IT in Produktion
und Logistik
Univ.-Prof. Dr.-Ing. Markus Rabe

Fakultät Maschinenbau

Technische Universität Dortmund

Masterarbeit

Entscheidungsbaumgestützte Auswahl von Data-Mining-Verfahren im produktionslogistischen Umfeld

bearbeitet von: Susanne Barbara Klöcker

Studiengang: Logistik
Matrikel-Nr.: 166851

Ausgegeben am: 01.07.2021
Eingereicht am: 25.01.2022

Erstprüfer: Dr.-Ing. Dipl.-Inform. Anne Antonia Scheidler
Zweitprüfer: M. Sc. Sahil-Jai Arora

Inhaltsverzeichnis

Abbildungsverzeichnis

Abkürzungsverzeichnis

ITPL Lehrstuhl IT in Produktion und Logistik
z. B. zum Beispiel
usw. und so weiter
KDD Knowledge Discovery in Databases
bspw. beispielsweise
bzw. beziehungsweise
BMI Body-Mass-Index
PCA Principal Component Analysis
TDIDT Top-Down Induction of Decision Trees
FP-Growth Frequent Pattern Growth
SPRINT Scalable Parallelizable Induction of Decision Tree
SAHN Sequentielle agglomerative hierarchische nichtüberlappende Clusteranalyse
SDHN Sequentielle divisive hierarchische nichtüberlappende Clusteranalyse
PAM Partitioning Around Medoids
CLARA Clustering Large Applications
CLARANS Clustering Large Applications based on Randomized Search

1 Einleitung

Das gezielte Entdecken von Wissen in vorhandenen Daten ist ein wichtiges Werkzeug für zielführende Entscheidungen in Geschäftsfragen. Heutzutage stehen den Unternehmen Daten in großen Mengen zur Verfügung. Einen Nutzen können diese daraus allerdings nur ziehen, wenn diese Daten einer geeigneten und qualitativen Verarbeitung unterzogen werden. Insbesondere im Bereich der Produktion und Logistik gibt es ein großes Potenzial, durch Datenanalyse Prozesse zu verbessern, bspw. hinsichtlich der Produktionsplanung sowie der Analyse von Fehlersymptomen und -ursachen (18). Um entscheidungsunterstützendes Wissen zu erlangen, ist die richtige Auswahl an Verfahren zwingend notwendig. Im Bereich des Data Mining wurde im Laufe der Jahre eine Vielzahl an Verfahren entwickelt, die nicht beliebig für jeden Datentyp und jede Aufgabenstellung geeignet ist. Einige Verfahren arbeiten entweder nur mit einem oder bevorzugen einen bestimmten Datentyp. Ausschlaggebend für die Anwendung von Data Mining ist zunächst die vorhandene Datengrundlage hinsichtlich der Menge, Art und Richtigkeit (11). Demnach ist ein hinreichendes Verständnis für die Vorbedingungen nötig, bevor ein Data-Mining-Prozess gestartet werden kann. Darüber hinaus hat die Vorverarbeitung der Daten einen großen Einfluss auf die Qualität der Data-Mining-Ergebnisse. Daher ist ein umfangreicher Überblick über möglichst viele Verfahren notwendig, um schlussendlich qualitative und begründete Entscheidungen bei der Verfahrensauswahl treffen zu können. Soll in der Praxis eine schnelle Entscheidung für ein Verfahren ohne ausreichende Erfahrung getroffen werden, ist die Gefahr einer nur mäßigen Recherche zu groß und aufgrund einer mangelnden Auswahl die Chance gering, das geeignete Verfahren auf Anhieb zu finden.

Das Ziel dieser wissenschaftlichen Arbeit ist es, diese Entscheidungsfindung durch eine ausführliche Recherche zu unterstützen, wodurch eine umfangreiche Übersicht jeglicher nutzbarer Verfahren zur Verfügung gestellt werden soll. Zu diesem Zweck wird ein Entscheidungsbaum konzeptioniert, anhand dessen in Abhängigkeit der vorliegenden Datengrundlage geeignete Data-Mining-Verfahren in kurzer Zeit für eine spezifische Aufgabenstellung gefunden werden können. Weiterhin soll die Arbeit einen verständlichen Überblick darüber geben, inwieweit Verfahren ggf. voneinander abhängig sind. Im Zuge dessen gilt es zu erörtern, ob und welche Verfahren zuvor den Einsatz anderer Verfahren benötigen. Der Entscheidungsbaum soll weiterhin den Anspruch haben, durch eine modulare Bauweise erweiterbar zu sein. Das Ziel ist eine Auslegung derart, dass neue Ansätze oder nicht berücksichtigte Ansätze angefügt werden können.

Bei der Konzeptionierung des Kategorisierungsmodells in Form eines Entscheidungsbaumes wird in Anlehnung an das Data-Mining-Modell nach (17) vorgegangen, womit sich schlussendlich die Struktur der Ebenen begründen lassen soll. Das Modell von Fayyad wird zunächst vorgestellt sowie Herausforderungen im Data-Mining-Prozess diskutiert, um an späterer Stelle Anforderungen an den Entscheidungsbaum stellen zu können. Dem Modell von (17) zufolge beginnt der Data-Mining-Prozess mit der *Datengrundlage*. Demnach werden zunächst die gängigen Datentypen sowie Dateneigenschaften, wie z. B. die Datenqualität, Datenmenge, -herkunft und -dichte, betrachtet. Im Zuge dessen werden ebenfalls vorhandene Aufgabentypen beleuchtet, da davon ausgegangen werden muss, dass der Nutzende des Entscheidungsbaumes geeignete Verfahren für eine bestimmte Problemstellung

sucht. Es folgen drei Datenvorbereitungsschritte, ehe das eigentliche Data Mining durchgeführt werden kann: *Datenselektion, Datenvorverarbeitung, Datentransformation.* Für jeden dieser Schritte werden geeignete Verfahren recherchiert und den Abhängigkeiten der zuvor veranschaulichten Datentypen zugeordnet. Hier bietet sich eine *Einflusstabelle* an, aus welcher Abhängigkeiten zwischen der Datengrundlage und den eruierten Vorbereitungsverfahren hervor gehen. Dadurch wird ersichtlich, welche Vorbereitung welche Datengrundlage benötigt und umgekehrt.

Nach der Datenvorbereitung erfolgt das *Data Mining.* Zu diesem Schritt werden ebenfalls geeignete Verfahren betrachtet und strukturiert. Nachdem der Stand der Technik aufbereitet ist, wird in Abhängigkeit der Informationen und mithilfe der erstellten Einflusstabellen mit dem Aufbau des Entscheidungsbaumes begonnen.

Zunächst werden grob einige Anforderungen an den Entscheidungsbaum gestellt, die sich aus der vorherigen Recherche heraus begründen lassen sollen. Diese Anforderungen können sich bspw. auf die Anwendbarkeit, Allgemeingültigkeit sowie die Reihenfolge der Prozessschritte beziehen. Der Aufbau wird analog zur Recherche auf Grundlage der Prozessreihenfolge nach (17) durchgeführt. Begonnen mit dem Aufgabentyp und der Datengrundlage werden Pfade hin zu geeigneten Vorverarbeitungsverfahren konstruiert und mit der geleisteten Vorarbeit begründet. Schließlich werden die Data-Mining-Verfahren eingefügt, begründet und diskutiert. Im Anschluss werden Auffälligkeiten kritisch betrachtet und mögliche Verbesserungspotenziale erörtert. Diese kritische Betrachtung soll sowohl eine erste Einschätzung der Anwendbarkeit geben, als auch eventuell mögliche Anpassungen anregen.

Zudem wird geprüft, ob die zuvor bestimmten Anforderungen erfüllt werden konnten. Im Zuge dessen muss der Entscheidungsbaum auf seine Anwendbarkeit, mögliche Chancen und Risiken sowie Potenziale zur Erweiterung begutachtet werden.

Abgeschlossen wird die Arbeit mit einem Fazit und einem Ausblick auf mögliche weitere Ansätze.

2 Grundlagen und Forschungsstand

2.1 Vorgehensmodell und Vorbedingungen im Data-Mining-Prozess

Data Mining ist ein Prozess zur Extraktion von impliziten, nicht im Voraus bekannten, aber potenziell nützlichen Informationen und Wissen aus einer Menge verrauschter, in verschiedenen Formen gespeicherter oder unvollständiger, großer Datensätze (53). Es existieren verschiedene Vorgehensmodelle, die den Ablauf eines Data-Mining-Prozesses beschreiben und anleiten. Die zwei bekanntesten Modelle sind einerseits das "Knowledge-Discovery in Databases (KDD)-Modell"von (17) und andererseits das CRISP-Data-Mining-Modell, das von einem Zusammenschluss mehrerer Unternehmen entwickelt wurde (11). Im Folgenden wird das Vorgehensmodell von (17) erläutert, da dies projektunabhängig genutzt werden kann (11) und als Basis für diese wissenschaftliche Arbeit dienen soll. Die Begründung liegt darin, dass (17) sich auf die Datenbereitstellung und die Datenanalyse konzentriert (11) und sich somit für das Ziel dieser Arbeit, ein Entscheidungsmodell für die Anwendung von Data Mining zu generieren, eignet. Das CRISP-Modell ist projektabhängiger und berücksichtigt das gesamte Data-Mining-Projekt in einem Unternehmen über den Analyseprozess hinaus. So sind unter anderem Teilprozesse aufgeführt, die zum einen die aktuelle Situation und Unternehmensziele erfassen und zum anderen der Einsatz von Ergebnisse im Unternehmen nach erfolgter Analyse (11). Aufgrund der Unternehmensunabhängigkeit dieser Arbeit, wird das KDD-Modell nach (17) bevorzugt.

Der Gesamtprozess ist in Abb.1 dargestellt. Er beginnt mit der Auswahl einer geeigneten Datengrundlage. Je nach vorhandener Datengrundlage müssen geeignete Vorverarbeitungsschritte gewählt und durchgeführt werden. Im Sinne der Ganzheitlichkeit muss jedoch erwähnt werden, dass einige Phasen zuvor durchlaufen werden müssen. Zum einen ist ein ausreichendes Verständnis für die zu lösende Aufgabe notwendig, für die das Data Mining eingesetzt werden soll. (17) berücksichtigen diesen Schritt zwar nicht in ihrem Modell, weisen jedoch darauf hin, dass ein gewisses Verständnis für den Anwendungsbereich sowie relevantes Vorwissen nötig ist, um das Ziel des Data-Mining-Prozesses formulieren zu können. Ebenso unterstreicht (27) die Relevanz von domänenspezifischen Kenntnissen und Erfahrungen, um sowohl eine sinnvolle Problemstellung als auch Hypothesen für das Problem bzw. die Probleme aufzustellen. Schlussendlich muss anhand definierter Erfolgskriterien deutlich sein, welche Ergebnisse erreicht werden sollen (11). Weiterhin sind laut (11) mögliche Risiken finanzieller Art oder eine eingeschränkte Nutzung von Daten aus rechtlichen oder firmnepolitischen Gründen zu beachten. Ebenso relevant ist das Verständnis für die verfügbaren Daten (11). Bevor mit der Bearbeitung begonnen werden kann, sollte im Vorfeld definiert werden, welche Daten für die Erreichung des benötigt werden. Da diese Schritte nach (11) stark projektabhängig sind, werden sie im weiteren Verlauf der Arbeit nicht berücksichtigt. Es wird davon ausgegangen, dass das notwendige Vorwissen und die Zieldefinition vorhanden sind, bevor mit dem Data-Mining-Prozess begonnen wird.

Wie in Abb. 1 zu sehen, besteht der Prozess von (17) aus fünf grundlegenden Schritten. Zuerst findet die Datenselektion statt. In dieser Phase werden die für die Zielerreichung benötigten Daten bestimmt und aus geeigneten und verfügbaren Datenquellen herangezo-

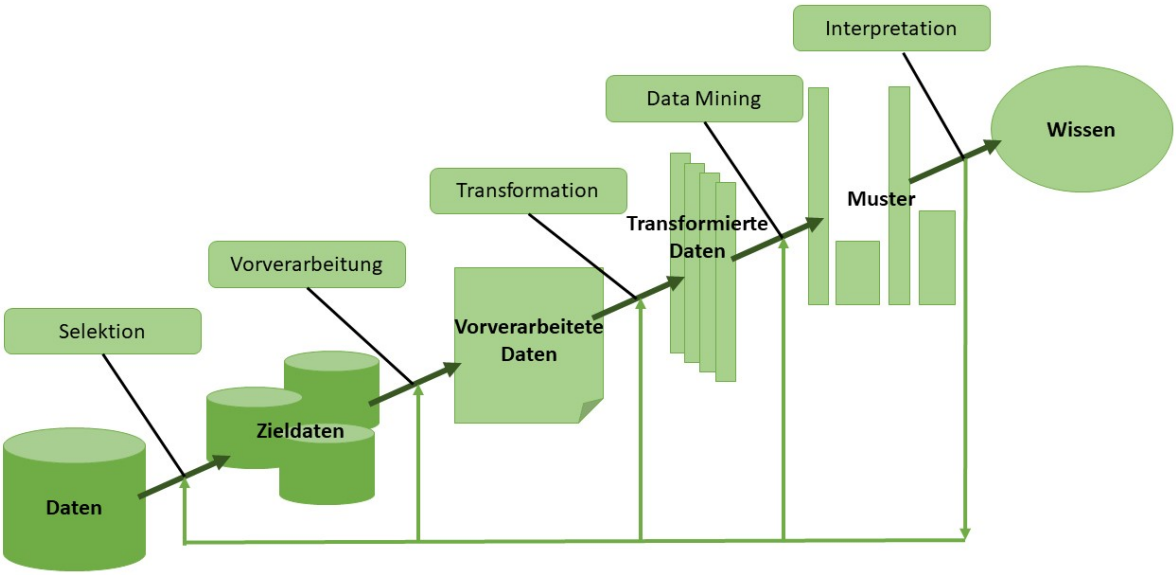

Abbildung 1: Ablauf des Data-Mining-Prozesses (17)

gen (17). Ebenfalls zu dieser Phase gehört die Datenintegration, wenn die ausgewählten Daten unterschiedlichen Quellen entstammen, z. B. unterschiedliche Tabellen oder Datenbanken (11). Dabei ist darauf zu achten, dass die ausgewählten Daten keinen technischen oder rechtlichen Restriktionen unterliegen und in einen Zieldatenbestand überführt werden können (11). Herausforderungen, die sich dabei ergeben, sind unterschiedliche Strukturen der Datenbanken und unterschiedliche Semantiken von Attributen. Laut (11) ist das Ergebnis der Datenintegration eine konsistente Tabelle, deren Datensätze schlüssig sind. Eine Fokussierung auf eine Teilmenge des Datensatzes ist ebenfalls möglich (17). Dies ist im Besonderen dann eine Lösung, wenn aufgrund technischer Restriktionen eine Überführung aller Daten in einen Zieldatanbestand nicht möglich ist, bspw. bei Datentypbeschränkungen des Zielsystems (11).

Das Ergebnis der Datenselektion ist ein Rohdatensatz, dessen Daten laut (47) häufig ungeordnet, fehlerhaft, unvollständig, teilweise redundant oder sogar unwichtig sind. Um die relevanten Daten auszuwählen, sind geeignete Merkmale festzulegen (47). Als Merkmale werden die Spalten einer Datentabelle bezeichnet, auch Attribute genannt. Die Relevanz der Merkmale ist anwendungsabhängig, weshalb die Merkmalsauswahl auf die jeweilig gewählte Analysemethode zugeschnitten sein muss (1). Die Datenselektion und -integration wird in Kapitel 2.2.2 betrachtet, Methoden der Merkmalsauswahl in Kapitel 2.3.2.

Nachdem der Zieldatensatz erstellt wurde, beginnt die zweite Phase, die *Datenvorverarbeitung*. Hier verbirgt sich die meiste und wichtigste Arbeit vor der Datenanalyse (21). In diesem Schritt werden die Daten qualitativ vorbereitet, wobei eine Abwägung zwischen der Genauigkeit der Daten und der Effizienz notwendig ist (49). Die analytischen Algorithmen im Data Mining sind auf die Qualität der Daten angewiesen, die für die Analyse

eingegeben werden (1). Der Zieldatensatz wird bereinigt, indem unter anderem Fehler beseitigt und fehlende, inkonsistente und redundante Werte korrigiert werden (1; 11). Für die verschiedenen Fehlerarten gibt es jeweils unterschiedliche Herangehensweisen, die in Kapitel 2.3.1 behandelt werden. Im Besonderen sind das Identifizieren und Behandeln von Ausreißern und Rauscheffekten sowie die Standardisierung bzw. Normalisierung der Daten Aufgaben der Datenvorverarbeitung (47). Die Standardisierung ist notwendig, da in den meisten Datensätzen unterschiedliche Merkmale vertreten sind, die wiederum unterschiedliche Bezugsskalen aufweisen und daher nicht miteinander verglichen werden können (47). Im Anschluss an die Behandlung der Fehler jeglicher Art ist es in einigen Fällen notwendig, den Datensatz zu reduzieren (27). Dies ist ebenfalls eine Aufgabe der Datenvorverarbeitung. Durch die Datenreduktion, die in Kapitel 2.3.2 beleuchtet wird, soll verhindert werden, dass die Durchführung des Data Minings mit einem ineffektiv großem Aufwand verbunden oder sogar unmöglich ist (11). (17) sehen das bereits erwähnte Finden nützlicher Merkmale in einem gemeinsamen Schritt mit der Datenreduktion. Eine geeignete Merkmalsauswahl ist sinnvoll, um eine hohe Modellkomplexität für die Datenanalyse zu verhindern (47). (47) weist die Merkmalsauswahl jedoch dem Schritt der Vorbereitung zu, die noch vor der Datenvorverarbeitung stattfindet.

Wurden die Daten bereinigt, müssen sie gegebenenfalls für das Data-Mining-Verfahren in ein kompatibles Datenformat transformiert werden. Viele Algorithmen bevorzugen bestimmte Datenformate oder setzen ein bestimmtes Format voraus (11). Aufgrund der großen Anzahl unterschiedlicher Datentypen, stehen entsprechend viele Transformationsmethoden zur Verfügung (1; 11). In Kapitel 2.3.3 werden die gängigsten Methoden der Transformation in Bezug auf zu verwendende Data-Mining-Algorithmen näher untersucht.

Im besten Fall liegt nun ein Datensatz vor, der bereinigte und derart transformierte Daten aufweist, dass das Data Mining mit geeigneten Algorithmen durchgeführt werden kann. Das übergeordnete Ziel des Data Minings ist im weitesten Sinne das Finden von Informationen in vorhandenen Datensätzen (13). Dabei kann es sich entweder um bereits vorhandene Informationen handeln oder um Erkenntnisse neuer und unbekannter Informationen (48). Ziele und Verfahren der gängigsten Anwendungsgebiete werden in Kapitel 2.4 ausführlich betrachtet.

Im Anschluss und letzten Schritt werden die Analyseergebnisse, die beispielsweise in Form von Mustern und Regeln gewonnen werden (17), interpretiert und ausgewertet (47).

2.2 Datengrundlage und -selektion

2.2.1 Datengrundlage

Der Prozess der Datenvorverarbeitung beginnt mit der Bewertung der Daten (38). Die Datengrundlage kann anhand verschiedener Eigenschaften beschrieben werden, die in Abb. 2 dargestellt sind.

Je mehr über die Datengrundlage bekannt ist und je besser man sie versteht, desto effizienter können diese vorverarbeitet werden.

Datentypen

Die Datentypen werden in der Literatur unterschiedlich detailliert kategorisiert.

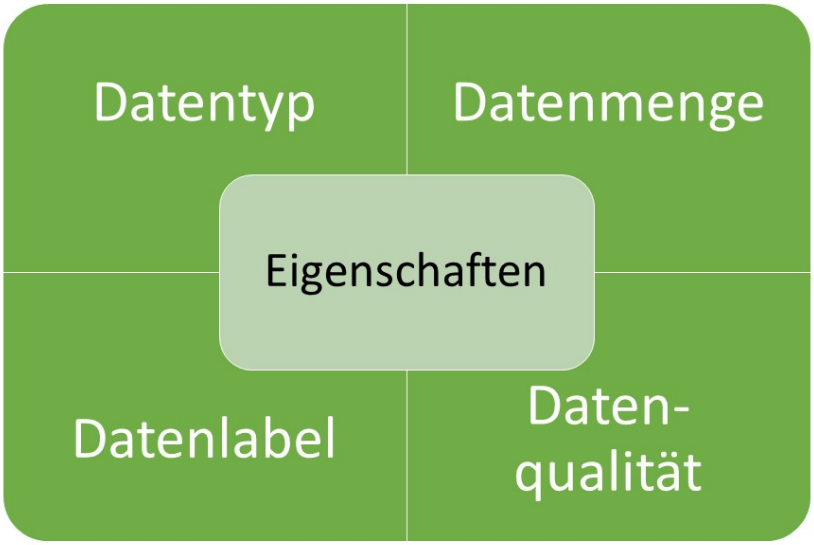

Abbildung 2: Eigenschaften der Datengrundlage

Im Allgemeinen werden die Datentypen drei Hauptkategorien zugeordnet: der *nominale* Datentyp, der *ordinale* Datentyp und der *metrische* Datentyp (11).

Der nominale Datentyp, der auch *kategorisch* genannt wird (1; 27; 33), dient dazu, Objekte in Kategorien einzuordnen, indem ihnen eine Bezeichnung (z.B. „Geschlecht“, „Monat“, „Gehalt“) zugeordnet wird. Diese Bezeichnung kann zwar numerisch sein, hat aber in jenem Fall keine mathematische Bedeutung und Rechenoperatoren können nicht auf sie angewandt werden (7). Kategorische Daten besitzen keine geordnete Rangfolge (11), allerdings kann jeder Kategorie eine fortlaufende Nummer zugewiesen werden, um sie voneinander zu unterscheiden (33). Die einzige Operation, die mit diesen Daten möglich ist, ist eine Unterscheidung dahingehend, ob sie *gleich* oder *ungleich* sind (3; 11; 33). Eine spezielle Form kategorischer Daten sind *Binärvariablen* mit höchstens zwei möglichen Werten (1). Kategorische Attribute können durch eine Codierung beider Werte mit 1 und 0 in metrische Attribute umgewandelt werden. Diese Art der Transformation wird Binärcodierung genannt (11) und in Kapitel 2.3.3 näher betrachtet.

Ordinale Daten besitzen eine sinnvolle Reihenfolge, können also bspw. nach *größer* und *kleiner* geordnet werden, erlauben jedoch keine Anwendung mathematischer Operatoren (3; 11). Mögliche Operationen sind einerseits die Unterscheidung und der Vergleich miteinander und andererseits die Kodierung. Die Kodierung macht es bspw. durch Gruppierung der ordinalen Daten möglich, numerische Werte für jene Gruppen zu bestimmen (33).

Metrische Daten sind numerische Daten, auf die Rechenoperatoren angewandt werden können (11). Ihnen können zwei Datentypen zugeordnet werden: Zum einen der *diskrete* Datentyp, der ganzzahlige Werte umfasst, zum anderen der *kontinuierliche* Datentyp, der jeden beliebigen reellen Wert innerhalb des festgelegten Definitionsbereichs erlaubt (11).

Nicht jeder numerische Wert ist gleichzeitig ein metrischer Datentyp, sie können ebenso dem ordinalen Datentyp zugeordnet werden oder, wie bereits erwähnt, dem kategorischen Datentyp. Demnach scheinen numerische Daten für sich zu stehen und können laut (11) und (7) ebenfalls unterschieden werden, weshalb diese hier für eine bessere Übersicht zu einer vierten Hauptkategorie zusammengefasst werden.

Numerische Daten können einerseits ganzzahlige Werte annehmen, mit denen mathematische Berechnungen möglich sind, die (11) als *absolutskalenbasierte Daten* bezeichnen, während (7) diese Kategorie als *Integer-Variablen* betitelt. Die Bedeutung ist dieselbe. Zu dieser Art von Daten gehören jene Objekte, die gezählt werden können. Zu unterscheiden sind diese allerdings von kategorischen Variablen, die lediglich eine numerische Form aufweisen (7). Des Weiteren können numerische Daten den *intervallbasierten* und den *verhältnisbasierten* Datentypen zugeordnet werden (7; 11), wobei es sich um Skalen handelt, die es ermöglichen, numerische Variablen mit einer theoretisch unendlichen Genauigkeit zu definieren und zu messen (27). Beide Skalen unterscheiden sich in der Definition des Nullpunktes. Der Nullpunkt einer Intervallskala wird willkürlich festgelegt (11; 27) und gibt laut (27) nicht die vollkommene Abwesenheit der zu messenden Größe an. Als Beispiel ist die Temperaturskala zu nennen, bei der 0°C nicht die Abwesenheit von Temperatur bedeutet (27). Weiterhin sind mit diesen Daten nur eingeschränkt mathematische Berechnungen möglich, denn, wenn man bei dem Beispiel der Temperaturskala bleibt, sind 20°C rein physikalisch nicht viermal wärmer als 5°C (11). Im Gegensatz dazu besitzt eine Verhältnisskala einen absoluten Nullpunkt, was Verhältnisrelationen für die Variablen bedeutet (27) sowie die Möglichkeit der Addition und Subtraktion mit sich bringt (11). Anhand dieser Skala können bspw. Entfernungen, Größen, Höhen und Längen miteinander verrechnet und in ein Verhältnis zueinander gebracht werden, vorausgesetzt, es liegt eine einheitliche Maßeinheit vor (11).

Die beschriebenen Datentypen fasst (1) unter der übergeordneten Kategorie der *Nichtabhängigkeitsorientierten Daten* zusammen. Darunter fallen alle Daten, die unabhängig voneinander behandelt werden können. Auf der anderen Seite stellt der Autor die Kategorie der *Abhängigkeitsorientierten Daten* vor. Abhängigkeitsorientierte Daten sind solche, die zeitlich, räumlich oder durch explizite Netzbeziehungen miteinander verbunden sind. Das Wissen über bestehende Abhängigkeiten beeinflusst den Data-Mining-Prozess, da erwartete Beziehungen in den Daten verändert werden (1). Die Abhängigkeit zwischen den Daten kann dabei explizit oder implizit sein. Eine explizite Abhängigkeit ist beispielsweise an Kanten zwischen zwei Datenpunkten zu erkennen. Implizite Abhängigkeiten sind dagegen solche, die zwar nicht explizit angegeben werden, aber „typischerweise“ vorliegen, wie z. B. die Ähnlichkeit aufeinanderfolgender Temperaturwerte (1). Zu dieser Kategorie gehören Zeitreihendaten und Zeichenketten, Raum-Daten, Raum-Zeit-Daten sowie Netzwerk- und Graphdaten (1).

Datenbeschriftung

Daten können entweder speziell bezeichnete Attribute („gelabelte Daten“) oder Attribute ohne eine spezielle Bezeichnung haben („ungelabelte Daten“) (7). Das Data Mining mit gelabelten Daten wird als *überwachtes Lernen* bezeichnet. Überwachtes Lernen zeichnet

sich dadurch aus, dass es eine Testdatenmenge gibt, die als Beispiele für zukünftige Daten zur Verfügung stehen (11). Sind die gelabelten Daten kategorisch, findet eine Klassifizierung (Kapitel 2.4.3) statt. Bei numerischen Attributen wird eine Regression durchgeführt (7). Das Lernen mit ungelabelten Daten wird *unüberwachtes Lernen* genannt, dessen Ziel es ist, möglichst viele Informationen aus dem Datensatz zu gewinnen (7). Mit dieser Art von Daten werden zum einen Cluster-Analysen durchgeführt (Kapitel 2.4.4), dessen Ziel es ist, ähnliche Objekte ohne zuvor bekannte Muster in Gruppen einzuteilen (11). Im Gegensatz zum überwachten Lernen, ist dabei keine „Lösung“ vorgegeben, mit der die Ergebnisse verglichen werden können (11). Zum anderen gehört das Finden von Beziehungen zwischen Objekten zum unüberwachten Lernen, die sogenannte Assoziationsanalyse (Kapitel 2.4.2). Eine der bekanntesten Anwendungen einer Assozationsanalyse ist die Warenkorbanalyse. Anhand von Assoziationsregeln wird vorhergesagt, welche Produkte mit hoher Wahrscheinlichkeit gemeinsam gekauft werden (7).

Datenqualität

Die Datenqualität ist entscheidend für die Ergebnisse der eingesetzten Data-Mining-Algorithmen (11; 49). Allgemeine Regeln für die Qualität der Daten festzulegen ist besonders aufgrund der Datenvielfalt im Zuge der unterschiedlichen Datenquellen nicht möglich (49). Herausforderungen können durch Charakteristika des Datensatzes einhergehen, die in der Literatur mit „5 V's“ bezeichnet werden (21; 49). Diese 5 V's bestehen aus: Volumen (Volume), Geschwindigkeit (Velocity), Vielfalt (Variety), Wahrhaftigkeit (Veracity) und Wert (Value) (21). Vorherrschende Qualitätsprobleme müssen an den vorliegenden Daten selbst identifiziert werden (49). Faktoren, die in die Datenqualität eingehen, sind Vollständigkeit, Konsistenz, Genauigkeit, Glaubwürdigkeit und Interpretierbarkeit (22). Jedoch weisen Datensätze in der Regel Fehler auf (Kapitel 2.1.1). Zum einen können ungewöhnliche Werte vorliegen, zum anderen können Werte fehlen oder es gibt Inkonsistenzen in den Daten (22). Einige Verfahren können zwar mit gewissen Fehlern umgehen und diese ignorieren, ohne dass die Ergebnisqualität leidet, allerdings kann davon nicht ausgegangen werden (11). Um eine hohe Datenqualität zu erreichen, ist die Datenvorverarbeitung verantwortlich (11).

2.2.2 Datenselektion und -integration

Bei der Datenselektion werden die benötigten Daten für die Analyse beschafft. Diese können aus unterschiedlichen Tabellen und Datenbanken stammen(11). Für die Anwendung von Data-Mining-Verfahren ist ein zusammenhängender Datensatz notwendig, weshalb eine Datenintegration durchgeführt werden muss, in der alle Datensätze in einer einheitlichen Datentabelle zusammengeführt werden (11; 26). Bei der Datenintegration müssen semantische Mehrdeutigkeiten beseitigt und die Daten bereinigt werden (53).

(38) schlagen vor, sich einen geeigneten Überblick über die verfügbaren Quellen zu verschaffen und Metadaten aus den unterschiedlichen Datensätzen zu gewinnen, indem vorläufige Analysen durchgeführt werden. Zu diesen Metadaten gehören:

- Der Identifikator der Datenquelle

- Das Format, also der Datentyp der Datenquelle
- Das Zeitfenster, das von den Daten widergespiegelt wird
- Die Anzahl der Datensätze der Datenquelle
- Die Anzahl der Attribute der Datenquelle
- Der zeitliche Maßstab
- Der räumliche Maßstab
- Die Größe der Datenquelle (Anzahl Bytes)

Mit der Selektion sowie der Integration können einige Probleme mit der Zusammenführung der Daten aus unterschiedlichen Quellen einhergehen. Zum einen können *Redundanzen* entstehen, wenn Attribute zwar syntaktisch unterschiedlich sind, die Semantik jedoch dieselbe ist (z. B. die Attribute „Name“ und „name“) (11). Diese können vermieden werden, indem jedes Attribut eine eigene Semantik besitzt und syntaktische Fehler, wie z. B. Rechtschreibfehler, korrigiert werden. Dadurch wird zusätzlich das *Entitätenidentifikationsproblem* angegangen, das die Semantik der Attribute betrifft. Haben zwei Attribute einen ähnlichen Namen, ist zu prüfen, ob sie ebenfalls dieselbe Bedeutung haben (z.B. „KundenID“ und „Kundennummer“) (11). Weiterhin können durch das Zusammenführen unterschiedlicher Datenquellen *Widersprüche* entstehen, die möglicherweise durch Daten verursacht werden, die nicht aktuell sind (11), zum Beispiel sind zwei unterschiedliche Wohnorte für dieselbe Person hinterlegt. So können für ein und dasselbe Attribut in einem Datensatz unterschiedliche Einträge bzw. Werte vorhanden sein. Ein ähnliches Problem wird durch *Datenwertkonflikte* hervorgerufen. Es liegt vor, wenn in den Datensätzen eines Attributs unterschiedliche Maßeinheiten auftreten, zum Beispiel eine Entfernung in Kilometern und Meilen (11). Nicht zuletzt sind *Verletzungen der referenziellen Integrität* möglich. Diese werden durch falsche Verweise eines Fremdschlüssels auf einen nicht existierenden Schlüsselwert einer anderen Tabelle hervorgerufen (11). Erst dann, wenn alle Daten in einer schlüssigen Tabelle vorliegen, kann mit der Datensäuberung (Kapitel 2.3.1) begonnen werden (11). Konzepte für Datenintegration werden unter anderem in den Arbeiten von (35) und (39) vorgestellt.

2.3 Datenvorverarbeitung und -transformation

2.3.1 Datensäuberung

In diesem Teilkapitel werden verschiedene Arten von Fehlern erläutert, die im Datensatz vorkommen können und die es zu behandeln gilt. Eine Vielzahl von Gründen kann für fehlerhafte Datensätze sorgen, wie bspw. Messfehler, subjektive Beurteilungen und Fehlfunktionen (7). Wie bereits erwähnt, hängen die Analyseergebnisse bei der Anwendung von Data-Mining-Verfahren maßgeblich von der Qualität der eingegebenen Daten ab. Daher ist dafür zu sorgen, dass fehlerhafte Daten die anschließende Analyse (Kapitel 2.4) nicht verzerren oder verhindern. Zu behandeln sind vor allem fehlende, falsche und inkonsistente Daten (1) sowie Ausreißer und verrauschte Werte (11). Des Weiteren ist eine

Skalierung bzw. Standardisierung der Daten notwendig, da mehrdimensionale Datensätze häufig unterschiedliche Wertebereiche enthalten (47). Es gilt, die unterschiedlichen Daten einerseits vergleichbar zu machen (47) und andererseits zu verhindern, dass einige Werte implizit ignoriert werden, da ihre Messskala niedriger ist (Bspw. Werte des Attributes „Alter“ im Vergleich zu Werten des Attributes „Gehalt“) (1).

Bei der Datenbereinigung sollte grundsätzlich darauf verzichtet werden, neue Informationen hinzuzufügen. Eingefügte Werte sollten derart informationsneutral sein, dass sie die vorhandenen Werte nicht verzerren (11).

Fehlende Werte

Der Umgang mit fehlenden Werten wird in der Arbeit von (20) ausführlich betrachtet. Fehlende Werte im Datensatz können zum einen ignoriert und die betroffene Spalte (das betroffene Attribut) aus der Daten-Tabelle *gelöscht* werden (11; 43). Dies gilt als die einfachste Methode zur Behandlung fehlender Werte (6; 27). Dabei muss beachtet werden, dass leere Felder eine Information enthalten können und Attribute nur gelöscht werden sollten, wenn genügend Attribute übrigbleiben (11). Des Weiteren können statt der Attribute einzelne Datensätze gelöscht werden, die einen fehlenden Wert enthalten. Nachteilig bei der Eliminierung von Datensätzen und Attributen ist, dass unter Umständen sehr viele Daten gelöscht werden (1), weshalb dieses Vorgehen nicht zu empfehlen ist (7). Eine andere Möglichkeit besteht darin, fehlende Werte *manuell in die Tabelle einzufügen*. Dies kann sich bei einer großen Datenmenge jedoch als sehr zeitintensiv herausstellen oder unter Umständen undurchführbar sein, wenn die Daten schlichtweg nicht verfügbar sind (11).

Sollten viele Werte fehlen oder ein leeres Feld als Information gelten, können diese Felder mit einer *globalen Konstante* gefüllt werden. Allerdings besteht dabei die Gefahr, dass diese Konstante implizit zu einem positiven Faktor wird, der in der Realität nicht vorhanden bzw. nicht gerechtfertigt ist (27). Liegen metrische Attribute vor, können leere Felder durch den *Durchschnittswert* aller Einträge ersetzt werden (7). Diese Art der Datensäuberung ist leicht umzusetzen und besonders bei einer Klassifikationsaufgabe zu wählen, wenn die Werte einer Klasse dicht beieinander liegen und davon ausgegangen werden kann, dass sich die fehlenden Werte im ähnlichen Wertebereich befinden (11). Des Weiteren können der Merkmalsmittelwert einer Klasse für die Ersetzung eines fehlenden Wertes genutzt werden, was allerdings nur bei einer Klassifikationsaufgabe möglich ist, bei denen die Klassen in der Stichprobe im Voraus identifiziert wurden (27). Durchschnittswerte eignen sich zudem besonders bei Zeitreihendaten. Dabei werden Durchschnittswerte aus den Werten berechnet, die unmittelbar vor und nach dem Zeitpunkt des fehlenden Wertes aufgezeichnet wurden (1).

Statt des Durchschnittswertes kann außerdem der *wahrscheinlichste Wert* eines Attributes in den jeweiligen Zellen eingefügt werden, der mit statistischen Methoden ermittelt werden kann. Dieser Wert sollte sinnvoll und begründet sein (11). Das Schätzen fehlender Attribute kann im Allgemeinen durch Methoden der Klassifizierung umgesetzt werden (1). Beispielsweise kann der k-Nearest-Neighbour-Algorithmus (Kapitel 2.4.3) zur Schätzung und Imputation fehlender Daten verwendet werden. Der Algorithmus kann sowohl

kategorische als auch numerische Attribute vorhersagen (20). (43) stellen eine Möglichkeit vor, fehlende Daten in Echtzeit für die Klassifikation zu behandeln, die ebenfalls für kategorische und numerische Daten eingesetzt werden kann. Zudem geben die Autoren einen umfangreichen Überblick über weitere Methoden, mit fehlenden Werten umzugehen. (45) stellen den sogenannten SimFiller vor, ein Algorithmus, der ähnliche Datensatzpaare findet, wobei mindestens einer der beiden Werte keinen Nullwert haben darf, und den fehlenden Wert durch den Wert des ähnlichsten Objektes übernimmt.

Daneben besteht die Möglichkeit, den *häufigsten Wert* einzusetzen (11). Besonders bei kategorischen Daten ist dieser Ansatz sinnvoll, wenn es einen Attributwert gibt, der besonders häufig vertreten ist. Dieser Attributwert kann die fehlenden Werte ersetzen (7). Weiterhin können bekannte *Relationen* zwischen Attributen dabei helfen, fehlende Werte zu bestimmen bzw. zu errechnen. Bspw. kann das Alter einer Person berechnet werden, wenn das Geburtsjahr bekannt ist (11). Eine weitere Möglichkeit ist die Vorhersage der Werte mithilfe von Methoden der Assoziationsanalyse (11). Die Nutzung von Assoziationsregeln zur Schätzung fehlender Werte gilt als zuverlässig (7). Sollte keine der beschriebenen Verfahren die richtige Wahl für die Behandlung der fehlenden Daten sein, besteht die letzte Möglichkeit darin, den Datensatz als *fehlerhaft zu kennzeichnen.* Diese werden von der weiteren Verarbeitung ausgeschlossen. Dies ist vor allem dann sinnvoll, wenn genügend vollständige Daten vorhanden sind, allerdings gilt das nur für die Trainingsdaten (11).

Bevor jedoch fehlende Daten ersetzt werden, sollte überprüft werden, ob der (geplante) einzusetzende Data-Mining-Algorithmus mit fehlenden Werten umgehen kann. Einige Methoden können trotz fehlender Daten eingesetzt werden (1; 27). In diesem Fall kann auf eine Bearbeitung verzichtet werden, wodurch eine mögliche Verzerrung der Daten verhindert werden kann (1).

Im Allgemeinen gilt, dass durch die künstliche Ergänzung fehlender Werte Veränderungen im Datensatz vorgenommen werden (1), weshalb mehrere Data-Mining-Lösungen mit und ohne den Merkmalen, die fehlende Werte haben, erstellt und interpretiert werden sollten (27).

Falsche und inkonsistente Werte

Inkonsistenzen treten in der Regel auf, wenn Daten aus verschiedenen Quellen in unterschiedlichen Formaten vorliegen und müssen im Rahmen der Datenintegration berücksichtigt werden (1). Neben Fehlern auf struktureller Ebene können allerdings auch menschliche Fehler zu falschen und inkonsistenten Daten führen, indem Schreibfehler oder fehlerhafte Einträge gemacht werden (4; 11). Zu diesen Fehlern gehören Werte, die den vorgegebenen Wertebereich verlassen, die Plausibilitätsbeziehungen verletzen, sowie widersprüchliche Daten, wie beispielsweise das Alter eines Kunden, das nicht zum dazugehörigen Geburtsjahr passt (11). (1) weist darauf hin, dass häufig Domänenwissen zwischen verschiedenen Attributen verfügbar ist und einige Tools dieses Wissen nutzen, um fehlerhafte Einträge zu erkennen. Als anschauliches Beispiel bringt der Autor die Attribute „Land“ und „Stadt“ vor. Wurde das Feld zum Attribut „Land“ mit „Vereinigte Staaten“ gefüllt, so muss ein Fehler vorliegen, wenn das zugehörige Attribut „Stadt“ den Wert „Shanghai“

aufweist. Offensichtliche Fehler können außerdem in der Form von Ausreißern vorliegen. So ist beispielsweise eine Angabe von „6m“, wenn es um die Körpergröße eines Menschen geht, nicht nur falsch, sondern dieser Wert weicht von den „normalen“ Werten stark ab, weshalb man den Fehler als Ausreißer erkennen kann. Ausreißer sind allerdings nicht immer automatisch Fehler, sondern beinhalten möglicherweise wichtige Informationen (1). Ausreißer werden im nächsten Abschnitt behandelt.

(4) schlagen eine Methode zur automatischen Erkennung von falschen und fehlenden Werten vor, die auf Assoziationsregeln basiert und den FP-Growth-Algorithmus verwendet (Kapitel 2.4.2). Weiterhin raten die Autoren bei großen Datenmengen dazu, eine Clusteranalyse für die Fehlererkennung zu nutzen. Die Nutzung von Assoziationsregeln macht insofern Sinn, als dass fehlerhafte Daten diese häufig verletzen. Daher verwenden (9) eine Methode, um häufige Beziehungen zwischen Daten zu erkennen und dadurch Anomalien zu extrahieren. (59) nutzen als Basis ihrer Methode zur Erkennung fehlerhafter Daten den lokalen Ausreißerfaktor (LOF).

Ausreißer und Rauschen

Ausreißer zeichnen sich dadurch aus, dass sie sich stark von den anderen Werten unterscheiden (11). Allerdings ist zu betonen, dass es sich dabei nicht automatisch um Fehler handelt. Daten, die als Ausreißer identifiziert werden, können durchaus „richtig“ sein und durch ihren ungewöhnlichen Wert wertvolle Informationen enthalten (47). Es gibt Data-Mining-Verfahren, die sich auf die Erkennung von Ausreißern konzentrieren, somit können sie das wesentliche Ergebnis einer Datenanalyse sein (27). Ein Beispiel dafür ist die Anwendung bei der Erkennung betrügerischer Kreditkartentransaktionen, bei der die selteneren „abnormalen“ Beobachtungen einen größeren Informationsgewinn liefern als die häufigeren „normalen“ Beobachtungen (27). Auf der anderen Seite führen Ausreißer bei vielen Data-Mining-Verfahren allerdings zu Problemen, da sie beispielsweise unerwünschte Verteilungen nach sich ziehen (27). Wie bereits im Abschnitt zu den falschen Daten beschrieben, können Ausreißer häufig leicht durch offensichtliche Fehler, wie Rechtschreibfehler, Kommasetzungsfehler und vereinzelte Plausibilitätsverletzungen usw. identifiziert werden. Dennoch raten (1) und (27) dazu, eine automatische Ausreißererkennung mit Vorsicht einzusetzen, damit wichtige Informationen nicht aussortiert werden. Im Folgenden werden Vorgehensweisen erläutert, um Ausreißer zu erkennen und zu behandeln.

Werte, die stark von den anderen Werten abweichen und somit als Ausreißer gelten, können anhand der 2-Sigma-Regel erfasst werden (47). Der Ausreißer wird dadurch erkannt, dass mindestens ein Merkmal um mehr als die doppelte Standardabweichung vom Mittelwert abweicht (47). Diese Regeln kann erweitert werden und entsprechend als 3-, 4-, oder 5-Sigma-Regel angewandt werden (47). Dieses Vorgehen ist allerdings nicht bei lokalen Ausreißern anwendbar, die den Wertebereich nicht verlassen (47). Solche Ausreißer können dagegen mit differentiellen Regeln erkannt werden, bei denen ein Wert mit dem Folgewert verglichen wird (47). Weiterhin gibt es Visualisierungsmethoden, die für die Erkennung von Ausreißern nützlich sein können, allerdings nur bei ein- bis dreidimensionalen Daten (27). (27) merkt an, dass herkömmliche Verfahren der Ausreißererkennung nicht auf sehr großen Datensätzen anwendbar sind. (1) schlägt für große Datensätze den Ansatz der

Assoziationsmusteranalyse vor, wenn Verfahren wie abstandsbasierte Algorithmen nicht eingesetzt werden können.

Der Begriff Rauschen wird in der Literatur unterschiedlich verwendet. (7) versteht unter einem verrauschten Wert einen Wert, der zwar gültig ist, aber falsch aufgezeichnet wurde. Dies kann zu Problemen bei der Erkennung eines Fehlers sein, beispielsweise dann, wenn statt des numerischen Wertes „1,345“ der Wert „13,45“ aufgenommen wurde. Weniger problematisch sind dagegen ungültige Werte, die leichter zu identifizieren und korrigieren sind, bspw. der Wert „1,34X“ (7). Verrauschte Werte in Datensätzen zu finden, kann einerseits durch grundlegende Analysen oder eine Sortierung der Werte erfolgen (7). Auffälligkeiten, die dadurch erkannt werden können, sind beispielsweise, dass eine numerische Variable nur sechs verschiedene Werte annehmen kann, wodurch die Möglichkeit besteht, diese Variablen als kategorisch zu behandeln statt kontinuierlich. Weiterhin können alle Werte einer Variablen gleich sein, weshalb das Attribut möglicherweise zwecks Datenreduktion ignoriert werden kann. Zudem können mögliche Eingabefehler bei Ausreißern identifiziert werden (7).

Skalierung und Standardisierung

Die Skalierung verfolgt das Ziel, Informationen angemessen abzubilden (3). Merkmale, die ein höheres Informationsniveau besitzen (numerische Merkmale) können auf ein niedrigeres Informationsniveau skaliert werden (bspw. durch Diskretisierung). Umgekehrt ist eine Skalierung nicht möglich, ohne mehr Informationen in ein Merkmal zu interpretieren (3).

Insbesondere für Data-Mining-Verfahren, die auf Berechnungen von Abständen zwischen den Objekten beruhen, ist die Standardisierung notwendig. Die Objektwerte werden entsprechend auf einen vorgegebenen Bereich skaliert, bspw. [0,1] (27). (50) stellen eine Standardisierungsstrategie vor, die auf der logarithmischen Kubikwurzel-Normalisierung basiert.

2.3.2 Merkmalsauswahl und Datenreduktion

Wurden die Fehler behandelt und bestenfalls komplett entfernt, gilt es in einigen Fällen, den Datensatz zu reduzieren, sollte dieser zu umfangreich sein. Dadurch soll verhindert werden, dass das Data Mining mit einem zu großen Aufwand verbunden oder sogar unmöglich ist (11). Eine große Datenmenge ist für das Data Mining zwar wünschenswert, jedoch ist häufig eine Reduzierung der Dimensionalität erforderlich, um eine zu hohe Modellkomplexität zu verhindern (47). Eine hohe Dimensionalität kann außerdem zu einer Überlastung der Daten führen, womit Data-Mining-Algorithmen schlussendlich Probleme haben können (27). Eine Datenreduktion geht zwar mit einem gewissen Verlust an Informationen einher, allerdings können rechenintensive Algorithmen besser auf kleineren Datensätzen angewandt werden (1). Eine Möglichkeit der Datenreduktion ist es, anhand einer Merkmalsauswahl die Attribute zu reduzieren, indem irrelevante Attribute gelöscht werden (57). Eine Merkmalsauswahl dient allerdings nicht nur der Reduzierung des Datensatzes im Allgemeinen, sondern sorgt dafür, dass relevante Attribute den irrelevanten vorgezogen werden. Zudem können irrelevante Attribute störende Effekte auf das Modell haben, wodurch die Qualität der Ergebnisse beeinträchtigt werden kann (47). Eine weitere

Möglichkeit ist das Zusammenfassen von abhängigen Attributen zu einem Attribut (11).

Für die Merkmalsauswahl stehen unterschiedliche Algorithmen zur Verfügung, deren Ziel es ist, die informativsten Merkmale im Hinblick auf die Klassenbezeichnung auszuwählen (1). Die Merkmalsauswahl kann einerseits mit Filtermodellen durchgeführt werden. Dafür muss ein eindeutiges, mathematisches Kriterium zu Verfügung stehen, durch das die Qualität eines Merkmals bewertet werden kann. Dadurch werden irrelevante Merkmale herausgefiltert (1). Dabei kann auch eine gleichzeitige Bewertung mehrerer Merkmale als Gruppe durchgeführt werden, was den Vorteil mit sich bringt, dass Redundanzen berücksichtigt werden (1). Bei zwei sehr stark korrelierenden Merkmalen ist es sinnvoll, nur eines der Merkmale zu verwenden, da durch das zweite Merkmal keine neuen Informationen geliefert werden (1). Nachteilig ist jedoch, dass durch die gleichzeitige Betrachtung mehrerer Merkmale, die Rechenintensität steigt. Daher werden in der Praxis meistens Methoden angewandt, in denen die Merkmale unabhängig voneinander betrachtet werden. Es werden dann die Merkmale ausgewählt, die die höchste Unterscheidungskraft besitzen (1).

Eines dieser Filtermodelle ist der *Gini-Index*. Dieser wird verwendet, um die Trennschärfe eines bestimmten Merkmals zu messen (1). Der Gini-Index wird in der Regel bei kategorischen Daten hinzugezogen, kann aber auch auf numerische Daten angewandt werden, wenn diese zuvor durch Diskretisierung verallgemeinert wurden (1). Niedrige Werte des Gini-Index gehen mit einer größeren Trennschärfe einher (1). Weiterhin kann die Messung der *Entropie* für die Bewertung der Attribute genutzt werden. Der Wert einer klassenbasierten Entropie liegt im Intervall [0, log2(k)]. Der Wert 0 bedeutet eine größtmögliche Unterscheidungskraft, je höher der Wert ist, desto stärker ist die „Vermischung“ der Klassen (1). Für numerische Attribute ist der Fisher-Score geeignet (1). Je größer der Fisher-Score, desto größer ist die Trennschärfe des Attributs. Das bedeutet, dass jene Attribute mit dem höchsten Fisher-Score für die Verwendung eines Klassifikationsalgorithmus ausgewählt werden können (1). Eine Verallgemeinerung des Fisher-Scores ist die *lineare Diskriminante nach Fisher* (1).

Filtermodelle für die Merkmalsauswahl sind unabhängig vom verwendeten Klassifizierungsalgorithmus (1). In einigen Fällen kann es allerdings sinnvoll sein, die Eigenschaften des ausgewählten Klassifizierungsalgorithmus für die Merkmalsauswahl zu nutzen. Beispielsweise kann ein linearer Klassifikator effektiver mit einem Satz von Merkmalen arbeiten, bei denen die Klassen am besten mit linearen Trennzeichen modelliert werden, während ein abstandsbasierter Klassifikator gut mit Merkmalen funktioniert, bei denen die Abstände die Klassenverteilung widerspiegeln (1). *Wrapper-Modelle* können die Merkmalsauswahl für den jeweiligen Klassifizierungsalgorithmus optimieren (1). Wrapper-Modelle gehen davon aus, dass ein Klassifizierungsalgorithmus zur Verfügung steht, der bewerten kann, wie gut der Algorithmus bei einer bestimmten Teilmenge von Merkmalen abschneidet. Um diesen Algorithmus herum wird ein Algorithmus für die Merkmalsuche erstellt, der die relevanten Merkmale bestimmt (1).

Nicht zuletzt können eingebettete Modelle für die Merkmalsauswahl verwendet werden. Dabei wird die Lösung eines Klassifikationsmodells verwendet, um Hinweise auf wichtige Merkmale zu extrahieren. Daraufhin werden diese Merkmale isoliert und das Klassifikati-

onsmodell anhand der Auswahl neu trainiert (1). Bspw. enthält der ID3 eine eingebettete Merkmalsauswahlmethode (1). Mit einem iterativen Ansatz werden rekursiv Merkmale eliminiert, anschließend wird der Klassifikator auf der verringerten Merkmalsmenge neu trainiert, um die Gewichtungen neu zu schätzen, woraufhin erneut die Merkmale mit den geringsten Gewichten eliminiert werden (1).

Eine Möglichkeit ist die *Aggregation*, bei der Daten zusammengefasst werden. Die Aggregation kann zeilen- oder spaltenweise durchgeführt werden. Zeilenweise können Daten durch Mittelwerte ersetzt oder geeignete Gesamtwerte berechnet werden. Spaltenweise versucht man, aus mehreren Attributen ein gemeinsames Attribut zu machen, z.B. kann das Attribut „Datum“ aus den drei Attributen „Tag“, „Monat“ und „Jahr“ zusammengefasst werden (11). Die *Dimensionsreduktion* kann durchgeführt werden, um einerseits irrelevante Attribute zu vernachlässigen und andererseits relevante Attribute einzugliedern (11). Relevante Attribute können dabei entweder schrittweise der Zielmenge zugewiesen werden oder irrelevante Attribute werden schrittweise aus der Gesamtmenge eliminiert (11). Irrelevante nominale Attribute sind solche, die beispielsweise keine neuen Informationen beitragen. Ist ein Datensatz vorhanden, in dem ausschließlich Daten über Frauen repräsentiert sind, so ist das Attribut „Geschlecht“ für die Analyse irrelevant (11)). Weiterhin gilt bei nominalen Attributen, dass bei der Erstellung eines Entscheidungsbaumes kein Erkenntnisgewinn zu erwarten ist, wenn alle Werte eines Attributes unterschiedlich sind. Allerdings sollten diese Werte nicht eliminiert werden, wenn Algorithmen wie der k-Nearest Neighbour angewandt werden sollen, da im Nachhinein bei der Anwendung durchaus Daten in das Modell eingepflegt werden können, die einen der vorhandenen Werte enthält (11). Soll eine Dimensionsreduktion auf metrischen Daten durchgeführt werden, bieten sich Verfahren wie die Hauptachsentransformation oder multidimensionale Skalierung an (11). Eine weitere Möglichkeit der Datenreduktion ist die *Datenkompression*, bei der die Anzahl an Attributen verringert wird, indem die Daten derart transformiert oder codiert werden, dass einige Attribute zusammengefasst werden können (11). Des Weiteren kann eine repräsentative Teilmenge des Datensatzes verwendet werden. Diese Art der Reduktion wird *Numerische Datenreduktion* genannt (11). Welche Möglichkeiten es gibt, diese Stichprobe zu bilden, wird in (11) erläutert. Die numerische Datenreduktion kann außerdem mithilfe von linearer Regression durchgeführt werden (11). *Stratified Sampling* ist ein geeignetes Verfahren für Klassifikationsaufgaben, da in den Teilmengen die Häufigkeitsverteilung des Zielattributes erhalten bleibt. Dadurch wird dafür gesorgt, dass alle möglichen Werte ausreichend vertreten sind, um sinnvolle Klassenvorhersagen auf Basis der Beispieldaten machen zu können (11).

Ausgehend von der Annahme, dass in den meisten Fällen die Datengrundlage in Form von Spalten, Zeilen und Merkmalwerten vorliegen, führt (27) drei grundlegende Operationen für einen Datenreduktionsprozess an: Das Löschen einer Spalte (Attribut), das Löschen einer Zeile (Stichprobe/Datensatz) und das Reduzieren der Anzahl der Werte einer Spalte, also das Glätten eines Merkmals. Durch diese Operationen wird die Wahrung des Charakters der ursprünglichen Daten angestrebt (27). Weitere Operationen neben der Reduzierung durch Löschung, beinhalten bspw. das Zusammenfügen von Merkmalen. Sind die Attribute Größe und Gewicht eines Menschen gegeben, können diese zu dem Attribut

„BMI“ verrechnet werden. Die Qualität der Ergebnisse wird dabei nicht beeinträchtigt, sondern ggf. in einigen Anwendungen sogar verbessert (27). Um sicher zu sein, welche Auswirkungen das Löschen von Zeilen, Spalten oder Werten hat, können verschiedene Parameter für einen Vergleich herangezogen werden.

Ein bekanntes Verfahren für die Merkmalsauswahl ist Principal Component Analysis (PCA). Durch das Verfahren werden Attribute mit der größten Variabilität erfasst, indem alle Attribute derart umgewandelt, dass Hauptkomponenten oder neue Attribute entstehen, die nicht miteinander korrelieren, eine große Varianz in der Datenmenge enthalten und über Gewichtungsfaktoren auf das ursprüngliche Attribut zurückgeführt werden können. Attribute mit geringem Gewichtungsfaktor in ihren Hauptkomponenten werden aus der Datenmenge gelöscht (32).

2.3.3 Arten und Verfahren der Datentransformation

Der verfügbare Datensatz ist häufig heterogen und enthält mehrere Datentypen (1). Welche Datentypen es gibt, wurde bereits behandelt. In diesem Abschnitt geht es darum, wie sie in einen anderen Typ transformiert werden können, der für das gewählte Analyseverfahren gegebenenfalls besser geeignet ist (50). (1) merkt an, dass es am besten sei, den Algorithmus für die Analyse an die vorliegende Kombination von Datentypen anzupassen, statt sie zu transformieren. So kann vermieden werden, dass möglicherweise die Darstellungsgenauigkeit und die Ausdruckskraft der Daten beeinträchtigt wird. Jedoch werden in dieser Arbeit bereits vorhandene Algorithmen und Standardverfahren untersucht, weshalb die Möglichkeiten der Transformationen hier Vorrang haben. Nicht zuletzt ist die Erstellung eines speziellen Algorithmus zeitaufwendig und in manchen Fällen unpraktisch (1).

Für viele Data-Mining-Verfahren, wie z. B. Algorithmen zur Erstellung von Entscheidungsbäumen werden kategorische Daten benötigt (7). Liegen die Daten in numerischer Form vor, so können sie durch *Diskretisierung* in den kategorischen Datentyp transformiert werden (1; 7; 11). Dabei wird der Bereich eines Attributes in n Bereiche eingeteilt und alle Attributwerte einem dieser Bereiche zugeordnet (1), demnach wird eine Intervallbildung durchgeführt (7). Ein einfaches Beispiel ist die Einteilung der Werte „Alter“ in Altersgruppen. (1) merkt an, dass Abweichungen innerhalb eines der n Bereiche nach der Diskretisierung nicht mehr unterscheidbar sind und dadurch gegebenenfalls Informationen für die Datenanalyse verloren gehen können. Eine Herausforderung der Diskretisierung besteht darin, die Intervallgrenzen festzulegen (7). Es ist möglich, dass die n Bereiche je nach Festlegung des Wertebereichs ungleichmäßig mit Daten befüllt werden. Teilt man die Bereiche gleich groß ein, kann eben dieses Problem der ungleichmäßigen Verteilung eintreten (1). Ein weiteres Problem geht damit einher, dass Attributwerte, die nah beieinander liegen, durch eine ungünstige Wahl der Intervallgrenzen in unterschiedliche Intervalle eingeordnet werden, wodurch die Ähnlichkeit nach der Diskretisierung nicht mehr als derart eindeutig gehandhabt wird (7). Eine weitere Möglichkeit der Intervallbildung besteht darin, gleich tiefe Bereiche so auszuwählen, dass jedem Bereich die gleiche Anzahl von Datensätzen zugewiesen wird (1; 7). Dabei werden die Werte eines Attributes zunächst sortiert und daraufhin in gleichgroße Bereiche geteilt (1). Dieser Ansatz führt allerdings zu ähnlichen

Problemen, wie der Ansatz, gleich große Intervalle zu bilden: Werte, die nahe beieinander liegen, werden voneinander getrennt. Dies betrifft vor allem die Anwendung von Klassifikationsverfahren, da Klassifizierungen bei der Bestimmung von Intervallgrenzen nicht berücksichtigt werden (7). (7) schlägt daher für die Intervallbildung Methoden der Klassifizierung vor. Weiterhin kann statt einer globalen Diskretisierung, d.h. eine endgültige Transformation aller Werte wie gerade beschrieben, eine lokale Diskretisierung durchgeführt werden. Bspw. kann der TDIDT-Algorithmus derart angepasst werden, dass jedes kontinuierliche Attribut an jedem Knoten des Entscheidungsbaumes in ein kategorisches Attribut umgewandelt werden kann (7). Ein weiteres Verfahren ist das ChiMerge, wobei es sich um eine statistische Technik handelt, die anhand einer Häufigkeitstabelle aller Werte geeignete Intervalle bestimmt, die als „statistisch unterscheidbar“ gelten (7). Eine beispielhafte Anwendung kann in (7) nachgelesen werden. (10) setzen außerdem Cluster-Algorithmen für die Diskretisierung ein, indem die Intervalle nach einer Clusterbildung erstellt werden, und zeigen, dass diese Methode eine bessere Leistung erzielt als die Einteilung in gleich große Intervalle oder auf Häufigkeit basierte Intervalle. Für die Bildung von Intervallen bieten sich automatische Verfahren, bspw. „Binning“ an (11).

Abgesehen davon, dass die Diskretisierung für viele Algorithmen notwendig ist, bringt diese Art der Transformation weitere Vorteile mit sich. Zum einen wird der Speicherbedarf verringert, da die Anzahl der Werte verkleinert wird, zum anderen wird die Verständlichkeit und Interpretation der Daten erleichtert sowie die Lerngeschwindigkeit und Leistung von Data-Mining-Verfahren erhöht (46).

Umgekehrt können kategorische Daten auch in numerische Daten transformiert werden. Dieses Verfahren wird *Binarisierung* oder *Binärcodierung* genannt (1; 11). Dies ist mit nahezu allen kategorischen Daten möglich. Dabei werden für das Attribut mit n verschiedenen Werten, n binäre Attribute erstellt. In jedem dieser neuen n Attribute nimmt genau ein Wert die 1 an, alle anderen Werte die 0 (1). Ein Nachteil dabei ist, dass der Datensatz durch die Ersetzung eines Attributes durch mehrere Attribute unter Umständen enorm vergrößert wird (11). Ordinale Daten können ebenfalls in numerische, bzw. metrische Datentypen transformiert werden. Der Unterschied zu einer Binärcodierung besteht darin, dass nicht nur zwei, sondern beliebig viele numerische Werte genutzt werden können, durch die ein ordinaler Wert ersetzt wird. Als Beispiel kann das Attribut „groß“ durch den numerischen Wert „1“ ersetzt werden, das Attribut „mittel“ durch „0,5“ und „klein“ durch „0“. Bei der Wahl der jeweiligen numerischen Werte gibt es Spielräume, wodurch willkürlich gesetzte Werte zu unterschiedlich großen Abständen zwischen den Werten führen können, die das Ergebnis der Analyse beeinflussen können (11).

2.4 Data Mining

Data Mining wird bekanntermaßen zur Lösung verschiedener Probleme eingesetzt. Diese Probleme unterscheiden sich einerseits in ihrem Ziel und andererseits in ihren Vorbedingungen für den Einsatz. Welche Unterschiede zu beachten sind und welche Ziele die jeweiligen Probleme haben, werden in diesem Kapitel erläutert. Diese Informationen sind für den Aufbau des Entscheidungsmodells notwendig, da sich aus den spezifischen Verfahren unterschiedliche Anforderungen ergeben. Sowohl an die Datengrundlage als auch an die Datenvorverarbeitung und die Datenanalyse an sich. Zunächst erfolgt in Kapitel 2.4.1 eine allgemeine Einführung in die verschiedenen Problemarten und Data-Mining-Verfahren. In den darauffolgenden Teilkapiteln wird gezielter auf diese Problemarten und auf einsetzbare Verfahren eingegangen.

2.4.1 Einführung in Data-Mining-Verfahren

Zurzeit wird im Data Mining auf zwei grundlegende Arten kategorisiert. Zum einen wird zwischen überwachtem und unüberwachtem Lernen unterschieden. Für überwachtes Data Mining sind gekennzeichnete Trainingsdaten notwendig, anhand derer eine Funktion abgeleitet wird. Diese Funktion basiert auf verallgemeinerte Beziehungen zwischen gekennzeichneten Ein- und Ausgabevariablen. Sie wird verwendet, um die Ausgangsvariabel für neue, nicht gekennzeichnete Eingabevariablen vorherzusagen (32). Wichtig für überwachtes Lernen ist eine ausreichende Anzahl gekennzeichneter Datensätze, um eine geeignete Funktion abzuleiten (32). Das unüberwachte Lernen wird dagegen mit nicht gekennzeichneten Datensätzen durchgeführt. Dabei wird versucht, Muster in den Daten anhand derer Beziehungen zueinander zu finden(32).

Zum anderen wird nach dem konkreten Anwendungsziel kategorisiert (siehe Abb. 3). Jedoch scheint bei der Auswahl eines Verfahrens nicht nur das Ziel der Anwendung von Bedeutung zu sein, sondern dessen Einsetzbarkeit. Beispielsweise können Klassifizierungsaufgaben ebenfalls mit einigen Regressionsalgorithmen gelöst werden (32).

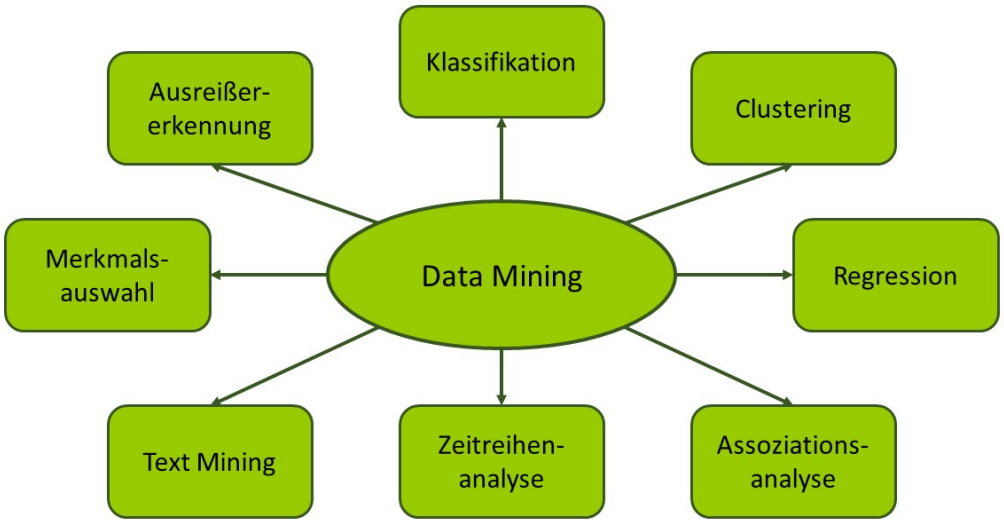

Abbildung 3: Kategorisierung nach Data-Mining-Aufgabe (32)

2.4.2 Assoziationsanalyse

Die Assoziationsanalyse ist eine der bekanntesten Methoden der Musterextraktion (2). Das übergeordnete Ziel der Assoziationsanalyse ist es, Korrelationen zwischen Objekten in Form von Assoziationsregeln zu finden (53). Häufig wird dies in der Praxis für eine Warenkorbanalyse angewandt, indem die Frage beantwortet wird, welche Produkte häufig gemeinsam gekauft werden (11). Diese Assoziationsregeln können als Verbindungen zwischen Attributen verstanden werden, die beispielsweise Aussagen der Art "Wenn Attribut A vorhanden ist, ist häufig auch Attribut B vorhanden" treffen (13). Die Assoziationsanalyse gehört zum überwachten Lernen (11).

Das Finden von Assoziationsregeln beinhaltet zwei Schritte: Zunächst werden häufige Objekte aus der Datenmenge identifiziert und daraufhin wertvolle Regeln aus der Menge häufiger Objekte extrahiert (12).

Für die Assoziationsanalyse werden zwei spezielle Kenngrößen verwendet, um Aussagen über die abgeleiteten Assoziationsregeln zu machen: Support und Konfidenz (12; 16). Der Support beschreibt die relative Häufigkeit eines Objekts in der gesamten Datenmenge (3). Je höher der Support einer Objektmenge ist, desto wichtiger ist diese Menge. Die Konfidenz enthält Informationen zu der Mächtigkeit und Stärke einer Regel (16). Sie gibt demnach einen Rückschluss auf die realistische Möglichkeit der Regeln an.

Eines der Standardverfahren der Assoziationsanalyse ist der Apriori-Algorithmus, für den eine Vielzahl von Varianten existiert (12). Der Algorithmus geht davon aus, dass jedes Objekt einer häufigen Objektgruppe selbst häufig ist (16). Das Ziel dieses Algorithmus ist es, Frequent Itemsets (Objektgruppen) in der Menge aller Items (Objekte) zu finden (11). Diese Frequent Itemsets sind Objekt-Mengen, deren relative Häufigkeit einen vorgegebenen Schwellwert überschreitet. Das Verfahren beinhaltet zwei Schritte. Zuerst werden Frequent Itemsets mit ausreichendem Support gesucht. Der Support ist dabei ein Maß für

die Häufigkeit, mit der die Kombination aus Vor- und Nachbedingungen einer bestimmten Regel in den Datensätzen auftritt (11; 53). Die Messung erfolgt durch einen reellen Wert zwischen 0 und 1, der angibt, welcher Anteil der Gesamtmenge von einer Regel erfasst wird (53). Der zweite Schritt erzeugt Assoziationsregeln aus allen zuvor gefundenen Frequent Itemsets (11; 53).

Für die Anwendung des Apriori-Algorithmus sind binäre Attribute geeignet, auf die die erzeugten Assoziationsregeln angewendet werden können (11). Durch eine Binarisierung können allerdings Regeln entstehen, die unsinnig sind, weshalb geeignete Filter angewandt werden sollten, um die Datensätze zu bereinigen. Zwingend notwendig sind binäre Attribute nicht. Auch ein Attribut, das beispielsweise drei Ausprägungen hat kann mit dem Apriori-Algorithmus behandelt werden. Bei der Berechnung des Supports und der Konfidenz wird gezählt, wie oft das Attribut die entsprechende Ausprägung hat (11).

Zum Nachteil kann eine hohe Laufzeit werden, wenn der Algorithmus bei jeder Iteration den gesamten Datensatz durchlaufen muss (12; 16). Zudem müssen die Schwellwerte für den Support und die Konfidenz gewählt werden. Eine ungeschickte Wahl kann zu einer hohen Anzahl Regeln führen. Aus diesem Grund gibt es einige „Verbesserungen“ des Apriori-Algorithmus (12; 53) und Alternativen, wie bspw. der AprioriTid (36), Apriori-Hybrid (60), BitApriori (63), Dynamic Itemset Counting (8) oder Frequent-PatternTree (23).

Des Weiteren gibt es das Verfahren Frequent Pattern Growth (FP-Growth), das als zweiter Hauptalgorithmus in der Assoziationsanalyse gilt (61). Der FP-Growth verzichtet auf die Generierung von Frequent Itemsets (11), was den Algorithmus zeitlich effizienter macht als den Apriori-Algorithmus (27; 61). Ausgehend von der relativen Häufigkeit der Items wird ein gerichteter Graph erzeugt, aus dem die Itemsets abgelesen werden können (11).

Es existieren zudem eine Reihe von Enumeration-Tree-Algorithmen, die für die Suche nach häufigen Mustern eingesetzt werden können (1). (63) geben einen ausführlichen Überblick über weitere Algorithmen für die Assoziationsanalyse.

2.4.3 Klassifikation

Das Klassifikationsproblem wird dem überwachten Lernen zugeordnet (11), da ein Beispieldatensatz verwendet wird, um die Struktur der Klassen zu lernen (1). Zunächst sind demnach Testdaten vorhanden, denen bereits Klassen zugeordnet wurden, woraufhin neue Datensätze anhand der vorhandenen Beispiele klassifiziert werden (11).

Ein Ansatz bei der Klassifikation besteht darin, alle Datensätze der Testmenge zu speichern und die Klasse eines neuen Objektes vorherzusagen, indem aus der bekannten Menge das ähnlichste Objekt gesucht und dessen Klasse übernommen wird. Dieses Vorgehen wird *instanzenbasiert* genannt (11). Eine andere Möglichkeit ist die *modellbasierte* Klassifizierung. Dabei wird ein Modell mithilfe der Beispieldatensätze berechnet, woraufhin diese Datensätze nicht weiter benötigt werden, da die Datenanalyse mithilfe des Modells durchgeführt wird (11).

Ein besonders wichtiger Schritt im Klassifizierungsprozess ist die Auswahl relevanter Merkmale. Werden irrelevante Merkmale gewählt, wird in der Regel die Genauigkeit des Klassifizierungsmodells beeinträchtigt (1).

Laut (1) besteht ein Klassifikationsverfahren typischerweise aus zwei Phasen:

1. Trainingsphase: Das Trainingsmodell wird aus den Trainingsdaten erstellt.
2. Testphase: Die Klassen neuer Datensätze werden bestimmt.

Bekannte Verfahren für Klassifikationsaufgaben sind beispielsweise der *Naive-Bayes-Algorithmus*, der *k-Nearest Neighbour*, die *lernende Vektorquantisierung*, die *lineare Diskriminanzanalyse* und *Entscheidungsbäume*. Jedes Verfahren hat Vor- und Nachteile und ist für bestimmte Datenarten besonders geeignet oder ungeeignet. Die speziellen Eigenschaften der Verfahren bestimmen maßgeblich die erforderliche Datengrundlage.

Der *Naive-Bayes-Algorithmus* ist ein wahrscheinlichkeitsbasiertes Verfahren, dessen Ziel es ist, die wahrscheinlichste Klasse vorherzusagen (47). Bei diesem Verfahren wird nicht mit Trainingsdaten gelernt, das heißt, es findet kein Training statt. Stattdessen wird die Klasse direkt aus den Trainingsdaten berechnet. Dabei wird von der Annahme ausgegangen, dass alle Attribute unabhängig voneinander sind (11). Dieses Verfahren ist laut (11) nicht für metrische Daten geeignet, da folgende Probleme auftreten:

1. Wenn zwei Werte nahezu gleich sind, bspw. 64 und 65, werden sie von dem Verfahren als unterschiedliche Ausprägungen angesehen. Dadurch werden viele kleine Zahlen entstehen, wenn der Naive Bayes die relativen Häufigkeiten der Ausprägungen zählt.
2. Das Verfahren scheitert spätestens dann, wenn ein Wert vorhergesehen werden soll, der im Testdatensatz nicht direkt aufgeführt ist, bspw. der Wert 63. Dabei ist es egal, wenn die Werte 62 und 64 vertreten sind.

Sind allerdings metrische Daten vorhanden, liegt die Lösung für das Problem in der *Diskretisierung* der betroffenen Daten in ordinale Daten, indem Intervalle gebildet werden (7; 47). Demnach gilt es abzuwägen, ob das Verfahren bei metrischen Daten sinnvoll ist. Liegen bereits ordinale Daten vor, zeichnet sich der Naive Bayes durch seine Effizienz

aus, da der Trainingsdatensatz zur Bestimmung der relativen Häufigkeit lediglich einmal durchlaufen wird (11). Falls in den Daten Einträge fehlen, können diese zudem ignoriert werden (47). In der Praxis liefert das Verfahren außerdem gute Ergebnisse (7). Nachteilig ist allerdings, dass das Verfahren von der stochastischen Unabhängigkeit der Attribute ausgeht, was aber meistens nicht der Fall ist (11; 47).

Der *k-Nearest Neighbour* ist eine Modifikation des Nearest-Neighbour-Algorithmus, der ein zu klassifizierendes Objekt der Klasse zuordnet, dem das Objekt den Beispieldaten angehört, das ihm am ähnlichsten ist. Der k-Nearest-Neighbour betrachtet allerdings nicht nur den einen nächsten Nachbarn, sondern die k nächsten Nachbarn (47). Dieses Verfahren benötigt ein Abstandsmaß für die zu klassifizierenden Daten, woraufhin die Klasse vorhergesagt wird, die am häufigsten unter den k Nachbarn vertreten ist (11). Zwar setzen viele Implementierungen reellwertige Attribute voraus, jedoch ist eine Klassifizierung auch mit ordinalen und nominalen Attributen möglich. Das Zielattribut, als das Attribut, das vorhergesagt werden sollte, sollte dagegen ordinal oder nominal sein. Ist es jedoch ebenfalls reellwertig, so wird der Mittelwert der am Ende ausgewählten ähnlichsten Objekte berechnet (11). Das bedeutet, anstatt der Zuweisung der Klasse durch einen nominalen oder ordinalen Wert (z.B. „groß“, „mittel“ oder „klein“ für das Zielattribut „Größe“), wird ein gemittelter Wert aus den ähnlichsten Objekten verwendet (z.B. Mittelwert aus den Werten „150cm“ und „170cm“ des Zielattributes „Größe“ – dem neuen Objekt wird daraufhin der Mittelwert „160cm“ vorausgesagt) Je nachdem, welcher Datentyp vorliegt, wird das Abstandsmaß mit unterschiedlichen Ansätzen berechnet.

Bei dem k-Nearest-Neighbour-Verfahren ist der Vorverarbeitungsschritt der Standardisierung bzw. Normalisierung der Attribute sehr wichtig, um zu verhindern, dass Attribute, die aufgrund ihrer Maßskala hohe Werte besitzen (z.B. das Attribut „Gehalt“ im Vergleich zu dem Attribut „Alter“), zu stark dominieren. Dieses Problem tritt beispielsweise dann auf, wenn der euklidische Abstand berechnet wird. Je größer das Intervall bei der Normalisierung gewählt wird, desto größer werden die Abstände und somit der Einfluss des Attributes (11).

Der Vorteil dieses Verfahrens liegt darin, dass es einfach durchzuführen ist (11; 47) und für jeden Datentyp (metrisch, ordinal, nominal) geeignet ist.

Ein Problem, bzw. eine Fehlerquelle kann darin liegen, dass vorgegeben werden muss, wie viele Nachbarn betrachtet werden sollen. Demnach muss k vorgegeben werden (11). Je nachdem, wie k gewählt wird, können unterschiedliche Klassen vorhergesagt werden. Aus diesem Grund ist es ratsam, verschiedene k zu benutzen und die Ergebnisse der unterschiedlichen Varianten miteinander zu vergleichen (11). Weiterhin ist die Wahl großer k sinnvoll, da dadurch der Einfluss von Ausreißern verringert wird (27). Zudem liegt eine Herausforderung in großen Datensätzen. Da für jeden Datensatz die Ähnlichkeit zu einem neuen Datensatz berechnet werden muss, kann die Laufzeit zu Problemen führen (11; 27). Positiv zu vermerken ist dabei allerdings, dass das Verfahren bei großen k (also bei einer größeren Menge ähnlicher Objekte) auch bei verrauschten Trainingsdaten sehr gut arbeitet, da gleichzeitig eine Glättung der Daten durchgeführt wird (11; 27). (27) weist allerdings zusätzlich darauf hin, dass bei großen k die Gefahr steigt, die Lokalität der Schätzung zu „zerstören“, da zunehmend entfernte Objekte in die Berechnung aufge-

nommen werden. Weiterhin merken (11) an, dass ein Problem darin liegen kann, dass alle Attribute in die Berechnung eingehen, was damit einhergeht, dass der Algorithmus „stark an Zuverlässigkeit verliert“, wenn nicht alle Attribute für den Vergleich relevant sind. Die Autoren schlagen daher vor, den Attributen jeweils Gewichte zuzuweisen, inwieweit sie für die Bestimmung der Ähnlichkeit relevant sind. Demnach wird das Abstandsmaß mit gewichteten Attributwerten berechnet. Eine Alternative dazu ist, irrelevante Attribute von vornherein auszuschließen (11). (11) schlagen als weitere Alternative für dieses Problem das Verfahren *Shepard's Method* vor. Sinn dieses Verfahrens ist es, den einzelnen Datensätzen ein Gewicht in Abhängigkeit ihres Abstandes zu dem neuen Datensatz zuzuweisen. So können alle Trainingsdatensätze genutzt werden, da Datensätze mit einem großen Abstand kaum ins Gewicht fallen. Ein Nachteil könnte allerdings in dem hohen Rechenaufwand liegen, wenn es zu der Vorhersage der richtigen Klasse kommt (11). Das Verfahren kann sowohl für diskrete als auch für reelle Zielattribute durchgeführt werden (11).

Zusammenfassend ergeben sich drei grundlegende Herangehensweisen, den k-Nearest Neighbour anzuwenden, wenn nicht jedes Attribut von Relevanz für die Entscheidung ist:

- Irrelevante Attribute von vornherein aus den Berechnungen ausschließen.
- Attribute vor den Berechnungen nach ihrer Relevanz gewichten (d.h. die Attribute gehen gewichtet in das Abstandmaß ein).
- Shepard's Method: Datensätze nach der Abstandsberechnung gewichten, um die wahrscheinlichste Klasse zu bestimmen.

Die *lernende Vektorquantisierung* ist ein heuristisches Verfahren, das zu einem gegebenen Datensatz für jede Klasse genau ein Prototyp bestimmt (47). Im Gegensatz zum Nearest-Neighbour-Verfahren, wird nicht der gesamte Datensatz durchsucht, sondern nur eine Menge von repräsentativen Datensätzen. Mit diesen Repräsentanten wird die Klassifikation wie bei dem Ansatz des Nearest-Neighbour-Verfahrens durchgeführt (47).

Weitere Verfahren für Klassifizierungsaufgaben sind *Entscheidungsbäume* (47), die laut (Priyam et al. 2013) zu den am häufigsten verwendeten Klassifikationsverfahren gehört. Diese werden verwendet, um Klassifzierungsregeln aufzustellen. Dies geschieht anhand eines Trainingsdatensatzes (7). Ein leistungsfähiger Algorithmus ist der Top-Down Induction of Decision Trees (TDIDT), der bereits in Verbindung der Diskretisierung von Daten gebracht wurde (7). Zu den bekanntesten Klassifizierungssystemen, dessen Grundlage der TDIDT bildet, gehören der ID3- und der C4.5-Algorithmus (7), die in diesem Kapitel beschrieben werden.

Für Entscheidungsbäume sind nominale bzw. kategorische Attribute notwendig, da durch metrische Attribute, je nach Wertebereich, eine zu große Anzahl an Kanten generiert werden muss. Mitunter kann das Erstellen eines Entscheidungsbaumes mit metrischen Attributen unmöglich sein (11). Die Begründung liegt allein in der hohen Anzahl möglicher Werte. Daher ist es sinnvoll, den metrischen Datentyp zu transformieren, indem die Daten bspw. in Intervalle zusammengefasst werden. Eine weitere Möglichkeit ist das Nutzen von Schwellenwerten. Dabei wird bei kontinuierlichen Attributwerten an jeder Kante erfragt, ob der festgelegte Wert unter- oder überschritten wird (7; 11).

Die Attribute werden als Knoten des Entscheidungsbaumes dargestellt. Die Knoten können im Allgemeinen beliebig angeordnet werden, weshalb verschiedene Entscheidungsbäume aus denselben Daten generiert werden können (11). Es gibt zwei verschiedene Arten von Knoten: Einerseits handelt es sich um ein Blatt, das eine Klasse angibt, andererseits handelt sich um einen Entscheidungsknoten, der einen Test für ein einzelnes Attribut darstellt (27). Entscheidungsbäume können sich mitunter in ihrer Größe unterscheiden, je nachdem, welche Attribute gewählt wurden. Erstrebenswert ist ein möglichst kompaktes Modell, das eine Klassenzuordnung nach wenigen Fragen liefern kann (11). Schlussfolgernd ist die Auswahl geeigneter Attribute für den Erfolg eines Entscheidungsbaumes von großer Bedeutung. (11) beschreiben drei Möglichkeiten, diese zu bestimmen. Zum einen ist die manuelle Auswahl möglich, was jedoch nur bei kleinen Attributmengen sinnvoll ist. Des Weiteren können die Attribute zufällig durch einen Zufallsgenerator gewählt werden. Mitunter kann dies aber zu unübersichtlichen und schlecht interpretierbaren Bäumen mit sehr langen Pfaden führen (11). Das dritte Vorgehen sieht die Berechnung geeigneter Attribute vor. Dabei wird automatisch von einem Algorithmus nach einem Attribut gesucht, das einen kompakten Baum erzeugt (11).

Es sind einige Verfahren vorhanden, mit denen Entscheidungsbäume generiert werden können. Der *ID3-Algorithmus* geht von der Annahme aus, dass die Komplexität des Entscheidungsbaumes stark mit dem sogenannten Informationsgehalt des Attributes zusammenhängt (27). Daher wählt der Algorithmus das Attribut aus, das den größten Informationsgewinn liefert, indem der Informationsgehalt der Attribute berechnet wird (11). Eine Alternative zum Informationsgehalt ist die Berechnung des Gini-Index (11).

Ein Nachteil des ID3-Algorithmus liegt darin, dass numerische Attribute nicht behandelt werden können. Zudem normalisiert er nicht den Informationsgewinn, wodurch Attribute favorisiert werden, die viele verschiedene Werte haben (11).

Beide Nachteile des ID3-Algorithmus werden durch den *C4.5-Algorithmus* aufgehoben. Dieser ist ein Nachfolger des ID3 und basiert auf der CLS-Methode von Hunt und kann einerseits numerische Attribute behandeln, indem der Algorithmus diese in Intervalle unterteilt und sie dadurch in ordinale Attribute umwandelt (11). Für diese Transformation werden Attribute bevorzugt, deren Einteilung in Klassen zu einer geringen Klassenentropie führt, das heißt, die Mehrheit der Datensätze sollten einer einzigen Klasse angehören (27). Andererseits normalisiert der C4.5 den Informationsgewinn (11).

Bei der Vorverarbeitung des Datensatzes ist für den Einsatz des C4.5-Algorithmus zu beachten, dass dieser von einem vollständigen Datensatz ausgeht. Fehlende Daten müssen behandelt werden (27). (27) schlägt vor, dass entweder alle Datensätze mit fehlenden Daten verworfen werden oder ein neuer Algorithmus definiert werden muss, der mit fehlenden Daten umgehen kann.

Sowohl der ID3-, als auch der C4.5-Algorithmus können nicht auf große Datensätze angewandt werden, da die Speicherung der Daten, bzw. eines Teils der Daten, notwendig ist (**?**). Eine Alternative für große Datensätze ist der von (51) entwickelte Algorithmus SPRINT. SPRINT steht für „Scalable Parallelizable Induction of Decision Tree“ (31). Der Algorithmus partitioniert den Trainingsdatensatz rekursiv mit der Breadthfirst-Greedy-Technik (**?**) und kann sowohl kontinuierliche als auch kategorische Attribute verarbeiten.

Nachteilig ist bei dem SPRINT die hohe Komplexität des entstehenden Modells (31).

Entscheidungsbäume sind laut (11) aufgrund ihrer guten Interpretierbarkeit sehr beliebt. Ein weiterer Vorteil ergibt sich aus der unkomplizierten und leichten Implementierung der Verfahren. Ein Nachteil dagegen ist, dass Entscheidungsbäume nicht weiterentwickelt werden können, sobald sie erstellt wurden. Sobald neue Daten gewonnen werden, muss ein neuer Entscheidungsbaum generiert werden. Weiterhin gehen einige Probleme mit dem Entscheidungsbaumlernen einher. Die Gefahr besteht, dass der Entscheidungsbaum die Trainingsdaten auswendig lernt. Das Modell wird dann zwar die Trainingsdaten richtig klassifizieren, mit den Testdaten allerdings Schwierigkeiten haben. Dieses Problem wird „Overfitting“ genannt. Verhindert werden kann dies durch ein künstliches Verkürzen des Baumes mithilfe von Pruning-Verfahren (3; 11). Durch Pruning (Beschneidung) werden Teilbäume durch Blätter ersetzt, wodurch der Entscheidungsbaum vereinfacht wird (27). Ein alternatives Verfahren, das sogenannte Postpruning, verfolgt das Ziel, einen vollständigen Entscheidungsbaum zu erzeugen und daraufhin generierte Unterbäume durch ein Blatt zu ersetzen (7; 11; 27). Der Postpruning-Ansatz wird von dem C4.5-Algorithmus verfolgt (27).

Die *lineare Diskriminanzanalyse* ist laut (47) ein effizientes Klassifikationsverfahren, das für korrelierte Merkmale geeignet ist. Das Verfahren setzt allerdings eine näherungsweise Gaußverteilung im Merkmalsraum voraus. Daher ist es nicht für stark nichtlineare Klassengrenzen geeignet, die aus anderen Verteilungen resultieren (47).

2.4.4 Clustering

Die Clusteranalyse gehört zum unüberwachten Lernen und das Ziel ist es, Objekte auf Grundlage ihrer Ähnlichkeit und Unähnlichkeit zu geeigneten Mengen zusammenzufassen (11; 56). Clusterverfahren sind dann angebracht, wenn in den vorliegenden Daten, im Gegensatz zu Klassifikationsaufgaben, keine Klasseninformationen vorhanden sind (47). Die Objekte, die innerhalb eines Clusters liegen, sollen eine möglichst hohe Ähnlichkeit zueinander aufweisen, während die Ähnlichkeit zu Objekten anderer Cluster möglichst gering sein soll (11; 52; 56). So sollen Strukturen in großen Datensätzen aufgedeckt werden (56). Es wird davon ausgegangen, dass ähnliche Objekte einen geringeren Abstand zueinander aufweisen als unähnliche Objekte (47). Demnach wird das Abstandsmaß benötigt, mit dessen Hilfe die Ähnlichkeit von Objekten quantifiziert werden kann (11).

Eine Clustermenge hat laut (11) vier Eigenschaften:

- Die Objekte der Clustermenge sind ein Teil der gegebenen gesamten Objektmenge.
- Die Qualität der Cluster soll maximiert werden.
- Kein Objekt wird mehreren Clustern zugeordnet.
- Jedes Objekt wird genau einem Cluster zugeteilt.

(11) merken im Hinblick auf die beiden letzten Eigenschaften an, dass es durchaus sinnvoll sein kann, diese unbeachtet zu lassen. So sei es bei Ausreißern sinnvoll, diese isoliert zu halten. Zudem können Objekte vorhanden sein, die nicht genau einem Cluster zugeordnet werden können und eine eindeutige Zuweisung das Analyseergebnis beeinträchtigen kann (56). Der Algorithmus Fuzzy-c-Means weist Objekten bspw. Anteile bzw. Wahrscheinlichkeiten an mehreren Clustern zu. Dieses Verfahren wird an späterer Stelle näher betrachtet.

(47) unterscheidet Clusterverfahren in „Sequentielle agglomerative hierarchische nichtüberlappende Clusteranalyse“ (SAHN) und „Sequentielle divisive hierarchische nichtüberlappende Clusteranalyse“ (SDHN). Der Unterschied beider Ansätze ist der, dass bei SAHN-Verfahren zunächst jedes Objekt ein Cluster bildet und diese einzelnen Cluster sukzessive paarweise miteinander verschmolzen werden. Diese Objektpaare müssen einen minimalen Abstand zueinander haben (47). Der Abstand kann durch den Minimalabstand (Single Linkage), Maximalabstand (Complete Linkage), mittleren Abstand (Average Linkage), Abstand der Zentren oder die Ward-Methode bestimmt werden (47). (56) vergleichen diese Techniken in Hinblick auf die Ergebnisse der Clusteranalyse miteinander. Die Cluster werden so lange zusammengefasst, bis alle Objekte einem Cluster angehören. Analog zu dem SAHN funktioniert der SDHN umgekehrt: Alle Objekte gehören zunächst dem gleichen Cluster an, das sukzessive in kleinere Cluster geteilt werden, bis jedes Objekt genau ein Cluster bildet (47).

Clusterverfahren lassen sich laut (11) in vier Unterklassen einteilen:

- Partitionierende Clusterbildung

- Hierarchische Clusterbildung
- Dichtebasierte Clusterbildung
- Clusterbildung mit Neuronalen Netzen

Partitionierende Clusterbildung

Partitionierende Clusterbildung zeichnet sich dadurch aus, dass zu Beginn eine zufällige Anfangspartitionierung von Clustern vorgenommen wird (11). Die Cluster werden durch den Medoid oder Centroid repräsentiert. Dabei handelt es sich um die Schwerpunkte der jeweiligen Cluster, die in diesem Kapitel an späterer Stelle näher erläutert werden. Auf Grundlage der Anfangspartitionierung werden die Objekte schrittweise zwischen den Clustern getauscht, so dass sich die Güte der Cluster (die zuvor festgelegt bzw. definiert wurde), verbessert. Beispielsweise wird dazu in jedem Schritt der Centroid (Schwerpunkt, der auch als „Repräsentant“ bezeichnet wird) berechnet. Anschließend werden alle Objekte dem Cluster zugeordnet, dessen Repräsentanten sie am nächsten sind. Das Verfahren schließt ab, wenn die Güte der Partitionierung sich nicht weiter verbessern lässt (11). Zu den partitionierenden Verfahren gehören bspw. der k-Means-Algorithmus, der k-Medoids-Algorithmus, PAM und CLARA (52).

Der *k-Means-Algorithmus* ist ein sehr bekanntes, iteratives Verfahren für die Clusteranalyse (62). Im ersten Schritt werden zufällig initiale Cluster gebildet und im zweiten Schritt werden die sogenannten Centroide der jeweiligen Cluster berechnet. Der *Centroid* ist der Schwerpunkt des Clusters (11). Dieser wird anhand der Euklidischen Distanz bestimmt (1). Anstatt zufällige Cluster zu bilden, können auch zufällige Centroide im ersten Schritt festgelegt werden. Außerdem können bereits „gute“ Centroide bestimmt werden, indem die einzelnen Centroide möglichst weit auseinander liegen, und somit eine hohe Unähnlichkeit aufweisen (11). Im dritten Schritt werden die Abstände aller Objekte zu den generierten Centroiden bestimmt und die Objekte werden demjenigen Centroid zugeordnet, dessen Abstand am geringsten ist. So werden die Cluster neu geordnet. Im nächsten Schritt werden die Centroiden erneut bestimmt und daraufhin die Objekte wiederum ihrem nächsten Centroiden zugeordnet (42). Das Vorgehen wird so lange iterativ durchgeführt, bis sich kein Objekt mehr einem anderen Centroiden zugehörig fühlt (11). Vorteile des k-Means-Algorithmus sind, dass die Anzahl an Iterationen vergleichsweise klein ist und sich das Verfahren leicht implementieren lässt, da lediglich Abstandsberechnungen und Neuzuordnungen genutzt werden (11). Auf der anderen Seite gibt es einige Nachteile. Zum einen ist das Verfahren abhängig von der Qualität der initialen Zerlegung, die zu Beginn durchgeführt wird (15). Eine ungünstige Zerlegung kann sich negativ auf die Anzahl der Iterationen auswirken. Demnach sind gute Initialcluster von großer Bedeutung. Zudem ist das Verfahren empfindlich gegen Rauschen und Ausreißern, da alle Objekte in die Berechnung des Centroiden eingehen. Ein Ausreißer zieht den Centroiden in seine Richtung und bewirkt somit eine Verzerrung des Clusters. Ein weiterer Nachteil ist der hohe Aufwand, da in jedem Schritt alle Distanzen neu berechnet werden. Es gibt allerdings Modifikationen des Verfahrens, die schneller zu einem guten Ergebnis führen, bspw. der von (42) entwickelte k-Means-MIND. Eine dieser Modifikationen sieht vor, dass die Centroide der neuen Cluster nach jeder einzelnen Umordnung neu berechnet werden und nicht erst dann,

wenn alle Objekte umgeordnet wurden. Der Vorteil dabei ist, dass nicht in jedem Schritt die Abstände aller Objekte neu berechnet werden müssen. Der Nachteil ist wiederum, dass die Geschwindigkeit des Verfahrens maßgeblich von der Anfangspartitionierung abhängt (11). Nicht zuletzt muss die Anzahl der Cluster vorgegeben werden (11), was nachteilig sein kann, wenn diese Anzahl ungünstig gewählt wird oder keine Vorstellung darüber vorhanden ist, wie viele Cluster es geben kann (41). Das Verfahren beantwortet nicht die Frage, was die optimale Clusteranzahl ist (52). (41), (55) und (54) beschäftigen sich mit der Auswahl der Clusteranzahl und schlagen Verfahren vor, die automatisch die Anzahl der Cluster erkennen. Weiterhin ist kritisch zu vermerken, dass der k-Means-Algorithmus konvexe Cluster liefert. Dadurch können bspw. Cluster mit beliebigen Formen nicht gefunden werden (62). Allerdings stehen für derartige Probleme andere Verfahren zu Verfügung, die einen dichtebasierten Ansatz verfolgen (11). Diese werden ebenfalls in diesem Kapitel beleuchtet.

Um den k-Means-Algorithmus zu verwenden, sind numerische Werte notwendig (24). Demnach müssen nominale und ordinale Attribute in numerische Werte umgewandelt werden. Zwar können Abstandmaße bei nominalen und ordinalen Daten mit der Hamming-Distanz berechnet werden, allerdings scheitert das Verfahren bei der Bestimmung der Centroiden mit dieser Datengrundlage (11). Eine Modifikation des k-Means-Algorithmus, der *k-Modes-Algorithmus* (24; 34), erlaubt dagegen nominale Attribute. Des Weiteren ist in der Datenvorverarbeitung die Normalisierung der Daten wichtig sowie die Verarbeitung von Ausreißern, damit die Cluster nicht verzerrt werden (11).

Im Gegensatz zum k-Means-Algorithmus ist der *k-Medoid-Algorithmus* unempfindlicher gegenüber Ausreißern. Bei diesem Verfahren wird nicht der Centroid eines Clusters berechnet, sondern der Medoid. Der Medoid zeichnet sich dadurch aus, dass es sich dabei um ein „reales“ Objekt aus der Datenmenge handelt, beispielsweise das Objekt, das dem Centroiden am nächsten ist (11). Der Algorithmus tauscht die Medoiden so lange untereinander aus, bis keine Verbesserung mehr möglich ist. Diese Verbesserung wird als „Kosten“ berechnet, die es zu minimieren gilt. Die Kosten sind die Summe aller Abstände zwischen den Objekten. Dabei gilt, dass ein Medoid nur durch ein anderes Objekt getauscht wird, wenn die Kosten gesenkt werden. Ist dies nicht der Fall, wird ein neues Objekt untersucht (11). Aufgrund der Nutzung des Medoiden als ein reales Objekt der Menge, ist der Algorithmus gegenüber Ausreißern weniger empfindlich, da kein arithmetisches Mittel berechnet wird (wie es bei dem Centroiden der Fall ist). Demnach kann ein Ausreißer das Cluster nicht verzerren. (25) schlagen eine Erweiterung des k-Medoid vor, die auf großen Datensätzen schneller arbeitet und dessen Ergebnisse mit steigender Datenmenge besser werden.

Einer der ersten k-Medoid-Algorithmen war *PAM:Partitioning Around Medoids* (29). Der PAM-Algorithmus sucht nach einem besten neuen Medoid, indem alle möglichen Tauschvarianten ausprobiert werden, um die größtmögliche Verbesserung zu erzielen (5). Der Tausch, der die geringsten Kosten herbeiführt, wird durchgeführt. Durch diese Ausführliche Suche werden sehr gute Cluster herbeigeführt, allerdings steigt die Laufzeit bei größeren Datenmengen an. Daher ist der Algorithmus eher für kleinere Datenmengen geeignet (11).

Ein weiteres k-Medoid-Verfahren ist das „Clustering Large Applications based on RANdomized Search“ (CLARANS) (40). Dieses Verfahren ist weniger gründlich als der PAM und liefert demnach schlechtere Ergebnisse, ist in der Praxis allerdings effizienter, da nicht alle Objekte nach einem potenziell neuen Medoiden durchsucht werden, sondern nur eine zufällige Teilmenge (11; 28).

Die Vorstufe des CLARANS ist das Verfahren „Clustering LARge Applications“ (CLARA), das von vornherein Cluster in einer Teilmenge bildet, auf die der PAM-Algorithmus angewandt wird. Dabei kann es passieren, dass in dieser zufälligen Teilmenge der optimale Medoid nicht aufgenommen wurde, weshalb das Verfahren mehrfach wiederholt und die Clusterbildung ausgewählt wird, die zu den geringsten Kosten führt (11).

Hierarchische Clusterbildung

Die hierarchische Clusterbildung zeichnet sich dadurch aus, dass eine Hierarchie von Clustern aufgebaut wird (56), bei der jeweils die Cluster zusammengefügt werden, deren Distanz zueinander minimal, also deren Ähnlichkeit am größten ist (11). Dadurch können übergeordnete Cluster gebildet werden. Durch eine Kombination von überwachten Entscheidungsbaumalgorithmen (Kapitel 2.4.3) und unüberwachten Clustering-Elementen lassen sich Entscheidungsbäume aus unklassifizierten Daten erstellen. Diese kombinierten Algorithmen werden als Entscheidungsbaum-Clustering (DTC) bezeichnet (47). Hierarchische Verfahren liefern Verbindungen zwischen den Clustern, was nützlich sein kann. Auf der anderen Seite haben sie Probleme bei dem Umgang mit Ausreißern und sind schlecht skalierbar und in der Praxis ergibt sich eine Beschränkung auf wenige tausend Elemente (11). Positiv ist die Anpassungsfähigkeit hierarchischer Methoden, da eine beliebige Anzahl von Gruppen gebildet werden können. Andererseits ist es schwierig, Änderungen an bereits erstellten Clustern vorzunehmen, sobald eine Vereinigung oder ein Teilprozess stattgefunden hat (56).

Dichtebasierte Clusterbildung

Dichtebasierte Clusterverfahren werden angewandt, wenn partitionierende Clusterverfahren scheitern, weil konvexe Cluster für die gegebenen Objekte nicht sinnvoll sind (11; 58). Beispielbild einfügen. Der dichtebasierte Ansatz fokussiert sich auf die Dichte der Punkte eines Clusters, die einen bestimmten Schwellenwert nicht unterschreiten darf (11). Die Punkte, bzw. Objekte, müssen eine minimale Anzahl an Nachbarn aufweisen, damit die Dichte den Vorgaben entspricht (37). Dadurch werden auch Cluster gefunden, die in ihrer Form sehr unterschiedlich sind und die Verwendung von Mittelpunkten nicht sinnvoll ist (11). Die Grundidee von dichtebasierten Clusterverfahren ist, freie Räume zwischen Clustern auf Grundlage geringer Dichte zu erkennen, durch die die Cluster getrennt werden (11). Der Abstand der Objekte kann bspw. durch die Euklidische Distanz oder die Manhattan-Distanz bestimmt werden (37).

Der *DBSCAN-Algorithmus* (14) ist ein bekanntes Verfahren für dichtebasiertes Clustering (11). Ein Vorteil des Verfahrens ist, dass die Clusteranzahl nicht vorgegeben werden muss, sondern von dem Algorithmus selbst bestimmt wird (11). Allerdings müssen für dieses Verfahren andere Parameter vorgegeben werden: Die minimale Anzahl Objekte, die

in der Nachbarschaft des betrachteten Objektes liegen müssen, sowie der Radius, in dem die zu berücksichtigenden Objekte liegen müssen (11).

Clusterbildung mit Neuronalen Netzen

Der *Fuzzy-c-Means-Algorithmus* wird angewandt, wenn die Grenzen zwischen zwei Clustern nicht eindeutig sind. Der k-Means kann scheitern, da dieser ein Objekt nur einem Cluster zuweisen kann, und dichtebasierte Clusterverfahren können scheitern, weil rigide entschieden wird, ob ein Objekt zu einem Cluster gehört oder nicht (11). Bei einem Datensatz, in dem alle Objekte nah beieinander sind, kann das Resultat ein einziges großes Cluster sein. Die Lösung für das Problem ist ein Algorithmus, der Objekte derart in Cluster ordnet, dass das Ergebnis nicht mehr disjunkt ist. Das bedeutet, dass ein Objekt mehreren Clustern zugehörig ist, zu unterschiedlichen Teilen. Ein Objekt kann demnach zu 10% Cluster 1 angehören und zu 90% Cluster 2. Folglich hat jedes Cluster mindestens ein Objekt inne, das möglicherweise aber eine geringe Zugehörigkeit aufweist (11). Bestimmt werden die Zuordnungen in einer Matrix. Die Trennschärfe der Cluster, also wie „weich“ oder „scharf“ die Clusterbildung erfolgt, wird durch einen Parameter festgelegt (11; 47).

3 Herausforderungen im Data Mining und Kategorisierungsansatz

Im folgenden Abschnitt werden Herausforderungen im Data Mining beleuchtet, die einen neuen Kategorisierungsansatz beeinflussen. Die derzeitige Kategorisierung in Abhängigkeit des Analyseziels, die in Kapitel 2.4 vorgestellt wurde, wird dabei bewusst nicht berücksichtigt. Vielmehr wird das Ziel verfolgt, zu erurieren, welche Eigenschaften die Verfahrensauswahl tatsächlich beeinflussen. Wie in Kapitel 2.4 bereits häufig erwähnt, haben Data-Mining-Verfahren in der Regel spezifische Voraussetzungen für die Anwendung, die erfüllt sein müssen.

Daraufhin werden Eigenschaften der Datengrundlage betrachtet, die einen Einfluss auf den Einsatz eines Data-Mining-Verfahrens haben können. Auf dieser Grundlage wird ein neuer Kategorisierungsansatz für die Auswahl geeigneter Data-Mining-Verfahren erarbeitet.

3.1 Herausforderungen im Data-Mining-Prozess

Im Data-Mining-Prozess wird der Anwender vor viele Herausforderungen gestellt. Für den neuen Kategorisierungsansatz sind zunächst diejenigen interessant, die für die Auswahl eines geeigneten Data-Mining-Verfahrens von Bedeutung sind.

Fluch der Dimensionalität

Von großer Bedeutung ist zunächst die Datenmenge. In der Literatur wird nicht ersichtlich, welche Menge „die richtige“ ist. Generell ist eine Datenmenge wünschenswert, die mehr Datensätze als Attribute aufweist. Diese Gegebenheit kann bspw. der Überanpassung (Overfitting) eines Verfahrens an die vorliegenden Daten verhindern. Bei einer geringen Anzahl Datensätze besteht die Gefahr, dass diese Daten vom Verfahren „auswendig gelernt“ werden und bei der Verarbeitung neuer Daten nicht gut funktioniert. Ein Begriff, der im Zusammenhang mit der Datenmenge häufig auftaucht, ist „Hochdimensionalität“. Damit sind Datenmengen gemeint, die eine hohe Anzahl an Attributen aufweisen und somit eine hohe Dimension haben. Für viele Verfahren, und somit für den Anwender, gilt dies als Herausforderung. (1) merkt bspw. an, dass abstandsbasierte Verfahren ihre Wirksamkeit mit zunehmender Dimensionalität verlieren. Problematisch sei eine hohe Dimensionalität vor allem durch eine große Anzahl irrelevanter Attribute. Diese Herausforderung wird „Fluch der Dimensionalität“ genannt (19). Um dem Fluch der Dimensionalität entgegenzuwirken, ist es demnach sinnvoll, relevante Attribute durch ein geeignetes Verfahren auszuwählen. Weiterhin ist ein „gutes“ Verhältnis von Attributen zu Datensätzen notwendig. Es gibt allerdings Verfahren, die trotz vergleichsweise wenig Datensätzen gute Ergebnisse liefern. Anscheinend ist es sinnvoll, die Anzahl an Datensätzen im Verhältnis zu der Dimensionalität in die Kategorisierung einzubeziehen.

Vorverarbeitung

Wie in Kapitel 2 ausführlich beschrieben, ist die Datenvorverarbeitung mit viel Arbeit verbunden. Die Herausforderung liegt darin, zu erkennen, welche Vorverarbeitungsschritte für

das einzusetzende Verfahren notwendig sind. Einige Verfahren können trotz fehlender Werte gute Ergebnisse liefern, andere werden durch fehlende Werte verfälscht. Weiterhin gilt vor allem für abstandsbasierte Verfahren, dass die Daten standardisiert werden müssen, um die richtigen Abstände berechnen zu können. Zudem stellt sich die Frage, in welcher Reihenfolge die Vorverarbeitungsschritte am sinnvollsten sind, um möglichst zeitsparend arbeiten zu können. Der Kategorisierungsansatz kann die Eigenschaften einiger Verfahren nutzen, um eine möglichst effiziente Vorverarbeitung zu ermöglichen.

Parametereinstellungen
Eine weitere Herausforderung, die sich im Hinblick auf die Verfahrensauswahl ergibt, ist die Einstellung von Parametern. Die Gefahren von parameterreichen Verfahren bestehen zum einen darin, dass eine nicht geeignete Einstellung dazu führt, dass der Algorithmus keine Muster in den Daten finden kann. Zum anderen kann eine fehlerhafte Einstellung sogar zu falschen Mustern führen, was ein weitaus größeres Problem darstellt. Parameterreiche Verfahren benötigen im Besonderen Expertenwissen, bzw. ein gutes Verständnis für die Auswirkungen der Parameter.

Speicherkapazität
In der praktischen Anwendung kann die Speicherkapazität ebenfalls eine Herausforderung darstellen. Sind dem Anwender keine geeigneten Speichermöglichkeiten verfügbar, beeinflusst dies dementsprechend die Verfahrensauswahl.

Analysedauer vs. Qualität In manchen Fällen muss eine Abwägung zwischen der Analysedauer und der Ergebnisqualität stattfinden.

Geschwindigkeit
Die Geschwindigkeit, in der Daten generiert werden, muss ebenfalls bei der Verfahrensauswahl berücksichtigt werden. Zum einen kann untersucht werden, welche Verfahren für Echtzeitanalysen geeignet sind, andererseits stellt sich die Frage, welche Verfahren neue Daten integrieren können. Entscheidungsbaumalgorithmen müssen bspw. mit jedem neuen Datensatz neu generiert werden. Dementsprechend ist der Einsatz eines solchen Algorithmus für Anwendungen weniger sinnvoll, wenn stetig neue Daten ergänzt werden.

3.2 Kategorisierung und Abhängigkeiten der Verfahren

3.2.1 Kategorisierungsansatz

Das Ziel ist es nun, einen Kategorisierungsansatz für die Auswahl von Data-Mining-Verfahren anzustreben. Zu diesem Zweck werden zunächst mögliche Einflüsse bestimmt, die sich aus der Recherche in Kapitel 2 und den Herausforderungen aus Kapitel 3.1 ergeben. In Kapitel 2.4 wurde bei der Betrachtung einiger Verfahren mehrmals aufgezeigt, dass zum Teil bestimmte Datentypen für den Einsatz vorausgesetzt werden. Dabei wird in der Literatur vor allem zwischen dem numerischen und dem kategorischen Datentyp unterschieden. Weiterhin ist zu beachten, dass zwischen Eingabe- und Ausgabevariablen unterschieden werden muss. Setzt man beispielsweise einen Entscheidungsbaum für die Analyse ein, können die Eingabevariablen sowohl numerische als auch kategorische Werte aufweisen. Die Ausgabevariable muss dagegen kategorisch sein. Ähnliches gilt für weitere Verfahren, die im nächsten Abschnitt genauer betrachtet werden.

Schlussfolgernd sind die ersten Einflüsse auf die Verfahrensauswahl zwei Datentypen: Datentyp der Eingabevariablen und Datentyp der Ausgabevariable. Dies setzt das Vorhandensein einer Ausgabevariable voraus. Weiterhin kann die Datenmenge als Einfluss berücksichtigt werden. Diese setzt sich aus der Attributmenge, der Dimension einer Datenmenge, und der Anzahl an Datensätzen zusammen. Eine große Anzahl von Datensätzen kann sich auf die Performance eines Algorithmus auswirken.

Beispielsweise durchläuft der Apriori-Algorithmus bei der Analyse jeden einzelnen Datensatz in jeder Iteration. Mit einer zunehmenden Anzahl an Datensätzen steigt die Analysedauer (Kapitel 2.4). Die Dimension eines Datensatzes kann ebenfalls als Einflusskriterium genutzt werden. Zum einen steigt mit einer höheren Dimension die Modellkomplexität (Kapitel 2.3.2). Zum anderen gilt für die Datenanalyse, relevante Attribute den irrelevanten Attributen vorzuziehen. In Kapitel 2.3.2 wurde bereits erläutert, dass die Nutzung irrelevanter Attribute störende Effekte mit sich bringen können und die Ergebnisqualität dadurch beeinträchtigen können.

Weiterhin sind für einige Verfahren, wie dem Naive Bayes, unabhängige Attribute notwendig. Diese Eigenschaften lassen schlussfolgern, dass die Attributmenge so informativ und dabei so gering wie möglich sein sollte. Aus der Recherche ist nicht hervorgegangen, wie groß die empfohlene Anzahl an Attributen bzw. Datensätze sein sollte. Häufig wird im Zusammenhang von Verfahren allerdings die Dimensionalität einer Datenmenge thematisiert. Bekannt ist bereits das Problem, wenn vergleichsweise viele Attribute und wenig Datensätze vorhanden sind. In einem solchen Fall wird die Datenmenge als hochdimensional bezeichnet. Die Herausforderung dahinter wird in Kapitel 3.1 als Fluch der Dimensionalität vorgestellt und kann nachweislich die Qualität einer Analyse beeinträchtigen. Eine bekannte negative Folge ist das Overfitting (Kapitel 2.4.3). Sind im Vergleich zu der Anzahl an Attributen wenig Datensätze vorhanden, besteht die Gefahr, dass Trainingsdaten durch das Modell auswendig gelernt werden. Dadurch werden neue Daten ungenauer den richtigen Klassen zugeordnet. Eine hochdimensionale Datenmenge bedeutet demnach, dass einerseits eine hohe Anzahl an Attributen vorliegt und vergleichsweise wenig Datensätze. Daraus folgt, dass die Einbeziehung des Verhältnisses beider Mengen im Hinblick

auf die Verfahrensauswahl sinnvoll erscheint.

Im Rahmen der durchgeführten Recherche konnte keine Angabe eines empfohlenen Verhältnisses ausgemacht werden. Vielmehr wird verallgemeinert, dass eine Datenmenge gewünscht wird, die deutlich mehr Datensätze als Attribute enthält und dass die Anzahl beider nicht ähnlich sein sollte. Demnach ist bereits ein annähernd ausgeglichenes Verhältnis bereits ein Zeichen für Hochdimensionalität. Schlussfolgernd lässt sich festhalten, dass dieses Verhältnis einen Einfluss auf die Verfahrensauswahl und die Ergebnisqualität haben kann und berücksichtigt werden sollte.

In Kapitel 3.1 wurde die Datenvorverarbeitung als Herausforderung im Data-Mining-Prozess thematisiert. Der Grund dafür sind die betrachteten Anforderungen einiger Verfahren in Kapitel 2.4. Ersichtlich wurde, dass nicht jedes Verfahren zwingend dieselbe Vorverarbeitung benötigt. Beispielweise kann der k-Nearest-Neighbour mit fehlenden Werten in der Datenmenge umgehen, ist dagegen aber auf eine Standardisierung der Daten angewiesen, da es sich um ein distanzbasiertes Verfahren handelt. Fehlende Werte können allerdings nicht grundsätzlich für jedes Verfahren ignoriert werden.

Für die Verfahrensauswahl kann es demnach sinnvoll sein, Anforderungen dieser Art zu berücksichtigen, um eine effiziente Vorverarbeitung planen zu können und gegebenenfalls nicht notwendige Vorverarbeitungsschritte zu vernachlässigen. Besonders für eine sehr große Datenmenge kann dies zu einer Zeitersparnis in der Vorbereitung führen.

Bei der Betrachtung der Verfahren in 2.4 haben sich folgende Vorverarbeitungsschritte herauskristallisiert, die in dem Entscheidungsmodell berücksichtigt werden sollten:

Die *Merkmalsauswahl* (Kapitel 2.3.2) verfolgt einige wichtige Ziele. Zum einen führt sie zu einer Datenreduktion. Dies kann sowohl Vorteile für die Übersichtlichkeit und Interpretation der Ergebnisse als auch für die Leistung der Analyse haben. Zudem besteht durch die Eliminierung irrelevanter Attribute die Möglichkeit, einer Hochdimensionalität entgegenzuwirken.

Zum anderen ist es für viele Verfahren notwendig, unkorrelierte, also unabhängige, Attribute für die Analyse zu nutzen. Daher sollten Attribute eliminiert werden, die mit einem weiteren Attribut korrelieren und somit keine neue Information für die Analyse liefern. Diese Eigenschaften haben bspw. Verfahren wie der Naive-Bayes-Algorithmus, k-Nearest-Neighbour-Algorithmus und Entscheidungsbaumalgorithmen wie der ID3 (Kapitel 2.4).

Zudem hängt die Geschwindigkeit von Algorithmen aus technischer Sicht mit der Anzahl an Attributen zusammen. Das Entfernen von Ausreißern und Rauschen (Kapitel 2.3.1) ist vor allem für distanzbasierte Verfahren von großer Bedeutung, wie in Kapitel 2.4 festgestellt wurde. Für andere Verfahren ist das Entfernen von Ausreißern und Rauschen ein optionaler Schritt.

Fehlende Werte (Kapitel 2.3.1) sind nicht für jedes Verfahren ein Problem. Die Behandlung fehlender Werte kann allerdings mit einem großen Aufwand verbunden sein. Möchte man ein Verfahren nutzen, das mit fehlenden Werten umgehen kann, so kann dieser Vorverarbeitungsschritt umgangen und Aufwand gespart werden.

Standardisierung(Kapitel 2.3.1) ist in Kapitel 2.4 insbesondere im Zusammenhang mit Verfahren erwähnt worden, die Distanzen berechnen. Durch Standardisierung wird verhindert, dass Werte, die aufgrund ihres Attributes sehr hoch sind, so stark dominieren, dass kleinere Werte nicht beachtet werden.

In diesem Kapitel wurden Eigenschaften untersucht, die einen möglichen Einfluss auf die Verfahrensauswahl haben. Betrachtet wurden dabei Eigenschaften der Datengrundlage sowie einiger Verfahren hinsichtlich der benötigten Vorverarbeitung. Der Kategorisierungsansatz kann somit zwei Ziele verfolgen: Die Auswahl eines Verfahrens auf Grundlage der vorliegenden Daten und eine effiziente Vorverarbeitung, indem lediglich die Schritte verfolgt werden, die laut Verfahrenseigenschaften notwendig sind. Demnach ergeben sich die folgenden Eigenschaften des neuen Kategorisierungsansatzes:

- Datentyp: Inputvariablen
- Datentyp: Outputvariable
- Datenmenge (Anzahl Datensätze, Anzahl Attribute, Verhältnis)
- Umgang mit fehlenden Werten
- Umgang mit Ausreißern und Rauschen
- Merkmalsauswahl (Attributabhängigkeit)
- Standardisierung

Anhand dieser Eigenschaften sollte es möglich sein, vorhandene Verfahren in Kategorien einzuteilen, um eine schnelle Entscheidung zu unterstützen.

3.2.2 Abhängigkeiten

Um im Rahmen eines Entscheidungsbaumes einen Pfad zu einem geeigneten Verfahren für die Datengrundlage zu erstellen, müssen mögliche Abhängigkeiten berücksichtigt werden, die sich im Allgemeinen im Hinblick auf Verfahrenseigenschaften ergeben.

Nun werden beispielhaft einige Verfahren, die bereits in Kapitel 2.4 beschrieben wurden, auf diese Eigenschaften untersucht, um die Realisierbarkeit dieser Kategorisierungsidee festzustellen.

Betrachten wir zunächst die aus der Literatur bekannten Eigenschaften des Naive Bayes. Dieser gilt als geeignet für kategorische Inputvariablen. Liegen numerische Variablen vor, müssen diese zunächst diskretisiert werden. Ebenfalls bekannt ist, dass die Outputvariable ebenfalls kategorisch sein muss. Anhand dieses Beispiels zeigt sich bereits, dass Einschränkungen hinsichtlich des Datentyps existieren. Weiterhin können fehlende Werte für den Einsatz dieses Verfahrens ignoriert werden. Ebenso gilt das Verfahren als robust gegenüber Ausreißern. Demnach können für eine effiziente Vorverarbeitung diese beiden

Eigenschaften	Naive Bayes	k-Nearest-Neighbour
Datentyp: Inputvariable	Kategorisch	Numerisch + kategorisch
Datentyp: Outputvariable	Kategorisch	Kategorisch
Datenmenge	Nicht bekannt	Nicht zu groß
Behandlung fehlender Werten	Nicht notwendig	Nicht notwendig
Behandlung von Ausreißern und Rauschen	Nicht notwendig	Notwendig
Merkmalsauswahl	Notwendig	Notwendig
Standardisierung	Nicht notwendig	Notwendig

Abbildung 4: Eigenschaften der betrachteten Verfahren im Hinblick auf den Kategorisierungsansatz

Schritte zunächst ausgeschlossen werden und dennoch möglicherweise gute Ergebnisse erzielt werden. Wichtig für den Einsatz des Naive Bayes ist allerdings die Merkmalsauswahl, da die genutzten Attribute unabhängig voneinander sein sollten.

Als weiteres Verfahren wird der k-Nearest-Neighbour (Kapitel 2.4.3) betrachtet. Das Verfahren eignet sich sowohl für numerische als auch für kategorische Inputvariablen. Die Outputvariable muss dagegen kategorisch sein. Für dieses Verfahren müssen Ausreißer entfernt sowie eine Standardisierung durchgeführt werden, da es sich um ein distanzbasiertes Verfahren handelt. Fehlende Werte gelten dagegen nicht als Einschränkung. Bei einer großen Datenmenge gilt das Verfahren außerdem als sehr rechenintensiv. Zudem gilt bei einer großen Datenmenge, dass die Unterschiede zwischen den Werten geringer ist und somit der nächst Nachbar ungenauer bestimmt wird. Für dieses Verfahren sind relevante Attribute notwendig.

Für andere distanzbasierte Verfahren, wie k-Means und k-Medoid (Kapitel 2.4.4) gelten ebenfalls die Voraussetzungen, dass eine Ausreißerbehandlung und eine Standardisierung der Daten durchgeführt werden muss. Dadurch kann die Vermutung festgehalten werden, dass für Verfahren, die aufgrund ihrer Berechnungen durch Ausreißer beeinträchtigt werden, ebenfalls eine Standardisierung der Daten notwendig ist.

Es ergeben sich demnach folgende Eigenschaften für die beiden betrachteten Verfahren (Abb. 4).

Die Untersuchung beispielhafter Verfahren und deren Abhängigkeiten in Bezug auf Datentypen und Vorverarbeitungsschritte sollte es ermöglichen, Verfahrenskategorien mit konkreten Eigenschaften zu erstellen. Die Betrachtung beider Verfahren im vorherigen Abschnitt hat bereits eine Wissenslücke hervorgebracht: Die Datenmenge. Das Beispiel des Naive Bayes beantwortet nicht die Frage, welche Datenmenge für dieses Verfahren geeignet ist. Für den k-Nearest-Neighbour wird in der Literatur geraten, keine großen Datensätze zu verwenden, aufgrund einer hohen Rechenintensität und ungenauer Ergebnisse. Jedoch konnte im Rahmen der Literaturrecherche kein Hinweis darauf gefunden werden, ab welcher Größe eine Datenmenge als groß bezeichnet werden kann. Es kann davon ausgegangen werden, dass von einer großen Datenmenge gesprochen wird, wenn die Anzahl der Datensätze sehr groß ist. Besteht eine Datenmenge aus einer großen Anzahl an Attributen, wird vielmehr von einer großen Dimension oder hohen Komplexität gesprochen. Dennoch gibt

es kaum Informationen oder Angaben einer spezifischen Anzahl.

Durch das Fehlen dieser Information ist es mit dem Hintergrund der durchgeführten Recherche nicht möglich, Kategorien auf der Grundlage der Daten zuverlässig zu erstellen. Allerdings sind für die meisten Verfahren die einsetzbaren Datentypen der Input- und Outputvariablen bekannt.

Es wäre beispielsweise möglich, eine Verfahrenskategorie zu entdecken, die für Datenmengen mit geringer Anzahl Datensätze geeignet ist. Oder eine Gruppe von Verfahren, die besonders robust sind und wenig Vorverarbeitung benötigen und dadurch möglicherweise kurzfristig für erste Erkenntnisse genutzt werden können. Ebenso hilfreich ist eine Verfahrensgruppe, die mit besonders großen Datenmengen gute Ergebnisse liefert.

Wie in Kapitel 3.2 erarbeitet, werden die Inhalte des Kategorisierungsansatzes in den Entscheidungsbaum integriert, die schlussendlich zu einem geeigneten Verfahren führen sollen. Der Zusammenschluss aller Pfade, die zu einem Verfahren führen, bilden die Verfahrenskategorie.

4 Konzeptionierung des Entscheidungsmodells

In diesem Kapitel wird erläutert, wie bei dem Aufbau des entscheidungsunterstützenden Modells in Form eines Entscheidungsbaumes vorgegangen wird. Die Ebenen des Baumes werden durch die in Kapitel 3.2 erarbeiteten Eigenschaften dargestellt. Zunächst werden in Kapitel 4.1 einige Anforderungen an das Modell gestellt. Zudem wird das Grundgerüst des Entscheidungsmodells vorgestellt. Kapitel 4.2 befasst sich mit einer genaueren Betrachtung der Knoten und Pfade, die sich auf die Datengrundlage beziehen. Die Darstellung der Datenvorverarbeitungsschritte wird in Kapitel 4.3 erläutert. Schlussendlich werden in Kapitel 4.4 die Blätter des Entscheidungsbaumes angestrebt, die die erwünschten Verfahrenskategorien enthalten. Kapitel 4.5 dient einer kritischen Betrachtung des aufgestellten Modells und der Herausforderungen, die mit der Konstruktion einhergehen.

4.1 Anforderungen an das Modell

Das Entscheidungsmodell wird in der Form eines Entscheidungsbaumes aufgebaut. Dies bietet sich einerseits aufgrund der Übersichtlichkeit an und andererseits aufgrund der hohen Anzahl an Entscheidungen, die innerhalb eines Data-Mining-Prozesses getroffen werden müssen. Insbesondere kann es in einem Data-Mining-Prozess vorkommen, dass verschiedene Möglichkeiten für den nächsten Handlungsschritt vorliegen, die in einem Entscheidungsbaum gut dargestellt werden können. Das Entscheidungsmodell soll dazu dienen, möglichst schnell einen geeigneten Pfad zu finden, um ein vorliegendes Data-Mining-Problem zu lösen.

Das Modell wird auf Grundlage der in Kapitel 3 erläuterten Herausforderungen und dem Kategorisierungsansatz erstellt.

Zu beachten ist allerdings, dass der Data-Mining-Prozess einen iterativen Ansatz verfolgt. Das bedeutet, dass die Wiederholung und Anpassung einzelner Schritte notwendig sein kann. Da ein Entscheidungsbaum keine Iterationen darstellt, muss der Anwender aufgrund der Ergebnisse eines Schrittes entscheiden, ob dieser wiederholt oder ein anderer Pfad gewählt werden muss.

Des Weiteren soll der Entscheidungsbaum kein allumfassender Wegweiser durch den Data-Mining-Prozess darstellen, sondern als Kategorisierungsunterstützung für Data-Mining-Verfahren dienen.

Das Grundgerüst des geplanten Entscheidungsmodells ist in Abb. 5 dargestellt und besteht aus fünf Ebenen. Wie die Ebenen im Entscheidungsmodell aufgebaut werden, wird in den einzelnen Kapiteln 4.2, 4.3 und 4.4 erläutert.

Die Ebenen des Modells haben sich aus dem Kategorisierungsansatz in Kapitel 3.2 heraus ergeben. Die Verfahrenskategorien, sollen zum Schluss als Blätter des Entscheidungsbaumes dargestellt werden, auf die keine weiteren Knoten bzw. Entscheidungen folgen. Die darauffolgende Ebene enthält schließlich die Verfahrenskategorie.

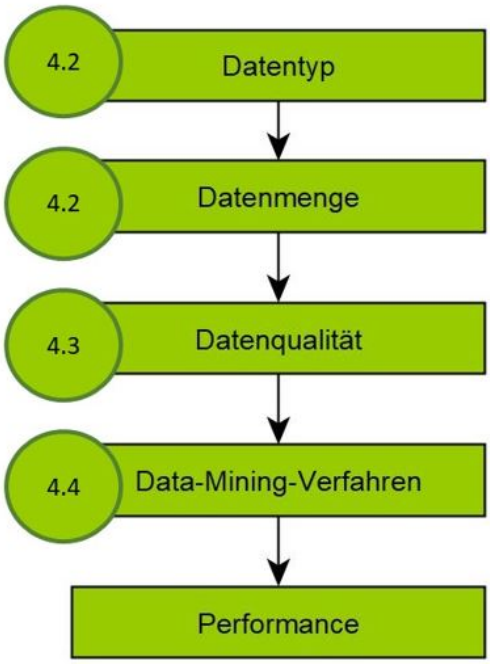

Abbildung 5: Grundgerüst des Kategorisierungsmodells

4.2 Vorbereitung der Datengrundlage

Die Wurzel des Entscheidungsbaumes wird durch die Abfrage des vorliegenden Datentyps der Inputvariablen dargestellt. Zunächst werden dabei der numerische und der kategorische Datentyp berücksichtigt. Das Entscheidungsmodell soll zunächst übersichtlich bleiben, weshalb die Beschränkung auf diese beiden gängigen Datentypen stattfindet.

Es stellt sich die Frage, wie die Datenmenge abgefragt werden sollte. Zur Auswahl steht die Anzahl der Attribute und die Anzahl der Datensätze. Wenn man die Bedeutung der Merkmalsauswahl berücksichtigt, bzw. die Notwendigkeit, für eine qualitative Analyse relevante Attribute zu verwenden, scheint der Einfluss auf die Attributanzahl sehr gering zu sein. Es ist nicht erstrebenswert, mehr Attribute als notwendig zu verwenden oder die Anzahl von Attributen künstlich zu vergrößern. Auf der anderen Seite kann es zu einer Einschränkung der Ergebnisqualität führen, wenn relevante Attribute nicht in die Analyse einbezogen werden, um die Dimension der Datenmenge zu verringern. Schlussendlich scheint es sinnvoll, relevante Attribute mit geeigneten Verfahren zu bestimmen und für diese Menge ein geeignetes Data-Mining-Verfahren zu verwenden.

Auf der anderen Seite kann die Anzahl der Datensätze im Besonderen Einfluss auf die Performance von Data-Mining-Verfahren haben. Obwohl grundsätzlich eine große Anzahl von Datensätzen gewünscht ist, kann der Fall eintreten, dass nur wenig Datensätze vorhanden sind und keine Möglichkeit besteht, mehr Datensätze zu generieren. Für diesen Fall wäre es hilfreich, geeignete Verfahren für wenig Datensätze vorzuschlagen.

Nun muss die Frage beantwortet werden, welche der beiden Abfragen für die Verfahrensauswahl sinnvoller ist. Da sich die Frage nicht pauschal beantworten lässt, wird für dieses Entscheidungsmodell zunächst davon ausgegangen, dass eine besonders kleine An-

zahl an Datensätzen mit einer größeren Herausforderung für den Anwender einhergeht. Die Attributanzahl gilt als Herausforderung, wenn man von einer hochdimensionalen Datenmenge spricht. Wie bereits in Kapitel 3.2.1 erörtert, gilt es, Hochdimensionalität zu vermeiden. Dies ist im Besonderen aus dem Grund wichtig, dass viele Data-Mining-Verfahren Probleme damit haben, wenn mehr Attribute als Datensätze vorhanden sind und dadurch falsche Ergebnisse liefern. Demnach ist es sinnvoll, das Verhältnis beider Mengen in das Entscheidungsmodell zu integrieren und dadurch die Anzahl an Attributen in der Datenmenge zu berücksichtigen.

In Abb. 6 wird der Wurzelknoten dargestellt, sowie die ersten Kanten, an denen auf der nächsten Ebene die Datenmenge abgefragt wird. Zusammenfassend ergeben sich für

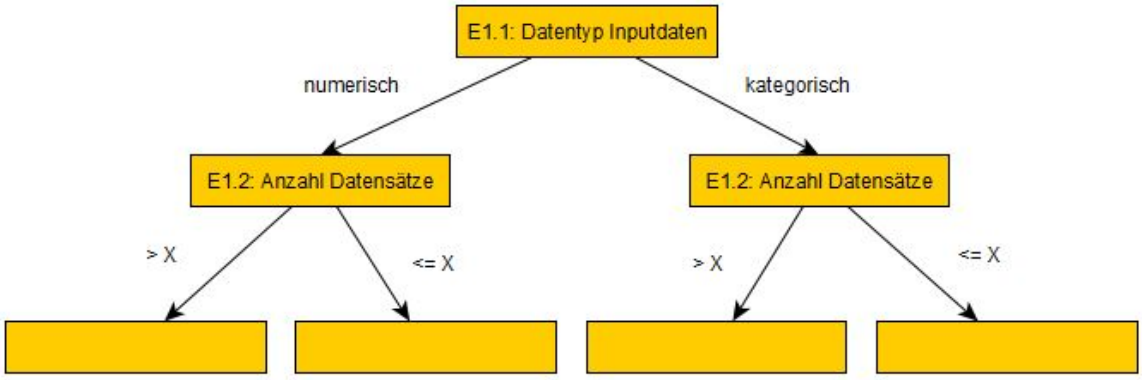

Abbildung 6: Kategorisierungsmodell: Datentyp und Anzahl der Datensätze

die ersten beiden Ebenen die folgenden Knotenentscheidungen:

Knoten E1.1 (Wurzel): Welcher Datentyp liegt für die Inputvariablen vor?
Die Entscheidung wird dabei zunächst zwischen numerisch und kategorisch getroffen.

Knoten E1.2: Wie viele Datensätze liegen vor?
Auf der Grundlage der durchgeführten Recherche lässt sich derzeit keine zuverlässige und wissenschaftliche Annahme treffen, wie die Intervalle einzuteilen sind. Die Anzahl der Datensätze, zwischen denen unterschieden werden muss, wird zunächst auf X gesetzt. Die Entscheidung wird dabei zwischen $<$ X und $>$ X getroffen.

Knoten E1.3: Liegt Hochdimensionalität vor?
Für die Entscheidung, ob Hochdimensionalität vorliegt, fehlen ebenfalls die notwendigen Informationen. Ausgehend von der Tatsache, dass deutlich mehr Datensätze als Attribute vorliegen sollten, wird ein Verhältnis von 1:1 (es gibt genauso viele Attribute wie es Datensätze gibt) ausgeschlossen. Ebenso gilt, dass eine annähernd gleiche Anzahl beider Mengen bereits als hochdimensional bezeichnet werden kann. Für das Entscheidungsmodell wird angenommen, dass eine Datenmenge als gut anwendbar gilt, wenn ein Verhältnis von 1:10 (1 Attribut auf 10 Datensätze) vorliegt.

In Abb. 7 sind nun die ersten drei Ebenen dargestellt, in der die Datengrundlage abgefragt wird. Wird der erste Pfad beispielhaft durchlaufen, haben wir eine Datengrundlage, deren Inputdaten numerisch sind, deren Anzahl an Datensätzen größer als X ist und als hochdimensional gilt, da das Verhältnis von Attribut zu Datensätzen größer als das festgelegte Minimum von 1:10 ist. Somit liegen weniger als 10 Datensätze pro Attribut vor.

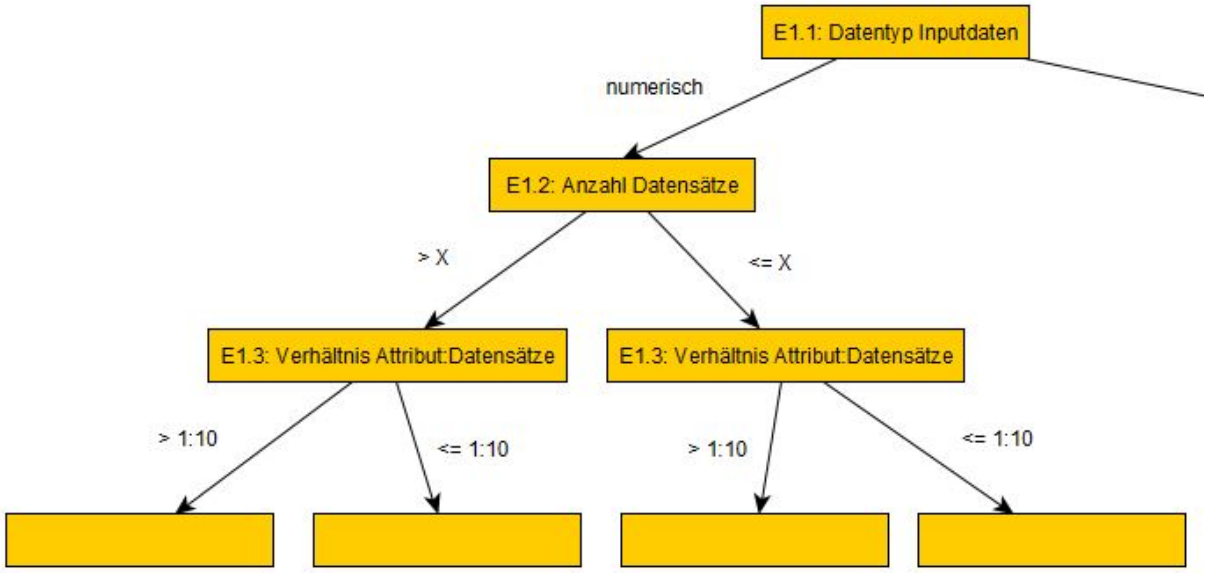

Abbildung 7: Kategorisierungsmodell: Darstellung der Knoten zur Visualisierung der vorliegenden Datenmenge: Datentyp, Anzahl Datensätze, Hochdimensionalität

Die Pfade werden im nächsten Abschnitt um die in Kapitel 3.2 festgelegten Vorverarbeitungsschritte ergänzt. Fest steht allerdings bereits, dass die Verfahrensgruppen, die zum Schluss für den linken Pfad entstehen, für numerische Inputdaten, eine hohe Anzahl an Datensätzen und gleichzeitig für eine hochdimensionale Datenmenge geeignet sein muss.

4.3 Verfahren der Datenvorverarbeitung in Abhängigkeit der Datengrundlage

Dieses Kapitel befasst sich damit, geeignete Vorverarbeitungsschritte in den Entscheidungsbaum zu integrieren. Der Versuch, einzelne Schritte einzubetten, hat zu einem unübersichtlichen Modell geführt. Aus diesem Grund wird nun der Versuch unternommen, Pakete an Vorverarbeitungsschritten zusammenzustellen, die durchgeführt werden können, bzw. müssen. Dabei werden möglichst sinnvolle Pakete geschnürt, woraufhin in der letzten Ebene die Verfahrenskategorie (Kapitel 4.4) folgt.

Für dieses Modell werden zunächst die Vorverarbeitungsschritte genutzt, die in Kapitel 3.2 im Kategorisierungsansatz berücksichtigt wurden.

Im Hinblick darauf, dass für den Anwender des Kategorisierungsmodells möglichst schnell ein Verfahren vorgeschlagen werden soll, gilt es zunächst zu überlegen: Welche Art von Verfahrenskategorien könnte für den Anwender nützlich sein?

Geht man von der Annahme aus, dass der Anwender anhand der vorliegenden Daten lediglich schnell eine erste Einschätzung erwartet, ohne vorherigen Aufwand, so bietet sich ein Vorverarbeitungspaket an, das keinen Vorverarbeitungsschritt beinhaltet. Wie in Kapitel 3.2 erläutert, sind durchaus Verfahren vorhanden, die mit Qualitätsproblemen (Fehlende Werte und Ausreißer) umgehen können und gute Ergebnisse liefern können. Dies soll für jede Datenmenge und jeden Datentyp gelten. Als Ausnahme kann allerdings die Merkmalsauswahl betrachtet werden. Möchte der Anwender tatsächlich eine erste Einschätzung der Daten erhalten, ohne zuvor viel Arbeit in die Vorverarbeitung zu investieren, bietet sich zumindest die Auswahl relevanter Daten an. Wie bereits in Kapitel 2.3.2 ausführlich beschrieben, kann eine hohe Dimensionalität der Datenmenge zu einer hohen Modellkomplexität führen. Zudem wird eine Interpretation der Ergebnisse mit zunehmender Dimension schwieriger. Daneben können Ergebnisse durch die Einbeziehung irrelevanter Attribute verfälscht werden. Diese Argumentation führt dazu, dass die Merkmalsauswahl in jedem Fall durchgeführt werden sollte. Im Hinblick auf die Tatsache, dass durch die Merkmalsauswahl gleichzeitig in den meisten Fällen eine Datenreduktion erfolgt, bietet sich der Schritt zu Beginn der Vorverarbeitung an. So können nachfolgende Schritte auf einer kleineren Datenmenge effizienter durchgeführt werden.

Abb. 8 zeigt einen beispielhaften Pfad, der lediglich die Merkmalsauswahl beinhaltet. Nach der Merkmalsauswahl, verändert sich unter Umständen das Verhältnis der Attributmenge und der Datensätze. Eine Datenmenge, die zuvor als hochdimensional eingestuft wurde, kann nach diesem Schritt ein besseres Verhältnis aufweisen. Aus diesem Grund wird zusätzlich nach der Merkmalsauswahl für den Pfad $>$ 1:10 erneut auf Hochdimensionalität geprüft. Wurde ein günstigeres Verhältnis erreicht, führt dieser Pfad mit dem ersten Pfad $<=$ 1:10 zusammen (siehe Abbildung 8).

Das in Abb. 8 dargestellte Modellstück soll als Grundgerüst für die nächsten Pfade genutzt werden. Wie bereits erwähnt, werden nun Pakete von Vorverarbeitungsschritten zusammengestellt.

Ein Pfad für jede Datengrundlage soll ohne weitere Vorverarbeitung durchlaufen werden. Dieser Pfad erhält zunächst die Bezeichnung A.

Ein weiterer Pfad soll alle drei weiteren Vorverarbeitungsschritte beinhalten. Demnach

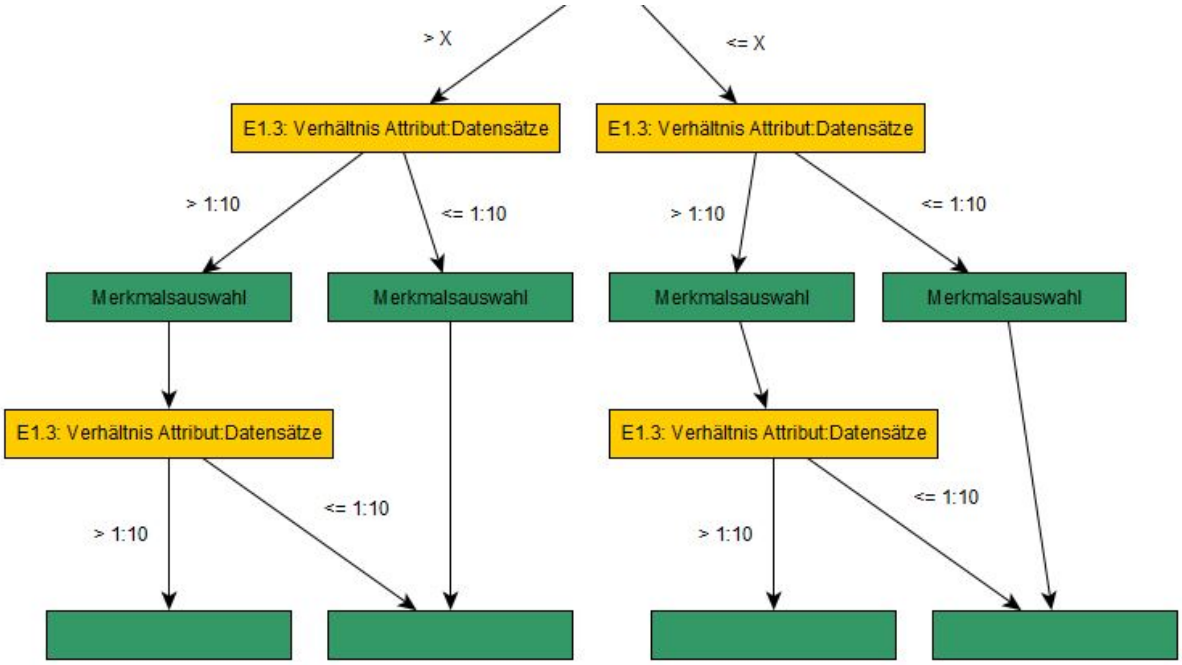

Abbildung 8: Grundgerüst des Entscheidungsmodells nach der Merkmalsauswahl

die Behandlung fehlender Werte, Ausreißer und Rauschen sowie der Standardisierung aller Daten. Dieser Pfad ist deutlich aufwendiger, führt jedoch höchstwahrscheinlich zu besseren Ergebnissen. Zudem sind diese Schritte für viele Verfahren zwingend notwendig. Dieser Pfad enthält die Bezeichnung B.

Ein weiteres Paket wird durch die Behandlung von Ausreißern und Rauschen sowie Standardisierung gebildet. Die Begründung liegt darin, dass sich diese Schritte gegenseitig einschließen. Die meisten Verfahren, die große Probleme mit Ausreißern haben, weil sie beispielsweise Distanzen berechnen, benötigen aus demselben Grund standardisierte Daten, da nicht standardisierte Daten dieselben rechnerischen Probleme auslösen. Dieser Pfad wird mit C bezeichnet.

Das vorerst letzte Paket beinhaltet schließlich lediglich die Behandlung fehlender Werte. Dabei kann die Art der Behandlung von der vorliegenden Datenmenge abhängig gemacht werden. Für eine große Datenmenge bietet sich das Löschen der betroffenen Datensätze an. Wie in Kapitel 2.3.1 beschrieben, empfehlen Experten, das Löschen, wenn ausreichend weitere Datensätze für die Analyse verbleiben. Der Hintergrund ist, dass eine Imputation von Durchschnittswerten oder Konstanten etc. immer mit einer Veränderung des Datensatzes einhergehen. Demnach kann der Vorschlag gemacht werden, Datensätze mit fehlenden Werten zu löschen, wenn der Pfad > X im Entscheidungsknoten E1.2 gewählt wird. Dieser Pfad wird mit D bezeichnet.

In Abb. 9 werden die vier Pfade beispielhaft dargestellt. Aus Gründen der Übersichtlichkeit wird jeder Pfad lediglich einmal modelliert. Jedoch gelten alle Pfade für die in Abb. 9 dargestellten Ausgangsknoten. Diese vier Pakete sollten demnach für jede Kombination aus Datentyp, Datenmenge und Datenverhältnis vorgeschlagen werden.

Auf die in Abb. 9 dargestellten Vorverarbeitungspakete folgen schließlich die Blät-

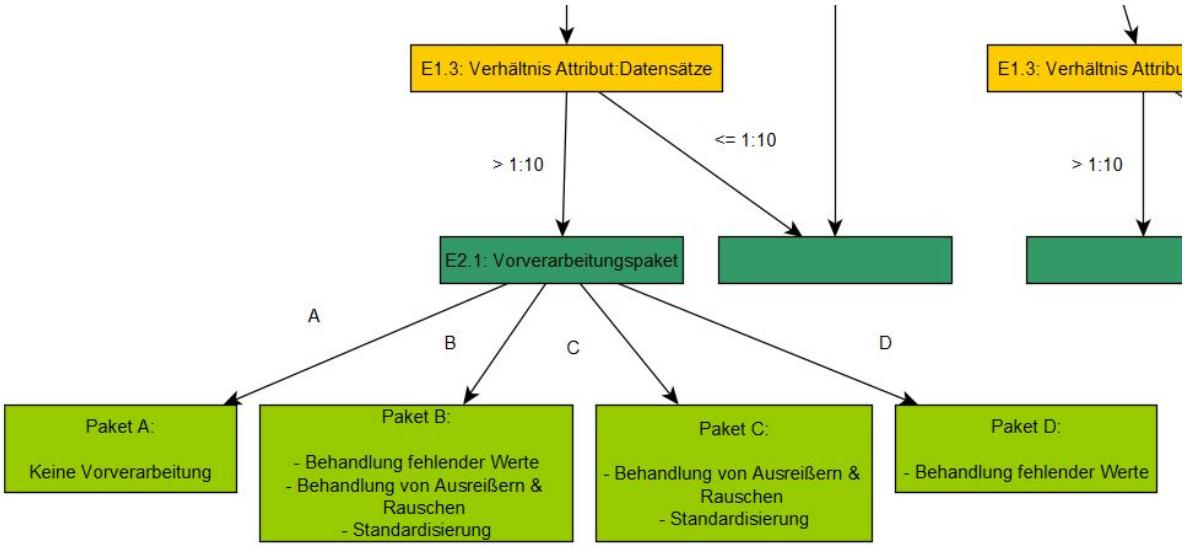

Abbildung 9: Entscheidungsmodell nach der Einbindung der Vorverarbeitungspakete

ter des Entscheidungsbaumes, durch die unterschiedliche Verfahrenskategorien gebildet werden. Diese Kategorien beinhalten schlussendlich alle Eigenschaften enthalten, die im Kategorisierungsansatz (Kapitel 3.2.1) festgelegt wurden.

Lediglich eine Eigenschaft muss noch abgefragt werden: Der Datentyp der Outpuvariable (sofern eine vorhanden ist). Wie bei dem Datentyp der Inputvariable zu Beginn, wird zunächst zwischen numerischen und kategorischen Datentypen unterschieden. Demnach folgt auf jeden Paket-Knoten die Abfrage, ob eine Outputvariable existiert oder nicht. Weiterhin führt der Pfad, der eine Outputvariable enthält zu der Entscheidung, ob diese numerisch oder kategorisch vorliegt. Dadurch ergeben sich für jedes Paket schlussendlich drei mögliche Verfahrenskategorien (siehe Abb. 10).

Die drei blauen Kästchen stellen die Blätter dar, die schlussendlich empfohlene Data-Mining-Verfahren für einen gesamten Pfad enthalten sollen. Im nächsten Kapitel werden die Inhalte der Blätter näher diskutiert.

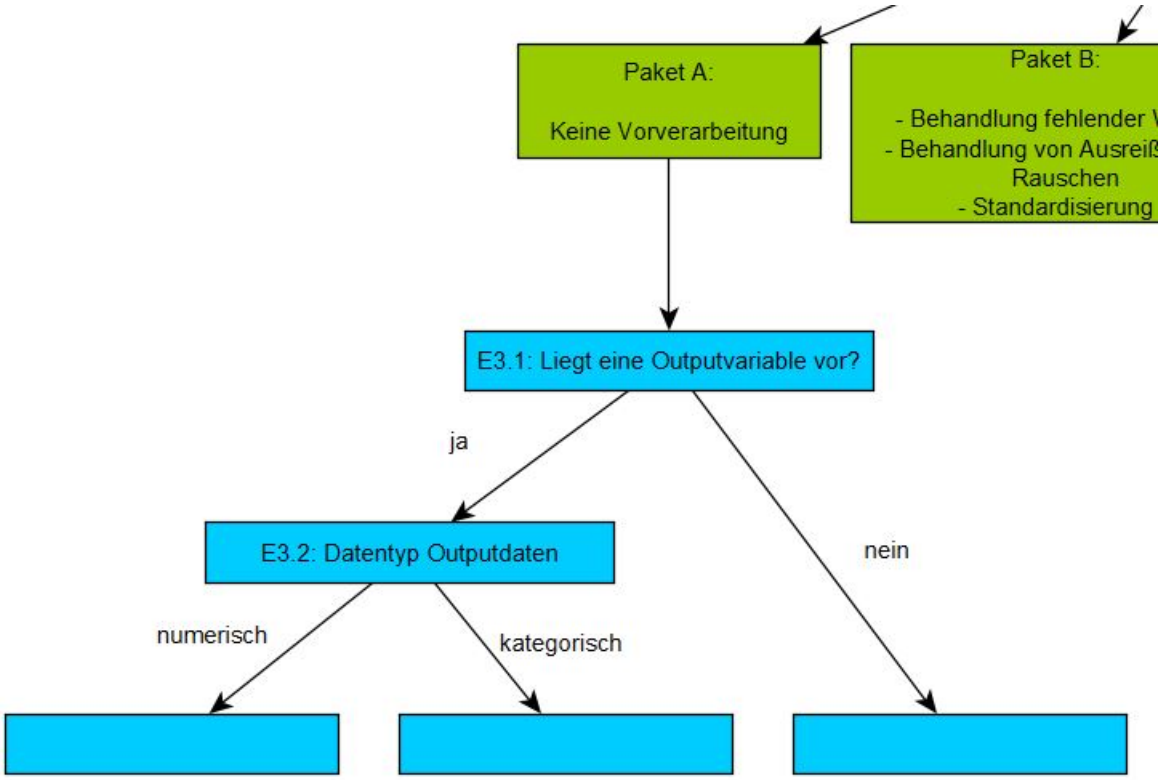

Abbildung 10: Entscheidungsmodell nach der Abfrage der Outputdaten und Erreichen der Blätter

4.4 Verfahren des Data Minings in Abhängigkeit der Datenvorverarbeitung

In den Blättern des Entscheidungsbaumes sollen Data-Mining-Verfahren vorgeschlagen werden. Diese sind dadurch automatisch einer Kategorie zugewiesen, dessen Eigenschaften durch die Pfade festgelegt sind.

In den letzten Kapiteln wurden die Pfade des Entscheidungsmodell derart erstellt, dass alle Eigenschaften des Kategorisierungsansatzes (Kapitel 3.2.1) berücksichtigt wurden und zu dem Blatt des Entscheidungsbaumes führen. In den Blättern sollen Data-Mining-Verfahren eingefügt werden, die dadurch automatisch einer Kategorisierung unterzogen werden.

Wie man an dem gesamten Modell erkennen kann, ist die Anzahl an Blättern sehr groß. Es ist nicht auszuschließen, dass einige Blätter keine Verfahren enthalten. Aus Gründen der Übersichtlichkeit wird das Entscheidungsmodell im Anhang in zwei Teile getrennt und lediglich die Seite für numerische Inputdaten dargestellt. Die Seite der kategorischen Inputdaten kann genau gleich aufgebaut werden.

Die Anzahl der Pfade und somit mögliche Kategorien sind zu groß, um in diesem Schritt zuverlässig Data-Mining-Verfahren zu empfehlen. Allerdings ist durch das Modell ein Grundgerüst entstanden, durch das es anhand einer visuellen Unterstützung möglich sein kann, diese Kategorien zu erstellen.

Besonders interessant könnten dabei Verfahrenskategorien sein, die folgende Eigen-

schaften erfüllen:

- Kleine Datenmenge und ein ungünstiges Verhältnis zu der Attributanzahl, die mit möglichst wenig Vorverarbeitungsaufwand analysiert werden soll
- Große Datenmenge und ein ungünstiges Verhältnis zu der Attributanzahl, die mit möglichst wenig Vorverarbeitungsaufwand analysiert werden soll

Die Verfahren, die diese Eigenschaften innehaben, können als robust angesehen werden und somit für eine kurzfristige erste Analyse geeignet sein. Unter Betrachtung der beschriebenen Verfahren in Kapitel 2.4 wird häufig angesprochen, dass distanzbasierte Verfahren nicht gut mit Ausreißern und ohne standardisierte Daten umgehen können. Verfahren, auf die diese Eigenschaften zutreffen, können je nach möglicher Datengrundlage in die Verfahrenskategorien eingeteilt werden, die auf das Vorverarbeitungspaket C folgen. Beispielsweise könnte zu dieser Kategorie der k-Nearest-Neighbour gehören.

Verfahren, die nachweislich mit fehlenden Werten und Ausreißern umgehen können, gehören dagegen jenen Verfahrenskategorien an, die auf das Vorverarbeitungspaket A folgen. Dazu kann beispielsweise der Naive Bayes gehören. Im Verlauf dieser Arbeit ist es allerdings nicht möglich, begründete Kategorien mit Verfahren zu bestücken, da einige grundlegende Informationen bezüglich der Datengrundlage fehlen. Des Weiteren fehlen zuverlässigere Angaben hinsichtlich der Datenvorverarbeitung.

Im nächsten Kapitel werden die Herausforderungen, die sich bei der Zuordnung von Data-Mining-Verfahren in das Entscheidungsmodell ergeben, diskutiert und erläutert.

4.5 Kritische Betrachtung des Entscheidungsmodells

Die Anforderung, dass das Modell zu einer möglichst schnellen Entscheidung beitragen soll, konnte noch nicht erfüllt werden. Deutlich geworden ist die große Bedeutung praktischer Erfahrung im Umgang mit Data-Mining-Verfahren und vor allem der Datenvorverarbeitung. In der Theorie werden zwar einige gute Hinweise dazu gegeben, wie eine Vorverarbeitung durchgeführt werden kann und welches Verfahren bestimmte Vorverarbeitungen benötigen, allerdings kann ohne eine tiefergehende Forschung und umfangreiche Evaluierungen keine zuverlässige Entscheidungsempfehlung in dieser Arbeit gewährleistet werden.

Erste Herausforderungen bei der Konstruktion des Modells wurden bei der Entscheidungssuche eines geeigneten Schwellenwertes für die Datenmenge ersichtlich. In Kapitel 2 konnten bei der Literaturrecherche keine Angaben dazu gefunden werden, ab welcher Anzahl von Datensätzen oder Attributen eine Datenmenge als groß gilt. Möchte man allerdings bei einer Entscheidung von der Datengrundlage, und somit auch von der Datenmenge, ausgehen, sind praktische Versuche notwendig, um zu erkennen, bei welcher Datenmenge ausgewählte Verfahren gute Ergebnisse liefern. Abhängig von der Datenmenge sind die Vorverarbeitungsschritte. Fragen, die man sich stellen muss, sind unter anderem: Ab welcher Anzahl von Datensätzen kann oder sollte ich eine Datenreduktion durchführen? Ab welcher Anzahl von Datensätzen lohnt es sich für die Analyse nicht mehr, Datensätze mit fehlenden Werten aus der Datenmenge zu löschen? Was ist eine geeignete Anzahl von Attributen, je nach Verfahren?

Die vorliegende Datenmenge beeinflusst die Strategie der Datenvorverarbeitung, sowohl die Anzahl der Datensätze als auch die Anzahl der Attribute.

In dem erstellten Entscheidungsmodell ist durch die Betrachtung des Verhältnisses von Attributen zu Datensätzen ein erster Schritt eingebaut, um der Gefahr einer Hochdimensionalität entgegenzuwirken, da die meisten betrachteten Verfahren zum Lernen deutlich mehr Datensätze als Attribute voraussetzen. Besonders für distanzbasierte Verfahren ist dies von Bedeutung. Eine Menge von Attributen oder Datensätzen anzugeben, die unter- bzw. überschritten werden sollte, ist in diesem Fall nicht sinnvoll. Für eine wissenschaftliche Einschätzung müssten in einer praxisorientierten Untersuchung unterschiedliche Verfahren auf dieselbe Datenmenge angewandt werden. Aufgrund der hohen Anzahl an verfügbaren Verfahren und Weiterentwicklungen könnte man dabei den Fokus zunächst auf die Basis-Verfahren legen. Die Weiterentwicklungen der Basis-Verfahren können dann als Alternativen eingesetzt werden.

Hinsichtlich einer großen Anzahl von Attributen gilt, dass abstandsbasierte Verfahren mit zunehmender Dimension an Wirkung verlieren. Demnach sind Verfahren, die beispielsweise für Clustering oder Klassifikation eingesetzt werden, auf Datenmengen ineffektiv, die eine hohe Dimension, also eine große Anzahl von Attributen, aufweisen. Jedoch bleibt die Frage unbeantwortet, welche Anzahl als groß definiert werden kann.

Bekannt ist, dass im Allgemeinen die Berücksichtigung relevanter Attribute für die Datenanalyse erforderlich ist. Zusätzlich unterstützt das Vorhandensein unkorrelierter Attribute eine effektive Analyse. Aus diesem Grund wird der Schritt der Merkmalsauswahl als erforderlicher Schritt in dem Modell aufgeführt. Zu welchem Zweck die Merkmalsaus-

wahl stattfindet, muss für die vorliegenden Daten spezifisch entschieden werden. Aus der Recherche heraus ergibt sich, dass eine Merkmalsauswahl zu Beginn durchgeführt werden sollte. Zum einen aufgrund der Notwendigkeit. Relevante Attribute zu nutzen ist für die meisten Verfahren von großer Bedeutung. Im Bereich der Assoziationsanalyse können durch irrelevante Attribute unsinnige Regeln entstehen. Im Bereich der Klassifikation und des Clusterings wurde bereits resümiert, dass hohe Dimensionen hinderlich sind. Zum anderen ist eine Datenmenge geringerer Komplexität hilfreich für eine effizientere weiterführende Vorverarbeitung. Beispielsweise bei der Behandlung von fehlenden Werten oder Ausreißern. Hier entsteht jedoch eine weitere Herausforderung: Auch Merkmalsauswahlverfahren haben mitunter Schwierigkeiten mit Daten, die fehlende Werte oder Ausreißer beinhalten. Demnach müssen diese Vorverarbeitungsschritte zuerst durchgeführt werden. Bei einer sehr großen Datenmenge kann dies zeitintensiv sein. Demnach gilt es, Merkmalsauswahlverfahren gezielt zu testen, inwieweit Ausreißer und fehlende Werte die Ergebnisse beeinflussen. Ist es möglich, irrelevante oder korrelierende Attribute in dem ersten Vorverarbeitungsschritt durchzuführen, weil das ausgewählte Verfahren trotz Ausreißern und fehlenden Werten gute Ergebnisse liefert, kann die darauffolgende Vorverarbeitung zeiteffizienter durchgeführt werden, als es bei einer umgekehrten Reihenfolge der Fall wäre.

Schlussendlich ist ein erster Schritt in die Richtung einer allumfassenden Kategorisierung von Data-Mining-Verfahren erfolgt, die möglichst viele Eigenschaften berücksichtigt. Aufgrund fehlender Informationen hinsichtlich der Datenmenge und fehlender technischer Tests, um geeignete Schwellenwerte wissenschaftlich zu begründen, kann in dieser Arbeit kein Data-Mining-Verfahren zuverlässig empfohlen werden, ohne willkürliche Annahmen zu treffen. Aus diesem Grund wurde darauf verzichtet. Allerdings sollte es mit umfangreichen Untersuchungen möglich sein, anhand des Kategorisierungsansatzes das Entscheidungsmodell schrittweise begründet zu füllen, da die Eigenschaften berücksichtigt wurden, die eine Verfahrensauswahl beeinflussen. Dadurch kann das Modell durchaus in der Zukunft als Entscheidungshilfe dienen.

5 Zusammenfassung und Ausblick

Das Ziel der Arbeit war es, mithilfe eines Entscheidungsbaumes eine neue Kategorisierung für Data-Mining-Verfahren anzustreben. Dafür war es notwendig, Einflüsse auf die Auswahl zu erörtern, die sich im gesamten Data-Mining-Prozess ergeben. Aus diesem Grund wurden in Kapitel 2.1 die einzelnen Schritte des Prozesses beleuchtet. Die Schritte wurden in den Kapiteln 2.2 und 2.3 näher beschrieben. Zunächst wurde dabei die Datengrundlage untersucht, die sich aus dem Datentyp, der Datenmenge, der Datenbeschriftung und der Datenqualität zusammensetzt. Daraufhin wurden Vorverarbeitungsschritte näher betrachtet, die vor der Datenanalyse durchgeführt werden müssen, um die Datenqualität zu erhöhen. In Kapitel 2.4 wurden schlussendlich für die Aufgabenfelder Klassifikation, Clustering und Assoziationsanalyse gängige Verfahren beschrieben und deren Eigenschaften beleuchtet. Dies geschah vor allem im Hinblick auf die Einsetzbarkeit für bestimmte Datentypen sowie notwendige Vorverarbeitungsschritte. Des weiteren wurden Vor- und Nachteile der Verfahren beschrieben, die die Auswahl eines Verfahrens beeinflussen können.

Auf Grundlage der durchgeführten Recherche wurden in Kapitel 3.1 einige Herausforderungen im Data-Mining-Prozess aufgegriffen, die in dem angestrebten Kategorisierungsansatz berücksichtigt werden sollen.

Im darauffolgenden Kapitel 3.2 wurden schlussendlich die wichtigsten Einflüsse und Voraussetzungen zu einem Kategorisierungsansatz verarbeitet. Berücksichtigt wurden dabei Einflüsse aus der Datengrundlage heraus, sowie Vorverarbeitungsschritte, von denen einige Verfahren abhängig sind. Diese Abhängigkeiten wurden anhand einiger beispielhafter Data-Mining-Verfahren erläutert. Durch das Zusammenfügen aller Eigenschaften des Kategorisierungsansatzes sollen sich die Kategorien ergeben, die geeignete Verfahren für die Vorbedingungen enthalten sollen.

Das Entscheidungsmodell, das als visuelle Unterstützung in Form eines Entscheidungsbaumes erstellt werden soll, wird in Kapitel 4 beschrieben. Kapitel 4.1 dient dabei zunächst zur Beschreibung von Anforderungen an das Modell. In den folgenden drei Kapiteln wird der Aufbau der einzelnen Ebenen beschrieben sowie die Intentionen und Überlegungen. In Kapitel 4.5 wird das Entscheidungsmodell kritisch betrachtet und diskutiert.

Es ist deutlich geworden, dass praktische Erfahrung im Bereich der Datenanalyse zwingend notwendig ist, um zuverlässige Empfehlungen hinsichtlich der Verfahrensauswahl aussprechen zu können.

Zusammenfassend lässt sich festhalten, dass ein neuer Kategorisierungsansatz auf Basis der Datengrundlage und der Vorverarbeitungsschritte durchaus Sinn machen kann, um eine effiziente Vorverarbeitungsphase zu ermöglichen. Zudem kann es förderlich sein, von vornherein zu wissen, ob ein Verfahren für die vorliegende Datenmenge geeignet ist. Es hat sich gezeigt, dass sich die Anforderungen stark zwischen Data-Mining-Verfahren unterscheiden und auf der anderen sehr ähnlich sein können. Aus diesem Grund ist die Betrachtung von möglichst vielen Kategorie-Ansätzen wünschenswert.

Bei der Erstellung des Entscheidungsmodells sind einige Herausforderungen deutlich geworden, die eine zuverlässige Empfehlung von Data-Mining-Verfahren in den Blättern des Entscheidungsbaumes erschweren. Es ist weder bekannt, wie groß die Anzahl von Datensätzen bzw. zu betrachtende Attribute für die in dieser Arbeit angesprochenen Ver-

fahren wünschenswert sind, noch wird ein geeignetes Verhältnis von Attribut zu Datensatz thematisiert. In der Literatur wird häufig allgemein von großen Datenmengen und hohen Dimensionen in Datenmengen gesprochen. Eine bestimmte Anzahl, die zu einer Orientierung beitragen könnte, wird dagegen nicht genannt. Eine willkürliche Vermutung ohne durchgeführte Validierungen sind in diesem Fall nicht wissenschaftlich begründbar. Daher wird vorgeschlagen, die Untersuchung fortzuführen, indem unterschiedliche Data-Mining-Verfahren auf dieselbe Datenmenge angewandt werden. Dadurch können die Ergebnisse vergleichbar gemacht werden. Die Datenmenge sowie das Verhältnis von der Anzahl der Datensätze zu der Attributanzahl sollten dabei variiert werden. Auf diese Weise kann eine fundierte Einschätzung hinsichtlich der Datenmenge erfolgen, die für ein Verfahren geeignet ist. Fest steht, dass die Datenmenge einen Einfluss auf die Eignung eines Verfahrens hat.

Aufgrund genannter Herausforderungen hat sich ergeben, dass das Ziel, eine Entscheidungsunterstützung zu liefern, nicht erreicht werden konnte. Stattdessen wurde ein Entscheidungsmodell konzipiert, in dem die Einflüsse der Verfahrensauswahl berücksichtigt wurden. Der Ansatz kann genutzt werden, um die Lücken des Modells mit validierten Ergebnissen zu füllen.

Im Hinblick auf eine Ergänzung des Entscheidungsmodells und somit einer Erweiterung des Kategorisierungsansatzes, können zudem weitere Eigenschaften von Verfahren untersucht werden, die die Perfomanceleistung betreffen. So sind einige Verfahren effizienter und für große Datenmengen besser geeignet als andere Verfahren. Verfahren werden stetig weiterentwickelt und es wird nach Verbesserungen gestrebt, die ein bestimmtes Problem beheben sollen. Daher ist es sinnvoll, in Zukunft einen genauen Blick auf mögliche Probleme einiger Verfahren zu werfen und alternative Verfahren vorzuschlagen. Demnach muss die derzeitige Kategorie nicht zwingend als Blatt gelten, sondern als ein Knoten, an den weitere Entscheidungen angeknüpft werden können.

Weitere Verfahrenseigenschaften, die einen großen Einfluss auf die Auswahl haben können, sind bspw. die notwendige Parametereinstellung und der Speicherbedarf.

Des Weiteren bietet es sich für zukünftige Untersuchungen an, weitere Datentypen zu berücksichtigen. Im Rahmen dieser Arbeit wurden zunächst kategorische und numerische Datentypen in die Kategorisierung aufgenommen. Um einen Schritt weiter zu gehen, können weitere Skalen sowie Zeitreihendaten und Textdaten einbezogen werden.

Die Bearbeitung des Themas hat einmal mehr die Tiefe des Data-Mining-Prozesses aufgezeigt. Ein hilfreiches Entscheidungsmodell für Data-Mining-Verfahren aufzustellen kann durchaus realistisch sein, allerdings ohne den Anspruch auf Vollständigkeit.

Basierend auf einer reinen Literaturrecherche ist dies allerdings nicht möglich, da sich das Problem als sehr komplex herausgestellt hat. Für zukünftige Arbeiten würde es sich anbieten, zunächst ein Verfahren nach dem anderen auf die anwendbare Datenmenge sowie unter varriierenden Paketen von Vorerarbeitungsschritten in praktischen Untersuchungen zu testen. Basierend auf diesen Ergebnissen können dadurch eventuell diese Verfahren in geeignete Pfade integriert und Verfahrenskategorien erstellt werden. Sind diese Verfahrenskategorien aufgestellt und wurden diese mit Data-Mining-Verfahren gefüllt, können jene untereinander auf Gemeinsamkeiten und eine innere Ordnung untersucht werden.

Das Ziel der Untersuchung müsste es zunächst sein, die nahezu unüberblickbare Palette an verfügbaren Verfahren im Hinblick auf den vorgestellten Kategorisierungsansatz in Verfahrensgruppen aufzuteilen, die dem Anwender in kurzer Zeit helfen soll, für die vorliegende Datengrundlage ein Verfahren vorzuschlagen. Erst wenn dies durch technische Tests erreicht werden konnte, ist die nähere Betrachtung einer inneren Ordnung innerhalb der Verfahrensgruppen sinnvoll.

In einem nächsten Schritt würde es sich anbieten, die Perfomance-Leistungen dieser Verfahren zu testen, um Fragen von Anwendern zu beantworten, wie lang die Analysedauer einzuschätzen oder wie groß der Speicherbedarf eines Verfahrens ist. Dies Bedarf allerdings voraussichtlich einer deutlich aufwendigeren Untersuchung.

Literaturverzeichnis

[1] Aggarwal, C. C. (2015). *Data Mining*. Springer International Publishing, Cham.

[2] Agrawal, R., Imieliński, T., und Swami, A. (1993). Mining association rules between sets of items in large databases. In Buneman, P. und Jajodia, S., Herausgeber, *Proceedings of the 1993 ACM SIGMOD international conference on Management of data - SIGMOD '93*, Seiten 207–216, New York, New York, USA. ACM Press.

[3] Bankhofer, U. und Vogel, J. (2008). *Datenanalyse und Statistik: Eine Einführung für Ökonomen im Bachelor ; [Bachelor geeignet*. Lehrbuch. Gabler, Wiesbaden, 1. aufl.. Auflage.

[4] Bekri, N. E., Peinsipp-Byma, E., und Syndikus, A. (2016). Cluster Rule Based Algorithm for Detecting Incorrect Data Records. In *2016 UKSim-AMSS 18th International Conference on Computer Modelling and Simulation (UKSim)*, Seiten 67–71. IEEE.

[5] Bo, F., Wenning, H., Gang, C., Dawei, J., und Shuining, Z. (2012). An improved PAM algorithm for optimizing initial cluster center. In *2012 IEEE International Conference on Computer Science and Automation Engineering*, Seiten 24–27. IEEE.

[6] Borrison, R., Kloepper, B., und Mullen, J. (2019). Data Preparation for Data Mining in Chemical Plants using Big Data. In *2019 IEEE 17th International Conference on Industrial Informatics (INDIN)*, Seiten 1185–1191. IEEE.

[7] Bramer, M. (2020). *Principles of Data Mining*. Springer London, London.

[8] Brin, S., Motwani, R., Ullman, J. D., und Tsur, S. (1997). Dynamic itemset counting and implication rules for market basket data. In Peckman, J. M., Ram, S., und Franklin, M., Herausgeber, *Proceedings of the 1997 ACM SIGMOD international conference on Management of data - SIGMOD '97*, Seiten 255–264, New York, New York, USA. ACM Press.

[9] Bruno, G., Garza, P., Quintarelli, E., und Rossato, R. (2007). Anomaly Detection in XML databases by means of Association Rules. In *18th International Conference on Database and Expert Systems Applications (DEXA 2007)*, Seiten 387–391. IEEE.

[10] Chmielewski, M. R. und Grzymala-Busse, J. W. (1996). Global discretization of continuous attributes as preprocessing for machine learning. *International Journal of Approximate Reasoning*, 15(4):319–331.

[11] Cleve, J. und Lämmel, U. (2016). *Data Mining*. Studium. De Gruyter Oldenbourg, Berlin and Boston, 2. auflage. Auflage.

[12] Cong, Y. (2020). Research on Data Association Rules Mining Method Based on Improved Apriori Algorithm. In *2020 International Conference on Big Data & Artificial Intelligence & Software Engineering (ICBASE)*, Seiten 373–376. IEEE.

[13] Drees, B. (2016). Text und Data Mining: Herausforderungen und Möglichkeiten für Bibliotheken: Perspektive Bibliothek, Bd. 5, Nr. 1 (2016).

[14] Ester, M., Kriegel, H.-P., Sander, J., und Xu, X. (1996). *A density-based algorithm for discovering clusters in large spatial databases with noise.*

[15] Esteves, R. M., Hacker, T., und Rong, C. (2013). Competitive K-Means, a New Accurate and Distributed K-Means Algorithm for Large Datasets. In *2013 IEEE 5th International Conference on Cloud Computing Technology and Science*, Seiten 17–24. IEEE.

[16] Farkisch, K. (2011). *Data-Warehouse-Systeme kompakt.* Springer Berlin Heidelberg, Berlin, Heidelberg.

[17] Fayyad, U., Piatetsky-Shapiro, G., und Smyth, P. (1996). From Data Mining to Knowledge Discovery in Databases. *AI Magazine*, 3(17).

[18] Freitag, M., Kück, M., Ait Alla, A., und Lütjen, M. (2015). Potenziale von Data Science in Produktion und Logistik: Teil 2 - Vorgehensweise zur Datenanalyse und Anwendungsbeispiele. *Industrie 4.0 Management*, 35:39–46.

[19] Frochte, J. (2019). *Maschinelles Lernen: Grundlagen und Algorithmen in Python.* Hanser, München, 2., aktualisierte auflage. Auflage.

[20] Gavankar, S. und Sawarkar, S. (2015). Decision Tree: Review of Techniques for Missing Values at Training, Testing and Compatibility. In *2015 3rd International Conference on Artificial Intelligence, Modelling and Simulation (AIMS)*, Seiten 122–126. IEEE.

[21] Guan, Z., Ji, T., Qian, X., Ma, Y., und Hong, X. (2017). A Survey on Big Data Preprocessing. In *2017 5th Intl Conf on Applied Computing and Information Technology/4th Intl Conf on Computational Science/Intelligence and Applied Informatics/2nd Intl Conf on Big Data, Cloud Computing, Data Science (ACIT-CSII-BCD)*, Seiten 241–247. IEEE.

[22] Han, J., Kamber, M., und Pei, J. (2011). *Data Mining: Concepts and Techniques: Concepts and Techniques.* Elsevier professional, s.l., 3. aufl.. Auflage.

[23] Han, J., Pei, J., und Yin, Y. (2000). Mining frequent patterns without candidate generation. *ACM SIGMOD Record*, 29(2):1–12.

[24] Huang, Z. (1998). Extensions to the k-Means Algorithm for Clustering Large Data Sets with Categorical Values. *Data Mining and Knowledge Discovery*, 2(3):283–304.

[25] Joshi, R., Patidar, A., und Mishra, S. (2011). Scaling k-medoid algorithm for clustering large categorical dataset and its performance analysis. In *2011 3rd International Conference on Electronics Computer Technology*, Seiten 117–121. IEEE.

[26] Kanjilal, S. und Sen, S. (2016). Data integration based approach to find shortest path within a city for different time periods. In *2016 5th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, Seiten 233–238. IEEE.

[27] Kantardzic, M. (2020). *Data mining: Concepts, models, methods, and algorithms.* IEEE Press and Wiley, Piscataway NJ and Hoboken New Jersey, third edtion. Auflage.

[28] Karo, I. M. K. und Huda, A. F. (2016). Spatial clustering for determining rescue shelter of flood disaster in South Bandung using CLARANS Algorithm with Polygon Dissimilarity Function. In *2016 12th International Conference on Mathematics, Statistics, and Their Applications (ICMSA)*, Seiten 70–75. IEEE.

[29] Kaufman, L. und Rousseeuw, P. J. (2005). *Finding groups in data: An introduction to cluster analysis*. Wiley-Interscience paperback series. Wiley, Hoboken, N.J.

[30] Khan, I., Luo, Z., Huang, J. Z., und Shahzad, W. (2020). Variable Weighting in Fuzzy k-Means Clustering to Determine the Number of Clusters. *IEEE Transactions on Knowledge and Data Engineering*, 32(9):1838–1853.

[31] Khoshgoftaar, T. M. und Seliya, N. (2002). Software quality classification modeling using the SPRINT decision tree algorithm. In *14th IEEE International Conference on Tools with Artificial Intelligence, 2002. (ICTAI 2002). Proceedings*, Seiten 365–374. IEEE Comput. Soc.

[32] Kotu, V. (2014). *Predictive Analytics and Data Mining: Concepts and Practice with RapidMiner*. Elsevier Science, Burlington.

[33] Koval, S. I. (2018). Data preparation for neural network data analysis. In *2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*, Seiten 898–901. IEEE.

[34] Li, M., Zhou, Y., Tang, W., und Lu, L. (2020). K-modes Based Categorical Data Clustering Algorithms Satisfying Differential Privacy. In *2020 International Conference on Networking and Network Applications (NaNA)*, Seiten 86–91. IEEE.

[35] Li, X. und Zhong, S. (2010). Data Integration Based on Mining Value Chain. In *2010 International Conference on E-Product E-Service and E-Entertainment*, Seiten 1–4. IEEE.

[36] Li, Z.-C., He, P.-L., und Lei, M. (2005). A high efficient AprioriTid algorithm for mining association rule. In *2005 International Conference on Machine Learning and Cybernetics*, Seiten 1812–1815 Vol. 3. IEEE.

[37] Liu, X. und Liu, H. (2007). Topological Cluster: A Generalized View for Density-based Spatial Clustering. In *2007 11th International Conference on Computer Supported Cooperative Work in Design*, Seiten 7–12. IEEE.

[38] Lopez, I. D., Figueroa, A., und Corrales, J. C. (2020). Multi-Dimensional Data Preparation: A Process to Support Vulnerability Analysis and Climate Change Adaptation. *IEEE Access*, 8:87228–87242.

[39] Narayanan, S., Jaiswal, A., Chiang, Y.-Y., Geng, Y., Knoblock, C. A., und Szekely, P. (2014). Integration and Automation of Data Preparation and Data Mining. In *2014 IEEE International Conference on Data Mining Workshop*, Seiten 1076–1085. IEEE.

[40] Ng, R. T. und Han, J. (2002). CLARANS: a method for clustering objects for spatial data mining. *IEEE Transactions on Knowledge and Data Engineering*, 14(5):1003–1016.

[41] Olukanmi, P. O., Nelwamondo, F., und Marwala, T. (2019). Learning the k in k-means via the Camp-Meidell Inequality. In *2019 6th International Conference on Soft Computing & Machine Intelligence (ISCMI)*, Seiten 211–216. IEEE.

[42] Olukanmi, P. O., Nelwamondo, F., und Marwala, T. (2020). k- Means-MIND: An Efficient Alternative to Repetitive k-Means Runs. In *2020 7th International Conference on Soft Computing & Machine Intelligence (ISCMI)*, Seiten 172–176. IEEE.

[43] Panda, B. S. und Kumar Adhikari, R. (2020). A Method for Classification of Missing Values using Data Mining Techniques. In *2020 International Conference on Computer Science, Engineering and Applications (ICCSEA)*, Seiten 1–5. IEEE.

[Priyam et al.] Priyam, A., Abhijeet, Gupta, R., Rathee, A., und Srivastava, S. Comparative analysis of decision tree classification algorithms. In *International Journal of Current Engineering and Technology*, Band 3, Seiten 334–337.

[45] Rehman, F. u., Abbas, M., Murtaza, S., Butt, W. H., Rehman, S., und Qamar, U. (2018). SimFiller. Similarity-Based Missing Values Filling Algorithm. In *2018 Thirteenth International Conference on Digital Information Management (ICDIM)*, Seiten 77–81. IEEE.

[46] Rosati, S., Balestra, G., Giannini, V., Mazzetti, S., Russo, F., und Regge, D. (2015). ChiMerge discretization method: Impact on a computer aided diagnosis system for prostate cancer in MRI. In *2015 IEEE International Symposium on Medical Measurements and Applications (MeMeA) Proceedings*, Seiten 297–302. IEEE.

[47] Runkler, T. A. (2010). *Data Mining: Methoden und Algorithmen intelligenter Datenanalyse*. Vieweg + Teubner, Wiesbaden, 1. aufl.. Auflage.

[48] Saffer, J. D. und Burnett, V. L. (2014). Introduction to biomedical literature text mining: context and objectives. *Methods in molecular biology (Clifton, N.J.)*, 1159:1–7.

[49] Saha, B. und Srivastava, D. (2014). Data quality: The other face of Big Data. In *2014 IEEE 30th International Conference on Data Engineering*, Seiten 1294–1297. IEEE.

[50] Sathya Durga, V. und Jeyaprakash, T. (2019). An Effective Data Normalization Strategy for Academic Datasets using Log Values. In *2019 International Conference on Communication and Electronics Systems (ICCES)*, Seiten 610–612. IEEE.

[51] Shafer, J., Agrawal, R., und Mehta, M. (1996). SPRINT: A scalable parallel classifier for data mining. 96:544–555.

[52] Shah, C. und Jivani, A. (2013). Comparison of data mining clustering algorithms. In *2013 Nirma University International Conference on Engineering (NUiCONE)*, Seiten 1–4. IEEE.

[53] Song, C. (2016). Research of association rule algorithm based on data mining. In *2016 IEEE International Conference on Big Data Analysis (ICBDA)*, Seiten 1–4. IEEE.

[54] Steinley, D. und Brusco, M. J. (2011). Choosing the number of clusters in K-means clustering. *Psychological methods*, 16(3):285–297.

[55] Tibshirani, R., Walther, G., und Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2):411–423.

[56] Vijaya, Sharma, S., und Batra, N. (2019). Comparative Study of Single Linkage, Complete Linkage, and Ward Method of Agglomerative Clustering. In *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*, Seiten 568–573. IEEE.

[57] Visalakshi, S. und Radha, V. (2014). A literature review of feature selection techniques and applications: Review of feature selection in data mining. In *2014 IEEE International Conference on Computational Intelligence and Computing Research*, Seiten 1–6. IEEE.

[58] Wang, X.-F. und Huang, D.-S. (2009). A Novel Density-Based Clustering Framework by Using Level Set Method. *IEEE Transactions on Knowledge and Data Engineering*, 21(11):1515–1531.

[59] Xu, X., Lei, Y., und Li, Z. (2020). An Incorrect Data Detection Method for Big Data Cleaning of Machinery Condition Monitoring. *IEEE Transactions on Industrial Electronics*, 67(3):2326–2336.

[60] Yanbo, W., Huiqiang, W., Xuefei, J., und Ming, Y. (2010). Research of AprioriHybird algorithm and application in network situational awareness. In *2010 3rd International Conference on Computer Science and Information Technology*, Seiten 170–172. IEEE.

[61] Ye, Y. und Chiang, C.-C. (2006). A Parallel Apriori Algorithm for Frequent Itemsets Mining. In *Fourth International Conference on Software Engineering Research, Management and Applications (SERA'06)*, Seiten 87–94. IEEE.

[62] Zhao, H., Han, Q., und Pan, H. (2010). A Hierarchical Clustering Algorithm Based on Grid Partition. In *2010 International Conference on Multimedia Communications*, Seiten 187–190. IEEE.

[63] Zheng, J., Zhang, D., Leung, S. C. H., und Zhou, X. (2010). An efficient algorithm for frequent itemsets in data mining. In *2010 7th International Conference on Service Systems and Service Management*, Seiten 1–6. IEEE.

Anhang

Anhang A Anhang