STUTTGARTER BEITRÄGE ZUR PRODUKTIONSFORSCHUNG

Markus Böhm

»Datenbasierte Bewertung kundeninduzierter Turbulenz in der volatilen Variantenfertigung«

Fraunhofer IPA

Universität Stuttgart

STUTTGARTER BEITRÄGE ZUR PRODUKTIONSFORSCHUNG BAND 194

Markus Böhm

»Datenbasierte Bewertung kundeninduzierter Turbulenz in der volatilen Variantenfertigung«

Herausgeber

Univ.-Prof. Dr.-Ing. Thomas Bauernhansl[1,2]
Univ.-Prof. Dr.-Ing. Dipl.-Kfm. Alexander Sauer[1,3]
Univ.-Prof. Dr.-Ing. Kai Peter Birke[4]
Univ.-Prof. Dr.-Ing. Marco Huber[1,2]

[1]Fraunhofer-Institut für Produktionstechnik und Automatisierung IPA, Stuttgart
[2]Institut für Industrielle Fertigung und Fabrikbetrieb (IFF) der Universität Stuttgart
[3]Institut für Energieeffizienz in der Produktion (EEP) der Universität Stuttgart
[4]Institut für Photovoltaik (*ipv*) der Universität Stuttgart

Kontaktadresse:
Fraunhofer-Institut für Produktionstechnik und Automatisierung IPA
Nobelstr. 12
70569 Stuttgart
Telefon 0711 970-1101
info@ipa.fraunhofer.de
www.ipa.fraunhofer.de

Bibliographische Information der Deutschen Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliographie; detaillierte bibliografische Daten sind im Internet über http://dnb.de abrufbar.

Zugl.: Stuttgart, Univ., Diss., 2025

D 93

2025

Druck und Weiterverarbeitung:
Fraunhofer-Druckerei, Stuttgart, 2025
Für den Druck des Buches wurde chlor- und säurefreies Papier verwendet.

Datenbasierte Bewertung kundeninduzierter Turbulenz in der volatilen Variantenfertigung

Von der Fakultät für
Konstruktions-, Produktions- und Fahrzeugtechnik
der Universität Stuttgart
zur Erlangung der Würde eines Doktor-Ingenieurs (Dr.-Ing.)
genehmigte Abhandlung

Vorgelegt von

Markus Böhm

aus Stetten im Unterallgäu

Hauptberichter:	Prof. Dr.-Ing. Thomas Bauernhansl
Mitberichterin:	Prof. Dr. rer. nat. Sabina Jeschke
Tag der mündlichen Prüfung:	18. März 2025

Institut für Industrielle Fertigung und Fabrikbetrieb (IFF)
der Universität Stuttgart

2025

Vorwort des Autors

Das vorliegende Werk ist das Ergebnis meiner Zeit als wissenschaftlicher Mitarbeiter am Fraunhofer Institut für Produktionstechnik und Automatisierung (IPA) und am Institut für Industrielle Fertigung und Fabrikbetrieb (IFF) der Universität Stuttgart.

Mein Dank gilt meinem Doktorvater Herrn Prof. Dr. Bauernhansl. Seine inspirierende Wirkung in den gemeinsamen fachlichen Diskussionen gepaart mit der bejahenden Offenheit für meine Ideen waren für mich Antrieb, Herausforderung und Motivation zugleich. Weiter danke ich meiner Mitberichterin Frau Prof. Dr. Sabina Jeschke für den akademischen Perspektivwechsel und die immer anspruchsvollen inhaltlichen Impulse. Darüber hinaus danke ich meinen Vorgesetzten und Kollegen Dr. Klaus Erlach und Dr. habil Hans-Hermann Wiendahl für die intensiven Gespräche zu meinem Thema auf höchstem akademischen Niveau mit Anspruch und Witz. Herrn Stefan Emminger, Geschäftsführer der Firma phg, danke ich für seine volle Unterstützung bei der Evaluierung meiner Ergebnisse.

Meine Tätigkeit prägten früh Thomas Schrodi und Matthias Dillmann durch ein Umfeld mit Spielraum und Verantwortung. An Roman, Felix, Markus, Silke, Su, Timo, DocWö, Chris, David, Axel, Lisa, Christian, Hans, Sarah, Marc-André, Tim und viele andere: Danke euch für die irre Zeit! Doro, Mark, Rebecca, Seyoum, Birgit, Christine, Elena, Barbara, Miriam und Silke: Ohne eure Geduld und Hilfe wären so viele Dinge gar nicht möglich.

Öbs, Roman, Moddin, Milex, Rapha, Janusch, Chris, Vroni, Shirin und Philipp, gut das ich euch um mich habe! Meinen Schwestern Eva, Elke und Anna danke ich dafür, dass wir – obwohl geografisch so weit voneinander entfernt – immer unbesiegbar zusammenhalten all die Zeit. Bei meiner Gefährtin Christin reichen keine Worte. Du trägst mich und machst mich zu dem, der ich sein möchte.

Diese Arbeit widme ich meinen Eltern Christa und Werner Böhm. Sie schenken mir alle Möglichkeiten und Freiheiten, indem sie mich von klein auf bis heute zu den richtigen Zeitpunkten als Erwachsenen behandeln oder mich als ihr Kind umsorgen.

Stuttgart, im März 2025 Markus Böhm

Kurzfassung

»Man muss die Welt nicht verstehen,
man muss sich nur darin zurechtfinden.«
Albert Einstein

Die industrielle Fertigung unterliegt in ihrer Historie dem vielbeschworenen Wandel. Neu ist die bislang unerreichte Geschwindigkeit, mit der dieser vollzogen wird. Um effizient und flexibel fertigen zu können, sind heutige Produktionssysteme geprägt von hoher Dynamik. Direkte Folge daraus ist ein Mangel an Transparenz des Systemzustands, was Verhaltensprognosen zusätzlich erschwert. Wünschenswert sind Werkzeuge zur Kennzahlenprognose, um auf drohende Turbulenzen frühzeitig reagieren zu können. Produktionsdaten liegen dafür umfänglich vor. Doch können wir Systeme dafür ausreichend genau modellieren? Und nutzen wir die vorhandenen Daten zielgerichtet? Methoden des maschinellen Lernens können ein Ansatz sein, um Turbulenzen prognostizierend zu bewerten.

Maschinelles Lernen wird vorwiegend auf statische Systeme angewendet. In solchen Systemen behalten angereicherte Datensätze über einen langen Zeitraum ihre Gültigkeit. Im Falle dynamischer Systeme gilt dies nicht und so sind Prognosen mit Hilfe herkömmlicher Produktionsmodelle kaum mehr möglich. Diese Arbeit verfolgt das Ziel, Datensätze dynamischer Produktionssysteme für die Turbulenzprognose nutzbar zu machen.

Das hier vorgeschlagene Vorgehen nutzt Entscheidungsbäume zur Turbulenzprognose. Hauptschwierigkeiten bei zuverlässigen Prognosen sind neben der Systemdynamik die zeitliche Latenz zwischen Ursache und Wirkung. Diesen soll mit Werkzeugen der Mathematik rechnergestützt begegnet werden. Damit trägt die Arbeit zum Verständnis für den Einsatz des maschinellen Lernens in der Produktion bei und kann später in der Anwendung bei Entscheidungsprozessen unterstützen. Validiert wird die Arbeit sowohl mit synthetischen Daten als auch mit einem realen Industriedatensatz.

Schlagworte: Turbulenz, Kennzahlenprognose, Systemdynamik, Maschinelles Lernen, Entscheidungsbäume

Abstract

»You don't have to understand the world.
You just have to find your own way around in it.«
Albert Einstein

Throughout history, industrial manufacturing has been subject to the frequently mentioned "change". The new aspect is the unprecedented speed in which change is taking place. In order to manufacture efficiently and flexibly, today's main attribute of production systems is high dynamics. A direct consequence of this is the lack of the system state's transparency which makes behavior predictions even more difficult. Forecasting tools are desirable to have the chance to react to impending turbulence at an early stage. Today, production data is easily available for this purpose. But can we model manufacturing systems with sufficient accuracy? Do we use the available data in a targeted manner? Machine learning methods are one approach to assess turbulence prediction.

Machine learning is mainly applied to static systems. In these cases, enriched data sets retain their validity over a long period of time. For dynamic systems, however, this is no longer feasible, and forecasting with the help of conventional production models is hardly possible anymore. This work pursues the goal of making data sets of dynamic production systems usable for turbulence forecasting.

The proposed approach uses decision trees for turbulence forecasting. Besides the system dynamics, time latencies between cause and effect represent the core problems. These are to be countered with mathematic computational tools. Thus, the work contributes to the understanding of the use of machine learning in production and supports the application in decision-making processes. The work is validated with synthetic data and with a real industrial dataset.

Keywords: turbulence, KPI prediction, system dynamics, machine learning, decision trees

Inhaltsverzeichnis

Abkürzungsverzeichnis

ASUM-DM . . Analytics Solutions Unified Method for Data Mining/Predictive Analytics

CART . . . . . Classification and Regression Trees

CBR . . . . . Case-based Reasoning

CEP . . . . . Complex Event Processing

CRISP-DM . . . . . Cross Industry Standard Process for Data Mining

DLZ . . . . . Durchlaufzeit

DMT . . . . . Data Mining Techniques

DoE . . . . . Design of Experiments

IoT . . . . . Internet of Things

KDD . . . . . Knowledge Discovery in Database

KI . . . . . Künstliche Intelligenz

kNN . . . . . Künstliche Neuronale Netze

KPI . . . . . Key Performance Indicator

LR . . . . . Logistische Regression

LTI . . . . . Linear Time-Invariant

MDI . . . . . Mean Decrease in Impurity

ML . . . . . Maschinelles Lernen

NB . . . . . Naive Bayes

NRMSE . . . . . Normalized Root Mean Squared Error

PPS . . . . . Produktionsplanung und -steuerung

SVM . . . . . Support Vector Machine

VUCA . . . . . Volatility, Uncertainty, Complexity, Ambiguity

WIP . . . . . Work in Process

XAI . . . . . Explainable Artificial Intelligence

Abbildungsverzeichnis

Tabellenverzeichnis

1 Einleitung

Mit der Finanzkrise 2008 und einigen weiteren weltweit wirksamen politischen Ereignissen erübrigten sich im letzten Jahrzehnt alle Illusionen: Das Verhalten globaler Märkte ist mit herkömmlichen Mitteln weder zu modellieren noch vorherzusagen. Allein ein Blick auf den Marktbedarf im deutschen Maschinenbau zeigt, dass relative Schwankungen im zweistelligen Bereich keine singulären Ereignisse sind (siehe Abbildung 1.1). Wir leben in einer unbeständigen, unsicheren, komplexen und mehrdeutigen Welt – immer verbreiteter als ‚VUCA' bezeichnet (Volatility, Uncertainty, Complexity und Ambiguity). Ursprünglich für Systembeschreibungen des Kalten Krieges im militärischen Kontext verwendet, wurde der Begriff auf organisatorische, wirtschaftliche und industrielle Bereiche übertragen und steht heute – wenn auch vereinfachend – für die Komplexität globaler Finanz- und Wirtschaftssysteme (Barber 1992; Mack et al. 2015, S. 5).

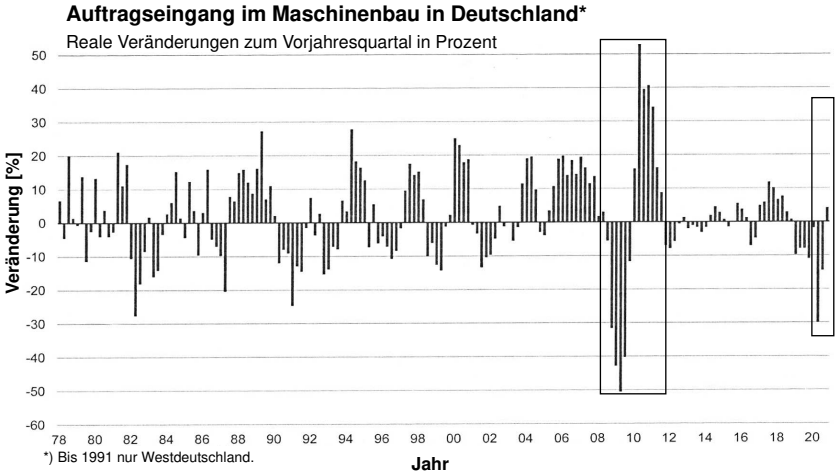


Abbildung 1.1: Marktvolatilität mit Finanz- und Coronakrise 2008 und 2020 (VDMA 2020, S. 45)

Die gute Nachricht vorweg: Der Mensch versteht in den seltensten Fällen die Systeme mit und in denen er agiert. Kaum jemand besitzt umfassende Kenntnis über das technische

System, wenn ein Auto gesteuert, eine elektronische Nachricht versendet oder Geld überwiesen wird. Es ist der Normalfall, lediglich Korrelationen und Ursache-Wirkung-Beziehungen beobachten zu können, um daraus Schlüsse zu ziehen und Handlungen abzuleiten. Mit dem technologischen Fortschritt wurden jedoch die Handlungsräume und damit die Menge an zu verarbeitenden Informationen vervielfacht. Dies führt zu weniger prognostizierbarem Verhalten, Entscheidungen werden vermehrt reaktiv denn proaktiv getroffen und Systeme somit immer stärker als *turbulent* empfunden (Wiendahl 2011, 292 f.). Eine Antwort darauf bieten uns moderne Technologien unter anderem mit Werkzeugen des maschinellen Lernens. Die Möglichkeit, große Mengen an Informationen zu verarbeiten, verhelfen uns zur Handlungsfähigkeit in komplexen Systemen. Diese Arbeit zeigt, wie mit Hilfe von Produktionsdaten Prognosen über durch Kennzahlen repräsentierte Turbulenzen in der Produktion erstellt werden können.

1.1 Ausgangssituation

Die Konsequenzen von VUCA reichen im industriellen Bereich hinein bis zum eigentlichen Ort der Wertschöpfung – der Produktion. So sehen sich die Unternehmen mit einer Vielzahl an Komplexitätstreibern konfrontiert: Kurze Lebenszyklen bei sowohl der Produkt- als auch der Produktionstechnologie, Auftragsänderungen in Art, Menge und Termin, ein immer noch fortschreitender Individualisierungstrend und kurze ‚Time-to-Money' bei steigenden Kundenansprüchen sind nur einige Faktoren, welche die interne und externe Komplexität produzierender Unternehmer immer weiter steigern (Westkämper & Löffler 2016, S. 55; Mattsson et al. 2011; Dombrowski et al. 2018, 716 f.). Gleichzeitig befindet sich die wertschöpfende Industrie in einem Zustand hoher technologischer Reife. Dies hat zur Folge, dass sich die Produktion im Bereich der physischen Arbeitsinhalte nur noch inkrementell weiterentwickelt. Die Leistungsfähigkeit konvergiert und der prognostizierte S-Kurven-Effekt stellt sich ein (Solow 1956; Netland & Ferdows 2016; Rammer 2018). So ist es heute durch hohe Automatisierung und eine global vernetzte Logistik weitgehend ortsunabhängig möglich, physische Produkte auf konkurrenzfähigem Qualitätslevel herzustellen. Differenzierung vom Wettbewerber erfolgt daher kaum mehr durch den physischen Mehrwert des Produktes. Wettbewerbsfaktoren wie Zeit, Effizienz, Varianz und Transparenz rücken ins Zentrum.

Schnelligkeit und Termintreue gelten nach wie vor als die zentralen logistischen Ziele (Schuh & Fuß 2015, S. 18). Die Ressourceneffizienz wurde um den Aspekt der Nachhaltigkeit im politischen und sozialen Raum erweitert (Miehe 2018) und sowohl höheren Varianz an Fertigungs- und Montageprozessen (Hering et al. 2013, S. 995) als auch Transparenz sind erwünschte Eigenschaften von Produktionen im Bezug auf Industrie 4.0 (Bauernhansl et al. 2016). Die Produktion trägt durch ihren großen Investitionsanteil innerhalb produzierender Unternehmen die Verantwortung zur Erreichung dieser Ziele. Doch wie können Akteure in einer komplexen Umgebung das auch in Zukunft leisten?

Wären alle Elemente und funktionalen Zusammenhänge eines Produktionssystems bekannt, so ließe es sich anhand des vollständigen Modells mathematisch optimieren. Die Systemkomplexität zeigt dem Ansatz seine Grenzen auf. Ein für die Lösung dieses Problems noch immer weitgehend ungenutztes Potenzial liegt in den Produktionsdaten als einer Säule der oben genannten Wettbewerbsfaktoren (Bauer et al. 2017; König et al. 2019; Röhrig 2020; Straubhaar 2021). Die in einer realen Produktion erhobenen Daten beinhalten implizit und explizit Informationen über Zustand und Verhalten und tragen, wenn richtig interpretiert, zur Transparenz und damit zur effizienten Nutzung der Produktionsressourcen bei. Die Ausgangslage heute ist denkbar günstig. So steigt die Anzahl der Geräte im Internet of Things (IoT) ungebrochen weiter an. Der weltweite Datenverkehr überschritt im Jahr 2021, den Vorhersagen entsprechend, die Grenze von 300 Exabytes – pro Monat (Cisco public 2018; Samulat 2017). Die IoT-Devices sind gegenwärtig im industriellen Kontext in ihrer Anzahl weniger verbreitet als solche im Privatbereich, die Zunahme hält jedoch mit dem allgemeinen Trend Schritt (Gartner 2017; Püschel et al. 2017). Zwar gilt Deutschland als ‚Datenweltmeister' (Eickelmann et al. 2015; Petersdorff 2019), da die Grenzkosten der Datenakquise gegen Null gehen und somit als Rohstoff in relevanten Mengen meist vorliegen, jedoch wird die Verwertung bisher kaum ausgeschöpft.

Unmittelbaren Einfluss auf die Produktion haben die Kundenaufträge. Ihre Attribute im Einzelnen und die Summe ihrer Wirkung im Ganzen prägen die Anforderungen an die Gesamtleistung eines Produktionssystems. Die Kundenaufträge bilden damit einen wesentlichen Komplexitätstreiber, dem das Produktionssystem mit Maßnahmen zur Erhöhung ihrer Systemrobustheit begegnet. Die negativen Konsequenzen der Komplexität, welche durch die Robustheit nicht kompensiert werden können, werden als *Turbulenz*

empfunden (Böhm & Bauernhansl 2021; Erlach et al. 2022a). Eine treffsichere Prognose und quantitative Bewertung dieser Turbulenzen *bevor* diese entstehen, würde die Ableitung von Maßnahmen und die Entwicklung von geeigneten Gegenstrategien in hohem Maße fördern. So sieht sich die Produktion fortwährend mit der Frage nach einer kontinuierlichen Bewertung der zukünftig auftretenden Turbulenzen konfrontiert.

1.2 Problemstellung

Unabhängig von der Größe eines betrachteten Systems – sei es lediglich ein verketteter Prozessabschnitt oder eine globale Supply Chain – wirken Systemkomplexität und Systemrobustheit immer gegeneinander (Malik 2015, 153 ff. Beer 2000, 270 ff.). Dabei eilt in der Regel die Komplexität in Umfang und Ausmaß der Robustheit zeitlich voraus. Um fortwährend den Abstand zwischen Komplexität und Robustheit zu halten oder besser noch zu minimieren, werden Systeme zielgerichtet angepasst (Malik 2015, S. 154). Im Kontext der Produktion sind typische, die Robustheit erhöhende Maßnahmen Kapazitätserweiterung, Technologieentwicklung, Produktanpassungen, Kundenentkopplung, Automatisierung, Korrektur der Wertschöpfungstiefe und Verfeinerung der Planung und Steuerung (Erlach 2020; Wiendahl 2011; Bauernhansl 2014b). Allein aus Gründen der Wirtschaftlichkeit ist eine vollständige Kompensation der **Komplexität** durch Robustheit praktisch nie gegeben. Aus diesem ‚Gap' zwischen Komplexität und Robustheit resultieren ständig unbeabsichtigte Verhaltensabweichungen des Systems. So führt beispielsweise eine kurzfristige Überschreitung der Systemkapazität zu erhöhten Pufferbeständen, längeren Durchlaufzeiten und schlechteren Servicegraden. Zur Dämpfung oder gar Vermeidung dieser Kennzahlenabweichung durch zielgerichtete Maßnahmen wäre eine Prognose von Art und Ausmaß der verursachten Turbulenz – somit eine **operative Turbulenzbewertung** – notwendig. Diese ist für **dynamische Systeme** bislang nicht ausgereift.

In dieser Arbeit umfasst der Begriff ‚Turbulenz' die durch Systemrobustheit nicht kompensierte, unbeabsichtigte und messbare Kennwertabweichung von einem Soll-Korridor (siehe dazu die Kapitel 1.3 zu den Forschungsfragen, 1.6 zur Arbeitshypothese und 3.2 zur Turbulenzbewertung). Die Kennzahlen lassen sich in solche mit absoluten und relativen Werten. Dies Systematik orientiert sich dabei an der Nomenklatur und Hierarchie

der Produktionskennzahlen von Erlach et al. (Erlach et al. 2022b, S. 560). Das Set an turbulenzbeschreibenden Kennzahlen setzt sich fallabhängig jeweils anders zusammen und ist entsprechend adaptierbar. Auf die Einteilung in Auftrags- und Systemkennzahlen wird in Abschnitt 5.2 eingegangen, da dies erst für die Umsetzung notwendig ist. Abbildung 1.2 zeigt ein mögliches Basisset an kundenauftragsabhängigen Turbulenzkennzahlen, anhand derer zukünftig auftretende Turbulenzen bewertet werden können.

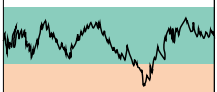
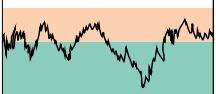
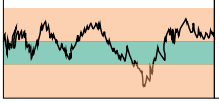
Turbulenz, wenn Toleranzkorridor der Kennzahl…	***Absolut-Kennzahlen***	***Relativ-Kennzahlen***	***Optimum (Turbulenzfreier Systemzustand) bei …***
… unterschritten wird. 	■ Ausbringung ■ Produktivität ■ …	■ Servicegrad ■ Gutausbeute ■ Qualitätsgrad ■ …	■ Maximierung / 100%
… überschritten wird. 	■ Durchlaufzeit ■ Verschleiß ■ Energieverbrauch ■ …	■ Ausschussanteil ■ Nacharbeitsanteil ■ Krankenstand ■ …	■ Minimierung / 0%
… verlassen wird. 	■ Bestände ■ Kapazitätsangebot ■ …	■ Auslastung ■ OEE ■ Zieltermineinhaltung ■ …	■ Zielwerteinhaltung (Optimaler Betriebspunkt)

Abbildung 1.2: Basis-Set an Turbulenzkennzahlen

Somit kann – beispielhaft – prognostiziert werden, ob die kurzfristige Beauftragung über ein erst kürzlich eingeführtes Produkt beim gegenwärtigen Kapazitätsangebot, der aktuellen Auslastung und den Lagerbeständen einen kritischen Abfall der Produktivität, Erhöhung des Krankenstandes und ungewollte Zielterminabweichungen zur Folge haben. Für die Turbulenzbewertung wird ein auftrags- und systembeschreibender Datensatz verwendet, daraus ein Modell erstellt und damit die Prognose durchgeführt. Die Problemstellung dieser Arbeit ergibt sich aus den oben hergeleiteten und nachfolgend beschriebenen drei Aspekten.

Komplexität

Unternehmen sind aufgrund der auf sie wirkenden äußeren und inneren Komplexität nicht in der Lage, ein für sie gültiges System vollständig zu modellieren, um Verhaltensvoraussagen treffen zu können (Beer 2000; Bauernhansl 2014b, S. 14; Böhm & Bauernhansl

2021, S. 687). Die Komplexität zieht die Untauglichkeit analytischer Ansätze zur Prognose unmittelbar nach sich. Turbulenzen, welche innerhalb komplexer Systeme nicht vermieden werden können, sind daher bisher kaum zu prognostizieren und quantitativ zu bewerten (Wiendahl 2007; Bauernhansl 2014b). Die bestehende Unsicherheit über Ort und Ausmaß zukünftiger in der Produktion auftretender Turbulenzen hat daher entweder eine schlechtere Systemgesamtleistung zur Folge oder erfordert zu deren Vermeidung eine unverhältnismäßig erhöhte Systemrobustheit. Beides ist kostenintensiv und würde umgekehrt bei Beherrschung einen Wettbewerbsvorteil bedeuten.

Operative Turbulenzbewertung

Bislang existieren kaum datenbasierte Vorgehen zur Turbulenzprognose und deren quantitativer Bewertung, was u.a. von H.-H. Wiendahl kritisiert wird (Wiendahl 2007). So wird bei Bewertungs- und Prognoseansätzen nur unzureichend zwischen Komplexität und Turbulenz unterschieden. Zudem beschränken sich die Ansätze weitgehend auf die Identifikation von Turbulenzquellen und schließen die Bewertung aus (Gille & Zwißler 2011). Neben der Turbulenzbewertung herrscht bereits in der Komplexitätsbewertung ein traditionelles Übermaß an lediglich qualitativen Ansätzen (Fricker 1996, S. 74). Die zur Absicherung unternehmerischer Entscheidungen verwendeten und stärker verbreiteten ‚Kennzahlensysteme' sind bis heute unternehmensindividuell aufgebaut und weisen somit geringe Allgemeingültigkeit auf (Preißler 2008, S. 3). Weiter vernachlässigen diese Ansätze die systemischen Abhängigkeiten zwischen den Kennzahlen untereinander und beschränken sich stark auf mittelwertbasierte betriebswirtschaftliche Kennzahlen. Es fehlt die Beschreibung, welche Datenmengen mindestens notwendig und wie diese zu verwenden sind. Ansätze zur Verwendung von maschinellem Lernen für die Turbulenzbewertung in Produktionen sind nicht beschrieben. Für qualitative Bewertungen existieren Vorschläge zur Darstellung des Bewertungsergebnisses (Nyhuis & Wiendahl 2012, 37 ff.), für quantitative Bewertungen liegen solche nicht vor (näher beschrieben in Abschnitt 3.2).

Dynamische Systeme

Um Bewertungen und Prognosen in dynamischen Systemen vorzunehmen, sind die wissenschaftlichen Vorarbeiten vor allem aus der Chaostheorie und der Kybernetik heranzuziehen

(Fricker 1996, S. 40, 88). Die Verwendung großer Mengen an Produktionsdaten ist in den frühen Arbeiten nur prinzipiell und nicht in der Anwendung konzipiert. Das ändert sich nun mit den Möglichkeiten des maschinellen Lernens (Monostori 2002; Bishop 2009; Jeschke 2015; Dangeti 2017). Prognosen zu einzelnen Kenngrößen für ausgewählte Produktionsabschnitte sind entwickelt und werden mit Predictive Data Analytics durchgeführt (Lingitz et al. 2018). Diese Ansätze blenden jedoch dynamische Systeme aus, was sie damit zeitlich und örtlich begrenzt nur einsatzfähig macht. Somit gibt es zur quantitativen Prognose kundeninduzierter Turbulenzen bisher kein geeignetes Vorgehen im Bereich des maschinellen Lernens, welches Produktionsdaten allgemeingültig auswertet und dafür nutzbar macht. Das menschliche Gehirn und zugehörige neurologische Messmethoden eignen sich als Modell für die datenbasierte Untersuchung eines Unternehmens. Ein solches kann ebenfalls in Areale eingeteilt werden und erhobene Kennzahlen und Zeitreihen – in der Abbildung die skizzierten Werteverläufe – werden zum Gegenstand von Auswertungen über Trends, Anomalien und Korrelationen. Im Sinne von Beer fehlt ein Ansatz, der an die Produktion eine Art Pulsmesser oder ein anderes Messgerät anlegt, um eine möglichst genaue Diagnose zu stellen (siehe Abbildung 1.3).

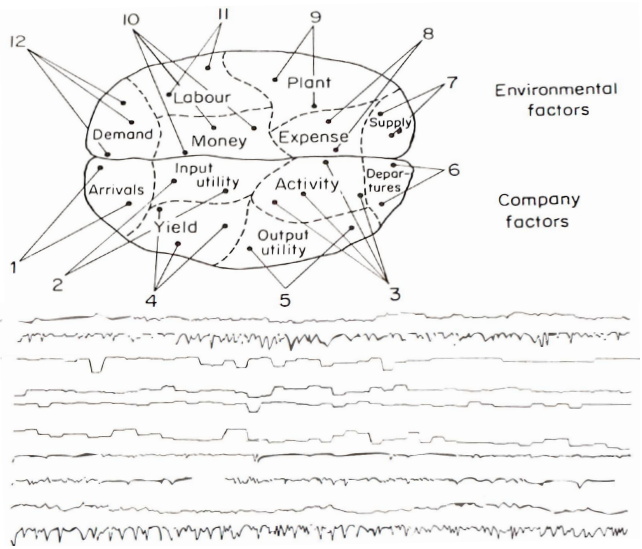


Abbildung 1.3: Enzephalografie der Produktion (Beer 2000, S. 139)

1.3 Zielsetzung und Forschungsfragen

Diese Arbeit hat die Zielsetzung, ein für die Anwendung geeignetes, datengetriebenes Vorgehen zu entwickeln und zu evaluieren, welches Kennzahlenabweichungen von einem Toleranzbereich prognostiziert und quantitativ bewertbar macht. Solche Abweichungen werden im Kontext dieser Arbeit *Turbulenzen* genannt. Die Kombination von bestehenden Werkzeugen aus den Bereichen des maschinellen Lernens und der Mathematik der Signalverarbeitung soll es ermöglichen, das implizit in den historischen Produktionsdaten vorliegende Wissen in Kennzahlenprognosen zu transformieren, welche dann Aussagen über Produktionsturbulenzen in der nahen Zukunft zulassen.

Die von Töpfer et al. vorgeschlagenen drei Ziele im Erkenntnisprozess sind deskriptiver, theoretischer und pragmatischer Art (Töpfer 2010, 52 ff. Chmielewicz 1994, 8 ff. Schweitzer 1978, 2 ff.). Kubicek beschreibt das pragmatische oder auch „technologische" Wissenschaftsziel als „[...] das wichtigste Ziel der betriebswirtschaftlichen Forschung. Es besteht in der Gewinnung von über Einzelfälle hinausgehende Aussagen zur Lösung von Entscheidungsproblemen" (Kubicek 1977, S. 5). Die drei Ziele können für die Inhalte dieser Arbeit so angegeben werden:

- Deskriptives Ziel: Darstellung von zu erwartenden Kennzahlabweichungen eines Produktionsauftrags als Turbulenzsteckbrief
- Theoretisches Ziel: Erstellung einer möglichst wertigen Informationsgrundlage zur Kennzahlenprognose für dynamische Produktionssysteme
- Pragmatisches Ziel: Allgemeingültiges Vorgehen zur Turbulenzbewertung für Produktionsaufträge

Alle Betrachtungen dieser Arbeit bewegen sich grundsätzlich im Kontext der industriellen Produktion. **Untersuchungsgegenstände** sind damit zum einen die Produktionsdaten an sich und zum anderen die Werkzeuge, mit denen diese Daten verarbeitet, Informationen extrahiert und Prognosen generiert werden können.

Zweck des Vorgehens ist folglich die Nutzbarmachung von Produktionsdaten zur prognostizierten Turbulenzbewertung von Produktionssystemen. So generiertes Prognosewissen über zukünftige Zustände der Produktion kann vielfältig zur Entscheidungsunterstützung

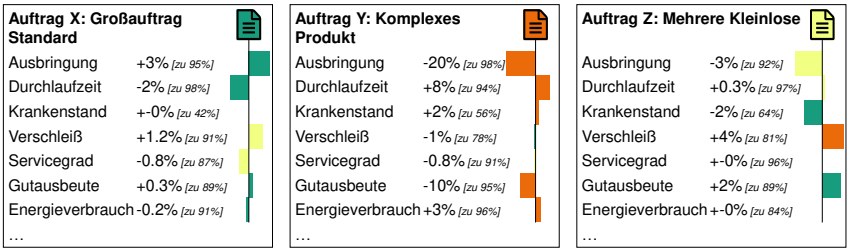


Abbildung 1.4: Beispiele für Turbulenzsteckbriefe von Aufträgen hoher, geringer und mittlerer Turbulenz (von links nach rechts)

oder für den autonomen Betrieb verwendet werden. In der Anwendung erstellt das in dieser Arbeit entwickelte Vorgehen sogenannte Turbulenzsteckbriefe (siehe Abbildung 1.4) für Kundenaufträge noch vor ihrem Produktionsstart. Dies schafft Transparenz und ermöglicht frühzeitiges Nachsteuern. Das so generierte Wissen kann Grundlage sein für strategische Entscheidungen kapazitiver (planerisch) und monetärer (Kosten) Art, steigert so kurzfristig den Kundenservice und trägt langfristig zur Wettbewerbsfähigkeit bei.

Zielgruppen dieser Arbeit sind für den theoretischen Zweck Forschende im Bereich des Process Minings und des maschinellen Lernens im Kontext der industriellen Forschung. Die behandelten Themen sind disziplinübergreifend und berühren die wissenschaftliche Community in den Feldern Data Mining, Signalverarbeitung, Systemtheorie und maschinelles Lernen. Auf Seiten der Industrie werden alle Anwender von der Praxis auf dem Shopfloor bis zur Führungsebene in mittleren und großen Unternehmen adressiert, welche sich mit der Nutzbarmachung von Daten für die Produktion befassen. Die oben beschriebene Zielsetzung der Arbeit adressiert die drei Dimensionen der Anwendung, der Gestaltung und des Ergebnisses in Anlehnung an Bauernhansl, welche Tabelle 1.1 zeigt (Bauernhansl 2003, S. 10).

Tabelle 1.1: Zielsetzung der Arbeit

Anwendungsdimension	- Diskrete, volatile Variantenfertigung - Komplexe Produktionssysteme - Transformation von Daten zu Wissen - Reaktives und proaktives Handeln - Prognose und Prävention - Autonome Entscheidungsfindung
Gestaltungsdimension	- Ausweitung der Nutzung von Daten als Informationsquelle - Standardisierung der Turbulenzbewertung - Vereinfachung des Prognoseprozesses für Kennzahlen - Quantifizierung von Produktionsturbulenzen
Ergebnisdimension	- Neutrale, quantitative Turbulenzbewertung - Detektion kontra-intuitiver Ursache-Wirkungs-Beziehungen - Praktikable Darstellung als Turbulenzsteckbriefe

Im Sinne des sokratischen Wissens über das Nichtwissen (vgl. (Treml 1994; Platon 1988, 308 f. Bd.IV)), also dem Bewusstsein über eine allgemeine Kenntnislücke, leitet sich aus Zielsetzung und den Bedarfen der Zielgruppen folgende **zentrale Forschungsfrage** ab:

»Wie kann eine datenbasierte Bewertung kundeninduzierter Turbulenzen in der Produktion ex ante erfolgen?«

Die im Kern auf die Neuheit abzielende und eher im Bereich der Theorie anzusiedelnde zentrale Forschungsfrage wirft implizit weitere Themen auf, welche stärker die Praxis der Umsetzung betreffen. So stellt sich die Frage nach dem Umfang an zu verarbeitenden Daten und damit nach unteren und oberen Limits. So ist zum einen eine Mindestanzahl an Datenpunkten notwendig, um valide Ergebnisse zu erhalten. Zum anderen steigt mit wachsender Datenmenge der Informationsgewinn nicht zwingend weiter an, was auf Sättigungsverhalten hinweist. Zudem sind für beide Punkte Abhängigkeiten vom Anwendungsfall zu erwarten. Die zentrale Forschungsfrage betrachtet die Produktion als System, was die Frage nach der Systemdynamik aufwirft. So entsprechen rein statische Systeme erstens nicht der

Realität und wären forscherisch trivial. Aussagen über völlig regellos dynamische Systeme sind ebenfalls wissenschaftlich nicht praktikabel. Es stellt sich einerseits die Frage, welches Maß an Systemdynamik gehandhabt werden kann und andererseits, welchen Anwendungs- und Gültigkeitsbereich ein solches Vorgehen hat. Am Ende steht die kritische Frage zu Aufwand und Ertrag, welche darüber entscheidet ob, das erarbeitete Vorgehen Einzug in die praktische Anwendung halten wird und sich langfristig etablieren kann. Aus der Beantwortung der zentralen Forschungsfrage ergeben sich daher zusammengefasst diese Unterfragen:

- Welches Datenvolumen ist für Turbulenzbewertung nötig?
- Wie viel Dynamik im Produktionssystem erlaubt das Vorgehen?
- Wie allgemein kann der Gültigkeitsbereich des Vorgehens gesetzt werden?
- In welchem Verhältnis steht der nötige Aufwand zum Mehrwert?

Die Beantwortung dieser Fragen trägt bei, die Welt – und in diesem Fall speziell die Welt der industriellen Produktion – mit theoretischem Wissen besser zu verstehen und in der praktischen Anwendung klüger in ihr agieren zu können. Da diese Forschungsfragen in ihrem Wesen *forschungsleitend* sind (siehe (Töpfer 2010, S. 155)), prägen sie wesentlich das Vorgehenskonzept mit. Das wissenschaftliche Vorgehen zur Beantwortung dieser Fragen wird im folgenden Abschnitt beschrieben.

1.4 Wissenschaftstheoretischer Hintergrund

»Distanz zum Gegenstand ist die Grundlage jeder Wissenschaft.«
(Gleitsmann-Topp et al. 2009, S. 16)

Zur wissenschaftstheoretischen Verortung orientiert sich der Autor an Ulrich und Hill mit den Aspekten des **Entdeckungs-**, **Begründungs-** und **Verwendungszusammenhangs** (Ulrich & Hill 1976, 304 f.) und betont entsprechend die „Unabgeschlossenheit" des Wissenschaftsbegriffs. Demnach ist die „Wissenschaft [...] kein analytisch ein für allemal definierbares Phänomen, sondern eine gesellschaftliche Erscheinung in jeweils einem

bestimmten soziokulturellen Zusammenhang". So sehen sich allein die sogenannten anerkannten Wissensquellen – landläufig als »Tatsachen« bezeichnet – einem ständigen kulturellen Wandel unterworfen und entwickelten sich von Tradition, Religion, Philosophie über erfahrbare Sinneseindrücke und wiederholbare Beobachtungen hin zu Messungen, Berechnungen und Statistiken (Chalmers & Bergemann 2007, S. 5–18). Die darauf beruhende offensichtliche Fehlbarkeit von wissenschaftlichen »Tatsachen« werden von Chalmers mit »Sehen heißt glauben« zusammengefasst, womit er die Schwachstelle des positivistischen und empiristischen Wissenschaftsverständnisses von beispielsweise Carnap (Gadenne 1999, S. 9) aufzeigt. Dennoch sind – wie auch in dieser Arbeit – Beobachtungen die unverzichtbare Grundlage wissenschaftlichen Arbeitens und stellen den Ausgangspunkt für den mit auf Bacon zurückgehenden *Induktivismus* (Gadenne 1999, S. 35) dar, welcher von beobachteten Tatsachen und Ereignissen auf allgemeine Aussagen, Gesetze und Theorien schließt. Von diesen wiederum können mittels *Deduktion* erklärend oder prognostisch Aussagen über weitere Fälle gemacht werden. Die Deduktion allerdings leidet an einer nie vermeidbaren Unvollkommenheit, da sie wahre Tatsachen voraussetzt und auf richtige und vollständige Anfangsbedingungen angewiesen ist (Chalmers & Bergemann 2007, S. 48). Da Produktionsdaten lediglich interpretierte streuungsbehaftete Messungen sind, gelten die Kritikpunkte am Induktivismus hier ebenfalls. Somit bewegt sich auch diese Arbeit im fehlerbehafteten Raum zwischen unvollständigen, nicht zwingend richtigen Beobachtungen von Produktionssystemen und folglich nicht absolut sicheren Aussagen über diese.

Popper versuchte diesen Problemen mit der Entwicklung der Lehre von der Falsifizierbarkeit zu begegnen. Daraus folgte später die Formulierung des kritischen Rationalismus (Popper 1935; Gadenne 1999, S. 87). Wissenschaftlicher Fortschritt erfolgt demnach durch die Formulierung von falsifizierbaren Aussagen und Vermutungen. „Bedeutsame Fortschritte werden durch die Bewährung von kühnen Vermutungen oder durch die Falsifikation von behutsamen Vermutungen gekennzeichnet", sagt Chalmers und fügt an, dass die Kategorien *kühn* und *behutsam* zeitlichen Änderungen unterworfen sind (Chalmers & Bergemann 2007, S. 67). Der Unterschied zwischen Induktivismus und Falsifikationismus liegt darin, dass ersterer eine Möglichkeit zur Suche nach Wahrheit ist, während letzterer die Suche nach Verbesserung beschreibt.

Abbildung 1.5: Induktion und Deduktion in Anlehnung an Chalmers (links) und im Kontext des zu erzielenden Arbeitsergebnisses (rechts)

Die Anwendung des Ergebnisses dieser Arbeit wird immer ein induktiv-deduktiver Prozess bleiben (siehe Abbildung 1.5 in Anlehnung an (Chalmers & Bergemann 2007, S. 46)). Die Vorgehensentwicklung und die Gesamtkonzeption der Arbeit setzt sich den Falsifikationsversuchen aus und will damit einen Beitrag zur Verbesserung leisten. Der Autor ist sich dem Grundproblem der Erkenntnislogik von Induktion und Deduktion bewusst und greift dennoch mit dem Anspruch auf Falsifizierbarkeit der gemachten Aussagen darauf zurück. Im Folgenden werden daher zur Lösung der Grundprobleme realwissenschaftlichen Denkens (Subjektivitäts- und Kommunikationsproblem) die eingangs bereits genannten Aspekte Entdeckungs-, Begründungs- und Verwendungszusammenhang für diese Arbeit näher beschrieben, unter der die Probleme zu lösen sind. Der Autor orientiert sich dabei am Begriffsverständnis von Ulrich et al. und Töpfer.

Entdeckungszusammenhang

Der „gedankliche Bezugsrahmen" oder die „konzeptuelle Basis" werden von Ulrich und Hill als heuristisch bezeichnet. So sind die Bedingungen, unter denen Wissenschaftler zu neuen Erkenntnissen kommen, beliebig unterschiedlich. Töpfer et al. sprechen von einer „weitgehenden Freiheit" im Bezug auf das Ideenspektrum, den Forschungsanlass und die Erkenntnismotivation sowie in „Art und Inhalt der eingesetzten Methoden". Der gedankliche Bezugsrahmen beschreibt die Bedingungen, unter denen der Forschungsprozess zum Ergebnis geführt werden soll.

Das Vorgehen des Autors ist motiviert durch den wissenschaftlichen Fortschritt und orientiert sich am Leitmotiv der Hypothesenprüfung (Kubicek 1977, S. 5). Im Sinne des pragmatischen Wissenschaftsziels erschließt sich die Problemstellung und Aufgabe *induktiv* aus einer Vielzahl vom Autor durchgeführten praxisnahen Industrie- und Forschungsprojekten. Der iterative Findungsprozess folgt der Prämisse, für ein spezielles Problem eine möglichst

allgemeine Lösung bereitzustellen, und beabsichtigt damit einen wesentlichen Beitrag zur späteren, dann *deduktiven* Prognose. Für die Lösung kombiniert der Autor die Aussagen von formal- und technikwissenschaftlichen Erkenntnissen. So werden Zeichensysteme aus der Mathematik und die daraus abgeleiteten Zusammenhänge, welche vorwiegend in der Signalverarbeitung Anwendung finden, auf Erkenntnisse in den Bereichen der Informatik und des maschinellen Lernens übertragen. Theoretische Basis bildet die Systemtheorie. Dazu werden ausgehend von den Arbeiten zu technischen Systemen von Ropohl Produktionssysteme, System Dynamics und Modelle ausführlich betrachtet (Ropohl 2009; Ropohl 2012). Die theoretische Herleitung der Zusammenhänge zwischen Produktionssystem und Turbulenz fußt auf den Grundlagenarbeiten zur Komplexität (Ashby 1957; Bertalanffy 1968; Hayek 1972; Beer 2000; Malik 2015) unter Berücksichtigung der themenverwandten Veröffentlichungen bis zur Gegenwart. Mit zum Entdeckungszusammenhang gehören die „Bildung und Modifikation von Hypothesen", welche in Abschnitt 1.6 genau erläutert werden (Kubicek 1977, S. 6).

Begründungszusammenhang

Zur Überprüfung der im gedanklichen Bezugsrahmen aufgestellten Hypothesen stehen zwei Themen im Zentrum. Zum einen das Ergebnis der Hypothesenprüfung als wissenschaftliches Ziel. Töpfer nennt dieses Standhalten der Hypothesen den „Bestätigungsgrad in der Realität". Zum anderen das Vorgehen der kausalen Kette sequenzieller Handlungsschritte für die Hypothesenprüfung an sich. Ziel ist also eine „möglichst exakte, intersubjektiv nachprüfbare und möglichst objektive Prüfung der Hypothesen" (Töpfer 2010, S. 49). Dies beschreibt die Bedingungen, unter denen „singuläre Beobachtungen überprüft und verallgemeinert" werden können (Ulrich 2001, S. 343). Dazu kommen „Anforderungen an die Form und den Aussagemodus der zu prüfenden Hypothesen" sowie die Prüfbedingungen und -verfahren (Kubicek 1977).

Diesem Verständnis folgt auch das Vorgehen dieser Arbeit, in der im Rahmen des Lösungsansatzes Hypothesen aufgestellt werden, die im Falle des Standhaltens einer Überprüfung die oben genannten Forschungsfragen beantworten. Die Lösungshypothesen basieren auf dem kombinierten Wissen zu komplexen Systemen, Produktionssystemen und Klassierungsmethoden. Sie werden empirisch-induktiv hergeleitet. Der Forschungsprozess geht von

praxisnahen Startwerten aus und variiert seine Parameter in Anlehnung an das Design of Experiments (DoE) für das als pragmatisches Ziel definierte Werkzeug. Dafür wird die Lösung rechnergestützt mit simulativ und randomisiert erzeugten sowie gesammelten Realdaten überprüft. Dies dient sowohl der Verifikation der grundsätzlichen Funktionalität der Lösung als auch der Möglichkeit zu Aussagen über die Güte des Ergebnisses (Validierung). Diese Arbeit ist folglich im Bereich der Normalwissenschaften einzuordnen (siehe Tabelle 1.2). Das sind solche, die innerhalb eines Paradigmas arbeiten und von ihren eigenen Fehlern lernen können (siehe (Chalmers & Bergemann 2007, S. 90) und (Kuhn 1970; Rapp 1973)). Der Ansatz dieser Arbeit soll die bestehenden Theorien im Bereich des maschinellen Lernens in Richtung dynamische Produktionssysteme erweitern, um bestehende Bewertungs- und Prognoseverfahren zu verbessern.

Tabelle 1.2: Normale und revolutionäre Wissenschaft

	Normale Wissenschaft	Revolutionäre Wissenschaft
Naturwissenschaft	Ausbau einer bestehenden Theorie	Umbruch der Denkweise
Technikwissenschaft	Verbesserung eines Verfahrens	Umwälzendes neues Prinzip

Verwendungszusammenhang

Töpfer betont die „zentrale praxisorientierte Zielsetzung der Übertragung gewonnener Erkenntnisse in die Realität“ als wesentliche Aufgabe des Forschungsprozesses und stellt damit den Bezug zur gesellschaftlichen Bedeutung in der Praxis her (Töpfer 2010). Die gesellschaftliche Funktion wissenschaftlicher Aussagen, ebenfalls beleuchtet von Ulrich, zielt bei dieser Arbeit auf den Betrieb von Produktionssystemen ab. Im Umgang mit komplexen Systemen dient die Lösung zunächst der Transparenz und damit dem besseren Verständnis solcher Systeme. Die Prognose und Bewertung von Turbulenzen in der Produktion kann Unternehmen helfen, ihre Ressourcen optimal zu nutzen und so auch den Zielen Nachhaltigkeit und Menschorientierung zu dienen. Mittel dafür sind Verfahren und Werkzeuge, die im Sinne von Rapp technische Gebilde oder Artefakte sind, die sich erst durch ihren praktischen Verwendungszweck von Kunstobjekten unterscheiden (Rapp 1973, S. 109).

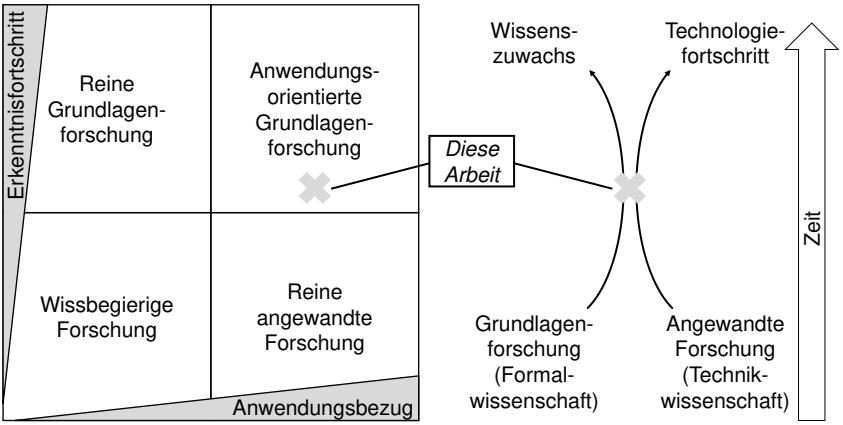


Abbildung 1.6: Quadranten- und Trajektorienmodell der wissenschaftlichen Forschung in Anlehnung an (Stokes 1997)

Die Forschungsmethodologie dieser Arbeit ist gleichzeitig problemorientiert motiviert und explorativ konzipiert. Beide Aspekte spiegeln sich im von Stokes beschriebenen Quadrantenmodell nach Pasteur (Stokes 1997, S. 73) wider. Das Modell spannt die beiden Dimensionen Anwendungsbezug und Erkenntnisfortschritt auf und identifiziert in den jeweils reinen Ausprägungen die Grundlagenforschung- und die angewandte Forschung als die beiden Gegenpole. Ihre Mischform wird als anwendungsorientierte Grundlagenforschung bezeichnet, in der diese Arbeit verortet wird (siehe Abbildung 1.6, links). Stokes erweitert das Modell um den Entwicklungspfad hin zu besserem Wissen über den Forschungsgegenstand und verbesserter Technologie. Er beschreibt dieses dynamische Modell mittels zwei Trajektorien mit den Zielen ‚Wissenszuwachs' und ‚Technologiefortschritt' als einen über die Zeit fortschreitenden Prozess (Stokes 1997, S. 88) (Abbildung 1.6, rechts). Je nach beabsichtigtem Forschungsziel lassen sich auch *Rigour*, also die theoretische Exaktheit, sowie *Relevance*, die praktische Relevanz analog den Achsen des Quadrantenmodells zuordnen (Töpfer 2010, S. 56; Anderson et al. 2001, S. 394). Somit kann jeder wissenschaftlichen Arbeit ein Theorie-Praxis-Verhältnis zugeordnet werden. Der von Töpfer als Königsweg bezeichnete hohe Anspruch an sowohl Rigour als auch an Relevance ist eine Eigenschaft der „pragmatischen Forschung", zu der diese Arbeit sich ebenfalls zählt.

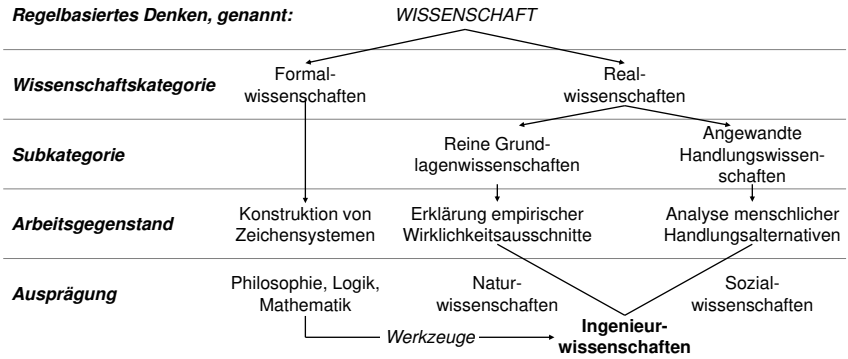


Abbildung 1.7: Wissenschaftssystematik in Erweiterung zu *Ulrich und Hill*

Diese Arbeit ist im klassischen Spektrum der **Wissenschaftskategorien** innerhalb der Realwissenschaften (Ulrich & Hill 1976, S. 305) zu verorten und enthält durch ihre anwendungsorientierte Konzeption Attribute der Sozialwissenschaften (siehe Abbildung 1.7). Hier steht nach Ulrich und Hill „die Analyse menschlicher Handlungsalternativen zwecks Gestaltung sozialer und technischer Systeme im Vordergrund". Gleichzeitig zielt das Ergebnis auf die Erklärung empirischer Wirklichkeitsausschnitte ab und ist somit im Bereich der hybriden „Ingenieurwissenschaften" zwischen Natur- und Sozialwissenschaften einzuordnen. Die Arbeit nutzt Werkzeuge der Formalwissenschaften wie Logik und Mathematik, um das angestrebte Ergebnis zu erzielen.

Bei der Einordnung der Arbeit hinsichtlich ihrer Erklärungs- und Gestaltungsaufgaben zeigt sich ebenfalls diese ‚Zwischenposition' (Ulrich & Hill 1976, S. 309). Diese wirkt sich in der operationsanalytischen Konzeption des Forschungsprozesses insbesondere auf die Modellkonstruktion aus (Ulrich & Hill 1976, S. 348). Während Grundlagenwissenschaften sich der Erklärung von Wirklichkeitsausschnitten und so Erklärungsmodelle bilden, zielen die angewandten Wissenschaften auf den Gestaltungsaspekt und entwickeln Entscheidungsmodelle (Ulrich & Hill 1976, S. 305). Aufgrund der Neuheit des Ansatzes, des hohen wissenschaftlichen Risikos und dem Ziel des Machbarkeitsnachweises, strebt die Arbeit ein **Erklärungsmodell** an. Es werden die in Abschnitt 1.6 aufgestellten Hypothesen überprüft, um „allgemeine Aussagen [zu erhalten] und somit eine größere Anzahl an Fällen [zu] erklären" (Ulrich & Hill 1976, S. 49).

Abschließende Bemerkungen

Die Arbeit besitzt damit die Eigenschaften anwendungsorientierter Forschung (Ulrich & Schwaninger 2001, S. 220; Ropohl 2012, 89 f.). Im Entdeckungszusammenhang wird die praxisnahe Problemquelle in Industrieprojekten und in der anwendungsbezogenen Forschung beschrieben. Mit dem soziotechnischen System der Produktion ist der Betrachtungs- und Forschungsgegenstand ein vom Menschen künstlich geschaffenes Artefakt und kein natürliches Phänomen. So liegt auch das Forschungsziel selbst wieder im Bereich neuer künstlicher Wirklichkeiten. Das Forschungsziel selbst enthält die (Turbulenz-) Bewertung und trifft damit nicht wertfrei deskriptive sondern normative Aussagen. Der Ansatz des maschinellen Lernens deutet bereits aus seiner Konzeption heraus stärker auf funktionale Nützlichkeit denn auf das Finden von Wahrheit, was weniger eine abgeschlossene Theorie, sondern ein regelbasiertes Vorgehen erfordert. Wenn auch die Allgemeingültigkeit ein Kriterium und Anspruch dieser Arbeit ist, wird diese sich dennoch an ihrer praktischen Problemlösungskraft messen lassen müssen.

1.5 Terminologie und Themeneingrenzung

Ausgehend von den drei Dimensionen der Anwendung, der Gestaltung und des Ergebnisses (Tabelle 1.1) werden nun allgemein verwendete Terminologien initial festgelegt und das betrachtete Themenfeld eingegrenzt. Dies dient dem einheitlichen Begriffsverständnis, der Orientierung im vorliegenden Werk und steckt den forscherisch notwendigen Rahmen ab. Die Themeneingrenzung legt zunächst den adressierten **Wirtschaftszweig** fest und wählt die **Fertigungstypen** aus, in denen die Anwendung des Arbeitsergebnisses als plausibel erachtet wird. Der Abschnitt grenzt weiter die Anwendungen ab, welche explizit nicht im Fokus der Lösung stehen. Abschließend werden – strukturiert durch den Aufbau des Titels dieser Arbeit – die in diesem Kapitel erarbeiteten Begriffe zusammenfassend definiert und die **Gestalt des Ergebnisses** beschrieben.

Wirtschaftszweig

Die Einordnung des konkreten Anwendungsbereiches bedient sich der Wirtschaftszweigklassifikationen des Statistischen Bundesamtes (Statistisches Bundesamt 2008). Sie

wird als allgemein gültige Grundlage für die Erstellung von Statistiken über „Produktionswerte, in den Produktionsprozess eingeflossene Produktionsfaktoren (Arbeit, Betriebsmittel und Werkstoffe, Energie usw.), Kapitalbildung und Finanztransaktionen dieser Einheiten" genutzt. Die Arbeit adressiert den Wirtschaftszweig des verarbeitenden Gewerbes (Abschnitt C aus 21 Abschnitten), und damit die mechanische, physikalische oder chemische Umwandlung von Stoffen oder Teilen in Waren. Die Herstellung ist hier definiert als eine „wesentliche Änderung oder Neugestaltung von Waren". Herstellungsverfahren haben entweder Fertigwaren für den Gebrauch oder Verbrauch und Halbwaren zur weiteren Be- oder Verarbeitung als Ergebnis. Dabei gilt das „Zusammenbauen der Teile von Waren" – genannt Montage – auch als Herstellung von Waren. Diese Eingrenzung schließt damit die Wirtschaftszweige aus, in denen das Arbeitsobjekt die belebte Natur (z.B. Forst, Fischerei), Dienstleistungen (z.B. Handel, Gastgewerbe, Versicherung), öffentliche Belange (Kommunikation, Bau, Versorgung, Verteidigung) oder der Mensch (Unterricht, Kultur, Gesundheit) ist. Gleichzeitig weist diese Eingrenzung noch immer ein breites Anwendungsfeld auf. Prinzipiell beabsichtigt der Ansatz dieser Arbeit eine hohe Allgemeingültigkeit. Daher erfolgte die Eingrenzung beim Fertigungstyp nicht prinzipiell sondern aus Gründen der Plausibilität.

Fertigungstyp der volatilen Variantenfertigung

Die Fertigungstypen werden klassisch nach Auflagenhöhe oder Losgröße und Wiederholhäufigkeit eingeteilt (Schomburg 1980; Lödding 2016, S. 125). Wöhe nennt mit der Organisation (Baustellen-, Werkstatt-, Gruppen- oder Fließfertigung) und der Ortsabhängigkeit (ortsgebunden oder ortsungebunden) der Fertigung zwei weitere Kriterien (Wöhe et al. 2020; Barthel 2006, S. 34). Aus der Perspektive der Betriebs- bzw. Auftragsabwicklung nennt Schuh Auftragsfertiger, Rahmenauftragsfertiger, Variantenfertiger und Lagerfertiger als zweckdienliche Einteilung (Schuh 2006, S. 25; Westkämper 2006, 199 f.). Abbildung 1.8 zeigt die Einordnung der Fertigungstypen im Raum der Dimensionen ‚Wiederholhäufigkeit' und ‚Auflagenhöhe oder Losgröße'. Lödding und Eversheim geben mit Losgrößen zwischen 10 und 1.000 Stück und Jahresstückzahlen zwischen 1.000 und 100.000 Stück konkrete Zahlen für die Großserienfertigung an (Eversheim 1989, S. 13; Lödding 2016, S. 125). Die variantenreiche Serienproduktion ist bei der Schnittmenge zwischen Klein- und Großserie zu verorten (Schönsleben 2004, 345 f.), da hier die Variantenanzahl im Verhältnis

zu den Skaleneffekten durch erhöhte Stückzahlen noch vergleichsweise ungünstig ist. Der Anwendungsraum dieser Arbeit beschränkt sich nicht auf einen bestimmten hier genannten Fertigungstyp, sondern erstreckt sich auf das erweiterte Feld der Serienproduktion. Es ist zu erwarten, dass bei der Einmal- und Einzelfertigung die Leistungsfähigkeit des datenbasierten Vorgehens aufgrund der kleinen Datenmenge limitiert bleibt. So bleibt die Validität eines trainierten Modells gering, wenn der Trainingsdatensatz aus sehr wenigen Datenpunkten besteht. Umgekehrt sind die Systeme und Subsysteme im Bereich der Massenfertigung in ihrem Verhalten relativ wenig komplex, wonach andere, wie etwa analytische, Modelle effizienter angewendet werden können. Das größte Potenzial kann folglich im Bereich der Serienfertigung erwartet werden. Variantenreiche Klein- und Großserienfertigungen mit permanent neu gearteten Produktionsaufträgen, deren Produktionssysteme sich durch streuende Maschinenstörungen und stark schwankenden Prozesszeiten auszeichnen, werden im Rahmen dieser Arbeit als *volatil* bezeichnet (siehe (Schickmair 2010, S. 15, 72)). Der Begriff der **volatilen Variantenfertigung** vereinigt damit variantenreiche Groß- als auch Kleinserienfertigungen mit hoher Parameterspreizung in ihrem Produktionssystem. Auftragsfertiger stehen als solche stark unter dem Einfluss des Kunden und damit der Produktionsaufträge. Dadurch entsteht eine ausreichende Wiederholhäufigkeit der Produkte und Arbeitsabläufe, um Werkzeuge des maschinellen Lernens plausibel anwenden zu können, und zudem ergibt sich so ein hohes Maß an Komplexität, was bisherige Lösungen im Bereich der Kennzahlenbewertung und -prognose an ihre Grenzen führt. Die volatile Variantenfertigung definiert damit den Gültigkeitsbereich dieser Arbeit.

Gestalt des konkreten Ergebnisses

Das Ergebnis dieser Arbeit ist kein Optimierungs- sondern ein **Bewertungswerkzeug**. Es hat nicht das Ziel, die Systemrobustheit zu erhöhen, sondern die Konsequenzen der gegebenen Systemrobustheit bewertend zu prognostizieren. Das Ergebnis hat damit *beschreibenden und messenden* Charakter (VDI 2017). So beschäftigt sich die Arbeit explizit nicht damit, wie Produktionsdaten generiert oder Methoden des maschinellen Lernens verbessert werden können, sondern damit, wie Auftragsdaten dynamischer Produktionssysteme zur Turbulenzprognose prinzipiell genutzt werden können. Die Ableitung geeigneter Maßnahmen ist ein wesentlicher Folgenutzen, jedoch nicht Teil dieser Arbeit. Abbildung 1.9

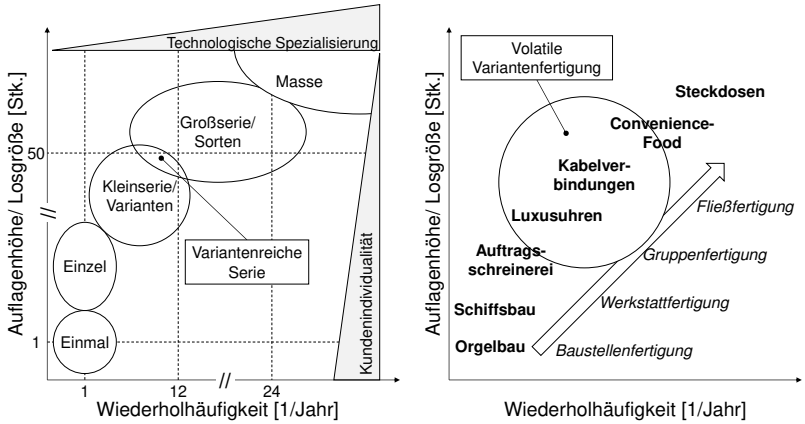


Abbildung 1.8: Zuordnung zu Fertigungstypen nach (Schomburg 1980) in Anlehnung an (Lödding 2016, S. 125)

enthält die zentralen Begriffe des Titels der Arbeit. Grundlage und ‚Ausgangsmaterial' der Lösung sind *Produktionsdaten*. Konkret werden Zeitreihen von Eingangs- und Zielgrößen genutzt, um Prognosewissen zu generieren. Die *Bewertung* bezieht sich auf ausgewählte Key Performance Indicators (KPIs) und deren Abweichung vom Sollwert. Zentralen Einfluss auf das Produktionssystem und damit die Eingangsgrößen haben die Auftragsdaten. Daher werden in der Herleitung zu dieser Lösung andere externe Faktoren vernachlässigt und im Speziellen die *kundeninduzierten* Einflüsse genau betrachtet. Bewertungsgegenstand ist die *Turbulenz*. Sie kann als in der Systemantwort abgebildete Abweichung von einem definierten Soll-Wert – also dem optimalen Betriebspunkt – verstanden werden. Auf den für diese Arbeit zentralen Begriff der Turbulenz wird im Unterabschnitt 2.1.1 ausführlich eingegangen. Der Betrachtungsrahmen ist die oben beschriebene *volatile Variantenfertigung*.

1.6 Lösungshypothesen

Im Werk ‚Logik der Forschung' bezieht sich Popper bereits in der Einführung auf Hypothesen als Tätigkeitsobjekte des wissenschaftlichen Forschers (Popper 1935, S. 1). Töpfer nennt Hypothesen ‚Kernstücke erkenntniswissenschaftlicher Forschungen' (Töpfer 2010,

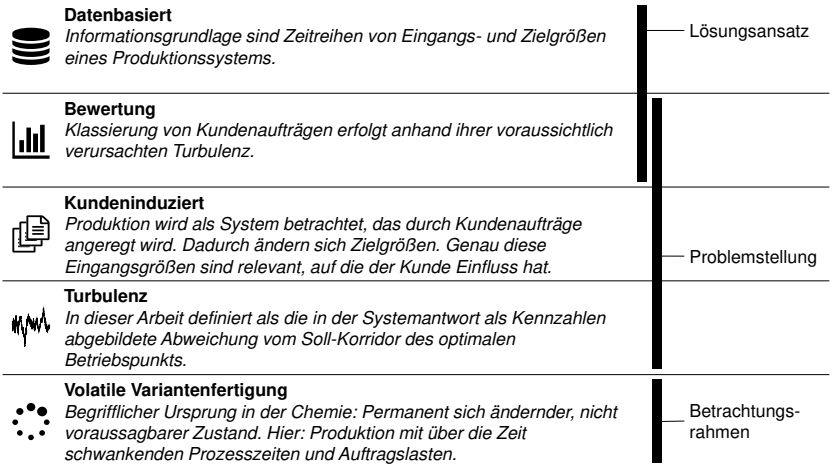


Abbildung 1.9: Terminologien im Titel der Arbeit

S. 173). Er zählt implizit folgende Kriterien für wissenschaftliche Hypothesen auf: Anspruch bezogen auf Wortwahl, Inhalt, Struktur, Stringenz und Falsifizierbarkeit. Aus dem Griechischen stammend kann Hypothese mit ‚Unterstellung' übersetzt und als „Vermutungen über die Gültigkeit bestimmter Ursachen-Wirkungs-Beziehungen zur Erklärung oder zur Prognose von realen Sachverhalten" beschrieben werden (Töpfer 2010, S. 175).

Das Aufstellen und Verwenden ‚alltäglicher' Hypothesen ist Teil des normalen Denkprozesses innerhalb und außerhalb des Produktionskontextes. Das Ergebnis dieser Arbeit generiert fortwährend mit funktionalen Abhängigkeiten und Prognosen weitere Hypothesen als Entscheidungsgrundlage für das Management, welche somit auch falsifiziert werden können. So schreibt Kubicek, dass Datengewinnung und Datenauswertung dem Zweck dienen, Hypothesen zu prüfen, um daraus Schlüsse für individuelle Probleme der Praxis zu ziehen (Kubicek 1977, S. 6). Die Lösungshypothesen dienen als Arbeitshypothesen und sind gehaltvoller als alltägliche Ad-hoc-(Hypo-)Thesen. Arbeitshypothesen können Ursache-Wirkungs-Beziehungen in Wenn-Dann-Beziehungen überführen. Durch einen Entwicklungs- und Reifeprozess wird aus einer Arbeitshypothese eine Wissenschaftshypothese. Voraussetzung für solche Hypothesen ist die oben erwähnte Falsifizierbarkeit. Auch Hypothesen, welche an der Realität scheitern, bieten wissenschaftlichen Fortschritt,

wenn die Gründe des Scheiterns erörtert oder neue Hypothesen aufgestellt werden. In der wissenschaftlichen Konzeption der Arbeit liegen also in diesem Kapitel Arbeitshypothesen vor, die im Falle ihrer Bestätigung zu wissenschaftlichen Hypothesen werden (Töpfer 2010, S. 178). Töpfer formuliert folgende Anforderungen an Hypothesen (Töpfer 2010, S. 179):

Eine Hypothese ...

- ... ist Teil eines systematischen Zusammenhangs.
- ... darf nicht in Widerspruch zu anderen Hypothesen des systematischen Zusammenhangs stehen.
- ... muss so formuliert sein, dass sie empirisch überprüfbar, und damit prinzipiell widerlegbar ist.
- ... ist operationalisierbar und somit in beobachtungsrelevante Begriffe übertragbar.
- ... kann auch Nebenbedingungen, moderierende bzw. intervenierende (zwischen Ursache und Wirkung liegende) Variablen enthalten.
- ... enthält keine Wertung.

Die Hypothesen dieser Arbeit gelten für den in Abschnitt 1.5 aufgezeigten systematischen Zusammenhang und kombinieren die *projektivo-pragmatische* Methode der Technik mit der *hypothetico-deduktiven* Methode der Naturwissenschaften (Rapp 1973, S. 127). So hat die Hypothesenerstellung naturwissenschaftlich-explorativen Charakter, während die Lösungsfindung ein pragmatisch ausgerichteter Prozess ist. Rapp schreibt, dass „Im Stadium der Konzeption [...] die Aufgabe zunächst darin [besteht], aufgrund eines heuristischen Modells bestimmte Arbeitshypothesen über die Struktur des zu untersuchenden Phänomenbereichs aufzustellen“. Erst danach können Ausgestaltung und Umsetzung folgen.

Aus den Überlegungen in den Abschnitten 1.2 bis 1.5 ergeben sich folgende notwendigen Grundannahmen, die als ‚konsensbedürftige Prämissen‘ vorausgesetzt werden:

- Wenn Systemkomplexität nicht vollständig kompensiert wird, entsteht Turbulenz.
- Um mit geeigneten Maßnahmen auf drohende Turbulenz proaktiv agieren zu können, bedarf es einer Voraussage und Bewertung dieser zu erwartenden Turbulenz.
- Das prognoserelevante Wissen dazu liegt implizit in Produktionsdaten vor.

- Es existieren die für die Bewertung notwendigen Daten in ausreichender Menge und Qualität.
- Für die Lösung notwendigen Rechen- und Speicherkapazitäten sind verfügbar.

Unter Berücksichtigung der oben beschriebenen Prämissen und der Anforderungen an Hypothesen nach Töpfer werden folgende *Arbeitshypothesen* formuliert:

1. »*Wenn historische Produktionsdaten statischer Produktionssysteme in praktikabler Menge und Qualität vorliegen, dann lassen sich damit mit Methoden des maschinellen Lernens rechnergestützt Turbulenzprognosen generieren.*«

2. »*Wenn datenbasierte Modelle bei Turbulenzprognosen für ein Produktionssystem vorliegen, dann lassen sich durch selektive Auswahl von Datenpunkten aus den Trainingsdaten Modelle für das dynamische Verhalten des Systems generieren.*«

Die Arbeitshypothesen bauen aufeinander auf. Die erste Hypothese betrifft die Frage, wie quantitative, datenbasierte Bewertung von Turbulenz in der Produktion *prinzipiell* ermöglicht werden kann. Sie beinhaltet Eigenschaften und Bedingungen der genutzten Daten (historisch, umfänglich, qualitativ), adressiert die Produktion als System und nennt mit der Turbulenzprognose den beobachtungsrelevanten Begriff. Zudem enthält die Arbeitshypothese mit dem maschinellen Lernen einen konkreten Lösungsweg. Die zweite Hypothese setzt den Nachweis der ersten voraus und geht darüber hinaus. So reicht das reine Prinzip der Turbulenzbewertung nicht aus, es ist vielmehr ein Modell notwendig. Die Hypothese betrifft die Problematik der Prognose in dynamischen Systemen und legt ebenfalls den Lösungsansatz – der selektiven Datenauswahl – fest. Damit werden die Arbeitshypothesen zu Lösungshypothesen.

1.7 Struktur der Arbeit

Ulrich betont in seinen Ausführungen zur anwendungsorientierten Wissenschaft die Bedeutung der Praxis. So beginnt der an seine Arbeiten angelehnte Forschungsprozess mit der Erfassung und Typisierung praxisrelevanter Probleme und endet mit dem Wissenstransfer in die Praxis durch Beratung (Ulrich & Schwaninger 2001, 194 f.). Damit bildet sie die

Klammer um den eigentlichen wissenschaftlichen Prozess. Abbildung 1.10 zeigt die an den Forschungsprozess nach Ulrich angelehnte Struktur dieser Arbeit als Übersicht. In Kapitel 1 findet sich die Praxis in der Ausgangssituation, der Problemstellung sowie der Darlegung von Ziel und Zweck der Arbeit wieder. Hier wird das praktische Interesse der Industrie dargelegt und davon die Forschungsfragen – und damit das theoretische Interesse – abgeleitet. In Kapitel 2 erfolgt die Erfassung und Interpretation von der Problemlösung dienlichen Theorien der Formalwissenschaften. Das Vorgehen dieser Arbeit nutzt die Systemtheorie als Rahmen der Betrachtung von Produktionssystemen. Weiter bedient es sich der formalwissenschaftlichen, logischen Sprache der Mathematik, um diese auf datenwissenschaftliche Aspekte anzuwenden. Darauf folgt in Kapitel 3 die Spezifizierung dieser Theorien und Hypothesen im Stand der Technikwissenschaft. Dieser beinhaltet die aus der Formalwissenschaft abgeleiteten problemrelevanten Verfahren. Der Stand in Formal- und Technikwissenschaft bilden den theoretischen Unterbau für die Herleitung der Lösung. Kapitel 4 definiert anhand der Systemanforderungen die Kriterien, anhand derer die Lösung gemessen und bewertet werden soll. Die eigentliche Lösung wird in Kapitel 5 hergeleitet und beschrieben. Die Struktur und Eigenschaften der verwendeten Produktionsdaten sowie das Vorgehen zur Erstellung des Prognosemodells bilden den Kern der Lösung. Das Ergebnis wird darauf aufbauend für dynamische Systeme weiterentwickelt. Kapitel 6 beschreibt, wie der Lösungsentwurf in ein ausführbaren Programm überführt werden kann. Verifikation und Validierung erfolgen in Kapitel 7. Hier werden die anfangs aufgestellten Hypothesen überprüft, indem die Lösung in drei Anwendungsfällen getestet wird. Die Funktionalität wird zunächst anhand eines auf Simulationsdaten basierenden Produktionsdatensatzes, welcher ein statisches Produktionssystem abbildet, verifiziert. Zweitens wird die Validität der Lösung an einem synthetischen Datensatz überprüft. Das soll bewerten, wie sich das Modell unter dynamischen Bedingungen verhält. Abschließend wird die Praktikabilität an einem Realdatensatz eines produzierenden Unternehmens nachgewiesen. Den Abschluss der Arbeit bildet mit Zusammenfassung und Ausblick das Kapitel 8, in dem sowohl das praktische als auch das theoretische Interesse dargelegt wird. Diese werden in Handlungsvorschlägen für die industrielle Anwendung und in neuen, während der Arbeit aufgeworfenen Forschungsfragen beschrieben.

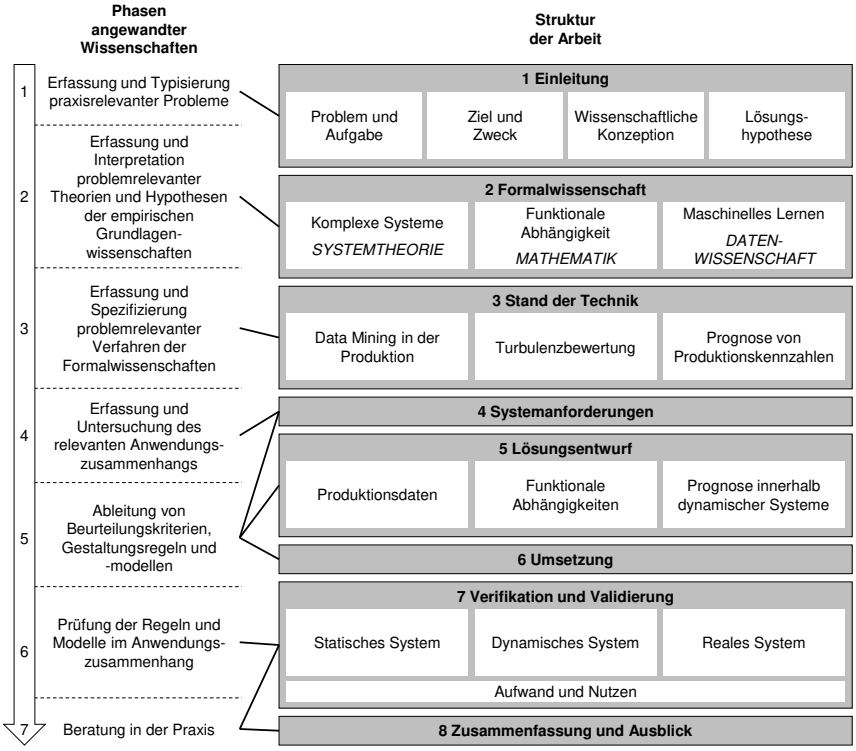


Abbildung 1.10: Struktur der Arbeit

2 Formalwissenschaftliche Grundlagen

Um ein Vorgehen zur quantitativen Bewertung von kundeninduzierten Turbulenzen innerhalb von Produktionssystemen zu entwickeln, werden in diesem Kapitel die formalwissenschaftlichen Grundlagen dargestellt. Dazu werden – basierend auf der Problemstellung entsprechend dem nach Ulrich vorgesehenen Forschungsprozess – die problem- und lösungsrelevanten Theorien der Formalwissenschaften erfasst und interpretiert. Ausgangspunkt ist die Betrachtung der Produktion als *turbulentes und komplexes System*. Daher werden in Abschnitt 2.1 die in der Literatur teilweise unterschiedlich verwendeten Begriffe zu Turbulenz und Komplexität erläutert und das für diese Arbeit herangezogene Begriffsverständnis festgelegt. Die Systemtheorie bildet den theoretischen Unterbau für die Betrachtungen. Dabei wird auch auf die begrenzte Modellierbarkeit in Abhängigkeit der Komplexität eingegangen. Der Abschnitt 2.2 enthält die mathematischen Grundlagen für die Lösung. Zentral sind dabei die funktionalen Abhängigkeiten der Komponenten innerhalb von Systemen. Der Fokus wird dabei auf die mathematische Darstellung von Ereignis und Zustand sowie die Detektion von kausalen Zusammenhängen gelegt. Den Abschluss des formalwissenschaftlichen Teils bildet die Darlegung der thematischen Grundlagen zum maschinellen Lernen (Abschnitt 2.3). Kern ist der ausgewählte Ansatz mit Entscheidungsbäumen.

2.1 Turbulente, komplexe Systeme

Das Forschungsfeld zu komplexen Produktionssystemen erfuhr in der Mitte der 2000er Jahre eine Renaissance (Herrera Vidal & Coronado Hernández 2021, S. 327). Die wissenschaftliche Dynamik innerhalb des Themas hält bis heute an. Dies begründet sich in den neuen Perspektiven zur Unterstützung in der Produktion durch die technischen Entwicklungen im Bereich der Datenverarbeitung und dem hohen Interesse am Thema Entscheidungsfindung allgemein. Das zeigt sich auch in der Popularität heuristischer Ansätze wie die von Gigerenzer (Gigerenzer 2015).

Es werden jedoch zentrale Begriffe gerade im Bezug auf Turbulenz und Komplexität unterschiedlich bis widersprüchlich verwendet. Für diese Arbeit soll durch Darlegung des Themen- und Begriffsverständnisses zu Turbulenz, System und Komplexität eine eindeutige Verwendung ermöglicht werden. Zur Turbulenz wird die Begriffsherkunft aus der Strömungsmechanik beschrieben (Unterabschnitt 2.1.1). Dies dient dazu, die Analogie des Turbulenzbegriffs für die Produktion zu verstehen. Darauf aufbauend wird aufgezeigt, wie die Turbulenz im industriellen Kontext in der Literatur verwendet wird. Unterabschnitt 2.1.2 beschreibt die systemtheoretischen Grundlagen, um damit den Betrachtungsrahmen aufzuspannen. Zudem bildet es den Brückenschlag zur Komplexität im Kontext der industriellen Anwendung (Unterabschnitt 2.1.3). Ein Fazit, in dem die Turbulenz als Konsequenz der Komplexität im Rahmen der Systemtheorie dargestellt wird, schließt das Themenfeld zu turbulenten, komplexen Systemen ab.

2.1.1 Turbulenz

„I am an old man now, and when I die and go to heaven, there are two matters on which I hope for enlightenment. One is quantum electrodynamics and the other is the turbulent motion of fluids. About the former, I am really rather optimistic."

zitiert nach Horace Lamb, 1932 (Goldstein 1969, S. 23)

Der Begriff der Turbulenz hat sprachlich seine Wurzeln im Lateinischen (*turbulentus* unruhig, stürmisch). Im Bereich der Wissenschaft ist die Turbulenz in der Strömungsmechanik – also der Physik von Fluiden (sowohl Gase als auch Flüssigkeiten) – beheimatet. Das klassische Verständnis der Turbulenz geht von der Analogie zur newtonschen Bewegungsgleichung von Flüssigkeiten aus (Gerthsen 2006, S. 115). Diese berücksichtigt Volumen-, Druck- und Reibungskräfte, welche auf alle Teilchenvolumen wirken. Abhängig von der Summe der angreifenden Kräfte wird das Teilchen beschleunigt. Die Wirkung der Kräfte wird in der Navier-Stokes-Gleichung bilanziert (Formel 2.1):

$$\varrho(\mathbf{a_1} + \mathbf{a_2}) = -\text{grad}\, p + \varrho\mathbf{g} + \eta\Delta\boldsymbol{v} \tag{2.1}$$

Dabei seien ϱ die Dichte, $\mathbf{a_1}$ und $\mathbf{a_2}$ die Beschleunigungsanteile des Gesamtsystems und des Teilchens selbst, p der Druck, $\mathbf{g}$ die Erdschwerebeschleunigung, η die materialabhän-

gige dynamische Viskosität und $\Delta \boldsymbol{v}$ der Geschwindigkeitsunterschied im Reibfall. Drei Strömungstypen sind dabei zu unterscheiden:

- Strömung idealer Flüssigkeiten
- laminare Strömung
- turbulente Strömung

Die ersten beiden Fälle beschreiben zum einen reibungs- und zum anderen beschleunigungsfreie Systeme (Viskosität $\eta \approx 0$ bzw. Geschwindigkeitsänderung $\mathbf{a_2} = 0$). Ändert sich die Geschwindigkeit entlang der Strömungslinien erheblich, kann $\varrho \mathbf{a_2}$ nicht mehr vernachlässigt werden und die stabile laminare Strömung geht über in turbulente Strömung. Der kritische Punkt des Übergangs von laminarer in turbulente Strömung (von Typ 2 zu Typ 3) ist systemabhängig. Um Systemmodellen laminares oder turbulentes Verhalten zuzuordnen, dient die Größe der Reynolds-Zahl Re. Die Kennzahl aus der hydrodynamischen Ähnlichkeitstheorie ist definiert als

$$Re = \varrho v l / \eta \tag{2.2}$$

mit Abmessungen l von Gefäß oder umströmten Körper, Strömungsgeschwindigkeit v, gegebener Dichte ϱ und Viskosität η der Flüssigkeit. Je nachdem, ob der Trägheitseinfluss oder der Reibungseinfluss überwiegt, tritt der turbulente oder laminare Fall auf. Für große Re ist Turbulenz zu erwarten. Das geschieht, wenn ausgehend von einem Turbulenzkeim lokal hohe Beschleunigungen vorliegen und damit der Trägheitseinfluss erhöht wird. Der Vergleich zu den beiden Aggregatzuständen *laminar* und *turbulent* liegt insofern nahe, als der laminare Zustand ebenfalls ‚unterkühlt' werden kann bis ein erster Turbulenzkeim entsteht.

Im Sinne einer besseren Vorstellung des Betrachtungsgegenstands dieser Arbeit können die physikalische Beschreibung oben und der Ansatz einer strömungsmechanischen Metapher von Nutzen sein (Efthymiou et al. 2009). So spricht beispielsweise Wiendahl vom *turbulenten Gebirgsbach* als einer Grundform von Produktionsauftragsströmen innerhalb einer Produktion (Wiendahl 2011, S. 284; Wiendahl et al. 2000). Auch nennt er die Auslöser von Turbulenzen im Auftragsmanagement Turbulenzkeime. Der Turbulenzbegriff hat sich

in den letzten beiden Jahrzehnten im wissenschaftlichen Bereich der Produktionsforschung und Fabrikplanung etabliert. Unterschiedliche Arbeiten beziehen sich auf Turbulenzen als Ursache für Planungsprobleme (Westkämper & Zahn 2009; Bullinger et al. 2003, S. 74; Zürn 2010, S. 19; Schuh 1997, S. 6) oder als eigentlicher Gegenstand wissenschaftlicher Betrachtungen (Wiendahl 2007). So entsprechen die Teilchen – in einem auf ein Produktionssystem mit Material- und Auftragsfluss übertragenen Fall – den Produktionsaufträgen. Die Volumenkräfte repräsentieren die von den Auftragsattributen abhängigen Eigenschaften wie Auftragsgröße oder Produktkomplexität. Die Beschreibung des Systemzustandes könnte analog zu den Druckkräften betrachtet werden. Beispiele wären die gegenwärtige Auftragslast, die Planungs- und Steuerungsregeln oder das aktuelle Kapazitätsangebot. Reibungskräfte entsprächen den lokalen oder temporären Abweichungen des Systemzustands von seinem Sollzustand, zum Beispiel Ressourcenausfälle, Fehlteile oder Ausschuss.

Auch für die oben genannten drei Strömungstypen lassen sich Analogien bilden: Der ideale, reibungslose Fall entspräche einer Produktion völliger Voraussagbarkeit und damit Planbarkeit bei vernachlässigbaren Wechselwirkungen. Der laminare Fall berücksichtigt Wechselwirkungen, jedoch sieht er keine Geschwindigkeitsänderungen vor (z.B. Eilaufträge). Im Falle der turbulenten Strömung wird aufgrund großer Geschwindigkeitsänderungen der (laminare) Auftragsfluss instabil. Trägheitsterme können dann nicht mehr vernachlässigt werden. Durch verschiedene Wechselwirkungen (in der Strömung der Flüssigkeiten eine Druckerhöhung), wird die Strömung ‚instabil' und demzufolge turbulent. Im Rahmen dieser Arbeit kann der Bereich des Umschlags zwischen laminarer und turbulenter Strömung für das Produktionssystem als der *optimale Betriebsbereich* bezeichnet werden.

Es ist offensichtlich, dass die Analogie nicht als vollständig betrachtet oder gar zur analytischen Berechnung von Zuständen komplexer Produktionssystemen verwendet werden kann. Dennoch scheint sich der Turbulenzbegriff als Ausgangspunkt wissenschaftlicher Überlegungen bewährt zu haben. Bullinger et al. nennen eine Liste an Turbulenzfaktoren: wechselnde Auftragslast, stark schwankende Stückzahlen, kleiner werdende oder wechselnde Losgrößen, die verstärkte Kundenindividualität der Produkte sowie die steigende Typen- und Variantenvielfalt (Bullinger et al. 2003, S. 74). Barthel spricht allgemein von „Umfeldturbulenz" (Barthel 2006, S. 65) und differenziert zwischen „eigen- und fremdinduzierten" Turbulenzen (Barthel 2006, S. 43), deren Reduktion, Vermeidung und Beherrschung einen

Teil seines Zielsystems darstellen (Barthel 2006, S. 33). Die Formulierungen implizieren auch hier, dass Turbulenzen nie völlig auflösbar sind. So spricht Wiendahl auch von der *wahrgenommenen* Turbulenz – einem subjektiven Eindruck, welcher in seiner Intensität durch das Maß an Überraschung und Überforderung definiert wird. In dem Zusammenhang wird die Turbulenz als die „Abweichung signifikanter Kennwerte von ihrem Mittelwert" (Wiendahl 2011, S. 208) definiert.

Bullinger spricht als Turbulenzursache von einem Ungleichgewicht zwischen externem Wandlungsdruck (Anforderungen) und interner Wandlungsfähigkeit (Kompetenzen und Ressourcen) (Bullinger et al. 2003, S. 76). Über die Schwierigkeit bei der Vorhersage von Turbulenzen schreibt Bullinger, dass sich wahrscheinlich „[...] diskontinuierlich verlaufende Entwicklungen gegenseitig aufschaukeln und sich zu Turbulenzen verdichten, die uns vor Herausforderungen stellen werden, auf die wir auf Grund der zahllosen interdependenten Verflechtungen unserer Welt nur mangelhaft vorbereitet sind" (Bullinger et al. 2003, S. 300).

Unabhängig von Definition und Gültigkeit der Analogien beziehen sich alle genannten Autoren auf nichtlineare Produktionssysteme – ähnlich der nichtlinearen Gleichungen von Navier-Stokes. Trotz des mathematisch schwer beschreibbaren Chaos der Instabilität besteht in Mehrkörpersystemen eine überlagernde, stabile Ordnung, in der ‚Muster' erkannt werden können (Gerthsen 2006, S. 129). So schreiben Gerthsen und Meschede zur Turbulenz:

„Tatsächlich verstehen wir viel besser, wie sich ein schwarzes Loch bildet, als wie schnell das Wasser aus einem Hahn strömt. [...] Erst in letzter Zeit scheint einiges Licht aus einer ganz unerwarteten Ecke zu kommen, nämlich der Spielerei mit Computern, bei der jeder auch ohne große Vorkenntnisse fundamental Neues entdecken kann, falls er genügend Spürsinn und Geduld hat." (Gerthsen 2006, 128 f.)

Diese Beschreibung skizziert im Grunde Motivation und Lösungsansatz für diese Arbeit und leitet damit zur genaueren Betrachtung des Wirkraums von Turbulenzen, nämlich den Produktionssystemen über.

2.1.2 Systemtheorie

Die Systemtheorie hat sich von Beginn an als wissenschaftsintegrierend konzipierte Denkweise erwiesen (Ropohl 2009, S. 71). Ihre systemtheoretischen Konzepte werden heute in zahlreichen Disziplinen der Natur- und Humanwissenschaften angewandt (Schwaninger 1996, S. 1947). Sie zeichnet sich als „[...] allgemeingültige, von fachlichen Restriktionen befreite Denkbasis" aus (Patzak 1982, S. 2). Ihre Eigenschaften sind inhaltliche Allgemeingültigkeit und Zweckorientiertheit bei formaler Abstraktheit. Sie gilt als umfassende sowie gleichzeitig strukturierende Betrachtungsweise. Unter anderen nennen Patzak und Ropohl einige Aspekte, die weitgehend unabhängig von der Denkrichtung der Systemtheorie gelten. So bestehen Systeme aus einer Menge von Teil- oder Subsystemen, die wiederum aus strukturgebenden Elementen mit Eigenschaften bestehen. Zwischen den Subsystemen untereinander, den Elementen und zum Gesamtsystem hin bestehen Relationen und Funktionen. Für Patzak hat die Systemtheorie im technischen Umfeld das Ziel, das Verhalten von Systemen durch Modelle zu beschreiben und vorherzusagen (Patzak 1982, S. 3). Daneben prägte sich die Systemtheorie interdisziplinär in viele weitere Richtungen aus. So seien hier mit der soziologischen Systemtheorie (Luhmann), der Kybernetik (Wiener und Ashby) und mathematischen Theorien zur Kommunikation (Shannon) einige wenige bedeutende Vertreter genannt (Luhmann 2002; Ashby 1957; Wiener 1961; Shannon 2001).

Im Wahrig wird die Systemtheorie als „Wissenschaft von der Beschreibung und Erklärung natürlicher, sozialer oder technischer Systeme, deren Strukturen und Funktionen sich gegenseitig bedingen und beeinflussen" beschrieben. Das System an sich sei ein „[...] in sich geschlossenes, geordnetes und gegliedertes Ganzes. Eine Gesamtheit und ein Gefüge von Teilen, die voneinander abhängig sind, ineinandergreifen oder zusammenwirken" (Wahrig-Burfeind 2011, S. 1451). Im Folgenden werden die Aspekte des systemtheoretischen Denkens herausgegriffen, welche für diese Arbeit relevant sind. Dazu zählen das Produktionssystem als System, die daraus weiterentwickelte Methodik ‚System Dynamics' und Modelle in der Wissenschaft. Das Kapitel schließt ab mit den thematischen Überschneidungen mit Komplexität, welche in Unterabschnitt 2.1.3 gesondert beleuchtet werden.

Produktionssystem

Die Autoren Bertalanffy – als Mitbegründer der *Allgemeinen Systemtheorie* – und Patzak unterscheiden offene und geschlossene Systeme (Bertalanffy 1968, S. 39; Patzak 1982,

S. 20). Nur offene Systeme können wachsen und sich entwickeln. Das erfolgt durch Materie-, Energie- und Informationsaustausch mit ihrer Umgebung. So kann trotz ständig wechselnder Elemente ein Fließgleichgewicht mit ihrer Umwelt etabliert werden. Für geschlossene Systeme ist die Zunahme der Entropie (Unordnung; 2. Hauptsatz der Thermodynamik) unumgänglich, nicht jedoch für offene Systeme. So können sie durch Aufnahme von Materie, Energie und Information immer andere Zustände, weiteres Wachstum und höhere Energieniveaus aufbauen, erreichen und halten (Schwaninger 1996, S. 1951).

Für die Beschreibung einer Produktion bietet die Systemtheorie drei unterschiedliche, sich gegenseitig nicht ausschließende Deutungen an: das strukturale, hierarchische und funktionale Konzept (Ropohl 2009, S. 75). Beim strukturalen Konzept stehen die Elemente und ihre Verknüpfung zu einer Ganzheit im Vordergrund. Die Aussage „das Ganze ist mehr als die Summe seiner Teile" fasst dies zusammen. Das hierarchische Konzept beschreibt die Produktion als System von Systemen. Es bestehen damit aus Super- und Subsystemen, die jeweils als Ganzheit betrachtet werden können. Es folgt daraus, dass sich Systeme beliebig genau, jedoch nie vollständig beschreiben lassen und impliziert die Eigenschaft von Modellen, immer nur Abbild von einem Ausschnitt der Wirklichkeit zu sein. Die ‚Black Box' ist die Metapher für das funktionale Konzept. Es erkennt an, Struktur und Hierarchie nie vollständig beschreiben zu können und fokussiert von außen beobachtbare Zusammenhänge zwischen seinen Eigenschaften. Zentrale Bedeutung haben hier die Eingangs- (Inputs) und Ausgangsgrößen (Outputs). Funktionales Systemdenken fragt also nicht „Was ist es?" sondern „Was tut es?", wie Ropohl schreibt. Hier beeinflusst neben den Eingangsgrößen der *Zustand* des Systems die Ausgangsgrößen. Als Zustand eines Systems wird die kleinste Menge von Variablen bezeichnet , die zu einem Zeitpunkt t bekannt sein müssen, um das Systemverhalten für jeden späteren Zeitpunkt voraussagen zu können (Schwaninger 1996, S. 1953; Law & Kelton 1991, S. 3). So lassen sich Input und Output mathematisch beschreiben und in Zusammenhang bringen. Die aus der Automatentheorie stammende formalisierte Systemdarstellung mit Zustandsgleichung (2.3) und Outputgleichung (2.4) schlägt die Brücke zu den Systembeschreibungen mit Zeitreihen (Schiemenz 1993).

$$x(t) = f[x(t_0, u(t_0, t), t)] \tag{2.3}$$

$$y(t) = g[x(t_0, n(t_0, t), t), t)] \tag{2.4}$$

Dabei ist t die Zeit, $x(t_0)$ der Vektor der Zustandsvariablen, $u(t_0, t)$ Vektor der Eingangsgrößen oder ‚Ursachen' zwischen t_0 und t, $y(t)$ der Vektor der Ausgangsgrößen oder ‚Wirkungen' und f und g die funktionalen Beziehungen mit dazugehörenden strukturellen Eigenschaften (Schiemenz 1993).

Unter Berücksichtigung der physischen und psychosozialen Bedürfnisse des Menschen wird das rein technische System der Produktion zum soziotechnischen System (Ropohl 2009, S. 25). In der Produktionsforschung hat sich der Begriff etabliert (Bullinger et al. 2003, S. 82; Westkämper & Löffler 2016, S. 59; Bauernhansl et al. 2016, S. 40). Bartolomei definiert das Wertschöpfungssystem als ein „komplexes soziotechnisches System, welches von Menschen so entworfen, entwickelt und gelenkt wird, um Nachfragern Werte zu liefern" (Bartolomei et al. 2006, S. 3). Die Definition von Bullinger et al. orientiert sich am funktionalen Konzept der Systemtheorie: „Unter soziotechnischen Systemen versteht man in diesem Zusammenhang offene und dynamische Systeme, die Inputs aus der Umwelt erhalten und Outputs in die Umwelt abgeben" (Bullinger et al. 2003, S. 82). Auch das Stuttgarter Unternehmensmodell bezieht sich explizit auf die Systemtheorie als Modellbasis und beschreibt ein offenes System, das mit seiner Umwelt Informationen, Material und Produkte austauscht (Westkämper & Zahn 2009, 47 ff.). Die funktionalen Abhängigkeiten seiner Elemente – etwa Personal und Ressourcen – sind geprägt durch Führungs- und Ausführungsprozesse. Beeinflusst werden Zustand und Output unter anderem durch Turbulenz. Das Stuttgarter Unternehmensmodell hat damit viele Parallelen zum allgemeineren Handlungssystem von (Ropohl 2009, S. 97). Basierend auf der Erkenntnis, dass soziotechnische Systeme – und Systeme im Allgemeinen – Veränderungen unterliegen (Honerkamp 1990) wird auf Ansätze im Bereich der System Dynamics folgend kurz eingegangen.

System Dynamics

Mit dem Anspruch, kontinuierliche Systeme zu simulieren, entwickelte Forrester in den 1960er Jahren am Massachusetts Institute of Technology (MIT) die System-Dynamics-Methodik (Forrester 1968; Schwaninger 2006). Sie beruht auf dem heute verbreiteten Ansatz, Modelle für die Simulation kontinuierlicher Systeme über zahlreiche Variablen und Rückkopplungen aufzubauen. Die Simulationsergebnisse lassen anhand visueller Abbildun-

gen wertvolle Einsichten in das Verhalten von Systemen zu (Schwaninger 1996, S. 1956). Der positivistisch geprägte Ansatz ist weit verbreitet und bietet viele Möglichkeiten im Bereich simulativer Steuerung von Systemen. Die Methodik versprach große Fortschritte im Unternehmensmanagement und wurde ebenfalls auf Produktions- und Logistiksysteme angewandt (Meyer et al. 2019; Coyle & Coyle 1996).

Bei Systemen mit sich stark veränderndem Umfeld gerät System Dynamics an seine Grenzen. Gefordert werden *robuste* Steuersysteme, die durch ihre Struktur wenig empfindlich reagieren. So soll auf sich verändernde Systemparameter reagiert werden, ohne zu übersteuern (Kaashoek 1990). Damit wird es möglich, *weitgehend unbekannte* Systeme zu steuern (Jones & Sbarbaro 1994). So werden in neueren integrativen Systemmethoden qualitative und quantitative Modellierungs- und Simulationsansätze verknüpft. Es kann festgehalten werden, dass die Dynamik in Systemen zu anderen Anforderungen sowohl an die Methoden für das Systemverständnis, als auch an die beschreibenden Modelle führt.

Modelle in den Wissenschaften

Die oben beschriebenen systemtheoretischen Betrachtungen erfordern Modelle für die Anwendung. Stachowiak nennt vier Arten von Modellen in der Wissenschaft. Das Demonstrationsmodell, das Experimentalmodell, das theoretische Modell und das operative Modell (Stachowiak 1973, S. 138). Die Grenzen verlaufen fließend. Idealtypisch sind *Demonstrationsmodelle* repräsentativ und veranschaulichen als Abbild Zusammenhänge des modellierten Systems. Zur Hypothesenprüfung oder -ermittlung werden *Experimentalmodelle* genutzt, die jedoch wie Demonstrationsmodelle ebenfalls der Veranschaulichung dienen können. Das *theoretische Modell* bildet das System in einem logischen Zeichensystem ab und das *operative Modell* schließlich dient als Entscheidungs- und Planungshilfe. Alle Arten von Modellen besitzen die gleichen Hauptmerkmale (Stachowiak 1973, S. 131). Modelle sind (1.) Abbildungen *von etwas* und repräsentieren künstliche oder natürliche Originale. Modelle können auch andere Modelle repräsentieren. Modelle sind insofern immer *unvollständig*, da sie die Attribute des Originals in Anzahl und Detaillierung (2.) immer verkürzt darstellen. Kriterium für die Auswahl der Attribute ist die subjektive Relevanz durch die Modellerschaffer und -nutzer. Die Attributauswahl und die Modellnutzung zeichnet (3.)

das pragmatische Merkmal aus. Modelle können unterschiedliche Originale repräsentieren und ihre Gültigkeit nur zu bestimmten Zeitintervallen und Bedingungen besitzen.

Für die Untersuchung von Systemen eignen sich laut Law und Kelton besonders Experimentalmodelle unterschiedlicher physischer und mathematischer Art (Law & Kelton 1991, S. 4). Dabei beziehen sich die Autoren gerade auf Produktionssysteme und wägen zur Auswahl der Untersuchungsart Kosten, physische Machbarkeit, praktische Folgen und Validität der Ergebnisse ab. So wird das (zunächst erwünschte) Experiment am eigentlichen System oftmals verworfen und durch Experimente an physischen, analytischen oder simulationsbasierten Modellen ersetzt (siehe Abbildung 2.1).

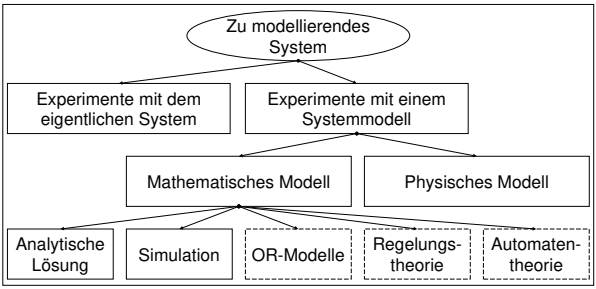


Abbildung 2.1: Arten, Systeme zu untersuchen in Anlehnung an (Law & Kelton 1991, S. 4)

Immer größeren Raum nehmen mit der Digitalisierung die mathematischen Modelle ein. Über die in Abbildung 2.1 gezeigten hinaus nennt Patzak noch Operations-Research-Modelle, regelungstheoretische Modelle und automatentheoretische Modelle. Deren Zweck ist explizit die Optimierung, Steuerung und Informationsverarbeitung von und innerhalb technischer Systeme (Patzak 1982, S. 306).

Die Systemtheorie kann also ein praktikabler Denkrahmen für die Lösung des vorliegenden Problems sein. Der Forschungsgegenstand ‚Produktion' lässt sich durch die Begriffe der Elemente, Relationen, Funktionen sowie In- und Output darstellen. Da für die Turbulenzbewertung eine Verhaltensprädiktion notwendig ist, fällt dieser Zweck mit dem Ziel der Systemtheorie, nämlich der Beschreibung und Vorhersage des Systemverhaltens, zusammen. In dieser Arbeit werden offene, kausale, zeitvariable und damit dynamische Produktionssysteme betrachtet. Aufgrund dessen bietet sich für die Systemmodellierung das funktionale Konzept an, in dem nicht das Verständnis zur inneren Struktur und dem hierar-

chischen Aufbau, sondern Ropohls „Was tut es?“ im Vordergrund steht. Die Zustands- und Outputgleichung können dabei zum Meta-Modell für diese Arbeit beitragen und stellen das theoretische Bindeglied zu systembeschreibenden Zeitreihen dar. Die Grundfragen der System Dynamics Methodik schwingen dabei immer mit. Sich verändernde Systeme sind nach wie vor der Grenzbereich, in dem die Herausforderungen liegen. Ursache dieser Herausforderung liegt für alle denkbaren Lösungsmodelle in der *Komplexität* des betrachteten Systems, weshalb im Unterabschnitt 2.1.3 vertieft auf den Begriff eingegangen wird.

2.1.3 Komplexität

„Most of the complex structures found in the world are enormously redundant, and we can use this redundancy to simplify their description. But to use it, to achieve the simplification, we must find the right representation.“ - Herbert A. Simon, 1932 (Simon 1969, S. 481)

In dem Aufsatz, aus welchem obiges Zitat entnommen ist, bedient sich Simon des Modells hierarchischer Systeme. Er übernimmt damit die Vorstellung, dass komplexe Systeme aus Subsystemen bestehen (Simon 1969, S. 468). Diese Annahme bezieht er auf physikalische, biologische, symbolische und eben auch auf soziale und technische Systeme. Solche Systeme sind evolutorisch bedingt und aus logisch-praktikablen Gründen hierarchisch und damit auch in Teileinheiten *zerlegbar*. Die **Komplexität** als ein Merkmal von Systemen entsteht in der Kombinatorik der Relationen zwischen den Teileinheiten. Simon bezieht sich auf Weaver, der in den 1940er Jahren zwischen ‚organisierter‘ und ‚unorganisierter‘ Komplexität unterschieden hat. So können je nach Grad der Ausprägung Verhaltensmodelle kategorisiert werden (Abbildung 2.2). In hierarchischen Systemen - mit einer Vielzahl an Relationen zwischen unterschiedlichen Elementen - herrscht organisierte Komplexität (z.B. eine Anlage mit einer Vielzahl an interagierenden Komponenten). Wenn die Anzahl an unterschiedlich gearteten Elementen zwar fassbar ist, die hohe absolute Elementanzahl jedoch nur statistische Modellierungen zulassen (z.B. eine Art Moleküle in Fluiden), spricht er von unorganisierter Komplexität (Weaver 1991).

Zu den im Unterabschnitt 2.1.1 beschriebenen Analogien zwischen der Fluidmechanik und einem Produktionssystem reiht sich Efthymious Analogie zwischen der Turbulenz

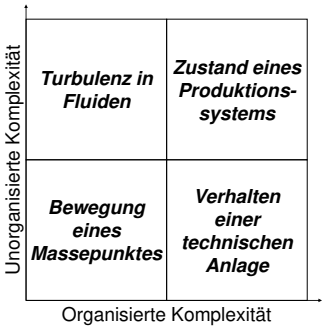


Abbildung 2.2: Komplexitätskategorie nach Weaver

und Komplexität ein (Efthymiou et al. 2009, S. 2). Beide teilen die Eigenschaft der Unvorhersagbarkeit in Bezug auf Geschwindigkeit, Druck und Temperatur (für die Fluidmechanik) sowie Flussgrad, Verzögerung und Termintreue (im Produktionssystem). Auch Bauernhansl bezieht sich mehrfach auf die durch Komplexität hervorgerufene Problematik im Kontext der digitalen Fabrik. Antworten darauf finden sich im Komplexitätsmanagement (Bauernhansl 2014b, S. 14; Bauernhansl 2014a; Schuh & Riesener 2018). Komplexität ist also ein zentrales Merkmal von Produktionssystemen und erschwert die Turbulenzprognose.

Nach der Beschreibung des Themenbezugs folgen nun die Ausführungen zu den Werken, welche als theoretische Basis für diese Arbeit den Komplexitätsbegriff und die Konsequenzen für das Systemmodell prägen. Malik bietet einen themenrelevanten Einstieg in das Begriffsverständnis (Malik 2015, 166 f.). So beschreibt er mehrere grundlegende, bereits angedeutete Aspekte: Wenngleich als Begriff inflationär verwendet, ist Komplexität eine gültige Rechtfertigung für die Reduktion von Problemen auf praktikable Modelle. Komplexität zwingt somit das Modell zur Verkürzung im Sinne Stachowiaks. Weiter betont Malik die umgangssprachliche Verwendung der Komplexität als Synonym für ‚kompliziert', ‚undurchschaubar' und ‚unverständlich'. Die in den Begriffen implizit gemeinte Überforderung steht in enger Verwandtschaft zum subjektiven Eindruck der wahrgenommenen Turbulenz von Wiendahl aus Unterabschnitt 2.1.1. Maliks Blick auf Systeme erlaubt einen analytischen Zugang trotz ausgeprägter Komplexität und die damit verbundenen starken Modellvereinfachungen erhalten so ihre Relevanz.

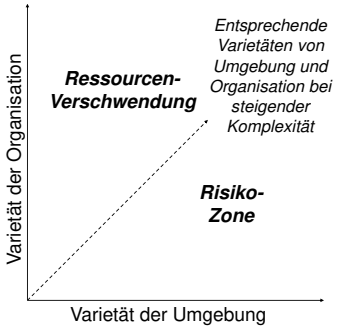


Abbildung 2.3: Varietätsanforderungen nach Ashby

Von großer Bedeutung ist der von Ashby etablierte Begriff der Varietät (Ashby 1957, S. 121) als Maßgröße für Komplexität in der Kybernetik. Varietät bezeichnet die Anzahl möglicher Zustände eines Systems (Lewis & Stewart 2003). Demnach kann ein System mit einer gegebenen Komplexität nur durch ein System gleicher oder höherer Komplexität kontrolliert werden. Bestimmungsgrößen für die Varietät sind „Verhaltensoptionen, Verhaltensrestriktionen und Bestimmbarkeit des Systemverhaltens“ (Schwaninger 1996, S. 1948). Als Konsequenz für das Management von Produktionssystemen muss die Varietät des Systems fortwährend auf die dem System begegnenden Varietät angepasst werden. Diese Anpassungsaktivitäten können unter dem Begriff der ‚Komplexitätsbewältigung‘ zusammengefasst werden. Andernfalls ist das Produktionssystem einem Risiko ausgesetzt oder Ressourcen werden verschwendet. Abbildung 2.3 veranschaulicht diesen Zusammenhang. Ziel ist also nicht eine minimale Komplexität des Produktionssystems, sondern eine an die Situation angepasste Komplexität (Fricker 1996, S. 3). Da die Varietät der Umgebung meist schwerer zu beeinflussen ist als das zu managende System, ist die Balance aus Ressourceneinsatz und Risikoaffinität die Optimierungsgröße (in der Darstellung die Ursprungsgerade). Daraus erwächst die Frage nach der idealen Komplexität und folglich deren Messbarkeit. So bestehen zwar exakte, doch gleichzeitig durch ihre Abstraktheit wenig praktikable Komplexitäts-Messmethoden mit Ansätzen aus der Informationstheorie und Mathematik unter Verwendung der Entropie (Brinzer & Schneider 2019). Andere Ansätze zur Messung der Komplexität - wie die von Brinzer oder auch Reiß - sind zwar praktikabler, jedoch noch immer nur qualitativer Art (Reiß 1992).

Abschließend soll noch auf das im Zusammenhang mit Komplexitätsbewältigung relevante Merkmal der *Resilienz* eingegangen werden. Mit die verbreitetste Definition ist die von Holling: Resilienz ist die Fähigkeit eines Systems, Änderungen zu kompensieren und zu seinem stabilen Gleichgewichtszustand nach einer Störung zurückzukehren (Holling 1973; Clapham 1983). Sandler dos Passos et al. haben gängige weitere Definitionen gesammelt, um die Begriffe der Robustheit und Antifragilität davon abzugrenzen (Sandler dos Passos et al. 2018). Aus Kompetenz-Perspektive ist Resilienz „[...] die Fähigkeit, ausreichend auf Ungewissheit durch einen Prozess des Lernens aus der Praxis zu reagieren und einen Wissensspeichers aus schwierigen Erfahrungen aufbauen zu können" (Sandler dos Passos et al. 2018, S. 9; Lengnick-Hall & Beck 2005). Robustheit hingegen ist „[...] eine Eigenschaft eines einfachen oder komplizierten Systems, das durch vorhersehbares Verhalten nach einer Störung wieder in seinen ursprünglichen Zustand zurückkehren kann" (Dahlberg 2015). Antifragilität ist die Fähigkeit, unter Druck besser zu werden (mehrere Quellen siehe (Sandler dos Passos et al. 2018, S. 9)). Aus der Kombination der unterschiedlichen Begriffe entwickeln Sandler dos Passos et al. den Begriff $Resilience_{new}$. Dieser beinhaltet damit die Anpassungs- und Lernfähigkeit und die Anforderung an ein gewisses Maß an Autonomie durch Selbstorganisation. Darüber hinaus sprechen Ivanov et al. von Stabilität, Widerstandsfähigkeit und Lebensfähigkeit (Ivanov & Dolgui 2020).

Auf Stabilität und den mathematischen Hintergrund wird in Unterabschnitt 2.2.3 gesondert eingegangen. Bei der Definition der übrigen Begriffe wird die Unschärfe in der Literatur deutlich. So werden Robustheit, Widerstandsfähigkeit und Resilienz autorenunabhängig synonym verwendet. Weiter prägen Thoma et al. den Begriff des ‚Resilienz-Engineering' (Thoma et al. 2016). Es beschreibt das Konzept, welches Methoden und Technologien für Systeme bereitstellt, um unvorhergesehene Ereignisse bei geringem Schaden oder sogar gestärkt zu überstehen. Im diesem Begriffsverständnis von Thoma et al. ist diese Arbeit ein Beitrag zum Resilienz-Engineering für Produktionen. Sie bedient sich für die Problemlösung der klassischen Resilienz-Definition von Holling mit den von Ponomarov et al. gesammelten ‚Komponenten' der Resilienz, die in Tabelle A.1 im Anhang dargestellt sind (Ponomarov & Holcomb 2009, S. 126).

Fazit:

Dieser Abschnitt beschreibt die formale Sicht auf turbulente, komplexe Systeme. Zunächst wird die Herkunft des Turbulenzbegriffs geklärt, die Bedeutung in der Physik beschrieben und auf die Produktion übertragen. Für diese Arbeit wird Turbulenz als die „Abweichung signifikanter Kennwerte von ihrem *Sollwert*" definiert und somit quantifizierbar. Die Produktion wird als offenes, stochastisches, kausales, zeitvariables und damit dynamisches System mit Elementen und Relationen betrachtet. Mit Bezug auf die volatile Variantenfertigung (siehe Abschnitt 1.5) wird der Komplexitätsbegriff herausgearbeitet. Ropohls funktionales Konzept ist aufgrund der stochastischen Eigenschaften das praktikabelste Beschreibungsmodell. Die Zustands- und Outputgleichungen bieten die Möglichkeit, ein System im Sinne von Griod mit Zeitreihen zu beschreiben: „Ein System ist die Abstraktion eines Prozesses oder Gebildes, das mehrere Signale zueinander in Beziehung setzt" (Girod et al. 1997, S. 6). Für eine solche Beschreibung sind Modelle notwendig. Die oben genannten Erkenntnisse zu Modellen werden für den Lösungsansatz und das Simulationsmodell für die Validierung und Evaluation herangezogen. Allen Modellen sind jedoch durch interne und externe Komplexität Grenzen gesetzt. Die ausführlichen Betrachtungen zur Komplexität führen den Varietätsbegriff nach Ashby und mit der Resilienz ein entsprechendes Bewältigungskonzept ein und bestätigen das Dilemma der nie vollständigen Modellierbarkeit. Mit der detaillierten Beschreibung der Resilienz und seinen Komponenten wird die dämpfende Wirkung auf Komplexität durch Resilienz beschrieben. So wird für diese Arbeit die Turbulenz T als Resultat nicht kompensierter – also durch Resilienz R nicht gedämpfte externe und interne Komplexität (K_e und K_i) – betrachtet. Dieser Zusammenhang kann als mathematische Relation dargestellt werden (Gleichung 2.5).

$$T \sim \frac{\sum(K_e + K_i)}{R} \tag{2.5}$$

Ausgehend vom in Abbildung 2.4 (Böhm & Bauernhansl 2021, S. 687) dargestellten funktionalen Systemverständnis für diese Arbeit stellt sich die Frage nach der Art der Modellierung. Das zu betrachtete System liegt als ‚Black Box' vor, weshalb die Gleichungen 2.3 und 2.4 als Ein- und Ausgangssignale sowie als Zustandsfunktion verwendet werden können. Eingangssignale sind unabhängig vom betrachteten System und werden davon nicht beeinflusst. Ausgangssignale beinhalten den Zustand als systembeschreibende Infor-

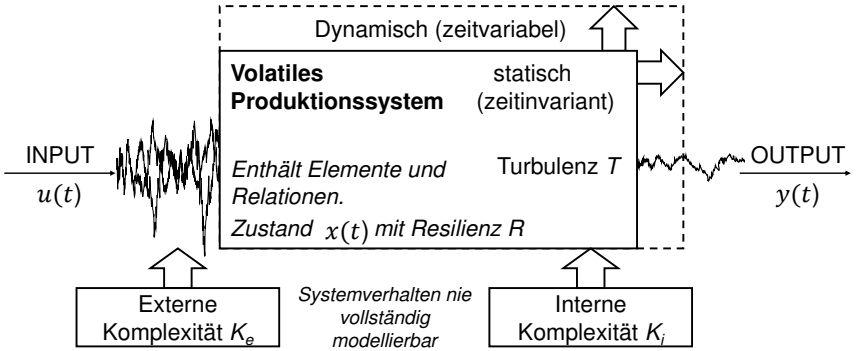


Abbildung 2.4: Übersicht Systemverständnis dieser Arbeit (Böhm et al.)

mationen als Systemantwort. Girod et al. betonen den hohen Abstraktionsgrad, der sich durch die Beschränkung auf die funktionalen Zusammenhänge zwischen Signalen ergibt. Damit wird die Systemtheorie zu einem mächtigen Werkzeug bei der Untersuchung von Produktionssystemen und trägt zu einer interdisziplinären Betrachtungsweise dieser Arbeit bei. Den Black-Box-Charakter greift Hayek auf, wenn er sagt:

„Die statistische Methode ist deshalb nur dort nützlich, wo wir die Beziehung zwischen den individuellen, mit verschiedenen Merkmalen versehenen Elementen entweder bewusst nicht beachten oder nicht kennen [...].“ (Hayek 1972, S. 19)

Der Ansatz, für die Turbulenzprognose die Produktion als signalverarbeitendes System zu betrachten, erfordert mathematische Methoden zur Funktionsbeschreibung und -modellierung. Auf die mathematischen Grundlagen zur Detektion und Beschreibung funktionaler Abhängigkeiten wird folgend in Abschnitt 2.2 eingegangen.

2.2 Funktionale Abhängigkeit

»Jetzt ist es an den Datensammlern, alle Details zu kennen und an den Mathematikern, die Ursachen zu finden. Letztere tun das oft, ohne alle Fakten zu wissen. [...] Denn in der Mathematik geht es um Muster, nicht um das eigentliche Ding.«

(Aus dem Englischen (Bouchier 1901):
Aristoteles - Posterior Analytics, Book 1, Chapter XIII.)

Diese frei übersetzte Passage schlägt die Brücke von komplexen Systemen zu deren Handhabung mit mathematischen Werkzeugen. Ausgehend vom Begriff der Systemtheorie in Abschnitt 2.1 zu komplexen, turbulenten Systemen stellt sich die Frage, wie solche Systeme mathematisch beschrieben und analysiert werden können. Im Fazit des vorausgehenden Kapitels wird die Notwendigkeit zur Betrachtung von Signalen und deren funktionale Zusammenhänge bereits angedeutet. Ziel ist es also, funktionale und damit systembeschreibende Abhängigkeiten zwischen Eingangs- und Ausgangsgrößen zu detektieren. Diese Arbeit nutzt dafür Zeitreihen, die das Verhalten von Systemen beschreiben. In solchen Zeitreihen sind Informationen über die Systemzustände enthalten. Dazu werden in Unterabschnitt 2.2.1 Zeitreihen allgemein sowie Möglichkeiten ihrer Beschreibung und Modellierung dargelegt. Auf die mathematische Handhabung mehrerer solcher Zeitreihen wird in 2.2.2 eingegangen, wobei Korrelation und Latenz von zentraler Bedeutung sind. Als Referenz für die wesentlichen Grundaussagen und Formeln zur mathematischen Systembeschreibung werden die übersichtlichen Erklärungen von Girod verwendet (Girod et al. 1997).

2.2.1 Zeitreihen und Regression

Dieser Abschnitt befasst sich mit der Handhabung von Systemsignalen. Signale beschreiben Größen, die sich über die Zeit ändern. Ein Signal ist also eine Wertefolge, die Informationen beinhaltet. Die Zeit t ist dabei die unabhängige Variable und kontinuierlich. Mit Computern verarbeitete Signale können nur zu bestimmten, einzelnen Zeitpunkten beschrieben werden. Solche Signale heißen *zeitdiskret*. Kontinuierliche Signale können für die digitale Verarbeitung durch Abtastung diskretisiert werden. Neben den Zeitpunkten können auch die Signalwerte (genannt Zustände) kontinuierlich oder diskret, deterministisch oder stochastisch und reell- oder komplexwertig sein.

Schwaninger nennt in seinen Ausführungen *Zeitreihen* ein Mittel der Kontrolltheorie, welches „[...] mathematische Verfahren für die Gestaltung und Optimierung [...] nutzbar macht" (Schwaninger 1996, S. 1953). Eine Zeitreihe ist eine nach dem Zeitindex t geordnete Menge von Zustandsmessungen $(x_t)_{t\in T}$ der Variablen X_t und beschreibt formal eine zeitlich geordnete Folge von Beobachtungen (Leiner 1982; Schlittgen & Streitberg 2001). Dabei ist $T = \{1, 2, ..., N\}$ die endliche, äquidistante, diskrete Menge aller N Beobachtungszeit-

punkte. Die Zeitintervalle zwischen den diskreten Zeitpunkten sind *im Regelfall* äquidistant und die Zeitreihenwerte sind vier unterschiedlichen Arten der Skalierung zuzuordnen:

- Nominalskala: Werte sind nicht in eine Reihenfolge zu bringen
- Ordinalskala: natürliche Rangordnung ohne fest vorgegebene Abstände
- Intervallskala: mit Rangordnung und Abständen; willkürlich festgelegter Nullpunkt
- Verhältnisskala: Intervallskala mit absolutem Nullpunkt; Quotientenbildung sinnvoll

Unabhängig von der Skala werden für Zeitreihen mehrere Hauptanwendungen genannt: Beschreibung, Modellierung, Prognose und Kontrolle. Diese Anwendungen finden sich vorwiegend in den Feldern der Ökonomie, Finanzen, Umwelt und Medizin wieder. Die dabei verwendeten Abtastraten sind anwendungsbezogen unterschiedlich und nehmen dabei im Zuge der technischen Möglichkeiten beim Datentransport und der Speicherung immer weiter zu (z.B. Arbeitslosenstatistik monatlich, Aktienkurs mindestens täglich, Temperaturverläufe stündlich und beim Elektrokardiogramm Raten im Zehntel- bis Hunderts-telsekundenbereich) (Brockwell & Davis 1991). Jedoch gilt für die meisten statistischen Methoden, dass die Daten mit der Verhältnis- oder Intervallskala gemessen werden, die beide gemeinsam „metrisch" genannt werden (Holland & Scharnbacher 2010, S. 6).

Zur Beschreibung von Zeitreihen können *empirische Momente* berechnet werden. Diese sind nur sinnvoll für stationäre Reihen. Solche Reihen beschreiben Objekte, die keiner systematischen Änderung über die Zeit unterliegen. Im Anhang werden die ‚Empirischen Momente' mit Formeln beschrieben, welche für diese Arbeit relevant sind und auf die immer wieder verwiesen wird. Die Formeln sind Grundlagenwerken entnommen und in Abschnitt A.2 aufgelistet (Holland & Scharnbacher 2010; Schlittgen & Streitberg 2001).

Für sowohl die Analyse als auch die Modellierung von Zeitreihen dient die Zerlegung in (Zeit-)Komponenten. Motivation dafür ist, dass die Annahme von stätionären Zeitreihen häufig nicht gilt. Je nach Quelle werden unterschiedliche Bezeichnungen verwendet. Es finden sich Trend-, Konjunktur-, Saison- und Restkomponente sowie die glatte, zyklische und irreguläre Komponente (Leiner 1982; Schlittgen & Streitberg 2001; Holland & Scharnbacher 2010). Eine langfristige systematische Veränderung, die sich entweder aufsteigend, niveauhaltend oder absteigend verhält, wird ‚Trend' genannt. Je nach Weite des betrachteten Zeitraums sind langfristige Schwingungen (Konjunktur) darin enthalten. Da diese

Arbeit volkswirtschaftliche Aspekte und Marktschwankungen prinzipiell zwar berücksichtigt, jedoch nicht modelliert, werden Trend und Konjunktur zur ‚glatten Komponente' g_t zusammengefasst. Die Saisonkomponente s_t modelliert wiederkehrende, periodische Schwankungen, die jedoch nicht zwingend Jahreswerte, Halbjahreswerte, Vierteljahreswerte oder Monatswerte sind. Auch typische Produktivitäts- oder Qualitätsunterschiede je nach Wochentag gehören zu den saisonalen Effekten. Die irreguläre Komponente u_t ist stochastisch, hat den Status einer Zufallsvariablen und beinhaltet die nicht direkt erklärbaren Einflüsse und Störungen. Die Komponenten können additiv (Gleichung A.8) oder multiplikativ (Gleichung A.9) verknüpft werden. Letztere wird durch Logarithmieren zur besseren Darstellung wieder in eine additive Form überführt (Gleichung A.10). Abbildung 2.5 zeigt die Komponenten grafisch.

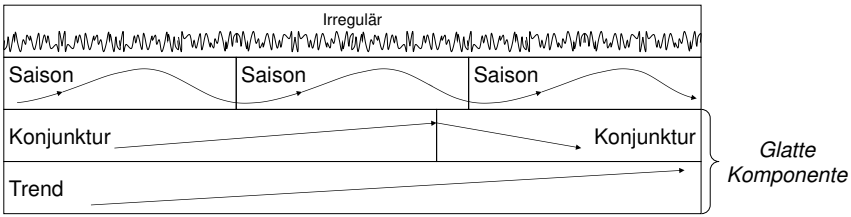


Abbildung 2.5: Komponenten von Zeitreihen

In dem Zusammenhang ist das *geometrische Mittel* $\bar{x}_{geom}$ zu nennen, welches das Wachstum über die Zeit beschreibt (Gleichung A.3) und die durchschnittliche Zuwachsrate angibt. Analog zum gewichteten arithmetischen Mittel berechnet sich das *gewichtete geometrische Mittel* $\bar{x}_{geom\_w}$ (Gleichung A.4). Abbildung 2.6 gibt einen Überblick über die verschiedenen hier verwendeten Mittelwerte.

Neben der Beschreibung von Zeitreihen durch statistische Maße kann es auch notwendig sein, Zeitreihen als kontinuierliche Funktion darzustellen. Eine solche Darstellung ist ein mathematisches Modell der diskreten Zeitreihe und wird *Regression* genannt. Sie bietet so die Möglichkeit, Zeitreihen ökonomisch zu speichern, weitere Werte zu generieren und sie mathematisch zu behandeln. Die hier dargestellten Regressionsverfahren beschränken sich auf die für diese Arbeit notwendigen. Im Kontext von Zeitreihen entspricht Regression der Trendanalyse. In dieser Arbeit wird Regression sowohl direkt für Zeitreihen als auch zur

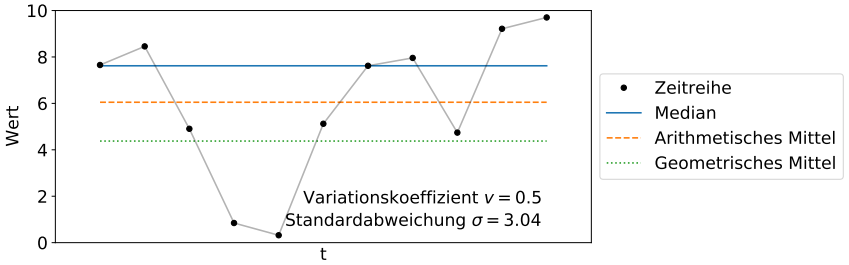


Abbildung 2.6: Diskrete Zeitreihe mit unterschiedlichen Mittelwerten

Beschreibung funktionaler Zusammenhänge zwischen den Zeitreihen verwendet. Die hier zur Trendschätzung genutzten Funktionen sind die lineare Funktion (Gleichung A.11), das Polynom k-ten Grades (Gleichung A.12) und die Exponentialfunktion (Gleichung A.13) mit a_k und b als Parameter (Leiner 1982, 8 f.). Neben den drei hier genannten für Regression genutzten Funktionen sind logistische Funktionen, Splines, der gleitende Durchschnitt und Gompertzkurven weit verbreitete nichtlineare Modellierungsfunktionen (siehe Abbildung 2.7) (Backhaus et al. 2015).

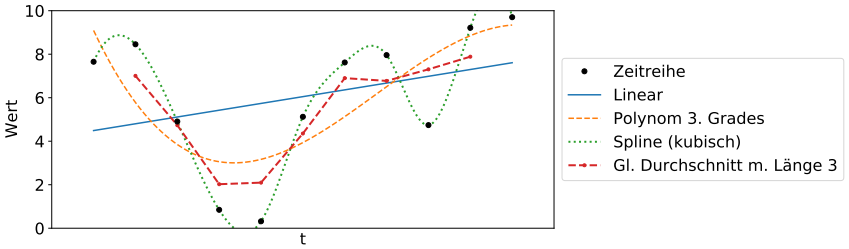


Abbildung 2.7: Vergleich unterschiedlicher Zeitreihenmodellierungen

Dieser Unterabschnitt behandelt Zeitreihen als Gestaltungs- und Optimierungsmittel der Kontrolltheorie. Dazu werden ihre grundlegenden Eigenschaften wie Diskretheit und die gängigen Skalen von Zeitreihen dargelegt. Zu ihrer Beschreibung dienen die sogenannten empirischen Momente wie arithmetisches Mittel und Varianz. Daneben werden weitere für diese Arbeit notwendigen Lage- und Streumaße erklärt. Für die Nutzung von Zeitreihen als Modelle ist es notwendig, sie in ihre Komponenten zu zerlegen. Mit der Zerlegung in glatte,

saisonale und irreguläre Komponenten geht auch die Regression der Zeitreihen einher. Die für diese Arbeit relevanten zeitabhängigen Funktionen werden kurz beschrieben.

Alle hier verwendeten Größen sind zeitabhängig und reellwertig mit nicht begrenztem Definitionsbereich. Bislang wurden nur isolierte Zeitreihen betrachtet. Im Folgenden werden die formalen mathematischen Ausdrücke für die Detektion und Beschreibung von Abhängigkeiten zwischen Zeitreihen dargelegt. Die Formeln werden vorwiegend in der Disziplin der Signalverarbeitung verwendet.

2.2.2 Korrelation und Diskretisierung durch Abtastung

In diesem Unterabschnitt werden die relevanten Gleichungen für die Untersuchung der Abhängigkeiten von Zeitreihenwerten zu unterschiedlichen Zeitpunkten oder zweier unterschiedlichen Zeitreihen beschrieben. Es wird von zwei Zeitreihen der Variablen X_t und Y_t mit den jeweiligen Einzelwerten $(x_t)_{t\in T}$ und $(y_t)_{t\in T}$ aus 2.2.1 ausgegangen. Ihre nichtlinearen Modellfunktionen der deterministischen Signale seien $f(t)$ und $g(t)$. Die allgemeingültigen Formeln sind ebenfalls Grundlagenbüchern entnommen (Schlittgen & Streitberg 2001; Girod et al. 1997).

Die *empirische Kovarianz* c beschreibt die Stärke des linearen Zusammenhangs zwischen den Beobachtungspaaren zweier Zeitreihen mit den Werten x_t und y_t (Gleichung A.14). Der für Vergleiche besser geeignete *Korrelationskoeffizient* r von Bravais-Pearson (Gleichung A.15) berechnet sich durch Normierung der Kovarianz r mit dem Produkt der einzelnen Standardabweichungen σ_x und σ_y der beiden Zeitreihen (siehe Gleichung A.6). Wegen der Normierung gilt $-1 \leq r \leq 1$ (Schlittgen & Streitberg 2001, S. 4).

Liegen keine Annahmen zur Wahrscheinlichkeitsverteilung der Variablen vor, werden parameterfreie Maße der Korrelation verwendet. Sie berücksichtigen nur den Rang (Position in der geordneten Liste) der beobachteten Werte. Der Kendall'scher Konkordanzkoeffizient (auch Kendall'sches Tau) ist robuster gegenüber Ausreißern und wird daher in dieser Arbeit eingesetzt.

Zur Untersuchung von Korrelationen bei zeitlicher Verschiebung um τ wird die Kreuzkorrelationsfunktion verwendet. Sie zeigt Spitzen bei Zeitverschiebungen, die der Signallaufzeit des Signals $f(t)$ zum Signal $g(t)$ entsprechen (siehe Abbildung 2.8). Die Zeitverschiebung

wird auch *Latenz* genannt. Die *Kreuzkorrelationsfunktion* $\varphi_{fg}(\tau)$ zweier deterministischer Signale $f(t)$ und $g(t)$ beschreibt die Verwandtschaft unter Berücksichtigung einer möglichen Verschiebung um τ (Gleichung A.16), mit $g^*(t)$ als die zu $g(t)$ konjugiert komplexe Funktion. Für $f(t) = g(t)$ erhält man die *Autokorrelationsfunktion* der Zeitfunktion $f(t)$. Sie gibt die Ähnlichkeit der Anteile eines Zeitsignals an, die zu verschiedenen Zeitpunkten auftreten.

Die Verknüpfung zweier Zeitfunktionen $f(t)$ und $g(t)$ wird *Faltung* genannt und mit $*$ bezeichnet (Gleichung A.18). Eine Anwendung der Faltung ist der Suchfilter. Dabei werden zwei Funktionen $f(t)$ und $g(t)$ betrachtet, deren zeitliche Verschiebung nicht bekannt ist. Die Kreuzkorrelationsfunktion deterministischer Signale kann damit auch durch einen Faltungsausdruck definiert werden (Gleichung A.19).

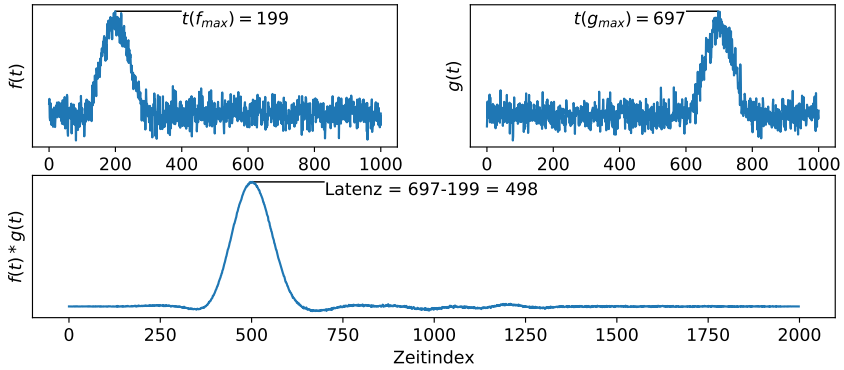


Abbildung 2.8: Detektion der Zeitverschiebung

Die vorangegangenen Gleichungen gelten für kontinuierliche Signale. Zur Speicherung und Verarbeitung auf Digitalrechnern ist es nötig, die Signale abzutasten. Das abgetastete kontinuierliche Zeitsignal $x(t)$ des Zustandes X wird als diskretes Signal $x[k]$ bezeichnet. Mit dem Abstandsintervall T ergibt sich die Zeitquantisierung, beschrieben in Gleichung A.20. f_a wird Abtastrate oder Abtastfrequenz genannt. ω_g ist dabei die Abtastkreisfrequenz gemäß Gleichung A.21. Die Abtastrate gibt an, wie oft das Signal je Zeiteinheit abgetastet wird. $f_a = \frac{\omega_g}{\pi}$ wird kritische Abtastung oder Nyquist-Frequenz genannt. Bei geringeren Abtastraten tritt eine Überfaltung der Spektren auf, genannt Aliasing (Arens et al. 2010, S. 1045; Honerkamp 1990, S. 276).

Die Detektion von Latenzen zwischen diskreten Zeitreihen ist eine wesentliche Voraussetzung für die Bestimmung funktionaler Abhängigkeiten. In der Praxis reicht die Faltung allein dafür meist nicht aus. Gründe dafür sind zu starkes Rauschen im Signal, nicht-lineare Abhängigkeiten und nicht-äquidistante Abstandsintervalle der Datenpunkte. Gerade bei nicht-linearen Abhängigkeiten stoßen Regression und Korrelationsanalysen an ihre Grenzen. Ahrens et al. raten sogar explizit, einen „[...] Blick auf die Punktewolke" zu werfen, „um die Unangemessenheit des gewählten Modells zu erkennen" (Arens et al. 2010, S. 1414). Konkrete, fallbezogene Möglichkeiten zur Bestimmung funktionaler Abhängigkeiten über Korrelation und Faltung hinaus werden im Lösungskapitel 5 beschrieben. Implizite Voraussetzung für die Existenz funktionaler Abhängigkeit ist Kausalität. Dazu und zu weiteren für die Lösung des Problems dieser Arbeit dienlichen Werkzeugen der folgende Unterabschnitt, welches den mathematischen Teil abschließt.

2.2.3 Kausalität, Stationarität und Stabilität

Im Abschnitt zur Systemtheorie wurden die notwendigen Kriterien zur Realisierung von Systemmodellen nicht näher erläutert. Wesentliche Begriffe in dem Zusammenhang sind *Kausalität*, *Stationarität* und *Stabilität* (Girod et al. 1997, S. 374). Zur formalen Exaktheit und für eine stringente Umsetzung des Datenhandlings im Programm werden die Begriffe nun erklärt und festgelegt.

Kausalität

Kausalität ist eine oft als trivial betrachtete Eigenschaft, dabei jedoch eine wichtige Voraussetzung für die mathematische Handhabung von Systemen. Girod et al. nennen Kausalität „[...] einen ursächlichen Zusammenhang zwischen zwei oder mehreren Vorgängen." Damit legt Kausalität nicht nur den logischen Zusammenhang fest, sondern auch die zeitliche Folge. Das Bedeutet im Generellen, dass die Ursache zeitlich vor ihrer Wirkung auftritt. Übersetzt gilt damit für die Systemtheorie: „Eingangssignal vor Ausgangssignal". Dies gelte für kontinuierliche und diskrete Systeme analog. Zwei gleiche in der Vergangenheit vor k_0 liegende Eingangssignale $x_1[k]$ und $x_2[k]$ bewirken also zwei identische Ausgangssignale

$y_1[k]$ und $y_2[k]$. Diese zentrale Aussage wird als mathematische Formulierung in Gleichung 2.6 und Abbildung 2.9 dargestellt.

$$x_1[k] = x_2[k] \text{ für } k < k_0 \Rightarrow y_1[k] = y_2[k] \text{ für } k < k_0 \tag{2.6}$$

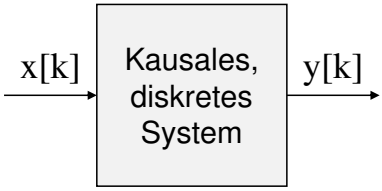


Abbildung 2.9: Kausalität eines diskreten Systems, nach (Girod et al. 1997, S. 375)

Wenn das abgetastete rechtsseitige Signal $y[k]$ frühestens bei $k = 0$ beginnt, wird es als kausales Signal bezeichnet. Für diese Arbeit ist die Definition der Kausalität insofern von großer Bedeutung, da sie als Merkmal dient, um Eingangs- und Ausgangssignale voneinander zu unterscheiden.

Für die Beschreibung realer Systeme kann davon ausgegangen werden, dass die gemessenen Signale nicht immer durch mathematische Funktionen beschreibbar sind. Grund dafür sind Störsignale wie Rauschen oder chaotische Schwingungen. Die Beschreibung solcher Signale erfolgt durch Erwartungswert, Varianz und Stationarität. Die ersten beiden wurden im Unterabschnitt 2.2.1 bereits für diskrete Signale eingeführt und werden hier auf kontinuierliche Signale übertragen. Damit können sogenannte nicht-deterministische Signale betrachtet werden. Nicht-deterministische, stochastische oder Zufallssignale sind solche, deren Verlauf nicht vorhersehbar ist.

Stationarität

Für kontinuierliche Systemmodelle lässt sich die Stationarität ausgehend vom Erwartungswert zweiter Ordnung ausdrücken, was im Folgenden beschrieben wird. Das arithmetische Mittel bei diskreten Messpunkten (siehe Gleichung A.1) heißt bei kontinuierlichen Systemen „linearer Mittelwert“ $\mu_x(t)$. Um zur Signalbeschreibung einen deterministischen Wert zu berechnen, gilt für μ_x mit dem Erwartungswert $E\{x(t)\}$ die Gleichung A.22. Hinzukommen

der Erwartungswert erster Ordnung $E\{f(x(t))\}$ für $x(t)$ zu *einem Zeitpunkt* t (Gleichung A.23). Sollen Abhängigkeiten zwischen *verschiedenen Zeitpunkten* eines Signals erfasst werden, gilt der Erwartungswert zweiter Ordnung. Die Verknüpfung eines Signals an zwei unterschiedlichen Zeitpunkten ist die Autokorrelationsfunktion (A.24). Für diskrete Signale ist die Definition bereits implizit in Gleichung A.16 eingeführt worden. Weiter lassen sich mit dem Mittelwert $E\{x(t)\}$ die Varianz $E\{(x(t) - \mu_x(t))^2\}$ (Gleichung A.26) und damit die Standardabweichung $\sigma_x(t)$ für kontinuierliche Systeme (Gleichung A.27) definieren. Abschließend sind mit der Differenz der Beobachtungszeitpunkte $\tau = t_1 - t_2$ die Autokorrelationsfunktion (A.28), die Autokovarianzfunktion (A.29) und die Kreuzkovarianzfunktion (A.30) festgelegt.

Damit liegen die signalbeschreibenden Momente für die Stationaritätsdefinition vor. Bei einem Signal, dessen statistische Eigenschaften sich mit der Zeit nicht ändern, ändert sich auch der Erwartungswert zweiter Ordnung nicht, wenn man t_1 und t_2 um den gleichen Wert Δt verschiebt. Die Definition lautet: „Ein Zufallsprozess heißt stationär, wenn seine Erwartungswerte zweiter Ordnung nur von der Differenz der Beobachtungszeitpunkte $\tau = t_1 - t_2$ abhängen." (Girod et al. 1997, S. 420). Übertragen auf Systeme bedeutet das, dass solche, deren statistische Eigenschaften sich nicht mit der Zeit ändern, als stationär bezeichnet werden. Die Definition wird in Gleichung 2.7 formal beschrieben.

$$E\{f(x(t_1), x(t_2))\} = E\{f(x(t_1 + \Delta t), x(t_2 + \Delta t))\} \tag{2.7}$$

Das Systemverhalten im Hinblick auf die Erfüllung der Stationaritätsbedingung ist ein zentraler Aspekt dieser Arbeit. Nicht-stationäre Systeme sind statistisch entsprechend schwerer zu erfassen und bedingen anderer mathematischer Handhabung. Die Erfüllung der Stationaritätsbedingung hat entsprechenden Einfluss auf die Turbulenzprognosen.

Stabilität

Offensichtlich ändern sich auch in nicht-stationären, also dynamischen Systemen die Zustände ihrer Elemente über die Zeit. Es stellt sich die Frage, ob die Ausgangssignale dennoch innerhalb nicht-unendlicher Grenzen liegen. Betrachtet werden für die Definition der Stabilität lineare, zeitinvariante Systeme (LTI-Systeme), da sie in der Systemtheorie eine wichtige Rolle einnehmen (Girod et al. 1997, S. 10). Die gängige Definition für Stabilität eines LTI-Systems ist die BIBO-Stabilität. BIBO steht dabei für Bounded Input → Bounded Output

und referenziert auf die oben angesprochenen Grenzen, innerhalb derer sich bei stabilen Systemen Ein- und Ausgangssignale bewegen. Die Definition lautet:

„Ein zeitkontinuierliches oder zeitdiskretes LTI-System heißt stabil, wenn es auf jede beschränkte Eingangsfunktion $x(t)$ oder Eingangsfolge $x[k]$ mit einer beschränkten Ausgangsfunktion $y(t)$ oder Ausgangsfolge $y[k]$ reagiert."

Das bedeutet für den kontinuierlichen Fall, dass die Impulsantwort absolut integrierbar und für den diskreten Fall, dass die Impulsantwort absolut summierbar sein muss (Girod et al. 1997, 390 f.). Da hier reale Produktionssysteme vorliegen, wird auf weitere Stabilitätskriterien wie die über Singularitäten der Systemfunktion mit Pol-Nullstellendiagrammen verzichtet. Solche sind relevant, wenn Systeme mathematisch modelliert werden. Die hier betrachteten Produktionssysteme sind aufgrund des physikalischen Gesetzes der Energieerhaltung prinzipiell stabil (Girod et al. 1997, S. 389). Dennoch sind je nach Systemgrenze instabile Zustände denkbar. So würde beispielsweise bei konstanter Auftragsfreigabe als Eingangssignal $x(t)$ bei permanenter Kapazitätsüberlastung das Ausgangssignal des Work in Process (WIP) für $T \to \infty$ unendlich groß werden und damit nicht mehr innerhalb endlicher Grenzen liegen. In der Realität würde zuvor jedoch entweder die Kapazität oder eben die Auftragsfreigabe angepasst werden, wodurch das Ausgangssignal nicht unendlich wird und das Stabilitätskriterium wieder erfüllt ist. Daher genügt für diese Arbeit die systemtheoretische Beschreibung der Stabilität von Schwaninger: „Stabilität eines dynamischen Systems liegt vor, wenn das durch eine Störung aus dem Gleichgewichtszustand gebrachte System wieder in diesen zurückkehrt." (Schwaninger 1996, S. 1953)

Fazit:

Der Abschnitt *Funktionale Abhängigkeit* enthält die formal-mathematischen Grundlagen zur Beschreibung von Systemen mittels Zeitreihen. Die Werkzeuge dienen dazu, Produktionsdaten als Zeitreihen korrekt zu verarbeiten, richtig zu interpretieren und allgemeingültig zu behandeln. Dazu wurde der Zeitreihenbegiff zunächst eingeführt und sowohl für kontinuierliche als auch für diskrete Modellierungen die notwendigen empirischen Momente hergeleitet und definiert. Mit Bezug zur Systemtheorie wird das Produktionssystem als kausales, diskretes System mit den Ein- und Ausgangsfunktionen $x[k]$ und $y[k]$ behandelt. Um funktionale Abhängigkeiten *zwischen* Ein- und Ausgangsfunktion zu detektieren, wurde Faltung und Korrelation definiert und beispielhaft veranschaulicht. Die Faltung dient

als eine Möglichkeit dazu, die Latenz zwischen zwei Signalen zu bestimmen, damit Ein- und Ausgangssignal zu unterscheiden und anhand der dann bekannten Zeitverschiebung die richtigen Datenpunkte für die Korrelationsanalyse gegenseitig zuzuordnen. Sowohl die Zeitreihen selbst als auch der funktionale Zusammenhang können mittels Regression modelliert werden, wozu die gängigen Methoden vorgestellt wurden. Abschließend wurden noch die zentralen Kriterien Kausalität, Stationarität und Stabilität für die Realisierung von Systemen erläutert. Eine Möglichkeit, wie aus Zeitreihen mittels maschinellem Lernen Wissen generiert werden kann, wird im folgenden Abschnitt behandelt.

2.3 Methoden des maschinellen Lernens

„Während es einerseits gewiss wünschenswert ist, unsere Theorien so falsifizierbar wie möglich zu machen, müssen wir anderseits in Gebiete vorstoßen, in denen, wenn wir vordringen, der Grad der Falsifizierbarkeit notwendigerweise abnimmt. Das ist der Preis, den wir für ein Vordringen in das Gebiet der komplexen Phänomene zu zahlen haben."
(Hayek 2007, S. 196)

Eine Ausprägung der hier von Hayek angedeuteten Entwicklung der Wissenschaften in Richtung nicht vollständig falsifizierbarer Bereiche ist die Etablierung des als ‚Künstliche Intelligenz (KI)' bezeichneten Konzeptes mit seiner Disziplin ‚Maschinelles Lernen (ML)'. Dieser Abschnitt behandelt den Themenkomplex KI in drei Schritten. Der erste Teil in 2.3.1 blickt zunächst knapp auf die historische Entwicklung der KI mit ihren unterschiedlichen Ausprägungen und enthält eine Zusammenstellung der relevanten Bezeichnungen. In 2.3.2 wird auf ML näher eingegangen, die für diese Arbeit relevante Lernaufgabe der Klassierung zur Segmentierung abgegrenzt und spezifiziert, sowie eine Übersicht zu möglichen Lernverfahren des ML dargelegt. 2.3.3 erklärt abschließend das formalwissenschaftliche Konzept der Entscheidungsbäume.

2.3.1 Überblick zur Historie und relevante Terminologien

In populärwissenschaftlichen Veröffentlichungen werden die Begriffe KI, ML und deren Unterkategorien unscharf voneinander abgegrenzt und nicht präzise verwendet. KI ist

das Fachgebiet, welches die Möglichkeiten erforscht, Rechenmaschinen das ‚Denken' beizubringen. Um das zu erreichen, wurde innerhalb der Informatik ein Paradigmenwechsel des Programmierens vollzogen. Die klassische Programmierung implementiert Regeln und berechnet anhand von Eingabedaten Antworten. Das neue Paradigma hingegen nutzt Daten und aus der Historie bekannte Antworten, um Regeln zu generieren (Chollet 2018, S. 23). Dieser Zweig der Informatik wird ML genannt. Er ist an der Schnittstelle zwischen Informatik, Statistik und Ingenieurswissenschaft angesiedelt (Dangeti 2017, S. 8). ML ist definiert als ein automatischer Prozess, der Muster aus Daten extrahiert (Kelleher et al. 2015, S. 43). Die drei typischen Kategorien sind überwachtes, unüberwachtes und bestärkendes Lernen. Die überwachten Lernstile adressieren Regressions- und Klassierungsaufgaben. Unüberwachte Lernstile werden für Clusterung und Dimensionsreduktion eingesetzt. Beim bestärkenden Lernen versuchen lernende Agenten ihren Gewinn durch Aktionsvariation zu maximieren. Der Gewinnbeitrag erfolgt über Feedback aus der Umgebung. Auf der Grundlage dieses Feedbacks evaluiert der Agent seine Aktion und passt sie gegebenenfalls an (Dangeti 2017, S. 9). Neben dem ML gibt es einige weitere Algorithmengruppen, die je nach Interpretation auch der KI hinzugerechnet werden können. Die unscharfe Grenze wird in Abbildung 2.10 durch die gestrichelte Linie markiert. Enthalten sind zudem die Gruppen der probabilistischen, wissensbasierten und heuristischen Algorithmen mit offenen Listen konkreter Implementierungen. Im Zentrum dieser Arbeit stehen die Algorithmen-Gruppen des ML. Dazu sind die wesentlichen Vertreter Entscheidungsbäume, die davon als Ensemble-Methode erstellten Random Forests, Support Vector Machine (SVM), Bayessche Modelle und Neuronale Netze. Auf sie wird im folgenden Unterabschnitt eingegangen. Eine Zusammenschau zur historischen Entwicklung von den ersten Computerkonzepten bis zu den aktuellen Algorithmen findet sich im Anhang A.3.

In engem Zusammenhang mit ML steht der Begriff ‚*Big Data*'. Je nach Quelle werden Datenmengen, welche drei (Chen et al. 2012), vier (Marr 2015), fünf (Dehghantanha 2019) oder sechs (Gandomi & Haider 2015) „V's" erfüllen als Big Data bezeichnet. Dazu zählen auf Englisch Volume, Variety, Velocity, Veracity, Variability und Value. Die ersten drei sind direkt messbare äußere Attribute. So sind Datenmengen mit großem Umfang und Heterogenität, welche in hoher Datenrate zu verarbeiten sind, entsprechend als Big Data zu bezeichnen. Die letzten drei V's beziehen sich auf den inhärenten Teil der Daten.

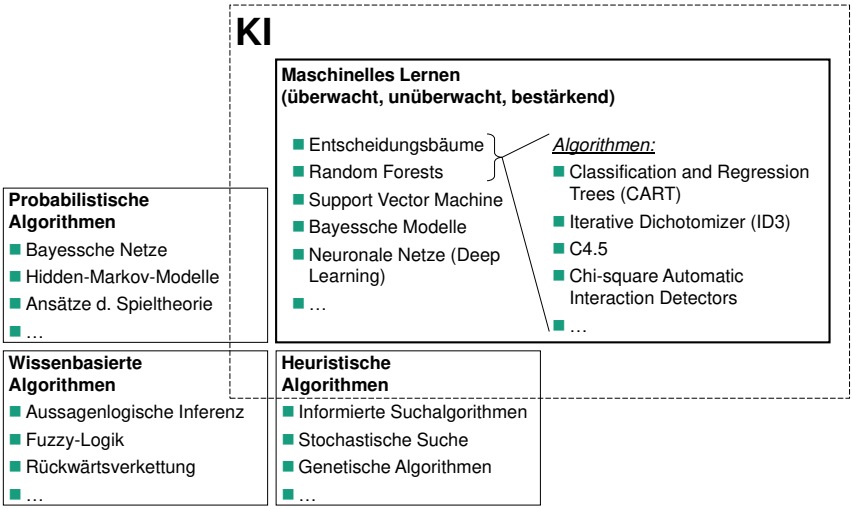


Abbildung 2.10: Übersicht Künstliche Intelligenz in Anlehnung an (Chollet 2018, S. 22; Bitkom 2019, S. 8)

Als Veracity gilt nach der Definition von IBM der Wahrheitsgehalt der Daten. Dieser variiert stark bei von Menschen generierten Daten wie beispielsweise in sozialen Netzwerken. Variability ist die Veränderlichkeit der Datenherkunft und der zu verarbeitenden Datenrate und wird oft mit Komplexität der Datenverarbeitung gleichgesetzt. Als letztes steht Value für die Informationsdichte. Sie kann als extrahierbares Wissen je Datenmenge oder je Zeit betrachtet werden. Damit können auch Informationsdichten mit größer werdender Datenmenge steigen, wenn dadurch mehr Wissen generiert werden kann (Gandomi & Haider 2015, S. 139). Alle Merkmale haben den gemeinsamen Nachteil, dass sie zeit- und fallabhängig unterschiedlich bewertet werden. Datenmengen, welche heute von einem durchschnittlichen Smartphone verarbeitet werden können, waren vor wenigen Jahren noch eine technische Herausforderung. Die Definition mit hohem allgemeinen Gültigkeitswert, die in der Literatur eher implizit genannt wird ist die, dass die Grenze zu Big Data immer dort verläuft, wo gegenwärtig die technologischen Möglichkeiten die Handhabung gerade noch zulassen. In diesem Verständnis können die in dieser Arbeit verwendeten Datenmengen nicht in ihrer Gesamtheit als Big Data bezeichnen werden. In der industriellen Anwendung des Ergebnisses jedoch könnten Big-Data-Problematiken

schnell auftreten. Für die Umsetzung sind dann geeignete Verarbeitungsstrukturen wie beispielsweise die Lambda-Architektur zu nutzen (Marz 2016).

Mit der Einführung in das Themenfeld KI und den wichtigsten damit zusammenhängenden Begrifflichkeiten wurde in diesem Abschnitt ML soweit dargelegt, dass die wesentlichen Methoden in ihren Eigenschaften in dem Maße bekannt sind, um später im Lösungsteil eine gezielte Auswahl treffen zu können.

2.3.2 Maschinelles Lernen und Klassierung

Maschinelles Lernen ist omnipräsent. Bildverarbeitung in der Medizin, Spracherkennung, Google-Suchvorschläge und Spam-Filter sind nur wenige Beispiele, in denen Algorithmen ihnen bis dahin unbekannte Datensätze verarbeiten. Ihre Gestalt entsteht mit Hilfe von Lernprozessen, in denen Regeln nicht direkt vorgegeben, sondern anhand von Trainingsdatensätzen selbständig ‚erlernt' werden. Maschinelles Lernen hat sich als eine Disziplin der Informatik etabliert und beinhaltet verschiedene Techniken des überwachten und nicht überwachten Lernens. Seine Werkzeuge befassen sich mit Problemen der automatischen oder halbautomatischen Analyse von komplexen Daten (Criminisi & Shotton 2013). Neben der in der Übersicht des vorausgehenden Unterabschnitts genannten gibt es einige weitere Typen des maschinellen Lernens, die auch als Lernstile bezeichnet werden (Ayodele 2010, S. 3; Monostori 2002, S. 120; Shalev-Shwartz & Ben-David 2014). Die gängigsten sind hier aufgelistet:

- Supervised learning: Inputdaten werden auf einen Lösungsraum abgebildet. Anhand eines Trainingsdatensatzes ist die Abbildungsfunktion bekannt.
- Unsupervised learning: Ohne das Wissen über die korrekte Antwort oder Feedback werden Muster in Datensätzen gesucht.
- Semi-supervised learning: Lerntyp, der gelabelte Trainingsdaten mit Mustern kombiniert, die in ungelabelten Daten gefunden wurden.
- Reinforcement learning: Nicht der korrekte Output-Wert ist während der Lernphase bekannt, sondern nur ein Belohnungswert.

- Transduction: Diese Erweiterung des supervised learnings generiert selbständig Lerndaten.
- Learning to learn: Die Methode nutzt Einschätzung des eigenen Bias zur Verbesserung des gelernten Modells.

Ausprägungen des ML unterscheiden sich in den jeweils adressierten Problemen. Die klassischen sind Klassierung, Clusterung, Regression, Dimensionsreduktion und Dichteschätzung. Bei letzterem wird versucht, in beobachtbaren Signalen nicht beobachtbare Dichtefunktionen abzuschätzen. Das ist ein Teilaufgabenbereich der später besprochenen Clusterung. Dimensionsreduktion hat den Zweck, Daten mit vieldimensionalen Räumen auf einen Raum mit geringer Dimensionsanzahl zu reduzieren, ohne die relevanten Eigenschaften der Daten zu verlieren. Die Ansätze sind in ‚linear' und ‚nichtlinear' einzuteilen und können beispielsweise für die Feature-Auswahl in der Vorverarbeitung von Daten für andere ML-Probleme genutzt werden. Zur Regression wurden in Unterabschnitt 2.2.1 die analytischen Methoden bereits vorgestellt. Mittels ML kann die funktionale Modellierung von Datenpunkten erfolgen. Klassierung und Clusterung hängen allein begrifflich bereits eng zusammen. Hinzu kommen noch Klassifikation oder Klassifizierung, welche oft unsauber getrennt und manchmal synonym verwendet werden. In dieser Arbeit wird daher die Terminologie aus Abbildung 2.11 verwendet. Klassifizierung beschreibt den Vorgang der Identifikation und Definition der Klassen. Das Ergebnis der Klassifizierung wird Klassifikation genannt. Die in dieser Arbeit relevante Zuordnung von Objekten zu Klassen wird als Klassierung bezeichnet. Zur besseren Unterscheidbarkeit wird für die Klassifizierung in dieser Arbeit der Begriff ‚Clusterung' genutzt (Eckstein 2014, S. 29).

Die kriterienbasierte Prognose von Kennzahlen ist ein Klassierungsproblem. Daher kommen verschiedene *Lernverfahren* in Frage. In Tabelle 2.1 werden die gängigen Klassierer im Hinblick auf die auswahlrelevanten Eigenschaften (Hengen et al. 2004, A28) gegenübergestellt[1].

Zunächst zu den Eigenschaften: Je nach Klassierungsproblem kann festgestellt werden, dass Trennlinien zwischen Klassen nichtlinear verlaufen oder nicht. Die Gestalt des Datensatzes selbst hat wesentlichen Einfluss auf die Leistungsfähigkeit von Klassierern. Diese

[1] Weitere Eigenschaften entlehnt von: https://dzone.com/articles/logistic-regression-vs-decision-tree und https://tinyurl.com/fwdbtbjb

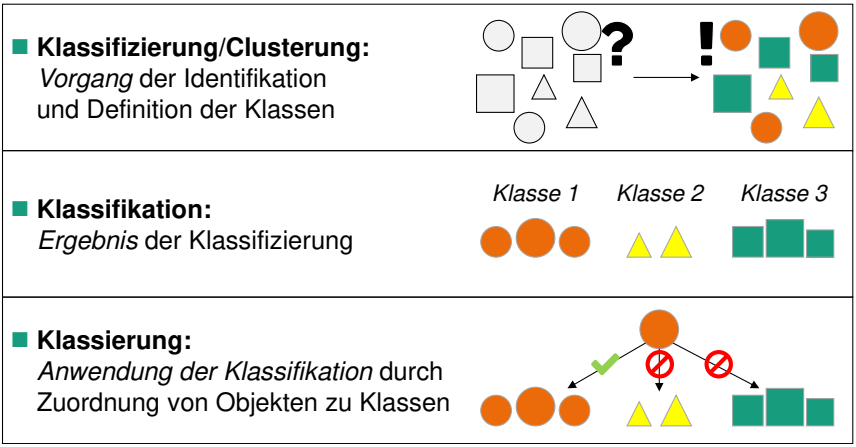


Abbildung 2.11: Klassifizierung, Klassifikation und Klassierung

unterscheidet sich, je nachdem ob kategoriale Daten auf einer Nominalskala (siehe 2.2.1) oder kontinuierliche Zahlenwerte vorliegen. Weiter beeinflusst die Verteilung der Gewichtung einzelner Klassen im Trainingsdatensatz abhängig vom Ansatz unterschiedlich stark das Modell, was gegebenenfalls Anpassungen erfordert. Die Robustheit gegenüber Ausreißern und Datenlücken ist unterschiedlich stark ausgeprägt. Manche Verfahren können Datenlücken nicht kompensieren und verhalten sich hochsensibel gegenüber einzelnen Datenpunkten mit großer Abweichung vom Mittelwert. Gerade im wissenschaftlichen Kontext ist die statistische Validierbarkeit von besonderer Bedeutung, eng verknüpft mit der Erklärbarkeit, also der Möglichkeit, das Zustandekommen des Klassierungsergebnisses oder der Prognose nachzuvollziehen.

Methoden im Vergleich

Die betrachteten Klassierer sind zunächst Logistische Regression (LR), die bereits genannte SVM und das als Naive Bayes (NB) bezeichnete Verfahren. Alle Klassierer, die auf den Classification and Regression Trees (CART)-Algorithmus zu Entscheidungsbäumen von Breimann zurückzuführen sind, werden hier als ‚tree-based' zusammengefasst. Als einziger nicht-überwachter Lernstil werden noch Künstliche Neuronale Netze (kNN) betrachtet, welche für die Klassierungsaufgabe potenziell ebenfalls in Frage kommen. Die wesentlichen Unterschiede und die speziellen Eigenschaften der einzelnen Klassierer werden hier knapp

wiedergegeben. Das dient als Entscheidungsgrundlage für die Auswahl eines geeigneten Verfahrens.

LR ist besonders geeignet für Probleme mit kontinuierlichen Datentypen und bei linear modellierbaren Grenzen zwischen den Klassen. Sie ist nach Umfragen mit die meist verwendete Methode, hat jedoch Nachteile in der Robustheit bei unreinen oder nicht vorprozessierten Datensätzen (Döbel et al. 2018, S. 11). Der Regressionsbegriff ist hier jedoch irreführend, denn LR gehört zu den klassischen Klassifizierungsalgorithmen und wurde schon vor der Verfügbarkeit der Computertechnologie angewandt. Heute wird er in der praktischen Anwendung oft initial für noch gänzlich unbekannte Datensätze verwendet, um einen „ersten Eindruck" für die Klassifizierungsaufgabe zu erhalten (Chollet 2018, S. 36)

SVM gilt als klar definiertes Framework der Algorithmengruppe der Kernel-Methoden, deren Prozesse gut nachvollziehbar sind. Nicht-lineare SVMs benötigen jedoch große Trainingsdatensätze und viel Rechenkapazität (Criminisi & Shotton 2013, S. viii). Dabei sucht der Algorithmus nach der Entscheidungsgrenze zwischen zwei oder mehreren Kategorien. Die Entscheidungsgrenze ist im zweidimensionalen Fall eine Linie, für höherdimensionale Probleme entsprechend eine Hyperebene. Da dies sehr rechenaufwändig ist, wird der namensgebende *Kernel-Trick* genutzt, welcher effizient Abstände von Punktepaaren im Raum berechnet (Chollet 2018, S. 38). Aus der soliden theoretischen Basis zu SVMs folgt eine hohe Erklärbarkeit und gute Validierbarkeit.

Der **NB**-Klassifikator gehört zu den probabilistischen Modellen und beruht auf dem Satz von Bayes. Dieser setzt stochastische Unabhängigkeit der Eingabedaten voraus, welche die namensgebende Annahme ist (Chollet 2018, S. 35). Überlappende Zustandsräume sind jedoch bei Signalen realer Systeme wahrscheinlich, weshalb diese Annahme oft ein schwer zu widerlegender Kritikpunkt am Modell ist. NB zeichnet sich durch vergleichsweise kurze Rechenzeiten aus (Onan et al. 2016, S. 246).

kNN bestehen aus in Reihe geschalteten Repräsentations-Layern. Diese stellen mehrstufige Operationen zum Extrahieren der relevanten Informationen dar. Die Struktur jedes Layers wird durch die ‚Gewichtung ' ihrer Eingabewerte definiert. Je nach Gewichtung werden Eingaben (wie etwa Bilder) Zielen (ein bestimmtes Label) zugeordnet. Die Gewichtung der einzelnen Layer-Elemente muss erlernt werden. kNN werden klassisch auf Eingangsdaten

angewendet, welche in ihren Kategorien homogen sind – ähnlich den Bilddaten oder Signalen zur Spracherkennung.

Entscheidungsbäume („**tree-based**") bestechen durch hohe Transparenz, weil sie sich leicht visualisieren und damit interpretieren lassen. Sie spielen eine zentrale Rolle im Bereich der Explainable Artificial Intelligence (XAI) (Burkart & Huber 2021; Barredo Arrieta et al. 2020). Dazu sind sie sehr robust gegenüber heterogenen, imperfekten Datensätzen, was sie universell einsetzbar macht – sowohl für Geometrieerkennung (Amit & Geman 1997) als auch und für kategoriale Daten (Chollet 2018, S. 39). Chollet nennt die tree-based Algorithmen die „fast immer zweitbesten Algorithmen" und spielt so auf die universelle Einsetzbarkeit von Entscheidungsbäumen an.

Jüngere Entwicklungen

Die Grundprinzipien der beschriebenen Methoden existieren seit Jahrzenten und entwickeln sich in den Details weiter. Heute verfügbaren Rechenleistungen ermöglichen die Realisierung bislang nur theoretisch gelöster Aufgaben. Das Sprachmodell ChatGPT nutzt aktuell 175 Milliarden trainierte Parameter (Ouyang et al. 2022, S. 2). Neben der weiter wachsenden Größe von verarbeitbaren Trainingsdatensätzen liegt auch Potenzial in der Gestaltung der Layerstruktur künstlicher neuronaler Netze. Ein Beispiel aus der Gruppe der tiefen neuronalen Netzwerke ist das Convolutional Neural Network (CNN). Namensgeber ist die mathematische Operation ‚Convolution'. CNNs haben neben den Convolutional Layer weitere unterschiedliche Layer, darunter die Non-Linearity Layer, die Pooling Layer und die Fully-Connected Layer. Vorteile liegen hier bei Anwendungen, die Bildmaterial verarbeiten (Albawi et al. 2017) und kontextbezogene Fragen in Textform beantworten sollen.

Für den Schritt von der schwachen (narrow oder weak) zur starken (broad oder general) KI wird aktuell an neuro-symbolischer KI gearbeitet. Dabei wird nicht versucht, ein großes, universelles neuronales Netz zu generieren, sondern Bild- und Spracherkennung funktional zu entflechten. Hier spielen symbolische Programme wieder verstärkt eine Rolle (Han et al. 2019). Auf diese Weise gewinnt KI an Transparenz – ein Beitrag zum Feld der ‚explainable KI' (Sovrano et al. 2021, S. 2). So betrachtet hat die symbolische KI auf Fortschritte der subsymbolischen KI ‚gewartet', um in neue Bereiche vorzustoßen (Silva et al. 2011).

Kritische Diskussion

Ob diese Entwicklungen den Schritt zur starken KI ermöglichen und was diese Verfahren leisten können ist kritisch zu diskutieren. Klar ist, dass die sogenannte starke KI mit revolutionären Konsequenzen heute nicht existiert und der Punkt der Singularität, bei der KI grundlegendes ändert, noch nicht erreicht ist. Die beschriebenen Anwendungsfälle nutzen jeweils solche Methoden, welche speziell für die jeweiligen Aufgaben gestaltet wurde. Die in dieser Arbeit eingesetzte und vergleichsweise simple Bibliothek zur Erstellung von baumbasierten Klassierern bietet Konfigurierbarkeit in 19 unterschiedlichen Parametern. Das nötige Wissen über die Bedienung der Methode lässt an der ‚Intelligenz' zweifeln und spricht stärker für die Eigenschaften eines zu bedienenden Werkzeuges. Auch für das Design der Layer-Struktur in neuronalen Netzen gibt es bislang keinen definierten Gestaltungsprozess und wird bis heute auf der Grundlage von Erfahrung, Problemverständnis und mit der Trial-and-Error-Methode durchgeführt. Dies bedeutet, dass die Entwicklung inkrementell und alles andere als revolutionär voranschreitet. Mit den Rechenleistungen werden natürlich immer neue Anwendungen ermöglicht, diese sind aber weiterhin auf ihren jeweiligen Anwendungsbereich begrenzt. So ist beispielsweise ChatGPT trotz seiner Modellgröße immer zeitlich auf seinen Trainingsdatensatz beschränkt und sein Faktenwissen ‚hinkt' der Gegenwart aktuell Monate hinterher. ‚KI'-Anwendungen wie Schach-Engines, Chat-Bots oder Bilderkennung zeigen, wie ‚beschränkt' die Technologie bislang noch ist. Die oben beschriebene neuro-symbolische KI zeigt jedoch, welche Potenziale in der Kombination der Methoden liegt. Was passirt, wenn KI damit beginnt, KI zu generieren? Da es bislang so scheint, als dass die Verfahren aktuell nicht über die Hürde ihres jeweilig zugedachten Anwendungsfalles gelangen, kann man zwar von künstliche erzeugtem intelligenten Verhalten sprechen – doch Lem nach noch lange nicht von künstlichem Verstand (Lem 2003, 116 ff.).

Die für obige Beschreibung genutzten Quellen ergeben das Bild, welches in Tabelle 2.1 dargestellt ist. Der Füllgrad in den einzelnen Kreisdarstellungen beschreibt auf einer Skala von Null bis Fünf die Fähigkeit, mit der die Klassierungsmethoden mit Anforderungen des Anwendungsfalls umgehen können. An dieser Stelle erfolgt nun eine vorgreifende Begründung für die Auswahl der baumbasierten Klassierer. Die Systemanforderungen für die Gesamtlösung in dieser Arbeit werden in Kapitel 4 in der Tiefe hergeleitet. Für die

Tabelle 2.1: Vergleich der gängigen Klassierungsmethoden

	LR	SVM	Naive Bayes	kNN	Tree-based
Nichtlineare Trennlinie	○	●	●	●	●
Kategoriale Daten	◔	◑	◑	◔	●
Inhomogene Klassengewichte	◕	◕	◑	◑	◕
Robustheit (Ausreißer)	◔	◑	◑	◕	●
Robustheit (Lücken)	◑	◔	◔	◔	●
Recheneffizienz	●	◕	●	◑	◔
Statistische Validierbarkeit	◕	●	◕	◕	●
Erklärbarkeit	◕	●	●	◑	●

Methodenauswahl sind dabei die in Abbildung 4.2 beschriebenen Anforderungen an den Klassierer relevant. Diese sind Bedienbarkeit, Robustheit und Aktualität, Richtigkeit und Transparenz.

Der **Bedienbarkeit** dienen die Datenhandhabung mit bekannten Tabellenprogrammen, strukturierte Datensätze und einfache Import- und Exportfunktionen. Die hier verwendeten Produktionsdaten in Kennzahlenform sind entsprechend strukturiert und erfordern daher nicht zwingend Deep Learning. Entscheidungsbäume, Support-Vector-Maschinen oder logistische Regression sind für strukturierte Datensätze besonders geeignet. Die Methoden Random Forests, Gradient Boosting und Deep Learning bieten bei größeren Datenmengen Vorteile in der Vermeidung von Over-Fitting. Die so erhöhte **Robustheit** der baumbasierten Methoden erstreckt sich zudem auf die Sensitivität gegenüber Ausreißern. Bei begrenzten Rechenressourcen sind Deep-Learning-Modelle im Nachteil und erschweren dadurch Entwicklung und späteren Einsatz in der Industrieumgebung. Eine hohe **Aktualität** – im besten Falle durch Echtzeitprognosen – ist mit Random Forests einfacher zu erzielen als mit rechenaufwändigen Modellen. Die Richtigkeit ergibt sich durch Vermeidung des Over-Fitting, Kontrolle des Lernprozesses und der Modell-**Transparenz**. Die ist im wissenschaftlichen Kontext dieser Arbeit von zentraler Bedeutung. Für Interpretier- und Erklärbarkeit haben Entscheidungsbäume und logistische Regression ihre Vorteile. Die erschwerte Interpretation des Verhaltens von Deep Learning-Modellen sind ein eigenes Forschungsfeld und würde den Fokus von den Forschungsfragen ablenken.

Die Diskussion der einzelnen Vorteile bestätigt das Rechercheergebnis aus Tabelle 2.1. Hinzu kommt der bei kNN-Methoden nicht direkt erfüllte, jedoch aus den Systemanforderungen abgeleitete Transparenzbedarf. Daher wird für die Lösung des in dieser Arbeit vorliegenden Problems die Methodengruppe der baumbasierten Klassierer gewählt und im folgenden Unterabschnitt theoretisch genauer betrachtet.

2.3.3 Entscheidungsbäume

Mit Entscheidungsbäumen können Regelwerke graphisch dargestellt werden, welche mehrdimensionale Eingabewerte mittels eines Modells einem Ausgabewert zuordnen. Dafür wird der Raum der Eingabeparameter in Regionen aufgeteilt. Die Menge aller Punkte innerhalb dieser Region ergeben wiederum ein Modell, welches im Falle, dass der Eingabewert Teil dieser Menge ist, den Eingabewert genau einem Ausgabewert zuordnet. Auf dieser Weise können Regeln zur Lösung mehrdimensionaler Entscheidungsprobleme als binärer Baum dargestellt werden. Abbildung 2.12 zeigt den von x und y aufgespannten zweidimensionalen Eingaberaum mit den Regionen, die alle Punkte innerhalb der Region einer Klasse – hier beispielhaft für fünf Tierklassen – zuordnet. Für drei Dimensionen ist eine Darstellung der Regionen mit Würfeln noch möglich. Bei höherdimensionalen Problemen enthält der Baum neben x und y weitere Entscheidungskriterien und weist eine entsprechend höhere *Tiefe* auf. Die Entscheidungspunkte entlang eines Pfades werden *Knoten* genannt, die Endknoten *Blätter*. Der Ansatz geht auf Arbeiten von Breimann et al. in den 1980er Jahren zurück, in welchen die CART-Algorithmen ausgearbeitet wurden (Breiman 1998).

Um einen Entscheidungsbaum aufzustellen, wird die ‚Node Impurity' für jedes Kriterium berechnet. Gesucht wird das Kriterium mit dem geringsten Impurity-Wert. Es gibt dafür mehrere Möglichkeiten. Die verbreitetsten Berechnungsmethoden sind ‚Gini ' und ‚Entropy' (Breiman 1998; Wu et al. 2007). Entropy eignet sich besser für Kriterien der Nominalskala, Gini für alle anderen bei durchschnittlich leicht erhöhten Rechenzeiten[2]. Je kleiner die Node Impurity, desto eindeutiger teilt ein Kriterium in die richtigen Klassen ein. Die Gleichungen 2.8 und 2.9 sind die Berechnungsvorschriften für die in Abbildung 2.13 gezeigten Kurven (Bishop 2009, S. 666).

[2] Anmerkung: Die Unterschiede sind fallabhängig und marginal. Internetquelle zur Aussage: http://paginas.fe.up.pt/ ec/files_1011/week%2008%20-%20Decision%20Trees.pdf

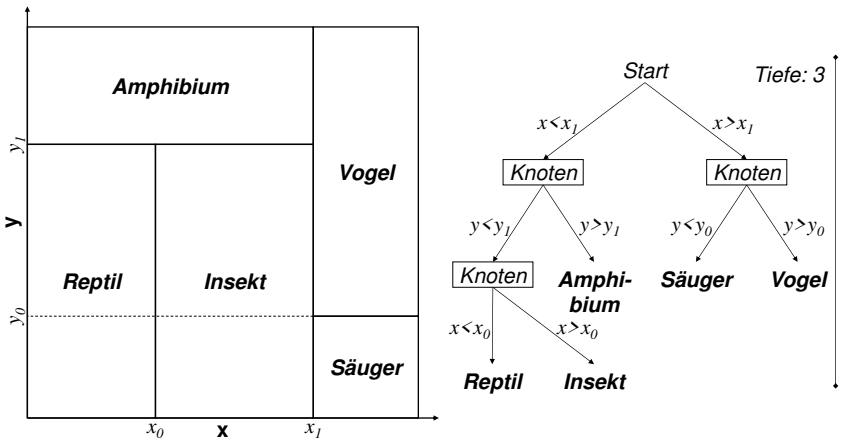


Abbildung 2.12: Zweidimensionaler Eingaberaum mit fünf Klassen und zugehöriger binärer Baum

$$Gini = 1 - \sum_{i=1}^{n} (p_i)^2 \tag{2.8}$$

$$Entropy = -\sum_{i=1}^{n} p_i \cdot \log_2(p_i) \tag{2.9}$$

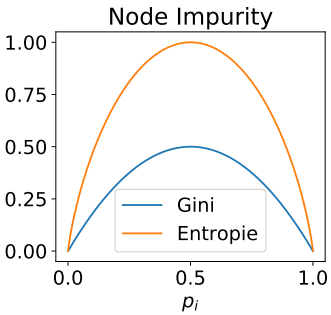


Abbildung 2.13: Gini- und Entropie-Werte

Bei eindeutigen Zuordnungswahrscheinlichkeiten p_i nahe 0 oder 100 % sind die Werte niedrig. Daher definiert immer das Kriterium den nächsten Entscheidungsknoten, welches den geringsten Node Impurity-Wert aufweist. Motivation dabei ist, Bäume mit möglichst

geringer Tiefe zu erzielen. Um die Eignung der Kriterien für ‚den nächsten' Knoten zu bewerten und miteinander zu vergleichen, wird für jedes Kriterium der Mean Decrease in Impurity (MDI) berechnet. Er ist als ‚mittlerer Beitrag zur Senkung der Node Impurity' definiert. Bäume können dennoch schnell groß werden. Eine Möglichkeit, das zu verhindern ist das sogenannte ‚Pruning' (Zuschneiden) der Bäume. Dafür werden Sequenzen von Knoten im Baum gesucht, die keine signifikant niedrigen Node-Impurity-Werte aufweisen. Diese kann man bei relativ geringem Verlust der Vorhersagequalität und entsprechend großem Gewinn an Vorhersagegeschwindigkeit zusammenfassen (Bishop 2009, S. 665).

Die Konstruktion von Entscheidungsbäumen wird als ‚Lernen' bezeichnet. Die bekanntesten Algorithmen dafür sind ‚ID3' und‚C4.5' (Quinlan 1993). Über die Node Impurity werden nacheinander die Kriterien gesucht, durch welche die Datenpunkte des Trainingsdatensatzes am eindeutigsten klassiert werden können. Singuläre, vollständig ausgeprägte Entscheidungsbäume neigen zum ‚Over-Fitting'. Dabei lernt das Modell ungewollt auch die Einflüsse des Rauschens mit zu berücksichtigen und spezialisiert sich damit zu sehr auf den genutzten Trainingsdatensatz (Kelleher et al. 2015, S. 52). Gegenmaßnahmen sind das oben bereits beschriebene Pruning, Festlegung einer Mindestanzahl an Datenpunkten pro Knoten oder einer maximalen Baumtiefe. Over-Fitting kann auch mit Ensemble-Methoden begegnet werden. Die Grundidee ist, mehrere unterschiedliche Bäume zu generieren und diese unabhängig voneinander Vorhersagen treffen zu lassen. Die Gesamtprognose ist dann eine Mehrheitsentscheidung.

Ensemble-Methoden lassen sich in zwei Gruppen einteilen: Averaging- und Boosting-Methoden. Erstere generieren randomisiert mehrere Entscheidungsbäume, um deren Einzelprognosen zu kombinieren, die zweiten erstellen die Menge an Bäumen sequenziell, um so gezielt Over-Fitting-Effekte der vorangegangenen Bäume zu minimieren (Pedregosa et al. 2011)[3]. Die klassische Ausprägung der Averaging-Methoden ist das ‚bagging', (kurz für: Bootstrap Aggregation). Dafür werden für die Konstruktion jedes einzelnen Baumes Auszüge aus dem Trainingsdatensatz genutzt. So entsteht eine Sammlung voneinander unabhängiger Bäume mit kleiner Tiefe, sogenannte ‚weak learner' (Kuhn & Johnson 2016, S. 279). Der verbreitetste Boosting-Algorithmus ist AdaBoost (kurz für: Adaptive Boosting) und wurde von Freund und Schapire entwickelt (Schapire 1990). Dabei werden falsch

[3]https://scikit-learn.org/stable/modules/ensemble.html

klassierte Datenpunkte der vorangegangenen Bäume stärker gewichtet, um den Fehler bei den neuen Bäumen zu vermeiden (Bishop 2009, S. 657).

Seit den 2000ern sind die Random-Forest-Algorithmen als robust und praxistauglich anerkannt. Boosting-Algorithmen sind dabei die leistungsstärksten (Chollet 2018, S. 39). Grund dafür sind die zahlreichen Vorteile von Entscheidungsbäumen und Random Forests. Da sie unabhängig von der Anzahl an Dimensionen visualisiert werden können, sind sie im Gegensatz zu kNN nachvollziehbar und interpretierbar („white box model"). Gleichzeitig ist die Form der Inputdaten eindeutig, was den Aufwand der Datenvorbereitung verringert. Es können nicht-normalisierte Daten verwendet werden und der Algorithmus ist robust gegenüber leeren Datenpunkten. Der Aufwand des Trainingsvorgangs verhält sich logarithmisch mit der Anzahl an Datenpunkten. Weiter können die Datenpunkte sowohl numerisch als auch nominal sein. Sowohl einzelne Bäume als auch die Ensemble-Methoden können mit statistischen Werkzeugen untersucht werden, um damit Aussagen über die Zuverlässigkeit zu treffen.

Ein Problem einzelner Bäume ist die mangelnde Generalisierung durch ‚überkomplizierte' Baumstrukturen. Die oben beschriebenen Maßnahmen, um das Over-Fitting zu vermeiden sind hier absolut notwendig. Einzelne Bäume können ‚instabil' sein. Das bedeutet, dass kleine Änderungen am Trainingsdatensatz zu völlig anders gestalteten Bäumen führen und sich daher nur wenig für die Extrapolation eignen. Das Training eines optimalen Entscheidungsbaumes ist NP-vollständig. Das heißt, dass sich mit heutiger Rechenleistung *wahrscheinlich kein optimaler* Entscheidungsbaum effizient finden lässt. Darum werden Entscheidungsbäume zufallsbasiert erstellt, die optimale Lösung kann nicht garantiert werden. Auch das erfordert Ensemble-Methoden, bei denen die Kriterien und Daten randomisiert verwendet und ersetzt werden. Sowohl einzelne Bäume als auch Wälder enthalten im Modell einen ‚Bias', wenn im Datensatz einzelne Klassen in der Anzahl dominieren. Der Datensatz sollte daher in der Anzahl der Klassenausprägungen austariert sein (Pedregosa et al. 2011)[4]. Abbildung 2.14 zeigt in jeder Zeile denselben Datensatz, mit dem unterschiedliche Prognosemodelle (Spalten) trainiert wurden. Die in den Farben rot, gelb und grün eingefärbten Regionen stellen die Klassen dar. Die Farbsättigung repräsentiert den Anteil korrekt zugeordneter Trainings-Samples. Dabei sind beim Entscheidungsbaum in

[4] https://scikit-learn.org/stable/modules/tree.html

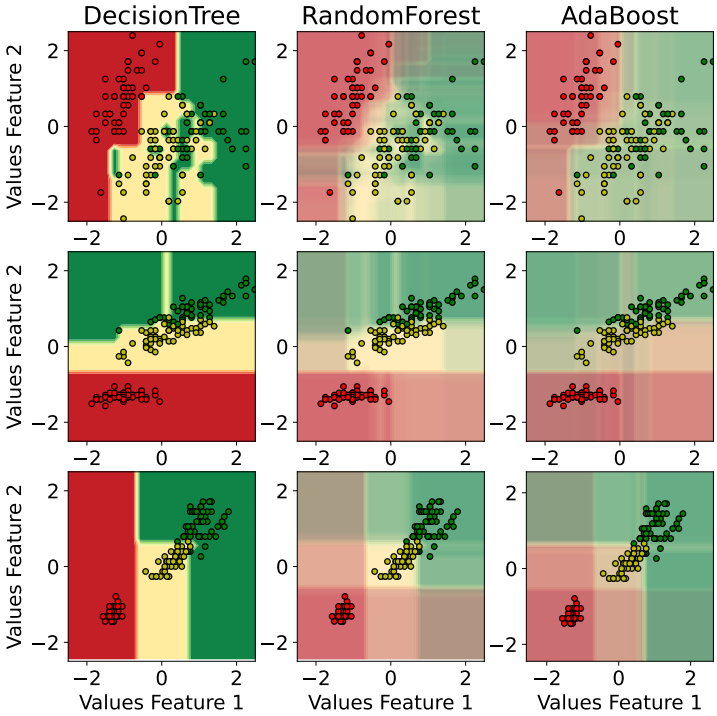


Abbildung 2.14: Unterschiedliche baumbasierte Prognosemodelle

der ersten Spalte gut die scharfen Grenzen zwischen den Regionen zu erkennen. Jeder einzelne Datenpunkt hat großen Einfluss auf den Bereich im aufgespannten Raum der beiden Kriterien. Bei den beiden Ensemble-Methoden ‚Random Forest' (= Bagging) und ‚AdaBoost' sind die Prognosen unscharf durch den Mittelwerteffekt der Prognosen mehrerer Bäume.

Der Abschnitt 2.3 gibt zunächst eine kurze Rückschau zur Geschichte der Forschung im Bereich der KI wieder und leitet davon ausgehend zum Teilbereich des ML über. Dabei wird im Hinblick auf das in dieser Arbeit zu lösende Problem besonders auf die Klassierungsmethoden des überwachten Lernens eingegangen. Genauer beleuchtet werden die gängigen Ansätze der LR, SVM, Naive Bayes, kNN und der Entscheidungsbäume.

Fazit:

Aufgrund der Problemstellung dieser Arbeit bieten sich Entscheidungsbäume und die daraus generierten Ensemble-Methoden an. Daher werden deren Ausprägungen umfänglich beschrieben, die dahinterliegende Mathematik komprimiert dargelegt und die Vor- und Nachteile dieser Klassierungsmethode aufgezählt.

Mit den Ausführungen zum maschinellen Lernen ist der dritte Block neben turbulenten, komplexen Systemen und der Mathematik zu funktionalen Abhängigkeiten abgeschlossen. Das Folgekapitel geht auf den Stand der Wissenschaft zur Turbulenzbewertung ein. Das beinhaltet den pragmatischen Übertrag der Systemtheorie mit Ein- und Ausgangssignalen auf Produktionsdaten, was Data Mining in der Produktion aktuell leisten kann, wie Turbulenz heute gemessen wird und mit welchen IT-Werkzeugen die Prognosemodelle für diese Arbeit umgesetzt werden.

3 Stand der Technik

Nach den Ausführungen zu den von Philosophie und Mathematik geprägten formalwissenschaftlichen Grundlagen folgt nun der gegenwärtige Stand der Technik. Dieser stellt die Forschungsregion dar, in der die größte Dynamik in der Anwendung von gesichertem Wissen vorherrscht. Daraus folgt direkt eine im Vergleich zur Formalwissenschaft wesentlich kürzere Gültigkeitsdauer. Die Dynamik im Stand der Technik ist geprägt vom technologischen Fortschritt der befähigenden Technologien und den gegenwärtig bestimmenden Forschungstrends. Im Fall dieser Arbeit ist das zunächst die Möglichkeit, auf Standard-Rechnern Rechenleistungen abrufen zu können, welche ML-Anwendungen zulassen. Großrechner sind für eine Vielzahl an Problemen nicht mehr notwendig. Forschungsseitig werden Daten als der Rohstoff der Zukunft nach wie vor gerade für den Industriebereich gefördert. Die Hightech-Strategie 2025[1] nennt ‚Wirtschaft und Arbeit 4.0' als ein Zukunftsthemen der Forschungsstrategie des Bundes. Demnach ist der Einsatz digitaler Technik an nahezu jedem Arbeitsplatz ein zentraler Baustein dieser Entwicklung.

Um den Stand der Technik im Bereich der Turbulenzbewertung zu umfassen, werden drei Perspektiven eingenommen. Zunächst wird das im Überbegriff **Data Mining** zusammengefasste Feld der datenbasierten Wissensgewinnung in der Produktion beleuchtet. Dieser Abschnitt beinhaltet den Stand der bereits in der Anwendung befindlichen Technik. Es folgt der Betrachtungsgegenstand der **Turbulenzbewertung** an sich, um den Stand der Technikwissenschaft abzudecken. Am Ende fokussieren die Betrachtungen auf die **Prognose von Produktionskennzahlen**, wobei die zentrale Forschungslücke und das Kernproblem bei der Erreichung des Ziels dieser Arbeit aufgezeigt werden.

Das ermöglicht im Fazit einen zweistufigen Überblick über die Literatur. Zum einen werden die theoretischen Vorarbeiten zur Turbulenzbewertung vergleichend dargestellt und zum andern die technischen Lösungsansätze dazu bewertet. Die Kriterien dazu ergeben sich aus dem Stand der Formalwissenschaft, der Forschungslücke und der zentralen Forschungsfrage.

[1] https://www.hightech-strategie.de/hightech/de/handlungsfelder/zukunftsthemen/zukunftsthemen_node.html

3.1 Data Science und Data Mining in der Produktion

Die von Papp et al. beschriebenen positiven Visionen zum Nutzen der KI in verschiedenen Branchen sind in Teilen bereits Realität. So ist allgemein davon die Rede, mit Daten „Produktionsprozesse zu verbessern". Dazu gehören die beiden Anwendungsfälle Optimierung und Vorhersage. In beiden Fällen lautet die Aufgabe Unterstützung bei Entscheidungen (Papp et al. 2019, S. 19; Flack et al. 2020, 829 f.). Konkrete Schwierigkeit im Branchenkontext der Produktion ist das erhöhte Datenvolumen und die spezielle Codierung, in der sich Produktionsdaten oftmals stark unterscheiden (Papp et al. 2019, S. 259; Dehghantanha 2019, S. 54). Besonders bei der Betrachtung historischer Daten tritt dies auf. Dies ist allerdings für den Vergleich mit Zeitreihen, Faktoren und allgemein Daten der Gegenwart notwendig, um so über Regression Aussagen treffen zu können (Papp et al. 2019, S. 260).

Die Verarbeitung von Daten mit dem Ziel, Wissen zu generieren wird allgemein als „data and knowledge mining" bezeichnet und beherbergt operativ unterschiedlichste Methoden, die den Data Mining Techniques (DMT) zugerechnet werden. Mit den ersten Datenbanken in den 1960er Jahren entstanden zunächst primitive Formen der Datenverarbeitung – die Geburtsstunde des Data Mining (Han & Kamber 2006, 2 ff.). Damit konnte die Pionierarbeit im Umgang mit größeren Datenmengen geleistet werden. Im Produktionskontext adressieren DMT vorwiegend die echtzeitnahe, dynamische und selbstanpassende Steuerung. Die Wissensgenerierung aus großen Datenmengen (Knowledge Discovery in Database (KDD)) etablierte sich in den 1990ern als ein eigenständiges Arbeitsfeld (Fayyad et al. 1996, 82 ff. Ester & Sander 2000, 1 ff.) und wurde von textscCheng et al. als ein „nicht trivialer Prozess der Identifizierung gültiger, neuartiger, potenziell nützlicher und letztlich verständlicher Muster aus einem Datensatz" definiert (Cheng et al. 2018, S. 1).

Cheng et al. fassen zusammen, dass die Hauptfunktionen der DMT in *beschreibend* und *prädiktiv* eingeteilt werden können (siehe Abbildung 3.1). Mit den in den Abschnitten 2.2 und 2.3 beschriebenen Werkzeugen werden solche Hauptfunktionen realisiert. Beschreibende Werkzeuge befassen sich mit der Suche nach verborgenen Zusammenhängen im Datensatz. Werden diese erkannt, können Phänomene des durch die Daten beschriebenen Systems verstanden, Erkenntnisse abgeleitet und somit das System besser gehandhabt werden. Bei der Generalisierung wird der Datensatz in seinen Eigenschaften untersucht. Sie

beinhaltet eine Bewertung der Datenqualität, beschreibt den Datensatz stochastisch und ermöglicht erste Grundaussagen zum beschriebenen System. Die Assoziierung zielt auf die Entdeckung von Zusammenhängen zwischen im Datensatz beschriebenen Ereignissen ab. Dies erfolgt oft mit Algorithmen (z.B. der Apriori-Algorithmus (Agrawal et al. 1993, 914 ff.)) oder entsprechenden oben bereits genannten mathematischen Methoden wie der Korrelationsanalyse. Mit Mustererkennung sollen zeitabhängige und wiederkehrende Merkmalssequenzen im Datensatz gefunden werden. Sie ist der Assoziierung ähnlich und bildet oft die Grundlage für Voraussagen. Klassifizierung ist der Vorgang der Identifikation und Definition von Klassen innerhalb einer Gruppe von Individuen basierend auf Ähnlichkeiten. So sollen die Unterschiede der Featureausprägungen von Individuen innerhalb einer Klasse minimal und die verschiedener Klassen maximal sein.

Die Klassifizierung bildet die Voraussetzung zur Klassierung, eine der prädiktiven Hauptfunktionen von DMT. Als Klassierer werden Funktionen oder Modelle bezeichnet, die Objekte einer Klasse zuordnen. Die Funktions- oder Modellerstellung wird als Trainingsphase bezeichnet. In der darauf folgenden Testphase wird der Klassierer auf seine Leistungsfähigkeit (Genauigkeit und Sicherheit) bewertet. Die Voraussage ist mit der Klassierung verwandt, hat dabei jedoch nicht die Zuordnung zu einer Klasse, sondern die Aussage über einen festen Wert zum Ziel und entspricht mathematisch der Regression. Die Zeitreihenanalyse nutzt Zeitreihen (eine spezielle Art strukturierter Daten entsprechend Unterabschnitt 2.2.1, um Aussagen zu zukünftigen Trends über kurz-, mittel- und langfristige Zeiträume zu treffen. Abschließend ist das Exception Knowledge Mining von Relevanz, das Wissen aus den Daten zu Objekten generiert, deren Verhalten nicht zu den identifizierten Regeln oder Klassen passt. Die Zuordnung der Anwendungsmöglichkeiten zur beschreibenden oder prädiktiven Funktion sind teilweise unscharf. So können beispielsweise Zeitreihen hinsichtlich ihres Verhaltens lediglich analysiert werden (beschreibend) oder eben Voraussagen getroffen werden (prädiktiv).

Nach dem Überblick über Datenverarbeitung im Allgemeinen werden die für diese Arbeit relevanten Kernaspekte für die praktische Umsetzung beleuchtet. Zunächst wird das Vorgehen beschrieben, mit dem Zustände und Ereignisse eines Produktionssystems digital repräsentiert werden können. Diese Arbeit bedient sich der Strukturen des Complex Event Processings (CEP). Weiter wird der allgemeine Ansatz des Case Based Reasoning

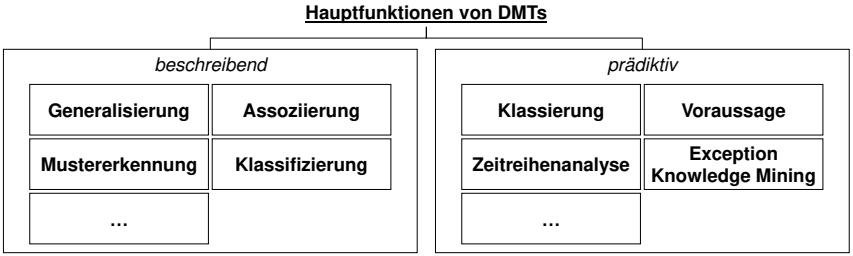


Abbildung 3.1: Hauptfunktionen der Data-Mining-Techniken in Anlehnung an (Cheng et al. 2018; Han & Kamber 2006)

(CBR) beschrieben, mit dem solche Vorgehensweisen entwickelt werden können, welche in der Lage sind, mit noch unbekannten Problemen und Situationen umzugehen. Schließlich wird auf Handlungssequenzen (Data Pipelines) eingegangen, welche sich im Rahmen von Data-Science-Projekte bewährt haben. Sie bilden mit die Grundlage für den späteren Lösungsansatz.

Complex Event Processing (CEP)

Als CEP bezeichnet man sowohl Softwaretechnologien, welche Muster in großen Mengen von Ereignissen detektieren, als auch „[...] eine Technik zur Identifikation mehrerer Ereignisse, die nicht notwendigerweise im [sic!] gleichen Zeitpunkt stattfinden müssen, die aber in einer Beziehung zueinander stehen und für das Auslösen einer Reaktion entscheidend sind“ (Hedtstück 2017, S. 3). Es stellt eine auf das Internet der Dinge ausgerichtete Möglichkeit des Data Analytics dar, mit der große Datenmengen echtzeitnah ohne Verwendung einer Datenbank prozessiert und ausgewertet werden können. (Etzion 2011, 15 f.). Ein Ereignis (Event) ist ein zeitlich punktueller, zustandsverändernder Vorgang innerhalb eines Systems, welches die Ursache einer diskreten Zustandsänderung darstellt. CEP-Software ist darauf ausgerichtet, einen kontinuierlichen Datenstrom auszuwerten. Zustandsänderungen können *diskret* oder *kontinuierlich* sein (siehe Unterabschnitt 2.2.1). Der dem Event zugeordnete Zeitpunkt heißt Zeitstempel (engl. timestamp) (Hedtstück 2017, 2 f.). Ein Event nimmt dabei keine Realzeit in Anspruch, sondern tritt immer zu einem bestimmten Zeitpunkt ein. Eine Aktivität (engl. activity) hingegen läuft während einer Zeitdauer ab, nimmt damit Realzeit in Anspruch und ändert den Zustand eines Systems nicht. Sie startet

und endet mit Anfangs- und Endereignis (Beispiel: Transport einer Palette mit Abfahrt an der Maschine und Ankunft im Lager) (Hedtstück 2017, S. 12).

Neben diesen ‚elementaren', ‚atomaren' oder auch ‚primitiven' Ereignissen werden noch ‚komplexe' Ereignisse definiert. Solche – auch als Ereignisinstanzen bezeichneten – Ereignisse sind eine „[...] endliche Menge von atomaren Ereignissen, die passend zu einem vorgegebenen Ereignismuster zueinander in Beziehung stehen" (Hedtstück 2017, S. 14). Ein Beispiel dafür ist die Sequenz der elementaren Events *Bearbeitungsende – Anlagenaustritt – Transportbeginn – Transportende*. Komplexe Ereignisse werden genutzt, um Ereignisströme zu strukturieren und damit handhabbarer zu machen. Ereignisströme werden beim CEP auf der Grundlage von gleitenden Längen- oder Zeitfenstern ausgewertet. Längenfenster werden durch die Anzahl von Ereignissen definiert, die Zeitfenster durch ein Zeitintervall. Der Zeitstempel zum tatsächlichen Ereignis wird expliziter Zeitstempel genannt, der Zeitstempel zur Ankunft des digitalen Repräsentanten des Ereignisses impliziter Zeitstempel. Somit ist der explizite Zeitstempel immer kleiner oder gleich dem impliziten Zeitstempel, die Differenz wird als Zeitversatz (engl. event time skew) bezeichnet. Es sind deshalb Realzeit (engl. event time) und Systemzeit (engl. processing time) in ihrer Bedeutung und Verwendung voneinander zu trennen (Hedtstück 2017, S. 23).

Weiter sind zwei Arten von CEP zu unterscheiden. Während beim detection-oriented CEP die zu findenden Ereignismuster bereits bekannt sind, müssen beim computation-oriented CEP neue, unbekannte Muster gefunden werden. Hier werden Techniken aus dem maschinellen Lernen und Data Mining eingesetzt und hier liegt auch der Berührpunkt zu dieser Arbeit. Wichtig ist dabei immer, das Produktionssystem für CEP kompatibel mit einem Modell zu beschreiben, welches durch einen Strom von Ereignissen repräsentiert werden kann. CEP bildet damit den festgelegten Rahmen für alle verwendeten Daten in dieser Arbeit.

Case-based Reasoning (CBR)

CBR ist ein methodisches Vorgehen, um Erfahrungswissen – unabhängig von der Art der Repräsentation des Wissens – auf neue Probleme anzuwenden (Watson 1999; Kolodner 1992; Schank & Leake 1989). Kolodner nennt fünf unterschiedliche Anwendungsfälle, welche in die Kategorien *interpretierend* und *problemlösend* eingeteilt werden können (siehe Abbildung 3.2). Problemlösendes CBR wird für Design- und Planungsaufgaben, Neudiagnosen

und Ursachenklärung verwendet. Dabei ist die bestehende Situation der *Prozessinput*, eine (neue) Lösung der *Output*. Das interpretative CBR beschäftigt sich mit der Deutung und Bewertung von vorliegenden Situationen und Lösungen. Dazu gehören rechtfertigendes Begründen, Klassieren, Effektvorhersage und interpretierendes Problemlösen.

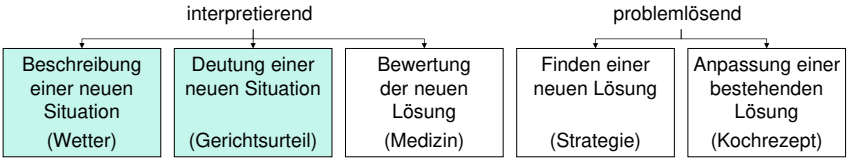


Abbildung 3.2: Fünf Anwendungsfälle des CBR (Kolodner 1992)

Unabhängig davon, welchen der *Stile* konkret angewendet werden, beschreibt Kolodner eine vorherrschende Sequenz an Prozessen des CBR. Diese in Abbildung 3.3 gezeigte Sequenz kann intuitiv und werkzeugunabhängig genutzt werden, um Lösungen für Probleme zu finden und wird daher auch auf rechnergestützte Abläufe übertragen. CBR wird als „simpel und einfach zugänglich" beschrieben. Für das initiale, falls möglich selektive, Aufrufen des Wissens wird schnell eine Groblösung als Ausgangspunkt generiert. Die wird auf die neue Situation angepasst, um – auch unter Nutzung des bestehenden Wissens – bestätigt oder kritisiert zu werden. Kritikpunkte können vor der Anwendung der Lösung weitere Anpassungen zur Folge haben. Nach der Anwendung wird die Leistungsfähigkeit der Lösung bewertet, um das neue Wissen dem bestehenden persistent hinzuzufügen und als erweitertes Gesamtwissen zur Verfügung zu stehen. CBR ist wegen seinen zahlreichen Vorteile als Framework verbreitet. Es bietet rein strukturell schnell erste Lösungen an und ist gleichzeitig unabhängig von der Komplexität der Aufgabe und dem Umfang des zur Verfügung stehenden Wissens. Weiter ist es in der Lage, vor der eigentlichen Anwendung der Lösung zu optimieren und als Gesamtsystem zu lernen. Kritikpunkte sind gleichzeitig die praktische Unmöglichkeit, Lösungen ‚Out of the Box' zu finden und die Gefahr von Bias und Over-Fitting (siehe Unterabschnitt 2.3.3 Entscheidungsbäume).

Der situationsbeschreibende und situationsdeutende Fall in Abbildung 3.2 sind für diese Arbeit relevant (in grün). Situationsbeschreibend deshalb, weil durch Kombination von Modell und Daten die Produktionssituation des Wertschöpfungssystems möglichst genau repräsentiert werden soll. Situationsdeutend, da aus der Situationsbeschreibung Schlüsse –

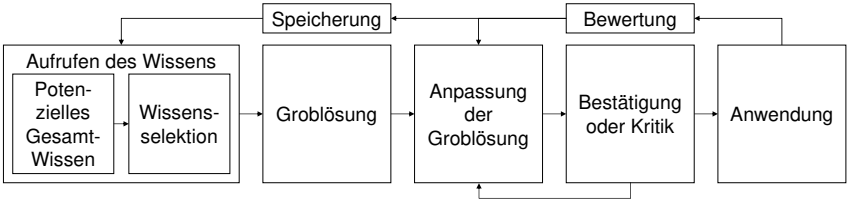


Abbildung 3.3: Vorgehenssequenz des CBR in Anlehnung an (Kolodner 1992, S. 22)

hier Prognosen einzelner Zielgrößen – gezogen werden. Die Erfolgsaussichten des oben beschriebenen allgemeinen CBR hängen für sowohl den situationsbeschreibenden als auch -deutenden Fall naturgemäß von mehreren Faktoren ab: der Größe des Erfahrungsschatzes, die Fähigkeit, neue Situationen auf der Grundlage des bestehenden Wissens zu verstehen, die Fähigkeit zur Anpassung und die Bewertungskompetenz (Kolodner 1992, S. 5). Das Wissen über diese Faktoren und deren Ausprägungen können verwendet werden, um die angestrebte Lösung zu generieren und zu verbessern.

Standardprozessmodelle

Ausgehend von den eingangs genannten KDD-Prozessen in den 1990er Jahren entwickelten sich die unterschiedlichsten Data-Processing-Methologien (Mariscal et al. 2010, S. 142). Mit Human-centered, SEMMA, Cabena et al., Two Crows und Anand & Buchner sind einige Evolutionsschritte genannt, die mit dem Cross Industry Standard Process for Data Mining (CRISP-DM) in einem allgemein anerkannten Standard mündeten (Kelleher et al. 2015, S. 52). Daraus entwickelte sich im Hause IBM später Analytics Solutions Unified Method for Data Mining/Predictive Analytics (ASUM-DM). Diese aktuell verwendeten und für diese Arbeit sinnigen Prozessmodelle werden hier kurz vorgestellt. Die Datenpipeline von Papp, welche die vorangegangenen Modelle ergänzt, vervollständigt das Repertoire an Frameworks zu Data Mining in der Produktion im Bereich Architektur (Papp et al. 2019, S. 130).

Das aktuell verbreitetste[2] branchenübergreifende Standardmodell für Data Mining ist **CRISP-DM**. Mit dem Ziel, ein anwendungs- und herstellerneutrales Prozessmodell bereitzustellen, wurde ab 1996 CRISP-DM unter Beteiligung einiger Unternehmen (Integral Solutions Ltd (ISL), Teradata, Daimler AG, NCR Corporation und OHRA) und EU-gefördert entwickelt

[2] https://www.kdnuggets.com/polls/2014/analytics-data-mining-data-science-methodology.html, zuletzt aufgerufen am 19.12.2021.

und um die Jahrtausendwende veröffentlicht (Wirth & Hipp 2000; Mariscal et al. 2010, S. 142). Im CRISP-DM umfasst ein Prozesszyklus von Data-Mining-Projekten folgende sechs Phasen, deren Reihenfolge nicht streng festgelegt ist (siehe auch Abbildung 3.4):

- **Geschäftsverständnis** (Business Understanding, BU):
  Konkrete Zielfestlegung und Anforderungen fürs Data Mining; Ergebnis ist eine formulierte Aufgabenstellung und Vorgehensbeschreibung
- **Datenverständnis** (Data Understanding, DU):
  Schaffung eines Überblicks über die verfügbaren Daten und deren Qualität (Analyse und Bewertung); Problemdokumentation mit Abschätzung des Risikos auf die Aufgabenstellung
- **Datenvorbereitung** (Data Preparation, DP):
  Erstellung eines finalen Datensatzes als Grundlage für die folgende Modellbildung
- **Modellierung** (Modeling, M):
  Anwendung der für die Aufgabenstellung geeigneten Methoden des Data Mining auf den Datensatz; Optimierung der Parameter und Erstellung von Alternativmodellen
- **Evaluierung** (Evaluation, E):
  Exakter Abgleich der erstellten Datenmodelle mit der Aufgabenstellung und Auswahl des passendsten Modells
- **Bereitstellung** (Deployment, D):
  Inhaltliche Aufbereitung der gewonnenen Ergebnisse mit Visualisierung

Mit dem Vorgehen zielt CRISP-DM darauf ab, qualitative Ergebnisse trotz begrenzter IT-Ressourcen und -Fähigkeiten sicherzustellen. Die Ergebnisse sollen auf die Aufgabe ausgerichtet und durch die Dokumentation nachhaltig nutzbar sein. Robustheit und Generalisierbarkeit sind dabei von den Entwicklern beabsichtigte Kerneigenschaften. Wegen der großen Beachtung des Modells („most widely used methodology for developing data mining projects“ (Mariscal et al. 2010, S. 138)), entwickelte sich der Standardprozess weiter.

ASUM-DM ist eine von IBM im Jahr 2015 veröffentlichte Weiterentwicklung von CRISP-DM für Data Mining und Predictive Analytics. So soll CRISP-DM überarbeitet und bestehende

Schwachstellen eliminiert werden. ASUM-DM umfasst fünf Phasen und zusätzlich einen Projektmanagementablauf, welcher Konsistenz, Zusammenarbeit und Kommunikation sicherstellt (Angée et al. 2018; Mockenhaupt 2021):

- **Analyse** (Analyze):
  Definition der Abnahmekriterien von funktionalen und nicht-funktionalen Eigenschaften durch alle Beteiligten
- **Design** (Design):
  Definition der Lösungskomponenten und deren Abhängigkeiten; Erstellung der Entwicklungsumgebung mit agiler, kurzzyklischer Überprüfung der Realisierbarkeit der Anforderung
- **Konfiguration** und Herstellung (Configure and Build):
  Test und Validierung der in der Design Phase erstellten Komponenten
- **Inbetriebnahme** (Deploy):
  Entwicklung von Wartungsplan, Anwender-Support und Service, Installation der Lösung
- **Betrieb** und Optimierung (Operate and Optimize):
  Planung der Wartungsarbeiten und durchgängige Überwachung der Qualität der Applikation; Parallel Projektmanagement mit koordinierter Kommunikation für optimale Zusammenarbeit

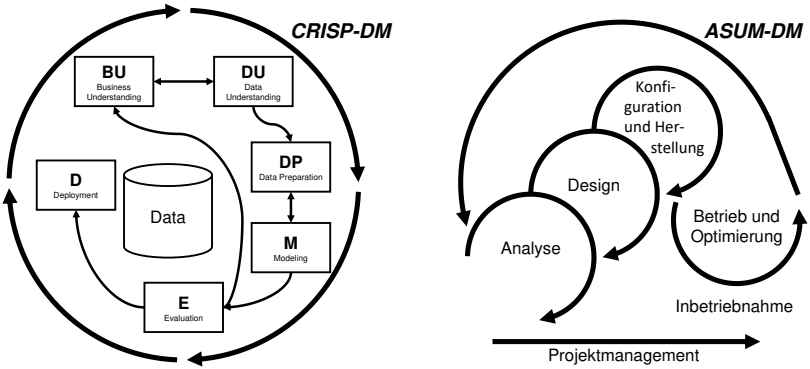


Abbildung 3.4: CRISP-DM und ASUM-DM in Anlehnung an (Ponsard et al. 2017)

Sowohl CRISP-DM als auch ASUM-DM betrachten isolierte Data-Mining-Projekte. Um kontinuierlich mit der Umwelt interagierende Data-Mining-Systeme zu etablieren, schlägt Papp eine Architektur der **Datenpipeline** mit sechs Stufen vor (Abbildung 3.5). Diese ist in der Lage, sich dem Zweck des ‚Datenprodukts' mit der Zeit anzupassen und in Interaktion mit der Realität seine Leistungsfähigkeit zu evaluieren. Dabei bezieht sich Papp speziell auf Klassierungsprobleme (Papp et al. 2019, S. 130).

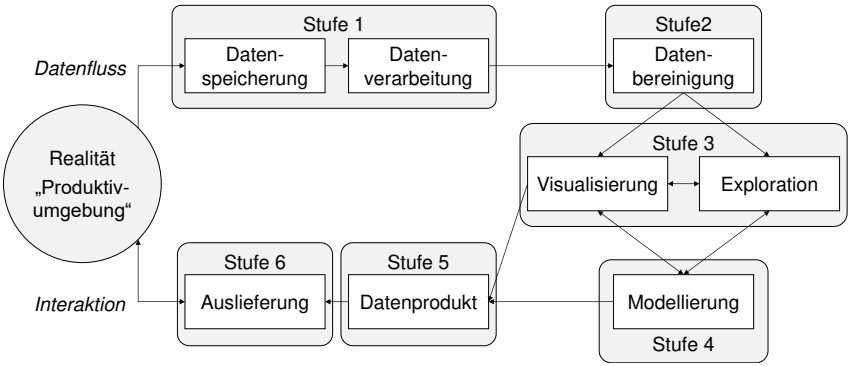


Abbildung 3.5: Architektur einer Datenpipeline nach Papp

Papp betrachtet die verwendeten Daten nicht als bereits vorliegend, sondern berücksichtigt die Notwendigkeit, Daten ausführungsparallel weiter zu sammeln (aus Quellen wie ERP-, CRM-Systemen, Sensoren oder dem Internet), um damit den Datensatz kontinuierlich zu aktualisieren. Zu dieser ersten Stufe der *Datenerhebung* werden sowohl Datenspeicherung als auch Datenverarbeitung gezählt. Bei der *Datenbereinigung* (Stufe 2) gilt es, die Daten einheitlich in die Form zu bringen, damit auf sie die jeweiligen Machine-Learning-Algorithmen angewandt werden können. Papp nennt für Datensätze folgende Eigenschaften, die erfüllt sein müssen, um als ‚analytisches Datenset' zu gelten:

1. Es gibt einen identifizierenden Schlüssel.
2. Sämtliche Attribute sind gleich enkodiert.
3. Die Abfrage und Exploration werden mittels einer Abfragesprache unterstützt.
4. Fehlende oder Standardwerte folgen einem festgelegten Schema.

Die *Exploration & Visualisierung* in Stufe 3 erfordert gewisses Erfahrungswissen, um den Datensatz zielgerichtet zu erforschen. Darstellungsarten wie Box-Plots, Histogramme, Streudiagramme sind dabei hilfreich (Mazarov et al. 2020). Diese Stufe kann iterativ mehrmals durchlaufen werden und wird auch zur Generierung des finalen Datenprodukts genutzt. Im Anschluss an die Exploration ist die Modellbildung in Stufe 4 notwendig, um entsprechende Ergebnisse automatisiert zu erzeugen. Die Modellbildung enthält deren Bau, Training, Bewertung und, bedarfsabhängig, Optimierung (siehe Unterabschnitt 2.3.2 Maschinelles Lernen und Klassierung). Stufe 5 *Interpretation* entspricht dem finalen Visualisierungsschritt und der zweckmäßigen Aufbereitung und Deutung des Datenproduktes. In Stufe 6 (*Auslieferung*) wird das Datenprodukt mit seiner Deutung in Maßnahmen übersetzt und in Interaktion mit der Realität in der Produktivumgebung umgesetzt.

Ausgehend vom Stand der Technik zum generellen Umgang mit Produktionsdaten erfolgen nun die Ausführungen zur Turbulenzbewertung auf der Anwendungsebene.

3.2 Turbulenzbewertung

Im Unterabschnitt 2.1.1 wurde bereits die messbare Turbulenz in Anlehnung an Wiendahl als die „Abweichung signifikanter Kennwerte von ihrem Mittelwert" definiert (Wiendahl 2011, S. 208). Damit ermöglicht Wiendahl den Übertrag der Definition vom Auftragsmanagement auf das gesamte Produktionssystem. Es sei betont, dass dabei Turbulenz die messbaren Auswirkungen auf das Produktionssystem und nicht die „objektiv messbaren Aspekte" im Sinne von Mintzberg darstellen, welche die Turbulenz-*Ursachen* beschreiben (Mintzberg 1994). Auch bei Wiendahl liegt der Fokus der Bewertung nicht auf den Turbulenzen selbst, sondern auf den turbulenzerzeugenden Faktoren – den Turbulenzkeimen. Um bewertet zu werden, müssen diese hier eine unternehmensindividuelle Relevanzschwelle überschreiten. Ist dies der Fall, werden die Turbulenzkeime anhand einer explizit subjektiven Expertenbewertung auf einer Skala zwischen 0 (geringer Einfluss) und 10 (hoher Einfluss) eingeordnet und mittelwertbasiert bewertet.

Trotz des Fokus auf die Ursachen trifft Wiendahl einzelne Aussagen zur objektiven Turbulenzbewertung: Er betont die hohe Bedeutung der in der Definition enthaltenen *Signifikanz* der Abweichung vom Mittelwert. So ist eine Abweichung dann signifikant, wenn

eine vereinbarte zulässige Toleranz überschritten wird. Damit ist ein Zielkorridor festgelegt, innerhalb dem der Soll-Wert liegt. Je nach Lage des Ist-Werts ergibt sich daraus eine signifikante Abweichung – oder eben nicht. Sowohl für Ursache als auch für Auswirkung wird ein Beispiel angegeben: Es werden der Turbulenzkeim „Kapazitätsschwankung" und seine Auswirkung auf das Erreichen der „Vorgabezeiten" dargestellt. Hier wird ein funktionaler Zusammenhang zwischen Ursache und Wirkung impliziert, eine Bewertung der Turbulenz, Regelableitung oder Maßnahmen finden jedoch nicht statt. Dennoch wird ersichtlich, dass für die Zuordnung, ob Turbulenz vorliegt, eine Zuteilung zu Klassen notwendig ist (in diesem Fall, ob eine Vorgabezeit innerhalb des Toleranzkorridors liegt).

In dem genannten Beispiel werden zwei Klassen (innerhalb und außerhalb der Toleranz) genutzt, jedoch sind prinzipiell beliebig viele Klassen vorstellbar (Cramer & Kamps 2017, S. 42; Eckstein 2014, S. 29). Zur Festlegung der Klassen werden deren Toleranzkorridore und zentrale Soll-Werte benötigt. Ein Zugang sind die von Nyhuis et al. beschriebenen logistischen Kennlinien (Nyhuis & Wiendahl 2012). Sie bringen verschiedene Zielgrößen wie Termineinhaltung, Durchlaufzeit, Leistung oder Kosten in funktionalen Zusammenhang zum WIP und finden Anwendung im Bereich der Intralogistik. Um solche Kennlinien zu erstellen, schlagen Nyhuis et al. wörtlich „Versuche" der Warteschlangentheorie und simulationsbasierte Ansätze vor. Aus Gründen der weiter oben beschriebenen Systemkomplexität scheitern diese Ansätze meist. Die Autoren schlagen weiter Näherungsgleichung auf Grundlage weniger Daten vor und beschreiben das Potenzial, das anhand von „Zielgrößen gemessene Verhalten von Produktionssystemen rechnerisch vorauszubestimmen" (Nyhuis & Wiendahl 2012, S. 14). Allerdings beziehen sich diese Ausführungen immer auf einen stationären Betriebszustand (Nyhuis & Wiendahl 2012, S. 37). Die Autoren beschreiben mehrere potenzielle rechnerische Modelle und betonen dabei ausdrücklich, dass die Schwierigkeit nicht die mathematische Formulierung ist, sondern das „mehr oder weniger intuitive Erkennen der relevanten elementaren Zusammenhänge" (Nyhuis & Wiendahl 2012, S. 39). Für jede Zielgröße werden Näherungsgleichungen abgeleitet, die dann Voraussagen zu den jeweiligen Zielgrößen zulassen. Damit existiert für jede Kennlinie ein optimaler Betriebspunkt, an dem bei isolierter Betrachtung der einen Zielgröße das System im Regelfall arbeiten sollte.

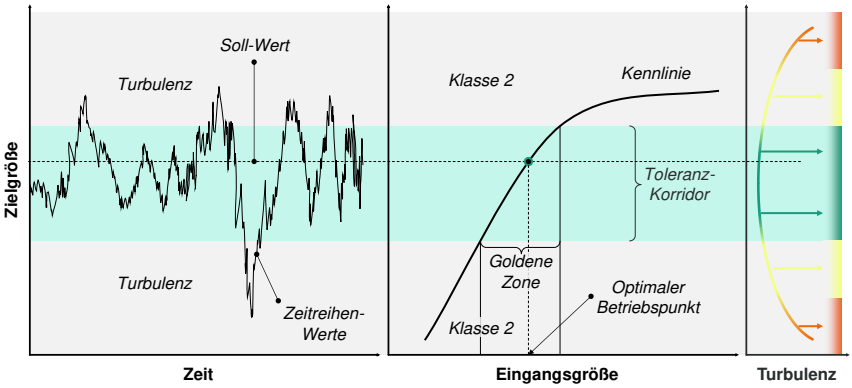


Abbildung 3.6: Zum Verständnis der Turbulenzbewertung

Optimaler Betriebspunkt

Der optimale Betriebspunkt ist derjenige, bei dem die zu optimierende Zielgröße dem Soll-Wert am nächsten kommt. Im klassischen Anwendungsfall sind dies die Maxima und Minima von Relativwerten wie dem Wirkungsgrad oder den Kosten je Einheit (Grote et al. 2018, S. M44). Im Fall von realen Produktionssystemen ist ein einzelner optimaler Betriebspunkt aus mindestens den folgenden Argumenten nicht plausibel: In Produktionssystemen wird meist mehr als eine Zielgröße verfolgt. Unterschiedliche Zielgrößen besitzen sowohl aus systemischen als auch aus mathematisch-logischen Gründen unterschiedliche optimale Betriebspunkte. Diese konkurrieren miteinander und beeinflussen je nach Gewichtung den globalen Betriebspunkt. Weiter, und der Punkt wiegt weit schwerer, sind Produktionssysteme eben nicht stationär, weshalb der optimale Betriebspunkt prinzipiell nicht festzulegen ist. Schuh et al. sprechen daher von der „Golden Zone", innerhalb der beispielsweise die Fertigungssteuerung den Betriebspunkt zu halten hat (Schuh & Fuß 2015, S. 105). Die Problematik konkurrierender Ziele greift Schickmair in seiner Arbeit auf, in der er den Begriff des optimalen Betriebspunktes der Mechanik auf die Produktion überträgt und ihn als die „der Zielsetzung entsprechenden idealen Kombination der teilweise konkurrierenden Einzelziele" definiert (Schickmair 2010, S. 23).

In Abbildung 3.6 werden von den unterschiedlichen Autoren beschriebenen Aspekte zur Turbulenzbewertung in einer Darstellung zusammengeführt. Jede messbare Zielgröße besitzt einen Soll-Wert innerhalb eines definierten Toleranzkorridors (grün). Im Zeitverlauf

können seine Ist-Werte naturgemäß auch außerhalb dieses Korridors liegen. Dann liegt definitorisch Turbulenz vor. Wenn die Zielgröße von einer Eingangsgröße funktionale Abhängigkeit aufweist, lässt sich eine Kennlinie finden, welche Ziel- und Eingangsgröße funktional verbindet. Mit dem Toleranzkorridor und dem Soll-Wert sind die „Golden Zone" und der optimale Betriebspunkt mathematisch bestimmt. Je nach Abweichung vom Sollwert ist somit der Grad der Turbulenz als Wert festgelegt und kann in beliebig viele (hier zwei) Turbulenzklassen klassiert werden.

Zusammenfassend weist die Literatur zur Turbulenzbewertung einige Lücken auf. So wird von Nyhuis et al. darauf hingewiesen, dass die Kennlinien eine idealisierte Beschreibung der Zusammenhänge sind und es sich in der Realität um mehr oder weniger dichte Verteilungen handelt (Punktwolken). Quellenübergreifend beschäftigt sich die Literatur vorwiegend mit der Ursachendetektion. Wenn Bewertungen vorgenommen werden, dann sind diese qualitativ und basieren auf Expertenwissen. Die quantitative Bewertung der nur mit hohem Aufwand oder praktisch nie gänzlich zu vermeidenden Turbulenz bleibt aus. Zentrale Lücke der bestehenden Ansätze ist die *Annahme von stationären Betriebszuständen*, was nicht den Realbedingungen existierender Produktionssystemen entspricht.

3.3 Prognose von Produktionskennzahlen

Auf die Bedeutung der Prognose im Themenfeld der Systemtheorie und die mathematischen Hintergründe wurde in Kapitel 2 bereits eingegangen. Die Vorhersage eines zukünftigen Zustandes oder Verhaltens eines Systems hat entsprechend hohe Relevanz für den Produktionsbetrieb. Cheng et al. fassen zusammen, dass Produktionsmanager „genaue Vorhersagen" über Produktionssysteme nach „kurzer Berechnungszeit" benötigen, um „Abweichungen, Fehler und unregelmäßigen Situationen" zu identifizieren (Cheng et al. 2018, S. 1). Der Zweck der Entscheidungsunterstützungsfunktion durch die Prognose ist damit impliziert. Im Abschnitt 3.2 zur Turbulenzbewertung wird gezeigt, dass die für Prognosen theoretisch vorausgesetzte Zeitstabilität (im historischen Datensatz gefundene Kausalzusammenhänge sind zeitunabhängig) in der Realität jedoch kaum erfüllt wird.

Prognoseverfahren lassen sich allgemein in vergangenheits- und zukunftsbasiert einteilen (Barthel 2006, S. 30). Neben intuitiven (Experten, Orakel) und graphischen (extrapolierte

Visualisierungen) sind die mathematischen Verfahren für die rechnergestützte Anwendung bedeutsam. Barthel führt die eingeschränkte Chance auf richtige Prognosen auf logisch limitierende Faktoren zurück. Abbildung 3.7 zeigt, wie aufgrund der Limitierung des Umfangs der vorhandenen, verfügbaren und verarbeitbaren Informationen und durch den Zufall die Prognosesicherheit (Anteil des Prognoseraums am Realitätsraum) sinkt. Hinzu kommt, dass nur Teilmengen der nötigen Informationen vorhanden, verfügbar und verarbeitbar sind. In allen anderen Fällen sind keine Prognosen möglich (Barthel 2006, S. 27; Ohl 2000, 216 f.).

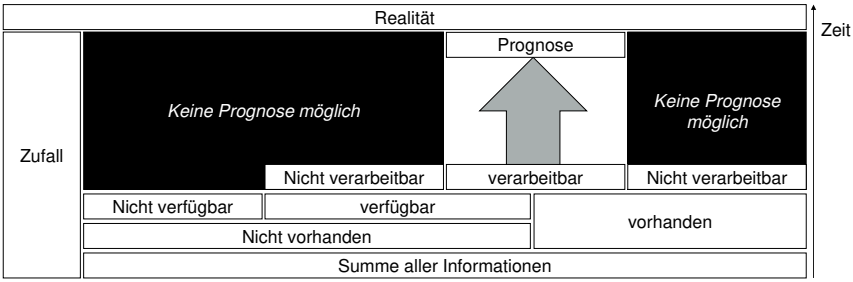


Abbildung 3.7: Prognosechance in Anlehnung an (Barthel 2006; Ohl 2000)

Die Problemstellung dieser Arbeit kann auf die Kennzahlenprognose eingeschränkt werden. Bei quasi-stabilen Zuständen sind Prognosen auf unterschiedlichen Wegen möglich und werden in sicherheitsrelevanten Systemen in den Bereichen Automotive, Luftfahrt oder Medizin eingesetzt (Nusser et al. 2009). ML findet entsprechend auch in der industriellen Produktion bei Prognosen seine Anwendung (Ruediger-Flore et al. 2008; Usuga Cadavid et al. 2020). Forschungsseitig ist die Kombination aus diskreter Event-Simulation und anschließendem Einsatz von ML-Verfahren zur Nutzung der generierten Daten verbreitet. Murphy et al. nutzen diskrete Event-Simulationen für die Generierung von Datensätzen über Durchlaufzeiten, um darauf hin verschiedene ML-Verfahren zu deren Prognose anzuwenden (Murphy et al. 2019). Sie nutzen das Modell einer linearen Prozesskette und betonen, dass die Übertragung auf reale Systeme und Erweiterung auf die ganze Supply Chain wesentliche Vorteile im Bereich der Produktionsplanung und -steuerung (PPS) ergäben, dies jedoch noch aussteht. Das Vorgehen ist etabliert und wird in der Produktionsforschung von vielen Autoren eingesetzt (Öztürk et al. 2006; Pfeiffer et al. 2016). Ihre Ergebnisse,

die ebenfalls auf Simulationsdaten beruhen, beziehen sich auf das WIP-Management. Zur Kombination von diskreter Event-Simulation und ML-Prognosen wird als Verarbeitungs- und Steuerungssoftware meist ‚Python' eingesetzt (Govaert et al. 2009).

Den Entscheidungsbäumen kommt – nicht zuletzt wegen ihrer flexiblen Einsetzbarkeit – in der Anwendung nach wie vor eine große Bedeutung zu. Ensemble-Methoden gelten dabei als die aktuell genauesten ML-Modelle, gerade im Hinblick auf XAI (Amit & Geman 1997; Barredo Arrieta et al. 2020, S. 94). Viele Arbeiten beschäftigen sich mit dem Vergleich der Leistungsfähigkeit unterschiedlicher Klassierer. Dazu wird meist eine bestimmte Kennzahl ausgewählt, welche mit verschiedenen ML-Methoden prognostiziert werden soll. Lingitz et al. beschäftigen sich dazu mit der Frage, welche logistischen Kenngrößen die Durchlaufzeit beeinflusst, um dann den jeweilig besten ML-Algorithmus zu finden (Lingitz et al. 2018). Die Algorithmen-Auswahl beinhaltet Feature-Reduktion. Lingitz et al. nennen unterschiedliche Fälle, in denen die Feature-Anzahl von mehreren Hundert auf zweistellige Zahlen gesenkt werden konnte, ohne nennenswert an Leistungsfähigkeit des Klassierers zu verlieren. Feature-Reduktion zielt darauf ab, die Rechenzeiten zu verkürzen und Over-Fitting zu vermeiden. In diesem Zusammenhang werden wieder Bagging und Boosting als die gängigsten Ensemble-Methoden genannt (siehe Unterabschnitt 2.3.3). Diese Optimierungen von Ensemble-Methoden führen jedoch zu Einbußen bei Rechenzeit, Speicherbedarf und Erklärbarkeit. Die genannten Publikationen ermöglichen einen Eindruck zu den Parametern für die Prognose von Produktionskennzahlen. Die Rechenlaufzeiten bewegen sich im Bereich von wenigen Minuten, die Feature-Anzahl ist meist zweistellig. Für Prognosen dieser Art genügen Ensembles mit weniger als 1000 Entscheidungsbäumen. Konkret bestätigt sich auch in der Arbeit von Lingitz et al. die Leistungsfähigkeit der Random Forests beim Vergleich des Normalized Root Mean Squared Errors (NRMSE) (Lingitz et al. 2018, S. 1054).

Analog zum Stand der Technikwissenschaft in der Turbulenzbewertung zeigt sich ein ähnliches Bild für die Kennzahlenprognose. ML ist für die Prognose von Produktionskennzahlen geeignet, verbreitet und anerkannt. Die einschlägigen Veröffentlichungen fokussieren stark auf PPS und die Auswahl des geeignetsten Algorithmus. Ensemble-Methoden zeichnen sich hier insgesamt als geeignet aus. Die Konsequenzen auf Prognosewerkzeuge durch Systemänderungen werden jedoch nicht adressiert oder explizit ausgeschlossen. Ledig-

lich Metan et al. sprechen davon, Entscheidungsbäume in Echtzeit „upzudaten", um Scheduling-Regeln aktuell zu halten (Metan et al. 2010, 6909 f.). Offen bleibt bei allen Autoren, welche Produktionsdaten konkret für die Features genutzt werden. Genannt werden lediglich Trainingsdaten, Simulationsdaten, Events, Zeitreihen, Attribute oder eben Features.

Produktionsdaten

Das in der Nutzung von Produktionsdaten liegende Potenzial ist unbestritten, die Festlegung eines monetären Wertes jedoch ist mindestens nicht trivial. Systematiken zur Datenwertbestimmung sind eine eigene Disziplin. Der Wert von Daten entsteht erst durch ihre Verwendung, nicht durch die reine Existenz. Voraussetzung für eine gewinnbringende Verwertung von Daten ist deren methodenabhängige richtige Auswahl, was eine sinnvolle Kategorisierung von Daten erfordert (Axmann et al. 2019; Preißler 2008). Die in der Produktion anfallenden Daten können auf vielfältige Weise strukturiert werden. Allgemein werden Produktionsdaten in Stamm- und Bewegungsdaten gegliedert. Daneben kann noch zwischen den Datenarten ‚Bestands- und Änderungsdaten' sowie ‚Anwendungs- und Kontrolldaten' differenziert werden. Stammdaten haben dabei die längste Gültigkeitsdauer und gelten unabhängig von Auftragslage und zeitabhängigen Ereignissen. Als auftragsneutrale Daten generieren sie jedoch hohen Pflegeaufwand, da sie als persistente Informationen nicht automatisch im Laufe der Zeit kontinuierlich ersetzt und dadurch korrigiert werden. Die Gegenspieler zu den Stammdaten sind die Bewegungsdaten, welche eben nicht produkt- sondern auftragsbezogen sind. Die bereits beschriebenen ‚Events' fallen zu einem großen Teil in diese Kategorie. Eine Zwischenrolle spielen Bestandsdaten, die zwar nicht auftragsbezogen, aber dennoch von kurzer Gültigkeitsdauer sind. Sie sind zustandsbeschreibend und beziehen sich meist auf Kennwerte und Mengen wie Kapazität, Auslastung und Bestände (Schuh & Schmidt 2014, S. 44). Im englischsprachigen Raum sind die Begriffe Online- und Offline-Daten geläufig. Online-Daten umfassen Maschinenstatus, WIP, Bestände usw. während Offline-Daten Grundinformationen über das Produktionssystem wie Prozesse, Stücklisten, Prozessparameter, Layout und Produktionsplane enthalten (Cheng et al. 2018, S. 2). Wöstman et al. kategorisieren Daten für Data Mining nach ihrem Ursprung mit externer oder interner Datenquelle (Wöstmann et al. 2017, S. 9). Es existiert bislang keine allgemeingültige Beschreibung für Produktionsdaten. Um Produktionsdaten für eine ge-

zielte Auswahl zur Kennzahlenprognose zu kategorisieren, werden, wie in Abbildung 3.8, die Gesamtmenge aller Produktionsdaten in Bewegungs-, Stamm-, Bestands-, Änderungs-, Kontroll- und Anwendungsdaten eingeteilt.

Abbildung 3.8: Datenarten in Anlehnung an Schuh et al.

Ausgehend vom Verständnis der Produktion als System sind für dessen Beschreibung Input-, Systemzustands- und Outputdaten notwendig (siehe Abbildung 2.4 und Unterabschnitt 2.2.3). Typische, das System anregende Daten sind damit solche, welche die Auftragslast beschreiben und hochgradig zeitabhängig (online) sind – also die Datenart der Bewegungsdaten. Für diese Arbeit sind das konkret die Auftragsdaten. Diese bestimmen das Systemverhalten und induzieren durch den Kunden potenziell Turbulenzen. Weniger zeit- und auftragsabhängig (offline) sind Stammdaten, die Auskunft über die Systemattribute enthalten. Sie beeinflussen gemeinsam mit den Bewegungsdaten den zeitlichen Verlauf der Bestandsdaten (daher wieder online). Bestandsdaten sind als beschreibende Kennzahlen von Werte- und Mengenstrukturen typische zu prognostizierende Indikatoren von Turbulenzen. Unter dieser Datenart sind daher im Sinne der Systemtheorie die Outputdaten zu erwarten. Der konkrete Einsatz dieser Kategorisierung der Produktionsdaten für die Turbulenzprognose und die tatsächlich verwendeten Kennzahlen werden in Abschnitt 5.2 zur Struktur des Datensatzes im Rahmen der Lösung beschrieben. Die drei in Abbildung 3.8 farblich markierten Datenarten sind als Ergebnis damit genau diejenigen, welche für Kennzahlenprognosen in der Produktion potenziell relevant sind.

Fazit:

Zur Bewertung von Turbulenzen in der Produktion finden sich in einigen – auch aktuellen – wissenschaftlichen Arbeiten Berührungspunkte zur Aufgabenstellung. Der inhaltliche Fokus liegt dabei auf den Turbulenzursachen und der qualitativen Bewertung sowie deren Relevanz. Die Bewertung der Turbulenzauswirkungen bleibt jedoch unvollständig. Die generelle Vermeidung von Turbulenzen in der Produktion steht stärker im Zentrum der Überlegungen als deren Messung. Quantitative Bewertungen bleiben daher weitgehend aus. Wenn Bewertungen erfolgen, basieren diese eher auf Expertenwissen, als auf Daten und sind daher qualitativer Art. Zur Prognose finden sich im Kontext der Turbulenz kaum spezialisierte Arbeiten. In Teilen wird die Wichtigkeit von Prognosen betont und die Schwierigkeiten bei der Umsetzung näher beschrieben. Die genannte zentrale Hürde ist die Dynamik in den Produktionssystemen. Die wird zwar identifiziert, jedoch kaum adressiert. Meist werden statische Systeme entweder implizit angenommen oder explizit vorausgesetzt. Der technische Reife der Turbulenzbewertung hat drei Mängel:

- die Ausarbeitung eines Bewertungsansatzes an sich
- das Vorgehen zur Informationsgenerierung für Turbulenzprognosen
- ein Ansatz zur Bewältigung der Problematik der Dynamik in Realsystemen

Um den Stand der Technik nach der Recherche in einer Übersicht darzustellen, sind die dafür geeigneten Bewertungskriterien herzuleiten und zu definieren. Diese ergeben sich aus den formalwissenschaftlichen Grundlagen zur Turbulenz, der nun identifizierten Forschungslücke und der zentralen Forschungsfrage. Zentrales Ergebnis der Untersuchungen zur Turbulenz ist der systemtheoretische Ansatz – konkret das darin verortete funktionales Konzept – als Zugang zur ‚Black Box' komplexer Systeme (Unterabschnitt 2.1.2). Dies führt zur notwendigen Betrachtung von Ursache und Wirkung in kausalen Systemen (Abbildung 2.9). Das Verständnis über Turbulenzursache und -wirkung sind die beiden ersten Kriterien. Mit dem Forschungsziel ‚Turbuelnzbewertung' ergibt sich automatisch die Frage nach qualitativer und quantitativer Bewertung. Genau hier liegt, wie in Abschnitt 3.2 beschrieben, die Forschungslücke, da qualitative Bewertungen – meist basierend auf Expertenwissen – beschrieben und durchgeführt werden, nicht so jedoch die quantitativen. Anhand dieser beiden weiter Kriterien soll die Literatur ebenfalls verglichen werden. Abschließend ergibt die

Forschungsfrage (Abschnitt 1.3) selbst die noch zu betrachteten Kriterien: Es wird verglichen, welche der bestehenden Turbulenzbewertungen nur auf den Ist-Zustand abzielen oder auch Turbulenz-Prognose („ex ante“) beinhalten, wo auch dynamische Systeme berücksichtigt sind („Produktion“) und ob diese Bewertungen „datenbasiert“ durchgeführt werden. Tabelle 3.1 zeigt den Erfüllungsgrad der jeweiligen Kriterien in der Literatur und bestätigt die Forschungslücke bei der quantitativen Prognose für dynamische Produktionssysteme.

Tabelle 3.1: Turbulenzbewertung in der Literatur

	Turbulenz-Ursache	Turbulenz-Wirkung	Subjektiv (Qualitativ)	Objektiv (Quantitativ)	Beinhaltet Prognose	Dynamisches System	Daten-basiert
Wiendahl	●	◔	●	◔	○	○	◔
Mintzberg	◕	◔	◕	○	○	◔	○
Schuh et al.	◕	◑	◕	◔	◔	◔	◕
Schickmair	◑	◑	◕	○	○	○	◑
Nyhuis et al.	◑	◑	◔	◑	◕	◔	◑

Die letzten (aus der Forschungsfrage abgeleiteten) drei Kriterien fächern sich im Hinblick auf die technische Umsetzung der Turbulenzbewertung in weitere auf. Die in Tabelle 3.2 betrachteten Kriterien sind Ergebnis des iterativen Prozesses der Überlegungen über die Forschungsfrage, des Studiums der vorhandenen Literatur und der Systemanforderungen in Kapitel 4. Die Kriterien der Ensemble-Methoden und der Produktionsdaten ergeben sich direkt aus dem Fazit zu den formalwissenschaftlichen Grundlagen (tree-based Ensembles sind geeignetster Lösungsansatz) und der Forschungsfrage mit dem (Anwendungsfeld Produktion). Die Literatur bestätigt die weite Verbreitung der ML-Verfahren allgemein und im speziellen, dass Ensemble-Methoden mit Entscheidungsbäume in der Anwendung im Produktionskontext weit etabliert und anerkannt sind. Auch der Anwendungsbezug zu Produktionsdaten ist vielfach erprobt, wenn auch nur auf bestimmte Kennzahlen, weniger auf ‚Kennzahlenbündel‘ (Set von konkurrierenden Zielkennzahlen), welche für Turbulenz-steckbriefe notwendig sind. Dieses Kriterium ergibt sich aus der Zielsetzung der Arbeit (1.3).

Erneut zeigt sich die zentrale Lücke im Umgang mit dynamischen Systemverhalten. Außer den Arbeiten von Lingitz et al. und Metan et al. befasst sich keine Arbeit explizit damit,

ob und wie die Prognosemodelle mit der Zeit anzupassen sind und welche Auswirkungen sich durch Systemänderungen auf die Auswahl der Trainingsdatensätze ergeben. Als letztes Kriterium wird noch die aus den Systemanforderungen sich ergebende Nutzung von Simulationsdaten betrachtet. In den wissenschaftlichen Arbeiten zur Kennzahlenprognose sind Simulationen ein verbreitetes Mittel, um valide Produktionsdaten zu generieren. Die Kombination von Python und PlantSimulation ist ein gängiges Vorgehen.

Tabelle 3.2: Datenbasierte Kennzahlenprognose in der Literatur

	Ensemble-Methoden	Produktionsdaten	Kennzahlenbündel	Dynamisches System	Nutzung von Simulation
Cheng et al.	●	◕	◔	○	○
Nusser et al.	○	◕	○	○	○
Ruediger-Flore et al.	◕	◑	○	○	○
Usuga Cadavid et al.	◕	◑	○	○	○
Murphy.	●	◕	◔	○	●
Pfeiffer et al.	◕	◑	◔	○	●
Govaert et al.	◕	◑	○	○	●
Lingitz et al.	●	◕	◔	◔	●
Metan et al.	●	◑	◑	◕	●

4 Systemanforderungen an die Lösung

Die Kapitel 2 und 3 zu Formalwissenschaft und zum Stand der Technik decken diejenigen Themenfelder ab, welche notwendiges inhaltliches Wissen, bereits bestehende potenziell einsetzbare Werkzeuge oder – soweit vorhanden – aktuelle Umsetzungen zur Turbulenzbewertung beinhalten. Sie bilden die Grundlage für sowohl den Lösungsansatz (Kapitel 5) als auch die Ableitung der **Anforderungen** an die Lösung, welche in diesem Kapitel erfolgt. Die Systemanforderungen ermöglichen eine zielgerichtete Lösungsentwicklung und stellen damit ein Zielsystem dar, gegen das die Lösung getestet werden kann. Der Autor nutzt dazu zum einen *Anforderungslisten* aus der Literatur für größtmögliche Vollständigkeit und zum anderen *Anwendungsszenarien*, um mögliche Anwendungen weitgehend zu konkretisieren und damit Detailanforderungen vor der Entwicklungsphase transparent zu machen.

Grundsätzlich sollte die Gestaltung technischer Systeme nicht ‚freihändig', sondern entlang eines methodischen Weges erfolgen (Erlach et al. 2020). Die Definition der Systemanforderungen ist ein wesentlicher Teil eines solchen Gestaltungsprozesses und wird in ihrer Bedeutung oft unterschätzt (Schroda 2000). Mit klar definierten Systemanforderungen und einem methodischen Vorgehen steigt die Chance, nicht nur *eine beliebige* aus der Vielzahl möglicher technischer Lösungen (‚Äquifunktionalität' (Ropohl 2012)) zu finden, sondern eine *möglichst gute*. Kesselring nennt für den Bereich Konstruktion fünf Gestaltungsprinzipien: Minimale Herstellungskosten, minimaler Raumbedarf, minimales Gewicht, minimale Verluste und günstige Handhabung (Kesselring 1954, 242 ff.). Gestaltungsempfehlungen solcher Art wurden von Pahl und Beitz seit den späten 1970er Jahren spezifiziert und weiterentwickelt (Feldhusen & Grote 2013). Die Ergebnisse dieser Arbeiten finden sich in den entsprechenden VDI-Richtlinien 2221 und 2223 wieder (VDI 1993; VDI 2004).

Ahrens definiert den Begriff Anforderung als „eine Vorgabe, deren Erfüllung den zielgerichteten Verlauf des jeweiligen Konstruktionsprozesses steuert und/oder Eigenschaften des betreffenden Produktes bestimmt" (Ahrens 2000, S. 213). An dieser Definition orientieren sich die Systemanforderungen in dieser Arbeit, welche in einer Auflistung der Anforderun-

gen resultiert. Eine solche Anforderungsliste für industrielle Produkte besteht nach Pahl und Beitz aus den Kernbereichen Identifikation, Organisation, Rückverfolgung und Inhalt. Da diese Arbeit keine bereits marktfähige Lösung sucht, beschränkt sie sich auf den Bereich ‚Inhalt'. Dieser gliedert sich in **Anforderungsbeschreibung**, **Anforderungsausprägung**, **Anforderungsart** und **Priorisierung** (Feldhusen & Grote 2013, S. 322).

Pahl und Beitz nutzen zur Erstellung und Vervollständigung von Anforderungslisten verschiedene Anforderungsquellen. Die Szenariotechnik ist eine bewährte Methode zum Erstellen und Ergänzen von Anforderungslisten. Diese Arbeit nutzt als Anforderungsquelle solche Anwendungsszenarien, weil diese durch die Perspektive externer Nutzerinnen und Nutzer den Anwendungskontext konkretisieren und so anwendungsnahe Aussagen möglich machen. Aus einer solchen Hauptmerkmalliste lassen sich durch Auswahl Fragen ableiten, mit denen Szenarien beschrieben werden können (Feldhusen & Grote 2013, S. 331). Für die Lösung dieser Arbeit sind folgende Fragen relevant:

- Wer benutzt die Lösung?
- Wo wird sie eingesetzt?
- Welche Art der Bedienung ist gewünscht?
- Welche Rechenleistung steht zur Verfügung?
- Welche Daten stehen zur Verfügung?
- Welche Dynamik besteht im betrachteten Produktionssystem?
- Wie robust soll die Lösung sein?
- Wie soll das Ergebnis visualisiert werden?
- Welche Fehlertoleranz ist gewünscht und noch akzeptabel?

Anhand dieser Fragen können ein oder mehrere Szenarien erstellt werden, innerhalb derer die zu entwickelnde Lösung angewendet werden soll. Die Fragen dienen der möglichst vollständigen Beschreibung eines Szenarios hinsichtlich ihrer für die Lösung relevanter Attribute. Die Szenarien werden in Abschnitt 4.1 beschrieben, Abschnitt 4.2 leitet die funktionalen Anforderungen an die Lösung ab. Diese sollen nach Möglichkeit die Kriterien

Korrektheit, Eindeutigkeit und Umsetzbarkeit nach Pahl und Beitz erfüllen (Feldhusen & Grote 2013, S. 338).

4.1 Anwendungsszenarien

Die Antworten auf die neun Fragen aus der einleitenden Passage ergeben einen morphologischen Kasten mit Attributen, deren plausible Kombinatorik zu drei Szenarien führen. Auf die ausführliche Beschreibung der Szenarienerstellung wird hier verzichtet. Das Ergebnis wird folgend beschrieben und ist in Abbildung 4.1 als Übersicht dargestellt.

Szenario 1: Proof of Concept

Im ersten Anwendungsszenario wird die Position lernender Forscherinnen und Forscher aus dem Bereich der Ingenieurs- und IT-Wissenschaften eingenommen. Ihr spezifisches Themenwissen ist allgemein hoch und sie sind in der Lage, sich den Umgang mit neuen Lösungen schnell anzueignen. Der Interessenfokus liegt auf dem grundsätzlichen Verständnis der Lösung und einer möglichen Übertragbarkeit auf andere Anwendungsfälle. Für sie stehen eine ästhetische und intuitive Bedienung nicht im Vordergrund, manuelle Ausführung von Programmskripten stellen keine Schwierigkeit dar. Die Verwendung gängiger Programmiersprachen ist geläufig und das für die Programmausführung notwendige Wissensniveau kann schnell erworben werden. Prinzipiell stehen zwar hohe Rechenkapazitäten im Forschungsumfeld zur Verfügung, es werden jedoch Lösungen bevorzugt, die auf herkömmlichen Rechnern ausgeführt werden können. Motivation und Zweck ist dabei, kurzfristig selbst Experimente, Adaptionen und Weiterentwicklungen durchzuführen und deren Ergebnisse schnell auswerten zu können. Datenbasis dieses Szenarios sind Simulationsdaten. Solche Daten, generiert mit beliebig detaillierten Modellen von Produktionssystemen, sind vergleichsweise realitätsnah bei gleichzeitig hoher Datenqualität. Zudem ist ihr Gesamtvolumen leicht skalierbar. Dieses erste Szenario soll laborähnlich ein statisches Produktionssystem beobachten. Die umgesetzte Lösung muss dafür keine hohe Robustheit aufweisen, da in einer Laborumgebung Programmabstürze keine weitreichenden Folgen haben, der Ablauf standardmäßig unter Beobachtung erfolgt und wegen erwartbar regelmäßigen Anpassungen der Aufwand einer hochrobusten monolithischen Programmgestaltung nicht den

Nutzen aufwiegt. Die Forscherinnen und Forscher erwarten jedoch Daten-Logs und deren Visualisierung, um weitergehende Analysen zu ermöglichen. Die Fehlertoleranz der Lösung ist im Szenario ‚Proof of Concept' allgemein hoch.

Szenario 2: Prognose bei Systemdynamik

Dieses Anwendungsszenario adressiert den Kernaspekt der Forschungsfrage dieser Arbeit: Die Prognosegüte von Klassierern bei Systemdynamik. Anwender sind ebenfalls die Forschenden von Szenario 1, nun jedoch mit dem Interessenfokus auf dem Verhalten der Lösung unter speziellen Bedingungen, und damit nicht der Lösung an sich. Um möglichst große Bereiche des Parameterraumes dynamischer Systeme abzutasten, bedient sich die Lösung statistischer Datensätze. Diese werden basierend auf den Erfahrungswerten aus Szenario 1 in großem Umfang synthetisch generiert. Mit generierten Datensätzen kann auch die Systemdynamik selbst gesteuert und somit zielgerichtet untersucht werden. Dies soll durch Repräsentierung unterschiedlicher Systemzustände und deren Übergänge ineinander im Zeitverlauf erfolgen. Anforderungen an Robustheit und Visualisierung entsprechen denen aus Szenario 1. Die Fehlertoleranz rückt jedoch in den Fokus, da die Leistungsfähigkeit der Lösung hier evaluiert werden soll.

Szenario 3: Realsystem

Für das dritte Szenario, in welchem reale Produktionsdaten verwendet werden, wird die Rolle von Angestellten des Produktionsmanagements eines Industrieunternehmens eingenommen. Wenngleich hohe Rechnerleistungen mit Cloud-Technologien vergleichsweise leicht verfügbar sind, bevorzugen die Anwenderinnen und Anwender aus Gründen der Praktikabilität leistungsoptimierte Lösungen. Dem Szenario entsprechend werden hier solche Daten als Trainings- und Testdaten verwendet, die in einem realen Produktionssystem im Betrieb erhoben wurden. Dementsprechend wird das Produktionssystem als eine Black Box betrachtet, über die (zunächst) keine Kenntnisse über deren Systemdynamik vorliegt. Dieses Szenario erfordert für die Lösung eine hohe Robustheit gegenüber inkonsistenten Daten. Weiter sind im industriellen Kontext detaillierte Auskünfte über die Leistungsfähigkeit der Lösung selbst und deren einzelne Parameter nötig. So sollte sich die Visualisierung auf die gewünschten Kennzahlen beschränken und die Turbulenzen übersichtlich in einer

ansprechend gestalteten visuellen Anwenderschnittstelle darstellen. Fehlertoleranzen sind hier wegen des wirtschaftlichen Interesses an der Umsetzung entsprechend klein und damit ein wesentliches Kriterium für die Evaluierung der Lösung.

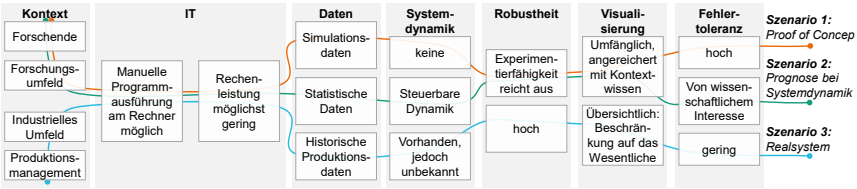


Abbildung 4.1: Drei Anwendungsszenarien für die Lösung

Die drei Anwendungsszenarien konkretisieren den Anwendungskontext und ermöglichen so das Ableiten der **funktionalen Anforderungen** an die Lösung. Die Arbeit strebt *eine* Lösung an, die alle hier beschriebenen Anwendungsszenarien abdeckt.

4.2 Funktionale Anforderungen

Zur Lösungsentwicklung sind Anforderungen für die beiden folgenden Lösungskomponenten zu definieren. Diese sind zunächst die funktionalen Anforderungen an das **Produktionsmodell** zur Datengenerierung mittels Simulation und weiter die Anforderungen an die Gesamtlösung für die **Turbulenzbewertung**.

Zunächst erfolgen die Erläuterungen zum Simulationsmodell. Fricker nennt vier „verschiedene grundlegende Anforderungen [...], die an ein Instrumentarium zur Modellierung, Analyse und Gestaltung zu richten sind“ (Fricker 1996, S. 85). Dazu gehören erstens die Abbildung von Ordnungsstrukturen durch Objekt(mengen) sowie deren Zustände und Relationen. Weiter soll zweitens das Modell durch graphisch-symbolische Darstellungsweise die Beschreibung unterstützen. Dann muss es drittens möglich sein, die Stabilität des Systems im Modell zu analysieren, um so Ursache für Komplexitäten identifizieren zu können. Schließlich sollte viertens die Aussagefähigkeit über gegenwärtigen Zustand und Struktur für die anwendungsorientierte Gestaltung von Produktionssystemen gegeben sein.

Da das Simulationsmodell volatile Variantenfertigungen (vergleiche Abbildung 1.8) repräsentieren soll, entsprechen die Attribute meist der variantenreichen Serienproduktion,

umfassen jedoch auch kleine Auflagenhöhen und geringe Wiederholhäufigkeiten. Die technischen Anforderungen sind als Überblick in Tabelle 4.1 **priorisiert** gesammelt und beinhalten sowohl Frickers Anforderungen als auch die aus den Anwendungsszenarien des vorausgehenden Abschnitts.

Wo notwendig, folgen nun nach Pahl und Beitz für ausgewählte Punkte der Tabelle 4.1 die **Anforderungsausprägungen** (Landherr 2014, S. 53). Die beiden Anforderungsarten Allgemeingültigkeit und Skalierbarkeit erfordern im Modell Anpassungsmöglichkeiten bei den Parametern Produktportfolio, Anzahl der Prozessressourcen und Verteilung der Auftragslast. Das Produktportfolio soll Produktfamilien und deren kundenspezifische mehrteilige Varianten abbilden. Die Ablaufart bei Montage und Fertigung sollte in der Ausführung in Gruppen, Linien oder im Fluss möglich sein. Die Ressourcen sollten in ihrer Anzahl skalierbar und die Prozesse in beliebiger Reihenfolge modellierbar sein, wodurch auch die Wertschöpfungstiefe abbildbar wird. Für die Auftragslast (Nachfrageverlauf) sollten Stückzahl, Produkt und die Verteilung über die Zeit (glatte, zyklische und irreguläre Komponente gemäß Unterabschnitt 2.2.1) statistisch einstellbar sein. Eine gesonderte Modellierung der PPS wird ausgeschlossen, da Regeländerungen in der Auftragseinlastung eine weitere Parametervariation bedeutet, die der möglichst hohen Transparenz beim Proof of Concept schadet. Die Dispositionsart bleibt damit im Simulationsmodell immer dieselbe: Aufträge werden nach Zieltermin sortiert und so eingesteuert. Zur Vollständigkeit dieser Attribute für das Produktionsmodell orientiert sich der Autor an den Fertigungscharakterisierungen mehrerer einschlägiger Quellen (Dürrschmidt 2001, S. 21; Lödding 2016, S. 626; Schuh & Stich 2012, S. 169).

Tabelle 4.1: Technische Anforderungen an das Simulationsmodell in Anlehnung an Landherr; priorisiert von hoch nach niedrig

Anforderungsart	**Beschreibung**	**Begründung**
Akkuranz und Relevanz	Angemessen genaue Abbildung der Realität	Grundvoraussetzung für das Modell
Richtigkeit	Syntaktisch, logisch und semantisch korrekte Konzeptualisierung der Realität	Voraussetzung für korrekte Systemabbildung in generierten Daten
Analysierbarbeit	Zugriffsmöglichkeit auf Ablauf- und Zustandsdaten während und nach der Simulation	Der so generierte Datensatz wird für die Untersuchung verwendet
Anpassbarkeit	Manuelle oder automatische Veränderung von Modellparametern	Einfache Modellanpassung für Abtastung unterschiedlicher Systemzustände
Allgemeingültigkeit	Prinzipielles Potenzial, beliebige Produktionssysteme abbilden zu können	Wissenschaftliche Motivation
Integrierbarkeit	Schnittstellendefinition von Simulations- und Auswertesoftware	Automatisierung von Datengenerierung, Auswertung und Visualisierung
Modularisierbarkeit	Zerlegung und Neukombination funktionaler Modell- und Programmeinheiten	Dient Modellskalierung und Programmrobustheit
Skalierbarkeit	Anpassung der Größe des modellierten Produktionssystems	Abtastung unterschiedlicher Ressourcen-, Prozess-, Varianten- und Auftragsanzahlen
Visualisierbarkeit	Darstellung von Zustand und Struktur des Modells	Dient Kontrolle und Steuerung während der Simulation und der Plausibilisierung und Evaluation der Ergebnisse

Die Anforderungen an die Gesamtlösung für die *Turbulenzbewertung* ergeben sich ebenfalls aus den Anwendungsszenarien von Abschnitt 4.1 und den für die Lösung in dieser Arbeit notwendigen Komponenten für Klassierungen mit ML. Für die Lösung sind erstens geeignete **Datensätze**, zweitens ein zweckdienlich gestalteter **Klassierer** und letztlich ein Werkzeug zur **Evaluation** der Lösung notwendig. Die einzelnen Anforderungen der Komponenten werden analog zu denen des Simulationsmodells in Abbildung 4.2 dargestellt und folgend beschrieben.

Datensatz

Die in dieser Arbeit verwendeten Datensätze haben die nach Wichtigkeit priorisierten Anforderungen der allgemeinen Nutzbarkeit zu erfüllen. Sie müssen auftrags- und systembeschreibend sein und Informationen so beinhalten, dass sie mit Klassierungswerkzeugen verarbeitet werden können. Der erste Punkt erscheint zwar selbstverständlich, jedoch liegt mit Werkzeugen in den Bereichen Big Data, Data-Stream Processing, Unsupervised Learning und Automated Data Mining ausgereifte Software vor, die auch weitgehend unstrukturierte Datensätze verarbeiten kann. Im Sinne der Nutzbarkeit sollen hier solche Datensätze verwendet werden, welche durch ihr begrenztes Volumen auf verbreiteten Speichermedien verwaltet und mit herkömmlichen Rechnerleistungen verarbeitet werden können. Für den universellen Einsatz im wissenschaftlichen und industriellen Umfeld sollen nur gängige Dateiformate verwendet werden, deren Semantik eine einfache Les- und Interpretierbarkeit garantiert. Der Datensatz sollte aufgrund der vorliegenden Fragestellung im Bezug auf dynamische Systeme die Möglichkeit bieten, fortwährend verändert und erweitert zu werden. Dies betrifft besonders die Datensätze für die Untersuchungen bei dynamischem Systemverhalten und den Realdatensatz mit historischen Produktionsdaten. Grundvoraussetzung für die Nutzbarkeit ist ein Mindestmaß an Informationsgehalt. Dieser muss sowohl das Produktionssystem als auch die zu bewertenden Produktionsaufträge mit Attributen abbilden. Für Aussagen über die Systemdynamik ist es erforderlich, dass die Daten mit Zeitstempeln verknüpft sind, um allen Datenwerten Zeitpunkte und damit auch Zeitintervalle zuordnen zu können. Um Klassierungen möglich zu machen, sollten abschließend die Attribute in Features repräsentiert und die Aufträge mit Targets ‚gelabelt' sein. Bei Erfüllung dieser Anforderungen kann der Datensatz für Turbulenzprognosen verwendet werden.

Klassierer

Der Klassierer erfordert – ausgehend von den Anwendungsszenarien – eine einfache Bedienbarkeit über Eingabeschnittstellen, jedoch nicht mit hoher Priorität. Es folgt die Anforderung, die eigenen Attribute transparent zu machen und Parameter über die Verwendung des Datensatzes, der angenommenen Latenzen und das Prognoseergebnis – also die bewertete Turbulenz – graphisch darzustellen. Hohe Wichtigkeit haben Robustheit und Aktualität. Diese resultieren aus dem Anwendungsbezug im Industrieumfeld. Robustheit ist besonders bei der Verarbeitung ständig aktualisierter Daten und bei Modellaktualisierungen notwendig. Das Modell muss damit vor allem die Grundanforderung der Richtigkeit erfüllen. Dazu sind aus den Überlegungen zur Systemtheorie die Systemdynamik selbst, die Berücksichtigung von Latenzzeiten zwischen Ursache und Wirkung, die selektive Auswahl bestimmter Samples für den Trainingsdatensatz und die Gewichtung von Zielkennzahlen für die Turbulenzbewertung notwendig. Hier besteht wieder der Bezug zur Anforderung an eine Eingabeschnittstelle. Insgesamt ergeben sich aus dem Klassierer zusammen mit den Anwendungsszenarien die Anforderungen an die Gestaltung der Evaluation, welche nun beschrieben wird.

Evaluation

Das Evaluationswerkzeug sollte Aussagen über den Zusammenhang von Datensatz- Attributen und dem Prognoseergebnis ermöglichen. Dazu gehören neben den Größenverhältnissen von Trainings- zu Testdatensatz auch die Beschreibung, welche – und auf welche Weise – einzelnen Samples als Trainings-Samples ausgewählt wurden. Das ist gerade für die Untersuchungen zum Prognoseverhalten bei dynamischen Systemen notwendig. Weiter sollten Aussagen über den Informationsgehalt des Datensatzes, also seine Verwertbarkeit allgemein, möglich sein. Wichtig sind zudem die Aussagen zum Modell, wonach bewertbare Aussagen zu dessen Eigenschaften wie Feature-Anzahl, Feature-Relevanz, sowie Baumtiefe und -anzahl enthalten sein sollen. Neben den Attributen sollen Aussagen zum Modellverhalten hinsichtlich Reaktivität und Stabilität bei Systemveränderung möglich sein. Dies beruht auf der hohen Wichtigkeit der Robustheit und Aktualität des Klassierers. Zuletzt sind Bewertungen zur Prognose an sich erforderlich. Diese Anforderung ergibt sich vor allem aus Szenario 3. Die klassischen Bewertungskennzahlen von Klassierern sind dabei Score und Probability (siehe 5.4). Für diese Arbeit ist das Verhalten dieser Kennzahlen sowohl bei

statischen als auch dynamischen Systemen und Aussagen über den Rechen- und Zeitbedarf relevant.

Fazit:

Dieses Kapitel beschreibt die Systemanforderungen an die in dieser Arbeit angestrebten Lösung. Die Herleitung der Anforderungen orientiert sich an den Ausführungen von Pahl und Beitz. Das Vorgehen folgt deren Vorschlag, aus Anwendungsszenarien die Anforderungen abzuleiten. Ergebnis sind die drei Szenarien Proof of Concept, Prognose bei Systemdynamik und Realsystem. Für alle Szenarien wurden anhand von Leitfragen Aussagen zu Nutzerrollen, Anwendungsumfeld, IT-Bedarf, Anforderungen an die Daten, Beschreibung der Systemdynamik, Robustheits- und Visualisierungsanforderungen sowie Fehlertoleranz beschrieben.

Von den Anwendungsszenarien ausgehend wurden die Systemanforderungen sowohl für das Simulationsmodell als auch die Umsetzung als Lösung für die Turbulenzbewertung abgeleitet. Vier Aspekte sichern dabei die inhaltliche Vollständigkeit. Zunächst die Anforderungsarten, in denen meist als alleinstehender Begriff eine oder mehrere Anforderungsbeschreibungen zusammengefasst werden. Die einzelnen Anforderungsausprägungen wurden beschrieben und auch die für die Umsetzung vorgesehene Priorisierung genannt.

Ausgehend von diesen Anforderungen wird im Folgekapitel der Lösungsentwurf erarbeitet und beschrieben. Wie für die Erarbeitung der Anforderungen orientiert sich auch die Gestaltung an den drei Grundregeln von Pahl und Beitz: „Gestalte eindeutig, einfach und sicher!“ (Feldhusen & Grote 2013, S. 583)

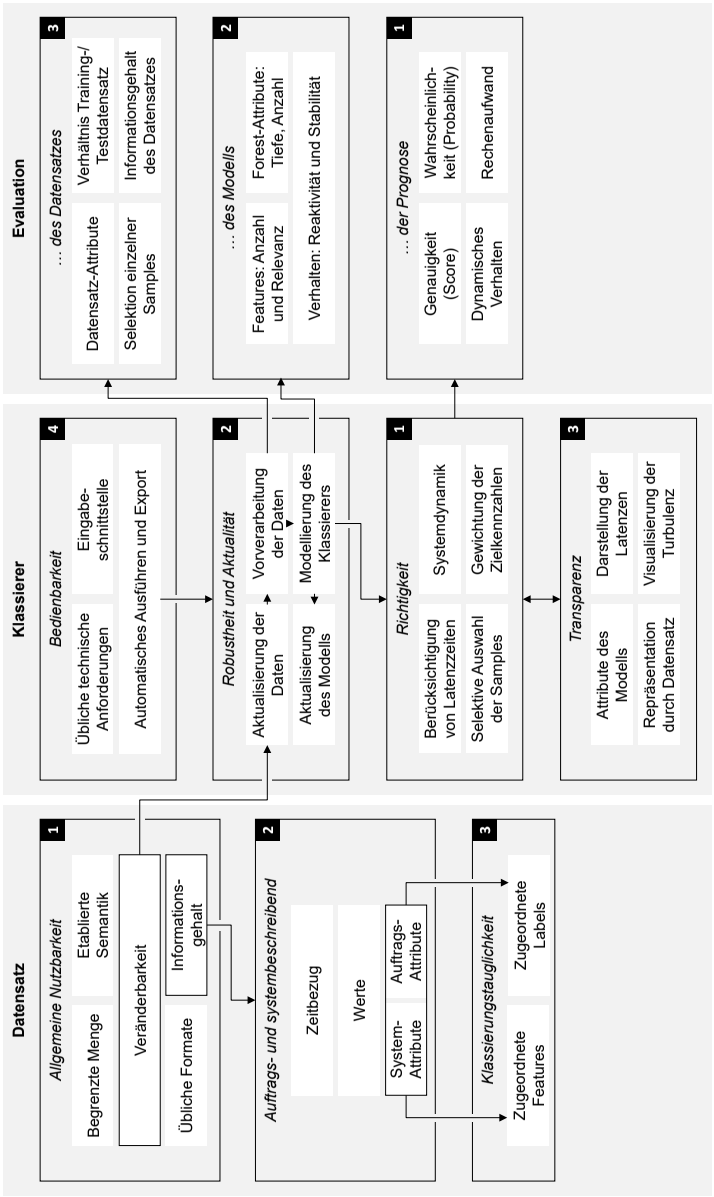


Abbildung 4.2: Anforderungen an die Lösung: Datensatz, Klassierer und Evaluation (Priorisierung: von 1–hoch bis 4–niedrig)

5 Datenbasierte Turbulenzbewertung

Zur Lösung des vorliegenden Problems werden die Erkenntnisse aus den vorangehenden Kapiteln genutzt. Ausgangspunkt ist laut Problemstellung, dass Prognose und Bewertung von Turbulenz dynamischer Produktionssysteme notwendig, jedoch aufgrund deren Komplexität nicht trivial sind (Abschnitte 1.1 und 1.2). Das Ziel, solche Turbulenzen dennoch zu bewerten und zu prognostizieren, führt zur Forschungsfrage, ob und wie dies datenbasiert erfolgen kann (Abschnitt 1.3). Der ingenieurwissenschaftliche Ansatz (Abschnitt 1.4) dieser Arbeit adressiert die volatile Variantenfertigung und orientiert sich daher bei der Parameterauswahl und der Datensatzerstellung an typischen Attributen dieses Fertigungstyps (Abschnitt 1.5). Die Lösungshypothesen (Abschnitt 1.6) münden in der Notwendigkeit, einerseits Produktionsdaten und andererseits Systemmodelle für diese Aufgabe wissenschaftlich zu betrachten.

Der Turbulenzbegriff wurde in seiner physikalischen Bedeutung formal eingeführt und seine Verwendung im Industriekontext erläutert (Unterabschnitt 2.1.1). Turbulenz bettet sich in die Theorie zu komplexen Systemen ein (Unterabschnitt 2.1.2 und 2.1.3) und wird im Rahmen dieser Arbeit als Verhältnis zwischen Systemkomplexität und Systemresilienz verstanden. Es folgt die Erkenntnis, dass Komplexität, Resilienz und die systemimmanenten Relationen nicht vollständig beschrieben werden können und Untersuchungen der Ursache-Wirkungs-Beziehungen eine Möglichkeit darstellen, Prognosen zu erstellen. Diese Arbeit nutzt daher mathematische Werkzeuge zur Beschreibung funktionaler Abhängigkeiten mit Zeitreihen und Modellierung kausaler, diskreter Systeme (Abschnitt 2.2). Der formalwissenschaftliche Teil schließt mit einem Überblick zum Thema KI ab (Abschnitt 2.3), in dem mit ML ein vielversprechendes Lösungsprinzip vorgestellt wird. Die vergleichende Betrachtung der gängigen Klassierungsmethoden lässt baumbasierte Ansätze als valide Möglichkeiten zur Lösung erscheinen (Unterabschnitt 2.3.2). Das in Unterabschnitt 2.3.3 beschriebene Grundprinzip der Entscheidungsbäume wird daher für die Lösung verwendet.

Zur konkreten Lösungsfindung erfolgt die Auswahl einer technischen Definition der Turbulenz für den Kontext dieser Arbeit in Abschnitt 3.2. Damit ergibt sich die genaue Gestalt

des Prognoseergebnisses und die Notwendigkeit, verschiedene DMT auf ihre Tauglichkeit für die Lösung zu untersuchen. Die Verwendung historischer Daten als Erfahrungswissen (CBR) und die Verarbeitung von Ereignissen zur Datensatzerstellung (CEP) erscheinen vielversprechend (Abschnitt 3.1). Zur Lösungsfindung ist ein Prozessmodell nötig. Diese Arbeit verwendet eine kombinierte Adaption von CRISP-DM und ASUM-DM und orientiert sich an der Datenpipeline von Papp bei der Gestaltung des Werkzeugs für den kontinuierlichen Betrieb. Die Betrachtungen zu den aktuellen Kenntnissen zur Prognose von Produktionskennzahlen legen nahe, dass die genaue Auswahl der Daten für die Anwendung auf dynamische Systeme von zentraler Bedeutung (Abschnitt 3.3).

Die Lösung adressiert die Systemanforderungen aus Kapitel 4. Direkten Einfluss auf die Gestalt der Lösung haben die Anwendungsszenarien in deren Bereichen ‚Daten', ‚Systemdynamik' und ‚Visualisierung' (Abschnitt 4.1). Der Szenario-Kontext wird implizit berücksichtigt, während ‚Robustheit' und ‚Fehlertoleranz' durch die jeweils höhere Anforderung bestimmt werden. Anforderungen an die IT werden automatisch durch Entwicklung und Betrieb auf entsprechender Hardware sichergestellt. Auf die detaillierten funktionalen Anforderungen (Abschnitt 4.2) von Modell und Werkzeug wird bei deren Beschreibung eingegangen.

5.1 Lösungsansatz

Als Prozessmodell für die Lösung werden CRISP-DM und ASUM-DM verwendet. So können aus der Gliederung des Forschungsprozesses die konkreten Inhalte für das Lösungsvorgehen im Kontext von Datenprojekten hergeleitet werden (Abbildung 5.1). In der ‚übersetzten' Kombination beider Modelle entspricht das Lösungskapitel dem Design-Teil von ASUM-DM, welches durch CRISP-DM detailliert wird. Es enthält damit die Gestaltung des verwendeten Datensatzes in seiner Struktur (Data Understanding, Abschnitt 5.2) und den anwendungsfallspezifischen Eigenschaften, welche diverse Operationen vor der eigentlichen Datensatzverwendung notwendig machen (Data Preparation, Abschnitt 5.3). Simulations- und Prognosemodell sind dem Modeling zuzuordnen (Abschnitte 5.5 und 5.4).

Die Umsetzung in Kapitel 6 wird dann wieder dem ‚Configure & Build' zugerechnet. Die inhaltliche Arbeit schließt mit der Evaluation in Kapitel 7 und stellt das Ergebnis als

Ganzes dar, welches zum Ende zusammengefasst wird (Deployment, Kapitel 8). Operate & Optimize erfolgt dann außerhalb des Rahmens dieser Arbeit im Industrieeinsatz.

Damit ist die Kombination beider Prozessmodelle für dieses Kapitel 5 als ‚Design-Phase' strukturgebend. So gliedert sich der Lösungsansatz analog zur statistischen Methodenlehre in zwei Teile: Den deskriptiven Teil, in welchem das Produktionssystem durch einen Datensatz repräsentiert wird und den induktiven Teil, in dem von Daten ausgehend auf die Gesamtheit geschlossen und ein Modell des Systems erstellt wird (Holland & Scharnbacher 2010, S. 3).

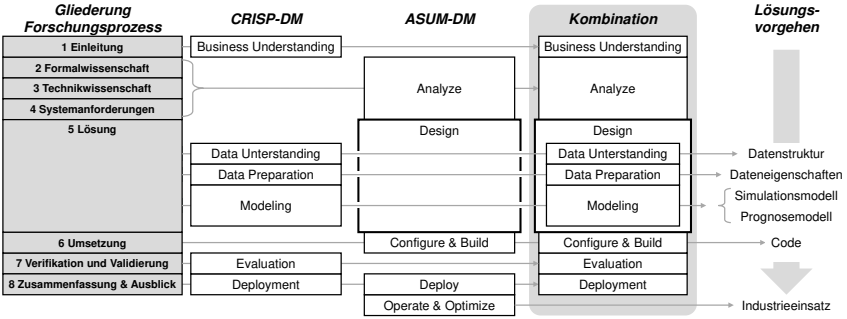


Abbildung 5.1: Herleitung des Lösungsvorgehens

Der im Abschnitt 1.3 beschriebene **Zweck** der Nutzbarmachung von Produktionsdaten zur Turbulenzbewertung wird durch Turbulenzsteckbriefe (siehe Abbildung 1.4) erfüllt. Für jeden Kundenauftrag wird vor seinem Produktionsstart ein Turbulenzsteckbrief erstellt, in dem die durch die Produktionssituation und den Auftrag selbst verursachte Auswirkungen auf ausgewählte KPIs beschrieben werden. Je nach strategischem Produktionsziel unterscheiden sich Anzahl und Auswahl der KPIs (Kennzahlenbündel). Diese Arbeit zielt nicht auf möglichst genaue Werteprognosen ab (Regression), sondern auf die richtige Zuordnung der Aufträge zu einer von drei Turbulenzklassen: niedrig, mittel und hoch (Klassierung). Grund für die Auswahl der Klassierung als Supervised-Learning-Methode ist zum einen der begrenzte Bedarf an Genauigkeit. So ist beispielsweise lediglich die Information relevant, *dass* ein Liefertermin eingehalten werden kann und nicht, *wie genau* die dafür erwartbare Durchlaufzeit (DLZ) ist. Der andere Grund liegt im Verfahren selbst. Mit der Genauigkeitsanforderung des Prognosewertes steigt die dafür notwendige Klassenanzahl,

was die Leistungsfähigkeit des Prognosewerkzeugs erheblich senkt (Böhm et al. 2021b, S. 128).

Die oben angesprochene Klassenanzahl weist auf das **Ziel** im Hinblick auf das Arbeitsergebnis hin. Dieses soll ein funktionales Prognosemodell sein, mit dem Produktions- und Auftragsdaten kontinuierlich verarbeitet werden können, um neue Aufträge einer Turbulenzklasse je nach Zielgröße zuzuordnen. Dies entspricht einem typischen Klassierungsproblem für überwachtes Lernen. Aufgrund des Vergleichs der Eigenschaften verschiedener klassischer Lernverfahren fällt die Entscheidung auf Random Forests (siehe Unterabschnitt 2.3.2). Die Grundidee zur Lösung entspricht also einem induktiven Ansatz: Aus den Informationen über eine große Anzahl von Produktionsaufträgen können Regeln extrahiert werden, die ein Prognosemodell bilden. Durch die Nutzung des Regelwerks ermöglicht das Prognosemodell (deduktiv) Aussagen zu bestimmten KPIs.

	Zweck	Ziel	Methode	Design, Configure & Built: Datensatz	Informationsbedarf	Datensatzgenerierung	Umsetzung
Arbeitsobjekt	KPI-Voraussage zu Aufträgen	Funktionales Prognosemodell	Supervised Learning	Struktur des Modell-Inputs	Systembeschreibung	Signalverarbeitung	Prognose-Werkzeug
Ausprägung	Turbulenzsteckbrief	Klassierung mit Ensembles	Klassierung: Random Forests	Features und Targets	Auftrags- und Zustandsgrößen	Korrelation und Latenz	Ausführbares Skript

Abbildung 5.2: Arbeitsobjekte und deren Ausprägung

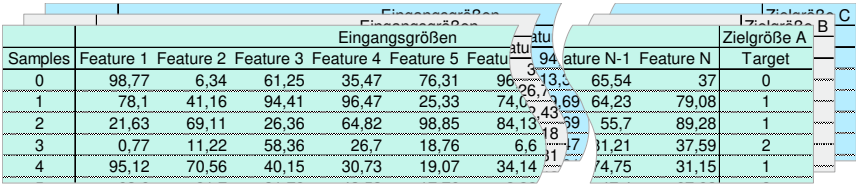
Mit der Entscheidung für baumbasierte Methoden ist die Gestalt des Modell-Inputs, also der Informationsbedarf, abstrakt definiert: Benötigt werden Datensätze mit einzelnen Samples, bestehend aus Features und zugeordneten Targets. Aus den Überlegungen zur Systemtheorie (Unterabschnitt 2.1.2) ergeben sich als Features auftrags- und systemcharakterisierende Daten und als Targets die Klasse der Zielgröße. Neben der damit festgelegten Struktur des Datensatzes kann der Informationsbedarf abgeleitet werden. Dieser beinhaltet Auftrags- und Zustandsgrößen des Systems. Die korrekte Zuordnung von auftragsbeschreibenden zu systembeschreibenden Datenpunkten ist für die Detektion von Korrelationen bei gleichzeitiger Latenz zwischen den Zeitreihen essenziell. Dafür werden die in Abschnitt 2.2 beschriebenen mathematischen Prinzipien zur Signalverarbeitung verwendet. Der Lösungsansatz mündet in der Umsetzung als ausführbares Programm, welches den Weg über den Rohdatensatz, die Datenverarbeitung und das Prognosemodell hin zum Turbulenzsteckbrief geht. Abbildung 5.2 fasst alle für die Lösung relevanten abstrakten Arbeitsobjekte und deren

konkrete Ausprägung zusammen. Die einzelnen Punkte zu Datensatz, Informationsbedarf und Datengenerierung sind der Designphase zuzuordnen und werden nun tiefergehend beschrieben.

5.2 Struktur des Datensatzes

In diesem Abschnitt wird die Gestalt des Prognosemodell-Inputs, also der konkrete **Datensatz**, entwickelt. Entsprechend der Grundstruktur, beschrieben im vorangestellten Abschnitt, besteht der Lerndatensatz aus einzelnen Samples, welche Features und Targets enthalten. Im Kontext dieser Arbeit entspricht ein Sample einem Produktionsauftrag mit den auftrags- und den systembeschreibenden Attributen. Diese Attribute werden durch die Features repräsentiert, welche den Aufbau des Prognosemodells bestimmen. Jedes Sample repräsentiert damit einen in der Vergangenheit real abgearbeiteten Produktionsauftrag. Seine Features beinhalten sowohl Informationen über den Auftrag selbst, als auch den Zustand der Produktion zum Zeitpunkt seiner Bearbeitung. Diese systembeschreibenden Attribute werden aus Zeitreihen (siehe Unterabschnitt 2.2.1) entnommen. So wird für jeden Auftrag der zeitabhängig passende Wert ausgewählt und dem Sample zugeordnet. Da die Zeitreihen bestimmte einzelne Systemattribute beschreiben, repräsentiert die Kombination der Werte aller Attribute zu einem bestimmten Zeitpunkt den Systemzustand der Produktion insgesamt. Dieser Zustand hat entsprechenden Einfluss auf die zu prognostizierende Zielgröße[1]. Für jede Zielgröße (= KPI im Steckbrief) ist ein Lerndatensatz notwendig, da jeweils ein eigenes Prognosemodell zu erstellen ist. Der Zielgrößenwert jedes Samples wird Target genannt und ist im Anwendungsfall dieser Arbeit eine von drei Turbulenzklassen (Abbildung 5.3). Zielgrößen umfassen alle im Turbulenzsteckbrief individuell als relevant erachteten KPIs. Mit den sowohl auftrags- als auch systembeschreibenden Attributen ist der kontextbezogene **Informationsbedarf** für die Turbulenzprognose abgedeckt. Prinzipiell ist es möglich, bei baumbasierten Methoden Werte mit unterschiedlicher Skala zu nutzen (siehe 2.2.1). Damit können für die Prognosemodelle prinzipiell alle Arten von Informationen mit einbezogen werden, wenn sie entweder Auftragsbezug haben, oder als Wert einer Zeitreihe über den Zeitpunkt einem Auftrag zugeordnet werden können. Zudem ergeben

[1] Siehe dazu die Darstellung einer „Enzephalografie des Unternehmens“ von Beer (Abbildung 1.3).



	Eingangsgrößen								Zielgröße A
Samples	Feature 1	Feature 2	Feature 3	Feature 4	Feature 5	Featu	ature N-1	Feature N	Target
0	98,77	6,34	61,25	35,47	76,31	96	65,54	37	0
1	78,1	41,16	94,41	96,47	25,33	74,0	64,23	79,08	1
2	21,63	69,11	26,36	64,82	98,85	84,13	55,7	89,28	1
3	0,77	11,22	58,36	26,7	18,76	6,6	1,21	37,59	2
4	95,12	70,56	40,15	30,73	19,07	34,14	74,75	31,15	1

Abbildung 5.3: Grundstruktur des Lerndatensatzes

einige Features weitere Sub-Features. So enthält jede Zeitangabe neben dem eigentlichen Zeitstempel zusätzliche Informationen wie Schicht, Wochentag, Monat und Quartal. Je nachdem, wo die Systemgrenzen gezogen werden, sind dann aus den betreffenden Zeitreihen Zustände des Systems als Features abbildbar. Typische Systemzustände sind die Angaben zum WIP oder dem Kapazitätsangebot zum jeweiligen Zeitpunkt. Damit werden auch andere systembeschreibenden Attribute wie Temperatur, Luftfeuchtigkeit, Jahreszeit oder auch zeitliche Nähe zu Feiertagen oder Sportereignissen denkbar. Abbildung 5.4 enthält eine nicht abgeschlossene Auflistung von für Prognosen nutzbaren Produktionsdaten. Die hier genannten Eingangs- und Zielgrößen sind eine repräsentative Auswahl typischer Größen im Produktionskontext. Diese Auswahl basiert auf den Erkenntnissen aus dem Abschnitt zur Turbulenzbewertung (Abschnitt 3.2), kombiniert mit den in Abschnitt 1.4 beschriebenen Erfahrungen im Entdeckungszusammenhang mit einer Vielzahl praxisnaher Industrie- und Forschungsprojekte.

Damit unterscheidet sich der Datensatz in drei Punkten von klassischen Datensätzen für baumbasierte Klassierer. Zunächst ist erstens zur Problemlösung nicht *ein* Prognosemodell notwendig, sondern *eines je KPI*, um mit dem Set an Prognosen den Turbulenzsteckbrief für einen Auftrag generieren zu können. Das stellt keine besondere Schwierigkeit in der Umsetzung dar, muss aber bedacht werden. Zweitens sollen mit dem Datensatz Prognosemodelle für *dynamische Systeme* generiert werden. Das hat zur Folge, dass Samples mit der Zeit ihre Gültigkeit verlieren und damit der Lerndatensatz in Teilen invalide wird. Die Modelle unterscheiden sich also nicht nur im Hinblick auf die zu prognostizierende Zielgröße, sondern auch in der Dimension Zeit. Als drittes sind die Latenzen zwischen Ursache und Wirkung zu nennen. Eine Erhöhung der Anzahl an freigegebener Aufträge führt nicht unmittelbar zu verlängerten DLZs. Somit können Systemattribute zeitlich nicht

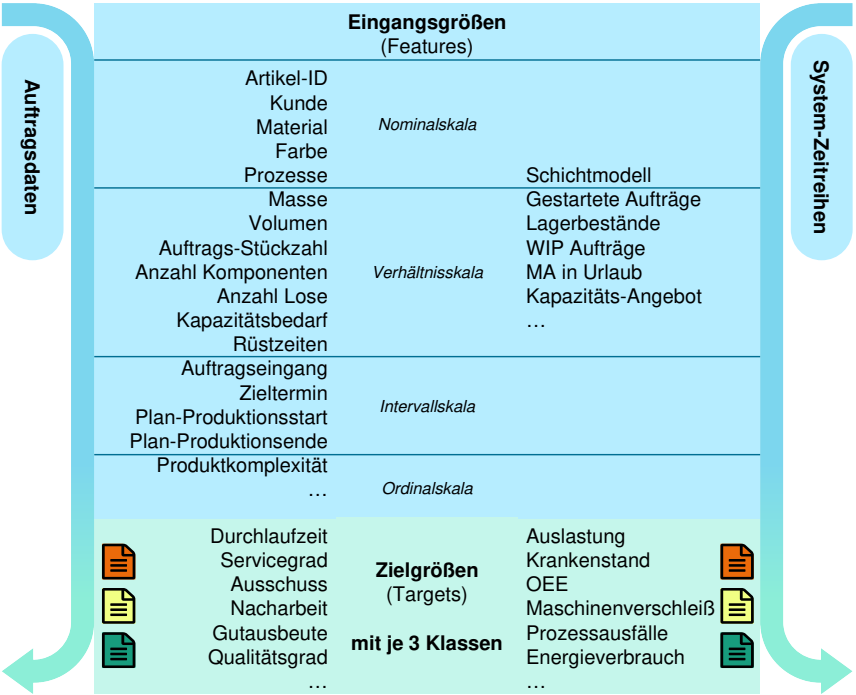


Abbildung 5.4: Für Klassierung nutzbare Produktionsdaten (Böhm et al. 2021b, S. 126)

direkt den aktuell laufenden Aufträgen zugeordnet werden. Jedes Feature, welches aus den Zeitreihen der Systemattribute entnommen wird, hat für *jede einzelne* Zielgröße eine andere, zunächst noch unbekannte, Latenzzeit. Diese und weitere für den Anwendungsfall dieser Arbeit speziellen Eigenschaften des Datensatzes werden im folgenden Abschnitt 5.3 behandelt.

5.3 Erforderliche Eigenschaften des Datensatzes

Dieser Abschnitt geht auf notwendige Eigenschaften des Datensatzes und damit auf die heterogenen Attribute unterschiedlicher Features und Targets im Produktionskontext ein. Für die Erstellung des Prognosemodells sind die einzelnen Eingangs- und Zielgrößen in die von Abschnitt 5.2 gezeigte Struktur zu überführen. Ziel ist es dabei, die relevanten

Eigenschaften des Rohdatensatzes zu identifizieren und den Umgang damit so zu definieren, dass die **Datensatzgenerierung** korrekt durchführbar wird. Abbildung 5.5 visualisiert exemplarisch vier Features (zwei auftrags- und zwei systembeschreibende) und *ein* ihnen zugeordnetes Target über die Zeit. Zur Unterscheidung sind *einzelne Aufträge* unterschiedlicher Skalenarten farbig markiert. Die Werte der Features (Kreise) ergeben zusammen mit dem zugehörigen Target (Rauten) ein Sample.

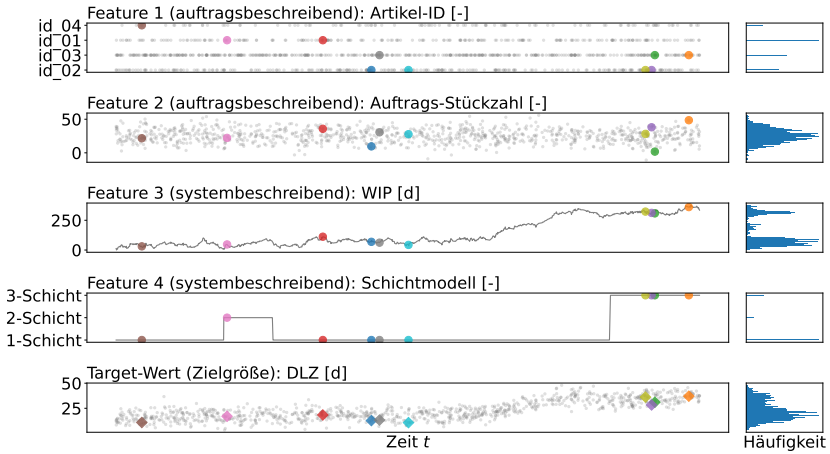


Abbildung 5.5: Visualisierung von Features und Targets unterschiedlicher Skalen und Attribute

Abbildung 5.5 zeigt die Wichtigkeit der korrekten zeitlichen Zuordnung der Aufträge zu den Systemzuständen. Bei auftragsbeschreibenden Features ist die Zuordnung trivial, da die Auftragseigenschaften inhärent und damit zeitunabhängig sind. Die systembeschreibenden Features eilen den die Targets beeinflussenden Ursachen zeitlich voraus. Die in der Abbildung einheitliche vertikale Zuordnung aller Features und Targets ist also nicht per se gegeben und muss erst erstellt werden. Für die Datengenerierung ist daher auf folgende Punkte zu achten:

1. Abtastrate der Datenpunkte nicht äquidistant
2. Mögliche Gleichzeitigkeit von mehreren Ereignissen
3. Realzeit entspricht nicht der Fabrikzeit
4. Latenzzeit zwischen Ursache und Wirkung für jedes Feature unterschiedlich

Zu Punkt 1: Die ungleichmäßige Abtastung von Zuständen und die zeitkontinuierliche Aufnahme von Ereignissen führt dazu, dass Anzahl und Reihenfolge der Datenpunkte nicht automatisch eine korrekte Zuordnung erlauben. Daher müssen auch für die eigentlich zeitunabhängigen auftragsbeschreibenden Features Zeitstempel mitgeführt werden. Aus zwei Gründen soll als referenzieller Auftragszeitpunkt einheitlich der Auftragsstart in der Produktion gelten. Zum einen wurden in Kapitel 4 alle Einflüsse aus der PPS ausgeschlossen, weshalb Reihenfolgeänderungen vor der physischen Materialbewegung nicht berücksichtigt werden müssen. Zum anderen sind bei der Prognose von noch nicht abgeschlossenen Aufträgen keine anderen Zeitpunkte (wie etwa DLZ und damit der mittlere Verweilzeitpunkt des Auftrags in der Produktion) bekannt. Mit der kritischen Abtastung (Nyquist-Frequenz, Formel A.21) kann eine Abtastrate gefunden werden, die alle Datenpunkte eindeutig unterscheidbar macht. Bei (hoch wahrscheinlich) eng beieinander liegenden Zeitstempeln wird diese Rate jedoch schnell sehr groß, was die Datenmenge unverhältnismäßig stark erhöht. In dieser Arbeit werden daher als untere und obere Schranken für die Abtastraten 1/Schicht und 6/Stunde genutzt. Das entspricht zwischen 10.000 und 500.000 Datenpunkten je Feature für einen Beobachtungszeitraum von etwa zehn Jahren. Da Ereignisse und Zustände sich naturgemäß in kleineren Zeitintervallen als die hier genannten ändern, folgt damit auch rein systematisch das Zusammenfallen mehrerer Zeitstempel.

Zu Punkt 2: Gleichzeitige Ereignisse müssen kontextbezogen behandelt werden. Im Rahmen dieser Arbeit werden alle Ereignisse mit den Operationen *Anzahl*, *Summe* und *Mittelwert* verarbeitet. Folgende Beispiele erläutern dies: Mehrere gleichzeitig startende Aufträge gleicher Artikel-ID werden gezählt. Gleichzeitig erfolgende Lagerzu- und -abgänge werden aufsummiert und zum selben Zeitpunkt zurückgemeldete Auslastungen müssen (gewichtet) gemittelt werden. Auf diese Weise gehen auch bei niedrigen Abtastraten Einzelereignisse nicht verloren, die zeitliche Zuordnung wird lediglich weniger genau. Bei Unzulänglichkeiten in der Datenaufnahme (Beispiel: Fertigbuchen mehrerer Aufträge auf einmal) können in den Daten Peaks entstehen. Diese nicht systemrepräsentierenden Datenpunkte werden durch Berechnung des gleitenden Durchschnitts ausgeglichen. Dabei orientiert sich die Auswahl der Intervallgröße an einem geeigneten Vielfachen der Abtastrate. Im Fall von Ereignissen mit gegenseitiger Abhängigkeit (zum Beispiel Bearbeitungsstart und -ende)

gehen bei zu grober Abtastung und damit verursachter ‚Gleichzeitigkeit' Informationen verloren.

Zu Punkt 3: Die auf der Intervallskala liegenden Zeitpunkte beliebiger Eingangs- und Zielgrößen sind nicht direkt für die Prognose zu verwenden. So würden DLZs von zwei Fabriktagen bei einem dazwischen liegenden Feiertag mit drei und bei einem Wochenende mit vier Tagen bewertet werden. Daher müssen die Realzeitstempel in Zeitstempel der Betriebstage in der Fabrik transformiert werden. Abbildung 5.6 zeigt die Funktion für ein Kalenderjahr. Bei Daten aus Simulationen oder synthetischen Daten kann vorausgesetzt werden, dass das Produktionssystem 24/7 betrieben wird. Die Transformation der Zeitstempel wird vor allem bei der Verwendung realer Daten relevant. Dieser Punkt ist als sehr kritisch zu betrachten, denn selbst innerhalb des Produktionssystems sind die Betriebszeiten nicht immer gleich aktiv. Allein unterschiedliche Schichtmodelle und Pausenzeiten verursachen Unschärfen.

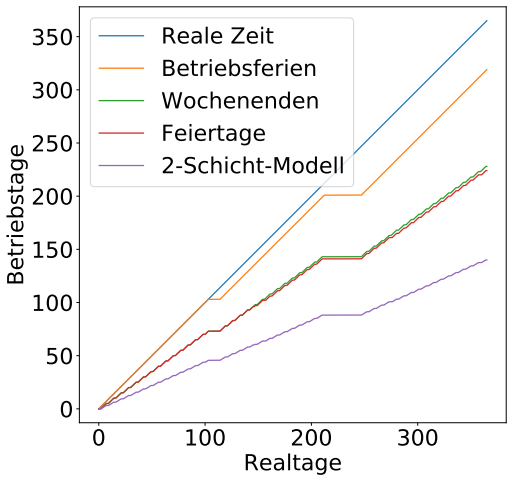


Abbildung 5.6: Fabriktage in Abhängigkeit von den Realtagen

Zu Punkt 4: Die Bestimmung der Latenzzeit zwischen Ursache und Wirkung hat mehrere Schwierigkeiten. Grundsätzlich ist, wie bereits gezeigt, Korrelation zwar ein Indikator, jedoch kein Nachweis für Kausalität. Gleichzeitig gilt, dass auch Scheinkorrelationen nützlich sind, solange sie gelten. Bei völlig unbekannten Ursache-Wirkungs-Beziehungen sind alle

Zeitreihen untereinander über alle Zeitverschiebungen abzutasten und auf funktionale Zusammenhänge zu untersuchen. Hohe Korrelationskoeffizienten sind damit nur ein Hinweis auf funktionale Zusammenhänge. Eben die können aber auch visuell an der Dichte ihrer Punktewolken erkannt werden. Hochdichte Punktewolken über einen großen Wertebereich deuten auf funktionale Zusammenhänge hin. Als letztes können die Vorhersagemodelle selbst für unterschiedliche (angenommene) Latenzzeiten erstellt und ihre Leistung untersucht werden. Die Latenzzeiten mit hoher Prognoseleistung versprechen die korrekten Ursache-Wirkungs-Verzögerungen wiederzugeben. Am Vorzeichen der Latenz ist Ursache und Wirkung zu unterscheiden (siehe ‚Kausalität' im Unterabschnitt 2.2.3). Um Rechenaufwände zu minimieren, kann die Detektion der Latenzzeit durch verschiedene plausible Einschränkungen begrenzt werden. Zielgrößen sind per Definition Messgrößen der Wirkung und treten daher zeitlich nachgelagert auf. Das halbiert bereits den Lösungsraum. Weiter müssen auch Latenzzeiten nicht beliebig genau bestimmt und damit mit einem Raster sinnvoller Größe abgetastet werden (siehe Punkte 1 und 2). Schließlich existieren noch plausible *maximale* Latenzzeiten, bei deren Überschreitung nicht mehr von kausalen Zusammenhängen ausgegangen werden kann. Diese liegen im Anwendungsfall dieser Arbeit bei einer Wochenanzahl im einstelligen Bereich.

Unter Berücksichtigung der oben aufgeführten vier Punkte hat der Datensatz die passenden Eigenschaften, um als Lerndatensatz für das Prognosemodell verwendet zu werden. Der folgende Abschnitt beschreibt den Weg vom Datensatz hin zum Prognosemodell.

5.4 Prognosemodell

Liegt ein vorverarbeiteter Datensatz mit den entsprechend oben beschriebenen Eigenschaften vor, kann für jede Zielgröße je ein Klassierer zur Prognose der Zielgrößen erstellt werden. Das Prognoseverfahren wird für die Umsetzung (Kapitel 6) als ausführbares Skript implementiert. Da diese Arbeit nicht für sich in Anspruch nimmt, die Prognoseverfahren an sich zu verbessern, werden hierfür bereits bestehende Programmmodule verwendet. Dieser Ansatz nutzt zur Prognose Entscheidungswälder. Andere Ensemble-Methoden werden nicht betrachtet (Begründung siehe Unterabschnitt 2.3.2). Der folgende Abschnitt geht

auf das Vorgehen ein, wie aus dem Datensatz Prognosemodelle für dynamische Systeme generiert werden.

Zunächst wird beispielhaft ein Prognosevorgang für *eine* Zielgröße mit künstlich erzeugten Daten aufgezeigt. Das Beispiel betrachtet entsprechend dem Anwendungsfall drei Klassen. Im Sinne der allgemeinen Gültigkeit des Vorgehens stehen die Klassen hier noch losgelöst vom Turbulenzbegriff, weshalb noch nicht die Farbkonvention Rot-Gelb-Grün verwendet wird. Im allgemeinen Fall werden n Features und das zu prognostizierende Target betrachtet. Abbildung 5.7 zeigt Zeitreihen von zwei Features und die zugehörigen Werte *eines* Targets. Die Target-Werte sind in drei Klassen Class_1, Class_2 und Class_3 eingeteilt und zur Unterscheidung entsprechend farblich gekennzeichnet. Die Grenzen zwischen den Klassen sind in der Abbildung zusätzlich durch die horizontalen Linien markiert. Es ist erkennbar, dass die Grenzen nicht auf die Feature-Zeitreihen übertragbar sind. Dies zeigt sich in der ‚Vermischung' der Datenpunkte unterschiedlicher Klassen in den Diagrammen für Feature 1 und 2.

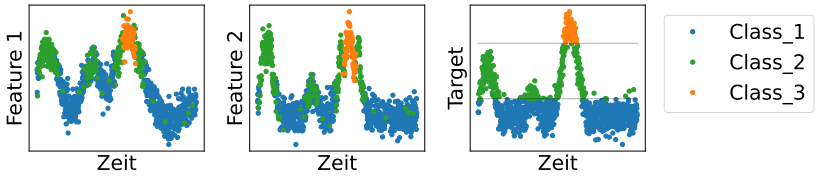


Abbildung 5.7: Zeitreihen eines Feature-Paares mit Target-Werten

Trotz der oben genannten Vermischung sind in Abbildung 5.7 funktionale Abhängigkeiten zwischen Features und Target zu erkennen. In diesem Fall werden hohe Feature-Werte auf hohe Target-Werte abgebildet. Es liegt damit potenziell eine Korrelation mit positivem Korrelationskoeffizienten vor. Die Darstellung der paarweisen Abhängigkeiten zwischen Features und Targets in Abbildung 5.8 bestätigen dies. Die Dichte der Punktwolke deutet auf eine korrekt identifizierte Latenz hin. Andernfalls würden den Targets falsche Features ohne kausalen Bezug zugeordnet werden, was sich in Punktwolken niedriger Dichte ohne erkennbare Korrelation niederschlägt. Folglich zeigen sich ebenfalls die scharfen Grenzen zwischen den Klassen in den Gegenüberstellungen von Features und Target. Die Vermischung ist bei der Gegenüberstellung der Features untereinander erkennbar. Zur Veranschaulichung

sind jeweils die Korrelationskoeffizienten r (Pearson-Korrelation, dimensionsloses Maß für lineare Zusammenhänge) mit angegeben. Es sei erneut darauf hingewiesen, dass dies zwar ein starker Indikator, jedoch nicht abschließend für die Bewertung und Auswahl der Zeitreihen-Latenz geeignet ist, da prinzipiell nicht-lineare Abhängigkeiten nicht nur möglich, sondern auch wahrscheinlich sind.

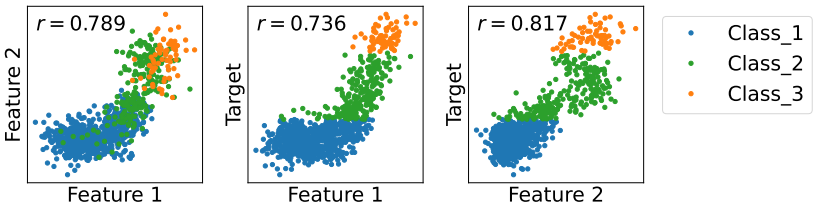


Abbildung 5.8: Darstellung der Abhängigkeiten zwischen Features und Target

Aus den einzelnen Samples, also die Kombination der Feature-Tupel und der Targets, kann nun ein Entscheidungsbaum abgeleitet werden (siehe Abbildung 5.9). Die Pfade enden entweder an einem Blatt mit eindeutiger Klassenzuordnung oder bei begrenzter Baumtiefe an Knoten mit Gini-Werten ungleich 0. Dies entspricht einer Zuordnung zu einer Klasse unter Unsicherheit. Die Knoten und Blätter weisen mit ihrer Farbe auf die jeweilige Klasse hin, weiße Knoten entsprechen einem Unentschieden. Im Anhang ist derselbe Entscheidungsbaum mit detaillierten Angaben zu Gini-Werten und Klassengewichten abgebildet (Abbildung A.14). Im Falle zweier Features kann der Entscheidungsbaum auch als Funktionswert im zweidimensionalen Raum interpretiert werden. Die Abbildungen 5.10 und 5.11 zeigen und bewerten zum einen die trainierten Prognosemodelle des obigen einzelnen Entscheidungsbaums und zum anderen die Klassierer als Ensemble (Entscheidungswald). Baum und Ensemble werden als farbig markierte Fläche repräsentiert. Im Ensemble-Fall repräsentiert die Farbsättigung die ‚Probability', also den Anteil korrekt zugeordneter Trainings-Samples am Ende eines Entscheidungspfades. An den in den Abbildungen geplotteten Testdatenpunkten ist zudem zu erkennen, welche der Test-Samples der Klassierer korrekt oder falsch zuordnet. Die zugehörige Score-Tafel gibt Auskunft über die Anzahl der jeweiligen Klassierungen der Samples zu den einzelnen Klassen. Diese Matrix stellt die Messung der einzelnen Fehlzuordnung dar und erlaubt so die Berechnung des

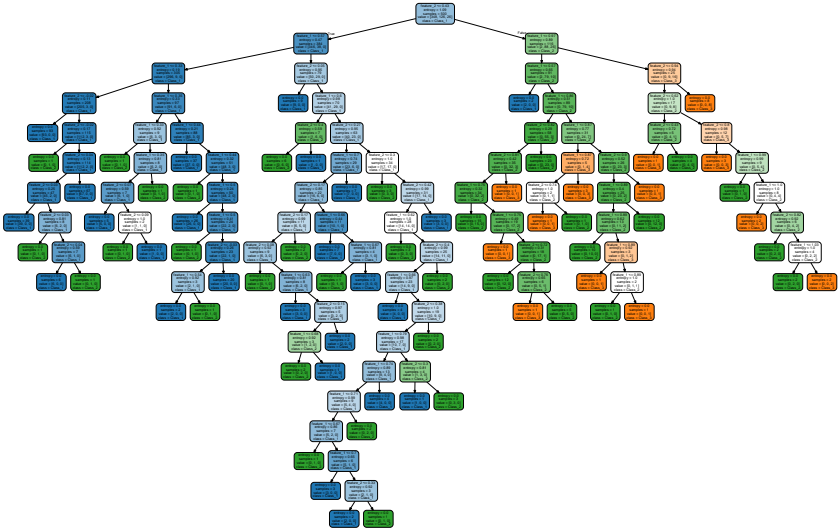


Abbildung 5.9: Prognosemodell: Entscheidungsbaum ohne Tiefenbeschränkung (Depth: 17)

Gesamt-Scores des Klassierers. Auf der Hauptdiagonalen befinden sich die Anzahlen der korrekt zugeordneten Samples, auf den jeweiligen von der Diagonalen abweichenden Elementen, die der Fehlzuordnungen. Die Bewertungen zeigen den Vorteil des Ensembles gegenüber dem einzelnen Baum im Score-Wert bei jeweils 500 Test- und Trainings-Samples. Der Anteil an korrekt zugeordneten Samples erhöht sich bei diesem Beispiel von 79 % auf 83,2 % bei wie erwartet von 100 % abweichender Probability. Das Modell nutzt Bagging-Methoden (‚`bootstrap = True`'). Der hier genutzte Trainingsdatensatz ergibt dabei Ensembles mit durchschnittlicher Baumtiefe von etwa 12 bei Nichtbeschränkung. Wie sich Ensembles unterschiedlicher Parameter bei diesem Trainingsdatensatz verhalten, stellt Abbildung A.15 im Anhang dar. Für das Verständnis der Lösung hier ist dies jedoch nicht weiter relevant. Eine Prognose von Produktionskennzahlen ist für statische Produktionssysteme damit grundsätzlich durchführbar. Struktur und Eigenschaften des Datensatzes sind beschrieben und die Erstellung des Prognosemodells dargelegt.

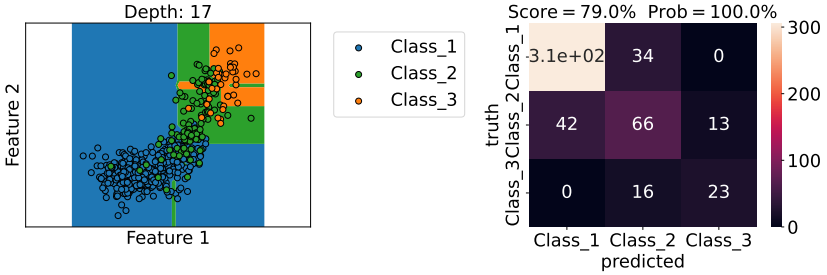


Abbildung 5.10: Zweidimensionale Modelldarstellung mit Score-Tafel (Baum)

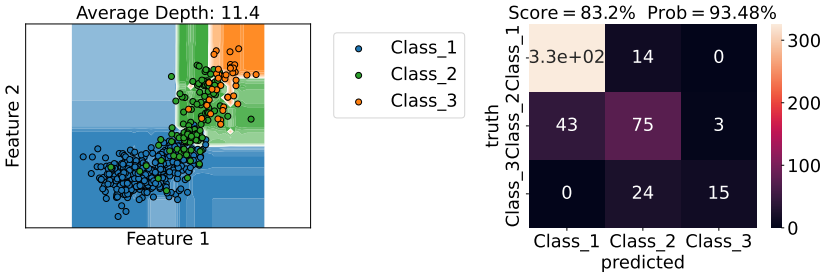


Abbildung 5.11: Zweidimensionale Modelldarstellung mit Score-Tafel (Ensemble)

Bei statischen Systemen werden vom zur Verfügung stehenden Trainingsdatensatz möglichst alle Samples für die Modellbildung verwendet. Bei dynamischen Systemen verliert der Trainingsdatensatz jedoch mit der Zeit seine Gültigkeit. Gültige und nicht mehr gültige Samples sind dabei nicht trivial voneinander zu trennen. Prinzipiell gilt aus den Überlegungen zur Systemtheorie für diese Lösung die Annahme, dass die Samples, welche zeitlich am kürzesten zurückliegen, die höchste Validität haben. So kann im vorliegenden Datensatz ‚rückwärts' gegangen werden, um einen maximal großen und gleichzeitig gültigen Trainingsdatensatz zu identifizieren. Durch Abtastung kann im eindimensionalen Lösungsraum der optimale Zeitpunkt gefunden werden, der den Trainingsdatensatz zeitlich begrenzt (siehe (Böhm et al. 2021a)). Bei Auflösung dieser Annahme, also unter Zulassung der Möglichkeit, dass Start- und Endzeitpunkt des ausgewählten Zeitintervalls beliebig sein können, wird das Problem zweidimensional. Jedoch besteht die Möglichkeit, dass weiter

zurückliegende und teilweise unterbrochene Zeitabschnitte ebenfalls Samples enthalten können, welche das Prognosemodell potenziell performanter machen. Bei Einteilung in n zu untersuchenden Zeitabschnitten bleiben $2^n - 2$ abzutastende Auswahloptionen. Die Option, keinen der Abschnitte zu wählen, bleibt als nicht plausibel außen vor. Die Auswahl aller Samples ist trivial und mit den statischen Systemen bereits abgedeckt.

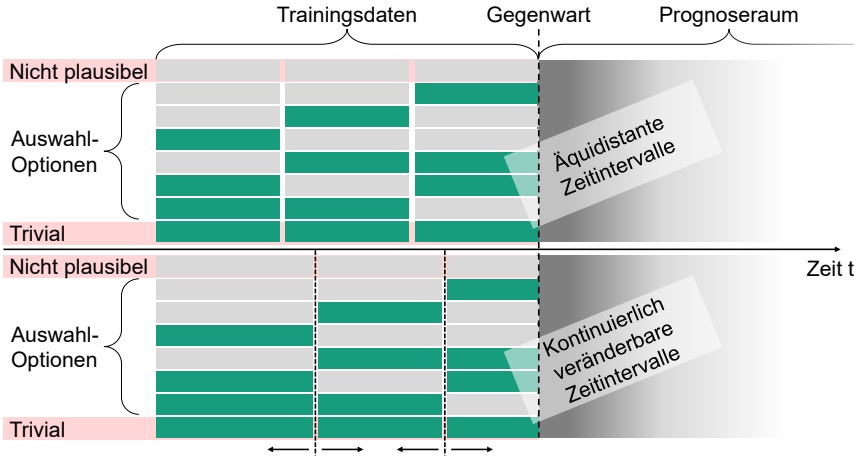


Abbildung 5.12: Möglichkeiten der Trainingsdatenauswahl

Abbildung 5.12 zeigt Möglichkeiten zur Einteilung und Auswahl der abzutastenden Samples. Grün sind dabei immer die für den Trainingsdatensatz genutzten Zeitabschnitte. Es wird eine rechenzeitadäquate Anzahl gleichgroßer Zeitintervalle abgetastet. Für große n und damit feiner Abtastung kann so jede existierende optimale Auswahl gefunden werden. Dies ist jedoch rechenintensiv – der Rechenaufwand steigt exponentiell. Daher werden für kleine n die Zeitintervalle kontinuierlich mit der Zeit verkleinert. Dies berücksichtigt erneut die Annahme, dass neuere Daten das gegenwärtige System besser repräsentieren. Die gegenwartsnahen Zeitintervalle sind kleiner und werden so differenzierter untersucht. Die potenziell weniger validen Samples in der entfernteren Vergangenheit werden mit relativ niedrigen Abtastraten behandelt.

5.5 Simulationsmodell

Für den ersten von drei Teilen der Evaluation soll die *prinzipielle Umsetzbarkeit* der Lösung an einem künstlich erzeugten Datensatz nachgewiesen werden. Die Datenakquise erfolgt hierfür über Simulationsdaten. So können die Bewegungsdaten in einem frei parametrierbaren Produktionsmodell aufgezeichnet werden.

Auch beim Simulationsmodell sollen die zehn technischen Anforderungen aus Tabelle 4.1 erfüllt werden: Akkuranz, Relevanz und Richtigkeit werden trivialerweise vorausgesetzt. Eine möglichst weitgehende Anpassbarkeit ist über Parametermanipulation gegeben. Durch größtmögliche Wertebereiche der Parameter können Modularisierung und Skalierbarkeit realisiert werden. Schnittstellen zu Parametrisierungs-, Auswerte- und Visualisierungssoftware erfüllen die pragmatische Anforderung der Integrierbarkeit. Über das eigentliche Modell hinaus kann bei einer Simulation die Analysierbarkeit über zwei Pfade realisiert werden: Zum einen die grafischen Darstellungen der Abläufe während der Simulation und zum anderen die Möglichkeit zur Aufbereitung der Exportdaten (Log-Files) danach. Das berührt weiter die Visualisierbarkeit, welche sowohl während als auch nach der Simulation gewährleistet sein sollte. Der letzte Punkt, Allgemeingültigkeit, hängt bei Modellen immer vom beabsichtigten Zweck in der Anwendung ab. Das erfordert im Kontext dieser Arbeit erstens die Variierbarkeit im Produktportfolio, zweitens eine realistische Repräsentation des Kundenbedarfs und drittens ein kapazitatives Abbild des Produktionssystems selbst. Die Auswahl der Modellparameter erschließt sich aus der Charakterisierung der volatilen Variantenfertigung in Anlehnung an Dürrschmidts typologische Merkmale (siehe Abschnitt 4.2) und die mathematische Beschreibung der Variabilitätskriterien von Wolf (Dürrschmidt 2001, S. 732; Wolf et al. 2019, S. 21). Diese beinhalten unter anderem die Variabilität des Auftragseingangs, des Mixes, der Menge, der Rüstzeiten, der Zykluszeiten, der Transportzeiten, der Verfügbarkeiten, der Ausfälle und der Wartezeiten. Durch die Parametergruppen Produktionssystem, Produktportfolio und Systemlasten werden die Merkmale im Modell berücksichtigt.

Im verwendeten Simulationsmodell bilden die Parameter zum **Produktportfolio** die Varianten unterschiedlicher Produktfamilien ab, welche sich in Produktionsprozessen, deren Reihenfolge, Rüstaufwände und Prozesszeiten unterscheiden. Damit ist Erzeugnisspektrum

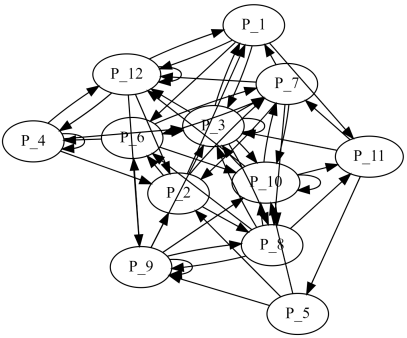
und -struktur sowie die Produktionstiefe definiert. Zudem bilden das **Produktionssystem** die Prozess- und Ressourcenanzahl sowie die Puffergrößen die wertschöpfende und logistische Kapazität ab (siehe Abbildung 5.13a, 5.13b und 5.13c). Diese wird über Ausfallwahrscheinlichkeiten, Verfügbarkeiten und weitere optionale Parameter statistisch modelliert. Damit sind Fertigungstiefe und Ablaufart in Fertigung und Montage als typologische Merkmale festgelegt. Die Parameter zu Auftragsanzahl, Saisonalität und Gewichtung der verhältnismäßigen Verteilung auf die Produktfamilien beschreiben die **Systemlast** (Merkmal des Nachfrageverlaufs). Die Auftragsauslöseart im Modell hier ist die der Produktion auf Bestellung mit Einzelaufträgen. Entsprechend den Anforderungen aus Abschnitt 4.2 sind im Modell die PPS, sowie Beschaffungs- und Dispositionsart nicht weiter ausgestaltet. Die Modellerstellung erfolgte unter Berücksichtigung der Vorgaben der entsprechenden VDI-Richtlinie 3633 (VDI 2018). Tabelle A.2 im Anhang stellt die veränderbaren Modellparameter als Übersicht dar.

Der Aufbau des Simulationsmodells und die Architektur der Software-Kombination von Modellkonfigurators und Simulation wird folgend beschrieben. Für die Anforderungserfüllung wird die Simulationssoftware Tecnomatix Plant Simulation (kurz: PlantSim) verwendet. Die Software ermöglicht Modellierung, Simulation und Visualisierung von Materialflüssen und logistischen Abläufen. Als Grundstruktur wurde ein Array von maximal 20 Prozessen mit jeweils maximal 20 Ressourcen vorgesehen (Abbildung 5.13d). Durch Festlegung von Prozess- und dazugehöriger Ressourcenanzahl werden einzelne Elemente des Arrays aktiviert. Die Prozessreihenfolge kann beliebig gewählt werden, wobei Schleifen und direkte Prozesswiederholungen ebenfalls möglich sind. Die Prozessreihenfolge definiert die Zugehörigkeit zu einer Produktfamilie.

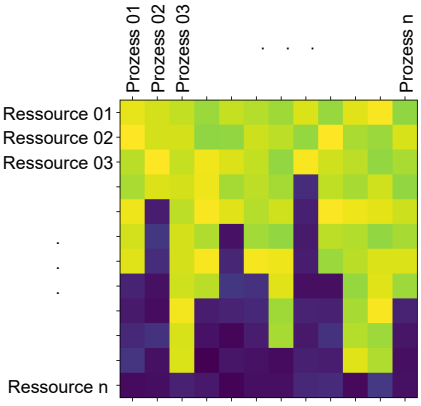
PlantSim wird hier zweistufig für Modellparametrisierung und Ergebnisexport verwendet. Importierte csv-Dateien legen alle Ressourcen- und Auftragsattribute im Simulationsprogramm fest. Diese generiert ein Python-Skript, welches die Anwenderschnittstelle zur Gesamtarchitektur darstellt (Konfiguration). Hier werden die Modellparameter zu Produktionssystem, Produktportfolio und Systemlast aus Tabelle A.2 in statistische Datensätze transformiert und von PlantSim verarbeitbare csv-Dateien exportiert (Modell). Danach startet das Skript die Simulation in PlantSim automatisch. Im ereignisgesteuerten Simulationspro-

	proc_id	proc_name	n_res	buf_size	mttr[days]	availab[%]	ct[sec]	sut[sec]
0	P_1	Name_P1	4	[5.0, 0.5]	[0.94, 0.09]	[67.49, 6.75]	[8745.4, 874.54]	[949.82, 94.98]
1	P_2	Name_P2	5	[5.0, 0.5]	[1.23, 0.12]	[79.01, 7.9]	[14507.14, 1450.71]	[1180.29, 118.03]
2	P_3	Name_P3	3	[5.0, 0.5]	[1.12, 0.11]	[74.64, 7.46]	[12319.94, 1231.99]	[1092.8, 109.28]
3	P_4	Name_P4	5	[5.0, 0.5]	[1.05, 0.11]	[71.97, 7.2]	[10986.58, 1098.66]	[1039.46, 103.95]
4	P_5	Name_P5	5	[5.0, 0.5]	[0.83, 0.08]	[63.12, 6.31]	[6560.19, 656.02]	[862.41, 86.24]

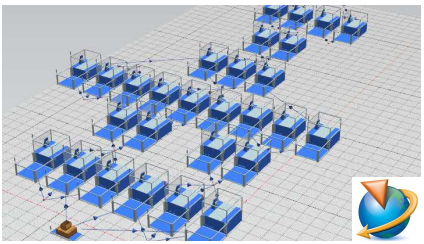
(a) Modellierung Einzelprozesse



(b) Prozessreihenfolge



(c) Kapazitäten



(d) Visualisierung Gesamtsystem

Abbildung 5.13: Exemplarische Auszüge aus dem Modellkonfigurator mit Visualisierung

gramm werden alle Einzelevents in Log-Dateien im txt-Format als Simulationsergebnis gespeichert. Diese werden mit einem weiteren Python-Skript für die Evaluation vorprozessiert und visualisiert. Mit dieser Architektur ist es möglich, ein Set an unterschiedlichen Simulationen automatisiert durchzuführen. Nutzerseitig ist nur die Parameterdefinition notwendig, die Verwendung des Visualisierungs-Skripts ist optional. Abbildung 5.14 zeigt den Gesamtaufbau der Architektur, welche nach dem Grundaufbau im Rahmen einer studentischen Arbeit intensiv getestet und an den Schnittstellen weiter optimiert wurde (Schmitt 2021).

Die Architektur soll die Verifikation des Lösungsansatzes auf folgende Weise ermöglichen: Beliebig komplexe statische Produktionssysteme können über Eingabe in ein Python-Skript modelliert werden. Um die Überlastung der Rechenkapazität zu vermeiden, sind Modelle bis zu 20 Prozesse mit je 20 Ressourcen möglich. Die Schnittstelle über csv-Dateien ermöglicht eine durchgängige Kontrolle und den Zugriff auf die Modellparameter. Mit PlantSim wird der Produktionsablauf über den vorbestimmten Zeitraum diskret simuliert und die Zustände in Einzelereignissen exportiert. Mit der Systemlast sind die Auftragsfeatures und mit den Einzelereignissen die Zustände, also die Systemfeatures, gegeben. Aus den Exportdateien lassen sich die Target-Werte extrahieren. Der so generierte Datensatz erfüllt die Voraussetzungen aus Abschnitt 5.2, kann somit in Trainings- und Testdaten aufgeteilt und für die Turbulenzprognose verwendet werden. Wie gut der Ansatz Voraussagen zu Systemkennzahlen ohne dezidiertes Systemwissen ermöglicht, wird für das Simulationsmodell im Abschnitt 7.2 zur Evaluation beschrieben.

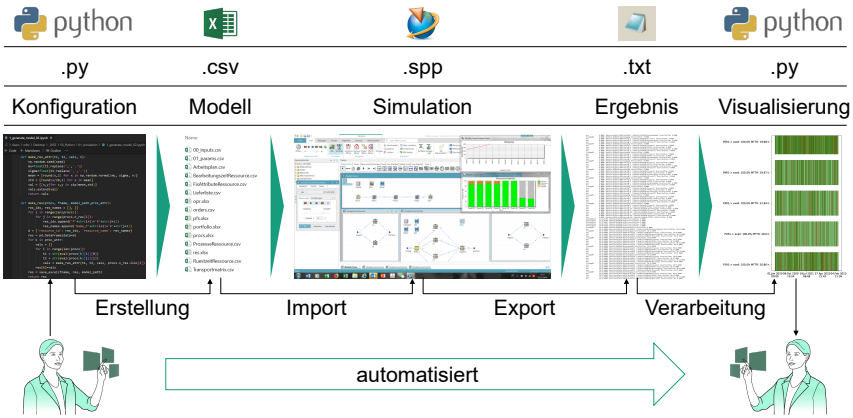


Abbildung 5.14: Gesamtarchitektur zur Simulation

Die aus den Log-Dateien der Simulation extrahierten Zeitreihen in Abbildung 5.15 ähneln erstaunlich genau den systembeschreibenden Zeitreihen in der ‚Enzephalografie der Produktion' von Beer (vgl. Abbildung 1.3). Die beispielhaft geplotteten Zeitreihen hier enthalten unter anderen WIP, Auslastung und Bestände einer für die Evaluation genutzten Produktionssimulation. Da später der Datensatz als Ganzes ohne explizites Systemwissen

verwendet werden soll, werden auch in der Abbildung nicht explizite Kennzahlen, sondern lediglich durchnummerierte Zeitreihen angegeben. Welchen Informationsgehalt die Zeitreihen bieten, ob Korrelationen vorliegen und wie gut damit Prognosen durchführbar sind, ist an dieser Stelle noch nicht zu beurteilen. Die Darstellung dient hier zur Veranschaulichung und Plausibilisierung.

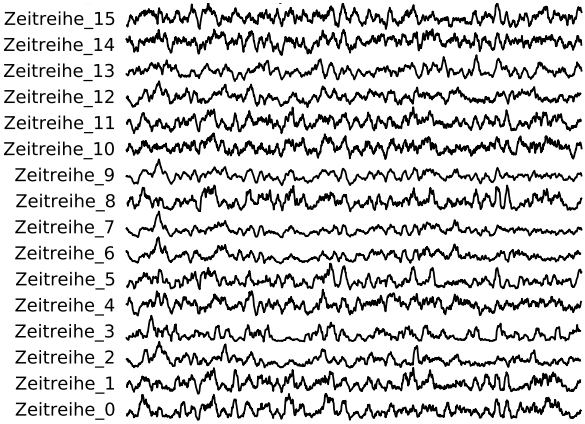


Abbildung 5.15: Zeitreihen aus dem Simulationsmodell

Fazit:

Dieses Kapitel beschreibt den Lösungsansatz zur datenbasierten Turbulenzbewertung. Nach einer allgemeinen Herleitung aus den Forschungsständen, Erkenntnissen und Anforderungen der Vorkapitel (Abschnitt 5.1) erfolgt eine detaillierte Beschreibung von Struktur und Eigenschaften des verwendeten Datensatzes (Abschnitt 5.2). Darauf folgt der Ansatz für das Prognosemodell, welcher Datensätze und Entscheidungsbäume für die konkrete Problemstellung dieser Arbeit verknüpft (Abschnitt 5.3). Dabei geht die Lösung auf vier Punkte ein, die sich aus dem Anwendungsbezug dieser Arbeit ergeben: Nicht-äquidistante Abtastpunkte, Gleichzeitigkeit von Ereignissen, Abweichung zwischen Real- und Fabrikzeit und die Latenzzeit zwischen Ursache und Wirkung. Der Lösungsansatz bietet für diese Punkte Vorgehensanweisungen für die Umsetzung und komplettiert dies mit einem Konzept zur Sample-Auswahl zur Verbesserung der Prognosequalität (Abschnitt 5.5). Abgeschlossen wird das Kapitel mit der Beschreibung des Simulationsmodells, mit dem die Lösung verifiziert

werden soll. Vor der Evaluation wird nun in Kapitel 6 die zweiteilige Umsetzung der Lösung in Python beschrieben. Dabei steht zunächst die Generierung eines synthetischen Datensatzes im Fokus. Auf diesen wird das in der Lösung beschriebene Vorgehen angewandt. Die Implementierung der Turbulenzbewertung an sich stellt den zweiten, abschließenden Teil des Umsetzungskapitels dar.

6 Umsetzung in Python

Die in dieser Arbeit verwendete Programmiersprache ist Python (van Rossum 2021; Lutz 2014; Chollet 2018). Aus dem Stand der Technikwissenschaft erschließt sich, dass sich die Open-Source-Software Python als Sprache für Datenverarbeitung im wissenschaftlichen Kontext bewährt hat. In Abschnitt 5.5 wird zudem Python in Kombination mit PlantSim für das Simulationsmodell bei der Konfiguration und Durchführung der Datengenerierung eingesetzt. Für die Umsetzung der Turbulenzprognose bietet sich die Programmiersprache ebenfalls an: Sie ist plattformunabhängig und besitzt umfangreiche Standardbibliotheken. Aufgrund ihrer freien Verfügbarkeit ist sie weit verbreitet und hat sich vor allem dadurch auch im wissenschaftlichen Bereich etabliert. Neben den für Python klassischen Programmbibliotheken nutzt diese Arbeit für die Aufgaben im Bereich des maschinellen Lernens die Bibliothek scikit-learn (Pedregosa et al. 2011). Sie beinhaltet Werkzeuge für die prädiktive Datenanalyse und zeichnet sich durch ihre detaillierte Dokumentation aus. Zur Generierung synthetischer Daten müssen für die Rohdaten zum Umgang mit ML-Problemen verschiedene Parameter festgelegt werden. Der hierbei verwendete Algorithmus ist angelehnt an den von Guyon (Guyon 2003).

Dieses Kapitel adressiert die beiden zentralen Aspekte der Lösungsumsetzung: Es behandelt erstens die **Datengenerierung** (6.1) des Verifikations- und Validierungsteils für dynamische Systeme und zweitens die Implementierung der eigentlichen **Turbulenzbewertung** (6.2).

6.1 Datengenerierung

Die Datengenerierung ist in seiner Dimensionierung an den später verwendeten Realdatensatz aus der Industrie angelehnt. 1000 Aufträge pro Geschäftsjahr stellen in dem Anwendungsfall die Größenordnung an Samples dar. Es soll untersucht werden, ob Datensätze von Volumen dieser Art ausreichen, um verwertbare Prognosen erstellen zu können. Die Feature-Anzahl liegt nach Abschnitt 5.2 bei etwa 50 Stück und wird im Rahmen der

Untersuchung um die Faktoren zwei bis zehn erhöht, um den Einfluss der Featurezahlen bewerten zu können. Die konkrete Auswahl hängt von der Zweckmäßigkeit für die Darstellung und die Grenzen der Rechenkapazität ab und wird jeweils explizit ausgewiesen.

6.1.1 Features und Targets von Produktionssystemen

Die mit scikit-learn generierten Daten basieren auf normalverteilten Zufallszahlen mit streuenden Mittelwerten μ und Standardabweichungen σ (Abbildung 6.1). Die einzelnen Objekte eines Datensatzes heißen Samples und haben ein oder mehrere beschreibende Attribute. Für jedes Attribut, die sogenannten Features, werden eine festzulegende Anzahl (*n_samples*) an Werten generiert. Die Mittelwerte μ selbst streuen um -1 und 1 bei mit der Feature-Anzahl degressiv ansteigenden Standardabweichungen σ. Dies stellt sicher, dass mit ansteigender Feature-Anzahl diese sich bereits in ihrer Stochastik unterscheiden. Abbildung 6.2 zeigt das stochastische Verhalten für die Feature-Anzahl zwischen 1 und 500. Diese Beschreibung gilt für den mit scikit-learn erstellten Datensatz, bestimmte Parameter können für das Datensatzdesign darauf aufsetzend angepasst werden. Die Anzahl an

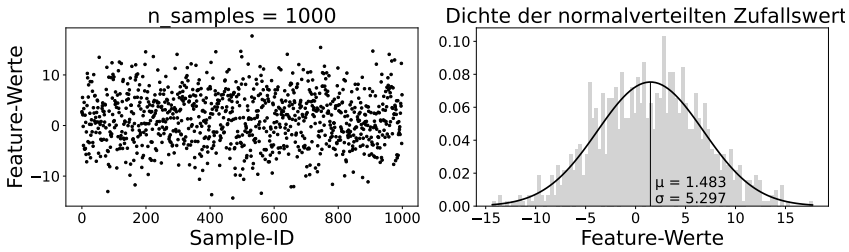


Abbildung 6.1: Zufallswerte für ein Feature

Klassen (*n_classes*) legt fest, welchen und wie vielen unterschiedlichen Klassen der Klassierer die Samples zuordnen soll. Prinzipiell sind beliebig viele Klassen größer Null möglich, wobei die Leistung des Klassierers mit der Anzahl an Klassen abnimmt. Der Standardfall in dieser Arbeit ist die Einteilung in hohe, mittlere und geringe Turbulenz. Damit ist $n\_classes = 3$. Um die Übersichtlichkeit zu gewährleisten, beschränken sich die Darstellungen teilweise auf lediglich zwei Klassen.

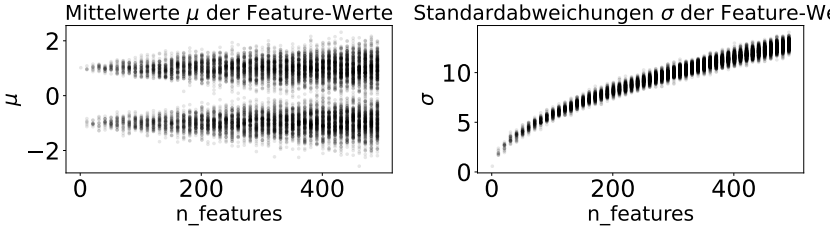


Abbildung 6.2: Stochastisches Verhalten der Feature-Werte

Um realistische Datensätze zu generieren, können die Feature-Werte manipuliert werden. Mit *n_informative* wird die Anzahl an Features mit unterschiedlichem Informationsgehalt festgelegt. *n_redundant* definiert die Anzahl an Features, welche Linearkombinationen der informationstragenden Features sind und *n_repeated* sind stochastisch identische Kopien der vorausgehenden Features. Beispiele für redundante Features wären die Summe der Attribute Breite und Länge als neues Feature und das Attribut Gewicht, einmal in [kg] und einmal in [g] als stochastisch identische Features. Beide verringern den relativen Informationsgehalt des Datensatzes (siehe Abbildung 6.3).

Die zur Gestaltung eines realistischen und dem Anwendungsfall entsprechenden Datensatzes wurden die in Tabelle 6.1 beschriebenen Parameter eingesetzt. Die jeweiligen Darstellungen zu den sechs zentralen Möglichkeiten der Datensatzmanipulation sind im Sinne der Übersichtlichkeit im Anhang zu finden und werden im Folgenden beschrieben. Bei realen Anwendungsfällen sind die einzelnen Klassen meist nicht gleich stark repräsentiert. Der Parameter *weights* gewichtet die Anzahl der den jeweiligen Klassen zugeordneten Samples (Abbildung A.1) und *class_sep* die Distanz der Feature-Werte (Abbildung A.2) je Klasse. Ein Kernproblem bei Realdatensätzen sind die darin enthaltenen Falschinformationen. Solche können mit *flip_y* absichtlich generiert werden, wodurch bei Anteilen der Samples deren Klassenzuordnungen zufällig vertauscht werden (Abbildung A.3). Um nichtnormalverteilte Feature-Werte abzubilden, können mit *n_clusters_per_class* Wertecluster aus mehreren Normalverteilungen je Klasse gebildet werden. Abbildung A.4 zeigt Beispiele für die Werte 1 bis 3. Die beiden letzten Zahlenparameter *shift* und *scale* verschieben den Mittelwert um den übergebenen Wert und Skalieren die Streuung um den angegebenen

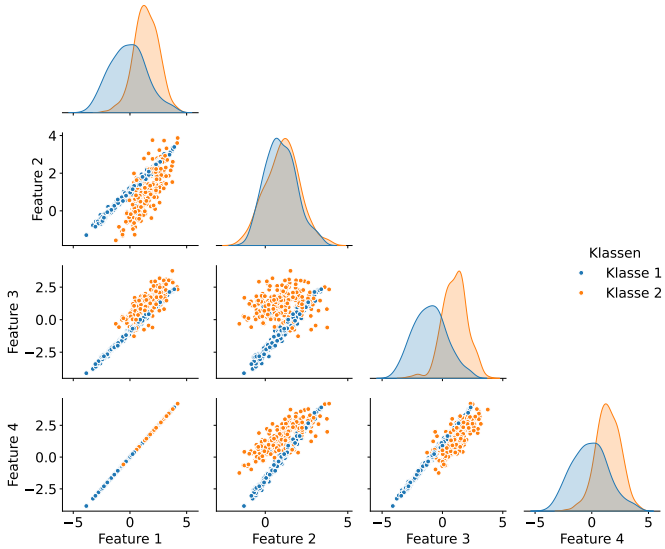


Abbildung 6.3: Redundanz bei Feature 1 und Feature 4

Faktor (Abbildungen A.5 und A.6). *hypercube* legt fest, ob die Cluster der Wertepaare zweier Features auf die Eckpunkte eines Hyperwürfels oder auf die Scheitelpunkte eines zufälligen Polytops gelegt werden. Ersteres hat lediglich Vorteile bei der Visualisierung des Datensatzes.

	f001t01	f002t01	f003t01	f004t01	f005t01	f006t01	f007t01	f008t01	f009t02	f010t02	...	f035t07	t01	t02	t03	t04	t05	t06	t07	ts
0	0.558404	1.526258	-0.895695	1.296337	-0.250709	0.253126	1.240222	0.914248	-2.610142	1.052023	...	1.606421	0.0	1.0	2.0	0.0	1.0	1.0	0.0	1.546298e+09
1	0.500985	1.448989	2.923340	-0.084668	-0.661229	3.616429	-1.474781	2.485278	-2.255381	2.599548	...	2.785195	1.0	1.0	1.0	0.0	1.0	1.0	1.0	1.546347e+09
2	0.491285	0.199638	-0.446527	0.606162	-0.883521	-1.054065	1.043410	-0.686345	-2.115791	-3.864220	...	-2.128467	0.0	0.0	2.0	2.0	1.0	1.0	1.0	1.546405e+09
3	0.430728	-0.877787	-0.423133	0.787897	-2.277547	-1.542084	-2.285775	-0.436649	-0.982409	-0.227833	...	1.170916	2.0	0.0	2.0	2.0	1.0	1.0	1.0	1.546425e+09
4	0.388439	2.726713	1.495146	-0.000108	0.046453	2.661427	0.139489	1.541423	-0.775073	3.373651	...	0.901720	1.0	1.0	0.0	0.0	1.0	1.0	1.0	1.546436e+09
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
996	-3.685787	-0.855272	1.333536	-1.127100	1.024605	0.874110	2.237673	0.097795	4.427945	-3.435796	...	-0.943620	0.0	2.0	1.0	1.0	2.0	2.0	2.0	1.577590e+09
997	-3.823339	-1.255562	2.729664	-5.799603	3.235559	5.187733	1.782523	4.362252	4.610982	4.040116	...	-1.113283	0.0	2.0	0.0	1.0	2.0	2.0	2.0	1.577630e+09
998	-3.897272	-0.071323	-1.783587	-1.155910	-1.362309	0.536535	1.309861	-0.899571	4.651949	1.106776	...	-1.732120	0.0	2.0	1.0	2.0	2.0	2.0	2.0	1.577638e+09
999	-4.541449	2.888821	2.897205	-1.343870	3.175600	2.215298	-0.280075	-2.432772	4.939490	1.265138	...	-0.271147	2.0	2.0	2.0	0.0	2.0	2.0	2.0	1.577749e+09
1000	-5.000784	3.714111	0.496321	-2.266430	1.795224	1.197205	-4.831985	-0.868850	5.240350	-1.690145	...	0.126108	2.0	2.0	1.0	1.0	2.0	2.0	2.0	1.577803e+09

Abbildung 6.4: Visuelle Darstellung des generierten Datensatzes (Screenshot)

Die Abbildung 6.4 zeigt eine vom Programm erstellte Darstellung des Datensatzes mit etwa 1000 Aufträgen, 100 Features und sieben Targets und die Erweiterung um Zeitstempel (Spalte ‚ts'). Der dem Verständnis dienende betreffende Skript-Ausschnitt, in dem der

Tabelle 6.1: Eingabeparameter für die Rohdatengenerierung

Eingabeparameter	Wert	Kommentar
n_samples	1000	Anzahl Aufträge/Jahr
n_features	$[1:500]$	System- und Auftragsattribute entspr. Abschn. 5.2
n_informative	$[1:500-2]$	Anzahl Features mit Informationsgehalt
n_redundant	1	Statistisch gleiche Features
n_repeated	1	Identische Features
n_classes	3	Anzahl an (Turbulenz-)Klassen
n_clusters_per_class	1	Statistische Häufungen innerhalb Klasse
weights	$[0:1]$	Häufigkeit der einzelnen Klassen
flip_y	0.01	Größe für Inkonsistenzen im Datensatz
class_sep	$[0:10]$	Unterscheidbarkeit der Klassen
hypercube	$True$	Dimensionale Verortung der Feature-Werte
shift	$[0, 10, 100]$	Verschiebung der Feature-Werte
scale	$[1, 10, 100]$	Streuung der Feature-Werte
shuffle	$True$	Durchmischung der Samples
random_state	1234	Startwert für pseudo-randomisierte Zufallszahlen

Trainings- und Testdatensatz generiert wird, ist als Pseudocode im Anhang unter A.1 zu finden. Die Datengenerierung basiert auf dem Package scikit-learn und nutzt das Modul ‚make_classification'. Die in Tabelle 6.1 beschriebenen Parameter werden in der Variable ‚kwarg' als Dictionary definiert. Der randomisiert erstellte Datensatz gliedert sich in Features und Targets und wird in einem Python-Dataframe kombiniert. Am Ende wird zufallsbasiert der Datensatz in Trainings- und Testdaten aufgeteilt.

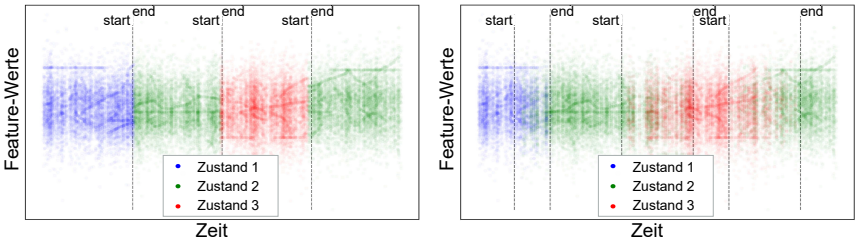
Damit können realistische Datensätze generiert werden, welche sowohl in Feature- als auch in Sample-Anzahl dem Anwendungsfall entsprechen. Zudem unterscheidet sich der Datensatz von den üblicherweise für Entscheidungsbäume konzipierten darin, dass er für ein ganzes Set an Targets gilt, um jede einzelne KPI des Turbulenzsteckbriefes entsprechend Abbildung 1.4 zu enthalten. Der Zeitstempel legt eine zeitliche Reihenfolge der Samples fest. Damit kann im Folgenden abgebildet werden, wie viel ‚Wissen' im Datensatz bereits vorhanden ist. Zudem sind die Zeitstempel wichtig, um Systemübergänge abzubilden, welche in diesen Daten bisher nicht beinhaltet sind. Die Erweiterung des Datensatzes um den Informationsgehalt zu dynamischen Systemen wird im folgenden Abschnitt behandelt.

6.1.2 Repräsentation dynamischer Systeme

Die in der Einleitung beschriebene Problemstellung und die sich daraus ergebende Motivation für diese Arbeit lässt sich in einem Satz zusammenfassen: „Die Systemdynamik verkürzt die Validität des Lerndatensatzes oder löst diese ganz auf." Zu dieser Grundproblematik beim Einsatz von CART-Modellen für dynamische Systeme wurde bereits vom Autor veröffentlicht (Böhm et al. 2021a). Hierbei wurde das Prognoseverhalten bei *einem* Systemwechsel untersucht. Für die Lösung wird nun die Betrachtung von einem auf *beliebig viele* Systemwechsel erweitert. Abbildung 6.5 zeigt die einheitenlosen Feature-Werte des Datensatzes über die Zeit. Die Übergänge zwischen den Systemzuständen können abrupt oder kontinuierlich sein. Für die Untersuchungen im Rahmen dieser Arbeit werden abrupte Systemübergänge gewählt, da so das Verhalten des Prognosewerkzeuges eindeutig diskreten Übergangspunkten zugeordnet werden kann. Gleichzeitig ist die Verallgemeinerung auf kontinuierliche und damit realistische Übergänge ohne weitere Anpassung immer möglich. Beide Fälle sind automatisch mitkonzipiert, da sie in den Daten enthalten sind und vom ausgeführten Code gar nicht erst unterschieden werden müssen.

Um unterschiedliche kontinuierlich oder abrupt ineinander übergehende Systemzustände abzubilden, sind Parametervariationen zwischen den Einzelzuständen notwendig. So wird hier jeder Systemzustand einzeln – analog zum Vorgehen für stationäre Systeme – definiert. Das in dieser Arbeit verwendete Standardmodell beinhaltet einen Datensatz für sieben Targets bei 100 Features. Festzulegen sind Anzahl möglicher Zustände (N_states), Anzahl Zustandswiederholungen (N_backswitch) und relative Zeitpunkte der Zustandsübergänge als übergeordnete Parameter. Diese können abrupte (t_points_d) oder kontinuierliche Übergänge (t_points_c) darstellen. Die folgenden Parameter definieren die Einzelzustände: *Anzahl Features je Zustand* (N_features, hier 100 in Summe), *Rauschen bei den Zeitstempeln* (noise_dt), *Rauschen der Feature-Werte* (noise_dv) und *Latenz zwischen Ursache und Wirkung* (lat_v, hier zwischen 0 und 30 Tagen). Darüber hinaus können die Feature-Zeitreihen je nach Kennzahlen-Charakteristik gestaltet werden. Optionen sind: Mittelwerte und Standardabweichung, Stetigkeit (etwa für Lagerbestände) und Diskretheit (Schichtmodell). Für dynamische Systeme werden drei unterschiedliche Zustände festgelegt, wobei einer wiederkehrt (N_backswitch). Zudem wird unterschieden zwischen abrupten und kontinuierlichen Übergänge und Anzahl der Features für die sieben Targets, Rauschen und Latenz wird

modelliert. Der Pseudocode-Ausschnitt zur Definition des Datensatzes findet sich im Anhang unter A.2.

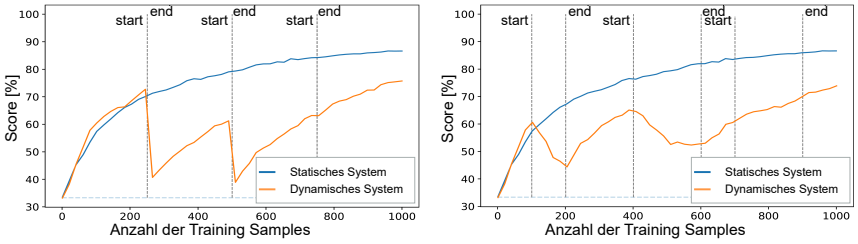


(a) abrupt (b) kontinuierlich

Abbildung 6.5: Datensatz mit dynamischen Systemübergängen

In der einheitenlosen Darstellung der Feature-Werte über die Zeit (Abbildung 6.5) sind die Unterschiede zwischen abruptem und kontinuierlichem Systemübergang erkennbar. Weiter erfolgt nach Zustand 3 ein Übergang zu einem bereits dagewesenen Zustand (grün).

Der Vergleich zwischen der Prognosegüte (Score) bei statischem und dynamischem System (Abbildung 6.6) zeigt das Problem des Validitätsverlustes durch die Systemänderung auf. Folgende Aspekte sprechen für die Plausibilität des erstellten Datensatzes und des Prognosemodells: Der Score liegt bei maximal kleinem Trainingsdatensatz bei etwa 33 %. Da drei Klassen vorliegen, entspricht das genau dem Score beim Würfeln. Mit der Größe des Trainingsdatensatzes steigt der Score im dynamischen System annähernd verhaltensgleich mit dem des statischen Systems an. Die geringen Abweichungen resultieren aus den zufällig erstellten Entscheidungswäldern und sind im Mittel 0. Je nachdem, ob ein vollständiger abrupter oder kontinuierlicher Systemübergang vorliegt, fallen die Score-Werte entsprechend steil ab und erholen sich im Anschluss, bleiben aber immer schlechter als die Scores des statischen Systems. Die Werte der Abbildung 6.6 zeigen, dass die synthetisch generierten Daten plausibel dynamische Systeme abbilden. Die Differenz zwischen den Score-Werten von dynamischem und statischem System veranschaulicht zudem grafisch die Problemstellung. Auf der Grundlage dieser Daten wird nun die Bewertung von Turbulenzen mittels Kennzahlenprognose durchgeführt.



(a) abrupt (b) kontinuierlich

Abbildung 6.6: Plausibilitätstest: Verschlechterung der Prognosegüte bei Systemübergängen

6.2 Turbulenzbewertung

Die Turbulenzbewertung (Prognose des Abweichungsgrades ausgewählter Kennzahlen von ihrem Sollwert entsprechend Abschnitt 3.2) umfasst drei Schritte:

1. Detektion und Bewertung der funktionalen Abhängigkeiten zwischen Features und Targets zur Featurereduktion (Ausschluss irrelevanter Features)
2. Bestimmung der im Kausalsystem enthaltenen Latenzen zwischen den Features und den zugehörigen Targets
3. Auswahl des geeigneten Ausschnitts aus dem Datensatz zur Kompensation der Systemdynamik

6.2.1 Detektion funktionaler Abhängigkeiten

Zur Detektion und Bewertung funktionaler Abhängigkeiten wird in dieser Arbeit zunächst der Korrelationskoeffizient verwendet. Da jedoch nicht immer von linearen Abhängigkeiten ausgegangen werden kann, wird das Vorgehen um eine geometrische Lösung erweitert, bei der nach möglichst dichten – und damit aussagekräftigen – Punktewolken in der zweidimensionalen Darstellung von Feature- und Target-Werten gesucht wird. Zudem wird noch die Möglichkeit genutzt, die Features in ihrem Aussagewert (*Feature-Relevanz*) direkt anhand von Baumklassierern zu testen.

Korrelationskoeffizient

Liegt lineare Abhängigkeit zwischen Feature und Target vor, ist diese mit dem Korrelations-

koeffizienten bewertbar. Für Prognosen ist es unerheblich, ob der Koeffizient positiv oder negativ ist. Erwünscht sind möglichst hohe Absolutwerte. Da nicht zwingend von normalverteilten Wertereihen ausgegangen werden kann, wird in der Lösung der Kendall'sche Konkordanzkoeffizient verwendet, welcher qualitativ dieselbe Aussage zur Abhängigkeit enthält und zudem robuster gegenüber Ausreißern ist. Im Code unterscheidet sich die Berechnung lediglich im übergebenen Parameter-

Zur sicheren Nachvollziehbarkeit wurden die randomisiert generierten Features sortiert, um die Validität des Vorgehens auch optisch bewerten zu können. Für sieben Targets wurden für die nun folgenden Darstellungen je zehn gültige Features generiert. Die jeweils anderen Features (der anderen Targets) beinhalten keine weiteren für eine Prognose nützlichen Informationen. Bei jedem Target wurde eines (immer das erste) der zehn Features als System-Zeitreihe modelliert, welches zeitlich dem Target vorauseilt (Ursache vor Wirkung). Weiter sind auch zwei der sieben Targets (konkret: Nummer vier und fünf) System-Targets und damit ebenfalls als Zeitreihe im Datensatz enthalten.

Abbildung 6.7 zeigt die Korrelationskoeffizienten zwischen Features und Targets *ohne* Korrektur der Latenz. In der Darstellung sind die Feature-Pakete mit hoher Relevanz (helle Punkte) für die einzelnen Targets erkennbar. Zudem sind immer die ersten Features (Feature 1, 11, 21 usw.) tendenziell wenig korrelierend. Dies sind diejenigen, welche als System-Features konzipiert und damit um einen Zeitversatz verschoben wurden. Ohne Korrektur der Latenz wird hier keine hohe Korrelation erkannt. Ausnahmen sind die Korrelationswerte von Target vier und fünf. Da sie als System-Targets zeitlich über alle Features verschoben wurden, weisen sie zwar Ähnlichkeiten untereinander, jedoch keine Muster mit Hinweisen zu funktionalen Zusammenhängen auf. Mit dem Einsatz geeigneter Schwellwerte kann die Unterscheidung zwischen bereits erkannten funktionalen Zusammenhängen und nicht relevantem Informationsgehalt noch eindeutiger dargestellt werden.

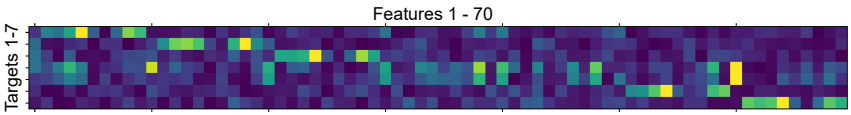


Abbildung 6.7: Korrelationskoeffizient zwischen Features und Targets ohne Latenzkorrektur

Analog zur Abbildung 6.7 mit Bezug auf den Korrelationskoeffizient erfolgen noch die für die geometrische Lösung und die Feature-Relevanz. Sie dienen ebenfalls der Nachvollziehbarkeit des generierten Datensatzes und als Plausibilisierungstest.

Geometrische Lösung

Da Feature-Target-Paare nicht zwingend lineare Zusammenhänge aufweisen, reicht der Korrelationskoeffizient als Indikator für funktionale Abhängigkeit nicht aus. eine praktikable Lösung ist die Bewertung der Punktwolkendichte im normierten zweidimensionalen Raum. Je dichter die Punkte verteilt sind, desto höher ist der potenzielle Informationsgehalt, da potenziell ein Zusammenhang vorliegt. Zur Berechnung der Punktwolkendichte (hier d) wird die verfügbare Darstellungsfläche in n gleichgroße Flächenelemente eingeteilt. Dabei ist n die Anzahl der Wertepaare. Berechnet wird der Anteil der nicht-belegten Flächenelemente an der Gesamtfläche. Ist der Anteil hoch, entspricht dies einer hohen Punktwolkendichte. Abbildung 6.8 zeigt, dass der Korrelationskoeffizient r bei nichtlinearen Abhängigkeiten ohne Aussage über mögliche funktionale Abhängigkeiten bleibt, während die Punktwolkendichte d vergleichbare Werte liefert.

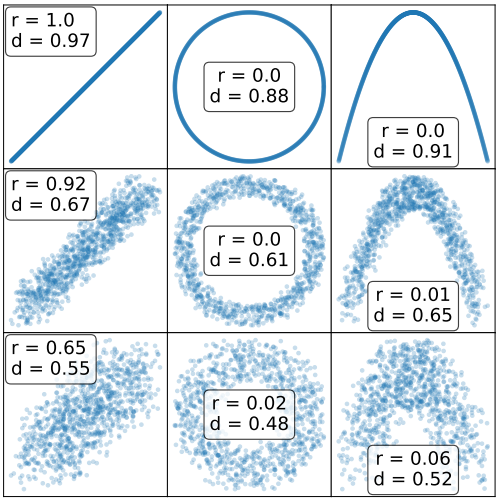


Abbildung 6.8: Korrelationskoeffizient und Punktwolkendichte im Vergleich mit unterschiedlicher Standardabweichung (Zeilen) sowie lineare und nichtlineare Zusammenhänge (Spalten)

Der Vergleich zwischen den Punktwolkendichten verschiedener Feature-Target-Paare ermöglicht weniger die Beurteilung des Grades der funktionalen Abhängigkeit (wie die Korrelation selbst), sondern erlaubt die Unterscheidung zwischen Auftrags- und Systemdaten. So lassen sich in Abbildung 6.9 die sowohl System-Features (1, 11, 21, usw.) als auch -Targets (Nummer 4 und 5) größtenteils eindeutig von den übrigen abgrenzen. Ausnahme sind auch hier die paarweisen Zusammenhänge von jeweils System-Feature und -Target. Praktisch sind die Zusammenhänge im Anwendungsfall dieser Arbeit nicht von Bedeutung, da nie allein von System auf System geschlossen werden soll, sondern immer der Auftragsbezug vorliegt.

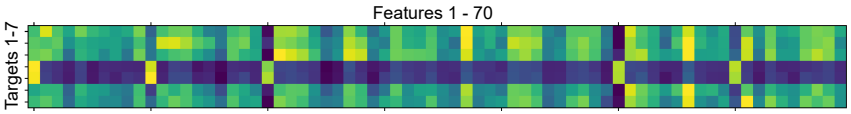


Abbildung 6.9: Punktwolkendichten je Feature-Target-Paar

Feature-Relevanz

Zur Detektion funktionaler Abhängigkeiten kann schließlich auch das Vorgehen zur Entscheidungsbaumerstellung selbst genutzt werden. So werden für jedes Target die Feature-Relevanzen berechnet. Diese ergeben sich aus dem Algorithmus von Breimann et al. mit Hilfe der Berechnung der Node Impurities (siehe Unterabschnitt 2.3.3). In Abbildung 6.10 werden die funktionalen Abhängigkeiten der Order-Features und -Targets noch deutlicher dargestellt als bei den Korrelationskoeffizienten. Die System-Features jedoch (Features 31–50) sind aufgrund der Latenz für die Baumerstellung bislang offensichtlich annähernd wertlos. Zur optimalen Ausnutzung des Informationsgehaltes im vorliegenden Datensatz ist es dementsprechend notwendig, die Latenz zwischen Features und Targets zu bestimmen, um diese Abweichung zu korrigieren.

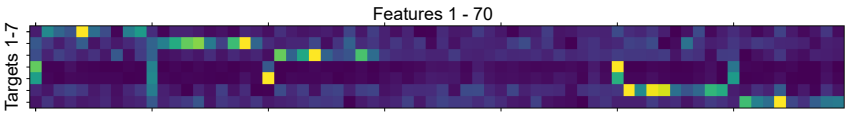


Abbildung 6.10: Feature-Relevanzen je Feature-Target-Paar

6.2.2 Latenz zwischen Features und Targets

Die Bestimmung der Latenz zwischen Features und Targets hat den Zweck, den Informationsgehalt des Trainingsdatensatzes besser auszunutzen. Ohne Berücksichtigung des zeitlichen Versatzes zwischen Ursache (Feature) und Wirkung (Target) werden mögliche kausale Zusammenhänge in den Entscheidungsbäumen unscharf oder falsch repräsentiert. Für die Bestimmung der Latenz wird genau der Zeitversatz gesucht, bei dem der Zusammenhang zwischen Feature und Target am *deutlichsten* wird. Dazu können prinzipiell die verschiedenen Vorgehen aus dem vorangehenden Unterabschnitt verwendet werden. Die Maxima der Kreuzkorrelationsfunktionen in Abhängigkeit der Latenz weisen für lineare Zusammenhänge auf den Wert des Zeitversatzes hin. Für nichtlineare Zusammenhänge bietet sich die Abtastung mit der geometrischen Lösung sowie die Feature-Relevanz mit Latenzreihen an.

Der generierte Datensatz (Abbildung 6.4) enthält zusätzlich zu Feature- und Target-Werten eine Wertespalte ‚ts'. Darin enthalten sind die dem Event zugeordneten Unix Timestamps. Diese werden Zeitstempel genannt. Sie stellen also keine Realzeiten im Sinne von Abschnitt 3.1 dar und beinhalten bereits Unschärfen durch den unterschiedlichen Zeitverzug zwischen Ereignis und Eventspeicherung. Sie dienen dennoch als einheitlicher zeitlicher Bezugspunkt, um die Latenzen zu bestimmen. Aus der Systemtheorie hergeleitet und logisch erfassbar eilen System-Features und -Targets denen der Aufträge immer voraus (Abbildung 6.11). Die Zeitstempel dienen hier ausschließlich der korrekten Zuordnung von Features zu Targets und der Bestimmung der Latenz. Für die spätere Generierung der Entscheidungsbäume und die eigentliche Prognose ist der Zeitstempel nicht mehr relevant, da bei der Erstellung von Entscheidungsbäumen die dann vollständigen, einzelnen Samples isoliert und nicht in ihrer zeitlichen Abfolge berücksichtigt werden.

Mit Hilfe der Kreuzkorrelationsfunktion (Gleichung A.19) können vergleichsweise einfach Zeitverschiebungen zwischen zwei Zeitreihen analytisch berechnet werden. Bei Realdatensätzen liegen jedoch zwei wesentliche Voraussetzungen meist nicht vor: stetige Signale und äquidistante Zeitstempel. Zudem enthalten reale Daten Lücken, Ausreißer und schlicht falsche Werte. Daher sind die mit Kreuzkorrelationsfunktionen gefunden Latenzen zwar durchaus valide Informationen, jedoch immer im Kontext zu bewerten. So beschränkt sich die Umsetzung hier darauf, nur plausible Latenzen zwischen Features und Targets zu

	f001t01	f002t01	f003t01	f004t01	f005t01	f006t01	f007t01	f008t01	f009t02	f010t02	...	f035t07	t01	t02	t03	t04	t05	t06	t07	ts
0	0.558404	1.526258	-0.895695	1.296337	-0.250709	0.253126	1.240222	0.914248	-2.610142	1.052023	...	1.606421	0.0	1.0	2.0	0.0	1.0	1.0	0.0	1.546298e+09
1	0.500985	1.448989	2.923340	-0.084668	-0.661229	3.616429	-1.474781	2.485278	-2.255381	2.599548	...	2.785195	1.0	1.0	1.0	0.0	1.0	1.0	1.0	1.546347e+09
2	0.491285	0.199638	-0.446527	0.606162	-0.883521	-1.054065	1.043410	-0.686345	-2.115791	-3.864220	...	-2.126467	0.0	0.0	2.0	2.0	1.0	1.0	1.0	1.546405e+09
3	0.430728	-0.877707	-0.423133	0.787897	-2.277547	-1.542084	-2.285775	-0.436649	-0.982409	-0.227833	...	-4.170916	2.0	2.0	2.0	2.0	1.0	1.0	1.0	1.546425e+09
4	0.388439	-2.126716	1.499146	-0.000108	0.046453	2.661427	0.139489	1.541423	-0.775073	3.373651	...	0.901720	1.0	1.0	0.0	0.0	1.0	1.0	1.0	1.546436e+09
...																				
996	-3.685787	-0.855272	1.333536	-1.127100	1.024605	0.874110	2.237673	0.097795	4.427945	-3.435798	...	-0.943620	0.0	2.0	1.0	1.0	2.0	2.0	2.0	1.577590e+09
997	-3.623339	-1.255562	2.729664	-5.799603	3.235559	5.187733	1.782523	4.382252	4.610982	4.040115	...	-1.113283	0.0	2.0	0.0	1.0	2.0	2.0	2.0	1.577630e+09
998	-3.897272	-0.071323	-1.783587	-1.155910	1.562308	0.536535	1.309861	-0.899571	4.651949	1.106778	...	-1.732120	0.0	2.0	1.0	2.0	2.0	2.0	2.0	1.577638e+09
999	-3.541443	2.888821	2.897205	-1.343870	3.175600	2.215298	-0.280075	-2.432772	4.939490	1.265139	...	-0.271147	2.0	2.0	2.0	0.0	2.0	2.0	2.0	1.577749e+09
1000	-5.009784	3.714111	0.496321	-2.266430	1.795224	1.197205	-4.831985	-0.868850	5.240350	-1.690145	...	0.126108	2.0	2.0	1.0	1.0	2.0	2.0	2.0	1.577803e+09

Abbildung 6.11: Datensatz mit Timestamps und Latenzverschiebung der Auftragsdaten (grün, nach ‚früher') und System-KPI's (blau, nach ‚später')

suchen, die in Zeitreihen des Produktionssystems enthalten sind. Diese Vereinfachung ist deshalb gültig, weil Auftragsattribute die Auftrags-KPI's anderer Aufträge (= zu späteren Zeitpunkten) nicht direkt, sondern nur über das Produktionssystem beeinflussen. Es sind in den synthetisch erstellten Datensätzen immer mehrere Targets und mindestens ein Feature je Target als Zeitreihe des Produktionssystems enthalten, um die Funktionalität der Latenzdetektion nachzuweisen. Darüber hinaus verringert diese Vorauswahl den Rechenaufwand. Kern der Latenzberechnung für Realdaten ist die Funktion *calc_delay*, welche, – auf die wesentlichen Schritte reduziert – im Anhang A.3 als Pseudocode beschrieben wird.:

Für jedes Zeitreihenpaar werden vier Korrelationsfunktionen (Faltung) berechnet, da umgekehrt proportionale Relationen sonst falsch gewichtet würden. Dazu werden die normierten Zeitreihen entsprechend aller Permutationen von 0 und 1 ‚umgeklappt' und gefaltet. Gesucht wird der Maximalwert mit der deutlichsten Ausprägung. Negative Latenzen können aus Plausibilitätsgründen ausgeschlossen werden, da die Wirkung immer der Ursache zeitlich nachfolgt. Abbildung 6.12 zeigt beispielhaft die vier Faltungsfunktionen über den errechneten Latenzwerten in Tagen. Die tatsächliche im Datensatz implementierte Latenz liegt in diesen Zeitreihen bei 20 Tagen, die durch Berechnung ermittelte bei 2,93 Tagen. Somit ist der Fehler bezogen auf die absolute Latenz 4,7 % und bezogen auf den betrachteten Zeitraum (1 Jahr) bei 2,5 ‰.

Die ergänzenden Latenzbestimmungen mittels Punktwolkendichte und Feature-Relevanz erfolgen ebenfalls über Abtastung und nutzen dieselben Code-Funktionen wie bei der Bestimmung des funktionalen Zusammenhangs und werden daher nicht erneut beschrieben.

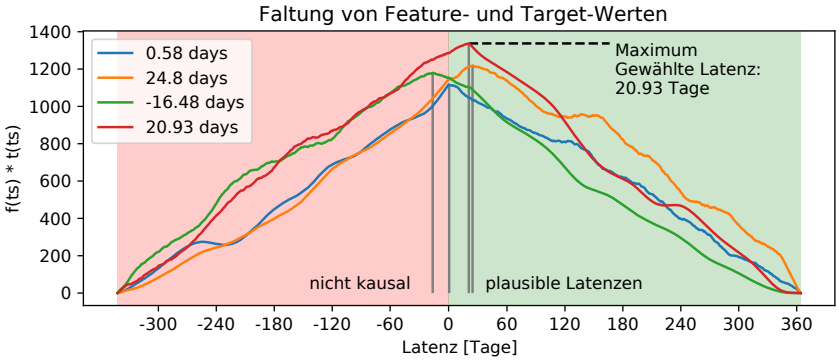


Abbildung 6.12: Faltung zur Bestimmung der Latenz

Die Bewertung ihrer Wirkfähigkeit erfolgt im Rahmen der Evaluation in Abschnitt 7.3. Nach Abschluss der Latenzbestimmung für jedes Feature-Target-Paar werden die Zeitreihen im Gesamtdatensatz auf Grundlage der gefundenen Latenzen, wie in Abbildung 6.11 angedeutet, entsprechend angepasst.

6.2.3 Sample-Auswahl

Um der Systemdynamik und der daraus folgenden Gültigkeitsabnahme des Trainingsdatensatzes Rechnung zu tragen, muss entsprechend dem in 5.4 beschriebenen Vorgehen der noch gültige Ausschnitt aus dem verfügbaren Datensatz gefunden und ausgewählt werden. Dazu wird mit anwachsendem Trainingsdatensatz die Prognose für den jeweils nächsten Auftrag durchgeführt. Mit der Größe des Datensatzes verbessert sich die Prognosegüte ausgehend von etwa 33 % im Falle von drei Klassen auf einen Maximalwert, welcher die bis dahin im Trainingsdatensatz enthaltenen Informationen optimal nutzt. Erfolgen dabei jedoch Änderungen im dynamischen System, *sinkt die Prognosegüte*. Abhängig von einem definierten Schwellwert wird dann der älteste Teil des Datensatzes nicht mehr für das Modelltraining mitverwendet, sondern um eben diesen reduziert.

Aus dem nicht länger berücksichtigten Datensatz wird durch kombinatorische Sample-Auswahl (siehe Abbildung 5.12) versucht, durch erneute Einbeziehung eine *Verbesserung der Prognose* zu erzielen. Ist dies der Fall, werden die ausgewählten ‚alten' Daten zusam-

men mit den aktuellen berücksichtigt und um diesen Teil *erweitert*. Um die Rechenzeit klein zu halten, werden alte Datenabschnitte in maximal sechs Bereiche eingeteilt, die in allen kombinatorischen Möglichkeiten abgetastet werden. Dazu werden entsprechende Abschnitte des Trainingsdatensatzes berücksichtigt (‚1') oder eben nicht (‚0'). Abbildung 6.13 fasst den simplen Algorithmus zur Sample-Auswahl zusammen.

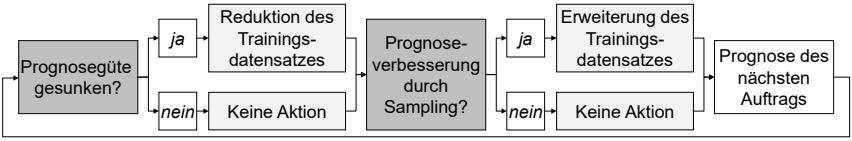


Abbildung 6.13: Algorithmus zur Sample-Auswahl

Die relevanten Code-Ausschnitte sind die des Regelkreises zur schwellwertabhängigen Reduktion des Trainingsdatensatzes (siehe Anhang A.4) und die der Tupel-Bildung für die Auswahl der Samples (siehe Anhang A.5).

Geeignete Werte für den Schwellwert sind nicht immer gleich, liegen jedoch im Bereich >90 %. Die Anzahl an abzutastenden Steps und Permutationen kann beliebig groß sein. Hierbei sollte auf eine sinnvolle Abwägung zwischen Genauigkeitsgewinn und Rechenzeitverluste geachtet werden. Bei jedem Schleifendurchlauf des Algorithmus wird ein neuer Baumklassierer erstellt. Die Parameter zur Generierung der Klassierer für den beschriebenen Anwendungsfall sind in Tabelle A.3 im Anhang dokumentiert.

Fazit:

Das Kapitel zur Umsetzung des Werkzeugs zur Turbulenzprognose befasst sich zunächst mit der Datengenerierung, welche die Grundlage für die Entwicklung und Evaluation des Prognosetools darstellt. Zentrale Punkte sind dabei der richtige Einsatz der Python-Bibliotheken für die Generierung von Trainingsdaten und die Manipulation, um dynamische Systeme zu repräsentieren. Der zweite Teil behandelt die Turbulenzprognose für dynamische Systeme an sich und erklärt das Vorgehen dieser Arbeit zur Bestimmung der funktionalen Zusammenhänge, Latenzzeiten zwischen Features und Targets und der gezielten Sample-Auswahl für den Trainingsdatensatz. Dazu werden jeweils die theoretischen Grundlagen auf die Anwendung übertragen und die konkreten Ausprägungen des Python-Codes anhand von Auszügen aus den Skripten als Pseudocode dargestellt. Das Kapitel 6 beschreibt damit

das Vorgehen, um auch bei vorhandener Systemdynamik anhand optimierter Trainingsdatensätze Zielkennzahlen mit verbesserter Güte zu prognostizieren. So ist es möglich, für jedes Target eine Auswahl an Features unter Korrektur der Latenz und der gezielten Auswahl von gültigen Samples Ensembles für die Vorhersage zu bilden. Abbildung 6.14 zeigt exemplarisch *einen* Entscheidungsbaum für *eine Kennzahl* aus dem Ensemble von Bäumen nach Featuredetektion, Latenzkorrektur und Sample-Auswahl.

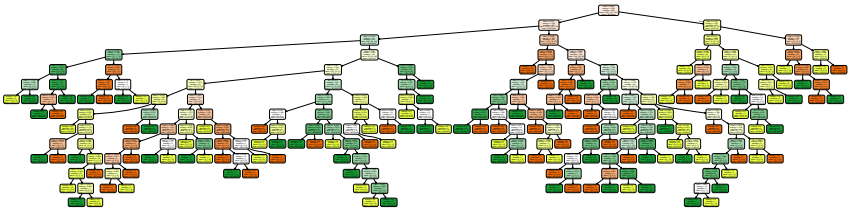


Abbildung 6.14: Entscheidungsbaum mit drei Klassen

Jedem Auftrag kann nun vor seinem Produktionsstart für jede KPI eine Turbulenzklasse mit entsprechender Eintrittswahrscheinlichkeit angegeben werden. Hinter jeder einzelnen KPI-Prognose steht ein Entscheidungswald, dessen Trainingsdatensatz unter der Berücksichtigung von Systemdynamik laufend angepasst und damit neu erstellt wird. Jede KPI hat ihr eigenes Set an Features, eigene Korrekturwerte der jeweiligen Latenzzeiten und eine individuelle Selektion der Samples, je nach ihrer Gültigkeit. Der Trainingsdatensatz wird mit jedem weiteren Sample potenziell angepasst. Abbildung 6.15 zeigt eine Auswahl von sechs exemplarischen Turbulenzsteckbriefen von aufeinanderfolgenden Aufträgen, welche bei Ausnutzung von etwa der Hälfte der Trainings-Samples erstellt wurden. Ein Steckbrief enthält für jede KPI die Turbulenz-Klassierung codiert in der jeweiligen Farbe und ihren Probability-Wert, dargestellt sowohl als Balkenlänge als auch mit Wertangabe in Prozent. Der Probability-Wert gibt an, wie sicher sich der Klassierer bei seiner Klassenzuordnung ist (siehe Abschnitt 5.4). Bei drei Klassen sind 33,3 % der im Mittel denkbar schlechteste, 100 % der maximale Wert. Die zentrale Linie stellt keinen Nullpunkt im mathematischen Sinne dar, sondern ist lediglich Ursprung der verhältnismäßigen Ausdehnung der positiven Werte für prognostizierte Kennzahlen im Toleranzbereich und negative Werte für die mit erwarteter Turbulenz. Die als mittel turbulent prognostizierten Werte sind achsensymmetrisch um diese zentrale Achse gelb dargestellt.

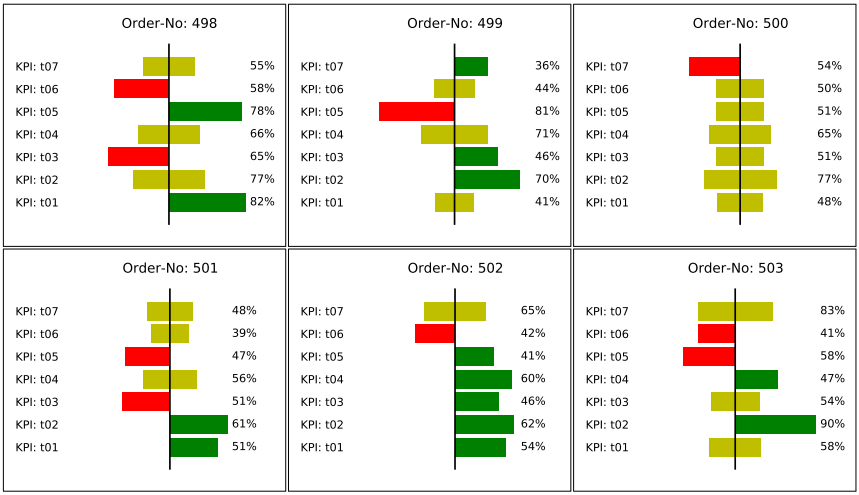


Abbildung 6.15: Turbulenzsteckbriefe für sechs ausgewählte Aufträge (synthetische Daten)

7 Verifikation und Validierung

Es folgt die Evaluation der Ergebnisse. So sollen die in Abschnitt 1.6 aufgestellten Hypothesen entweder bestätigt oder falsifiziert werden. Dazu wird versucht, sowohl die Funktionserfüllung als auch die Relevanz in der Anwendung nachzuweisen. Die Evaluation gliedert sich damit in die Teile **Verifikation**, in welcher die prinzipielle Funktionalität des Modells überprüft wird und die **Validierung**, durch die – soweit zum gegenwärtigen Entwicklungsstand möglich – ein Vergleich mit den aktuellen Begebenheiten bei einem Anwendungsfall erfolgen soll. Beim Vorgehen werden sowohl für die Verifikation als auch für die Validierung die Anforderungen an die empirische Forschung genutzt, um Aussagen zur Validität der Ergebnisse zu machen. Gesammelt dargestellt wurden diese von (Töpfer 2010, 231 ff.) – zurückgehend auf verschiedene weitere Autoren (Hermann et al. 2008, 10 ff. Diekmann 2018, 247 ff. Himme 2007, 485 ff.). Es wird Bezug genommen auf die vier Anforderungen Objektivität, Validität, Reliabilität und Generalisierbarkeit, welche die Erfolgskriterien für die Evaluation darstellen. Letztendlich beantwortet dieses Kapitel die Frage, ob und wie sehr das Arbeitsergebnis zum Fortschritt bei Prognosen in der Produktion beitragen kann. Das Kapitel schließt mit einer sowohl qualitativen als auch quantitativen und fallabhängig übertragbaren **Abschätzung von Aufwand und Nutzen**.

7.1 Darstellung des Vorgehens im Überblick

Betrachtet werden *drei unterschiedliche Systeme*, die als Übersicht in Tabelle 7.1 dargestellt sind. Zunächst ein statisches Produktionssystem (Abschnitt 7.2), welches digital modelliert worden (Abschnitt 5.5) ist. Statisch bedeutet, dass – abweichend von der Realität – die Konfiguration des Systems konstant bleibt. Variiert wird somit nur die Systemlast über die Auftragsanzahl pro Zeitintervall. Ziel ist der grundsätzliche Funktionsnachweis sowie die Prüfung der Plausibilität des Gesamtvorgehens. Dies adressiert somit die **Forschungsfrage 1** zum grundsätzlich benötigten Datenvolumen. Am zweiten System wird der dynamische Fall betrachtet (Abschnitt 7.3). Hier werden synthetische Daten verwendet, die im Gegensatz

Tabelle 7.1: Vorgehen für die Evaluation

	Datensorte	**Datenherkunft**	**Evaluations-zweck**
Statisches System (Kap. 7.2)	synthetisch	Simulation	Verifikation
Dynamisches System (Kap. 7.3)	synthetisch	randomisiert generiert	Verifikation und Validierung
Reales System (Kap. 7.4)	real	ERP/MES	Validierung

zum statischen Fall jedoch nicht mittels Simulation sondern randomisiert generiert wurden, um gezielt und in großer Anzahl Systemänderungen in den Daten darzustellen und experimentell abzutasten. Ziel ist die Überprüfung, ob und wie gut das Arbeitsergebnis auf sich verändernde Systeme reagieren kann (**Forschungsfrage 2**). Als letztes wird das Vorgehen auf einen Realdatensatz angewendet (Abschnitt 7.4), um Aufschlüsse zur Tauglichkeit und Hinweise zu Mängeln im Einsatz zu erhalten. In diesem Teil liegt der Fokus auf der Verbesserung des Verfahrens (siehe Abschnitt 1.4) für die Anwendung. Es werden dabei die Aussagen zum Gültigkeitsbereich des Vorgehens generiert (**Forschungsfrage 3**). In Abschnitt 7.5 wird abschließend die **Forschungsfrage 4** zum Aufwand und Mehrwert beantwortet.

7.2 Verifikation im statischen System mit Simulationsdaten

Um Aussagen zur prinzipiellen Anwendbarkeit des Vorgehens machen zu können, ist die Nutzung möglichst realistischer Produktionsdaten einer komplexen Produktion entsprechend Abschnitt 2.1 anzustreben. Gleichzeitig sollte der informatorische Inhalt des Datensatzes bekannt sein, um Plausibilitätsprüfungen zu ermöglichen und entdeckte Kausalitäten nachvollziehen zu können. Die Entscheidung zur Verifikation des Vorgehens fiel daher auf durch Simulation generierte Produktionsdaten eines statischen Systems. Das Simulationsmodell erfüllt hier mehrere Zwecke: Zunächst ist es deterministisch und ermöglicht durch seine Transparenz eine weitgehende Durchführungs-, Auswertungs- und Interpretationsobjektivität (Objektivität). Weiter beschränkt sich das Modell auf die zu erhe-

benden Größen und erfüllt damit die Anforderung an die Validität und somit den Anspruch, „[...] dass das gemessen wird, was gemessen werden soll" (Töpfer 2010, S. 231). Die Reliabilität, also die Anforderung, dass wiederholte Messungen zum gleichen Messergebnis führen, ist ebenfalls durch die determinierten Eigenschaften eines Modells gegeben. Die Generalisierbarkeit der Erkenntnisse eines Modells ist niemals absolut. Töpfer fordert, dass ihr Ausmaß „möglichst groß" sein sollte. In Zusammenhang mit Modellen hilft der Begriff der Repräsentativität. Um diese in möglichst hohem Maße zu erfüllen, wurde ein parametrierbares Produktionsmodell erstellt, welches (nur begrenzt durch die Möglichkeiten des Programms und der Rechenkapazität) *beliebig* komplexe Systeme darstellen kann (siehe Abschnitt 5.5). Parametriert werden Intervalle, Mittelwerte und Streuung der in Tabelle A.2 angegebenen Merkmale des Produktionsmodells. Die genauen Parameterwerte für die jeweilige Verifikationsbetrachtung sind im Anhang (Kap. A) angegeben.

Im statischen Fall bleibt die Konfiguration des Produktionssystems konstant. Anzahl von Prozessen und Ressourcen, sowie deren Bearbeitungs-, Rüst- und Transportzeiten unterliegen keiner zeitlichen Änderung. Weiter bleibt das Produktportfolio über das gesamte Zeitintervall, für das die Produktionsdaten generiert werden, unverändert. Da bei den Beobachtungen für das statische System die Transparenz und Nachvollziehbarkeit im Fokus stehen, werden auch Latenzzeiten zwischen Ursache und Wirkung – wie beispielsweise zwischen WIP und DLZ – nicht berücksichtigt. Die Beobachtungen beschränken sich auf die Zielkennzahl DLZ, da die Übertragbarkeit auf beliebig viele Zielkennzahlen in Kapitel 5 dargelegt und in Abschnitt 7.3 ausführlich behandelt wird. Ausschließlich der Kundenbedarf, welcher die Systemlast definiert, variiert hier. Der Entscheidungswald besteht aus 100 Schätzern mit den Standardeinstellungen aus Tabelle A.3. Um die Reproduzierbarkeit zu gewährleisten, wurde der Eingabeparameter *random_state* nicht variiert. In diesem ersten Schritt sollten zwei Dinge erzielt werden: Getroffene Aussagen über das Verhalten des entwickelten Vorgehens beim Einsatz mit realitätsnahen Produktionsdaten und der prinzipielle Nachweis der Machbarkeit von Vorhersagen über Zielgrößen der Produktion mit solchen Datensätzen.

Dazu werden drei Simulationsläufe, die sich nur in der Auftragslast pro Zeiteinheit unterscheiden, durchgeführt und ausgewertet. Je nach Lastfall sollten die Features für die Prognose unterschiedlich relevant sein. Beispielsweise sind für die Durchlaufzeit bei geringer

Auslastung die Bearbeitungszeiten relevant, bei hoher Auslastung eher die Liegezeiten. Zunächst wird das System mit geringer Last beaufschlagt. Die wenigen Kundenaufträge erhöhen bei Überkapazitäten die Bestände nicht signifikant. Folglich liegen die DLZ nahe an den Summen der Prozesszeiten. Es ist also zu erwarten, dass die DLZ im Wesentlichen von der jeweiligen Produktfamilie mit den dazugehörigen Prozesszeiten sowie den Auftragsgrößen beeinflusst wird. Der Einfluss des Systemzustands (Auslastung von Puffer und Ressourcen) sollte den der Auftragseigenschaften nicht um das Vielfache übersteigen.

In Abbildung 7.1 kann die Relevanz der einzelnen Features bei Unterlast verglichen werden. Dargestellt sind die MDI (siehe Unterabschnitt 2.3.3) für jedes Feature, was als anteiliger Informationsgehalt der einzelnen Features gedeutet werden kann. Gemeinsam ergeben sie die Summe 1. Die Features sind entsprechend Abschnitt 5.2 in auftragsbeschreibend und systembeschreibend eingeteilt. Auf der Seite der auftragsbeschreibenden Features zeigen sich für den Fall der Unterlast erwartungsgemäß die höchsten Feature-Relevanzen bei den Auftragseigenschaften: Größten Einfluss auf die DLZ hat die Auftragsgröße in Stück, gefolgt von der Zuordnung zur einer Produktfamilie mit Produkt_ID und der Anzahl an Produktionsprozessen. Welche Prozesse genau involviert sind, spielt bei geringer Auslastung eine untergeordnete Rolle. Plausibel sind zudem die Werte für die Prozesse 4, 5 und 7. Die Features *Prozess_4*, *Prozess_5* und *Prozess_7* haben keinen Einfluss, da diese Prozesse von allen Produktfamilien durchlaufen werden und somit die Information, dass sie Teil der Prozesskette sind, redundant ist. Bei Überlast ergibt sich die Verteilung der MDI

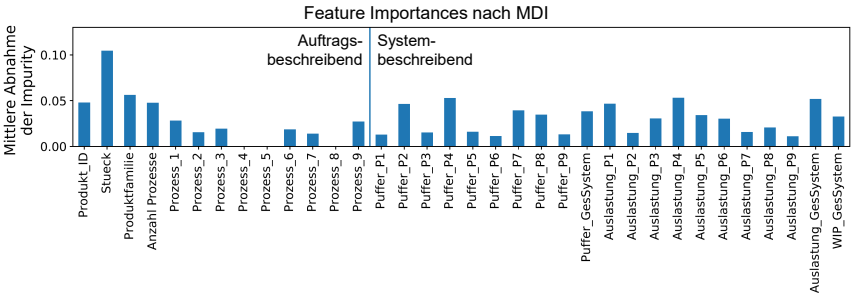


Abbildung 7.1: Relevanz der Features bei Unterlast

nach Abbildung 7.2. Die Relevanz der auftragsbeschreibenden Features nimmt ab. Einfluss auf die DLZ hat im wesentlichen das WIP des Gesamtsystems. Die in den Zeitreihen zu

Puffer- und Ressourcenauslastung enthaltenen Informationen sind in Teilen redundant zum Gesamt-WIP, enthalten jedoch in bestimmten Fällen zusätzliche Informationen. Prozess 3 ist im Produktionsmodell beim Lastfall der am häufigsten auftretende temporäre Engpass. Die Größe der DLZ hängt daher in hohem Maße von diesem Prozess ab, weshalb dessen Auslastungen den stärksten Einfluss auf die DLZ haben.

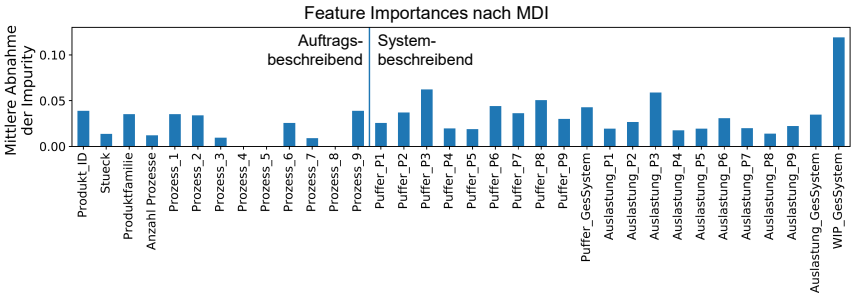


Abbildung 7.2: Relevanz der Features bei Überlast

Zur Vollständigkeit stellt Abbildung 7.3 einen Lastfall im Übergangsbereich zwischen Unter- und Überlast dar. Die Feature-Relevanz verteilt sich vergleichsweise homogen auf die Features. Einzig die Stückzahl je Auftrag hat einen leicht erhöhten Einfluss. Das WIP des Gesamtsystems hat noch keinen ausgeprägten Einfluss. Die Auftragseigenschaften haben die Aussagekraft bereits weitgehend verloren haben.

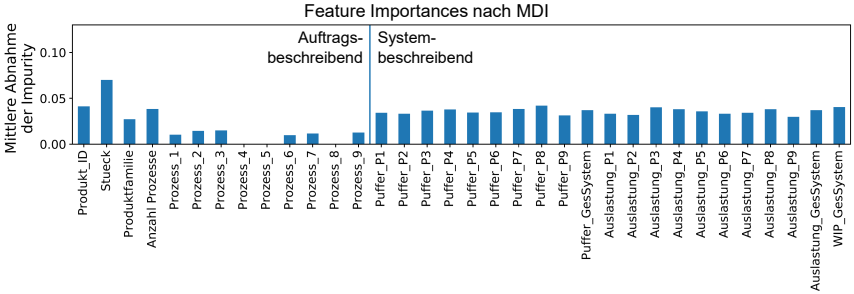


Abbildung 7.3: Relevanz der Features am Übergang zur Vollauslastung

Abbildung 7.4 zeigt den Anteil richtiger Prognosen nach n Samples. Je stärker einzelne Relevanzen der Features ausgeprägt sind, desto höher ist die Prognosegüte. So ist die

Zielgröße im Überlastfall relativ eindeutig vom WIP abhängig, wodurch gute Prognosen möglich werden. Im Unterlastfall und Übergangsbereich verteilen sich die Einflussrelevanzen stärker und die Prognosegüte sinkt. Das Verhalten der Prognosegüte bei geringer Sample-Anzahl (hier < 100) ist in Teilen zufallsbehaftet. Grund dafür sind ‚Glückstreffer' bei den ersten Prognosen. Diese beeinflussen den Anteil richtiger Prognosen in erhöhtem Maße. Im Unterlastfall konvergiert die Kurve am schnellsten. Die wenigen auftragsbeschreibenden Features sind schnell im Entscheidungswald repräsentiert und neue Informationen kommen kaum mehr hinzu. Beim Überlastfall treten anfangs zufällig gehäuft richtige Prognosen auf, die im weiteren Verlauf statistisch korrigiert werden. Die Kurve zeigt konvergierendes Verhalten, gefolgt von einer leichten Abnahme. Das rührt daher, dass sich bei Überlast kein Lastfall wirklich wiederholt, da das WIP immer weiter ansteigt. Dies führt zu negativen Effekten, ausgelöst durch sich ständig verändernde Eingangsgrößen und punktuellem Over-Fitting (siehe Unterabschnitt 2.3.3). Die Prognosegüte im Grenzfall ist am niedrigsten. Hier bildet sich die fehlende Eindeutigkeit bei den relevanten Features ab. Die Prognosegüte pendelt sich auf vergleichbar niedrigem Wert ein. Die Bereiche von Unter- und Überlast werden nie erreicht.

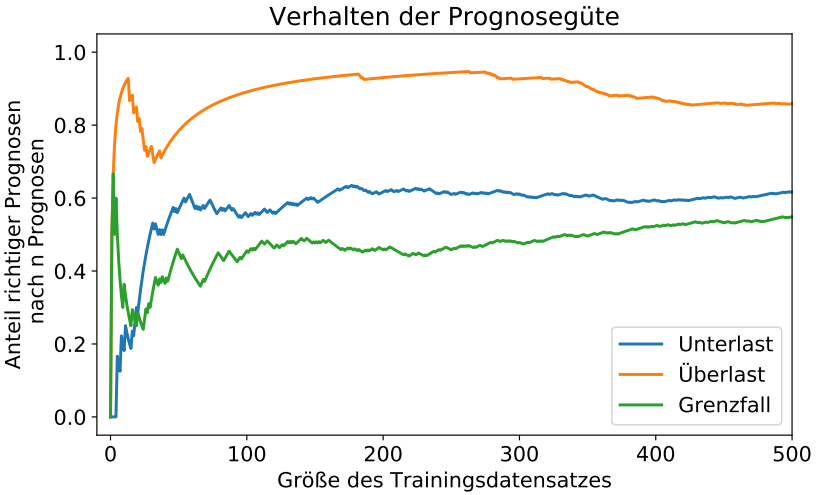


Abbildung 7.4: Vergleich der Prognosegüten je Auslastungsgrad

Es konnte anhand generisch erzeugter Simulationsdaten gezeigt werden, dass Prognosen über Produktionssysteme mittels Auftrags- und Systemkennzahlen prinzipiell möglich sind. Der Funktionsnachweis erfolgte über ein generisches Simulationsmodell mit beliebiger Systemkonfiguration, Produktportfolio und Systemlast. Die randomisierte Modellerzeugung reduziert das Risiko, im Forschungsprozess das gewünschte Ergebnis durch bewusste oder unbewusste Einflussnahme zu erzielen. Dies erhöht die wissenschaftliche Distanz und den Grad der Objektivität. Unter diesen Voraussetzungen wurden verschiedene Aussagen gemacht, welche für die prinzipielle Funktionalität im beschriebenen Anwendungskontext sprechen. Die Durchführung und Bewertung erfolgte unabhängig von Feature-Anzahl und Target-Gewichtung und weist daher auf eine weitgehende Generalisierbarkeit der Aussagen hin. Die zu Beginn des Kapitels 7 genannten Anforderungen sind damit in Teilen bereits erfüllt.

Tabelle 7.2: Wald-Performance für statischen Fall

	Unterlast	**Überlast**	**Grenzfall**
Score	70.2 %	98.1 %	58.7 %
Probability	67.9 %	94.9 %	65.6 %

Die Graphen aus Abbildung 7.4 spiegeln nicht die tatsächliche Leistungsfähigkeit der Klassierung wider, deuten jedoch bereits an, bei welchen Lastfällen höhere Performance-Werte zu erwarten sind. Für Entscheidungswälder mit 100 Schätzern und einem Volumenverhältnis von Trainings- zu Lerndatensatz von 1/1 ergeben sich die Performance-Werte für Score und Probability aus Tabelle 7.2. Im Überlastfall, der praktisch nur eine Systemabhängigkeit vom WIP darstellt, sind Modellbildung und Prognose einfach umsetzbar. Die annähernd lineare Abhängigkeit vom WIP führt zu beinahe sicheren Prognosen mit einem Score von >98 %. Der in der praktischen Anwendung kritische Grenzfall weist Score-Werte unter 60 % auf und liegt damit dennoch robust über den 33,3 % des Würfelns. Die niedrigen Werte rühren daher, dass aus systemtheoretischer Sicht (siehe Unterabschnitt 2.1.2) im Grenzfall die funktionalen Abhängigkeiten nicht mehr eindeutig sind. Hier wechseln diese zwischen den Abhängigkeiten von Auftrag und Systemzustand instabil hin- und her und es liegt kein statisches System im für diesen Abschnitt beschriebenen Sinne vor. Neben der prinzipiellen

Funktionstüchtigkeit des Vorgehens gibt die Betrachtung des statischen Falls eine erste Abschätzung der zu erwartenden Performance-Werte, die für sich komplex verhaltende Systeme im Bereich von 50 %, bei deterministischem Verhalten bis nahe an die 100 % reichen können. Damit liegen die Werte signifikant über denen des zufälligen Würfelns, treffen so aber noch keine Aussage über die Güte des Prognosetools bei dynamischen Systemen. Der folgende Abschnitt 7.3 evaluiert das Vorgehen für diesen dynamischen Fall mit synthetisch erzeugten Daten.

7.3 Validierung im dynamischen System mit synthetischen Daten

Zur Evaluation des Vorgehens im dynamischen Fall werden die Daten entsprechend den Vorgaben aus Unterabschnitt 6.1.2 synthetisch generiert. Dies beinhaltet die Erweiterung der Feature-Anzahl auf bis zu 100 und die Berücksichtigung von nun sieben unterschiedlichen zu prognostizierenden Zielkennzahlen (Targets). Dabei werden die Möglichkeiten zur Datensatzgenerierung genutzt, um Streuung, Fehl- und Falschdatenanteil sowie die Latenz zu variieren.

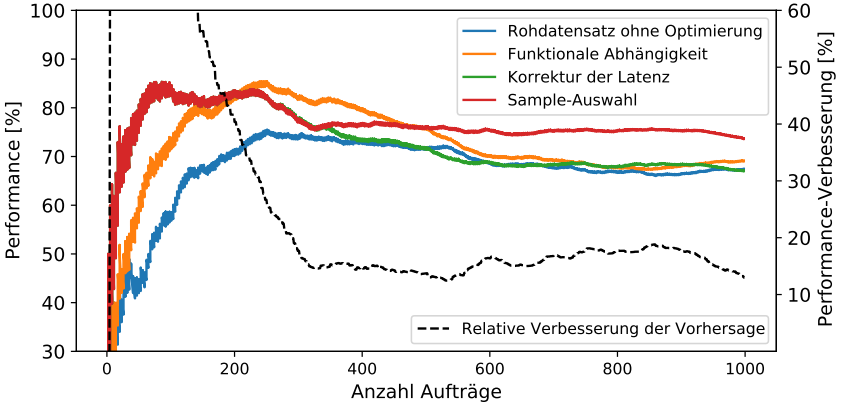


Abbildung 7.5: Performance-Verbesserung im Vergleich zu Rohdatenmodell

Betrachtet werden nun die Auswirkungen der drei Optimierschritte ‚Detektion funktionaler Abhängigkeiten', ‚Korrektur der Latenz' und ‚Sample-Auswahl' auf die Prognoseleistung (Performance). Diese werden genau in der Reihenfolge additiv eingesetzt. Abbildung 7.5

zeigt dazu die Anteile der bis zu dem jeweiligen Auftrag richtig prognostizieren Turbulenzwerte. Die der Prognose zugrunde gelegten Daten entsprechen im Bezug auf die Systemdynamik denen aus Unterabschnitt 6.1.2 mit einer ersten Systemänderung nach ungefähr 250 Aufträgen. Ab etwa diesem Punkt steigt auch die Performance bei Nutzung des Rohdatensatzes (ohne funktionale Abhängigkeiten, Latenzkorrektur und Sample-Auswahl) nicht weiter an. Verglichen werden nun die Optimierschritte mit genau dem Klassierer, welcher mit dem Rohdatensatz ohne Optimierung erstellt wurde. Zusätzlich dargestellt ist die relative Verbesserung der Vorhersage. Das ist der auf den Rohdatensatz ohne Optimierung bezogenen Performance-Zuwachs. Dieser ist bei kleinen Auftragszahlen größer 100 %, da hier die Sample-Anzahl im Vergleich zur Feature-Anzahl relativ gering ist. So haben falsch zugeordnete Feature-Werte bei noch nicht korrekt bestimmter Latenz große negative Auswirkungen auf das Prognoseergebnis, die durch die Bestimmung der funktionalen Abhängigkeiten korrigiert werden. Die dynamischen Systemänderungen (etwa ab Auftrag 250) senken die Performance dann spürbar ab. Die Optimierungen sollen genau diese Auswirkungen dämpfen. Tatsächlich liegen die Performance-Werte durchgehend über denen der Prognosen mit dem Rohdatensatz. Erkennbar ist zudem, dass sich die Performance nach Latenzkorrektur und Sample-Auswahl bis zur ersten Systemveränderung nicht unterscheidet. Danach bringt die selektive Auswahl der Samples einen weiteren Performance-Vorteil, der mit wachsender Anzahl an Aufträgen und den folgenden weiteren Systemänderungen immer deutlicher wird. Die relative Verbesserung der Prognose pendelt sich über den gesamten Zeitraum auf Werte zwischen 10 und 20 % ein. In Summe trägt vor allem die Sample-Auswahl zur signifikanten Verbesserung bei, weil sie die Informationen aus bestimmten Zeitabschnitten der Vergangenheit besser berücksichtigt (siehe Abbildung 6.5).

Kritisch zu betrachten ist der kaum messbare Vorteil durch die Latenzkorrektur. Sie trägt auch nach der ersten Systemänderung kaum mehr zur Verbesserung bei. In Teilen liegt die Prognose-Performance sogar unter der des Rohdatensatzmodells. Daher werden nun die für die Latenzbestimmung berechneten Werte im Rahmen der Evaluation gesondert betrachtet:

Im Vergleich zur Kreuzkorrelation (siehe Abbildung 6.12) erweist sich die Latenzbestimmung mittels Punktwolkendichte bereits als weit weniger tragfähig. Zwar eignet sich dieses Vorgehen gut zur Unterscheidung zwischen Auftrags- und System-Features (siehe Abbildung 6.9 mit nicht kausalen und plausiblen Latenzen in rot und grün), jedoch ist das Rauschen um die tatsächliche Latenz (hier im Beispiel 20 Tage) im Vergleich zur Kreuzkorrelation wesentlich größer. Es lässt sich kein eindeutiger Extremwert festlegen. Abbildung 7.6 zeigt für das selbe Feature-Target-Paar aus Abbildung 6.12, dass die Abtastung der Punktwolkendichten kein eindeutiger Index für die Latenz ist. So sind um den Punkt der erwarteten Latenz von 20 Tagen keine signifikanten Extremwerte zu erkennen, mit denen der Latenzwert bestimmt werden könnte.

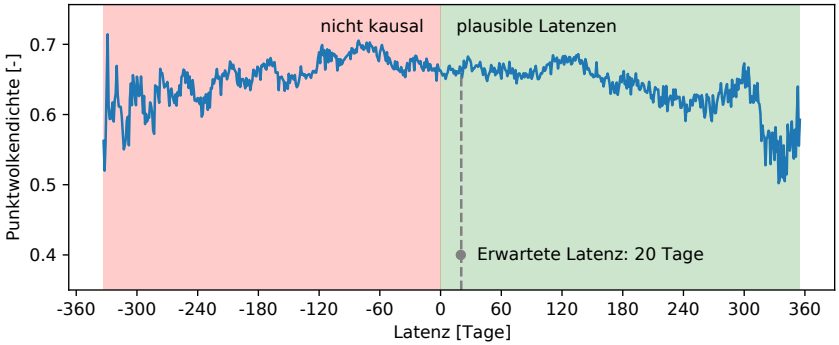


Abbildung 7.6: Punktwolkendichte zur Bestimmung der Latenz

Neben der Nutzung der Punktwolkendichten werden zur Prüfung der Tauglichkeit der Feature-Relevanzen für die Latenzbestimmung alle plausiblen Feature-Target-Paare auf ihre Vorhersagequalitäten über die möglichen Zeitversätze abgetastet. Hier zeigt sich im Vergleich zur Punktwolkendichte lokal ein scheinbar passendes Maximum, jedoch ist eine eindeutige Bestimmung der Verzögerung zwischen Ursache und Wirkung nicht möglich (Abbildung 7.7).

Die Latenzbestimmung mittels Punktwolkendichte und Feature-Relevanz zeigt hier erhebliche Nachteile im Vergleich zur Faltungsmethode. Die beiden Ansätze sind damit für die genaue Latenzbestimmung nicht robust einsetzbar. Verwendung finden sie für die Eingrenzung auf plausible Zeitbereiche (nicht-negativ, wenige Wochen), welche die Ergebnisse der

Faltungsmethode bestätigt oder Latenzbereiche ausgeschlossen werden.

Die Anwendung des Vorgehens auf einen synthetischen Datensatz mit enthaltener Sys-

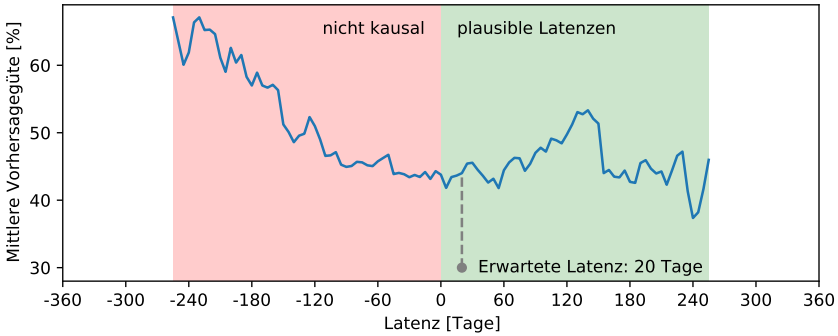


Abbildung 7.7: Feature-Relevanz zur Bestimmung der Latenz

temdynamik zeigt eine Performance-Verbesserung von mindestens 10 % gegenüber der bisherigen direkten Nutzung von Rohdatensätzen. Die Detektion von Latenzen stößt jedoch beim Einsatz der Punktwolkendichten und der Feature-Relevanzen an ihre Grenzen. Der größte Performance-Sprung wird bei einem Datensatz mit 1000 Aufträgen durch die gezielte Auswahl der Samples erzielt. Im Folgenden wird die Methodik auf einen Realdatensatz angewandt, um den Grad der Verbesserung in der Anwendung zu evaluieren.

7.4 Validierung im realen System mit Produktionsdaten

Um das Ergebnis im Anwendungszusammenhang abschließend zu prüfen, soll das Vorgehen auf Daten eines realen Produktionssystems angewandt werden. Von der *phg Peter Hengstler GmbH + Co. KG* in Deißlingen wurden dazu dankenswerterweise die vorliegenden Daten bereitgestellt.

7.4.1 Unternehmen der Primärdatenquelle

Die *phg* bietet mit etwa 300 Mitarbeiterinnen und Mitarbeitern individuelle Lösungen in den Bereichen Verbindungs- und Datentechnik an. Seit 1974 ist die *phg* branchenübergreifender Partner für die Kernprodukte Gerätesteck- und Rundsteckverbinder, umspritzte und

Hybrid-Kabel inklusive einbaufertiger Komplettsystemlösungen mit 100 %-Prüfung (siehe Abbildung 7.8).

Über mehrere Industrieprojekte entwickelte sich eine vertrauensvolle Zusammenarbeit, auf welcher auch der transparente Austausch zu den über Jahre erhobenen Produktionsdaten fußt. Die Datenakquise sollte die für Klassierung nutzbaren Produktionsdaten (siehe Abbildung 5.4) möglichst breit abdecken. Dazu gehört zum einen, die Daten in ihrer Art (Auftrags- und Systemdaten) darzustellen, sowie zum anderen, den für die ML-Methode ausreichenden Umfang (>10.000 Aufträge/Jahr) zu erreichen. Der etwa 1,2 Gigabyte große Datensatz enthält Exporte aus dem betriebsinternen BDE-System sowie in tabellarischer Form dokumentierte Daten.

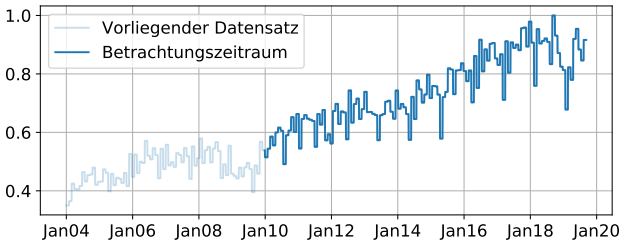
Abbildung 7.8: Auszug aus dem Produktportfolio des Anwendungsfalls (Bildquelle: www.phg.de)

Für die **zu prognostizierenden Turbulenzen** wurden folgende **Zielkennzahlen** festgelegt: Von den auftragsbezogenen Daten werden Durchlaufzeit in der Produktion, Durchlaufzeit des Gesamtauftrags und Ausschussquote betrachtet. Bei den Systemdaten sind es WIP, Strom- und Gasverbrauch sowie Krankheitsquote.

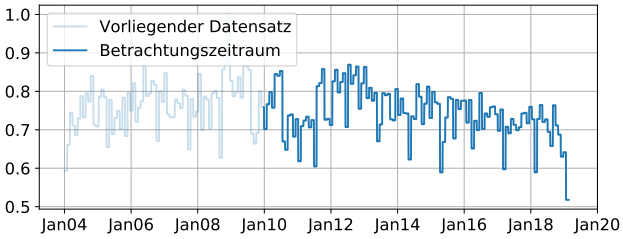
Der betrachtete Zeitraum erstreckt sich von Februar 2010 bis September 2020. Darin enthalten sind fortwährende Änderungen in der Mitarbeiteranzahl (operative) und der Organisationsstruktur im wertschöpfenden Bereich. So entstand im Betrachtungszeitraum eine weitere Arbeitsgruppe und die Kapazitäten in einzelnen Bereichen wurden fortwährend erhöht, im Reinraum sogar verdoppelt. Weiter änderten sich permanent Produktportfolio,

Kundenstruktur und Auftragsmengen. Der Aufgabe der Arbeit entsprechend enthält dieser Datensatz Informationen über die volatile Variantenfertigung eines dynamischen Produktionssystems.

Die Datensätze zu Strom- und Gasverbrauch (Abbildungen 7.9 und 7.10) tasten monatsfein ab und enthalten sowohl Absolutwerte als auch auf die Produktivität bezogene Relativwerte. An den Zeitreihen ist zu erkennen, dass sie den Betrachtungszeitraum nicht immer vollständig abdecken (z.B. Gasverbrauch ab Mitte 2014). Wegen teilweise fehlenden Werten und nicht plausiblen Datenpunkten wurde vor der Überführung in den Trainingsdatensatz an den Zeitreihen das obligatorische Preprocessing inklusive Normierung durchgeführt.



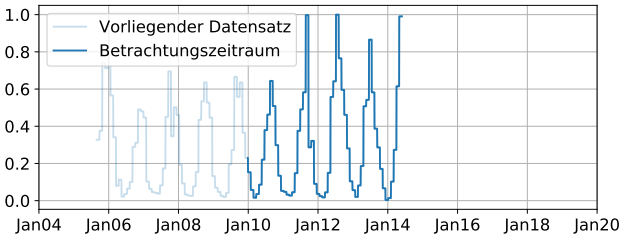
(a) Absoluter Verbrauch



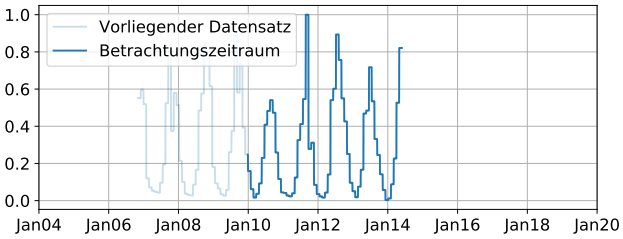
(b) Verbrauch je produzierter Einheit

Abbildung 7.9: Zeitreihe Energie: Stromverbrauch (normiert)

Die Informationen zu den Beständen liegen in Form von Eventsequenzen von Lagerzu- und -abgängen vor. Die kumulierte Betrachtung ergibt die Zeitreihen der Absolutbestände einzelner Lagerstufen (Abbildung 7.11). Abschnittsweise zeigt sich sehr ähnliches Verhalten



(a) Absoluter Verbrauch



(b) Verbrauch je produzierter Einheit

Abbildung 7.10: Zeitreihe Energie: Gasverbrauch (normiert)

zwischen den Zeitreihen. Dies deutet entweder auf für die Prognosen dienliche hohe Korrelationen oder lediglich auf redundanten Informationsgehalt hin.

Die anonymisierten Daten zu den Krankheitstagen sind nach Fertigungsbereichen strukturiert (Abbildung 7.12). Somit können bei entsprechender Datenlage Turbulenzen dieser Kennzahl bereichsfein prognostiziert werden. Da die Krankheitstage der Einzelbereiche als Absolutwerte immer im niedrigen einstelligen Bereich liegen und die Abtastrate mit monatlichen Werten relativ grob ist, wird als zu prognostizierende Zielkennzahl die Gesamtsumme aller Krankheitstage gewählt. In ähnlicher Form wie die Krankheitstage liegen die Zeitreihen zu Plankapazitäten und tatsächlichem Kapazitätsangebot vor. Nachdem diese sich ähnlich stetig und analog zur Anzahl der Mitarbeitenden verhalten, wird auf eine gesonderte Darstellung verzichtet.

Für die auftragsbeschreibenden Daten stehen die Log-Daten von über 140.000 Produktionsaufträgen und etwa 1,2 Millionen Vorgängen mit zusammen über 150 Attributen (im Trainingsdatensatz dann ‚Features') zur Verfügung. Die wichtigsten in den Attributen enthaltenen Informationen sind Starts und Abschlüsse von Produktionsaufträgen, Anzahl

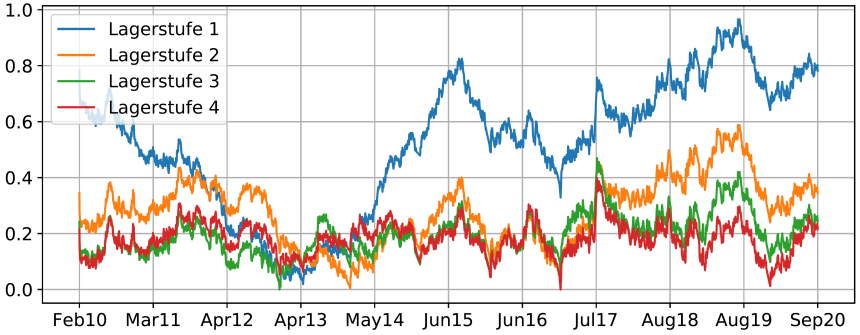


Abbildung 7.11: Zeitreihen Bestände (normiert, Zeitausschnitt)

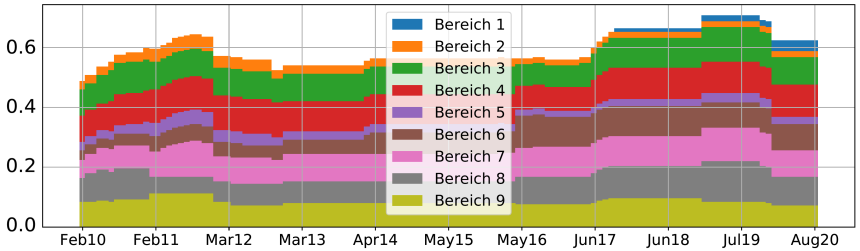


Abbildung 7.12: Zeitreihen Krankheitstage (normiert, Zeitausschnitt)

an Komponenten, Bearbeiter, Baugruppennummer, Freigabemenge, Kunde, Losgröße und Materialnummer. Das Preprocessing reduziert die Attributeanzahl um diejenigen mit redundanten oder gänzlich ohne Informationsgehalt und schließt auch die Auftrags-Samples aus, welche lückenhafte Daten beinhalten. Umgekehrt wird der Datensatz noch um Sekundärattribute erweitert. So wurden für jeden Auftrag der aktuelle Wochentag, das Schichtmodell und die zum Zeitpunkt freigegebenen Aufträge hinzugefügt. Über die Auftragsnummer und den Auftragszeitstempel wurden die einzelnen Auftragsdaten mit den zugehörigen Informationen aus den Vorgangsdaten und den System-Zeitreihen kombiniert, wonach am Ende ein Trainingsdatensatz mit 143.127 Samples und 133 Features vorlag. Dieser wurde zur Erstellung des Prognosemodells für die sieben eingangs genannten Zielkennzahlen verwendet.

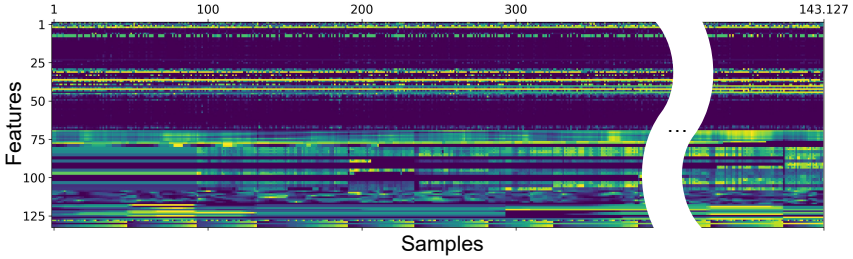
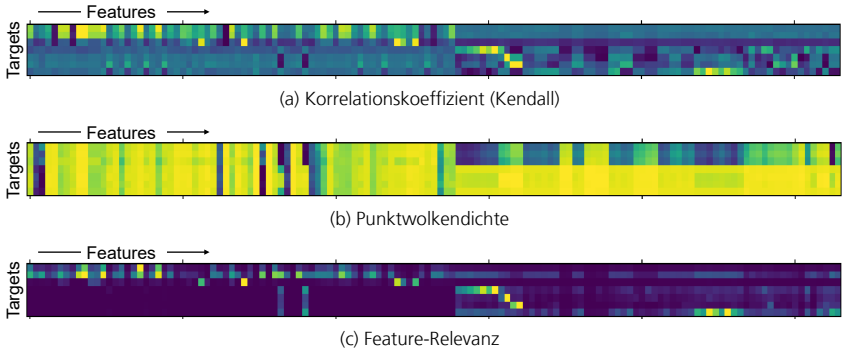


Abbildung 7.13: Visualisierung des Realdatensatzes

7.4.2 Rohdatensatz und Prognoseoptimierung

Abbildung 7.13 visualisiert in Auszügen den Rohdatensatz. Bei den Features von Nummer 1 bis 70 sind die kurzzyklisch stark unterschiedlichen Werte zu erkennen – also die Auftragsattribute. Die Farbe codiert den normierten Wert, weshalb ähnlich niedrige Werte der Features auch über mehrere Hundert Aufträge in dunklem Blau scheinbar gleich dargestellt werden. Bei den übrigen Features (71–133) handelt es sich um die Systemattribute, welche sich eher stetig verhalten und sich entsprechend langzyklischer ändern. Optisch erkennbar sind für die Systemattribute kontinuierliche Wertänderungen. Bei den Auftragsattributen zeigen sich diskrete, für jeden aufeinander folgenden Punkt unterschiedliche Werte.



(a) Korrelationskoeffizient (Kendall)

(b) Punktwolkendichte

(c) Feature-Relevanz

Abbildung 7.14: Funktionale Abhängigkeiten im Realdatensatz

Auf den Datensatz werden nun sequenziell die Methodenschritte zur Detektion der funktionalen Abhängigkeiten, Latenzbestimmung und die Auswahl der Samples, wie in Abschnitt 6.2 beschrieben, angewandt. Der Test mittels dem Korrelationskoeffizienten

ohne Latenzkorrektur (Abbildung 7.14a) bestätigt das implizite Vorwissen über Auftrags- und Systemfeatures. Noch deutlicher zeigt sich dieser Unterschied in Abbildung 7.14b, in der mit der Bewertung der Punktwolkendichten nicht-lineare Zusammenhänge aufgezeigt werden. Für den Ausschluss von Features ohne relevanten Informationsgehalt wird noch die ‚Feature Importance' in Abbildung 7.14c dargestellt. Bei den nun folgenden Optimierungen der Ensembles werden für jedes Target dann entsprechend die performantesten Features ausgewählt.

Als Bezugsgröße werden die Vorhersagegüten des Vorgehens ohne die drei Optimierschritte verwendet. Hier zeigen die Abbildungen 7.15 und 7.16 die Prognoseergebnisse für den Rohdatensatz ohne Feature-Auswahl, Latenzbestimmung und Sample-Selektion. Da zufallsbasierte Baumklassierer in ihrem Prognoseergebnis streuen, sind die einzelnen Versuchswerte im Hintergrund grau dargestellt (gut zu erkennen in Abbildungen 7.15 (b), DLZ_Auftrag) und deren Standardabweichungen als vertikale Fehlerbalken mit angegeben. Den Legenden entsprechend farbig dargestellt sind die Mittelwerte.

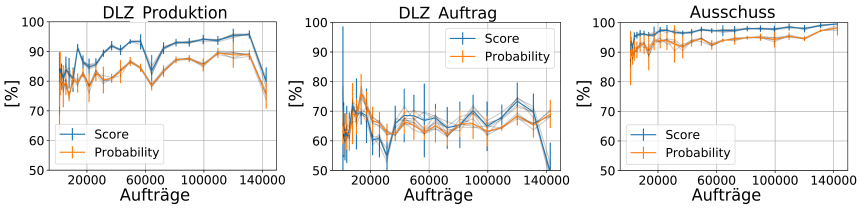


Abbildung 7.15: Vorhersagegüte Zielkennzahlen (Auftrag)

Zeitpunkt und Ausmaß dynamischer Systemänderungen sind nun nicht mehr bekannt. Die abschnittsweise schlechteren Score-Werte sind dabei ein deutlicher Hinweis darauf, dass der Trainingsdatensatz nicht durchgehend nur *ein* vorherrschendes System repräsentiert. Zunächst erfolgt die Betrachtung der Auftrags-Zielkennzahlen Durchlaufzeit in der Produktion, Durchlaufzeit des Gesamtauftrags und Ausschussquote. Beim Ausschuss zeigt sich allgemein eine hohe Prognosegüte von über 90 % bei weitgehender Unabhängigkeit von den Systemänderungen. Mit größer werdendem Datensatz steigt der Score über den gesamten Betrachtungszeitraum weiter an. Performance-Verbesserungen sind wegen der bereits hohen Prognosegüte hier kaum zu erwarten. Bei den beiden Durchlaufzeiten

jedoch zeigen sich mehrfach zwischenzeitliche Performance-Abfälle, die auf systemische Änderungen hindeuten. Am augenscheinlichsten ist die niedrige Prognosequalität der letzten Datenpunkte, welche das Corona-Jahr 2020 umfassen. Die Score-Werte liegen mit 80 % bis 95 % bei der Durchlaufzeit in der Produktion und 50 % bis 75 % bei der Durchlaufzeit des Gesamtauftrags niedriger als beim Ausschuss. Hier sind damit die größten Performance-Verbesserungen zu erwarten.

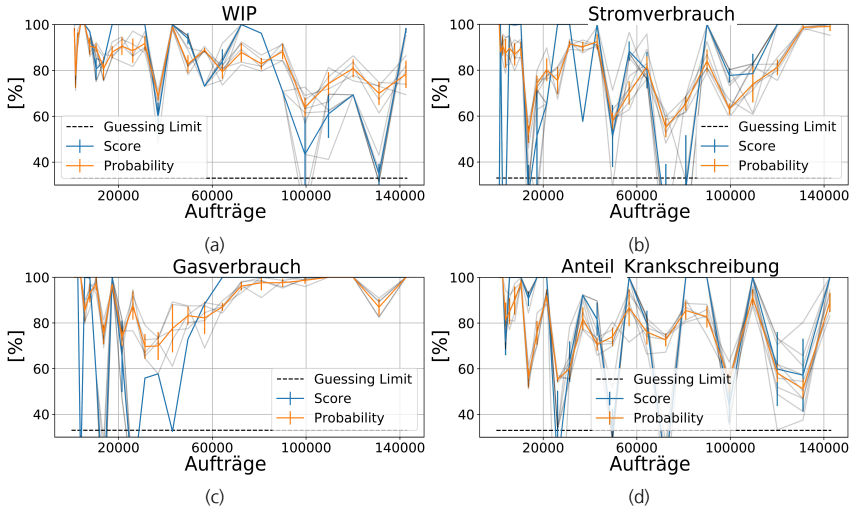


Abbildung 7.16: Vorhersagegüte Zielkennzahlen (System)

Bei der Betrachtung der System-Zielkennzahlen WIP, Strom- und Gasverbrauch sowie Krankenstand zeigt sich ein wesentlich unschärferes Bild: WIP wird initial mit annähernd 100 % sehr gut prognostiziert, zeigt dann jedoch immer wieder – teils erheblich – schlechtere Prognosegüten mit Werten nahe am zufälligen Würfeln. Ähnlich verhalten sich die Vorhersagen von Stromverbrauch und Krankenstand. Abschnitte mit extrem guten Prognosewerten werden gefolgt von solchen unterhalb der Würfelgrenze. Bei den Werten zum Gasverbrauch sei darauf hingewiesen, dass ab etwa der Hälfte der Aufträge *keine* Daten vorlagen (siehe Abbildung 7.10), was zu durchgehenden Score-Werten von 100 % führt. Die im Vergleich zu den Auftragsdaten zuvor wesentlich stärkeren Schwankungen der Score-Werte lassen eindeutige Performance-Verbesserungen in den Turbulenzprognosen

erwarten. Die Vorhersagen unter Zuhilfenahme der drei Optimierschritte können nun gegen die des Rohdatensatzes getestet werden.

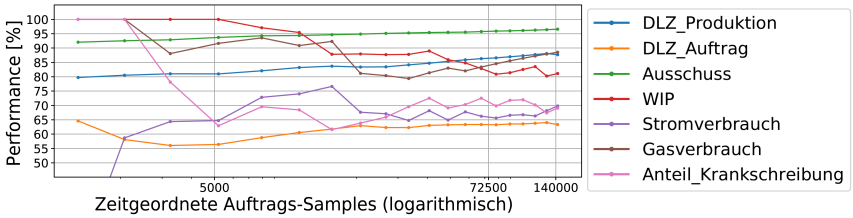


Abbildung 7.17: Performance der Referenzprognosen

Abbildung 7.17 zeigt dazu den Anteil kumulierter richtiger Prognosen für den Referenzdatensatz. Um das Prognoseverhalten auch in der Anfangsphase bei noch wenigen Aufträgen gut erkennbar zu machen, wird eine logarithmische Skala auf der Sample-Achse verwendet. Datensätze, welche das Produktionssystem gleichbleibend repräsentieren, ergeben annähernd konstante Datenreihen. Mit zunehmend größer werdenden Datensatz kann die Performance dabei weiter ansteigen. Typische Beispiele für diesen Fall sind die Werte zum Ausschuss (grün) und zur Durchlaufzeit Produktion (blau). Ähnlich verhält sich die Gesamtdurchlaufzeit (orange), jedoch auf entsprechend niedrigerem Performance-Niveau. Analog zu den Score-Werten aus Abbildung 7.16 werden die Prognosen durch das dynamische System stärker negativ beeinflusst. Keine der Kennzahlen kann das zwischenzeitlich erreichte Niveau halten. Gegen diese Referenzprognosen werden nun die drei Verbesserungsschritte getestet.

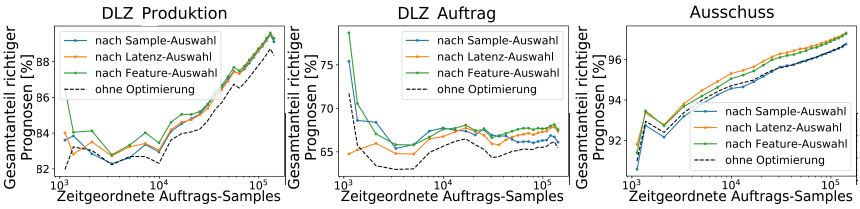


Abbildung 7.18: Vorhersagegüte optimiert (Auftrag)

Zunächst zeigen die Graphen in Abbildung 7.18 die Wirkung der *einzelnen* Optimierschritte auf die Prognosen zu den Auftragskennzahlen. Das bedeutet, dass in den nachfolgenden

Darstellungen die Optimierschritte alternativ angewandt und dargestellt werden. Bei der späteren Evaluierung der Performance-Verbesserung insgesamt werden die Schritte additiv kombiniert. Zunächst erfolgt die Betrachtung der Einzelschritte.

Bei den beiden Durchlaufzeiten entfaltet sich die Wirkung der Methode erwartungsgemäß ab einer gewissen Größe des Datensatzes. Initial sind bei kleiner Sample-Anzahl teilweise geringe Performance-Verluste zu verzeichnen. Ursache dafür sind die noch fehlenden Einbußen durch Systemdynamiken und den zwangsläufig auftretenden Verlusten durch Manipulation des eigentlich gut repräsentierenden Datensatzes. Mit größer werdender Sample-Anzahl tritt die Methodenwirksamkeit ein und die Anteile korrekter Prognosen liegen im Folgenden durchgehend über den Referenzwerten ohne Optimierung. Dies gilt auch für die bereits ohne Optimierschritte gut prognostizierbaren Werte zur Zielkennzahl Ausschuss. Die Sample-Auswahl zeigt im Vergleich zu den Untersuchungen mit synthetischen Daten (Abschnitt 7.3) jedoch in den Graphen b) und c) die gewünschten Vorteile nicht bis über alle Auftrags-Samples. Der Grund dafür liegt in der relativ geringen Auflösung der Abtastung, die sich aufgrund der Limitierung der Rechenleistung auf sechs Abschnitte beschränkt (siehe Abbildung 5.12). Kurzzyklische Systemänderungen in Zeiträumen von kleiner einem Sechstel des Gesamtdatensatzvolumens werden dann nicht mehr abgebildet und können nicht mehr verbessernd auf die Performance wirken.

Für die Systemkennzahlen zeigt sich auf den gesamten Datensatz bezogen ein ebenso positives Ergebnis (Abbildung 7.19): Die kumulierten Prognose-Performances liegen über der Referenzprognose ohne Optimierung. Die genaue Betrachtung der Einzelkennzahlen offenbart an einigen Stellen auffälliges Verhalten. So verbessert die Latenzauswahl zum WIP erst bei großer Sample-Anzahl die Turbulenzprognose positiv. Grund dafür ist die Vielfältigkeit der Ursachen für WIP-Änderungen. Beispielsweise beeinflussen sowohl die Menge der Auftragsfreigaben in einem bestimmten Zeitintervall als auch die bereits gestarteten Aufträge die Kennzahl. Die Methodik hier detektiert für jede Feature-Target-Beziehung eine konstante Latenz, und somit keine mit zunehmenden Aufträgen veränderliche. Dieser Schwachpunkt des heutigen Stands der Methodik kommt hier zum Tragen.

Die übrigen drei Kennzahlen b), c) und d) zeigen über das erste Drittel des Datensatzes keinerlei Effekte durch die Methodik. Im volatilen System scheint in dem Sample-Bereich noch kein ausreichend großer Informationsgehalt im Datensatz vorhanden zu sein, um

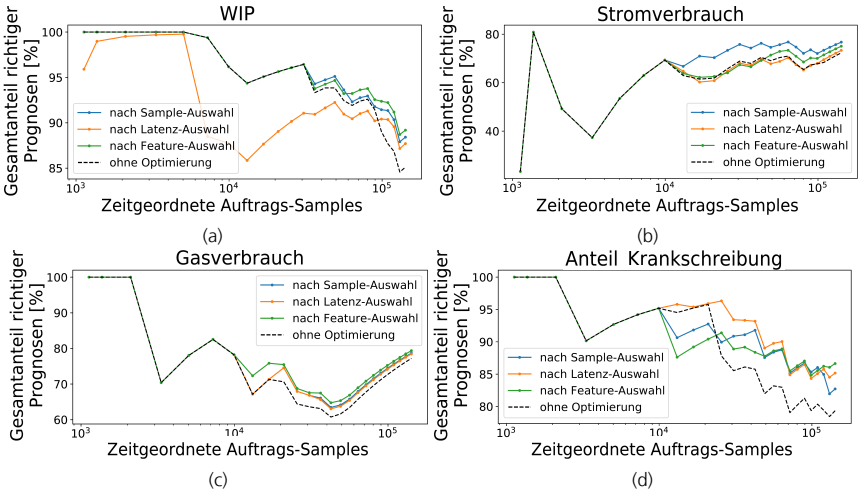


Abbildung 7.19: Vorhersagegüte optimiert (System)

daraus durch Selektion von Features oder Samples Verbesserungen erzielen zu können. Die Beobachtung deckt sich mit den stark streuenden Performances aus den Turbulenzprognosen zu den Zielkennzahlen mit dem Referenzdatensatz (siehe Abbildung 7.16). Bei den Energiekennzahlen zu Strom- und Gasverbrauch zeigt sich erneut ein zwischenzeitlich positiver Effekt durch die Sample-Auswahl, der sich mit größer werdendem Datensatz analog zur Erläuterung bei den Auftragskennzahlen aus Gründen der Abtastrate wieder weitgehend auflöst. Beim Anteil der Krankschreibungen – die Kennzahl mit dem größten Performance-Abfall – zeigt sich auch die größte Wirkung auf die Prognosegüte: Nach den Optimierschritten liegen die Werte zwar zunächst temporär darunter, über den gesamten Prognosezeitraum jedoch am deutlichsten oberhalb der Referenzprognosen.

7.4.3 Evaluierung

Die Evaluation eines Forschungsergebnisses erfolgt am besten durch Vergleich mit einem bestehenden Verfahren. Zunächst war angedacht, die Anteile der richtig prognostizierten Turbulenzen mit den auf Erfahrungswissen basierenden Vorhersagen aus der Produkti-

on direkt zu vergleichen. Dieses Vorgehen wurde aus zwei Gründen verworfen: Erstens bestand bislang keine Kennzahlenvorhersage in dem Sinne, dass Durchlaufzeiten oder voraussichtlicher Krankenstand zur Entscheidungsfindung für Anpassungen im System prognostiziert wurden. Die Auftragsplanung übernimmt diese Aufgabe. Mit überraschend auftretenden Turbulenzen wird daher eher reaktiv denn proaktiv umgegangen. Zweitens ist eine nachträgliche auf Erfahrungswissen basierende Vorhersage anhand der Features im Datensatz nicht praktikabel umsetzbar, da bei 140.000 Samples in überschaubarer Zeit keine für einen Vergleich statistisch belastbare Anzahl an ‚menschlich erzeugten' Prognosen erstellt werden können.

Daher beinhaltet die Evaluation nicht den Vergleich mit dem gegenwärtigen Verfahren im Unternehmen (da nicht existent), sondern mit dem Prognoseergebnis *ohne* Optimierschritte. Eine solche Prognsoe nutzt den zu jedem Zeitpunkt möglichst großvolumigen Trainingsdatensatz zur Modellerstellung – also die oben beschriebenen initialen Referenzprognosen und deren Performances. Die so erzielten Scores werden verglichen mit den jeweils besten Werten der einzelnen Optimierschritte bzw. deren Kombination. Der Vergleich ergibt eine prozentuale Performance-Verbesserung, die in Abbildung 7.20 für jede Zielkennzahl dargestellt ist.

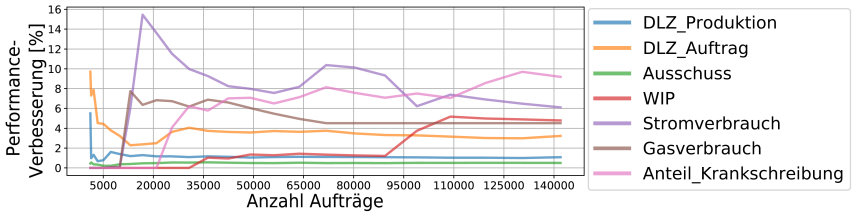


Abbildung 7.20: Performance-Verbesserung im Vergleich zu Referenzprognosen

Gut zu erkennen sind die kategorischen Unterschiede zwischen den Prognosen zu Auftrags- und Systemturbulenzen. Erstere weisen auch die drei geringsten Performance-Verbesserungen auf, während die übrigen allesamt signifikant darüber liegen. Dies steht im Einklang mit den Erkenntnissen aus den Prognosen ohne Optimierung: Auftragskennzahlen mit hier vergleichsweise geringer Abhängigkeit von der Systemdynamik lassen entsprechend geringe Performance-Steigerungen der ohnehin bereits hohen Score-Werte

erwarten. Gleichzeitig werden die Prognosegüten bei den Systemkennzahlen, welche durch Systemänderungen stark beeinflusst sind, durch gezielte Auswahl und Verarbeitung des Trainingsdatensatzes verbessert.

Die Prognoseverbesserungen der drei Auftragskennzahlen pendeln sich nach nicht relevanten initialen Schwankungen bei kleiner Auftragsanzahl auf annähernd konstante Werte mit am Ende 1,1 % (DLZ Produktion), 3,2 % (DLZ Auftrag) und 0,5 % (Ausschuss) ein. Darüber liegen Gasverbrauch (4,5 %), WIP (4,8 %), Stromverbrauch (6,1 %) und der Anteil der Krankschreibungen mit 9,2 %. Die Turbulenzprognosen für das System konnten somit insgesamt und in den Fällen von WIP und Krankschreibung sogar über den ganzen Zeitraum kontinuierlich verbessert werden. Das zeigt, dass das Vorgehen gerade für die Kennzahlen starke Verbesserungen erzielt, welche bei herkömmlicher Verwendung des Trainingsdatensatzes vergleichsweise schlechte Ergebnisse liefern. In Summe ist die Verbesserung eindeutig belegbar und darüber hinaus für die beiden Kennzahlenkategorien ‚Auftrag' und ‚System' auch plausibel mit Werten bezifferbar.

Um für am Thema Forschende und potenzielle zukünftige Nutzer eine Entscheidung für den Einsatz des Vorgehens zu erleichtern, bereitet der letzte Abschnitt des Kapitels die Erfahrungen im Rahmen dieser Arbeit zur Modellerstellung und Übertragung auf einen Realdatensatz so auf, dass Aufwand und Nutzen abgeschätzt werden können.

7.5 Abschätzung von Aufwand und Nutzen

Mit ‚Daten sind das neue Öl' wird gemeinhin zum Ausdruck gebracht, dass es in zunächst nicht näher bestimmter Weise lohnenswert sei, Daten als Rohstoff zu verwerten und für sich nutzbar zu machen. Nach Abschluss der Lösungsumsetzung in dieser Arbeit kann auch hier der Versuch einer solchen Nutzenbewertung unternommen werden. Die Schwierigkeit besteht darin, dass zum einen weder im Markt verfügbare noch beim Unternehmen des Anwendungsfalls eingesetzte konkurrierende Lösungen für einen Leistungsvergleich existieren. Zum anderen ist die in dieser Arbeit vorgestellte Lösung eine Untersuchung darüber, wie Turbulenzprognosen in volatilen Systemen verbessert werden können und eben nicht ein industriell einsetzbares fertiges Produkt. Dennoch lässt sich der potenzielle Nutzen der Lösung strukturiert beschreiben. Durch einen Vergleich mit dem dafür nötigen Aufwand wird

es möglich, Argumente für oder gegen den Einsatz der Methodik zu finden. Die folgenden Betrachtungen orientieren sich an der allgemeinen Aufwands- und Nutzenabschätzung beim Einsatz von Software (Fröhlich 1995, 155 ff.).

Drei Phasen werden dabei unterschieden: Einführung, Hochlauf und eingeschwungene Nutzungsphase (Abbildung 7.21). Ziel ist dabei aus betriebswirtschaftlicher Sicht, die Amortisationszeit zu bestimmen, um die Investitionsentscheidung zu treffen. Für die Bewertung von Aufwand und Nutzen wird die Lösung quantitativ und qualitativ analysiert.

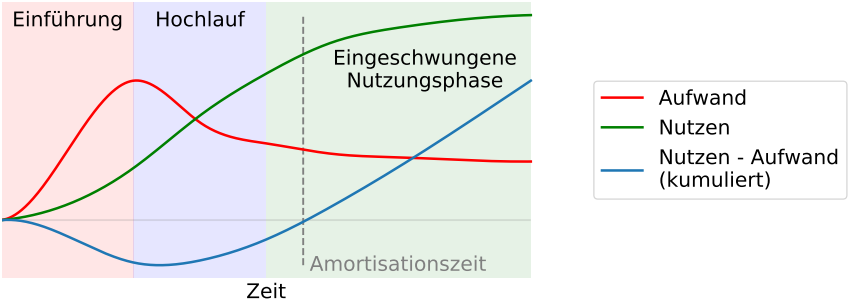


Abbildung 7.21: Phasen des Methodeneinsatzes

Ein bewährtes Vorgehen für die Bewertung des Aufwands ist die Sammlung der monetären und zeitlichen Kosten. Die für die Lösung dieser Arbeit auftretenden Kostentreiber der jeweiligen Phasen sind folgend zusammengefasst:

Einführung: Bedarfsanalyse, Überzeugungsarbeit, Hardwarebeschaffung, Datenauswahl und -akquise, Schulung der Nutzer, Rollout

Hochlauf: Debugging, Systemanpassung und -skalierung, Funktionsergänzungen, ineffiziente Nutzung bzw. noch nicht erfolgte Lerneffekte

Eingeschwungene Nutzungsphase: Fortbildung, Weiterentwicklung, Updates, permanente Datenakquise, Wartung

Den Kosten gegenüber steht der Nutzen, welcher *quantitativ* bewertbar in Kostenreduzierung bzw. Ertragsverbesserung eingeteilt wird:

Kostenreduzierungen durch: Verringerung von Ausfallzeiten, Krankenstand, Ausschuss, Umweltauswirkungen, ungeplanten Beständen und frühzeitig erkannter Schulungsbedarf in der Belegschaft

Ertragsverbesserungen durch: Umsatzsteigerung durch Verbesserung von Produktivität, Servicegrad, Produktqualität, Kapazitätsausnutzung, Einhaltung von Qualitätsstandards, Prozesseffizienz sowohl auf Ebene der Planung und Steuerung als auch in den Wertschöpfungsbereichen sowie verbesserte Energieeffizienz, verbessertes Predictive Maintenance mit dadurch verlängerter Lebensdauer der Ressourcen

Durch die Validierung im Fall 3 (Realdaten), können die Aufwände für die Implementierung realistisch abgeschätzt werden. Die Bedarfsanalyse, welche im wesentlichen die Auswahl der für den Turbulenzsteckbrief erwünschten KPIs enthält, wird mit ausgewählten Leitungsfunktionen durchgeführt. Dazu gehören Datenauswahl und -akquise. Dies kann binnen eines Arbeitstages abgearbeitet werden. Das Preprocessing der Inputdaten benötigt mit diesem Werkzeug durchschnittlich einen Personentag pro Turbulenz-KPI. Hinzu kommen etwa eine Arbeitswoche Schulung für den sicheren Umgang mit der Softwarelösung. Nach der Hardwarebeschaffung, welche sich konservativ auf **< €5.000,-** beläuft erfordert der Rollout eine weitere Arbeitswoche. Damit ist die Einführung abgeschlossen. Für den Hochlauf ist mit etwa zwei weiteren Monaten an Personalaufwand zu rechnen. Das bedeutet aufaddiert ein initialer Aufwand von etwa **drei Personenmonaten**, welcher auch mit der herangezogenen Vergleichsquelle plausibel erscheint (Fröhlich 1995, S. 157). In der eingeschwungenen Nutzungsphase ist bei intensivem Einsatz und voller Nutzenerzielung eine Person permanent gebunden.

In Summe trägt der quantifizierbare Nutzen direkt zur Wettbewerbsfähigkeit des Unternehmens bei. Hinzu kommen noch *qualitative* Vorteile durch den Einsatz des Werkzeugs durch Haupt- und Nebeneffekte. Dazu zählt zunächst die grundsätzliche Erhöhung der Systemtransparenz durch eine übersichtliche Prognosedarstellung als Turbulenzsteckbrief. Abbildung 7.22 zeigt solche Steckbriefe für konkrete Auftrage des realen Datensatzes.

Die darin enthaltenen Informationen können für automatisierte Frühwarnsystem verwendet werden. Die Visualisierung bietet darüber hinaus eine Option zur Dokumentation von Entscheidungswegen. Die Lösung ermöglicht weiter die Identifikation neuer, bislang noch nicht bekannter Anwendungen oder Einsichten. Dies wird erst aufgrund der mit dem Werkzeug generierten Daten möglich, welche Informationen einbeziehen, die herkömmlich nicht berücksichtigt werden konnten (wie beispielsweise Ursache-Wirkungs-Zusammenhänge

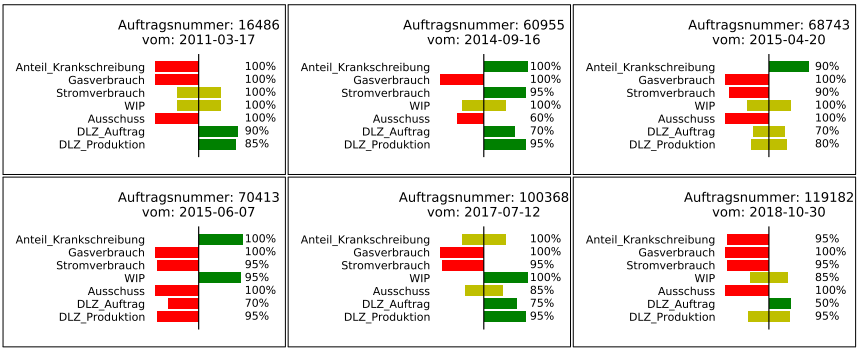


Abbildung 7.22: Turbulenzsteckbriefe für sechs ausgewählte Aufträge (Realdaten)

zum Krankenstand). Langfristig verbessert das die Geschäftsprozesse und erhöht die Datenqualität durch Ermöglichung von Plausibilitätstests.

Auch wenn ein Vergleich mit bislang eingesetzten Lösungen aus oben beschriebenen Gründen nicht möglich ist, können zukünftig diese Aspekte den Nutzen vergleichend ausweisen: Zeitaufwand für KPI-Berechnungen aus bestehenden Bestandsdatensystemen, Trendermittlung und Prognosen, Geschwindigkeit bei Einführung und Hochlauf sowie Kosten für Softwarelizenzen, Zufriedenheit und Vertrauen in die Ergebnisse der Mitarbeitenden bei den jeweiligen Werkzeugen und die Intuitivität bei der Benutzung. Daran anknüpfend seien die bereits offensichtlichen Vorteile des vorliegenden Ansatzes genannt: Implementierung in Open-Source-Software ist bei gleichzeitigem Betrieb auf herkömmlicher Hardware möglich. Der Ansatz zeichnet sich durch hohe Robustheit gegenüber mangelnder Datenqualität und einfache Erweiterung bei zusätzlich gewünschten Features und Ziel-KPIs aus. Zentraler Vorteil ist, dass beim Ansatz mit Entscheidungsbäumen für die Prognosen kein analytisches Wirkmodell notwendig ist. So sind auch Fehler in der Modellbildung ausgeschlossen. Scheinkorrelationen sind wenig wahrscheinlich und werden – selbst wenn sie bestehen – nicht mehr berücksichtigt, sobald sie die Prognosequalität negativ beeinflussen. Die Lösung kann in kurzer Zeit installiert werden und ist in sich ein skalierendes, lernendes System, welches Prognosen trotz Systemdynamik ermöglicht.

Oft sieht sich der Standpunkt ‚Daten sind das neue Öl' dem verkürzten Vorwurf ausgesetzt, dass die so generierten Informationen nicht nachweisbar als direkter Unternehmensnutzen bewertet werden können. Mit dem Wissen jedoch, dass sich die Datensammler, welche 1881 die ersten verlässlichen Wetterdaten in Deutschland erhoben (Deutscher Wetterdienst 2023), demselben Vorwurf ausgesetzt sahen, kann man diesem heute mit einer gewissen Gelassenheit begegnen. Der Nutzen der Datenerhebung wird oft erst mit den daraus gewonnenen Informationen und deren Einsatz ersichtlich.

Fazit:

Kapitel 7 gibt zunächst einen Überblick zum Evaluationsvorgehen dieser Arbeit. Dieses betrachtet ein statisches, ein dynamisches und ein reales Produktionssystem. Die Betrachtungen beim statischen Produktionssystem bestätigten zunächst die Hypothese, dass Produktionskennzahlen und damit als Turbulenz definierte Abweichungen von ihren Soll-Werten durch das hergeleitete Vorgehen prognostiziert werden können. Die Untersuchungen im Überlast-, Unterlast und Grenzfall haben zudem gezeigt, dass bei typischer Feature-Anzahl bei Industrieaufträgen Datenvolumen von mehr als 100 Samples notwendig sind, um stabile Prognosen durchzuführen (**Forschungsfrage 1**).

Daraus folgernd wurden dynamische Systeme untersucht, in denen drei Systemübergänge verteilt auf 1000 Aufträge im Datensatz repräsentiert wurden. Zur Untersuchung, wie viel Systemdynamik das System abfangen kann (**Forschungsfrage 2**), wurden sowohl kontinuierliche als auch abrupte Systemübergänge modelliert. Die drei Optimierschritte Detektion der funktionalen Abhängigkeit, Korrektur der Latenz und Sample-Auswahl wurden auf den Datensatz angewandt. Dabei konnte nachgewiesen werden, dass die Performance-Verbesserung durch die Optimierschritte bei industrieähnlichen Datensätzen ab 200 Samples stabil über 10 % liegt.

Für die Beantwortung von **Forschungsfrage 3** zum Gültigkeitsbereich des Vorgehens dienen die Erkenntnisse der synthetischen Daten zur Datensatzbeschaffenheit und die abschließenden Experimente am Realdatensatz. Hier bestätigen sich die theoretisch im Abschnitt 5.2 (Struktur des Datensatzes) hergeleiteten grundlegenden Unterschiede zwischen Auftrags- und Systemdaten. Diese spiegeln sich in den Prognosegüten und den jeweilig potenziellen Verbesserungen durch die Optimierschritte wider: Während bei den

auftragsbezogenen Kennzahlen die Prognose-Performance im Vergleich zu den Systemkennzahlen besser ist, wirken sich die Optimierschritte bei den Systemkennzahlen stärker aus und erzielen so Prognoseverbesserungen zwischen vier und zehn Prozent im dynamischen System. Kritisch zu betrachten sind die im Vergleich zu den synthetischen Daten auffallend schwache Verbesserungen durch die Sample-Auswahl. Dies wird in der kritischen Würdigung in Abschnitt 8.2 erneut aufgegriffen.

Für die Frage über das Verhältnis des nötigen Aufwands zum Mehrwert (**Forschungsfrage 4**) wird abschließend dargelegt, welche Vorteile die Einführung einer CART-basierten Turbulenzprognose mit sich bringt. Erwartungsgemäß sind die dafür notwendigen Aufwände nicht abschließend monetär und zeitlich bezifferbar, jedoch überwiegt nach der Aufwand- und Nutzenabschätzung beim Einsatz von Software der Nutzen bei Weitem. Die Hardwarekosten für den Methodeneinsatz sind zu vernachlässigen und die zeitlichen Aufwände beschränken sich auf wenige Personenmonate.

8 Zusammenfassung

Produktionssysteme sind geprägt von hoher Dynamik. Sich volatil verhaltende Auftragslasten führen zu unerwarteten Abweichungen in den Zielkennzahlen – genannt Turbulenz. Herkömmliche Ansätze bilden die Komplexität in der Produktion für eine Turbulenzprognose nicht ausreichend genau ab: Ursache-Wirkungs-Beziehungen sind nie vollständig erfassbar und Systemänderungen über der Zeit beeinträchtigen das Ergebnis. Es fehlt ein datenbasierter Ansatz zur Prognose und Bewertung drohender Turbulenz, um frühzeitig reagieren zu können.

Die Lösung dieser Arbeit nutzt baumbasierte Klassierer. Die neu entwickelten Optimierschritte ermöglichen verbesserte Prognoseergebnisse gegenüber dem herkömmlichen Vorgehen. Der Machbarkeitsnachweis erfolgt mittels Anwendung auf Simulationsdaten, synthetisch generierte Daten und Daten eines realen Produktionssystems. Die Ergebnisse zeigen Prognoseverbesserungen von bis zu 10 %. Konkret nutzbar sind *Turbulenzsteckbriefe*, welche die zugehörigen Kennzahlen und ihre jeweiligen Prognosesicherheiten darstellen.

8.1 Inhalt und Ergebnisse

Ausgangssituation und Problemstellung werden in **Kapitel 1** dargelegt: Moderne Produktionssysteme sind von äußerer und innerer Komplexität geprägt. Dieser begegnen Unternehmen mit Robustheit. Die Konsequenzen aus der sich nie vollständig schließenden Lücke zwischen Komplexität und Robustheit werden in dieser Arbeit als Turbulenz bezeichnet. Sie schlägt sich in unerwünschten Abweichungen von Zielkennzahlen nieder und wird so quantitativ bewertbar. Die Schwierigkeit liegt darin, Turbulenz ex ante zu quantifizieren.

Kapitel 2 befasst sich mit den formalwissenschaftlichen Grundlagen. Produktionssysteme werden als offene Systeme betrachtet und in einem Übersichtsmodell beschrieben. Dieses stellt den Zusammenhang zwischen Komplexität, Resilienz und Turbulenz in volatilen Produktionssystemen dar. Grundlage für das Problemverständnis sind mathematische Werkzeuge, welche in solche zur Zeitreihenbetrachtung und Korrelationen sowie Kausalität,

Stationarität und Stabilität eingeteilt und beschrieben werden. Das Kapitel schließt mit Ausführungen zu Methoden des maschinellen Lernens mit besonderem Augenmerk auf Klassierer, welche einen Teil der späteren Lösung darstellen.

In **Kapitel 3** werden für die Aufgabe geeignete Standardmodelle im Bereich von Data Science und Data Mining betrachtet. Die Lösung enthält eine geeignete Kombination daraus. Der heutige Stand zur Turbulenzbewertung zeigt die disziplinübergreifend unterschiedliche Verwendung des Turbulenzbegriffs. Dabei liegt die Forschungslücke darin, dass klassische Ansätze auf der qualitativen Bewertung von kaum reproduzierbaren und verifizierbaren Aussagen beruhen.

Aus der Aufgabenstellung werden in **Kapitel 4** Systemanforderungen für den Datensatz, den Klassierer, die Lösung, und die Evaluation hergeleitet. Dazu werden Anwendungsszenarien genutzt, um die Detailanforderungen durch möglichst hohen Praxisbezug zu erkennen. Die Szenarien sind Proof of Concept, Prognose bei Systemdynamik und Realsystem.

Die Lösung wird in **Kapitel 5** hergeleitet. Diese definiert die Struktur des Datensatzes, das Prognosemodell als parametrisierter Baumklassierer und die zugehörigen Optimierschritte. Der Datensatz enthält neben den Auftragseigenschaften die zum jeweiligen Zeitpunkt vorliegenden systembeschreibenden Attribute. Der zentrale Teil der Lösung ist die Vorverarbeitung, Erstellung und Analyse von systembeschreibenden Zeitreihen: Mittels Faltung und Korrelationsanalysen lassen sich die zeitversetzten kausalen Zusammenhänge finden und diese zur Turbulenzbewertung einsetzen.

Kapitel 6 beschreibt die konkrete Umsetzung der Lösung in Python. Genutzt werden dafür Random Forests. Kern der Umsetzung sind die drei Optimierschritte, welche die Prognosegüte bei Turbulenzen in dynamischen Systemen verbessern: Detektion funktionaler Abhängigkeiten zwischen Features und Targets, Latenzbestimmung von auslösender Ursache zu gemessener Wirkung sowie gezielte Auswahl der trotz Systemänderung noch gültigen Samples. Es werden beispielhaft die Turbulenzsteckbriefe für sechs Aufträge mit jeweils sieben Zielkennzahlen grafisch dargestellt.

Die Evaluation in **Kapitel 7** ist dreiteilig. Zunächst zeigt die Verifikation, dass der Ansatz sowohl bei Systemüber- und -unterlast als auch im Übergangsbereich plausible Ergebnisse liefert. Danach wird der Ansatz für dynamische Systeme mit synthetischen, realitätsnahen Daten getestet. Es zeigen sich durchgehend Verbesserungen von 10–20 % gegenüber dem

herkömmlichen Prognosevorgehen. Abschließend werden am Industrieanwendungsfall die erwartete Turbulenz der Zielkennzahlen DLZ (Produktion und Auftrag), Ausschuss, WIP, Strom- und Gasverbrauch sowie Anteil Krankschreibungen prognostiziert. Kennzahlen mit bereits hoher Vorhersagegüte profitieren kaum von den Optimierschritten. Die mit vergleichsweise niedriger Performance erfahren jedoch Verbesserungen bis zu annähernd 10 %. Die Abschätzung von Aufwand und Nutzen zeigt potenzielle Kostenreduzierungen und Ertragsverbesserungen auf.

8.2 Kritische Würdigung

In diesem Abschnitt erfolgt die Betrachtung von Anspruch und Ergebnis dieser Arbeit. Kriterien sind die Beantwortung der Forschungsfragen, die Erfüllung der Anforderungen der empirischen Forschung (Objektivität, Validität, Reliabilität und Generalisierbarkeit) sowie die Bewertung der Lösung. Am Ende werden die methodischen Schwachstellen benannt.

Die Arbeit ist als *Überprüfung der Machbarkeit* eines gänzlich neuen methodischen Ansatzes konzipiert, wobei trotz des erhöhten wissenschaftlichen Risikos ein Scheitern der Lösungsidee nicht eingetreten ist. Die zentrale Forschungsfrage zielt auf das *Wie* bei der Bewertung kundeninduzierter Turbulenz ab. Dafür werden Systeme als theoretisches Konstrukt, die Turbulenz von ihrer physikalischen Bedeutung und die aktuell operativ eingesetzten ‚Turbulenzbewerter' untersucht. Aus Formalwissenschaft, Technikwissenschaft und dem Stand der Technik konnte die Lösung zielgerichtet erarbeitet werden (Objektivität).

Die Unterfragen werden folgendermaßen adressiert: Zum nötigen Datenvolumen, sind realistische Untergrenzen im Kapitel 7 beschrieben. Trotz Abhängigkeit vom Anwendungsfall erweisen sich die angegebenen Werte bislang als gut übertragbar, da sie für Simulationsdaten, den synthetisch generierten und den realen Datensatz in ähnlicher Weise funktionieren (Validität). Der induktive Schluss auf *alle* Anwendungsfälle ist natürlich nicht zulässig.

Weit fallabhängiger sind Aussagen zur erlaubten Dynamik im Produktionssystem (zweite Unterfrage). Dazu wurden abrupte und kontinuierliche Systemübergänge betrachtet. Die dazu erstellten Grafiken in Unterabschnitt 6.1.2 geben Auskunft über die Sensibilität des Vorgehens gegenüber Systemänderungen. Da in den hier verwendeten synthetischen Daten

die Änderungen grundlegender Art waren, sind alle dazu getroffenen Aussagen *konservativ* und das Vorgehen kann aller Erwartung nach auch kurzzyklischere Änderungen handhaben (Reliabilität).

Die Allgemeingültigkeit kann als hoch angesehen werden für solche Produktionssysteme, welche die hier verwendeten Turbulenzkennzahlen einsetzen (Generalisierbarkeit). Bei extrem großem Kundentakt und damit geringer Auftragsanzahl (z.B. Schiffs- und Orgelbau) verlässt das Vorgehen den plausiblen Bereich, wenn die jährliche Sample-Anzahl zweistellig oder gar kleiner wird. Am anderen Ende der Skala – bei hoher Sample- und Feature-Anzahl – werden die Grenzen aktuell durch die verfügbare Rechenkapazität gesetzt.

Auch die Unterfrage zum Aufwand und Mehrwert konnte positiv beantwortet werden. Das Verhältnis des später sich erschließenden Sekundärnutzens guter Prognosen zu den geringen Investitionsausgaben überzeugt. Die möglichen Kostenreduzierungen und Ertragsverbesserungen werden umfangreich genannt. Die Anforderungen der empirische Forschung (Töpfer 2010, 231 ff.) werden damit erfüllt.

Das praktische Interesse steht im Fokus anwendungsorientierter Grundlagenforschung. Dafür wurde die theoretische Lösung in ausführbaren Code übertragen. Grundlage dafür sind drei Szenarien, von denen zwei auf die praktische Nutzung in der Forschung und die Anwendung auf Daten realer Produktionssysteme abzielt. Dazu werden zehn Attribute herausgearbeitet (Abbildung 4.2), deren Erfüllung im Evaluationskapitel 7 nachgewiesen wird.

Das Vorgehen weist an zwei Punkten Schwachstellen auf. Erstens werden die Latenzen bislang als konstant angenommen. Latenzzeiten sind sowohl produktspezifisch unterschiedlich als auch über der Zeit vom Systemzustand abhängig. Die vereinfachende Annahme der Konstanz wirkt sich negativ auf das Prognoseergebnis aus und kann zu falsch detektierten Abhängigkeiten führen. Die erarbeitete Lösung zur Latenzzeitbestimmung ist zwar praktikabel, jedoch mathematisch wenig elegant und in ihrer Umsetzung rechenintensiv. So mangelt es aufgrund dieser Schwächen an Präzision und Recheneffizienz.

Zweiter Kritikpunkt ist der Umgang mit Produktiv- und Stillstandszeiten. Zwar sind diese sowohl in den Daten implizit enthalten, werden aber im Umgang bei Latenzbestimmung und Feature-Auswahl nicht speziell behandelt. Die in der Realität auftretenden zeitidenti-

schen Buchungen führen zu künstlichen Produktivitäts-Peaks, welche nicht die tatsächlich aufgetretenen Ereignissen repräsentieren. Somit wird der als kontinuierlich angenommene Zeitfortschritt in den Daten künstlich gestaucht und gedehnt. Eine Bereinigung der Zeitreihen um Stillstände und eine Plausibilisierung von Gleichzeitereignissen sind bislang nicht enthalten.

8.3 Ausblick

Aus den beiden methodischen Kritikpunkten ergeben sich der zukünftige Forschungsbedarf sowie Handlungsvorschläge. Da diese Arbeit in ihrer gesamten Konzeption eine *Untersuchung der Machbarkeit* ist, sind folgende wissenschaftliche Entwicklungspfade und Möglichkeiten in der industriellen Anwendung denkbar:

Bislang erfolgte noch keine Optimierung des Klassierers an sich. Somit sind die Möglichkeiten hinsichtlich Prognoseleistung bei weitem nicht ausgeschöpft. Potenziale liegen in der Auswahl und Parametrierung der Klassierer und versprechen Performance- und Recheneffizienzsteigerungen. Gerade im identifizierten Grenzbereich, wenn zu hohe Systemdynamik gerichtete Prognosen unmöglich machen, könnte nicht-überwachtes Lernen der nächste Schritt sein. Zudem sind die im Datensatz enthaltenen Zeitabhängigkeiten noch nicht ausreichend beforscht. Völlig offen ist der Umgang mit nicht-konstanten Latenzen zwischen Ursache und Wirkung.

Zentrale Handlungsanweisung für die Industrie ist die Nutzung der in dem speziell strukturierten Datensatz enthaltenen Informationen. So kann die Turbulenzbewertung *einzelner* Aufträge auf Trendanalysen erweitert werden. Die Korrelationsanalyse wird so mittels Ursache-Wirkungs-Detektion zur Grundlage für die automatisierte Entscheidungsunterstützung. Der Vorteil liegt darin, dass innerhalb solch komplexer – wenngleich klar abgrenzbarer – Produktionssysteme Scheinkorrelationen statistisch höchst unwahrscheinlich sind. Der Einsatz als Detektor für nicht offensichtliche kausale Zusammenhänge kann die Basis für Kennzahlenoptimierungen außerhalb klassischer Verbesserungsprozesse sein. Die Optimierungen sind leicht auf Kenngrößen wie Energieverbrauch oder Kundenzufriedenheit erweiterbar. Eine Übertragung auf PPS durch die Unterstützung der Entwicklung neuer Planungsregeln ist denkbar und wurde in einer ersten Veröffentlichung bereits angedacht

(Bauer et al. 2021). Der Transfer vom B2B-Bereich auf B2C ist im Hinblick auf *Dynamic Pricing* vielversprechend. So wären Turbulenzzuschläge bei ‚unpassenden' Bestellungen oder entsprechende Rabatte bei Auftragsänderungen, die turbulenzsenkend wirken, möglich. Der Ansatz kann so zu einem Ausgangspunkt auf dem Weg zum tiefen Verständnis komplexer Produktionssysteme werden.

fin

Literatur

Agrawal et al. 1993 Agrawal, Rakesh; Imielinski, Tomasz; Swami, Arun, 1993. Database Mining: A Performance Perspective. *IEEE transactions on knowledge and data engineering* **5** (6), S. 914–925. DOI: 10.1109/69.250074

Ahrens 2000 Ahrens, Gritt, 2000. *Das Erfassen und Handhaben von Produktanforderungen: Methodische Voraussetzungen und Anwendung in der Praxis*. Berlin, Technische Universität Berlin, Univ., Diss., 2000. URN: urn:nbn:de:kobv:83-opus-428

Albawi et al. 2017 Albawi, Saad; Mohammed, Tareq Abed; Al-Zawi, Saad, 2017. Understanding of a convolutional neural network. In: *2017 International Conference on Engineering and Technology (ICET)*. Antalya, S. 1–6. ISBN: 978-1-538-61949-0

Amit & Geman 1997 Amit, Yali; Geman, Donald, 1997. Shape Quantization and Recognition with Randomized Trees. In: Ullmann, Shimon (Hrsg.): *Neural Computation*, S. 1545–1588. DOI: 10.1162/neco.1997.9.7.1545

Anderson et al. 2001 Anderson, Neil; Herriot, Peter; Hodgkinson, Gerard P., 2001. The practitioner-researcher divide in Industrial, Work and Organizational (IWO) psychology: Where are we now, and where do we go from here? *Journal of Occupational and Organizational Psychology* **74** (4), S. 391–411. DOI: 10.1348/096317901167451

Angée et al. 2018 Angée, Santiago; Lozano-Argel, Silvia I.; Montoya-Munera, Edwin N.; Ospina-Arango, Juan-David; Tabares-Betancur, Marta S., 2018. Towards an Improved ASUM-DM Process Methodology for Cross-Disciplinary Multi-organization Big Data & Analytics Projects. In: Uden, Lorna; Hadzima, Branislav; Ting, I-Hsien (Hrsg.): *Knowledge Management in Organizations*, S. 613–624. ISBN: 978-3-319-95204-8

Arens et al. 2010 Arens, Tilo; Hettlich, Frank; Karpfinger, Christian; Kockelkorn, Ulrich; Lichtenegger, Klaus; Stachel, Hellmuth, 2010.
Mathematik. 2., korrigierter Nachdr. Heidelberg: Spektrum Akademischer Verlag.
ISBN: 978-3-827-41758-9

Ashby 1957 Ashby, W. Ross, 1957.
An introduction to cybernetics. Second Impression. London: Chapman & Hall.
ISBN: 978-1-614-27765-1

Axmann et al. 2019 Axmann, Bernhard; Hamberger, Werner; Liegl, Thomas, 2019.
Digitalisierung der Fabrik – Datenqualität als Schlüssel zum Erfolg.
ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb **114** (5), S. 302–305.
DOI: 10.3139/104.112083

Ayodele 2010 Ayodele, Taiwo Oladipupo, 2010.
Types of Machine Learning Algorithms.
In: Zhang, Yagang (Hrsg.): *New Advances in Machine Learning*. Rijeka, Croatia: InTech, S. 19–47.
DOI: 10.5772/9385

Backhaus et al. 2015 Backhaus, Klaus; Erichson, Bernd; Weiber, Rolf, 2015.
Fortgeschrittene multivariate Analysemethoden: Eine anwendungsorientierte Einführung. 3., überarbeitete und aktualisierte Auflage. Berlin und Heidelberg: Springer Gabler.
ISBN: 978-3-662-46086-3

Barber 1992 Barber, Herbert F., 1992.
Developing Strategic Leadership: The US Army War College Experience.
Journal of Management Development **11** (6), S. 4–12.
DOI: 10.1108/02621719210018208

Barredo Arrieta et al. 2020 Barredo Arrieta, Alejandro; Díaz-Rodríguez, Natalia; Del Ser, Javier; Bennetot, Adrien; Tabik, Siham; Barbado, Alberto; Garcia, Salvador; Gil-Lopez, Sergio; Molina, Daniel; Benjamins, Richard; Chatila, Raja; Herrera, Francisco, 2020.
Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI.
Information Fusion **58** (3), S. 82–115.
DOI: 10.1016/j.inffus.2019.12.012

Barthel 2006 Barthel, Holger, 2006.
Modell zur Analyse und Gestaltung des Bestellverhaltens für die variantenreiche Serienproduktion. Heimsheim: Jost-Jetter-Verl. IPA-IAO Forschung und Praxis 449.
Univ., Diss., 2006.
ISBN: 978-3-939-89002-7

Bartolomei et al. 2006 Bartolomei, Jason E.; Hastings, Daniel E.; de Neufville, Richard; Rhodes, Donna H., 2006.
Screening for Real Options In an Engineering System: A Step Towards Flexible System Development: PART I: The Use of Coupled Design Matrices to Create an End-to-End Representation of a Complex Socio-Technical System.
In: *INCOSE International Symposium*.
Wiley Online Library.
Los Angeles, CA, S. 1241–1257.
URL: https://incose.onlinelibrary.wiley.com/doi/pdfdirect/10.1002/j.2334-5837.2006.tb02809.x
Zugriff am: 10.08.2023

Bauer et al. 2017 Bauer, Dennis; Maurer, Tobias; Henkel, Christian; Bildstein, Andreas, 2017.
Big-Data-Analytik: Datenbasierte Optimierung Produzierender Unternehmen. Stuttgart.
DOI: 10.5281/zenodo.803099

Bauer et al. 2021 Bauer, Dennis; Böhm, Markus; Bauernhansl, Thomas; Sauer, Alexander, 2021.
Increased resilience for manufacturing systems in supply networks through data-based turbulence mitigation.
Production Engineering **15** (3-4), S. 385–395.
DOI: 10.1007/s11740-021-01036-4

Bauernhansl 2003 Bauernhansl, Thomas, 2003.
Bewertung von Synergiepotenzialen im Maschinenbau.
Wiesbaden, Univ., Diss: Deutscher Universitätsverlag.
Wirtschaftswissenschaft.
Univ., Diss., 2003.
ISBN: 978-3-824-40688-3

Bauernhansl 2014a Bauernhansl, Thomas, 2014.
Automobilindustrie ohne Band und Takt - Forschungscampus ARENA2036.
In: *23. Deutscher Materialfluss-Kongress*.
ISBN: 978-3-180-92232-4

Bauernhansl 2014b Bauernhansl, Thomas, 2014.
Die Vierte Industrielle Revolution – Der Weg in ein wertschaffendes Produktionsparadigma.
In: Bauernhansl, Thomas; ten Hompel, Michael; Vogel-Heuser, Birgit (Hrsg.): *Industrie 4.0 in Produktion, Automatisierung und Logistik*.
Wiesbaden: Springer Fachmedien, S. 5–35.
ISBN: 978-3-658-04681-1

Bauernhansl et al. 2016 Bauernhansl, Thomas; Krüger, Jörg; Reinhart, Gunther; Schuh, Günther, 2016.
WGP-Standpunkt Industrie 4.0. Darmstadt: Wissenschaftliche Gesellschaft für Produktionstechnik WGP e. V.
URL: https://wgp.de/wp-content/uploads/WGP-Standpunkt_Industrie_4-0.pdf

Beer 2000 Beer, Stafford, 2000.
Decision and control: The meaning of operational research and management cybernetics. Reprinted. Chichester: Wiley.
ISBN: 0-471-94838-1

Bertalanffy 1968 Bertalanffy, Ludwig von, 1968.
General system theory: Foundations, development, applications. Rev. ed. New York, NY: Braziller.
ISBN: 978-0-807-60453-3

Bishop 2009 Bishop, Christopher M., 2009.
Pattern recognition and machine learning. Corrected at 8th printing 2009. New York, NY: Springer.
ISBN: 978-0387-31073-2

Bitkom 2019 Bitkom, 2019.
Blick in die Blackbox: Nachvollziehbarkeit von KI-Algorithmen in der Praxis,
URL: https://www.bitkom.org/sites/default/files/2019-10/20191016_blick-in-die-blackbox.pdf
Zugriff am: 16.10.2019

Böhm et al. 2021a Böhm, Markus; Bauernhansl, Thomas; Jeschke, Sabina, 2021.
Behavior of decision forest classification in dynamic manufacturing systems.
Procedia CIRP **104**, S. 524–529.
DOI: 10.1016/j.procir.2021.11.088

Böhm & Bauernhansl 2021 Böhm, Markus; Bauernhansl, Thomas, 2021.
Data-based turbulence evaluation in production systems.
Procedia CIRP **99**, S. 686–691.
DOI: 10.1016/j.procir.2021.03.119

Böhm et al. 2021b Böhm, Markus; Erlach, Klaus; Bauernhansl, Thomas, 2021.
Maschinelles Lernen zur Prognose von Auftragskennzahlen/Machine learning for the forecasting of key figures of customer orders.
wt Werkstattstechnik online **111** (03), S. 124–129.
DOI: 10.37544/1436-4980-2021-03-32

Bouchier 1901 Bouchier, Edmund Spenser, 1901.
Aristoteles: Posterior Analytics. Translated by E.S. Bouchier.
Oxford: Blackwell

Breiman 1998 Breiman, Leo, 1998.
Classification and regression trees. Repr. Boca Raton: Chapman & Hall.
ISBN: 978-0-412-04841-8

Brinzer & Schneider 2019 Brinzer, Boris; Schneider, Konstanze, 2019.
Komplexitätsbewertung in der Produktion.
ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb **114** (10), S. 647–652.
DOI: 10.3139/104.112168

Brockwell & Davis 1991 Brockwell, Peter J.; Davis, Richard A., 1991.
Time series: Theory and methods. 2. ed. New York, Berlin und Heidelberg: Springer.
ISBN: 978-1-441-90319-8

Buescher et al. 2011 Buescher, Christian; Mayer, Marcel; Schilberg, Daniel; Jeschke, Sabina, 2011.
Artificial Cognition in Autonomous Assembly Planning Systems.
In: Jeschke, Sabina (Hrsg.): *Proceedings of the 4th international conference on Intelligent Robotics and Applications - Volume Part II*.
Berlin, Heidelberg: Springer-Verlag, S. 168–178.
ISBN: 978-3-642-25488-8

Bullinger et al. 2003 Bullinger, Hans-Jörg; Warnecke, Hans Jürgen; Westkämper, Engelbert, 2003.
Neue Organisationsformen im Unternehmen. Berlin, Heidelberg: Springer.
DOI: 10.1007/978-3-662-08934-7

Burkart & Huber 2021 Burkart, Nadia; Huber, Marco F., 2021.
A Survey on the Explainability of Supervised Machine Learning.
Journal of Artificial Intelligence Research **70**, S. 245–317.
DOI: 10.1613/jair.1.12228

Chalmers & Bergemann 2007 Chalmers, Alan F.; Bergemann, Niels, 2007.
Wege der Wissenschaft: Einführung in die Wissenschaftstheorie. 6., verbesserte Auflage. Berlin, Heidelberg: Springer-Verlag.
ISBN: 978-3-540-49490-4

Chen et al. 2012 Chen; Chiang; Storey, 2012.
Business Intelligence and Analytics: From Big Data to Big Impact.
MIS Quarterly **36** (4), S. 1165.
DOI: 10.2307/41703503

Cheng et al. 2018 Cheng, Ying; Chen, Ken; Sun, Hemeng; Zhang, Yongping; Tao, Fei, 2018.
Data and knowledge mining with big data towards smart production.
Journal of Industrial Information Integration **9** (9), S. 1–13.
DOI: 10.1016/j.jii.2017.08.001

Chmielewicz 1994 Chmielewicz, Klaus, 1994.
Forschungskonzeptionen der Wirtschaftswissenschaft. 3., unveränd. Aufl. Stuttgart: Schäffer-Pöschel. Sammlung Poeschel 92.
ISBN: 978-3-791-09197-6

Chollet 2018 Chollet, François, 2018.
Deep Learning mit Python und Keras: Das Praxis-Handbuch : vom Entwickler der Keras-Bibliothek. 1. Auflage. Frechen: MITP.
ISBN: 978-3-958-45838-3

Cisco public 2018 Cisco public, 2018.
Cisco Visual Networking Index: Forecast and Trends, 2017–2022.
San José, URL:
https://cyrekdigital.com/uploads/content/files/white-paper-c11-741490.pdf
Zugriff am: 03.10.2021

Clapham 1983 Clapham, Wentworth B., 1983.
Natural ecosystems. 2. ed. New York, NY: Macmillan Publ. Collier-Macmillan.
ISBN: 978-0-023-22520-8

Coyle & Coyle 1996 Coyle, Robert G.; Coyle, Robert Geoffrey, 1996.
System dynamics modelling: A practical approach. 1. ed. London und Weinheim: Chapman & Hall.
ISBN: 978-0-412-61710-2

Cramer & Kamps 2017 Cramer, Erhard; Kamps, Udo, 2017.
Grundlagen der Wahrscheinlichkeitsrechnung und Statistik: Eine Einführung für Studierende der Informatik, der Ingenieur- und Wirtschaftswissenschaften. 4., korrigierte und erweiterte Auflage. Berlin: Springer Spektrum.
ISBN: 978-3-662-54160-9

Criminisi & Shotton 2013 Criminisi, Antonio; Shotton, Jamie, 2013.
Decision forests for computer vision and medical image analysis. London: Springer. Advances in Computer Vision and Pattern Recognition.
ISBN: 978-1-447-14928-6

Dahlberg 2015 Dahlberg, Rasmus, 2015.
Resilience and Complexity: Conjoining the Discourses of Two Contested Concepts.
Ecology and Society **7** (3), S. 541–557.
DOI: 10.3384/cu.2000.1525.1572541

Dangeti 2017 Dangeti, Pratap, 2017.
Statistics for machine learning: Build supervised, unsupervised, and reinforcement learning models using both Python and R.
Birmingham, UK: Packt Publishing.
ISBN: 978-1-788-29122-4

Dehghantanha 2019 Dehghantanha, Ali, 2019.
Handbook of Big Data and IoT Security. Cham: Springer International Publishing AG.
ISBN: 978-3-030-10542-6

Deutscher Wetterdienst 2023 Deutscher Wetterdienst, 2023.
Wetter und Klima - Deutscher Klimaatlas,
URL: https://www.dwd.de/DE/klimaumwelt/klimaatlas/klimaatlas_node.html
Zugriff am: 14.02.2023

Diekmann 2018 Diekmann, Andreas, 2018.
Empirische Sozialforschung: Grundlagen, Methoden, Anwendungen. 12. Auflage, vollständig überarbeitete und erweiterte Neuausgabe August 2007. Reinbek bei Hamburg: Rowohlt Taschenbuch Verlag. 55678.
ISBN: 978-3-499-55678-4

Döbel et al. 2018 Döbel, Inga; Leis, Miriam; Vogelsang, Manuel; Neustroev, Dmitry, 2018.
Maschinelles Lernen: Eine Analyse zu Kompetenzen, Forschung und Anwendung.
München, URL:
https://www.bigdata-ai.fraunhofer.de/content/dam/bigdata/de/documents/Publikationen/Fraunhofer_Studie_ML_201809.pdf
Zugriff am: 15.08.2023

Dombrowski et al. 2018 Dombrowski, Uwe; Karl, Alexander; Wortmann, Timo; Tschöpe, Sebastian, 2018.
Entwicklung einer Systematik zur Bewertung des Varianten-einflusses auf die Gesamtfabrik.
ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb **113** (11), S. 715–722.
DOI: 10.3139/104.111967

Dürrschmidt 2001 Dürrschmidt, Stephan, 2001.
Planung und Betrieb wandlungsfähiger Logistiksysteme in der variantenreichen Serienproduktion. München: Utz. 152. Univ., Diss., 2001.
ISBN: 978-3-831-60023-6

Eckstein 2014 Eckstein, Peter P., 2014.
Repetitorium Statistik: Deskriptive Statistik - Stochastik - Induktive Statistik. 8., aktualisierte u. erw. Aufl. Wiesbaden: Springer Gabler.
ISBN: 978-3-658-05747-3

Efthymiou et al. 2009 Efthymiou, Konstantinos; Papakostas, Nikolaos; Mourtzis, Dimitris; Chryssolouris, George, 2009.
Fluid Dynamics Analogy to Manufacturing Systems.
In: *42nd CIRP Conference on Manufacturing Systems*.
Grenoble, France.

Eickelmann et al. 2015 Eickelmann, Michel; Wiegand, Mario; Konrad, Benedikt; Deuse, Jochen, 2015. Die Bedeutung von Data-Mining im Kontext von Industrie 4.0. *Zeitschrift für wirtschaftlichen Fabrikbetrieb* **110** (11), S. 738–743. DOI: 10.3139/104.111433

Erlach et al. 2020 Erlach, Klaus; Böhm, Markus; Hartleif, Silke; Leipoldt, Christoph; Ungern-Sternberg, Roman, 2020. Gestaltungsrichtlinien in den Technikwissenschaften: Eine wissenschaftstheoretische Betrachtung. *ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb* **1-2**, S. 77–81. DOI: 10.3139/104.112206

Erlach 2020 Erlach, Klaus, 2020. *Wertstromdesign: Der Weg zur schlanken Fabrik*. 3. Auflage. Berlin und Heidelberg: Springer Vieweg. ISBN: 978-3-662-58907-6

Erlach et al. 2022a Erlach, Klaus; Berchtold, Marc-André; Gessert, Stephan; Kaucher, Christian; Ungern-Sternberg, Roman, 2022. Flexibilität, Wandlungsfähigkeit und Agilität in der Fabrikplanung. *ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb* **117** (7-8), S. 442–447. DOI: 10.1515/zwf-2022-1102

Erlach et al. 2022b Erlach, Klaus; Böttcher, Lena; Teriete, Tim, 2022. Systematik für Kennzahlen in der Produktion. *Zeitschrift für wirtschaftlichen Fabrikbetrieb* **117** (9), S. 558–565. DOI: 10.1515/zwf-2022-1112

Ester & Sander 2000 Ester, Martin; Sander, Jörg, 2000. *Knowledge Discovery in Databases: Techniken und Anwendungen*. Berlin, Heidelberg: Springer. ISBN: 978-3-540-67328-6

Etzion 2011 Etzion, Opher, 2011. *Event processing in action*. Greenwich, Conn.: Manning. ISBN: 978-1-935-18221-4

Eversheim 1989 Eversheim, Walter, 1989. *Organisation in der Produktionstechnik: Arbeitsvorbereitung*. Zweite, neubearbeitete Auflage. Berlin, Heidelberg: Springer. ISBN: 978-3-662-22685-8

Fayyad et al. 1996 Fayyad, Usama M.; Piatetsky-Shapiro, Gregory; Smyth, Padhraic, 1996. Knowledge Discovery and Data Mining: Towards a Unifying Framework. In: Simoudis, Evangelos; Han, Jiawei; Fayyad, Usama M. (Hrsg.): *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. Portland, Oregon, S. 82–88. URL: https://cdn.aaai.org/KDD/1996/KDD96-014.pdf Zugriff am: 09.07.2023

Feldhusen & Grote 2013 Feldhusen, Jörg; Grote, Karl-Heinrich, 2013. *Konstruktionslehre: Methoden und Anwendung erfolgreicher Produktentwicklung*. 8., vollständig überarbeitete Auflage. Berlin und Heidelberg: Springer Vieweg. ISBN: 978-3-642-29568-3

Flack et al. 2020 Flack, Christian; Dreher, Simon; Birk, Alexander; Wilhelm, Yannick, 2020. Process Mining in der Produktion. *Zeitschrift für wirtschaftlichen Fabrikbetrieb* **115** (11), S. 829–833. DOI: 10.3139/104.112459

Forrester 1968 Forrester, Jay W., 1968. Industrial Dynamics - A Response to Ansoff and Slevin. *Management Science* **14** (9), S. 601–618. DOI: 10.1287/mnsc.14.9.601

Fricker 1996 Fricker, Achim Roland, 1996. *Eine Methodik zur Modellierung, Analyse und Gestaltung komplexer Produktionsstrukturen*. 1. Aufl. Aachen: Verl. der Augustinus-Buchh. Aachener Beiträge zu Humanisierung und Rationalisierung 27. Techn. Hochsch., Diss., 1996. ISBN: 978-3-860-73457-5

Fröhlich 1995 Fröhlich, Peter, 1995. *FEM-Leitfaden: Einführung und praktischer Einsatz von Finite-Element-Programmen*. Berlin, Heidelberg: Springer. ISBN: 978-3-642-79383-7

Gadenne 1999 Gadenne, Volker, 1999. *Wissenschaftsphilosophie*. 1. Aufl. Freiburg und München: Alber. 5. ISBN: 978-3-495-48005-2

Gandomi & Haider 2015 Gandomi, Amir; Haider, Murtaza, 2015. Beyond the hype: Big data concepts, methods, and analytics. *International Journal of Information Management* **35** (2), S. 137–144. DOI: 10.1016/j.ijinfomgt.2014.10.007

Gartner 2017 Gartner, 2017.
Prognose zur Anzahl der vernetzten Geräte im Internet der Dinge (IoT) weltweit in den Jahren 2016 bis 2020,
URL: https:
//de.statista.com/statistik/daten/studie/537093/umfrage/anzahl-der-vernetzten-geraete-im-internet-der-dinge-iot-weltweit/
Zugriff am: 03.10.2021

Gerthsen 2006 Gerthsen, Christian, 2006.
Physik. 23., überarb. Aufl. Berlin und Heidelberg: Springer.
ISBN: 978-3-540-25421-8

Gigerenzer 2015 Gigerenzer, Gerd, 2015.
Simply rational: Decision making in the real world. New York, NY: Oxford University Press. Oxford series in evolution and cognition.
ISBN: 978-0-199-39007-6

Gille & Zwißler 2011 Gille, Christian; Zwißler, Frank, 2011.
Bewertung von Wandlungstreibern.
ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb **106** (5), S. 310–313.
DOI: 10.3139/104.110553

Girod et al. 1997 Girod, Bernd; Rabenstein, Rudolf; Stenger, Alexander, 1997.
Einführung in die Systemtheorie. Wiesbaden: Teubner.
ISBN: 978-3-663-09884-3

Gleitsmann-Topp et al. 2009 Gleitsmann-Topp, Rolf-Jürgen; Kunze, Rolf-Ulrich; Oetzel, Günther, 2009.
Technikgeschichte. 1. Aufl. Konstanz: UVK-Verl.-Ges. UTB Geschichte 3126.
ISBN: 978-3-825-23126-2

Goldstein 1969 Goldstein, Sydney, 1969.
Fluid Mechanics in the First Half of this Century.
Annual Review of Fluid Mechanics **1** (1), S. 1–29.
DOI: 10.1146/annurev.fl.01.010169.000245

Govaert et al. 2009 Govaert, Tim; Neirynck, Sven; van Volsem, Sofie; van Landeghem, Hendrik, 2009.
Combining scripting and commercial simulation software to simulate in-plant logistics.
In: *European Simulation and Modelling Conference (ESM)*, S. 114–118.
ISBN: 978-9-077-38152-6

Grote et al. 2018 Grote, Karl-Heinrich; Bender, Beate; Göhlich, Dietmar, 2018.
Dubbel: Taschenbuch für den Maschinenbau. 25., aktualisierte Aufl. 2018. Berlin, Heidelberg: Springer.
ISBN: 978-3-662-54804-2

Guyon 2003 Guyon, Isabelle, 2003.
Design of experiments of the NIPS 2003 variable selection benchmark.
In: *NIPS 2003 workshop on feature extraction and feature selection*.
URL: http://www.clopinet.com/isabelle/Projects/NIPS2003/Slides/NIPS2003-Datasets.pdf
Zugriff am: 22.04.2022

Han et al. 2019 Han, Chi; Mao, Jiayuan; Gan, Chuang; Tenenbaum, Joshua B.; Wu, Jiajun, 2019.
Visual Concept-Metaconcept Learning.
In: *Advances in Neural Information Processing Systems (NIPS)*.
Vancouver, Canada.
URN: 2002.01464v1

Han & Kamber 2006 Han, Jiawei; Kamber, Micheline, 2006.
Data mining: Concepts and techniques. 2. ed. Amsterdam: Elsevier/Morgan Kaufmann.
ISBN: 978-1-558-60901-3

Hayek 1972 Hayek, Friedrich A. Von, 1972.
Die Theorie komplexer Phänomene. Tübingen: Mohr. 36.
ISBN: 978-3-163-33691-9

Hayek 2007 Hayek, Friedrich A. Von, 2007.
Wirtschaftstheorie und Wissen: Aufsätze zur Erkenntnis- und Wissenschaftslehre. Tübingen: Mohr Siebeck. Band 1.
ISBN: 978-3-161-47917-5

Hedtstück 2017 Hedtstück, Ulrich, 2017.
Complex event processing: Verarbeitung von Ereignismustern in Datenströmen. Berlin: Springer Vieweg.
ISBN: 978-3-662-53450-2

Hengen et al. 2004 Hengen, Heiko; Feid, Michael; Pandit, Madhukar, 2004.
Überwacht lernende Klassifikationsverfahren im Überblick, Teil 5.
at - Automatisierungstechnik **52** (10), S. A25–A28.
DOI: 10.1524/auto.52.10.A25.44807

Hering et al. 2013 Hering, Niklas; Meißner, Jan; Reschke, Jan, 2013.
ProSense – Untersuchung „Produktion am Standort Deutschland 2013".
Zeitschrift für wirtschaftlichen Fabrikbetrieb **108** (12), S. 995–998.
DOI: 10.3139/104.111060

Hermann et al. 2008 Hermann, Andreas; Homburg, Christian; Klarmann, Martin, 2008.
Marktforschung: Ziele, Vorgehensweise und Nutzung.
In: Herrmann, Andreas; Homburg, Christian; Klarmann, Martin (Hrsg.): *Handbuch Marktforschung*.
Wiesbaden: Gabler, S. 3–20.
ISBN: 978-3-834-90342-6

Herrera Vidal & Coronado Hernández 2021 Herrera Vidal, Germán; Coronado Hernández, Jairo Rafael, 2021.
Complexity in manufacturing systems: a literature review.
Production Engineering **15** (3-4), S. 321–333.
DOI: 10.1007/s11740-020-01013-3

Himme 2007 Himme, Alexander, 2007.
Gütekriterien der Messung: Reliabilität, Validität und Generalisierbarkeit.
In: Albers, Sönke; Klapper, Daniel; Konradt, Udo; Walter, Achim; Wolf, Joachim (Hrsg.): *Methodik der empirischen Forschung*,
S. 375–390.
DOI: 10.1007/978-3-8349-9121-8_25

Holland & Scharnbacher 2010 Holland, Heinrich; Scharnbacher, Kurt, 2010.
Grundlagen der Statistik: Datenerfassung und -darstellung, Maßzahlen, Indexzahlen, Zeitreihenanalyse. 8., aktualisierte Aufl. Wiesbaden: Gabler.
ISBN: 978-3-834-92010-2

Holling 1973 Holling, Crawford S., 1973.
Resilience and stability of ecological systems.
Annual review of ecology and systematics **4** (1), S. 1–23.
URL: https://www.annualreviews.org/doi/pdf/10.1146/annurev.es.04.110173.000245
Zugriff am: 17.09.2022

Honerkamp 1990 Honerkamp, Josef, 1990.
Stochastische dynamische Systeme: Konzepte, numerische Methoden, Datenanalysen. Weinheim: VCH.
ISBN: 978-3-527-27945-6

Huber 2020 Huber, Marco F., 2020.
Lernen aus der Black Box: Cognitive Deep Learning soll neuronale Netze und Wissensverarbeitung kombinieren.
Physik Journal **19** (4)

Ivanov & Dolgui 2020 Ivanov, Dmitry; Dolgui, Alexandre, 2020.
Viability of intertwined supply networks: extending the supply chain resilience angles towards survivability. A position paper motivated by COVID-19 outbreak.
International Journal of Production Research **58** (10), S. 2904–2915.
DOI: 10.1080/00207543.2020.1750727

Jeschke 2015 Jeschke, Sabina, 2015.
Kybernetik und die Intelligenz verteilter Systeme: Nordrhein-Westfahlen auf dem Weg zum digitalen Industrieland. Wuppertal.
DOI: 10.1007/978-3-658-11755-9_14

Jones & Sbarbaro 1994 Jones, Richard W.; Sbarbaro, Daniel G., 1994.
Process Design and Adaptive Robust Control for Partially Known Systems.
IFAC Proceedings Volumes **27** (7), S. 239–244.
DOI: 10.1016/S1474-6670(17)47993-6

Kaashoek 1990 Kaashoek, Marinus A., 1990.
Robust Control of Linear Systems and Nonlinear Control: Proceedings of the International Symposium MTNS-89, Volume II. Boston, MA: Birkhäuser. 4.
ISBN: 978-1-461-24484-4

Kelleher et al. 2015 Kelleher, John D.; MacNamee, Brian; D'Arcy, Aoife, 2015.
Fundamentals of machine learning for predictive data analytics: Algorithms, worked examples, and case studies. Cambridge, Massachusetts und London, England: The MIT Press.
ISBN: 978-0-262-02944-5

Kesselring 1954 Kesselring, Fritz, 1954.
Technische Kompositionslehre: Anleitung zu technisch-wirtschaftlichem und verantwortungsbewußtem Schaffen. Berlin, Heidelberg: Springer.
ISBN: 978-3-642-92625-9

Kolodner 1992 Kolodner, Janet L., 1992.
An introduction to case-based reasoning.
Artificial Intelligence Review **6** (1), S. 3–34.
DOI: 10.1007/BF00155578

König et al. 2019 König, Ulrich Matthias; Röglinger, Maximilian; Urbach, Nils, 2019.
Industrie 4.0 in kleinen und mittleren Unternehmen – Welche Potenziale lassen sich mit smarten Geräten in der Produktion heben?
HMD Praxis der Wirtschaftsinformatik **56** (6), S. 1233–1249.
DOI: 10.1365/s40702-019-00567-w

Körbel 2019 Körbel, Anabelle, 2019.
Unsere Stimme verrät unsere Stimmung,
URL: https://www.brandeins.de/magazine/brand-eins-wirtschaftsmagazin/2019/gefuehle/unsere-stimme-verraet-unsere-stimmung
Zugriff am: 10.06.2019

Kubicek 1977 Kubicek, Herbert, 1977.
Heuristische Bezugsrahmen und heuristisch angelegte Forschungsdesigns als Elemente einer Konstruktionsstrategie empirischer Forschung.
In: Köhler, Richard (Hrsg.): *Empirische und handlungstheoretische Forschungskonzeptionen in der Betriebswirtschaftslehre*.
Stuttgart: Poeschel, S. 3–36.
ISBN: 978-3-791-00214-9

Kuhn & Johnson 2016 Kuhn, Max; Johnson, Kjell, 2016.
Applied predictive modeling. Corrected at 5th printing. New York: Springer.
ISBN: 978-1-461-46848-6

Kuhn 1970 Kuhn, Thomas S., 1970.
The structure of scientific revolutions. 2. ed., enlarged. Chicago: The Univ. of Chicago Press. 411.
ISBN: 978-0-226-45803-8

Landherr 2014 Landherr, Martin Hubert, 2014.
Integrierte Produkt- und Montagekonfiguration für die variantenreiche Serienfertigung. Stuttgart: Fraunhofer-Verl.
Stuttgarter Beiträge zur Produktionsforschung 39.
Univ., Diss., 2014.
ISBN: 978-3-839-60809-8

Law & Kelton 1991 Law, Averill M.; Kelton, W. David, 1991.
Simulation modeling and analysis. Second editon. New York u. a.: McGraw-Hill Inc.
ISBN: 978-0-070-36698-5

Leiner 1982 Leiner, Bernd, 1982.
Einführung in die Zeitreihenanalyse. München: Oldenbourg.
ISBN: 978-3-486-26671-9

Lem 2003 Lem, Stanislaw, 2003.
Die Megabit-Bombe: Essays zum Hyperspace. 1. Aufl.
Hannover: Heise. Telepolis.
ISBN: 978-3-936-93100-6

Lengnick-Hall & Beck 2005 Lengnick-Hall, Cynthia A.; Beck, Tammy E., 2005.
Adaptive Fit Versus Robust Transformation: How Organizations Respond to Environmental Change.
Journal of Management **31** (5), S. 738–757.
DOI: 10.1177/0149206305279367

Lewis & Stewart 2003 Lewis, Gerard J.; Stewart, Neil, 2003.
The measurement of environmental performance: an application of Ashby's law.
Systems Research and Behavioral Science **20** (1), S. 31–52.
DOI: 10.1002/sres.524

Lingitz et al. 2018 Lingitz, Lukas; Gallina, Viola; Ansari, Fazel; Gyulai, Dávid; Pfeiffer, András; Sihn, Wilfried; Monostori, László, 2018. Lead time prediction using machine learning algorithms: A case study by a semiconductor manufacturer. *Procedia CIRP* **72**, S. 1051–1056. DOI: 10.1016/j.procir.2018.03.148

Lödding 2016 Lödding, Hermann, 2016. *Verfahren der Fertigungssteuerung: Grundlagen, Beschreibung, Konfiguration*. 3. Aufl. 2016. Berlin, Heidelberg: Springer Berlin Heidelberg. VDI-Buch. ISBN: 978-3-662-48458-6

Luhmann 2002 Luhmann, Niklas, 2002. *Soziale Systeme: Grundriß einer allgemeinen Theorie*. 1. Aufl., 10. [Nachdr.] Frankfurt am Main: Suhrkamp. 666. ISBN: 978-3-518-28266-2

Lutz 2014 Lutz, Mark, 2014. *Python - kurz & gut: Für Python 3.x und 2.7*. 5. Auflage. Beijing u. a.: O'Reilly. ISBN: 978-3-955-61770-7

Mack et al. 2015 Mack, Oliver; Khare, Anshuman; Kramer, Andreas, 2015. *Managing in a VUCA World*. 1. Aufl. Springer-Verlag. ISBN: 978-3-319-16888-3

Malik 2015 Malik, Fredmund, 2015. *Strategie des Managements komplexer Systeme: Ein Beitrag zur Management-Kybernetik evolutionärer Systeme*. 11. Auflage. Bern: Haupt Verlag. ISBN: 978-3-258-07918-9

Mariscal et al. 2010 Mariscal, Gonzalo; Marban, Oscar; Fernandez, Covadonga, 2010. A survey of data mining and knowledge discovery process models and methodologies. *The Knowledge Engineering Review* **25** (2), S. 137–166

Marr 2015 Marr, Bernard, 2015. *Big data: Using smart big data, analytics and metrics to make better decisions and improve performance*. 1. publ. Chichester, West Sussex, United Kingdom: John Wiley and Sons Inc. ISBN: 978-1-118-96583-2

Marz 2016 Marz, Nathan, 2016. *Big Data: Entwicklung und Programmierung von Systemen für große Datenmengen und Einsatz der Lambda-Architektur*. 1st ed. Frechen: MITP. ISBN: 978-3-958-45177-3

Mattsson et al. 2011 Mattsson, Sandra; Gullander, Per; Davidsson, Anna, 2011. Method for Measuring Production Complexity. In: *28th International Manufacturing Conference*

Mazarov et al. 2020 Mazarov, Jürgen; Schmitt, Jacqueline; Deuse, Jochen; Richter, Ralf; Kühnast-Benedikt, Robin; Biedermann, Hubert, 2020.
Visualisierung in Industrial-Data-Science-Projekten: Nutzen grafischer Darstellung von Informationen und Daten in Industrial-Data-Science-Projekten.
Industrie 4.0 Management (6), S. 63–66

Metan et al. 2010 Metan, Gokhan; Sabuncuoglu, Ihsan; Pierreval, Henri, 2010.
Real time selection of scheduling rules and knowledge extraction via dynamically controlled data mining.
International Journal of Production Research **48** (23), S. 6909–6938.
DOI: 10.1080/00207540903307581

Meyer et al. 2019 Meyer, Bernd E.; Stübel, Günter; Schneider, Hans J., 2019.
Computergestützte Unternehmensplanung: Eine Planungsmethodologie mit Planungsinstrumentarium für das Management. Reprint 2019. Berlin und Boston: De Gruyter.
DOI: 10.1515/9783110867022

Miehe 2018 Miehe, Robert, 2018.
Methodik zur Quantifizierung der nachhaltigen Wertschöpfung von Produktionssystemen an der ökonomisch-ökologischen Schnittstelle anhand ausgewählter Umweltprobleme. Stuttgart. Stuttgarter Beiträge zur Produktionsforschung Band 76. Univ., Diss., 2017.
ISBN: 978-3-839-61291-0

Mintzberg 1994 Mintzberg, Henry, 1994.
That's Not "Turbulence", Chicken Little, It's Really Opportunity.
Planning Review **22** (6), S. 7–9.
DOI: 10.1108/eb054485

Mockenhaupt 2021 Mockenhaupt, Andreas, 2021.
Datengetriebene Prozessanalyse.
In: Mockenhaupt, Andreas (Hrsg.): *Digitalisierung und Künstliche Intelligenz in der Produktion*.
Wiesbaden und Heidelberg: Springer Vieweg, S. 271–284.
ISBN: 978-3-658-32773-6

Monostori 2002 Monostori, László, 2002.
AI and Machine Learning Techniques for Managing Complexity, Changes and Uncertainties in Manufacturing.
IFAC Proceedings Volumes **35** (1), S. 119–130.
DOI: 10.3182/20020721-6-ES-1901.01644

Murphy et al. 2019 Murphy, Rory; Newell, Anthony; Hargaden, Vincent; Papakostas, Nikolaos, 2019.
Machine learning technologies for order flowtime estimation in manufacturing systems.
Procedia CIRP **81**, S. 701–706.
DOI: 10.1016/j.procir.2019.03.179

Netland & Ferdows 2016 Netland, Torbjørn H.; Ferdows, Kasra, 2016. The S-Curve Effect of Lean Implementation. *Production and Operations Management* **25** (6), S. 1106–1120. DOI: 10.1111/poms.12539

Nusser et al. 2009 Nusser, Sebastian; Otte, Clemens; Hauptmann, Werner; Leirich, Oskar; Krätschmer, Manfred; Kruse, Rudolf, 2009. Maschinelles Lernen von validierbaren Klassifikatoren zur autonomen Steuerung sicherheitsrelevanter SystemeMachine Learning of Verifiable Classifiers for Autonomous Control of Safety-related Systems. *at - Automatisierungstechnik* **57** (3), S. 138–145. DOI: 10.1524/auto.2009.0761

Nyhuis & Wiendahl 2012 Nyhuis, Peter; Wiendahl, Hans-Peter, 2012. *Logistische Kennlinien: Grundlagen, Werkzeuge und Anwendungen*. 3. Aufl. 2012. Berlin, Heidelberg: Springer. ISBN: 978-3-540-92838-6

Ohl 2000 Ohl, Stefan, 2000. *Prognose und Planung variantenreicher Produkte am Beispiel der Automobilindustrie*. Düsseldorf, Karlsruhe: VDI-Verl. 120. Univ., Diss., 2000. ISBN: 978-3-183-12016-1

Onan et al. 2016 Onan, Aytuğ; Korukoğlu, Serdar; Bulut, Hasan, 2016. Ensemble of keyword extraction methods and classifiers in text classification. *Expert Systems with Applications* **57** (8), S. 232–247. DOI: 10.1016/j.eswa.2016.03.045

Ouyang et al. 2022 Ouyang, Long; Wu, Jeff; Jiang, Xu; Almeida, Diogo; Wainwright, Carroll L.; Mishkin, Pamela; Zhang, Chong; Agarwal, Sandhini; Slama, Katarina; Ray, Alex; Schulman, John; Hilton, Jacob; Kelton, Fraser; Miller, Luke; Simens, Maddie; Askell, Amanda; Welinder, Peter; Christiano, Paul; Leike, Jan; Lowe, Ryan, 2022. *Training language models to follow instructions with human feedback*. URN: 2203.02155v1

Öztürk et al. 2006 Öztürk, Atakan; Kayalıgil, Sinan; Özdemirel, Nur E., 2006. Manufacturing lead time estimation using data mining. *European Journal of Operational Research* **173** (2), S. 683–700. DOI: 10.1016/j.ejor.2005.03.015

Papp et al. 2019 Papp, Stefan; Weidinger, Wolfgang; Meir-Huber, Mario; Ortner, Bernhard; Langs, Georg; Wazir, Rania, 2019. *Handbuch Data Science: Mit Datenanalyse und Machine Learning Wert aus Daten generieren*. München: Hanser. ISBN: 978-3-446-46040-9

Patzak 1982 Patzak, Gerold, 1982.
Systemtechnik - Planung komplexer innovativer Systeme: Grundlagen, Methoden, Techniken. Berlin: Springer.
ISBN: 978-3-540-11783-0

Pedregosa et al. 2011 Pedregosa, Fabian; Varoquaux, Gaël; Gramfort, Alexandre; Michel, Vincent; Thirion, Bertrand; Grisel, Olivier; Blondel, Mathieu; Müller, Andreas; Nothman, Joel; Louppe, Gilles; Prettenhofer, Peter; Weiss, Ron; Dubourg, Vincent; Vanderplas, Jake; Passos, Alexandre; Cournapeau, David; Brucher, Matthieu; Perrot, Matthieu; Duchesnay, Édouard, 2011.
Scikit-learn: Machine Learning in Python.
Journal of Machine Learning Research **12**, S. 2825–2830.
URN: 1201.0490v4

Petersdorff 2019 Petersdorff, Thomas, 2019.
Innovation made in Germany,
URL: https://www.behoerden-spiegel.de/2019/11/25/innovation-made-in-germany/
Zugriff am: 03.10.2021

Pfeiffer et al. 2016 Pfeiffer, András; Gyulai, Dávid; Kádár, Botond; Monostori, László, 2016.
Manufacturing Lead Time Estimation with the Combination of Simulation and Statistical Learning Methods.
Procedia CIRP **41**, S. 75–80.
DOI: 10.1016/j.procir.2015.12.018

Platon 1988 Platon, 1988.
Sämtliche Dialoge. [Vollst. und unveränd. Nachdr.] Hamburg: Meiner.
ISBN: 978-3787309207

Ponomarov & Holcomb 2009 Ponomarov, Serhiy Y.; Holcomb, Mary C., 2009.
Understanding the concept of supply chain resilience.
The International Journal of Logistics Management **20** (1), S. 124–143.
DOI: 10.1108/09574090910954873

Ponsard et al. 2017 Ponsard, Christophe; Touzani, Mounir; Majchrowski, Annick, 2017.
Combining Process Guidance and Industrial Feedback for Successfully Deploying Big Data Projects.
Open Journal of Big Data (OJBD) **3**, S. 26–41.
URL: https://www.researchgate.net/publication/321944704_Combining_Process_Guidance_and_Industrial_Feedback_for_Successfully_Deploying_Big_Data_Projects
Zugriff am: 26.04.2022

Popper 1935 Popper, Karl, 1935.
Logik der Forschung: Zur Erkenntnistheorie der Modernen Naturwissenschaft. Vienna: Springer.
ISBN: 978-3-709-12021-7

Preißler 2008 Preißler, Peter R., 2008.
Betriebswirtschaftliche Kennzahlen: Formeln, Aussagekraft, Sollwerte, Ermittlungsintervalle. München und Wien: Oldenbourg.
ISBN: 978-3-486-23888-4

Püschel et al. 2017 Püschel, Louis; Röglinger, Maximilian; Schlott, Helen, 2017.
Smart Things im Internet der Dinge — ein Klassifikationsansatz.
Wirtschaftsinformatik & Management **9** (2), S. 54–61.
DOI: 10.1007/s35764-017-0042-1

Quinlan 1993 Quinlan, John Ross, 1993.
C4.5 - programs for machine learning: Programs for Machine Learning. 1. Aufl. San Mateo, Calif.: Kaufmann. The Morgan Kaufmann series in machine learning.
ISBN: 978-1-558-60238-0

Rammer 2018 Rammer, Christian, 2018.
Produktivitätsparadoxon im Maschinenbau. Mannheim und Karlsruhe: VDMA Verlag GmbH

Rapp 1973 Rapp, Friedrich, 1973.
Technik und Naturwissenschaften - eine methodologische Untersuchung.
In: Lenk, Hans; Moser, Simon (Hrsg.): *Techne, Technik, Technologie*.
Pullach bei München: Verl. Dokumentation, S. 108–132.
ISBN: 978-3-794-02622-7

Reiß 1992 Reiß, Michael, 1992.
Optimale Unternehmenskomplexität: Schlüsselgröße für exzellente Unternehmensführung.
Personal (9), S. 414–418.
URL:
https://elib.uni-stuttgart.de/bitstream/11682/5604/1/rei71.pdf
Zugriff am: 14.03.2022

Röhrig 2020 Röhrig, Katharina, 2020.
Auf dem Weg in die 5. industrielle Revolution: Künstliche Intelligenz und MES.
IT&Production (11), S. 26–28

Ropohl 2009 Ropohl, Günter, 2009.
Allgemeine Technologie: Eine Systemtheorie der Technik. 3., überarbeitete Auflage. Karlsruhe: Universitätsverlag Karlsruhe.
ISBN: 978-3-866-44374-7

Ropohl 2012 Ropohl, Günter, 2012.
Allgemeine Systemtheorie: Einführung in transdisziplinäres Denken. Berlin: Edition Sigma.
ISBN: 978-3-836-03586-6

Ruediger-Flore et al. 2008 Ruediger-Flore, Patrick; Glatt, Moritz; Aurich, Jan C., 2008.
Maschinelles Lernen bei hohem Variantenreichtum und kleinen Serien.
In: Schuh, Günther; Stölzle, Wolfgang; Straube, Frank (Hrsg.): *Anlaufmanagement in der Automobilindustrie erfolgreich umsetzen*.
Berlin, Heidelberg: Springer, S. 538–543.
ISBN: 978-3-540-78406-7

Samulat 2017 Samulat, Peter, 2017.
Die Digitalisierung der Welt. Wiesbaden: Springer Fachmedien.
ISBN: 978-3-658-15510-0

Sandler dos Passos et al. 2018 Sandler dos Passos, Danielle; Coelho, Helder; Mori Sarti, Flavia, 2018.
From Resilience to the Design of Antifragility.
In: Hübner, Michael (Hrsg.): *Pesaro 2018*.
Wilmington, DE, USA: IARIA, S. 7–11.
ISBN: 978-1-612-08628-6

Schank & Leake 1989 Schank, Roger C.; Leake, David B., 1989.
Creativity and learning in a case-based explainer.
Artificial Intelligence **40** (1), S. 353–385.
DOI: 10.1016/0004-3702(89)90053-2

Schapire 1990 Schapire, Robert E., 1990.
The strength of weak learnability.
Machine Learning **5** (2), S. 197–227.
DOI: 10.1007/BF00116037

Schickmair 2010 Schickmair, Martin, 2010.
Optimaler Betriebspunkt für die Sachgüterproduktion: ... bei divergenten technischen und betriebswirtschaftlichen Unternehmenszielen. Saarbrücken: VDM Verlag Dr. Müller.
ISBN: 978-3-639-29832-1

Schiemenz 1993 Schiemenz, Bernd, 1993.
Betriebswirtschaftliche Systemtheorie.
In: Kern, Werner (Hrsg.): *Handwörterbuch der Produktionswirtschaft*.
Stuttgart: Schäffer-Poeschel, S. 4127–4140.
ISBN: 978-3-791-08040-6

Schlittgen & Streitberg 2001 Schlittgen, Rainer; Streitberg, Bernd H. J., 2001.
Zeitreihenanalyse. 9., unwesentlich veränd. Aufl. München und Wien: Oldenbourg.
ISBN: 978-3-486-25725-0

Schmitt 2021 Schmitt, Alexander, 2021. *Parametrierte Modellierung eines Produktionssystems zur Erzeugung synthetischer Daten*. Stuttgart. Univ., BA, 2021

Schomburg 1980 Schomburg, Eckart, 1980. *Entwicklung eines betriebstypologischen Instrumentariums zur systematischen Ermittlung der Anforderungen an EDV-gestützte Produktionsplanungs- und -steuerungssysteme im Maschinenbau*. Aachen. Techn. Hochsch., Diss., 1980

Schönsleben 2004 Schönsleben, Paul, 2004. *Integrales Logistikmanagement: Planung und Steuerung der umfassenden Supply Chain*. 4., überarb. und erw. Aufl. Berlin und Heidelberg: Springer. ISBN: 978-3-540-21177-2

Schroda 2000 Schroda, Frauke, 2000. *Über das Ende wird am Anfang entschieden: Zur Analyse der Anforderungen von Konstruktionsaufträgen*. Berlin: Technische Universität Berlin. Univ., Diss., 2000. DOI: 10.14279/depositonce-248

Schuh 1997 Schuh, Günther, 1997. *Komplexität und Agilität: Steckt die Produktion in der Sackgasse?* Berlin und Heidelberg: Springer. ISBN: 978-3-642-64579-2

Schuh 2006 Schuh, Günther, 2006. *Produktionsplanung und -steuerung: Grundlagen, Gestaltung und Konzepte*. 3., völlig neu bearb. Aufl. Berlin und Heidelberg: Springer. ISBN: 978-3-540-40306-7

Schuh & Stich 2012 Schuh, Günther; Stich, Volker, 2012. *Produktionsplanung und -steuerung 1: Grundlagen der PPS*. 4. Aufl. Berlin: Springer. SpringerLink Bücher. ISBN: 978-3-642-25422-2

Schuh & Schmidt 2014 Schuh, Günther; Schmidt, Carsten, 2014. *Produktionsmanagement: Handbuch Produktion und Management 5*. 2., vollst. neu bearb. und erw. Aufl. Berlin: Springer Vieweg. 5. ISBN: 978-3-642-54287-9

Schuh & Fuß 2015 Schuh, Günther; Fuß, Christine, 2015. *ProSense: Ergebnisbericht des BMBF-Verbundprojektes ; hochauflösende Produktionssteuerung auf Basis kybernetischer Unterstützungssysteme und intelligenter Sensorik*. 1. Aufl. Aachen: Apprimus Verl. ISBN: 978-3-86359-346-9

Schuh & Riesener 2018 Schuh, Günther; Riesener, Michael, 2018. *Produktkomplexität managen: Strategien - Methoden - Tools*. 3., vollständig überarbeitete Auflage. München: Hanser. ISBN: 978-3-446-45225-1

Schwaninger 1996 Schwaninger, Markus, 1996. Systemtheorie. In: Kern, Werner (Hrsg.): *Handwörterbuch der Produktionswirtschaft*. Stuttgart: Schäffer-Poeschel, S. 1946–1960. ISBN: 978-3-791-08044-4

Schwaninger 2006 Schwaninger, Markus, 2006. System dynamics and the evolution of the systems movement. *Systems Research and Behavioral Science* **23** (5), S. 583–594. DOI: 10.1002/sres.800

Schweitzer 1978 Schweitzer, Marcell, 1978. *Auffassungen und Wissenschaftsziele der Betriebswirtschaftslehre*. Darmstadt: Wissenschaftl. Buchges. 502. ISBN: 978-3-534-07160-9

Shalev-Shwartz & Ben-David 2014 Shalev-Shwartz, Shai; Ben-David, Shai, 2014. *Understanding machine learning: From theory to algorithms*. Cambridge u. a.: Cambrige University Press. ISBN: 978-1-107-05713-5

Shannon 2001 Shannon, Claude E., 2001. A mathematical theory of communication. *SIGMOBILE Mob. Comput. Commun. Rev.* **5** (1), S. 3–55. DOI: 10.1145/584091.584093

Silva et al. 2011 Silva, João; Coheur, Luísa; Mendes, Ana Cristina; Wichert, Andreas, 2011. From symbolic to sub-symbolic information in question classification. *Artificial Intelligence Review* **35** (2), S. 137–154. DOI: 10.1007/s10462-010-9188-4

Simon 1969 Simon, Herbert A., 1969. The architecture of complexity. *Proceedings of the American Philosophical Society 106 (6)*, S. 467–482. URL: https://www.cc.gatech.edu/classes/AY2013/cs7601_spring/papers/Simon-Complexity.pdf Zugriff am: 14.12.2022

Solow 1956 Solow, Robert M., 1956. A Contribution to the Theory of Economic Growth. *The Quarterly Journal of Economics* **70** (1), S. 65. DOI: 10.2307/1884513

Sovrano et al. 2021 Sovrano, Francesco; Vitali, Fabio; Palmirani, Monica, 2021. *Making Things Explainable vs Explaining: Requirements and Challenges under the GDPR*, URL: http://arxiv.org/pdf/2110.00758v1

Stachowiak 1973 Stachowiak, Herbert, 1973. *Allgemeine Modelltheorie*. Wien: Springer. ISBN: 978-3-211-81106-1

Statistisches Bundesamt 2008 Statistisches Bundesamt, 2008. *Klassifikation der Wirtschaftszweige: Mit Erläuterungen*. Wiesbaden, URL: https://www.destatis.de/static/DE/dokumente/klassifikation-wz-2008-3100100089004.pdf

Stokes 1997 Stokes, Donald E., 1997. *Pasteur's quadrant: Basic science and technological innovation*. Washington, D.C.: Brookings Institution Press. ISBN: 978-0-815-78177-6

Straubhaar 2021 Straubhaar, Thomas, 2021. Neue Datenökonomik für neue Datenökonomie?: Postcoronomics: was ist neu? *Industrie 4.0 Management* **37** (1), S. 55–58

Thoma et al. 2016 Thoma, Klaus; Scharte, Benjamin; Hiller, Daniel; Leismann, Tobias, 2016. Resilience Engineering as Part of Security Research: Definitions, Concepts and Science Approaches. *European Journal for Security Research* **1** (1), S. 3–19. DOI: 10.1007/s41125-016-0002-4

Töpfer 2010 Töpfer, Armin, 2010. *Erfolgreich Forschen: Ein Leitfaden für Bachelor-, Master-Studierende und Doktoranden*. 2., überarb. und erw. Aufl. Berlin, Heidelberg: Springer. ISBN: 978-3-642-13901-7

Treml 1994 Treml, Alfred K., 1994. Über die Unwissenheit. *Zeitschrift für Pädagogik* **40** (40-4), S. 529–537. DOI: 10.25656/01:10848

Ulrich 2001 Ulrich, Hans, 2001. Die Unternehmung als produktives soziales System. In: Ulrich, Hans; Schwaninger, Markus (Hrsg.): *Systemorientiertes Management*. Bern, Stuttgart und Wien: Haupt. ISBN: 978-3-258-06359-1

Ulrich & Schwaninger 2001 Ulrich, Hans; Schwaninger, Markus (Hrsg.), 2001. *Systemorientiertes Management: Das Werk von Hans Ulrich*. Studienausg. Bern, Stuttgart und Wien: Haupt. ISBN: 978-3-258-06359-1

Ulrich & Hill 1976 Ulrich, Peter; Hill, Wilhelm, 1976. Ulrich1976_Wissenschaftstheoretische Grundlagen der BWL: Wissenschaftstheoretische Grundlagen der Betriebswirtschaftslehre (Teil I). *WiST Zeitung für Ausbildung und Hochschulkontakt* (7), S. 304–309

Usuga Cadavid et al. 2020 Usuga Cadavid, Juan Pablo; Lamouri, Samir; Grabot, Bernard; Pellerin, Robert; Fortin, Arnaud, 2020. Machine learning applied in production planning and control: a state-of-the-art in the era of industry 4.0. *Journal of Intelligent Manufacturing* **31** (6), S. 1531–1558. DOI: 10.1007/s10845-019-01531-7

van Rossum 2021 van Rossum, Guido, 2021. *Python*, URL: https://www.python.org/ Zugriff am: 03.06.2021

VDI 1993 VDI, Verband Deutscher Ingenieure, 1993. Richtlinie 2221. Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte. *DK62.001/.002:621:68.:66.0*

VDI 2004 VDI, Verband Deutscher Ingenieure, 2004. Richtlinie 2223. Methodisches Entwerfen technischer Produkte. *ICS 03.100.40*

VDI 2017 VDI, Verband Deutscher Ingenieure, 2017. Richtlinie 5201. Beschreibung und Messung der Wandlungsfähigkeit produzierender Unternehmen: Beispiel Medizintechnik. *ICS 03.100.50, 11.020.99*

VDI 2018 VDI, Verband Deutscher Ingenieure, 2018. Richtlinie 3633. Simulation von Logistik-, Materialfluss- und Produktionssystemen. *ICS 01.040.03, 03.100.10*

VDMA 2020 VDMA, 2020. *Statistisches Handbuch für den Maschinenbau 2020: Ausgabe 2021*. 1. Auflage. Frankfurt am Main: VDMA. ISBN: 978-3-8163-0742-8

Wahrig-Burfeind 2011 Wahrig-Burfeind, Renate, 2011. *Brockhaus, Wahrig, Deutsches Wörterbuch: Mit einem Lexikon der Sprachlehre*. 9., vollst. neu bearb. und aktualisierte Aufl., Neuausg. Gütersloh und München: Wissenmedia. Brockhaus 11. ISBN: 978-3-577-07595-4

Watson 1999 Watson, Ian, 1999.
Case-Based Reasoning is a Methodology not a Technology.
In: Miles, Roger; Moulton, Michael; Bramer, Max (Hrsg.): *Research and Development in Expert Systems XV: Proceedings of ES98, the Eighteenth Annual International Conference of the British Computer Society Specialist Group on Expert Systems, Cambridge, December 1998*,
S. 213–223.
ISBN: 978-1-447-10835-1

Weaver 1991 Weaver, Warren, 1991.
Science and complexity.
In: Klir, George J. (Hrsg.): *Facets of Systems Science*.
Boston, MA: Springer, S. 449–456.
ISBN: 978-1-489-90718-9

Westkämper 2006 Westkämper, Engelbert, 2006.
Einführung in die Organisation der Produktion. Berlin und Heidelberg: Springer.
ISBN: 978-3-540-26039-4

Westkämper & Zahn 2009 Westkämper, Engelbert; Zahn, Erich, 2009.
Wandlungsfähige Produktionsunternehmen: Das Stuttgarter Unternehmensmodell. Berlin, Heidelberg: Springer-Verlag.
ISBN: 978-3-540-21889-0

Westkämper & Löffler 2016 Westkämper, Engelbert; Löffler, Carina, 2016.
Strategien der Produktion: Technologien, Konzepte und Wege in die Praxis. Berlin, Heidelberg: Springer.
ISBN: 978-3-662-48913-0

Wiendahl et al. 2000 Wiendahl, Hans-Hermann; Rempp, Burkhard; Schanz, Michael, 2000.
Turbulenzen erschweren die Planungssicherheit: Analogien aus der Physik erklären das Entstehen und Beherrschen von Turbulenzen.
In: *io management*.
Zürich, Schweiz: Handelszeitung Fachverlag, S. 38–43

Wiendahl 2007 Wiendahl, Hans-Hermann, 2007.
Turbulence Germs and their Impact on Planning and Control – Root Causes and Solutions for PPC Design.
CIRP Annals **56** (1), S. 443–446.
DOI: 10.1016/j.cirp.2007.05.106

Wiendahl 2011 Wiendahl, Hans-Hermann, 2011.
Auftragsmanagement der industriellen Produktion: Grundlagen, Konfiguration, Einführung. 2011. Aufl. Heidelberg: Springer.
ISBN: 978-3-642-19148-0

Wiener 1961 Wiener, Norbert, 1961.
Cybernetics: Or control and communication in the animal and the machine. Second edition. Cambridge, Massachusetts: M.I.T. Press.
ISBN: 978-0-262-73009-9

Wirth & Hipp 2000 Wirth, Rüdiger; Hipp, Jochen, 2000.
CRISP-DM: Towards a standard process model for data mining. In: Mackin, Neil (Hrsg.): *Proceedings of the Fourth International Conference on the Practical Application of Knowledge Discovery and Data Mining: 11th - 13th April 2000, Crowne Plaza Midland Hotel, Manchester, UK*,
S. 29–39.
ISBN: 978-1-902-42608-2

Wöhe et al. 2020 Wöhe, Günter; Döring, Ulrich; Brösel, Gerrit, 2020.
Einführung in die allgemeine Betriebswirtschaftslehre. 27., überarbeitete und aktualisierte Auflage. München: Verlag Franz Vahlen.
ISBN: 978-3-800-66300-2

Wolf et al. 2019 Wolf, Patrick; Deuse, Jochen; Richter, Ralph, 2019.
Einfluss und Ursachen von Variabilität in der kunden-auftragsspezifischen Produktion.
ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb **114** (11), S. 730–733.
DOI: 10.3139/104.112184

Wöstmann et al. 2017 Wöstmann, Rene; Nöhring, Fabian; Deuse, Jochen; Klinkenberg, Ralf; Lacker, Thomas, 2017.
Big Data Analytics in der Auftragsabwicklung. Erschließung ungenutzter Potenziale in der variantenreichen Kleinserienfertigung.
Industrie 4.0 Management - Gegenwart und Zukunft industrieller Geschäftsprozesse **33** (4), S. 7–11

Wu et al. 2007 Wu, Xindong; Kumar, Vipin; Ross Quinlan, J.; Ghosh, Joydeep; Yang, Qiang; Motoda, Hiroshi; McLachlan, Geoffrey J.; Ng, Angus; Liu, Bing; Yu, Philip S.; Zhou, Zhi-Hua; Steinbach, Michael; Hand, David J.; Steinberg, Dan, 2007.
Top 10 algorithms in data mining.
In: *Knowledge and Information Systems*.
Springer, S. 1–37.
DOI: 10.1007/s10115-007-0114-2

Zürn 2010 Zürn, Michael, 2010.
Referenzmodell für die Fabrikplanung auf Basis von Quality Gates. Heimsheim: Jost-Jetter. IPA-IAO Forschung und Praxis 497.
Univ., Diss., 2010.
ISBN: 978-3-939-89061-4

A Anhang

A.1 Tabellen

Tabelle A.1: Komponenten der Resilienz in Anlehnung an Ponomarov et al.

Elastizität	Die Schnelligkeit der Wiederherstellung eines stabilen Gleichgewichtszustands nach einer Störung
Amplitude	Der Bereich der störbedingten Zustandsabweichung, aus dem das System in seinen Ausgangszustand zurückkehrt
Hysterese	Der Grad der Spiegelbildlichkeit des Abweichungspfades und des Erholungspfades nach einer längeren, chronischen Zustandsabweichung
Verformbarkeit	Das Ausmaß, in dem sich der nach der Störung hergestellte stationäre Zustand vom ursprünglichen Zustand unterscheidet
Dämpfung	Das Ausmaß, in dem der Weg zur Wiederherstellung des Ausgangszustands durch äußere Kräfte verändert wird

A.2 Formeln und Gleichungen

Empirische Momente

Das *arithmetische Mittel* $\bar{x}$ gibt die zentrale Lage einer Zeitreihe an und kann ungewichtet oder gewichtet ($\bar{x}_w$) wie in den Gleichungen A.1 und A.2 (mit $t = T$ und den Gewichten $w_t \geq 0$) berechnet werden:

$$\bar{x} = \frac{1}{N} \sum_{t=1}^{N} x_t \tag{A.1}$$

$$\bar{x}_w = \frac{\sum_{t=1}^{N} w_t \cdot x_t}{\sum_{t=1}^{N} w_t} \tag{A.2}$$

Geometrisches und gewichtetes geometrisches Mittel:

$$\bar{x}_{geom} = \sqrt[N]{\prod_{t=1}^{N} x_t} = \sqrt[N]{x_1 \cdot x_2 \ldots x_N} \tag{A.3}$$

$$\bar{x}_{geom\_w} = \left(\prod_{t=1}^{N} x_t^{w_t} \right)^{\frac{1}{w}} = \sqrt[w]{\prod_{t=1}^{N} x_t^{w_t}}, w := \sum_{t=1}^{N} w_t \tag{A.4}$$

Weitere Zentralwerte sind Median und Modus. Der *Median* x_{med} ist definiert als die Merkmalsausprägung desjenigen Wertes, der die nach Größe geordnete Zeitreihe halbiert. Der *Modus* D ist definiert als der am häufigsten in der Zeitreihe vorkommende Wert. Bei fein abgetasteten Zeitreihen ist dieses Lagemaß jedoch wenig aussagekräftig, weshalb er hier lediglich zur Vollständigkeit genannt wird.

Unter den empirischen Momenten finden sich auch Streuungsmaße[1] (Holland & Scharnbacher 2010, S. 56). Sie beschreiben die Abweichung von den oben beschriebenen Lagemaßen. Die *Spannweite* ist die Differenz zwischen größter und kleinster Merkmalsausprägung der Zeitreihe. Die *empirische Varianz* σ^2 (Gleichung A.5) beschreibt die Stärke der Schwankung der Zeitreihenwerte oder die mittlere absolute Abweichung der Zeitreihenwerte vom arithmetischen Mittel. Die Wurzel der Varianz wird *Standardabweichung* σ genannt (Gleichung A.6).

$$\sigma^2 = \frac{1}{N} \sum_{t=1}^{N} (x_t - \bar{x})^2 \tag{A.5}$$

$$\sigma = \sqrt{\sigma^2} = \sqrt{\frac{1}{N} \sum_{t=1}^{N} (x_t - \bar{x})^2} \tag{A.6}$$

Dem Vergleich von Streuungen unterschiedlicher Zeitreihen mit sich signifikant unterscheidenden arithmetischen Mittelwerten dient der *Variationskoeffizient* v (Gleichung A.7).

$$v = \frac{\sigma}{\bar{x}} \tag{A.7}$$

Komponenten von Zeitreihen:

$$x_t = g_t + s_t + u_t \tag{A.8}$$

$$x_t = g_t \cdot s_t \cdot u_t \tag{A.9}$$

$$\log x_t = \log(g_t) + \log(s_t) + \log(u_t) \tag{A.10}$$

[1] Streuungsmaße und empirische Momente überschneiden sich in Teilen. Die genaue Zuordnung ist hier nicht weiter relevant, weshalb darauf verzichtet wird.

Zeitreihen als kontinuierliche Funktion:

$$x(t) = a_0 + a_1 \cdot t \tag{A.11}$$

$$x(t) = a_0 + a_1 \cdot t^1 + ... + a_k \cdot t^k \tag{A.12}$$

$$x(t) = a \cdot e^{b \cdot t} \tag{A.13}$$

Empirische Kovarianz:

$$c = \frac{1}{N} \sum_{t=1}^{N} (x_t - \bar{x})(y_t - \bar{y}) \tag{A.14}$$

Korrelationskoeffizient:

$$r = \frac{c}{\sigma_x \cdot \sigma_y} = \frac{\frac{1}{N} \sum_{t=1}^{N} (x_t - \bar{x})(y_t - \bar{y})}{\sqrt{\frac{1}{N} \sum_{t=1}^{N} (x_t - \bar{x})^2} \cdot \sqrt{\frac{1}{N} \sum_{t=1}^{N} (y_t - \bar{y})^2}} \tag{A.15}$$

Kreuzkorrelationsfunktion:

$$\varphi_{fg}(\tau) = \int_{-\infty}^{\infty} f(t + \tau) g^*(t) dt \tag{A.16}$$

Eine komplexwertige Funktion $x(t)$ besitzt konjugierte Symmetrie, wenn gilt:

$$x(t) = x^*(-t) \tag{A.17}$$

Faltung:

$$\int_{-\infty}^{\infty} f(\tau) g(t - \tau) d\tau = f(t) * g(t) \tag{A.18}$$

Kreuzkorrelationsfunktion durch Faltungsausdruck:

$$\varphi_{fg}(\tau) = \int_{-\infty}^{\infty} f(t + \tau) g^*(t) dt = f(\tau) * g^*(-\tau) \tag{A.19}$$

Zeitquantisierung:

$$x[k] = x(kT), k \in \mathbb{Z} \tag{A.20}$$

Abtastrate:

$$f_a = \frac{1}{T} = \frac{\omega_g}{2\pi} \tag{A.21}$$

Erwartungswert (Kontinuierliches Systemmodell):

$$\mu_x = E\{x(t)\} = \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{T} x(t)dt \tag{A.22}$$

Erwartungswert erster Ordnung zu einem Zeitpunkt t:

$$E\{f(x(t))\} = \lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} f(x_i(t)) \tag{A.23}$$

Autokorrelationsfunktion (Erwartungswert zweiter Ordnung):

$$\varphi_{xx}(t_1, t_2) = E\{x(t_1) \cdot x(t_2)\} \tag{A.24}$$

Entsprechend ist die Kreuzkorrelationsfunktion (A.25) definiert als:

$$\varphi_{xy}(t_1, t_2) = E\{x(t_1) \cdot y(t_2)\} \tag{A.25}$$

Mit dem Mittelwert $E\{x(t)\}$ die Varianz $E\{(x(t) - \mu_x(t))^2\}$ gilt:

$$E\{(x(t) - \mu_x(t))^2\} = \sigma_x^2(t) \tag{A.26}$$

Standardabweichung $\sigma_x(t)$ für kontinuierliche Systeme (Gleichung A.27):

$$\sigma_x(t) = \sqrt{\sigma_x^2(t)} = \sqrt{E\{(x(t) - \mu_x(t))^2\}} \tag{A.27}$$

Autokorrelationsfunktion:

$$\varphi_{xx}(\tau) = E\{x(t)x(t-\tau)\} \tag{A.28}$$

Autokovarianzfunktion:

$$\psi_{xx}(\tau) = E\{(x(t) - \mu_x)(x(t-\tau) - \mu_x)\} = \varphi_{xx}(\tau) - \mu_x^2 \tag{A.29}$$

Kreuzkovarianzfunktion:

$$\psi_{xy}(\tau) = E\{(x(t) - \mu_x)(y(t-\tau) - \mu_y)\} = \varphi_{xy}(\tau) - \mu_x\mu_y \tag{A.30}$$

A.3 Historie zum maschinellen Lernen

Seit den ersten Computerkonzepten, wie das der ‚Analytical Engines' von Babbage and Lovelace zwischen 1830 und 1840 wurden Rechenmaschinen so erdacht, dass sie vom Mensch festgelegte Befehle ausführen. Mit der Entwicklung von der rein mechanischen Analytical Engine zu elektronischen Allzweckcomputern im Laufe der ersten Hälfte des 20. Jahrhunderts entstand durch die Pionierarbeit von Alan Turing in den 1950er Jahren die Idee, mit Hilfe der Informatik Computern menschliches Verhalten anzutrainieren (Chollet 2018, S. 22). In den folgenden zwei Jahrzehnten entstanden viele der Ansätze für heute gängige Anwendungen für ‚kNN', Chatbots und humanoide Robotik. Aus Gründen enttäuschter Erwartungen und Limitierung der damals technisch möglichen Rechenleistung folgt der „KI-Winter" (Huber 2020, S. 38). Die Forschungsaktivitäten zur KI erstreckten sich mittlerweile von der Systemtheorie über die Philosophie mit Schnittstellen zur Neurophysik bis zum Bereich der Informatik. In den 1990ern erfuhr die KI neuen Schwung mit ersten Industrieanwendungen und medienwirksamen Erfolgen wie beispielsweise dem Sieg eines Computers über dem Schachweltmeister Kasparov. Wegen der nicht immer adäquaten oder zumindest zu diskutierenden (Chollet 2018, S. 27) Wortwahl mit Bezug zu KI, kNN oder ‚Deep Learning' wird immer wieder die Analogie zum menschlichen Gehirn gezogen. Tatsächlich ist rechnerbasierter ‚Ersatz' menschlicher Sinne (engl. "Cognition"siehe (Buescher et al. 2011, S. 169)) teilweise bereits Realität. Text- und Bilderkennung haben sich etabliert. Heutige KI-Algorithmen ‚verstehen' in akustischen Signalen nicht nur den semantischen Inhalt, sondern sind auch in der Lage, die Emotionen des Sprechenden im Schallsignal zu detektieren (Körbel 2019).

A.4 Pseudo-Code

Listing A.1: Datensatzgenerierung statisches System

```
# import scikit-learn package
# define parameters:
kwarg = dict(n_samples = n_samples, [...],
        random_state = random_state)
# create dataset of features (f) and targets (t)
f, t     = make_classification(**kwarg)
features = pd.DataFrame({feature_names: f}) # features
targets  = pd.DataFrame({target_name: t})   # targets
df       = pd.concat([features, targets])
# divide into train- and testset
ix                      = random.sample(N_test)
test                    = np.zeros(len(df), dtype=bool)
test[ix], df['is_test'] = True, test
test_df                 = (df[df.is_test == True])
train_df                = (df[df.is_test != True])
```

Listing A.2: Datensatzgenerierung dynamisches System

```
N_states     = 3       # Number of states [-]
N_backswitch = 1       # Number of repeated states [-]
t_points_d   = [[0.2, 0.2], [0.5, 0.5], [0.7, 0.7]] # change ↩
   times
N_features = [34, 20, 16, 12, 8, 6, 4] # Number of features
noise_dt   = .01*np.random.random(7)   # timestamp noise
noise_dv   = .02*np.random.random(7)   # feature value noise
lat_v      = -30*24*60*60*np.ones(7)   # latency [0:30 days]
```

Listing A.3: Latenzbestimmung

```
def calc_delay(f_df, t_df): # Berechnet paarweise Latenz zwischen Feature- und Targetzeitreihen (f_df und t_df)
# Gleitender Durchschnitt (Glaettung der Ausreisser)
f_df['feat'] = f_df.feat.rolling(), center=True).mean()
t_df['targ'] = t_df.targ.rolling(), center=True).mean()
# N Abtastpunkte im Zeit-Bereich von Feature und Target
sampl       = np.linspace(min(ts), max(ts), N)
df          = pd.DataFrame(data = sampl, columns = ['ts'])
df['check'] = 'check' # Labeling der Abtast-Samples
df          = pd.concat([f_df,t_df]).sort_values(by=['ts'])
df['dti']   = df.ts.apply(lambda x: datetime(x))
df          = df.set_index('ts') # Timestamps als Index für
df          = df.interpolate()   # Lineare Interpolation

# Preprocessing: Normieren auf 0 <= Werte <= 1
# "Flipping-Vector" um normierte Zeitreihen "umzuklappen"
vec  = [[1,0,1,0],[-1,1,1,0],[1,0,-1,1],[-1,1,-1,1]]
# Berechne je Flipping eine Korr-Funktion; (signal.correlate ist performanter als np.correlate)
for v in vec:
value = signal.correlate(v[0]*ts_targ.targ+v[1],v[2]*ts_feat.feat+v[3],'full')
idx   = np.argmax(value) # Suche Index mit max Korr.
delay = dt*idx+ts_targ.ts.min()-ts_feat.ts.max()
store.append(delay)
# Plausibilisierung: Schwell- und Grenzwerte a und b
store = [x for x in range(len(store)) if
store[x] > a and store[x] < b]
delay = store[x] # Waehle Latenz mit maximaler Korr.

return np.round(delay/24/60/60,2) # Uebergebe Latenz [Tage]
```

Listing A.4: Trainingssatzreduktion

```
#Für jeden neuen Auftrag: Reduktion des Datensatzes
rfclf = RandomForestClassifier()       # Erstelle Klassierer
rfclf.fit(training_data)               # Training
prediction = rfclf.predict(next_order) # Prognose
new_val    = eval_pred(prediction)     # Prognosebewertung
# thresh: Schwellwert
if new_val < thresh * max_val:
strt         += 2*step    # Verschiebung des Startsamples
max_val       = new_val                 # Aktualisierung
training_data = full_data[strt:idx-1,:]         # Auswahl
```

Listing A.5: Tupel für Sample-Auswahl

```
# Erstellung der Tupel:
from itertools import permutations
def myperm(n):
lst, prm = [], [] # Generiere Listen mit Nullen und einsen
[lst.append(0) for i in range(n-1)]
[lst.append(1) for i in range(n-1)]
# Generiere Tupel ohne Wiederholungen
[prm.append(x) for x in permutations(lst,n) if x not in prm]
prm.append(tuple([int(x) for x in np.ones(n)]))
# Tupel-Sortierung von "all data" zu "only last data"
s   = [sum(x) for x in prm]
prm = [prm[i] for i in np.argsort(s)][::-1]
return prm
# Output für myperm(5):
[(1, 1, 1, 1, 1),
(1, 1, 1, 0, 1),
. . .
(0, 1, 0, 0, 0),
(0, 0, 0, 0, 1)]
```

A.5 Visualisierung Datensatzmanipulation

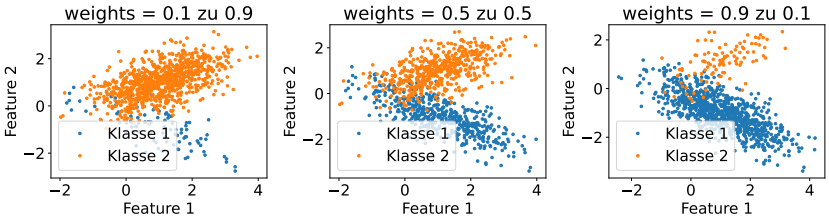


Abbildung A.1: Unterschiedliche Gewichtung der Klassen

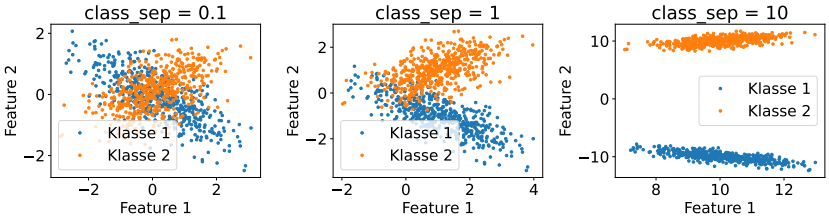


Abbildung A.2: Distanz der Feature-Werte je Klasse

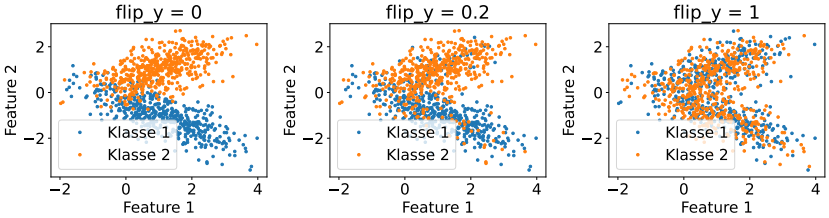


Abbildung A.3: Fehler in der Klassenzuordnung

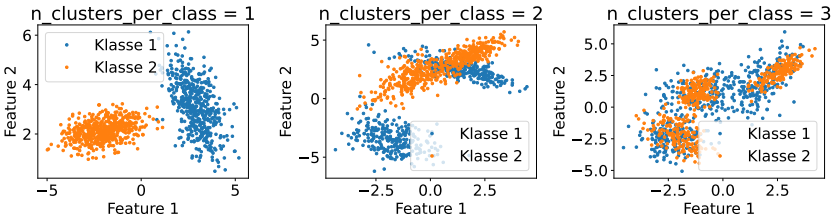


Abbildung A.4: Unterschiedliche Anzahl an Clustern je Klasse

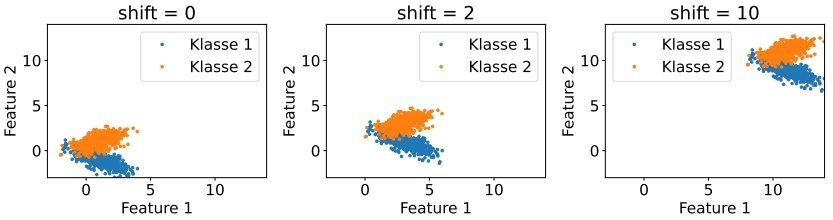


Abbildung A.5: Verschiebung der Clustermittelwerte

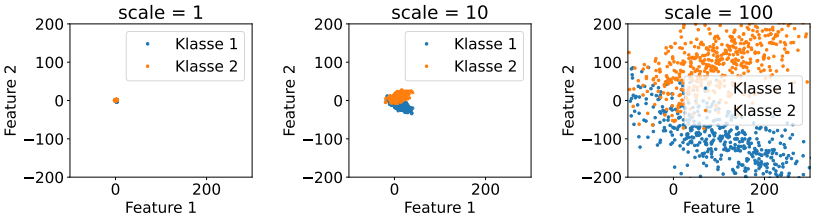


Abbildung A.6: Skalierung der Clusterstreuung

A.6 Prognoseperformance Einzelkennzahlen

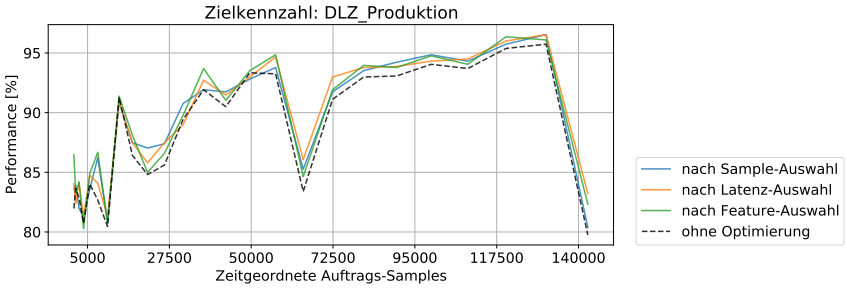


Abbildung A.7

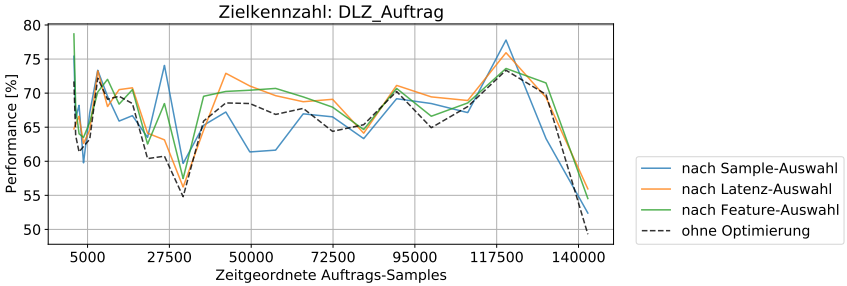


Abbildung A.8

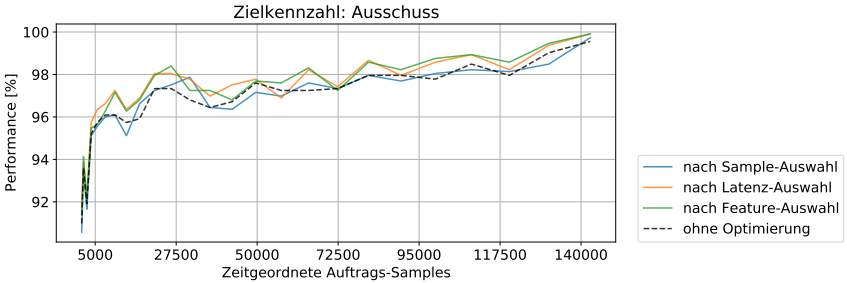


Abbildung A.9

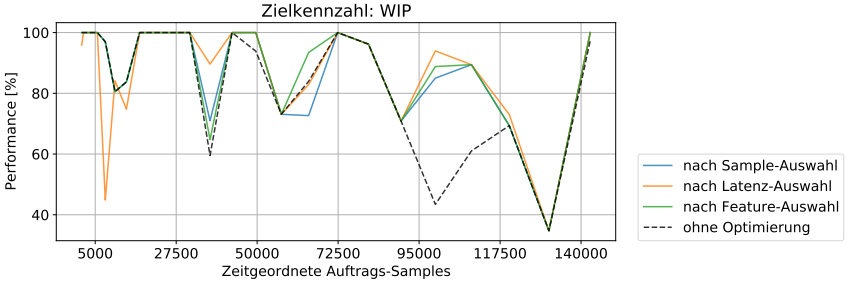


Abbildung A.10

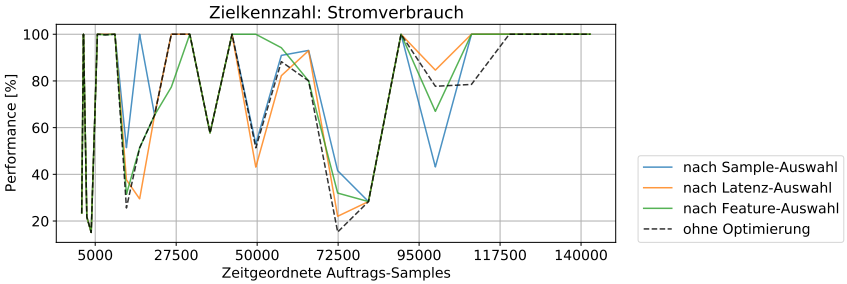


Abbildung A.11

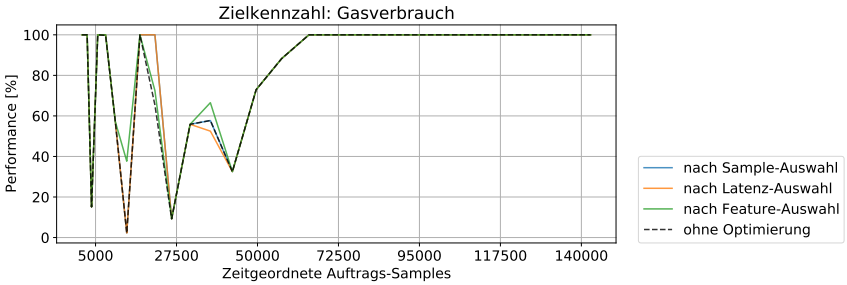


Abbildung A.12

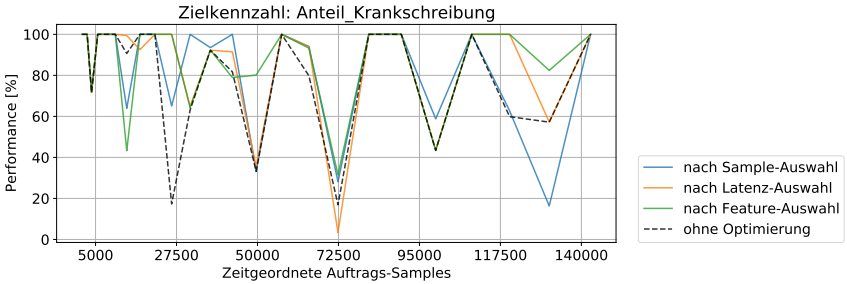


Abbildung A.13

A.7 Erläuterung Ensemble-Methoden

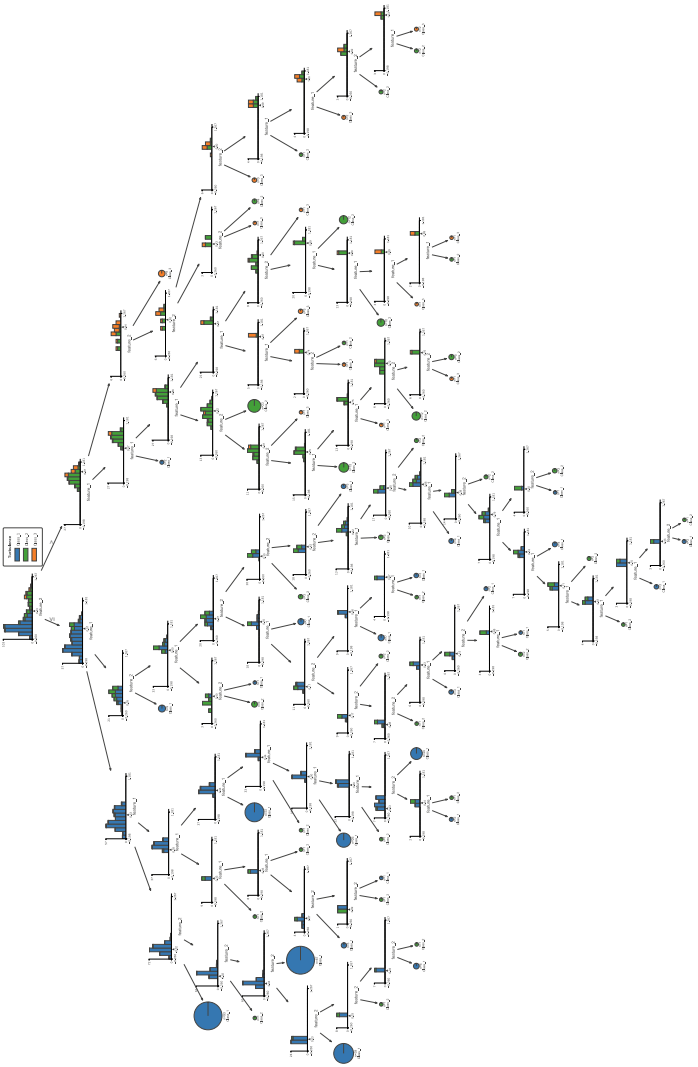


Abbildung A.14: Detaillierte Darstellung eines Entscheidungsbaums (siehe Abschnitt 5.4)

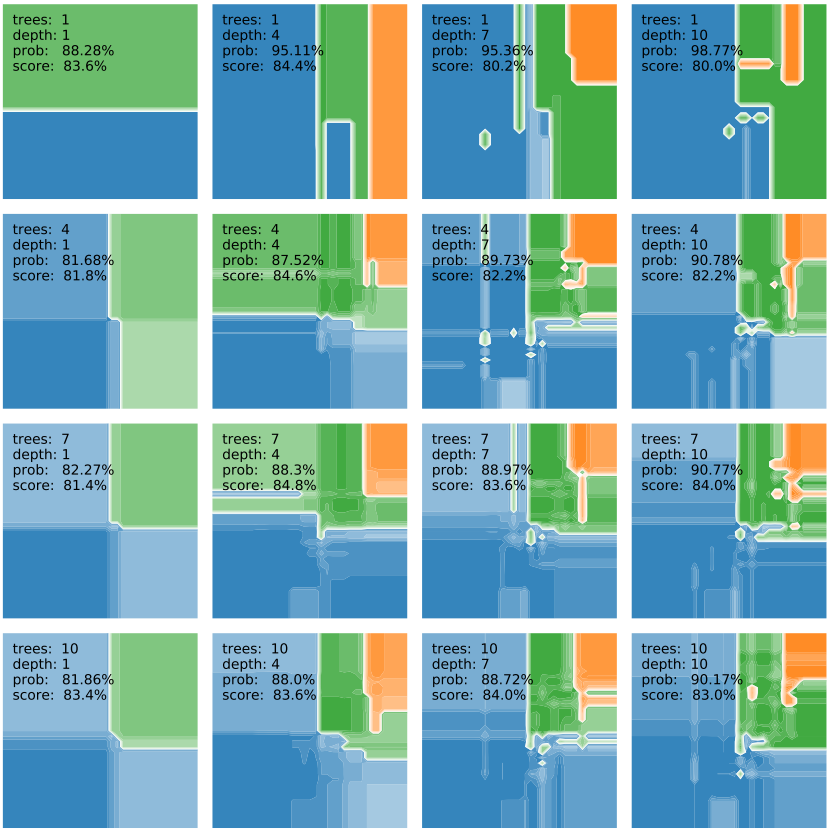


Abbildung A.15: Verhalten der Entscheidungswälder in Abhängigkeit von Baumtiefe (dp) und Anzahl der Schätzer (tr)

A.8 Parameterübersicht

Tabelle A.2: Eingabeparameter für das Produktionsmodell

Eingabeparameter	Wert	Beschreibung
Produktionssystem		
procs	$[1:20]$	Anzahl Prozesse
n_res	$[1:20]$	Anzahl Ressourcen je Prozess
buf_size	$[0:\infty]$	Puffergröße je Ressource
mttr	$[0:\infty]$	Mean Time To Repair in [Tage]
avail	$[0:100]$	Verfügbarkeit in [%]
ct	$[0:\infty]$	Zykluszeiten in [Sekunden]
sut	$[0:\infty]$	Rüstzeiten in [Sekunden]
seed	$[0:2^{32}-1]$	Startwert, gewährleistet Reproduzierbarkeit
transport	$n_{n\_res} \times n_{n\_res}$	Distanzmatrix zwischen den Ressourcen
optional weitere	$[0:\infty]$	Energieverbrauch je Maschinenstatus, Abschreibung, Invest- und Betriebskosten...
Produktportfolio		
n_pfs	$[0:\infty]$	Anzahl Produktfamilien (PF)
n_proc	$[0:\infty]$	Anzahl Prozesse je PF
n_var	$[0:\infty]$	Anzahl Varianten je PF
processes	$P_1 ... P_{n\_procs}$	Prozesse und Reihenfolge je PF
mean_ct	$[0:\infty]$	*mittlere* Zykluszeiten in [Sekunden]
mean_sut	$[0:\infty]$	*mittlere* Rüstzeiten in [Sekunden]
Systemlast		
period	$[0:\infty]$	Simuliertes Zeitintervall in [Sekunde]
n_orders	$[0:\infty]$	Anzahl Aufträge im simulierten Zeitintervall
oz	$[0:\infty]$	Anzahl Teile je Auftrag
sales_ratio	$[0:1]$	Anteile am Gesamtvolumen je PF
seasonality	$[0:1]$	Einfluss von Saisonalität
reoc_seasons	$[0:\infty]$	Anzahl Zyklen im simulierten Zeitintervall
trend	$[a, b, .., z]$	Trendverlauf als Funktionswerte der Zeit
scatter	$[a, b, .., z]$	Streuung als Funktionswerte der Zeit

Tabelle A.3: Eingabeparameter für die Baumgenerierung

Eingabeparameter	Wert	Kommentar
n_estimators	50	Ausreichend für qualitative Aussage
criterion	$gini$	Default
max_depth	34	Maximale Feature-Anzahl pro Target
min_samples_split	2	Default
min_samples_leaf	1	Default
min_weight_fraction_leaf	0.0	Default
max_features	34	Maximale Feature-Anzahl pro Target
max_leaf_nodes	$None$	Default
min_impurity_decrease	0.0	Default
min_impurity_split	$None$	Default
bootstrap	$True$	Default
oob_score	$False$	Default
n_jobs	$None$	Default
random_state	42	Reproduzierbarkeit
verbose	0	Default
warm_start	$False$	Default
class_weight	$None$	Keine Gewichtung in generierten Daten
ccp_alpha	0.0	Default
max_samples	$None$	Default