TECHNISCHE UNIVERSITÄT DRESDEN

Fakultät Maschinenwesen: Professur für Virtuelle Produktentwicklung

Ein Framework zur Analyse komplexer Produktportfolios mittels Machine Learning

Dissertation

zur Erlangung des akademischen Grades

Doktoringenieur (Dr.-Ing.)

vorgelegt der Fakultät Maschinenwesen

der Technischen Universität Dresden

von

Jan Mehlstäubl

Vorsitzender: Prof. Dr.-Ing. Jens Krzywinski

Erster Gutachter: Prof. Dr.-Ing. Kristin Paetzold-Byhain

Zweiter Gutachter: Prof. Dr.-Ing. Matthias Kreimeyer, Ingénieur ECP

Hauptfachprüfer: Prof. Dr.-Ing. Kristin Paetzold-Byhain

Nebenfachprüfer: Prof. Dr.-Ing. habil. Michael Völker

Tag der Einreichung: 28.04.2023

Tag der Verteidigung: 21.11.2023

Danksagung

Die vorliegende Dissertation entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter von 2019 bis 2023 am Institut für Technische Produktentwicklung an der Universität der Bundeswehr in München. Während dieser lehrreichen Zeit bekam ich die Möglichkeit spannende wissenschaftliche Themenstellungen im industriellen Kontext zu untersuchen und umzusetzen. Viele Menschen haben mich auf diesem Weg begleitet und unterstützt, wofür ich mich bedanken möchte.

Mein besonderer Dank gilt zuallererst meiner Doktormutter Prof. Kristin Paetzold-Byhain, die mir die Möglichkeit zur Promotion an ihrem Institut gab. Ihre Unterstützung und ihr Vertrauen bei der Durchführung unseres Forschungsprojekts sowie bei der Ausarbeitung meiner Dissertation haben wesentlich zu dessen Erfolg beigetragen. Darüber hinaus hat sie mir den Freiraum gelassen, um meine Ideen umzusetzen und mich persönlich weiterzuentwickeln. Zudem möchte ich Prof. Matthias Kreimeyer für seine Unterstützung und sein wertvolles Feedback in den letzten drei Jahren bedanken, welches eine große Bereicherung für meine Arbeit darstellte.

Ohne meine Partner aus der Industrie wäre diese Promotion so nicht möglich gewesen. Besonderer Dank gilt dabei Felix Braun von der MAN Truck & Bus SE. Felix hat durch sein Vertrauen und seine Unterstützung unser Forschungsprojekt in Zeiten von Corona am Leben gehalten und dafür gesorgt, dass ich meine Ideen in der Industrie umsetzen konnte. Daneben stand er mir bei meinen Veröffentlichungen und Präsentationen zur Seite. Zusätzlich möchte ich mich bei Ralf Kraul von der MAN Truck & Bus SE für seine Ideen und seine fachliche Unterstützung bedanken. Daneben gilt mein Dank all jenen, die mich im Rahmen dieser Dissertation als Feedbackgeber und Interviewpartner unterstützt haben.

Neben den Partnern aus der Industrie möchte ich mich bei meinen Kollegen und Freunden vom Institut für Technische Produktentwicklung und der Universität der Bundeswehr München Alexander Atzberger, Emir Gadzo, Martin Denk, Julian Schönwald, Joaquin Montero, Marvin Michalides, Laura Wirths, Simon Nicklas, Sebastian Weber, Alexander Schmidt, Lea Strauß und Michael Ascher für die Unterstützung, den fachlichen Austausch sowie unsere gemeinsamen Aktivitäten bedanken. Des Weiteren gilt mein Dank den Studenten, die mich im Rahmen ihrer Abschlussarbeiten unterstützt haben. Besonders danken möchte ich dabei Christoph Pfeiffer.

Nicht zu vergessen ist die Bayerische Forschungsstiftung, die unseren Forschungsverbunde „FORCuDE“ gefördert hat, in welchem diese Dissertation entstanden ist. In diesem Zuge möchte ich mich auch bei allen Kollegen aus dem Forschungsverbund für die gute Zusammenarbeit, das Feedback und den regelmäßigen Austausch bedanken.

München, April 2023 Jan Mehlstäubl

Vorangehende Veröffentlichungen

Die folgenden Veröffentlichungen sind Teil der in dieser Dissertation vorgestellten Arbeit:

Mehlstäubl, J., Braun, F. and Paetzold, K. (2021a), „Artificial Intelligence in Product Portfolio and Variety Management in Commercial Vehicle Industry – An Overview about Expectations, Challenges and Use Cases“, Presentation on the Prostep Ivip Symposium 2021.

Mehlstäubl, J., Braun, F. and Paetzold, K. (2021b), „Data Mining in Product Portfolio and Variety Management – Literature Review on Use Cases and Research Potentials“, 2021 IEEE Technology & Engineering Management Conference-Europe (TEMSCON-EUR), pp. 442–447.

Mehlstäubl, J., Nicklas, S., Gerschütz, B., Sprogies, N., Schleich, B., Lohner, T., Wartzack, S., et al. (2021), „Voraussetzungen für den Einsatz datengetriebener Methoden in der Produktentwicklung“, Proceedings of the 32nd Symposium Design for X (DfX 2021).

Mehlstäubl, J. and Braun, F. (2022), „Künstliche Intelligenz im Produktportfolio- und Variantenmanagement: Machine Learning zur Handhabung der Portfoliokomplexität am Beispiel der MAN Truck & Bus SE“, Präsentation auf der Smart Variant, Berlin.

Mehlstäubl, J., Braun, F., Denk, M., Kraul, R. and Paetzold, K. (2022), „Using Machine Learning for Product Portfolio Management: A Methodical Approach to Predict Values of Product Attributes for Multi-Variant Product Portfolios“, Proceedings of the Design Society, Vol. 2, pp. 1659–1668.

Mehlstäubl, J., Gadzo, E., Atzberger, A. and Paetzold, K. (2022), „Herausforderungen datengetriebener Methoden in der Produktentwicklung / Challenges of data-driven methods in product development“, VDI Konstruktion, Vol. 74 No. 06, pp. 60–66.

Mehlstäubl, J., Braun, F., Gadzo, E. and Paetzold, K. (2023), „Machine Learning to generate Knowledge for Decision-making Processes in Product Portfolio and Variety Management“, 9th International Conference on Research Into Design.

Mehlstäubl, J., Braun, F. and Paetzold-Byhain, K. (2023), „Reduktion komplexer Produktportfolios durch die Ableitung von Kombinatorikregeln aus verkauften Produktkonfigurationen mit einer Assoziationsanalyse“, Stuttgarter Symposium für Produktentwicklung.

Mehlstäubl, J., Pfeiffer, C., Kraul, R., Braun, F. and Paetzold-Byhain, K. (2023), „Methodical approach to cluster configurations of product variants of complex product portfolios“, 24th International Conference on Engineering Design (ICED).

Gerschütz, B., Sauer, C., Wallisch, A., Mehlstäubl, J., Kormann, A., Schleich, B., Alber-Laukant, B., et al. (2021), „Towards Customized Digital Engineering: Herausforderungen und Potentiale bei der Anpassung von Digital Engineering Methoden für den Produktentwicklungsprozess“, Stuttgarter Symposium Für Produktentwicklung 2021 (SSP 2021), Fraunhofer-Institut für Arbeitswirtschaft und Organisation IAO.

Zusammenfassung

Die Nachfrage der Kunden nach individualisierten Produkten, die Globalisierung, neue Konsummuster sowie kürzere Produktlebenszyklen führen dazu, dass Unternehmen immer mehr Varianten anbieten. Aufgrund der Arbeitsteilung und der unterschiedlichen Perspektiven können einzelne Entwickler die Komplexität des Produktportfolios nicht durchdringen. Dennoch sind die heutigen Verfahren im Produktportfolio- und Variantenmanagement geprägt durch manuelle und erfahrungsbasierte Aktivitäten. Eine systematische Analyse und Optimierung des Produktportfolios sind damit nicht möglich. Unternehmen benötigen stattdessen intelligente Lösungen, welche das gespeicherte Wissen in Daten nutzen und einsetzen, um Entscheidungen über Innovation, Differenzierung und Elimination von Produktvarianten zu unterstützen.

Zielstellung dieses Forschungsvorhabens ist die Entwicklung eines Frameworks zur Analyse komplexer Produktportfolios mittels Machine Learning. Machine Learning ermöglicht es, Wissen aus Daten unterschiedlicher Lebenszyklusphasen einer Produktvariante automatisiert zu generieren und zur Unterstützung des Produktportfolio- und Variantenmanagements einzusetzen. Für die Unterstützung der Entscheidungen über Produktvarianten ist Wissen über deren Abhängigkeiten und Beziehungen sowie die Eigenschaften der einzelnen Elemente erforderlich. Dadurch soll ein Beitrag zur besseren Handhabung komplexer Produktportfolios geleistet werden.

Das Framework zur Analyse komplexer Produktportfolios mittels Machine Learning besteht aus drei Bausteinen, die das zentrale Ergebnis dieser Arbeit darstellen. Zuerst wird in Baustein 1 auf die Wissensbedarfe bei der Analyse und Anpassung komplexer Produktportfolios eingegangen. Anschließend werden in Baustein 2 die Daten, welche für Entscheidungen und somit für die Wissensgenerierung im Produktportfolio- und Variantenmanagement erforderlich sind, beschrieben und charakterisiert. Abschließend findet in Baustein 3 die Datenvorbereitung und die Implementierung der Machine Learning Verfahren statt. Es wird auf unterschiedliche Verfahren eingegangen und eine Unterstützung bei der Auswahl und Evaluation der Algorithmen sowie die Möglichkeiten zum Einsatz des generierten Wissens für die Analyse komplexer Produktportfolios aufgezeigt.

Das Framework wird in einer Fallstudie bei einem Industriepartner aus der Nutzfahrzeugbranche mit einem besonders komplexen Produktportfolio angewendet. Dabei werden die drei Anwendungsfälle Prognose von „marktspezifischen und technischen Eigenschaften der Produktvarianten“, Ermittlung von „Ähnlichkeiten von Produktvarianten“ und Identifikation von „Korrelationen zwischen Merkmalsausprägungen“ mit realen Daten des Industriepartners umgesetzt. Das Framework sowie die in der Fallstudie beim Industriepartner erzielten Ergebnisse werden anschließend Experten im Produktportfolio- und Variantenmanagement vorgestellt. Diese bewerten die

Ergebnisse hinsichtlich der funktionalen Eigenschaften sowie dem Mehrwert aus Sicht der Forschung und industriellen Praxis anhand zuvor definierter Kriterien.

Summary

Customer demand for individualised products, globalisation, new consumption patterns and shorter product life cycles are pushing companies to offer more and more product variants. Due to the division of labour and the different perspectives, individual engineers cannot understand and overlook the complexity of the product portfolio. Nevertheless, today's processes in product portfolio and variety management are characterised by manual and experience-based activities. A systematic analysis and optimisation of the product portfolio is therefore not possible. Instead, companies need intelligent solutions that use the knowledge stored in data to support decisions about innovation, differentiation, and the elimination of product variants.

The aim of this research is to develop a framework for analysing complex product portfolios using machine learning. Machine learning makes it possible to automatically generate knowledge from data of different life cycle phases of a product variant and to use it to support product portfolio and variety management. To support decisions about product variants, knowledge about their dependencies and relationships as well as the properties of the individual elements is required. Therefore, the framework contributes to better handling of complex product portfolios.

The framework for analysing complex product portfolios using machine learning consists of three building blocks that represent the main result of this work. First, building block 1 deals with the knowledge needs for the analysis and adaptation of complex product portfolios. Subsequently, building block 2 describes and characterises the data required for decisions and thus for knowledge generation in product portfolio and variety management. Finally, in building block 3, the data preparation and the implementation of the machine learning methods take place. Different methods are discussed and support for the selection and evaluation of algorithms as well as the possibilities of using the generated knowledge for the analysis of complex product portfolios are presented.

The framework is applied in a case study with an industrial partner from the commercial vehicle industry with a particularly complex product portfolio. The three use cases of forecasting "market-specific and technical attributes of product variants", determining "similarities of product variants" and identifying "correlations between customer choices" are implemented with real-world data from the industrial partner. The framework and the results of the case study are then presented to product portfolio and variety management experts. They evaluate the results in terms of functionality and added value from a research and industrial perspective using previously defined criteria.

Inhaltsverzeichnis

1 Einführung 1

1.1 Motivation 1

1.2 Komplexe Produktportfolios: Eine Industrieperspektive 3

1.3 Zielsetzung und Forschungsfragen 5

1.4 Aufbau der Arbeit 6

2 Grundlagen zur Analyse von Produktportfolios mittels Machine Learning 9

2.1 Komplexe Produktportfolios 9

2.1.1 Terminologie komplexer Produktportfolios 9

2.1.2 Strukturierung komplexer Produktportfolios 13

2.1.3 Analyse und Anpassung komplexer Produktportfolios 15

2.1.4 Zusammenfassung: Komplexe Produktportfolios 17

2.2 Machine Learning 18

2.2.1 Machine Learning als Teil der künstlichen Intelligenz 18

2.2.2 Terminologie Machine Learning 19

2.2.3 Wissensgenerierung mit Machine Learning 21

2.2.4 Datenanalyseprozess 22

2.2.5 Machine Learning Verfahren und Algorithmen 25

2.2.6 Zusammenfassung: Machine Learning 42

3 Ansätze zur Analyse von Produktportfolios mittels Machine Learning 43

3.1 Kriterien zur Bewertung bestehender Ansätze 43

3.2 Bestehende Ansätze aus der Literatur 44

3.2.1 Einsatz überwachter Lernverfahren 44

3.2.2 Einsatz unüberwachter Lernverfahren 47

3.2.3 Einsatz kombinierter Lernverfahren 54

3.3 Resultierender Forschungsbedarf 55

4 Forschungsvorgehen 58
4.1 Design Research Methodology (DRM) 58
4.2 Vorgehen und Methodeneinsatz 59
4.3 Kriterien für die Entwicklung des Frameworks 62
4.4 Schlussfolgerungen zum Forschungsvorgehen 64

5 Framework zur Analyse komplexer Produktportfolios 66
5.1 Übersicht über das Framework 66
5.2 Baustein 1: Wissensbedarfe zur Analyse komplexer Produktportfolios 67
5.2.1 Informationssuche 67
5.2.2 Formulierung von Alternativen 69
5.2.3 Prognose 69
5.2.4 Kriterien zur Auswahl der Wissensbedarfe 70
5.3 Baustein 2: Datenbasierte Beschreibung komplexer Produktportfolios 71
5.3.1 Produktdatenmodell 72
5.3.2 Vertriebsdaten 75
5.3.3 Nutzungsdaten 77
5.4 Baustein 3: Systematische Generierung und Einsatz von Wissen 78
5.4.1 Baustein 3.0: Vorbereitung von Produktportfoliodaten 78
5.4.2 Baustein 3.1: Regressionsanalyse 82
5.4.3 Baustein 3.2: Klassifikationsanalyse 87
5.4.4 Baustein 3.3: Clusteranalyse 91
5.4.5 Baustein 3.4: Assoziationsanalyse 95
5.5 Anwendung des Frameworks 97
5.6 Schlussfolgerung zum Framework 99

6 Validierung des Frameworks 101
6.1 Konzept der Validierung 101
6.2 Baustein 1: Wissensbedarfe zur Analyse komplexer Produktportfolios 102
6.3 Baustein 2: Datenbasierte Beschreibung komplexer Produktportfolios 105
6.4 Baustein 3: Systematische Generierung und Einsatz von Wissen 106

6.4.1 Marktspezifische und technische Produkteigenschaften 107
6.4.2 Ähnlichkeiten von Produktvarianten 121
6.4.3 Korrelationen zwischen Merkmalsausprägungen 130
6.5 Erfolgsvalidierung mit einer Expertenbefragung 133
6.6 Schlussfolgerung zur Validierung 138

7 Diskussion 140
7.1 Nutzen und Einschränkungen 140
7.2 Ergebnisbeitrag für die Forschung 142
7.3 Ergebnisbeitrag für die Industrie 142

8 Zusammenfassung und Ausblick 144
8.1 Zusammenfassung 144
8.2 Ausblick 145

9 Literaturverzeichnis 147

10 Abbildungsverzeichnis 167

11 Tabellenverzeichnis 172

Anhang A-1

1 Einführung

„Ich interessiere mich nicht für Daten um ihrer selbst willen,
sondern dafür, wie sie genutzt werden können, um Dinge zu verbessern.“
(Bill Gates)

1.1 Motivation

Die Zahl der von Unternehmen angebotenen Produktvarianten ist in den letzten Jahrzehnten aufgrund der Nachfrage der Kunden nach individuellen Produkten, der Globalisierung, der kürzeren Produktlebenszyklen und der neuen Konsummuster stetig gestiegen (Krause und Gebhardt 2018). Diese Erhöhung der externen Produktvielfalt führt zu einer **steigenden Produktportfoliokomplexität** (Wildemann 2011), welche sich auf die Produkte und Prozesse eines Unternehmens auswirkt und für eine Zunahme der Kosten sorgt (Schuh et al. 2018; Song und Kusiak 2009). Ziel beim Umgang mit komplexen Produktportfolios ist zum einen die optimale Produktvielfalt zu definieren, welche zur Maximierung des Unternehmensgewinns beiträgt (Rathnow 1993), und zum anderen die Handhabung der auf das Produktportfolio eingehenden sowie ausgehenden Komplexität mittels geeigneter Instrumente (Kipp 2012).

Die Aufnahme kundenspezifischer Produktvarianten in das Produktportfolio führt zu einer kontinuierlichen Erhöhung der Produkt-, Teile- und Verfahrensvielfalt (Gembrys 1998). Aufgrund veränderter Marktanforderungen und anderer externer Faktoren wie Normen oder Vorschriften, die Produktvarianten obsolet werden lassen, müssen bestehende Produktvarianten auf ihre Notwendigkeit hin überprüft und das Produktportfolio entsprechend angepasst werden (Gebhardt et al. 2016). Zudem lassen Unternehmen bei der Einführung neuer Produktvarianten häufig geringe Absatzmengen außer Acht, sodass die Komplexitätskosten für Verluste und somit für einen Wettbewerbsnachteil sorgen (Hu 2013). Viele Unternehmen richten ihren Fokus daher auf Projekte zur **Rationalisierung der Produktvielfalt** (Bannasch und Bouché 2016). Eine Reduktion der externen Produktvielfalt um bestehende Produktvarianten führt auch zu einer Verringerung der internen Komplexität und damit zu einer Senkung der Komplexitätskosten (Heina 1999).

Eine Anpassung und Rationalisierung des Produktportfolios erfordert umfassendes Wissen über das Produkt, den Markt sowie unternehmensinterne Größen, wie zum Beispiel Konstruktions- und Fertigungsverfahren (Tucker und Kim 2009). Die Aufgabe der operativen Produktportfoliogestaltung ist die Variantenvielfalt zu analysieren und Entscheidungen über entsprechende Maßnahmen zu treffen (Heina 1999). Trotz der immer weiter steigenden Produktportfoliokomplexität basieren heutzutage die Pro-

duktportfolioentscheidungen in Unternehmen oft auf dem **Expertenwissen der Entwickler** (Riesener et al. 2019a). Jedoch ist der menschliche Verstand nur in der Lage, drei bis vier entscheidungsrelevante Faktoren auf einmal zu berücksichtigen (Blase et al. 2016). Dies hat häufig unzureichende Entscheidungsergebnisse sowie eine mangelnde Nachvollziehbarkeit und Transparenz des Entscheidungsprozesses zur Folge (Jank 2021). Für die Analyse komplexer Produktportfolios sind daher neue Ansätze erforderlich (Schuh et al. 2018), welche fundierte Entscheidungen über die zukünftig am Markt angebotenen Produktvarianten ermöglichen (Tucker und Kim 2009).

Für die Analyse komplexer Produktportfolios stehen den Unternehmen durch die Digitalisierung neue technologische Möglichkeiten zur Verfügung. Speziell der Einsatz des **Wissens aus Daten** kann Unternehmen einen Wettbewerbsvorteil bieten (Song und Kusiak 2009). Mit voranschreitender Individualisierung von Produkten, Erschließung neuer Märkte und Ausbau internationaler Wertschöpfungsketten nehmen die Daten, die rund um die Produktvarianten entstehen und gespeichert werden, erheblich zu (Helms und Kissel 2016). Entscheidungen über das Produktportfolio erfordern die Verfügbarkeit von Informationen in einer Form, die von den Entwicklern aus verschiedenen Funktionen verstanden werden (Battistello et al. 2021). Dabei gilt es zunächst die entscheidungsrelevanten Daten aufwendig aus den unterschiedlichen Systemen in den Unternehmen zu beschaffen, zu bereinigen und aufzubereiten (Helms und Kissel 2016). Die Analyse der Daten erfolgt dabei weitestgehend manuell und erfordert somit einen hohen Aufwand, welcher eine vollumfängliche Betrachtung komplexer Produktportfolios unmöglich macht (Kissel 2014).

Damit die großen Datenmengen komplexer Produktportfolios umfassend analysiert und in die Entscheidungen einbezogen werden können, sind digitale Methoden und Werkzeuge erforderlich (Agard und Kusiak 2004b). Mit datenbasierten Analyseverfahren können effektive Entscheidungen im Produktportfolio- und Variantenmanagement ermöglicht werden (Riesener et al. 2019b). Ein besonders großes Potenzial, um die Komplexität des Portfolios besser handhabbar zu machen, bieten Ansätze des **Machine Learning** (Riesener et al. 2020). Durch den Einsatz von Machine Learning kann wertvolles und bisher unbekanntes Wissen über die Produkte und Märkte aus großen Datensätzen extrahiert werden, um komplexe Produktportfolios systematisch zu analysieren (Moon et al. 2006).

Im Produktportfolio- und Variantenmanagement ist der Einsatz von datenbasierten Analysemethoden bisher nur wenig erforscht (Riesener et al. 2020). In der Industrie wird Machine Learning im gesamten Umfeld der Produktentwicklung kaum eingesetzt (Bertoni et al. 2017). In der Literatur finden sich bereits erste Anwendungsfälle von Machine Learning im Produktportfolio- und Variantenmanagement (Mehlstäubl et al. 2021b). Allerdings gibt es **keinen ganzheitlichen methodischen Ansatz**, wie Wissen systematisch generiert und für die Analyse und Anpassung komplexer Produktportfolios eingesetzt werden kann (Mehlstäubl et al. 2023a).

1.2 Komplexe Produktportfolios: Eine Industrieperspektive

Die Komplexität heutiger Produktportfolios und die damit verbundenen Herausforderungen werden im Folgenden aus einer Industrieperspektive erläutert und dadurch die zuvor beschriebene Motivation dieser Arbeit untermauert. Die varianteninduzierte Komplexität aus Sicht der Entwicklung ergibt sich nach Kreimeyer (2012) aus drei Faktoren: Erstens aus der externen Vielfalt in Form von Merkmalen und Merkmalsausprägungen. Zweitens aus den daraus resultierenden technischen Komponenten und Komponentenvarianten, um die externe Vielfalt umzusetzen sowie drittens aus der Positionierung der Komponentenvarianten in einer Produktvariante.

Produktportfolios mit einer besonders hohen Komplexität sind beispielsweise Automobile oder Nutzfahrzeuge. In der Automobilindustrie ist die Variantenzahl von 30-50 optionalen Merkmalsausprägungen in den 1970ern zu 300-500 optionale Merkmalsausprägungen in den 2000ern gestiegen (Fricke und Schulz 2005). Die Kombinierbarkeit der Merkmalsausprägungen führt dazu, dass zum Beispiel ein 7er BMW bis zu 10^{17} mögliche Produktvarianten besitzt (Hu et al. 2008). Nutzfahrzeughersteller stellen Fahrzeuge zwischen 7,5 und 44 Tonnen Gesamtgewicht, Sonderfahrzeuge bis 250 Tonnen Zuggesamtgewicht sowie Busse für Stadt- und Überlandfahrten her (Kreimeyer et al. 2011). Diese Vielzahl an Einsatzszenarien ergibt eine Fülle an Merkmalen und Merkmalsausprägungen, welche vom Kunden ausgewählt werden können, sowie daraus resultierende technischen Komponenten und Komponentenvarianten. Daraus entsteht eine enorme Anzahl an Produktvarianten, die vom Kunden konfiguriert werden können (Kusiak et al. 2007). Das Produktportfolio eines Nutzfahrzeugherstellers kann über 30 000 Komponentenvarianten und 100 000 Stücklistenelemente enthalten, welche in über 10^{300} Produktvarianten resultieren (Braun 2021; Kreimeyer 2012; Braun et al. 2018). Die Kombinierbarkeit zwischen Merkmalsausprägungen sowie die Zuordnung der Kombinationen aus Merkmalsausprägungen zu den Komponentenvarianten wird beschrieben mit bis zu 100 000 Booleschen Regeln (Braun 2021), welche wiederum stark untereinander vernetzt sind. Erschwerend kommen in der Nutzfahrzeugbranche im Vergleich zur Automobilbranche die geringen Stückzahlen und fehlenden Skaleneffekte hinzu. Im Jahr 2019 hat Volkswagen knapp 680 000 Fahrzeuge des Golf Derivats verkauft (Volkswagen AG 2019). Der Nutzfahrzeughersteller MAN Truck & Bus hat in 2019 dagegen mit allen Lastkraftwagen lediglich knapp 83 000 Fahrzeuge abgesetzt (MAN Truck & Bus SE 2019).

In vielen Unternehmen besteht die Analyse solch komplexer Produktportfolios heutzutage in erster Linie aus manuellen Aktivitäten und dem Einsatz von Erfahrungswissen. In Abbildung 1-1 ist das bei einem Industriepartner aus der Nutzfahrzeugbranche eingesetzte Vorgehen dargestellt. Es werden zuerst aus unterschiedlichen Datenquellen (z. B. Exceltabellen und Datenbanksystemen) manuell Daten zusammengeführt. Diese Daten beinhalten das Produktdatenmodell, d. h. Information darüber, welche Merkmalsausprägungen miteinander kombiniert werden können, sowie die verkauften Stückzahlen einzelner Merkmalsausprägungen und Kombinationen dieser. Die

Daten werden in unüberschaubaren Matrizen aufbereitet (siehe Abbildung 1-2). Darin werden die Stückzahlen der Merkmalsausprägungen mehrerer Merkmale in einer Matrix gegenübergestellt.

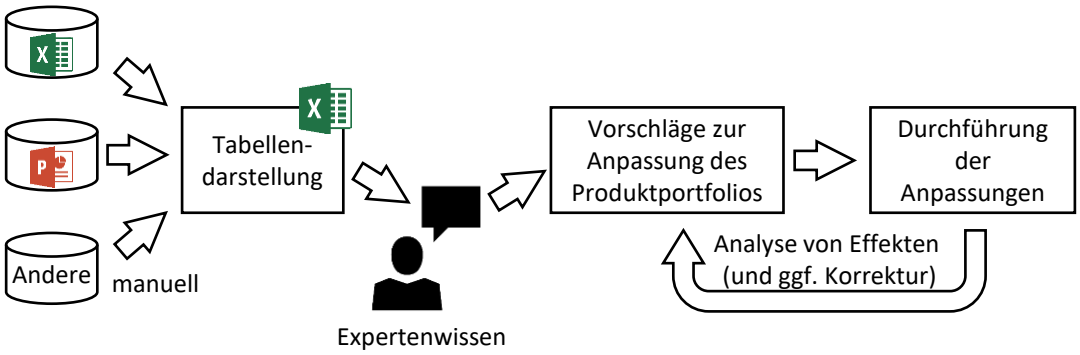


Abbildung 1-1 Aktuelles Vorgehen zur Analyse komplexer Produktportfolios bei einem Industriepartner

Anschließend werden sogenannte Null- und Wenigdreher, d. h. Produktvarianten mit geringen Stückzahlen, ermittelt und von Experten mit langjähriger Erfahrung in der Nutzfahrzeugbranche bewertet und Anpassungen des Produktportfolios vorgeschlagen. Die Vorschläge werden in Gremien mit Experten aus unterschiedlichen Disziplinen, wie Entwicklung und Vertrieb, bewertet und bilden den Ausgangspunkt für die Entscheidungen im Produktportfolio- und Variantenmanagement.

Abbildung 1-2: Gegenüberstellung mehrerer Merkmale eines komplexen Produktportfolios bei einem Industriepartner

Die Betrachtungen aus der Perspektive der Industrie zeigen, dass die aktuellen Ansätze der Komplexität der Produktportfolios nicht gerecht werden und diese nicht handhaben können. Aus diesem Grund sind neue und intelligente Verfahren erforderlich, um komplexe Produktportfolios automatisiert zu analysieren und dadurch handhabbar zu machen. Jedoch fehlt es in der Industrie aktuell an Wissen über intelligente Verfahren wie Machine Learning, um Anwendungsfälle im Produktportfolio- und Variantenmanagement zu identifizieren und zu implementieren (Mehlstäubl et al. 2021a). Daher benötigen Unternehmen eine Unterstützung beim Einsatz von Machine Learning zur Analyse komplexer Produktportfolios. Die Anforderungen an einen solchen Ansatz werden in Kapitel 3.1 näher erläutert.

1.3 Zielsetzung und Forschungsfragen

Die Zielsetzung dieser Arbeit leitet sich aus der oben beschriebenen Motivation und Problemstellung der Industrie ab und unterstützt die übergeordnete Vision die Produktportfoliokomplexität in Industrieunternehmen besser handhabbar zu machen. Das **Ziel** ist die Entwicklung eines Frameworks zur systematischen Generierung von Wissen für die Analyse komplexer Produktportfolios mittels Machine Learning. Unter einem Framework wird hier ein Rahmenwerk verstanden, welches ein Verständnis für die Machine Learning Verfahren, Algorithmen und Werkzeuge im spezifischen Kontext vermittelt.

Die Anforderungen an das Framework lassen sich weiter hinsichtlich des Anwendungsbereichs sowie der Inhalte und der Ergebnisse konkretisieren. Das Framework soll es ermöglichen komplexe Produktportfolios unter Berücksichtigung der Daten aus unterschiedlichen Systemen und unter Verwendung von verschiedene Machine Learning Verfahren zu analysieren. Die Analyse soll auf der Ebene der operativen Produktportfoliogestaltung stattfinden und die Reduktion von Produktvarianten unterstützen. Dafür ist zum einen aufzuzeigen, welches Wissen für die Rationalisierung des Produktportfolios relevant ist, indem der Prozess zur Analyse und Anpassung von Produktportfolios untersucht wird. Des Weiteren ist ein Verständnis für die Produktportfoliodaten und deren Charakteristiken bereitzustellen, was für den Einsatz von Machine Learning unabdingbar ist. Für die Implementierung ist eine Unterstützung bei der Vorbereitung, Erstellung und Evaluation der Machine Learning Modelle zu bieten sowie deren Möglichkeiten zur Generierung von Wissen im spezifischen Kontext aufzuzeigen. Den Nutzern soll durch das Framework ermöglicht werden, Machine Learning im Kontext der Analyse komplexer Produktportfolios zu verwenden, um deren spezifische Bedürfnisse gerecht zu werden. Die Kriterien werden in Kapitel 3.1 weiter detailliert.

Aus der Zielsetzung leiten sich die folgenden drei Forschungsfragen ab:

Forschungsfrage 1: Welches Wissen kann mittels Machine Learning für die Analyse komplexer Produktportfolios generiert werden?

Durch die Beantwortung von Forschungsfrage 1 werden aktuelle Wissenslücken bei der Entscheidungsfindung im Produktportfolio- und Variantenmanagement aufgezeigt. Der Fokus liegt auf den Wissensbedarfen, welche mit Machine Learning bedient werden können. Dabei werden Anwendungsfälle von Machine Learning aus der Literatur und Praxis betrachtet, zu Wissensbedarfen geclustert und dem Entscheidungsfindungsprozess zur Analyse und Anpassung komplexer Produktportfolios zugeordnet. Dies ermöglicht eine systematische Unterstützung der einzelnen Entscheidungsphasen. Es ist nicht beabsichtigt, eine vollständige Darstellung aller möglichen Anwendungsfälle von Machine Learning zu geben. Vielmehr wird eine Momentaufnahme aus der Literatur und der Industrie dargestellt, welche im Laufe der Zeit veränderlich ist.

Forschungsfrage 2: Welche Daten sind für die Generierung von Wissen zur Analyse komplexer Produktportfolios notwendig?

Forschungsfrage 2 verfolgt die Ermittlung und Analyse von Datenbedarfen über komplexe Produktportfolios. Es sind Datenbedarfe durch die Betrachtung der zuvor ermittelten Anwendungsfälle und Wissensbedarfe zu identifizieren, zu beschreiben und hinsichtlich der Datencharakteristika zu untersuchen. Dadurch wird eine anschließende Datenvorbereitung und die Umsetzung spezifischer Anwendungsfälle von Machine Learning zur Analyse komplexer Produktportfolios ermöglicht.

Forschungsfrage 3: Wie kann mittels Machine Learning Verfahren Wissen für die Analyse komplexer Produktportfolios generiert und eingesetzt werden?

Forschungsfrage 3 fokussiert die systematische Generierung von Wissen aus den zuvor ermittelten Datenbedarfen zur Analyse komplexer Produktportfolios. Dabei werden Informationen für die Datenvorbereitung und Modellierung in Abhängigkeit der Machine Learning Verfahren im spezifischen Kontext komplexer Produktportfolios betrachtet. Der Fokus liegt auf den überwachten Lernverfahren Regression und Klassifikation sowie den unüberwachten Lernverfahren Clustering und Assoziation.

1.4 Aufbau der Arbeit

Im Folgenden wird auf die Inhalte der einzelnen Kapitel eingegangen. Eine Übersicht wird zudem mit Abbildung 1-3 gegeben.

Kapitel 1 beschreibt die Motivation der Arbeit und untermauert diese durch eine Industrieperspektive. Aus dieser leitet sich die Zielsetzung sowie drei Forschungsfragen ab. Des Weiteren wird ein Überblick über den Aufbau der Arbeit gegeben.

In **Kapitel 2** werden die begrifflichen und methodischen Grundlagen erläutert, welche für das Verständnis der Arbeit erforderlich sind. Es werden die Grundlagen komplexer Produktportfolios sowie die des Machine Learning dargelegt. Diese beinhalten insbesondere die Definitionen der zentralen Begrifflichkeiten, die Betrachtung der Ent-

scheidungsfindung und Wissensgenerierung sowie die Vorstellung der grundlegenden Verfahren und Algorithmen.

Kapitel 3 behandelt den aktuellen Stand der Forschung beim Einsatz von Machine Learning zur Analyse komplexer Produktportfolios. Dafür werden zuerst Kriterien zur Bewertung bisherigen Ansätze definiert. Anschließend werden die Ansätze in Abhängigkeit der eingesetzten Machine Learning Verfahren beschrieben und anhand der zuvor definierten Bewertungskriterien gegenübergestellt sowie der Forschungsbedarf für dieses Promotionsvorhaben abgeleitet.

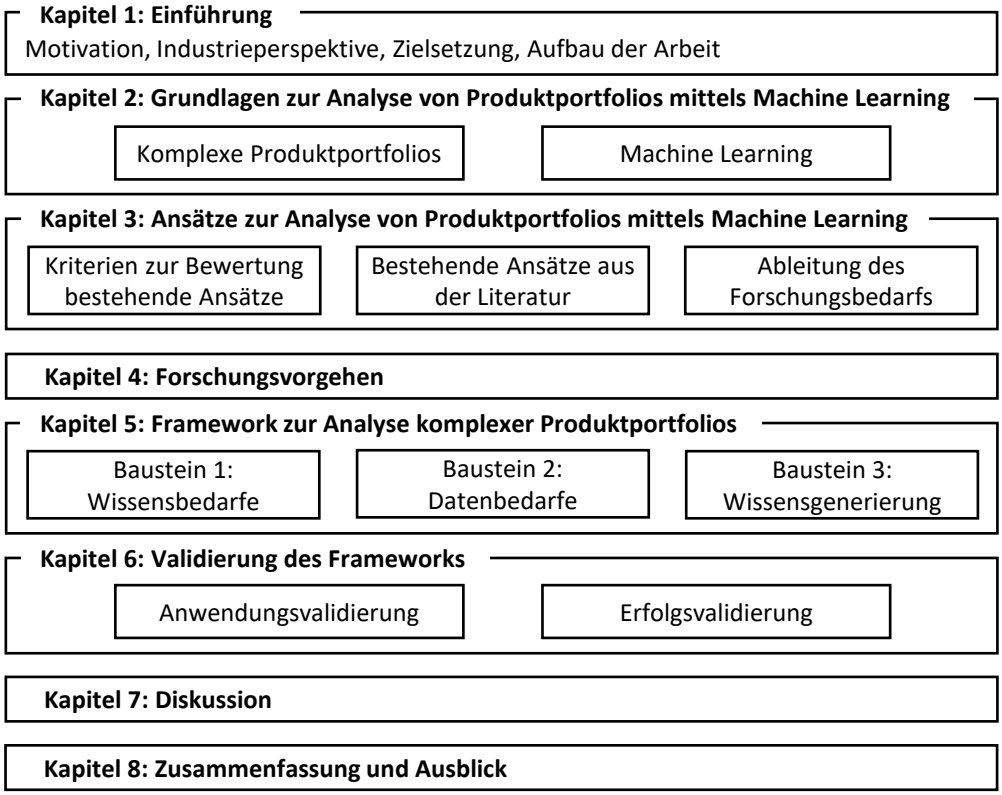


Abbildung 1-3: Aufbau der Arbeit

In **Kapitel 4** wird die übergeordnete Forschungsmethodik, welche für die Erstellung dieser Arbeit verfolgt wird, vorgestellt. Diese orientiert sich an der Design Research Methodology (DRM) von Blessing und Chakrabarti (2009). Daneben wird auf den Methodeneinsatz in den einzelnen Phasen der DRM eingegangen sowie Kriterien für die Validierung des zu entwickelnden Frameworks hergeleitet.

Kapitel 5 beinhaltet das Framework zur systematischen Generierung von Wissen für die Analyse komplexer Produktportfolios. Dieses orientiert sich an dem Datenanalyseprozess CRISP-DM von Wirth und Hipp (2000). Zuerst wird ein Verständnis für die Wissensbedarfe zur Analyse und Anpassung komplexer Produktportfolios generiert. Anschließend wird ein Verständnis für die Daten in der Anwendungsdomäne bereit-

gestellt. Eine systematische Datenvorbereitung, Modellierung, Evaluation und Einsatz von Machine Learning in Abhängigkeit des eingesetzten Verfahrens im Produktportfolio- und Variantenmanagement schließen das Framework ab.

In **Kapitel 6** findet die Validierung des Frameworks statt. Das Framework wird in einem Industrieunternehmen der Nutzfahrzeugbranche im Rahmen einer Fallstudie angewendet. Anschließend bewerten Experten das Framework und die Ergebnisse der Anwendung hinsichtlich der zuvor definierten Validierungskriterien.

Kapitel 7 beinhaltet eine kritische Diskussion der Ergebnisse dieses Forschungsvorhabens. Darin werden zuerst der Nutzen und die Einschränkungen betrachtet und anschließend auf den Beitrag aus Sicht der Forschung und Industrie eingegangen.

In **Kapitel 8** werden die Ergebnisse der Arbeit zusammengefasst und ein Ausblick für zukünftige Forschungstätigkeiten gegeben.

2 Grundlagen zur Analyse von Produktportfolios mittels Machine Learning

In diesem Kapitel wird auf die begrifflichen und methodischen Grundlagen zum Umgang mit komplexen Produktportfolios und Machine Learning eingegangen. Bevor anschließend in Kapitel 3 bisherige Ansätze aus der Literatur zur Analyse von Produktportfolios mittels Machine Learning vorgestellt werden. Die begrifflichen und methodischen Grundlagen bilden den Ausgangspunkt für das Verständnis der bisherigen Ansätze sowie die Entwicklung des Frameworks zur systematischen Wissensgenerierung für die Analyse komplexer Produktportfolios mittels Machine Learning.

2.1 Komplexe Produktportfolios

Zuerst werden grundlegende Begrifflichkeiten für den Umgang mit komplexen Produktportfolios erläutert sowie Ansätze zur Handhabung der varianteninduzierten Komplexität vorgestellt. Anschließend wird ein typischer Entscheidungsprozess zur Analyse und Anpassung von Produktportfolios betrachtet.

2.1.1 Terminologie komplexer Produktportfolios

Das **Produktportfolio** oder Produktprogramm bezieht sich auf die Gesamtheit aller Produkte und/oder Dienstleistungen, die ein Unternehmen auf dem Markt anbietet (Jonas 2013). Das Produktportfolio wird durch die Portfoliobreite und -tiefe charakterisiert (Baumberger 2007). Die **Portfoliobreite** beschreibt die Anzahl an unterschiedlichen Produktfamilien und die **Portfoliotiefe** spezifiziert die Menge der Produktvarianten innerhalb einer Produktfamilie (Lingnau 1994) (siehe Abbildung 2-1). Eine **Produktfamilie** ist eine Auswahl von ähnlichen Produkten, die auf einer gemeinsamen Produktplattform entwickelt werden und jeweils besondere Funktionalitäten aufweisen, um spezielle Kundenanforderungen zu erfüllen (Meyer und Lehnerd 1997). Das Produktportfolio besteht in der Regel aus mehreren solcher Produktfamilien (Blees 2011). Eine **Produktplattform** ist dabei ein Satz gemeinsamer Komponenten, Module oder Teile, aus denen Produktvarianten effizient abgeleitet und auf den Markt gebracht werden können (Meyer und Lehnerd 1997). Die einzelnen Vertreter einer Produktfamilie werden als Produktvarianten bezeichnet (Dellanoi 2006). Nach der Norm DIN 199-1 sind **Produktvarianten** „Gegenstände ähnlicher Form oder Funktion mit einem in der Regel hohen Anteil identischer Gruppen oder Teile“ (DIN 199-1 2002). Die Produktvarianten unterscheiden sich in mindestens einer Ausprägung einer kundenrelevanten Eigenschaft (Franke und Firchau 2000).

Die **Produktvielfalt** charakterisiert sowohl die Anzahl als auch den Unterschied zwischen den Produktvarianten (Gembrys 1998). Hierbei muss zwischen der externen und internen Vielfalt unterschieden werden (Gebhardt et al. 2016). Die **externe**

Produktvielfalt beschreibt den Umfang der am Markt angebotenen Produktvarianten und die **interne Produktvielfalt** stellt die Varianz an Bauteilen und -gruppen, Produkten und Prozessen zur Erzeugung der externen Vielfalt dar (Bartuschat 1995).

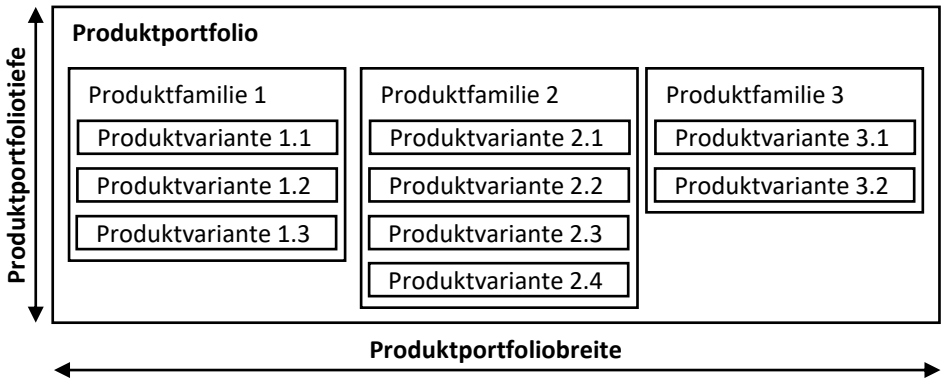


Abbildung 2-1: Gliederung des Produktportfolios in Produktfamilien und Produktvarianten in Anlehnung an Kieckhäfer (2013)

Die Produktvielfalt nimmt in den meisten Unternehmen aufgrund externer und interner Ursachen kontinuierlich zu (Ehrlenspiel et al. 1998; Gebhardt et al. 2016). **Externe Ursachen** sind zum Beispiel die Globalisierung, zunehmender Wettbewerbsdruck und verkürzte Produktlebenszyklen (Kesper 2012). **Interne Ursachen** hingegen resultieren aus organisatorischen oder technischen Defiziten im Unternehmen (ElMaraghy et al. 2013) sowie den Erwartungen an den Nutzen einer hohen Produktvielfalt (Heina 1999). Eine Steigerung der Produktvielfalt hat meist die Gewinnung neuer Kunden zum Ziel (Rathnow 1993).

Die Erhöhung der externen Vielfalt erzeugt in den meisten Fällen eine Zunahme der internen Vielfalt und dadurch eine Komplexitätssteigerung in allen Bereichen eines Unternehmens (Gebhardt et al. 2016). Unter **Komplexität** wird in der Systemtheorie die Anzahl, die Vielfalt und die Beziehungen der Elemente sowie deren Zustände und Veränderlichkeit verstanden (Krause und Gebhardt 2018). **Komplexe Produktportfolios** ergeben sich aus der Anzahl, Vielfalt und der zeitlichen Veränderlichkeit der Merkmale, Merkmalsausprägungen, Komponenten, Komponentenvarianten, deren Beziehungen ausgedrückt durch die Booleschen Regelwerke sowie die resultierenden Produktkonfigurationen (Mehlstäubl et al. 2023b). Die **varianteninduzierte Komplexität** beschreibt die Anteile der unternehmensinternen Komplexität, die durch die Produktvielfalt mit ihrer Anzahl und Verschiedenheit varianter Objekte, wie Komponenten, Produkte oder Prozesse, entstehen (siehe Abdelkafi 2008). Beispiele für Produktportfolios mit einer hohen varianteninduzierten Komplexität sind die von Automobil- oder Nutzfahrzeughersteller mit hunderten oder tausenden von wählbaren Ausstattungsmerkmalen (siehe Greisel et al. 2013).

Aus der varianteninduzierten Komplexität der Produkte und Prozesse ergeben sich Kosten, die sogenannten **Komplexitätskosten** (Ehrlenspiel et al. 1998; Mariotti 2007). Hierzu gehören zum Beispiel die Kosten aufgrund des erhöhten Konstruktions-, Verwaltungs-, Dokumentations-, Schulungs-, Lager- oder Logistikaufwands (Gebhardt et al. 2016). Der Großteil dieser Kostenwirkungen betrifft die Gemeinkosten eines Unternehmens, welche auf alle Produkte umgelegt werden (Kesper 2012). Der Pfeil in Abbildung 2-2 zeigt auf, dass die Standardprodukte die Kosten exotischer Produktvarianten mit geringeren Stückzahlen oder geringerer Profitabilität tragen und so diese quersubventionieren (Schuh 2005). Zusätzlichen Produktvarianten erscheinen so profitabler als sie es tatsächlich sind.

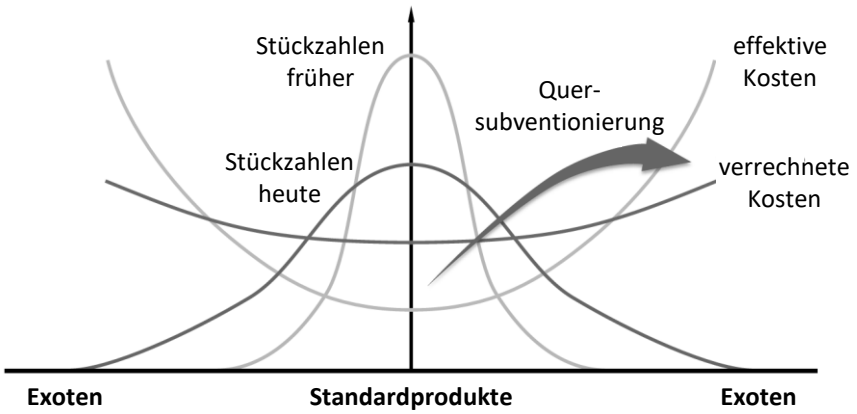


Abbildung 2-2: Quersubventionierung von exotischen Produktvarianten in Anlehnung an Schuh und Schwenk (2001)

Für die Gestaltung und Handhabung der externen und internen Produktvielfalt sowie der daraus resultierenden Komplexität in einem Unternehmen sind die Disziplinen des Produktportfolio- und Variantenmanagements erforderlich. Das Portfoliomanagement hat seinen Ursprung in den 1950er-Jahren im Kontext von Anlageentscheidungen am Kapitalmarkt und verfolgt das Ziel, eine Zusammenstellung von Wertpapieranlagen zu finden, die bei begrenzten Ressourcen ein Optimum an Ertrag und Risiko darstellen (Marx 1996). Das **Produktportfoliomanagement** beschreibt die Anwendung von Verfahren und Modellen des Portfoliomanagements im Rahmen der Entwicklung von Produkten (siehe Cardozo und Smith 1983). Es umfasst den dynamischen Entscheidungsprozess, in dem die aktiven Produktprojekte eines Unternehmens ständig aktualisiert und überprüft werden (Cooper et al. 2000). In diesem Prozess werden neue Produktprojekte bewertet, ausgewählt und priorisiert, bestehende Projekte können beschleunigt, gestoppt oder verlangsamt werden (Cooper et al. 2000). Der Entscheidungsprozess ist dabei von hoher Komplexität geprägt, da er eine wechselnde Informationsgrundlage, immer neue Chancen, verschiedene Zielstellungen und Wechselwirkungen zwischen Projekten berücksichtigen muss und mehrere Entscheidungsträger beinhaltet (Cooper et al. 2000).

Das Produktportfoliomanagement kann weiter in die Bereiche Marktanalyse, Programmplanung und Zukunftsplanung unterteilt werden (Krause und Gebhardt 2018) (siehe Abbildung 2-3). Die **Marktanalyse** befasst sich sowohl mit den Strukturen des Marktes als auch mit der Marktsituation des Produktportfolios eines Unternehmens. Die Methoden der **Programmplanung** zielen darauf ab, die technischen Eigenschaften des Produktportfolios an die Bedürfnisse des Marktes auszurichten. Die **Zukunftsplanung** untersucht Trends im Markt und deren Auswirkungen auf die Entwicklung des Produktportfolios.

Das **Variantenmanagement** umfasst alle Maßnahmen zur Steuerung des Angebots an Produktvarianten eines Unternehmens und zur Bewältigung der daraus resultierenden Effekte über den gesamten Produktlebenszyklus (ElMaraghy et al. 2013). Es lässt sich in die Aktivitäten der Variantengenerierung, -vermeidung, -steuerung und -reduktion unterteilen (Heina 1999) (siehe Abbildung 2-3). Die **Variantengenerierung** hat das Ziel, die geforderten Kundenbedürfnisse in die richtigen Produkteigenschaften zu übersetzen und die Varianten marktgerecht zu definieren. Bei der **Variantenvermeidung** wird versucht, die interne Vielfalt durch geeignete Strukturierungskonzepte wie Modularisierung oder Plattformstrategien gering zu halten. Auf die Strukturierung von Produktportfolios wird im nachfolgenden Kapitel 2.1.2 eingegangen. Ziel der **Variantenbeherrschung** ist es, die entstehende interne Vielfalt und die daraus resultierende Komplexität effizient durch die Geschäftsprozesse zu steuern. Die **Variantenreduktion** beinhaltet die Eliminierung von nicht nachgefragten oder unrentablen Varianten.

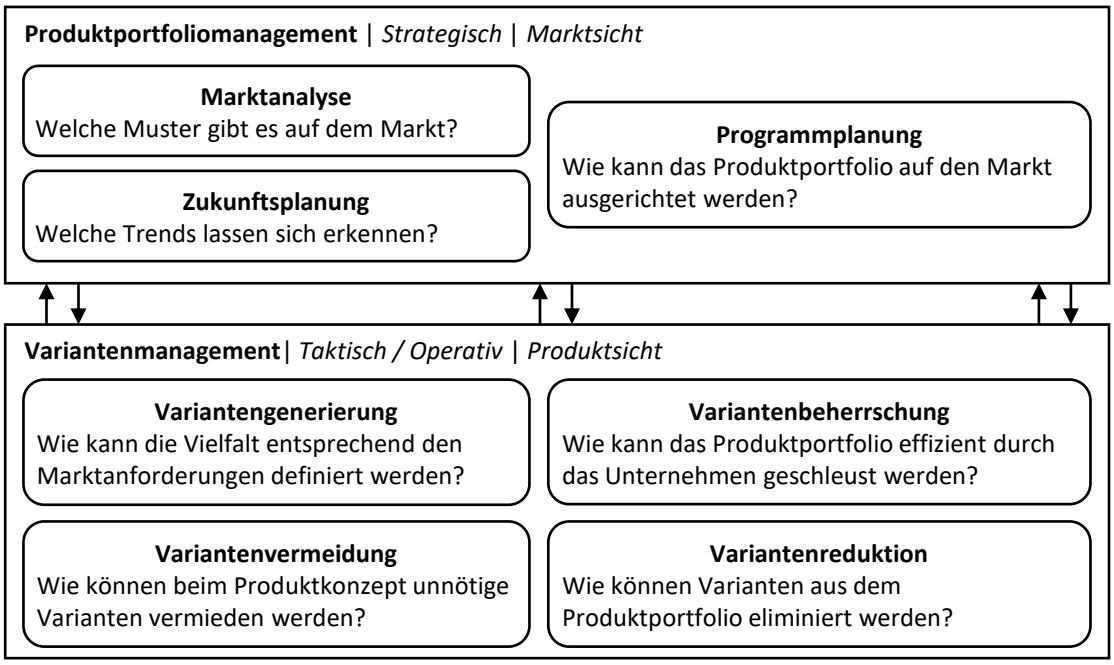


Abbildung 2-3: Produktportfolio- und Variantenmanagement nach Mehlstäubl et al. (2021b)

Das Produktportfoliomanagement hat einen strategischen Charakter mit einem mittel- bis langfristigen Entscheidungshorizont, während das Variantenmanagement die varianteninduzierte Komplexität operativ entschärft und moderiert (ElMaraghy et al. 2013). Der Übergang zwischen den beiden Disziplinen kann als fließend betrachtet werden. Beide Disziplinen werden für den effizienten und effektiven Umgang mit einem komplexen und variantenreichen Produktportfolio benötigt.

2.1.2 Strukturierung komplexer Produktportfolios

Unternehmen verfolgen unterschiedliche Strategien, um der von den Kunden nachgefragten Vielfalt gerecht zu werden und dem beschriebenen Problem der steigenden Anzahl an Varianten und Komplexität entgegenzuwirken (Schmieder und Thomas 2005). Ein wesentlicher Stellhebel zur Beherrschung der Variantenvielfalt ist eine variantengerechte Produktarchitektur (Feldhusen und Grote 2013). Das Ziel der **Produktarchitektur** ist es durch Standardisierung, Modularität und Wiederverwendung von Komponenten die Handhabung der Komplexität zu unterstützen (Förg et al. 2016). Die Produktarchitektur beinhaltet die Anordnung der funktionalen Elemente in der Funktionsstruktur, die Spezifikation der Schnittstellen zwischen den physischen Komponenten in der Produktstruktur sowie die Verknüpfung der funktionalen Elemente mit den physischen Komponenten (Ulrich 1995) (siehe Abbildung 2-4). Daneben sind die Kombinierbarkeit der marktseitigen und kundenrelevanten Merkmale in der Produktarchitektur festgelegt (Förg et al. 2016).

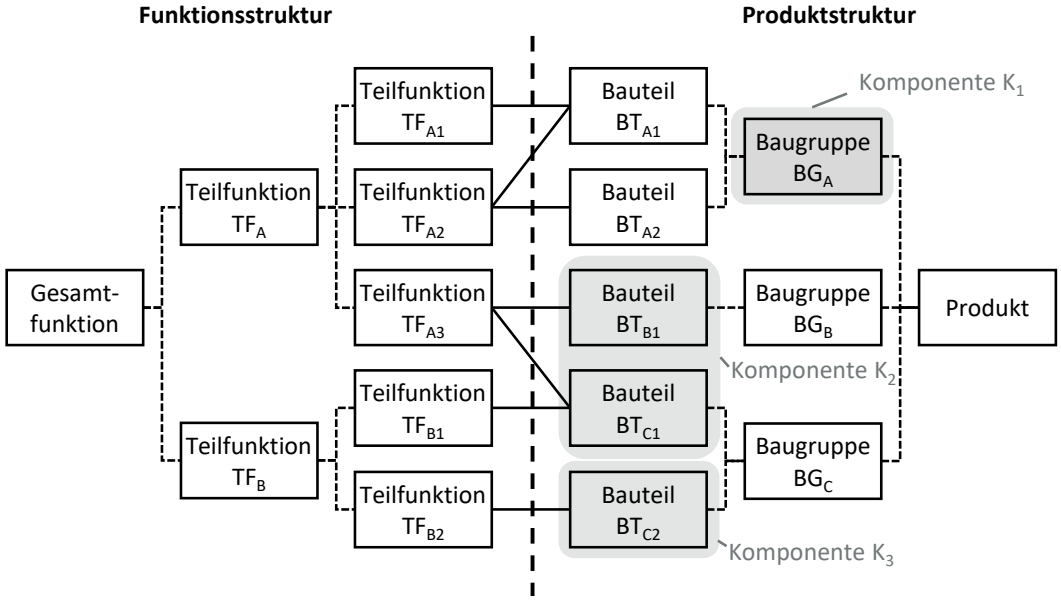


Abbildung 2-4: Produktarchitektur nach Göpfert (1998) und Krause et al. (2021)

Ein Architekturansatz für die effiziente Gestaltung variantenreicher Produktportfolios ist die **Baukastenarchitektur** (Jeschke 1997). Ein Baukasten ist eine Menge von Bausteinen, die spezifisch ausgewählt und unter Beachtung der Verträglichkeit miteinander kombiniert werden, um unterschiedliche Produktvarianten zu konfigurieren (Baumgart 2005) (siehe Abbildung 2-5). Eine Baukastenarchitektur ermöglicht es, Skaleneffekte entlang der gesamten Wertschöpfungskette durch Kommunalitäten zwischen Produkten und Produktfamilien zu erzielen (Schuh 2012) und dadurch eine hohe Produktvielfalt mit vergleichsweise geringem Entwicklungsaufwand zu realisieren (Feldhusen und Grote 2013).

Die Bausteine oder **Komponenten** (z. B. Zylinderkopf) mit ihren Varianten, den sogenannten **Komponentenvarianten** (z. B. Zylinderkopf für vier Zylinder), sind vorausgedachte technische Lösungen des Baukastensystems, die unterschiedliche Funktionen und Eigenschaften realisieren und alle Hardware- und Softwarekomponenten des Endprodukts umfassen (Braun 2021). Meist handelt es sich dabei um **Module**, welche funktional und physisch relativ unabhängige Einheiten darstellen, damit sie einfach ausgetauscht und unterschiedlich miteinander kombiniert werden können, um Produktvarianten zu erzeugen (Huang und Kusiak 1998). Die Modularität der Komponenten und Komponentenvarianten ist die Grundlage dafür, dass diese weitestgehend unabhängig voneinander entwickelt und hergestellt sowie in unterschiedlichen Produktfamilien eingesetzt oder ausgetauscht werden können (Krause et al. 2021). Die Konfiguration von Baukastenprodukten erfolgt nach dem Muster und den definierten Freiheitsgraden und Restriktionen von Architekturvorgaben (Kreimeyer et al. 2013a).

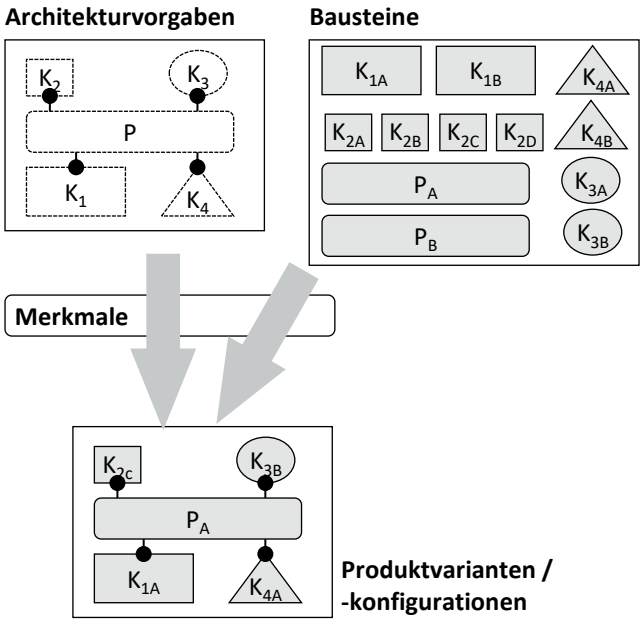


Abbildung 2-5: Baukastenarchitektur nach Kreimeyer et al. (2013b)

Die Gültigkeit von Architekturvorgaben und Bausteinen im konkreten Auftragsfall eines Produkts wird mit Hilfe von **Merkmalen** (z. B. Kabine) und **Merkmalsausprägungen** (z. B. große Kabine oder kleine Kabine), welche vom Kunden ausgewählt werden, gesteuert (Kreimeyer et al. 2013a). Sie bilden die Soll-Eigenschaften des Produktes aus Kundensicht ab. In der industriellen Praxis können dies sowohl indirekte als auch direkte Produkteigenschaften sein (Braun 2021). **Direkte Produkteigenschaften** (engl. characteristics) können von den Entwicklern direkt beeinflusst oder bestimmt werden (z. B. Struktur, Form und Material) (Weber 2005). Die **indirekten Produkteigenschaften** (engl. properties) beschreiben das Verhalten des Produkts (z. B. Leistung, Sicherheit und Zuverlässigkeit) und können nur indirekt über die direkten Eigenschaften gesteuert werden (Weber 2005). Die Konfiguration einer Produktvariante ist eine besondere Art der Entwicklungstätigkeit, bei der die Produktvariante oder **Produktkonfiguration** aus den vordefinierten Bausteinen des Baukastens unter Berücksichtigung der definierten Einschränkungen zusammengesetzt wird (Mittal und Frayman 1989). Die Konfiguration von Produktvarianten erfolgt heutzutage durch Produktkonfiguratoren (Rapp 1999).

2.1.3 Analyse und Anpassung komplexer Produktportfolios

Die Entscheidungen im Produktportfolio- und Variantenmanagement lassen sich in zwei Gliederungsebenen einteilen. Diese sind die Produktportfolio- und Produktgestaltung (Meffert et al. 2015) (siehe Abbildung 2-6). Die **Produktportfoliogestaltung** kann weiterhin in die strategische und operative Produktportfoliogestaltung unterteilt werden (Kieckhäfer 2013). Die **strategische Produktportfoliogestaltung** trifft Entscheidungen über Innovationen, Variation, Differenzierung und Elimination von Produktlinien. Die **operative Produktportfoliogestaltung** untersucht dagegen die genannten Aspekte auf der Ebene von Produktvarianten. Die **Produktgestaltung** beinhaltet die Umsetzung der getroffenen Entscheidungen am technischen Produkt. Fokus dieser Arbeit liegt auf der Analyse von Produktvarianten auf der Ebene der operativen Produktportfoliogestaltung.

Das Produktportfolio eines Unternehmens ist im stetigen Wandel und es müssen ständig Entscheidungen zur Modifikation, Differenzierung und Elimination getroffen werden (Meffert et al. 2015). Im herkömmlichen Ansatz der Entscheidungsfindung werden die einzelnen Produktvarianten bewertet. Die große Anzahl möglicher Produktvarianten, welche sich durch die Kombination der Vielzahl an Merkmalen in einem Baukastensystem ergeben, kann jedoch nicht effizient analysiert werden (Heina 1999). Komplexe Produktportfolios können hunderte oder sogar tausende Merkmale aufweisen, die von den Kunden ausgewählt und miteinander kombiniert werden können (siehe Greisel et al. 2013). Außerdem führt die Entfernung von Produktvarianten zu einer Verringerung der möglichen Kombinationen von Merkmalsausprägungen und nicht zwingend zur Reduktion von Merkmalsausprägungen oder Komponentenvarianten

mit den assoziierten Kosten (siehe Abbildung 2-7). Aus diesem Grund ist heute vor allem das Wissen über Merkmale und Komponenten entscheidungsrelevant.

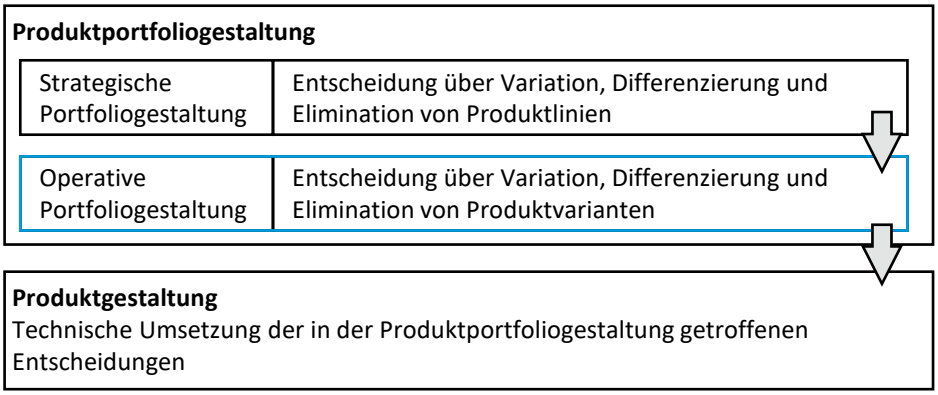


Abbildung 2-6: Entscheidungsebenen und -gegenstände in Anlehnung an Kieckhäfer (2013) und Meffert et al. (2015)

Ein Großteil der geschaffenen Produktvielfalt in Unternehmen ist nicht profitabel (Bannasch und Bouché 2016), weshalb Projekte zur Rationalisierung der Produktvielfalt initiiert werden (Darrell 2017; Hirose et al. 2017). In der Literatur existieren mehrere Methoden zur Reduktion des Produktportfolios (siehe Wildemann 2011). Diese beinhalten zum Beispiel die Einteilung der Produktvarianten in A-, B- und C-Kategorien auf der Grundlage ihres Umsatzes und Deckungsbeitrags, woraufhin Initiativen zur Bereinigung unrentabler C-Produkte festgelegt werden (Hvam et al. 2020).

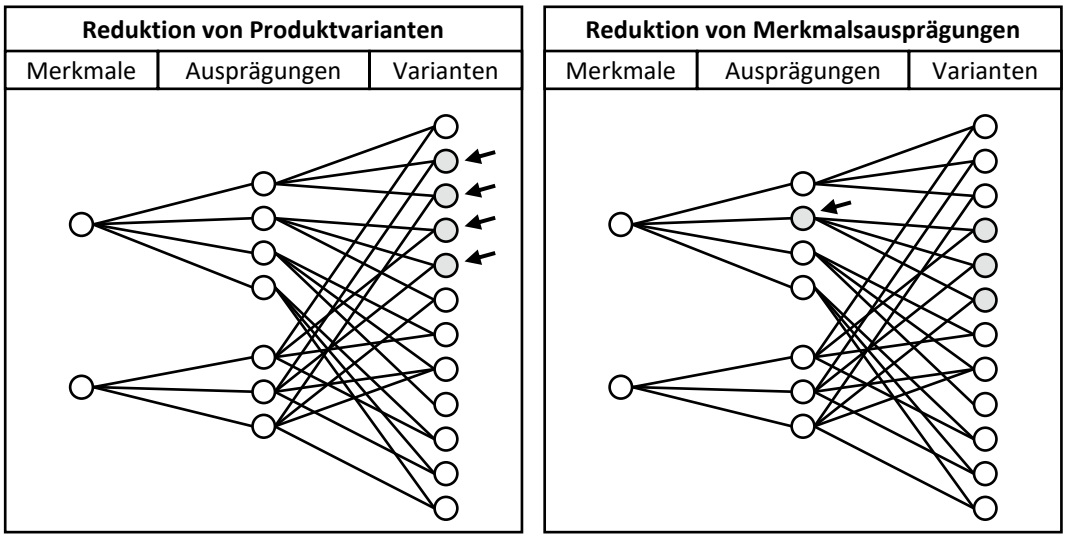


Abbildung 2-7: Betrachtungsgegenstände der Entscheidungsfindung nach Heina (1999)

In Abbildung 2-8 ist der **Entscheidungsprozess** zur Analyse und Anpassung von Produktportfolios in Anlehnung an Vogel (1989) und Gembrys (1998) visualisiert.

Heutzutage ist dieser Prozess geprägt durch manuelle und erfahrungsbasierte Tätigkeiten (Mehlstäubl et al. 2023a). Er wird ausgelöst, wenn eine Soll-Ist-Abweichung im Hinblick auf die Zusammensetzung des Produktportfolios festgestellt wird. Dies kann entweder spontan oder durch die Überwachung von definierten Zielen geschehen. Die Analyse des Produktportfolios beginnt mit der Suche und Auswahl der wesentlichen Informationen und deren Darstellung. Das Sammeln der Informationen besteht meist aus der Zuordnung von verkauften Stückzahlen zu Merkmalen und Komponenten. Teilweise werden Kostenkalkulationen in die Betrachtungen aufgenommen. Im nächsten Schritt werden mögliche Alternativen zum aktuellen Produktportfolio identifiziert, mit denen die Soll-Ist-Abweichung reduziert oder beseitigt werden soll. Ziel der Prognosephase ist es, die Auswirkungen der Veränderungen des Produktportfolios auf das Unternehmen und den Markt vorherzusagen. Dies bildet die Grundlage für die Bewertung und Auswahl von Alternativen in der letzten Phase des Entscheidungsprozesses. In einem Gremium werden die Vorschläge evaluiert und eine Entscheidung getroffen.

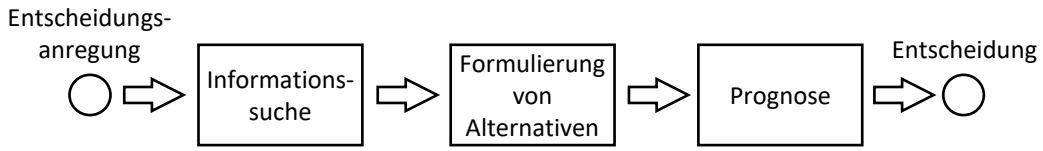


Abbildung 2-8: Entscheidungsprozess zur Analyse und Anpassung von Produktportfolios in Anlehnung an Vogel (1989), Gembrys (1998) und Mehlstäubl et al. (2023a)

2.1.4 Zusammenfassung: Komplexe Produktportfolios

Um ein Produktportfolio anbieten zu können, das optimal auf den Markt abgestimmt ist, erhöhen die Unternehmen ihre externe Angebotsvielfalt. Dies sorgt für eine Zunahme der internen Vielfalt und dadurch zu einer Steigerung der varianteninduzierten Komplexität in den Produkten und Prozessen. Die dadurch entstehenden Komplexitätskosten sorgen für einen Wettbewerbsnachteil. Für die effiziente Gestaltung komplexer und variantenreicher Produktportfolios werden unterschiedliche Architekturansätze wie die Baukastenarchitektur eingesetzt. Diese ermöglichen es mit vorgegebenen Merkmalen und Merkmalsausprägungen eine Vielzahl an unterschiedlichen Produktvarianten zu konfigurieren. Das Produktportfolio- und Variantenmanagement hat die Aufgabe das Produktportfolio zu definieren und die resultierenden Effekte über den gesamten Produktlebenszyklus zu steuern. Um die volatilen Kundenbedürfnisse zu befriedigen sowie die Rentabilität hochzuhalten, ist heute eine dynamische Analyse und Anpassung des Produktportfolios im Rahmen eines Entscheidungsprozesses notwendig. Ein solcher Entscheidungsprozess besteht aus den Phasen der Informationssuche, der Formulierung von alternativen Ausprägungen sowie der Prognose der Auswirkungen, welche sich aus den Anpassungen ergeben. Fokus dieser Arbeit liegt auf

der operativen Produktportfoliogestaltung, welche Entscheidungen über die Variation, Differenzierung und Elimination von Produktvarianten trifft.

2.2 Machine Learning

Im folgenden Abschnitt wird auf die Machine Learning Grundlagen eingegangen. Diese werden im späteren Verlauf der Forschungsarbeit im Framework zur Analyse komplexer Produktportfolios aufgegriffen und sind für dessen Verständnis sowie Nutzung erforderlich. Im Folgenden wird Machine Learning zuerst in den Kontext der künstlichen Intelligenz eingebettet und eine Abgrenzung zu anderen Begrifflichkeiten vorgenommen. Anschließend wird auf das Lernen aus Daten und die Wissensgenerierung mit Machine Learning eingegangen. Abschließend werden die wichtigsten Machine Learning Verfahren und deren Algorithmen vorgestellt.

2.2.1 Machine Learning als Teil der künstlichen Intelligenz

Künstliche Intelligenz (KI) ist heutzutage in aller Munde und wird im industriellen Kontext oft als Synonym für Machine Learning verwendet. Jedoch ist Machine Learning nur ein kleines Teilgebiet der künstlichen Intelligenz (Helm et al. 2020) (siehe Abbildung 2-9). In der Literatur existiert keine einheitliche Definition von KI. McCarthy (1955) definiert künstliche Intelligenz mit dem Ziel Maschinen zu entwickeln, die sich so verhalten, als ob sie intelligent wären (Ertel 2011). Andere Definitionen dagegen sind durch den spezifischen Kontext geprägt. Goodfellow et al. (2012) beschreiben KI im Kontext von Deep Learning und der Bilderkennung. Für sie besteht die Aufgabe von KI darin, Probleme zu lösen, die für Menschen leicht zu bewältigen, aber schwer formal zu beschreiben sind.

Für den industriellen Einsatz von KI zur Analyse komplexer Produktportfolios sind die eingesetzten Technologien und die daraus resultierenden Möglichkeiten von Bedeutung. Eine Definition, welche diesem Anspruch gerecht wird, ist die von Gartner. KI wendet Advanced Analytics und regelbasierte Techniken an, um Ereignisse zu interpretieren, Entscheidungen zu unterstützen und zu automatisieren sowie Maßnahmen zu ergreifen (Gartner 2022). Regelbasierte Techniken, sogenannte Expertensysteme, „schlussfolgern bei gegebener Anfrage (Ziel oder Hypothese) aus einer Menge von Regeln (Wissensbasis) und Fakten auf neue Aussagen („Konklusionen“) und kommen dadurch zu Empfehlungen (Müller und Lenz 2013). Der manuelle Prozess der Regeldefinition für die Entwicklung von Expertensystemen ist zeitaufwändig und erfordert ein hohes Maß an Fachwissen (Haug et al. 2012).

Der Begriff **Advanced Analytics** ist ebenfalls in der Literatur nicht eindeutig beschrieben (Schuh et al. 2018). Es handelt sich jedoch um ein Bündel übergeordneter Data Mining und Machine Learning Verfahren, mit denen aktuelle und historische Daten effizient analysiert werden können (Sherman 2014). **Data Mining** ist die Anwendung

spezifischer Algorithmen zur Extraktion von Mustern aus Daten (Fayyad et al. 1996). Die meisten Data Mining Methoden basieren auf bewährten Techniken aus dem Machine Learning, der Mustererkennung und der Statistik (Fayyad et al. 1996).

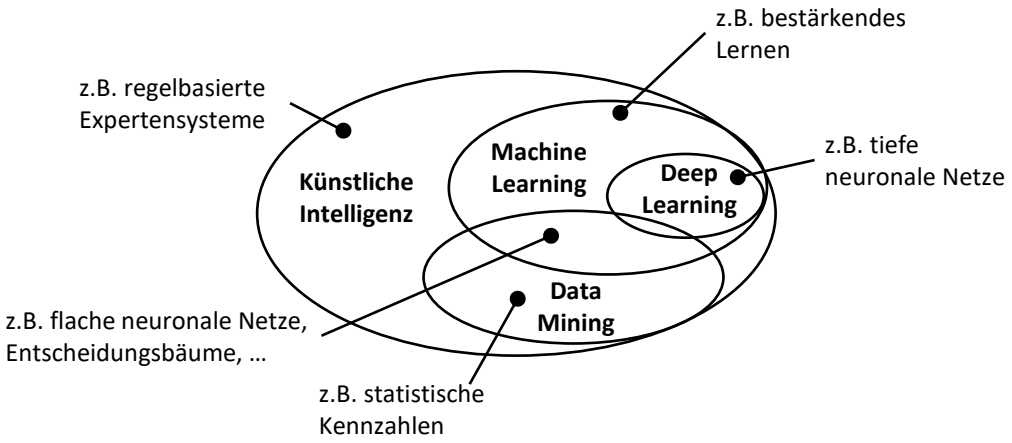


Abbildung 2-9: Zusammenhang künstliche Intelligenz, Data Mining und Machine Learning in Anlehnung an Stack Exchange (2020)

Machine Learning ist das Fachgebiet, das Computern die Fähigkeit verleiht, zu lernen, ohne ausdrücklich programmiert zu werden (Samuel 1959). Wobei ein Computerprogramm aus Erfahrung E in Bezug auf eine Aufgabe T und ein Leistungsmaß P lernt, wenn sich seine Leistung T, gemessen an dem Leistungsmaß P, mit der Erfahrung E verbessert (Mitchell 1997). Im Lernprozess wird auf der Grundlage der Daten ein statistisches Modell entwickelt, welches anschließend zur Lösung der Aufgabe eingesetzt wird (Burkov 2019). Goodfellow et al. (2012) beschreiben Machine Learning als Verfahren, welche in der Lage sind, sich eigenes Wissen durch die Extraktion von Mustern aus Daten anzueignen. Murphy (2012) erweitert die Definition um die Nutzung der Muster für die Vorhersage und die Entscheidungsunterstützung. Er definiert Machine Learning als eine Reihe von Verfahren, die automatisiert Muster in Daten erkennen und dann die aufgedeckten Muster nutzen, um zukünftige Ereignisse vorherzusagen oder andere Arten von Entscheidungen unter Unsicherheit zu treffen. **Deep Learning** ist ein Teil des Machine Learning, wobei tiefe neuronale Netze eingesetzt werden, um komplexe Muster in großen unstrukturierten Datensätzen zu erkennen (siehe LeCun et al. 2015). Tiefe neuronale Netze besitzen eine Vielzahl an Neuronen und verborgene Schichten (Aggarwal 2018). Auf neuronale Netze und deren Aufbau wird in Kapitel 2.2.5 näher eingegangen.

2.2.2 Terminologie Machine Learning

Im Folgenden wird auf die Grundterminologie des Machine Learning eingegangen, welche in Abbildung 2-10 visualisiert ist. Beim Machine Learning geht es grundsätzlich

darum, aus Daten die **Eingangsmerkmale** (engl. Features) auszuwählen und zu nutzen, um **Modelle** zu erstellen und so eine definierte **Aufgabe** bestmöglich zu erfüllen (Flach 2012). Aufgaben stellen grundsätzliche Probleme dar, die im Hinblick auf eine gegebene Datenbasis zu lösen sind (Flach 2012). Für das Erfüllen der Aufgabe bzw. das Lösen des Problems können je nach Art unterschiedliche Machine Learning Verfahren wie beispielsweise eine Regression oder Klassifikation eingesetzt werden. Die Outputs dieser Modelle werden als **Zielvariablen** bezeichnet. Ihr Wert wird auf Basis der Eingangsmerkmale und der **Modellparameter** berechnet (Pereira und Borysov 2019). Durch die Auswahl der Eingangsmerkmale werden die Qualität der Modelle und deren Aussagekraft im Hinblick auf die Aufgabenstellung beeinflusst (Flach 2012). Die Bestimmung der Modellparameter geschieht in einem separaten Lernprozess, welcher als Training bezeichnet wird und der Verwendung des Modells vorausgeht.

Das **Training** besteht darin, dass mittels einer Teilmenge der gegebenen Datenbasis, den sogenannten Trainingsdaten, die Kostenfunktion eines Lernalgorithmus iterativ minimiert oder maximiert wird (Flach 2012). Die **Kostenfunktion** basiert auf der Differenz der tatsächlichen Werten in den Trainingsdaten und den korrespondierenden Vorhersagewerten des Modells (Pereira und Borysov 2019). Das Minimum oder Maximum wird mit einem **Optimierungsverfahren** wie zum Beispiel dem Gradientenabstiegsverfahren ermittelt (siehe Ruder 2016). Eine Übersicht von Optimierungsverfahren für Machine Learning kann Gambella et al. (2021) entnommen werden. Die Bewertung des resultierenden Modells hingegen erfolgt anhand statistischer Kriterien (Botchkarev 2018) und der Vorhersagen für noch nicht zuvor gesehenen Daten, den so genannten **Testdaten** (Mahesh 2020).

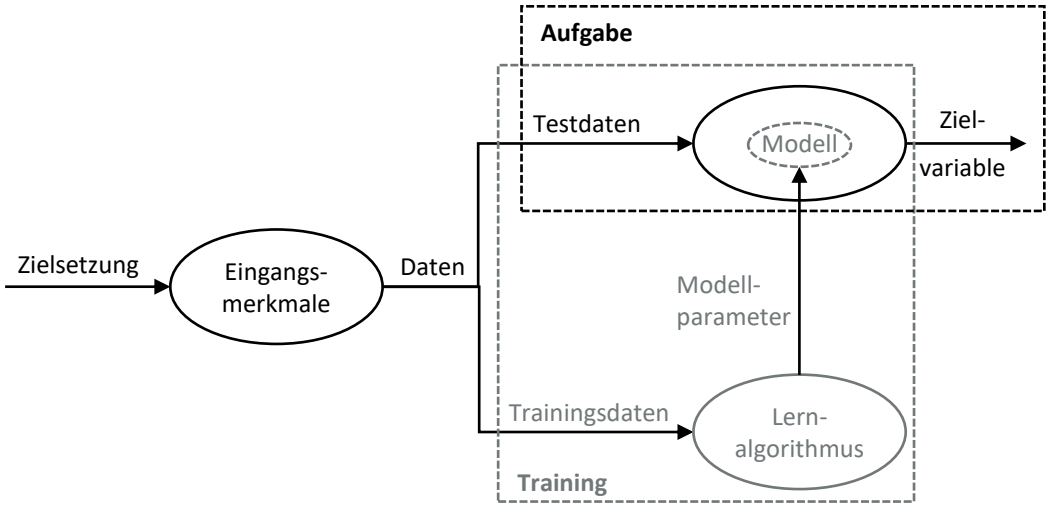


Abbildung 2-10: Terminologie Machine Learning in Anlehnung an Flach (2012)

2.2.3 Wissensgenerierung mit Machine Learning

Für die Definition von Wissen ist eine Abgrenzung zu den Begriffen Daten und Informationen erforderlich (VDI 5610 Blatt 1 2009). **Daten** sind Zeichen, welche eine gewisse Syntax besitzen, aber nicht interpretiert sind (z.B. die Zahl 25) (Probst et al. 2010). Aus Daten werden **Informationen**, wenn diese einen Bezug und damit eine Bedeutung besitzen (z.B. ein Lastkraftwagen besitzt eine zulässige Gesamtmasse von 25 t) (Binz et al. 2016)**. Wissen** sind Informationen, welche zu einem bestimmten Zweck miteinander vernetzt werden (z.B. ein Lastkraftwagen benötigt mindestens 3 Achsen, da er eine zulässige Gesamtmasse von 25 t besitzt) (North 2016). Wissen impliziert die Fähigkeit, Inputs mit Outputs zu verknüpfen, um Regelmäßigkeiten in Informationen zu beobachten, zu kodieren, zu erklären und schließlich vorherzusagen (Carnegie Bosch Institute (CBI) 1995). Dabei kann zwischen implizitem Wissen und explizitem Wissen unterschieden werden (Polanyi 1985; Nonaka und Takeuchi 1997). Implizites Wissen ist nicht formalisiert und an Personen gebunden. Durch Lernen werden implizites Wissen und neue Fähigkeiten erworben. Implizites Wissen kann mit Hilfe von Regeln formalisiert und explizit gemacht werden (Jiao und Zhang 2004).

Nach Mishra (2020) kann bei der Generierung von Wissen zwischen einem traditionellen Ansatz und einem Machine Learning Ansatz unterschieden werden. Géron (2017) beschreibt zudem die Möglichkeit, durch die systematische Analyse der Machine Learning Modelle zu lernen und dadurch Wissen zu generieren. Im **traditionellen Ansatz** definieren Experten auf Basis ihres Erfahrungswissens Regeln, welche die Zusammenhänge zwischen den Inputs und Outputs darlegen und explizit programmiert werden (Lee 2019). Dadurch wird das implizite Expertenwissen, welches sie durch die Interaktion mit ihrer Umwelt erlangt haben, explizit formalisiert (siehe Abbildung 2-11 a)). Als Ergebnis entstehen die in Kapitel 2.2.1 beschriebenen regelbasierten Expertensysteme. Der **Machine Learning Ansatz** zur Wissensgenerierung folgt einem ähnlichen Prinzip. Es werden jedoch anstatt der manuellen Definition von Regeln auf Basis des Erfahrungswissens der Experten, Daten mithilfe der Machine Learning Algorithmen analysiert (siehe Abbildung 2-11 b)). Bei der Wissensgenerierung mit Machine Learning werden Muster durch das Training von Lernalgorithmen aus den Daten gewonnen. Die Algorithmen analysieren große Datensätze und erkennen automatisiert Zusammenhänge und bilden diese auf unterschiedliche Weise in ihren Modellen ab. Durch eine Interpretation und Evaluation der Muster wird daraus Wissen (Fayyad et al. 1996). Daneben kann eine **Analyse der Machine Learning Modelle** zur Wissensgenerierung stattfinden. Die Parameter der trainierten Modelle und ihr Verhalten können untersucht werden, um die Zusammenhänge zu verstehen und neues Wissen zu gewinnen (siehe Abbildung 2-11 c)).

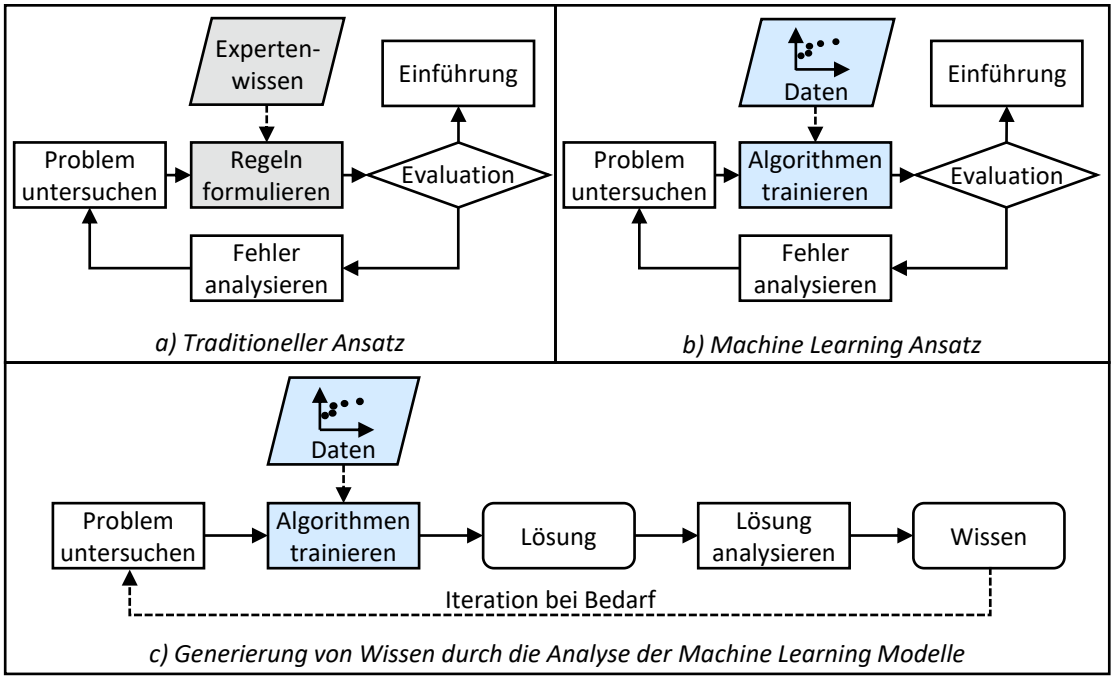


Abbildung 2-11 Traditioneller Ansatz vs. Machine Learning zur Wissensgenerierung in Anlehnung an Géron (2017) und Mehlstäubl et al. (2023a)

2.2.4 Datenanalyseprozess

Der Einsatz von Machine Learning Algorithmen stellt nur eine Phase in einem gesamten Datenanalyseprozess dar. Für die Nutzung von Machine Learning zur Analyse komplexer Produktportfolios muss ein Datenanalyseprozess vollständig durchlaufen werden. In der Literatur sind eine Vielzahl an Datenanalyseprozessen zu finden. Am verbreitetsten sind der Knowledge Discovery in Databases (KDD) Prozess von Fayyad et al. (1996) und der CRoss Industry Standard Process for Data Mining (CRISP-DM) von Wirth und Hipp (2000). Eine Übersicht weiterer Datenanalyseprozesse kann Wilberg (2020) entnommen werden.

Knowledge Discovery in Databases (KDD)

Der KDD-Prozess konzentriert sich auf den Gesamtprozess der Wissensentdeckung aus Datenbanken. Abbildung 2-12 gibt einen Überblick über die einzelnen Phasen des Prozesses sowie deren Ergebnisse (siehe Ester und Sander 2000; Fayyad et al. 1996). Zunächst findet auf Basis der Zielsetzung des Datenanalyseprojekts die *Auswahl* der Zieldaten statt. In der *Vorverarbeitung* werden die ausgewählten Daten bereinigt. Zu den grundlegenden Operationen gehören das Entfernen von Ausreißern und die Festlegung von Strategien für den Umgang mit fehlenden Werten. In der *Transformation* werden Verfahren zur Kodierung und Dimensionsreduktion eingesetzt, um die Daten in eine für den Algorithmus lesbare Form zu bringen und die Leistung des Algorithmus zu verbessern. Anschließend werden geeignete *Data Mining* Verfahren ausgewählt und implementiert. Wie in Kapitel 2.2.1 beschrieben, bedient sich das Data Mining an

Verfahren der klassischen Statistik und des Machine Learning, weshalb der KDD-Prozess auch für Machine Learning Problemstellungen gültig ist. In der letzten Phase finden die *Interpretation und Evaluation* der extrahierten Muster und Modelle sowie die Bewertung der Erfüllung der definierten Ziele statt.

Für die industrielle Anwendung von Machine Learning ist zu Beginn ein Verständnis für betrachtete Domäne sowie die Datenlage zu erarbeiten. Dies beinhaltet Kenntnisse über die Ziele und Herausforderungen sowie die Ableitung von Anwendungsfällen. Daneben sind die vorhandenen Daten zu ermitteln und hinsichtlich deren Erzeugung und Ablage sowie deren Charakteristiken zu analysieren. Diese Tätigkeiten werden im KDD nicht im Detail betrachtet, weshalb er für dieses Promotionsvorhaben nur bedingt geeignet ist.

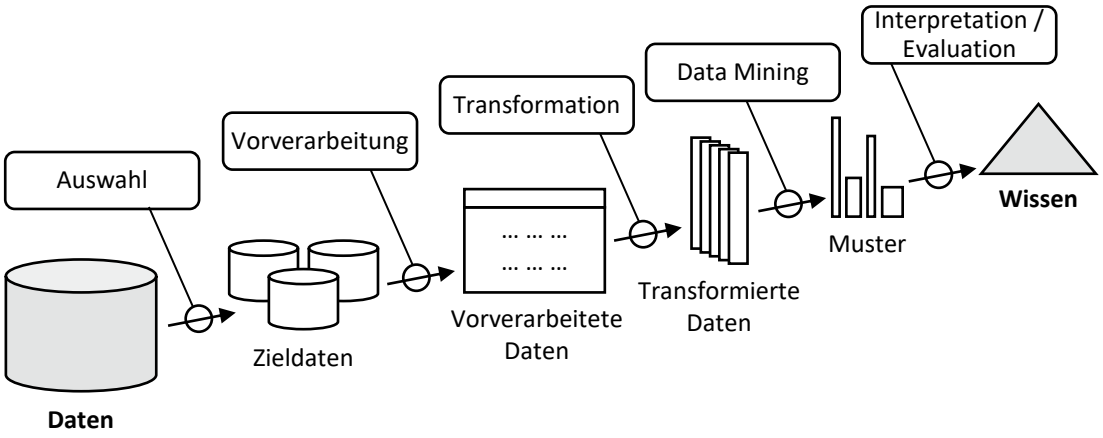


Abbildung 2-12: Knowledge Discovery in Databases (KDD) Prozess nach Fayyad et al. (1996)

CRoss Industry Standard Process for Data Mining (CRISP-DM)

Der CRISP-DM von Wirth und Hipp (2000) wurde von Vertretern aus verschiedenen Industrieunternehmen entwickelt. Er legt den Schwerpunkt auf den industriellen Einsatz von Datenanalysen. Im Vergleich zum KDD-Prozess enthält er zu Beginn die Phasen des Geschäfts- und Datenverständnisses, in denen die Domäne und die Datenlage analysiert werden. Die einzelnen Phasen des CRISP-DM und deren Abhängigkeiten sind in Abbildung 2-13 dargestellt und werden im Folgenden näher beschrieben.

Die Phase des *Geschäftsverständnisses* fokussiert das Verständnis über die Aufgaben und die Herausforderungen der betrachteten Domäne sowie die Erarbeitung der Ziele und die Planung des Datenanalyseprojekts. Anschließend wird ein *Datenverständnis* generiert, indem erste Daten gesammelt und beschrieben werden. Die Daten werden hinsichtlich interessanter Teilmengen, erster Muster und Charakteristiken untersucht sowie deren Qualität überprüft. Es besteht eine enge Abhängigkeit zwischen der Phase des Geschäftsverständnisses und der des Datenverständnisses, da für die Zieldefinition Kenntnisse über die verfügbaren Daten benötigt werden.

Die Phase der *Datenvorbereitung* beinhaltet alle Aktivitäten, die für die Überführung der Rohdaten in den finalen Datensatz erforderlich sind. Zu diesen Aufgaben gehören die Datenbereinigung, die Konstruktion neuer Attribute und die Transformation der Daten für die Modellierungswerkzeuge. In der *Modellierung* werden verschiedene Machine Learning Algorithmen ausgewählt und die optimalen Werte für die Modellparameter bestimmt. Es besteht ebenfalls ein enger Zusammenhang zwischen der Datenvorbereitung und der Modellierung, da oft erst bei der Modellierung Qualitätsprobleme in den Daten erkannt werden oder Ideen für die Konstruktion neuer Daten aufkommen.

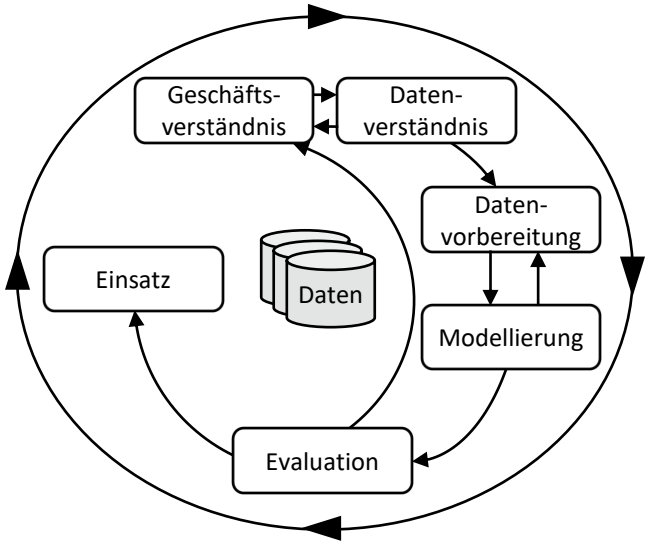


Abbildung 2-13: CRISP-DM Prozess nach Wirth und Hipp (2000)

In der *Evaluation* liegen ein oder mehrere Machine Learning Modelle vor, welche hinsichtlich der in der der ersten Phase definierten Ziele bewertet und verglichen werden. Am Ende dieser Phase ist eine Entscheidung über die Verwendung und den anschließenden Einsatz der Ergebnisse zu treffen. Für den *Einsatz* des erlangten Wissens im industriellen Kontext wird dieses dem späteren Nutzer zur Verfügung gestellt. Je nach Anforderungen und Zielsetzung kann dieser Schritt unterschiedliche Ausprägungen annehmen. Er kann von der Erstellung eines Berichts bis hin zur Implementierung eines wiederholbaren Datenanalyseprozesses in Form eines Softwaretools reichen. Eine Übersicht der Aktivitäten der einzelnen Phasen des CRISP-DM wird in Tabelle 2-1 gegeben.

Der CRISP-DM berücksichtigt im Vergleich zu anderen Prozessen die Phasen des Geschäfts- und Datenverständnisses sowie den Einsatz der Ergebnisse und ermöglicht so eine industrielle Nutzung von Machine Learning. Aus diesem Grund wird der CRISP-DM im weiteren Verlauf des Forschungsvorhabens herangezogen. Er wird für die Entwicklung des Frameworks zur Analyse komplexer Produktportfolios mittels Machine Learning verwendet. Dabei wird zuerst ein Geschäfts- und Datenverständnis für die

operativen Produktportfoliogestaltung generiert. Anschließend werden in Abhängigkeit der Produktportfoliodaten die erforderlichen Schritte zur Datenvorbereitung, Modellierung und Evaluation bereitgestellt. Abschließend werden Möglichkeiten aufgezeigt, wie die Modelle und das Wissen für die operative Produktportfoliogestaltung eingesetzt werden können.

Tabelle 2-1: Übersicht der Aktivitäten des CRISP-DM in Anlehnung an Wirth und Hipp (2000)

Geschäfts-verständnis	Daten-verständnis	Daten-vorbereitung	Modellierung	Evaluation	Einsatz
Geschäftsziele bestimmen	Erste Daten sammeln	Daten auswählen	Algorithmen auswählen	Ergebnisse auswerten	Einsatz planen
Situation bewerten	Daten beschreiben	Daten bereinigen	Testentwurf generieren	Prozess überprüfen	Überwachung und Pflege planen
Ziele der Datenanalyse festlegen	Daten untersuchen	Daten konstruieren	Modell erstellen	Nächste Schritte festlegen	Projekt dokumentieren
Projektplan erstellen	Datenqualität überprüfen	Daten transformieren	Modell bewerten		Projekt reviewen

2.2.5 Machine Learning Verfahren und Algorithmen

Machine Learning kann grundsätzlich in drei Arten des Lernens unterteilt werden. Diese sind das überwachte Lernen, das unüberwachte Lernen und das bestärkende Lernen (siehe Abbildung 2-14). Beim **überwachten Lernen** (engl. supervised learning) enthalten die Daten die gewünschte Lösung, die sogenannten Labels, die zum Trainieren und Testen der Modelle verwendet werden (Géron 2017). Überwachte Lernverfahren sind die Regressionsanalyse und die Klassifikationsanalyse. Bei der Regressionsanalyse wird die Beziehung zwischen einer abhängigen Variablen und einer oder mehreren unabhängigen Variablen modelliert (Backhaus et al. 2015). Bei der Klassifikationsanalyse wird ein Datenpunkt einer von mehreren vordefinierten Klassen zugewiesen (Weiss und Kulikowski 1991). Beim **unüberwachten Lernen** (engl. unsupervised learning) enthalten die Trainingsdaten keine Labels. Das Ziel des unüberwachten Lernens besteht darin, Muster in den Eingangsmerkmalen zu erkennen, ohne dass spezifische Zielvariablen vorliegen (Russell 2010). Gängige Verfahren sind die Clusteranalyse und die Assoziationsanalyse. Bei der Clusteranalyse werden Datenpunkte in Gruppen oder Cluster eingeteilt, so dass die Objekte im selben Cluster so ähnlich wie möglich und Objekte aus verschiedenen Clustern so unähnlich wie möglich sind (Ester und Sander 2000). Die Assoziationsanalyse drückt Regeln über häufig vorkommende Beziehungen in Daten aus (Zhao und Bhowmick 2003). Eine dritte Form des Machine Learning ist das **bestärkende Lernen** (engl. reinforcement learning). Ein Agent lernt aus der direkten Interaktion mit seiner Umgebung, ohne sich auf gelabelte Beispiele zu verlassen (Sutton und Barto 2018).

Der Fokus dieser Arbeit liegt auf dem Einsatz von überwachtem und unüberwachtem Machine Learning zur Analyse komplexer Produktportfolios. Es werden die Verfahren der Regressions- und Klassifikationsanalyse des überwachten Lernens und die Cluster- und Assoziationsanalyse des unüberwachten Lernens berücksichtigt. Aufgrund des iterativen Trainingsprozesses und der Komplexität des Produktportfolio- und Variantenmanagements, welche eine enorm aufwendige Definition der Umwelt zur Folge hat, wird das bestärkende Lernen in diesem Promotionsvorhaben nicht weiter berücksichtigt.

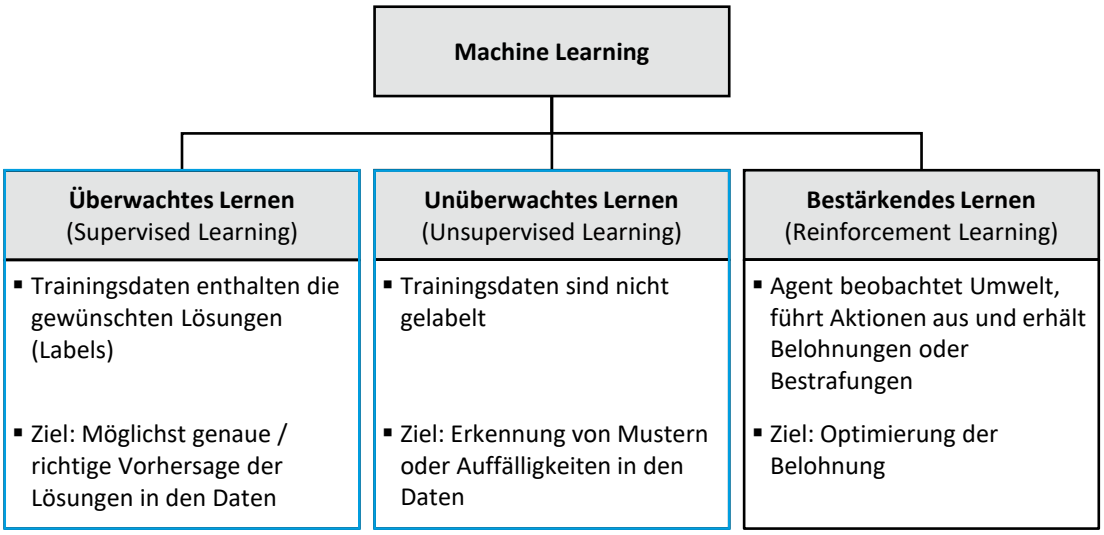


Abbildung 2-14: Arten des Machine Learning

2.2.5.1 Überwachtes Lernen

Im Folgenden wird auf die meistverbreiteten überwachten Lernalgorithmen eingegangen, welche auch im Rahmen des Frameworks dieser Arbeit berücksichtigt werden. Die meisten dieser Algorithmen können sowohl für Regressionsanalysen als auch für Klassifikationsanalysen eingesetzt werden. Überwachtes Lernen hat das Ziel, eine Abbildung von Eingaben x auf Ausgaben y zu lernen (Cunningham et al. 2008). Bei einer Regressionsanalyse ist y eine kontinuierliche Variable und bei einer Klassifikationsanalyse ist $y \in \{1, ..., C\}$, wobei C die Anzahl der Klassen darstellt (Murphy 2012). Ist $C = 2$, spricht man von einer binären Klassifikation und ist $C > 2$, handelt es sich um eine Multi-Klassen-Klassifikation (Jeatrakul und Wong 2009). Wenn sich die Klassenlabels nicht gegenseitig ausschließen und mehr als eine Klasse gewählt werden kann, spricht man von einer Multi-Label-Klassifikation (Murphy 2012).

Lineare Regression

Die lineare Regression ist ein verbreiteter Lernalgorithmus, bei dem eine gerade Linie an eine Reihe von Datenpunkten angepasst wird und so kontinuierliche Werte wie zum Beispiel die Größe, Breite oder das Gewicht vorhergesagt werden können (Mishra

2020) (siehe Abbildung 2-15). Die Gerade wird durch eine Linearkombination aus den Eingangsmerkmalen wie folgt ausgedrückt (Seber und Lee 2012):

$$y_{pred} = \sum_{i=1}^{n} w_i x_i + \varepsilon \qquad \textit{Formel 2-1}$$

Beim Training werden die Modellparameter, also die Gewichtsfaktoren w_i und der Restfehler ε, iterativ angepasst und die Kostenfunktion minimiert (Weisberg 2005).

Die lineare Regression hat den Vorteil, dass der Algorithmus durch die Verwendung einer Linearkombination als Geradengleichung leicht zu verstehen und nachzuvollziehen ist. Zudem ist es einfach eine Überanpassung zu vermeiden. Überanpassung bedeutet, dass ein Modell die Trainingsdaten perfekt abbildet, aber aufgrund einer fehlenden Generalisierung schlechte Werte für die noch nicht gesehenen Testdaten prognostiziert (Ying 2019). Der Nachteil der linearen Regression besteht darin, dass lediglich lineare Abhängigkeiten zwischen den Eingangsmerkmalen und der Zielvariablen abgebildet werden können und dies für die meisten praktischen Anwendungen eine zu starke Vereinfachung darstellt (Ray 2019). Zudem haben viele Eingangsmerkmale und Datenpunkte negative Auswirkungen auf die Güte der Modelle.

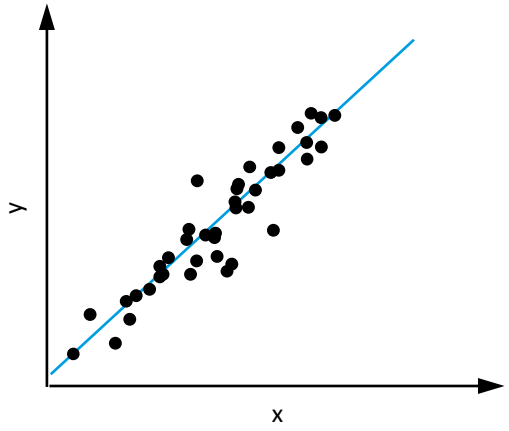


Abbildung 2-15: Ein zweidimensionales lineares Regressionsmodell nach Mishra (2020)

Logistische Regression

Die logistische Regression wird entgegen dem Namen für Klassifikationsanalysen eingesetzt und schätzt die Wahrscheinlichkeit, dass eine Instanz zu einer bestimmten Klasse gehört (Bonaccorso 2018). Dabei wird die Ausgabe der linearen Regression verwendet und mithilfe einer Sigmoidfunktion und einer Entscheidungsgrenze (siehe Abbildung 2-16) in einen Wert zwischen 0 und 1 umgewandelt (Burkov 2019):

$$y_{pred}\,(x) = б(\sum_{i=1}^{n} w_i x_i) \qquad \textit{Formel 2-2}$$

Mit der Sigmoidfunktion б:

$$б(x) = \frac{1}{1 + e^{-x}} \qquad \textit{Formel 2-3}$$

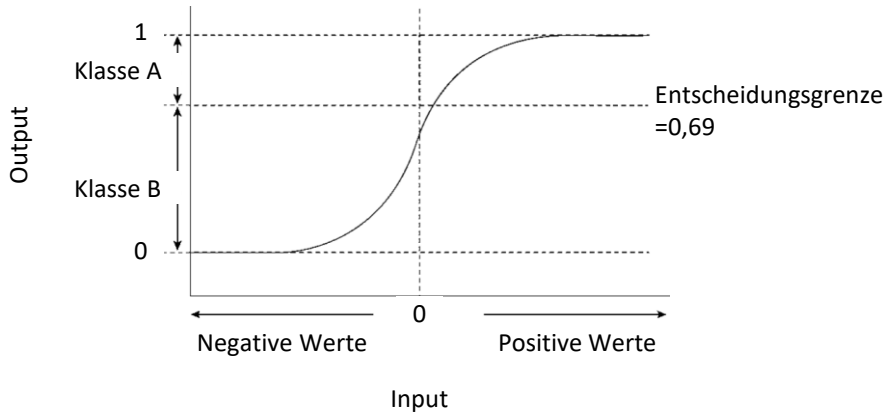


Abbildung 2-16: Sigmoidfunktion der logistischen Regression nach Mishra (2020)

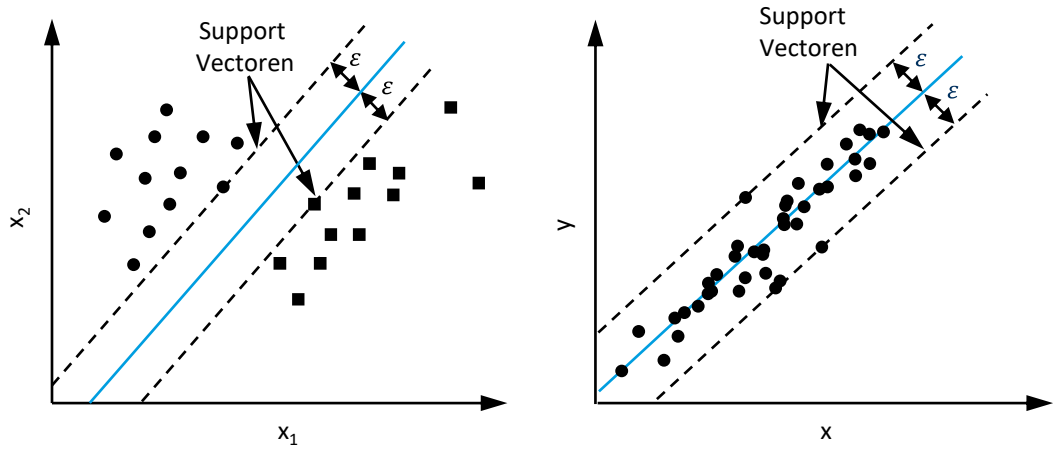
Das Ziel des Trainings besteht darin, den Parametervektor w_i sowie die Entscheidungsgrenze so einzustellen, dass das Modell große Wahrscheinlichkeiten für positive Instanzen (y_{real} = 1) und niedrige Wahrscheinlichkeiten für negative Instanzen (y_{real} = 0) in den Trainingsdaten vorhersagt.

Die logistische Regression verhält sich ähnlich wie die lineare Regression. Sie besitzt durch den Einsatz einer Linearkombination eine hohe Transparenz und ist daher leicht zu interpretieren und robust gegenüber einer Überanpassung. Daneben werden lediglich lineare Abhängigkeiten abgebildet und das Modell hat Schwierigkeiten mit vielen Datenpunkten und Eingangsmerkmalen (Ray 2019).

Support Vector Machine

Eine Support Vector Machine (SVM) ist ein Modell, das sowohl für Klassifikationsanalysen als auch für Regressionsanalysen verwendet werden kann (Mishra 2020). Zur Klassifikation wird eine Entscheidungsgrenze gebildet, welche die Datenpunkte in Klassen unterteilt. Das Ziel ist es, eine Entscheidungsgrenze zu finden, welche den Abstand zwischen den beiden Klassen maximiert (Cortes und Vapnik 1995). Dafür werden Stützvektoren (engl. support vectors) an den Rändern der Klassen gebildet und deren Abstand zur Entscheidungsgrenze ε beim Training maximiert (Mishra 2020) (siehe Abbildung 2-17 a). Bei der Support Vector Regression wird dagegen versucht, alle Datenpunkte innerhalb der Stützvektoren zu platzieren (Hard Margin) oder es werden einige Übertretungen (Soft Margin) erlaubt (siehe Abbildung 2-17 b). Der Abstand zwischen der Entscheidungsgrenze ε und den Stützvektoren ist dabei zu minimieren. Die Entscheidungsgrenze dient bei der Regression als Vorhersagefunktion.

Die Stärke der SVM ist die Fähigkeit, nicht-lineare Entscheidungsgrenzen zu schaffen, indem sie eine mathematische Funktion verwendet, die als Kernel bezeichnet wird und dazu dient, jeden Eingabepunkt in einen höherdimensionalen Raum zu transformieren, in dem eine lineare Entscheidungsgrenze gefunden werden kann (Hearst et al. 1998). Zudem besitzt die SVM eine geringe Wahrscheinlichkeit zur Überanpassung, jedoch ist es schwierig die richtige Kernelfunktion zu identifizieren (Ray 2019). Die Transformationen der Eingangsmerkmale in einen höherdimensionalen Raum erfordert bei vielen Eingangsmerkmalen und Datenpunkten eine hohe Rechenleistung und daher eine lange Trainingszeit (Singh et al. 2016).



a) Support Vector Klassifikation *b) Support Vector Regression*

Abbildung 2-17: Zweidimensionale Support Vector Machine zur Klassifikation und Regression in Anlehnung an Cristianini und Shawe-Taylor (2000)

K-Nearest Neighbors

K-Nearest Neighbors (kNN) ist ein nicht-parametrischer Lernalgorithmus (Burkov 2019). Das heißt, die Anzahl an Modellparameter wird vor dem Training nicht konkret festgelegt. Der kNN Algorithmus sucht nach den k Trainingsbeispielen, welche einem neuen noch unbekannten Beispiel am nächsten kommen und bildet bei einer Regression den Durchschnittswert oder bei einer Klassifikation die am häufigsten vorkommende Kennzeichnung (siehe Abbildung 2-18) (Kramer 2013). Im Gegensatz zu anderen Algorithmen, welche die Trainingsdaten nach der Modellbildung verwerfen, behält der kNN diese für die Vorhersagen (Burkov 2019). Die Nähe zu den Beispielen aus den Trainingsdaten wird mit einer Abstandsfunktion bestimmt.

Der kNN Algorithmus ist einfach nachzuvollziehen, da die Vorhersage auf den k nächsten Datenpunkte beruht. Das Training besteht lediglich aus der Speicherung der Trainingsdaten und benötigt daher wenig Aufwand. Jedoch wird durch das Beibehalten der Trainingsbeispiele im Modell bei großen Datensätzen eine hohe Speicherkapazität und Rechenleistung benötigt. Zudem hängt die Leistungsfähigkeit stark von dem

gewählten Parameter für k ab und es kann bei hochdimensionalen Daten zu einem Rückgang der Genauigkeit in einzelnen Bereichen kommen (Singh et al. 2016).

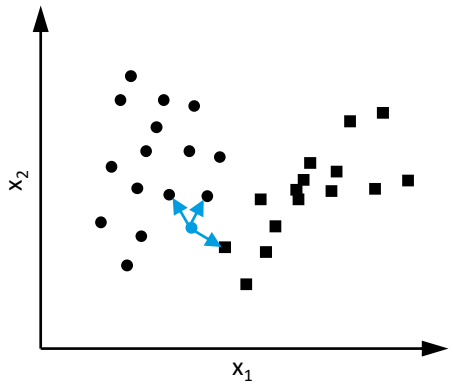


Abbildung 2-18: Zweidimensionale kNN-Klassifikation mit k=3 in Anlehnung an Peterson (2009)

Entscheidungsbaum

Ein Entscheidungsbaum ist eine baumartige Struktur, bei der jeder übergeordnete Knoten eine Entscheidungsgrenze und die untergeordneten Knoten die Entscheidungsergebnisse darstellen (Mishra 2020). Er kann sowohl für Klassifikationsanalysen als auch Regressionsanalysen verwendet werden. Der Grundgedanke eines jeden mehrstufigen Ansatzes besteht darin, eine komplexe Entscheidung in mehrere einfachere Entscheidungen aufzuteilen (Safavian und Landgrebe 1991). Der oberste Knoten des Baumes wird als Wurzelknoten bezeichnet (Mishra 2020). Die untersten Knoten enthalten das Ergebnis und werden Blattknoten genannt. In Abbildung 2-19 ist beispielhaft ein Entscheidungsbaum zur Lösung eines zweidimensionalen Klassifikationsproblems abgebildet. Zu Beginn des Trainings, also der Erstellung des Entscheidungsbaums, besteht der Entscheidungsbaum lediglich aus dem Startknoten, der alle Trainingsbeispiele enthält, sodass die Vorhersage für sämtliche Eingaben identisch ist. Anschließend wird die Menge in zwei Unterdatenmengen aufgeteilt, welche zwei neue Blattknoten bilden. Die Herausforderung bei der Bildung eines Entscheidungsbaums besteht darin, die beste Aufteilung für eine Entscheidung zu finden (Kotsiantis 2013).

Ein Vorteil des Entscheidungsbaums ist, dass die logischen Regeln einfach zu interpretieren sind (Kotsiantis 2013). Zudem kann ein Entscheidungsbaum redundante Eingangsmerkmale handhaben, ist robust gegenüber Rauschen und kann lineare Abhängigkeiten sowie eine Vielzahl an Eingangsmerkmalen abbilden (Singh et al. 2016). Ein Nachteil ist, dass er instabil sein kann und die Größe des Baums schwierig zu kontrollieren ist, weshalb es zu einer Überanpassung kommen kann (Ray 2019).

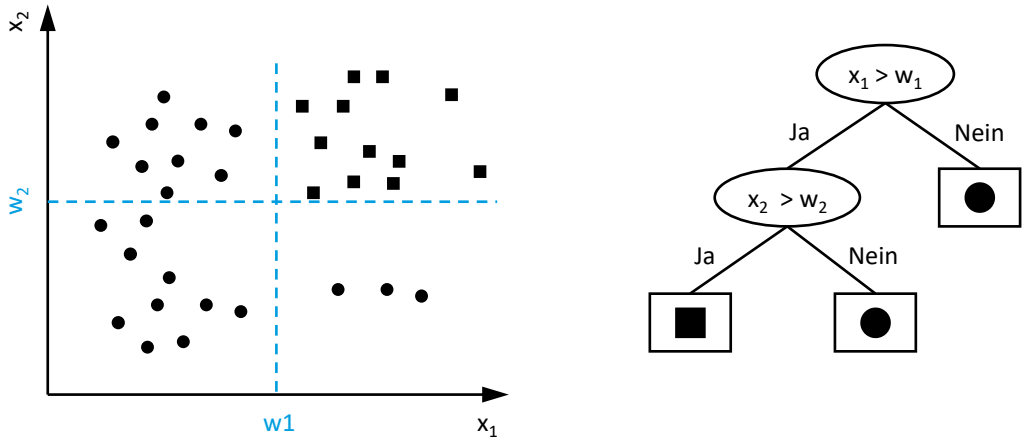


Abbildung 2-19: Entscheidungsbaum in Anlehnung an Alpaydin (2020)

Random Forest

Random Forest Algorithmen nutzen das sogenannte Ensemble Learning, um die Leistungsfähigkeit von Entscheidungsbäumen zu verbessern (Genuer et al. 2008). Beim Ensemble Learning wird kein präzises Modell erlernt, sondern eine Vielzahl weniger genauer Modelle trainiert und deren Vorhersagen kombiniert (siehe Abbildung 2-20). Beim Training werden zufällige Stichproben aus der Trainingsdatenmenge entnommen und mit jeder wird ein Entscheidungsbaum gebildet (Breiman 2001). Das Training wird nach wenigen Iterationen abgebrochen, sodass flache Entscheidungsbäume mit geringer Genauigkeit entstehen. Die Vorhersage für ein neues Beispiel erhält man bei einer Regression durch den Mittelwert der Modelle oder bei einer Klassifikation durch eine Mehrheitsentscheidung (TIBCO 2022).

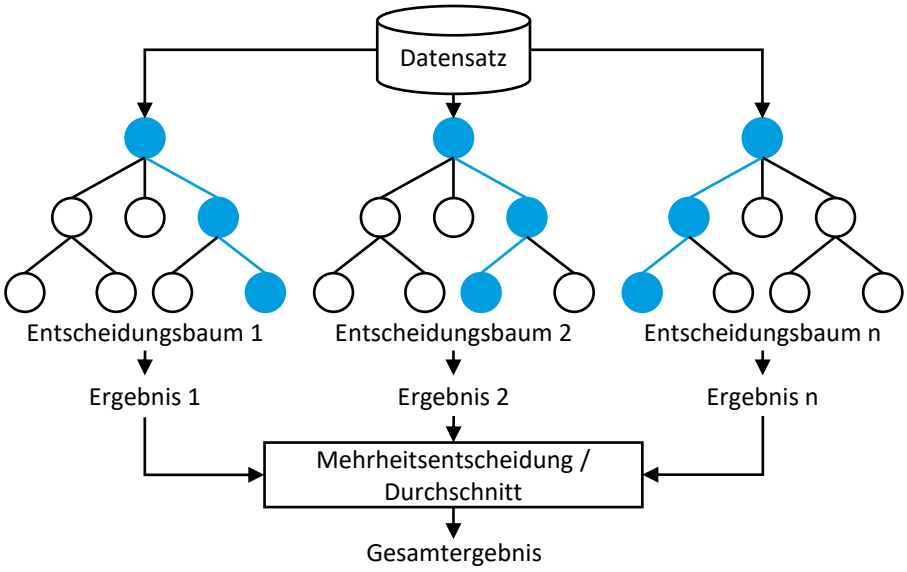


Abbildung 2-20: Aufbau eines Random Forest nach TIBCO (2022)

Random Forest Algorithmen sind schnell und einfach zu implementieren, liefern hochpräzise Vorhersagen und können mit einer großen Anzahl von Eingabevariablen umgehen, ohne dass die Gefahr einer Überanpassung besteht (Biau 2012). Ein Nachteil des Random Forest Algorithmus ist, dass er durch die Kombination der Ergebnisse mehrerer Entscheidungsbäume schwer nachzuvollziehen und zu interpretieren ist (Ziegler und König 2014).

Künstliche neuronale Netze

Künstliche neuronale Netze sind Modelle deren Aufbau von biologischen neuronalen Netzen inspiriert ist (Gupta 2013). Neuronale Netze bestehen aus Neuronen, die durch Kanten miteinander verbunden sind. Jede Kante hat einen Gewichtungswert, der mit dem Wert des Neurons multipliziert wird, von dem die Verbindung ausgeht. Jedes Neuron arbeitet, indem es die Summe seiner Eingaben berechnet und diese Summe durch eine Aktivierungsfunktion leitet. Der Ausgangswert des Neurons ist das Ergebnis der Aktivierungsfunktion. Beim Training werden die Gewichtungen sukzessive angepasst. Die Neuronen eines neuronalen Netzes sind in mehreren Schichten (engl. layer) angeordnet. Es können drei Arten von Layern unterschieden werden (Mishra 2020):

- Input Layer: Erhält direkt die Eingaben für die Berechnung. Anzahl der Neuronen ist gleich der Anzahl der Eingangsmerkmale.
- Output Layer: Liefert die Ausgabe der Berechnung. In dieser Schicht können je nach Art des Problems ein oder mehrere Neuronen vorhanden sein.
- Hidden Layer: Befinden sich zwischen dem Input und Output Layer. Im Gegensatz zum Input und Output Layer kann ein neuronales Netz aus beliebig vielen Hidden Layer mit einer beliebigen Anzahl an Neuronen bestehen. Ein einfaches neuronales Netz muss nicht unbedingt eine verborgene Schicht haben. Komplexe neuronale Netze haben eine Vielzahl an Hidden Layer mit jeweils einer großen Anzahl an Neuronen. Die Neuronen im Input und Output Layer sind mit den Neuronen im ersten bzw. letzten Hidden Layer verbunden.

Abbildung 2-21 zeigt ein einfaches neuronales Netz mit zwei Neuronen im Input Layer und einem Neuron im Output Layer (Regressionsproblem) (Mishra 2020).

Der große Vorteil neuronaler Netze ist, dass sie stark nicht-lineare Abhängigkeiten mit einer Vielzahl an Datenpunkten und Eingangsmerkmalen abbilden können. Sie erfordern jedoch viel Zeit zum Training und neigen zur Überanpassung (Singh et al. 2016). Darüber hinaus besitzen sie aufgrund der Kopplung einer Vielzahl an Neuronen mit deren Aktivierungsfunktionen und Gewichtungen eine geringe Transparenz und Nachvollziehbarkeit.

Zusammenfassung: Überwachtes Lernen

Die meisten überwachten Lernalgorithmen können sowohl für Regressionsaufgaben als auch für Klassifikationsaufgaben eingesetzt werden. Zusammenfassend wird mit Tabelle 2-2 eine Übersicht über den Inhalt sowie die Vor- und Nachteile der vorgestellten Algorithmen gegeben. Aufgrund der Eignung zur Analyse großer Datensätze und

Abbildung nicht-linearer Zusammenhänge sind mit Hinblick auf die operative Gestaltung komplexer Produktportfolios vor allem der Random Forest sowie die künstlichen neuronalen Netze vielversprechend.

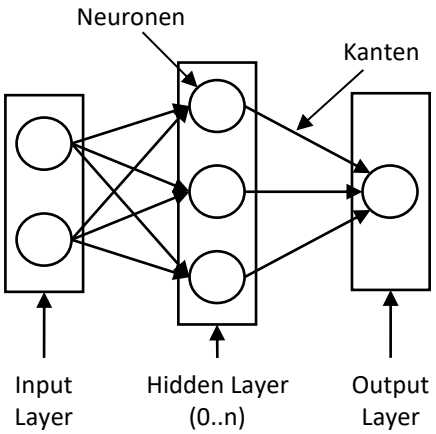


Abbildung 2-21: Struktur eines neuronalen Netzes nach Mishra (2020)

Tabelle 2-2: Übersicht überwachter Lernalgorithmen

Algorithmus	Verfahren	Inhalt	Vor- und Nachteile
Lineare Regression	Regression	Einsatz einer Linearkombination als Geradengleichung	+ Hohe Transparenz + Robust gegen Überanpassung - Lediglich nicht-lineare Abhängigkeiten abbildbar - Probleme bei hochdimensionalen Daten
Logistische Regression	Klassifikation	Geradengleichung wird mit einer Sigmoidfunktion zur Bestimmung der Klassenzugehörigkeit verwendet	+ Hohe Transparenz + Robust gegen Überanpassung - Lediglich nicht-lineare Abhängigkeiten abbildbar - Probleme bei hochdimensionalen Daten
Support Vector Machine	Regression / Klassifikation	Stützvektoren werden an den Klassenrändern gebildet und deren Abstand maximiert bzw. minimiert	+ Nicht-lineare Entscheidungsgrenzen möglich + Geringe Wahrscheinlichkeit zur Überanpassung - Auswahl der richtigen Kernelfunktion schwierig - Lange Trainingszeiten
K-Nearest Neighbors	Regression / Klassifikation	Ermittlung der k Datenpunkte, welche einem neuen Datenpunkt am nächsten kommen	+ Hohe Transparent + Wenig Aufwand zum Training - Hohe Speicherkapazität erforderlich - Ergebnisse stark von k abhängig
Entscheidungs-baum	Regression / Klassifikation	Baumartige Struktur aus Entscheidungsknoten und -ergebnissen	+ Einfach zu interpretieren - Abbildung nichtlinearer Zusammenhänge - Instabil
Random Forest	Regression / Klassifikation	Kombination der Ergebnisse mehrerer Entscheidungsbäume mit geringer Genauigkeit	+ Großen Anzahl an Eingangsmerkmalen und Datenpunkten + Hochdimensionale Zusammenhänge abbildbar - Schwer interpretierbar
Künstliche neuronale Netze	Regression / Klassifikation	Aufbau inspiriert durch biologische neuronale Netze mit Neuronen, Kanten und Layern	+ Große Anzahl an Eingangsmerkmalen und Datenpunkten + Stark nicht-lineare Zusammenhänge abbildbar - Trainingszeit hoch - Neigt zur Überanpassung

2.2.5.2 Unüberwachtes Lernen

Im Vergleich zu überwachten Lernverfahren unterscheiden sich unüberwachte Lernverfahren hinsichtlich der Zielsetzung und der Vorgehensweise signifikant voneinander. Daher wird im Folgenden zuerst auf Clusteralgorithmen und anschließend auf Assoziationsalgorithmen eingegangen.

2.2.5.2.1 Clusteralgorithmen

Das Clustering ist eine Technik, bei der in einer Datenmenge einander ähnliche Instanzen identifiziert und in homogene Gruppen (engl. Cluster) eingeteilt werden (Ester und Sander 2000). Formal kann dies ausgedrückt werden als die Unterteilung von n Datenpunkten eines Datensatzes $D = \{x_1, x_2, ..., x_n\}$ in k disjunkte Teilmengen von D, bezeichnet als $C_1, C_2, ..., C_k$. Jeder Datenpunkt ist dabei beschrieben als ein Vektor aus Eingangsmerkmalen (Kubat 2021). Das Clustering, also der Output des Verfahrens, entspricht dann $C = \{C_1, C_2, ..., C_k\}$ (Hennig und Meila 2015). Für die Identifikation von Clustern können distanzbasierte, hierarchische, dichtebasierte und probabilistische Algorithmen unterschieden werden. Im Folgenden werden diese näher betrachtet.

Distanzbasiertes Clustering – k-Means Algorithmus

Der k-Means Algorithmus identifiziert innerhalb eines Datensatzes k Gruppen, die vom jeweiligen Zentrum der Gruppe repräsentiert werden, den sogenannten Centroiden (Alpaydin 2020). Für jedes Cluster wird ein Centroid ermittelt, dessen Position sich aus dem Mittelwert (engl. mean) aller darin enthaltenen Instanzen ergibt (Marsland 2011). Für die Zuordnung der Instanzen zu ihrem nächstgelegenen Centroid wird eine Distanzfunktion verwendet (Marsland 2011). In den meisten Fällen wird die quadrierte euklidische Distanz genutzt. Die Summe der quadrierten euklidischen Distanz über alle Instanzen eines Clusters gibt dessen Kompaktheit an und dient als Optimierungskriterium für den Lernprozess (Ester und Sander 2000). Der k-Means Algorithmus erfordert die Festlegung der Anzahl zu bildender Cluster k. In einem Initialisierungsschritt werden zufällig k Instanzen aus den Eingangsdaten D ausgewählt und als Centroide μ_j festgelegt. Danach folgen pro Iteration zwei Schritte (siehe Abbildung 2-22). Im ersten Schritt wird jeder Datenpunkt x_i einem Cluster zugeordnet. Dazu wird für jedes x_i die Distanz zu allen μ_j ermittelt und dem jeweils nächstgelegenen zugewiesen. Im zweiten Schritt werden die Centroide aktualisiert. Dies geschieht, indem der Mittelwert aller im jeweiligen Cluster enthaltenen x_i berechnet wird und μ_j an der ermittelten Stelle neu positioniert wird. Diese zwei Schritte werden so lange iteriert, bis sich die Positionierung der Centroide durch die Aktualisierung nicht mehr verändert (Marsland 2011).

Der k-Means Algorithmus ist aufgrund seiner Geschwindigkeit sehr populär (Mirkin 2016). Jedoch existieren bestimmte Datenstrukturen, für die der Algorithmus keine akkuraten Ergebnisse liefert. Dazu gehören Datensätze mit Clustern stark unterschiedlicher Größe, stark unterschiedlicher Punktdichte oder einer sphärischen Form (Ester und Sander 2000). Darüber hinaus handelt es sich bei der vom k-Means gefundenen Lösung um ein lokales Minimum. Daher hat die Initialisierung der Centroide einen

signifikanten Einfluss auf die Güte der Lösung, weshalb in der Anwendung das Clustering mehrmals durchgeführt und das beste Ergebnis ermittelt wird (Flach 2012). Ein weiterer Faktor für die Güte des Ergebnisses des Algorithmus ist die Auswahl der Anzahl gesuchter Cluster k.

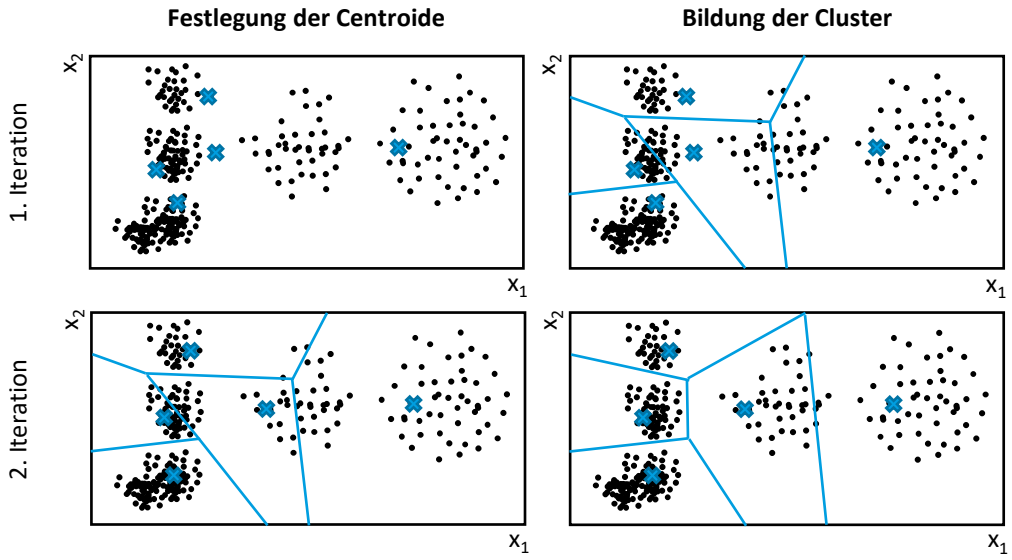


Abbildung 2-22: Distanzbasiertes Clustering mit k-Means Algorithmus in Anlehnung an Marsland (2011)

Hierarchisches Clustering – Single-Linkage Algorithmus

Hierarchische Clusterverfahren erzeugen eine mehrschichtige Anordnung von Clustern und Subclustern sowie deren Rangordnung untereinander. Der Single-Linkage Algorithmus ist einer der meistgenutzten Algorithmen dieser Art (Murtagh 2016). Er folgt einem Bottom-Up-Prinzip, welches die Cluster von unten nach oben aufbaut. Als Input für den Algorithmus dient eine Dissimilaritätsmatrix. Diese gibt die Abstände aller Elemente der gegebenen Datenmenge zueinander an. Eingangs bildet dabei jeder einzelne Datenpunkt sein eigenes Cluster. In jedem Iterationsschritt werden dann die beiden nächstgelegenen Cluster vereint, bis letztlich alle Punkte in nur einem Cluster zusammengefasst sind (Contreras und Murtagh 2015). Abbildung 2-23 veranschaulicht die Reihenfolge (Nummerierung I-V) in der dieser Algorithmus Datenpunkte zu Clustern vereint.

Hierarchische Clusteralgorithmen haben den Vorteil, dass die Anzahl der gesuchten Cluster im Voraus nicht gegeben werden muss (Flach 2012). In jedem Iterationsschritt wird ein Clustering gefunden, dessen Güte beurteilbar ist (Flach 2012). Nachteilig ist, dass die Algorithmen Heuristiken darstellen, welche per Definition nicht zwangsläufig zu einem globalen Optimum führen (Murphy 2012).

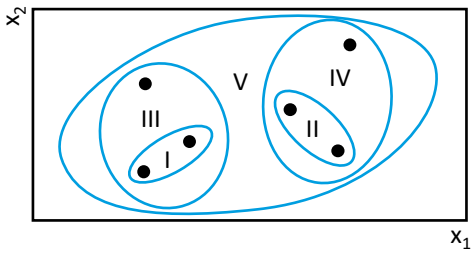


Abbildung 2-23: Hierarchisches Clustering mit Single-Linkage Algorithmus

Dichtebasiertes Clustering – DBSCAN Algorithmus

Bei dichtebasierten Clusterverfahren werden Cluster definiert als Regionen mit hoher Dichte von Datenpunkten, separiert durch Regionen mit geringer Dichte (Ester und Sander 2000). Ein verbreiteter Vertreter dieser Methoden ist der DBSCAN Algorithmus (Ester et al. 1996). Der Grundgedanke des DBSCAN ist, dass Cluster dort liegen, wo die Punktdichte in der Umgebung jedes Punktes des Clusters einen bestimmten Grenzwert überschreitet. Diese Grenzdichte wird durch den Radius ε, der die zu betrachtende Umgebung um einen Datenpunkt p definiert, sowie den Grenzwert MinPts, welcher die Mindestanzahl von Datenpunkten innerhalb von ε um p angibt, bestimmt. Ein Beispiel für ein solches Clustering ist in Abbildung 2-24 zu sehen. Die Initialisierung des Algorithmus besteht aus der zufälligen Auswahl eines Datenpunktes p. Für diesen wird geprüft, ob die Punktdichte in seiner Umgebung den gegebenen Grenzwert überschreitet. Ist dies nicht der Fall, wird er als Rauschen klassifiziert und zum nächsten Punkt übergegangen. Wird der Grenzwert jedoch überschritten, wird ein neuer Clusterindex erstellt. Diesem werden dann p sowie alle innerhalb von ε und um p liegende Punkte zugeordnet. So werden iterativ alle Punkte des Clusters identifiziert. Finden sich keine neuen Clustermitglieder mehr, wird die Schrittfolge ausgehend von einem noch nicht klassifizierten Punkt erneut durchgeführt. Der Algorithmus endet, wenn jeder Punkt der Datenmenge klassifiziert ist (Ester und Sander 2000).

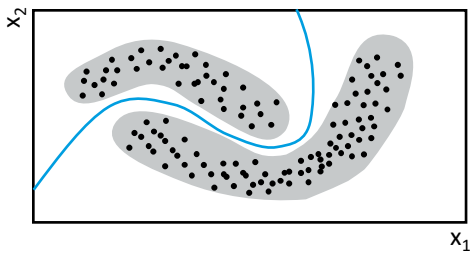


Abbildung 2-24: Dichtebasiertes Clustering mit DBSCAN Algorithmus

Der entscheidende Vorteil von dichtebasierten Verfahren gegenüber schwerpunktbasierten oder hierarchischen Algorithmen besteht darin, dass sie die Identifikation von Clustern beliebiger Form ermöglichen (Han et al. 2012). Der Algorithmus findet jedoch

kein präzises Clustering, wenn die gegebene Datenmenge Cluster aufweist, deren Dichte stark variiert, die sich überschneiden oder die eng bei Bereichen mit Rauschen liegen (Ester und Sander 2000).

Probabilistisches Clustering – EM Algorithmus

Probabilistische Verfahren definieren Cluster als eine Menge von Datenpunkten, die der gleichen Wahrscheinlichkeitsverteilung entstammen (Ester und Sander 2000) (siehe Abbildung 2-25). Ihnen zugrunde liegen sogenannte Mischmodelle, welche das gesamte Clustering als eine Zusammensetzung mehrerer Verteilungen beschreiben, die untereinander eine bestimmte Gewichtung besitzen (Murphy, 2012). Die gebräuchlichste Verteilung ist dabei die Gaußsche Normalverteilung (Marsland 2011). Die Parameter der Verteilungen, sprich ihre Mittelwerte und Kovarianzmatrizen sowie ihre Gewichtung untereinander sind eingangs nicht bekannt und müssen zunächst identifiziert werden. Ein verbreiteter Vertreter ist der EM Algorithmus (Erwartungs-Maximierungs-Algorithmus) (Dempster et al. 1977). In diesem werden die Parameter zunächst zufällig festgelegt und ihre realen Werte mittels Erwartungswertmaximierung berechnet. Der Algorithmus weist große Ähnlichkeit mit dem k-Means auf, da er grundsätzlich aus zwei Schritten besteht, die iteriert werden, bis sich die Cluster nicht mehr verändern. Zudem benötigt er als Inputvariable ebenso die Anzahl gesuchter Cluster k. Im ersten Schritt findet eine Zuordnung von Instanzen zu den Clustern statt, indem mittels des Satzes von Bayes pro Instanz für jedes Cluster die Wahrscheinlichkeit berechnet wird, dass sie diesem angehört. Aus diesen Wahrscheinlichkeiten werden im zweiten Schritt die Parameter der Verteilung und deren Gewichtung untereinander neu berechnet. Ergebnis des Algorithmus ist kein hartes Clustering, sondern die Zuordnung von Instanzen zu mehreren Clustern mit unterschiedlichen Zugehörigkeitswahrscheinlichkeiten (Murphy 2012).

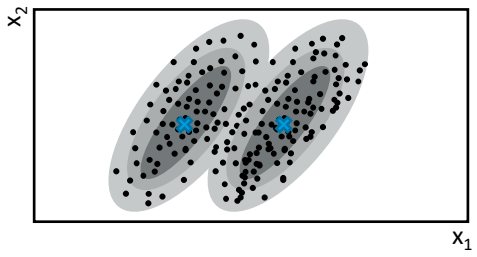


Abbildung 2-25: Probabilistisches Clustering mit EM Algorithmus (elliptischer Clusterform)

Ein Vorteil des EM Algorithmus ist es, dass er nicht nur kugelförmige Cluster finden kann, sondern auch solche die eine elliptische Form aufweisen (Burkov 2019). Abbildung 2-25 zeigt außerdem, dass durch die Ermittlung des Mittelwerts und der Kovarianzmatrix der Wahrscheinlichkeitsverteilungen des Mischmodells nicht nur die Zentren der Cluster, sondern auch ihre Ausdehnung, Form und Orientierung identifiziert werden (Ester und Sander 2000). Nachteilig ist, dass die Qualität des Clustering stark

von der Initialisierung sowie der Wahl der Inputvariablen k abhängt (Ester und Sander 2000). Daher muss der Algorithmus in der Praxis stets mehrfach durchgeführt werden, um die bestmögliche Lösung zu finden.

Zusammenfassung: Clusteralgorithmen

Zusammenfassend ist in Tabelle 2-3 eine Übersicht der vorgestellten Clusteralgorithmen dargestellt. Darin sind deren Vorgehen bei der Bildung der Cluster sowie deren Vor- und Nachteile gegenübergestellt. Für die Eignung der Clusteralgorithmen zur Analyse komplexer Produktportfolios ist es schwierig eine Aussage zu treffen, da die Form der Cluster kaum abgeschätzt werden können.

Tabelle 2-3: Übersicht Clusteralgorithmen

Algorithmus	Inhalt	Vor- und Nachteile
k-Means *distanzbasiert*	Identifikation von Clustern, die vom jeweiligen Zentrum repräsentiert werden	+ Geringer Rechenaufwand - Lokales Minimum und daher Abhängig von Initialisierung - Ungeeignet für Cluster mit unterschiedlicher Größe und Dichte
Linkage *hierarchisch*	Mehrschichtige Anordnung von Clustern und Subclustern sowie deren Rangordnung	+ Anzahl der gesuchten Cluster muss nicht vorab festgelegt werden - Identifikation eines lokalen Optimums
DBSCAN *dichtebasiert*	Regionen mit hoher Dichte von Datenpunkten, separiert durch Regionen mit geringer Dichte	+ Identifikation von Clustern beliebiger Form - Dichte stark variiert oder eng an Bereichen mit Rauschen
EM *probabilistisch*	Datenpunkte eines Clusters entstammen der gleichen Wahrscheinlichkeitsverteilung	+ Kugelförmige und elliptische Cluster - Stark von Initialisierung abhängig

2.2.5.2.2 Assoziationsalgorithmen

Die Assoziationsanalyse hat ihren Ursprung in der Analyse von Warenkörben im Einzelhandel und ermittelt Muster (z. B. Elemente, Subsequenzen oder Substrukturen), welche in einem Datensatz häufig auftreten (Han et al. 2012). Agrawal et al. (1993) definieren die grundsätzliche Problemstellung einer Assoziationsanalyse wie folgt: Sei $Z = I_1, I_2, ..., I_m$ eine Menge von binären Attributen, genannt Elemente. T sei eine Datenbank mit Transaktionen. Jede Transaktion t wird als binärer Vektor dargestellt, wobei $t[k] = 1$ ist, wenn t das Element I_k beinhaltet, und $t[k] = 0$, wenn nicht. Unter einer Assoziationsregel wird eine Implikation der Form $X \rightarrow I_j$ verstanden, wobei X eine Menge der Elemente in Z und I_j ein einzelnes Element in Z ist, das in X nicht vorhanden ist. Die Regeln bei einer Assoziationsanalyse werden mithilfe des Supports ermittelt. Der Support gibt an, wie oft eine Regel im gesamten Datensatz vorkommt (Tan et al. 2019). Im Folgenden wird auf die wichtigsten Vertreter von Assoziationsalgorithmen eingegangen.

Agrawal, Imielinski, Swami (AIS)

Der AIS Algorithmus war der erste Algorithmus, der zur Ermittlung von Assoziationsregeln eingesetzt wurde (Agrawal et al. 1993). In Abbildung 2-26 ist beispielhaft das Vorgehen beim AIS abgebildet. Die Support-Werte für jedes einzelne Element werden

beim ersten Durchlauf durch den Datensatz berechnet. Die Elemente, welche einen vorgegebenen Support-Wert unterschreiten, werden anschließend aus der Liste gestrichen. Für die verbleibenden Elemente werden im zweiten Durchlauf durch den Datensatz die Support-Werte für alle Kombinationen aus zwei Elementen berechnet und mit dem minimalen Support-Wert verglichen. Die Generierung von Kandidatenmengen und häufigen Mengen wird so lange wiederholt, bis eine der Mengen leer ist oder ein Abbruchkriterium erreicht wird (Kumbhare und Chobe 2014).

Der größte Nachteil des AIS Algorithmus besteht darin, dass bei jedem Durchlauf viele Support-Werte berechnet werden, welche niedriger als der vorgegebene Schwellenwert sind. Gleichzeitig erfordert der Algorithmus viele Durchläufe durch den gesamten Datensatz (Zhao und Bhowmick 2003).

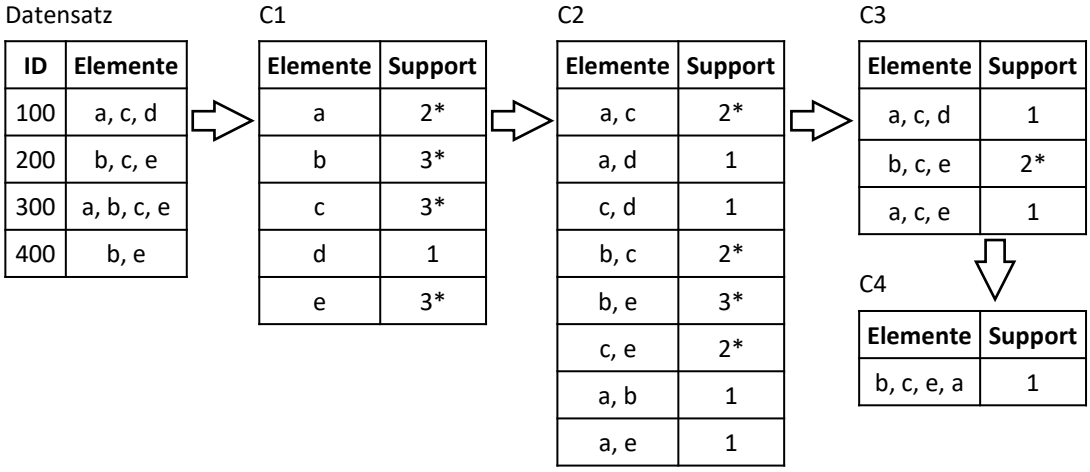
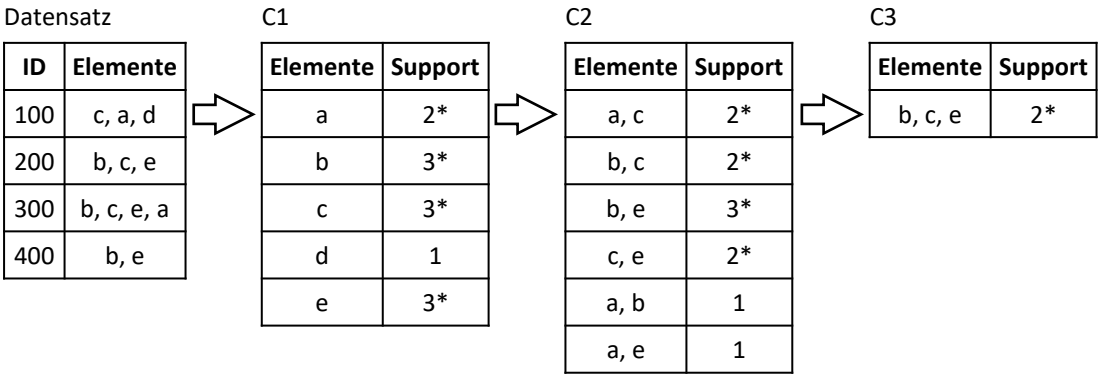


Abbildung 2-26: Vorgehen AIS Algorithmus (min Support=2) in Anlehnung an Khurana und Sharma (2013)

Apriori

Der Apriori Algorithmus wurde von Agrawal und Srikant (1994) eingeführt. Er versucht im Vergleich zum AIS Algorithmus die Anzahl der zu berechnenden Support-Werte pro Durchlauf zu verringern. Er beruht auf der Annahme, dass jede Teilmenge einer häufig auftretenden Menge an Elementen ebenfalls häufig auftreten muss. Daher werden Elemente, welche den vorgegebenen Support-Wert unterschreiten auch für die folgenden Berechnungsschritte verworfen. In Abbildung 2-27 ist das Vorgehen des Apriori Algorithmus beispielhaft dargestellt. Es werden zuerst die Support-Werte aller einzelnen Elemente bestimmt. Anschließend wird für die Elemente, welche den minimalen Support-Wert überschreiten, die Support-Werte aller Teilmengen aus zwei Elementen mit einem Durchlauf durch den Ausgangsdatensatz bestimmt. Im Vergleich zum AIS werden keine Kombinationen mit Elementen, welche zuvor den Schwellenwert unterschritten haben, betrachtet. Dies wird so lange wiederholt, bis keine Kandidaten mehr übrig sind oder ein Abbruchkriterium erreicht wird.



ID	Elemente
100	c, a, d
200	b, c, e
300	b, c, e, a
400	b, e

Elemente	Support
a	2*
b	3*
c	3*
d	1
e	3*

Elemente	Support
a, c	2*
b, c	2*
b, e	3*
c, e	2*
a, b	1
a, e	1

Elemente	Support
b, c, e	2*

Abbildung 2-27: Vorgehen Apriori Algorithmus (min Support=2) in Anlehnung an Khurana und Sharma (2013)

Der Aufwand für die Berechnung der Support-Werte ist beim Apriori gegenüber dem AIS geringer. Jedoch ist dieser für die Analyse großer Datensätze immernoch hoch. Außerdem muss der Ausgangsdatensatz mehrfach durchlaufen werden.

AprioriTid und AprioriHybrid

Der AprioriTid und AprioriHybrid sind Abwandlungen des Apriori Algorithmus, welche versuchen die Leistungsfähigkeiten in den späteren Durchläufen zu verbessern. Beim AprioriTid wird der Datensatz nach dem ersten Durchgang nicht mehr für die Ermittlung der Support-Werte verwendet (Agrawal und Srikant 1994). Stattdessen werden die Kandidaten des vorangegangenen Schrittes C_k durchsucht, um die Support-Werte zu ermitteln (Khurana und Sharma 2013). In den ersten Durchläufen kann der Kandidatendatensatz jedoch größer als der Ausgangsdatensatz sein und daher mehr Rechenleistung erfordern. In späteren Durchläufen wird die Größe schnell kleiner als der Ausgangsdatensatz (Li et al. 2005). Wie beim Apriori Algorithmus werden beim AprioriTid viele Durchläufe durch den Datensatz benötigt.

Der AprioriHybrid verwendet in den ersten Phasen den Ausgangsdatensatz zur Ermittlung der Support-Werte, wie beim Apriori Algorithmus, und schaltet auf den Kandidatendatensatz, wie beim AprioriTid, um, wenn dieser kleiner als der Ausgangsdatensatz ist (Girotra et al. 2013). Dadurch wird die Rechenleistung zur Bestimmung der Support-Werte reduziert. Dennoch werden wie bei den anderen bisher vorgestellten Algorithmen auch beim AprioriHybrid viele Durchläufe durch den Datensatz benötigt.

Frequent Pattern (FP) Growth

Der FP-Growth Algorithmus versucht den Ausgangsdatensatz in eine stark komprimierte baumartige Datenstruktur zu überführen und dadurch die beiden Nachteile des Apriori Algorithmus zu umgehen (Han et al. 2000). Der Durchlauf durch den sogenannten FP-Baum erfordert zum einen weniger Rechenleistung und zum anderen ist keine Ermittlung von Kandidatenelementen erforderlich. Für die Erstellung des Baumes

werden zuerst alle Elemente in den Transaktionen bestimmt und in absteigender Reihenfolge sortiert. Anschließend wird der Baum gebildet indem jede Transaktion ausgehend von der Wurzel des Baumes aufgetragen wird (siehe Abbildung 2-28). Jedem Knoten wird die Anzahl der Transaktionen zugeordnet, die diesen Knoten durchlaufen.

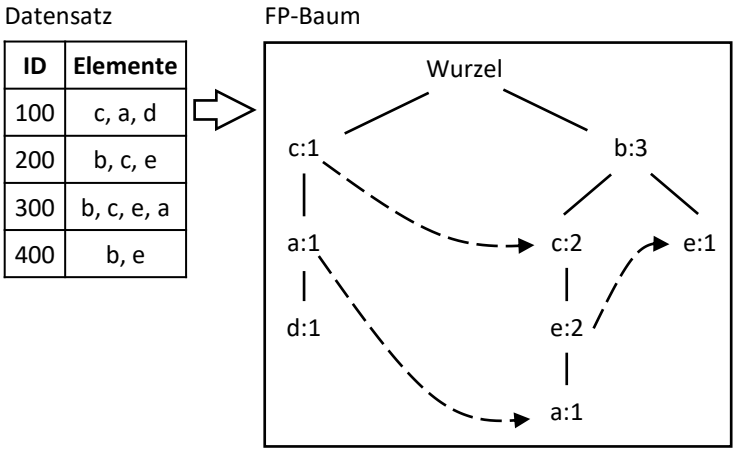


Abbildung 2-28: Bildung de FP-Growth Baums in Anlehnung an Han et al. (2000)

Nachdem alle Transaktionen im Baum erfasst wurden, kommt es zur Ableitung der Assoziationsregeln (Han et al. 2000). In Tabelle 2-4 ist die Ableitung der Assoziationsregeln für einen minimalen Support-Wert von zwei abgebildet. Dabei wird beginnend mit dem Element mit dem geringsten Support-Wert der Baum von oben nach unten durchlaufen und dessen bedingte Muster identifiziert. Daraus wird ein reduzierter FP-Baum gebildet, welcher die Häufigkeit zu anderen Elementen darstellt. Ausgehend davon werden die Regeln abgeleitet. Dies wird bis zum zweithäufigsten Element wiederholt.

Tabelle 2-4: Ableitung der Assoziationsregeln aus dem FP-Baum (min Support=2) in Anlehnung an Han et al. (2000)

Elemente	Bedingte Muster	Bedingter FP-Baum	Assoziationsregeln
d	{c,a:1}	<c:1,a:1>	-
a	{c:1},{b,c,e:1}	<c:2,b:1,e:1>	{c,a:2}
e	{b,c:2},{b:1}	<b:3,c:2>	{e,b:3},{e,c:2},{e,c,b:2}
c	{b:2}	<b:2>	{c,b:2}

Der FP-Growth reduziert die Durchläufe durch die Daten auf zwei. Im ersten Schritt wird der FP-Baum erzeugt und im zweiten Schritt die Assoziationsregeln ermittelt. Aufgrund der kompakten Struktur des Baumes und der nicht erforderlichen Ermittlung

der Kandidaten wird weniger Rechenleistung und Speicherplatz benötigt. Jedoch ist der Algorithmus schwerer verständlich und zu implementieren.

Zusammenfassung: Assoziationsalgorithmen

Eine Übersicht der vorgestellten Algorithmen, deren Inhalte sowie der Vor- und Nachteile sind in Tabelle 2-5 dargestellt. Aus diesen geht hervor, dass aufgrund der Vielzahl an Merkmale, Komponenten sowie Produktvarianten komplexer Produktportfolios für deren Analyse mit einer Assoziationsanalyse in erster Linie der AprioriHybrid sowie der FP-Growth in Frage kommen.

Tabelle 2-5: Übersicht Assoziationsalgorithmen

Algorithmus	Inhalt	Vor- und Nachteile
AIS	Für alle möglichen Elemente werden in jedem Schritt die Support-Werte bestimmt	+ Einfach zu implementieren - Viele Durchläufe - Hoher Aufwand zur Berechnung der Support-Werte
Apriori	Elemente, welche einen vorgegebenen Support-Wert unterschreiten, werden verworfen	+ Einfach zu implementieren + Geringerer Aufwand als beim AIS - Viele Durchläufe durch den Datensatz
AprioriTid	Kandidatendatensatz des vorangegangenen Schritts wird für die Ermittlung der Support-Werte verwendet	+ Geringere Rechenleistung in später Phase - Hohe Rechenleistung in früher Phase - Viele Durchläufe durch den Datensatz
AprioriHybrid	Support-Werte werden zuerst mit dem Ausgangsdatensatz und dann mit dem Kandidatendatensatz ermittelt	+ Geringere Rechenleistung - Viele Durchläufe durch den Datensatz
FP-Growth	Überführung des Datensatzes in eine komprimierte Baumstruktur	+ Zwei Durchläufe durch den Datensatz + Weniger Rechenleistung und Speicherplatz erforderlich - Schwieriger zu verstehen und implementieren

2.2.6 Zusammenfassung: Machine Learning

Machine Learning ist ein Teilgebiet der künstlichen Intelligenz. Es beinhaltet Verfahren, die automatisiert Muster in Daten erkennen und die aufgedeckten Muster nutzen, um zukünftige Ereignisse vorherzusagen. Im Vergleich zum traditionellen Ansatz werden beim Machine Learning Daten anstatt der Erfahrung von Experten genutzt, um Wissen zu generieren. Fokus dieser Arbeit liegt auf dem überwachten und unüberwachten Machine Learning sowie den Verfahren Regressions-, Klassifikations-, Cluster- und Assoziationsanalyse. Die Verfahren beinhalten unterschiedliche Algorithmen, welche in Abhängigkeiten der Daten und Aufgaben verschiedene Charakteristiken besitzen. Eine Gegenüberstellung der Algorithmen anhand hergeleiteter Auswahlkriterien findet im Rahmen des Frameworks in Kapitel 5 statt. Der Einsatz von Machine Learning Verfahren ist nur eine Phase in einem Datenanalyseprozess. Zuerst muss ein Verständnis für die Domäne und die Daten generiert sowie die Daten vorbereitet werden. Nach dem Training der Modelle müssen diese evaluiert und eingesetzt werden. Ein Datenanalyseprozess, welcher die industrielle Anwendung von Machine Learning ermöglicht und im Folgenden genutzt wird, ist der CRISP-DM.

3 Ansätze zur Analyse von Produktportfolios mittels Machine Learning

In diesem Kapitel wird der Stand der Forschung zum Einsatz von Machine Learning zur Analyse komplexer Produktportfolios näher beleuchtet. Dies beinhaltet die Analyse der bisherigen Ansätze aus der Literatur und die Ableitung des Forschungsbedarfs anhand zuvor definierter Kriterien.

3.1 Kriterien zur Bewertung bestehender Ansätze

Die Zielsetzung dieses Forschungsvorhabens ist die Entwicklung eines Frameworks zur systematischen Generierung von Wissen für die Analyse komplexer Produktportfolios mittels Machine Learning. Für die Entwicklung neuer Ansätze in der Produktentwicklung ist es erforderlich, die bestehenden Artefakte, welche ein ähnliches Ziel verfolgen, zu untersuchen und im Hinblick auf diese den Forschungsbedarf abzuleiten. Die Ableitung des Forschungsbedarfs erfordert die Definition von Bewertungskriterien für den Stand der Forschung. In Anlehnung an Jank (2021) wird zwischen den objektorientierten und zielorientierten Kriterien unterschieden. Die **objektorientierten Kriterien** berücksichtigen den Betrachtungshorizont und den Anwendungsbereich des Frameworks. Die Kriterien für diese Arbeit leiten sich aus den beschriebenen Grundlagen und dem Fokus der Arbeit aus Kapitel 1 und 2 sowie einer Betrachtung der Herausforderungen beim Einsatz von Machine Learning in der Literatur und Industrie ab (siehe Mehlstäubl et al. 2022b):

- **Analyse komplexer Produktportfolios**: Das Framework soll auf Produktportfolios mit einer besonders großen Produktportfoliobreite und -tiefe anwendbar sein.
- **Berücksichtigung verfügbarer Daten**: Das Framework soll für die Implementierung ausschließlich Daten berücksichtigen, welche im Lebenszyklus eines Produkts erzeugt werden.
- **Betrachtung der operativen Produktportfoliogestaltung**: Das Framework soll Wissen über Produktvarianten auf der Ebene der operativen Produktportfoliogestaltung generieren und einsetzen.
- **Einbeziehung verschiedener Machine Learning Verfahren**: Das Framework soll es ermöglichen, unterschiedliche überwachte und unüberwachte Lernverfahren einzusetzen. Dadurch können Anwender anforderungsspezifische Anwendungsfälle zur Analyse deren Produktportfolio implementieren.
- **Übertragbarkeit auf unterschiedliche Produktportfolios**: Das Framework soll eine allgemeingültige Beschreibung für die Analyse von Produktportfolios unterschiedlicher Unternehmen und deren Produktdatenmodelle besitzen.

Die **zielorientierten Kriterien** leiten sich aus der beschriebenen Problemstellung und Industrieperspektive aus Kapitel 1 ab und berücksichtigen die Ergebnisse der Phasen des CRISP-DM (vgl. Kapitel 2.2.4) für die industrielle Anwendung von Machine Learning:

- **Systematisierung der Wissensbedarfe**: Das Framework soll Wissensbedarfe zur Analyse komplexer Produktportfolios auf der Ebene von Produktvarianten aufzeigen und systematisch darstellen.
- **Datenbasierte Beschreibung komplexer Produktportfolios**: Das Framework soll eine datenbasierte Beschreibung komplexer Produktportfolios enthalten.
- **Vorbereitung von Produktportfoliodaten**: Das Framework soll den Nutzern bei der systematischen Vorbereitung der Produktportfoliodaten für die anschließende Modellierung helfen.
- **Auswahl und Evaluation von Algorithmen**: Das Framework soll eine Unterstützung bei der Auswahl der Algorithmen sowie der Bewertung deren Güte bieten.
- **Einsatz des Wissens zur Analyse komplexer Produktportfolios**: Das Framework soll aufzeigen, wie die erzeugten Machine Learning Modelle für die Analyse komplexer Produktportfolios eingesetzt werden können.

3.2 Bestehende Ansätze aus der Literatur

Im Folgenden wird auf bestehende Arbeiten aus der Literatur eingegangen, welche überwachte oder unüberwachte Lernverfahren einsetzen, um Produktportfolios zu analysieren. Diese beruhen auf der systematischen Literaturrecherche von Mehlstäubl et al. (2021b). Das Vorgehen bei der Literaturrecherche sowie der Auswahl der Ansätze ist im Forschungsvorgehen dieser Arbeit in Kapitel 4.2 beschrieben. Die Ansätze werden hinsichtlich der eingesetzten Lernverfahren in drei Themengebiete unterteilt: Ansätze zum Einsatz überwachter Lernverfahren (I), unüberwachter Lernverfahren (II) und kombinierter Lernverfahren (III). Im Folgenden werden die Zielsetzung, der Inhalt, die Anwendung sowie die Stärken und Schwächen der Ansätze beschrieben, bevor in Kapitel 3.3 die Erfüllung der Kriterien bewertet und der Forschungsbedarf abgeleitet wird.

3.2.1 Einsatz überwachter Lernverfahren

Datengetriebene Entscheidungsbaum-Klassifikation zur Optimierung der Produktportfoliogestaltung nach Tucker und Kim a)

Tucker und Kim (2009) stellen eine Methode vor, die unter Verwendung eines Entscheidungsbaums eine Reihe von Produktvarianten hinsichtlich der Zahlungsbereitschaft der Kunden analysiert. Sie verwenden hierfür einen C4.5 Entscheidungsbaum Algorithmus zur Analyse eines Umfragedatensatzes (Quinlan 1986). Der Umfrage-

datensatz besteht aus Kundenmerkmalen als Eingangsmerkmale und verschiedene Preiskategorien als Zielvariablen und muss für die Anwendung der Methode erst generiert werden. Nach der Datenvorbereitung und der Anwendung des Entscheidungsbaums werden die generierten Konzepte durch Entwickler analysiert und eine Auswahl auf Basis des voraussichtlichen Gewinns getroffen (siehe Abbildung 3-1). Eine Anwendung findet im Rahmen einer Fallstudie von Mobiltelefonen statt. Es wird ein synthetischer Kundendatensatz mit 40 000 Produktvarianten und 576 eindeutigen Merkmalskombinationen (die gesamte Menge möglicher Produktkonzepte) auf 46 Produktkonzepte eingegrenzt und anschließend durch Konstrukteure analysiert. Der Ansatz zeigt, dass durch den Einsatz einer Klassifikationsanalyse Preiskategorien für Produktvarianten auf der Ebene der operativen Produktportfoliogestaltung automatisiert prognostiziert werden können. Durch die Methode kann ein Produktportfolio auf eine gewisse Anzahl an Standardprodukten reduziert werden. Jedoch sind Kunden es heutzutage gewohnt, sich die Produkte in Abhängigkeit ihrer Bedürfnisse individuell zu konfigurierten, so dass es nicht wettbewerbsfähig ist, eine bestimmte Anzahl an Standardprodukten anzubieten.

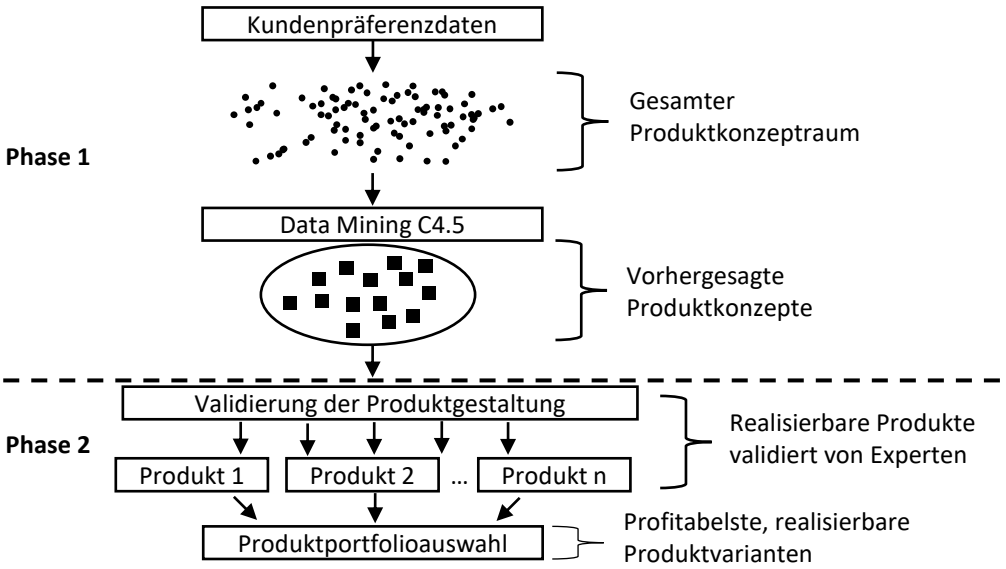


Abbildung 3-1: Optimierung der Produktportfoliogestaltung nach Tucker und Kim (2009)

Trend Mining für die prädiktive Produktgestaltung nach Tucker und Kim b)

Tucker und Kim (2011a; 2011b) stellen einen mehrstufigen Ansatz vor, der Veränderungen in den Verbraucherpräferenzen im Laufe der Zeit erfasst. Dafür setzen sie den Zeitreihenalgorithmus Holt-Winters mit exponentieller Glättung ein (Chatfield 1978), um ein Nachfragemodell der Merkmale auf Basis von Kundenrezensionen zu erzeugen. Die Vorhersagen stellen die Grundlage für die anschließende Einteilung der Merkmale in Standard, Nicht-Standard oder veraltet dar. Eine Anwendung wird im Rahmen einer Fallstudie anhand eines Datensatzes eines Herstellers von Mobiltelefonen

durchgeführt. Der Datensatz enthält 12 000 Instanzen mit sechs kategorischen Merkmalen und insgesamt 16 Merkmalsausprägungen. Als Zielvariable wird der Preis verwendet, welcher fünf Ausprägungen annehmen kann. Der Ansatz demonstriert, wie mit Machine Learning die zeitliche Entwicklung der Nachfrage nach einzelnen Merkmalen modelliert werden kann. Die Datengrundlage basiert auf Kundenrezensionen, welche nicht in jedem Unternehmen zur Verfügung stehen. Zudem ist das in der Anwendung beschriebene Produktportfolio wenig komplex.

Vorhersage des Produktpreises mit Hilfe neuronaler Netze nach Boyarkin et al.

Boyarkin et al. (2019) stellen ein Modell zur Vorhersage des Preises unter Verwendung eines neuronalen Netzes vor (siehe Abbildung 3-2). Dabei werden die kontinuierlichen Merkmale x_i (z. B. Leistung oder Druck) eines Produkts bei der Regressionsanalyse als Eingangsmerkmale verwendet. Der Preis Y wird als Zielvariable herangezogen. Weitere Produktportfoliodaten und die Schritte für deren Vorbereitung werden nicht beschrieben. Das trainierte Modell wird eingesetzt, um die Preise für noch nicht verkaufte Produkte zu bestimmen. Es findet eine Anwendung des Ansatzes auf Pumpensysteme statt. Dabei werden neun kontinuierliche Eigenschaften als Eingangsmerkmale herangezogen. Der Datensatz umfasste 520 Pumpenelemente, von denen 500 für das Training und 20 zum Testen genutzt werden. Im Vergleich zu den zuvor vorgestellten Arbeiten verwenden Boyarkin et al. (2019) in Unternehmen vorhandene Vertriebsdaten. Ein Vergleich zwischen unterschiedlichen Algorithmen findet jedoch ebenfalls nicht statt. Außerdem wird der Ansatz an einem wenig komplexen Produktportfolio angewendet.

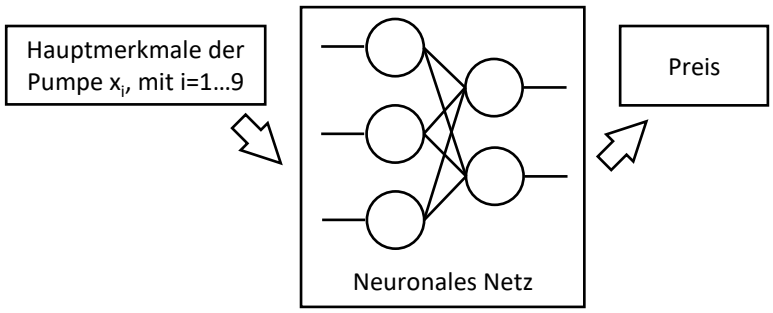


Abbildung 3-2: Datenverarbeitung mit dem neuronalen Netz nach Boyarkin et al. (2019)

Implementierung von neuronalen Netzen im Portfoliomanagement zur Unterstützung von Entscheidungsprozessen nach Riesener et al.

Riesener et al. (2019b) stellen eine Methode vor, bei der die Korrelationen zwischen portfoliorelevanten unternehmensspezifischen Indikatoren ermittelt und zur Vorhersage der zukünftigen Trends verwendet werden. Dies soll es Unternehmen ermöglichen, die zukünftige Entwicklung ihres Produktportfolios zu prognostizieren und dieses proaktiv zu verwalten. Sie orientieren sich bei ihrem Ansatz an dem KDD-Prozess

und beschreiben die Schritte zur Datenvorbereitung und verwenden für die Zeitreihenanalyse ein neuronales Netz (siehe Abbildung 3-3). Riesener et al. (2020) erweitern das Vorgehen um Handlungsempfehlungen zur Steuerung des Produktportfolios. Die Methode wird anhand einer Fallstudie bei einem variantenreichen Hersteller von Maschinenteilen angewendet. Als Eingangsmerkmale nutzen sie die vier Kennzahlen Produktionskosten, Lieferverspätung, zusätzliche Zeit sowie Portfolio-Fitness-Index und als Zielvariable die Gewinnmarge. Die zur Verfügung gestellten Daten erstreckten sich über insgesamt 49 Monate. Jeder Indikator wird für jeden Monat über eine Produktlinie gemittelt. Durch den Ansatz wird aufgezeigt, wie komplexe Produktportfolios auf der Ebene der strategischen Produktportfoliogestaltung gesteuert werden kann. Einzelne Produktvarianten und deren Elemente werden jedoch nicht betrachtet.

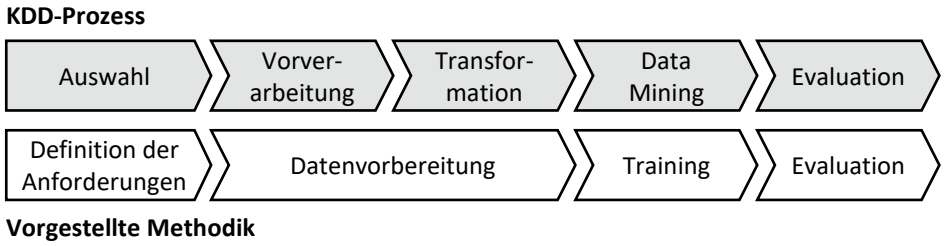


Abbildung 3-3: Unterstützung von Entscheidungsprozessen im Produktportfoliomanagement nach Riesener et al. (2019b)

3.2.2 Einsatz unüberwachter Lernverfahren

Ansatz zur Erstellung allgemeiner Stücklisten für die Variantenkonstruktion nach Romanowski und Nagi

Romanowski und Nagi (2004) stellen einen Ansatz zur Bildung von generischen Stücklisten (GBOMs) vor, welche zum einen die verschiedenen Produktvarianten einer Produktfamilie darstellen und zum anderen die Suche nach ähnlichen Produktvarianten sowie die Konfiguration neuer Produktvarianten erleichtert (siehe Abbildung 3-4). Zuerst findet eine Generalisierung der Kaufteile statt. Dafür werden mit Text Mining die wichtigsten Identifikationsmerkmale bestimmt und Ähnlichkeitswerte berechnet. Anschließend werden die BOMs sowie Unterbaugruppen mit einem k-Medoid Clusteralgorithmus gruppiert (Romanowski und Nagi 2005). Beim k-Medoid Algorithmus werden im Vergleich zum k-Means reale Datenpunkte als Centroide festgelegt (Kaufman und Rousseeuw 2009). Im dritten Schritt findet eine Baumvereinigung zur Vereinheitlichung der Stücklisten in den einzelnen Clustern statt. Abschließend werden mit Hilfe eines Apriori Algorithmus Assoziationen in Form von Konfigurations- und Entwicklungseinschränkungen als Regeln auf der Ebene der operativer Produktportfoliogestaltung definiert. Der Ansatz wird im Rahmen einer Fallstudie mit Krankenpflegerrufsystemen angewendet. Die Systeme haben insgesamt vier Hauptproduktfamilien mit jeweils mehreren hundert Variantenstücklisten. Davon wird eine Stichprobe von 405

Stücklisten für die Bildung der GBOMs aus einer der wichtigsten Produktfamilien verwendet. Die GBOMs helfen bei der Konfiguration neuer Varianten und unterstützen die Suche nach ähnlichen Komponenten. Dadurch können Ähnlichkeiten zwischen den Produktvarianten identifiziert werden, jedoch findet keine Unterstützung der Rationalisierung des Produktportfolios statt. Des Weiteren wird der Ansatz an einem reduzierten Datensatz angewendet, wobei auf die Vorbereitung der Daten sowie andere Algorithmen und deren Evaluation nicht näher eingegangen wird.

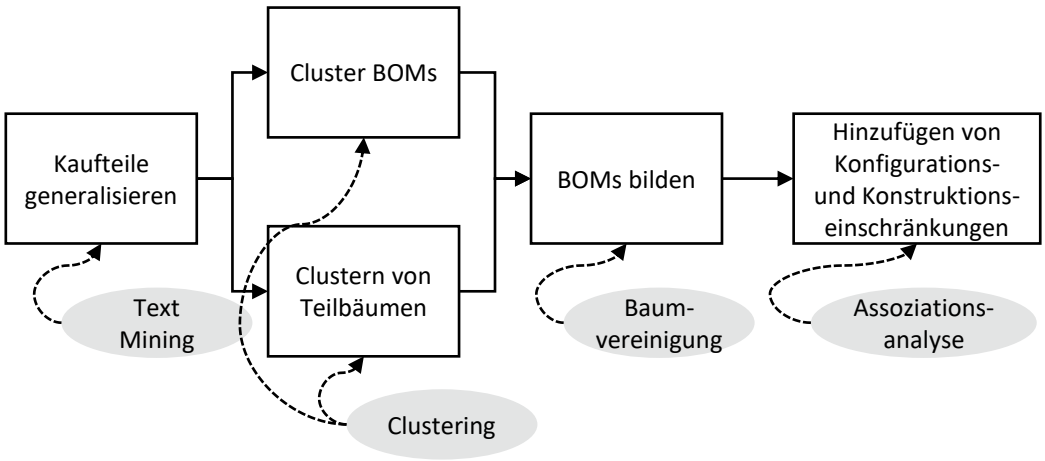


Abbildung 3-4: Methode zur Bildung generischer GBOMs nach Romanowski und Nagi (2004)

Planung von Produktkonfigurationen auf der Grundlage von Vertriebsdaten nach Kusiak et al.

Kusiak et al. (2007) nutzen die Konfigurationen von verkauften Produktvarianten, um das Produktportfolio auf Basis der Kundenpräferenzen neu zu planen. Die Ähnlichkeit der Kunden wird anhand der von ihnen gewählten Merkmalsausprägungen bewertet. Weitere Produktportfoliodaten und die Schritte zur Datenvorbereitung werden nicht beschrieben. Es findet zuerst ein Clustering mit einem modifizierten k-Means Algorithmus statt. Da die Cluster unterschiedliche Größen und Eigenschaften besitzen und daher nicht jedes vom Algorithmus erzeugte Cluster für eine Produktkonfiguration geeignet ist, findet eine manuelle Auswahl der besten Cluster statt. Die Centroide der gebildeten Cluster werden mit Metriken zur Bestimmung der Konfigurationsqualität bewertet und sortiert. Der beschriebene Ansatz wird anhand einer industriellen Fallstudie mit 6 216 Vertriebsdaten von Nutzfahrzeugen, die über einen Zeitraum von einem Jahr erhoben wurden, demonstriert. Der Datensatz enthält die verkauften Merkmalsausprägungen sowie den Preis des Fahrzeugs. Der Kunde konnte bei der Konfiguration des Fahrzeugs zehn Merkmale (z. B. Fahrerhaus oder Motor) auswählen. Die Fahrzeuge werden anschließend hinsichtlich drei Gesichtspunkten geclustert. Es fand ein Clustering mit lediglich den verkauften Merkmalen, eines mit den Preisen der Merkmale und eines mit sowohl den Merkmalen als auch den Preisen statt. Daneben wird ein gewichtetes Clustering durchgeführt, bei dem jedem Merkmal ein Wichtigkeitswert

zugeordnet wird. Wie bei Tucker und Kim (2009) wird durch den Ansatz versucht, das Produktportfolio auf wenige Produktvarianten zu reduzieren, was im Widerspruch zu den individuellen Bedürfnissen der Kunden steht. Eine Anwendung findet lediglich an beispielhaften Nutzfahrzeugdaten statt, weshalb die Komplexität des analysierten Produktportfolios im Vergleich zu den in Kapitel 1.2 beschriebenen Produktportfolios gering ist.

Optimierung von Produktkonfigurationen mit einem Data Mining Ansatz nach Song und Kusiak

Song und Kusiak (2009) wenden eine Assoziationsanalyse an, um Regeln aus historischen Vertriebsdaten zu extrahieren und nutzen diese für die Bildung von Unterbaugruppen und Hauptproduktkonfigurationen (siehe Abbildung 3-5). Die mit einem Apriori Algorithmus ermittelten Regeln werden anschließend für die Definition von Unterbaugruppen eingesetzt. Bei einer zu großen Anzahl an Regeln wird der ROC Clusteralgorithmus herangezogen (King 1980), um ähnliche Regeln zusammenzufassen und so deren Anzahl zu verringern. Danach werden anhand der Unterbaugruppen die Produktkonfigurationen abgeleitet. Der Ansatz wird auf vereinfachte Automobilvarianten angewendet, welche neun Merkmale besitzen. Jedes Merkmal hat mehrere Merkmalsausprägungen, die von den Kunden ausgewählt werden müssen. Die gebildeten Hauptproduktkonfigurationen werden mit den Centroiden eines k-Means Clusteralgorithmus verglichen.

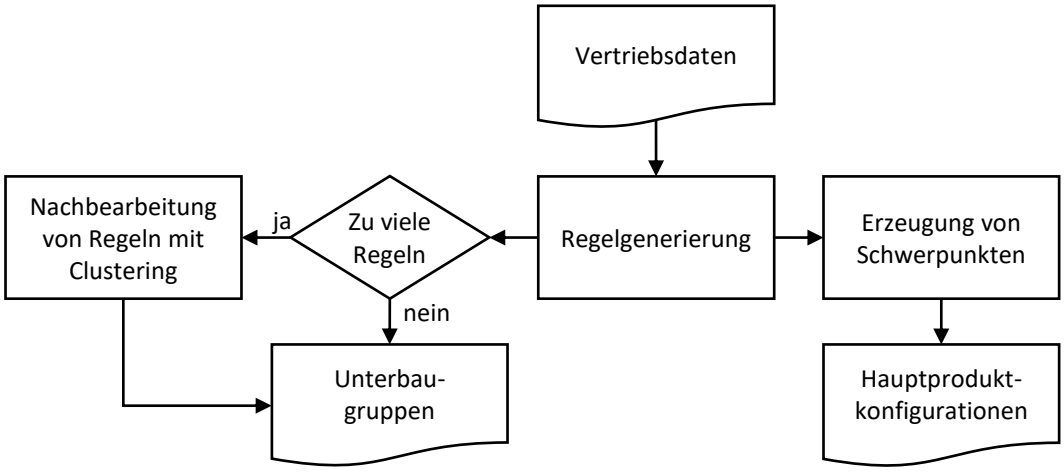


Abbildung 3-5: Prozess zum Mining von Vertriebsdaten zur Ermittlung von Mustern in Unterbaugruppen und Hauptproduktkonfigurationen nach Song und Kusiak (2009)

Methode zur Wissensentdeckung für die Unterstützung der Entwicklung von Produktfamilien nach Moon et al.

Moon et al. (2010) stellen eine Methode zur Wissensentdeckung vor, welche die Entwicklung von Produktfamilien durch den Einsatz von Cluster- und Assoziationsanalysen unterstützt (siehe Abbildung 3-6). Sie beschreiben die Produkte des Produktportfolios

auf drei Hierarchieebenen: Module, Funktionen und Attribute (siehe Moon et al. 2006). Zuerst wird ein c-Means Algorithmus eingesetzt, um die Funktionen mit ihren Attributen zu Modulen zu clustern. Anschließend werden Assoziationsregeln zwischen den einzelnen Attributen ermittelt, welche als Entwicklungseinschränkungen dienen. Die Methode wird im Rahmen einer Fallstudie zur Generierung von Wissen für die Entwicklung einer Familie von Elektrowerkzeugen (Stichsäge, Kreissäge, Schleifmaschine, Bohrmaschine und Nagler) eingesetzt. Dabei werden 75 Funktionen mit jeweils fünf Attributen analysiert.

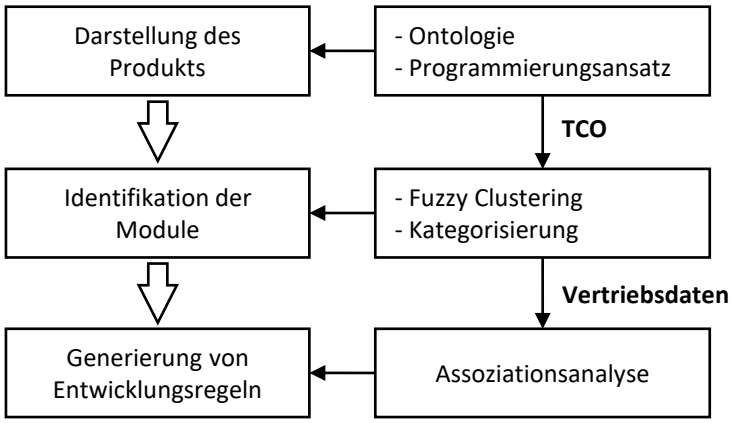


Abbildung 3-6: Prozess zur Wissensentdeckung für die Unterstützung der Entwicklung von Produktfamilien nach Moon et al. (2010)

ReliefF Eigenschaftsgewichtung und x-Means Clustering Methode für die Top-down Optimierung von Produktfamilien nach Tucker et al.

Tucker et al. (2010) beschreiben einen Ansatz, bei dem zuerst die Produkteigenschaften mit dem ReliefF Algorithmus gewichtet werden, um die Reihenfolge ihrer Bedeutung im Datensatz zu ermitteln. Der ReliefF ist eine Erweiterung des Relief Algorithmus (Kira und Rendell 1992) und ermöglicht die Anwendung auf Multi-Klassen-Klassifikationen und den Umgang mit fehlenden Werten innerhalb eines Datensatzes (Kononenko 1994). Es findet eine Gewichtung der Eigenschaften hinsichtlich ihres Einflusses auf den Preis der Produktvarianten statt. Anschließend wird der x-Means Algorithmus eingesetzt, um Gruppen ähnlicher Betriebszustände innerhalb des Datensatzes zu identifizieren (siehe Abbildung 3-7). Der x-Means Clusteralgorithmus ist eine Abwandlung des k-Means Algorithmus, bei dem die Anzahl der Cluster automatisiert bestimmt werden (Pelleg und Moore 2000). Der Ansatz wird im Rahmen einer Fallstudie an einem aerodynamischen Partikelabscheider mit acht kontinuierlichen Produkteigenschaften angewendet. Es wird ein Datensatz von 1 000 Betriebszuständen erzeugt, um die große Varianz an Anforderungen und Umweltbedingungen zu simulieren, die das breite Spektrum an Anwendungen eines aerodynamischen Partikelabscheiders abdecken. Durch das Clustering werden fünf Produktvarianten definiert.

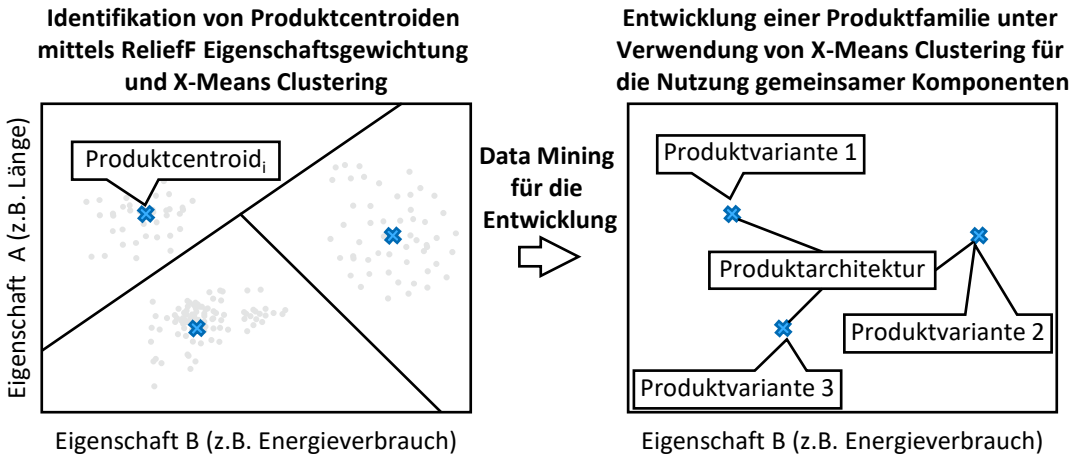


Abbildung 3-7 Top-down Optimierung von Produktfamilien

Marktsegmentierung zur Identifikation der idealen Punkte für die Entwicklung neuer Produkte nach Chan et al.

Chan et al. (2012) stellen eine Methode vor, um eine Marktsegmentierung auf der Grundlage von Kundenanforderungen durchzuführen (siehe Abbildung 3-8). Unscharfe (engl. fuzzy) Daten können nicht eindeutig einer Klasse zugeordnet werden. Stattdessen wird der Grad der Zugehörigkeit zu mehreren Klassen angegeben (Bandemer und Näther 2012). Die Methode erfordert zuerst eine Kundenbefragung, in der die Wichtigkeit der einzelnen Produktanforderungen auf einer Skala abgefragt werden. Anschließend werden die unscharfen Umfragedaten auf zwei Dimensionen mit einer Neural Network-based Principle Component Analyse (NPCA) reduziert und visualisiert (Denoeux und Masson 2004). Die NPCA stellt eine Erweiterung der Principal Component Analyse (PCA) dar und ermöglicht die Dimensionsreduktion von unscharfen Daten. Nachdem die Daten in Marktsegmente mit einem abgewandelten c-Means Clusteralgorithmus von Hung und Yang (2005) geclustert wurden, werden die Centroide der Marktsegmente als ideale Punkte für die Entwicklung neuer Produkte genutzt. Die Effektivität des entwickelten Ansatzes wird im Rahmen einer Fallstudie anhand von zehn Anforderungen an Digitalkameras demonstriert. Dabei werden 50 potenzielle Kunden mit einem Fragebogen befragt und anschießend die idealen Punkte für ein sowie drei Cluster bestimmt.

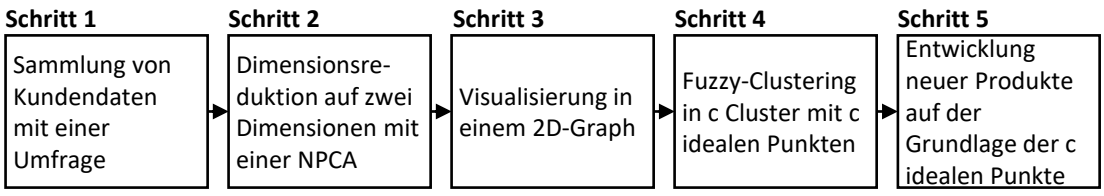


Abbildung 3-8: Fünf Schritte zur Identifikation der idealen Punkte für die Entwicklung neuer Produkte nach Chan et al. (2012)

Analyse der Portfoliokomplexität nach Neis

Neis (2015) versucht in seiner Dissertation die Variantenkomplexität zu reduzieren, indem er homogene Produktgruppen identifiziert und daraus eine Referenzproduktstruktur ableitet (siehe Abbildung 3-9). Auf Basis von Stücklisten wendet er hierfür ein zweistufiges Clustering an. Im ersten Schritt wird mit einem hierarchischen Clustering das Produktportfolio strukturiert und die Anzahl der Cluster bzw. Produktfamilien bestimmt. Im zweiten Schritt wird mit einem partitionierenden Clustering die Referenzproduktstruktur ermittelt. Dafür schlägt er ein k-Medoid Verfahren vor, bei dem der Medoid jedes Clusters als Ausgangspunkt für die Referenzproduktstruktur dient. Eine Anwendung des Ansatzes findet weder auf synthetische noch reale Daten statt.

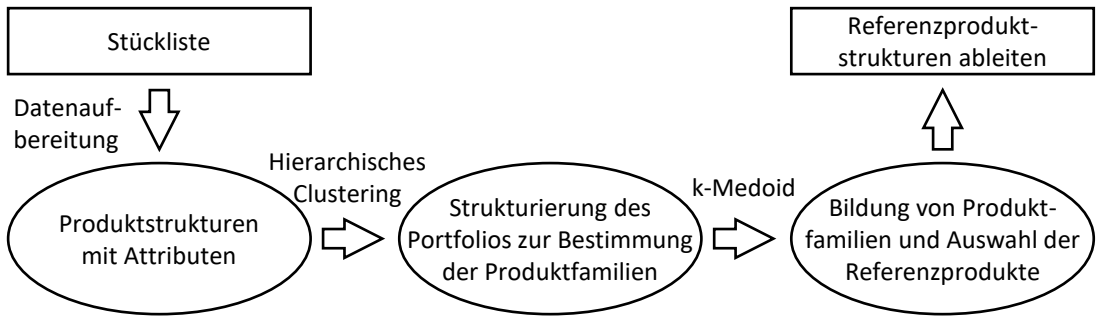


Abbildung 3-9: Ansatz zur Analyse der Portfoliokomplexität unter Anwendung von Clustering in Anlehnung an Neis (2015)

Assoziationsanalyse und auf kognitiven paarweisen Bewertungen basierende Portfolioanalyse für den Entwurf von Produktfamilien nach Wang

Wang (2019) stellen ein Framework mit dem Ziel einer nutzerorientierten Produktportfolioanalyse und Produktfamiliengestaltung vor (siehe Abbildung 3-10). Zunächst werden die Produktfamilien mit Hilfe sogenannter hedonischer Attribute (HAs) (z. B. Gehäusematerial) und utilitaristischer Attribute (UAs) (z. B. CPU-Leistung) beschrieben. Unter Verwendung des Apriori Algorithmus werden mit einer Assoziationsanalyse die wichtigsten HAs der Produktfamilien aus der Wahrnehmung der Nutzer bestimmt und mit den Metriken Support, Confidence und Lift bewertet. Im dritten Schritt wird ein kognitives paarweises Rating durchgeführt, um die Nutzerpräferenzen für UAs zu ermitteln. Schließlich wird die Technik der Präferenzordnung durch Ähnlichkeit mit der idealen Lösung (TOPSIS) verwendet, um die optimalen Portfolios von UAs zu priorisieren. Die Validierung findet im Rahmen einer Fallstudie an den Produktfamilien der Padbooks, Ultrabooks und Notebooks statt. Dabei werden fünf HAs und sechs UAs untersucht. Die Ergebnisse zeigen, dass die „Tastaturschnittstelle“, das „Gehäusematerial“ und die „Bildschirmgröße“ die wichtigsten HAs für die Differenzierung der Produktfamilie sind, während die „CPU-Leistung“ das wichtigste UA für die Konfiguration von Padbooks, Ultrabooks und Notebooks ist.

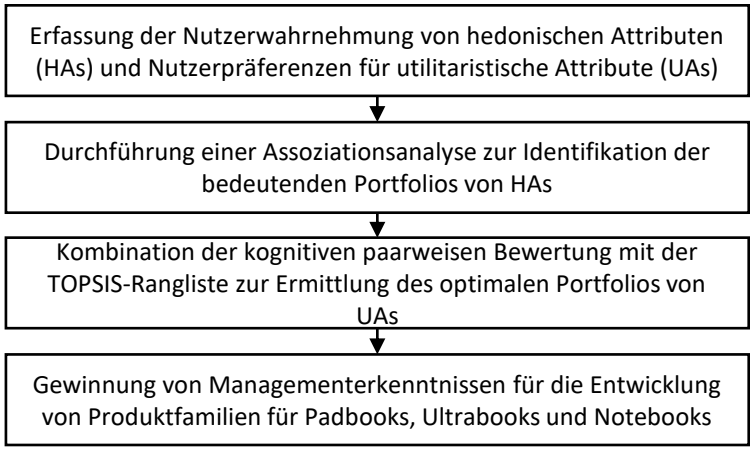


Abbildung 3-10: Framework zur Umsetzung von Produktdifferenzierung und Produktkonfiguration nach Wang (2019)

Dynamische Kundenpräferenzanalyse zur Identifikation des Produktportfolios mit Hilfe von sequentiellem Pattern Mining nach Yu et al.

Yu et al. (2017) stellen ein Modell für die sequentielle Mustererkennung vor, um implizites Wissen aus chronologischen Vertriebsdatensätzen zu ermitteln (siehe Abbildung 3-11). Zunächst werden historische Produktkonfigurationen gesammelt und in einer Datenbank akkumuliert. Jede Produktkonfiguration wird durch Kundenbedürfnisse (CNs), Produktspezifikationen (PSs) und einem Zeitstempel charakterisiert. Die Daten werden mit einem Sequential Pattern Mining unter Verwendung des AprioriAll Algorithmus analysiert (Agrawal und Srikant 1995).

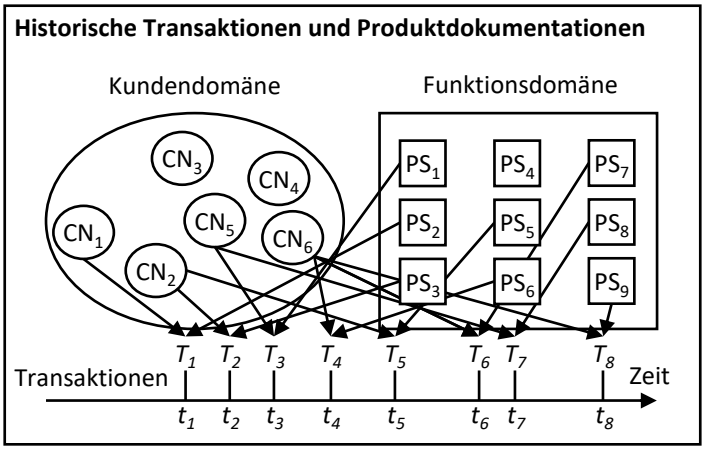


Abbildung 3-11: Produktportfolioidentifikation mit sequentiellem Pattern Mining nach Yu et al. (2017)

Die aufgedeckten sequentiellen Muster werden modifiziert und für die weitere Anwendung gespeichert. Entwickler können diese nutzen, um CNs zu bewerten, die zeitlichen Zusammenhänge von CNs und PSs zu verstehen, Markttrends zu ermitteln und PSs mit geringen Kosten und hoher Effizienz zu entwickeln. Zur Veranschaulichung der

vorgeschlagenen Methode wird eine vereinfachte Fallstudie zur Anpassung von Computern herangezogen. Die Computer besitzen drei CNs mit insgesamt sieben Ausprägungen sowie acht PSs mit in Summe 20 Ausprägungen.

3.2.3 Einsatz kombinierter Lernverfahren

Standardisierung von Komponenten, Produkten und Prozessen nach Agard und Kusiak

Agard und Kusiak (2004a, 2004b) beschreiben ein Vorgehen zur Standardisierung von Komponenten, Produkten und Prozessen mit unterschiedlichen Machine Learning Verfahren. Zur Standardisierung von Komponenten beschreiben sie den Einsatz einer Assoziationsanalyse, um Zusammenhänge zwischen Anforderungen in Form von Regeln zu identifizieren. Dies ermöglicht es, Anforderungen, welche regelmäßig zusammen nachgefragt werden, mit standardisierten Komponenten oder Komponentengruppen zu realisieren. Die Standardisierung von Produkten wird durch eine Marktsegmentierung mit einer Clusteranalyse erreicht. Sie clustern Anforderungen, um ähnliche Kunden zu gruppieren. Dies ermöglicht es, anschließend die Anforderungen für ein standardisiertes Produkt für jedes Cluster manuell zu definieren. Sie beschreiben ebenfalls ein überwachtes Lernverfahren zur Zuordnung von neuen Kunden zu den zuvor gebildeten Clustern mit einer Klassifikation. Agard und Kusiak (2004a, 2004b) gehen lediglich beispielhaft auf den Aufbau der Daten ein und geben keine Informationen über den Einsatz von Algorithmen und deren Evaluation. Zudem finden keine Anwendung und Validierung des Vorgehens statt.

Prädiktive datengetriebene Methode zur Entwicklung von Produktfamilien nach Ma und Kim

Ma und Kim (2016) stellen eine Methode zur prädiktiven datengetriebenen Entwicklung von Produktfamilien vor, welche das Clustering um einen marktgesteuerten Ansatz erweitert. Produktarchitekturkandidaten werden mithilfe des k-Means Clusteralgorithmus für unterschiedliche Werte von k generiert. Der marktorientierte Ansatz beinhaltet eine Gewinnmodellierung mit einer Zeitreihenanalyse mit exponentieller Glättung, um die optimale Position und Anzahl von Produktarchitekturen unter den Produktarchitekturkandidaten zu bestimmen (siehe Ma und Kim 2014; Ma et al. 2014). Dies soll die Erfassung von Trends in den Kundenpräferenzen und Unsicherheiten in der Vorhersagemodellierung ermöglichen. Anhand einer Fallstudie für die Entwicklung eines Elektromotors wird die Umsetzung des Ansatzes demonstriert. Dabei werden synthetische Daten mit acht kontinuierlichen Entwicklungsvariablen (z. B. Radius und Dicke des Motors) und vier resultierende Leistungsparameter (z. B. Drehmoment und Wirkungsgrad) in insgesamt 14 Millionen Instanzen analysiert. Ein auf den synthetischen Daten basierender Vergleich der Ergebnisse mit denen eines lediglich clusterbasierten Ansatzes zeigt, dass der Gewinn durch die Integration der marktbasierten Zeitreihenbetrachtung maximiert werden kann.

3.3 Resultierender Forschungsbedarf

Die vorgestellten Ansätze zur Analyse von Produktportfolios mittels Machine Learning werden analysiert und anhand der in Kapitel 3.1 definierten Kriterien verglichen. Die Ergebnisse sind in Tabelle 3-1 zu finden. Durch die bisherigen Ansätze aus der Literatur werden nicht alle Kriterien abgedeckt. Anhand der Gegenüberstellung mit den gestellten Anforderungen leiten sich die folgenden Forschungsbedarfe ab:

Komplexe Produktportfolios: Es besteht ein Forschungsbedarf bei der Analyse komplexer Produktportfolios. Die bisherigen Ansätze betrachten wenig komplexe Produktportfolios, synthetische oder reduzierte Datensätze. Kusiak et al. (2007) analysieren zwar die Daten eines Nutzfahrzeugherstellers, jedoch sind die Anzahl der Merkmale und Merkmalsausprägungen geringer als bei den in Kapitel 1.2 beschriebenen Industriebeispielen. Riesener et al. (2019b) analysieren ein komplexes Produktportfolio, jedoch auf der Ebene der strategischen Produktportfoliogestaltung.

Systematisierung der Wissensbedarfe: Forschungsbedarf existiert bei der Generierung eines Geschäftsverständnisses für die Analyse komplexer Produktportfolios. Verfahren des Machine Learning können auf unterschiedliche Weise eingesetzt werden, um Wissen für die Analyse komplexer Produktportfolios zu generieren. Damit Unternehmen Machine Learning im Produktportfolio- und Variantenmanagement einsetzen können, sind die Wissensbedarfe systematisch aufzubereiten und in den Prozess zur Analyse und Anpassung von Produktportfolios auf der Ebene der operativen Produktportfoliogestaltung einzuordnen. Darüber hinaus sind Kriterien aufzustellen, wie diese ausgewählt und priorisiert werden können.

Datenbasierte Beschreibung komplexer Produktportfolios: Ein Forschungsbedarf herrscht ebenfalls bei der Schaffung eines Datenverständnisses. Von den bisherigen Ansätzen aus der Literatur gehen lediglich Moon et al. (2010) auf ein spezifisches Datenmodell eines Produktportfolios ein. Für die Implementierung verschiedener Anwendungsfälle zur Analyse komplexer Produktportfolios ist es erforderlich, die Daten zur Beschreibung der Entscheidungen über komplexe Produktportfolios zu systematisieren. Dies beinhaltet die Ermittlung und Beschreibung der Daten sowie deren Generierung und Speicherung.

Vorbereitung von Produktportfoliodaten: Forschungsbedarf besteht bei der Unterstützung der Vorbereitung der Produktportfoliodaten. In Abhängigkeit des Analyseverfahrens und der Daten mit ihren spezifischen Charakteristiken sind unterschiedliche Schritte für die Datenvorbereitung erforderlich. Die bisherigen Ansätze aus der Literatur gehen lediglich teilweise auf die Datenvorbereitung in Abhängigkeit der von ihnen verwendeten Daten und Machine Learning Verfahren ein (siehe Riesener et al. 2019b; Tucker und Kim 2009; Yu et al. 2017). Eine Unterstützung bei der Auswahl und Anwendung in Abhängigkeit der verwendeten Produktportfoliodaten und Verfahren des überwachten und unüberwachten Lernen wird nicht bereitgestellt.

Tabelle 3-1: Gegenüberstellung der bisherigen Ansätze mit den gestellten Anforderungen

		Objektorientierte Kriterien					**Zielorientierte Kriterien**				
		Komplexe Produktportfolios	Verfügbare Daten	Operative Produktportfoliogestaltung	Verschiedene Machine Learning Verfahren	Unterschiedliche Produktportfolios	Systematisierung der Wissensbedarfe	Beschreibung von Produktportfolios	Vorbereitung von Produktportfoliodaten	Auswahl und Evaluation von Algorithmen	Einsatz des Wissens zur Analyse
Überwachte Lernverfahren	Tucker und Kim a)	○	○	●	○	◐	○	○	◐	○	◐
	Tucker und Kim b)	○	○	●	○	◐	○	○	○	○	◐
	Boyarkin et al.	○	●	●	○	◐	○	○	○	◐	●
	Riesener et al.	◐	◐	○	○	●	◐	○	◐	○	◐
Unüberwachte Lernverfahren	Romanowski und Nagi	○	●	●	◐	●	○	○	○	○	◐
	Kusiak et al.	◐	●	●	○	◐	○	○	○	○	◐
	Song und Kusiak	○	●	●	◐	◐	○	○	○	○	◐
	Moon et al.	○	●	◐	◐	●	○	◐	○	◐	●
	Tucker et al.	○	●	●	○	●	○	○	○	○	◐
	Chan et al.	○	○	●	○	●	○	○	○	○	◐
	Neis	○	●	●	○	●	○	○	○	○	◐
	Wang	○	●	●	○	●	○	○	○	◐	◐
	Yu et al.	○	●	●	○	●	○	○	◐	○	◐
Kombinierte Lernverfahren	Agard und Kusiak	○	●	●	◐	●	○	○	○	○	●
	Ma und Kim	◐	●	●	◐	●	○	○	○	○	●

Legende: ○ = nicht erfüllt; ◐ = teilweise erfüllt; ● = erfüllt

Auswahl und Evaluation von Algorithmen: Ein Forschungsbedarf existiert hinsichtlich der Unterstützung bei der Auswahl der Algorithmen und der Evaluation deren Güte. Wie in Kapitel 2.2.5 beschrieben, besitzen die einzelnen Algorithmen unterschiedliche Eigenschaften und daher in Abhängigkeit der Aufgabenstellung eine unterschiedliche Eignung. Es müssen mehrere geeignete Algorithmen bereitgestellt werden. Zudem bedarf es in Abhängigkeit des Machine Learning Verfahrens geeignete Evaluationskriterien. Lediglich Moon et al. (2010) setzen ein Kriterium für die Bewertung von Clustern und Wang (2019) für Assoziationsregeln ein.

Einsatz des Wissens zur Analyse: Es besteht ein Forschungsbedarf bezüglich des Einsatzes der erzeugten Machine Learning Modelle. In keinem der bisherigen Ansätze

wird auf die unterschiedlichen Möglichkeiten eingegangen, wie mit den erzeugten Modellen in Abhängigkeit des eingesetzten Machine Learning Verfahrens im Sinne der Ausführungen aus Kapitel 2.2.3 Wissen generiert und für die Analyse komplexer Produktportfolios eingesetzt werden kann.

4 Forschungsvorgehen

In diesem Kapitel wird das Forschungsvorgehen, welches zur Erstellung dieser Arbeit verfolgt wird, vorgestellt. Hierfür wird zuerst auf die zugrundeliegende Forschungsmethodik, die Design Research Methodology (DRM), eingegangen. Anschließend wird erläutert, wie diese angewendet wird und welche Artefakte dabei erzeugt werden. Des Weiteren werden die zugrundeliegenden Forschungsmethoden, welche zur Erstellung der Artefakte eingesetzt werden, beleuchtet.

4.1 Design Research Methodology (DRM)

Für die Durchführung des Promotionsvorhabens wird die Design Research Methodology (DRM) von Blessing und Chakrabarti (2009) herangezogen. Die DRM ist eine Methodik zur Durchführung von Forschungstätigkeiten im Bereich der Produktentwicklung (engl. Design Research). Die Forschung in der Produktentwicklung beinhaltet zum einen die Formulierung und Validierung von Modellen und Theorien über die Produktentwicklung mit all seinen Facetten (z. B. Mensch, Produkt, Organisation, …) und zum anderen die Entwicklung und Validierung von Unterstützungsmaßnahmen auf der Grundlage dieser Modelle und Theorien, um die Produktentwicklungspraxis und ihre Ergebnisse zu verbessern (Blessing und Chakrabarti 2009). Das Ergebnis sind Artefakte, welche eine erste Lösung für ein Problem bieten oder bestehende Lösungen verbessern (Hevner und Chatterjee 2010). Diese Artefakte können zum Beispiel Modelle, Methoden oder Frameworks sein (Peffers et al. 2012).

Die vier Phasen der DRM mit den zugrundeliegenden Methoden und Ergebnissen sind in Abbildung 4-1 dargestellt. Zuerst findet auf Basis der Literatur die **Klärung des Forschungsziels** statt, um ein Verständnis für die Ausgangslage zu erarbeiten sowie die gewünschte Situation zu beschreiben. Auf Basis dessen werden das Forschungsziel und die Forschungsfragen der Arbeit sowie Kriterien zur Messung des Erfolgs aufgestellt. In der **deskriptiven Studie I** wird ein tieferes Verständnis hinsichtlich der Ausgangssituation erarbeitet. Dafür ist eine fokussierte und zielorientierte Literaturrecherche erforderlich. Falls in der Literatur nicht ausreichend Belege zu finden sind, um die zentralen Faktoren zu untermauern, kann eine Befragung oder Beobachtung von Entwicklern erforderlich sein, um diese Faktoren zu adressieren. Auf Basis des tiefen Verständnisses werden die zuvor definierten Kriterien weiter geschärft. In der **präskriptiven Studie** wird das erlangte Wissen über die bestehende Situation genutzt, um die anfängliche Beschreibung der gewünschten Situation zu korrigieren. Auf Basis dessen wird mit der systematischen Entwicklung eines Artefakts zur Einstellung der gewünschten Situation begonnen. Dafür wird das gewonnene Verständnis und die ausgearbeitete Beschreibung der gewünschten Situation sowie die Erfahrung bei der Entwicklung von Gestaltungshilfen genutzt. In der **deskriptiven Studie II** wird untersucht, ob die entwickelte Gestaltungshilfe die angestrebte Wirkung besitzt und die

gewünschte Situation dadurch erreicht wird. Dafür sind zwei empirische Studien durchzuführen, um ein Verständnis für den tatsächlichen Nutzen der Unterstützung zu gewinnen. Die erste Studie dient dazu, die Anwendbarkeit des Artefakts zu validieren. Die zweite Studie beinhaltet die Erfolgsvalidierung, welche die Nützlichkeit der Gestaltungshilfe anhand der zuvor entwickelten Kriterien bewertet.

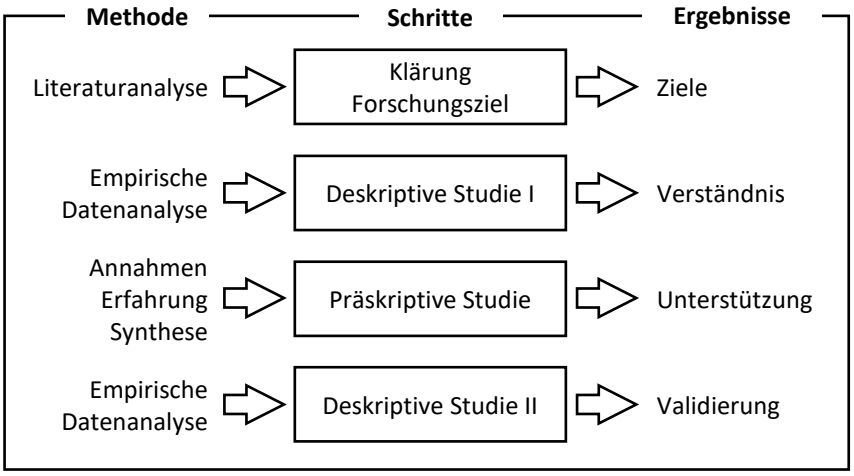


Abbildung 4-1: Design Research Methodology (DRM) in Anlehnung an Blessing und Chakrabarti (2009)

In der DRM lassen sich sieben Forschungstypen unterscheiden, bei denen die einzelnen Phasen je nach Forschungsfragen und der zur Verfügung stehenden Zeit und Ressourcen unterschiedlich durchlaufen werden. Eine Phase kann entweder review-basiert, initial oder umfassend behandelt werden. Eine **review-basierte Studie** beruht auf der Analyse der Literatur. Eine **umfassende Studie** beinhaltet sowohl eine Literaturbetrachtung als auch eine Studie, bei der die Ergebnisse vom Forscher selbst erarbeitet werden, d. h. der Forscher führt eine empirische Studie durch, entwickelt oder bewertet Gestaltungsansätze. Eine **initiale Studie** bildet den Abschluss eines Projekts und umfasst die ersten Schritte einer bestimmten Phase, um die Konsequenzen der Ergebnisse aufzuzeigen und die Ergebnisse für die Verwendung durch andere vorzubereiten.

4.2 Vorgehen und Methodeneinsatz

Das Forschungsvorgehen des Promotionsvorhabens beruht auf dem **Forschungstyp 5** der DRM „Entwicklung der Unterstützung auf der Grundlage einer umfassenden Studie der bestehenden Situation“. Das Ziel ist es, eine Unterstützung für die systematische Generierung von Wissen für die Analyse komplexer Produktportfolios in Form eines Frameworks zu entwickeln. Da jedoch der Kenntnisstand über die bestehende Situation in der Literatur gering ist, umfasst die Forschung sowohl die Entwicklung des Verständnisses im Rahmen einer umfassenden deskriptiven Studie I als auch, darauf

aufbauend, die Entwicklung der Unterstützung in einer umfassenden präskriptiven Studie. Darauf folgt eine initiale deskriptive Studie II zur ersten Validierung des Frameworks.

Für die **Klärung des Forschungsziels** und der Forschungsfragen wird zunächst ein grundlegendes Verständnis über die Ausgangssituation erarbeitet und anschließend das Ziel der Arbeit abgeleitet. Die Informationsgrundlage stützt zum einen auf eine Analyse ausgewählter Referenzen im Bereich Produktportfolio- und Variantenmanagement und zum anderen auf ersten Gesprächen und Beobachtungen bei einem Industriepartner aus der Nutzfahrzeugbranche.

In der **deskriptiven Studie I** wird ein tieferes Verständnis hinsichtlich der festgelegten Forschungsziele erarbeitet. Es werden die Grundlagen komplexer Produktportfolios und des Machine Learning aufbereitet, welche den Ausgangspunkt für die Entwicklung des Frameworks bilden. Es werden mithilfe einer Literaturrecherche und einer Interviewstudie bei einem Industrieunternehmen die aktuelle Situation im Produktportfolio- und Variantenmanagement erarbeitet und Ansatzpunkte in Form von Anwendungsfällen ermittelt.

Für die Durchführung der *Literaturrecherche* werden die sechs Schritte nach Machi und McEvoy (2016) befolgt. Zuerst wird eine Matrix mit den relevanten Begriffen formuliert und in einen Suchstring übersetzt (siehe Anhang A1.1). Die Suche in der Datenbank Scopus ergibt 643 Artikel. Es werden alle Artikel ausgewählt, die ein Machine Learning Verfahren zur Unterstützung des Produktportfolio- und Variantenmanagements anwenden. Insgesamt handelte es sich um 20 Artikel. Durch die Analyse ihrer Referenzen und Zitate werden weitere 13 Ansätze identifiziert. Die insgesamt 33 Artikel beschreiben 41 Anwendungen von Machine Learning im Produktportfolio- und Variantenmanagement (siehe Anhang A1.2). Die Anwendungsszenarien werden im Hinblick auf die Tätigkeiten des Produktportfolio- und Variantenmanagements, die Machine Learning Verfahren und Algorithmen sowie die Datenanforderungen näher beschrieben. Die detaillierten Ergebnisse können Mehlstäubl et al. (2021b) entnommen werden.

Experteninterviews bei einem Industriepartner werden durchgeführt, um die Relevanz des Themas, die Herausforderungen und Potenziale sowie Wissensbedarfe aus der Praxis zu identifizieren. Dabei werden acht Experten im Produktportfolio- und Variantenmanagement befragt. Eine Übersicht über das Profil der Teilnehmer kann Anhang A2.1 entnommen werden. Es werden semistrukturierte Interviews durchgeführt, bei denen Fragen vorab definiert werden. Die Reihenfolge ist flexibel und zusätzliche Fragen können während des Interviews hinzugefügt werden. Die Fragen können Anhang A2.2 entnommen werden. Die Interviews werden anschließend mit einer qualitativen Inhaltsanalyse nach Mayring (2010) kodiert und ausgewertet. Die Ergebnisse sind in Mehlstäubl et al. (2021a) im Detail aufbereitet.

In der **präskriptiven Studie** wird das Framework zur Analyse komplexer Produktportfolios mittels Machine Learning entwickelt. Das Framework besteht aus drei Bausteinen und berücksichtigt die einzelnen Phasen des CRISP-DM Prozessmodells (siehe Kapitel 2.2.4), welches die industrielle Anwendung von Machine Learning Verfahren fokussiert. In *Baustein I* werden zunächst die Wissensbedarfe ermittelt und den einzelnen Phasen des Prozesses zur Analyse und Anpassung komplexer Produktportfolios (siehe Kapitel 2.1.3) zugeordnet. Als Eingangsgrößen dienen die Ergebnisse der Interviewstudie aus Mehlstäubl et al. (2021b) sowie der Literaturrecherche zu Anwendungsfällen von Machine Learning im Produktportfolio- und Variantenmanagement aus Mehlstäubl et al. (2021a). Diese werden in einem ersten Schritt gruppiert und zu Wissensbedarfen abstrahiert. Im zweiten Schritt werden die Wissensbedarfe zu den einzelnen Phasen des Entscheidungsprozesses zur Analyse und Anpassung von Produktportfolios zugeordnet. Durch diesen Baustein wird die erste Forschungsfrage *„Welches Wissen kann mittels Machine Learning für die Analyse komplexer Produktportfolios generiert werden?*“ beantwortet.

In *Baustein II* findet eine datenbasierte Beschreibung komplexer Produktportfolios statt. Die Datenbedarfe werden durch die Analyse der zuvor identifizierten Anwendungsfälle und Wissensbedarfe definiert und deren Zusammenhänge auf Basis der Literatur und den Analysen bei einem Industriepartner beschrieben. Durch diesen Baustein wird die zweite Forschungsfrage *„Welche Daten sind für die Generierung von Wissen zur Analyse komplexer Produktportfolios notwendig?“* beantwortet.

In *Baustein III* wird die dritte Forschungsfrage *„Wie kann mittels Machine Learning Verfahren Wissen für die Analyse komplexer Produktportfolios generiert und eingesetzt werden?“* fokussiert. Es werden zuerst die Schritte der Datenvorbereitung und anschließend die Anwendung der unterschiedlichen Machine Learning Verfahren beschrieben. Es wird eine Unterstützung bei der Auswahl und Evaluation der Algorithmen sowie bei der systematischen Aufbereitung und Einsatz der Ergebnisse zur Analyse komplexer Produktportfolios bereitgestellt.

In der **deskriptiven Studie II** wird das entwickelte Framework validiert. Die Methode zur Validierung von Forschungsarbeiten in der Produktentwicklung ist abhängig von der Art des erzeugten Artefakts. Für die initiale Validierung eines Frameworks eignet sich eine Fallstudie (Peffers et al. 2012). Es wird eine *Einzelfallstudie* bei einem Industrieunternehmen aus der Nutzfahrzeugbranche durchgeführt. Aufgrund der globalen Wettbewerbssituation und der Vielzahl an Transportaufgaben und Anwendungsszenarien verfügen die Nutzfahrzeughersteller über ein besonders breites und tiefes Produktportfolio (Kreimeyer et al. 2013a) und sind daher für eine Einzelfallstudie besonders gut geeignet. Für die Durchführung der Fallstudie wird das Vorgehen nach Yin (2014) herangezogen. Die initiale Validierung des Frameworks im Rahmen des DRM-Typ 5 soll die prinzipielle Anwendbarkeit und Nützlichkeit der Entwicklungsunterstützung nachweisen und beinhaltet daher eine Anwendungs- und Erfolgsvalidierung. In der Anwendungsvalidierung wird das Framework auf reale Produktportfoliodaten des

Industriepartners angewendet. In der Erfolgsvalidierung wird die Nützlichkeit der dabei generierten Ergebnisse sowie der wissenschaftliche Mehrwert des Frameworks bewertet. In Abbildung 4-2 ist eine Übersicht der Inhalte der einzelnen Phasen der DRM in diesem Forschungsvorhaben dargestellt.

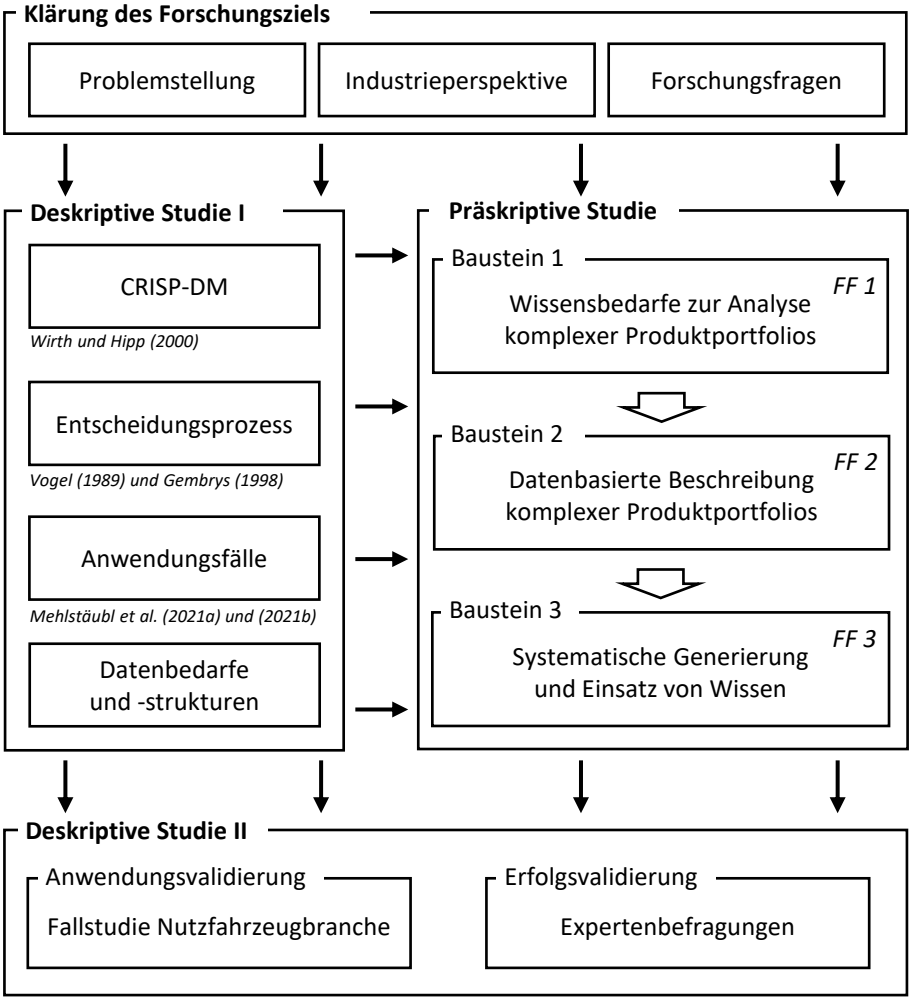


Abbildung 4-2: Übersicht der Entwicklung des Frameworks

4.3 Kriterien für die Entwicklung des Frameworks

Im Folgenden werden Kriterien für die Entwicklung und Validierung des Frameworks definiert. Dabei wird in Anlehnung an Jank (2021) zwischen inhaltlichen und formalen Kriterien unterschieden. Die inhaltlichen Kriterien beziehen sich auf die funktionalen Eigenschaften, welche das Framework erfüllen muss. Daneben sind Forschungsartefakte anhand von Kriterien zu bewerten, die aus den Anforderungen des jeweiligen Kontexts resultieren (Peffers et al. 2012). Hierfür werden formale Kriterien verwendet, welche die Anforderungen aus Sicht der Forschung in der Produktentwicklung an das zu entwickelnde Framework stellen.

Inhaltliche Kriterien

Die inhaltlichen Kriterien beziehen sich auf funktionale Eigenschaften, welche das zu entwickelnde Framework erfüllen muss, und werden im Sinne der Anforderungserhebung in der Produktentwicklung ermittelt. In der Produktentwicklung stellen Anforderungen technische Entwicklungsziele beziehungsweise gewünschte Produkteigenschaften dar (Lindemann 2009). Nach Pahl et al. (2005) dient die Klärung und Präzisierung der Aufgabenstellung der Informationsbeschaffung und hat als Ergebnis die Anforderungen an das Produkt. Die inhaltlichen Kriterien sind abgeleitet aus den Ergebnissen der Klärung des Forschungsziels, der Präzisierung der Aufgabenstellung und dem resultierenden Forschungsbedarf in der deskriptiven Studie I, welche in Kapitel 1 bis 3 beschrieben werden.

Aus der Motivation und Zielsetzung in Kapitel 1 ergibt sich die Forderung nach der Analyse **komplexer Produktportfolios**, welche eine Vielzahl an Merkmalsausprägungen und Komponentenvarianten besitzen (siehe Kapitel 2.1.1). Die Analyse und Anpassung von Produktportfolios erfolgt im Rahmen eines Entscheidungsprozesses. Innerhalb dessen ist die Generierung eines Geschäftsverständnisses durch die **Systematisierung der Wissensbedarfe** zur Analyse komplexer Produktportfolios erforderlich. Aktuelle Herausforderungen beim Einsatz von Machine Learning in Industrieunternehmen sind ein fehlendes Wissen über die Verfahren sowie eine Intransparenz in den Datenstrukturen. Daher ist eine **datenbasierte Beschreibung komplexer Produktportfolios** zur Bereitstellung eines Datenverständnisses erforderlich. Zur Analyse komplexer Produktportfolios gibt es eine Vielzahl möglicher Anwendungsfälle, weshalb ein flexibles Framework notwendig ist. Dieses muss eine systematische Unterstützung bei der **Vorbereitung von Produktportfoliodaten** sowie bei der **Auswahl und Evaluation von Algorithmen** bereitstellen, sodass unterschiedliche Machine Learning Verfahren in verschiedenen Unternehmenskontexten mit den individuellen Produktdatenmodellen genutzt werden können. Die erzeugten Machine Learning Modelle können auf unterschiedliche Weise eingesetzt werden, um Wissen über komplexe Produktportfolios zu generieren. Daher ist der **Einsatz der Machine Learning Modelle** in Abhängigkeit der einzelnen Verfahren zu beschreiben.

Formale Kriterien

Neben den inhaltlichen Anforderungen sind für den Nachweis des wissenschaftlichen Mehrwerts im Bereich der Forschung in der Produktentwicklung formale Kriterien erforderlich (Aier und Fischer 2011). Für die Validierung von Forschungsarbeiten in der Produktentwicklung sind in der Literatur unterschiedliche Ansätze zu finden. March und Smith (1995) definieren für die Validierung von Entwicklungsmethoden die Kriterien Benutzerfreundlichkeit, Effizienz, Allgemeinheit und Operationalität. Hevner et al. (2004) nennen Funktionalität, Vollständigkeit, Konsistenz, Genauigkeit, Leistung, Zuverlässigkeit, Benutzerfreundlichkeit, Eignung für die Organisation und andere relevante Qualitätsmerkmale als Kriterien für die Validierung von IT-Artefakten. Aier und Fischer (2011) greifen die Kriterien auf und erweitern diese für eine allgemeingültige

Bewertung von Artefakten in der Produktentwicklungsforschung. Die Kriterien sind Nützlichkeit, Konsistenz, Zweck und Umfang, Einfachheit sowie Fruchtbarkeit für weiterer Forschung und werden für die Entwicklung und Bewertung dieser Arbeit herangezogen.

Ein Artefakt in der Produktentwicklungsforschung erfüllt das Kriterium der **Nützlichkeit**, wenn es den zugrundeliegenden Zweck erfüllt und der Zweck selbst nützlich ist (Aier und Fischer 2011). Bei der **Konsistenz** des Frameworks kann zwischen der internen und externen Konsistenz unterschieden werden. Die interne Konsistenz bezieht sich auf die Konsistenz zwischen den Elementen des Artefakts, wie zum Beispiel die Verwendung einer einheitlichen Terminologie. Die externe Konsistenz untersucht die Übereinstimmung zwischen den Theorien des erzeugten Artefakts mit dem allgemeinen Wissen innerhalb der Produktentwicklung. Sie stellt die Verbindung zwischen dem Artefakt und der Wissensbasis her (Hevner et al. 2004). Der **Umfang** und Zweck eines Artefakts in der Produktentwicklungsforschung sollten möglichst weit gefasst sein. Je breiter der Anwendungsbereich und der Zweck eines Artefakts ist, desto größer ist der Mehrwert aus Sicht der Forschung (Aier und Fischer 2011). Ein breiter Anwendungsbereich und Zweck besagen, dass das Artefakt durch geringfügige Anpassungen für verschiedene Zwecke und Bereiche erweitert werden kann, ohne dass sein Nutzen abnimmt. **Einfachheit** bedeutet, dass das erzeugte Artefakt leicht für die Nutzer zu verstehen und anwendbar ist. Die Verständlichkeit und Einfachheit eines Artefakts in der Produktentwicklung können dessen Akzeptanz bei den Nutzern erhöhen. Ein neues Artefakt gehört zur Wissensbasis der Disziplin und muss die Grundlage für weitere Forschung bilden. Aus diesem Grund ist ein Kriterium für die Bewertung von Artefakten die **Fruchtbarkeit** der neuen Forschungsergebnisse.

4.4 Schlussfolgerungen zum Forschungsvorgehen

Die vorliegende Arbeit ist Teil der Forschung in der Produktentwicklung und erzeugt als Artefakt ein Framework zur Analyse komplexer Produktportfolios mittels Machine Learning. Es wird versucht das bekannte Problem der Analyse komplexer Produktportfolios mit Machine Learning als neue Lösung zu verbessern. Daneben bietet Machine Learning das Potenzial neue Lösungen für neue Probleme im Produktportfolio- und Variantenmanagement bereitzustellen. Dadurch wird nach Gregor und Hevner (2013) ein Wissensbeitrag für die Forschung in der Produktentwicklung geleistet (siehe Abbildung 4-3). Als Forschungsmethodik wird die DRM-Typ 5 eingesetzt und innerhalb der einzelnen Phasen der DRM werden unterschiedliche Forschungsmethoden verwendet. Die einzelnen Phasen der DRM können direkt den Kapiteln dieser Arbeit zugeordnet werden. In Kapitel 1 findet die Klärung des Forschungsziels statt. Anschließend wird in Kapitel 2 und 3 im Rahmen der deskriptiven Studie I ein tiefergehendes Verständnis erarbeitet, welches als Ausgangspunkt für die Entwicklung des Frameworks in der präskriptiven Studie dient. Das Framework wird im folgenden Kapitel 5

vorgestellt. In Kapitel 6 findet die Validierung mit einer Fallstudie bei einem Nutzfahrzeughersteller und einer Expertenbefragung im Rahmen der deskriptiven Studie II statt.

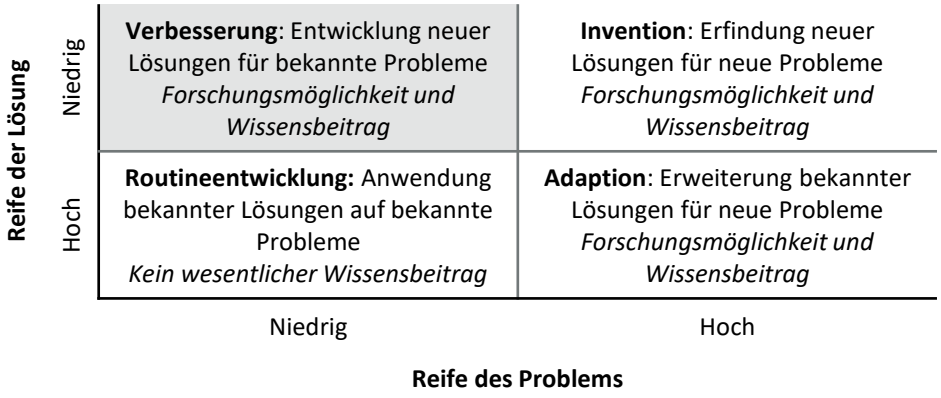


Abbildung 4-3: Einordnung des wissenschaftlichen Mehrwerts nach Gregor und Hevner (2013)

5 Framework zur Analyse komplexer Produktportfolios

Im folgenden Kapitel wird das Framework zur Analyse komplexer Produktportfolios mittels Machine Learning eingeführt. Hierfür wird zuerst eine Übersicht über das Framework gegeben, bevor anschließend die einzelnen Bausteine im Detail erläutert werden. Abschließend wird das Vorgehen zur Anwendung des Frameworks im industriellen Kontext beschrieben.

5.1 Übersicht über das Framework

Das Framework ist in Abbildung 5-1 dargestellt und besteht aus drei Bausteinen. Im ersten Baustein wird ein Geschäftsverständnis für die operative Produktportfoliogestaltung bereitgestellt. Dafür werden **Wissensbedarfe zur Analyse komplexer Produktportfolios** (Kapitel 5.2) zu den einzelnen Phasen des Entscheidungsprozesses (siehe Kapitel 2.1.3) zugeordnet sowie Bewertungskriterien für deren Auswahl bereitgestellt. Im zweiten Baustein findet eine **datenbasierte Beschreibung komplexer Produktportfolios** (Kapitel 5.3) statt, wodurch ein Datenverständnis generiert wird. Dabei wird auf das Produktdatenmodell, die Vertriebsdaten und die Nutzungsdaten sowie deren Datencharakteristiken eingegangen.

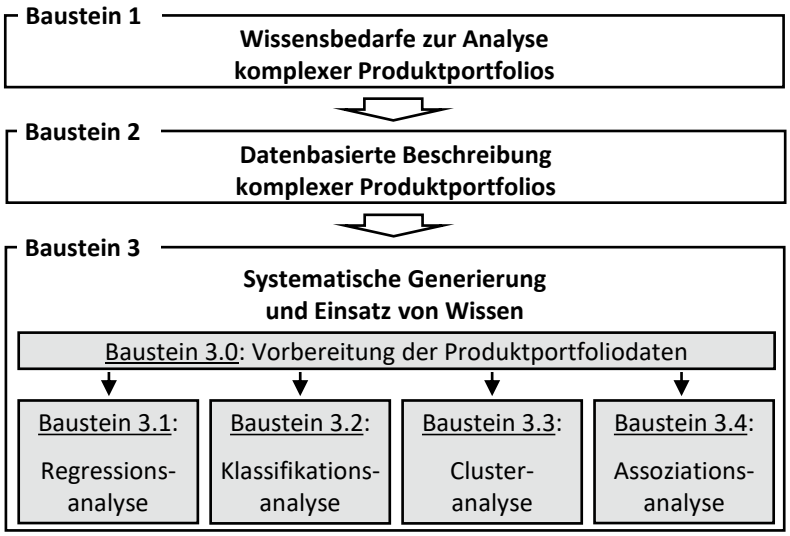


Abbildung 5-1: Framework zur Analyse komplexer Produktportfolios

Baustein 3 beinhaltet die **systematische Generierung von Wissen zur Analyse komplexer Produktportfolios** (Kapitel 5.4) und ist unterteilt in weitere vier Unterbausteine. In Baustein 3.0 findet zuerst die Datenvorbereitung statt. Es werden die Schritte zur Datenbereinigung und -transformation in Abhängigkeit der verwendeten Produktportfoliodaten und eingesetzten Analyseverfahren erläutert. Baustein 3.1 bis

3.4 sind die Analysebausteine und beinhalten die Anwendung der Regressions-, Klassifikations-, Cluster- und Assoziationsanalyse. In den einzelnen Bausteinen wird eine Unterstützung für die Auswahl und Evaluation der Algorithmen geliefert sowie der systematische Einsatz der Ergebnisse zur Analyse komplexer Produktportfolios auf der Ebene der operativen Produktportfoliogestaltung beschrieben.

5.2 Baustein 1: Wissensbedarfe zur Analyse komplexer Produktportfolios

Für den systematischen Einsatz von Machine Learning zur Analyse komplexer Produktportfolios wird im Folgenden durch die Beschreibung der Wissensbedarfe innerhalb der einzelnen Phasen des Entscheidungsprozesses ein Geschäftsverständnis bereitgestellt (siehe Mehlstäubl et al. 2023a). Die ermittelten Wissensbedarfe der einzelnen Phasen des Entscheidungsprozesses, die zugrundeliegenden Anwendungsfälle und erforderlichen Machine Learning Verfahren sind in Tabelle 5-1 zu finden. Die Wissensbedarfe sind unternehmensabhängig auszuwählen und zu konkretisieren. In diesem Baustein wird kein Anspruch erhoben, alle Wissensbedarfe zur Analyse komplexer Produktportfolios umfassend zu definieren. Stattdessen werden einerseits Einblicke in die aktuellen Herausforderungen bei der Analyse komplexer Produktportfolios und andererseits ein Ausgangspunkt für Unternehmen, Machine Learning in die operative Produktportfolioentwicklung zu integrieren, gegeben.

5.2.1 Informationssuche

In der Phase der Informationssuche werden Informationen über Produktvarianten und einzelne Merkmalsausprägungen gesammelt und vernetzt. Mit Machine Learning können *marktspezifische Eigenschaften von Produktvarianten (W1)* prognostiziert werden. Mit Hilfe von Regressions- und Klassifikationsanalysen können Korrelationen in den verkauften und produzierten Produktvarianten identifiziert und darauf basierend Vorhersagen getroffen werden. So können beispielsweise der Preis und die Zahlungsbereitschaft für neue Produktvarianten vorhergesagt werden (siehe Boyarkin et al. 2019). Dieser Ansatz lässt sich auf andere marktspezifische Faktoren, wie zum Beispiel das Vertriebsland, ausweiten. Neben den marktspezifischen Faktoren können ebenfalls *technische Eigenschaften der Produktvarianten (W2)* (z. B. Gewichte und Längen) bestimmt werden. Drittens kann Wissen über die *zeitliche Entwicklung marktspezifischer Größen (W3)* mit Hilfe einer Zeitreihenanalyse generiert werden (siehe Tucker und Kim 2011b). Ein weiterer interessanter Faktor sind Gewinn- und Kostenentwicklungen der einzelnen Merkmalsausprägungen, welche aber zunächst genau zugeordnet werden müssen.

In den ersten beiden Anwendungen von Machine Learning wird Wissen auf der Ebene der Produktvarianten generiert. Daneben ist Wissen über einzelne Merkmals-

ausprägungen von Bedeutung. Hierfür können die Machine Learning Modelle, wie in Kapitel 2.2.3 beschrieben, analysiert werden. Die explizite Zuordnung von marktspezifischen oder technischen Werten zu einzelnen Merkmalsausprägungen ist in der Regel nicht möglich, da die Zusammenhänge zwischen den Merkmalsausprägungen und den Produkteigenschaften nicht linear sind. Die Werte ergeben sich aus der Kombination mit anderen Merkmalsausprägungen. Allerdings können die für die technischen und marktspezifischen Produkteigenschaften wesentlichen Merkmale und Merkmalsausprägungen mit der Feature Importance, welche diesen einen Wichtigkeitswert zuweist, ermittelt oder die Entscheidungen durch die Analyse der Modellparameter nachvollzogen werden (z. B. Entscheidungsbaum). Außerdem können mit den Modellen die Auswirkungen von Änderungen in den Merkmalsausprägungen der Produktvarianten simuliert und dadurch die Eigenschaften optimiert werden.

Tabelle 5-1: Einordnung der Wissensbedarfe und Anwendungsfälle in den Entscheidungsprozess nach Mehlstäubl et al. (2023a)

Phasen	Wissensbedarfe	Anwendungsfälle	Verfahren
Informationssuche	W1: Marktspezifische Eigenschaften der Produktvarianten	Vorhersage von Preis und Zahlungsbereitschaft	Regression, Klassifikation
		Abschätzen der Kaufentscheidungen	Klassifikation
		Bestimmung der wichtigsten Produktmerkmale für den Preis	Klassifikation
		Ermittlung der wesentlichen Produktmerkmale der Produktfamilien	Assoziation
	W2: Technische Eigenschaften der Produktvarianten	Vorhersage technischer Produkteigenschaften	Regression, Klassifikation
	W3: Zeitliche Entwicklung marktspezifischer Größen	Prognose der Gewinnentwicklungen	Regression
		Vorhersage der Nachfragetrends	Regression
Formulierung von Alternativen	W4: Nicht profitable Merkmalsausprägungen	Entwicklung von Vorschlägen für Portfolioentscheidungen	Klassifikation
	W5: Ähnlichkeit von Produktvarianten	Segmentierung der Märkte	Clustering
		Standardisierung der Produkte	Clustering
		Identifikation ähnlicher Komponenten und Baugruppen	Clustering
	W6: Korrelation zwischen Merkmalsausprägungen	Identifikation von Korrelationen zwischen Produktmerkmalen	Assoziation
		Identifikation von Korrelationen zwischen Komponenten	Assoziation
		Identifikation von sequentiellen Korrelationen zwischen Produktmerkmalen	Assoziation
Prognose der Auswirkungen	W7: Präferenzen von Kunden und Kundensegmenten	Zuweisung von Kunden zu Marktsegmenten und Konfigurationen	Klassifikation
	W8: Auswirkungen der Produktportfolioänderungen	Simulation der Auswirkungen von Portfolioänderungen	Regression
		Ermittlung von Abhängigkeiten zwischen Portfolio- und Unternehmenskennzahlen	Regression

5.2.2 Formulierung von Alternativen

Die nächste Phase des Prozesses besteht darin, mögliche Alternativen zum aktuellen Produktportfolio zu definieren, mit denen eine Verbesserung erzielt wird. Viele Unternehmen haben heute mit dem historischen und unkontrollierten Wachstum ihres Produktportfolios zu kämpfen. Teil dieser Phase ist daher die Generierung von Wissen über *nicht profitable Merkmalsausprägungen (W4).* Mithilfe von Machine Learning lassen sich Merkmalsausprägungen auf der Grundlage früherer Entscheidungen klassifizieren und so Vorschläge für Portfolioentscheidungen generieren. Um die Vielfalt zu reduzieren, ist das Wissen über die *Ähnlichkeit von Produktvarianten (W5)* wichtig. Machine Learning kann eingesetzt werden, um Produktvarianten anhand der Merkmalsausprägungen oder Komponentenvarianten mittels Clustering zu segmentieren (Zhang et al. 2007). Dies ermöglicht eine Standardisierung durch die Identifikation von Ähnlichkeiten und Gruppierung von Konfigurationen. Neben der Ähnlichkeit von Produkten und Komponenten bietet das Wissen über die *Korrelation zwischen Merkmalsausprägungen (W6)* einen Mehrwert für die Entscheidungsfindung im Produktportfolio- und Variantenmanagement. Durch das Wissen, welche Komponenten oder Merkmale zusammen auftreten, können Module gebildet oder die Kombinierbarkeit von Merkmalsausprägungen eingeschränkt werden (siehe Moon et al. 2010). Das sequentielle Pattern Mining kann auch eingesetzt werden, um Korrelationen zwischen Produktmerkmalen im Zeitverlauf zu identifizieren und das Kundenverhalten abzuschätzen (Yu und Zhang 2014).

5.2.3 Prognose

Inhalt dieser Phase ist es, die Auswirkungen von Veränderungen im Produktportfolio auf interne und externe Ziele vorherzusagen und zu steuern. Zu diesem Zweck kann mit Machine Learning Wissen über die *Präferenzen von Kunden und Kundensegmenten (W7)* ermittelt werden. Kunden können durch eine Klassifikation auf Basis ihrer Eigenschaften zu Konfigurationen oder Marktsegmenten zugeordnet werden und so das Kundenverhalten bei Produktabkündigungen simuliert werden. Ein weiterer wichtiger Punkt ist das Wissen über die *Auswirkungen der Produktportfolioänderungen (W8)* (z. B. Absatzzahlen und Umsätze). Durch den Einsatz von Machine Learning können die Auswirkungen von Portfolioveränderungen prognostiziert werden. So kann z. B. untersucht werden, wie sich das Entfernen eines Merkmals auf die Anzahl der verkauften Einheiten anderer Merkmale auswirkt. Darüber hinaus können die Abhängigkeiten zwischen den Portfolioparametern und verschiedenen Unternehmensindikatoren erfasst und die Auswirkungen von Änderungen simuliert werden (siehe Riesener et al. 2019b).

5.2.4 Kriterien zur Auswahl der Wissensbedarfe

Unternehmen haben unterschiedliche Voraussetzungen und Herausforderungen im Produktportfolio- und Variantenmanagement, weshalb es notwendig ist, die Wissensbedarfe zu bewerten und für die anschließende Konkretisierung auszuwählen. Im Folgenden werden für die Auswahl der Wissensbedarfe und Anwendungsfälle Bewertungskriterien hergeleitet. In der Literatur sind eine Vielzahl an Arbeiten zu finden, welche Kriterien für die Bewertung von neuen Produktideen bereitstellen (siehe z.B. Herrmann et al. 2018; Messerle 2016; Tzokas et al. 2004). Diese Kriterien können jedoch nicht ohne Adaption für die Analyse von Anwendungsfällen für Machine Learning im Produktportfolio- und Variantenmanagement herangezogen werden, da sie zum einen nicht den spezifischen Arbeitskontext der Produktentwicklung berücksichtigen und zum anderen nicht auf die Voraussetzungen für Datenanalysen eingehen. Disselkamp (2015) schlägt vor, die zwei Hauptkategorien Machbarkeit und Attraktivität für die Bewertung zu verwenden. Wilberg (2020) beschreibt in diesem Zusammenhang für die Auswahl von Anwendungsfällen im Rahmen der Implementierung einer Nutzungsdatenstrategie für die Machbarkeit die folgenden Kriterien: Änderungen am aktuellen Produkt, Datenschutz, Nutzerakzeptanz, Implementierungszeit, erforderliche Datenanalysefähigkeiten der Mitarbeiter, IT-Infrastruktur, Personalressourcen, Datensicherheit, Datenspeicherung und Datenübertragung. Für die Attraktivität führt er die Kriterien Innovationsgrad, Marktattraktivität, Wettbewerbsvorteil, Kosteneinsparungspotenzial, Prozessverbesserung, Qualitätsverbesserung, Nutzen für die Anwender oder Kunden, Auswirkungen auf die Kundenzufriedenheit und Geschäftspotenzial auf.

Der VDMA Bayern (2020) definiert als Kriterien für die Bewertung der **Machbarkeit** von Machine Learning Anwendungsfällen die Datenverfügbarkeit, die Datenmenge und die Datenqualität. Unter der *Datenverfügbarkeit* wird der Zugriff auf die Daten, welche zur Implementierung eines Anwendungsfalls erforderlich sind, verstanden. Die *Datenmenge* berücksichtigt die Repräsentativität der verfügbaren Daten, sodass sich verallgemeinerbare Gesetzmäßigkeiten ableiten lassen. Ein weiteres wichtiges Kriterium beim Umgang mit Daten ist der *Datenschutz* und die *Datensicherheit*. In einem industriellen Kontext enthalten Daten weniger persönliche Inhalte. Dennoch werden z. B. in Kundendaten oder Daten aus der Nutzungsphase, Informationen über persönliche Muster und Vorlieben gesammelt und gespeichert. Unternehmen sind verpflichtet, Maßnahmen zu ergreifen, um diese personenbezogenen Daten vor Missbrauch zu schützen (Wilberg et al. 2017). Andererseits enthalten Daten in Unternehmen Geheimnisse und Wissen, die geschützt werden müssen, um ihre Wettbewerbsfähigkeit zu erhalten. Ein weiteres Kriterium, um die Machbarkeit zu bewerten, ist die *Nutzungsbereitschaft*. Die Einführung neuer Methoden oder Werkzeuge in Unternehmen kann auf Widerstand bei den Nutzern stoßen (Birkhöfer et al. 2002). Machine Learning ist für die Entwickler eine neue Technologie, mit der Entscheidungen auf der Grundlage von Daten getroffen werden. Zudem sind beim Einsatz von Machine Learning Modellen die Zusammenhänge zwischen den Eingangsmerkmalen und der Zielvariable nur schwer nachvollziehbar (Eckert et al. 2020). Darüber hinaus ist eine enge *Zusammenarbeit mit*

Nutzern für die Konkretisierung, Umsetzung und Bewertung des Anwendungsfalls erforderlich (Mehlstäubl et al. 2021c).

Die **Attraktivität** von Machine Learning Anwendungsfällen wird zunächst durch den *Nutzen*, welcher sich durch die Implementierung ergibt, charakterisiert (Disselkamp 2015). Durch die Umsetzung eines Anwendungsfalls muss ein relativer Vorteil gegenüber den aktuellen Ansätzen erzeugt werden. Der relative Vorteil beschreibt das Ausmaß, in dem eine Methode als besser wahrgenommen wird als diejenige, die von ihr abgelöst wird (Rogers et al. 2014). Nach Géron (2017) bringt Machine Learning vor allem einen Nutzen, bei

- Problemen, für die bestehende Lösungen viel Handarbeit oder eine Vielzahl an Regeln erfordern,
- komplexen Problemen, für die es mit den herkömmlichen Ansätzen überhaupt keine Lösung gibt,
- schwankenden Umgebungsbedingungen, an die sich ein Machine Learning Modell schnell anpassen kann,
- großen Datenmengen, welche durch den Menschen nicht analysiert werden können.

Dem Nutzen ist der *Aufwand* für die Umsetzung gegenüberzustellen und zu bewerten (Kerka et al. 2011). Oft werden Ideen in frühen Phasen verworfen, weil ihre Umsetzung zu zeit- und kostenaufwendig ist (Disselkamp 2015). Des Weiteren spielt das *Integrationsrisiko*, welches die Wahrscheinlichkeit für Fehler und deren Auswirkungen innerhalb des Anwendungsfalls beschreibt, eine Rolle (Messerle 2016). Die Bewertungskriterien sind in Abhängigkeit des Unternehmens individuell auszuwählen und einzusetzen.

5.3 Baustein 2: Datenbasierte Beschreibung komplexer Produktportfolios

In diesem Kapitel wird auf die Daten eingegangen, welche zur Analyse komplexer Produktportfolios für die operative Produktportfoliogestaltung und zur Generierung des in Baustein 1 beschriebenen Wissens mit Machine Learning erforderlich sind. Ausgehend von den Wissensbedarfen wurden mit Experten eines Industriepartners die wichtigsten Daten ermittelt. Diese sind das Produktdatenmodell, die Vertriebsdaten und die Nutzungsdaten (siehe Abbildung 5-2). Im Sinne des Design for X (DfX) beinhalten alle Daten, die im Lebenszyklus einer Produktvariante generiert werden (z. B. Produktionsdaten), Informationen für das Produktportfolio- und Variantenmanagement (siehe Kiritsis et al. 2003). Der Ausgangspunkt ist das Produktdatenmodell. Auf Basis dessen werden im Konfigurationsprozess Produktvarianten konfiguriert und anschließend produziert. Die Vertriebsdaten enthalten Informationen über die zugrundeliegenden Konfigurationen der Produktvarianten, Produkteigenschaften und Kunden.

Während der Produktnutzung werden das Verhalten der Produktvarianten und deren Interaktion mit dem Nutzer als Nutzungsdaten gespeichert. Dabei handelt es sich um die relevantesten Daten zur Analyse komplexer Produktportfolios.

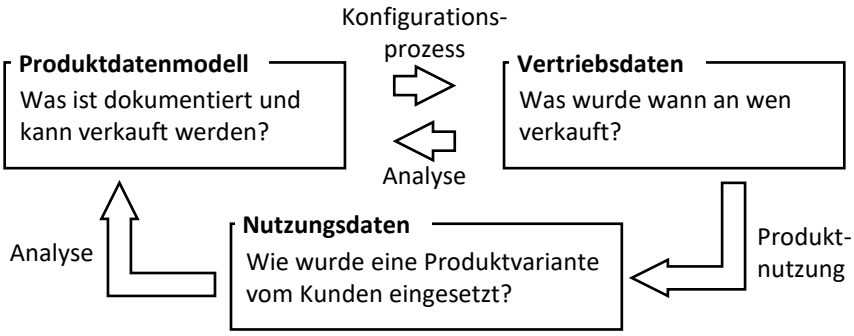


Abbildung 5-2: Daten zur Analyse komplexer Produktportfolios

Für die Vorbereitung von Daten sind deren Charakteristiken zu berücksichtigen. Die typischen Datencharakteristiken sind das Volumen, die Geschwindigkeit, die Vielfalt und die Richtigkeit der Daten (L'Heureux et al. 2017). Das Volumen beschreibt den Umfang der verfügbaren Datensätze (Sivarajah et al. 2017). Dabei kann ein großes Volumen herausfordernd zu verarbeiten sein, aber auch ein zu kleines Volumen kann dafür sorgen, dass eine Analyse keine verwertbaren Ergebnisse liefert (Mehlstäubl et al. 2022b). Die Vielfalt bezieht sich auf die strukturelle Heterogenität in den Daten (Che et al. 2013). Die Geschwindigkeit bezeichnet die Rate, mit der die Daten erzeugt und verarbeitet werden müssen (Gandomi und Haider 2015). Da für die Analyse komplexer Produktportfolios die Daten nicht nach der Erzeugung in Echtzeit ausgewertet werden müssen, wird diese Charakteristik in der folgenden Arbeit nicht weiter berücksichtigt. Die Richtigkeit zeigt die Unzuverlässigkeit eines Datensatzes auf (Kantardzic 2011). Damit sind die Verzerrungen, Unsicherheiten, Eindrücke, Unwahrheiten sowie fehlenden Werte in den Daten gemeint (Zicari 2014).

5.3.1 Produktdatenmodell

Die Produktarchitektur komplexer Produktportfolios ist in einem generischen Produktdatenmodell dokumentiert, welches für alle Produktvarianten gültig ist (siehe Braun et al. 2017). Ein Vergleich verschiedener Produktdatenmodelle wird von Tidstam und Malmqvist (2010) sowie Braun (2021) durchgeführt. Tidstam und Malmqvist (2010) gehen auf Produktdatenmodelle aus Industrie und Wissenschaft ein (siehe z. B. Generic Bill of Materials nach McKay et al. (1996), Feature models nach Bühne et al. (2004) oder K- & V-Matrix nach Bongulielmi et al. (2001)). Die Modelle besitzen zwar keine einheitliche Struktur, jedoch herrschen signifikante Ähnlichkeiten, welche Tidstam und Malmqvist (2010) in ihrem Produktkonfigurationsframework aufgreifen. Das Framework besteht aus einer Merkmals- und Produktstruktur sowie einem Kombinatorik-

und Teileauswahlregelwerk. In Abbildung 5-3 ist der Aufbau des Frameworks anhand eines Beispiels dargestellt. In diesem umfasst die Merkmalsstruktur eines Bürostuhls die Merkmale „Farbe“, „Fahrbarkeit“ und „Drehbarkeit“, welche jeweils zwei oder mehr Merkmalsausprägungen annehmen können. Kombinatorikregeln zur Einschränkung der Kombinierbarkeit von Merkmalsausprägungen werden mit einer Matrix formuliert. Zum Beispiel besagt die Regel 2, dass fahrbare Bürostühle nicht „nicht drehbar“ sein dürfen.

Teileauswahlregeln	Rot	Blau	Fahrbar	Nicht fahrbar	Nicht drehbar	Drehbar		Rot	Blau	Fahrbar	Nicht fahrbar	Nicht drehbar	Drehbar	**Kombinatorik-regeln**
Bürostuhl	x	x							x		x			Regel 1
> Unterrahmen			x	x						x			x	Regel 2
>> Ständer					x	x								
--- Roter, fahrbarer, drehbarer Ständer	x		x		x									
--- Roter, nicht fahrbarer, drehbarer Ständer	x			x	x									
Eltern:														**Eltern:**
Bürostuhl				■	x			■						Rot
Unterrahmen					■	x			■					Blau
Ständer						■				■				Fahrbar
Merkmalsstruktur											■			Nicht Fahrbar
												■		Drehbar
													■	Nicht drehbar
	Bürostuhl			Unterrahmen		Ständer	**Kinder**	Rot	Blau	Fahrbar	Nicht fahrbar	Nicht drehbar	Drehbar	**Produkt-struktur**

Abbildung 5-3: Produktkonfigurationsframework nach Tidstam und Malmqvist (2010)

Daneben beinhaltet das Framework Komponenten (z. B. Ständer) und deren Komponentenvarianten (z. B. roten, fahrbaren und drehbaren Ständer) sowie Teileauswahlregeln zwischen Merkmalsausprägungen und Komponentenvarianten. Die Teileauswahlregeln werden ebenfalls mit einer Matrix definiert. In Abbildung 5-3 wird zum Beispiel das „rote, fahrbare und drehbare Gestell“ gezogen, wenn die Merkmalsausprägungen „rot“, „fahrbar“ und „drehbar“ vom Kunden ausgewählt werden.

Kreimeyer et al. (2016) greifen das Framework in ihrem integrierten Produktinformationsmodell auf, konkretisieren und entwickeln es weiter. Kernbestandteile sind die Merkmalsstruktur mit den Merkmalen und Merkmalsausprägungen, die Produktgliederung mit den Komponenten und Komponentenvarianten sowie der Lösungsraum mit der CAD-Struktur (Kreimeyer et al. 2016). Jedes Lösungselement einschließlich seiner Unterstrukturen (Stücklistenelemente und deren Teilenummern) stellt eine

konkrete technische Lösung für eine Komponentenvariante dar (siehe Ziethen 2007). Die Positionsstruktur enthält die Achssystem- und Transformationsinformationen, um die Lösungselemente entsprechend geometrisch korrekt auszurichten und zu positionieren. In Abbildung 5-4 ist die Merkmalsstruktur, Produktgliederung und Lösungsraum sowie Kombinatorik- und Teileauswahlregelwerk des Produktinformationsmodells beispielhaft mit Elementen des Produktportfolios eines Nutzfahrzeugs dargestellt.

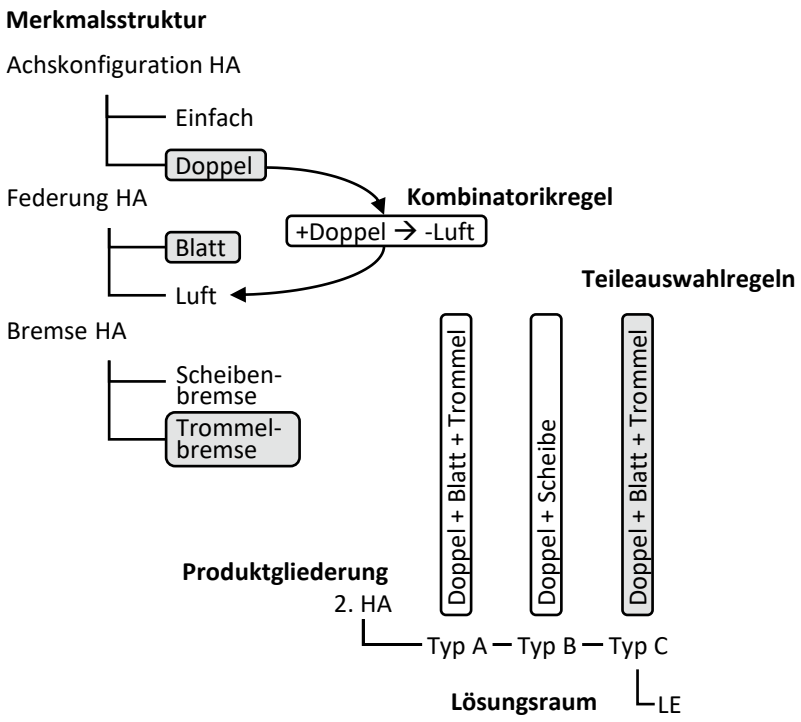


Abbildung 5-4: Produktinformationsmodell in Anlehnung an Kreimeyer et al. (2016)

Um die Varianz zu steuern und zu beherrschen, definieren sie ebenfalls Kombinatorik- und Teileauswahlregeln sowie Lageauswahlregeln. Die Lageauswahlregeln legen die korrekte Positionierung jedes Lösungselements fest und funktionieren auf die gleiche Weise wie die Teileauswahlregeln. Im Vergleich zu Tidstam und Malmqvist (2010) werden hier Boolesche Regeln verwendet, welche die einzelnen Merkmalsausprägungen in Beziehungen zueinander setzen („Merkmalsausprägung A erfordert Merkmalsausprägung B“ oder „Merkmalsausprägung A verbietet Merkmalsausprägung B“). Das Produktdatenmodell wird im Produktdatenmanagementsystem (PDM-System) gespeichert. Die Merkmalstruktur und die Produktgliederung sind im stetigen Wandel, da Elemente hinzugefügt, reduziert oder zusammengefasst werden (Kreimeyer et al. 2016). Im Folgenden wird als Produktdatenmodell das Produktinformationsmodell nach Kreimeyer et al. (2016) herangezogen.

5.3.2 Vertriebsdaten

Die Vertriebsdaten beschreiben, welche Produktvarianten an wen verkauft wurden. Sie können in den Vertriebsdatenkopf und -rumpf unterteilt werden. Der **Vertriebsdatenrumpf** enthält Informationen über die exakte Konfiguration der Produktvarianten und wird durch den Konfigurationsprozess erzeugt, bei dem Komponentenvarianten so ausgewählt und angeordnet werden, dass die Kundenspezifikation in Form der Merkmalsausprägungen erfüllt wird (Sabin und Weigel 1998). Der Konfigurationsprozess findet heutzutage unter Einsatz eines Produktkonfigurators statt (Rapp 1999), welcher bei der Konfiguration von Produktvarianten komplexer Produktportfolios durch den Vertrieb in Abstimmung mit dem Kunden bedient wird (Schuh und Riesener 2018). Der Produktkonfigurator interagiert mit dem Produktdatenmodell, um die Merkmalsausprägungen anzubieten, deren Konfigurierbarkeit einzuschränken sowie darauf aufbauend die Komponentenvarianten mit zugehörigen Lösungselementen auszuwählen und ein Auftrag zu erstellen (siehe Abbildung 5-5). Vom Kunden wird dabei für jedes Merkmal eine Merkmalsausprägung ausgewählt und der Konfigurator zieht aus jeder Komponente eine entsprechende Komponentenvariante (Braun et al. 2018).

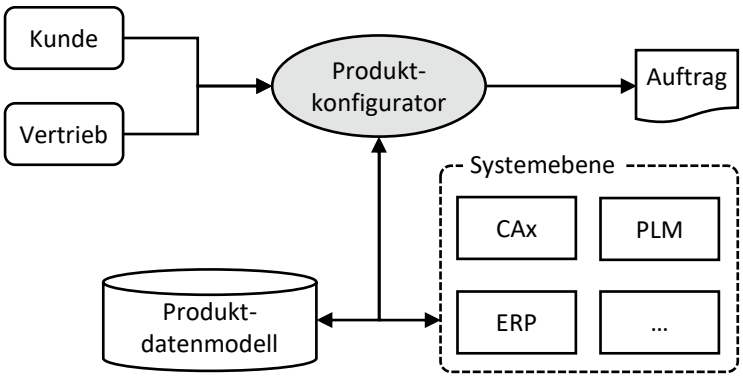


Abbildung 5-5: Produktkonfiguration und Produktkonfigurator in Anlehnung an Schuh und Riesener (2018)

Der Auftrag enthält die ausgewählten Merkmalsausprägungen sowie die resultierenden Komponentenvarianten und Sachnummern. Er wird in der Regel nach der Auftragserstellung im Konfigurator ins Auftragseingangssystem übergeben und in einem Ablagesystem (z. B. einem Information Warehouse) gespeichert und kann von dort exportiert werden (siehe Mehlstäubl et al. 2022a). Im Auftragseingangssystem besteht jede Produktvariante aus einer eindeutigen Identifikationsnummer (ID) sowie einer Merkmals- und Produktstruktur (siehe Tabelle 5-2). Die Produktstruktur in Unternehmen ist aufgrund neuer Anforderungen im ständigen Wandel. Neue Merkmale und Merkmalsausprägungen kommen hinzu und bestehende werden aus dem Produktportfolio entfernt. Dies führt zu vielen fehlenden Werten im Vertriebsdatenrumpf. Zudem kann es vorkommen, dass in einzelnen Merkmalen bisher nur eine Ausprägung

verkauft wurde, so dass sie konstant sind. Darüber hinaus sind Vertriebsdaten komplexer Produktportfolios nicht ausgewogen, sondern besitzen aufgrund der unterschiedlichen Stückzahlen der Standardprodukte und exotischen Produktkonfigurationen ein starkes Ungleichgewicht.

Tabelle 5-2: Exemplarische Struktur des Vertriebsdatenrumpfs in Anlehnung an Mehlstäubl et al. (2022a)

	Merkmale und Merkmalsausprägungen					Komponenten und Komponentenvarianten			
ID	**Modell**	**Anwendung**	**Fahrer-haus**	**Federung**	**…**	**Kraftstoff-tank**	**Ersatzrad**	**Steuergerät**	**…**
1	XL	Bau	Groß	Blatt/Luft		Klein	Ohne	ASM	
2	S	Bau	Medium	Blatt/Blatt		Klein	Mit (hinten)	EDC	
3	M	Bau	Klein	Blatt/Luft		Klein	Mit (hinten)	EDC	
4	M	Getränke	Klein	Luft/Luft		Medium	Mit (rechts)	EDC	
5	L	Distribution	Klein	Blatt/Luft		Groß	Mit (rechts)	ASM	

Der **Vertriebsdatenkopf** beinhaltet Informationen über die Kunden (z. B. Name), die organisatorischen Abläufe (z. B. Bestelldatum) sowie die technischen (z. B. Gewicht) und nicht-technischen Eigenschaften (z. B. Kaufpreis) der Produktvarianten (siehe Tabelle 5-3). Die Produkteigenschaften der verkauften Konfigurationen werden nach Abschluss des Verkaufsprozesses oder des Produktionsprozesses dokumentiert. Die Informationen des Vertriebsdatenkopfs werden im Auftragsabwicklungsprozess zum Großteil manuell befüllt. Aus diesem Grund können sie neben fehlenden Werten auch fehlerhafte Werte enthalten.

Tabelle 5-3 Struktur des Vertriebsdatenkopfs mit synthetischen Daten

	Organisatorische Daten			Kundendaten		Produkteigenschaften		
ID	**Datum Eingang**	**Datum Lieferung**	**Montage-band**	**Kunde**	**Land**	**Gewicht**	**Kaufpreis**	**CO_2**
1	11.02.2021	28.02.2021	Y	Kunde_A	DE	5 480 kg	82 000	560 $\frac{g\,CO_2}{km}$
2	11.02.2021	28.02.2021	Y	Kunde_A	DE	5 480 kg	82 000	560 $\frac{g\,CO_2}{km}$
3	10.03.2021	29.04.2021	M	Kunde_B	FR	9 260 kg	109 000	610 $\frac{g\,CO_2}{km}$
4	17.03.2021	29.04.2021	M	Kunde_C	SI	9 940 kg	106 000	590 $\frac{g\,CO_2}{km}$
5	08.04.2021	15.06.2021	M	Kunde_D	NL	10 315 kg	130 000	740 $\frac{g\,CO_2}{km}$
6	21.09.2020	05.05.2021	Y	Kunde_E	NL	5 480 kg	90 000	570 $\frac{g\,CO_2}{km}$

5.3.3 Nutzungsdaten

Technische Produkte bestehen aufgrund der Digitalisierung nicht mehr ausschließlich aus physischen Teilen, sondern auch aus Sensoren, Mikroprozessoren und zusätzlichen Komponenten für die Konnektivität (Porter und Heppelmann 2014). Heutzutage erzeugen die Produkte kontinuierlich Daten und zeichnen diese in Form von Nutzungsdaten auf (siehe Panzner et al. 2022). Durch die Rückführung dieser Daten in die Produktentwicklung und insbesondere in die Produktplanung können Unternehmen ihre Entwicklungs- und Entscheidungsprozesse optimieren. Aus diesem Grund sind Produkte mit einer hohen Anzahl intelligenter Komponenten sowie langen Betriebszeiten für eine Analyse der Nutzungsdaten im Produktportfolio- und Variantenmanagement erforderlich (siehe Holler et al. 2016).

Die Daten können ständig oder bei bestimmten Ereignissen (z. B. Wartung oder Ausfällen) übertragen werden (siehe Kiritsis et al. 2003; Wilberg et al. 2017). Aus diesem Grund wird bei einem Industriepartner aus der Nutzfahrzeugbranche eine Unterscheidung hinsichtlich der Übertragung der Nutzungsdaten zwischen den Telematikdaten, welche ständig übertragen werden, und Trenddaten, welche bei bestimmten Ereignissen ausgelesen werden, vorgenommen. Kreutzer (2019) klassifiziert dagegen die Nutzungsdaten hinsichtlich der Erzeugung in Sensoren und Aktoren, Nutzerdaten und Systemdaten. Meyer et al. (2022) führen fünf Kategorien für die Daten ein, welche in der Nutzungsphase eines Produkts erzeugt werden. Diese sind die Nutzungsdaten, Nutzerverhaltensdaten, Servicedaten, Produktverhaltensdaten und Statusdaten. Die Nutzungsdaten beschreiben, wie ein Produkt von den Nutzern verwendet wird, die Nutzerverhaltensdaten erfassen das Verhalten der Nutzer bei der Verwendung des Produkts, die Servicedaten enthalten Informationen über Probleme und die Qualität des Produkts, die Produktverhaltensdaten gehen darauf ein, wie sich das Produkt während des Betriebs verhält und die Statusdaten beinhalten den Status und die Zustände des Produkts (Meyer et al. 2022).

Das Sammeln von Daten über den tatsächlichen Einsatz eines Produkts führt unweigerlich zu umfangreichen Nutzungsdaten, sowohl was das Volumen als auch die Vielfalt der Daten betrifft (Hou und Jiao 2020). Nutzungsdaten können fehlende Werte enthalten, da die Nutzer die Daten nicht übermitteln oder nicht auslesen lassen. Daneben können durch eine Nutzung außerhalb des Nutzungskontexts, Fehlverhalten oder defekte Sensorik fehlerhafte Werte in den Daten entstehen. Für den Einsatz von Nutzungsdaten im Produktportfolio- und Variantenmanagement müssen die Rohdaten in einem Betriebsdatenmanagementsystem aufbereitet und in statistische Aussagen überführt werden. Zum Beispiel können sie Auskunft darüber geben, wie lange ein Produkt sich in einem bestimmten Zustand befand oder wie viel Kraftstoff durchschnittlich verbraucht wurde. In Tabelle 5-4 ist ein exemplarischer Auszug aus aufbereiteten Nutzungsdaten von fünf Fahrzeugen dargestellt.

Tabelle 5-4: Struktur der Nutzungsdaten mit synthetischen Daten

ID	Jährliche Laufleistung [km/Jahr]	Durchschnitts-geschwindigkeit [km/h]	Mittlere Geschwindigkeit [km/h]	Tägliche Laufleistung [km/Tag]	Durchschnitts-verbrauch [l/100km]	Mittlere Gesamtmasse [t]
1	135 235,14	43,36	43,70	370,51	28,32	25,21
2	27 765,57	10,51	12,21	76,07	42,31	21,05
3	22 640,94	24,22	17,78	62,03	28,04	17,08
4	51 290,60	40,71	40,96	140,52	26,97	14,64
5	34 081,18	43,63	44,68	93,37	33,57	26,37

5.4 Baustein 3: Systematische Generierung und Einsatz von Wissen

Im Folgenden wird auf den Baustein zur systematischen Generierung von Wissen mit Machine Learning und dessen Einsatz zur Analyse komplexer Produktportfolios eingegangen, welcher sich wiederum aus fünf Unterbausteinen zusammensetzt. Zuerst wird auf die „Vorbereitung von Produktportfoliodaten“ (Kapitel 5.4.1) eingegangen. Anschließend werden die vier Analysebausteine „Regressionsanalyse“ (Kapitel 5.4.2), „Klassifikationsanalyse“ (Kapitel 5.4.3), „Clusteranalyse“ (Kapitel 5.4.4) und „Assoziationsanalyse“ (Kapitel 5.4.5) vorgestellt, welche jeweils aus der Modellierung, Evaluation und Einsatz in Abhängigkeit des betrachteten Verfahrens bestehen.

5.4.1 Baustein 3.0: Vorbereitung von Produktportfoliodaten

Um Algorithmen des Machine Learning anwenden zu können, müssen die Daten zuerst bereinigt und anschließend transformiert werden. Im Folgenden wird auf die erforderlichen Verfahren zur Bereinigung und Transformation von Produktportfoliodaten eingegangen. Dabei ist darauf zu achten, dass in Abhängigkeit der verwendeten Daten und Analyseverfahren unterschiedliche Schritte durchzuführen sind. Eine Unterstützung bei der systematischen Auswahl der Schritte bietet Tabelle 5-5.

Tabelle 5-5: Auswahl von Datenbereinigungs- und Transformationsverfahren

	Konstante Merkmale	Fehlende Werte	Encoding	Skalierung	Dimensions-reduktion
Vertriebsdaten	x	o	x	-	o
Nutzungsdaten	-	o	-	x	o
Regressionsanalyse	x	x	o	o	-
Klassifikationsanalyse	x	x	o	o	-
Clusteranalyse	x	x	o	o	x
Assoziationsanalyse	x	-	o	o	-

x = erforderlich, - nicht erforderlich, o = keine Abhängigkeit

5.4.1.1 Datenbereinigung

Die Datenbereitung beinhaltet im Produktportfolio- und Variantenmanagement in erster Linie den Umgang mit konstanten Eingangsmerkmalen sowie Strategien zur Handhabung fehlender Werte.

Konstante Eingangsmerkmale

Konstante Eingangsmerkmale bieten keine zusätzlichen Informationen hinsichtlich der Muster in den Eingangsmerkmalen sowie deren Beziehungen zu einer Zielvariable. Aus diesem Grund können konstante Eingangsmerkmale entfernt werden. In erster Linie kann es bei Vertriebsdaten vorkommen, dass Merkmale oder Komponenten lediglich eine Ausprägung in den Daten enthalten, da die anderen Ausprägungen bisher noch nicht verkauft wurden oder Merkmale einen organisatorischen Zweck, wie zum Beispiel eine Kennzeichnung der Produktgeneration, erfüllen.

Fehlende Werte

Reale Datensätze enthalten fehlende Werte, welche vor der Modellierung bereinigt werden müssen. Einige Machine Learning Algorithmen können zwar mit fehlenden Werten umgehen, jedoch haben diese auf die Güte der Modelle negative Auswirkungen. Die Betrachtung der Herkunft der fehlenden Werte ist hilfreich für deren Interpretation und die Bestimmung einer geeigneten Strategie (Ester und Sander 2000). Im Konfigurationsprozess wird, wie in Kapitel 5.3 beschrieben, für jedes Merkmal eine Merkmalsausprägung und für jede Komponente eine Komponentenvariante ausgewählt. Aus diesem Grund ergeben sich bei der Konfiguration keine fehlenden Werte. Fehlende Einträge resultieren vielmehr aus den ebenfalls in Kapitel 5.3 beschriebenen Anpassungen der Produktarchitektur und dadurch des Produktdatenmodells, bei denen Merkmale und Komponenten hinzugefügt oder entfernt werden. Alle Konfigurationen die vor bzw. nach dem Zeitpunkt der Änderung erzeugt wurden, erhalten in der betreffenden Spalte keinen Wert oder den Wert „Not a Number“ (NaN) (siehe Fischer und Hofer 2011). Bei Nutzungsdaten resultieren fehlende Werte zum einen aus den unterschiedlichen Ausstattungen der Produkte und zum anderen daraus, dass nicht alle Kunden ihre Daten übermitteln bzw. auslesen lassen.

Eine mögliche Bereinigungsstrategie für fehlende Werte ist die betroffene Instanz zu löschen. Da auf diese Weise jedoch viele nützliche Informationen verloren gehen, ist dies nur bedingt sinnvoll (Han et al. 2012). Falls Konfigurationen oder Merkmale einen hohen Anteil an fehlenden Einträgen aufweisen, kann ein solches Vorgehen dennoch zweckmäßig sein. Wenn nur ein kleiner Teil der Werte fehlt, sollten diese durch Konstanten ersetzt werden. Dazu können bei numerischen Eingangsmerkmalen der Median (Han et al. 2012) oder der Mittelwert (Grzymala-Busse und Grzymala-Busse 2009) und bei kategorischen Eingangsmerkmalen die häufigsten Einträge verwendet werden. Alle diese Verfahren zielen darauf ab, fehlende Werte durch den Wert mit der höchsten Wahrscheinlichkeit zu ersetzen. Eine weitere mögliche Strategie ist das Verfahren des Global Closest Fit nach Grzymala-Busse und Grzymala-Busse (2009). Für

eine Instanz mit einem fehlenden Wert wird ermittelt, mit welcher anderen Instanz die meisten Eingangsmerkmale gemeinsam existieren. Der fehlende Wert wird dann ersetzt durch das korrespondierende Eingangsmerkmal der ähnlichsten Instanz. Die Strategie ist anwendungsspezifisch danach auszuwählen, bei welchen Eingangsmerkmalen oder Instanzen sowie in welchem Umfang fehlende Werte in den konkret vorliegenden Zieldaten identifiziert werden.

5.4.1.2 Datentransformation

Die Datentransformation umfasst die Überführung der Daten in eine für den Algorithmus lesbare Form sowie die Vorverarbeitung der Daten, damit die Machine Learning Algorithmen leistungsfähiger sind. Im Folgenden wird auf die Kodierung und die Dimensionsreduktion im Kontext komplexer Produktportfolios eingegangen.

Kodierung

Machine Learning Algorithmen benötigen für die Verarbeitung einen numerischen Input. Hierfür müssen die kategorischen Variablen in numerische Werte umgewandelt werden. Dies geschieht im Rahmen einer Kodierung, wofür zwei unterschiedliche Verfahren herangezogen werden können: Ordinale Kodierung und One-hot Kodierung.

Eine **ordinale Kodierung** wird verwendet, wenn zwischen den einzelnen Ausprägungen der Kategorien in den Eingangsmerkmalen eine Rangfolge besteht (z. B. Model S, M, L, XL). Ordinale Kodierer weisen den unterschiedlichen Werten einer kategorialen Variable einen eindeutigen ganzzahligen Wert zu (z. B. Model S=1, M=2, L=3, XL=4) (siehe Abbildung 5-6) (Eye und Clogg 1996). Die Rangfolge richtet sich nach der Bedeutung der jeweiligen Variablen.

ID	Modell	Anwendung	Fahrerhaus	Federung
1	XL	Bau	Groß	Blatt/Luft (B/L)
2	S	Bau	Medium	Blatt/Blatt (B/B)
3	M	Bau	Klein	Blatt/Luft (B/L)
4	M	Getränke	Klein	Luft/Luft (L/L)
5	L	Distribution	Klein	Blatt/Luft (B/L)

Ordinale Kodierung

ID	Modell	Anwendung	Fahrerhaus	Federung
1	4	1	3	2
2	1	1	2	1
3	2	1	1	2
4	2	2	1	3
5	3	3	1	2

Abbildung 5-6: Ordinale Kodierung

Eine **One-hot Kodierung** wandelt die Daten in eine Matrix um, in der jede Spalte einem möglichen Wert eines Eingangsmerkmals entspricht (Pentreath 2015). Durch ein One-hot Kodierung wird eine diskrete nominale Variable x, welche die Werte x_1, x_2, ..., x_n annehmen kann, in einen binären Vektor v umgewandelt. Soll eine bestimmte Ausprägung x_i von x codiert werden, erhält jedes Element von v den Wert null außer das i-te Element, das den Wert eins annimmt (Hancock und Khoshgoftaar 2020). Auf diese Weise entsteht für jede Ausprägung a_s eines Eingangsmerkmals m_r eine Spalte, wodurch sich die Größe des Datensatzes erhöht. Im Ergebnis entsteht eine Matrix, die ausschließlich binäre Werte enthält. Abbildung 5-7 zeigt anhand eines Beispiels, wie sich die ursprüngliche Form der Zieldaten durch diese Art der Kodierung verändert.

ID	Modell	Anwendung	Fahrerhaus	Federung
1	XL	Bau	Groß	Blatt/Luft (B/L)
2	S	Bau	Medium	Blatt/Blatt (B/B)
3	M	Bau	Klein	Blatt/Luft (B/L)
4	M	Getränke	Klein	Luft/Luft (L/L)
5	L	Distribution	Klein	Blatt/Luft (B/L)

One-hot Kodierung

ID	S	M	L	XL	Bau	Getränke	Distribution	Klein	Medium	Groß	B/B	B/L	L/L
1	0	0	0	1	1	0	0	0	0	1	0	1	0
2	1	0	0	0	1	0	0	0	1	0	1	0	0
3	0	1	0	0	1	0	0	1	0	0	0	1	0
4	0	1	0	0	0	1	0	1	0	0	0	0	1
5	0	0	1	0	0	0	1	1	0	0	0	1	0

Abbildung 5-7: One-hot Kodierung in Anlehnung an Mehlstäubl et al. (2022a)

Skalierung

Unterschiedliche Skalen in den Eingangsmerkmalen können Auswirkungen auf die Güte von Machine Learning Algorithmen haben (Géron 2017). Dies ist im Produktportfolio- und Variantenmanagement in erster Linie bei Nutzungsdaten sowie den Produkteigenschaften im Vertriebsdatenkopf der Fall. Um alle Attribute auf dieselbe Skala zu bringen, gibt es zwei gängige Verfahren: Normalisierung und Standardisierung.

Bei der **Normalisierung** werden die Werte von X durch Subtraktion des Minimalwerts X_{min} und Division durch den Maximalwert minus den Minimalwert neu skaliert, sodass sie einen Wert zwischen null und eins annehmen (Burkov 2019):

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \qquad \textit{Formel 5-1}$$

Eine Normalisierung ist sinnvoll, wenn die Ober- und Untergrenze der Daten bekannt ist und keine Ausreißer existieren bzw. diese entfernt werden.

Die **Standardisierung** subtrahiert hingegen zuerst den Mittelwert μ von den Werten von X und dividiert anschließend durch die Varianz σ (Ali et al. 2014):

$$X' = \frac{X - \mu}{\sigma} \qquad \textit{Formel 5-2}$$

Dadurch werden die Werte zwar nicht auf einen bestimmten Bereich begrenzt, jedoch ist der Einfluss der Ausreißer im Vergleich zur Normalisierung geringer.

Dimensionsreduktion

Die Komplexität der Berechnungen steigt bei lernenden Algorithmen mit der Anzahl an Eingangsvariablen des Datensatzes an (Alpaydin 2020). Die Dimensionsreduktion dient dazu, die originalen Daten in einen Eingaberaum niedrigerer Dimensionalität zu projizieren, ohne den Informationsgehalt zu reduzieren (Han et al. 2012). Für einen kodierten Datensatz kann die Multiple Correspondence Analysis (MCA) angewendet werden. Sie stellt eine Erweiterung der Correspondence Analysis dar und ermöglicht es die Relationen von mehreren kategorischen Eingangsmerkmalen zu untersuchen (Abdi und Valentin 2007). Neben den reduzierten Dimensionen wird durch die Ermittlung der Singulär- und Eigenwerte ermöglicht, die erklärte Varianz abzuleiten und dadurch die Übereinstimmung mit dem originalen Datensatz transparent zu machen.

5.4.2 Baustein 3.1: Regressionsanalyse

Regressionsanalysen können genutzt werden, um marktspezifische und technische Eigenschaften von Produktvarianten, die zeitliche Entwicklung marktspezifischer Größen sowie die Auswirkungen von Produktportfolioänderungen zu ermitteln (siehe Kapitel 5.2). Im Folgenden wird eine Unterstützung bei der Auswahl der Regressionsalgorithmen sowie deren Evaluation und Einsatz gegeben (siehe Mehlstäubl et al. 2022a).

5.4.2.1 Regression

Für die Modellbildung werden zuerst geeignete Regressionsalgorithmen in Abhängigkeit verschiedener Faktoren ausgewählt und mit den Trainingsdaten trainiert sowie mit den Testdaten getestet.

In dieser Arbeit werden die Algorithmen lineare Regression, Support Vector Machine, K-Nearest Neighbor, Entscheidungsbaum, Random Forest und neuronales Netz betrachtet, welche in Kapitel 2.2.5.1 erläutert wurden. Für die Vorhersagegenauigkeit der Regressionsmodelle und somit für die Auswahl der Algorithmen spielen mehrere Faktoren eine Rolle. Einer der wichtigsten Faktoren ist die Anzahl der Datenpunkte. Einige Algorithmen können Korrelationen mit nur wenigen Datenpunkten erkennen, wohingegen andere für große Datensätze geeignet sind (siehe Singh et al. 2016). Die Anzahl der Eingangsmerkmale und der Grad der Linearität zwischen den Eingangsmerkmalen und der Zielvariablen haben ebenfalls einen großen Einfluss auf das

Verhalten der Regressionsalgorithmen (siehe Ray 2019). Bei vielen Anwendungen des Machine Learning ist die Trainingszeit ein kritischer Faktor (Riesener et al. 2020). Bei der Optimierung komplexer Produktportfolios spielt die Trainingszeit jedoch eine untergeordnete Rolle, da die Modelle nicht vor oder parallel zur Anwendung neu trainiert werden müssen. Dagegen spielt in einigen Anwendungsfällen im Produktportfolio- und Variantenmanagement die Vorhersagezeit eine Rolle (Mehlstäubl et al. 2022a). Zum Beispiel müssen Produkteigenschaften teilweise schon während des Produktkonfigurationsprozesses in Echtzeit vorhergesagt werden, ohne dass die resultierende Produktstruktur bekannt ist. Ein weiterer relevanter Faktor im industriellen Kontext ist die Transparenz (Riesener et al. 2020). Für die Akzeptanz der Modelle ist es wichtig, deren Verhalten zu verstehen. Dies ermöglicht zudem die Analyse der Modelle und die Gewinnung zusätzlicher Erkenntnisse über Muster in den Daten (siehe Kapitel 2.2.3). Ein Überblick über die Faktoren und zugehörigen Eigenschaften der Regressionsalgorithmen ist in Tabelle 5-6 dargestellt.

Die Daten werden je nach Datenmenge in einem Verhältnis zwischen 70 % / 30 % und 95 % / 5 % in Trainings- und Testdaten aufgeteilt (Burkov 2019). Dabei ist wichtig, dass die Anzahl an Testdaten nicht zu gering ist. Das bedeutet, dass bei wenigen Datenpunkten der prozentuale Anteil der Testdaten höher festgelegt werden sollte. Bei der Implementierung sind mehrere vielversprechende Algorithmen zu trainieren und deren Genauigkeit mit statistischen Kriterien auf der Grundlage der Vorhersagen mit den Testdaten zu vergleichen und erst anschließend das präziseste Modell auszuwählen.

Tabelle 5-6: Faktoren für die Algorithmenauswahl bei einer Regressionsanalyse nach Mehlstäubl et al. (2022a)

	Daten-punkte	Nicht-linearität	Feature-anzahl	Vorhersage-zeit	Trainings-zeit	Transparenz
Lineare Regression	sehr niedrig	sehr niedrig	niedrig	sehr niedrig	gering	sehr hoch
Support Vector Machine	niedrig	niedrig	niedrig	sehr hoch	sehr hoch	mittel
K-Nearest Neighbors	mittel	mittel	mittel	mittel	sehr gering	mittel
Entscheidungsbaum	mittel	mittel	hoch	sehr niedrig	gering	hoch
Random Forest	hoch	hoch	hoch	sehr niedrig	mittel	hoch
Neuronales Netz	sehr hoch	sehr hoch	sehr hoch	niedrig	hoch	sehr niedrig

5.4.2.2 Evaluation

Evaluationskriterien messen bei einer Regressionsanalyse die Güte der Modelle auf Basis der Vorhersagen für die Testdaten. Typische Kriterien, mit denen Regressionsalgorithmen trainiert werden, sind der mittlere quadratische Fehler (MSE, engl. mean squared error) und das Bestimmtheitsmaß (R^2, engl. R-Squared). Der MSE beschreibt den Durchschnitt der Quadrate der Fehler, d. h. die durchschnittliche quadratische Differenz zwischen den vorhergesagten Werten $\hat{y}$ und den tatsächlichen Werten y und berechnet sich wie folgt (Botchkarev 2018):

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \qquad \textit{Formel 5-3}$$

R^2 ist ein Maß für die Anpassungsgüte des Modells und gibt den Anteil der dadurch erklärten Varianz von der gesamten Varianz in den Daten an. R^2 kann Werte zwischen 0 und 1 annehmen und wird wie folgt berechnet (Fahrmeir et al. 2007):

$$R^2 = \frac{Erklärte\ Varianz}{Gesamte\ Varianz} = \frac{\sum_{i=1}^{n}(\hat{y}_i - \bar{y})}{\sum_{i=1}^{n}(y_i - \bar{y})} \qquad \textit{Formel 5-4}$$

Der mittlere absolute Fehler (MAE, engl. mean absolute error) und der mittlere absolute prozentuale Fehler (MAPE, engl. mean absolute percentage error) werden für die klare Kommunikation der Ergebnisse empfohlen. Der MAE beschreibt die durchschnittliche Abweichung zwischen dem vorhergesagten Wert und dem Wert in den Daten (siehe Botchkarev 2018). Er wird wie folgt berechnet:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \qquad \textit{Formel 5-5}$$

Der MAPE beschreibt die durchschnittliche prozentuale Abweichung und berechnet sich wie folgt (Aman et al. 2014):

$$MAPE = \frac{100\ \%}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \qquad \textit{Formel 5-6}$$

Die Kennzahlen machen Aussagen über die durchschnittlichen Genauigkeiten der Modelle auf Basis der Vorhersagen für die Testdaten. Über Ausreißer werden keine Erkenntnisse gewonnen, weshalb diese separat analysiert werden müssen. Hier empfiehlt sich eine graphische und tabellarische Gegenüberstellung der vorhergesagten Werte mit den tatsächlichen Werten in den Daten sowie deren Differenz. In Abbildung 5-8 sind die Vorhersageergebnisse beispielhaft graphisch dargestellt. Dabei werden den tatsächlichen Werten y zum einen die vorhergesagten Werte $\hat{y}$ und zum anderen die Differenz zwischen den vorhergesagten und den tatsächlichen Werten $y_i - \hat{y}_i$ gegenübergestellt. Für die detaillierte Analyse der Ausreißer sind die exakten tatsächlichen und vorhergesagten Werte sowie deren Differenz gegenüberzustellen (Tabelle 5-7). Durch die ID können im nächsten Schritt die konkreten Produktkonfigurationen ermittelt und der Fehler analysiert werden.

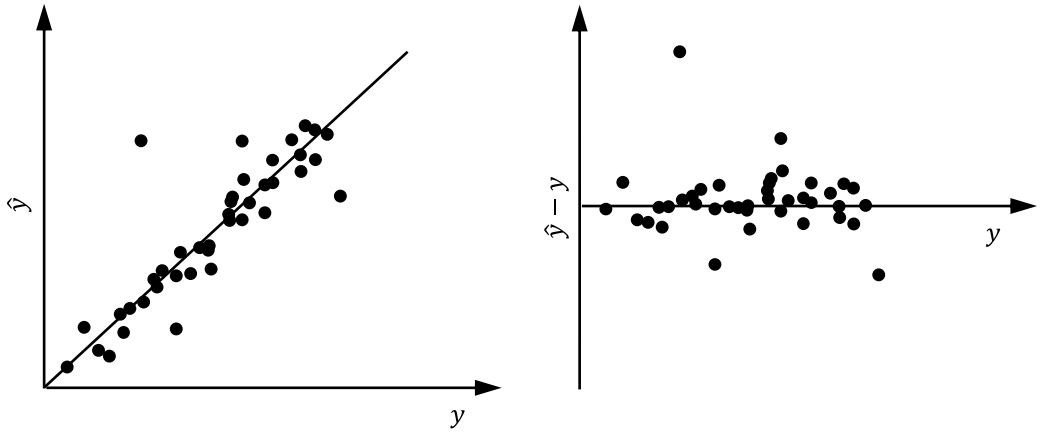


Abbildung 5-8: Graphische Darstellung der Regressionsergebnisse

Tabelle 5-7: Tabellarische Darstellung der Regressionsergebnisse am Beispiel von Gewichten

ID	y_i	$\hat{y}_i$	$\lvert y_i - \hat{y}_i \rvert$
1	3 835 kg	3 940 kg	105 kg
2	9 189 kg	9 110 kg	79 kg
3	6 962 kg	6 885 kg	77 kg
4	6 169 kg	6 100 kg	69 kg
5	7 316 kg	7 375 kg	59 kg
6	6 155 kg	6 100 kg	55 kg
7	7 184 kg	7 190 kg	6 kg

5.4.2.3 Einsatz

Regressionsmodelle können auf unterschiedliche Weise eingesetzt werden, um Wissen über komplexe Produktportfolios zu generieren (vgl. Kapitel 2.2.3). Die Modelle selbst können zur **Eigenschaftsvorhersage** eingesetzt werden. Es können die technischen und marktspezifischen Eigenschaften von Produktvarianten, welche zuvor noch nicht gebaut und verkauft wurden, bestimmt werden. Dafür sind als Eingangsgröße die Konfigurationen der Produktvarianten sowie die erwünschte Produkteigenschaft erforderlich. Des Weiteren können die Regressionsmodelle für die **Eigenschaftsoptimierung** von Produktvarianten eingesetzt werden (siehe Abbildung 5-9). Ausgehend von den zu optimierenden Konfigurationen mit den entsprechenden Merkmalen und Merkmalsausprägungen werden die Freiheitsgrade für die Optimierung ausgewählt. Anschließend werden unter Berücksichtigung der Einschränkungen der Kombinierbarkeit im Produktdatenmodell die möglichen neuen Produktvarianten bestimmt. Das Machine Learning Modell bestimmt dann für jede neue Produktvariante die Produkteigenschaften. Im Vergleich zu traditionellen Berechnungen oder Simulationen haben Machine Learning Modelle den Vorteil, dass die Berechnungsdauer um ein Vielfaches kürzer und dadurch die Eigenschaften von einer größeren Anzahl an Produktvarianten verglichen werden können. Darüber hinaus ist kein Wissen über die konkreten Lösungselemente der Produktvarianten erforderlich.

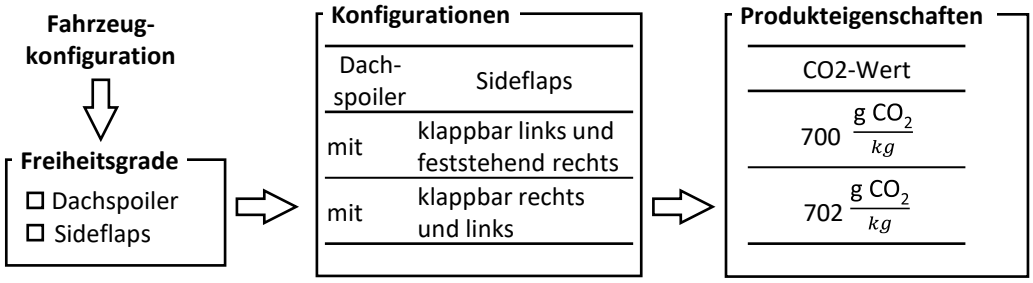


Abbildung 5-9: Eigenschaftsoptimierung von Produktvarianten mit Regressionsmodellen

Neben der direkten Nutzung der Regressionsmodelle kann Wissen durch die **Analyse der Modelle** generiert werden (vgl. Kapitel 2.2.3). Dies ist jedoch lediglich bei Modellen mit einer hohen Transparenz möglich, wie zum Beispiel bei der linearen Regression oder dem Entscheidungsbaum. Eine Möglichkeit ist die Bestimmung der Feature Importance, welche eine Punktzahl für die Wichtigkeit aller Merkmale eines Machine Learning Modells zuweist. Je nach verwendetem Modell ist das Verfahren zur Bestimmung der Wichtigkeitswerte unterschiedlich. Bei einer linearen Regression ist die Wichtigkeit identisch mit den Konstanten der Linearkombination. Bei einem Entscheidungsbaum oder Random Forest ergibt sie sich aus der Wahrscheinlichkeit, dass die Knoten des Merkmals bei der Entscheidung erreicht werden. Des Weiteren kann zum Beispiel bei einem Entscheidungsbaum die Struktur oder der Entscheidungspfad analysiert werden, um Muster zu erkennen und Entscheidungen zu verstehen (siehe Abbildung 5-10).

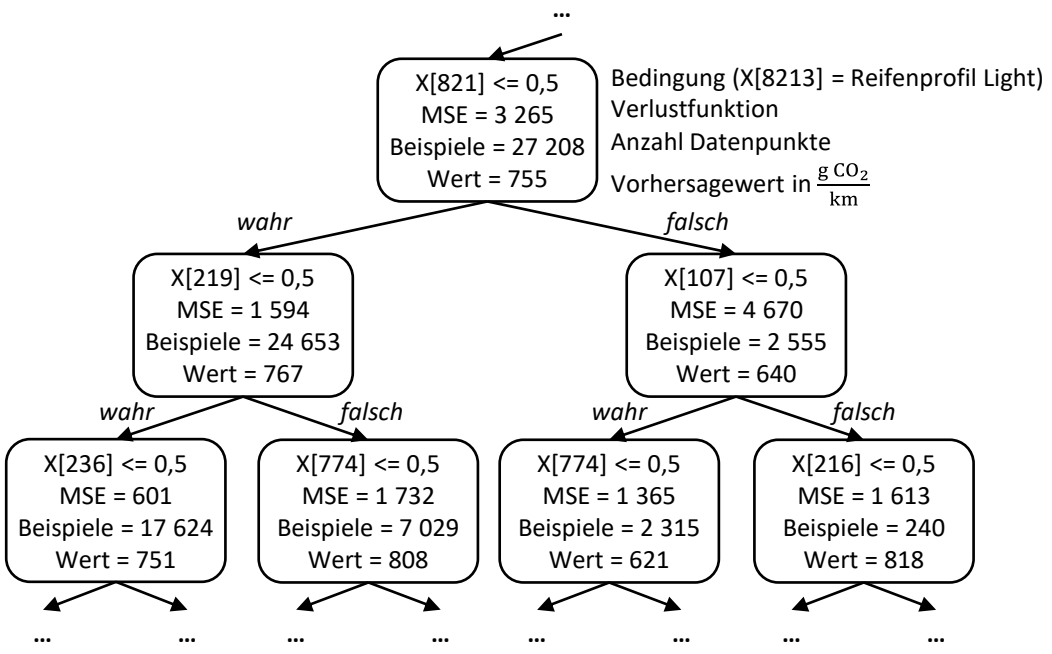


Abbildung 5-10: Ausschnitt eines Entscheidungsbaums am Beispiel der CO_2-Emission

Im Entscheidungsbaum können in jedem Knoten die Bedingung, der Wert der Verlustfunktion, die Anzahl der Datenpunkte aus den Trainingsdaten, die diesen Knoten durchlaufen haben, und der aktuelle Vorhersagewert ausgelesen werden.

5.4.3 Baustein 3.2: Klassifikationsanalyse

Im folgenden Unterkapitel wird auf die Auswahl und Evaluation von Klassifikationsalgorithmen sowie den Einsatz der Modelle eingegangen. Klassifikationsanalysen können unter anderem eingesetzt werden, um kategorische marktspezifische und technische Eigenschaften von Produktvarianten zu ermitteln (siehe Kapitel 5.2).

5.4.3.1 Klassifikation

Für die Analyse komplexer Produktportfolios sind vor allem Multi-Klassen-Klassifikationen Gegenstand der Analyse. Es gibt Algorithmen, wie zum Beispiel den Random-Forest, welcher direkt Multi-Klassen-Klassifikationsprobleme lösen kann. Andere Algorithmen, wie zum Beispiel eine SVM, sind rein binäre Klassifikatoren. Um einen binären Klassifikator für Multi-Klassen-Klassifikationen zu verwenden, können grundsätzlich zwei Strategien verfolgt werden (siehe Garcia-Pedrajas und Ortiz-Boyer 2006). Erstens die One-versus-All (OvA)-Strategie, bei der für jede Zielklasse ein binärer Klassifikator trainiert wird. Zweitens die One-versus-One (OvO)-Strategie, bei der hingegen für jedes Klassenpaar der Zielgröße ein binärer Klassifikator trainiert wird. In der Regel ist hier die OvA-Strategie zu wählen. Einige Algorithmen (z. B. SVM) skalieren schlecht mit der Größe der Trainingsmenge, sodass es schneller ist, viele Klassifikatoren auf kleine Trainingsmengen zu trainieren als wenige Klassifikatoren auf große Trainingsmengen und daher eine OvO-Strategie sinnvoll sein kann. Wie in Kapitel 2.2.5 beschrieben, können die meisten Regressionsalgorithmen auch für Klassifikationsaufgaben herangezogen werden. Lediglich die lineare Regression kann nicht für eine Klassifikation genutzt werden, weshalb die logistische Regression betrachtet wird. Die Eigenschaften der Algorithmen verhalten sich bei einer Klassifikationsaufgabe vergleichbar wie bei der Regression (Tabelle 5-8).

Tabelle 5-8: Faktoren für die Algorithmenauswahl bei einer Klassifikationsanalyse in Anlehnung an Mehlstäubl et al. (2022a)

	Datenpunkte	Nichtlinearität	Featureanzahl	Vorhersagezeit	Trainingszeit	Transparenz
Logistische Regression	sehr niedrig	sehr niedrig	niedrig	sehr niedrig	gering	sehr hoch
Support Vector Machine	niedrig	niedrig	niedrig	sehr hoch	sehr hoch	mittel
K-Nearest Neighbors	mittel	mittel	mittel	mittel	sehr gering	mittel
Entscheidungsbaum	mittel	mittel	hoch	sehr niedrig	gering	hoch
Random Forest	hoch	hoch	hoch	sehr niedrig	mittel	hoch
Neuronales Netz	sehr hoch	sehr hoch	sehr hoch	niedrig	hoch	sehr niedrig

Bei Klassifikationsaufgaben zur Analyse komplexer Produktportfolios sind die Zielgrößen oft keine absoluten Größen. Zum Beispiel kann es sein, dass eine Produktvariante mit einer identischen Konfiguration in unterschiedliche Länder verkauft wird. In einem solchen Fall ist es sinnvoll, nicht die tatsächliche Klasse zu bestimmen, sondern die Zugehörigkeitswahrscheinlichkeiten zu den Klassen. Diese gibt die Wahrscheinlichkeit auf Basis der Trainingsdaten an, dass die einzelnen Klassen eintreten (siehe Géron 2017).

5.4.3.2 Evaluation

Für die Evaluation der Klassifikationsergebnisse mit den Testdaten können, wie bei Regressionsmodellen, unterschiedliche statistische Kriterien herangezogen werden. Dabei ist die Auswahl vom Klassifikationsverfahren sowie der Klassifikationsaufgabe abhängig. Ein Werkzeug für die Leistungsmessung eines Klassifikationsproblems beim Machine Learning, bei dem die Ausgabe aus zwei oder mehr Klassen bestehen kann, ist die Konfusionsmatrix. Dabei handelt es sich um eine Matrix, welche die tatsächlichen Werte aus den Daten den vorhergesagten Werten gegenüberstellt. In Tabelle 5-9 ist eine binäre Konfusionsmatrix dargestellt.

Tabelle 5-9: Binäre Konfusionsmatrix nach Powers (2020)

		Tatsächliche Werte	
		Positiv	Negativ
Vorhergesagte Werte	Positiv	Richtig positiv (RP)	Falsch positiv (FP)
	Negativ	Falsch negativ (FN)	Richtig negativ (RN)

Die Konfusionsmatrix bildet den Ausgangspunkt für die Berechnung der Kriterien Accuracy, Precision, Recall und F1-Score (siehe Powers 2020). Die **Accuracy** ist der Anteil der richtigen Vorhersagen an der Gesamtzahl der untersuchten Fälle. Vorhersagen sind richtig, wenn sie mit den Werten in den Daten übereinstimmen. Die Accuracy ist geeignet für die Bewertung von Klassifikationsproblemen, bei denen kein Ungleichgewicht zwischen den Klassen besteht. Sie berechnet sich wie folgt:

$$Accuracy = \frac{richtige\ Vorhersagen}{alle\ Vorhersagen} = \frac{RP + RN}{RP + FP + FN + RP} \qquad \textit{Formel 5-7}$$

Die **Precision** gibt Auskunft darüber, welcher Anteil der positiven Vorhersagen wirklich positiv ist. Sie ist eine gute Wahl, wenn man sich bei einer positiven Vorhersage sehr sicher sein möchte. Die Precision wird wie folgt berechnet:

$$Precision = \frac{richtige\ positive\ Vorhersagen}{alle\ positiven\ Vorhersagen} = \frac{RP}{RP + FP} \qquad \textit{Formel 5-8}$$

Der **Recall** untersucht den Anteil der richtig klassifizierten positiven Vorhersagen gegenüber den tatsächlich positiven Ergebnissen. Der Recall ist ein gutes Bewertungskriterium, wenn möglichst viele richtig positiv Ergebnisse erzielt werden sollen:

$$Recall = \frac{richtige\ positive\ Vorhersagen}{richtig\ positive\ Vorhersagen + falsch\ negative\ Vorhersagen} = \frac{RP}{RP + FN} \quad \textit{Formel 5-9}$$

Der **F1-Score** ist eine Zahl zwischen 0 und 1 und ist das harmonische Mittel aus Precision und Recall. Damit soll für ein Gleichgewicht zwischen den beiden Scores gesorgt werden. Er berechnet sich wie folgt:

$$F1\ Score = \frac{2 * (Precision * Recall)}{Precision + Recall} \quad \textit{Formel 5-10}$$

Die Ergebnisse von Klassifikationsaufgaben zur Analyse komplexer Produktportfolios sind, wie im vorangegangenen Kapitel beschrieben, oft nicht absolut. Die beschriebenen Kriterien eignen sich nur bedingt für die Bewertung der Vorhersagen für die Zugehörigkeitswahrscheinlichkeiten zu den Klassen. Eine Metrik, welche auf der Zugehörigkeitswahrscheinlichkeit beruht und die Fähigkeit eines Modells zwischen den einzelnen Klassen zu unterscheiden quantifiziert, ist die Area Under the Receiver Operating Characteristic Curve **(ROC AUC)** (siehe Bex 2021). Sie gibt an, wie gut die Wahrscheinlichkeiten der positiven Klassen von den negativen Klassen getrennt sind. Zu Beginn wird eine Entscheidungsschwelle nahe 0 gewählt. Anhand dieser Schwelle werden Vorhersagen getroffen und eine Konfusionsmatrix erstellt. Auf Basis dieser werden zuerst die Richtig-Positive-Rate (RPR, gleich wie Recall) und Falsch-Positiv-Rate (FPR) berechnet:

$$RPR = \frac{RP}{RP + FN} \quad \textit{Formel 5-11}$$

$$FPR = \frac{FP}{FP + RN} \quad \textit{Formel 5-12}$$

Anschließend wird ein neuer, höherer Schwellenwert gewählt und eine neue Konfusionsmatrix erstellt. Dieser Vorgang wird für mehrere Entscheidungsschwellenwerte zwischen 0 und 1 wiederholt. Am Ende werden alle RPR und FPR gegeneinander aufgetragen und es entsteht eine Kurve wie sie in Abbildung 5-11 dargestellt ist. Umso größer die Fläche unter der Kurve, je besser sind die positiven Klassen von den negativen Klassen getrennt. Die ROC AUC ist jedoch nicht gut geeignet für die Evaluation von Aufgabenstellungen mit einem Ungleichgewicht in der Verteilung der Klassen. Bei einer Multi-Klassen-Klassifikation wird eine solche Kurve für jede Klasse berechnet.

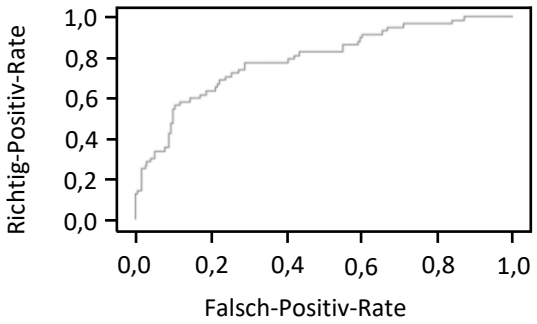


Abbildung 5-11: Beispielhafte ROC nach Bex (2021)

Eine robustere Metrik zur Evaluation von Zugehörigkeitswahrscheinlichkeiten ist die **Kreuz-Entropie Loss (Log LOSS)** (siehe Géron 2017)**.** Dabei handelt es sich um eine Fehlerfunktion, welche die Unsicherheit der Modelle berücksichtigt. Das heißt, umso kleiner die Kreuz-Entropie, desto besser das Klassifikationsmodell. Er berechnet sich aus der Multiplikation der negativen wahren Wahrscheinlichkeitsverteilung mit der logarithmischen vorhergesagten Wahrscheinlichkeitsverteilung über alle Klassen der Verteilung wie folgt:

$$H(P^*|P) = -\sum_i P^*(i) \log P(i) \qquad \textit{Formel 5-13}$$

5.4.3.3 Einsatz

Die Klassifikationsmodelle können für die **Eigenschaftsvorhersage** unterschiedlicher kategorischer marktspezifischer oder technischer Produkteigenschaften von neuen und noch nicht produzierten Produktvarianten genutzt werden. Hierfür sind als Eingabe die Konfiguration der neuen Produktvariante sowie die Produkteigenschaft auszuwählen. Eine mögliche Eingabeschnittstelle kann auf die gleiche Weise wie bei der Vorhersage kontinuierlicher Produkteigenschaften aufgebaut werden. Zudem kann eine **Eigenschaftsoptimierung** bei kategorischen Zielvariablen stattfinden, indem wie bei der Regression Freiheitsgrade in den Merkmalen definiert und die Eigenschaften für die resultierenden Produkteigenschaften bestimmt werden. Eine **Analyse der Modelle** mit einer hohen Transparenz (z. B. logistische Regression und Entscheidungsbaum) kann, wie bei der Regressionsanalyse, durch die Bestimmung der Feature Importance oder Analyse der Modellparameter stattfinden. Dies ermöglicht Rückschlüsse auf den Einfluss einzelner Merkmale auf das Vorhersageergebnis.

Die Klassifikationsmodelle können auch zur Analyse der **Eigenschaften von Teilkonfigurationen** eingesetzt werden. Da jedes Merkmal eine kategorische Variable darstellt, können auf Basis von Teilkonfigurationen die Wahrscheinlichkeiten für die Wahl der Merkmalsausprägungen eines Zielmerkmals vorhergesagt werden. Dies ermöglicht es, z. B. beim Wegfall einzelner Merkmalsausprägungen die Substitute zu ermitteln oder den Konfigurationsprozess zu unterstützen. Hierfür werden in einem ersten Schritt die

zu untersuchenden Eingangsmerkmale und das Zielmerkmal gewählt. Anschließend werden die Algorithmen trainiert und die Zielwerte vorhergesagt. Für eine sinnvolle Analyse sind die Einschränkungen im Produktdatenmodell zu berücksichtigen.

5.4.4 Baustein 3.3: Clusteranalyse

Eine Clusteranalyse kann im Produktportfolio- und Variantenmanagement eingesetzt werden, um die konfigurierten und verkauften Produktvarianten zu gruppieren (siehe Mehlstäubl et al. 2023c). Dies ermöglicht es, Ähnlichkeiten sowie Unterschiede zwischen den einzelnen Produktvarianten zu identifizieren. Dadurch können Produktvarianten zusammengefasst und Ausreißer im Produktportfolio identifiziert werden.

5.4.4.1 Clustering

In den einzelnen Clusteralgorithmen variiert die grundsätzliche Definition eines Clusters und dadurch die zugrundeliegenden Berechnungsschritte. Aus diesem Grund sind die einzelnen Algorithmen für bestimmte Datenstrukturen geeignet und für andere dagegen nicht. Clustert man dieselbe Datenmenge unter Verwendung verschiedener Algorithmen, so unterscheiden sich aus diesem Grund die gebildeten Cluster. Produktportfoliodaten besitzen eine große Varianz und eine hohe Dimensionalität, sodass ihre Struktur nur schwer ersichtlich ist. Durch eine Abbildung der Produktvarianten in einem zwei- oder dreidimensionalen Raum können Annahmen über deren Struktur getroffen werden. Dennoch ist es, wie beim überwachten Lernen, erforderlich, zunächst mehrere verschiedene Algorithmen zu implementieren und im Nachgang deren Leistungsfähigkeit bezüglich der gegebenen Daten zu vergleichen. Tabelle 5-10 zeigt eine Übersicht der verbreitetsten Clusteralgorithmen mit den zugehörigen Charakteristiken. Ein zweiter wichtiger Faktor für die Güte eines Clustering ist die Anzahl der Cluster. Aus diesem Grund ist ein sinnvoller Bereich für die Anzahl der Cluster vorzugeben, innerhalb dessen die Daten mehrfach unter Variation der Clusteranzahl geclustert werden.

5.4.4.2 Evaluation

Die Güte der ermittelten Cluster hängt, wie im vorangegangenen Kapitel erläutert, sowohl von der Wahl des geeigneten Algorithmus als auch von der Anzahl der Cluster ab. Daher müssen die Ergebnisse des Clustering unter Berücksichtigung dieser beiden Faktoren evaluiert werden. Zu diesem Zweck werden sogenannte Clustervalidierungsindizes (CVIs) verwendet. Jeder dieser CVIs untersucht einen spezifischen Aspekt der Struktur eines Clustering. Für die Evaluation der identifizierten Cluster sind daher mehrere CVIs zu berücksichtigen. Im Folgenden wird auf drei gebräuchliche CVIs eingegangen, welche unabhängig von dem verwendeten Algorithmus einsetzbar sind.

Tabelle 5-10: Überblick über die Algorithmen, deren Merkmale und Eignung nach Mehlstäubl et al. (2023c)

Algorithmus	Clusterdefinition	Merkmale	Eignung
k-Means *distanzbasiert*	Gruppe von Instanzen, die durch ihren nächstgelegenen Centroid repräsentiert werden	• simple Berechnung • abhängig von Initialisierung	+ sphärische Cluster - Clusterdichte & -größe variiert stark
Mini-batch k-Means *distanzbasiert*		• hohe Geschwindigkeit für große Datenmengen • geringere Genauigkeit	+ sphärische Cluster - Clusterdichte & -größe variiert stark
x-Means *distanzbasiert / probabilistisch*		• automatische Bestimmung optimaler Clusteranzahl k	+ sphärische Cluster - Überanpassung bei wenigen Instanzen pro Cluster - elliptische Cluster
EM *probabilistisch*	Clustering bestehend aus mehreren Wahrscheinlichkeitsverteilungen	• Form und Dichte von Verteilungen bestimmt • durchführbar für versch. Kovarianzmatrizen (z. B. rund, sphärisch)	+ elliptische oder sphärische Cluster - hohe Dimensionalität - wenige Instanzen pro Cluster
DBSCAN *dichtebasiert*	Cluster sind zusammenhängende Bereiche hoher Punktdichte	• Identifikation von Ausreißern möglich • Dichteparameter schwer zu bestimmen	+ Cluster beliebiger Form - Clusterdichte variiert stark
Linkage *hierarchisch*	Clustering besteht aus mehrstufiger Hierarchie ineinander liegender Cluster	• k muss nicht vorab gegeben werden • unterschiedliche Verknüpfungsfunktionen anwendbar	+ überlappende Cluster + sphärische Cluster - je nach Verknüpfungsfunktion: variierende Clustergröße & -dichte

Der erste betrachtete CVI ist der **Davies-Bouldin-Index**, der die durchschnittliche Ähnlichkeit eines Clusters zu dem jeweils ähnlichsten Cluster abbildet (Davies und Bouldin 1979). Er wird wie folgt berechnet:

$$\bar{R} = \frac{1}{N} \sum_{i=1}^{N} \frac{S_i + S_j}{M_{ij}} \qquad \textit{Formel 5-14}$$

S_i ist die Streuung der Instanzen innerhalb eines Clusters i und S_j ist die Streuung der Instanzen innerhalb des ähnlichsten Clusters j. Die Streuung bildet die durchschnittliche Entfernung zwischen den einzelnen Datenpunkten zum Centroid des Clusters ab. M_{ij} ist der Abstand der Centroide von Cluster i und dem ihm ähnlichsten Cluster j. Ein Clustering ist umso besser, je geringer die durchschnittliche Ähnlichkeit der darin enthaltenen Cluster ist. Daher sind kleinere Werte des Davies-Bouldin-Index Indikatoren für ein präziseres Clustering.

Der zweite CVI ist der **Silhouettenkoeffizient**. Er gibt an, wie akkurat die Zuordnung von Instanzen zu einem Cluster ist, indem er deren Ähnlichkeit mit den Instanzen im gleichen Cluster ins Verhältnis zu der Ähnlichkeit mit den Instanzen des nächstgelegenen Clusters setzt (Rousseeuw 1987). Der Silhouettenkoeffizient wird wie folgt berechnet:

$$s_i = \frac{b_i - a_i}{max\{a_i, b_i\}} \qquad \textit{Formel 5-15}$$

Dabei ist a_i die durchschnittliche Distanz einer Instanz i zu allen Instanzen, die demselben Cluster wie i zugeordnet sind. b_i ist die durchschnittliche Distanz von i zu allen Instanzen des i nächstgelegenen Clusters. Daraus folgt, dass je höher der Silhouettenkoeffizient ist, desto besser ist die Güte der Cluster eines Clustering. Der Silhouettenkoeffizient kann Werte zwischen -1 und 1 annehmen. Ein positiver Wert bedeutet, dass die Instanzen im selben Cluster ähnlicher sind als die im nächstgelegenen Cluster. Der gesamte Silhouettenkoeffizient eines Clustering wird aus dem Mittelwert der Silhouettenkoeffizienten der einzelnen Instanzen berechnet.

Der dritte berücksichtigte CVI wird als **Trägheit** bezeichnet und ergibt sich aus der Summe der quadrierten Distanzen der Instanzen zu dem jeweils nächstgelegenen Centroid. Dieser Index gibt die durchschnittliche Kompaktheit der Cluster eines Clustering an und wird wie folgt berechnet:

$$T = \sum_{i=1}^{k} \sum_{p \in C_i} dist(p, \mu_{C_i})^2 \qquad \textit{Formel 5-16}$$

Dabei steht C_i für die einzelnen Cluster, μ_{Ci} für deren Centroide sowie p für die Instanzen. Je kleiner die Trägheit eines Clustering, desto näher liegen die Instanzen an dem korrespondierenden Centroid und desto kompakter sind die einzelnen Cluster (Ester und Sander 2000). Mit einer steigenden Anzahl an Clustern sinkt die Trägheit zwangsläufig, da jede Instanz zu ihrem Centroid näher ist, umso mehr Centroide es gibt. Daher kann das präziseste Clustering nicht anhand eines besonders niedrigen Wertes der Trägheit abgeleitet werden, sondern muss durch die sogenannte Ellenbogen-Methode bestimmt werden (siehe Ester und Sander 2000). Dabei wird ein Graph erzeugt, der die Trägheit der Clusteranzahl gegenüberstellt (Abbildung 5-12).

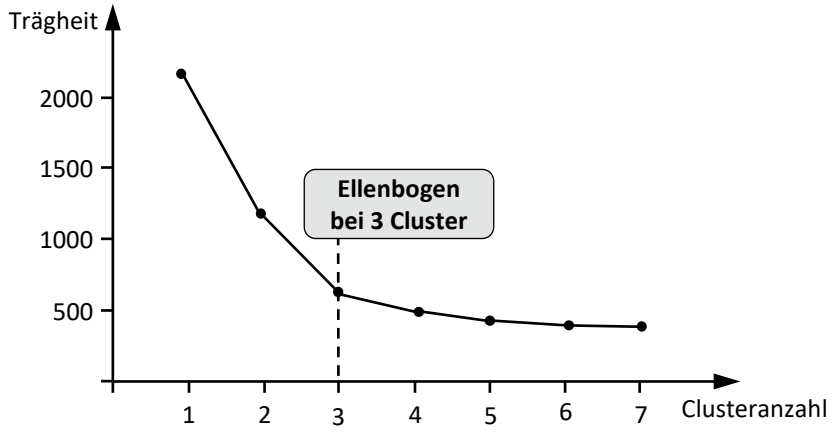


Abbildung 5-12: Ellenbogenverfahren in Anlehnung an Géron (2017)

Ein Ellenbogen ist ein Punkt, an dem die negative Steigung stark abflacht. Ein solcher Ellenbogen deutet darauf hin, dass die betreffende Anzahl von Clustern ein präzises Clustering ergibt, weil die Zerteilung der Instanzen in eine höhere Anzahl von Clustern nur in geringem Maße zu einer Verbesserung des Trägheitswertes führt.

Aus den vorangegangenen Erläuterungen zu den CVIs geht hervor, dass diese jeweils einen anderen Aspekt der Struktur des jeweiligen Clusters erfassen und daher ein einzelner CVI keine allgemeingültigen Aussagen ermöglicht. Aus diesem Grund ist es erforderlich alle drei CVIs gemeinsam zu betrachten (siehe Abbildung 5-13). Dies gilt sowohl für die Auswahl des Algorithmus als auch für die Identifikation der Clusteranzahl.

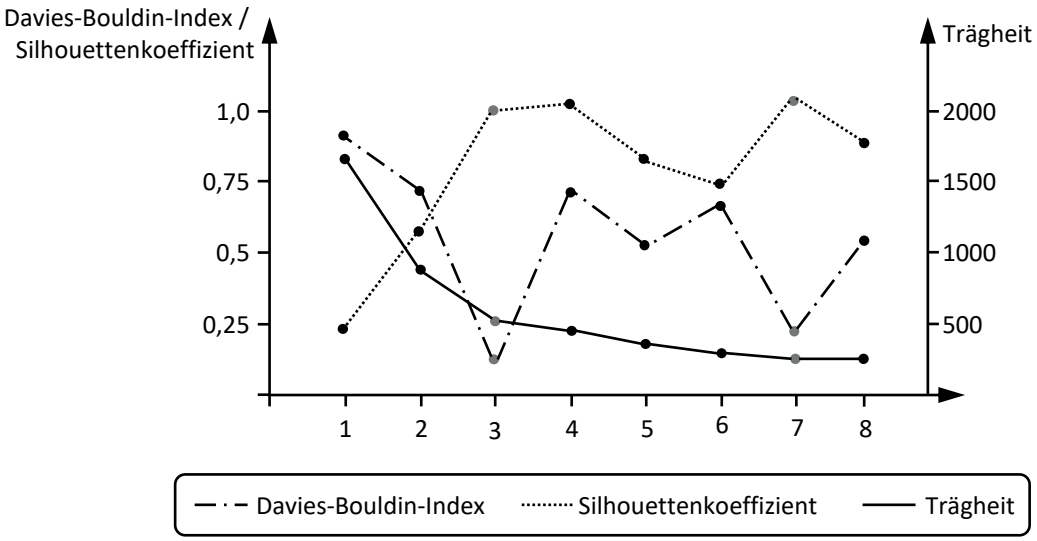


Abbildung 5-13: Identifikation des genauesten Clustering mit CVIs

5.4.4.3 Einsatz

Das Ergebnis des Clustering besteht aus der Zuordnung der IDs zu den Clustern. Um die Cluster interpretieren zu können, müssen die IDs mit den Merkmalen und Merkmalsausprägungen der Produktvarianten zusammengeführt werden. In der Analyse muss ausgewertet und visualisiert werden, wie häufig die Merkmale unter den Produktvarianten eines Clusters vorkommen. Daraus lassen sich die **charakteristischen Merkmalsausprägungen oder Komponentenvarianten** der Cluster ableiten. Merkmalsausprägungen sind charakteristisch für ein Cluster, wenn sie in allen oder fast allen Produktvarianten des Clusters vorkommen. Dies ermöglicht die Analyse von Alleinstellungsmerkmalen und Gemeinsamkeiten innerhalb der Cluster sowie zwischen den Clustern (siehe Abbildung 5-14).

Darüber hinaus können die **Abstände** zwischen den Clustern bzw. deren Centroide sowie den einzelnen Produktvarianten untersucht werden. Dies kann entweder durch eine graphische Visualisierung der Cluster im zwei- oder dreidimensionalen Raum oder durch eine Erfassung der exakten Werte und deren Gegenüberstellung in einer Matrix

geschehen. Eine Visualisierung ermöglicht es, auf einen Blick Auffälligkeiten im Clustering zu erkennen (siehe Abbildung 5-14). Durch die Gegenüberstellung in einer Matrix können die exakten Werte untersucht werden.

Ein weiterer Ansatzpunkt ist die Untersuchung der Cluster hinsichtlich **marktspezifischer Eigenschaften**. Dies beinhaltet die Zuordnung dieser zu den Produktvarianten und Clustern. Die Stückzahlen können direkt den Eingangsdaten entnommen werden. Daneben sind Daten zu Kosten und Erlösen relevant, um beispielsweise abzuschätzen, ob und wie rentabel einzelne Cluster im Produktportfolio sind.

Merkmal	Gemeinsame Merkmalsausprägungen
Anwendung	Distribution
Federungsart	Blatt/Luft
Fahrerhaus	Groß
…	…

Merkmal	Einzigartige Merkmalsausprägungen
Radformel	8x4
Motor	Typ X
Fahrerhaus	Extra klein
…	…

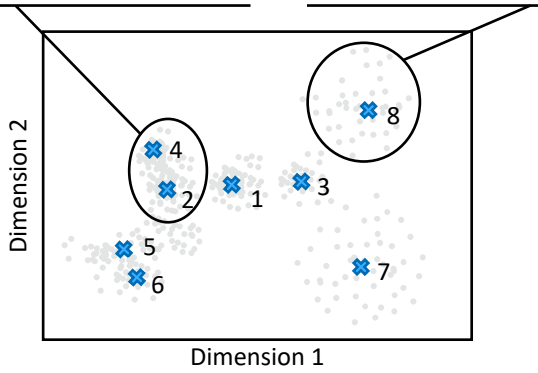


Abbildung 5-14: Untersuchung der ermittelten Cluster

5.4.5 Baustein 3.4: Assoziationsanalyse

Mit einer Assoziationsanalyse können komplexe Produktportfolios im Hinblick auf Korrelationen zwischen Merkmalsausprägungen oder Komponentenvarianten analysiert werden (siehe Mehlstäubl et al. 2023b). Diese geben Einblicke darüber, welche Merkmalsausprägungen häufig zusammen auftreten. Diese können beispielsweise zu Modulen zusammengefasst werden. Im Umkehrschluss ermöglichen die Assoziationsregeln aber auch die Ermittlung von Kombinationen aus Merkmalsausprägungen, welche nie oder selten zusammen verkauft werden.

5.4.5.1 Assoziation

Die Assoziationsalgorithmen unterscheiden sich, wie in Kapitel 2.2.5 beschrieben, in erster Linie hinsichtlich der Ermittlung der Kandidaten und der Anzahl der Durchläufe. Dies hat wiederum Auswirkungen auf die Speicherauslastung, die Geschwindigkeiten in den frühen und späten Phasen, die Genauigkeit sowie die Anwendung der Algorithmen (Vani 2015; Prithiviraj und Porkodi 2015). Die Daten komplexer Produktportfolios

beinhalten eine Vielzahl an Merkmalen und Merkmalsausprägungen sowie Produktvarianten, weshalb für deren Analyse in erster Linie der AprioriHybrid und der FP-Growth Algorithmus in Frage kommen.

Tabelle 5-11: Gegenüberstellung der Assoziationsalgorithmen in Anlehnung an Vani (2015) und Prithiviraj und Porkodi (2015)

	AIS	Apriori	AprioriTid	AprioriHybrid	FP-Growth
Speicherauslastung	sehr hoch	hoch	hoch	mittel	gering
Anzahl der Durchläufe	viele	viele	viele	viele	zwei
Geschwindigkeit (frühe Phasen)	sehr gering	gering	gering	hoch	hoch
Geschwindigkeit (späte Phasen)	sehr gering	gering	hoch	hoch	hoch
Genauigkeit	sehr gering	gering	mittel	hoch	sehr hoch
Anwendung (Größe Datensatz)	klein	klein	klein	mittel	groß

5.4.5.2 Evaluation

Ein Kriterium zur Messung der Güter der erzeugten Assoziationsregeln im Produktportfolio- und Variantenmanagement ist der Confidence. Der Confidence-Wert misst, wie oft Y in Produktkonfigurationen, welche X beinhalten, auftreten (Han et al. 2000). Er wird wie folgt berechnet:

$$\text{Confidence, c}(\text{X} \rightarrow Y) = \frac{\#(X \cup \text{Y})}{\#(X)} \qquad \text{Formel 5-17}$$

Ist der Confidence-Wert nahe oder gleich eins, treten die Merkmalsausprägungen fast immer oder immer zusammen auf. Dadurch kann z. B. eine Regel der Form „Merkmalsausprägung A erfordert Merkmalsausprägung B“ eingeführt werden. Die Regel schränkt die Kombinierbarkeit aller anderen Merkmalsausprägungen der entsprechenden Merkmale ein.

5.4.5.3 Einsatz

In der Produktportfoliogestaltung werden beim Aufbau des Produktdatenmodells hauptsächlich technische Einschränkungen berücksichtigt. Durch eine Assoziationsanalyse der verkauften Produktkonfigurationen werden Einschränkungen aufgrund der Kundenbedürfnisse ermittelt. Daher werden in diesem Schritt zuerst die bestehenden Einschränkungen im Produktdatenmodell mit den durch die Assoziationsanalyse generierten Regeln verglichen. Ein solches Regelwerk kann bei der Betrachtung komplexer Produktportfolios tausende von Regeln besitzen (Braun et al. 2017), weshalb ein sequenzieller Ansatz auf der Grundlage der einzelnen Merkmale zu empfehlen ist. Die Festlegung von Kombinatorikregeln schränkt die Konfigurierbarkeit der Merkmalsausprägungen ein. Dies führt zunächst zu einer Verringerung der unnötigen externen

Vielfalt. Durch die Verknüpfung von Kunden- und Technikperspektive über die Teileauswahlregeln ergeben sich direkte Auswirkungen auf die interne Vielfalt und Komplexität. Daher muss ermittelt werden, welche Baugruppen aufgrund der neuen Einschränkungen überflüssig werden und welche Kosteneinsparungen möglich sind. In Abbildung 5-15 wurden die Kombinatorikregeln „B1 verbietet C3“, „B2 erfordert C2“ und „A2 verbietet B1“ als Beispiel dargestellt, so dass die Komponentenvarianten K1b und K1c nicht mehr benötigt werden.

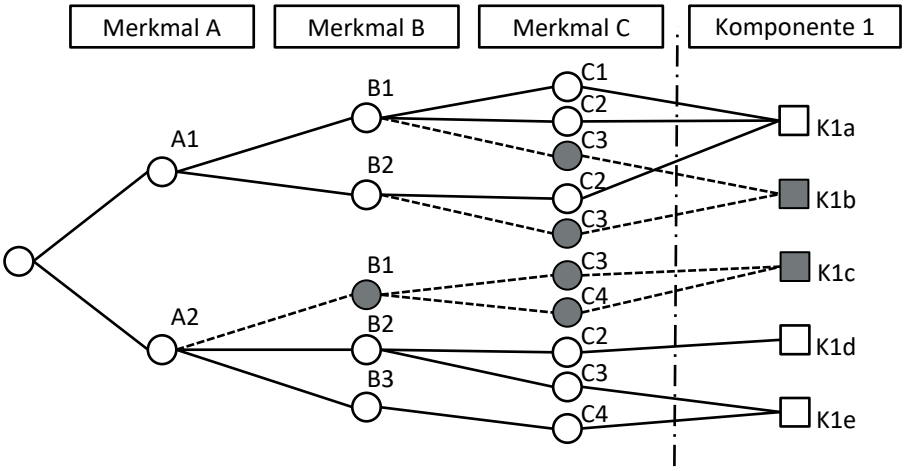


Abbildung 5-15: Reduktion des Produktportfolios durch die Einführung von Kombinatorikregeln nach Mehlstäubl et al. (2023b)

5.5 Anwendung des Frameworks

Nachdem die einzelnen Bausteine sowie deren Elemente im Detail beschrieben wurden, wird in diesem Kapitel auf das Vorgehen zur Anwendung des Frameworks eingegangen. In Abbildung 5-16 ist eine Übersicht der Aktivitäten sowie deren Eingangs- und Ausgangsgrößen dargestellt. Hervorgehoben sind dabei die einzelnen Artefakte, welche durch das Framework bereitgestellt werden.

In Baustein 1 bilden die beschriebenen Wissensbedarfe den Ausgangspunkt. Diese sind in einem ersten Schritt mit den definierten Bewertungskriterien für Machine Learning Anwendungsfälle in Abhängigkeit der im jeweiligen Unternehmen vorliegenden Gegebenheiten zu bewerten. Anschließend sind die Wissensbedarfe hinsichtlich des Nutzens sowie des Vorgehens und des Machine Learning Verfahrens zu konkretisieren.

Baustein 2 nutzt als Basis die zuvor beschriebenen Anwendungsfälle. Auf Basis dieser und den beschriebenen Produktportfoliodaten des Frameworks werden die Datenbedarfe für deren Umsetzung ausgewählt. Anschließend sind die Daten im Unternehmen zu beschaffen und zu charakterisieren.

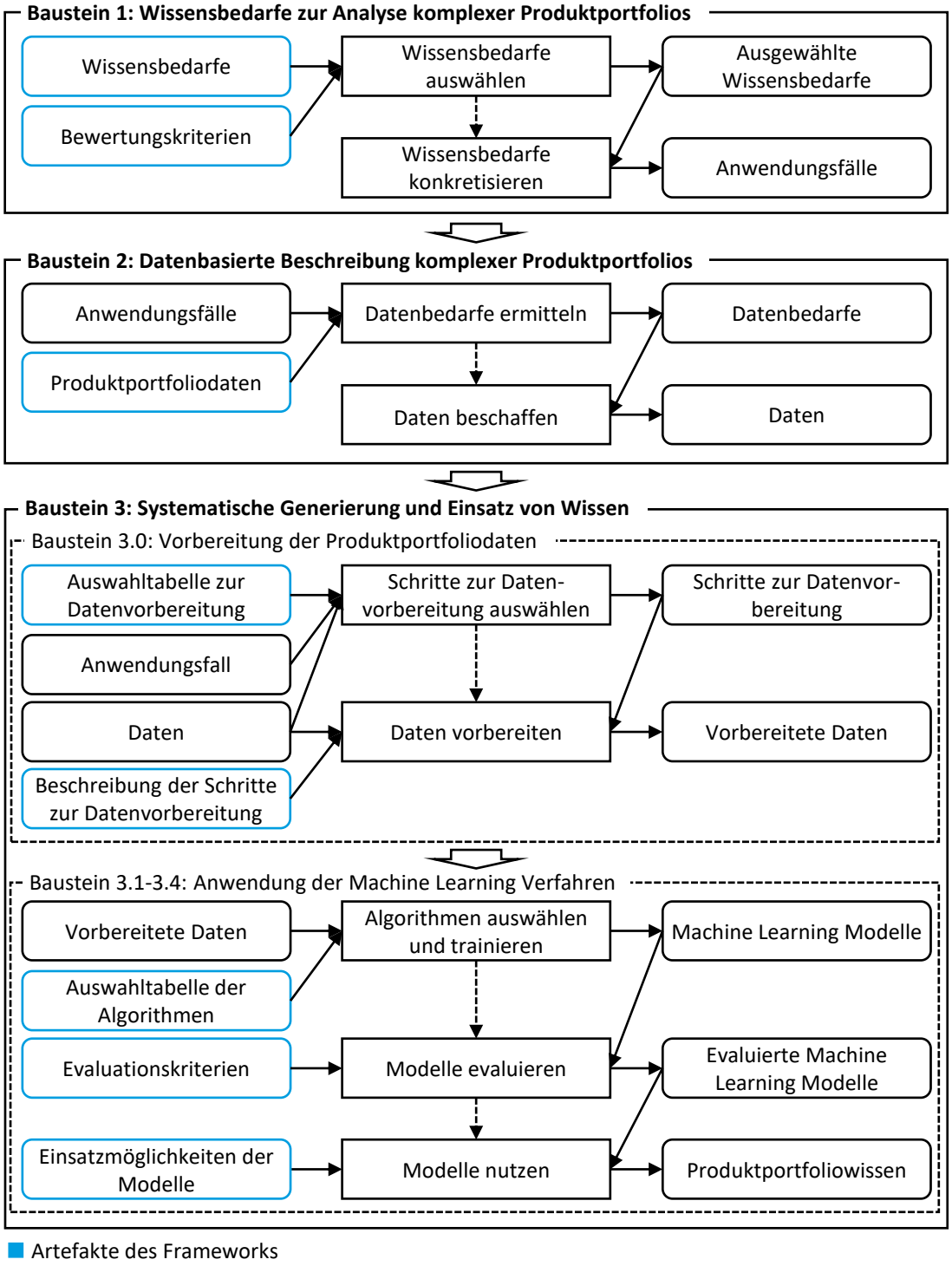


Abbildung 5-16: Vorgehen zur Anwendung des Frameworks

In Baustein 3 findet die Datenvorbereitung, Modellbildung und -evaluation sowie der Einsatz der Modelle und des generierten Wissens statt. Hierzu werden in Baustein 3.0 zuerst die erforderlichen Schritte für die Datenvorbereitung mit der im Framework bereitgestellten Auswahltabelle auf Basis der verwendeten Daten und Analyseverfahren

ausgewählt. Anschließend finden die Bereinigung und Transformation der Produktportfoliodaten mit den beschriebenen Verfahren statt.

In den Bausteinen 3.1 bis 3.4 werden die vorbereiteten Daten in Abhängigkeit des erforderlichen Machine Learning Verfahrens analysiert. Im ersten Schritt werden hierfür mit der jeweiligen Auswahltabelle die geeigneten Algorithmen ausgewählt und die Modelle trainiert. Hierbei sind die erfolgversprechendsten Algorithmen zu implementieren und anschließend mit den zur Verfügung gestellten Kriterien zu evaluieren. Das Modell mit der besten Güte wird anschließend zur Analyse des Produktportfolios eingesetzt. Dafür werden in jedem Analysebaustein verschiedene Einsatzmöglichkeiten beschrieben.

5.6 Schlussfolgerung zum Framework

In diesem Kapitel wurde das Framework zur systematischen Generierung von Wissen für die Analyse komplexer Produktportfolios mittels Machine Learning eingeführt. Die Entwicklung des Frameworks geschah anhand definierter Kriterien, welche zum einen aus dem Stand der Forschung und zum anderen aus den Beobachtungen in der Industrie abgeleitet wurden (siehe Kapitel 4.3). Das Framework bedient sich an Verfahren und Techniken des Machine Learning und bringt diese in den spezifischen Kontext der operativen Produktportfoliogestaltung, um die Handhabbarkeit der Produktportfoliokomplexität in Industrieunternehmen zu verbessern.

Im ersten Baustein des Frameworks wurde ein Geschäftsverständnis für den Einsatz von Machine Learning zur Generierung von Wissen über komplexe Produktportfolios beschrieben. Dazu wurden Anwendungsfälle aus der Literatur und Praxis untersucht und durch die Einordnung in den Entscheidungsprozess zur Analyse und Anpassung komplexer Produktportfolios zu Wissensbedarfen zusammengefasst. Dabei wurde kein Anspruch auf Vollständigkeit erhoben. Stattdessen wurde zum einen ein Verständnis für die aktuellen Herausforderungen bei der Analyse komplexer Produktportfolios bereitgestellt und zum anderen ein Ausgangspunkt für Unternehmen für die Einführung von Machine Learning in der operativen Produktportfoliogestaltung dargelegt.

Anschließend wurde im zweiten Baustein ein Datenverständnis für komplexe Produktportfolios generiert. Dabei wurde auf das Produktdatenmodell, die Vertriebsdaten und die Nutzungsdaten eingegangen. Der Ausgangspunkt bildet das Produktdatenmodell. Ausgehend von diesem werden im Konfigurationsprozess die Produktvarianten konfiguriert und anschließend produziert. Informationen über die Struktur der Produktvarianten, die Produkteigenschaften und die Kunden werden in den Vertriebsdaten gespeichert. Bei der Produktnutzung wird das Verhalten der Produktvarianten sowie deren Interaktion mit dem Nutzer in den Nutzungsdaten gespeichert. Neben dem Aufbau der Daten wurden deren typische Charakteristiken mit den zugrundeliegenden Ursachen beschrieben. Wodurch die anschließende Analyse ermöglicht wird. In

diesem Baustein wurde ebenfalls kein Anspruch auf Vollständigkeit erhoben, da im Sinne des Design for X alle im Lebenszyklus einer Produktvariante generierten Daten Rückschlüsse auf die Entwicklung zulassen. Stattdessen wurden die wichtigsten Daten für die Umsetzung der Anwendungsfälle ermittelt.

Ziel des dritten Bausteins war es, eine Unterstützung bei der systematischen Generierung und dem Einsatz von Wissen für die Analyse komplexer Produktportfolios bereitzustellen. Dabei wurden auf Basis der datenbasierten Beschreibung aus Baustein 2 die Datenvorbereitung, Modellbildung und Evaluation sowie der Einsatz im Kontext der Analyse komplexer Produktportfolios in Abhängigkeit der Machine Learning Verfahren beschrieben. Für die Datenvorbereitung wurde eine Auswahltabelle entwickelt, welche die notwendigen Verfahren für den spezifischen Kontext beinhaltet. In den Machine Learning Unterbausteinen wurde die Auswahl und Evaluation der Algorithmen sowie der zielorientierte Einsatz dargelegt. Die Eingrenzung auf die verbreitetsten Algorithmen ist für die initiale Implementierung von Machine Learning Anwendungsfällen und damit für den Zweck dieses Frameworks ausreichend. Ebenfalls konnten beim Einsatz der Algorithmen nicht alle im ersten Baustein des Frameworks aufgeführten Wissensbedarfe vollumfänglich erläutert werden. Stattdessen werden die grundsätzlichen Fähigkeiten und Eigenschaften der Modelle dargelegt, damit diese sinnvoll für Anwendungsfälle des Machine Learning zur Analyse komplexer Produktportfolios eingesetzt werden können. Dadurch wurde die Grundlage bereitgestellt, um die in Baustein 1 beschriebenen Wissensbedarfe zielorientiert in Unternehmen zu implementieren und einzusetzen.

6 Validierung des Frameworks

In diesem Kapitel wird die Validierung des entwickelten Frameworks im industriellen Kontext erläutert. Hierfür wird zuerst auf das Konzept der Validierung eingegangen. Anschließend wird die Anwendung des Frameworks bei einem Industriepartner aus der Nutzfahrzeugbranche vorgestellt. Abschließend werden die Ergebnisse der Bewertung des Frameworks durch Experten anhand der definierten Kriterien im Rahmen der Erfolgsvalidierung dargelegt.

6.1 Konzept der Validierung

Die Validierung der drei Bausteine des Frameworks erfolgte im Rahmen der deskriptiven Studie II nach Blessing und Chakrabarti (2009). Diese beinhaltet eine Anwendungs- und Erfolgsvalidierung sowie Implikationen und Empfehlungen zur Verbesserung der Entwicklungsunterstützung. Die Anwendungsvalidierung enthält Untersuchungen, ob die Entwicklungsunterstützung im industriellen Kontext tatsächlich eingesetzt werden kann und ob die Schlüsselfaktoren wie beabsichtigt beeinflusst werden. Die Erfolgsvalidierung zielt darauf ab, die Nützlichkeit der Unterstützung zu bewerten. D. h. zu untersuchen, ob das entwickelte Artefakt den gewünschten Gesamteffekt und die Erfolgskriterien erreicht und ein Mehrwert aus Sicht der Forschung darstellt.

In dieser Arbeit wird der Forschungstyp 5 der DRM verfolgt, welcher eine **initiale deskriptive Studie II** vorsieht. Eine initiale deskriptive Studie II beinhaltet alle Schritte einer umfassenden deskriptive Studie II, jedoch mit geringerem Umfang oder in weniger detaillierter Form. Ziel ist die prinzipielle Anwendbarkeit und Nützlichkeit der Entwicklungsunterstützung nachzuweisen. Die **Anwendungsvalidierung** fand durch den Einsatz des Frameworks bei einem Nutzfahrzeugherstellers satt. Nutzfahrzeuge besitzen eine Vielzahl an unterschiedlichen Transportaufgaben und Einsatzszenarien, woraus eine besonders hohe Variantenvielfalt resultiert (Kreimeyer et al. 2016). Darüber hinaus reagieren die Hersteller aufgrund ihrer strategischen Ausrichtung auf den Wettbewerbsdruck mit einer horizontalen und vertikalen Erweiterung des Produktportfolios (Lehmann und Grzegorski 2008). Im Rahmen der Validierung von Baustein 1 und Baustein 2 wurden Gespräche und Dokumentenanalysen beim Industriepartner durchgeführt, um Wissensbedarfe auszuwählen und näher zu spezifizieren sowie ein Verständnis für die Daten und Systeme zu erlangen. In Baustein 3 wurden reale Daten des Industriepartners verwendet, um Wissen für die operative Produktportfoliogestaltung zu generieren. Die Ergebnisse der Anwendungsvalidierung dienten als Ausgangsbasis für die Durchführung von Befragungen zur **Erfolgsvalidierung**. Diese wurden zum einen mit vier Experten beim Industriepartner und zum anderen mit fünf Experten außerhalb des Unternehmens durchgeführt, um die Erfüllung der definierten Ziele und der gestellten Anforderungen zu messen sowie den Bedarf an weiteren Forschungsaktivitäten abzuleiten. Die Ergebnisse des Einsatzes des Frameworks wurden den Exper-

ten vorgestellt und anschließend von diesen anhand der in Kapitel 4.3 definierten Kriterien bewertet. Dafür wurden die Kriterien in konkrete Aussagen umformuliert und die Zustimmung der Experten abgefragt.

6.2 Baustein 1: Wissensbedarfe zur Analyse komplexer Produktportfolios

In Baustein 1 wurde beim Industriepartner ein Verständnis für die Wissensbedarfe generiert. Dafür wurden diese hinsichtlich der Machbarkeit und der Attraktivität für das Unternehmen mit den in Kapitel 5.2.4 definierten Kriterien bewertet. Die Ergebnisse sind in Tabelle 6-1 dargestellt. Vor allem die Datenverfügbarkeit und die Datenmenge stellten bei einigen Wissensbedarfen eine Herausforderung dar, da die Entscheidungsgrundlage von Experten z. B. über nicht profitable Merkmalsausprägungen oder Kombinationen von Merkmalsausprägungen nicht oder lediglich lokal dokumentiert wurden.

Tabelle 6-1: Bewertung und Auswahl der Wissensbedarfe

Phasen	Wissensbedarfe	Datenverfügbarkeit	Datenmenge	Datensicherheit	Nutzungsbereitschaft	Zusammenarbeit Nutzer	Nutzen	Aufwand	Integrationsrisiko
Informationssuche	W1: Marktspezifische Eigenschaften der Produktvarianten	x	x	x	x	x	x	x	x
	W2: Technische Eigenschaften der Produktvarianten	x	x	x	x	x	x	x	x
	W3: Zeitliche Entwicklung marktspezifischer Größen	x	-	x	x	x	x	x	x
Formulierung von Alternativen	W4: Nicht profitable Merkmalsausprägungen	-	-	x	x	x	x	x	x
	W5: Ähnlichkeiten von Produktvarianten	x	x	x	x	x	x	x	x
	W6: Korrelationen zwischen Merkmalsausprägungen	x	x	x	x	x	x	x	x
Prognose der Auswirkungen	W7: Präferenzen von Kunden und Kundensegmenten	-	-	x	x	x	x	-	x
	W8: Auswirkungen der Produktportfolioänderungen	-	-	x	x	x	x	-	x

x = erfüllt, - nicht erfüllt

Die aufgrund der Datenlage negativ bewerteten Wissensbedarfe bilden jedoch einen Ansatzpunkt für das systematische Erzeugen und Speichern von Daten, um die Anwendungsfälle zukünftig umsetzen zu können. Anhand der Bewertung wurden vier Wissensbedarfe ermittelt, welche im Rahmen der Fallstudie beim Industriepartner umgesetzt wurden. Diese sind die Ermittlung von „marktspezifischen Eigenschaften der Produktvarianten“, „technischen Eigenschaften der Produktvarianten“, „Ähnlichkeiten von Produktvarianten“ und „Korrelationen zwischen Merkmalsausprägungen“. Für die Anwendung wurden die Prognose der marktspezifischen und technischen Eigenschaften der Produktvarianten zusammengefasst, da in beiden Fällen Regressions- und Klassifikationsanalysen verwendet wurden. Im Folgenden werden die Zielsetzung, der Inhalt sowie die aktuellen Verfahren beim Industriepartner und die Vorteile von Machine Learning in den Anwendungsfällen diskutiert.

Technische und marktspezifische Eigenschaften der Produktvarianten

Unternehmen mit komplexen Produktportfolios müssen die Eigenschaften neuer Produktkonfigurationen kennen, bevor diese verkauft und gebaut werden. Diese müssen zum einen den Kunden während des Konfigurationsprozesses zur Verfügung gestellt werden. Zum anderen sind sie notwendig, um zu beurteilen, ob die neuen Konfigurationen zum Beispiel die rechtlichen Anforderungen erfüllen und zugelassen werden können. Ziel des Anwendungsfalls war es, die Werte von technischen Fahrzeugeigenschaften (z. B. Produktgewicht oder -länge) und marktspezifischen Fahrzeugeigenschaften (z. B. Kosten oder Vertriebsland) auf der Grundlage von Merkmalen und deren Merkmalsausprägungen vorherzusagen (siehe Abbildung 6-1).

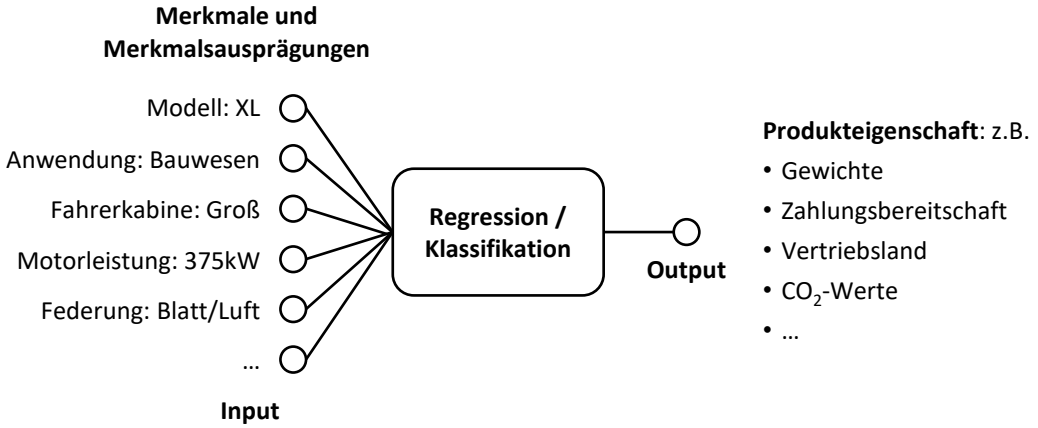


Abbildung 6-1: Architektur zur Prognose von Produkteigenschaften nach Mehlstäubl et al. (2022a)

Die ermittelten technischen Produkteigenschaften waren die Fahrzeuggewichte und CO_2-Emissionen und die marktspezifischen Produkteigenschaften waren die Zahlungsbereitschaft der Kunden und die Vertriebsländer. Je nach Art der Fahrzeugeigenschaft wurde eine Regressions- oder Klassifikationsanalyse durchgeführt. Handelte es sich bei

der Produkteigenschaft um eine kontinuierliche Größe (z. B. Gewicht in kg), wurde eine Regressionsanalyse gewählt, um die Beziehung zwischen den Konfigurationen aus Merkmalsausprägungen und den Werten der Produkteigenschaft zu modellieren. Ist die Produkteigenschaft eine kategorische Größe (z. B. Vertriebsland), wurde eine Klassifikationsanalyse durchgeführt, um die Produktkonfiguration einer der vordefinierten Klassen zuzuordnen. Je nachdem wie die Produkteigenschaften zuvor bestimmt wurden, ergaben sich durch Machine Learning in diesem Anwendungsfall unterschiedliche Vorteile. Machine Learning ermöglichte die automatisierte Ableitung der Zusammenhänge zwischen Merkmalsausprägungen und den Produkteigenschaften, ohne dass die jeweiligen Komponenten dazwischen identifiziert werden mussten. Dies reduzierte den Aufwand im Vergleich zu einer manuellen Regeldefinition. Außerdem konnte die Vorhersagegeschwindigkeit sowie die Genauigkeit der Prognosen erhöht werden.

Ähnlichkeiten von Produktvarianten

Die Identifikation der Ähnlichkeiten zwischen Produktvarianten beinhaltete die Bestimmung der Abstände bzw. Nähe zwischen den Datenpunkten. Es wurde eine Clusteranalyse eingesetzt, um die Produktvarianten in Gruppen mit ähnlichen Produktvarianten einzuteilen. Als Datengrundlage wurden die Merkmale und Merkmalsausprägungen von verkauften und gebauten Fahrzeugen herangezogen. Durch eine Untersuchung der Cluster konnten deren Centroide und charakteristischen Merkmalsausprägungen, welche für die Gemeinsamkeiten und Unterschiede verantwortlich sind, ermittelt werden. Heutzutage gibt es beim Industriepartner kein systematisches Verfahren, welches auf Basis der Merkmalsausprägungen ähnliche Produktvarianten identifiziert. Aktuell werden diese durch sogenannte Grundfahrzeuge und das Wissen von Experten über die Merkmale und deren Bedeutung für das Fahrzeug ermittelt. Durch das Clustering konnte dies automatisiert für alle Fahrzeugkonfigurationen objektiv auf Basis der Daten bestimmt werden. Wissen über die Ähnlichkeiten von Produktvarianten bietet einen Mehrwert in vielerlei Hinsicht. Es können zum Beispiel Produktvarianten zusammengefasst werden oder bei einer Reduktion des Produktportfolios Substitute ermittelt werden.

Korrelationen zwischen Merkmalsausprägungen

Die immer größer werdende Anzahl an kombinierbaren Merkmalsausprägungen resultiert in einer enormen Anzahl an möglichen Produktkonfigurationen (siehe Kapitel 1.2). Entscheidungen über Einschränkungen werden folglich nicht mehr auf Produktebene, sondern auf der Ebene von Merkmalsausprägungen getroffen (Heina 1999). In diesem Anwendungsfall wurden mit einer Assoziationsanalyse Regeln zwischen den Merkmalsausprägungen ermittelt, evaluiert und als Kombinatorikregeln zur Einschränkung des Produktportfolios formuliert. Die Einschränkung der Kombinierbarkeit der Merkmalsausprägungen führte zu einer Reduktion von nicht erforderlicher Varianz. Heutzutage werden diese Regeln von Experten definiert. Dabei werden in erster Linie die Entwickler einbezogen. Durch die Analyse der Daten verkaufter Produktkon-

figurationen konnten die Kundenbedürfnisse bei der Definition von Produktportfolioeinschränkungen berücksichtigt werden.

6.3 Baustein 2: Datenbasierte Beschreibung komplexer Produktportfolios

Für die Umsetzung der beschriebenen Anwendungsfälle wurden das Produktdatenmodell und die Vertriebsdaten benötigt. Darüber hinaus wurden ebenfalls die Nutzungsdaten beim Industriepartner betrachtet. Für die technischen und marktspezifischen Eigenschaften von Produktvarianten wurden aus dem Vertriebsdatenrumpf die exakten Konfigurationen in Form der Merkmale und Merkmalsausprägungen entnommen. Die zu prognostizierenden Produkteigenschaften wurden im Vertriebsdatenkopf gespeichert. In diesem Fall waren das die Fahrzeuggewichte, CO_2-Werte, Vertriebsländer und die Fahrzeugpreise. Für die Ermittlung der Ähnlichkeiten von Produktvarianten und der Korrelationen zwischen Merkmalsausprägungen war ebenfalls der Vertriebsdatenrumpf notwendig. Für die Identifikation von Ähnlichkeit zwischen Produktvarianten können zusätzlich Nutzungsdaten in die Analyse mit einbezogen werden.

Das Produktdatenmodell bestand beim Industriepartner, wie auch in Kapitel 5.3.1 beschrieben, aus einer Merkmalsstruktur mit Merkmalen und Merkmalsausprägungen sowie einer Produktstruktur mit Komponenten und Komponentenvarianten. Das Produktdatenmodell wurde im PDM-System definiert und gespeichert. Abfragen zur Konfigurierbarkeit konnten über eine Rule Engine gemacht werden. Anfang 2020 wurde beim Industriepartner im Zuge einer neuen Produktgeneration ebenfalls ein neues Produktdatenmodell eingeführt, weshalb die Daten zuvor nicht für die Analyse verwendet werden konnten. In dem Produktdatenmodell waren die einzelnen Elemente mit eindeutigen Kürzeln abgebildet. Zum Beispiel wurde das Merkmal „Radformel“ mit „0KHV“ und dessen Merkmalsausprägung „4x2“ mit „0P3B1“ sowie die Komponente „Bremsregelventile Hinterachse“ mit „005P“ und deren Komponentenvariante „Bremsregelventile Hinterachse, Basic“ mit „81.#005P-0001“ kodiert.

Der Konfigurationsprozess erfolgte beim Industriepartner mit einem Konfigurator. Die dabei erzeugten Daten wurden in einem Auftragseingangssystem gespeichert und im Laufe der Auftragsabwicklung mit Produkteigenschaften angereichert. Vom Auftragseingangssystem wurden tägliche Kopien der Daten in einem Datenbanksystem gespeichert. Daraus wurden die Daten für die Analyse mit verschiedenen SQL-Abfragen exportiert. Aufgrund der neuen Produktgeneration und dem damit verbundenen neuen Produktdatenmodell wurde lediglich eine Auswahl von Fahrzeugen betrachtet, welche zwischen April 2020 und März 2022 konfiguriert wurden. Vom Industriepartner wurde ein Datensatz mit 189 802 Fahrzeugkonfigurationen bereitgestellt. Für die einzelnen Produkteigenschaften standen weniger Daten zur Verfügung. Dies kam daher, dass die Fahrzeuge zwar konfiguriert und verkauft, aber teilweise noch nicht gebaut wurden

und dadurch keine Informationen über z. B. die Fahrzeuggewichte vorhanden waren. Außerdem war die Datenzugänglichkeit bei den Preisen der Fahrzeuge eingeschränkt.

Die Nutzungsdaten wurden entweder beim Service der Fahrzeuge in einer der Niederlassungen des Industriepartners ausgelesen oder über eine Telematikeinheit des Fahrzeugs kontinuierlich übertragen. Gespeichert wurden die Daten anschließend in einer Datenbank, welche als „Data Lake“ bezeichnet wird. Mit einem Tool zur Betriebsdatenanalyse fand eine erste Vorbereitung der unstrukturierten Sensordaten statt. Dabei wurden diese als statistische Daten aufbereitet und geben Auskunft über das Nutzungsverhalten des Fahrzeugs. Aufgrund des neuen Produktdatenmodells und der daraus resultierenden Beschränkung auf die Daten der Produktvarianten nach April 2020 waren nur wenige Nutzungsdaten verfügbar.

6.4 Baustein 3: Systematische Generierung und Einsatz von Wissen

Im Folgenden wird auf die Implementierung der Machine Learning Anwendungsfälle mit den realen Daten des Industriepartners eingegangen. Diese erfolgte mit der Programmiersprache Python in der Entwicklungsumgebung PyCharm. Dabei wurden unterschiedliche Bibliotheken für die Datenvorbereitung sowie das Training und die Evaluation der Machine Learning Modelle herangezogen. Eine Übersicht der verwendeten Bibliotheken kann Tabelle 6-2 entnommen werden.

Tabelle 6-2: Übersicht der verwendeten Python Bibliotheken

Bibliothek	Funktionalität
NumPy	• Effiziente Handhabung von Arrays • Vektoren- und Matrizenberechnungen
Pandas	• Effiziente Handhabung von DataFrames • Bearbeiten von Daten in tabellarischer Form
SciPy	• Durchführung mathematischer Berechnungen • Lösen algebraischer Gleichungen
Matplotlib	• Werkzeuge für die Datenvisualisierung • Plotten von Funktionen
Prince	• Funktionen für Dimensionsreduktion • Implementierung von CA und MCA
Scikit-learn	• Funktionen für Machine Learning Algorithmen und Datenvorverarbeitung • Implementierung von Machine Learning Algorithmen
Mlxtend	• Machine Learning Extensions • Funktionen für Data Science Anwendungen
Keras	• High-Level-API für TensorFlow 2 • Implementierung neuronaler Netze

6.4.1 Marktspezifische und technische Produkteigenschaften

Im Folgenden werden zuerst die Ergebnisse der Anwendung der Regressionsanalyse zur Generierung von Wissen über kontinuierliche marktspezifische und technische Produkteigenschaften vorgestellt (siehe Mehlstäubl et al. 2022a). Anschließend wird auf die kategorischen Merkmale, die mit einer Klassifikationsanalyse generiert wurden, eingegangen. Dabei wurden die kontinuierlichen Größen der Fahrzeuggewichte, CO_2-Emission und Preise bzw. Zahlungsbereitschaft der Kunden berücksichtigt sowie die kategorische Größe der Vertriebsländer.

6.4.1.1 Prognose der Fahrzeuggewichte, CO_2-Werte und Zahlungsbereitschaft mit einer Regression

In diesem Unterkapitel werden die Ergebnisse des Bausteins 3.0 Datenvorbereitung und 3.1 Regressionsanalyse am Beispiel der Prognose der Fahrzeuggewichte, der CO_2-Emissionen und der Zahlungsbereitschaft vorgestellt.

6.4.1.1.1 Datenvorbereitung

Für die Auswahl der Verfahren zur Datenvorbereitung wurde die Auswahltabelle aus Kapitel 5.4.1 herangezogen (siehe Tabelle 6-3). Für diesen Anwendungsfall wurde eine Regressionsanalyse auf Vertriebsdaten angewendet. Die erforderlichen Verfahren zur Datenbereinigung sind das Entfernen der konstanten Merkmale und die Bereinigung der fehlenden Werte. Für die Transformation wird aufgrund der nominalen Ausprägungen der Merkmale, welche nicht alle in eine natürliche Reihenfolge gebracht werden konnten, eine One-hot Kodierung verwendet. Im Folgenden werden die Schritte zur Datenvorbereitung für die einzelnen Produkteigenschaften differenziert erläutert. Aufgrund der Geheimhaltung können keine genauen Aussagen über fehlende Werte gemacht werden.

Tabelle 6-3: Schritte zur Datenvorbereitung für die Prognose von Produkteigenschaften mit einer Regression

	Konstante Merkmale	Fehlende Werte	Encoding	Skalierung	Dimensionsreduktion
Vertriebsdaten	x	o	x	-	o
Nutzungsdaten	-	o	-	x	o
Regressionsanalyse	x	x	o	o	-
Klassifikationsanalyse	x	x	o	o	-
Clusteranalyse	x	x	o	o	x
Assoziationsanalyse	x	-	o	o	-

x = erforderlich, - nicht erforderlich, o = keine Abhängigkeit

Fahrzeuggewichte

Für die Prognose der Fahrzeuggewichte standen 73 202 Konfigurationen von verkauften und gebauten Produktvarianten zur Verfügung. Jede dieser Konfigurationen bestand vor der Datenvorbereitung aus 1 106 Merkmalen. Zuerst wurden die Duplikate der Produktvarianten in den Daten entfernt. Duplikate sind in diesem Fall Produktvarianten mit den gleichen Merkmalsausprägungen und identischen Gewichten. Duplikate entstehen, da Kunden oft mehrere Fahrzeuge mit derselben Konfiguration kaufen. Diese wurden entfernt, da sie zum einen keine zusätzlichen Erkenntnisse bieten und zum anderen die Evaluationsergebnisse verfälschen. Ohne Duplikate beinhaltete der Datensatz noch 58 776 Produktvarianten. Anschließend wurden alle konstanten Merkmale, d. h. Merkmale mit lediglich einer Merkmalsausprägung in den Daten, entfernt. Dies führte zu einer Reduktion auf 960 Merkmale. Danach wurden Merkmale und Produktvarianten entfernt, welche eine Vielzahl an fehlenden Werten besitzen. Diese resultieren daraus, dass das Produktdatenmodell sich im ständigen Wandel befindet und neue Merkmale hinzugefügt und entfernt werden. Dadurch wurde der Datensatz auf 58 642 Produktvarianten und 893 Merkmale reduziert. Durch die One-hot Kodierung wurden die 893 Merkmale in eine Matrix mit 10 890 Spalten transformiert.

CO_2-Emissionen

Das gleiche Verfahren wurde bei der Datenvorbereitung für die Vorhersage der CO_2-Werte verfolgt. Vom Industriepartner wurden initial 50 763 Fahrzeugkonfigurationen aus 1 106 Merkmalen sowie den zugehörigen CO_2-Werten zur Verfügung gestellt. Diese wurden durch Entfernen der Duplikate auf 39 908 Fahrzeugkonfigurationen reduziert. Anschließend wurden die Merkmale und Konfigurationen mit zu vielen fehlenden Werten entfernt und der Datensatz auf 39 906 Produktvarianten mit 840 Merkmalen verringert. Durch die One-hot Kodierung wurden die Merkmale in die 8 618 Merkmalsausprägungen überführt.

Zahlungsbereitschaft

Für die Bestimmung der Zahlungsbereitschaft der Kunden standen die Preise von insgesamt 40 257 Konfigurationen mit 1 109 Merkmalen zur Verfügung. Neben den Merkmalen wurden als Eingangsgrößen noch das Baujahr und eine Kategorisierung der Kunden hinzugefügt. Durch das Entfernen der Duplikate und konstanten Merkmale wurde der Datensatz auf 22 591 Fahrzeuge und 1 025 Merkmale verringert. Eine anschließende Bereinigung der fehlenden Werte führte zu 22 577 Produktkonfigurationen mit 886 Merkmalen. Durch die One-hot Kodierung wurde eine Matrix mit 9 724 Spalten erzeugt.

6.4.1.1.2 Regression

Nach der Datenvorbereitung fand die Bildung der Regressionsmodelle statt. Hierfür wurden die Regressionsalgorithmen, welche zuvor im Framework beschrieben und in Tabelle 5-6 gegenübergestellt wurden, implementiert. Da der Datensatz aus einer

großen Anzahl an Fahrzeugkonfigurationen mit einer Vielzahl an Merkmalen und Merkmalsausprägung bestand und zudem eine hohe Nichtlinearität zwischen den Merkmalsausprägungen und der Produkteigenschaft zu erwarten war, sind die Algorithmen neuronales Netz, Random Forest und Entscheidungsbaum in diesem Fall geeignet. Es wurden zusätzlich die lineare Regression, die Support Vector Machine und der k-Nearest Neighbors berücksichtigt, um die zuvor beschriebenen Faktoren für die Auswahl der Algorithmen zu belegen. Als Verlustfunktion wurde der quadratische Fehler verwendet und die Aufteilung zwischen Trainings- und Testdaten wurde aufgrund der hohen Anzahl an Datenpunkten mit 90 % zu 10 % gewählt.

6.4.1.1.3 Evaluation

Fahrzeuggewichte

Für die Evaluation der Modelle wurden die Kriterien MSE, MAE, MAPE und R^2 herangezogen (siehe Kapitel 5.4.2.2). Die Ergebnisse der Anwendung der Bewertungskriterien auf die Vorhersagen der Testdaten sind in Tabelle 6-4 dargestellt. Die lineare Regression und Support Vector Machine lieferten erwartungsgemäß ungeeignete Vorhersagen, da diese die nicht-linearen Beziehungen zwischen den verschiedenen Kombinationen der Merkmalsausprägungen und den Fahrzeuggewichten nicht abbilden konnten. Die anderen vier Algorithmen lieferten vielversprechende Ergebnisse. Vor allem das neuronale Netz erzielte mit einem MAE von 46 kg und einer Vorhersagezeit von 2,25 x 10^{-4} s die besten Ergebnisse. Die Daten umfassten Fahrzeuge von unter 4 000 kg bis zu über 14 000 kg. 46 kg entspricht einem MAPE von nur 0,63 %.

Tabelle 6-4: Evaluationsergebnisse für die Prognose der Fahrzeuggewichte

	Lineare Regression	**Support Vector Machine**	**K-Nearest Neighbors**	**Entscheidungsbaum**	**Random Forest**	**Neuronales Netz**
MSE [kg^2]	4,63 x 10^{23}	1080779	45262	26587	23245	15657
MAE [kg]	3,53 x 10^{10}	641	95	70	59	46
MAPE [%]	4,35 x 10^{8}	9,94	1,29	0,97	0,81	0,63
R^2	7,52 x 10^{-4}	0,869	0,984	0,991	0,995	0,994
Trainingszeit [s]	500	17588	0,95 s	81 s	373 s	222 s
Vorhersagezeit [s]	1,24 x 10^{-4}	0,65	1,13 x 10^{-2}	3,62 x 10^{-5}	4,34 x 10^{-5}	2,52 x 10^{-4}

Bisher wurden die Gewichte auf der Grundlage von Expertenregeln berechnet, die von einem Team von Entwicklern über mehrere Monate hinweg formuliert wurden. Mit dem regelbasierten Expertensystem erreichte das Unternehmen eine Genauigkeit von etwa 97 %, und die Berechnungszeit lag in der realen Anwendung zwischen einer und

zwei Sekunden. Durch den Einsatz von Machine Learning wurden die Regeln automatisiert erzeugt. Es wurde eine Genauigkeit von 99,37 % erzielt, wodurch der Fehler bei den Vorhersagen um knapp 80 % reduziert wurde. Darüber hinaus ist die Vorhersagezeit der Machine Learning Modelle um ein Vielfaches geringer (2,52 x 10^{-4} s vs. 1-2 s).

In Abbildung 6-2 sind die Prognosen des neuronalen Netzes visualisiert. Die meisten Vorhersagen des Modells sind im Einklang mit den tatsächlichen Messwerten in den Daten. Allerdings existieren ebenfalls einige Ausreißer. Insbesondere das Fahrzeug mit der ID 6060XXXX4 hatte ein vorhergesagtes Gewicht von 11 971 kg und einen gewogenen Wert von 8 105 kg, was einer Abweichung von 3 866 kg entsprach. Bei dem Fahrzeug handelte es sich um einen Schwerlastkraftwagen mit vier angetriebenen Achsen, welcher zu den schwersten Fahrzeugen im Produktportfolio des Industriepartners zählt. Außerdem konnte eine identische Konfiguration identifiziert werden, die vom gleichen Kunden mit einer späteren ID bestellt wurde und ein tatsächliches Gewicht von 11 995 kg aufwies. Daraus ließ sich schließen, dass es sich hierbei nicht um eine fehlerhafte Vorhersage, sondern um einen Fehler in den gewogenen Daten handelt.

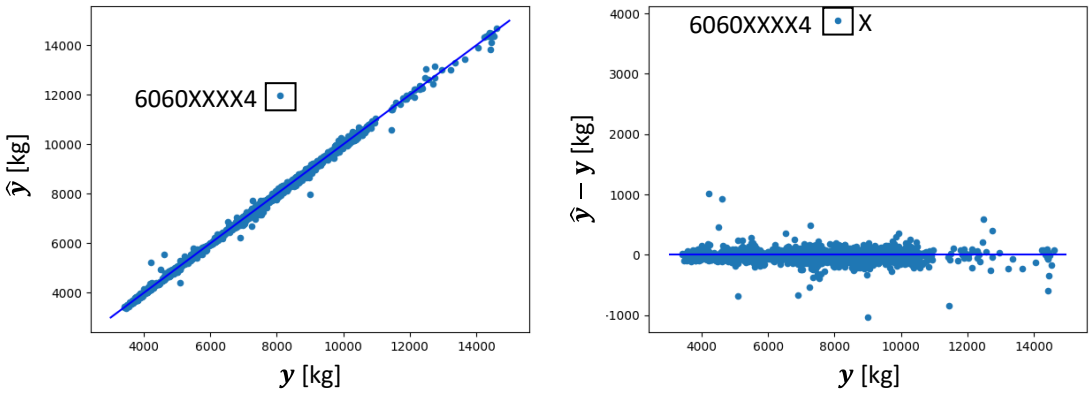


Abbildung 6-2: Visualisierung der mit einem neuronalen Netz prognostizierten Fahrzeuggewichte

CO_2-Emissionen

Die Evaluationsergebnisse der einzelnen Regressionsalgorithmen für die Vorhersage der CO_2-Emission lieferte ein vergleichbares Bild wie die der Fahrzeuggewichte. Eine Übersicht wird in Tabelle 6-5 gegeben. Wie schon zuvor bei den Gewichten schnitten die lineare Regression und die Support Vector Machine am schlechtesten ab. Die komplexeren Algorithmen erzielten wie erwartet besser Ergebnisse. Die präzisesten Ergebnisse erbrachte der Random Forest Algorithmus mit einem MAE von 4,8 g CO_2 / km und einem prozentualen Fehler von lediglich 0,65 %. Im Vergleich zu den aktuellen Simulationen der CO_2-Emissionen beim Industriepartner bietet das Machine Learning Modell speziell einen Vorteil in der Vorhersagezeit. Während die rechenaufwendigen

Simulationen im Zeitraum von einer Stunde lagen, beträgt die Vorhersagezeit mit dem Random Forest Modell 3,5 x 10^{-5} s.

Tabelle 6-5: Evaluationsergebnisse für die Prognose der CO_2-Werte

	Lineare Regression	Support Vector Machine	K-Nearest Neighbors	Entscheidungsbaum	Random Forest	Neuronales Netz
MSE [(g/km)^2]	8,24 x 10^{21}	2502	711	209	153	302
MAE [g/km]	5,2 x 10^{9}	26,3	12,2	5,4	4,8	11,0
MAPE [%]	6,75 x 10^{8}	3,49	1,63	0,72	0,65	1,49
R^2	1,82 x 10^{-3}	0,764	0,925	0,977	0,983	0,973
Trainingszeit [s]	246,86	7737,3	0,37	48,06	268,3	305,62
Vorhersagezeit [s]	2,05 x 10^{-5}	0,57	7,85 x 10^{-3}	2,75 x 10^{-5}	3,5 x 10^{-5}	2,43 x 10^{-4}

In Abbildung 6-3 werden die Vorhersageergebnisse mit den tatsächlichen Werten in den Daten gegenübergestellt. Die meisten Prognosen stimmten gut mit den Werten in den Daten überein. Es waren jedoch auch einige Ausreißer zu erkennen.

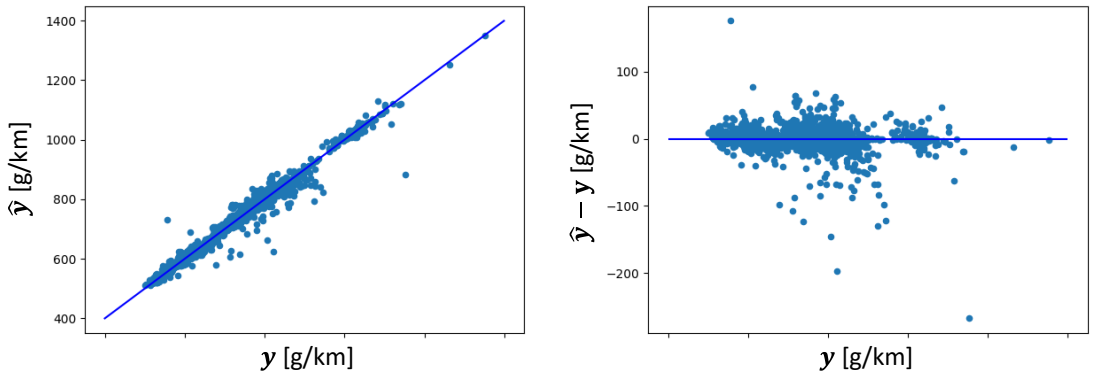


Abbildung 6-3: Visualisierung der mit dem Random Forest Algorithmus prognostizierten CO_2-Werte

Die Ausreißer wurden anschließend näher betrachtet und mit Experten der CO_2-Berechnung evaluiert. Es wurde festgestellt, dass es sich um Fahrzeuge mit besonders hohem Modifikationsanteil handelte. Das bedeutet, dass die Fahrzeuge nach dem Bau noch nach bestimmten Kundenwünschen individuell angepasst wurden und die Anpassungen nicht vollumfänglich in den Daten abgebildet wurde. Bisher wurden die CO_2-Werte simuliert. Hierfür waren jedoch zum einen die konkreten Sachnummern, welche im Fahrzeug verbaut wurden, erforderlich und zum anderen dauert eine solche

Simulation bis zu einer Stunde. Mit dem Machine Learning Modell konnte die Zeit auf $3{,}5 \times 10^{-5}$ s reduziert werden.

Zahlungsbereitschaft

Neben technischen Produkteigenschaften wie den Gewichten und CO_2-Werten können auch nicht-technische Produkteigenschaften mit einer Regression bestimmt werden. Im Folgenden fand eine Anwendung am Beispiel der Zahlungsbereitschaft statt. Beim Industriepartner wurde der Preis einer Produktvariante in Abhängigkeit von den gewählten Merkmalen, dem Vertriebsland und dem Kunden kalkuliert. Es wurde ein Listenpreis auf der Grundlage der Kundenmerkmale festgelegt. Je nach Vertriebsland wurde auf einen bestimmten Prozentsatz verzichtet und jeder Verkäufer konnte dem jeweiligen Kunden zusätzlich einen Rabatt gewähren. Daher war es mit einer Kalkulation nicht möglich, genau zu bestimmen, was jeder Kunde für einzelne Merkmalsausprägungen und dadurch für eine Produktvariante bereit ist zu zahlen.

Die Algorithmen verhielten sich bei der Zahlungsbereitschaft ähnlich wie bei der Vorhersage der Gewichte und CO_2-Werte (siehe Tabelle 6-6). Die exaktesten Vorhersagen lieferte der Random Forest Algorithmus mit einem R^2 von 0,909 und einem MAPE von 3,78 %. Aus Gründen der Geheimhaltung können hier keine genauen Werte für die Preise genannt werden. Aufgrund der geringen Zahl an Datenpunkten (22 577) im Vergleich zu den verschiedenen Merkmalsausprägungen (9 724) waren die Prognosen im Vergleich zu denen der Fahrzeuggewichte und CO_2-Werte etwas ungenauer. Hinzu kommt, dass es sich bei der Zahlungsbereitschaft nicht um einen absoluten Wert handelt, sondern dieser abhängig von dem jeweiligen Kunden ist. So kann die Zahlungsbereitschaft für eine identische Fahrzeugkonfiguration in Abhängigkeit des Kunden unterschiedliche Ausprägungen annehmen.

In Abbildung 6-4 sind die Vorhersageergebnisse des Random Forest Algorithmus visualisiert. Es ist ersichtlich, dass das Modell Zusammenhänge zwischen den Merkmalsausprägungen und der Zahlungsbereitschaft erkennen konnte. Die Ausreiser wurden mit Experten aus der Preisgestaltung des Industriepartners analysiert. Diese resultierten vor allem aus den Fahrzeugen, welche einen speziellen Aufbau von einem externen Aufbauhersteller erhalten, welcher so nicht in den Daten abgebildet war.

Tabelle 6-6: Evaluationsergebnisse für die Prognose der Zahlungsbereitschaft

	Lineare Regression	Support Vector Machine	K-Nearest Neighbors	Entscheidungsbaum	Random Forest	Neuronales Netz
MAPE [%]	$5{,}40 \times 10^{9}$ %	16,23 %	4,71 %	4,82 %	3,78 %	4,46 %
R^2	$3{,}356 \times 10^{-3}$	0,670	0,869	0,835	0,909	0,901
Trainingszeit [s]	302,32	1923 s	0,25 s	24,62 s	346,08 s	231,87 s
Vorhersagezeit [s]	$2{,}12 \times 10^{-5}$	0,23 s	$3{,}71 \times 10^{-3}$	$3{,}22 \times 10^{-5}$	$5{,}21 \times 10^{-5}$	$2{,}67 \times 10^{-4}$

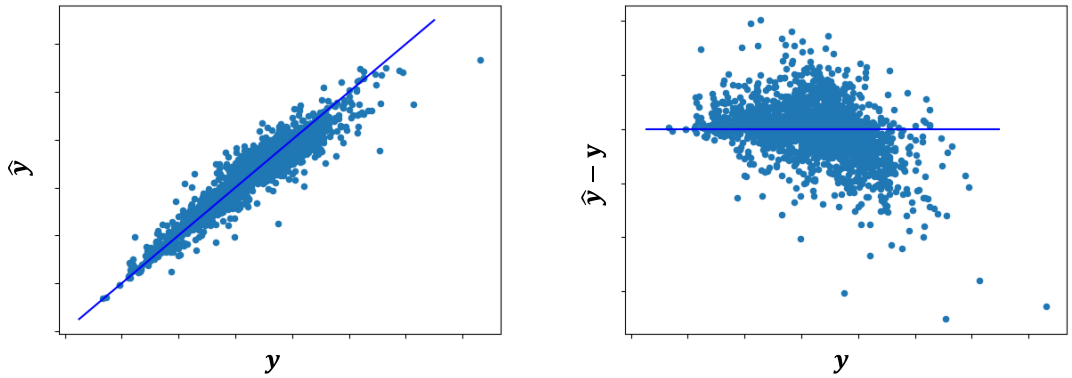


Abbildung 6-4: Visualisierung der mit dem Random Forest prognostizierten Zahlungsbereitschaft

6.4.1.1.4 Einsatz

Die erzeugten Regressionsmodelle wurden auf unterschiedliche Weiße eingesetzt, um die Analyse komplexer Produktportfolios beim Industriepartner zu unterstützen. Auf der einen Seite wurden sie für die Vorhersage von technischen und marktspezifischen Produkteigenschaften neuer, noch nicht gebauter und verkaufter Konfigurationen von Produktvarianten eingesetzt. Dadurch war es möglich, eine Optimierung der Produkteigenschaften durchzuführen. Auf der anderen Seite wurden die Modelle und deren Verhalten näher analysiert und Rückschlüsse auf einzelne Merkmale und Merkmalsausprägungen gezogen. Im Folgenden wird auf die Eigenschaftsvorhersage, Eigenschaftsoptimierung sowie die Analyse der Modelle beim Industriepartner eingegangen.

Eigenschaftsvorhersage und -optimierung am Beispiel der Fahrzeuggewichte und CO_2-Emissionen

Im Rahmen der Anwendungsvalidierung beim Industriepartner wurde ein Demonstrator in Form eines Softwaretools aufgebaut. Hierfür wurde mit PTC ThingWorx eine Eingabeschnittstelle definiert, welche auf die mit Python erzeugten Modelle zugreift. Der Demonstrator ermöglichte die Nutzung der Machine Learning Modelle durch die Entwickler und dadurch deren Einbindung ins Tagesgeschäft des Industriepartners. In Abbildung 6-5 ist die zugrundeliegende Softwarearchitektur des Demonstrators abgebildet. Bestandteile sind das ThingWorx Modul, welches die Eingabeschnittstelle beinhaltet, und ein Python Modul, welches das Training und den Einsatz der Modelle enthält. Das Python Modul besitzt eine Schnittstelle zum Datenbanksystem des Industriepartners, wodurch Trainings- und Testdaten abgerufen werden. Das ThingWorx Modul hat eine Schnittstelle zum Produktkonfigurator, wodurch die Konfigurationen übermittelt werden können. Zudem hat es eine Schnittstelle zum Produktdatenmodell, um die Konfigurationen, welche sich durch die Definition der Freiheitsgrade für die Optimierung ergeben, zu ermitteln. Die Konfigurationen werden von der Eingabeschnittstelle an das Python Modul übergeben, welches die Vorhersagen berechnet und die Ergebnisse zurück an das ThingWorx Modul gibt.

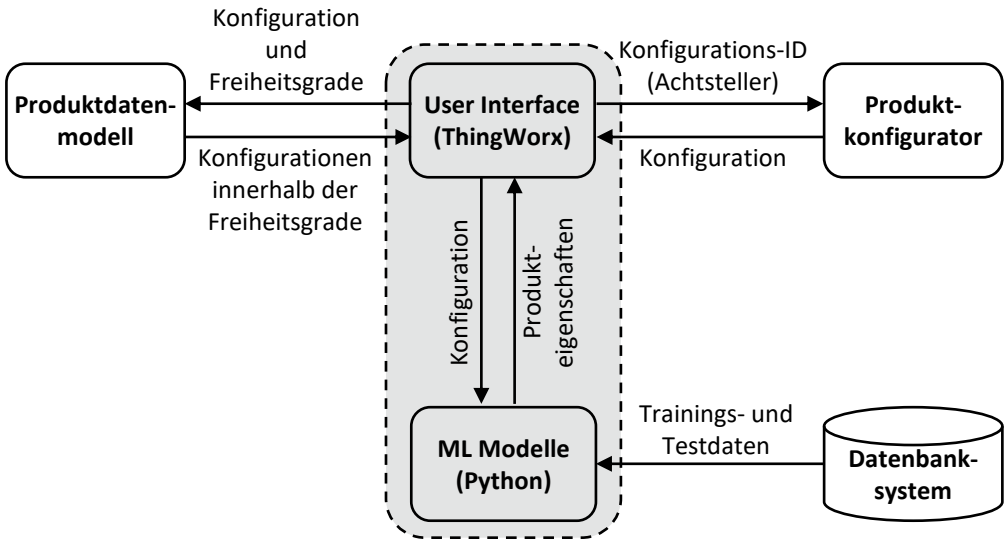


Abbildung 6-5: Architektur des Demonstrators

Die Eingabeschnittstelle zur Vorhersage von Produkteigenschaften beim Industriepartner ist in Abbildung 6-6 dargestellt. Diese ermöglicht dem Nutzer eine Konfiguration einer Produktvariante in Form eines JSON-Files, welches zuvor im Produktkonfigurator erstellt und exportiert wurde, hochzuladen oder direkt Konfigurationen aus dem Produktkonfigurator mit einer sogenannten Achtsteller ID zu laden. Für diese werden dann im Demonstrator die Produkteigenschaften bestimmt. Der Demonstrator ermöglicht es, eine Vielzahl an Konfigurationen parallel hochzuladen und in Echtzeit zu berechnen sowie deren Ergebnisse in einem XLSX-File zu exportieren.

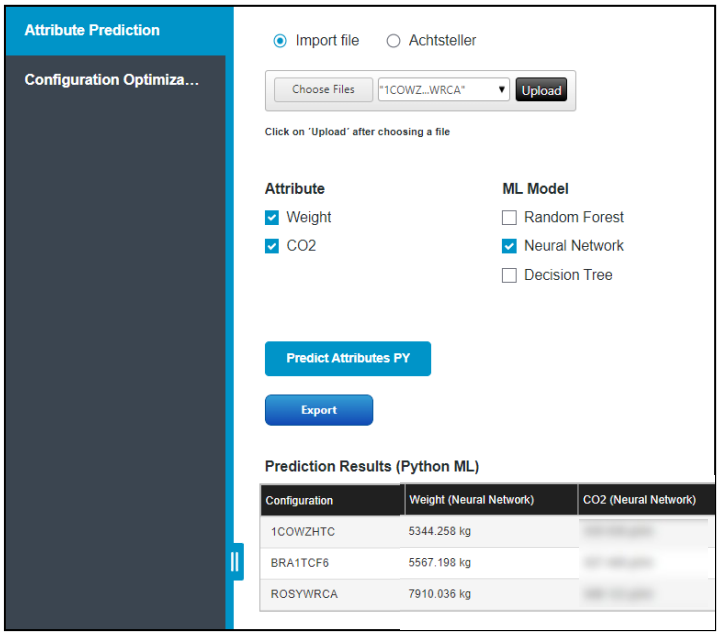


Abbildung 6-6: Einsatz der Regressionsmodelle zur Eigenschaftsvorhersage

Für die Berechnung kann zwischen unterschiedlichen Modellen (Entscheidungsbaum, Random Forest oder neuronalen Netz) und Produkteigenschaften (Fahrzeuggewichte oder CO_2-Emissionen) ausgewählt werden. Aus Gründen der Geheimhaltung wurden die CO_2-Emissionen im Folgenden unkenntlich gemacht.

Die Modelle wurden ebenfalls für eine Optimierung der Eigenschaften der Produktkonfigurationen eingesetzt. Diese Funktionalität wurde ebenfalls im Softwaredemonstrator beim Industriepartner implementiert und ist in Abbildung 6-7 dargestellt. Die Eingabeschnittstelle erfordert zum einen eine Fahrzeugkonfiguration und zum anderen die für die Optimierung zur Verfügung stehenden Freiheitsgrade. Durch eine Schnittstelle zum Produktdatenmodell werden die sich dadurch ergebenden neuen Konfigurationen ausgegeben und mit dem Machine Learning Modell berechnet. Anschließend kann der optimale Wert in den Produkteigenschaften ermittelt werden.

Eine Eigenschaftsoptimierung hinsichtlich der CO_2-Emission wurde am Beispiel der Konfiguration „ROSYWRCA“, welches einem Fahrzeug mit der Fahrgestellart „Sattel“ und der Radformel „6x2/2“ entspricht, durchgeführt (Tabelle 6-7). Als Freiheitsgrade für die Optimierung wurden die Merkmale „0KK8 – Dachspoiler“, „0KKJ – Sideflaps“ und „0KQM – Lackierung Dachspoiler/Aeropaket“ definiert. Die Ausgangskonfiguration hatte in diesen Merkmalen die Ausprägungen „0P1X7 – Dachspoiler“, „0P1Y9 - Ohne Sideflaps“ und „0P6X3 - Decklackierung Dachspoiler“. Ausgehend von dieser Konfiguration wurde in Tabelle 6-7 die prozentuale Differenz der CO_2-Emissionen abgebildet. Es ist zu erkennen, dass die CO_2-Emissionen der Konfiguration durch das Hinzufügen von Sideflaps verbessert werden kann. Der geringste CO_2-Wert wurde mit „Sideflaps, klappbar rechts und feststehend links“ und mit „Dachspoiler“ erreicht. Dies entsprach einer Reduktion der CO_2-Emission um 2,51 %. Jedoch wird das Gewicht um 3,7 kg erhöht.

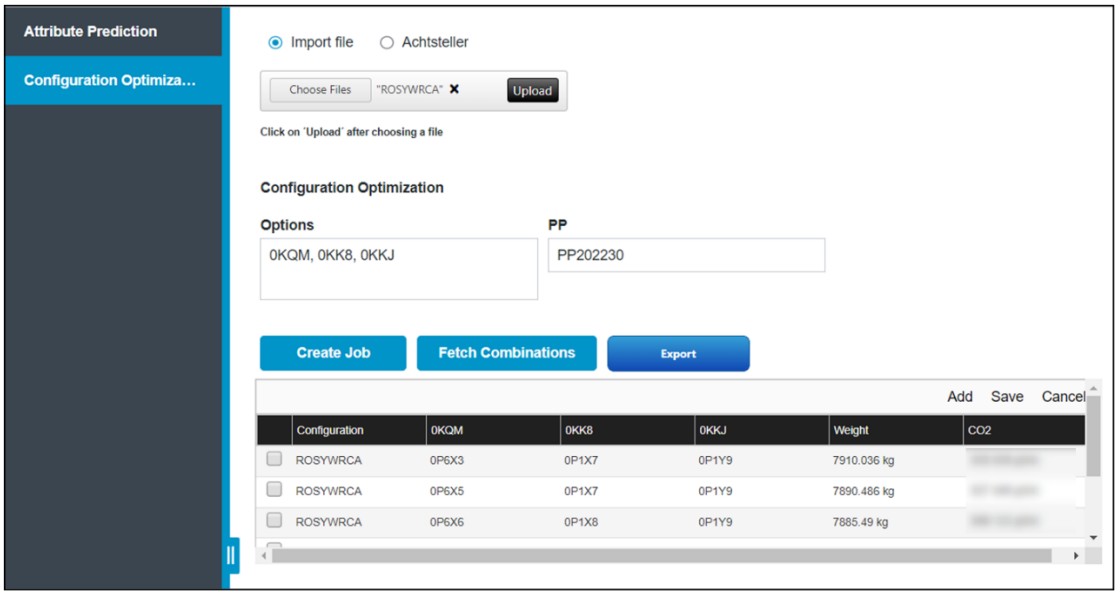


Abbildung 6-7: Einsatz der Regressionsmodelle zur Eigenschaftsoptimierung

Tabelle 6-7: Ergebnis Eigenschaftsoptimierung

Dachspoiler	Sideflaps	Lackierung Dachspoiler	Gewichte	CO_2
Dachspoiler	Sideflaps, klappbar rechts und feststehend links	Strukturlackierung Aero-Paket	7913,738 kg	-2,51 %
Dachspoiler	Sideflaps, klappbar rechts und links	Strukturlackierung Aero-Paket	7921,762 kg	-2,48 %
Dachspoiler	Sideflaps, klappbar rechts und feststehend links	Decklackierung Aero-Paket	7937,035 kg	-2,31 %
Dachspoiler	Sideflaps, klappbar rechts und links	Decklackierung Aero-Paket	7945,059 kg	-2,28 %
Dachspoiler	Ohne Sideflaps	Strukturlackierung Dachspoiler	7890,486 kg	-0,12 %
Dachspoiler	Ohne Sideflaps	Decklackierung Dachspoiler	7910,036 kg	0,00 %
Ohne Dachspoiler	Ohne Sideflaps	Ohne Lackierung Dachspoiler/ Aero-Paket	7885,49 kg	+1,73 %

Analyse der Modelle am Beispiel der Zahlungsbereitschaft

Die Vorhersage von Produkteigenschaften wurde vom Industriepartner genutzt, um durch die Analyse der Modelle Rückschlüsse auf die Bedeutung einzelner Merkmalsausprägungen zu ziehen. Zur Demonstration wurde die Zahlungsbereitschaft für die Testdaten beispielhaft einmal mit der Merkmalsausprägung „ohne Steckdose 230 V“ und einmal mit der Merkmalsausprägung „mit Steckdose 230 V“ sowohl für das neuronale Netz als auch für den Random Forest Algorithmus berechnet. Die Zahlungsbereitschaft für die Konfigurationen „mit Steckdose 230 V“ war mit dem Random Forest Algorithmus im Durchschnitt um 0,071 % höher und mit dem neuronalen Netz um 0,075 % höher.

Für die Bestimmung der Relevanz einzelner Merkmale und Merkmalsausprägungen für die Zahlungsbereitschaft wurde die Feature Importance aus dem Random Forest Modell abgeleitet. Je höher die Punktzahl der Feature Importance ist, desto größer ist der Einfluss eines Merkmals auf die Zahlungsbereitschaft. Neben den Merkmalen mit großen Auswirkungen konnten auch Merkmale und Merkmalsausprägungen mit geringem oder keinem Einfluss auf die Zahlungsbereitschaft identifiziert werden. Im Hinblick auf die Zahlungsbereitschaft wurden vor allem jene Merkmale mit einer Wichtigkeit von Null kritisch analysiert, da sie aus Kundensicht keinen Mehrwert bieten. Die Feature Importance selbst gibt jedoch keine Auskunft darüber, ob das Vorhandensein einer Merkmalsausprägung die Zahlungsbereitschaft positiv oder negativ beeinflusst. Tabelle 6-8 zeigt ein Beispiel für die Feature Importance des Merkmals „Steckdose 230 V“ sowie dessen Ausprägungen. Die Summer der Feature Importance aller Merkmale ist gleich eins.

Tabelle 6-8: Feature Importance am Beispiel der Steckdose mit 230 V

Merkmal	Merkmalsausprägung	Feature Importance
Steckdose 230V (Importance: 3,32204 x 10^{-4})	Ohne Steckdose 230 V	1,51061 x 10^{-4}
	Mit Steckdose 230 V	1,81143 x 10^{-4}

6.4.1.1.5 Schlussfolgerung

Der Anwendungsfall demonstrierte die Vorhersage von kontinuierlichen Produkteigenschaften auf Basis von Konfigurationen aus Merkmalsausprägungen mit Machine Learning. Machine Learning ermöglichte in dieser Anwendung die Vorhersage der Produkteigenschaften schon während des Konfigurationsprozesses in Echtzeit und reduzierte den Aufwand im Vergleich zur Definition regelbasierter Expertensysteme auf ein Minimum. Darüber hinaus konnten mit den trainierten Modellen bei der Bestimmung von Fahrzeuggewichten sowohl eine höhere Vorhersagegenauigkeit als auch eine Vorhersagegeschwindigkeit erzielt werden, ohne Kenntnisse über die verbauten Komponentenvarianten und Sachnummern zu besitzen. Die Vorhersagegeschwindigkeit brachte vor allem im Vergleich zu den Simulationen der CO_2-Emissionen einen entscheidenden Vorteil und ermöglichte dort eine Eigenschaftsoptimierung.

6.4.1.2 Prognose der Vertriebsländer mit einer Klassifikation

Neben der Vorhersage von kontinuierlichen Werten können mit Machine Learning auch kategorische Produkteigenschaften mit einer Klassifikation vorhergesagt werden. Die Datenvorbereitung verhält sich gleich wie bei den kontinuierlichen Produkteigenschaften. Lediglich der Einsatz der Algorithmen während der Modellierung sowie die statistischen Kriterien zur Evaluation sind unterschiedlich. Als Anwendungsbeispiel wurden beim Industriepartner die Vertriebsländer als Produkteigenschaften ausgewählt.

6.4.1.2.1 Datenvorbereitung

Als Datenbasis standen für die Vorhersage der Länder 189 773 Fahrzeugkonfigurationen und 1 105 Merkmale zur Verfügung. Das Produktdatenmodell beim Industriepartner enthielt einige organisatorische Merkmale wie zum Beispiel das Zielland. Dieses wurde für die Klassifikation als Zielgröße gewählt. Daneben gibt es organisatorische Merkmale wie zum Beispiel die „Sprache der Bedienungsanleitung“ oder die „Displaysprache“, die wenig mit den Funktionalitäten des Fahrzeugs zu tun haben, aber indirekt Aufschluss über das Zielland geben. Diese wurden in einem ersten Schritt entfernt, sodass 1 100 Merkmale übrig blieben. Für die Datenvorbereitung wurden die gleichen Schritte durchgeführt wie bei der Regressionsanalyse zuvor (siehe Tabelle 6-9). Durch das Entfernen der Duplikate und konstanten Merkmale wurde der Datensatz auf 76 563 Fahrzeugkonfigurationen und 980 Merkmale reduziert. Durch die Beseitigung der fehlenden Werte wurde der Datensatz weiter auf 76 265 Fahrzeuge und 879 Merkmale verringert.

Tabelle 6-9: Schritte zur Datenvorbereitung für die Prognose von Produkteigenschaften mit einer Klassifikation

	Konstante Merkmale	Fehlende Werte	Encoding	Skalierung	Dimensions-reduktion
Vertriebsdaten	x	x	x	-	o
Nutzungsdaten	-	x	-	x	o
Regressionsanalyse	x	x	o	o	-
Klassifikationsanalyse	x	x	o	o	-
Clusteranalyse	x	x	o	o	x
Assoziationsanalyse	x	-	o	o	-

x = erforderlich, - nicht erforderlich, o = keine Abhängigkeit

Zusätzlich wurde die statistische Verteilung der Produktvarianten auf die Klassen bzw. Zielländer untersucht. Die Fahrzeugkonfigurationen in den Datenbeispielen wurden in insgesamt 103 Ländern verkauft. Dabei waren neun Länder enthalten, in die lediglich eine Produktvariante verkauft wurde. Daran lässt sich zum einen das große Ungleichgewicht zwischen den Klassen erkennen und zum anderen, dass einige Klassen zu selten vorkommen, um statistische Aussagen treffen zu können. Aus diesem Grund wurden alle Länderklassen, welche weniger als 20-mal in den Daten vorkommen nicht in der Klassifikation betrachtet. Dadurch wurde die Anzahl an Länder auf 68 und die Produktvarianten auf 75 984 reduziert. Durch die anschließende One-hot Kodierung wurden 12 357 Spalten erzeugt.

6.4.1.2.2 Klassifikation

Für die Modellbildung wurden auf Basis des Vergleichs aus Kapitel 5.4.3.1 die Algorithmen k-Nearest Neighbor, Entscheidungsbaum, Random Forest und neuronales Netz implementiert. Die logistische Regression und Support Vektor Machine wurden aufgrund fehlender Eignung für hochdimensionale Daten sowie nicht-lineare Zusammenhänge zwischen den Eingangsmerkmalen und der Zielvariablen im Folgenden nicht weiter berücksichtigt. Es handelt sich bei der Vorhersage der Vertriebsländer um eine Multi-Klassen-Klassifikation. Da eine Produktvariante in mehrere Länder verkauft werden kann, wurden die Zugehörigkeitswahrscheinlichkeiten mit den Modellen berechnet. Für die rein binären Klassifikationsmodelle wurde eine OvR-Strategie gewählt. Die Aufteilung in Trainings- und Testdaten wurde, wie bei der Regression, in einem Verhältnis von 90 % zu 10 % vorgenommen.

6.4.1.2.3 Evaluation

Für die Bewertung der Modelle wurden Accuracy, Precision, Recall und F1-Score anhand der Vorhersagen, für die den Modellen noch unbekannten Konfigurationen aus den Testdaten ermittelt. Da die voraussichtlichen Vertriebsländer relativ sind, wurde ebenfalls die Cross Entropie bestimmt. Aufgrund des großen Ungleichgewichts in der Zuteilung der Produktvarianten auf die einzelnen Länder bzw. Klassen wurde die ROC

AUC Kurve nicht betrachtet. In Tabelle 6-10 ist eine Übersicht der einzelnen Algorithmen mit den berechneten Metriken dargestellt. Das präziseste Ergebnis liefert das neuronale Netz aufgrund der mit Abstand geringsten Kreuzentropie (0,095) und höchsten Accuracy (0,984), Precision (0,912), Recall (0,889) und F1-Score (0,893). Aus diesem Grund wurden im Folgenden die Ergebnisse des neuronalen Netzes verwendet.

Tabelle 6-10: Evaluationsergebnisse der Klassifikationsmodelle für die Vertriebsländer

	K-Nearest Neighbors	**Entscheidungsbaum**	**Random Forest**	**Neuronales Netz**
Accuracy	0,915	0,937	0,943	0,984
Precision	0,740	0,810	0,886	0,912
Recall	0,664	0,693	0,768	0,889
F1-Score	0,684	0,725	0,811	0,893
Cross Entropy	0,639	0,770	0,482	0,095
Trainingszeit [s]	1,16	270,96	23,52	3286,57
Vorhersagezeit [s]	0,013	$1{,}52 \times 10^{-4}$	5.3×10^{-5}	$1{,}73 \times 10^{-4}$

In Tabelle 6-11 ist ein Ausschnitt aus der Konfusionsmatrix des neuronalen Netzes dargestellt. Diese ist nach der Häufigkeit der betrachteten Länder in den Testdaten geordnet. Die meisten Fahrzeuge wurden innerhalb Deutschlands und an die Nachbarländer verkauft.

Tabelle 6-11: Konfusionsmatrix der Vorhersagen des neuronalen Netzes

	DE	**FR**	**PL**	**AT**	**GB**	**ES**	**BE**	**IT**	**NL**	**CZ**	**RU**	**IL**	**…**
Deutschland (DE)	2799	0	2	2	0	4	1	0	2	0	0	0	…
Frankreich (FR)	3	636	0	0	0	0	1	0	0	0	0	0	…
Polen (PL)	0	0	437	0	0	0	0	0	0	0	0	0	…
Österreich (AT)	4	0	0	399	0	0	0	0	0	0	0	0	…
Vereinigtes Königreich (GB)	0	0	0	0	318	0	1	0	0	0	0	0	…
Spanien (ES)	2	0	0	0	0	313	0	0	0	0	0	0	…
Belgien (BE)	0	0	0	0	0	0	293	0	0	0	0	0	…
Italien (IT)	0	0	2	0	0	0	0	226	0	1	0	0	…
Niederlande (NL)	1	0	0	0	0	0	0	0	174	0	0	0	…
Tschechische Republik (CZ)	0	0	0	0	0	0	0	0	0	148	0	0	…
Russische Föderation (RU)	1	0	0	0	0	0	0	0	0	0	128	0	…
Israel (IL)	0	0	0	0	0	0	0	0	0	0	0	108	…
…	…	…	…	…	…	…	…	…	…	…	…	…	…

Es ist zu erkennen, dass das Modell in den meisten Fällen die richtigen Vertriebsländer prognostiziert, jedoch auch einige der Vorhersagen nicht mit den tatsächlichen Ländern übereinstimmen. Darüber hinaus wird das Ungleichgewicht in den Daten deutlich.

6.4.1.2.4 Einsatz

Die Klassifikationsmodelle können auf die gleiche Weise wie die Regressionsmodelle eingesetzt werden. In der Validierung wurden die Vertriebsländer für noch nicht verkaufte Fahrzeuge vorhergesagt und die Feature Importance bestimmt. In Tabelle 6-12 ist ein Ausschnitt der vorhergesagten Zugehörigkeitswahrscheinlichkeit für die Testdaten abgebildet. Dabei ist zu erkennen, dass der Algorithmus für die meisten Konfigurationen direkt ein Land zuordnen konnte. Für gewisse Konfigurationen bestand jedoch auch die Wahrscheinlichkeit, dass sie in unterschiedliche Länder verkauft werden.

Neben der Prognose der Vertriebsländer wurden beim Industriepartner die Modelle näher analysiert. In Tabelle 6-13 sind die Merkmale mit der höchsten und niedrigsten Feature Importance für die Vorhersage der Vertriebsländer abgebildet. Diese zeigen die vom Algorithmus am bedeutendsten sowie am unbedeutendsten eingestuften Merkmale. Die Summe der Feature Importance aller Merkmale ist gleich eins. Bei der Betrachtung der Feature Importance sind die Abhängigkeiten aus dem Regelwerk zu berücksichtigen. Diese setzen Merkmale und Merkmalsausprägungen in Beziehungen zueinander und können so die Ergebnisse beeinflussen.

Tabelle 6-12: Vorhersage der Vertriebsländer mit dem neuronalen Netz

AENR	y	$\hat{y}$												
		DE	**FR**	**PL**	**AT**	**GB**	**ES**	**BE**	**IT**	**NL**	**CZ**	**RU**	**IL**	**…**
88XXXX603	0P3GA	1,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	…
60XXXX166	0P3GG	0,36	0,23	0,00	0,00	0,00	0,01	0,00	0,00	0,00	0,00	0,00	0,01	…
68XXXX910	0P3I5	0,98	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	…
50XXXX269	0P3IG	0,00	0,00	0,00	0,00	0,00	1,00	0,00	0,00	0,00	0,00	0,00	0,00	…
79XXXX159	0P3I1	0,02	0,00	0,98	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	…
66XXXX777	0P3GT	0,00	0,01	0,00	0,00	0,00	0,03	0,00	0,91	0,00	0,00	0,00	0,00	…
80XXXX119	0P3GA	1,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	…
79XXXX169	0P3GA	0,02	0,03	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,00	0,00	…
70XXXX178	0P3HW	0,00	0,00	0,00	1,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	…
60XXXX499	0P3HS	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	1,00	0,00	0,00	0,00	…
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…

Tabelle 6-13: Bestimmung der Feature Importance mit dem Random Forest Model

Features	Feature Importance
0KTN - Notwendiger Reifengeschwindigkeitsindex	0,0395
0KFP - Fahrtschreiber	0,0331
0KG1 - Mauterfassungssystem Toll Collect	0,0263
0KIE - Anzahl Fahrzeugschlüssel	0,0237
0KI0 - Rückfahrwarner	0,0187
0KSA - Zulässiges ZGG NatZu	0,0173
0K45 - Druckluftbehälter Material	0,0159
0KHA - CEMT Bescheinigung	0,0153
…	…
0KLJ - Ausführung Hydrauliktank rechts	0,0000
0KN7 - Adaptersatz für NA	0,0000
0KRP - Anh.-Bremsanschlüsse an Tiefkuppelbock links	0,0000
0KRT - Ölkühlung für Verteilergetriebe	0,0000
0KUR - Zusätzliche Wendelflex-Leitungen	0,0000

6.4.1.2.5 Schlussfolgerung

Durch den Anwendungsfall wurde demonstriert, dass Machine Learning für die Prognose von kategorischen Eigenschaften von Produktvarianten eingesetzt werden kann. Am Beispiel der Vertriebsländer wurde gezeigt, dass dadurch sogar relative Größen bzw. deren Wahrscheinlichkeit für die Zugehörigkeit zu den einzelnen Klassen präzise bestimmt werden können. Damit wurden objektive Aussagen über die Vertriebsländer noch nicht verkaufter Konfigurationen ermöglicht und im Vergleich zu aktuellen Experteneinschätzungen ein Mehrwert generiert. Mit Hilfe des Random Forest wurden die wichtigsten und unwichtigsten Merkmale für die Vorhersage der Vertriebsländer bestimmt. Es wurde deutlich, dass sich die Algorithmen bei einer Klassifikation ähnlich verhalten wie bei einer Regression sowohl bei der Vorhersage als auch bei der Analyse der Modelle.

6.4.2 Ähnlichkeiten von Produktvarianten

Im Folgenden wurde die Datenvorbereitung (Baustein 3.0) und die Clusteranalyse (Baustein 3.3) des Frameworks angewendet, um Ähnlichkeiten zwischen Produktvarianten zu ermitteln und näher zu untersuchen (siehe Mehlstäubl et al. 2023c).

6.4.2.1 Datenvorbereitung

Für die Vorbereitung von Vertriebsdaten für eine Clusteranalyse ergaben sich aus dem Framework die Bereinigung fehlender und konstanter Werte, die Kodierung und die Dimensionsreduktion (siehe Tabelle 6-14). Der Vertriebsdatensatz für das Clustering enthielt die Konfigurationen von insgesamt 189 802 Produktvarianten mit 986 Merkmalen und insgesamt 12 511 Merkmalsausprägungen.

Die Effizienz des Clusteralgorithmus hängt stark von der Dimension der Eingangsmerkmale und der Anzahl der Produktvarianten ab. Mit 986 Merkmalen war die Dimensionalität des gegebenen Datensatzes enorm hoch, sodass die verfügbaren Ressourcen kein effizientes Clustering zuließen. Aus diesem Grund wurde die Anzahl der Merkmale zusammen mit Experten des Industriepartners reduziert. In einem früheren Produktlogikprojekt wurde ein Satz von 246 Merkmalen ausgewählt, die einen großen Einfluss auf den Bauzustand des Fahrzeugs haben. In der Datenvorbereitung wurden im ersten Schritt, wie bei der Regression und Klassifikation, die Duplikate entfernt, da diese zum einen die Effizienz der Berechnungen der Clusteralgorithmen verringert und zum anderen keinen inhaltlichen Mehrwert bezüglich der Varianz des Produktportfolios des Industriepartners darstellen. Trotzdem wurden die Duplikate nicht komplett verworfen, sondern für die spätere Analyse der Cluster beibehalten.

Tabelle 6-14: Schritte zur Datenvorbereitung für das Clustering von Produktvarianten

	Konstante Merkmale	Fehlende Werte	Encoding	Skalierung	Dimensions-reduktion
Vertriebsdaten	x	o	x	-	o
Nutzungsdaten	-	o	-	x	o
Regressionsanalyse	x	x	o	o	-
Klassifikationsanalyse	x	x	o	o	-
Clusteranalyse	x	x	o	o	x
Assoziationsanalyse	x	-	o	o	-

x = erforderlich, - nicht erforderlich, o = keine Abhängigkeit

Im nächsten Schritt wurden die Konfigurationen auf Vollständigkeit untersucht. Zunächst wurde der Anteil der fehlenden Werte in jedem Merkmal betrachtet. Ein Merkmal und vier Konfigurationen wurden aufgrund fehlender Werte aus dem Datensatz entfernt. Alle verbleibenden fehlenden Werte in der Produktstruktur wurden durch die häufigsten Merkmalsausprägungen in den Merkmalen ersetzt. Der resultierende Datensatz enthielt 65 452 Produktvarianten und 245 Merkmale sowie 1 828 Merkmalsausprägungen. Anschließend wurde der Datensatz mit einer One-hot Kodierung kodiert, wodurch die Anzahl der Spalten des Datensatzes auf 1 828 erhöht wurde. Trotz der Reduktion der Eingangsmerkmale durch Experten wiesen die Daten eine hohe Dimensionalität auf. Für ein effizientes Clustering wurde daher zusätzlich eine Dimensionsreduktion mit einer MCA durchgeführt. Abbildung 6-8 zeigt die Beziehung zwischen der Anzahl der Dimensionen und der erklärten Varianz sowie die Auswirkung der Benzécri-Korrektur. In diesem Anwendungsfall wurden die angegebenen Daten auf 118 Dimensionen reduziert, wodurch 99 % der Varianz der Produktkonfigurationen repräsentiert wurden. Dadurch wurde sichergestellt, dass das nachfolgende Clustering mit dieser reduzierten Darstellung zu nahezu denselben Ergebnissen wie für den ursprünglichen Datensatz führt.

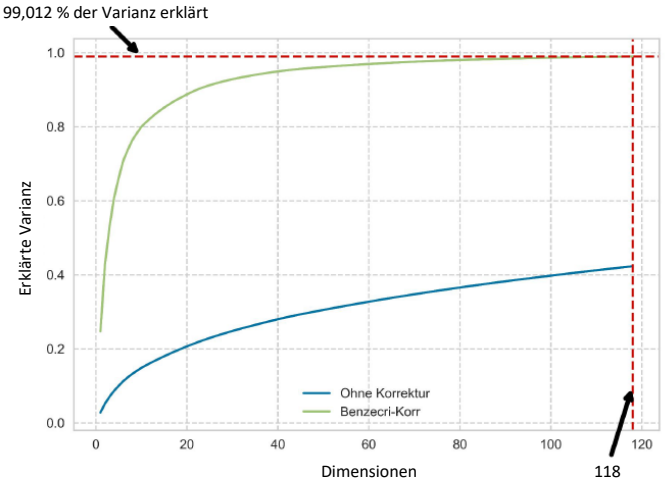


Abbildung 6-8: Zusammenhang zwischen der Anzahl an reduzierten Dimensionen und der erklärten Varianz

6.4.2.2 Clustering

Im Folgenden wurden die Algorithmen k-Means, Mini-batch k-Means, x-Means, Linkage und DBSCAN implementiert. Jeder wurde mehrfach iteriert und die Anzahl der gesuchten Cluster dabei variiert. Das Intervall für die Anzahl gesuchter Cluster k wurde gemeinsam mit dem Industriepartner zwischen 10 und 100 festgelegt. Eine noch höhere Granularität des Clustering würde die Interpretation und anschließende Analyse der Cluster erschweren. Das Vorgehen bei der Iteration der verschiedenen Algorithmen für die Werte des gewählten Intervalls unterscheidet sich danach, ob die Zahl gesuchter Cluster k direkt als Inputparameter gewählt wird oder nicht.

Der EM Algorithmus ermöglicht die Suche nach Mischmodellen mit kugelförmigen („spherical“) oder elliptischen („diag“) Verteilungen sowie nach Mischmodellen, bei denen jede Verteilung die identische Form („tied“) oder eine unterschiedliche Form („full“) aufweist. Für die Eingangsdaten lieferte die Variante „spherical“ sowohl für den Silhouettenkoeffizienten als auch für den Davies-Bouldin-Index stets die besten Werte, weshalb diese im Folgenden genutzt wurde. Der x-Means Algorithmus hat die Eigenschaft, selbstständig das optimale Clustering in der vorgegebenen Spannweite von k zu identifizieren. Im vorliegenden Fall wurde mit dem x-Means kein sinnvolles Clustering identifiziert. Der Algorithmus ermittelte zunächst 100 Cluster. Infolgedessen wurde der Maximalwert für k auf 200, 300 und 500 erhöht. Jede Wiederholung ergab, dass die Anzahl der identifizierten Cluster gleich dem Maximalwert für k war. Weitere Untersuchungen zeigten, dass der Algorithmus die Aufteilung des Datensatzes erst bei 2 245 Clustern beendete. Dieses Ergebnis wurde nicht als nützlich angesehen, um Ähnlichkeiten zwischen Produktvarianten zu identifizieren. Daher wurde der Algorithmus nicht weiter betrachtet. Bei dem Linkage Algorithmus kann zwischen Single-, Complete-, Average- und Ward-Kriterium gewählt werden. Bei der vorliegenden Datenbasis wurde mit dem Ward-Kriterium das beste Clustering erzielt, weshalb dieser

im Folgenden näher untersucht wurde. Ester und Sander (2000) stellen fest, dass Datenstrukturen mit hierarchischen Clustern, mit Clustern unterschiedlicher Dichte und Clustern, die eng an Bereichen mit Rauschen liegen, problematisch für den DBSCAN Algorithmus sind. Der DBSCAN ermittelte lediglich zwei Cluster, welche in Abbildung 6-9 in braun und gelb markiert sind. Darüber hinaus wurden 585 Instanzen als Rauschen (markiert in schwarz) klassifiziert. Daraus lässt sich schließen, dass der Algorithmus für die vorliegenden Daten kein geeignetes Clusterverfahren darstellt. Er wurde daher im Zuge der nachfolgenden Evaluation und Einsatz nicht weiter berücksichtigt.

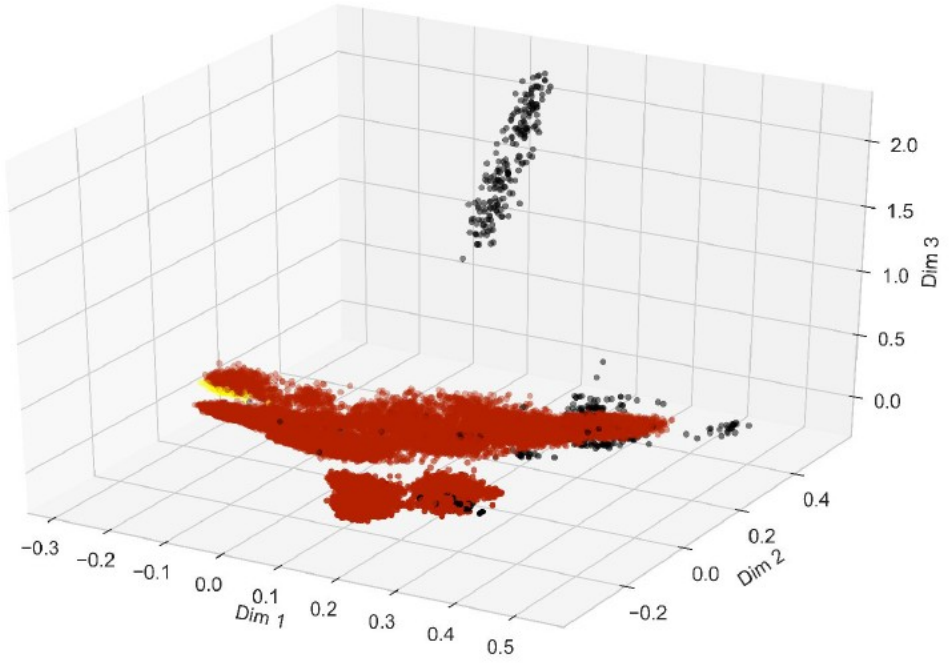


Abbildung 6-9: Clusterergebnisse unter Anwendung des DBSCAN Algorithmus

6.4.2.3 Evaluation

In Abbildung 6-10 sind die Evaluationsergebnisse der Algorithmen k-Means, Mini-batch k-Means, Ward-Linkage und EM (gaußsches Mischmodell spherical) zu sehen. Der Graph für die Trägheit zeigt, dass der k-Means und Ward-Linkage nahezu identische Verläufe besitzen, die niedriger und demzufolge besser als die der beiden anderen Algorithmen sind. Gleiches gilt für den Silhouettenkoeffizienten und den Davies-Bouldin-Index.

Der EM Algorithmus führt für alle CVIs zu den mit Abstand schlechtesten Werten. Der Graph des Mini-batch k-Means Algorithmus liegt für alle Indizes nah bei den Graphen von k-Means und Ward-Linkage, besitzt jedoch stets schlechtere Werte als die anderen beiden Algorithmen. Aus der gemeinsamen Betrachtung der CVIs lässt sich daher ablesen, dass sich der k-Means und der Ward-Linkage Algorithmus am besten für das Clustering der gegebenen Konfigurationsdaten eignen. Um eindeutig

bestimmten zu können, welcher der Algorithmen am leistungsfähigsten ist, wurde eine Normalisierung der CVIs vorgenommen. Die normalisierten Werte wurden dann pro Algorithmus über alle Clusterings aufsummiert. So ergaben sich für den EM und den Mini-batch k-Means Algorithmus eine Gesamtbewertung von 96,13 und 175,96. Die Summe des k-Means Algorithmus beträgt 219,84 und die des Ward-Linkage Algorithmus 221,17. Letzterer konnte damit als leistungsfähigster Algorithmus für die gegebenen Daten ausgemacht werden.

Abbildung 6-10: Evaluation der Clusterergebnisse der unterschiedlichen Algorithmen nach Mehlstäubl et al. (2023c)

In Abbildung 6-11 sind die CVIs für die Ergebnisse des Ward-Linkage in einem Graph dargestellt. Daraus wurde ersichtlich, dass die besten CVIs bei 28, 31 und 32 Cluster liegen.

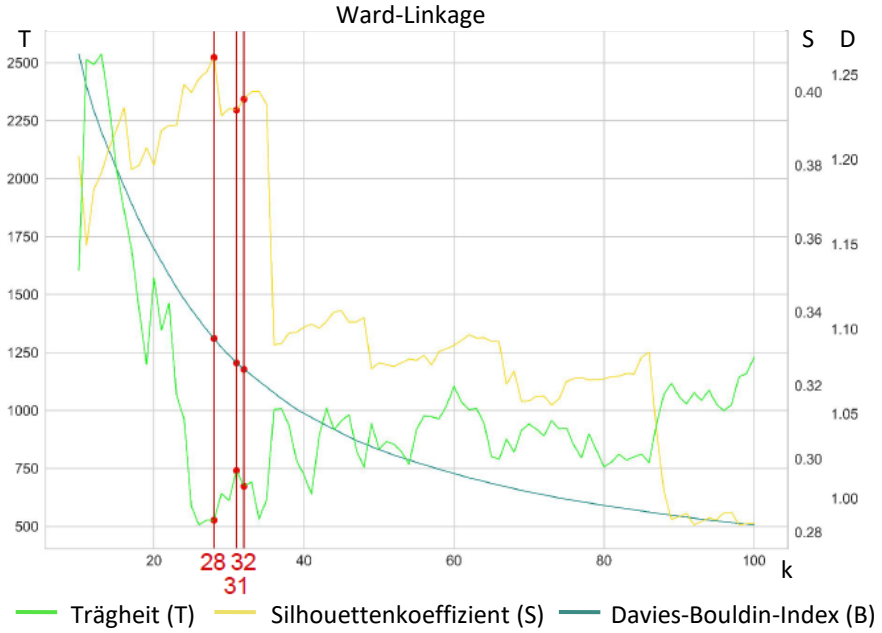


Abbildung 6-11: Visualisierung der CVIs für den Ward-Linkage Algorithmus nach Mehlstäubl et al. (2023c)

In Tabelle 6-15 sind die normalisierten Werte der CVIs für den Ward-Linkage Algorithmus abgebildet. Daraus ist zu entnehmen, dass das Clustering bestehend aus 31 Clustern die höchste Gesamtbewertung mit 2,324 erhielt. Gefolgt von 32 Clustern mit 2,321 und 28 Clustern mit 2,304. Aus diesem Grund wird ein Clustering mit 31 Clustern gewählt und im Folgenden für die Analyse herangezogen.

Tabelle 6-15: Normalisierte CVIs des Ward-Linkage Algorithmus für 28, 31 und 32 Cluster

k	T	S	D	T‘	T“	$T“_{skal}$	T_{norm}	S_{norm}	D_{norm}	sum_{norm}
31	1205	0,40	1,02	30,45	3,20	0,11	0,55	0,89	0,88	2,324
32	1177	0,40	1,01	26,69	2,52	0,09	0,49	0,91	0,92	2,321
28	1310	0,41	0,99	38,78	2,39	0,06	0,31	1,00	0,99	2,304

6.4.2.4 Einsatz

Die Ergebnisse des Clustering werden im Folgenden hinsichtlich der charakteristischen Merkmalsausprägungen der Cluster, dem Abstand zwischen den Clustern und den Stückzahlen der einzelnen Cluster analysiert.

Charakteristische Merkmalsausprägungen

In Tabelle 6-16 ist ein Auszug der Cluster 0 und 1 mit den charakteristischen Merkmalen dargestellt. Darin sind pro Cluster die zwei häufigsten Merkmalsausprägungen mit dem zugehörigen prozentualen Anteil aufgeführt. So wurde beispielsweise für das Cluster mit dem Index 0 ersichtlich, dass alle darin enthaltenen Konfigurationen

Sattelzugmaschinen waren, von denen knapp 88 % die Tonnageklasse 21 und 11 % die Tonnageklasse 22 aufwiesen. Zudem besitzen fast alle Konfigurationen den Fahrzeugtyp K oder J.

Tabelle 6-16: Ermittlung der charakteristischen Merkmalsausprägungen der Cluster

	Cluster 0		**Cluster 1**	
	Ausprägung 1	**Ausprägung 2**	**Ausprägung 1**	**Ausprägung 2**
Fahrzeugtyp	LKW K	LKW J	LKW F1	LKW F2
	45,83 %	42,51 %	42,47 %	34,69 %
Tonnageklasse	Tonnageklasse 21	Tonnageklasse 22	Tonnageklasse 22	Tonnageklasse 21
	88,34 %	11,55 %	50,30 %	40,61 %
Fahrgestellart	Sattel	-	Chassis	Sattel
	100 %	-	99,72 %	00,28 %
Fahrzeugart	Sattelzugmaschine (SA)	-	Pritschenwagen und Fahrgestell (LK)	Kipper (KI)
	100 %	-	67,27 %	15,76 %

Eine weitere Erkenntnis war, dass einige Cluster Auffälligkeiten in den Grundmerkmalen aufwiesen, welche den Fahrzeugtyp definieren. Z. B. im Cluster 24 war die Fahrgestellart mit knapp 57 % Chassis und 43 % Sattelzugmaschinen (Tabelle 6-17). Auch in den Merkmalen Motorfamilie, Fahrerhaus und Abgasemissionsstufe sind die häufigsten Merkmalsausprägungen dieses Clusters unterschiedlich. Dennoch wies das Cluster 146 Merkmale auf, in denen 95 % der Fahrzeuge dieselbe Merkmalsausprägung besaßen, welche auf den ersten Blick nicht offensichtlich waren.

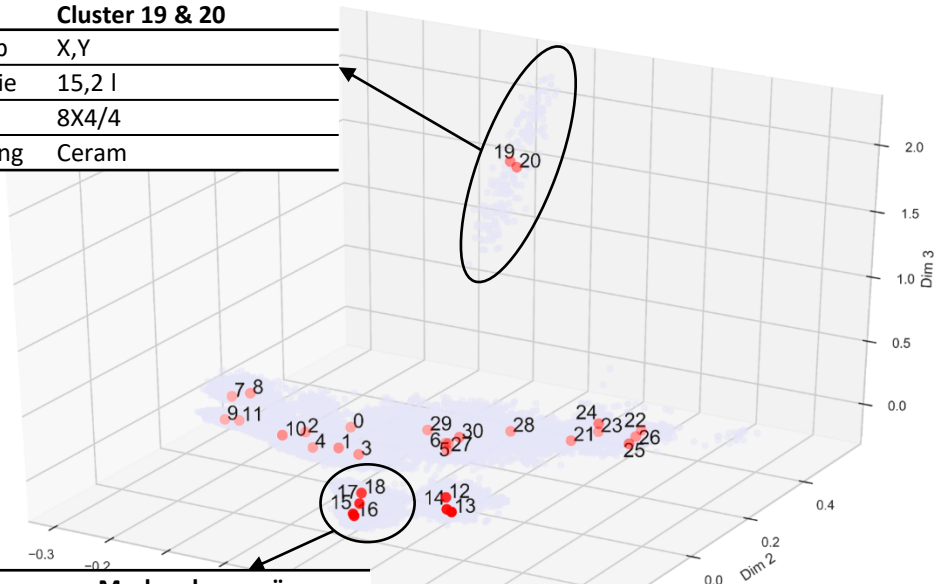
Tabelle 6-17: Cluster mit starker Homogenität in den Grundmerkmalen

	Cluster 24	
	Ausprägung 1	**Ausprägung 2**
Fahrzeugtyp	Typ LKW E	Typ LKW J
	32,24 %	23,68 %
Fahrgestellart	Chassis	Sattel
	56,58 %	43,42 %
Fahrzeugart	Kipper (KI)	Sattelzugmaschine (SA)
	51,32 %	43,42 %
Fahrerhaus	schmal, mittellang, normalhoch	schmal, lang, normalhoch
	69,74 %	21,71 %
Motorfamilie	10,5 l	12,4 l
	53,95 %	46,05 %
Abgasemissionsstufe	Euro 2	Euro 5
	54,61 %	44,74 %

Abstände

Zur Analyse der Ähnlichkeiten wurden die Abstände der Clusterzentren zum einen graphisch in einem dreidimensionalen Raum sowie in einer Matrix dargestellt. In Abbildung 6-12 sind die Clusterzentren in den Punktwolken rot markiert. So ist in der Abbildung beispielsweise zu erkennen, dass Cluster 19 und 20 eng beieinander liegen und von allen anderen Clustern weit entfernt sind. Davon ausgehend konnte aus der Analyse der charakteristischen Merkmalsausprägungen dieser Cluster auf deren Alleinstellungsmerkmale geschlossen werden. So gibt es Merkmalsausprägungen, wie beispielsweise die Fahrzeugtypen X und Y, die Radformel 8X4/4, die Fahrkupplung Ceram und der Motor 15,2 l, die ausschließlich in diesen beiden Clustern vorkamen. Auf die gleiche Weise wurden die Gemeinsamkeiten von Clustern untersucht, die sehr dicht beieinander liegen, wie die Cluster 15, 16, 17 und 18. Sie besaßen unter anderem einheitlich die Fahrgestellart Chassis, die Motoren 6,9 l oder 4,6 l, das Fahrerhaus kompakt oder komfortabel und die Bauart normalhoch.

Merkmale	**Merkmalsausprägungen Cluster 19 & 20**
Fahrzeugtyp	X,Y
Motorfamilie	15,2 l
Radformel	8X4/4
Fahrkupplung	Ceram



Merkmale	**Merkmalsausprägungen Cluster 15, 16, 17 & 18**
Fahrgestellart	Chassis
Fahrzeugart	Pritsche/Fahrgestell
Fahrerhaus	Kompakt, komfortabel
Motorfamilie	6,9 l, 4,6 l
Abgasemissionsstufe	Euro 6
Radformel	4x2
Bauart	Normalhoch

Abbildung 6-12: Graphische Darstellung der Centroide im dreidimensionalen Raum nach Mehlstäubl et al. (2023c)

Marktspezifische Eigenschaften

Weitere Erkenntnisse ließen sich aus den marktspezifischen Eigenschaften der Cluster ableiten. Hier wurde die quantitative Verteilung der Produktvarianten in den einzelnen Clustern betrachtet. Abbildung 6-13 zeigt die entsprechenden Auswertungen für die geclusterten 65 452 Konfigurationen sowie für das gesamte betrachtete Auftragsvolumen inklusive der Duplikate mit 189 798 Fahrzeugen. Dies zeigt, dass das Volumen der verkauften Produktvarianten in den Clustern unterschiedlich verteilt war. Während Cluster 11 mit 16 318 Produktvarianten die meisten Instanzen aufwies, beinhaltete Cluster 26 nur eine Produktvariante. Die charakteristischen Merkmale des Clusters 11 sind beispielsweise die Fahrgestellart Sattel, die Radformel 4x2, der Hauptradstand 3600 mm und die Tonnageklassen 22 (72 %) oder 21 (21 %), welche den Standardkonfigurationen für Sattelzugmaschinen für den Langstreckenverkehr entspricht. Auch bei den kleinen Clustern ließen sich die geringen Stückzahlen durch die charakteristischen Merkmalsausprägungen begründen. Die zuvor beschriebenen Cluster 19 und 20, die nur 62 und 141 Produktvarianten enthielten, konnten aufgrund des großen Motors und der speziellen Radformel auf eine Schwerlast-Sattelzugmaschine zurückgeführt werden. Gleiches gilt auch für das aus nur 117 Instanzen bestehende Cluster 25, welches als einziges Cluster ein Fahrzeug mit militärscher Ausführung aufwies. Die Zuordnung von nur einer ID zu Cluster 26 kann als Ausreißer erachtet werden. Aus mathematischer Sicht haben die CVIs aufgezeigt, dass diese Gruppierung in der Gesamtheit die kompaktesten und am besten differenzierten Cluster ergibt. Wie in Abbildung 6-13 zu erkennen ist, liegt das Cluster 26 nah an Cluster 25 und könnte daher in der Auswertung mit diesem gemeinsam betrachtet werden.

6.4.2.5 Schlussfolgerung

Durch dieses Unterkapitel wurde gezeigt, dass der Baustein 3.3 des Frameworks zum Clustering von Produktvarianten genutzt werden kann. Darüber hinaus konnte nachgewiesen werden, dass der Baustein 3.0 auch für das Clustering von Vertriebsdaten verwendet werden kann. Das Clustering ermöglichte es, Ähnlichkeiten zwischen Produktvarianten objektiv auf Basis von Daten zu ermitteln. Durch die Analyse der einzelnen Cluster konnte Wissen über das Produktportfolio des Industriepartners generiert werden. Es wurden zum einen die charakteristischen Merkmale der einzelnen Cluster betrachtet sowie deren Abstände zueinander. Des Weiteren wurden die Stückzahlen den einzelnen Clustern zugeordnet.

Die Ergebnisse wurden elf Experten aus verschiedenen Abteilungen und mit unterschiedlichem Hintergrund vorgestellt. Sie bestätigten den Mehrwert des Anwendungsfalls für die Analyse komplexer Produktportfolios. Allerdings wird der Mehrwert des Frameworks höher eingeschätzt als der Wert der ermittelten Cluster. Die Experten hielten den Einsatz des Clustering für vielversprechend und potenziell rentabel, wenn es als flexibler Ansatz in einem Softwaretool implementiert wird. Die eingegebenen Merkmale müssen wählbar sein, so dass individuelle und gezielte Analysen möglich

werden. Hinsichtlich der Auswahl der 246 Merkmale wurde angemerkt, dass diese eine Vielzahl unterschiedlicher Sichtweisen auf Produktvarianten abbilden und somit die Aussage nicht spezifisch genug für die potentielle Nutzergruppe ist. Aufgrund seiner Anpassungsfähigkeit kann der Ansatz von verschiedenen Anwendern in unterschiedlichen Kontexten zur Identifikation einer optimalen Gruppierung von Produktkonfigurationen genutzt werden und lässt sich auch leicht auf Komponenten oder sogar Stücklisten übertragen.

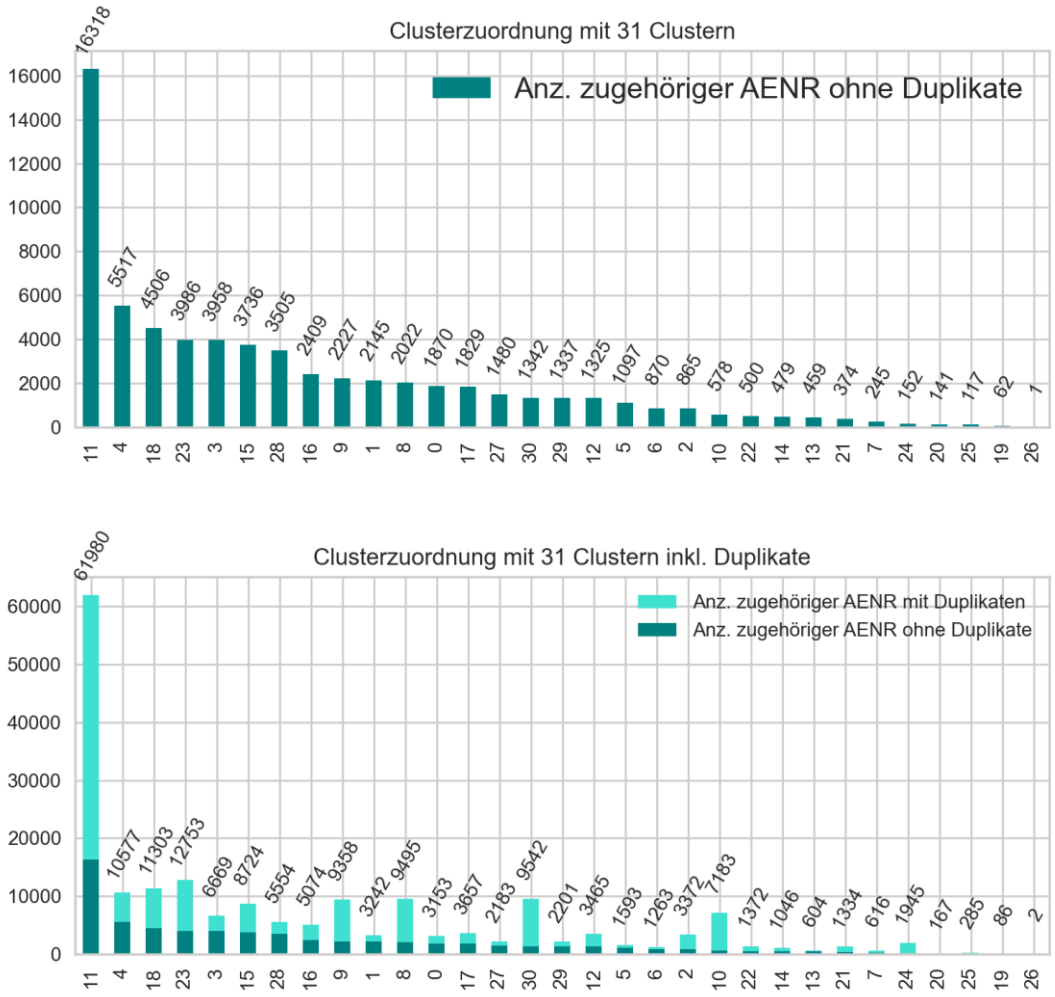


Abbildung 6-13: Quantitative Verteilung der Produktvarianten in den einzelnen Clustern

6.4.3 Korrelationen zwischen Merkmalsausprägungen

In diesem Unterkapitel wurden der Baustein 3.0 Datenvorbereitung und der Baustein 3.4 Assoziationsanalyse des Frameworks angewendet, um Korrelationen zwischen Merkmalsausprägungen in Form von Regeln abzubilden und das Produktportfolio dadurch zu reduzieren (siehe Mehlstäubl et al. 2023b).

6.4.3.1 Datenvorbereitung

Für die Vorbereitung der Vertriebsdaten zur Anwendung einer Assoziationsanalyse wurden konstante Werte entfernt sowie eine Kodierung der Daten durchgeführt (siehe Tabelle 6-18). Die 189 802 Fahrzeugkonfigurationen mit 986 Merkmalen und insgesamt 12 511 Merkmalsausprägungen wurden in einem ersten Schritt, wie beim Clustering, mit Experten auf 246 Merkmale und 1 824 Merkmalsausprägungen, welche eine hohe Relevanz für die Eigenschaften der Produktvarianten haben, reduziert. Anschließend wurde eine One-hot Kodierung durchgeführt.

Tabelle 6-18: Schritte zur Datenvorbereitung für die Ableitung von Korrelationen zwischen Merkmalsausprägungen

	Konstante Merkmale	Fehlende Werte	Encoding	Skalierung	Dimensions-reduktion
Vertriebsdaten	x	o	x	-	o
Nutzungsdaten	-	o	-	x	o
Regressionsanalyse	x	x	o	o	-
Klassifikationsanalyse	x	x	o	o	-
Clusteranalyse	x	x	o	o	x
Assoziationsanalyse	x	-	o	o	-

x = erforderlich, - nicht erforderlich, o = keine Abhängigkeit

6.4.3.2 Assoziationsanalyse

Die Assoziationsalgorithmen benötigen als Eingabeparameter einen minimalen Support-Wert für die Regeln sowie eine maximale Regeltiefe. Der minimale Support-Wert wurde auf 0,8 und die maximale Regeltiefe auf drei festgelegt. Für die Modellierung wurden sowohl der FP-Growth Algorithmus als auch der Apriori Algorithmus verwendet. Bei der Durchführung der Assoziationsanalyse mit dem Apriori Algorithmus trat ein Speicherplatzfehler auf, da dieser nicht ausreichend für die Analyse des Datensatzes war. Der FP-Growth Algorithmus identifizierte 327 212 Regeln, von denen 9 499 zwei Merkmalsausprägungen und 317 713 drei Merkmalsausprägungen enthielten.

6.4.3.3 Evaluation

Für die ermittelten Regeln wurden anschließend die Confidence-Werte berechnet. In Tabelle 6-19 ist ein Auszug der kodierten Regeln mit dem Confidence und Support-Wert abgebildet. Aufgrund der großen Anzahl von Regeln war ein iterativer Ansatz auf der Grundlage der einzelnen Merkmale sinnvoll. Im Folgenden werden die Evaluation der Regeln und die Reduktion des Produktportfolios beispielhaft an einer der identifizierten Regeln demonstriert. Die Regel lautet „Ohne Hydrauliktank, links (Material)“ → „Luftansaugung, hinter Fahrerhaus, hochgezogen“ und besitzt einen Support von

0,973 und Confidence von 0,991. Das bedeutet, dass in 99,1 % der Fälle diese beiden Merkmalsausprägungen zusammen verkauft wurden.

Tabelle 6-19: Auszug aus den Assoziationsregeln mit zugehörigem Support- und Confidence-Wert

X		Y	Support	Confidence
0P0CE	0P1LZ	0P1FU	0,915	0,995
0P1LM	0P4FS	0P1I7	0,863	0,995
0P0CE	0P1IN	0P3BP	0,839	1,000
0P0R8	0P1NV	0P1I7	0,863	1,000
0P4YW	0P4DC	0P0GZ	0,915	0,929
0P1S6	0P0FD	0P4FU	0,954	0,991
0P0OI	0P1R6	0P0B7	0,814	0,901
0P4FU	0P1KP	0P0B7	0,809	0,900
…	…	…	…	…

6.4.3.4 Einsatz

Im nächsten Schritt wurde die Regel mit den bestehenden Produktportfolioeinschränkungen im Produktdatenmodell verglichen. Die Merkmalsausprägung „Ohne Hydrauliktank, links (Material)“ kann aktuell mit den Merkmalsausprägungen „Luftansaugung, hinter Fahrerhaus, hochgezogen“, „Vorbereitung für hochgezogene Luftansaugung“ und „Luftansaugung, hinter Fahrerhaus, oberhalb Getriebe (nicht für Standardaufbauten)“ konfiguriert werden. Die Kombinatorikregel „Ohne Hydrauliktank, links (Material)“ erfordert „Luftansaugung, hinter Fahrerhaus, hochgezogen“ wurde definiert und dadurch die Kombinierbarkeit mit den anderen beiden Merkmalsausprägungen eingeschränkt.

Die Definition von Kombinatorikregeln schränkte zunächst die externe Vielfalt des Produktportfolios ein. Diese Einschränkungen reduzieren jedoch ebenfalls die interne Vielfalt durch die Abhängigkeit über die Booleschen Teileauswahlregeln zu den Komponentenvarianten und dadurch die Komplexität und Kosten. Die Auswirkungen der Kombinatorikregeln wurden beim Industriepartner mit einer Rule Engine berechnet. Dabei wurden 14 Komponentenvarianten identifiziert, die nicht mehr benötigt werden, da sie aufgrund der einen eingeführten Regel nicht mehr vom Konfigurator ausgewählt werden können.

6.4.3.5 Schlussfolgerung

Der Anwendungsfall ermöglichte die Identifikation von Korrelationen zwischen Merkmalsausprägungen komplexer Produktportfolios. Dadurch konnte eine Reduktion des Produktportfolios vorgenommen werden, indem Kombinatorikregeln aus den Assoziationsregeln abgeleitet wurden. Durch die Verwendung des FP-Growth Algorithmus konnte das komplexe Produktportfolio des Industriepartners mit einer Vielzahl von Merkmalen und Merkmalsausprägungen analysiert werden. Es wurde gezeigt, wie

Assoziationsregeln in konkrete Kombinatorikregeln übersetzt und externe und interne Varianz reduziert werden kann. Dies ermöglichte die automatisierte Analyse des Produktportfolios beim Industriepartner. Der Abgleich der ermittelten Assoziationsregeln mit dem bestehenden Regelwerk im Produktdatenmodell erfolgte manuell. Darüber hinaus konnten mit dem FP-Growth Algorithmus lediglich Regeln mit hohen Support-Werten bestimmt werden. Daraus resultierten Kombinatorikregeln der Form „Merkmalswert A erfordert Merkmalswert B“. Merkmale, die selten zusammen auftreten, wurden bisher nicht bestimmt.

6.5 Erfolgsvalidierung mit einer Expertenbefragung

Die übergeordnete Vision dieser Arbeit ist die Produktportfoliokomplexität in Industrieunternehmen besser handhabbar zu machen. Dafür wurde ein Framework mit der Zielsetzung der systematischen Generierung von Wissen für die Analyse komplexer Produktportfolios mittels Machine Learning eingeführt. Für die Entwicklung wurden in Kapitel 4.3 Kriterien definiert, um den Mehrwert im Hinblick auf die Zielsetzung und den Forschungsbedarf sowie aus Sicht der Wissenschaft bewertbar zu machen.

In der Erfolgsvalidierung wurden zur Bewertung der Erfüllung dieser Kriterien Expertenbefragungen durchgeführt. Insgesamt wurden neun Experten befragt, von denen vier von dem Industriepartner kamen, bei dem die Anwendungsvalidierung durchgeführt wurde, und daher Teile der Ergebnisse kannten. Diese stammten aus den Bereichen „Virtuelle Produktentwicklung“ und „Gesamtfahrzeug“. Die fünf anderen Experten wurden von außerhalb des Unternehmens gewählt, um die Allgemeingültigkeit der Bewertungsergebnisse sicherzustellen. Dabei handelte es sich zum Großteil um Berater, welche auf Erfahrung im Variantenmanagement unterschiedlicher Unternehmen und Branchen zurückgreifen konnten. Eine Übersicht der Experten mit deren Charakteristiken kann Tabelle 6-20 entnommen werden.

In der Durchführung der Expertenbefragungen wurden zuerst das Framework sowie die Ergebnisse der Fallstudie aus der Anwendungsvalidierung präsentiert. Anschließend wurde eine Umfrage durchgeführt, in der die Kriterien in Form von Aussagen aufbereitet und die Zustimmung der Experten mit einer 5-Punkt-Likert-Skala von „1 = trifft nicht zu“ bis „5 = trifft zu“ abgefragt wurde. Zudem stand es den Teilnehmern frei „keine Aussage möglich“ zu wählen. Die Bewertungsergebnisse wurden im Anschluss präsentiert und mit den Experten diskutiert. Eine detaillierte Übersicht der Bewertungen der einzelnen Experten kann Anhang A3.2 entnommen werden.

Ergebnisse der Bewertung der inhaltlichen Kriterien

Die inhaltlichen Kriterien wurden von den Experten mehrheitlich positiv bewertet. Eine Übersicht der Bewertungsergebnisse kann Tabelle 6-21 entnommen werden. Darin ist das arithmetische Mittel sowie die Spannweite der Angaben abgebildet. Durch

die Anwendung des Frameworks bei einem Nutzfahrzeughersteller mit einem besonders breiten und tiefen Produktportfolio, welches sich im ständigen Wandel befindet, wurde bestätigt, dass das Framework auf **komplexe Produktportfolios** angewendet werden kann. Die Experten stimmten der Erfüllung dieser Anforderung mit Ø 4,7 zu.

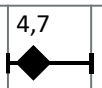
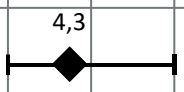
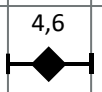
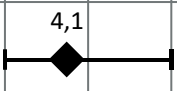
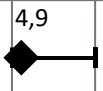
Tabelle 6-20: Übersicht der Experten der Erfolgsvalidierung

Name	Position	Domäne	Erfahrung im P&V	Produkte / Branchen
Experte A	Abteilungsleiter	Virtuelle Produktentwicklung	5-10 Jahre	Nutzfahrzeuge
Experte B	Haupt-abteilungsleiter	IT für die Entwicklung	10-20 Jahre	Nutzfahrzeuge
Experte C	Experte	Varianten-management	5-10 Jahre	Nutzfahrzeuge
Experte D	Experte	Gesamtfahrzeug	2-5 Jahre	Nutzfahrzeuge
Experte E	Partner	Produktkosten-optimierung	2-5 Jahre	Nutzfahrzeuge, Automotive, Motoren
Experte F	Account Manager	Produktkonfiguration	5-10 Jahre	Maschinen- und Anlagenbau, Nutzfahrzeuge
Experte G	Consultant	Digitale Entwicklung & Produktion	<2 Jahre	Automotive
Experte H	Leiter Consulting	Produktkonfiguration	>20 Jahre	Maschinen- und Anlagenbau, Baugewerbe, Elektronik und Mechatronik
Experte I	Senior Consultant	Produktentwicklung & Innovation	2-5 Jahre	Nutzfahrzeuge, Landmaschinen

Das Framework stellt ein Verständnis für die aktuellen **Wissensbedarfe** für den Einsatz von Machine Learning aus der Literatur und Industrie bereit und ordnet diese dem Prozess zur Analyse und Anpassung von Produktportfolios zu. Dadurch wurde ein Geschäftsverständnis für Machine Learning im Produktportfolio- und Variantenmanagement generiert und ein Ausgangspunkt für Unternehmen für den Einsatz von Machine Learning bereitgestellt. Die Experten stimmten der Erfüllung des Kriteriums teilweise bis voll zu (Ø 4,3). Einer der Befragten merkte an, dass es zwar den Ausgangspunkt für den Einsatz von Machine Learning bildet, jedoch der Anwender sowie die Bestimmung des monetären Nutzens der Anwendungsfälle nicht stark genug berücksichtigt werden.

Eine **datenbasierte Beschreibung komplexer Produktportfolios** beinhaltet das Framework durch die allgemeine Darstellung von Produktdatenmodellen, Vertriebsdaten und Nutzungsdaten mit deren Elementen. Die Experten bestätigten mit einem Wert von Ø 4,6, dass dies die zentralen Daten für die Analyse komplexer Produktportfolios sind und durch deren Beschreibung ein Datenverständnis erzeugt wird. Sie bestätigten auch, dass unterschiedliche Unternehmen auf der Ebene der operativen Produktportfoliogestaltung vergleichbare Produktdatenmodelle haben. Auf der Ebene der Stücklistenelemente sehen die Modelle jedoch teileweise anders aus.

Tabelle 6-21: Bewertungsergebnisse für die inhaltlichen Kriterien

	5 4 3 2 1
1. Das Framework ermöglicht die Ermittlung von Eigenschaften **komplexer Produktportfolios** mit einer Vielzahl an Merkmalsausprägungen und Komponentenvarianten.	
2. Das Framework vermittelt ein Geschäftsverständnis für die Analyse komplexer Produktportfolios durch die **Systematisierung der Wissensbedarfe** im Entscheidungsprozess.	
3. Durch das Framework findet eine **datenbasierte Beschreibung komplexer Produktportfolios** statt, wodurch ein Datenverständnis in der betrachteten Domäne bereitgestellt wird.	
4. Das Framework bietet eine systematische Unterstützung bei der **Vorbereitung von Produktportfoliodaten**.	
5. Das Framework liefert für die industrielle Anwendung von Machine Learning eine Hilfestellung bei **der Auswahl und Evaluation von Algorithmen**.	
6. Das Framework erläutert verschiedene Möglichkeiten zum **Einsatz der Machine Learning Modelle** für die Analyse komplexer Produktportfolios.	

Legende:
5 – trifft zu | 4 – trifft eher zu | 3 – trifft teilweise zu | 2 – trifft eher nicht zu | 1 – trifft nicht zu

Eine Hilfestellung bei der **Vorbereitung von Produktportfoliodaten** bietet das Framework zum einen durch die Zuordnung der Produktportfoliodaten und den Machine Learning Verfahren zu den einzelnen Schritten der Datenvorbereitung. Zum anderen werden die einzelnen Verfahren unter Berücksichtigung der entsprechenden Datencharakteristiken beschrieben. Die Experten stimmten der Erfüllung dieses Kriteriums mit einer Ausnahme voll zu (Ø 4,8). Letzterer gab an, dass für ihn die Anforderung teilweise erfüllt ist, da für die Datenaufbereitung neben den Verfahren auch eine entsprechende Toolunterstützung sowie ein konkreter Programmcode erforderlich ist.

Die **Auswahl und Evaluation von Algorithmen** wird durch deren Gegenüberstellung sowie die Beschreibung von Evaluationskriterien unter Berücksichtigung der Datencharakteristiken unterstützt. Der Erfüllung dieses Kriteriums stimmten die Experten mit Ø 4,1 zu. Auch hier wurde angemerkt, dass neben der Auswahl auch die Implementierung in Form von konkretem Programmcode eine Rolle in Unternehmen spielt, was im Framework nicht betrachtet wird. Zudem sind die Algorithmen abhängig von der Struktur der Daten. Ein Kritikpunkt der Experten war daher, dass im Framework nicht darauf eingegangen wird, welche Strukturen die Daten unterschiedlicher Produktportfolios annehmen können.

Das Framework erläutert verschiedene Möglichkeiten zum **Einsatz der Machine Learning Modelle** für die Analyse komplexer Produktportfolios. Dabei werden unterschiedliche Wege erläutert, wie in Abhängigkeit des Machine Learning Verfahrens die

Modelle in der operativen Produktportfoliogestaltung eingesetzt werden können. Dieses Kriterium wurde innerhalb der inhaltlichen Kriterien von den Experten mit Ø 4,9 am besten bewertet.

Ergebnisse der Bewertung der formalen Kriterien

Die Experten bewerteten auch die formalen Kriterien mehrheitlich als erfüllt. Das arithmetische Mittel sowie die Spannweite der Angaben der Experten können Tabelle 6-22 entnommen werden. Für die Untersuchung der **Nützlichkeit** wurde zum einen abgefragt, ob der Zweck des Frameworks nützlich ist, und zum anderen, ob das Framework diesen Zweck erfüllt. Die Experten stimmten mit einem Wert von Ø 4,8 zu, dass die heutigen Verfahren in Unternehmen die Komplexität nicht handhaben können und daher intelligente und datengetriebene Lösungen erforderlich sind. Es wurde jedoch angemerkt, dass dies nur bei Unternehmen mit einer großen Anzahl von Varianten der Fall ist. Ein Experte berichtete aus seiner Erfahrung, dass Unternehmen mit mehr als ungefähr 1 000 Produktvarianten Probleme mit der Varianz haben und Unterstützung benötigen. Von den Experten wurde ebenfalls mit einem Wert von Ø 4,5 bestätigt, dass der Zweck durch das Framework erfüllt wird und ein Beitrag zur besseren Handhabung komplexer Produktportfolios durch die Generierung von Wissen aus Daten mittels Machine Learning geleistet wird. Es wurde betont, dass durch das Framework und den Einsatz von Machine Learning in den Anwendungsfällen der Validierung eine Verbesserung im Vergleich zu dem bisherigen Einsatz von Expertenwissen sowie eine Erhöhung der Objektivität erzielt wurde. Jedoch wurde angemerkt, dass das Framework zwar eine Unterstützung bei der Anwendung von Machine Learning zur Analyse komplexer Produktportfolios bietet, jedoch bei manchen Anwendungsfällen die Verknüpfung zum Einsatz des Wissens im Tagesgeschäft noch hergestellt werden muss. Für weitere Forschungstätigkeiten wurde daher der Fokus auf die Einbindung der Ergebnisse der Anwendungsfälle in die Prozesse des Produktportfolio- und Variantenmanagements sowie die Herausstellung des monetären Mehrwerts angeregt.

Für die Bewertung der **Konsistenz** wurde eine Unterscheidung zwischen der internen Konsistenz des Frameworks sowie der externen Konsistenz im Hinblick auf die allgemeinen Ansichten und Normen im Produktportfolio- und Variantenmanagement vorgenommen. Die Beurteilung der internen Konsistenz im Rahmen der Befragungen auf Basis der vorgestellten Ergebnisse war für die Teilnehmer schwer und wurde mit einer Zustimmung von Ø 4,2 bewertet. Die Beteiligten bestätigten mit einem Wert von Ø 4,6, dass das Framework nicht im Widerspruch zu den allgemeinen Ansichten im Produktportfolio- und Variantenmanagement steht.

Für die Bewertung des **Umfangs** des Frameworks wurde zum einen dessen Übertragbarkeit auf andere Unternehmen sowie auf andere Bereiche der Produktentwicklung betrachtet. Die Übertragbarkeit auf andere Unternehmen trifft nach Einschätzung der Experten mit Ø 4,0 eher zu. Die Experten bestätigten, dass diese auf Produktportfolios anderer Unternehmen auch aus anderen Branchen auf der Ebene der Merkmale und Komponenten aufgrund der vergleichbaren Produktdatenmodelle möglich ist. Dies ist

nach Meinung einiger der Experten auf der Ebene von Stücklisten und Sachnummern oft nicht der Fall, weshalb die Übertragbarkeit hier eingeschränkt ist. Aufgrund der Fokussierung der Arbeit auf die operative Produktportfoliogestaltung, die Entscheidungen auf der Ebene von Produktvarianten über Merkmale und Komponenten trifft, ist dies auch nicht erforderlich. Jedoch müssen die Unternehmen ein Baukastensystem besitzen und für die Anwendung ist immer ein Branchenverständnis erforderlich. Aufgrund der allgemeinen Formulierung ist nach Einschätzung der Experten auch eine Übertragbarkeit des Ansatzes auf andere Bereiche der Produktentwicklung möglich (Ø 4,6). Jedoch ist eine Anpassung und vor allem die Erarbeitung eines Geschäfts- und Datenverständnisses erforderlich.

Tabelle 6-22: Bewertungsergebnisse für die formalen Kriterien

	5 4 3 2 1
7. In Unternehmen sind intelligente und datengetriebene Lösungen zur Produktportfolioanalyse notwendig, da die **heutigen Verfahren die Komplexität nicht handhaben** können.	4,8
8. Das entwickelte Framework leistet einen Beitrag zur **besseren Handhabung komplexer Produktportfolios** durch die Generierung von Wissen aus Daten mittels Machine Learning.	4,5
9. Die einzelnen Bausteine des Frameworks und ihre Beschreibung sind **in sich widerspruchsfrei**.	4,2
10. Die Inhalte des Frameworks stehen **nicht im Widerspruch zu dem allgemeinen Wissen** im Produktportfolio- und Variantenmanagement.	4,6
11. Das Framework kann auf komplexe Produktportfolios **unterschiedlicher Unternehmen** angewendet werden.	4,0
12. Das Framework kann mit geringfügigen Anpassungen **auf andere Bereiche in der Produktentwicklung erweitert** werden, ohne seinen Nutzen wesentlich zu verringern.	4,6
13. Die Bausteine des Frameworks sind **einfach zu verstehen** und können von potenziellen Nutzern eingesetzt werden.	4,0
14. Das Framework bildet die **Grundlage für neue Forschungstätigkeiten** zum Einsatz von Machine Learning im Produktportfolio- und Variantenmanagement und in der Produktentwicklung.	5,0

Legende:
5 – trifft zu | 4 – trifft eher zu | 3 – trifft teilweise zu | 2 – trifft eher nicht zu | 1 – trifft nicht zu

Die Bewertung der **Einfachheit** der Anwendung des Frameworks ging von „trifft eher nicht zu“ bis „trifft zu“ bei den Experten am weitesten auseinander (Ø 4,0). Dennoch waren sich die Experten bei der anschließenden Diskussion einig, dass das Framework an sich einfach formuliert und für Personen mit Erfahrung im Produktportfolio- und

Variantenmanagement sowie Kenntnissen im Machine Learning einfach verständlich und anwendbar ist. Da die Disziplinen des Produktportfolio- und Variantenmanagements sowie Machine Learning beide selbst kompliziert sind, gab es Teilnehmer, die diesen Punkt mit „trifft eher nicht“ oder „trifft teilweise zu“ bewertet haben.

Die Experten stimmten einheitlich zu, dass das Framework den Ausgangspunkt für neue Forschungstätigkeiten zum Einsatz von Machine Learning im Produktportfolio- und Variantenmanagement und in der Produktentwicklung bildet und damit dessen **Fruchtbarkeit** für neue Forschungstätigkeiten gegeben ist (Ø 5,0). Ein Experte bemerkte, dass Machine Learning im Produktportfolio- und Variantenmanagement noch ganz am Anfang steht und die Unternehmen noch lange begleiten wird.

6.6 Schlussfolgerung zur Validierung

Für die Validierung gemäß der DRM, wurde das Framework zuerst angewendet und anschließend eine Bewertung des Erfolgs vorgenommen. Der Einsatz des Frameworks im Rahmen einer Fallstudie bei einem Nutzfahrzeughersteller mit realen Fahrzeugdaten hat dessen industrielle Anwendbarkeit bestätigt. Durch das Framework wurden drei Anwendungsfälle für Machine Learning zur Analyse eines komplexen Produktportfolios umgesetzt. Deren Auswahl und Implementierung fand auf Basis der im Framework bereitgestellten Wissensbedarfe und beschriebenen Produktportfoliodaten statt.

Durch den ersten Anwendungsfall wurden technische und marktspezifische Produkteigenschaften für neue Produktvarianten bestimmt. Am Beispiel der Fahrzeuggewichte und CO_2-Emissionen konnte gezeigt werden, dass Machine Learning schneller, genauer und unter geringerem Ressourceneinsatz die Produkteigenschaften für neue Konfigurationen prognostizieren kann. Mit der Vorhersage der Preise und Vertriebsländer konnten Eigenschaften ermittelt werden, welche aktuell lediglich durch Experten abgeschätzt werden können. Das Verhalten der Modelle auf z. B. Exoten des Produktportfolios ist noch näher zu untersuchen. Das Clustering des Produktportfolios hat es ermöglicht, ähnliche Produktvarianten zusammenzufassen, exotische Cluster zu erkennen sowie charakteristische Merkmale objektiv darzustellen. Die Einbindung des dadurch erzeugten Wissens über die Eigenschaften des Produktportfolios in das Tagesgeschäft des Industriepartners bleibt in der Anwendung teilweise noch offen. Mit einer Assoziationsanalyse konnten Korrelationen zwischen einzelnen Merkmalsausprägungen identifiziert und das Produktportfolio durch die Definition von Kombinatorikregeln eingeschränkt werden. Aktuell können jedoch nur Regeln mit hohen Support-Werten bestimmt werden.

In der Erfolgsvalidierung wurden das Framework sowie die Ergebnisse sowohl hinsichtlich der inhaltlichen als auch der formalen Kriterien positiv bewertet. Durch die Erfüllung der inhaltlichen Kriterien konnte ein Beitrag in Bezug auf die definierte Zielsetzung der Arbeit sowie den hergeleiteten Forschungsbedarf geleistet werden. Durch

die positive Bewertung der formalen Kriterien wurde die Erfüllung der Anforderungen an das Framework aus Sicht der Forschung in der Produktentwicklung ebenfalls erfüllt.

7 Diskussion

In diesem Kapitel werden die Ergebnisse des Forschungsvorhabens diskutiert. Dafür wird zuerst auf den Nutzen und die Einschränkungen eingegangen, bevor anschließend der Mehrwert aus Sicht der Forschung und Industrie herausgestellt wird.

7.1 Nutzen und Einschränkungen

In Unternehmen basieren Entscheidungen zur Anpassung des Produktportfolios trotz der immer weiter steigenden Komplexität entweder auf dem Expertenwissen der Entwickler oder auf der manuellen Analyse von Daten. Dies führt zu unzureichenden Entscheidungsergebnissen und einem Mangel an Nachvollziehbarkeit und Transparenz. Um einen Beitrag zur besseren Handhabung der Varianz zu leisten, wurde in dieser Arbeit ein Framework zur Analyse komplexer Produktportfolios eingeführt. Das Framework soll Unternehmen in die Lage versetzen, Daten aus verschiedenen Systemen automatisiert zu analysieren, um die marktgerechte Anpassung des Produktportfolios zu unterstützen.

Im ersten Baustein des Frameworks werden die Wissensbedarfe beschrieben, welche für den Prozess zur Analyse und Anpassung von Produktportfolios und somit für Rationalisierungsprojekte relevant sind und mit Machine Learning abgedeckt werden können. Dadurch wurde die erste Forschungsfrage „*Welches Wissen kann mittels Machine Learning für die Analyse komplexer Produktportfolios generiert werden?*“ beantwortet und ein Verständnis für die aktuellen Herausforderungen im Produktportfolio- und Variantenmanagement bereitgestellt. Dabei wurde kein Anspruch auf Vollständigkeit erhoben, stattdessen stellt der Baustein eine Momentaufnahme aus Sicht der Literatur und Industrie dar und bildet den Ausgangspunkt für Unternehmen für die Einführung von Machine Learning in der operativen Produktportfoliogestaltung. Aus diesem Grund ist eine regelmäßige Überprüfung der Aktualität sowie die Durchführung von Anpassungen erforderlich. Das Vorgehen zur Ermittlung des Wissens berücksichtigte neben der Literatur lediglich die Bedürfnisse eines Industriepartners. In der Erfolgsvalidierung wurde jedoch deren allgemeine Relevanz von Experten aus mehreren anderen Unternehmen bestätigt. Für ein Geschäftsverständnis in einem Unternehmen sind zusätzlich die Einsparpotentiale und der monetäre Nutzen der einzelnen Anwendungsfälle, welche aktuell im Framework nicht betrachtet werden, zu bestimmen.

Der zweite Baustein des Frameworks enthält eine allgemeine Beschreibung der Produktportfoliodaten. Diese sind das Produktdatenmodell, die Vertriebsdaten und die Nutzungsdaten. Mit Hilfe dieser Daten kann das in Baustein 1 beschriebene Wissen unter Verwendung von Machine Learning Verfahren erzeugt werden. Dadurch wurde die zweite Forschungsfrage „*Welche Daten sind für die Generierung von Wissen zur Analyse komplexer Produktportfolios notwendig?*“ beantwortet. In diesem Baustein wurde ebenfalls kein Anspruch auf Vollständigkeit erhoben, da grundsätzlich alle

Daten, die im Laufe des Produktlebenszyklus einer Produktvariante entstehen, Rückschlüsse auf die operative Produktportfoliogestaltung zulassen. Stattdessen wurden mit Experten die relevantesten Daten ermittelt, welche die Umsetzung der zuvor beschriebenen Anwendungsfälle ermöglichen. Dennoch sind weitere Daten und deren Nutzen für den Einsatz von Machine Learning zur Analyse komplexer Produktportfolios zu untersuchen. Die Betrachtung der Daten fand auf der Ebene von Merkmalen und Komponenten statt. Untersuchungen auf der Ebene von Stücklistenelementen wurden bisher nicht im Framework berücksichtigt.

Baustein 3 stellt zum einen verschiedene Verfahren zur Datenvorbereitung vor, welche in Abhängigkeit der Daten und deren Charakteristiken sowie dem verwendeten Machine Learning Verfahren ausgewählt und eingesetzt werden können. Darin werden für jedes Verfahren die unterschiedlichen Algorithmen gegenübergestellt und deren Auswahl unterstützt. Zudem werden Metriken für die Evaluation der Modelle bereitgestellt und auf die unterschiedlichen Möglichkeiten, wie die Modelle eingesetzt werden können, eingegangen. Dadurch wurde die dritte Forschungsfrage *„Wie kann mittels Machine Learning Verfahren Wissen für die Analyse komplexer Produktportfolios generiert und eingesetzt werden?“* beantwortet. Dabei wurde eine Einschränkung hinsichtlich der Verfahren der Regressions-, Klassifikations-, Cluster- und Assoziationsanalyse vorgenommen. Eine Untersuchung weiterer Machine Learning Verfahren und deren Potenziale und Einschränkungen in der operativen Produktportfoliogestaltung wurde nicht vorgenommen. Zudem wird nicht darauf eingegangen, welche unterschiedliche Struktur Produktportfoliodaten annehmen können und welche Auswirkungen diese auf die Verfahren haben. Außerdem beinhaltet das Framework keine Aspekte der Toolunterstützung und ist in keinem Tool, welches zu den einzelnen Verfahren den entsprechenden Programmcode bereitstellt, implementiert.

Durch die Anwendung des Frameworks mit realen Daten bei einem Industriepartner konnte sowohl die Anwendbarkeit bestätigt als auch durch die Implementierung von drei Anwendungsfällen der Nutzen von Machine Learning zur Analyse komplexer Produktportfolios im Vergleich zu den aktuellen erfahrungs- und regelbasierten Verfahren herausgestellt werden. Eine Einschränkung, welche sich durch das Vorgehen zur Entwicklung und Validierung des Frameworks ergibt, ist, dass die Identifikation der Wissensbedarfe und die datenbasierte Beschreibung komplexer Produktportfolios mit demselben Industriepartner durchgeführt wurde wie die Anwendungsvalidierung. Die Experten bestätigten in der Erfolgsvalidierung zwar die Anwendbarkeit auf die Produktportfolios anderer Unternehmen, jedoch ist ein tatsächlicher Einsatz auf unterschiedliche Produktportfolios noch ausstehend. Für die Verwendung des Produktportfolios durch Dritte ergibt sich die Einschränkung, dass Wissen über komplexe Produktportfolios sowie Machine Learning erforderlich ist.

Zusammenfassend lässt sich sagen, dass das Forschungsvorhaben die erhobenen Forschungsfragen beantwortet, jedoch ebenfalls einige Einschränkungen hinsichtlich der Vorgehensweise sowie den Ergebnissen beinhaltet, welche den Ausgangspunkt für

weitere Forschungstätigkeiten bilden. Die Erfüllung der in Kapitel 4.3 definierten Anforderungen an das Framework wurde von den Experten positiv bewertet, wodurch der Beitrag für die Forschung und Industrie bestätigt wurde. Dieser Ergebnisbeitrag wird im Folgenden näher beleuchtet.

7.2 Ergebnisbeitrag für die Forschung

In dieser Arbeit wurde ein Framework entwickelt, mit dem Machine Learning Verfahren flexibel eingesetzt werden können, um komplexe Produktportfolios zu analysieren. Bestehende Ansätze aus der Wissenschaft implementieren einen Anwendungsfall für eine spezifische Problemstellung und gehen lediglich auf die dafür erforderlichen Daten ein. Im eingeführten Framework wird dagegen ein allgemeines Geschäftsverständnis bereitgestellt, welches Wissensbedarfe und Anwendungsfälle den einzelnen Phasen des Prozesses zur Analyse und Anpassung komplexer Produktportfolios zuordnet. Für die Auswahl der Wissensbedarfe werden Bewertungskriterien definiert. Eine zusätzliche allgemeine Beschreibung der relevanten Daten bildet die Grundlage für die Implementierung unterschiedlicher Anwendungsfälle zur Analyse komplexer Produktportfolios.

Die bisherigen Ansätze gehen zudem kaum auf die Schritte zur Datenvorbereitung ein und setzen diese nicht in Beziehung zu den verwendeten Daten und Analyseverfahren. Außerdem wird nicht auf unterschiedliche Algorithmen, Evaluationskriterien und Einsatzmöglichkeiten der Machine Learning Modelle eingegangen. Im Gegensatz dazu werden im Framework die Produktportfoliodaten und die Datenanalyseverfahren mit den Schritten zur Datenvorbereitung verknüpft. Ebenfalls stellt das Framework unterschiedliche Algorithmen gegenüber und beschreibt deren Evaluation und die Ermittlung von Ausreißern. Darüber hinaus werden die Möglichkeiten der unterschiedlichen Machine Learning Modelle zur Wissensgenerierung über komplexe Produktportfolios erläutert. Dadurch ermöglicht das Framework eine unternehmensspezifische Implementierung unterschiedlicher Anwendungsfälle.

7.3 Ergebnisbeitrag für die Industrie

Das Produktportfolio- und Variantenmanagement in Unternehmen ist geprägt durch manuelle und erfahrungsbasierte Tätigkeiten. Aufgrund der Vielzahl an Einflussfaktoren können einzelne Personen die komplexen Produktportfolios nicht mehr handhaben und benötigen Unterstützung. Durch das Framework können Unternehmen in Abhängigkeit ihrer Bedürfnisse und Voraussetzungen unterschiedliche Machine Learning Verfahren implementieren. Es wird ein Ausgangspunkt durch die Bereitstellung von Wissensbedarfen gegeben, welche von den Unternehmen erweitert und individuell angepasst werden können. Darüber hinaus werden die erforderlichen Daten

beschrieben und eine Hilfestellung bei der Implementierung durch die Berücksichtigung einer Auswahl unterschiedlicher Verfahren geboten.

Der Mehrwert von Machine Learning für Industrieunternehmen wurde durch die Fallstudie und die darin implementierten Anwendungsfälle deutlich. Es konnten Produkteigenschaften unter geringerem Ressourceneinsatz, schneller und genauer vorhergesagt werden. Zudem konnten Ähnlichkeiten zwischen Produktvarianten objektiv bestimmt werden, für die aktuell in Unternehmen lediglich das Expertenwissen zur Verfügung steht. Mit dem letzten Anwendungsfall wurde gezeigt, wie automatisiert Korrelationen im Produktportfolio ermittelt und auf Basis der Kundenbedürfnisse Einschränkungen des Produktportfolios definiert werden können.

8 Zusammenfassung und Ausblick

Dieses Kapitel fasst das Forschungsvorhaben und die darin erzielten Ergebnisse zusammen. Darüber hinaus wird ein Ausblick auf die Inhalte zukünftiger Forschungsaktivitäten gegeben.

8.1 Zusammenfassung

Unternehmen bieten aufgrund der Nachfrage nach individualisierten Produkten eine immer größere Anzahl an Varianten am Markt an. Dadurch steigt die interne Varianz in den Produkten und Prozessen immer weiter. Die aktuellen erfahrungs- und regelbasierten Verfahren können die komplexen Produktportfolios nicht handhaben, weshalb intelligente und datengetriebene Lösungen im Produktportfolio- und Variantenmanagement nötig sind.

Die **Zielsetzung** dieses Forschungsvorhaben war die Entwicklung eines Frameworks zur systematischen Generierung von Wissen für die Analyse komplexer Produktportfolios mittels Machine Learning. Dadurch soll ein Beitrag zur übergeordneten Vision die Produktportfoliokomplexität in Industrieunternehmen besser handhabbar zu machen geleistet werden. Für die Erreichung der Zielsetzung wurde ein **forschungsmethodisches Vorgehen** angelehnt an die „Design Research Methodology“ Typ 5 verfolgt. Zuerst wurden die begrifflichen und methodischen Grundlagen über komplexe Produktportfolios und Machine Learning erarbeitet. Es wurden die wesentlichen Ansätze zur Strukturierung sowie der Prozess zur Analyse und Anpassung **komplexer Produktportfolios** vorgestellt. Der Fokus der Arbeit liegt dabei auf den Entscheidungen über Produktvarianten auf der Ebene der operativen Produktportfoliogestaltung. Um ein Verständnis für **Machine Learning** zu schaffen, wurden die Generierung von Wissen aus Daten und Datenanalyseprozesse diskutiert. Diese Arbeit orientiert sich dabei am CRISP-DM, da dieser auf den industriellen Einsatz von Datenanalysen ausgerichtet ist. Zusätzlich wurden die Verfahren und Algorithmen beschrieben, welche im Framework eingesetzt werden. Der Fokus des Forschungsvorhabens wurde dabei auf das überwachte und unüberwachte Lernen gelegt. Anschließend wurde der **Stand der Forschung** betrachtet, indem die bisherigen Ansätze aus der Literatur, welche Machine Learning in der operativen Produktportfoliogestaltung einsetzen, untersucht und zuvor definierten Kriterien verglichen wurden. Auf Basis des dadurch identifizierte Forschungsbedarfs wurden die Kriterien für die Entwicklung des Frameworks abgeleitet.

Das **Framework** zur Analyse komplexer Produktportfolios besteht aus drei Bausteinen, welche aufeinander aufbauen und sich an der Struktur des CRISP-DM orientieren. **Baustein 1** beschreibt die Wissensbedarfe zur Analyse komplexer Produktportfolios sowie Kriterien für deren Bewertung und Auswahl. Diese basieren auf Anwendungsfällen aus Literatur und Industrie, welche gruppiert und den einzelnen Phasen des Entscheidungsprozesses zur Analyse und Anpassung komplexer Produktportfolio zugeordnet

sind. **Baustein 2** enthält eine datenbasierte Beschreibung komplexer Produktportfolios. Zentraler Bestandteil ist das Produktdatenmodell, welches die Produktarchitektur beinhaltet. Die Vertriebsdaten enthalten Informationen über die auf Basis des Produktdatenmodells konfigurierten und verkauften Produktvarianten sowie deren Eigenschaften. Einblicke über den tatsächlichen Gebrauch der Produktvarianten wird in den Nutzungsdaten gespeichert. In **Baustein 3** wird auf die systematische Generierung und den Einsatz des Wissens mittels Machine Learning und den zuvor ermittelten Daten eingegangen. Dieser Baustein ist weiter unterteilt in fünf Unterbausteine. Im ersten Baustein liegt der Schwerpunkt auf der Datenvorbereitung in Abhängigkeit der Daten und des Machine Learning Verfahrens. Die restlichen Analysebausteine gehen auf die Auswahl und Evaluation der Algorithmen sowie auf den Einsatz der Machine Learning Modelle zur Analyse komplexer Produktportfolios ein. Es werden Regressions-, Klassifikations-, Cluster- und Assoziationsanalysen betrachtet.

Für die **Validierung** des Frameworks wurde zuerst eine Anwendungsvalidierung mit einer Fallstudie bei einem Nutzfahrzeughersteller und anschließend eine Erfolgsvalidierung mit Expertenbefragungen durchgeführt. In der **Anwendungsvalidierung** wurden zuerst im Rahmen des ersten Bausteins des Frameworks drei Wissensbedarfe unter Verwendung der definierten Bewertungskriterien für die Implementierung ausgewählt. Diese waren die Ermittlung von „marktspezifischen und technischen Eigenschaften der Produktvarianten“, „Ähnlichkeiten von Produktvarianten“ und „Korrelationen zwischen Merkmalsausprägungen“. In Baustein 2 wurden die realen Daten beim Industriepartner ermittelt, beschafft und beschrieben. Die Implementierung der Anwendungsfälle mit realen Daten beim Industriepartner fand im dritten Baustein statt. Dadurch konnte die Anwendbarkeit des Frameworks bestätigt, der Nutzen von Machine Learning demonstriert und ein Mehrwert für das Industrieunternehmen geschaffen werden. In der **Erfolgsvalidierung** wurden das Framework sowie die Ergebnisse der Fallstudie neun Experten im Produktportfolio- und Variantenmanagement vorgestellt und die Erfüllung der definierten Kriterien bewertet. Diese bestätigten, dass das Framework zum einen die inhaltlichen Anforderungen und zum anderen die formalen Anforderungen erfüllt und dadurch einen Beitrag zur Analyse komplexer Produktportfolios aus Sicht der Wissenschaft und Industrie leistet.

8.2 Ausblick

Durch das Framework zur Analyse komplexer Produktportfolios konnte die Zielsetzung dieser Arbeit erreicht werden. Nichtsdestotrotz existieren zahlreiche Möglichkeiten das Framework weiterzuentwickeln und dessen Bausteine weiter zu detaillieren.

Mit dem Framework können komplexe Produktportfolios für die operative Produktportfoliogestaltung auf der Ebene der Merkmale und Komponenten analysiert werden. Das Ziel folgender Forschungsprojekte sollte demnach die Einbindung von Betrachtungen für die **Produktgestaltung** auf der Ebene der Stücklistenelemente sein.

Dadurch können konkrete technische Aspekte in die Analysen einbezogen werden. Ebenfalls sollten Gesichtspunkte der **strategischen Produktportfoliogestaltung** auf der Ebene von Produktfamilien in das Framework aufgenommen werden. Dadurch können strategische Entscheidungen über die Ausrichtung des Produktportfolios mit dem Wissen aus Daten unterstützt werden. Das Framework beschreibt Verfahren zur Datenvorbereitung, Algorithmen und Evaluationskriterien. Weitere Forschungstätigkeiten sollten die Überführung dieser in eine **Toolbox** mit Bausteinen, welche aus Programmcode bestehen und miteinander verknüpft werden können, beinhalten. Dadurch können Produktportfolio- und Variantenmanager befähigt werden, eigenständig Machine Learning Anwendungsfälle zu implementieren.

Das Framework berücksichtigt das Produktdatenmodell, die Vertriebsdaten und die Nutzungsdaten und vermittelt dadurch ein Verständnis für die wichtigsten Daten im Produktportfolio- und Variantenmanagement. In Unternehmen existieren dennoch weitere Daten, welche Rückschlüsse auf einzelne Produktvarianten zulassen. Aus diesem Grund sind **zusätzliche Daten** wie z.B. Produktions- oder Logistikdaten in das Framework zu integrieren. In gleicher Weise können **weitere Machine Learning Verfahren** in das Framework aufgenommen werden. Aktuell werden die Regressions-, Klassifikations-, Cluster- und Assoziationsanalyse berücksichtigt. Verfahren des Natural Language Processing oder generative Machine Learning Verfahren sind auf deren Potenziale im Produktportfolio- und Variantenmanagement in zukünftigen Forschungstätigkeiten zu untersuchen. Durch die Anwendung des Frameworks bei einem Industriepartner konnten drei Anwendungsfälle implementiert und der Nutzen von Machine Learning demonstriert werden. Das Produktportfolio von Unternehmen ist jedoch im stetigen Wandel. Zum Beispiel gibt es in der industriellen Praxis regelmäßig neue Produktgenerationen, welche grundlegende Änderungen des Produktdatenmodells zur Folge haben. Zukünftige Forschungstätigkeiten sollten daher Aspekte der **Wartung der Machine Learning Modelle** in das Framework integrieren. Dabei sollten Aspekte der Anpassung an Änderungen im Produktdatenmodell berücksichtigt werden.

Zusammenfassend wurde in diesem Promotionsvorhaben ein Framework zur Analyse komplexer Produktportfolios mittels Machine Learning entwickelt, angewendet und validiert. Das Framework ermöglicht es, Wissen aus Daten zu generieren und die operative Produktportfoliogestaltung automatisierter und objektiver zu gestalten. Dadurch wird ein Beitrag zur besseren Handhabung komplexer Produktportfolios geleistet.

9 Literaturverzeichnis

Abdelkafi, Nizar (2008): Variety induced complexity in mass customization: concepts and management. In: *35031102*.

Abdi, Hervé; Valentin, Dominique (2007): Multiple correspondence analysis. In: *Encyclopedia of measurement and statistics* 2 (4), S. 651–657.

Agard, Bruno; Kusiak, Andrew (2004a): Data-mining-based methodology for the design of product families. In: *International journal of production research* 42 (15), S. 2955–2969. DOI: 10.1080/00207540410001691929.

Agard, Bruno; Kusiak, Andrew (2004b): Standardization of components, products and processes with data mining. In: *International Conference on Production Research Americas*, S. 1–9.

Aggarwal, Charu C. (2018): Neural networks and deep learning. In: *Springer* 10, S. 973–978.

Agrawal, Rakesh; Imieliński, Tomasz; Swami, Arun (1993): Mining association rules between sets of items in large databases. In: Proceedings of the 1993 ACM SIGMOD international conference on Management of data, S. 207–216.

Agrawal, Rakesh; Srikant, Ramakrishnan (1994): Fast algorithms for mining association rules. In: Proc. 20th int. conf. very large data bases, VLDB, Bd. 1215: Citeseer, S. 487–499.

Agrawal, Rakesh; Srikant, Ramakrishnan (1995): Mining sequential patterns. In: Proceedings of the eleventh international conference on data engineering: IEEE, S. 3–14.

Aier, Stephan; Fischer, Christian (2011): Criteria of progress for information systems design theories. In: *Information Systems and E-Business Management* 9 (1), S. 133–172.

Ali, Peshawa Jamal Muhammad; Faraj, Rezhna H.; Koya, Erbil; Ali, Peshawa J. Muhammad (2014): Data Normalization and Standardization: A Technical Report. In: *Mach Learn Tech Rep* 1 (1), S. 1–6.

Alpaydin, Ethem (2020): Introduction to machine learning: MIT press.

Aman, Saima; Simmhan, Yogesh; Prasanna, Viktor K. (2014): Holistic measures for evaluating prediction models in smart grids. In: *IEEE Transactions on Knowledge and Data Engineering* 27 (2), S. 475–488.

Backhaus, Klaus; Erichson, Bernd; Plinke, Wulff; Weiber, Rolf (2015): Multivariate Analysemethoden. Eine anwendungsorientierte Einführung. 14. Aufl. 2016. Berlin, Heidelberg: Springer Berlin Heidelberg. Online verfügbar unter http://link.springer.com/10.1007/978-3-662-46076-4.

Bandemer, Hans; Näther, Wolfgang (2012): Fuzzy data analysis: Springer Science & Business Media (20).

Bannasch, Fabian; Bouché, Florian (2016): Finding the true cost of portfolio complexity. In: *The McKinsey Quarterly*.

Bartuschat, Martin (1995): Beitrag zur Beherrschung der Variantenvielfalt in der Serienfertigung: Vulkan-Verlag.

Battistello, Loris; Haug, Anders; Trattner, Alexandria; Hvam, Lars (2021): A classification of barriers to product variety reduction. In: *CIRP Journal of Manufacturing Science and Technology* 35, S. 517–525.

Baumberger, Georg Christoph (2007): Methoden zur kundenspezifischen Produktdefinition bei individualisierten Produkten.

Baumgart, Inka Martine (2005): Modularisierung von Produkten im Anlagenbau. Zugl.: Aachen, Techn. Hochsch., Diss., 2004. 1. Aufl. Aachen: Mainz.

Bertoni, A.; Larsson, T.; Larsson, J.; Elfsberg, J. (2017): Mining data to design value: A demonstrator in early design. In: *Proceedings of the International Conference on Engineering Design, ICED* 7 (DS87-7), S. 21–29.

Bex, T. (2021): Comprehensive Guide to Multiclass Classification Metrics. In: *Towards Data Science*, S. 1–19. Online verfügbar unter https://towardsdatascience.com/comprehensive-guide-on-multiclass-classification-metrics-af94cfb83fbd, zuletzt geprüft am 23.01.2023.

Biau, Gérard (2012): Analysis of a random forests model. In: *The Journal of Machine Learning Research* 13 (1), S. 1063–1095.

Binz, Hansgeorg; Roth, Daniel; Laukemann, Alexander (2016): Wissensmanagement. In: Handbuch Produktentwicklung, S. 247–274.

Birkhöfer, H.; Kloberdanz, H.; Berger, B.; Sauer, T. (2002): Why Methods Don't Work and How To Get Them Work. In: *Engineering Design in Integrated Product Development*, S. 29–36.

Blase, P.; DiFilippo, D.; Feindt, M.; Yager, F. (2016): Data-driven: Big decisions in the intelligence age. In: *PwC's Global Data and Analytics Survey: Big Decisions*.

Blees, Christoph (2011): Eine Methode zur Entwicklung modularer Produktfamilien. In: *Technische Universität Hamburg Harburg, Institut für Konstruktion und Produktionstechnik*.

Blessing, Lucienne T.M.; Chakrabarti, Amaresh (2009): DRM, a Design Research Methodology. London: Springer London.

Bonaccorso, Giuseppe (2018): Mastering machine learning algorithms: expert techniques to implement popular machine learning algorithms and fine-tune your models: Packt Publishing Ltd.

Bongulielmi, Luca; Henseler, Patrick.; Puls, Christoph; Meier, Markus (2001): The K- & V-matrix method - An approach in analysis and description of variant products. In: International Conference on Engineering Design (ICED 2001), Glasgow.

Botchkarev, Alexei (2018): Performance metrics (error measures) in machine learning regression, forecasting and prognostics: Properties and typology. In: *arXiv preprint arXiv:1809.03006*.

Boyarkin, G. N.; Revina, I. V.; Shevelyova, O. G. (2019): Forecasting the Price of Products Using Neural Networks. In: *IOP Conference Series: Materials Science and Engineering* 582 (1), S. 12005. DOI: 10.1088/1757-899X/582/1/012005.

Braun, Felix (2021): Application of algorithm-based validation tools for the validation of complex, multi-variant products.

Braun, Felix; Kreimeyer, Matthias; Kopal, Bastian; Paetzold, Kristin (2017): Herausforderungen in der Validierung der Variantenbeschreibung komplexer Produkte. In: *DFX 2017: Proceedings of the 28th Symposium Design for X*, S. 61–73.

Braun, Felix; Kreimeyer, Matthias; Paetzold, Kristin (2018): Procedural model to ensure consistency and validity of complex, variant-oriented product portfolios. In: *Proceedings of NordDesign: Design in the Era of Digitalization, NordDesign 2018*.

Breiman, Leo (2001): Random forests. In: *Machine learning* 45 (1), S. 5–32.

Bühne, Stan; Lauenroth, Kim; Pohl, Klaus (2004): Why is it not sufficient to model requirements variability with feature models. In: Proceedings of Workshop: Automotive Requirements Engineering: Citeseer, S. 5–12.

Burkov, Andriy (2019): Machine Learning kompakt: Alles, was Sie wissen müssen: MITP-Verlags GmbH & Co. KG.

Cardozo, Richard N.; Smith, David K. (1983): Applying financial portfolio theory to product portfolio decisions: An empirical study. In: *Journal of Marketing* 47 (2), S. 110–119.

Carnegie Bosch Institute (CBI) (1995): Knowledge in international corporations outline of research area: CBI Pittsburg.

Chan, Kit Yan; Kwong, C. K.; Hu, B. Q. (2012): Market segmentation and ideal point identification for new product design using fuzzy data compression and fuzzy clustering methods. In: *Applied Soft Computing Journal* 12 (4), S. 1371–1378. DOI: 10.1016/j.asoc.2011.11.026.

Chatfield, Chris (1978): The Holt-winters forecasting procedure. In: *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 27 (3), S. 264–279.

Che, Dunren; Safran, Mejdl; Peng, Zhiyong (2013): From big data to big data mining: Challenges, issues, and opportunities. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 7827 LNCS. Berlin, Heidelberg: Springer, S. 1–15.

Contreras, Pedro; Murtagh, Fionn (2015): Hierarchical clustering. In: *Handbook of Cluster Analysis; Henning, C., Meila, M., Murtagh, F., Rocci, R., Eds*, S. 103–123.

Cooper, Robert G.; Edgett, Scott J.; Kleinschmidt, Elko J. (2000): New product portfolio management: Practices and performance. In: *IEEE Engineering Management Review* 28 (1), S. 13–29. DOI: 10.1111/1540-5885.1640333.

Cortes, Corinna; Vapnik, Vladimir (1995): Support-vector networks. In: *Machine learning* 20 (3), S. 273–297.

Cristianini, Nello; Shawe-Taylor, John (2000): An Introduction to Support Vector Machines and Other Kernel-based Learning Methods: Cambridge University Press. Online verfügbar unter https://www.cambridge.org/core/product/identifier/9780511801389/type/book.

Cunningham, Pádraig; Cord, Matthieu; Delany, Sarah Jane (2008): Supervised learning. In: Machine learning techniques for multimedia: Springer, S. 21–49.

Darrell, K. Rigby (2017): Management tools 2017: An executive's guide. In: *Boston: Bain & Company, Retrieved from*.

Davies, David L.; Bouldin, Donald W. (1979): A cluster separation measure. In: *IEEE transactions on pattern analysis and machine intelligence* (2), S. 224–227.

Dellanoi, Richard (2006): Kommunalitäten bei der Entwicklung variantenreicher Produktfamilien.

Dempster, Arthur P.; Laird, Nan M.; Rubin, Donald B. (1977): Maximum likelihood from incomplete data via the EM algorithm. In: *Journal of the Royal Statistical Society: Series B (Methodological)* 39 (1), S. 1–22.

Denoeux, Thierry; Masson, Marie-Hélène (2004): Principal component analysis of fuzzy data using autoassociative neural networks. In: *IEEE Transactions on Fuzzy Systems* 12 (3), S. 336–349.

DIN 199-1 (2002): Technische Produktdokumentation-CAD-Modelle, Zeichnungen und Stücklisten-Teil 1: Begriffe: Beuth Berlin.

Disselkamp, Marcus (2015): Innovationsmanagement: Instrumente und Methoden zur Umsetzung im Unternehmen: Springer-Verlag.

Eckert, Claudia; Isaksson, Ola; Eckert, Calandra; Coeckelbergh, Mark; Hagström, Malin Hane (2020): Data Fairy in Engineering Land: The Magic of Data Analysis as a Sociotechnical Process in Engineering Companies. In: *Journal of*

Mechanical Design, Transactions of the ASME 142 (12). DOI: 10.1115/1.4047813.

Ehrlenspiel, Klaus; Kiewert, Alfons; Lindemann, Udo; Mörtl, Markus (1998): Kostengünstig entwickeln und konstruieren: Springer.

ElMaraghy, H.; Schuh, G.; Elmaraghy, W.; Piller, F.; Schönsleben, P.; Tseng, M.; Bernard, A. (2013): Product variety management. In: *CIRP Annals - Manufacturing Technology* 62 (2), S. 629–652. DOI: 10.1016/j.cirp.2013.05.007.

Ertel, Wolfgang (2011): Introduction to Artificial Intelligence Series editor.

Ester, Martin; Kriegel, Hans-Peter; Sander, Jörg; Xu, Xiaowei (1996): A density-based algorithm for discovering clusters in large spatial databases with noise. In: kdd, Bd. 96, S. 226–231.

Ester, Martin; Sander, Jörg (2000): Knowledge Discovery in Databases. Berlin, Heidelberg: Springer Berlin Heidelberg. Online verfügbar unter http://link.springer.com/10.1007/978-3-642-58331-5.

Eye, Alexander von; Clogg, Clifford C. (1996): Categorical variables in developmental research: Methods of analysis: Elsevier.

Fahrmeir, Ludwig; Künstler, Rita; Pigeot, Iris; Tutz, Gerhard (2007): Der Weg zur Datenanalyse. In: *Aufl. Heidelberg*.

Fayyad, Usama; Piatetsky-Shapiro, Gregory; Smyth, Padhraic (1996): From data mining to knowledge discovery in databases. In: *AI Magazine* 17 (3), S. 37–53.

Feldhusen, Jörg; Grote, Karl-Heinrich (2013): Pahl/Beitz Konstruktionslehre: Springer-Verlag.

Fischer, Peter; Hofer, Peter (2011): Lexikon der Informatik. Hg. v. Springer-Verlag Berlin Heidelberg 2011: Springer Berlin, Heidelberg.

Flach, Peter (2012): Machine Learning: The Art and Science of Algorithms that Make Sense of Data. Cambridge: Cambridge University Press. Online verfügbar unter https://www.cambridge.org/core/books/machine-learning/621D3E616DF879E494B094CC93ED36A4.

Förg, Armin; Karrer-müller, Eva; Kreimeyer, Matthias (2016): 05 Produktarchitektur – Einordnung und Grundlagen. In: Handbuch Produktentwicklung, S. 99–109.

Franke, Hans-Joachim; Firchau, Norman L. (2000): Variantenmanagement in der Einzel-und Kleinserienfertigung. In: DFX 2000: Proceedings of the 11th Symposium on Design for X, Schnaittach/Erlangen, Germany, 12.-13.20. 2000.

Fricke, Ernst; Schulz, Armin P. (2005): Design for changeability (DfC): Principles to enable changes in systems throughout their entire lifecycle. In: *Systems Engineering* 8 (4), no-no.

Gambella, Claudio; Ghaddar, Bissan; Naoum-Sawaya, Joe (2021): Optimization problems for machine learning: A survey. In: *European Journal of Operational Research* 290 (3), S. 807–828. DOI: 10.1016/j.ejor.2020.08.045.

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

Garcia-Pedrajas, Nicolas; Ortiz-Boyer, Domingo (2006): Improving multiclass pattern recognition by the combination of two strategies. In: *IEEE transactions on pattern analysis and machine intelligence* 28 (6), S. 1001–1006.

Gartner (2022): Artificial Intelligence (AI). Online verfügbar unter https://www.gartner.com/en/information-technology/glossary/artificial-intelligence.

Gebhardt, Nicolas; Kruse, Moritz; Krause, Dieter; Lindemann, Udo (2016): Gleichteile-, Modul-und Plattformstrategie. In: Handbuch Produktentwicklung: Carl Hanser Verlag GmbH & Co. KG München, S. 111–149.

Gembrys, Sven-Norman (1998): Ein Modell zur Reduzierung der Variantenvielfalt in Produktionsunternehmen: Fraunhofer-Institut für Produktionsanlagen und Konstruktionstechnik, IPK Berlin.

Genuer, Robin; Poggi, Jean-Michel; Tuleau, Christine (2008): Random Forests: some methodological insights. In: *arXiv preprint arXiv:0811.3619*.

Géron, Aurélien (2017): Hands-on machine learning with Scikit-Learn and TensorFlow : concepts, tools, and techniques to build intelligent systems.

Girotra, Manisha; Nagpal, Kanika; Minocha, Saloni; Sharma, Neha (2013): Comparative survey on association rule mining algorithms. In: *International Journal of Computer Applications* 84 (10).

Goodfellow, Ian; Bengio, Yoshua; Courville, Aaron (2012): Deep Learning. In: *Foreign Affairs* 91 (5), S. 1689–1699.

Göpfert, Jan (1998): Modulare Produktentwicklung.

Gregor, Shirley; Hevner, Alan R. (2013): Positioning and presenting design science research for maximum impact. In: *MIS quarterly*, S. 337–355.

Greisel, Markus; Kissel, Maximilian; Spinola, Benjamin; Kreimeyer, Matthias (2013): Design for adaptability in multi-variant product families. In: *Proceedings of the International Conference on Engineering Design, ICED* 4 DS75-04 (August), S. 179–188.

Grzymala-Busse, Jerzy W.; Grzymala-Busse, Witold J. (2009): Handling missing attribute values. In: Data mining and knowledge discovery handbook: Springer, S. 33–51.

Gupta, Neha (2013): Artificial neural network. In: *Network and Complex Systems* 3 (1), S. 24–28.

Han, Jiawei; Pei, Jian; Tong, Hanghang (2012): Data Mining: Concepts and Techniques.

Han, Jiawei; Pei, Jian; Yin, Yiwen (2000): Mining frequent patterns without candidate generation. In: *ACM sigmod record* 29 (2), S. 1–12.

Hancock, John T.; Khoshgoftaar, Taghi M. (2020): Survey on categorical data for neural networks. In: *Journal of Big Data* 7 (1), S. 1–41.

Haug, Anders; Hvam, Lars; Mortensen, Niels Henrik (2012): Definition and evaluation of product configurator development strategies. In: *Computers in Industry* 63 (5), S. 471–481. DOI: 10.1016/j.compind.2012.02.001.

Hearst, Marti A.; Dumais, Susan T.; Osuna, Edgar; Platt, John; Scholkopf, Bernhard (1998): Support vector machines. In: *IEEE Intelligent Systems and their applications* 13 (4), S. 18–28.

Heina, Jürgen (1999): Variantenmanagement: Springer-Verlag.

Helm, J. Matthew; Swiergosz, Andrew M.; Haeberle, Heather S.; Karnuta, Jaret M.; Schaffer, Jonathan L.; Krebs, Viktor E. et al. (2020): Machine learning and artificial intelligence: definitions, applications, and future directions. In: *Current reviews in musculoskeletal medicine* 13 (1), S. 69–76.

Helms, Bergen; Kissel, Maximilian (2016): Engineering Intelligence – Von der graphenbasierten Modellierung zur wissensbasierten Datenanalyse. In: Handbuch Produktentwicklung: Carl Hanser Verlag GmbH & Co. KG München, S. 979–1012.

Hennig, Christian; Meila, Marina (2015): Cluster analysis: an overview. In: *Handbook of Cluster Analysis*, S. 1–20.

Herrmann, Thorsten; Roth, Daniel; Binz, Hansgeorg (2018): Approach for Identifying and Initially Assessing Radical Product Ideas. In: *2018 IEEE International Conference on Engineering, Technology and Innovation, ICE/ITMC 2018 - Proceedings*. DOI: 10.1109/ICE.2018.8436353.

Hevner, Alan; Chatterjee, Samir (2010): Design Research in Information Systems. Boston, MA: Springer US (Integrated Series in Information Systems, 22). Online verfügbar unter http://link.springer.com/10.1007/978-1-4419-5653-8.

Hevner, Alan R.; March, Salvatore T.; Park, Jinsoo; Ram, Sudha (2004): Design science in information systems research. In: *MIS quarterly*, S. 75–105.

Hirose, Rogerio; Sodhi, Davinder; Thiel, Alexander (2017): Fighting portfolio complexity. In: *The McKinsey Quarterly*.

Hochdorffer, J.; Laule, C.; Lanza, G. (2018): Product variety management using data-mining methods — Reducing planning complexity by applying clustering analysis on product portfolios. In: *IEEE International Conference on Industrial Engineering and Engineering Management* 2017-Decem, S. 593–597. DOI: 10.1109/IEEM.2017.8289960.

Holler, Manuel; Stoeckli, Emanuel; Uebernickel, Falk; Brenner, Walter (2016): Towards understanding closed-loop PLM: The role of product usage data for product development enabled by intelligent properties.

Hou, Liang; Jiao, Roger J. (2020): Data-informed inverse design by product usage information: a review, framework and outlook. In: *Journal of Intelligent Manufacturing* 31, S. 529–552.

Hu, S. J.; Zhu, X.; Wang, H.; Koren, Y. (2008): Product variety and manufacturing complexity in assembly systems and supply chains. In: *CIRP Annals - Manufacturing Technology* 57 (1), S. 45–48. DOI: 10.1016/j.cirp.2008.03.138.

Hu, S. Jack (2013): Evolving paradigms of manufacturing: From mass production to mass customization and personalization. In: *Procedia CIRP* 7, S. 3–8.

Huang, Chun Che; Kusiak, Andrew (1998): Modularity in design of products and systems. In: *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans* 28 (1), S. 66–77. DOI: 10.1109/3468.650323.

Hung, Wen-Liang; Yang, Miin-Shen (2005): Fuzzy clustering on LR-type fuzzy numbers with an application in Taiwanese tea evaluation. In: *Fuzzy sets and systems* 150 (3), S. 561–577.

Hvam, Lars; Hansen, Christian Lindschou; Forza, Cipriano; Mortensen, Niels Henrik; Haug, Anders (2020): The reduction of product and process complexity based on the quantification of product complexity costs. In: *International journal of production research* 58 (2), S. 350–366.

Jank, Merle-Hendrikje (2021): Produktportfoliosteuerung mittels präskriptiver Datenanalyseverfahren. Aachen, GERMANY: Apprimus Wissenschaftsverlag. Online verfügbar unter http://ebookcentral.proquest.com/lib/unibwm/detail.action?docID=6533029.

Jeatrakul, P.; Wong, Kok Wai (2009): Comparing the performance of different neural networks for binary classification problems. In: 2009 Eighth International Symposium on Natural Language Processing: IEEE, S. 111–115.

Jeschke, Andrea (1997): Beitrag zur wirtschaftlichen Bewertung von Standardisierungsmaßnahmen in der Einzel-und Kleinserienfertigung durch die Konstruktion: Inst. für Konstruktionslehre, Maschinen-und Feinwerkelemente.

Jiao, J.; Zhang, L.; Zhang, Y.; Pokharel, S. (2008): Association rule mining for product and process variety mapping. In: *International Journal of Computer*

Integrated Manufacturing 21 (1), S. 111–124. DOI: 10.1080/09511920601182209.

Jiao, Jianxin; Zhang, Yiyang (2004): Product portfolio identification based on association rule mining. In: *CAD Computer Aided Design* 37 (2), S. 149–172. DOI: 10.1016/j.cad.2004.05.006.

Jonas, Henry (2013): Eine Methode zur strategischen Planung modularer Produktprogramme: Technische Universität Hamburg-Harburg.

Kantardzic, Mehmed (2011): Data mining: concepts, models, methods, and algorithms: John Wiley & Sons (3).

Kaufman, Leonard; Rousseeuw, Peter J. (2009): Finding groups in data: an introduction to cluster analysis: John Wiley & Sons.

Kerka, Friedrich; Kriegesmann, Bernd; Happich, Jan (2011): „Big Ideas “erkennen und Flops vermeiden-Instrumente zur stufenweisen Bewertung und Auswahl von Innovationsideen.

Kesper, Heiner (2012): Gestaltung von Produktvariantenspektren mittels matrixbasierter Methoden: Technische Universität München.

Khurana, Komal; Sharma, Simple (2013): A comparative analysis of association rule mining algorithms. In: *International Journal of Scientific and Research Publications* 3 (5), S. 0.

Kieckhäfer, Karsten (2013): Marktsimulation zur strategischen Planung von Produktportfolios: Dargestellt am Beispiel innovativer Antriebe in der Automobilindustrie: Springer-Verlag.

King, James R. (1980): Machine-component grouping in production flow analysis: an approach using a rank order clustering algorithm. In: *International journal of production research* 18 (2), S. 213–232.

Kipp, Thomas (2012): Methodische Unterstützung der variantengerechten Produktgestaltung.

Kira, Kenji; Rendell, Larry A. (1992): A practical approach to feature selection. In: Machine learning proceedings 1992: Elsevier, S. 249–256.

Kiritsis, Dimitris; Bufardi, Ahmed; Xirouchakis, Paul (2003): Research issues on product lifecycle management and information tracking using smart embedded systems. In: *Advanced engineering informatics* 17 (3-4), S. 189–202.

Kissel, Maximilian Philipp (2014): Mustererkennung in komplexen Produktportfolios.

Kononenko, Igor (1994): Estimating attributes: Analysis and extensions of RELIEF. In: European conference on machine learning: Springer, S. 171–182.

Kotsiantis, Sotiris B. (2013): Decision trees: a recent overview. In: *Artificial Intelligence Review* 39 (4), S. 261–283.

Kramer, Oliver (2013): K-nearest neighbors. In: *Dimensionality reduction with unsupervised nearest neighbors*, S. 13–23.

Krause, Dieter; Gebhardt, Nicolas (2018): Methoden zur Entwicklung modularer Produktfamilien (№3).

Krause, Dieter; Vietor, Thomas; Inkermann, David; Hanna, Michael; Richter, Timo; Wortmann, Nadine (2021): Produktarchitektur. In: Pahl/Beitz Konstruktionslehre: Springer, S. 335–393.

Kreimeyer, M.; Baumberger, C.; Deubzer, F.; Ziethen, D. (2016): An integrated product information model for variant design in commercial vehicle development. In: *Proceedings of International Design Conference, DESIGN* DS 84 (1), S. 707–716.

Kreimeyer, M.; Förg, A.; Lienkamp, M. (2013a): Mehrstufige modulorientierte Baukastenentwicklung für Nutzfahrzeuge. In: *VDI-Berichte* (2186), S. 99–112.

Kreimeyer, Matthias (2012): A product model to support plm-based variant planning and management. In: *Proceedings of International Design Conference, DESIGN* DS 70, S. 1741–1752.

Kreimeyer, Matthias; Förg, Armin; Lienkamp, Markus (2013b): Multi-level modular kit development for commercial vehicles. In: *VDI Fachtagung Nutzfahrzeuge 2013*, S. 99–112.

Kreimeyer, Matthias; Kindsmiller, Hubert; Landsherr, Thomas; Heintze, Wilhelm (2011): Systematische Planung der Produktarchitektur von Nutzfahrzeugen in den frühen Phasen der Entwicklung Systematic product architecture planning in the early phases of engineering design. In: *VDI Nutzfahrzeuge Tagung*.

Kreutzer, Ramon (2019): Methodik zur Bestimmung der Nutzenpotenziale von Felddaten cyber-physischer Systeme: Apprimus Wissenschaftsverlag.

Kubat, Miroslav (2021): An Introduction to Machine Learning.

Kumbhare, Trupti A.; Chobe, Santosh V. (2014): An overview of association rule mining algorithms. In: *International Journal of Computer Science and Information Technologies* 5 (1), S. 927–930.

Kusiak, Andrew; Smith, Mathew R.; Song, Zhe (2007): Planning product configurations based on sales data. In: *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews* 37 (4), S. 602–609. DOI: 10.1109/TSMCC.2007.897503.

L'Heureux, Alexandra; Grolinger, Katarina; Elyamany, Hany F.; Capretz, Miriam A.M. (2017): Machine Learning with Big Data: Challenges and Approaches. In: *IEEE Access* 5, S. 7776–7797. DOI: 10.1109/ACCESS.2017.2696365.

LeCun, Yann; Bengio, Yoshua; Hinton, Geoffrey (2015): Deep learning. In: *nature* 521 (7553), S. 436–444.

Lee, Wei-Meng (2019): Python machine learning: John Wiley & Sons.

Lehmann, Frank H.; Grzegorski, Andreas (2008): Anlaufmanagement in der Nutzfahrzeugindustrie am Beispiel Daimler Trucks. In: Anlaufmanagement in der Automobilindustrie erfolgreich umsetzen: Springer, S. 81–90.

Li, Zhi-Chao; He, Pi-Lian; Lei, Ming (2005): A high efficient AprioriTid algorithm for mining association rule. In: 2005 international conference on machine learning and cybernetics, Bd. 3: IEEE, S. 1812–1815.

Lindemann, Udo (2009): Methodische Entwicklung technischer Produkte.

Lingnau, Volker (1994): Variantenmanagement: Produktionsplanung im Rahmen einer Produktdifferenzierungsstrategie: Erich Schmidt Verlag GmbH & Co KG (58).

Ma, Jungmok; Kim, Harrison M. (2014): Continuous preference trend mining for optimal product design with multiple profit cycles. In: *Journal of Mechanical Design, Transactions of the ASME* 136 (6), S. 1–14. DOI: 10.1115/1.4026937.

Ma, Jungmok; Kim, Harrison M. (2016): Product family architecture design with predictive, data-driven product family design method. In: *Research in Engineering Design* 27 (1), S. 5–21. DOI: 10.1007/s00163-015-0201-4.

Ma, Jungmok; Kwak, Minjung; Kim, Harrison M. (2014): Demand trend mining for predictive life cycle design. In: *Journal of Cleaner Production* 68, S. 189–199. DOI: 10.1016/j.jclepro.2014.01.026.

Machi, Lawrence A.; McEvoy, Brenda T. (2016): The Literature Review - The Six Steps To Success. In: *2706-6495* 3. Online verfügbar unter https://books.google.com/books?hl=en&lr=&id=d3uzDAAAQBAJ&oi=fnd&pg=PP1&dq=review+sharing+economy&ots=Jr8iZxEO7B&sig=Nvh2pY8OWR-VaR-EvrQWOP0I1suI.

Mahesh, Batta (2020): Machine learning algorithms-a review. In: *International Journal of Science and Research (IJSR).[Internet]* 9, S. 381–386.

MAN Truck & Bus SE (2019): Annual report 2019. München.

March, Salvatore T.; Smith, Gerald F. (1995): Design and natural science research on information technology. In: *Decision support systems* 15 (4), S. 251–266.

Mariotti, John L. (2007): The Complexity Crisis: Why too many products, markets, and customers are crippling your company-and what to do about it: Simon and Schuster.

Marsland, Stephen (2011): Machine learning: an algorithmic perspective: Chapman and Hall/CRC.

Marx, Stefan (1996): Portfolio-Theorie. In: Aktienprognosen zur Portfolio-Optimierung: Springer, S. 49–79.

Mayring, Philipp (2010): Qualitative Inhaltsanalyse. Grundlagen und Techniken. Beltz: Weinheim.

McKay, Alison; Erens, Frederick; Bloor, M. Susan (1996): Relating product definition and product variety. In: *Research in Engineering Design* 8, S. 63–80.

Meffert, Heribert; Burmann, Christoph; Kirchgeorg, Manfred (2015): Marketing. Wiesbaden: Springer Fachmedien Wiesbaden. Online verfügbar unter http://link.springer.com/10.1007/978-3-658-02344-7.

Mehlstäubl, Jan; Braun, Felix; Denk, Martin; Kraul, Ralf; Paetzold, Kristin (2022a): Using Machine Learning for Product Portfolio Management: A Methodical Approach to Predict Values of Product Attributes for Multi-Variant Product Portfolios. In: *Proceedings of the Design Society* 2, S. 1659–1668. DOI: 10.1017/pds.2022.168.

Mehlstäubl, Jan; Braun, Felix; Gadzo, Emir; Paetzold, Kristin (2023a): Machine Learning to generate Knowledge for Decision-making Processes in Product Portfolio and Variety Management. In: 9th International Conference on Research Into Design.

Mehlstäubl, Jan; Braun, Felix; Paetzold, Kristin (2021a): Artificial Intelligence in Product Portfolio and Variety Management in Commercial Vehicle Industry – An Overview about Expectations, Challenges and Use Cases. In: Presentation on the prostep ivip Symposium 2021.

Mehlstäubl, Jan; Braun, Felix; Paetzold, Kristin (2021b): Data Mining in Product Portfolio and Variety Management – Literature Review on Use Cases and Research Potentials. In: 2021 IEEE Technology & Engineering Management Conference-Europe (TEMSCON-EUR), S. 442–447.

Mehlstäubl, Jan; Braun, Felix; Paetzold-Byhain, Kristin (2023b): Reduktion komplexer Produktportfolios durch die Ableitung von Kombinatorikregeln aus verkauften Produktkonfigurationen mit einer Assoziationsanalyse. In: Stuttgarter Symposium für Produktentwicklung.

Mehlstäubl, Jan; Gadzo, Emir; Atzberger, Alexander; Paetzold, Kristin (2022b): Herausforderungen datengetriebener Methoden in der

Produktentwicklung/Challenges of data-driven methods in product development. In: *Konstruktion* 74 (06), S. 60–66. DOI: 10.37544/0720-5953-2022-06-60.

Mehlstäubl, Jan; Nicklas, Simon; Gerschütz, Benjamin; Sprogies, Nicolai; Schleich, Benjamin; Lohner, Thomas et al. (2021c): Voraussetzungen für den Einsatz datengetriebener Methoden in der Produktentwicklung. In: Proceedings of the 32nd Symposium Design for X (DFX2021).

Mehlstäubl, Jan; Pfeiffer, Christoph; Kraul, Ralf; Braun, Felix; Paetzold-Byhain, Kristin (2023c): Methodical approach to cluster configurations of product variants of complex product portfolios. In: 24th International Conference on Engineering Design (ICED).

Messerle, Mathias (2016): Methodik zur Identifizierung der erfolgversprechendsten Produktideen in den frühen Phasen des Produktentwicklungsprozesses. Online verfügbar unter https://elib.uni-stuttgart.de/handle/11682/8937.

Meyer, M. H.; Lehnerd, Alvin P. (1997): The Power of Product Platforms: Building Value and Cost Leadership, 1997. In: *New York, NY* 10020, S. 39.

Meyer, Maurice; Panzner, Melina; Koldewey, Christian; Dumitrescu, Roman (2022): 17 Use Cases for Analyzing Use Phase Data in Product Planning of Manufacturing Companies. In: *Procedia CIRP* 107, S. 1053–1058.

Mirkin, Boris (2016): Quadratic error and k-means. In: *Handbook of Cluster Analysis*, S. 33–52.

Mishra, Abhishek (2020): Machine Learning for iOS Developers: Wiley. Online verfügbar unter https://online-library.wiley.com/doi/book/10.1002/9781119602927.

Mitchell, Tom M. (1997): Machine learning: McGraw-hill New York (1).

Mittal, Sanjay; Frayman, Felix (1989): Towards a Generic Model of Configuraton Tasks. In: IJCAI, Bd. 89: Citeseer, S. 1395–1401.

Moon, Seung Ki; Kumara, Soundar R.T.; Simpson, Timothy W. (2006): Data mining and fuzzy clustering to support product family design. In: *Proceedings of the ASME Design Engineering Technical Conference* 2006 (814), S. 1–9. DOI: 10.1115/detc2006-99287.

Moon, Seung Ki; Simpson, Timothy W.; Kumara, Soundar R.T. (2010): A methodology for knowledge discovery to support product family design. In: *Annals of Operations Research* 174 (1), S. 201–218. DOI: 10.1007/s10479-008-0349-7.

Müller, Roland M.; Lenz, Hans-Joachim (2013): Business Intelligence [Elektronische Ressource]. In: *0170-6012*.

Murphy, Kevin P. (2012): Machine learning: a probabilistic perspective: MIT press.

Murtagh, Fionn (2016): A Brief history of cluster analysis. In: *Handbook of Cluster Analysis*, S. 21–33.

Neis, Jan (2015): Analyse der Produktportfoliokomplexität unter Anwendung von Verfahren des Data Mining: Shaker Verlag.

Nonaka, Ikujirō; Takeuchi, Hirotaka (1997): Die Organisation des Wissens. Wie japanische Unternehmen eine brachliegende Ressource nutzbar machen. Frankfurt/Main: Campus-Verl.

North, Klaus (2016): Wissensorientierte Unternehmensführung: Wissensmanagement gestalten: Springer-Verlag.

Pahl, G.; Beitz, W.; Feldhusen, J.; Grote, K. H. (2005): Konstruktionslehre. Grundlagen erfolgreicher Produktentwicklung ; Methoden und Anwendung. 6. Aufl. Berlin, Heidelberg: Springer (Springer-Lehrbuch). Online verfügbar unter https://www.amazon.de/Pahl-Beitz-Konstruktionslehre-erfolgreicher-Produktentwicklung/dp/3540220488.

Panzner, Melina; Enzberg, Sebastian von; Meyer, Maurice; Dumitrescu, Roman (2022): Characterization of Usage Data with the Help of Data Classifications. In: *Journal of the Knowledge Economy*, S. 1–22.

Peffers, Ken; Rothenberger, Marcus; Tuunanen, Tuure; Vaezi, Reza (2012): Design science research evaluation. In: International Conference on Design Science Research in Information Systems: Springer, S. 398–410.

Pelleg, Dan; Moore, Andrew W. (2000): X-means: Extending k-means with efficient estimation of the number of clusters. In: Icml, Bd. 1, S. 727–734.

Pentreath, Nick (2015): Machine learning with spark: Packt Publishing Birmingham.

Pereira, Francisco Câmara; Borysov, Stanislav S. (2019): Machine Learning Fundamentals. In: Constantinos Antoniou, Loukas Dimitriou und Francisco Pereira (Hg.): Mobility Patterns, Big Data and Transport Analytics: Elsevier, S. 9–29. Online verfügbar unter https://www.sciencedirect.com/science/article/pii/B9780128129708000026.

Peterson, Leif E. (2009): K-nearest neighbor. In: *Scholarpedia* 4 (2), S. 1883.

Polanyi, Michael (1985): Implizites wissen: Suhrkamp.

Porter, Michael E.; Heppelmann, James E. (2014): How smart, connected products are transforming competition. In: *Harvard business review* 92 (11), S. 64–88.

Powers, David M. W. (2020): Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. In: *arXiv preprint arXiv:2010.16061*.

Prithiviraj, P.; Porkodi, R. (2015): A comparative analysis of association rule mining algorithms in data mining: a study. In: *Open J. Comput. Sci. Eng. Surv* 3 (1), S. 98–119.

Probst, Gilbert J. B.; Raub, Steffen P.; Romhardt, Kai (2010): Wissen managen. Wie Unternehmen ihre wertvollste Ressource optimal nutzen. 6., überarb. und erw. Aufl. Wiesbaden: Gabler.

Quinlan, J. Ross (1986): Induction of decision trees. In: *Machine learning* 1 (1), S. 81–106.

Rapp, Thomas (1999): Produktstrukturierung: Komplexitätsmanagement durch modulare Produktstrukturen und -plattformen.

Rathnow, Peter J. (1993): Integriertes Variantenmanagement: Bestimmung, Realisierung und Sicherung der optimalen Produktvielfalt: Vandenhoeck & Ruprecht (20).

Ray, Susmita (2019): A quick review of machine learning algorithms. In: 2019 International conference on machine learning, big data, cloud and parallel computing (COMITCon): IEEE, S. 35–39.

Riesener, M.; Dolle, C.; Schmitt, L.; Jank, M.-H. (2019a): Development of a Methodology to Design Product Portfolios in Accordance to Corporate Goals Using an Evolutionary Algorithm. In: *IEEE International Conference on Industrial Engineering and Engineering Management* 2019-Decem, S. 1466–1470. DOI: 10.1109/IEEM.2018.8607559.

Riesener, Michael; Dölle, Christian; Dierkes, Christopher; Jank, Merle Hendrikje (2020): Applying Supervised and Reinforcement Learning to Design Product Portfolios in Accordance with Corporate Goals. In: *Procedia CIRP* 91, S. 127–133. DOI: 10.1016/j.procir.2020.02.157.

Riesener, Michael; Dölle, Christian; Schuh, Guenther; Zhang, Wenjia; Jank, Merle Hendrikje (2019b): Implementing neural networks within portfolio management to support decision-making processes. In: *PICMET 2019 - Portland International Conference on Management of Engineering and Technology: Technology Management in the World of Intelligent Systems, Proceedings* 0, S. 1–7. DOI: 10.23919/PICMET.2019.8893760.

Rogers, Everett M.; Singhal, Arvind; Quinlan, Margaret M. (2014): Diffusion of innovations. In: An integrated approach to communication theory and research: Routledge, S. 432–448.

Romanowski, Carol J.; Nagi, Rakesh (2002): A data mining and graph theoretic approach to building generic bills of materials. In: *The 11th industrial engineering research conference*.

Romanowski, Carol J.; Nagi, Rakesh (2004): A data mining approach to forming generic bills of materials in support of variant design activities. In: *Journal of Computing and Information Science in Engineering* 4 (4), S. 316–328. DOI: 10.1115/1.1812556.

Romanowski, Carol J.; Nagi, Rakesh (2005): On comparing bills of materials: A similarity/distance measure for unordered trees. In: *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans* 35 (2), S. 249–260. DOI: 10.1109/TSMCA.2005.843395.

Rousseeuw, Peter J. (1987): Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. In: *Journal of computational and applied mathematics* 20, S. 53–65.

Ruder, Sebastian (2016): An overview of gradient descent optimization algorithms. In: *arXiv preprint arXiv:1609.04747*.

Russell, Stuart J. (2010): Artificial intelligence a modern approach: Pearson Education, Inc.

Sabin, Daniel; Weigel, Rainer (1998): Product configuration frameworks-a survey. In: *IEEE Intelligent Systems and their applications* 13 (4), S. 42–49.

Safavian, S. Rasoul; Landgrebe, David (1991): A survey of decision tree classifier methodology. In: *IEEE transactions on systems, man, and cybernetics* 21 (3), S. 660–674.

Samuel, A. L. (1959): Some Studies in Machine Learning Using the Game of Checkers. In: *IBM J. Res. & Dev.* 3 (3), S. 210–229. DOI: 10.1147/rd.33.0210.

Schmieder, Matthias; Thomas, Sven (2005): Plattformstrategien und Modularisierung in der Automobilentwicklung: Shaker.

Schuh, Günther (2005): Produktkomplexität managen.

Schuh, Günther (2012): Innovationsmanagement: Handbuch Produktion und Management 3: Springer-Verlag.

Schuh, Günther; Riesener, Michael (2018): Produktkomplexität managen. München: Carl Hanser Verlag GmbH & Co. KG.

Schuh, Günther; Riesener, Michael; Jank, Merle-Hendrikje (2018): Managing Customized and Profitable Product Portfolios Using Advanced Analytics. In: Customization 4.0, S. 203–216. Online verfügbar unter http://link.springer.com/10.1007/978-3-319-77556-2_13.

Schuh, Günther; Schwenk, Urs (2001): Produktkomplexität managen. Strategien - Methoden - Tools. München, Wien: Hanser.

Seber, George A. F.; Lee, Alan J. (2012): Linear regression analysis: John Wiley & Sons.

Sherman, Rick (2014): Business intelligence guidebook: From data integration to analytics: Newnes.

Singh, Amanpreet; Thakur, Narina; Sharma, Aakanksha (2016): A review of supervised machine learning algorithms. In: 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom): IEEE, S. 1310–1315.

Sivarajah, Uthayasankar; Kamal, Muhammad Mustafa; Irani, Zahir; Weerakkody, Vishanth (2017): Critical analysis of Big Data challenges and analytical methods. In: *Journal of Business Research* 70, S. 263–286. DOI: 10.1016/j.jbusres.2016.08.001.

Song, Z.; Kusiak, A. (2009): Optimising product configurations with a data-mining approach. In: *International journal of production research* 47 (7), S. 1733–1751. DOI: 10.1080/00207540701644235.

Stack Exchange (2020): Distinction between AI, ML, Neural Networks, Deep learning and Data mining. Online verfügbar unter https://softwareengineering.stackexchange.com/questions/366996/distincti%0Aon-between-ai-ml-neural-networks-deep-learning-and-data-mining.

Sutton, Richard S.; Barto, Andrew G. (2018): Reinforcement learning: An introduction: MIT press.

Tan, Pang-Ning; Steinbach, Michael; Kumar, Vipin; Karpatne, Anuj (2019): Introduction to Data Mining: Pearson Deutschland. Online verfügbar unter https://elibrary.pearson.de/book/99.150005/9780273775324.

Tensa, Melissa; Edmonds, Katherine; Ferrero, Vincenzo; Mikes, Alex; Soria Zurita, Nicolas; Stone, Rob; DuPont, Bryony (2019): Toward automated functional modeling: An association rules approach for mining the relationship between product components and function. In: Proceedings of the International Conference on Engineering Design, ICED, 2019-Augus: Cambridge University Press, S. 1713–1722.

TIBCO (2022): Was ist ein Random Forest? Online verfügbar unter https://www.tibco.com/de/reference-center/what-is-a-random-forest.

Tidstam, Anna; Malmqvist, Johan (2010): Information modelling for automotive configuration. In: Proceedings of NordDesign 2010, Göteborg, Sweden, S. 275–286.

Tucker, Conrad; Kim, Harrison M. (2011a): Predicting emerging product design trend by mining publicly available customer review data. In: *ICED 11 - 18th International Conference on Engineering Design - Impacting Society Through Engineering Design* 6 (August), S. 43–52.

Tucker, Conrad S. (2014): Quantifying the relevance of product feature classification in product family design. In: *Advances in Product Family and Product Platform Design: Methods and Applications*, S. 147–177. DOI: 10.1007/978-1-4614-7937-6_6.

Tucker, Conrad S.; Kim, Harrison M. (2008): Optimal product portfolio formulation by merging predictive data mining with multilevel optimization. In: *Journal of Mechanical Design, Transactions of the ASME* 130 (4). DOI: 10.1115/1.2838336.

Tucker, Conrad S.; Kim, Harrison M. (2009): Data-driven decision tree classification for product portfolio design optimization. In: *Journal of Computing and Information Science in Engineering* 9 (4), S. 1–14. DOI: 10.1115/1.3243634.

Tucker, Conrad S.; Kim, Harrison M. (2011b): Trend mining for predictive product design. In: *Journal of Mechanical Design, Transactions of the ASME* 133 (11), S. 1–11. DOI: 10.1115/1.4004987.

Tucker, Conrad S.; Kim, Harrison M.; Barker, Douglas E.; Zhang, Yuanhui (2010): A ReliefF attribute weighting and X-means clustering methodology for top-down product family optimization. In: *Engineering Optimization* 42 (7), S. 593–616. DOI: 10.1080/03052150903353328.

Tzokas, Nikolaos; Hultink, Erik Jan; Hart, Susan (2004): Navigating the new product development process. In: *Industrial Marketing Management* 33 (7), S. 619–626. DOI: 10.1016/j.indmarman.2003.09.004.

Ulrich, Karl (1995): The role of product architecture in the manufacturing firm. In: *Research Policy* 24, S. 419–440.

Vani, K. (2015): Comparative Analysis of Association Rule Mining Algorithms Based on Performance Survey. In: *International Journal of Computer Science and Information Technologies* 6 (4), S. 3980–3985.

VDI 5610 Blatt 1 (2009): Wissensmanagement im Ingenieurwesen Knowlegde management for engineering. In: *Deutsches Institut für Normung: Produktentwicklung und Konstruktion* (March), S. 1–28.

VDMA Bayern (2020): Leitfaden Künstliche Intelligenz – Potenziale und Umsetzungen im Mittelstand. In: *Online unter http://ki. vdma. org/documents/106096/53103997/VDMA% 2520Bayern_Leitfaden_KI_2020_1601889305004. pdf Search in*.

Vogel, Reinhard (1989): Der Prozess der Produktelimination aus entscheidungsorientierter Sicht: Deutsch.

Volkswagen AG (2019): Annual report 2019. Wolfsburg.

Wang, Chih Hsuan (2019): Association rule mining and cognitive pairwise rating based portfolio analysis for product family design. In: *Journal of Intelligent Manufacturing* 30 (4), S. 1911–1922. DOI: 10.1007/s10845-017-1362-y.

Weber, Christian (2005): CPM/PDD–an extended theoretical approach to modelling products and product development processes. In: proceedings of the 2nd German-Israeli symposium on advances in methods and systems for development of products and processes, Bd. 6: Fraunhofer-IRB-Verlag Stuttgart.

Weisberg, Sanford (2005): Applied linear regression: John Wiley & Sons (528).

Weiss, Sholom M.; Kulikowski, Casimir A. (1991): Computer systems that learn: classification and prediction methods from statistics, neural nets, machine learning, and expert systems: Morgan Kaufmann Publishers Inc.

Wilberg, Julian (2020): From data to value: facilitating strategy development for connected products: Dr. Hut. Online verfügbar unter https://mediatum.ub.tum.de/1468778.

Wilberg, Julian; Triep, Isabell; Hollauer, Christoph; Omer, Mayada (2017): Big Data in Product Development: Need for a data strategy. In: PICMET 2017 - Portland International Conference on Management of Engineering and Technology: Technology Management for the Interconnected World, Proceedings, 2017-Janua, S. 1–10.

Wildemann, H. (2011): Variantenmanagement: Leitfaden zur Komplexitätsreduzierung,-beherrschung und–vermeidung. In: *Transfer-Centrum, München*.

Wirth, Rüdiger; Hipp, Jochen (2000): CRISP-DM : Towards a Standard Process Model for Data Mining. In: Proceedings of the Fourth International Conference on the Practical Application of Knowledge Discovery and Data Mining, Bd. 1: Springer-Verlag London, UK, S. 29–39.

Xia, Shi Sheng; Wang, Li Ya (2010): Customer requirements mapping method based on association rules mining for mass customisation. In: *International Journal of Computer Applications in Technology* 37 (3-4), S. 198–203. DOI: 10.1504/IJCAT.2010.031935.

Yin, Robert K. (2014): Case study research. Design and methods. 5. edition. Los Angeles, London, New Delhi, Singapore, Washington, DC: SAGE (Applied social research methods series). Online verfügbar unter https://books.google.de/books?id=Cdk5DQAAQBAJ.

Ying, Xue (2019): An overview of overfitting and its solutions. In: Journal of physics: Conference series, Bd. 1168: IOP Publishing, S. 22022.

Yu, Li; Wang, Liya (2010): Product portfolio identification with data mining based on multi-objective GA. In: *Journal of Intelligent Manufacturing* 21 (6), S. 797–810. DOI: 10.1007/s10845-009-0255-0.

Yu, Li; Zhang, Zai Fang (2014): Trend Analysis of Product Function Using Sequential Pattern Mining. In: *Applied Mechanics and Materials* 519-520, S. 736–740. DOI: 10.4028/www.scientific.net/amm.519-520.736.

Yu, Li; Zhang, Zaifang; Shen, Jin (2017): Dynamic customer preference analysis for product portfolio identification using sequential pattern mining. In: *Industrial Management and Data Systems* 117 (2), S. 365–381. DOI: 10.1108/IMDS-12-2015-0496.

Zhang, Linda (2012): Identifying mapping relationships between functions and technologies with association rule mining. In: *International Journal of Computer Integrated Manufacturing* 25 (January 2013), S. 37–41.

Zhang, Yiyang; Jiao, Jianxin; Ma, Yongsheng (2007): Market segmentation for product family positioning based on fuzzy clustering. In: *Journal of Engineering Design* 18 (3), S. 227–241. DOI: 10.1080/09544820600752781.

Zhao, Qiankun; Bhowmick, Sourav S. (2003): Association rule mining: A survey. In: *Nanyang Technological University, Singapore* 135.

Zicari, Roberto V. (2014): Big Data: Challenges and Opportunities. In: Big data computing, S. 103–128.

Ziegler, Andreas; König, Inke R. (2014): Mining data with random forests: current options for real-world applications. In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 4 (1), S. 55–63.

Ziethen, Dieter R. (2007): CATIA V5: Baugruppen, Zeichnungen: Hanser.

10 Abbildungsverzeichnis

Abbildung 1-1 Aktuelles Vorgehen zur Analyse komplexer Produktportfolios bei einem Industriepartner....4
Abbildung 1-2: Gegenüberstellung mehrerer Merkmale eines komplexen Produktportfolios bei einem Industriepartner....4
Abbildung 1-3: Aufbau der Arbeit....7
Abbildung 2-1: Gliederung des Produktportfolios in Produktfamilien und Produktvarianten in Anlehnung an Kieckhäfer (2013)....10
Abbildung 2-2: Quersubventionierung von exotischen Produktvarianten in Anlehnung an Schuh und Schwenk (2001)....11
Abbildung 2-3: Produktportfolio- und Variantenmanagement nach Mehlstäubl et al. (2021b)....12
Abbildung 2-4: Produktarchitektur nach Göpfert (1998) und Krause et al. (2021)....13
Abbildung 2-5: Baukastenarchitektur nach Kreimeyer et al. (2013b)....14
Abbildung 2-6: Entscheidungsebenen und -gegenstände in Anlehnung an Kieckhäfer (2013) und Meffert et al. (2015)....16
Abbildung 2-7: Betrachtungsgegenstände der Entscheidungsfindung nach Heina (1999)....16
Abbildung 2-8: Entscheidungsprozess zur Analyse und Anpassung von Produktportfolios in Anlehnung an Vogel (1989), Gembrys (1998) und Mehlstäubl et al. (2023a)....17
Abbildung 2-9: Zusammenhang künstliche Intelligenz, Data Mining und Machine Learning in Anlehnung an Stack Exchange (2020)....19
Abbildung 2-10: Terminologie Machine Learning in Anlehnung an Flach (2012)....20
Abbildung 2-11 Traditioneller Ansatz vs. Machine Learning zur Wissensgenerierung in Anlehnung an Géron (2017) und Mehlstäubl et al. (2023a)....22
Abbildung 2-12: Knowledge Discovery in Databases (KDD) Prozess nach Fayyad et al. (1996)....23
Abbildung 2-13: CRISP-DM Prozess nach Wirth und Hipp (2000)....24
Abbildung 2-14: Arten des Machine Learning....26

Abbildung 2-15: Ein zweidimensionales lineares Regressionsmodell nach Mishra (2020) 27
Abbildung 2-16: Sigmoidfunktion der logistischen Regression nach Mishra (2020) ... 28
Abbildung 2-17: Zweidimensionale Support Vector Machine zur Klassifikation und Regression in Anlehnung an Cristianini und Shawe-Taylor (2000) 29
Abbildung 2-18: Zweidimensionale kNN-Klassifikation mit k=3 in Anlehnung an Peterson (2009) 30
Abbildung 2-19: Entscheidungsbaum in Anlehnung an Alpaydin (2020) 31
Abbildung 2-20: Aufbau eines Random Forest nach TIBCO (2022) 31
Abbildung 2-21: Struktur eines neuronalen Netzes nach Mishra (2020) 33
Abbildung 2-22: Distanzbasiertes Clustering mit k-Means Algorithmus in Anlehnung an Marsland (2011) 35
Abbildung 2-23: Hierarchisches Clustering mit Single-Linkage Algorithmus 36
Abbildung 2-24: Dichtebasiertes Clustering mit DBSCAN Algorithmus 36
Abbildung 2-25: Probabilistisches Clustering mit EM Algorithmus (elliptischer Clusterform) 37
Abbildung 2-26: Vorgehen AIS Algorithmus (min Support=2) in Anlehnung an Khurana und Sharma (2013) 39
Abbildung 2-27: Vorgehen Apriori Algorithmus (min Support=2) in Anlehnung an Khurana und Sharma (2013) 40
Abbildung 2-28: Bildung de FP-Growth Baums in Anlehnung an Han et al. (2000) 41
Abbildung 3-1: Optimierung der Produktportfoliogestaltung nach Tucker und Kim (2009) 45
Abbildung 3-2: Datenverarbeitung mit dem neuronalen Netz nach Boyarkin et al. (2019) 46
Abbildung 3-3: Unterstützung von Entscheidungsprozessen im Produktportfoliomanagement nach Riesener et al. (2019b) 47
Abbildung 3-4: Methode zur Bildung generischer GBOMs nach Romanowski und Nagi (2004) 48

Abbildung 3-5: Prozess zum Mining von Vertriebsdaten zur Ermittlung von Mustern in Unterbaugruppen und Hauptproduktkonfigurationen nach Song und Kusiak (2009) 49
Abbildung 3-6: Prozess zur Wissensentdeckung für die Unterstützung der Entwicklung von Produktfamilien nach Moon et al. (2010) 50
Abbildung 3-7 Top-down Optimierung von Produktfamilien 51
Abbildung 3-8: Fünf Schritte zur Identifikation der idealen Punkte für die Entwicklung neuer Produkte nach Chan et al. (2012) 51
Abbildung 3-9: Ansatz zur Analyse der Portfoliokomplexität unter Anwendung von Clustering in Anlehnung an Neis (2015) 52
Abbildung 3-10: Framework zur Umsetzung von Produktdifferenzierung und Produktkonfiguration nach Wang (2019) 53
Abbildung 3-11: Produktportfolioidentifikation mit sequentiellem Pattern Mining nach Yu et al. (2017) 53
Abbildung 4-1: Design Research Methodology (DRM) in Anlehnung an Blessing und Chakrabarti (2009) 59
Abbildung 4-2: Übersicht der Entwicklung des Frameworks 62
Abbildung 4-3: Einordnung des wissenschaftlichen Mehrwerts nach Gregor und Hevner (2013) 65
Abbildung 5-1: Framework zur Analyse komplexer Produktportfolios 66
Abbildung 5-2: Daten zur Analyse komplexer Produktportfolios 72
Abbildung 5-3: Produktkonfigurationsframework nach Tidstam und Malmqvist (2010) 73
Abbildung 5-4: Produktinformationsmodell in Anlehnung an Kreimeyer et al. (2016) 74
Abbildung 5-5: Produktkonfiguration und Produktkonfigurator in Anlehnung an Schuh und Riesener (2018) 75
Abbildung 5-6: Ordinale Kodierung 80
Abbildung 5-7: One-hot Kodierung in Anlehnung an Mehlstäubl et al. (2022a) 81
Abbildung 5-8: Graphische Darstellung der Regressionsergebnisse 85

Abbildung 5-9: Eigenschaftsoptimierung von Produktvarianten mit Regressionsmodellen 86
Abbildung 5-10: Ausschnitt eines Entscheidungsbaums am Beispiel der CO_2-Emission 86
Abbildung 5-11: Beispielhafte ROC nach Bex (2021) 90
Abbildung 5-12: Ellenbogenverfahren in Anlehnung an Géron (2017) 93
Abbildung 5-13: Identifikation des genauesten Clustering mit CVIs 94
Abbildung 5-14: Untersuchung der ermittelten Cluster 95
Abbildung 5-15: Reduktion des Produktportfolios durch die Einführung von Kombinatorikregeln nach Mehlstäubl et al. (2023b) 97
Abbildung 5-16: Vorgehen zur Anwendung des Frameworks 98
Abbildung 6-1: Architektur zur Prognose von Produkteigenschaften nach Mehlstäubl et al. (2022a) 103
Abbildung 6-2: Visualisierung der mit einem neuronalen Netz prognostizierten Fahrzeuggewichte 110
Abbildung 6-3: Visualisierung der mit dem Random Forest Algorithmus prognostizierten CO_2-Werte 111
Abbildung 6-4: Visualisierung der mit dem Random Forest prognostizierten Zahlungsbereitschaft 113
Abbildung 6-5: Architektur des Demonstrators 114
Abbildung 6-6: Einsatz der Regressionsmodelle zur Eigenschaftsvorhersage 114
Abbildung 6-7: Einsatz der Regressionsmodelle zur Eigenschaftsoptimierung 115
Abbildung 6-8: Zusammenhang zwischen der Anzahl an reduzierten Dimensionen und der erklärten Varianz 123
Abbildung 6-9: Clusterergebnisse unter Anwendung des DBSCAN Algorithmus 124
Abbildung 6-10: Evaluation der Clusterergebnisse der unterschiedlichen Algorithmen nach Mehlstäubl et al. (2023c) 125
Abbildung 6-11: Visualisierung der CVIs für den Ward-Linkage Algorithmus nach Mehlstäubl et al. (2023c) 126
Abbildung 6-12: Graphische Darstellung der Centroide im dreidimensionalen Raum nach Mehlstäubl et al. (2023c) 128

Abbildung 6-13: Quantitative Verteilung der Produktvarianten in den einzelnen Clustern 130

11 Tabellenverzeichnis

Tabelle 2-1: Übersicht der Aktivitäten des CRISP-DM in Anlehnung an Wirth und Hipp (2000) 25
Tabelle 2-2: Übersicht überwachter Lernalgorithmen 33
Tabelle 2-3: Übersicht Clusteralgorithmen 38
Tabelle 2-4: Ableitung der Assoziationsregeln aus dem FP-Baum (min Support=2) in Anlehnung an Han et al. (2000) 41
Tabelle 2-5: Übersicht Assoziationsalgorithmen 42
Tabelle 3-1: Gegenüberstellung der bisherigen Ansätze mit den gestellten Anforderungen 56
Tabelle 5-1: Einordnung der Wissensbedarfe und Anwendungsfälle in den Entscheidungsprozess nach Mehlstäubl et al. (2023a) 68
Tabelle 5-2: Exemplarische Struktur des Vertriebsdatenrumpfs in Anlehnung an Mehlstäubl et al. (2022a) 76
Tabelle 5-3 Struktur des Vertriebsdatenkopfs mit synthetischen Daten 76
Tabelle 5-4: Struktur der Nutzungsdaten mit synthetischen Daten 78
Tabelle 5-5: Auswahl von Datenbereinigungs- und Transformationsverfahren 78
Tabelle 5-6: Faktoren für die Algorithmenauswahl bei einer Regressionsanalyse nach Mehlstäubl et al. (2022a) 83
Tabelle 5-7: Tabellarische Darstellung der Regressionsergebnisse am Beispiel von Gewichten 85
Tabelle 5-8: Faktoren für die Algorithmenauswahl bei einer Klassifikationsanalyse in Anlehnung an Mehlstäubl et al. (2022a) 87
Tabelle 5-9: Binäre Konfusionsmatrix nach Powers (2020) 88
Tabelle 5-10: Überblick über die Algorithmen, deren Merkmale und Eignung nach Mehlstäubl et al. (2023c) 92
Tabelle 5-11: Gegenüberstellung der Assoziationsalgorithmen in Anlehnung an Vani (2015) und Prithiviraj und Porkodi (2015) 96
Tabelle 6-1: Bewertung und Auswahl der Wissensbedarfe 102
Tabelle 6-2: Übersicht der verwendeten Python Bibliotheken 106

Tabelle 6-3: Schritte zur Datenvorbereitung für die Prognose von Produkteigenschaften mit einer Regression 107
Tabelle 6-4: Evaluationsergebnisse für die Prognose der Fahrzeuggewichte 109
Tabelle 6-5: Evaluationsergebnisse für die Prognose der CO_2-Werte 111
Tabelle 6-6: Evaluationsergebnisse für die Prognose der Zahlungsbereitschaft 112
Tabelle 6-7: Ergebnis Eigenschaftsoptimierung 116
Tabelle 6-8: Feature Importance am Beispiel der Steckdose mit 230 V 117
Tabelle 6-9: Schritte zur Datenvorbereitung für die Prognose von Produkteigenschaften mit einer Klassifikation 118
Tabelle 6-10: Evaluationsergebnisse der Klassifikationsmodelle für die Vertriebsländer 119
Tabelle 6-11: Konfusionsmatrix der Vorhersagen des neuronalen Netzes 119
Tabelle 6-12: Vorhersage der Vertriebsländer mit dem neuronalen Netz 120
Tabelle 6-13: Bestimmung der Feature Importance mit dem Random Forest Model 121
Tabelle 6-14: Schritte zur Datenvorbereitung für das Clustering von Produktvarianten 122
Tabelle 6-15: Normalisierte CVIs des Ward-Linkage Algorithmus für 28, 31 und 32 Cluster 126
Tabelle 6-16: Ermittlung der charakteristischen Merkmalsausprägungen der Cluster 127
Tabelle 6-17: Cluster mit starker Homogenität in den Grundmerkmalen 127
Tabelle 6-18: Schritte zur Datenvorbereitung für die Ableitung von Korrelationen zwischen Merkmalsausprägungen 131
Tabelle 6-19: Auszug aus den Assoziationsregeln mit zugehörigem Support- und Confidence-Wert 132
Tabelle 6-20: Übersicht der Experten der Erfolgsvalidierung 134
Tabelle 6-21: Bewertungsergebnisse für die inhaltlichen Kriterien 135
Tabelle 6-22: Bewertungsergebnisse für die formalen Kriterien 137

Anhang

A1 Literaturrecherche A-2

A1.1 Suchmatrix A-2

A1.2 Anwendungsfälle A-2

A2 Experteninterviews A-4

A2.1 Teilnehmer A-4

A2.2 Fragebogen A-4

A2.3 Anwendungsfälle A-6

A3 Erfolgsvalidierung A-7

A3.1 Fragebogen A-7

A3.2 Ergebnisse A-8

A1 Literaturrecherche

A1.1 Suchmatrix

UND

ODER		Aspekte	
	Begriffe	Variety Management	Data Mining
		Product Family Design	Machine Learning
		Product Platform Design	Knowledge Discovery in Databases
		Product Portfolio Management	
		Product Programme Management	

A1.2 Anwendungsfälle

Aktivität	Anwendungsfall	Verfahren	Algorithmus	Referenzen
Marktanalyse	Segmentierung der Märkte	Clustering	Keine Aussage	Agard und Kusiak (2004a), (2004b)
			Fuzzy-c-Means	Chan et al. (2012)
	Zuweisung von Kunden zu Marktsegmenten und Konfigurationen	Klassifikation	Keine Aussage	Agard und Kusiak (2004a), (2004b)
	Identifikation von Korrelationen zwischen Produktmerkmalen	Assoziation	Keine Aussage	Agard und Kusiak (2004a), (2004b)
	Abschätzen der Kaufentscheidungen	Klassifikation	Naïve Bayes	Tucker und Kim (2008)
	Bewertung der Zahlungsbereitschaft für Produkteigenschaften	Klassifikation	C4.5	Tucker und Kim (2009)
Programmplanung	Ermittlung von Abhängigkeiten zwischen Portfolio- und Unternehmenskennzahlen	Regression	Neuronales Netz	Riesener et al. (2019b) und (2020)
Zukunftsplanung	Vorhersage der Nachfragetrends	Regression	Holt-Winters	Tucker und Kim (2011b), Tucker (2014)
		Regression	Automatic Time-Series	Ma et al. (2014), Ma und Kim (2014)
		Klassifikation, Regression	Naïve Bayes Holt-Winters	Tucker und Kim (2011a)
	Prognose der Gewinnentwicklungen	Regression	Automatic Time-Series	Ma und Kim (2016)

Aktivität	Anwendungsfall	Verfahren	Algorithmus	Referenzen
Variantengenerierung	Korrelationen zwischen Kundenanforderungen und Produkteigenschaften	Assoziation	Apriori	Jiao und Zhang (2004), Xia und Wang (2010; Zhang)
			Genetic	Yu und Wang (2010)
	Korrelationen zwischen Produkteigenschaften und Komponenten	Assoziation	Apriori	Tensa et al. (2019)
	Identifikation von sequentiellen Korrelationen zwischen Produktmerkmalen	Assoziation	AprioriAll	Yu und Zhang (2014), Yu et al. (2017)
Variantenvermeidung	Korrelationen zwischen Produktmerkmalen	Assoziation	Apriori	Moon et al. (2006), (2010)
		Assoziation, Clustering	Apriori K-Means	Kusiak et al. (2007)
	Ermittlung der wesentlichen Produktmerkmale der Produktfamilien	Assoziation	Apriori	Wang (2019)
	Bestimmung der wichtigsten Produktmerkmale für den Preis	Klassifikation	ReliefF	Tucker et al. (2010)
	Vorhersage der Preise	Regression	Neuronales Netz	Boyarkin et al. (2019)
Variantenbeherrschung	Unterstützung der Produktionsplanung	Clustering Klassifikation	Keine Aussage	Agard und Kusiak (2004b)
		Assoziation	Apriori	Jiao et al. (2008)
		Clustering	kMM, k-prototype, k-Medoids, UFL fuzzy art	Hochdorffer et al. (2018)
Variantenreduktion	Standardisierung der Produkte	Clustering	K-Medoid	Romanowski und Nagi (2002), (2004) und (2005), Neis (2015)
			K-Means	Kusiak et al. (2007), Ma und Kim (2016)
			Fuzzy-c-Means	Zhang et al. (2007)
			X-Means	Tucker et al. (2010)

A2 Experteninterviews

A2.1 Teilnehmer

	Position	Fachbereich	Erfahrung
Experte 2.1	Abteilungsleiter	Virtuelle Produktentwicklung	5-10 Jahre
Experte 2.2	Hauptabteilungsleiter	IT für die Entwicklung	> 20 Jahre
Experte 2.3	Bereichsleiter	Produktkostenoptimierung	> 20 Jahre
Experte 2.4	Hauptabteilungsleiter	Produktstrategie und -planung	10-20 Jahre
Experte 2.5	Abteilungsleiter	Vertriebsstammdaten	5-10 Jahre
Experte 2.6	Abteilungsleiter	Produktionsplanung	0-5 Jahre
Experte 2.7	Abteilungsleiter	Produktarchitektur	10-20 Jahre
Experte 2.8	Hauptabteilungsleiter	Produktarchitektur und Stückliste	10-20 Jahre

A2.2 Fragebogen

Demografie und Allgemeines

- Welche Position haben Sie inne?
- Was ist Ihr Aufgabenfeld im Unternehmen?
- Seit wie vielen Jahren sind Sie in Ihrem Unternehmen tätig? Welche Stationen haben Sie im Unternehmen und davor durchlaufen?
- Seit wie vielen Jahren beschäftigen Sie sich mit dem Produktportfolio- und Variantenmanagement?
- Was verstehen Sie unter Produktportfoliomanagement und was unter Variantenmanagement? Wie grenzen Sie die Disziplinen voneinander ab?
- Haben Sie schon Erfahrungen im Bereich Data Mining? Wenn ja, welche?
- Haben Sie schon Erfahrungen im Bereich Machine Learning? Wenn ja, welche?

Anwendung von Machine Learning im eigenen Unternehmen

- Nutzen Sie aktuell schon Data Mining und Machine Learning im Varianten- und Produktportfoliomanagement Ihres Unternehmens? Wenn ja, welche Ziele werden verfolgt? Welche Daten (-quellen) werden ausgewertet? Welches Tool verwenden Sie für die Analyse? Wenn nein, warum nicht?
- Werden diese in anderen Bereichen Ihres Unternehmens (außerhalb des Produktportfolio- und Variantenmanagement) (intensiver) eingesetzt? Wenn ja, in welchen Unternehmensbereichen?
- Sind Sie mit bisherigen Ergebnissen von Data Mining und Machine Learning zufrieden? Falls nein, warum nicht?

Herausforderungen von Data Mining und Machine Learning

- Wo sehen Sie die größten Herausforderungen bei dem Einsatz von Data Mining und Machine Learning im Produktportfolio- und Variantenmanagement in Ihrem Unternehmen?
- Welche Schwierigkeiten könnten bei der Nutzung der Ergebnisse durch die Entwickler aufkommen?
- Welche Rolle spielt in diesem Kontext Ihrer Meinung nach das Management? Welchen aktuellen Herausforderungen muss es sich stellen?
- Warum ist der Einsatz von Data Mining und Machine Learning im Produktportfolio- und Variantenmanagement bisher immer gescheitert?
- Welche Unterstützung würden Sie für einen gezielteren Einsatz datengetriebener Methoden benötigen?

Potenziale und Anwendungsfälle von Data Mining und Machine Learning im Produktportfolio- und Variantenmanagement

- Wie würden Sie die Potenziale von Data Mining und Machine Learning im Produktportfolio- und Variantenmanagement einstufen? (sehr gering, gering, mittel, hoch, sehr hoch) Bitte begründen Sie Ihre Einstufung.
- Glauben Sie, dass in der Zukunft Data Mining und Machine Learning an Bedeutung gewinnen werden und eine entscheidende Rolle im Produktportfolio- und Variantenmanagement in Ihrem Unternehmen aber auch in anderen Unternehmen mit komplexen Produktportfolios spielen werden? Bitte begründen Sie Ihre Aussage.
- Denken Sie, dass der Einsatz von Data Mining und Machine Learning mit Risiken verbunden ist? Wenn ja, mit welchen?
- Wo sehen Sie in Zukunft die vielversprechendsten Einsatzmöglichkeiten für Data Mining und Machine Learning in Ihrem Variantenmanagement?
- Wo sehen Sie in Zukunft die vielversprechendsten Einsatzmöglichkeiten für Data Mining und Machine Learning in Ihrem Portfoliomanagement?
- Fallen Ihnen konkreten Problemstellungen ein, wo Sie diese in Ihrer Abteilung nutzen würden?

A2.3 Anwendungsfälle

Anwendungsfall	Verfahren
Standardisierung der Produkte	Clustering
Segmentierung der Märkte	Clustering
Identifikation ähnlicher Komponenten und Baugruppen	Clustering
Simulation der Auswirkungen von Portfolioänderungen	Regression
Ermittlung von Abhängigkeiten zwischen Portfolio- und Unternehmenskennzahlen	Regression
Identifikation von Korrelationen zwischen Komponenten	Assoziation
Identifikation von Korrelationen zwischen Produktmerkmalen	Assoziation
Informationen zur Entscheidungsunterstützung bereitstellen	Regression, Klassifikation
Zuweisung von Kunden zu Marktsegmenten und Konfigurationen	Klassifikation
Vorhersage technischer Produkteigenschaften	Regression, Klassifikation
Zeitliche Entwicklung marktspezifischer Größen	Regression
Entwicklung von Vorschlägen für Portfolioentscheidungen	Klassifikation

A3 Erfolgsvalidierung

A3.1 Fragebogen

1. Das Framework ermöglicht die Ermittlung von Eigenschaften komplexer Produktportfolios mit einer Vielzahl an Merkmalsausprägungen und Komponentenvarianten.
2. Das Framework vermittelt ein Geschäftsverständnis für die Analyse komplexer Produktportfolios durch die Systematisierung der Wissensbedarfe im Entscheidungsprozess.
3. Durch das Framework findet eine datenbasierte Beschreibung komplexer Produktportfolios statt, wodurch ein Datenverständnis in der betrachteten Domäne bereitgestellt wird.
4. Das Framework bietet eine systematische Unterstützung bei der Vorbereitung von Produktportfoliodaten.
5. Das Framework liefert für die industrielle Anwendung von Machine Learning eine Hilfestellung bei der Auswahl und Evaluation von Algorithmen.
6. Das Framework erläutert verschiedene Möglichkeiten zum Einsatz der Machine Learning Modelle für die Analyse komplexer Produktportfolios.
7. In Unternehmen sind intelligente und datengetriebene Lösungen zur Produktportfolioanalyse notwendig, da die heutigen manuellen und erfahrungsbasierten Verfahren die Komplexität nicht mehr handhaben können.
8. Das entwickelte Framework leistet einen Beitrag zur besseren Handhabung komplexer Produktportfolios durch die Generierung von Wissen aus Daten mittels Machine Learning.
9. Die einzelnen Bausteine des Frameworks und ihre Beschreibung sind in sich widerspruchsfrei.
10. Die Inhalte des Frameworks stehen nicht im Widerspruch zu dem allgemeinen Wissen im Produktportfolio- und Variantenmanagement.
11. Das Framework kann auf komplexe Produktportfolios unterschiedlicher Unternehmen angewendet werden.
12. Das Framework kann mit geringfügigen Anpassungen auf andere Bereiche in der Produktentwicklung erweitert werden, ohne seinen Nutzen wesentlich zu verringern.
13. Die Bausteine des Frameworks sind einfach zu verstehen und können von potenziellen Nutzern eingesetzt werden.
14. Das Framework bildet die Grundlage für neue Forschungstätigkeiten zum Einsatz von Machine Learning im Produktportfolio- und Variantenmanagement sowie in der Produktentwicklung.

A3.2 Ergebnisse

	Fragen													
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**
Experte A	5	5	4	5	4	5	4	4	5	4	4	5	5	5
Experte B	4	4	5	5	4	4	5	5	3	4	5	5	4	5
Experte C	4	-	5	3	4	5	5	5	4	4	3	4	2	5
Experte D	5	4	4	5	5	5	5	5	-	-	5	5	4	5
Experte E	4	5	4	5	4	5	5	5	5	5	3	4	5	5
Experte F	5	3	4	-	5	5	4	4	-	5	3	5	5	5
Experte G	5	5	5	5	-	5	5	-	4	-	4	5	3	5
Experte H	5	4	5	5	4	5	5	5	-	5	5	4	5	5
Experte I	5	4	5	5	3	5	5	3	-	5	4	4	3	5

Legende: „5“ – trifft zu | „4“ – trifft eher zu | „3“ – trifft teilweise zu |
„2“ – trifft eher nicht zu | „1“ – trifft nicht zu | „-“ – keine Aussage möglich