



**BERGISCHE  
UNIVERSITÄT  
WUPPERTAL**

Fakultät für Maschinenbau und Sicherheitstechnik  
Fachgebiet Verkehrssicherheit und Zuverlässigkeit  
Jun.-Prof. Dr. Antoine Tordeux

## **Masterthesis**

---

# **Evaluation und Vergleich der Methoden zur prädiktiven Instandhaltung im Kontext von Industrie 4.0**

---

Name: Margarita Kartashova

Matrikel-Nr.: 1666681

Studiengang: Master Qualitätsingenieurswesen

Betreuer: Jun.- Prof. Dr. Antoine Tordeux

Datum der Ausgabe: 01.07.2021

Datum der Abgabe: 30.11.2021

## **Erklärung**

Hiermit, erkläre ich, dass ich die von mir eingereichte Abschlussarbeit (Master-Thesis) selbstständig verfasst und keine andere als die angegebene Quelle und Hilfsmittel benutzt sowie Stellen der Abschlussarbeit, die anderen Werken dem Wortlaut oder Sinn nach entnommen wurden, in jedem Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Ich bin damit einverstanden, dass die Arbeit durch Dritte eingesehen und unter Wahrung urheberrechtlicher Grundsätze zitiert werden darf.

Wuppertal, den 30.11.2021

---

## **Zusammenfassung**

Die vorausschauende Instandhaltung gilt als eine zentrale Innovation der Industrie 4.0. Die wurde als zentrales Thema dieser Masterarbeit identifiziert und wird als Voraussetzung für zukünftigen Erfolg erkannt, da mit ihr Instandhaltungsmaßnahmen und damit verbundene Entscheidungsprozesse in einem komplexen industriellen Umfeld gezielter und proaktiver getroffen werden können. In diesem Beitrag werden die derzeitigen modernen Instandhaltungsstrategien und Methoden der vorausschauenden Instandhaltung ausführlicher erörtert.

In Analyseteil der Arbeit wurden verschiedene Methoden der prädiktiven Instandhaltung implementiert, ausgewertet und miteinander verglichen. Das Schwerpunkt der Implementierung lag auf dem Zustand des technischen Systems auf Basis von Sensordaten abzuschätzen und klassifizieren. Zu diesem Zweck wurde ein Triebwerksdatensatz aus dem Referenzdatenspeicher der NASA verwendet. Für die Datenanalyse wurden folgende Klassifikationsalgorithmen verwendet: logistische Regression, künstliches neuronales Netz, Support Vector Machine, Entscheidungsbaumanalyse, Naive Bayes, K-Means-Algorithmus und Hierarchische Clusteranalyse. Es wurde untersucht, wie genau verschiedene Klassifizierungsalgorithmen den Zustand eines Flugzeugturbinensystems erkennen können. Die Ergebnisse zeigen, dass die überwachten Algorithmen im Vergleich zu den unüberwachten den Systemzustand genauer ermittelt haben. Unter den überwachten Algorithmen haben die künstliche neuronale Netz mit dem Fehlerrate von 0,1259 und Support Vector Machine mit dem Fehlerrate von 0,1339 das beste Ergebnis erzielt.

## **Abstract**

Predictive maintenance is considered a key innovation of Industry 4.0. It has been identified as a central topic of this master thesis and is recognised as a prerequisite for future success, as it allows maintenance actions and related decision-making processes to be taken in a more targeted and proactive manner in a complex industrial environment. In this paper, the current modern maintenance strategies and methods of predictive maintenance are discussed in more details.

In the analysis part of the thesis, different methods of predictive maintenance were implemented, evaluated and compared with each other. The focus of the implementation was on estimating and classifying the condition of the technical system on the basis of sensor data. For this purpose, an engine data set from NASA's reference data repository was used. The following classification algorithms were used for the data analysis: logistic regression, artificial neural network, support vector machine, decision tree analysis, Naive Bayes, K-Means algorithm and hierarchical cluster analysis. It was investigated how accurately different classification algorithms can detect the state of an aircraft turbine system. The results show that the supervised algorithms were more accurate in detecting the system state compared to the unsupervised ones. Among the supervised algorithms, Artificial Neural Network with the error rate of 0.1259 and Support Vector Machine with the error rate of 0.1339 achieved the best result.

## Abkürzungsverzeichnis

AHC	Agglomerative hierarchische Clustermethode
CHAID	Chi-square Automatic Interaction Detectors
CART	Classification and Regression Trees
CPS	Cyber-Physische-Systeme
CPPS	Cyber-physisches Produktionssystem
DGUV	Deutsche Gesetzliche Unfallversicherung
DIN	Deutsches Institut für Normung
IKT	Informations- und Kommunikationstechnik
KMA	K-means-Algorithmus
LR	Lineare Regression
NN	Neuronales Netz
PC	Principal Component
RF	Random Forest
RFID	Radio-Frequency Identification
RMSE	Root mean square error
SVM	Support-Vector-Machine

# Tabellenverzeichnis

Tabelle 1: Software- und Hardware-Zuverlässigkeit .....	9
Tabelle 2: Details zum Datensatz .....	36
Tabelle 3: Messungen am Triebwerk [61] .....	37
Tabelle 4: Übersicht auf die verwendete ML-Algorithmen .....	39
Tabelle 5: Mittelwert von Fehlerrate der überwachten Algorithmen.....	46
Tabelle 6: Mittelwert der Fehlerrate von unüberwachten Algorithmen.....	50
Tabelle 7: Mittelwert der Fehlerrate von überwachten Algorithmen anhand der normalisierten Datensätze .....	57
Tabelle 8: Mittelwert der Fehlerrate von unüberwachten Algorithmen anhand normalisierter Datensätze .....	61
Tabelle 9: Mittelwert der ges. Fehlerrate von allen Arten der Algorithmen anhand normalisierter Datensätze .....	61

# Abbildungsverzeichnis

Abbildung 1: Struktur der Arbeit und zentrale Forschungsfragen der jeweiligen Kapitel .....	3
Abbildung 2: Stufen der industriellen Revolution [BITKOM 2014] .....	4
Abbildung 3: Kerntechnologien von Industrie 4.0 [Geisberger et al. 2012] .....	5
Abbildung 4: Komponenten eines CPS [Broy 2010] .....	5
Abbildung 5: a) Badewannenkurve für Hardware-Zuverlässigkeit, b) Badewannenkurve für Software-Zuverlässigkeit [Pan 1999] .....	9
Abbildung 6: Instandhaltungsarten in Anlehnung an [DIN 31051 2019] .....	12
Abbildung 7: Idealtypischer Verlauf einer Abbaukurve (Modifiziert nach [DIN 31051 2019]) .....	13
Abbildung 8: Einteilung der maßnahmenbezogenen Instandhaltungsstrategien Instandhaltung in Anlehnung an [Schenk 2010] .....	14
Abbildung 9: Verlauf des Abnutzungsvorrates bei periodische Instandhaltung in Anlehnung an [Schenk 2010] .....	16
Abbildung 10: Verlauf des Abnutzungsvorrates bei präventive zustandsorientierte in Anlehnung an [Schenk 2010] .....	17
Abbildung 11: Verlauf des Abnutzungsvorrates bei präventive zustandsorientierte in Anlehnung an [Schenk 2010] .....	19
Abbildung 12: Entwicklung der entstehenden Gesamtkosten über den Instandhaltungsaufwand und Zuordnung der drei Instandhaltungsarten [Hodapp 2009]	20
Abbildung 13: Methoden des maschinellen Lernens in Anlehnung an [Kudelina et al. 2021] .....	22
Abbildung 14: Darstellung der Regressions- und Klassifikationsverfahren .....	23
Abbildung 15: Darstellung der logistischen Funktion auf dem Bereich $-4 \leq \eta \leq 4$ [Kalisch et al. 2021] .....	25
Abbildung 16: Prinzipielle Funktionsweise der SVM in Anlehnung an [Rebala et al. 2019]	26
Abbildung 17: Darstellung der Regressions- und Klassifikationsbäume in Anlehnung an [Bankhofer et al. 2008] .....	27
Abbildung 18: Entscheidung über Pruning eines Entscheidungsbaums [Bankhofer et al. 2008] .....	28
Abbildung 19: Visualisierungen der Ergebnisse von Clusteranalyse in Anlehnung an [Schacht et al. 2019] .....	29
Abbildung 20: Darstellung von eine Clusternalyse und Anomalieerkennung in Anlehnung an [Schacht et al. 2019] .....	30
Abbildung 21: Schematische Darstellung eines künstlichen Neuronales Netzes mit einem Eingangs- und Ausgangsneuron [Schenk 2010] .....	32
Abbildung 22: Unteranpassung und Überanpassung in Abhängigkeit der Modellkomplexität [Schacht et al. 2019] .....	34
Abbildung 23: Überanpassung bei einer zu kleinen Trainingsmenge [Schacht et al. 2019] ...	34
Abbildung 24: Vereinfachtes Darstellung des simulierten Turbinenriebwerks in Anlehnung an [Saxena et al. 2008] .....	36
Abbildung 25: Die Anzahl der Hauptkomponenten für 4 Unterdatensätze .....	39
Abbildung 26: Datenanalyse während der Beobachtung .....	41

Abbildung 27: Prozess der Erkennung des Turbinenanlagenzustands.....	42
Abbildung 28: Mittlere Fehlerrate der überwachten Algorithmen bei der Ermittlung des .....	43
Abbildung 29: Fehlerrate der überwachten Algorithmen bei der Ermittlung des Systemausfalls .....	44
Abbildung 30: Fehlerrate der überwachten Algorithmen bei der Ermittlung des betriebsfähigen Systemzustands.....	45
Abbildung 31: Mittlere Genauigkeit der unüberwachten Algorithmen bei der Ermittlung des Systemzustands .....	47
Abbildung 32: Genauigkeit der unüberwachten Algorithmen bei der Ermittlung des Systemausfalls.....	48
Abbildung 33: Genauigkeit der unüberwachten Algorithmen bei der Ermittlung des betriebsfähigen Systemzustands.....	49
Abbildung 34: Darstellung von Anzahl der Betriebszyklen für jede Turbinenanlage.....	51
Abbildung 35: Darstellung von Datenverteilung der normalisierten Datensatz .....	52
Abbildung 36: Mittlere Genauigkeit der überwachten Algorithmen bei der Ermittlung des Systemzustands anhand der normalisierten Daten .....	53
Abbildung 37: Genauigkeit der überwachten Algorithmen bei der Ermittlung des Systemausfalls anhand der normalisierten Daten.....	54
Abbildung 38: Genauigkeit der überwachten Algorithmen bei der Ermittlung des betriebsfähigen Systemzustands anhand der normalisierten Daten .....	56
Abbildung 39: Mittlere Genauigkeit der unüberwachten Algorithmen bei der Ermittlung des Systemzustands anhand normalisierter Datensätze.....	58
Abbildung 40: Genauigkeit der unüberwachten Algorithmen bei der Ermittlung des Systemausfalls anhand normalisierter Datensätze .....	59
Abbildung 41: Genauigkeit der unüberwachten Algorithmen bei der Ermittlung des betriebsfähigen Systemzustands anhand normalisierter Datensätze .....	60
Abbildung 42: Darstellung der ges. Fehlerrate von allen Algorithmen .....	62

# Inhaltsverzeichnis

<b>1. Einleitung</b> .....	1
1.1 Aktualität des Themas .....	1
1.2 Problemstellung .....	2
1.3 Zielstellung .....	2
1.4 Aufbau der Arbeit .....	2
<b>2. Stand der Technik</b> .....	4
2.1 Industrie 4.0 .....	4
2.2 Big Data .....	6
2.3 Zuverlässigkeit in der Industrie 4.0 .....	7
2.4 Grundstrategien der Instandhaltung .....	11
2.4.1 Instandhaltungsformen .....	11
2.4.2 Korrektive Instandhaltung .....	14
2.4.3 Präventive Instandhaltung .....	15
2.4.3.1 Periodische Instandhaltung .....	15
2.4.3.2 Präventive zustandsorientierte Instandhaltung .....	16
2.4.3.3 Präventiv vorrausschauende Instandhaltung .....	18
2.4.3.4 Auswahl der Instandhaltungsstrategie .....	19
2.5 Methoden der prädiktiven Instandhaltung .....	20
2.5.1 Machine Learning .....	20
2.5.2 Überwachtes Lernen .....	22
2.5.2.1 Logistische Regression .....	24
2.5.2.2 Support-Vektor-Maschinen .....	26
2.5.2.3 Entscheidungsbaum .....	27
2.5.3 Unüberwachtes Lernen .....	28
2.5.4 Verstärkers Lernen .....	31
2.5.5 Künstliche neuronale Netze .....	32
2.3.6 Unteranpassung und Überanpassung .....	33
<b>3 Anwendungsbeispiel der prädiktiven Instandhaltung</b> .....	35
3.1 Datenquelle und Programmiersprache .....	35
3.1.1 Datensatz .....	35
3.1.2 Programmiersprache .....	37
3.2 Analyse .....	38
3.2.1 Festlegung der Labels .....	38
3.2.2 Hauptkomponentenanalyse .....	38
3.2.3 Eingesetzte Klassifikationsalgorithmen .....	39

3.2.4 RMSE.....	40
3.3 Ergebnisse.....	41
3.4 Bias-Problem .....	50
3.5 Ergebnisse nach der Normalisierung der Datensätze .....	52
<b>4 Fazit .....</b>	<b>63</b>
<b>Literaturverzeichnis .....</b>	<b>64</b>

# 1. Einleitung

## 1.1 Aktualität des Themas

Die Anforderungen der Märkte steigen ständig, und was die Industrie heute leisten kann, wird bald nicht mehr ausreichen. Die Verbraucher erwarten, dass Produkte immer schneller geliefert werden und Innovationen in immer kürzerer Zeit verfügbar sind. Die Vielfalt der Möglichkeiten und der Umfang der geforderten Dienstleistungen führen dazu, dass die Komplexität der Produkte ständig zunimmt [Obermaier 2019]. Gleichzeitig gehen sie mit vielfältigen negativen ökologischen (z. B. Verbrauch fossiler Rohstoffe, Treibhausgasemissionen) und sozialen Konsequenzen (z. B. gesundheitliche Gefährdung, Korruption, Kinderarbeit) einher. Gleichzeitig liegt die Konzentration auf eine langfristig effizientere Energie- und Ressourcennutzung sowohl im eigenen Interesse der Unternehmen als auch im Interesse der zukünftigen Umwelt und der Menschheit.

Neue Möglichkeiten zur Steigerung der Ressourceneffizienz in der Wirtschaft werden heute aus der digitalen Transformation erwartet. Die durchgängige Vernetzung aller Wirtschaftsbereiche wird sowohl die industrielle Produktion selbst als auch die von Unternehmen angebotenen Produkte und Serviceleistungen tiefgreifend verändern. Zudem wird diese Vernetzung zu neuen Geschäftsmodellen und Wertschöpfungsprozessen führen [Berger 2015].

Im Rahmen der Digitalisierung, Automatisierung und intelligenter Systeme erleben wir eine weitere industrielle Revolution und befinden uns in der Industrie 4.0. Unter dem Begriff Industrie 4.0 werden zum einen im engeren Sinne (produzierende) Unternehmen verstanden, zum anderen bezeichnet er weniger einen Prozess als vielmehr einen bestimmten Zustand, zu dem die digitale Transformation der Produktion führt. In einer Studie des Bundesministeriums für Wirtschaft und Energie (BMWi) im Rahmen der Begleitforschung zum Technologieprogramm AUTONOMIK für Industrie 4.035 findet sich folgende Definition: „Der Begriff Industrie 4.0 bezeichnet im industriellen Kontext die Verbindung der digitalen Welt des Internets mit den klassischen Prozessen und Dienstleistungen der produzierenden Wirtschaft. Es geht um die horizontale und vertikale Vernetzung entlang der Wertschöpfungskette mit einer Verlagerung der Steuerung von oben nach unten " [Wischmann et al. 2015].

Hinter dem Begriff "Industrie 4.0" verbirgt sich auch die Vision, Maschinen, Fertigungsprozesse und Lagersysteme mit Hilfe von IT-Lösungen zu vernetzen. Ein zentrales Element ist dabei die "intelligente Fabrik". Bei diesem Konzept geht es um intelligente Produktionssysteme und -prozesse sowie um die Realisierung verteilter und vernetzter Produktionsstätten. Wichtige Ziele sind dabei die Entwicklung von intelligenten Produkten, Verfahren und Prozessen, die es den einzelnen Bereichen ermöglichen, selbstständig Informationen auszutauschen [Catelani et al. 2018].

Das Konzept der vorausschauenden Instandhaltung (Predictive Maintenance) als wesentliche Elemente der Industrie 4.0 verfolgt einen vorausschauenden, aktiven Ansatz bei der Instand-

haltung von Maschinen und Anlagen. Dabei werden aus Mess- und Produktionsdaten von Maschinen und Anlagen Instandhaltungsinformationen abgeleitet, mit dem Ziel, Ausfallzeiten möglichst zu vorhersagen und zu verhindern [Schüller et al. 2015]. Die erfolgreiche Vorhersage künftiger Ereignisse ist ein großer Vorteil. Ein wichtiger Schlüssel zur erfolgreichen Vorhersage künftiger Ereignisse findet sich in den Mustern vergangener Ereignisse. Dies ist das Spielfeld der statistischen Methoden, die unter den Oberbegriffen Data Mining und Predictive Analytics zusammengefasst werden [Schüller et al. 2015].

## **1.2 Problemstellung**

Die vierte industrielle Revolution hat zu einem hohen Grad an technischer Ausrüstung in den Unternehmen geführt, aber die Methoden zu deren Bedienung und Instandhaltung haben sich nicht geändert. In modernen Produktionsbetrieben werden oft die gleichen Wartungsintervalle für die unterschiedlichen Anlagentypen angewandt, obwohl diese Anlagen anders betrieben werden und der Verschleißprozess somit anders verläuft. Die historischen Ausfallursachen sind oft unbekannt, da früher reaktiv gewartet wurde und die ausgefallenen Komponenten einfach ausgetauscht wurden. Um jedoch eine zuverlässige Prognose für die Zukunft erstellen zu können, sind zuverlässige und objektive Daten aus der Vergangenheit erforderlich. Mit dem zunehmenden Einfluss des Internets der Dinge gibt es immer mehr Möglichkeiten, physische Objekte miteinander zu verbinden und so den Zugriff auf objektbezogene Daten wie Sensordaten zu erleichtern. Dabei handelt es sich um eine riesige Datenmenge, die auf vielfältige Weise genutzt werden kann. Die Analyse und Auswertung solcher Daten ist heute für viele Unternehmen ein Problem. Eine Möglichkeit ist die vorausschauende (prädiktive) Instandhaltung, bei der Daten genutzt werden, um drohende Ausfälle zu erkennen und entsprechende Instandhaltungsmaßnahmen zu ergreifen, indem Wartungs- und Serviceintervalle individuell festgelegt werden, Maschinenstillstände vermieden werden und das Ersatzteilmanagement besser geplant werden kann.

## **1.3 Zielstellung**

Die vorliegende Masterarbeit hat zum Ziel, wichtige Elemente der Zuverlässigkeitsanalyse, bestehende Instandhaltungsstrategien und Methoden der vorausschauenden Instandhaltung im Kontext von Industrie 4.0 zu betrachten. Diese Arbeit befasst sich mit dem Ziel, den Zustand eines technischen Systems zu bewerten. Dazu wird ein datenbasiertes Triebwerksturbinensystem verwendet und dessen Zustand mit verschiedenen Methoden der vorausschauenden Instandhaltung klassifiziert. Der Fokus liegt auf der Implementierung der überwachten und unüberwachten Algorithmen zur Zustandsbewertung und zur besseren Identifizierung relevanter Zusammenhänge zur Zustandsbewertung.

## **1.4 Aufbau der Arbeit**

Diese Arbeit besteht aus vier Kapiteln. Abbildung 1 zeigt die Struktur und erläutert das zentrale Thema, das jedem Kapitel zugrunde liegt.

Nach der Einleitung folgt Kapitel 2, in dem der Stand der Technik mit den Grundelementen der Zuverlässigkeitsanalyse, den bestehenden Instandhaltungsstrategien und den Instandhaltungsmethoden für technische Systeme dargestellt wird. Hier werden die Grundlagen für nachfolgende Bereiche wie Industrie 4.0, Digitalisierung oder Big Data erläutert. Sobald die Grundlagen geschaffen sind, wird der Themenbereich der Instandhaltung betrachtet. Die Analyse von Instandhaltungsstrategien und Methoden der vorausschauenden Instandhaltung sind hier besonders zu erwähnen.

Ein Anwendungsbeispiel für die vorausschauende Instandhaltung eines komplexen Triebwerksturbinensystems wird in Kapitel 3 erörtert. Zu diesem Zweck wird der bereits gesammelte Datensatz mit statistischen Methoden so analysiert, dass eine Klassifizierung der Turbinenleistung bzw. des Ausfalls möglich ist. Die Resultate der Analyse werden genauer ausgewertet und es wird die Genauigkeit der verschiedenen Methoden verglichen. Abschließend werden die Ergebnisse zusammengefasst und Vorhersagen über die Einsatzfähigkeit des Instandhaltungssystems in Kapitel 4 getroffen.

<b>Kapitel 1: Einführung</b>
<ul style="list-style-type: none"> <li>• Ist das Thema der Masterarbeit relevant?</li> <li>• Was ist das Ziel der Arbeit?</li> <li>• Welche Forschungsfragen stehen im Fokus?</li> </ul>
<b>Kapitel 2: Stand der Technik</b>
<ul style="list-style-type: none"> <li>• Was ist Industrie 4.0?</li> <li>• Welche Bedeutung hat die Zuverlässigkeit in der Industrie 4.0?</li> <li>• Welche Faktoren beeinflussen die Zuverlässigkeit?</li> <li>• Was sind die wichtigen Aspekte der Zuverlässigkeitsanalyse und -bewertung in der Industrie 4.0?</li> <li>• Welche Instandhaltungsformen existieren?</li> <li>• Welche Methoden des maschinellen Lernens sind bekannt und wie sie verwendet können?</li> <li>• Was ist die Überanpassung und die Unteranpassung?</li> </ul>
<b>Kapitel 3: Anwendungsbeispiel</b>
<ul style="list-style-type: none"> <li>• Welchen Datensatz und Programmiersprache wurden für die Untersuchung verwendet?</li> <li>• Wie wird die Zustandsbewertung für ein technisches System durchgeführt?</li> <li>• Funktioniert der entwickelte Algorithmus wie erwartet?</li> <li>• Kann die Prognosefähigkeit verbessert werden?</li> </ul>
<b>Kapitel 4: Fazit</b>
<ul style="list-style-type: none"> <li>• Was sind die gewonnenen Erkenntnisse dieser Arbeit?</li> <li>• Wo gibt es Ansatzpunkte für zukünftige Forschungsthemen?</li> </ul>

Abbildung 1: Struktur der Arbeit und zentrale Forschungsfragen der jeweiligen Kapitel 3

## 2. Stand der Technik

### 2.1 Industrie 4.0

Das Thema Industrie 4.0 bezeichnet die vierte Stufe der industriellen Revolution und ist die konsequente Weiterentwicklung des bisherigen Einsatzes von Informationstechnik und Automatisierung in der Produktion [BITKOM 2014]. In der Abbildung 2 ist zu sehen, wie sich der Komplexitätsgrad und die Art der Aufgaben in den Unternehmen im Laufe der Zeit verändert haben.

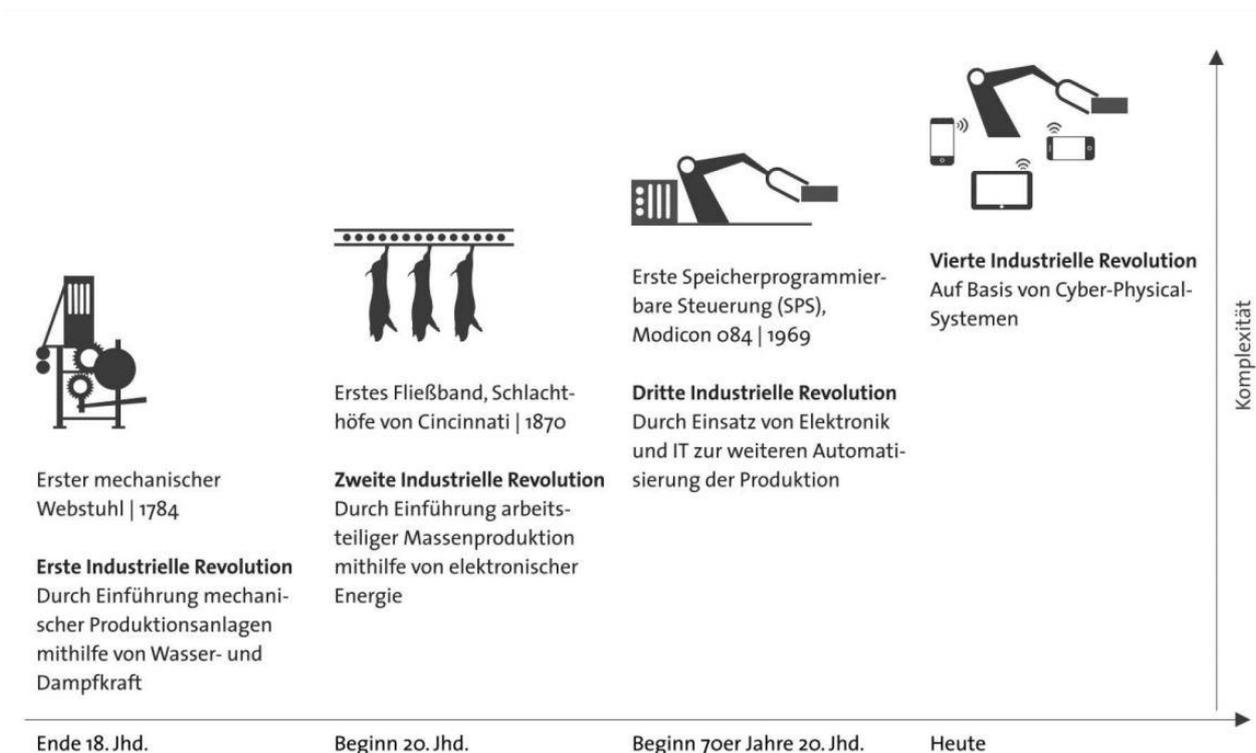


Abbildung 2: Stufen der industriellen Revolution [BITKOM 2014, S.2]

Zu den Kerntechnologien von Industrie 4.0 gehören cyber-physische Systeme, das Internet der Dinge, Cloud, Big Data, Augmented Reality und Dienste sowie Simulation, Virtualisierung und Analytics [Bauernhansl et al. 2014] [Plattform I4.0 2015] [Freiknecht et al. 2015] (Abbildung 3). Diese ermöglichen eine umfassende Vernetzung von intelligenten Produktionssystemen und eine detaillierte Erfassung und Auswertung betrieblicher Prozesse. Auf diese Weise eröffnen sie neue Produktionsformen der Organisation [Geisberger et al. 2012].

Die Entstehung von cyber-physischen Systemen (CPS) und der Übergang zu Industrie 4.0 führt zu einem Paradigmenwechsel von der deskriptiven zur prädiktiven und präskriptiven Instandhaltung, die den höchsten Reifegrad und die höchste Komplexität der wissensbasierten Instandhaltung darstellt [Ansari et al. 2018]. Heutzutage hilft dieser Übergang wettbewerbsfähigen Unternehmen bei der Optimierung ihrer Produktionsprozesse. Im Allgemeinen können CPS als

eingebettete Systeme mit erweiterten Fähigkeiten verstanden werden. Dazu gehört insbesondere die Fähigkeit, sich selbstständig mit anderen CPS zu vernetzen. Auf diese Weise können CPS miteinander kommunizieren und Daten austauschen [Broy 2010] [Lee 2008].

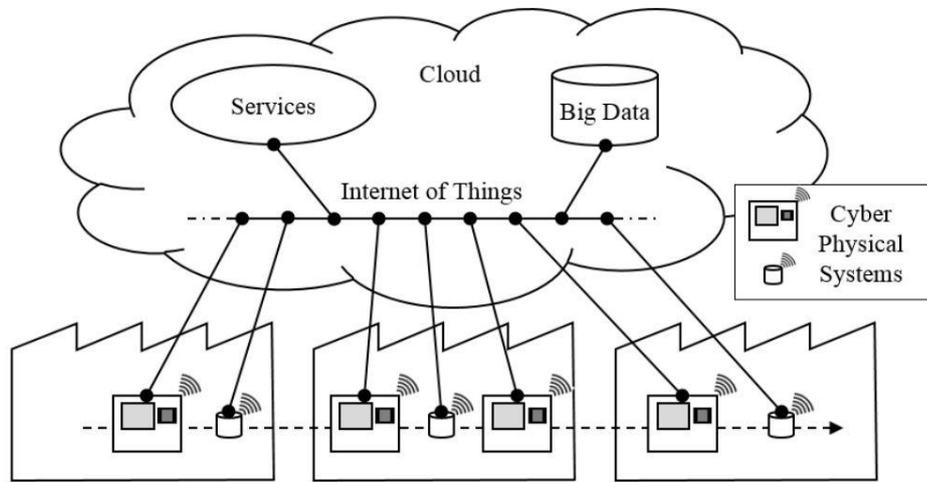


Abbildung 3: Kerntechnologien von Industrie 4.0 [Geisberger et al. 2012, S.3]

CPS sind in der Lage, ihre Umgebung durch Sensoren zu erfassen, Daten zu verarbeiten, ihre Umgebung durch Aktoren zu beeinflussen und verfügen über Schnittstellen zur Kommunikation mit der Umgebung (Abbildung 4) [Broy 2010].

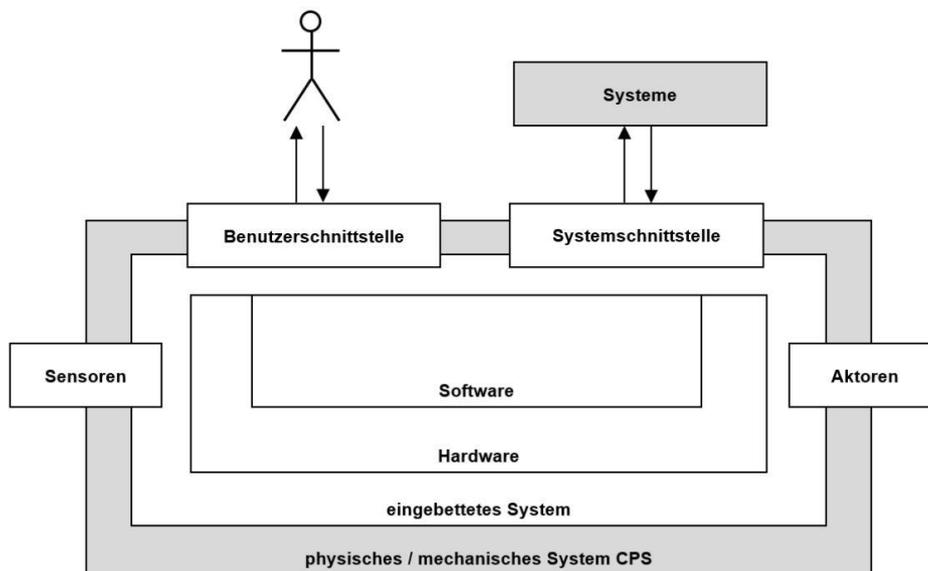


Abbildung 4: Komponenten eines CPS [Broy 2010, S.24]

Die Vernetzung einzelner Systeme und Komponenten untereinander auf Basis von CPS und dem Internet der Dinge erfordert eine effiziente und energiesparende Vernetzung mit aktuellen Netzwerktechnologien (z.B. RFIDs, 6LoWPAN, Wireless Sensors und Actuators Networks, 4G und 5G) und verfolgt verschiedene Ziele wie die Schaffung von intelligenten Fabriken

(Smart Factory) mit Adaptivität, Flexibilität, Ressourceneffizienz und ergonomischen Arbeitsbedingungen. Ermöglicht wird dies durch die Einführung des neuen *Internetprotokolls* (IPv6), das ausreichend Adressen zur Verfügung stellt, um die Vernetzung von Ressourcen, Informationen, Objekten und Menschen zu ermöglichen [Catelani et al. 2018]. Es sollen alle Bereiche des täglichen Lebens, wie Smart Mobility, Smart Logistic, Smart Buildings, Smart Product und Smart Grid mit der Fabrik der Zukunft vernetzt werden. Die dezentralen Einheiten realisieren die Anforderungen dynamisch durch Kooperation und können sich an die geänderten Randbedingungen anpassen. Gleichzeitig kommen auf die automatisierten Systeme neue Herausforderungen bezüglich der standardisierten Kommunikation, der Entwicklung intelligenter und kooperierender Module, der Architektur der Produktionsanwendungen, der Zuverlässigkeit, der Safety und der Security [Schüller et al. 2015]. In solch einem Netz sind jederzeit aktuelle Informationen über den Zustand von Maschinen und ihrer Umgebung abrufbar. Über große räumliche Distanzen hinweg können hochkomplexe technische Prozesse gesteuert werden. Mit Hilfe cyber-physischer Systeme wird sogar die Wertschöpfung über den gesamten Produktlebenszyklus kontrollierbar.

Die wichtigste Rolle für eine effiziente und kosteneffiziente Produktion spielen leistungsfähige und ständig verfügbare Produktionsanlagen. In diesem Zusammenhang trägt die Instandhaltung dazu bei, die maximale Anlagenverfügbarkeit zu gewährleisten und gleichzeitig den Material- und Zeitaufwand für Wartung und Reparatur zu minimieren. Um die Zuverlässigkeit und Betriebssicherheit von Produktionsanlagen zu erhöhen, werden Instandhaltungsmaßnahmen wie Reparaturen oder Ersatzbeschaffungen, die Modernisierung von Maschinen (z.B. im Hinblick auf Veränderungen der Produktionsgeschwindigkeit), geplante Überholungen und Korrekturmaßnahmen durchgeführt. Diese Maßnahmen beeinflussen die Flexibilität, die Durchlaufzeit und die Qualität auf der betrieblichen und logistischen Ebene eines Unternehmens [Waeyenbergh et al. 2002].

## 2.2 Big Data

Big Data gilt als einer der Impulsgeber für die vierte industrielle Revolution (Industrie 4.0). Der Begriff Big Data wird in einem Artikel von Bitkom [BITKOM 2012] wie folgt beschrieben: „Big Data bezeichnet die wirtschaftlich sinnvolle Gewinnung und Nutzung entscheidungsrelevanter Erkenntnisse aus qualitativ vielfältigen und unterschiedlich strukturierten Informationen, die einem schnellen Wandel unterliegen und in bisher ungekanntem Umfang anfallen.“ Big Data ist das Ergebnis der zunehmenden Digitalisierung ehemals analoger Datenquellen sowie des Aufkommens neuer digitaler Datenquellen. Nicht nur die Datenmenge (engl. *Volume*), sondern auch die Geschwindigkeit der Datenerzeugung, -verarbeitung und -speicherung (engl. *Velocity*) sowie die Heterogenität der Datentypen und -quellen (engl. *Variety*) haben in den letzten Jahren deutlich zugenommen [Laney 2001].

Die Daten können aus sehr unterschiedlichen Quellen stammen (IP, Sensoren, soziale Medien, Verbindungsdaten, Transaktionsdaten, Geschäftszahlen usw.), unterschiedliche Datentypen (Texte, Zahlen, Bilder usw.) enthalten und in strukturierter oder unstrukturierter Form vorliegen. Wichtige Motoren des Datenwachstums sind die zunehmende (mobile) Internetnutzung,

die steigende Nutzungsintensität sozialer Netzwerke, die fortschreitende Digitalisierung der Produktion, der Haushalte, der Energieversorgung sowie der Mobilität (Internet of Things, IoT) und der Dienstleistungen (Internet of Services, IoS). Darüber hinaus sinken die Hardwarekosten für Endgeräte, Server und Speichermedien. Insgesamt wächst das verfügbare Datenvolumen mit hoher Geschwindigkeit. Schätzungen gehen davon aus, dass das weltweit verfügbare Volumen elektronischer Daten im Jahr 2025 auf etwa 175 Zettabyte (1 Zettabyte =  $10^{21}$  Byte) [IDC 2018] ansteigen wird.

Zur Verarbeitung dieser enormen Datenmengen werden unter anderem auch entsprechend leistungsfähige Prozessoren benötigt. Die Erfassung und Verarbeitung dieser riesigen, meist heterogenen Datenmengen erfordert meist auch neue Datenbank- und Verarbeitungstechnologien. Denn herkömmliche Data-Warehouse-Konzepte, die auf relationalen Datenbanken betrieben und mit SQL gelesen und verarbeitet werden, stoßen bei Big Data schnell an ihre Grenzen. Zu den neuen, für Big Data geeigneten Systemen gehören neben so genannten NoSQL-Datenbanken (Not only SQL") beispielsweise die Technologien Apache Hadoop und Apache Spark, die als verteilte Software die Durchführung rechenintensiver Prozesse mit großen Datenmengen (im Petabyte-Bereich) auf Computerclustern ermöglichen. Hadoop ist ein in Java geschriebenes Open-Source-Framework zur parallelen Datenverarbeitung auf sehr hoch skalierbaren Serverclustern [INTEL 2012].

Alternativ oder ergänzend zu Hadoop-Technologien kommen In-Memory-Systeme zum Einsatz, bei denen Rechenprozesse direkt im Hauptspeicher verarbeitet werden und somit deutlich schneller durchgeführt werden können [Freiknecht et al. 2018]. Alternativ oder in Kombination mit Hadoop können im Bereich Big Data auch Massive Parallel Processing Systeme eingesetzt werden. Dabei handelt es sich um eine Rechnerarchitektur, bei der zahlreiche Prozessoren gleichzeitig einen Teil der Gesamtdaten verarbeiten.

### **2.3 Zuverlässigkeit in der Industrie 4.0**

Der Begriff Zuverlässigkeit bezeichnet die Wahrscheinlichkeit, dass ein Produkt, ein System oder eine entsprechende Dienstleistung über einen bestimmten Zeitraum hinweg seine beabsichtigte Funktion ordnungsgemäß erfüllt. Eine geringe Zuverlässigkeit eines Systems bedeutet nicht nur kostspielige Reparaturen und Auswechslungen, sondern auch einen Produktionsrückgang und folglich einen geringeren Umsatz. Die Bedeutung der Zuverlässigkeit von Produktionssystemen ist schon seit vielen Jahrzehnten sehr hoch [Miriayala 1989] [Chakraborty et al. 1989]. Diese Bedeutung hat sich bis heute nicht geändert, im Gegenteil, sie hat mit der zunehmenden Komplexität der Fertigungssysteme sogar noch zugenommen [Xia et al. 2018] [Catelani et al. 2018]. Was sich allerdings geändert hat, ist, dass zunehmend Informationen und Daten zur Verfügung stehen, die eine effizientere und genauere Zuverlässigkeitsanalyse von Systemen ermöglichen. Daher besteht nun die Möglichkeit, umfangreiche Expertenkenntnisse zu verbessern und zu überprüfen, die normalerweise die Zuverlässigkeitsanalyse unterstützten.

Ein Grund dafür ist, dass sich die Zuverlässigkeit eines Fertigungssystems auf eine Reihe anderer wichtiger Leistungsindikatoren auswirkt, z. B. auf die Produktionskosten, die Wartungskosten und die Kosten für das Ersatzteilmanagement, was sich letztlich auf die Produktivität auswirkt. Dies wiederum wirkt sich auf die Qualitäts- und Umsatzentwicklung der Produktion aus, da eine geringe Zuverlässigkeit zu häufigen Ausfällen und damit zu häufigen Unterbrechungen der Produktionsprozesse führt. Bei verwandten Kennzahlen wie der Zuverlässigkeit werden auch die Ausfallzeiten von Systemen und Komponenten aufgrund von Reparaturen berücksichtigt, so dass einige dieser Leistungskennzahlen eine umfassendere Bewertung ermöglichen.

Industrie 4.0 schafft neue Möglichkeiten für die Automatisierung von Prozessen, die für die Durchführung von Zuverlässigkeitsanalysen von Fertigungssystemen bedeutsam sind, und führt zu neuartigen Datentypen zu allen relevanten Faktoren, einschließlich der Interaktion zwischen Mensch und Maschine. Die Entwicklung neuer Technologien und Dateninfrastrukturen ermöglicht auch die Entwicklung von Methoden zur Validierung aller historisch verwendeten Zuverlässigkeitsmodelle und ihrer Erweiterungen oder Kalibrierungen [ACATECH 2013].

Darüber hinaus wurde die prädiktive und präskriptive Instandhaltung von Maschinen kürzlich als die wichtigste Anwendung der industriellen Datenanalyse identifiziert, wie Columbus in [Columbus 2016] feststellt. Da die Zuverlässigkeit eine der wichtigsten Messgrößen für die Bewertung von Instandhaltungsplänen, -strategien oder -verfahren ist, ist eine genaue Zuverlässigkeitsanalyse und -bewertung ebenso wichtig [Catelani et al. 2018] [ACATECH 2013]. Entscheidungen, die auf der Grundlage ungenauer Modelle getroffen werden, können sich oft nachteilig auf die Leistung eines Systems auswirken und sowohl seine Sicherheit als auch seine Zuverlässigkeit beeinträchtigen.

Da Produktionsanlagen sogenannte cyber-physische Systeme sind, wird die Zuverlässigkeit von drei Faktoren beeinflusst: Software-Zuverlässigkeit, Hardware-Zuverlässigkeit und Zuverlässigkeit infolge in Bezug auf menschliche Aktivitäten. Diese drei Faktoren sind von sehr unterschiedlicher Art und müssen daher zunächst einzeln analysiert und dann zusammengefasst werden, um ein einheitliches Maß für die Gesamtzuverlässigkeit zu ermitteln.

Beim Einsatz von Hardwarekomponenten gibt es drei Phasen: zu Beginn, während des Betriebs, wenn die Auswirkungen von Fertigungsfehlern auftreten, dann stabilisiert sich die Ausfallrate bis zur "Abnutzungsphase", in der die Ausfallrate aufgrund der Alterung der Komponenten wieder ansteigt (siehe Abbildung 5 a). Softwarekomponenten haben die höchste Ausfallrate zu Beginn, während der Integrations- und Testphase. Während dieser Testphase werden Fehler beseitigt, und die Ausfallrate sinkt und stabilisiert sich bis zur nächsten Aktualisierung. Der effektivste Weg zur Erstellung zuverlässiger Software ist die Vermeidung von Fehlern bereits im Softwareentwicklungsprozess. Die Softwarekomponenten haben keine abschließende "Abnutzungsphase", in der die Ausfallrate steigt (siehe Abbildung 5 b). Stattdessen ist die letzte Phase bei Software die Veralterung, in der die Komponente veraltet und keine Upgrades mehr angeboten werden. Einer der größten Unterschiede zwischen Software- und Hardware-Zuverlässigkeit besteht darin, dass Software-Zuverlässigkeit nicht von der Zeit abhängt, Hardware-Zuverlässigkeit hingegen schon. Andererseits ist die Software-Zuverlässigkeit in hohem Maße

von Veränderungen in der Umgebung abhängig und reagiert empfindlich auf diese, z. B. auf Aktualisierungen von Hardware- oder Softwarekomponenten sowie auf andere relevante Ereignisse [Pan 1999].

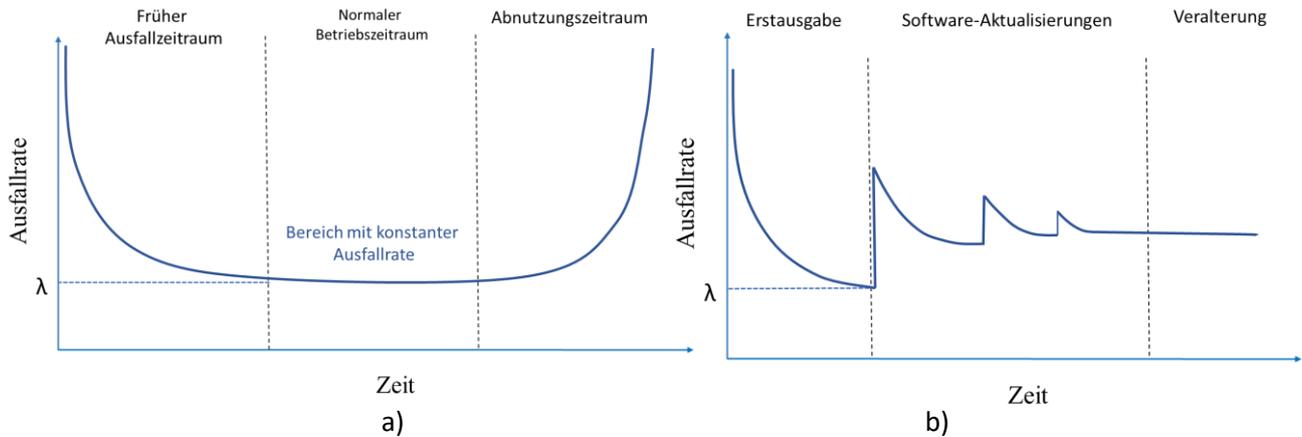


Abbildung 5: a) Badewannenkurve für Hardware-Zuverlässigkeit, b) Badewannenkurve für Software-Zuverlässigkeit in Anlehnung an [Pan 1999]

In der Tabelle 1 sind die weitere wichtige Faktoren zu Software- und Hardware-Zuverlässigkeit dargestellt.

Tabelle 1: Software- und Hardware-Zuverlässigkeit [Gesus 2006, S. 21]

Hardware-Zuverlässigkeit	Software-Zuverlässigkeit
Versagen (Ausfall) wird durch Mängel in der Konstruktion, Produktion und Instandhaltung ( sowie durch die Bedienung ) verursacht.	Versagen ist in erster Linie auf Fehler in der Konstruktion (inhärente Fehler) und auf Bedienungsfehler zurückzuführen. Verbesserungen (Reparaturen, Instandhaltung) werden durch Änderungen an den Konstruktionen vorgenommen. Auf diese Weise wird Robustheit gegenüber Bedingungen erreicht, die zu Ausfällen führen können.
Ein Versagen tritt aufgrund von Abnutzung und Verschleiß auf.  Manchmal ist eine Warnung möglich, bevor der Ausfall eintritt.	Es gibt keine Abnutzungs- und Verschleißerscheinungen. Softwarefehler treten ohne Vorwarnung auf. Code, der schon lange im Einsatz ist, kann aufgrund von neuen Fehlern, die bei Updates (engl. Upgrade) eingeführt werden, eine steigende Ausfallrate aufweisen.
Die Zuverlässigkeit der Anlage kann durch Instandhaltungsmaßnahmen erhöht werden.	Es gibt keinen äquivalenten Ersatz für vorbeugende Instandhaltungsmaßnahmen im Bereich der Software.
Die Zuverlässigkeit ist zeitabhängig. Die Ausfallrate kann mit der Betriebszeit abnehmen, konstant bleiben oder zunehmen (siehe Abbildung 4a).	Die Zuverlässigkeit ist nicht zeitabhängig (siehe Abbildung 4b). Ein Fehler tritt auf, wenn der logische Pfad, der den Fehler enthält, ausgeführt wird. Eine Erhöhung der Zuverlässigkeit ist zu beobachten, wenn Fehler entdeckt und behoben wurden.

Die Zuverlässigkeit ist von den Umgebungsbedingungen abhängig.	Externe Umgebungsbedingungen haben keinen signifikanten Einfluss auf die Zuverlässigkeit. Interne Umgebungsbedingungen, wie zu wenig Speicher oder ungeeignete Takraten, beeinflussen die Software-Zuverlässigkeit.
Ausfallraten von Hardwarekomponenten sind bis zu einem gewissen Grad nach bekannten Schemata vorhersehbar.	Ausfallraten von Softwarekomponenten sind nicht vorhersehbar.
Hardware-Schnittstellen sind sichtbar.	Software-Schnittstellen sind gedanklicher Natur.
Die Zuverlässigkeit kann gewöhnlich durch Redundanz erhöht werden.	Die Zuverlässigkeit kann nicht durch Redundanz erhöht werden, da diese nur den gleichen Fehler erzeugt. Die Zuverlässigkeit kann durch Diversität erhöht werden.
In der Hardwareentwicklung werden Standardkomponenten verwendet.	In der Softwareentwicklung werden in der Regel keine Standardkomponenten verwendet ( bislang).

Neben der Software- und Hardware-Zuverlässigkeit müssen wir auch die Auswirkungen menschlicher Interaktion auf die Zuverlässigkeit eines Systems berücksichtigen, da Fertigungssysteme auf bestimmten Ebenen oft von Menschen bedient werden. Die menschliche Interaktion erhöht die Wahrscheinlichkeit, dass das Fertigungssystem auf unbeabsichtigte oder falsche Weise verwendet oder bedient wird, was zu einer geringeren Zuverlässigkeit und einer größeren Fehleranfälligkeit führen würde. Ein großer Teil der Fertigungssysteme ist für eine intensive Interaktion mit dem Menschen ausgelegt, so dass der Mensch in hohem Maße Einfluss auf den Betrieb dieser Systeme hat.

Neuere Entwicklungen im Bereich der IKT und deren Einsatz in Produktionssystemen bieten daher erhebliche Möglichkeiten für die effektive Erfassung von Fehler- und Ausfalldaten aus diesen Systemen und lassen sich zur Ergänzung von Expertenwissen und zur Erstellung genauerer Zuverlässigkeitsmodelle nutzen. Die Entwicklung von Industrie 4.0 hat eine Menge neuer Fragen und damit verbundener Möglichkeiten aufgeworfen, die den Ansatz zur Analyse und Bewertung der Zuverlässigkeit von Fertigungssystemen verändern könnten. Wir konzentrieren uns dabei insbesondere auf die folgenden Aspekte:

- die Verfügbarkeit und Einfachheit der Datenerfassung

Aufgrund der Verfügbarkeit und der Einfachheit der Datenerfassung wurden neuartige Daten gesammelt, wobei gleichzeitig neue und ausgefeilter Methoden entwickelt werden mussten, um die Daten vollständig nutzen zu können. Die Daten werden beispielsweise in der Praxis meist in Form von Zeitreihen erhoben, ohne dass die Ausfallereignisse gezielt erfasst werden. Das bedeutet, dass es einen hohen Bedarf an Methoden gibt, die sich auf die Erkennung von Ereignissen konzentrieren, so dass Fehler und andere Ereignisse aus den

Zeitreihendaten extrahiert werden können. Darüber hinaus wird eine genaue Methodik für die Ursachenanalyse die nächste Anforderung sein, z. B. um die Abhängigkeiten zwischen Ereignissen zu extrahieren und zu modellieren. Wenn diese Methoden weit genug fortgeschritten sind, kann die Zuverlässigkeitsanalyse von Systemen automatisiert werden, z. B. auf der Datenbasis von Produktionsmaschinen. Da die meisten Produktionsanlagen nicht unbedingt sicherheitskritisch sind und Fehler/Ausfälle relativ häufig vorkommen, sind datenbasierte Ansätze sehr geeignet. Eine Datenerfassung für Zuverlässigkeitsanalysen in solchen sicherheitskritischen Systemen wäre unmöglich, da Ausfälle in diesen Systemen Menschenleben kosten können. Beispiele sind Automobilsysteme oder Flugzeuge [Bauernhansl et al. 2014].

- dieselben flexiblen Maschinen werden von verschiedenen Herstellern für unterschiedliche Zwecke eingesetzt

Die Erwartung flexibler Maschinen, die unterschiedliche Aufgaben erfüllen können, was bedeutet, dass dieselben Maschinen repetitiv eingesetzt werden. Die Datenerfassung und die neuen Technologien führen jedoch auch zu einer höheren Komplexität der Systeme. Diese erhöhte Komplexität bedeutet auch eine größere Anfälligkeit für Fehler und Ausfälle. Wenn Komponenten Aufgaben von anderen Komponenten übernehmen könnten, heißt das nicht, dass der Ausfall einer Komponente nicht zum Ausfall des Gesamtsystems führen kann. Es könnte auch der Fall eintreten, dass eine spezielle Komponente nicht einfach durch eine andere ersetzt werden kann. Sollte diese nicht redundant verfügbar sein, kann diese spezielle Aufgabe von keiner anderen Komponente übernommen werden. Eine Betrachtung der Zuverlässigkeit ist somit unerlässlich [Bauernhansl et al. 2014].

- neue Technologien führen zu komplexeren und störanfälligeren Systemen

Beispielsweise sind CPS-Prozesse häufig von Sensordaten abhängig, so dass ein Fehler in den Sensoren zu erheblichen Schäden und kostspieligen Folgen führen kann. Diese Anfälligkeit unterstreicht nur die Notwendigkeit einer genauen und effizienten Zuverlässigkeits- und Zustandsbewertung von Produktionsanlagen [Bauernhansl et al. 2014].

## **2.4 Grundstrategien der Instandhaltung**

### **2.4.1 Instandhaltungsformen**

In diesem Abschnitt werden die aktuell in der Industrie verwendeten Instandhaltungsformen und die Bedeutung der Fehlerdiagnose und -prognose beschrieben.

Instandhaltung umfasst alle im Lebenszyklus durchgeführten technischen und administrativen Maßnahmen, die der Instandhaltung des abgenutzten Lagerbestandes bzw. dessen Versetzung in einen funktionsfähigen Stand zum Ziel haben. Instandhaltung ist der Oberbegriff für die vier Grundmaßnahmen Wartung, Inspektion, Instandsetzung und Verbesserung [DIN 31051 2019]. Das Ziel von Instandhaltungsmaßnahmen (z.B. Reinigen, Schmierem oder Nachstellen) ist es, den Verschleißbestand zu verlangsamen (siehe Abbildung 6).

Die Wartung umfasst alle Maßnahmen, die dazu dienen, den Soll-Zustand zu erhalten oder die Verschlechterung eines vorhandenen Verschleißbestandes von Beobachtungseinheiten zu verzögern, wie z.B. Schmierung.

Zur Inspektion gehören alle Maßnahmen zur Feststellung und Beurteilung des Ist-Zustandes, einschließlich der Ermittlung von Verschleißursachen und Ableitung notwendiger Konsequenzen für die zukünftige Verwendung (z.B. Messen, Prüfen oder Diagnostizieren) [DIN 31051 2019].

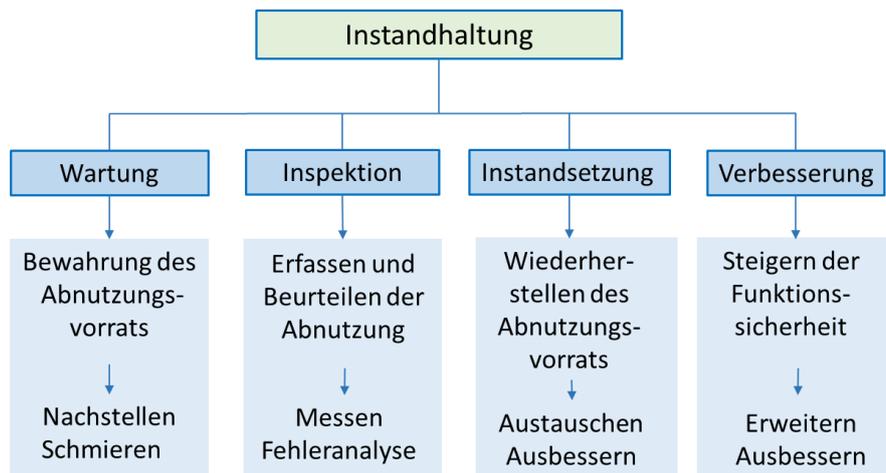


Abbildung 6: Instandhaltungsarten in Anlehnung an [DIN 31051 2019]

Eine Instandsetzung ist die Rückführung eines Bauteils in den betriebsfähigen Zustand, d.h. in den Zustand vor dem Ausfall ohne Verbesserung, z.B. Austausch eines Bauteils durch ein gleichwertiges Ersatzteil [DIN 31051 2019]

Als Verbesserung wird die Kombination aller technischen, administrativen und kontrollierenden Maßnahmen bezeichnet, die die funktionale Sicherheit des betreffenden Bauteils erhöhen, ohne die Funktion des Bauteils zu verändern, z. B. die Beseitigung einer Schwachstelle durch Änderung der Bauteilkonstruktion [DIN 31051 2019].

Die Abbildung 7 zeigt einen idealtypischen Verlauf der Abnutzungskurve eines technischen Systems sowie die Wirksamkeit der vier Instandhaltungsmaßnahmen. Nach der Herstellung, Wartung, Instandsetzung oder Verbesserung befindet sich das System zunächst in seinem Ausgangszustand. Sie erfüllt unter bestimmten Bedingungen die ihr bestimmte Aufgabe im Funktionsbereich. Ohne Instandhaltungsmaßnahmen ist die Abnutzungsreserve zu einem bestimmten Zeitpunkt verbraucht. Dies wird als Systemausfall bezeichnet. Es befindet sich dann in einem Störungszustand und ist nicht mehr funktionsfähig.

Um das System wieder in einen funktionsfähigen Zustand zu versetzen, ist eine ungeplante Instandhaltung erforderlich.

Laut [Leidinger 2014] sollen mit den Instandhaltungsmaßnahmen vor allem die folgenden Ziele erreicht werden:

- Erhöhung der Sicherheit - Es soll keine Gefahr von der Systemanlage ausgehen.

- Erhöhung der Verfügbarkeit - Die Systemanlage muss jederzeit in Betrieb gesetzt werden können.
- Erhöhung der Zuverlässigkeit - Das System muss störungsfrei arbeiten können.
- Erhalten des Werts - Das System soll eine hohe Restlebenserwartung aufweisen.

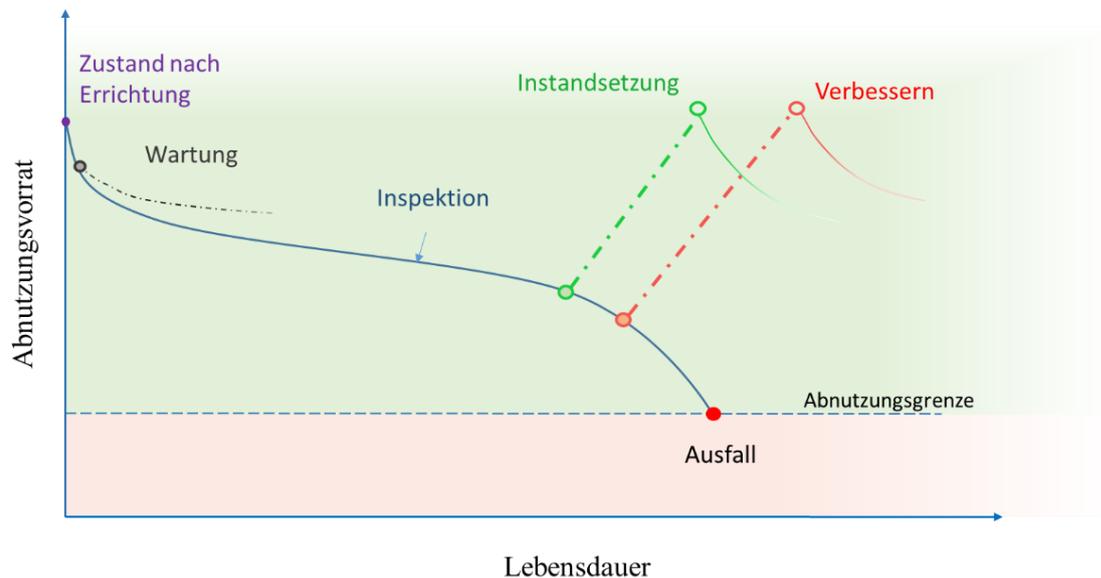


Abbildung 7: Idealtypischer Verlauf einer Abbaukurve (Modifiziert nach [DIN 31051 2019])

Die Ziele werden entweder als gleichrangig betrachtet oder einzeln priorisiert. Mögliche Nebenziele sind nach [Pawellek 2016] die Verbesserung der Produktqualität und der Kapazitätsauslastung oder auch die Reduzierung der Auftragsdurchlaufzeiten. Das grundsätzliche Ziel der Instandhaltung kann mit unterschiedlichen Strategien realisiert werden. Die Instandhaltungsstrategie regelt, zu welchem Zeitpunkt welche Arten von Instandhaltungsmaßnahmen durchgeführt werden sollen. Bei der Festlegung der Instandhaltungsstrategie sind rechtliche, sicherheitstechnische, technische, produktionstechnische und wirtschaftliche Aspekte zu berücksichtigen [Schenk 2010].

In der Abbildung 8 ist eine Klassifizierung von Instandhaltungsstrategien dargestellt. In der Literatur wird eine Vielzahl von weiteren Ansätzen vorgestellt, die sich auf die dargestellten Grundtypen beziehen [Pawellek 2016]. Nach [DIN 31051 2019] wird grundsätzlich zwischen zwei Arten unterschieden: der präventiven und der korrektiven Instandhaltung. In der Literatur werden alternativ auch die Begriffe proaktive oder reaktive Instandhaltung verwendet [Schröder 2010].

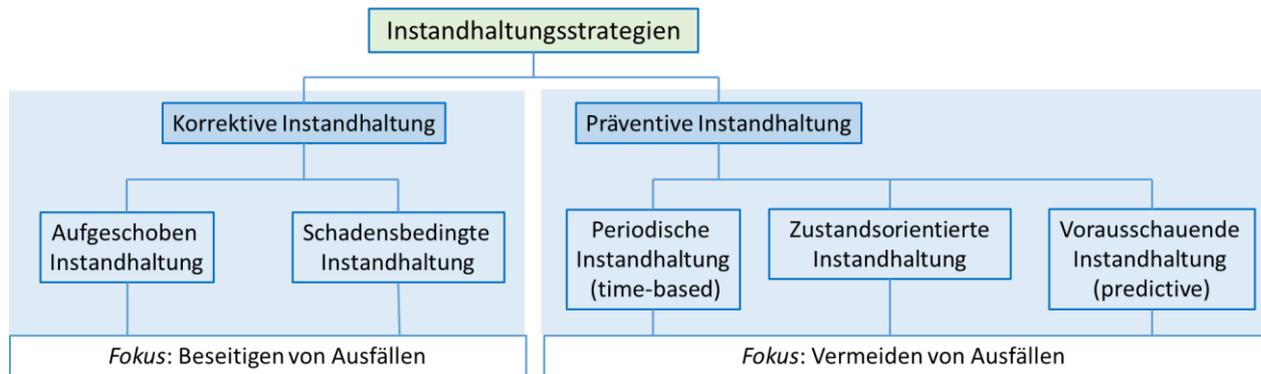


Abbildung 8: Einteilung der maßnahmenbezogenen Instandhaltungsstrategien Instandhaltung in Anlehnung an [Schenk 2010]

## 2.4.2 Korrektive Instandhaltung

Als korrektive Instandhaltung wird eine Instandhaltungsmaßnahme erst nach dem Ausfall eines Bauteils oder bei Erreichen einer definierten Schadensgrenze durchgeführt. Auf eine zwischenzeitliche Wartung oder Inspektion wird konsequent verzichtet. Diese Vorgehensweise ist allerdings keine Strategie im Sinne des Wortes. Eine Strategie ist die langfristige, geplante Verfolgung eines Ziels. Das beschriebene Vorgehen erfordert jedoch spontanes und schnelles Arbeiten und verlangt vom Instandhaltungspersonal besondere Fähigkeiten, den aktuellen Zustand der Anlage wahrzunehmen und zu beurteilen. Zudem ist das Vorgehen unfallträchtig und in vielen Fällen aufgrund des vorhandenen Gefahrenpotenzials, das von vielen technischen Systemen ausgeht, unzulässig.

Diese Strategie scheint zunächst die kostengünstigste zu sein, da kein Planungsaufwand erforderlich ist und nur das ausgetauscht wird, was tatsächlich defekt ist. Auf diese Weise wird die vorhandene Abnutzungsreserve eines jeden Bauteils tatsächlich genutzt, da die Anlagen so lange betrieben werden, bis ein Schaden auftritt. Tritt ein Schaden auf, so ist dies meist ein plötzliches, ungeplantes Ereignis. Die Folge ist, dass die Instandsetzung unter großem Zeitdruck erfolgt und die notwendigen Instandhaltungsressourcen wie Personal, Ersatzteile, Werkzeuge, Ausrüstungen und Hilfsmittel oft nicht sofort verfügbar sind oder für den Schadensfall in Reserve gehalten werden müssen. Da in diesem Fall die Schadensgrenze in der Regel unterschritten wird, kommt es häufig auch zu einer unzulässigen Schädigung anderer Anlagenteile, die zu unvorhersehbaren Folgeschäden an anderen Anlagen, Maschinen, Komponenten, Baugruppen oder Teilen führen kann [Straube 1988].

Die Strategie der ungeplanten korrektiven Instandhaltung führt damit im Vergleich zu den anderen Instandhaltungsstrategien zu den höchsten Stillstandszeiten und den höchsten Ausfallfolgekosten. Ein weiterer Nachteil ist die schlechte Planungsmöglichkeit von Instandhaltungsmaßnahmen. Aufgrund der mangelnden Prognosefähigkeit von Störfällen kann keine Aussage über eine garantierte Anlagenverfügbarkeit getroffen werden [Straube 1988].

Die Anwendung dieser Strategie sollte daher auf solche Anlagen bzw. deren Komponenten beschränkt werden, die im Unternehmen von untergeordneter Bedeutung sind. Dies sind in der Regel Anlagenteile, deren Ausfall keine sicherheitstechnischen Anforderungen tangiert und deren Ausfall zu keiner Produktionsunterbrechung führt. Diese Strategie sollte nur dann verfolgt werden, wenn redundante Systeme vorhanden sind oder alle Instandhaltungsressourcen zur Schadensbehebung schnell zur Verfügung stehen [Straube 1988].

Dieses Konzept hat vor allem den Vorteil, dass eine möglichst lange Lebensdauer des Bauteils erreicht wird und der Planungsaufwand vernachlässigbar ist. Dabei wird ein hoher Anteil an kurzfristigen Maßnahmen und die daraus resultierenden Stillstandszeiten durch nicht vorhersehbare Ausfälle angenommen. Diese Vorgehensweise sollte verfolgt werden, wenn eine schnelle Verfügbarkeit aller Instandhaltungsressourcen gewährleistet ist, ein redundantes System vorhanden ist oder keine sicherheitskritische Komponente im Einsatz ist [Leidinger 2014].

## **2.4.3 Präventive Instandhaltung**

### **2.4.3.1 Periodische Instandhaltung**

Bei der Strategie der periodischen präventiven Wartung erfolgt eine Reparatur oder ein Austausch von Systemkomponenten nach definierten Nutzungsintervallen, unabhängig vom tatsächlichen Bauteilzustand. Diese können zeitlich oder ereignisbezogen definiert werden, z.B. nach Kalenderzeit, Betriebsstunden, Stückzahlen, gefahrenen Kilometern usw. (vgl. Abbildung 9). Die Anwendung dieser Strategie ist dann zu empfehlen, wenn Sicherheits- und Umweltauflagen dies erfordern oder wenn die erwartete Lebensdauer von Komponenten sehr gut bekannt ist. Das Risiko eines Anlagenausfalls ist deutlich geringer als bei der reaktiven Instandhaltungsstrategie, da die Maßnahmen über die Zeit oder Nutzungsintervalle sehr gut planbar sind und somit die notwendigen Instandhaltungsressourcen zur Verfügung gestellt werden können. Die Instandhaltungsmaßnahmen werden planmäßig in Zeitfenstern durchgeführt, in denen keine Produktion stattfindet. Die Ausführungszeiten selbst werden durch einen hohen Wiederholungsgrad und planbare Abläufe optimiert, was zu einer geringen Anlagenstillstandszeit führt [Matyas 2002].

Im Gegensatz dazu steht die Tatsache, dass bei einer periodischen vorbeugenden Instandhaltungsstrategie der Austausch einer Komponente meist zu früh erfolgt. Dadurch wird die Verschleißreserve der Komponenten verschwendet und der Verbrauch von Ersatzteilen und Material erhöht. Eine zunehmende Anwendung der präventiven Methode führt somit einerseits zu einer Reduzierung der Ausfallkosten, andererseits aber zu einer Erhöhung der Präventionskosten. Dies zeigt einen Zielkonflikt zwischen der reaktiven und der periodischen präventiven Strategie. Es wird deutlich, dass die reaktive Instandhaltungsstrategie zwar die Verschleißreserve voll ausschöpft, aber die Nachteile dieser Strategie die Nachteile der periodisch-präventiven Instandhaltungsstrategie überwiegen, vor allem wegen der Unplanbarkeit der Maßnahmen und der zu erwartenden Folgeschäden an anderen Bauteilen durch Unterschreitung akzeptierter Schadensgrenzen.

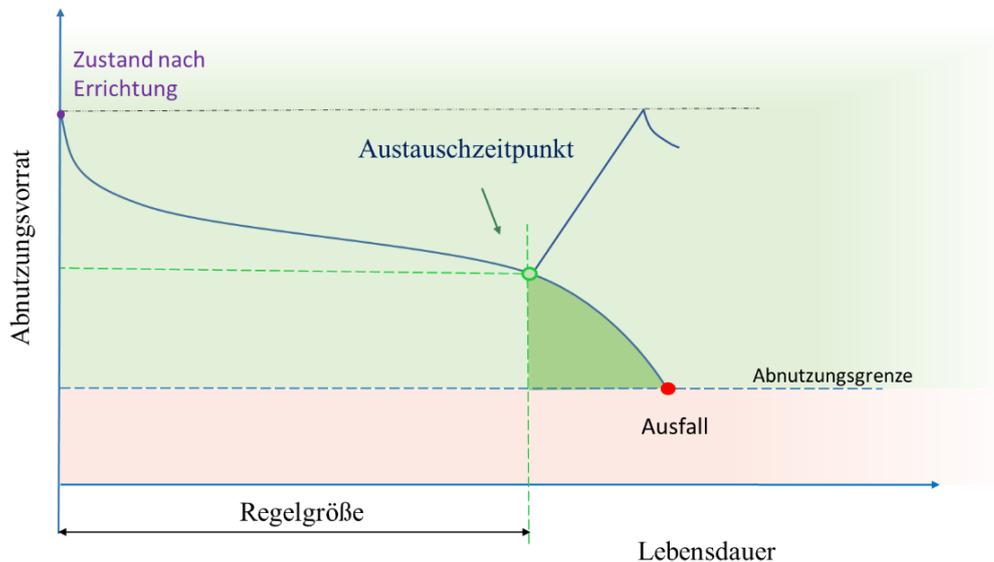


Abbildung 9: Verlauf des Abnutzungsvermögens bei periodischer Instandhaltung in Anlehnung an [Schenk 2010]

Bei der periodischen vorbeugenden Instandhaltung kann es vorkommen, dass das Austauschintervall für ein Teil aufgrund erhöhter Anlagenauslastung früher erreicht wird oder durch eine falsche Berechnung zu lang angesetzt ist. In diesem Fall werden die Vorteile der Strategie nicht ausgenutzt, weil es zu ungeplanten Ausfällen von Komponenten kommt.

Genau wie bei der reaktiven Instandhaltungsstrategie müssen in diesen Fällen ungeplante Maßnahmen ergriffen werden. Dies hat die gleichen negativen Folgen wie bei der reaktiven Instandhaltungsstrategie. In der Folge kann jedoch davon ausgegangen werden, dass die Ausfallzeit im Vergleich zu einer rein reaktiven Strategie wesentlich geringer ist.

Die Herausforderungen bei der Umsetzung einer periodischen präventiven Instandhaltungsstrategie liegen darin, dass alle Komponenten eines technischen Systems ein unterschiedliches Ausfallverhalten und damit auch unterschiedliche Lebenserwartungen haben. Um die Ausnutzung der Verschleißreserve der Komponenten zu optimieren, müssen periodische Instandhaltungsmaßnahmen durchgeführt werden [Schröder 2010].

Bei der vorbeugenden Instandhaltung werden verschleißanfällige Komponenten ausgetauscht, bevor die Verschleißreserve erschöpft ist. Dadurch können die Leistungsfähigkeit und Zuverlässigkeit technischer Systeme erhöht und Kosten durch ungeplante Wartungsarbeiten reduziert werden. Die Bestimmung des idealen Austauschzeitpunkts erfordert jedoch Planung [Schröder 2010].

Für die Ermittlung des Austauschzeitpunkts einer technischen Anlage gibt es zwei Ansätze: die zustandsorientierte und die vorausschauende Instandhaltung.

### 2.4.3.2 Präventive zustandsorientierte Instandhaltung

Die geringsten Ausfallzeiten bei sehr guter Ausnutzung der Verschleißreserve werden durch die zustandsorientierte Instandhaltungsstrategie erreicht, die oft auch als Inspektionsstrategie

bezeichnet wird. Bei dieser Strategie wird zusätzlich zur geplanten Instandhaltung regelmäßig der Zustand der Anlagen überprüft, um mögliche Fehlerquellen rechtzeitig zu erkennen und zu beseitigen.

Der wesentliche Unterschied zu den anderen Strategien besteht darin, dass sich die Regelgröße zur Auslösung von Instandhaltungsmaßnahmen ändert. Während bei der periodischen vorbeugenden Instandhaltung diese Regelgröße letztlich die Zeit ist, handelt es sich hier um den Anlagenzustand, der z.B. über die Verschleißreserve beschrieben werden kann (vgl. Abbildung. 10) [Schenk 2010].

Mit der zustandsorientierten Instandhaltungsstrategie ist es möglich, Wartungs- und Instandsetzungsmaßnahmen genau zu dem Zeitpunkt durchzuführen, zu dem der Verschleißbestand es erfordert. Dies setzt allerdings die Bereitstellung von aktuellen Zustandsinformationen über die technische Systemanlage voraus. Die Wartungsintervalle werden an die Veränderungen des Verschleißbestandes angepasst.

Der Inspektionsaufwand unterscheidet sich zwischen der einfachen Betrachtung der Komponente, der Demontage oder dem Einsatz von diagnostischen Methoden (z.B. Temperatur- oder Schwingungsmessung) [Leidinger 2014].

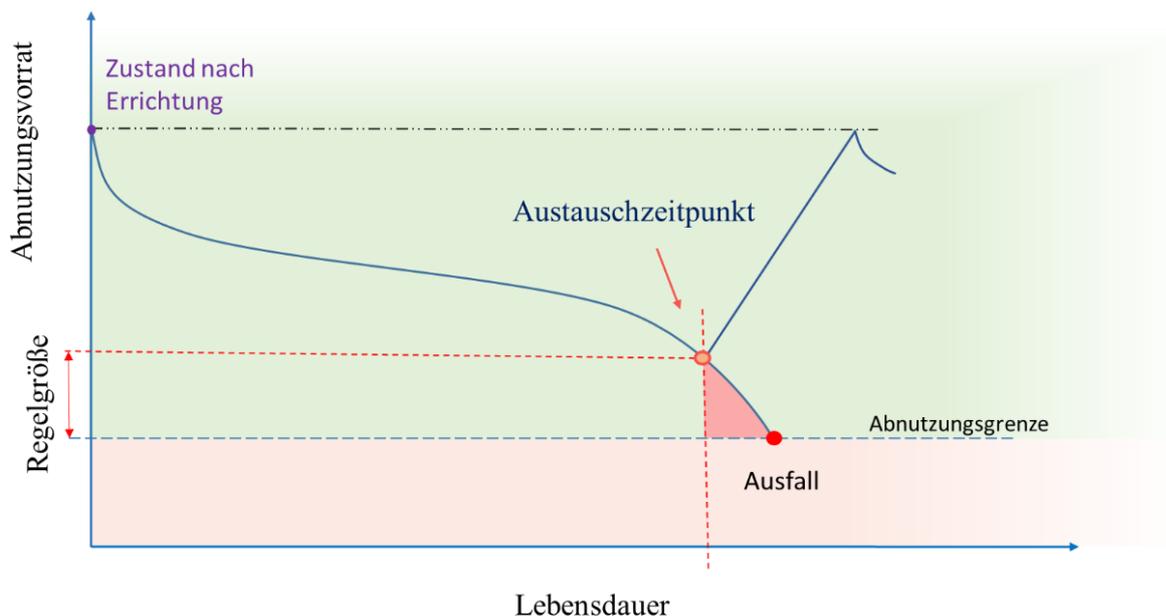


Abbildung 10: Verlauf des Abnutzungsvermögens bei präventive zustandsorientierte in Anlehnung an [Schenk 2010]

Die Informationen bezüglich des Anlagenzustandes ermöglichen es, die Lebensdauer einer Anlage optimal zu nutzen, im Gegensatz zur vorausschauenden Instandhaltung. Je genauer die Bestimmung, desto höher der Nutzen. Ein Nachteil sind die hohen Anschaffungs- und Entwick-

lungskosten, die bei der Zustandsüberwachung mittels Sensoren entstehen. Zudem sind in vielen Fällen umfangreiche Kenntnisse über das Anlagenverhalten erforderlich [Jardine et al. 2006].

Ein wesentlicher Aspekt der zustandsorientierten Instandhaltung ist die Inspektion. Das setzt allerdings die Bereitstellung von aktuellen Zustandsinformationen über die technische Anlage voraus. Die Instandhaltungsintervalle werden dabei jeweils auf die Veränderungen im Abnutzungsvorrat angepasst. Der Inspektionsaufwand variiert zwischen dem bloßen Betrachten, der Demontage eines Bauteils oder dem Einsatz von Diagnosetechniken (z.B. Temperatur- oder Schwingungsmessung) [Leidinger 2014].

Durch die Informationen über den Anlagenzustand kann die Lebensdauer einer Anlage im Gegensatz zu einer vorausschauenden Instandhaltung optimal genutzt werden. Je genauer die Bestimmung, desto höher der Nutzen. Ein Nachteil sind die hohen Anschaffungs- und Entwicklungskosten, die bei der Zustandsüberwachung mittels Sensoren entstehen. Zudem sind in vielen Fällen umfangreiche Kenntnisse über das Anlagenverhalten erforderlich [Jardine et al. 2006].

Die Zielsetzung ist, mit möglichst wenigen Sensoren eine maximale Anzahl von Komponenten zu überwachen, um die Investitionskosten für ein CMS gering zu halten und keine neuen fehlerhaften Komponenten in das System zu bekommen. Die zustandsorientierte Instandhaltung findet Anwendung, wenn die Veränderung des Verschleißes messbar, d.h. technisch machbar ist und eine messtechnische Erfassung zustandsbestimmender Parameter auch wirtschaftlich vertretbar ist. Inspektionen ermöglichen es, Schäden rechtzeitig zu erkennen und das Auftreten von Schäden sehr genau vorherzusagen. Durch geeignete zustandserhaltende Maßnahmen werden Anlagenausfälle weitgehend vermieden, was zur Erhöhung der Zuverlässigkeit beiträgt. Darüber hinaus steht dem Instandhaltungsbereich ein ausreichend langer Zeitraum zur Verfügung, um die Schadensbehebung durch gut vorbereitete Reparaturmaßnahmen schnell und kostengünstig zu einem geeigneten Zeitpunkt durchzuführen. Dies wirkt sich unmittelbar auf die Instandhaltungskosten aus.

### **2.4.3.3 Präventiv vorausschauende Instandhaltung**

Als eine folgerichtige Weiterentwicklung der zustandsorientierten Instandhaltungsstrategie gilt die vorausschauende Instandhaltungsstrategie. Bei dieser Strategie wird eine voraussichtliche Zustandsveränderung von Komponenten noch früher erkannt. Ziel ist es, potentielle (versteckte) Störungen bereits zu identifizieren und deren weitere Entwicklung gezielt zu verhindern. Ausgangspunkt für die Anwendung dieser Instandhaltungsstrategie ist die Definition von Aufgaben und möglichen Fehlfunktionen. Die Funktionen von technischen Systemen lassen sich im Wesentlichen in drei Kategorien einteilen [Moubray 1996]. Je genauer eine Abschätzung der Lebensdauer ist, desto präziser kann eine Maßnahme durchgeführt werden, ohne die Produktion wesentlich zu stoppen. Ein optimiertes Ersatzteilmanagement ist von Bedeutung, da Ersatzteile direkt vor Ort angefordert werden und keinen ungenutzten Lagerplatz belegen. Der größte Vorteil einer durch Störungsprognosen unterstützten Instandhaltung liegt insbesondere bei altersbedingten Ausfällen (vgl. Abbildung 11) [Schenk 2010].

Durch den regelmäßigen technischen Verschleiß kommt es in dieser Phase zu einer messbaren Systemdegradation, die mit Hilfe von Wissen über das Verschleißverhalten vorhergesagt werden kann [Hodapp 2009].

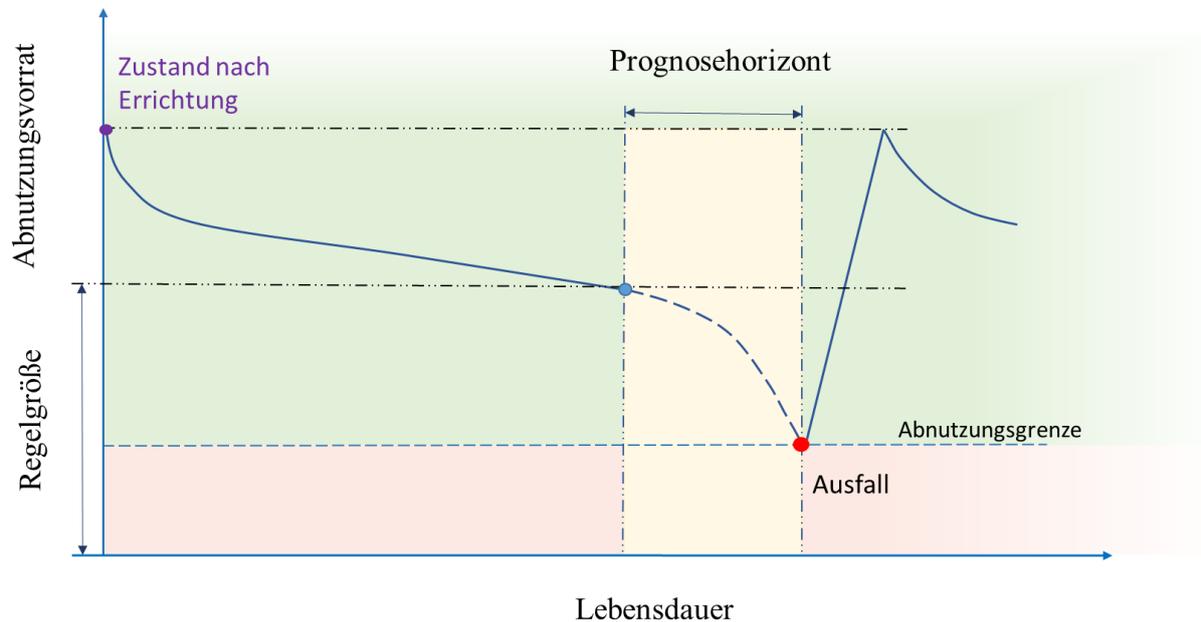


Abbildung 11: Verlauf des Abnutzungsgrades bei präventive zustandsorientierte in Anlehnung an [Schenk 2010].

#### 2.4.3.4 Auswahl der Instandhaltungsstrategie

Die Auswahl einer optimalen Instandhaltungsstrategie ist von großer Bedeutung, um die definierten Ziele zu erreichen. In der Regel erfolgt die Auswahl der Strategie auf der Anlagenebene, bei komplexen technischen Systemen sogar auf der Komponentenebene. Die Entscheidung für oder gegen eine der dargestellten Strategien hängt daher immer von der konkreten Situation der technischen Anlage und der Instandhaltungspolitik im Unternehmen ab. Daraus folgt, dass es keine allgemeingültige Instandhaltungsstrategie geben kann, sondern dass im Unternehmen immer ein Hybrid aus den möglichen Strategien anzutreffen ist. Es ist daher wichtig, diesen optimaler Weise zu gestalten [Hodapp 2009].

In der Regel wird der Instandhaltungsaufwand so gewählt, dass die Summe aus Ausfallzeiten und Instandhaltungskosten möglichst gering ist. Dieser Zusammenhang ist in Abbildung 12 dargestellt. Wird ein korrektives Konzept verfolgt, d.h. die technische Anlage wird mit minimalem Aufwand betrieben, steigt das Risiko eines ungeplanten Ausfalls, so dass die Ausfallkosten überproportional steigen. Dazu gehören nicht nur die Kosten für Ersatz und Betriebsunterbrechung, sondern auch Folgeschäden durch Reputationsverluste oder Prozesse im Zusammenhang mit dem Ausfall [Hodapp 2009].

Die Wahrscheinlichkeit eines Ausfalls ist umso geringer, je mehr Aufwand betrieben wird. Voraussetzung dafür ist allerdings, dass die Präventionskosten mit zunehmendem Umsetzungsgrad der vorausschauenden Instandhaltung überproportional ansteigen. Mit Hilfe der zustandsorientierten Instandhaltung versuchen Unternehmen, den Kostentrog zwischen den beiden Extremen zu finden [Hodapp 2009].

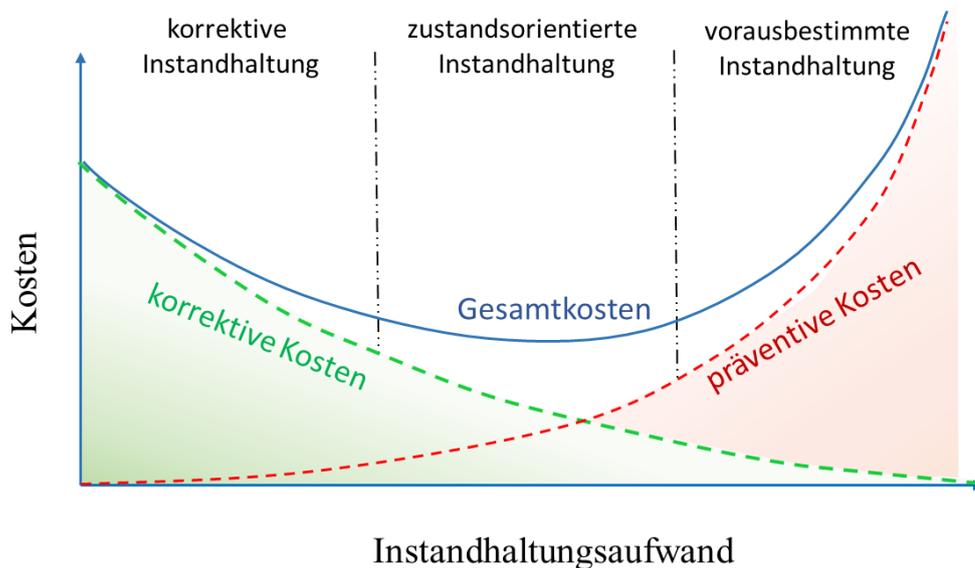


Abbildung 12: Entwicklung der entstehenden Gesamtkosten über den Instandhaltungsaufwand und Zuordnung der drei Instandhaltungsarten in Anlehnung an [Hodapp 2009]

Jede vorgestellte Instandhaltungsmethode hat ihre Vorteile und Nachteile. Ein optimaler Ansatz kann daher durch die Kombination der verschiedenen Arten erreicht werden. Dies bedeutet, dass in einem Unternehmen in der Regel verschiedene Instandhaltungsarten für verschiedene Anlagen, Systeme, Komponenten und Bauteile gleichzeitig vorhanden sind. Damit wird versucht, eventuelle Nachteile eines Konzeptes zu umgehen [Leidinger 2014].

## 2.5 Methoden der prädiktiven Instandhaltung

### 2.5.1 Machine Learning

In der Forschung ist das maschinelle Lernen ein Teilbereich der "Künstlichen Intelligenz (KI)". Der Einsatz von maschinellem Lernen ist inzwischen allgegenwärtig und unterstützt vor allem solche Tätigkeiten, die eine regelmäßige Datenverarbeitung erfordern. Im Kontext des maschinellen Lernens erweisen sich die komplizierten menschlichen Aufgaben als relativ einfach, während sich die einfachen menschlichen Aufgaben als überraschend schwierig zu lösen herausstellen (Moravec'sches Paradox) [Leidinger 2014]. Aufgrund der zunehmenden Datenmenge ist es schwierig, wertvolle Informationen aus einfachen Datenanalysen zu extrahieren. Hier unterstützt maschinelles Lernen dabei, diese Informationen herauszufiltern, Muster zu erkennen und Vorhersagen zu ermöglichen.

Durch den Einsatz von maschinellen Lernen können moderne KI-Systeme so trainiert werden, dass sie sich an veränderte Umweltbedingungen anpassen, sich entsprechend optimieren und damit deutlich bessere Ergebnisse erreichen als frühere KI-Systeme, die auf genau beschriebenen und festgelegten Regeln basierten [Brühl 2019]. Zu den klassischen ML-Aufgaben gehören Klassifizierung und Regression, Lernen von Assoziationen, Clustering und andere Aufgaben des maschinellen Lernens wie Verstärkungslernen, Lernen von Rangfolgen und Strukturvorhersage [Lawrynowicz 2014].

Das Endziel der Künstlichen Intelligenz ist die Erschaffung eines rationalen und autonomen Modells, der z. B. in der Lage ist, Daten über Sensoren relevante Umweltinformationen zu erfassen und daraus optimale Schlussfolgerungen im Hinblick auf bestimmte Leistungsparameter zu ziehen. Diese wiederum werden über Aktoren in physische (Robotik) oder virtuelle Aktionen (z.B. Chatbots) umgesetzt [Brühl 2019].

Zur Entwicklung eines KI-Systems, das selbständig Entscheidungen trifft oder als Entscheidungshilfe dienen kann, sind zwei Hauptkomponenten erforderlich:

- Daten: Durch Messung oder Beobachtung erhobene Zahlen, Werte und Fakten, die von Menschen oder Maschinen (Computern) gelesen oder verarbeitet werden können [Leidinger 2014]
- Algorithmus: eine begrenzte Menge von einzigartigen Operationen, die vom Menschen, aber auch von Maschinen (Computern) ausgeführt werden können. Algorithmen des maschinellen Lernens sorgen dafür, dass die Ergebnisse von Aktionen oder Veränderungen der Umgebungsbedingungen erfasst werden und so neue Informationen entstehen, die in einem iterativen Prozess die Zielerfüllung des Systems schrittweise optimieren [Brühl 2019].

Auf der Basis eines Datensatzes, der so genannten Trainingsdaten, können ML-Algorithmen ein Modell erstellen, das Vorhersagen ermöglicht und Entscheidungen trifft [Ayodele 2010].

Unabhängig von der spezifischen Anwendung müssen auf maschinellem Lernen basierende Modelle vor ihrem Einsatz mit verfügbaren Basisdaten trainiert und getestet werden, um die bestmögliche Vorhersagequalität zu erreichen [Brühl 2019].

Verschiedene Methoden des maschinellen Lernens wurden spezialisiert für bestimmte Situationen und Datensätze entworfen, haben aber alle eine wichtige Gemeinsamkeit, nämlich sie können aus Erfahrungen lernen und sich stetig verbessern, ohne dass sie im Einzelnen dazu programmiert werden müssen. Diese sind Überwachtes Lernen (englisch: supervised machine learning), Unüberwachtes Lernen (englisch: unsupervised machine learning), Teilüberwachtes Lernen (englisch: semi-supervised learning) und Verstärkendes Lernen (englisch: reinforcement learning) (Abbildung 13).

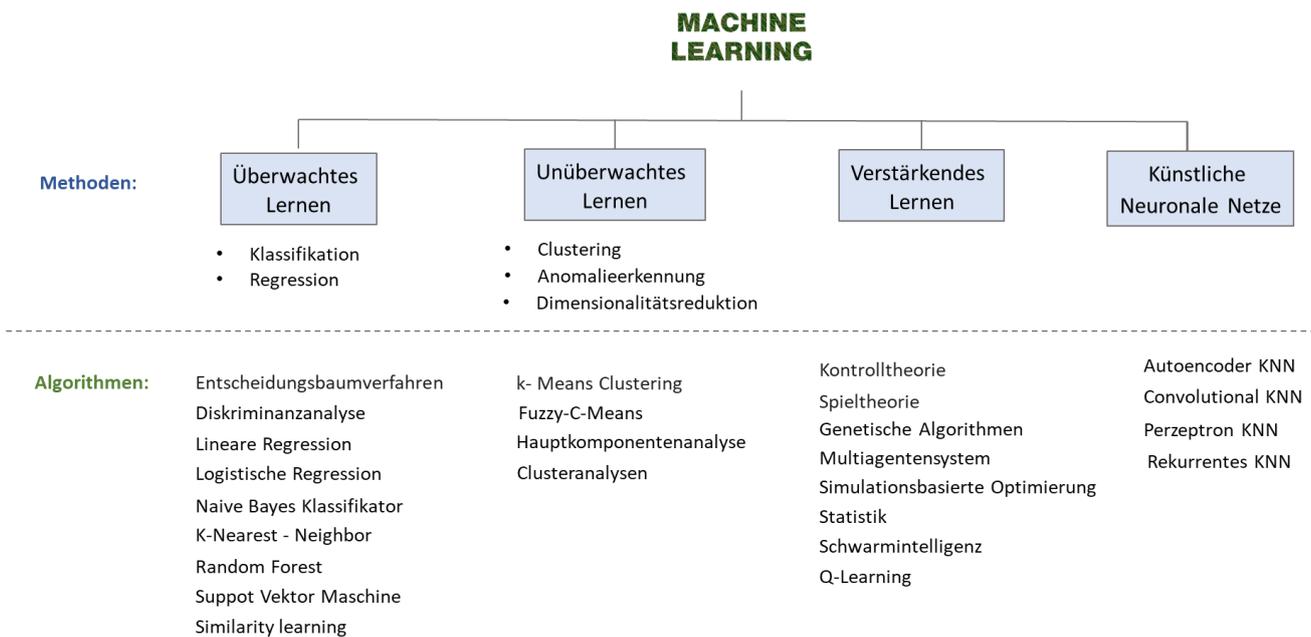


Abbildung 13: Methoden des maschinellen Lernens in Anlehnung an [Kudelina et al. 2021]

In enger Verbindung mit den Methoden des maschinellen Lernens ist das Data Mining zu sehen, das zur Entdeckung neuer Datenmuster in großen Datensätzen eingesetzt werden kann. Der Hauptunterschied besteht darin, dass beim maschinellen Lernen der Schwerpunkt auf adaptivem Verhalten und operativem Einsatz liegt, während sich Data Mining auf die Verarbeitung großer Datensätze und die Entdeckung unbekannter Muster bezieht. Auf diese Weise können ML-Algorithmen auf Basis eines Datensatzes, der so genannten Trainingsdaten, ein Modell erstellen, das Vorhersagen und Entscheidungen trifft [Ayodele 2010].

## 2.5.2 Überwachtes Lernen

Beim überwachten maschinellen Lernen (engl. supervised learning) wird anhand von Eingabe-Ausgabe-Paare eine Funktion bestimmt, die gegebene Eingabewerte auf bekannte Zielwerte oder Strukturen abbildet. Es gibt somit eine explizite Zielgröße, für die in den Trainingsdaten die wahren Werte bekannt sind. In vielen Fällen kann es schwierig sein, die Ausgaben automatisch zu erfassen, so dass sie von einem menschlichen "Supervisor" bereitgestellt werden müssen. Die Zielwerte werden beim maschinellen Lernen typischerweise als Label bezeichnet. Ist das Label nominal, dann handelt es sich um eine Klassifikation. Bei metrischer Zielgröße liegt eine Regression oder auch numerische Vorhersage vor [Leidinger 2014]. Aus Daten wird ein Trainings- und ein Testdatensatz gebildet. Das Modell enthält letztendlich den Zusammenhang zwischen Trainingsdaten und ihren dazugehörigen Ergebnissen in abstrahierter Form. Während des Trainings überprüft der Algorithmus immer wieder die selbst vorhergesagten Ergebnisse mit den vorhandenen Ergebnissen der Trainingsdaten und passt sein Modell entsprechend an. Das Training wird so oft wiederholt, bis das Modell die Ergebnisse mit einer ausreichenden Genauigkeit vorhersagen kann. Das so entwickelte Modell lässt sich dann für Vorhersagen mit dem Testdatensatz verwenden.

Die zentrale Aufgabe beim maschinellen Lernen besteht darin, dass bei neuen, bisher unbekanntem Daten gute Ergebnisse erzielen müssen - und nicht nur bei Trainingsdaten. Die Fähigkeit, eine gute Leistung bei zuvor nicht beobachteten Daten zu erzielen, wird als Generalisierung bezeichnet.

Überwachten maschinellen Lernen wird weiter in zwei Unterkategorien eingeteilt: Klassifikations- und Regressions-Fragestellungen (Abbildung 14) Unter Klassifikation versteht man die Zugehörigkeit eines Merkmals zu einer bestimmten Gruppe (Klasse). Der Zielwert wird durch die Klassenzugehörigkeit repräsentiert. Im maschinellen Lernen ist die Klassifizierung sicherlich die am häufigsten betrachtete Aufgabe. Im Rahmen der binären Klassifikation werden die Objekte in eine von zwei Klassen eingeordnet [Leidinger 2014].

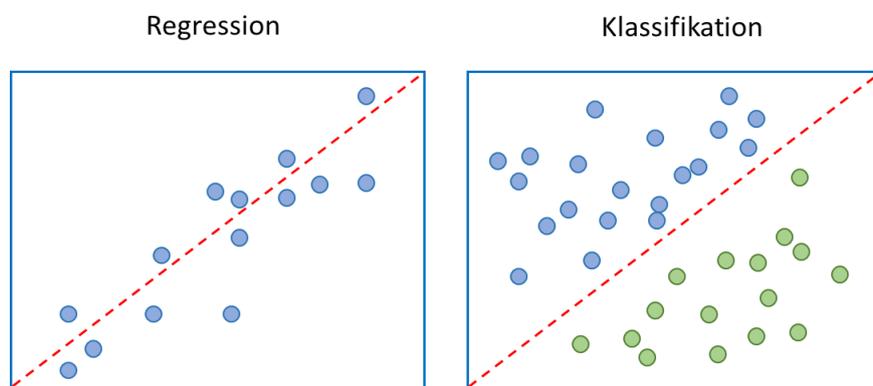


Abbildung 14: Darstellung der Regressions- und Klassifikationsverfahren in Anlehnung an [Leidinger 2014].

Häufig werden beim maschinellen Lernen die beiden Klassen der binären Klassifikation als positiv und negativ bezeichnet. Positiv steht dabei nicht unbedingt für eine vom Nutzer als positiv empfundene Eigenschaft, sondern für die Klasse, die für die Anwendung von größerer Relevanz ist. Bei expliziter Zuordnung der Klassen zu den Wahrheitswerten richtig und falsch, spricht man auch von einem Konzept und die Aufgabe wird als Konzeptlernen bezeichnet [Leidinger 2014].

Zur Beurteilung der Leistung eines Klassifikators werden die bekannten wahren Werte der Zielgröße mit den Vorhersagen des Klassifikators verglichen. Da die Zielgröße nominal ist, gibt es bei der binären Klassifizierung nur die beiden Möglichkeiten, dass der vorhergesagte Wert richtig ist oder nicht [Leidinger 2014]. Anwendungsfälle finden sich z. B. bei der Spracherkennung, bei der Kreditwürdigkeitsprüfung, bei der Erkennung von Objekten auf Bildern (Autos, Lastwagen, Fußgänger, Straßenschilder, etc.) [Brühl 2019].

Die Regression ist eine Vorhersage von metrischen Variablen (Zielvariable). Die Regressionsaufgabe ist klar spezifiziert durch die Erstellung einer Funktion, die die Zielgröße auf der Grundlage der Eingabewerte errechnet. Der Unterschied bei den Lernverfahren liegt insbesondere in der Art bzw. Komplexität der verwendbaren Funktionen und in der Art und Weise, wie

eine Überanpassung an die Trainingsdaten vermieden werden kann. Bei einer numerischen Zielgröße ist es offensichtlich, dass eine hinreichend komplexe Funktion jeden Datenpunkt im Trainingssatz abbilden kann, solange es keine Widersprüche in den Daten gibt [Leidinger 2014]. Anwendungsfelder finden sich z. B. in der Berechnung der Wahrscheinlichkeit eines Kreditausfalls auf der Grundlage persönlicher Daten, die dynamische Preisgestaltung oder die Wahrscheinlichkeit, dass es morgen regnet [Kudelina et al. 2021].

Die am häufigsten verwendeten überwachten Algorithmen für maschinelles Lernen sind beispielsweise lineare und logistische Regression, Entscheidungsbäume, k-Nächste-Nachbarn-Algorithmen (k-NN), Support-Vektor-Maschinen (SVM) und neuronale Netze.

### 2.5.2.1 Logistische Regression

Die logistische Regression [Kalisch et al. 2021] ist das wohl am weitesten verbreitete statistische Lernverfahren für binäre Klassifikationsprobleme.

Das Ziel ist die Modellierung einer binären Zielvariablen  $Y \in \{0,1\}$ .  $Y$  nimmt nur die zwei Werte 0 oder 1 an, die für die zwei möglichen Zustände der Zielvariablen stehen (z. B. 0: "Ausfall" und 1: "Betriebsbereit"). Als Verteilung eignet sich die Bernoulli-Verteilung, die nur einen Parameter, die Erfolgswahrscheinlichkeit  $p \in [0,1]$ , hat. Sie beträgt  $p = P(Y = 1)$ , d. h.

$$Y \begin{cases} 1 & \text{Wahrscheinlichkeit } p \\ 0 & \text{Wahrscheinlichkeit } 1 - p. \end{cases}$$

Im direkten Zusammenhang mit der Bernoulli-Verteilung steht die Binomialverteilung. Die Binomialverteilung modelliert die Anzahl der Erfolge in  $n$  unabhängigen Bernoulli-Verteilungen ("Experimenten") mit der Erfolgswahrscheinlichkeit  $p$ . Dies wird mit  $\text{Bin}(n, p)$  beschrieben. In diesem Sinne kann die Bernoulli-Verteilung auch als Binomialverteilung mit  $n = 1$  interpretiert werden [Kalisch et al. 2021].

Das Modell modelliert die Erfolgswahrscheinlichkeit  $p$  als Funktion der erklärenden Variable  $x$ , d. h.

$$Y \sim \text{Bernoulli}(p(x)).$$

Wenn die logistische Regression als zweistufiges Modell geschrieben werden soll, fehlt nur noch der funktionale Zusammenhang zwischen der Erfolgswahrscheinlichkeit  $p(x)$  und der erklärenden Variable  $x$ . Der gleiche Ansatz wie bei der linearen Regression liegt auf der Hand: Die Gewinnwahrscheinlichkeit wird als lineare Funktion der erklärenden Variable (linearer Prädiktor) modelliert, d. h.  $p(x) = \beta_0 + \beta_1 \cdot x$ . Je nach dem Wert der erklärenden Variable  $x$  kann der lineare Prädiktor  $\eta = \eta(x) = \beta_0 + \beta_1 \cdot x$  eine beliebige Zahl sein. Die Erfolgswahrscheinlichkeit liegt aber im Intervall  $[0,1]$  [Kalisch et al. 2021].

Diese Aufgabe lässt sich durch Transformation des linearen Prädiktors lösen, so dass das Ergebnis für jeden Wert von  $x$  immer im gewünschten Intervall  $[0,1]$  liegt.

$$h(\eta) = \frac{e^\eta}{1 + e^\eta}, \eta \in \mathbb{R}.$$

Dieses Verfahren hat seinen eigenen Namen, es ist die so genannte logistische Funktion  $h$ . (Abbildung 15)

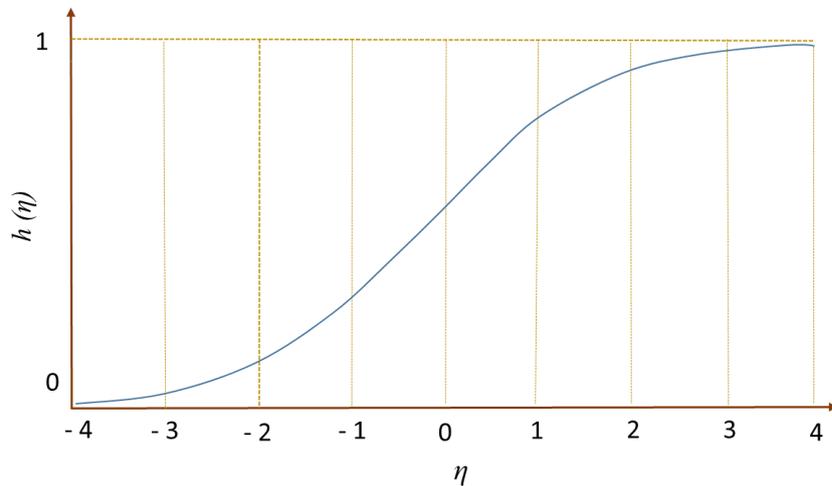


Abbildung 15: Darstellung der logistischen Funktion auf dem Bereich  $-4 \leq \eta \leq 4$  [Kalisch et al. 2021, S. 18]

$$\begin{aligned} \mathbb{P}(Y = 1 \mid X = x) &= p(x) = h(\eta(x)) = h(\beta_0 + \beta_1 \cdot x) \\ &= \frac{\exp(\beta_0 + \beta_1 \cdot x)}{1 + \exp(\beta_0 + \beta_1 \cdot x)} \in [0, 1]. \end{aligned}$$

Dadurch wird sichergestellt, dass die modellierte Wahrscheinlichkeit für jeden Wert von  $x$  immer einen Wert zwischen 0 und 1 erhält.

Die Erfolgswahrscheinlichkeit  $p(x)$  wird über die "Umwege" der Log-Odds (Link-Funktion) mit einer linearen Funktion (linearer Prädiktor) modelliert. Durch die Log-Odds wird die Erfolgswahrscheinlichkeit so transformiert, dass sie ohne technische Probleme direkt mit dem linearen Prädiktor modelliert werden kann [Kalisch et al. 2021]. Im Allgemeinen wird eine Funktion  $g$ , die einen technisch sinnvollen Zusammenhang zwischen Erwartungswert (Erfolgswahrscheinlichkeit  $p$ ) und linearem Prädiktor  $\eta$  ermöglicht, als Linkfunktion bezeichnet

$$g(p) = \log\left(\frac{p}{1-p}\right).$$

Die logistische Regression verwendet als Verknüpfungsfunktion die Log-Odds (oder Logit-Funktion) und entspricht der Umkehrung der logistischen Funktion [Kalisch et al. 2021].

Die Modellanpassung bzw. die Schätzung der Parameter findet in der Regel mittels Maximum-Likelihood-Schätzung statt. Da sich nach dem Nullsetzen der Score-Gleichungen ein mehrdimensionales, nichtlineares Gleichungssystem ergibt, wird zur Lösung der ML-Gleichungen in

der Regel ein iteratives Verfahren wie der Newton-Raphson-Algorithmus oder der Fisher-Score-Algorithmus verwendet.

### 2.5.2.2 Support-Vektor-Maschinen

Die Support Vector Machine (SVM) ist laut [Paluszek et al. 2019] [Panesar 2019] ein überwachtes Lernmodell, das auf einem nicht-probabilistischen binären linearen Klassifikator basiert und sowohl für Klassifizierungs- als auch für Regressionsprobleme verwendet werden kann. Zur Klassifizierung von Daten in zwei Klassen wird eine Hyperebene so dargestellt, dass der Abstand der Punkte aus beiden Klassen von der Hyperebene maximal ist. Die Hyperebene dient als Trennfläche zwischen den beiden Klassen und hat eine geringere Dimension als der Merkmalsraum. Zum Beispiel ist die Hyperebene für einen 2-dimensionalen Merkmalsvektor eine Linie [Rebala et al. 2019] [Bankhofer et al. 2008].

In der Regel kann SVM für linear trennbare Daten verwendet werden. Wie in Abbildung 15 dargestellt, hat die Hyperebene (b) den größten Abstand zwischen den beiden Klassen. Die Punkte, die den Linien a und c am nächsten liegen, stellen die Support-Vektoren dar [Rebala et al. 2019].

Im Fall von nicht linear trennbaren Datenpunkten (linke Seite von Abbildung 16) verwendet die SVM die Kernel-Funktion, um die Datenpunkte in einen höherdimensionalen Raum abzubilden und sie linear trennbar zu machen (rechte Seite von Abbildung 16) [Panesar 2019] [Rebala et al. 2019].

Die SVM verarbeitet numerische Eingaben und funktioniert gut bei kleinen Datensätzen. Die Fähigkeit, das Modell zu verstehen und zu erklären, nimmt jedoch mit zunehmender Dimensionalität ab. Mit zunehmender Größe des Datensatzes kann sich auch die Trainingszeit erhöhen. SVMs erzielen auch schlechtere Ergebnisse bei verrauschten Daten [Panesar 2019].

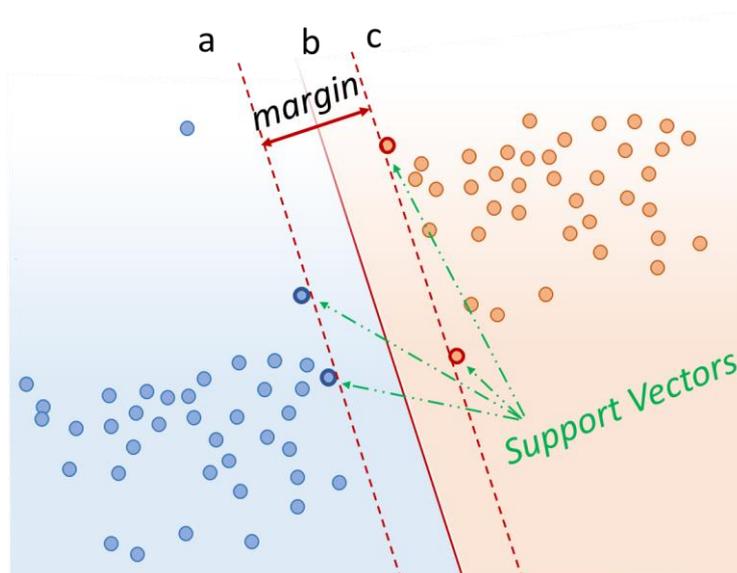


Abbildung 16: Prinzipielle Funktionsweise der SVM in Anlehnung an [Rebala et al. 2019]

### 2.5.2.3 Entscheidungsbaum

Entscheidungsbäume dienen der Einteilung von Objekten in Gruppen auf der Grundlage geeigneter Merkmale in Bezug auf eine festgelegte Zielgröße. Im Wesentlichen lassen sich Entscheidungsbäume in zwei Varianten unterteilen: Klassifikationsbäume und Regressionsbäume. Klassifikationsbäume werden mit nominalskalierten Variablen als abhängige Zielgröße verwendet, während Regressionsbäume eine quantitative Variable als abhängige Zielgröße haben [Bankhofer et al. 2008].

Die Struktur eines Entscheidungsbaums ist baumartig, mit einer Wurzel, mehreren Blättern (Knoten), inneren Knoten und Kanten. Jedem Blatt (Knoten) ist eine Klasse zugeordnet; pro Klasse sind mehrere Blattknoten möglich. Jedem inneren Knoten ist ein Merkmal zugeordnet; pro Merkmal sind mehrere innere Knoten möglich. Um ein Objekt mit einem Entscheidungsbaum zu klassifizieren, wird mit der Wurzel begonnen und das Objekt den inneren Knoten einer neuen Unterteilung zugeordnet, bis das Objekt durch einen Blattknoten eindeutig klassifiziert ist. (Abbildung 17) [Bankhofer et al. 2008].

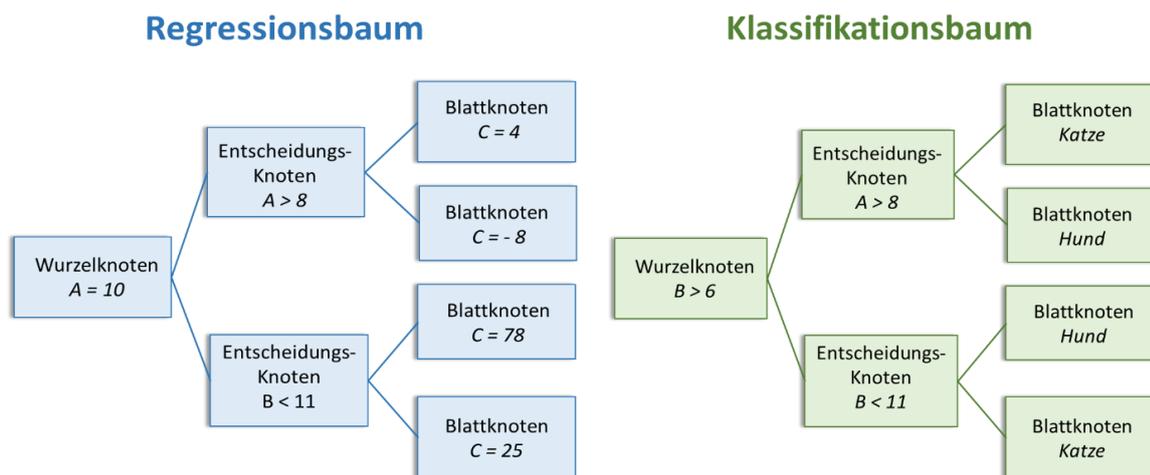


Abbildung 17: Darstellung der Regressions- und Klassifikationsbäume in Anlehnung an [Bankhofer et al. 2008]

Ein Entscheidungsbaum ist grundsätzlich so lange gewachsen, wie neue Unterteilungen gefunden werden können, die die Fähigkeit des Baumes unterstützen, einzelne Untersuchungsobjekte in geeigneter Weise in einzelne Klassen aufzuteilen. Dabei ergibt sich das Problem der sogenannten Überanpassung, d.h. der Überspezialisierung des Entscheidungsbaums. Sie entsteht, wenn die Untersuchungsgegenstände in Klassen aufgeteilt werden, die nur auf den jeweiligen Untersuchungsgegenstand zutreffen und nicht generell [Bankhofer et al. 2008].

Zur Vermeidung des Problems der Überspezialisierung (Overfitting) des Entscheidungsbaums können beispielsweise zwei alternative Ansätze im Entscheidungsbaumprozess eingesetzt werden, nämlich Pruning und Bonsai-Technik. Als Pruning bezeichnet man das Ausdünnen des

Entscheidungsbaums nach der vollständigen Entwicklung durch Entfernen von Blättern und Ästen, die nicht verallgemeinert werden können. Dies geschieht, um die Leistung des Entscheidungsbaums zu verbessern und ein Overfitting zu vermeiden. Diese Methode basiert auf zuvor ausgewählten Testdaten. Sobald die Fehlerrate in den Testdaten steigt, wird der Baum nicht mehr unterteilt, wie in Abbildung 18 dargestellt [Bankhofer et al. 2008].

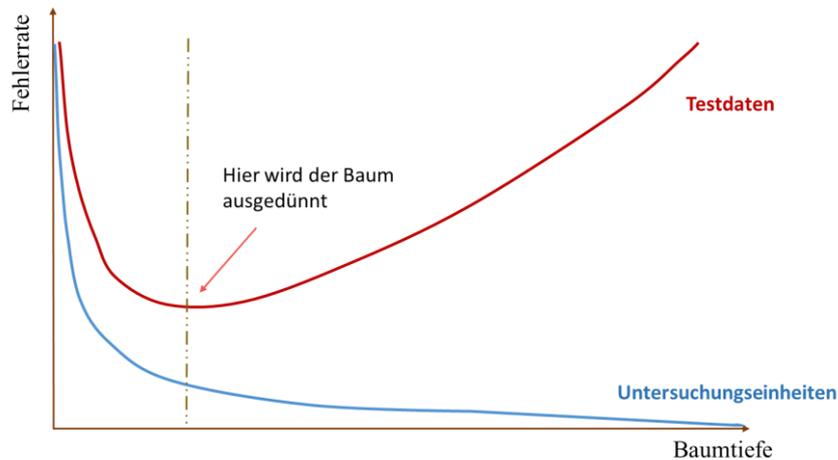


Abbildung 18: Entscheidung über Pruning eines Entscheidungsbaums [Bankhofer et al. 2008, S. 283]

Im Rahmen der Bonsai-Technik hingegen wird das Wachstum des Baumes gebremst, bevor er sich zu weit aufteilt. Durch verschiedene Tests an jeder Wurzel wird versucht zu prüfen, ob eine weitere Teilung ratsam ist. Die Schwierigkeit besteht jedoch darin, dass diese Technik ausschließlich auf Tests an den untersuchten Objekten basiert [Bankhofer et al. 2008].

Algorithmen wie CART [Breiman et al. 1984], CHAID [Kass 1980] oder C4.5 [Quinlan 1993] werden häufig zur Erstellung von Entscheidungsbäumen verwendet. Diese Algorithmen unterscheiden sich im Detail, ihre allgemeine Struktur ist jedoch sehr ähnlich. Sie basieren in der Regel auf einer Divide-and-Conquer-Strategie, bei der ein Datensatz in immer einheitlichere Teildatensätze auf der Basis jeweils eines Merkmals zerlegt wird.

### 2.5.3 Unüberwachtes Lernen

Unüberwachtes maschinelles Lernen (engl. unsupervised learning) ist eine Art des Lernens, bei der Label aus unbekanntem Daten entdeckt werden sollen [Greene et al. 2008] [Michau et al. 2019]. In diesem Fall gibt es nur Trainingsdaten, und das Ziel ist es, Objekte in Clustern zu gruppieren und/oder eine große Menge der gegebenen Daten zu reduzieren. Die Ergebnisse der Clusteranalyse werden häufig mit Hilfe eines Streudiagramms veranschaulicht. (siehe Abbildung 19) [Schacht et al. 2019].

Die Suche nach Labels bezieht sich auf die gesamten Trainingsdaten und ist nicht wie beim überwachten Lernen auf Zusammenhänge mit einer konkreten Zielvariablen beschränkt. Ohne eine konkrete Zielvariable mit bekannten Werten kann es jedoch kein explizites Richtig oder

Falsch einer Lösung geben und damit auch keine externe Rückmeldung über die Qualität der gefundenen Labels während des Lernprozesses [Schacht et al. 2019].

Praktische Anwendungen des unüberwachten maschinellen Lernens sind in der Regel Anomalieerkennung, Dimensionsreduktion und Clustering.

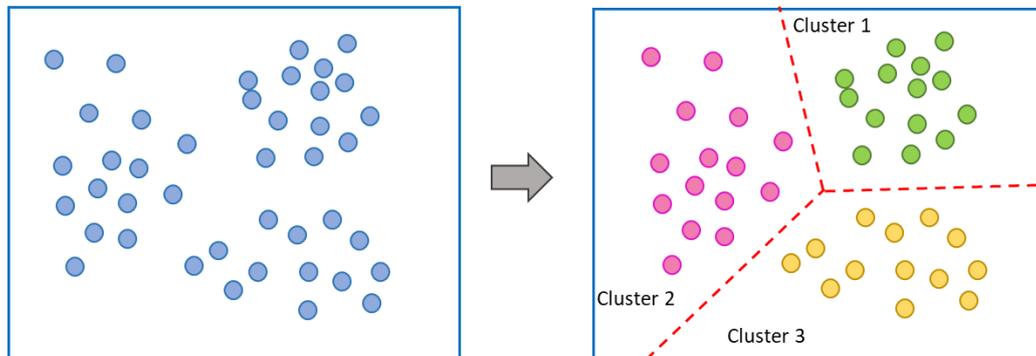


Abbildung 19: Visualisierungen der Ergebnisse von Clusteranalyse in Anlehnung an [Schacht et al. 2019]

Bei der Clusteranalyse handelt es sich um eine explorative Datenanalyse, mit der Informationen in aussagekräftige Untergruppen (Cluster) aufgeteilt werden können, ohne dass genau bekannt ist, zu welcher Gruppe die Informationen gehören. Bei der Analyse wird jedes Cluster als eine Gruppe von Objekten definiert, die bestimmte Eigenschaften gemeinsam haben, sich aber von Objekten in anderen Gruppen deutlich unterscheiden. Aus diesem Grund wird das Clustering auch als "unüberwachte Klassifizierung" bezeichnet. Die Clusteranalyse lässt sich hervorragend zur Strukturierung von Informationen und zur Ableitung sinnvoller Beziehungen zwischen Daten einsetzen. So können beispielsweise Marketingfachleute Kunden auf der Grundlage ihrer Interessen in Gruppen einteilen.

Die Clusteranalyse gehört zu den am weitesten verbreiteten prädiktiven Analysemethoden und findet in einer ganzen Reihe von Bereichen Anwendung, z. B. bei der Analyse von Kriminalitätsmustern, in der medizinischen Forschung, im Bildungswesen, in der Archäologie, in der Astronomie oder in der Industrie.

Mit Hilfe der Anomalie- oder Ausreißerererkennung wird in einem Datensatz nach Objekten gesucht, die von den meisten anderen Objekten abweichen. Die Ausreißer- und Anomaliebegriffe werden üblicherweise als Synonyme verwendet. Die Ausreißer werden in einigen Fällen als Oberbegriff betrachtet, der neben Fehlern in den Daten und im Rauschen auch die Anomalien enthält, die im Rahmen einer Anwendung im engeren Sinne besonders relevant sind [Aggarwal 2016].

Die Anomalieerkennung und die Clusteranalyse sind komplementäre Aufgaben. Bei der Clusteranalyse werden ähnliche Punkte in einem Cluster gruppiert. Die Ausreißer zeichnen sich

häufig durch eine Abweichung (Unähnlichkeit) von den erkannten Clustern aus, die durch die roten Umrandungen angezeigt ist (Abbildung 20) [Schacht et al. 2019].

Der erste Anwendungsbereich liegt in der Vorbereitungsphase für die Erstellung eines Modells. Hier wird diese Analyse verwendet, um Fehler in den Daten zu erkennen, die korrigiert oder ignoriert werden können, oder um ungewöhnliche Werte zu identifizieren, die tatsächlich gültig sind und in dem zu erstellenden Modell berücksichtigt werden müssen. Solche Ausreißer können die Art des in der Analyse verwendeten Algorithmus beeinflussen, da einige Algorithmen empfindlicher auf Ausreißer reagieren als andere [Weber 2020].

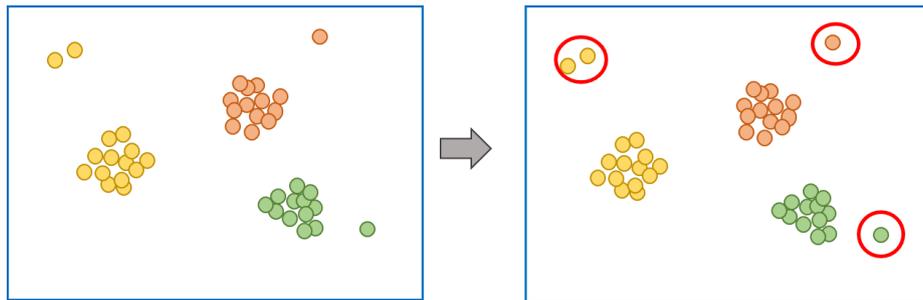


Abbildung 20: Darstellung von eine Clusternanalyse und Anomalieerkennung in Anlehnung an [Schacht et al. 2019]

Als zweite Einsatzmöglichkeit ist die Erkennung von Ausreißern oder Anomalien zu sehen, die z. B. bei der Betrugsanalyse auf ungewöhnliche Aktivitäten hinweisen. Diese Aufgabe ist in vielen Bereichen anwendbar: bei der Produktionsprozesskontrolle, der Fehlererkennung, der Erkennung von Störungen, und der Betrugserkennung. Eines der größten Potentiale ergibt sich sicherlich aus der Anwendung der Algorithmen in der Betrugserkennung auf möglichst große Datensätze (z.B. Transaktionsdaten im Finanzsektor). Hierbei sind die Datenmengen extrem groß und können sogar in Echtzeit ausgewertet werden [Weber 2020].

Bei der Dimensionalitätsreduktion handelt es sich um die Umwandlung eines hochdimensionalen Datensatzes in einen niedrigdimensionalen Raum, ohne dass relevante oder wertvolle Informationen für die weitere Datenauswertung dabei verloren gehen. Die Dimensionsreduktion ist keine Aufgabe des maschinellen Lernens im klassischen Sinne, sondern vielmehr ein Schritt der Datenvorverarbeitung. Die Dimensionsreduktion kommt insbesondere als Vorstufe bei der explorativen Datenanalyse zum Einsatz, wobei oft eine Reduktion auf höchstens drei Dimensionen (Merkmale) angestrebt wird. Dies geschieht auf unterschiedliche Weise, z. B. wird mit möglichst wenigen Attributen versucht, die Gesamtvarianz zu erhalten, oder es werden schwach korrelierte Attribute entfernt. Einer der bekanntesten Vertreter ist die Hauptkomponentenanalyse (PCA) [Schacht et al. 2019].

Beim Einsatz der Dimensionalitätsreduktion zur Datenvorverarbeitung muss darauf geachtet werden, dass das entsprechende Modell der Dimensionalitätsreduktion ausschließlich auf den Trainingsdaten erstellt wird. Da es anschließend auch für die Testdaten und andere neue Daten

verwendet werden muss, damit auch diese Daten für die folgende Aufgabe entsprechend aufbereitet sind, wird die Dimensionalitätsreduktion häufig wie ein Vorhersagemodell eingesetzt. [Schacht et al. 2019].

Als Lernalgorithmen können z. B. Bayes-Netze, Hidden Markov Modelle oder k-Means-Modelle verwendet werden.

Ein besonderer Vorteil der Hauptkomponentenanalyse besteht darin, dass die Variablen und die Objekte in einer gemeinsamen Grafik (einem Biplot) dargestellt werden können. Ein weiterer Vorteil ist, dass die PCA sehr gut nachvollziehbar und damit interpretierbar ist. Darüber hinaus kann die PCA auch als Ausgangspunkt für weitere Verfahren dienen, z.B. zur Dimensionsreduktion für Clusteranalysen, oder zur Durchführung einer Hauptkomponentenregression. Die Hauptnachteile sind, dass die PCA auf den euklidischen Abstand angewiesen ist und dass es häufig nicht gelingt, einen ausreichenden Anteil der Information in eine Ebene zu bringen.

## 2.5.4 Verstärkers Lernen

Verstärkendes maschinelles Lernen (engl. reinforcement learning) gehört zu den speziellen Arten des maschinellen Lernens, bei denen die Trainingsdaten bereits zu Beginn des Lernprozesses zur Verfügung stehen, entweder mit oder ohne Kennzeichnung als Beispiele oder Beobachtungen. Beim Verstärkungslernen steht ein Modell im Vordergrund, das durch Interaktion mit seiner Umgebung Daten erzeugt. Aus diesem Grund heißt das Verstärkungslernen auch Lernen durch Interaktion [Schacht et al. 2019].

Der verwendete Lernalgorithmus lernt nach dem Prinzip von Lob und Tadel. Dabei wird dem Algorithmus nicht gezeigt, welche Aktion oder Operation in der jeweiligen Situation die richtige ist. Das Feedbacksignal drückt entweder eine Belohnung oder eine Bestrafung (negative Belohnung) im Hinblick auf die Erreichung eines Ziels aus, das der verwendete Lernalgorithmus zu maximieren versucht. Hauptziel ist es, durch dieses Trial-and-Error-Verfahren die Strategie zu erlernen, mit der die Gesamtbelohnung für eine abgeschlossene Handlungsabfolge maximiert wird. Beispielsweise führt bei einem Spiel wie Schach oder Go das Gewinnen oder Verlieren zu einem Feedbacksignal für eine Folge von Zügen während des Spiels [Schacht et al. 2019].

Durch die Interaktion wächst der Erfahrungsschatz des Lernalgorithmus, der oft nur einen Teil der möglichen Handlungen kennt, die in der Vergangenheit zu einer größeren oder kleineren Belohnung geführt haben. Zur Bewältigung dieses Problems, typischerweise im Sinne von Versuch und Fehler, müssen auch bisher unbekannte (nicht bewertete) Handlungen vom Lernalgorithmus ausprobiert werden, um den Handlungsspielraum im Rahmen seiner Strategie zu vergrößern [Schacht et al. 2019].

Der zentrale Unterschied zwischen überwachtem und unüberwachtem Lernen besteht darin, dass das Verstärkungslernen keine Beispieldaten im Voraus benötigt. Durch unzählige Iterationen entwickelt der Algorithmus seine eigene Strategie in einer Simulationsumgebung.

## 2.5.5 Künstliche neuronale Netze

Als neuronale Netze werden mathematische Strukturen bezeichnet, in denen das Verhalten des technischen Systems in Abhängigkeit von verschiedenen Eingangsparametern in Form von Neuronen, in deren Anordnung und in der Stärke der Verbindungen zwischen den Neuronen gespeichert ist (Abbildung 21) [Schenk 2010]. Ein entscheidender Nutzen dieser Methode besteht darin, dass auf eine komplizierte analytische Beschreibung der Zusammenhänge zwischen Eingangsdaten und dem Ausgangswert weitgehend verzichtet werden kann. Außerdem sind neuronale Netze auch durch folgende Eigenschaften gekennzeichnet [Seraphin 1994].

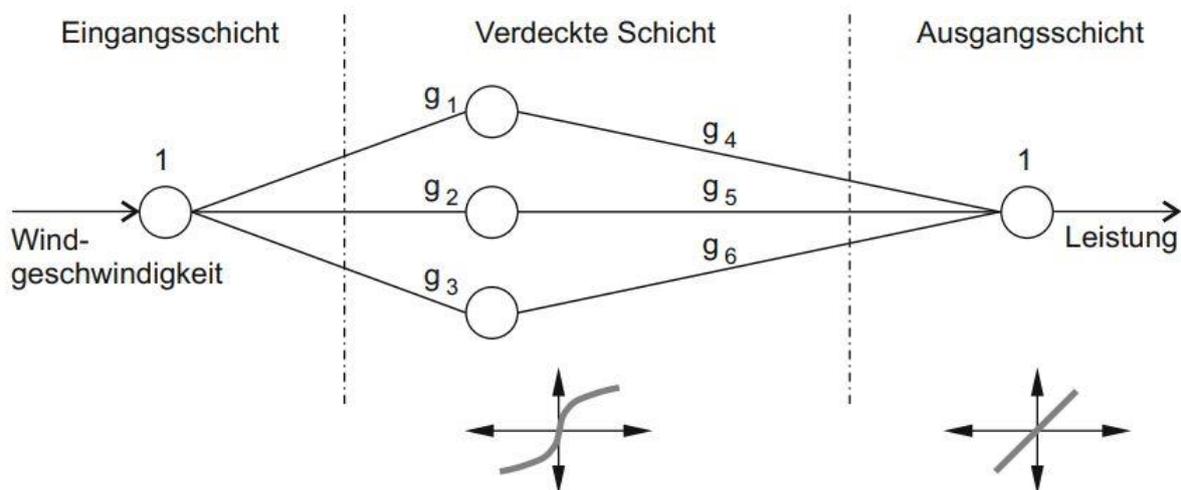


Abbildung 21: Schematische Darstellung eines künstlichen Neuronales Netzes mit einem Eingangs- und Ausgangsneuron [Schenk 2010, S.136]

- Assoziationsfähigkeit im Sinne von Erkennen und Zuordnen ähnlicher Eingangsdaten zu einem Ausgangsergebnis.
- Fehlertoleranz, so dass trotz teilweise fehlerhafter oder unvollständiger Eingabedaten die gewünschten Ergebnisse erzeugt werden.
- Lernfähigkeit, bei der neue Informationen, z.B. neue Eingabedaten und deren Auswirkungen, durch Veränderung der Stärken von Neuronenverbindungen gespeichert werden.
- Ausfallsicherheit, die es ermöglicht, trotz des Ausfalls von Teilbereichen eines neuronalen Netzes, dessen Fähigkeiten weitgehend zu erhalten.
- Die Parallelverarbeitung befähigt zu einer hohen Leistungsfähigkeit der Strukturen eines neuronalen Netzes mittels paralleler Informationsverarbeitung.

Künstliche neuronale Netze werden als eine Form des maschinellen Lernens sowohl im Bereich des überwachten als auch des unüberwachten Lernens eingesetzt.

## 2.3.6 Unteranpassung und Überanpassung

Zu den größten allgemeinen Herausforderungen beim maschinellen Lernen gehört das Problem der Überanpassung (eng. Overfitting) und der Unteranpassung.

Beim Lernen aus Daten ist die Überanpassung sicherlich das größte Risiko für einen erfolgreichen operativen Einsatz. Es kann vorkommen, dass zu viele Merkmale oder Variablen (Features) in die Analyse mit einfließen und das Modell sich so zu sehr an den Trainingsdatensatz anpasst. Dieses Phänomen wird "Überanpassung" genannt. Eine Überanpassung liegt vor, wenn ein Modell zufällige Merkmale der Daten (Rauschen) und nicht die zugrundeliegenden Korrelationen (das Signal) beschreibt. Übermäßige Modellkomplexität ist die Hauptursache [Schacht et al. 2019].

Um eine Überanpassung festzustellen, muss die Modellqualität anhand von Daten bewertet werden, die nicht zum Lernen verwendet wurden. Dazu müssen die verfügbaren Daten in einen Trainings- und einen Test- oder Validierungssatz aufgeteilt werden. Auf diese Weise lässt sich die Leistung in überwachten Lernszenarien leicht bewerten, da die Werte der Zielvariablen (Label) sowohl für die Test- oder Validierungsdaten als auch für die Trainingsdaten bekannt sind [Schacht et al. 2019].

Falls die Leistung bei den unbekanntenen Daten schlechter ist als bei den Trainingsdaten, besteht eine Überanpassung an die Trainingsdaten wie in Abbildung 22 dargestellt. Beim Training mit einer Trainingsmenge muss der Algorithmus nicht einfach alles "auswendig" lernen, sondern er muss die Logik dahinter begreifen [Schacht et al. 2019]. Überanpassung zu erkennen und idealerweise zu vermeiden, ist eine zentrale Herausforderung beim überwachten maschinellen Lernen. Sind die Modelle, die ein gewähltes Lernverfahren erzeugen kann, nicht komplex genug, dann besteht die Gefahr, dass keine ausreichende Leistung erreicht werden kann. In diesem Fall spricht man von Unteranpassung (eng. underfitting) [Schacht et al. 2019]. Wenn die Trainingsmengen im Vergleich zur Komplexität des Modells zu klein sind, kommt es schnell zu einer Überanpassung.

Im Falle einer zu kleinen Trainingsmenge im Verhältnis zur Modellkomplexität kommt es ebenfalls schnell zu einer Überanpassung. Abbildung 23 zeigt die Qualität eines Modells, das einen sehr kleinen Trainingsatz praktisch auswendig lernt. Die Fähigkeit eines Modells, bei gleichbleibender Komplexität die Trainingsdaten auswendig zu lernen, nimmt mit zunehmender Größe der Trainingsmenge immer mehr ab. Dies lässt sich an der Verringerung der Güte der Trainingsdaten ablesen. Gleichzeitig ist das Modell gezwungen, die tatsächlichen Zusammenhänge in den Daten zu lernen, so dass es immer besser verallgemeinern kann. Dadurch nimmt die Güte der Testdaten zu. Unter der Voraussetzung, dass die Datenqualität mindestens konstant bleibt, sind mehr Daten immer besser und können ein Überanpassen verhindern [Schacht et al. 2019].

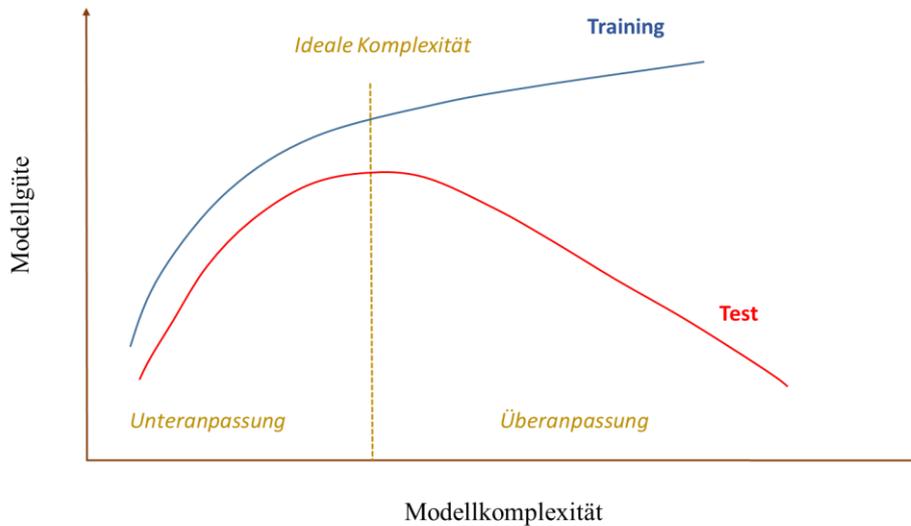


Abbildung 22: Unteranpassung und Überanpassung in Abhängigkeit der Modellkomplexität [Schacht et al. 2019, S.120]

Durch eine Vergrößerung der Trainingsdaten und eine Vielzahl von Iterationen lassen sich die Vorhersagefehler verringern. Diese Datenmengen müssen umso größer sein, je komplexer das zu erkennende Label ist. Dazu das Expertenwissen und die Erfahrung eines Ingenieurs für maschinelles Lernen spielen bei der Auswahl des richtigen Algorithmus für maschinelles Lernen ebenfalls eine wichtige Rolle. Das liegt daran, dass nicht jeder Algorithmus für alle Problemstellungen gleichermaßen geeignet ist. Er muss für das spezifische Ziel ausgewählt und mit geeigneten Trainingsparametern angewendet werden, um optimale Ergebnisse zu gewährleisten [Kudelina et al. 2021].

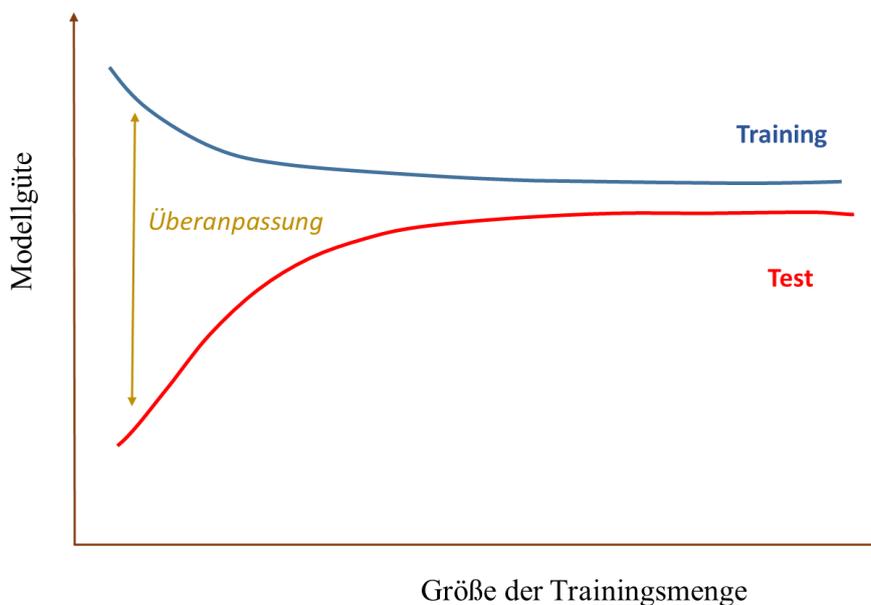


Abbildung 23: Überanpassung bei einer zu kleinen Trainingsmenge [Schacht et al. 2019, S. 120]

## **3 Anwendungsbeispiel der prädiktiven Instandhaltung**

Nicht vorhersehbare Betriebsstörungen führen oft zu schweren wirtschaftlichen Verlusten und können manchmal katastrophale Folgen haben. Ein typisches Beispiel aus dem Bereich der Industrie sind unerwartete Anlagenausfälle, die zu Produktionsausfällen, Lieferverzögerungen, schlechter Kundenzufriedenheit und Sicherheitsproblemen führen. Zustandsüberwachung (Condition Monitoring) ist eines der am häufigsten angewandten prädiktiven Instandhaltungsmaßnahmen. Im Rahmen der zustandsorientierten Instandhaltung spielen eine zuverlässige Zustandsüberwachung und -diagnose eine wichtige Rolle. Dabei werden Sensordaten verwendet, um Fehler in einem frühen Betriebszustand zu diagnostizieren, den Zustand des Bauteils genau zu beurteilen und Rückschlüsse auf die Restlebensdauer zu ziehen [Mobley 2002].

Im Rahmen dieser Arbeit wurde ein Beispiel für die Anwendung der Systemzustandserkennung vorgestellt, die mittels der Klassifikationsalgorithmen durchgeführt wurde. Zur Abschätzung wurden statistische Methoden, prädiktive Instandhaltungsmethoden und Machine Learning basierte Methoden eingesetzt. In diesem Zusammenhang wurden die Klassifikationsalgorithmen, wie logistische Regression, Künstliches neuronales Netz, Support Vector Machine (SVM), Entscheidungsbaumanalyse, K-Means Algorithmus und Hierarchische Clusteranalyse Algorithmus evaluiert und hinsichtlich ihrer Genauigkeit der Systemzustandserkennung miteinander verglichen.

### **3.1 Datenquelle und Programmiersprache**

#### **3.1.1 Datensatz**

Für die Analyse wurden die Triebwerksdegradationssimulation-Datensatz von NASA verwendet [Saxena et al. 2008]. Die wichtigsten Gründe, die für NASA- Datensatz sprechen, sind:

- die freie Verfügbarkeit der Daten
- der NASA-Datensatz [Saxena et al. 2008] ist einer der am weitesten verbreiteten Datensätze für die Prognostik und bietet die Möglichkeit, die Ergebnisse mit bestehenden Datenanalysen zu vergleichen.

Der simulierte mit C-MAPSS (Commercial Modular Aero-Propulsion System Simulation) Datensatz besteht, wie in der folgenden Tabelle 2 dargestellt, aus 4 Unterdatensätzen mit unterschiedlichen Betriebs- und Fehlerbedingungen.

Dieser Datensatz simuliert die Degradation von 100 Turbinenanlagen in vier Arbeitsumgebungen, um die entsprechenden Unterdatensätze FD001, FD002, FD003 und FD004 zu erhalten. Jeder Unterdatensatz besteht aus einem Trainingsatz, einem Testsatz und entsprechenden Restnutzungsdauerwerten. Der Trainingsdatensatz enthält vollständige Run-to-Failure-Daten, mit denen Prognosealgorithmen entwickelt werden können, während der Testdatensatz keine Daten

bis zum Totalausfall enthält, sondern die tatsächliche Restlebensdauer als Information zur Verfügung gestellt wird.

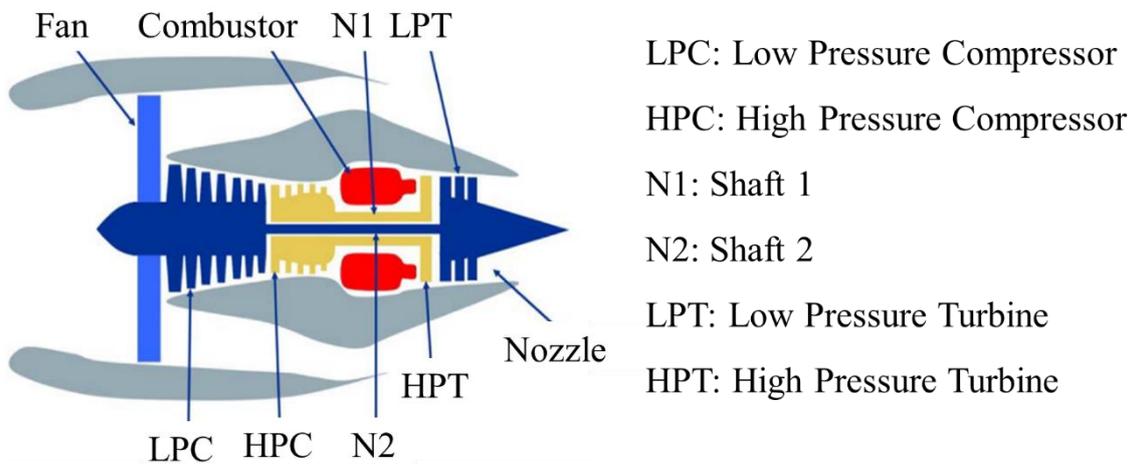


Abbildung 24: Vereinfachtes Darstellung des simulierten Turbinentriebwerks in Anlehnung an [Saxena et al. 2008]

Tabelle 2: Details zum Datensatz in Anlehnung an [Saxena et al. 2008]

Data Set Name	Training Data set	Testing Data set	Operating Regions	Fault Modes
FD001	100	100	1	HPC Degradation
FD002	260	259	6	HPC Degradation
FD003	100	100	1	Fan Degradation & HPC Degradation
FD004	248	249	6	Fan Degradation & HPC Degradation

In jedem Unterdatensatz sind Betriebseinstellungen und Sensormessungen (siehe Tabelle 3) für mehrere Triebwerke und für jeden Zyklus ihrer Lebensdauer vorhanden. Jede Zeitreihengruppe enthält 26 Spalten: Motornummer, Zeitschritt, drei betriebsbereite Sensoreinstellungen und 21 Sensormessungen, die durch Leerzeichen getrennt sind. Jede Zeile ist eine Momentaufnahme von Daten, die während eines einzelnen Betriebszyklus erfasst wurden. Jede Spalte entspricht einer Variablen [Saxena et al. 2008].

Die Spalten haben folgende Bedeutung:

- Spalte 1 - Gerätenummer
- Spalte 2 - Laufzeit in Zyklen
- Spalten 3-5 - Betriebseinstellungen
- Spalten 6-26 - Sensormessungen 1-21

Die Datensätze bestehen aus mehreren Zeitreihengruppen, wobei jede Zeitreihengruppe von einem Triebwerk stammt. Hierbei handelt es sich um Triebwerke desselben Typs [Saxena et al. 2008].

Tabelle 3: Messungen am Triebwerk in Anlehnung an [Saxena et al. 2008]

Symbol	Description	Units
T2	Total temperature at fan inlet	°R
T24	Total temperature at LPC outlet	°R
T30	Total temperature at HPC outlet	°R
T50	Total temperature at LPT outlet	°R
P2	Pressure at fan inlet	psia
P15	Total pressure in bypass-duct	psia
P30	Total pressure at HPC outlet	psia
Nf	Physical fan speed	rpm
Nc	Physical core speed	rpm
epr	Engine pressure ratio (P50/P2)	-
Ps30	Static pressure at HPC outlet	psia
phi	Ratio of fuel flow to Ps30	pps/psi
NRf	Corrected fan speed	rpm
NRc	Corrected core speed	rpm
BPR	Bypass Ratio	-
farB	Burner fuel-air ratio	-
htBled	Bleed Enthalpy	-
Nf_dmd	Demanded fan speed	rpm
PCNfR_dmd	Demanded corrected fan speed	rpm
W31	HPT coolant bleed	lbm/s
W32	LPT coolant bleed	lbm/s

### 3.1.2 Programmiersprache

Die Programmiersprache R Version 3.6.0 wurde zur Erkennung des Systemzustands eingesetzt, da die Statistik-Software R eine objekt-orientierte, interaktive Programmiersprache ist, mit der einfach statistische Auswertungen vorgenommen, vielfältige Grafiken erstellt und Simulationen durchgeführt werden können.

Die Software R dient nicht nur als Programmiersprache, sondern auch als interaktive Umgebung für die Beschäftigung mit Data Science. Zur Unterstützung der Interaktion ist R eine viel flexiblere Sprache als viele andere Sprachen. Ein großer Vorteil von R ist die Möglichkeit, auf einfache Weise eigene Grammatiken für bestimmte Teile des Data-Science-Prozesses zu entwickeln [Wickham et al. 2018]. R verfügt über die breiteste Auswahl an Algorithmen, was es sowohl auf der erklärenden als auch auf der prädiktiven Ebene der Datenanalyse stark macht. Zusätzlich zu den Auswertungen und Statistiken bietet R auch Bibliotheken für maschinelles Lernen. So werden z. B. Vorhersagemodelle generiert (Predictive Analytics) oder Data Mining mittels unüberwachten maschinellen Lernens betrieben, d. h. die Entdeckung neuer Korrelationen durch die automatisierte Exploration von Daten.

Gerade vor dem Hintergrund der Anforderungen, die Predictive Maintenance an eine statistische Lösung stellt, wie z.B. der Funktionsumfang, die Integrationsmöglichkeiten oder die Möglichkeiten zur grafischen Darstellung der Ergebnisse, ist R eine der besten Alternativen. Zusätzlich zu R wurde in einem Teil der Analyse die Pandas-Bibliothek von Python zur Normalisierung der Daten verwendet, da sie umfangreiche Funktionen zur Vorverarbeitung der Datensätze bietet.

## 3.2 Analyse

Das Skript zu der Analyse wurde von Jun.-Prof. Dr. Antoine Tordeux, Leiter des Lehrstuhls für Verkehrssicherheit und Zuverlässigkeit an der Universität Wuppertal, verfasst und zur Verfügung gestellt.

100 Turbinenanlagen laufen beim Start im normalen Zustand an und beginnen sich irgendwann während der Zeitreihe zu verschlechtern (Degradation). Die Degradation in den Trainingssätzen nimmt zu bis hin zu einem Ausfall. Die Degradation endet in den Testdatenätzen vor dem Ausfall. Das Ziel besteht darin, mithilfe der verfügbaren Sensormessdaten automatisch den Zustand des Systems genau erkennen und klassifizieren.

Bei der Klassifizierung wurden die Daten manuell den Klassen *betriebsbereit* ( $Y=0$ ) und *ausgefallen* ( $Y=1$ ) zugeordnet. Hierzu wird anhand von Trainingsdaten ermittelt, welche Objekte zu welchen Klassen gehören.

### 3.2.1 Festlegung der Labels

Zur Analyse des Systemzustands mit den überwachten Algorithmen wird ein Referenzwert (Label) benötigt. Da bereits bekannt ist, dass sich der Systemzustand mit der Zeit verschlechtert und in einen fehlerhaften Zustand übergeht, wurde der Durchschnittswert der ersten zehn gemessenen Systemzustände, die dem funktionsfähigen Betriebszustand der Turbinenanlage entsprechen, als Referenzwert genommen.

### 3.2.2 Hauptkomponentenanalyse

Die Hauptkomponentenanalyse gehört zu den unüberwachten Methoden und wird verwendet, um die wahre Dimension der Daten zu ermitteln, redundante Daten zu identifizieren und die Daten in einer reduzierten Dimension darzustellen.

Bei großen Datenmengen kann es eine Menge redundanter Daten geben. Daher ist eine gute Beschreibung der Daten oft nur mit kleineren Dimensionen der Daten möglich.

Mit Hilfe der Hauptkomponentenanalyse als Datenreduktionsmethode wurde die wahre Dimension der Daten deutlich gemacht. Dazu wurde ein kleinerer Dimensionsraum geschaffen, der durch neue künstliche Variablen aufgespannt wurde. Diese neuen Variablen stellen Linearkombinationen der ursprünglichen Variablen dar und erklären so viel Varianz wie möglich.

Die Hauptkomponenten wurden aufgelistet und durchnummeriert (Principal Components PC1 bis PCn), wobei der Anteil der erklärten Varianz von Hauptkomponente zu Hauptkomponente immer weiter abnimmt. Basierend auf der durchgeführten Hauptkomponentenanalyse wurden für jeden Unterdatensatz die Hauptkomponenten abgeleitet und es hat sich ergeben, dass nur die erste Komponente die Information über den Ausfall der Turbinenanlagen enthält (siehe Abbildung 25).

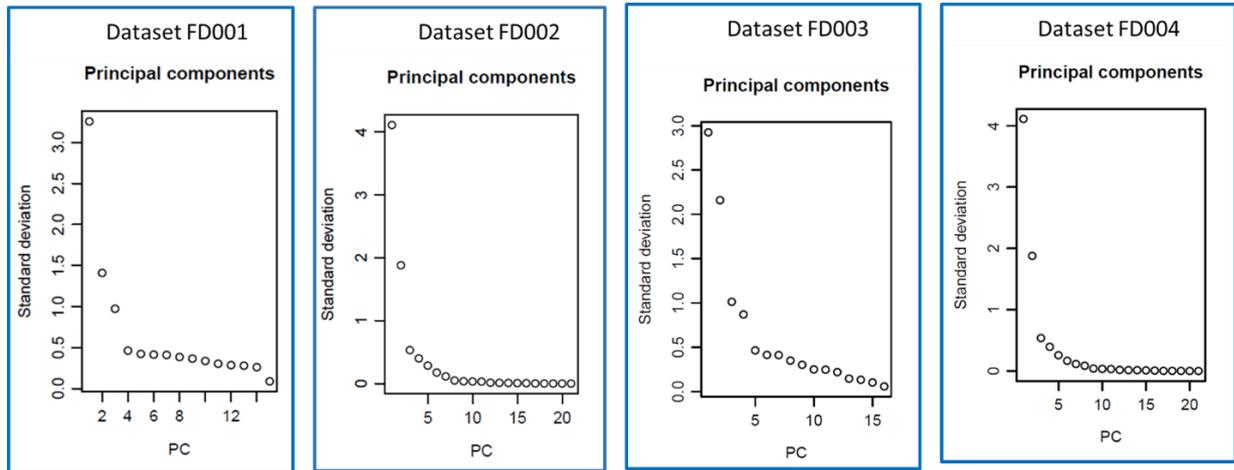


Abbildung 25: Die Anzahl der Hauptkomponenten für 4 Unterdatensätze

### 3.2.3 Eingesetzte Klassifikationsalgorithmen

Methoden- und Algorithmenauswahl wird unter Berücksichtigung der zu stellenden Problemstellung durchgeführt. Für die Erkennung des Systemzustands wurden die am häufigsten in der Praxis verwendeten Klassifizierungsmethoden und Algorithmen verwendet (siehe Tabelle 4), die auch in Kapitel 2 erläutert wurden.

Tabelle 4: Übersicht auf die verwendete ML-Algorithmen

Methode	Aufgabe	Algorithmen	Bezeichnung	Modell
Überwachtes Lernen	Klassifikation	Logistische Regression	REG	Trennlinie
		Iterative Dichotomiser	DT	Entscheidungsbaum
		Stützvektormaschine	SVM	Hyperebene
		Bayessche Inferenz	BN	Bayessche Modell
Unüberwachtes Lernen	Clustering	K-Means	KM	Clustermittelpunkte
		Hierarchische Clusteranalyse	H	
Verschiedene	Verschiedene	Rückwärtspropagierung	NN	Künstliche Neuronale Netze

Klassifikationsanalyse wurden folgende R - Paketen verwendet:

- **rpart** wurde für die Realisierung der Entscheidungsbaumanalyse mittels Funktion `rpart` eingesetzt.

Die Methode ist als Splitfunktion für die R-Funktion `rpart` umgesetzt. `Rpart` umfasst eine rekursive Partitionierung für Klassifikation und Regression [Breiman et al. 1984]. `Rpart` ermöglicht es, eigene Splitfunktionen einzufügen, und zwar in Form von Unterfunktionen. Drei Unterfunktionen sind dafür notwendig: eine Initialisierungsfunktion, eine Auswertungsfunktion und eine Splitfunktion [Therneau et al. 2015]. Die Initialisierungsfunktion nimmt die Parameter entgegen und führt Prüfungen durch. Die Auswertungsfunktion wird einmal pro Knoten ausgeführt und berechnet für jeden Knoten eine Abweichung. Außerdem gibt sie dem Knoten ein Label, d. h. den Namen der Klasse, die die Beobachtungen in diesem Knoten als Vorhersage erhalten würden. Die Splitfunktion hingegen lässt man für jede Kovariante einmal laufen und berechnet den besten Split für diese Kovariante. Die Abweichung der besten Aufteilung, d. h. in diesem Fall die Kosten der besten Aufteilung, wird als "Güte" aufgezeichnet und wird verwendet, um zu entscheiden, mit welcher Kovariante aufgeteilt werden soll. Dazu wurde auch die Funktion `round` eingesetzt, die eine numerische Eingabe auf eine bestimmte Anzahl von Nachkommastellen rundet.

- **e1071** wurde für die Realisierung des Algorithmus SVM mittels Funktion `svm` eingesetzt.

Wir haben die Funktion `svm` verwendet, um einen Klassifikator von Referenzvektoren für einen bestimmten Wert des `cost`-Parameters anzupassen.

- **Neuronenet** – wurde für die Realisierung des Algorithmus Neuronale Netze verwendet.

Mit dem Argument `linear.output` wird angegeben, ob eine Regression (`linear.output=TRUE`) oder eine Klassifizierung (`linear.output=FALSE`) durchgeführt werden soll.

In einem weiteren Schritt wurden 20 Messpunkte mit dem Zeitintervall  $t_2$  definiert, an denen die ausgewählten Algorithmen den Systemzustand ermitteln sollten. Der von den Algorithmen zu jedem Messpunkt ermittelte Systemzustand wird dann mit der zuvor definiertem Referenzwert verglichen und der Systemzustand in funktionierende oder ausgefallene Zustände klassifiziert. Anschließend wurde für alle Algorithmen eine Fehlerberechnung durchgeführt und Leistung der Algorithmen bewertet.

### 3.2.4 RMSE

Root Mean Square Error (RMSE) ist eine Kennzahl, die angibt, wie weit vorhergesagte Werte im Durchschnitt von beobachteten Werten abweichen. Ein Modell umso schlechter, je größer die Differenz zwischen den vorhergesagten und den beobachteten Werten ist. Umgekehrt gilt: Je geringer der RMSE, desto genauer ist die Anpassung eines Modells an die zu analysierenden Daten. Es gilt demnach, durch die Steuerung von Einflussfaktoren den kleinstmöglichen RMSE zu erhalten, um so die Güte eines Modells zu verbessern.

Ein Vergleich des RMSE von verschiedenen Algorithmen kann sich als besonders nützlich erweisen, um zu ermitteln, bei welchem Algorithmen die Daten tatsächlich besser passen. Für das Anwendungsbeispiel wird ein Vorhersagemodell erstellt, das die Ausfallwahrscheinlichkeit für einen Turbinenanlagenausfall auf Basis der Sensormessdaten ermittelt. Das Modell lernt, wie die Vorhersagevariablen (Sensormessdaten) zu kombinieren sind, um bestmöglich eine Wahrscheinlichkeit für die Zielvariable (Turbinenanlagenausfall) vorherzusagen. Da es sich hierbei um die Prognose von Wahrscheinlichkeiten handelt, sind diese auf den Wertebereich  $[0,1]$  beschränkt. So muss ein statistisches Modell Anwendung finden, das in der Lage ist, den Zusammenhang zwischen Sensormessdaten und Fehlerauftreten zu modellieren.

### 3.3 Ergebnisse

Als erstes Ergebnis der Datenanalyse können wir in Abbildung 26 sehen, dass die Daten in 2 Klassen eingeteilt wurden, nämlich: Blau - funktionsfähig und Rot - ausgefallen. Das bedeutet, dass unser Beobachtungssystem in zwei verschiedene Zustände hatte: zu Beginn der Beobachtung war das System funktionsfähig und am Ende der Beobachtung war das System nicht betriebsbereit. Bei den Datensätzen FD002 und FD004 hat der Clusteralgorithmus den Unterschied zwischen den beiden Datenklassen nicht so deutlich definiert.

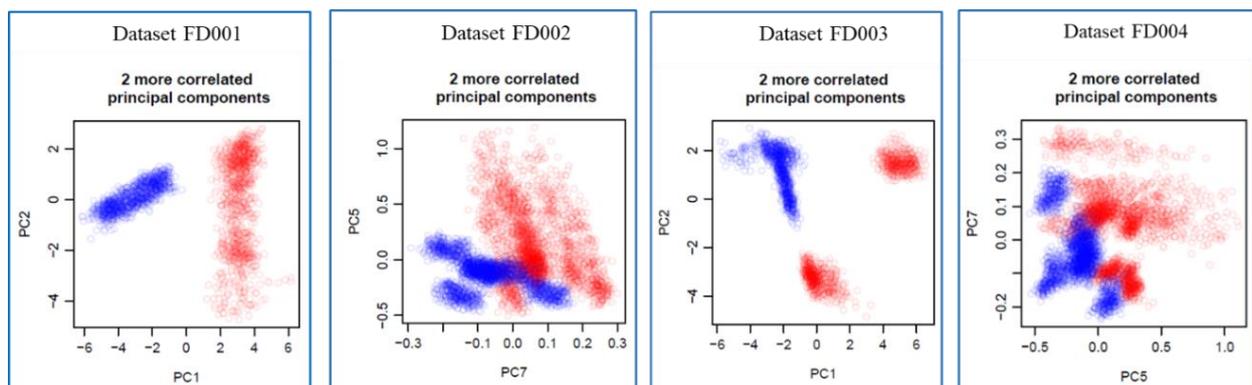


Abbildung 26: Datenanalyse während der Beobachtung

Die Abbildung 27 zeigt detailliert den Prozess der Identifizierung des Turbinenanlagenzustands während der Beobachtung. Am Anfang des Turbinenlaufs kann der Algorithmus nicht zwischen Betriebs- und Nichtbetriebszustand unterscheiden. Dies liegt daran, dass am Anfang der Beobachtung alle Turbinenanlagen noch funktionsfähig sind. Wie aus dem Diagramm ersichtlich ist, beginnt der Algorithmus jedoch ab etwa 160 Zyklen, den Arbeitszustand (rote Punkte) vom Ausfallzustand (blaue Punkte) zu unterscheiden. Am Ende des Turbinenlaufs stellt der Algorithmus dann einen Ausfall der Turbine fest, was bedeutet, dass die Turbine im Zyklus 368 außer Betrieb ist.

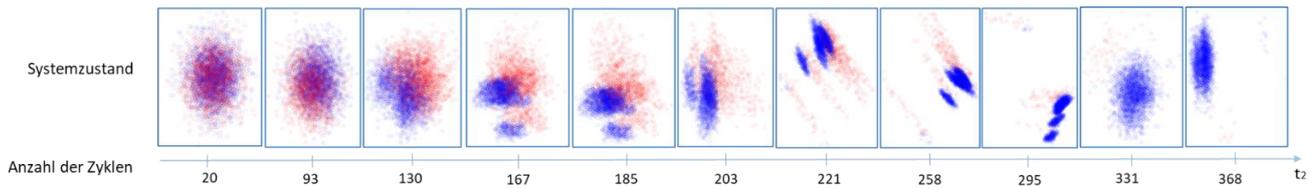


Abbildung 27: Prozess der Erkennung des Turbinenanlagenzustands

Die folgenden Abbildungen 28-30 zeigen die Erkennungsraten des Systemzustands für die fünf überwachten Algorithmen. Für jeden Datensatz werden drei Arten von Kurvendiagrammen erstellt, die die Genauigkeit der Zustandserkennung von einzelnen Algorithmen im Zeitverlauf anzeigen. Dabei entspricht "Fehler 1" der Fehlerrate der Algorithmen bei der Ermittlung des defekten Systemzustands, "Fehler 2" der Fehlerrate der Algorithmen bei der Ermittlung des funktionierenden Systemzustands und der „Fehler ges.“ dem Durchschnittswert von beiden Fehlerraten.

Auf dem Abbildung 28 ist zu sehen, dass die SVM und NN Algorithmen bis etwa Zyklus 70 eine maximale Fehlerrate von etwa 0,3 aufweisen. Nach dem 160. Zyklen für die Datensätze FD001 und FD002 und nach dem 200. Zyklen für die Datensätze FD003 und FD004 verbessern sich die Leistungen der Algorithmen und sie erkennen den Systemzustand ohne Fehler. Die REG und DT erzielten während des gesamten Versuchs ungefähr die gleichen Leistungen. Zu Beginn lag die Erkennungsgenauigkeit bei etwa 50 % und im Laufe der Beobachtung verbesserte sich dieser Wert schrittweise. Nach dem 200. Zyklen für die Datensätze FD001 und FD002 und nach dem 300. Zyklen für die Datensätze FD003 und FD004 erkannten die Algorithmen die Systemzustände fehlerfrei. Beim NB sind die Ergebnisse ähnlich wie bei den REG und DT. Nur beginnt die fehlerfreie Zustandserkennung erst nach dem 250. Zyklen für die Datensätze FD001 und FD002 und nach dem 400. Zyklen für die Datensätze FD003 und FD004.

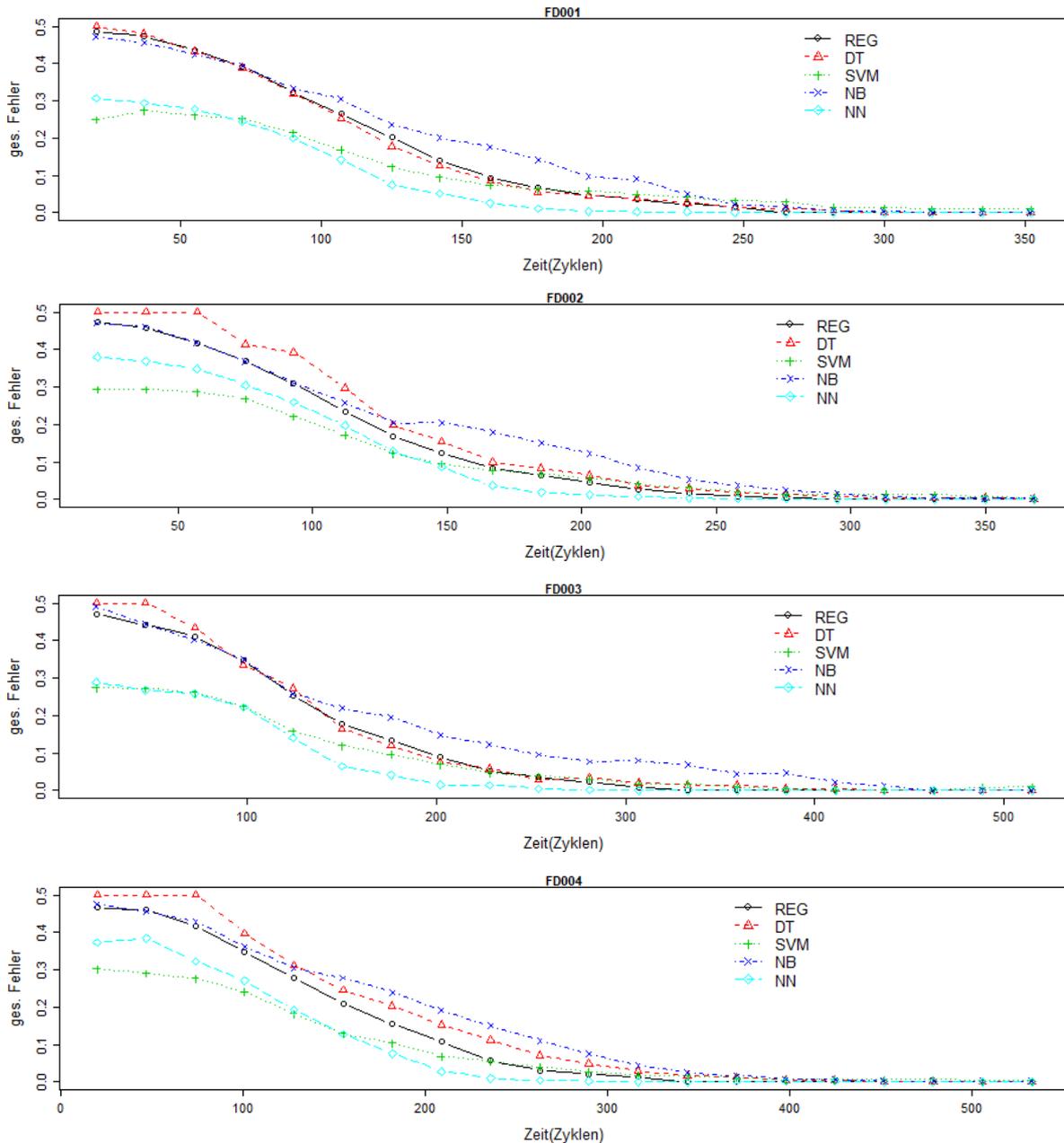


Abbildung 28: Mittlere Fehlerrate der überwachten Algorithmen bei der Ermittlung des Systemzustands

Abbildung 29 zeigt die Erkennungsgenauigkeit der Algorithmen im Falle von Systemausfällen. Die besten und zuverlässigsten Leistungen erbrachte der NN-Algorithmus. Dieser hatte zu Beginn der Beobachtung eine Genauigkeit von etwa 30%. Im Laufe der Beobachtung verbesserte sich die Erkennungsgenauigkeit, so dass nach dem 170. Zyklus für die Datensätze FD001 und FD002 und nach dem 250. Zyklus für die Datensätze FD003 und FD004 den Systemausfall fehlerlos erkannt wurde. Auch der REG erzielte gute und stabile Ergebnisse mit einer Genauigkeit von etwa 50% zu Beginn der Beobachtung. Nach dem 270. Zyklus für die Datensätze FD001 und FD002 und nach dem 350. Zyklus für die Datensätze FD003, FD004 wurde der Systemausfall mit einer Genauigkeit von 100 % ermittelt. Bei den Algorithmen NB, SVM und

DT wurden ebenfalls schwankende und geringe Genauigkeiten erzielt. NB und DT hatten zu Beginn der Beobachtung instabile Ergebnisse, der SVM dagegen wies am Ende der Beobachtung eine enorme große Fehlerrate auf.

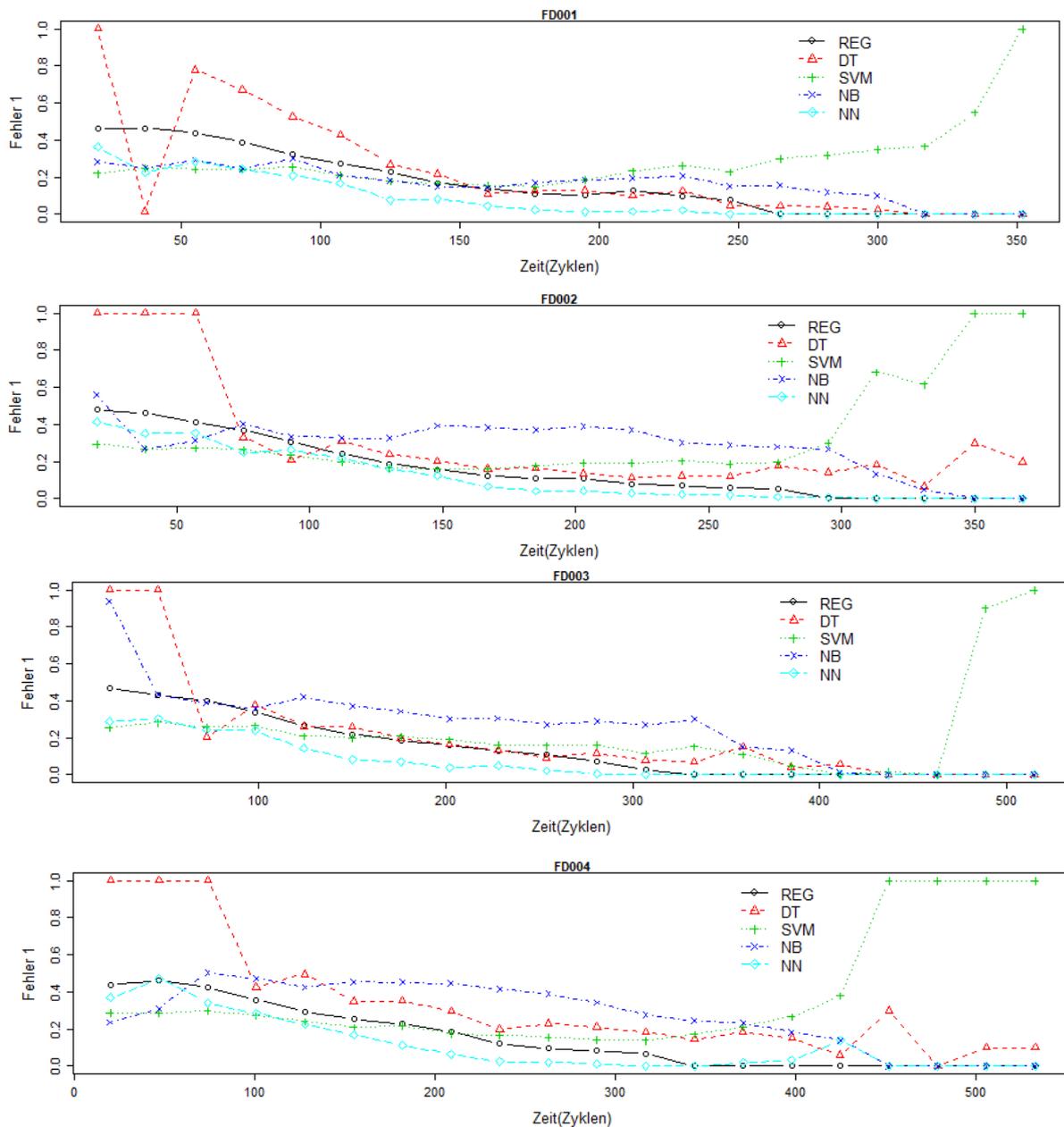


Abbildung 29: Fehlerrate der überwachten Algorithmen bei der Ermittlung des Systemausfalls

Aus Abbildung 30 ist ersichtlich, mit welcher Genauigkeit die Algorithmen den betriebsfähigen Systemzustand identifizieren. Die NN und SVM haben die höchste Genauigkeit und Stabilität erreicht. Diese haben eine Erkennungsgenauigkeit von etwa 30% zu Beginn der Beobachtung erzielt. Mit der Zeit stieg die Erkennungsgenauigkeit, so dass nach dem 150. Zyklus für die Datensätze FD001 und FD002 und nach dem 200. Zyklus für Datensätze FD003 und FD004 der betriebsbereite Systemzustand korrekt festgestellt wurde. Der REG erzielte ebenfalls gute und stabile Ergebnisse mit einer Genauigkeit von etwa 50% zu Beginn der Beobachtung. Nach

dem 170. Zyklen für die Datensätze FD001 und FD002 und nach dem 250. Zyklen für die Datensätze FD003 und FD004 wurde der betriebsbereite Systemzustand mit 100%er Genauigkeit ermittelt. Die NB und DT zeigten zu Beginn der Beobachtung unstabile Treffer bei der Erkennung des funktionsfähigen Systemzustands. Nach dem 200. Zyklen für die Datensätze FD001 und FD002 und nach dem 250. Zyklen für die Datensätze FD003 und FD004 wurde der funktionsfähige Systemzustand mit hoher Genauigkeit erkannt.

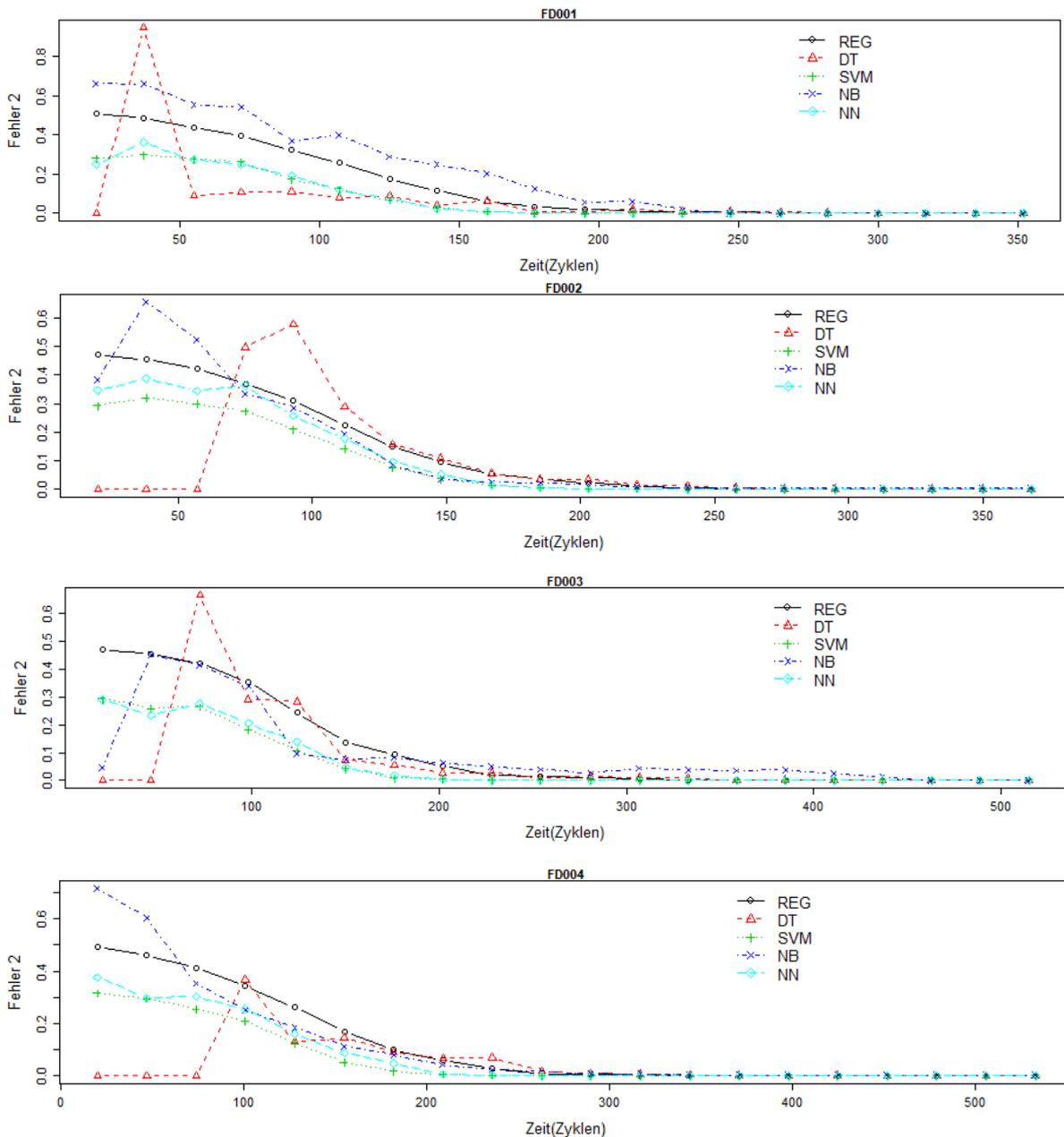


Abbildung 30: Fehlerrate der überwachten Algorithmen bei der Ermittlung des betriebsfähigen Systemzustands

Die oben genannten Ergebnisse sind in Tabelle 5 in numerischer Form dargestellt, nämlich die durchschnittliche von der Fehlerrate der Algorithmen.

Tabelle 5: Mittelwert von Fehlerrate der überwachten Algorithmen

Algorithmen / Datensatz		REG	DT	SVM	NB	NN
FD001	ges. Fehler	0,1501	0,14815	0,1025	0,17105	0,0846
	Fehler 1	0,1698	0,2325	0,29385	0,167	0,09
	Fehler 2	0,1418	0,0797	0,0762	0,20925	0,0812
FD002	ges. Fehler	0,1402	0,16615	0,10615	0,1691	0,10795
	Fehler 1	0,16115	0,3087	0,3387	0,2879	0,11855
	Fehler 2	0,1313	0,08995	0,0839	0,13005	0,1024
FD003	ges. Fehler	0,12205	0,12905	0,08335	0,154	0,0659
	Fehler 1	0,14055	0,21005	0,2352	0,26405	0,0742
	Fehler 2	0,11335	0,07345	0,05835	0,09395	0,06035
FD004	ges. Fehler	0,1282	0,1556	0,0902	0,1593	0,0898
	Fehler 1	0,15025	0,33905	0,38155	0,2759	0,1141
	Fehler 2	0,1173	0,0459	0,0642	0,1217	0,07695

Auf den Abbildungen 31-33 sind die Ergebnisse der Analyse der unüberwachten Algorithmen dargestellt, nämlich K-Means (KM) und Hierarchische Clusteranalyse (H).

In den Abbildungen 31 und 33 ist zu erkennen, dass die Genauigkeit der unüberwachten Algorithmen bei der Identifizierung des Systemzustands vom Datensatz abhängt. Die Datensätze FD001 und FD003 weisen bis zum Mitte der Beobachtung nahezu gleiche Ergebnisse zwischen den beiden Algorithmen auf. Nach dem 200. Zyklus (FD001) und nach dem 300. Zyklus (FD003) werden die Ergebnisse unstabil und die Erkennungsgenauigkeit der Algorithmen nimmt ab. Für die Datensätze FD002 und FD004 sind die Leistungen der beiden Algorithmen unterschiedlich. H-Algorithmus erkennt den Systemzustand mit einer Genauigkeit von etwa 50% während der ganzen Beobachtung. KM ist genauer, aber weniger stabil.

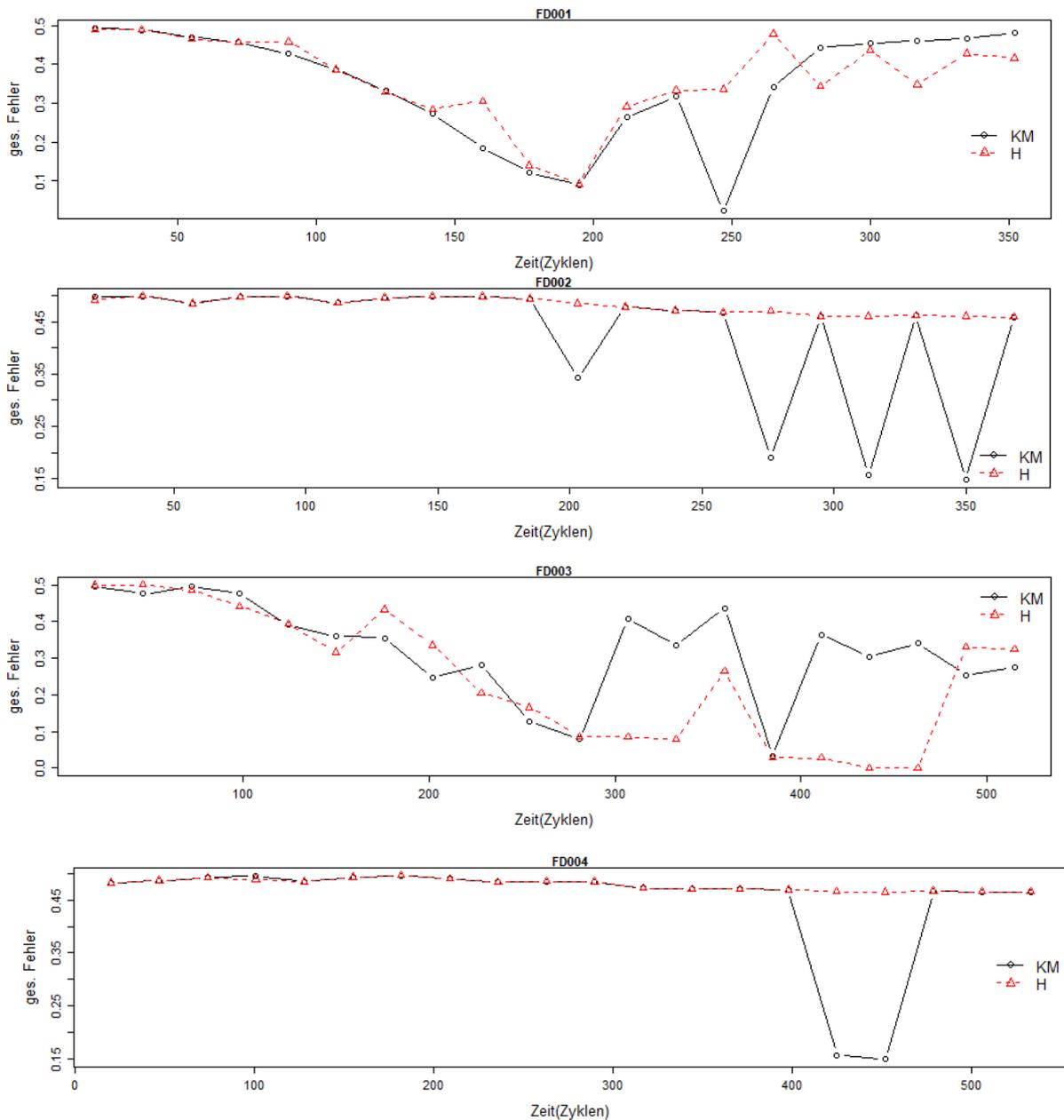


Abbildung 31: Mittlere Genauigkeit der unüberwachten Algorithmen bei der Ermittlung des Systemzustands

Aus Abbildung 32 lässt sich die Genauigkeit der Systemstanderkennung mittels unüberwachten Algorithmen im System ablesen. Wie aus der Abbildung ersichtlich ist, erkennen die unüberwachten Algorithmen Systemfehler eher schlecht. Die Genauigkeit der Systemfehlererkennung ist bei beiden Algorithmen für alle Datensätze instabil und verschlechtert sich mit der Zeit.

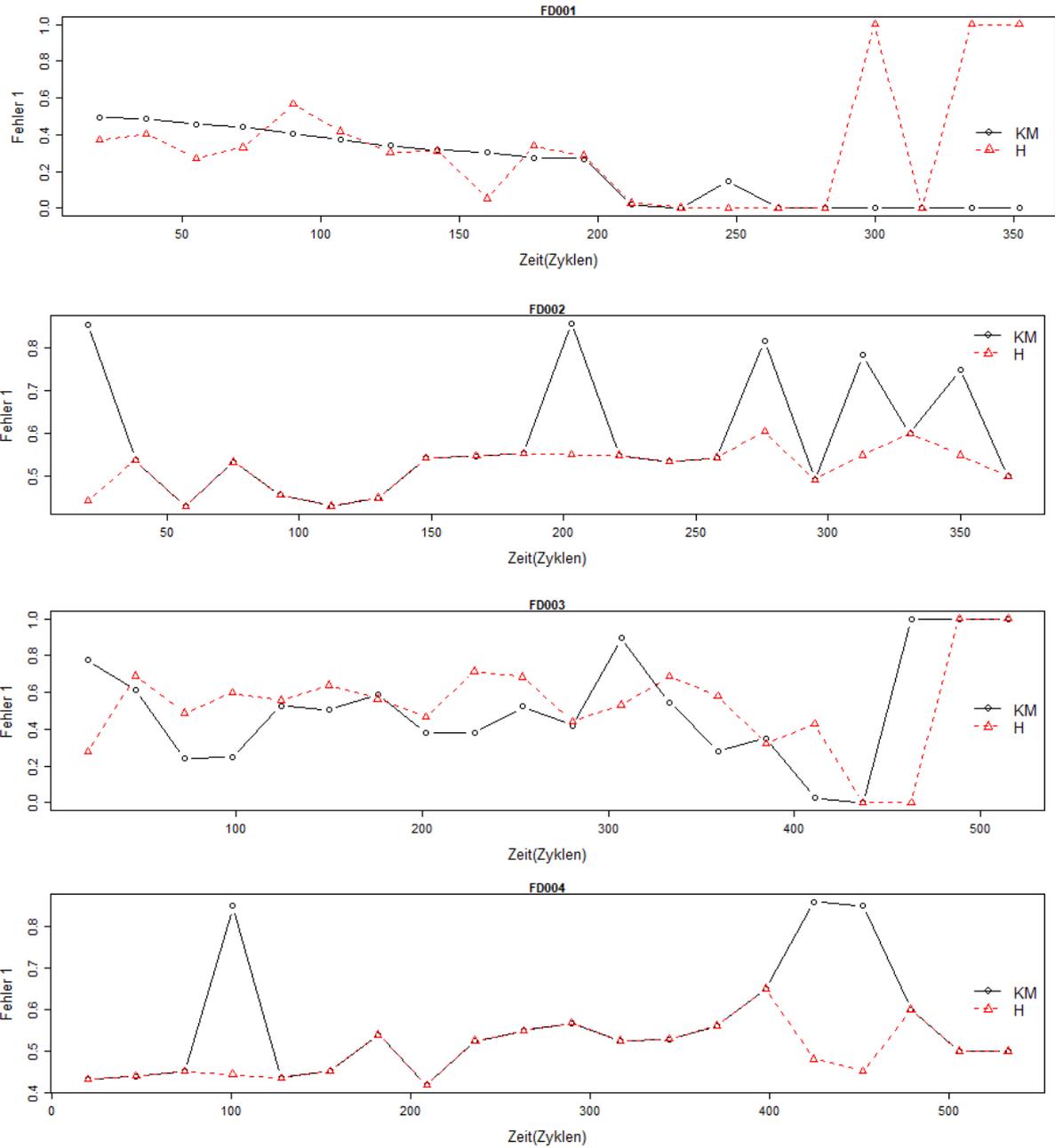


Abbildung 32: Genauigkeit der unüberwachten Algorithmen bei der Ermittlung des Systemausfalls

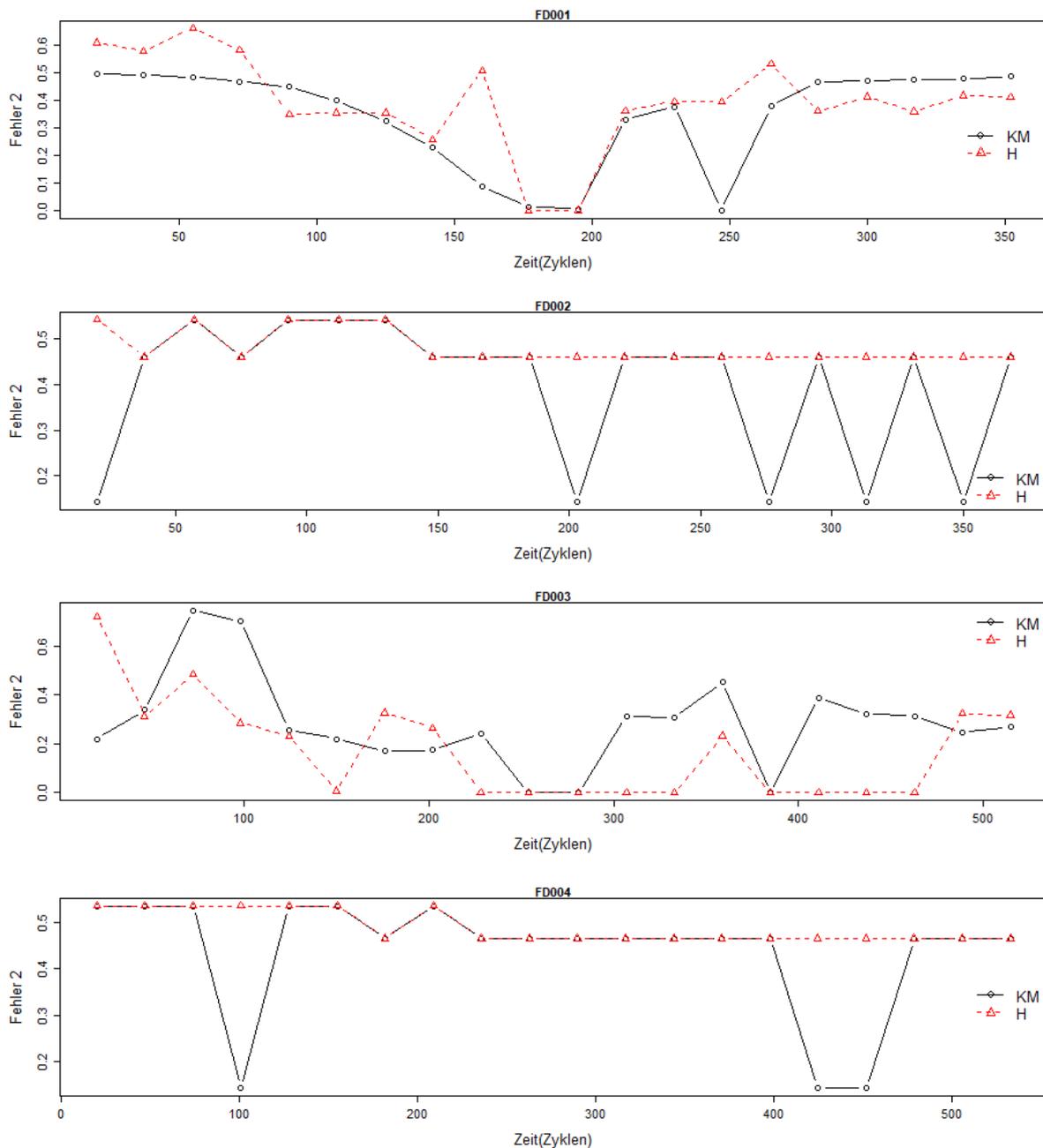


Abbildung 33: Genauigkeit der unüberwachten Algorithmen bei der Ermittlung des betriebsfähigen Systemzustands

Aus Tabelle 6 lässt sich die durchschnittliche Genauigkeit der nicht überwachten Algorithmen bei der Erkennung des Systemzustands ablesen, was auch das zuvor dargestellte grafische Ergebnis widerspiegelt.

Tabelle 6: Mittelwert der Fehlerrate von unüberwachten Algorithmen

Datensatz	Algorithmen	ges. Fehler	Fehler 1	Fehler 2
FD001	KM	0,34875	0,2165	0,34555
	H	0,3647	0,33435	0,3939
FD002	KM	0,42925	0,58835	0,39615
	H	0,481	0,5203	0,4795
FD003	KM	0,3265	0,5151	0,2839
	H	0,24975	0,53315	0,17495
FD004	KM	0,44845	0,5611	0,43755
	H	0,47945	0,5018	0,4895

### 3.4 Bias-Problem

Anhand der vorliegenden Ergebnisse lässt sich erkennen, dass die Genauigkeit der Identifizierung von Systemausfällen für alle Arten der Algorithmen nach dem 300. Zyklus für die Datensätze FD001 und FD002 und nach dem 400. Zyklus für die Datensätze FD003 und FD004 abnimmt. Dies lässt sich auf den Mangel an Daten zurückführen, da viele Turbinenanlagen ausgefallen sind, bevor sie 300 Betriebszyklen (FD001, FD002) und 400 Betriebszyklen (FD003, FD004) erreicht haben. Für die Systemzustandsanalyse hat jeder einzelnen Stichprobe eine große Wirkung auf die Genauigkeit der Algorithmen. Auf Abbildung 34 sind Histogramme dargestellt, die die Anzahl der Betriebszyklen für jede Turbinenanlage repräsentieren und bestätigen, dass die meisten Turbinenanlagen nach etwa 200 Zyklen ausfallen. Aus dem Datensatz FD001 ist ersichtlich, dass aus den 100 Turbinenanlagen nur 58 Turbinenanlagen weniger als 200 Zyklen aufweisen, von denen 16 Turbinenanlagen nach 150 Zyklen ausfallen.

Ein zweiter wichtiger Aspekt, der sich aus der uneinheitlichen Anzahl von Betriebszyklen ergibt, ist das Degradationsverhältnis der Betriebszyklen für jede Turbinenanlage im Laufe der Zeit unterschiedlich. Dies bedeutet, dass eine bestimmte Zykluszahl für verschiedene Turbinenanlagen unterschiedliche Degradationsgrade aufweisen, was sich erheblich auf die Klassifizierungsfähigkeit der Algorithmen auswirkt. So wird beispielsweise eine Turbinenanlage, die nach dem 120. Zyklus ausfällt, zum Zeitpunkt des 100. Zyklus eine viel höhere Degradationsrate aufweisen als eine andere Turbinenanlage, die erst nach dem 350. Zyklus ausfallen wird.

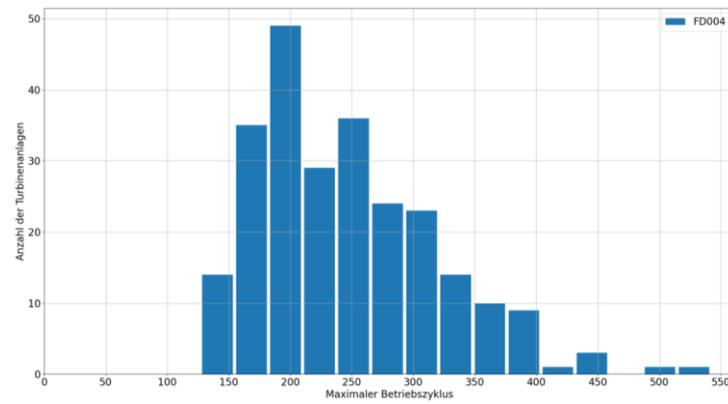
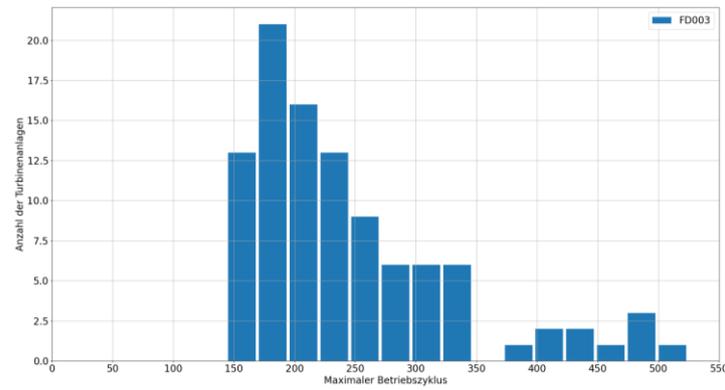
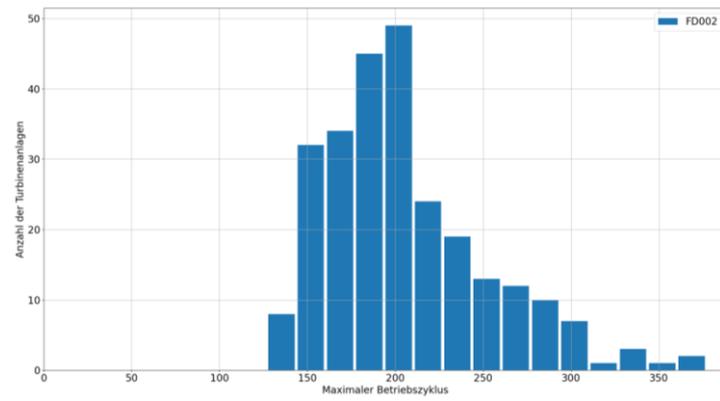
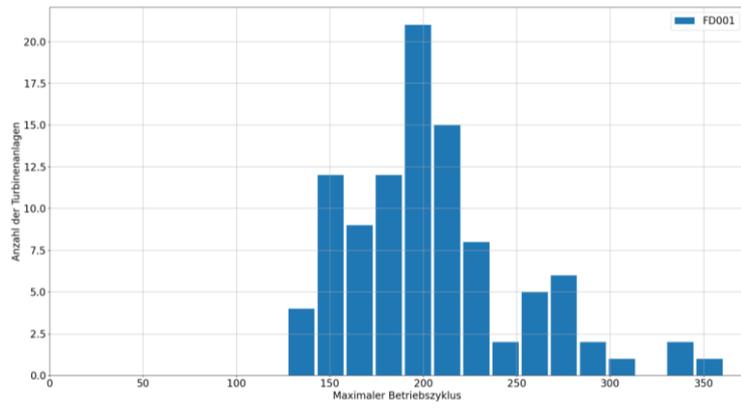


Abbildung 34: Darstellung von Anzahl der Betriebszyklen für jede Turbinenanlage

Als Lösung für dieses Problem und um ein einheitliches Degrationsmaß für alle Turbinen zu erhalten, wurde eine Neuabtastung (engl. *resampling*) der Datensätze vorgenommen. Neuabtastung generiert eine eindeutige Stichprobenverteilung auf Basis der tatsächlichen Daten. Nach dieser Methode wurde ein neuer Datensatz erzeugt, so dass alle Turbinenanlagen 100 Betriebszyklen erreicht haben (siehe Abbildung 35). Für die Normalisierung konnte eine beliebige Anzahl von Zyklen festgelegt werden, aber um die Interpretation der Ergebnisse zu vereinfachen, wurden 100 Zyklen gewählt, was auch den Prozentsatz der Degradation zeigt. Das bedeutet, dass im ersten Betriebszyklus alle Turbinenanlagen in Betrieb sind und im 100. Betriebszyklus alle Turbinenanlagen ausgefallen sind (100 % Degradation).

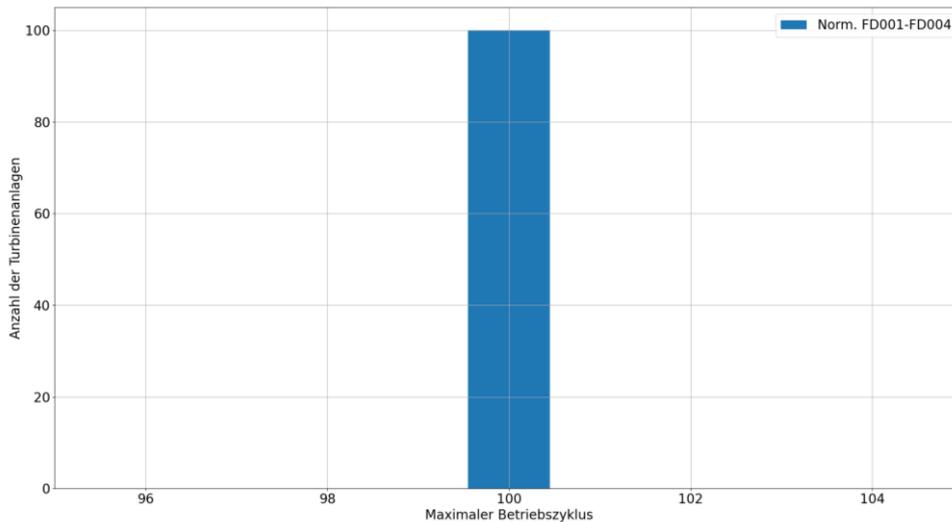


Abbildung 35: Darstellung von Datenverteilung der normalisierten Datensatz

Dieses Verfahren kann als Methode zur Normalisierung der Datensätze verwendet werden. Die Ergebnisse der Analyse von normalisierten Datensätzen sind aussagekräftig, weil sie auch dem tatsächlichen Degradationsverlauf entsprechen. Datennormalisierung wurde mittels Funktion *resample* der Pandas-Bibliothek durchgeführt.

### 3.5 Ergebnisse nach der Normalisierung der Datensätze

In den Abbildungen 36-38 sind die Ergebnisse der Analyse der überwachten Algorithmen basierend auf normalisierten Datensätzen dargestellt.

Auf dem Abbildung 36 ist zu sehen, dass alle fünf überwachten Algorithmen eine zuverlässige Klassifikationsleistung erbracht haben. Eine höhere Erkennungsgenauigkeit wurde mit den NN- und SVM-Algorithmen erreicht. Der NN erkennt den Systemzustand nach 80% der Betriebszyklen für die Datensätze FD001 und FD002 und nach 70% der Betriebszyklen für die Datensätze FD003 und FG004 ohne Fehler. Die anderen drei Algorithmen NB, REG und DT erreichten ungefähr das gleiche Genauigkeitsniveau. Der REG war in der Lage, den Systemzustand nach 85% der Betriebszyklen fehlerfrei zu erkennen. Der DT konnte für die Datensätze FD001 und FD003 nach 85% der Betriebszyklen den Systemzustand fehlerfrei erkennen, aber für die Datensätze FD002 und FD004 erreichte der Algorithmus sogar nach 90% der Betriebs-

zyklen keine fehlerfreie Systemzustandserkennung. Der NB konnte nur für den ersten Datensatz FD001 nach dem 86% der Betriebszyklen eine fehlerfreie Systemzustandserkennung erreichen. Für die anderen Datensätze war der Algorithmus nach 90% der Betriebszyklen nicht in der Lage, eine fehlerfreie Systemzustandserkennung zu erreichen.

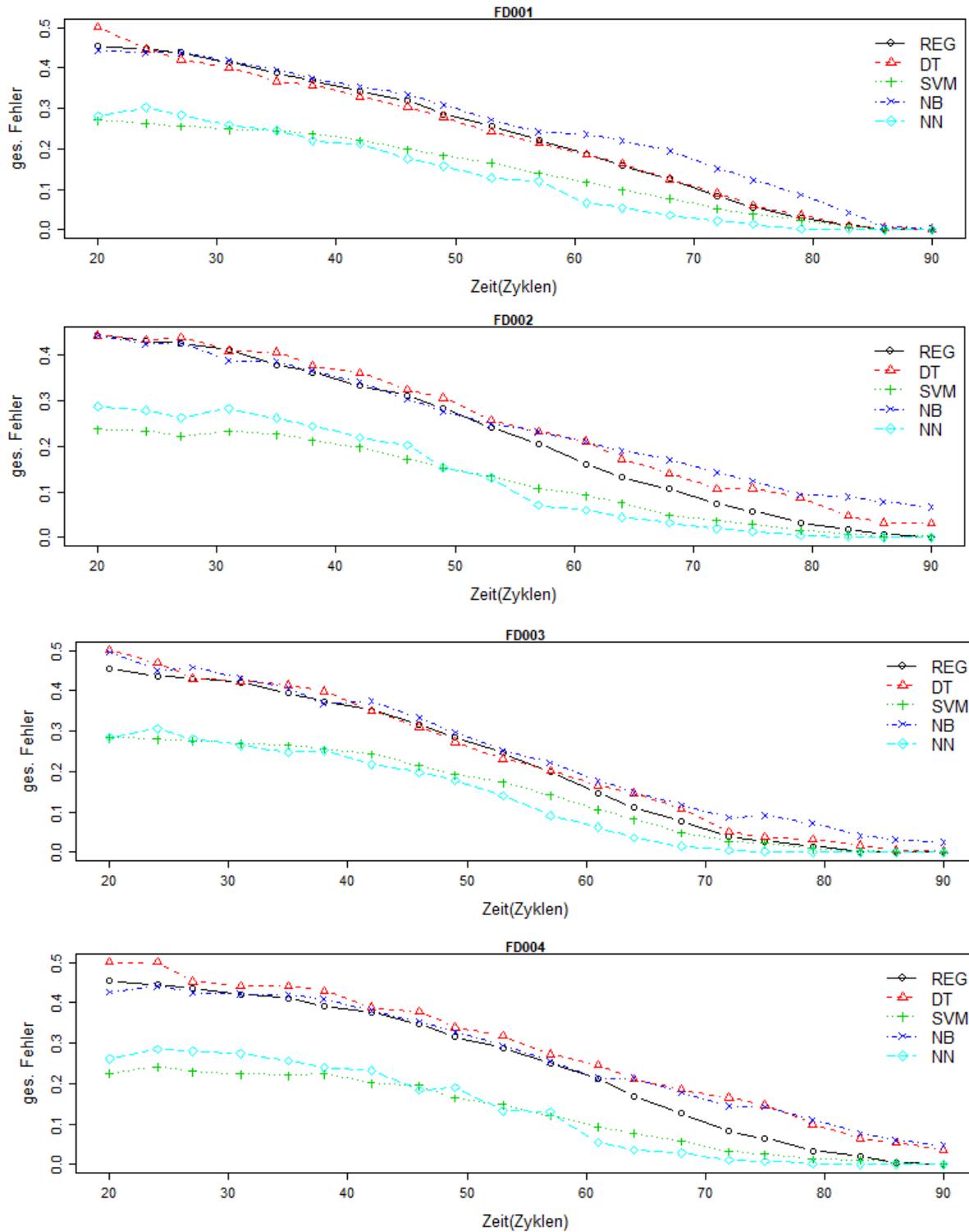


Abbildung 36: Mittlere Genauigkeit der überwachten Algorithmen bei der Ermittlung des Systemzustands anhand der normalisierten Daten

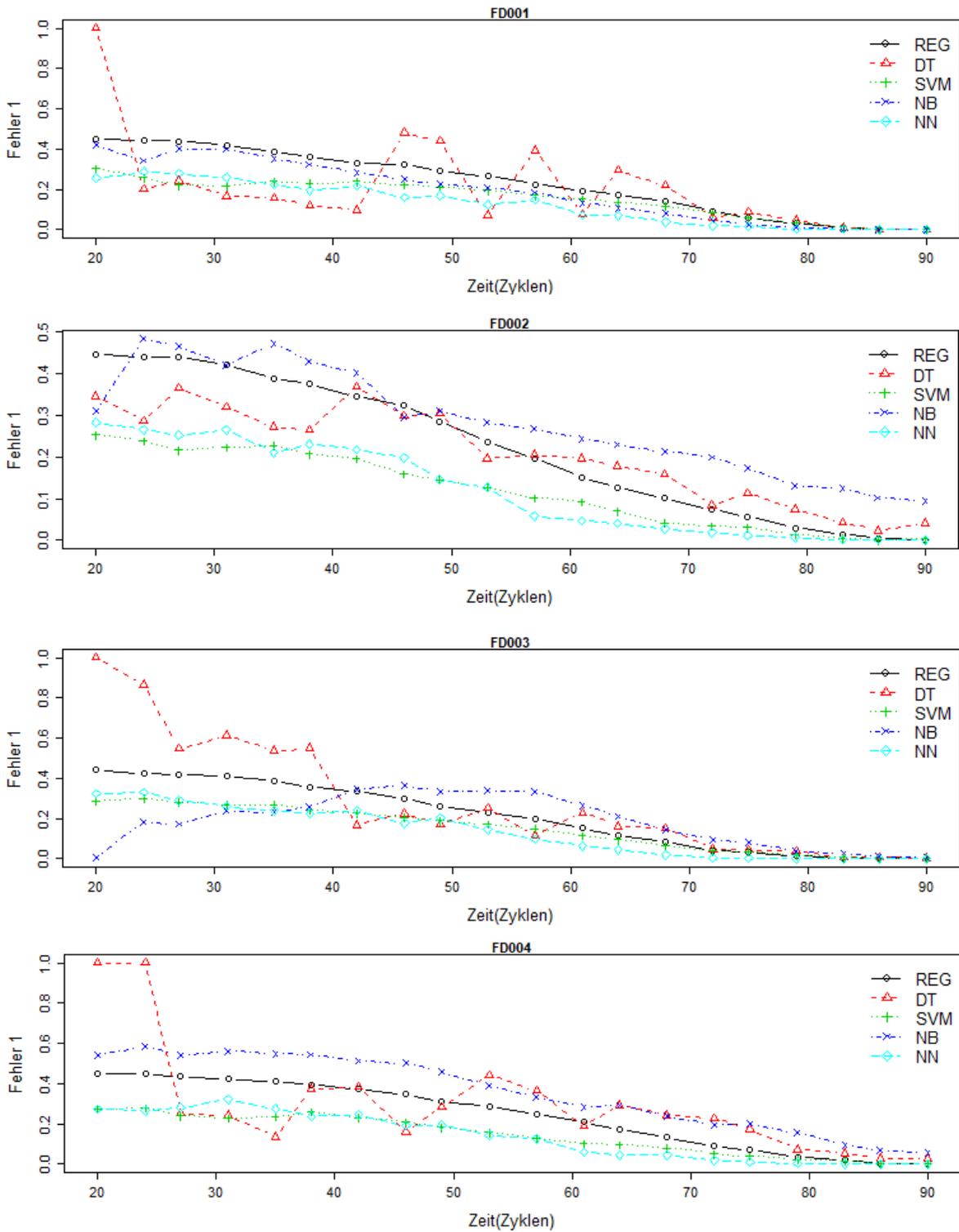


Abbildung 37: Genauigkeit der überwachten Algorithmen bei der Ermittlung des Systemausfalls anhand der normalisierten Daten

Aus den neuen Ergebnissen zur Erkennung von Systemausfällen (siehe Abbildung 37) wird deutlich, dass der NN-Algorithmus eine immer noch hohe Genauigkeit aufweist. Die Genauigkeit des SVM-Algorithmus erheblich verbessert und stabilisiert wurde, sodass keine unangemessenen Schwankungen bei der Klassifizierung des Systemzustands im Laufe der Zeit auftreten.

ten. Der NN und SVM konnten den Systemausfall nach 80% der Betriebszyklen für die Datensätze FD001 und FD002 und nach 70% der Betriebszyklen für die Datensätze FD003 und FD004 ohne Fehler erkennen. Der REG und NB zeigten ebenfalls recht stabile, aber weniger genaue Ergebnisse im Vergleich zu NN und SVM. Die REG konnte den Systemausfall nach dem 80% der Betriebszyklen für die Datensätze FD001 und FD003 und nach dem 85% der Betriebszyklen für die Datensätze FD002 und FD004 korrekt ermitteln. Die Klassifizierungsgenauigkeit von Systemausfällen der DT ist unstabil, so dass am Anfang der Beobachtung wenig genaue Ergebnisse liefert, aber manchmal in bestimmten Zeitintervalle eine höhere Genauigkeit als NN oder SVM aufweisen. Für den Datensätze FD001 und FD003 konnten die DT und NB nach dem 80 % der Betriebszyklen den Systemausfall fehlerfrei zu identifizieren, aber für Datensätze FD003 und FD004 waren die Algorithmen nicht in der Lage, nach dem 90% der Betriebszyklen der Systemausfall ohne Fehler zu erkennen.

Aus Abbildung 38 ist ersichtlich, dass bei der Erkennung der betriebsfähige Systemzustand die NN und SVM haben die höchste Genauigkeit und Stabilität erreicht und nach dem 70% der Betriebszyklen erkennen die Algorithmen fehlerlose Systemzustandserkennung. Der REG erzielte ebenfalls gute und stabile Ergebnisse mit einer Erkennungsgenauigkeit von etwa 50% zu Beginn der Beobachtung und nach dem 80% der Betriebszyklen wurde der betriebsbereite Systemzustand mit 100%er Genauigkeit ermittelt. NB und DT zeigen unterschiedliche Ergebnisse für verschiedene Datensätze. Beispielsweise zeigte NB eine gute Stabilität und Genauigkeit bei der Klassifizierung des betrieblichen Systemzustands für die Datensätze FD004 und FD002 und eine weniger genaue Klassifizierung für die Datensätze FD001 und FD003. Der DT hatte die schlechteste Klassifizierungsgenauigkeit unter allen Algorithmen (außer den Datensatz FD003) und war während der Beobachtung für die Datensätze FD001, FD003 und FD004 weniger stabil.

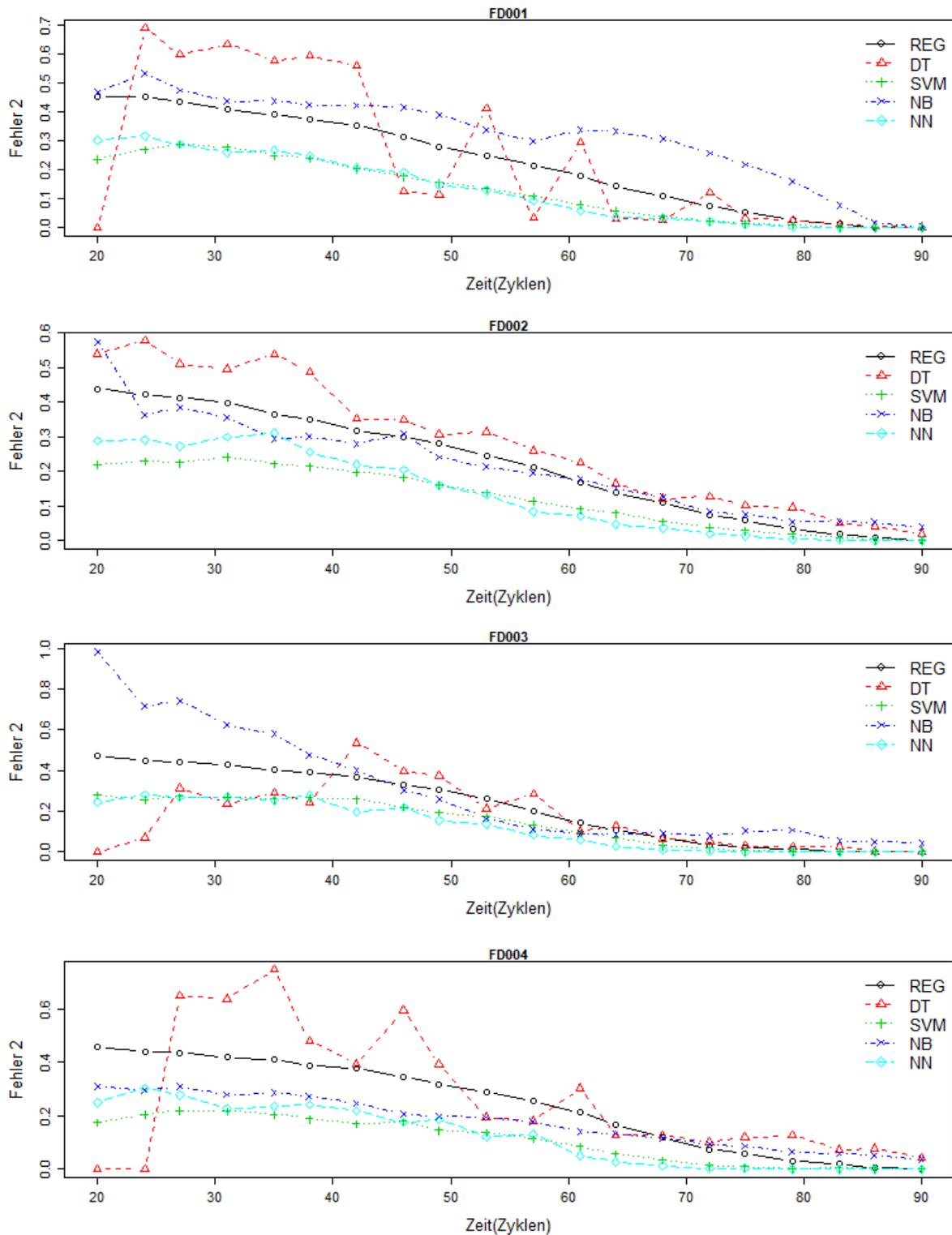


Abbildung 38: Genauigkeit der überwachten Algorithmen bei der Ermittlung des betriebsfähigen Systemzustands anhand der normalisierten Daten

In der Tabelle 7 ist die Mittelwert von den Ergebnissen der überwachten Algorithmen Anhang normalisierten Datensätze dargestellt.

Tabelle 7: Mittelwert der Fehlerrate von überwachten Algorithmen anhand der normalisierten Datensätze

Datensatz/Algorithmen		REG	DT	SVM	NB	NN
FD001 (norm)	ges. Fehler	0,22945	0,2265	0,14255	0,2541	0,1271
	Fehler 1	0,23215	0,2081	0,15595	0,1901	0,1266
	Fehler 2	0,2268	0,24485	0,1294	0,3179	0,12745
FD002 (norm)	ges. Fehler	0,2203	0,24385	0,12205	0,24945	0,12115
	Fehler 1	0,22275	0,207	0,11965	0,2752	0,1142
	Fehler 2	0,2179	0,284	0,1243	0,2164	0,12815
FD003 (norm)	ges. Fehler	0,21635	0,22815	0,1451	0,24355	0,1294
	Fehler 1	0,1991	0,2864	0,1491	0,1836	0,12335
	Fehler 2	0,2225	0,1695	0,141	0,30355	0,1354
FD004 (norm)	ges. Fehler	0,24235	0,2831	0,12595	0,26685	0,12615
	Fehler 1	0,2429	0,2975	0,14275	0,355	0,1338
	Fehler 2	0,2418	0,26895	0,10925	0,1786	0,11835

Tabelle 7 zeigt die Mittelwerte der Fehlerrate der überwachten Algorithmen bei der Erkennung des Systemzustands, die die oben genannten Ergebnisse auch ergänzen. Wie aus den Mittelwerten der Genauigkeit der Algorithmen ersichtlich ist, waren die NN und SVM für jeden normalisierten Datensatz am genauesten.

Obwohl sich der Mittelwert der Fehlerrate der Algorithmen im Allgemeinen verschlechtert hat, entsprechen die neuen Mittelwerte aufgrund der Datennormalisierung eher der tatsächlichen Erkennungsgenauigkeit der Algorithmen.

Auf die gleiche Weise wurde eine Analyse anhand normalisierter Datensätze zur Erkennung des Systemzustands durch unüberwachte Algorithmen durchgeführt. In den Abbildungen 39-41 sind die Ergebnisse dieser Analyse abgebildet.

In den Abbildungen 39 ist zu erkennen, dass die Genauigkeit der unüberwachten Algorithmen bei der Systemzustandserkennung vom Datensatz abhängig ist. Für den ersten Datensatz FD001 zeigen also beide unüberwachten Algorithmen gleiche Genauigkeit der Systemzustandserkennung. Zu Beginn des Versuchs erkennen beide Algorithmen den Systemzustand mit 50% Genauigkeit, dann verbessert sich die Genauigkeit und nach dem 90% der Betriebszyklen beträgt die Genauigkeit der beiden Algorithmen 100%. Ein ähnliches Verhalten der Algorithmen ist im dritten Datensatz FD003 zu vermerken, d.h. beide Algorithmen zeigen zu Beginn des Versuchs eine 50-prozentige Genauigkeit bei der Erkennung des Systemzustands und dann verbessern sich die Ergebnisse ebenfalls, aber nach 90% der Betriebszyklen erkennen beide Algorithmen den Systemzustand nur noch mit etwa 30% Genauigkeit. Für den zweiten und vierten Datensatz (FD002, FD004) ist die Genauigkeit der Erkennung des Systemzustands bei beiden Algorithmen fast gleich und liegt während der gesamten Beobachtung bei etwa 50%.

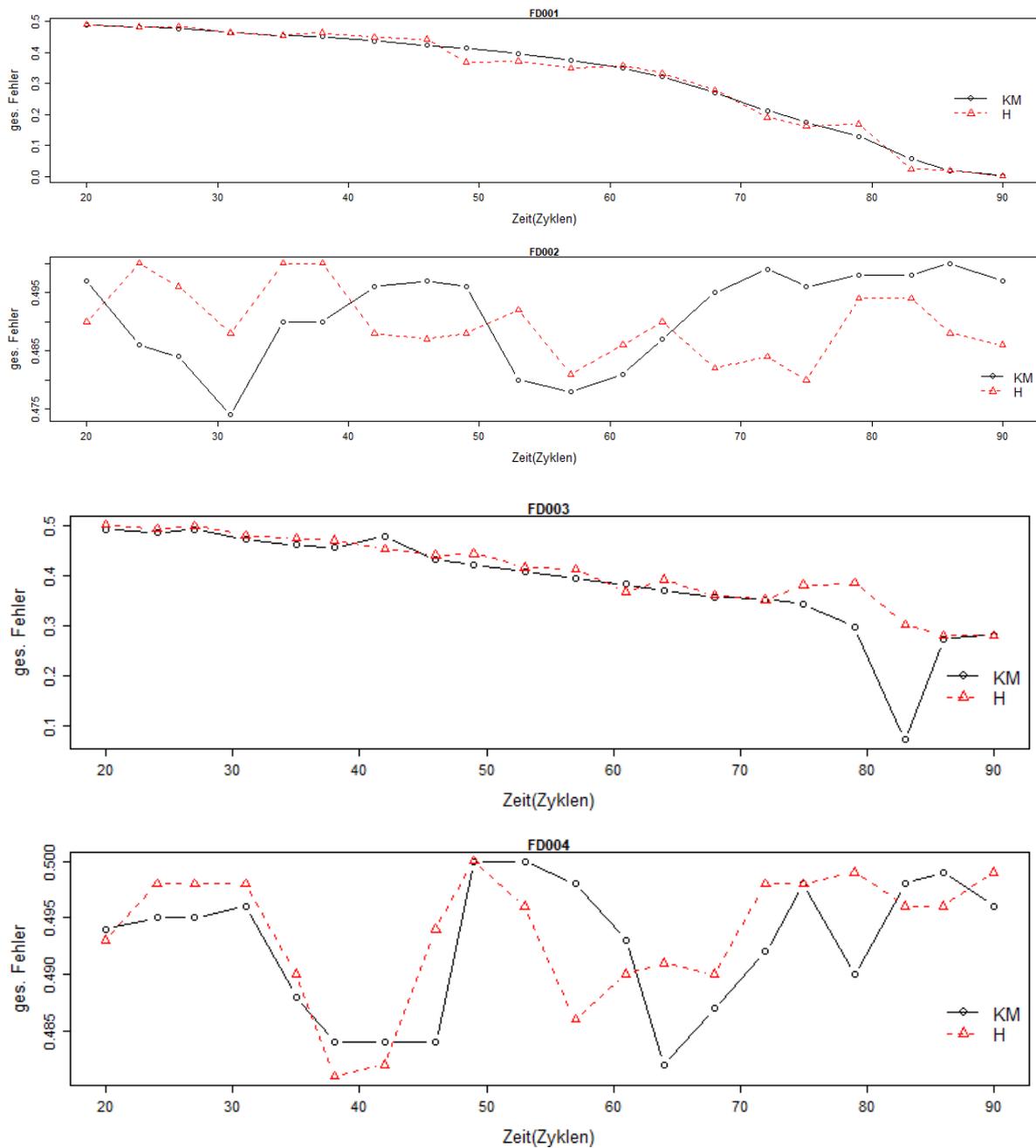


Abbildung 39: Mittlere Genauigkeit der unüberwachten Algorithmen bei der Ermittlung des Systemzustands anhand normalisierter Datensätze

In Abbildung 40 ist die Genauigkeit der Erkennung von Systemausfällen durch unüberwachte Algorithmen anhand von normalisierten Daten dargestellt. Wie es zu sehen ist, unterscheidet sich die Genauigkeit der Ausfallerkennung auch je nach Datensatz. Allein für den ersten Datensatz FD001 erreichten beide Algorithmen eine fehlerfreie Erkennung von Systemausfällen nach dem etwa 85% der Betriebszyklen. Für den dritten Datensatz zeigten beide Algorithmen eher un stabile Ergebnisse bei der Erkennung von Systemausfällen. Die Unstabilität dauerte vom Beginn des Versuchs bis etwa zur Mitte (50%) der Betriebszyklen an, dann folgte ein kurzes Zeitintervall mit 50%-er stabile Erkennungsgenauigkeit, und nach Erreichen von 80% der Betriebszyklen wurde der KM bei der Erkennung von Systemausfällen wieder un stabil. Die

Genauigkeit des KM für den zweiten und vierten Datensatz (FD002, FD004) liegt für die gesamte Beobachtungsdauer bei etwa 50%. Die Genauigkeit des H-Algorithmus schwankt während der ganze Beobachtung zwischen 30% und 70%. Sobald er 70% der Betriebszyklen erreicht hat, bleibt die Genauigkeit der Fehlererkennung bei etwa 70% liegen.

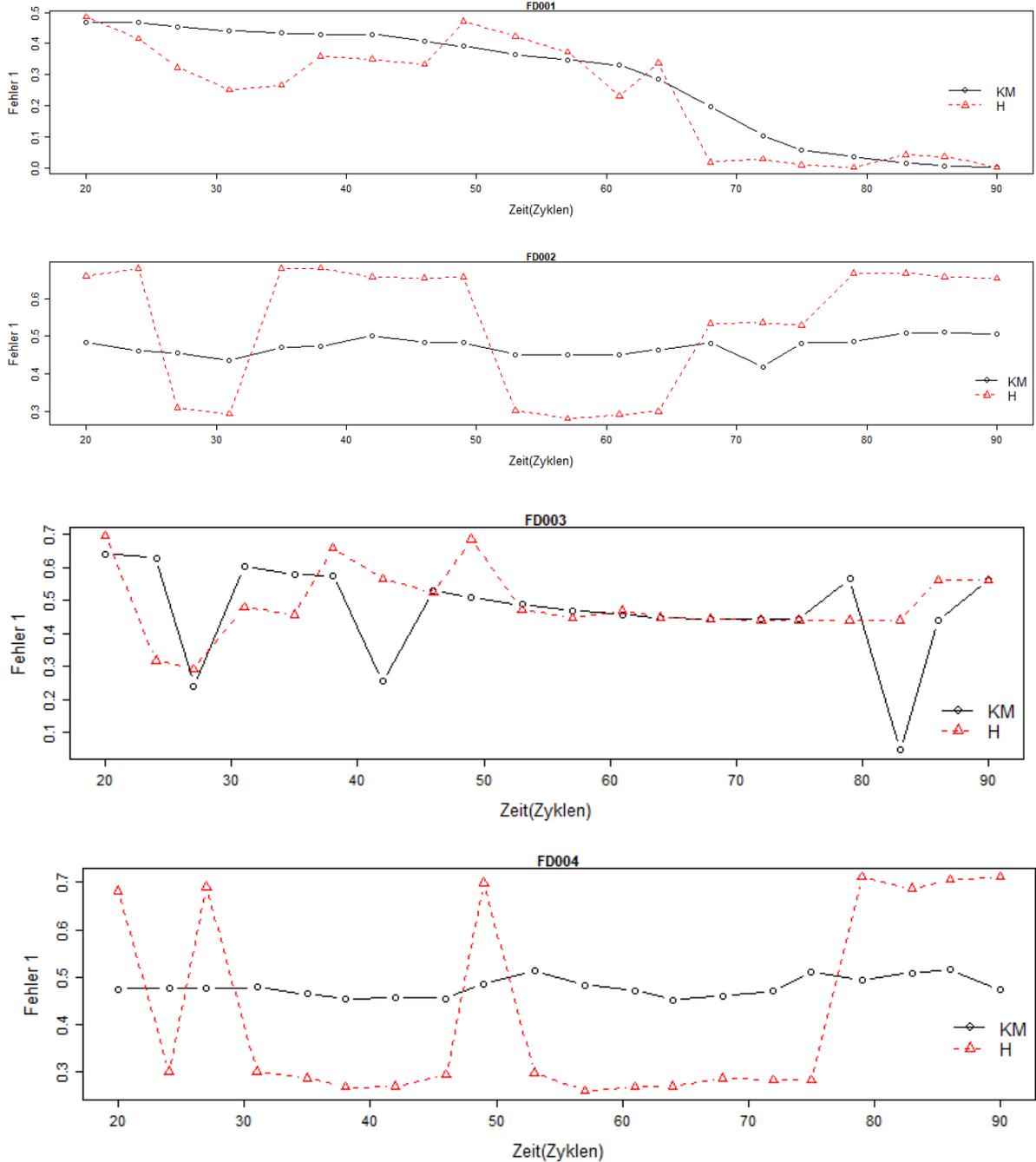


Abbildung 40: Genauigkeit der unüberwachten Algorithmen bei der Ermittlung des Systemausfalls anhand normalisierter Datensätze

Auf dem Abbildung 41 ist die Genauigkeit der Erkennung des Systemzustands von unüberwachten Algorithmen anhand normalisierter Daten dargestellt. Für den ersten und dritten Datensatz (FD001, FD003) liegt die Erkennungsgenauigkeit der KM und H für den betriebsfähigen Systemzustand während der ganze Beobachtung bei etwa 50%. Erst nach 90% der Betriebszyklen für den KM und nach 85% der Betriebszyklen für den H identifizieren den Algorithmen den Systemzustand fehlerfrei. Für den zweiten und vierten Datensatz (FD002, FD004) erkennt der KM der betriebliche Systemzustand mit Genauigkeit etwa 50% während der ganze Beobachtung.

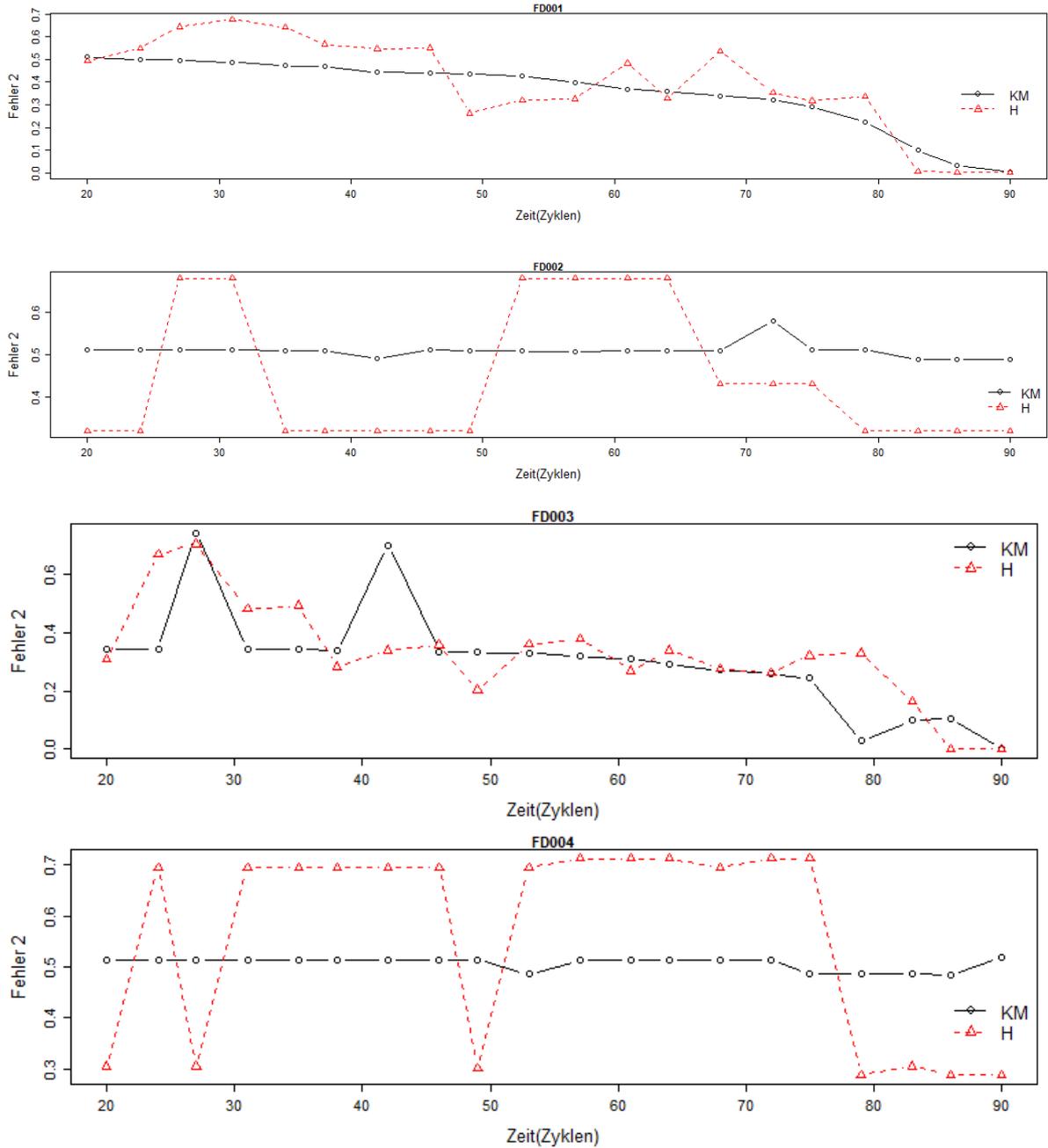


Abbildung 41: Genauigkeit der unüberwachten Algorithmen bei der Ermittlung des betriebsfähigen Systemzustands anhand normalisierter Datensätze

Tabelle 8: Mittelwert der Fehlerrate von unüberwachten Algorithmen anhand normalisierter Datensätze

Datensatz	Algorithmen	ges. Fehler	Fehler 1	Fehler 2
FD001 (norm)	KM	0,31975	0,2836	0,35575
	H	0,31755	0,2382	0,39665
FD002 (norm)	KM	0,49095	0,4729	0,50905
	H	0,4897	0,5349	0,4444
FD003 (norm)	KM	0,3942	0,49305	0,2952
	H	0,40865	0,4913	0,32595
FD004 (norm)	KM	0,49275	0,4785	0,5071
	H	0,49365	0,42715	0,56005

Tabelle 8 zeigt die Mittelwerte der Fehlerrate der unüberwachten Algorithmen bei der Erkennung des Systemzustands, die die oben genannten Ergebnisse auch vervollständigen. Wie aus den Mittelwerten der Fehlerraten ersichtlich ist, zeigten beide Algorithmen untereinander ungefähr die gleichen Ergebnisse.

In die Tabelle 9 wurden die Mittelwert der ges. Fehlerrate von allen Algorithmen und für alle Datensätze zusammengefasst.

Tabelle 9: Mittelwert der ges. Fehlerrate von allen Arten der Algorithmen anhand normalisierter Datensätze

Mittelwert	REG	DT	SVM	NB	NN	KM	H
ges. Fehler	0,2271	0,2454	0,1339	0,2535	0,1259	0,4244	0,4274

Wie es aus Tabelle 9 zu sehen ist, liefern die zur Analyse verwendeten Algorithmen unterschiedliche Ergebnisse. Diese Unterschiede sind auf die spezifischen Eigenschaften der einzelnen Algorithmen zurückzuführen. Die Klassifizierungsfähigkeit hängt entscheidend von der Fähigkeit eines datenbasierten Algorithmus ab, den momentanen Zustand eines technischen Systems anhand von Sensordaten abzuschätzen. Die Ergebnisse haben gezeigt, dass der Typ des Algorithmus (überwacht oder unüberwacht) eine wichtige Rolle spielt. Beim Vergleich der Algorithmen lässt sich feststellen, dass die überwachten Algorithmen genauere und stabilere Ergebnisse liefern (Abbildung 42). Unter den überwachten Algorithmen haben die künstliche neuronale Netz mit dem Fehlerrate von 0,1259 und Support Vector Machine mit dem Fehlerrate von 0,1339 das beste Ergebnis erzielt.

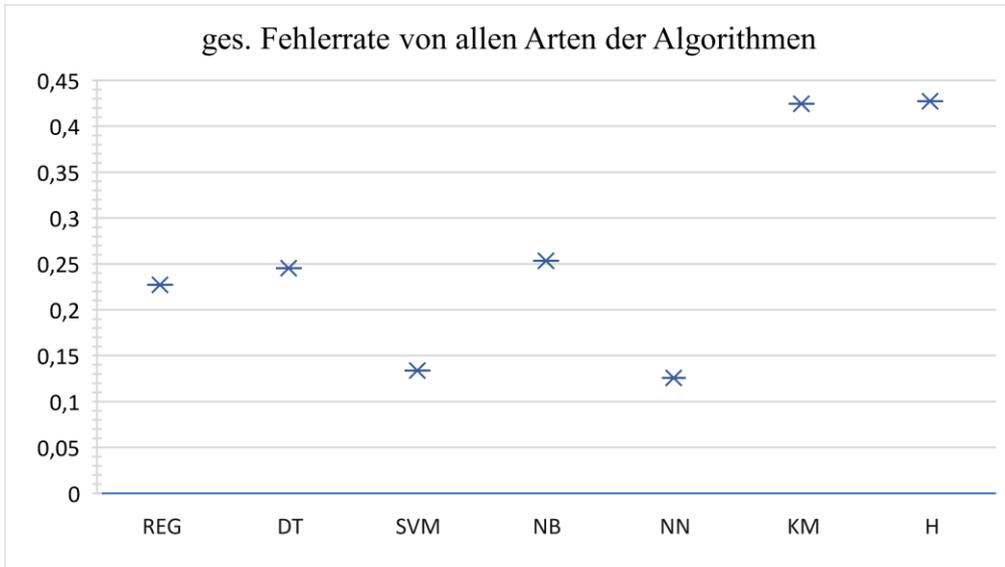


Abbildung 42: Darstellung der ges. Fehlerrate von allen Algorithmen

## 4 Fazit

Im Rahmen dieser Arbeit wurde wichtige Elementen der Zuverlässigkeitsanalyse, Instandhaltungsstrategien und Methoden der vorausschauenden Instandhaltung im Kontext von Industrie 4.0 untersucht. Insbesondere wurde die Auswirkungen der Datenverfügbarkeit auf die Zuverlässigkeitsanalyse und –bewertung betrachtet. Die Verfügbarkeit neuer Informations- und Kommunikationstechnologien hat sich mit der Zeit relativ geändert. Die größte Neuerung ist die einfache Erfassung und Verarbeitung von Daten, was auch ein großes Potenzial im Bereich der vorausschauenden Instandhaltung aufweist. Das Potenzial wird in der Möglichkeit gesehen, Schäden oder Mängel an Produktionsanlagen oder Endprodukten frühzeitig zu erkennen. Diese Instandhaltungsstrategie ist deshalb so innovativ, da sie Schäden verhindern kann, bevor sie zu Produktionsausfällen mit sehr hohen Kosten führen. Unter prädiktiver Instandhaltung versteht man eine Instandhaltungsstrategie, bei der der Anwender durch Erfassung und Überwachung von Echtzeit-Zustandsdaten mittels Sensoren den Ausfallzeitpunkt eines technischen Systems vorhersagen kann. Diese Daten werden dann mit Algorithmen aus verschiedenen Auswertemethoden analysiert und auf Basis der so gewonnenen Erkenntnisse wird die Wartung und Instandhaltung entsprechend eingeplant. Auf diese Weise kann kein Teil des Systems zu irgendeinem Zeitpunkt während des Betriebs durch normalen technischen Abnutzung defekt werden.

Außerdem wurden verschiedene Methoden der prädiktiven Instandhaltung implementiert, ausgewertet und miteinander verglichen. Das Schwerpunkt der Implementierung lag auf dem Zustand des technischen Systems auf Basis von Sensordaten abzuschätzen und klassifizieren. Dafür wurden die folgenden Methoden zur Klassifizierung verwendet: logistische Regression, künstliches neuronales Netz, Support Vector Machine, Naive Bayes, Entscheidungsbaumanalyse, K-Means-Algorithmus und Hierarchische Clusteranalyse. Für die Klassifizierung wurden zwei Klassen von Systemzuständen definiert, nämlich betriebsbereit und nicht betriebsbereit. Die Ergebnisse der Implementierung zeigen, dass die Art des Algorithmus eine wichtige Rolle spielt. Aus dem Fehlerratenvergleich der einzelnen Algorithmen folgt, dass die überwachten Algorithmen den Systemzustand im Vergleich zu den unüberwachten Algorithmen genauer ermitteln. Unter den überwachten Algorithmen haben die künstliche neuronale Netz mit dem Fehlerrate von 0,1259 und Support Vector Machine mit dem Fehlerrate von 0,1339 das beste Ergebnis erzielt.

Für zukünftige Forschungen ist es möglich, die Anzahl der Klassen für die Klassifizierung des Systemzustands zu erhöhen. So könnten beispielsweise die folgenden 4 Systemzustandsklassen definiert werden: gesund, betriebsbereit, möglicher Ausfall (Gefahr) und ausgefallen. Dies ermöglicht es, potenzielle (versteckte) Fehler rechtzeitig zu erkennen und ihre weitere Entwicklung zu verhindern.

## Literaturverzeichnis

[ACATECH 2013] Acatech – Deutsche Akademie der Technikwissenschaften e.V.: Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0 - Abschlussbericht des Arbeitskreises Industrie 4.0, Promotorengruppe Kommunikation der Forschungsunion Wirtschaft – Wissenschaft, 2013.

[Aggarwal 2016] Aggarwal, C.: Outlier Analysis, 2. Aufl. Springer & Incorporated, 2016.

[Ansari et al. 2018] Ansari, F.; Hold, P.; Sihm, W.: Human-Centered Cyber Physical Production System: How Does Industry 4.0 Impact on Decision-Making Tasks? IEEE Technology and Engineering Management Society Conference, 27 June-1 July, IL, USA, 2018.

[Ayodele 2010] Ayodele, T.O. : Types of Machine Learning Algorithms. In New Advances in Machine Learning; Zhang, Y., Ed.; IntechOpen: London, UK, 2010.

[Bauernhansl et al. 2014] Bauernhansl, T.; Hompel, M.; Vogel-Heuser, B.: Industrie 4.0 in Produktion, Automatisierung und Logistik, Wiesbaden, Springer Vieweg, 2014. ISBN 978-3-658-04681-1

[Bankhofer et al. 2008] Bankhofer, U.; Vogel, J.: Datenanalyse und Statistik. Eine Einführung für Ökonomen im Bachelor, Betriebswirtschaftlicher Verlag Dr. Th. Gabler, GWV Fachverlage GmbH, Wiesbaden, 1. Auflage 2008.

[Berger 2015] Berger, R.: Die digitale Transformation der Industrie. Was sie bedeutet. Wer gewinnt. Was jetzt zu tun ist. 1. Auflage, Bundesverband der Deutschen Industrie e.V. (BDI), 2015.

[BITKOM 2012] BITKOM: Big Data im Praxiseinsatz – Szenarien, Beispiele, Effekte. Berlin, 2012.

[BITKOM 2014] BITKOM, Industrie 4.0 -Volkswirtschaftliches Potenzial für Deutschland, Berlin, 2014.

[Breiman et al. 1984] Breiman, L.; Friedman, J. H.; Olshen, R. A; Stone, C. J.: Classification and Regression Trees. Chapman & Hall/CRC, 1984.

[Broy 2010] Broy, M.: Cyber-physical Systems Innovation durch Software intensive eingebettete Systeme, Springer, 2010.

[Brühl 2019] Brühl, V.: Big Data, Data Mining, Machine Learning und Predictive Analytics: Ein konzeptioneller Überblick, CFS Working Paper Series, No. 617, Goethe University Frankfurt, Center for Financial Studies (CFS), Frankfurt a. M., 2019.

[Catelani et al. 2018] Catelani, M.; Ciani, L.; Venzi, M.J.: RBD Model-Based Approach for Reliability Assessment in Complex Systems, S. 1-9. 2018.

[Chakraborty et al. 1989] Chakraborty, S.; Ankiyah, B.: Assessment of Manufacturing System Reliability: A Case Study, The Journal of the Operational Research Society, S. 55-63, 1989.

[Columbus 2016] Columbus, L.: Industrial Analytics Based On Internet Of Things Will Revolutionize Manufacturing, in, Forbes, 2016.

[DIN 31051 2019] DIN EN 31051:2019-06. Grundlagen der Instandhaltung. Beuth Verlag, Berlin

[Freiknecht et al. 2018] Freiknecht, J.; Papp, S.: Big Data in der Praxis: Lösungen mit Hadoop, Spark, HBase und Hive. Daten speichern, aufbereiten, visualisieren, München, 2018.

[Geisberger et al. 2012] Geisberger, E. ; Broy, M.: Agenda CPS Integrierte Forschungsagenda Cyber-Physical Systems, 2012.

[Gesus 2006] Gesus: Grundlagen der Zuverlässigkeitstechnik Abgerufen am 29. November 2021, von <http://www.taurus-lightning.de/UNI/S6/GZT.pdf>

[Gölzer et al. 2015] Gölzer, P.; Cato, P.; Amberg, M.: Data Processing Requirements of Industry 4.0 – Use Cases for Big Data, S. 1–13, in: European Conference on Information Systems ECIS, 2015.

[Greene et al. 2008] Greene, D.; Cunningham, P.; Mayer, R.: Unsupervised Learning and Clustering. In Machine Learning Techniques for Multimedia: Case Studies on Organization and Retrieval; Cunningham, P., Cord, M., Eds.; Springer: Berlin/Heidelberg, Germany, 2008.

[Hodapp 2009] Hodapp, W.: Die Bedeutung einer zustandsorientierten Instandhaltung: Einsatz und Nutzen in der Investitionsgüterindustrie. In: REICHEL, J., G. MÜLLER und J. MANDEL-ARTZ (Herausgeber): Betriebliche Instandhaltung, Seiten 135–149. Springer, Berlin, 2009.

[IDC 2018] IDC: Data Age 2025 - The Digitization of the World, Framingham, 2018.

[INTEL 2012] Intel, Big Data 101: Unstructured Data Analytics, Santa Clara, 2012.

[Jardine et al. 2006] Jardine, A. K.S.; Lin D.; Banjevic, D.: A review on machinery diagnostics and prognostics implementing condition-based maintenance. Mechanical Systems and Signal Processing, 20(7):1483–1510, 2006.

[Kalisch et al. 2021] Kalisch, M.; Meier, L.: Logistische Regression. Eine anwendungsorientierte Einführung mit R. Springer Spektrum, 2021.

[Kass 1980] Kass, G. V.: An exploratory technique for investigating large quantities of categorical data. *Applied Statistics*, 29:119–127, 1980.

[Kudelina et al. 2021] Kudelina, K.; Vaimann, T.; Asad, B.; Rassölkin, A.; Kallaste, A.; Demidova, G.: Trends and Challenges in Intelligent Condition Monitoring of Electrical Machines Using Machine Learning. *Appl. Sci.* 2021.

[Laney 2001] Laney D.: *3D Data Management: Controlling Data Volume, Velocity and Variety*, Stamford, 2001.

[Lawrynowicz 2014] Lawrynowicz, A.; Tresp, V.: Introducing machine learning. In *Perspectives on Ontology Learning*; Lehmann, J., Voelker, J., Eds.; IOS Press: Heidelberg, Germany, 2014.

[Lee 2008] Lee, E. A.: Cyber Physical Systems: Design Challenges, in: *IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC)*, S. 363–369, 2008.

[Leidinger 2014] Leidinger, B.: *Wertorientierte Instandhaltung: Kosten senken, Verfügbarkeit erhalten*. Springer Gabler, Wiesbaden, 2014.

[Matyas 2002] Matyas, K.: Ganzheitliche Optimierung durch individuelle Instandhaltungsstrategien. In: *Industrie Management*, 2002, Nr. 2, S. 13 – 16. – ISSN 1434–1980.

[Michau et al. 2019] Michau, G.; Fink, O.: Unsupervised Fault Detection in Varying Operating Conditions. In *Proceedings of the 2019 IEEE International Conference on Prognostics and Health Management (ICPHM)*, San Francisco, CA, USA, 2019.

[Miriayala 1989] Miriyala, K.; Viswanadham, N.: Reliability analysis of flexible manufacturing systems, *International Journal of Flexible Manufacturing Systems*, 2, S. 145-162, 1989.

[Mobley 2002] Mobley, R. K.: *An Introduction to Predictive Maintenance*. Burlington, MA, USA: Elsevier Butterworth-Heinemann, 2002.

[Moubray 1996] Moubray, J.: *RCM - die hohe Schule der Zuverlässigkeit von Produkten und Systemen*. 1. Auflage, Landsberg, Deutschland: im Verlag Moderne Industrie, 1996. – ISBN 978–3–478–91630–5.

[Obermaier 2019] Obermaier, R.: *Handbuch Industrie 4.0 und Digitale Transformation*, Springer Fachmedien Wiesbaden GmbH, ein Teil von Springer Nature, 2019.

[Paluszek et al. 2019] Paluszek M.; Thomas, S.: *MATLAB machine learning recipes: A problemsolution approach*, Second edition. 2019.

[Pan 1999] Pan, J.: Software reliability, Dependable Embedded Systems, Carnegie Mellon University, 1999.

[Panesar 2019] Panesar, A.: Machine Learning and AI for Healthcare. 2019.

[Pawellek 2016] Pawellek, G.: Integrierte Instandhaltung und Ersatzteillogistik. Springer Berlin, Berlin, 2. Auflage, 2016.

[Plattform I4.0 2015] Plattform I4.0, Umsetzungsstrategie Industrie 4.0 - Ergebnisbericht der Plattform Industrie 4.0, Berlin, 2015.

[Quinlan 1993] Quinlan, J. R.: C4.5 Programs for Machine Learning. Morgan Kaufmann, 1993.

[Rebala et al. 2019] Rebala, G.; Ravi, A.; Churiwala, S.: An Introduction to Machine Learning. Springer, 2019.

[Saxena et al. 2008] Saxena, A.; Goebel, K.; Simon, D.; Eklund, N.: Damage propagation modeling for aircraft engine run-to-failure simulation, in Proc. 1st Int. Conf. Prognostics Health Manage., Denver, CO, USA, 2008.

[Schacht et al. 2019] Schacht, S.; Lanquillon, C.: Blockchain und maschinelles Lernen. Wie das maschinelle Lernen und die Distributed-Ledger-Technologie voneinander profitieren, Springer Vieweg, 2019.

[Schenk 2010] Schenk, M.: Instandhaltung technischer Systeme. Methoden und Werkzeuge zur Gewährleistung eines sicheren und wirtschaftlichen Anlagenbetriebs. Springer Heidelberg, 2010.

[Schröder 2010] Schröder, W.: Ganzheitliches Instandhaltungsmanagement: Aufbau, Ausgestaltung und Bewertung. Gabler Verlag, Wiesbaden, 2010.

[Schüller et al. 2015] Schüller, A.; Epple, U.: Ein Referenzmodell zur Prozedurbeschreibung - Eine Basis für Industrie 4.0- at - Automatisierungstechnik. Band 63, Heft 2, Seiten 87–98, 2015, ISSN (Online) 2196-677X, ISSN (Print) 0178-2312, DOI: 10.1515/auto-2014-1152.

[Seraphin 1994] Seraphin, M.: Neuronale Netze und Fuzzy-Logik: Verknüpfung der Verfahren, Anwendungen, Vor- und Nachteile, Simulationsprogramm. 1. Auflage, München, Deutschland: Franzis-Verlag GmbH, 1994. ISBN 3-7723-6385-7.

[Straube 1988] Straube, F.: Kriterien zur Planung und Realisierung von Instandhaltungskonzepten in logistikorientierten Unternehmen. 1. Auflage, München, Deutschland: Huss-Verlag GmbH, 1988. – ISBN 3-921455-47-4.

[Thereneau et al. 2015] Thereneau, T.; Clinic, M.: User written splitting functions for RPART, 2015.

[Waeyenbergh et al. 2002] Waeyenbergh, G.; Pintelon, L.: A framework for maintenance concept development. *International Journal of Production Economics* 77 (3): S. 299-313, 2002.

[Waltl 2019] Waltl, D.: Erklärbarkeit und Transparenz im Machine Learning. In: Mainzer K. (eds) *Philosophisches Handbuch Künstliche Intelligenz*. Springer Reference Geisteswissenschaften. Springer VS, Wiesbaden, 2019

[Weber 2020] Weber, F.: *Künstliche Intelligenz für Business Analytics. Algorithmen, Plattformen und Anwendungsszenarien*, Springer Vieweg, 2020.

[Wickham et al. 2018] Wickham, H.; Grolemund, G.: *R für Data Science. Daten importieren, bereinigen, umformen, modellieren und visualisieren*. 1. Auflage, 2018.

[Wischmann et al. 2015] Wischmann, S.; Wangler, L.; Botthof, A.: *Begleitforschung zum Technologieprogramm AUTONOMIK für Industrie 4.0* 35, 2015

[Xia et al. 2018] Xia, T.; Dong, Y.; Xiao, L.; Du, S.; Pan, E.; Xi, L.; J.R.E.; Safety, S.: *Recent advances in prognostics and health management for advanced manufacturing paradigms*, 2018.