technische universität dortmund

IT in Produktion und Logistik

Methode der Wissensentdeckung in Datenbanken und des maschinellen Lernens für die prädiktive Wartung in produzierenden Unternehmen

Masterarbeit zur Erlangung des Grades M. Sc.

Vorgelegt von:	Frederik Beyer
Matrikelnummer:	214015
Studiengang:	Wirtschaftsingenieurwesen

Ausgabedatum:	29.04.2024
Abgabedatum:	14.10.2024

Erstprüferin:	Dr.-Ing. Dipl.-Inform. Anne Antonia Scheidler
Zweitprüfer:	Florian Hochkamp, M. Sc.

Technische Universität Dortmund

Fakultät Maschinenbau

Fachgebiet IT in Produktion und Logistik

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Abkürzungsverzeichnis

AI	Artifizielle Intelligenz
CRISP-DM	Cross Industry Standard Process for Data Mining
DM	Data Mining
ERM	Entity-Relationship-Modell
IoT	Internet of Things
KDD	Knowledge Discovery in Databases
KDID	Knowledge Discovery in Industrial Databases
KNN	K-Nearest Neighbors
ML	Maschinelles Lernen
MSE	Mittlere quadratische Abweichung
RUL	Remaining Useful Lifetime
SEMMA	Sample, Explore, Modify, Model, Asses
SMOTE	Synthetic Minority Over-sampling Technique

1 Einleitung

Digitale Daten werden als „Rohstoff“ der vierten industriellen Revolution bezeichnet (Arbeitskreis Smart Service Welt 2015). Die Datenmenge auf der Welt wächst derzeit exponentiell und es wird prognostiziert, dass dieser Trend auch in Zukunft anhalten wird (IDC 2023). Dies wird durch die Tatsache verdeutlicht, dass im Jahr 2017 90% aller Daten weltweit erst in den letzten zwei Jahren entstanden sind (Kolanovic und Krishnamachari 2017). Diese Entwicklung unterstreicht die enorme Relevanz und Aktualität des Themengebietes Datenwissenschaften, in dem es darum geht, Daten zu sammeln und sinnvoll zu nutzen, um bessere Entscheidungen treffen zu können (Zulqarnain 2024).

Digitale Daten allein generieren jedoch keinen Mehrwert für Unternehmen. Daher werden im Rahmen der Datenwissenschaften diverse Methoden, Prozesse und Algorithmen eingesetzt, um die digitalen Daten in entscheidungsunterstützendes Wissen zu transformieren (Provost und Fawcett 2013). Die Anwendung diverser Methoden der Datenwissenschaften eröffnet produzierenden Unternehmen die Möglichkeit, aus Daten nutzbringendes Wissen zu generieren. Die Umsetzung dieser Methoden kann zu einer Steigerung der Wirtschaftlichkeit des Unternehmens beitragen. Einer dieser Methoden ist das Vorgehensmodell zur Wissensentdeckung in Datenbanken (KDD). Die verschiedenen KDD-Vorgehensmodelle umfassen einen strukturierten Ansatz, der Elemente aus den übergeordneten Bereichen Statistik, maschinelles Lernen, Mathematik und Datenbanken integriert (Portela 2022). Das Ziel besteht in der Extraktion von Wissen aus strukturierten und unstrukturierten Daten zu einer zuvor definierten Fragestellung. Insbesondere das maschinelle Lernen wird im Rahmen von KDD-Vorgehensmodellen und anderen datenbasierten Methoden als entscheidendes Werkzeug erachtet, um mehrwertstiftendes Wissen und Prognosen auf Grundlage von digitalen Daten zu generieren (Chavan et al. 2023). Gemäß dem Gartner Hype Cycle, welcher den Zeitraum der Aufmerksamkeit für eine neue Technologie sowie deren Reifegrad beschreibt, erreichte das maschinelle Lernen im Jahr 2016 seinen Höhepunkt an Aufmerksamkeit und entwickelte sich in den Folgejahren als etablierte Technologie (Gartner 2016).

Der technische Fortschritt in der Produktion, exemplarisch demonstriert durch IoT-Geräte, welche die Übertragung von Sensordaten mittels drahtloser Kommunikation ermöglichen, resultiert in einer steigenden Verfügbarkeit digitaler Daten in Unternehmen, welche für Produktions- und Planungsprozesse relevant sind (Soori et al. 2024). Zur Verarbeitung dieser aufgenommenen digitalen Daten ist im industriellen Kontext die strukturierte Anwendung von datenverarbeitenden Methoden von Vorteil, um Parameter wie Kosten und Zeit in der Produktion effizient gestalten zu können (Lieber et al. 2013b). Die Reduzierung von Kosten und die Steigerung der Produktionszeit sind in

produzierenden Unternehmen beispielsweise durch das Erkennen von Qualitätsproblemen bei Produkten oder die Reduzierung von Anlagenausfallzeiten durch prädiktive Wartung möglich. Seit Beginn der Industrie 4.0 sind verschiedene Konzepte der intelligenten Fertigung entstanden, von denen Kusiak 2017) die prädiktive Wartung als einen der wichtigsten Bereiche identifiziert hat. In den vergangenen Jahren konnte ein signifikanter Anstieg der Relevanz der prädiktiven Wartung für Industrieunternehmen beobachtet werden. Der Markt für prädiktive Wartung wird in den USA zwischen 2023 und 2030 nach Prognosen eine jährliche Wachstumsrate von 29,5 % aufweisen (Grand View Research 2023). Die Unternehmen befassen sich mit diesem Thema, da sie eine Verbesserung der Produktionseffizienz und Anlagenverfügbarkeit durch die Reduzierung von Ausfallzeiten in den Produktionslinien erwarten.

Das Ziel dieser Arbeit ist es, die Themengebiete der prädiktiven Wartung, des maschinellen Lernens sowie der KDD zu verknüpfen, um effiziente und robuste Wartungsempfehlungen prognostizieren zu können. Insbesondere soll untersucht werden, inwiefern sich KDD zur Vorbereitung von Algorithmen des maschinellen Lernens im Rahmen der prädiktiven Wartung eignet. Im Zuge dieser Untersuchung wird in der wissenschaftlichen Arbeit eine Methode für das Thema prädiktive Wartung entwickelt, die zunächst die Anwendung von KDD-Vorgehensmodellen umfasst, deren generiertes Wissen anschließend Algorithmen des maschinellen Lernens zugeführt werden kann.

Die vorliegende Arbeit eröffnet mit einer Einführung in das Thema der Datenwissenschaften, da sich alle nachfolgenden Themen in diesen Kontext einordnen lassen. In der Literatur existieren verschiedene Vorgehensmodelle zur Wissensentdeckung in Datenbanken, von denen einige anschließend vorgestellt und miteinander in Beziehung gesetzt werden. Danach wird das Thema maschinelles Lernen und einige Algorithmen aus den Bereichen überwachtes und unüberwachtes maschinelles Lernen beschrieben. Um Hintergrundwissen für den Anwendungsbereich prädiktive Wartung zu schaffen, wird dieses Thema zudem aufgegriffen und in den Kontext der Instandhaltung sowie anderer Wartungspraktiken gesetzt.

Im dritten Kapitel erfolgt eine Zusammenführung der beiden Themengebiete „Maschinelle Lernverfahren” und „KDD-Vorgehensmodelle” im Kontext der prädiktiven Wartung. Nachdem das Grundlagenkapitel mit einer Zusammenfassung der datenwissenschaftlichen Herausforderungen bei der Einführung von prädiktiven Wartungsstrategien in produzierenden Unternehmen schließt, soll im Anschluss erarbeitet werden, wie die in dieser Arbeit entwickelte Methode diesen Schwächen entgegenwirken kann. Zunächst werden die Anforderungen an die Eingangsdaten für Algorithmen des maschinellen Lernens

im Rahmen der prädiktiven Wartung erörtert. In der Folge wird eine Methode entwickelt, welche die Integration von KDD-Vorgehensmodellen und maschinellem Lernen umfasst und als prädiktive Wartungsstrategie dienen soll. Die in dieser Arbeit entwickelte Methode wird im vierten Kapitel anhand eines Beispieldatensatzes validiert. Im fünften Kapitel erfolgt eine Diskussion des Nutzens der entwickelten Methode sowie eine Erwähnung weiterführender Forschungsmöglichkeiten.

2 Grundlagen der Datenwissenschaften und der prädiktiven Wartung in produzierenden Unternehmen

Dieses Kapitel liefert zunächst Grundlagen zur Disziplin Datenwissenschaften, indem elementare Begriffe definiert und KDD-Vorgehensmodelle als Teildisziplin der Datenwissenschaften beleuchtet werden. Es folgt die grundlegende Beschreibung des maschinellen Lernens als weitere Teildisziplin der Datenwissenschaften. Zuletzt werden im zweiten Kapitel Instandhaltungsstrategien mit dem Fokus auf die prädiktive Wartung vorgestellt, womit dem Lesenden eine Einordnung und ein Gesamtüberblick zur prädiktiven Wartung und seinen Zusammenhängen geliefert wird. Anschließend werden die zuvor genannten Themengebiete des maschinellen Lernens, der KDD-Vorgehensmodelle sowie der prädiktiven Wartung zusammengeführt, um auf dieser Grundlage eine Methode zu entwickeln, welche im dritten Kapitel präsentiert wird.

2.1 Grundbegriffe der Datenwissenschaften

Datenwissenschaften ist das übergeordnete Thema dieser Arbeit, da die beiden Themen KDD-Vorgehensmodelle und maschinelles Lernen, die später im Anwendungsgebiet prädiktive Wartung miteinander verknüpft werden, als Teildisziplin der Datenwissenschaften gelten. Die Datenwissenschaft wird als eine interdisziplinäre Praxis definiert, die auf diverse Techniken zurückgreift, um Datenmengen zu analysieren und interpretieren (Fawcett und Provost 2013). Die gewonnenen Erkenntnisse bilden die Grundlage für eine datengestützte Entscheidungsfindung. Das disziplinübergreifenden Forschungsfeld der Datenwissenschaften ist geprägt durch Strömungen aus verschiedenen Bereichen, darunter Mathematik, maschinelles Lernen, künstliche Intelligenz, Statistik, Datenbanken und Optimierung (Dhar 2012). Diese Vielfalt an Perspektiven ist laut Dhar (2012) von zentraler Bedeutung, um das Hauptziel der Datenwissenschaften, die verallgemeinerbare Gewinnung von Wissen aus Daten, aus unterschiedlichen Blickwinkeln zu betrachten. Neben den technischen und analytischen Aspekten sind darüber hinaus oftmals die domänenspezifischen Fähigkeiten von Experten von entscheidender Bedeutung für ein erfolgreiches datenbasiertes Ergebnis (Vinay und Pub 2024).

2.1.1 Daten, Informationen und Wissen

Die Basis für die in dieser Arbeit behandelten Themen bildet die Kenntnis grundlegender Begriffe aus dem Bereich der Daten sowie die Fähigkeit, aus Daten nutzbares Wissen abzuleiten. Die Generierung von fundierten Entscheidungen auf Basis von Daten stellt eine wesentliche Grundlage für Unternehmen dar, um auf operativer wie auch auf strategischer Ebene adäquate Entscheidungen zu treffen (Tao et al. 2018). Per Definition sind Daten Objekte und Beziehungen mit der Eigenschaft, durch Merkmale beschrieben

werden zu können (Mertens et al. 2017). Die Merkmalsausprägungen verschiedener Daten ermöglichen Strukturierungsansätze wie Klassifikationen in Abhängigkeit von der Merkmalsausprägung. Für die Strukturierungsansätze müssen die Daten in möglichst einheitlicher Form vorliegen, um vergleichbar zu sein. Dies kann Vorarbeiten erfordern, da die Daten bereits in strukturierter, aber auch in unstrukturierter Form vorliegen können. Strukturierte Daten, wie beispielsweise Daten aus Transaktionsvorgängen, liegen bereits in einem standardisierten Format vor und können schematisch in eine Datentabelle geschrieben werden, wobei jede Spalte ein Attribut und jede Zeile einen Datensatz enthält (Chen et al. 2009). Die Daten bestehen meist aus Werten wie Zahlen, Kurztext oder Datumsangaben. Unstrukturierte Daten hingegen lassen sich nicht in eine einheitliche Form bringen, da es sich zum Beispiel um Bilder, Videos oder Texte handelt (Chen et al. 2009). Gerade die starke Zunahme unstrukturierter Datenformate in den letzten Jahren hat zu Ansätzen wie Big Data beigetragen, bei denen es vor allem um komplexere Datenstrukturen geht, die mit den Methoden der traditionellen Datenverarbeitung nicht bearbeitet werden können (Intel IT Center 2012). NESSI, ein Zusammenschluss von Akteuren aus Industrie, Forschung und Wissenschaft, definiert Big Data folgendermaßen:

„Big Data ist ein Begriff, der die Verwendung von Techniken zur Erfassung, Verarbeitung, Analyse und Visualisierung potenziell großer Datenmengen in einem angemessenen Zeitrahmen umfasst, die mit Standard-IT-Technologien nicht zugänglich sind“ (NESSI White Paper 2012, S. 6)

Der Begriff "Big Data" bezeichnet somit nicht nur eine große Datenmenge, sondern ein umfassendes Konzept, welches Informationen, Technologien, Methoden und Auswirkungen berücksichtigt (Dumbill 2013). Ein Ansatz zur Identifikation von Daten als Big Data bietet das 5-V-Modell nach NESSI, welches sich auf die Eigenschaften „Volume“, „Velocity“, „Variety“, „Veracity“ und „Value“ bezieht (NESSI White Paper 2012). Das Modell stellt die wohl geläufigste Hilfestellung zur Einordnung von Daten in die Datenlandschaft Big Data dar (Lovelace 2016). Eines der Identifikationskriterium für Big Data ist demnach eine sehr große Datenmenge, wobei die Daten schnell generiert und ausgewertet werden können. Die Daten werden schnell produziert und anschließend analysiert, wobei die Verarbeitungsgeschwindigkeit kontextabhängig zu betrachten und je nach technischer und wirtschaftlicher Intention an den Anwendungsfall anzupassen ist (Quix 2021). „Veracity“ besagt, dass sich die Datenqualität in einem regulatorischen Rahmen bewegen sollte, der Vertrauen bezüglich inhaltlicher Richtigkeit und Herkunftsquelle der Daten schafft (Hiba et al. 2015). Schließlich zeichnet sich ein Big-Data-

Datensatz dadurch aus, dass er einen Nutzen liefert, um unternehmerischen Mehrwert zu schaffen (NESSI White Paper 2012).

Vorherrschende Modelle wie die von North entwickelte Wissenstreppe bieten einen schrittweisen Prozessansatz zur Erklärung der Generierung von Wissen oder des unternehmerischen Mehrwerts aus Daten (North 2011). Die Abbildung 1 präsentiert einen Ausschnitt des stufenweisen Ansatzes der Wissenstreppe von North (North 2011).

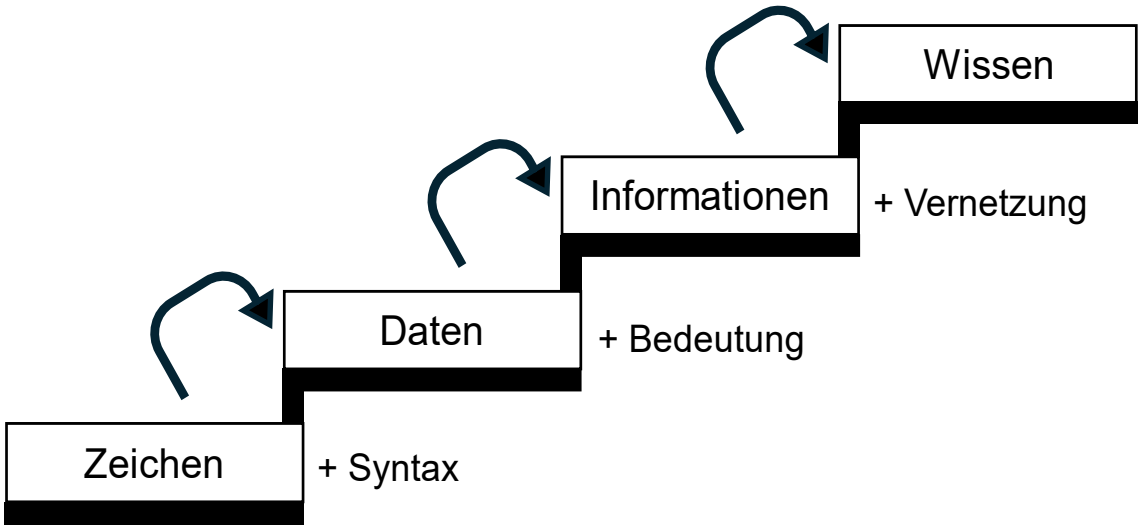


Abbildung 1: Wissenstreppe nach North (in Anlehnung an North 2011, S.36)

Die Wissenstreppe zeigt, dass Zeichen die Grundlage von Daten sind, aus Daten Informationen entstehen können und Informationen die Vorstufe zu Wissen sind. Die unterste Stufe der Wissenstreppe wird durch Zeichen abgebildet (Krcmar 2015). Zeichen können beispielsweise in Form von Buchstaben oder Ziffern auftreten (Krcmar 2015). Eine Strukturierung der Zeichen durch eine einheitliche Syntax ist Voraussetzung dafür, dass Zeichen zu Daten werden können. North beschreibt weiter, dass Daten selbst nicht aussagekräftig sind und erst an Bedeutung gewinnen, wenn ein kontextueller Bezug zu den Daten hergestellt wird. Stehen die Daten in einem Bezugsrahmen zu einem bestimmten Thema, gelten sie schließlich als Informationen. Erst durch die Vernetzung von Informationen unterschiedlicher Art kann schließlich Wissen entstehen. Die Generierung von Wissen ist stets an eine menschliche Beteiligung geknüpft und daher als personengebunden zu betrachten (Orth et al. 2018). Die individuellen Erfahrungen und der jeweilige Kontext des Einzelnen beeinflussen folglich den Prozess der Wissensentstehung (Probst et al. 2003). In dieser Arbeit wird der Begriff des Wissens gemäß der Definition von diskursivem Wissen verwendet, welches durch ein methodisches Vorgehen und eine logische Beweisführung generiert wird (Schütte 1998).
In den finalen Phasen des North-Modells wird dargelegt, dass durch

Vernetzungstechniken wie Strukturierung und Verarbeitung erworbenes oder entdecktes Wissen zu unternehmerischen Kompetenzen gebündelt werden kann, welche letztlich zu Wettbewerbsvorteilen führen. Die Entwicklung von Daten zu Wissen lässt sich anhand der Eigenschaften der Abstufungen veranschaulichen. Daten werden dabei zunächst als strukturiert, isoliert und kontextunabhängig beschrieben, während Wissen als unstrukturiert, vernetzt und kontextabhängig gilt (Bodendorf 2003).

Um den aktuellen Herausforderungen, welche die Industrie 4.0 mit sich bringt, zu begegnen, wurde eine Erweiterung der Wissenstreppe von North, die Wissenstreppe 4.0, konzipiert (North und Maier 2018). Diese dient der Bewältigung der Herausforderungen, die sich im Laufe der Zeit manifestiert haben. Die Weiterentwicklung digitaler Technologien hat einen signifikanten Einfluss auf den Prozess der Umwandlung von Daten in Wissen, welches als Grundlage für die Generierung von Wertschöpfung dient (North und Maier 2018). Um die damit einhergehenden Zusammenhänge adäquat zu beschreiben, wurde die Wissenstreppe von North und Maier um die Dimensionen „Technologie“, „Mensch und Organisation“ erweitert. Diese Entwicklung kann als Antwort auf die durch Wissen 4.0 entstandenen Themen wie die kognitiven und vernetzten Systeme, die Digitalisierung des Alltags und von Einrichtungen betrachtet werden. Softwarelösungen wie High-Performance Data Analytics ermöglichen eine automatisierte Vorverarbeitung großer Datenmengen, wodurch diese in Informationen transformiert werden (Institut für DLR Softwaretechnologie 2022). Die Ergänzung der Wissenstreppe um Modelle aus dem maschinellen Lernen, mit denen Sensordaten automatisiert verarbeiten und Muster erkannt und interpretiert werden können, stellt eine wesentliche Neuerung dar. Die neu hinzugekommene Ebene „Mensch und Organisation“ verweist außerdem auf Phänomene wie Emojis als zusätzliche Ausprägungsform von Symbolen oder den mobilen Zugriff auf Daten.

Die Wissenstreppe veranschaulicht in vereinfachter Form die Kernaufgabe der Datenwissenschaften, nämlich die Extraktion von Wissen aus Daten zur Lösung unternehmerischer Probleme. Für die praktische Implementierung dieses Prozesses sind jedoch leistungsfähige Datenbanksysteme unerlässlich, die nachfolgend näher beschrieben werden.

2.1.2 Datenbankensysteme

Datenbanksysteme fungieren in der Regel sowohl als Datenquelle für die weitere Verarbeitung der Daten als auch als Speicherplatz für Ergebnisdaten datenwissenschaftlicher Methoden (Mertens et al. 2017). Infolgedessen werden sie in dieser Einführung näher beleuchtet.

Ein Datenbanksystem dient der dauerhaften Speicherung, Beschreibung und dem jederzeitigen Abruf von Daten und besteht wie in Abbildung 2 zu sehen aus einer Datenbank und einem zugehörigen Datenbankmanagementsystem für den administrativen Zugriff auf die Datenbank (Ester und Sander 2000; Mertens et al. 2017).

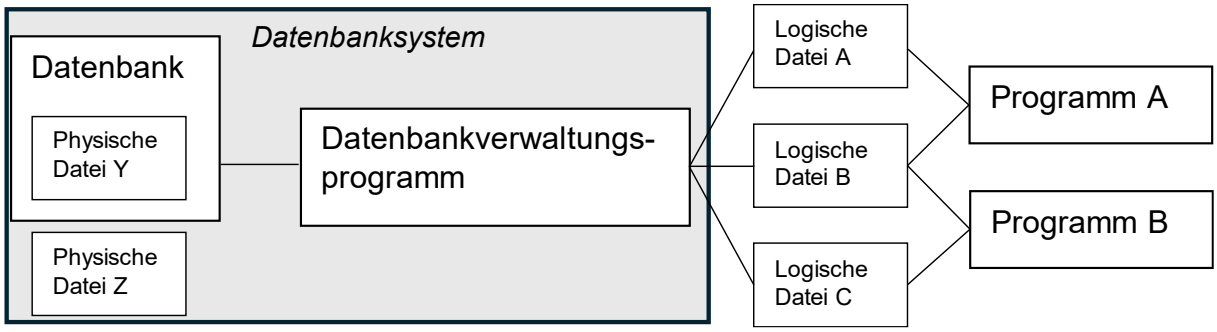


Abbildung 2: Datenbanksysteme (in Anlehnung an Mertens et al. (2017), S.40)

Mertens et al. (2017) führen weiterhin aus, dass die Datenbank selbst die Funktion eines Sammelpunktes für die Speicherung zusammengehöriger Daten erfüllt, während das Datenbankmanagementsystem ein Programmsystem ist, mit dem die Zugriffsrechte der einzelnen Anwendungsprogramme auf die Datenbank und die Änderungsrechte an den Inhalten verwaltet werden. Die zentrale Datenbank erlaubt zwar den Zugriff der einzelnen Anwendungsprogramme, hat aber eine zentrale Position und ist somit unabhängig von den einzelnen Anwendungsprogrammen. Das Datenbankmanagementsystem stellt den zugreifenden Programmen lediglich eine eigens erstellte logische Datei zur Verfügung. Die ursprüngliche physische Datei verbleibt redundanzfrei und konsistent in der Datenbank (Mertens et al. 2017).

Datenbanken gibt es in unterschiedlichen Modellausprägungen, je nach Anforderungen an Eigenschaften wie die Speicherstruktur der Daten, die Komplexität der Datenstruktur, die genauen Anforderungen an die Nutzung der Daten, aus denen sich z.B. Eigenschaften wie die Abfragegeschwindigkeit ergeben, oder die Kostenunterschiede, die bei der Nutzung unterschiedlicher Datenbankmodelle entstehen (Meier und Kaufmann 2016). Übergeordnet lassen sich die Datenbankmodelle zum einen in eine strukturierte Datenhaltung einteilen, die datenbankinterne Auswertungen z.B. über die Abfragesprache SQL ermöglicht, dies sind vor allem relationale Datenbanksysteme (Chavan et al. 2023). Unstrukturierte Daten hingegen erfordern eine komplexere Analyse durch z.B. maschinelles Lernen und müssen zudem meist vorverarbeitet werden, um sie in ein gewünschtes Format zu bringen, das für die Analyse und Weiterverarbeitung geeignet ist (Dhar 2012; Chen et al. 2009).

Datenbanksysteme fungieren zusammenfassend als Speicher- und Verwaltungsort von Daten, die anschließend, je nach Absicht, mit verschiedenen Modellen und Methoden weiterverarbeitet werden können. Daher erfolgt im Folgenden eine Einordnung der Begriffe "Modelle" und "Methoden" im Kontext der Datenwissenschaften.

2.1.3 Modelle und Methoden

Wie zuvor dargelegt, besteht das Ziel der Datenwissenschaften in der Generierung von Wissen. Zu diesem Zweck existieren diverse Modelle sowie korrespondierende Methoden.

Die Modellbildung zielt darauf ab, ein System in vereinfachter Form darzustellen und zu analysieren (Gutenschwager et al. 2017). In der Konsequenz wird das reale Bezugssystem nie vollständig abgebildet, sondern je nach Modellzweck ein mehr oder weniger detaillierter Systemausschnitt modelliert. Des Weiteren sind die Abbildungseigenschaft eines natürlichen oder künstlichen Systems sowie die pragmatische Eigenschaft, welche die Zweckgebundenheit von Modellen beschreibt und deren Einsatz in bestimmten Anwendungsbereichen sogar die Ersetzung des Originalsystems ermöglicht, als Merkmale zu nennen (Stachowiak 1973). In einer Präzisierung des Zwecks eines Modells differenzieren 2017 zwischen der Erklärungskraft eines Modells und entscheidungsorientierten Absichten. In diesem Kontext sind reine Datenmodelle, Organigramme oder Softwarearchitekturen als Beispiele für beschreibende, weniger aber für erklärende oder entscheidungsorientierte Modelle zu nennen. Referenzmodelle und Prognosemodelle hingegen verfolgen in erster Linie entscheidungsunterstützende und erklärende Absichten. Diese Eigenschaften werden noch stärker von mathematischen Optimierungsmodellen oder Simulationsmodellen erfüllt. Eine besondere Art von Referenzmodellen stellen die sogenannten Vorgehensmodelle dar (Lemke und Brenner 2015). Diese fungieren für bestimmte thematische Anwendungen als Leitfaden für ein mögliches Vorgehen (Schütte 1998). Ein wissenschaftlich fundiertes Vorgehensmodell basiert dabei auf einer Reihe von Rahmentätigkeiten und Methoden bestehend aus verschiedenen Aufgaben, deren schrittweise Anwendung empfohlen wird, um die jeweiligen Problemstelllungen zu bewältigen (Pressman 2005).

Methoden zielen darauf ab, durch die Anwendung unterschiedlicher Vorgehensweisen und einer objektiven Sichtweise wissenschaftliche Erkenntnisse zu gewinnen (Mariscal et al. 2010). Sie repräsentieren die einzelnen Instrumente, welche die Erreichung der Zielsetzung des Vorgehensmodells gewährleisten sollen. Schatten et al. 2010 beschreiben, dass Vorgehensmodelle im Projektmanagement eine wesentliche Funktion erfüllen, da sie allen Projektbeteiligten einen strukturierten Vorgehensrahmen mit einzelnen

Prozessschritten bereitstellen und auch dem Auftraggeber eine transparente Beschreibung des Prozessablaufs liefern. Sie stellen somit eine dokumentierte Strategie für ein Projekt dar, welche Empfehlungen bezüglich der zeitlichen Abfolge der Prozessschritte organisiert und in der Regel auch inhaltliche Qualitätsmerkmale beinhaltet (Schatten et al. 2010). Vorgehensmodelle sollten im Allgemeinen für den Anwender gut handhabbar, problemfrei anwendbar und mit einem messbaren Erfolg verbunden sein (Pressman 2005). Dabei präsentiert das Vorgehensmodell eine Zusammenfassung der erforderlichen Maßnahmen zur Lösung einer Problemstellung sowie der integrierten Methoden zu deren Umsetzung (Mariscal et al. 2010).

Demgegenüber können eher beschreibende und konzeptionelle Modelle unmittelbar mit der in Kapitel 2.1.2 dargelegten Datenbankthematik in Verbindung gebracht werden (Elmasri und Navathe 2010). So werden in Datenmodellen Beziehungen und Datenklassen definiert, bevor entschieden wird, mit welchem Datenbanksystem die Daten später gespeichert und weiterverarbeitet werden (Meier und Kaufmann 2016). Das Entity-Relationship-Modell (ERM) stellt beispielsweise eine Darstellungsform zur Erstellung eines konzeptionellen Datenbankmodells dar, welches anschließend in Datenbankschemata, wie die relationale Datenbank, überführt werden kann (Kurbel 2024). Ausgangspunkt dieses Modells, das an feste grafische Modellierungsregeln gebunden ist, ist eine klar identifizierbare und charakteristische Entität. Beziehungen zu anderen Entitäten werden durch Relationen dargestellt, die durch die Angabe von Mengenkardinalitäten näher definiert werden (Deeg und Ditze 2010).

Nachdem im vorangehenden Kapitel Grundlagen zu Datenmodellen dargelegt wurden, widmet sich das folgende Kapitel der Datenanalyse. Diese ermöglicht die gezielte Auswertung und Nutzbarmachung der durch Modelle und Methoden erfassten Informationen.

2.1.4 Datenanalyse

Die Datenanalyse stellt ein wesentliches Element der Datenwissenschaft dar. Ihre Untergliederung erfolgt in Abhängigkeit von der jeweiligen Funktion in unterschiedliche Teilbereiche.

Die deskriptive Datenanalyse zielt auf die Beschreibung und grafische Darstellung von Daten ab, um einen Überblick über die betrachteten Daten zu gewinnen. Dies erfolgt beispielsweise durch die Darstellung von Diagrammen, Graphen und Tabellen oder die Berechnung von Mittelwert und Streuung. Dies ermöglicht die Erstellung eines ersten Überblicks über die Daten sowie die Detektion etwaiger Fehler (Fahrmeir et al. 2023).

Weitere statistische Verfahren können auf Datensätze angewendet werden, um beispielsweise ungewollte Phänomene wie Rauschen in den Daten oder Ausreißer zu kompensieren. Unter Rauschen wird die Präsenz zufälliger, ungewollter und irrelevanter Datenpunkte in einem Datensatz verstanden (Bishop 2016). Als Ausreißer werden Datenpunkte bezeichnet, die durch Fehler bei der Datenerfassung, statistische Streuung oder außergewöhnliche Bedingungen deutlich von den übrigen Werten eines Datensatzes abweichen (Cios et al. 2007).

Die explorative Datenanalyse hingegen, bei der Daten auf Strukturen und Zusammenhängen untersucht werden, stellt eine Weiterführung der deskriptiven Datenanalyse dar. Methodisch wird das Gebiet der klassischen Statistik durch computergestützte Verfahren erweitert, sodass Untersuchungen auch auf große Datenmengen, wie sie im Zuge von Big Data auftreten, angewandt werden können (Fahrmeir et al. 2023). Klassische statistische Maßzahlen können nach wie vor dazu beitragen, einen Datensatz besser zu verstehen und zu bewerten (Cleve und Lämmel 2020). Zudem können sie zur Erfolgsbewertung der Anwendung von Data Mining oder Algorithmen des maschinellen Lernens herangezogen werden. So geben statistische Kennzahlen beispielsweise Aufschluss über die Signifikanz von Analyseergebnissen. Die Zuverlässigkeit eines Modells zur Vorhersage lässt sich beispielsweise mit der mittleren quadratischen Abweichung (MSE) überprüfen, welche die Abweichung zwischen einer Schätzfunktion T und dem tatsächlichen Wert θ angibt (Fahrmeir et al. 2023). Damit errechnet der MSE, wie gut ein Modell im Durchschnitt den tatsächlichen Wert θ vorhersagt.

Die reine Statistik kann bei komplexerer Datenauswertung aber an ihre Grenzen stoßen, sodass es zu unterschiedlichen Analyseergebnissen kommen kann, wenn dieselbe Datenbasis von verschiedenen Statistikern ausgewertet wird (Chavan et al. 2023). Die Ursache hierfür ist in den großen Datenmengen zu finden, aus denen einzelne Datensätze bestehen können. Diese lassen sich mit KDD-Techniken besser untersuchen als mit statistischen Ansätzen (Fahrmeir et al. 2023). Neben einer Zunahme von strukturierten Daten ist ein exponentieller Anstieg unstrukturierter und heterogener Daten zu verzeichnen, welche sich einer einheitlichen und vergleichbaren Strukturierung entziehen und aus komplexen Beziehungen zwischen einzelnen Datenelementen bestehen (Dhar 2012). Durch diese Zunahme von Big Data und damit größeren Datensätzen mit mehr Datenstrukturen werden neben den klassischen statistischen Methoden weitere Datenanalysetechniken benötigt, die der Wissensgenerierung dienen (Fayyad et al. 1996b; Xiaoling Shu und Yiwan Ye 2023). Hierfür stehen im Rahmen des Data Mining Verfahrensansätze zur Verfügung, die auf den bestehenden klassischen Methoden der Statistik und des maschinellen Lernens aufbauen und sowohl zur Analyse von strukturierten

als auch unstrukturierten Daten verwendet werden (Imielinski und Mannila 2000; Mariscal et al. 2010). Data Mining umfasst verschiedene methodische Ansätze die aus Techniken, Methoden und Algorithmen bestehen, um in Massendaten nach unbekannten Mustern und Zusammenhängen zu suchen (Han et al. 2012). In produzierenden Unternehmen können dies Daten über einen Fertigungsprozess sein, aus denen mit Hilfe von Data Mining nützliches Wissen gewonnen werden soll (Xiaoling Shu und Yiwan Ye 2023). Um einen Bezug zur Wissenstreppe von North (vgl. Abbildung 1) herzustellen wird die Anwendung von Data Mining vollzogen, um von der Datenbasis oder darauf basierenden Informationen an Wissen auf Basis von gefundenen Mustern oder zusammenhängen in den Daten zu gelangen.

In Kapitel 2.1 wurden einige grundlegende Konzepte der Datenwissenschaften erörtert. In diesem Kontext wurde zunächst der Begriff der Daten definiert. Aus Daten kann in mehreren Stufen Wissen generiert werden, das beispielsweise zur Optimierung von Prozessen in produzierenden Unternehmen nutzbringend eingesetzt werden kann. Im Anschluss wurde die Funktion von Datenbanken als Speicher- und Verwaltungsmedium für Daten erörtert. Die in den Datenbanken gespeicherten Daten können mit Hilfe verschiedener Methoden und Modelle analysiert werden. Es wurde dargelegt, dass eine rein statistische Analyse bei der Wissensentdeckung an ihre Grenzen stoßen kann. In der Folge werden in Kapitel 2.2 die sogenannten KDD-Vorgehensmodelle präsentiert.

2.2 Wissensentdeckung in Datenbanken

Im vorherigen Kapitel wurde bereits das Data Mining erwähnt, welches einen Teilprozess der Wissensentdeckung in Datenbanken darstellt (Fayyad et al. 1996b). Im Folgenden soll der Gesamtprozess der Wissensentdeckung in Datenbanken erörtert und eine Auswahl an dafür entwickelten Vorgehensmodellen vorgestellt und grob verglichen werden.

Die Begriffe KDD und Data Mining werden von einigen Autoren synonym verwendet (Adriaans und Zantinge 1998). Für diese Arbeit wird sich jedoch zugrunde gelegt, dass Data Mining den Kernprozess der Wissensentdeckung darstellt, jedoch als alleiniger Schritt keine aussagekräftigen neuen Erkenntnisse aus den Daten generieren kann (Fayyad et al. 1996b). Der alleinige Einsatz von Data-Mining-Methoden führt sogar zu nicht aussagekräftigen bis hin zu fehlerhaften Ergebnissen (Gabriel et al. 2011; Mariscal et al. 2010). Erst in Kombination mit vor- und nachgelagerten Prozessschritten kann von dem Gesamtprozess zur Wissensentdeckung in Datenbanken gesprochen werden (Frawley et al. 1992). Diese Thematik wird seit dem ersten KDD-Workshop im Jahre 1989 in der wissenschaftlichen Literatur diskutiert und in der Industrie angewendet (Frawley et al.

1992). Frawley et al. 1992 führen aus, dass der Grundidee zufolge das Wissen, welches das Resultat eines datenbasierten Entdeckungsprozesses ist, aus unterschiedlichen methodischen Perspektiven zu betrachten ist. KDD ist in dieser Zeit von den Einflüssen verschiedener Bereiche geprägt worden, die sich mit der Generierung von Modellen aus Daten beschäftigen. Zu den einflussnehmenden Bereichen zählen die Mustererkennung mittels angewandter Statistik, maschinellem Lernen und neuronalen Netzwerken sowie das Themengebiet Datenbanksysteme (Ester und Sander 2000; Fayyad et al. 1996b).

Die in Kapitel 2.1.2 erwähnten Datenbankabfragen erweisen sich für KDD als nur bedingt geeignet. Die Verwendung einfacher Abfragen, wie beispielsweise die Standardsprache SQL in relationalen Datenbanken, ist lediglich zur Zusammenfassung von Daten zu einer bestimmten Fragestellung sinnvoll. Dabei ist eine definierte Fragestellung vorgegeben und auf Grundlage dessen werden Daten herausgegeben, die dem abgefragten Muster entsprechen (Dhar 2012). Dhar 2012) führt weiterhin aus, dass KDD hingegen darauf abzielt, systematisch nach Mustern zu suchen, die sich aus den betrachteten Daten ableiten lassen. Fayyad et al. definieren KDD demnach als

„[…] nicht-triviale[n] Prozess der Identifizierung gültiger, neuer, potenziell nützlicher und letztlich verständlicher Muster in Daten“ (Fayyad et al. 1996a, S. 43).

Im Laufe der Zeit haben sich zur systematischen Wissensentdeckung in Datenbanken verschiedene Vorgehensmodelle etabliert. Als Vorgehensmodell wird die Abfolge von iterativen und mit dem Anwender gekoppelten interaktiven Prozessschritten bezeichnet, welche als Leitfaden für die Durchführung von KDD dienen sollen (vgl. Kapitel 2.1.3). Die Vorgehensmodelle dienen durch die Beschreibung von auszuführenden Rahmentätigkeiten mit jeweiligen Inputs und Outputs als Orientierungshilfe, um im Allgemeinen ein Verständnis über das spezifische Projekt und die zu untersuchenden Daten zu erlangen (Pressman 2005). Im Anschluss verfolgen die Vorgehensmodelle das Ziel, die betrachteten Daten strukturiert aufzubereiten, zu analysieren und auf diese Weise zu Ergebnissen zu gelangen, die anschließend angewendet werden können. Die Vorgehensmodelle stellen somit jeweils einen möglichen vereinheitlichenden Rahmen zur Durchführung von KDD dar. Die Vorgehensmodelle unterstützen durch ihre schrittweise Prozessbeschreibung die Generierung nützlichen Wissens, sodass nicht wahllos Data-Mining-Methoden zum Einsatz kommen, die ohne sinnvolle vor- und nachgelagerte Prozessschritte zu nicht brauchbarem Wissen führen können (Fayyad et al. 1996a). Die Ausführung der Prozessschritte, welche nachfolgend in detaillierter Form beschrieben werden, kann bis zur Interpretation, durch unterschiedliche Tools und Algorithmen weitgehend automatisiert erfolgen. Dazu wurden Softwarelösungen entwickelt, die vorgefertigte Bausteine

für den KDD-Prozess bereitstellen. Die Evaluierung der Sinnhaftigkeit und Nützlichkeit des erworbenen Wissens bedingt jedoch eine subjektive Evaluierung durch eine menschliche Instanz, wie bereits bei der Definition des Wissensbegriffs von North beschrieben wurde (Fayyad et al. 1996a; North 2011).

2.2.1 Vorgehensmodelle zur Wissensentdeckung in Datenbanken

In diesem Kapitel erfolgt eine Beschreibung und Analyse ausgewählter Vorgehensmodelle zur Wissensentdeckung in Datenbanken. Dabei wird insbesondere auf die einzelnen Prozessschritte sowie die Ziele, die die Autoren ursprünglich mit der Erstellung der KDD-Vorgehensmodelle verfolgten, eingegangen.

Modell von Fayyad et al. (1996)

Das Modell von Fayyad et al. (1996) gilt als ursprüngliches Vorgehensmodell zur Wissensentdeckung in Datenbanken. Es wurde erstmals im Jahre 1996 in einer Fachzeitschrift veröffentlicht und soll Organisationen bei der Generierung von Wissen aus der stetig zunehmenden Verfügbarkeit von Daten unterstützen. Infolge des rapiden Anstiegs digital verfügbarer Daten erachten Fayyad et al. (1996) die dringende Notwendigkeit, computergestützte Methoden zur Gewinnung von Wissen aus Daten zu entwickeln (Fayyad et al. 1996b). KDD zielt darauf ab, den Menschen zu entlasten, für den die stetig wachsende Datenmenge nicht mehr handhabbar ist. Der wissenschaftliche Vorschlag für ein Vorgehensmodell von Fayyad et al. (1996) gliedert sich hauptsächlich in die in Abbildung 3 dargestellten fünf Vorgehensschritte und einige Zwischenschritte, die Fayyad et al. (1996) jedoch nicht separat in ihrer Grafik zu dem Vorgehensmodell aufführen.

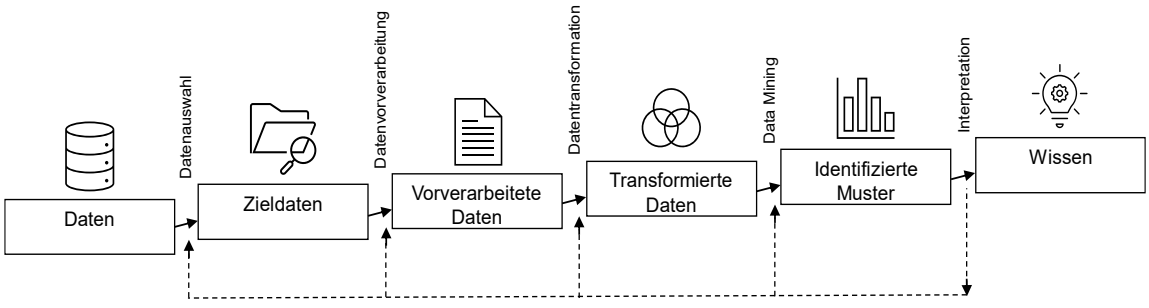


Abbildung 3: KDD-Vorgehensmodell nach Fayyad (in Anlehnung an Fayyad et al. 1996b, S.3)

Das Modell ist prozessorientiert und umfasst mehrere Prozessschritte, die teilweise miteinander in Beziehung stehen und bei Bedarf iterativ durchlaufen werden. Fayyad et al. 1996b) beschreiben folgende Prozessschritte:

1. **Datenauswahl:** Im ersten Schritt erfolgt die Auswahl der Daten. Im Rahmen dieses Schritts erfolgt die Selektion eines adäquaten Datensatzes. In Bezug auf den Zieldatensatz ist zu eruieren, ob die gewünschten Daten bereits in einer Datenbank oder einer ähnlichen Datenstruktur vorhanden sind oder noch erhoben werden müssen. Der Datensatz stellt den Input für die Untersuchungen dar und ist von enormer Bedeutung, da die Zielerreichung von der Qualität des behandelten Datensatzes abhängt.
2. **Vorverarbeiten der Daten:** Im zweiten Schritt erfolgt die Vorverarbeitung der Daten. Durch die Anwendung diverser Techniken erfolgt eine Vervollständigung des Datensatzes sowie dessen Konsistenzprüfung, um eine möglichst korrekte, einheitliche und aktuelle Datenbasis zu gewährleisten. In Abhängigkeit der Zielsetzung kann es erforderlich sein, Daten aus unterschiedlichen Quellen zu einem Gesamtdatensatz zu integrieren oder die Anzahl der relevanten Variablen zu reduzieren.
3. **Transformieren der Daten:** Die Transformation erfolgt in eine Darstellungsform, die im Anschluss die Anwendung direkter Methoden des Data Mining ermöglicht. Hier sei als Beispiel die Attributselektion genannt (Ester und Sander 2000).
4. **Data Mining:** Die eigentliche Mustersuche wird im Schritt Data Mining durch die Anwendung von Algorithmen auf den transformierten Datensatz durchgeführt.
5. **Interpretation:** Die durch das Data Mining identifizierten Muster bedürfen einer Interpretation, um sie als valides Wissen zu legitimieren. Im Rahmen dieses Schritts erfolgt zudem eine Evaluierung des Beitrags des entdeckten Wissens zur Gesamtproblemlösung.

Die Datenauswahl zu Beginn des Prozesses ist eng mit den beiden zusätzlichen Prozessschritten „Bereitstellung von Hintergrundwissen" und „Zieldefinition" verknüpft. Die Zieldefinition erfordert je nach Anwendungsziel die Formulierung einer Hypothese, auf deren Grundlage die relevanten Daten ausgewählt werden (Fayyad et al. 1996b). In diesem Kontext wird zwischen einem überprüfenden Ansatz, bei dem eine aufgestellte Hypothese verfolgt wird, und einem explorativen Ansatz unterschieden. Letzterer zielt darauf ab, Muster zu entdecken, ohne dass zuvor formulierte Hypothesen existieren müssen (Fayyad et al. 1996c). Bei der Formulierung der Hypothese sollten bereits vorhandenes Vorwissen oder Expertenmeinungen zum behandelten Thema berücksichtigt werden. Ein weiterer Zwischenschritt ist die Auswahl der Data-Mining-Methode. Diese soll mit Bedacht getätigt werden, bevor der eigentliche Data-Mining-Algorithmus durchgeführt wird. In die Auswahlentscheidung zur richtigen Methode fließen das genaue Ziel

des KDD-Prozesses und der verwendete Datentyp, auf den der Algorithmus angewendet wird, ein (Ester und Sander 2000).

Fayyad et al. (1996) führen darüber hinaus einen vierten zusätzlichen Prozessschritt auf, der die sinnvolle Nutzung des generierten Wissens nach dessen Interpretation vorsieht. Dies impliziert die nachvollziehbare Dokumentation der Erkenntnisse für alle involvierten Parteien oder deren Weiterleitung an andere relevante Bereiche. Zur Umsetzung der verständlichen Dokumentation kann eine Visualisierung sinnvoll sein, die sich häufig mit Darstellungen aus der Statistik wie Box-Plots durchführen lässt.

Der in Abbildung 3 von dem Prozessschritt Interpretation abgehende Pfeil verdeutlicht, dass der Prozess bei Bedarf beliebig oft durchlaufen werden kann. Sofern im Rahmen der Evaluation festgestellt wird, dass bei der Datenvorverarbeitung nicht alle redundanten Daten eliminiert wurden, wird zu diesem Schritt zurückgekehrt und der KDD-Prozess ab diesem Punkt erneut durchlaufen.

Nachdem das KDD-Vorgehensmodell von Fayyad beschrieben wurde, wird nun das vier Jahre später veröffentlichte CRISP-DM-Modell vorgestellt.

Cross Industry Standard Process for Data Mining

Das Cross Industry Standard Process for Data Mining (CRISP-DM) wurde im Jahr 2000 von Akteuren vier unterschiedlicher industrieller Unternehmen veröffentlicht und zielt darauf ab, den Lebenszyklus eines Data-Mining-Projektes abzubilden (Chapman et al. 2000). In ihrer Beschreibung des Vorgehensmodells erachten die Autoren das CRISP-DM-Modell als geeigneten Ansatz, um die Qualität von Data-Mining-Projekten zu gewährleisten (Chapman et al. 2000). Chapman et al. (2000) bewerten die Vorteile des Vorgehensmodells dahingehend, dass die erforderlichen Fähigkeiten des Anwenders im Bereich Data Mining weniger stark ausgeprägt sein müssten. Es wird angenommen, dass durch die Anwendung des Vorgehensmodells und die damit einhergehenden Erfahrungen eine Wiederverwendung des Modells in anderen Anwendungsfällen möglich ist. Dies begründet sich durch den verallgemeinerbaren Verwendungszweck, die dadurch erreichte Stabilität bei Anwendungsfällen aus verschiedenen Domänen, die Robustheit gegenüber Veränderungen in der Umgebung sowie die Möglichkeit, Tools einzubeziehen.

Chapman et al. (2000) geben mit dem Vorgehensmodell eine Empfehlung von Aktivitäten, die im Rahmen eines Data-Mining-Projektes durchgeführt werden können. Jede Aktivität setzt sich aus einer Reihe von Aufgaben zusammen. Mit den Pfeilverbindungen

sollen die wichtigsten Beziehungen und Abhängigkeiten zwischen den Aufgaben eines Data-Mining-Projektes symbolisiert werden, wobei die Verbindungen je nach Projekt beliebig verlaufen können. Der in Abbildung 4 dargestellte äußere Kreis symbolisiert die zyklische Eigenschaft von Data-Mining-Projekten. Diese Eigenschaft impliziert, dass ein Data-Mining-Projekt nach der Generierung eines Ergebnisses nicht als abgeschlossen betrachtet werden kann, da es kontinuierlich verbessert werden kann oder auch als Grundlage für andere Data-Mining-Projekte dienen kann. Die Ausführung von Datenvorverarbeitungsaufgaben beispielsweise erfolgt in der Praxis häufig in mehreren Iterationen während des Wissensentdeckungsprozesses (Mariscal et al. 2010). In diesem Kontext erwähnen Mariscal et al. (2010) außerdem, dass die Abfolge der Phasen des CRISP-DM-Lebenszyklus nicht als sequenziell zu betrachten ist, sondern ein Wechsel zwischen den einzelnen Phasen zu jedem Zeitpunkt möglich ist. In vielen Fällen ist nach der Auswahl des Modellierungswerkzeugs eine Rückkehr zum Schritt der Vorverarbeitung erforderlich, da die ausgewählten Modelle spezifische Anforderungen an die Trainingsdaten stellen. Das Modell besteht aus sechs iterativen Schritten, deren Beschreibung im Folgenden nach Chapman et al. (2000) erfolgt:

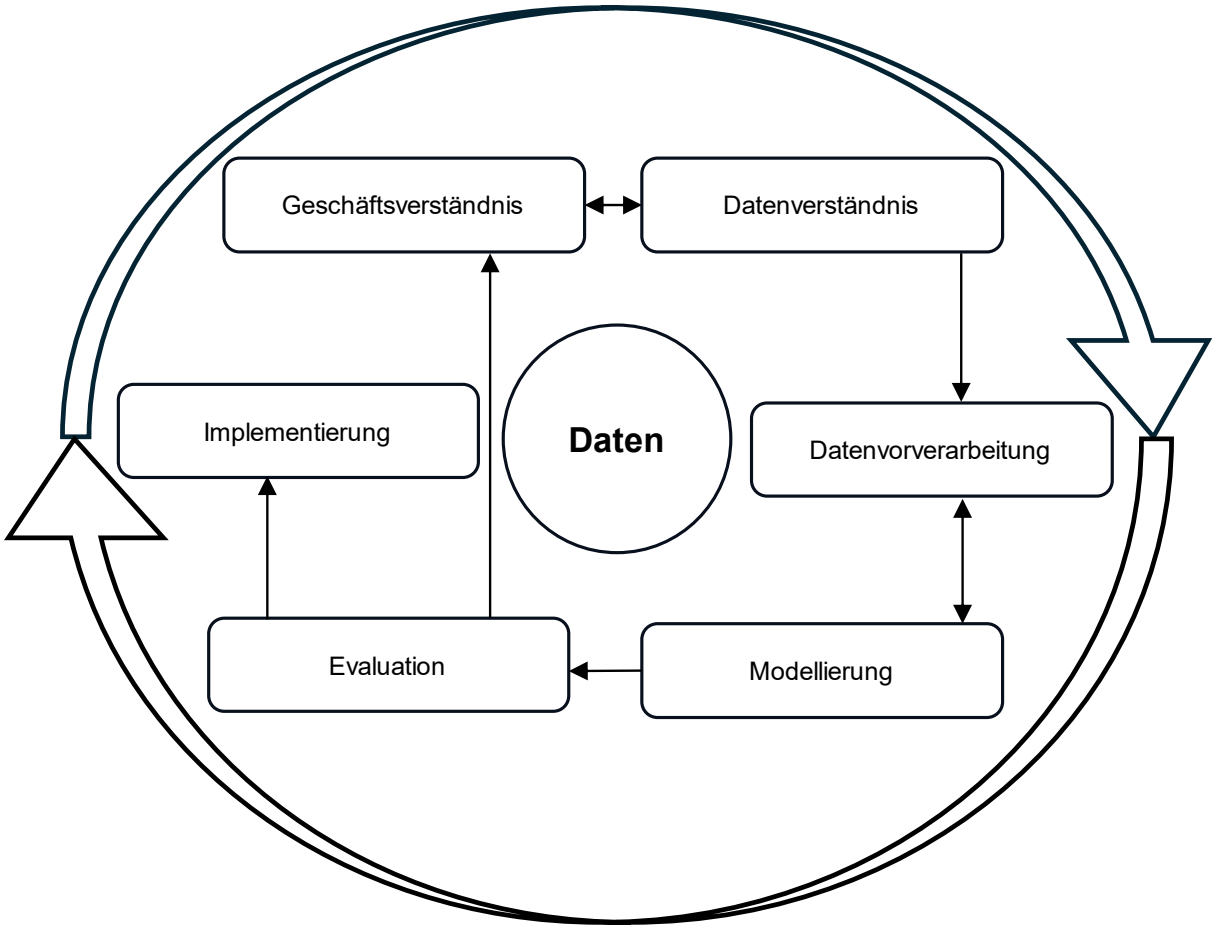


Abbildung 4: CRISP-DM-Vorgehensmodell (in Anlehnung an Chapman et al. 2000, S.13)

1. **Geschäftsverständnis:** Das Geschäftsverständnis umfasst die Bestimmung der Unternehmensziele aus geschäftlicher Perspektive sowie die Bewertung der Situation unter Berücksichtigung identifizierter Einschränkungen und Annahmen. Auf Basis der zuvor ermittelten Unternehmensziele sowie der Anforderungen aus Anwendersicht erfolgt die Entwicklung einer Data-Mining-Problemstellung sowie eines Planes zur Datenerfassung. Es ist von Vorteil, bereits vorhandenes Wissen über den Prozess zu nutzen und Experten unterschiedlicher Bereiche einzubeziehen, um eine Problemdefinition und eine vorläufige Planung zur Zielerfüllung zu entwickeln. Des Weiteren ist es erforderlich, bereits im Vorfeld Kriterien für ein zufriedenstellendes Projektergebnis aus geschäftlicher und technischer Sicht zu definieren.
2. **Datenverständnis:** Im Rahmen der Beschäftigung mit Datenerhebung erfolgt zunächst eine oberflächliche Untersuchung und Beschreibung der Daten, um sich einen Gesamtüberblick über die Datenbasis zu verschaffen. Diesbezüglich können etwaige Probleme bezüglich der Datenqualität oder auffällige Teilmengen in den Daten identifiziert sowie Hypothesen über potenzielle Informationen gebildet werden.
3. **Datenvorverarbeitung:** Im Rahmen der Hauptanalyse wird der finale Datensatz bereitgestellt, welcher durch unterschiedliche Aktivitäten und Techniken aus den ursprünglichen Rohdaten generiert wird. In diesem Kontext umfasst die Tätigkeit die Selektion relevanter Daten und Attribute, gegebenenfalls die Erstellung fehlender Daten sowie deren Transformation und Bereinigung. Der so erstellte Datensatz stellt das Inputmaterial für die gewählte Modellierungsmethode dar.
4. **Modellierung:** Im vierten Schritt erfolgt eine Auswahl und Anwendung diverser Data-Mining-Techniken unter Berücksichtigung einer Parameteroptimierung. Eine valide Evaluierung der Ergebnisse erfordert die vorherige Durchführung von Trainings- und Testphasen sowie die Definition klarer Kennzahlen und Benchmarks zur Bewertung.
5. **Evaluation:** Im fünften Schritt erfolgt die Interpretation der Ergebnisse des Data Mining sowie eine Bewertung des Beitrags zur Lösung des Zielproblems. Auf dieser Grundlage wird das Modell für den endgültigen Einsatz ausgewählt. Im Rahmen der Evaluierung ist insbesondere zu prüfen, inwiefern die einzelnen Schritte auf die Erreichung des Geschäftsziels ausgerichtet sind. In diesem Kontext wird darüber hinaus empfohlen, den gesamten bisherigen Prozess einer kritischen Prüfung zu unterziehen und eine Entscheidung über die Verwendung der durch Data Mining gewonnenen Erkenntnisse zu treffen.
6. **Implementierung:** Zuletzt wird das entdeckte Wissen verständliche dargestellt, angewendet und geplant, wie die Data-Mining-Ergebnisse in das Unternehmen zu implementieren sind.

Das gesamte Vorgehensmodell ist in einen methodischen Rahmen eingebettet, wobei die zugehörige Beschreibung Empfehlungen für die Durchführung ausgewählter Aufgaben zu den Prozessschritten liefert. Dazu werden den Prozessschritten generische Aufgaben zugeordnet, die zunächst in allgemeiner Form formuliert sind, um bei einer Vielzahl von Data-Mining-Problemstellungen Abhilfe zu schaffen. Sie sind dabei als Lösungsvorschlag für den gesamten Data-Mining-Prozess und unabhängig von der im Detail verwendeten Data-Mining-Anwendung zu betrachten. Die generischen Aufgaben sind wiederum mit spezifischeren Aufgaben verknüpft, welche die Ausführung von Aktionen in bestimmten, speziellen Situationen beschreiben. Des Weiteren beinhaltet das Handbuch detaillierte Beschreibungen potenzieller Data-Mining-Aufgaben sowie Empfehlungen für die Verwendung geeigneter Werkzeuge. CRISP-DM gehört laut Umfrage zu der häufigsten verwendeten Methode für Data-Mining-Projekte (Piatstsky 2014).

Im Laufe der Zeit haben sich zu den beiden beschriebenen Vorgehensmodellen aus der Anfangszeit des Themengebiets Wissensrepräsentation in Datenbanksystemen zahlreiche Erweiterungen entwickelt, die zeitrelevanten Themen gerecht werden wollen. So wurden beispielsweise ab dem Jahr 2006 Workshops zu einem CRSIP-DM 2.0-Vorgehensmodell abgehalten, in welchen auf neue Möglichkeiten durch weiterentwickelte Techniken für die Datenvorverarbeitung oder den Einbezug von Datenquellen wie Text- und Webdaten eingegangen wurde (Keith McCormick 2007). Im Rahmen der weiteren Optimierungen wurde nach Mariscal et al. (2010) der Einsatz des Vorgehensmodells in Echtzeitumgebungen in Erwägung gezogen. Dies setzt voraus, dass die relevanten Informationen direkt aus großen Datenbanken und nicht lediglich aus exportierten Datensätzen bezogen werden können. Ein weiterer Aspekt ist der nahtlose Übergang der Ergebnisse des Vorgehensmodells in den Geschäftsprozess (Mariscal et al. 2010).

Im Folgenden werden zwei weitere auf dem Vorgehensmodell von Fayyad oder CRISP-DM aufbauende Modelle – Knowledge Discovery in Industrial Databases (KDID) und Sample, Explore, Modify, Model, Asses (SEMMA) – kurz erläutert.

Knowledge Discovery in Industrial Databases

Das Knowledge Discovery in Industrial Databases (KDID) wurde von Lieber et al. (2013) entwickelt, um praxisrelevanten Problemstellungen in der Industrie zu begegnen. Das Referenzmodell von Fayyad et al. (1996) ist für bereits direkt verwendbare Datenbestände ausgelegt, wie sie in der Industrie üblicherweise nicht direkt vorkommen (Lieber et al. 2013a). Zur Lösung dieses Problems wird eine Erweiterung des bekannten KDD-Prozesses von Fayyad et al. (1996) vorgeschlagen. Diese Erweiterung umfasst

insbesondere die Phasen „Datensammlung" und das Einbeziehen von Expertenwissen und wird von Lieber et al. (2013) folgendermaßen beschrieben:

1. **Projektziele und Data-Mining-Aufgabenstellung definieren:** Der erste Schritt dient der Zieldefinition des Wissensentdeckungsprozesses und beinhaltet die Eingrenzung der in Frage kommenden Data-Mining-Verfahren je nach Aufgabenstellung.
2. **Ist-Zustand der IT-Struktur und des Expertenwissens aufnehmen:** Im zweiten Schritt gilt es sich Hintergrundwissen zum behandelten Thema durch Experten anzueignen und den IT-Hintergrund der Datenablage zu verstehen. Nach diesem Schritt ist der erste von zwei Meilensteinen erreicht.
3. **Vorstudie durchführen:** Die Durchführung der Vorstudie erfolgt anhand einer Stichprobe mit dem Ziel, vorab Auffälligkeiten in den Datenbeständen zu identifizieren und Zusammenhänge zu erkennen. Statistische Anwendungen in der Datenuntersuchung wie Verteilungen oder Korrelationen dienen dabei der Bewertung des zu erwartenden Werts der Wissensdurchführung auf den Datenbestand.
4. **Datensammlung:** Im Rahmen der Datensammlung sollte der Fokus auf einer umfassenden Datenbeschaffung in konsistenter und vollständiger Form sowie einer strukturierten Speicherung der Daten liegen. Diesbezüglich sind die „operativen und technischen Voraussetzungen für die Sammlung, Speicherung, Aufbereitung und Abfrage von Prozess- und Planungsdaten" zu schaffen.
    i. Integration der Daten aus IT-Systemen durchführen
    ii. Daten erfassen und speichern
5. **Datenvorverarbeitung:** Im vierten Schritt werden verschiedene Techniken angewendet, um die Qualität des Datensatzes auf ein möglichst hohes Niveau zu bringen. Dabei sollte insbesondere auf die Richtigkeit und Vollständigkeit der Daten geachtet werden. Zudem werden die Daten in der Form transformiert, in der sie als Grundlage für Data-Mining-Algorithmen fungieren können.
6. **Data-Mining-Modell erstellen:** Im Rahmen der Modellentwicklung ist es erforderlich, verschiedene Modellausprägungen zu erstellen und diese anhand von Kriterien wie Robustheit, Genauigkeit, Allgemeingültigkeit und Prognosekraft zu vergleichen.
7. **Ergebnisse hinsichtlich Zielerreichung interpretieren:** In diesem Zusammenhang ist es erforderlich, erneut Datenverarbeitungsschritte einzubeziehen, um eine sinnvolle Rückführung in den Produktions- und Planungsprozess zu gewährleisten.
8. **Gewonnenes Wissen integrieren:** Es wird empfohlen, das Ereignis für alle beteiligten Parteien in einer visuell verständlichen Darstellung zu präsentieren. Mit Schritt sieben wird der zweite Meilenstein des Prozesses erreicht.

9. **IT-Prototyp zur Wissensentdeckung und -einsatz erstellen:** Die Entwicklung eines IT-basierten Entscheidungstools, welches sowohl Data-Mining-Modelle als auch die Wissensrückführung in den Planungs- und Entscheidungsprozess integriert, stellt eine sinnvolle Ergänzung des Prozesses dar.

Lieber et al. (2014) heben im Zuge ihres Modells insbesondere die Experteneinbindung hervor. Diese lässt sich durch Data-Mining-Algorithmen nicht ersetzen, stattdessen kann Data Mining als Mittel zur Unterstützung von Entscheidungen betrachtet werden. Ein weiterer Schritt, der hervorgehoben wird, ist die Datensammlung. In der industriellen Anwendung besteht das Problem, dass die Datenbestände häufig heterogen, unvollständig und inkonsistent sind. Dies ist auf die über die Jahre entstandene, ungeordnete IT-Landschaft zurückzuführen (Deuse et al. 2014).

Nachdem in der vorherigen Betrachtung das KDID-Vorgehensmodell vorgestellt wurde, welches auf dem KDD-Vorgehensmodell von Fayyad et al. (1996) aufbaut, soll im Folgenden das ASUM-Modell erörtert werden, welches auf dem CRISP-DM-Vorgehensmodell basiert.

Sample, Explore, Modify, Model, Asses

Das Vorgehensmodell „Sample, Explore, Modify, Model, Assess“ (SEMMA) wurde von dem Unternhemen SAS entwickelt (SAS 2017). Azevedo und Santos (2008) konstatieren, dass das SEMMA-Vorgehensmodell im Gegensatz zum CRISP-DM, welches einen umfassenden Überblick über den gesamten Data-Mining-Prozess bietet, stärker auf die Modellierungsphase fokussiert. Das Vorgehensmodell CRISP-DM umfasst sechs Phasen, darunter auch die Phasen des Geschäfts- und Datenverständnisses. Das Modell SEMMA hingegen legt den Schwerpunkt auf fünf Phasen, welche das Arbeiten mit den Daten und die Modellerstellung betreffen (SAS 2017):

1. Sample: Im ersten Schritt wird zunächst eine repräsentative Stichprobe der Daten gezogen, um eine valide Modellierung zu ermöglichen.
2. Explore: Im Rahmen der Datenanalyse werden die Daten zunächst einer explorativen Untersuchung unterzogen, um Muster, Anomalien und wesentliche Einflussgrößen zu identifizieren.
3. Modifiy: Die Daten werden modifiziert, wobei verschiedene Methoden zum Einsatz kommen. Dazu zählen beispielsweise die Bereinigung, die Transformation sowie die Feature-Erstellung.
4. Model: Im vierten Schritt erfolgt die Erstellung eines Modells, welches das gewünschte Ergebnis prognostiziert. Grundlage hierfür sind die modifizierten Daten.

5. Assess: Abschließend erfolgt eine Evaluierung des Modells, wobei insbesondere die Güte der Vorhersage bewertet wird. Sofern erforderlich, wird das Modell im Rahmen dieses Schritts angepasst.

Während SEMMA weniger Fokus auf die Geschäftsziele legt und dafür stärker auf die praktische Modellierung eingeht, ist CRISP-DM umfassender und flexibler, da es sich auf den gesamten Data-Mining-Lebenszyklus konzentriert (Azevedo und Santos 2008).

In der wissenschaftlichen Literatur finden sich zahlreiche weitere KDD-Vorgehensmodelle. Für einen umfassenden Überblick verschiedener KDD-Vorgehensmodelle sei zum Beispiel auf die Ausarbeitung von Kurgan und Musilek 2006) hingewiesen. Aufgrund der hohen Frequentierung in der wissenschaftlichen Literatur wurden das Vorgehensmodell von Fayyad et al. (1996) und das von CRISP-DM ausgewählt und ein jeweils darauf aufbauendes Modell beschrieben. Im Folgenden erfolgt ein Vergleich sowie eine nähere Beschreibung der gemeinsamen Prozessschritte.

2.2.2 Vergleich der Vorgehensmodelle zur Wissensentdeckung in Datenbanken

Ein Vergleich zwischen den Phasen der bekanntesten beiden Vorgehensmodelle von Fayyad et al. (1996) und dem CRISP-DM wird in der Literatur unter anderem von (Mariscal et al. 2010) vorgenommen. Im Rahmen derer Untersuchung wird festgestellt, dass das CRISP-DM sechs und das Vorgehensmodell von Fayyad et al. (1996) unter Einbezug der Zwischenschritte neun Prozessschritte umfasst. In der Literatur wird darauf verwiesen, dass der Datenvorbereitungsschritt des CRISP-DM-Vorgehensmodells bei Fayyad et al. (1996) in die beiden Prozessschritte Datenbereinigung und -vorverarbeitung sowie Datenreduktion und -projektion aufgeteilt wird. Der Modellierungsschritt des CRISP-DM besteht gemäß Fayyad et al. (1996) aus drei Phasen: der Auswahl der Data-Mining-Funktion, der Auswahl des Data-Mining-Algorithmus sowie dem eigentlichen Data-Mining. Obschon die einzelnen Phasen bei Fayyad et al. (1996) detaillierter dargestellt sind, werden auch die Phasen des CRISP-DM-Vorgehensmodells wie erwähnt durch generische Aufgaben und das Nutzerhandbuch näher beschrieben. Des Weiteren berücksichtigt das CRISP-DM-Vorgehensmodell zu Beginn des Prozesses das Geschäftsverständnis und am Ende die Implementierungsmöglichkeiten der Ergebnisse. Diese Aspekte werden von Fayyad et al. (1996) zwar auch erwähnt, stehen aber nicht weiter im Fokus (Rotondo und Quilligan 2020). Des Weiteren ist zu konstatieren, dass sich die Ausgangslage beider Modelle unterscheidet. Das CRISP-DM-Vorgehensmodell thematisiert die Datenerfassung, während das Vorgehensmodell von Fayyad et al. (1996) von einer bereits vorhandenen Datenbasis ausgeht (Azevedo und Santos 2008).

Das übergeordnete Ziel jedes KDD-Vorgehensmodells besteht in der Generierung von Wissen aus Daten (Maimon und Rokach 2005). Dafür werden die Vorgehensmodelle durch Einflüsse aus diversen Bereichen, darunter Datenbanken, Statistik, Mathematik, Logik und künstliche Intelligenz geprägt, die dazu beitragen, Datenanalyse und Wissensextraktion aus unterschiedlichen Perspektiven zu beleuchten (Mariscal et al. 2010). Obgleich sich die gängigen KDD-Vorgehensmodelle wie beschrieben in einigen Zwischenstufen voneinander unterscheiden und einen jeweils unterschiedlichen Fokus aufweisen, beispielsweise hinsichtlich des Geschäftsverständnisses oder des Datenverständnisses, lassen sich grundlegende übergeordnete Prozessschritte identifizieren, die in den einzelnen Modellen mehr oder weniger detailliert behandelt werden, teilweise verschieden benannt sind oder in ihren Unterschritten einen anderen Fokus haben.

In der vorverarbeitenden Phase werden die Zieldefinition und Datenvorverarbeitung als Schnittstelle identifiziert (Scheidler 2017). Dem Schritt der Datenvorverarbeitung wird mit ca. 80% laut Gabriel et al. (2011) der gesamten zeitlichen, technischen und personellen Ressourcen der größte Aufwand beigepflegt. Die Datenvorbereitung zielt darauf ab, die Qualität des Datenbestandes durch verschiedene Techniken auf ein Niveau zu heben, das eine aussagekräftige Auswertung ermöglicht (Gabriel et al. 2011). Die Vorbereitung der Daten ist eine grundlegende Voraussetzung für die Anwendung von Data-Mining-Algorithmen, da nur auf diese Weise eine Grundlage für die Erkennung von Mustern geschaffen wird (Chapman et al. 2000). Zu den vorbereitenden Maßnahmen zählt die Eliminierung von Inkonsistenzen und Datenrauschen sowie der Umgang mit fehlenden Werten. In diesem Kontext steht eine Vielzahl von Imputationsmethoden zur Verfügung, die darauf abzielen, Datenlücken auf systematische Weise zu schließen. Dazu werden z.B. Methoden wie das Ersetzen der Werte durch Mittelwerte oder Schätzungen angewandt (Pyle 2007). Eine weitere Methode, die unter den Oberbegriff der Datenvorverarbeitung fällt, ist die Datenreduktion. Die Datenreduktion kann zu einer Verbesserung der Ergebnisse von Data-Mining-Algorithmen führen, da sie durch einen verringerten Datensatz spezifischere Analysen ermöglicht (Pyle 2007). Einige Data-Mining-Algorithmen liefern nämlich bei einer hohen Datendimensionalität keine eindeutigen Ergebnisse, die sich auf den gesamten Datensatz anwenden lassen und die Analyse großer Datensätze beansprucht häufig viel Zeit. Die Reduktion der Datengröße kann folglich als wesentlicher Schritt bei der Datenvorverarbeitung betrachtet werden, da sie die Effizienz von Data-Mining-Algorithmen verbessert, eine genauere Analyse der großen Datensätze ermöglicht und durch den verringerten Bedarf an Speicherplatz und Rechenleistung zu Kosteneinsparungen führen kann (Bishop 2016; Leskovec et al. 2020). Zur Datenreduktion stehen Methoden wie die Dimensionsreduktion oder die Merkmalsauswahl

bereit, die sich damit beschäftigt irrelevante oder redundante Merkmale aus Datensätzen zu entfernen (Guyon und Elisseeff 2003).

Die Methodenanwendung in Form des Data Minings wird von Scheidler (2017) als weitere Gemeinsamkeit von KDD-Vorgehensmodellen identifiziert. Das Thema Data Mining kam erstmalig in der Mitte der 1960er Jahr in Zusammenhang mit der Statistik auf und wird seitdem als datenbasierte Analysemethode gesehen (Elder und Pregibon 1998). Fayyad et al. (1996) weisen darauf hin, dass bei einer ausreichend langen Suche Muster entdeckt werden können, die auf den ersten Blick aussagekräftig und neu zu sein scheinen, jedoch in Wahrheit keine neuen Erkenntnisse darstellen. Diese These wird durch die Behauptung unterstrichen, dass 70–80 % der entdeckten Zusammenhänge und Muster in den Daten keine Neuigkeit darstellen (Otte et al. 2004). In diesem Kontext erweist sich der KDD-Prozess, bei dem Data Mining lediglich einen Zwischenschritt darstellt, dessen Ergebnisse anschließend einer detaillierten Analyse und Einordnung unterzogen werden, als besonders hilfreich. Das Data-Mining-Verfahren selbst besteht ausgehend von der Zielsetzung aus einigen Grundmethoden (Fayyad et al. 1996a). Die klassischen Verfahren teilen sich in die Bereiche Klassifikation, Regression, zur Untersuchung von Zusammenhängen, Clusteranalyse, zur Einteilung von Objekten in Gruppen und Wirkzusammenhänge, zum Aufzeigen kausaler Beziehungen ein (Alpar und Niedereichholz 2000). Alpar und Niedereichholz (2000) führen aus, dass bei der Klassifikation Verfahren wie k-Nearest-Neighbourhood, Nearest Neighbourhood, Neuronale Netzwerke oder Entscheidungsbäume zum Einsatz kommen, um Objekte beispielsweise bei der Bilderkennung in Klassen einzuordnen. Prognosezielstellungen fallen unter den Aufgabenbereich Regression, wo mit Analyseverfahren Beziehungen zwischen einer abhängigen und einer oder mehreren unabhängigen Variablen untersucht wird. Im Gegensatz zur Klassifikation handelt es sich um eine stetige, anstatt diskrete abhängige Variable. Um Aufgabenstellungen aus diesem Bereich zu untersuchen, kommen Verfahren wie Lineare-, Poisson- oder linear gemischte Regression zum Einsatz. Die gefundenen Muster sollten möglichst robust sein (Dhar 2012).

In der Literatur findet sich eine Unterteilung der verschiedenen Data-Mining-Methoden in beschreibende und vorhersagende Aufgaben (Han et al. 2012). Die Bewältigung dieser Aufgaben erfolgt nach Han et al. (2012) unter Zuhilfenahme von Information Retrieval, statistischen oder maschinellen Lernmethoden. Die Anwendung von Data Mining unterliegt unabhängig von der gewählten Methode vier bestimmten Bedingungen (Cleve und Lämmel 2020):

1. Es sind ausreichend Daten zu dem betrachteten Thema vorhanden

2. Die Daten sind historisch und aktuell richtig und voraussichtlich auch für die Zukunft repräsentativ
3. Die Daten enthalten die Informationen, die aus dem Datensatz extrahiert werden sollen
4. Die Verwendung der Daten ist rechtlich genehmigt, auch in Bezug auf Datenschutz

Uneinigkeit herrscht in der Literatur in dem Punkt, ob Data-Mining hypothesenfrei anzuwenden ist. So gibt es die Meinung, dass der Data-Mining-Prozess ein hypothesenfreies Datenanalyseverfahren ist, bei dem im Vorfeld keine konkreten Annahmen und möglichen Ergebnisse antizipiert werden (Multhaupt 2000). Wie beim Vorgehensmodell von Fayyad et al. (1996) erwähnt, wird aber auch der Ansatz des explorativen Data Minings vertreten, das gezielt aufgestellte Hypothesen überprüft (Fayyad et al. 1996b).

Ein weiterer übergreifender Prozessschritt, der in den etablierten KDD-Vorgehensmodellen von Scheidler (2017) identifiziert wird, ist die Ergebnisevaluation. Dieser Schritt umfasst die Auswertung und Interpretation der erzielten Ergebnisse. Im Rahmen dessen werden durch den Menschen aus den Ergebnissen Erkenntnisse abgeleitet (Kohlhammer et al. 2013).

Neben den herausgestellten Prozessschritten als Gemeinsamkeit der KDD-Vorgehensmodelle, ist zudem eine mehr oder weniger ausgeprägte Iterativität der Vorgehensmodelle zu beobachten (Scheidler 2017). Diese wird insbesondere bei CRISP-DM durch die Pfeilverbindungen zwischen den Schritten hervorgehoben oder von Fayyad et al. (1996) in Veröffentlichungen beschrieben oder im Prozessmodell ebenfalls durch Pfeilverbindungen dargestellt. Dies eröffnet die Möglichkeit einer kontinuierlichen Anpassung der Prozesse, stößt jedoch auch auf Kritik, dass die Reihenfolge der Prozessschritte nicht klar definiert sei. Der Grund hierfür liegt in der Möglichkeit, von jeder Phase in jede beliebige andere Phase zu springen, wodurch klar definierte Iterationen und Wechselwirkungen fehlen (Rotondo und Quilligan 2020).

In der industriellen Datenanalyse wird wie bereits erwähnt das CRISP-DM-Vorgehensmodell auf häufigsten eingesetzt. Für das Vorgehensmodell spricht dessen klar definierte Projektstruktur, die eine intuitive Anwendung ermöglicht und sich unabhängig von der jeweiligen Branche einsetzen lässt (Deuse et al. 2024).

2.3 Maschinelles Lernen

Maschinelles Lernen ist eine Anwendungsform der künstlichen Intelligenz (AI) und umfasst die Wissenschaft, automatisiert Entscheidungen auf Grundlage von Daten zu treffen, um Wissen zu generieren (Russell 2010). Beim maschinellen Lernen lernt ein

Algorithmus durch Erfahrung dazu, ohne dass er für eine explizite Problemstellung programmiert worden ist. Maschinelle Lernverfahren trainieren sich anhand von Daten ein möglichst optimales Verhalten an, ohne dass spezifische Einzelfälle explizit programmiert werden müssen (Murphy 2013). Im Gegensatz zur klassischen Programmierung, bei der ein Programm, wie in Abbildung 5 dargestellt, auf Basis von Daten als Input einen Output erzeugt, wird beim maschinellen Lernen das Programm als Ergebnis generiert. Dazu werden Trainingsdaten, gepaart mit anderen Daten als Grundlage zur Erstellung des Programms herangezogen (Murphy 2013).

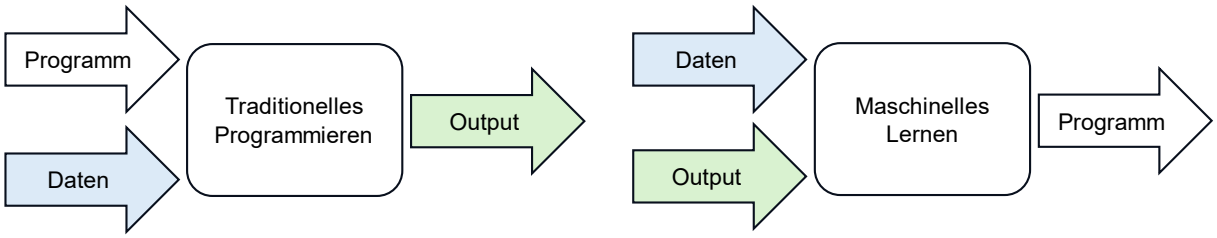


Abbildung 5: Vergleich Traditionelle Programmierung und maschinelles Lernen (in Anlehnung an Natras und Schmidt 2021)

Die Algorithmen des maschinellen Lernens verfolgen verschiedene Funktionsziele, sodass sich eine Gliederung in unterschiedliche Gruppen durchführen lässt. Die Hauptlernparadigmen umfassen das überwachte, unüberwachte, halbüberwachte und Verstärkungslernen, wovon die ersten beiden Klassen nachfolgend erörtert werden (Murphy 2013).

Überwachtes maschinelles Lernen

Die Generierung eines verwertbaren Ergebnisses durch die eigenständige Entwicklung eines Programms setzt die Befolgung der fünf Grundschritte voraus, die im Rahmen einer Aufgabe des überwachten maschinellen Lernens klassischerweise befolgt werden.

1. Datensammlung
2. Datenvorverarbeitung (Bereinigung, Aufbereitung etc.)
3. Anpassung und Training des Modells
4. Evaluation der Leistung des Modells
5. Hyperparametertuning, um die Leistung des Modells zu steigern

Vor der Anwendung eines Modells für prognostische Zwecke ist nach Murphy (2013) eine Prüfung der korrekten Funktionsweise unerlässlich. Zu diesem Zweck erfolgt eine Unterteilung der für das Modell vorgesehenen Daten in Trainings- und Testdaten, um die Leistungsfähigkeit des Modells hinsichtlich der Bewältigung von Daten, die dem

Modell nicht bekannt sind, zu evaluieren. Das Training des Algorithmus erfolgt auf Basis der Trainingsdaten. Die Testdaten hingegen stellen Daten dar, welche dem Algorithmus zuvor nicht bekannt waren. Ihre Funktion besteht in der Bewertung der Leistungsfähigkeit des Modells, wobei die Ergebnisse mit jenen der Trainingsdaten als Input verglichen werden. Die Bewertung erfolgt anhand von Kennzahlen, beispielsweise dem in Kapitel 2.1.4 beschriebenen mittleren quadratischen Fehler. Murphy (2013) führt weiter aus, dass die Leistung eines Vorhersagemodells stark davon abhängt, wie die Daten in die Trainings- und Testgruppen aufgeteilt werden. Um eine präzisere Einschätzung der Güte des Modells zu erhalten, können Methoden der Kreuzvalidierung angewendet werden. Diese teilen die Daten mehrfach in Trainings- und Testdatensätze auf, die unabhängig voneinander in das Modell gegeben werden. Anschließend werden die Ergebnisse gemittelt, um eine genauere Schätzung der Modelleistung zu erhalten (Müller und Guido 2017). Die Kreuzvalidierung hilft zudem gegen Überanpassung eines Modells. Überanpassung schränkt die Vorhersagekraft eines Modells ein, da es beispielsweise durch eine begrenzte Menge an Daten zu gut an die Trainingsdaten angepasst ist und dadurch die Generalisierungsfähigkeit gering ist (Fayyad et al. 1996c). Enthalten die Trainingsdaten beispielsweise Rauschen, zufällige Fehler, sowie Muster, die nicht den eigentlichen, realen Bedingungen entsprechen, basiert das trainierte Modell auf diesen Daten von geringer Güte. Infolgedessen ist die Zuverlässigkeit des Modells bei der Anwendung auf ungesehene Daten eingeschränkt. Eine weiterer Schritt zur Verbesserung der Leistungsfähigkeit von maschinellen Lernmodellen ist das Feature Engineering, bei dem relevante Daten extrahiert werden, um sie komprimiert in einer geeigneten Form für die Modellierung bereitzustellen (Guyon und Elisseeff 2006). Die erstellten Features können auf die Erfassung nichtlinearer Zusammenhänge, die Vermeidung von Überanpassung, eine verbesserte Generalisierbarkeit oder ähnliches abzielen (Guyon und Elisseeff 2006).

Beim überwachten Lernen erfolgt typischerweise eine Verarbeitung von Daten, die bereits eine strukturierte Form aufweisen (Murphy 2013). Die Zuordnung eines einzelnen Datenpunkts zu einer bestimmten Kategorie erfolgt mittels entsprechender Labels. Murphy (2013) beschreibt weiter, dass die Zielvariable, auch als abhängige Variable bezeichnet, durch andere unabhängige Variablen, wie relevanten Merkmalen oder Vorhersagevariablen, prognostiziert werden soll. In den Bereich des überwachten maschinellen Lernens lassen sich Regressionsmodelle einordnen, bei denen die Zielvariable immer eine kontinuierliche Variable ist. Zur anderen Gruppe dieses Bereiches zählen Klassifizierungsmodelle, bei denen die Zielvariablen kategorisch ist und die für Zielvorhersagen wie beispielsweise baldiger Ausfall eines Maschinenbauteils oder Nicht-Ausfall des Bauteils eingesetzt werden. Einer der etablierten Klassifizierungsalgorithmen im Bereich

überwachtes Lernen ist der k-Nearest Neighbors-Algorithmus (KNN). Er basiert darauf, dass ähnliche Objekte hinsichtlich eines betrachteten Merkmales in der grafischen Darstellung nah beieinander liegen. Der Algorithmus findet die k nächstgelegenen Nachbarn zu einem Datenpunkt und sagt auf Grundlage der am häufigsten vertretenen Merkmalsausprägungen der nahegelegenen Objekte einen neuen Datenpunkt voraus (Hastie et al. 2009). Dem neuen Datenpunkt wird also die Merkmalsausprägung zugeschrieben, die bei seinen k betrachteten Nachbarn am häufigsten vorkommt. Die k nächstgelegenen Datenpunkte werden durch Abstandsfunktionen bestimmt.

Falls die Entscheidungsgrenze bei Klassifizierungsproblemen nicht linear ist, bietet es sich an, den Support Vector Maschine-Algorithmus (SVM) einzusetzen. In diesem Verfahren werden die Datenpunkte durch eine Hyperebene in zwei Klassen getrennt. Dabei hat die gefundene Hyperebene den größtmöglichen Abstand zwischen Datenpunkten verschiedener Klassen. Die zu den Hyperebene nächstgelegenen Punkte der relevanten Klassen werden als Support-Vektoren bezeichnet. Das Ziel besteht in der Setzung der Hyperebene in einer Weise, dass der Abstand zwischen den Support-Vektoren maximiert wird (Hastie et al. 2009). Wenn betrachtete Daten also nicht linear trennbar sind, können die Datenpunkte im Rahmen von SVM durch Kernel-Funktionen in einen höher dimensionierten Raum transformiert werden, in dem dann eine lineare Trennung möglich ist. Ein weiterer Algorithmus, der sowohl bei Regressionsaufgaben als auch bei Klassifizierungen eingesetzt wird, ist der Entscheidungsbaum. Diese grafisch klar definierte Darstellungsform besteht aus einer Wurzel, übergeordneten und untergeordneten Knoten, Blättern und einer Entscheidungsregelung an jedem Zweig. Im Rahmen von Klassifizierungsaufgaben ermöglicht der Entscheidungsbaum durch die Definition einer Entscheidungsregel eine Reduktion des Unreinheitsmaßes an den einzelnen Knoten. Dieses sogenannte Gini-Maß dient der Beschreibung der Güte der Trennung der betrachteten Elemente verschiedener Klassen. Ein niedriger Gini-Wert indiziert eine homogene Struktur des Knotens sowie eine verlässliche Vorhersage. Im Bereich der Regression besteht das Ziel, den MSE an jedem Knoten zu reduzieren, um eine möglichst aussagekräftige Vorhersage zu erhalten (Albon und Gallatin 2023). Um potenziellen Problemen, wie einer hohen Varianz, die bei der Entscheidungsbaummethode auftreten können, entgegenzuwirken, wurde mit Random Forest ein weiterführendes Modell entwickelt. Dieses Modell soll eine verbesserte Genauigkeit und Stabilität bei Vorhersagen gewährleisten (Murphy 2013). Dabei ist ein Random-Forest-Algorithmus eher robuster gegenüber hochdimensionalen Daten als ein Entscheidungsbaum. Dies ist darauf zurückzuführen, dass jeder der genutzten einzelnen Entscheidungsbäume eine zufällige Auswahl von Features verwendet. In der Konsequenz erfolgt eine Reduktion der Dimensionalität

der betrachteten Daten, wobei die zufällige Auswahl eine gegen Überanpassung wirkende Komponente darstellt (Murphy 2013).

Ein gängiges Maß zur Bewertung der Vorhersageleistung bei Klassifizierungsmodellen ist beispielsweise der F1-Score (Powers 2008). Powers (2008) beschreibt, dass der F1-Score insbesondere bei Datensätzen Anwendung findet, bei denen die Klassen unausgewogen vertreten sind. Er stellt einen harmonischen Mittelwert zwischen der Genauigkeit und Empfindlichkeit dar. Die Genauigkeit berücksichtigt dabei die präzise Vorhersage von Maschinenfehlern, während die Empfindlichkeit die Frage beantwortet, ob alle relevanten Fehler erkannt werden. Ein weiteres nützliches Instrument zur Evaluierung der Leistungsfähigkeit eines Klassifikationsmodells stellt die sogenannte Confusion Matrix dar. Die Confusion Matrix stellt die tatsächlichen und vorhergesagten Klassifikationen einander gegenüber und demonstriert die Güte des Modells bei der Differenzierung der Klassen.

Unüberwachtes maschinelles Lernen

Im Gegensatz zu überwachten Lernverfahren arbeiten Algorithmen des unüberwachten Lernverfahrens mit nicht gelabelten Datenpunkten (Murphy 2013). Murphy (2013) beschreibt das Ziel dieser Lernverfahren in der Erkennung von Mustern und Strukturen in diesen Datenpunkten. Das Resultat der Anwendung der Algorithmen ist nicht vorhersehbar und es existiert keine übergeordnete Instanz, die den Algorithmus zu Trainingszwecken beeinflusst, wie dies beim überwachten Lernen der Fall ist. Ein Verfahren des unüberwachten Lernens ist die Clusteranalyse. Im Gegensatz zu Klassifizierungsmethoden, bei denen vorab Gruppen mit Merkmalen definiert werden, erfolgt die Unterteilung einer Menge von Objekten in viele Teilmengen durch Cluster-Algorithmen selbstständig. Innerhalb der Teilmengen sollten die Objekte möglichst ähnlich und gleichzeitig möglichst unterschiedlich zu Objekten anderer Cluster sein. Zu den Cluster-Algorithmen zählt auch K-Means, dessen Ziel es ist, durch einen iterativen Prozess die Mittelpunkte in den einzelnen Clustern zu identifizieren. Diese Methode erweist sich jedoch als weniger zuverlässig, wenn die Cluster komplexe Formstrukturen aufweisen oder die einzelnen Datenpunkte nicht eindeutig voneinander zu trennen sind. Des Weiteren ist es erforderlich, die Anzahl der Cluster zu Beginn auszuwählen. Die optimale Anzahl von Clustern kann mit verschiedenen eigenständigen Methoden festgelegt werden, wobei die jeweilige Vorgehensweise von der konkreten Aufgabenstellung im Rahmen des maschinellen Lernens abhängt (Fahrmeir und Brachinger 1996).

Die Qualität und Aussagekraft der Ergebnisse sowohl unüberwachter als auch überwachter Algorithmen des maschinellen Lernens sind maßgeblich von den Informationen

abhängig, die zum Trainieren des Algorithmus bereitgestellt werden (Murphy 2013). In diesem Zusammenhang kommt dem Feature Engineering zur Repräsentation der zugrunde liegenden Daten eine entscheidende Bedeutung zu (Dong und Liu 2017). Die als Features bezeichneten Attribute oder Variablen werden aus vorliegenden Daten abgeleitet, um ein individuelles Datenobjekt zu beschreiben. Dong und Liu (2017) weisen darauf hin, dass die verschiedenen Möglichkeiten im Rahmen des Feature Engineerings von den verfügbaren Daten und der Zielstellung im Domänenbereich abhängig sind. Des Weiteren besteht die Möglichkeit, aus bereits bestehenden Features neue zu kreieren oder anzupassen (Feature Transformation), um auf diese Weise neue, sinnvolle Features aus den betrachteten Daten zu generieren oder zu selektieren, sodass sie von maschinellen Lernalgorithmus gut interpretiert werden können (Zheng und Casari 2018). Ein adäquates Feature Engineering im Kontext des maschinellen Lernens kann zu einer Verbesserung diverser Modelle, wie beispielsweise Vorhersagealgorithmen, beitragen, da es durch die komprimierte Informationsbereitstellung in Form eines Features gut an die Problemstellung angepasst wird (Zheng und Casari 2018). Dies ist darauf zurückzuführen, dass es zum Erlernen robusterer Muster sowie zur Vermeidung von Überanpassung förderlich ist.

Nachdem die Grundlagen zur Wissensentdeckung in Datenbanken sowie das Thema maschinelles Lernen erörtert wurden, erfolgt im nachfolgenden Kapitel eine vertiefende Auseinandersetzung mit dem Anwendungsgebiet der prädiktiven Wartung. Ziel ist es, diese drei Themenbereiche im Anschluss miteinander zu verknüpfen.

2.4 Einordnung der prädiktiven Wartung

Die prädiktive Wartung stellt ein anwendungsbezogenes Themenfeld der Datenanalyse dar, das im Rahmen der Industrie 4.0 entwickelt wurde. Es verbindet Elemente aus den Bereichen Datenanalyse, maschinelles Lernen, Big Data und Datenbankmanagementsysteme mit der klassischen Instandhaltung (Lu 2017). In diesem Kapitel erfolgt zunächst eine Einordnung des Begriffs Wartung in Bezug auf die Instandhaltung. Im Anschluss werden Wartungsmethoden beschrieben, die sich im Zuge des technischen Fortschritts weiterentwickelt haben.

2.4.1 Überblick über Instandhaltungsstrategien

Gemäß DIN 31051 (S.4) bezeichnet der Begriff der Instandhaltung:

„Die Kombination aller technischen und administrativen Maßnahmen während des Lebenszyklus einer Betrachtungseinheit zur Feststellung und Beurteilung des Ist-

Zustandes sowie zur Erhaltung des funktionsfähigen Zustandes oder der Rückführung in diesen“

Im produktionsspezifischen Kontext umfasst die Instandhaltung, wie in Abbildung 6 dargestellt, das Management von technischen und verwaltungstechnischen Maßnahmen zur Durchführung von Inspektionsaufgaben, Wartungs- und Instandsetzungsarbeiten sowie die Verbesserung von Maschinen und Anlagen (Mühlnickel et al. 2018).

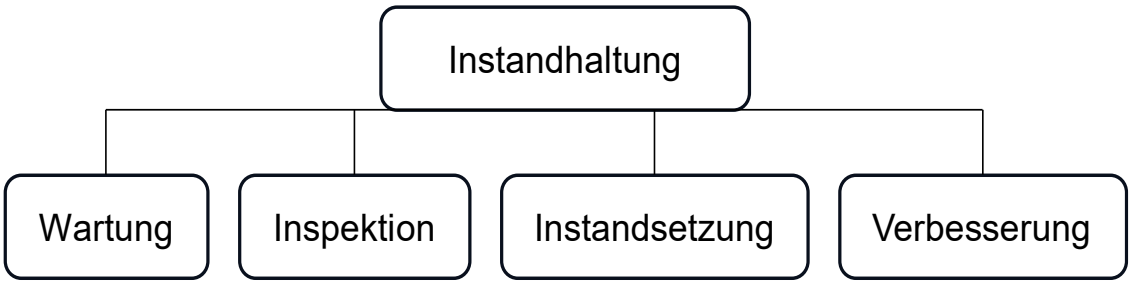


Abbildung 6: Unterteilung der Instandhaltung (in Anlehnung an DIN 31051 / 2012-09, S.2)

Damit ist die Wartung neben der Inspektion und Überholung von Systemen ein Teilbereich des Oberthemas Instandhaltung. Mühlnickel et al. (2018) beschreiben, dass die genannten Teilbereiche darauf abzielen, durch repetitive Anwendung das übergeordnete Ziel der Sicherstellung der Funktionsfähigkeit von Maschinen und Anlagen zu erreichen. Dies dient der Vermeidung von Störungen und Ausfallzeiten in der Produktion sowie der Lieferung einer zuverlässigen und kosteneffizienten Produktion. Jeder Stillstand und Ausfall einer Maschine in Produktionslinien verursacht nämlich durch Verfügbarkeitsverluste wirtschaftliche Nachteile für ein Unternehmen (Mehmeti et al. 2018). Um die vollumfängliche Funktionsfähigkeit zu gewährleisten, werden die Anlagen über ihren gesamten Lebenszyklus instandhaltungstechnisch begleitet. Ein gängiges Instrument zur Evaluierung der Ausfallwahrscheinlichkeit technischer Komponenten über den gesamten Lebenszyklus ist die in Abbildung 7 dargestellte Badewannenkurve.

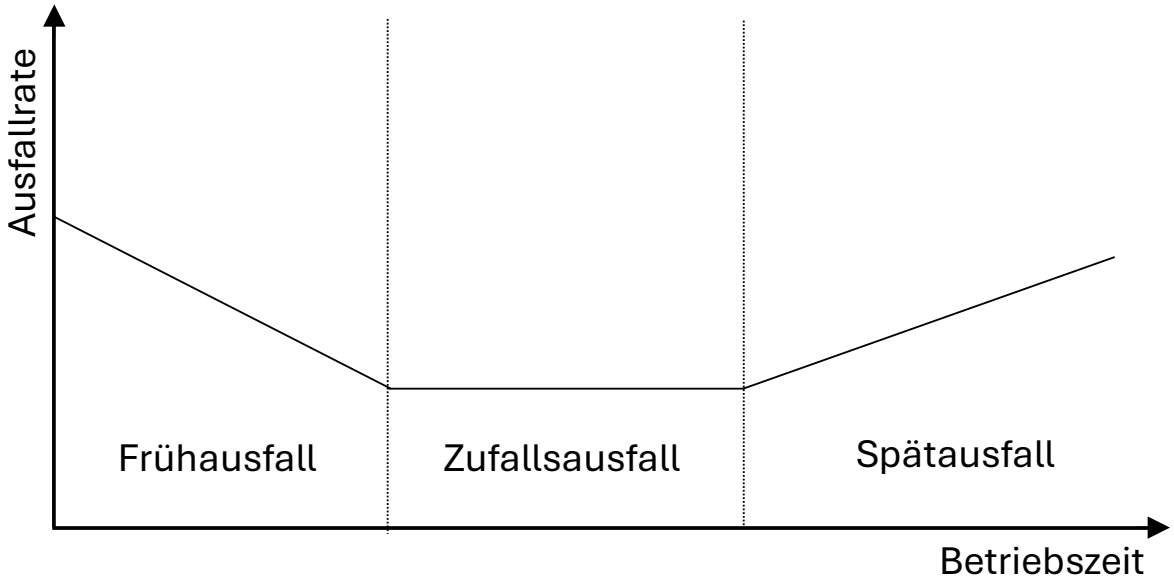


Abbildung 7: Badewannenkurve (in Anlehnung an Mühlnickel et al. 2018)

Diese zeigt, dass Ausfälle von Maschinenkomponenten zwar in Abhängigkeit der bereits absolvierten Betriebszeit unterschiedlich häufig auftreten, ein Ausfall jedoch grundsätzlich in jeder Lebenszyklusphase erfolgen kann (Hodapp 2018). Die Badewannenkurve veranschaulicht, dass Ausfälle in der frühen Lebensdauer eines Bauelements durchschnittlich am häufigsten auftreten. Dies ist laut Hodapp (2018) häufig auf vorausgegangene Fehler bei der Projektierung, Konstruktion, Fertigung, Montage des Elements zurückzuführen. Im weiteren Zeitverlauf treten Ausfälle eher zufällig und mit einer durchschnittlich geringen Häufigkeit auf, ehe es in der Spätausfallphase durch alterungsbedingten Verschleiß wieder zu einem Anstieg der Ausfallswahrscheinlichkeit kommt. In Abhängigkeit von den verfügbaren finanziellen Ressourcen, der technischen Ausstattung, der Kritikalität einer Komponente sowie der jeweiligen Lebenszyklusphase von Maschinenteilen werden unterschiedliche Instandhaltungsstrategien angewendet, die im Folgenden näher erläutert werden. In der Gesamtschau lässt sich konstatieren, dass sich die Möglichkeiten der Instandhaltungsanwendung von einer reaktiven Maßnahme zur Behebung von Störungen hin zu einer proaktiven Planungsdisziplin, bedingt durch technischen Fortschritt, entwickelt haben. Diese unterstützt produzierende Unternehmen dabei, Kosten einzusparen (Porter und Heppelmann 2015). Die verschiedenen Abstufungen der Wartungsmethoden werden im Folgenden im Rahmen der unterschiedlichen Instandhaltungsmethoden aufgegriffen.

Reaktive Instandhaltung

Die reaktive Instandhaltungsstrategie kann als ursprüngliche Instandhaltungsstrategie bezeichnet werden, die vor der Verfügbarkeit von gesammelten Produktions- und Maschinendaten etabliert war (Mühlnickel et al. 2018). Diese Strategie verfolgt keine vorher geplanten Instandhaltungsmaßnahmen, sondern es wird erst nach dem Auftreten einer Störung reagiert, um das betrachtete Betriebsmittel wieder in seine ursprüngliche Funktionsform zu bringen. Das Eintreten von Fehlfunktionen wird bewusst akzeptiert und damit Verschleißreserven von Maschinenkomponenten möglichst lang ausgenutzt (Wöstmann et al. 2019). Dabei steht das schnellstmögliche Wiedererlangen der Funktionsfähigkeit im Vordergrund, während zukünftige Themen wie der Werterhalt der Anlage eher zweitrangig sind und damit möglicherweise Folgeschäden in Kauf genommen werden. Der Verzicht auf eine präventive Instandhaltungsplanung und -durchführung kann für ein Unternehmen zwar auf der einen Seite Kosten einsparen, auf der anderen Seite kann es jedoch durch unvorhersehbare Maschinenausfälle zu einem Ausfall der gesamten Produktion und somit zu einem finanziellen Verlust für das Unternehmen kommen (Mühlnickel et al. 2018). Des Weiteren sind die Instandhalter einem hohen Druck ausgesetzt, da Ausfälle spontan auftreten und möglichst schnell behoben werden müssen, um weitere finanzielle Verluste durch Stillstandszeiten zu vermeiden. Dies kann insbesondere bei Komponenten der Fall sein, die keine allzu hohe Wertigkeit im finanziellen und für den Gesamtproduktionsprozess haben.

Nachdem die reaktive Instandhaltung, bei der erst nach dem Auftreten eines Fehlers reagiert wird, erörtert wurde, soll im Folgenden die präventive Instandhaltung als vorausschauende Herangehensweise vorgestellt werden.

Präventive Instandhaltung

Eine weitere Instandhaltungsstrategie, die in DIN 31051 (S.5) Erwähnung findet und sich vor der Einführung der Sammlung von Maschinen- und Produktionsdaten etabliert hat, ist die präventive Instandhaltung, die auch als zeitbasierte Instandhaltung bezeichnet wird. Durch regelmäßige Wartungsarbeiten wie beispielsweise Schmierung, Einstellung und Austausch von Teilen in festen Wartungszyklen soll die Ausfallwahrscheinlichkeit von Maschinen verringert werden. Die Instandhaltungsmaßnahmen werden auf Basis einer Ausfallzeitanalyse getroffen, welche auf Grundlage von historischen Ausfallzeitdaten, Verbraucherangaben zur Maschine, Betriebszeit oder produzierten Teilen seit der letzten Wartung geschätzt wird (Lee et al. 2006). Dabei können Annahmen beispielsweise auch auf dem durchschnittlichen altersbedingten Ausfallverhalten basieren, welches durch die in Abbildung 7 gezeigte Badewannenkurve beschrieben wird. Eine

weitere statistisch zu ermittelnde Kennzahl ist die mittlere Zeit bis zum Ausfall, die auf Grundlage von vergangenen Ausfällen kalkuliert wird (Rosmaini und Shahrul 2012). Die zeitliche Entwicklung findet ebenfalls bei dem Weibull-Modell Berücksichtigung, mit dem die sich mit der Zeit ändernde Ausfallwahrscheinlichkeit von Komponenten berechnet wird (Ghodrati 2006).

Die Vorteile einer zeitbasierten Instandhaltungsstrategie liegen in einer potenziell geringeren Störungshäufigkeit durch regelmäßige Wartungs- und Inspektionsdurchführungen im Vergleich zur reaktiven Instandhaltung (Samon und Shalom 2023). Wöstmann et al. (2019) führen weiter aus, dass strategische Planungen eine Reduktion der Durchführungszeiten von Instandhaltungsmaßnahmen ermöglichen, da diese im Vorfeld berücksichtigt werden können. Im Gegensatz dazu werden Maßnahmen bei der reaktiven Instandhaltung häufig ad hoc durchgeführt. Als potenzieller Nachteil ist zu nennen, dass die präventive Wartung zu Mehrkosten für Planung und Durchführung führt. Des Weiteren besteht die Option, dass Bauteile ausgetauscht werden, obschon sie ihre Lebensdauer noch nicht erreicht haben und folglich die Verschleißreserven der Bauteile nicht vollständig ausgeschöpft werden (Wöstmann et al. 2019). Daher ist eine sorgfältige Auswahl der Wartungshäufigkeit erforderlich, um unnötige Kosten durch den Austausch eines noch intakten Bauteils sowie durch häufige Stillstandzeiten zu vermeiden. Für Unternehmen kann die Expertise von Fachkräften von großem Nutzen sein, da diese den Zustand der Komponenten durch Auswertung früherer Schadenfälle sowie durch die langfristige Arbeit mit den Maschinen einschätzen können (Rosmaini und Shahrul 2012). Diese Strategie kann beispielsweise bei Bauteilen Anwendung finden, deren Verschleiß gut prognostizierbar ist, oder bei gesetzlich vorgeschriebenen Vorschriften zum regelmäßigen Austausch von Bauteilen.

Zustandsabhängige Instandhaltung

Die zustandsabhängige Instandhaltung setzt auf Methoden zur Überwachung von Maschinenzuständen, um Produktionsausfallkosten möglichst gering zu halten (Mühlnickel et al. 2018; Rosmaini und Shahrul 2012; Hashemian und Bean 2011). Präventive Wartungsplanungen, beispielsweise basierend auf der Badewannenkurve aus Abbildung 7, sind in erster Linie intuitiv, auf Grundlage von Erfahrungen, anstatt wissenschaftlich belegt. Dies lässt sich dadurch erklären, dass nach gängigen Studien 80–85 % der Maschinenausfälle auf Zufallsfaktoren zurückzuführen sind und nicht auf das Alter von Bauteilen (Hashemian und Bean 2011). Betriebsbedingungen von betrachteten Maschinenkomponenten werden bei Ansätzen wie der Badewannenkurve als konstant gesehen, was somit nicht das reale Verhalten widerspiegelt. Im Rahmen der zustandsabhängigen

Instandhaltung erfolgt eine Auseinandersetzung mit den genannten Problemen. Dies erfolgt durch eine Bewertung des Zustandes von Komponenten, beispielsweise mittels Condition-Monitoring-Systemen (CMS). Hierbei werden durch Sensoren kritische Parameter wie Beschleunigung, Kraft, Vibration oder Temperatur kontinuierlich ermittelt, woraus ein Kennwert zur Zustandsbewertung gebildet wird (Mühlnickel et al. 2018). Die über Sensoren ermittelten Zustandskennwerte können je nach Bauteil und Zielstellung kontinuierlich oder periodisch gemessen werden (Rosmaini und Shahrul 2012). Rosmani und Shahrul (2012) führen weiter aus, dass auch manuelle Messtechniken mit zum Beispiel Schallemissionsgeräten zur Zustandsermittlung eingesetzt werden können. Die Interpretation der aufgenommenen Messwerte erfolgt durch den Vergleich mit einer vorher definierten Ausfallsgrenze. Wird ein kritischer Kennwert überschritten, qualifiziert sich das Bauteil für entsprechende Wartungsmaßnahmen. Die übergeordnete Zielsetzung der Strategie besteht in der Reduktion von Kosten, welche durch die Planung von Instandhaltungsmaßnahmen unter Berücksichtigung des jeweiligen Betriebszustandes, welcher durch Sensordaten beschrieben wird, erreicht werden soll (Gupta und Lawsirirat 2006). Dies ermöglicht Wartungsarbeiten deutlich effektiver, je nach Zustand der betrachteten Komponente durchzuführen.

Die Grundlage für die Zustandsbewertung bildet zunächst die Rohdatenbasis, welche die ursprünglichen und unverarbeiteten Originaldaten direkt von einer Quelle, wie beispielsweise einem Sensor, umfasst (van der Valk et al. 2020). Sensordaten bestehen meist aus Zeitreihensignalen, die in regelmäßigen Zeitabständen erfasst werden. Im Rahmen der Aufbereitung werden die Merkmale aus den Sensordaten extrahiert und beispielsweise durch das Hinzuziehen von Fehlerprotokollen mit Zusatzinformationen versehen (Wang et al. 2017). Nach Wang et al. (2017) wird dadurch eine bessere Einordnung ermöglicht und eine Eignung für überwachte maschinelle Lernalgorithmen gewährleistet. Schwankungen im elektronischen Signal, Umwelteinflüsse, eine nicht sachgemäße Justierung oder Abnutzung können dazu führen, dass die aufgenommenen Daten wie im Kapitel 2.1.4 beschrieben verrauscht oder gestört sind.

Zusammenfassend ermöglichen zustandsorientierte Instandhaltungsmethoden Aktionen auf Grundlage des tatsächlich erfassten Zustands einer Maschine oder Anlage. Darauf aufbauend können vorausschauende Ausfallprognosen erstellt werden, die zufällig auftretende Störungen kalkulierbarer machen (Hodapp 2018). Dieses Prognostizieren von Ausfällen fällt unter den Bereich der prädiktiven Instandhaltung, der im Folgenden beschrieben wird.

Prädiktive Instandhaltung

Im Rahmen der prädiktiven Instandhaltung werden mittels Datenanalyse und maschinellem Lernen Prognosen darüber erstellt, mit welcher Wahrscheinlichkeit betrachtete Maschinen oder Anlagen in Zukunft ausfallen werden (Jardine et al. 2006). Bezogen auf den maschinellen Lernbereich kommen dabei meistens überwachte Lernverfahren wie die Mehrklassenklassifikation, Regression und binären Klassifikation zum Einsatz (Esteban et al. 2022). Das Ziel der prädiktiven Wartung besteht in der kontinuierlichen Datensammlung und -auswertung zur Verbesserung des Produktionsprozesses durch präventive Vorbeugung von Störungen durch Vorhersagen (Henke et al. 2015). Zu diesem Zweck werden Algorithmen oder Simulationstechniken herangezogen, welche auf Basis historischer Daten, Sensordaten, Wartungsprotokollen sowie externer Umweltdaten Trends, Muster und Zusammenhänge identifizieren (Schnell et al. 2018; Samon und Shalom 2023). Die Algorithmen durchlaufen dabei einen Lernprozess, indem die erhobenen Daten mit historischen Daten der betrachteten Einheit oder einer anderen Einheit der gleichen Art verglichen werden. Die Vorhersagegenauigkeit verbessert sich mit jeder zusätzlich analysierten Maschine sowie mit der Länge der Datenaufnahme. Dadurch entwickelt sich ein lernendes System, das aus jedem Schadensfall neue Informationen aufnimmt und somit seine Prognosefähigkeit kontinuierlich verbessert. Das Ziel ist, effektiver als bei den zuvor beschriebenen Instandhaltungsmethoden zu handeln, indem der notwendige Wartungsaufwand minimiert und die Reaktionszeit vor Ausfällen maximiert wird. Die Relevanz prädiktiver Instandhaltungsmaßnahmen wird durch eine Untersuchung von McKinsey verdeutlicht. Demnach können Stillstandszeiten von Maschinen durch prädiktive Instandhaltungsmaßnahmen um bis zu 50 % reduziert und gleichzeitig der Produktlebenszyklus um bis zu 40 % verlängert werden (McKinsey & Company 2015).

Neben den führenden Zielen der prädiktiven Instandhaltung, nämlich der Maximierung der Anlagenlebensdauer und der Minimierung ungeplanter Ausfallzeiten, soll der Einsatz von Wartungsarbeiten so effizient wie möglich gestaltet werden (Samon und Shalom 2023). Die prädiktive Wartung ermöglicht die effiziente Planung sowohl des exakten Zeitpunkts von Wartungsarbeiten als auch der Art der Wartung (Esteban et al. 2022). In der vorliegenden Arbeit wird der Begriff der prädiktiven Instandhaltung als Oberbegriff verwendet, der das gesamte Unterthemengebiet der prädiktiven Wartung miteinschließt. Die frühzeitige Erkennung potenzieller Probleme ermöglicht eine gezielte Wartung zum optimalen Zeitpunkt, anstatt eine beliebige Wartung durchzuführen, wie es bei der präventiven Wartung der Fall ist (Jardine et al. 2006). Die zustandsorientierte Bewertung von Anlagen und Maschinen resultiert in einer Verlängerung des Zeitintervalls zwischen

der Erkennung eines potenziellen Fehlers und dessen tatsächlichem Auftreten im Vergleich zu den traditionellen Instandhaltungstechniken (Fraunhofer IVI o.D.). Ein potenzieller Fehler kann folglich durch Methoden zur Bewertung des Zustandes deutlich früher identifiziert werden, sodass im Rahmen der prädiktiven Wartung Präventivmaßnahmen an der potenziellen Fehlerquelle ergriffen werden können, bevor die Störung tatsächlich eintritt. Die Vergrößerung des Zeitraums, in dem ein Fehler erkannt und wirksam wird, erlaubt zudem eine flexiblere Durchführung geplanter Wartungsmaßnahmen, da ein größeres Zeitfenster zur Durchführungsplanung oder Beschaffung von Ersatzteilen zur Verfügung steht.

Die Entwicklung der möglichen Instandhaltungsaktivitäten durch neue technische Möglichkeiten hat dazu geführt, dass sich die Aktivität in dieser Disziplin von reinen Reaktionen auf Störungsereignisse hin zu proaktiven Vermeidungsstrategien von ungeplanten Stillständen entwickelt hat (Mühlnickel et al. 2018). Damit hat sich je nach Instandhaltungsstrategieanwendung die Bedeutung des Bereiches Instandhaltung und damit verbundenen Wartungsarbeiten in Unternehmen geändert. Während frühere Instandhaltungspraktiken in reaktiver und präventiver Form klassischerweise als Kostenverursacher betrachtet wurden, werden sie heute als Kostenvermeider angesehen (Mühlnickel et al. 2018). Auch bei den Instandhaltern ist eine Verschiebung des geforderten Kompetenzbereichs zu beobachten. Dieser umfasst nicht mehr ausschließlich die klassischen technischen, mechanischen und elektrotechnischen Disziplinen, sondern beinhaltet zunehmend auch interdisziplinäre Themen wie Sensorik, Kommunikationstechniken, Software, Daten- und Asset-Management (Alcalde Rasch 2000; Mühlnickel et al. 2018). Trotz der beschriebenen Vorteile durch zustandsorientierte und darauf aufbauenden Instandhaltungsmethoden sollten diese Strategien nicht wahllos auf alle beliebigen Betriebsmittel angewendet werden, sondern stets individuell nach Bedeutung des Betriebsmittels für den Gesamtprozess, Ausfallrisiko und den wirtschaftlichen Folgen abgewogen werden (Hodapp 2018). Eine Instandhaltungsstrategie für eine Produktionsanlage kann somit aus verschiedenen Instandhaltungsmethoden bestehen und ist stets anlagenspezifisch.

2.4.2 Umsetzung der prädiktiven Instandhaltung

Verfahren aus dem Bereich maschinellen Lernens sind der Kernpunkt bei der Entdeckung von Anomalien also Abweichungen von erwarteten Mustern und daraus abgeleiteten Vorhersagen für Wartungsmaßnahmen (Fraunhofer SCAI o.D.; Paul et al. 2024). Prädiktive Instandhaltungsmethoden lassen in modellbasierte und datengesteuerte Ansätze sowie einer Hybridversion aus beiden einteilen (Lee 1998). Modellbasierte

Ansätze basieren auf einem physikalischen Verständnis von Prozessen und nutzen analytische Modelle, um das Systemverhalten zu simulieren und vorherzusagen und berechnen Fehler durch die Abweichung der modellierten Berechnung und tatsächlichen Werten (Peng et al. 2010). Der datengesteuerte Ansatz zielt auf die Konstruktion eines Modells ab, welches durch den Input verschiedener Datenquellen und Verfahren wie maschinelles Lernen und Data Mining das reale Systemverhalten erlernen soll (Fraunhofer SCAI o.D.). Ein hybrider Ansatz verbindet die zuvor genannten Praktiken. Ein Beispiel für einen hybriden Ansatz ist der sogenannte digitale Zwilling, der die Simulation eines Systems durch physikalische Modelle, Sensor- und historische Daten durchführt, um die Lebensdauer des abgebildeten Systems abzubilden (Glaessgen und Stargel 2012). Das Konzept des Digitalen Zwillings umfasst demnach das physische Abbild des Produkts, das virtuelle Abbild desselben sowie die bidirektionalen Daten- und Informationsflüsse zwischen den beiden Produkten.

Die datengesteuerten Modelle untergliedern sich weiterhin in sensorbasierte, protokollbasierte oder hybride Varianten aus beiden Praktiken (Gutschi et al. 2019). Die protokollbasierten Modelle, die als Top-Down-Ansatz bezeichnet werden, verarbeiten historische Ereignisprotokolldaten, um Algorithmen des maschinellen Lernens zu trainieren. Dies ermöglicht die gleichzeitige Betrachtung einer Vielzahl von Komponenten mit einem gemeinsamen Datenstrom. Anwendung finden diese Modelle zumeist in der Umgebung von IT-Systemen wie Geldautomaten (Wang et al. 2015).
Die sensorbasierte Wartungsplanungen, die häufig in Fertigungsprozessen Anwendung finden, werden als Bottom-up-Ansatz gesehen und basieren auf Zeitreihensignalen der Sensoren (Hashemian und Bean 2011). Hashemian und Bean (2011) führen aus, dass dabei die Überwachung von einzelnen mit Sensoren versehenen Komponenten im Vordergrund steht. Daraus können dann im Themenfeld prädiktive Wartung entweder Kennzahlen zur Zustandsbewertung von Einheiten oder aber Fehlerprognosen erstellt werden. Die Indikatoren zur Bewertung des Zustands dienen dem System als Entscheidungshilfe zur Unterteilung in einen funktionierenden oder fehlerhaften Betrieb. Eine einfache Entscheidungsgrundlage ist die Ermittlung eines Schwellenwertes für bestimmte Zustandsindikatoren. Überschreiten die betrachteten Sensorwerte diesen Schwellenwert, liegt ein fehlerhafter Zustand vor. Für eine genauere Abschätzung bezüglich eines fehlerhaften Zustandes bietet sich ein Vergleich mit statistischen Verteilungen von Indikatorwerten an, um die Wahrscheinlichkeit eines spezifischen Fehlerzustandes abzuschätzen. Zur Zustandsüberwachung werden durch die Sensoren Daten zu Parametern wie Temperatur, Druck, Spannung, Geräuschen oder Vibration gemessen. In diesem Kontext sei darauf verwiesen, dass es sich bei den als "Sensordaten" bezeichneten Daten um Rohdaten handelt. Erst nach einer Vorverarbeitung der Daten, welche Prozesse

wie Reinigen, Filtern, Aggregieren, Reduzieren oder das Bringen in ein einheitliches Format umfasst, sind diese für weitere Analysen und Modellierungen sinnvoll nutzbar (Welte et al. 2020). Im Rahmen der prädiktiven Wartung finden zudem Log-Dateien, welche Ergebnisse und Fehlercodes dokumentieren, manuelle Inspektionsberichte, SPS-Daten sowie Kommunikationsprotokolle als weitere Rohdatenquellen Berücksichtigung (Schnell et al. 2018). Des Weiteren besteht die Möglichkeit, unstrukturierte Daten, wie beispielsweise Wartungsberichte, Bilder, Videos oder Audiodateien von Maschinengeräuschen, in die Auswertung miteinzubeziehen (Schnell et al. 2018).

Neben der Zustandsbewertung ist das zweite verbreitete Themenfeld im Kontext der sensorbasierten prädiktiven Wartung das Prognostizieren von Fehlern betrachteter Einheiten. Zusammengefasst wird dabei durch das Kombinieren von Daten vergangener und aktueller Zustände einer Maschine versucht vorherzusagen, wann ein fehlerhafter Zustand auftreten wird (Meddaoui et al. 2024). Durch Techniken des maschinellen Lernens werden somit zeitliche Prognosen zur verbleibende Nutzungsdauer (RUL) einer betrachteten Maschine oder Komponente oder der geschätzten Zeit bis zum Ausfall der Maschine gemacht, indem Algorithmen zukünftige Werte der relevanten Zustandsindikatoren vorhersagen. Die berechnete RUL einzelner Komponenten dient dann als Grundlage zur Erstellung eines sinnvollen Wartungsplanes, um Anlagenausfälle präventiv zu vermeiden.

Um auf die Verknüpfung zwischen KDD-Vorgehensmodellen und maschinellem Lernen in der später vorgestellten Methode hinzuarbeiten, wird im folgenden Kapitel beschrieben, welche datenspezifische Problematiken bei der Einführung und Nutzung prädiktiver Instandhaltungsmethoden bestehen.

2.4.3 Datenwissenschaftliche Herausforderungen bei der Umsetzung prädiktiver Instandhaltungsmethoden

Die bisherige Ausarbeitung hat gezeigt, dass die prädiktive Instandhaltung in produzierenden Unternehmen darauf abzielt, Wissen über zukünftige Ausfälle von Anlagen oder Anlagenkomponenten zu generieren, um potenzielle Anlagenausfälle prognostizieren und mit der prädiktiven Wartung Wartungsaktivitäten optimieren zu können. Die Erfassung von Wissen über Maschinenzustände erfordert die Berücksichtigung von Informationen und Daten, die auf den nach der Northschen Wissenstreppe (vgl. Kapitel 2.1.1) vorgelagerten Stufen zu finden sind. Die Qualität der Prognosen, die durch maschinelle Lernmodelle generiert werden, ist maßgeblich von den Daten und Informationen abhängig, die dem maschinellen Lernen als Input zur Verfügung gestellt werden (vgl. Kapitel 2.4.1).

Eine Herausforderung, die sich bei der Nutzung prädiktiver Wartungsstrategien in produzierenden Unternehmen daher ergibt, ist die Bereitstellung einer aussagekräftigen Datenbasis als Grundlage für die Analysen und Prädiktionen von Ausfällen. Wie bereits beschrieben, ist insbesondere die sensorbasierte Erfassung von Daten anfällig für Datenmängel (vgl. Kapitel 2.4.1). So kann beispielsweise eine zu große Anzahl von Sensoren, mit denen viele nicht benötigte Daten gesammelt werden, zu Rauschen und einer hohen Übermittlungs- und Verarbeitungszeit der Daten führen (Esteban et al. 2022). Weitere Datenqualitätsmängel können fehlende Werte im Datensatz, Ausreißer, die das Ergebnis verfälschen, redundante oder inhaltlich überflüssige Daten oder eine zu hohe Dimensionalität sein (Cios et al. 2007). Die Generierung eines robusten Vorhersagemodells zur Unterstützung von Instandhaltungsentscheidungen erfordert die Bereitstellung aussagekräftiger Trainingsdaten. Die Verwendung von unzureichenden Daten, welche durch inkonsistente Datenerfassung oder unvollständige Datensätze gekennzeichnet sind, kann zu einer signifikanten Verschlechterung der Vorhersagequalität führen (vgl. Kapitel 2.1.4). Wenn die Trainingsdaten, die dem Modell zur Verfügung gestellt werden, Rauschen enthalten, d. h. zufällige Fehler und Sichtbarkeiten, die nicht die tatsächlichen Beziehungen zwischen den Eingangs- und Ausgangsvariablen widerspiegeln, besteht die Gefahr, dass ein überangepasstes Modell dieses Rauschen integriert und zu suboptimalen Vorhersagen für neue Daten führt (Paul et al. 2024). Robuste Vorhersagemodelle hingegen sind in der Lage, verlässliche Prognosen zu treffen, auch wenn die Daten unvollständig oder von Rauschen geprägt sind. Zudem können sie auf neue, bislang unbekannte Datenquellen angewendet werden und liefern dabei verallgemeinerbare Ergebnisse (vgl. Kapitel 2.4.1).

Eine weitere Herausforderung datengetriebener prädiktiver Instandhaltung besteht in der Strukturierung, Analyse und Integration industrieller Big-Data-Datensätze in die bestehende unternehmensinterne IT-Landschaft. Diese In Kapitel 2.1.1 beschriebenen Datensätze stammen in der Regel aus heterogenen Quellen und weisen daher eine hohe Komplexität auf (Kurrewar et al. 2021). Durch die meist hohe Dimensionalität dieser Datensätze wird außerdem die Wahrscheinlichkeit erhöht, dass das maschinelle Lernen ein Programm antrainiert, das nicht auf das betrachtete Gesamtproblem bezogen werden kann, da es nur für einen kleinen Datenausschnitt relevant ist (Yan et al. 2017). Die Verwendung von Algorithmen, welche in kleineren Datensammlungen zu validen Resultaten führen, ist aufgrund des oftmals großen Volumens realer Datenbestände nicht möglich. Dies liegt darin begründet, dass die Algorithmen entweder zu viel Zeit benötigen oder aufgrund der Größe der Datenmenge nicht mehr berechenbar sind (Bissantz et al.).

Zusätzlich kann die prädiktive Wartung je nach Zielstellung des Unternehmens als Echtzeitproblem aufgefasst werden. Aus dieser Perspektive ist es erforderlich, die handlungsrelevanten Prognosen stets unter Berücksichtigung der aktuellen Entwicklungen aufzuarbeiten, um Abweichungen und Handlungsbedarf in der gewünschten Zeit zu erkennen (Paul et al. 2024).

Als wesentliches Hindernis für die Implementierung von Analysemethoden in industriellen Kontexten kann außerdem die fehlende Verfügbarkeit strukturierter Ansätze für die Durchführung datenbezogener Methoden identifiziert werden (Rotondo und Quilligan 2020). Auch die Schaffung einer Umgebung zur Implementierung des maschinellen Lernens in bestehende Betriebsprozesse stellt Anwender vor Herausforderungen wie das Anpassen bestehender Prozesse oder das Bereitstellen leistungsfähiger Rechenressourcen, Speicher und Datenverarbeitungskapazität.

In Kapitel 2.2.1 sind bereits KDD-Vorgehensmodelle als möglicher Ansatz einer datenbezogenen Methode zur Generierung von Wissen beschrieben worden. Um eine detaillierte auf die prädiktive Wartung zugeschnittene Methode zu entwickeln, wird im folgenden Kapitel 3 Data Mining im Rahmen von KDD mit maschinellem Lernen kombiniert. Diese Idee ergibt sich aus dem unterschiedlichen Anwendungszweck beider Methoden. Wie im Kapitel 2.21 dargelegt, besteht das Ziel des Data Mining darin, im Rahmen des KDD-Vorgehensmodells Wissen in Form von nützlichen, verständlichen, neuen und nicht trivialen Mustern zu entdecken. In Abhängigkeit der Zielstellung werden auf Basis der betrachteten Daten statistische Methoden und Algorithmen angewendet. Die Ermittlung von Wirkzusammenhängen, um kausale Beziehungen aufzuzeigen, die Klassifizierung von Objekten, die Identifikation von Beziehungen zwischen Variablen durch Regressionsanalysen oder die Gruppierung von Objekten mittels Clusteranalysen stellen dabei die häufigsten Aufgaben in der Anwendung von Data Mining dar (Runkler 2015). Das Data Mining dient zusammengefasst der Wissensgewinnung aus Daten. Zu diesem Zweck werden unter anderem maschinelle Lernmethoden als Werkzeug eingesetzt. Darüber hinaus umfasst das Data Mining auch Techniken zur Datenverwaltung und -analyse (vgl. Kapitel 2.2.2). Beim maschinellen Lernen hingegen lernt ein Computer aus vorliegenden Daten oder daraus abgeleiteten Eigenschaften und ist dadurch in der Lage, komplexe Aufgaben zu lösen (vgl. Kapitel 2.3). Im Gegensatz zum Data Mining, bei dem mindestens beim Evaluationsschritt der entdeckten Muster und Erkenntnisse eine menschliche Expertise erforderlich ist, erfolgt die Entwicklung von Algorithmen, die aus Daten lernen, im Rahmen des maschinellen Lernens ohne menschliches Eingreifen.

Die dargelegten datenwissenschaftlichen Herausforderungen bei der Einführung einer prädiktiven Wartungsstrategie in produzierenden Unternehmen und die unterschiedlichen Anwendungsziele von Data Mining und maschinellem Lernen bilden den Ausgangspunkt für die nachfolgend im dritten Kapitel entwickelte Methode.

3 Entwicklung einer Methode als prädiktive Wartungsstrategie

In diesem Kapitel erfolgt die Zusammenführung der im zweiten Kapitel grundlegend beschriebenen datenwissenschaftlichen Themen maschinelles Lernen und Data Mining im Rahmen von KDD-Vorgehensmodellen auf das Anwendungsgebiet der prädiktiven Wartung in produzierenden Unternehmen. Die hier beschriebene Methode zielt darauf ab, einen Beitrag zur Bewältigung der im Kontext der prädiktiven Wartung beschriebenen datenbezogenen Herausforderungen zu leisten. Zu diesem Zweck wird in Kapitel 3.1 zunächst der Anwendungsbereich der entwickelten Methode durch die Auswahl eines Teilbereiches der prädiktiven Wartung sowie die Wahl eines maschinellen Lernbereichs und eines KDD-Vorgehensmodells eingegrenzt. Anschließend folgt in Kapitel 3.2 die Entwicklung der Methode, so wie das Herausarbeiten möglicher Vorteile in Kapitel 3.3.

3.1 Abgrenzung des Anwendungsbereichs und relevanter Verfahren der zu entwickelnden Methode

In Kapitel 2 wurde gezeigt, dass verschiedene Ansätze zur vorausschauenden Instandhaltung sowie verschiedene Algorithmen des maschinellen Lernens und KDD-Vorgehensmodelle existieren. Um eine zielgerichtetere Methode entwickeln zu können, werden die drei genannten Themen im Folgenden jeweils abgegrenzt.

Auswahl eines Teilbereichs der prädiktiven Wartung:

Die Entwicklung einer Methode erfordert zunächst die präzise Definition des Teilgebiets der prädiktiven Wartung, auf das sie abzielen soll. Im Rahmen der vorliegenden Untersuchung wird eine datenbasierte Perspektive der prädiktiven Wartung gewählt, um eine Methode zur Festlegung effizienter Wartungszeitpunkte zu definieren. Die Auswahl der datenbasierten Perspektive ist auf die geplante Integration von KDD-Vorgehensmodellen in die Methode zurückzuführen. In der Definition von Fayyad et al. (1996) wird das Ziel von KDD-Vorgehensmodellen wie folgt beschrieben: Es geht darum, gültige, innovative und nützliche Muster in Daten aufzudecken (vgl. Kapitel 2.2). Daher erweist es sich als zweckdienlich, sich auf die datengesteuerten Ansätze zu beschränken und modellbasierte Ansätze (vgl. Kapitel 2.4.2) zur Erstellung prädiktiver Wartungspläne im Folgenden nicht zu berücksichtigen. Dies ist darin begründet, dass sich letztere, wie in Kapitel 2.4.2 beschrieben, zumeist nicht auf eine datengesteuerte Perspektive, sondern auf physikalisches Verständnis, Expertenwissen und analytische Methoden stützen. Der Fokus liegt stattdessen im Rahmen dieser Arbeit auf datengesteuerten Ansätzen, welche

durch die Integration verschiedener Datenquellen und maschinellen Lernens eine Vorhersage generieren.

Sensorbasierte Methoden finden vornehmlich in Fertigungsprozessen Anwendung (vgl. Kapitel 2.4.2), sodass sie sich in besonderem Maße für die vorliegende Arbeit eignen, da der Bereich der prädiktiven Wartung in produzierenden Unternehmen im Fokus steht. Bei sensorbasierten Methoden erfolgt die Bereitstellung der Rohdaten zur weiteren Verarbeitung direkt durch die Sensoren (vgl. Kapitel 2.4.2). In der Konsequenz lassen sich im Themenfeld der prädiktiven Wartung Kennzahlen zur Zustandsbewertung von Bauteilen oder Maschinen sowie Fehlerprognosen generieren. Die Funktionalität der beschriebenen prädiktiven sensorbasierten Verfahren, wie beispielsweise die Zustandsbewertung oder die Vorhersage von Ausfällen, basiert im Wesentlichen auf maschinellem Lernen (vgl. Kapitel 2.4.2). Die Auswahl eines adäquaten Algorithmus ist dabei maßgeblich von der präzisen Zielsetzung in der Anwendung prädiktiver Wartung sowie dem Grad der gewünschten Vorhersagegenauigkeit abhängig.

In der unternehmerischen Praxis zeigt sich jedoch, dass produzierenden Unternehmen nicht immer die benötigten Datenstrukturen zur Verfügung stehen, die für jeden maschinellen Lernalgorithmus geeignet sind. In diesem Kontext ist zu eruieren, welcher Bereich des maschinellen Lernens auf Basis der verfügbaren Betriebsmittel und Datenbasis am geeignetsten erscheint. Die Auswahl des maschinellen Lernverfahrens für die zu entwickelnde Methode kann anhand verschiedener Kriterien erfolgen. Dazu zählt beispielsweise die Untersuchung der vorliegenden Daten hinsichtlich ihrer Labelung mit Fehlerereignisse sowie der Datentypen, wobei hier zwischen kategorischen und kontinuierlichen Daten unterschieden werden kann (vgl. Kapitel 2.3).

Auswahl des maschinellen Lernbereichs:

Im Rahmen dieser Arbeit erfolgt eine Fokussierung auf die Entwicklung der Methode im Bereich der prädiktiven Wartung unter Einbezug überwachter Algorithmen des maschinellen Lernens. Dies impliziert die Kenntnis von Informationen bezüglich des Auftretens von Fehlern und eines korrekten Anlagenzustandes sowie die Klassenzuteilung der Maschinenzustände durch Labeln. Diese Vorgehensweise ist notwendig, da in Kapitel 2.3 dargelegt wurde, dass überwachte Algorithmen nur mit strukturierten, gelabelten Daten arbeiten können. Demgegenüber stehen unüberwachte Lernmethoden, wie beispielsweise die Identifizierung verschiedener Anlagenbedingungen durch Clustering. In diesem Kontext sind keine Kenntnisse über einen ordnungsgemäßen Betrieb und etwaige Fehler erforderlich, sodass die Daten auch nicht-gelabelt sein können (vgl. Kapitel 2.3). Für spezifische Aufgaben der prädiktiven Wartung, wie beispielsweise die Vorhersage

von Fehlern oder die Einordnung von Betriebszuständen, scheinen sich überwachte Algorithmen besser zu eignen. Im Kapitel 2.3 wurde dargelegt, dass unüberwachte Algorithmen nicht auf spezifische, sondern vielmehr auf allgemeine Muster oder Anomalien ausgerichtet sind. Des Weiteren birgt die Abwesenheit von gelabelten Daten das Risiko von Fehlern, da keine reale, gemessene Referenz zum Vergleich zur Verfügung steht, anhand derer ein normales Maschinenverhalten von einem fehlerhaften Zustand unterschieden werden kann. Im Gegensatz dazu ermöglicht das überwachte Lernen die Bestimmung von linearen oder nichtlinearen Wechselwirkungen zwischen Daten, wobei eine große Anzahl von Merkmalen berücksichtigt wird. Die zwingende Beschriftung der Daten bei überwachten Algorithmen bedingt bereits weiterreichende Kenntnisse über beispielsweise den fehlerhaften oder intakten Zustand, welche mit den Daten verknüpft sind. Dadurch sind zum Beispiel Prognosen über den zukünftigen Zustand möglich, wie das Vorhersagen einer Zielvariable. Da die Daten bei unüberwachten Algorithmen in der Regel nicht beschriftet sind (vgl. Kapitel 2.3), erfolgt durch den Algorithmus eine eher blinde Suche nach Betriebszuständen oder verborgenen Mustern. Die Ergebnisse aus überwachten Algorithmen lassen sich also besser interpretieren und nachvollziehen. Auch in der Praxis zeigt sich, dass im Bereich der prädiktiven Instandhaltung Probleme am häufigsten mit gelabelten Daten in Bezug auf den aktuellen Betriebszustand des betrachteten Systems behandelt werden (vgl. Kapitel 2.4.1). In Kapitel 2.4.1 wurde außerdem beschrieben, dass im Bereich des überwachten Lernens insbesondere die Mehrklassenklassifikation, gefolgt von der Regression und der binären Klassifikation als am häufigsten genutzte Techniken des maschinellen Lernens gelten.

Im vorherigen Abschnitt dieses Kapitels wurde ein Fokus auf die sensorbasierte prädiktive Wartung festgelegt, welche sich in die Zustandsbewertung und die Vorhersage der verbleibenden Nutzungsdauer beziehungsweise die Bestimmung des Eintrittszeitpunkts eines Fehlers unterteilen lässt. Im Rahmen der Zustandsbewertungsaufgabe erweisen sich insbesondere Klassifizierungsalgorithmen als geeignet (vgl. Kapitel 2.4.1). Dies ist darauf zurückzuführen, dass im Wesentlichen eine kategorische Unterteilung der gelabelten Daten in "fehlerhaft" oder "nicht fehlerhaft" vorgenommen wird. In Abhängigkeit der vorliegenden Daten kann eine binäre Klassifizierung zwischen den beiden Zuständen oder, zur detaillierteren Fehlervorhersage, eine Multiklassenklassifizierung vorgenommen werden. Eine Multiklassenklassifizierung setzt allerdings voraus, dass sich die Daten mit eindeutigen Fehlermodi und Zusammenhängen kennzeichnen lassen. Im Gegensatz dazu erweisen sich Regressionsalgorithmen für die Erstellung von RUL-Prognosen als besonders geeignet, da die Zielvariable numerisch ist und durch die Berücksichtigung eines oder mehrerer weiterer Variablen prognostiziert wird.

Im Folgenden soll untersucht werden, inwiefern sich die Anwendung eines KDD-Vorgehensmodells vor Durchführung des maschinellen Lernens bei beiden Arten von Algorithmengruppen als vorteilhaft erweist und welche Anwendungsmöglichkeiten sich daraus ergeben. Das Ziel besteht in der Verbesserung der Ergebnisse des maschinellen Lernens im Rahmen der prädiktiven Wartung, um die Effizienz und Zuverlässigkeit von Produktionsprozessen zu steigern. Wie in Kapitel 2.1.4 dargelegt, besteht das Ziel des Data Mining in der Generierung neuartiger und nützlicher Erkenntnisse aus Daten. Die ausschließliche Anwendung von Data-Mining-Methoden kann jedoch zu nicht aussagekräftigen bis fehlerhaften Informationen führen (vgl. Kapitel 2.2). Daher werden mit KDD-Vorgehensmodellen strukturierte Rahmen mit vor- und nachgelagerten Schritten zur vorherigen Bearbeitung der Daten und Analyse bereitgestellt

Im weiteren Verlauf dieser Arbeit soll untersucht werden, inwiefern die Prognosegüte von Algorithmen des maschinellen Lernens durch vom KDD-Vorgehensmodell generiertes Wissen verbessert werden kann. Wie in Kapitel 2.4.2 dient Data Mining der Wissensgewinnung aus großen Datensätzen, während maschinelles Lernen für komplexere Aufgaben wie das Prädiktieren von Ausfallzeitpunkten von Anlagen geeignet ist (vgl. Kapitel 2.4.2).

In Kapitel 2.2.2 wurde dargelegt, dass Data Mining vielfach als eine hypothesenfrei durchgeführte Suche nach Datenbeziehungen definiert wird. Es existieren jedoch auch Meinungen, die besagen, dass Data Mining hypothesenbehaftet ist (vgl. Kapitel 2.2.2). Im Rahmen des hypothesenfreien Ansatzes erfolgt keine Prüfung einer im Vorfeld definierten Hypothese, sondern eine möglichst unvoreingenommene Erforschung allgemeingültigen Wissens. Dies kann zur zufälligen Entdeckung bislang unbekannter Muster und Zusammenhänge in den Daten führen. In der Praxis erweist sich die Umsetzung einer vollständig hypothesenfreien Durchführung von Data Mining jedoch als anspruchsvoll. Demgemäß kann argumentiert werden, dass bei einem Vorgehen ohne Hypothesen sämtliche verfügbaren Werkzeuge und Spektren des Data Mining ohne eine bestimmte Methode oder Zielsetzung genutzt werden müssten. Die Auswahl eines Werkzeugs oder Spektrums wäre folglich bereits mit einer latenten Absicht und damit einer Hypothese verbunden. Des Weiteren gestaltet sich die Evaluierung des identifizierten Wissens anspruchsvoll, sofern zuvor keine Ziel- und Unterstellungsbildung erfolgt ist, auf deren Basis eine Bewertung der Ergebnisse möglich wäre. Zudem basiert der gesamte KDD-Prozess von Beginn an auf der Hypothese, dass aus den betrachteten Daten nützliches Wissen extrahiert werden kann. In einer Gegenposition wird daher postuliert, dass zu Beginn des Data-Mining-Prozesses eine Hypothese als deduktiver Ansatz der Wissensentdeckung aufgestellt wird, auf deren Grundlage weitere Maßnahmen zur

Wissensentdeckung durchgeführt werden (vgl. Kapitel 2.2.2). Im Rahmen der nachfolgenden Methode können zu explorativen Zwecken sowohl hypothesenfreie Data-Mining-Verfahren angewendet werden als auch Annahmen oder Theorien bezüglich der betrachteten Daten genauer überprüft werden.

Aufgrund der divergierenden Schwerpunkte von Data Mining und maschinellem Lernen erscheint eine Kombination beider Methoden in einem Gesamtmodell zur prädiktiven Wartung als sinnvoll. In der Modellierungsphase bedient sich das Data Mining in der Regel Methoden des maschinellen Lernens, wie sie in den KDD-Vorgehensmodellen definiert sind (vgl. Kapitel 2.2.2). Die Anwendung von Methoden des maschinellen Lernens zum Prädiktieren von Maschinen oder zur Zustandsbewertung ist jedoch andersrum nicht zwingend an Data Mining gebunden. Im Kontext der prädiktiven Wartung produzierender Unternehmen erscheint es daher sinnvoll, implizites Wissen, Regeln und Muster sowie daraus resultierende Informationen aus Sensordaten durch Data Mining zu identifizieren und anschließend im Rahmen des maschinellen Lernens das Verhalten des Fertigungssystems oder der betrachteten Maschine zu trainieren, um ein nützliches Vorhersagemodell für Prognosen von RUL oder Ausfallwahrscheinlichkeiten zu entwickeln.

In einer zusammenfassenden Betrachtung lässt sich festhalten, dass das Data Mining im Kontext der prädiktiven Wartung dazu geeignet ist, Erklärungen für vergangene Ereignisse zu suchen, während das maschinelle Lernen versucht, Prognosen für künftige Ereignisse zu treffen. Im Folgenden wird erörtert, welches KDD-Vorgehensmodell sich zur Anwendung von Data Mining als vorbereitender Schritt für das maschinelle Lernen im Kontext der sensorbasierten prädiktiven Wartung am besten eignet.

Auswahl eines KDD-Vorgehensmodells:

Wie im Kapitel 2.2 dargelegt, zielt der KDD-Prozess darauf ab, valide, innovative, verständliche und nützliche Muster aus Datensätzen durch einen strukturierten Prozess zu identifizieren. Im Folgenden soll untersucht werden, ob eines der in Kapitel 2.2.2 verglichenen KDD-Vorgehensmodelle für den Anwendungsbereich der prädiktiven Wartung geeignet scheint. Im Bereich KDD-Vorgehensmodelle gilt das Modell von Fayyad et al. (1996) als ursprünglicher Ansatz, während das CRISP-DM und darauf aufbauende Modelle als zentraler Ansatz gesehen werden (vgl. Kapitel 2.2.2). Darüber hinaus existiert in der Literatur noch eine Vielzahl weiterer KDD-Vorgehensmodelle (vgl. Kapitel 2.2.2). Dabei ist festzuhalten, dass es nicht ein bestes Modell gibt, sondern dass jedes Modell mit spezifischen Stärken und Schwächen einhergeht. Diese sind abhängig vom Anwendungsbereich sowie den Zielen des Wissensentdeckungsprozesses.

Das CRISP-DM-Vorgehensmodell qualifiziert sich als KDD-Vorgehensmodell für die zu entwickelnde Methode durch seine klar definierte Projektstruktur, die eine intuitive Anwendung ermöglicht und sich auf den Anwendungsbereich prädiktive Wartung zuschneiden lässt (vgl. Kapitel 2.2.2). Obgleich das Vorgehensmodell an sich zunächst einen allgemeingültigen Ansatz zur Bewältigung sämtlicher Data-Mining-Situationen darstellt, geben die generischen Aufgabenstellungen sowie eine Auswahl an Werkzeugen zur Lösung spezifischer Data-Mining-Fragestellungen Aufschluss über die Anwendungsmöglichkeiten des Modells (vgl. Kapitel 2.2.2). Fayyad et al. (1996) hingegen geben keine konkreten Empfehlungen für Data-Mining-Techniken oder geeignete Werkzeuge für spezifische Problemstellungen (vgl. Kapitel 2.2.2).

Für das CRISP-DM spricht ferner, dass das KDD-Vorgehensmodell von Fayyad et al. (1996) den Prozessschritt der Datenerfassung nur unzureichend berücksichtigt, da Fayyad et al. (1996) von einer bereits vorhandenen Datenbasis ausgehen (vgl. Kapitel 2.2.2). Da in dieser Arbeit jedoch sensorbasierte Daten als Grundlage für Prädiktionen verwendet werden, ist es sinnvoll, den Datenerfassungsschritt gesondert zu berücksichtigen.

Ein weiterer Vorteil für die zu entwickelnde Methode, der sowohl beim CIRSP-DM als auch bei Vorgehensmodell von Fayyad et al. (1996) gegeben ist, ist der iterativer Ansatz des Phasendurchlaufs (vgl. Kapitel 2.2.2). Für die zu entwickelnde Methode erweist sich diese Vorgehensweise als vorteilhaft, da die Ergebnisse der einzelnen Prozessschritte einer kontinuierlichen Neubewertung unterzogen und bei Bedarf weiterverarbeitet werden können. Der Anwendungsbereich der prädiktiven Wartung erfordert eine kontinuierliche Anpassung und Verbesserung der Modelle, da sich Maschinendaten im Laufe der Zeit mit hoher Wahrscheinlichkeit ändern. Die zusätzliche zyklische Eigenschaft des CRISP-DM-Vorgehensmodells kann in der Gesamtmethode nutzbringend angewendet und ausgedehnt werden. Die Erkenntnisse aus dem maschinellen Lernen können folglich in den KDD-Prozess zurückfließen und einer erneuten Anpassung unterzogen werden. Der Gesamtprozess erlaubt somit eine fortlaufende Anpassung, ohne dass eine vollständige Neustrukturierung des Projekts und des Modells oder eine erneute Durchlaufung aller Phasen erforderlich ist. Dadurch kann gewährleistet werden, dass die verarbeiteten Daten stets relevant sind und die Ergebnisse mit den in der Phase des Geschäftsverständnisses definierten Zielen übereinstimmen. Im Gegensatz zum CRISP-DM-Vorgehensmodell weist das Modell von Fayyad et al. (1996) eine begrenzte Rückkopplungsschleife auf. Dies wird ersichtlich, da die Prozessschritte in der Modellabbildung (siehe Abbildung 3) nicht untereinander verbunden sind, sondern lediglich die am Ende identifizierten Muster in alle vorherigen Schritte zurückgeführt werden können.

Eine weitere Abgrenzung von CRISP-DM zum Vorgehensmodell von Fayyad et al. (1996) besteht in einem größeren Fokus auf die Definition von Geschäftszielen sowie dem Integrationsprozess in bestehende Infrastrukturen eines Unternehmens (vgl. Kapitel 2.2.2). Die Hauptintention bei der Nutzung prädiktiver Wartungsstrategien besteht in der Regel in einer effizienteren Anlagenutzung und damit einhergehenden Kosteneinsparungen. Vor diesem Hintergrund erscheint die Verwendung des CRISP-DM als Vorgehensmodell im genannten Anwendungsbereich als sinnvoll. Dieses Modell orientiert sich zu Beginn des Prozesses an den Geschäftszielen und dem genauen Kontext. Das Einbeziehen von Hintergrundwissen und eine Zieldefinition aus Kundenperspektive werden zwar auch von Fayyad et al. (1996) Im Zuge der Datenphase erwähnt, diesem Punkt wird jedoch keine eigene Phase wie beim CRISP-DM-Vorgehensmodell gewidmet (vgl. Kapitel 2.2.2). Für die Wahl des CRISP-DM-Vorgehensmodell für die Methode spricht darüber hinaus, dass die Einbeziehung von Geschäftsexperten und Datenexperten in der ersten Phase des Prozesses empfohlen wird (vgl. Kapitel 2.2.1). Das ist in der prädiktiven Wartung gerade in der Neuimplementierung eines Prognosesystems für Maschinenausfälle von großer Bedeutung, da die Experten wichtige Informationen und Wissen in den Prozess einbringen können. Bei rein datengesteuerten Ansätzen ohne eine menschliche Instanz besteht indessen das Risiko, dass Zeit und Ressourcen aufgewendet werden, um Beziehungen zu entdecken, die bereits bekannt sind oder als trivial erachtet werden. Ein Beispiel für triviales Wissen ist die Erkenntnis, dass die Ausfallwahrscheinlichkeit von Bauteilen mit fortschreitender Betriebszeit ansteigt.

Die Vielzahl der eruierten Argumente für den Einsatz des CRISP-DM-Vorgehensmodells führt zu dessen Auswahl als Referenzmodell für die Gesamtmethode.

3.2 Aufbau der Methode

Die in diesem Kapitel entwickelte Methode zur Strukturierung einer sensorbasierten prädiktiven Wartungsstrategie kann gemäß der in Kapitel 2.1.3 vorgenommenen Definition als Vorgehensmodell bezeichnet werden. Dies begründet sich daraus, dass sie einen strukturierten Vorgehensrahmen mit einzelnen Prozessschritten sowie weitere Empfehlungen hinsichtlich der zeitlichen Ausführung und weitergehende Erklärungen beinhaltet. Ihre Basis bilden die zuvor definierten Einschränkungen: Betrachtet werden sensorbasierte prädiktive Instandhaltungsmethoden zur Prognose von Maschinenausfällen auf Grundlage von Zustandsbewertungen durch überwachte Lernalgorithmen. Ziel ist eine Verbesserung der Vorhersageleistung durch vorherige Anwendung von KDD-Schritten. Die Gesamtmethode setzt sich aus den in den Tabellen eins und zwei aufgeführten Schritten zusammen. Der erste Teil umfasst KDD auf den Bereich sensorbasierte

prädiktive Wartung zugeschnitten, in Anlehnung an das CRISP-DM-Vorgehensmodell. Der zweite Teil der Methode dreht sich um die Integration des aus dem ersten Teil erlangten Wissens in ein maschinelles Lernmodell zum Prädiktieren von Maschinenausfällen.

*Tabelle 1:*Teil 1 der entwickelten Methode

Teil 1: KDD in Anlehnung an das CRISP-DM-Vorgehensmodell	
Phase	**Kurzbeschreibung**
1. Prozessverständnis	Übergeordnete Zielstellung der prädiktiven Wartungsstrategie definieren unter Einbezug von Anlagen- und Datenexperten
2. Datenverständnis	Beschäftigung mit der effektiven Integration von Sensoren in den Fertigungsprozess und erste explorative Analyse der erfassten Sensordaten für ein übergeordnetes Verständnis
3. Datenaufbereitung	Anwendung von Datenaufbereitungsverfahren zur Sicherstellung einer vollständigen und fehlerfreien Datenbasis durch Entfernen unvollständiger, inkonsistenter oder fehlerhafter Daten und Imputation fehlender Werte. Daten in gelabeltes und einheitliches zur Weiterverarbeitung geeignetes Format bringen
4. Modellierung	Data-Mining-Verfahren anwenden, um tiefergehendes Wissen wie Korrelationen zwischen Sensorparametern oder Gruppierungen von Maschinenzuständen zur Erkennung von Ausfallrisiken zu erhalten
5. Evaluation	Bewertung der Erkenntnisse aus der Data-Mining-Anwendung hinsichtlich Relevanz für die Zielstellung
6. Implementierung	Das durch das Data Mining erworbene Wissen kann dem Anwender helfen den Fertigungsprozess zu verbessern und ein genaueres Augenmerk auf kritische Zusammenhänge zu legen. Das erlangte Wissen kann darüber hinaus als Input für maschinelle Vorhersagealgorithmen für Maschinenausfälle dienen

Im Folgenden wird den einzelnen Schritten der vorgeschlagenen Methode für die sensorbasierte prädiktive Wartung (Tabelle 1) einer genaueren Betrachtung unterzogen:

Prozessverständnis: Als vorbereitende Phase der Methode sollte zunächst für ein umfassendes Verständnis des Fertigungsprozesses mit allen bekannten Ausfallrisiken gesorgt werden. Die explizite Hinzufügung einer übergeordneten Zieldefinition in der Anwendung der Methode dient der frühzeitigen Auseinandersetzung mit den beiden Kernaktivitäten der Methode, dem maschinellen Lernen zur Prädiktion von Analgenausfällen und vorgelagertem Data Mining im Rahmen von KDD. Unter Berücksichtigung des Geschäftsverständnisses, welches die übergeordneten technischen und wirtschaftlichen Ziele der prädiktiven Instandhaltung definiert, kann das maschinelle Lernproblem von Beginn an eingegrenzt werden. Die Festlegung auf einen übergeordneten Bereich des maschinellen Lernens, wie beispielsweise die Klassifikation oder Regression, erlaubt zudem die Definition übergeordneter, hypothetischer Ziele bezüglich des durch KDD zu generierenden Wissens, welches als Input für das maschinelle Lernen dient. Gerade bei Neueinführung eines prädiktiven Wartungssystems im Unternehmen ist es sinnvoll, alle zu durchlaufenden Phasen der Methode von Experten bezüglich technischem- und Datenwissen begleiten zu lassen.

Datenverständnis: Die Datenverständnis-Phase zielt darauf ab, sich mit der Datenerhebung für die prädiktive Wartung zu beschäftigen und einen Überblick über diese zu erlangen und zu bewerten, ob die Daten nach den in Kapitel 2.2.2 beschriebenen Bedingungen für die Anwendung von Data Mining geeignet sind. Der Fokus bei der Datenerhebung liegt auf der Erfassung relevanter Parameter zur Beurteilung eines Maschinenzustandes durch Sensoren. Auf Basis der Zieldefinition sollte ersichtlich sein, welche Maschinen mit welchen Sensoren ausgestattet werden sollten, wie die Funktionsfähigkeit der Sensoren gewährleistet wird, wie die erfassten Daten gespeichert werden oder in welchen Zeitintervallen die Sensoren Daten speichern sollen. Daran anknüpfend ist eine Strategie zur Übertragung und Speicherung der Datenströme in Datenbanken zu entwickeln. Es kann ebenfalls für das spätere Training des Vorhersagemodells mehrwertstiftend sein, Sensordaten aus länger aufgezeichneten ähnlichen Fertigungsprozessen anderer Unternehmen zu verwenden, um eine breitere Datenbasis zu erhalten. Um ein Verständnis der aufgenommenen Daten zu erlangen, erfolgt eine explorative Grobanalyse der Daten, um schnell ersichtliche statistische Zusammenhänge oder Aussagen über die Datenqualität zu identifizieren, die durch die spätere Anwendung von Data Mining genauer untersucht werden könnten.

Datenaufbereitung: Das Ziel des Datenaufbereitungsschrittes ist es, eine konsistente und aussagekräftige Datenbasis zu schaffen, auf deren Grundlage Data Mining durchgeführt werden kann. Dazu werden historisch gespeicherte Sensordaten verwendet. Zu den Standardverfahren der Datenaufbereitung gehören das Entfernen von Ausreißern,

der Umgang mit fehlenden Werten oder das Entfernen von fehlerhaften Daten (vgl. Kapitel 2.1.4). Am Ende dieser Phase sollte der Datensatz in einem einheitlichen Format vorliegen.

Modellierung: Durch die Anwendung von Data-Mining-Verfahren gilt es in der Modellierungsphase Regeln, Korrelationen, Cluster oder ähnliches zu identifizieren, die neue Erkenntnisse über den Zusammenhang der durch Sensoren erfassten Parameter und das Risiko für einen Maschinenausfall liefern. Welche Art von Data-Mining-Verfahren sinnvoll sein kann, wird maßgeblich von der initial vorgenommenen Zieldefinition sowie den Resultaten der Datenphase beeinflusst. In Abhängigkeit von der konkreten Problemstellung bzw. den angestrebten maschinellen Lernalgorithmen lässt sich eine Vielzahl potenziell nutzbringender Data-Mining-Methoden und -Absichten identifizieren.

Evaluation: Zur Beurteilung des Nutzens des generierten Wissens bedarf es einer menschlichen Bewertung. Bei mehreren angewendeten Data-Mining-Verfahren muss bewertet werden, welche Erkenntnisse besonders mehrwertstiftend sind.

Implementierung: Das durch das Data Mining erworbene Wissen kann so signifikant sein, dass es dem Anwender wichtige Zusammenhänge liefert, die sich in die Wartungsstrategie implementieren lassen. Dadurch kann der Fertigungsprozess anpassend verbessert werden oder der Zustand kritischer Komponenten kann stärker überwacht, beziehungsweise Maschineteile gezielter gewartet werden.

Neben der Bereitstellung der erlernten Zusammenhänge für den Menschen, kann das Wissen aber auch für anschließendes maschinelles Lernen zur Prädiktion von Maschinenausfällen verwendet werden. Das auf dem KDD-Prozess aufbauende maschinelle Lernen stellt den zweiten Teil der Methode dar.

Tabelle 2: Teil 2 der entwickelten Methode

	Teil 2: Maschinelles Lernen	
7.	Feature-Generierung	Darstellung des aus dem KDD-Prozess erworbenen Wissens als Features für die Integration in das maschinelle Lernmodell
8.	Maschinelles Lernen	Auswahl eines geeigneten überwachten maschinellen Lernalgorithmus auf Basis der Zieldefinition und des im KDD erworbenen Wissens. Zunächst erfolgt die Modellierung des maschinellen Lernmodells auf Grundlage von historischen Sensordaten als Trainings- und Testdaten und den erstellten Features. Danach werden

	Vorhersagen auf Grundlage der aktuell gemessenen Sensordaten getroffen
9. Implementierung und Optimierung	Die Vorhersagen müssen in die bestehende Unternehmensstruktur integriert werden, indem sie den relevanten Personen zur Verfügung gestellt werden und in die Wartungsplanung integriert werden. Die Vorhersagen sollten in Kombination mit dem aus dem ersten Teil der Methode erworbenen Wissen stets kritisch hinterfragt werden

Feature-Generierung: Die im ersten Teil der KDD-Methode erworbenen Kenntnisse können durch die Erstellung neuer Features in das Training des maschinellen Lernmodells integriert werden. Dabei bietet sich die Komprimierung des Wissens in Form signifikanter Zusammenhänge zwischen Parametern, welche Maschinenausfälle verursachen können, an.

Maschinelles Lernen: Die Auswahl eines überwachten maschinellen Lernalgorithmus erfolgt auf Basis der anfänglichen Zieldefinition, die einschränkt, ob ein Klassifikations- oder Regressionsproblem vorliegt. Darüber hinaus kann das Wissen aus dem ersten Teil der Methode zur Auswahl des Algorithmus hinzugezogen werden. Die vorverarbeiteten Daten aus dem ersten Teil der Methode und die erstellten Features werden zur Generierung des maschinellen Lernmodells in Trainings- und Testdaten unterteilt. Der ausgewählte, trainierte Algorithmus verarbeitet anschließend die aktuellen an den Maschinen erfassten Sensordaten, um Prognosen zu generieren. Da in der Methode die Beschränkung auf überwachte maschinelle Lernalgorithmen vorliegt, muss jeder einfließende Datensatz mit Zeitstempel, Sensor- und Maschinentyp gelabelt werden.

Implementierung und Optimierung: Auf Grundlage der Prädiktion von Anlagenausfällen oder RUL in der vorherigen Phase lässt sich ein Wartungsplan erstellen. Die Zeitpunkte der Anlagenausfälle sollten regelmäßig hinterfragt und somit der Gesamtprozess stets kritisch evaluiert werden. prädiktiven Wartungsplan erstellen. In der Implementierungsphase ist eine Verknüpfung des prädiktiven Wartungssystems mit bestehenden Unternehmensinfrastrukturen vorzunehmen, um das prädiktierte Wissen für alle Beteiligten bereitzustellen. Dies ist erforderlich, um die Einhaltung der Wartungszeitpunkte auf Grundlage des prädiktierten Wissens sicherzustellen. Nachdem der Algorithmus des maschinellen Lernens unter Einbezug der Wissensgenerierungsphase trainiert worden ist, muss zudem eine möglichst automatisierte technische Strategie entwickelt werden.

Diese umfasst die Vorverarbeitung der Sensordaten aus dem Prozess und deren direkte Eingabe in den maschinellen Lernalgorithmus. Dadurch wird sichergestellt, dass der Algorithmus stets mit aktuellen Daten arbeitet.

Der Ablauf und die Datenflüsse der Gesamtmethode ist in Abbildung 8 nochmal separat dargestellt. Dabei fasst der Teil „KDD“ die in Tabelle 1 vorgestellten Phasen zusammen. Das maschinelle Lernen besteht aus den Phasen aus Tabelle 2. Die Verknüpfung beider Teile der Methode erfolgt durch den menschlichen Anwender, der die KDD-Ergebnisse bewertet, diese für sich nutzt oder als geeignete Feature-Generierung für das maschinelle Lernen betrachtet.

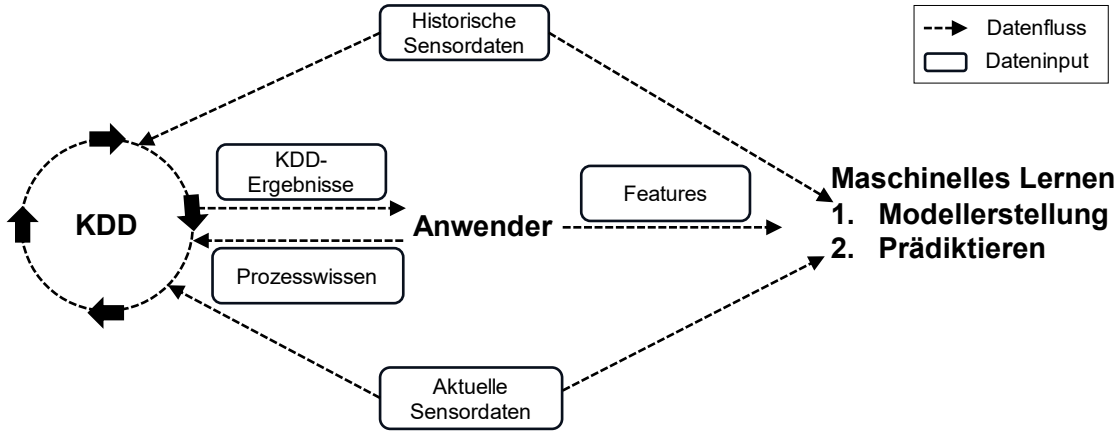


Abbildung 8: Prozessablauf und Datenflüsse der Methode

Der Kreislauf des Bereichs „KDD“ veranschaulicht den iterativen Prozess der Methode. Der Prozess zur Erstellung und Verbesserung des maschinellen Vorhersagemodells ermöglicht demnach eine fortlaufende Anpassung und kann nicht als abgeschlossen betrachtet werden. In Anlehnung an das CRISP-DM-Vorgehensmodell besteht also die Möglichkeit, die Phasen des ersten Teils der Methode nicht zwingend sequenziell, sondern in Abhängigkeit vom Ergebnis der jeweiligen Phase zu durchlaufen. Ein Rücksprung in eine andere Phase ist folglich jederzeit möglich. Auch nach der Modellerstellung des maschinellen Lernens im zweiten Teil der Methode können Anpassungen rückwirkend zur Modellierungs- oder Datenphase des KDD-Teils erfolgen, um bestimmtes Wissen nochmal genauer zu untersuchen oder hervorzuheben. In den KDD-Prozess fließen wie in Abbildung 8 zu sehen zunächst historische aufgenommene Daten der installierten Sensoren oder ergänzend externe Sensordaten aus vergleichbaren Fertigungsprozessen ein. Zu Beginn des Prozesses werden die über die Zeit gesammelten Sensordaten (historische Sensordaten) zur Generierung von Wissen verwendet. Diese Sensordaten werden später ebenfalls zum Training des maschinellen Lernmodelles verwendet. Dabei ist zu erwähnen, dass diese Daten stets durch neue aufgenommene

Daten ergänzt werden können. Daher verläuft der Datenflusspfeil in Abbildung 8 von den aktuellen Sensordaten auch in den KDD-Teil. Dies macht Sinn, weil in den Informationen, welche dem maschinellen Lernen ursprünglich als Trainingsdaten bereitgestellt wurden, einige potenzielle Ausfallursachen nicht erfasst worden sein könnten. Diese Daten können zu den anderen gespeicherten historischen Daten in strukturierte Datenbanksysteme übertragen werden (vgl. Kapitel 2.1.2). Durch das Ergänzen der historischen Daten durch aktuelles und regelmäßiges Durchlaufen des KDD-Teils der Methode können neue Muster entdeckt werden, die als Features in das maschinelle Lernen fließen können. Daher ist eine kontinuierliche Generierung von neuem Wissen, das zur Anpassung und Verbesserung der maschinellen Lernmodelle genutzt werden kann, empfehlenswert. Dies ermöglicht die Entdeckung bisher unentdeckter Zusammenhänge oder die Anpassung an sich ändernde Bedingungen. Hierbei ist auf das Risiko hinzuweisen, dass maschinelle Lernmodelle durch das regelmäßige Nachtrainieren mit aktuellen Daten schlechter im Erkennen von alten Mustern werden könnte. Daher sollte beim regelmäßigen Nachtrainieren des Vorhersagemodells neben den neuen Daten historische Daten von alten Maschinenausfällen mit einbezogen werden.

Erst nachdem ein zuverlässiges maschinelles Lernmodell erstellt wurde, sollen die aktuell aufgenommenen Sensordaten je nach Zielstellung als eine Art Live-Daten möglichst zeitnah und konsistent in den Algorithmus zur Vorhersage integriert werden. Dabei ist wie in der Beschreibung zur Phase „Maschinelles Lernen“ erwähnt eine grobe Datenaufbereitung für die Vorhersagen erneut unerlässlich, um die korrekte Datenstruktur, korrekte Beschriftungen für überwachte Algorithmen sowie die Vermeidung von stark verrauschten Daten sicherzustellen.

Des Weiteren ist die in der Phase „Prozessverständnis“ erwähnte kontinuierliche Integration von Domänen- und Prozesswissen in das Vorgehensmodell über alle Phasen hinweg von entscheidender Bedeutung. Dies ermöglicht eine effiziente Nutzung von Zeit und Ressourcen, beispielsweise durch die Vermeidung der Generierung bereits bekannten Wissens. Insbesondere bei der erstmaligen Einführung eines prädiktiven Wartungssystems in einem produzierenden Unternehmen und der damit einhergehenden erstmaligen Anwendung der Methode ist die Einbindung von Experten von essenzieller Bedeutung. In der Prozessverständnisphase können Experten, beispielsweise erfahrenes Wartungspersonal, zum Gesamtprozessverständnis beitragen und somit dazu beitragen, die mit der Einführung der prädiktiven Wartung aus technischer oder wirtschaftlicher Sicht verfolgten Ziele zu konkretisieren. Dies kann beim Wartungspersonal zum Beispiel durch gesammelte Erfahrungen mit bestimmten Maschinen erfolgen. Datenexperten können zudem eine Bewertung bezüglich des in der Wissensgenerierungsphase

ermittelten Wissens vornehmen und dieses Wissen wie in Abbildung 8 dargestellt zurück in den KDD-Prozess geben oder anderweitig nutzen. Die Entstehung von Wissen ist stets mit einer menschlichen Beteiligung verbunden (vgl. Kapitel 2.1.1), die somit durch die Bewertung von Experten erfüllt ist.

Funktional lässt sich die Methode in eine je nach Zielstellung des Data Mining explorative oder deskriptive und anschließend prädiktive Phase unterteilen. Im ersten Teil erfolgt eine Generierung von Wissen aus historischen Sensordaten mit dem Ziel, nützliche Informationen für die nachfolgende Prädiktionsphase zu finden. Die Erkenntnisse aus dem KDD können dem Anwender wertvolle Informationen geben, die durch ein reines Prädiktieren durch maschinelles Lernen, wie es in Kapitel 2.2.3 beschrieben ist, nicht ersichtlich gewesen wären. Der erste Teil der Methode soll also unbekannte Zusammenhänge über den Fertigungsprozess und Ausfallrisiken aufdecken. Darüber hinaus kann dieses Wissen die Grundlage für die Anwendung anschließenden maschinellen Lernens im zweiten Teil der Methode darstellen, welches auf Basis aktueller Sensordaten Prognosen trifft.

Nachdem der Vorgehensrahmen einer möglichen datenbasierten prädiktiven Wartungsstrategie aufgestellt wurde, wird im Folgenden genauer erörtert, welche Vorzüge die Methode im Vergleich mit einer alleinigen Anwendung von maschinellem Lernen bietet. Der Schwerpunkt der Betrachtung liegt dabei auf den potenziellen Vorteilen, die sich durch KDD als Vorschritt für das maschinelle Lernen ergeben.

3.3 Mehrwert der entwickelten Methode

Die im Folgenden erörterten potenziellen Vorzüge der entwickelten Methode sind insbesondere eine Antwort auf die in Kapitel 2.4.3 herausgearbeiteten Herausforderungen, mit denen Unternehmen bei der Implementierung prädiktiver Wartung in der Praxis konfrontiert sind.

Im Rahmen der Methodenentwicklung sind bereits folgende Vorteile durch die Methode angeklungen, die in diesem Kapitel näher ausgeführt werden:

- Entscheidungshilfe für die Auswahl geeigneter maschineller Lernalgorithmen
- Tieferes Verständnis von Zusammenhängen für die Anwender als bei Durchführung reinen maschinellen Lernens
- Gezielte Bereitstellung von KDD-Wissen als Features für das maschinelle Lernen zur Verbesserung der Aussagekraft von potenziellen Maschinen-/Anlagenausfällen
- Validieren und Testen des maschinellen Lernalgorithmus

Der erste Teil der entwickelten Methode bietet, angelehnt an das CRISP-DM-Vorgehensmodell, einen strukturierten Rahmen, um durch Data Mining Informationen über Zusammenhänge, die zu Maschinenausfällen führen zu erhalten. Anstatt der reinen Vorhersage von Ausfallzeitpunkten durch das maschinelle Lernen erhält der Anwender also zusätzliches Wissen über nicht direkt ersichtliche Muster, Korrelationen und Beziehungen zwischen den durch Sensoren gemessenen Parametern. Der erste Teil der entwickelten Methode ermöglicht es also, die Sensordaten systematisch zu durchleuchten und Zusammenhänge zu entdecken, die bei einer einfachen Vorverarbeitung der Sensordaten für das maschinelle Lernen nicht gefunden worden wären. Durch das dem maschinellen Vorhersagealgorithmus vorgeschaltete Data Mining werden somit die vorhergesagten Ausfälle für den Anwender besser nachvollziehbar und interpretierbar als durch die alleinige Anwendung von maschinellem Lernen.

Im Folgenden wird als weiterer möglicher Vorteil der Kombination von Data Mining und maschinellem Lernen zunächst erörtert, auf welche Weise auf der Grundlage der vorliegenden Daten, die am besten geeignete Modelle des maschinellen Lernens ausgewählt werden können. Im Anschluss erfolgt eine Betrachtung weiterer potenziell nutzbringender Vorteile, die durch die Methode generiert werden können.

Entscheidungshilfe für die Auswahl geeigneter maschineller Lernalgorithmen

In der Beschreibung der Methode ist bereits im Rahmen des Prozessverständnisschrittes eine Eingrenzung des maschinellen Lernbereichs und der damit in Frage kommende Algorithmen empfohlen worden, um in der Modellierungsphase gezielter nach Wissen suchen zu können, das für eine spezifische Art von maschinellen Lernalgorithmen geeignet ist. Sofern jedoch keine hinlänglichen Kenntnisse über die Eignung spezifischer maschineller Lernalgorithmen für die betreffende Problemstellung vorhanden sind, kann auch wie in der Beschreibung der Phase „Maschinelles Lernen“ erwähnt, auf Grundlage des aus dem KDD-Teil erworbenen Wissens eine Auswahl der am besten geeigneten Algorithmen erfolgen.

Die Erkenntnisse, welche durch die Anwendung des ersten Teils der entwickelten Methode auf die historisch gesammelten Sensordaten gewonnen werden, können einen Mehrwert bei der Auswahl des am besten geeigneten maschinellen Lernalgorithmus darstellen. Data-Mining-Techniken ermöglichen beispielsweise die Identifikation von nicht-linearen und komplexen Beziehungen sowie linearen Mustern und einfachen Beziehungen zwischen betrachteten Features, welche für die RUL-Prognose oder Zustandsbestimmung herangezogen werden. Bei einer Vielzahl von nicht-linearen Mustern kann beispielsweise der Einsatz des Random-Forest-Algorithmus in Betracht gezogen

werden, wobei die jeweilige Zielstellung zu berücksichtigen ist. Dieser Algorithmus nutzt eine größere Anzahl an Entscheidungsbäumen, um komplexe Beziehungen zu modellieren (vgl. Kapitel 2.3). Werden demgegenüber überwiegend lineare Beziehungen zwischen Variablen identifiziert, so führt eine Veränderung von Features zu einer proportionalen Veränderung des Zielwerts. Folglich erweisen sich lineare Methoden wie die lineare Regression oder Support-Vector-Machines mit linearem Kernel als besser geeignet. Die Anwendung von Data Mining im Rahmen der entwickelten Methode auf die relevanten Daten könnte darüber hinaus zu der Erkenntnis führen, dass die Daten eine hohe Dimensionalität aufweisen. Auch mit diesem Wissen könnte man zur Auswahl der Random-Forest-Algorithmus kommen, da dieser eher robust gegenüber hochdimensionalen Daten ist. Dies liegt darin begründet, dass jeder der genutzten Entscheidungsbäume eine zufällige Auswahl von Features verwendet. In der Konsequenz wird die Dimensionalität der betrachteten Daten reduziert, wobei die zufällige Auswahl eine gegen Überanpassung wirkende Komponente darstellt (vgl. Kapitel 2.3).

Tieferes Verständnis von Zusammenhängen für den Anwender

Die Datenbeschreibung und Zusammenfassung wird im Benutzerhandbuch des CRISP-DM-Vorgehensmodells als eine mögliche Data-Mining-Aufgabe erachtet (vgl. Kapitel 2.1.1). Im Kontext der prädiktiven Wartungsmethode kann beispielsweise mittels explorativer Datenanalyse die Suche nach Mustern und Trends in großen Datenmengen erfolgen, um bislang unentdeckte Beziehungen aufzudecken oder Annahmen zu treffen. Dazu können klar definierte statistische Verfahren wie die Datenvisualisierung, die deskriptive Statistik, die Korrelationsanalyse, Häufigkeitstabellen und ähnliche Methoden eingesetzt werden, um systematisch Beziehungen zwischen Variablen zu ermitteln (vgl. Kapitel 2.1.4). So könnte beispielsweise eruiert werden, welche Produktvarianten oder Kombinationen von Produktvarianten auf einer Produktionslinie eine höhere Anzahl an Alarmen auslösen. Diese Information kann an das maschinelle Lernprogramm zur Aussage von Ausfällen weitergegeben oder aber auch anderweitig detaillierter untersucht werden, um die Gründe für das Auslösen von Alarmen durch bestimmte Produkte zu ermitteln. Da der Prozess des maschinellen Lernens auf automatisierter Weise auf Basis von Daten erfolgt (vgl. Kapitel 2.3), sind die genauen Zusammenhänge, die zu Vorhersagen führen, häufig nicht nachvollziehbar. Die vorherige Identifikation der Muster und Zusammenhänge mittels Data Mining sowie die Bereitstellung dieser Informationen an das maschinelle Lernen ermöglichen eine signifikante Steigerung des Vertrauens in die Vorhersage, da die zugrunde liegenden Zusammenhänge bekannt sind.

Unterstützung im Feature Engineering

Das Hervorheben wichtiger entdeckter Zusammenhänge, die Bereitstellung von aktualisierten Erkenntnissen aus neueren Sensordaten und die Kompression der Trainingsdaten für das maschinelle Lernmodell kann durch die Bildung von Features erfolgen. Es bestehen Möglichkeiten in der Generierung neuer, sinnvoller Features, der Identifikation relevanter Features sowie der Transformation bestehender Features als vorverarbeiteter Input für das maschinelle Lernen. Der Erfolg des maschinellen Lernens ist in vielen Fällen maßgeblich von der Effektivität des Feature-Engineerings abhängig (vgl. Kapitel 2.3). Ein wesentlicher Aspekt bei der Anwendung von Features ist die Relevanz der enthaltenen Informationen. Diese sind von entscheidender Bedeutung für die Funktionalität des maschinellen Lernmodells, da dieses die bereitgestellten Features für die Trainingszwecke nutzt und somit auf deren korrekter Funktionsweise aufbaut. Die bereitgestellten Features können folglich einen direkten Einfluss auf die Ergebnisse der maschinellen Lernmodelle ausüben, da sie das zugrunde liegende Problem adäquater darstellen als die reinen Daten (vgl. Kapitel 2.3). Die Ziele des Feature Engineerings sind vielfältig. Sie können die Erfassung nichtlinearer Zusammenhänge, die Vermeidung von Überanpassung, eine verbesserte Generalisierbarkeit oder ähnliches umfassen (vgl. Kapitel 2.3). So könnten im Rahmen der Methode einfache statistischer Features erstellt werden, die Minimum, Maximum, Median, Standardabweichung, Länge des Datensatzes oder Häufigkeit der Datenaufnahme repräsentieren. Die Relevanz der bereitgestellten Features für die Performance des maschinellen Lernmodells könnte jedoch ein Argument dafür sein, den ersten Teil der Methode auf das Feature Engineering zu fokussieren und aufwendigere Merkmale zu erzeugen. Die entwickelte Methode fungiert dabei als strukturierter Rahmen. Das Einbeziehen von Expertenwissen und einem Geschäftsverständnis erweist sich insbesondere bei der Neueinführung von prädiktiver Wartung in produzierenden Unternehmen als vorteilhaft, da Experten durch Erfahrung mit der Anlage und ähnlichen Fällen bereits über ein grundlegendes Verständnis von Zusammenhängen, die zu Ausfällen führen könnten, verfügen. In der Konsequenz erlaubt das Feature-Engineering eine gezieltere Ausgestaltung sowie die Adressierung spezifischer Aspekte der Maschinenwartung, gesteuert durch das Expertenwissen. Wird beispielsweise durch die Anwendung einer Clusteranalyse im Data Mining ein Zusammenhang zwischen Parametern wie einer hohen erfassten Luftfeuchtigkeit in Kombination mit einem bestimmten Vibrationsmuster als starker Hinweis auf einen bevorstehenden Ausfall identifiziert, so kann dieser Zusammenhang in einem Feature dargestellt werden. Inhaltlich kann das Feature aus mathematischen Operationen oder der Berechnung repräsentativer Zeitwerte bestehen. Als repräsentative Größe bietet sich die Berechnung gleitender Mittelwerte über einen bestimmten Zeitraum an, um kurzfristige

Trends zu erfassen. In Bezugnahme auf das angeführte Beispiel lässt sich festhalten, dass eine zunehmende Luftfeuchtigkeit in Kombination mit einem steigenden Vibrationswert auf einen bevorstehenden Ausfall hindeuten kann. Diesbezüglich kann auch eine einfache Multiplikation beider Parameter aussagekräftig sein. Eine weitere Möglichkeit besteht in der Ausgabe des Ergebnisses des Features als binärer Wert, welcher eine Änderung erfährt, sobald beide Parameter ihren jeweiligen kritischen Wert überschritten haben (Guyon und Elisseeff 2006).

Auch nach der Modellierung eines Vorhersagemodells erweist sich die entwickelte Methode aufgrund ihres iterativen Charakters weiterhin als geeignet. Die Möglichkeit, in jeder Phase des Prozesses zurückzukehren, erlaubt die Einbeziehung von Erkenntnissen aus dem Data Mining in die Features, um bei Bedarf eine Anpassung und Verbesserung der Featureauswahl und -erstellung zu ermöglichen. Sofern durch Data-Mining-Methoden wie die Assoziationsanalyse Korrelationen und Abhängigkeiten von Variablen oder das Clustering Gruppierungsstrukturen in den Daten identifiziert wurden, können diese Informationen in die anschließend dem maschinellen Lernen bereitgestellten Features integriert werden. Dadurch können bestimmte Zusammenhänge in den Daten betont werden, die im Rahmen des maschinellen Lernens ansonsten möglicherweise nicht hinreichend erfasst worden wären. Selbst wenn das maschinelle Lernmodell die Beziehungen selbst gelernt hätte, könnte das Feature Engineering in der Lage sein, das Lernen zu beschleunigen und präziser zu gestalten, indem es diese wichtigen Informationen direkt zugänglich macht.

Neben dem Extrahieren wichtiger Beziehungen durch Data Mining zur Verbesserung von Features kann als weitere Data-Mining-Aufgabe auch die Bestimmung der Relevanz einzelner Features bewertet werden. Zu diesem Zweck stehen verschiedene Techniken zur Verfügung, darunter die Feature-Importance-Analyse (Kozue 2023). Die Bewertung eines Features erfolgt anhand der Korrelation mit der Zielvariable. Ein Feature wird als nützlich erachtet, wenn eine hohe Korrelation festzustellen ist.

Da eine hohe Dimensionalität und Komplexität an Daten Nachteile mit sich bringen kann (vgl. Kapitel 2.2.2) scheint auch die Identifikation weniger relevanter Features als mehrwertstiftend. Die als unwichtig identifizierten Features können aus dem Datensatz gelöscht werden, um die Modellkomplexität zu vereinfachen. Die Abbildung der Merkmale auf einen Raum mit geringerer Dimensionalität kann zu einer Beschleunigung der Reaktionszeit maschineller Lernalgorithmen sowie zu einer Erhöhung der Vorhersagegenauigkeit führen.

Der Fokus auf das Feature Engineering dient demnach der Identifikation der optimalen Darstellung historischer, von Sensoren generierter Daten, mit denen maschinelle Lernmodelle trainiert werden. Infolgedessen startet das maschinelle Lernen nicht "blind", sondern mit einer wissensbasierten Grundlage, wodurch der Trainingsprozess effizienter und schneller gestaltet werden kann. Im Idealfall resultiert dies in besseren Vorhersagen.

Validieren und Testen des maschinellen Lernalgorithmus

Die Evaluierung der Prognosen zum Ausfallverhalten unter Einsatz maschinellen Lernens erweist sich in der praktischen Anwendung als anspruchsvoll. Die Prognose von Ausfällen auf Basis maschinellen Lernens ermöglicht die Umsetzung vorbeugender Wartungsmaßnahmen. Allerdings lässt sich nach der durchgeführten Instandhaltung nicht eruieren, ob ein tatsächlicher Ausfall der Maschine oder Anlage in naher Zukunft erfolgt wäre oder nicht. Evident sind nur positiv-falsch-Ergebnisse, also ein Anlagenausfall, der durch das maschinelle Lernen nicht vorhergesagt worden ist. Wollte man sichergehen, dass die prognostizierten Ausfälle auch tatsächlich eintreten, müsste man auf vorbeugende Instandhaltungsmaßnahmen verzichten und einen möglichen Ausfall in Kauf nehmen.

Zur Bewertung der Vorhersagekraft ist jedoch auch die iterative Wechselwirkung mit der KDD-Phase der entwickelten Methodik geeignet. Dies kann durch die Verwendung historischer Daten und bekannter Muster erfolgen. Werden bekannte Daten, die vor einem bevorstehenden Maschinenausfall aufgezeichnet wurden, in das maschinelle Lernmodell eingespeist, so sollte dieses in der Lage sein, den Ausfall vorherzusagen. Wenn das maschinelle Lernmodell bestimmte Muster nicht richtig erkennt oder falsche Vorhersagen trifft, kann es durch Anpassung der Hyperparameter oder der Trainingsdaten verbessert werden. Auch mit diesem Ansatz lassen sich die aktuellen Vorhersagen nicht verifizieren, aber man erhält einen guten Überblick darüber, wie gut historische Ausfälle vorhergesagt worden wären.

Zusammenfassend soll die entwickelte Vorgehensmethodik als Entscheidungshilfe bei der Auswahl bzw. Validierung geeigneter maschineller Lernalgorithmen sowie zur Lösung der in Kapitel 2.4.3 identifizierten Datenprobleme eingesetzt werden. Durch die verschiedenen Möglichkeiten der Nutzung der Wissensgenerierungsphase, wie zum Beispiel Strukturierung der Daten und Identifikation von Zusammenhängen, Reduktion der Dimensionalität oder Generierung von Features, sollen letztendlich genauere, robustere und besser interpretierbare Vorhersagen über den Ausfall einer Maschine und darauf basierende Instandhaltungsentscheidungen getroffen werden. Ob diese Methode

bessere Vorhersageergebnisse liefert als die alleinige Anwendung von maschinellem Lernen auf Sensordaten, wird im nächsten Kapitel untersucht.

4 Testen der entwickelten Methode

Die detaillierte Betrachtung jeder einzelnen Phase der entwickelten Methode anhand des in diesem Kapitel behandelten Anwendungsbeispiels würde den Umfang dieser Arbeit übersteigen. Daher liegt der Fokus im Folgenden vor allem auf einem wichtigen Kernaspekte, nämlich der Anwendung von Data-Mining-Verfahren vor der Vorhersage durch maschinelles Lernen.

Zur exemplarischen Untersuchung der identifizierten Vorteile durch die vorherige Anwendung von Data Mining vor dem maschinellen Lernen zum Vorhersagen von Maschinenausfällen, wird in diesem Kapitel zunächst der verwendete Datensatz beschrieben und anschließend ein maschineller Lernalgorithmus auf einen Datensatz angewendet. Zum Zwecke des Vergleichs wird derselbe Datensatz auf denselben maschinellen Lernalgorithmus angewendet, wobei jedoch zuvor dem Data Mining zuzuordnende Verfahren zum Einsatz kommen. In Bezug auf die im vorherigen Kapitel entwickelte Methode stehen somit die Phasen "Modellierung", "Feature-Generierung" und "Maschinelles Lernen" im Fokus der Betrachtung.

4.1 Datensatz

Da der Schwerpunkt auf der vergleichenden Untersuchung von maschinellem Lernen und maschinellem Lernen mit vorherigem Data Mining liegen soll, wurde ein bereits vorverarbeiteter synthetischer Datensatz ausgewählt, der das Verhalten realer Industriemaschinen widerspiegeln soll (UCI Machine Learning Repository 2020). Infolgedessen kann im nachfolgenden Anwendungsbeispiel auf umfangreichere Datenvorverarbeitungsschritte, wie beispielsweise das Auffüllen von fehlenden Werten, verzichtet werden. Der Datensatz umfasst 10.000 Einträge, die jeweils 14 Spalten aufweisen. Jeder Datensatz ist mit einer eindeutigen Identifikationsnummer (UDI) bzw. Product-ID sowie der Information, ob zum jeweiligen Zeitpunkt ein Maschinenausfall eingetreten ist oder nicht, versehen. In 96,6 % der Fälle liegt ein fehlerfreier Maschinenzustand und in 3,4 % ein fehlerhafter Maschinenzustand vor. Des Weiteren umfasst der Datensatz Angaben zum betrachteten Maschinentyp sowie durch Sensoren erfasste Informationen zur Lufttemperatur, Prozesstemperatur, Rotationsgeschwindigkeit, zum Drehmoment und zum Werkzeugverschleiß. Neben der allgemeinen binären Angabe, ob ein Maschinenfehler vorliegt, lässt sich der genaue Ausfallgrund durch weitere binäre Angaben zu den Kategorien "Fehler durch Werkzeugverschleiß" (TWF), "Überhitzung" (HDF), "Probleme mit der Stromversorgung" (PWF), "Überlastung" (OSF) oder "zufälliger Fehler" (RNF) bestimmen. Der Datensatz eignet sich in exemplarischer Weise für die Anwendung der in Kapitel 3 entwickelten Methode, da die Daten mit eindeutigen IDs und Informationen

über Maschinenausfälle gelabelt sind und daher für die getroffene Einschränkung von überwachten Lernmethoden geeignet scheinen. Des Weiteren entspricht die in Kapitel 3 beschlossene Fokussierung auf die sensorbasierte prädiktive Wartung den in dem Datensatz angegebenen Prozessparametern, welche aus Sensormesswerten bestehen und als Eingabedaten genutzt werden sollen, um einen Maschinenfehler vorherzusagen.

4.2 Anwendung der Methode

In Kapitel 2.4.1 sind die Regression und Klassifikation als geeignete maschinelle Lernmethoden im Rahmen der prädiktiven Wartung beschrieben worden. Für das Anwendungsbeispiel wird daher exemplarisch ein Klassifizierungsproblem aus dem Bereich überwachtes maschinelles Lernen behandelt. Der vorliegende Datensatz eignet sich grundsätzlich für eine Mehrklassen-Klassifikation, bei der neben dem Zeitpunkt auch der genaue Ausfallgrund vorhergesagt werden soll (vgl. Kapitel 2.4.1). Der gesamte Programmcode des im Rahmen dieser Arbeit in Python entwickelten Modells ist im Anhang zu finden und wird nachfolgend auszugsweise beschrieben.

Die Spalten mit den Ausfallgründen TWF, HDF, PWF, OSF und RNF wurden zu Beschränkung auf eine binäre Klassifizierung zur Vorhersage eines defekten oder funktionierenden Maschinenzustandes zunächst aus dem Datensatz entfernt. Um eine Grundlage für einen Vergleich der entwickelten Methode zur Anwendung von Data Mining und anschließendem maschinellen Lernen zu schaffen, wurde zunächst maschinelles Lernen ohne Data Mining durchgeführt. Wie in Kapitel 4.1 dargelegt, ist der verwendete Datensatz bereits vollständig und enthält plausible Werte, sodass auf eine umfassende Vorverarbeitung der Sensordaten verzichtet werden kann. Im Rahmen der Vorverarbeitung wird lediglich, wie der Programmausschnitt Algorithmus 1 zeigt, das One-Hot-Encoding für die kategoriale Variable "Maschinentyp" ausgeführt.

Algorithmus 1: Auszug aus dem Python-Programmcode für Modell 1

```
. # One-Hot-Encoding für kategoriale Variablen
. df = pd.get_dummies(df, columns=['Type'], drop_first=True)
. # Modellierung mit Random Forest
. X = df.drop(columns=['Machine failure'])
. y = df['Machine failure']
. # Train-Test-Split
. X_train, X_test, y_train, y_test = train_test_split(X, y,
          test_size=0.2, stratify=y, random_state=42)attributes
```

```
            = list(data.columns)
. # SMOTE nur auf den Trainingsdaten anwenden
. smote = SMOTE(random_state=42)
. X_train_res, y_train_res = smote.fit_resample(X_train,
            y_train)
5. # Random Forest Classifier
6. clf_rf = RandomForestClassifier(random_state=42)
```

Dadurch wird für jeden Maschinentyp eine eigene Spalte generiert, die mit binären Werten gefüllt wird. Diese binären Werte können vom maschinellen Lernmodell im Gegensatz zu kategorialen Variablen verarbeitet werden. Die Zielvariable des zugrunde liegenden Klassifizierungsproblems ist die Angabe eines potenziellen Maschinenfehlers, wobei der Wert "0" keinen Fehler und der Wert "1" einen fehlerhaften Zustand indiziert. Im Anschluss erfolgt eine Aufteilung der Daten in ein Trainings- und Testset im Verhältnis von 80 zu 20. Um dem Problem der geringen Häufigkeit von Maschinenausfällen im Vergleich zu funktionierendem Maschinenzustand im Datensatz entgegenzuwirken, findet, wie in Algorithmus 1 angewendet, die Synthetic Minority Over-sampling Technique (SMOTE) Anwendung. Mit SMOTE soll verhindert werden, dass das maschinelle Lernmodell überwiegend „keinen Maschinenausfall“ vorhersagt, nur weil dies die dominante Klasse in dem Datensatz darstellt. Basierend auf dem identifizierten Verteilungsmuster der Minderheitsklasse werden synthetische Datenpunkte für die Minderheitsklasse erzeugt. Durch das stärkere Repräsentieren der Minderheitsklasse durch die synthetischen Daten sollen mehr Daten für das Lernen der Minderheitsklasse bereitstehen, damit das Modell aufgrund der unausgewogenen Verteilung nicht immer die Mehrheitsklasse voraussagt.

Für die eigentliche Vorhersage der Klassifizierung von Maschinenzuständen in "fehlerhaft" oder "nicht fehlerhaft" kommt der Random-Forest-Classifier zur Anwendung.

Im Rahmen der vorliegenden Untersuchung wird darüber hinaus eine Kreuzvalidierung über fünf Folds angewendet. Dies impliziert eine Aufteilung der Daten in fünf gleiche Teile sowie die Nutzung eines Teils der Daten als Test-Set in jedem Fold, während die übrigen vier Teile als Trainingsdaten dienen. Wie in Kapitel 2.3 beschrieben, zielt dieses Vorgehen darauf ab, die Leistungsfähigkeit des Modells zu stabilisieren, indem es nicht nur auf einen spezifischen Teil der Daten, sondern auf die verschiedenen Folds getestet wird Im Anschluss an die Berechnung des F1-Scores (vgl. Kapitel 2.3) zur Einschätzung

der Modellleistung über die verschiedenen Folds erfolgt das Training des Modells auf den gesamten Trainingsdatensatzes. Die Evaluierung der Vorhersagegüte des trainierten Modells auf ungesehene Daten erfolgt schließlich durch die Anwendung auf den separaten Testdatensatz.

Erweiterung um Data Mining (Modell 2)

Für die exemplarische Anwendung der in Kapitel 3 entwickelten Methode wurden die im ersten Teil dieses Kapitels vorgestellten Schritte übernommen und in einem zweiten Modell um vorgelagerte Data-Mining-Funktionen ergänzt (siehe Anhang: Programmcode: Random-Forest mit vorherigen Data Mining). In diesem Rahmen wird zunächst eine Korrelationsanalyse durchgeführt, um die Beziehung zwischen den verschiedenen sensorisch gemessenen Parametern und deren Auswirkung auf einem möglichen Maschinenausfall zu untersuchen. Dafür wird, wie in Algorithmus 2 geschrieben, die Pearson-Korrelation aller numerischen Spalten in dem Datensatz mit der Zielvariable Maschinenausfall in Python mittels des Befehls „*df.corr()*“ berechnet und in einer Korrelationsmatrix ausgegeben.

Algorithmus 2: Auszug aus dem Python-Programmcode für Modell 2

```
. # Data Mining – Korrelationsanalyse
. correlation_matrix = df.corr()
. # Identifizieren der am stärksten korrelierten Features
. correlated_features = correlation_matrix['Machine failu-
      re'].abs().sort_values(ascending=False)
```

Im Anschluss werden die zwei durch die Korrelationsanalyse als einflussreich identifizierten Parameter verwendet, um die Daten im Rahmen einer Clusteranalyse zu gruppieren. Das Clustern erfolgt unter Zuhilfenahme des KMeans-Algorithmus, wobei eine manuelle Angabe einer Anzahl von drei Clustern getroffen wird. Im Rahmen der Clusteranalyse werden die Datenpunkte hinsichtlich der zwei identifizierten Parameter dahingehend untersucht, ob sich Gruppen aus diesen beiden Parametern bilden lassen, die auf einen Ausfall oder einwandfreien Betrieb der Maschine hindeuten. Das ermittelte Cluster-Muster wird zur anschaulichen Darstellung in einem zweidimensionalen Diagramm präsentiert und im Kapitel 4.3 einer detaillierten Erläuterung unterzogen. Als weiterer Data Mining-Methode für einen vertiefenden Einblick in die Daten wird ein Entscheidungsbaum eingesetzt. Ziel ist die Identifikation von Schwellenwerten für die gemessenen Parameter, die auf einen Maschinenausfall hindeuten. Der Entscheidungsbaum wurde wie bei maschinellen Lernmodellen des Random Forest mit einer Aufteilung

der Daten zu Trainings- und Testzwecken und einer anschließenden Anwendung des DecisionTreeClassifier implementiert. Der Entscheidungsbaum beinhaltet neben dem Schwellenwert der Parameter für einen Maschinenausfall oder Nicht-Maschinenausfall auch Angaben zum Gini-Wert (vgl. Kapitel 2.3), der Anzahl der Proben (Samples), die in diesem Blatt enthalten sind, sowie der Klasse, die in dem Blatt am häufigsten vorkommt (Value).

Zusammenfassend werden im Data Mining Prozess zunächst Korrelationen zwischen den verschiedenen gemessenen Prozessparametern und der Zielvariable „Maschinenausfall“ berechnet. Aus der Liste der korrelierten Features werden diejenigen mit der höchsten Korrelation in Bezug auf einen Maschinenausfall für das Clustering ausgewählt. Im Rahmen des Clusterings werden die Features mittel KMeans-Clustering in Cluster eingeteilt, um Zusammenhänge zwischen den beiden ausgewählten Parametern zu entdecken. Des Weiteren erfolgt eine Mustererkennung mittels eines trainierten Entscheidungsbaums, um zusätzliche Zusammenhänge zwischen den Parametern zu identifizieren, die häufig zu Ausfällen führen. Zur Veranschaulichung der Entscheidungsfindung werden der Gini-Wert, die Anzahl der Proben und die Klassenzugehörigkeit für jedes Blatt ausgegeben.

Zur Integration des aus den Data-Mining-Schritten erworbenen Wissens ist, wie in Algorithmus 3 geschrieben, ein neues Feature namens „Combi_TopFeatures“ erstellt worden.

Algorithmus 3: Auszug aus dem Python-Programmcode für Modell 2

```
. # Feature Engineering - Neues Feature basierend auf den Top-
      Features
. df['Combi_TopFeatures'] = df[top_features[0]] * df[top_fea-
      tures[1]]
. # Entscheidungsbaum-Pfad als neues Feature
. first_feature = clf.tree_.feature[0]
. first_threshold = clf.tree_.threshold[0]
. df['Path_Node_1'] = (df[X.columns[first_feature]] <=
      first_threshold).astype(int)
```

Es stellt die Wechselwirkung zwischen den beiden als am wichtigsten identifizierten Features dar, indem es aus einer einfachen Multiplikation dieser erstellt wird. Das neu generierte Feature fließt als Input in die anschließende Anwendung des Random-Forest-Algorithmus ein, sodass ein gezielteres Training des Modells mit dem als wichtig identifizierten Feature erfolgt. Darüber hinaus wird ein weiteres Feature, welches das aus dem

Entscheidungsbaum erworbene Wissen beinhaltet, als Eingabe für Random Forest bereitgestellt (vgl. Algorithmus 3). Die Erstellung des Entscheidungsbaum-Features erfolgt auf Basis des ersten Splits des Entscheidungsbaums. Somit kann die Entscheidungslogik des Entscheidungsbaumes, basierend auf den ermittelten Schwellenwerten, direkt in Random Forest integriert werden.

4.3 Ergebnisse

In diesem Kapitel erfolgt die Darstellung der Resultate des in Kapitel 4.2 beschriebenen Vorgehens. Dabei wird zunächst auf die Ergebnisse aus den Data-Mining-Schritten Bezug genommen.

Die visualisierte Darstellung der durchgeführten Korrelationsanalyse der einzelnen Parameter mit der Zielvariable „Maschinenausfall“ ist in der Korrelationsmatrix in Abbildung 9 zu sehen.

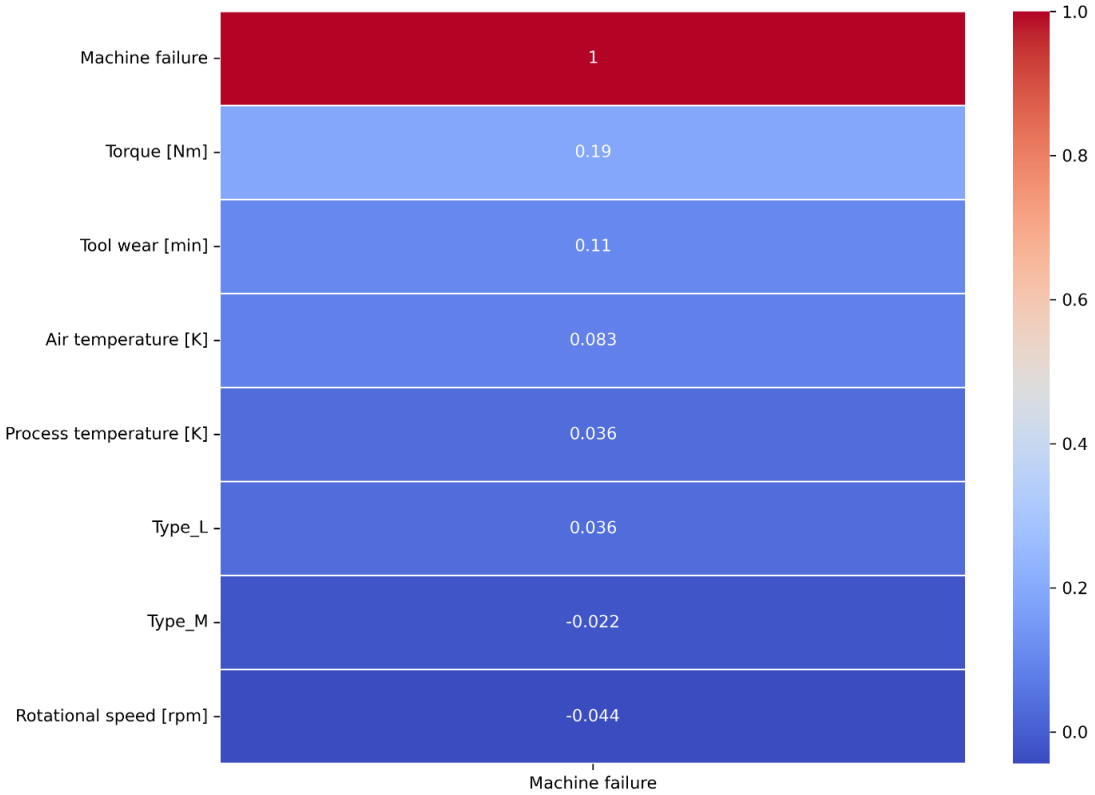


Abbildung 9: Korrelationsmatrix zur Darstellung des Zusammenhanges der Prozessparameter mit einem Maschinenausfall

Die Matrix veranschaulicht die paarweise Korrelation zwischen den Parametern des Datensatzes mit einem Maschinenausfall. Ein Wert von eins steht für einen proportionalen Zusammenhang beider Variablen und somit eine starke positive Korrelation. Eine Korrelation von null hingegen weist darauf hin, dass es keine lineare Korrelation zwischen

den Variablen gibt. Die Korrelationsmatrix zeigt, dass die beiden Parameter Drehmoment (Torque) und Werkzeugverschleiß (Tool wear) am stärksten mit einem Maschinenausfall zusammenhängen. Infolgedessen wurden diese beiden Parameter als einflussreichste Features für die Clusteranalyse selektiert. Wie in Abbildung 10 dargestellt, erfolgt bei der Clusteranalyse eine Gegenüberstellung des Parameters "Drehmoment" in Newtonmeter (Nm) auf der X-Achse und des Parameters "Werkzeugverschleiß" in Minuten auf der Y-Achse.

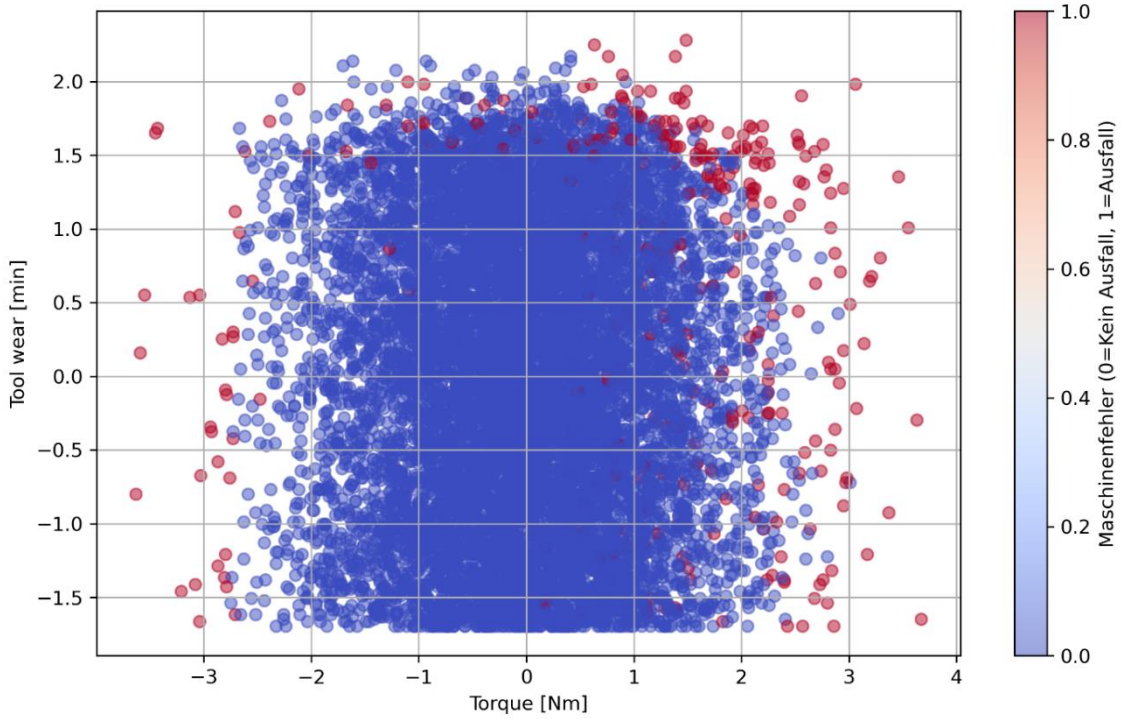


Abbildung 10: Clusterdarstellung zur Darstellung des Zusammenhangs zwischen den Parameterausprägungen Drehmoment und Werkzeugverschleiß mit einem Maschinenausfall

Die Achsenskalierung liefert dabei keine direkten, gemessenen Rohwerte, sondern zeigt die relativen Positionen der Features basierend auf einer standardisierten Skala. Aus der grafischen Darstellung lässt sich ableiten, wie weit die einzelnen Datenpunkte relativ zum Mittelwert vom Zentrum entfernt liegen und wie sie sich in Clustern anordnen. Es ist zu erkennen, dass die Datenpunkte mit Ausnahme weniger Ausreißer relativ gleichmäßig um die Mittelwerte der beiden betrachteten Parameter verteilt sind. Des Weiteren sind die Datenpunkte blau eingefärbt, sofern ein geringes Ausfallrisiko angenommen werden kann, während eine rote Färbung ein wahrscheinliches Ausfallrisiko indiziert. Die roten Farbpunkte sind überwiegend am Rande der Verteilung zu finden. Die größte Clustergruppe für einen wahrscheinlichen Maschinenausfall tritt bei einem im Vergleich zum Mittelwert sehr hohen Drehmoment und einem eher hohen Werkzeugverschleiß auf. Zusammenfassend lässt sich im Bezug auf die beiden Parameter sagen, dass das Risiko

für einen Maschinenausfall mit der Entfernung der gemessenen Werte vom Mittelwert zunimmt.

Als dritte Data-Mining-Methode zur Erkennung von Mustern und Zusammenhängen in dem Datensatz vor Anwendung des maschinellen Lernens wurde ein Entscheidungsbaum eingesetzt, dessen Ausschnitte in Abbildung 11 dargestellt sind. Die Farbgebung des Wurzelknotens und Blattknotens erfolgt anhand der Klassifikation "Maschinenausfall", welche durch die Farbe Blau repräsentiert wird, sowie "kein Maschinenausfall", welche durch die Farbe Orange dargestellt wird. Sind die Verteilungen der beiden Klassen nicht eindeutig, wird eine Mischfarbe dargestellt.

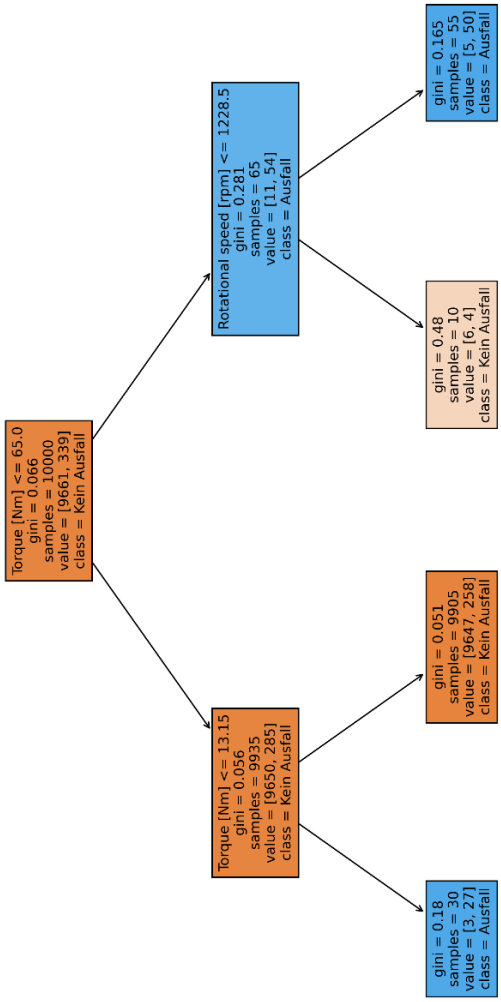


Abbildung 11: Entscheidungsbaumausschnitt zur Darstellung des Zusammenhangs zwischen Maschinenausfällen und Schwellenwerten der Parameter Drehmoment und Rotationsgeschwindigkeit

Auch beim Entscheidungsbaum wurde das Drehmoment als einflussreichster Parameter als Betrachtungsausgangspunkt und damit als Wurzelknoten gewählt. Infolgedessen wird das Drehmoment als wesentlichster Parameter für die Vorhersage eines Maschinenausfalls erachtet. Die Angaben im Wurzelknoten legen nahe, dass Maschinen mit einem Drehmoment von weniger als 65 Nm mit hoher Wahrscheinlichkeit nicht ausfallen. Des Weiteren lässt sich erkennen, dass bei diesem Drehmomentwert 9661 Beispiele aus dem Gesamtdatensatz der Klasse "kein Ausfall" und 339 der Klasse "Ausfall" zugeordnet werden. Der Gini-Index von 0,066 zeigt, dass die Unreinheit in diesem Blattknoten relativ gering ist. Auf der ersten Ebene des Entscheidungsbaums wird sodann der Schwellenwert für das Drehmoment verfeinert. Bei einem Drehmoment von unter 13,15 Nm fällt weiterhin eine Anzahl von 9650 Maschinen nicht aus. Sofern die Bedingung "Drehmoment < 65 Nm" nicht erfüllt ist, ist der rechte Blattknoten für Werte > 65 Nm zu betrachten. Der vorliegende Datensatz zeigt, dass bei einer Drehzahl von unter 1228,5 Umdrehungen pro Minute die Wahrscheinlichkeit für einen Ausfall steigt. Es ist jedoch zu berücksichtigen, dass die absolute Anzahl an Beispielen mit 11 nicht ausgefallenen und 54 ausgefallenen Maschinen vergleichsweise gering ist. Der Gini-Index von 0,281 weist zudem auf eine höhere Durchmischung zwischen nicht ausgefallenen und ausgefallenen Maschinen hin.

Nach der Betrachtung der Resultate des Data Minings erfolgt nun der Vergleich zwischen der Performance des reinen Random-Forest-Modells (Modell 1) mit der des erweiterten Modells, bei dem durch Feature-Generierung Wissen aus dem Data-Mining einfließt (Modell 2). In dem durchgeführten Anwendungsbeispiel zeigt sich, dass der F1-Score beim reinen maschinellen Lernmodell mit 0,9774 und beim um Data Mining ergänzten Modell mit 0,9757 nahezu identisch ist (vgl. Tabelle 3). Die Standardabweichung des ersten Modells ist dabei jedoch etwas geringer, was darauf hindeutet, dass die Vorhersagen konsistenter sind als beim erweiterten Modell.

Tabelle 3: Kennzahlen zur Vorhersageleistung von Modell 1 und Modell 2

Kennzahl	Modell 1	Modell 2
F1-Score (Standardabweichung)	0,9774 (0,0025)	0,9757 (0,0032)
MSE	0,0440	0,0310

Demgegenüber weist das um Data Mining ergänzte Modell einen geringeren MSE-Wert auf, was darauf hindeuten kann, dass es weniger fehlerträchtige Vorhersagen trifft. Die exakten Vorhersageergebnisse, basierend auf einem Datenauszug von 2000

Datenpunkten, lassen sich durch einen Blick auf die Confusion Matrizen (vgl. Kapitel 2.3) in Tabelle 4 und 5 erläutern.

Tabelle 4: Confusion Matrix zum Vergleich der Vorhersage von Modell 1 mit den tatsächlich eingetretenen Ereignissen

Modell 1	Vorhersage „Kein Ausfall“	Vorhersage „Ausfall“
„Kein Ausfall“ eingetreten	1864	68
„Ausfall“ eingetreten	20	48

Tabelle 5: Confusion Matrix zum Vergleich der Vorhersage von Modell 2 mit den tatsächlich eingetretenen Ereignissen

Modell 2	Vorhersage „Kein Ausfall“	Vorhersage „Ausfall“
„Kein Ausfall“ eingetreten	1895	37
„Ausfall“ eingetreten	25	43

Die Auswertung der Ergebnisse zeigt, dass das erweiterte Modell 31 Fälle mehr als korrekt eingestuft (1895 im Vergleich zu 1864) und 31 Maschinen weniger als fälschlicherweise als Ausfall klassifiziert (37 gegenüber 68). Für die Vorhersage der Klasse "Ausfall" zeigt sich hingegen eine leicht geringere Performance des erweiterten Modells (43 gegenüber 48 richtigen Vorhersagen), wobei zudem fünf Maschinen fälschlicherweise nicht als Ausfall identifiziert wurden (25 statt 20).

5 Diskussion und Fazit

In Kapitel 4 wurde die Performance eines Random-Forest-Vorhersagemodells, stellvertretend für ein maschinelles Lernmodell, welches Produktionsdaten als Eingabe erhält, mit der eines Random-Forest-Vorhersagemodells verglichen, welches neben den Produktionsdaten zusätzliche, auf Data-Mining-Wissen basierende Merkmale als Eingabe erhält. Mit dem Anwendungsbeispiel sollte exemplarisch die in Kapitel 3 entwickelte Methode bewertet werden. Der erste Teil der Methode basiert auf dem CRISP-DM-Vorgehensmodell, welches auf den spezifizierten Bereich der sensorbasierten prädiktiven Wartung angewendet wird. Die Auswahl des CRISP-DM-Vorgehensmodells erfolgte anhand der Darstellung von Gemeinsamkeiten in KDD-Vorgehensmodellen, vor allem basierend auf Vergleichen wie von Mariscal et al. (2010), um später eines der beiden Vorgehensmodelle für die Methode auszuwählen. In der Literatur finden sich abgesehen von den beiden im Fokus betrachteten Vorgehensmodellen zahlreiche weitere Vorschläge für strukturierte Data-Mining-Verfahren, wie beispielsweise das ebenfalls beschriebene KDID-Vorgehensmodell (vgl. Kapitel 2.1.1). Für das maschinelle Lernproblem wurden überwachte Algorithmen als am geeignetsten für die Methode gesehen, da sie nur Sensordaten bearbeiten können, denen eine Information über fehlerhaften oder einwandfreien Maschinenzustand angehängt ist. Darauf aufbauend wurde das Anwendungsbeispiel als Klassifikationsproblem als eine mögliche Ausprägung des überwachten maschinellen Lernens definiert und der Random-Forest-Algorithmus zu Vorhersage eingesetzt. Für weiterführende Forschungen kann das Abschneiden von anderen Algorithmen aus unüberwachte Lernmethoden, Deep Learning oder Reinforcement Learning untersucht werden. Diese könnten sich, in Abhängigkeit vom jeweiligen Anwendungsziel, durch eine noch höhere Vorhersageleistung auszeichnen. Des Weiteren ist eine weitere Untersuchung von Regressionsalgorithmen im Rahmen der Methode erforderlich, da RUL-Vorhersagen ebenfalls als geeignet für die Methode eingestuft worden sind und im Anwendungsbeispiel lediglich ein Klassifikationsproblem behandelt wurde. Für eine literarische Arbeit, die den Vergleich von unterschiedlichen maschinellen Lernverfahren im Bereich der prädiktiven Wartung vornimmt, sei zum Beispiel auf Carvalho et al. (2019) oder Wöstmann et al. (2019) verwiesen.

Im Anwendungsbeispiel wurde der MSE als einer der Indikatoren zur Bewertung der Vorhersageperformance des maschinellen Lernmodells herangezogen. Dabei konnte festgestellt werden, dass das um Data Mining erweiterte Modell in Bezug auf diese Kennzahl eine höhere Leistungsfähigkeit aufweist. Da der MSE jedoch in der Regel bei Regressionsproblemen, die kontinuierliche Zielwerte betrachten, verwendet wird, um den Unterschied zwischen den vorhergesagten und tatsächlichen Werten zu messen

(vgl. Kapitel 2.3), ist diese Kennzahl bei dem angewendeten Klassifikationsproblem, das binäre Zielwerte betrachtet, eher weniger aussagekräftig. Folglich wurden darüber hinaus der F1-Score und die Konfusionsmatrix als Bewertungskennzahlen herangezogen. Der F1-Score zwischen beiden Modellen zeigt lediglich einen nicht signifikanten Unterschied. Eine weitere Auswertung mit der Confusion Matrix (vgl. Tabelle 4 und 5) ergibt jedoch, dass die auf Basis des durch Data Mining generierten Wissens entwickelten Features eine positive Auswirkung auf die Vorhersage der Klasse "kein Ausfall" aufweisen, während sie sich auf die Vorhersage der Klasse "Ausfall" leicht verschlechternd auswirken. In dem erweiterten Modell sind auf Grundlage des durch Data Mining erworbenen Wissens zwei Features erstellt worden, die in das maschinelle Lernen einfließen. Das erstellte Feature „Combi_TopFeatures“ setzt sich dabei aus einer Multiplikation der beiden durch Data Mining als am wichtigsten identifizierten Features zusammen. Eine weiterführende, gezieltere Erstellung von Features, die über eine einfache Multiplikation hinausgeht, könnte möglicherweise zu einem noch besseren Ergebnis führen. Das zweite generierte Feature basiert auf dem erstellten Entscheidungsbaum und integriert Schwellenwerte für Ausfallrisiken in die Vorhersage. Dies erfolgt auf Basis des ersten Splits des Entscheidungsbaums, wobei der Wurzelknoten das als am wichtigsten identifizierte Feature "Drehmoment" darstellt (vgl. Abbildung 11), um eine hohe Generalisierbarkeit bezüglich des Gesamtdatensatzes zu gewährleisten. Das erstellte Feature soll als zusätzliches Kriterium für die Entscheidungsfindung im Rahmen des maschinellen Lernens dienen. Es sei jedoch darauf hingewiesen, dass das Risiko einer Redundanz besteht, da das Modell diese Information möglicherweise bereits nutzt.

Allgemein ist in Hinblick auf den verwendeten Datensatz, der aus 10.000 Einträgen besteht, von denen nur 3,4% fehlerhafte Maschinenzustände sind, das gegebene Risiko von Überanpassung zu erwähnen (vgl. Kapitel 2.3). Würden also in Anlehnung an die entwickelte Methode aktuelle von Sensoren erfasste Daten in das Beispielmodell gegeben werden, könnte dieses bei den Vorhersagen unzureichend abschneiden, da es zu sehr an die beschränkte Trainingsdatenbasis gewöhnt ist. Gerade bei der Verwendung von Big-Data-Datensätzen, die sich aus heterogenen Quellen zusammensetzen und eine hohe Dimensionalität aufweisen, ist die Wahrscheinlichkeit erhöht, dass das maschinelle Lernen ein Programm antrainiert, das nicht auf das betrachtete Gesamtproblem bezogen werden kann und lediglich für einen kleinen Datenausschnitt relevant ist Dies verweist erneut auf die in Kapitel 2.4.3 identifizierte allgemeine Herausforderung bei der Bereitstellung einer geeigneten Datenbasis für die prädiktive Wartung in produzierenden Unternehmen. Auch im Hinblick auf die Gesamtmethode, bei der das Training des maschinellen Lernmodells auf Basis der historisch durch installierte Sensoren gesammelten Daten erfolgen soll, manifestiert sich das Problem, dass die Sensoren über

einen ausreichend langen Zeitraum die relevanten Parameter aufzeichnen müssen, um eine geeignete Datenbasis zum Modelltraining zu generieren. Dabei ist es erforderlich, dass verschiedene Arten von Maschinenausfällen erfasst werden. Sofern ein Unternehmen bislang keine Sensordaten generiert und gespeichert hat, ist vor der Implementierung eines prädiktiven Wartungssystems zunächst ein gewisser Zeitraum für die Datensammlung aufzuwenden, oder es ist auf alternative Datensätze ähnlicher Fertigungsanlagen zurückzugreifen. Wie in Kapitel 2.4.1 dargelegt, verbessert sich die Vorhersageleistung mit jeder zusätzlich analysierten Maschine. Dabei ist zu berücksichtigen, dass das Modell nicht ausschließlich auf die neueren Daten fokussiert ist, sondern auch ältere Ausfallmuster weiterhin erkennt (vgl. Kapitel 3.2).

Die abschließende Bewertung der Performance des Vorhersagemodells auf aktuelle Sensordaten erweist sich wie bereits herausgearbeitet als anspruchsvoll. Soweit den prädiktiven Wartungsempfehlungen basierend auf dem Vorhersagemodell Folge geleistet wird und dementsprechend in den Fertigungsprozess präventiv eingegriffen wird, kann nicht abschließend bewertet werden, ob der Ausfallfehler auch wirklich eingetreten wäre.

In Bezugnahme auf die exemplarisch erstellten Vorhersagemodelle aus Kapitel 4 lässt sich festhalten, dass die Generierung von Features aus Data-Mining-Wissen in diesem Fall zu einer leicht verbesserten Vorhersageleistung des maschinellen Lernmodells für die eine Klasse und einer leichten Verschlechterung für die andere Klasse führt. In einer weiterführenden Untersuchung muss eruiert werden, ob eine gezieltere Feature-Generierung zu einer noch besseren Leistung führen kann. Das ungezielte Einbringen von Features in das maschinelle Lernen ohne signifikanten zusätzlichen Informationsgehalt könnte zu Rauschen führen und das Modell mit unnötigen Informationen belasten (vgl. Kapitel 2.1.4). Der Nutzen der Methode bezüglich der Vorhersageverbesserung durch Features lässt sich für die Praxisanwendung, wo die Datensätze wesentlich größer sind, nicht abschließend validieren.

Festzuhalten ist hingegen, dass der Anwender der Methode jedoch von der Bereitstellung identifizierter Muster durch das Data Mining profitiert. In dem vorliegenden Anwendungsbeispiel wurden exemplarisch die Korrelationsmatrix, das Clusterdiagramm sowie der Entscheidungsbaum betrachtet. Damit wurden nach Kapitel 2.1.4 sowohl Teile der deskriptiven Datenanalyse durch die einfache Ermittlung von Korrelationen als auch explorative Ansätze wie der Entscheidungsbaum zur Analyse tieferliegender Zusammenhänge durchgeführt. Die Korrelationsanalyse liefert dem Anwender Informationen über den Zusammenhang zwischen den gemessenen Parametern und einem

Maschinenausfall. Auf Basis dieser Erkenntnisse wurde eine Clusteranalyse durchgeführt, um detailliertere Einsichten in die Zusammenhänge zwischen den beiden maßgeblichen Parametern "Drehmoment" und "Werkzeugverschleiß" zu gewinnen. Dies ermöglicht die proaktive Identifikation von Maschinen, die ähnliche Parameterwerte aufweisen und mit hoher Wahrscheinlichkeit ausfallen werden. Des Weiteren wurden die beiden als wesentlich identifizierten Parameter in einem generierten Feature kombiniert, welches dem maschinellen Lernen zugeführt wird. Zudem wird ein weiteres Feature basierend auf dem Entscheidungsbaum erstellt. Der Entscheidungsbaum ermöglicht die Identifikation von Entscheidungsregeln und Schwellenwerten für die wichtigsten Parameter, welche das Auftreten eines Maschinenausfalls beeinflussen. Zur Vereinfachung wurde lediglich der erste Split des Entscheidungsbaums dargestellt (vgl. Abbildung 11). Eine weiterführende Analyse ist an dieser Stelle möglich.

Die alleinige Anwendung maschinellen Lernens mit Datenvorverarbeitungsschritten ermöglicht dem Anwender lediglich die Erlangung von Prognosen, jedoch keine transparente Erklärung und kein Verständnis der zugrundeliegenden Zusammenhänge. Die bereitgestellten Muster aus dem Data Mining dienen der Erweiterung des Wissensstands des Anwenders, sodass dieser in der Lage ist, gezielte Anpassungen an risikobehafteten Anlagenkomponenten vorzunehmen beziehungsweise eine verstärkte Beobachtung derselben zu gewährleisten, um potenziellen Anlagenausfällen vorzubeugen. Die implementierten Data-Mining-Verfahren wurden dabei für das Anwendungsbeispiel in einer Python-Umgebung programmiert. Für umfangreichere und weiterführende Analysen können Umgebungen wie Rapid Miner hinzugezogen werden, in denen mit vorgefertigten Bausteinen eine Vielzahl von Data-Mining-Verfahren angewendet werden können.

Die entwickelte Methode basiert auf der Erkenntnis, dass Data Mining und maschinelles Lernen unterschiedliche Zwecke verfolgen und diese sich gegenseitig ergänzen können. Die Kombination von Data Mining und maschinellem Lernen im Bereich der sensorbasierten prädiktiven Wartung stellt insofern eine Neuheit dar und bietet zusammenfassend auf mehreren Ebenen einen Nutzen. Das in Kapitel 4 präsentierte Anwendungsbeispiel zeigt, dass das Ziel einer erhöhten Vorhersagegenauigkeit durch vorheriges Data Mining nicht vollständig validiert werden konnte. Die Vorhersage wurde durch die zusätzlichen Features teilweise präziser und teilweise weniger präzise im Vergleich zum reinen maschinellen Lernmodell. In diesem Zusammenhang sind weitere Tests und Überlegungen zur Feature-Generierung erforderlich, um ein Modell zu entwickeln, das für alle Klassen bessere Vorhersagen trifft. Aufgrund von Unterschieden in den Datensätzen, dem Datenumfang oder unterschiedlichen Zielstellungen ist eine Verallgemeinerung dieser Problemstellung nicht möglich. Somit muss bei jeder Implementierung der

Methode eine individuelle Behandlung erfolgen. Ein weiterer identifizierter Nutzen der Methode besteht in der beschriebenen Wissenserweiterung des Anwenders, die durch die reine Anwendung von maschinellem Lernen nicht erreicht worden wäre. Auf Basis dieses Wissens können neben Entscheidungen bezüglich der Fertigungsanlage auch die Auswahl und Validierung der am besten geeigneten maschinellen Lernalgorithmen erfolgen, wie zuvor beschrieben. Die Kombination von Data Mining und maschinellem Lernen in der entwickelten Methode resultiert in einer hohen Anpassbarkeit und Flexibilität der prädiktiven Wartungsstrategie. Die Anpassbarkeit ist durch den iterativen Charakter der Phasen gegeben, sodass neue Daten in den KDD-Teil integriert werden können, um das maschinelle Lernmodell zu optimieren. Zudem kann das Durchlaufen der Methode nach dem KDD-Teil beendet werden, falls dadurch bereits signifikante Zusammenhänge identifiziert wurden.

Zu Beginn dieses Kapitels wurde bereits darauf hingewiesen, dass eine Erweiterung der Untersuchung auf andere maschinelle Lernverfahren als die überwachten Verfahren möglich und vielversprechend wäre, um die Methode zu verbessern. In diesem Zusammenhang kann auch der eingeschränkte sensorbasierte prädiktive Wartungsbereich in weitergehenden Untersuchungen auf modellbasierte Ansätze oder das Einbeziehen eines digitalen Zwillings ausgedehnt werden, um eine noch umfassendere und genauere prädiktive Wartungsstrategie zu entwickeln.

Zusammenfassend lässt sich sagen, dass das exemplarische Anwendungsbeispiel teilweise den Nutzen der Methode validiert hat. Eine umfassendere Evaluierung des Methodengebrauchs erfordert die Prüfung der in diesem Kapitel identifizierten aufbauenden Erweiterungsmöglichkeiten sowie die Sammlung von Erfahrungen aus der praktischen Anwendung der Methode. Diesbezüglich ist insbesondere die Anwendung der Methode auf weitere, umfangreichere Datensätze zu untersuchen. Des Weiteren sind die in Kapitel 2.4.3 dargelegten datenwissenschaftlichen Herausforderungen zu berücksichtigen, welche im Rahmen der Methodenentwicklung nicht vollständig Berücksichtigung fanden. Die Methode präsentiert einen strukturierten Rahmen zur Implementierung und langfristigen Nutzung von prädiktiver Wartung und thematisiert die Herausforderung einer aussagekräftigen Datenqualität. Die technischen Umsetzungsmöglichkeiten bezüglich der Integration von Big-Data-Datensätzen und Analyse- und Vorhersagetools in die unternehmensinterne IT-Landschaft wurden hingegen weniger stark berücksichtigt.

6 Zusammenfassung und Ausblick

Dieses Kapitel dient der abschließenden Zusammenfassung der Arbeit und liefert einen Ausblick auf weitere, auf der Arbeit aufbauende Forschungsthemen.

Die Zielsetzung der vorliegenden Arbeit besteht in der Entwicklung einer Methode, welche die datenwissenschaftlichen Themengebiete Data Mining und maschinelles Lernen kombiniert. Der Anwendungsbereich der Methode erstreckt sich auf die prädiktive Wartung als potenzielle Instandhaltungsstrategie für produzierende Unternehmen. Auf Basis der dargelegten und für das Verständnis der Arbeit essenziellen Grundlagen zu den Bereichen Datenwissenschaften, KDD, maschinellem Lernen und prädiktiver Wartung wird in Kapitel 3 ein konkretes Vorgehensmodell entwickelt. In einem ersten Schritt erfolgt eine Einschränkung des Themengebietes "prädiktive Wartung" auf Basis der dargelegten Grundlagen. Die Methode ist folglich auf die sensorbasierte prädiktive Wartung zur Vorhersage von Maschinenausfällen oder RUL einzelner Bauteile ausgerichtet. Im ersten Teil des Vorgehensmodells erfolgt die Anwendung von KDD auf sensorbasierte Prozessdaten auf Basis des CRISP-DM-Vorgehensmodells. Die Auswahl eines adäquaten KDD-Vorgehensmodells für die Methode erfolgt durch einen Vergleich des Vorgehensmodells von Fayyad et al. mit dem CRISP-DM-Vorgehensmodell (vgl. Kapitel 2.2.2). Die Auswahl fällt auf das CRISP-DM-Modell (vgl. Kapitel 3.1), da es sich in der Industrie als weit verbreitet etabliert hat, eine klar definierte Projektstruktur aufweist, die Berücksichtigung der Datenerfassung ermöglicht, was insbesondere bei sensorbasierten Daten von grundlegender Bedeutung ist, sowie einen iterativen Ansatz verfolgt, der eine fortlaufende Anpassung erlaubt. Nach Durchlaufen des KDD-Prozesses als initialem Schritt der entwickelten Methode verfügt der Anwender über Kenntnisse bezüglich Muster und Beziehungen in den Sensordaten, die in Verbindung mit einem Maschinen- oder Anlagenausfall stehen. Die gewonnenen Erkenntnisse können dazu verwendet werden, den gesamten Produktionsprozess anzupassen, oder aber auch, um ein verstärktes Augenmerk auf kritische Faktoren zu legen und deren Wartung entsprechend anzupassen. Des Weiteren kann das erworbene Wissen als Input oder Entscheidungshilfe für den zweiten Teil der entwickelten Methode dienen. Der zweite Teil der Methode umfasst im Wesentlichen die Anwendung von maschinellem Lernen zum Prädiktieren von Maschinenausfällen oder RUL sowie die vorherige Erstellung von Features aus dem durch den ersten Teil der Methode erlangten Wissen. In Bezug auf den maschinellen Lernalgorithmus wird eine Entscheidung zugunsten der in der Praxis der prädiktiven Wartung am weitesten verbreiteten überwachten Algorithmen getroffen (vgl. Kapitel 3.1). Diese Vorgehensweise wird damit begründet, dass unüberwachte Algorithmen eine stärkere Ausrichtung auf allgemeine Muster oder Anomalien als auf spezifische Aufgaben aufweisen.

Zudem wird die Verwendung gelabelter Daten als vorteilhafter erachtet. Die Einstufung nicht gelabelter Daten als fehleranfälliger ist darauf zurückzuführen, dass eine Differenzierung zwischen normalem und fehlerhaftem Maschinenverhalten durch die Daten für den Algorithmus nicht möglich ist und stattdessen durch das maschinelle Lernmodell selbst vorgenommen werden muss. In diesem Kontext erweisen sich überwachte Lernalgorithmen als sinnvoll, da sie, wie zuvor dargelegt, lediglich gelabelte Daten als Eingabe verarbeiten können.

In Kapitel 4 erfolgt eine exemplarische Validierung der Methode. Dabei werden die zuvor als möglichen Vorteil des vor dem maschinellen Lernen durchgeführten KDD in Form von einer Auswahlunterstützung für die geeignetsten maschinellen Lernalgorithmen oder dem Validieren und Testen des maschinellen Lernens durch den ersten Teil der Methode nicht behandelt. Eine Untersuchung dieser Vorteile ist neben der Anwendung von Regressionsalgorithmen für RUL-Prognosen (vgl. Kapitel 5) als Inhalt möglicher weiterführender Forschungen anzusehen. Der Fokus liegt stattdessen auf der Validierung der Vorteile eines tieferen Verständnisses von Zusammenhängen in den Daten bei einem Maschinenausfall sowie einer Verbesserung der Vorhersageleistung des maschinellen Lernmodells durch aus dem KDD-Wissen generierte Features. Zur Validierung der entwickelten Methode wird nicht die gesamte Methode angewendet. Stattdessen wird ein vorverarbeiteter Datensatz genutzt und in einem ersten Modell die Vorhersagegenauigkeit für Maschinenausfälle eines Random Forest Algorithmus bewertet. In einem zweiten Modell werden dem Random-Forest-Algorithmus in Anlehnung an die entwickelte Methode Data Mining Aktivitäten vorgelagert. Dies umfasst die Erstellung einer Korrelationsmatrix, welche die Korrelation der einzelnen gemessenen Prozessparameter mit einem Maschinenausfall darstellt. Des Weiteren erfolgt eine Gegenüberstellung der zwei am stärksten mit einem Maschinenausfall korrelierenden Features im Rahmen einer Clusteranalyse. Das Ziel dieser Vorgehensweise ist die Darstellung derjenigen Kombination der Merkmalsausprägungen, die mit der größten Wahrscheinlichkeit zu einem Maschinenausfall führt. Im Anschluss erfolgt die Generierung eines Entscheidungsbaums, welcher die Bestimmung von Schwellenwerten einzelner Parameter vornimmt, die zu einem Maschinenausfall führen. Die Erkenntnisse der Korrelationsberechnung sowie des Entscheidungsbaums werden jeweils in Form eines Features umgesetzt, welches neben dem Datensatz als Input in das Random-Forest-Vorhersagemodell einfließt.

Es zeigt sich, dass das Einbringen von Features aus Data-Mining-Wissen für das Anwendungsbeispiel für die Vorhersage der Klasse "kein Ausfall" eine Verbesserung und für die Vorhersage der Klasse „Ausfall“ eine leichte Verschlechterung im Vergleich zum alleinigen maschinellen Lernen erwirkt. Ein generalisierbarer Nutzen konnte somit nicht

bewiesen werden und es empfiehlt sich in weiterführenden Untersuchungen eine noch gezieltere Feature-Erstellung vorzunehmen (vgl. Kapitel 5) Festzuhalten ist hingegen, dass der Anwender der Methode jedoch von der Bereitstellung identifizierter Muster durch Korrelationsmatrix, das Clusterdiagramm sowie der Entscheidungsbaum profitiert und mehrwertstiftendes Wissen über den Produktionsprozess erhält (vgl. Kapitel 5).

Die Kombination eines KDD-Vorgehensmodells mit der Integration datenbasierten Wissens in das Prädiktieren durch maschinelles Lernen stellt eine Neuheit dar. Der wesentliche Vorteil besteht, wie gezeigt, in der Fähigkeit, im Vergleich zur Anwendung eines reinen maschinellen Lernmodells mit Datenvorverarbeitung, Zusammenhänge zwischen den durch Sensoren gemessenen Parametern und einem Maschinenausfall aufzuzeigen, welche das maschinelle Lernmodell allein nicht liefern kann. Des Weiteren konnte nachgewiesen werden, dass die Bereitstellung dieser Zusammenhänge als Feature für das maschinelle Lernen zu einer teilweise verbesserten Vorhersageleistung führt.

Für eine Erweiterung der Methode bietet sich eine Untersuchung der Integration des in Kapitel 2.4.2 erwähnten digitalen Zwillings an, durch dessen Einsatz weiteren wichtigen Erkenntnisse durch Simulationsmöglichkeiten gewonnen werden könnten.

Literaturverzeichnis

Adriaans, Pieter; Zantinge, Dolf (1998): Data Mining. 3. Aufl. Harlow: Addison-Wesley.

Albon, Chris; Gallatin, Kyle (2023): Machine learning with Python cookbook. Practical solutions from preprocessing to deep learning. 2. Aufl. Sebastopool: O'Reilly.

Alcalde Rasch, Alejandro (2000): Erfolgspotential Instandhaltung. Theoretische Untersuchung und Entwurf eines ganzheitlichen Instandhaltungsmanagements. Berlin: Erich Schmidt (Duisburger betriebswirtschaftiche Schriften, 21).

Alpar, Paul; Niedereichholz, Joachim (Hg.) (2000): Data Mining im praktischen Einsatz. Verfahren und Anwendungsfälle für Marketing, Vertrieb, Controlling und Kundenunterstützung. Wiesbaden: Vieweg+Teubner Verlag (Business Computing).

Arbeitskreis Smart Service Welt (2015): Smart Service Welt – Umsetzungsempfehlungen für das Zukunftsprojekt Internet-basierte Dienste für die Wirtschaft. Hg. v. acatech. Berlin (Abschlussbericht).

Azevedo, Ana; Santos, Manuel Filipe (2008): KDD, SEMMA and CRISP-DM: a parallel overview. In: IADIS European Conf. Data Mining. Amsterdam, Niederlande, 22.06-27.06.2008. Online verfügbar unter https://api.semanticscholar.org/CorpusID:15309704.

Bishop, Christopher M. (2016): Pattern recognition and machine learning. New York: Springer (Information science and statistics).

Bissantz, Nicolas; Hagedorn, Jürgen; Mertens, Peter: Data Mining. In: Mucksch, Behme (Hg.) 2000 Das Data Warehouse Konzept, S. 377–407.

Bodendorf, Freimut (2003): Daten- und Wissensmanagement. Berlin, Heidelberg: Springer Berlin Heidelberg; Imprint; Springer (Springer-Lehrbuch).

Carvalho, Thyago; Soares, Fabrizzio; Vita, Roberto; Francisco, Roberto; Basto, João; G. Soares Alcalá, Symone (2019): A systematic literature review of machine learning methods applied to predictive maintenance. In: *Computers & Industrial Engineering* 137, S. 106024. DOI: 10.1016/j.cie.2019.106024.

Chapman, Peter; Clinton, Julian; Kerber, Randy; Khabaza, Thomas; Reinartz, Thomas; Shearer, Colin; Wirth, Rüdiger (2000): CRISP-DM 1.0: Step-by-step data mining guide. In: SPSS inc.

Chavan, Pallavi; Mahalle, Parikshit N.; Mangrulkar, Ramchandra; Williams, Idongesit (Hg.) (2023): Data science. Techniques and intelligent applications. First edition. Boca Raton, FL: Chapman & Hall/CRC Press.

Chen, Yi; Wang, Wei; Liu, Ziyang; Lin, Xuemin (2009): Keyword search on structured and semi-structured data. In: Association for Computing Machinery (Hg.): Proceedings of the 2009 ACM SIGMOD International Conference on Management of Data. Providence Rhode Island, USA, 29.06 - 02.07.2009fraun. New York, NY, USA: Association for Computing Machinery (SIGMOD '09), S. 1005–1010.

Cios, Krzysztof; Pedrycz, Witold; Swiniarski, Roman; Kurgan, Lukasz (2007): Data Mining: A Knowledge Discovery Approach. In: *Data Mining: A Knowledge Discovery Approach*. DOI: 10.1007/978-0-387-36795-8.

Cleve, Jürgen; Lämmel, Uwe (2020): Data Mining. 3. Aufl. München: De Gruyter Oldenbourg (De Gruyter Studium).

Deeg, Maria; Ditze, Andreas (2010): Statische und dynamische Modellierung von Anforderungen. In: Michael Aschenbrenner, Ralph Dicke, Bertel Karnarski und Franz Schweiggert (Hg.): Informationsverarbeitung in Versicherungsunternehmen. Berlin: Springer, S. 417–439.

Deuse, Jochen; Klinkenberg, Ralf; West, Nikolai (Hg.) (2024): Industrielle Datenanalyse. Entwicklung einer Datenanalyse-Plattform für die wertschaffende, kompetenzorientierte Kollaboration in dynamischen Wertschöpfungsnetzwerken. Wiesbaden: Springer Fachmedien Wiesbaden; Springer Vieweg.

Deuse, Jochen.; Erohin, Olga.; Lieber, Daniel. (2014): Wissensentdeckung in vernetzten, industriellen Datenbeständen. In: Industrie 4.0. Tagung der Hochschulgruppe für Arbeits- und Betriebsorganisation e.V. (HAB). Gito Verlag. Berlin.

Dhar, Vasant (2012): Data Science and Prediction. In: *Communications of the ACM* 56.

Dong, Guozhu; Liu, Huan (Hg.) (2017): Feature Engineering for Machine Learning and Data Analytics. First edition. London, England: Taylor and Francis (Chapman & Hall/CRC data mining and knowledge discovery series).

Dumbill, Edd (2013): A Revolution That Will Transform How We Live, Work, and Think: An Interview with the Authors of Big Data. In: *Big Data* 1 (2), S. 73–77. DOI: 10.1089/big.2013.0016.

Elder, John; Pregibon, Daryl (1998): A Statistical Perspective on Knowledge Discovery in Databases.

Elmasri, Ramez; Navathe, Sham (2010): Fundamentals of database systems. 6. Aufl. Boston: Addison Wesley.

Esteban, Aurora; Zafra, Amelia; Ventura, Sebastián (2022): Data mining in predictive maintenance systems: A taxonomy and systematic review. In: *WIREs Data Mining and Knowledge Discovery* 12 (5), e1471. DOI: 10.1002/widm.1471.

Ester, Martin; Sander, Jörg (2000): Knowledge Discovery in Databases. Techniken und Anwendungen. Berlin, Heidelberg: Springer.

Fahrmeir, Ludwig; Brachinger, Wolfgang (Hg.) (1996): Multivariate statistische Verfahren. 2., überarb. Aufl. Berlin: De Gruyter.

Fahrmeir, Ludwig; Heumann, Christian; Künstler, Rita; Pigeot, Iris; Tutz, Gerhard (2023): Statistik. Der Weg zur Datenanalyse. 9. Aufl. Berlin, Heidelberg: Springer.

Fawcett, Tom; Provost, Foster (2013): Data Science for Business. Sebastopol: O'Reilly Media, Inc.

Fayyad, Usama; Piatetsky-Shapiro, Gregory; Smyth, Padhraic (Hg.) (1996a): Advances in the knowledge discovery of data mining. 5. Aufl. Menlo Park, Calif.: AAAI Press.

Fayyad, Usama; Piatetsky-Shapiro, Gregory; Smyth, Padhraic (1996b): From Data Mining to Knowledge Discovery in Databases. In: *AI Magazine* 17 (3), S. 37. DOI: 10.1609/aimag.v17i3.1230.

Fayyad, Usama; Piatetsky-Shapiro, Gregory; Smyth, Padhraic (1996c): Knowledge discovery and data mining: towards a unifying framework. In: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining. Oregon, 02.08 - 04.08.1996: AAAI Press (KDD'96), S. 82–88.

Fraunhofer IVI (o.D.): Zustandsbasierte prädiktive Instandhaltung. Unter Mitarbeit von Ute Gläser. Hg. v. Fraunhofer IVI. Online verfügbar unter https://www.ivi.fraunhofer.de/de/forschungsfelder/zivilschutz-und-sicherheit/infrastrukturmanagement/zustandsbasierte-praediktive-instandhaltung.html, zuletzt geprüft am 12.07.2024.

Fraunhofer SCAI (o.D.): Prädiktive Wartung. Hg. v. Fraunhofer SCAI. Online verfügbar unter https://www.scai.fraunhofer.de/de/querschnittsthemen/maschinelles-lernen/praediktive-wartung.html, zuletzt geprüft am 01.07.2024.

Frawley, William J.; Piatetsky-Shapiro, Gregory; Matheus, Christopher J. (1992): Knowledge Discovery in Databases: An Overview. In: *AI Magazine* 13 (3), S. 57. DOI: 10.1609/aimag.v13i3.1011.

Gabriel, Roland; Gluchowski, Peter; Pastwa, Alexander (2011): Data warehouse & data mining. 1. Aufl., 1. Nachdr. Herdecke, Witten: W3L-Verl. (Informatik).

Gartner (2016): Gartner's 2016 Hype Cycle for Emerging Technologies Identifies Three Key Trends That Organizations Must Track to Gain Competitive Advantage. Stamford. Amy Ann Forni; Rob van der Meulen.

Ghodrati, Behzad (2006): Weibull and exponential renewal models in spare parts estimation: A comparison. In: *International Journal of Performability Engineering* 2, S. 135–147.

Glaessgen, Edward; Stargel, David (2012): The digital twin paradigm for future NASA and U.S. air force vehicles. In: Langley Research Center (Hg.): The Digital Twin Paradigm for Future NASA and U.S. Air Force Vehicles. Honolulu, 23.04.-26.04.2012.

Grand View Research (2023): Predictive Maintenance Market Size, Share & Trends Analysis Report By Component, By Solution, By Service, By Deployment, By Enterprise Size, By Monitoring Technique, By End-use, By Region, And Segment Forecasts. Online verfügbar unter https://www.grandviewresearch.com/industry-analysis/predictive-maintenance-market, zuletzt geprüft am 10.05.2024.

Gupta, Aparna; Lawsirirat, Chaipat (2006): Strategically optimum maintenance of monitoring-enabled multi-component systems using continuous-time jump deterioration models. In: *Journal of Quality in Maintenance Engineering* 12, S. 306–329. DOI: 10.1108/13552510610685138.

Gutenschwager, Kai.; Rabe, Markus.; Spieckermann, Sven.; Wenzel, Sigrid. (2017): Simulation in Produktion und Logistik. Grundlagen und Anwendungen. Berlin, Heidelberg: Springer Berlin Heidelberg; Imprint; Springer Vieweg.

Gutschi, Clemens; Furian, Nikolaus; Suschnigg, Josef; Neubacher, Dietmar; Voessner, Siegfried (2019): Log-based predictive maintenance in discrete parts manufacturing. In: CIRP Conference on Intelligent Computation in Manufacturing Engineering, Bd. 79. Neapel, 16.07 - 19.07.2019, S. 528–533.

Guyon, Isabelle; Elisseeff, André (2003): An Introduction of Variable and Feature Selection. In: *J. Machine Learning Research Special Issue on Variable and Feature Selection* 3, S. 1157–1182. DOI: 10.1162/153244303322753616.

Guyon, Isabelle; Elisseeff, André (2006): An Introduction to Feature Extraction. In: Isabelle Guyon, Masoud Nikravesh, Steve Gunn und Lotfi A. Zadeh (Hg.): Feature Extraction: Foundations and Applications. Berlin, Heidelberg: Springer Berlin Heidelberg, S. 1–25.

Han, Jiawei; PEI, Jian; Kamber, Micheline (2012): Data mining. Concepts and techniques. 3. Aufl. Amsterdam, Boston, Heidelberg: Morgan Kaufmann; Elsevier (The Morgan Kaufmann series in data management systems).

Hashemian, H.; Bean, Wendell (2011): State-of-the-Art Predictive Maintenance Techniques*. In: *IEEE Transactions on Instrumentation and Measurement* 60, S. 3480–3492. DOI: 10.1109/TIM.2009.2036347.

Hastie, Trevor; Tibshiriani, Robert; Friedman, Jerome (2009): The elements of statistical learning. Data mining, inference, and prediction. 2. Aufl. New York: Springer.

Henke, Michael; Heller, Thomas; Stich, Volker (2015): Smart Maintenance – Der Weg vom Status quo zur Zielvision. acatech; Fraunhofer IML.

Hiba, Jasim; Hadi, Hiba; Hameed Shnain, Ammar; Hadishaheed, Sarah; Haji, Azizahbt (2015): BIG DATA AND FIVE V'S CHARACTERISTICS. In: *International Journal of Advances in Electronics and Computer Science* (2), S. 2393–2835.

Hodapp, Wilhelm (2018): Die Bedeutung einer zustandsorientierten Instandhaltung. Einsatz und Nutzen in der Investitionsgüterindustrie. In: Jens Reichel, Gerhard Müller und Jean

Haeffs (Hg.): Betriebliche Instandhaltung. 2. Auflage. Berlin, Germany: Springer Vieweg (VDI-Buch), S. 135–152.

IDC (2023): Volumen der jährlich generierten/replizierten digitalen Datenmenge weltweit von 2010 bis 2022 und Prognose bis 2027. IDC. Online verfügbar unter https://de.statista.com/statistik/daten/studie/267974/umfrage/prognose-zum-weltweit-generierten-datenvolumen/, zuletzt geprüft am 22.08.2024.

Imielinski, Tomasz; Mannila, Heikki (2000): A Database Perspective on Knowledge Discovery. In: *Communications of the ACM* 39. DOI: 10.1145/240455.240472.

Institut für DLR Softwaretechnologie (2022): HPDA-Grundlagensoftware. Online verfügbar unter https://www.dlr.de/de/sc/forschung-transfer/projekte/hpda-grundlagensoftware, zuletzt geprüft am 19.07.2024.

Intel IT Center (2012): Einführung in Big Data: Die Analyse unstrukturierter Daten. Intel IT Center. Online verfügbar unter https://www.intel.de/content/dam/www/public/emea/de/de/pdf/unstructured-data-analytics-paper.pdf.

Jardine, Andrew K.S.; Lin, Daming; Banjevic, Dragan (2006): A review on machinery diagnostics and prognostics implementing condition-based maintenance. In: *Mechanical Systems and Signal Processing* 20 (7), S. 1483–1510. DOI: 10.1016/j.ymssp.2005.09.012.

Keith McCormick (2007): CRISP-DM 2.0. Online verfügbar unter https://keithmccormick.com/crisp-dm-20/, zuletzt geprüft am 22.08.2024.

Kohlhammer, Jörn; Proff, Dirk U.; Wiener, Andreas (2013): Visual business analytics. Effektiver Zugang zu Daten und Informationem. 1 Aufl. Heidelberg: Dpunkt.verlag (Edition TDWI).

Kolanovic, Marco; Krishnamachari, Rajesh T. (2017): Big Data and AI Strategies Machine Learning and Alternative Data Approach to Investing. Hg. v. J.P.Morgan.

Kozue (2023): Machine Failure Predictions. Online verfügbar unter https://www.kaggle.com/datasets/shashanknecrothapa/machine-failure-predictions, zuletzt geprüft am 12.06.2024.

Krcmar, Helmut (2015): Einführung in das Informationsmanagement. 2., überarb. Aufl. Berlin, Heidelberg: Springer Gabler (Springer-Lehrbuch).

Kurbel, Karl (2024): Modellierung Betrieblicher Informationssysteme. Modelle, Methoden und Werkzeuge. 1. Aufl. Basel/Berlin/Boston: Walter de Gruyter GmbH (De Gruyter Studium Series).

Kurgan, Lukasz; Musilek, Petr (2006): A survey of Knowledge Discovery and Data Mining process models. In: *Knowledge Eng. Review* 21, S. 1–24. DOI: 10.1017/S0269888906000737.

Kurrewar, Harshad; Bekar, Ebru Turanouglu; Skoogh, Anders; Nyqvist, Per (2021): A Machine Learning Based Health Indicator Construction in Implementing Predictive Maintenance: A Real World Industrial Application from Manufacturing. In: Alexandre Dolgui, Alain Bernard, David Lemoine, Gregor von Cieminski und David Romero (Hg.): IFIP International Conference on Advances in Production Management Systems (APMS), AICT-632. Nantes, 05.09 - 09.09.2021. Nantes, France: Springer International Publishing (Advances in Production Management Systems. Artificial Intelligence for Sustainable and Resilient Production Systems), S. 599–608. Online verfügbar unter https://inria.hal.science/hal-04022129.

Kusiak, Andrew (2017): Smart Manufacturing Must Embrace Big Data. In: *Nature*, S. 23–25. DOI: 10.1038/544023a.

Lee, Jay (1998): Teleservice engineering in manufacturing: challenges and opportunities. In: *International Journal of Machine Tools and Manufacture* 38 (8), S. 901–910. DOI: 10.1016/S0890-6955(97)00135-1.

Lee, Jay; Ni, Jun; Djurdjanovic, Dragan; Qiu, Hai; Liao, Haitao (2006): Intelligent prognostics tools and e-maintenance. In: *Computers in Industry* 57 (6), S. 476–489. DOI: 10.1016/j.compind.2006.02.014.

Lemke, Claudia; Brenner, Walter (2015): Einführung in die Wirtschaftsinformatik. Berlin, Heidelberg: Springer Gabler (Springer-Lehrbuch).

Leskovec, Jure; Rajaraman, Anand; Ullman, Jeffrey (2020): Mining of Massive Datasets. Cambridge: Cambridge University Press.

Lieber, Daniel; Erohin, Olga; Deuse, Jochen (2013a): Knowledge discovery in industrial databases - Challenges and applications. In: *ZWF Zeitschrift fuer Wirtschaftlichen Fabrikbetrieb* 108, S. 388–393.

Lieber, Daniel; Erohin, Olga; Deuse, Jochen (2013b): Wissensentdeckung im industriellen Kontext. Herausforderungen und Anwendungsbeispiele. In: *Zeitschrift für wirtschaftlichen Fabrikbetrieb* 108 (6), S. 388–393. DOI: 10.3139/104.110948.

Lovelace, Robin (2016): The Data Revolution: Big Data, Open Data, Data Infrastructures and Their Consequences, by Rob Kitchin. 2014. Thousand Oaks, California: Sage Publications. 222+xvii. ISBN: 978-1446287484, 100. In: *Journal of Regional Science* 56, S. 722–723. DOI: 10.1111/jors.12293.

Lu, Yang (2017): Industry 4.0: A Survey on Technologies, Applications and Open Research Issues. In: *Journal of Industrial Information Integration* 6. DOI: 10.1016/j.jii.2017.04.005.

Maimon, Oded; Rokach, Lior (2005): Introduction to Knowledge Discovery in Databases. In: Oded Maimon und Lior Rokach (Hg.): Data Mining and Knowledge Discovery Handbook. Boston, MA: Springer US, S. 1–17.

Mariscal, Gonzalo; Marbán, Oscar; Fernández, Covadonga (2010): A survey of data mining and knowledge discovery process models and methodologies. In: *Knowledge Eng. Review* 25, S. 137–166. DOI: 10.1017/S0269888910000032.

McKinsey & Company (2015): Industry 4.0 – How to navigate digitization of the manufacturing sector. Online verfügbar unter https://www.mckinsey.com/capabilities/operations/our-insights/industry-four-point-o-how-to-navigae-the-digitization-of-the-manufacturing-sector, zuletzt geprüft am 19.07.2024.

Meddaoui, Anwar; Hachmoud, Adil; Mustapha, Hain (2024): Advanced ML for Predictive Maintenance: Case Study on Remaining Useful Life Prediction and Reliability Enhancement.

Mehmeti, Xhemajl; Mehmeti, Besart; Sejdiu, Rrahim (2018): The equipment maintenance management in manufacturing enterprises. In: *IFAC-PapersOnLine* 51, S. 800–802. DOI: 10.1016/j.ifacol.2018.11.192.

Meier, Andreas; Kaufmann, Michael (2016): SQL- & NoSQL-Datenbanken. 8., überarb. u. erw. Aufl. 2016. Berlin, Heidelberg: Springer Berlin Heidelberg; Imprint; Springer Vieweg (EXamen.press).

Mertens, Peter; Freimut, Bodendorf; König, Wolfgang; Schumann, Matthias; Hess, Thomas; Buxmann, Peter (2017): Grundzüge der Wirtschaftsinformatik. 12. Aufl. 2017. Berlin, Heidelberg: Springer Berlin Heidelberg; Imprint Springer Gabler.

Mühlnickel, Helmut; Kurz, Cäcilia Maria; Jussen, Philipp; Emonts-Holley, Roman (2018): Smart Maintanance. Instandhaltung im Kontext der Industrie 4.0. In: Jens Reichel, Gerhard Müller und Jean Haeffs (Hg.): Betriebliche Instandhaltung. 2. Auflage. Berlin, Germany: Springer Vieweg (VDI-Buch).

Müller, Andreas C.; Guido, Sarah (2017): Introduction to machine learning with Python. A guide for data scientists. 1. Aufl. Sebastopol, CA: O'Reilly Media, Inc.

Multhaupt, Marko (2000): Data mining und Text mining im strategischen Controlling. Aachen: Shaker (Berichte aus der Betriebswirtschaft).

Murphy, Kevin P. (2013): Machine learning. A probabilistic perspective. 4. print. (fixed many typos). Cambridge, Mass.: MIT Press (Adaptive computation and machine learning series).

NESSI White Paper (2012): Big Data A New World of Opportunities. NESSI.

North, Klaus (2011): Wissensorientierte Unternehmensführung. Wiesbaden: Gabler.

North, Klaus; Maier, Ronald (2018): Wissen 4.0 – Wissensmanagement im digitalen Wandel. In: *HMD* 55 (4), S. 665–681. DOI: 10.1365/s40702-018-0426-6.

Orth, Ronald; Steinhöfel, Erik; Galeitzke, Mila; Hecklau, Fabian (2018): Wissensmanagement im Kontext von Industrie 4.0. In: *ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb* 113, S. 377–380. DOI: 10.3139/104.111924.

Otte, Ralf; Otte, Viktor; Kaiser, Volker (2004): Data Mining für die industrielle Praxis. München, Wien: Hanser.

Paul, Anthony; Odu, Anthony; Oluwaseyi, Joseph (2024): Predictive Maintenance: Leveraging Machine Learning for Equipment Health Monitoring.

Peng, Ying; Dong, Ming; Zuo, Ming (2010): Current status of machine prognostics in condition-based maintenance: A review. In: *International Journal of Advanced Manufacturing Technology* 50, S. 297–313. DOI: 10.1007/s00170-009-2482-0.

Piatstsky, Gregory (2014): CRISP-DM, still the top methodology for analytics, data mining, or data science projects. Hg. v. KDnuggets. Online verfügbar unter https://www.kdnuggets.com/2014/10/crisp-dm-top-methodology-analytics-data-mining-data-science-projects.html.

Portela, Filipe (2022): Data Science and Knowledge Discovery. Basel: MDPI - Multidisciplinary Digital Publishing Institute.

Porter, Michael E.; Heppelmann, James E. (2015): Wie smarte Produkte Unternehmen verändern. In: *Harvard-Business-Manager : das Wissen der Besten* (37), S. 52–73.

Powers, David (2008): Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation. In: *Mach. Learn. Technol.* 2.

Pressman, R. S. (2005): Software engineering. A practitioner's approach. 6. ed. New York, NY: McGraw-Hill (McGraw-Hill series in computer science).

Probst, Gilbert; Raub, STefan; Romhardt, Kai (2003): Wissen managen. Wie Unternehmen ihre wertvollste Ressource optimal nutzen. 4., überarbeitete Auflage. Wiesbaden: Gabler Verlag.

Provost, Foster; Fawcett, Tom (2013): Data Science and Its Relationship to Big Data and Data-Driven Decision Making. In: *Big Data* 1. DOI: 10.1089/big.2013.1508.

Pyle, Dorian (2007): Data preparation for data mining. San Francisco: Morgan Kaufmann.

Quix, Christoph (2021): Big-Data-Technologien. In: Detlev Frick, Andreas Gadatsch, Jens Kaufmann, Birgit Lankes, Christoph Quix, Andreas Schmidt und Uwe Schmitz (Hg.): Data Science. Konzepte, Erfahrungen, Fallstudien und Praxis. 1st ed. 2021. Wiesbaden: Springer Fachmedien Wiesbaden; Springer Vieweg, S. 133–148.

Rosmaini, Ahmad; Shahrul, Kamaruddin (2012): An overview of time-based and condition-based maintenance in industrial application. In: *Computers & Industrial Engineering* 63 (1), S. 135–149. DOI: 10.1016/j.cie.2012.02.002.

Rotondo, Anna; Quilligan, Fergus (2020): Evolution Paths for Knowledge Discovery and Data Mining Process Models. In: *SN Computer Science* 1. DOI: 10.1007/s42979-020-0117-6.

Runkler, Thomas A. (2015): Data Mining. Modelle Und Algorithmen Intelligenter Datenanalyse: Vieweg + Teubner Verlag.

Russell, Stuart (2010): Artificial Intelligence: A Modern Approach. 3. Aufl. Pearson.

Samon, Daniel; Shalom, Joseph (2023): Using Machine Learning to Monitor Equipment Health in Predictive Maintenance, 2023.

SAS (2017): Introduction to SEMMA. Online verfügbar unter https://documentation.sas.com/doc/en/emref/14.3/n061bzurmej4j3n1jnj8bbjjm1a2.htm, zuletzt aktualisiert am 30.08.2017.

Schatten, Alexander; Biffl, Stefan; Demolsky, Markus; Gostischa-Franta, Erik; Oestreicher, Thomas; Winkler, Dietmar (2010): Best Practice Software-Engineering. Eine praxiserprobte Zusammenstellung von komponentenorientierten Konzepten, Methoden und Werkzeugen. Heidelberg: Spektrum Akademischer Verlag.

Scheidler, Anne Antonia (2017): Methode zur Erschließung von Wissen aus Datenmustern in Supply-Chain-Datenbanken. Göttingen: Cuvillier Verlag (Schriftenreihe Fortschritte in der IT in Produktion und Logistik, v.1).

Schnell, Marcus; Jussen, Philipp; Moser, Benedikt (2018): Smart Services - Datenbasierte Dienstleistungen in der Instandhaltung. In: Jens Reichel, Gerhard Müller und Jean Haeffs (Hg.): Betriebliche Instandhaltung. 2. Auflage. Berlin, Germany: Springer Vieweg (VDI-Buch).

Schütte, Reinhard (1998): Grundsätze ordnungsmäßiger Referenzmodellierung. Konstruktion konfigurations- und anpassungsorientierter Modelle. Wiesbaden: Gabler Verlag; Imprint (Neue Betriebswirtschaftliche Forschung, 233).

Soori, Mohsen; Karimi Ghaleh Jough, Fooad; Dastres, Roza; Arezoo, Behrooz (2024): Internet of Things and Data Analytics for Predictive Maintenance in Industry 4.0, A Review.

Stachowiak, Herbert (1973): Allgemeine Modelltheorie. Wien, New York: Springer.

Tao, Fei; Cheng, Jiangfeng; Qi, Qinglin; Zhang, Meng; Zhang, He; Sui, Fangyuan (2018): Digital twin-driven product design, manufacturing and service with big data. In: *The International Journal of Advanced Manufacturing Technology* 94. DOI: 10.1007/s00170-017-0233-1.

UCI Machine Learning Repository (2020): AI4I 2020 Predictive Maintenance Dataset. Online verfügbar unter https://doi.org/10.24432/C5HS5C.

van der Valk, Hendrik; Haße, Hendrik; Möller, Frederik; Arbter, Michael; Henning, Jan-Luca; Otto, Boris (2020): A Taxonomy of Digital Twins. In:

Vinay, S.; Pub, Iaeme (2024): DATA SCIENTIST COMPETENCIES AND SKILL ASSESSMENT: A COMPREHENSIVE FRAMEWORK 1, S. 1–11.

Wang, Chen; Vo, Hoang; Ni, Peng (2015): An IoT Application for Fault Diagnosis and Prediction. In: IEEE International Conference on Data Science and Data Intensive Systems (DSDIS), S. 726–731.

Wang, J.; Li, C.; Han, S.; Sarkar, S.; Zhou, X. (2017): Predictive maintenance based on event-log analysis: A case study. In: *IBM Journal of Research and Development* 61 (1), 11:121-11:132. DOI: 10.1147/JRD.2017.2648298.

Welte, Rebecca; Estler, Manfred; Lucke, Dominik (2020): A Method for Implementation of Machine Learning Solutions for Predictive Maintenance in Small and Medium Sized Enterprises. In: *Procedia CIRP* 93, S. 909–914. DOI: 10.1016/j.procir.2020.04.052.

Wöstmann, René; Barthelmey, André; West, Nikolai; Deuse, Jochen (2019): A Retrofit Approach for Predictive Maintenance. In: Thorsten Schüppstuhl, Kirsten Tracht und Jürgen Roßmann (Hg.): Tagungsband des 4. Kongresses Montage Handhabung Industrieroboter. Berlin, Heidelberg: Springer Berlin Heidelberg, S. 94–106.

Xiaoling Shu; Yiwan Ye (2023): Knowledge Discovery: Methods from data mining and machine learning. In: *Social Science Research* 110, S. 102817. DOI: 10.1016/j.ssresearch.2022.102817.

Yan, Jihong; Meng, Yue; Lu, Lei; Li, Lin (2017): Industrial Big Data in an Industry 4.0 Environment: Challenges, Schemes and Applications for Predictive Maintenance. In: *IEEE Access* PP, S. 1. DOI: 10.1109/ACCESS.2017.2765544.

Zheng, Alice; Casari, Amanda (2018): Feature Engineering for Machine Learning: Principles and Techniques for Data Scientists. 1st: O'Reilly Media, Inc.

Zulqarnain, Ali (2024): Data Science: Its Role and Importance Data Science: Its Role and importance.

Anhang

1. Programmcode: Random Forest

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split, StratifiedKFold, cross_val_score
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import confusion_matrix, classification_report, mean_squared_error
from imblearn.over_sampling import SMOTE


# Datensatz laden
df = pd.read_csv(r'C:\Users\Freddy\Desktop\Masterarbeit\Datensatz\machine failure.csv')


# Entfernen von ID-Spalten und Ausfallgründen
df = df.drop(columns=['UDI', 'Product ID', 'TWF', 'HDF', 'PWF', 'OSF', 'RNF'])


# One-Hot-Encoding für kategoriale Variablen
df = pd.get_dummies(df, columns=['Type'], drop_first=True)


# Modellierung mit Random Forest
X = df.drop(columns=['Machine failure'])
y = df['Machine failure']


# Train-Test-Split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, stratify=y, random_state=42)


# SMOTE nur auf den Trainingsdaten anwenden
smote = SMOTE(random_state=42)
X_train_res, y_train_res = smote.fit_resample(X_train, y_train)


# Random Forest Classifier
```

```
clf_rf = RandomForestClassifier(random_state=42)


# Cross-validation
kfold = StratifiedKFold(n_splits=5, shuffle=True, ran-
dom_state=42)
scores = cross_val_score(clf_rf, X_train_res, y_train_res,
cv=kfold, scoring='f1')


# Ergebnisse ausgeben
print(f"Durchschnittlicher F1-Score für das Random Forest-Mo-
dell: {np.mean(scores):.4f}")
print(f"Standardabweichung des F1-Scores für das Random Forest-
Modell: {np.std(scores):.4f}")


# Modell trainieren und evaluieren
clf_rf.fit(X_train_res, y_train_res)
y_pred_rf = clf_rf.predict(X_test)


# Evaluation
mse_rf = mean_squared_error(y_test, y_pred_rf)
print(f"Mean Squared Error für das Random Forest-Modell:
{mse_rf:.4f}")


conf_matrix_rf = confusion_matrix(y_test, y_pred_rf)
print("Confusion Matrix für das Random Forest-Modell:")
print(conf_matrix_rf)


print("Classification Report für das Random Forest-Modell:")
print(classification_report(y_test, y_pred_rf))
```

2. Programmcode: Random-Forest mit vorherigen Data Mining

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split,
StratifiedKFold, cross_val_score
from sklearn.ensemble import RandomForestClassifier
```

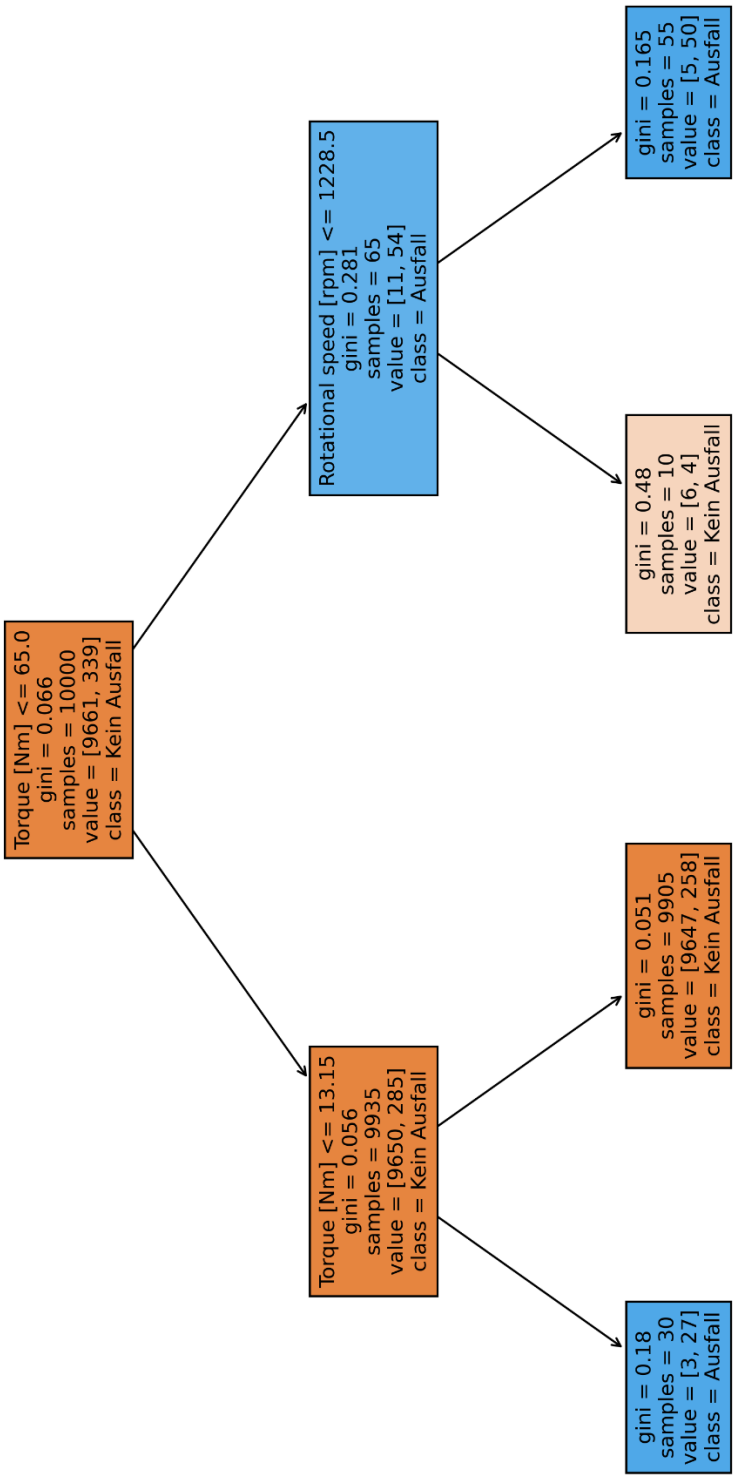
```
from sklearn.metrics import confusion_matrix, classification_report, mean_squared_error
from imblearn.over_sampling import SMOTE
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import KMeans
from sklearn.tree import DecisionTreeClassifier
from sklearn import tree

# Datensatz laden
df = pd.read_csv(r'C:\Users\Freddy\Desktop\Masterarbeit\Datensatz\machine failure.csv')

# Entfernen von ID-Spalten und Ausfallgründen
df = df.drop(columns=['UDI', 'Product ID', 'TWF', 'HDF', 'PWF', 'OSF', 'RNF'])

# One-Hot-Encoding für kategoriale Variablen
df = pd.get_dummies(df, columns=['Type'], drop_first=True)

# Data Mining - Korrelationsanalyse
plt.figure(figsize=(10, 8))
correlation_matrix = df.corr()
sns.heatmap(correlation_matrix[['Machine failure']].sort_values(by='Machine failure', ascending=False),
            annot=True, cmap='coolwarm', linewidths=0.5)
plt.show()

# Identifizieren der am stärksten korrelierten Features
correlated_features = correlation_matrix['Machine failure'].abs().sort_values(ascending=False)
print("Top 5 korrelierte Features mit Maschinenausfall:")
print(correlated_features.head(6))  # inkl. 'Machine failure'

# Clustering der einflussreichsten Features
top_features = correlated_features.index.tolist()[1:3]  # Nimm die ersten zwei, überspringe 'Machine failure'
X_cluster = df[top_features]

# Standardisierung der Daten für das Clustering
scaler = StandardScaler()
```

```
X_cluster_scaled = scaler.fit_transform(X_cluster)

# KMeans-Clustering durchführen
kmeans = KMeans(n_clusters=3, random_state=42)
clusters = kmeans.fit_predict(X_cluster_scaled)

# Cluster zu den ursprünglichen Daten hinzufügen
df['Cluster'] = clusters

# Clustermuster plotten
plt.figure(figsize=(10, 6))
scatter = plt.scatter(X_cluster_scaled[:, 0], X_cluster_scaled[:, 1],
                      c=df['Machine failure'], cmap='coolwarm', alpha=0.5)
plt.xlabel(top_features[0])
plt.ylabel(top_features[1])
plt.colorbar(scatter, label='Maschinenfehler (0=Kein Ausfall, 1=Ausfall)')
plt.grid()
plt.show()

# Mustererkennung mit Entscheidungsbaum
X = df.drop(columns=['Machine failure'])
y = df['Machine failure']

# Entscheidungsbaum-Modell trainieren
clf = DecisionTreeClassifier(random_state=42, max_depth=2)
clf.fit(X, y)

# Visualisierung des Entscheidungsbaums
plt.figure(figsize=(15, 8))
tree.plot_tree(clf, feature_names=X.columns, class_names=['Kein Ausfall', 'Ausfall'],
               max_depth=3, filled=True, fontsize=10)
plt.show()

# Feature Engineering - Neues Feature basierend auf den Top-Features
```

```
df['Combi_TopFeatures'] = df[top_features[0]] * df[top_fea-
tures[1]]


# Entscheidungsbaum-Pfad als neues Feature: Erstellen eines Fea-
tures, das angibt, ob der Knoten 1 (nach dem ersten Split im
Baum) aktiv ist

first_feature = clf.tree_.feature[0]

first_threshold = clf.tree_.threshold[0]

df['Path_Node_1'] = (df[X.columns[first_feature]] <=
first_threshold).astype(int)


# Modellierung mit Random Forest

X = df.drop(columns=['Machine failure'])  # Unabhängige Variab-
len inkl. neuem Feature

y = df['Machine failure']  # Zielvariable


X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, stratify=y, random_state=42)


# SMOTE nur auf den Trainingsdaten anwenden

smote = SMOTE(random_state=42)

X_train_res, y_train_res = smote.fit_resample(X_train, y_train)


# Random Forest Classifier

clf_rf = RandomForestClassifier(random_state=42)


# Cross-validation

kfold = StratifiedKFold(n_splits=5, shuffle=True, ran-
dom_state=42)

scores = cross_val_score(clf_rf, X_train_res, y_train_res,
cv=kfold, scoring='f1')


# Ergebnisse ausgeben

print(f"Durchschnittlicher F1-Score für das Random Forest-Mo-
dell: {np.mean(scores):.4f}")

print(f"Standardabweichung des F1-Scores für das Random Forest-
Modell: {np.std(scores):.4f}")


# Modell trainieren und evaluieren

clf_rf.fit(X_train_res, y_train_res)

y_pred_rf = clf_rf.predict(X_test)
```

```
# Evaluation
mse_rf = mean_squared_error(y_test, y_pred_rf)
print(f"Mean Squared Error für das Random Forest-Modell:
{mse_rf:.4f}")

conf_matrix_rf = confusion_matrix(y_test, y_pred_rf)
print("Confusion Matrix für das Random Forest-Modell:")
print(conf_matrix_rf)

print("Classification Report für das Random Forest-Modell:")
print(classification_report(y_test, y_pred_rf))
```

3. Entscheidungsbaum



4.