualität in der Montage“. *wt Werkstattstechnik online* 11-12 (2022), S. 783–787. DOI: 10.37544/1436-4980-2022-11-12-57.

[132] Esther Cameron und Mike Green. *Making Sense of Change Management: A Complete Guide to the Models, Tools and Techniques of Organizational Change*. 6. Aufl. London, United Kingdom: Kogan Page, 2024. ISBN: 978-1-398-61288-4.

[133] Ross K. Kennedy. *Understanding, Measuring, and Improving Overall Equipment Effectiveness: How to Use OEE to Drive Significant Process Improvement*. New York, NY, USA: Productivity Press, 2017. ISBN: 978-1-138-30556-2.

[134] Steven Nelson. *Pro Data Backup and Recovery*. Berkeley, CA, USA: Apress, 2011. ISBN: 978-1-4302-2662-8.

[135] T. W. Anderson und D. A. Darling. „Asymptotic Theory of Certain Goodness of Fit Criteria Based on Stochastic Processes“. *The Annals of Mathematical Statistics* 23.2 (1952). DOI: 10.1214/aoms/1177729437.

[136] Adrian Stetco, Fateme Dinmohammadi, Xingyu Zhao, Valentin Robu, David Flynn, Mike Barnes, John Keane und Goran Nenadic. „Machine learning methods for wind turbine condition monitoring: A review“. *Renewable Energy* 133 (2019), S. 620–635. DOI: 10.1016/j.renene.2018.10.047.

[137] Salvador García, Sergio Ramírez-Gallego, Julián Luengo, José Manuel Benítez und Francisco Herrera. „Big data preprocessing: methods and prospects“. *Big Data Analytics* 1.1 (2016). DOI: 10.1186/s41044-016-0014-0.

[138] Steffen Klein, Yannick Wilhelm, Andreas Schütze und Tizian Schneider. „Combination of generic novelty detection and supervised classification pipelines for industrial condition monitoring“. *tm - Technisches Messen* 91.9 (2024), S. 454–465. DOI: 10.1515/teme-2024-0016.

[139] Sebastian Raschka. *Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning*. arXiv. 2020. DOI: 10.48550/arXiv.1811.12808.

[140] Eamonn Keogh und Shruti Kasetty. „On the Need for Time Series Data Mining Benchmarks: A Survey and Empirical Demonstration“. *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 23–26 July 2002*. Edmonton, Alberta, Canada: ACM, 2002, S. 102–111. DOI: 10.1145/775047.775062.

[141] Tizian Schneider. „On Automatic Machine Learning for Industrial Condition Monitoring“. Dissertation. Universität des Saarlandes, Naturwissenschaftlich-Technische Fakultät, 2024. DOI: 10.22028/D291-42447.

[142] Oracle Corporation. *VirtualBox*. URL: https://www.virtualbox.org/ (besucht am 21. 10. 2024).

[143] Jay LaCroix. *Mastering Ubuntu Server: Master the Art of Deploying, Configuring, Managing, and Troubleshooting Ubuntu Server 18.04*. Second Edition. Birmingham, UK: Packt Publishing Ltd, 2018. ISBN: 978-1-800-56464-0.

[144] Google LLC. *Angular 13.3.4 Release*. URL: https://www.gitclear.com/open_repos/angular/angular/release/13.3.4 (besucht am 21. 10. 2024).

[145] Mengnan Liu, Shuiliang Fang, Huiyue Dong und Cunzhi Xu. „Review of digital twin about concepts, technologies, and industrial applications“. *Journal of manufacturing systems* 58 (2021), S. 346–361. DOI: 10.1016/j.jmsy.2020.06.017.

[146] Katrin Weller. „B 6 Ontologien“. *Grundlagen der praktischen Information und Dokumentation*. Hrsg. von Rainer Kuhlen, Wolfgang Semar und Dietmar Strauch. Berlin, Boston: De Gruyter Saur, 2013, S. 207–218. DOI: 10.1515/9783110258264.207.

[147] MathWorks. *MATLAB Online*. URL: https://de.mathworks.com/products/matlab-online.html (besucht am 21. 10. 2024).

[148] ZeMA gGmbH. *PIA Repository*. URL: https://github.com/ZeMA-gGmbH/-PIA (besucht am 21. 10. 2024).

[149] Michael Bartl, Sebastian Bertram, Peter Burggräf, Matthias Dannapfel, Arne Fischer, Jörg Grams, Jan Knau, Kai Kreisköther und Johannes Wagner. „Agile Low-Cost Montage“. Mai 2017, S. 231–259. ISBN: 978-3-86359-512-8.

[150] Marco Ehrlich, Lukasz Wisniewski und Jürgen Jasperneite. „Usage of retrofitting for migration of industrial production lines to industry 4.0“. *Jahreskolloquium Kommunikation in der Automation (KommA)* (2015). URL: https://www.researchgate.net/profile/Marco-Ehrlich/publication/280529750_Usage_of_Retrofitting_for_Migration_of_Industrial_Production_Lines_to_Industry_40/links/56655fa008ae418a786ec862/Usage-of-Retrofitting-for-Migration-of-Industrial-Production-Lines-to-Industry-40.pdf (besucht am 21. 10. 2024).

[151] National Instruments. *cRIO-9040 Specifications*. URL: https://www.ni.com/docs/de-DE/bundle/crio-9040-specs/page/specs.html (besucht am 21. 10. 2024).

[152] National Instruments. *NI 9232 Specifications*. URL: https://www.ni.com/docs/en-US/bundle/ni-9232-specs/page/specs.html (besucht am 21. 10. 2024).

[153] National Instruments. *NI 9215 Specifications*. URL: https://www.ni.com/docs/en-US/bundle/ni-9215-specs/page/specs.html (besucht am 21. 10. 2024).

[154] National Instruments. *The NI TDMS File Format - What is a TDMS File?* 2024. URL: https://www.ni.com/en/support/documentation/supplemental/06/the-ni-tdms-file-format.html?srsltid=AfmBOor-tfTd6fb5LQe6AT-UQCur1LECLbohPG4fYDmaQOX3lBjHtOvR (besucht am 21. 10. 2024).

[155] Bernd Künne. *Einführung in die Maschinenelemente*. Wiesbaden, Deutschland: Vieweg+Teubner Verlag, 2001. ISBN: 978-3-663-05920-2. DOI: 10.1007/978-3-663-05920-2.

[156] Schaeffler Technologies AG & Co. KG. *Wälzlagerschäden - Schadenserkennung und Begutachtung gelaufener Wälzlager*. 2013. URL: https://www.schaeffler.de/de/news_medien/mediathek/downloadcenter-detail-page.jsp?id=114074 (besucht am 21. 10. 2024).

[157] Case School of Engineering. *Case western reserve university bearing data set*. URL: https://engineering.case.edu/bearingdatacenter (besucht am 21. 10. 2024).

[158] Christian Lessmeier, James Kuria Kimotho, Detmar Zimmer und Walter Sextro. „Condition Monitoring of Bearing Damage in Electromechanical Drive Systems by Using Motor Current Signals of Electric Motors: A Benchmark Data Set for Data-Driven Classification“. *Proceedings of the European Conference of the PHM Society 2016, 5-8 July 2016*. Bd. 3. 1. Bilbao, Spain: PHM Society, 2016, S. 5–10. DOI: 10.36001/phme.2016.v3i1.1577.

[159] Vinayak Tyagi. *NASA Bearing Dataset*. 2023. URL: https://www.kaggle.com/datasets/vinayak123tyagi/bearing-dataset (besucht am 21. 10. 2024).

[160] Christopher Schnur, Yannick Robin, Payman Goodarzi, Tanja Dorst, Andreas Schütze und Tizian Schneider. „Development of a Bearing Test-Bed for Acquiring Data for Robust and Transferable Machine Learning“. *Proceedings of the IEEE International Instrumentation and Measurement Technology Conference (I2MTC), 22-25 May 2023*. Kuala Lumpur, Malaysia: IEEE, 2023, S. 1–6. DOI: 10.1109/I2MTC53148.2023.10176017.

[161] DIN Media GmbH. *ISO 20816-3:2022-10 Mechanical vibration - Measurement and evaluation of machine vibration - Part 3: Industrial machinery with a power rating above 15 kW and operating speeds between 120 r/min and 30 000 r/min*. Standard. International Organization for Standardization, 2022.

[162] International Organization for Standardization. *BS ISO 7919-3+A1:2009-03-31 - Mechanical vibration - Evaluation of machine vibration by measurements on rotating shafts - Coupled industrial machines*. Techn. Ber. 2009.

[163] Schaeffler Technologies. *NU206-E-XL-TVP2 Product Information*. URL: https://medias.schaeffler.de/de/produkt/rotary/waelz--und-gleitlager/rollenlager/zylinderrollenlager/nu206-e-xl-tvp2/p/368765#Product%20Information (besucht am 21. 10. 2024).

[164] Nicolas Carpi, Alexander Minges und Matthieu Piel. *eLabFTW: An open source laboratory notebook for research labs*. Version 1.4.0. 2017. DOI: 10.21105/joss.00146.

[165] Christophe Bahim, Carlos Casorrän-Amilburu, Makx Dekkers, Edit Herczog, Nicolas Loozen, Konstantinos Repanas, Keith Russell und Shelley Stall. „The FAIR Data Maturity Model: An Approach to Harmonise FAIR Assessments“. *Data Science Journal* 19.1 (2020). DOI: 10.5334/dsj-2020-041.

[166] Schaeffler Technologies AG & Co. KG. *Einreihige vollrollige Zylinderrollenlager - Mindestbelastung*. 2024. URL: https://medias.schaeffler.de/de/single-row-full-complement-cylindrical-roller-bearings#mindestbelastung (besucht am 21. 10. 2024).

[167] S. Lacey. *An Overview of Bearing Vibration Analysis*. 2008. URL: https://www.schaeffler.com/remotemedien/media/_shared_media/08_media_library/01_publications/schaeffler_2/technicalpaper_1/download_1/vibration_analysis_en_en.pdf (besucht am 21. 10. 2024).

[168] Eric Bechhoefer. „A quick introduction to bearing envelope analysis“. *Green Power Monit. Syst* (2016). URL: https://mfpt.org/wp-content/uploads/2018/03/MFPT-Bearing-Envelope-Analysis.pdf (besucht am 21. 10. 2024).

[169] Dhiraj Neupane und Jongwon Seok. „Bearing Fault Detection and Diagnosis Using Case Western Reserve University Dataset With Deep Learning Approaches: A Review“. *IEEE Access* 8 (2020), S. 93155–93178. DOI: 10.1109/ACCESS.2020.2990528.

[170] Kaiming He, Xiangyu Zhang, Shaoqing Ren und Jian Sun. *Deep Residual Learning for Image Recognition*. 2015. DOI: 10.48550/arXiv.1512.03385.

[171] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior und Koray Kavukcuoglu. *WaveNet: A Generative Model for Raw Audio*. 2016. DOI: 10.48550/arXiv.1609.03499.

[172] Simon Haykin. *Neural Networks: A Comprehensive Foundation*. Upper Saddle River, NJ, USA: Prentice Hall, 1998. ISBN: 978-0-132-73350-2.

[173] D. Ho und R.B. Randall. „Optimisation of Bearing Diagnostic Techniques Using Simulatend and Actual Bearing Fault Signals“. *Mechanical Systems and Signal Processing* 14.5 (2000), S. 763–788. DOI: 10.1006/mssp.2000.1304.

[174] Manqi Zhao und Venkatesh Saligrama. „Anomaly Detection with Score Functions Based on Nearest Neighbor Graphs“. *Advances in Neural Information Processing Systems 22: Proceedings of the 2009 Conference, 7-10 December 2009*. Vancouver, Canada: Curran Associates, Inc., 2009, S. 2250–2258. ISBN: 978-1-61567-911-9.

[175] Robert J. Marks II. *Handbook of Fourier Analysis and its Applications*. Oxford, United Kingdom: Oxford University Press, 2009. ISBN: 978-0-19-533592-7.

[176] J. Schauer, H. El Moutaouakil, P. Goodarzi, C. Schnur und A. Schütze. „Interpretable Machine Learning Algorithm for Bit Damage Detection in a Screwing Process based on Accelerometers“. *Proceedings of the 17th Dresdner Sensor-Symposium, Presentation, 25-27 November 2024*. Dresden, Germany, 2024.

[177] Dominik Kuhn, Rainer Müller, Leenhard Hörauf, Martin Karkowski und Manuel Holländer. „Wandlungsfähige Montagesysteme für die nachhaltige Produktion von morgen“. *wt Werkstattstechnik online* 110 (2020), S. 579–584. DOI: 10.37544/1436-4980-2020-09-9.

[178] Karl-Heinz Kloos und Wolfgang Thomala. *Schraubenverbindungen: Grundlagen, Berechnung, Eigenschaften, Handhabung*. 5. Auflage. Berlin, Deutschland: Springer-Verlag, 2007. ISBN: 978-3-540-21282-9.

[179] Dytran Instruments. *Triaxial Accelerometer 3233A Specifications*. URL: https://www.hbkworld.com/web/dytran/global/en/products/accelerometers/voltage--iepe-ccld-/triaxial/3233a#Specifications-909f70388e (besucht am 21. 10. 2024).

[180] EMSYST spol. s r.o. *EMS20 Universal Membrane Force Sensor Specifications*. URL: https://www.emsyst.sk/en/products/force-sensors-load-cells/standard/EMS20 (besucht am 21. 10. 2024).

[181] Gyungmin Toh, Jaesoo Gwon und Junhong Park. „Determination of Clamping Force Using Bolt Vibration Responses during the Tightening Process". *Applied Sciences* 9.24 (2019). DOI: 10.3390/app9245379.

[182] Nicolaj Baramsky, Arthur Seibel und Josef Schlattmann. „Friction-Induced Vibrations during Tightening of Bolted Joints—Analytical and Experimental Results". *Vibration* 1.2 (2018), S. 312–337. DOI: 10.3390/vibration1020021.

[183] OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal und Ahmad Lama. *GPT-4 Technical Report*. ArXiv. 2023. DOI: 10.48550/arXiv.2303.08774.

[184] Partha Pratim Ray. „ChatGPT: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope". *Internet of Things and Cyber-Physical Systems* 3 (2023), S. 121–154. DOI: 10.1016/j.iotcps.2023.04.003.

[185] A Shaji George und AS Hovan George. „A review of ChatGPT AI's impact on several business sectors". *Partners Universal International Innovation Journal* 1.1 (2023). DOI: https://doi.org/10.5281/zenodo.7644359.

[186] Paresh Girdhar und Cornelius Scheffer. *Practical Machinery Vibration Analysis and Predictive Maintenance*. Amsterdam, Netherlands: Elsevier, 2004. ISBN: 978-0-7506-6275-8.

[187] MathWorks Deutschland. *Fast Fourier transform - MATLAB fft - MathWorks Deutschland*. URL: https://de.mathworks.com/help/matlab/ref/fft.html#buuutyt-6 (besucht am 21. 10. 2024).

[188] MathWorks Deutschland. *Phase angle - MATLAB angle*. URL: https://de.mathworks.com/help/matlab/ref/angle.html (besucht am 21. 10. 2024).

[189] MathWorks Deutschland. *Average or mean value of array - MATLAB mean*. URL: `https://de.mathworks.com/help/matlab/ref/mean.html` (besucht am 21. 10. 2024).

[190] MathWorks Deutschland. *Standard deviation - MATLAB std*. URL: `https://de.mathworks.com/help/matlab/ref/std.html` (besucht am 21. 10. 2024).

[191] MathWorks Deutschland. *Skewness - MATLAB skewness*. URL: `https://de.mathworks.com/help/stats/skewness.html` (besucht am 21. 10. 2024).

[192] MathWorks Deutschland. *Kurtosis - MATLAB kurtosis*. URL: `https://de.mathworks.com/help/stats/kurtosis.html` (besucht am 21. 10. 2024).

[193] William H. Press, Saul A. Teukolsky, William T. Vetterling und Brian P. Flannery. *Numerical Recipes: The Art of Scientific Computing*. 3rd. Cambridge, UK: Cambridge University Press, 2007. ISBN: 978-0-521-88068-8.

[194] Bernhard Schölkopf und Alexander J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Adaptive Computation and Machine Learning Series. Cambridge, MA, USA: MIT Press, 2018. ISBN: 978-0-262-53657-8.

[195] Trevor Hastie, Robert Tibshirani und Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2. Aufl. New York, NY, USA: Springer, 2009. ISBN: 978-0-387-84858-7.

[196] Kenji Kira und Larry A. Rendell. „A Practical Approach to Feature Selection“. Morgan Kaufmann, 1992. DOI: `10.1016/B978-1-55860-247-2.50037-1`.

[197] MathWorks Deutschland. *Find k-nearest neighbors using input data - MATLAB knnsearch*. URL: `https://de.mathworks.com/help/stats/knnsearch.html` (besucht am 21. 10. 2024).

[198] MathWorks Deutschland. *Distance Transform of a Binary Image*. URL: `https://de.mathworks.com/help/images/distance-transform-of-a-binary-image.html` (besucht am 21. 10. 2024).

Eigene Veröffentlichungen

Journalbeitrag A **C. Schnur**, P. Goodarzi, Y. Lugovtsova, J. Bulling, J. Prager, K. Tschöke, J. Moll, A. Schütze und T. Schneider: Towards Interpretable Machine Learning for Automated Damage Detection Based on Ultrasonic Guided Waves, in *Sensors 2022, 22(1) special issue Smart Sensors for Damage Detection*, 406, (2022), DOI: 10.3390/s22010406

Journalbeitrag B A. Blum, Y. Wilhelm, S. Klein, **C. Schnur**, P. Reimann, R. Müller und A. Schütze: Ganzheitlicher Ablaufplan für wissensgetriebene Projekte des maschinellen Lernens in der Produktion, *tm - Technisches Messen*, 89(5), 363-383, (2022), DOI: 10.1515/teme-2022-0027

Journalbeitrag C **C. Schnur**, S. Klein, A. Schütze, T. Schneider und A. Blum: Steigerung der Datenqualität in der Montage, *WT Werkstattstechnik*, 112 (2022), NR. 11-12, S. 783 - 787, DOI: 10.37544/1436-4980-2022-11-12-57

Journalbeitrag D **C. Schnur**, T. Dorst, K. Deshmukh, S. Zimmer, P. Litzenburger, T. Schneider, L. Margies, R. Müller und A. Schütze: PIA - A Concept for a Personal Information Assistant for Data Analysis and Machine Learning in industrial application, *ing.grid*,1(2), (2023) DOI: 10.48694/inggrid.3827

Posterbeitrag A **C. Schnur**, S. Klein, A. Blum, T. Schneider, R. Müller und A. Schütze: Mess- und Datenplanung für Modelle des maschinellen Lernens an Bestandsanlagen, *16. Dresdner Sensor-Symposium*, Dresden, 5-7 Dezember, (2022) DOI: 10.5162/16dss2022/P47

Konferenzbeitrag A **C. Schnur**, J. Moll, Y. Lugovstovo und A. Schütze: Explainable Machine Learning for Damage Detection in Carbon Fiber Composite Plates Under Varying Temperature Conditions, *QNDE 2021 – 48th Annual Review of Progress in Quantitative Nondestructive Evaluation* (online), 28-30 Juli, (2021), DOI: 10.1115/QNDE2021-75215

Konferenzbeitrag B **C. Schnur**, T. Dorst, K. Deshmukh, S. Zimmer, P. Litzenburger, L. Margis, T. Schneider, R. Müller und A. Schütze: PIA – A concept for a Personal Information Assistant for Data Analysis, *NFDI4Ing Conference 2022* (online), 26-27 Oktober, (2022), DOI: 10.5281/zenodo.7362038

Konferenzbeitrag C S. Wöckel, **C. Schnur**, A. Schütze und U. Hampel: Potenziale der energieautarken Prozessüberwachung mittels Energy Harvesting und maschinellem Lernen, *Jahrestreffen der ProcessNet-Fachgemeinschaften "Prozess-, Apparate- und Anlagentechnik"*, Frankfurt, 21-22 November, (2022)

Konferenzbeitrag D **C. Schnur**, Y. Robin, P. Goodarzi, T. Dorst, A. Schütze, T. Schneider: Development of a bearing test-bed for acquiring data for robust and transferable machine learning, *IEEE I2MTC 2023, International Instrumentation and Measurement Technology Conference*, Kuala Lumpur, 22-25 Mai, (2023), DOI: 10.1109/I2MTC53148.2023.10176017

Konferenzbeitrag E J. Schauer, H. El Moutaouakil, P. Goodarzi, **C. Schnur** und A. Schütze: Interpretable Machine Learning Algorithm for Bit Damage Detection in a Screwing Process based on Accelerometers, *17. Dresdner Sensor-Symposium*, Dresden, 25-27 November, (2024), DOI: 10.5162/17dss2024/7.1

Software A **C. Schnur**, T. Dorst, K. Deshmukh, S. Zimmer, P. Litzenburger, T. Schneider, L. Margies, R. Müller und A. Schütze: PIA - A Concept for a Personal Information Assistant for Data Analysis and Machine Learning in industrial application, *Github*, (2023), `https://github.com/ZeMA-gGmbH/-PIA`

Sonstiges A **C. Schnur**, S. Klein und A. Blum: Checkliste – Mess- und Datenplanung für das maschinelle Lernen in der Montage, *Zenodo*, DOI: 10.5281/zenodo.6943476, (2022)

Sonstiges B C. Bur, **C. Schnur**, A. Schütze: Mess- und Datenplanung als Grundlage für eine valide und robuste Modellbildung mittels Maschinellem Lernen, *Jahresmagazin Mess- und Sensortechnik 2022/2023*, Institut für wissenschaftliche Veröffentlichungen (IWV), S. 100-105, (2022), ISSN: 1618-8357

Sonstiges C **C. Schnur**, S. Klein und A. Blum: Checklist – Measurement and data planning for machine learning in assembly, *Zenodo*, DOI: 10.5281/zenodo.7556875, (2022)

Sonstiges D **C. Schnur**, D. Schmidt, D. Becker und A. Blum: KI-Projekte erfolgreich umsetzen: Eine Checkliste für den Mittelstand, *Mittelstand-Digital Zentrum Saarbrücken*, `https://zenodo.org/records/100695`, (2024)

A Anhang

A.1 Grundlagen

A.1.1 Herausforderungen in der Produktion

Tabelle A.1: Herausforderungen für die Produktion nach [11].

Nr	Herausforderung	Beschreibung
P1	Heterogene Produkte, Prozesse und Betriebsmittel	Je nach Fokus und Automatisierungsgrad der Anwendung werden unterschiedlichste Produkte mit verschiedensten Betriebsmitteln gefertigt. Ausschlaggebend ist hier u.a. die Losgröße (Anzahl produzierter Teile, von Stückzahl 1 bis zur Serienproduktion) und die Beständigkeit der Betriebsmittel.
P2	Vielfältige Fehlertypen	Speziell in der Industrie gehen mit einer hohe Komplexität von Produkten, Prozessen oder Betriebsmitteln eine hohe Anzahl verschiedener Fehlertypen einher, welche das Endergebnis (z.B. montiertes Produkt) beeinflussen können.
P3	Ungleiche Datenverteilung	In der Regel ist die Datenverteilung in der Industrie aus Gründen der Effizienz unausgewogen. So wird in einer Produktion i.d.R. versucht die Anzahl des Ausschusses so gering wie möglich zu halten. Dies ist in der Datenanalyse und bei deren Interpretation unbedingt zu berücksichtigen.
P4	Nicht-lineare Prozesse und Schleifen	Prozesse folgen oftmals nicht einer linearen Reihenfolge oder müssen in Schleifen durchgeführt werden. Beispiele hierfür sind Nacharbeits- bzw. Korrekturschleifen oder auch Referenzfahrten.
P5	Konzept- bzw. Datensatzverschiebungen	Im Laufe der Zeit kann sich die Verteilung der Daten z.B. durch Verschleiß oder die Anpassungen von Parametern ändern bzw. verschieben, was die Gültigkeit der Datenanalyse bzw. des maschinellen Lernmodells beeinflussen kann.

Tabelle A.2: Herausforderungen für die Daten nach [11].

Nr	**Herausforderung**	**Beschreibung**
D1	Komplexe Datenfusion aus verschiedenen Datenquellen	Die Kombination von Daten mehrerer Datenquellen (z.B. Sensoren) kann sich komplex gestalten.
D2	Fehlende Eindeutigkeit und Zuordenbarkeit	Die Eindeutigkeit und Zuordenbarkeit von Daten zu Produkten, Prozessen oder Betriebsmitteln stellt eine Grundlage zur Fusion von Daten (Herausforderung D1) dar.
D3	Fehlende Synchronisation	Sind Systeme, z.B. aufgrund von Abweichungen der Uhrzeit und damit auch des Zeitstempels, nicht synchron zueinander und Herausforderung D2 ist ebenfalls nicht erfüllt, so können Daten nicht fusioniert werden.
D4	Doppelte und ungültige Messwerte	Doppelte und ungültige Messwerte können durch fehlerhafte Messung, Referenzfahrten oder Kalibriermessungen auftreten und sind unbedingt als solche zu kennzeichnen.
D5	Fehlende Metadaten	Metadaten sind von hoher Bedeutung bei der Datenanalyse. Fehlende Metadaten können sich insbesondere negativ auf die Interpretierbarkeit der Ergebnisse (Herausforderung A2) auswirken.
D6	Unzureichende Datenaufnahme	Um verwertbare Ergebnisse bei der Datenanalyse zu erzielen, müssen die gesuchten Informationen sich in den Daten wiederfinden.
D7	Nicht dokumentierte Systemänderungen	Jede Änderung am System, speziell an Prozessen und Betriebsmitteln, muss dokumentiert werden, um z.B. mögliche Verschiebungen in den Daten (Herausforderung P5) ihrer Ursache zuordnen zu können.

Tabelle A.3: Herausforderungen für die Analyse nach [11].

Nr	Herausforderung	Beschreibung
A1	Vertrauen in Datenanalysen und maschinelle Lernmodelle	Mit steigenden Sicherheitsanforderungen an Produkte, Prozesse und Betriebsmittel, steigt auch der Anspruch an die Zuverlässigkeit der Datenanalyse und die maschinellen Lernmodelle. Wichtig ist hierbei die Interpretierbarkeit bzw. Nachvollziehbarkeit der Ergebnisse.
A2	Rückverfolgbarkeit der Analyseergebnisse zu ihren Ursachen	Analog zu Herausforderung A1, spielt die Rückverfolgbarkeit der Analyseergebnisse zu ihren Ursachen eine übergeordnete Rolle in der Interpretation der Ergebnisse und bei der Erzeugung robuster maschineller Lernmodelle.
A3	Verwaltung und Kontrolle von ML-Modellen	Bei der Verwendung mehrerer Modelle kann deren Verwaltung und die Kontrolle der Gültigkeit der Modelle zu einer Herausforderung werden.
A4	Feature Engineering und die Auswahl von Modellen und ihren Parametern	Zur Erzeugung robuster und interpretierbarer Modelle ist oftmals die gezielte Auswahl von Features (mittels Merkmalsextraktion, -selektion) bzw. Modellen und ihren Parametern notwendig. Grundlage hierfür ist die Einbindung von Wissen über Produkte, Prozesse und Betriebsmittel.

Tabelle A.4: Herausforderungen für die Software nach [11].

Nr	Herausforderung	Beschreibung
S1	Fehlende Datenschnittstellen	Speziell ältere Maschinen oder Anlagen besitzen keine (moderne) Datenschnittstelle bzw. propreritäre Systeme, wodurch diese Datenquellen nicht oder nur schwer zugänglich sind.
S2	Programmierung individueller Lösungen	Auch die Programmierung individueller Lösungen (bspw. Softwareskripte für maschinelle Lernmodelle) kann zur Herausforderung werden, insbesondere wenn diese einen sehr beschränkten Anwendungsbereich haben.

A.1.2 Indikatoren des FAIR-Data-Maturity-Modell

Nachfolgend werden die aus dem Englischen übersetzten Indikatoren des FAIR-Data-Maturity-Modells nach [165] aufgelistet.

Tabelle A.5: Findable-Indikatoren des FAIR-Data-Maturity-Modell, übersetzt aus [165].

FAIR	Indikator	Beschreibung	Relevanz
F1	RDA-F1-01M	Die Metadaten sind durch eine ID eindeutig und dauerhaft identifizierbar.	Essentiell
F1	RDA-F1-01D	Die Daten sind durch eine ID eindeutig und dauerhaft identifizierbar.	Essentiell
F1	RDA-F1-02M	Die Metadaten sind global einzigartig, identifizierbar.	Essentiell
F1	RDA-F1-02D	Die Daten sind global einzigartig identifizierbar.	Essentiell
F2	RDA-F2-01M	Reichhaltige Metadaten sind vorhanden.	Essentiell
F3	RDA-F3-01M	Die Metadaten enthalten die ID der Daten.	Essentiell
F4	RDA-F4-01M	Die Metadaten werden in einer erfassbaren und indizierbaren Form bereitgestellt.	Essentiell

Tabelle A.6: Accessible-Indikatoren des FAIR-Data-Maturity-Modell, übersetzt aus [165].

FAIR	Indikator	Beschreibung	Relevanz
A1	RDA-A1-01M	Die Metadaten enthalten Informationen, welche dem Nutzer den Zugang zu den Daten erlauben.	Wichtig
A1	RDA-A1-02M	Der Nutzer kann auf Metadaten manuell zugreifen.	Essentiell
A1	RDA-A1-02D	Der Nutzer kann auf Daten manuell zugreifen.	Essentiell
A1	RDA-A1-03M	Die Metadaten-ID führt zu einem Metadatensatz.	Essentiell
A1	RDA-A1-03D	Die Daten-ID führt zu einem digitalen Objekt.	Essentiell
A1	RDA-A1-04M	Auf die Metadaten kann durch ein standardisiertes Protokoll zugegriffen werden.	Essentiell
A1	RDA-A1-04D	Auf die Daten kann durch ein standardisiertes Protokoll zugegriffen werden.	Essentiell
A1	RDA-A1-05D	Auf die Daten kann automatisiert zugegriffen werden.	Wichtig
A1.1	RDA-A1.1-01M	Auf die Metadaten kann mittels eines kostenlosen Protokolls zugegriffen werden.	Essentiell
A1.1	RDA-A1.1-01D	Auf die Daten kann mittels eines kostenlosen Protokolls zugegriffen werden.	Wichtig
A1.2	RDA-A1.2-01D	Auf die Daten kann durch Authentifizierung zugegriffen werden.	Nützlich
A2	RDA-A2-01M	Auf die Metadaten kann zugegriffen werden, nachdem die Daten nicht mehr vorhanden sind.	Essentiell

Tabelle A.7: Interoperable-Indikatoren des FAIR-Data-Maturity-Modell, übersetzt aus [165].

FAIR	**Indikator**	**Beschreibung**	**Relevanz**
I1	RDA-I1-01M	Die Metadaten nutzen eine Wissensrepresentation in einem standardisierten Format.	Wichtig
I1	RDA-I1-01D	Die Daten nutzen eine Wissensrepresentation in einem standardisierten Format.	Wichtig
I1	RDA-I1-02M	Die Metadaten nutzen eine maschinenlesbare Form der Wissensrepresentation.	Wichtig
I1	RDA-I1-02D	Die Daten nutzen eine maschinenlesbare Form der Wissensrepresentation.	Wichtig
I2	RDA-I2-01M	Die Metadaten nutzen FAIR-konforme Bezeichnungen.	Wichtig
I2	RDA-I2-01D	Die Daten nutzen FAIR-konforme Bezeichnungen.	Nützlich
I3	RDA-I3-01M	Die Metadaten enthalten Referenzen zu anderen Metadaten.	Wichtig
I3	RDA-I3-01D	Die Daten enthalten Referenzen zu anderen Daten.	Nützlich
I3	RDA-I3-02M	Die Metadaten enthalten Referenzen zu anderen Daten.	Nützlich
I3	RDA-I3-02D	Die Daten enthalten qualifizierte Referenzen auf andere Daten.	Nützlich
I3	RDA-I3-03M	Die Metadaten enthalten qualifizierte Referenzen auf andere Metadaten.	Wichtig
I3	RDA-I3-04M	Die Metadaten enthalten qualifizierte Referenzen auf andere Daten.	Nützlich

Tabelle A.8: Reusable-Indikatoren des FAIR-Data-Maturity-Modell, übersetzt aus [165].

FAIR	Indikator	Beschreibung	Relevanz
R1	RDA-R1-01M	Eine Vielzahl relevanter und genauer Attribute ist vorhanden, um die Wiederverwendbarkeit zu gewährleisten.	Essentiell
R1.1	RDA-R1.1-01M	Die Metadaten enthalten Informationen über die Lizenz der Wiederverwendbarkeit.	Essentiell
R1.1	RDA-R1.1-02M	Die Metadaten verweisen auf eine standardisierte Lizenz der Wiederverwendbarkeit.	Wichtig
R1.1	RDA-R1.1-03M	Die Metadaten verweisen auf eine maschinenlesbare Lizenz der Wiederverwendbarkeit.	Wichtig
R1.2	RDA-R1.2-01M	Die Metadaten enthalten Informationen über die (Daten-)Herkunft, gemäß der community-spezifischen Standards.	Wichtig
R1.2	RDA-R1.2-02M	Die Metadaten enthalten Informationen über die (Daten-)Herkunft in einer community-übergreifenden Sprache.	Nützlich
R1.3	RDA-R1.3-01M	Die Metadaten entsprechen dem Standard der jeweiligen Community.	Essentiell
R1.3	RDA-R1.3-01D	Die Daten entsprechen dem Standard der jeweiligen Community.	Essentiell
R1.3	RDA-R1.3-02M	Die Metadaten sind gemäß einem maschinenlesbaren Standard der jeweiligen Community gestaltet.	Essentiell
R1.3	RDA-R1.3-02D	Die Daten sind gemäß einem maschinenlesbaren Standard der jeweiligen Community gestaltet.	Wichtig

A.1.3 Algorithmen der Toolbox

In den folgenden Abschnitten werden die Algorithmen der ML-Toolbox [8], mit Ausnahme der bereits vorgestellten Algorithmen Principal Component Analysis (PCA) (vgl. Abschnitt 2.6.1.3) und LDA (vgl. Abschnitt 2.6.1.4) näher erläutert.

A.1.3.1 Adaptive Lineare Approximation

Bei der Adaptive Lineare Approximation (ALA) werden die Daten, z.B. ein Messzyklus, in n Segmente unterteilt, wovon jedes Segment linear durch eine Gerade approximiert wird. Pro Segment werden jeweils die Merkmale Mittelwert und Steigung extrahiert. Die Segmentlänge ist dabei adaptiv bzw. variabel und wird durch den minimalen Approximationsfehler bestimmt [112].

Die ALA eignet sich zur Extraktion lokaler Details im Zeitbereich und zur Unterdrückung von Rauschen. Nachteilig ist der vergleichsweise hohe Rechenaufwand

und die Annahme linearer Zusammenhänge sowie die Notwendigkeit strukturell gleicher Messzyklen [8].

In der ML-Toolbox sind $n = 10$ Segmente als Standardwert definiert, wodurch die Daten in 10 Segmente eingeteilt werden und somit 20 Merkmale extrahiert werden.

A.1.3.2 Beste Fourierkoeffizienten

Vibrationen entstehen bei praktisch allen automatisierten Maschinen und Anlagen und stellen daher ein wichtiges Element in der Datenanalyse und speziell der Zustandsüberwachung dar. Insbesondere bei rotierenden Teilen treten Fehlerzustände periodisch auf, wodurch sich die Betrachtung der Signale im Frequenzspektrum anbietet [98, 186].

Hier können durch die Merkmalsextraktionsmethode Beste Fourier Koeffizienten (BFC) Merkmale mit Informationen über lokale Details aus dem Frequenzspektrum mit den zugehörigen Phasen bei vergleichsweise geringer Rechenkomplexität gewonnen werden [8]. In der ML-Toolbox werden die BFC durch die MATLAB-Funktionen `fft()` [187] und `phase()` [188] bestimmt. Dabei wird mittels diskreter Fourier-Transformation die Fouriertransformation $Y(k)$ eines Eingangssignals X der Länge n anhand von

$$Y(k) = \sum_{j=1}^{n} X(j) W_n^{(j-1)(k-1)} \tag{A.1}$$

mit

$$W_n = e^{-2\pi i/n} \tag{A.2}$$

bestimmt [187]. Anschließend extrahiert die ML-Toolbox die 10 % der Frequenzen mit den größten Amplituden und ihrer entsprechenden Phase als Merkmale.

Weiterführende Informationen über die Implementierung und Anwendung der BFC finden sich in [8, 187, 188].

A.1.3.3 Beste Daubechies Wavelets

Mittels der Merkmalsextraktionsmethode Beste Daubechies Wavelets (BDW) können Informationen im Zeit-Frequenz-Bereich, sowohl für lokale Details als auch für die allgemeine Zyklusform extrahiert werden, bei gleichzeitig geringer Rechenkomplexität [8].

In der ML-Toolbox wird eine diskrete Wavelet-Transformation mit dem Daubechies-4-Wavelet [115] als Filterbank durchgeführt und die 10 % der Wavelet-Koeffizienten mit dem höchsten Absolutwert als Merkmale extrahiert.

Eine detaillierte Beschreibung über Wavlets findet sich in [115] und über die Implementierung der BDW innerhalb der ML-Toolbox in [52].

A.1.3.4 Statistische Momente

Die ML-Toolbox verwendet die vier statistischen Momente (SM) Mittelwert, Standardabweichung, Schiefe (engl. skewness) und Wölbung (engl. kurtosis). Dabei werden bei der Bildung der Statistische Momente (SM) Merkmale mit Informationen über die statistische Verteilung der Daten extrahiert [8].

Für die Berechnung der vier statistischen Momente eines Datenvektors d mit N Beobachtungen gilt:

1. Moment **Mittelwert** μ [189]:

$$\mu = \overline{d} = \frac{1}{N} \sum_{i=1}^{N} d_i \tag{A.3}$$

2. Moment **Standardabweichung** σ [190]:

$$\sigma = \sqrt{\frac{1}{N-1} \sum_{i=1}^{N} \mid d_i - \mu \mid^2} \tag{A.4}$$

3. Moment **Schiefe** s_1 [191]:

$$s_1 = \frac{\frac{1}{N} \sum_{i=1}^{N} (d_i - \mu_d)^3}{\left(\sqrt{\frac{1}{N} \sum_{i=1}^{N} (d_i - \mu_d)^2} \right)^3} \tag{A.5}$$

4. Moment **Wölbung** k_1 [192]:

$$k_1 = \frac{\frac{1}{N} \sum_{i=1}^{N} (d_i - \mu_d)^4}{\left(\frac{1}{N} \sum_{i=1}^{N} (d_i - \mu_d)^2 \right)^2} \tag{A.6}$$

In der automatisierten Variante der ML-Toolbox (vgl. Listing 2.1) werden die Daten bzw. Messzyklen in 10 Abschnitte unterteilt und von jedem der resultierenden Abschnitte die vier SM gebildet.

A.1.3.5 Pearson Korrelation

Mit Hilfe der Pearson-Korrelation bzw. dem Pearsonschen Korrelationskoeffizienten r, mit $r \in [-1, 1]$ wird der lineare Zusammenhang zweier Variablen berechnet [121]. In der ML-Toolbox wird die Korrelation zwischen einer Datenmatrix X und einer Zielgrößenmatrix Y berechnet. Es gilt [193]:

$$r(a,b) = \frac{\sum_{i=1}^{n}(X_{a,i} - \overline{X_a})(Y_{b,i} - \overline{Y_b})}{\sqrt{\{\sum_{i=1}^{n}(X_{a,i} - \overline{X_a})^2(Y_{b,i} - \overline{Y_b})^2\}}} \tag{A.7}$$

mit

$$\overline{X_a} = \frac{1}{n}\sum_{i=1}^{n}(X_{a,i}), \tag{A.8}$$

und

$$\overline{Y_b} = \frac{1}{n}\sum_{i=1}^{n}(X_{b,i}). \tag{A.9}$$

$X_a \in \mathbb{R}^{n\times 1}$ bezeichnet eine Spalte der Datenmatrix X und $Y_b \in \mathbb{R}^{n\times 1}$ eine Spalte der Zielgrößenmatrix Y.

Nachdem die ML-Toolbox den Korrelationskoeffizienten der Merkmale berechnet hat, werden diese nach absteigendem Wert sortiert. Die Selektion der Merkmale erfolgt anschließend durch die `NumFeatRanking()`-Funktion der ML-Toolbox.

Innerhalb der ML-Toolbox wird die Pearson-Korrelation einerseits als Vorselektionsmethode für ReliefF und Recursive Feature Elimination Support Vector Machine (RFESVM), als auch als eigenständige Merkmalsselektionsmethode verwendet (vgl. Abbildung 2.13). Als Vorselektionsmethode werden, sollte die extrahierte Merkmalsanzahl 500 Merkmale überschreiten, die 500 Merkmale mit den höchsten Korrelationskoeffizienten ausgewählt [8].

A.1.3.6 Recursive Feature Elimination Support Vector Machine

Die Recursive Feature Elimination Support Vector Machine (RFESVM) ist eine Wrapper-Methode, welche die Methoden Support Vector Machine (SVM) und Recursive Feature Elimination (RFE) miteinander kombiniert [194]. Dabei versucht die SVM eine multidimensionale Hyperebene

$$< \vec{w}, \vec{x} > +b = 0, \tag{A.10}$$

mit dem Normalenvektor $\vec{w}$, dem Stützvektor $\vec{x}$ und einem Bias b zu finden, um ein binäres Klassifikationsproblem optimal zu lösen. Um Klassifikationsmethoden mit mehr als zwei Klassen zu lösen, können z.B. die Methoden One-vs-All oder Pairwise Classifikation verwendet werden [194].

In einem Vergleich mit 66 Merkmalsselektoren war die RFESVM der zuverlässigste Algorithmus [97]. Vorteilhaft sind hier die Robustheit gegenüber Overfitting und die Berücksichtigung von Merkmalsinteraktionen [8].

In der ML-Toolbox wird eine SVM der L1-Norm (Soft Margin SVM) verwendet, welche die Generalisierbarkeit durch eine gewisse Fehlertoleranz (in Form von Fehlklassifizierungen) erhöht. Es gilt

$$Q(\vec{w}, b, \vec{\xi}) = \frac{1}{2}|\vec{w}|^2 + C\sum_{i=1}^{M}\xi_i, \tag{A.11}$$

mit dem Regulierungsparameter C, der Anzahl an Trainingsdatenpunkten M und der Schlupfvariablen ξ [195]. Durch die Verwendung des Kernel-Tricks können auch nicht linear trennbare Probleme durch die SVM gelöst werden [194]. Eine detaillierte mathematische Beschreibung der SVM, sowie weitere Informationen über Kernel finden sich in [194].

Die Sortierung der Merkmale erfolgt durch die RFE Methode, bei welcher alle Merkmale nach ihrem Beitrag zur Unterteilung der zwei Klassen bewertet werden. Das Merkmal mit dem geringsten Beitrag zur Klassentrennung wird entfernt. Anschließend wird der Prozess wiederholt, bis das beste Merkmal gefunden und somit alle Merkmale nach absteigendem Beitrag sortiert wurden [117, 118].

Die finale Merkmalsanzahl wird durch die `NumFeatRanking()`-Funktion der ML-Toolbox bestimmt.

A.1.3.7 ReliefF

Der ReliefF-Algorithmus basiert auf dem Relief-Algorithmus [196]. Im Gegensatz zur ursprünglichen Version ist ReliefF nicht auf zwei-Klassen-Probleme limitiert [119]. In einem Vergleich unter 66 Merkmalsselektoren konnte ReliefF nach RFESVM die zweit höchste Zuverlässigkeit erzielen [97].

Innerhalb der ML-Toolbox wurde der ReliefF-Algorithmus durch die MATLAB-Funktion `knnsearch()` [197] realisiert. In der Standardkonfiguration der ML-Toolbox werden die drei nächsten Nachbarn mit Hilfe der City-Block-Distance-Metric [198] bestimmt. Dabei werden Merkmale der gleichen Gruppe als Treffer und Merkmale anderer Gruppen als Niete markiert. Anschließend werden die Merkmale von der ML-

Toolbox sortiert. Merkmale, welche eine hohe Distanz zu anderen Gruppen und eine geringe Distanz zur eigenen Gruppe aufweisen, erreichen dabei ein höheres Ranking.

Die Anzahl der selektierten Merkmale wird über die `NumFeatRanking()`-Funktion der ML-Toolbox bestimmt.

A.2 Anwendungsfall 1

A.2.1 Implementierung in PIA

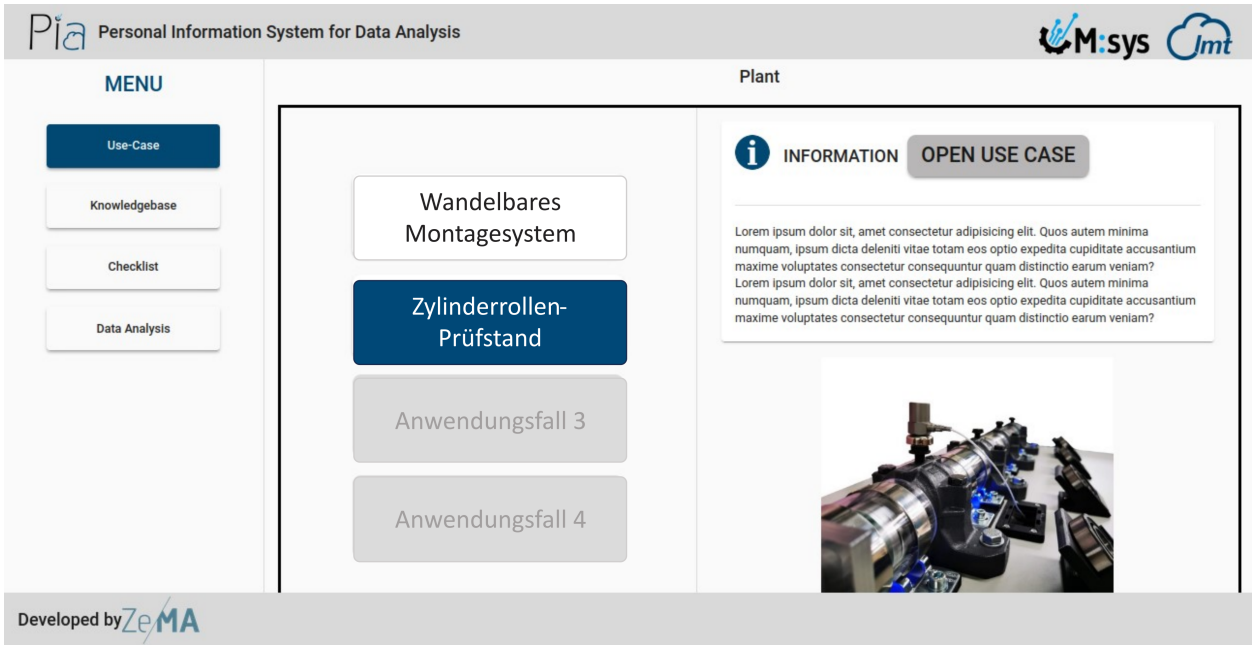


Abbildung A.1: Implementierung des Anwendungsfalls mit Abbildung und einem Platzhalter für eine genauere Beschreibung.

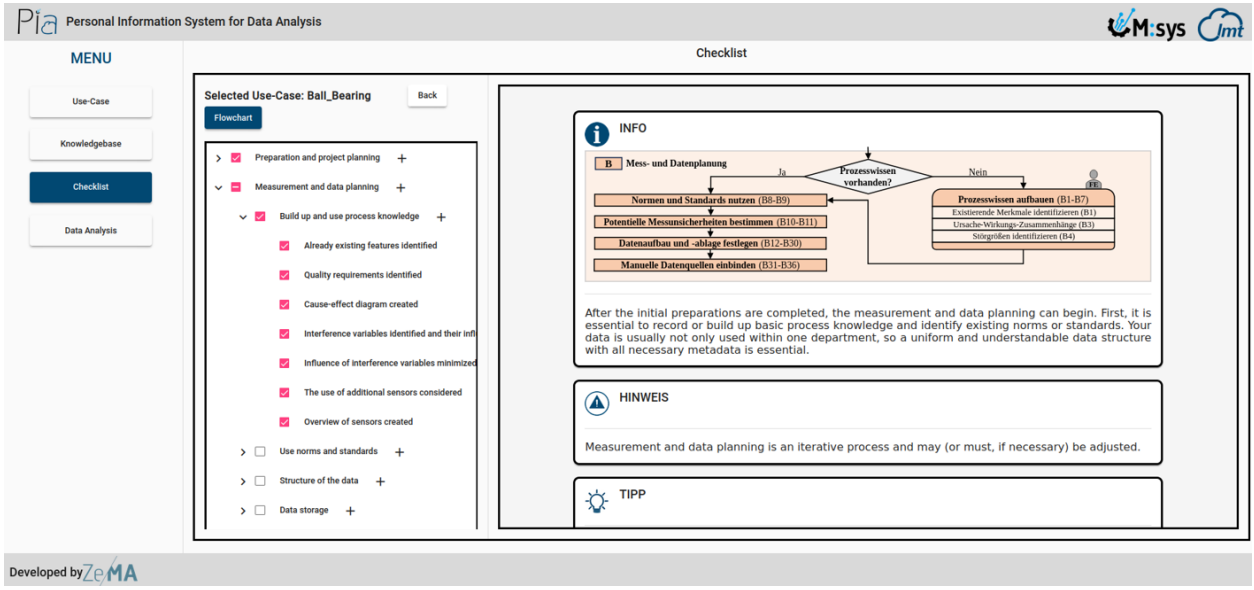


Abbildung A.2: Implementierung der Checkliste in PIA.

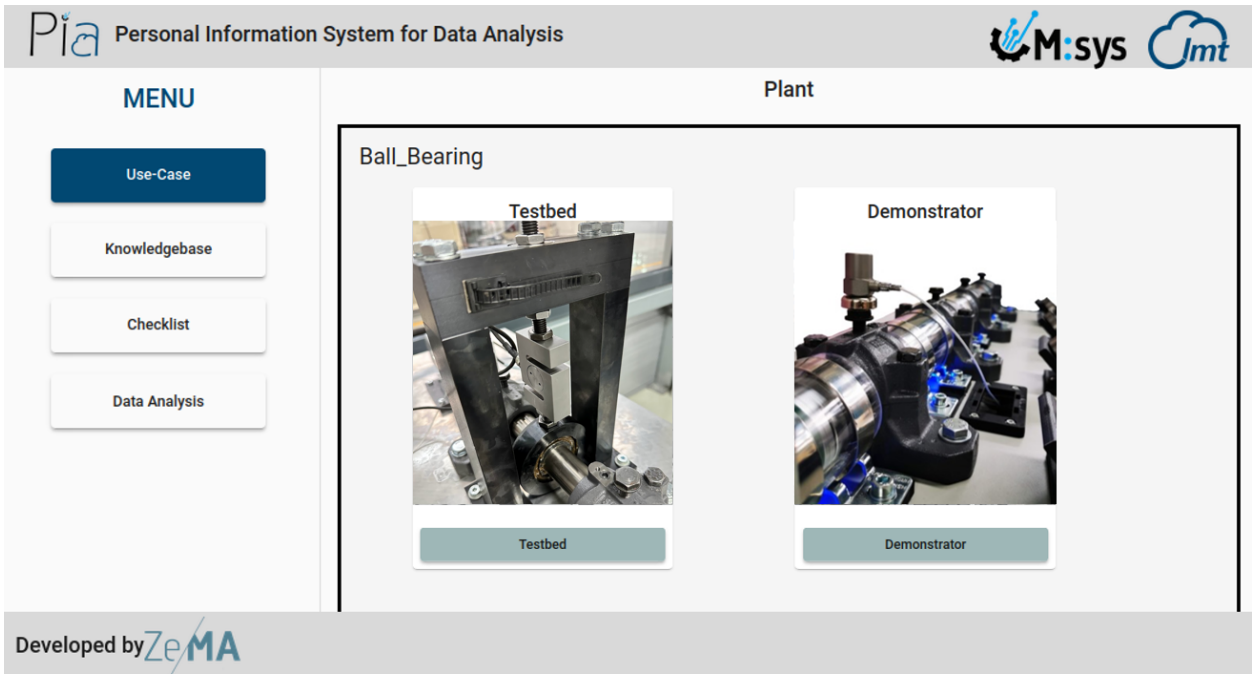


Abbildung A.3: Übersicht der Stationen des Aufbaus. Einerseits fungiert dieser als Prüfstand und andererseits als Demonstrator.

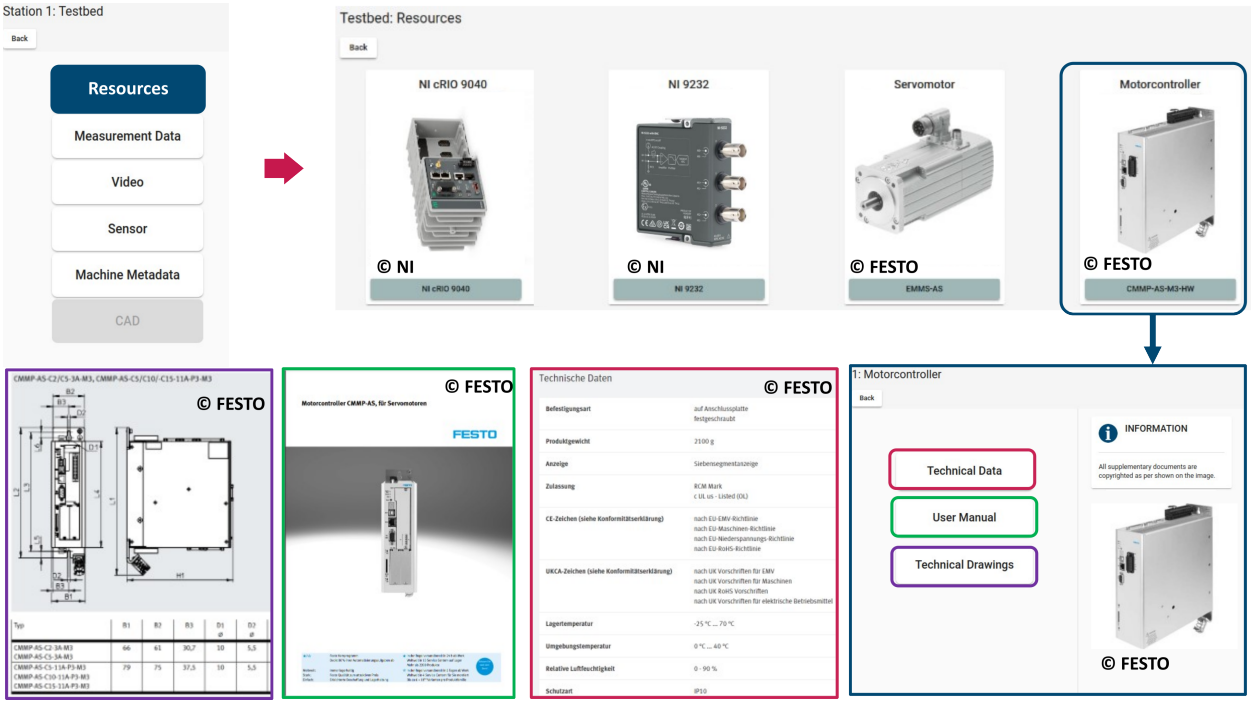


Abbildung A.4: Einbindung der Betriebsmittel (Resources). Für das Betriebsmittel Motorcontroller wurden exemplarisch die jeweiligen Zusatzinformationen Technische Daten (rot), Betriebsanleitung (grün) und Technische Zeichnung (lila) dargestellt.

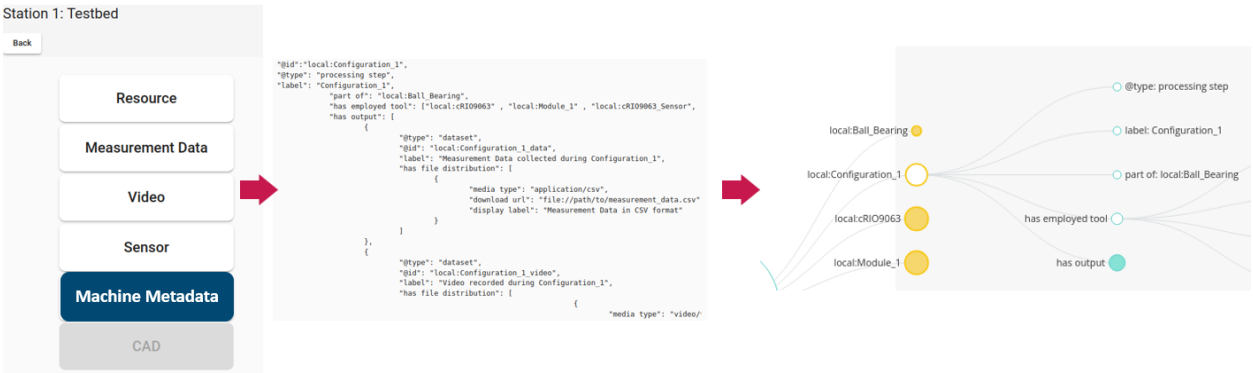


Abbildung A.5: Implementierung der maschinenlesbaren Metadaten. Diese können ausgelesen und die Zusammenhänge visualisiert werden.

Abbildung A.6: Einbindung eines Videos in PIA in dem die Funktionsweise des Aufbaus genauer erklärt wird.

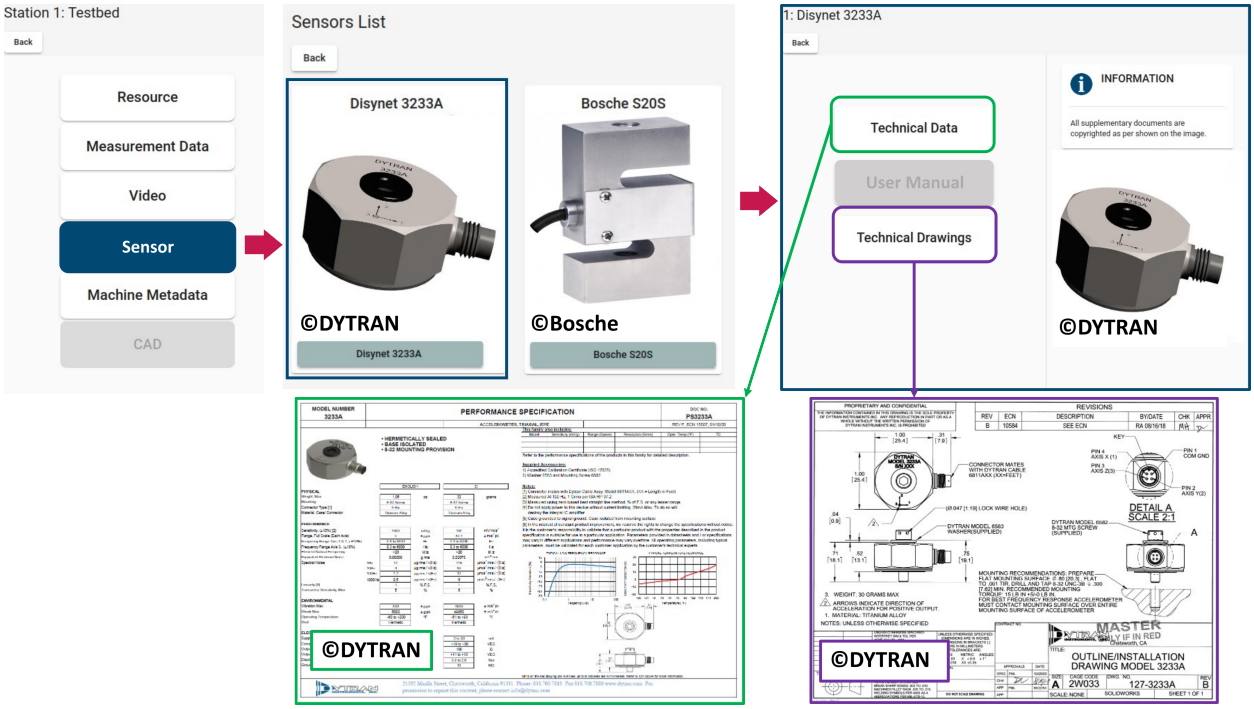


Abbildung A.7: Einbindung der verwendeten Sensorik in PIA und exemplarische Darstellung der Zusatzinformationen Technische Daten (grün) und Technische Zeichnung (lila).

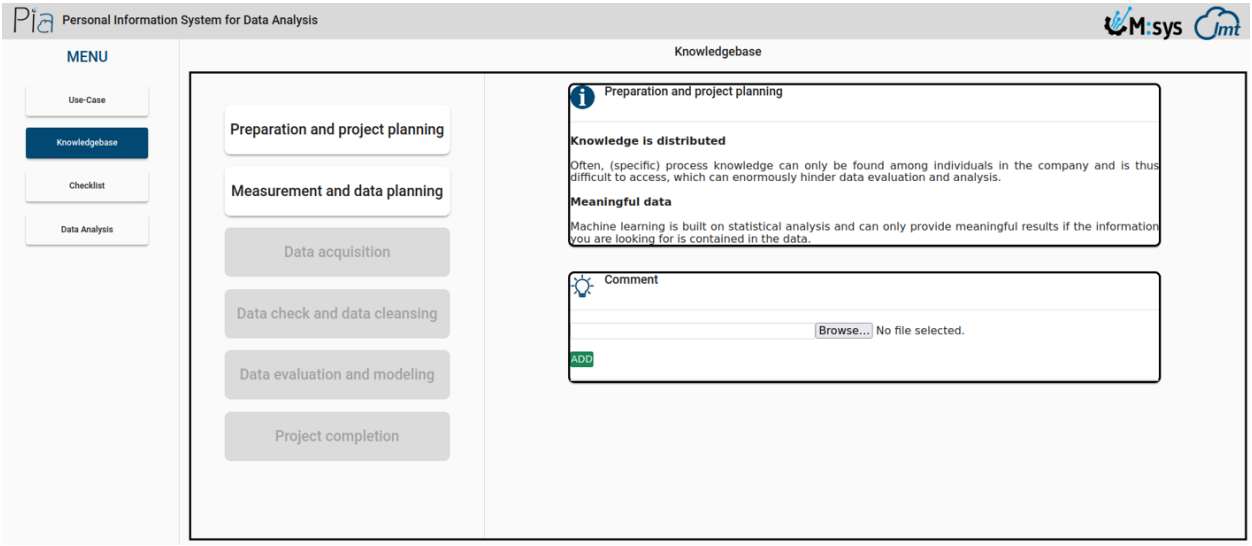


Abbildung A.8: Exemplarische Darstellung der Wissensdatenbank.

A.2.2 Übersicht Sensorik

Übersicht der Sensorik

Mittelstand-Digital Zentrum Saarbrücken

Ersteller: Christopher Schnur
Projekt: VProSaar UC: Zylinderrollenlager
Datum: 26.05.2024

		Sensor 1	Sensor 2	Sensor 3	Sensor 4
Allgemein	Sensorbezeichnung	Beschleunigungssensor	Kraftsensor	Motorcontroller	Temperatursensor
	IP-Adresse	/	/	/	/
	Hersteller	Dytran Instruments	Lorenz Messtechnik	Festo	Govee
	Messgröße und Einheit	Beschleunigung [g]	Last [N]	Rotationsgeschwindigkeit [1/min]	Temperatur [°C] Rel. Luftfeuchte [%]
	Messposition	Lagerdeckel	Kraftaufbringung	Motor	Wand
	Erwartete Effekte	Amplituden bis +-5g	Last bis 4 kN	Drehzahl bis 1000 [1/min]	18°C-20°C
	Anzahl Sensoren	1	1	1	1
Sensordaten	Sensorart	Analog	Analog	Analog	Digital
	Messprinzip	Piezoelektrisch	DMS	Encoder	/
	Auslesemethode	cRIO	cRIO	cRIO	WLAN
	Abtastrate	20000 Hz	20000 Hz	20000 Hz	1 Hz
	Bandbreite	0.3-6000 Hz (z-Achse)	/	/	/
	Auflösung	/	/	/	0.1 °C, %
	Messbereich	+- 5g	+-10000 N	<6450 [1/min]	-20°C – 60°C
	Messunsicherheit	2.7 %	0.1%	/	+- 0,3 °C, +-3%
	Kanäle	3	1	1	1
	Ausgangssignal	V	V	V	/

Abbildung A.9: Übersicht der Sensorik in Anwendungsszenario 1.

A.2.3 Ausschnitt des Versuchsplans

Day 1
Bearing 1 – No Damage

#	Bearing	Case	Damage	Run	Position	Last [N]	Speed [rpm]	Notes
	Bearing	1 – IR 2 – RE 3 - OR	0 – No D 1 - Small (2mm) 2 - large (10mm)	Run 1-3 (1 Run = Pos. A-D)	1 - Pos. A, 2 - Pos. B, 3 - Pos. C, 4 - Pos. D	50, 2500, 1600, 3300	706, 201, 969, 85, 592, 392	E.g.: • Multiple Measurements because first one failed. • Abnormal behavior of ... • Etc.
1	1	1	0	1	1	50	Cycle	
2						2500	Cycle	
3						1600	Cycle	
4						3300	Cycle	
5					2	50	Cycle	
6						2500	Cycle	
7						1600	Cycle	
8						3300	Cycle	
9					3	50	Cycle	
10						2500	Cycle	
11						1600	Cycle	
12						3300	Cycle	
13					4	50	Cycle	
14						2500	Cycle	
15						1600	Cycle	
16						3300	Cycle	
17				2	1	50	Cycle	
18						2500	Cycle	
19						1600	Cycle	
20						3300	Cycle	
21					2	50	Cycle	
22						2500	Cycle	
23						1600	Cycle	
24						3300	Cycle	
25					3	50	Cycle	
26						2500	Cycle	
27						1600	Cycle	
28						3300	Cycle	
29					4	50	Cycle	
30						2500	Cycle	
31						1600	Cycle	
32						3300	Cycle	
33				3	1	50	Cycle	
34						2500	Cycle	
35						1600	Cycle	
36						3300	Cycle	
37					2	50	Cycle	
38						2500	Cycle	
39						1600	Cycle	
40						3300	Cycle	
41					3	50	Cycle	
42						2500	Cycle	
43						1600	Cycle	
44						3300	Cycle	
45					4	50	Cycle	
46						2500	Cycle	
47						1600	Cycle	
48						3300	Cycle	

Check now, that you have 48 files; for each configuration at least one measurement (it can be more if you repeated measurements) with an appropriate file size.

Please hand this sheet to the supervisor and ask to add the small damage.

Legende				
0	1	2	3	4

Name _____ Date _____

Abbildung A.10: Ausschnitt des Versuchsplans für Lager 1 an Messtag 1.

A.2.4 Anwendung der Checkliste

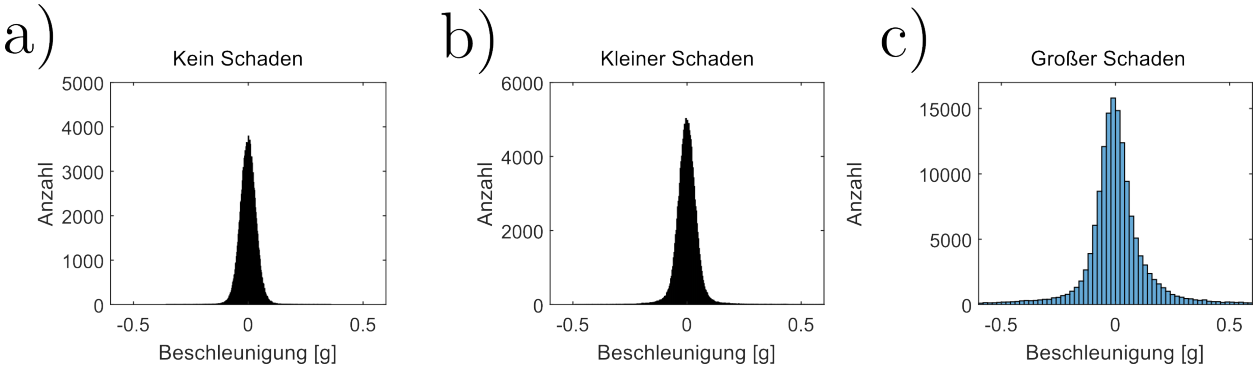


Abbildung A.11: Histogramm des Messungen des Beschleunigungssensors **a)** ohne Beschädigung, **b)** mit kleiner Beschädigung und **c)** mit großer Beschädigung.

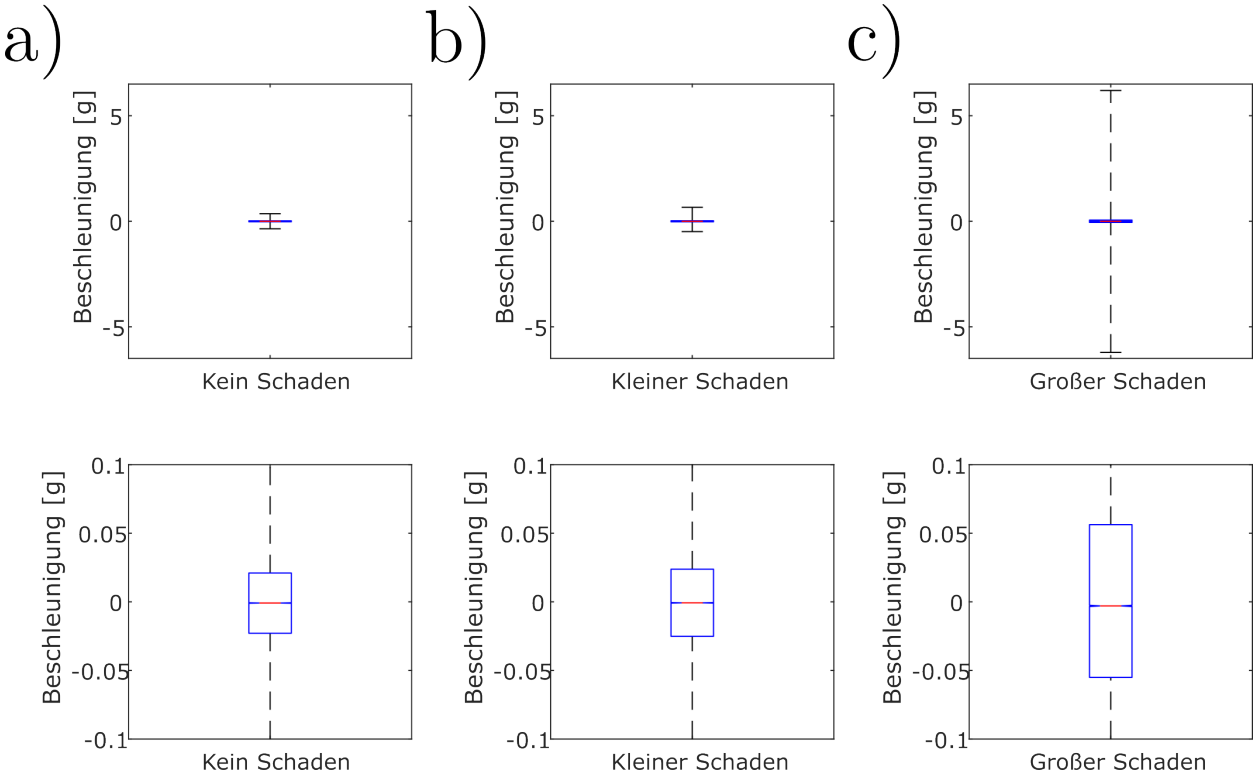


Abbildung A.12: **a)** Boxplot-Diagramme der Messsignale aus Abbildung 5.11 für die Zustände kein Schaden, kleiner Schaden und großer Schaden. **b)** Vergrößerter Ausschnitt aus a).

A.3 Anwendungsfall 2

A.3.1 Implementierung in PIA

Abbildung A.13: Implementierung des Anwendungsfalls mit Abbildung und einem Platzhalter für eine genauere Beschreibung.

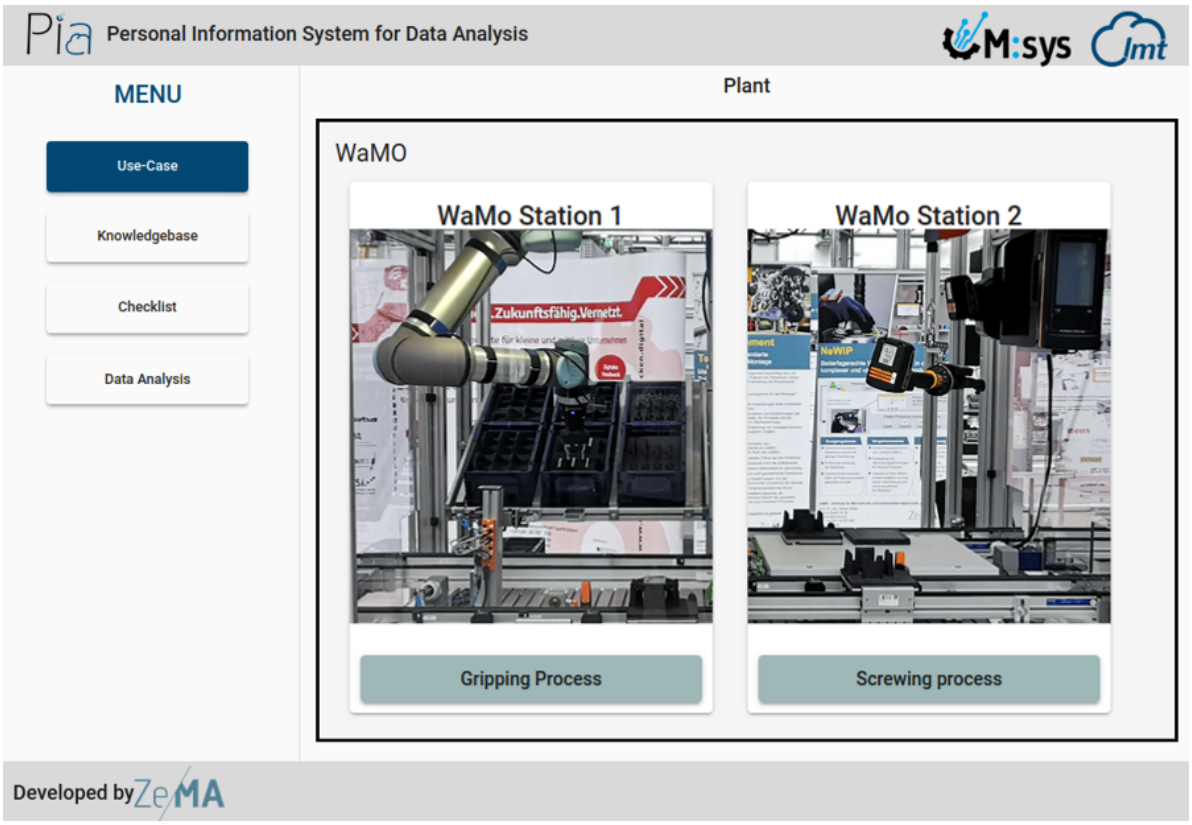


Abbildung A.14: Übersicht der Stationen zwei Stationen Greifprozess (links) und Schraubprozess (rechts) des WaMo.

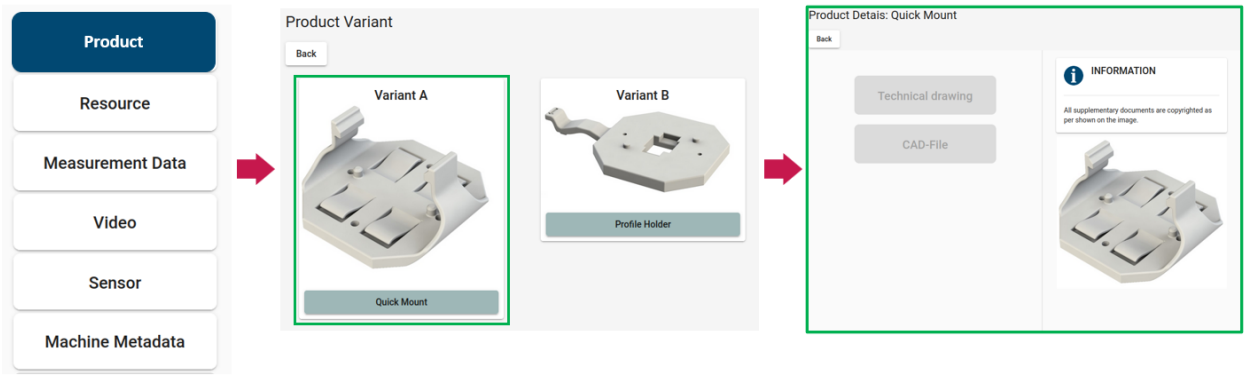


Abbildung A.15: Einbindung des auf der WaMo gefertigten Produkts mit den beiden Varianten A und B.

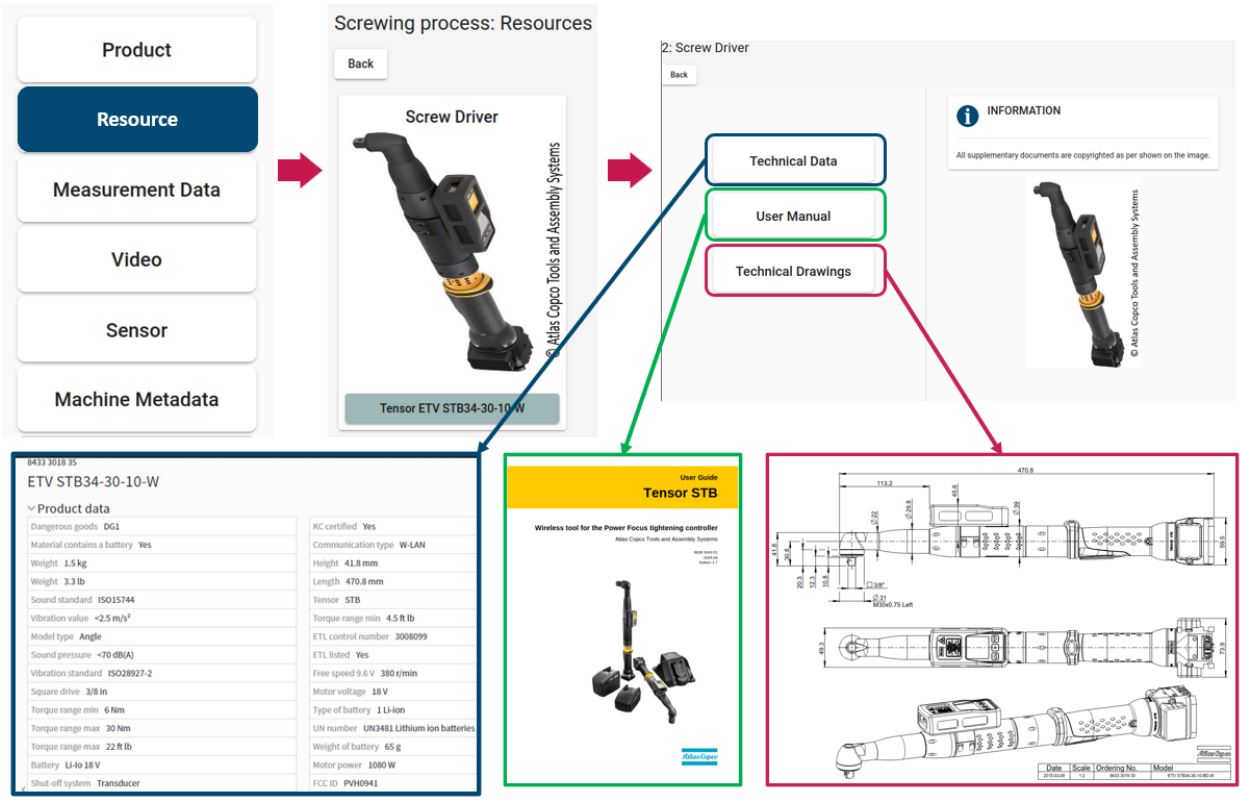


Abbildung A.16: Einbindung der Betriebsmittel. Für das Betriebsmittel Schrauber wurden exemplarisch die jeweiligen Zusatzinformationen Technische Daten (blau), Betriebsanleitung (grün) und Technische Zeichnung (rot) dargestellt.

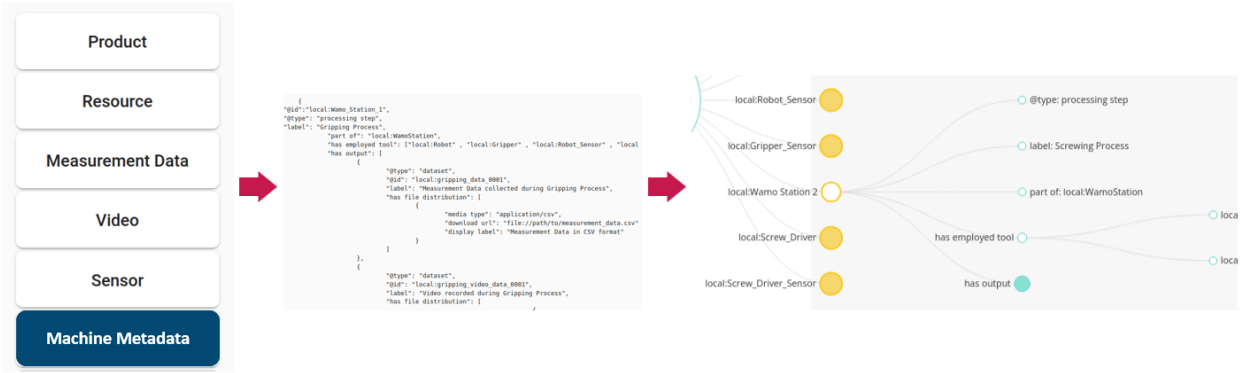


Abbildung A.17: Implementierung der maschinenlesbaren Metadaten des WaMo. Diese können ausgelesen und die Zusammenhänge visualisiert werden.

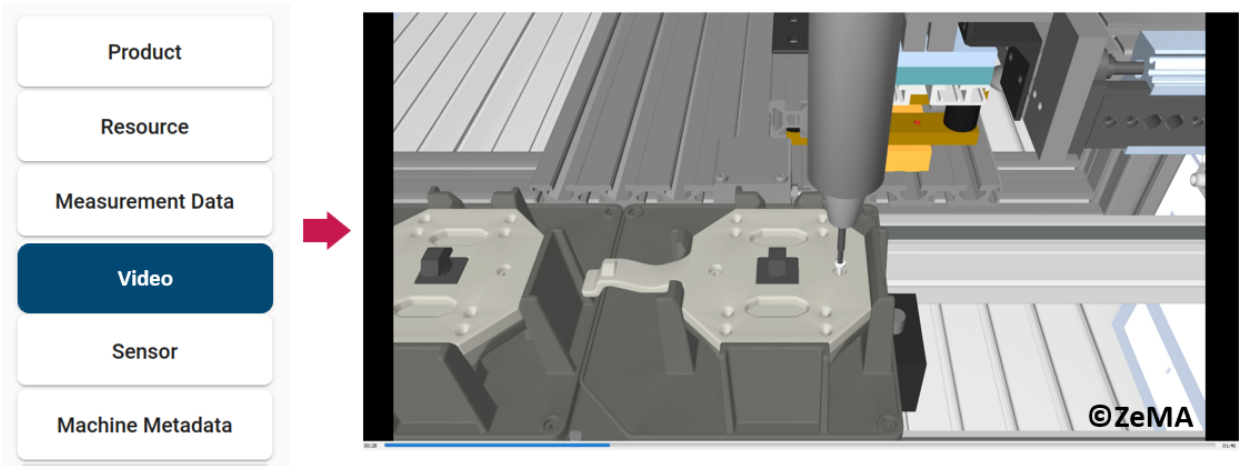


Abbildung A.18: Einbindung eines Videos in dem die Funktionsweise des WaMos genauer erklärt wird.

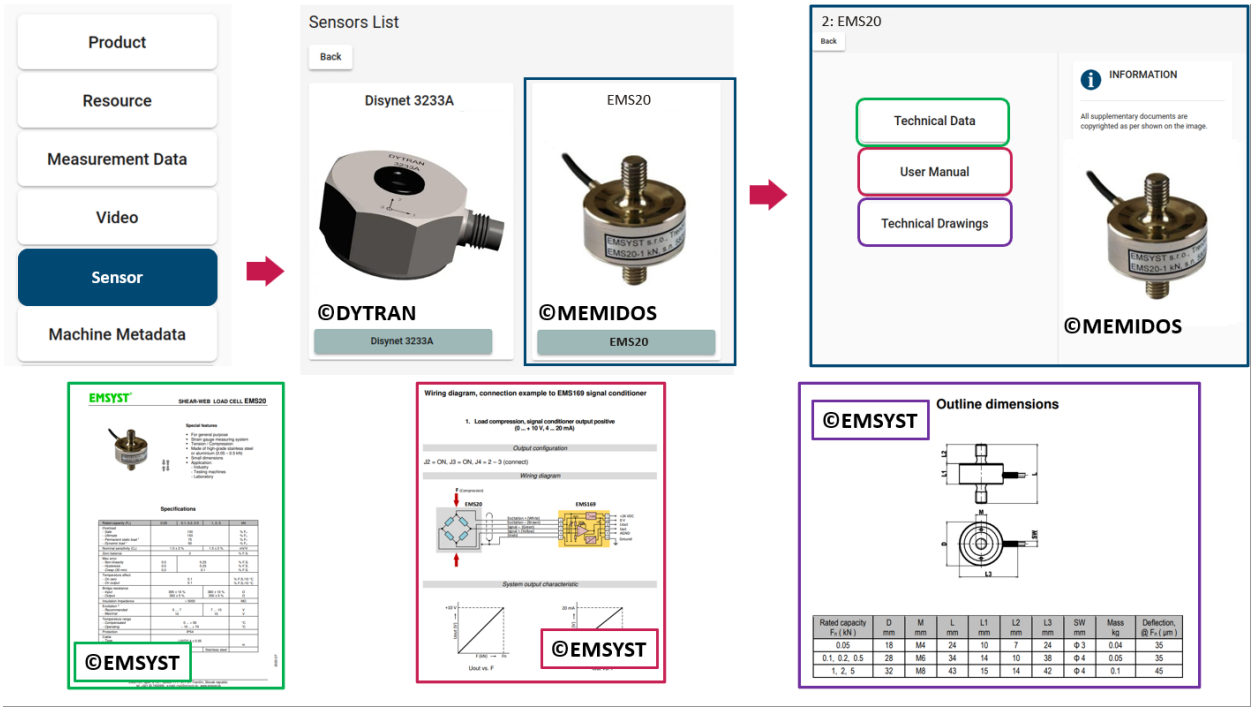


Abbildung A.19: Einbindung der verwendeten Sensorik des WaMo bzw. des Messkoffers in PIA und die exemplarische Darstellung der Zusatzinformationen Technische Daten (grün) und Technische Zeichnung (lila).

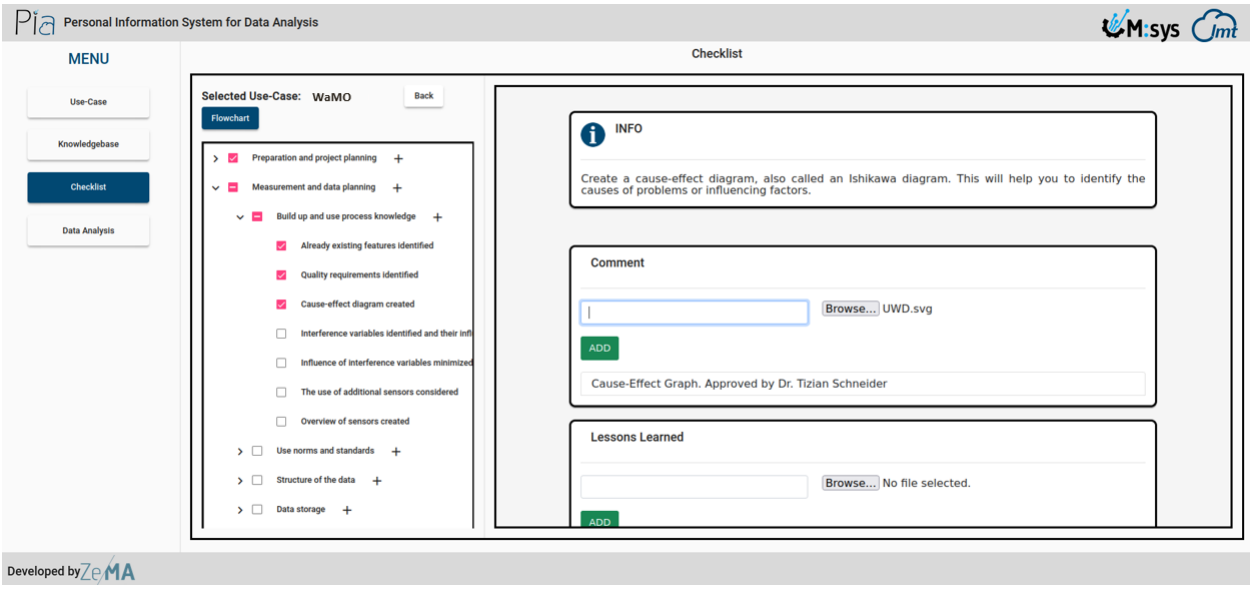


Abbildung A.20: Implementierung der Checkliste in PIA.

A.3.2 15 IQ-Dimensionen

Tabelle A.9: Bewertung der Datenqualität mittels der 15 IQ-Dimensionen in positiv und negativ (nach [67]).

Nr.	IQ-Dimension	Bewertung	Anmerkung
IQ-1	Zugänglichkeit	positiv	Unkompliziert einlesbar.
IQ-2	Angemessener Umfang	positiv	Hinsichtlich Messdauer und Metadaten.
IQ-3	Glaubwürdigkeit	positiv	Daten glaubwürdig.
IQ-4	Vollständigkeit	positiv	Hinsichtlich der Messreihen pro Sensor vollständig.
IQ-5	Übersichtlichkeit	positiv	Daten und Metadaten übersichtlich dargestellt.
IQ-6	Einheitliche Darstellung	positiv	Daten und Metadaten einheitlich.
IQ-7	Bearbeitbarkeit	positiv	Nach dem Einlesen leicht bearbeitbar.
IQ-8	**Fehlerfreiheit**	**negativ**	**Fehlerhafte Messungen des Kraftsensors.**
IQ-9	Eindeutige Auslegbarkeit	positiv	Die Daten sind eindeutig auslegbar.
IQ-10	Objektivität	positiv	Unbearbeitete Rohdaten.
IQ-11	Relevanz	positiv	Relevant für die Machbarkeitsstudie.
IQ-12	Hohes Ansehen	positiv	Datenerfassungssystem von NI kann als vertrauenswürdig eingestuft werden.
IQ-13	Aktualität	positiv	Daten aktuell, da Analyse unmittelbar nach Datenaufnahme.
IQ-14	Verständlichkeit	positiv	Daten und Metadaten verständlich.
IQ-15	Wertschöpfung	positiv	Erlaubt die Einschätzung der Daten vor Start der Messkampagne.

A.3.3 Visualisierung der Daten

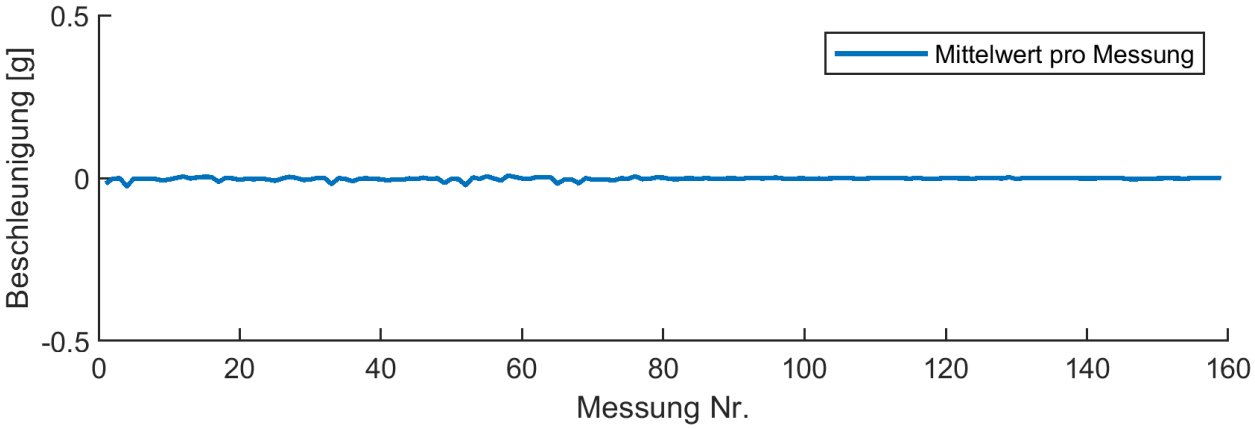


Abbildung A.21: Quasistatisches Signal gebildet aus den Mittelwerten einer Messung des Beschleunigungssensors.

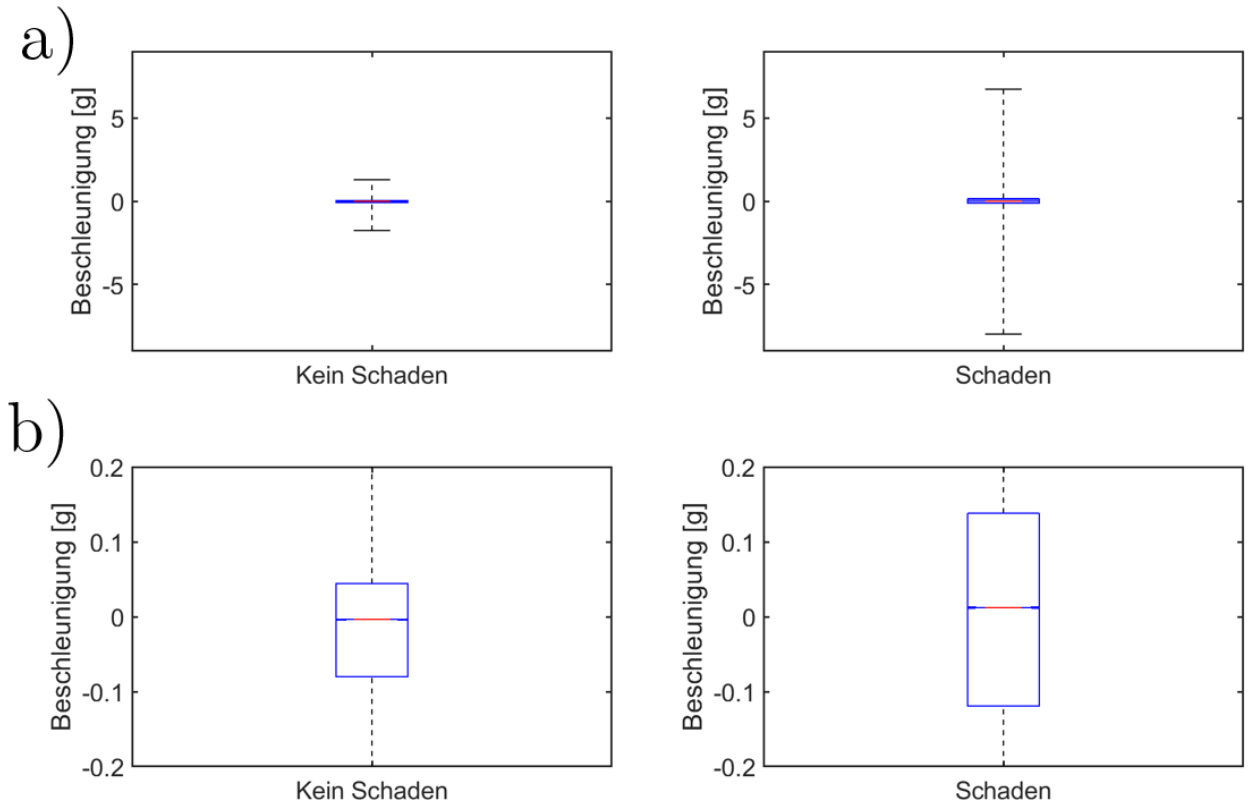


Abbildung A.22: **a)** Boxplot-Diagramme der Messsignale aus Abbildung 5.24 für einen unbeschädigten und einen beschädigten Schrauberbit. **b)** Vergrößerter Ausschnitt aus a).

A.4 Zusammenfassung und Ausblick

Tabelle A.10: Verbesserungspotenziale der Checkliste, wobei Nr. 1-8 allgemeinen, Nr. 9-17 Checkpunkt-spezifischen und Nr. 18 Ablaufplan-bezogenen Verbesserungen entsprechen.

Nr.	Verbesserungspotenzial
1	Entfernen der doppelten Indikatoren (Zahl und Buchstabe) der Überschriften.
2	Neben Muss- und Best-Practice-Checkpunkten sollten als dritte Kategorie die *Fehlervermeidungs*-Checkpunkte eingeführt und explizit als solche ausgewiesen werden.
3	Anwender sollten über die Interpolation- und Extrapolation-Fähigkeit von ML-Modellen informiert werden.
4	Der Umgang mit Kalibrierungs- und Justageprozessen und deren Einfluss auf das ML-Modell sollte in die Checkliste aufgenommen werden.
5	Die Checkpunkte B10 und B11 in Abschnitt 3.3 behandeln die Messunsicherheit, die bereits in Checkpunkt B7 integriert ist. Sie sollten daher in Abschnitt 3.1 vorgezogen werden.
6	Abschnitt 6.1 ist eng mit Kapitel 5 verbunden und sollte in dieses integriert werden.
7	Der Abschnitt Fazit kann in die Einleitung integriert werden.
8	Die Checkpunkte A10 und A14 sollten getauscht werden, da der Zeitplan sinnvollerweise nach der Definition der Lasten erstellt wird.
9	In Checkpunkt B4 kann der Einfluss lediglich durch Expertenwissen abgeschätzt werden, jedoch ohne Daten nicht quantifiziert werden.
10	Checkpunkt B22 sollte allgemeingültiger formuliert werden, wie z.B. *Kennzeichnen von internen Prüfprozessen*.
11	Die Checkpunkte C8 und C9 stellen Alternativen dar, wobei einer der Checkpunkte erfüllt werden muss. Dies muss dem Anwender verdeutlicht werden.
12	Checkpunkt D4 sollte in Kapitel 3 Mess- und Datenplanung vorgezogen werden.
13	Die Bezeichnung von Checkpunkt D6 sollte zu *Referenzen und Nullpunkte sichergestellt* geändert werden, da diese bereits in der Mess- und Datenplanung angepasst wurden.
14	Die Entfernung von Ausreißern in Checkpunkt D8 kann als Teil der Bereinigung von fehlerhaften Messungen in Checkpunkt D5 erfolgen.
15	Die Bezeichnung von Checkpunkt E19 sollte zu *Unsicherheiten des Modells bestimmt* geändert werden.
16	Zwischen Checkpunkt E19 und E20 sollte ein Checkpunkt *Modell mit allen Daten trainiert* eingefügt werden.
17	Checkpunkt F1 sollte als Muss-Checkpunkt ausgewiesen werden.
18	Bei der Abfrage *Modell geeignet* (E18) in Kapitel E wird eine Endlosschleife erzeugt, wenn die Qualität oder Quantität der Daten nicht ausreicht. In den Ablaufplan sollte daher ein Abbruchkriterium integriert werden, wenn die notwendigen Informationen nicht in den Daten vorhanden sind.

Danksagung

Eine Promotion gleicht einer langen Reise, geprägt von Herausforderungen, Erfolgen, Rückschlägen und wertvollen Erkenntnissen. Zu Beginn liegt der Weg zum Gipfel oft im Nebel und erst rückblickend werden die Etappen und Wegpunkte ersichtlich. Trotz der Tatsache, dass der größte Teil dieser Reise allein bewältigt werden muss, wäre sie ohne die Unterstützung zahlreicher Weggefährten nicht möglich gewesen. An dieser Stelle möchte ich all diesen Begleitern meinen aufrichtigen und tief empfundenen Dank aussprechen. In erster Linie möchte ich meinem Doktorvater, Prof. Dr. Andreas Schütze, meinen tiefsten Dank aussprechen. Deine Balance aus Unterstützung, Ansporn und angemessenem Druck war rückblickend genau richtig. Wenn ich auf meiner Reise den Kurs aus den Augen verlor, hast du mich mit sachkundigen Hinweisen und frischen Anregungen wieder auf den richtigen Pfad gebracht. Darüber hinaus gilt mein Dank Prof. Dr.-Ing. Rainer Müller, der sich bereit erklärt hat, das Zweitgutachten zu übernehmen.

Ein weiterer Dank gilt meinen langjährigen Kolleginnen und Kollegen. Einige von euch haben ihre eigene Reise bereits erfolgreich beendet: Dr. Nicolas Michaelis, Dr. Christian Bur, Dr. Tanja Dorst, Dr. Tizian Schneider und Dr. Yannick Robin – vielen Dank für eure Begleitung und Unterstützung auf meinem Weg. Andere stehen kurz vor dem Ziel: Steffen Klein, Henrik Lensch, Julian Joppich und Payman Goodarzi. Wieder andere sammeln gerade Kraft für die letzten Meter: Eliseo Pignanelli und Marco Schott – auch euch danke ich herzlich für eure Weggefährtenschaft. Darüber hinaus möchte ich mich auch bei Julian Schauer, Oliver Brieger, Johannes Amman, Houssam El Moutaouakil, Christian Fuchs, Sebastian Pültz, Dennis Arendes, My Sa Marschibois und Wolfhard Reimringer bedanken. Ihr alle habt einen Teil meines Weges mit mir geteilt und ich bin dankbar für die Zeit und den Austausch mit euch.

Mein besonderer Dank gilt auch den folgenden Personen:

- Dr. Tanja Dorst, deren sorgfältige Kommentare als Lektorin äußerst wertvoll für diese Dissertation waren.

- Dr. Anne Blum, die als langjährige Kollegin mit wertvollen Anmerkungen und stetiger Unterstützung in unseren gemeinsamen Projekten zum Gelingen dieser Arbeit beigetragen hat.
- Christiana Dabove und Harald Nagel für ihre Unterstützung bei der Organisation.
- Timo Klaumann, dessen Fachkenntnisse und Fähigkeiten in der spanenden Bearbeitung mir entscheidend weitergeholfen haben, und der auch unlogisch erscheinende Fertigungsaufträge (wie z.B. die Gewindeplatte in Anwendungsfall 2) für mich durchgeführt hat.
- Kapil Deshmukh, Ali Ali Ahmad und Yage Zhang für die praktische Unterstützung.
- Meinen Freunden, die stets Verständnis dafür gezeigt haben, wenn die Arbeit an der Dissertation wieder einmal Vorrang hatte.
- Meinen Eltern Marion und Bruno, meinen Großeltern Uschi und Rainer, sowie meiner gesamten Familie, die mich nicht nur während meiner Promotion, sondern auch schon während des Studiums unermüdlich unterstützt haben.
- Meiner Frau Aline, die mich während dieser zeitintensiven Reise stets mit Verständnis und Rückhalt begleitet hat.
- Meinem Sohn Lio, dessen Geburt mich dazu brachte, jeden Tag etwas entschlossener an meiner Dissertation zu arbeiten. Zugleich machte er mir bewusst, dass mit dem Ende dieser Dissertation nun eine neue, noch bedeutsamere Reise beginnt.

Abschließend bleibt mir nur zu sagen: Diese Reise war herausfordernd, aber mit eurer Unterstützung habe ich sie erfolgreich gemeistert. Vielen Dank!