IT in Produktion und Logistik | Fakultät Maschinenbau | Technische Universität Dortmund

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

MASTERARBEIT

zur Erlangung des Grades M. Sc.

Systematisierung von Data-Mining-Verfahren in Klassifikatoren und deren prototypische Implementierung und Validierung in RapidMiner

Vorgelegt von: Tobias Klein

Matrikel-Nr.: 222578
Studiengang: Logistik

Ausgegeben am: 17.05.2022
Eingereicht am: 14.11.2022

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

Inhaltsverzeichnis

Abkürzungsverzeichnis

CRISP-DM	Cross Industry Standard Process for Data Mining
DBMS	Datenbank-Managementsystem
EPK	Ereignisgesteuerte Prozesskette
et al.	et alii (lat. „und andere“)
ID	Identifikationsnummer
IDC	International Data Corporation
KDD	Knowledge Discovery in Databases
KNN	K-Nearest-Neighbor
MAR	missing at random
MCAR	missing completely at random
NMAR	not missing at random
PCA	Principal Component Analysis
RMSE	Root Mean Squared Error
TU Dortmund	Technische Universität Dortmund
ZB	Zettabyte

Abbildungsverzeichnis

Tabellenverzeichnis

1 Einleitung

„There is an urgent need for a new generation of computational theories and tools to assist humans in extracting useful information (knowledge) from the rapidly growing volumes of digital data."
(Fayyad et al. 1996a)

Bereits vor über 25 Jahren waren sich Forschende bewusst: Daten werden in so rasantem Tempo gesammelt, dass Menschen zukünftig von computergestützten Theorien und Werkzeugen unterstützt werden müssen, um effizient nützliches Wissen in Daten entdecken zu können (Fayyad et al. 1996a). Weltweit werden von Unternehmen und Institutionen Daten gesammelt. Handelsunternehmen sammeln Kundendaten für Marktanalysen, Forschungsinstitute speichern Versuchsdaten zu Forschungszwecken, Behörden speichern Daten aus Gründen des öffentlichen Interesses und produzierende Unternehmen sammeln Daten für eventuell mögliche Prozessoptimierungsanalysen (Cleve und Lämmel 2020). Aufgrund der vorhanden technischen Möglichkeiten werden Informationen von Suchmaschinen und sozialen Netzwerken immer feiner gegliedert und automatisiert gesammelt (Stahl und Staab 2017).

Insgesamt wird damit ein rasant wachsendes Datenvolumen generiert. In einem Bericht der IDC („International Data Corporation") wurde die Summe der weltweit erzeugten und verarbeiteten Daten geschätzt und als *global Datasphere* bezeichnet. Demnach betrug die *global Datasphere* 2010 zwei ZB (Zettabyte) und 2018 bereits 33 ZB. Für 2025 wird eine *global Datasphere* von 175 ZB geschätzt. (Reinsel et al. 2018) Ein ZB entspricht einer Billion Gigabyte (Aust 2021).

Die Daten werden dabei aus unterschiedlichen Gründen gesammelt und gespeichert. Zum einen ist dies für Unternehmen technisch und finanziell möglich und zum anderen wird eine Speicherung gewisser Daten über einen bestimmten Zeitraum sogar gesetzlich gefordert. Die Sammlung von Daten geht jedoch meist über die gesetzlichen Anforderungen hinaus, da Unternehmen erwarten, einen Mehrwert aus diesen Daten zu generieren. Mithilfe leistungsfähiger Computer können diese Daten nicht nur gespeichert, sondern auch analysiert werden, um neue und nützliche Informationen zu generieren. Bei der Menge an Daten lässt sich durch bloße Sichtung oder die Erstellung von Tabellen mithilfe einfacher Tabellenkalkulationstools jedoch keine aussagekräftige Auswertung erstellen. Selbst statistische Analyseverfahren stoßen hier an ihre Grenzen. Hierfür werden, wie schon 1996 von Forschenden erkannt, leistungsstarke Datenanalyse-Techniken benötigt. Ziel dieser Datenanalyse-Techniken ist es, neue Muster, also Zusammenhänge, in den Daten zu erkennen. Diese Suche ist Gegenstand des Data Minings, weshalb diese Datenanalyse-Techniken auch Data-Mining-Verfahren genannt werden. Dabei wird die Datenmenge mithilfe dieser Verfahren verarbeitet, um bisher unbekannte Strukturen, Trends, Zusammenhänge oder Muster in den Daten zu entdecken. (Cleve und Lämmel 2020)

Üblicherweise werden Data-Mining-Verfahren sowohl in der Fachliteratur als auch in spezieller Data-Mining-Software nach ihrer Aufgabenstellung strukturiert (Beekmann und Chamoni 2006). Die Strukturierung erfolgt demnach mit der Voraussetzung, schon vor der Analyse der Daten mithilfe der Data-Mining-Verfahren wissen zu müssen, welches Analyseverfahren verwendet werden soll. Da die verschiedenen Verfahren verschiedene

Anforderungen an die Datengrundlage haben, muss zuerst bekannt sein, welches Verfahren angewendet werden soll. Danach kann anhand der definierten Aufgabenstellung eine passende Datengrundlage ausgewählt, beziehungsweise durch eine gezielte Datenvorverarbeitung geschaffen werden. (Cleve und Lämmel 2020; Düsing 2006)

Das Hauptziel dieser Arbeit ist es, eine neue Systematisierung von Data-Mining-Verfahren anhand von Klassifikatoren vorzunehmen. Dadurch soll ermöglicht werden, einen Datenbestand zu analysieren, ohne vorher zwingend eine Aufgabenstellung definieren zu müssen, indem mithilfe einer strukturierten Betrachtung des Datenbestandes schon vor Beginn einer Analyse Aufgabenstellungen ausgeschlossen werden können. Dafür werden zuerst die wichtigsten Grundlagen in Data-Mining-Prozessen dargestellt. Zu Beginn werden die beiden gängigsten Vorgehensmodelle für Data-Mining Prozesse vorgestellt: Das Vorgehensmodell nach Fayyad und das industrienahe CRISP-DM Vorgehensmodell. Die einzelnen Phasen der Vorgehensmodelle werden erläutert. Auf die Datenvorverarbeitung wird anschließend gesondert eingegangen, da diese einen aufwendigen Schritt in Data-Mining-Prozessen darstellt. Die nach Aufgabenstellung strukturierten Klassifizierungsformen werden vorgestellt, um den aktuellen Stand der Forschung darzulegen. Danach werden die Klassifizierungsformen der Data-Mining-Software RapidMiner erläutert, um die Struktur der in dieser Arbeit angewendeten Data-Mining-Software zu erläutern. RapidMiner wurde an der TU Dortmund entwickelt, und verfügt über eine umfangreiche Auswahl an Data-Mining-Verfahren und für den Prozess erforderliche Operatoren (Mierswa et al. 2006). Durch eine grafische Oberfläche können die Operationen übersichtlich und ohne Programmieraufwand verknüpft werden, wodurch sich die Software für diese Arbeit anbietet.

Im Anschluss werden die Herausforderungen bei der Anwendung von Data-Mining-Verfahren erörtert, um mögliche Einflüsse auf die Strukturierung der Verfahren und auf die Verfahren allgemein zu erkennen.

Nach der Erarbeitung der notwendigen Grundlagen und der häufigsten Herausforderungen in Data-Mining-Prozessen wird ein Konzept zur Systematisierung von Data-Mining-Prozessen in Klassifikatoren erarbeitet. Die herkömmliche Herangehensweise bei der Verwendung von Data-Mining-Verfahren birgt die Herausforderung, schon vor der Analyse der Daten die zu verfolgende Aufgabenstellung festzulegen. Deshalb sollte auch die Datengrundlage als Ausgangspunkt für die Systematisierung der Data-Mining-Verfahren in Klassifikatoren in Betracht gezogen werden. Dazu werden zuerst die Anforderungen an eine Datengrundlage definiert. Anschließend wird darauf hingearbeitet, ein Klassifizierungskonzept mithilfe von Klassifikatoren zu erstellen. Hierfür sollen zunächst die Anforderungen an die Klassifikatoren erarbeitet, die Kriterien für Klassifikatoren definiert und schlussendlich Klassifikatoren erstellt werden. Um die Klassifikatoren objektiv bewerten zu können wird zudem eine Validierungsmethodik erarbeitet. Dabei soll sowohl auf eine explorative Bewertungsmethode als auch auf mögliche Kennzahlen zur Validierung der Klassifikatoren hingearbeitet werden.

Ein weiteres Ziel dieser Arbeit ist die prototypische Implementierung der erarbeiteten Klassifikatoren und eine Validierung dieser mithilfe der Software RapidMiner an öffentlich zugänglichen Beispieldatensätzen aus einem ökonomischen und produktionslogistischen Umfeld. Durch die Systematisierung in Klassifikatoren soll anhand der Kriterien eines vorhandenen Datenbestandes die Auswahl eines passenden Data-Mining-Verfahrens ermöglicht werden, ohne vorher eine Aufgabenstellung definieren zu müssen. Dazu werden die eindeutig definierten

Kriterien herangezogen und anschließend validiert. Die Arbeit endet mit einer objektiven Validierung durch Anwendung des erarbeiteten Bewertungskonzepts.

2 Grundlagen in Data-Mining-Prozessen

Ziel des Data Minings ist es, unentdecktes Wissen in Daten zu erkennen und dieses zu extrahieren. Wissen bedeutet dabei das Erkennen von interessanten Mustern, welche allgemein gültig, nicht trivial, nützlich, verständlich und neu sind. (Fayyad et al. 1996a; Runkler 2010; Hand et al. 2001) Ob die genannten Eigenschaften auf erkannte Muster zutreffen, wird in jedem Data-Mining-Prozess neu beurteilt. Dadurch entsteht ein Prozess, welcher so oft durchlaufen werden kann, bis ein für den Analysten zufriedenstellendes Ergebnis erreicht wird. (Runkler 2010) Wann das Ergebnis als zufriedenstellend angesehen wird, hängt vom Anwendungsfall ab.

Die für diese Arbeit relevanten Grundlagen eines solchen iterativen Prozesses werden in diesem Kapitel erläutert.

2.1 Gängige Vorgehensmodelle in Data-Mining-Prozessen

Eine strukturierte Anwendung von Data-Mining-Verfahren in Form eines Vorgehensmodells ist wichtig, um die Verfahren mit vorhergehenden und nachfolgenden Schritten auf die zu analysierenden Daten anzuwenden. Werden Data-Mining-Verfahren ohne eine Vorbereitung der Datengrundlage angewandt, führt dies häufig lediglich zu der Entdeckung von irrelevanten Informationen (Fayyad et al. 1996c; Kurgan und Musilek 2006).

Eines der ersten Vorgehensmodelle für Data-Mining-Prozesse wurde in Fayyad et al. 1996d vorgestellt. Das Vorgehensmodell nach Fayyad dient seitdem als Vorlage für weitere Modelle, sowohl in der Industrie als auch im akademischen Kontext. (Kurgan und Musilek 2006) Die Arbeiten von Fayyad haben mit einem h-Index von 21 eine weltweit hohe Wahrnehmung. Die meisten Zitation sind dabei auf Fayyad et al. 1996a zurückzuführen. (Scopus 2022) Das CRISP-DM Vorgehensmodell ist industrie- und praxisorientiert und baut auf früheren Modellen auf. (Wirth und Hipp 2000)

In den beiden folgenden Unterkapiteln werden diese beiden Vorgehensmodelle vorgestellt. So wird mit dem Vorgehensmodell nach Fayyad eine eher theoretische Herangehensweise und mit dem CRISP-DM Vorgehensmodell eine praxisorientierte Variante den typischen Ablauf eines Data-Mining-Prozesses verdeutlicht, um im weiteren Verlauf der Arbeit Ansatzpunkte für Klassifikatoren innerhalb des Data-Mining-Prozesses herauszuarbeiten.

2.1.1 Vorgehensmodell nach Fayyad

Im Jahr 1996 wurde mit der Veröffentlichung des Buches *Advances in Knowledge Discovery and Data Mining* von Fayyad et al. der Grundstein für die Entwicklung eines Vorgehensmodells gelegt. Das Vorgehensmodell, welches in dem 1996 veröffentlichen Artikel *From Data Mining to Knowledge Discovery in Databases* von Fayyad et al. übersichtlich dargestellt wird, wird in der Fachliteratur häufig zitiert, ist auf akademische Forschung ausgerichtet und diente als eines der ersten Modelle als Grundlage für neue und weiter entwickelte Vorgehensmodelle. (Kurgan und Musilek 2006)

Der nicht-triviale Prozess der Identifizierung gültiger, neuartiger, potenziell nützlicher und letztlich verständlicher Muster in Daten wird von Fayyad et al. als *Knowledge Discovery in Databases* (KDD) bezeichnet. Das eigentliche Data Mining ist lediglich ein Teilschritt des KDD (Cleve und Lämmel 2020). Muster sind dabei Zusammenhänge, die eine Teilmenge der Daten

oder ein auf eine Untermenge der Daten anwendbares Modell beschreiben. Allgemeiner formuliert geht es um das Auffinden von Strukturen in den Daten. Nicht-trivial bedeutet keine einfache Berechnung von vordefinierten Größen, wie beispielsweise die Berechnung eines Durchschnitts, sondern eine involvierte Suche oder Interferenz. Der Begriff Prozess impliziert bereits, dass der von Fayyad et al. definierte KDD-Prozess mehrere Phasen umfasst. (Fayyad et al. 1996a)

Insgesamt besteht der von Fayyad et al. definierte Prozess aus fünf Phasen: Datenselektion, Datenvorverarbeitung, Datentransformation, Data Mining und Evaluation & Interpretation (siehe Abbildung 2-1). (Fayyad et al. 1996a)

Je nach Literatur wird unter KDD der Data-Mining-Prozess als Ganzes verstanden oder KDD als Synonym für den Teilschritt Data Mining verwendet (Cleve und Lämmel 2020). In dieser Arbeit wird der Begriff KDD als Synonym für das Vorgehensmodell nach Fayyad verwendet und der Prozess der Wissensentdeckung wird als Data-Mining-Prozess bezeichnet.

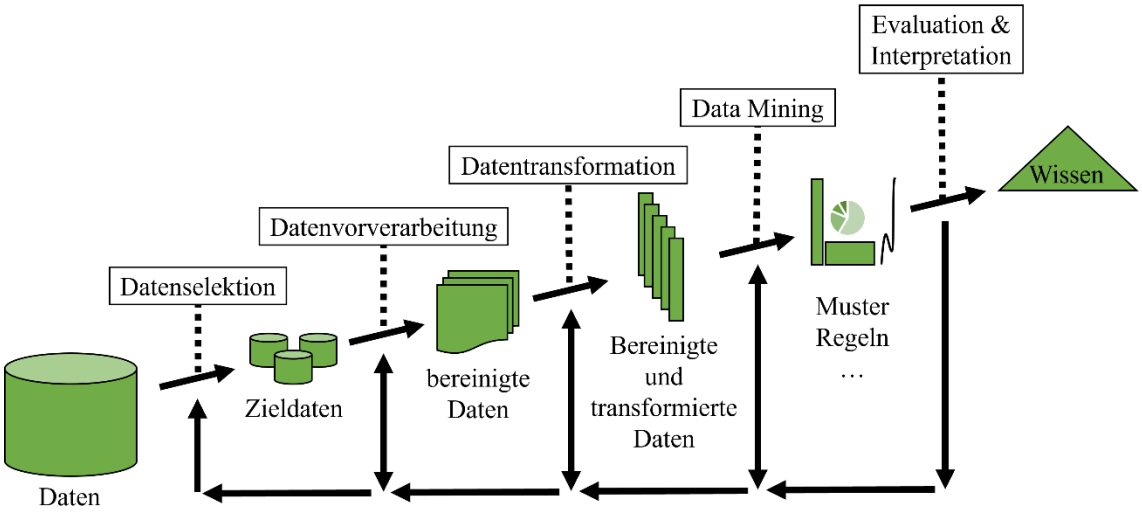


Abbildung 2-1: Vorgehensmodell nach Fayyad et al. 1996a; Cleve und Lämmel 2020

Die in Abbildung 2-1 dargestellten Phasen des Vorgehensmodells werden im Folgenden nach den Erläuterungen von Fayyad et al. 1996c zusammengefasst vorgestellt:

Datenselektion: Der Prozess beginnt damit, die verfügbaren Daten zu sichten, zu verstehen und eine Zielsetzung zu erarbeiten. So kann ein Zieldatensatz erstellt werden. Dazu wird der Zieldatensatz entweder extrahiert, oder es wird auf einer Teilmenge oder Stichprobe des ursprünglichen Datensatzes gearbeitet. Ergebnis dieser Phase sind die Zieldaten.

Datenvorverarbeitung: Zu dieser Phase gehören grundlegende Vorgänge wie das Entfernen von Ausreißern, Sammeln der erforderlichen Informationen zur Modellierung, und der Festlegung von Strategien für den Umgang mit fehlenden oder widersprüchlichen Daten. Das Ergebnis dieser Phase ist ein bereinigter Datensatz.

Datentransformation: Da die verschiedenen Data-Mining-Methoden verschiedene Anforderungen an die Datengrundlage haben, müssen die Daten nicht nur bereinigt, sondern auch entsprechend transformiert werden. Dazu gehört die Suche nach nützlichen Merkmalen zur Repräsentation der Daten. Welche Merkmale repräsentativ sind, hängt von der dem Prozess zugrundeliegenden Zielsetzung ab. Ebenso gehören eventuelle Methoden zur

Dimensionsreduktion zur Datentransformation. Ziel dieser Methoden ist es, die effektive Anzahl der betrachteten Variablen zu reduzieren. Als Ergebnis dieser Phase liegen bereinigte und transformierte Daten vor.

Data Mining: In dieser Phase findet das eigentliche Data Mining statt. Hierfür ist ein geeignetes Data-Mining-Verfahren für die Suche nach Mustern in den Daten zu wählen und anschließenden auf die Daten anzuwenden. Als Ergebnis dieser Phase werden bei erfolgreicher Anwendung Muster oder Regeln in den Daten erkannt.

Evaluation & Interpretation: In der letzten Phase werden die entdeckten Muster interpretiert und evaluiert. Dabei können die extrahierten Muster zum besseren Verständnis visualisiert werden. Redundante oder irrelevante Muster werden entfernt und die nützlichen Muster werden in für die Benutzer verständliche Begriffe überführt, da es sich bei den Benutzern nicht unbedingt um die Datenanalysten selbst handelt. Bei der Evaluation der extrahierten Muster können in vielen Fällen quantitative Bewertungsmethoden herangezogen werden. So ist es beispielsweise möglich, Maße für die geschätzte Klassifizierungsgenauigkeit oder den Nutzen in Form eines erwarteten Gewinns zur Evaluation zu verwenden. Wurden keine nützlichen Muster entdeckt, kann zu einer der vorherigen Phasen zurückgekehrt werden, um die Vorgehensweise zu überarbeiten. Dann können irrelevante Muster entfernt und Parametereinstellungen überarbeitet werden.

Beim KDD-Prozess handelt es sich um einen interaktiven und iterativen Prozess mit vielen Entscheidungen, die vom Anwender getroffen werden müssen. Dabei kann der Prozess beliebig viele Iterationen und Schleifen zwischen den Phasen enthalten. Der Ablauf der Phasen ist in Abbildung 2-1 dargestellt. Die Pfeile nach jeder Phase zeigen, dass bei nicht zufriedenstellenden Zwischenergebnissen zur vorherigen Phase zurückgekehrt werden kann. (Fayyad et al. 1996a)

2.1.2 CRISP-DM Vorgehensmodell

Ein industrieorientiertes Vorgehensmodell für Data-Mining-Prozesse ist das CRISP-DM (CRoss-Industry Standard Process for Data Mining) Vorgehensmodell (Kurgan und Musilek 2006). Ein erster Entwurf des Modells wurde durch ein Unternehmenskonsortium bereits Ende 1996 konzipiert, anschließend weiterentwickelt und im August 2000 vorgestellt (Wirth und Hipp 2000; Chapman et al. 2000). CRISP-DM baut auf früheren Versuchen auf, Methoden zur Wissensentdeckung zu definieren (Wirth und Hipp 2000). Das Vorgehensmodell ist phasenorientiert und hat das Ziel, einen Data-Mining-Prozess zu definieren, in welchem ein Analyse-Modell entwickelt und das erarbeitete Modell anschließend validiert wird. Die Phase des Data Mining, im CRISP-DM Vorgehensmodell in Abbildung 2-2 als Modeling bezeichnet, ist hierbei nicht nur ein einzelner Teilschritt, sondern iterativ mit der Datenvorverarbeitung verbunden. In Abbildung 2-2 ist der zyklische Charakter des Modells mit seinen sechs Phasen zu sehen. (Cleve und Lämmel 2020)

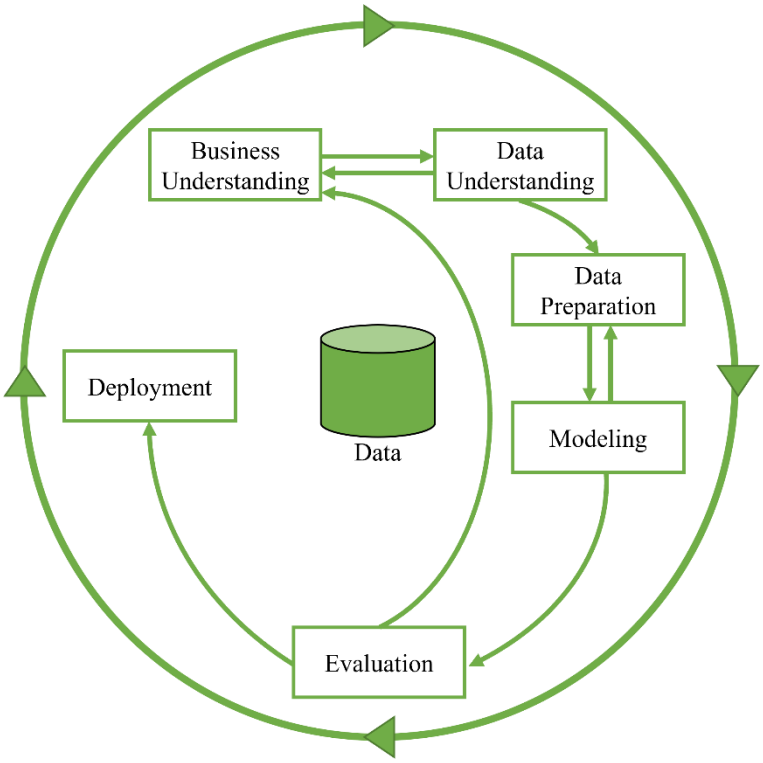


Abbildung 2-2: CRISP-DM Vorgehensmodell nach Chapman et al. 2000

Das CRISP-DM Vorgehensmodell bietet einen Überblick über den Lebenszyklus eines Data-Mining-Projekts. Im Gegensatz zum Vorgehensmodell nach Fayyad zeigt das Vorgehensmodell lediglich die jeweiligen Aufgaben der Phasen und nicht ihre Ergebnisse. Das bedeutet nicht, dass die einzelnen Phasen keine Ergebnisse haben, die in den nächsten Phasen genutzt werden. Die Absenz der Ergebnisse in Abbildung 2-2 verdeutlicht vielmehr, dass der Gesamtprozess und nicht die Ergebnisse einzelner Phasen im Vordergrund stehen. Es enthält die Phasen eines Projekts mit ihren jeweiligen Aufgaben und die Beziehungen zwischen den Aufgaben. Beziehungen können zwischen beliebigen Aufgaben bestehen, abhängig von den jeweiligen Zielen, dem Hintergrund, dem Interesse des Anwenders und vor allem den Daten. (Wirth und Hipp 2000)

Insgesamt gliedert sich der Lebenszyklus eines Data-Mining-Projekts im CRISP-DM Vorgehensmodell in sechs Phasen, die in Abbildung 2-2 dargestellt sind. Dabei ist auch zu beachten, dass die Reihenfolge der Phasen nicht festgelegt ist. Ein Wechsel zwischen den verschiedenen Phasen ist jederzeit möglich. Der äußere Kreis in Abbildung 2-2 soll den zyklischen Charakter eines Data-Mining-Projekts verdeutlichen. Damit ist gemeint, dass ein Data-Mining-Projekt nicht endet, nur weil eine Lösung eingesetzt wird. Während des gesamten Prozesses und auch nach Implementierung der Lösung können neue Erkenntnisse gewonnen werden und so präzisere oder neue Zielbeschreibungen formuliert werden. Aus diesen Erfahrungen können nachfolgende Data-Mining-Prozesse profitieren. (Chapman et al. 2000)

Die sechs Phasen werden im Folgenden nach dem *CRISP-DM 1.0 Step-by-step data mining guide* von Chapman et al. vorgestellt:

Business Understanding: In der ersten Phase steht das Verständnis der Projektziele und Projektanforderungen aus Sicht des Unternehmens im Vordergrund. Das so erlangte Wissen soll dann in eine Zielbeschreibung und einen vorläufigen Plan zur Erreichung der Ziele einfließen.

Data Understanding: Die Phase des Datenverständnisses beginnt damit, die benötigten Daten zusammenzustellen und sich mit den Daten vertraut zu machen. Bei einem ersten Blick in die Daten können Qualitätsprobleme innerhalb der Daten identifiziert und eventuell sogar interessante Teilmengen erkannt werden, um Hypothesen über verborgene Information aufstellen zu können.

Data Preparation: Diese Phase umfasst alle Aktivitäten die nötig sind, um die Rohdaten in den Datensatz zu überführen, auf den die Data-Mining-Verfahren angewendet werden können. Die Aktivitäten der Datenvorbereitung werden häufig mehrmals durchlaufen. Zu den Aufgaben gehören die Auswahl von Tabellen, Datensätzen und erforderlichen Attributen sowie die Transformation und Bereinigung der Daten, um diese für die Data-Mining-Verfahren vorzubereiten.

Modelling: In dieser Phase werden Data-Mining-Verfahren ausgewählt und angewendet. Dabei werden ihre Parameter auf optimale Werte kalibriert. Da einige Verfahren spezifische Anforderungen an die Datengrundlage haben, ist möglicherweise ein Rückschritt zur Data Preparation erforderlich, um die Daten entsprechend vorzubereiten. Die Phasen weisen einen iterativen Charakter auf, was auch die zwei Pfeile in Abbildung 2-2 zwischen Data Preparation und Modelling verdeutlichen.

Evaluation: Wurden ein oder mehrere Modelle erstellt, die aus Sicht des Anwenders von ausreichender Qualität sind, ist es unbedingt erforderlich, die Modelle gründlich zu evaluieren. So soll sichergestellt werden, dass ein Modell die Anforderungen auch wirklich erfüllt und neue Muster entdeckt wurden.

Deployment: In der letzten Phase des Modells soll das entwickelte Modell integriert und angewendet werden. Dabei ist es wichtig, gewonnenes Wissen so zu organisieren und zu präsentieren, dass es der Anwender und nicht nur der Datenanalyst nutzen kann. Je nach Anforderung kann diese Phase unterschiedlich komplex sein.

Beide Vorgehensmodelle beginnen mit datenvorverarbeitenden Schritten und die Datengrundlage ist ein wichtiger Faktor für die Entwicklung von Klassifikatoren. Daher beschäftigt sich das folgende Unterkapitel mit den Schritten der Datenvorverarbeitung in Data-Mining-Prozessen.

2.2 Datenvorverarbeitung in Data-Mining-Prozessen

Ein wichtiger Aspekt eines Data-Mining-Prozesses ist der Zeitaufwand der unterschiedlichen Phasen (Cios und Kurgan 2005). Je nach Vorgehensmodell gibt es unterschiedliche Phasen, aber sowohl das Vorgehensmodell nach Fayyad, als auch das CRISP-DM Vorgehensmodell enthalten Phasen, die sich auf das Verständnis der Domäne und Daten, der Datenvorverarbeitung, des Data Mining selbst und der Evaluierung sowie Integration der Ergebnisse beschäftigen (Kurgan und Musilek 2006). Um einen besseren Überblick über die aufgewendete Zeit innerhalb der jeweiligen Phasen eines Data-Mining-Prozesses zu erhalten, ist es sinnvoll, diesen Zeitaufwand im Verhältnis zur für den gesamten Prozess aufgewendeten Zeit zu betrachten. Schätzungen zufolge werden, wie in Abbildung 2-3 veranschaulicht, 10 bis 20 % des Zeitaufwands für das Verständnis der Domäne und der Daten, 30 bis 60 % für die Datenvorverarbeitung, 10 bis 25 % für das tatsächliche Data Mining und 5 bis 10 % für Evaluation und Integration aufgewendet. (Cios und Kurgan 2005)

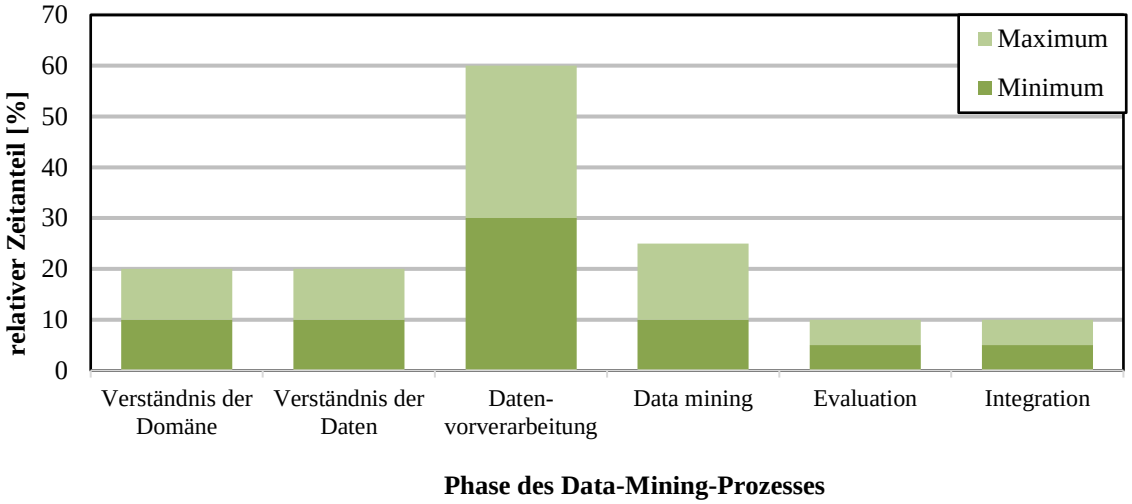


Abbildung 2-3: Relativer Zeitaufwand der Phasen eines Data-Mining-Prozesses nach Kurgan und Musilek 2006

Bei einem Vergleich der Zeitaufwände und Betrachtung der Abbildung 2-3 wird deutlich, dass die vorbereitenden Schritte, also das Erlangen eines Verständnisses für die Domäne und der Daten sowie der Datenvorverarbeitung, den höchsten relativen Zeitaufwand haben. Der Schritt des Data Mining selbst hat einen vergleichsweisen geringen Zeitaufwand. Für die nachbereitenden Schritte, also die Evaluation und anschließende Integration fällt der geringste Zeitaufwand an. Im Vorgehensmodell nach Fayyad gehören die Phasen der Datenselektion, Datenvorverarbeitung und Datentransformation zu den vorbereitenden Schritten. Im CRISP-DM Vorgehensmodell stellen die Phasen des Business und Data Understanding sowie die Data Preparation die vorbereitenden Schritte dar.

Daten aus der realen Welt sind in der Regel unvollständig, verrauscht oder inkonsistent. Dadurch werden interessante Muster in den Daten verschleiert. Fehlende Attributwerte, das komplette Fehlen relevanter Attribute oder zu stark aggregierte Daten können zu einem unvollständigen Datensatz führen. Aufgabe der vorbereitenden Schritte ist es deshalb, den Datensatz für die weitere Analyse so zu bearbeiten, dass dieser um die fehlenden Daten erweitert und von irrelevanten Attributen, Anomalien sowie Duplikaten befreit wird. Es wird schnell deutlich, dass es sich bei diesen Schritten um aufwendige Aufgaben handelt. (Zhang et al. 2003)

Ziel der Datenvorverarbeitung ist es, die Qualität der Daten zu verbessern und somit die Chancen auf eine erfolgreiche Analyse zu steigern. Da sich jedes Data-Mining-Projekt voneinander unterscheidet, ist es wichtig, ein grundlegendes Verständnis über den vorliegenden Datensatzes zu erlangen. Eine einfache und sinnvolle Methode dazu können einfache statistische Tests sein, um eine erste Übersicht über die Daten zu erhalten. (Cleve und Lämmel 2020) Die Datenerkundung für eine erste Übersicht wird auch als Data Exploration bezeichnet. Dabei handelt es sich um einen kontinuierlichen Prozess, der während des gesamten Projektes immer wieder stattfindet, um einen Überblick über die Daten zu behalten. (Aust 2021) Handelt es sich um numerische Attribute, geben die statistische Maße wie arithmetisches Mittel, Median, Standardabweichung, Maximum und Minimum einen schnellen ersten Eindruck. Bei nominalen und ordinalen Daten verschaffen die Angaben über die möglichen Werte und deren Häufigkeit

ein gutes erstes Verständnis. Sowohl bei numerischen als auch bei nominalen und ordinalen Daten ist die Zahl der fehlenden Werte interessant. (Cleve und Lämmel 2020)

Der zeitlich hohe Aufwand der vorbereitenden Schritte und insbesondere der Datenvorverarbeitung hat jedoch, wie bereits in Kapitel 2.1 deutlich wird, einen wichtigen Grund für den gesamten Prozess: Bei schlecht vorbereiteten Daten liefern die Data-Mining-Verfahren keine guten Ergebnisse. Die Verfahren selbst sind nicht in der Lage, die Fehler und Ungenauigkeiten der Datengrundlage zu korrigieren. Deshalb befasst sich die Datenvorverarbeitung intensiv mit der Qualität der Daten. (Cleve und Lämmel 2020)

Sobald ein grundsätzliches Verständnis über die Domäne und die Daten erlangt wurde, kann mit der Datenvorverarbeitung begonnen werden. Die wichtigsten Arten und Phasen der Datenvorverarbeitung werden in den folgenden Kapiteln erläutert.

2.2.1 Datenselektion und -integration

Vor Beginn eines Data-Mining-Projektes müssen die passenden Daten bereitgestellt und ausgewählt werden: Die Datenselektion. Diese Daten stammen häufig aus unterschiedlichen Tabellen oder Datenbanken und müssen nun zu einer Datentabelle zusammengeführt werden. Dies ist der Schritt der Datenintegration. Dabei können schon Probleme auftreten: Jeder Standort eines zusammengehörigen Unternehmens könnte eine eigene Datenbank betreiben, die Datenbanken verschiedener Standorte könnten unterschiedliche Strukturen besitzen und Attribute können unterschiedliche Bedeutungen in unterschiedlichen Datenbanken haben. Liegen mehrere Datensätze vor, ist es Aufgabe der Datenintegration diese so zusammenzuführen, dass idealerweise eine konsistente Tabelle mit schlüssigen Datensätzen vorliegt. (Cleve und Lämmel 2020)

Dabei können folgende Probleme auftreten: (Cleve und Lämmel 2020)

Entitätenidentifikationsproblem: Hierbei besitzen Merkmale dieselbe Semantik. Beispielsweise könnte ein Attribut *Nutzer_ID* und eins *Nutzerkennung* heißen, wobei sich sofort die Frage stellen würde, ob die Attribute die gleiche Bedeutung haben. Hierüber können die Metadaten Auskunft geben. (Cleve und Lämmel 2020)

Redundanzen: Durch Inkonsistenzen in der Nomenklatur von Attributen oder Dimensionen können Redundanzen entstehen. Die Attribute *Geschlecht* und *geschlecht* sind beispielsweise syntaktisch unterschiedlich, haben aber dieselbe Semantik. (Cleve und Lämmel 2020)

Widersprüche: Werden verschiedene Datenquellen zusammengeführt, können dabei Widersprüche entstehen. Zum Beispiel kann es verschiedene Nachnamen für dieselbe Person geben. (Cleve und Lämmel 2020)

Datenwertkonflikte: Ein Attribut aus unterschiedlichen Datensätzen kann unterschiedlich formatierte Attributsausprägungen haben. Ein Beispiel hierfür ist das in Deutschland gebräuchliche Datumsformat am Beispiel des vierten März 2022 mit 04.03.2022 und das in den Vereinigten Staaten von Amerika gebräuchliche Format mit 03-04-2022. Ein weiteres Beispiel hierfür sind Gewichtsangaben in Kilogramm oder Pfund. (Cleve und Lämmel 2020)

2.2.2 Datenbereinigung

Die für ein Data-Mining-Projekt vorliegenden Daten können teilweise unvollständig, fehlerhaft, mit Ausreißern behaftet oder inkonsistent vorliegen. Aufgabe der Datenbereinigung ist es, diese Probleme zu behandeln. Unbereinigte Daten führen zu Fehlern im Date-Mining-Prozess und damit schlussendlich zu unzuverlässigen oder sogar falschen Ergebnissen. Hierbei ist zu beachten, dass die neu eingefügten Daten so weit wie möglich informationsneutral sind, um die ursprünglichen Informationen nicht zu verfälschen oder zu verzerren. Dabei kann jedoch eine absolute Informationsneutralität bei eingefügten Werten nicht immer garantiert werden, da jeder eingefügte Wert kein Wert aus der reellen Welt darstellt. (Cleve und Lämmel 2020)

Im Folgenden wird auf die häufigsten Probleme eingegangen: (Cleve und Lämmel 2020)

Fehlende Daten

Bevor an einer passenden Lösung für fehlende Werte gearbeitet wird, muss unbedingt sichergestellt werden, dass es ein Fehler im eigentlichen Sinn ist. Die Werte können unbewusst, beispielsweise durch Vergessen, oder bewusst weggelassen werden. Ein Beispiel für bewusstes Weglassen von Daten wäre das Ausfüllen eines Antrags, bei dem eine Person eine Information nicht preisgeben will. Hierbei wäre dann möglicherweise das Wissen über die Intention für das Zurückhalten der Information nützlicher als die Information selbst. (Cleve und Lämmel 2020)

Insgesamt können fehlende Werte laut García et al. in drei Kategorien eingeteilt werden:

- *Missing at random (MAR)*
  Bei diesen fehlenden Werten ist ein Muster innerhalb der Daten erkennbar, welches erklärt, warum die Werte fehlen. Ob ein Wert innerhalb eines Datensatzes fehlt, ist abhängig von den anderen Werten im Datensatz, jedoch nicht von der Variable selbst. Das oben genannte Beispiel, bei dem eine Person bewusst Informationen beim Ausfüllen eines Antrags offenlässt, um beispielsweise ihr Alter nicht preiszugeben, wäre ein Fehler der Kategorie MAR. (García et al. 2015)
- *Missing completely at random (MCAR)*
  Bei fehlenden Werten dieser Kategorie handelt es sich um einen Sonderfall von MAR. Solche fehlenden Werte stehen auf keine Weise mit anderen Variablen in Verbindung, auch nicht indirekt. Es gibt also keine systematischen Unterschiede zwischen Datensätzen mit fehlenden Werten und Datensätzen ohne fehlende Werte. (García et al. 2015) Ein Beispiel für einen fehlenden Werte der Kategorie MCAR wäre die Beschädigung einer Probe in einem Labor, was dazu führt, dass die daraus resultierenden Beobachtungen fehlen. (Kaiser 2014)
- *Not missing at random (NMAR)*
  In diesem Fall ist das Fehlen des Wertes vom fehlenden Wert selbst abhängig und möglicherweise auch zusätzlich von anderen Werten des Datensatzes. (García et al. 2015) Fehlende Werte der Kategorie NMAR dürfen nicht ignoriert werden, da das Ergebnis sonst verfälscht werden könnte. Hierbei muss geklärt werden, warum die Informationen fehlen und die Werte müssen mithilfe dieser Informationen ersetzt werden. Ein Beispiel für einen Mechanismus, der fehlende Werte dieser Kategorie zur

Folge hat wäre ein Temperatursensor, der Temperaturen unter einem bestimmten Schwellenwert nicht erkennt. (Kaiser 2014)

Für den Umgang mit fehlenden Werten gibt es verschiedene Strategien. Handelt es sich um einen Datensatz, der aus Stichproben zusammengesetzt ist, kann der Datensatz um die Stichproben reduziert werden, die fehlende Werte enthalten. Eine weitere Strategie ist, fehlende Werte als besondere Werte zu kennzeichnen und gesondert zu behandeln. Es gibt auch einfache Verfahren, um die fehlenden Werte zu ersetzen. Solche Verfahren sind jedoch nur für fehlende Werte der Kategorie MCAR und MAR geeignet. Fehlende Werte der Kategorie NMAR können nicht durch simple Verfahren ersetzt werden. In diesem Fall muss geklärt werden, warum die Werte genau fehlen, und ein Modell entwickelt werden, um die fehlenden Werte zu schätzen. (Kaiser 2014)

Im Folgenden werden häufig genutzte einfache Lösungsmöglichkeiten für den Umgang mit fehlenden Werten erläutert:

- *Ersetzen durch den durchschnittlichen Wert:*
  Bei metrischen Attributen können fehlende Werte durch den Durchschnitt oder den Median des Attributs ersetzt werden. Dieses Vorgehen kann präzisiert werden, wenn sich das Data-Mining-Projekt mit Klassifikationsaufgaben beschäftigt. Dann können für die Berechnung des durchschnittlichen Wertes nur die Datensätze derselben Klasse verwendet werden. Handelt es sich nicht um Klassifikationsaufgaben, kann der Ansatz des k-Nearest-Neighbors verwendet werden. Dabei werden die durchschnittlichen Werte der Datensätze genommen, die dem aktuellen Datensatz am ähnlichsten sind. (Cleve und Lämmel 2020)
- *Lineare Interpolation*
  Bei der linearen Interpolation wird der fehlende Datenwert mithilfe der angrenzenden Datenpunkte geschätzt (Huang 2021). Bei der Berechnung der fehlenden Werte wird die Absolute Größe der angrenzenden Werte mit in die Berechnung einbezogen (Hettich et al. 2009). Es wurde nachgewiesen, dass das Ersetzen fehlender Werte durch Lineare Interpolation bessere Ergebnisse erzielen kann, als das Ersetzen durch den durchschnittlichen Wert (Noor et al. 2014).
- *Ersetzen durch den häufigsten Wert:*
  Eine einfache, aber wirksame Methode, ein nominales Attribut zu schätzen, ist das Verwenden des am häufigsten vorkommenden Wertes. Diese Methode kann jedoch nur bei Werten von Attributen angewendet werden, die eindeutig ungleichmäßig verteilt sind. Sind die Werte gleichwertig verteilt, kann die Anwendbarkeit dieses Ansatzes nicht gerechtfertigt werden. (Bramer 2020)
- *Attribut entfernen*:
  Die einfachste Strategie ist, Attribute, bei denen mindestens ein Wert fehlt komplett zu löschen. Vorteil dieser Strategie ist, dass keine Datenfehler mehr vorhanden sein können. Beim Entfernen von Attributen gehen jedoch möglicherweise wichtige Informationen verloren, wodurch die Zuverlässigkeit der aus den Daten gewonnen Informationen stark beeinträchtigt sein kann. Diese Strategie kann nur dann zum Einsatz kommen, wenn der Anteil der fehlenden Werte gering ist. Allgemein wird sie jedoch wegen der genannten Nachteile nicht empfohlen. Liegt ein hoher Anteil an

fehlenden Werten bei vielen Attributen vor, ist diese Strategie definitiv unbrauchbar. (Cleve und Lämmel 2020; Bramer 2020)

- *Ersetzung durch Schätzung des wahren Wertes:*
  Wenn der Anteil der fehlenden Werte einer Variablen gering ist, kann der fehlende Wert durch eine Schätzung des wahren Wertes ersetzt werden. Dies geschieht beispielsweise unter Zuhilfenahme statistischer Methoden (Cleve und Lämmel 2020). Ist die Anzahl der fehlenden Werte gering, wird dies wahrscheinlich nur kleine Auswirkungen auf die aus den Daten abgeleiteten Ergebnisse haben. Zu beachten ist hierbei jedoch, dass jeder geschätzte Wert Auswirkungen auf das Ergebnis hat. (Bramer 2020) Diese Lösungsmöglichkeit ist auch für Fehler der Kategorie NMAR geeignet (Kaiser 2014)

Rauschen und Ausreißer

Daten aus der realen Welt sind normalerweise nicht fehlerfrei sondern häufig verfälscht, was unter anderem durch Messfehler entstehen kann. Diese Verfälschungen werden auch als Rauschen bezeichnet. Die Interpretation der Daten und das Entwickeln von Modellen können dadurch unter Umständen stark beeinträchtigen werden. (García et al. 2015; Runkler 2010)

Rauschen kann besonders dann zu Problemen führen, wenn zur Analyse ein Zielattribut bestimmt wurde, da so die Beziehung zwischen den informativen Merkmalen und dem Ergebnis verändern kann. Deshalb kommt es durch Rauschen besonders bei Klassifikations- und Regressionsaufgaben zu verfälschten Modellen, wodurch die Wissensextraktion stark behindert wird. Die erstellten Modelle sollten demnach möglichst robust sein und nur wenig unter den Auswirkungen des Rauschens leiden. Je robuster der Algorithmus eines Data-Mining-Verfahrens ist, desto ähnlicher sind die Modelle, die dieser aus sauberen und aus verrauschten Daten erstellen würde. In der Literatur gibt es viele Ansätze die untersuchen, wie mit verrauschten Daten umzugehen ist, um eine möglichst hohe Genauigkeit zu erzielen. (García et al. 2015)

Die am häufigsten genutzten sind: (García et al. 2015)

- *Robust learners:*
  Hierbei handelt es sich um Algorithmen, die dadurch charakterisiert sind, dass diese weniger von rauschenden Daten beeinflusst, werden als andere. (García et al. 2015) Es gibt verschiedene Algorithmen, die dieses Kriterium erfüllen. Ein Beispiel aus RapidMiner ist zum Beispiel der K-Nearest-Neighbor-Algorithmus (KNN) (Arunadevi et al. 2018).
- *Rauschfilter:*
  Diese Filter identifizieren verrauschte Instanzen, welche dann aus den Trainingsdaten eliminiert werden können. Auf diese Weise können auch auf verrauschten Daten Algorithmen eingesetzt werden, die empfindlich gegenüber Rauschen sind. (García et al. 2015) Dabei ist jedoch zu beachten, dass die Daten hierbei verändert werden (Runkler 2010).
- *Klasseneinteilung (binning):*
  Hierbei werden verrauschte Daten gruppiert und anschließend durch Mittelwerte oder

Grenzwerte ersetzt (Cleve und Lämmel 2020). In RapidMiner kann die Anzahl der Intervalle manuell festgelegt, oder der Software selbst überlassen werden.

- *Regression*
  Für die Daten wird eine mathematische Funktion erstellt, die diese beschreibt. Im Anschluss werden die verrauschten Datenwerte durch die berechneten Funktionswerte, der mittels linearer Regression gefundenen Funktion ersetzt. (Cleve und Lämmel 2020)

Neben verrauschten Daten können aber auch Werte vorliegen, die stark von den Erwartungen abweichen (García et al. 2015). Solche Werte werden als Ausreißer bezeichnet.

Ausreißer können unter anderem durch Erfassungs- und Verarbeitungsfehler verursacht werden, die wiederum zu fehlerhaften Daten oder Datensätzen führen können. Besonders dort, wo Daten manuell erfasst werden, ist die Wahrscheinlichkeit der Entstehung solcher Fehler hoch. Beispiele dafür können sein: Vertauschen einzelner Ziffern, Tippfehler, Verwechseln einer Einheit (z.B. Zentimeter anstatt Meter), Verwenden eines falschen Formats (z.B. Datumsformat), Verwenden unterschiedlicher Dezimaltrennzeichen (Komma anstatt eines Semikolons oder Punktes). Beim Aufzählen von Beispielen wird schnell deutlich, dass bei manueller Erfassung von Daten an vielen Stellen Fehler entstehen können, die zu Ausreißern in den Daten führen. Ausreißer können aber auch beim Austausch von Daten zwischen Systemen entstehen, die verschiedene Datenformate verwenden. Ein bekanntes Beispiel hierfür ist die unterschiedliche Bedeutung von Punkt und Komma: Ein Punkt kann sowohl als ein Tausendertrennzeichen als auch als ein Dezimaltrennzeichen interpretiert werden. Bei fehlerbehaftetem Austausch könnten Werte so möglicherweise um den Faktor 1000 zu klein dargestellt werden. Bei solchen Fehlern handelt es sich um zufällige Fehler. (Runkler 2010)

Neben solchen zufälligen Fehlern kann es jedoch auch zu systematischen Fehlern kommen, die dann als Ausreißer in den Daten erkennbar sein können. Das können beispielsweise Messfehler sein, die durch falsche Kalibrierung oder Skalierung eines Messinstrumentes entstehen. Werden solche systematischen Fehler erkannt, können diese meist vollständig korrigiert werden. Voraussetzung dafür ist, dass die Systematik der Fehler erkannt wird. (Runkler 2010)

Zum Erkennen von Ausreißern gibt es in der Literatur verschiedene Ansätze. Im Folgenden werden häufig genutzte Verfahren vorgestellt:

- *Clustering*
  Von ähnlichen Werten können Cluster gebildet werden. Dies geschieht beispielsweise über dichtebasierte Verfahren. Werte, die außerhalb dieses Clusters liegen, werden hierbei dann als Ausreißer definiert. (Cleve und Lämmel 2020) Das Vorgehen entspricht der in Unterkapitel 2.3.1 beschriebenen Clusteranalyse (Beekmann und Chamoni 2006).
- *Kombinierte Maschine/Mensch-Untersuchung:*
  Zuerst wird mithilfe einer Software eine Liste der Werte erstellt, die diese als Ausreißer erkannt hat. Danach wird diese Liste von einem Menschen so gefiltert, dass dieser aufgrund seines Wissens und seiner Erfahrung die Werte entfernt, die er ebenfalls als Ausreißer definiert. (Cleve und Lämmel 2020)

Zum Umgang mit Ausreißern gibt es ebenfalls verschiedene Ansätze. Eine mögliche Methode ist der gleiche Umgang wie mit fehlerhaften Daten (Cleve und Lämmel 2020). Eine weitere bekannte, aber besonders bei großen Datensätzen ineffiziente Methode ist, die Markierung und gesonderte Behandlung von Ausreißern. Vorteil dabei ist, dass der Datensatz selbst vollständig bleibt. (Runkler 2010)

Inkonsistente und falsche Daten

Neben Fehlern auf struktureller Ebene, auf die im Unterkapitel 2.2.1 genauer eingegangen wird, gibt es noch weitere Fehlermöglichkeiten und -ursachen. Diese können durch menschliche Fehler wie Fehleinträge oder Schreibfehler entstehen, aber auch Einträge wie Abkürzungen oder Werte außerhalb eines zulässigen Wertebereichs können unter Umständen Probleme verursachen. Dabei können auch Werte auftreten, die ausgehend von ihrer Definition oder ihrem Wertebereich korrekt sind und daher nicht sofort als Ausreißer oder verrauscht auffallen, trotzdem aber unplausibel oder widersprüchlich sind. Beispiele dafür könnten ein vergleichsweise kleiner Lieferant mit einer einmalig großen Liefermenge oder der Städtename eines Empfängers, der nicht zu seiner Postleitzahl passt, sein. Ebenso kann es vorkommen, dass identische Datensätze mehrfach vorkommen, wodurch dieser Datensatz von vielen Verfahren als besonders wichtig angesehen werden würde. (Cleve und Lämmel 2020)

Um solche inkonsistenten und falschen Daten zu korrigieren ist häufig eine manuelle Korrektur erforderlich. Handelt es sich um die gleiche Domäne kann ein speziell auf das jeweilige Problem zugeschnittenes Vorverarbeitungsprogramm Abhilfe schaffen. Im allgemein gibt es jedoch nur zwei Möglichkeiten zur Datenkorrektur: Löschen oder die Zuhilfenahme anderer Datensätze. Nachteil des Löschens solcher Daten ist das Schrumpfen des Datenbestandes. Eine Korrektur unter der Zuhilfenahme anderer Datensätze kann sehr zeit- und arbeitsaufwendig werden. (Cleve und Lämmel 2020)

2.2.3 Datenreduktion

Wie in Kapitel 1 beschrieben, wächst das Datenvolumen stetig. Das führt dazu, dass immer größer werdende Datenmengen in Datenbanken gespeichert werden. Moderne Systeme sammeln aufgrund des *Datenvolumens*, der *Geschwindigkeit* der Datenverarbeitung, der *Veränderungsdynamik*, des voraussichtlichen unternehmerischen *Mehrwerts* und einer hohen geforderten *Datenqualität* der erfassten Daten immer komplexer werdende Datenströme, was letztlich zu den fünf V's von Big Data führt: *volume*, *velocity*, *variety*, *value* und *validity*. (Bachmann 2014). Da es jedoch effizienter ist und zu besseren Ergebnissen führt nur mit den relevanten Daten und nicht mit den rohen und möglicherweise redundanten, inkonsistenten und verrauschten Daten zu arbeiten, ist eine Reduktion der Daten sinnvoll. Ein weiteres bekanntes Problem ist, dass die Vielzahl an Variablen großer Datensätze den *Fluch der Dimensionalität* verursacht, der hohe Rechenressourcen erfordert, um verwertbare Muster aufzudecken. Im Folgenden werden einige wichtige Methoden, die zur Reduktion der Daten eingesetzt werden können, vorgestellt. (Muhammad Habib et al. 2016)

Aggregation

Unter Aggregation wird die Zusammenfassung mehrerer Datensätze untergeordneter

Aggregationsebenen zu einem Datensatz einer höheren Aggregationsebene verstanden. Häufig geschieht dies, indem Daten durch ihre Mittelwerte ersetzt oder einzelne Teilwerte zu ihrer Gesamtsumme aufsummiert werden. Eine solche Zusammenfassung ist auch immer mit einem Informationsverlust verbunden. Um diesem Verlust entgegenzuwirken, können auf einer höheren Aggregationsebene neue Merkmale wie beispielsweise Lage-, Streuungs- oder Zusammenhangsmaße, welche sich auf die aggregierten Merkmale beziehen, eingeführt werden. Dabei ist zu beachten, dass nicht zu viele neue Merkmale eingeführt werden, da die Aggregation sonst ihren Zweck nicht mehr erfüllen kann. Ein gewisser Informationsverlust ist also dabei unabdingbar. (Cleve und Lämmel 2020; Petersohn 2005)

Dimensionsreduktion

Die wichtigsten Strategien zur Datenreduktion sind Techniken der Dimensionsreduktion. Ziel dabei ist die Anzahl der im Datensatz vorhandenen Attribute oder Instanzen zu reduzieren (García et al. 2015). Sie empfiehlt sich dann, wenn die Anzahl der Merkmale so reduziert werden soll, dass Attribute mit starken Korrelationen zueinander gesondert behandelt werden sollen. (Petersohn 2005)

Dabei bieten sich folgende Vorgehensweisen an: (Petersohn 2005)

- *Manuelle Vorauswahl:*
  Bei dieser Vorgehensweise werden mithilfe subjektiver Erfahrungswerte der Analysten Analysemerkmale ausgewählt, die diese für geeignet erachten. Dabei werden für Gruppen von korrelierenden Merkmalen Repräsentanten der jeweiligen Gruppen ausgewählt. Nachteil dabei ist jedoch, dass die Bedeutung einzelner Merkmale unterschätzt werden kann. (Petersohn 2005)
  Hierbei kann es jedoch auch Attribute geben, die nur eine identische Ausprägung haben. Dann haben diese Attribute auch keinen informativen Mehrwert und können bedenkenlos gelöscht werden. Ein Beispiel für so ein Attribut könnte sein, wenn nur Daten innerhalb Deutschlands ausgewertet werden und das Attribut *Land* nur die Ausprägung *Deutschland* hat. (Cleve und Lämmel 2020)
- *Automatische Vorauswahl:*
  Mithilfe von Data-Mining-Verfahren können Korrelationen zwischen den Merkmalen festgestellt werden. Auf diese Weise werden die Zusammenhänge zwischen abhängigen und unabhängigen Merkmalen untersucht und schließlich automatisiert Repräsentanten selektiert. (Petersohn 2005)
- *Erstellen synthetischer Merkmale:*
  Hierbei werden synthetische Merkmale mit Varianten von Faktoranalysen berechnet und generiert. Danach liegt eine wesentlich kleinere Zahl von Attributen vor: relativ unkorrelierende Faktoren. Diese sind für nicht mathematisch-statistisch geschulte Analysten jedoch nur sehr schwer nachvollziehbar, was auch die Interpretation der Ergebnisse deutlich schwieriger macht. (Petersohn 2005)

Insgesamt kann die Dimensionsreduktion also auf zwei Arten erfolgen: über das Ausblenden einzelner Attribute und das Ermitteln und Zusammenfassen korrelierender Merkmale (Cleve und Lämmel 2020).

Numerische Datenreduktion

Bei der numerischen Datenreduktion wird eine für die Gesamtmenge repräsentative Teilmenge ausgewählt. Dies wird häufig über die Ziehung von Stichproben realisiert. Dadurch wird nicht die gesamte Datenmenge für die Analyse verwendet, sondern eine deutlich kleinere Stichprobe. (Cleve und Lämmel 2020)

Hierbei ist elementar, eine Stichprobe zu ziehen, die die realen Zusammenhänge der Grundgesamtheit widerspiegelt. Dafür können folgende Stichproben gebildet werden: (Cleve und Lämmel 2020; Petersohn 2005)

- *Zufällige Stichprobe*
  Aus der Gesamtmenge werden rein zufällig Datensätze ausgewählt. (Cleve und Lämmel 2020)
- *Repräsentative Stichprobe*
  Auch hierbei werden aus der Gesamtmenge zufällig Datensätze gewählt (Cleve und Lämmel 2020). Dabei ist jedoch darauf zu achten, dass repräsentative Merkmale in ausreichender Anzahl gewählt werden. Ansonsten besteht die Möglichkeit, dass Merkmale mit gewissen Ausprägungen zu stark unterrepräsentiert sind, oder unter Umständen gar nicht enthalten sind, was schlussendlich zu einer nicht repräsentativen Stichprobe führt. (Petersohn 2005) Die Auswahl der Attribute wird hierbei mithilfe von Häufigkeitsverteilungen getroffen. Des Weiteren sollte jede Attributsausprägung bei nominalen und ordinalen Attributen vertreten sein. (Cleve und Lämmel 2020)
- *Geschichtete Stichprobe*
  Hierbei werden die Datensätze ebenfalls zufällig gewählt. Bei einer geschichteten Stichprobe wird jedoch darauf geachtet, dass Attribute von Bedeutung in angemessener Zahl ausgewählt werden. (Cleve und Lämmel 2020)
- *Inkrementelle Stichprobe*
  Bei diesem Vorgehen wird die Stichprobe inkrementell, also schrittweise erweitert. Es wird zu Beginn mit einer beliebig gebildeten Stichprobe, also beispielsweise einer zufälligen, repräsentativen oder geschichteten Stichprobe, gearbeitet. (Cleve und Lämmel 2020) Diese Stichprobe wird dann schrittweise erweitert. Optimalerweise geschieht dies so lang, bis der Zufallsfehler nicht mehr bedeutend gesenkt werden kann, um den Umfang der Stichprobe nur so groß wie nötig zu halten. Ansonsten würde der Zweck der Stichprobe zur Datenreduktion nicht mehr erfüllt werden. Inkrementelle Stichproben sind aufgrund ihrer schrittweisen Erweiterungen besonders für Analysen geeignet, die iterativ durchgeführt werden. (Petersohn 2005)
- *Average Sampling*
  Bei diesem Vorgehen wird die Gesamtmenge in mehrere Teile aufgeteilt. Auf jedem dieser Teile wird unabhängig von den anderen Teilen eine Analyse durchgeführt. Am Ende werden die Ergebnisse der einzelnen Analysen gemittelt und zu einem Gesamtergebnis zusammengeführt. (Cleve und Lämmel 2020)
- *Selektive Stichprobe*
  Anhand der vom Analysten festgelegten Kriterien wird der Datenbestand nach

relevanten Attributen gefiltert. Nicht aussagekräftige Datensätze werden entfernt und aus den gefilterten Daten wird eine Stichprobe gezogen. (Petersohn 2005; Cleve und Lämmel 2020)

- *Windowing*
  Wie bei der inkrementellen Stichprobe wird auch beim Widowing die Stichprobe nach einer erfolgten Analyse schrittweise erweitert. Dies geschieht jedoch mit dem Unterschied, dass die Stichprobe nicht mit zufälligen, sondern mit relevanten Datensätzen erweitert wird. Dabei werden die Datensätze als relevant angesehen, die beispielsweise vorher zu falsch klassifizierten Objekten geführt haben. (Cleve und Lämmel 2020; Petersohn 2005)
- *Clustergestützte Stichprobe*
  Hierbei werden ähnliche Datensätze in Clustern zusammengefasst. Aus den Clustern werden dann, in beliebiger Anzahl je nach gewünschter Stichprobengröße, Repräsentanten für die anschließende Analyse gewählt. (Cleve und Lämmel 2020)

2.2.4 Datentransformation

Nach den Schritten der Datenselektion und -integration, Datenbereinigung und Datenreduktion folgt als letzter Schritt der vorbereitenden Schritte die Datentransformation. Dabei gibt es Datentransformationen, die unabhängig vom eingesetzten Data-Mining-Verfahren durchgeführt werden können und Transformationen, die erforderlich sind, um ein Verfahren überhaupt erst zu ermöglichen. Nicht jede Datenform ist für jedes Data-Mining-Verfahren geeignet. (Cleve und Lämmel 2020; Petersohn 2005)

Folgende Transformationen sind vom Verfahren unabhängig möglich, können aber trotzdem sinnvoll sein: (Cleve und Lämmel 2020)

- *Kombination oder Separation von Attributen:*
  Unter Umständen kann es sinnvoll sein, Attribute zu kombinieren oder separieren. Häufig wird diese Transformation im Zusammenhang mit Datumsangaben angewandt. Je nach Analyse ist weniger das Datum an sich von Interesse, sondern eher der jeweilige Wochentag, oder sogar nur die Frage, ob es sich um einen Tag in der Arbeitswoche oder am Wochenende handelt. Gleiches kann für Jahreszeiten, welche aus dem Datum entnommen werden können, gelten. (Cleve und Lämmel 2020)
- *Neuberechnung abgeleiteter Attribute:*
  Je nach Aufgabenfeld und Datenbasis kann es sinnvoll sein, aus den gegebenen Attributen neue Attribute abzuleiten (Cleve und Lämmel 2020). So könnte aus Produktionsbeginn und Produktionsende beispielsweise die Produktionsdauer berechnet werden, wenn diese von Interesse sein könnte.
- *Aggregation:*
  Die Aggregation wurde bereits für die Datenreduktion vorgestellt. Jedoch bietet sich eine Aggregation unter Umständen auch als Datentransformation, an um interessante Informationen besser sichtbar zu machen (Petersohn 2005). Beispielsweise könnten Informationen auf einer niedrigen Aggregationsebene zu speziell sein. Ein Beispiel dafür könnten auf Stadteile bezogene Daten eines Versanddienstleisters sein: Sind bei

der Analyse Informationen auf nationaler oder sogar internationaler Ebene gesucht, könnte eine Aggregation auf eine höhere Aggregationsebene wie Städte oder Länder sinnvoll sein.

- *Glättung:*
  Die Glättung als Transformation ist dem Bereinigen der Daten von Rauschen und Ausreißern in Ihrem grundsätzlichen Vorgehen sehr ähnlich. Jedoch besteht das Hauptziel der Glättung hier nicht darin, Rauschen oder Ausreißer zu entfernen, sondern sie zu erkennen. Wurden verrauschte Werte erkannt, können diese anstatt gelöscht auch korrigiert werden. Das ist jedoch nur möglich, wenn die zur Glättung notwendigen Informationen in den nicht verrauschten Daten vorhanden sind. (García et al. 2015)

Um Datenformen in ein für bestimmte Data-Mining-Verfahren passendes Format zu überführen sind teilweise spezielle Transformationen nötig (Cleve und Lämmel 2020). Unter anderem kann es sein, dass der Algorithmus eines Data-Mining-Verfahrens ein bestimmten Datenformat nicht verarbeiten kann, wie zum Beispiel ein Entscheidungsbaum nicht direkt auf stetige Werte angewandt werden kann. Wenn ein Data-Mining-Verfahren grundsätzlich auf ein Datenformat angewendet werden kann, könnte es ohne Transformationen keine aussagekräftigen Ergebnisse liefern. Deshalb werden häufig verschiedene Transformationen angewandt, um die ursprünglichen Werte so umzuwandeln, dass sie von den Algorithmen besser verarbeitet werden können. (García et al. 2015) Diese Transformationen werden im Folgenden genauer erläutert.

Skalierung der Daten

Die Eigenschaften einer messbaren empirischen Variable werden mit einer Skala numerisch repräsentiert. Wichtiges Unterscheidungskriterium von Variablen ist deren Skalenniveau. Es gibt verschiedene Skalenniveaus, die Gemeinsamkeiten, beruhend auf Mengen zulässiger Skalentransformationen, untereinander aufweisen. Unterschieden wird in der Regel zwischen den Skalenniveaus Nominalskala, Ordinalskala, Intervallskala und Verhältnisskala. Die Verhältnisskala kann auch als Ratioskala bezeichnet werden (Völkl und Korb 2018). (Petersohn 2005) Je nach Literatur ist auch von Skalentyp anstelle von Skalenniveau die Rede (Petersohn 2005; Völkl und Korb 2018). Es kann auch noch die Absolutskala als ein fünftes Skalenniveau und besonderer Fall der Verhältnisskala angeführt werden (Völkl und Korb 2018). Die grundsätzlichen Eigenschaften der Skalenniveaus können der Tabelle 2-1 entnommen werden.

Die in Tabelle 2-1 aufgeführten Skalenniveaus sind in der Tabelle hierarchisch geordnet. Der Informationsgehalt nimmt von der Nominalskala zur Absolutskala zu. Dies hat auch zur Folge, dass jedes höhere Skalenniveau auch die Eigenschaften der niedrigeren Skalenniveaus aufweist. (Völkl und Korb 2018)

Tabelle 2-1: Eigenschaften der Skalenniveaus (Völkl und Korb 2018)

Skalen-niveau	**Nominal-skala**	**Ordinal-skala**	**Intervallskala**	**Verhältnis-skala**	**Absolutskala**
Empirische Relation	Unterscheid-barkeit	Rang-ordnung	Linearität/ konstante Abstände	Natürlicher Nullpunkt	Natürliche Einheiten
Term	$A \neq b$, $a = b$	$A < b$, $a > b$	$a < b < c < d$ und damit $b – a = d - c$	$a/b = c/d$	$A = a$ Wert an sich
Beispiel	Geschlecht	Höchster Schul-abschluss	Jahr der Geburt	Einnahmen im Monat (in €)	Anzahl der Geschwister

Die Intervall-, Verhältnis- und Absolutskalen werden unter dem Oberbegriff metrische Skalen oder auch Kardinalskalen zusammengefasst. Nominal- und Ordinalskalen werden als nicht metrische Skalen bezeichnet. Besonders im Kontext der Data Science wird häufig zusätzlich noch zwischen kategorialen und nicht kategorialen Variablen unterschieden: (Völkl und Korb 2018)

- *Kategoriale Variablen:*
  Variablen mit wenigen Merkmalsausprägungen, also vor allem nominale und ordinale Variablen (Völkl und Korb 2018).
- *Nicht Kategoriale Variablen:*
  Variablen mit sehr vielen Merkmalsausprägungen, somit charakteristisch für metrische Variablen (Völkl und Korb 2018).

Mit dem Begriff Skalierung sind diverse Transformationen von Werten auf ein bestimmtes Skalenniveau gemeint. Sie ist dann nötig, wenn die Werte für ein bestimmtes Verfahren in ein adäquates Format transformiert werden müssen. (Cleve und Lämmel 2020)

Eine häufig angewendete Skalierung von Daten für bestimmte Data-Mining-Verfahren ist die Diskretisierung. Sie wird dann angewendet, wenn ein Verfahren eingesetzt werden soll, welches keine numerischen, sondern lediglich kategoriale Werte verarbeiten kann. Eine Diskretisierung transformiert die numerischen in kategoriale Attribute. Dazu gibt es einfache Verfahren, die den Wertebereich eines Attributs in gleich große Intervalle, bzw. Intervalle mit gleicher Häufigkeit enthaltener Attributwerte teilen. Bei komplexeren Verfahren der Diskretisierung wird zusätzlich noch eine mögliche Klassenzugehörigkeit bei der Einteilung der Intervalle vom Diskretisierungsverfahren berücksichtigt, um die Werte der Attribute einer Klasse nach Möglichkeit demselben Intervall zuzuordnen. Dies kann unter Umständen dabei helfen, den Informationsgewinn im Zusammenhang mit der Klassenzugehörigkeit zu maximieren. (Ester und Sander 2000) Nichtsdestotrotz handelt es sich bei der Diskretisierung um eine so genannte Abwärtstransformation, weil durch das Bilden der Intervalle Informationen verloren gehen können (Völkl und Korb 2018).

Eine Umwandlung von nominalen und ordinalen Daten in metrische Daten kann ebenso notwendig sein. Hierbei ist jedoch zu beachten, dass eine korrekte Umwandlung nur von einem höheren auf ein niedrigeres Skalenniveau statistisch korrekt möglich ist. Umgekehrt ist dies nicht

möglich. (Petersohn 2005) Bei einer Umwandlung mehrerer nominal skalierten Merkmale, wie beispielsweise Farben, in codierte numerische Werte würde zwangsläufig eine Rangfolge entstehen. Eine Rangfolge ist jedoch, wie in Tabelle 2-1 zu erkennen ist, keine Eigenschaft nominal skalierter Werte (Völkl und Korb 2018). Eine Umwandlung ordinaler in numerische Attributsausprägungen ist möglich. Ein Beispiel dafür wären ordinale Ausprägungen für Geschwindigkeiten in der Form *langsam*, *mittel*, *schnell* und *sehr schnell*. Möchte man auf diesen Datensatz nun ein Verfahren anwenden, das ausschließlich numerische Werte verarbeiten kann, muss das ordinale Attribut in ein numerisches Attribut transformiert werden. Dazu könnte man die ordinalen Werte codieren: *langsam = 0*, *mittel = 0,33*, *schnell = 0,66* und *sehr schnell = 1*. Dabei ist auch eine andere Codierung möglich. Wichtig ist dabei zu beachten, dass die Rangordnung der Ordinalskala eingehalten wird. Je nach angewendetem Verfahren kann zusätzlich auch der Abstand zwischen den numerischen Codierungen entscheidend sein. (Cleve und Lämmel 2020)

Zusätzlich können für bestimmte Verfahren auch noch folgende Transformationen nötig sein:

- *Anpassung von Datentypen:*
  Wenn ein Verfahren beispielsweise nur mit Ganzzahlen und nicht mit Gleitkommazahlen arbeiten kann, muss der Datentyp angepasst werden. (Cleve und Lämmel 2020)
- *Anpassung von Zeichenketten:*
  Einige Verfahren können nicht mit Umlauten, Sonderzeichen, Groß- und Kleinschreibung oder Ähnlichem umgehen. Sollte dies noch nicht bei der Datenintegration geschehen sein, weil ein gesondertes Verfahren angewendet werden soll, muss dies an dieser Stelle geschehen. (Cleve und Lämmel 2020)

Normierung von Daten

Die Begriffe Normierung und Normalisierung werden in der Literatur teilweise als Synonym für Skalierung verwendet. Normierung oder Normalisierung ist aber eine Form der Skalierung. Da die Normierung im Data-Mining-Prozess eine wichtige Rolle spielen kann, wird sie hier neben anderen Skalierungen separat betrachtet. (Cleve und Lämmel 2020)

Auch nach den vorbereitenden Schritten im Data-Mining-Prozess können nach der Datenbereinigung noch Werte vorliegen, die vor allem eine spezielle Bedeutung in dem Bereich haben, aus dem sie stammen. Zum Beispiel Produktions- oder Maschinendaten. Diese Daten sind so konzipiert, dass sie vor allem mit dem operativen System, in dem sie erfasst worden sind, funktionieren. Besonders die Verteilung der Daten kann dazu führen, dass auch bereinigte Daten noch nicht für ein Verfahren geeignet sind. In diesem Fall können verschiedene Normierungen die Verteilung so umwandeln, dass die Data-Mining-Verfahren besser funktionieren. (García et al. 2015) Im Folgenden werden die häufigsten Normalisierungen erläutert:

- *Min-Max-Normalisierung*:
  Alle numerischen Werte eines numerischen Attributs werden linear auf einen neu definierten Bereich skaliert. Häufig wird hierzu das Intervall zwischen 0 und 1 genutzt, es ist aber auch jedes andere Intervall möglich. Dabei wird der kleinste Wert auf 0 und

der höchste Wert auf 1 skaliert. (García et al. 2015; Cleve und Lämmel 2020) Durchführbar ist diese Normalisierung nur dann, wenn die Minimal- und Maximalwerte der zu normalisierenden Attribute bekannt sind. Die Min-Max-Normalisierung eignet sich besonders für Datensätze, auf die ein distanzbasiertes Verfahren angewendet werden soll. Durch die Skalierung aller Daten auf denselben Wertebereich wird vermieden, dass Attribute mit einer großen Differenz bei der Abstandberechnung gegenüber denen mit einer kleineren Differenz dominieren und Attribute mit einer größeren Differenz mehr Bedeutung gegeben wird. (García et al. 2015)

- *Z-Wert-Normalisierung*:
  Sind die Minimal- oder Maximalwerte von einem Attribut nicht bekannt, oder sind Ausreißer in den Daten vorhanden, die in der Datenbereinigung nicht entfernt werden sollten, ist die Min-Max-Normalisierung nicht durchführbar, beziehungsweise kann durch die Ausreißer verfälscht werden. In diesem Fall bietet sich besonders die Z-Wert-Normalisierung an. (García et al. 2015) Hierbei werden die Daten mithilfe des Mittelwerts und der Standardabweichung transformiert, indem die Differenz aus den Daten und der Mittelwerte durch die Standardabweichung teilt. (Cleve und Lämmel 2020) Bei einer Variation der Z-Wert-Normalisierung kann die mittlere absolute Abweichung anstelle der Standardabweichung verwendet werden. Die mittlere absolute Abweichung hat gegenüber der Standardabweichung den Vorteil, dass sie deutlich robuster gegen Ausreißer ist. (García et al. 2015)
- *Dezimalskalierung*:
  Eine sehr einfache Möglichkeit, die absoluten Werte eines nummerischen Attributs zu reduzieren ist durch das Verschieben des Dezimalkommas (García et al. 2015). Dabei handelt es sich um eine lineare Skalierung mit der eine Transformation in ein vorgegebenes Intervall und somit eine Normierung ermöglicht wird (Cleve und Lämmel 2020).

Hauptkomponentenanalyse

Bei der Hauptkomponentenanalyse, oder kurz PCA (aus dem englischen von Principal Component Analysis), handelt es sich um ein Verfahren aus der Statistik. Dabei wird die Datenstruktur über eine lineare Projektion abgebildet, um die Varianz in der niedrigdimensionalen Projektion zu maximieren. (Runkler 2015)

Durch die Hauptkomponentenanalyse eines Datensatzes wird dieser mithilfe einer linearen Transformation als Verkettung einer Translation und Rotation dargestellt. Dieser Schritt der Transformation wird Hauptkomponententransformation genannt. Dabei handelt es sich um eine inverse Transformation, welche mithilfe einer vom Datensatz abhängigen Rotationsmatrix durchgeführt wird. Um diese Rotationsmatrix zu bestimmen, wird die Varianz der projizierten Daten maximiert. Damit die Varianz maximiert werden kann und die Transformationsmatrix nur eine Rotation und keine Streckung durchführt, muss ein Optimierungsproblem mit Nebenbedingungen gelöst werden. Dazu wird eine Langrange-Funktion mit Nebenbedingungen verwendet. Die Lösung des Optimierungsproblems führt zu einem Eigenwertproblem, welches in ein homogenes Gleichungssystem umgewandelt und gelöst werden kann. Durch Lösung des

Gleichungssystems ergeben sich die Varianzen der projizierten Daten. Mithilfe der Werte der Varianzen kann nun eine Dimensionsreduktion mit maximaler Varianz durchgeführt werden. (Runkler 2015)

Mithilfe der Hauptkomponentenanalyse können hochdimensionale Daten also in zwei Dimensionen abgebildet werden (Runkler 2015). Aus Linearkombinationen bestehender Attribute werden neue Attribute gebildet, welche dann als Hauptkomponenten bezeichnet werden. Die Attribute eines Datensatzes werden mit einem Koeffizienten multipliziert und es wird eine Summe gebildet. Die so entstandenen neuen Attribute werden so zusammengesetzt und sortiert, dass durch die Hauptkomponenten ein Großteil der Streuung erklärt werden kann. Wenn die Streuung erklärt werden kann, konzentriert sich die folgende Analyse auf die ersten Hauptkomponenten und die anderen werden weggelassen. Auf diese Weise wird eine Dimensionsreduktion erzielt. (Aust 2021)

2.3 Data-Mining-Verfahren

Nach Abschluss der Datenvorverarbeitung findet in der Regel eine etwas gründlichere Data Exploration, also eine Erkundung der Daten mithilfe einfacher statistischer Tests statt (Aust 2021; Cleve und Lämmel 2020). Bei numerischen Attributen geben die Werte zu Durchschnitt, Median, Standardabweichung, Maximum und Minimum und bei nominalen und ordinalen Attributen die jeweiligen Häufigkeiten einen schnellen ersten Eindruck (Cleve und Lämmel 2020). Nachdem die Daten vorverarbeitet wurden, sollte erneut die Struktur der Daten überprüft werden. Dabei sind neben statistischen Kennzahlen auch Grafiken über die Verteilung der Variablen interessant und auskunftsreich. Dabei können Histogramme und Punktwolken hilfreich sein. (Aust 2021)

Interessant ist hierbei auch der Korrelationskoeffizient. Dabei handelt es sich um eine Maßzahl, die den linearen Zusammenhang zwischen zwei Variablen misst. Diese Maßzahl liegt zwischen -1 und 1. Liegt der Korrelationskoeffizient bei 1, besteht ein perfekter positiver und bei -1 ein perfekter negativer Zusammenhang, was bedeutet, dass die beiden verglichenen Variablen auf genau einer Linie liegen. (Aust 2021) Eine 0 gibt einen deutlichen Hinweis darauf, dass kein Zusammenhang zwischen den Verteilungen besteht, was durch weitere Untersuchungen abgesichert werden muss. Ähnliches gilt auch für einen perfekten positiven oder negativen Zusammenhang von 1, bzw. -1. Besteht eine Korrelation von beispielsweise 0,19, bedeutet dies nicht unbedingt, dass lediglich ein schwacher Zusammenhang besteht. Die Beurteilung der Stärke oder Schwäche eines Zusammenhangs hängt immer vom jeweiligen Anwendungsgebiet ab. Generelle Aussagen können hierzu nicht getroffen werden. Nichtsdestotrotz ist der Korrelationskoeffizient häufig eine gute Unterstützung, um einen Überblick über Zusammenhänge zu erhalten. Dabei können auch grafische Darstellungen der Korrelationen, beispielsweise mit einer Korrelationsmatrix hilfreich sein. (Stocker und Steinke 2022)

In Abhängigkeit von der vorliegenden Datenlage und des definierten Analyseziels wird nun ein Data-Mining-Verfahren angewendet. Die Verfahren werden dabei sowohl in der Fachliteratur als auch in spezieller Data-Mining-Software nach ihrer Aufgabenstellung strukturiert (Beekmann und Chamoni 2006). In den folgenden Kapiteln wird die Klassifizierungsform von Data-Mining-Verfahren nach ihrer Aufgabenstellung und die Klassifizierungsformen der Verfahren in RapidMiner vorgestellt.

2.3.1 Gängige Klassifizierungsformen nach Aufgabenstellung

Zur erfolgreichen Wissensentdeckung bei Anwendung eines Data-Mining-Prozesses liegt üblicherweise eine präzise Formulierung einer Aufgabenstellung vor. Diese Aufgabenstellung leitet sich aus einer zugrunde liegenden Zielsetzung ab, die in einem Data-Mining-Prozess bereits vor Beginn eines Prozesses formuliert wird. (Beekmann und Chamoni 2006) Fayyad et al. unterscheiden Data-Mining-Verfahren in ihren übergeordneten Zielen Vorhersage und Beschreibung. Dabei machen Sie jedoch auch deutlich, dass es keine klare Grenze zwischen Vorhersage und Beschreibung geben muss. Die Unterscheidung sehen sie jedoch trotzdem als nützlich an, um das allgemeine Ziel der Wissensentdeckung besser verstehen zu können. (Fayyad et al. 1996a)

Um die in der Fachliteratur aktuell gängigen Klassifizierungsformen nach Aufgabenstellung nachvollziehen zu können, werden im Folgenden einige der Häufig in der Praxis angewendeten und in der Literatur aufgeführten Aufgabenstellungen erläutert.

Clusteranalyse

Ziel von Clusteranalysen ist es, Daten automatisch oder halbautomatisch so in Kategorien, Klassen oder Gruppen einzuteilen, dass sich ähnelnde Objekte möglichst im gleichen Cluster, also Teilmengen, befinden (Ester und Sander 2000; Cleve und Lämmel 2020). Objekte aus verschiedenen Clustern sollen sich möglichst unähnlich sein. Vor der Analyse sind noch keine Kategorien, Klassen oder Gruppen bekannt (Beekmann und Chamoni 2006). Um die Einteilung in Cluster zu erreichen, ist vor Beginn einer Clusteranalyse eine geeignete Modellierung der Ähnlichkeit zwischen den Datenobjekten erforderlich. Dabei ist eine möglicherweise unterschiedliche Größe, Form und Dichte, sowie eventuelle hierarchische Verschachtelungen der Cluster zu berücksichtigen. (Ester und Sander 2000)

Die Entdeckung der Cluster wird weitestgehend mithilfe von gut entwickelten Methoden der multivariaten Statistik realisiert (Beekmann und Chamoni 2006). Um die Ähnlichkeit zwischen Datenobjekten zu definieren, kann eine Distanzfunktion modelliert werden, die für Objektpaare definiert ist. Dabei ist die Distanz zwischen Objekten über direkte oder abgeleitete Eigenschaften der Objekte definiert. Auf diese Weise kann die Ähnlichkeit der Objekte quantifiziert werden (Cleve und Lämmel 2020). Die Abstände werden so interpretiert, dass kleine Distanzen gleichbedeutend mit ähnlichen Objekten und größere Distanzen gleichbedeutend mit unähnlichen Objekten sind. Bei einer Distanzfunktion gilt also: je kleiner der Wert, desto ähnlicher sind sich zwei Objekte. Je nach Literatur wird auch eine Ähnlichkeitsfunktion verwendet. Bei einer Ähnlichkeitsfunktion gilt dann umgekehrt: je größer der Wert, desto ähnlich sind sich zwei Objekte. Die Güte einer Clusteranalyse hängt vor allem von der Vollständigkeit und Korrektheit der Distanzfunktion ab. (Ester und Sander 2000) Eine zusätzliche Qualitätsfunktion ist sehr hilfreich, um die verschiedenen möglichen Cluster-Bildungen zu Vergleichen (Cleve und Lämmel 2020). Die Entdeckung von Clustern ist mithilfe geeigneter Methoden einfach möglich. Die Interpretation der gefundenen Cluster hingegen ist oftmals problematisch, da dies mit Expertenwissen des jeweiligen Anwendungsfeldes verbunden ist. Ein typisches Beispiel für ein Anwendungsfeld der Clusteranalyse ist das Finden von Kundengruppen im Marketing für eine individuellere Ansprache der Kunden. (Beekmann und Chamoni 2006)

Vorteil von Distanzfunktionen ist, dass Sie für numerische und kategorische Attributwerte erstellt werden können. Außerdem müssen für Distanzfunktionen nicht unbedingt Funktionsgleichungen gegeben sein. Je nach Anwendung genügt auch eine, besonders für nicht mathematisch und statistisch geschulte Analysten, leicht verständlichere Distanzmatrix. (Ester und Sander 2000)

Bei der Clusteranalyse steht eine Beschreibung des Datenbestands im Vordergrund. Dies soll über die Entdeckung von Wissen über die Ähnlichkeit der Objekte erreicht werden. (Beekmann und Chamoni 2006)

Klassifikation

Klassifikation hat die Aufgabe, Objekte aufgrund ihrer Attributswerte einer vorgegebenen Klasse zuzuordnen. Dabei sind die in einer Datenbank auftretenden Klassen bei einer Klassifikation, im Gegensatz zum Clustering, schon vorher bekannt. Die Zuordnung von Objekten zu Klassen geschieht aufgrund ihrer Attributswerte. Die Klasse angebende Variable wird also durch andere Attributswerte erklärt (Beekmann und Chamoni 2006). Dabei kann die Aufgabe der Klassifikation in zwei Teilaufgaben zerlegt werden: Das Zuordnen von Objekten zu Klassen und das Generieren von Klassifikationswissen. Bei der Zuordnung von Objekten zu Klassen werden Objekte aufgrund von Attributswerten einer Klasse zugeordnet. Dabei geht es nur um das Zuordnen zu den Klassen, nicht aber darum, die Zuordnungen zu verstehen und die Gründe dafür zu erfahren. Das Generieren von Klassifikationswissen hat das Ziel, explizites Wissen über die Generierung der Klassen zu erlangen. Somit handelt es sich also nur bei dieser Teilaufgabe um tatsächliche Wissensgewinnung. (Ester und Sander 2000)

Das grundsätzliche Vorgehen bei der Klassifikation hat die Voraussetzung, dass zu Beginn eine Menge von Objekten gegeben sein muss, also die Teilmenge einer Gesamtmenge, von denen sowohl die für die Klassifikation relevanten Attributswerte als auch die Klasse selbst bekannt sind. Die Attributswerte können sowohl numerisch als auch kategorisch sein. Mithilfe eines eigenen Attributs, einer die Klasse angebenden Variablen, wird die Klassenzugehörigkeit der Objekte ausgezeichnet (Beekmann und Chamoni 2006; Ester und Sander 2000). Die Anzahl der Attributsausprägungen dieses Attributs sind häufig klein. Wenn von einer Teilmenge einer Gesamtmenge die Klassenzugehörigkeit bekannt ist, erfolgt die Einteilung der unbekannten Objekte in die bekannten Klassen mithilfe von Klassifikatoren. Ein Klassifikator ist dabei eine Funktion, die mithilfe von Regeln Objekte den jeweiligen Klassen zuweist. (Ester und Sander 2000)

Je nach angewendetem Verfahren und den gewählten Parametern können aus einer gegebenen Trainingsmenge verschiedene Klassifikatoren erlernt werden. Aus diesem Grund ist es wichtig, Klassifikatoren bewerten zu können. Dies kann geschehen, indem die Leistungsfähigkeit der Klassifikatoren miteinander verglichen wird. Wichtiges Leistungsmaß ist dabei der Klassifikationsfehler, also der Anteil der Objekte, die falsch klassifiziert werden. Bei der Ermittlung des Klassifikationsfehlers ist es wichtig, die erlernten Klassifikatoren auf einer Trainingsmenge zu trainieren und den Klassifikationsfehler über eine separate Testmenge zu schätzen. Wird der Klassifikator auf der gleichen Menge getestet, auf der er trainiert wird, kommt es mit hoher Wahrscheinlichkeit dazu, dass der Klassifikator zu sehr auf die Trainingsdaten optimiert wird. Auf den Testdaten würde ein solcher Klassifikator nicht gut funktionieren, da

dieser die Trainingsdaten auswendig gelernt hätte (Cleve und Lämmel 2020). Dieser Effekt wird als Overfitting bezeichnet (Cleve und Lämmel 2020; Ester und Sander 2000). Es sollte also nach Möglichkeit eine Trainingsmenge zum Lernen eines Klassifikators und eine Testmenge zum Schätzen des Klassifikationsfehlers verwendet werden. Ist die Anzahl der Objekte, von denen die Klassen bekannt sind, klein, kann dieses Verfahren jedoch nicht verwendet werden. Dann sollten alle Objekte sowohl zum Training als auch zum Testen verwendet werden. In einem solchen Fall kann dann eine *m*-fache Kreuzvalidierung (Cross-Validation) Abhilfe schaffen. Diese teilt die Gesamtmenge in *m* gleich große Teilmengen. Von diesen Teilmengen verwendet man jeweils *m*-1 Teilmengen zum Training und die verbleibende Teilmenge zum Testen. Die erhaltenen m-Klassifikationsfehler werden anschließend kombiniert. (Ester und Sander 2000)

Typisches Beispiel für eine Klassifikation ist eine Kreditwürdigkeitsprüfung (Beekmann und Chamoni 2006; Cleve und Lämmel 2020). Dabei wird ein Klassifikator aus einem bekannten Datenbestand generiert, um einen Neukunden, der einen Kredit beantragt, anhand seines Regelsystems als kreditwürdig oder als kreditunwürdig einzustufen. Ein weiteres typisches Beispiel ist die Einteilung in eine Risikoklasse einer Versicherung (Beekmann und Chamoni 2006; Ester und Sander 2000).

Numerische Vorhersage

Ziel einer numerischen Vorhersage ist die Prognose eines beliebigen numerischen Wertes. Dies wird durch die Approximation einer Funktion mithilfe von Beispieldaten ermöglicht. Die Grundlage dafür bilden Trainingsdaten, welche sowohl aus Datensätzen als auch aus den zugehörigen Funktionswerten bestehen. Auf Basis dieser Trainingsdaten können dann die Werte zukünftiger Datensätze berechnet werden. Dazu wird mithilfe der bekannten Daten eine Funktion berechnet, die den tatsächlichen Verlauf der Daten approximiert. Diese Funktion kann dann für neue Datensätze einen Vorhersagewert berechnen. Bei numerischen Vorhersagen können so Werte berechnet werden, die aus einem großen numerischen Wertebereich stammen. (Cleve und Lämmel 2020)

Häufiges Problem von numerischen Vorhersagen ist die Modellierung des Zusammenhangs von abhängigen und unabhängigen Variablen. Zeitreihen bestehen normalerweise aus Beobachtungswerten, die zu Zeitpunkten mit demselben Abstand zueinander, also zu gleichbleibenden Intervallen, erhoben wurden. Ein Modell einer solchen Zeitreihe kann dann als Regel verstanden werden, welche den Prozess der Erzeugung der Beobachtungswerte beschreibt. Ein solches Modell zur Prognose der Beobachtungswerte hat jedoch die Voraussetzung, dass der die Beobachtungswerte erzeugende Prozess sich nicht verändert. Diese Voraussetzung wird auch als Stationarität bezeichnet. (Petersohn 2005) Stationarität beschreibt also, stark vereinfacht gesagt, die Voraussetzung, dass Eigenschaften des Beobachtungswerte erzeugenden Prozesses auch zu zukünftigen Zeitpunkten gelten. Diese Annahme wird häufig einfach vorausgesetzt. (Kreiß und Neuhaus 2006) Eine ausführliche Anleitung über den korrekten Nachweis von Stationarität ist in Kreiß und Neuhaus 2006 auf S. 17 ff. zu finden.

Ein Beispiel für eine numerische Vorhersage ist ein Regressionsmodell, welches das Absatzvolumen der Automobilindustrie vorhersagen kann. Die Variable PKW-Neuzulassungen kann dabei durch andere quantitative Größen geschätzt werden. Ein weiteres Beispiel wäre die

Schätzung von Wahrscheinlichkeiten mithilfe eines logistischen Regressionsmodells, bei dem die abhängige Variable nur Werte zwischen 0 und 1 annehmen kann. (Beekmann und Chamoni 2006)

Entdecken von Abhängigkeiten

Aufgabe hierbei ist es, Abhängigkeiten zwischen Merkmalen oder einzelnen Merkmalsausprägungen eines Datenbestandes, beziehungsweise einer Teilmenge eines Datenbestandes zu entdecken. Dabei gibt es a priori keine Annahmen zu Abhängigkeiten. Diese sollen automatisch von Verfahren aufgefunden werden. Die erkannten Abhängigkeiten werden nach dem Auffinden jedoch lediglich angezeigt und nicht kausal erklärt. Deshalb muss einzeln geprüft werden, ob auch tatsächlich Abhängigkeiten vorliegen. Sollten tatsächlich Abhängigkeiten gefunden worden sein, werden diese von den Verfahren vor allem beschreibend dargestellt. Daraus können jedoch im weiteren Verlauf einer Analyse Prognosemodelle erstellt werden. (Beekmann und Chamoni 2006) Dazu werden die Daten zuerst analysiert mit dem Ziel Regelmäßigkeiten zu entdecken, mit dem das Verhalten neuer Datensätze vorhergesagt werden soll. (Cleve und Lämmel 2020)

Eine häufig genutzte Möglichkeit zum Entdecken von Abhängigkeiten ist das Auffinden von Assoziationsregeln zwischen einzelnen Attributwerten in einem Datensatz mit kategorischen Attributen. Solche Regeln beschreiben beispielsweise Beziehungen zwischen Artikeln in einem Kaufhaus in einer Warenkorbanalyse, oder Eigenschaften von Käufern Bestimmter Produkte. (Beekmann und Chamoni 2006)

Nachdem die Aufgabenstellungen vorgestellt wurden, anhand dessen Data-Mining-Verfahren in der Literatur häufig strukturiert werden, werden im folgenden Kapitel die Klassifizierungsformen in RapidMiner vorgestellt.

2.3.2 Klassifizierungsformen in RapidMiner

Die Data-Mining-Software RapidMiner wurde ursprünglich an der TU Dortmund entwickelt, verfügt über eine umfangreiche Auswahl an Data-Mining-Verfahren und über die für den Prozess erforderlichen Operatoren. Die grafische Benutzeroberfläche ermöglicht die Anwendung von Data-Mining-Verfahren ohne Programmieraufwand sowie eine Veranschaulichung des gesamten Prozesses durch die grafische Verknüpfung verschiedener Operatoren. Dabei erlaubt die beliebige Verknüpfung der Operatoren, ähnlich wie bei Programmiersprachen, die Verwendung von Konzepten wie Schleifen, Bedingungen und andere Anwendungsschemata (Mierswa et al. 2006). Da RapidMiner in dieser Arbeit verwendet wird, werden im Folgenden kurz die Benutzeroberfläche und die wichtigsten Werkzeuge vorgestellt.

In die Benutzeroberfläche sind die vier Panels *Repository*, *Operators*, *Process*, *Parameters* und *Help* integriert. Über den Panel *Repository* können die Daten und Prozesse für einen schnellen Zugriff hinterlegt werden und per Drag-and-drop geöffnet werden. Das Panel *Operator* ermöglicht den Zugriff auf alle Operatoren in RapidMiner. Die Operatoren sind in die Gruppen Data Access, Blending, Cleansing, Modelling, Scoring, Validation, Utility, Extensions und Deployment gegliedert. Über ein Suchfeld wird die Suche nach einem gewünschten Operator vereinfacht. Eine Ansicht und Bearbeitungsmöglichkeit der Parameter sind über das Panel *Parameter* gegeben. Hier werden die Parameter des Operators angezeigt, der im Prozess

angewählt ist. Das Panel *Help* zeigt eine kurze Beschreibung, detaillierte Erklärung und einen Beispielprozess zum ausgewählten Parameter an. Die einzelnen Schritte des Analyseprozesses sind über das Panel *Process* zu sehen. Hier können neue Operatoren hinzugefügt, entfernt oder die Verknüpfungen verändert werden. Es ist auch möglich, einen Unterprozess (*Subprocess*) zu bilden. Die Panels können beliebig angeordnet werden. (RapidMiner 2014)

Neben der Strukturierung von Data-Mining-Verfahren in der Literatur ist auch die Strukturierung in der für diese Arbeit verwendeten Data-Mining-Software RapidMiner von Interesse. Da keine Literatur über die Strukturierung der Operatoren in RapidMiner (Abbildung 2-4) gefunden werden kann, wird diese im Folgenden in eigenen Worten beschrieben.

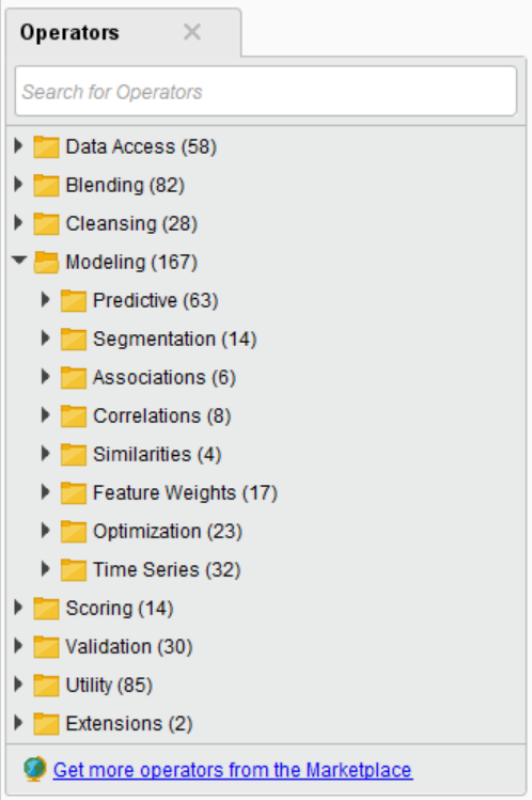


Abbildung 2-4: Screenshot des Panels *Operators* in RapidMiner

Die Data-Mining-Verfahren werden in RapidMiner in dem Panel *Operators* als *Model* bezeichnet (RapidMiner 2014). In dem Unterordner *Modeling* sind die Verfahren und verfahrensabhängige Operatoren in acht Kategorien eingeteilt. Diese Kategorien werden im Folgenden kurz erläutert, um die Strukturierung in RapidMiner nachvollziehen zu können.

Predictive: In dem Unterordner *Predictive* werden insgesamt 63 Operatoren zur Vorhersage zusammengefasst. Innerhalb des Unterordners befinden sich weitere Unterordner. Diese weiteren Unterordner scheinen vor allem für eine bessere Übersicht gewählt worden zu sein. Der Fokus der Strukturierung liegt hier jedoch nicht auf der zu Grunde liegenden Aufgabenstellung, sondern auf der Art des Verfahrens selbst. So wird beispielsweise zwischen *Trees*, *Neural Nets* und *Functions*, also damit, wie die Verfahren aufgebaut sind unterschieden.

Neben Operatoren zu Data-Mining-Verfahren sind hier auch Operatoren zum Gruppieren und Updaten, beispielsweise zum Updaten oder Gruppieren von bereinigten und unbereinigten

Daten, sowie Operatoren zur Formelgenerierung, beispielsweise aus einem Regressionsmodell, zu finden.

Segmantation: Der Unterordner *Segmantation* beinhaltet insgesamt 14 Operatoren ohne einen weiteren Unterordner. Hierbei handelt es sich bei allen Operatoren um Verfahren und Visualisierungsoperatoren aus der Aufgabenstellung des Clustering. Alle Operatoren eignen sich lediglich zur Beschreibung, nicht aber zur Vorhersage.

Associations: Im Unterordner *Association* sind sechs Operatoren zu finden. Alle Operatoren stammen aus der Aufgabenstellung des Entdeckens von Abhängigkeiten, oder genauer zum Auffinden von Assoziationsregeln. Die Operatoren eigenen sich sowohl zur Beschreibung der Zusammenhänge von Attributen, indem Assoziationsregeln generiert werden, als auch zur Vorhersage, indem die generierten Assoziationsregeln zur Klassifikation verwendet werden.

Correlations: Der Unterordner *Correlations* verfügt über 8 Operatoren. Hierbei handelt es sich lediglich um Operatoren zur Visualisierung und zur Auswahl von Attributen im Zusammenhang mit Korrelationen und nicht um Data-Mining-Verfahren. Wie bereits erklärt, sind Korrelationen jedoch hilfreich für die Auswahl nützlicher Attribute bei Data-Mining-Verfahren sowie zur Beschreibung von Daten.

Similarities: Mit dem Unterordner *Similarities* werden vier Operatoren zusammengefasst. Bei den vier Operatoren handelt es sich um Verfahren zur Ermittlung und Visualisierung von Ähnlichkeitsmaßen, die alle der Aufgabenstellung der Clusteranalyse zugeordnet werden können. Die Operatoren eigenen sich zur Beschreibung von Daten, nicht aber zur Vorhersage.

Feature Weights: Im Unterordner *Feature Weights* sind 17 Operatoren aufgeführt. Alle Operatoren eignen sich zur Gewichtung von Attributen. Zur Gewichtung stehen dabei verschiedene Verfahren zur Auswahl wie beispielsweise über Korrelationen oder die Fehler bei Klassifikationen. Die Operatoren eignen sich zur verfahrensabhängigen Vorbereitung der Daten.

Optimization: Der Unterordner *Optimization* enthält insgesamt 23 Operatoren. Die Operatoren können zur verfahrensabhängigen Optimierung von Data-Mining-Verfahren verwendet werden. Dies wird beispielsweise über automatisierte Optimierungen von Parametereinstellungen oder Verfahren zur automatisierten Attributsauswahl ermöglicht.

Time Series: Im Unterordner *Time Series* sind insgesamt 32 Operatoren zur Zeitreihenanalyse zusammengefasst. Diese Operatoren sind in weitere Unterordner unterteilt, die unter anderem Operatoren zu vorbereitenden Schritten wie der Reduktion durch Attributsauswahl oder Stichprobenziehung und Transformation wie der Normierung, aber auch der Prognose und Validierung enthalten. Die Verfahren der Zeitreihenanalyse können der numerischen Vorhersage zugeordnet werden (Petersohn 2005).

Die Strukturierung der Data-Mining-Verfahren und der verfahrensabhängigen Operatoren wird in RapidMiner also nicht an der Aufgabenstellung orientiert. Der Gedanke von Fayyad et al. 1996a, die Verfahren in ihre übergeordneten Ziele Vorhersage und Beschreibung strukturieren zu können, scheint hier jedoch teilweise umgesetzt worden zu sein. Fast alle Verfahren und verfahrensabhängigen Vorverarbeitungsschritte sind unter dem Ordner *Predictive* zusammengefasst. Bis auf den Unterordner *Time Series* handelt es sich bei den in den anderen Unterordnern zusammengefassten Verfahren und Operatoren zur Vorverarbeitung um

Werkzeuge, die zur Beschreibung der Daten dienen. Der Unterordner *Time Series* enthält zwar ausschließlich Operatoren zur Zeitreihenanalyse, diese sind jedoch in weitere Unterordner unterteilt, welche unter anderem Operatoren zur Prognose enthalten. Des Weiteren ist festzuhalten, dass nicht nur nach den Data-Mining-Verfahren und ihrer Struktur, sondern ebenfalls mit den dazu anwendbaren verfahrensabhängigen Operatoren zur Vorverarbeitung der Daten strukturiert wird.

2.4 Herausforderungen bei Anwendung von Data-Mining-Verfahren

In diesem Unterkapitel werden wichtige Herausforderungen bei der Anwendung von Data-Mining-Verfahren erläutert. Unter Berücksichtigung dieser Herausforderungen werden die Anforderungen an das Klassifizierungskonzept präzisiert.

2.4.1 Umgang mit Big Data

In Kapitel 1 wird bereits beschrieben, in welchem rasanten Tempo heute von Unternehmen und Institutionen Daten gesammelt werden. Durch die fortschreitende technologische Entwicklung und den zunehmenden weltweiten Einsatz digitaler Geräte werden auch von Privatpersonen immer mehr digitale Datenströme generiert. Diese Faktoren führen zu dem, was heute auch als Big Data bezeichnet wird. (Che et al. 2013) Für den Ausdruck Big Data gibt es keine eindeutige Definition, häufig ist in der Literatur aber von den drei V's *volume*, *velocity*, *variety* oder in einer erweiterten Definition von fünf V's mit *value* und *validity* die Rede (Bachmann 2014; Gandomi und Haider 2015; Gartner, Inc. 2022).

Volume steht für das große Datenvolumen von Big Data. Durch die Integration immer weiterer Datenquellen, vor allen Dingen die Integration von Daten aus dem Internet, werden die weltweiten Datenmengen nicht mehr in Giga- und Terabyte beschrieben (Der Begriff Tera definiert eine Zahl mit zwölf Nullen) sondern in Zetta- und Yottabytes (Yotta definiert eine Zahl mit 24 Nullen). (Bachmann 2014)

Velocity steht für die steigende Verarbeitungsgeschwindigkeit der Datenmengen. Dabei spielt aktuell vor allem die InMemory-Technologie eine wichtige Rolle. Mithilfe dieser Technologie werden, vereinfacht und kurz gesagt, Daten nicht mehr durch sequenzielle Schritte vom Festplattenspeicher prozessiert, sondern in Echtzeit im Hauptspeicher verfügbar gemacht. Dadurch können auch große Datenmengen in Echtzeit analysiert werden. (Bachmann 2014)

Mit *variety* wird die Vielfalt der Datenstruktur beschrieben. Daten werden nicht mehr nur noch in den klassischen Strukturen relationaler Datenbanksysteme mit streng definierten Kriterien zur Ordnung gespeichert. Besonders durch das Internet als Quelle der Daten liegen im Kontext der Big Data häufig unstrukturierte Datenmengen vor. (Bachmann 2014)

Value beschreibt den unternehmerischen Mehrwert der Daten. Damit ist gemeint, dass die Analyse der Datenmengen einen unternehmerischen Mehrwert haben muss. Nicht für jedes Unternehmen muss ein unternehmerischer Mehrwert daraus entstehen, sich mit Big Data auseinanderzusetzen. (Bachmann 2014)

Validity steht für die Notwendigkeit, sich mit der Validität, also der Gültigkeit der Daten auseinandersetzen zu müssen. Dabei ist sicherzustellen, dass eine hohe Datenqualität aufrechterhalten werden kann, um entscheidungsrelevantes Wissen generieren zu können. (Bachmann 2014)

Data-Mining-Verfahren wurden lange Zeit dazu eingesetzt, um unbekannte Muster aus homogenen und aus heutiger Sicht kleinen Datensätzen zu entdecken. Durch die vielen neuen Datenquellen entstehen jedoch größere und heterogene Datenmengen. Dies wird insbesondere mit den v's *variety* und *volume* deutlich. Auf solche Daten Data-Mining-Verfahren anzuwenden, bringt auch neue Herausforderungen mit sich. Eine solche Herausforderung ist, strukturierte, halbstrukturierte oder sogar unstrukturierte Daten gleichzeitig zu verarbeiten. Strukturierte Daten können nahezu problemlos in bekannten Datenbanksystemen strukturiert werden. Halbstrukturierte Daten können unter Umständen auch in solche Datenbanksysteme integriert werden, unstrukturierte Daten jedoch definitiv nicht. Solche Daten werden für gewöhnlich in Dateien gespeichert. Hinzu kommt noch die *velocity*, also die steigenden Verarbeitungsgeschwindigkeiten, mit denen die Daten verarbeitet werden sollen. Dabei stoßen aktuelle Datenbank-Managementsysteme (DBMS) an ihre Grenzen. Um diesen Herausforderungen zu begegnen, müssen gezielt neue und angepasste DBMS eingesetzt werden. (Che et al. 2013)

Das stetig steigende Datenvolumen bringt auch noch eine weitere Herausforderung: Datenmüll (Che et al. 2013). Datensätze haben somit ein steigendes Volumen und enthalten irrelevante Daten. Es ist jedoch effizienter und führt zu besseren Ergebnissen, nur mit den relevanten Daten und nicht mit Rohdaten zu arbeiten. (Muhammad Habib et al. 2016) Die Gründe dafür werden in Unterkapitel 2.2.3 genauer erläutert.

2.4.2 Datenkompetenz

Der Begriff Datenkompetenz oder auch Data Literacy beschreibt die Fähigkeit eines Menschen, Daten zu sammeln, managen, bewerten und anzuwenden. Dies soll auf kritische Art und Weise erfolgen. Innerhalb der Wissenschaft und Industrie ist eine grundlegende Datenkompetenz mittlerweile unerlässlich. (Ludwig und Thiemann 2020)

Zur heutigen Zeit spielen Daten schon eine wichtige Rolle und werden dies auch in den kommenden Jahren immer mehr tun. Dabei sorgt der technische Fortschritt für ein exponentielles Wachstum an Leistung und Geschwindigkeit von IT-Systemen. Zusätzlich steigt die Anzahl der Datenproduzenten wie beispielsweise Messgeräte, Sensoren und Kameras. Dabei wird von Nutzern der Daten erwartet jederzeit auf diese Daten zugreifen zu können. (Ludwig und Thiemann 2020)

Um die steigenden Mengen an Daten beherrschen zu können, ist Datenkompetenz wichtig und wird zu einer Schlüsselkompetenz des aktuellen Jahrhunderts. Im Bericht *Future Skills* von Schüller et al. (2019) des Hochschulforums Digitalisierung werden fünf Kompetenzbereiche der Data Literacy beschrieben. Diese orientieren sich am Lebenszyklus der Daten, also von der Erzeugung bis zur Nutzung dieser. (Ludwig und Thiemann 2020; Schüller et al. 2019)

Im Folgenden sollen die Kompetenzbereiche kurz erläutert werden:

Konzeptioneller Rahmen: Es soll Wissen über und Verständnis für Daten aufgebaut werden, um so die Nutzung und Anwendung dieser verstehen zu können. (Ludwig und Thiemann 2020)

Datensammlung: Daten aus verschiedenen Quellen wie Sensoren, Messgeräten oder Ergebnisse aus Simulationen sollen kritisch bewertet werden bezüglich ihrer Zuverlässigkeit und Qualität der Daten. (Ludwig und Thiemann 2020)

Datenmanagement: Eine intensive Auseinandersetzung mit der Qualität der Daten und daraus folgende Beseitigung von Anomalien und Ausreißern ist unerlässlich. Unter Umständen müssen Datenformate konsolidiert oder Daten konvertiert werden. Wichtig ist hierbei die Annotierung der Daten mit Metainformationen, um diese später weiterverarbeiten zu können. Zusätzlich beschäftigt man sich hier auch mit der Speicherung und möglicherweise Langzeitarchivierung der Daten. (Ludwig und Thiemann 2020)

Datenevaluation: Dieser Kompetenzbereich befasst sich mit der numerischen und grafischen Auswertung von Daten mithilfe geeigneter Methoden und Werkzeuge. Dabei werden Daten interpretiert und präsentiert und während Entscheidungsfindungsprozessen verwertet. (Ludwig und Thiemann 2020)

Datenanwendung: In diesen Kompetenzbereich fällt die Auseinandersetzung mit Datenethik, Datenzitierung, Datenverteilung und der Evaluierung von datenbasierten Entscheidungen. (Ludwig und Thiemann 2020)

In der Wissenschaft und Industrie müssen sich Mitarbeitende regelmäßig mit Verfahren zur Datensammlung, zum Datenmanagement, zur Datenevaluierung und zur Datenanwendung beschäftigen und auseinandersetzen. Dabei sollten sie diese Vorgänge beurteilen und bewerten können und bei Gestaltung solcher Verfahren und Vorgänge detaillierte Kenntnisse in diesen Kompetenzbereichen aufweisen. (Ludwig und Thiemann 2020) Sie sollten die Informationen herausfiltern können, die sie in ihren Analysen weiterbringen. Dazu müssen Mitarbeitende verstehen, welche Daten ihnen nützen und welche lediglich Ressourcen verbrauchen. Sie müssen also über die Fähigkeit verfügen, Daten zu filtern. Hierbei werden aus objektiv neutralen Daten subjektiv nützlichere Informationen. Diese Daten müssen aber zusätzlich auch analysiert werden können. In welcher Form diese analysiert werden, hängt vom Kontext ab, in und aus welchem die Daten verwertet und verwendet werden sollen. (Lexa 2021)

Bei den heutigen, kaum zu überschauenden Mengen an Daten muss kein Mitarbeiter dazu in der Lage sein, diese einzeln zu sichten und zu analysieren. Sie sollten jedoch dazu in der Lage sein, geeignete Methoden dazu zu finden, mithilfe von Software, Schlüsse aus den Daten ziehen zu können. (Lexa 2021) Über welche Fähigkeiten Menschen beim Umgang mit Daten laut aktueller Forschung grundsätzlich verfügen sollten, wurde über die fünf Kompetenzbereiche der Data Literacy beschrieben (Wolff et al. 2016).

Insgesamt ist festzuhalten, dass die Ausbildung einer umfassenden Datenkompetenz eine dringende Herausforderung der aktuellen Zeit ist. Wie in diesem Kapitel dargestellt, handelt es sich dabei um eine Schlüsselkompetenz für die Aufgaben des 21. Jahrhunderts. Nicht nur in der Wissenschaft, sondern auch in der Wirtschaft. (Ludwig und Thiemann 2020) Diese Kompetenz gilt es nach Schüller et al. systematisch in die Bildung von Hochschulen zu integrieren. Dabei soll neben dem Wissen und der Fähigkeit jenes Wissen anzuwenden besonders die Bereitschaft dies auch zu tun gelehrt werden. (Schüller et al. 2019)

2.4.3 Evaluation und Interpretation der Ergebnisse

Werden Data-Mining-Verfahren erfolgreich auf eine Datenmenge angewendet, kann bisher unentdecktes Wissen erkannt und extrahiert werden. Ob dieses Wissen jedoch auch einen Mehrwert bringt und somit überhaupt von Interesse ist, muss zunächst geprüft werden. (Geng und Hamilton 2006)

Die Evaluationsmethoden für Data-Mining-Prozesse in der Literatur befassen sich meist mit spezifischen Techniken zum Testen der Data-Mining-Ergebnisse. Ein Großteil davon verwendet dazu lediglich ein Teil der ursprünglichen Datenmenge wie die Testdaten nach einem Trainings- und Testsplit. Die anderen Phasen des Data-Mining-Prozesses werden dabei nicht berücksichtigt. (Scheidler und Rabe 2021)

Eine gängige Möglichkeit, Muster wie Klassifizierungsregeln, Assoziationsregeln oder aus Clustern abgeleitete Regeln mithilfe spezifischen Techniken formal zu bewerten sind Interessantheitsmaße (Geng und Hamilton 2006). Beispiele für Interessantheitsmaße einer Regel $A \rightarrow B$ sind beispielsweise Konfidenz und Support: (Cleve und Lämmel 2020)

- *Konfidenz*:
  Mit diesem Interessantheitsmaß soll die Sicherheit einer Regel gemessen werden. Sie prüft, mit welcher Wahrscheinlichkeit man auf *B* schließen kann, wenn *A* erfüllt ist. (Cleve und Lämmel 2020)
- *Support*:
  Dieses Interessantheitsmaß soll Auskunft über die Allgemeingültigkeit einer Regel in der Form geben, dass sie misst, wie oft *A* und *B* gemeinsam auf der gesamten Testmenge vorkommen. (Cleve und Lämmel 2020)

Um die Güte einer Klassifikation oder numerischen Vorhersage zu messen, kann der Fehler berechnet werden. Dazu wird der erwartete Wert mit dem prognostizierten Wert verglichen. Umgekehrt kann auch die Erfolgsrate berechnet werden. (Cleve und Lämmel 2020)

Generell sind folgende vier Fälle bei einer Klassifikation möglich: (Runkler 2015)

- *True positive*: Anzahl der richtig positiv klassifizierten Datensätze (Runkler 2015)
- *True negative*: Anzahl der richtig negativ klassifizierten Datensätze (Runkler 2015)
- *False positive*: Anzahl der negativen Datensätze, die als positiv klassifiziert werden (Runkler 2015)
- *False negative*: Anzahl der positiven Datensätze, die als negativ klassifiziert werden (Runkler 2015)

Aus diesen vier Fällen können und sollten, je nach Anwendungsfall, weitere Kenngrößen wie beispielsweise die Relevanz (*Relevanz = true positive + false negative*) oder die Korrektheitsrate ($Korrektheitsrate = \frac{\text{true positive+true negative=Korrekte Klassifikationen}}{\text{true negative+flase negative = Negativität}}$) berechnet werden. (Runkler 2015; Cleve und Lämmel 2020)

Bei numerischen Vorhersagen können Maße für die Sicherheit oder den Nutzen zur Evaluation herangezogen werden. Die Sicherheit einer Vorhersage kann beispielsweise über eine geschätzte Vorhersagegenauigkeit bei Anwendung des generierten Modells auf neuen Daten gemessen werden. Maße über den Nutzen können beispielsweise über eingesparte Ausgaben

aufgrund eines Vorhersagemodells oder der Beschleunigung von Reaktionen auf Ereignisse gemessen und so verglichen und evaluiert werden. (Fayyad et al. 1996b)

Insgesamt können also grundsätzlich quantitative Maße für die Bewertung der extrahierten Muster definiert werden. Die genannten Beispiele dafür sind die objektiv messbaren Kenngrößen. Die, wie in Kapitel 2.1 beschriebene, von entdeckten Mustern geforderte Neuartigkeit und Verständlichkeit ist jedoch deutlich subjektiver. (Fayyad et al. 1996b)

Um das vermeintlich generierte Wissen zu bewerten, wird meist eines der folgenden Verfahren gewählt: Manuelle Analyse durch Experten oder eine formale Bewertung mithilfe statistischer Tests. Bei der manuellen Analyse durch Experten wird die Nützlichkeit und Interessantheit der Ergebnisse mithilfe von Erfahrungswerten, Branchenwissen und vorher festgelegten Projektzielen beurteilt. Beim Verfahren der formalen Bewertung werden statistische Tests wie eine Kreuzvalidierung auf die Ergebnisse angewendet. Solche Bewertungsverfahren umfassen Messgrößen, die vor allem von den angewendeten Algorithmen und den vordefinierten Zielen abhängen. Deshalb müssen die Ergebnisse nach der statistischen Überprüfung trotzdem noch von Experten überprüft werden. Die statistische Überprüfung dient also hauptsächlich dazu, schon vor der manuellen Analyse Muster ohne Relevanz auszuschließen. (Kurgan und Musilek 2006)

3 Konzept zur Systematisierung von Data-Mining-Verfahren in Klassifikatoren

In diesem Kapitel wird ein Konzept entwickelt, mit dem Data-Mining-Verfahren anhand von Klassifikatoren systematisiert werden können. In Unterkapitel 2.3 werden die gängige Vorgehensweise zur Strukturierung von Data-Mining-Verfahren nach ihrer Aufgabenstellung vorgestellt. Dabei wird deutlich, dass Prozessen, in denen Data-Mining-Verfahren angewendet werden, eine Aufgabenstellung zugrunde liegt. Die Aufgabenstellung leitet sich wiederum aus einer zugrunde liegenden Zielsetzung ab, welche vor Beginn eines solchen Prozesses formuliert wird. Mit Hilfe einer strukturierten Betrachtung des Datenbestandes sollen schon vor Beginn einer Analyse Aufgabenstellungen ausgeschlossen werden. Das soll durch Klassifikatoren ermöglicht werden.

Nach Betrachtung der beiden gängigsten Vorgehensmodell für Data-Mining-Prozesse ist festzuhalten, dass das CRISP-DM Vorgehensmodell einen insgesamt zyklischeren Charakter als das Vorgehensmodell nach Fayyad hat. Bei einem Blick auf Abbildung 2-1 und Abbildung 2-2 wird dies durch die kreisförmige Darstellung deutlich. Im Vorgehensmodell nach Fayyad werden nach jeder Phase Ergebnisse dargestellt, was die Ableitung von Ergebnissen nach jeder Phase erfordert. Die Darstellung ohne Zwischenergebnisse beim CRISP-DM Vorgehensmodell stellt den Data-Mining-Prozess als Ganzes, weniger die einzelnen Phasen in den Vordergrund.

Da die Systematisierung der Data-Mining-Verfahren in Klassifikatoren ohne vorher festgelegte Zielsetzung ermöglicht werden soll, ist das CRISP-DM Vorgehensmodell das geeignetere Vorgehensmodell für diese Arbeit. Neben einer Aufgabenstellung soll auch der Datenbestand als Grundlage für die Auswahl eines geeigneten Data-Mining-Verfahrens dienen und der Prozess iterativ durchlaufen werden. So wie in Abbildung 2-2 beim CRISP-DM Vorgehensmodell die Daten im Mittelpunkt des Data-Mining-Prozesses stehen, wird auch bei einer Systematisierung nach Klassifikatoren die Datenbasis als zusätzlicher Ausgangspunkt der Erarbeitung dienen.

In Unterkapitel 2.1 wird die Wichtigkeit der datenvorverarbeitenden Schritte für den gesamten Prozess deutlich. Der zeitlich hohe Aufwand der vorbereitenden Schritte im Verhältnis zum gesamten Data-Mining-Prozess wird in Unterkapitel 2.2 dargestellt. Nach Klassifikatoren systematisierte Data-Mining-Verfahren sind in dieser Phase eine Möglichkeit, Teile eines Datensatzes von Beginn an auszuschließen, da diese teilweise von den für die Analyse relevanten Verfahren gar nicht verarbeitet werden können. Dies wird in Unterkapitel 2.3.1 deutlich. Durch das Ausschließen von Teilen eines Datensatzes zu Beginn eines Prozesses werden Analysten bei der Datenvorverarbeitung unterstützt und der zeitliche Aufwand für diese Phase reduziert.

3.1 Beschaffenheit der Datengrundlage

In diesem Kapitel wird die Beschaffenheit der Datengrundlage, auf die die in dieser Arbeit entwickelten Klassifikatoren angewendet werden, genauer erläutert. Als Quelle der Daten dient die öffentlich zugängliche Plattform kaggle.com.

3.1.1 Anforderungen an die Beschaffenheit der Datengrundlage

Die Datenbeschaffenheit der Beispieldatensätze soll möglichst nah an Datensätzen aus der realen Welt sein. Die Datensätze können daher sowohl kategorische, numerische, als auch gemischte Attribute enthalten. Eine Beschränkung der Datensätze auf nur ein Skalenniveau soll vermeiden werden, da dies in der realen Welt auch nur selten anzutreffen ist. Dabei sollen Datensätze verwendet werden, die vorher nicht bereinigt wurden.

Die Datenkomplexität der Bespieldatensätze muss möglichst nah an Datensätzen aus der realen Welt sein. Durch eine realitätsnahe Datenkomplexität kann sichergestellt werden, dass die Klassifikatoren auch bei nicht intuitiv erkennbaren Zusammenhängen funktionieren. Als Quelle der Datensätze sollen öffentlich zugänglichen Plattformen mit Datensätzen aus einem ökonomischen und produktionslogistischen Umfeld dienen.

3.1.2 Beispieldatensätze aus öffentlich zugänglichen Plattformen

Im Folgenden werden die gewählten Datensätze aus der öffentlich zugänglichen Plattform kaggle.com genauer erläutert.

Der Walmart Datensatz ist ein 2014 auf kaggle.com von Walmart veröffentlichter Datensatz. Er wurde im Rahmen eines Rekrutierungswettbewerbs veröffentlicht. (Walmart Inc. 2014) Es handelt sich also um einen Datensatz aus der realen Welt. Dieser enthält sowohl kategorische als auch numerische Attribute.

Der Rossmann Datensatz wurde 2015 auf kaggle.com im Rahmen eines Wettbewerbs von der Drogeriekette Rossmann GmbH veröffentlicht. Ziel des Wettbewerbs war es, die Umsätze der Filialen vorherzusagen. (Rossmann GmbH 2015b) Auch hierbei handelt es sich also um einen Datensatz aus der realen Welt. Es sind kategorische und numerische Attribute enthalten.

Walmart

Als erster Beispieldatensatz dienen historische Umsatzdaten für 45 Filialen des Einzelhandelunternehmens Walmart in verschiedenen Regionen. Jede Filiale hat dabei mehrere Abteilungen. Während des Erfassungszeitraums wurden das ganze Jahr über mehrere Preisreduzierungsaktionen durchgeführt. Diese Preisreduzierungen fanden vor Feiertagen wie dem Super Bowl, Labor Day, Thanksgiving und Weihnachten statt. Die Wochen, in denen diese Feiertage liegen, werden fünfmal höher bei der Auswertung gewichtet als Wochen ohne Feiertage. Insgesamt sind die Daten in drei Tabellen aufgeteilt: *Stores*, *Features* und *Sales*. (Walmart Inc. 2014)

Die Datei *Stores* enthält anonymisierte Informationen zu den 45 Filialen in 45 Zeilen mit Angabe der Nummer der Filiale, Art und Größe des Geschäfts.

In der Datei *Features* sind zusätzliche Merkmale zu den Filialen, den Abteilungen und weitere regionale Informationen in 12 Attributen und 8.190 Zeilen enthalten.

Die in *Features* enthaltenen Attribute werden im Folgenden nach dem Bereitsteller der Daten von kaggle.com, Singh, genauer erläutert:

- *Store*: die Nummer der Filiale
- *Date*: die Woche
- *Temperature*: Durchschnittstemperatur der Region

- *Fuel_Price*: Kosten für Kraftstoff in der Region
- *MarkDown1-5*: anonymisierte Daten über Sonderangebotsabschläge, welche erst seit November 2011 und nicht immer für alle Filialen verfügbar sind
- *CPI*: der Verbraucherpreisindex
- *Unemployment*: die Arbeitslosenquote
- *IsHoliday*: ist wahr, wenn die Woche eine Feiertagswoche ist

Die Datei *Sales* enthält historische Verkaufsdaten aus dem Zeitraum 05.02.2010 bis 01.11.2012 in fünf Attributen und 421.570 Zeilen. Folgend werden die Attribute der von Singh bereitgestellten Datei kurz erläutert:

- *Store*: die Nummer der Filiale
- *Dept*: die Nummer der Abteilung
- *Date*: die Woche
- *Weekly_Sales*: der wöchentliche Umsatz für die angegebene Abteilung in der angegebenen Filiale
- *IsHoliday*: ist wahr, wenn die Woche eine Feiertagswoche ist

Um aus den drei Tabellen eine Tabelle mit allen Informationen zu generieren, werden diese mit Joins zusammengefügt. Zuerst werden die Tabellen *Sales* und *Features* über einen Inner Join zusammengeführt. Dabei ist die Tabelle *Sales* links und die Tabelle *Features* rechts. Als Schlüsselattribute dienen Date und Store, welche in beiden Tabellen enthalten sind. Doppelte Attribute werden entfernt. Um auch die Informationen aus *Stores* einzubinden, werden nun die neue Tabelle und die Tabelle *Stores* mithilfe eines Inner Joins zusammengeführt. Die neue Tabelle ist dabei links und die Tabelle *Stores* rechts. Als Schlüsselattribut dient bei diesem Join das Attribut *Store*, welches in beiden Tabellen enthalten ist. Die nun vorliegende Tabelle enthält 16 Attribute und 421.570 Zeilen. Von den Attributen sind zwei kategorisch, 13 numerisch und eins ist ein Datum. Die Attribute *MakrDown1-5* enthalten auch fehlende Werte.

Damit der Datensatz für eine umfängliche Analyse genutzt werden kann, werden aus dem Datensatz noch zwei weitere Datensätze generiert. Dazu wird der Datensatz, der sowohl numerische als auch kategorische Daten enthält, in einen metrischen und einen kategorischen Datensatz umgewandelt. Mithilfe dieser drei Datensätze werden in Kapitel 3.2 die in der Software RapidMiner enthaltenen Werkzeuge analysiert, die Anwender beim Prozess der Datenvorverarbeitung unterstützen sollen. Durch eine Analyse mithilfe eines gemischten, eines rein metrischen und eines rein kategorischen Datensatzes wird das Verhalten der Software bei unterschiedlichen Skalenniveaus genauer betrachtet.

Für die Umwandlung in einen rein metrischen Datensatz müssen die zwei kategorischen Attribute *IsHoliday* und *Type* in metrische Attribute umgewandelt werden. Da es sich bei dem Attribut *IsHoliday* um ein binäres Attribut handelt, werden die beiden Ausprägungen *TRUE* und *FALSE* durch *1* und *0* ersetzt. Das Attribut *Type* hat die Ausprägungen *A, B* und *C*. Damit keine Rangfolge bei der Umwandlung in ein metrisches Attribut entsteht, werden mithilfe einer Dummy-Kodierung drei neue Attribute generiert: *Type = A*, *Type = B* und *Type = C*. Die Ausprägungen sind dann jeweils binär mit *1* und *0*.

Um einen rein kategorischen Datensatz zu generieren, müssen insgesamt 13 numerische Attribute in kategorische Attribute umgewandelt werden. Die Umwandlung der Attribute *Store* und *Dept* kann ohne Informationsverlust erfolgen, da bei den Ausprägungen keine Rangfolge besteht. Die Umwandlung der Attribute *Weekly_Sales*, *Temperature*, *Fuel_Price*, *MarkDown1-5*, *CPI* und *Size* kann nur über eine Diskretisierung realisiert werden. Diese Diskretisierung erfolgt über Binning. Als letztes nicht kategorisches Attribut bleibt das Datum. Zur Umwandlung wird das Datum aufgeteilt in *Date_month*, *Date_day* und *Date_year* und diese anschließend in kategorische Attribute umgewandelt.

Nun liegen drei Datensätze vor: ein Datensatz hat gemischte Skalenniveaus, einer nur numerische und einer nur kategorische. Alle drei Datensätze enthalten auch fehlende Attribute. Für eine genaue Analyse werden zusätzlich noch jeweils Datensätze ohne fehlende Attribute generiert. Mithilfe der Datensätze mit und ohne fehlende Attribute werden in Kapitel 3.2 die in der Software RapidMiner enthaltenen Werkzeuge analysiert, welche Anwender beim Prozess der Datenvorverarbeitung unterstützen sollen. Durch eine Analyse mit Datensätzen, die sich lediglich durch das Fehlen bzw. durch das Nicht Fehlen einiger Werte unterscheiden, wird das Verhalten der Software im Umgang mit fehlenden Werten genauer betrachtet. Bei dem gemischten und dem rein numerischen Datensatz werden die fehlenden Werte durch den Wert *0* ersetzt. Bei dem rein kategorischen Datensatz erfolgt das Binning der numerischen Attribute erst nach der Ersetzung der fehlenden Werte durch den Wert *0* um einen rein kategorischen Datensatz zu erzeugen. So sind am Ende also sechs Datensätze zur Analyse vorhanden:

- Kategorische und metrische Attribute mit fehlenden Werten
- Kategorische und metrische Attribute ohne fehlende Werte
- Nur numerische Attribute mit fehlenden Werten
- Nur numerische Attribute ohne fehlende Werte
- Nur kategorische Attribute mit fehlenden Werten
- Nur kategorische Attribute ohne fehlende Werte

Die fünf selbst generierten Datensätze sollen für die Analyse unterstützenden Datenvorverarbeitungsprozesse in RapidMiner dienen, auf die in Kapitel 3.2 intensiv eingegangen wird.

Rossmann

Der zweite Datensatz enthält historische Verkaufsdaten aus 1.115 Rossmann-Filialen. Einige Filialen waren während der Datenerfassung wegen Renovierungsarbeiten vorrübergehend geschlossen. Die Daten sind in drei Tabellen aufgeteilt: *Store*, *Train* und *Test*. Die Datei *Test* ist enthalten, um den Teilnehmern des Wettbewerbs das Testen ihres entwickelten Modells zu ermöglichen. (Rossmann GmbH 2015a)

Die Datei *Store* enthält zehn Attribute in 1.115 Zeilen mit Informationen über die 1.115 Filialen. Die Attribute werden im Folgenden nach den Angaben von Rossmann GmbH auf kaggle.com erläutert:

- *Store*: eindeutige ID für jede Filiale
- *StoreType*: unterscheidet zwischen vier verschiedenen Ladenmodellen: *a*, *b*, *c*, *d*
- *Assortment*: beschreibt eine Sortimentsstufe: *a* = basic, *b* = extra, *c* = extended

- *CompetitionDistance*: Entfernung zum Markt des nächstgelegenen Konkurrenten in Metern
- *CompetitionOpenSinceMonth*: der ungefähre Monat, in dem der Markt des nächstgelegenen Konkurrenten eröffnet wurde
- *CompetitionOpenSinceYear*: das ungefähre Jahr, in dem der Markt des nächstgelegenen Konkurrenten eröffnet wurde
- *Promo2*: eine fortlaufende und aufeinanderfolgende Werbeaktion für einige Geschäfte: *0* = Geschäft nimmt nicht teil, *1* = Geschäft nimmt teil
- *Promo2SinceWeek*: die Kalenderwoche, in der die Filiale mit der Teilnahme an *Promo2* begonnen hat
- *Promo2SinceYear*: das Jahr, in dem die Filiale mit der Teilnahme an *Promo2* begonnen hat
- *PromoInterval*: beschreibt die Intervalle, in denen *Promo2* gestartet wird. Es werden die Monate genannt, in denen die Aktion neu gestartet wird. *Feb,May,Aug,Nov* bedeutet beispielsweise, dass jede Aktion für diese Filiale im Februar, Mai, August und November eines bestimmten Jahres beginnt

Die Datei Test enthält historische Verkaufsdaten aus dem Zeitraum 01.01.2013 bis 31.07.2015 in neun Attributen und 1.017.209 Zeilen. Die Attribute werden im Folgenden nach den Angaben von Rossmann GmbH auf kaggle.com erläutert:

- *Store*: eindeutige ID für jede Filiale
- *DayOfWeek*: gibt den Wochentag an: *1* = Montag, *2* = Dienstag, *3* = Mittwoch usw.
- *Date*: das Datum des jeweiligen Datensatzes
- *Sales*: der Umsatz eines Tages
- *Customers*: die Anzahl der Kunden an einem Tag
- *Open*: gibt an, ob der Laden geöffnet war: *0* = geschlossen, *1* = geöffnet
- *Promo*: gibt an, ob eine Filiale an diesem Tag eine Werbeaktion durchführt: *0* = Werbeaktion findet nicht statt, *1* = Werbeaktion findet statt
- *StateHoliday*: gibt gesetzliche Feiertage an: *a* = Feiertag, *b* = Osterfeiertag, *c* = Weihnachten, *0* = kein Feiertag. Es wird von Rossmann GmbH darauf hingewiesen, dass Geschäfte bis auf wenige Ausnahmen und Schulen grundsätzlich an gesetzlichen Feiertagen geschlossen sind
- *SchoolHoliday*: gibt an, ob die Filiale an diesem Daten von der Schließung öffentlicher Schulen betroffen war: *0* = nicht betroffen, *1* = betroffen

Um eine Tabelle mit allen Informationen zu generieren, werden die beiden Tabellen *Store* und *Test* mit einem Join zusammengeführt. Dazu wird ein Inner Join verwendet. Die Tabelle *Sales* ist links und die Tabelle *Store* rechts. Dabei dient das Attribut *Store* als Schlüsselattribut. Die so generierte Tabelle enthält 18 Attribute und 1.017.209 Zeilen. Von den 18 Attributen sind acht kategorisch, neun metrisch und eins ist ein Datum. Die Attribute *CompetitionDistance*, *CompetitionOpenSinceMonth*, *CompetitionOpenSinceYear*, *Promotion2SinceWeek*, *Promotion2SinceYear* und *PromoInterval* enthalten fehlende Werte.

3.2 Untersuchung des unterstützenden Datenvorverarbeitungsprozesses in RapidMiner

Data-Mining-Verfahren werden in RapidMiner nicht nach ihrer Aufgabenstellung strukturiert, was wie unter 2.3.1 erläutert in der Forschung und Literatur üblich ist. Bei der Erarbeitung der Klassifizierungsformen in RapidMiner in Unterkapitel 2.3.2 wird deutlich, dass die Strukturierung teilweise zusammen mit verfahrensabhängigen Operatoren in ihre übergeordneten Ziele Vorhersage und Beschreibung realisiert wird. Der überwiegende Teil der Verfahren und verfahrensabhängigen Operatoren für Vorverarbeitungsschritte mit dem Ziel der Vorhersage ist in dem Ordner Predictive zusammengefasst. Lediglich im Ordner Time Series sind noch Verfahren und Operatoren zur Vorhersage von Zeitreihen enthalten. Die anderen Ordner enthalten Verfahren und Operatoren zur Datenvorverarbeitung und Beschreibung der Daten. Data-Mining-Verfahren sind also nicht nur nach ihrer Struktur, sondern auch mit den dazu anwendbaren verfahrensabhängigen Operatoren strukturiert. Die Gründe dafür liegen möglicherweise in der schrittweisen Entwicklung der Software und der Realisierung einer hohen Benutzerfreundlichkeit. Dazu können jedoch weder Literatur noch Information von RapidMiner ausfindig gemacht werden.

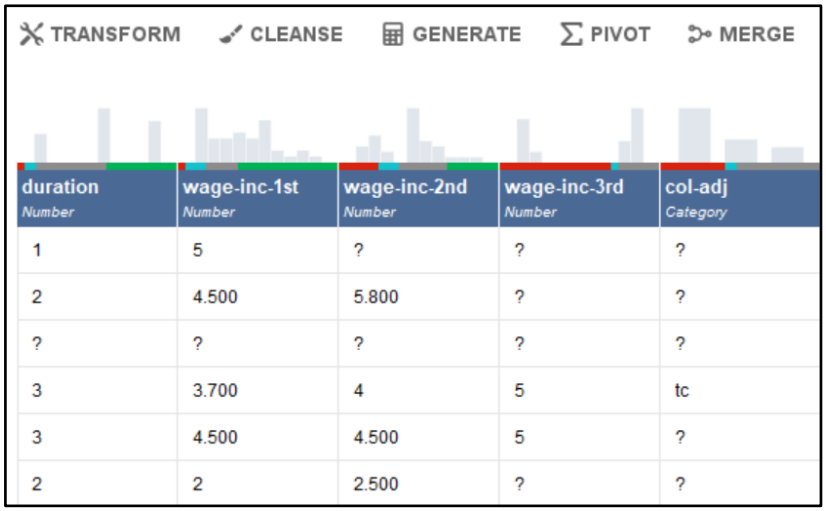
Die Software RapidMiner enthält standardmäßig Werkzeuge, welche Anwender beim Prozess der Datenvorverarbeitung und bei Anwendung eines Modells unterstützen sollen. *Turbo Prep* wurde entwickelt, um Anwender bei der Datenvorverarbeitung zu unterstützen. Das Werkzeug enthält eine Benutzeroberfläche, auf der die Daten während der Bearbeitung die ganze Zeit über sichtbar sind. Innerhalb dieser Oberfläche können Schritt für Schritt Änderungen vorgenommen werden und die Ergebnisse sind sofort sichtbar. Das Werkzeug *Auto Model* soll Anwender beim Prozess der Erstellung und Validierung eines oder verschiedener Modelle unterstützen. (RapidMiner 2022)

In diesem Kapitel soll die unterstützende Datenvorverarbeitung mithilfe des Werkzeugs *Turbo Prep* genauer analysiert und so mögliche von RapidMiner genutzte Klassifikatoren identifiziert und untersucht werden.

3.2.1 Geführter Ansatz in RapidMiner

Um einen Datensatz in *Turbo Prep* zu bearbeiten, muss dieser zuvor in RapidMiner importiert werden. Sobald der Datensatz importiert ist, kann dieser in *Turbo Prep* geöffnet werden. Wenn ein Datensatz geladen ist, öffnet sich die in Abbildung 3-1 dargestellte Benutzeroberfläche.

Auf dieser Benutzeroberfläche sind die Daten und einige allgemeine statistische Informationen zu sehen. Unter anderem sind das die statistische Verteilung, das Minimum, das Maximum, der Durchschnitt des jeweiligen Attributs und von RapidMiner eigens definierte Kennzahlen.



duration *Number*	wage-inc-1st *Number*	wage-inc-2nd *Number*	wage-inc-3rd *Number*	col-adj *Category*
1	5	?	?	?
2	4.500	5.800	?	?
?	?	?	?	?
3	3.700	4	5	tc
3	4.500	4.500	5	?
2	2	2.500	?	?

Abbildung 3-1: Screenshot der Benutzeroberfläche in *Turbo Prep*

Neben den bekannten statistischen Kennzahlen, dem Minimum, Maximum und Durchschnitt, stellt RapidMiner also auch eigene Kennzahlen zur Verfügung, welche alle in Prozent angegeben und in der Übersicht farblich als Balken dargestellt werden. Sie werden im Hilfebereich der *Turbo Prep* wie folgt definiert:

- *Missing* (rot): Die Anzahl der fehlenden Werte einer Spalte geteilt durch die Anzahl der Zeilen (RapidMiner 2022b)
- *Infinite* (rot): Die Anzahl der Werte die gegen unendlich gehen einer Spalte geteilt durch die Anzahl der Zeilen (RapidMiner 2022b)
- *ID-ness* (blau) Die Anzahl der unterschiedlichen Werte einer Spalte geteilt durch die Anzahl der Zeilen (RapidMiner 2022b)
- *Stability* (grau): Die Anzahl der am häufigsten vorkommenden (nicht fehlenden) Werte einer Spalte geteilt durch die Anzahl der Zeilen (RapidMiner 2022b)
- *Valid* (grün): Der Anteil der Werte dieser Spalte, die nicht als fehlend, gegen unendlich, als ID oder stabil gezählt werden (RapidMiner 2022b)

Die Benutzeroberfläche der *Turbo Prep* ist, wie in Abbildung 3-1 zu erkennen, in fünf Kategorien für die Datenvorverarbeitung unterteilt:

- *Transform*: Über *Transform* sind Datentransformationen wie die Umbenennung von Attributen, das Filtern nach selbst wählbaren Kriterien oder die Transformation des Datentyps möglich.
- *Cleanse*: Hierüber sind Datenbereinigungsschritte wie das Entfernen von Ausprägungen mit selbst einstellbaren Kriterien, das Entfernen von Ausprägungen mit besonders hohen oder niedrigen Korrelationen, das Ersetzen von fehlenden Werten mit einfachen Verfahren, die Anwendung einer Normalisierung, Diskretisierung und PCA sowie das Entfernen von Duplikaten möglich. Ebenso ist hierüber die Anwendung des *Auto Cleansing* möglich, auf das später in diesem Unterkapitel genauer eingegangen wird.
- *Generate*: Mithilfe von *Generate* ist es möglich, neue Attribute über einem Formeleditor zu generieren

- *Pivot*: Hier können Datensätze gruppiert, Werte aggregiert und Tabellen in verschiedene Formate gedreht werden
- *Merge*: Über *Merge* ist das Zusammenführen unterschiedlicher Tabellen mit den Join Funktionen Inner, Outer, Left und Right Join über selbst wählbare Join Keys möglich

Insgesamt ermöglicht *Turbo Prep* Anwendern also eine umfängliche Datenvorverarbeitung inklusive Visualisierungen ohne Programmieraufwand. Das ist, wie Kapitel 2.4 verdeutlicht, besonders für Analysten ohne umfängliche Datenkompetenz hilfreich. Eine noch weitergehende Hilfestellung bei der Datenvorverarbeitung bietet die Funktion *Auto Cleansing* in der *Turbo Prep*. Mithilfe dieser Funktionen können gängige Bereinigungen eines Datensatzes von RapidMiner automatisch durchgeführt werden. Anwender können dabei angeben, ob im späteren Verlauf der Analyse eine Vorhersage auf ein bekanntes Zielattribut angewendet wird oder ob kein Zielattribut bekannt ist. Danach können Anwender einige Entscheidungen über Typenkonvertierungen und Datentransformationen treffen, welche in Unterkapitel 2.2.4 genauer erläutert werden. (RapidMiner 2022b) RapidMiner führt folgende Schritte automatisch durch:

- Entfernen von Spalten mit niedriger Qualität
- Ersetzen von fehlenden Werten
- Dummy-Kodierung (Umwandlung kategorischer in nummerische Attribute)
- Diskretisierung (Umwandlung numerischer in kategorische Attribute)
- PCA (optional und nur auf numerische Attribute anwendbar)
- Normalisierung (optional und nur auf numerisch Attribute anwendbar)

Die Funktion *Auto Cleansing* wird im Folgenden genauer analysiert, um zu untersuchen, auf welchen Grundlagen automatisierte Entscheidungen bei der Datenvorverarbeitung in RapidMiner getroffen werden.

In der Dokumentation von RapidMiner finden sich nur wenige Informationen zu *Auto Cleansing*. Dort wird lediglich beschrieben, dass von der Funktion *Auto Cleansing* Daten von geringer Qualität automatisch entfernt werden (RapidMiner 2022).

Um die Funktion zu untersuchen, werden die in Kapitel 3.1.2 aus dem Walmart Datensatz generierten Datensätze mit den Einzelhandelsdaten verwendet. Damit eine umfängliche Analyse erfolgen kann, werden aus dem Datensatz mit gemischtem Skalenniveau insgesamt noch fünf weitere Datensätze, jeweils mit unterschiedlichem Skalenniveau und sowohl mit als auch ohne fehlenden Werten, generiert. Diese sechs Datensätze werden nun alle mit *Auto Cleansing* bearbeitet. Exemplarisch wird im Folgenden der originale Walmart Datensatz, also ein Datensatz mit kategorischen und metrischen Attributen sowie fehlenden Werten, in *Auto Cleansing* bearbeitet.

Bei Start von *Auto Cleansing* öffnet sich ein neues Fenster in RapidMiner mit einem Balken, der insgesamt fünf Schritte anzeigt: *Define Target*, *Improve Quality*, *Change Types*, *Handle Numbers* und *Summary*. Beim ersten Schritt, *Define Target*, kann eine Spalte der Tabelle, also ein Zielattribut, ausgewählt werden, oder die Option keine Zielspalte zu bestimmen gewählt werden. Dabei wird lediglich nach der Spalte des Zielattributs gefragt, das in Kapitel 2.3.1 beschriebene übergeordnete Ziel, also ob es sich um eine Vorhersage oder Beschreibung handelt, spielt dabei keine Rolle und steht auch nicht zur Auswahl. Bei diesem Durchgang wird keine

Zielspalte gewählt. Danach geht RapidMiner zum nächsten Schritt: *Improve Quality*. Dieser Schritt dient laut einem kurzen Text im *Auto Cleansing* Fenster lediglich zur Information (RapidMiner 2022b). RapidMiner wird die gekennzeichneten Spalten entfernen, da diese laut RapidMiner eine zu niedrige Qualität für maschinelle Lernverfahren haben (RapidMiner 2022b). Die fehlenden Werte werden laut RapidMiner ersetzt (RapidMiner 2022b). Wie diese Werte ersetzt werden, wird von RapidMiner nicht definiert. Gekennzeichnet werden hier die Attribute *IsHoliday* mit „High stability“ und *MarkDown2* mit „Many Missing“. Danach geht RapidMiner zum Schritt *Change Types.* Bei diesem Schritt kann das Skalenniveau aller Attribute entweder so beibehalten werden, wie es ist, alle Attribute in numerische oder alle Attribute in kategorische umgewandelt werden. Wie die Umwandlung der Attribute erfolgen soll, kann nicht ausgewählt werden. Zusätzlich erscheint ein Hinweis, dass bei einer späteren Anwendung von *Auto Model* keine Umwandlung erfolgen soll, da dies durch *Auto Model* selbst je nach angewendetem Data-Mining-Verfahren erfolgen würde. Bei diesem Durchgang wird die Option gewählt, alle Attribute so zu lassen, wie sie sind. Nach diesem Schritt geht RapidMiner zum Schritt *Handle Numbers*. Beim Schritt *Handle Numbers* besteht die Möglichkeit, eine PCA oder eine Normalisierung auf die metrischen Attribute anzuwenden. Diese Datentransformationen werden in Unterkapitel 2.2.4 genauer erläutert. Bei diesem Durchgang wird keines der Verfahren gewählt. Als letzter Schritt wird bei *Summary* eine Zusammenfassung der Schritte angezeigt, die RapidMiner durchführen wird: Die Attribute entfernen, die im Schritt *Improve Quality* gekennzeichnet wurden und die fehlenden Werte ersetzen. Danach schließt sich das geöffnete Fenster von *Auto Cleansing* und der von RapidMiner bearbeitete Datensatz wird angezeigt. Wie angekündigt, werden die im Schritt *Improve Quality* gekennzeichneten Attribute entfernt. Die fehlenden Werte bei *MarkDown1* und *MarkDown3-5* werden durch den Durchschnitt des jeweiligen Attributs ersetzt.

Um zu analysieren, welche Entscheidungen von *Auto Cleansing* bei welchem Skalenniveau getroffen werden und ob ein bekanntes Zielattribut und fehlende Werte Auswirkungen auf die Entscheidungen von RapidMiner zur Vorverarbeitung haben, wird *Auto Cleansing* in insgesamt 12 Durchgängen auf die sechs Datensätze strukturiert angewendet. Die Informationen zu den Datensätzen, welche bei den Durchgängen verwendet werden, können Tabelle 3-1 entnommen werden.

Tabelle 3-1: Informationen zu den Datensätzen

Durchgang	Zielvariable	Skalenniveau	Fehlende Werte
1	unbekannt	metrisch	ja
2	unbekannt	metrisch	nein
3	bekannt	metrisch	ja
4	bekannt	metrisch	nein
5	unbekannt	gemischt	ja
6	unbekannt	gemischt	nein
7	bekannt	gemischt	ja
8	bekannt	gemischt	nein
9	unbekannt	kategorisch	ja
10	unbekannt	kategorisch	nein
11	bekannt	kategorisch	ja
12	bekannt	kategorisch	nein

Die bei den 12 Durchgängen durchgeführten fünf Schritte und gewählten Vorverarbeitungsschritte im Fenster von *Auto Cleansing* können Tabelle 3-2 entnommen werden.

Tabelle 3-2: Informationen zu den Schritten in *Auto Cleansing*

Durch-gang	Define Target	Improve Quality	Change Types	Handle Numbers	Summary
1	No Target column	Schritt findet statt	Keep Original	Ohne Auswahl	1. Remove low quality columns 2. Replace missing Values
2	No Target column	Schritt findet statt	Keep Original	Ohne Auswahl	1. Remove low quality columns 2. Replace missing Values
3	Weekly_ Sales	Schritt findet statt	Keep Original	Ohne Auswahl	1. Target column: Weekly_Sales 2. Remove low quality columns 3. Replace missing values
4	Weekly_ Sales	Schritt findet statt	Keep Original	Ohne Auswahl	1. Target column: Weekly_Sales 2. Remove low quality columns 3. Replace missing values
5	No Target column	Schritt findet statt	Keep Original	Ohne Auswahl	1. Remove low quality columns 2. Replace missing Values
6	No Target column	Schritt findet statt	Keep Original	Ohne Auswahl	1. Remove low quality columns 2. Replace missing Values
7	Weekly_ Sales	Schritt findet statt	Keep Original	Ohne Auswahl	1. Target column: Weekly_Sales 2. Remove low quality columns 3. Replace missing values
8	Weekly_ Sales	Schritt findet statt	Keep Original	Ohne Auswahl	1. Target column: Weekly_Sales 2. Remove low quality columns 3. Replace missing values
9	No Target column	Schritt findet statt	Keep Original	Wird über-sprungen	1. Remove low quality columns 2. Replace missing Values
10	No Target column	Schritt findet statt	Keep Original	Wird über-sprungen	1. Remove low quality columns 2. Replace missing Values
11	Weekly_ Sales	Schritt findet statt	Keep Original	Wird über-sprungen	1. Target column: Weekly_Sales 2. Remove low quality columns 3. Replace missing values
12	Weekly_ Sales	Schritt findet statt	Keep Original	Wird über-sprungen	1. Target column: Weekly_Sales 2. Remove low quality columns 3. Replace missing values

In den folgenden Absätzen werden die erkannten Unterschiede bei den Kennzeichnungen der Attribute der jeweiligen Durchgänge beschrieben. Diese sind zur Übersicht in Abbildung 3-2 als Matrix dargestellt.

Durchgang	Fehlende Werte	Kennzeichnung / Skalenniveau	*Store*	*Dept*	*Date*	*Weekly_Sales*	*IsHoliday*	*Temperature*	*Fuel_Price*	*MarkDown1*	*MarkDown2*	*MarkDown3*	*MarkDown4*	*MarkDown5*	*CPI*	*Unemployment*	*Type*	*Size*
1	ja	metrisch																
2	nein	metrisch																
3	ja	metrisch																
4	nein	metrisch																
5	ja	gemischt																
6	nein	gemischt																
7	ja	gemischt																
8	nein	gemischt																
9	ja	kategorisch																
10	nein	kategorisch																
11	ja	kategorisch																
12	nein	kategorisch																

Legende	
Zielattribut	
Many Missing	
High Stability	
Many Values	

Abbildung 3-2: Kennzeichnung der Attribute in *Turbo Prep*

Bei Durchgang 1, 3, 5 und 7 werden die Attribute *IsHoliday* mit „High stability“ und *MarkDown2* mit „Many Missing“ gekennzeichnet und anschließend gelöscht. Alle diese Durchgänge enthalten fehlende Werte und ein metrisches oder gemischtes Skalenniveau. Bei den Durchgängen 2, 4, 6 und 8, welche keine fehlenden Werte und ebenfalls ein metrisches oder gemischtes Skalenniveau enthalten, wird lediglich das Attribut *IsHoliday* mit „High stability“ gekennzeichnet und anschließend gelöscht. Das Attribut *MarkDown2* wird nicht mehr gekennzeichnet und auch nicht gelöscht. Hier ist also festzuhalten, dass bei den Durchgängen 1-8, also sowohl bei den gemischten als auch bei den rein metrischen Attributen, jeweils die gleichen Attribute gekennzeichnet und gelöscht werden. Bei den Durchgängen 3, 4, 7 und 8 wird *Weekly_Sales* als Zielattribut gekennzeichnet. Dabei kann kein Unterschied zu den Durchgängen ohne bekannte Zielvariable festgestellt werden.

Die Durchgänge 9 bis 12 enthalten rein kategorische Attribute. Bei den Durchgängen 9 und 10 wird kein Zielattribut gekennzeichnet und RapidMiner hat 6, bzw. 7 Attribute gekennzeichnet. Bei Durchgang 9 werden die Attribute *Dept* mit „Many Values“ und *Weekly_Sales*, *IsHoliday*, *MarkDown2*, *MarkDown3* und *MarkDown5* mit „High Stability“ gekennzeichnet. Bei Durchgang 10 werden die gleichen Attribute und zusätzlich auch *MarkDown4* mit „High Stability“ gekennzeichnet. Alle gekennzeichneten Attribute werden von RapidMiner gelöscht. Wird *Weekly_Sales* wie in den Durchgängen 11 und 12 als Zielattribut gekennzeichnet, wird dieses nicht mehr mit „High Stability“ gekennzeichnet und auch nicht mehr gelöscht. Bei den Durchgängen 9 und 11 werden die fehlenden Werte mit „MISSING“ ersetzt. Bei den Durchgängen 10 und 12 listet RapidMiner im Schritt *Summary* „Replace missing values“

auf. Bei diesen Durchgängen gibt es jedoch keine fehlenden Werte, die RapidMiner ersetzen kann.

Insgesamt scheinen die Entscheidungen von RapidMiner in der Datenvorverarbeitung vor allem auf die beim Schritt *Improve Quality* gekennzeichneten Attributen zu beruhen. Hier werden Attribute mit „Many Missing“, „High Stability“ und “Many Values” gekennzeichnet. Mit „Many Values“ werden lediglich kategorische Attribute gekennzeichnet. Dieses Vorgehen wird im folgenden Kapitel genauer untersucht.

3.2.2 Identifizierung der von RapidMiner genutzten Klassifikatoren

RapidMiner nutzt in der unterstützenden Datenvorverarbeitung mit *Auto Cleansing* eigene Kennzahlen, um zu entscheiden, ob die Qualität von Attributen zu gering zur Nutzung in Data-Mining-Verfahren ist. Während des Schritts *Improve Quality* in der Funktion *Auto Cleansing* werden Attribute mit „Many Missing“, „High Stability“ und „Many Values” gekennzeichnet und anschließend gelöscht. Es handelt sich hierbei also um eine Dimensionsreduktion, die in Kapitel 2.2.3 genauer erläutert wird.

Zu der Kennzeichnung der kategorischen Attribute mit „Many Values“ gibt es keine Erklärung in der Dokumentation von RapidMiner und auch keine Informationen in der Software selbst. Die grundsätzliche Aussage dieser Kennzeichnung lässt sich so erklären, dass ein kategorisches Attribut zu viele Ausprägungen hat und sich deshalb nicht zur weiteren Analyse eignet. Im Beispieldatensatz sind es 81 Ausprägungen.

Die Kennzeichnungen „Many Missing“ und „High Stability“ lassen sich mit den in Kapitel 3.2.1 erläuterten RapidMiner eigenen Kennzahlen in Verbindung bringen. „Many Missing“ bedeutet also, dass hier die Kennzahl *Missing* berechnet wurde: Die Anzahl der fehlenden Werte dieser Spalte geteilt durch die Gesamtanzahl der Zeilen (RapidMiner 2022b). Im Hilfebereich ist die Information zu finden, dass RapidMiner Spalten mit „Many Missing“ kennzeichnet, wenn mehr als 70% der Werte dieser Spalte fehlen (RapidMiner 2022b). Dies kann auch bei Anwendung der Beispieldatensätze in Unterkapitel 3.2.1 festgestellt werden. In Tabelle 3-3 ist die von RapidMiner berechnete Kennzahl *Missing* beispielhaft bei einem der Durchgänge in *Turbo Prep* in Unterkapitel 3.2.1 zu erkennen. Das Attribut *MarkDown2* wird gekennzeichnet und im weiteren Verlauf automatisch entfernt. Die Attribute *MarkDown1* und *MarkDown3-5* bleiben knapp unter dem definierten Schwellenwert von 70% und werden nicht gekennzeichnet und auch nicht entfernt. Leider sind weder im Hilfebereich noch in der Dokumentation von RapidMiner Informationen dazu zu finden, warum ab einem Schwellenwert von 70% Attribute von RapidMiner gelöscht werden. In der Literatur gibt es viele verschiedene Meinungen dazu, wann ein Attribut zu viele fehlende Werte hat und entfernt werden sollte. Häufig wird jedoch die Meinung vertreten, dass es vor allem vom später anzuwenden Data-Mining-Verfahren abhängig gemacht werden sollte, wie mit welcher Anzahl von fehlenden Werten umzugehen ist (Pratama et al. 2016). Da in dieser Phase der Datenvorverarbeitung für RapidMiner jedoch noch nicht bekannt ist, welches Verfahren angewendet werden soll, ist der Schwellenwert von 70% kritisch zu betrachten, da dieser keine wissenschaftliche Grundlage hat.

Werden Attribute in *Auto Cleansing* mit “High Stability” gekennzeichnet, bedeutet das, dass RapidMiner die RapidMiner eigene Kennzahl *Stability* berechnet. Diese Kennzahl wird

berechnet, indem die Anzahl der am häufigsten vorkommenden Werte für eine Spalte durch die Anzahl der Zeilen dividiert wird (RapidMiner 2022b). Es werden also Attribute gekennzeichnet, in denen fast alle Werte identisch sind. Im Hilfebereich von RapidMiner ist die Information zu finden, dass die Spalten mit „High Stability“ gekennzeichnet werden, wenn die Kennzahl einen Wert von mehr als 90% aufweist. Auch bei Anwendung der Beispieldatensätze in Unterkapitel 3.2.1 kann dieser Schwellenwert festgestellt werden. In Tabelle 3-3 ist die von RapidMiner berechnete Kennzahl *Stability* beispielhaft bei einem der Durchgänge in *Turbo Prep* in Unterkapitel 3.2.1 zu erkennen. Das Attribut *IsHoliday* wird gekennzeichnet und im weiteren Verlauf automatisch entfernt. Das Attribut *Type* hat ebenfalls einen hohen Wert der Kennzahl, jedoch unter dem definierten Schwellenwert von 90% und wird auch nicht gekennzeichnet oder entfernt. Auch zu dieser Kennzahl sind weder in der Dokumentation noch im Hilfebereich von RapidMiner weitere Informationen zu finden, warum Attribute ab einem Wert von 90% gelöscht werden. In der Literatur wird der Schwellenwert von 90% auch genutzt, beispielsweise in Sohrabei et al. 2022 und Mandalapu und Gong 2019. Jedoch ist auch dort keine Begründung für den Schwellenwert zu finden. Generell ist in der Literatur die Begründung zu finden, dass sich die Stabilität von Attributen zuverlässig über einen Vergleich der Anzahl gleicher Werte berechnen lässt (Zdravevski et al. 2011).

Tabelle 3-3: Kennzahlen *Stability* und *Missing* beim Walmart Datensatz mit gemischtem Skalenniveau und fehlenden Werten

Attribut	Kennzahl Stability	Kennzahl Missing	Attribut wird von RapidMiner entfernt
Store	2,6%	0,0%	Nein
Dept	1,8%	0,0%	Nein
Date	1,0%	0,0%	Nein
Weekly_Sales	0,1%	0,0%	Nein
IsHoliday	92,8%	0,0%	Ja
Temperature	0,2%	0,0%	Nein
Fuel_Price	0,6%	0,0%	Nein
MarkDown1	0,3%	64,3%	Nein
MarkDown2	0,6%	73,6%	Ja
MarkDown3	0,6%	67,5%	Nein
MarkDown4	0,3%	68,0%	Nein
MarkDown5	0,3%	64,1%	Nein
CPI	0,3%	0,0%	Nein
Unemployment	1,4%	0,0%	Nein
Type	50,6%	0,0%	Nein
Size	5,5%	0,0%	Nein

Wie in Kapitel 2.1 deutlich wird, handelt es sich bei Data-Mining-Projekten um einen Prozess. Um den Prozess der Datenvorverarbeitung in *Auto Cleansing* von RapidMiner übersichtlich darzustellen, bietet es sich an, diesen als eine Ereignisgesteuerte Prozesskette (EPK) abzubilden. Die EPK beginnt, wie in Abbildung 3-3 dargestellt, damit, dass eine Datenmenge mit der Funktion *Auto Cleansing* bearbeitet wird. Nachdem die Funktion gestartet wurde, kann im Schritt *Define Target* ein Zielattribut ausgewählt werden. Dabei kann eine Spalte der Tabelle oder die Option gewählt werden, dass kein Zielattribut bekannt ist. Unabhängig davon, ob ein Zielattribut

ausgewählt wurde oder nicht, geht RapidMiner zum Schritt *Improve Quality*, also der Bestimmung der Datenqualität.

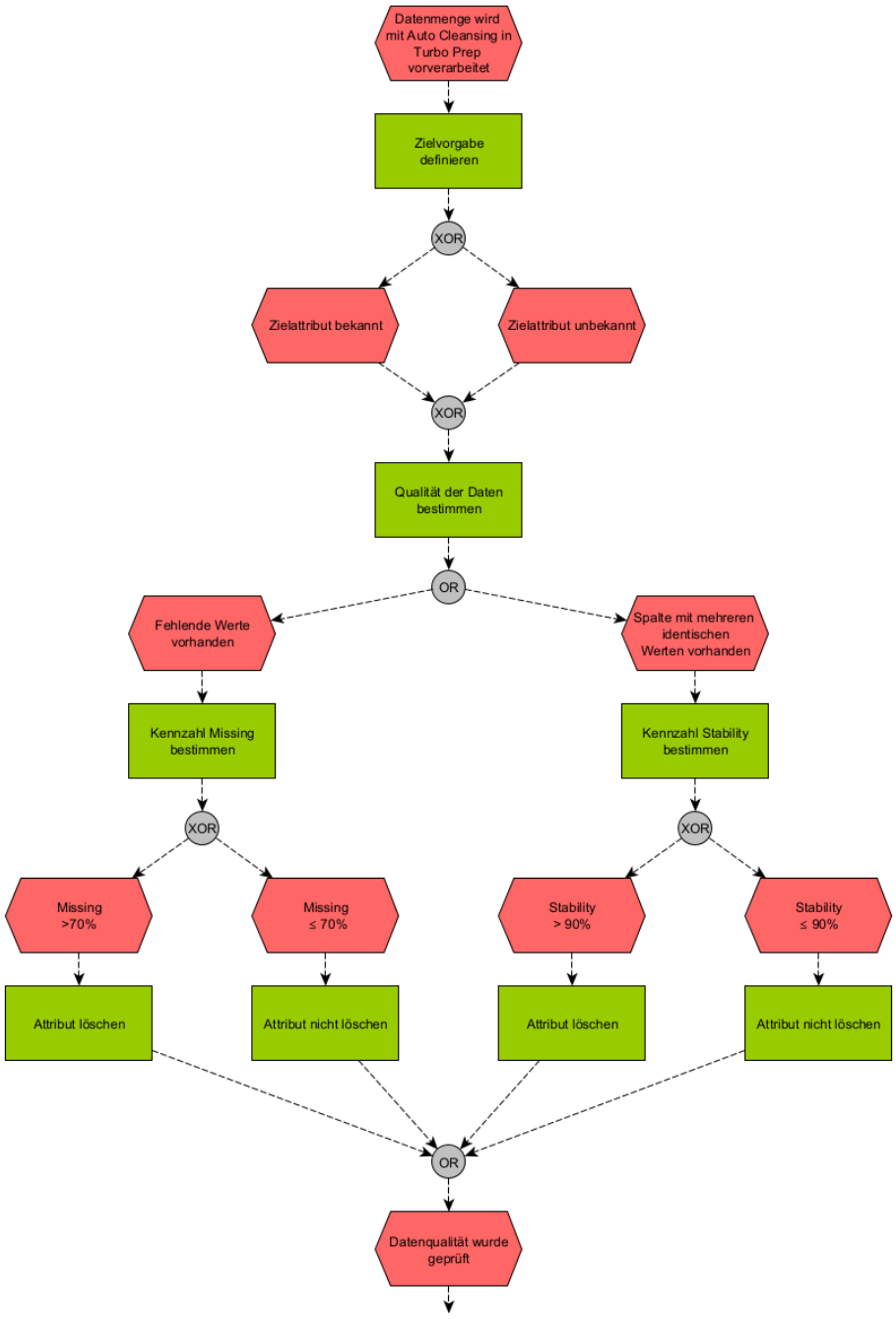


Abbildung 3-3: Prozess der Funktion *Auto Cleansing*: *Define Target* und *Improve Quality*

Beim Schritt *Improve Quality* werden die zuvor in diesem Kapitel genauer erläuterten Kennzahlen *Stability* und *Missing* von RapidMiner bestimmt. Enthält ein Attribut fehlende Werte ist die Kennzahl *Missing* von Relevanz. Dabei gibt es für RapidMiner nur zwei Optionen: *Missing* ist kleiner oder gleich 70% und das Attribut bleibt erhalten, oder *Missing* ist größer als 70% und das Attribut wird entfernt. Bei Attributen mit mehreren identischen Werten ist die Kennzahl *Stability* von Relevanz. Auch hier gibt es nur die zwei Optionen, dass die Kennzahl *Stability* kleiner oder gleich 90% ist und das Attribut erhalten bleibt, oder dass die Kennzahl größer als 90% ist und das Attribut entfernt wird. Für jedes Attribut werden jeweils beide Kennzahlen *Missing* und *Stability* berechnet und betrachtet. Liegt einer oder beide Werte über ihren definierten Schwellenwerten,

wird das Attribut entsprechend gekennzeichnet und anschließend entfernt. Nach Berechnung und Betrachtung der Kennzahlen ist der Schritt *Improve Quality* abgeschlossen und RapidMiner geht, wie in Abbildung 3-4 zu erkennen, zum nächsten Schritt *Change Types*.

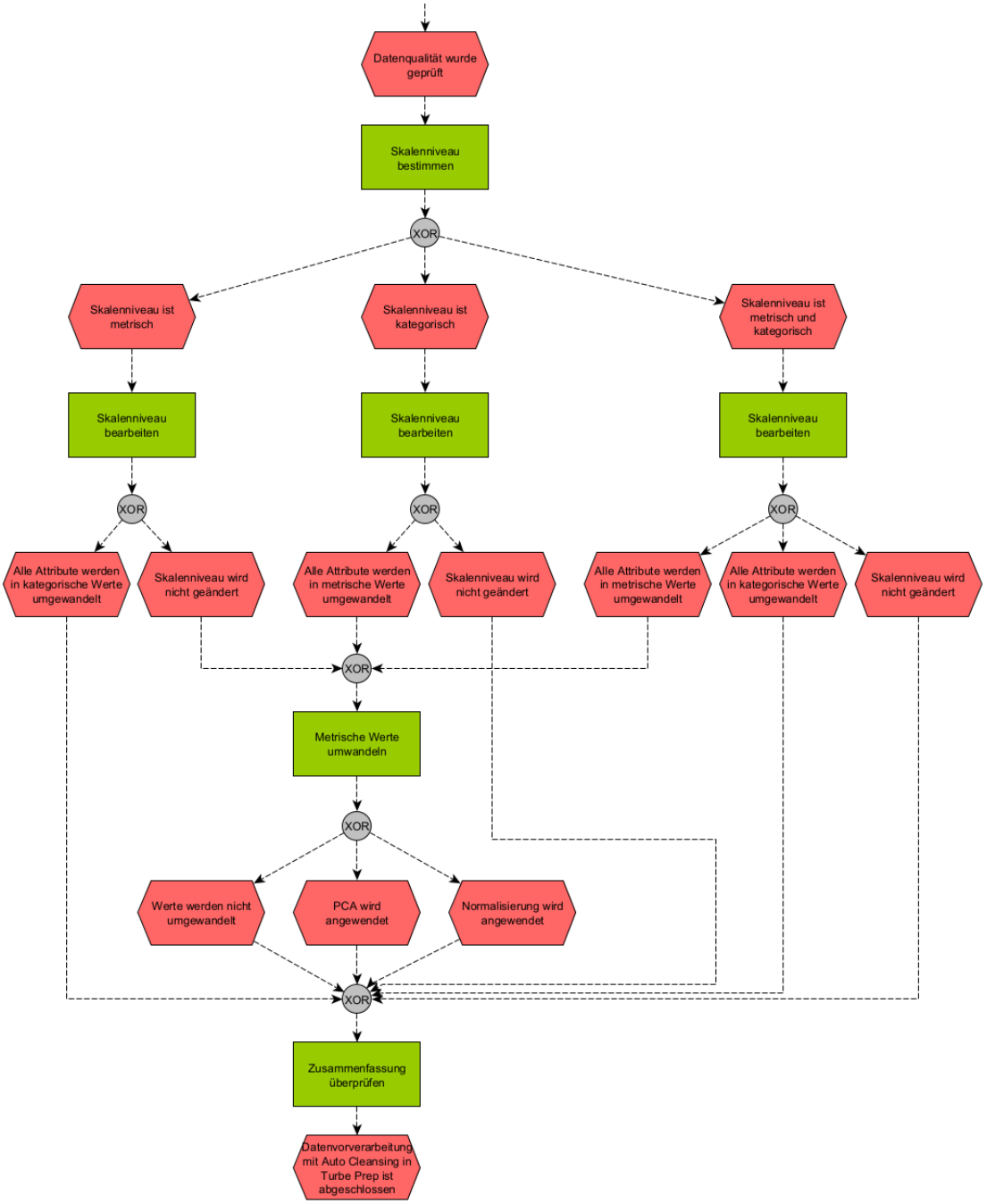


Abbildung 3-4: Prozess der Funktion *Auto Cleansing*: *Change Types*, *Handle Numbers* und *Summary*

Beim Schritt *Change Types* bietet RapidMiner Anwendern die Möglichkeit, das Skalenniveau des Datensatzes zu bearbeiten. Liegt ein rein metrisches Skalenniveau vor, kann das Skalenniveau so belassen werden, oder alle Attribute können in kategorische Werte umgewandelt werden. Wie die Umwandlung in kategorische Attribute erfolgt, können Anwender dabei nicht entscheiden. Wenn ein rein kategorisches Skalenniveau vorliegt, kann das Skalenniveau so belassen oder alle Attribute in metrische Werte umgewandelt werden. Auch bei der Umwandlung metrischer in

kategorische Attribute können Anwender nicht entscheiden, wie die Umwandlung erfolgen soll. Liegt ein gemischtes Skalenniveau vor, kann das Skalenniveau so belassen werden wie es ist, alle Attribute können in metrische oder alle Attribute in kategorische Attribute umgewandelt werden. Auch an dieser Stelle können Anwender nicht entscheiden, wie die Umwandlung der Attribute in ein anderes Skalenniveau umgesetzt werden soll. Liegt ein metrisches Skalenniveau vor, oder wurden kategorische Attribute in metrische Attribute umgewandelt, können die metrischen Attribute im Schritt *Handle Numbers* bearbeitet werden. Hierbei kann auf den Datensatz eine PCA oder eine Normalisierung angewendet werden. Diese Transformationen werden in Unterkapitel 2.2.4 genauer erläutert. Liegen rein kategorische Attribute vor, oder wurden alle Attribute in kategorische Attribute umgewandelt, wird der Schritt *Handle Numbers* von RapidMiner übersprungen, da keine metrischen Werte enthalten sind, die transformiert werden könnten.

Als letzten Schritt fasst RapidMiner die durchzuführenden Schritte von *Auto Cleansing* im Schritt *Summary* aufgelistet zusammen. Bei Bestätigung der Zusammenfassung vom Anwender, werden die aufgelisteten Schritte auf den Datensatz angewendet und der Prozess *Auto Cleansing* ist abgeschlossen.

Insgesamt werden in der Funktion *Auto Cleansing* die drei Klassifikatoren Zielattribut definieren, Datenqualität bestimmen und Skalenniveau bearbeiten identifiziert. Inwieweit diese Klassifikatoren in das neu entwickelte Klassifizierungskonzept integriert werden, wird im folgenden Kapitel betrachtet.

3.3 Entwicklung eines neuen Klassifizierungskonzepts

In diesem Kapitel wird ein neues Klassifizierungskonzept für Data-Mining-Verfahren entwickelt. Dazu werden zuerst die Anforderungen für die Klassifikatoren und anschließend Klassifikatoren definiert.

3.3.1 Anforderungen an Klassifikatoren

Ziel des Konzepts ist es, mit Hilfe von Klassifikatoren eine Übersicht über anwendbare Verfahren auf eine vorliegende Datengrundlage zu ermöglichen. Diese Übersicht muss nicht nur dann ermöglicht werden, wenn die Aufgabenstellung schon vorher bekannt ist, was, wie in Kapitel 2.3 gezeigt wird, bei Data-Mining-Prozessen üblich ist. Eine Übersicht über die auf den Datenbestand anwendbaren Verfahren muss auch ohne eine vorher festgelegte oder unbekannte Aufgabenstellung ermöglicht werden.

In Unterkapitel 2.2 wird deutlich, dass für die datenvorverarbeitenden Schritte der größte Zeitaufwand im Datenvorverarbeitungsprozess aufgewendet wird. Mithilfe der Klassifikatoren soll ermöglicht werden, dass schon vor Beginn einer Analyse Aufgabenstellungen ausgeschlossen werden und so die Anzahl der Iterationen im Datenvorverarbeitungsprozess reduziert werden kann.

In Unterkapitel 2.4 wird gezeigt, wie wichtig die Datenkompetenz der Anwender von Data-Mining-Verfahren für den gesamten Prozess ist. Aus diesem Grund werden die Klassifikatoren auf Basis der aufgezeigten Herausforderungen entwickelt, um auch Anwendern ohne besondere Datenkompetenz die Möglichkeit zu geben, grundlegende datenvorverarbeitende Schritte

durchzuführen. Dazu sollen die in RapidMiner identifizierten Klassifikatoren für die Datenvorverarbeitung als Unterklassifikatoren in das Konzept integriert werden.

In Tabelle 3-4 werden den Anforderungen Prioritäten und eine ID (Identifikationsnummer) zugeordnet.

Tabelle 3-4: Anforderungen an Klassifikatoren

ID	Anforderung	Priorität
A1	Eine Übersicht über anwendbare Verfahren ohne bekannte Aufgabenstellung ist gegeben.	Muss
A2	Aufgabenstellungen können schon vor Beginn der Analyse ausgeschlossen und Iterationen bei der Datenvorverarbeitung vermieden werden.	Soll
A3	In RapidMiner identifizierte Klassifikatoren werden als Unterklassifikatoren in das Konzept integriert.	Soll

Die Muss-Anforderung A1 ist zwingend umzusetzen, da es sich bei dieser Anforderung um das Hauptziel dieser Arbeit handelt. Wird sie nicht erfüllt, ist das Konzept nicht funktionsfähig. Sie ist von ihrer Priorität höher einzuordnen als die Soll-Anforderungen A2 und A3. Bei den Soll-Anforderungen handelt es sich um Anforderungen, die den Klassifikatoren zwar einen Mehrwert bringen können, welche aber nicht essenziell für die Funktionalität des Konzepts sind. Die Zuordnung der IDs und Prioritäten erfolgt, um den Erfüllungsgrad der Anforderungen nach Anwendung des Klassifizierungskonzepts strukturiert zu bewerten.

3.3.2 Definieren von Klassifikatoren

Als erster Klassifikator dient die Zielsetzung, beziehungsweise der Zweck der Analyse. Dabei wird unterschieden, ob die Daten nur beschrieben werden, eine Prognose erarbeitet wird, oder ob die Zielsetzung bisher, bewusst oder unbewusst, unbekannt ist. So ist schnell erkennbar, welche Verfahren auf die vorliegende Datenbasis bei welcher Zielsetzung anwendbar sind. Ein bekanntes Ziel, oder ein bekannter Zweck einer Analyse ist hierbei nicht gleichzusetzen mit einer Hypothese. In den Unterkapiteln 2.1 und 2.3 wird deutlich, dass Data-Mining zwar grundsätzlich hypothesenfrei ist, eine Unterscheidung zwischen den übergeordneten Zielen Prognose und Beschreibung in der Literatur jedoch üblich ist.

Ein weiterer Klassifikator wird über das Skalenniveau realisiert. Die Wichtigkeit die Skalenniveaus von Attributen zu beachten, wird in Unterkapitel 2.2.4 dargestellt. Für die spätere Analyse ist entscheidend, ob es sich um kategorische, metrische oder gemischte Attribute handelt.

Mit dem Klassifikator des Skalenniveaus wird direkt ein Klassifikator über die Datenvorverarbeitung realisiert. Wird eine Datenvorverarbeitung durchgeführt, kann das Skalenniveau, wie in Kapitel 2.2.4 beschrieben, bearbeitet werden. An dieser Stelle werden die in RapidMiner identifizierten Klassifikatoren als Unterklassifikatoren integriert.

Das Klassifizierungskonzept ist in Abbildung 3-5 als theoretisches Modell dargestellt.

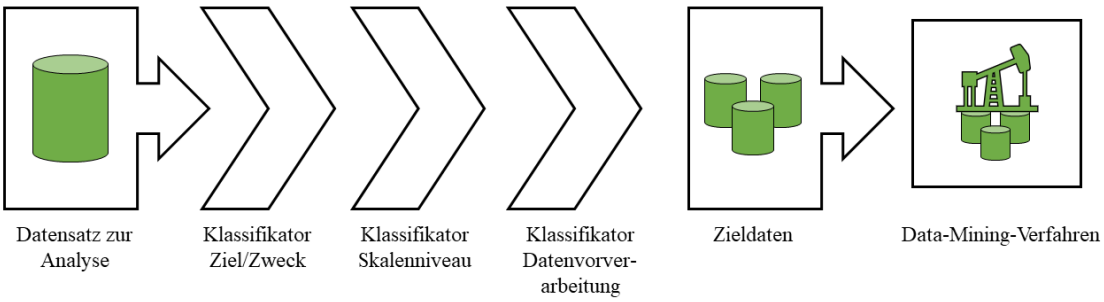


Abbildung 3-5: Klassifizierungskonzept

Wie in Abbildung 3-5 zu erkennen ist, dient der zur Analyse vorliegende Datensatz als Grundlage. Auf diesen Datensatz werden die drei Klassifikatoren Ziel/Zweck, Skalenniveau und Datenvorverarbeitung angewendet, um einen Zieldatensatz zu definieren. Anhand der angewendeten Klassifikatoren werden die anwendbaren Data-Mining-Verfahren systematisiert und auf die Zieldaten anwendbare Verfahren angezeigt.

Bei der Untersuchung der unterstützenden Datenvorverarbeitung in *Auto Cleansing* konnten die drei Klassifikatoren Zielattribut definieren, Datenqualität bestimmen und Skalenniveau bearbeiten identifiziert werden. RapidMiner verwendet Kennzahlen zur Entscheidungsfindung. Für den Schwellenwert der Kennzahlen kann keine wissenschaftliche Grundlage in der Literatur gefunden werden, jedoch wird festgestellt, dass die verwendeten Schwellenwerte auch in wissenschaftlichen Arbeiten zu finden sind. Um unerfahrenen Analysten oder Anwendern mit eingeschränkter Datenkompetenz bei der Entscheidungsfindung in der Datenvorverarbeitung zu unterstützen, werden die Kennzahlen in das Konzept integriert. Ob die Schwellenwerte der Kennzahlen funktionieren, wird bei der Implementierung untersucht. In Unterkapitel 2.2.2 wird deutlich, dass bei fehlenden Werten festzustellen ist, um welche Kategorie fehlender Werte es sich handelt, da es sonst zu einer Verzerrung der Ergebnisse kommen kann. Des Weiteren wird der Schwellenwert für das Löschen von Attributen noch weiter ausgearbeitet. In der Literatur gibt es dazu verschiedene Meinungen und Ansätze, weshalb hier ein eigener Ansatz zur Anwendung kommt. Der Ansatz von RapidMiner zum Umgang mit numerischen Werten wird nach Erarbeitung eines Konzepts als Unterklassifikator in das Konzept übernommen.

Um die Klassifikatoren im Prozess darzustellen, wird das Klassifizierungskonzept als EPK dargestellt.

3.3.3 Definieren eines Konzepts zur Anwendung der Klassifikatoren

In diesem Kapitel wird das in Unterkapitel 3.3.2 definierte theoretische Modell (Abbildung 3-5) als EPK ausgearbeitet und dargestellt. Grundlage für die EPK ist ein Datensatz, der zur Analyse vorliegt. Mithilfe von Klassifikatoren wird die EPK so ausgearbeitet, dass Anwender mithilfe der Klassifikatoren verschiedene Entscheidungen treffen können und ihnen am Ende die auf den Zieldatensatz anwendbaren Data-Mining-Verfahren angezeigt werden. Anschließend werden die erarbeiteten Unterklassifikatoren in das Konzept integriert.

3.3.3.1 Detailkonzept Phase 1: Darstellung der definierten Klassifikatoren als Prozess

Die EPK beginnt in Abbildung 3-6 mit dem Klassifikator Ziel/Zweck der Analyse. Das übergeordnete Ziel der Analyse kann in diesem Schritt bekannt oder unbekannt sein. Ist das übergeordnete Ziel bekannt, kann zwischen einer Beschreibung und einer Vorhersage gewählt werden. Nachdem das Ziel gewählt wurde, wird der nächste Klassifikator angewendet: Skalenniveau. Dieser Schritt ist ebenfalls in Abbildung 3-6 dargestellt. Hierzu wird bestimmt, ob es sich um ein metrisches, kategorisches oder gemischtes Skalenniveau handelt. Ist das übergeordnete Ziel unbekannt, wird ebenfalls das Skalenniveau bestimmt.

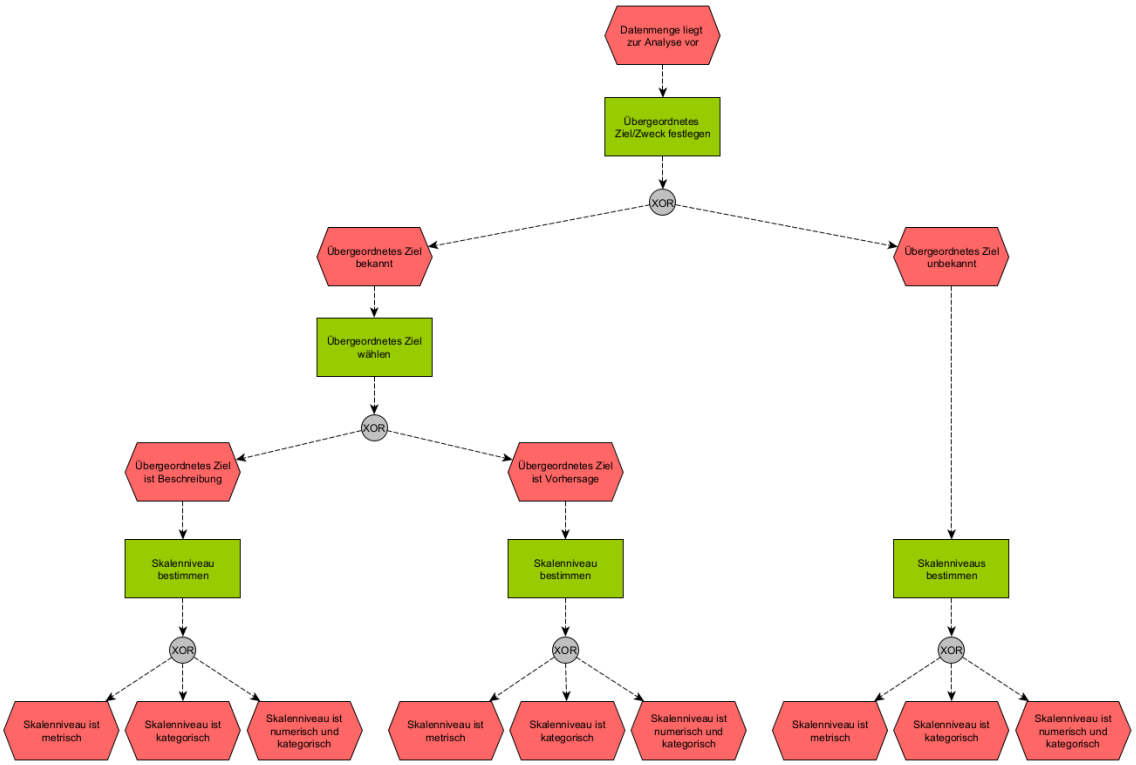


Abbildung 3-6: Systematisierung in Klassifikatoren: Übergeordnetes Ziel und Skalenniveau

Durch die Klassifikatoren Zielsetzung und Skalenniveau sind nun insgesamt neun Verzweigungen der EPK entstanden. In allen Verzweigungen folgt nun der Klassifikator Datenvorverarbeitung, welcher in Abbildung 3-7 dargestellt ist. Bei diesem Klassifikator wird entschieden, ob eine Datenvorverarbeitung durchgeführt wird, beziehungsweise ob Anwender über eine in Unterkapitel 2.4.2 beschriebene Datenkompetenz verfügen, oder nicht.

Der Schritt der Datenvorverarbeitung hat durch die Möglichkeit einer im Unterkapitel 2.2.4 beschriebene Datentransformation unmittelbaren Einfluss auf die im nächsten Schritt wählbaren Data-Mining-Verfahren, da diese vom Skalenniveau der Attribute abhängig sind. Welche Data-Mining-Verfahren welches Skalenniveau voraussetzen wird in Kapitel 2.3.1 verdeutlicht. Eine Darstellung der EPK als Gesamtübersicht mit allen wählbaren Klassifikatoren ist an dieser Stelle nicht möglich. Die Gesamtübersicht ist dem Anhang A (EPK_Detailkonzept-Phase-1.png) zu entnehmen.

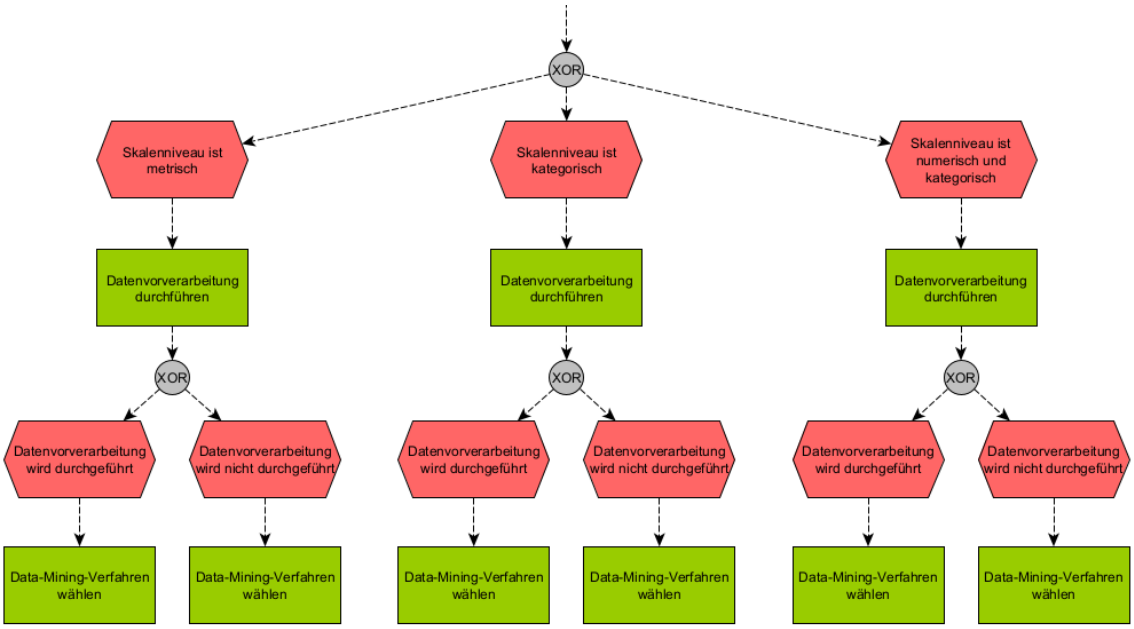


Abbildung 3-7: Systematisierung in Klassifikatoren: Datenvorverarbeitung

Abbildung 3-8 zeigt die wählbaren Data-Mining-Verfahren, wenn es sich beim übergeordneten Ziel um eine Beschreibung des Datensatzes handelt (in der Gesamtübersicht Anhang A: EPK_Detailkonzept-Phase-1.png, Raster B).

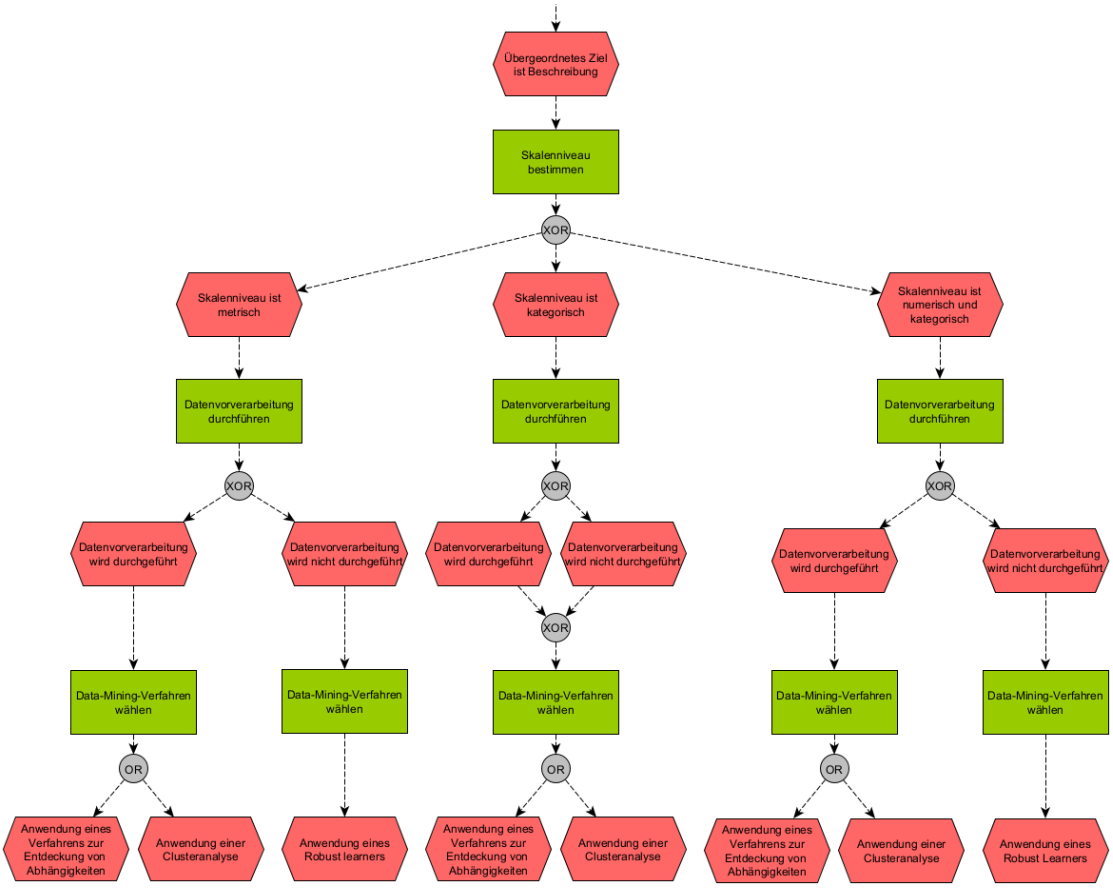


Abbildung 3-8: Systematisierung in Klassifikatoren: Auswahl der Verfahren mit übergeordnetem Ziel Beschreibung

Bei einem metrischen Skalenniveau und der Durchführung einer Datenvorverarbeitung können Verfahren zur Entdeckung von Abhängigkeiten oder eine Clusteranalyse angewendet werden. Wird keine Datenvorverarbeitung durchgeführt, kann lediglich ein Robust Learner auf den Datensatz angewendet werden. Die Gründe dafür und die Vorteile eines Robust Learners werden in Unterkapitel 2.2.2 näher beschrieben. Für eine effektive Anwendung eines Verfahrens zur Entdeckung von Abhängigkeiten oder einer Clusteranalyse muss der Datensatz vorher bereinigt bzw. mithilfe von Datentransformationen das Skalenniveau bearbeitet werden. Die Gründe dafür und Methoden dazu werden in Unterkapitel 2.2 erläutert.

Liegt ein kategorisches Skalenniveau vor, hat die Durchführung einer Datenvorverarbeitung lediglich Auswirkungen auf die Performance, nicht aber auf die Auswahlmöglichkeiten der Verfahren. In beiden Fällen kann ein Verfahren zur Entdeckung von Abhängigkeiten und eine Clusteranalyse auf den Datensatz angewendet werden. Wie in Unterkapitel 2.2 erläutert, liefern die Verfahren jedoch bei schlecht vorbereiteten Daten keine guten Ergebnisse. Beide Verfahren können aber auf einen rein kategorischen Datensatz angewendet werden.

Handelt es sich um einen Datensatz mit kategorischen und numerischen Attributen, kann ohne Datenvorverarbeitung lediglich ein Robust Learner angewendet werden. Dabei ist jedoch mit wenig aussagekräftigen Ergebnissen zu rechnen. Die Gründe dafür werden in Unterkapitel 2.2 ausführlich erläutert.

Da eine Betrachtung der EPK an dieser Stelle im Ganzen nicht möglich ist, zeigt Abbildung 3-9 beispielhaft die wählbaren Verfahren, wenn es sich beim übergeordneten Ziel um eine Vorhersage handelt (siehe Anhang A: EPK_Detailkonzept-Phase-1.png, Raster C).

Wird keine Datenvorverarbeitung durchgeführt, liegt ein Datensatz mit sowohl kategorischen als auch numerischen Attributen vor der unter Umständen stark verrauscht ist. In diesem Fall kann lediglich ein Robust Learner auf den Datensatz angewendet werden. Die Gründe dafür werden in den vorherigen Absätzen und in Unterkapitel 2.2 sowie 2.3.1 ausführlich erläutert.

Wird eine Datenvorverarbeitung durchgeführt, besteht die Möglichkeit, in Unterkapitel 2.2.4 erläuterte Datentransformationen durchzuführen und die Attribute in rein numerische oder rein kategorische Attribute umzuwandeln. Bei rein numerischen Attributen können eine numerische Vorhersage oder die Anwendung einer Clusteranalyse und das anschließende Ableiten von Regeln auf den Datensatz angewendet werden. Bei rein kategorischen Attributen können die Anwendung eines Verfahrens zur Entdeckung von Abhängigkeiten und das anschließende Ableiten von Regeln oder eine Klassifizierung auf den Datensatz angewendet werden. Die Verfahren werden in Unterkapitel 2.3.1 näher erläutert.

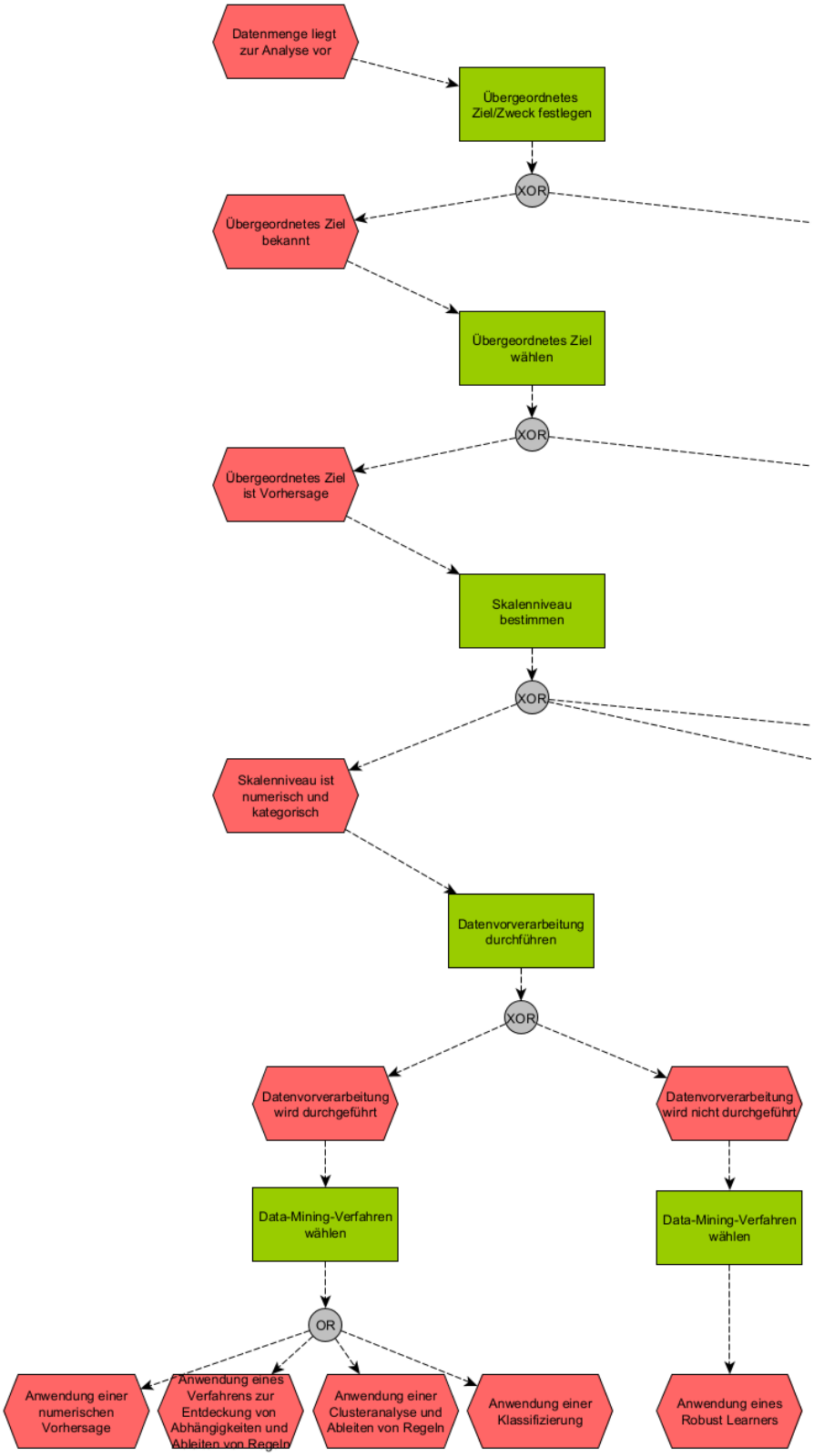


Abbildung 3-9: Systematisierung in Klassifikatoren: Beispielhafter Ausschnitt der EPK mit übergeordnetem Ziel Vorhersage und gemischtem Skalenniveau

3.3.3.2 Detailkonzept Phase 2: Integration von Unterklassifikatoren in die Datenvorverarbeitung

Durch die Klassifikatoren Zielsetzung und Skalenniveau sind die der Abbildung 3-6 zu entnehmenden neun Verzweigungen der EPK entstanden. In allen Verzweigungen folgt nun die Datenvorverarbeitung. Unabhängig vom Skalenniveau und vom übergeordneten Ziel können Anwender nun entscheiden, ob eine Datenvorverarbeitung durchgeführt wird oder nicht. In Phase 1 des Detailkonzepts werden Anwendern in der EPK nun die Data-Mining-Verfahren angezeigt, die auf den Datensatz anwendbar sind. Der Klassifikator der Datenvorverarbeitung wird in Phase 2 um die in RapidMiner in Unterkapitel 3.2.2 identifizierten Unterklassifikatoren erweitert. Des Weiteren werden die Unterklassifikatoren optimiert.

Wird eine Datenvorverarbeitung durchgeführt, unterstützen die in *Auto Cleansing* identifizierten und optimierten Unterklassifikatoren in RapidMiner Anwender dabei, die Datenvorverarbeitung durchzuführen. Eine EPK dieses Prozesses ist in Abbildung 3-10 dargestellt. Dazu wird zuerst die Qualität der Daten bestimmt, also auf fehlende Werte oder Spalten mit mehreren identischen Werten geprüft.

Wenn Spalten mit mehreren identischen Werten vorhanden sind, wird die von RapidMiner definierte Kennzahl *Stability* betrachtet. Ist *Stability* größer als 90%, wird das Attribut entfernt, ansonsten bleibt es erhalten. Der von RapidMiner definierte und verwendete Schwellenwerte für *Stability* wird in das Konzept übernommen, da dieser Schwellenwert, wie in Unterkapitel 3.2.2 festgestellt wird, auch in anderen wissenschaftlichen Arbeiten verwendet wird und dort funktioniert.

Wenn fehlende Werte vorhanden sind, ist es entscheidend, ob diese Werte zufällig fehlen, oder ob ein Muster erkennbar ist. Dies wird in Unterkapitel 2.2.2 dargestellt. Aus diesem Grund wird ein Unterklassifikator entwickelt, der Anwender beim Umgang mit fehlenden Werten mehr unterstützt als dies in der Turbo Prep von RapidMiner der Fall ist. Dazu ist es sinnvoll, die Kategorie der fehlenden Werte zu bestimmen. Die unterschiedlichen Kategorien fehlender Werte und wie mit diesen umgegangen werden kann wird ebenfalls in Unterkapitel 2.2.2 genauer erläutert. Ist ein Muster zu erkennen, wird, im Optimalfall mithilfe der Datenquelle, der Grund für das Fehlen der Werte ermittelt. Wenn die fehlenden Werte geschätzt werden können, wird daraufhin ein Modell entwickelt, welches die fehlenden Werte schätzt. Können die fehlenden Werte nicht geschätzt werden, wird das Attribut komplett aus dem Datensatz entfernt. Ist kein Muster bei den fehlenden Werten zu erkennen, wird die Kennzahl *Missing* bestimmt. Ist *Missing* kleiner als 10%, wird nicht das Attribut gelöscht, sondern nur die Datensätze, die fehlende Werte enthalten. Bei einem Anteil von unter 10% ist dies problemlos möglich, da nach Entfernen der Datensätze mit fehlenden Werten noch genug Datensätze für eine Analyse vorhanden sind. Liegt *Missing* zwischen 10% und 70%, können die fehlenden Werte durch einfache Verfahren ersetzt werden. Anwendbare einfache Lösungsmöglichkeiten und Verfahren zum Umgang mit fehlenden Werten werden ebenfalls in Unterkapitel 2.2.2 näher erläutert. Bei einem Anteil fehlender Werte von über 70% wird das Attribut gelöscht. Der Schwellenwert von 70% zum Entfernen eines Attributs kommt auch in RapidMiner in der Funktion *Turbo Prep* bei der von RapidMiner verwendeten Kennzahl *Missing* zum Einsatz. Dies wird in Unterkapitel 3.2.2 näher erläutert. In diesem Unterkapitel wird auch ausgeführt, dass es in der Literatur verschiedene Ansätze und

Meinungen dazu gibt, wann ein Attribut zu viele fehlende Werte aufweist und entfernt werden sollte. Wie dort erwähnt, wird in der Literatur häufig die Meinung vertreten, dass es vom später anzuwendenden Data-Mining-Verfahren abhängig gemacht werden sollte, wie mit den fehlenden Werten umzugehen ist (Pratama et al. 2016). Um auch Anwender mit eingeschränkter Datenkompetenz bei der Entscheidungsfindung im Umgang mit fehlenden Werten zu unterstützen, wird an dieser Stelle der von RapidMiner genutzte Schwellenwert von 70% zum Entfernen eines Attributs bei der Kennzahl *Missing* in das Konzept übernommen.

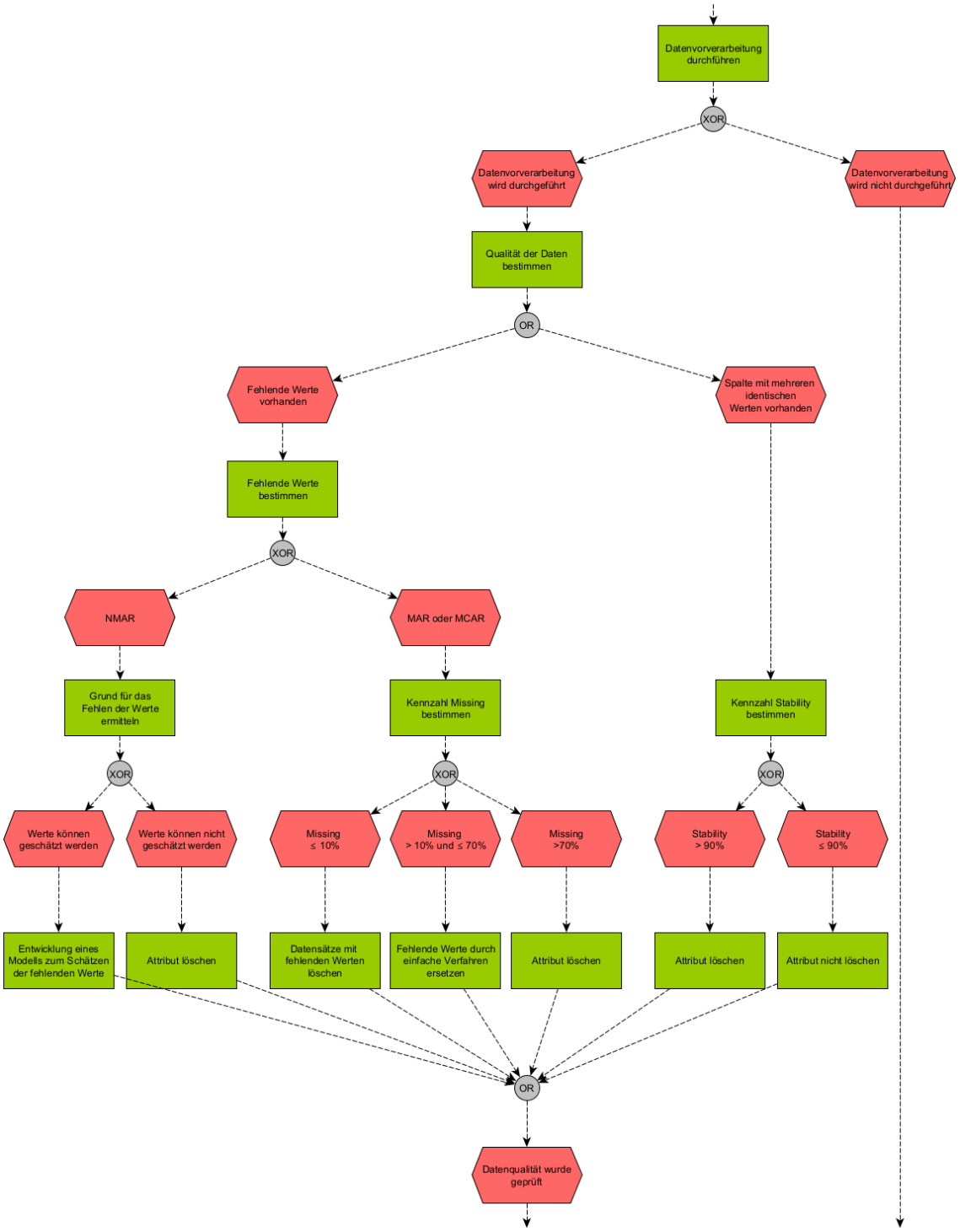


Abbildung 3-10: Systematisierung in Klassifikatoren: Datenvorverarbeitung mit Unterklassifikatoren

Nachdem die vom Skalenniveau unabhängige Datenvorverarbeitung durchgeführt wurde, wird eine vom Skalenniveau abhängige Vorverarbeitung durchgeführt.

Handelt es sich um ein metrisches Skalenniveau, sieht die Datenvorverarbeitung wie in Abbildung 3-11 dargestellt aus. Hier besteht die Möglichkeit, alle Attribute in kategorische Werte umzuwandeln oder das Skalenniveau nicht zu ändern. Wird das Skalenniveau nicht geändert und bleibt metrisch, kann bei Bedarf eine PCA oder Normierung angewendet werden. Nach diesem optionalen Schritt wird ein Data-Mining-Verfahren gewählt. Bei einem metrischen Skalenniveau besteht ebenfalls die Möglichkeit, alle Attribute in kategorische Werte umzuwandeln. Für kategorische Werte ist keine weitere Bearbeitung vorgesehen. Auch nach der Umwandlung in kategorische Werte wird ein Data-Mining-Verfahren gewählt.

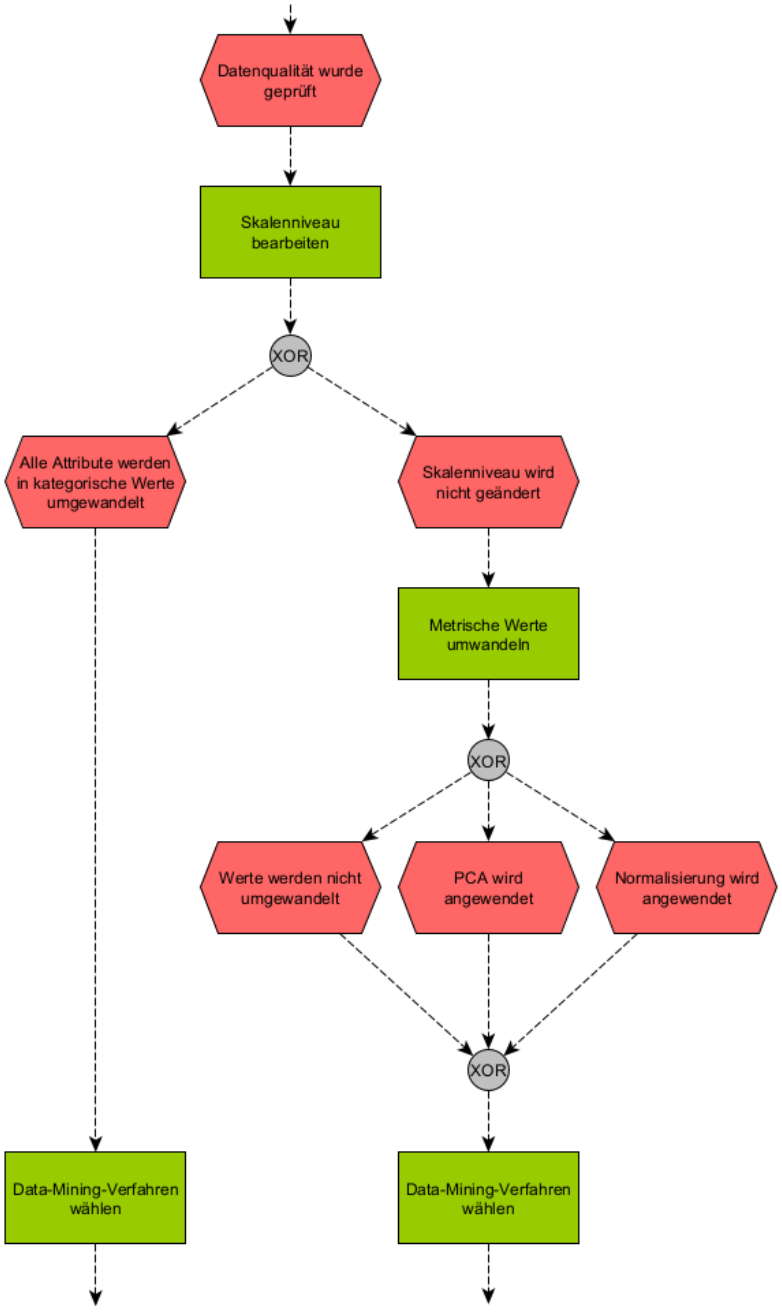


Abbildung 3-11: Systematisierung in Klassifikatoren: Skalenniveau bearbeiten bei metrischen Attributen mit Unterklassifikatoren

Neben einem rein metrischen Skalenniveau kann auch, wie in Abbildung 3-12 dargestellt, ein rein kategorisches Skalenniveau vorliegen. In diesem Fall werden entweder alle Attribute kategorisch belassen und direkt ein Data-Mining-Verfahren gewählt, oder alle Attribute werden in metrische Werte umgewandelt. Werden die Attribute in metrische Werte umgewandelt, kann bei Bedarf auf diese eine PCA oder Normalisierung angewendet werden. Es besteht auch die Möglichkeit, die Attribute nach der Umwandlung in metrische Werte nicht weiter zu behandeln. Nach diesem optionalen Schritt wird ein Data-Mining-Verfahren gewählt.

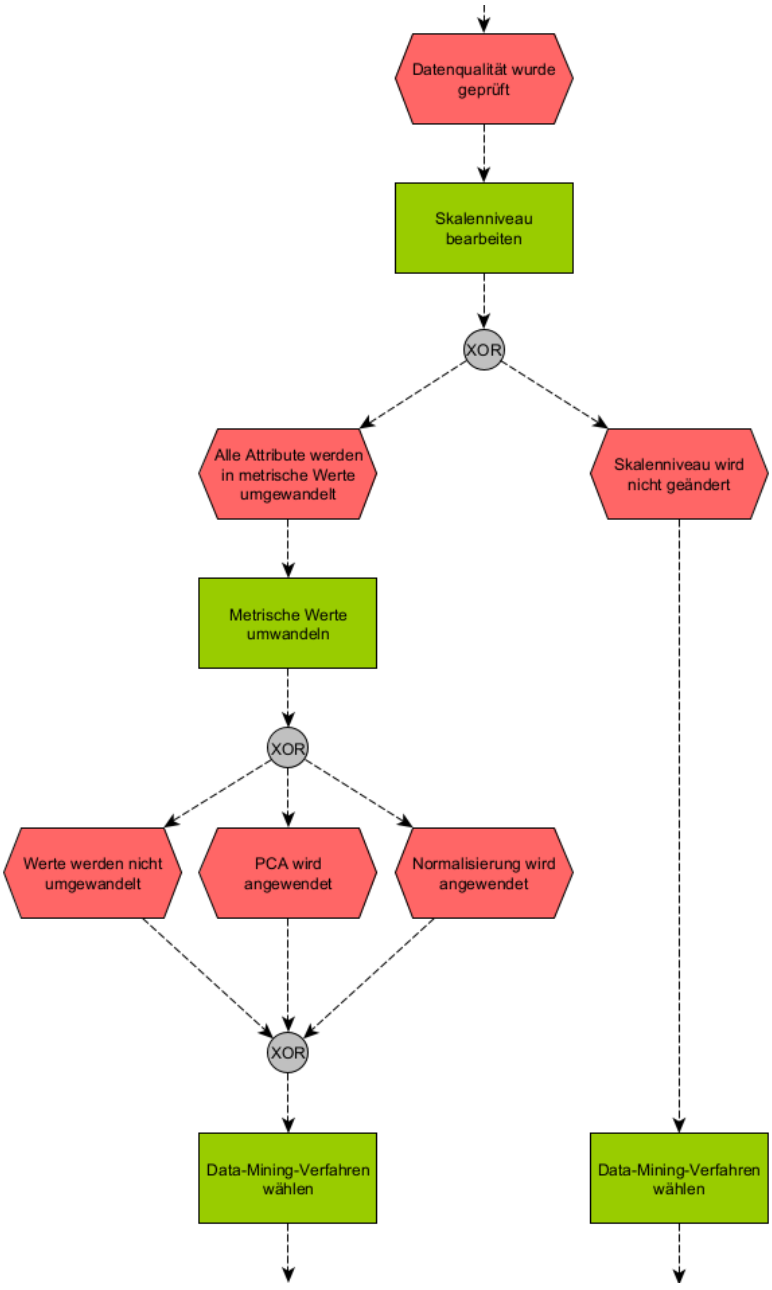


Abbildung 3-12: Systematisierung in Klassifikatoren: Skalenniveau bearbeiten bei kategorischen Attributen mit Unterklassifikatoren

Neben einem rein metrischen oder kategorischen Skalenniveau, kann auch ein gemischtes Skalenniveau vorliegen. Dies ist in Abbildung 3-13 dargestellt. In diesem Fall werden die Attribute gemischt gelassen, in ein rein metrisches oder in ein rein kategorisches Skalenniveau umgewandelt. Werden die Werte wie ursprünglich belassen, wird direkt mit dem Schritt Auswahl

eines Data-Mining-Verfahren fortgefahren. Werden alle Attribute in kategorische Attribute umgewandelt, wird ebenfalls direkt mit dem Schritt ein Data-Mining-Verfahren zu wählen fortgefahren. Wenn alle Attribute in metrische Werte umgewandelt werden, kann auf diese bei Bedarf eine PCA oder eine Normalisierung angewendet werden. Es besteht auch die Möglichkeit, die umgewandelten Werte nicht weiter zu bearbeiten. Danach wird ein Data-Mining-Verfahren gewählt.

Wie bei Betrachtung der gesamten EPK in Anhang A (EPK_Detailkonzept-Phase-2.png) deutlich wird, unterscheiden sich die wählbaren Data-Mining-Verfahren je nach gewählten Klassifikatoren und Unterklassifikatoren.

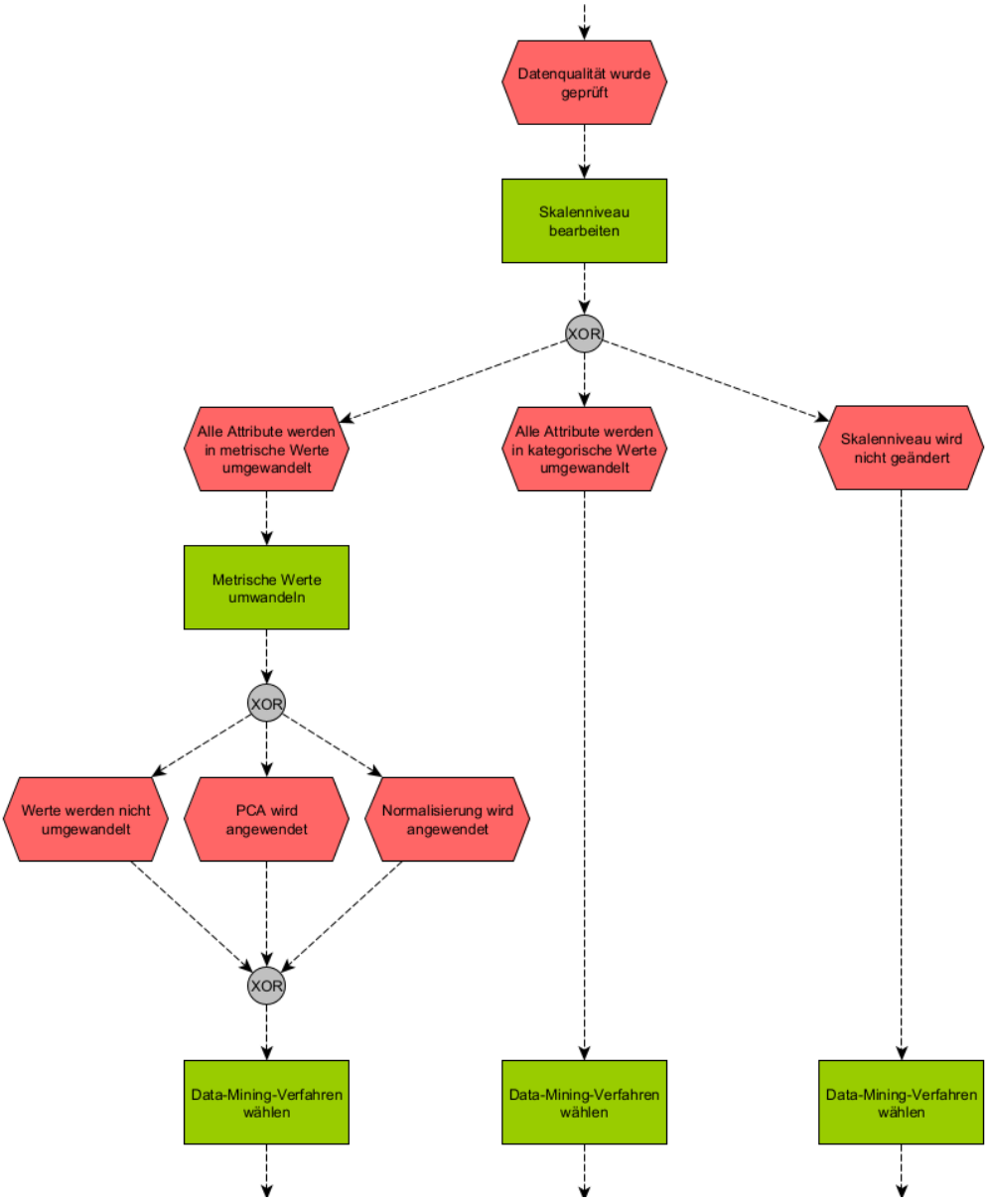


Abbildung 3-13: Systematisierung in Klassifikatoren: Skalenniveau bearbeiten bei kategorischen und metrischen Attributen mit Unterklassifikatoren

Eine Abbildung der gesamten EPK ist an dieser Stelle nicht möglich. Aus diesem Grund soll im Folgenden beispielhaft die Betrachtung eines Ausschnittes erfolgen. Die gesamte EPK mit integrierten Unterklassifikatoren kann Anhang A (EPK_Detailkonzept-Phase-2.png) entnommen werden. In dem in Abbildung 3-14 dargestellten Ausschnitt ist das übergeordnete Ziel mit einer

Vorhersage bekannt und das Skalenniveau ist mit sowohl numerischen als auch kategorischen Attributen gemischt (siehe Anhang A: EPK_Detailkonzept-Phase-2.png, Raster A2.3). Anwender haben nun die Möglichkeit, sich für das Durchführen einer Datenvorverarbeitung oder dagegen zu entscheiden.

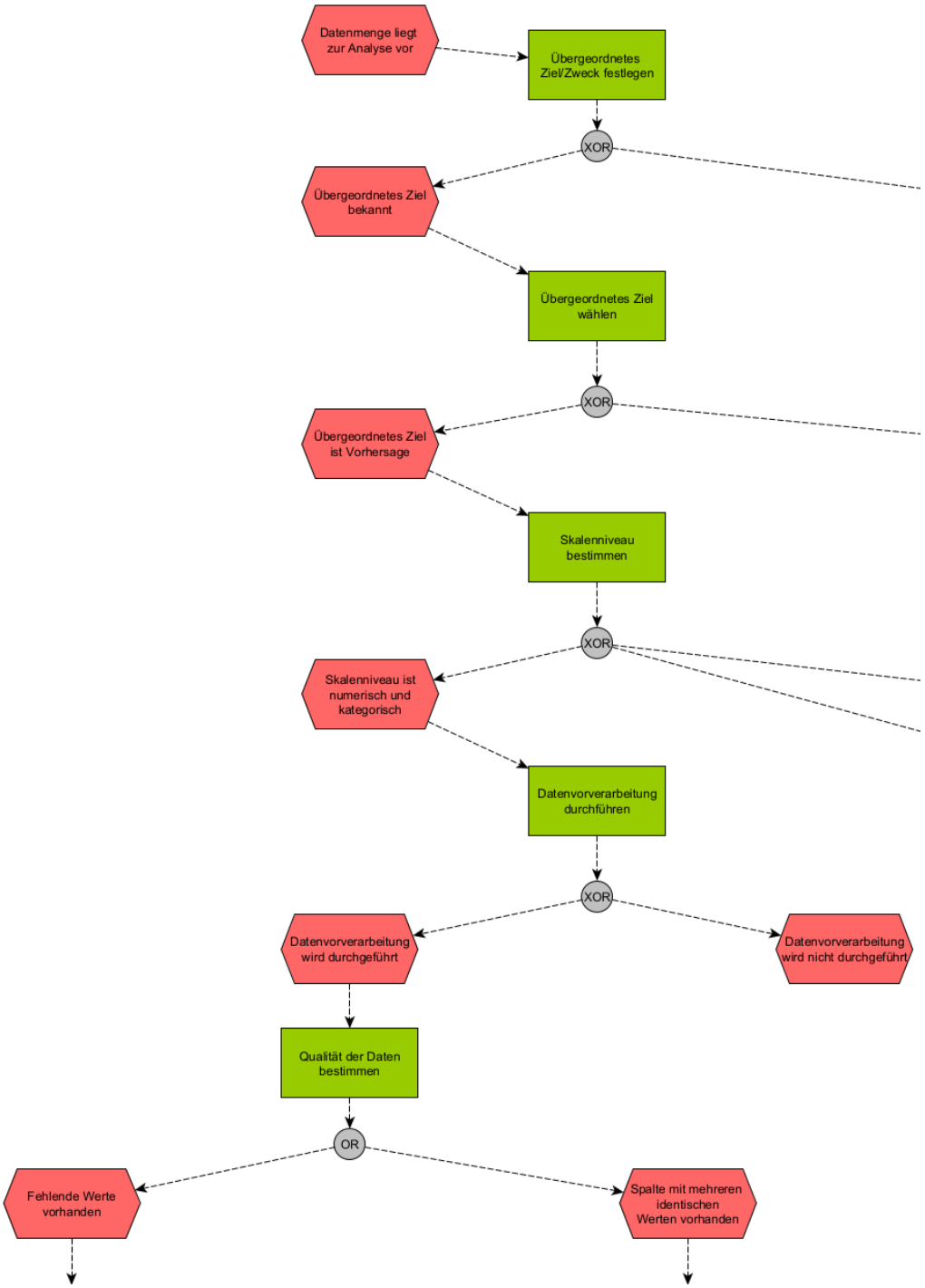


Abbildung 3-14: Systematisierung in Klassifikatoren: Beispielhafter Ausschnitt der EPK erster Teil

Wird keine Datenvorverarbeitung durchgeführt, kann direkt ein Data-Mining-Verfahren gewählt werden, wie in der EPK Abbildung 3-15 zu erkennen ist. Bei einem gemischten Skalenniveau und dem Ziel einer Vorhersage, kann jedoch lediglich ein Robust Learner auf den Datensatz angewendet werden. Die Gründe dafür werden in Unterkapitel 2.2.2 erläutert.

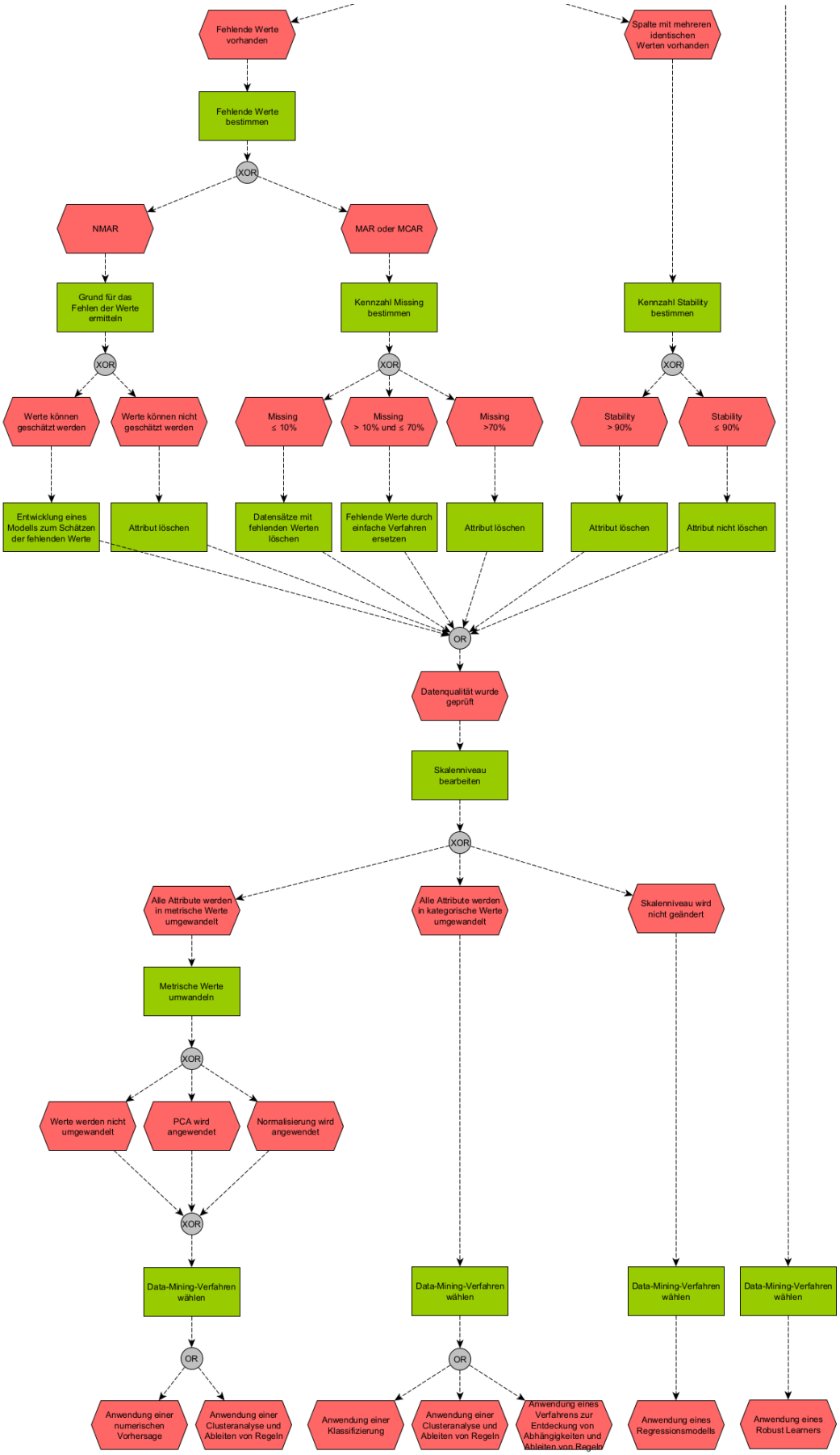


Abbildung 3-15: Systematisierung in Klassifikatoren: Beispielhafter Ausschnitt der EPK zweiter Teil

Anwender können so direkt erkennen, dass eine Datenvorverarbeitung erfolgen muss, wenn ein bestimmtes Data-Mining-Verfahren zur Anwendung kommen soll. Entscheidet man sich also für die Durchführung einer Datenvorverarbeitung beginnt diese damit, die Qualität der Daten zu bestimmen. Was in diesem Schritt geschieht, wird bereits zu Beginn dieses Unterkapitels erläutert.

Nachdem die Datenqualität geprüft wurde, wird das Skalenniveau bearbeitet, indem alle Attribute in metrische oder kategorische Attribute umgewandelt werden. Alternativ bleibt das Skalenniveau unverändert. Wird das Skalenniveau nicht bearbeitet, kann ein Regressionsmodell auf den Datensatz angewendet werden, das ein gemischtes Skalenniveau verarbeiten kann. Beispiele für solche Regressionsmodelle sind Data-Mining-Verfahren wie Random-Forest oder Deep-Learning-Algorithmen. Entscheiden sich Anwender dafür, alle Attribute in rein kategorische Attribute umzuwandeln, wird im nächsten Schritt aus drei Data-Mining-Verfahren gewählt: Anwendung einer Klassifizierung, Anwendung einer Clusteranalyse und Ableiten von Regeln und Anwendung eines Verfahrens zur Entdeckung von Abhängigkeiten und Ableiten von Regeln. Entscheiden sich Nutzer dazu, alle Attribute in numerische Attribute umzuwandeln, besteht danach noch die Möglichkeit, auf die numerischen Werte eine PCA oder eine Normalisierung anzuwenden. Die Anwendung dieser Verfahren hat jedoch keinen Einfluss auf die Auswahl der Data-Mining-Verfahren. Diese Datentransformationen haben lediglich einen Einfluss auf die Performance, nicht auf die generelle Anwendbarkeit. Die Gründe dafür wurden in Unterkapitel 2.2 erläutert. Auf rein numerische Attribute können Verfahren in Form der Anwendung einer numerischen Vorhersage oder Anwendung einer Clusteranalyse und dem anschließenden Ableiten von Regeln angewendet werden.

In RapidMiner werden in der Funktion *Auto Cleansing* folgende drei Klassifikatoren identifiziert:

- Zielattribut definieren
- Datenqualität bestimmen
- Skalenniveau bearbeiten

Anhand der in Unterkapitel 3.3.1 definierten Anforderungen werden folgende eigene Klassifikatoren definiert:

- Ziel/Zweck der Analyse
- Skalenniveau bestimmen
- Datenvorverarbeitung durchführen

Um die definierten Anforderungen zu erfüllen und damit auch der in Unterkapitel 2.4.2 identifizierten Herausforderung der Datenkompetenz zu begegnen, werden die in RapidMiner identifizierten Unterklassifikatoren zur Bestimmung der Datenqualität mithilfe der Kennzahlen *Missing* und *Stability* in den Klassifikator der Datenvorverarbeitung integriert und optimiert. Dabei wird auch der Umgang mit fehlenden Werten aufgrund der in Unterkapitel 2.2.2 erläuterten Kriterien optimiert.

Über eine Betrachtung des vorhandenen Datenbestandes kann mithilfe der EPK schon vor Beginn der eigentlichen Analyse untersucht werden, welche Verfahren durch welche

Bearbeitungsschritte auf die Daten angewendet werden können. Durch die Unterklassifikatoren in der Datenvorverarbeitung wird es auch Anwendern ohne starke Datenkompetenz ermöglicht, eine grundlegende Datenvorverarbeitung durchzuführen. Bei der Erarbeitung der für die Datenvorverarbeitung notwendigen Schritte wird deutlich, dass es dabei Verfahren und Transformationen gibt, die unabhängig vom eingesetzten Data-Mining-Verfahren durchgeführt werden können und solche die erforderlich sind, um ein Verfahren überhaupt erst zu ermöglichen. Die Verfahren, die unabhängig vom eingesetzten Verfahren angewendet werden können, sollten auch durchgeführt werden. Die Gründe dafür werden unter anderem in den Unterkapiteln 2.2.2 und 2.2.3 erläutert. Mithilfe der Klassifikatoren kann schon früher erkannt werden, welche verfahrensabhängigen Vorbereitungsschritte durchgeführt werden müssen. So können Iterationen innerhalb des Data-Mining-Prozesses zwischen dem Data Mining selbst und den vorbereitenden Schritten vermieden werden.

4 Prototypische Anwendung und Validierung der Klassifikatoren

In diesem Kapitel wird das in Kapitel 3 entwickelte Klassifizierungskonzept auf die in Unterkapitel 3.1.2 beschriebenen Beispieldatensätze angewendet. Anschließend werden die Klassifikatoren validiert.

4.1 Prototypische Implementierung des Klassifikationskonzepts in RapidMiner

In den folgenden beiden Unterkapiteln wird das Klassifizierungskonzept zuerst auf den Walmart und anschließend auf den Rossmann Datensatz angewendet. Zur Implementierung des Klassifikationskonzepts dient die Software RapidMiner.

4.1.1 Exemplarische Anwendung auf den Walmart Datensatzes

Der Walmart Datensatz wird in Unterkapitel 3.1.1 beschrieben. Insgesamt liegen 16 Attribute vor, von denen zwei kategorisch und 14 numerisch sind. Für die Analyse wird angenommen, dass kein übergeordnetes Ziel bekannt ist.

Die Analyse beginnt wie in Abbildung 4-1 dargestellt mit dem Klassifikator Ziel/Zweck der Analyse. Bei dieser Analyse wird angenommen, dass kein Ziel vorhanden ist, der Klassifikator wird also entsprechend gewählt. Als nächster Schritt folgt der Klassifikator Skalenniveau. Bei diesem Datensatz liegen sowohl metrische als auch kategorische Attribute, also ein gemischtes Skalenniveau vor. Im kommenden Schritt folgt der Klassifikator Datenvorverarbeitung. Bei diesem Klassifikator ist unmittelbar erkennbar, dass bei Nichtdurchführung der Datenvorverarbeitung lediglich ein Robust Learner auf den Datensatz angewendet werden kann. Um eine größere Auswahl an anwendbaren Verfahren zu haben, muss also eine Datenvorverarbeitung durchgeführt werden. Die Gründe dafür werden in Unterkapitel 2.2 erläutert.

Für die Datenvorverarbeitung werden die in Unterkapitel 3.3.3.2 integrierten Unterklassifikatoren angewendet. Dazu wird die Qualität der Daten bestimmt, indem die fehlenden Werte genauer untersucht werden und anschließend mit Hilfe der entwickelten Kennzahlen *Missing* und *Stability* über den Umgang mit Attributen, die identische oder fehlende Werte enthalten, entschieden wird. Die Unterklassifikatoren werden in Unterkapitel 3.3.3.2 genau erläutert und können Abbildung 3-10 entnommen werden.

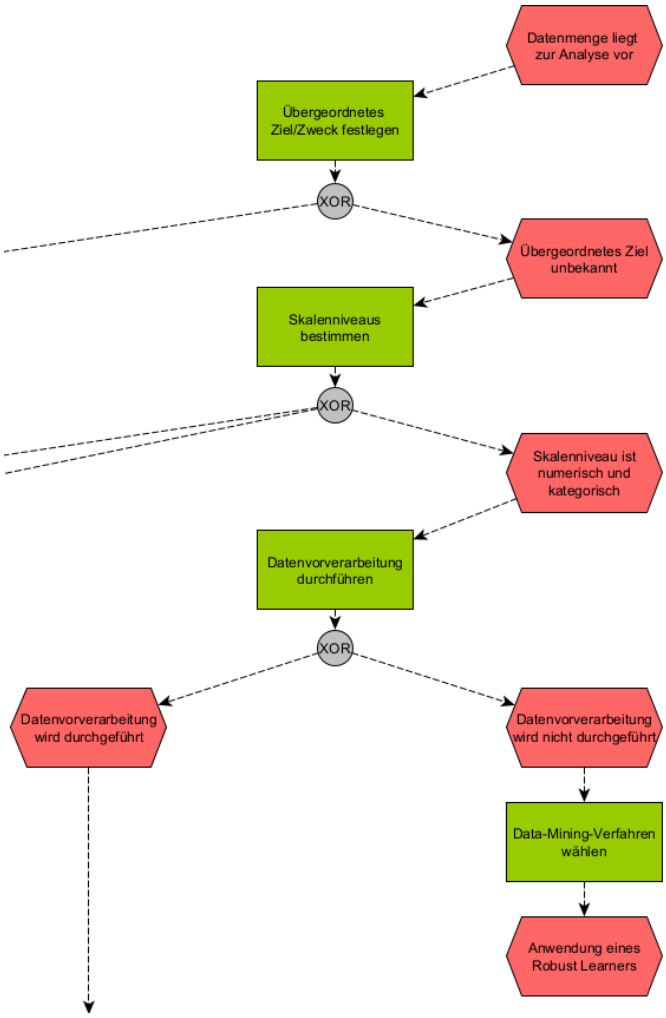


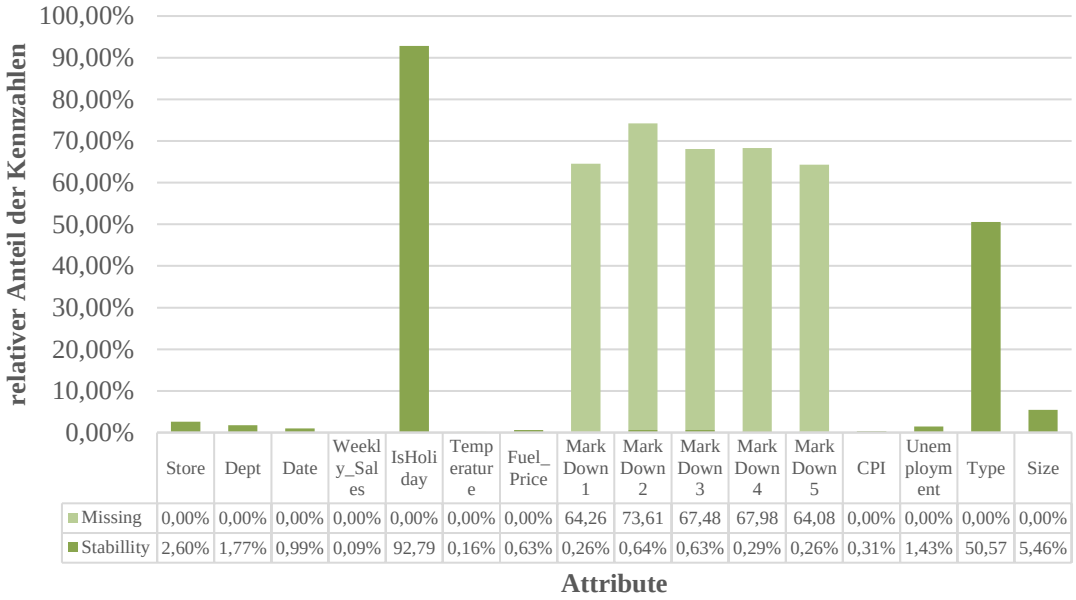
Abbildung 4-1: Anwendung der Klassifikatoren Ziel/Zweck, Skalenniveau und Datenvorverarbeitung auf den Walmart Datensatz

Zu Beginn der Datenvorverarbeitung wird die Qualität der Daten mithilfe der Unterklassifikatoren bestimmt. Dies geschieht in RapidMiner mithilfe des Operators Quality Measures. Dieser Operator bestimmt verschiedene Kennzahlen eines Datensatzes, unter anderem die für diese Arbeit relevanten *Stability* und *Missing*. Die beiden Kennzahlen *Missing* und *Stability* sind in Abbildung 4-2 als Balkendiagramm dargestellt.

Zuerst wird die Kennzahl *Stability* bei allen Attributen betrachtet. Dabei fällt auf, dass die Attribute *IsHoliday* und *Type* die höchsten Werte aufweisen. Das Attribut *Type* bleibt mit einem Prozentsatz von 51% unter dem definierten Schwellenwert von 90% und wird somit nicht gelöscht. Das Attribut *IsHoliday* hat einen *Stability* Wert von 93% und wird deshalb bei Durchführung der Datenvorverarbeitung aus dem Datensatz entfernt.

Bevor die Kennzahl *Missing* genau betrachtet wird, wird geprüft, ob es sich bei den fehlenden Werten um den Typ NMAR, MAR oder MCAR handelt. Bei Fehlern der Kategorie NMAR handelt es sich um fehlende Werte, bei denen das Fehlen vom Wert selbst und zusätzlich möglicherweise auch von anderen Werten abhängig ist. Fehler diese Kategorie dürfen nicht ignoriert werden. Bei Fehlern der Kategorie MAR oder MCAR ist zwar möglicherweise ein Muster erkennbar, welches das Fehlen der Daten erklärt, das Fehlen der Werte ist aber von anderen Werten im Datensatz abhängig und nicht vom Wert selbst. Fehler dieser beiden

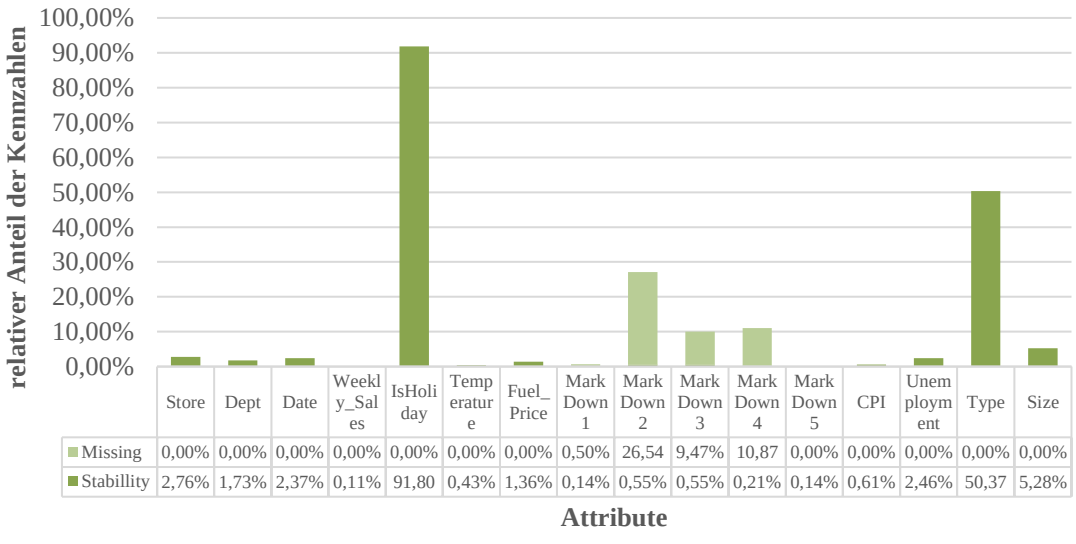
Kategorien können ignoriert und die Werte durch einfache Verfahren ersetzt werden. Die Kategorien und einfache Verfahren zum Ersetzen fehlender Werte wurden in Unterkapitel 2.2.2 genauer erläutert.



	Store	Dept	Date	Weekly_Sales	IsHoliday	Temperature	Fuel_Price	MarkDown1	MarkDown2	MarkDown3	MarkDown4	MarkDown5	CPI	Unemployment	Type	Size
Missing	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	64,26	73,61	67,48	67,98	64,08	0,00%	0,00%	0,00%	0,00%
Stabillity	2,60%	1,77%	0,99%	0,09%	92,79	0,16%	0,63%	0,26%	0,64%	0,63%	0,29%	0,26%	0,31%	1,43%	50,57	5,46%

Abbildung 4-2: Relevante Qualitätskennzahlen Walmart Datensatz

Bei Betrachtung der Abbildung 4-2 fällt auf, dass die Attribute MarkDown1-5 als einzige Attribute fehlende Werte aufweisen. Bei der Erläuterung der Attribute in Unterkapitel 3.1.1 wurde erwähnt, dass die Daten über die Sonderangebotsabschläge erst seit November 2011 und nicht immer für alle Filialen verfügbar sind. Bei den fehlenden Werten vor November 2011 handelt es sich also um die in Unterkapitel 2.2.2 genauer erläuterte Kategorie MAR, denn das Fehlen der Werte ist abhängig vom Attribut Datum und nicht von der Variable selbst. Um zu untersuchen, welche Eigenschaften die Daten nach diesem Zeitraum aufweisen, wird der Datensatz ab November 2011 zusätzlich einzeln betrachtet. Eine Übersicht zu den Qualitätskennzahlen ab diesem Zeitraum ist Abbildung 4-3 zu entnehmen.



	Store	Dept	Date	Weekly_Sales	IsHoliday	Temperature	Fuel_Price	MarkDown1	MarkDown2	MarkDown3	MarkDown4	MarkDown5	CPI	Unemployment	Type	Size
Missing	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,50%	26,54	9,47%	10,87	0,00%	0,00%	0,00%	0,00%	0,00%
Stabillity	2,76%	1,73%	2,37%	0,11%	91,80	0,43%	1,36%	0,14%	0,55%	0,55%	0,21%	0,14%	0,61%	2,46%	50,37	5,28%

Abbildung 4-3: Relevante Qualitätskennzahlen Walmart Datensatz ab November 2011

Ab diesem Zeitraum haben die Attribute MarkDown1 0,5%, MarkDown2 26,5%, MarkDown3 9,5%, MarkDown4 10,9% und MarkDown5 0% fehlende Werte. Die Kennzahl *Stability* der einzelnen Attribute hat sich nur geringfügig verändert. Um einen genaueren Eindruck der Sonderangebotsabschläge in Form der Attribute MarkDown1-5 zu erhalten, ist es sinnvoll, diese grafisch darzustellen. Eine grafische Darstellung dieser Attribute ist Abbildung 4-4 zu entnehmen.

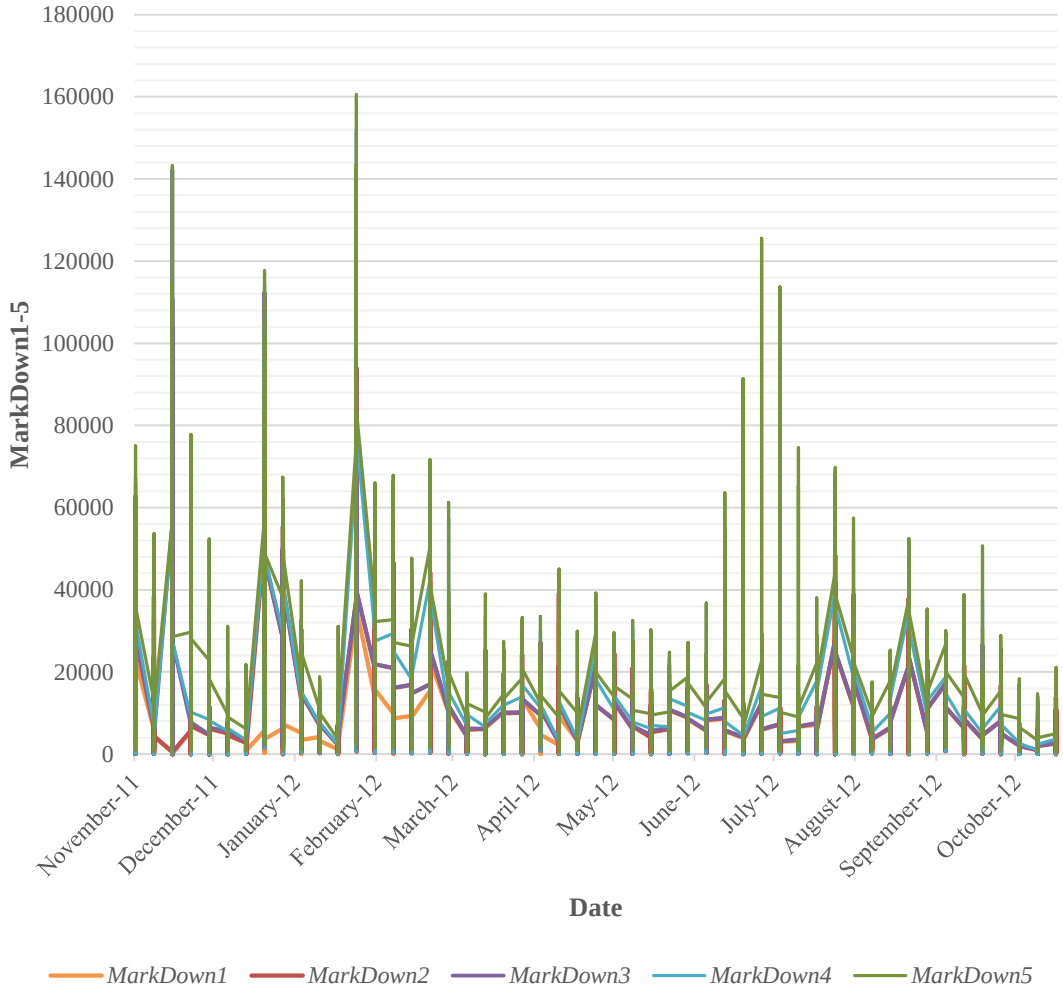


Abbildung 4-4: Grafische Darstellung der Attribute *MarkDown1-5*

Bei der Beschreibung des Datensatzes in Unterkapitel 3.1.2 wird erklärt, dass vor Feiertagen Preisreduzierungen stattfinden und die Wochen der Feiertage fünfmal höher gewichtet werden als Wochen ohne Feiertage. Die Ausschläge im November 2011 können mit dem Feiertag Thanksgiving am 24. November 2011 erklärt werden. Die Anstiege Ende Dezember und Anfang Januar 2011 können mit den Weihnachtsfeiertagen und Silvester in Verbindung gebracht werden. Der Anstieg Mitte Januar ist wahrscheinlich auf den Martin Luther King Day zurückzuführen der 2012 auf den 16. Januar fiel. Die Ausschläge im Februar sind durch den Super Bowl am 06. Februar 2012 verursacht. Der Anstieg im Juli 2012 kann mit dem Unabhängigkeitstag am 04. Juli erklärt werden.

Von besonderer Bedeutung ist jedoch, ob das Fehlen von Werten vom Wert selbst abhängig ist. Bei einem analytischen Blick auf die Daten fällt auf, dass besonders viele Werte von *MarkDown2* und *MarkDown4* bei der Ausprägung *C* des Attributs *Store* fehlen. Bei Filialen vom

Typ *C* handelt es sich um die kleinste Kategorie der Filialen des Unternehmens. In der Beschreibung wird auch erwähnt, dass Daten über die Sonderangebotsabschläge, also auch die Attribute *MarkDown2* und *MarkDown4*, nicht immer zur Verfügung stehen. Es ist hier also ein Muster indiziert, wann Werte im Datensatz fehlen, das Fehlen der Werte ist allerdings abhängig vom anderen Werten im Datensatz und nicht vom Wert selbst. Deshalb handelt es sich, wie in Unterkapitel 2.2.2 näher erläutert, um fehlende Werte der Kategorie MAR. Zur weiteren Entscheidungsfindung im Umgang mit den fehlenden Werten wird also die Kennzahl *Missing* herangezogen. Ist *Missing* kleiner oder gleich 10%, werden die Datensätze des Attributs mit fehlenden Werten gelöscht, ist *Missing* größer als 10% und kleiner oder gleich 70% werden die fehlenden Werte durch einfache Verfahren ersetzt und ist *Missing* größer als 70% wird das Attribut entfernt.

Vom Attribut MarkDown1 fehlen 64,3 % der Werte, weshalb die fehlenden Werte durch einfache Verfahren ersetzt werden. MarkDown2 weist einen Anteil von 73,6 % fehlender Werte auf, weshalb das Attribut im weiteren Verlauf aus dem Datensatz entfernt wird. Die Kennzahl *Missing* beträgt bei den Attributen MarkDown3 67,5%, MarkDown4 68 % und bei MarkDown5 64,1 %, womit die fehlenden Werte auch bei diesen Attributen durch einfache Verfahren ersetzt werden.

Aus der Beschreibung der Daten geht hervor, dass es sich bei den Attributen *MarkDown1-3* und *MarkDown5* um Sonderangebotsabschläge handelt und diese erst ab November 2011 erfasst wurden. Da es nicht möglich ist, diese Abschläge vor November 2011 zu schätzen und ein Ersetzen durch den Mittelwert das Ergebnis der Analyse verzerren könnte, wird die Annahme getroffen, dass es vor November 2011 keine Sonderangebotsabschläge gab. Die fehlenden Werte werden in diesem Zeitraum daher durch den Wert *0* ersetzt. Die Gründe für das Fehlen der Werte nach November 2011 werden nicht näher in der Beschreibung des Datensatzes erläutert und sind im Rahmen dieser Arbeit auch nicht feststellbar. Da es im Zeitraum ab November 2011 Sonderangebotsabschläge gab und diese Attribute wichtige Informationen enthalten könnten, wird hier angenommen, dass die Werte aufgrund technischer Fehler oder Defekte fehlen. Ein einfaches Ersetzen der Werte durch den Mittelwert könnte das Ergebnis jedoch verzerren. Deshalb werden die Werte ab November 2011 mithilfe einer linearen Interpolation ersetzt. In Unterkapitel 2.2.2 werden die Vorteile einer linearen Interpolation zum Ersetzen fehlender Werte näher erläutert.

Nach Prüfung der Datenqualität folgt der Klassifikator der Datenvorverarbeitung. Unterklassifikator ist dabei die Bearbeitung des Skalenniveaus. Hierbei gibt es die Möglichkeiten, alle Attribute in metrische Werte umzuwandeln, alle Attribute in kategorische Attribute umzuwandeln oder das Skalenniveau nicht zu ändern. Die nun vorliegenden Attribute weisen die in Tabelle 4-1 erkennbaren Eigenschaften auf, welche in Unterkapitel 2.2.4 genauer erläutert wurden.

Tabelle 4-1: Eigenschaften der Attribute im Walmart Datensatz nach der Datenvorverarbeitung

Attribut	Datentyp	Skalenniveau
Store	Ganzzahl	Metrisch
Dept	Ganzzahl	Metrisch
Date	Datum	Metrisch
Weekly_Sales	Gleitkommazahl	Metrisch
Temperature	Gleitkommazahl	Metrisch
Fuel_Price	Gleitkommazahl	Metrisch
MarkDown1	Gleitkommazahl	Metrisch
MarkDown3	Gleitkommazahl	Metrisch
MarkDown4	Gleitkommazahl	Metrisch
MarkDown5	Gleitkommazahl	Metrisch
CPI	Gleitkommazahl	Metrisch
Unemployment	Gleitkommazahl	Metrisch
Type	Buchstabe	Kategorisch
Size	Ganzzahl	Metrisch

Insgesamt liegen also 14 Attribute vor, von denen 13 ein metrisches Skalenniveau und nur eins ein kategorisches Skalenniveau aufweisen. Die Möglichkeit, die 13 metrischen Attribute mithilfe von Diskretisierungen in kategorische Attribute umzuwandeln besteht zwar, ist aber mit einem relativ hohen Aufwand und vor allem erheblichen Informationsverlust verbunden. Das Attribut *Weekly_Sales* könnte beispielsweise zur Diskretisierung in die Klassen *Niedrig*, *Mittel* und *Hoch*, oder mithilfe von Binning Klassen eingeteilt werden. In beiden Fällen werden bei der Klasseneinteilung Annahmen getroffen und dadurch, dass die Klassen ein bestimmtes Intervall abdecken, gehen zwangsläufig Informationen verloren. Die Gründe für den Informationsverlust wurden in Unterkapitel 2.2.4 näher erläutert. Werden die Attribute so belassen wie sie sind und das Skalenniveau nicht geändert, ist kein geeignetes Verfahren zur Anwendung auf den Datensatz vorhanden. Die sinnvollste Vorgehensweise mit dem vorliegenden Datensatz weiter zu verfahren, ist demnach alle Attribute in metrische Attribute umzuwandeln.

Mit dem Attribut *Type* liegt das einzige kategorische Attribut vor. Das Attribut hat die Ausprägungen *A*, *B* und *C*. In Unterkapitel 2.2.4 wurden die Möglichkeiten zur Transformation von kategorischen zu metrischen Werten erläutert. Da zwischen den Ausprägungen des nominalen Attributs keine Rangordnung besteht, darf bei der Umwandlung auch keine entstehen. Da das Attribut nur drei Ausprägungen hat, ist eine Dummy-Kodierung zielführend. Das nominale Attribut wird also mithilfe einer Dummy-Kodierung in drei metrische Attribute mit jeweils binärer Ausprägung umgewandelt. So liegen nun insgesamt 16 metrische Attribute vor. In einem nächsten Schritt kann, wie in Abbildung 4-5 zu erkennen ist, eine PCA oder eine Normalisierung auf die Werte angewendet werden. In Unterkapitel 3.3.3.2 wurde genauer auf die Transformationen eingegangen. Der vorliegende Datensatz wird nicht transformiert, da dies zwar positive Auswirkungen auf die Performance der Verfahren hat, die Ergebnisse dadurch unter Umständen jedoch schwieriger interpretiert werden können. Der Fokus dieser Arbeit liegt nicht

auf der Performance der Verfahren, sondern der Anwendbarkeit der Klassifikatoren und der Systematisierung und der Verfahren.

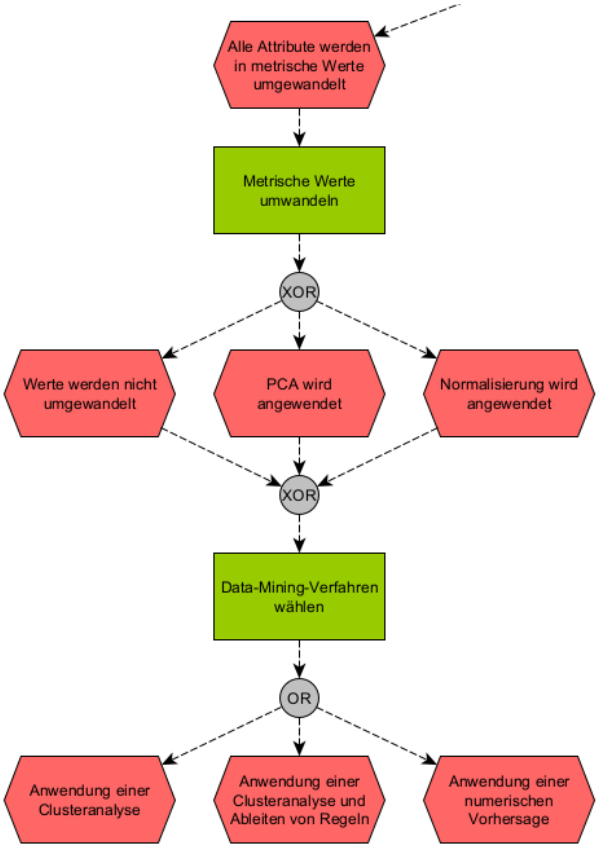


Abbildung 4-5: Handhabung metrischer Werte und anwendbare Verfahren auf den Walmart Datensatz

Nachdem die vorverarbeitenden Schritte angewendet wurden, kann ein Data-Mining-Verfahren auf den Datensatz angewendet werden. Analog dem Vorgehen der EPK können nun, wie in Abbildung 4-5 unten zu erkennen, die folgenden Verfahren auf den Zieldatensatz angewendet werden: Anwendung einer Clusteranalyse, Anwendung einer Clusteranalyse und Ableiten von Regeln, Anwendung einer numerischen Vorhersage.

Als erstes Verfahren wird eine Clusteranalyse auf den Zieldatensatz angewendet. In Unterkapitel 2.3.1 wurden das Ziel und die Funktionsweise solcher Analyseverfahren genauer erläutert. Im Vordergrund einer Clusteranalyse steht eine Beschreibung des Datensatzes. Da vorab keine Zielsetzung definiert wurde, eignet sich das Verfahren gut, um erste Zusammenhänge innerhalb des Datensatzes zu erkennen und zu beschreiben. Im weiteren Verlauf der Analyse können die so entdeckten Zusammenhänge hilfreich sein.

Da das Verfahren ohne weitere Schritte auf den Zieldatensatz angewendet werden soll, eignet sich der X-Means-Algorithmus hier für eine Clusteranalyse. Dieser hat gegenüber dem k-Means-Algorithmus den Vorteil, dass die Anzahl der Cluster vom Verfahren selbst optimiert wird und der Anwender diese nicht wie beim k-Means Verfahren vorher definieren muss. Die beiden Verfahren weisen noch weitere Unterschiede auf, die jedoch nicht im Fokus dieser Arbeit liegen. Bei Anwendung auf diesen Datensatz werden vom Verfahren zwei Cluster als optimal angesehen.

Das erste Cluster, in der Abbildung blau dargestellt, enthält 209.119 und das zweite Cluster, in der Abbildung rot dargestellt, 212.451 Datensätze. Die beiden Cluster sind in Abbildung 4-6 zu erkennen.

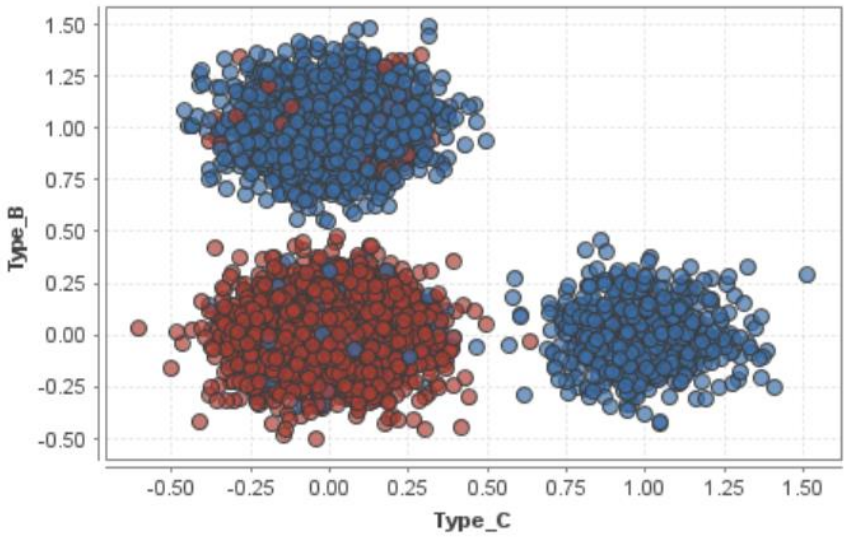


Abbildung 4-6: Visualisierung der im Walmart Datensatz identifizierten Cluster

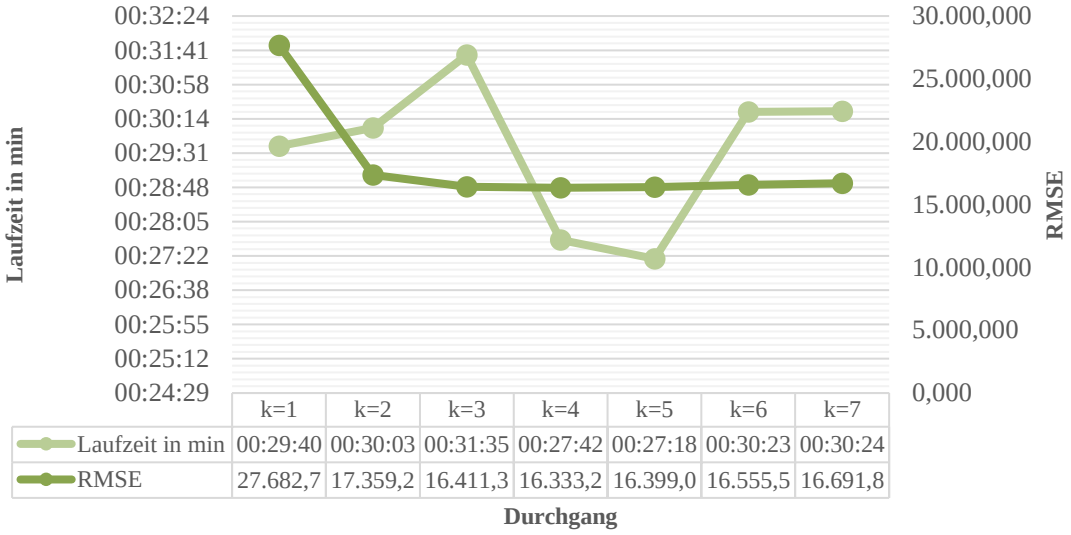
Wie auch in der Abbildung erkennbar ist, werden die Cluster über die Attribute *Type*_A, Type_*B* und *Type_C* definiert. Im weiteren Verlauf einer Analyse könnte an dieser Stelle angesetzt und geprüft werden, inwieweit die Attribute, über die die Cluster definiert werden, zur weiteren Analyse genutzt werden können. Auf diese Weise könnten Zusammenhänge erkannt und Attribute mit hohem Informationsgehalt identifiziert werden. Da dies jedoch nicht der Zweck dieser Arbeit ist, wird der Zusammenhand an dieser Stelle nicht genauer betrachtet.

Festzuhalten ist, dass die grundsätzliche Anwendung einer Clusteranalyse mithilfe der Klassifikatoren erfolgreich durchgeführt werden kann.

Als nächstes Verfahren wird eine Clusteranalyse auf den Datensatz angewendet und anschließend Regeln aus dieser Analyse abgeleitet, um eine Vorhersage zu realisieren. Das Ziel und die Funktionsweise dieser Analysemethode wurde in Unterkapitel 2.3.1 näher erläutert. Im Vordergrund steht nun nicht mehr die reine Beschreibung des Datensatzes, sondern eine Vorhersage.

Als Verfahren, welches eine Clusteranalyse auf den Datensatz anwendet und anschließend Regeln aus dieser Analyse ableitet, eignet sich ein K-Nearest-Neighbour-Algorithmus. Mithilfe des Algorithmus soll das Attribut *Weekly_Sales* prognostiziert werden, da dies in der Praxis von Interesse für Walmart sein dürfte. Ein Einzelhandelsunternehmen könnte diese Daten nutzen, um beispielsweise den zu erwartenden Kundenandrang abzuschätzen oder das nötige Personal planen zu können. Da an dieser Stelle durch die Klassifikatoren keine weitere Auswahl an Attributen vorgesehen ist, wird das Verfahren mit allen Attributen des Zieldatensatzes angewendet. Um ein zielführendes K zu finden, wird damit begonnen, das Verfahren mit K=1 zu beginnen und danach bei jedem Durchgang das K um 1 zu erhöhen, bis keine Verbesserung der Ergebnisse mehr feststellbar ist. Auf die Herausforderungen bei der Evaluation der Ergebnisse wird in Unterkapitel 2.4.3 genauer eingegangen. Um ein quantitatives Maß für die Bewertung der Performance des

Verfahrens heranzuziehen, wird für die Evaluation des KNN-Algorithmus die Wurzel des mittleren quadratischen Fehlers (RMSE) genutzt. Die Ergebnisse der Durchgänge sind Abbildung 4-7 zu entnehmen.



	k=1	k=2	k=3	k=4	k=5	k=6	k=7
Laufzeit in min	00:29:40	00:30:03	00:31:35	00:27:42	00:27:18	00:30:23	00:30:24
RMSE	27.682,7	17.359,2	16.411,3	16.333,2	16.399,0	16.555,5	16.691,8

Abbildung 4-7: Performance des KNN-Algorithmus auf dem Walmart Datensatz

Bei Betrachtung von Abbildung 4-7 und der Ergebnisse der Durchgänge ist zu erkennen, dass ab k = 4 eine leichte Verbesserung der Laufzeit mit einer leichten Verschlechterung der Schätzgenauigkeit einhergeht. Der kleinstmögliche RMSE bei Anwendung des KNN-Algorithmus auf den Zieldatensatz liegt mit 16.399,01 bei k = 5. Bei einer Prognose des Attributs *Weekly_Sales* mithilfe dieses Verfahrens und dem vorliegenden Zieldatensatz wäre im Durchschnitt eine Abweichung der Schätzergebnisse von 16.399,01 zu erwarten. Das Attribut *Weekly_Sales* hat einen Mittelwert von 15.981,258. Bei einem Mittelwert von 15.981,258 des Attributs *Weekly_Sales* ist diese Prognose so ungenau, dass sie in der Praxis nur sehr bedingt hilfreich für das Einzelhandelsunternehmen wäre.

Insgesamt kann die grundsätzliche Durchführung einer Clusteranalyse und das anschließende Ableiten von Regeln zur Prognose eines Attributs mithilfe der Klassifikatoren erfolgreich durchgeführt werden. Die Ergebnisse der Prognose sind jedoch nur sehr bedingt nutzbar.

Als letztes Verfahren wird eine numerische Vorhersage auf den Zieldatensatz angewendet. Die grundsätzliche Funktionsweise dieser Verfahren wurde in Unterkapitel 2.3.1 näher erläutert. Der Fokus eines solchen Verfahrens liegt ausschließlich auf einer Vorhersage und nicht auf einer Beschreibung der vorliegenden Daten.

Für die numerische Vorhersage wird eine lineare Regression verwendet. Wie bei Anwendung der Clusteranalyse und dem Ableiten der Regeln zur Vorhersage, wird auch mithilfe der linearen Regression das Attribut *Weekly_Sales* vorhergesagt. Auch hierbei wird der RMSE als quantitatives Maß für die Bewertung der Performance des Verfahrens herangezogen.

In RapidMiner gibt es die Möglichkeit, eine automatische Auswahl von Attributen, welche sich am besten für die Vorhersage mit einer linearen Regression nutzen lassen, vornehmen zu lassen. Auf die Möglichkeit einer automatischen Vorauswahl durch Data-Mining-Verfahren wird

in Unterkapitel 2.2.3 kurz eingegangen. Um eine möglichst genaue Vorhersage zu erzielen, wird die lineare Regression einmal mit jeder automatischen Vorauswahl ausgeführt und anschließend die Performance des Modells anhand des RMSE verglichen. Die Ergebnisse der Durchgänge sind in Abbildung 4-8 zu erkennen.

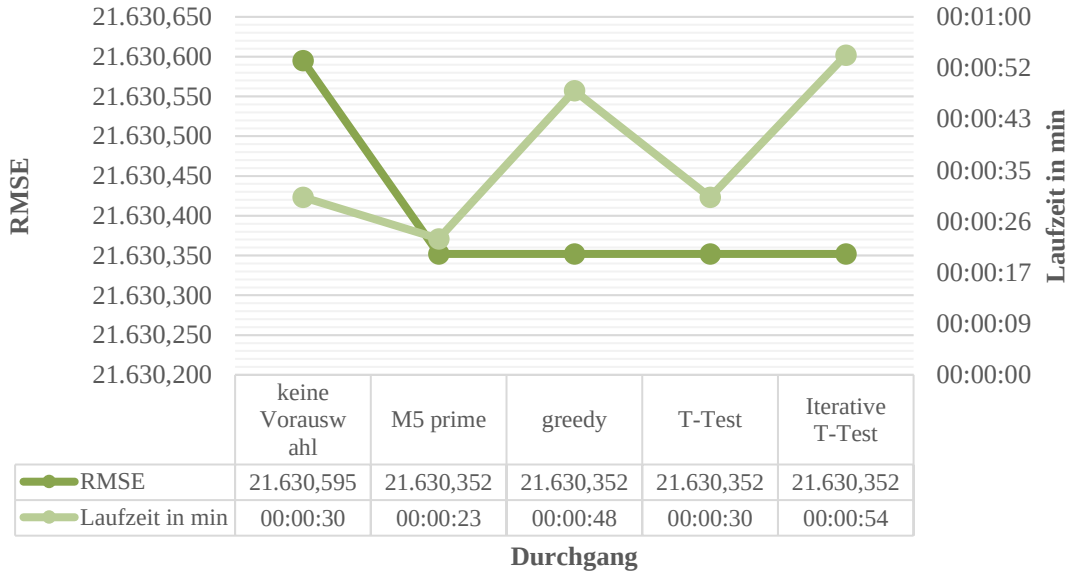


Abbildung 4-8: Performance der linearen Regression auf dem Walmart Datensatz

Bei Betrachtung der Abbildung 4-8 und der Ergebnisse fällt auf, dass der RMSE bei den vier Vorauswahlverfahren genau gleich groß mit 21.630,595 ausfällt. Wird keine automatische Vorauswahl durchgeführt, ist der RMSE nur minimal größer. Bei der Dauer der Laufzeit sind Unterschiede erkennbar. Die kürzeste Laufzeit hat die lineare Regression bei einer Vorauswahl mithilfe des M5 prime Algorithmus. Da dieser auch den geringeren RMSE aufweist, handelt es sich hierbei im Vergleich zu den anderen vier Durchläufen um die zielführendste lineare Regression. Das Attribut *Weekly_Sales* hat einen Mittelwert von 15.981,258. Für die Prognose mit Hilfe der linearen Regression gilt das gleiche wie für den KNN-Algorithmus. Aufgrund des im Vergleich zum Mittelwert sehr hohen RMSE ist die Aussagekraft stark eingeschränkt.

Mithilfe der Klassifikatoren kann jedoch grundsätzlich auch eine numerische Vorhersage in Form einer linearen Regression auf den Zieldatensatz angewendet werden. Auch die Ergebnisse dieser Prognose sind jedoch nur sehr bedingt nutzbar.

4.1.2 Exemplarische Anwendung auf den Rossmann Datensatz

Der Rossmann Datensatz wird in Unterkapitel 3.1.2 genau erläutert. Insgesamt liegen im kombinierten Datensatz 18 Attribute vor. Von diesen 18 Attributen sind acht kategorisch und zehn numerisch.

Die Analyse beginnt, wie in Abbildung 4-9 zu erkennen, mit dem Klassifikator Ziel/Weck der Analyse. Bei der Erläuterung des Datensatzes wird erwähnt, dass dieser im Rahmen eines Wettbewerbs veröffentlich wurde. Ziel war es damals, die Umsätze der Filialen vorherzusagen. Dieses Ziel wird auch in der folgenden Analyse verfolgt. Das übergeordnete Ziel ist also bekannt und kann als Prognose gewählt werden. Als nächstes folgt der Klassifikator Skalenniveau. Hier liegen sowohl metrische als auch kategorische Attribute und somit ein gemischtes Skalenniveau

vor. Nachdem das Skalenniveau bestimmt wurde, folgt der Klassifikator Datenvorverarbeitung. Hier ist wieder erkennbar, dass lediglich ein Robust Learner auf den Datensatz angewendet werden kann, wenn die Datenvorverarbeitung nicht durchgeführt wird. Wird keine Datenvorverarbeitung durchgeführt, liegen Rohdaten vor, die möglicherweise stark verrauscht sind. Auch wenn andere Verfahren theoretisch auf die Datengrundlage angewendet werden könnten, ist dabei nicht mit aussagekräftigen Ergebnissen zu rechnen. Die Gründe dafür werden in Unterkapitel 2.2 ausführlich erläutert. Da gezielt ein Verfahren auf den Datensatz angewendet werden soll, wird eine Datenvorverarbeitung durchgeführt.

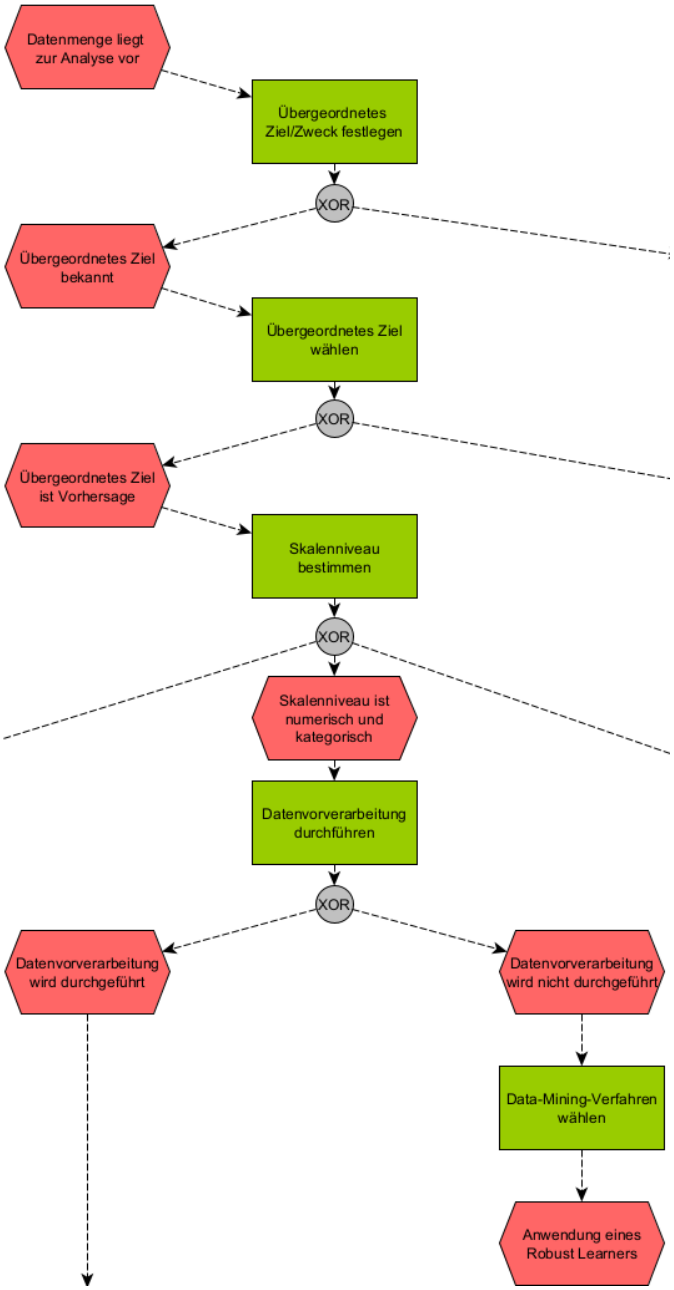


Abbildung 4-9: Anwendung der Klassifikatoren Ziel/Zweck, Skalenniveau und Datenvorverarbeitung auf den Rossmann Datensatz

An dieser Stelle folgt die Datenvorverarbeitung mithilfe der in Unterkapitel 3.3.3.2 integrierten Unterklassifikatoren. Dazu wird Qualität der Daten bestimmt, indem die fehlenden Werte genauer

untersucht werden und anschließend mithilfe der entwickelten Kennzahlen *Missing* und *Stability* über den Umgang mit Attributen, die identische oder fehlende Werte enthalten, entschieden wird. Die Unterklassifikatoren werden in Unterkapitel 3.3.3.2 genau erläutert und können Abbildung 3-10 entnommen werden.

Als erster Unterklassifikator kommt die Bestimmung der Datenqualität. In RapidMiner wird dieser Schritt mithilfe des Operators Quality Measures realisiert. Der Operator bestimmt die Kennzahlen *Stability* und *Missing*. In Abbildung 4-10 sind diese als Balkendiagramm dargestellt.

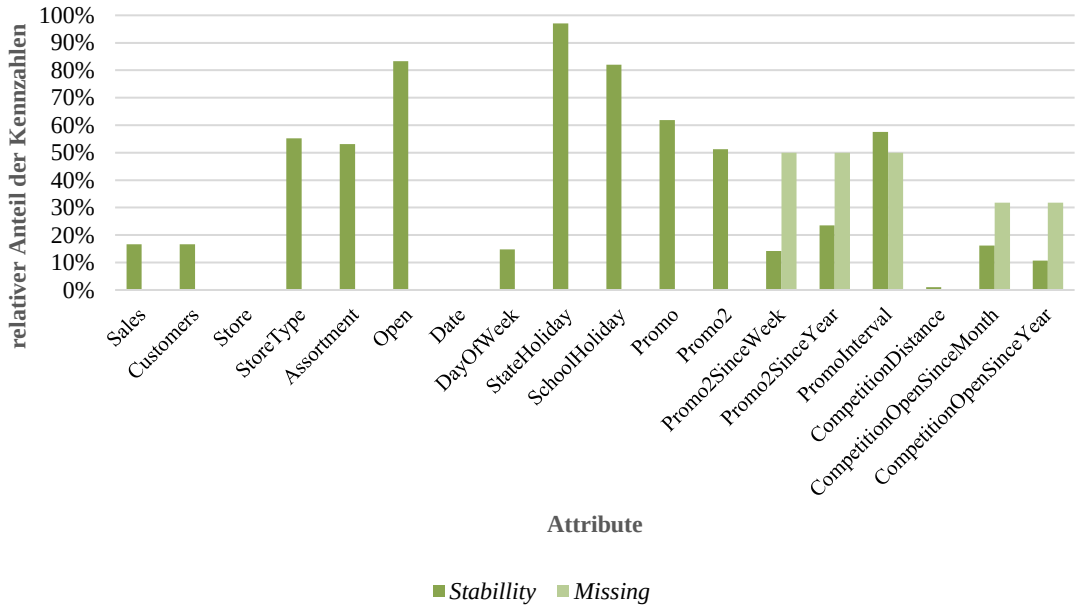


Abbildung 4-10: Relevante Qualitätskennzahlen Rossmann Datensatz

Als erstes wird die Kennzahl *Stability* betrachtet. Das Attribut *StateHoliday* weist mit 97,1% den höchsten Wert dieser Kennzahl auf. Damit ist das Attribut über dem in den Unterklassifikatoren definierten Schwellenwert von 90% und wird im Folgenden aus dem Datensatz entfernt. Die Attribute *Open* und *SchoolHoliday* haben mit 83% und 82% ebenfalls hohe *Stability* Werte, bleiben jedoch unter dem definierten Schwellenwert und werden somit nicht entfernt.

Bevor die Kennzahl *Missing* betrachtet wird, ist zu prüfen, ob es sich bei den fehlenden Werten um den Typ NMAR, MAR oder MCAR handelt.

Bei einem Blick auf Abbildung 4-10 fällt auf, dass bei den Attributen *Promo2SinceWeek*, *Promo2SinceYear*, *PromoInterval*, *CompetitionDistance*, *CompetitionOpenSinceMonth* und *CompetitionOpenSinceYear* Werte fehlen. Alle diese Attribute stammen ursprünglich aus dem Datensatz *Store*. Aus diesem Grund wird der Datensatz *Store* separat betrachtet. Eine mithilfe des Operators Sample zufällig gezogene Zufällige Stichprobe aus dem Datensatz ist in Tabelle 4-2 zu sehen. Bei den mit einem *?* gekennzeichneten Ausprägungen handelt es sich um fehlende Werte.

Bei einer genaueren Analyse der Tabelle fällt auf, dass die Werte von *Promo2SinceWeek*, *Promo2SinceYear* und *PromoInterval* immer dann fehlen, wenn die Ausprägung von *Promo2 0* ist. Das ist ein klares Muster innerhalb der Daten. Damit handelt es sich bei den fehlenden Werten um die Kategorie MAR. Die Fehlerkategorien werden in Unterkapitel 2.2.2 genauer erläutert.

Tabelle 4-2: Zufällig gezogene Stichprobe aus dem Store Datensatz

Store	StoreType	Assortment	Competition Distance	Competition OpenSince Month	Competition OpenSince Year	Promo2	Promo2 SinceWeek	Promo2 SinceYear	Promo Interval
16	a	c	3270	?	?	0	?	?	?
72	a	a	2200	12	2009	1	13	2010	Jan,Apr,Jul,Oct
103	d	c	5210	5	2015	0	?	?	?
128	d	c	2000	?	?	1	1	2013	Jan,Apr,Jul,Oct
193	a	a	520	?	?	0	?	?	?
206	a	c	380	?	?	1	14	2012	Jan,Apr,Jul,Oct
233	a	a	1890	?	?	0	?	?	?
253	a	c	250	?	?	1	5	2013	Feb,May,Aug,Nov
337	d	c	10600	7	2005	1	45	2014	Feb,May,Aug,Nov
395	a	a	3620	2	2013	0	?	?	?
430	d	c	12870	10	2008	0	?	?	?
527	d	c	5830	4	2008	0	?	?	?
647	a	c	7420	4	2013	0	?	?	?
679	a	a	4140	9	2012	0	?	?	?
871	d	c	10620	?	?	0	?	?	?
872	a	c	3860	9	2014	1	23	2015	Mar,Jun,Sept,Dec
958	a	a	440	11	2013	0	?	?	?
1076	a	c	90	?	?	1	1	2013	Jan,Apr,Jul,Oct
1091	a	c	9990	?	?	0	?	?	?
1097	b	b	720	3	2002	0	?	?	?

Bei den Attributen *CompetitionOpenSinceMonth* und *CompetitionOpenSinceYear* fällt auf, dass die Werte jeweils zusammen fehlen. Vom Attribut *CompetitionDistance* fehlen insgesamt drei Werte. Bei der Beschreibung des Datensatzes wird erwähnt, dass einige Filialen während der Datenerfassung wegen Renovierungsarbeiten vorrübergehend geschlossen waren. Eventuell könnte das eine Erklärung für das Fehlen der Werte sein. An dieser Stelle kann jedoch nur gemutmaßt und kein genauer Grund für das Fehlen ermittelt werden. Innerhalb der Daten ist im Rahmen dieser Arbeit kein Muster erkennbar, welches erklärt warum die Werte fehlen. Es wird daher angenommen, dass es sich bei den fehlenden Werten um die Kategorie MAR oder sogar MCAR handelt.

Nach Ermittlung der Fehlerkategorie als MAR oder MCAR kann die Kennzahl *Missing* genau betrachtet werden. Diese ist Abbildung 4-10 zu entnehmen. Bei den Attributen *Promo2SinceWeek*, *Promo2SinceYear* und *PromoInterval* fehlen jeweils 50% der Werte. Mithilfe des Unterklassifikators wird deshalb entschieden, dass die fehlenden Werte durch einfache Verfahren ersetzt werden können. Da die Werte immer dann fehlen, wenn ein Geschäft nicht an einer Promotion-Aktion teilnimmt, können diese fehlenden numerischen Werte von *Promo2SinceWeek* und *Promo2SinceYear* durch *0* ersetzt werden und die kategorischen von *PromoInterval* ebenfalls durch *0*. Vom Attribut CompetitionDistance fehlen lediglich 0,3% der Werte. Aus diesem Grund werden die Zeilen mit fehlenden Werten aus dem Datensatz entfernt.

Von den Attributen *CompetitionOpenSinceMonth* und *CompetitionOpenSinceYear* fehlen jeweils 31,8%. Mithilfe des Unterklassifikators wird deshalb entschieden, die Fehlenden Werte durch einfache Verfahren zu ersetzen. Einige häufig genutzte einfache Verfahren sowie deren Vor- und Nachteile zum Ersetzen fehlender Werte wurden in Unterkapitel 2.2.2 genauer erläutert. Da die beiden Attribute jedoch möglicherweise wichtige Informationen enthalten können und ein Ersetzen durch den Mittelwert oder häufigsten Wert nicht sinnvoll wäre, werden die fehlenden Werte mithilfe eines KNN-Algorithmus geschätzt. Dabei wird in Kauf genommen, dass die geschätzten Werte Auswirkungen auf das Ergebnis haben können.

Nachdem die Datenqualität geprüft wurde, folgt in der Datenvorverarbeitung der Unterklassifikator der Bearbeitung des Skalenniveaus. Hier besteht die Möglichkeit, alle Attribute in metrische oder alle in kategorische Werte umzuwandeln oder die Attribute in ihrem Ursprung zu belassen. Um an dieser Stelle eine fundierte Entscheidung zu treffen, werden die Eigenschaften der Attribute betrachtet. Diese sind in Tabelle 4-3 zu erkennen.

Tabelle 4-3: Eigenschaften der Attribute im Rossmann Datensatz nach der Datenvorverarbeitung

Attribut	Datentyp	Skalenniveau
Sales	Ganzzahl	Metrisch
Customers	Ganzzahl	Metrisch
Store	Ganzzahl	Metrisch
StoreType	Buchstabe	Kategorisch
Assortment	Buchstabe	Kategorisch
Open	Binominal	Metrisch
Date	Datum	Metrisch
DayOfWeek	Ganzzahl	Metrisch
SchoolHoliday	Buchstabe	Kategorisch
Promo	Binominal	Metrisch
Promo2	Binominal	Metrisch
Promo2SinceWeek	Ganzzahl	Metrisch
Promo2SinceYear	Ganzzahl	Metrisch
PromoInterval	Nominal	Kategorisch
CompetitionDistance	Ganzzahl	Metrisch
CompetitionOpenSinceMonth	Ganzzahl	Metrisch
CompetitionOpenSinceYear	Ganzzahl	Metrisch

Insgesamt liegen somit sieben kategorische und 10 numerische Attribute vor. Alle Attribute in metrische oder alle in kategorische Attribute umzuwandeln ist nicht zielführend, da dabei wichtige Informationen verloren gehen oder verzerrt werden könnten. Aus diesem Grund wird das Skalenniveau nur bei einigen Attributen geändert.

Das Attribut *Date* umfasst insgesamt einen Zeitraum von 940 Tagen. Das genaue Datum ist für eine Vorhersage jedoch nicht entscheidend. Wichtiger ist der Tag der Woche, welcher im Attribut *DayOfWeek* bereits enthalten ist, und der Monat. Aus diesem Grund wird aus dem Attribut Date der Monat extrahiert, und das Attribut so in ein kategorisches Attribut diskretisiert.

Die Attribute *Store* und *DayOfWeek* werden in kategorische Werte umgewandelt, damit ein Algorithmus keine Rangfolge innerhalb der Werte erkennen kann. Die Thematik der Transformationen und Behandlung von Skalenniveaus wird in Unterkapitel 2.2.4 genauer behandelt und erläutert. So liegen nun insgesamt 10 kategorische und sieben metrische Attribute vor.

Nachdem das Skalenniveau bearbeitet wurde und der Zieldatensatz vorliegt, steht fest, dass das Skalenniveau nach wie vor gemischt ist. Welche Verfahren nun laut Klassifikatoren auf den Zieldatensatz angewendet werden können, ist der Abbildung 4-11 zu entnehmen.

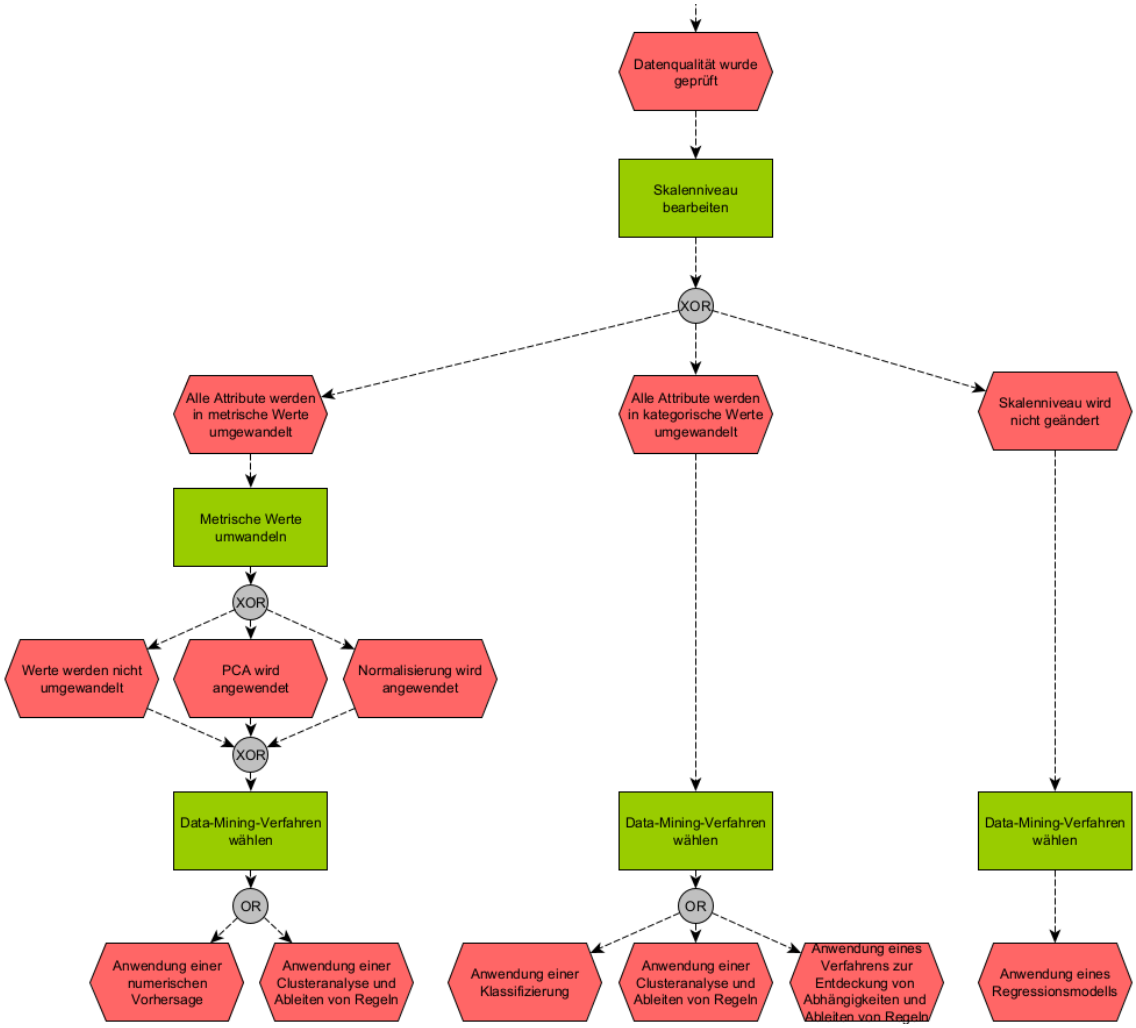


Abbildung 4-11: Bearbeitung des Skalenniveaus und folgende anwendbare Verfahren auf den Walmart Datensatz

Wie in Abbildung 4-11 zu erkennen ist, wird im Folgenden analog dem Vorgehen in der EPK ein Regressionsmodell auf den Zieldatensatz zur Prognose des Attributs *Sales* angewendet. Hierzu wird ein Random Forest verwendet.

In Unterkapitel 2.3.1 wurde das Ziel und die Funktionsweise von Vorhersagemodellen genauer erläutert. Da die Zielsetzung bei dieser Analyse klar mit der Prognose des Attributs *Sales* definiert wurde, steht diese auch im Fokus. Der Random Forest wird mit einer Anzahl von 100 Bäumen und einer steigenden Blattgröße in mehreren Durchgängen angewendet. Die Ergebnisse der jeweiligen Durchgänge können Tabelle 4-4 entnommen werden

Tabelle 4-4: Performance des Random Forest auf dem Rossmann Datensatz

Durchgang	Blattgröße	Anzahl Bäume	RMSE	Laufzeit in min
1	10	100	1.011,172	00:03:14
2	11	100	932,291	00:03:36
3	12	100	872,277	00:04:33
4	13	100	826,217	00:05:05
5	14	100	792,254	00:05:52
6	15	50	769,863	00:04:25
7	16	30	766,435	00:03:03

Bei einem Blick in Tabelle 4-4 fällt auf, dass die Anzahl der Bäume des Random Forest in Durchgang 6 und 7 deutlich niedriger sind als in den Durchgängen 1-5. Diese Reduzierung musste vorgenommen werden, da der für diese Arbeit verwendete Computer nicht in der Lage ist, einen Durchgang mit 100 Bäumen und einer Blattgröße von mehr als 14 zu berechnen. Da der Fokus dieser Arbeit nicht auf einer optimalen Performance der Verfahren, sondern auf der Anwendung der Klassifikatoren liegt, wurde nach Durchgang 7 abgebrochen.

Zur besseren Ansicht sind die beiden bereits zuvor verwendeten Kennzahlen RMSE und die Laufzeit in den jeweiligen Durchgängen als Grafik in Abbildung 4-12 dargestellt.

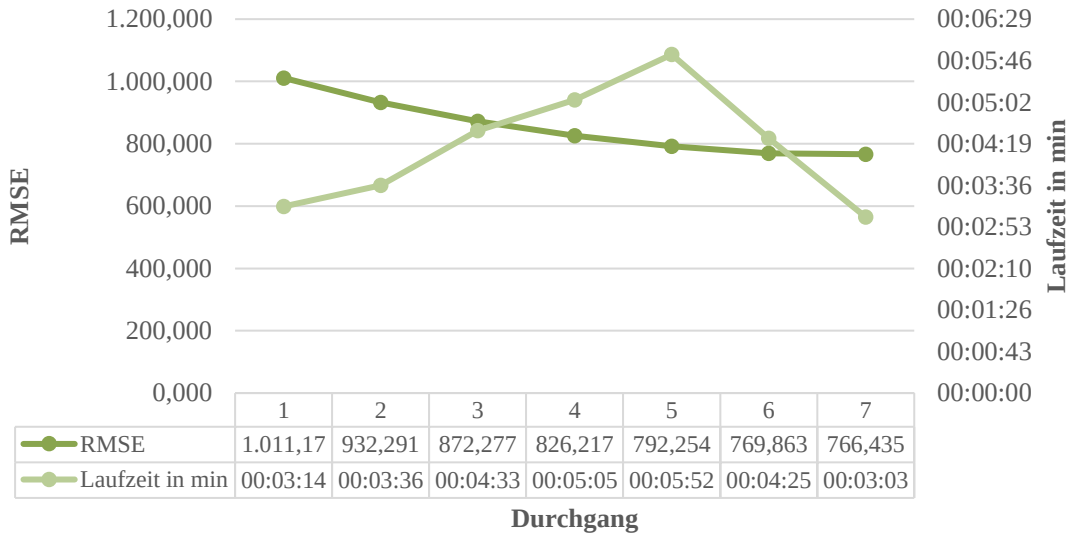


Abbildung 4-12: Performance des Random Forest auf dem Rossmann Datensatz

Das beste Ergebnis wurde in Durchgang 7 mit einer Baumanzahl von 30 Bäumen und einer Blattgröße von 16 erzielt. Mit einem RMSE von 766,435 und handelt es sich bei einem Mittelwert des Zielattributs von 5777,043 um eine relativ verlässliche Vorhersage. Die vom Verfahren geschätzten Werte werden also nur leicht von den wahren Werten abweichen. Ein Einzelhandelsunternehmen könnte aus einer solchen Vorhersage hilfreiche Informationen ableiten.

Insgesamt kann mithilfe der Klassifikatoren erfolgreich ein aussagekräftiges Prognosemodell auf den Zieldatensatz angewendet werden. Mithilfe der Klassifikatoren wird zu Beginn unmittelbar erkannt, dass ohne eine Datenvorverarbeitung lediglich ein Robust Learner auf den Datensatz angewendet werden kann. Aus diesem Grund wird eine Datenvorverarbeitung

durchgeführt. Die Unterklassifikatoren unterstützen mit den Kennzahlen *Stability* und *Missing* dabei, die Datenqualität zu bestimmen und über das Entfernen von Attributen und den Umgang mit fehlenden Werten zu entscheiden. Nach der Bearbeitung des Skalenniveaus einiger Attribute liegt ein gemischtes Skalenniveau vor, auf das ein Regressionsmodell angewendet werden kann. Ein Random Forest liefert aussagekräftige Ergebnisse.

4.2 Validierung der Klassifikatoren

Im Folgenden werden die in Unterkapitel 4.1 angewendeten Klassifikatoren validiert und im Anschluss wird die Anforderungserfüllung überprüft. Dazu wird zuerst die angewendete Validierungsmethodik vorgestellt und anschließend auf die Klassifikatoren angewendet.

4.2.1 Vorstellung der Validierungsmethodik

Um eine objektive Validierung der Klassifikatoren zu ermöglichen, wird zuerst eine explorative Validierungsmethode entwickelt und anschließend eine kennzahlenbasierte Validierungsmethode erarbeitet.

Entwicklung einer explorativen Validierungsmethode

Bei der explorativen Validierung werden die in Kapitel 2 erarbeiteten Grundlagen und Herausforderungen aus der Literatur und aktueller Forschung mit den in Unterkapitel 4.1 angewendeten Klassifikatoren kritisch auf ihre praktische Anwendbarkeit untersucht und bewertet. Welche praktischen Anwendungsfelder dabei zu untersuchen sind, ist in Abbildung 4-13 dargestellt. Die explorative Validierung wird in Form eines Kreises mit Pfeilen in der Mitte dargestellt, um die Rekursivität der Validierung und unscharfe Trennung der praktischen Anwendungsfelder zu verdeutlichen.

Abbildung 4-13: Explorative Validierung der Klassifikatoren

Es ist zu bewerten, ob das Hauptziel dieser Arbeit, die Systematisierung von Data-Mining-Verfahren in Klassifikatoren, erfolgreich umgesetzt werden konnte. Dabei ist zu prüfen, ob die

systematisierten Verfahren auch erfolgreich auf die Beispieldatensätze angewendet werden können.

Neben der reinen Systematisierung und Anwendbarkeit der Verfahren wird das Systematisierungskonzept als EPK dargestellt. Hier ist zu prüfen, wie erfolgreich der Prozess anhand der Beispieldatensätze durchlaufen und das Konzept schrittweise angewendet werden kann.

Des Weiteren sollen die entwickelten Klassifikatoren Anwender bei der Datenvorverarbeitung unterstützen. Mithilfe der Klassifikatoren sollen auch Anwender, die, wie in Unterkapitel 2.4.2 näher erläutert, über eine geringere Datenkompetenz bei der Entscheidungsfindung und Datenvorverarbeitung verfügen, unterstützt werden.

Insgesamt handelt es sich bei Data-Mining-Prozessen, wie in Unterkapitel 2.1 anhand der gängigen Vorgehensmodelle erläutert, um einen iterativen Prozess. Das als EPK dargestellte Konzept ist nur ein Teilschritt des gesamten Data-Mining-Prozesses. Wie bereits zu Beginn von Kapitel 3 erläutert, handelt es sich beim in Unterkapitel 2.1.2 vorgestellten CRISP-DM-Vorgehensmodell um das dieser Arbeit zugrundeliegende Modell. Aus diesem Grund ist schlussendlich zu prüfen, inwieweit das in dieser Arbeit entwickelte Konzept in das CRISP-DM Vorgehensmodell integriert werden kann.

Entwicklung einer kennzahlenbasierten Validierungsmethode

Mithilfe der Klassifikatoren sollen neben der Systematisierung der Data-Mining-Verfahren auch Anwender bei der Analyse unterstützt werden. Um zu validieren, wie gut Anwender unterstützt werden, ist zu überprüfen, ob das Konzept schrittweise einwandfrei auf die Beispieldatensätze angewendet werden konnte, oder weitere Klassifikatoren sinnvoll gewesen wären.

Um die Unterstützung der Anwender mithilfe der Klassifikatoren und Unterklassifikatoren objektiv zu messen, wird der Klassifikationsindex eingeführt. Dazu wird die Anzahl der vom Konzept gegebenen Klassifikatoren durch die Anzahl der Klassifikatoren geteilt, die bei der Anwendung der EPK auf die Beispieldatensätze sinnvoll gewesen wären.

$$Klassifikationsindex = \frac{Anzahl\ gegebener\ Klassifkatoren}{Anzahl\ notwendiger\ Klassifikatoren}$$

Ein Klassifikationsindex von 1 bedeutet, dass keine weiteren Klassifikatoren für eine zielführende Unterstützung bei der Analyse des Beispieldatensatzes nötig sind. Liegt der Index unter 1, bedeutet dies, dass weitere Klassifikatoren für eine zielführende Unterstützung bei der Analyse sinnvoll wären. Je kleiner der Klassifikationsindex, desto weniger konnten die Klassifikatoren bei der Analyse unterstützen.

4.2.2 Anwendung der Validierungsmethodik

Im Folgenden wird die in Unterkapitel 4.2.1 entwickelte Validierungsmethodik angewendet. Dabei wird zuerst die explorative und anschließend die kennzahlenbasierte Validierung angewendet.

Anwendung der explorativen Validierung

Zuerst wird bewertet, ob das Hauptziel dieser Arbeit, Data-Mining-Verfahren in Klassifikatoren zu systematisieren, erfolgreich umgesetzt werden konnte. Das zu diesem Zweck entwickelte Konzept wird, um den Prozesscharakter eines Data-Mining-Projektes zu verdeutlichen, als eine EPK dargestellt. In der Fachliteratur werden Data-Mining-Verfahren, wie in Unterkapitel 2.3.1 genauer erläutert wird, üblicherweise nach ihrer Aufgabenstellung klassifiziert. Die Klassifizierung von Data-Mining-Verfahren in der Software RapidMiner, welche ansatzweise nach den übergeordneten Zielen Vorhersage und Beschreibung umgesetzt wird, wird in Unterkapitel 2.3.2 vorgestellt. Sowohl die Strukturierung in der Fachliteratur als auch in RapidMiner setzen voraus, dass schon vor der Analyse der Daten mithilfe eines Data-Mining-Verfahren festgelegt sein muss, welches Analyseverfahren durchgeführt werden soll. Dabei wird die vorliegende Datengrundlage nicht berücksichtigt. Mithilfe des neu entwickelten Konzepts wird sowohl die Zielsetzung als auch die Datengrundlage zur Systematisierung der Verfahren genutzt. Anwender haben mithilfe des Konzepts nur auf Grundlage des vorliegenden Datenbestands und der wählbaren Datenvorverarbeitungsschritte eine Übersicht der anwendbaren Verfahren ohne eine vorher festgelegte Zielsetzung.

Eine Anwendung der Systematisierung nach Klassifikatoren kann erfolgreich auf die Beispieldatensätze angewendet werden. Die Wahl der Verfahren geschieht am Beispiel des Walmart Datensatzes lediglich aufgrund der vorliegenden Datengrundlage. Dabei werden eine Clusteranalyse mit dem Ziel der Beschreibung des Datensatzes, sowie eine Clusteranalyse und das anschließende Ableiten von Regeln und eine lineare Regression zur Prognose erfolgreich angewendet. Das Hauptziel dieser Arbeit wird also erreicht.

Ebenfalls ist neben der reinen Systematisierung auch die Anwendbarkeit der einzelnen in der EPK dargestellten Prozessschritte zu prüfen. Bei beiden Beispieldatensätzen wird die EPK von Beginn an durchlaufen.

Beim ersten Beispieldatensatz wird keine Zielsetzung vorausgesetzt und alle Schritte können durchlaufen werden. Auch die Unterklassifikatoren der Datenvorverarbeitenden Schritte werden erfolgreich auf den Datensatz angewendet. So können mithilfe der als Prozessschritte dargestellten Unterklassifikatoren Attribute von Beginn an für die Analyse ausgeschlossen werden. Dabei stellt sich jedoch heraus, dass noch weitere Klassifikatoren, besonders in der Datenvorverarbeitung sinnvoll wären. Darauf wird im folgenden Absatz, der sich mit der Validierung der Unterstützung von Anwendern bei der Datenvorverarbeitung beschäftigt, genauer eingegangen. Die Data-Mining-Verfahren werden am Ende des Prozesses grundsätzlich erfolgreich auf den Zieldatensatz angewendet. Die Ergebnisse aller drei angewendet Verfahren sind beim ersten Beispieldatensatz jedoch nicht aussagekräftig. Das kann auf eine mangelhafte Datenvorverarbeitung, zu geringe Datenqualität oder die Auswahl falscher erklärender Attribute zurückzuführen sein.

Der zweite Datensatz wird mit der bekannten Zielsetzung einer Vorhersage analysiert. Auch hierbei können alle Prozessschritte ohne Probleme durchlaufen werden. Da auch hierbei eine Datenvorverarbeitung durchgeführt wird, können mithilfe der Unterklassifikatoren beim Durchlaufen des Prozesses Attribute für die weitere Analyse ausgeschlossen werden. Auch hier wäre ein weiterer Klassifikator in der Datenvorverarbeitung hilfreich. Am Ende des Prozesses

wird erfolgreich ein Regressionsmodell auf den Zieldatensatz angewendet und liefert aussagekräftige Ergebnisse, die auch in der Praxis Anwendung finden könnten.

Die Klassifikatoren können grundsätzlich erfolgreich als Prozessschritte beim Durchlaufen der EPK angewendet werden. Dabei sollen die Klassifikatoren auch Anwender unterstützen, die über weniger ausgeprägte Datenkompetenz verfügen. Die Unterklassifikatoren der Datenvorverarbeitung unterscheiden fehlende Werte in den Kategorien MAR, MCAR und NMAR. Wie in Unterkapitel 2.2.2 deutlich wird, ist ein bewusster Umgang mit fehlenden Werten bei einer Analyse essenziell, da es sonst zu verfälschten Ergebnissen kommen kann. Bei Anwendung des Unterklassifikators zur Bestimmung fehlender Werte wird bei beiden Beispieldatensätzen deutlich, dass dazu eine tiefergehende Analyse der Datengrundlage erforderlich ist. Bei der Analyse der fehlenden Werte gibt es keine Unterklassifikatoren die Anwender dabei unterstützen, die Kategorie der fehlenden Werte zu erkennen. Hierfür ist eine grundlegende Datenkompetenz erforderlich. Die Anwendung des Unterklassifikators die Kennzahl *Missing* zu bestimmen und anhand dieser Kennzahl über den weiteren Umgang zu entscheiden, kann erfolgreich umgesetzt werden und unterstützt dabei auch Anwender ohne herausragende Datenkompetenz. Auch die Entscheidungsunterstützung mithilfe der Kennzahl *Stability* hilft Anwendern bei der Entscheidungsfindung.

Bei der Anwendung auf die Beispieldatensätze wird deutlich, dass eine tiefergehende Datenvorverarbeitung sinnvoll wäre. Die Datenqualität wird lediglich anhand der Kennzahlen *Missing* und *Stability* untersucht. Zusätzlich wäre jedoch eine gründliche Data Exploration anhand weiterer Kennzahlen wie beispielsweise Korrelationen zwischen den unterschiedlichen Attributen von großem Mehrwert für die weitere Analyse. Dies wird insbesondere bei der Anwendung der Verfahren auf den Walmart Datensatz in Unterkapitel 4.1.1 deutlich. Hier könnten Attribute mit redundanten Informationen entfernt werden, die anhand von Korrelationen identifiziert werden könnten. Eine solche Data Exploration wird in Unterkapitel 2.3 näher erläutert. Der Informationsgehalt der Attribute wird bei der Datenvorverarbeitung im Konzept nicht berücksichtigt. Attribute wie *Type* und *Size* enthalten beide Informationen über die Größe der Filialen, weshalb zwischen diesen Attributen eine hohe Korrelation besteht. Wenn bei Regressions- oder Clusteranalysen erklärende Variablen verwendet werden, die stark miteinander korreliert sind, ist Multikollinearität gegeben, was die Ergebnisse der Analysen verzerren kann (Urban und Mayerl 2018). Dadurch erkennen Algorithmen Zusammenhänge die offensichtlich sind und für die Analyse schädlich sein können. Zusammenhänge können aufgrund der korrelierten Attribute von Algorithmen stärker gewichtet werden und überlagern dann weniger offensichtliche, aber wichtige Zusammenhänge. Dadurch können Ergebnisse verfälscht werden und nicht offensichtliche Zusammenhänge sind für Anwender schwerer zu erkennen. So hat der Algorithmus bei Anwendung der Clusteranalyse auf den Walmart Datensatz beispielsweise Cluster anhand der Filialtypen und Größe der Filialen gebildet, was ein eigentlich offensichtlicher Zusammenhang ist.

Beim Klassifikator der Umwandlung des Skalenniveaus und Bearbeiten der numerischen Werte wird unerfahrenen Anwendern nicht deutlich, wozu diese Umwandlungen dienen. Die Umwandlungen haben, wie in Unterkapitel 3.3.3 auch erwähnt wird, lediglich Auswirkungen auf die Performance der Verfahren, nicht auf die Anzahl der wählbaren Verfahren. Hier wären weitere

Unterklassifikatoren von Vorteil, die unerfahrenen Anwendern deutlich machen, wie sich die Transformationen im weiteren Verlauf auswirken.

Anhand der Beispieldatensätze konnte die grundsätzliche Funktionalität des entwickelten Klassifizierungskonzepts nachgewiesen werden. Zu Beginn dieses Unterkapitels wird bereits erwähnt, dass das Hauptziel dieser Arbeit grundsätzlich umgesetzt werden konnte. Dabei ist noch zu prüfen, wie das in dieser Arbeit entwickelte Konzept in das CRISP-DM Vorgehensmodell integriert werden kann. Bei einem Blick auf das in Abbildung 2-2 in Unterkapitel 2.1.2 abgebildete und dort näher erläuterte CRISP-DM Vorgehensmodell ist zu erkennen, dass das entwickelte Konzept in die Phasen Data Preparation und Modeling einzuordnen ist. Die Einordung in das Vorgehensmodell ist in Abbildung 4-14 anhand des roten Rahmens erkennbar.

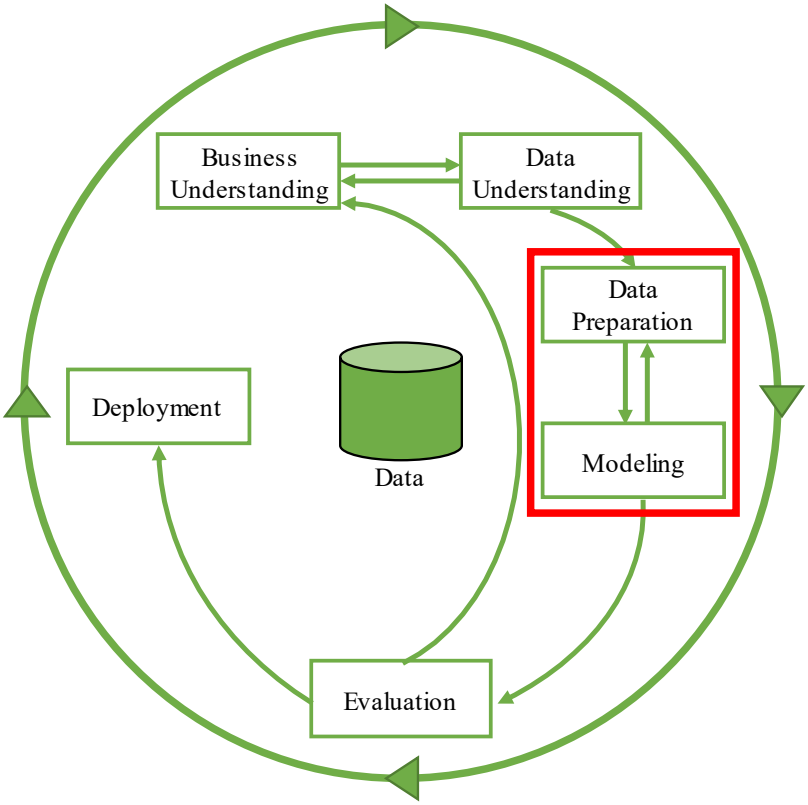


Abbildung 4-14: Einordnung des entwickelten Konzepts in das CRISP-DM Vorgehensmodell

Bei einem Blick auf die EPK wird jedoch schnell deutlich, dass diese nicht iterativ gestaltet ist. Das offenbart eine Schwäche des entwickelten Konzepts, da Data-Mining-Prozesse, wie in Kapitel 2 verdeutlicht wird, iterative Prozesse sind. Das als Kreislauf gestaltete CRISP-DM Vorgehensmodell verdeutlicht dies zusätzlich.

Insgesamt kann das entwickelte Konzept grundsätzlich in das CRISP-DM Vorgehensmodell integriert werden. Eine iterativere Gestaltung der EPK wäre dafür jedoch wünschenswert.

Anwendung der kennzahlenbasierten Validierung

Bei der Anwendung der Klassifikatoren auf den Walmart Datensatz sind folgende Klassifikatoren gegeben:

$$\begin{aligned} &Anzahl\ gegebener\ Klassifikatoren \\ &= Übergeordnetes\ Ziel\ bzw. Zweck\ der\ Analyse \\ &+ Skalenniveau\ bestimmen + Datenvorverarbeitung\ durchführen \\ &+ Qualität\ der\ Daten\ bestimmen + Skalenniveau\ bearbeiten \\ &+ Metrische\ Werte\ umwandeln + Data\text{–}Mining\text{–}Verfahren\ wählen \\ &= 7 \end{aligned}$$

Die Analyse des Beispieldatensatzes verdeutlicht, dass weitere Klassifikatoren als Unterstützung bei der Analyse hilfreich wären:

- Bei der Datenvorverarbeitung wäre nicht nur eine Bearbeitung des Skalenniveaus sinnvoll, sondern auch eine Umwandlung der Werte einiger Attribute. Die Einheit Fahrenheit könnte bei *Temperature* von Fahrenheit in Celsius geändert werden, um die Daten für Anwender verständlicher zu gestalten. Auch das Attribut *Date* kann in die jeweilige Kalenderwoche transformiert werden, da die Daten nur einmal pro Woche erfasst wurden.
- Bei der Datenvorverarbeitung können Attribute zusammengeführt oder entfernt werden, da diese den gleichen Informationsgehalt haben. Attribute wie *Type* und *Size* enthalten gleiche Informationen über die Filialen. Das Attribut *Size* kann ganz entfernt werden, da diese Information bereits in *Type* enthalten ist.

Somit sind mehr Klassifikatoren für eine zielführende Analyse notwendig:

$$\begin{aligned} &Anzahl\ notwendiger\ Klassifikatoren \\ &= Anzahl\ gegebener\ Klassfikatoren \\ &+ Umwandlung\ von\ Attributswerten \\ &+ Löschen\ von\ Attributen\ mit\ gleichen\ Informationen = 7 + 2 = 9 \end{aligned}$$

Damit ergibt sich der folgende Klassifikationsindex:

$$Klassifikationsindex = \frac{Anzahl\ gegebener\ Klassifkatoren}{Anzahl\ notwendiger\ Klassifikatoren} = \frac{7}{9} = 0{,}78$$

Ein Klassifikationsindex von 0,78 deutet darauf hin, dass für eine zielführende Anwendung des Klassifizierungskonzepts auf den Walmart Datensatz noch weitere Klassifikatoren nötig sind.

Bei der Analyse des Rossmann Datensatzes sind folgende Klassifikatoren durch das entwickelte Konzept gegeben:

$$\begin{aligned} &Anzahl\ gegebener\ Klassifikatoren \\ &= Übergeordnetes\ Ziel\ bzw. Zweck\ der\ Analyse \\ &+ Skalenniveau\ bestimmen + Datenvorverarbeitung\ durchführen \\ &+ Qualität\ der\ Daten\ bestimmen + Skalenniveau\ bearbeiten \\ &+ Metrische\ Werte\ umwandeln + Data\text{–}Mining\text{–}Verfahren\ wählen \\ &= 7 \end{aligned}$$

Es fällt auf, dass es sich um die gleichen gegebenen Klassifikatoren wie bei der Analyse des Walmart Datensatzes handelt. Da bei beiden Analysen eine Datenvorverarbeitung durchgerührt wird, liegt der Grund dafür beim Klassifikator der Datenvorverarbeitung und der dort integrierten Unterklassifikatoren. Bei Nichtdurchführung einer Datenvorverarbeitung fallen die Klassifikatoren Qualität der Daten bestimmen, Skalenniveau bearbeiten und Umwandeln in metrische Werte weg. Der Klassifikator Datenvorverarbeitung durchführen wird durch den Klassifikator Datenvorverarbeitung nicht durchführen ersetzt.

Auch bei dieser Analyse zeigt sich, dass noch ein weiterer Klassifikatoren zielführend wäre:

- Bei der Datenvorverarbeitung hätten Attribute zusammengeführt oder entfernt werden können, da diese die gleichen Informationen enthalten. Das Attribut *Store* enthält keine Informationen, die einen Mehrwert bringen. Das Attribut *StoreType* enthält die für die Analyse relevanten Informationen.

Damit wäre ein zusätzliche Klassifikator für eine zielführende Analyse notwendig gewesen:

$$\begin{aligned} &Anzahl\ notwendiger\ Klassifikatoren \\ &= Anzahl\ gegebener\ Klassfikatoren \\ &+ Löschen\ von\ Attributen\ mit\ gleichen\ Informationen = 7 + 1 = 8 \end{aligned}$$

Damit ergibt sich der folgende Klassifikationsindex:

$$Klassifikationsindex = \frac{Anzahl\ gegebener\ Klassifkatoren}{Anzahl\ notwendiger\ Klassifikatoren} = \frac{7}{8} = 0{,}88$$

Ein Klassifikationsindex von 0,88 zeigt, dass für eine zielführende Analyse zwar noch weitere Klassifikatoren sinnvoll wären, die Analyse mithilfe der Klassifikatoren aber gut funktioniert.

4.2.3 Anforderungserfüllung

Im Folgenden werden die in Unterkapitel 3.3.1 definierten Anforderungen auf ihre Erfüllung überprüft. Die in Unterkapitel 3.3.1 definierten Anforderungen können Tabelle 3-4 entnommen werden.

Die Muss-Anforderung A1 ist zwingend umzusetzen. Bei Nichterfüllung gilt das Konzept als nicht funktionsfähig. Ihre Priorität ist höher einzuordnen als die Soll-Anforderungen A2 und A3. Bei diesen handelt es sich um Anforderungen, die den Klassifikatoren einen Mehrwert bringen können, aber nicht essenziell für die Funktionalität des Konzepts sind. Eine detaillierte Beschreibung der definierten Anforderungen ist Unterkapitel 3.3.1 zu entnehmen.

Bei A1 handelt es sich um eine Muss-Anforderung. Wie bereits in Unterkapitel 4.2.2 deutlich wird, ist durch die Klassifikatoren sowohl mit als auch ohne bekannte Aufgabenstellung eine Übersicht über die auf die vorliegende Datengrundlage anwendbare Verfahren gegeben. Dies konnte erfolgreich mit den Beispieldatensätzen in Unterkapitel 4.1 angewendet werden. Die Anforderung A1 wird also erfüllt.

A2 ist eine Soll-Anforderung. Diese Anforderung wird mithilfe der Klassifikatoren Skalenniveau und Datenvorverarbeitung umgesetzt. Gehen Anwender lediglich von der vorliegenden Datengrundlage aus, ist intuitiv erkennbar, welche Verfahren mit oder ohne eine Datenvorverarbeitung auf das vorliegende Skalenniveau angewendet werden können. Soll ein bestimmtes Verfahren zum Einsatz kommen, liegt damit auch fest, wie das Skalenniveau zu bearbeiten ist, um das Verfahren anwenden zu können. So werden unnötige Iterationen bei der Datenvorverarbeitung vermieden. Ist die Menge an Attributen, die transformiert werden müssen, unverhältnismäßig hoch, kann eine Aufgabenstellung schon frühzeitig ausgeschlossen werden. Die Anforderung A2 wird somit vollständig erfüllt.

Bei Anforderung A3 handelt es sich ebenfalls um eine Soll-Anforderung. Die in Unterkapitel 3.2.2 in RapidMiner identifizierten Klassifikatoren konnten erfolgreich in das neu entwickelte Konzept integriert werden. Durch die Unterklassifikatoren werden Anwender bei der Datenvorverarbeitung unterstützt. Die Anforderung A3 wird somit ebenfalls vollständig erfüllt.

4.3 Fazit

Das in Kapitel 3 als EPK erarbeitete Systematisierungskonzept wird in Kapitel 4 erfolgreich auf zwei Beispieldatensätze angewendet. Für die prototypische Implementierung dient die Software RapidMiner.

Insgesamt wird bei der Anwendung des Klassifizierungskonzepts als EPK auf die Beispieldatensätze die Funktionstüchtigkeit der Anwendung der definierten Klassifikatoren zur Systematisierung von Data-Mining-Verfahren nachgewiesen. Über eine Betrachtung des vorhandenen Datenbestandes wird mithilfe der EPK schon vor Beginn der eigentlichen Analyse untersucht, welche Verfahren durch welche Bearbeitungsschritte auf die Daten angewendet werden können. Das Hauptziel dieser Arbeit wurde also erfolgreich als Konzept entwickelt und beispielhaft angewendet.

Im ersten Beispiel wird das Konzept auf einen Datensatz des Einzelhandelsunternehmens Walmart angewendet. In dieser Anwendung wird keine Zielsetzung oder Aufgabenstellung unterstellt. Aufgrund der Klassifikatoren Ziel/Zweck mit unbekanntem Ziel und dem Klassifikator Skalenniveau mit einem gemischten Skalenniveau kann ohne die Durchführung einer Datenvorverarbeitung lediglich ein Robust Learner auf den Datensatz angewendet werden. Im Folgenden wird eine Datenvorverarbeitung durchgeführt, um eine größere Auswahl an Data-Mining-Verfahren zur Bearbeitung des Datensatzes zu erhalten. Die Datenvorverarbeitung wird analog des in der EPK dargestellten Vorgehens auf den Datensatz angewendet. Dazu wird zuerst mithilfe der Kennzahlen *Stability* und *Missing* die Qualität der Daten bestimmt. Aufgrund der festgelegten Schwellenwerte zur Löschung von Attributen mithilfe der Kennzahl *Stability* können Entscheidungen über das Entfernen von Attributen aufgrund zu hoher Kennwerte der Stabilität getroffen werden. Daraufhin werden, wie in der EPK dargestellt, die fehlenden Werte genau

untersucht und die definierten Schwellenwerte entsprechend behandelt. Nach Prüfung der Datenqualität wird aufgrund der hohen Anzahl numerischer Attribute entschieden, die restlichen Attribute in numerische Werte umzuwandeln. Nach diesen Bearbeitungsschritten stehen die folgenden Data-Mining-Verfahren zur Verfügung: Anwendung einer Clusteranalyse, Anwendung einer numerischen Vorhersage sowie Anwendung einer Clusteranalyse und Ableiten von Regeln. Alle drei Verfahren werden erfolgreich auf den Datensatz angewendet.

Im zweiten Beispiel wird das Konzept mit einem Datensatz der Drogeriemarktkette Rossmann erprobt. Hierbei wird die Zielsetzung im Vorhinein festgelegt. Auch hier ist aufgrund der Klassifikatoren Ziel/Zweck mit bekanntem Ziel der Vorhersage und dem Klassifikator Skalenniveau mit einem gemischten Skalenniveau erkennbar, dass bei Verzicht auf eine Datenvorverarbeitung lediglich ein Robust Learner auf den Datensatz angewendet werden kann. Bei der Anwendung wird entschieden, eine Datenvorverarbeitung durchzuführen, um eine größere Auswahl an Data-Mining-Verfahren zur Verfügung zu haben. Die Datenvorverarbeitung wird analog des in der EPK dargestellten Vorgehens auf den Datensatz angewendet. Zuerst wird dazu mithilfe der Kennzahlen *Stability* und *Missing* die Qualität der Daten bestimmt. Auch bei dieser Anwendung wird aufgrund der definierten Schwellenwerte zur Löschung von Attributen mithilfe der Kennzahl *Stability* ein Attribut entfernt. Analog des in der EPK dargestellten Vorgehens, werden die fehlenden Werte analysiert und entsprechend der definierten Schwellenwerte behandelt. Nach Prüfung der Datenqualität wird entschieden, das Skalenniveau der Attribute nicht zu bearbeiten, da dabei wichtige Informationen verloren gehen können. Als Data-Mining-Verfahren steht danach noch ein Regressionsmodell zur Verfügung. Das Verfahren kann erfolgreich auf den Datensatz angewendet werden.

Anhand der Systematisierung der Data-Mining-Verfahren mithilfe von Klassifikatoren kann schon früh erkannt werden, welche verfahrensabhängigen Vorbereitungsschritte durchgeführt werden müssen, wenn ein bestimmtes Verfahren angewendet werden soll. Mithilfe des entwickelten Konzepts können so Iterationen innerhalb des Data-Mining-Prozesses zwischen dem Data Mining selbst und den vorbereitenden Schritten vermieden werden.

Durch die Unterklassifikatoren in der Datenvorverarbeitung wird Anwendern mit eingeschränkter Datenkompetenz ermöglicht, eine grundlegende Datenvorverarbeitung durchzuführen. Bei Anwendung auf die Beispieldatensätze wird jedoch deutlich, dass die Unterklassifikatoren für eine optimale Analyse noch weiterentwickelt werden müssen. In der Datenvorverarbeitung sind also weitere Unterklassifikatoren sinnvoll. Neben der Betrachtung der Kennzahlen für fehlende Werte und identische Werte innerhalb eines Attributs ist bei der Analyse der Beispieldatensätze auch eine kritische Betrachtung des Informationsgehalts der Attribute und eine Umwandlung einiger Werte sinnvoll. Dies gilt sowohl für den Datensatz von Walmart als auch für den von Rossmann.

Die Funktionalität der Klassifikatoren zur Systematisierung von Data-Mining-Verfahren wird anhand der Beispieldatensätze nachgewiesen. Bei Anwendung des Konzepts liegt der Fokus dieser Arbeit auf der Funktionalität der Klassifikatoren. Bei der Analyse des Walmart Datensatzes konnten zwar alle nach den Klassifikatoren systematisierte Verfahren auf den Zieldatensatz angewendet werden, die Ergebnisse der Data-Mining-Verfahren waren dabei jedoch von geringer Qualität. Dies ist wahrscheinlich auf die Untersuchung der Datenqualität lediglich mit Hilfe der

Kennzahlen *Missing* und *Stability* zurückzuführen. Tiefergehende Betrachtungen der Attribute, wie beispielsweise mit Instrumenten der in Unterkapitel 2.3 näher erläuterten Data Exploration, werden bei den Klassifikatoren nicht vorgenommen. Das hat auch zur Folge, dass der Informationsgehalt der Attribute nicht berücksichtigt wird. Dies wird, wie in Unterkapitel 4.2.2 näher erläutert, bei der Anwendung des Konzepts auf den Walmart Datensatz deutlich. Dort enthalten Attribute teilweise redundante Informationen. Dadurch erkennen Algorithmen möglicherweise Zusammenhänge, die keinen Mehrwert für eine Analyse bringen. Unter diesen Voraussetzungen können Ergebnisse nicht zielführend oder sogar verfälscht sein.

5 Zusammenfassung und Ausblick

Die vorliegende Arbeit hatte das Hauptziel, eine neue Systematisierung von Data-Mining-Verfahren anhand von Klassifikatoren zu erarbeiten. Durch die Anwendung dieses Konzepts wird die Analyse eines Datenbestandes ermöglicht, ohne vorher eine Aufgabenstellung definieren zu müssen. Grundlage dafür ist eine strukturierte Betrachtung des Datenbestandes schon vor Beginn der Analyse. Das entwickelte Konzept wurde mithilfe der Software RapidMiner anhand öffentlich zugänglicher Beispieldatensätzen aus einem ökonomischen und produktionslogistischen Umfeld angewandt und erfolgreich validiert.

Um das Konzept zu entwickeln, wurden in Kapitel 2 die Grundlagen in Data-Mining-Prozessen erarbeitet. Dabei wurde deutlich, dass es sich beim CRISP-DM Vorgehensmodell um das für diese Arbeit im Vergleich zum Vorgehensmodell nach Fayyad geeignetere Vorgehensmodell handelt. Ebenfalls wurden der zeitlich hohe Aufwand und die Wichtigkeit der datenvorverarbeitenden Schritte eines Data-Mining-Prozesses deutlich. Eine signifikante Herausforderung bei Anwendung von Data-Mining-Verfahren besteht dabei in der grundlegenden Datenkompetenz der Anwender.

Nach Erarbeitung der Grundlagen wurde in Kapitel 3 ein neues Konzept entwickelt, mit dem Data-Mining-Verfahren anhand von Klassifikatoren systematisiert werden können. Zusätzlich wurde in Unterkapitel 3.2 untersucht, inwieweit RapidMiner Anwender bei der Datenvorverarbeitung mithilfe integrierter Werkzeuge unterstützt. Dieser geführte Ansatz zur Datenvorverarbeitung wurde mithilfe von Datensätzen aus der realen Welt, bei denen das Skalenniveau und die Menge an fehlenden Werten vorab gezielt bearbeitet wurde, systematisch untersucht. Hierbei zeigte sich, dass Entscheidungen von RapidMiner in der geführten Datenvorverarbeitung unter anderem von Kennzahlen ausgemacht werden: *Missing* und *Stability*. Um die von RapidMiner genutzten Klassifikatoren zu identifizieren, wurde der Prozess der Datenvorverarbeitung in *Auto Cleansing* als EPK dargestellt. Auf diese Weise wurden in der Funktion *Auto Cleansing* drei Klassifikatoren identifiziert: Zielattribut definieren, Datenqualität bestimmen, Skalenniveau bearbeiten. Bei der nachfolgenden Entwicklung des Konzepts wurde geprüft, inwieweit die in RapidMiner identifizierten Klassifikatoren als Unterklassifikatoren in das eigene Konzept integriert werden können.

Für die Entwicklung des Konzepts wurden zuerst Anforderungen mit unterschiedlichen Prioritäten an Klassifikatoren definiert, um eine systematische Entwicklung und anschließende Validierung zu ermöglichen. Anschließend wurden Klassifikatoren und ein Klassifizierungskonzept als theoretisches Modell entwickelt. Nach Anwendung der Klassifikatoren liegt im theoretischen Konzept ein Zieldatensatz vor, auf den Data-Mining-Verfahren angewendet werden können. Um die Klassifikatoren als Prozess darzustellen, wurde das Klassifizierungskonzept in Unterkapitel 3.3.3 ausführlich als EPK ausgearbeitet und dargestellt. Das Hauptziel der Arbeit, anhand der Kriterien eines Datenbestandes Data-Mining-Verfahren zu systematisieren, ohne vorher zwingend eine Aufgabenstellung definieren zu müssen, wurde damit erfüllt.

Weiteres Ziel dieser Arbeit war eine prototypische Anwendung und Validierung des Konzepts an öffentlich zugänglichen Beispieldatensätzen mithilfe der Software RapidMiner. Bei Anwendung des Konzepts lag der Fokus dieser Arbeit auf der Funktionalität der Klassifikatoren.

Bei der ersten Anwendung wurde das Konzept erfolgreich auf einen Datensatz eines Einzelhandelskonzerns angewendet. Bei der zweiten Anwendung wurde das Konzept ebenfalls erfolgreich auf einen Datensatz einer Drogeriemarktkette angewendet.

Für die Validierung der Klassifikatoren wurde eine explorative und eine kennzahlenbasierte Validierungsmethode vorgestellt und anschließend angewendet. Bei der Anwendung der Validierungsmethoden wurde festgestellt, dass das erarbeitete Systematisierungskonzept erfolgreich auf die Beispieldatensätze angewendet werden konnte. Anhand der Beispieldatensätze konnte also nachgewiesen werden, dass die Klassifikatoren zur Systematisierung von Data-Mining-Verfahren funktionieren. Die zuvor definierten Anforderungen konnten dabei erfüllt werden. Bei der Validierung wurde jedoch auch festgestellt, dass für eine gründlichere Datenvorverarbeitung noch weitere Unterklassifikatoren nötig sind. Attribute hätten aufgrund der enthaltenen Informationen entfernt oder Werte aufgrund ihrer Einheiten umgewandelt werden können. Dies wird mithilfe des in Unterkapitel 4.2.1 definierten Klassifikationsindex in Unterkapitel 4.2.2 quantifiziert. Ebenfalls wurde festgestellt, dass die Ergebnisse der angewendeten Data-Mining-Verfahren nicht immer zielführend und aussagekräftig sind. Insgesamt werden die definierten Anforderungen jedoch vollständig erfüllt.

In zukünftigen Arbeiten könnte auf Grundlage des in dieser Arbeit entwickelten Konzepts ein stärkerer Fokus auf eine unterstützende Datenvorverarbeitung gelegt werden. Dabei könnte neben der reinen Systematisierung von Data-Mining-Verfahren die Performance und die Aussagekraft der Ergebnisse in den Fokus gerückt werden. So könnten die Unterklassifikatoren weiterentwickelt werden. Neben der Betrachtung der in dieser Arbeit verwendeten Kennzahlen zur Datenvorverarbeitung ist eine tiefergehende Betrachtung der vorliegenden Attribute, der enthaltenen Werte und des Informationsgehalts der Attribute sinnvoll. Hierzu könnte ein weiterer Unterklassifikator zur Data Exploration entwickelt werden, um weitere Kennzahlen wie Korrelationen zwischen Attributen in das Konzept zu integrieren. Außerdem wäre es sinnvoll, die Umwandlung von Werten zu berücksichtigen. Ebenso ist das in dieser Arbeit entwickelte Konzept nicht rekursiv. In einem weiteren Schritt könnte das Konzept rekursiv gestaltet und in ein Vorgehensmodell für Data-Mining-Prozesse integriert werden. Auf diese Weise müssten die vorliegenden Datensätze nicht von Beginn an zusammengeführt werden, sondern könnten einzeln iterativ mithilfe der weiter entwickelten Unterklassifikatoren zur Datenvorverarbeitung analysiert, bearbeitet und im Laufe des Prozesses zusammengeführt werden. Dieses weiterentwickelte Konzept könnte mit Blick auf die Performance der angewendeten Verfahren in Form eines gesamten Data-Mining-Prozesses angewendet und validiert werden. Auf diese Weise könnten die Ergebnisse eines Data-Mining-Prozesses ohne das integrierte Klassifizierungskonzept mit den Ergebnissen eines Data-Mining-Prozesses mit integriertem Klassifizierungskonzept verglichen werden. So könnte auch der Nutzen von Klassifikatoren für das Analyseergebnis mithilfe dazu entwickelter Kennzahlen quantitativ gemessen und validiert werden. Der in dieser Arbeit entwickelte Klassifikationsindex bewertet lediglich, ob weitere Klassifikatoren für die Analyse zielführend gewesen wären und nicht, ob das Analyseergebnis ohne Anwendung des Konzepts besser ausgefallen wäre. Eine Weiterentwicklung des Klassifikationsindex mit Berücksichtigung der Analyseergebnisse wäre ebenfalls zielführend.

Zahlreiche für das Jahr 2023 angesetzte Konferenzen wie die *23th Industrial Conference on Data Mining*, die *6th International Conference on Big Data and Smart Computing* oder die *4th International Conference on Data Science, E-learning and Information Systems* zu den Themengebieten Data Mining und Big Data zeigen die Aktualität des Forschungsfeldes dieser Arbeit. Eine Thematisierung der Strukturierungsformen von Data-Mining-Verfahren oder einer unterstützenden Datenvorverarbeitung ist während dieser Konferenzen bisher nicht vorgesehen. Bei der *13th International Conference on Learning Analytics & Knowledge (LAK)* im Jahr 2023 wird die in dieser Arbeit identifizierte Herausforderungen der erforderlichen Datenkompetenz thematisiert. Die Ergebnisse dieser Konferenzen können für zukünftige Arbeiten, die sich mit neuen Systematisierungsformen von Data-Mining-Verfahren oder einer unterstützenden Datenvorverarbeitung beschäftigen von Interesse sein. Da die Funktionalität eines Systematisierungskonzept anhand von Klassifikatoren nachgewiesen wurde, bieten sich weitere Arbeiten zur Systematisierung anhand von Klassifikatoren an.

6 Literaturverzeichnis

Arunadevi, J.; Ramya, S.; Ramesh Raja, M. (2018): A study of classification algorithms using Rapidminer. In: International Journal of Pure and Applied Mathematics, 119 (12), 15977-15988.

Aust, H. (2021): Das Zeitalter der Daten. Was Sie über Grundlagen, Algorithmen und Anwendungen wissen sollten. Berlin, Heidelberg: Springer.

Bachmann, R. (2014): Big Data - Fluch oder Segen? Unternehmen im Spiegel gesellschaftlichen Wandels. Frechen: mitp.

Beekmann, F.; Chamoni, P. (2006): Verfahren des Data Mining. In: Chamoni, P.; Gluchowski, P. (Hg.). Analytische Informationssysteme. Business Intelligence-Technologien und -Anwendungen. Berlin, Heidelberg: Springer, 263–282.

Bramer, M. (2020): Principles of Data Mining. 4. Aufl. London: Springer London; Imprint Springer.

Chapman, P.; Clinton, J.; Kerber, R.; Khabaza, T.; Reinartz, T.; Shearer, C.; Wirth, R. (2000): CRISP-DM 1.0: Step-by-step data mining guide. CRISP-DM Consortium: NCR Systems Engineering Copenhagen (USA and Denmark), DaimlerChrysler AG (Germany), SPSS Inc. (USA), and OHRA Verzekeringen en Bank Groep B.V.

Che, D.; Safran, M.; Peng, Z. (2013): From Big Data to Big Data Mining: Challenges, Issues, and Opportunities. In: Hutchison, F.; Kanade, T.; Kittler, J. et al. (Hg.). Database systems for advanced applications. 18th international conference, DASFAA 2013, international workshops: BDMA, SNSM, SeCoP, Proceedings, Wuhan, 22. bis 25.04.2013. Berlin: Springer, 1–15.

Cios, K. J.; Kurgan, L. A. (2005): Trends in Data Mining and Knowledge Discovery. In: Advanced Techniques in Knowledge Discovery and Data Mining. London: Springer, 1–26.

Cleve, J.; Lämmel, U. (2020): Data Mining. 3. Aufl. Berlin, Boston: De Gruyter.

Düsing, R. (2006): Knowledge Discovery in Databases. In: Chamoni, P.; Gluchowski, P. (Hg.). Analytische Informationssysteme. Business Intelligence-Technologien und -Anwendungen. Berlin, Heidelberg: Springer, 241–262.

Ester, M.; Sander, J. (2000): Knowledge discovery in databases. Techniken und Anwendungen. Berlin, Heidelberg: Springer.

Fayyad, U.; Piatetsky-Shapiro, G.; Smyth, P. (1996a): From Data Mining to Knowledge Discovery in Databases. In: AI Magazine, 17 (3), 37–54.

Fayyad, U.; Piatetsky-Shapiro, G.; Smyth, P. (1996b): Knowledge Discovery and Data Mining: Towards a Unifying Framework. In: Proc. 2nd Int. Conf. on Knowledge Discovery and Data Mining, Portland, 82–88.

Fayyad, U.; Piatetsky-Shapiro, G.; Smyth, P. (1996c): The KDD process for extracting useful knowledge from volumes of data. In: Communications of the ACM, 39 (11), 27–34.

Fayyad, U. ;Piatetsky-Shapiro, G. ;Smyth, P. ;Uthurusamy, R. (Hg.) (1996d): Advances in knowledge discovery and data mining. 5. Aufl. Menlo Park, Calif.: AAAI Press.

Gandomi, A.; Haider, M. (2015): Beyond the hype: Big data concepts, methods, and analytics. In: International Journal of Information Management, 35 (2), 137–144.

García, S.; Luengo, J.; Herrera, F. (2015): Data Preprocessing in Data Mining. Cham: Springer International Publishing.

Gartner, Inc. (2022): Definition of Big Data - Gartner Information Technology Glossary. Online verfügbar unter https://www.gartner.com/en/information-technology/glossary/big-data (abgerufen am 25.09.2022).

Geng, L.; Hamilton, H. J. (2006): Interestingness measures for data mining. In: ACM Computing Surveys, 38 (3), 9.

Hand, D.; Mannila, H.; Smyth, P. (2001): Principles of data mining. In: Drug Saf., 30 (7), 621–622.

Hettich, G.; Jüttler, H.; Luderer, B. (2009): Mathematik für Wirtschaftswissenschaftler und Finanzmathematik. Mit Aufgaben und Lösungen. 10. Aufl. München: Oldenbourg.

Huang, G. (2021): Missing data filling method based on linear interpolation and lightgbm. In: Journal of Physics: Conference Series, 1754 (1), 12187.

Kaiser, J. (2014): Dealing with Missing Values in Data. In: Journal of Systems Integration, 5 (1), 42–51.

Kreiß, J.-P.; Neuhaus, G. (2006): Einführung in die Zeitreihenanalyse. Berlin, Heidelberg: Springer.

Kurgan, L. A.; Musilek, P. (2006): A survey of Knowledge Discovery and Data Mining process models. In: The Knowledge Engineering Review, 21 (1), 1–24.

Lexa, C. (2021): Fit für die digitale Zukunft. Wiesbaden: Springer Fachmedien Wiesbaden.

Ludwig, T.; Thiemann, H. (2020): Datenkompetenz – Data Literacy. In: Informatik Spektrum, 43 (6), 436–439.

Mandalapu, V.; Gong, J. (2019): Studying Factors Influencing the Prediction of Student STEM and Non-STEM Career Choice. In: Desmarais, M. C.; Lynch, C. F.; Merceron, A. et al. (Hg.). Proceedings of the 12th International Conference on Educational Data Mining. EDM 2019, Montréal, 02. bis 05.07.2019, 607–610.

Mierswa, I.; Wurst, M.; Klinkenberg, R.; Scholz, M.; Euler, T. (2006): YALE: Rapid prototyping for complex data mining tasks. In: Ungar, L. (Hg.). Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, the 12th ACM SIGKDD international conference, Philadelphia, 20. bis 23.08.2006. New York: ACM Press, 935–940.

Muhammad Habib, R.; Liew, C. S.; Abbas, A.; Jayaraman, P. P.; Wah, T. Y.; Khan, S. U. (2016): Big Data Reduction Methods: A Survey. In: Data Science and Engineering, 1 (4), 265–284.

Noor, N. M.; Al Bakri Abdullah, M. M.; Yahaya, A. S.; Ramli, N. A. (2014): Comparison of Linear Interpolation Method and Mean Method to Replace the Missing Values in Environmental Data Set. In: Materials Science Forum, 803, 278–281.

Petersohn, H. (2005): Data Mining. Verfahren, Prozesse, Anwendungsarchitektur. München, Wien: Oldenbourg.

Pratama, I.; Permanasari, A. E.; Ardiyanto, I.; Indrayani, R. (2016): A review of missing values handling methods on time-series data. In: 2016 International Conference on Information Technology Systems and Innovation (ICITSI), 2016 International Conference on Information Technology Systems and Innovation (ICITSI), Bandung, Bali, 24. bis 27.10.2016. IEEE, 1–6.

RapidMiner (2014): RapidMiner Studio Manual. Boston. Online verfügbar unter https://docs.rapidminer.com/downloads/RapidMiner-v6-user-manual.pdf.

RapidMiner (2022): A guided approach to RapidMiner Studio - RapidMiner Documentation. Online verfügbar unter https://docs.rapidminer.com/latest/studio/guided/ (abgerufen am 13.09.2022).

RapidMiner (2022b): RapidMiner Studio. Version 9.10.007. Online verfügbar unter https://rapidminer.com/platform/educational/ (abgerufen am 12.08.2022).

RapidMiner (2022): Turbo Prep - RapidMiner Documentation. Online verfügbar unter https://docs.rapidminer.com/latest/studio/guided/turbo-prep/ (abgerufen am 13.09.2022).

Reinsel, D.; Gantz, J.; Rydning, J. (2018): The Digitization of the World. From Edge to Core. IDC White Paper - #US44413318. IDC. Online verfügbar unter https://www.seagate.com/files/www-content/our-story/trends/files/idc-seagate-dataage-whitepaper.pdf (abgerufen am 08.06.2022).

Rossmann GmbH (2015a): Rossmann Store Sales - Data. Online verfügbar unter https://www.kaggle.com/competitions/rossmann-store-sales/data?select=store.csv (abgerufen am 29.09.2022).

Rossmann GmbH (2015b): Rossmann Store Sales - Overview. Online verfügbar unter https://www.kaggle.com/competitions/rossmann-store-sales/overview (abgerufen am 29.09.2022).

Runkler, T. A. (2010): Data Mining. Methoden und Algorithmen intelligenter Datenanalyse. Wiesbaden: Vieweg+Teubner.

Runkler, T. A. (2015): Data Mining. Modelle und Algorithmen intelligenter Datenanalyse. 2. Aufl. Wiesbaden: Springer Fachmedien Wiesbaden.

Scheidler, A. A.; Rabe, M. (2021): Integral verification and validation for knowledge discovery procedure models. In: Int. J. Business Intelligence and Data Mining (1), 73–87.

Schüller, K.; Busch, P.; Hindinger, C. (2019): Future Skills: Ein Framework für Data Literacy. 47. Online verfügbar unter https://hochschulforumdigitalisierung.de/sites/default/files/dateien/HFD_AP_Nr_47_DALI_Kompetenzrahmen_WEB.pdf (abgerufen am 01.08.2022).

Scopus (2022): Fayyad, U. Scopus. Online verfügbar unter https://www.scopus.com/hirsch/author.uri?accessor=authorProfile&auidList=660372211 9&origin=AuthorProfile&display=hIndex (abgerufen am 20.07.2022).

Singh, M. (2017): Retail Data Analytics. Online verfügbar unter https://www.kaggle.com/datasets/manjeetsingh/retaildataset (abgerufen am 15.09.2022).

Sohrabei, S.; Salari, R.; Ayyoubzadeh, S. M.; Atashi, A. A. (2022): Prediction Axillary Lymph Node Involvement Status on Breast Cancer Data. In: Multidisciplinary Cancer Investigation, 6 (2), 1–9.

Stahl, R.; Staab, P. (2017): Die Vermessung des Datenuniversums. Berlin, Heidelberg: Springer Berlin Heidelberg.

Stocker, T. C.; Steinke, I. (2022): Statistik. Grundlagen und Methodik. 2. Aufl. Berlin: De Gruyter Oldenbourg.

Urban, D.; Mayerl, J. (2018): Angewandte Regressionsanalyse: Theorie, Technik und Praxis. Wiesbaden: Springer Fachmedien Wiesbaden.

Völkl, K.; Korb, C. (2018): Variablen und Skalenniveaus. In: Deskriptive Statistik. Springer VS, Wiesbaden, 7–28.

Walmart Inc. (2014): Walmart Recruiting - Store Sales Forecasting. Online verfügbar unter https://www.kaggle.com/competitions/walmart-recruiting-store-sales-forecasting/overview/description (abgerufen am 21.09.2022).

Wirth, R.; Hipp, J. (2000): CRISP-DM: Towards a standard process model for data mining. In: Mackin, N. (Hg.). Proceedings of the Fourth International Conference on the Practical Application of Knowledge Discovery and Data Mining, Crowne Plaza Midland Hotel, Manchester, 11. bis 13.08.2000. Blackpool, Lancashire: Practical Application Company, 29–39.

Wolff, A.; Gooch, D.; Cavero Montaner, J. J.; Rashid, U.; Kortuem, G. (2016): Creating an Understanding of Data Literacy for a Data-driven Society. In: The Journal of Community Informatics, 12 (3).

Zdravevski, E.; Lameski, P.; Kulakov, A. (2011): Weight of evidence as a tool for attribute transformation in the preprocessing stage of supervised learning algorithms. In: Proceedings of International Joint Conference on Neural Networks, 31.07. bis 05.08.2011. San Jose.

Zhang, S.; Zhang, C.; Yang, Q. (2003): Data preparation for data mining. In: Applied Artificial Intelligence, 17 (5-6), 375–381.

Anhang

Anhang A: Ereignisgesteuerte Prozessketten

EPK Turbo Prep: EPK_Turbo-Prep.png
EPK Detailkonzept Phase 1: EPK_Detailkonzept-Phase-1.png
EPK Detailkonzept Phase 2: EPK_Detailkonzept-Phase-2.png

Anhang B: Beispieldatensätze und RapidMiner-Prozesse

Walmart Datensatz: Retail_Data_Analytics.zip
Rossmann Datensatz: rossmann-store-sales.zip
Anwendung-des-Klassifizierungskonzepts_Walmart-Datensatz.rmp
Anwendung-des-Klassifizierungskonzepts_Rossmann-Datensatz.rmp