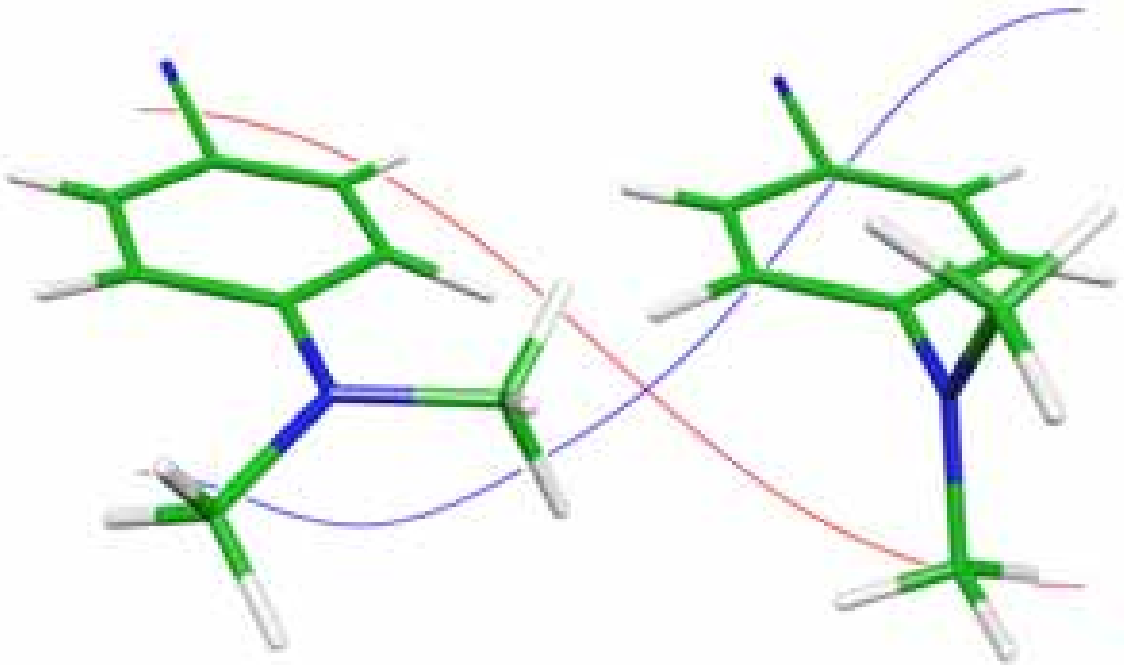
Uwe Steinlein

Data Mining als Instrument der Responseoptimierung im Direktmarketing:

Methoden zur Bewältigung niedriger Responsequoten



Cuvillier Verlag Göttingen

Data Mining als Instrument der Responseoptimierung im Direktmarketing: Methoden zur Bewältigung niedriger Responsequoten

Dissertation
zur Erlangung des Doktorgrades
der Wirtschaftswissenschaftlichen Fakultät
der Universität Augsburg

vorgelegt von
Dipl.-Kfm. Uwe Steinlein

Erstgutachter: Prof. Dr. Otto Opitz
Zweitgutachter: PD Dr. Andreas Hilbert
Vorsitzender der mündlichen Prüfung: Prof. Dr. Günter Bamberg
Tag der mündlichen Prüfung: 28.11.2003

Augsburg, im Oktober 2003

Bibliografische Information Der Deutschen Bibliothek
Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über http://dnb.ddb.de abrufbar.
1. Aufl. - Göttingen : Cuvillier, 2004
Zugl.: Augsburg, Univ., Diss., 2003
ISBN 3-89873-981-3

Nonnenstieg 8, 37075 Göttingen
Telefon: 0551-54724-0
Telefax: 0551-54724-21
www.cuvillier.de

1. Auflage, 2004
Gedruckt auf säurefreiem Papier

ISBN 3-89873-981-3

Für meine Eltern

Danksagung

Ich bedanke mich an erster Stelle bei meinem Doktorvater, Herrn Professor Dr. Otto Opitz von der Universität Augsburg. Er hat mich als externen Doktoranden aufgenommen, durch zahlreiche interessante und kreative Diskussionen das Entstehen und die Entwicklung der Arbeit unterstützt und war mir stets durch sein profundes Fachwissen eine Hilfe.
Auch danke ich Herrn PD Dr. Andreas Hilbert als Zweitgutachter und Herrn Professor Dr. Günter Bamberg als Prüfungsvorsitzenden der mündlichen Prüfung.

Auch möchte ich mich bei Herrn Dr. Spiess, Herrn Blum und Herrn Glaser bedanken, die mir seitens WEKA MEDIA eine externe Promotion mit Praxisbezug ermöglichten. Für die Erklärung der umfangreichen Datenbank und der Beantwortung meiner zahlreichen Nachfragen zum Datenverständnis danke ich vor allem Herrn Lauxtermann und Frau Rampf.

Weiterhin bedanke ich mich insbesondere bei Frau Theusinger und Herrn Österer vom SAS Institute für die Aufnahme in das SAS Fellowship Programm und die stets gewährte Unterstützung.

Nicht zuletzt möchte ich meinen Eltern und meinen Freunden für Ihre Unterstützung danken.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1. Problemüberblick und Zielrichtung

Unternehmen sammeln viele Daten über ihre Kunden, ihr Kontakt- und Umsatzverhalten. Diese Daten werden hauptsächlich zur Abwicklung der Geschäftsprozesse benötigt. Allerdings kann auch das Marketing bzw. der Vertrieb von dieser Datengrundlage profitieren. Im Rahmen der Kundenbewerbung können diese Informationen bei einer Direktmarketingaktion für eine gezielte Selektion von Kunden genutzt werden, um Streuverluste zu vermeiden. Dabei können die Selektionskriterien entweder durch einen Experten im Unternehmen oder mit Hilfe mathematischer bzw. statistischer Verfahren bestimmt werden. Der Einsatz von mathematischen und statistischen Verfahren im Marketing zur Analyse von großen Datenmengen ist sowohl aufgrund der Leistungsfähigkeit von Computern als auch durch die Entwicklung effizienter Algorithmen positiv beeinflusst worden. Das Ziel für Unternehmen ist dabei, eine möglichst individuelle Kundenbeziehung aufzubauen, indem aus dem vergangenen Verhalten des Kunden gelernt wird. Es wird versucht, Kundenbedürfnisse zu antizipieren und die Kommunikation mit dem Kunden zu optimieren. Großes Einsparpotenzial liegt vor allem in der Reduzierung von Streuverlusten bei Direktmarketingaktionen. Dabei werden die Kunden in der Datenbank nach ihrer Attraktivität für das Unternehmen bewertet und anschließend nur die am besten geeigneten Adressen zur Bewerbung ausgewählt. In dieser Arbeit werden verschiedene Vorgehensweisen und Verfahren zur Ermittlung der Kundenattraktivität für ein bestimmtes Produkt vorgestellt, angewandt und anhand einer empirischen Studie verglichen. Der Fall niedriger Responsequoten und großer Datenmengen wird dabei im Speziellen untersucht.

In diesem Kapitel werden nachfolgend die Begriffe Marketing, Direktmarketing, Databasemarketing, CRM und Responseoptimierung definiert. Im Anschluss wird die konkrete Problemstellung und deren Praxisrelevanz gezeigt, bevor die Zielstellung und der Aufbau dieser Arbeit vorgestellt werden.

1.1 Begriffsdefinitionen

Im Allgemeinen wird **Marketing** definiert als „eine menschliche Tätigkeit, die darauf abzielt, durch Austauschprozesse Bedürfnisse und Wünsche zu befriedigen“ (Kotler, 1989, S. 19). Marketing Management ist nach Kotler (1989, S. 23) Planung, Realisation

und Kontrolle von Programmen, mit deren Hilfe bestimmte Austauschprozesse mit ausgewählten Märkten geschaffen, aufgebaut und aufrechterhalten werden sollen, um betriebliche Ziele zu verwirklichen. Zur Erreichung dieser Ziele dienen die vier Marketinginstrumente Produktpolitik, Preispolitik, Distributionspolitik und Kommunikationspolitik (Gierl, 1995, S. 25):
Unter Produktpolitik werden dabei alle Entscheidungen verstanden, welche die Zusammensetzung der Absatzleistung eines Unternehmens bestimmen, beispielsweise die Festlegung des Produktionsprogramms und die Produkteigenschaften.
Zur Preis- oder Kontrahierungspolitik zählt die Preisfestlegung und -änderung, Rabattgewährung sowie die Gestaltung der Zahlungsbedingungen.
Die physische Distribution der Ware, die Wahl der Absatzwege und die Gestaltung des Vertriebs zählen zur Distributionspolitik.
Die Kommunikationspolitik umfasst die planmäßige Gestaltung aller Aktivitäten zur Übermittlung der auf den Markt gerichteten Informationen zum Zweck der Beeinflussung von Einstellungen und Verhaltensweisen.

Für den Begriff **Direktmarketing** existieren verschiedene Definitionen. Eine sehr einfache Definition liefert Bird (1989, S. 28): „Direct Marketing is any advertising activity which creates and exploits a direct relationship between you and your prospect or customer as an individual." Die Hauptaussage ist dabei, dass es sich beim Direktmarketing um direkte Kommunikation handelt, die eine individuelle und direkte Beziehung zwischen dem Unternehmen und dem Kunden bzw. Interessenten entstehen lässt.
Die Definition von Stone (1989, S. 3) geht einen Schritt weiter und berücksichtigt sowohl die Messbarkeit der Reaktionen auf Direktmarketingaktionen als auch den Einsatz verschiedener Medien: „Direct Marketing is an interactive system of marketing which uses one or more advertising media to effect a measurable response and/or transaction at any location".
Eine weitere Definition, die sehr generell gefasst ist, liefert Dallmer (2002, S. 11): „Direct Marketing umfasst alle Marktaktivitäten, die sich einstufiger (direkter) Kommunikation und/oder des Direktvertriebs bzw. des Versandhandels bedienen, um Zielgruppen in individueller Einzelansprache gezielt zu erreichen. Direct Marketing umfasst ferner solche marktgerichteten Aktivitäten, die sich mehrstufiger Kommunikation bedienen, um einen direkten, individuellen Kontakt herzustellen".

Laut Nash (2000, S. 21) ist Direktmarketing durch fünf spezifische Elemente im direkten Kontakt mit dem Kunden gekennzeichnet: Produkt, Angebot bzw. Preis, Medium, Vertriebsmethode und Kreativität.

Im Gegensatz zum generellen Marketing, das vor allem durch anonyme Verbrauchergruppen bzw. grob eingeteilte Marktsegmente gekennzeichnet ist, die überwiegend mit Massenkommunikationsmitteln angesprochen werden, sollen beim Direktmarketing einzelne, individuell bekannte Personen angesprochen und dadurch eine interaktive Beziehung aufgebaut werden (Musiol, 1999, S. 8). Die durch die Massenansprache hervorgerufenen Streuverluste können durch die direkte Kommunikation des Direktmarketings gesenkt werden (Holland, 1992, S. 5). Wenn der Kunde die Möglichkeit einer Rückkopplung erhält, handelt es sich beim Direktmarketing um das sogenannte Dialogmarketing (Kirchner, 1985, S. 180).

Neben der Kommunikationspolitik umfasst das Direktmarketing vor allem auch die Distributionspolitik. Der Begriff des Direktvertriebs ist dabei besonders von Bedeutung, das heißt bei der Distribution von Waren oder Dienstleistungen werden keine Handelsbetriebe zwischen Konsument und Produzent eingeschaltet (Dallmer, 1997, S. 5). Das Angebot zur Inanspruchnahme der Direktbelieferung erfolgt hier beispielsweise über das Mailing oder ein Verlagsverzeichnis. Die Zustellung der Ware wird beispielsweise über die Deutsche Post abgewickelt.

Direktmarketing lässt sich zusammenfassen als Marketing mit den Schwerpunkten auf den Marketinginstrumenten Kommunikations- und Distributionspolitik, da direkt mit dem Kunden kommuniziert und direkt an den Kunden geliefert wird.

In den 50’er Jahren begann die mediale Entwicklung des Direktmarketing mit dem Direktverkauf, später folgte das Direct Mail, dann der Verkauf über das Telefon, anschließend wurden Neue Medien eingesetzt und in den 90’er Jahren setzten sich computerunterstützte Verkaufstechniken durch (Meffert, 2002, S. 41).

Typische Messgrößen im Direktmarketing sind vor allem die Kosten pro Auftrag oder Cost per Order (CPO), und die Rücklauf- bzw. Responsequote (Breitschuh, 2001, S. 76ff.):

$$\text{CPO} = \frac{\text{Werbekosten der Aussendung}}{\text{Anzahl der Aufträge}},$$

$$\text{Rücklaufquote} = \text{Responsequote} = \frac{\text{Anzahl der Reaktionen}}{\text{Anzahl der Aussendungen}}.$$

Response ist ein Fachterminus für die unmittelbare Wirkung einer Marketingaktivität, z.B. die einer Werbeaktion direkt zuordenbare Absatzsteigerung (Diller, 1992, S. 1013).

Im Rahmen der individuellen Ansprache von Konsumenten und der zunehmenden Heterogenität der Zielgruppen ist eine marketingorientierte, leistungsfähige Kundendatenbank für ein erfolgreiches Direktmarketing wichtig (Kreutzer, 1992, S. 326). Die schnelle Entwicklung im Bereich der elektronischen Datenverarbeitung und der daraus resultierenden Möglichkeit, große Datenmengen zu speichern und effizient zu verarbeiten, führten dazu, dass Informationen aus der Kundendatenbank gezielt für Marketingaktionen genutzt werden. **Database Marketing** ist Marketing auf der Basis von Kundendatenbanken (Link/Hildebrand, 1997a, S. 19). „Datenbankgestütztes Marketing“ wurde Mitte der achtziger Jahre in Deutschland eingeführt und wird seitdem von einer stetig steigenden Zahl von Unternehmen eingesetzt (Link/Hildebrand, 1993, S. 29ff.).
Huldi (1992, S. 31) definiert Database Marketing als „... einen Regelkreis, mit dem ermöglicht wird, die bestehenden Daten zu analysieren und danach bestehende oder potenzielle Kunden individuell angepasst und koordiniert hauptsächlich mit Kommunikationsmitteln des Direkt-Marketing anzusprechen, die so erzielten Reaktionen nach der Aktion wieder in die Database einfließen zu lassen, diese Informationen wiederum auszuwerten, damit der ganze Prozess mit einer noch gezielteren, individuelleren Ansprache erneut durchgeführt werden kann und so zu einem langfristigen, interaktiven und individuellen Dialog mit dem Kunden führt“. Als Voraussetzung für Database Marketing müssen für alle Kunden und Interessenten marketingrelevante Informationen in einer Datenbank gespeichert werden. Eine moderne Datenbank umfasst deshalb neben den klassischen Daten wie Name, Adresse und Telefonnummer auch zusätzliche Informationen aus soziodemographischen, psychographischen und kaufverhaltensorientierten Merkmalen (Kotler/Bliemel, 1995, S. 1100). Die zur Verfügung stehenden Daten lassen sich, wie in Abbildung 1 dargestellt, in vier Kategorien einteilen: Grunddaten, Aktionsdaten, Potenzialdaten und Reaktionsdaten.

Grunddaten:
Name, Anschrift, Telefon, Telefax, Email, Bundesland, Branche, Unternehmensgröße (Umsatz, Mitarbeiterzahl), geographische Merkmale, soziodemographische Merkmale, Lifestyle-Merkmale, etc.

Potenzialdaten:
Produktbedarf des Kunden, Zeitpunkt des Bedarfs, Kundentyp, Kundenpotenzialwert, etc.

Aktionsdaten:
Art der Unternehmenskontakte (Mailing, Telefonkontakt, Außendienst), Intensität (Umfang und Dauer der Werbeaktionen), Häufigkeit, Zeitpunkte und Inhalte der erhaltenen Werbeaktionen (Produkt, Preis), etc.

Reaktionsdaten:
Produkte, Höhe des Auftragseingangs, Zeitpunkt des Auftrags, Kundenanfragen, Reklamationen, Retouren, Umsatzhöhe, Deckungsbeitrag, etc.

Abbildung 1: Informationen in einer Kundendatenbank
Quelle: In Anlehnung an Link/Hildebrand (1993, S. 36)

Der Grundgedanke des Database Marketing ist die Individualisierung; mit Hilfe der gespeicherten Informationen besteht beispielsweise die Möglichkeit, den richtigen Kunden zum richtigen Zeitpunkt mit der richtigen Maßnahme der Werbung anzusprechen (Link/Hildebrand, 1993, S. 30). Neben der Individualisierung der Kundenansprache werden beim Databasemarketing auch die Ziele der Segmentierung des unüberschaubaren Kundenbestands in homogenere Teile, die Früherkennung von Absatzpotenzialen und die verbesserte Erfolgskontrolle durchgeführter Marketing-Aktionen verstanden (Schinzer, 1997, S. 107).
Mit Hilfe dieser Informationen bzw. deren Nutzung lassen sich strategische Wettbewerbsvorteile erzielen. Durch die Individualisierung lässt sich eine Kundenbeziehung aufbauen, die anderen Wettbewerbern den Eintritt in die eigenen Kundensegmente wesentlich erschweren soll (Link/Hildebrand, 1993, S. 86). Database Marketing betreibende Unternehmen werden schneller auf Marktgegebenheiten reagieren können, da sie

durch die Auswertung ihrer Kundenreaktionsdaten frühzeitig Chancen und Risiken identifizieren können. Durch eine sukzessive Auswertung der Vielzahl von Daten entsteht im Laufe der Zeit ein Bild vom jeweiligen Kaufverhalten. Dieses Verhaltensprofil gibt dem lernenden Unternehmen die Möglichkeit, die Bedürfnisse der Kunden gezielt anzusprechen. Das Lernen des Kundenverhaltens und die Erkennung von Chancen und Risiken führen weiterhin zu einem höheren Innovationspotenzial. Durch den zielgerichteten Einsatz des gesamten Marketing-Instrumentariums können auch deutliche Kosteneinsparungen verbunden sein.

Ein neuer Schwerpunkt der Unternehmensstrategie ist neben der Produktion und dem Absatz von Produkten die Kundenorientierung, das heißt der Aufbau und das Management von Kundenbeziehungen. Kunden werden zunehmend illoyal und kennen ihre Marktmacht, Märkte werden transparenter, Produktunterschiede geringer, Kommunikationskanäle zwischen Kunde und Unternehmen nehmen stark zu - aus diesen Rahmenbedingungen entwickelte sich das **Customer Relationship Management** (CRM). Das Ziel dabei ist, Kundenbeziehungen langfristig aufzubauen, nachhaltig zu pflegen und profitabel zu gestalten.
Hippner/Wilde (2001b, S. 6) liefern folgende Definition: CRM ist nicht nur ein Konzept, sondern „eine kundenorientierte Unternehmensphilosophie, die mit Hilfe moderner Informations- und Kommunikationsbeziehungen versucht, auf lange Sicht profitable Kundenbeziehungen durch ganzheitliche und differenzierte Marketing-, Vertriebs- und Servicekonzepte aufzubauen und zu festigen“.
Kunden sollen durch eine individuelle Betreuung stärker an das Unternehmen gebunden werden. Häufig werden bei Unternehmen Marketing-, Vertriebs- und Serviceinformationen in unterschiedlichen Systemen gespeichert, die unabhängig voneinander entwickelt und betrieben werden. Deshalb ist häufig keine einheitliche Sicht auf alle vorhandenen Kundendaten möglich. Ziel des CRM-Ansatzes ist die Zusammenführung aller Daten in eine einzige Kundendatenbank, die eine ganzheitliche Sicht auf den Kunden zulässt und auf die alle Unternehmensbereiche zugreifen können (Hippner/Wilde, 2001b, S. 12f.).

Im Allgemeinen werden drei Bereiche des CRM genannt:

- **Kommunikatives** bzw. **kollaboratives CRM**

  Dieser Bereich steht für die Steuerung, Unterstützung und Synchronisation aller Kommunikationskanäle zum Kunden. Ziel ist eine effektive und effiziente Kommunikation zwischen dem Unternehmen und dem Kunden sowie ein einheitliches, abgestimmtes Auftreten des Unternehmens gegenüber dem Kunden (one face to the customer).

- **Operatives CRM**

  Das Ziel dieses Bereichs ist die Unterstützung aller kundenorientierten Geschäftsprozesse, beispielsweise Marketing, Vertrieb und Service.

- **Analytisches CRM**

  Die Basis für das analytische CRM bildet eine Kundendatenbank, in der alle Kundeninformationen vorhanden sind. Mit Hilfe bestimmter Auswertungs- und Analysemethoden soll das Kundenverhalten untersucht und bewertet werden.

Zwischen diesen drei Bereichen findet ein ständiger Informationsaustausch statt (siehe Abbildung 2). Daten aus dem operationalen System fließen in die Kundendatenbank ein. Im operativen CRM werden die Werbeaktionen koordiniert und mit Instrumenten des kommunikativen CRM durchgeführt. Die Kundenreaktionen werden in der Kundendatenbank erfasst und im analytischen CRM mit dem Ziel Kundenwünsche zu antizipieren ausgewertet. Die gewonnenen Erkenntnisse bilden wiederum die Grundlage für neue Maßnahmen im operativen bzw. kommunikativen CRM. Das stetige Lernen aus der Kundenbeziehung führt dazu, dass der Kunde immer individueller beworben wird und dadurch im Idealfall eine lebenslange Beziehung zum Unternehmen aufbaut (Rapp, 2002, S. 83f.). CRM kann somit als lernendes System (Closed Loop Architecture) verstanden werden.

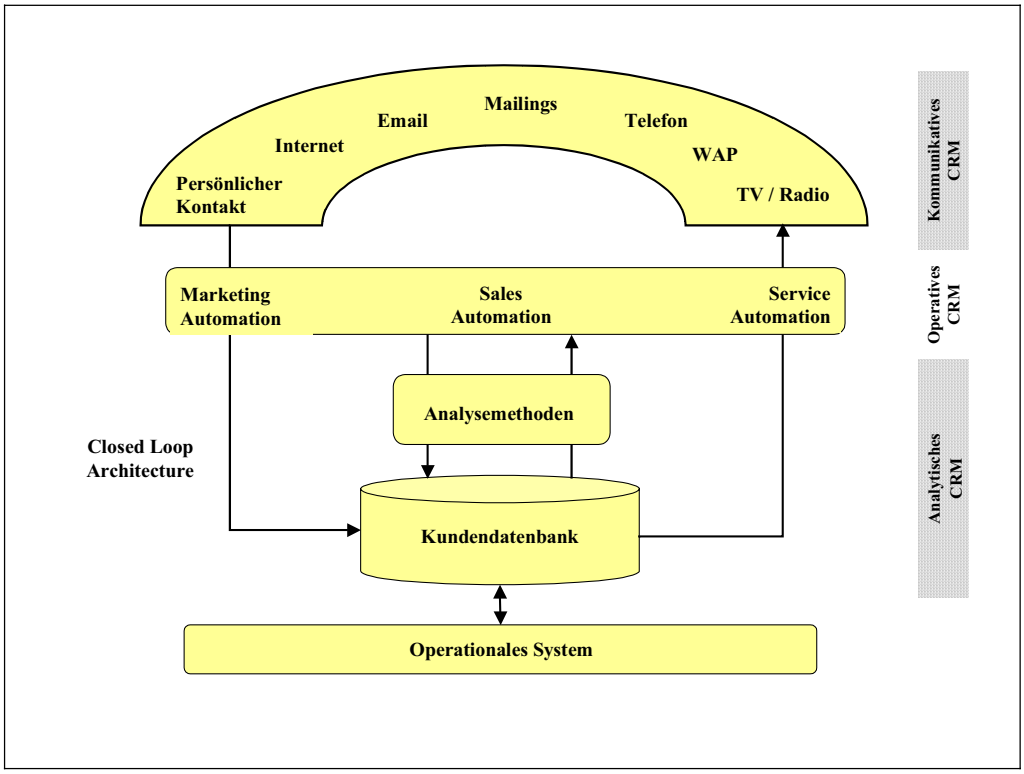


Abbildung 2: CRM Closed Loop Architecture
Quelle: In Anlehnung an Hippner/ Wilde (2001a, S. 14).

Im Direktmarketing fallen durch die unmittelbare Kommunikation mit dem Kunden sehr viele Kontaktinformationen an, die in Datenbanken gespeichert werden (siehe Abbildung 1, S. 5). Diese Informationen sollen nun gewinnbringend bei Direktmarketingaktionen eingesetzt werden. Sind die wesentlichen Komponenten einer Direktmarketingaktion festgelegt, stellt sich die Frage, welche Adressen dabei mit einbezogen werden sollen (Cohen, 1985, S. 254ff.). Bezogen auf die Datenbank bedeutet dies, geeignete Selektionskriterien festzulegen (Bausch, 1991, S. 86). Eine Untersuchung von American Express ergab, dass 50% der Response eines Mailing auf die richtige Auswahl der Zielgruppe zurückzuführen sind, je 20% entfallen auf das richtige Timing bzw. die Angebotsform und die restlichen 10% auf die Gestaltung des Werbemittels (Mayer/Middecke, 1997, S. 355). Kunden sollen also bewertet werden, damit für eine bestimmte Marketingaktion die erfolgversprechendsten Adressen ausgewählt werden können. Der Begriff **Responseoptimierung** bedeutet allgemein, die Responsequote bei einer bestimmten Werbeaktion zu erhöhen. Dieser Begriff wird im folgenden speziell auf die Optimierung der Adress-Selektion bezogen, das heißt das Timing, die Angebotsform und das Werbemittel stehen fest. Die Erhöhung der Responsequote soll erreicht werden, indem nur die attraktivsten Kunden beworben werden. Das Ziel ist somit die Identifikation der attraktivsten Kunden bzw. der Besteller anhand aller vorliegender

Informationen in der Datenbank. Häufig werden dafür synonym die Begriffe Response-Analyse (Musiol/Steinkamp, 2001, S. 742), Vorhersage von Responsewahrscheinlichkeiten zum Optimieren einer Direktmarketingaktion (Ittner/Sieber/Trautzsch, 2001) oder Zielgruppendefinition mit Hilfe eines Responsemodells zur Erhöhung der Responsequote (Poloni/Belke, 2001) verwendet. Meist ist das Budget für den Versand vorgegeben und es sollen die am besten für die jeweilige Bewerbung geeigneten Kunden selektiert werden. Dies kann **subjektiv**, beispielsweise durch einen Experten, oder **objektiv**, beispielsweise methodengestützt durch eine Punktbewertung, geschehen. **Punktbewertungsverfahren** oder **Scoring-Modelle** werden verwendet, um anhand aller verfügbaren Informationen (siehe Abbildung 1, S. 5) auf das künftige Bestellverhalten zu schließen. Dabei wird jedem Kunden ein bestimmter **Scorewert** zugeordnet, der die Attraktivitätseinstufung widerspiegelt, so dass die besten Kunden ausgewählt werden können. Opitz (1978, S. 7) spricht in diesem Zusammenhang von **Identifikation**, das heißt ausgehend von einer Objektmenge, für die eine klassifikatorische Struktur vorgegeben ist, wird versucht, aus einem relevanten Merkmalskatalog die für die Klassen wesentlichen Merkmale und deren Gewichtung aufzudecken. Anhand der Informationen, die zum Zeitpunkt der Bewerbung vorliegen, soll für neue Objekte auf deren Zugehörigkeit zu einer bestimmten vordefinierten Klasse, beispielsweise attraktiv bzw. nicht attraktiv, geschlossen werden.

1.2 Problemstellung bei WEKA MEDIA

Die WEKA MEDIA GmbH & Co. KG (im folgenden kurz: WEKA MEDIA) ist einer der führenden Verlage Deutschlands für Fachinformationen im Business-to-Business-Bereich. Anfang 2001 ist WEKA MEDIA aus dem Zusammenschluss der WEKA Baufachverlage, der WEKA Management-Verlage und des WEKA Fachverlags für Technische Führungskräfte hervorgegangen. Der Fachverlag ist ein Unternehmen der in fünf europäischen Ländern operierenden WEKA Business Information GmbH & Co. KG.
WEKA MEDIA bietet qualitativ hochwertige praxisbezogene Produkte und Services an und nutzt dafür ihre hohe Medienkompetenz: Das Spektrum reicht von Print-, Software- und Onlineprodukten über Videos bis hin zu Seminaren und Kongressen. Das Produktportfolio wendet sich an Fach- und Führungskräfte aus den Bereichen Technik, Sicherheit, Gesundheit und Umwelt, Management und Behörden, Bauhandwerk, Architektur und Immobilienwirtschaft sowie Informationstechnologie.

Die Abteilung „Customer Research“ beschäftigt sich vor allem mit Kundenanalysen, der Entdeckung von Trends und Aufdeckung von Informationen, die für die Unternehmenssteuerung wichtig sind. Dabei werden wichtige Erkenntnisse gewonnen, die Einfluss auf die Unternehmensstrategie haben. Durch das frühzeitige Entdecken bestimmter Trends können rechtzeitig Maßnahmen ergriffen werden, um sich auf veränderte Marktbedingungen einzustellen. Das Ziel der Abteilung liegt hauptsächlich in der Kosteneinsparung beim Vertriebsbudget durch Effizienzsteigerung. Dies wird auch durch eine Reduzierung der Streuverluste bei Direktmarketingaktionen erreicht. Ein Aufgabengebiet der Kundenanalyse bei WEKA MEDIA ist damit die Responseoptimierung für einzelne Werbeaktionen. Dabei ist das Werbebudget meist vorgegeben. Es sollen dann die erfolgversprechendsten Kunden, das heißt Ansprechpartner eines Unternehmens, für diese Werbeaktion selektiert werden, um die Responsequote zu erhöhen.

Die Bewerbung der Produkte wird vor allem über Direktmarketingmaßnahmen abgewickelt. Dabei nutzt WEKA MEDIA hauptsächlich die Vertriebswege Direct Mail, Telephonmarketing, Internet, Vertretervertrieb, Key Account Management und Anzeigen/Beilagen.

WEKA MEDIA hat ca. 5 Millionen Adressen in ihrer Datenbank, wobei zu jeder Adresse viele weitere Informationen und die Bewerbungs- bzw. Transaktionshistorie vorliegen. Bei Direct Mail Aktionen liegt die Responsequote zum Teil deutlich unter 1%. Es gibt verschiedene Vorgehensweisen, wenn eine Klasse nur sehr selten vorkommt. Bei einem Bestelleranteil von 1% erzielt ein Verfahren, das alle Objekte als Nichtbesteller einstuft, 99% Genauigkeit. Jedoch sollen geeignete Adressen ausgewählt und Produkte verkauft werden. Dieses Problem muss in der Praxis im Rahmen der Responseoptimierung gelöst werden. Die Auswirkung verschiedener Vorgehensweisen auf die Ergebnisse unterschiedlicher Algorithmen ist bisher noch nicht umfassend untersucht worden.

1.3 Zielrichtung und Aufbau der Arbeit

In dieser Arbeit werden verschiedene Vorgehensweisen und Methoden zur Responseoptimierung (siehe S. 8f.) verglichen. Dabei wird folgende Fragestellung untersucht: Welche Ausprägungen der unabhängigen Variablen $x_1,...,x_p$ aus Abbildung 1 (siehe S. 5) führen im Verbund zu y=0 bzw. y=1. Dabei bedeutet y=0, dass keine Bestellung erfolgt ist und y=1, dass eine Bestellung erfolgt ist.
Der allgemeine funktionale Zusammenhang lässt sich als $y=f(x_1,...,x_p)$ formulieren. Einerseits wird der Typ des Zusammenhangs von f gesucht, andererseits müssen die Modellparameter von f geschätzt werden. Dies entspricht einem Klassifikationsmodell mit unabhängigen Variablen, die metrisch, ordinal oder nominal skaliert sein können und einer abhängigen Variable, die nominal binär skaliert ist. Die Besteller, also y=1, werden im folgenden als 1-Klasse bezeichnet, die Nichtbesteller als 0-Klasse.

Die Zuordnung neuer Kunden, also die Prognosefunktion des Modells, und die Planung und Steuerung der Adressauswahl können als Folgeaufgaben angesehen werden. Eine offensive Planung bedeutet dabei, dass nur sehr gute Adressen bezogen auf die Attraktivität ausgewählt werden, während bei einer defensiven Planung nur sehr schlechte Adressen eliminiert werden sollen.

Aufgrund des Datenumfangs $n >> 10.000$ und der niedrigen Responsequote $y=1 << 1\%$ scheiden klassische Vorgehensweisen zur Analyse aus. Es empfiehlt sich, verschiedene Methoden parallel anzuwenden und die Ergebnisse zu vergleichen. Insbesondere werden in dieser Studie verschiedene Möglichkeiten zur Bewältigung niedriger Responsequoten untersucht und verglichen. Das Ziel ist dabei jeweils, $y \subset \{0;1\}$ mit Hilfe der unabhängigen Variablen $x_1,...,x_p$ mit minimalem Fehler zu bestimmen. Dies kann bei diesen speziellen Rahmenbedingungen zu deutlichen Unterschieden führen, die im Hauptteil dieser Studie vorgestellt und diskutiert werden.

Nach dieser Einführung in die Thematik erfolgt im **zweiten Kapitel** die Darstellung der **informationstechnischen und methodischen Grundlagen**. Zuerst werden die Begriffe Datawarehouse und Knowledge Discovery in Databases definiert, bevor verschiedene traditionelle Kundenbewertungsverfahren sowie die Begriffe Data Mining und OLAP erläutert werden. Das Kapitel schließt mit einer Zusammenfassung und zeigt, wie die

beiden Bereiche zusammenhängen. Nachdem die theoretischen und methodischen Grundlagen aufbereitet sind, werden im folgenden Teil der Arbeit die Datenbasis vorgestellt, ausgewählte Verfahren ausführlich erklärt und im Rahmen einer empirischen Untersuchung angewandt und verglichen.

Im **dritten Kapitel** wird zu Beginn eine Matrixform zur strukturierten Darstellung der Datenbasis gewählt und die **Datenvorverarbeitung** durchgeführt. Zur **Bewältigung der Problematik niedriger Responsequoten** werden Methoden der Stichprobenplanung und Verfahren der Clusteranalyse zur Unterstützung der Stichprobenziehung genutzt. Zum Abschluss dieses Kapitels werden der Versuchsaufbau vorgestellt und die bisherigen Ergebnisse zusammengefasst.

Die drei nachfolgenden Kapitel befassen sich mit den angewandten Algorithmen und den empirischen Ergebnissen.

Das **vierte Kapitel** zeigt die erste Gruppe von Verfahren zur Responseoptimierung: **Entscheidungsbäume**. Dabei werden zuerst verschiedene Entscheidungsbaumvarianten, Partitionierungskriterien und ausgewählte Pruningtechniken vorgestellt. Je nach Kombination dieser Bestandteile entstehen drei bekannte Entscheidungsbaumverfahren. Danach folgen die empirischen Ergebnisse mit Bewertung und eine Zusammenfassung.

Im **fünften Kapitel** wird die Responseoptimierung mit der **binären logistischen Regression** durchgeführt. Zuerst werden das Logitmodell, die Parameterschätzung und die Tests im Rahmen einer logistischen Regression erklärt. Danach folgen die empirischen Ergebnisse. Diese werden, auch im Vergleich zu den Entscheidungsbäumen, bewertet und zusammengefasst.

Das **sechste Kapitel** stellt die Responseoptimierung mit **Künstlichen Neuronalen Netzen** vor. Dabei werden sowohl Varianten und Struktur Neuronaler Netze als auch verschiedene Gütekriterien erklärt. Im Anschluss folgen die empirischen Ergebnisse sowie deren Bewertung und Zusammenfassung, auch im Vergleich zu den beiden vorher angewandten Verfahren.

Zum **Vergleich der Modelle** anhand der verwendeten Variablen wird im **siebten Kapitel** eine Mehrdimensionale Skalierung durchgeführt. Zuerst wird dieses Verfahren beschrieben, dann werden die empirischen Ergebnisse gezeigt. Weiterhin werden wichtige Variablen identifiziert und Auswirkungen einer Komplexitätsreduktion untersucht. Darauf folgen die Zusammenfassung und die Bewertung der Ergebnisse.

Die Arbeit schließt mit einer **Zusammenfassung aller Ergebnisse** und gibt Anregungen für weitere Forschungsvorhaben.

Abbildung 3 zeigt den Aufbau der Arbeit in der Übersicht:

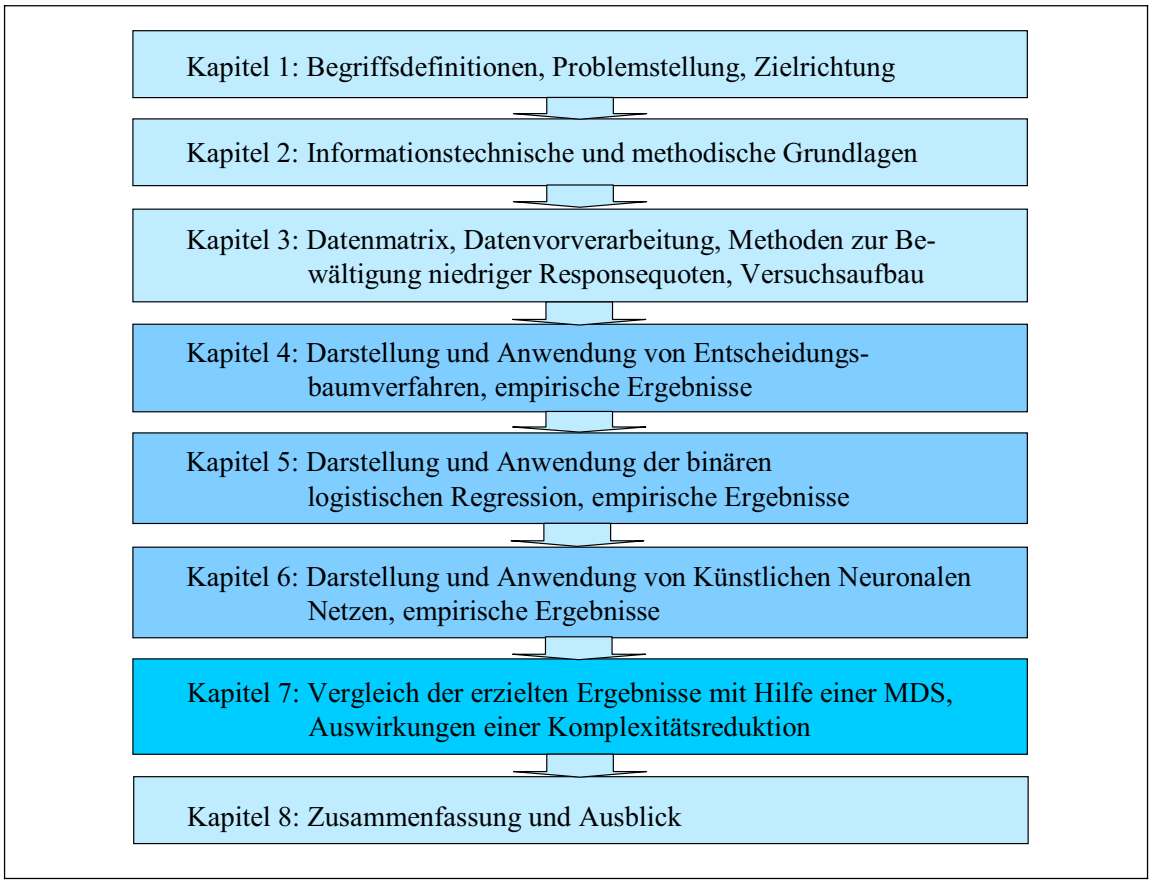


Abbildung 3: Aufbau der Arbeit
Quelle: Eigene Darstellung

2. Informationstechnische und methodische Grundlagen

Nachdem im ersten Kapitel die Aufgabenstellung thematisiert und abgegrenzt wurde, erfolgt nun eine Einführung in die informationstechnischen und methodischen Grundlagen. Dabei werden das Datawarehouse-Konzept, der Begriff Knowledge Discovery in Databases, verschiedene traditionelle Kundenbewertungsverfahren, sowie die Methoden Data Mining und OLAP beschrieben. Der Zusammenhang der beiden Bereiche bildet den Abschluss dieses Kapitels.

2.1 Informationstechnische Grundlagen

2.1.1 Das Datawarehouse Konzept

In der heutigen Zeit entsteht ein Wettbewerbsvorteil vor allem durch die effektive Nutzung von Informationen, die in speziellen Systemen gesammelt werden. Da die operationalen DV-Systeme für eine Unterstützung dieser Art nicht konzipiert sind, ist es wichtig, diese Informationen zu erschließen (Anahory/Murray, 1997, S. 19). Die operationalen Systeme sind für einen reibungslosen Ablauf des täglichen Geschäfts zuständig, das heißt die Rechnungsstellung, Einkauf, etc. Mit dem Begriff **Datawarehouse** (DWH) wird eine von den operationalen DV-Systemen isolierte Datenbank umschrieben, die als unternehmensweite Datenbasis für Managementunterstützungssysteme dient (Muksch/Holthuis/Reiser, 1996, S. 421). Ein Datawarehouse ist eine themenorientierte, integrierte, nicht-volatile und zeitraumbezogene Sammlung von Daten zur Entscheidungsunterstützung des Managements (Inmon, 1996, S. 33).

Im Gegensatz zur Funktions- und Anwendungsorientierung der operationalen Systeme, beispielsweise Einkauf oder Logistik, steht bei der Konzeption des DWH die Orientierung an Gegenstandsbereichen des Unternehmens, beispielsweise Geschäftsbereiche oder Kundengruppen, im Vordergrund. Anstatt innerbetrieblicher Funktionen und Prozesse sind für den Anwender interessante und relevante Daten bedeutsam.

Das DWH strebt eine unternehmensweite Integration von Daten in einem einheitlich gestalteten System an. Dabei wird eine Struktur- und Formatvereinheitlichung vorgenommen: Werden für eine Merkmalsausprägung in unterschiedlichen operationalen Datenbanken verschiedene Bezeichnungen verwendet, beispielsweise in englischer und deutscher Sprache, so werden diese bei der Übertragung in das DWH vereinheitlicht.

Um die Nicht-Volatilität der Daten im DWH zu gewährleisten, erfolgen Zugriffe auf das DWH nur lesend.
Im Gegensatz zu den operationalen System, welche die Daten zeitpunktbezogen betrachten, ist das DWH zeitraumbezogen, da die Entwicklung des Unternehmens über einen bestimmten Zeitraum abgebildet werden soll. Ein Zeithorizont von fünf bis zehn Jahren erscheint sinnvoll, um auch Trendanalysen über historische Daten zu ermöglichen (Inmon, 1996, S. 36).

Es gibt viele weitere Definitionen, allen gemeinsam ist jedoch die Forderung nach einer nutzengerechten Aufbereitung entscheidungsrelevanter Informationen (Krahl/Windheuser/Zick, 1998, S. 50).

Die Aufgabe des DWH ist die Integration von Daten aus unterschiedlichen Quellen, um das operationale DV-System durch zusätzliche Auswertungen nicht zu überlasten. Es sollen alle relevanten Daten in einer Datenbank vorhanden sein und so aufbereitet werden können, dass sie dem Management entscheidungsrelevante Informationen liefern.

Der Weg der Daten von den operationalen Datenbanksystemen und die Einbindung externer Daten in das DWH wird in Abbildung 4 gezeigt.

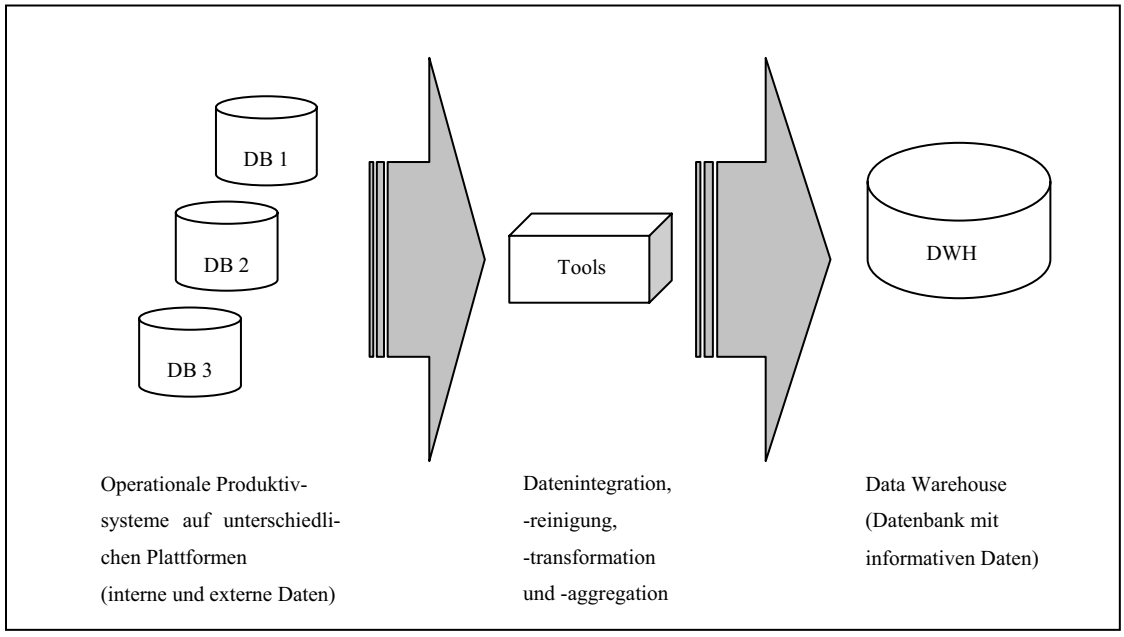


Abbildung 4: Datawarehouse – Allgemeine Struktur
Quelle: Eigene Darstellung

Die Datenbasis bilden interne und externe Daten. Die internen Daten kommen aus unternehmenseigenen Datenquellen, beispielsweise Daten aus unterschiedlichen Anwen-

dungssystemen mit eigenen Datenbanken oder aus geographisch verteilten Organisationen des Unternehmens, wie Kundendaten, Produktdaten oder Transaktionsdaten.
Unter externen Daten werden Daten verstanden, die z.B. von Markt- oder Meinungsforschungsinstituten zugekauft werden können, beispielsweise demographische Daten, psychometrische Daten oder Wirtschaftsdaten.
Diese Daten werden aus den entsprechenden Datenquellen extrahiert, transformiert und für die Nutzung im DWH aufbereitet. Bei diesem Schritt werden die Struktur und das Format der Daten vereinheitlicht und fehlerhafte Datensätze ausgeschlossen. Das Ergebnis ist eine für Analysezwecke geschaffene, einheitliche und unternehmensweite Sammlung von Daten.

Während das DWH eine unternehmensweite Datenbasis für Analysen bietet, sind **Data-Marts** Ausschnitte aus dem DWH. Unter einem Data Mart versteht man eine logische themenspezifische oder abteilungsspezifische Teilmenge des DWH. Es sind beispielsweise nur die für eine bestimmte Abteilung interessanten Daten enthalten.

Der Marketing Data-Mart von WEKA MEDIA besteht beispielsweise aus fünf Tabellen, die in Tabelle 1 beschrieben werden.

Tabelle	Beschreibung
Person	Alle Informationen zu dem Ansprechpartner bzw. der Firma (alle Daten seit ca. 1980).
Position	Alle Informationen zu den getätigten Umsätzen je Kunde (alle Daten seit ca. 1995).
Serial_Order	Alle Informationen zu den Abonnements (alle Daten seit ca. 1986).
Campaign	Alle Informationen zu dem Inhalt und dem Zeitpunkt der Werbeaktionen (alle Daten seit ca. 1995).
Result	Alle Informationen darüber, welche Kunden für welche Werbeaktion bisher selektiert wurden (alle Daten seit ca. 1995).

Tabelle 1: Tabellen im WEKA MEDIA Marketing Data-Mart

Mit der Existenz umfangreicher Datenmengen ist der Bedarf an effizienter Datenanalyse gestiegen. Mit Hilfe von Computertechnologie sollte aus großen Datenmengen schnell Wissen gewonnen werden, jedoch veranschaulicht folgendes Zitat die Problematik da-

bei: “Computers have promised us a fountain of wisdom but delivered a flood of data. – A frustrated MIS Executive” (Frawley/Piatetsky-Shapiro/Matheus, 1991, S. 1).
Aus dieser Problematik heraus entwickelte sich eine Disziplin, die sich mit der Analyse großer Datenmengen beschäftigt.

2.1.2 Knowledge Discovery in Databases

Der Prozess der Wissensgewinnung/-entdeckung aus Datenbanken wird **Knowledge Discovery in Databases** genannt, der nicht nur auf eine oder mehrere Phasen der Wissensentdeckung abzielt, sondern den gesamten Entdeckungsprozess umfasst (Düsing, 1999, S. 347). „Knowledge Discovery in Databases (KDD) is the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data“ (Frawley/Piatetsky-Shapiro/Matheus, 1991, S. 3). Das Ziel ist, das in den Datenbanken vorhandene Wissen zu entdecken und zugänglich zu machen. Dazu werden im Rahmen des KDD-Prozesses weitgehend autonom Beziehungsmuster in den Daten, z.B. Zusammenhänge oder Abhängigkeiten, ermittelt. Diese Muster müssen für einen möglichst großen Teil des Datenbestandes gültig sein und möglichst bislang unbekannte, potenziell nützliche und leicht verständliche Zusammenhänge in den Daten zum Ausdruck bringen. Aus den ermittelten Beziehungsmustern wird schließlich explizites Wissen abgeleitet und zur Unterstützung betriebswirtschaftlicher Entscheidungen herangezogen.

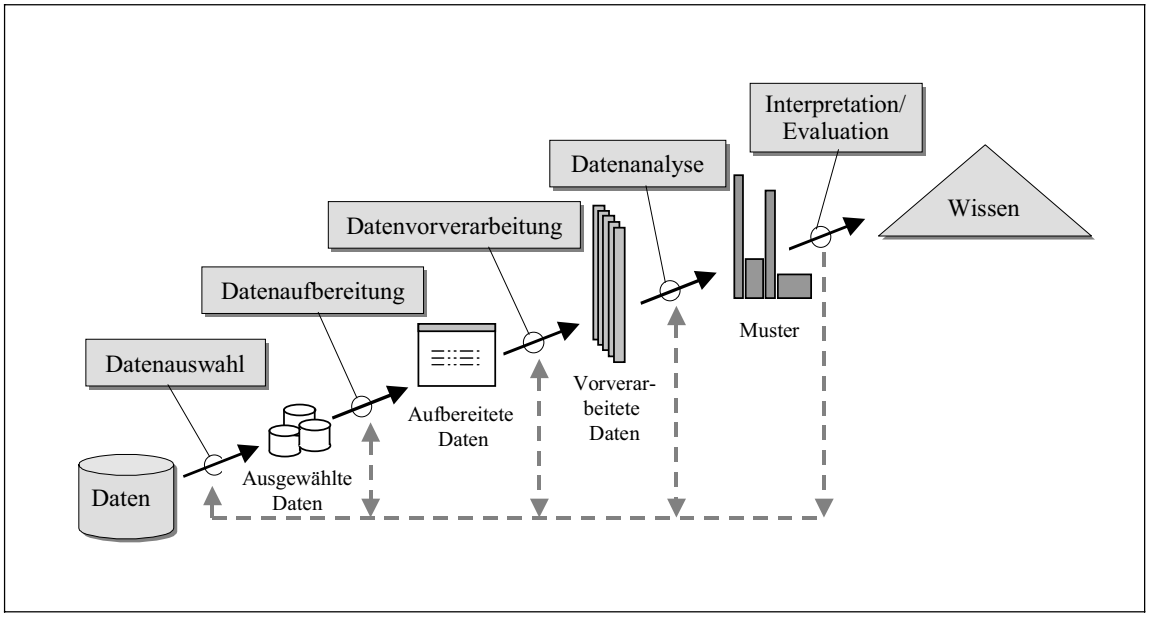


Abbildung 5: Der KDD-Prozess
Quelle: In Anlehnung an Fayyad/Piatetsky-Shapiro/Smyth (1996, S. 10)

In Abbildung 5 beginnt der KDD-Prozess mit der **Auswahl der relevanten Daten**. Dabei wird im Idealfall auf Daten aus dem DWH zugegriffen. Im nächsten Schritt, der **Datenaufbereitung**, wird die Datenmatrix für die Analyse aufbereitet, das heißt es wird aus den unterschiedlichen Rohtabellen in der Datenbank eine **Datenmatrix**, die oft auch als **flat file** bezeichnet wird, erstellt. Dabei werden häufig zusätzliche Variablen berechnet, z.B. die Retourenquote oder der Gesamtumsatz eines Kunden. Darauf folgt die **Datenvorverarbeitung**, beispielsweise werden hier Variablen umkodiert bzw. transformiert, Ausreißer eliminiert und fehlende Werte ersetzt. Nachdem die Voranalysen beendet sind, findet die **Datenanalyse** statt, das heißt es werden verschiedene mathematische und statistische Verfahren auf die Daten mit dem Ziel angewandt, Beziehungsmuster in den Daten aufzudecken. Die Erkenntnisse aus der Analyse werden im letzten Schritt interpretiert und bewertet. Das Ziel des gesamten Prozesses ist die **Gewinnung von Wissen**. Dies kann beispielsweise die Erkenntnis sein, welche Variablen bei der Attraktivitätseinstufung eines Kunden benutzt werden bzw. welche Kunden für eine Werbeaktion geeignet sind. Wichtig dabei ist, dass es nach jedem Schritt möglich ist, zu einem der vorherigen Schritte zurückzugehen, da möglicherweise neue Erkenntnisse vorliegen, die eine Änderung der bisher getätigten Einstellungen nötig machen.

Die Firma SAS™ Institute hat mit der "**SEMMA-Methodology**" einen eigenen Prozess für die Durchführung des KDD entwickelt (siehe Abbildung 6). SEMMA steht dabei für die einzelnen Schritte **S**ample, **E**xplore, **M**odify, **M**odel und **A**ssessment. Diese Vorgehensweise beginnt bei den aufbereiteten Daten (siehe Abbildung 5, S. 18).

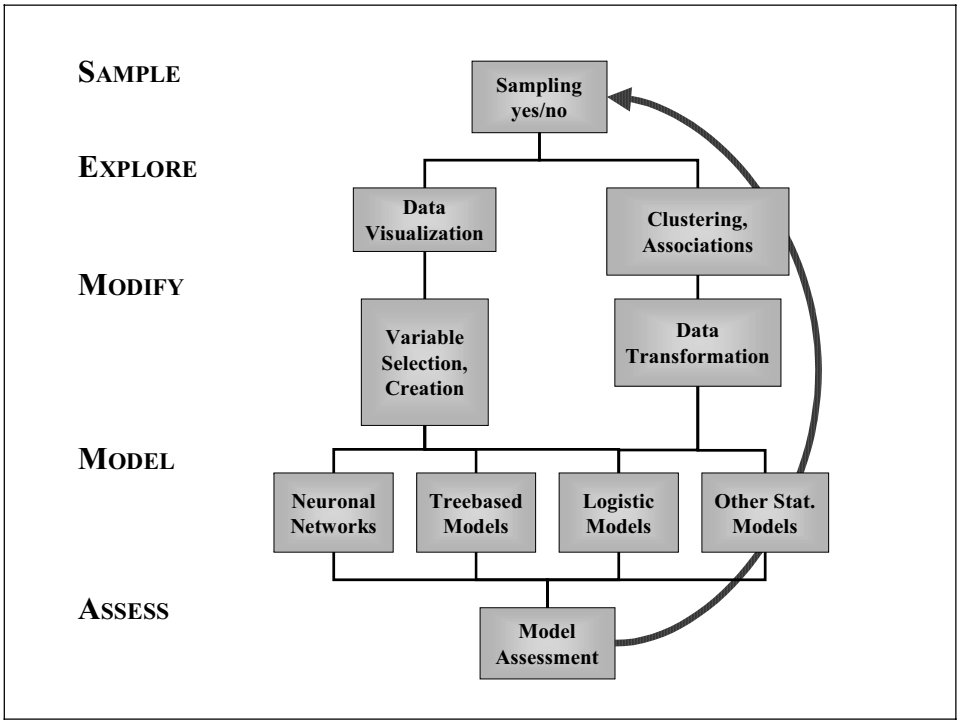


Abbildung 6: SEMMA-Methode
Quelle im Internet: SAS Institute GmbH (2002)

In der in dieser Studie verwendeten Data-Mining-Software SAS™ Enterprise Miner wird dem Analysten zu jedem der einzelnen Schritte eine Vielzahl an Methoden angeboten. Grundsätzlich sieht der Prozess vor, zuerst eine Stichprobe zu ziehen, diese deskriptiv und explorativ auszuwerten, um ein Verständnis für die Daten zu entwickeln und sie schließlich für die Analyse vorzubereiten, indem Ausreißer eliminiert, fehlende Werte ersetzt oder Variablen neu gebildet bzw. umkodiert werden. Auf dieser analysegerechten Datenmatrix werden dann die relevanten Verfahren angewandt und im Anschluss daran die Ergebnisse verglichen, um das beste Modell auszuwählen. Häufig geben diese Ergebnisse neuen Input oder zeigen Fehler in dem Datenbestand auf, so dass wieder zum ersten Schritt zurückgegangen werden muss.

Eine Alternative ist **CRISP-DM**, der **Cr**oss **I**ndustrie **S**tandard **P**rocess of **D**ata **M**ining (Chapman et al., 1998). Er ist aus einer Zusammenarbeit von NCR, DaimlerChrysler, OHRA und SPSS entstanden (siehe Abbildung 7).

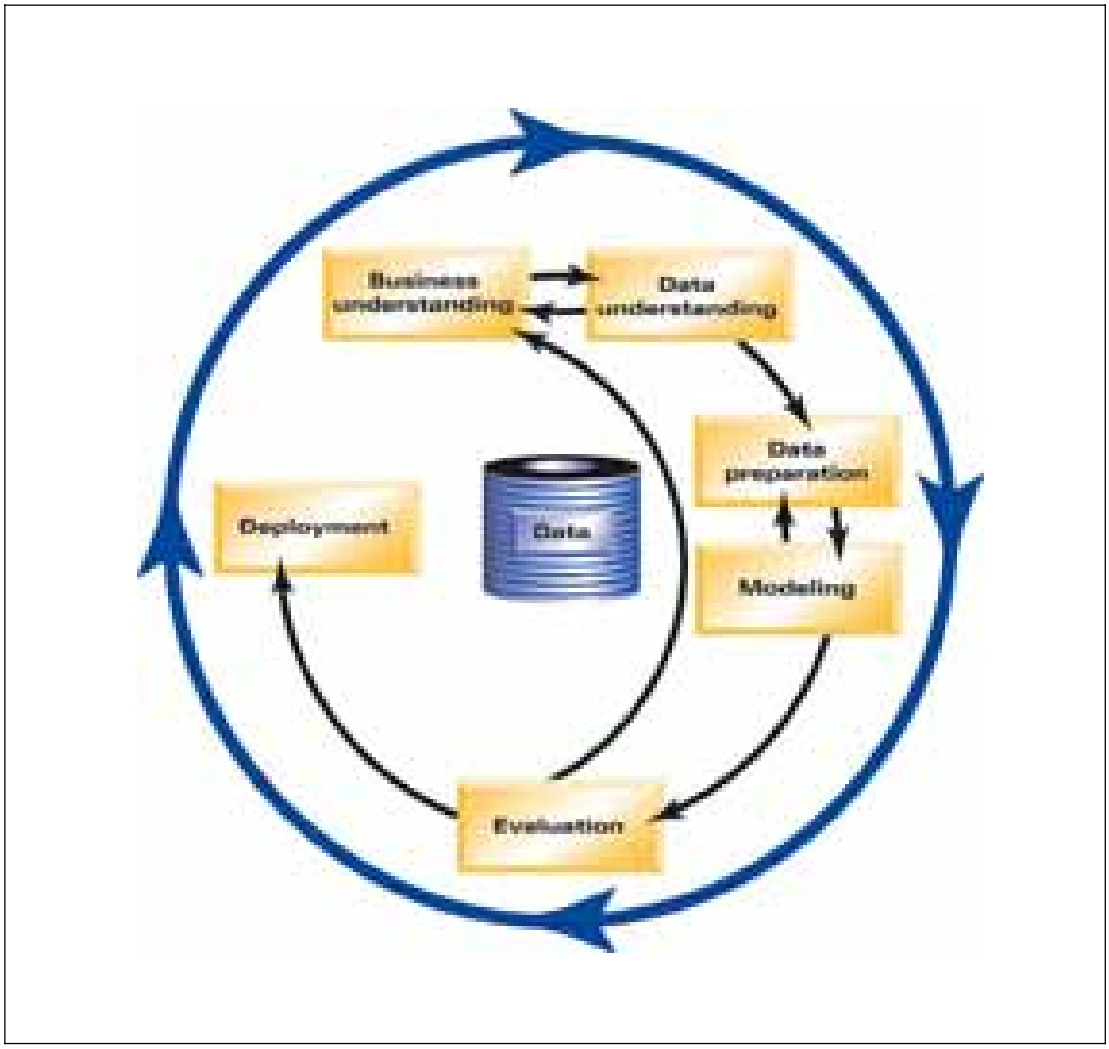


Abbildung 7: CRISP
Quelle im Internet: CRISP-DM (2003)

Hier werden im Vergleich zur SEMMA-Methode zusätzlich die Phasen Business Understanding, Data Understanding, Evaluation und Deployment eingeführt. Im Vergleich zum KDD-Prozess (siehe Abbildung 5, S. 18) wird dabei die Phase Business Understanding zusätzlich verwendet. Die Beschreibung der einzelnen Phasen orientiert sich an Chapman et al. (1998). **Business Understanding**, als erste Phase, fokussiert auf das Projektziel und dessen Übersetzung in eine analysegeeignete Fragestellung. Weiterhin wird hier der Projektplan ausgearbeitet, in welchem die nächsten Schritte und die anzuwendende Vorgehensweise zur Zielerreichung festgelegt werden. **Data Understanding** ist ein wesentlicher Punkt, denn hier werden die Daten gesammelt, auf Plausibilität geprüft und erste deskriptive Analysen durchgeführt, um mit den Daten vertraut zu werden. In der **Data Preparation-Phase** wird die endgültige Analysedatenmatrix gebildet. Beim **Modeling** werden die relevanten Verfahren ausgewählt und angewandt. Eine Überprüfung der Ergebnisse aus der Analyse, das heißt, ob sie beispiels-

weise ihrer Zielsetzung genügen, findet in der **Evaluationsphase** statt. In der **Deployment**phase wird das ausgewählte Modell in die bestehenden Prozesse eingegliedert, um die Umsetzung der Ergebnisse zu gewährleisten. Auch hier sind an verschiedenen Stellen Möglichkeiten vorgesehen, im Prozess nochmals einen oder mehrere Schritte zurückzugehen.

Abbildung 8 zeigt in Anlehnung an Abbildung 5 (siehe S. 18) die Verteilung des zeitlichen Aufwands der einzelnen Phasen des KDD-Prozesses. Die Datenauswahl, -aufbereitung und -vorverarbeitung nimmt dabei ca. 60% der Gesamtzeit in Anspruch (Cabena et al., 1998, S. 43). Als Basis für alle weiteren Schritte, stellt die Datenaufbereitung sowohl den aufwändigsten als auch den wichtigsten Schritt im gesamten KDD-Prozess dar. Für die Aufgabendefinition werden ca. 20% der Gesamtzeit benötigt, die Datenanalyse und die Interpretation der Ergebnisse werden mit jeweils ca. 10% angegeben. Zu beachten ist, dass die Konsolidierung der Ergebnisse, das heißt wenn Fehler entdeckt werden und beispielsweise nochmals bei der Datenauswahl begonnen werden muss, in dieser Darstellung nicht berücksichtigt ist.

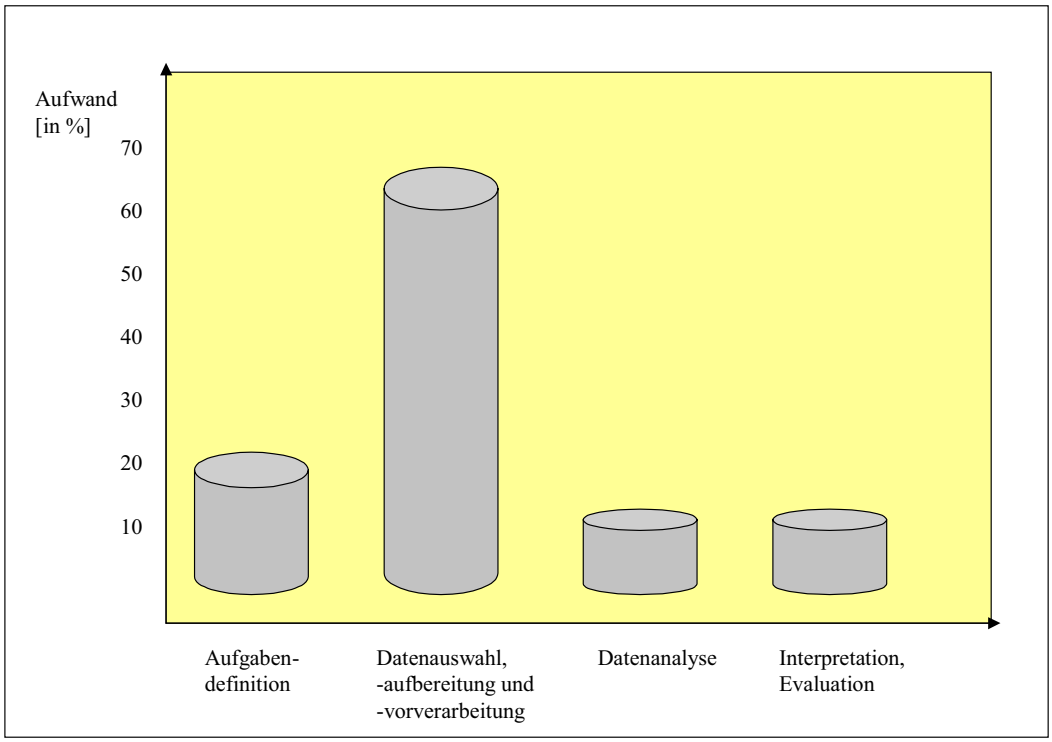


Abbildung 8: Zeitverteilung im KDD-Prozess
Quelle: In Anlehnung an Cabena et al. (1998, S. 43)

2.2 Methodische Grundlagen

Im folgenden werden die methodischen Grundlagen vorgestellt. Dazu zählen traditionelle Kundenbewertungsverfahren, Data Mining und das OLAP-Konzept.

2.2.3 Traditionelle Kundenbewertungsverfahren

Bei großen Kundendatenbeständen ist es sinnvoll, die Kunden mittels bestimmter Kriterien zu bewerten. Denn nicht für alle bestehenden und potenziellen Kunden ist die Individualisierung eine sinnvolle Marketingstrategie (Dallmer, 2002, S. 15). Die Kundentypisierung und nachfolgende Segmentierung bilden die Basis für einen differenzierten Kundendialog (Rapp, 2002, S. 83). Laut Link/Hildebrand (1993, S. 56ff.) gibt es vier wesentliche Ansätze der Kundenbewertung:

- das Loyalitätsleiter-Konzept,
- die monetären Bewertungsverfahren,
- die Punktbewertungsverfahren und
- die Clusteranalyse.

Im folgenden werden die einzelnen Ansätze kurz vorgestellt. Auf die Punktbewertungsverfahren wird ausführlicher eingegangen, da das Prinzip dieses Ansatzes in dieser Arbeit verwendet wird.

2.2.3.1 Das Loyalitätsleiter-Konzept

Die aktuellen und potenziellen Kunden werden bei diesem Ansatz nach der erreichten Stufe ihrer Bindung zum Unternehmen unterschieden und segmentiert (Kreutzer, 1991, S. 632f.). Die Bindung wird dabei aufgrund ihrer bisher getätigten Transaktionen mit dem Unternehmen bestimmt (siehe Abbildung 9).

Abbildung 9: Loyalitätsleiter
Quelle: In Anlehnung an Kreutzer (1991, S. 633)

Die vorhandenen Kunden sollen auf die jeweils nächsthöhere Stufe gebracht werden, da der Nutzen für das Unternehmen bei „Erhöhung der Stufenverteilung“ steigt. Jede Ebene stellt unterschiedliche Anforderungen an Inhalt, Art und Intensität der Kundenansprache. Während den potenziellen Kunden erst das Unternehmen als Kompetenzträger und Partner nähergebracht werden muss, kann bei einem Mehrfachkäufer verstärkt auf den Nutzen der Produkte eingegangen werden, da er aufgrund seiner Mehrfachkäufe bereits mit dem Unternehmen vertraut ist. Der Stammkunde zeichnet sich durch hohe Loyalität zum Unternehmen aus. Weiterhin ist zu erwähnen, dass die sogenannten „Datenbankleichen“, also Adressen, die in der Datenbank vorhanden sind, jedoch nicht mehr genutzt werden, entweder zu der Gruppe der potenziellen Kunden mit Kenntnis über das Unternehmen/Produkte gezählt oder aus der Betrachtung ausgeklammert werden können.

2.2.3.2 Monetäre Bewertungsverfahren

Die monetären Bewertungsverfahren beinhalten nach Link/Hildebrand (1997b, S. 162ff.) die Kundenumsatzanalyse, die Kundendeckungsbeitragsrechnung, das Kundendeckungsbeitragspotenzial und den Lebenszeitwert eines Kunden oder Customer Lifetime Value.

Die **Kundenumsatzanalyse** entspricht der ABC-Analyse und ermöglicht eine differenzierte Betrachtung der Kunden. Diese werden dabei anhand der getätigten Umsätze in

einer Periode bewertet. Abbildung 10 zeigt zur Veranschaulichung eine Grafik, wobei auf der Abszisse der kumulierte Anteil der Kunden angetragen wird und auf der Ordinate der kumulierte Umsatzanteil.

Wenn gilt:

	Umsatzanteil	Kundenanteil
A-Kunden	u_1	k_1
B-Kunden	u_2	k_2
C-Kunden	u_3	k_3

mit $u_1 > u_2 > u_3$, dann ist im Idealfall $k_1 < k_2 < k_3$.

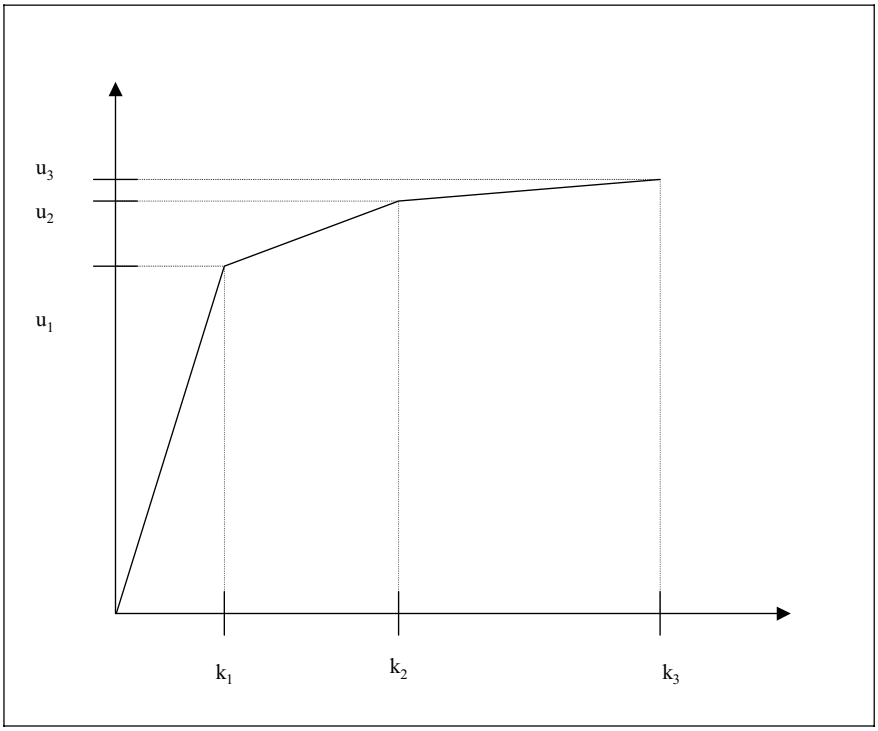


Abbildung 10: Beispielhafte Kundenumsatzanalyse
Quelle: Eigene Darstellung

Bei der **Kundendeckungsbeitragsrechnung** werden die Kosten, die dem Kunden direkt zugerechnet werden können, von dessen Umsatz abgezogen (Link/Hildebrand, 1997b, S. 163). Somit lässt sich ein konkreter Wert bestimmen - der Deckungsbeitrag. Er gibt an, wie profitabel ein Kunde ist. Anhand dieser Informationen lässt sich ebenfalls wieder eine ABC-Analyse durchführen. Voraussetzung dafür ist allerdings ein entwickeltes Rechnungswesen, das die individuelle Zurechnung der Kosten ermöglicht. Zu beachten ist, dass die schlechtesten 20% der Kunden häufig mehr Kosten als Ertrag verursachen (Hughes, 1996, S. 155).

Das **Kundendeckungsbeitragspotenzial** berücksichtigt zusätzlich zu dem aktuellen noch den für die Zukunft erwarteten Deckungsbeitrag. Um einen Neukunden zu gewinnen, werden anfangs Verluste in Kauf genommen, wenn erwartet werden kann, dass diese mit zunehmender Dauer der Kundenbeziehung zumindest kompensiert werden können. Bei der Prognose für die zukünftige Entwicklung orientiert man sich an Vergleichs- bzw. Referenzkunden mit ähnlichem Merkmalsprofil (Link/Hildebrand, 1997b, S. 163).

Die Methode des **Customer-Lifetime-Value** (CLV) orientiert sich an den Prinzipen der Investitionsrechnung. Der CLV ist der Kapitalwert der Deckungsbeiträge eines Kunden, die vor, während und nach der Kundenbeziehung entstehen (Gierl/Koncz, 2002, S. 940; Shaw/Stone, 1988, S. 136ff.). Es werden, wie bei dem vorherigen Verfahren, sämtliche erwarteten zukünftigen Umsätze und Kosten gegenübergestellt, jedoch auf den heutigen Zeitpunkt diskontiert. Der Vorteil des CLV ist, dass beispielsweise die maximalen Akquisitionskosten für neue Kunden oder die maximale Höhe der Kundenbindungskosten für vorhandene Kunden abgeschätzt werden können (Gierl/Koncz, 2002, S. 939).

Die beiden zuletzt genannten Methoden sind aufgrund der notwendigen Prognosen insofern problematisch, als das Kundenverhalten über einen längeren Zeitraum vorherbestimmt werden muss (Huldi, 1997, S. 608).

2.2.3.3 Punktbewertungsverfahren

Die Punktbewertungsverfahren enthalten die Scoring-Modelle und die Kundenportfolio-Analyse, bei denen jeweils für jeden Kunden eine aggregierte Kennzahl zu dessen Bewertung berechnet wird. Diese Kennzahl kann auch als Index für die Attraktivität des Kunden für das Unternehmen interpretiert werden.

Die **Scoring-Modelle** berücksichtigen neben monetären Größen auch andere kaufverhaltensrelevante Merkmale zur Vorhersage des künftigen Kaufverhaltens (Link/Hildebrand, 1997b, S. 166).

Das bekannteste Modell ist die **RFMR-Methode**, die von amerikanischen Versandhandelsunternehmen in den zwanziger Jahren eingeführt wurde (Stone, 1989, S. 30ff.; Link/Hildebrand, 1993, S. 48ff.; Schaller, 1988, S. 122ff.; Hughes, 1996, S. 156ff.).

RFMR setzt sich aus drei Merkmalen zusammen:

R	=	Recency	=	Zeitpunkt des letzten Kaufes,
F	=	Frequency	=	Kaufhäufigkeit,
MR	=	Monetary Ratio	=	Wert des Kaufes.

Bei der Punktevergabe unter Berücksichtigung der RFMR-Methode (siehe Abbildung 11) wird Kunden, deren Käufe noch nicht so lange zurückliegen, ein höherer Wert gutgeschrieben als Kunden, die seit längerem nicht gekauft haben. Mehrfachbesteller erhalten mehr Punkte als Einmalkunden. Kunden mit einem höheren Umsatz pro Bestellung werden ebenfalls besser bewertet (Schaller, 1988, S. 122f.). Andererseits werden Punkte abgezogen, wenn in den Kunden investiert wird, z.B. bei einer Mailingaktion oder bei einem Katalogversand. Die Anzahl der Punkte, die pro Vorgang gutgeschrieben bzw. abgezogen werden, wird individuell vom Anwender festgelegt. Je mehr Punkte der Kunde ansammelt, desto höher ist er in seiner Bedeutung für das Unternehmen einzustufen. Konkrete Marketingmaßnahmen könnten somit an eine bestimmte Mindestpunktezahl gebunden werden. Das hilft beispielsweise Kosten zu sparen, wenn Kunden mit niedriger Attraktivität ausgeschlossen werden sollen.

Ein hypothetisches Beispiel für eine Bewertung von Experten nach der RFMR-Methode zeigt Abbildung 11:

Kriterien						
Startwert:	25 Punkte					
Letztes Kaufdatum	bis 6 Monate: + 40 Punkte	6 bis 9 Monate: + 25 Punkte	9 bis 12 Monate: + 15 Punkte	12 bis 18 Monate: + 5 Punkte	18 bis 24 Monate: - 5 Punkte	>= 24 Monate: - 15 Punkte
Häufigkeit der Käufe in den letzten 18 Monaten	Zahl der Aufträge multipliziert mit dem Faktor 6					
Durchschnittlicher Umsatz der letzten 3 Käufe	bis 50 GE: + 5 Punkte	50 bis 100 GE: + 15 Punkte	100 bis 200 GE: + 25 Punkte	200bis 300 GE: + 35 Punkte	300 bis 400 GE: + 40 Punkte	>= 400 GE: + 45 Punkte
Anzahl Retouren	0-1: 0 Punkte	2-3: -5 Punkte	4-6: -10 Punkte	7-10: -20 Punkte	11-15: -30 Punkte	über 15: -40 Punkte
Zahl der Werbesendungen seit letzten Kauf	Hauptkatalog: je -12 Punkte		Sonderkatalog: je -6 Punkte		Mailing: je -2 Punkte	

Abbildung 11: RFMR-Methode
Quelle: In Anlehnung an Link/Hildebrand (1993, S. 49)

Es ist die Einbeziehung weiterer Merkmale denkbar, obwohl sich die drei Merkmale der RFMR-Methode immer wieder als außerordentlich treffliche Indikatoren für das künftige Kaufverhalten erwiesen haben (Link/Hildebrand, 1997b, S. 166).

Eine weitere Möglichkeit ist die Modellierung der Response eines Kunden auf Basis einer Testaussendung oder bereits abgeschlossener Werbeaktionen mit multivariaten Analysemethoden, beispielsweise einer logistischen Regression (Bausch, 1991, S. 86; Huldi, 1992, S. 139). Es werden dabei im Vergleich zur RFMR-Methode alle vorhandenen Informationen, die über den Kunden zum Zeitpunkt der Werbeaktion vorliegen, verwendet. Die berechneten Scorewerte für jeden Kunden werden mit einem vom Anwender festgelegten Schwellenwert verglichen. Liegt der Scorewert über dem Schwellenwert, wird der Kunde als attraktiv eingestuft, sonst als nicht attraktiv.

Bei der **Kundenportfolio-Analyse** wird jeder Kunde hinsichtlich seiner Investitionswürdigkeit anhand zweier unterschiedlicher Dimensionen, beispielsweise Kundenattraktivität und die eigene Unternehmensposition im Wettbewerb (siehe Abbildung 12), bewertet (Link/Hildebrand, 1993, S. 50). Bewertungskriterien für die Kundenattraktivität können beispielsweise der gegenwärtige und zukünftig zu erwartende Gesamtbedarf, seine Bonität, seine Preissensibilität oder eine Aggregation aus mehreren Kriterien sein. Als Kriterien für die eigene Unternehmensposition wären z.B. der eigene Lieferanteil, das Produkt/Firmen-Image oder eine Aggregation aus mehreren Kriterien zu nennen. Die Kunden werden anhand der ermittelten Koordinaten klassifiziert. Für jedes Feld

können nun Handlungsempfehlungen hinsichtlich zukünftiger Investitionsentscheidungen abgeleitet werden.

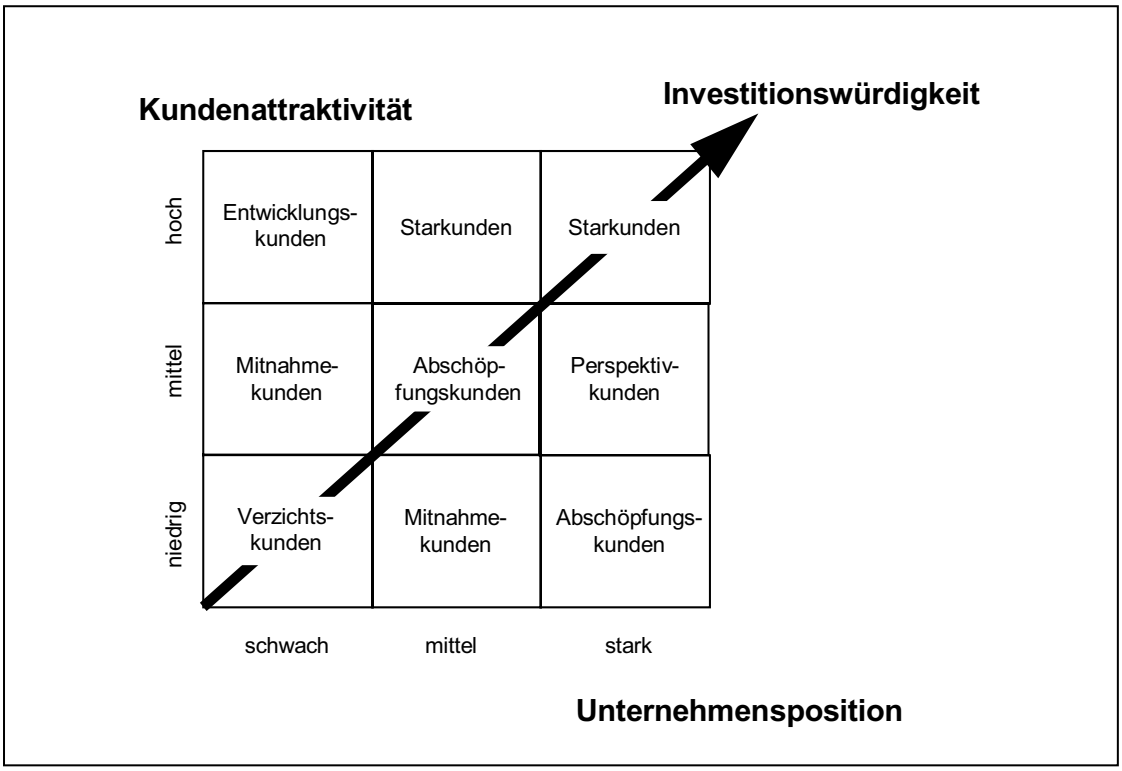


Abbildung 12: Kundenportfolio
Quelle: Link/Hildebrand (1993, S. 52)

Aus den neun Feldern lassen sich sechs interpretativ verschiedene Kundensegmente bilden (Link/Hildebrand, 1993, S. 52f.). Die Starkunden, die so gesehen wertvollsten Kunden, rechtfertigen hohe Investitionen, um sie lange an das Unternehmen zu binden. Bei den Entwicklungskunden ist im Einzelfall zu prüfen, ob sich der Kunde zum Starkunden entwickeln lässt, indem die eigene Unternehmensposition verbessert wird, oder, falls die Konkurrenz beispielsweise zu groß ist, als Mitnahmekunde behandelt wird. In Perspektivkunden ist zu investieren, da sie zu Starkunden werden können. Falls deren Attraktivität nach einem festgelegten Zeitraum nicht den Erwartungen des Unternehmens entspricht, sollten sie als Abschöpfungskunden eingestuft werden. Der Mitteleinsatz in Abschöpfungskunden sollte so bemessen sein, dass die eigene Position verteidigt werden kann. Bei diesen Kunden werden die Mittel gewonnen, die für die Investition in Perspektiv- oder Entwicklungskunden benötigt werden. In Verzichtskunden wird nicht mehr investiert, da ihr Umsatzpotenzial und die Wachstumsaussichten gering sind. Allerdings sollte immer bedacht werden, dass zum einen oft keine eindeutige Zuordnung

möglich und zum anderen eine unreflektierte Übernahme von Normstrategien nicht immer sinnvoll ist (Link/Hildebrand, 1997b, S. 169).

2.2.3.4 Clusteranalytische Verfahren

Die Bildung von Kundentypen mit Hilfe der Clusteranalyse hat den Vorteil, dass es auf der Grundlage der im Unternehmen vorhandenen Kundendaten durchgeführt werden kann (Link/Hildebrand, 1993, S. 56). Die Clusteranalyse bildet auf der Basis der vorgegebenen Merkmale Gruppen, die in sich möglichst homogen, untereinander jedoch weitgehend heterogen sind. Kunden, die in einer Klasse sind, weisen somit ein ähnliches Merkmalsprofil auf. Auf dieser Basis können dann clusterspezifische Marketingstrategien im Rahmen des CRM entwickelt werden, die auf das jeweilige Kundenverhalten zugeschnitten sind.

2.2.2 Data Mining

Data Mining entspricht der Phase Datenanalyse des in Abbildung 5 (siehe S. 18) dargestellten KDD-Prozesses. Fayyad/Piatetsky-Shapiro/Smyth (1996, S. 9) definieren Data Mining als einen Teilschritt des KDD-Prozesses, der aus bestimmten Algorithmen besteht, die in akzeptabler Rechenzeit aus einer vorgegebenen Datenbasis eine Menge von Mustern liefern.

Die Ziele und Aufgaben des Data Mining sind klar festgelegt. Aus einer vorgegebenen Datenmatrix sollen Zusammenhänge abgeleitet werden, die dem Anwender neues Wissen liefern. Um dieses Wissen zu erlangen, werden Algorithmen aus unterschiedlichen Bereichen angewandt. Es wird neben Verfahren der klassischen Datenanalyse auch auf neuere heuristische Verfahren zurückgegriffen. Als Ziel des Data Mining wird die Bereitstellung allgemein verwendbarer, effizienter Methoden angegeben, um weitgehend autonom aus großen Datenbeständen die bedeutsamsten und aussagekräftigsten Muster zu identifizieren (Matheus/Chan/Piatetsky-Shapiro, 1993, S. 903).

Eine Begriffsabgrenzung ist im Hinblick auf die Autonomie dieser Methoden vorzunehmen. Einige Autoren beschreiben Data Mining als einen Prozess, der mit einer vom Anwender vorgegebenen Hypothese beginnt und dann schrittweise und abhängig von Zwischenergebnissen zu Aussagen über die Daten führt (Agrawal/Imilienski/Swami,

1993, S. 914). Andere Autoren vertreten die Ansicht, dass keine vorgegebene Hypothesen getestet, sondern allgemein formulierte Auffälligkeiten gesucht werden sollen, die dem Anwender dann in Form von Aussagen oder Regeln präsentiert werden (Gebhardt, 1994, S. 9). Säuberlich (2000a, S. 11) geht davon aus, dass am Anfang jeder Data Mining Anwendung ein mehr oder weniger konkret spezifiziertes Analyseziel steht. Dies muss nicht in Form zu überprüfender Hypothesen vorliegen, aber das Anwendungsgebiet und die mit der Analyse zu erreichenden Zielsetzungen sollten bekannt sein.

Die Aufgabenstellung des Data Mining umfasst somit vor allem Methoden und Modelle zur Aufdeckung von Abhängigkeiten und Zusammenhängen, sowie zur Datenverdichtung. Folgende Bezeichnungen haben sich eingebürgert:

- ∉ Modelle und Methoden der Klassifikation / Regression,
- ∉ Modelle und Methoden der Segmentierung,
- ∉ Modelle und Methoden der Abhängigkeitsanalyse.

Im Bereich der **Klassifikation** geht es um die Frage, ob und gegebenenfalls wie eine nominale abhängige Variable durch relevante unabhängige Variablen erklärt werden kann. Kann eine entsprechende Abhängigkeit nachgewiesen werden, so kann für jedes neue Objekt, dessen Ausprägungen bei den unabhängigen Variablen bekannt sind, die Ausprägung der abhängigen Variable geschätzt bzw. prognostiziert werden. Ist die abhängige Variable metrisch, so spricht man von einer **Regression**. Opitz (1978, S. 7) verwendet für diese Aufgabenstellung den Begriff **Identifikation**.

Identifikationsverfahren lassen sich folgendermaßen unterteilen (in Anlehnung an Opitz, 1980, S. 158):

- Sukzessive Verfahren, z.B. Entscheidungsbaumverfahren und K-nächste Nachbarn-Verfahren,
- Simultan-statistische Verfahren, dazu zählen z.B. die Diskriminanzanalyse, Regression und logistische Regression ,
- Simultan-numerische Verfahren, beispielsweise Neuronale Netze.

Entscheidungsbaumverfahren untersuchen die Abhängigkeit der unabhängigen Variablen von der Zielvariable in sukzessiver Weise. Sowohl die unabhängigen Variablen

als auch die abhängige Variable können beliebiges Skalenniveau aufweisen. Die Objekte werden anhand der Ausprägungen ausgewählter unabhängiger Variablen so lange sukzessive in Teilmengen unterteilt, bis auf der untersten Ebene in bezug auf die abhängige Variable weitgehend homogene Gruppen entstanden sind. In Kapitel 4 werden diese Verfahren ausführlich beschrieben und angewandt.

Das **K-Nächste-Nachbarn** Verfahren ist ebenfalls ein sukzessives Verfahren. Die abhängige Variable kann nominal oder metrisch sein, die unabhängigen Variablen können beliebiges Skalenniveau aufweisen. In einem ersten Schritt werden für ein noch nicht klassifiziertes Objekt aufgrund seiner Merkmalsausprägungen die K ähnlichsten Objekte herangezogen, für welche die Zielvariable bereits bekannt ist. Die Ähnlichkeit der Objekte wird dabei anhand der Distanz zwischen den Objekten bestimmt. Das neue Objekt wird schließlich derjenigen Klasse zugeordnet, welcher die Mehrzahl dieser K Objekte angehört. Bei einer metrischen abhängigen Variable wird dem neuen Objekt der Mittelwert dieser K Objekte zugewiesen. Dieses Verfahren basiert auf dem Prinzip „Lernen durch Analogien“ (Dasarathy, 1991; Mitchell, 1997, S. 230ff.). Formal gesehen wird bei diesem Algorithmus zuerst die Distanz zwischen dem neuen, noch nicht klassifizierten Objektvektor x_{neu} und allen anderen Objektvektoren x_j, für welche die Klasse bereits bekannt ist, berechnet: $d(x_{neu}, x_j)\ \&\ j$. Im Anschluss werden die x_1, ..., x_K bestimmt, welche die geringste Distanz zu x_{neu} aufweisen. Diese werden mit $N_K(x_{neu})$ als Nachbarschaft von x_{neu} bezeichnet. x_{neu} wird bei einer nominalen Zielvariable y der Modalwert der y-Werte der Nachbarschaft von x_{neu} zugewiesen:

$$\hat{f}(x_{neu}) \mid \text{mod}/y \mid x \subset N_K(x_{neu})0.$$

Bei einer metrischen Zielvariable wird x_{neu} folgender Wert zugeordnet:

$$\hat{f}(x_{neu}) \mid \frac{\overset{K}{\overline{\quad}} y(x_i)}{\underset{i\mid 1}{\quad}}{K}.$$

Die Aufgabe der **Diskriminanzanalyse**, die zu den simultan-statistischen Verfahren zählt, ist die Analyse von Gruppenunterschieden (Anderson, 1958). Die abhängige Variable entspricht dabei der Klassenzugehörigkeit der Objekte und ist somit nominal skaliert, die unabhängigen Variablen sind metrisch. Handelt es sich um zwei Klassen,

spricht man von einer einfachen, bei mehr als zwei Klassen von multipler Diskriminanzanalyse.
Formal bedeutet dies im linearen Fall:
$Y - g_1X_1+...+g_pX_p$, wobei X_1, ..., X_p metrisch und Y nominal.

Gesucht sind die Gewichte g_1, ..., g_p, so dass die Linearkombination $\sum_{i=1}^{p} g_i X_i \approx \hat{y}$ die durch y gegebene Struktur bestmöglich approximiert. Sind zwei Objekte aus einer Klasse, so soll die errechnete Differenz der beiden $\hat{R}$ sehr klein sein, sind sie aus verschiedenen Klassen, soll die Differenz der beiden $\hat{R}$ möglichst groß sein.

Bei der **linearen Regression** ist die abhängige Variable im Unterschied zur Diskriminanzanalyse metrisch skaliert, ebenso die unabhängigen Variablen. Formal gesehen, ist $y=X\eta+u$. $\hat{\eta}$ wird dabei aus $\hat{\eta} \approx (X^T X)^{41} X^T y$ berechnet, damit ist $\hat{y} \approx X\hat{\eta}$. Das Ziel dabei ist, $\hat{\eta}$ mit Hilfe der Kleinst-Quadrat-Schätzung so zu bestimmen, dass die Summe der quadratischen Abweichungen zwischen y und $\hat{R}$ minimiert wird.

Bei der **logistischen Regression**, deren Ansatz nicht-linear ist, ist die abhängige Variable nominal, die unabhängigen Variablen können beliebig skaliert sein. Hier wird die Wahrscheinlichkeit der Zugehörigkeit zu einer Klasse bestimmt. In Kapitel 5 wird dieses Verfahren ausführlich beschrieben und angewandt.

Neuronale Netze, als Vertreter der simultan-numerischen Verfahren, sind in der Lage, nichtlineare Strukturen abzubilden. Die unabhängigen Variablen können dabei beliebiges Skalenniveau aufweisen, die abhängige Variable kann metrisch oder nominal skaliert sein. Es wird ein Netz aus mehreren miteinander verknüpften Neuronen gebildet. Die Gewichte zwischen den Neuronen werden simultan nach einem vorgegebenen Lernalgorithmus iterativ so angepasst, dass die Netzwerkfunktion die Werte der abhängigen Variablen möglichst gut erklärt. In Kapitel 6 wird dieses Verfahren ausführlich beschrieben und angewandt.

Typische Anwendungen für eine Klassifikation bzw. Regression sind beispielsweise:

- Responseoptimierung bzw. Berechnung von Bestellwahrscheinlichkeiten: Die Zielvariable ist hier nominal binär, die unabhängigen Variablen weisen beliebiges Skalenniveau auf. Das Ziel ist dabei die Minimierung von Streuverlusten.

Magidson (1988, S. 17) stellt fest, dass die Modellierung mit Data Mining Verfahren der RFMR-Methode überlegen ist.

- Kreditwürdigkeitsprüfung: Es wird für jeden Kunden geprüft, ob er kreditwürdig ist. Die Zielvariable ist hier wiederum nominal binär, während die unabhängigen Variablen beliebig skaliert sein können.
- Umsatzprognose: Es soll für jeden Kunden eine möglichst genaue Schätzung seines künftigen Umsatzes mit dem Unternehmen abgegeben werden. Die Zielvariable ist hier metrisch, die unabhängigen Variablen sind beliebig skaliert.

Im Bereich **Segmentierung** geht es um die Frage, wie die Datenmenge in Gruppen aufgeteilt oder Einzelobjekte zu Gruppen zusammengefasst werden können, so dass Objekte innerhalb einer Gruppe sehr ähnlich zueinander und Objekte zwischen den einzelnen Gruppen sehr unähnlich sind. Damit ist das Ziel der Segmentierung, aus einer großen Menge von Daten überschaubare und interpretierbare Gruppen zu bilden.
Zur Segmentierung werden im wesentlichen Clusteranalyseverfahren oder Kohonen Self Organizing Maps (SOM) verwendet. Die Basis bei **Clusteranalyseverfahren** ist eine Distanzmatrix, die die paarweise Unähnlichkeit zwischen den Objekten zum Ausdruck bringt. Darauf können anschließend unterschiedliche Algorithmen angewandt werden, um möglichst homogene Gruppen zu erhalten. Im Bereich der Voranalyse wird dieses Verfahren in Kapitel 3.3.2 (S. 61ff.) ausführlich beschrieben und angewandt. Bei den **Kohonen SOM** handelt es sich um ein auf Neuronalen Netzen basierendes Verfahren. Dabei gibt es nur die Eingabe- und Ausgabeschicht, wobei die Eingabe- und Ausgabeneuronen vollständig miteinander verbunden sind. Zu Beginn erfolgt eine zufällige Zuordnung der Objekte. Die Objekte werden dann auf die Ausgabeschicht projiziert und einem bestimmten Ausgabeneuron zugeordnet. Anschließend werden die Verbindungsgewichte angepasst und das Verfahren wird iterativ fortgesetzt, bis ein bestimmtes Abbruchkriterium erreicht ist. Zur genauen Beschreibung des Algorithmus wird auf Kohonen (1997) verwiesen.

Eine typische Anwendung ist beispielsweise die Kundensegmentierung. Dabei sollen aus der großen Menge aller Kunden interessante Gruppen entdeckt werden. In dieser Arbeit wird die Clusteranalyse im Rahmen der Voranalyse zur Unterstützung der Stichprobenziehung verwendet (siehe Kapitel 3.3.2, S. 61ff.).

Das Ziel von sogenannten **Abhängigkeitsanalysen** ist die Aufdeckung von strukturellen Zusammenhängen in den Daten. Dazu werden sowohl die Korrelationsanalyse als auch Assoziationsregelalgorithmen, z.B. der A-Priori Algorithmus (Agrawal/Srikant, 1994), eingesetzt. Die Korrelationsanalyse dient dazu, Zusammenhänge zwischen den Variablen aufzudecken. Ein umfassender Überblick ist in Hilbert (1998) zu finden. Assoziationsregelalgorithmen liefern Aussagen in der folgenden Form: A $\Downarrow$ B, das heißt, wenn A, dann folgt B. Dabei werden meist mehrere Kennzahlen zur Bewertung mitgeliefert, zwei davon sind z.B. (Küsters, 2001, S. 115):

$$\text{Support} = P(A \sim B),$$

$$\text{Confidence} = P(B \mid A).$$

Der Support gibt dabei den Anteil an Beobachtungen an, der die Regel erfüllt. Die Confidence gibt den Anteil an Beobachtungen an, die B beinhalten, an der Menge von Beobachtungen, die A enthalten.

Typische Anwendungsgebiete sind hier:

- Variablenreduktion: Mit Hilfe der Korrelationsanalyse sollen redundante Variablen identifiziert werden.
- Warenkorbanalyse oder Cross-Selling Analyse: Dabei werden Produkte, die häufig zusammen gekauft werden, identifiziert.
- Sequenzanalyse: Ist zusätzlich noch eine Zeitvariable vorhanden, so können auch sequentielle Muster im Zeitablauf beschrieben werden. Dies ist beispielsweise bei Logfile-Analysen von Internetseiten der Fall.

Zu beachten ist, dass als Voraussetzung für alle Verfahren eine angemessene Datenbasis vorhanden sein muss. In diesem Zusammenhang wird häufig das Sprichwort „Garbage in – Garbage out“ verwendet. Die Datenaufbereitung bildet die Grundlage für alle nachfolgenden Analysen und trägt damit einen großen Teil für den Erfolg eines Data Mining Projektes bei. Dabei treten häufig eines oder mehrere der folgenden Probleme auf (Säuberlich, 2000a, S. 14f.):

∉ Dynamik in den Daten

Teilweise ändern sich Daten sehr kurzfristig, deshalb ist es wichtig, diese mit einem Zeitstempel zu versehen. Es ist erforderlich, die Daten im Analysebestand regelmäßig zu aktualisieren.

- Verschmutzte Daten

  Fehlerhafte Daten oder Fehler bei der Erfassung von Daten treten in der Praxis häufig auf und können zu falschen Mustern führen. Diese müssen im Rahmen der Datenvorverarbeitung eliminiert werden.

- Fehlende Werte

  Fehlende Werte, das heißt Einzelausprägungen bei einer Variablen, können Data Mining Verfahren erheblich beeinträchtigen. Diese müssen im Rahmen der Datenvorverarbeitung beispielsweise mit geeigneten Imputationsverfahren ersetzt werden.

- Unvollständige Daten

  Häufig werden nur die im operationalen Prozess benötigten Daten gespeichert. Es kann jedoch sein, dass für die Analyse wichtige Daten, das heißt Variablen, nicht vorliegen. Diese sollten entweder vorher erhoben, nacherhoben oder abgeschätzt werden.

- Redundanz

  Sich bedingende Informationen können an mehreren Stellen in der Datenbank gespeichert sein. Diese können bei Anwendung von Data Mining Verfahren zu verzerrten Ergebnissen führen. Deshalb ist es wichtig, dass der Anwender derartige Redundanzen entdeckt.

- Datenvolumen

  Zu große Datenvolumen können dazu führen, dass bestimmte Algorithmen nicht auf der gesamten Datenmatrix angewandt werden können. Deshalb wird häufig nur auf Basis einer Teildatenmatrix und/oder einer Auswahl von Merkmalen gearbeitet.

Insgesamt ist zu dem Data Mining Prozess anzumerken, dass der Erkenntnisgewinn keinesfalls in „vollautomatischer“ Form abläuft. Er ist eher das Ergebnis eines vergleichsweise flexiblen Datenanalyseprozesses, der dem Anwender immer noch ein angemessenes Maß an methodischem Know-how abverlangt (Temme/Decker, 1999, S. 16; Murthy ,1998, S. 374).

2.2.3 OLAP

Zur Analyse bzw. Darstellung umfangreicher Daten werden spezielle Werkzeuge benötigt. Dazu wurde von Codd/Codd/Sally (1993) das Konzept des **On-Line Analytical Processing** (OLAP) entwickelt. Dabei werden betriebswirtschaftlich relevante Daten in einem multidimensionalen Datenwürfel abgebildet (Abbildung 13).

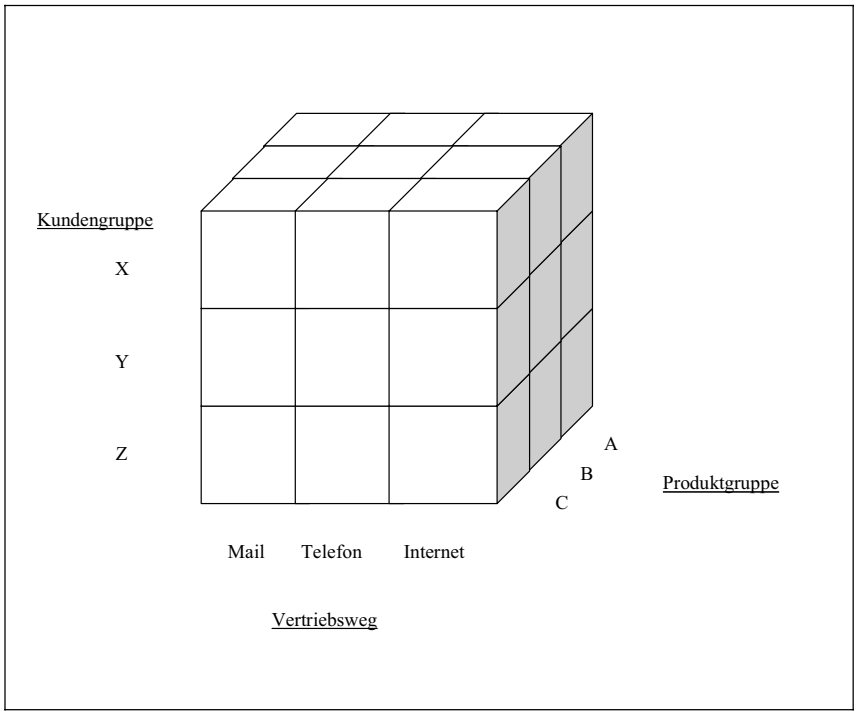


Abbildung 13: OLAP-Datenwürfel
Quelle: Eigene Darstellung

Die Dimensionen entsprechen meist betriebswirtschaftlichen Merkmalen oder Gruppierungsergebnissen, beispielsweise Produktgruppen, Kundengruppen oder Vertriebswege (siehe Kapitel 2.1.1, S. 15). Ein Gruppierungsergebnis könnte beispielsweise auch eine Kundenpotenzialgruppe sein, die durch Data Mining Methoden gebildet wurde. Entlang dieser Dimensionen können die entsprechenden Maßzahlen aufgebrochen oder aggregiert werden. Typische Navigationsfunktionen sind beispielsweise Drill down, Roll up, Slice und Dice. **Drill Down** bedeutet dabei eine Erhöhung des Detaillierungsgrades einer Dimension, indem beispielsweise die Daten von einer Kundengruppe auf einzelne Kunden aufgebrochen werden. **Roll Up** stellt genau das Gegenteil dar, dabei wird eine Senkung des Detaillierungsgrades bewirkt. Beim **Slicing** wird ein Unterraum des betrachteten Datenwürfels durch Weglassen einer Dimension betrachtet, beispielsweise wird die Dimension Vertriebsweg weggelassen. Dadurch reduziert sich der Würfel auf

ein Quadrat. Beim **Dicing** wird der betrachtete Datenwürfel durch Weglassen von Ausprägungen einer Dimension vergröbert, beispielsweise werden die Ausprägungen Mail und Internet bei der Dimension Vertriebsweg weggelassen.
Mit der intuitiv bedienbaren Benutzeroberfläche soll dem Management direkter Zugriff auf möglichst viele Daten gegeben werden (Wilde, 2001, S. 12).

Zu beachten ist allerdings, dass OLAP eine rein deskriptive Darstellung der Daten innerhalb vorher festgelegter Möglichkeiten liefert. Der Anwender gibt die zu analysierenden Dimensionen stets vor, das heißt OLAP ist nur zur manuellen Suche nach interessanten Datenausschnitten in den Daten geeignet. OLAP-Werkzeuge sind beispielsweise zur explorativen Datenanalyse oder zur anschaulichen Präsentation von Daten sehr gut geeignet.

2.3 Zusammenhang der informationstechnischen und methodischen Grundlagen

Im folgenden wird der Zusammenhang der **informationstechnischen** und **methodischen Grundlagen** dargestellt. Weiterhin wird der Bezug zu den Bereichen CRM bzw. Direktmarketing hergestellt.

Im DWH werden alle Daten, die in einem Unternehmen vorhanden sind, zusammengetragen und in einer Datenbank gespeichert. Dadurch liegen alle im Unternehmen vorhandenen Informationen gesammelt und aufbereitet vor. Allerdings ist es aufgrund der großen Menge an Information schwierig, wirklich wertvolle Informationen bzw. interessante Muster in den Daten zu finden. John Naisbett brachte dies mit folgenden Worten zum Ausdruck: „We are drowning in information, but starving for knowledge“ (Fayyad/Piatetsky-Shapiro/Smyth, 1996, S. 1). Deshalb werden Algorithmen benötigt, die in großen Datenmengen weitgehend autonom wertvolle Informationen finden können; die Disziplin KDD entwickelte sich.

OLAP bzw. Data Mining, die zu der Datenanalysephase im KDD-Prozess (Abbildung 5, S. 18) zählen, und DWHs ergänzen sich. DWHs organisieren und speichern die Daten, OLAP dient der Anzeige von Datenausschnitten aus dem DWH und Data Mining deckt methodengestützt die Zusammenhänge in den Daten auf. Dadurch werden aus Daten entscheidungsrelevante Informationen und Wissen gewonnen. Zur Abgrenzung Data

Mining – OLAP lässt sich sagen, dass OLAP eine maschinell gestützte, manuelle Suche nach in den Daten verborgenen, interessanten Zusammenhängen ist, während Data Mining eine manuell unterstützte, maschinelle Suche darstellt. Zur Veranschaulichung bestimmter Datenausschnitte dient OLAP, diese müssen jedoch vom Anwender vorgegeben werden. Data Mining Verfahren helfen methodengestützt dabei, in großen Datenmengen weitgehend autonom interessante Muster zu entdecken. Der gesamte Prozess der Wissensentdeckung wird Knowledge Discovery in Databases genannt.

Dies alles geschieht mit dem Ziel, mehr Wissen über den Kunden zu erlangen und die Kommunikation mit dem Kunden effizienter zu gestalten. Im Rahmen der Kundenorientierung der Unternehmen ist es das Ziel mit dem Konzept des CRM den Kontakt mit dem Kunden zu individualisieren, um eine langfristige Beziehung zu dem Kunden aufzubauen. Es wird versucht, durch die Analyse aller vorhandenen Kundendaten den Kundenbedarf zu antizipieren und dadurch die Kommunikation mit dem Kunden zu optimieren. Die Basis hierfür liefern Ergebnisse aus dem analytischen CRM, das heißt aus OLAP und Data Mining. Kundeninformationen werden somit zum strategischen Erfolgsfaktor (Link/Hildebrand, 1997a, S. 21).

Die Entwicklung vom Direktmarketing über das Database Marketing zum CRM ist vor allem durch die Entwicklung von Speichermedien und Systemen zur Abwicklung bzw. Analyse von Geschäftsprozessen geprägt. Das Database Marketing basiert auf Informationen, die in den Datenbanken eines Unternehmens gespeichert werden. Meist erfolgt die Speicherung der relevanten Daten bereits in einem DWH. Aus dem Databasemarketing heraus entwickelte sich der Bedarf, die Kunden zu bewerten. Dadurch wird es möglich, Kunden nach ihrer Attraktivität für das Unternehmen einzuordnen. Es wurden verschiedene traditionelle Kundebewertungsverfahren vorgestellt, die häufig mit Hilfe von Data Mining Methoden durchgeführt werden. Dazu zählt auch die Responseoptimierung, bei der Kunden für eine Werbeaktion zu einem bestimmten Produkt bewertet werden, um im Rahmen des Database Marketings nur die erfolgversprechendsten Kunden für eine Direktmarketingaktion auswählen zu können.
Das traditionelle Databasemarketing wird im Rahmen des CRM-Konzepts durch intelligente Auswertungs- (analytisches CRM) und Steuerungssysteme (operatives CRM) erweitert (Meffert, 2002, S. 48).

3. Zweckmäßige Voranalysen

Ausgehend von den verfügbaren Daten erfolgt im folgenden in Anlehnung an den KDD-Prozess die Datenaufbereitung und die Datenvorverarbeitung (siehe Abbildung 5, S. 18). Dabei werden vor allem verschiedene Methoden zur Bewältigung niedriger Responsequoten vorgestellt. Außerdem wird ein neuer Datenvorverarbeitungsschritt, eine clusteranalysegestützte Stichprobenziehung, beschrieben. Nach den theoretischen Ausführungen folgt jeweils die empirische Analyse einer Datenmatrix, die für eine Responseoptimierung bei WEKA MEDIA generiert wurde. Zum Schluss erfolgt eine Zusammenfassung der Ergebnisse und der Versuchsaufbau dieser Studie wird übersichtlich dargestellt. Zur Durchführung der Analysen wird das Data Mining Tool „SAS™ Enterprise Miner│ (Version 4.0)“ (SAS EM) des SAS™ Institutes (siehe Anhang K, S. 222f.) eingesetzt.

3.1 Datenerfassung

In dieser Studie soll eine Werbeaktion für ein bestimmtes Produkt optimiert werden. Als Basis für eine Responseoptimierung (siehe S. 8f.) können entweder eine spezielle Testaussendung oder geeignete, bereits abgeschlossene Werbeaktionen verwendet werden. Hier werden fünf abgeschlossene Werbeaktionen für denselben Artikel als Basis verwendet. Das Ergebnis ist eine Datenmatrix mit den Objektvektoren in den Zeilen und den Merkmalsvektoren in den Spalten. Eine Zeile in der Datenmatrix entspricht somit einem Kunden. Im folgenden wird dafür der Begriff Objektvektor oder Objekt verwendet. Die relevanten Merkmale sollen dabei nach ökonomischen und sachlogischen Gründen ausgewählt werden (Anders, 1995, S. 28; Baun, 1994, S. 134f.; Sodeur, 1974, S. 70). In der Praxis wird die Auswahl der Variablen meist von Experten der Marketingabteilung getroffen, die durch langjährige Erfahrung bereits bestimmte kaufrelevante Kriterien identifizieren konnten. Eine weitere Quelle stellen die in der Datenbank vorhandenen Daten dar. Im nächsten Schritt wird eine Datenmatrix gebildet, die zu den beworbenen Kunden eine Vielzahl an Informationen in Form von Merkmalen aus Abbildung 1 (siehe S. 5) enthält und ein zusätzliches Merkmal, das die Bestellung eines Kunden auf die entsprechende Werbeaktion anzeigt. Hat ein Objekt bei mehreren Werbeaktionen teilgenommen, so wird im Fall einer Bestellung dieser Objektvektor verwendet. Im Falle der Nichtbestellung wird ein Objektvektor zufällig ausgewählt, so dass

jedes Objekt nur einmal in der Datenmatrix vorhanden ist. Diese Schritte entsprechen der Datenauswahl und Datenaufbereitung im KDD-Prozess (siehe Abbildung 5, S. 18).

3.2 Datenvorverarbeitung

Die Datenvorverarbeitung umfasst alle notwendigen Schritte, um eine analysefähige Datenmatrix zu erhalten. Typischerweise beginnt man seit der Arbeit von Tukey (1977) mit einer explorativen Datenanalyse, bei der die einzelnen Variablen mit Hilfe deskriptiver Statistiken inspiziert werden. „Die deskriptive Statistik dient der Beschreibung, der Strukturierung und der Verdeutlichung unübersichtlichen und umfangreichen Datenmaterials“ (Hartung, 1995, S. 15). Bei nominalen bzw. ordinalen Daten werden vor allem univariate Häufigkeitsauszählungen und bivariate Kreuztabellen zur ersten Veranschaulichung verwendet. Zur Beschreibung metrischer Merkmale wird auf statistische Kennzahlen der Lage und Streuung zurückgegriffen. Im Anschluss werden weitere Datenvorverarbeitungsschritte, wie Variablenmodifikation, Behandlung fehlender Werte, Variablenreduktion und Ausreißeranalyse durchgeführt.

3.2.1 Variablenmodifikation

Hier werden bestimmte Variablen umkodiert, transformiert oder komplett neu berechnet. Bei bestimmten Variablen kann es sinnvoll sein, Ausprägungen zusammenzufassen. Das Merkmal Branche hat beispielsweise 937 hierarchisch geordnete Ausprägungen, wobei einzelne Ausprägungen nur sehr selten auftreten. Die erste Stelle gibt die Hauptbranche an, die nächsten vier Stellen brechen die Struktur sukzessive bis auf die genaueste Ebene herunter. Hier werden zwei neue Variablen gebildet: Das Merkmal *Branche2* besteht aus den ersten beiden Stellen, das Merkmal *Br_k* nur aus der ersten Stelle der ursprünglichen Variablenausprägungen (siehe Anhang A, S. 207f.). In der Datenmatrix werden diejenigen Variablenausprägungen der beiden neuen Variablen, die weniger als 20 Objekte bei der 1-Klasse haben, in eine neue Kategorie „Sonstiges“ zusammengefasst. Dadurch reduziert sich die Anzahl der Ausprägungen bei dem Merkmal *Branche2* von 42 auf zehn und bei dem Merkmal *Br_k* von 17 auf acht.
Analog wird bei den Merkmalen *Rechtsform* und *Bundesland* verfahren, so dass sich die Ausprägungen von 14 auf sieben bzw. von 16 auf elf reduzieren (siehe Anhang A, S. 207f.).

Ohne Relevanz für die Fallstudie zählt die Skalentransformation ebenfalls zur Variablenmodifikation. Beispielsweise können metrische unabhängige Variablen auf ein ordinales oder nominales bzw. ordinale unabhängige Variablen auf ein nominales Skalenniveau transformiert werden. Zu beachten ist, dass bei der Skalentransformation auf eine niedrigere Ebene ein Informationsverlust zu verzeichnen ist.
Tabelle 2 zeigt eine Aufstellung aller zur Verfügung stehenden Variablen.

Nr.	Variablenname	Beschreibung	Ausprägungen	Skalenniveau
Schlüsselvariablen				
1	Actionno	Schlüssel zur Identifikation der Werbeaktion	n ⊂ N	Nominal
2	Id_locat	Schlüssel zur Identifikation des Kunden	n ⊂ N	Nominal
Grunddaten				
3	Anl_date	Anlagedatum der Kundenadresse	SAS-Datumsformat	Metrisch
4	Aust	Ausstattung des Kunden	PC, CD-ROM, Video	Nominal
5	Branche2	Branche der Firma	2-stellige Codierung	Nominal
6	Br_k	Branche der Firma	1-stellige Codierung	Nominal
7	Bula	Bundesland	B: Berlin, BR: Brandenburg, BW: Baden-Württemberg, BY: Bayern, HB: Bremen, HE: Hessen, HH: Hamburg, MV: Mecklenburg-Vorpommern, NS: Niedersachsen, NW: Nordrhein-Westfalen, RP: Rheinland-Pfalz, S: Sachsen, SA: Sachsen-Anhalt, SL: Saarland, SH: Schleswig-Holstein, TH: Thüringen	Nominal
8	Ma_zahl	Mitarbeiterzahl der Firma	1-9; 10-25; 26-49; 50-99; 100-499; > 500; U	Ordinal

9	Rechtsf	Rechtsform der Firma	GmbH; GmbH & Co; GmbH & Co KG; KG; AG; SO; U	Nominal
Reaktionsdaten				
10	Abostat	Abonnementstatus des Kunden	0: kein Abonnement 1: ein Abonnement 2: mehrere Abonnements	Ordinal
11	Aktiv_0	Aktivitätsstatus des Kunden	0: Interessent 1: inaktiver Kunde 2: aktiver Kunde	Ordinal
12	Anz_aabo	Anzahl aktiver Abonnements des Kunden	$n \subset N_0$	Metrisch
13	Anz_al	Anzahl Aktualisierungslieferungen, die der Kunde erhalten hat	$n \subset N_0$	Metrisch
14	Bes_l6m	Bestellung in den letzten 6 Monaten erfolgt	0: Nein 1: Ja	Binär
15	Bes_l12m	Bestellung in den letzten 12 Monaten erfolgt	0: Nein 1: Ja	Binär
16	Best_all	Anzahl Bestellungen des Kunden insgesamt	$n \subset N_0$	Metrisch
17	Best_gw	Anzahl Grundwerksbestellungen des Kunden	$n \subset N_0$	Metrisch
18	Best_j	Anzahl Bestellungen im laufenden Kalenderjahr	$n \subset N_0$	Metrisch
19	Best_j1	Anzahl Bestellungen im letzten Kalenderjahr	$n \subset N_0$	Metrisch
20	Best_j2	Anzahl Bestellungen im vorletzten Kalenderjahr	$n \subset N_0$	Metrisch
21	Best_oth	Anzahl Bestellungen außer Grundwerke	$n \subset N_0$	Metrisch
22	Di_ebest	Tage seit erster Bestellung	$n \subset N$	Metrisch
23	Di_lbest	Tage seit letzter Bestellung	$n \subset N$	Metrisch
24	Di_lpos	Tage seit letzter erhaltener Rechnung	$n \subset N$	Metrisch
25	Kaufstat	Kaufstatus des Kunden	0: Interessent 1: Einfachkäufer 2: Mehrfachkäufer	Ordinal

26	Kdtyp_0	Kundentyp	0: Interessent 1: Kunde 2: Premiumkunde	Ordinal
27	Nums_al	Nettoumsatz mit Aktualisierungslieferungen	$r \subset R$	Metrisch
28	Nums_all	Nettoumsatz insgesamt	$r \subset R$	Metrisch
29	Nums_gw	Nettoumsatz mit Grundwerken	$r \subset R$	Metrisch
30	Nums_j	Nettoumsatz im laufenden Kalenderjahr	$r \subset R$	Metrisch
31	Nums_j_1	Nettoumsatz im letzten Kalenderjahr	$r \subset R$	Metrisch
32	Nums_j_2	Nettoumsatz im vorletzten Kalenderjahr	$r \subset R$	Metrisch
33	Nums_oth	Nettoumsatz außer Grundwerke und Aktualisierungslieferungen	$r \subset R$	Metrisch
34	Pr_4048	Status bei Produkt 4048, das im selben Segment wie das zu bewerbende Produkt ist	A: Aktiv ; I: Inaktiv ; U: noch nie bestellt	Ordinal
35	Pr_5594	Status bei Produkt 5594 (wie bei Pr_4048)	Siehe pr_4048	Ordinal
36	Pr_8048	Status bei Produkt 8048 (wie bei Pr_4048)	Siehe pr_4048	Ordinal
37	Pr_8091	Status bei Produkt 8091 (wie bei Pr_4048)	Siehe pr_4048	Ordinal
38	Pr_9588	Status bei Produkt 9588 (wie bei Pr_4048)	Siehe pr_4048	Ordinal
39	Pr_9664	Status bei Produkt 9664 (wie bei Pr_4048)	Siehe pr_4048	Ordinal
40	Rem_abo	Anzahl Abonnementremissionen	$n \subset N_0$	Metrisch
41	Rem_gw	Anzahl Grundwerkremissionen	$n \subset N_0$	Metrisch
42	Rem_stor	Anzahl Abonnementstornierungen	$n \subset N_0$	Metrisch
43	Ums_al	Bruttoumsatz mit Aktualisierungslieferungen	$r \subset R$	Metrisch
44	Ums_all	Bruttoumsatz insgesamt	$r \subset R$	Metrisch

45	Ums_gw	Bruttoumsatz mit Grundwerken	$r \subset R$	Metrisch
46	Ums_j	Bruttoumsatz im laufenden Kalenderjahr	$r \subset R$	Metrisch
47	Ums_j_1	Bruttoumsatz im letzten Kalenderjahr	$r \subset R$	Metrisch
48	Ums_j_2	Bruttoumsatz im vorletzten Kalenderjahr	$r \subset R$	Metrisch
49	Ums_l6m	Umsatz in den letzten 6 Monaten erfolgt	0: Nein 1: Ja	Binär
50	Ums_l12m	Umsatz in den letzten 12 Monaten erfolgt	0: Nein 1: Ja	Binär
51	Ums_oth	Bruttoumsatz außer Grundwerke und Aktualisierungslieferungen	$r \subset R$	Metrisch
52	Ums_wa	Umsatz pro Werbeaktion	$r \subset R$	Metrisch
Aktionsdaten				
53	Anz_7409	Anzahl Werbeaktionen, die der Kunde für Produkt 7409 erhalten hat	$n \subset N_0$	Metrisch
54	Anz_dm	Anzahl Direct Mails, die der Kunde insgesamt erhalten hat	$n \subset N_0$	Metrisch
55	Anz_tm	Anzahl Telefonmarketingaktionen, an denen der Kunde insgesamt teilgenommen hat	$n \subset N_0$	Metrisch
56	Anz_wa	Anzahl Werbeaktionen insgesamt, für die der Kunde selektiert wurde	$n \subset N_0$	Metrisch
57	Anz_wam6	Anzahl Werbeaktionen, für die der Kunde in den letzten 6 Monaten selektiert wurde	$n \subset N_0$	Metrisch
58	Di_e_wa	Tage seit Erhalt der ersten Werbeaktion	$n \subset N$	Metrisch
59	Di_l_wa	Tage seit Erhalt der letzten Werbeaktion	$n \subset N$	Metrisch
60	Wa_anl	Quotient aus der Anzahl Werbeaktionen dividiert durch Anzahl Tage, seitdem der Kunde in der Datenbank ist	$q \subset Q$	Metrisch

Zielvariable				
61	Best_jn	Ist eine Bestellung erfolgt	0: Nein 1: Ja	Binär

Tabelle 2: Auflistung aller Variablen, Ausprägungen, Skalenniveau

Insgesamt umfasst die Datenmatrix 92.074 Objekte mit 61 Variablen, diese gliedern sich in 2 Schlüsselvariablen, 7 Grunddaten, 43 Reaktionsdaten, 8 Aktionsdaten und 1 Zielvariable (siehe Abbildung 1, S. 5). In der Datenmatrix sind 394 Besteller vorhanden, so dass die Responsequote 0,43% beträgt. Anzumerken ist dabei, dass diese Responsequote nicht der durchschnittlichen Responsequote einer Werbeaktion bei WEKA MEDIA entspricht, sondern nur bei dieser Aktion gilt.

3.2.2 Fehlende Werte

Ein Problem bei Auswertungen stellen fehlende Werte oder Missing Values dar. Es gibt verschiedene Möglichkeiten, wie mit solchen Lücken umzugehen ist:

Wenn bei einzelnen Merkmalen eine große Zahl fehlender Werte auftritt, sollten diese gestrichen werden (Bausch/Opitz, 1993, S. 24). Beispielsweise haben Arndt/Gersten/ Wirth (2001, S. 598) Merkmale, die mehr als 80% fehlende Werte aufweisen, eliminiert. In dieser Datenmatrix ist die Variable Aust (Nr. 4) nur in 0,15% aller Fälle gefüllt und wird deshalb von der weiteren Analyse ausgeschlossen. Bei den Variablen rem_abo (Nr. 40) und rem_stor (Nr. 42) ist in 86% bzw. 93% aller Fälle die Ausprägung „nicht vorhanden“ angegeben, diese Variablen werden ebenfalls eliminiert.

Alternativ können Missing Values mittels sogenannter Imputationsverfahren ersetzt werden. Ein Überblick über Imputationsverfahren ist bei Bankhofer (1995) zu finden. Bei nominalen Merkmalen kann auch die Ausprägung „unbekannt“ vergeben werden (Hippner/Wilde, 2001a, S. 55). Dies führt oft zu interessanten Schlussfolgerungen, da auch die Information „unbekannt“ Hinweise auf ein bestimmtes Kundenverhalten geben kann (Berry/Linoff, 1997, S. 71; Frawley/Piatetsky-Shapiro/ Matheus, 1991, S. 9).

In der Datenmatrix weist keine der ordinalen und metrischen Variablen fehlende Werte auf. Bei den nominal skalierten Variablen wird bei einem fehlenden Wert die Kategorie unbekannt („U“) zusätzlich vergeben. Beispielsweise treten bei der Variable Rechtsf

(Nr. 9) fehlende Werte auf. Diese werden mit der Kategorie „unbekannt“ versehen, so dass diese Information einer Merkmalsausprägung entspricht.

3.2.3 Variablenreduktion

Die Variablenreduktion dient der Komplexitätsreduzierung (Urban, 1998, S. 94). Es sollen bereits im Vorfeld irrelevante bzw. redundante Einflussfaktoren aus der Analyse ausgeschlossen werden.

Im ersten Schritt werden die Merkmale in dieser Studie aufgrund ökonomischer und statistischer Kriterien auf Abhängigkeit von der Zielvariable überprüft. Die Zusammenhänge werden mit dem statistischen Hilfsmittel der Korrelations- bzw. der Kontingenzanalyse bestimmt. Die Korrelationsanalyse gibt Hinweise auf die Richtung und Stärke des Zusammenhangs, während die Kontingenzanalyse lediglich die Stärke des Zusammenhangs misst (Bamberg/Baur, 1998, S. 35ff.). Für intervallskalierte Variablen wird der Korrelationskoeffizient nach Bravais-Pearson, für ordinale Variablen der Rangkorrelationskoeffizient nach Spearman und für nominale Variablen der Kontingenzkoeffizient berechnet, um ein Maß für die empirische Abhängigkeit der unabhängigen Variablen zu der Zielvariablen zu erhalten. Es werden alle Variablen in die Analyse aufgenommen, die einen statistisch signifikanten Einfluss bei einer Irrtumswahrscheinlichkeit von ζ=0,05 auf die abhängige Variable haben (siehe Anhang B, S. 209).

Bei diesem Vorverarbeitungsschritt werden folgende 17 Variablen eliminiert:
Nr. 3, Nr. 16, Nr. 18, Nr. 21, Nr. 34, Nr. 35, Nr. 36, Nr. 37, Nr. 38, Nr. 39, Nr.41, Nr. 52, Nr.54, Nr.55, Nr. 56, Nr. 57, Nr. 60 (siehe Tabelle 2, S. 41ff.).

Die statistischen Kennzahlen sollten allerdings nicht die alleinige Grundlage für die Variablenreduzierung sein, vielmehr sind auch ökonomisch sinnvolle Zusammenhänge zu beachten (Urban, 1998, S. 100). Aufgrund von Expertenmeinung werden die Variablen Anl_date (Nr. 3), Best_all (Nr. 16), Best_j1 (Nr. 18) und Anz_wam6 (Nr. 57) in der Analyse belassen.

Nach der Identifikation der signifikanten Einflussgrößen folgt als nächster Schritt die Überprüfung auf Korrelation der unabhängigen Variablen untereinander. Eine übliche

Vorgehensweise identifiziert zuerst das am höchsten korrelierende Variablenpaar und eliminiert dann eine der beiden Variablen. Die Entscheidung, welche der beiden stark korrelierenden Variablen eliminiert wird, sollte unter ökonomisch sinnvollen Gesichtspunkten erfolgen, um in dieser Hinsicht sinnvolle erklärende Variablen für die Prognose zu erhalten (Urban, 1998, S. 104). Es muss allerdings beachtet werden, dass die Elimination von Variablen einen Informationsverlust bedeuten und die Güte des Modells negativ beeinflussen kann. Aus diesem Grunde kann es vernünftig sein, bestimmte Merkmale trotz einer hohen Korrelation in der Datenmatrix zu belassen, da aus ökonomischen Gründen oder Expertenmeinung eine Eliminierung einen zu großen Verlust in der Beschreibung des Kundenverhaltens bedeutet.

Bei einem Wert des Korrelationskoeffizienten $|r| \subset [0{,}8;1]$ wird bei metrischen Variablen eine der beiden Variablen eliminiert. Ein Gesamtüberblick mit den Werten des Korrelationskoeffizienten ist im Anhang C (siehe S. 210ff.) zu finden.
Folgende 16 Variablen werden eliminiert (siehe Tabelle 2, S. 41ff.): Nr. 10, Nr. 11, Nr. 13, Nr. 17, Nr. 23, Nr. 25-33, Nr. 43 und Nr. 44.

Nach den Voranalysen verbleiben 29 von 61 Variablen, davon sind sechs Variablen nominal polytom, fünf nominal binär, eine ordinal und 17 metrisch skaliert (siehe Tabelle 3).

Nr.	Variablenname	Beschreibung	Ausprägungen	Skalenniveau
Schlüsselvariablen				
1	Actionno	Schlüssel zur Identifikation der Werbeaktion	$n \subset N$	Nominal
2	Id_locat	Schlüssel zur Identifikation des Kunden	$n \subset N$	Nominal
Grunddaten				
3	Anl_date	Anlagedatum der Kundenadresse	SAS-Datumsformat	Metrisch
5	Branche2	Branche der Firma	2-stellige Codierung	Nominal
6	Br_k	Branche der Firma	1-stellige Codierung	Nominal

7	Bula	Bundesland	B: Berlin, BR: Brandenburg, BW: Baden-Württemberg, BY: Bayern, HB: Bremen, HE: Hessen, HH: Hamburg, MV: Mecklenburg-Vorpommern, NS: Niedersachsen, NW: Nordrhein-Westfalen, RP: Rheinland-Pfalz, S: Sachsen, SA: Sachsen-Anhalt, SL: Saarland, SH: Schleswig-Holstein, TH: Thüringen	Nominal
8	Ma_zahl	Mitarbeiterzahl der Firma	1-9; 10-25; 26-49; 50-99; 100-499; > 500; U	Ordinal
9	Rechtsf	Rechtsform der Firma	GmbH; GmbH & Co; GmbH & Co KG; KG; AG; SO; U	Nominal
Reaktionsdaten				
12	Anz_aabo	Anzahl aktiver Abonnements des Kunden	$n \subset N_0$	Metrisch
14	Bes_l6m	Bestellung in den letzten 6 Monaten erfolgt	0: Nein 1: Ja	Binär
15	Bes_l12m	Bestellung in den letzten 12 Monaten erfolgt	0: Nein 1: Ja	Binär
16	Best_all	Anzahl Bestellungen des Kunden insgesamt	$n \subset N_0$	Metrisch
18	Best_j	Anzahl Bestellungen im laufenden Kalenderjahr	$n \subset N_0$	Metrisch
19	Best_j1	Anzahl Bestellungen im letzten Kalenderjahr	$n \subset N_0$	Metrisch
20	Best_j2	Anzahl Bestellungen im vorletzten Kalenderjahr	$n \subset N_0$	Metrisch
22	Di_ebest	Tage seit erster Bestellung des Kunden	$n \subset N$	Metrisch
24	Di_lpos	Tage seit letzter erhaltener Rechnung des Kunden	$n \subset N$	Metrisch

45	Ums_gw	Bruttoumsatz mit Grundwerken	$r \subset R$	Metrisch
46	Ums_j	Bruttoumsatz im laufenden Kalenderjahr	$r \subset R$	Metrisch
47	Ums_j_1	Bruttoumsatz im letzten Kalenderjahr	$r \subset R$	Metrisch
48	Ums_j_2	Bruttoumsatz im vorletzten Kalenderjahr	$r \subset R$	Metrisch
49	Ums_l6m	Umsatz in den letzten 6 Monaten erfolgt	0: Nein 1: Ja	Binär
50	Ums_l12m	Umsatz in den letzten 12 Monaten erfolgt	0: Nein 1: Ja	Binär
51	Ums_oth	Bruttoumsatz außer Grundwerke und Aktualisierungslieferungen	$r \subset R$	Metrisch
Aktionsdaten				
53	Anz_7409	Anzahl Werbeaktionen, die der Kunde für Produkt 7409 erhalten hat	$n \subset N_0$	Metrisch
57	Anz_wam6	Anzahl Werbeaktionen, für die der Kunde in den letzten 6 Monaten selektiert wurde	$n \subset N_0$	Metrisch
58	Di_e_wa	Tage seit Erhalt der ersten Werbeaktion	$n \subset N$	Metrisch
59	Di_l_wa	Tage seit Erhalt der letzten Werbeaktion	$n \subset N$	Metrisch
Zielvariable				
61	Best_jn	Ist eine Bestellung erfolgt	0: Nein 1: Ja	Binär

Tabelle 3: Liste der verbleibenden Variablen

Zusammenfassend zeigt folgende Abbildung die Vorgehensweise bei der Variablenreduktion:

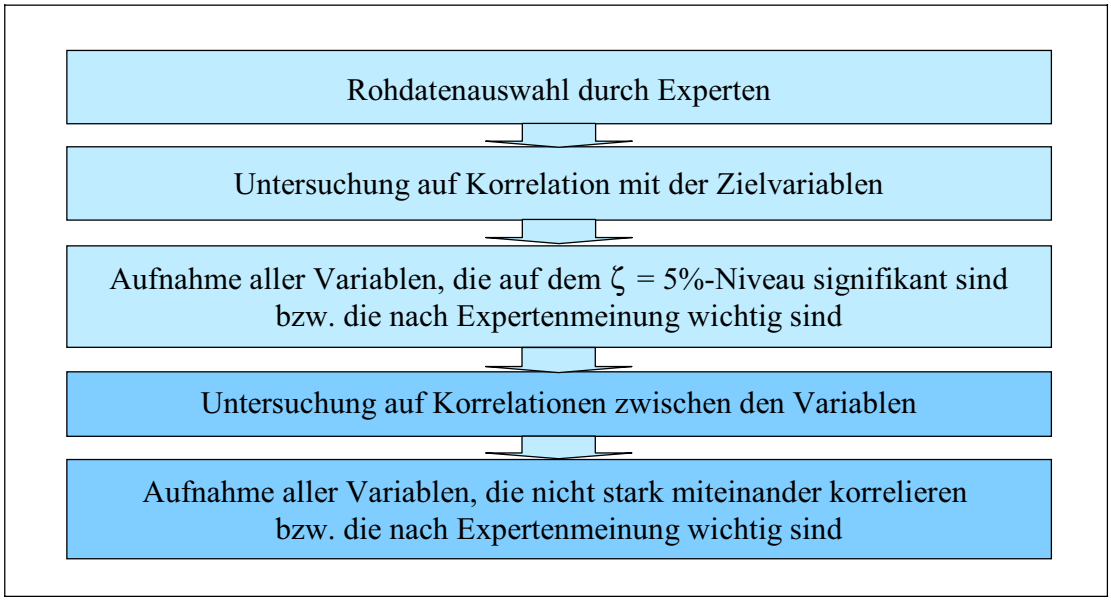


Abbildung 14: Vorgehensweise bei der Variablenreduktion
Quelle: Eigene Darstellung

3.2.4 Ausreißer-Analyse

Falls ein Objekt die Plausibilitätsprüfung im Sinne von inhaltlich möglichen, sinnvollen und realistischen Werten durch einen Experten nicht besteht, muss es eliminiert werden, um den Datenbestand sinnvoll zu bereinigen (Küsters, 2001, S. 98). Ausreißer in einzelnen Variablen können sowohl bei der Clusteranalyse als auch bei Prognoseverfahren zu Verzerrungen führen. Deshalb müssen alle Objektvektoren mit Ausreißern in den Merkmalsausprägungen genauer untersucht und gegebenenfalls eliminiert werden.

In dieser Studie werden zur Ausreißerprüfung verschiedene Lageparameter verwendet (siehe Anhang D, S. 213). Es zeigt sich, dass einige Objekte hinsichtlich ihres Verbleibs in der Datenmatrix untersucht werden müssen. Ein Objektvektor enthält bei der Variable Nr. 3 beispielsweise einen Datenfehler: „31.12.1899“. Dieses Objekt wird eliminiert. Weiterhin fallen folgende extreme reale Ausreißer auf: Objekte, die für mehr als 50 Werbeaktionen in den letzten 6 Monaten selektiert wurden (Nr. 57), mehr als 9 laufende Abonnements besitzen (Nr. 12) oder einen Umsatz von mehr als 6.000 Einheiten außerhalb des Loseblattwerk-Geschäfts getätigt haben (Nr. 51). Diese Kunden werden separat geführt und in der folgenden Analyse nicht mehr berücksichtigt. Die Ausreißer werden allerdings nur aus den Trainings- und Validierungsdaten entfernt. Diese Aufteilung der Datenmatrix wird im folgenden vorgestellt.

3.2.5 Aufteilung der Datenmatrix in Trainings-, Validierungs- und Testdaten

Die Anwendung vieler Data Mining Methoden beruht auf der Zerlegung der Gesamtmenge von Objektvektoren in drei überschneidungsfreie Teilmengen: Trainings-, Validierungs- und Testdaten (Berry/Linoff, 2000, S. 193f.). In der Gesamtdatenmatrix entspricht diese Teilung einer zeilenweisen Orientierung.
Auf die erste Gruppe von Objektvektoren, die sogenannte Trainingsdaten, werden Data Mining Modelle angewandt, Ergebnisse erzielt und festgehalten. Anschließend werden anhand der zweiten Gruppe von Objektvektoren, den sogenannte Validierungsdaten, die mit den Trainingsdaten erzielten Ergebnisse überprüft und gegebenenfalls verbessert, indem die Parameter entsprechend angepasst werden. Die dritte Gruppe von Objektvektoren, die sogenannte Testdaten, zeigen die Modellperformance für neue, noch nicht verwendete Daten. Das Ziel dabei ist, die Generalisierbarkeit des Modells zu prüfen und eine realistische zu erwartende Fehlerbeurteilung für neue Daten anzugeben. Diese Vorgehensweise wird **Holdout-Methode** genannt. Lt. Hofmann (1990, S. 949) sollte die Gesamtdatenmenge für die Holdout-Methode „hinreichend“ groß sein, das heißt mehr als 1.000 Objekte. Ist dies nicht der Fall, kann auf die Verwendung der Testdaten verzichtet und statt dessen zum Modellvergleich ebenfalls auf die Validierungsdaten zurückgegriffen werden. Allerdings kann dies lt. Bishop (1995, S. 372) zu einer Überanpassung des Modells hinsichtlich der Validierungsdaten führen.
Zum Aufteilungsverhältnis gibt es verschiedene Empfehlungen: Breiman et al. (1984, S. 11) empfehlen beispielsweise 2/3 der Daten als Trainingsdaten und 1/3 als Validierungsdaten zu verwenden. Sie verwenden dabei keine Testdaten. Berry/Linoff (2000, S. 194) schlagen eine Aufteilung in 60% Trainings-, 30% Validierungs- und 10% Testdaten vor. Murthy/Salzberg (1995b, S. 224) genügen bereits 10% der Daten als Validierungsdaten. Urban (1998, S. 89) empfiehlt aufgrund eines Literaturüberblicks, dass 75-90% der Daten als Trainings- und Validierungsdaten, dabei ein Verhältnis von Trainings- zu Validierungsdaten von 3:1, und die restlichen 10-25% der Daten als Testdaten verwendet werden sollten.

Wenn nur wenige Objektvektoren, beispielsweise weniger als 1.000, zur Verfügung stehen, ist die **n-fache Kreuzvalidierung** eine geeignete Vorgehensweise. Dabei wird die Gesamtdatenmenge in n ca. gleich große überschneidungsfreie Teilmengen geteilt. $n-1$ Teile werden zum Trainieren und der restliche Teil zum Testen des Modells ver-

wendet. Dies wiederholt sich n mal, so dass jede Teilmenge einmal zum Testen verwendet wird. Der Vorteil dabei ist, dass die Daten relativ optimal genutzt werden, da jedes Objekt sowohl in den Trainings- als auch in den Testdaten verwendet wird. Im Unterschied zur Holdout-Methode werden hier keine Validierungsdaten verwendet. Bei dieser Vorgehensweise kann aufgrund der Erstellung von n Modellen eine durchschnittliche Fehlerbeurteilung angegeben werden, die wiederum eine realistische Schätzung der echten Fehlerbeurteilung liefert. Allerdings kann das Ergebnis nicht direkt erklärt werden, da sich die Bewertung eines einzelnen Objektes beispielsweise als Durchschnittswert der Bewertung der n Einzelmodelle berechnet. In der Praxis wird häufig n=10 gewählt (Kohavi, 1995, S. 75).

Verwendet man **alle Daten**, um die Modellparameter zu bestimmen, so tritt häufig das Problem einer Überanpassung oder Overfitting auf, das heißt die Daten werden ohne Kontrolle „zu genau“ abgebildet (Witten/Frank, 2001, S. 129). Dies bedeutet, dass bei der Verwendung aller Daten naturgemäß eine zu optimistische Fehlerbeurteilung, beispielsweise die Anzahl korrekt klassifizierter Objekte im Vergleich zu der zu erwartenden Fehlerbeurteilung bei unbekannten Daten, ausgegeben wird.

In dieser Arbeit wird die Holdout-Methode angewandt, da genügend Daten zur Verfügung stehen. Dabei wird die Gesamtdatenmatrix in Anlehnung an Urban (1998, S. 89) zuerst in eine Trainings- und Validierungsdatenmenge (85%) und eine Testdatenmenge (15%) aufgeteilt. Das Verhältnis von Trainings- zu Validierungsdaten beträgt 4:1 (siehe Tabelle 4). Die Zielvariable wird jeweils als Schichtungsvariable verwendet, um das gleiche Verhältnis in allen drei Teilmengen zu gewährleisten.

Anteil		Art
85	80	Trainingsdaten
	20	Validierungsdaten
15		Testdaten

Tabelle 4: Aufteilung der Daten bei der Holdout-Methode

Die Ausreißer werden in dieser Studie nur aus den Trainings- und Validierungsdaten entfernt. Die Testdaten sollen möglichst „echt“ im Sinne von „unverändert“ bleiben, um die Modellgüte zum einen angemessen beurteilen zu können und zum anderen würde

das Scoring-Modell bei der Anwendung und Implementierung auf der Datenbank ebenfalls mit diesen Originaldaten arbeiten. Es werden 66 Objekte eliminiert, davon stammen 2 Objekte aus der 1-Klasse. Somit ergibt sich folgende Aufteilung (siehe Tabelle 5):

	Trainings- und Validierungsdaten	Testdaten
Besteller	333	59
Nichtbesteller	77.864	13.752
Summe	78.197	13.811

Tabelle 5: Übersicht Trainings-, Validierungs- und Testdaten in dieser Studie

Zur Bestätigung, dass ausreichend Testdaten vorhanden sind, kann beispielsweise Schaller (1997, S. 583) herangezogen werden, der mindestens „40 - 50 Besteller“ empfiehlt. Genauer befasst sich Knauff (1991, S. 583) mit der Problematik und legt folgende Formel zur Berechnung zu Grunde:

$$N = R(100-R) * S^2 / F^2$$

mit N = Anzahl der zu testenden Adressen,
R = Anzahl Reagierer (in %),
S = Sicherheitsgrad; ausgehend von einer Normalverteilung, wird das 1-ζ/2 - Fraktil verwendet, bei ζ=5% beträgt S=1,96,
F = Fehlertoleranz, wird subjektiv vorgegeben.

Wenn ζ=5% und F=10%, beträgt der Stichprobenumfang N bei einer Responsequote von 0,4%:
$N = 0,4(100-0,4) * 1,96^2 / 0,1^2$=15.305, also 15.305 Adressen mit ca. 61 Reagierern.

Diese Zahlen werden mit 13.752 Nichbestellern und 59 Bestellern in etwa erreicht, so dass insgesamt davon ausgegangen werden kann, dass in der vorgelegten Studie ausreichend Testdaten vorhanden sind, um die Modellgüte angemessen beurteilen zu können.

3.3 Bewältigung niedriger Responsequoten

Niedrige Responsequoten bedeuten, dass die 1-Klasse sehr gering und die 0-Klasse sehr groß ist. Die Problematik bei stark ungleich verteilter Anzahl von 1-Klasse und 0-Klasse besteht darin, dass einige Modelle nicht mehr in der Lage sind, zwischen 1-Klasse und 0-Klasse zu unterscheiden (Milley/Seabolt/Williams, 1998, S. 14; siehe Kapitel 1.2, S. 10). Würden alle Objekte als Nichtbesteller eingestuft, ergäbe dies bei einer Responsequote von 0,4% eine korrekte Zuordnung von 99,6% aller Objekte. Dies zeigt, weshalb Modelle bei einem derart extremen Verhältnis von 1-Klasse zu 0-Klasse dazu neigen, alle Objekte als 0-Klasse einzustufen. Kubat/Holte/Matwin (1997) führten einen Versuch durch, beginnend mit 50 Objekten in der 1-Klasse und 50 Objekten in der 0-Klasse, wobei die Objekte in der 0-Klasse von 50 mit einer Schrittweite von 50 auf 800 steigen. Die Modellgüte wird an Testdaten mit derselben Verteilung wie bei den entsprechenden Trainingsdaten getestet. Das Ergebnis zeigt, dass bei steigender Zahl von Objekten in der 0-Klasse mehr als 90% dieser Objekte richtig, die wenigen Objekte der 1-Klasse jedoch zunehmend falsch klassifiziert werden.

Ein Überblick über empirische Untersuchungen anderer Autoren im Falle eines geringen Anteils der 1-Klasse an den Gesamtdaten zeigt Tabelle 6.

Literaturverweis	Anzahl Gesamtdaten	Anzahl Objekte der 1-Klasse	Anteil der 1-Klasse in den Trainingsdaten
Arndt/Gersten/Wirth, 2001	ca. 90.000	5.350	50%
Baetge/Uthoff, 1998	13.356	413	50%
Bonne/Arminger, 2001	ca. 25.000	6.781	27%
Buja/Lee, 2001	768	268	50%
Enache, 1998, S.74	38.176	6.401	50%
Fawcett/Provost, 1996	5.000	1.000	20%
Guo/Murphey, 2001	863	136	16%
Ittner/Sieber/Trautzsch, 2001	20.000	371	1,9%
Musiol, 1996	75.434	4.979	6,6%
Musiol/Steinkamp, 2001	ca. 200.000	4.134	2%
Nauta/Matkovski, 1999, S. 8	2.107	153	7%

Säuberlich, 2000b	ca. 22.000	1.600	7%
Weingärtner, 2001	13.368	894	6,7%
Weiss/Indurkhya,1998, S.156f	2.079	< 207	< 10%
Weiss/Indurkhya, 1998, S.156f	7.133	< 142	< 2%
Wittmann/Ruhland, 2001	186.152	656	2,3%

Tabelle 6: Überblick empirischer Untersuchungen mit niedrigem Anteil der 1-Klasse

Diese Aufstellung verdeutlicht, dass in empirischen Untersuchungen deutliche Unterschiede sowohl bei der Größe der Datenmenge als auch bei dem Anteil der 1-Klasse in den Trainingsdaten herrschen.
Einige Autoren arbeiten mit dem ursprünglichen Anteil der 1-Klasse, beispielsweise Ittner/Sieber/Trautzsch (2001) oder Weingärtner (2001), andere ändern deren Anteil in den Trainingsdaten bis hin zu einem Anteil von 50%, wie beispielsweise Arndt/Gersten/Wirth (2001) oder Baetge/Uthoff (1998) (siehe Tabelle 6).
Allgemein gesehen ist es umso besser, je mehr Daten zur Verfügung stehen (Turney, 1995, S. 31). Garbe et al. (1995, S. 2487) kommen bei Ihren Versuchen allerdings zu dem Ergebnis, dass ab einem Gesamtdatenumfang von ca. 2.000 Objekten relativ ähnliche Muster entdeckt werden. Krahl/Windheuser/Zick (1998, S. 154) sprechen von einer Untergrenze von 1.000 Objekten. In dem Literaturüberblick zeigt sich, dass der Großteil mit deutlich größeren Datenvolumen arbeitet. Es gibt jedoch auch Studien, die mit weniger als 1.000 Objekten bei den Trainingsdaten durchgeführt wurden, beispielsweise Baetge/Uthoff (1998), Buja/Lee (2001) oder Guo/Murphey (2001) (siehe Tabelle 6).

In der hier verwendeten Datenmatrix ist eine vergleichsweise große Zahl an Objekten vorhanden, allerdings ist der Anteil der 1-Klasse sowohl prozentual gesehen mit 0,43% als auch absolut gesehen mit 393 Bestellern sehr gering.

Deshalb werden in dieser Arbeit verschiedene Vorgehensweisen bei der Analyse im Fall niedriger Responsequoten untersucht.

Die einfachste Möglichkeit ist, **alle Trainingsdaten** mit dem Originalverhältnis von 1-Klasse zu 0-Klasse zur Analyse zu verwenden. Weiss/Indurkhya (1998, S. 171) behaupten, dass das Lernen mit den Gesamtdaten normalerweise eine nahezu optimale Lösung liefern sollte.

Alternativ dazu fordern Berry/Linoff (2000, S. 53), dass die 1-Klasse mindestens 15-30% der Gesamtdatenmenge entsprechen sollten. Anderson (1972, S. 34) behauptet, dass bei gleichem Anteil von 1-Klasse und 0-Klasse in der Zielvariable bessere Ergebnisse erzielt werden. Auch Krahl/Windheuser/Zick (1998, S. 155) geben zu bedenken, dass eine kleine Klasse bei heuristischen Verfahren häufig nicht beachtet wird.
Zwei Möglichkeiten, den Anteil der 1-Klasse zu erhöhen, werden im folgenden vorgestellt.

Eine häufig angewandte Möglichkeit ist das **Downsizing** (auch One-Side-Sampling oder Oversampling), das heißt es werden alle Objekte der 1-Klasse verwendet, jedoch nur ein Teil der 0-Klasse (Berry/Linoff, 2000, S. 197). Die Auswahl der Objekte aus der 0-Klasse erfolgt beispielsweise mittels einer Stichprobenziehung. Das Verhältnis von 1-Klasse zu 0-Klasse kann dann über die Anzahl der auszuwählenden Objekte aus der 0-Klasse gesteuert werden. Somit wird ein gleicher Anteil von 1-Klasse und 0-Klasse in der Zielvariable erreicht, indem ebenso viele Objekte aus der 0-Klasse wie vorhandene Objekte aus der 1-Klasse ausgewählt werden. Kubat/Matwin (1997, S. 185) und Provost/Oates/Jensen (1999, S. 31) empfehlen beispielsweise bei einem extremen Verhältnis von 1-Klasse zu 0-Klasse ein Downsizing durchzuführen. Zur Auswahl der Objekte aus der 0-Klasse wird in dieser Arbeit eine Zufallsstichprobe verwendet. Es werden folgende Verhältnisse von 1-Klasse zu 0-Klasse untersucht: 1:1, 1:5 und 1:10. Dabei werden zu jedem Verhältnis drei Stichproben aus der 0-Klasse gezogen, um die Unterschiede, die aus der Stichprobenziehung resultieren können, aufzuzeigen.

Das **Duplizieren** der Objekte aus der 1-Klasse stellt eine weitere Möglichkeit dar, den Anteil der 1-Klasse zu erhöhen. Es müssen dadurch im Vergleich zum Downsizing keine Objekte weggelassen werden, allerdings nimmt die Anzahl der Objekte insgesamt stark zu. Downsizing und Duplizieren haben gemäß Ohno-Machado (1996a, S. 74) einen ähnlichen Effekt, allerdings gehen beim Downsizing Informationen verloren. In dieser Arbeit werden beim Duplizieren alle Objekte der 1-Klasse so lange an die Datenmatrix angehängt, bis deren Anzahl gerade noch kleiner der Zahl der Objekte der 0-Klasse ist. Insgesamt umfasst diese Datenmenge 155.453 Objekte, davon 77.589 Objekte der 1-Klasse und 77.864 Objekte der 0-Klasse.

Eine weitere Möglichkeit stellt die Verwendung einer **Profitmatrix** dar. Dabei wird die komplette Datenmatrix für jedes Auswertungsverfahren verwendet. Allerdings wird jeder möglichen Zuordnung von realer zu vorhergesagter Ausprägung der Zielvariablen ein vom Anwender vorzugebender Profit zugewiesen. In der Datenmatrix gibt es zu jedem Objektvektor die bekannte Zielvariablenausprägung und eine durch das Auswertungsverfahren prognostizierte (siehe Tabelle 7).

Kunde	alle Merkmale	reale Zielvariable	prognostizierte Zielvariable des Verfahrens
A	...	1	0
B	...	0	0
C	...	1	1
...	...	...	...

Tabelle 7: Veranschaulichung reale bzw. modellbasierte Zuordnung

Die Wahl der Werte der Profitmatrix hat direkten Einfluss auf die Ergebnisse und sollte sich an den tatsächlichen Kosten bzw. Deckungsbeiträgen orientieren (Bonne, 1999, S. 42f.). Jeder Zuordnung von realer zu vorhergesagter Ausprägung der Zielvariablen werden Kosten bzw. Erträge (siehe Tabelle 8) zugewiesen. Erfolgt eine korrekte Zuordnung der 1-Klasse, ergibt dies einen Ertrag von 248 Einheiten. Dies entspricht dem Verkaufspreis des hier beworbenen Produkts. Bei einer korrekten Zuordnung der 0-Klasse wird ein Ertrag von einer Einheit erreicht, da die Kosten eines Direct Mails eingespart werden können. Wird die 0-Klasse modellbasiert der 1-Klasse zugeordnet, fallen Kosten in Höhe einer Geldeinheit an, da dieser Kunde erfolglos angeschrieben wird. Falls eine 1-Klasse modellbasiert der 0-Klasse zugeordnet wird, fallen Opportunitätskosten in Höhe von 248 Einheiten an, da ein potenzieller Besteller nicht angeschrieben wurde.

		modellbasierte Zuordnung	
		1-Klasse	0-Klasse
Reale Zuordnung	1-Klasse	248	-248
	0-Klasse	-1	1

Tabelle 8: Beispiel einer Profitmatrix

Die Auswirkung der Profitmatrix auf die einzelnen Verfahren wird im jeweiligen Kapitel beschrieben.

3.3.1 Stichprobenplanung

Mit Hilfe der Stichprobenplanung soll der Anteil der 0-Klasse an den Gesamtdaten reduziert werden. Die verschiedenen Auswahlverfahren werden in nachfolgender Abbildung veranschaulicht:

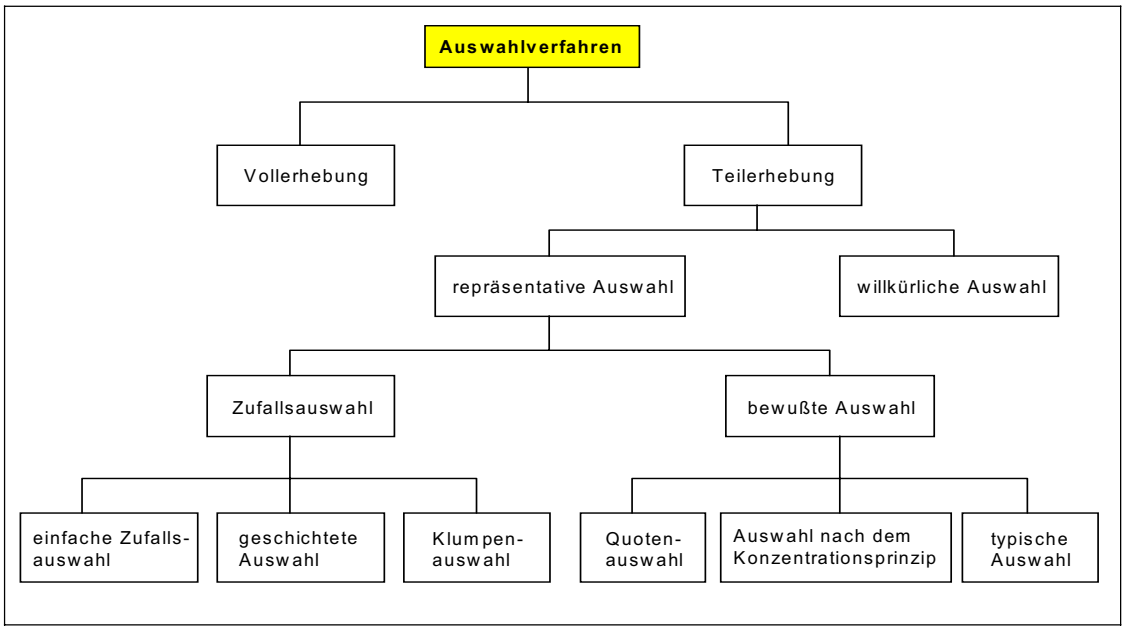


Abbildung 6: Auswahlverfahren
Quelle: Berekoven/Eckert/Ellenrieder (2001, S. 62)

Die Verfahren der Zufallsauswahl basieren auf Zufallsmechanismen, das heißt jede Einheit hat die gleiche Chance, in die Stichprobe zu gelangen. Je mehr Einheiten ausgewählt werden, desto größer ist die Wahrscheinlichkeit, dass die Stichprobe in ihrer Zusammensetzung der Grundgesamtheit entspricht.
Bei der bewussten Auswahl hingegen wird gezielt nach bestimmten Merkmalen ausgewählt, jedoch auch mit dem Ziel, eine repräsentative Auswahl zu treffen.
Weiterführende Stichprobenverfahren werden bei Reinartz (1999, S. 85ff.) untersucht und diskutiert. Ein kurzer Überblick wird auch bei Hippner/Wilde (2001a, S. 48f.) gegeben.

Beim Downsizing wird in dieser Arbeit eine einfache Zufallsauswahl aus der 0-Klasse vorgenommen. Gerade dann, wenn wenige Objekte aus der 0-Klasse gezogen werden sollen, kann dies problematisch sein. Eine in dieser Studie untersuchte Alternative ist die clusteranalysegestützte Stichprobenziehung.

3.3.2 Clusteranalytische Verfahren zur Unterstützung der Stichprobenziehung

Um den gleichen Anteil von 1-Klasse zu 0-Klasse bei der Zielvariable zu erreichen, wird beim Downsizing eine Zufallsauswahl von 333 Objekten (siehe Tabelle 5, S. 55) vorgenommen. Dabei werden aus einer sehr großen Zahl von Nichtbestellern sehr wenige Objekte ausgewählt. Eine Stichprobe von 333 aus knapp 77.864 kann bei wiederholter Ausführung stark unterschiedliche Ergebnisse liefern. Lewis/Catlett (1994, S. 148) zeigen, dass eine zufällige Stichprobenziehung ineffizient sein kann, wenn eine Klasse sehr selten auftritt. Im folgenden wird ein neu entwickelter Ansatz zur Stabilisierung der Stichprobenziehung vorgestellt. Dieser basiert auf einer vorgeschalteten Clusteranalyse.

„Die Clusteranalyse ist ein Instrumentarium zum Erkennen von Strukturen einer Menge von Objekten“ (Hartung/Elpelt, 1995, S. 443). Das Ziel der Clusteranalyse ist die Zusammenfassung von Objekten zu Klassen, so dass zwischen den Objekten derselben Klasse größtmögliche Ähnlichkeit und zwischen den Objekten verschiedener Klassen größtmögliche Verschiedenheit erreicht wird (Opitz, 1980, S. 65).
Zuerst wird der Klassifikationstyp festgelegt, das heißt, ob ein Objekt genau, mindestens oder höchstens einer Klasse zugeordnet werden darf. In dieser Arbeit wird eine exhaustive, disjunkte Klassifikation durchgeführt, das heißt jedes Objekt wird genau einer Klasse zugeordnet.

Einen Überblick über einige Verfahren der Clusteranalyse zeigt die nachfolgende Abbildung:

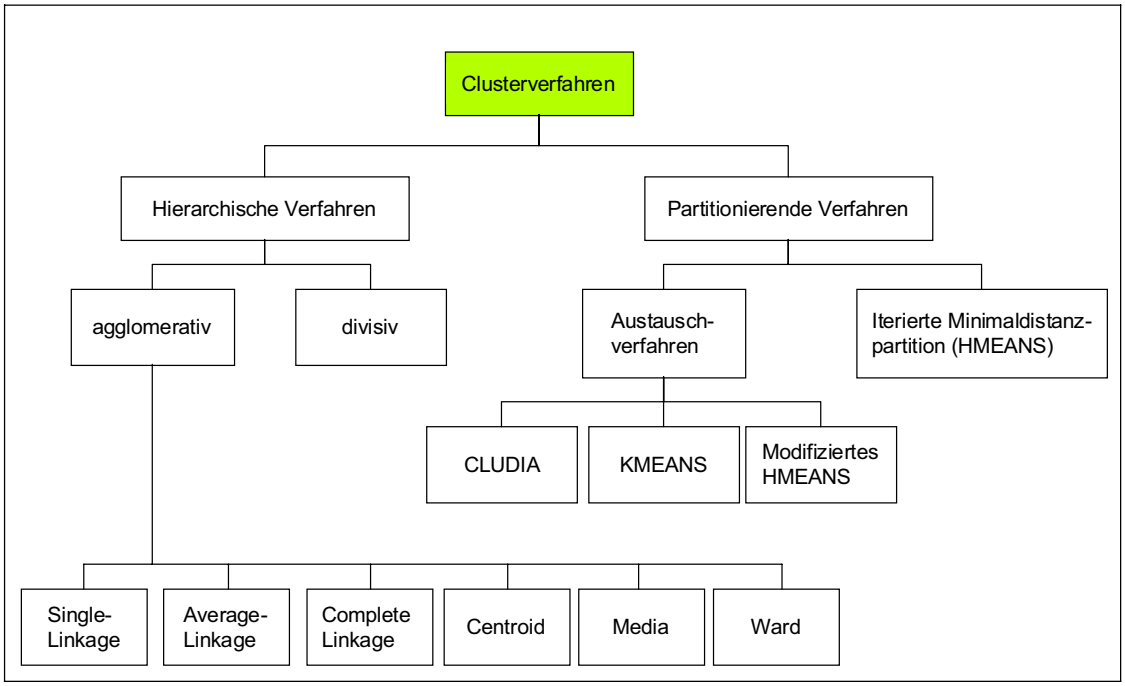


Abbildung 15: Überblick Clusterverfahren
Quelle: In Anlehnung an Backhaus et al. (2000, S. 281)

Divisive Verfahren gehen von einer Objektklasse aus und teilen diese schrittweise in Gruppen auf. Der Ausgangspunkt der **agglomerativen Verfahren** ist eine Anfangszerlegung in einelementige Objektklassen. Ziel ist die schrittweise Zusammenfassung von ähnlichen Objekten zu immer größeren Klassen. Die Grundlage hierfür ist ein Verschiedenheitsindex für Klassenpaare (Opitz, 1980, S. 98).
Wird eine **Zerlegung** der Objektmenge angestrebt, dann verwendet man Austauschverfahren, die trotz Vorgabe der Klassenzahl oft eine „gute“ Zuordnung finden. Sie basieren auf dem Austausch von Objekten bei einer vorgegebenen Startpartition. Diese kann beispielsweise durch ein hierarchisches Verfahren, eine geeignete Heuristik oder eine Zufallsauswahl gegeben sein. Es wird nun für alle Objekte geprüft, ob sie je nach Kriterium in eine andere Klasse besser passen. Gegebenenfalls wird die beste Neuzuordnung vorgenommen (Opitz, 1980, S. 87f.).

Die hierarchischen Verfahren sind bei sehr großen Datenmengen ungeeignet, da sie zum einen sehr speicherintensiv und zum anderen sehr rechenintensiv sind. Sobald die Analyse mehr als ca. 1.000 Datensätze umfasst, ist die Verwendung der hierarchischen Verfahren nicht mehr empfehlenswert, da das Ausmaß an Berechnungen mit zunehmender Anzahl an Objekten überproportional steigt (Brosius, 1998, S. 721).

Deshalb sind gerade bei Data Mining Anwendungen schnelle Algorithmen, wie beispielsweise der in dieser Studie verwendete k-means Algorithmus, zweckmäßig. Das k-means Verfahren, bei dem k für die Anzahl der vorgegebenen Klassen steht, arbeitet mit dem Varianzkriterium und setzt deshalb metrisches Merkmalsniveau voraus. Da es sich um eine Heuristik handelt, muss die Verwendung von nominalen bzw. ordinalen Daten jedoch nicht grundsätzlich ausgeschlossen werden (Bausch/Opitz, 1993, S. 58). Die Schnelligkeit begründet sich vor allem aus der effizienten iterativen Berechnung des Klassenaustauschs. Während bei agglomerativen Algorithmen bei jeder Iteration eine neue Distanzmatrix gebildet werden muss, wird bei k-means immer nur nach dem Varianzkriterium die Klassenzugehörigkeit von einzelnen Objekten überprüft. Der Nachteil der partitionierenden Verfahren liegt in der vorzugebenden Klassenzahl, während bei hierarchischen Verfahren aufgrund der Transparenz des Verfahrens anhand des Dendrogramms oder des Ellbogenkriteriums eine sinnvolle Anzahl bestimmt werden kann. Grundlage für Klassifikationsverfahren als Teil der multivariaten Datenanalyse ist die Ermittlung einer aggregierten Distanzmatrix auf Objektpaaren (Bausch/Opitz, 1993, S. 43).

In dieser Studie orientiert sich die weitere Beschreibung des Ablaufs an den im SAS EM implementierten Bausteinen.

Zuerst wird eine Distanzmatrix unter Verwendung der euklidischen Distanz gebildet. Im Fall nominaler bzw. ordinaler Daten werden diese gemäß dem Verfahren von Ralambondrainy (1995) in binäre Variablen umgewandelt, die dann als metrische Variablen behandelt werden. Der Nachteil dieser Umwandlung ist die stark steigende Anzahl an Variablen, was zu einer Erhöhung der Rechenzeit führt. Allerdings lassen sich auf diese Weise auch wichtige nicht-metrische Variablen in die Analyse mit einbeziehen. Da zwischen den Objekten und den Clusterzentren im folgenden Abstände berechnet werden müssen, empfiehlt es sich, eine Standardisierung der Variablen durchzuführen (Chamoni/Budde, 1997, S. 23).

Im nächsten Schritt wird eine Startpartition gebildet, der gemäß einer Studie von Milligan (1980, S. 339) große Bedeutung im Hinblick auf die Güte und Schnelligkeit des Algorithmus zukommt.

MacQueen's (1967) k-means Algorithmus sieht vor, die ersten k Datensätze als Klassenzentren zu verwenden, während Anderberg (1973, S. 159ff.) weitere Möglichkeiten, wie beispielsweise eine Zufallsauswahl oder vom Analysten vorab bestimmte Klassenzentren, empfiehlt. Opitz (1980, S. 93) schlägt beispielsweise vor, ein Objekt zufällig herauszugreifen und als nächstes das am weitesten entfernte als neues Klassenzentrum zu wählen. Nun wird das Objekt als zusätzliches Klassenzentrum bestimmt, das am weitesten von den beiden bisher bestehenden entfernt ist. Dies wird iterativ fortgeführt, bis die Anzahl k der vorher festgelegten Klassen erreicht ist.

Im SAS EM läuft ein mehrstufiges Verfahren zur Bestimmung der k Klassenzentren ab, das sich an Hartigan (1975, S. 74-78) und Tou/Gonzalez (1974, S. 90-92) orientiert:

1) Der erste vollständige Objektvektor wird als erstes Klassenzentrum z_1 gewählt.
2) Der nächste vollständige Objektvektor n_j, der nachfolgende Bedingung erfüllt, wird zu einem zweiten Klassenzentrum bestimmt, so dass die Zahl der Klassenzentren s=2.

$$d(n_j,z_1) > r$$

mit $d(n_j,z_1)$ = Distanz zweier Objekte,
n_j = neues Objekt j, mit j=1,...,n,
r = kritische Schranke für die Distanz (vorgegeben).

Falls die Schranke r zu groß gewählt wird, kann es vorkommen, dass kein zweites Klassenzentrum entsteht. In diesem Fall muss r verringert werden, bis ein zweites Klassenzentrum gefunden wird.

3) Falls k=2 folgt Schritt 5, sonst wird der nächste vollständige Objektvektor n_j, der nachfolgende Bedingung erfüllt, zu einem weiteren Klassenzentrum bestimmt.

$$d(n_j,z_i) > r, \quad \text{für } i = 1,..., s$$

mit $s \subset$ ″2,...,k€,
z_i = Klassenzentrum i.

Falls $d(z_i, n_j) < r$ wird Schritt 4 geprüft.

4) Ist ein Objekt gemäß Schritt 3) nicht als Klassenzentrum geeignet, so werden zwei Tests unternommen, ob es ein Klassenzentrum besser ersetzen kann.

a) Ein bestehendes Klassenzentrum wird ersetzt, wenn die Distanz zwischen dem neuen Objekt und dem nächstliegenden Klassenzentrum größer ist, als die kleinste Distanz zwischen den vorhandenen Klassenzentren. Dasjenige Klassenzentrum, von den beiden sich am nächsten liegenden, wird ersetzt, das die kürzeste Entfernung zu einem der restlichen Klassenzentren aufweist:

$$z_u \text{ mit } \min_i d(n_j, z_i) \mid d(n_j, z_u)$$

$$a \mid \min_{i,l;i\neq l} d(z_i, z_l)$$

falls $d(n_j,z_u) > a$ dann:

$$z_v \text{ mit } \min_i d(z_u, z_i) \mid d(z_u, z_v)$$

$$b \mid \min_{i,u,;i\neq u} d(z_u, z_i)$$

$$c \mid \min_{i,v;i\neq u} d(z_v, z_i)$$

falls $b > c$, dann streiche z_v, falls $c > b$, dann streiche z_u und wähle n_i als neues Klassenzentrum.

Ein Beispiel zeigt Abbildung 16:

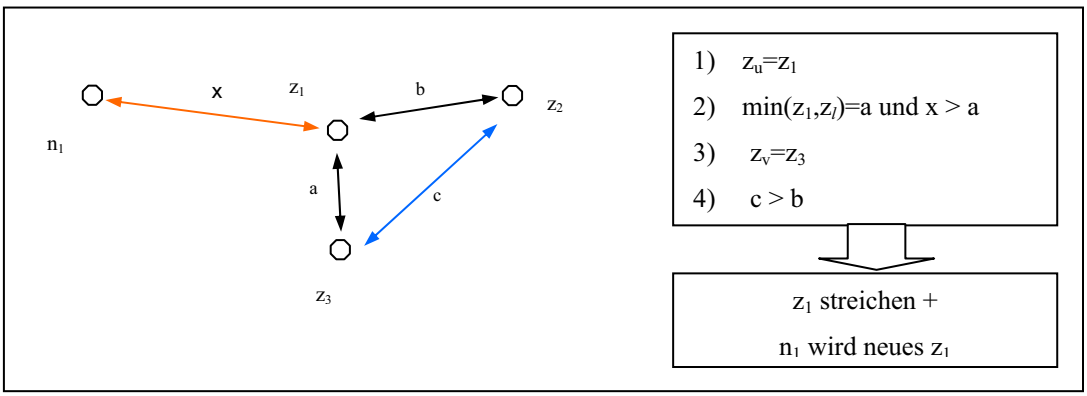


Abbildung 16: Test 1 bei Bildung der Startpartition
Quelle: Eigene Darstellung

b) Falls a) nicht zutrifft, ersetzt das neue Objekt das nächstliegende Klassenzentrum, falls folgendes zutrifft: Die kleinste Distanz von dem neuen Objekt zu allen anderen Klassenzentren ist größer als die kürzeste Distanz von dem betroffenem Klassenzentrum zu allen anderen Klassenzentren.

z_u mit $\min_i d(n_j, z_i) \mid d(n_j, z_u)$

a | $\min_{i,i\neq u} d(n_j, z_i)$

b | $\min_{i,i\neq u} d(z_u, z_i)$

falls a > b, dann streiche z_u und wähle n_i als neues Klassenzentrum.

Ein Beispiel zeigt Abbildung 17:

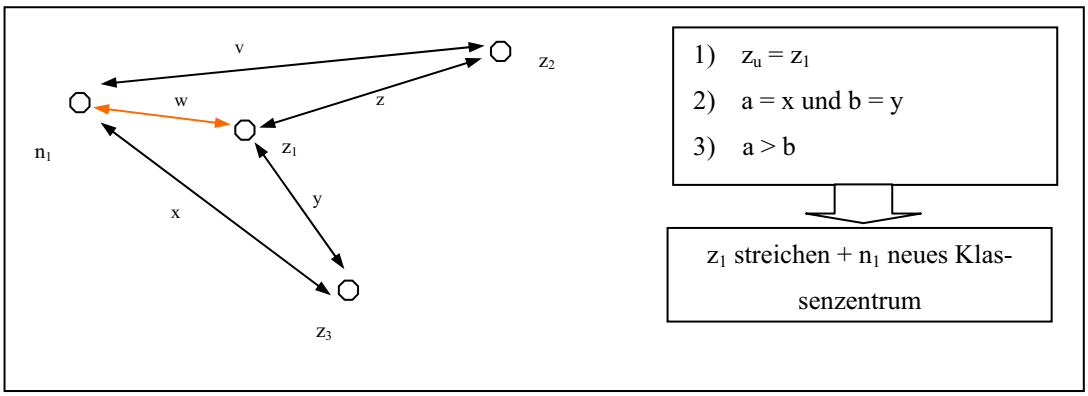


Abbildung 17: Test 2 bei Bildung der Startpartition
Quelle: Eigene Darstellung

5) Dabei entstehen s Klassenzentren mit s ⊂ ″2,...,k∈ Die Anzahl der möglichen Klassenzentren k ist abhängig von der vorgegebenen kritischen Schranke r, also k(r). Wird ein r zu groß gewählt, kann es vorkommen, dass weniger als k Klassenzentren entstehen. In diesem Fall sollte r weiter verringert werden, bis k Klassenzentren gefunden werden.

Für s=k: Bestimmung der Klassenzentren ist abgeschlossen,
für s < k und $n_j < n_n$: Schritt 3,
für s < k und $n_j = n_n$: Verringerung von r, Schritt 2.

Nach diesem Durchlauf sind die k Klassenzentren gefunden. Die einzelnen Schritte des k-means Algorithmus werden im folgenden dargestellt:

1) Klassenzentren: Die eben gebildeten Klassenzentren bilden die Grundlage.
2) Klassenvervollständigung: Es werden temporäre Cluster gebildet, indem jedes Objekt einmal dem am nächsten gelegenen Klassenzentrum zugeordnet wird. Bei jeder Zuordnung wird das Klassenzentrum als Mittelwert des neuen Clusters aktualisiert. Dies wird häufig auch als inkrementelles, online oder adaptives Training bezeichnet.
3) Austausch: Es werden wiederum Cluster gebildet, indem jedes Objekt dem am nächsten gelegenen Klassenzentrum zugeordnet wird. Nachdem alle Objekte zugeordnet wurden, werden die Klassenzentren durch Bildung des Mittelwerts neu berechnet. Dieser Schritt wird so lange wiederholt, bis die Maximalzahl an Austauschiterationen oder das Abbruchkriterium in Form einer geforderten Verbesserung der Güte erreicht ist.

Je öfter Austauschiterationen durchlaufen werden und je kleiner die geforderte Verbesserung der Güte als Abbruchkriterium gewählt wird, desto besser wird tendenziell die Güte der Klassifikation. Die Güte der Klassifikation wird nach jedem Schritt v dabei durch die kumulierten Innerklassenvarianzen der einzelnen Cluster $K \subset K^v$ bestimmt (Bausch/ Opitz, 1993, S. 58).

Das Ergebnis einer k-means Clusteranalyse ist stark abhängig von der Anzahl k vorgegebener Klassen. Die Güte einer Clusterlösung kann durch R^2 bestimmt werden:

$$R^2 \mid 14 \frac{IKV}{GV},$$

mit IKV = Varianz der Objekte innerhalb einer Klasse, aufsummiert über alle Klassen,
GV = Gesamtvarianz aller Objekte.

Zur Maximierung von R^2 muss also die Innerklassenvarianz minimiert werden, da die Gesamtvarianz konstant ist.

Ein Hilfsmittel zur Wahl der Klassenzahl ist das im SAS EM implementierte **Cubic Clustering Criterion** (CCC) von Sarle (1983), das im folgenden vereinfacht vorgestellt wird.
Bei p Merkmalen geht das CCC davon aus, dass im ungünstigsten Fall die Objektvektoren gleichverteilt ein p-dimensionales Rechteck mit den Seitenlängen $s_1, s_2, ..., s_p$ bilden.

Im Fall der Annahme von k Clustern ist das p-dimensionale Rechteck geeignet in k p-dimensionale Quadrate mit Seitenlänge c, die den Clustern entsprechen, zu zerlegen.

Angenommen, die Kanten des p-dimensionalen Rechtecks sind an den Koordinatenachsen ausgerichtet und s_j ist die Länge einer Rechteckkante entlang der j-ten Dimension, dann berechnet sich das Volumen v des p-dimensionalen Rechtecks nach:

$$v = \prod_{j=1}^{p} s_j .$$

Im Fall einer exhaustiven Klassifikation muss das Rechteckvolumen gleich der Summe der Quadratvolumen sein. Teilt man das p-dimensionale Rechteck in k p-dimensionale Quadrate mit Kantenlänge c, dann entspricht das Volumen des p-dimensionalen Rechtecks der Summe aller p-dimensionalen Quadrate: $v = k \cdot c^p$.
Daraus folgt:

$$c = \left(\frac{v}{k} \right)^{1/p} .$$

Im folgenden wird u_j als die Zahl der p-dimensionalen Quadrate entlang der j-ten Dimension des p-dimensionalen Rechtecks definiert:

$$u_j = \frac{s_j}{c} .$$

Ferner muss s_j für alle j=1,...,p ein ganzzahliges Vielfaches von c sein, das heißt es existiert

$u_1,...,u_p \subset K$ mit $c \cdot u_j = s_j$.

Da s_j der Spannweite des Merkmals j entspricht, kann diese gegebenenfalls so erweitert werden, dass u_j ganzzahlig ist.

Abbildung 18 zeigt zur Veranschaulichung eine beispielhafte Darstellung eines 2-dimensionalen Rechtecks mit k=6 Clustern.

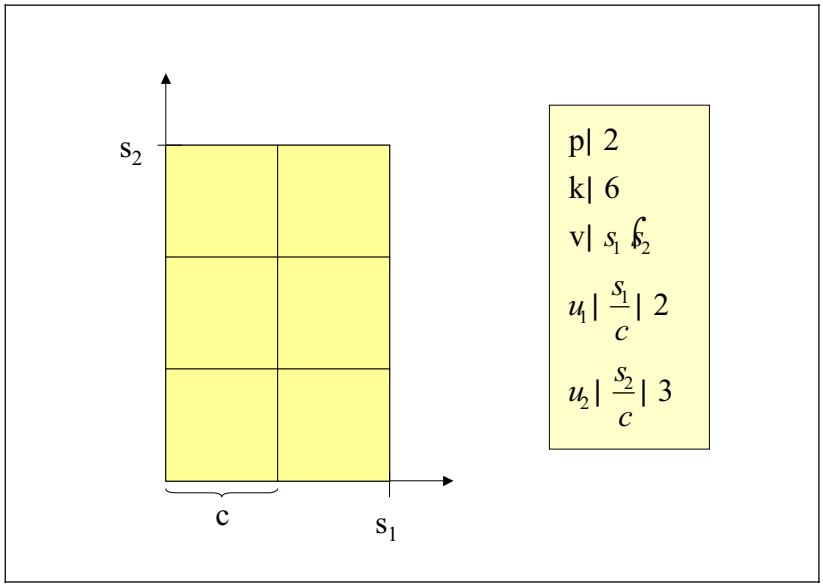


Abbildung 18: Grafische Darstellung des CCC
Quelle: Eigene Darstellung

Verwendet man statt IKV $\frac{}{p}c^2$ und statt GV $\frac{}{p}s_j^2$, kann anstatt R^2 ein $\tilde{R}^2$, das in ähnlicher Form bei der Berechnung des CCC benutzt wird, folgendermaßen berechnet werden:

$$\tilde{R}^2 = 1 - \frac{\frac{}{p}c^2}{\frac{}{p}s_j^2} \qquad \subset \Psi, 1\beta.$$

Sarle (1983, S. 8) kommt auf dieser Basis durch Simulationen zu einer heuristischen Approximation für $E/\tilde{R}^2($, aus welchem sich das CCC folgendermaßen berechnen lässt:

$$CCC - f_1\left(\ln\left(\frac{1 - E(\tilde{R}^2)}{1 - \tilde{R}^2}\right), f_2/n, E(\tilde{R}^2)0.\right.$$

Die Funktion f_2 ist immer > 0, somit ist

CCC > 0, wenn $\tilde{R}^2 > E/\tilde{R}^2($,

CCC < 0, wenn $E/\tilde{R}^2(> \tilde{R}^2$.

Die Formel zum CCC ist empirisch hergeleitet worden als Versuch die Varianz bei verschieden großen Zahlen von Beobachtungen, Variablen und Cluster zu stabilisieren.

Positive Werte des CCC bedeuten, dass das erhaltene $\widetilde{R}^2$ größer ist als $E(\widetilde{R}^2)$ und zeigen somit ein mögliches Vorhandensein von Clustern an.

Sarle (1983, S. 9) empfiehlt, das CCC gegen die Zahl der Cluster zu plotten. Maxima des CCC-Plots, die Werte von 2 oder mehr erreichen, sind ein Indikator für eine gute Klassifikation. Werden nur Werte zwischen 0 und 2 erreicht, so ist das ein Hinweis auf ein mögliches Klassifikationsergebnis, jedoch sollten Interpretationen vorsichtig vorgenommen werden. Werden stark negative Werte ausgegeben, können Ausreißer der Grund dafür sein. Milligan/Cooper (1983) zeigten in einer Untersuchung, dass das CCC ein geeignetes Kriterium zur Wahl der Klassenanzahl darstellt.

Zur Berechnung des CCC im SAS EM wird zuerst die maximal zu prüfende Klassenzahl k_{max} vorgegeben. Dann wird auf allen Daten mit der vorgegebenen maximalen Klassenzahl k_{max} ein k-means Clustering mit einer Iteration durchgeführt (siehe S. 63ff.). Im Anschluss daran werden die dadurch erhaltenen Klassenzentren mit einer Ward-Clusteranalyse geclustert. Dabei werden schrittweise k_{max}, k_{max-1}, ..., 2 Klassen gebildet und für jede Klassenzahl im Intervall [2; k_{max}] das CCC berechnet. Anhand des CCC-Plots kann der Anwender die Anzahl an Klassen für die endgültige Clusterlösung festlegen.

Damit sind die Grundlagen geschaffen, um dieses Verfahren zur **clusteranalysegestützten Stichprobenziehung** zu verwenden. Die Idee dabei ist, beim Downsizing anstatt einer Zufallsauswahl die Trainings- und Validierungsdaten zuerst mit einer k-means Clusteranalyse in Gruppen einzuteilen und dann aus diesen Gruppen Objekte zufällig auszuwählen. Durch eine derartige Auswahl von Objekten sollen die Unterschiede in den Stichproben bei wiederholter Ausführung deutlich geringer sein als bei einer einfachen Zufallsauswahl. Gerade bei der Auswahl von Objekten aus der 0-Klasse soll in dieser Arbeit dargestellt werden, inwiefern dieses clusteranalysegestützte Verfahren im Vergleich zu einer zufälligen Auswahl Vorteile bei der Stabilität und Güte der Modelle bringt.

Es werden dabei zwei Varianten auf Basis der Trainings- und Validierungsdaten untersucht:

Zum einen werden sowohl Objekte aus der 1-Klasse als auch Objekte aus der 0-Klasse auf Basis aller unabhängigen Variablen getrennt geclustert, wobei die Klassenzahl nach dem CCC-Kriterium bestimmt wird (siehe Abbildung 19).

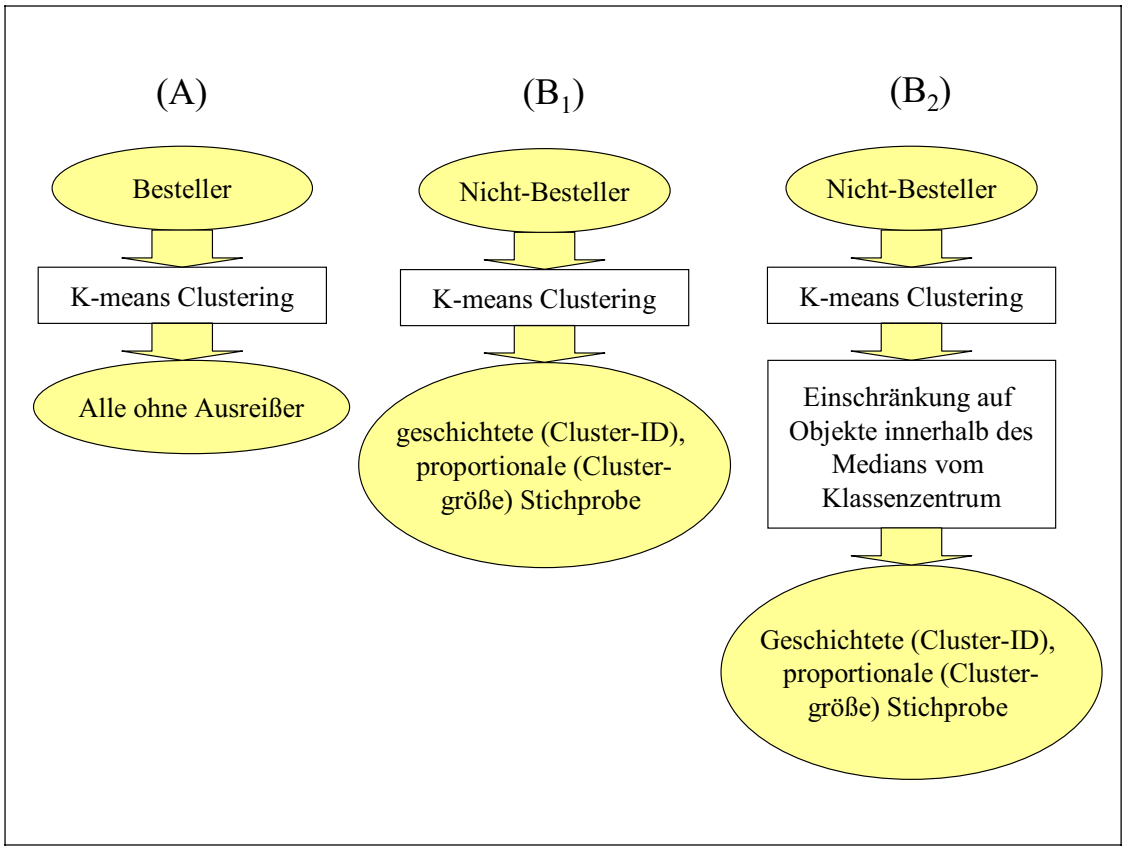


Abbildung 19: Übersicht clustergestützte Stichprobenziehung (1)
Quelle: Eigene Darstellung

Entstehen bei den Bestellern einelementige Klassen, werden diese aus der Analyse eliminiert und speziell abgespeichert, da es sich dabei um Ausreißer handelt (siehe Abbildung 19, Fall A).

Bei den Objekten aus der 0-Klasse wird im Anschluss an die Clusteranalyse eine nach der Klassenzugehörigkeit orientierte Stichprobe durchgeführt. Es wird eine proportional zur Klassengröße geschichtete Stichprobe gezogen (siehe Abbildung 19, Fall B_1), wobei die Klassenzugehörigkeit (auch Clusternummer oder Cluster-ID) der Schichtungsvariablen entspricht. Es wird dadurch der unterschiedlichen Klassengröße Rechnung getragen, das heißt eine große Klasse ist mit vielen, eine kleine Klasse mit wenigen Objekten in der Stichprobe vertreten. Der Ansatz des Prototyping von Reinartz (1999, S. 104ff.) sieht beispielsweise vor, nur ein oder sehr wenige Objekte als Repräsentanten je Klasse auszuwählen. Diese Objekte sollen eine gesamte Klasse vertreten. Bei der Vorgehensweise in dieser Arbeit gelangen bei großen Klassen auch sehr viele Objekte in die

Stichprobe, so dass nicht mehr vom Prototyping im eigentlichen Sinne gesprochen werden kann. Das gewünschte Verhältnis von 1-Klasse zu 0-Klasse in der Zielvariablen kann über die Anzahl der zu ziehenden Objekte gesteuert werden.
Bei einer Erweiterung dieser Variante wird vor der Stichprobenziehung auf Objekte, deren Distanz zum Klassenzentrum kleiner als die Mediandistanz ist, eingeschränkt (siehe Abbildung 19, Fall B_2). Damit gelangen Objekte aus „Grenzregionen“ zu anderen Clustern nicht in die Stichprobe.

Alternativ zur ersten Variante werden die gesamten Trainings- und Validierungsdaten, also 1-Klasse und 0-Klasse gemeinsam, geclustert (siehe Abbildung 20). Es werden somit Cluster gebildet, die Objekte beider Klassen enthalten. Diese Variante wird nur für einen Fall verwendet: Innerhalb jedes Clusters werden alle Objekte der 1-Klasse und ebenso viele Objekte der 0-Klasse mit Hilfe einer Zufallsauswahl ausgewählt.

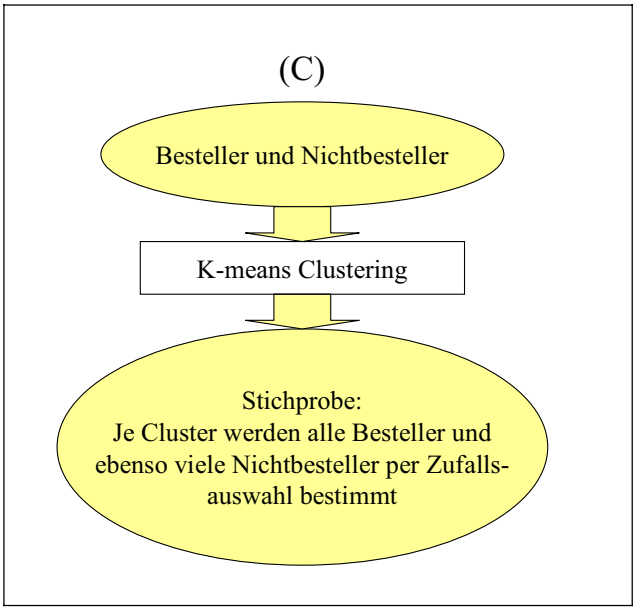


Abbildung 20: Übersicht clustergestützte Stichprobenziehung (2)
Quelle: Eigene Darstellung

Die optimale Clusterzahl wird anhand des CCC-Plots bestimmt, wobei jeweils $k_{max}=100$ vorgegeben wird. Bei Fall A aus Abbildung 19 (S. 71) wird eine 3-Klassen-Lösung, bei Fall B aus Abbildung 19 (S. 71) eine 15-Klassen-Lösung und bei Fall C aus Abbildung 20 (S. 72) eine 8-Klassen-Lösung gewählt (siehe Anhang E, S. 214f.). Tabelle 9 zeigt einen Überblick über die einzelnen Klassengrößen und die Güte, wobei sich die Güte b_K in dem Cluster K wie folgt berechnet: $b_K = \sqrt{IKV_K}$. In der Tabelle wird bei der Cluste-

ranalyse aus Fall C (siehe Abbildung 20, S. 72) die Anzahl der Objekte aus der 1-Klasse je Cluster in Klammern angegeben.

Cluster	Klassengröße	Güte b_K
Fall A: Cluster 1	76	0,2832
Fall A: Cluster 2	240	0,2614
Fall A: Cluster 3	17	0,2415
Fall B: Cluster 1	3185	0,2450
Fall B: Cluster 2	2359	0,1794
Fall B: Cluster 3	13251	0,2200
Fall B: Cluster 4	1656	0,2405
Fall B: Cluster 5	8926	0,1863
Fall B: Cluster 6	2300	0,1957
Fall B: Cluster 7	7345	0,1896
Fall B: Cluster 8	1072	0,2156
Fall B: Cluster 9	1573	0,2086
Fall B: Cluster 10	3320	0,1927
Fall B: Cluster 11	5448	0,1923
Fall B: Cluster 12	2747	0,2187
Fall B: Cluster 13	1406	0,2341
Fall B: Cluster 14	21514	0,1524
Fall B: Cluster 15	1762	0,2407
Fall C: Cluster 1	9422 (52)	0,1927
Fall C: Cluster 2	23695 (63)	0,1688
Fall C: Cluster 3	6861 (43)	0,2454
Fall C: Cluster 4	7616 (18)	0,1921
Fall C: Cluster 5	18677 (89)	0,2370
Fall C: Cluster 6	3876 (23)	0,2620
Fall C: Cluster 7	2217 (15)	0,2085
Fall C: Cluster 8	5833 (30)	0,1974

Tabelle 9: Überblick Clusterlösungen

3.4 Zusammenfassung

Im Rahmen der zweckmäßigen Voranalysen wurden die Datenbeschaffung (siehe S. 41f.), die anschließende Datenvorverarbeitung (siehe S. 42-55) und Methoden zur Bewältigung niedriger Responsequoten (siehe S. 56-73) vorgestellt.
Die verfügbare Datenmatrix enthält 394 Besteller und 91.680 Nichtbesteller (siehe S. 47), dies entspricht einer Responsequote von 0,43%. Der erste Schritt diente zur Vorbereitung der Datenmatrix für die Analyse. Dabei wurden bestimmte Variablen umkodiert (siehe S. 42), indem einzelne Ausprägungen sinnvoll zusammengefasst wurden. Fehlende Werte erhielten bei nominalen und ordinalen Variablen die Ausprägung „U“ für „unbekannt“ (siehe S. 47). Darauf folgte eine Komplexitätsreduktion, das heißt Variablen, die nicht signifikant mit der Zielvariable zusammenhängen, wurden eliminiert (siehe S. 48f.). Vier Variablen sind dabei aufgrund der Erfahrung von Experten wieder in die Analyse aufgenommen worden. Außerdem wurden Variablen, die untereinander sehr stark korrelieren, eliminiert, so dass die Anzahl an Variablen von 61 auf 29 reduziert werden konnte (siehe S. 49). Dann folgte die Aufteilung der Datenmatrix in Trainings-, Validierungs- und Testdaten (siehe S. 53f.). 66 Ausreißer konnten dabei in den Trainings- und Validierungsdaten identifiziert und eliminiert werden (siehe S. 52 und S. 55). Anschließend wurden ausgewählte Methoden zur Bewältigung niedriger Responsequoten vorgestellt (siehe S. 56ff.): „Alle“, „Duplizierung“, „Downsizing“ und „Profitmatrix“. Ein neuer Ansatz im Bereich Downsizing befasst sich mit der Stichprobenziehung auf Basis der ausführlich vorgestellten Clusteranalyse (siehe S. 61ff.). Um die Unterschiede der Methoden zur Bewältigung niedriger Responsequoten zu analysieren, werden insgesamt 27 verschiedene Datenmatrizen gebildet (siehe Tabelle 10).

Anzahl	Modell
1	Alle Daten (siehe S. 57)
1	Profitmatrix (siehe S. 59)
1	Duplizieren (siehe S. 58)
9	Downsizing, je drei einfache Stichproben im Verhältnis 1:1, 1:5, 1:10 (siehe S. 58)
9	Downsizing, jeweils dreimal clusteranalysegestützte Auswahl im Verhältnis 1:1, 1:5, 1:10 (siehe Abbildung 19, Fall A mit B_1, S. 71)

3	Downsizing, je dreimal clusteranalysegestützte Auswahl innerhalb der Mediandistanz je Cluster im Verhältnis 1:1 (siehe Abbildung 19, Fall A mit B_2, S. 71)
3	Downsizing, Clusteranalyse aller Trainings- und Validierungsdaten und anschließend je dreimal Zufallsauswahl im Verhältnis 1:1 (siehe Abbildung 20, S. 72)

Tabelle 10: Überblick Anzahl Datenmatrizen für die Analyse

In den folgenden Kapiteln erfolgt die Vorstellung und Anwendung von Entscheidungsbaumverfahren (Kapitel 4), logistischer Regression (Kapitel 5) und Neuronalen Netzen (Kapitel 6). Anhand der Testdaten wird jeweils die Modellgüte der 81 Varianten verglichen, um den Einfluss der verschiedenen Methoden beurteilen zu können (Kapitel 4.6, 5.4, 6.3). Weiterhin werden bei den Verfahren Entscheidungsbaum und logistischer Regression, das heißt bei 54 Varianten, die verwendeten Variablen je Variante untersucht (Kapitel 7.3). Das Verhalten der Verfahren bei einer Reduzierung der Variablenzahl wird in Kapitel 7.5 beschrieben.
Somit hebt sich diese Studie von den bisher veröffentlichten Arbeiten in diesem Bereich vor allem dadurch ab, dass verschiedene Methoden bei drei unterschiedlichen Verfahren verglichen werden. Weiterhin wird die Sensitivität der Stichprobenziehung bezogen auf die Ergebnisstabilität bei unterschiedlichem Anteil der 1-Klasse untersucht. Der Vergleich der verwendeten unabhängigen Variablen anhand einer MDS je Methode und Modellvariante bei den beiden Verfahren logistische Regression und Entscheidungsbäume ist ebenfalls neu.

Abbildung 21 zeigt die Vorgehensweise in dieser Studie im Gesamtüberblick.

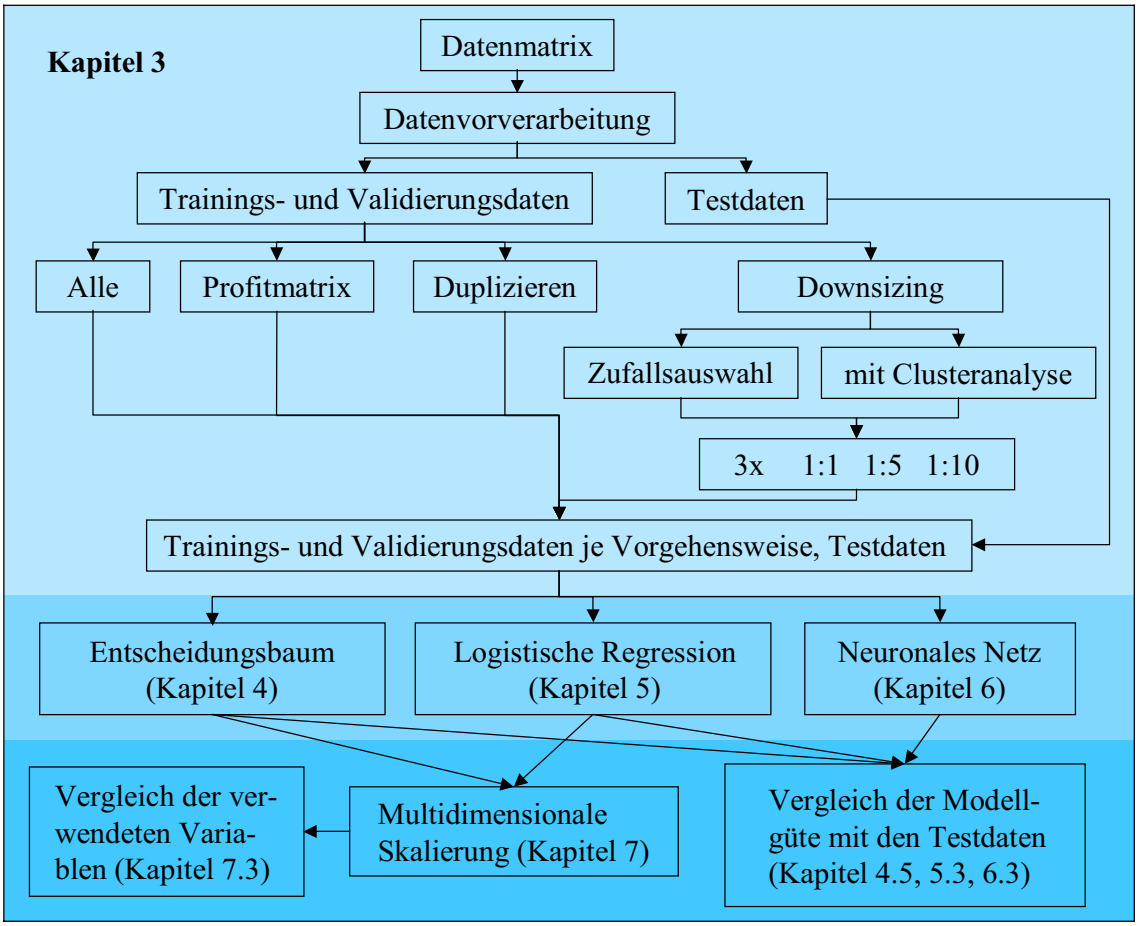


Abbildung 21: Skizzierung der Vorgehensweise dieser Studie
Quelle: Eigene Darstellung

4. Responseoptimierung mit Entscheidungsbaumverfahren

Ein Entscheidungsbaum soll mit Hilfe der unabhängigen Variablen die abhängige Variable identifizieren. Dabei findet eine sukzessive Zerlegung von Wertebereichen der Merkmale statt, um in bezug auf die Zielvariable möglichst homogene Objektmengen zu erhalten (vgl. Breiman et al., 1984; Kass, 1980; Quinlan, 1993).

4.1 Entscheidungsbaumvarianten

Im folgenden werden die Baumbegriffe und der Zerlegungsmechanismus vorgestellt. Abbildung 22 zeigt zur Veranschaulichung der Struktur und zur Erklärung der verwendeten Begriffe einen Entscheidungsbaum.

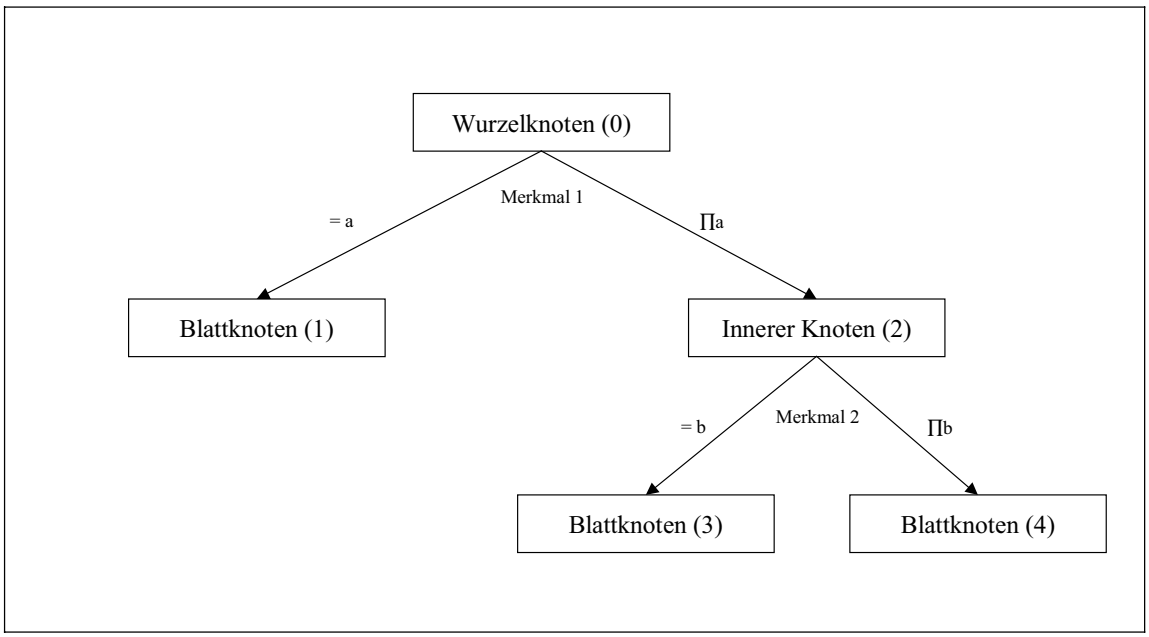


Abbildung 22: Skizze eines Entscheidungsbaums
Quelle: Eigene Darstellung

Ein Entscheidungsbaum besteht aus Knoten und Kanten. In Abbildung 22 entsprechen die Rechtecke den **Knoten** und die Pfeile den **Kanten**. Ausgehend von der gesamten Objektmenge wird ein Merkmal ausgewählt, das die Objektmenge anhand der Merkmalsausprägungen in zwei oder mehr disjunkte Teilmengen zerlegt bzw. **splittet**. Diese Teilmengen sollen bezüglich der Zielvariablen möglichst homogen sein. An den Kanten sind häufig die **Splitwerte** bzw. **–bereiche** des ausgewählten Merkmals bzw. der Splitvariable angetragen. Knoten, die nicht der **Wurzelknoten** sind und an welche weitere

Knoten folgen, bezeichnet man als **innere Knoten** oder „non-terminal nodes“, beispielsweise Knoten (2) in Abbildung 22. Als **Blattknoten** oder „terminal nodes“ werden Knoten bezeichnet, die nicht mehr weiter unterteilt werden, z.B. Knoten (1), (3) und (4). Weiterhin werden die entstehenden Knoten bei einem Split **Sohnknoten**, beispielsweise Knoten (3) und (4), des übergeordneten Knotens oder **Vaterknotens** (2) bezeichnet. Jedes Objekt wird beim Durchlaufen des erstellten Baumes genau einem Blattknoten zugeordnet.

Nahezu allen Entscheidungsbaumverfahren gemeinsam ist die übergeordnete Vorgehensweise nach dem „teile und herrsche“ (divide and conquer) Prinzip; das heißt es werden rekursiv feinere Partitionen gebildet. Somit sind nahezu alle Baumalgorithmen divisive oder Top-Down Verfahren. Es gibt zwei Arten einen Entscheidungsbaum zu generieren. Der **enumerative Ansatz** erzeugt alle möglichen Bäume. Dadurch kann der bestklassifizierende bezüglich der Zielvariablen und kürzeste Baum gefunden werden, allerdings ist diese Vorgehensweise ineffizient (Lusti, 2002, S. 293). Der **heuristische Ansatz** versucht mit Hilfe eines bestimmten Bewertungskriteriums sukzessive eine optimale Auswahl der unabhängigen Variablen und Trennwerte für alle Stufen des Baumes zur Zerlegung zu finden. Es wird ein greedy-Algorithmus benutzt, das heißt es wird nach jedem Split jeweils nur die nächste unabhängige Variable zum nächsten Split ausgewählt (Ester/Sander, 2000, S. 127). Dabei soll jeweils die unabhängige Variable ausgewählt werden, bei der die entstehenden Sohnknoten möglichst homogen bezüglich der Zielvariable sind. Diese Vorgehensweise wird in der Praxis aufgrund sehr großer Daten- und Merkmalsmengen nahezu immer angewandt und ist sehr effizient, findet aber nicht notwendigerweise den bestklassifizierenden Baum (Lusti, 2002, S. 293).

Nach Säuberlich (2000a, S. 88) weisen divisive Verfahren zur Konstruktion von Entscheidungsbäumen folgende drei Haupteigenschaften auf:

(I) Schrittweise Auswahl einer bestmöglichen Unterteilung;
(II) Abbruchkriterium zur Bestimmung der Endknoten;
(III) Zuweisung einer Klassenbezeichnung zu einem Endknoten.

Die Erzeugung von Entscheidungsbäumen teilt sich in aller Regel in eine **Wachstumsphase** oder **Growing** und eine **Stutzungsphase** oder **Pruning**.

Ausgangspunkt sind alle Objektvektoren, die im Wurzelknoten vorliegen (Ripley, 1996, S. 213). Vor jedem Split werden alle unabhängigen Variablen gemäß dem verwendeten **Partitionierungskriterium** bewertet. Die Sohnknoten sollen bezüglich der Zielvariablen möglichst homogen sein. Das Partitionierungskriterium gibt für jede unabhängige Variable an, wie gut die Anpassung bei einem Split mit dieser Variable an die Zielvariable ist. Diejenige Variable, die die beste Bewertung erhält, wird ausgewählt und die Objekte werden gemäß den verschiedenen Merkmalsausprägungen aufgeteilt (Borgelt/Kruse, 1998, S. 82). In Abbildung 22 entspricht dies beispielsweise der Aufteilung in Knoten (1) und (2). Dieses Verfahren wird nun rekursiv auf die sich ergebenden Teilmengen angewandt; das heißt es wird anhand des Partitionierungskriteriums geprüft, ob es eine Variable gibt, welche die Objekte in Knoten (1) bzw. (2) in bezug auf die Zielvariable weiter sinnvoll aufteilen kann.

Die **Rekursion bricht ab**, wenn alle Objekte in allen Endknoten einer bestimmten Klasse der Zielvariablen angehören, keine Variable mehr zu einer Verbesserung der Klassifikation führt, keine weiteren Variablen vorhanden sind oder ein bestimmtes Abbruchkriterium vorgegeben ist (Säuberlich, 2000a, S. 96f.).

Wenn der Baum fertiggestellt ist, folgt das Pruning oder Zurückschneiden des Baums. Dabei wird geprüft, ob bestimmte Teilbäume wieder entfernt werden können, um Overfitting (siehe S. 53) zu vermeiden.

Zum Schluss wird jedem Blattknoten eine **Klassenbezeichnung** bezüglich der Zielvariablen zugewiesen. Bei einer nominalen Zielvariablen wird jedem Blattknoten die Klasse zugeordnet, die am häufigsten in diesem Blattknoten auftritt. Bei einer metrischen Zielvariablen wird jedem Blattknoten der Mittelwert aller Objekte in diesem Blattknoten zugewiesen. Bei der Prognose wird jedes neue zu bewertende Objekt genau einem Endknoten zugeordnet und erhält dann diese Klassenbezeichnung. Dieser Wert wird als **Scorewert** bezeichnet.

Im folgenden werden unterschiedliche Bezeichnungen bei Entscheidungsbäumen eingeführt.

Ist die Zielvariable nominal, spricht man von einem **Klassifikationsbaum**, bei einer metrischen Zielvariablen von einem **Regressionsbaum**. Bei einer Responseoptimierung ist die abhängige Variable in der Regel binär, somit werden in der weiteren Ausführung nur Klassifikationsbäume betrachtet.

Entscheidungsbäume lassen sich auch nach der Anzahl der verwendeten unabhängigen Variablen pro Split unterscheiden. Wird immer nur eine Variable bei einem Split verwendet, so spricht man von einem **univariaten Baum**. Bei einem **multivariaten Baum** sind nicht alle Splits univariat, das heißt es wird mindestens ein Split beispielsweise auf Basis einer Linearkombination von mindestens 2 Variablen ausgeführt. Der Vorteil liegt in der genaueren Abbildungsmöglichkeit von Beziehungen, allerdings wird dabei die Interpretierbarkeit unter Umständen stark eingeschränkt.

Die Anzahl der zugelassenen Aufteilungen pro Variable ist ein weiteres Unterscheidungskriterium. Bei **binären Splits** schließen sich an jeden Knoten genau zwei Kanten an; die entstehenden Bäume werden häufig auch Binärbäume genannt. Bei **mehrfach- bzw. multiway Splits** können an einen Knoten auch mehrere Kanten folgen, im Höchstfall bis zur Anzahl unterschiedlicher Variablenausprägungen.
Zuerst wird der Fall bei binären Splits dargelegt: Sowohl bei kategorialen als auch bei quantitativen unabhängigen Variablen werden alle möglichen binären Splits gebildet und anhand des Partitionierungskriteriums der Beste ausgewählt. Die Anzahl möglicher binärer Splits beträgt bei nominalen Variablen 2^{L-1}-1, wobei L der Anzahl an verschiedenen Variablenausprägungen entspricht. Hat eine nominale Variable beispielsweise vier Ausprägungen, so gibt es 2^{4-1}-1=7 mögliche Aufteilungen. Ordinale Variablen lassen genau L-1 mögliche Splits zu, da die Reihenfolge zu beachten ist und nur bei aufeinanderfolgenden Werten eine Trennung erfolgen sollte. Bei quantitativen Variablen entspricht die Anzahl der Splits ebenfalls L-1. Auch hier erfolgt die Zuordnung gemäß $x \Omega l$ *und* $x > l$, wobei l einem bestimmten Wert der Variablen entspricht.
Im Falle von multiway Splits wächst die Anzahl möglicher Splits bei kategorialen Daten exponentiell gemäß der Bell-Number -1, bei 6 Ausprägungen wären beispielsweise bereits 202 Splits möglich. Quantitative Daten lassen $\frac{(L \,4\, 1)!}{(L \,4\, B)!(B \,4\, 1)!}$ Splits zu, wobei B der Anzahl der maximal zugelassenen Splits entspricht; das heißt bei L=21 und B=10 gibt es bereits 167.960 Möglichkeiten.
Es zeigt sich, dass ein exhaustiver Algorithmus, der alle möglichen Splits bei jedem Knoten untersucht, unter Umständen eine extrem lange Rechenzeit verursacht (Loh/Shih, 1997, S. 816). Im SAS EM sind einige Einstellungen möglich, die zu einer Verringerung der Rechenzeit führen: Falls die Anzahl an Objekten in einem Knoten beispielsweise größer als 5.000 ist, wird eine Stichprobe von 5.000 gezogen. Weiterhin

kann ein Algorithmus verwendet werden, der die Anzahl zu testender Splits bei nominalen und ordinalen Merkmalen deutlich reduziert (siehe Potts, 1999, S. 14).
In der Praxis werden häufig binäre Splits bevorzugt, da die Komplexität im Vergleich zu multiway Splits deutlich geringer ist (Potts, 1999, S. 14). Außerdem lässt sich jeder multiway Split über mehrere binäre Splits einer Variablen reproduzieren.

4.2 Partitionierungskriterien

Die Aufgabe des Partitionierungskriteriums ist also, diejenige unabhängige Variable auszuwählen, die den Vaterknoten in bezug auf die Zielvariable bestmöglich unterteilt.
Um die Berechnung des Partitionierungskriteriums zu veranschaulichen, wird ein Beispiel mit einer binären abhängigen Variablen und zwei binären unabhängigen Variablen eingeführt. Die Zielvariable zeigt hier die Unterscheidung Besteller (1) / Nichtbesteller (0), die Variable A zeigt an, ob ein Umsatz in den letzten 12 Monaten getätigt wurde und die Variable B steht für das Geschlecht.
Bei einer nominalen Zielvariablen bildet eine Kontingenztabelle die Basis zur Berechnung des Partitionierungskriteriums. Es ergeben sich in nachfolgendem fiktiven Beispiel folgende Kontingenztabellen:

Allgemeine Form einer Kontingenztabelle (vgl. Bamberg/Baur, 1998, S. 202):

	y	Zielvariable		
x		0	1	
Variable	0	h_{11}	h_{12}	$h_{1.}$
	1	h_{21}	h_{22}	$h_{2.}$
		$h_{.1}$	$h_{.2}$	N

Tabelle 11: Allgemeine Form einer 2x2 Kontingenztabellen

Die Daten für das Berechnungsbeispiel zeigt Tabelle 12:

		Zielvariable		
		0	1	
Variable A	Kein Umsatz	20	20	40
	Umsatz	20	40	60
		40	60	100

		Zielvariable		
		0	1	
Variable B	m	20	0	20
	w	20	60	80
		40	60	100

Tabelle 12: Beispielhafte Kontingenztabellen für Variable A und B

Die beiden möglichen Entscheidungsbäume für die erste Aufteilung bei Verwendung der Beispieldaten zeigt Abbildung 23.

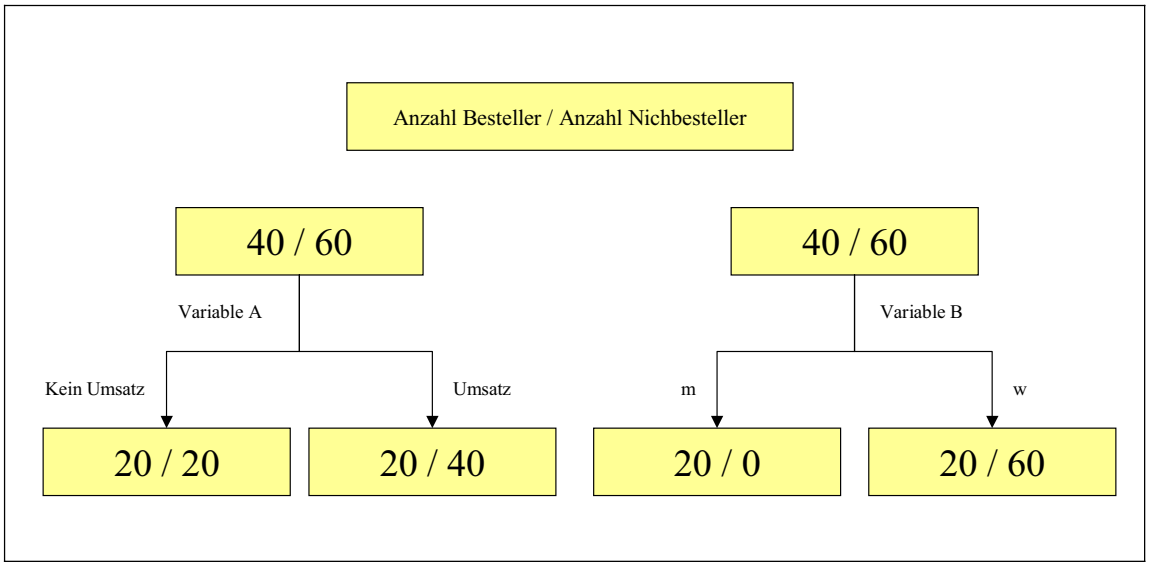


Abbildung 23: Mögliche Entscheidungsbäume aus dem Beispiel
Quelle: Eigene Darstellung

Der Baum, der unter Verwendung der Variablen A entsteht, enthält im linken Endknoten genauso viele Besteller wie Nichtbesteller und im rechten Endknoten doppelt so viele Nichtbesteller wie Besteller. Der Baum, der bei Verwendung der Variablen B ent-

steht, bildet einen Endknoten, der nur Besteller enthält, und einen, der dreimal so viele Nichtbesteller wie Besteller enthält. Im Allgemeinen ist zu erwarten, dass Split B als besser geeignet eingeschätzt wird.

Nachfolgend werden drei Partitionierungskriterien vorgestellt: Der Gini-Index, die Entropie und der θ^2-Unabhängigkeitstest. Es gibt zahlreiche weitere Partitionierungskriterien, auf die hier jedoch nicht näher eingegangen wird.

4.2.1 Gini-Index

Der **Gini-Index** basiert auf der Berechnung der Unreinheit jedes Knotens. Die Unreinheit wird aus dem Anteil verschiedener Klassen in einem Knoten bestimmt (Breiman et al., 1984, S. 39).

Die **Unreinheit des Vaterknotens** t berechnet sich bei einer binären Zielvariablen wie folgt:

$$i(t) = 1 - \left(\frac{h_{.1}}{N}\right)^2 - \left(\frac{h_{.2}}{N}\right)^2 .$$

Im Fall einer binären Zielvariablen ist die Unreinheit am größten, wenn jede Klasse gleich viele Objekte in einem Knoten stellt. Die Unreinheit i(t) nimmt beim Gini-Index im Maximum den Wert 0,5 an, wenn $h_{.1}=h_{.2}=N/2$. Falls nur noch Objekte einer Klasse in einem Knoten sind, ist er rein. Im Idealfall befinden sich in den entstehenden Sohnknoten also nur noch Objekte, die einer bestimmten Klasse angehören: i(t)=0, wenn $h_{.1}$ oder $h_{.2}=0$ und damit $h_{.2}$ bzw. $h_{.1}=N$.

Die **Unreinheit in den Sohnknoten** berechnet sich wie folgt:

$$i(A = l) = 1 - \sum_{z} \left(\frac{h_{lz}}{h_{l.}}\right)^2 ,$$

mit l = Ausprägung der Variable A,

z = Ausprägung der Zielvariablen, $z \subset \{0,1\}$

Die **Verbesserung der Unreinheit** durch die Aufteilung mit der Variablen A errechnet sich aus der Differenz der Unreinheit des Vaterknotens und der gewichteten Summe der Unreinheiten der Sohnknoten:

$$i \mid i(t) - \sum_{l} \frac{h_l}{N} i(A \mid l).$$

Im Beispiel aus Tabelle 12 errechnen sich für Variable A folgende Werte:

$$i(t) \mid 1 - \left(\frac{40}{100}\right)^2 - \left(\frac{60}{100}\right)^2 \mid 0{,}48,$$

$$i(A \mid 0) \mid 1 - \left(\frac{20}{40}\right)^2 - \left(\frac{20}{40}\right)^2 \mid 0{,}5,$$

$$i(A \mid 1) \mid 1 - \left(\frac{20}{60}\right)^2 - \left(\frac{40}{60}\right)^2 \mid 0{,}44,$$

$$i \mid 0{,}48 - \frac{40}{100} 0{,}5 - \frac{60}{100} 0{,}44 \mid 0{,}013.$$

Analog errechnet sich für die Variable B:

$$i(B \mid 0) \mid 1 - \left(\frac{20}{20}\right)^2 - \left(\frac{0}{20}\right)^2 \mid 0,$$

$$i(B \mid 1) \mid 1 - \left(\frac{20}{80}\right)^2 - \left(\frac{60}{80}\right)^2 \mid 0{,}375,$$

$$i \mid 0{,}48 - \frac{20}{100} 0 - \frac{80}{100} 0{,}375 \mid 0{,}18.$$

In diesem Fall wird Variable B als erste Splitvariable ausgewählt, da durch diese Aufteilung eine größere Verbesserung erzielt wird.

Ohne näher auf die relevanten Grundlagen der Wahrscheinlichkeitsrechnung einzugehen, wird der Gini-Index häufig allgemein wie folgt dargestellt:

$$G \mid 1 - \sum p_z^2$$

mit p_z = Eintrittswahrscheinlichkeit der Klasse z = 1,...,q.

Bei einer binären Zielvariablen, das heißt q=2, berechnet sich der Gini-Index als $G = 2p_1(1 - p_1)$.

4.2.2 Informationsgewinn

Quinlan's Informationsgewinn basiert auf der Entropie. Die **Entropie** ist ebenfalls ein Maß, das die Unreinheit in einem Knoten bestimmt (Mitchell, 1997, S. 55). Sie misst die minimale Anzahl von Bits, die zum Codieren einer Nachricht benötigt wird, um die Klasse einer zufälligen Beobachtung zu bestimmen. Eine weitere Interpretationsmöglichkeit ist, dass die Entropie der Anzahl an Ja/Nein-Fragen entspricht, die nötig sind, um die Klasse eines Objektes zu bestimmen. Die Entropie kann Werte zwischen 0 und 1 annehmen, wobei 0 bedeutet, dass eine Nachricht mit 0 Bits – also keine Nachricht – verschickt werden muss, da alle Objekte dieselbe Klasse besitzen. Es wird dabei der Logarithmus zur Basis 2 berechnet, wenn die Nachricht in Bitcode dargestellt werden soll (Mitchell, 1997, S. 57). Im Zweiklassen-Fall nimmt die Entropie dann den Wert 1 an, wenn genauso viele Objekte der 0-Klasse wie 1-Klasse in einem Knoten sind. Der Informationsgewinn repräsentiert die Zahl an Bits, die bei Kenntnis des Wertes einer bestimmten unabhängigen Variablen eingespart werden kann, um ein beliebiges Objekt zu klassifizieren (Borgelt/Kruse, S. 84f.). Shannon (1948) hat für den allgemeinen Fall gezeigt, dass die Entropie eine Untergrenze für die Zahl der notwendigen Fragen darstellt.

Die **Entropie für eine binäre Zielvariable** lässt sich wie folgt berechnen (Quinlan, 1986, S. 89):

$$I = -\frac{h_{.1}}{N} ld\left(\frac{h_{.1}}{N}\right) - \frac{h_{.2}}{N} ld\left(\frac{h_{.2}}{N}\right).$$

I entspricht der durchschnittlich benötigten Information, um neue Beobachtungen einer Klasse zuzuordnen.

Weiterhin lässt sich der **Informationsgehalt der ersten Ausprägung der binären unabhängigen Variablen A** aus dem Beispiel wie folgt berechnen:

$$\tilde{I}(A_1) = -\frac{h_{11}}{h_{1.}} ld\left(\frac{h_{11}}{h_{1.}}\right) - \frac{h_{12}}{h_{1.}} ld\left(\frac{h_{12}}{h_{1.}}\right).$$

In der Kontingenztabelle wird für jede Zeile dieser Wert berechnet, um die **erwartete Information E(A)** berechnen zu können:

$$E(A) = \frac{h_{1.}}{N}\tilde{I}(A_1) + \frac{h_{2.}}{N}\tilde{I}(A_2).$$

Der **Informationsgewinn** der Variablen A berechnet sich aus der Differenz des Informationsgehalts des Vaterknotens und der Information der Sohnknoten:

$$gain(A) = I - E(A).$$

Der Informationsgewinn wird nun für jede Variable berechnet und diejenige mit dem höchsten Wert wird als Splitvariable ausgewählt.

Bezogen auf das Beispiel errechnen sich folgende Werte:

I = - 40/100 ld (40/100) – 60/100 ld (60/100) = 0,97 ,
I(A=0) = - 20/40 ld (20/40) – 20/40 ld (20/40) = 1 ,
I(A=1) = - 20/60 ld (20/60) – 40/40 ld (40/60) = 0,5+0,4 = 0,9 ,
E(A) = 40/100 x 1 + 60/100 x 0,9 = 0,94 ,
Gain(A) = 0,97-0,94 = 0,03.

Analog errechnet sich für Variable B:
I(B=0) = - 20/20 ld (20/20) – 0/20 ld (0/20) = 0 ,
I(B=1) = - 20/80 ld (20/80) – 60/80 ld (60/80) = 0,5+0,3 = 0,8 ,
E(B) = 20/100 x 0 + 80/100 x 0,8 = 0,64 ,
Gain(B) = 0,97-0,64 = 0,33.

Der maximale Informationsgewinn wird bei Variable B erreicht, die somit als Splitvariable ausgewählt wird.

Ohne näher auf die relevanten Grundlagen der Wahrscheinlichkeitsrechnung einzugehen, wird die Entropie allgemein häufig wie folgt dargestellt:

$$entropie = -\sum_{z=1}^{q} p_z \, ld \, p_z .$$

Der Informationsgewinn bevorzugt beim Split gemäß Quinlan (1986, S. 100) Variablen mit vielen Ausprägungen. Deshalb schlägt Quinlan (1993, S. 23) statt dessen das **Informationsgewinnverhältnis** vor. Dabei wird der Informationsgewinn bei dem Split mit Variable X durch den Informationswert des Splits (si) dividiert, um so etwas wie den „Gewinn an nutzbarer Information“ zu erhalten:

$$gain\ ratio = \frac{gain(X)}{si(X)},$$

wobei $si(X) = -\sum_{l} \frac{|T_l|}{|T|} \ld \frac{|T_l|}{|T|}$

mit |T| bzw. $|T_l|$=Anzahl Objekte in Vaterknoten T bzw. Sohnknoten T_l.

si(X) berechnet die Splitinformation, wenn ein Knoten *T* anhand der Variablen X in *l* Sohnknoten gesplittet wird. Dabei kann der Nenner für Splits, die die Ausgangsdatenmenge in wenige große Teilmengen splitten, sehr klein sein. Deshalb kann es vorkommen, dass das Informationsgewinnverhältnis diese Splits sehr hoch einschätzt, obwohl der Zähler, also der Informationsgewinn gain(X), klein ist. Aus diesem Grund schlägt Quinlan (1993, S. 23) vor, nur diejenigen Unterteilungen zu betrachten, für die der Informationsgewinn den durchschnittlichen Informationsgewinn aller Unterteilungen übertrifft.

Im Beispiel errechnen sich folgende Werte:
si(A)= -40/100 ld (40/100) - 60/100 ld (60/100) = 0,52877 + 0,44218 = 0,971 ,
si(B)= -20/100 ld (20/100) - 80/100 ld (80/100) = 0,46439 + 0,25754 = 0,72193.

Damit errechnet sich folgendes gain-ratio:
Gain-ratio(A) = 0,03/0,971= 0,0309 ,
Gain-ratio(B) = 0,33/0,72193 = 0,45711.

Die Entscheidung bleibt bei Variable B.

4.2.3 θ^2-Unabhängigkeitstest

Ein weiteres Partitionierungskriterium ist der **θ^2-Unabhängigkeitstest** (Bamberg/Baur, 1998, S. 202f.). Es wird ein statistischer Test auf Unabhängigkeit durchgeführt und diejenige Variable zum Split ausgewählt, die den stärksten Abhängigkeitsgrad, also den kleinsten p-Wert bzw. den größten Wert aus der Teststatistik, aufweist.
Dabei wird der Pearson θ^2-Test angewandt. Die Teststatistik misst den Unterschied zwischen den tatsächlich beobachteten Werten in der Kontingenztabelle und den erwarteten Häufigkeiten jeder Zelle, wenn die Zeilen und Spalten unabhängig voneinander wären. Die Teststatistik berechnet sich wie folgt (Bamberg/Baur, 1998, S. 202f.):

$$\theta_v^2 = \sum_l \sum_z \frac{(h_{lz} - \tilde{h}_{lz})^2}{\tilde{h}_{lz}}$$

mit v = Anzahl der Freiheitsgrade = (L-1)(q-1) ,

l = Variablenausprägung mit l=1,...,L,

z = Klasse mit z = 1,...,q,

$\tilde{h}_{lz} = \frac{h_{i.} \cdot h_{.j}}{n}$ = erwartete Häufigkeit.

Mit Hilfe der θ_v^2 - Verteilung wird der p-Wert des Splits bestimmt. Das Signifikanzniveau eines Hypothesentests gibt die maximale Irrtumswahrscheinlichkeit an, mit der die Nullhypothese, dass die beiden Variablen unabhängig sind, abgelehnt wird. Der Wert der Teststatistik muss in einem bestimmten Bereich liegen, damit das Signifikanzniveau erfüllt ist. Der p-Wert dagegen gibt an, wie „extrem“ der realisierte Wert der Teststatistik unter Gültigkeit der Nullhypothese ist (Musiol, 1999, S. 43). Er gibt die Irrtumswahrscheinlichkeit vor, die gerade noch zur Verwerfung der Nullhypothese reicht. Ist der p-Wert kleiner oder gleich dem vorgegebenen Signifikanzniveau, so wird die Nullhypothese abgelehnt (Bamberg/Baur, 1998, S. 213). Der p-Wert kann somit direkt zur Entscheidungsfindung herangezogen werden.

Die θ^2-Teststatistik favorisiert Splits in größere Verzweigungen, jedoch wird dieser Nachteil beim p-Wert durch die Anzahl der Freiheitsgrade korrigiert. Wenn nur binäre Splits verwendet werden, ist keine Adjustierung nötig, da die Anzahl der Freiheitsgrade 1 beträgt.

Angewandt auf das Beispiel errechnen sich folgende Werte:

θ^2-Teststatistik für Variable A=
$=(20-16)^2 / 16 + (20-24)^2 / 24 + (20-24)^2 / 24 + (40-36)^2 / 36 =$
= 2,77, das entspricht einem p-Wert von 0,09852.
θ^2-Teststatistik für Variable B=
$=(20-8)^2 / 8 + (0-12)^2 / 12 + (20-32)^2 / 32 + (60-48)^2 / 48 =$
= 37,5 , das entspricht einem p-Wert von < 0,001.

Auch bei diesem Partitionierungskriterium wird die Variable B ausgewählt, da sie einen deutlich geringeren p-Wert aufweist.

4.3 Pruning-Methoden

Nachdem der Baumaufbau abgeschlossen ist, folgt das **Pruning** oder **Zurückstutzen** des Entscheidungsbaums. Es wird dabei das Ziel verfolgt, eine Überanpassung des Baums zu vermeiden (Ester/Sander, 2000, S. 133). Eine Überanpassung oder Overfitting liegt oft dann vor, wenn eine gute Klassifikationsgüte auf den Trainingsdaten erzielt wird, jedoch bei Anwendung auf die Validierungsdaten der Prozentsatz der falsch klassifizierten Objekte deutlich höher ausfällt (Säuberlich, 2000a, S. 96). Es soll der Baum mit der geringsten Komplexität und der besten Prognosefähigkeit gebildet werden (Breiman et al., 1984, S. 59).

Allgemein ist zwischen Pre-Pruning-Techniken oder Early-Stopping, die bereits beim Aufbau des Baumes durch Anwendung bestimmter Abbruchkriterien das weitere Anwachsen verhindern, und Post-Pruning-Techniken, die erst nach dem Aufbau des Baumes einzelne Äste wieder zurückschneiden, zu unterscheiden.

Unter **Pre-Pruning** werden Methoden bezeichnet, die den Baumaufbau begrenzen. Normalerweise würde der Baum so lange aufgebaut, bis alle Blätter einelementig sind bzw. alle Objekte eines Blattes derselben Klasse angehören. Das Ergebnis ist oft ein sehr komplexer Baum, der die Daten zu genau abbildet (Quinlan, 1993, S. 35). Pre-Pruning beendet jedoch in bestimmten Fällen als Top-Down Ansatz den Baumaufbau,

um Overfitting schon frühzeitig zu unterbinden. Solche Regeln können sein (Säuberlich, 2000a, S. 96f.):

(1) Alle Objektvektoren eines Knotens gehören derselben Klasse an.
(2) Alle Objektvektoren eines Knotens besitzen gleiche Merkmalsausprägungen.
(3) Eine bestimmte Mindestanzahl von Objekten pro Knoten wird unterschritten.
(4) Die Verbesserung weiterer möglicher Unterteilungen, gemessen anhand des Partitionierungskriteriums, ist zu gering.
(5) Der θ^2-Unabhängigkeitstest zwischen einer unabhängigen Variablen und der Zielvariablen wird nicht verworfen.
(6) Eine bestimmte Tiefe des Baumes wird überschritten.

Punkt (4) und (5) sind getrennt aufgeführt, da im Unterschied zum Gini-Index oder der Entropie beim θ^2-Unabhängigkeitstest keine subjektive Schranke vorgegeben, sondern mittels eines statistischen Tests in diesem Sinne objektiv entschieden wird.
Feng/Michie (1994, S. 61) schlagen vor, die Aufbauphase eines Entscheidungsbaumes zu beenden, wenn die Unreinheit eines Knotens eine bestimmte Schranke unterschreitet. Dabei können bestimmte Endknoten auch mehrere Klassen enthalten, in diesem Fall würde der Entscheidungsbaum „Class Probability Tree“ genannt werden, da allen Objekten in einem Blatt für jede Klasse eine Wahrscheinlichkeit zugeordnet wird, die der relativen Häufigkeit des Auftretens in diesem Blatt entspricht.
Für die Mindestanzahl an Objekten pro Knoten empfehlen Berry/Linoff (2000, S. 335) einen Wert zwischen 0,25% und 1% der Trainingsdaten. Buja/Lee (2001, S. 30) geben sogar die Empfehlung, ca. 5% der Trainingsdaten als minimale Blattgröße anzugeben.

Ein Nachteil des Pre-Pruning liegt darin, dass es nur auf Informationen aus den Trainingsdaten beruht und somit keine verlässliche, exakte Grenze bestimmen kann, ab wann Overfitting eintritt. Deshalb ist das Pre-Pruning vorsichtig einzusetzen, allerdings kann die Vorgabe einer bestimmten Mindestanzahl an Objekten pro Knoten bereits zu sehr gut angepassten Bäumen führen (Berry/Linoff, 2000, S. 118).

Beim **Post-Pruning** werden die überflüssigen Knoten in einem Bottom-Up Ansatz weggeschnitten. Im folgenden wird das Minimal Cost Complexity Pruning ausführlich vorgestellt. Weiterhin wird das Reduced Error Pruning kurz erklärt.

Das **Minimal Cost Complexity Pruning** geht auf Breiman et al. (1984, S. 66ff. und 303ff.) zurück. Es wird dabei versucht, den Baum mit der geringsten Komplexität gemessen an der Zahl der Endknoten und der besten Modellgüte auf Basis der Validierungsdaten zu ermitteln. Diese beiden Ziele stehen in gewisser Weise in Konflikt zueinander, da der Baum mit der größten Komplexität, also mit den meisten Blättern, auch die beste Modellgüte bei den Trainingsdaten aufweist. Der optimale Baum soll jedoch eine möglichst geringe Komplexität und auf Basis der Validierungsdaten eine sehr gute Modellgüte aufweisen. Zur Bestimmung des optimalen Baums wird ein zweistufiger Prozess durchlaufen. Zuerst werden die Komplexitätskosten auf Basis der Trainingsdaten berechnet (Breiman et al., 1984, S. 66):

$$R_\zeta(T) = R(T) + \zeta \cdot l(T) \qquad \text{mit} \qquad R(T) = \frac{m}{N^{Train}},$$

T: Entscheidungsbaum ,
m: Anzahl falsch klassifizierter Objekte ,
N^{Train}*: Anzahl Trainingsdaten ,*
l(T): Anzahl an Blättern des Baums T.

Die Komplexitätskosten $R_\zeta(T)$ setzen sich somit aus der Gesamtfehlklassifikationsrate $R(T)$ und einer Bestrafung für zunehmende Komplexität des Baums zusammen. $R(T)$ sinkt monoton mit zunehmender Baumtiefe und erreicht den Wert 0 beim Maximalbaum, das heißt dem Baum mit der größten Komplexität T_{max}, da hier alle Objekte in jedem Blatt nur einer Klasse angehören. Die zunehmende Komplexität des Baums wird durch den zweiten Term mit zunehmender Baumtiefe bestraft, indem für jeden zusätzlichen Endknoten Kosten in Höhe von ζ anfallen.

Für $\zeta = 0$ minimiert T_{max} die Komplexitätskosten $R_\zeta(T)$, für $\zeta \to \infty$ besteht der optimale Baum nur aus dem Wurzelknoten. Breiman et al. (1984, S. 285ff.) zeigen, dass für jedes ζ ein Baum existiert, der $R_\zeta(T)$ minimiert. Es existieren Intervalle von ζ, innerhalb derer die Werte von $R_\zeta(T)$ identisch sind. Wird beispielsweise durch die Fusion zweier Endknoten die Komplexität $l(T)$ um 1 bei identischem Wert von $R_\zeta(T)$ reduziert, so sollte der kleinere Teilbaum verwendet werden. Da eine Suche über alle möglichen Teilbäume, die durch verschiedene Folgen von Fusionen entstehen, aufgrund der riesigen Zahl entstehender Teilbäume unter Berücksichtigung der Zahl und Art der verwendeten Variablen zu zeitraubend ist, haben Breiman et al. den sogenannten „Wea-

kest-Link-Algorithmus“ entworfen (Hofman, 1990, S. 947). Dabei wird eine Folge von Teilbäumen erzeugt, wobei als weakest-link derjenige Endknoten eines Teilbaums fusioniert wird, für den ζ minimal ist. Dies wird im folgenden veranschaulicht:
Die Kostenkomplexität eines Knotens t, falls alle Nachkommen dieses Knotens zurückgeschnitten werden, ist $R(t) 2 \check{U}$. Die Kostenkomplexität eines Knotens t besitzt genau dann den gleichen Wert wie die Kostenkomplexität aller Sohnknoten des Knotens t, falls gilt (Breiman et al., 1984, S. 69):

$R(t) 2 \check{U} | \; R(T_t) 2 \check{U} \hat{l}(T)$
$R(t) 4 R(T_t) | \; \check{U} \hat{l}(T) 4 \check{U}$
$R(t) 4 R(T_t) | \; \check{U}(l(T) 4 1)$

$$\zeta | \frac{R(t) 4 R(T_t)}{/l/T0 4 10}$$

mit $l/T0$ = Anzahl Blätter im Teilbaum T,
$R(t)$ = Fehlklassifikationsrate im Knoten t,
$R(T_t)$ = Fehlklassifikationsrate aller Sohnknoten von t.

Es wird für jeden Teilbaum der kritische ζ - Wert berechnet und an dem Knoten t zurückgeschnitten, der den geringsten ζ - Wert hat. Dabei werden alle Nachkommen von t entfernt, t wird zum Endknoten und bekommt die Klassenbezeichnung zugewiesen, die in dem Blatt am Häufigsten vorkommt. Mit dem daraus resultierenden Teilbaum wird dieser Schritt iterativ wiederholt, was zu einer endlichen Folge von Bäumen mit abnehmender Komplexität führt.

Im folgenden wird anhand eines exemplarischen Beispiels die Berechnung der Ŭ–Werte durchgeführt.

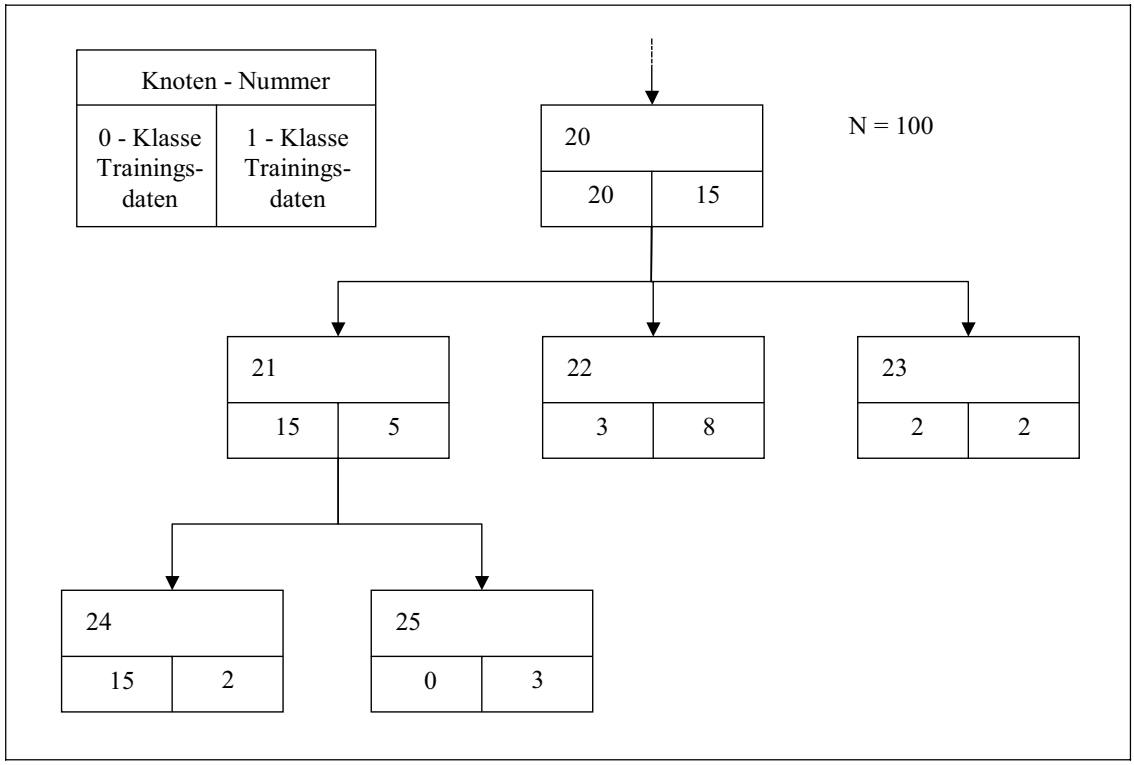


Abbildung 24: Beispiel Pruning-Berechnung
Quelle: Eigene Darstellung

Angenommen, es entsteht beim Baumaufbau folgender Teilbaum, dann wird zuerst für jeden inneren Knoten der Ŭ-Wert berechnet. In diesem Beispiel wird dies exemplarisch für die beiden Knoten 21 und 20 gezeigt.

Für Knoten 21:

$$\zeta_{21} = \frac{R(21) - R(T_{21})}{|\tilde{T}| - 1} = \frac{5/100 - (2/100 + 0)}{2 - 1} = 0{,}03 .$$

Für Knoten 20:

$$\zeta_{20} = \frac{R(20) - R(T_{20})}{|\tilde{T}| - 1} = \frac{15/100 - (2/100 + 0 + 2 \cdot 3/100 + 2 \cdot 2/100))}{4 - 1} = \frac{8/100}{3} = 0{,}02667 .$$

Somit wird bei Knoten 20 zurückgeschnitten, da hier der kleinere Ŭ-Wert erreicht wird. Diese Berechnung wird iterativ für alle restlichen inneren Knoten in dem daraus resultierenden Baum fortgesetzt, die in diesem exemplarischen Beispiel allerdings nicht weiter aufgeführt sind. Somit entsteht eine Folge von Bäumen mit unterschiedlicher Komplexität.

Im nächsten Schritt wird aus dieser Folge der Baum als endgültiger Baum ausgewählt, der die geschätzten Fehlklassifikationskosten insgesamt minimiert. Diese können nicht anhand der Trainingsdaten bestimmt werden, da sonst immer der maximale Baum den besten Wert erreicht. Deshalb wird auf Basis der Validierungsdaten der endgültige Klassifikationsbaum aus der ausschließlich auf Grundlage der Trainingsdaten gebildeten Folge von Bäumen wie folgt ausgewählt:
Es wird der Schätzwert der Fehlklassifikationskosten bezüglich der Validierungsdaten mit dem kleinsten $R_\zeta(T)$ berechnet (Breiman et al., 1984, S. 75):

$$R_{\zeta_0} = \min_{\zeta} R_\zeta^{Val}(T).$$

Weiterhin ist $\breve{u}$ die Standardabweichung dieses Baums, die sich wie folgt berechnet:

$$\iota = \sqrt{\frac{R_{\zeta_0}(100 - R_{\zeta_0})}{N^{Val}}}.$$

Es wird dann der Baum $R_{\zeta^*}(T)$ aus R_{ζ_0} ausgewählt, der die folgende Ungleichung gerade noch erfüllt:

$$R_{\zeta^*}(T) \leq R_{\zeta_0} + b\,\iota \quad \text{mit } b \geq 0.$$

Meist wird b = 1 gewählt, so dass der kleinste Baum ausgewählt wird, dessen Fehlklassifikationskosten innerhalb einer Standardabweichung der minimalen Fehlklassifikationskosten liegen.

Im ersten Schritt wird also auf Basis der Trainingsdaten eine Folge von Bäumen für jedes ζ erstellt und im zweiten Schritt der beste Teilbaum anhand der Fehlklassifikationskosten auf Basis der Validierungsdaten ausgewählt.

Beim **Reduced Error Pruning** (Quinlan, 1993, S. 35ff.) wird der Entscheidungsbaum E, der mit den Trainingsdaten generiert wurde, ebenfalls mit den Validierungsdaten optimiert. Der Algorithmus bestimmt in jedem Schritt denjenigen Teilbaum von E, dessen Abschneiden den Klassifikationsfehler auf Basis der Validierungsdaten am stärksten reduziert und entfernt diesen Teilbaum. Das Pruning ist beendet, wenn kein weiterer Teilbaum mehr existiert, dessen Zurückschneiden den Klassifikationsfehler verringert.

Mingers (1989b, S. 242) zeigt, dass das Cost Complexity Pruning und das Reduced Error Pruning sehr gute Ergebnisse liefern. Es gibt noch verschiedene weitere Pruningverfahren, ein Überblick ist bei Esposito/Malerba/Semeraro (1997) zu finden.

Beim Pre-Pruning ist es schwierig klare Grenzen zu ziehen, wann der Baumaufbau gestoppt werden soll. Breiman et al. (1984, S. 61) und Quinlan (1993, S. 37) kommen unabhängig voneinander zu dem Ergebnis, dass Post-Pruning erfolgversprechender ist als Pre-Pruning. Allerdings schließen sich die beiden Verfahren nicht gegenseitig aus.

4.4 Spezielle Verfahren

Im folgenden werden die drei bekanntesten Entscheidungsbaumverfahren kurz vorgestellt. Vorab ein Überblick:

Name	CART	C 4.5	CHAID
Quelle	Breiman et al. (1984)	Quinlan (1993)	Kass (1980)
Datenniveau (unabh. Var.)	Nominal, metrisch	Nominal, metrisch	Nominal, ordinal
Datenniveau (abh. Var.)	Nominal, ordinal, metrisch	Nominal	Nominal
Auswahlmaß	Gini-Index (S. 83f.)	Informationsgewinnverhältnis (S. 87)	θ^2-Maß (S. 88)
Unterteilungsarten	Binäre Splits	Binäre bzw. multiway-Splits	Multiway-Splits
Abbruchkriterium (Pre-Pruning)	(1) (2) (3) (S. 90)	(1) (4) (S. 90)	(5) (S. 90)
Post-Pruning	Minimal-Cost-Complexity-Pruning (S. 91ff.)	Reduced-Error-Pruning (S. 94)	-

Tabelle 13: Überblick über bekannte Entscheidungsbaumverfahren

Die Bezeichnung **CART** steht für **C**lassification **a**nd **R**egression **T**rees (Breiman et al., 1984). Dieser Algorithmus kann sowohl nominale als auch metrische unabhängige Va-

riablen verarbeiten und lässt bei der Zielvariablen nominales, ordinales oder metrisches Skalenniveau zu (Neville, 1999, S. 19). Bei einer nominalen abhängigen Variablen wird als Partitionierungskriterium der Gini-Index (siehe S. 83f.) oder die Twoing-Regel verwendet (Breiman et al., 1984, S. 38). Bei einer ordinalen Zielvariablen wird die „ordered Twoing Regel“ (Breiman et al., 1984, S. 108) herangezogen und bei einer metrischen Zielvariablen entweder die Reduzierung des quadrierten Fehlers oder die mittlere absolute Abweichung vom Median (Breiman et al., 1984, S. 221ff. und S. 255ff.). Weiterhin sind nur binäre Splits zugelassen.

Eine Besonderheit des Verfahrens ist der Umgang mit fehlenden Daten, da hier die sogenannte „Surrogate Rule“ eingesetzt wird. Dabei wird ein Objekt, das bei der entsprechenden Split-Variablen einen fehlenden Wert aufweist, anhand der zweitbesten Splitvariablen - bzw. falls diese ebenfalls fehlend ist anhand der drittbesten und so weiter - einem der beiden Sohnknoten zugeteilt.

Der CART-Algorithmus läuft zweistufig ab: Zuerst die Growing Phase, in welcher der Baum aufgebaut wird, dann die Pruning Phase, in welcher der Baum zurückgestutzt wird. Anhand der Trainingsdaten wird so lange iterativ nach dem besten Split gemäß dem Gini-Kriterium gesucht, bis die Stopping-Regel (1), (2) oder (3) (siehe S. 90) erreicht ist. In der nächsten Phase wird der Baum anhand der Validierungsdaten gemäß dem Cost Complexity Pruning (siehe S. 91ff.) zurückgestutzt, da die Autoren zu dem Schluss kommen, dass eine angemessene Stopping-Regel nicht bekannt sein kann, bevor die Daten nicht analysiert wurden (Neville, 1999, S. 19). Der Ablauf des Algorithmus wird in Abbildung 25 dargestellt.

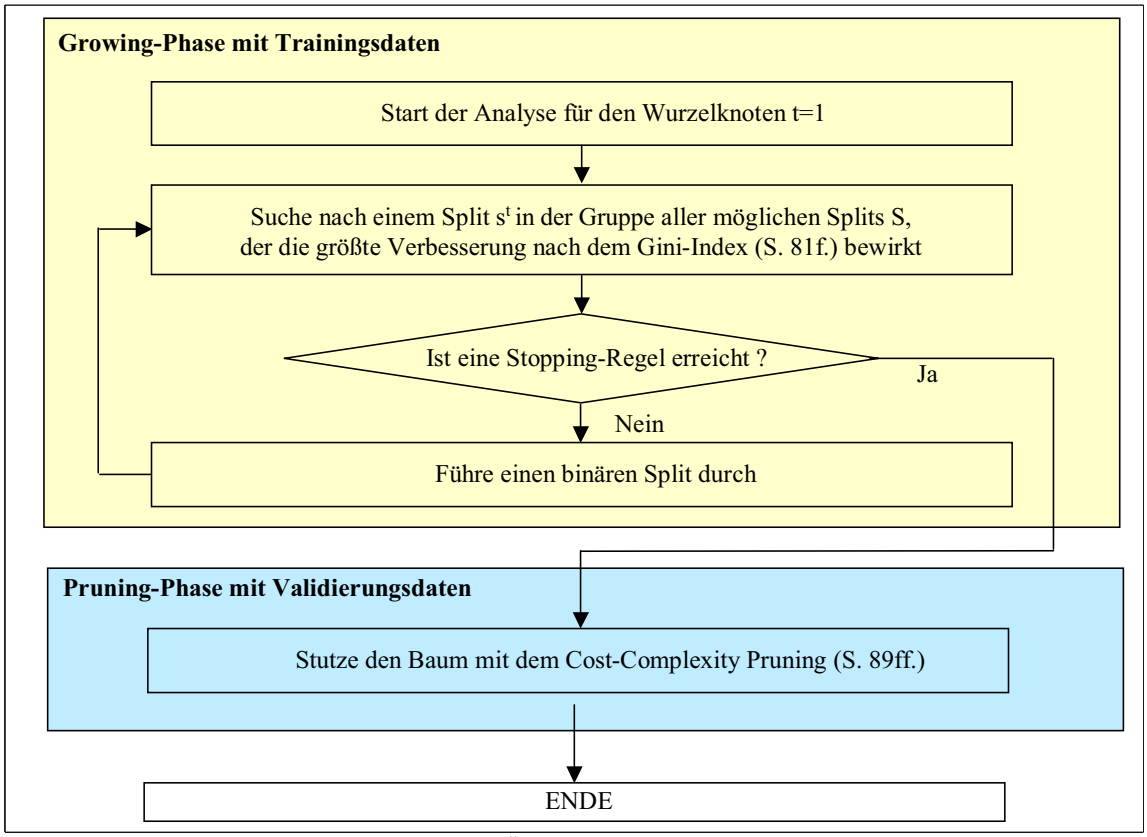


Abbildung 25: Überblick CART Verfahren
Quelle: Eigene Darstellung

C4.5 ist auf Quinlan (1993) zurückzuführen. Die unabhängigen Variablen können nominal oder metrisch sein, die abhängige Variable kann nur nominales Skalenniveau aufweisen. Als Partitionierungskriterium wird das Informationsgewinnverhältnis (siehe S. 87) verwendet. Ist die unabhängige Variable metrisch, so wird ein binärer Split durchgeführt, bei nominalen Variablen sind auch multiway Splits möglich. Als Post-Pruning wird das Reduced-Error-Pruning (siehe S. 94) eingesetzt. Weiterhin kann auf spezielle Imputationstechniken zurückgegriffen werden, die hier nicht näher erläutert werden (siehe Quinlan, 1993, S. 27ff.).

Dieser Algorithmus durchläuft ebenfalls eine Growing- und eine Pruning-Phase. Anhand der Trainingsdaten wird so lange iterativ nach dem besten Split gemäß dem Informationsgewinnverhältnis gesucht, bis die Stopping-Regel (1) oder (4) (siehe S. 90) erreicht ist. In der nächsten Phase werden mit dem Reduced-Error-Pruning anhand der Validierungsdaten die zu viel gebildeten Knoten zurückgestutzt. Abbildung 26 skizziert den Ablauf des C4.5-Algorithmus.

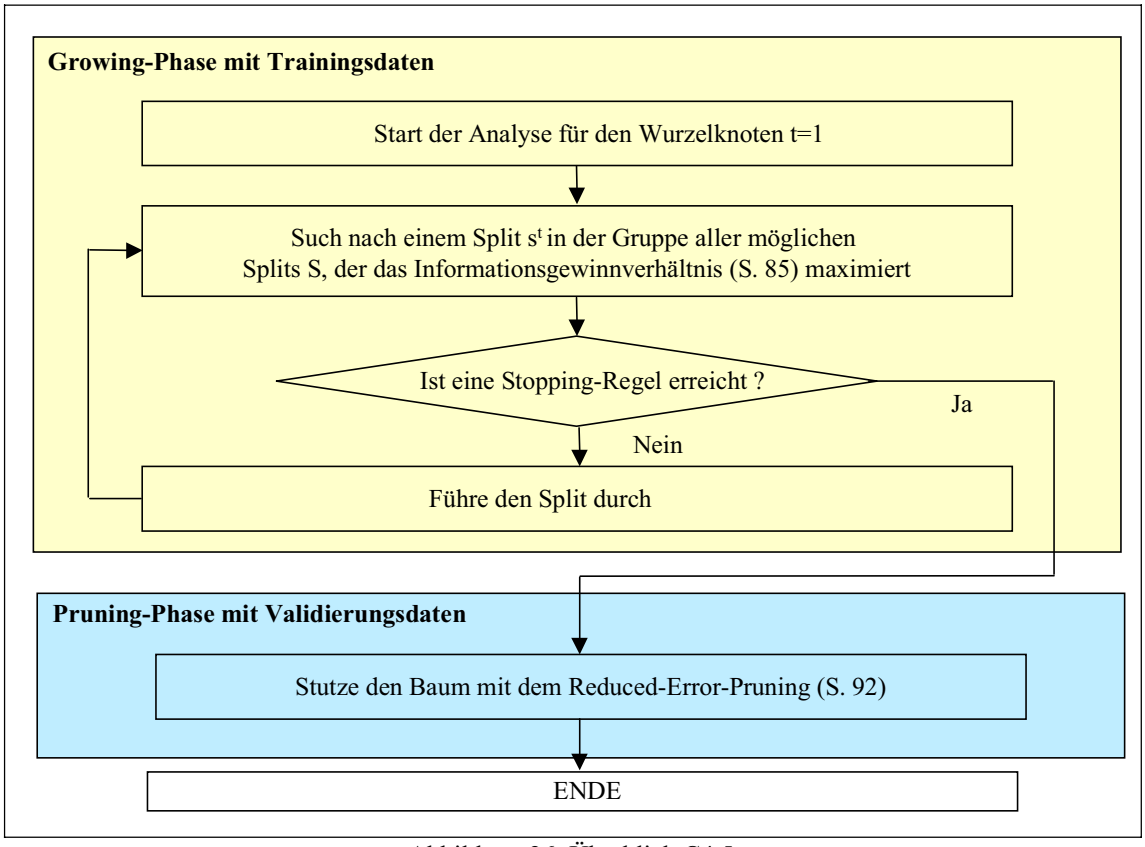


Abbildung 26: Überblick C4.5
Quelle: Eigene Darstellung

Hervorgegangen aus AID von Morgan/Sonquist (1963) ist der auf Kass (1980) zurückgehende **CHAID** (**Ch**i-square **A**utomatic **I**nteraction **D**etector) ein weiteres Entscheidungsbaumverfahren. Die unabhängigen Variablen müssen bei diesem Verfahren entweder nominal oder ordinal sein, das heißt metrische Variablen müssen auf ein ordinales Skalenniveau transformiert werden. Die abhängige Variable muss nominales Skalenniveau aufweisen. Es wird die θ^2-Teststatistik (siehe S. 88) als Partitionierungskriterium verwendet. Bei diesem Verfahren wird kein Post-Pruning durchgeführt und es werden keine Validierungsdaten benötigt.

Der Ablauf des CHAID-Algorithmus lässt sich in Anlehnung an Kass (1980) und Magidson (1994) wie folgt beschreiben (siehe Abbildung 27):

Bei der Initialisierung wird die gesamte Datenmatrix als erste zu betrachtende Gruppe gewählt. Als nächste Phase folgt das Merging. Diese Phase wird im weiteren nur für Knoten durchgeführt, die eine bestimmte Mindestanzahl an Objekten enthalten (Musiol, 1999, S. 45). Dabei werden iterativ Ausprägungen zusammengefasst, die sich in bezug auf die Zielvariable nicht signifikant unterscheiden. Dem Algorithmus kann beim Zusammenfassen der Ausprägungen vorgegeben werden, ob die einzelnen Ausprägungen beliebig miteinander kombiniert werden können („Free-Typ“), beispielsweise bei nomi-

nalen Variablen, oder nur benachbarte Ausprägungen zusammengefasst werden dürfen („Monotonic-Typ“), beispielsweise bei ordinalen oder kategorisierten metrischen Variablen. Weiterhin gibt es noch den Spezialfall, dass nur benachbarte Kategorien zusammengefasst werden dürfen mit der Ausnahme einer einzigen Kategorie, die zu jeder anderen Kategorie zugeordnet werden darf („Float-Typ“); beispielsweise besitzen fehlende Werte eine eigene Ausprägung und dürfen jeder anderen Ausprägung zugeordnet werden. Im nächsten Schritt folgt die Splitting-Phase. Hier wird für jede Variable aus dem θ^2-Unabhängigkeitstest mit der Zielvariablen der p-Wert bestimmt und diejenige mit dem kleinsten p-Wert, falls dieser eine bestimmte maximale Irrtumswahrscheinlichkeit ζ_E (Eligibility-Level) unterschreitet, zur Unterteilung herangezogen. Falls alle p-Werte größer ζ_E sind, erfolgt keine weitere Unterteilung und der Algorithmus geht in die Stopping-Phase. Bei der endgültigen Bestimmung der p-Werte wird beim CHAID-Algorithmus die sogenannte Bonferroni-Korrektur verwendet. Diese Korrektur berücksichtigt, dass die Zahl der Kategorien in der Merging Phase eventuell reduziert wurde und diese Auswahl nur eine von B möglichen darstellt, für die ein Test auf Unabhängigkeit durchzuführen wäre. Deshalb schlägt Kass (1980, S. 122) vor, nicht den p-Wert p^* der als optimal eingestuften Kontingenztabelle zu verwenden, sondern den adjustierten p-Wert $p^{adj.} \mid B \int p^*$. Diese Korrektur ist nach Musiol (1999, S. 50) sehr konservativ und kann dazu führen, dass ein tatsächlich vorhandener Zusammenhang nicht erkannt wird. Die Stopping Phase beendet den Baumaufbau, wenn alle Objekte analysiert sind oder es nur noch Knoten gibt, die nicht über die geforderte minimale Anzahl an Objekten verfügen.

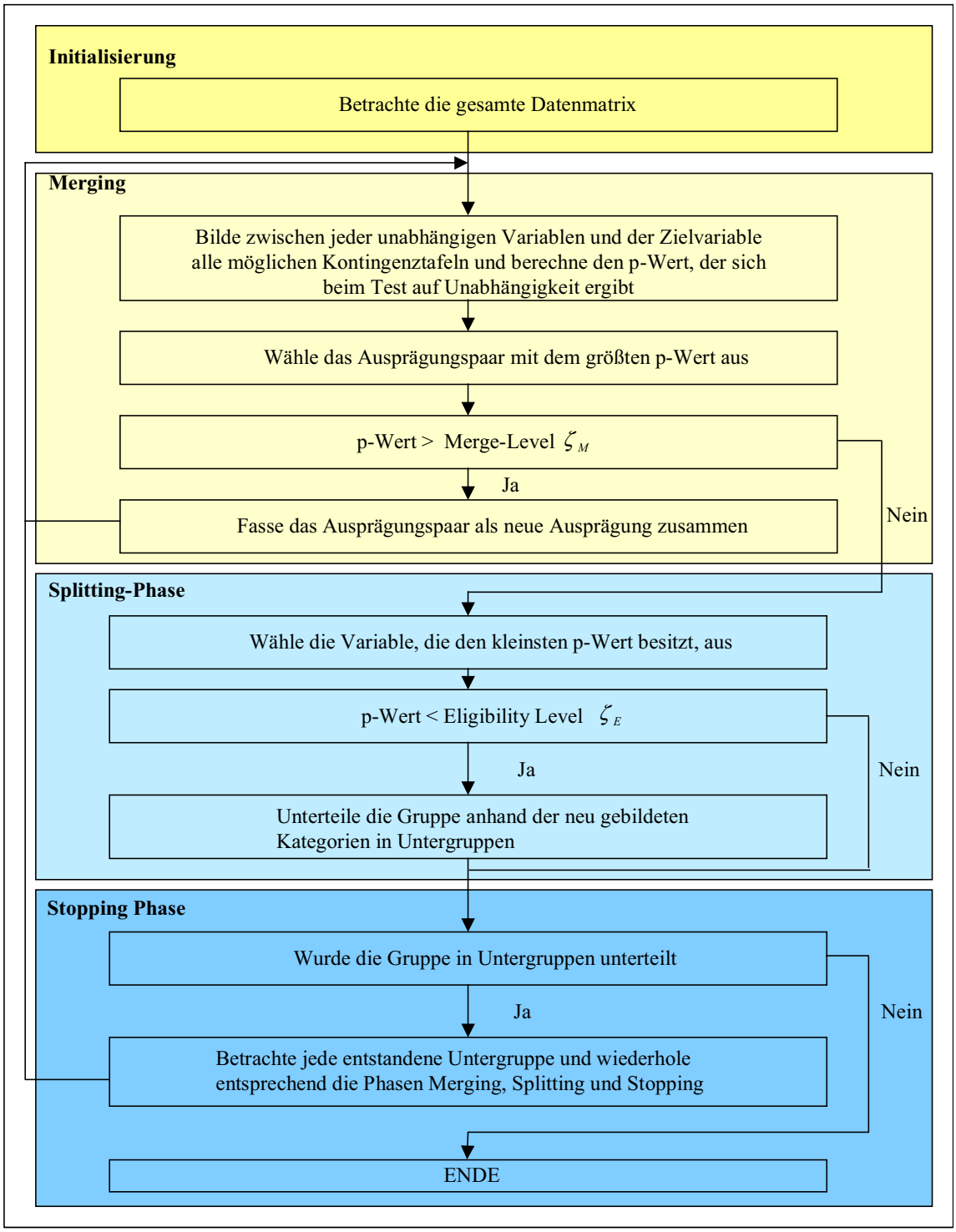


Abbildung 27: Überblick CHAID
Quelle: In Anlehnung an Musiol/Steinkamp (1998, S. 583)

4.5 Probleme bei Entscheidungsbaumverfahren

Nachfolgend erfolgt eine Aufstellung verschiedener Probleme und Empfehlungen aus der Literatur. Dabei werden folgende Punkte aufgeführt:

- Hoch korrelierte unabhängige Variablen,
- die Wahl des Partitionierungskriteriums,
- die heuristische Suche nach einem Split,
- Splits bei metrischen unabhängigen Variablen,
- blattweise Beurteilung der Objekte,
- Pruning bei niedrigen Responsequoten,
- die Instabilität von Entscheidungsbäumen.

Die Interpretierbarkeit von Bäumen ist schwierig, wenn **hoch korrelierte unabhängige Variablen** vorliegen. Hoch korrelierte unabhängige Variablen treten innerhalb eines Baums in einer mehr oder weniger zufälligen Reihenfolge auf, wobei eine Variable zum Teil denselben Informationsgehalt hat. Aufgrund zunehmender Baumkomplexität kann es zu Fehlinterpretationen kommen (Buja/Lee, 2001, S. 31). In dieser Arbeit wurden bereits in der Voranalyse hoch korrelierende unabhängige Variablen eliminiert (siehe S. 48f.).

Bei der **Wahl des Partitionierungskriteriums** kommen Mingers (1989a, S. 338), Borgelt/Kruse (1998, S. 97) und Feng/Michie (1994, S. 63) zu dem Schluss, dass kein Partitionierungskriterium allen anderen überlegen ist. Buja/Lee (2001, S. 29) erwähnen, dass bei einer binären Zielvariablen kein klarer Unterschied zwischen dem Gini-Index und der Entropie besteht. Bei einer polytomen Zielvariablen jedoch scheint der Gini-Index die Majoritätsklasse eher in einen reinen Knoten zu bringen und die Minoritätsklasse in den anderen Knoten, während die Entropie versucht eher gleich große Knoten zu bilden. Für diesen Fall ist laut Breiman (1996a, S. 45) deshalb der Gini-Index zu bevorzugen.

Zu dem Problem, dass die Heuristik beim Baumaufbau immer nur den nächsten Split betrachtet und diesen beim weiteren Baumaufbau nicht mehr hinterfragt, gibt es eine Reihe von Untersuchungen. Unter der Bezeichnung **lookahead** oder backtracking werden Erweiterungen verstanden, die beim Baumaufbau mehrere Ebenen „nach vorne schauen“, um zu sehen, ob ein anderer Split, der gemäß dem Partitionierungskriterium

an dieser Stelle des Baums zwar schlechter ist, im späteren Verlauf des Baumaufbaus jedoch zu einer besseren Gesamtgüte geführt hätte. Weiss/Kulikowski (1991, S. 130f.) empfehlen kein lookahead durchzuführen, da es keine Verbesserung bringt und sehr viel Rechenzeit kostet. Murthy/Salzberg (1995a, S. 1029) schließen sich dem an und verweisen darauf, dass Pruning mehr Einfluss hat. Die meisten Studien kommen zu dem Ergebnis, dass lookahead keinen Vorteil gegenüber dem greedy-Baumaufbau hat (Murthy, 1998, S. 364).

Die **Trennung bei metrischen unabhängigen Variablen** wird ebenfalls häufig kritisiert. In der Regel hängt es sehr stark von den Trainingsdaten ab, ob ein Split beispielsweise beim Alter bei 41 oder 42 stattfindet (Edelstein, 1999, S. 16). Durch eine unterschiedliche Aufteilung der Datenmatrix in Trainings- und Validierungsdaten könnte beispielsweise eine andere Altersgrenze beim Split ausgewählt werden. Allerdings sind Entscheidungsbaumverfahren durch die Teilung des Vaterknotens in beispielsweise nur zwei Sohnknoten sehr robust gegen Ausreißer, da nur Werte > oder Ωals der Splitwert betrachtet werden.

Als Nachteil von Entscheidungsbäumen ist **die blattweise Bewertung der Objekte** (siehe S. 79) zu nennen. Alle Objekte in einem Blatt erhalten dieselbe Bewertung. Gibt es bei einem Baum beispielsweise nur sechs Endknoten, so gibt es maximal sechs unterschiedliche Bewertungen. Dies ist beispielsweise dann problematisch, wenn nur die besten 5.000 Objekte ausgewählt werden sollen und das Blatt mit der besten Bewertung jedoch bereits 10.000 Objekte enthält.

Pruning bei C4.5 und CART zielt auf die Minimierung der Fehlklassifikationsrate ab und wird bei einer sehr selten auftretenden 1-Klasse alles bis auf den Wurzelknoten zurückschneiden, da bei einer Zuordnung aller Objekte zur 0-Klasse der minimale Fehler erreicht wird (Zadrozny/Elkan, 2001, S. 205). Ezawa/Singh/Norton (1996, S. 140) und Berry/Linoff (2000, S. 335) zeigen, dass ein Entscheidungsbaum häufig nur aus dem Wurzelknoten besteht, falls in einer Datenmatrix die 1-Klasse nur wenige Prozent der Gesamtmenge ausmacht. Ein Ausweg wäre, beim Pruning anstatt der Fehlklassifikationsrate ebenfalls den Gini-Index zu verwenden. Dieser Baum wird bei Breiman et al. (1984, S. 121ff.) „Class Probability Tree“ genannt.

Entscheidungsbäume sind **instabil**, das heißt bereits kleine Änderungen in den Trainingsdaten können zu unterschiedlichen Modellen führen (Breiman, 1994, 1996b). Eine Möglichkeit, die insgesamt zu stabileren Ergebnissen führt, ist das wiederholte Ausführen von Entscheidungsbaumverfahren auf unterschiedlichen Datenausschnitten, beispielsweise durch Bagging (Breiman, 1996b), Boosting (Freund/Schapire, 1996), Arcing (Breiman, 1998) oder Random Forests (Breiman, 2001). Da diese Techniken in dieser Arbeit nicht angewandt werden, wird zur genaueren Beschreibung auf die Literatur verwiesen. Bauer/Kohavi (1999), Dietterich (2000) und Quinlan (1996) zeigen in Versuchen die Überlegenheit in der Prognosefähigkeit von derartigen Methoden. Der Nachteil dabei ist, dass die Interpretation der Ergebnisse in der Regel problematisch ist.

4.6 Empirische Ergebnisse

Im folgenden werden die empirischen Ergebnisse der Responseoptimierung mit Entscheidungsbäumen vorgestellt. Es wird der CART-Algorithmus als „Class Probability Tree“ angewandt. Dabei wird als minimale Anzahl an Objekten je Blatt ca. 1% der Trainingsdaten gewählt.

Bei der Methode Profitmatrix (siehe S. 59) hat diese beim Baumaufbau keinen Einfluss. Beim Post-Pruning wird gemäß dem Cost-Complexity Pruning (siehe S. 91ff.) eine Folge von Teilbäumen erstellt. Im zweiten Schritt wird dann bei dieser Methode der Teilbaum ausgewählt, der den Gesamtprofit maximiert.

Im Anschluss an die Modellbildung werden die Modelle verglichen und bewertet. Es wird die Modellgüte anhand der Testdaten verglichen. Dabei stehen verschiedene Möglichkeiten zur Auswahl, beispielsweise die Anzahl korrekt klassifizierter Fälle, der Gesamtfehler oder ein Gains-Chart.

In dieser Arbeit wird zum Vergleich der Prognosefähigkeit der **Gains-Chart** herangezogen. Ling/Li (1998, S. 75) zeigen, dass der Gains-Chart gerade bei Daten mit einer sehr selten auftretenden Klasse sehr gut geeignet ist, die Modellgüte zu beurteilen. Der Gains-Chart stellt in der kumulierten Darstellung die bekannteste Form dar, da dieser Chart prinzipiell dem Modell der Lorenzkurve gleicht. Dabei werden auf der Abszisse die kumulierte Anzahl der Objekte in absteigender Reihenfolge bezüglich ihrer Score-

werte (meist in Dezilen) und auf der Ordinate der entsprechende Anteil an der Gesamtbestellerzahl abgetragen. Es wird somit angegeben, wie viel Prozent der Merkmalsträger jeweils über wie viel Prozent der Merkmalssumme verfügen (Bamberg/Baur, 1998, S. 24ff.; Gierl, 1995, S. 549ff.). Analog zum Prinzip der Lorenzkurve werden in einem (u,v)-Koordinatensystem zusätzlich zur Winkelhalbierenden die Punkte (0,0), (u_1,v_1), ... (u_n,v_n) abgetragen, wobei u_v der kumulierte Anteil an Objekten und v_v den kumulierten relativen Bestelleranteil darstellt:

$$v_v = \frac{\sum_{i=0}^{v} z_i}{\sum_{i=0}^{n} z_i} \quad , \quad u_v = \frac{v}{n}$$

mit n = Anzahl Objekte,

v = die v Objekte mit dem höchsten Scorewert, wobei 1 Ωv Ωn,

z_i = Objekt i ist Besteller.

Abbildung 28 zeigt einen exemplarischen Verlauf eines Gains-Charts.

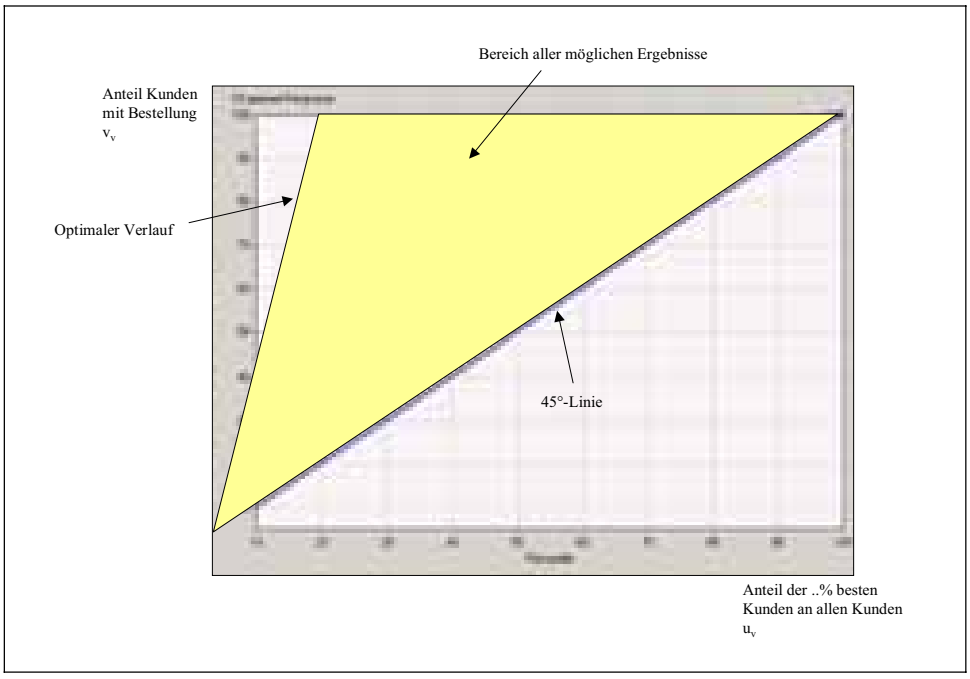


Abbildung 28: Exemplarischer Gains-Chart
Quelle: Eigene Darstellung

In Abbildung 28 ist beispielsweise zu erkennen, dass mit 10 Prozent der besten Kunden bereits ca. 40 Prozent der Besteller erreicht werden können. Mit 20 Prozent der besten

Kunden werden ca. 60 Prozent der Besteller erreicht und mit 80 Prozent der besten Kunden werden ca. 96 Prozent der Besteller erreicht. Bei einer zufälligen Anordnung der Objekte würde die Kurve um die 45°-Linie schwanken. Wenn insgesamt 20% Besteller in der Datenmatrix vorhanden wären, zeigt der optimale Verlauf bei 20% der Gesamtdaten bereits 100% der Besteller. Dabei verläuft die Kurve konstant steil ansteigend bis zum Maximum der Ordinate und ab diesem Punkt parallel zur Abszisse. Ein modelloptimierter Kurvenverlauf wird zwischen der 45°-Linie und dem optimalen Verlauf liegen. Ziel bei der Modellierung ist demnach, möglichst den optimalen Kurvenverlauf zu erreichen.

Alternativ kann zum Modellvergleich auch die **Anzahl korrekt klassifizierter Objekte** oder der **Gesamtfehler** verwendet werden. Diese beiden Gütemaße bestimmen allerdings die korrekte Vorhersage für die gesamte Datenmenge, während der Gains-Chart die Prognosegenauigkeit für einen bestimmten Datenausschnitt zeigt (Hughes, 1996, S. 283f.). Da es bei einer Responseoptimierung das Ziel ist, geeignete Teilpopulationen mit weit überdurchschnittlicher Bestellquote zu finden und nicht, das Bestellverhalten der gesamten Gruppe zu messen, ist der Gains-Chart das geeignetere Maß (Piatetsky-Shapiro/Masand, 1999, S. 187).

Im Direktmarketing ist vor allem die Leistungsfähigkeit in den Top-Segmenten oder aber die Einsparung von Adressen wichtig, deshalb werden zum Modellvergleich nur folgende Punkte aus dem Gains-Chart verwendet: Der Wert bei dem 10%, 20%, 40% und 80% Dezil.

Im folgenden werden die empirischen Ergebnisse der elf verschiedenen Varianten gezeigt. Die Bezeichnungen werden in Anlehnung an Tabelle 10 (siehe S. 75) in Tabelle 14 erklärt:

Modellname	Modellbeschreibung
alle	Alle Daten gehen in das Modell ein (siehe S. 57)
profit	Alle Daten mit Profitmatrix (siehe S. 59)
dupl	Alle Besteller so lange dupliziert, bis Anzahl Nichtbesteller erreicht (siehe S. 58)
z 1:10	Zufallsauswahl der Nichtbesteller im Verhältnis 1:10 (siehe S. 58)
cl 1:10	Clustergestützte Auswahl der Nichtbesteller im Verhältnis 1:10 (siehe S. 71)
z 1:5	Zufallsauswahl der Nichtbesteller im Verhältnis 1:5 (siehe S. 56)
cl 1:5	Clustergestützte Auswahl der Nichtbesteller im Verhältnis 1:5 (siehe S. 71)
z 1:1	Zufallsauswahl der Nichtbesteller im Verhältnis 1:1 (siehe S. 58)
cl 1:1	Clustergestützte Auswahl der Nichtbesteller im Verhältnis 1:1 (siehe S. 71)
cl all 1:1	Clusteranalyse aller Objekte, dann Zufallsauswahl je Cluster mit Verhältnis 1:1 (siehe S. 72)
cl med 1:1	Wie cl 1:1, jedoch eingeschränkt auf Objekte, die innerhalb des Medians je Cluster liegen (siehe S. 71)

Tabelle 14: Überblick über die 11 Varianten bei der Modellerstellung

Falls mehrere Stichproben bei einer Variante erstellt werden, erfolgt nach dem Modellnamen eine aufsteigende Nummerierung. Der Modellname „z 1:1 1“ bedeutet somit, dass es sich um das Ergebnis mit der Methode „z 1:1“, erste Stichprobe handelt. Somit ergeben sich aus den elf Varianten insgesamt 27 verschiedene Modellergebnisse, da bei einigen Varianten drei Stichproben gezogen werden.

Tabelle 15 zeigt die empirischen Ergebnisse der Entscheidungsbaumverfahren auf Basis der Testdaten.

	10%	20%	40%	80%
alle	22	38	64	89
profit	40	50	73	99
dupl	39	49	68	93
z 1:10 1	32	46	64	88
z 1:10 2	37	47	64	88
z 1:10 3	31	45	68	99
cl 1:10 1	39	44	61	95
cl 1:10 2	41	50	72	99
cl 1:10 3	44	53	71	96
z 1:5 1	28	41	64	92
z 1:5 2	37	48	72	98
z 1:5 3	40	49	70	94
cl 1:5 1	34	48	70	98
cl 1:5 2	40	55	68	99
cl 1:5 3	40	54	68	95
z 1:1 1	37	49	66	88
z 1:1 2	39	54	66	95
z 1:1 3	28	51	66	91
cl 1:1 1	41	55	75	98
cl 1:1 2	34	47	70	99
cl 1:1 3	40	51	74	99
cl all 1:1 1	28	43	66	92
cl all 1:1 2	42	52	71	98
cl all 1:1 3	33	46	76	96
cl med 1:1 1	27	41	65	92
cl med 1:1 2	39	51	62	100
cl med 1:1 3	20	42	65	90

Tabelle 15: Gains-Chart Ergebnisse auf Basis der Testdaten bei Entscheidungsbaumverfahren

Die Ergebnisse zeigen, dass deutliche Unterschiede zwischen den Vorgehensweisen bestehen. Die Schwankungsbreite ist dabei relativ groß, besonders auffällig ist die schlechte Performance der Methode „alle“. Der Unterschied zum besten Modell (cl 1:10 3) beträgt beim 10%-Wert 22 Prozentpunkte. Der Mittelwert bei 10% der Einsatzmenge beträgt 35%, beim 20%-Wert liegt er bei 48%. Beim 40%-Wert werden im Schnitt 68% der Besteller erreicht, beim 80%-Wert in etwa 95%.

Die Methode „alle“ schneidet beim 10%-Wert deutlich schlechter ab als alle restlichen Modellvarianten. Auch im weiteren Verlauf werden keine guten Ergebnisse erreicht. Diese Methode scheint bei niedrigen Responsequoten für Entscheidungsbaumverfahren ungeeignet zu sein.

Die Methode „profit“ erreicht bei allen Werten gute bis sehr gute Ergebnisse. Die Verwendung einer Profitmatrix scheint zur Bewältigung niedriger Responsequoten geeignet zu sein.

Die Duplizierung der Besteller führt zu ähnlichen Ergebnissen wie bei der Methode „profit“, allerdings wird aufgrund der größeren Datenmenge mehr Rechenzeit benötigt. Ein weiterer Nachteil liegt in der deutlich steigenden Baumkomplexität. Oates/Jensen (1997, S. 261) zeigen in einem Versuch, dass eine Vergrößerung der Trainingsdatenmenge nur zu tieferen Bäumen führt, die jedoch keine bessere Modellgüte erzielen. Dies wird durch die Ergebnisse in dieser Arbeit bestätigt.

Bei der Methode „z 1:10“ werden drei Modelle verglichen, da dreimal eine Zufallsstichprobe aus den Nichtbestellern gezogen wird. Beim 10%-Wert sind die Ergebnisse schlecht bis unterdurchschnittlich, während beim 80%-Wert ein sehr gutes und zwei unterdurchschnittliche Ergebnisse vorliegen. Dies zeigt die Instabilität von Entscheidungsbaumverfahren bei Verwendung unterschiedlicher Daten. Die Streuung insgesamt ist relativ stark.

Die Ergebnisse der Methode „cl 1:10“ liegen deutlich über den Ergebnissen der Methode „z 1:10“. Der zusätzliche Vorverarbeitungsschritt scheint sich hier positiv auf die Ergebnisse auszuwirken.

Bei der Methode „z 1:5“ sind die Ergebnisse beim 10%-Wert sehr unterschiedlich. Das schlechteste und das beste Modell liegen 12 Prozentpunkte auseinander. Im weiteren Verlauf gleichen sich die Modellschwankungen an. Im Vergleich zu „z 1:10“ ist die Streuung insgesamt etwas größer.

Die Methode „cl 1:5“ weist im Vergleich zu „z 1:5“ deutlich geringere Schwankungen auf. Die Streuung von „cl 1:5“ ist der von „cl 1:10“ sehr ähnlich. Insgesamt werden gute bis sehr gute Werte erreicht.

Die Methode „z 1:1“ erzielt sehr ähnliche Ergebnisse wie „z 1:5“.

Die Methode „cl 1:1“ erzielt ebenfalls sehr ähnliche Ergebnisse wie bei „cl 1:5“, allerdings werden hier insgesamt etwas bessere Ergebnisse erzielt. Im Vergleich zu „z 1:1“ werden insgesamt bessere Ergebnisse bei geringerer Schwankungsbreite erzielt.

Die Methode „cl all 1:1“ weist sehr starke Schwankungen auf. Dabei werden sowohl sehr gute als auch sehr schlechte Ergebnisse erzielt.

Ähnlich verhält es sich bei der Methode „cl med 1:1“. Hier wird das schlechteste Ergebnis bei dem 10%-Wert erzielt. Allerdings werden bei einem Modell mit 80% der Kunden bereits alle Besteller identifiziert.

Abbildung 29 zeigt in einem Säulendiagramm das Ergebnis aller Modelle beim 10%-Wert im Gains-Chart, das heißt wie viel Prozent der Besteller mit den besten 10% aller Adressen erreicht werden. Die Abbildung verdeutlicht, dass die Schwankung bei den Zufallsauswahlmethoden größer ist, als bei den clusteranalysegestützten Auswahlmethoden (ausgenommen die Methoden „cl all 1:1“ und „cl med 1:1“). Zudem sind die Ergebnisse der clusteranalysegestützten Auswahlmethoden insgesamt besser. Die beiden Methoden „cl all 1:1“ und „cl med 1:1“ zeigen im Vergleich zu „cl 1:1“ sowohl eine deutlich stärkere Streuung als auch schlechtere Ergebnisse und sind somit nicht zu empfehlen. Ebenfalls ist die Methode „alle“ deutlich unterlegen. Insgesamt liegen die Ergebnisse zwischen 22% und 44%.

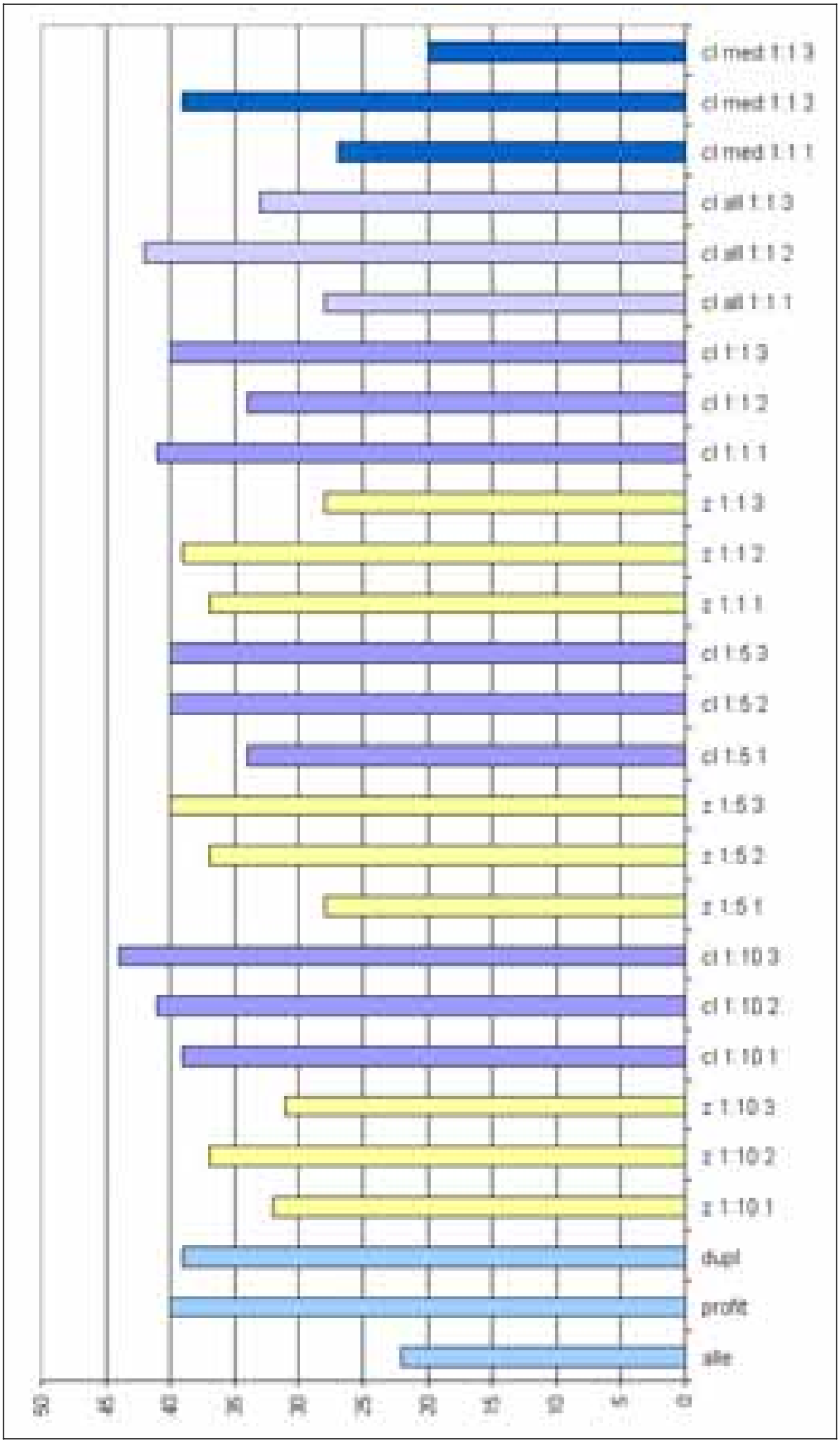


Abbildung 29: Gains-Chart Ergebnisse beim 10%-Wert auf Basis der Testdaten bei Entscheidungsbaumverfahren
Quelle: Eigene Darstellung

Werden die Ergebnisse der Methoden z 1:10, cl 1:10, z 1:5, cl 1:5, z 1:1, cl 1:1, cl all 1:1 und cl med 1:1 gemittelt, ergibt sich Abbildung 30.

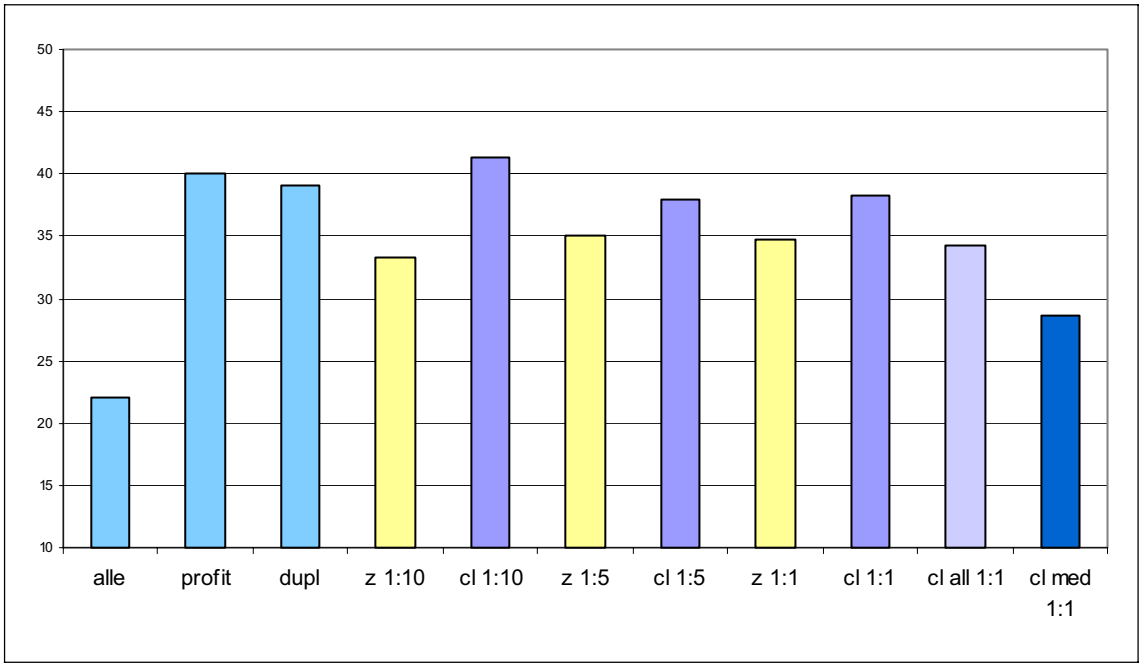


Abbildung 30: Gemittelte Gains-Chart Ergebnisse beim 10%-Wert auf Basis der Testdaten bei Entscheidungsbaumverfahren
Quelle: Eigene Darstellung

Durch die Mittelwertbildung können Ausreißer abgefedert werden. Es zeigt sich in dieser Darstellung nochmals, dass die Modellgüte bei den Zufallsauswahlmethoden jeweils deutlich schlechter ist als bei den entsprechenden clusteranalysegestützten Auswahlmethoden (ausgenommen die Methoden „cl all 1:1“ und „cl med 1:1“).

Abbildung 31 zeigt die Modellergebnisse beim 80%-Wert aus dem Gains-Chart. Auch hier fällt die deutlich größere Streuung bei den Modellvarianten mit Zufallsauswahl auf. Außerdem sind die clusteranalysegestützten Methoden auch in der Modellgüte insgesamt deutlich überlegen. Hier erzielen vor allem die Methoden „profit“ und „cl 1:1“ sehr gute Ergebnisse.

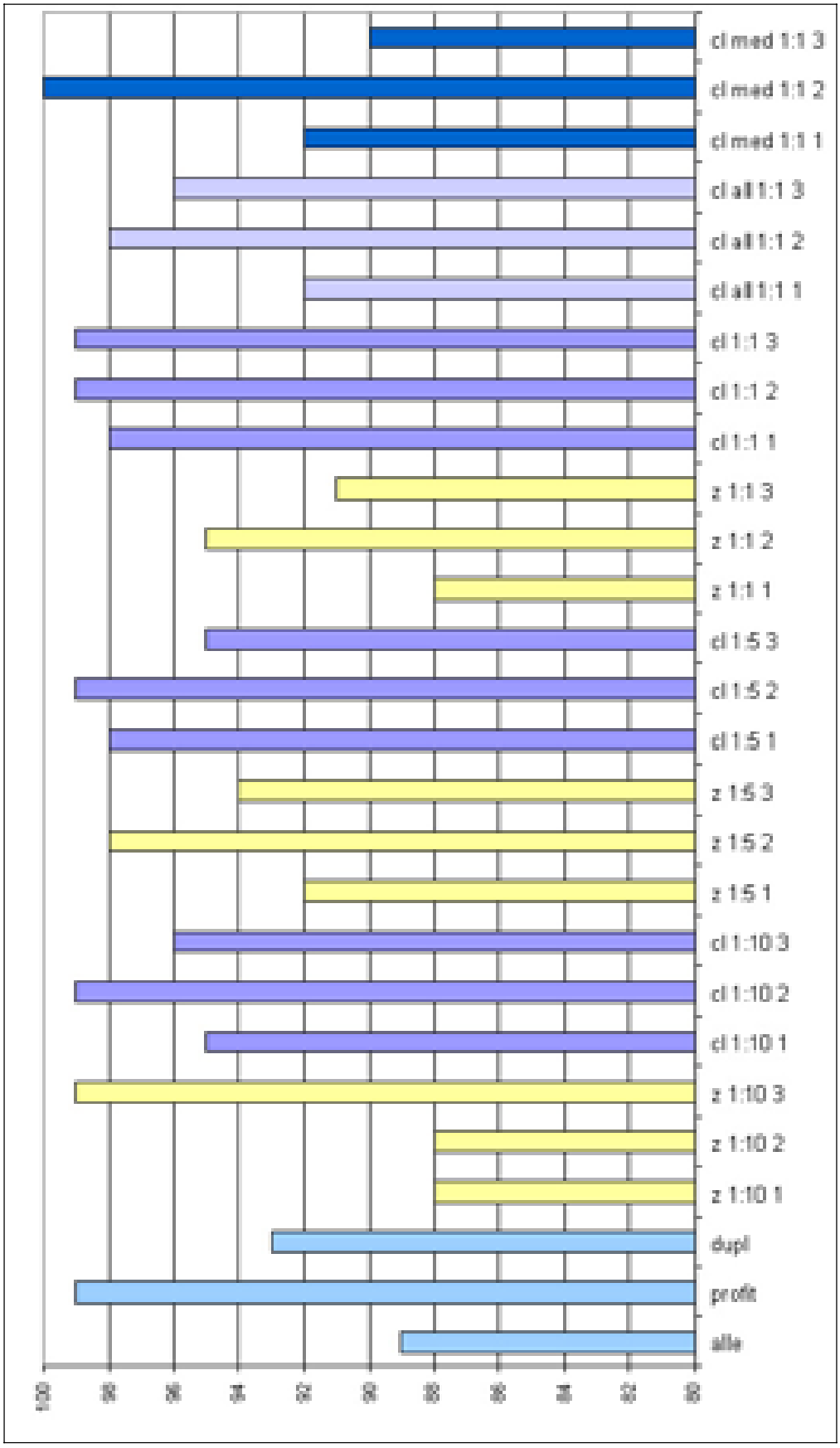


Abbildung 31: Gains-Chart Ergebnisse beim 80%-Wert auf Basis der Testdaten bei Entscheidungsbaumverfahren
Quelle: Eigene Darstellung

Zum Abschluss zeigt nachfolgende Tabelle 16, bei wie vielen Modellvarianten die einzelnen Variablen verwendet werden. Die Entscheidungsbaumverfahren nutzen einige Variablen überhaupt nicht bzw. nur selten. Am häufigsten werden die Variablen Nr. 3, 5, 6, 7, 8, 22, 53, 57, 58 und 59 herangezogen.

Variable	Nr. 3	Nr. 5	Nr. 6	Nr. 7	Nr. 8	Nr. 9	Nr. 12	Nr. 14	
Anzahl	19	14	11	16	26	5	0	0	
Variable	Nr. 15	Nr. 16	Nr. 18	Nr. 20	Nr. 22	Nr. 24	Nr. 45	Nr. 46	
Anzahl	0	2	0	0	18	7	4	0	
Variable	Nr. 47	Nr. 48	Nr. 49	Nr. 50	Nr. 51	Nr. 53	Nr. 57	Nr. 58	Nr. 59
Anzahl	1	4	0	0	2	21	16	12	27

Tabelle 16: Verwendete Variablen bei den Entscheidungsbaumvarianten

4.7 Zusammenfassung

Insgesamt bieten Entscheidungsbäume eine sehr einfach nachvollziehbare und verständliche Visualisierung. Es können Daten mit unterschiedlichem Skalenniveau verwendet werden und das Verfahren stellt keinerlei Ansprüche an die Verteilung der unabhängigen Variablen, da es sich um ein nicht-parametrisches Verfahren handelt.

Die empirischen Ergebnisse belegen die Instabilität von Entscheidungsbäumen. Bei den Modellen, die mit unterschiedlichen Datenausschnitten arbeiten, wird zum Teil eine deutlich unterschiedliche Modellgüte erzielt. Durch die Bildung eines Durchschnittwertes der Modellgüte werden Ausreißer in beide Richtungen abgemildert, allerdings zu Lasten der Interpretation, da das Ergebnis aus möglicherweise deutlich unterschiedlichen Entscheidungsbäumen entsteht.

Je nach Zielsetzung, das heißt, ob nur die besten Adressen ausgewählt oder nur die schlechtesten Adressen eingespart werden sollen, empfehlen sich unterschiedliche Methoden. Bei beiden Varianten erzielen die Methoden „profit“ und „cl 1:1“ sehr gute Ergebnisse.

Sollen nur die besten 10% der Objekte angeschrieben werden, erreichen die besten Entscheidungsbaumverfahren („profit“, „cl 1:10“, „cl 1:1“) bereits ca. 40% aller Besteller.
Sollen die schlechtesten 20% weggelassen werden, so werden mit den besten Entscheidungsbaumverfahren („profit“, „cl 1:1“) bereits ca. 99% der Besteller erreicht.

In dieser Studie ist aufgrund der Ergebnisse subjektiv die Methode „profit“ zu empfehlen, da hier nur ein Entscheidungsbaum zur Auswahl steht und dabei sehr gute Ergebnisse erzielt werden. Wählt man die Methode „cl 1:1“, so muss der Anwender zusätzlich entscheiden, ob er das beste der drei Modellvarianten zum Scoring auswählt oder alle drei verwendet und den Mittelwert der Scoringwerte zur endgültigen Auswahl heranzieht. Die erste Variante bei Wahl der Methode „cl 1:1“ ist dabei als eher aggressiv zu umschreiben, während die zweite Variante eher sicherheitsorientiert ist, da damit möglicherweise nicht das beste, jedoch auch nicht das schlechteste Ergebnis erzielt wird, allerdings zu Lasten der Interpretation.

Insgesamt lässt sich zu den Entscheidungsbäumen sagen, dass hier die Vorgehensweise „alle“ nicht geeignet ist. Die Methode „profit“ erzielt sehr gute Ergebnisse. Das Duplizieren erreicht relativ gute Ergebnisse, ist dem Downsizing jedoch nicht überlegen und aufgrund der längeren Rechenzeit und der stark steigenden Baumkomplexität ebenfalls nicht zu empfehlen. Zu diesem Schluss kommen auch Japkowicz (2000, S. 116) und Ling/Li (1998, S. 77) in ihren Versuchen. Die clusteranalysegestützten Methoden (ausgenommen die Methoden „cl all 1:1“ und „cl med 1:1“) erzielen bessere Ergebnisse als die Methoden mit reiner Zufallsauswahl und sind diesen somit überlegen. Chan/Stolfo (1998, S. 167), Ling/Li (1998, S. 77) und Weiss/Provost (2001, S. 5) erzielen in ihren Versuchen in bezug auf den Anteil der 1-Klasse in der Zielvariable ähnliche Ergebnisse und empfehlen eine 50/50 Verteilung. Dies kann in dieser Arbeit nicht uneingeschränkt bestätigt werden, da bei einer 1:5 Verteilung beispielsweise ebenfalls sehr gute Ergebnisse erzielt werden.

5. Responseoptimierung mit der binären logistischen Regression

Bei der **logistischen Regression** wird die Wahrscheinlichkeit der Zugehörigkeit zu einer Klasse als nichtlineare Funktion einer oder mehrerer Variablen betrachtet.
Die unabhängigen Variablen können bei der logistischen Regression sowohl nominales als auch metrisches Skalenniveau aufweisen. Die nominalen Variablen werden dabei binärisiert. Die Zielvariable oder abhängige Variable kann binär (binär logistische Regression) oder nominal polytom (multinomial logistische Regression) sein. Die logistische Regression zählt zu der Klasse der Generalisierten Linearen Modelle (GLM). Im folgenden wird der Ansatz der binären logistischen Regression vorgestellt.

5.1 Das Logit-Modell

Im linearen Regressionsansatz geht man von folgender linearen Beziehung aus:

$$y_i \mid \eta_0 2 \eta_1 x_{i1} 2 \eta_2 x_{i2} 2 \ldots 2 \eta_j x_{ij} 2 \ldots 2 \eta_p x_{ip} 2 u_i \qquad \&i \mid 1,\ldots,n$$

mit :

y_i | Ausprägung der abhängigen Variablen bei Objekt i,

x_{ij} | Ausprägung der j - ten unabhängigen Variablen bei Objekt i mit $j \mid 1,\ldots, p$,

η_j | Koeffizient der unabhängigen Variablen mit $j \mid 1,\ldots, p$,

η_0 | Absolutglied,

u_i | Residuum bei Objekt i .

Dieser Ausdruck impliziert prinzipiell eine Spannweite der Zielvariablen von $4 \leftarrow$ bis $2 \leftarrow$ bei metrischem Skalenniveau. Bei einer binären Zielvariablen muss der Wert jedoch $\subset \{0,1\}$ sein, das heißt es muss eine funktionale Spezifikation vorliegen, die diese Tatsache berücksichtigt.

In Anlehnung an Cox (1970, S. 18f.) lautet die logistische Funktion:

$$p(y_i \mid 1) \mid \frac{1}{1 2 e^{4(\eta_0 2 \eta_1 x_{i1} 2 \ldots 2 \eta_j x_{ij} 2 \ldots 2 \eta_k x_{ik})}} \qquad \subset [0,1].$$

Um die Form eines linearen Ansatzes zu erhalten, wird zunächst die Wahrscheinlichkeit für das Eintreten eines Ereignisses durch die Wahrscheinlichkeit für das Nicht-Eintreten dieses Ereignisses dividiert:

$$\frac{p(y_i \mid 1)}{1 - p(y_i \mid 1)} \mid \text{Odds.}$$

Nach dieser Transformation liegt der Wertebereich der modifizierten abhängigen Variablen zwischen 0 und + ←. Um den Wertebereich auch auf den negativen Bereich zu erweitern, werden die Odds logarithmiert, so dass die transformierte Zielvariable nun Werte von - ←bis + ←annimmt. Die logarithmierten Odds werden als Logit der Wahrscheinlichkeit $p(y_i=1)$ oder Logit-Transformation bezeichnet (Hosmer/Lemeshow, 1989, S. 6):

$$z_i \mid \ln\left\{\frac{p(y_i \mid 1)}{1 - p(y_i \mid 1)}\right\} \mid \beta_0 + \beta_1 x_{i1} + \ldots + \beta_j x_{ij} + \ldots + \beta_p x_{ip} + u_i \quad .$$

Für den Fall einer einzigen metrischen unabhängigen Variablen nimmt die logistische Funktion einen S-förmig gekrümmten Verlauf an (siehe Abbildung 32). Dem Gedanken der klaren Klassenteilung wird insofern Rechnung getragen, als der Großteil des Definitionsbereichs entweder sehr kleinen oder sehr großen Wahrscheinlichkeiten entspricht (Bausch, 1991, S. 92).

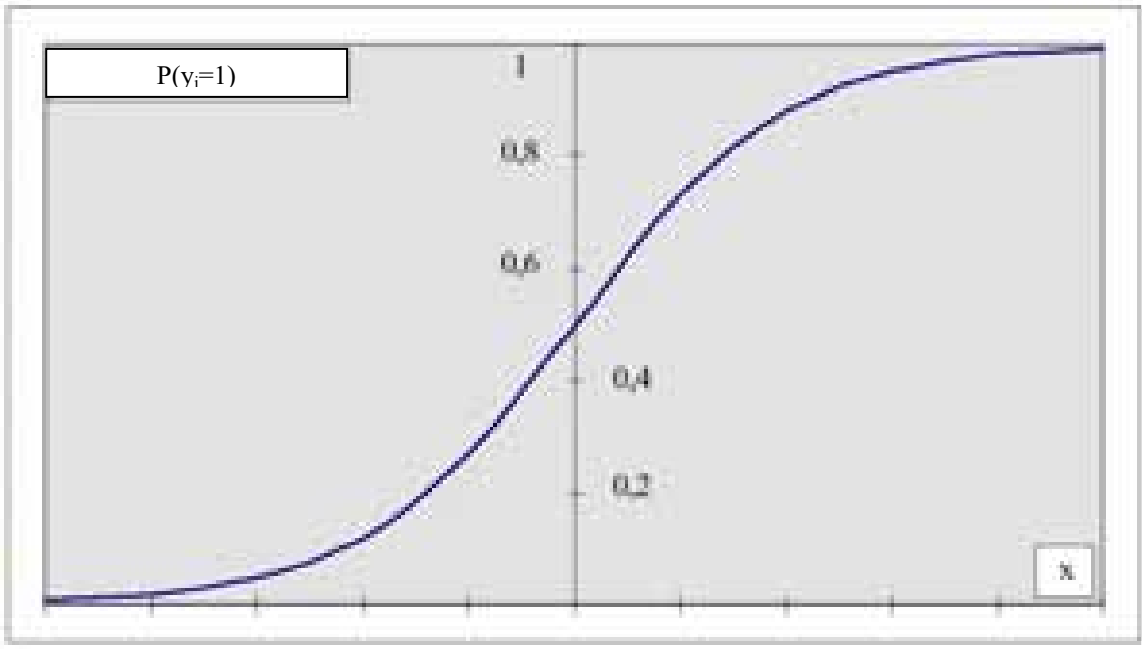


Abbildung 32: Logistische Funktion im eindimensionalen Fall
Quelle: Eigene Darstellung in Anlehnung an Bausch (1991, S. 92)

Probleme bei der logistischen Regression verursachen gemäß Aldrich/Nelson (1984, S. 49):

- Multikollinearität der unabhängigen Variablen und
- Autokorrelation der abhängigen Variablen.

Um die Multikollinearität auszuschließen, müssen die unabhängigen Variablen auf ihre Unabhängigkeit untereinander untersucht werden. Derartige Bemühungen wurden bereits in der Datenvorverarbeitung (siehe S. 42f.) durchgeführt.

Zum Test auf Autokorrelation, das heißt auf Unabhängigkeit der Zielvariablen, wird der Durbin-Watson Test verwendet (Backhaus et al., 2000, S. 41). Allerdings tritt Autokorrelation vor allem bei Zeitreihen auf, so dass diese Voraussetzung in der vorgelegten Studie nicht weiter berücksichtigt werden muss (Hippner/Rupp, 2001, S. 696).

Als eine sinnvolle Untergrenze zur Durchführung einer logistischen Regression gelten 50 Beobachtungen (Backhaus et al., 2000, S. 137). Erst ab Stichproben von $n > 100$ erweisen sich die Schätzergebnisse einer Logit-Analyse denjenigen anderer Verfahren überlegen (McFadden, 1974, S. 123). Diese Bedingung ist in dieser Arbeit ebenfalls erfüllt.

Nachfolgend ein Vergleich von logistischer Regression und linearer Regression:

	Logistische Regression	Lineare Regression
Ansatz	$y_i \subset \{0,1\}$	y_i metrisch
	$z_i = \ln\left(\frac{p(y_i = 1)}{1 - p(y_i = 1)}\right) = x_i^T \beta + u_i$	$y_i = x_i^T \eta + u_i$
Allgemein	y hängt nichtlinear von x_{ij} ab	y hängt linear von x_{ij} ab
Stochastische Spezifikationen	$E(u_i)=0$	$E(u_i)=0$
	$E(y_i) = p(y_i = 1) = \frac{1}{1 + e^{-x_i^T \beta}}$	$E(y_i) = x_i^T \beta$
	$Var(y_i) = Var(u_i) =$ $p(y_i = 1)\,[1 - p(y_i = 1)]$ (Heteroskedastizität)	$Var(y_i) = Var(u_i) = \omega^2$ (Homoskedastizität)
	$Cov(u_i,u_j)=0$ für i,j (keine Autokorrelation)	$Cov(u_i,u_j)=0$ für i,j (keine Autokorrelation)
	Keine Multikollinearität	Keine Multikollinearität

Tabelle 17: Vergleich von logistischer und linearer Regression
Quelle: In Anlehnung an Urban (1998, S. 47)

5.2 Parameterschätzung, Tests auf Signifikanz und Aufnahme von Variablen

Zur Schätzung der Parameter oder Koeffizienten wird bei der logistischen Regression die Maximum-Likelihood-Methode (ML-Methode) angewandt. Das Maximum-Likelihood-Prinzip besteht bei Vorliegen der Beobachtungen y_i darin, die unbekannten Parameter η_j so zu schätzen, dass für die Parameterschätzwerte $\hat{\eta}_j$ die Wahrscheinlichkeit des Eintretens der Beobachtung y_i maximiert wird (Bamberg/Baur, 1998, S. 153; Fahrmeir/Hamerle, 1996, S. 59). Im ersten Schritt muss eine Schätzfunktion, die sogenannte Likelihood-Funktion gebildet werden. Sie beschreibt die Wahrscheinlichkeit der Beobachtung y_i in Abhängigkeit von den unbekannten Parametern und geht davon aus, dass die n Beobachtungen mit „Ereignis tritt ein“ und „Ereignis tritt nicht ein“ Realisationen unabhängig identisch Bernoulli-verteiler Zufallsvariablen sind (Bausch, 1991, S. 93). Es werden diejenigen Parameter ermittelt, die diese Funktion maximieren.

Die Likelihood-Funktion (L) ergibt sich aus dem Produkt der Wahrscheinlichkeiten der Zuordnung zur jeweils korrekten Gruppe; aus Gründen der leichteren Rechenbarkeit wird häufig die logarithmierte Likelihood-Funktion (LL) verwendet (Hosmer/ Lemeshow, 1989, S. 9):

$$L(\beta_0, \beta_1, ..., \beta_p) = \prod_{y_i = 1} p(y_i = 1) \prod_{y_i = 0} (1 - p(y_i = 1)),$$

$$LL(\beta_0, \beta_1, ..., \beta_p) = \ln[L(\beta_0, \beta_1, ..., \beta_p)] = \sum_{y_i = 1} \ln[p(y_i = 1)] + \sum_{y_i = 0} \ln(1 - p(y_i = 1)).$$

Die Schätzwerte für die $\hat{\eta}_j$ ergeben sich gemäß dem ML-Prinzip aus der Maximierung der Log-Likelihood-Funktion. Dies geschieht durch partielles Ableiten nach den einzelnen Parametern und Nullsetzen der Ableitungen.

Nachdem die Parameter geschätzt wurden, müssen diese auf ihre statistische Signifikanz untersucht werden. Dazu wird die Nullhypothese H_0 formuliert, die besagt, dass die j-te Variable keinen Einfluss auf die abhängige Variable ausübt:

H_0: $\hat{\eta}_j = 0$,

H_1: $\hat{\eta}_j \neq 0$.

Zur Überprüfung dieser Hypothesen werden der Likelihood-Ratio-Test, der Wald-Test oder der Score-Test verwendet.

Der **Likelihood-Ratio-Test** basiert auf dem Vergleich zweier Modelle. Es wird betrachtet, ob das Modell mit allen unabhängigen Variablen (L_1) die abhängige Variable besser erklärt als das Modell ohne die zu testende unabhängige Variable x_j (L_0). Dazu werden die beobachteten Werte der abhängigen Variablen mit den jeweils in beiden Fällen prognostizierten Werten verglichen (Hosmer/Lemeshow, 1989, S. 12).
Die Teststatistik lautet:

$$G = -2\ln\frac{|L_0|}{|L_1|} = -2(|LL_0| - |LL_1|)$$

mit L_0 = Likelihood des Modells unter H_0,

L_1 = Likelihood des Modells unter H_1,

LL_0 = Loglikelihood des Modells unter H_0,

LL_1 = Loglikelihood des Modells unter H_1.

Soll beispielsweise der Einfluss von x_1 getestet werden, wird die Nullhypothese H_0: $\hat{\eta}_1 = 0$ geprüft. Die Teststatistik berechnet sich wie folgt:

$$G = -2\left(LL\left(\hat{\eta}_0^{H_0}, 0, \hat{\eta}_2^{H_0}, ..., \hat{\eta}_p^{H_0}\right) - LL\left(\hat{\eta}_0^{H_1}, \hat{\eta}_1^{H_1}, \hat{\eta}_2^{H_1}, ..., \hat{\eta}_p^{H_1}\right)\right).$$

Die Teststatistik G ist asymptotisch θ^2-verteilt und die Anzahl der Freiheitsgrade berechnet sich aus der Differenz der Anzahl der geschätzten Parameter in beiden Modellen, in diesem Fall also p-(p-1) = 1.

Die Nullhypothese wird abgelehnt, falls der Wert der Teststatistik in den Ablehnungsbereich fällt. Der Ablehnungsbereich lautet:
$G > \theta^2_{df;1-\alpha}$.

Zur Beurteilung der Signifikanz wird der sogenannte p-Wert herangezogen.

Im univariaten Fall wird das Modell mit der Variablen und das Modell, das nur den konstanten Term enthält, verglichen. Im multivariaten Fall kann entweder jede einzelne Variable auf Signifikanz getestet werden oder aber es wird ein Test auf alle p Parameter durchgeführt. Dabei wird das Modell, das alle Variablen enthält, mit dem Modell, das nur den konstanten Term enthält, verglichen (Hosmer/Lemeshow, 1989, S. 31). Die Teststatistik G ist dann asymptotisch θ^2-verteilt mit p Freiheitsgraden. Wird die Nullhypothese abgelehnt, bedeutet dies, dass zumindest einer, möglicherweise auch alle p+1 Parameter signifikant von Null verschieden sind. Um die einzelnen Parameter auf ihre statistische Signifikanz hin zu untersuchen, kann beispielsweise auch der Wald-Test verwendet werden.

Der **Wald-Test** eignet sich dazu, die einzelnen Parameter auf ihre Signifikanz hin zu überprüfen. Die Wald-Statistik, die auch im SAS EM verwendet wird, berechnet sich durch den Vergleich des ML-Schätzers $\hat{\eta}_j$ mit seiner Standardabweichung Var($\hat{\eta}_j$).

Die Wald-Teststatistik lautet:

$$W \mid \frac{(\hat{\eta}_j)^2}{Var(\hat{\eta}_j)}.$$

Sie besitzt unter H_0 die gleiche asymptotische θ^2-Verteilung wie die Likelihood-Ratio-Teststatistik (Hauck/Donner, 1977, S. 851).

Ein Nachteil der Wald-Teststatistik besteht darin, dass bei großen absoluten Parameterschätzwerten und großen Standardabweichungen ein zu kleiner Wert der Teststatistik ausgegeben wird und dies dann fälschlicherweise nicht zur Ablehnung von H_0 führen kann (Hauck/Donner, 1977, S. 853).

Bei Verwendung des **Score-Tests** müssen nicht in jedem Schritt die Maximum-Likelihood-Schätzer $\hat{\eta}_j$ berechnet werden. Dies bedeutet bei großen Datenmengen und zahlreichen Variablen einen deutlich geringeren Rechenaufwand. Der Score-Test basiert auf der bedingten Verteilung der Ableitungen des Loglikelihood, da davon ausgegangen wird, dass die betreffende Variable noch nicht in das Modell aufgenommen wurde (Kleinbaum/Kupper/Chambless, 1982, S. 513). Es wird dabei die erste Ableitung der Loglikelihood nach $\hat{\eta}_j$ unter der Bedingung „x_j nicht im Modell“ berechnet. Die Score-Statistik berechnet sich aus der quadrierten Ableitung multipliziert mit der bedingten Varianz des betreffenden Parameters. Die Berechnung der Score-Statistik im multivariaten Fall erfordert für die einzelnen Parameter einige Matrizenoperationen, auf die hier nicht näher eingegangen werden soll. Eine ausführliche Darstellung findet sich in Cox/Hinkley (1974, S. 324 ff.).

Umfasst die Datenmatrix viele Variablen, so erscheint es sinnvoll, eine Methode zur Entscheidung über Aufnahme bzw. Nichtaufnahme der entsprechenden Variablen in das Modell anzuwenden. Das Ziel ist, diejenigen Variablen zu selektieren, die zu dem „besten“ Modell führen (Hosmer/Lemeshow, 1989, S. 82). Es gibt beispielsweise die Methoden Einschluss, Rückwärtsselektion, Vorwärtsselektion und schrittweise Selektion. Hosmer/Lemeshow (1989, S. 83) geben zu bedenken, dass, je mehr Variablen in das

Modell aufgenommen werden, die Standardabweichungen umso größer werden. Dies kann zu einer Art Overfitting führen.

Beim erzwungenen Einschluss werden alle vorher definierten Variablen in das Modell miteinbezogen.
Die Rückwärtsselektion beginnt mit allen Variablen im Modell und schließt so lange iterativ Variablen aus, bis alle im Modell verbliebenen Variablen gemäß dem Wald-Test oder Likelihood-Ratio-Test signifikanten Einfluss auf die Zielvariable besitzen (Agresti/Finlay, 1997, S. 529).
Bei der Vorwärtsselektion beginnt das Modell ohne Variablen. Es werden so lange iterativ Variablen hinzugefügt, bis keine der verbliebenen Variablen gemäß dem Score-Test, Wald-Test oder Likelihood-Ratio-Test einen signifikanten Einfluss auf die Zielvariable ausübt (Agresti/Finlay, 1997, S. 531).
Die schrittweise Selektion besteht aus einer Kombination von Vorwärts- und Rückwärtsselektion. Die Variablen werden bei dieser Methode schrittweise bezüglich ihrer Aufnahme in das Modell bzw. ihres Ausschlusses aus dem Modell überprüft. Es wird im folgenden die Vorwärtsselektion mit anschließendem Test auf Rückwärtseliminierung vorgestellt (Hosmer/Lemeshow, 1989, S. 106ff.). Begonnen wird wiederum mit einem konstanten Modell, also ohne unabhängige Variablen. Die schrittweise Selektion im SAS EM basiert auf einem Algorithmus, der alle Variablen anhand ihrer statistischen Signifikanz gemäß dem Score-Test auf ihre Wichtigkeit prüft und im Anschluss die Variable, die eine festgelegte Signifikanzschranke erfüllt, ins Modell aufnimmt. Danach werden die Variablen, die sich im Modell befinden, mit Hilfe des Wald-Tests auf ihren Ausschluss untersucht. Falls eine unabhängige Variable eine bestimmte festgelegte Signifikanzschranke nicht erfüllt, wird sie eliminiert. Der Prozess endet, wenn keine Variable ein definiertes Signifikanzniveau bei der Vorwärtsselektion unterschreitet, die zuletzt aufgenommene Variable wieder entfernt wird oder eine vorher festgelegte Maximalanzahl an Variablen erreicht wurde.

Es entsteht bei dieser Vorgehensweise eine Reihe von Modellen mit unterschiedlicher Anzahl an Variablen. Zur endgültigen Modellauswahl gibt es im SAS EM die Möglichkeit, aus verschiedenen Kriterien zu wählen, beispielsweise das Akaike Information Criterion (AIC) oder das Schwarz-Bayesian-Criterion (SBC). Zur Wahl des Schwellenwertes für die Aufnahme und den Verbleib von Variablen gibt es dabei unterschiedliche

Auffassungen. Während Hosmer/Lemeshow (1989, S. 108) im Allgemeinen für die Aufnahme von Variablen ein Signifikanzniveau von 0,15 bis 0,30 vorschlagen, schließen sich Shtatland/Cain/Barton (2001, S. 4) nur bei Verwendung des AIC dieser Meinung an, empfehlen jedoch bei Verwendung des SBC für ein erklärendes Modell einen Wert von 0,001 bis 0,05. In dieser Arbeit wird das SBC mit einem Signifikanzniveau für Aufnahme und Verbleib von jeweils 0,05 verwendet. Dabei wird die Aufnahme zusätzlicher Variablen wie folgt bestraft:

$$SBC = (n)\ln\left(\frac{SSE}{n}\right) + (p)\ln(n)$$

mit SSE: Quadrierter Gesamtfehler,
n: Anzahl Beobachtungen,
p: Anzahl Variablen im Modell.

Die endgültige Entscheidung über die Aufnahme bestimmter Variablen ins Modell sollte jedoch sowohl aufgrund statistischer als auch aufgrund inhaltlicher Überlegungen getroffen werden (Hosmer/Lemeshow, 1989, S. 91).

5.3 Goodness-of-fit - Tests

Nach dem Abschluss der Modellbildung soll nun die Effektivität des Modells im Hinblick auf die Prognose der abhängigen Variablen untersucht werden. Dieser Test wird auch „Goodness-of-fit“ genannt (Hosmer/Lemeshow, 1989, S. 135). Im folgenden werden die Devianz und die Pearson θ^2-Statistik vorgestellt.

Die Likelihood-Ratio-Statistik wird häufig auch als **Devianz** bezeichnet (Agresti/Finlay, 1997, S. 594). Dabei wird der Log-Likelihood-Wert eines saturierten Modells, das in der Schätzung eine perfekte Replikation der beobachteten y-Verteilung liefert, bei 0 liegen. Alle Werte größer 0 indizieren also Abweichungen von einem Modell mit perfekter Anpassungsgüte. Die Devianz D ist definiert als:

$$D = -2\{LL_1 \quad .$$

Die Devianz ist asymptotisch θ^2-verteilt mit (n-p) Freiheitsgraden.

Alternativ wird häufig auch die **Pearson θ^2-Statistik** G angegeben. Sie basiert auf den Residuen, die sich aus der Differenz zwischen beobachteten und geschätzten Eintrittswahrscheinlichkeiten ergibt:

$\text{Residuum}_i = y_i - p(y_i \mid 1)$.

Die Teststatistik berechnet sich nun aus dem Verhältnis von Residuenquadratsumme zu deren Varianz:

$$G = \sum_{i=1}^{n} \frac{(y_i - p(y_i \mid 1))^2}{p(y_i \mid 1)(1 - p(y_i \mid 1))} .$$

Dadurch werden die quadrierten Residuen mit ihrer Varianz gewichtet. Es wird dabei berücksichtigt, dass y bei kleiner Varianz leichter zu schätzen sein sollte als bei großer Varianz und deshalb ein möglicher Schätzfehler bei kleiner Varianz besonders gravierend einzuschätzen ist.

Es gibt noch einige weitere Goodness-of-fit Tests, beispielsweise den Hosmer/Lemeshow-Test (Hosmer/Lemeshow, 1989, S. 140ff.), auf die hier jedoch nicht weiter eingegangen wird.

Allerdings sind die Goodness-of-fit Tests nicht unumstritten. Allison (1999, S. 56) zeigt, dass es Fälle gibt, bei welchen das Modell die Zielvariable sehr gut prognostiziert, allerdings gemäß der Devianz sehr schlecht sein müsste bzw. aufgrund der Goodness-of-fit Werte ein sehr gutes Modell vorliegt, das jedoch ausgesprochen schlechte Prognosefähigkeit besitzt. Deshalb wird in dieser Studie zum Vergleich der Modelle der Gains-Chart verwendet, da hier die Prognosegüte aller Modelle einheitlich vergleichbar auf Basis der Testdaten gezeigt wird.

5.4 Empirische Ergebnisse

In dieser Studie wird eine binäre logistische Regression mit schrittweiser Variablenselektion durchgeführt. Dabei besteht das Modell zu Beginn nur aus dem absoluten Glied, dann werden schrittweise Variablen aufgenommen. Somit entsteht eine Folge von Modellen mit unterschiedlicher Anzahl an unabhängigen Variablen, die auf Basis der Trainingsdaten gebildet werden. Im SAS EM wird das Modell ausgewählt, das bei den Va-

lidierungsdaten den geringsten Wert beim SBC aufweist. Bei der Methode Profitmatrix (siehe S. 59) wird das Modell ausgewählt, das auf Basis der Validierungsdaten den größten Profit bringt. Die Profitmatrix hat nur auf die Modellauswahl, nicht auf die Parameterschätzung Einfluss.

Tabelle 18 zeigt die Ergebnisse der verschiedenen Varianten (siehe Tabelle 14, S. 106) der logistischen Regression auf Basis der Testdaten.

	10%	20%	40%	80%
alle	36	47	64	93
profit	32	50	68	92
dupl	34	47	58	93
z 1:10 1	27	43	60	97
z 1:10 2	36	44	64	97
z 1:10 3	29	44	67	97
cl 1:10 1	37	44	59	88
cl 1:10 2	31	49	63	90
cl 1:10 3	36	44	59	90
z 1:5 1	32	47	67	93
z 1:5 2	34	45	69	93
z 1:5 3	30	47	65	93
cl 1:5 1	36	45	68	97
cl 1:5 2	34	45	71	95
cl 1:5 3	34	46	65	93
z 1:1 1	27	47	68	92
z 1:1 2	31	36	61	92
z 1:1 3	32	44	68	96
cl 1:1 1	32	49	69	95
cl 1:1 2	31	50	66	95
cl 1:1 3	32	52	69	98
cl all 1:1 1	28	45	70	98
cl all 1:1 2	21	35	60	94
cl all 1:1 3	36	46	66	92
cl med 1:1 1	36	51	68	93
cl med 1:1 2	32	43	68	95
cl med 1:1 3	27	51	65	98

Tabelle 18: Gains-Chart Ergebnisse auf Basis der Testdaten bei der logistischen Regression

Die Ergebnisse zeigen, dass Unterschiede zwischen den Vorgehensweisen bestehen. Allerdings ist die Schwankungsbreite zwischen den unterschiedlichen Vorgehensweisen nicht so groß wie bei den Entscheidungsbäumen. Es gibt auch keine Methode, die ex-

trem schlecht abschneidet. Dies zeigt, dass die logistische Regression erwartungsgemäß relativ robust bei niedrigen Responsequoten ist, da es sich um ein statistisches Verfahren und keine Heuristik handelt. Im Durchschnitt erreichen die Modelle bei 10% der Testdaten bereits ca. 32% der Besteller. Beim 20%-Wert werden 46% der Besteller erreicht, beim 40% Wert 65%. Beim 80%-Wert liegt der Mittelwert bei 94%.

Die Methode „alle“ erreicht beim 10%-Wert einen sehr guten Wert. Bei den restlichen Werten werden eher durchschnittliche Ergebnisse erzielt.

Die Methode „profit“ erreicht bei dem 20%-Wert ein überdurchschnittliches Ergebnis. Bei den restlichen Werten liegen die Ergebnisse in etwa im Durchschnitt.

Die Duplizierung der Besteller liefert bis auf den 40%-Wert durchschnittliche Ergebnisse. Dort schneidet das Verfahren stark unterdurchschnittlich ab. Aufgrund der größeren Datenmenge wird außerdem deutlich mehr Rechenzeit benötigt.

Bei der Methode „z 1:10“ werden drei Modellvarianten erzeugt. Beim 10%-Wert schwanken die Ergebnisse von sehr gut bis schlecht, während beim 80%-Wert nur sehr gute Ergebnisse vorliegen.

Im direkten Vergleich dazu sind bei der Methode „cl 1:10“ die Ergebnisse beim 10%-Wert sehr gut, bei dem 40%- und 80%-Wert dagegen deutlich unterdurchschnittlich.

Bei der Methode „z 1:5“ sind die Ergebnisse insgesamt durchschnittlich und beim 40%-Wert etwas überdurchschnittlich. Hier ist die Schwankungsbreite zwischen den einzelnen Modellen im Vergleich relativ gering.

Die Methode „cl 1:5“ erreicht ein deutlich überdurchschnittliches Ergebnis beim 10%- und 40%-Wert. Insgesamt werden bei allen Messpunkten bereits gute Ergebnisse erreicht.

Die Methode „z 1:1“ erzielt insgesamt unterdurchschnittliche Ergebnisse. Auffällig ist die starke Schwankung beim 20%-Wert.

Die Methode „cl 1:1“ erzielt insgesamt deutlich überdurchschnittliche Ergebnisse. Beim 20%-Wert werden hier die besten Ergebnisse erreicht. Im Vergleich zu „z 1:1“ gibt es außerdem insgesamt gesehen eine deutlich geringere Streuung, so dass hier der zusätzliche Vorverarbeitungsschritt Vorteile bringt.

Die Methode „cl all 1:1“ weist sehr starke Schwankungen und durchschnittliche bis deutlich unterdurchschnittliche Ergebnisse auf.

Die Methode „cl med 1:1“ erzielt insgesamt durchschnittliche bis gute Ergebnisse. Auffällig ist bei diesem Modell, dass die Auswahl eher zentral gelegener Objekte bei der clusteranalysegestützten Stichprobenziehung im Vergleich mit „cl 1:1“ nicht zu stabileren Modellen führt. Auch die Ergebnisse bei den Entscheidungsbäumen zeigen ein ähnliches Verhalten.

Anderson (1972, S. 25) zeigt, dass exogen geschichtete Stichproben bei der logistischen Regression lediglich einen Einfluss auf die Schätzung des absoluten Gliedes haben, nicht jedoch auf die restlichen Parameter. Unter exogen geschichteten Stichproben werden dabei Stichproben verstanden, bei welchen sich der Anteil an 1-Klasse und 0-Klasse von deren Anteil in der Grundgesamtheit unterscheidet (Bonne/Arminger, 2001). Dies zeigt sich auch bei den Ergebnissen in dieser Studie, da die Methode „alle“ ähnliche Ergebnisse erzielt wie die Methoden, bei welchen der Anteil der 1-Klasse in der Zielvariablen geändert wurde.

Gray (1976, S. 2268) kommt in seiner Studie zu dem Ergebnis, dass statistische Verfahren beispielsweise Probleme haben, eine selten auftretende Klasse zu erkennen. Harrell (2001, S. 61) empfiehlt in diesem Zusammenhang, dass bei der binären logistischen Regression das Verhältnis von 1-Klasse zu 0-Klasse in etwa bei 1:3 liegen sollte. Beide Einschätzungen können in dieser Studie nicht bestätigt werden, da zwischen den untersuchten Methoden keine extremen Unterschiede auftreten.

Abbildung 33 zeigt in einem Säulendiagramm das Ergebnis aller Modelle beim 10%-Wert im Gains-Chart. Die Abbildung verdeutlicht, dass die Schwankung bei den Zufallsauswahlmethoden etwas größer ist als bei den clusteranalysegestützten Auswahlmethoden, ausgenommen die Methoden „cl all 1:1“ und „cl med 1:1“. Die beiden Metho-

den „cl all 1:1“ und „cl med 1:1“ zeigen im Vergleich zu „cl 1:1“ eine deutlich stärkere Streuung. Insgesamt liegen die Ergebnisse zwischen 21% und 36%. Bei den Entscheidungsbaumverfahren werden hier Werte zwischen 22% und 44% erreicht, allerdings ist die Streuung vor allem bei den Zufallsauswahlmethoden größer (siehe Abbildung 29, S. 110).

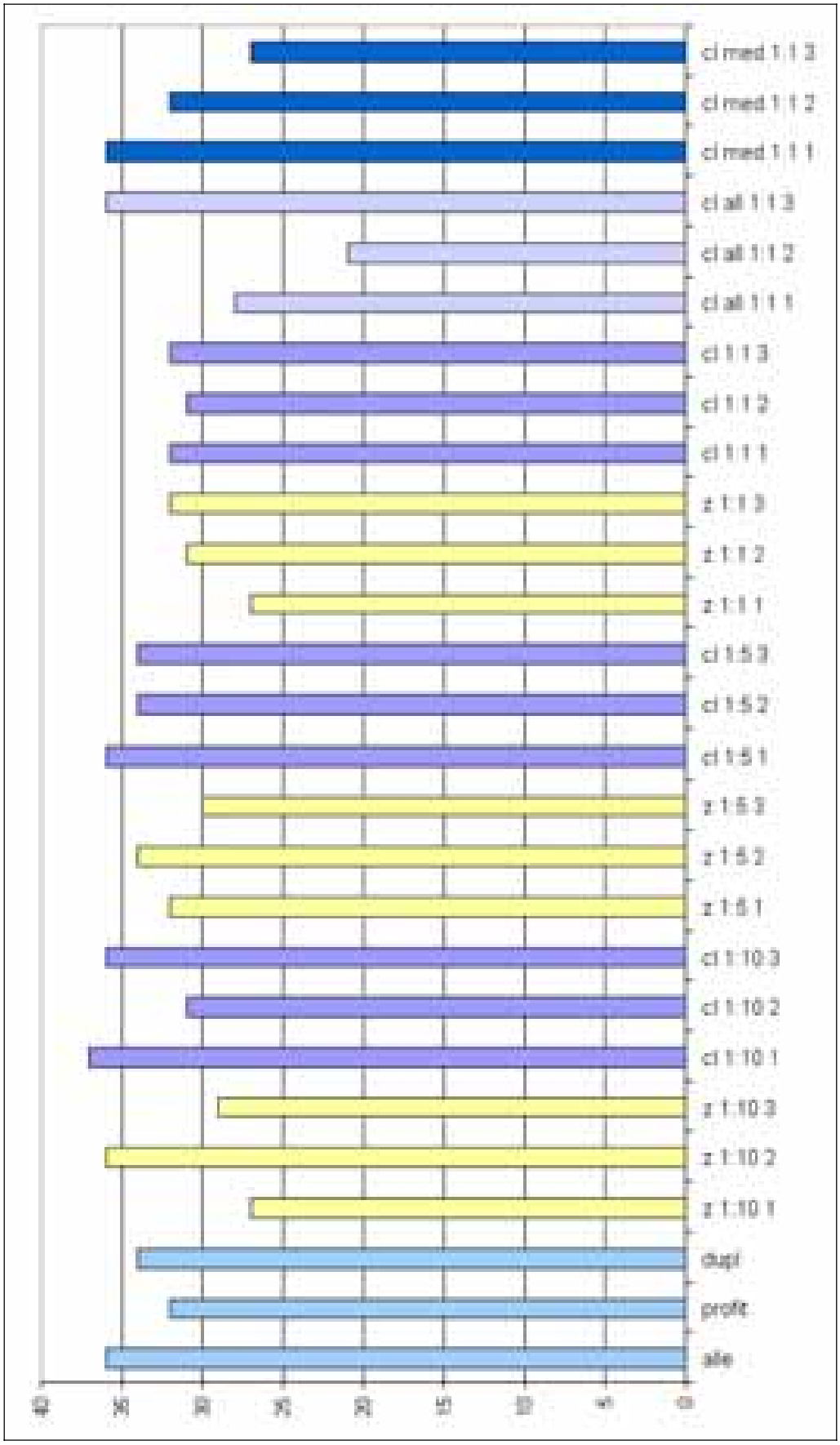


Abbildung 33: Gains-Chart Ergebnisse beim 10%-Wert auf Basis der Testdaten bei der logistischen Regression
Quelle: Eigene Darstellung

Abbildung 34 zeigt die gemittelten Ergebnisse der Methoden „z 1:10“, „cl 1:10“, „z 1:5“, „cl 1:5“, „z 1:1“, „cl 1:1“, „cl all 1:1“ und „cl med 1:1“. Aus Abbildung 34 erkennt man, dass die logistische Regression bei allen Methoden beim 10%-Wert relativ stabile Ergebnisse liefert. Es zeigt sich allerdings auch, dass die clusteranalysegestützten Methoden bessere Ergebnisse als die auf einer Zufallsauswahl basierenden Methoden erzielen. Dies ist auch bei den Entscheidungsbaumverfahren zu beobachten (siehe Abbildung 30, S. 111). Auffällig beim Vergleich mit den Entscheidungsbaumverfahren ist vor allem der Unterschied bei der Methode „alle“, die bei der logistischen Regression ein sehr gutes Ergebnis erzielt.

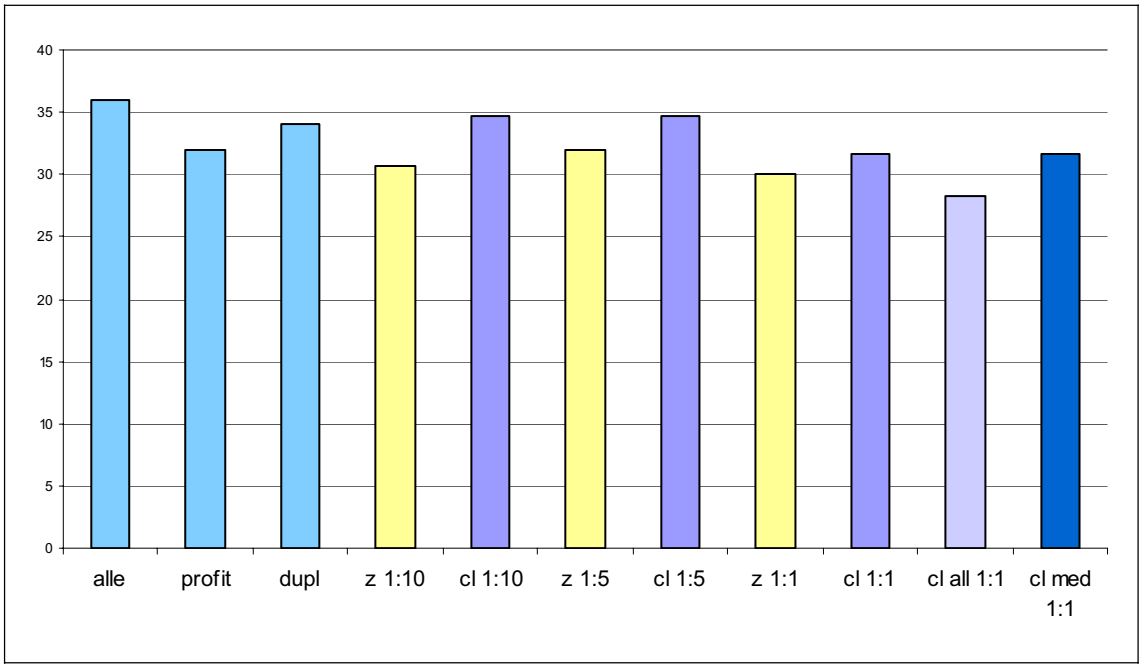


Abbildung 34: Gemittelte Gains-Chart Ergebnisse beim 10%-Wert auf Basis der Testdaten bei der logistischen Regression
Quelle: Eigene Darstellung

Abbildung 35 zeigt die Performance der einzelnen Methoden beim 80%-Wert. Auffällig ist hier, dass die Methode „z 1:10“ sehr stabile und gute Ergebnisse erzielt. Ähnlich gute Ergebnisse werden nur mit der Methode „cl 1:1“ erreicht. Die restlichen Methoden schneiden relativ ähnlich ab. Im Vergleich zu den Entscheidungsbaumverfahren streuen die Ergebnisse hier deutlich weniger. Allerdings erzielen sieben Entscheidungsbäume hier bereits Werte von über 98% (siehe Abbildung 31, S. 112).

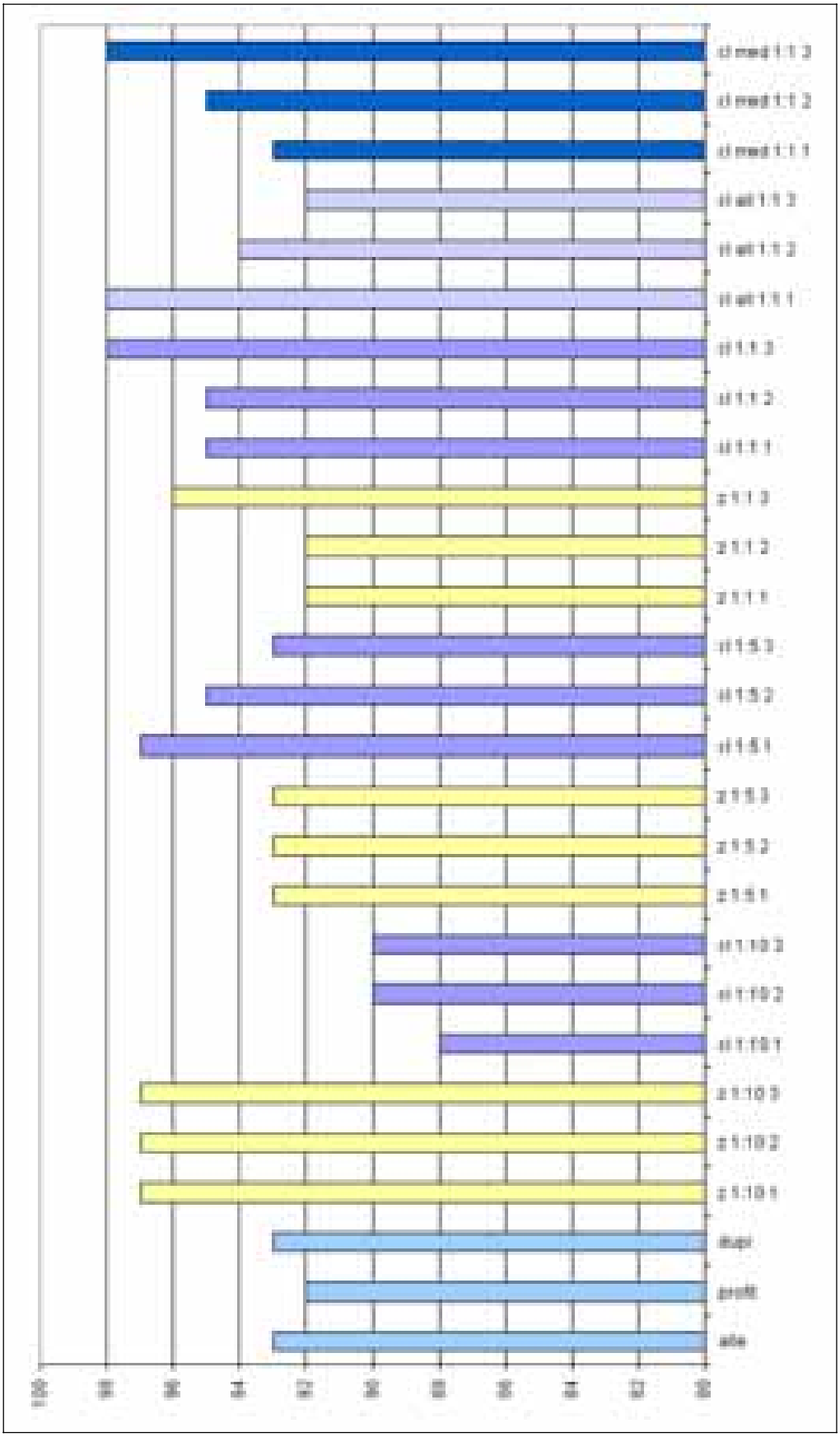


Abbildung 35: Gains-Chart Ergebnisse beim 80%-Wert auf Basis der Testdaten bei der logistischen Regression
Quelle: Eigene Darstellung

Nachfolgende Tabelle 19 zeigt wiederum, bei wie vielen Modellvarianten der logistischen Regression die einzelnen Variablen verwendet werden. Im Vergleich zu den Entscheidungsbaumverfahren werden hier deutlich mehr Variablen überhaupt nicht bzw. nur selten genutzt. Zu den wichtigsten Variablen bei der logistischen Regression scheinen die Nr. 8, 22, 24, 53 und 57 zu zählen.

Variable	Nr. 3	Nr. 5	Nr. 6	Nr. 7	Nr. 8	Nr. 9	Nr. 12	Nr. 14	
Anzahl	7	0	2	1	24	3	0	1	
Variable	Nr. 15	Nr. 16	Nr. 18	Nr. 20	Nr. 22	Nr. 24	Nr. 45	Nr. 46	
Anzahl	0	2	3	0	27	13	3	0	
Variable	Nr. 47	Nr. 48	Nr. 49	Nr. 50	Nr. 51	Nr. 53	Nr. 57	Nr. 58	Nr. 59
Anzahl	3	4	0	0	0	27	11	2	0

Tabelle 19: Verwendete Variablen bei den Varianten der logistischen Regression

5.5 Zusammenfassung

Wie bei Entscheidungsbäumen entstehen bei der Verwendung der logistischen Regression interpretierbare Modelle. Aufgrund relativ schwacher Modellvoraussetzungen (siehe S. 117), kann dieses Verfahren häufig angewandt werden.

Die empirischen Ergebnisse zeigen, dass die Schwankungsbreite bei der logistischen Regression insgesamt gesehen relativ gering ist. Gerade im Vergleich mit den Entscheidungsbaumverfahren ist sie bei niedrigen Responsequoten deutlich robuster. Allerdings sind die Ergebnisse bei den Entscheidungsbaumverfahren etwas besser.

Insgesamt lässt sich zur logistischen Regression sagen, dass hier keine Vorgehensweise grundsätzlich ausgeschlossen werden kann. Das Duplizieren erzielt wie bei den Entscheidungsbäumen relativ gute Ergebnisse, jedoch bei deutlich höherer Rechenzeit. Die clusteranalysegestützten Methoden erzielen beim 10%-Wert bessere Ergebnisse als die Methoden mit reiner Zufallsauswahl und sind somit zu bevorzugen. Die Methode „profit“ erzielt eher durchschnittliche Ergebnisse.

Je nach Zielsetzung, das heißt, ob nur die besten Adressen ausgewählt oder nur die schlechtesten Adressen eingespart werden sollen, empfehlen sich unterschiedliche Methoden. Bei beiden Varianten erzielt die Methode „cl 1:5“ sehr gute Ergebnisse, wobei der Anwender entscheiden muss, ob er das beste Modell wählt oder den Mittelwert aus mehreren Modellen.

Sollen nur die besten 10% der Objekte angeschrieben werden, empfehlen sich bei der logistischen Regression die Methoden „alle“, „cl 1:10“ und „cl 1:5“. Dabei werden bereits 36% aller Besteller erreicht. Entscheidungsbaumverfahren erreichen bei diesem Wert 40% aller Besteller. Subjektiv wäre hier die Methode „alle“ zu empfehlen.
Sollen die schlechtesten 20% weggelassen werden, so werden mit der Methoden „z 1:10“ 97% der Besteller erreicht. Alternative Methoden sind: „cl 1:1“, „cl 1:5“ und „cl med 1:1“. Die Entscheidungsbaumverfahren erzielen hier bereits Werte von 99%. In diesem Fall wäre hier subjektiv die Methode „cl 1:1“ zu empfehlen, da im Allgemeinen bei der clusteranalysegestützten Zufallsauswahl stabilere Modelle zu erwarten sind.

6. Responseoptimierung mit Künstlichen Neuronalen Netzen

Mit Hilfe **Künstlicher Neuronaler Netze** (KNN) sollen intelligente und kognitive Handlungen auf Computern nachgebildet werden (Dorffner, 1991, S. 3). KNN imitieren den Ablauf der Informationsverarbeitung im menschlichen Gehirn und können dadurch die Problemlösungsfähigkeit von Computern verbessern. Die Anfänge der Entwicklung von KNN gehen bis auf McCullagh/Pitts (1943), und Hebb (1949) zurück. Das erste KNN, das Perceptron, stellte Rosenblatt (1958) vor.

KNN sind informationsverarbeitende Systeme, die aus einfachen Recheneinheiten, den **Neuronen**, bestehen. Diese Neuronen senden sich Informationen über gerichtete Verbindungen zu. KNN sind nicht-linear, können mit einer großen Anzahl von Variablen arbeiten und sind lernfähig, das heißt es muss keine Annahme über die Form des Zusammenhangs zwischen unabhängigen und abhängiger Variablen gemacht werden (Wiedemann/Buckler, 2001, S. 45). KNN können beliebige Funktionen approximieren. Die Daten sollen im Idealfall keine fehlenden Werte und können beliebiges Skalenniveau aufweisen.

Bei der Verwendung von KNN hat sich im Vergleich zur Statistik eine unterschiedliche Terminologie herausgebildet. In Tabelle 20 werden einige Begriffe aus dem Umfeld der KNN und der Statistik gegenübergestellt.

KNN	Statistik
Eingabevariablen oder Inputs	Unabhängige Variablen
Ausgabevariablen oder Outputs	Prognostizierte Werte
Zielwert, Trainingswert	Abhängige Variable
Fehler	Residuen
Training, Lernen, Adaption	Schätzung
Fehlerfunktion	Schätzkriterium
Gewichte	Parameterschätzer

Tabelle 20: Terminologie KNN / Statistik
Quelle: In Anlehnung an Sarle (1994, S. 2)

6.1 Varianten und Architektur von KNN

Im Allgemeinen kann bei KNN zwischen überwachtem und unüberwachtem Lernen unterschieden werden (siehe Abbildung 36). Die Zielsetzung beim überwachten Lernen ist die Strukturabbildung, beim unüberwachten Lernen ist es die Strukturentdeckung (Poddig/Sidorovitch, 2001, S. 366f.).

Zum **unüberwachten Lernen** zählen die Self Organizing Feature Maps (siehe S. 34), die, beispielsweise analog zur Clusteranalyse, unbekannte Strukturen in der zu analysierenden Datenbasis ohne a priori Informationen entdecken und visualisieren können (Kohonen, 1997).
Beim **überwachten Lernen** wird in Abbildung 36 unterschieden in Funktionsapproximation, Klassifikation und Assoziativspeicher, wobei die Netze, die unter Funktionsapproximation geführt werden, ebenso zur Klassifikation verwendet werden können.

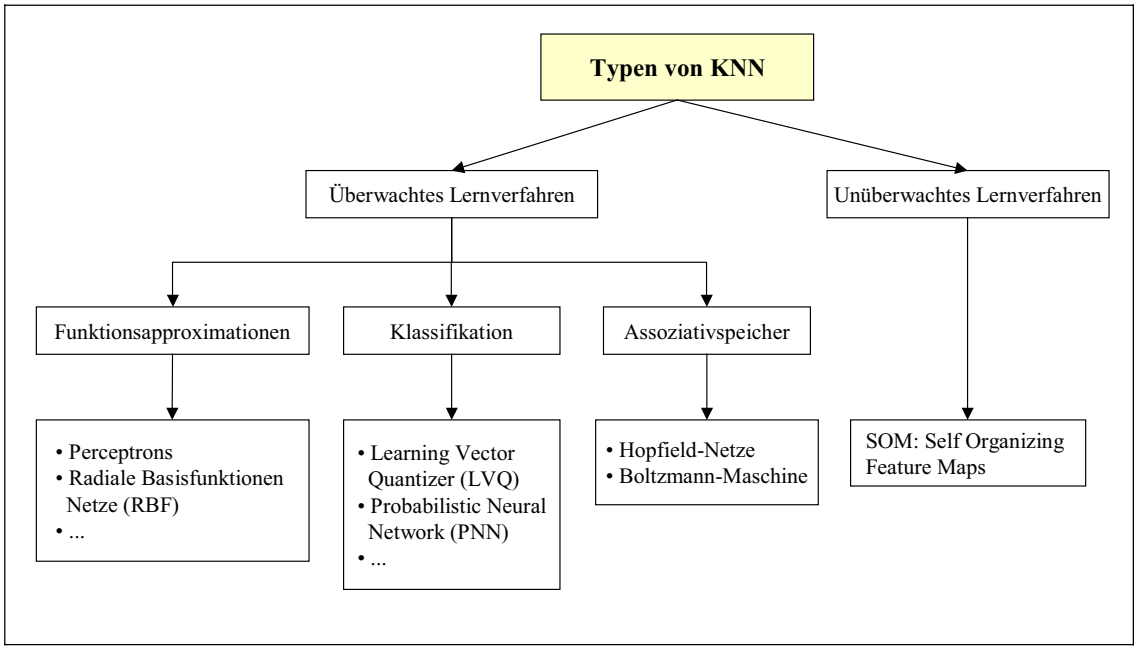


Abbildung 36: Typen von KNN
Quelle: In Anlehnung an Poddig/Sidorovitch (2001, S. 367)

In einem Assoziativspeicher kann eine Menge von Fragemuster-Anwort-Tupeln abgelegt werden. Wird eine Frage an den Assoziativspeicher gestellt, wird mit einer gewissen Fehlertoleranz die entsprechende Antwort gegeben (Poddig/Sidorovitch, 2001, S. 365). Ein Beispiel wäre die Personenerkennung anhand des Gesichts bei der Zugangs-

kontrolle. In diese Kategorie fallen beispielsweise die Hopfield-Netze (Hopfield, 1982 und 1984) und die Boltzman Maschine (Ackley et al., 1985).
Zur Klassifikation aus Abbildung 36 zählen beispielsweise der LVQ (Kohonen, 1986) und das PNN (Specht, 1990). Auch die Perceptrons oder die RBF (Neuneier/Tresp, 1994) können durch eine geringfügige Variation innerhalb ihrer Architektur zur Klassifikation verwendet werden.
Unter Funktionsapproximation fallen KNN, die besonders geeignet sind, Modelle der Art y=f(x) zu schätzen.

Diese Arbeit orientiert sich in der weiteren Beschreibung am Perceptron, das zum Bereich der überwachten Lernmodelle zählt.

Trotz vieler Unterschiede in der Architektur der verschiedenen Typen von KNN gibt es doch einige gemeinsame Grundelemente. KNN können als Baukastensysteme verstanden werden. Ihre Bausteine, die **Neuronen**, sind die Grundrechenoperatoren. Durch eine geeignete Kombination dieser Neuronen sind prinzipiell alle logischen Zusammenhänge darstellbar. Die Neuronen sind über Ein- und Ausgabekanäle miteinander verbunden. Die geeignete Zusammenschaltung der Neuronen ergibt ein KNN.
In Abbildung 37 wird der grundlegende Aufbau eines Neurons symbolisch dargestellt.

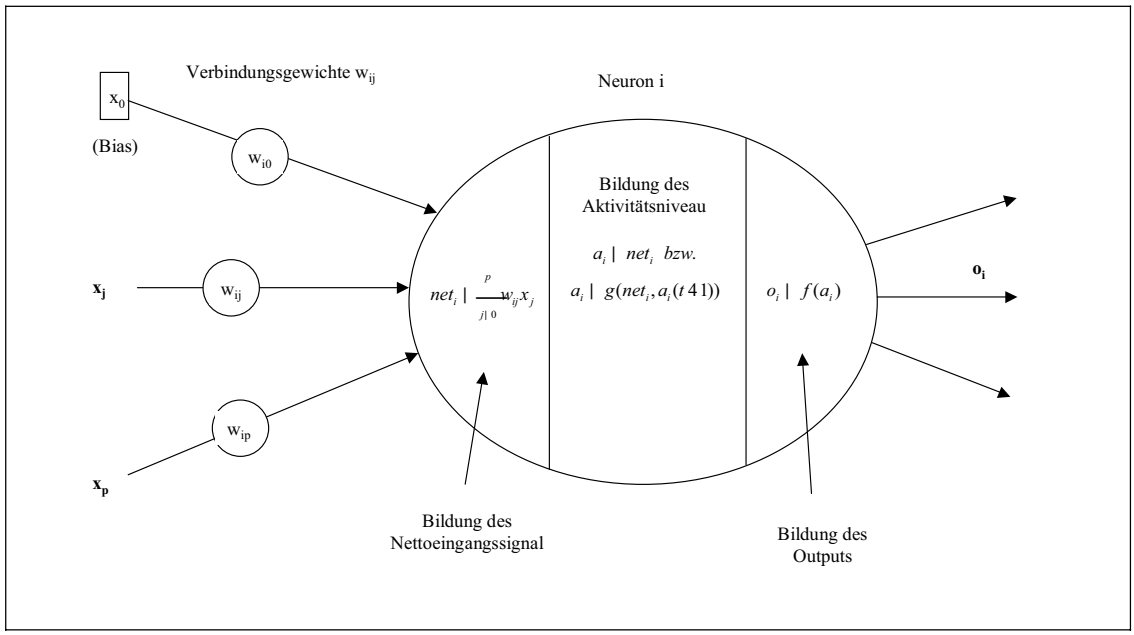


Abbildung 37: Das Modell eines Neurons
Quelle: In Anlehnung an Poddig/Sidorovitch (2001, S. 370)

Die Verarbeitung in diesem Neuron findet in drei Schritten statt. Zuerst werden die Eingangswerte, hier die Variablen $x_{j,}$ über eine sogenannte Inputfunktion zum Nettoinput net_i verarbeitet. Die Komponente x_0 wird nicht immer verwendet. Dieses Signal wird auch „Bias“ genannt, mit dem die Modellierung eines Absolutglieds bei der Berechnung des Nettoinputs ermöglicht wird. Die Verbindungsgewichte w_{ij} geben an, mit welcher Stärke das i-te Neuron von der j-ten Schicht Signale empfängt. In dieser Grafik entspricht diese den unabhängigen Variablen x_j, es könnten jedoch auch die Ausgangssignale einer vorgeschalteten Neuronenschicht sein. Die Funktion zur Bildung des Nettoeingangssignals ist meist eine gewichtete Aufsummierung der Eingangswerte:

$$net_i = \sum_{j=0}^{p} w_{ij} x_j .$$

Anschließend wird für das Neuron *i* ein **Aktivitätszustand** a_i bestimmt, wobei dieser häufig dem Inputwert entspricht: $a_i = net_i$.

Es kann jedoch auch sein, dass eine **Aktivierungsfunktion** g zur Bestimmung des Aktivitätsniveaus verwendet wird, die beispielsweise zusätzlich von dem Wert des Aktivitätsniveaus im zeitlich vorhergehenden Verarbeitungsschritt t-1 abhängt. Dann berechnet sich das Aktivitätsniveau a_i aus:

$$a_i = g(net_i, a_i(t-1)).$$

Im letzten Schritt wird der Ausgabewert o_i des Neurons berechnet, der an die nächste Schicht weitergegeben wird. Dieser ergibt sich aus einer normalerweise nicht-linearen Funktion, die den Aktivitätszustand transformiert. Die **Outputfunktion** f lautet: $o_i = f(a_i) = f(net_i)$.

Tabelle 21 zeigt einen Auszug alternativer Outputfunktionen.

Beschreibung	Funktion	Wertebereich
Linear	$f(net_i) = net_i$	$f(net_i) \subset R$
Lineare Schwellenwert-funktion	$f(net_i) = \begin{cases} 1, & net_i > q \\ net_i, & net_i \subset [q;q] \\ -1, & net_i < q \end{cases}$	$f(net_i) \subset [-1;1]$ q, q vorgegeben
Sigmoide Schwellenwert-funktion	$f(net_i) = \frac{1}{1 + e^{-4net_i}}$	$f(net_i) \subset [0;1]$
Tangens Hyperbolicus	$f(net_i) = \tanh(net_i) = \frac{e^{net_i} - e^{-4net_i}}{e^{neti} + e^{net_i}}$	$f(net_i) \subset [-1;1]$

Tabelle 21: Übersicht bekannter Outputfunktionen
Quelle: In Anlehnung an Alex (1998, S. 88)

Die Art und Weise, wie die Neuronen eines Neuronalen Netzes miteinander verbunden sind, wird Aufbau des Netzwerks oder **Topologie** bezeichnet.

Durch die Verknüpfung vieler Neuronen zu einem Netzwerk entsteht die besondere Leistungsfähigkeit der KNN. Meist sind in einem KNN nicht alle Neuronen miteinander verbunden, es werden vielmehr Schichten oder Layer gebildet, die aus einem oder mehreren Neuronen bestehen können (Kruse, 1991, S. 27). Es werden drei Gruppen von Schichten unterschieden: Eingabeschicht oder Input Layer, verborgene Schicht oder Hidden Layer und Ausgabeschicht oder Output Layer (siehe Abbildung 38).

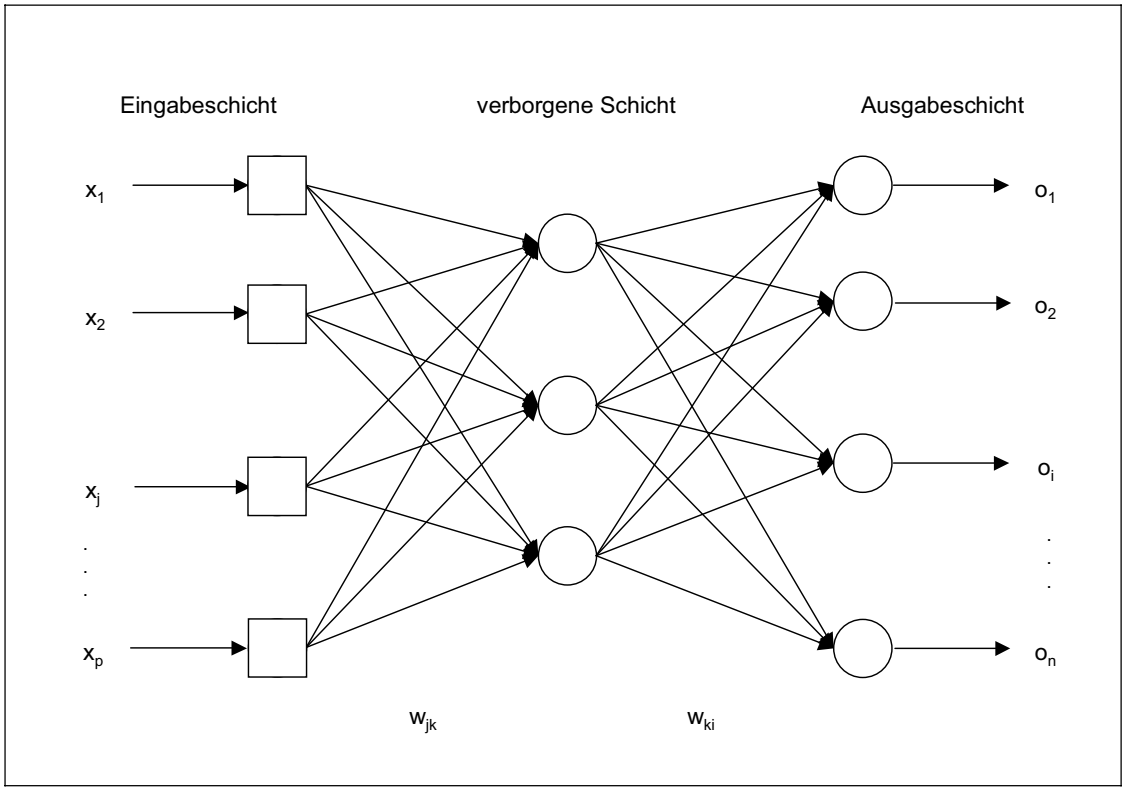


Abbildung 38: Schematischer Aufbau eines KNN
Quelle: Eigene Darstellung

Die Eingabeschicht besteht aus Eingabeneuronen, die die Werte der unabhängigen Variablen aufnehmen und an das System weiterleiten. Sie besitzen selbst keine Informationsverarbeitungsfähigkeit, deshalb sind sie in Abbildung 38 durch Rechtecke und nicht durch Kreise gekennzeichnet. Die verborgene Schicht befindet sich vollständig innerhalb des Netzwerks. Sowohl die Eingabe als auch die Ausgabe erfolgt nur von bzw. an Neuronen innerhalb des Netzwerks. Die Leistungsfähigkeit des KNN ist stark von der Art der verborgenen Schicht, der Anzahl an verborgenen Schichten und der Verknüpfungsstruktur abhängig. Die Ausgabeschicht gibt die Informationen des Systems wieder an die Außenwelt ab.

Abbildung 39 zeigt einige Netzwerkarchitekturen.

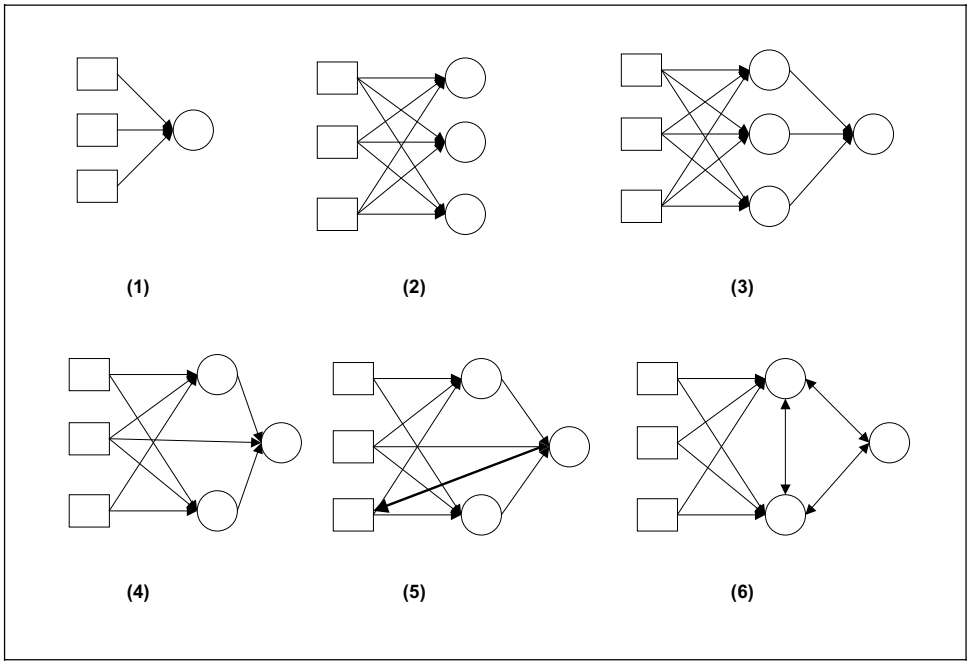


Abbildung 39: Auswahl einiger Netzwerkarchitekturen
Quelle: In Anlehnung an Poddig/Sidorovitch (2001, S. 374)

Die Eingabeschicht ist wiederum jeweils als Rechteck gekennzeichnet, während die Neuronen der verborgenen Schicht bzw. Ausgabeschicht als Kreise dargestellt werden. Bild (1) zeigt ein einschichtiges Netzwerk mit einer Eingabeschicht und einem Outputneuron. Anzumerken ist hier, dass manche Autoren die Inputschicht bei der Zählung der Schichten mit berücksichtigen. Bild (2) zeigt ebenfalls ein einschichtiges Netzwerk mit mehreren Outputneuronen, während Bild (3) ein zweischichtiges Netzwerk mit einer verborgenen und einer Ausgabeschicht darstellt. Bei Bild (4) sind auch Direktverbindungen der Eingabeschicht zur Ausgabeschicht zugelassen. Bild (5) und Bild (6) lassen Rückkopplungen innerhalb des Netzwerks zu, wobei es bei Bild (5) keine Rückkopplung während der Verarbeitung, sondern erst bei der Verarbeitung des nächsten Inputs gibt. Auf diese Weise lässt sich eine zeitliche Dynamik durch das Netzwerk induzieren, weshalb sie häufig als zeitrekurrente Netze bezeichnet werden (Poddig/Sidorovitch, 2001, S. 374f.). Bei Bild (6) hingegen findet die Rückkopplung während der Verarbeitung statt, wie beispielsweise bei Hopfield-Netzen und der Boltzman-Maschine.

Die Netzwerke in den Bildern (1) bis (4) stellen vorwärtsgerichtete oder feed-forward Netze dar, das heißt die Verarbeitung des Inputs erfolgt nur in eine Richtung, während die Bilder (5) und (6) Rückkopplung zulassen und deshalb feedbackward Netze genannt werden.

Im folgenden wird zunächst das **Single-Layer Perceptron** beschrieben.
Hierbei handelt es sich um ein einschichtiges Netz (siehe Abbildung 39, Bild (1) und (2)). Angenommen es gibt nur ein Ausgabeneuron und die Outputfunktion f ist die Identitätsfunktion, dann berechnet sich dessen Ausgabewert nach:

$$o_i = \sum_{j=0}^{p} w_{ij} x_j \quad \forall i.$$

Diese Form entspricht der linearen Regression. Die logistische Regression könnte beispielsweise mit der Outputfunktion „Sigmoider Schwellenwert“ (siehe S. 139) nachgebildet werden.

Wird dem Single-Layer Perceptron mindestens eine verborgenen Schicht hinzugefügt, entsteht ein **Multi-Layer Perceptron** oder MLP (siehe Abbildung 39, Bild (3) und (4)). Voraussetzung ist dabei allerdings, dass zumindest eine Outputfunktion der Neuronen nicht-linear ist. Wären alle Outputfunktionen linear, so könnte dieses Netzwerk auch durch ein Single-Layer Perceptron dargestellt werden (Poddig, 1992, S. 238f.). Die Verknüpfung vieler Neuronen mit nicht-linearen Ausgabefunktionen über mehrere Schichten repräsentiert aus mathematischer Sicht äußerst komplexe Zusammenhänge zwischen Eingabe- und Ausgabewerten. Daraus resultiert die Eigenschaft, nichtlineare Zusammenhänge in den Daten approximieren zu können (Hruschka, 1991, S. 219). Kolmogorov konnte 1957 nachweisen, dass jede stetige Funktion durch KNN mit einer nichtlinearen Outputfunktion beliebig genau dargestellt werden kann (Rojas, 1996, S. 205f.). Allgemein gilt: Je komplexer die zu modellierenden nichtlinearen Zusammenhänge in den Daten, umso mehr verdeckte Neuronen sind erforderlich (Poddig/Sidorovitch, 2001, S. 380).

Im folgenden wird die **Parameterschätzung** eines MLP beschrieben. Dieser Prozess wird auch häufig „Lernen“ genannt. Steinbuch (1971, S. 134) definiert den Begriff „Lernen“ in bezug auf technische Systeme folgendermaßen: „Lernen eines Systems besteht darin, dass es entsprechend früheren Erfolgen oder Misserfolgen (Erfahrungen) das interne Modell der Außenwelt verbessert“. Übertragen auf KNN kann die Modellverbesserung als Wissenszuwachs bzw. Lernen gesehen werden. Das Wissen eines KNN steckt in den Verbindungen zwischen den Neuronen, so dass Lernen die Veränderung der Verbindungsgewichte bewirkt. Das MLP wird häufig auch als Backpropagation-Netzwerk bezeichnet, da das Backpropagation-Lernfahren und damit die schichtwei-

se Rückführung (Backpropagation) von Ausgabefehlern durch das Netz verwendet wird (Rojas, 1996, S. 149). Rumelhart/Hinton/Williams (1986) haben den Backpropagation-Algorithmus entwickelt, allerdings lässt er sich auf eine Arbeit von Werbos (1974) zurückführen. Ziel des Backpropagation-Algorithmus ist die Schätzung der Verbindungsgewichte w_{ij} zwischen den Neuronen eines MLP bei Minimierung einer Fehlerfunktion E. Der quadratischen Gesamtfehler zwischen den Sollausgabewerten der Zielvariablen y_i und den mit dem Verfahren ermittelten Werten o_i über alle Objekte i soll dabei minimiert werden:

$$E \mid \sum_i E_i \mid \sum_i \frac{1}{2}(y_i - o_i)^2 \quad \Downarrow \min .$$

Die variablen Größen dabei sind die Verbindungsgewichte w_{ij}. Somit ergibt sich die Fehlerfunktion E in Abhängigkeit der Verbindungsgewichte w_{ij} mit $W=(w_{ij})_{i,j}$:
$E(W|y_i,o_i)$, die minimiert werden soll (Hilbert/Dittmar, 1997, S. 30).

Die Optimierung erfolgt iterativ in zwei Phasen. Zuerst werden die Eingabewerte x_j über die Eingabeschicht in das Netz eingebracht und vorwärts durch dieses propagiert, um die Netzausgabewerte o_i zu erhalten. Im ersten Durchlauf werden die Verbindungsgewichte beliebig vorgegeben. Dann wird der Fehlerfunktionswert nach obiger Formel berechnet und zur Gewichtsadaption verwendet, wobei gilt, dass eine Erhöhung des Fehlers E zu einer höheren Gewichtsveränderung $\div w_{ij}$ führt (Rumelhart/Hinton/Williams, 1986, S. 322ff.). Die Fehlerfunktion E kann als n-dimensionales Hypergebirge in einem n-dimensionalen Gewichtsraum aufgefasst werden, wobei n der Anzahl an Verbindungsgewichten in dem Netz entspricht. Abbildung 40 zeigt das Fehlergebirge E für den zweidimensionalen Fall, also den Zusammenhang zwischen dem Fehler E und den Verbindungsgewichten w_{11} und w_{12}.

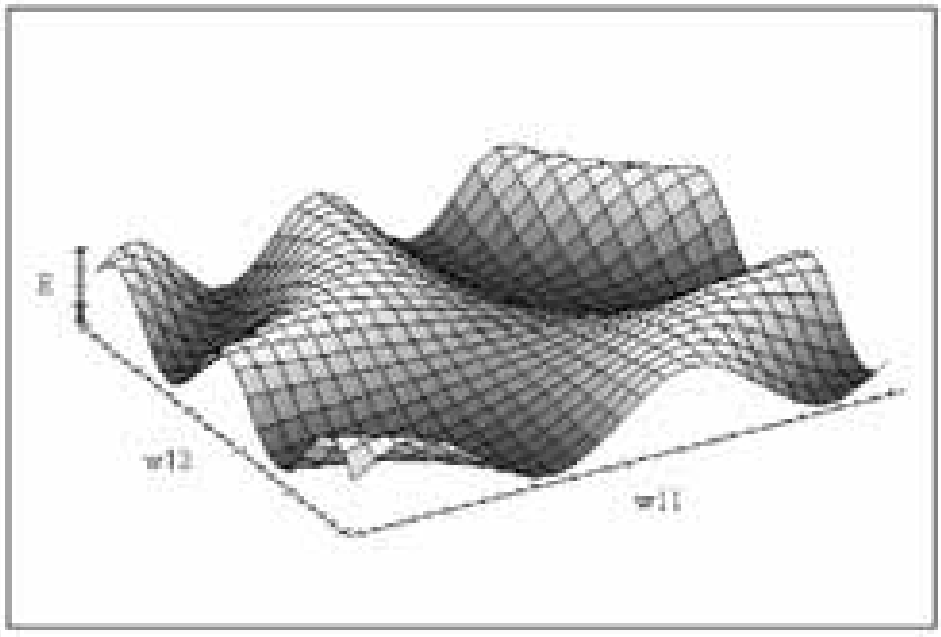


Abbildung 40: Fehlergebirge E im zweidimensionalen Fall
Quelle: Urban (1998, S. 76)

Durch die zufällige Initialisierung der Verbindungsgewichte ist der Startpunkt in diesem Gebirge beliebig. Das Ziel des Algorithmus ist, von diesem Startpunkt das globale Minimum der Fehlerfunktion zu finden. Dies geschieht mit Hilfe eines Gradientenverfahrens, das heißt die Verbindungsgewichte werden über ein Gradientenverfahren so lange verändert bis das Minimum erreicht ist. Zur genauen Darstellung des Algorithmus wird auf Rumelhart/Hinton/Williams (1986) oder Oberhofer/Zimmerer (1998, S. 17ff.) verwiesen. Einige Probleme des Gradientenverfahrens und Möglichkeiten zu deren Bewältigung werden bei Hilbert/Dittmar (1997, S. 30ff.) beschrieben.

6.2 Optimale Netzwerkstruktur

Die optimale Netzwerkstruktur eines KNN hängt ab von der Anzahl an verborgenen Schichten, der Anzahl Neuronen und der Struktur der Verbindungsgewichte. Hierfür gibt es keine allgemeingültige Regel, meist wird die optimale Netzwerkstruktur in einem Trial-and-Error Verfahren ermittelt (Urban, 1998, S. 89). Die Anzahl der Neuronen in der Ein- und Ausgabeschicht ergibt sich meist aus der Aufgabenstellung, somit ist die Anzahl an Neuronen in den verborgenen Schichten zu optimieren. Allerdings ist zu beachten, dass eine zu große Anzahl an Neuronen zu Overfitting führen kann. Nach einer Faustregel von Harrell/Lee/Mark (1996, S. 364) sollten die für jedes Neuron mindestens 10 Objekte in den Trainingsdaten vorhanden sein.

Um eine gute Modellkomplexität zu finden, sind einige Heuristiken entwickelt worden, beispielsweise das „Growing“ (Wiedemann/Buckler, 2001, S. 62), „Pruning“ (Zell, 1994, S. 319ff.) und „Regulation“ (Wiedemann/Buckler, 2001, S. 62), auf die jedoch nicht näher eingegangen wird.

Auch eine zu lange Dauer des Lernens kann zu Overfitting führen. Je größer die Iterationszahl in der Lernphase, desto genauer lernt das KNN die Trainingsdaten. Eine Methode das Lernen sinnvoll zu beenden, ist das im SAS EM implementierte **Early Stopping** (Potts, 2000, S. 182). Neben den Trainingsdaten, die für die Bestimmung der Gewichte verwendet werden, wird nach jedem Schritt das Modell auf die Validierungsdaten angewandt und die Güte, hier der Gesamtfehler E, protokolliert. Dies ergibt die Darstellung aus Abbildung 41.

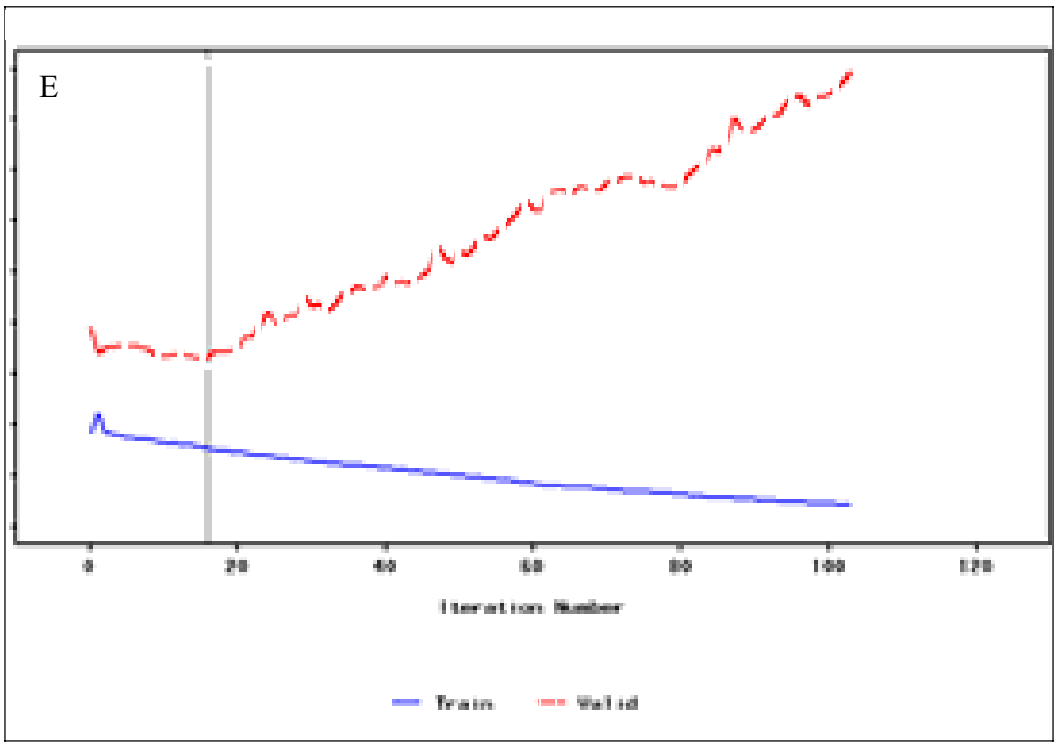


Abbildung 41: Early Stopping
Quelle: Screencopy SAS Enterprise Miner

Während der Fehler bei den Trainingsdaten kontinuierlich sinkt, neigt der Fehler in den Validierungsdaten ab einem gewissen Zeitpunkt dazu, wieder anzusteigen. Dies ist ein Kennzeichen für Overfitting. Die Lernphase sollte im globalen Minimum der Validierungsdaten beendet werden. Zu beachten ist, dass auch eine zu große Zahl an Neuronen das Overfitting begünstigen kann. In diesem Fall sollte nach Anders (1995, S. 37) jedoch besser die Zahl der Neuronen reduziert werden.

Zum Abschluss lässt sich sagen, dass mit KNN hochdimensionale, nichtlineare Zusammenhänge in den Daten modelliert werden können. Ein Nachteil von KNN liegt allerdings in deren "Black Box"-Charakter: Aufgrund der Komplexität insbesondere bei mehrschichtigen Netzen ergibt sich folgende Konsequenz: „Neuronale Netze ... sind im Allgemeinen nicht verifizierbar; die Ergebnisse sind nicht nachvollziehbar. Die funktionale Überprüfung eines neuronalen Netzes kann nur per Validierung ... erfolgen" (Kratzer, 1990, S. 89). Bei steigender Komplexität ist somit kaum transparent, welche Variablen wichtig sind oder wie sie interagieren (Sarle, 1997).

6.3 Empirische Ergebnisse

In dieser Studie wird ein Standard backpropagation MLP mit 2 verborgenen Schichten, die jeweils 3 Neuronen beinhalten, verwendet. Die Aktivierungsfunktion ist dabei linear, als Outputfunktion wird der hyperbolische Tangens gewählt. Es wird das Early-Stopping angewandt und dabei das Modell ausgewählt, das auf Basis der Validierungsdaten den kleinsten Fehler aufweist. Bei der Methode Profitmatrix (siehe S. 59) wird das Modell mit dem maximalen Profit auf Basis der Validierungsdaten ausgewählt. Auch hier beeinflusst die Profitmatrix nur die Wahl des Modells, nicht die Schätzung der einzelnen Verbindungsgewichte.

Tabelle 22 zeigt die empirischen Ergebnisse der verschiedenen Varianten (siehe Tabelle 14, S. 106) der KNN bei den Testdaten.

	10%	20%	40%	80%
alle	16	26	44	81
profit	18	37	46	78
dupl	32	39	56	90
z 1:10 1	31	47	66	90
z 1:10 2	31	44	61	88
z 1:10 3	37	47	64	85
cl 1:10 1	34	42	71	95
cl 1:10 2	34	49	76	97
cl 1:10 3	36	41	56	88
z 1:5 1	27	37	69	86
z 1:5 2	19	37	71	90
z 1:5 3	32	39	64	95
cl 1:5 1	34	47	61	90
cl 1:5 2	24	44	78	97
cl 1:5 3	31	44	63	90
z 1:1 1	32	46	63	81
z 1:1 2	27	47	64	86
z 1:1 3	36	41	56	88
cl 1:1 1	34	51	70	90
cl 1:1 2	27	41	69	92
cl 1:1 3	32	41	75	93
cl all 1:1 1	27	34	61	89
cl all 1:1 2	31	37	53	88
cl all 1:1 3	24	29	53	90
cl med 1:1 1	29	41	64	92
cl med 1:1 2	22	36	59	93
cl med 1:1 3	25	34	64	97

Tabelle 22: Gains-Chart Ergebnisse auf Basis der Testdaten bei KNN

Die Ergebnisse zeigen, dass deutliche Unterschiede zwischen den Vorgehensweisen bestehen. Die Schwankungsbreite zwischen den unterschiedlichen Vorgehensweisen ist relativ groß, besonders auffällig ist die schlechte Performance der Methode „alle“ und „profit“. Der Unterschied zwischen der besten und der schlechtesten Modellvariante beträgt beim 10%-Wert 20 Prozentpunkte. Dies zeigt, dass KNN ebenso wie Entscheidungsbäume sehr stark auf Unterschiede in den Daten reagieren. Die meisten Modelle erreichen bei 10% der Einsatzmenge zwischen 16% und 37% der Besteller, im Mittel 29%. Beim 20%-Wert aus dem Gains-Chart liegt der Mittelwert bei 41%. Diesen Wert erreichen einige Entscheidungsbäume bereits beim 10%-Wert. Beim 80%-Wert werden mit 90% aller Besteller im Mittel ebenfalls unterdurchschnittliche Ergebnisse erzielt.

Die Methode „alle“ schneidet beim 10%-, 20%- und 40%-Wert am schlechtesten ab. Beim 80%-Wert wird das zweitschlechteste Ergebnis erzielt. Diese Methode scheint bei niedrigen Responsequoten für KNN völlig ungeeignet zu sein. Curry/Rumelhart (1990, S. 223f.) verwenden eine Faustregel, dass backpropagation KNN Probleme haben, eine Klasse, die seltener als 1% vorkommt, zu erkennen. Weitere Autoren belegen, dass KNN dazu neigen, eine selten auftretende Klasse zu ignorieren (Lowe/Webb, 1990, S. 309; Lu et al., 1998; Ohno-Machado, 1996b, S. 174). Dies kann aufgrund der Ergebnisse in dieser Studie bestätigt werden.

Die Methode „profit“ erzielt bei allen Werten sehr schlechte Ergebnisse. Diese Methode scheint bei niedrigen Responsequoten für KNN ebenfalls ungeeignet zu sein.

DeRouin et al. (1991, S. 138f.) empfehlen die Methode „duplizieren“ bei einer selten auftretenden Klasse. Die Duplizierung der Besteller erreicht beim 10%-Wert ein sehr gutes Ergebnis. Beim 40%-Wert wird ein unterdurchschnittliches Ergebnis erzielt, der Rest liegt in etwa im Durchschnitt. Die Ergebnisse dieser Studie zeigen, dass diese Methode bei geringen Responsequoten trotz der bekannten Nachteile geeignet ist.

Bei der Methode „z 1:10“ werden bis auf den 80%-Wert deutlich überdurchschnittliche Ergebnisse erreicht.

Die Methode „cl 1:10“ erzielt bei allen Werten deutlich überdurchschnittliche Ergebnisse. Es werden stets bessere Ergebnisse als bei der Methode „z 1:10“ erzielt. Der zusätzliche Vorverarbeitungsschritt wirkt sich hier positiv auf die Ergebnisse aus.

Bei der Methode „z 1:5“ sind die Ergebnisse bis auf den 40%-Wert unterdurchschnittlich. Das Modell schneidet deutlich schlechter als die Methode „z 1:10“ ab.

Die Methode „cl 1:5“ weist im Vergleich zu „z 1:5“ bessere Ergebnisse auf. Bis auf den 10%-Wert werden bei allen Messpunkten überdurchschnittliche Ergebnisse erreicht. Das Modell schneidet jedoch ebenfalls schlechter als die Methode „cl 1:10“ ab.

Die Methode „z 1:1“ erzielt bei den ersten beiden Messpunkten etwas überdurchschnittliche und bei den letzten beiden Messpunkten etwas unterdurchschnittliche Ergebnisse.

Es werden bessere Ergebnisse als bei „z 1:5“ erreicht, jedoch schlechtere im Vergleich zu „z 1:10“.

Die Methode „cl 1:1“ erzielt ebenfalls an allen Messpunkten sehr gute Ergebnisse. Im Vergleich zu „z 1:1“ werden beim 40%- und 80%-Wert deutlich bessere Ergebnisse erreicht.

Die Methode „cl all 1:1“ weist bei allen Werten unterdurchschnittliche Ergebnisse auf. Diese Methode scheint bei KNN ebenfalls nicht geeignet zu sein.

Die Methode „cl med 1:1“ erzielt beim 10%-Wert ein unterdurchschnittliches Ergebnis, beim 80%-Wert wird das beste Ergebnis erreicht. Im Vergleich zu „cl 1:1“ ist diese Methode jedoch nicht zu bevorzugen.

Die Ergebnisse in dieser Studie bestätigen die Ergebnisse von Japkowicz (2000, S. 116) und Ling/Li (1998, S. 77), die zeigen, dass das Duplizieren dem Downsizing nicht überlegen ist. Hier werden bei beiden Methoden ähnliche Ergebnisse erzielt.

Abbildung 42 zeigt in einem Säulendiagramm das Ergebnis aller Modelle beim 10%-Wert im Gains-Chart. Die Abbildung zeigt, dass die Schwankung bei den Zufallsauswahlmethoden erwartungsgemäß größer ist als bei den clusteranalysegestützten Auswahlmethoden.

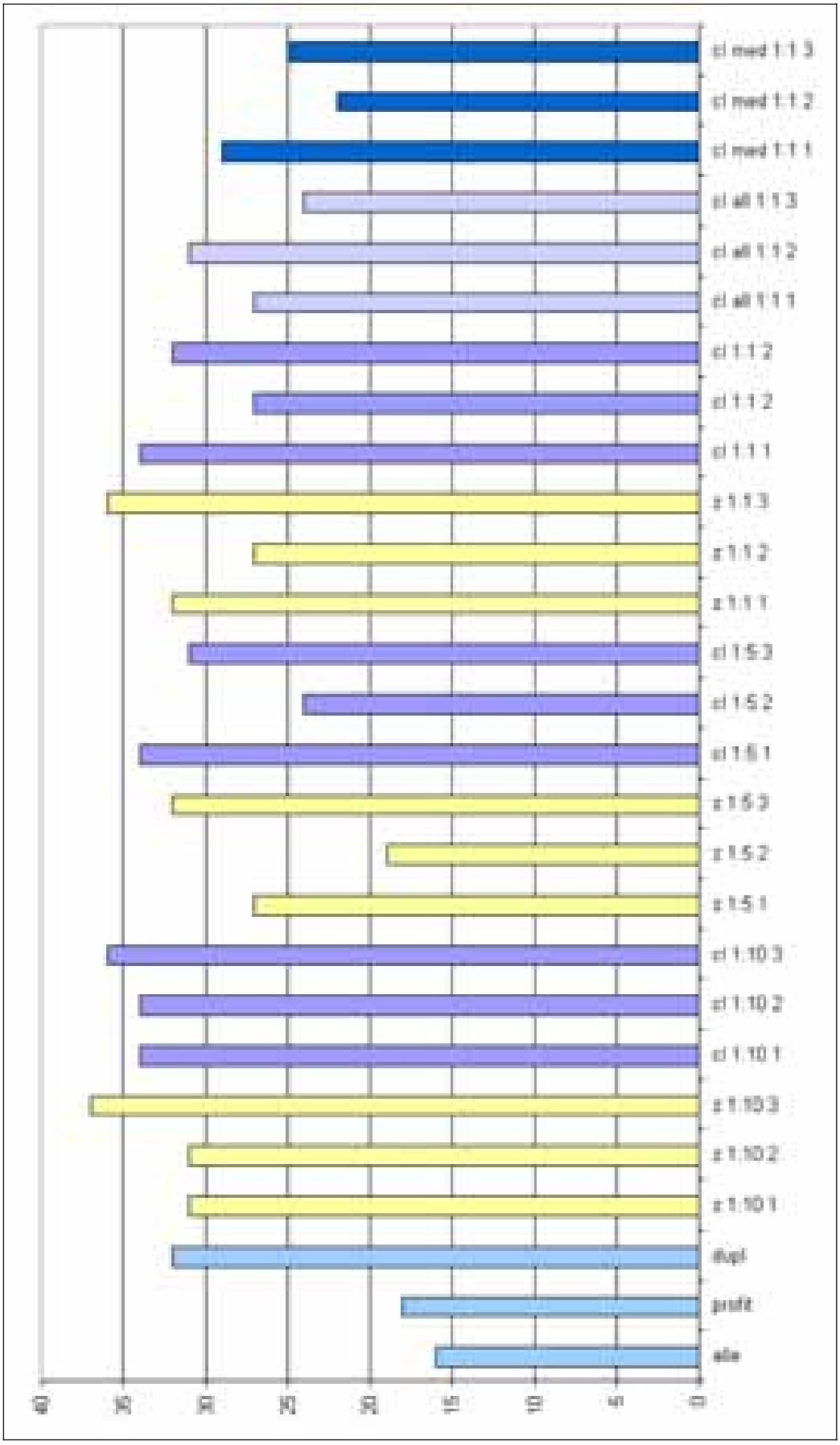


Abbildung 42: Gains-Chart Ergebnisse beim 10%-Wert auf Basis der Testdaten bei KNN
Quelle: Eigene Darstellung

Werden die Ergebnisse der Methoden z 1:10, cl 1:10, z 1:5, cl 1:5, z 1:1, cl 1:1, cl all 1:1 und cl med 1:1 gemittelt, ergibt sich Abbildung 43:

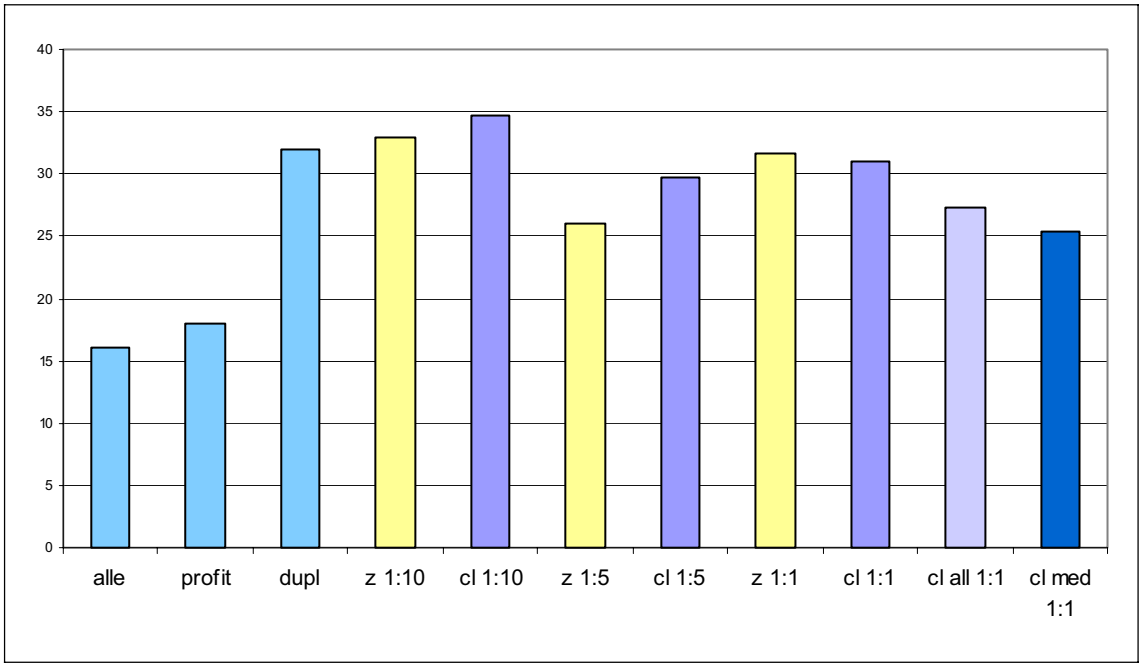


Abbildung 43: Gemittelte Gains-Chart Ergebnisse beim 10%-Wert auf Basis der Testdaten bei KNN
Quelle: Eigene Darstellung

Durch die Mittelwertbildung werden Ausreißer abgefedert. Es zeigt sich in dieser Darstellung, dass die Modellgüte bei einem Verhältnis der Zielvariable von 1:10 bzw. 1:5 bei den Zufallsauswahlmethoden schlechter als bei den clusteranalysegestützten Auswahlmethoden ist. Die Ergebnisse der Methoden „z 1:1“ und „cl 1:1“ sind nahezu identisch, während „cl all 1:1“ und „cl med 1:1“ deutlich schlechter abschneiden.

Abbildung 44 zeigt die Modellergebnisse beim 80%-Wert aus dem Gains-Chart im Vergleich. Auch hier fällt die größere Streuung bei den Methoden mit Zufallsauswahl auf. Außerdem sind hier alle clusteranalysegestützten Methoden in der Modellgüte deutlich überlegen.

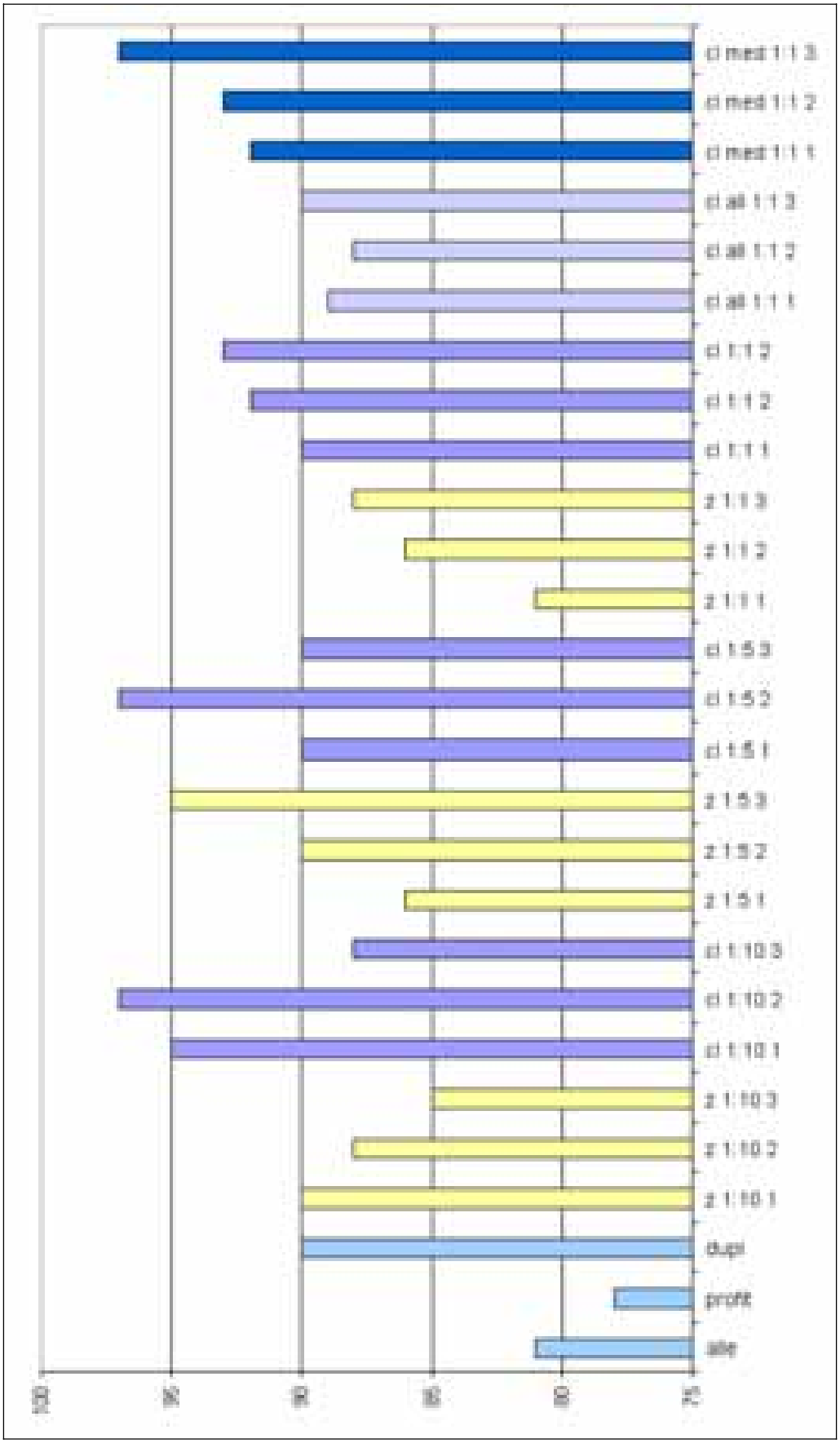


Abbildung 44: Gains-Chart Ergebnisse beim 80%-Wert auf Basis der Testdaten bei KNN
Quelle: Eigene Darstellung

6.4 Zusammenfassung

KNN sind sehr gut geeignet, wenn der Zusammenhang in den Daten unbekannt ist. Der große Nachteil der KNN liegt in der fehlenden Transparenz des Modells. Aufgrund des Black-Box Charakters können hier keine interpretierbaren Regeln ausgegeben werden (siehe S. 146).

Die empirischen Ergebnisse zeigen, dass die Schwankungsbreite bei den KNN insgesamt gesehen relativ groß ist. Gerade im Vergleich mit den Entscheidungsbäumen und der logistischen Regression weisen sie die größte Streuung auf. Insgesamt lässt sich zu den KNN sagen, dass die Methoden „alle“ und „profit“ ungeeignet sind. Das Duplizieren erzielt wie bei den Entscheidungsbäumen und der logistischen Regression relativ gute Ergebnisse, benötigt jedoch deutlich längere Rechenzeit und mehr Systemressourcen. Die clusteranalysegestützten Methoden erzielen nahezu immer bessere Ergebnisse als die Methoden mit reiner Zufallsauswahl und sind somit zu bevorzugen. In dieser Arbeit zeigt sich, dass bei KNN der Anteil der selten auftretenden Klasse in der Zielvariablen erhöht werden sollte. Die Ergebnisse von Ohno-Machado (1996b, S. 174), die gerade bei KNN denselben Anteil von 1-Klasse und 0-Klasse in der Zielvariablen empfiehlt, können hier nicht direkt bestätigt werden, da auch „cl 1:10“ beispielsweise sehr gute Resultate erzielt. Allerdings sollte das Verhältnis von 1-Klasse zu 0-Klasse nicht zu extrem sein.

Je nach Zielsetzung, das heißt, ob nur die besten Adressen ausgewählt oder nur die schlechtesten Adressen eingespart werden sollen, empfehlen sich unterschiedliche Methoden. Bei beiden Varianten erzielt die Methode „cl 1:10“ sehr gute Ergebnisse. Der Anwender muss hier wiederum entscheiden, ob er das beste Modell oder eine Durchschnittsbildung bevorzugt. Allerdings stellt sich hier nicht die Frage nach der möglicherweise problematischen Interpretation bei Verwendung der Durchschnittsbildung.

Sollen nur die besten 10% der Objekte angeschrieben werden, empfehlen sich bei KNN die Methoden „cl 1:10“, „z 1:10“ und „duplizieren“. Bei „cl 1:10“ werden im Mittel 35% aller Besteller erreicht. Entscheidungsbaumverfahren erreichen bei diesem Wert bereits 40% aller Besteller, bei der logistischen Regression werden 36% erreicht.

Sollen die schlechtesten 20% weggelassen werden, so werden mit der Methode „cl med 1:1“ im Mittel 94% der Besteller erreicht, mit der nächstbesten Methode, „cl 1:10“, im Mittel 93%. Die besten Entscheidungsbaumverfahren erzielen hier bereits Werte von 99%, bei der logistischen Regression werden 97% erreicht.

7. Vergleich der verwendeten Variablen verschiedener Modellvarianten

In diesem Kapitel erfolgt mit Hilfe einer Multidimensionalen Skalierung eine Positionierung der verschiedenen Modellvarianten. Die Modellvarianten werden anhand der Einflussstärke der verwendeten unabhängigen Variablen auf die Zielvariable verglichen. Da KNN diesbezüglich keine Rückschlüsse zulassen, werden diese hier nicht weiter betrachtet. Es soll vor allem die Auswirkung des zusätzlichen Vorverarbeitungsschritts, die clusteranalysegestützte Stichprobenziehung, gezeigt werden. Weiterhin wird das Verhalten der einzelnen Modellvarianten bei einer Variablenreduzierung untersucht.

7.1 Bildung einer Distanzmatrix

Die Basis zur Durchführung einer Multidimensionalen Skalierung bildet eine **Distanzmatrix**. Diese enthält in den Zeilen die einzelnen Modellvarianten und in den Spalten die unabhängigen Variablen (siehe Tabelle 23).

	Variablen Nummer (siehe S. 49ff.)					Anzahl verwendete unab-
Modellvariante	Nr. 3	Nr. 12	Nr. 15	...	Nr. 59	hängige Variablen/Blätter
alle	6	6	6	...	6	6
profit	3	6	6	...	6	6

Tabelle 23: Beispielhafter Auszug aus der Datenmatrix für die MDS

Die Werte für die einzelnen Elemente der Datenmatrix bestimmen sich nach der Wichtigkeit der Variablen. Dabei wird je Modellvariante auf einer Skala von 1 bis 6 angegeben, ob und mit welchem Einfluss eine bestimmte unabhängige Variable in der entsprechenden Modellvariante vorkommt. Wird eine Variable nicht verwendet, erhält die Methode an dieser Stelle die maximale Ausprägung „6“, ist es die stärkste Einflussvariable, so wird die „1“ zugeordnet. Die maximale Tiefe bei Entscheidungsbäumen beträgt fünf. Bei den Entscheidungsbaumverfahren wird beispielsweise der Variable, die zum ersten Split verwendet wird, der Wert „1“ zugeordnet, kommt sie erst in der fünften Ebene zum ersten Mal vor, wird ihr der Wert „5“ zugeordnet. Bei der logistischen Regression werden folgenden Werten aus der Wald-Statistik nachfolgende Ausprägungen in Anlehnung an den Maximalwert bei Entscheidungsbäumen zugewiesen. Die einzelnen Wertebereiche sind so definiert, dass in jedem Bereich in etwa gleich viele Nennungen auftreten.

Wert der Wald-Statistik	Ausprägung in der Datenmatrix
< 96; . >	1
< 50;96]	2
< 28;49]	3
< 14;27]	4
[0;14]	5

Das Skalenniveau dieser Variablen ist ordinal. Zusätzlich wird noch eine metrische Variable zum Vergleich herangezogen: Die Anzahl der verwendeten unabhängigen Variablen bei der logistischen Regression und die Anzahl der Endknoten nach dem Pruning bei den Entscheidungsbäumen. Die gesamte Datenmatrix ist im Anhang F zu finden (siehe S. 216f.).

Nach Wördenweber (1985, S. 72ff.) gibt es verschiedene Ansätze, wie man im Fall gemischtskalierter Daten eine Distanzmatrix bestimmen kann. Eine Möglichkeit, die alle Skalenniveaus berücksichtigt, bietet die Aggregation einzelner Distanzwerte der partiellen Distanzmatrizen. Es wird für jede Variablengruppe, die aus Variablen mit dem gleichen Skalenniveau besteht, eine partielle Distanzmatrix D_P berechnet und diese dann zu einer Gesamtdistanzmatrix aggregiert (Bausch/Opitz, 1993, S.45 ff.; Opitz, 1980, S. 57ff.; Wördenweber, 1985, S. 117ff.). Ein Problem stellt die Gewichtung der einzelnen Variablen dar. Es sind Ausreißer oder Extremwerte zu beachten, da dies sonst zu einer künstlichen Untergewichtung der Distanzen führen kann. Zu jeder unabhängigen Variablen *k* kann eine individuelle nichtnegative Merkmalsgewichtung g_k vergeben werden, die bei den verschiedenen Distanzmaßen so eingerechnet wird, dass eine Normierung eintritt (Bausch/Opitz, 1993, S. 46).

Für die Berechnung der paarweisen Objektdistanzen von ordinalen unabhängigen Variablen wird die Summe der absoluten Rangdifferenzen verwendet. r_{ik} bezeichnet dabei den Rang der Ausprägung des Merkmals *k* bei Objekt *i*. Bei der metrischen unabhängigen Variablen wird die absolute Differenz der Ausprägung a_{io} verwendet. Insgesamt bestimmt sich die Distanz folgendermaßen:

$$d_{ij}^{ges} \mid \sum_{k} \left| r_{ik} - r_{jk} \right| \cdot g \left| a_{i0} - a_{j0} \right| .$$

Bei der Bildung der Distanzmatrix werden alle ordinalen unabhängigen Variablen mit Gewicht Eins versehen und die metrische unabhängige Variable mit dem Gewicht 1/(90%-Quantil – 10%-Quantil), um die unterschiedliche Spannweite der Variablen und Ausreißer zu berücksichtigen. Bei der metrischen Variablen treten Werte im Intervall [2;26] auf, bei den ordinalen Variablen Werte im Intervall [1;6].

7.2 Beschreibung der Multidimensionalen Skalierung

Ziel der Verfahren der **Multidimensionalen Skalierung** (MDS) ist die Positionierung von Untersuchungsobjekten in einem niedrig-dimensionalen Raum, so dass die Distanzen der Objektpositionen den empirischen Ähnlichkeitsbeziehungen möglichst gut entsprechen (Opitz/Schwaiger, 1998, S. 563). Es wird die nichtmetrische MDS nach Kruskal verwendet, deren Ausgangspunkt eine empirische Distanzmatrix $D=(d(i,j))_{n,n}$ ist. Das Ziel besteht darin, eine Konfiguration $X=(x_{ik})_{n,q}$ im Raum $\mathbb{R}^q$ mit den Objektvektoren x(i) so zu bestimmen, dass die euklidischen Distanzen $d_{ij}(X)$ der Repräsentation X möglichst widerspruchsfrei mit den empirischen Distanzen in D sind. Je näher zwei Objekte im Wahrnehmungsraum beieinander liegen, desto ähnlicher werden sie empfunden (Backhaus et al., 2000, S. 506). Optimal wäre eine Konfiguration X, wenn für alle Paare *(i,j)* von Objekten $d_{ij}=d_{ij}(X)$ gilt. Da häufig in den empirisch ermittelten Distanzen auch qualitative Daten verarbeitet werden, wird die weniger scharfe Monotoniebedingung $d_{ij} \Omega d_{uv} \heartsuit d_{ij}(X) \Omega d_{uv}(X)$ mit $i,j,u,v \subset N$ gefordert (Bausch/Opitz, 1993, S. 70). Ein Objektpaar *(i,j)*, das eine kleinere empirische Distanz aufweist als das Paar *(u,v)*, soll auch im Darstellungsraum mit einer kleineren Entfernung versehen sein. Die d_{ij} werden aufsteigend angeordnet und parallel dazu die jeweils zugehörige Distanz $d_{ij}(X)$. Bei Abweichungen der Monotoniebedingung werden die $d_{ij}(X)$ so lange mittels einer monotonen Regression in ι_{ij} überführt, bis die Monotoniebedingung erfüllt ist. Das Vorgehen zur Bestimmung der ι_{ij} ist bei Opitz (1980, S.130 ff.) beschrieben. Als Gütemaß der Konfiguration wird die Summe der quadratischen Abweichungen zwischen $d_{ij}(X)$ und ι_{ij} verwendet:

$$b(x_1,...,x_n) \mid \sum_{i\{j} (d_{ij}(X) 4 \iota_{ij})^2 .$$

Kruskal bezeichnet den Bewertungsindex *b* als Stress. Er wird genau dann 0, wenn die Monotoniebedingung erfüllt ist. Um die Stressfunktion auf den Bereich [0;1] zu normieren, wird der maximale Stress bestimmt, der sich aus folgender Formel errechnen lässt:

$$b_{\max} \mid \sum_{i\{j} (d_{ij}(X) - \bar{d}(X))^2 .$$

Der normierte Stress b^*_{norm} errechnet sich wie folgt:

$$b^*_{norm} \mid \frac{b}{b_{\max}} \quad .$$

Der Stress misst also, wie gut eine Konfiguration die Monotoniebedingung erfüllt (Backhaus et al., 2000, S. 520). Mit Hilfe eines Gradientenverfahrens werden die Objekte nun mit einer dynamischen Schrittweite so lange verschoben, bis eine möglichst gute räumliche Anordnung gefunden ist (Bausch/Opitz, 1993, S. 71f.). Für die Bewertung einer gefundenen Konfiguration anhand des normierten Stress gibt Kruskal folgende Faustregel an:

Nicht zufriedenstellend,	wenn $0{,}20 < b^*_{norm} \leq 1$,
Ausreichend,	wenn $0{,}15 < b^*_{norm} \leq 0{,}20$,
Zufriedenstellend,	wenn $0{,}10 < b^*_{norm} \leq 0{,}15$,
Gut,	wenn $0{,}05 < b^*_{norm} \leq 0{,}10$,
Sehr gut,	wenn $0{,}00 < b^*_{norm} \leq 0{,}05$.

Die MDS ist also ein iteratives Verfahren, das ausgehend von einer Startlösung in weiteren Schritten eine Verbesserung der Objektpositionierung anstrebt. Der Ablauf dieses Verfahrens wird bei Opitz (1980, S. 109ff.) beschrieben.

In dieser Arbeit wird eine MDS nach Kruskal mit zufälligen Startwerten und dynamischer Schrittweite durchgeführt.

7.3 Empirische Ergebnisse

Die MDS wird mit dem Softwareprogramm MSTAT (Bausch/Opitz, 1993) durchgeführt.

Abbildung 45 zeigt, wie oft die einzelnen unabhängigen Variablen in den insgesamt 54 Modellvarianten (davon je 27 Entscheidungsbäume bzw. logistische Regression) verwendet werden. Die Variable Nr. 8 (siehe Tabelle 3, S. 51ff.) wird bei 50 von 54 Modellvarianten verwendet. Die Mitarbeiterzahl der Unternehmen scheint für diese Fragestellung ein wichtiges Merkmal zu sein. Mit 48 Nennungen folgt die Variable Nr. 53, also die Anzahl der bereits erhaltenen Werbeaktionen zu dem beworbenen Produkt. Die Variable Nr. 22, die den Zeitpunkt der ersten Bestellung angibt, ist mit 45 Nennungen ebenfalls in den meisten Modellvarianten enthalten. Auffällig ist die Variable Nr. 59, die den Zeitpunkt angibt, wann der Kunde die letzte Werbeaktion erhielt. Diese Variable tritt bei allen Entscheidungsbaumverfahren auf, wird jedoch nie bei der logistischen Regression herangezogen. Insgesamt werden vier Variablen zur Abbildung der Werbehistorie (Nr. 53, 57, 58, 59), 14 Variablen zur Beschreibung des Bestellverhaltens (Nr. 12, 14, 15, 16, 18, 20, 22, 24, 45, 47, 48, 49, 50, 51) und sechs Grunddaten (Nr. 3, 5, 6, 7, 8, 9) in den Modellvarianten verwendet. Anzumerken ist, dass die beiden manuell aufgenommenen Variablen Nr. 3 und Nr. 57 (siehe S. 48) häufig verwendet werden.

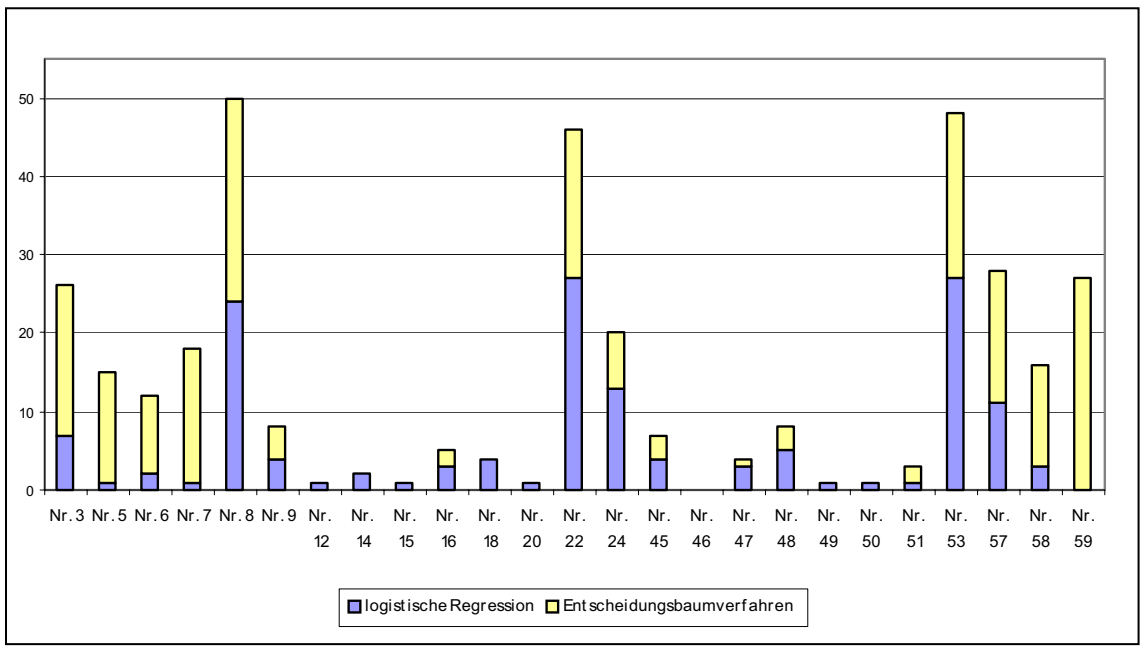


Abbildung 45: Häufigkeitsverteilung der verwendeten Variablen
Quelle: Eigene Darstellung

Abbildung 46 zeigt das Ergebnis der MDS aller Modellvarianten ausgenommen der Methode „duplizieren“, da diese deutlich von allen anderen entfernt liegt. Der Grund dafür ist, dass bei dieser Methode deutlich mehr unabhängige Variablen als bei allen anderen Methoden verwendet werden und sie damit so weit außen liegt, dass sie die gesamte Darstellung stark verzerren würde. Diese Methode wird bei allen folgenden Darstellungen nicht weiter berücksichtigt. Der Stress der nachfolgenden Darstellung liegt bei 0,0783 und ist als gut zu bezeichnen. Die Legende erfolgt in Anlehnung an Tabelle 14 (siehe S. 106).

Logistische Regression		Entscheidungsbäume	
Nr.	Variante	Nr.	Variante
1	alle	28	alle
2	profit	29	profit
4	z 1:10 1	31	z 1:10 1
5	z 1:10 2	32	z 1:10 2
6	z 1:10 3	33	z 1:10 3
7	cl 1:10 1	34	cl 1:10 1
8	cl 1:10 2	35	cl 1:10 2
9	cl 1:10 3	36	cl 1:10 3
10	z 1:5 1	37	z 1:5 1
11	z 1:5 2	38	z 1:5 2
12	z 1:5 3	39	z 1:5 3
13	cl 1:5 1	40	cl 1:5 1
14	cl 1:5 2	41	cl 1:5 2
15	cl 1:5 3	42	cl 1:5 3
16	z 1:1 1	43	z 1:1 1
17	z 1:1 2	44	z 1:1 2
18	z 1:1 3	45	z 1:1 3
19	cl 1:1 1	46	cl 1:1 1
20	cl 1:1 2	47	cl 1:1 2
21	cl 1:1 3	48	cl 1:1 3
22	cl all 1:1 1	49	cl all 1:1 1
23	cl all 1:1 2	50	cl all 1:1 2
24	cl all 1:1 3	51	cl all 1:1 3
25	cl med 1:1 1	52	cl med 1:1 1
26	cl med 1:1 2	53	cl med 1:1 2
27	cl med 1:1 3	54	cl med 1:1 3

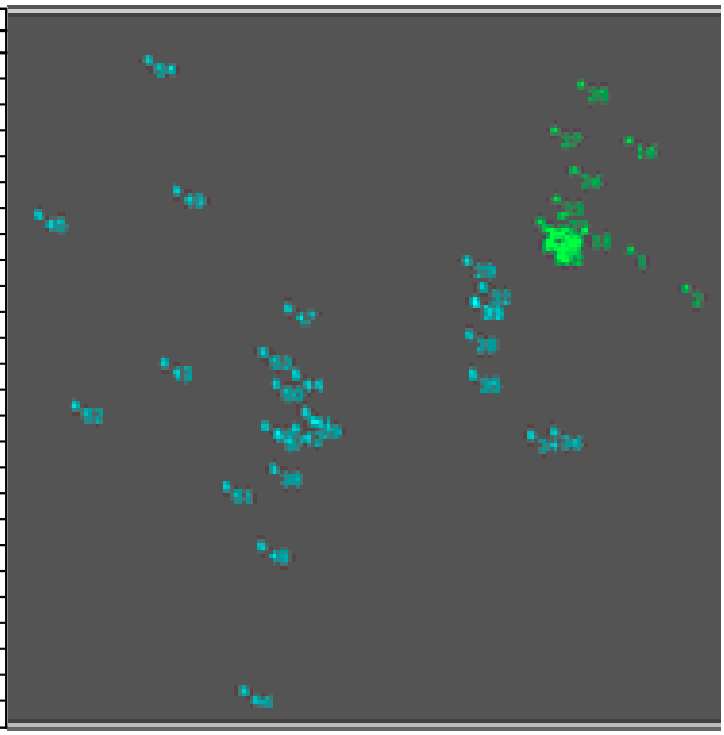


Abbildung 46: Ergebnis der MDS aller Modellvarianten
Quelle: Screenshot MSTAT

Die Modellvarianten der logistischen Regression (A bzw. Nr. 1-27 in Abbildung 46) belegen den Bereich rechts oben in der Grafik, während die Entscheidungsbaumverfahren (B bzw. Nr. 28-54 in Abbildung 46) über die restliche Fläche streuen. Um die Unterschiede zwischen den einzelnen Varianten besser zu veranschaulichen, werden im folgenden die logistische Regression und die Entscheidungsbaumverfahren getrennt betrachtet.

Abbildung 47 zeigt die Ergebnisse der MDS bei Einschränkung auf Modellvarianten der logistischen Regression. Die zweidimensionale Repräsentation kann aufgrund eines Stress-Werts von 0,0569 als nahezu sehr gut betrachtet werden.

Logistische Regression	
Nr.	Variante
1	alle
2	profit
4	z 1:10 1
5	z 1:10 2
6	z 1:10 3
7	cl 1:10 1
8	cl 1:10 2
9	cl 1:10 3
10	z 1:5 1
11	z 1:5 2
12	z 1:5 3
13	cl 1:5 1
14	cl 1:5 2
15	cl 1:5 3
16	z 1:1 1
17	z 1:1 2
18	z 1:1 3
19	cl 1:1 1
20	cl 1:1 2
21	cl 1:1 3
22	cl all 1:1 1
23	cl all 1:1 2
24	cl all 1:1 3
25	cl med 1:1 1
26	cl med 1:1 2
27	cl med 1:1 3

Abbildung 47: Ergebnis der MDS der logistischen Regression
Quelle: Screenshot MSTAT

Aus dieser Abbildung erkennt man, dass einzelne Modellvarianten auch näher zusammenliegen. Die Methode „cl med 1:1“ (R bzw. 25, 26, 27) befindet sich in der Grafik im rechten Randbereich. Dies liegt daran, dass nur bei dieser Variante die Variable Nr. 9 als wichtigste Variable verwendet wird. Die Methoden „profit“ (B bzw. 2) und „alle“ (A bzw. 1) liegen ebenfalls außen. Hier werden relativ ähnliche Variablen verwendet, wobei die Methode „profit“ zwei Variablen mehr enthält. Im Vergleich zu den restlichen Modellvarianten werden hier beispielsweise die beiden Variablen Nr. 16 und Nr. 47 verwendet, die sonst kein anderes Verfahren nutzt. Eine auffällig starke Streuung zeigt die Methode „z 1:1“ (L bzw. 16, 17, 18). Diese drei Modellvarianten verwenden nur zwei Variablen mit gleicher Einflussstärke gemeinsam, Variable Nr. 22 und Nr. 53. Weiterhin beinhalten zwei Modellvarianten sechs Variablen (16, 17) und eine nur vier (18). Die beiden Modellvarianten (16) und (17) nutzen dabei nur vier Variablen gemeinsam. Um eine detailliertere Darstellung zu ermöglichen wird auf eine überschauba-

re Anzahl an Methoden eingeschränkt. Dazu wird für die Varianten, bei welchen jeweils drei Stichproben untersucht werden, ein Verschiedenheitswert bestimmt, der sich aus der Summe der Distanzen zueinander berechnet. Für Variante „z 1:1“ beispielsweise: $d_{\text{Variante „z 1:1“}} = d(\text{z 1:1 1} , \text{z 1:1 2}) + d(\text{z 1:1 1} , \text{z 1:1 3}) + d(\text{z 1:1 2} , \text{z 1:1 3})$.

Somit ergeben sich bei der logistischen Regression folgende Werte:

Variante	Verschiedenheitswert
z 1:10	12
cl 1:10	6,12
z 1:5	10,16
cl 1:5	12,16
z 1:1	38,3
cl 1:1	2
cl all 1:1	16,46
cl med 1:1	16,46

Abbildung 48 zeigt nun die fünf Varianten mit den niedrigsten Verschiedenheitswerten. Der Stress-Wert von 0,0328 ist als sehr gut zu bewerten.

Logistische Regression	
Nr.	Variante
4	z 1:10 1
5	z 1:10 2
6	z 1:10 3
7	cl 1:10 1
8	cl 1:10 2
9	cl 1:10 3
10	z 1:5 1
11	z 1:5 2
12	z 1:5 3
13	cl 1:5 1
14	cl 1:5 2
15	cl 1:5 3
19	cl 1:1 1
20	cl 1:1 2
21	cl 1:1 3

Abbildung 48: Ergebnis der MDS mit ausgewählten Modellvarianten der logistischen Regression
Quelle: Screenshot MSTAT

Hier erkennt man, dass die Methode „cl 1:10“ (F bzw. 7, 8, 9) relativ nahe zusammenliegt, während die Methode „z 1:10“ (D bzw. 4, 5, 6) relativ weit streut. Bei der Methode „z 1:10“ werden bei allen drei Varianten nur vier Variablen verwendet. Verfahren (5) verwendet dabei die Variable Nr. 47, während die beiden anderen an deren Stelle Variable Nr. 48 verwenden. Bei der Methode „cl 1:10“ (F bzw. 7, 8, 9) werden bei den drei Modellvarianten fünf gleiche Variablen verwendet, nur Variante (8) verwendet eine weitere Variable. Deshalb liegt sie in Abbildung 48 auch etwas entfernt. Zwischen den Methoden „z 1:5“ (H bzw. 10, 11, 12) und „cl 1:5“ (J bzw. 13, 14, 15) besteht wenig Unterschied. Alle sechs Modellvarianten verwenden drei Variablen gemeinsam, Nr. 8, Nr. 22 und Nr. 53. Bei den restlichen Variablen gibt es zum Teil Unterschiede. Auffällig ist, dass Verfahren (6) und (13) dieselben Variablen mit sehr ähnlichen Werten des Wald-Tests verwenden. Bei der Methode „cl 1:1“ (H bzw. 19, 20, 21) liegen die Objekte wiederum relativ nahe zusammen, alle drei Modellvarianten verwenden dieselben Variablen (Nr. 8, Nr. 22, Nr. 45, Nr. 53), drei davon mit derselben Einflussstärke bezüglich der Wald-Teststatistik.

Insgesamt lässt sich zu den Ergebnissen der MDS sagen, dass bei der logistischen Regression die einzelne Methoden sehr nahe zusammenliegen, ausgenommen der Methode „z 1:1“. Die Methoden „cl 1:10“ und „cl 1:1“ weisen deutlich geringere Distanzen zueinander auf als „z 1:10“ bzw. „z 1:1“. Bei „cl 1:5“ und „z 1:5“ sind die Distanzen fast gleich. Dies belegt, dass die clusteranalysegestützte Stichprobenziehung bei der logistischen Regression zu stabileren Modellergebnissen führt. Gerade bei der Variante „z 1:1“ bzw. „cl 1:1“ zeigt sich der Vorteil besonders deutlich.

Abbildung 49 zeigt das Ergebnis der MDS bei den Entscheidungsbaumverfahren. Diese Darstellung erfolgt in einem dreidimensionalen Raum, um einen guten Stress-Wert zu erreichen.

Entscheidungsbäume	
Nr.	Variante
28	alle
29	profit
31	z 1:10 1
32	z 1:10 2
33	z 1:10 3
34	cl 1:10 1
35	cl 1:10 2
36	cl 1:10 3
37	z 1:5 1
38	z 1:5 2
39	z 1:5 3
40	cl 1:5 1
41	cl 1:5 2
42	cl 1:5 3
43	z 1:1 1
44	z 1:1 2
45	z 1:1 3
46	cl 1:1 1
47	cl 1:1 2
48	cl 1:1 3
49	cl all 1:1 1
50	cl all 1:1 2
51	cl all 1:1 3
52	cl med 1:1 1
53	cl med 1:1 2
54	cl med 1:1 3

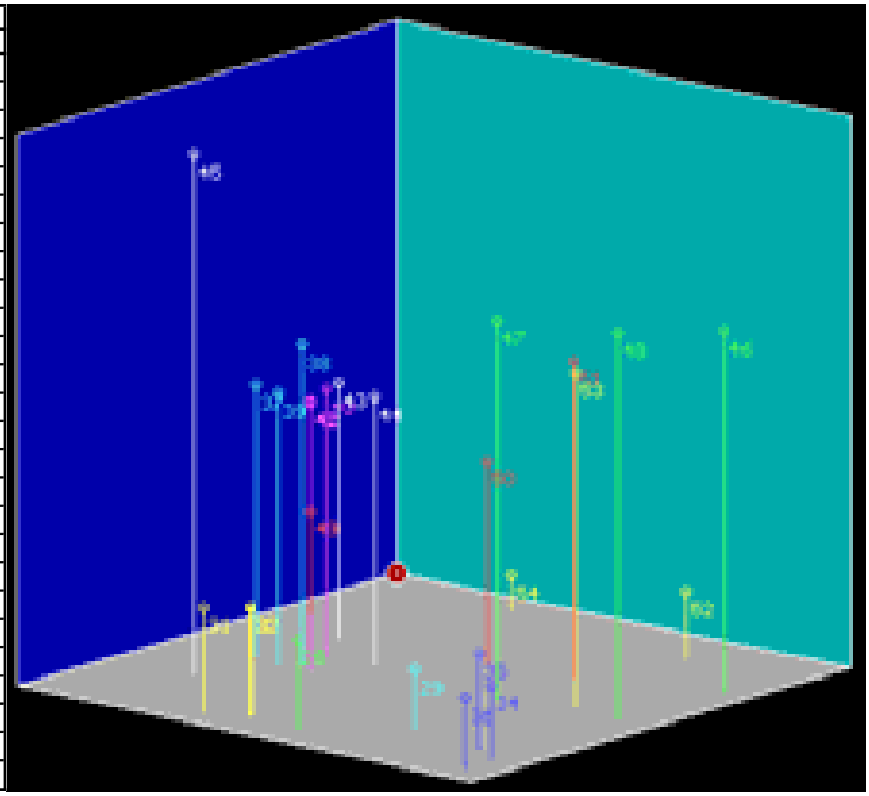


Abbildung 49: Ergebnis der MDS bei Entscheidungsbaumverfahren
Quelle: Screenshot MSTAT

Man erkennt, dass die einzelnen Methoden tendenziell wiederum eigene Bereich belegen. Allerdings ist die Streuung hier im Vergleich zur logistischen Regression deutlich größer und somit sind die Bereich nicht mehr eindeutig.

Gerade hier ist eine weitere Einschränkung der betrachteten Methoden notwendig, um eine detailliertere Darstellung zu ermöglichen. Dazu wird wiederum für die Varianten, bei welchen jeweils drei Stichproben untersucht werden, ein Verschiedenheitswert (siehe S. 162) bestimmt. Es ergeben sich folgende Werte:

Variante	Verschiedenheitswert
z 1:10	10,33
cl 1:10	30,46
z 1:5	24,46
cl 1:5	12,32
z 1:1	58,92
cl 1:1	51,26
cl all 1:1	60,46
cl med 1:1	80,77

Abbildung 50 zeigt die vier Entscheidungsbaumvarianten mit dem niedrigsten Verschiedenheitswert. Der Stress ist mit 0,0034 sehr gut.

Entscheidungsbäume	
Nr.	Variante
31	z 1:10 1
32	z 1:10 2
33	z 1:10 3
34	cl 1:10 1
35	cl 1:10 2
36	cl 1:10 3
37	z 1:5 1
38	z 1:5 2
39	z 1:5 3
40	cl 1:5 1
41	cl 1:5 2
42	cl 1:5 3

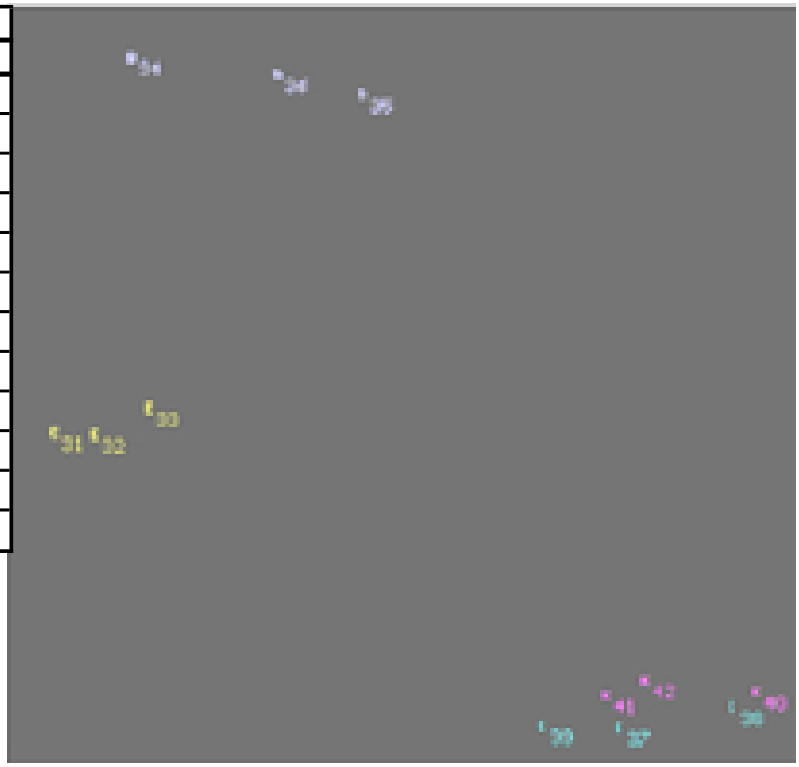


Abbildung 50: Ergebnis der MDS mit ausgewählten Modellvarianten von Entscheidungsbaumverfahren
Quelle: Screenshot MSTAT

Die Varianten „z 1:10“ (E bzw. 31, 32, 33) und „cl 1:10“ (B bzw. 34, 35, 36) sind sich innerhalb jeweils sehr ähnlich, wobei die Entscheidungsbäume bei der clusteranalysegestützten Methode mehr Blätter aufweisen. Alle Bäume bei „z 1:10“ beginnen mit einem Split nach dem Zeitpunkt der letzten Bestellung (Nr. 22), während alle Bäume bei „cl 1:10“ mit einem Split nach dem Zeitpunkt der letzten erhaltenen Bewerbung starten (Nr. 59). Die Varianten „z 1:5“ (C bzw. 37, 38, 39) und „cl 1:5“ (K bzw. 40, 41, 42) sind sich sehr ähnlich. Bei beiden wird immer mit einem Split der Variablen Nr. 22, gefolgt von Variablen Nr. 59, begonnen.

Bei den Entscheidungsbäumen ist die Streuung erwartungsgemäß wesentlich größer. Aufgrund der Instabilität des Verfahrens entstehen zum Teil deutlich unterschiedliche Ergebnisse. Die clusteranalysegestützte Stichprobenziehung zeigt bei der Verteilung „1:1“ und „1:5“ Vorteile, jedoch bleibt eine gewisse Streuung vorhanden.

7.4 Gesamtinterpretation der empirischen Ergebnisse in Verbindung mit der Repräsentation

Im folgenden werden die empirischen Ergebnisse der Responseoptimierung aus dem Gains-Chart in Zusammenhang mit der Lage der Modelle in der MDS untersucht.

Nachfolgende Abbildung 51 zeigt das Ergebnis einer MDS bei der logistischer Regression (siehe Abbildung 47, S. 161) mit einer farblichen Kennzeichnung der Modellgüte beim 10%-Wert aus dem Gains-Chart (siehe Tabelle 18, S. 125).
Zur Modellbewertung werden die Ergebnisse in fünf Quantile eingeordnet, welchen folgende Farben zugewiesen werden:

Sehr schlecht: Grün,
Schlecht: Blau,
Mittel: Rot,
Gut: Lila,
Sehr gut: Gelb.

Es lassen sich in Abbildung 51 keine eindeutig schlechten oder guten Regionen erkennen. Vielmehr liegen gute und schlechte Methoden unmittelbar nebeneinander, beispielsweise die Modellvarianten (4), (6) und (13). Es zeigen sich auch keine Konzentrationstendenzen bei der Modellgüte, so dass man nicht ableiten kann, dass Methoden, die in Randgebieten der MDS liegen, besonders gute oder schlechte Ergebnisse erzielen. Dies lässt darauf schließen, dass Methoden, die ähnliche unabhängige Variablen in ähnlicher Gewichtung verwenden, trotz allem beispielsweise aufgrund unterschiedlicher ß-Werte eine deutlich unterschiedliche Güte erzielen können.

Logistische Regression	
Nr.	Variante
1	alle
2	profit
4	z 1:10 1
5	z 1:10 2
6	z 1:10 3
7	cl 1:10 1
8	cl 1:10 2
9	cl 1:10 3
10	z 1:5 1
11	z 1:5 2
12	z 1:5 3
13	cl 1:5 1
14	cl 1:5 2
15	cl 1:5 3
16	z 1:1 1
17	z 1:1 2
18	z 1:1 3
19	cl 1:1 1
20	cl 1:1 2
21	cl 1:1 3
22	cl all 1:1 1
23	cl all 1:1 2
24	cl all 1:1 3
25	cl med 1:1 1
26	cl med 1:1 2
27	cl med 1:1 3

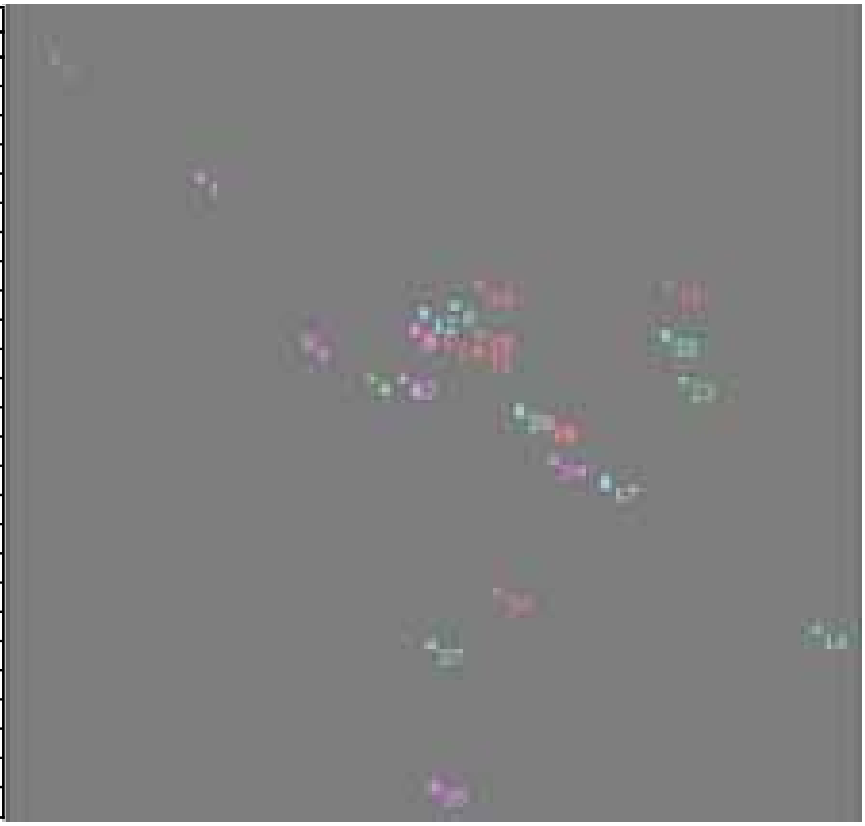


Abbildung 51: MDS der logistischen Regression mit farblicher Kennzeichnung der Ergebnisse beim 10%-Wert aus dem Gains-Chart

Ähnliche Ergebnisse werden bei der logistischen Regression auch beim Vergleich der MDS und der Ergebnisse beim 80%-Wert aus dem Gains-Chart erzielt. Auf die grafische Darstellung wird hier verzichtet.

Nachfolgende Abbildung 52 zeigt das Ergebnis der MDS bei Entscheidungsbaumverfahren (siehe Abbildung 49, S. 164) mit derselben farblichen Kennzeichnung der Modellgüte beim 10%-Wert aus dem Gains-Chart (siehe Tabelle 15, S. 107).

Es lassen sich wiederum keine eindeutig schlechten oder guten Regionen in der Grafik erkennen. Vielmehr liegen gute und schlechte Methoden ebenfalls unmittelbar nebeneinander, beispielsweise die Modellvarianten (37), (38), (39) und (40). Dies lässt darauf schließen, dass ähnliche Entscheidungsbäume trotz allem eine deutlich unterschiedliche Güte erzielen können. Die Zuordnung der einzelnen Variablenausprägungen kann beispielsweise trotz identischer Splitvariablen zu Unterschieden führen.

Entscheidungsbäume	
Nr.	Variante
28	alle
29	profit
31	z 1:10 1
32	z 1:10 2
33	z 1:10 3
34	cl 1:10 1
35	cl 1:10 2
36	cl 1:10 3
37	z 1:5 1
38	z 1:5 2
39	z 1:5 3
40	cl 1:5 1
41	cl 1:5 2
42	cl 1:5 3
43	z 1:1 1
44	z 1:1 2
45	z 1:1 3
46	cl 1:1 1
47	cl 1:1 2
48	cl 1:1 3
49	cl all 1:1 1
50	cl all 1:1 2
51	cl all 1:1 3
52	cl med 1:1 1
53	cl med 1:1 2
54	cl med 1:1 3

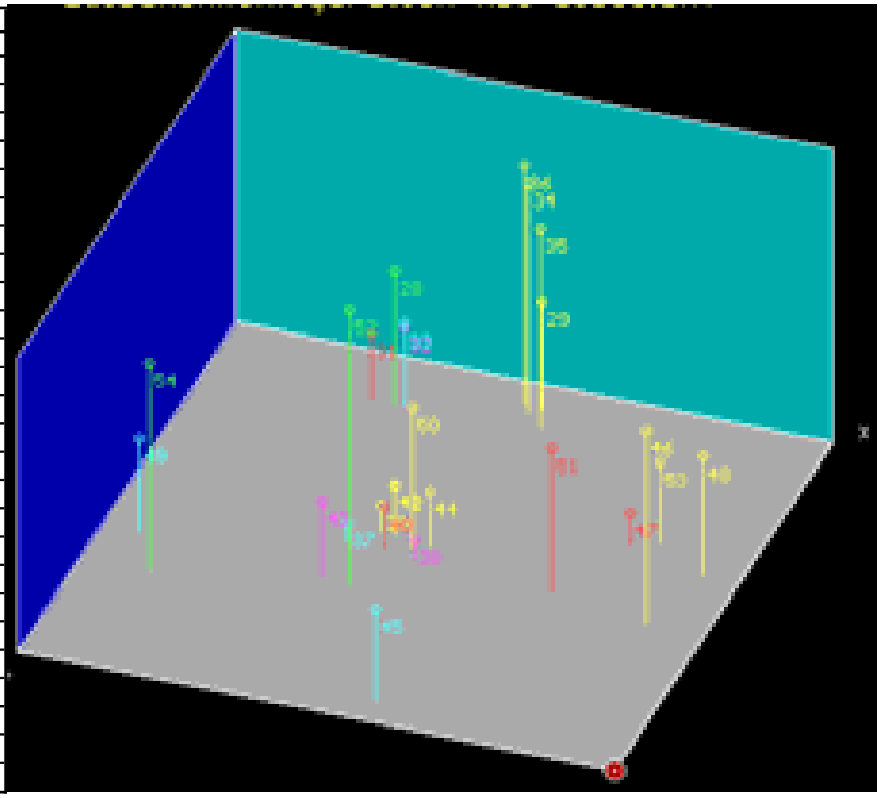


Abbildung 52: MDS der Entscheidungsbaumverfahren mit farblicher Kennzeichnung der Ergebnisse beim 10%-Wert aus dem Gains-Chart

Ähnliche Ergebnisse zeigen sich bei den Entscheidungsbaumverfahren auch beim Vergleich der MDS und der Ergebnisse beim 80%-Wert aus dem Gains-Chart.

7.5 Auswirkungen einer Reduzierung der Anzahl unabhängiger Variablen

Es zeigt sich, dass die einzelnen Modellvarianten zum Teil deutlich unterschiedliche Variablen verwenden. Dies wirft die Frage auf, wie sich eine Reduzierung der Anzahl unabhängiger Variablen sowohl auf die Modellgüte als auch auf die Interpretation auswirkt. Im folgenden werden nur die jeweils fünf am häufigsten verwendeten Variablen bei der logistischen Regression und den Entscheidungsbaumverfahren herangezogen (siehe Tabelle 24), das heißt die Variablen Nr. 3, 8, 22, 24, 53, 57 und 59.

	logistische Regression	Entscheidungsbaum
Nr. 3	7	19
Nr. 5	0	14
Nr. 6	2	11
Nr. 7	1	16
Nr. 8	24	26
Nr. 9	3	5
Nr. 12	0	0
Nr. 14	1	0
Nr. 15	0	0
Nr. 16	2	2
Nr. 18	3	0
Nr. 20	0	0
Nr. 22	27	18
Nr. 24	13	7
Nr. 45	3	4
Nr. 46	0	0
Nr. 47	3	1
Nr. 48	4	4
Nr. 49	0	0
Nr. 50	0	0
Nr. 51	0	2
Nr. 53	27	21
Nr. 57	11	16
Nr. 58	2	12
Nr. 59	0	27

Tabelle 24: Häufigkeit der Verwendung der unabhängigen Variablen je Verfahren

Die Werte aus dem Gains-Chart bei erneutem Durchlauf aller Modellvarianten bei der logistischen Regression zeigt Tabelle 25. Die Ergebnisse ändern sich dabei im Vergleich zu Tabelle 18 (siehe S. 125) kaum. Das heißt eine Einschränkung auf diese sieben Variablen führt hier nicht zu einer deutlichen Verschlechterung bei den Werten aus dem Gains-Chart.

	10%	20%	40%	80%
alle	36	48	64	89
profit	32	51	66	93
dupl	32	47	61	92
z 1:10 1	31	45	66	95
z 1:10 2	32	47	66	95
z 1:10 3	32	49	64	93
cl 1:10 1	37	44	59	88
cl 1:10 2	31	49	63	90
cl 1:10 3	36	44	59	90
z 1:5 1	34	46	59	90
z 1:5 2	34	49	64	95
z 1:5 3	31	48	65	93
cl 1:5 1	37	46	59	92
cl 1:5 2	34	36	68	93
cl 1:5 3	34	47	63	93
z 1:1 1	27	48	68	92
z 1:1 2	27	32	64	92
z 1:1 3	32	44	68	96
cl 1:1 1	32	53	68	95
cl 1:1 2	32	39	69	92
cl 1:1 3	32	51	73	93
cl all 1:1 1	27	45	70	98
cl all 1:1 2	21	35	60	94
cl all 1:1 3	29	49	63	90
cl med 1:1 1	34	47	63	93
cl med 1:1 2	36	46	64	94
cl med 1:1 3	31	49	62	95

Tabelle 25: Gains-Chart Ergebnisse auf Basis der Testdaten bei der logistischen Regression (2)

Tabelle 26 zeigt, wie häufig die jeweilige Variable den höchsten (=Nr. 1), den zweithöchsten (=Nr. 2) oder dritthöchsten Wert (=Nr. 3) in der Wald-Statistik bei den jeweiligen Modellvarianten erhalten hat. Betrachtet man nun die verwendeten Variablen, zeigt sich, dass drei davon besonders häufig verwendet werden (siehe Tabelle 26). Die wichtigste Variable scheint die Anzahl erhaltener Werbeaktionen zu dem beworbenen Produkt zu sein (Nr. 53). Als nächstes ist die Mitarbeiterzahl des beworbenen Unternehmens zu nennen (Nr. 8), gefolgt von dem Zeitabstand seit der ersten Bestellung (Nr. 22).

	anl_date	ma_zahl	di_ebest	di_lpos	anz_7409	anz_wam6	di_l_wa
	Nr. 3	Nr. 8	Nr. 22	Nr. 24	Nr. 53	Nr. 57	Nr. 59
Nr. 1	0	6	0	0	18	0	0
Nr. 2	0	15	1	0	6	1	0
Nr. 3	1	0	15	6	0	0	0

Tabelle 26: Wichtigkeit der verwendeten unabhängigen Variablen bei der logistischen Regression

Bei Betrachtung der Tabelle 27 zeigt sich, dass die Ergebnisse bei den Entscheidungsbaumvarianten im Vergleich zu Tabelle 15 (siehe S. 107) trotz der Variablenreduktion ebenfalls kaum Unterschiede aufweisen. Es verbessern sich die Werte bei den Varianten „alle“, „z 1:10“, „z 1:1“ und „cl med 1:1“. Insgesamt gesehen scheint eine Reduzierung der Variablen bei den Entscheidungsbäumen von Vorteil.

	10%	20%	40%	80%
alle	31	47	65	93
profit	40	53	74	99
dupl	42	50	65	97
z 1:10 1	38	53	68	98
z 1:10 2	37	48	68	98
z 1:10 3	34	48	68	99
cl 1:10 1	44	47	63	95
cl 1:10 2	36	48	65	98
cl 1:10 3	46	57	69	98
z 1:5 1	27	48	68	98
z 1:5 2	35	44	64	97
z 1:5 3	40	50	67	99
cl 1:5 1	36	50	69	98
cl 1:5 2	40	50	68	99
cl 1:5 3	41	52	68	98
z 1:1 1	45	52	66	98
z 1:1 2	39	55	72	91
z 1:1 3	44	53	68	100
cl 1:1 1	36	52	72	92
cl 1:1 2	34	47	66	99
cl 1:1 3	45	56	74	98
cl all 1:1 1	39	51	74	97
cl all 1:1 2	39	52	73	98
cl all 1:1 3	38	47	72	92
cl med 1:1 1	41	49	66	85
cl med 1:1 2	48	54	76	93
cl med 1:1 3	36	47	57	94

Tabelle 27: Gains-Chart Ergebnisse auf Basis der Testdaten bei Entscheidungsbäumen (2)

Tabelle 28 zeigt, wie häufig die jeweilige Variable zum ersten Split (= Nr. 1) bzw. erstmals in der zweiten (=Nr. 2) bzw. der dritten Baumebene (=Nr. 3) herangezogen wurde. Bei den verwendeten Variablen fällt auf, dass auch hier nur vier davon häufig genutzt werden (siehe Tabelle 28). Die wichtigste Variable ist nicht eindeutig zu identifizieren. Die drei Variablen Nr. 22 (Zeitraum seit der ersten Bestellung), Nr. 53 (Anzahl erhaltener Werbeaktionen zu dem beworbenen Produkt) und Nr. 59 (Zeitraum seit Erhalt der letzten Werbeaktion) werden gleich häufig als erste Splitvariable genutzt. Danach folgt die Mitarbeiterzahl des beworbenen Unternehmens (Nr. 8). Die restlichen drei Variablen nehmen eher eine untergeordnete Rolle ein.

	anl_date	ma_zahl	di_ebest	di_lpos	anz_7409	anz_wam6	di_l_wa
	Nr. 3	Nr. 8	Nr. 22	Nr. 24	Nr. 53	Nr. 57	Nr. 59
Nr. 1	0	0	9	0	9	0	9
Nr. 2	0	5	4	0	4	2	18
Nr. 3	8	15	5	2	3	7	0

Tabelle 28: Wichtigkeit der verwendeten unabhängigen Variablen bei den Entscheidungsbaumverfahren

Am deutlichsten reagieren Neuronale Netze auf die Variableneinschränkung (siehe Tabelle 29). Nahezu alle Modellvarianten erzielen bessere Ergebnisse, am deutlichsten verbessern sich die Werte bei der Methode „alle“ (vgl. Tabelle 22, S. 147).

	10%	20%	40%	80%
alle	29	41	61	85
profit	18	27	45	82
dupl	34	54	68	100
z 1:10 1	34	54	68	92
z 1:10 2	34	51	71	95
z 1:10 3	31	48	66	93
cl 1:10 1	32	58	68	97
cl 1:10 2	34	47	61	86
cl 1:10 3	37	42	68	93
z 1:5 1	32	49	66	98
z 1:5 2	39	49	80	98
z 1:5 3	31	54	76	93
cl 1:5 1	37	51	68	95
cl 1:5 2	32	51	69	98
cl 1:5 3	29	46	66	92
z 1:1 1	36	49	71	95
z 1:1 2	34	44	67	97
z 1:1 3	36	44	61	83
cl 1:1 1	39	54	64	98
cl 1:1 2	41	53	73	97
cl 1:1 3	34	56	71	95
cl all 1:1 1	31	49	64	95
cl all 1:1 2	24	42	61	92
cl all 1:1 3	32	42	66	93
cl med 1:1 1	32	37	64	88
cl med 1:1 2	31	42	69	93
cl med 1:1 3	29	54	64	98

Tabelle 29: Gains-Chart Ergebnisse auf Basis der Testdaten bei Neuronalen Netzen (2)

Insgesamt zeigt sich, dass eine Reduktion der Variablenzahl zu einer Verbesserung der Ergebnisse bei den Entscheidungsbäumen und Neuronalen Netzen führen kann. Eine mögliche Ursache könnte Overfitting sein, das heißt die Modelle haben Artefakte gelernt, die sich bei den Testdaten nicht bestätigen. Die Einschränkung auf einige „wichtige“ Variablen scheint hier im Sinne der Generalisierung von Vorteil zu sein. Die logistische Regression zeigt sich im Allgemeinen sehr stabil. Sowohl bei den Entscheidungs-

baumverfahren als auch bei der logistischen Regression zählen die Variablen Nr. 8, 22 und 53 zu den wichtigsten. Interessanterweise tritt die Variable Nr. 59 bei den Entscheidungsbäumen sehr häufig in der ersten bzw. zweiten Ebene auf, während diese Variable bei der logistischen Regression relativ unbedeutend ist.

Bei den Entscheidungsbaumverfahren und der logistischen Regression stellt sich die Frage, ob diese Einschränkung der Variablenzahl die Interpretation der einzelnen Modellvarianten erleichtert. Dazu dient eine Datenmatrix, die wie in Kapitel 7.1 (siehe S. 155) in den Zeilen die einzelnen Modellvarianten und in den Spalten die unabhängigen Variablen enthält (siehe Anhang G, S. 218; Anhang H, S. 219). Die Werte für die einzelnen Elemente der Datenmatrix bestimmen sich wiederum nach der Wichtigkeit der Variablen. Dabei wird je Modellvariante angegeben, ob und mit welchem Einfluss eine bestimmte unabhängige Variable in der entsprechenden Modellvariante vorkommt. Bei den Entscheidungsbaumverfahren wird der Variable, die zum ersten Split verwendet wird, der Wert „1“ zugeordnet, kommt sie erst in der fünften Ebene zum ersten Mal vor, wird ihr der Wert „5“ zugeordnet. Bei der logistischen Regression erhält die Variable mit dem höchsten Wert der Wald-Statistik eine „1“, die Variable mit dem zweithöchsten Wert eine „2“, und so weiter.

Bei Betrachtung der Datenmatrix bei der logistischen Regression (siehe Anhang G, S. 218) zeigt sich, dass die Variablen Nr. 8 und 53 in nahezu allen Modellen mit derselben Wichtigkeit verwendet werden. Die Variablen Nr. 3, 57 und 59 werden ebenfalls entweder nicht oder mit einer eher geringen Wichtigkeit verwendet. Die Variablen Nr. 22 und 24 werden häufig verwendet, allerdings treten hier Unterschiede bezüglich der Wichtigkeit auf. Insgesamt zeichnet sich jedoch bereits ein relativ einheitliches Bild ab.

Bei den Entscheidungsbaumverfahren (siehe Anhang H, S. 219) lässt sich aus der Datenmatrix erkennen, dass es scheinbar drei Hauptbäume gibt. Wenn der erste Split mit Variable Nr. 22 oder Variable Nr. 53 durchgeführt wird, folgen darauf meist die Variablen Nr. 59 und Nr. 8. Wird mit Variable Nr. 59 begonnen, so folgen danach häufig die Variablen Nr. 22 und Nr. 53. Somit werden in den oberen Baumbereichen häufig diese vier Merkmale verwendet. Die Variablen Nr. 3 und Nr. 24 werden häufig benutzt, allerdings immer nur in unteren Baumregionen.

Im folgenden wird eine weitere Einschränkung bei den unabhängigen Variablen durchgeführt, um die weitere Entwicklung sowohl bei den Werten aus dem Gains-Chart als auch bei der Variablenwahl je Modellvariante bei der logistischen Regression und den

Entscheidungsbäumen zu untersuchen. Dabei werden nun die Variablen Nr. 8, 22, 53 und 59 verwendet, da diese Variablen bei diesen Verfahren am häufigsten verwendet wurden.

Wiederum sind die Ergebnisse bei der logistischen Regression relativ robust über alle Modellvarianten (siehe Tabelle 30), auch wenn die Werte aus dem Gains-Chart nun größtenteils etwas unter den vergleichbaren Werten der Modellvarianten mit sieben Variablen liegen (vgl. Tabelle 25, S. 170).

	10%	20%	40%	80%
alle	32	43	69	93
profit	32	43	69	93
dupl	32	44	69	93
z 1:10 1	32	44	70	97
z 1:10 2	31	42	68	97
z 1:10 3	31	42	68	97
cl 1:10 1	33	45	68	93
cl 1:10 2	33	45	68	93
cl 1:10 3	32	44	69	93
z 1:5 1	33	44	69	92
z 1:5 2	32	42	68	93
z 1:5 3	32	44	69	93
cl 1:5 1	34	43	67	93
cl 1:5 2	33	45	68	93
cl 1:5 3	32	43	69	93
z 1:1 1	31	49	67	93
z 1:1 2	32	41	67	93
z 1:1 3	43	50	68	88
cl 1:1 1	34	49	68	93
cl 1:1 2	31	49	68	93
cl 1:1 3	34	49	67	93
cl all 1:1 1	21	35	61	94
cl all 1:1 2	21	35	60	94
cl all 1:1 3	34	52	66	93
cl med 1:1 1	34	44	64	98
cl med 1:1 2	36	44	64	98
cl med 1:1 3	34	46	65	98

Tabelle 30: Gains-Chart Ergebnisse auf Basis der Testdaten bei der logistischen Regression (3)

Betrachtet man die verwendeten Variablen, so zeigt sich ein sehr einheitliches Bild, das heißt bei vielen Modellvarianten werden die drei Variablen in einer ähnlichen Reihenfolge bezüglich der Wald-Statistik verwendet (siehe Tabelle 31). Es zeigt sich, dass Variable Nr. 8 am häufigsten den höchsten Wald-Wert erhält, gefolgt von Variable Nr. 53 und Nr. 22. Interessanterweise hat sich die Reihenfolge der beiden wichtigsten Variablen im Vergleich zu Tabelle 26 (siehe S. 170) gedreht. Die Wald-Werte der beiden Variablen Nr. 8 und 53 liegen allerdings immer sehr nahe zusammen, so dass man davon

ausgehen kann, dass diese beiden Merkmale eine ähnliche Bedeutung bezüglich dieser Maßgröße haben.

	ma_zahl	di_ebest	anz_7409
	Nr. 8	Nr. 22	Nr. 53
Nr. 1	20	2	4
Nr. 2	4	2	22
Nr. 3	1	23	1

Tabelle 31: Wichtigkeit der drei verwendeten unabhängigen Variablen bei der logistischen Regression

Die Datenmatrix, die die Wichtigkeit der einzelnen Variablen je Modellvariante beschreibt, zeigt ebenfalls ein eindeutiges Bild (siehe Anhang I, S. 220). Die meisten Modellvarianten verwenden alle drei Variablen in der Reihenfolge Nr. 8, gefolgt von Nr. 53 und Nr. 22.

Entscheidungsbäume erzielen bei Verwendung dieser vier Variablen bei einigen Varianten sehr gute Ergebnisse (siehe Tabelle 32), z.B. bei „cl 1:1“. Insgesamt gesehen tritt nun jedoch eine sehr große Streuung auf und einige Modellvarianten erzielen deutlich schlechtere Ergebnisse als bei der Verwendung von sieben Variablen (vgl. Tabelle 27. S. 171). Ein Grund dafür könnte sein, dass die verbliebenen Variablen zwar in den oberen Baumebenen eine sinnvolle Aufteilung der Daten finden, die allerdings immer noch zu grob ist, um gute Ergebnisse zu erzielen. Beispielsweise fehlen die Variablen Nr. 3 und Nr. 24, die häufig nur in den unteren Baumebenen verwendet wurden, dort jedoch möglicherweise die entscheidende Zuordnung der Objekte vornehmen.

	10%	20%	40%	80%
alle	30	43	64	89
profit	40	52	74	99
dupl	42	55	74	98
z 1:10 1	32	46	64	88
z 1:10 2	34	50	64	100
z 1:10 3	35	49	68	99
cl 1:10 1	40	50	65	90
cl 1:10 2	43	53	75	100
cl 1:10 3	40	52	64	100
z 1:5 1	29	46	63	92
z 1:5 2	37	46	66	100
z 1:5 3	33	57	73	100
cl 1:5 1	28	35	52	85
cl 1:5 2	34	49	72	100
cl 1:5 3	34	48	67	92
z 1:1 1	15	30	64	99
z 1:1 2	14	27	64	99
z 1:1 3	19	37	64	97
cl 1:1 1	45	55	75	97
cl 1:1 2	37	54	74	99
cl 1:1 3	42	55	75	99
cl all 1:1 1	40	52	73	97
cl all 1:1 2	40	50	70	99
cl all 1:1 3	32	47	73	98
cl med 1:1 1	51	56	67	99
cl med 1:1 2	38	54	73	100
cl med 1:1 3	43	53	68	100

Tabelle 32: Gains-Chart Ergebnisse auf Basis der Testdaten bei Entscheidungsbäumen (3)

Bei Betrachtung der verwendeten Variablen zeigt sich kein eindeutiges Bild (siehe Tabelle 33). Am häufigsten wird Variable Nr. 59 zum ersten Split verwendet, Variable Nr. 8 wird nur in der zweiten und dritten Baumebene herangezogen während Variable Nr. 53 in allen Ebenen verwendet wird. Variable Nr. 22 kann ebenfalls in allen Baumebenen auftreten, allerdings wird sie nicht immer verwendet. Während vorher drei Variablen (Nr. 8, 22, 59) zum ersten Split verwendet wurden (vgl. Tabelle 28), kann nun eine Reihenfolge erkannt werden.

	ma_zahl	di_ebest	anz_7409	di_l_wa
	Nr. 8	Nr. 22	Nr. 53	Nr. 59
Nr. 1	0	3	9	15
Nr. 2	11	2	8	11
Nr. 3	15	5	5	0

Tabelle 33: Wichtigkeit der vier verwendeten unabhängigen Variablen bei den Entscheidungsbaumverfahren

Bei Betrachtung der Datenmatrix je Modellvariante zeigt sich, dass meist drei verschiedene Baumvarianten entstehen (siehe Anhang J, S. 221). Wird mit einem Split nach Variable Nr. 59 begonnen, so folgen entweder die Variablen Nr. 8 und Nr. 53 oder Nr. 53 und Nr. 8. Manche Bäume beginnen mit Variable Nr. 53, danach folgen meist Nr. 8 und Nr. 59. Bei Verwendung der sieben Variablen wurde Variable Nr. 22 häufig zum ersten Split bzw. in der zweiten Ebene verwendet (siehe Tabelle 28, S. 172), bei Verwendung von nur vier Variablen wird sie nur selten zum ersten Split und bei einigen Baumvarianten überhaupt nicht verwendet. Dies wird häufig auch als Instabilität bei Entscheidungsbäumen bezeichnet (siehe S. 103). Somit stellt sich gerade bei diesem Verfahren die Frage, welcher Baum als ideal anzusehen ist und auch zur Interpretation verwendet werden kann.

Insgesamt zeichnet sich bei der logistischen Regression und den Entscheidungsbaumverfahren eine unterschiedliche Reihenfolge der entscheidenden Variablen ab. Bei der logistischen Regression wurde zur Beurteilung der Wichtigkeit einer Variablen ein statistisches Kriterium, bei den Entscheidungsbaumverfahren die Reihenfolge des Auftretens in der Baumhierarchie verwendet. Insgesamt kristallisieren sich folgende vier Variablen als geeignet zur Beschreibung der Zielgruppe bzw. zur Attraktivitätseinstufung eines Kunden für das beworbene Produkt heraus: Nr. 8, Nr. 22, Nr. 53 und Nr. 59. Diese Variablen werden je Verfahren und Anzahl verwendeter Variablen in unterschiedlicher Reihenfolge zur Bewertung herangezogen, jedoch liefern sie insgesamt den größten Erklärungsbeitrag, so dass eine geeignete Zielgruppe identifiziert und beschrieben werden kann. Die attraktivsten Kunden scheinen insgesamt eher weniger Werbeaktionen zu dem beworbenen Produkt bisher erhalten zu haben (Nr. 53), die erste Bestellung liegt noch nicht allzu lange zurück (Nr. 22), ebenso die letzte erhaltene Werbeaktion (Nr. 59) und es sind vor allem Unternehmen mittlerer Größe (Nr. 8). Die Marketingabteilung kann anhand dieses Wissens nun eine geeignete Zielgruppe selektieren und sowohl die Werbemittel als auch das Anschreiben entsprechend anpassen.

7.6 Zusammenfassung

Betrachtet man die Ergebnisse der drei angewandten Verfahren im Gesamten, so zeigt sich, dass eine Änderung der Anteile von 1-Klasse und 0-Klasse in der Zielvariablen häufig zu besseren Ergebnissen führt. Bei der logistischen Regression erzielt die Vorge-

hensweise „alle“ beispielsweise ebenfalls sehr gute Ergebnisse. Weiterhin zeigt sich der zusätzliche Vorverarbeitungsschritt, die clusteranalysegestützte Auswahl der Nichtbesteller beim Downsizing als nützlich.

Nachfolgende Tabelle gibt einen Überblick über die jeweils beste erreichte Modellgüte beim 10%- bzw. 80%-Wert aus dem Gains-Chart:

Verfahren	10%-Wert		80%-Wert	
	Beste Variante	Gemittelte Werte	Beste Variante	Gemittelte Werte
Entscheidungsbaum (siehe S. 107)	44	41	100	99
Logistische Regression (siehe S. 125)	37	35	98	97
Neuronale Netze (siehe S. 147)	37	35	97	94

Tabelle 34: Gütevergleich aller Modelle

Aus Tabelle 34 ist ersichtlich, dass die Entscheidungsbäume in dieser Arbeit den anderen Verfahren überlegen sind. Häufig gibt es zwischen den einzelnen Verfahren bei „idealer“ Parameterwahl kaum Unterschiede (Liehr, 2000, S. 737). Im Allgemeinen hängt es allerdings sehr stark von den Daten ab, welches Verfahren die besten Ergebnisse erzielt. Deshalb ist es sinnvoll, unterschiedliche Verfahren und Varianten zu vergleichen und dann aufgrund der Ergebnisse bei den Testdaten das Beste auszuwählen. In dieser Studie fällt die Wahl auf das Entscheidungsbaumverfahren mit der Methode „profit“.

Eine Vorgehensweise zur Verbesserung der Gesamtprognose, die hier allerdings nicht näher ausgeführt wird, wäre beispielsweise, die geschätzten y-Werte der einzelnen Modellvarianten wiederum als unabhängige Variablen in eine logistische Regression zu geben, um so eine weitere Schätzung des tatsächlichen y-Werts durchzuführen und auf diese Weise eine „Gewichtung“ der einzelnen Modellvarianten zu erhalten.

8. Zusammenfassung und Ausblick

Zum Abschluss werden die wichtigsten Ergebnisse dieser Arbeit nochmals zusammengefasst und ein kurzer Ausblick auf mögliche zukünftige Forschungsaktivitäten im Bereich KDD gegeben.

Im ersten Teil der Arbeit, das heißt im ersten und zweiten Kapitel, wurden die grundlegenden Begriffe definiert. Nachdem im ersten Kapitel Definitionen zu Marketing, Direktmarketing, Databasemarketing, CRM und Responseoptimierung erfolgten, wurden im Anschluss die Firma WEKA MEDIA und die spezielle Problemstellung mit der Zielrichtung vorgestellt. Im folgenden Kapitel erfolgte die Definition der Begriffe Datawarehouse, Knowledge Discovery in Databases auf der informationstechnischen Seite. Weiterhin wurden die methodischen Grundlagen, zu denen die traditionellen Kundenbewertungsverfahren, sowie Data Mining und OLAP zählen, vorgestellt. Zum Abschluss wurde der Zusammenhang zwischen den einzelnen Bereichen hergestellt.
Der zweite Teil der Arbeit beschäftigte sich mit der empirischen Analyse. Dabei wurde jeweils zuerst der theoretische Hintergrund jedes Verfahrens dargestellt, bevor die Ergebnisse diskutiert wurden. Im dritten Kapitel erfolgte die Voranalyse, das heißt die Vorstellung der Datenmatrix und die Datenvorverarbeitung: Variablenmodifikation, Ersetzung fehlender Werte, Variablenreduktion und Ausreißer-Elimination, sowie die Aufteilung der Datenmatrix. Im Anschluss wurden verschiedene Methoden zur Bewältigung niedriger Responsequoten und ein neuer Ansatz zur Verbesserung der Stichprobenziehung ausführlich vorgestellt.
In den Kapiteln 4, 5 und 6 wurden jeweils die verwendeten Modelle, das heißt Entscheidungsbaumverfahren, logistische Regression und Neuronale Netze, und die damit erzielten empirischen Ergebnisse vorgestellt.
Im siebten Kapitel erfolgte eine räumlich Darstellung der einzelnen Methoden mit Hilfe einer Repräsentation, um die Auswirkung der clusteranalysegestützten Stichprobenziehung und den Zusammenhang zwischen Lage der Modellvariante und Modellgüte zu untersuchen. Weiterhin wurde die Auswirkung sowohl auf die Modellergebnisse als auch auf die Modelleinflussgrößen untersucht, wenn die Anzahl unabhängiger Variablen reduziert wird.

Die Analysen haben gezeigt, dass je Modell und Methode deutliche Unterschiede bezüglich der Bewältigung niedriger Responsequoten existieren. Weiterhin zeigte sich bei dem Vergleich der Modellvarianten mit Hilfe der MDS, dass der zusätzlich eingeführte Vorverarbeitungsschritt, die clusteranalysegestützte Stichprobenziehung, vorteilhaft ist. Es konnte zu jedem Verfahren eine bestimmte Methode empfohlen werden. Gerade aufgrund der Variablenreduktion konnten einige Variablen identifiziert werden, die eine Beschreibung der Zielgruppe ermöglichen.

Insgesamt erwies sich die Methode „alle“ nur bei der logistischen Regression als geeignet, die Methode „profit“ lieferte sowohl bei der logistischen Regression als auch bei den Entscheidungsbäumen sehr gute Ergebnisse mit dem Vorteil, dass hier nur ein Ergebnis entsteht. Bei den Methoden, die den Anteil der Besteller in der Zielvariablen ändern, war das Duplizieren dem Downsizing nicht generell überlegen. Insgesamt zeigte sich, dass bei der Methode Downsizing die clusteranalysegestützte Stichprobenziehung zu stabileren Ergebnissen und besserer Modellgüte als bei einfacher Zufallsauswahl führt. Innerhalb der clusteranalysegestützten Auswahl der Daten brachten die beiden zusätzlich untersuchten Methoden „cl all 1:1“ und „cl med 1:1“ im Allgemeinen keine Verbesserung zu „cl 1:1“.
Die logistische Regression zeigte sich erwartungsgemäß robust bei niedrigen Responsequoten. Bei Entscheidungsbäumen ist dagegen die Methode „profit“ oder eine Downsizing-Technik bei einer sehr selten auftretenden Klasse zu bevorzugen. Bei Neuronalen Netzen sollte in diesem Fall ebenfalls ein Downsizing durchgeführt werden.
Im Allgemeinen empfiehlt es sich bei einer Datensituation mit sehr niedrigen Responsequoten, verschiedene Modelle und Methoden anzuwenden und auf Basis der Ergebnisse bei den Testdaten eine Variante auszuwählen. Sinnvolle Methoden sind dabei vor allem „alle“, „profit“, „cl 1:10“, „cl 1:5“ und „cl 1:1“.
Es zeigte sich außerdem, dass mit Hilfe mathematischer und statistischer Verfahren deutliche Verbesserungen bei der Adressauswahl möglich sind. Wie die empirischen Ergebnisse belegen, kann bei Aussendung der besten 10% aller Adressen in diesem Fall eine bis zu viermal höhere Responsequote im Vergleich zu einer Zufallsauswahl erzielt werden. Beim 20%-Wert werden ebenfalls noch respektable Steigerungen erreicht. Bei Weglassen der 20% unattraktivsten Adressen werden bei manchen Verfahren trotz allem nahezu alle Besteller erreicht.

Ein Feld, in welchem weiterer Forschungsbedarf existiert, wäre beispielsweise bei Entscheidungsbaumverfahren, eine Möglichkeit zur Interpretation von mehreren Entscheidungsbäumen zu finden. Werden mehrere Bäume mit unterschiedlichen Datenausschnitten gebildet, z.B. durch wiederholte Stichprobenziehung oder Methoden wie Bagging, werden meist unterschiedliche unabhängige Variablen beim Baumaufbau verwendet. Interessant wäre, aus diesen verschiedenen Bäumen eine Wichtigkeitsreihenfolge der unabhängigen Variablen und eine Interpretationshilfe abzuleiten.

Eine Unterstützung des Anwenders bei der Frage nach der optimalen Modellkomplexität, das heißt wie viele und welche unabhängigen Variablen idealerweise zu verwenden sind, ist ebenfalls ein wichtiges Themengebiet.

Die Entwicklung effizienter Stichprobentechniken bzw. Repräsentationstechniken wäre ein weiteres geeignetes Forschungsvorhaben, um aus großen Datenmengen „geschickt" wenige repräsentative Objekte zu ermitteln.

Auch die Untersuchung über den Umgang mit verschiedenen Scorewerten zählt dazu: Wenn beispielsweise ein Objekt von einem Entscheidungsbaumverfahren einen sehr hohen und von einer logistischen Regression nur einen sehr niedrigen Scorewert erhält.

Literaturverzeichnis

Ackley, D. E. / Hinton, G. E. / Sejnowski, T. J. (1985): A learning algorithm for Boltzmann machines, in: Cognitive Science, 9, S. 147-169.

Agrawal, R. / Imilienski, T. / Swami, A. (1993): Database Mining: A Performance Perspective, IEEE Transactions on Knowledge and Data Engineering, Vol. 5, No. 6, S. 914-925.

Agrawal, R. / Srikant, R. (1994): Fast Algorithms for Mining Association Rules, Proceedings of the 20th VLDB Conferences, S. 487-499.

Agresti, A. / Finlay, B. (1997): Statistical Methods for the Social Sciences, Prentice Hall, Upper Saddle River.

Aldrich, J. / Nelson, F. (1984): Linear Probability, Logit, and Probit Models, Sage Publications, Beverly Hills.

Alex, B. (1998): Künstliche Neuronale Netze in Mangementinformations-systemen: Grundlagen und Einsatzmöglichkeiten, Gabler, Wiesbaden.

Allison, P. D. (1999): Logistic Regression Using the SAS System: Theory and Application, SAS Institute Inc., Cary.

Anahory, S. / Murray, D. (1997): Data Warehouse: Planung, Implementierung und Administration, Addison-Wesley, Bonn, S. 19-23.

Anderberg, M. R. (1973): Cluster Analysis for Applications, Academic Press, New York.

Anders, U. (1995): Neuronale Netze in der Ökonometrie: Die Entmythologisie-rung ihrer Anwendung, Discussion Paper No. 95-26, Mannheim: Zentrum für europäische Wirtschaftsforschung.

Anderson, J. A. (1972): Separate Sample Logistic Discrimination, Biometrika, 59, S. 19-35.

Anderson, T. (1958): An Introduction to Multivariate Statistical Analysis, Wiley & Sons, New York.

Arndt, D. / Gersten, W. / Wirth, R. (2001): Kundenprofile zur Prognose der Markenaffinität im Automobilsektor, in: Handbuch Data Mining im Marketing, Hippner, H. / Küsters, U. / Meyer, M. / Wilde, K. (Hrsg.), Vieweg, Braunschweig/Wiesbaden, S. 591-606.

Backhaus, K. / Erichson, B. / Plinke, W. / Weiber, R. (2000): Multivariate Analysemethoden. Eine anwendungsorientierte Einführung, Springer, Berlin [u.a.].

Baetge, J. / Uthoff, C. (1998): Entwicklung eines Bonitätsindexes auf der Basis von Wirtschaftsauskünften der Vereine Creditreform mit Künstlichen Neuronalen Netzen, in: Data Mining: theoretische Aspekte und Anwendungen, Nakhaeizadeh, G. (Hrsg.), Physika, Heidelberg, S. 289-308.

Bamberg, G. / Baur, F. (1998): Statistik, 10. Auflage, Oldenbourg, München/Wien.

Bankhofer, U. (1995): Unvollständige Daten- und Distanzmatrizen in der Multivariaten Datenanalyse, Eul, Bergisch-Gladbach [u.a.].

Bauer, E. / Kohavi, R. (1999): An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants, Machine Learning, Vol. 36, No. 1-2, S. 105-139.

Baun, S. (1994): Neuronale Netze in der Aktienkursprognose, in: Neuronale Netze in der Ökonomie: Grundlagen und finanzwirtschaftliche Anwendungen, Rehkugler, H./Zimmermann, H. G. (Hrsg.), Vahlen, München, S. 131-208.

Bausch, T. (1991): Gewinnoptimale Kundenselektion im Direktmarketing, in: Marketing ZFP, Nr. 2, 13.&14. Jg., S. 86-96.

Bausch, T. / Opitz, O. (1993): PC-gestützte Datenanalyse mit Fallstudien aus der Marktforschung, Vahlen, München.

Behrens, K. C. (1963): Absatzwerbung, Gabler, Wiesbaden.

Berekoven, L. / Eckert, W. / Ellenrieder, P. (2001): Marktforschung: methodische Grundlagen und praktische Anwendungen, 9. Auflage, Gabler Verlag, Wiesbaden.

Berry, M. J. A. / Linoff, G. (1997): Data Mining techniques for Marketing, Sales and Customer Support, Wiley, New York.

Berry, M. J. A. / Linoff, G. (2000): Mastering Data Mining: The Art and Science of Customer Relationship Management, Wiley, New York.

Bird, D. (1989): Commonsense Direct Marketing, 2nd ed., Kogan Page, London.

Bishop, C. M. (1995), Neural Networks for Pattern Recognition, Oxford University Press, Oxford.

Bonne, T. (1999): Kostenorientierte Klassifikationsanalyse, Eul, Lohmar [u.a.].

Bonne, T. / Arminger, G. (2001) : Der Einsatz automatischer Klassifikation zur Bonitätsprüfung im Direktvertrieb, in: Handbuch Data Mining im Marketing, Hippner, H./Küsters, U./Meyer, M./Wilde, K. (Hrsg.), Vieweg, Braunschweig/Wiesbaden, S. 653-670.

Borgelt, C. / Kruse, R. (1998): Attributauswahlmaße für die Induktion von Entscheidungsbäumen: Ein Überblick, in: Data Mining: theoretische Aspekte und Anwendungen, Nakhaeizadeh, G. (Hrsg.), Physika, Heidelberg, S. 77-98.

Breiman, L. / Friedman, J. H. / Olshen, R. A. / Stone, C. J. (1984): Classification and Regresion Trees, Chapman & Hall, New York.

Breiman, L. (1994): Heuristics of Instability and stabilization in model selection, Technical Report 416, Department of Statistics, University of California, Berkeley.

Breiman, L. (1996a): Technical Note: Some Properties of Splitting Criteria, Machine Learning, 24, S. 41-47.

Breiman, L. (1996b): Bagging Predictors, Machine Learning, 24, S. 123-140.

Breiman, L. (1998): Arcing Classifiers, Annals of Statistics, 26, S. 801-849.

Breiman, L. (2001): Random Forests, Machine Learning, 45, No. 1, S. 5-32.

Breitschuh, J. (2001): Versandhandelsmarketing: Aspekte erfolgreicher Neukundengewinnung, Oldenbourg, München/Wien.

Brosius, F. (1998): SPSS 8.0: Professionelle Statistik unter Windows, MITP, Bonn.

Bühl, A. / Zöfel, P. (1998): SPSS für Windows Version 7.5: Praxisorientierte Einführung in die moderne Datenanalyse, 4. Auflage, Addison-Wesley, Bonn [u.a.].

Buja, A. / Lee, Y.-S., (2001): Data Mining Criteria for Tree-Based Regression and Classification, in: KDD 2001: Proceedings of the seventh ACM SIGKDD International Conference on Knowledge Disvovery and Data Mining, S. 27-36.

Cabena, P. / Hadjinian, P. / Stadler, R. / Verhees, J. / Zanasi, A. (1998): Discovering data mining – from concept to implementation, Prentice Hall, Upper Saddle River.

Chamoni, P. / Budde, C. (1997): Methoden und Verfahren des Data Mining, Diskussionsbeiträge des Fachbereichs Wirtschaftswissenschaft der Gesamthochschule Duisburg, Nr. 232.

Chan, P. / Stolfo, S. (1998) : Toward scalable Learning with Non-uniform Class and Cost Distribution : A Case Study in Credit Card Fraud Detection, in : Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining, S. 164-168.

Chapman, P. / Clinton, J. / Hejlesen, J. H. / Kerber, R. / Khabaza, T. / Reinartz, T. / Wirth, R. (1998): The Current CRISP-DM Process Model for Data Mining, Discussion Paper for the Second SIG Meeting, London.

Codd, E. F. / Codd, S. B. / Sally, C. T. (1993): Providing OLAP (on-line Analytical processing) to user-analysts – an IT mandat, White Paper, E. F. Codd & Associates.

Cohen, W. A. (1985): Building a Mail Order Business, 2. Auflage, Wiley, New York.

Cox, D. R. (1970): The Analysis of Binary Data, Methuen, London.

Cox, D. R. / Hinkley, D. V. (1974): Theoretical Statistics, Chapman Hall, London.

CRISP-DM, o. V. (2003): CRISP, www.crisp-dm.org/Process/index.htm, Stand: 01.10.2003.

Curry, B. / Rumelhart, D. E. (1990) : Msnet : A neural network that classifies mass spectra, Tetrahedron Computer Methodology, 1990, 3, S. 213-237.

Dallmer, H. (1997): System des Direct-Marketing – Entwicklung und Zukunfts-perspektiven, in: Handbuch Direct Marketing, Dallmer, H. (Hrsg.), 6. Auflage, Gabler, Wiesbaden, S. 3-19.

Dallmer, H. (2002): Das System des Direct-Marketing – Entwicklungsfaktoren und Trends, in: Handbuch Direct Marketing, Dallmer, H. (Hrsg.), 8. Auflage, Gabler, Wiesbaden, S. 3-33.

Dasarathy, B.V. (Hrsg) (1991): Nearest Neighbor (NN) Norms: NN Pattern, Classification Techniques, IEEE Computer Society Press, Los Alamitos.

DeRouin, E. / Brown, J. / Beck, H. / Fausett, L. / Schneider, M. (1991): Neural Network Training on Unequally Represented Classes, in: Intelligent Engineering Systems Through Artificial Neural Networks, Dagli, C.H. / Kumara, S.R.T. / Shin, Y.C. (Hrsg.), ASME Press, New York, S. 135-145.

Dietterich, T. G. (2000): An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization, Machine Learning, 40 (2), S. 139-158.

Diller, H. (1992): Response, in: Vahlens Großes Marketinglexikon, Diller, H. (Hrsg.), Beck/Vahlen, München, S. 1013.

Dorffner, G. (1991): Konnektionismus: Von neuronalen Netzwerken zu einer „natürlichen“ KI, Teubner, Stuttgart.

Düsing, R. (1999): Knowledge Discovery in Databases und Data Mining, in: Analytische Informationssysteme, Chamoni, P./Gluchowski P. (Hrsg.), 2. Aufl., Springer-Verlag, Berlin [u.a.], S. 345-354.

Edelstein, H. (1999): Introduction to Data Mining and Knowledge Discovery, Third Edition, Two Crows Corporation, Potomac.

Eherler, D. / Lehmann, T. (2001): Responder Profiling with CHAID and dependency analysis, in: ECML/PKDD-01 Workshop: Data Mining for Marketing Applications, Gersten, W. / Vanhoof, K. (Hrsg.), Freiburg, S. 49-58.

Enache, D. (1998): Künstliche neuronale Netze zur Kreditwürdigkeitsüberprüfung von Konsumentenkrediten, Eul, Lohmar [u.a.].

Esposito, F. / Malerba, D. / Semeraro, G. (1997): A Comparative Analysis of Methods for Pruning Decision Trees, in: IEEE Transactions on Pattern Analysis and Machine Learning, Vol. 19, No. 5 S. 476-491.

Ester, M. / Sander, J. (2000): Knowledge Discovery in Databases: Techniken und Anwendungen, Springer, Berlin [u.a.].

Ezawa, K. / Singh, M. / Norton, S. (1996): Learning Goal Oriented Bayesian Networks for Telecommunications Risk Management, in: Proceedings of the 13th International Conference on Machine Learning, S. 139-147.

Fahrmeir, L. / Hamerle, A. (1996): Grundlegende multivariate Schätz- und Testprobleme, in: Multivariate statistische Verfahren, Fahrmeir, L. / Hamerle, A. / Tutz, G. (Hrsg.), Walter de Gruyter, Berlin/New York, S. 49-92.

Fawcett, T. / Provost, F. (1996) : Combining Data Mining and Machine Learning for Effective User Profiling, in: Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining, S. 8-13.

Fayyad, U. M. / Piatetsky-Shapiro, G. / Smyth, P. (1996): From Data Mining to Knowledge Discovery: An overview, in: Advances in Data Mining and Knowledge Discovery, Fayyad, U. M. / Piatetsky-Shapiro, G. / Smyth, P. / Uthurusamy, R. (Hrsg.), MIT Press, Cambridge 1996, S. 1-34.

Feng, C. / Michie, D. (1994): Machine Learning of Rules and Trees, in: Machine Learning, Neural and Statistical Classification, Michie, D. / Spiegelhalter, D. J. / Taylor, C. C. (Hrsg.), Ellis Horwood, New York [u.a.], S. 50-83.

Frawley, W. F. / Piatetsky-Shapiro, G. / Matheus, C. J. (1991): Knowledge Discovery in Databases: an overview, in: Knowledge Discovery in Databases, Piatetsky-Shapiro, G. / Frawley, W. F. (Hrsg.), Menlo Park, S. 1-27.

Freund, Y. / Schapire, R. E. (1996): Experiments with a new boosting algorithm, in: Proceedings of the 13th International Conference on Machine Learning, S. 148-156.

Garbe, C. et al. (1995) : Primary cutaneous melanoma : Identification of Prognostic Groups and Estimation of Individual Prognosis for 5093 Patients, in: Cancer, 75, S. 2484-2491.

Gebhardt, F. (1994): Interessantheit als Kriterium für die Bewertung von Ergebnissen, Informatik Forschung und Entwicklung, Vol. 9, S. 9-21.

Gierl, H. (1995): Marketing, Kohlhammer, Stuttgart/Berlin/Köln.

Gierl, H. / Koncz, J. (2002): Customer Lifetime Value, in: Handbuch Direct Marketing, Dallmer, H. (Hrsg.), 8. Auflage, Gabler, Wiesbaden, S. 939-956.

Gray, N. (1976): Constraints on learning machine classification methods, Analytical Chemistry, 48 (14), S. 2265-2268.

Guo, H. / Murphey, Y. (2001): Neural Learning From Unbalanced Data Using Noise Modeling, Lecture Notes in Computer Sciences, Vol. 2070, S. 259-268.

Harrell, F. E. / Lee, K. L. / Mark, D. B. (1996): Multivariate Prognostic Models: Issues in Developing Models, Evaluating Assumptions and Adequacy, and Measuring and Reducing Errors, Statistics in Medicine, 15 (4), S. 361-387.

Harrell, F. E. (2001): Regression modeling strategies: with applications to linear models, logistic regression, and survival analysis, Springer, New York.

Hartigan, J. A. (1975): Clustering Algorithms, John Wiley & Sons, New York.

Hartung, J. / Elpelt, B. (1995): Multivariate Statistik, 5. Auflage, Oldenbourg, München [u.a.].

Hauck, W. W. / Donner, A. (1977): Wald's Test as Applied to Hypothesis in Logit Analysis, in: Journal of the American Statistical Association, 12/1977, Vol. 72, S. 851-853.

Hebb, D. O. (1949): The Organization of Behaviour: A Neuropsychological Theory, John Wiley & Sons, New York/London.

Hilbert, A. / Dittmar, T. (1997): Bonitätsprüfung von Firmenkunden mit Hilfe Küstlicher Neuronaler Netze, Arbeitspapiere zur Mathematischen Wirtschaftsforschung, Universität Augsburg, Heft 156.

Hilbert, A. (1998): Zur Theorie der Korrelationsmaße, Eul, Lohmar [u.a.].

Hippner, H. / Rupp, A. (2001): Kreditwürdigkeitsprüfung im Versandhandel, in: Handbuch Data Mining im Marketing, Hippner, H. / Küsters, U. / Meyer, M. / Wilde, K. (Hrsg.), Vieweg, Braunschweig/Wiesbaden, S. 685-706.

Hippner, H. / Wilde, K. (2001a): Der Prozess des Data Mining im Marketing, in: Handbuch Data Mining im Marketing, Hippner, H. / Küsters, U. / Meyer, M. / Wilde, K. (Hrsg.), Vieweg, Braunschweig/Wiesbaden, S. 22-94.

Hippner, H. / Wilde, K. (2001b): CRM – ein Überblick, in: Effektives Customer Relationship Management Instrumente – Einführungskonzepte – Organisation, Helmke, S. / Dangelmaier, W. (Hrsg.), Vieweg, Braunschweig/Wiesbaden, S. 3-38.

Hofmann, H.-J. (1990): Die Anwendung des CART-Verfahrens zur statistischen Bonitätsanalyse von Konsumentenkrediten, ZfB, 60. Jg., Heft 9, S. 941-961.

Holland, H. (1992): Direktmarketing, Vahlen, München.

Hopfield, J. J. (1982): Neural networks and physical systems with emergent collective computational abilities, in: Proceedings of the National Academy of Sciences, 79, S. 2554-2558.

Hopfield, J. J. (1984): Neurons with graded response have collective computational properties like those of two-state neurons, in: Proceedings of the National Academy of Sciences, 81, S. 3088-3092.

Hosmer, D. W. / Lemeshow, S. (1989): Applied Logistic Regression, John Wiley, New York [u.a.].

Hruschka, H. (1991): Einsatz künstlicher neuraler Netz zur Datenanalyse im Marketing, in: Marketing ZFP, Heft 4, IV. Quartal, S. 217-225.

Hughes, A. M. (1996): The complete database marketer: second-generation strategies and techniques for tapping the power of your customer database, Irwin, Chicago [u.a.].

Huldi, Ch. (1992): Database-Marketing, Thexis, St. Gallen 1992.

Huldi, Ch. (1997): Mittels Datenanalyse und Kundenbewertung zu Effektivität im (Direct-) Marketing, in: Handbuch Direct-Marketing, Dallmer, H., 7. Auflage, Gabler, Wiesbaden 1997, S. 603-617.

Inmon, W. (1996): Building the Data Warehouse, John Wiley & Sons, New York.

Ittner, A. / Sieber, H. / Trautzsch, S. (2001): Nichtlineare Entscheidungsbäume zur Optimierung von Direktmailingaktionen, in: Handbuch Data Mining im Marketing, Hippner, H. / Küsters, U. / Meyer, M. / Wilde, K. (Hrsg.), Vieweg, Braunschweig/Wiesbaden, S. 707-723.

Japkowicz, N. (2000): The Class Imbalance Problem: Significance and Strategies, in: Proceedings of the 2000 International Conference on Artificial Intelligence, Vol. 1, S. 111-117.

Kass, G. V. (1980): An exploratory technique for investigating large quantities of categorical data, Applied Statistics, Vol. 29, No. 2, S. 119-127.

Kirchner, G. (1985): Moderne Direktmarketing-Methoden und die Zukunft des Dialogmarketing, in: Direktmarketing leicht gemacht, Richter von Proeck, M. / Dichter, E. / Schweiger, G. / Kirchner, G. (Hrsg.), Moderne Industrie, Landsberg am Lech.

Kleinbaum, D. G. / Kupper, L. L. / Chambless, L. E. (1982): Logistic Regression Analysis of Epidemiologic Data: Theory and Practice, in: Communications in Statistics, 11 (5), S. 485-547.

Knauff, D. (1991): Testverfahren im Direct Marketing, in: Handbuch Direct Marketing, Dallmer, H. (Hrsg.), 7. Aufl., Gabler, Wiesbaden, S. 581-590.

Kohavi, R. (1995): Wrappers for performance enhancement and oblivious decision graphs, Dissertation, Department of Computer Science, Stanford University.

Kohonen, T. (1986): Learning Vector Quantization for Pattern Recognition, Technical Report No. TKK-F-A601, Helsinki University of Technology, Finland.

Kohonen, T. (1997): Self-Organizing Maps, Springer, Berlin [u.a.].

Kotler, P. (1989): Marketing-Management, 5. Auflage, Prentice Hall, London [u.a.].

Kotler, P. / Bliemel, F. (1995): Marketing-Management: Analyse, Planung, Umsetzung und Steuerung, Schaeffer-Poeschel, Stuttgart.

Krahl, D. / Windheuser, U. / Zick, F.-K. (1998): Data Mining: Einsatz in der Praxis, Addison-Wesley, Bonn.

Kratzer, K.-P. (1990): Neuronale Netze: Grundlagen und Anwendungen, Hanser, München [u.a.].

Kreutzer, R. T. (1991): Database Marketing – Erfolgsstrategien für die neunziger Jahre, in: Handbuch Direct-Marketing, Dallmer, H. (Hrsg.), 6. Auflage, Gabler Verlag, Wiesbaden 1991, S. 623-642.

Kreutzer, R. T. (1992): Zielgruppen Management mit Kunden-Datenbanken, in: DBW 52, Heft 3, S. 325-340.

Kruse, H. (1991): Programmierung Neuronaler Netze: eine Turbo-Pascal Toolbox, Addison-Wesley, Reading [u.a.].

Kubat, M. / Holte, R. / Matwin, S. (1997): Learning when Negative Examples Abound, Proceedings of the European Conference on Machine Learning, ECML'97, S. 146-153.

Kubat, M. / Matwin, S. (1997): Adressing the Curse of Imbalanced Training Sets: One-Sided Selection, in: Proceedings of the 14th International Conference on Machine Learning, S. 179-186.

Küsters, U. (2001): Data Mining Methoden: Einordnung und Überblick, in: Handbuch Data Mining im Marketing, Hippner, H. / Küsters, U. / Meyer, M. / Wilde, K. (Hrsg.), Vieweg, Braunschweig/Wiesbaden, S. 95-130.

Lewis, D. / Catlett, J. (1994): Heterogeneous Uncertainty Sampling for Supervised Learning, in: Machine Learning: Proceedings of the 11th International Conference on Machine Learning, S. 148-156.

Liehr, T. (2000): Data Matching bei Finanzdienstleistungen: Steigerung des Share of Wallet bei Top-Kunden, in: Handbuch Data Mining im Marketing, Hippner, H. / Küsters, U. / Meyer, M. / Wilde, K. (Hrsg.), Vieweg, Braunschweig/Wiesbaden, S. 725-740.

Ling, C. / Li, C. (1998): Data Mining for Direct Marketing: Problems and Solutions, in: Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining, S. 73-79.

Link, J. / Hildebrand, V. (1993): Database Marketing und CAS, Vahlen, München.

Link, J. / Hildebrand, V. (1997a): Grundlagen des Database Marketing, in: Handbuch Database Marketing, Link, J. / Brändli, D. / Schleuning, Chr. / Kehl, R.E. (Hrsg.), IM Fachverlag, Ettlingen 1997, S. 15-36.

Link, J. / Hildebrand, V. (1997b): Ausgewählte Konzepte der Kundenbewertung im Rahmen des Database Marketing, in: Handbuch Database Marketing, Link, J. / Brändli, D. / Schleuning, Chr. / Kehl, R.E. (Hrsg.), IM Fachverlag, Ettlingen 1997, S. 159-173.

Loh, W.-Y. / Shih, Y.-S. (1997): Split Selection Methods for Classification Trees, Statistica Sinica, Vol. 7, S. 815-840.

Lowe, D. / Webb, A. R. (1990): Exploiting Prior Knowledge in Network Optimization: An Illustration from Medical Prognosis, Network, 1, S. 299-323.

Lu, Y. / Guo, H. / Feldkamp, L. (1998): Robust Neural Learning from unbalanced Data Samples, Proceedings of the International Joint Conference on Neural Networks, IEE IJCNN, Vol. 3, S. 1816-1821.

Lusti, M. (2002): Data Warehousing und Data Mining: eine Einführung in entscheidungsunterstützende Systeme, Springer, Berlin [u.a.].

MacQueen, J. B. (1967): Some Methods for Classification and Analysis of Multivariate Observations, Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, 1, S. 281-297.

Magidson, J. (1988): Improved Statistical Technique for Response Modeling, Journal of direct marketing, Vol. 2, No. 4, S. 6-18.

Magidson, J. (1994): The CHAID Approach to Segmentation Modeling: Chi-squared Automatic Interaction Detection, in: Advanced Methods of Marketing Research, Bagozzi, R.P. (Hrsg.), Oxford, S. 118-159.

Matheus, C. J. / Chan, P. K. / Piatetsky-Shapiro, G. (1993): Systems for KDD, IEEE Transactions on Knowledge and Data Engineering, Vol. 5, No. 6, S. 903-913.

Mayer, C. / Middecke, U. (1997): Konzeption einer Direct-Response-Werbeaktion, in: Handbuch Direct Marketing, 7. Aufl., Dallmer, H. (Hrsg.), Gabler, Wiesbaden, S. 353-364.

McCullagh, W. S. / Pitts, W. (1943): A Logical Calculus of the Ideas Immanent in Nervous Activity, in: Bulletin of Mathematical Biophysics, Vol. 5, S. 115-133.

McFadden, D. (1974): Conditional Logistic Regression of Qualitative Choice Behaviour, in: Frontiers in Econometrics, Zarembka, P. (Hrsg.), Academic Press, New York, S. 105-142.

Meffert, H. (2002): Direct Marketing und marktorientierte Unternehmensführung, in: Handbuch Direct Marketing, Dallmer, H. (Hrsg.), 8. Auflage, Gabler, Wiesbaden, S. 33-56.

Meinig, W. (1992): Direktmarketing (Direct Marketing), in: Vahlens Großes Marketinglexikon, Diller, H. (Hrsg.), Beck/Vahlen, München, S. 205-209.

Milley, A. / Seabolt, J. / Willoiams, J. (1998): Data Mining and the Case for Sampling, SAS White Paper.

Milligan, G. W. (1980): An Examination of the Effect of six Types of Error Perturbation on fifteen Clustering Algorithms, Psychometrika, Vol 45, No. 3, S. 325-342.

Milligan, G. W. / Cooper, M. C. (1983): An Examination of Procedures for Determining the Number of Clusters in a Data Set, College of Administrative Science Working Paper Series 83-51, Columbus, The Ohio State University.

Mingers, J. (1989a): An empirical Comparison of Selection Measures of Decision-Tree Induction, Machine Learning, Vol. 3, S. 319-342.

Mingers, J. (1989b): An empirical Comparison of Pruning Methods for Decision Tree Induction, Machine Learning, Vol. 4, S. 227-243.

Mitchell, T. (1997): Machine Learning, McGraw-Hill, Boston [u.a.].

Morgan, J. A. / Sonquist, J. N. (1963): Problems in the analysis of survey data: and a proposal, Journal of the American Statistical Association, 58, S. 415-434.

Muksch, H. / Holthuis, J. / Reiser, M. (1996): Das Data-Warehouse-Konzept – ein Überblick, in: Wirtschaftsinformatik, 4/1996, S. 421-433.

Murthy, S. (1998): Automatic Construction of Decision Trees from Data: A Multi-Disciplinary Survey, in: Knowledge Discovery and Data Mining, Vol. 2, No. 4, S. 345-389.

Murthy, S. / Salzberg, S. (1995a): Lookahead and Pathology in Decision Tree Induction, Proceedings of International Joint Conference on Artificial Intelligence, JCAI-95, S. 1025-1031.

Murthy, S. / Salzberg, S. (1995b): Decision Tree Induction: How Effective is the Greedy Heuristic ?, in: Proceedings of the First International Conference on Knowledge Discovery and Data Mining, S. 222-227.

Musiol, G. (1996): Adressselektionen bei Mail-Order-Aktionen mittels kategorieller Regression, Beiträge des Instituts für empirische Wirtschaftsforschung, Universität Osnabrück.

Musiol. G. (1999): Database Marketing: Optimale Zielgruppenbestimmung mit Hilfe statistischer Verfahren, Dr. Kovac, Hamburg.

Musiol, G. / Steinkamp, G. (1998): CHAID - Ein Instrument für die empirische Marktforschung, in: Computer Based Marketing, Hippner, H. / Meyer, M. / Wilde, K. (Hrsg.), Vieweg, Braunschweig/Wiesbaden, S. 581-590.

Musiol, G. / Steinkamp, G. (2001): Data Mining als Instrument des Fundraising in Nonprofit-Organisationen, in: Handbuch Data Mining im Marketing, Hippner, H. / Küsters, U. / Meyer, M. / Wilde, K. (Hrsg.), Vieweg, Braunschweig/Wiesbaden, S. 741-753.

Nash, E. (2000): Direct Marketing, strategy, planning, execution, 4. Auflage, McGraw-Hill, New York [u.a.].

Nauta, K. / Matkovski, I. (1999): Using Data Mining Techniques for Fraud Detection, SAS White Paper.

Neville, P. G. (1999) : Decision Trees for Predictive Modeling, SAS Institute Inc, Cary.

Neuneier, R. / Tresp, V. (1994): Radiale Basisfunktionen, Dichteschätzungen und Neuro-Fuzzy, in: Neuronale Netzwerke in der Ökonomie, Rehkugler, H. / Zimmermann, H. G. (Hrsg.), Vahlen, München [u.a.], S. 347-371.

Oates, T. / Jensen, D. (1997): The Effects of Training Set Size on Decision Tree Complexity, in: Proceedings of the 14th International Conference on Machine Learning, S. 254-262.

Oberhofer, W. / Zimmerer, T. (1998): Wie Künstliche Neuronale Netz lernen: Ein Blick in die Black-Box der Backpropagation Netzwerke, Regensburger Diskussionsbeiträge zur Wirtschaftswissenschaft Nr. 292, Universität Regensburg.

Ohno-Machado, L. (1996a): Medical Applications of Neural Networks: Connectionist Models of Survival, Dissertation, Stanford University.

Ohno-Machado, L. (1996b): Sequential Use of Neural Networks for Survival Prediction in AIDS, in: Proceedings of the 1996 American Medical Informatics Fall Meeting, S. 170-174.

Opitz, O. (1978): Zielsetzung und Überblick, in: Numerische Taxonomie in der Marktforschung, Opitz, O. (Hrsg.), Vahlen, 1978, S. 1-19.

Opitz, O. (1980): Numerische Taxonomie, Gustav Fischer, Stuttgart.

Opitz, O. / Schwaiger, M. (1998): Zur Interpretation Mehrdimensionaler Skalierungsergebnisse, in: Computer Based Marketing, Hippner, H. / Meyer, M. / Wilde, K. (Hrsg.), Vieweg, Braunschweig Wiesbaden, S. 563-580.

Piatetsky-Shapiro, G. / Masand, B. (1999): Estimating Campaign Benefits and Modeling Lift, in: Chaudhuri, S. / Madigan, D., Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, S. 185-193.

Poddig, T. (1992): Künstliche Intelligenz und Entscheidungstheorie, Dt.-Univ. Verlag, Wiesbaden.

Poddig, T. / Sidorovitch, I. (2001): Künstliche Neuronale Netze: Überblick, Einsatzmöglichkeiten und Anwendungsprobleme, in: Handbuch Data Mining im Marketing, Hippner, H. / Küsters, U. / Meyer, M. / Wilde, K. (Hrsg.), Vieweg, Braunschweig/Wiesbaden, S. 363-402.

Potts, W. (1999): Decision Tree Modeling Course Notes, SAS Institute Inc., Cary.

Potts, W. (2000): Neural Networks Modeling Course Notes, SAS Institute Inc., Cary.

Provost, F. / Oates, T. / Jensen, D. (1999): Efficient Progressive Sampling, in: Proceedings of the Fifth International Conference on Knowledge Discovery and Data Mining, S. 23-32.

Quinlan, J. R. (1986): Induction of Decision Trees, Machine Learning, 1, S. 81-106.

Quinlan, J. R. (1993): C4.5: Programs for Machine Learning, Morgan Kaufman Publishers, San Francisco.

Quinlan, J. R. (1996): Bagging, boosting, and C4.5, in: Proceedings of the 13th National Conference on Artificial Intelligence, S. 725-730.

Ralambondrainy, H. (1995): A conceptual version of the K-means algorithm, Pattern Recognition, Vol. 16, S. 1147-1157.

Rapp, R. (2002): Die Rolle des Direct Marketing im CRM, in: Handbuch Direct Marketing, Dallmer, H. (Hrsg.), 8. Auflage, Gabler, Wiesbaden, S. 73-86.

Reinartz, T. (1999): Focusing Solutions for Data Mining, Springer, Berlin [u.a.].

Ripley, B. D. (1996): Pattern Recognition and Neural Networks, University Press, Cambridge.

Rojas, R. (1996): Theorie der Neuronalen Netze – Eine systematische Einführung, Springer, Berlin [u.a.].

Rosenblatt, F. (1958): The Percepron: A Probabilistic Model for Information Storage and Organization in the Brain, Psychological Review, Vol. 65, S. 386-408.

Rumelhart, D. E. / Hinton, G. E. / Williams, R. J. (1986): Learning Internal Representations by Error Propagation, in: Parallel Distributed Processing: Explanations in the Microstructure of Cognition, Rumelhart, D. E. / McClelland, J. L. (Hrsg.), Vol. 1, Foundations, MIT Press, Cambridge, S. 318-362.

Säuberlich, F. (2000a): KDD und Data Mining als Hilfsmittel zur Entscheidungsunterstützung, Peter Lang, Frankfurt.

Säuberlich, F. (2000b): Web Mining – Effektives Marketing im Internet, in: Neuronale Netze im Marketing-Management, Wiedemann, K.-P. / Buckler, F. (Hrsg.), Gabler, Wiesbaden, S. 103-121.

Sarle, W. S. (1983): Cubic Clustering Criterion, SAS Technical Report A-108, SAS Institute Inc., Cary.

Sarle, W. S. (1994) : Neural Networks and Statistical Models, Proceedings of the 19th Annual SAS Users Group International Conference, SAS Institute Inc., Cary.

Sarle, W. S. (1997) : how to measure the importance of inputs, ftp://ftp.sas.com/pub/neural/importance.html, 01.10.03.

SAS Institute GmbH (2002), o. V.: SEMMA, www.sas.com/products/miner/sample.html, Stand: 24.07.2002.

Schaller, G. (1988): Markterfolge aus der Datenbank: Aufbau, Entwicklung und Pflege leistungsfähiger Marketing-Datenbanken, Verlag Moderne Industrie, Landsberg am Lech 1988.

Schaller, G. (1997): Organisation der Erfolgskontrolle im Direct Marketing, in: Handbuch Direct Marketing, Dallmer, H. (Hrsg.), 7. Aufl., Gabler, Wiesbaden, S. 579-589.

Schinzer, H. D. (1997): Database Marketing, in: Mertens, P. (Hrsg.): Lexikon der Wirtschafsinformatik, 3. Auflage, Springer, Berlin [u.a.], S. 106-108.

Shannon, C. E. (1948): The Mathematical Theory of Communications, The Bell Systems Technical Journal, 27, S. 379-423.

Shaw, R./Stone, M. (1988): Database Marketing, Gower Publishing Company, Southampton.

Shtatland, E. S. / Cain, E. / Barton, M. B. (2001): The Perils of Stepwise Logistic Regression and How to Escape them using information criteria and the output delivery system, Paper 222-26, SAS SUGI 26.

Siegert, W. (1974): Wesen und Bedeutung der Absatzwerbung im Versandhandel, Dissertation, Universität Berlin.

Sodeur, W. (1974): Empirische Verfahren zur Klassifikation, Teubner, Stuttgart.

Specht, D. F. (1990): Probabilistic Neural Networks, in: Neural Networks, Vol. 3, S. 109-118.

Steinbuch, K. (1971): Automat und Mensch, Springer, Berlin [u.a.].

Stone, B. (1989): Successful Directmarketing Methods, 4th ed., NTC Business Books, Lincolnwood.

Temme, T. / Decker, R. (1999): CHAID als Instrument des Data Mining in der Marktforschung, Diskussionspapier Nr. 439, Universität Bielefeld, Fakultät für Wirtschaftswissenschaften.

Tou, J. T. / Gonzalez, R. C. (1974): Pattern Reognition Principle, Addison-Wesley, Reading.

Tukey, J. W. (1977): Exploratory Data Analysis, Addison-Wesley, Reading.

Turney, P. (1995): Technical Note: Bias and Quantification of Stability, Machine Learning, 20, S. 23-33.

Urban, A. (1998): Einsatz künstlicher Neuronaler Netze bei der operativen Werbemitteleinsatzplanung im Versandhandel im Vergleich zu ökonometrischen Verfahren, Dissertation, dissertation.de.

Weingärtner, S. (2001): Web Mining – Ein Erfahrungsbericht, in: Handbuch Data Mining im Marketing, Hippner, H. / Küsters, U. / Meyer, M. / Wilde, K. (Hrsg.), Vieweg, Braunschweig/Wiesbaden, S. 889-903.

Weiss, G. / Hirsh, H. (2000): Learning to Predict Extremely Rare Events, Fourth International Conference on Knowledge Discovery and Data Mining, AAAI Press, Menlo Park, S. 359-363.

Weiss, G. / Provost, F. (2001): The Effect of Class Distribution on Classifier Learning, Technical Report ML-TR-43, Department of Computer Science, Rutgers University.

Weiss, S. H. / Indurkhya, N. (1998): Predictive Data Mining – A Practical Guide, Morgan Kaufman, San Francisco.

Weiss, S. H. / Kulikowski, C. (1991): Computer systems that learn: classification and prediction methods from statistics, neural nets, machine learning, and expert systems, Morgan Kaufman, San Mateo.

Werbos, P. (1974): Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences, PhD Thesis, Harvard University, Cambridge.

Wiedemann, K.-P. / Buckler, F. (2001): Neuronale Netze im Management, in: Neuronale Netze im Marketing-Management: Praxisorientierte Einführung in modernes Data Mining, Wiedemann, K.-P. / Buckler, F. (Hrsg.), Gabler, Wiesbaden, S. 35-100.

Wilde, K. D. (2001): Data Warehouse, OLAP und Data Mining im Marketing – Moderne Informationstechnologien im Zusammenspiel, in: Handbuch Data Mining im Marketing, Hippner, H. / Küsters, U. / Meyer, M. / Wilde, K. (Hrsg.), Vieweg, Braunschweig/Wiesbaden, S. 1-19.

Witten, I. / Frank, E. (2001): Data Mining: praktische Werkzeuge und Techniken für das maschinelle Lernen, Hanser, München [u.a.].

Wittmann, T. / Ruhland, J. (2001): Neuro-Fuzzy Data Mining zur Zielgruppen-selektion im Bankenbereich, in: Handbuch Data Mining im Marketing, Hippner, H. / Küsters, U. / Meyer, M. / Wilde, K. (Hrsg.), Vieweg, Braunschweig/Wiesbaden, S. 787-804.

Wördenweber, M. (1985): Clusteranalysen bei gemischtskalierten Datensätzen, Lit Verlag, Münster.

Zadrozny, B. / Elkan, C. (2001): Obtaining calibrated probability estimates from decision trees and naive Bayesian classifiers, in: Proceedings of the Eighteenth International Conference on Machine Learning, S. 204-213.

Zell, A. (1994): Simulation Neuronaler Netze, Addison-Wesley, Bonn [u.a.].

Anhang

A Umkodierungen

Rechtsf (Nr. 9)		
	0-Klasse	1-Klasse
AG	3542	20
GmbH	46968	213
GmbH & Co	1299	11
GmbH & Co KG	9107	51
KG	2453	13
Sonstiges	454	1
U	27839	83
Summe	91680	392

MA_Zahl (Nr. 8)		
	0-Klasse	1-Klasse
01-09	4651	12
10-25	3372	17
26-49	4602	11
50-99	7960	64
100-499	7543	49
>500	5271	32
U	58281	207
Summe	91680	392

Bula (Nr. 7)		
	0-Klasse	1-Klasse
BW	14265	81
BY	13856	55
HE	8223	34
NS	8190	31
NW	20951	88
RP	3298	22
S	4621	15
SA	2047	12
SH	3985	20
Sonstiges	10338	23
TH	1906	11
Summe	91680	392

Branche2 (Nr. 5)		
	0-Klasse	1-Klasse
DJ	3897	31
DK	4357	29
DL	4328	24
FA	10052	25
GA	8162	41
IA	4644	20
KA	12380	73
NA	3692	20
Sonstiges	13661	60
U	26507	69
Summe	91680	392

Br_k (Nr. 6)		
	0-Klasse	1-Klasse
D	22265	136
F	10063	25
G	8163	41
I	4644	20
K	12380	73
N	3692	20
Sonstiges	3966	8
U	26507	69
Summe	91680	392

Legende:

Bundesländer (Bula, Nr. 7):

BW: Baden-Württemberg
BY: Bayern
HE: Hessen
NS: Niedersachsen
NW: Nordrhein-Westfalen
RP: Rheinland-Pfalz
S: Sachsen
SH: Schleswig-Holstein
Sonstiges: restliche Bundesländer
TH: Thüringen

Branche2 (Nr. 5):

DJ: Metallerzeugung und -bearbeitung, Herstellung von Metallerzeugnissen

DK: Maschinenbau

DL: Herstellung von Büromaschinen, Datenverarbeitungsgeräten und -einrichtungen; Elektrotechnik, Feinmechanik und Optik

FA: Baugewerbe

GA: Handel; Instandhaltung und Reparatur von Kraftfahrzeugen und Gebrauchsgütern

IA: Verkehr und Nachrichtenübermittlung

KA: Grundstücks- und Wohnungswesen, Vermietung beweglicher Sachen, Erbringung von Dienstleistungen überwiegend für Unternehmen

NA: Gesundheits-, Veterinär- und Sozialwesen

U: unbekannt

Sonstiges: restliche Branchen

Br_k (Nr. 6):

D: Verarbeitendes Gewerbe

F: Baugewerbe

G: Handel; Instandhaltung und Reparatur von Kraftfahrzeugen und Gebrauchsgütern

I: Verkehr und Nachrichtenübermittlung

K: Grundstücks- und Wohnungswesen, Vermietung beweglicher Sachen, Erbringung von Dienstleistungen überwiegend für Unternehmen

N: Gesundheits-, Veterinär- und Sozialwesen

U: unbekannt

Sonstiges: restliche Branchen

B Korrelation mit der Zielvariablen

nominale Variablen:

	branche2 (Nr. 5)	br_k (Nr. 6)	bula (Nr. 7)	ma_zahl (Nr. 8)	rechtsf (Nr. 9)
best_jn	54,1856	56,6776	25,7395	53,2775	23,2115
	<.0001	<.0001	0,0041	<.0001	0,0007
	bes_l6m (Nr. 14)	bes_l12m (Nr. 15)	pr_4048 (Nr. 34)	pr_5594 (Nr. 35)	pr_8048 (Nr. 36)
best_jn	15,6223	14,5152	2,5243	1,3386	0,5593
	0,0001	<.0001	0,283	0,5121	0,756
	pr_8091 (Nr. 37)	pr_9588 (Nr. 38)	pr_9664 (Nr. 39)	um_l6m (Nr. 49)	ums_l12m (Nr. 50)
best_jn	2,8776	3,9361	0,6152	14,7444	24,3453
	0,2372	0,1397	0,7352	<.0001	<.0001

Legende	Variablenname
Zielvariable	Wert der Teststatistik
	p-Wert

metrische Variablen:

	anl_date (Nr. 3)	Abostat (Nr. 10)	Aktv_0 (Nr. 11)	anz_aabo (Nr. 12)	Anz_al (Nr. 13)	best_all (Nr. 16)	Best_gw (Nr. 17)	best_j (Nr. 18)
best_jn	0,00205	0,01622	0,02124	0,01102	0,01324	0,00339	0,00952	0,00593
	0,5348	< 0,0001	< 0,0001	0,0008	< 0,0001	0,3042	0,0039	0,0722
	best_j1 (Nr. 19)	best_j2 (Nr. 20)	Best_oth (Nr. 21)	di_ebest (Nr. 22)	Di_lbest (Nr. 23)	di_lpos (Nr. 24)	Kaufstat (Nr. 25)	Kdtyp_0 (Nr. 26)
best_jn	0,01625	0,01001	0,00223	0,2612	-0,02044	-0,02017	0,01619	0,02221
	< 0,0001	0,0024	0,4992	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001
	Nums_al (Nr. 27)	Nums_all (Nr. 28)	nums_gw (Nr. 29)	Nums_j (Nr. 30)	nums_j_1 (Nr. 31)	nums_j_2 (Nr. 32)	nums_oth (Nr. 33)	rem_gw (Nr. 41)
best_jn	0,00734	0,00922	0,01273	0,01363	0,01406	0,01448	0,00866	-0,00023
	0,026	0,0051	0,0001	< 0,0001	< 0,0001	< 0,0001	0,0086	0,9444
	ums_al (Nr. 43)	ums_all (Nr. 44)	ums_gw (Nr. 45)	ums_j (Nr. 46)	ums_j_1 (Nr. 47)	ums_j_2 (Nr. 48)	ums_oth (Nr. 51)	ums_wa (Nr. 52)
best_jn	0,00771	0,00896	0,00812	0,01729	0,01661	0,01291	0,00707	0,00265
	0,0192	0,0065	0,0137	< 0,0001	< 0,0001	< 0,0001	0,0319	0,4125
	anz_7409 (Nr. 53)	anz_dm (Nr. 54)	anz_tm (Nr. 55)	anz_wa (Nr. 56)	anz_wam6 (Nr. 57)	di_e_wa (Nr. 58)	di_l_wa (Nr. 59)	wa_anl (Nr. 60)
best_jn	-0,02743	0,00098	-0,00156	0,00092	-0,00006	-0,00975	0,0067	0,00065
	< 0,0001	0,7651	0,6365	0,7811	0,9845	0,0031	0,0422	0,8433

Legende	Variablenname
Zielvariable	R^2
	p-Wert

C Korrelationen der verbleibenden Variablen untereinander

	di_e_wa (Nr. 58)	di_l_wa (Nr. 59)	anz_wam6 (Nr. 57)	anz_7409 (Nr. 53)	anz_aabo (Nr. 12)	anz_al (Nr. 13)	ums_oth (Nr. 51)	ums_j (Nr. 46)	ums_j_1 (Nr. 47)	ums_j_2 (Nr. 48)	ums_gw (Nr. 45)
di_e_wa (Nr. 58)	100.000	0.08471 <.0001	-0.04999 <.0001	0.13190 <.0001	-0.03242 <.0001	-0.20926 <.0001	-0.10937 <.0001	-0.12064 <.0001	-0.14480 <.0001	-0.18558 <.0001	-0.08272 <.0001
di_l_wa (Nr. 59)	0.08471 <.0001	100.000	-0.34062 <.0001	-0.30464 <.0001	-0.04464 <.0001	-0.08064 <.0001	-0.05907 <.0001	-0.08108 <.0001	-0.08533 <.0001	-0.09760 <.0001	-0.14198 <.0001
anz_wam6 (Nr. 57)	-0.04999 <.0001	-0.34062 <.0001	100.000	0.68067 <.0001	0.24991 <.0001	0.37288 <.0001	0.27471 <.0001	0.34124 <.0001	0.39145 <.0001	0.43895 <.0001	0.56391 <.0001
anz_7409 (Nr. 53)	0.13190 <.0001	-0.30464 <.0001	0.68067 <.0001	100.000	0.16608 <.0001	0.21424 <.0001	0.17351 <.0001	0.18147 <.0001	0.19934 <.0001	0.26707 <.0001	0.46471 <.0001
anz_aabo (Nr. 12)	-0.03242 <.0001	-0.04464 <.0001	0.24991 <.0001	0.16608 <.0001	100.000	0.23472 <.0001	0.14494 <.0001	0.35002 <.0001	0.32784 <.0001	0.29327 <.0001	0.53507 <.0001
anz_al (Nr. 13)	-0.20926 <.0001	-0.08064 <.0001	0.37288 <.0001	0.21424 <.0001	0.23472 <.0001	100.000	0.38058 <.0001	0.60328 <.0001	0.71001 <.0001	0.79736 <.0001	0.40698 <.0001
ums_oth (Nr. 51)	-0.10937 <.0001	-0.05907 <.0001	0.27471 <.0001	0.17351 <.0001	0.14494 <.0001	0.38058 <.0001	100.000	0.32805 <.0001	0.60525 <.0001	0.47811 <.0001	0.30792 <.0001
ums_j (Nr. 46)	-0.12064 <.0001	-0.08108 <.0001	0.34124 <.0001	0.18147 <.0001	0.35002 <.0001	0.60328 <.0001	0.32805 <.0001	100.000	0.58646 <.0001	0.53670 <.0001	0.46254 <.0001
ums_j_1 (Nr. 47)	-0.14480 <.0001	-0.08533 <.0001	0.39145 <.0001	0.19934 <.0001	0.32784 <.0001	0.71001 <.0001	0.60525 <.0001	0.58646 <.0001	100.000	0.68937 <.0001	0.52057 <.0001
ums_j_2 (Nr. 48)	-0.18558 <.0001	-0.09760 <.0001	0.43895 <.0001	0.26707 <.0001	0.29327 <.0001	0.79736 <.0001	0.47811 <.0001	0.53670 <.0001	0.68937 <.0001	100.000	0.62340 <.0001
ums_gw (Nr. 45)	-0.08272 <.0001	-0.14198 <.0001	0.56391 <.0001	0.46471 <.0001	0.53507 <.0001	0.40698 <.0001	0.30792 <.0001	0.46254 <.0001	0.52057 <.0001	0.62340 <.0001	100.000
ums_al (Nr. 43)	-0.24756 <.0001	-0.09291 <.0001	0.44213 <.0001	0.27479 <.0001	0.25324 <.0001	0.87322 <.0001	0.38938 <.0001	0.54495 <.0001	0.67220 <.0001	0.83597 <.0001	0.51936 <.0001
ums_all (Nr. 44)	-0.23566 <.0001	-0.10883 <.0001	0.49991 <.0001	0.32827 <.0001	0.31515 <.0001	0.84721 <.0001	0.57672 <.0001	0.57935 <.0001	0.75073 <.0001	0.87048 <.0001	0.63418 <.0001
nums_gw (Nr. 29)	-0.08828 <.0001	-0.10111 <.0001	0.45149 <.0001	0.32513 <.0001	0.69228 <.0001	0.39218 <.0001	0.29111 <.0001	0.45024 <.0001	0.51536 <.0001	0.59783 <.0001	0.83567 <.0001
nums_all (Nr. 28)	-0.24197 <.0001	-0.09888 <.0001	0.47002 <.0001	0.29352 <.0001	0.31526 <.0001	0.86078 <.0001	0.53093 <.0001	0.57652 <.0001	0.75134 <.0001	0.85074 <.0001	0.57398 <.0001
nums_al (Nr. 27)	-0.24800 <.0001	-0.09082 <.0001	0.43475 <.0001	0.26716 <.0001	0.25168 <.0001	0.87597 <.0001	0.38680 <.0001	0.54431 <.0001	0.66998 <.0001	0.83115 <.0001	0.50461 <.0001
nums_oth (Nr. 33)	-0.11873 <.0001	-0.05944 <.0001	0.28476 <.0001	0.17379 <.0001	0.15819 <.0001	0.40834 <.0001	0.87903 <.0001	0.35617 <.0001	0.66877 <.0001	0.44139 <.0001	0.31169 <.0001
nums_j (Nr. 30)	-0.11968 <.0001	-0.07208 <.0001	0.30619 <.0001	0.16124 <.0001	0.34657 <.0001	0.59836 <.0001	0.31338 <.0001	0.93576 <.0001	0.54533 <.0001	0.51308 <.0001	0.38476 <.0001
nums_j_1 (Nr. 31)	-0.14204 <.0001	-0.07424 <.0001	0.35797 <.0001	0.17856 <.0001	0.32622 <.0001	0.70586 <.0001	0.60403 <.0001	0.57662 <.0001	0.98355 <.0001	0.66499 <.0001	0.45356 <.0001
nums_j_2 (Nr. 32)	-0.19155 <.0001	-0.08712 <.0001	0.41325 <.0001	0.24170 <.0001	0.30227 <.0001	0.84102 <.0001	0.42363 <.0001	0.55386 <.0001	0.72017 <.0001	0.94372 <.0001	0.55253 <.0001
ums_wa (Nr. 52)	-0.15356 <.0001	0.03308 <.0001	0.09860 <.0001	0.02906 <.0001	0.21049 <.0001	0.50725 <.0001	0.34396 <.0001	0.39613 <.0001	0.46570 <.0001	0.54281 <.0001	0.30616 <.0001
best_j (Nr. 18)	-0.01089 0.0009	-0.05539 <.0001	0.17912 <.0001	0.12840 <.0001	0.24359 <.0001	0.10363 <.0001	0.08739 <.0001	0.53446 <.0001	0.14390 <.0001	0.13502 <.0001	0.34620 <.0001
best_j1 (Nr. 19)	-0.03183 <.0001	-0.08664 <.0001	0.30968 <.0001	0.16832 <.0001	0.28891 <.0001	0.19563 <.0001	0.16981 <.0001	0.30060 <.0001	0.44892 <.0001	0.24793 <.0001	0.55327 <.0001
best_j2 (Nr. 20)	-0.06745 <.0001	-0.08563 <.0001	0.34340 <.0001	0.20335 <.0001	0.27006 <.0001	0.27401 <.0001	0.20450 <.0001	0.29209 <.0001	0.35592 <.0001	0.57052 <.0001	0.65734 <.0001
best_gw (Nr. 17)	-0.09370 <.0001	-0.13822 <.0001	0.55903 <.0001	0.43986 <.0001	0.57571 <.0001	0.41587 <.0001	0.31221 <.0001	0.46814 <.0001	0.52699 <.0001	0.62736 <.0001	0.98916 <.0001
di_ebest (Nr. 22)	0.02642 <.0001	0.02752 <.0001	-0.03132 <.0001	-0.03298 <.0001	0.02884 <.0001	-0.02164 <.0001	-0.01045 0.0015	-0.02069 <.0001	-0.02285 <.0001	-0.02840 <.0001	-0.02431 <.0001
di_lpos (Nr. 24)	-0.02790 <.0001	0.04193 <.0001	-0.31752 <.0001	-0.27639 <.0001	-0.35183 <.0001	-0.29886 <.0001	-0.26797 <.0001	-0.31665 <.0001	-0.33325 <.0001	-0.40062 <.0001	-0.51411 <.0001
di_lbest (Nr. 23)	-0.00372 0.2592	0.07195 <.0001	-0.34630 <.0001	-0.30497 <.0001	-0.37710 <.0001	-0.31540 <.0001	-0.28157 <.0001	-0.33922 <.0001	-0.35518 <.0001	-0.42098 <.0001	-0.55117 <.0001
aktiv_0 (Nr. 11)	-0.03099 <.0001	-0.04997 <.0001	0.34875 <.0001	0.25908 <.0001	0.39455 <.0001	0.41924 <.0001	0.33660 <.0001	0.45153 <.0001	0.46066 <.0001	0.48785 <.0001	0.53141 <.0001
kdtyp_0 (Nr. 26)	-0.02254 <.0001	-0.04519 <.0001	0.33229 <.0001	0.25822 <.0001	0.38943 <.0001	0.37471 <.0001	0.32117 <.0001	0.38753 <.0001	0.41556 <.0001	0.45821 <.0001	0.52170 <.0001
kaufstat (Nr. 25)	-0.05025 <.0001	-0.08416 <.0001	0.38381 <.0001	0.30355 <.0001	0.44287 <.0001	0.39780 <.0001	0.37904 <.0001	0.42899 <.0001	0.43419 <.0001	0.46399 <.0001	0.54178 <.0001
abostat (Nr. 10)	-0.03087 <.0001	-0.04734 <.0001	0.29405 <.0001	0.19846 <.0001	0.93718 <.0001	0.30098 <.0001	0.18034 <.0001	0.38615 <.0001	0.36392 <.0001	0.38038 <.0001	0.62062 <.0001

	ums_al (Nr. 43)	ums_all (Nr. 44)	nums_gw (Nr. 29)	nums_all (Nr. 28)	nums_al (Nr. 27)	nums_oth (Nr. 33)	nums_j (Nr. 30)	nums_j_1 (Nr. 31)	nums_j_2 (Nr. 32)	ums_wa (Nr. 52)	best_j (Nr. 18)
di_e_wa (Nr. 58)	-0.24756 <.0001	-0.23566 <.0001	-0.08828 <.0001	-0.24197 <.0001	-0.24800 <.0001	-0.11873 <.0001	-0.11968 <.0001	-0.14204 <.0001	-0.19155 <.0001	-0.15356 <.0001	-0.01089 0.0009
di_l_wa (Nr. 59)	-0.09291 <.0001	-0.10883 <.0001	-0.10111 <.0001	-0.09888 <.0001	-0.09082 <.0001	-0.05944 <.0001	-0.07208 <.0001	-0.07424 <.0001	-0.08712 <.0001	0.03308 <.0001	-0.05539 <.0001
anz_wam6 (Nr. 57)	0.44213 <.0001	0.49991 <.0001	0.45149 <.0001	0.47002 <.0001	0.43475 <.0001	0.28476 <.0001	0.30619 <.0001	0.35797 <.0001	0.41325 <.0001	0.09860 <.0001	0.17912 <.0001
anz_7409 (Nr. 53)	0.27479 <.0001	0.32827 <.0001	0.32513 <.0001	0.29352 <.0001	0.26716 <.0001	0.17379 <.0001	0.16124 <.0001	0.17856 <.0001	0.24170 <.0001	0.02906 <.0001	0.12840 <.0001
anz_aabo (Nr. 12)	0.25324 <.0001	0.31515 <.0001	0.69228 <.0001	0.31526 <.0001	0.25168 <.0001	0.15819 <.0001	0.34657 <.0001	0.32622 <.0001	0.30227 <.0001	0.21049 <.0001	0.24359 <.0001
anz_al (Nr. 13)	0.87322 <.0001	0.84721 <.0001	0.39218 <.0001	0.86078 <.0001	0.87597 <.0001	0.40834 <.0001	0.59836 <.0001	0.70586 <.0001	0.84102 <.0001	0.50725 <.0001	0.10363 <.0001
ums_oth (Nr. 51)	0.38938 <.0001	0.57672 <.0001	0.29111 <.0001	0.53093 <.0001	0.38680 <.0001	0.87903 <.0001	0.31338 <.0001	0.60403 <.0001	0.42363 <.0001	0.34396 <.0001	0.08739 <.0001
ums_j (Nr. 46)	0.54495 <.0001	0.57935 <.0001	0.45024 <.0001	0.57652 <.0001	0.54431 <.0001	0.35617 <.0001	0.93576 <.0001	0.57662 <.0001	0.55386 <.0001	0.39613 <.0001	0.53446 <.0001
ums_j_1 (Nr. 47)	0.67220 <.0001	0.75073 <.0001	0.51536 <.0001	0.75134 <.0001	0.66998 <.0001	0.66877 <.0001	0.54533 <.0001	0.98355 <.0001	0.72017 <.0001	0.46570 <.0001	0.14390 <.0001
ums_j_2 (Nr. 48)	0.83597 <.0001	0.87048 <.0001	0.59783 <.0001	0.85074 <.0001	0.83115 <.0001	0.44139 <.0001	0.51308 <.0001	0.66499 <.0001	0.94372 <.0001	0.54281 <.0001	0.13502 <.0001
ums_gw (Nr. 45)	0.51936 <.0001	0.63418 <.0001	0.83567 <.0001	0.57398 <.0001	0.50461 <.0001	0.31169 <.0001	0.38476 <.0001	0.45356 <.0001	0.55253 <.0001	0.30616 <.0001	0.34620 <.0001
ums_al (Nr. 43)	100.000	0.96806 <.0001	0.51556 <.0001	0.97861 <.0001	0.99918 <.0001	0.41395 <.0001	0.52750 <.0001	0.65716 <.0001	0.86674 <.0001	0.57402 <.0001	0.11840 <.0001
ums_all (Nr. 44)	0.96806 <.0001	100.000	0.60305 <.0001	0.98963 <.0001	0.96465 <.0001	0.57084 <.0001	0.55054 <.0001	0.72838 <.0001	0.87295 <.0001	0.58301 <.0001	0.16587 <.0001
nums_gw (Nr. 29)	0.51556 <.0001	0.60305 <.0001	100.000	0.58661 <.0001	0.50181 <.0001	0.30177 <.0001	0.40854 <.0001	0.48199 <.0001	0.59573 <.0001	0.34170 <.0001	0.30651 <.0001
nums_all (Nr. 28)	0.97861 <.0001	0.98963 <.0001	0.58661 <.0001	100.000	0.97746 <.0001	0.57715 <.0001	0.55785 <.0001	0.73882 <.0001	0.88172 <.0001	0.58369 <.0001	0.14601 <.0001
nums_al (Nr. 27)	0.99918 <.0001	0.96465 <.0001	0.50181 <.0001	0.97746 <.0001	100.000	0.41183 <.0001	0.52984 <.0001	0.65754 <.0001	0.86521 <.0001	0.57333 <.0001	0.11491 <.0001
nums_oth (Nr. 33)	0.41395 <.0001	0.57084 <.0001	0.30177 <.0001	0.57715 <.0001	0.41183 <.0001	100.000	0.34535 <.0001	0.67597 <.0001	0.45372 <.0001	0.32538 <.0001	0.08763 <.0001
nums_j (Nr. 30)	0.52750 <.0001	0.55054 <.0001	0.40854 <.0001	0.55785 <.0001	0.52984 <.0001	0.34535 <.0001	100.000	0.54330 <.0001	0.53365 <.0001	0.37230 <.0001	0.39888 <.0001
nums_j_1 (Nr. 31)	0.65716 <.0001	0.72838 <.0001	0.48199 <.0001	0.73882 <.0001	0.65754 <.0001	0.67597 <.0001	0.54330 <.0001	100.000	0.70093 <.0001	0.45687 <.0001	0.12041 <.0001
nums_j_2 (Nr. 32)	0.86674 <.0001	0.87295 <.0001	0.59573 <.0001	0.88172 <.0001	0.86521 <.0001	0.45372 <.0001	0.53365 <.0001	0.70093 <.0001	100.000	0.54081 <.0001	0.12272 <.0001
ums_wa (Nr. 52)	0.57402 <.0001	0.58301 <.0001	0.34170 <.0001	0.58369 <.0001	0.57333 <.0001	0.32538 <.0001	0.37230 <.0001	0.45687 <.0001	0.54081 <.0001	100.000	0.09770 <.0001
best_j (Nr. 18)	0.11840 <.0001	0.16587 <.0001	0.30651 <.0001	0.14601 <.0001	0.11491 <.0001	0.08763 <.0001	0.39888 <.0001	0.12041 <.0001	0.12272 <.0001	0.09770 <.0001	100.000
best_j1 (Nr. 19)	0.23102 <.0001	0.30533 <.0001	0.46239 <.0001	0.26805 <.0001	0.22240 <.0001	0.16441 <.0001	0.21218 <.0001	0.34594 <.0001	0.22877 <.0001	0.16123 <.0001	0.18423 <.0001
best_j2 (Nr. 20)	0.32898 <.0001	0.40731 <.0001	0.49857 <.0001	0.36146 <.0001	0.31844 <.0001	0.20787 <.0001	0.25517 <.0001	0.32003 <.0001	0.45593 <.0001	0.23311 <.0001	0.13942 <.0001
best_gw (Nr. 17)	0.53126 <.0001	0.64310 <.0001	0.84928 <.0001	0.58661 <.0001	0.51674 <.0001	0.31670 <.0001	0.39078 <.0001	0.46186 <.0001	0.56093 <.0001	0.31605 <.0001	0.34231 <.0001
di_ebest (Nr. 22)	-0.03228 <.0001	-0.03191 <.0001	-0.01243 0.0002	-0.02994 <.0001	-0.03187 <.0001	-0.00888 0.0071	-0.02021 <.0001	-0.02167 <.0001	-0.02536 <.0001	-0.02075 <.0001	-0.01546 <.0001
di_lpos (Nr. 24)	-0.38538 <.0001	-0.44529 <.0001	-0.45182 <.0001	-0.42047 <.0001	-0.37815 <.0001	-0.27561 <.0001	-0.27899 <.0001	-0.29691 <.0001	-0.36451 <.0001	-0.40941 <.0001	-0.17356 <.0001
di_lbest (Nr. 23)	-0.38372 <.0001	-0.45241 <.0001	-0.49230 <.0001	-0.42569 <.0001	-0.37618 <.0001	-0.28921 <.0001	-0.30038 <.0001	-0.31809 <.0001	-0.38517 <.0001	-0.39574 <.0001	-0.18689 <.0001
aktiv_0 (Nr. 11)	0.50673 <.0001	0.56061 <.0001	0.54584 <.0001	0.54923 <.0001	0.50046 <.0001	0.35537 <.0001	0.41415 <.0001	0.42944 <.0001	0.48707 <.0001	0.48362 <.0001	0.23077 <.0001
kdtyp_0 (Nr. 26)	0.47234 <.0001	0.52785 <.0001	0.53502 <.0001	0.51483 <.0001	0.46535 <.0001	0.33819 <.0001	0.35493 <.0001	0.38210 <.0001	0.45219 <.0001	0.46642 <.0001	0.19600 <.0001
kaufstat (Nr. 25)	0.46169 <.0001	0.53391 <.0001	0.53300 <.0001	0.51773 <.0001	0.45601 <.0001	0.39617 <.0001	0.38558 <.0001	0.39886 <.0001	0.45621 <.0001	0.43607 <.0001	0.21878 <.0001
abostat (Nr. 10)	0.36862 <.0001	0.43494 <.0001	0.73118 <.0001	0.43042 <.0001	0.36438 <.0001	0.19471 <.0001	0.37532 <.0001	0.35279 <.0001	0.38731 <.0001	0.29352 <.0001	0.26111 <.0001

	best_j1 (Nr. 19)	best_j2 (Nr. 20)	best_gw (Nr. 17)	di_ebest (Nr. 22)	di_lpos (Nr. 24)	di_lbest (Nr. 23)	aktiv_0 (Nr. 11)	kdtyp_0 (Nr. 26)	kaufstat (Nr. 25)	abostat (Nr. 10)
di_e_wa	-0.03183	-0.06745	-0.09370	0.02642	-0.02790	-0.00372	-0.03099	-0.02254	-0.05025	-0.03087
(Nr. 58)	<.0001	<.0001	<.0001	<.0001	<.0001	0.2592	<.0001	<.0001	<.0001	<.0001
di_l_wa	-0.08664	-0.08563	-0.13822	0.02752	0.04193	0.07195	-0.04997	-0.04519	-0.08416	-0.04734
(Nr. 59)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
anz_wam6	0.30968	0.34340	0.55903	-0.03132	-0.31752	-0.34630	0.34875	0.33229	0.38381	0.29405
(Nr. 57)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
anz_7409	0.16832	0.20335	0.43986	-0.03298	-0.27639	-0.30497	0.25908	0.25822	0.30355	0.19846
(Nr. 53)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
anz_aabo	0.28891	0.27006	0.57571	0.02884	-0.35183	-0.37710	0.39455	0.38943	0.44287	0.93718
(Nr. 12)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
anz_al	0.19563	0.27401	0.41587	-0.02164	-0.29886	-0.31540	0.41924	0.37471	0.39780	0.30098
(Nr. 13)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
ums_oth	0.16981	0.20450	0.31221	-0.01045	-0.26797	-0.28157	0.33660	0.32117	0.37904	0.18034
(Nr. 51)	<.0001	<.0001	<.0001	0.0015	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
ums_j	0.30060	0.29209	0.46814	-0.02069	-0.31665	-0.33922	0.45153	0.38753	0.42899	0.38615
(Nr. 46)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
ums_j_1	0.44892	0.35592	0.52699	-0.02285	-0.33325	-0.35518	0.46066	0.41556	0.43419	0.36392
(Nr. 47)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
ums_j_2	0.24793	0.57052	0.62736	-0.02840	-0.40062	-0.42098	0.48785	0.45821	0.46399	0.38038
(Nr. 48)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
ums_gw	0.55327	0.65734	0.98916	-0.02431	-0.51411	-0.55117	0.53141	0.52170	0.54178	0.62062
(Nr. 45)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
ums_al	0.23102	0.32898	0.53126	-0.03228	-0.38538	-0.38372	0.50673	0.47234	0.46169	0.36862
(Nr. 43)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
ums_all	0.30533	0.40731	0.64310	-0.03191	-0.44529	-0.45241	0.56061	0.52785	0.53391	0.43494
(Nr. 44)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
nums_gw	0.46239	0.49857	0.84928	-0.01243	-0.45182	-0.49230	0.54584	0.53502	0.53300	0.73118
(Nr. 29)	<.0001	<.0001	<.0001	0.0002	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
nums_all	0.26805	0.36146	0.58661	-0.02994	-0.42047	-0.42569	0.54923	0.51483	0.51773	0.43042
(Nr. 28)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
nums_al	0.22240	0.31844	0.51674	-0.03187	-0.37815	-0.37618	0.50046	0.46535	0.45601	0.36438
(Nr. 27)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
nums_oth	0.16441	0.20787	0.31670	-0.00888	-0.27561	-0.28921	0.35537	0.33819	0.39617	0.19471
(Nr. 33)	<.0001	<.0001	<.0001	0.0071	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
nums_j	0.21218	0.25517	0.39078	-0.02021	-0.27899	-0.30038	0.41415	0.35493	0.38558	0.37532
(Nr. 30)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
nums_j_1	0.34594	0.32003	0.46186	-0.02167	-0.29691	-0.31809	0.42944	0.38210	0.39886	0.35279
(Nr. 31)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
nums_j_2	0.22877	0.45593	0.56093	-0.02536	-0.36451	-0.38517	0.48707	0.45219	0.45621	0.38731
(Nr. 32)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
ums_wa	0.16123	0.23311	0.31605	-0.02075	-0.40941	-0.39574	0.48362	0.46642	0.43607	0.29352
(Nr. 52)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
best_j	0.18423	0.13942	0.34231	-0.01546	-0.17356	-0.18689	0.23077	0.19600	0.21878	0.26111
(Nr. 18)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
best_j1	100.000	0.23920	0.56124	-0.01634	-0.28997	-0.31159	0.34817	0.33260	0.34660	0.33007
(Nr. 19)		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
best_j2	0.23920	100.000	0.65855	-0.02073	-0.36109	-0.38737	0.33270	0.32996	0.32158	0.32689
(Nr. 20)	<.0001		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
best_gw	0.56124	0.65855	100.000	-0.02197	-0.51627	-0.55340	0.54013	0.53037	0.55271	0.65197
(Nr. 17)	<.0001	<.0001		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
di_ebest	-0.01634	-0.02073	-0.02197	100.000	-0.09515	-0.04564	0.06458	0.08727	0.03914	0.03968
(Nr. 22)	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
di_lpos	-0.28997	-0.36109	-0.51627	-0.09515	100.000	0.93364	-0.78553	-0.82713	-0.75770	-0.45393
(Nr. 24)	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001	<.0001	<.0001
di_lbest	-0.31159	-0.38737	-0.55340	-0.04564	0.93364	100.000	-0.70596	-0.75031	-0.79137	-0.48101
(Nr. 23)	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001	<.0001
aktiv_0	0.34817	0.33270	0.54013	0.06458	-0.78553	-0.70596	100.000	0.98488	0.81389	0.50496
(Nr. 11)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001
kdtyp_0	0.33260	0.32996	0.53037	0.08727	-0.82713	-0.75031	0.98488	100.000	0.82840	0.50499
(Nr. 26)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001
kaufstat	0.34660	0.32158	0.55271	0.03914	-0.75770	-0.79137	0.81389	0.82840	100.000	0.56010
(Nr. 25)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001
abostat	0.33007	0.32689	0.65197	0.03968	-0.45393	-0.48101	0.50496	0.50499	0.56010	100.000
(Nr. 10)	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	

Legende	Variablenname
Variablen-name	R^2
	p-Wert

D Ausreißeranalyse

Nr	Variable	N	Mean	Minimum	1st Pctl	5th Pctl	50th Pctl
58	di_n_wa	92972	655.4681771	-1.0000000	-1.0000000	-1.0000000	765.0000000
59	di_l_wa	92972	104.0782866	0	0	63.0000000	91.0000000
24	di_lpos	92972	5641.20	60.0000000	61.0000000	74.0000000	9999.00
57	anz_wam6	92972	5.3243223	0	0	0	4.0000000
53	anz_7409	92972	2.1798484	0	0	0	1.0000000
51	ums_oth	92974	188.4157665	0	0	0	0
46	ums_j	92974	25.7669653	0	0	0	0
47	ums_j_1	92974	127.4874358	0	0	0	0
48	ums_j_2	92974	163.6425433	0	0	0	0
45	ums_gw	92974	144.5407753	0	0	0	0
16	best_all	92974	3.6777889	0	0	0	0
18	best_j	92974	0.0204771	0	0	0	0
19	best_j1	92974	0.0579973	0	0	0	0
20	best_j2	92974	0.1531268	0	0	0	0
22	di_mbest	92972	38.1939895	-1.0000000	-1.0000000	-1.0000000	-1.0000000
3	ANR_DATE	92972	13265.36	-21315.00	12216.00	12216.00	13489.00
12	anz_aabo	92972	0.1787623	0	0	0	0

Nr.	Variable	Maximum	95th Pctl	99th Pctl
58	di_n_wa	1222.00	1119.00	1119.00
59	di_l_wa	1856.00	246.0000000	322.0000000
24	di_lpos	9999.00	9999.00	9999.00
57	anz_wam6	288.0000000	15.0000000	24.0000000
53	anz_7409	33.0000000	7.0000000	10.0000000
51	ums_oth	63950.40	588.0000000	1883.10
46	ums_j	3344.93	246.0000000	517.3800000
47	ums_j_1	63475.20	786.5000000	2980.15
48	ums_j_2	12980.00	952.1000000	2310.36
45	ums_gw	4312.00	734.0000000	1536.00
16	best_all	99.0000000	21.0000000	51.0000000
18	best_j	6.0000000	0	1.0000000
19	best_j1	11.0000000	1.0000000	2.0000000
20	best_j2	9.0000000	1.0000000	2.0000000
22	di_mbest	9999.00	-1.0000000	1619.00
3	ANR_DATE	15855.00	14272.00	14272.00
12	anz_aabo	16.0000000	1.0000000	3.0000000

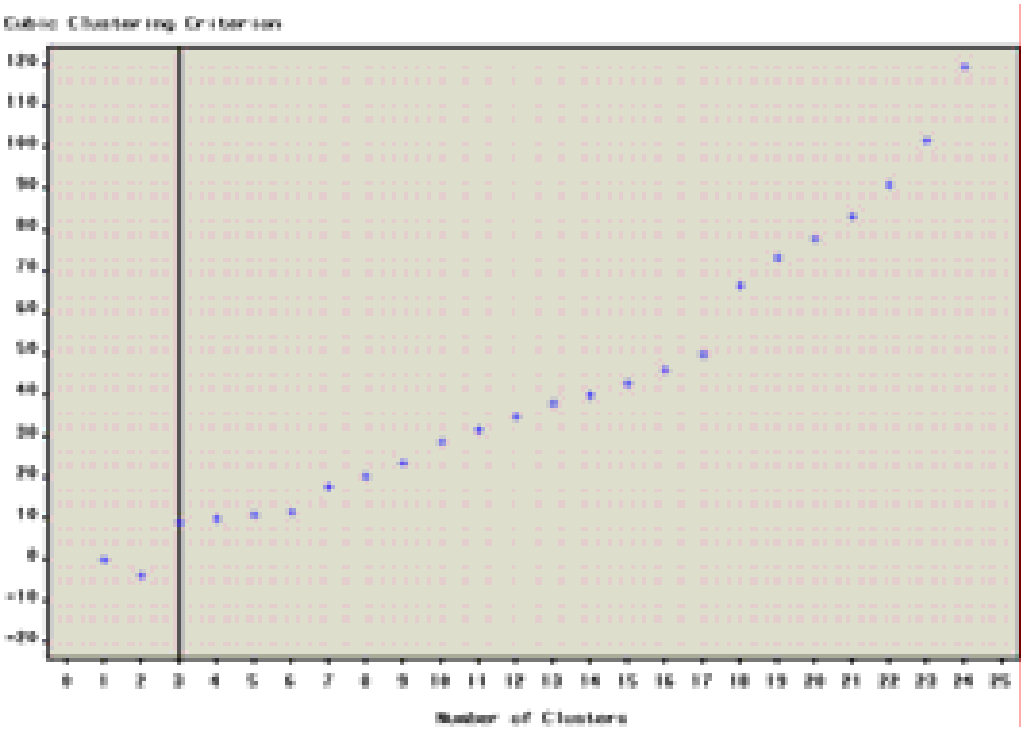
Definition: 1st Pctl: 1% Perzentil

xth Pctl: x% Perzentil

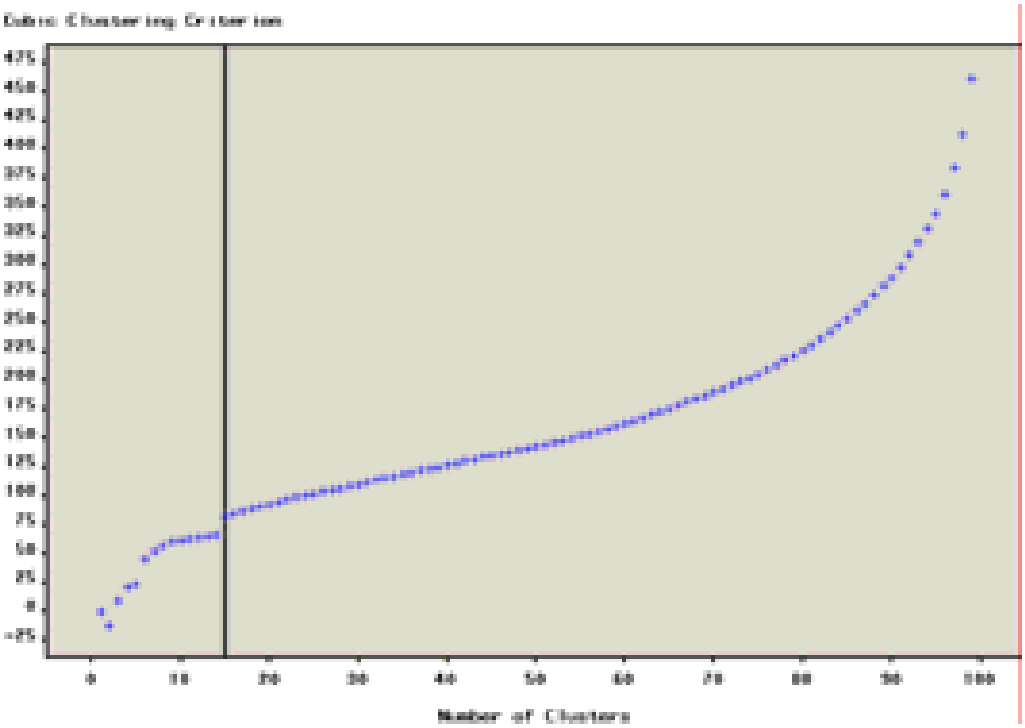
Bei Variable Nr. 59, di_l_wa, besagt das 5th Pctl., dass 5% der Objekte vor 63 oder weniger Tagen die letzte Werbeaktion erhalten haben bzw. dass es bei 95% der Objekte 63 Tage oder länger her ist, seitdem sie ihre letzte Werbeaktion erhalten haben.

E CCC-Plots

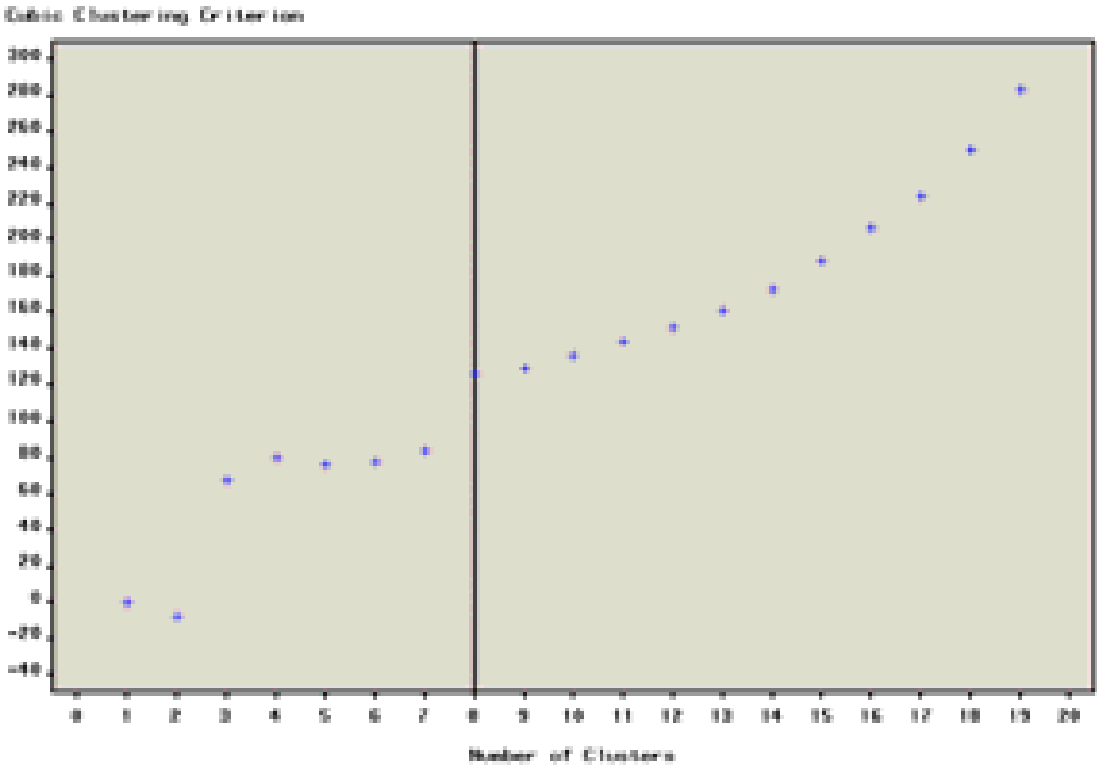
1-Klasse (Fall A):



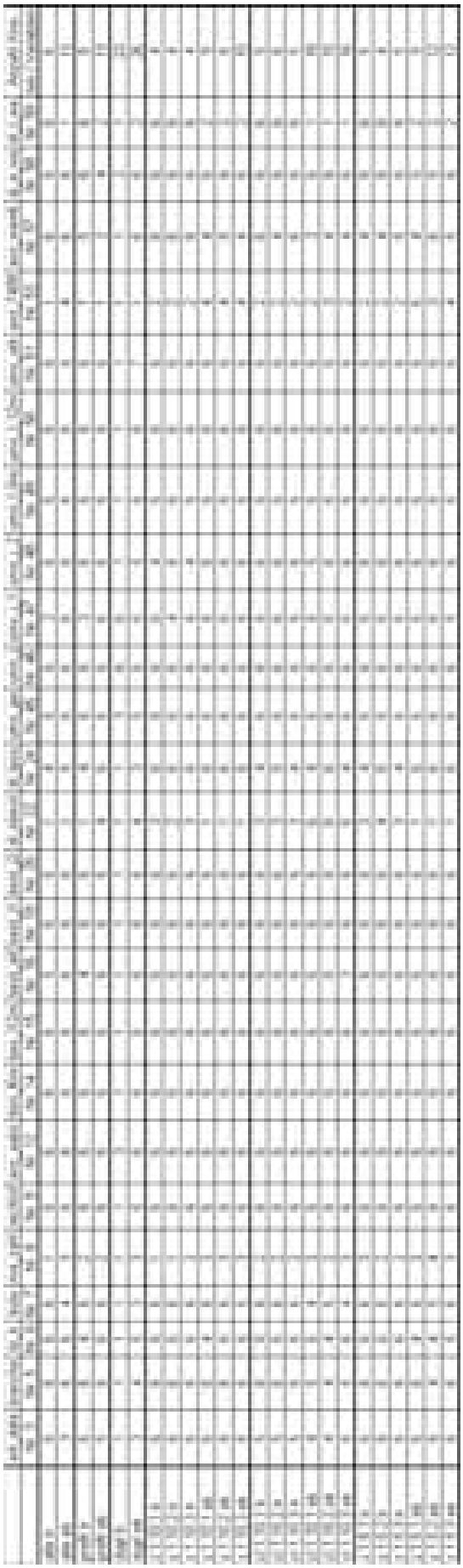
0-Klasse (Fall B)

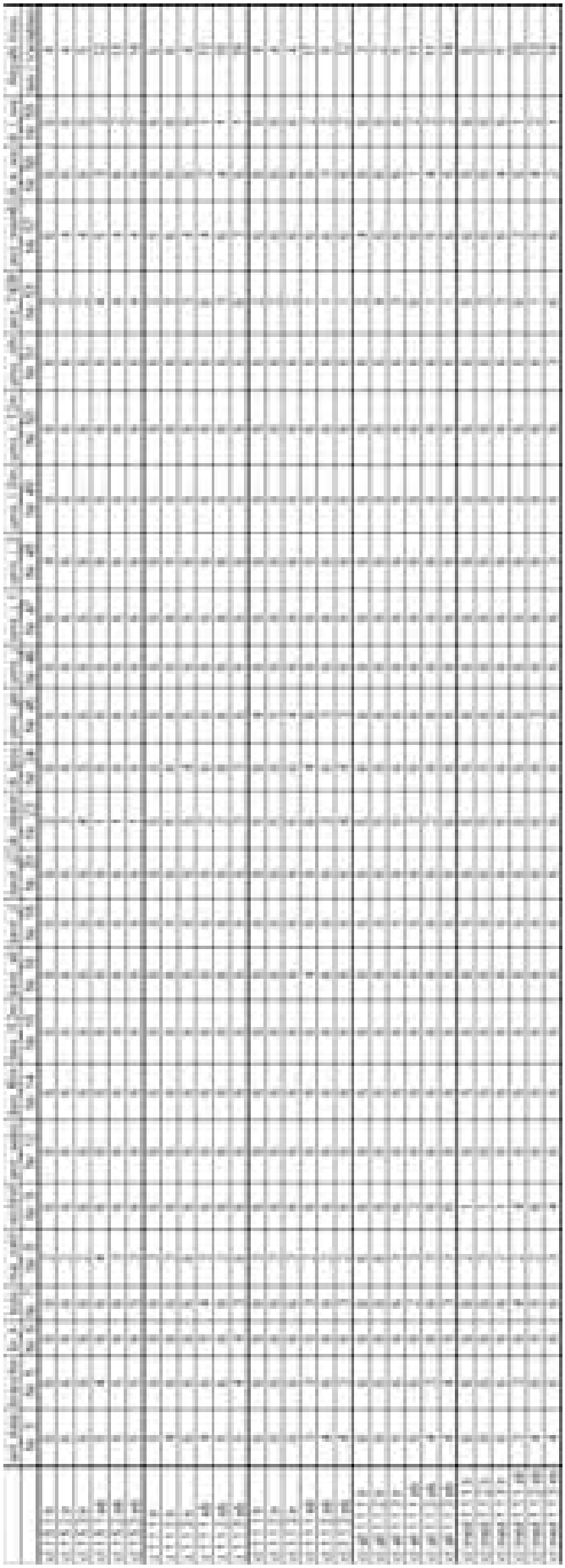


Alle Trainingsdaten (Fall C);



F Datenmatrix für MDS





G Datenmatrix bei 8 Variablen für logistische Regression

	anl_date	ma_zahl	di_ebest	di_lpos	anz_7409	anz_wam6	di_l_wa
	Nr. 3	Nr. 8	Nr. 22	Nr. 24	Nr. 53	Nr. 57	Nr. 59
alle		2	1	4	3		
profit		3	1	4	2	5	
dupl	6	2	3	4	1	5	7
z 1:10 1		2	3		1	4	
z 1:10 2		2	3	5	1	4	
z 1:10 3		2	3	5	1	4	
cl 1:10 1	5	1	3	4	2		
cl 1:10 2	6	2	3	5	1	4	
cl 1:10 3	5	2	3	4	1		
z 1:5 1		2	3	4	1	5	
z 1:5 2		2	3	5	1	4	
z 1:5 3		2	3	4	1	5	
cl 1:5 1	5	1	3	4	2		
cl 1:5 2	5	2	3		1	4	
cl 1:5 3	6	2	3	4	1	5	
z 1:1 1	6	2	4	3	1	5	
z 1:1 2	3	2	4		1	5	
z 1:1 3			4	3	1	2	
cl 1:1 1		2	3	4	1	5	
cl 1:1 2		2	5		1	3	4
cl 1:1 3		2	5	3	1	4	
cl all 1:1 1			3		1	2	
cl all 1:1 2			2		1		
cl all 1:1 3	4	1	3		2		
cl med 1:1	4	1	6	3	2	5	
cl med 1:2		1	4	3	2		
cl med 1:3		1	4	3	2		

H Datenmatrix bei 8 Variablen für Entscheidungsbäume

	anl_date	ma_zahl	di_ebest	di_lpos	anz_7409	anz_wam6	di_l_wa
	Nr. 3	Nr. 8	Nr. 22	Nr. 24	Nr. 53	Nr. 57	Nr. 59
alle	3	3	2	5	4	5	1
profit	5	2	4	5	1	3	2
dupl	3	3	3	3	1	4	2
z 1:10 1	5	3	1		4	4	2
z 1:10 2	5	3	1		4	5	2
z 1:10 3		3	1		4	4	2
cl 1:10 1	4	3		4	2	3	1
cl 1:10 2	4	2			3	4	1
cl 1:10 3	4	3		4	2	4	1
z 1:5 1	4	3	1	5	4	4	2
z 1:5 2	4	4	1	5	3	5	2
z 1:5 3	5	3	1		4	5	2
cl 1:5 1	4	3	1	5	4	5	2
cl 1:5 2	5	3	1		4	4	2
cl 1:5 3	5	3	1		4	4	2
z 1:1 1	4	4	2	4	2	3	1
z 1:1 2	5	2	2		3	5	1
z 1:1 3	4	5	3	3	4	2	1
cl 1:1 1	3	2		4	1	3	2
cl 1:1 2	3	2	3		1	3	2
cl 1:1 3	3	2	3	5	1	4	2
cl all 1:1 1	3	4	3		1	2	2
cl all 1:1 2		3	2	4	1	4	2
cl all 1:1 3	4	3		4	1		2
cl med 1:1	3	2	3	5	5	3	1
cl med 1:2	4	2			1	3	2
cl med 1:3	3	3		4	2	4	1

I Datenmatrix bei 3 Variablen für logistische Regression

	ma_zahl	di_ebest	anz_7409
	Nr. 8	Nr. 22	Nr. 53
alle	2	1	3
profit	3	1	2
dupl	1	3	2
z 1:10 1	1	3	2
z 1:10 2	1	3	2
z 1:10 3	1	3	2
cl 1:10 1	1	3	2
cl 1:10 2	1	3	2
cl 1:10 3	1	3	2
z 1:5 1	1	3	2
z 1:5 2	1	3	2
z 1:5 3	1	3	2
cl 1:5 1	1	3	2
cl 1:5 2	1	3	2
cl 1:5 3	1	3	2
z 1:1 1	1	3	2
z 1:1 2	1	3	2
z 1:1 3	1	3	2
cl 1:1 1	1	3	2
cl 1:1 2	2	3	1
cl 1:1 3	2	3	1
cl all 1:1 1		2	1
cl all 1:1 2		2	1
cl all 1:1 3	1	3	2
cl med 1:1	1	3	2
cl med 1:2	1	3	2
cl med 1:3	1	3	2

J Datenmatrix bei 4 Variablen für Entscheidungsbäume

	ma_zahl	di_ebest	anz_7409	di_l_wa
	Nr. 8	Nr. 22	Nr. 53	Nr. 59
alle	3	2	4	1
profit	2	4	1	2
dupl	3	4	1	2
z 1:10 1	3	1	4	2
z 1:10 2	3	1	4	2
z 1:10 3	3	1	4	2
cl 1:10 1	3		2	1
cl 1:10 2	2		3	1
cl 1:10 3	3		2	1
z 1:5 1	3		2	1
z 1:5 2	3		2	1
z 1:5 3	3		2	1
cl 1:5 1	3		2	1
cl 1:5 2	2		3	1
cl 1:5 3	2		3	1
z 1:1 1			2	1
z 1:1 2	2		3	1
z 1:1 3	2		3	1
cl 1:1 1	2	3	1	2
cl 1:1 2	2	3	1	2
cl 1:1 3	2	3	1	2
cl all 1:1 1	3	2	1	2
cl all 1:1 2	3	3	1	2
cl all 1:1 3	3	4	1	2
cl med 1:1	2	3	4	1
cl med 1:2	2		1	
cl med 1:3	3	4	2	1

K Informationen zum SAS Institute

SAS Unternehmensporträt

Zahlen und Fakten

1976 durch Dr. James H. Goodnight gegründet, ist SAS heute das weltweit größte private Software-Unternehmen und führender Anbieter von Business Intelligence-Lösungen.

Hauptsitz des Unternehmens ist Cary, North Carolina. Sitz von SAS Deutschland ist Heidelberg. Von hier aus werden die Niederlassungen in Berlin, Frankfurt/M., Hamburg, Köln und München betreut. In Heidelberg befindet sich auch das internationale Headquarter für die Regionen EMEA (Europa, Naher Osten und Afrika) und Asien/Pazifik.

Der Umsatz der SAS Institute GmbH ist in Deutschland in 2002 um 3,1 Prozent auf 131 Millionen Euro gestiegen. Weltweit hat SAS einen Umsatz von 1,18 Milliarden US-Dollar erzielt und setzt damit sein kontinuierliches Wachstum zum 26. Mal in Folge fort.

25 Prozent seines Jahresumsatzes investiert SAS traditionell in Forschung und Entwicklung.

Weltweit sind bei SAS mehr als 9.000 Mitarbeiter in 138 Niederlassungen tätig. In Deutschland sind derzeit mehr als 730 Mitarbeiter beschäftigt.

Mehr als 40.000 Unternehmen und Organisationen (öffentliche Verwaltungen, Universitäten, Forschung) nutzen die SAS Software – darunter 90 Prozent der weltweiten Fortune-500-Unternehmen.

Lösungen

SAS ist der weltweit führende Anbieter von Business Intelligence-Lösungen, die aus Geschäftsdaten konkretes Wissen („Intelligence“) für strategische Entscheidungen gewinnen. Mit den SAS Lösungen können Unternehmen ihre Kunden- und Lieferantenbeziehungen profitabel machen und die gesamte Organisation steuern – und damit Kosten reduzieren, Risiken minimieren und die Wertschöpfung vergrößern.

Zu den SAS Lösungen gehören unter anderem die Analytical Applications für Strategic Performance Management, Supply Chain Intelligence, Customer Intelligence und Risikomanagement. Damit erhalten Fachabteilungen in Unternehmen vorkonfigurierte Lö-

sungspakete für ihre spezifischen Arbeitsbereiche. Diese hat SAS aus praxiserprobten Business Intelligence-Anwendungen geschnürt. Die Analytical Applications enthalten standardisierte Frontends, Datenmodelle und Zugriffsroutinen. Außerdem bringt SAS sein langjähriges Data Mining-Know-how ein. Vordefinierte Methoden und Arbeitsweisen erleichtern den Anwendern fundierte Analysen. Auf dieser Basis lassen sich Strategien aufsetzen sowie interne und externe Prozesse effizienter gestalten.

Technologische Basis der analytischen Anwendungen ist die SAS Intelligence Architecture. In dieser Architektur sind alle Funktionalitäten und Prozesse vereint, um aus operativen Daten Wissen zu gewinnen. Herausragendes Merkmal dieser Plattform ist der flexible, modulare Aufbau.

SAS richtet sich mit seinen Lösungen an Kunden aus allen Branchen: von Finanzdienstleistern bis zu Pharmaunternehmen, von Industrie bis Telekommunikation, von Transport bis Handel. Auch Wissenschaft und Forschung arbeitet mit SAS Lösungen: Forscher zahlreicher Universitäten verlassen sich bei ihren Studien auf die umfangreichen statistischen Auswertungsmöglichkeiten, die ihnen die SAS Software bietet.

Uwe Steinlein
Klopstockstr. 6
80804 München
Email: steinlein@web.de

Persönliche Angaben

Familienstand:	ledig
Staatsangehörigkeit:	deutsch
Geburtsdatum:	17.07.1972
Geburtsort:	München

Schulbildung

September 1983 bis Juli 1992	Gymnasium Gilching

Berufsausbildung

Oktober 1992 bis Juli 1994	Banklehre bei der Bayerischen Vereinsbank AG, München

Studium

Oktober 1994 bis August 1999	Studium der Betriebswirtschaftslehre an der Universität Augsburg

Berufspraxis

September 1999 bis August 2002	WEKA Media GmbH, Kissing Doktorandenvertrag, Customer Research
seit Oktober 2002	BMW Bank GmbH, München Manager analytisches CRM

WEKA MEDIA GmbH & Co. KG
Römerstraße 4
86438 Kissing

Fon 0 82 33.23-0
Fax 0 82 33.23-75 00
Email: info@weka.de

Mehr Infos unter:
www.weka.de