Faculty of Science
Department of Informatics
University of Fribourg, Switzerland

Inductive Fuzzy Classification in Marketing Analytics

Doctoral Thesis

presented to the Faculty of Science
of the University of Fribourg, Switzerland
for the award of the academic grade of
Doctor of Computer Science,
Doctor scientiarum informaticarum, Dr. sc. inf.
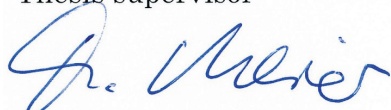
by

Michael Alexander Kaufmann

from Recherswil SO

Thesis No: 1751
Bern: Ruf
2012

Accepted by the Faculty of Science of the University of Fribourg (Switzerland) upon the recommendation of Prof. Dr. Ulrich Ultes-Nitsche (president of the jury), Prof. Dr. Andreas Meier (thesis supervisor, examiner) and Prof. Dr. Laurent Donzé (co-examiner) from the University of Fribourg, and Prof. Dr. Kilian Stoffel, Université de Neuchâtel (co-examiner)

in Fribourg, on May 11th, 2012.

Thesis supervisor   Dean

Prof. Dr. Andreas Meier   Prof. Dr. Rolf Ingold

For Alice.
Follow the white rabbit.

"The Caterpillar took the hookah out of its mouth and yawned once or twice, and shook itself. Then it got down off the mushroom, and crawled away into the grass, merely remarking as it went, "One side will make you grow taller, and the other side will make you grow shorter."
"One side of *what*? The other side of *what*? " thought Alice to herself.
"Of the mushroom," said the Caterpillar, just as if she had asked it aloud; and in another moment it was out of sight."

Lewis Carrol (1916, p. 29)

4

Acknowledgements

Abstract

"Inductive fuzzy classification" (IFC) is the process of assigning individuals to fuzzy sets for which membership functions are based on inductive inferences. In this thesis, different methods for membership function induction and multivariate aggregation are analyzed. For univariate membership function induction, the current thesis proposes the normalized comparisons (ratios and differences) of likelihoods. For example, a normalized likelihood ratio can represent a membership degree to an inductive fuzzy class. If the domain of the membership function is numeric, continuous membership functions can be derived using piecewise affine interpolation. If a target attribute is continuous, it can be mapped into the "Zadehan" domain of numeric truth-values between 0 and 1, and membership degrees can be computed by a normalized ratio of likelihoods of fuzzy events.

A methodology for multivariate IFC for prediction has been developed a part of this thesis: First, data is prepared into a matrix format. Second, the relevant attributes are selected. Third, for each relevant attribute, a membership function to the target class is induced. Fourth, transforming the attributes into membership degrees in the inductive fuzzy target class fuzzifies these attributes. Fifth, for every data record, the membership degrees of the single attribute values are aggregated into a membership degree in the target class. Finally, the prediction accuracy of the inductive membership degrees in comparison to the target variable is evaluated.

The proposed membership function induction method can be applied to analytics for selection, visualization, and prediction. First, transformation of attributes into inductive membership degrees in fuzzy target classes provides a way to test the strength of target associations of categorical and numerical variables using the same measure. Thus, relevant attributes with a high target correlation can be selected. Second, the resulting membership functions can be visualized as a graph. This provides an intuitive depiction of target associations for different values of relevant attributes and allows human decision makers to recognize interesting parts of attribute domains regarding the target. Third, transformation of attribute values into membership degrees in inductive fuzzy classes can improve prediction of statistical models because nonlinear associations can be represented in membership functions.

In marketing analytics, these methods can be applied in several domains. Customer analytics on existing data using attribute selection and visualization of inductive membership functions provide insights into different customer classes. Product analytics can be improved by

evaluating likelihood of product usage in the data for different customer characteristics with inductive membership functions. Transforming customer attributes into inductive membership degrees, which are based on predictive models, can optimize target selection for individual marketing and enhance the response rate of campaigns. This can be embedded in integrated analytics for individual marketing.

A case study is presented, in which IFC was applied in online marketing of a Swiss financial service provider. Fuzzy classification was compared to crisp classification and to random selection. The case study showed that, for individual marketing, a scoring approach can lead to better response rates than a segmentation approach because of compensation of threshold effects.

A prototype was implemented that supports all steps of the prediction methodology. It is based on a script interpreter that translates inductive fuzzy classification language (IFCL) statements into corresponding SQL commands and executes them on a database server. This software supports all steps of the proposed methodology, including data preparation, membership function induction, attribute selection, multivariate aggregation, data classification and prediction, and evaluation of predictive models.

The software IFCL was applied in an experiment in order to evaluate the properties of the proposed methods for membership function induction. These algorithms were applied to 60 sets of real data, of which 30 had a binary target variable and 30 had a gradual target variable, and they were compared to existing methods. Different parameters were tested in order to induce an optimal configuration of IFC. Experiments showed a significant improvement in average predictive performance of logistic regression for binary targets and regression trees for gradual targets when, prior to model computation, the attributes were inductively fuzzified using normalized likelihood ratios or normalized likelihood differences respectively.

8

Zusammenfassung

Induktive unscharfe Klassifikation (inductive fuzzy classification, IFC) ist der Prozess der Zuordnung von Individuen zu unscharfen Mengen, deren Zugehörigkeitsfunktionen auf induktiven Schlussfolgerungen basieren. In der vorliegenden These werden verschiedene Methoden für die Induktion von Zugehörigkeitsfunktionen und deren multivariaten Aggregation analysiert. Es wird vorgeschlagen, für die Induktion von univariaten Zugehörigkeitsfunktionen Vergleiche (Verhältnisse und Differenzen) von Wahrscheinlichkeiten (Likelihoods) zu normalisieren. Zum Beispiel kann das normalisierte Wahrscheinlichkeitsverhältnis (normalized Likelihood Ratio, NLR) einen Zugehörigkeitsgrad zu einer induktiven unscharfen Klasse repräsentieren. Wenn der Wertebereich der Zugehörigkeitsfunktion nummerisch ist, können stetige Zugehörigkeitsfunktionen über eine stückweise affine Interpolation hergeleitet werden. Wenn die Zielvariable stetig ist, kann sie in den „Zadeh'schen" Wertebereich der nummerischen Wahrheitswerte zwischen 0 und 1 abgebildet werden; und die Zugehörigkeitsgrade können als normalisierte Verhältnisse von empirischen bedingten Wahrscheinlichkeiten unscharfer Ereignisse (Likelihoods of fuzzy events) berechnet werden.

Eine Methode für multivariate induktive unscharfe Klassifikation wurde in dieser These entwickelt. Als erstes werden die Daten in einem Matrix-Format aufbereitet. Als zweites werden die relevanten Attribute ausgewählt. Drittens wird für jedes relevante Attribut eine Zugehörigkeitsfunktion zur Zielklasse mittels normalisierten Likelihood-Vergleichen induziert. Viertens werden die Attributwerte in Zugehörigkeitsgrade zur induktiven unscharfen Zielklasse transformiert. Fünftens werden die Zugehörigkeitsgrade der einzelnen Attributwerte jedes Datensatzes in einen Zugehörigkeitsgrad in der Zielklasse aggregiert. Schliesslich wird die Vorhersageleistung der induktiven Zugehörigkeitsgrade im Vergleich mit der effektiven Zielvariable evaluiert.

Die vorgeschlagenen Methoden können in der Analytik für Selektion, Visualisierung und Prognose angewendet werden. Erstens bietet die Transformation von Attributen in Zugehörigkeitswerte zu induktiven unscharfen Mengen ein Mittel, um die Stärke der Zielassoziation von nummerischen und kategorischen Variablen mit der gleichen Metrik zu testen. So können relevante Attribute mit einer hohen Korrelation mit dem Ziel selektiert werden. Zweitens können die resultierenden Zugehörigkeitsfunktionen als Graphen visualisiert werden. Das bietet eine intuitive Darstellung der Zielassoziation von

verschiedenen Werten von relevanten Attributen, und ermöglicht es menschlichen Entscheidungsträgern, die interessanten Teilbereiche der Attributdomänen bezüglich des Ziels zu erkennen. Drittens kann die Transformation der Attributwerte in Zugehörigkeitswerte zu induktiven unscharfen Klassen die Prognose von statistischen Modellen verbessern, weil nicht-lineare Zusammenhänge in den Zugehörigkeitsfunktionen abgebildet werden können.

In der Marketing-Analytik können diese Methoden in verschiedenen Bereichen angewendet werden. Kundenanalytik aufgrund von bestehenden Daten mittels Attributselektion und Visualisierung von induktiven Zugehörigkeitsfunktionen liefert Erkenntnisse über verschiedene Kundenklassen. Produktanalytik kann verbessert werden, indem die Wahrscheinlichkeit der Produktnutzung in den Daten für verschiedene Kundenmerkmale mit induktiven Zugehörigkeitsfunktionen evaluiert wird. Zielgruppenselektion im individualisierten Marketing kann optimiert werden, indem Kundenattribute in induktive Zugehörigkeitsgrade transformiert werden, um die Rücklaufrate von Kampagnen zu erhöhen, welche auf prädiktiven Modellen basieren. Dies kann für individualisiertes Marketing in die integrierte Analytik eingebettet werden.

Eine Fallstudie wird vorgestellt, in der die induktive unscharfe Klassifikation bei einem Schweizer Finanzdienstleister im online Marketing angewendet wurde. Die unscharfe Klassifikation wurde mit der scharfen Klassifikation und einer Zufallsauswahl verglichen. Dies zeigte, dass im individualisierten Marketing ein Scoring-Ansatz zu besseren Rücklaufquoten führen kann als ein Segmentierungsansatz, weil Schwellenwerteffekte kompensiert werden können.

Ein Prototyp wurde implementiert, welcher alle Schritte der multivariaten IFC-Methode unterstützt. Er basiert auf einem Skript-Interpreter, welcher Aussagen der Sprache für die induktive unscharfe Klassifikation (Inductive Fuzzy Classification Language, IFCL) in entsprechende SQL-Kommandos übersetzt und auf einem Datenbankserver ausführt. Diese Software unterstützt sämtliche Schritte der vorgeschlagenen Methode, einschliesslich Datenvorbereitung, Induktion von Zugehörigkeitsfunktionen, Attributselektion, multivariate Aggregation, Datenklassifikation, Prognose, und die Evaluation von Vorhersagemodellen.

Die Software wurde in einem Experiment angewendet, um die Eigenschaften der vorgeschlagenen Methoden der Induktion von Zugehörigkeitsfunktionen zu testen. Diese Algorithmen wurden bei sechzig Dateien mit realen Daten angewendet, davon bei dreissig mit binärer Zielvariable und dreissig mit einer graduellen Zielvariable. Sie wurden mit existierenden Methoden verglichen. Verschiedene Parameter wurden getestet, um eine optimale Konfiguration der

induktiven unscharfen Klassifikation zu induzieren. Die Experimente
zeigten, dass die durchschnittliche Prognoseleistung der logistischen
Regression für binäre Zielvariablen und der Regressionsbäume für
graduelle Zielvariablen signifikant verbessert werden kann, wenn vor
der Berechnung der Prognosemodelle die Attribute mit normalisierten
Likelihood-Verhältnissen respektive mit normalisierten Likelihood-
Differenzen in Zugehörigkeitsgrade zu induktiven unscharfen Mengen
transformiert werden.

Table of Contents

☺ This section is based on a publication by Kaufmann & Meier (2009)

☼ This section is based on a publication by Kaufmann & Graf (2012)

15

List of Figures

List of Tables

1 A Gradual Concept of Truth

"Would you describe a single grain of wheat as a heap? No. Would you describe two grains of wheat as a heap? No. ... You must admit the presence of a heap sooner or later, so where do you draw the line?" (Hyde, 2008)

How many grains does it take to constitute a heap? This question is known as the sorites paradox (Hyde, 2008). It exemplifies that our semantic universe is essentially vague, and with any luck, this vagueness is ordinal and gradual. This applies to all kinds of statements. Especially in science, different propositions or hypotheses can only be compared to each other with regard to their relative accuracy or predictive power. *Fuzziness* is a term that describes vagueness in the form of boundary imprecision. Fuzzy concepts are those that are not clearly delineated, such as the concept of a "heap of grain."

The classical notion of truth claims metaphysical dualism, and thus divides thought into exactly two categories: true and false. A gradual concept of truth leads our consciousness toward a *metaphysical monism*: all possible statements belong to the same class; but there is a gradation of *degree*. The *continuum of propositions*, ranging from completely false to completely true, contains all the information that is in-between.

Fuzzy set theory provides a tool for mathematically precise definitions of fuzzy concepts, if those concepts can be ordered: assigning gradual membership degrees to their elements. This gradual concept of truth is the basis for fuzzy logic or *approximate reasoning*, as proposed by Zadeh (1975a) and Bellmann and Zadeh (1977). Fuzzy logic, based on the concept of fuzzy sets introduced by Zadeh (1965), allows propositions with a gradual truth-value and, thus, supports approximate reasoning, gradual and soft consolidation.

Fuzzy propositions define *fuzzy classes,* which allow gradual, fuzzy class boundaries. In data analysis, or *"the search for structure in data"* (Zimmermann H. J., 1997), fuzzy classification is a method for gradation in data consolidation, as presented by Meier, Schindler, and Werro (2008) and Del Amo, Montero, and Cutello (1999). The application of fuzzy classification to marketing analytics (Spais & Veloutsou, 2005) has the advantage of precisiation (sic; Zadeh, 2008) of fuzzy concepts in the context of decision support for direct customer contact, as proposed by Werro (2008). This precisiation can be achieved by inducing membership functions to fuzzy target classes (Setnes, Kaymak, & van Nauta Lemke, 1998).

Inductive, or probabilistic, inference and fuzzy, or gradual, logic have been seen as incompatible, for example, by Elkan (1993). Whereas probabilistic induction can amplify experience, membership functions can precisiate vagueness. Combined, measured probabilities can be applied to precisely define the semantics of vague or fuzzy concepts by membership function induction. This thesis, eventually, demonstrates how probabilistic and fuzzy logics can be synthesized to constitute a method of inductive gradual reasoning for classification.

1.1 Research Questions

Werro (2008) and Meier et al. (2008) proposed the application of Fuzzy classification to customer relationship management (CRM). A suggestion for further research by Werro (2008), namely the integration of data mining techniques into fuzzy classification software, has inspired the present thesis. As illustrated in Figure 1, the motivation of the current research was to develop and evaluate inductive methods for the automated derivation of membership functions for fuzzy classification and to propose possible applications to marketing analytics.

The first motivation for research on "*inductive fuzzy classification*" (IFC) was the development of methods for automated derivations of understandable and interpretable membership functions to fuzzy classes. The aim was to develop and evaluate algorithms and methods to induce functions that indicate inductively inferred target class memberships.

The second motivation was to improve marketing analytics with IFC. Kaufmann and Meier (2009) have presented a methodology and a case study for the application of IFC to predictive product affinity scoring for target selection. This thesis extends the application of this methodology to marketing in the field of integrated customer and product analytics in order to provide means to deal with fuzziness in marketing decisions and to enhance accountability by application of analytics to precisiate fuzzy marketing concepts, as proposed by Spais and Veloutsou (2005).

The third motivation was to develop a prototype implementation of software for computing IFCs, showing the feasibility of the proposed algorithms as a proof of concept and allowing an evaluation of the proposed methodology.

Seven research questions were developed at the beginning of the current research (Kaufmann, 2008). These questions guided the development of the thesis from the beginning. The answers are

presented in the subsequent document and summarized in the conclusions of the thesis.

1. What is the theoretical basis of IFC and what is its relation to inductive logic?

2. How can membership functions be derived inductively from data?

3. How can a business enterprise apply IFC in practice?

4. How can the proposed methods be implemented in a computer program?

5. How is IFC optimally applied for prediction?

6. Which aggregation methods are optimal for the multivariate combination of fuzzy classes for prediction?

7. Can it be statistically supported that the proposed method of IFC improves predictive performance?
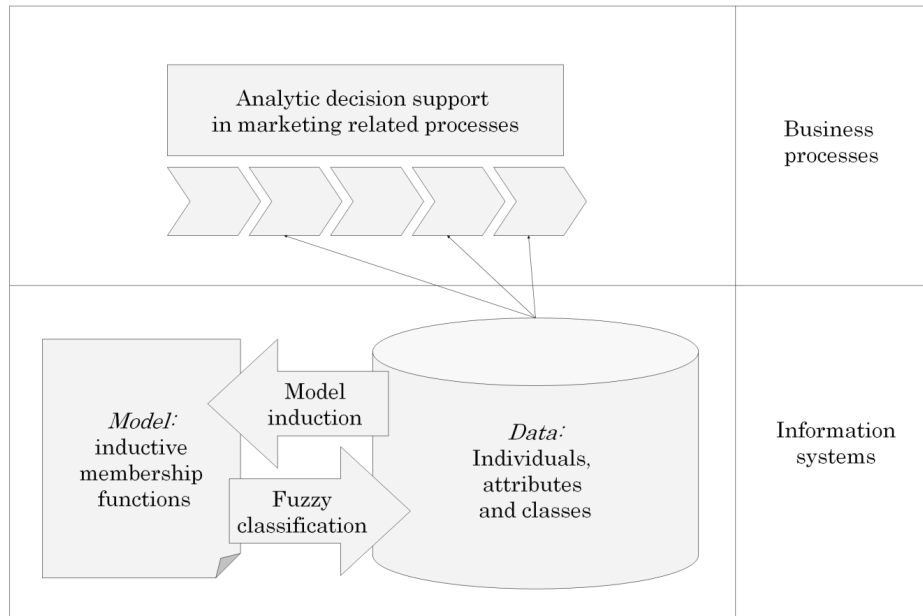
Figure 1: The motivation of the current research was to develop and evaluate inductive methods for the automated derivation of membership functions for fuzzy classification and to propose possible applications to marketing analytics.

Following a constructive approach to business informatics and information systems research (Oesterle, et al., 2010), this thesis presents a design of new methods for IFC and proposes applications of

these new methods to analytics in marketing. Thus, the methodology for developing the thesis includes the following: The theoretical background is analyzed prior to presentation of the constructive design, a case study exemplifies the designed approach, software prototyping shows technical feasibility and enables experimental evaluation, and empirical data collection in combination with statistical inference is applied to draw conclusions about the proposed method.

1.2   Thesis Structure

As shown in Figure 2, this thesis has three thematic categories that are reflected in the structure. First, the theoretical foundations of IFC are examined. This theory is based on a synthesis of fuzzy (gradual) and inductive (probabilistic) logics. Second, business applications of this approach are analyzed. A possible application is studied for analytic quantitative decision support in marketing. Third, technological aspects of the proposed method are examined by software prototypes for evaluation of the proposed constructs.

The second chapter, which addresses theory, analyzes the theoretical foundations of IFC by approaching it from the viewpoints of logic, fuzziness, and induction. It presents proposals of likelihood-based methods for membership function induction. In Section 2.1, the basic concepts of logic and classification are analyzed. In Section 2.2, cognitive and conceptual fuzziness are discussed and approaches for resolution of conceptual fuzziness, fuzzy logic, and fuzzy classification are outlined. In Section 2.3, inductive classification, and induction automation are examined. In Section 2.4, the application of induction to fuzzy classification is explained, and a methodology for membership function induction using normalized ratios and differences of empirical conditional probabilities and likelihoods is proposed.

The third chapter, which deals with designing applications of IFC, presents a study of analytics and examines the application of membership function induction to this discipline in marketing. In Section 3.1, the general methodology of logical data analysis, called analytics, is analyzed, and applications of IFC to three sub-disciplines, selection, visualization, and prediction, are proposed. In Section 3.2, applications of IFC to analytic marketing decision support, or marketing analytics (MA), are listed. A case study is presented, in which the proposed methods are applied to individual online marketing.

The fourth chapter, which presents designs of technology for IFC, explains prototype implementations of the proposed IFC methodology and shows evaluation results. In Section 4.1, three prototype implementations of IFC are presented. The prototype of an inductive

fuzzy classification language (IFCL) is described. This description encompasses the architecture of the software and its functionality. Two additional prototypes that were developed in collaboration with master's students guided by the author, iFCQL (Mayer, 2010) and IFC-Filter (Graf, 2010), are briefly discussed. In Section 4.2, a systematic experimental application of the IFCL prototype is described and the empirical results of testing the predictive performance of different parameters of the proposed methodology are statistically analyzed.
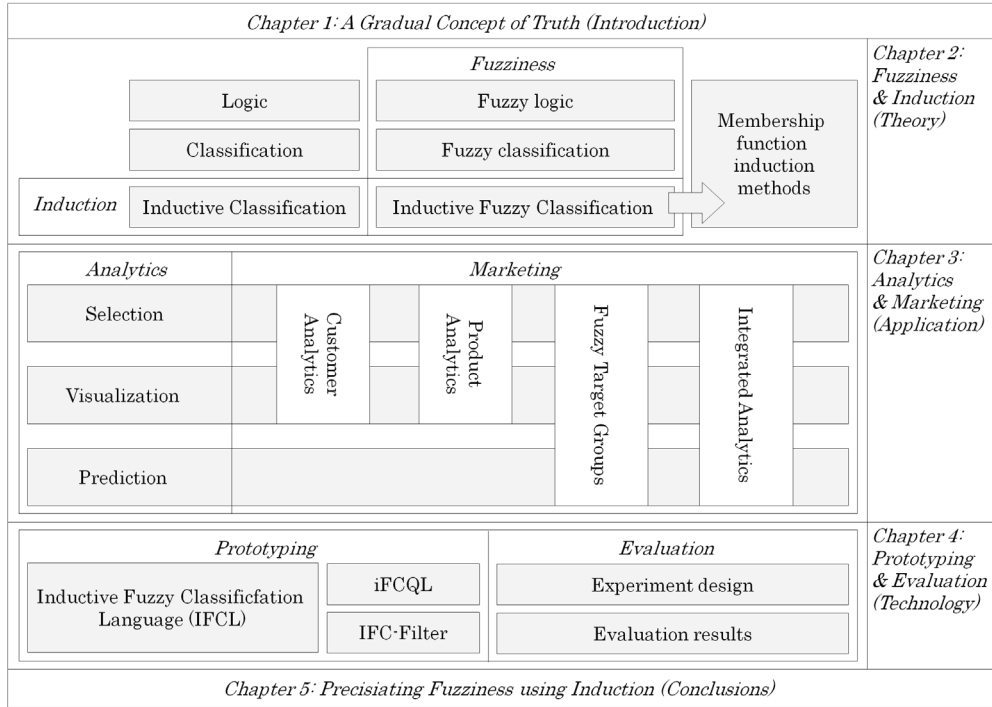
Figure 2: Thesis structure. This thesis has three thematic categories that are reflected in the structure: theory, application, and technology.

The fifth chapter, which concludes the thesis, summarizes the scientific contributions of the current research, discusses the results, and proposes further research topics for IFC. Literature references are indicated in the sixth chapter. Research details can be found in Chapter A, the appendix at the end of the thesis.

2 Fuzziness & Induction

This chapter examines the foundations of IFC by analyzing the concepts of deduction, fuzziness, and induction. The first subsection explains the classical concepts of sharp and deductive logic and classification; in this section, it is presupposed that all terms are clearly defined. The second section explains what happens when those definitions have fuzzy boundaries and provides the tools, fuzzy logic and fuzzy classification, to reason about this. However, there are many terms that do not only lack a sharp boundary of term definition but also lack a priori definitions. Therefore, the third subsection discusses how such definitions can be inferred through inductive logic and how such inferred propositional functions define inductive fuzzy classes. Finally, this chapter proposes a method to derive precise definitions of vague concepts—membership functions—from data. It develops a methodology for membership function induction using normalized likelihood comparisons, which can be applied to fuzzy classification of individuals.

2.1 Deduction

This subsection discusses deductive logic and classification, analyzes the classical as well as the mathematical (Boolean) approaches to propositional logic, and shows their application to classification. Deduction provides a set of tools for reasoning about propositions with a priori truth-values—or inferences of such values. Thus, in the first subsection, the concepts of classical two-valued logic and algebraic Boolean logic are summarized. The second subsection explains how propositional functions imply classes and, thus, provide the mechanism for classification.

2.1.1 Logic

In the words of John Stuart Mill (1843), logic is "the science of reasoning, as well as an art, founded on that science" (p. 18). He points out that the most central entity of logic is the statement, called a *proposition*:

> The answer to every question which it is possible to frame, is contained in a proposition, or assertion. Whatever can be an object of belief, or even of disbelief, must, when put into words, assume the form of a proposition. All truth and all error lie in propositions. What, by a convenient misapplication of an abstract term, we call a truth, is simply a true proposition. (p. 27)

The central role of propositions indicates the importance of linguistics in philosophy. Propositions are evaluated for their truth, and thus, assigned a truth-value because knowledge and insight is based on true statements.

Consider the universe of discourse in logic: The set of possible statements or propositions, $\mathcal{P}$. Logicians believe that there are different levels of truth, usually two (true or false); in the general case, there is a set, $\mathcal{T}$, of possible truth-values that can be assigned to propositions. Thus, the proposition $p \in \mathcal{P}$ is a meaningful piece of information to which a truth-value, $\tau(p) \in \mathcal{T}$, can be assigned. The corresponding mapping of $\tau: \mathcal{P} \longrightarrow \mathcal{T}$ from propositions $\mathcal{P}$ to truth-values $\mathcal{T}$ is called a *truth function*.

In general logic, operators can be applied to propositions. A unary operator, $O_1: \mathcal{P}_1 \longrightarrow \mathcal{T}$, maps a single proposition into a set of transformed truth-values. Accordingly, a binary operator, $O_2: \mathcal{P}_1 \times \mathcal{P}_2 \longrightarrow \mathcal{T}$, assigns a truth-value to a combination of two propositions, and an *n*-ary operator, $O_n: \mathcal{P}_1 \times \cdots \times \mathcal{P}_n \longrightarrow \mathcal{T}$, is a mapping of a combination of n propositions to a new truth-value.

The logic of two-valued propositions is the science and art of reasoning about statements that can be either true or false. In the case of two-valued logic, or *classical logic* (CL), the set of possible truth values, $\mathcal{T}^{CL} := \{\text{true, false}\}$, contains only two elements, which partitions the class of imaginable propositions $\mathcal{P}$ into exactly two subclasses: the class of false propositions and the class of true ones.

With two truth-values, there is only one possible unary operator other than identity: A proposition, $p \in \mathcal{P}$, can be *negated* (not p), which inverts the truth-value of the original proposition. Accordingly, for a combination of two propositions, p and q, each with two truth-values, there are 16 (2^{2^2}) possible binary operators. The most common binary logical operators are disjunction, conjunction, implication, and equivalence: A *conjunction* of two propositions, p and q, is true if both propositions are true. A *disjunction* of two propositions, p or q, is true if one of the propositions is true. An *implication* of q by p is true if, whenever p is true, q is true as well. An *equivalence* of two propositions is true if p implies q and q implies p.

Classical logic is often formalized in the form of a propositional calculus. The syntax of classical propositional calculus is described by the concept of variables, unary and binary operators, formulae, and truth functions. Every proposition is represented by a variable (e.g., p); every proposition and every negation of a proposition is a term; every combination of terms by logical operators is a formula; terms and formulae are themselves propositions; negation of the proposition p is represented by $\neg p$; conjunction of the two propositions p and q is represented by $p \wedge q$; disjunction of the two propositions p and q is

represented by $p \lor q$; implication of the proposition q by the proposition p is represented by $p \Rightarrow q$; equivalence between the two propositions p and q is represented by $p \equiv q$; and there is a truth function, $\tau^{CL}: \mathcal{P} \rightarrow \mathcal{T}^{CL}$, mapping from the set of propositions $\mathcal{P}$ into the set of truth values $\mathcal{T}$. The semantics of propositional calculus are defined by the values of the truth function, as formalized in Formula 1 through Formula 5.

$$\tau^{CL}(\neg\, p) := \begin{cases} \text{if}(\tau(p) = \text{true}) & \text{false} \\ \text{else} & \text{true.} \end{cases}$$

<div align="right">Formula 1</div>

$$\tau^{CL}(p \land q) := \begin{cases} \text{if } (\tau(p) = \tau(q) = \text{true}) & \text{true} \\ \text{else} & \text{false.} \end{cases}$$

<div align="right">Formula 2</div>

$$\tau^{CL}(p \lor q) := \begin{cases} \text{if } (\tau(p) = \tau(q) = \text{false}) & \text{false} \\ \text{else} & \text{true.} \end{cases}$$

<div align="right">Formula 3</div>

$$\tau^{CL}(p \Rightarrow q) := \tau^{CL}(\neg\, p \lor q)$$

<div align="right">Formula 4</div>

$$\tau^{CL}(p \equiv q) := \tau^{CL}(p \Rightarrow q \land q \Rightarrow p)$$

<div align="right">Formula 5</div>

George Boole (1847) realized that logic can be calculated using the numbers 0 and 1 as truth values. His conclusion was that logic is mathematical in nature:

> I am then compelled to assert, that according to this view of the nature of Philosophy, Logic forms no part of it. On the principle of a true classification, we ought no longer to associate Logic and Metaphysics, but Logic and Mathematics. (p. 13)

In Boole's mathematical definition of logic, the numbers 1 and 0 represents the truth-values and logical connectives are derived from arithmetic operations: subtraction from 1 as negation and multiplication as conjunction. All other operators can be derived from these two operators through application of the laws of logical equivalence. Thus, in Boolean logic (*BL*), the corresponding propositional calculus is called *Boolean algebra,* stressing the

conceptual switch from metaphysics to mathematics. Its syntax is defined in the same way as that of CL, except that the Boolean truth function, $\tau^{BL} \colon \mathcal{P} \longrightarrow \mathcal{T}^{BL}$, maps from the set of propositions into the set of Boolean truth values, $\mathcal{T}^{BL} := \{0,1\}$, that is, the set of the two numbers 0 and 1.

The Boolean truth function τ^{BL} defines the semantics of Boolean algebra. It is calculated using multiplication as conjunction and subtraction from 1 as negation, as formalized in Formula 6 through Formula 8. Implication and equivalence can be derived from negation and disjunction in the same way as in classical propositional calculus.

$$\tau^{BL}(\neg\, p) := 1 - \tau^{BL}(p) \qquad\qquad \text{Formula 6}$$

$$\tau^{BL}(p \wedge q) := \tau^{BL}(p) \cdot \tau^{BL}(q) \qquad\qquad \text{Formula 7}$$

$$\begin{aligned} \tau^{BL}(p \vee q) &:= \neg\,(\neg p \wedge \neg q) \\ &= 1 - \left(1 - \tau^{BL}(p)\right) \cdot \left(1 - \tau^{BL}(q)\right) \end{aligned} \qquad \text{Formula 8}$$

2.1.2   Classification

Class logic, as defined by Glubrecht, Oberschelp, and Todt (1983), is a logical system that supports statements applying a *classification operator.* Classes of objects can be defined according to logical propositional functions. According to Oberschelp (1994), a class, $C = \{\, i \in U \mid \Pi(i)\,\}$, is defined as a collection of individuals, i, from a universe of discourse, U, satisfying a propositional function, Π, called the *classification predicate*. The domain of the classification operator, $\{\,.\mid.\,\} \colon \mathbb{P} \longrightarrow U^{*}$, is the class of propositional functions $\mathbb{P}$ and its range is the powerclass of the universe of discourse U^{*}, which is the class of possible subclasses of U. In other words, the class operator assigns subsets of the universe of discourse to propositional functions. A universe of discourse is the set of all possible individuals considered, and an individual is a real object of reference. In the words of Bertrand Russell (1919), a propositional function is "an expression containing one or more undetermined constituents, such that, when values are assigned to these constituents, the expression becomes a proposition" (p. 155).

In contrast, *classification* is the process of grouping individuals who satisfy the same predicate into a class. A (Boolean) classification corresponds to a membership function, $\mu_C : U \longrightarrow \{0,1\}$, which indicates with a Boolean truth-value whether an individual is a member of a class, given the individual's classification predicate. As shown by Formula 9, the membership μ of individual i in class $C = \{\, i \in U \mid \Pi(i)\,\}$ is defined by the truth-value τ of the classification predicate $\Pi(i)$. In Boolean logic, the truth-values are assumed to be certain. Therefore, classification is *sharp* because the truth values are either exactly 0 or exactly 1.

$$\mu_C(i) := \tau\big(\Pi(i)\big) \in \{0,1\} \qquad\qquad \text{Formula 9}$$

Usually, the classification predicate that defines classes refers to attributes of individuals. For example, the class "tall people" is defined by the predicate "tall," which refers to the attribute "height." An attribute, X, is a function that characterizes individuals by mapping from the universe of discourse U to the set of possible characteristics χ (Formula 10).

$$X : U \longrightarrow \chi \qquad\qquad \text{Formula 10}$$

There are different types of values encoding characteristics. *Categorical* attributes have a discrete range of symbolic values. *Numerical* attributes have a range of numbers, which can be natural or real. *Boolean* attributes have Boolean truth-values $\{0,1\}$ as a range. *Ordinal* attributes have a range of categories that can be ordered.

On one hand, the distinction between univariate and multivariate classification, the *variety*, depends on the number of attributes considered for the classification predicate. The *dimensionality* of the classification, on the other hand, depends on the number of dimensions, or linearly independent attributes, of the classification predicate domain.

In a *univariate classification* (UC), the classification predicate Π refers to one attribute, X, which is true for an individual, i, if the feature $X(i)$ equals a certain characteristic, $c \in \chi$.

$$\mu_{UC}(i) := \tau^{BL}(X(i) = c) \qquad\qquad \text{Formula 11}$$

In a *multivariate classification* (MVC), the classification predicate refers to multiple element attributes. The classification predicate is true

for an individual, i, if an aggregation, a, of several characteristic constraints has a given value, $c \in \chi$.

$$\mu_{MVC}(i) := \tau^{BL}\left(a\big(X_1(i), \cdots, X_n(i)\big) = c\right) \qquad \text{Formula 12}$$

A *multidimensional classification* (MDC) is a special case of a multivariate classification that refers to n-tuples of attributes, such that the resulting class is functionally dependent on the combination of all n attributes.

$$\mu_{MDC}(i) := \tau^{BL}\left(\begin{bmatrix} X_1(i) \\ \vdots \\ X_n(i) \end{bmatrix} = \begin{bmatrix} C_1 \\ \vdots \\ C_m \end{bmatrix}\right) \qquad \text{Formula 13}$$

This distinction between multivariate and multidimensional classification is necessary for the construction of classification functions. Multivariate classifications can be derived as functional aggregates of one-dimensional membership functions, in which the influence of one attribute to the resulting aggregate does not depend on the other attributes. In contrast, in multidimensional classification, the combination of all attributes determines the membership value, and thus, one attribute has different influences on the membership degree for different combinations with other attribute values. Therefore, multidimensional classifications need multidimensional membership functions that are defined on n-tuples of possible characteristics.

2.2   Fuzziness

"There are many misconceptions about fuzzy logic. Fuzzy logic is not fuzzy. Basically, fuzzy logic is a precise logic of imprecision and approximate reasoning." (Zadeh, 2008, p. 2751)

Fuzziness, or vagueness (Sorensen, 2008), is an uncertainty regarding concept boundaries. In contrast to ambiguous terms, which have several meanings, vague terms have one meaning, but the extent of it is not sharply distinguishable. For example, the word *tall* can be ambiguous, because a tall cat is usually smaller than a small horse. Nevertheless, the disambiguated predicate "tall for a cat" is vague, because its linguistic concept does not imply a sharp border between tall and small cats.

Our brains seem to love boundaries. Perhaps, making sharp distinctions quickly was a key cognitive ability in evolution. Our brains are so good at recognizing limits, that they construct limits where there are none. This is what many optical illusions are based on: for example, Kaniza's (1976) Illusory Square (Figure 3).
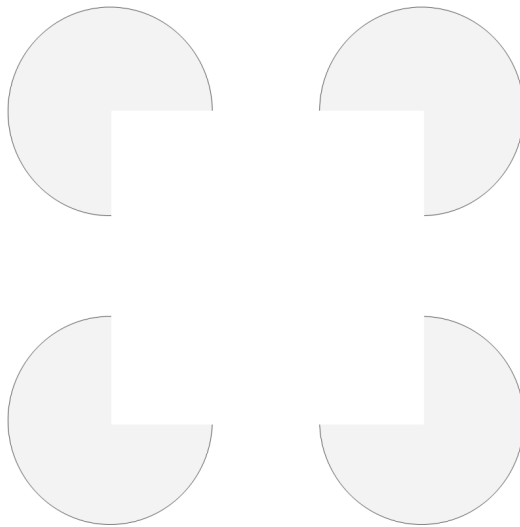
Figure 3: There is no square. Adapted from "Subjective Contours" by G. Kaniza, 1976,

Copyright 1976 by Scientific American, Inc.

An ancient symbol of sharp distinction between classes is the yin and yang symbol (Figure 4). It symbolizes a dualistic worldview—the cosmos divided into light and dark, day and night, and so on.
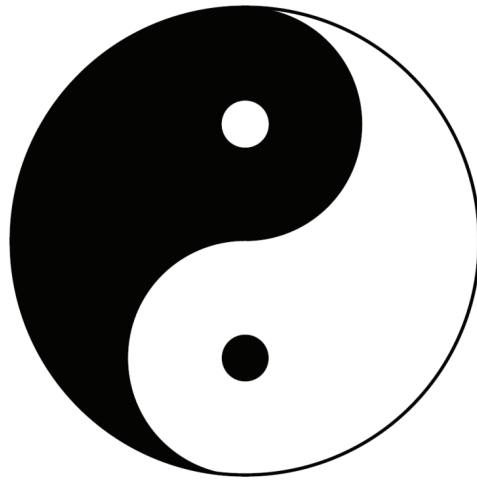
Figure 4: Black or white: conventional yin and yang symbol with a sharp distinction between opposites, representing metaphysical dualism.

Adapted from http://www.texample.net/tikz/examples/yin-and-yang/ (accessed 02.2012) with permission (creative commons license CC BY 2.5).
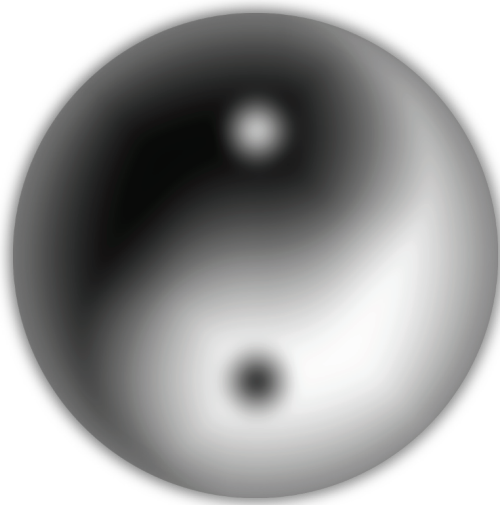
Figure 5: Shades of grey: fuzzy yin and yang symbol with a gradation between opposites, representing metaphysical monism.

Nevertheless, in reality, the transition between light and dark is gradual during the 24 hours of a day. This idea of gradation of our perceptions can be visualized by a fuzzy yin and yang symbol (Figure 5). Sorensen (2008) explains that many-valued logics have been proposed to solve the philosophical implications of vagueness. One many-valued approach to logic is fuzzy logic, which allows infinite truth-values in the interval between 0 and 1.

In the next section, introducing membership functions, fuzzy sets, and fuzzy propositions are discussed; these are the bases for fuzzy logic, which in fact, is a precise logic for fuzziness. Additionally, it is shown how fuzzy classifications are derived from fuzzy propositional functions.

2.2.1   Fuzzy Logic

Lotfi Zadeh (2008) said, "Fuzzy logic is not fuzzy" (p. 2751). Indeed, it is a precise mathematical concept for reasoning about fuzzy (vague) concepts. If the domain of those concepts is ordinal, membership can be distinguished by its *degree*. In classical set theory, an individual, i, of a universe of discourse, U, is either completely a member of a set or not at all. As previously explained, according to Boolean logic, the membership function $\mu_S : U \longrightarrow \{0,1\}$, for a crisp set S, maps from individuals to sharp truth-values. As illustrated in Figure 6, a sharp set (the big dark circle) has a clear boundary, and individuals (the small bright circles) are either a member of it or not. However, one individual is not entirely covered by the big dark circle, but is also not outside of it. In contrast, a set is called *fuzzy* by Zadeh (1965) if individuals can have a gradual degree of membership to it. In a fuzzy set, as shown by Figure 7, the limits of the set are blurred. The degree of membership of the elements in the set is gradual, illustrated by the fuzzy gray edge of the dark circle. The membership function, $\mu_F : U \rightarrow [0,1]$, for a fuzzy set, F, indicates the degree to which individual i is a member of F in the interval between 0 and 1. In Figure 6, the degree of membership of the small circles i is defined by a normalization n of their distance d from the center c of the big dark circle b, $\mu_b(i) = n\big(d(i,c)\big)$. In the same way as in classical set theory, set operators can construct complements of sets and combine two sets by union and intersection. Those operators are defined by the fuzzy membership function. In the original proposal of Zadeh (1965), the set operators are defined by subtraction from 1, minimum and maximum. The complement, $\bar{F}$, of a fuzzy set, F, is derived by subtracting its membership function from 1; the union of two sets, $F \cup G$, is derived from the maximum of the membership degrees; and the intersection of two sets, $F \cap G$, is derived from the minimum of the membership degrees.
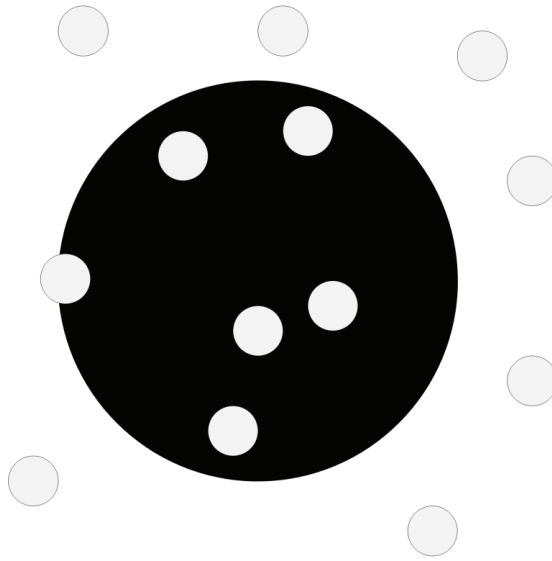
Figure 6: A visualization of a classical set with sharp boundaries.
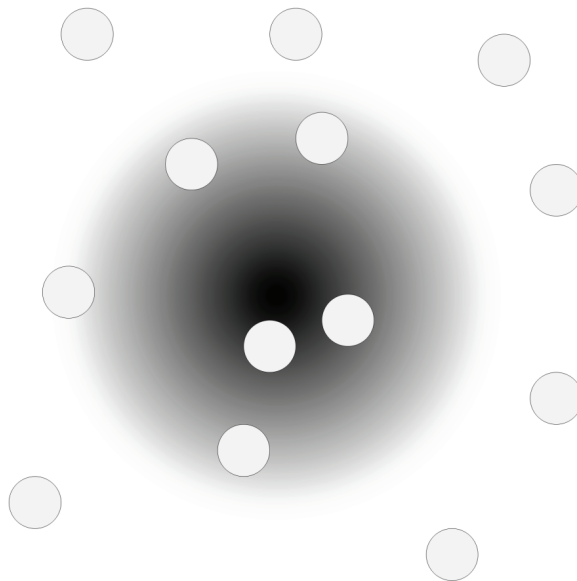
Figure 7: A visualization of a fuzzy set.

Accordingly, fuzzy subsets and fuzzy power sets can be constructed. Consider the two fuzzy sets A and B on the universe of discourse U. In general, A is a fuzzy subset of B if the membership degrees of all its elements are smaller or equal to the membership degrees of elements in B (Formula 14). Thus, a fuzzy power set, $B^{\tilde{*}}$, of a (potentially fuzzy) set B is the class of all its fuzzy subsets (Formula 15).

$$A \cong B := \forall x \in U : \mu_A(x) \leq \mu_B(x) \qquad \text{Formula 14}$$

$$B^{\tilde{*}} := \{A \cong U \mid A \cong B\} \qquad \text{Formula 15}$$

With the tool of fuzzy set theory in hand, the sorites paradox cited in the introduction (Chapter 1) can be tackled in a much more satisfying manner. A heap of wheat grains can be defined as a fuzzy subset, $Heap \cong \mathbb{N}$, of natural numbers $\mathbb{N}$ of wheat grains. A heap is defined in the English language as "a great number or large quantity" (merriam-webster.com, 2012b). For instance, one could agree that 1,000 grains of wheat is a large quantity, and between 1 and 1,000, the "heapness" of a grain collection grows logarithmically. Thus, the membership function of the number of grains $n \in \mathbb{N}$ in the fuzzy set $Heap$ can be defined according to Formula 16. The resulting membership function is plotted in Figure 8.

$$\mu_{Heap}(n) := \begin{cases} 0 \text{ if } n = 0 \\ 1 \text{ if } n > 1000 \\ 0.1448 \ln(n) \text{ else.} \end{cases} \qquad \text{Formula 16}$$
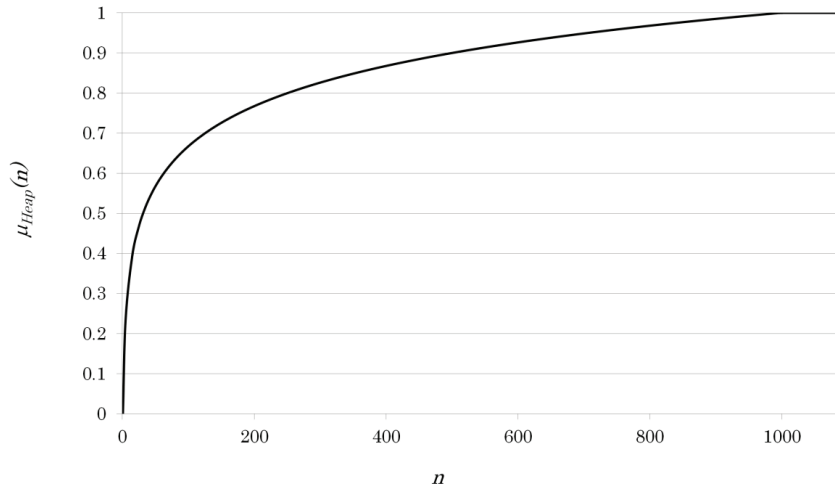
Figure 8: Fuzzy set theory applied to the sorites paradox.

Based on the concept of fuzzy sets, Zadeh (1975a) derived *fuzzy propositions* (FP) for approximate reasoning: A fuzzy proposition has the form "*x is L*," where x is an individual of a universe of discourse U and L is a *linguistic term*, defined as a fuzzy set on U. As stated by Formula 17, the truth-value τ^{FL} of a fuzzy proposition is defined by the degree of membership μ_L of x in the linguistic term L.

$$\tau^{FP}(x\ is\ L) := \mu_L(x) \qquad \text{Formula 17}$$

If A is an attribute of x, a fuzzy proposition can also refer to the corresponding attribute value, such as $x\ is\ L := A(x)\ is\ L$. The fuzzy set L on U is equivalent to the fuzzy set L on the domain of the attribute, or dom(A). In fact, the set can be defined on arbitrarily deep-nested attribute hierarchies concerning the individual. As an example, let us look at the fuzzy proposition, "Mary is blond." In this sentence, the linguistic term "blond" is a fuzzy set on the set of people, which is equivalent to a fuzzy set *blond* on the color of people's hair (Formula 18).

$$\tau^{FP}(\text{"Mary is blond"}) = \mu_{blond}(\text{Mary})$$
$$\equiv \mu_{blond}\big(\text{color}(\text{hair}(\text{Mary}))\big) \qquad \text{Formula 18}$$

Fuzzy propositions (FP) can be combined to construct fuzzy formulae using the usual logic operators *not* ($\neg$), *and* ($\wedge$), and *or* ($\vee$), for which the semantics are defined by the fuzzy truth function $\tau^{FP}: \mathcal{F} \rightarrow [0,1]$, mapping from the class of fuzzy propositions $\mathcal{F}$ into the set of Zadehan truth values in the interval between 1 and 0. Let "*x is P*" and "*x is Q*" be two fuzzy propositions on the same individual. Then their combination to fuzzy formulae is defined as follows (Formula 19 through Formula 21): negation by the inverse of the corresponding fuzzy set, conjunction by intersection of the corresponding fuzzy sets, and disjunction by union of the corresponding fuzzy sets.

$$\tau^{FP}\big(\neg(x\ is\ P)\big) := \mu_{\bar{P}}(x); \qquad \text{Formula 19}$$

$$\tau^{FP}(x\ is\ P\ \wedge\ x\ is\ Q) := \mu_{P\cap Q}(x); \qquad \text{Formula 20}$$

$$\tau^{FP}(x\ is\ P\ \vee\ x\ is\ Q) := \mu_{P\cup Q}(x) \qquad \text{Formula 21}$$

Zadeh's fuzzy propositions are derived from statements of the form "X is Y." They are based on the representation operator $is : U \times U^{\tilde{*}} \to \mathcal{F}$ mapping from the universe U of discourse and its fuzzy powerset $U^{\tilde{*}}$ to the class of fuzzy propositions $\mathcal{F}$. Consequently, fuzzy propositions in the sense of Zadeh are limited to statements about degrees of membership in a fuzzy set.

Generally, logic with fuzzy propositions—or more precisely, a propositional logic with Zadehan truth values in the interval between 0 and 1, a *"Zadehan Logic"* (*ZL*)—can be viewed as a generalization of Boole's mathematical analysis of logic to a gradual concept of truth. In that sense, *ZL* is a simple generalization of Boolean logic (*BL*), in which the truth value of any proposition is not only represented by numbers, but also can be anywhere in the interval between 0 and 1.

According to the *Stanford Encyclopedia of Philosophy* (Hajek, 2006), fuzzy logic, in the narrow sense, is a "symbolic logic with a comparative notion of truth developed fully in the spirit of classical logic" ("Fuzzy Logic," paragraph 3). If *ZL* is viewed as a generalization of *BL*, fuzzy propositions of the form *"X is Y"* are a special case, and propositions and propositional functions of any form can have gradual values of truth. Accordingly, *ZL* is defined by the truth function $\tau^{ZL} : \mathcal{P} \to \mathcal{T}^{ZL}$ mapping from the class of propositions $\mathcal{P}$ to the set of Zadehan truth-values $\mathcal{T}^{ZL} := [0,1]$. Consequently, fuzzy set membership is a special case of fuzzy proposition, and the degree of membership of individual x in another individual y can be defined as the value of truth of the fuzzy proposition $x \in y$ (Formula 22).

$$\mu_y(x) := \tau^{ZL}(x \in y) \qquad \text{Formula 22}$$

The Zadehan truth function τ^{ZL} defines the semantics of *ZL*. As in Boolean algebra, its operators can be defined by subtraction from 1 as negation, and multiplication as conjunction, as formalized in Formula 23 and Formula 24. Disjunction, implication, and equivalence can be derived from negation and conjunction in the same way as in Boolean logic.

$$\tau^{ZL}(\neg\, p) := 1 - \tau^{ZL}(p) \qquad \text{Formula 23}$$

$$\tau^{ZL}(p \wedge q) := \tau^{ZL}(p) \cdot \tau^{ZL}(q) \qquad \text{Formula 24}$$

In that light, any proposition with an uncertain truth-value smaller than 1 or greater than 0 is a fuzzy proposition. Additionally, every function with the range [0,1] can be thought of as a truth function for a propositional function. For example, statistical likelihood $L(y|x)$ can be seen as a truth function for the propositional function, "y is likely if x," as a function of x. This idea is the basis for IFC proposed in the next section. The usefulness of this generalization is shown in the chapter on applications, in which fuzzy propositions such as "customers with characteristic X are likely to buy product Y" are assigned truth-values that are computed using quantitative prediction modeling.

2.2.2   Fuzzy Classification

A *fuzzy class,* $\widetilde{C} := \sim \{\, i \in U \mid \widetilde{\Pi}(i) \,\}$, is defined as a fuzzy set $\tilde{C}$ of individuals i, whose membership degree is defined by the Zadehan truth-value of the proposition $\widetilde{\Pi}(i)$. The classification predicate, $\widetilde{\Pi}$, is a propositional function interpreted in *ZL.* The domain of the *fuzzy class operator,* $\sim \{\,.\mid.\,\} : \mathbb{P} \longrightarrow U^{\tilde{*}}$, is the class of propositional functions, $\mathbb{P}$, and the range is the fuzzy power set, $U^{\tilde{*}}$ (the set of fuzzy subsets) of the universe of discourse, U. In other words, the fuzzy class operator assigns fuzzy subsets of the universe of discourse to propositional functions.

Fuzzy classification is the process of assigning individuals a membership degree to a fuzzy set, based on their degrees of truth of the classification predicate. It has been discussed, for example, by Zimmermann (1997), Del Amo et al. (1999), and Meier et al. (2008). A fuzzy classification is achieved by a membership function, $\mu_{\tilde{C}} : U \longrightarrow [0,1]$, that indicates the degree to which an individual is a member of a fuzzy class, $\tilde{C}$, given the corresponding fuzzy propositional function, $\widetilde{\Pi}$. This membership degree is defined by the Zadehan truth-value of the corresponding proposition, $\widetilde{\Pi}(i)$, as formalized in Formula 25.

$$\mu_{\tilde{C}}(i) := \tau^{ZL}\left(\widetilde{\Pi}(i)\right) \qquad\qquad \text{Formula 25}$$

In the same way as in crisp classification, the fuzzy classification predicate refers to attributes of individuals. Additionally, Zadehan logic introduces two new types of characteristics. *Zadehan attributes* have a range of truth values represented by $\mathcal{T}^{ZL} := [0,1]$. Linguistic attributes have a range of linguistic terms (fuzzy sets) together with the Zadehan truth-value of membership in those terms (Zadeh, 1975b).

In a *univariate fuzzy classification* (*UF*), the fuzzy classification predicate $\widetilde{\Pi}$ refers to one attribute, X, and it corresponds to the membership degree of the attribute characteristic $X(i)$ in a given fuzzy

restriction (Zadeh, 1975a), $R \in \chi^{\tilde{*}}$, which is a fuzzy subset of possible characteristics χ (Formula 26).

$$\mu_{UF}(i) \coloneqq \tau^{ZL}(X(i) \ is \ R) \qquad\qquad \text{Formula 26}$$

In a *multivariate fuzzy classification (MVF)*, $\tilde{\Pi}$ refers to multiple attributes. The truth function of the classification predicate for an individual, i, equals to an aggregation, a, of several fuzzy restrictions of multiple attribute characteristics, $X_j(i)$, $j = 1 \dots n$ (Formula 27).

$$\mu_{MVF}(i) \coloneqq a\big(\tau^{ZL}(X_1(i) \ is \ R_1, \dots, X_n(i) \ is \ R_n)\big) \qquad \text{Formula 27}$$

In a *multidimensional fuzzy classification (MDF)*, $\tilde{\Pi}$ refers to n-tuples of functionally independent attributes. The membership degree of individuals in a multidimensional class is based on an n-dimensional fuzzy restriction, R^n (Formula 28), which is a multidimensional fuzzy set on the Cartesian product of the attribute ranges with a multidimensional membership function of $\mu_{MDF}: range(X_1) \times \dots \times range(X_n) \rightarrow [0,1]$.

$$\mu_{MDF}(i) \coloneqq \tau^{ZL}\left(\begin{bmatrix} X_1(i) \\ \vdots \\ X_n(i) \end{bmatrix} is \ R^n\right) \qquad \text{Formula 28}$$

2.3 Induction

Given a set of certainly true statements, deduction works fine. The problem is that the only certainty philosophy can offer is Descartes's "I think therefore I am" proposition; however, postmodern philosophers are not so sure about the *I* anymore (Precht, 2007, p. 62 ff). Therefore, one should be given a tool to reason under uncertainty, and this tool is induction. In this chapter, inductive logic is analyzed, the application of induction to fuzzy classification is discussed, and a methodology for membership function induction using normalized ratios and differences of empirical conditional probabilities and likelihoods is proposed.

2.3.1 Inductive Logic

Traditionally, induction is defined as drawing general conclusions from particular observations. Contemporary philosophy has shifted to a different view because, not only are there inductions that lead to *particular* conclusions, but also there are deductions that lead to *general* conclusions. According to Vickers (2009) in the *Stanford Encyclopedia of Philosophy* (SEP), it is agreed that induction is a form of inference that is "contingent" and "ampliative" ("The contemporary notion of induction", paragraph 3), in contrast to deductive inference, which is necessary and explicative. Induction is contingent, because inductively inferred propositions are not necessarily true in all cases. And it is ampliative because, in Vickers words, "induction can amplify and generalize our experience, broaden and deepen our empirical knowledge" ("The contemporary notion of induction", paragraph 3). In another essay in the SEP, inductive logic is defined as "a system of evidential support that extends deductive logic to less-than-certain inferences" (Hawthorne, 2008, "Inductive Logic," paragraph 1). Hawthorne admits that there is a degree of fuzziness in induction: In an inductive inference, "the premises should *provide some degree of support* for the conclusion" ("Inductive Logic," para. 1). The degree of support for an inductive inference can thus be viewed as a fuzzy restriction of possible inferences, in the sense of Zadeh (1975a). Vickers (2009) explains that the problem of induction is two-fold: The epistemic problem is to define a method to distinguish appropriate from inappropriate inductive inference. The metaphysical problem is to explain in what substance the difference between reliable and unreliable induction actually exists.

Epistemologically, the question of induction is to find a suitable method to infer propositions under uncertainty. State of the art methods rely on empirical probabilities or likelihoods. There are many interpretations of probability (Hájek, 2009). For the context of this

thesis, one may agree that a mathematical probability, $P(A)$, numerically represents how probable it is that a specific proposition A is true: $P(A) \equiv \tau^{ZL}("A \text{ is probable"})$; and that the disjunction of all possible propositions, the probability space Ω, is certain, i.e., $P(\Omega) = 1$.

In practice, probabilities can be estimated by relative frequencies, or sampled empirical probabilities p in a sample of n observations, defined by the ratio between the number of observations, i, in which the proposition A_i is true, and the total number of observations (Formula 29).

$$P(A) \approx p(A) := \frac{\sum_{i=1}^{n} \tau(A_i)}{n} \qquad \text{Formula 29}$$

A conditional probability (Weisstein, 2010a) is the probability for an outcome x, given that y is the case, as formalized in Formula 30.

$$P(x \mid y) = \frac{P(x \wedge y)}{P(y)} \qquad \text{Formula 30}$$

Empirical sampled conditional probabilities can be applied to compute likelihoods. According to James Joyce, "in an unfortunate, but now unavoidable, choice of terminology, statisticians refer to the inverse probability $P_H(E)$ as the 'likelihood' of H on E" (Joyce, 2003, "Conditional Probabilities and Bayes' Theorem," paragraph 5). The likelihood of the hypothesis H is an estimate of how probable the evidence or known data E is, given that the hypothesis is true. Such a probability is called a "posterior probability" (Hawthorne, 2008, "inductive Logic," paragraph 5), that is, a probability after measurement, shown by Formula 31.

$$L(H \mid E) := p(E \mid H) \qquad \text{Formula 31}$$

In the sense of Hawthorne (2008), the general law of likelihood states that, for a pair of incompatible hypotheses H_1 and H_2, the evidence E supports H_1 over H_2, if and only if $p(E \mid H_1) > p(E \mid H_2)$. The likelihood ratio ($LR$) measures the strength of evidence for H_1 over H_2 (Formula 32). Thus, the "likelihoodist" (sic; Hawthorne, 2008, "Likelihood Ratios, Likelihoodism, and the Law of Likelihood," paragraph 5) solution to the epistemological problem of induction is the likelihood ratio as measure of support for inductive inference.

$$LR(H_1 > H_2 \mid E) := \frac{L(H_2 \mid E)}{L(H_1 \mid E)} \qquad \text{Formula 32}$$

According to Hawthorne, the prior probability of a hypothesis, $p_0(H)$, that is, an estimated probability prior to measurement of evidence E, plays an important role for inductive reasoning. Accordingly, Bayes' theorem can be interpreted and rewritten using measured posterior likelihood and prior probability in order to apply it to the evaluation of scientific hypotheses. According to Hawthorne (2008), the posterior probability of hypothesis H conditional to evidence E is equal to the product of the posterior likelihood of H given E and the prior probability of H, divided by the (measured) probability of E (Formula 33).

$$p(H \mid E) = \frac{L(H \mid E) \cdot p_0(H)}{p(E)} \qquad \text{Formula 33}$$

What if there is fuzziness in the data, in the features of observations, or in the theories? How is likelihood measured when the hypothesis or the evidence is fuzzy? If this fuzziness is ordinal, that is, if the extent of membership in the fuzzy terms can be ordered, a membership function can be defined, and an empirical probability of fuzzy events can be calculated. Analogous to Dubois and Prade (1980), a fuzzy event $\tilde{A}$ in a universe of discourse U is a fuzzy set on U with a membership function $\mu_{\tilde{A}}: U \longrightarrow [0,1]$. For categorical elements of U, the estimated probability after n observations is defined as the average degree of membership of observations i in $\tilde{A}$, as formalized in Formula 34.

$$P(\tilde{A}) \approx p(\tilde{A}) = \frac{\sum_{i=1}^{n} \mu_{\tilde{A}}(i)}{n} \qquad \text{Formula 34}$$

By application of Formula 34 to Formula 31, the likelihood of ordinal fuzzy hypothesis $\tilde{H}$, given ordinal fuzzy evidence $\tilde{E}$, can be defined as a conditional probability of fuzzy events, as shown in Formula 35.

$$L(\tilde{H}|\tilde{E}) = p(\tilde{E}|\tilde{H}) = \frac{\sum_{i=1}^{n} \mu_{\tilde{H} \cap \tilde{E}}(i)}{\sum_{i=1}^{n} \mu_{\tilde{H}}(i)} \qquad \text{Formula 35}$$

The question of the metaphysical problem of induction is: what is the substance of induction? In what kind of material does the difference between reliable and unreliable inductive inference exist? The importance of this question cannot be underestimated, since reliable induction enables prediction. A possible answer could be that the substance of an induction is the amount of information contained in the inference. This answer presupposes that information is a realist category, as suggested by Chmielecki (1998). According to Shannon's information theory (Shannon, 1948), the information contained in evidence x about hypothesis y is equal to the difference between the uncertainty (entropy), $H(y)$, about the hypothesis y and the resulting uncertainty, $H_x(y)$, after observation of the evidence x, $I(x,y) = H(y) - H_x(y) = \Sigma_x \Sigma_y p(x \wedge y) \log_2 p(x \wedge y)/\big(p(x)p(y)\big)$. Shannon's quantity of information is defined in terms of joint probabilities. However, by application of Shannon's theory, the metaphysical problem of induction is transferred to a metaphysical problem of probabilities because, according to Shannon, the basic substance of information is the probability of two signals occurring simultaneously compared to the probability of occurring individually. (One could link this solution to the concept of quantum physical particle probability waves [Greene, 2011], but this would go beyond the scope of this thesis and would be highly speculative; therefore, this link is not explored here. Suffice it to state that probability apparently is a fundamental construct of matter and waves as well as of information and induction.)

2.3.2 Inductive Classification

Inductive classification is the process of assigning individuals to a set based on a classification predicate derived by an inductive inference. Inductive classification can be automated as a form of supervised machine learning (Witten & Frank, 2005): a class of processes (algorithms or heuristics) that learn from examples to decide whether an individual, i, belongs to a given class, y, based on its attributes. Generally, supervised machine learning processes induce a model from a dataset, which generalizes associations in the data in order to provide support for inductive inference. This model can be used for predicting the class membership of new data elements. Induced classification models, called *classifiers*, are first trained using a training set with known class membership. Then, they are applied to a test or prediction set in order to derive class membership predictions. Examples of

classification learning algorithms that result in classifications are decision trees, classification rules, and association rules. In those cases, the model consists of logical formulae of attribute values, which predict a crisp class value.

Data are signs (signals) that represent knowledge such as numbers, characters, or bits. The basis for automated data analysis is a systematic collection of data on individuals. The most frequently used data structure for analytics is the matrix, in which every individual, i (a customer, a transaction, a website, etc.), is represented by a row, and every attribute, X_k, is represented by a column. Every characteristic, $X_k(i)$, of individual i for attribute X_k is represented by one scalar value within the matrix.

A training dataset d is an $m \times (n + 1)$ matrix with m rows, n columns for $X_1, \ldots, X_n$ and a column Y indicating the actual class membership. The columns $X_k, 1 \leq k \leq n$ are called *analytic variables*, and Y is called the *target variable*, which indicates membership in a target class y. In case of a binary classification, for each row index i, the label $Y(i)$ is equal to 1 if and only if individual i is in class y (Formula 36).

$$Y(i) := \begin{cases} 1 & \text{if } i \in y \\ 0 & \text{else.} \end{cases}$$
Formula 36

A machine learning process for inductive sharp classification generates a model $M_y(i)$, mapping from the Cartesian product of the analytic variable ranges into the set {0,1}, indicating inductive support for the hypothesis that $i \in y$. As discussed in the section on induction, the model should provide support for inductive inferences about an individual's class membership: Given $M_y(i) = 1$, the likelihood of $i \in y$ should be greater than the likelihood of $i \notin y$.

The inductive model M_y can be applied for prediction to a new dataset with unknown class indicator, which is either a test set for performance evaluation or a prediction set, where the model is applied to forecast class membership of new data. The test set or prediction set d' has the same structure as the training set d, except that the class membership is unknown, and thus, the target variable Y is empty. The classifier M_y, derived from the training set, can be used for predicting the class memberships of representations of individuals $i \in d$. The model output prediction $M_y(i)$ yields an inductive classification defined by $\{ i \mid M_y(i) = 1 \}$.

In order to evaluate the quality of prediction of a crisp classifier model, several measures can be computed. In this section, likelihood

ratio and Pearson correlation are mentioned. The greater the ratio between likelihood for target class membership, given a positive prediction, and the likelihood for target class membership, given a negative prediction, the better the inductive support of the model. Thus, the predictive model can be evaluated by the likelihood ratio of target class membership given the model output (Formula 37).

$$LR\big(Y(i) = 1 \mid M_y(i) = 1\big) := \frac{p\big(M_y(i) = 1 \mid Y(i) = 1\big)}{p\big(M_y(i) = 1 \mid Y(i) = 0\big)} \qquad \text{Formula 37}$$

Working with binary or Boolean target indicators and model indicators allows the evaluation of predictive quality by a measure of correlation of the two variables M_y and Y (Formula 38). The correlation between two numerical variables can be measured by the Pearson correlation coefficient as the ratio between the covariance of the two variables and the square root of the product of individual variances (Weisstein, 2010b).

$$\text{corr}\big(M_y, Y\big) = \frac{E\left(\big(M_y - \text{avg}(M_y)\big)\big(Y - \text{avg}(Y)\big)\right)}{\text{stddev}\big(M_y\big) \cdot \text{stddev}(Y)} \qquad \text{Formula 38}$$

The advantage of the correlation coefficient is its availability in database systems. Every standard SQL (structured query language) database has an implementation of correlation as an aggregate function. Thus, using the correlation coefficient, evaluating the predictive performance of a model in a database is fast and simple. However, it is important to stress that evaluating predictions with a measure of correlation is only meaningful if the target variable as well as the predictive variable are Boolean, Zadehan, or numeric in nature. It will not work for ordinal or categorical target classes, except if they are transformed into a set of Boolean variables.

For example, in database marketing, the process of target group selection uses classifiers to select customers who are likely to buy a certain product. In order to do this, a classifier model can be computed in the following way: Given set of customers C, we know whether they have bought product A or not. Let c be an individual customer, and C_A be the set of customers who bought product A. Then, the value $Y(c)$ of target variable Y for customer c is defined in Formula 39.

$$Y(c) = \begin{cases} 1 & \text{if } c \in C_A \\ 0 & \text{else.} \end{cases}$$

<div style="text-align: right">Formula 39</div>

The analytic variables for customers are selected from every known customer attribute, such as age, location, transaction behavior, recency, frequency, and monetary value of purchase. The aim of the classifier induction process is to learn a model, M_{C_A}, that provides a degree of support for the inductive inference that a customer is interested in the target product A. This prediction, $M_{C_A}(c) \in \{0,1\}$, should provide a better likelihood to identify potential buyers of product A, and it should optimally correlate with the actual product usage of existing and future customers.

2.4 Inductive Fuzzy Classification

The understanding of IFC in the proposed research approach is an inductive gradation of the degree of membership of individuals in classes In many interpretations, the induction step consists of learning *fuzzy rules* (e.g., Dianhui, Dillon, & Chang, 2001; Hu, Chen, & Tzeng, 2003; Roubos, Setnes, & Abonyi, 2003; Wang & Mendel, 1992). In this thesis, IFC is understood more generally as inducing membership functions to fuzzy classes and assigning individuals to those classes. In general, a membership function can be any function mapping into the interval between 1 and 0. Consequently, IFC is defined as the process of assigning individuals to fuzzy sets for which membership functions are generated from data so that the membership degrees are based on an inductive inference.

An inductive fuzzy class, y', is defined by a predictive scoring model, $M_y: U \to [0,1]$, for membership in a class, y. This model represents an inductive membership function for y', which maps from the universe of discourse U into the interval between 0 and 1 (Formula 40).

$$\mu_{y'}: U \to [0,1] := M_y \qquad \text{Formula 40}$$

Consider the following fuzzy classification predicate $P(i,y) :=$ "i is likely a member of y." This is a fuzzy proposition (Zadeh, 1975a) as a function of i and y, which indicates that there is inductive support for the conclusion that individual i belongs to class y. The truth function, τ^{ZL}, of this fuzzy propositional function can be defined by the membership function of an inductive fuzzy class, y'. Thus, $P(i,y)$ is a fuzzy restriction on U defined by $\mu_{y'}$ (Formula 41).

$$\tau^{ZL}(\text{"}i \text{ is likely a member of } y\text{"}) := M_y(i) \qquad \text{Formula 41}$$

In practice, any function that assigns values between 0 and 1 to data records can be used as a fuzzy restriction. The aim of IFC is to calculate a membership function to a fuzzy set of likely members in the target class. Hence, any type of classifier with a normalized numeric output can be viewed as an inductive membership function to the target class, or as a truth function for the fuzzy proposition $P(i,y)$. State of the art methods for IFC in that sense include linear regression, logistic regression, naïve Bayesian classification, neural networks, fuzzy classification trees, and fuzzy rules. These are classification methods

yielding numerical predictions that can be normalized in order to serve as a membership function to the inductive fuzzy class y' (Formula 42).

$$y' := \{i \in U \mid i \text{ is likely a member of } y \} \qquad \text{Formula 42}$$

2.4.1 Univariate Membership Function Induction

This section describes methods to derive membership functions for one variable based on inductive methods. First, unsupervised methods are described, which do not require learning from a target class indicator. Second, supervised methods for predictive membership functions are proposed.

Numerical attributes can be fuzzified in an unsupervised way, that is, without a target variable, by calculating a membership function to a fuzzy class *x is a large number*, denoted by the symbol ↑: the fuzzy set of attribute values that are large relative to the available data. This membership function, $\mu_\uparrow: \text{dom}(C) \rightarrow [0,1]$, maps from the attribute domain of the target variable into the set of Zadehan truth values. This unsupervised fuzzification serves two purposes. First, it can be used to automatically derive linguistic interpretations of numerical data, such as "large" or "small." Second, it can be used to transform numerical attributes into Zadehan target variables in order to calculate likelihoods of fuzzy events. There are two approaches proposed here to compute a membership function to this class: percentile ranks and linear normalization based on minimum and maximum.

For a numeric or ordinal variable X with a value $x \in \text{dom}(X)$, the percentile rank (PR) is equal to the sampled probability that the value of the variable X is smaller than x. This sampled probability is calculated by the percentage of values in $\text{dom}(X)$ that are smaller than or equal to x. This sampled probability can be transformed into a degree of membership in a fuzzy set. Inductively, the sampled probability is taken as an indicator for the support of the inductive inference that a certain value, X_i, is large in comparison to the distribution of the other attribute values. The membership degree of x in the fuzzy class *"relatively large number"*, symbolized by ↑, is then defined as specified in Formula 43.

$$\mu_{\uparrow PR}(x) := p(X < x) \qquad \text{Formula 43}$$

For example, customers can be classified by their profitability. The percentile rank of profitability can be viewed as a membership function

of customers in the fuzzy set ↑ of customers with a high profitability. Figure 9 shows an example of an IFC-PR of customer profitability for a financial service provider.
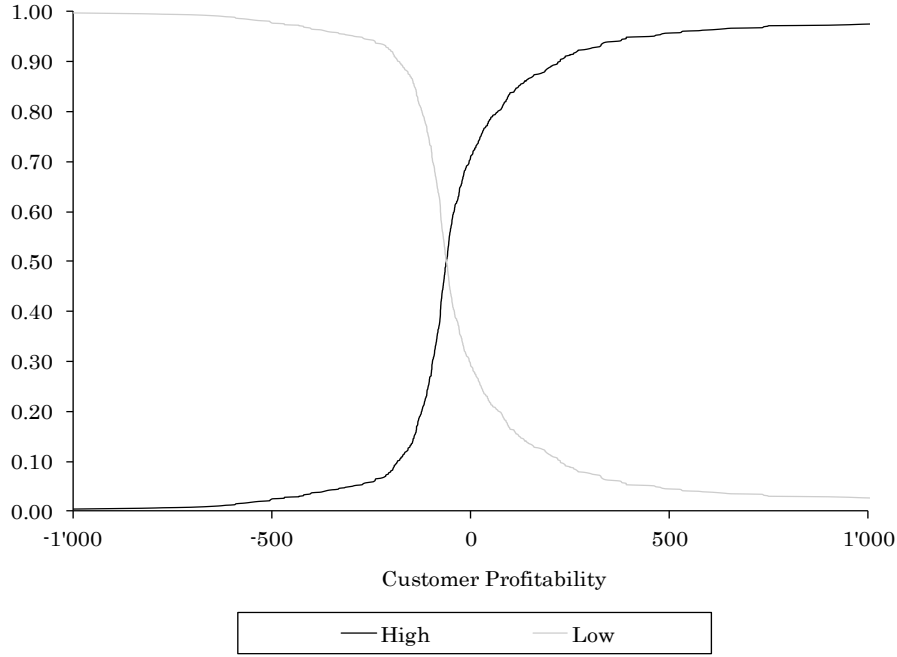
Figure 9: Unsupervised IFC by percentile rank.

Adapted from "An Inductive Approach to Fuzzy Marketing Analytics," by M. Kaufmann, 2009, In M. H. Hamza (Ed.), Proceedings of the 13th IASTED International Conference on Artificial Intelligence and Soft Computing: Copyright 2009 by Publisher.

A simpler variant of unsupervised fuzzification for generating a membership function for a *relatively large number* (↑) is *linear normalization* (IFC-LN). For a numerical attribute C, it is defined as the relative distance to the minimal attribute value, as specified in Formula 44.

$$\mu_\uparrow\big(C(i)\big) := \frac{C(i) - \min(C)}{\max(C) - \min(C)} \qquad \text{Formula 44}$$

For the *membership function induction* (MFI) methods in the following sections, the target variable for supervised induction must be a Zadehan variable, $Y: U \rightarrow [0,1]$, mapping from the universe of discourse (the set of possible individuals) into the interval of Zadehan

truth values between 0 and 1. Thus, $Y(i)$ indicates the degree of membership of individual i in the target class y. In the special case of a Boolean target class, $Y(i)$ is equal to 1 if $i \in y$, and it is equal to 0 if $i \notin y$. In the analytic training data, a target class indicator Y can be deduced from data attributes in the following way:

o If an attribute, A, is Zadehan with a range between 0 and 1, it can be defined directly as the target variable. In fact, if the variable is Boolean, this implies that it is also Zadehan, because it is a special case (Formula 45).

$$\text{Zadehan}(A) \Longrightarrow \mu_y(i) := A(i)$$

<div align="right">Formula 45</div>

o If an attribute, B, is categorical with a range of n categories, it can be transformed into n Boolean variables μ_{y^k} ($k = 1, 2, \dots, n$), where $\mu_{y^k}(i)$ indicates whether record i belongs to class k, as specified by Formula 46.

$$\text{categorical}(B) \Longrightarrow \mu_{y^k}(i) := \begin{cases} 1 & \text{if } B(i) = k \\ 0 & \text{else.} \end{cases}$$

<div align="right">Formula 46</div>

o If an attribute, C, is numeric, this thesis proposes application of an unsupervised fuzzification, as previously specified, in order to derive a Zadehan target variable, as formalized in Formula 47. This is called an *inductive target fuzzification* (ITF).

$$\text{numerical}(C) \Longrightarrow \mu_y(i) := \mu_\uparrow\big(C(i)\big)$$

<div align="right">Formula 47</div>

The second approach for univariate membership function induction is supervised induction based on a target variable. In order to derive membership functions to inductive fuzzy classes for one variable based on the distribution of a second variable, it is proposed to normalize comparisons (ratios and differences) of likelihoods for membership function induction. For example, a normalized likelihood ratio can represent a membership degree to an inductive fuzzy class.

The basic idea of *inductive fuzzy classification based on normalized likelihood ratios* (IFC-NLR) is to transform inductive support of target class membership into a membership function with the following properties: The higher the likelihood of $i \in y$ in relation to $i \notin y$, the greater the degree membership of i in y'. For an attribute X, the NLR function calculates a membership degree of a value $x \in \text{dom}(X)$ in the predictive class y', based on the likelihood of target class membership. The resulting membership function is defined as a relation between all values in the domain of the attribute X and their NLRs.

As discussed in Section 2.3.1, following the principle of likelihood (Hawthorne, 2008), the ratio between the two likelihoods is an indicator for the degree of support for the inductive conclusion that $i \in y$, given the evidence that $X(i) = x$. In order to transform the likelihood ratio into a fuzzy set membership function, it can be normalized in the interval between 0 and 1. Luckily, for every ratio, R = A/B, there exists a normalization, $N = A/(A+B)$, having the following properties:

o N is close to 0 if R is close to 0.

o N is equal to 0.5 if and only if R is equal to 1.

o N is close to 1 if R is a large number.

This kind of normalization is applied to the aforementioned likelihood ratio in order to derive the NLR function. Accordingly, the membership μ of an attribute value x in the target class prediction y' is defined by the corresponding NLR, as formalized in Formula 48.

$$\mu_{y'}(x) := NLR(y \mid x) = \frac{L(y \mid x)}{L(y \mid x) + L(\neg y \mid x)}$$

Formula 48

In fact, one can demonstrate that the NLR function is equal to the posterior probability of y, conditional to x, if both hypotheses y and $\neg y$ are assumed to be of equal prior probability (Formula 52), by application of the second form of Bayes' theorem (Joyce, 2003), as presented in Formula 50. The trick is to express the probability of the evidence $p(x)$ in terms of a sum of products of prior probabilities, p_0, and measured likelihoods, L, of the hypothesis and its alternative by application of Formula 33.

Theorem.

$$NLR(y \mid x) = p(y \mid x) \Leftrightarrow p_0(y) = p_0(\neg y) \qquad \text{Formula 49}$$

Proof.

$$p(y \mid x) = \frac{p_0(y)L(y \mid x)}{p(x)} \qquad \text{(c.f. Formula 36)}$$

$$= \frac{p_0(y)L(y \mid x)}{p_0(y)L(y \mid x) + p_0(\neg y)L(\neg y \mid x)}$$

$[if \quad p(x) = p(y)p(x \mid y) + p(\neg y)p(x \mid \neg y) \qquad \text{Formula 50}$

$and \quad p(y) := p_0(y) \ and \ p(x \mid y) := L(y \mid x)]$

$$= \frac{L(y \mid x)}{L(y \mid x) + L(\neg y \mid x)} \quad [if \ p_0(y) :=: p_0(\neg y)]$$

$=: NLR(y \mid x), \quad q.e.d.$

Alternatively, two likelihoods can be compared by a normalized difference, as shown in Formula 51. In that case, the membership function is defined by a *normalized likelihood difference* (NLD), and its application for classification is called *inductive fuzzy classification by normalized likelihood difference* (IFC-NLD). In general, IFC methods based on *normalized likelihood comparison* can be categorized by the abbreviation IFC-NLC.

$$\mu_{y'}(x) := NLD(y \mid x) = \frac{L(y \mid x) - L(\neg y \mid x) + 1}{2} \qquad \text{Formula 51}$$

If a target attribute is continuous, it can be mapped into the Zadehan domain of numeric truth-values between 0 and 1, and membership degrees can be computed by a normalized ratio of likelihoods of fuzzy events. If the target class is fuzzy, for example because the target variable is gradual, the likelihoods are calculated by fuzzy conditional relative frequencies based on fuzzy set cardinality (Dubois & Prade, 1980). Therefore, the formula for calculating the likelihoods is generalized in order to be suitable for both sharp and fuzzy characteristics. Thus, in the general case of variables with fuzzy truth-values, the likelihoods are calculated as defined in Formula 52.

$$L(y \mid x) := \frac{\sum_{i=1}^{n} \mu_x(i)\mu_y(i)}{\sum_{i=1}^{n} \mu_y(i)}$$

<div align="right">Formula 52</div>

$$L(\neg y \mid x) := \frac{\sum_{i=1}^{n} \mu_x(i)(1 - \mu_y(i))}{\sum_{i=1}^{n} (1 - \mu_y(i))}$$

Accordingly, the calculation of membership degrees using the NLR function (Formula 52) works for both categorical and fuzzy target classes and for categorical and fuzzy analytic variables. For numerical attributes, the attribute values can be discretized using quantiles, and a piecewise linear function can be approximated to average values in the quantiles and the corresponding NLR. A membership function for individuals based on their attribute values can be derived by aggregation, as explained in Section 2.4.2.

Following the different comparison methods for conditional probabilities described by Joyce (2003), different methods for the induction of membership degrees using conditional probabilities are proposed in Table 1. They have been chosen in order to analytically test different Bayesian approaches listed by Joyce (2003) for their predictive capabilities. Additionally, three experimental measures were considered: logical equivalence, normalized correlation, and a measure based on minimum and maximum. In those formulae, x and y are assumed to be Zadehan with a domain of [0,1] or Boolean as a special case. These formulae are evaluated as parameters in the meta-induction experiment described in Section 4.2.

<div align="center">
Table 1

Proposed Formulae for Induction of Membership Degrees
</div>

Method	Formula
Likelihood of y given x (L)	$L(y\|x) = p(x\|y)$
Normalized likelihood ratio (NLR)	$NLR(y\|x) = \dfrac{p(x\|y)}{p(x\|y) + p(x\|\neg y)}$
Normalized likelihood ratio unconditional (NLRU)	$NLRU(y\|x) = \dfrac{p(x\|y)}{p(x\|y) + p(x)}$
Normalized likelihood difference (NLD)	$NLD(y\|x) = \dfrac{p(x\|y) - p(x\|\neg y) + 1}{2}$
Normalized likelihood difference unconditional (NLDU)	$NLDU(y\|x) = \dfrac{p(x\|y) - p(x) + 1}{2}$

Conditional probability of y given x (CP)	$p(y\|x)$
Normalized probability ratio (NPR)	$NPR(y\|x) = \dfrac{p(y\|x)}{p(y\|x) + p(y\|\neg x)}$
Normalized probability ratio unconditional (NPRU)	$NPRU(y\|x) = \dfrac{p(y\|x)}{p(y\|x) + p(y)}$
Normalized probability difference (NPD)	$NPD(y\|x) = \dfrac{p(y\|x) - p(y\|\neg x) + 1}{2}$
Normalized probability difference unconditional (NPDU)	$NPDU(y\|x) = \dfrac{p(y\|x) - p(y) + 1}{2}$
Equlivalence - if and only if (IFF)	$\mathrm{avg}\left(\left(1 - \mathrm{x} \cdot (1 - \mathrm{y})\right) \cdot \left(1 - \mathrm{y} \cdot (1 - \mathrm{x})\right) \right)$
Minimum – maximum (MM)	$\dfrac{p(y\|x) + \min_{z \in \mathrm{dom}(X)}\left(p(y\|z)\right)}{\min_{z \in \mathrm{dom}(X)}\left(p(y\|z)\right) + \max_{z \in \mathrm{dom}(X)}\left(p(y\|z)\right)}$
Normalized correlation (NC)	$\dfrac{\mathrm{corr}(\mathrm{x}, y) + 1}{2}$

A method for discretization of a numerical range is the calculation of quantiles or n-tiles for the range of the analytical variable. A quantile discretization using n-tiles partitions the variable range into n intervals having the same number of individuals. The quantile $Q_n^Z(i)$ for an attribute value $Z(i)$, of a numeric attribute Z and an individual i, is calculated using Formula 53, where n is the number of quantiles, m is the total number of individuals or data records, $rank_Z(i)$ is the position of the individual in the list of individuals sorted by their values of attribute Z, and $trunc(r)$ is the closest integer that is smaller than the real value r.

$$Q_n^Z(i) := trunc\left(\frac{n}{m}(rank_Z(i) - 1) \right) \qquad \text{Formula 53}$$

The rank of individual i relative to attribute Z, $rank_Z(i)$, in a dataset S is the number of other individuals, h, that have higher values in attribute Z, calculated using Formula 54.

$$rank_Z(i) := |\{h \in S \mid \forall i \in S : Z(h) > Z(i)\}| \qquad \text{Formula 54}$$

In order to approximate a linear function, the method of two-dimensional *piecewise linear function approximation* (PLFA) is proposed. For a list of points in $\mathbb{R}^2$, ordered by the first coordinate, $(\langle x_1, y_1\rangle, \langle x_2, y_2\rangle, \dots, \langle x_n, y_n\rangle)$, for every point $\langle x_i, y_i\rangle$ except the last one

$(i = 1,2, \dots, n-1)$, a linear function, $f_i(x) = a_i x + b_i$, can be interpolated to its neighbor point, where a_i is the slope (Formula 55) and b_i is the intercept (Formula 56) of the straight line.

$$a_i := \frac{(y_{i+1} - y_i)}{(x_{i+1} - x_i)} \qquad \text{Formula 55}$$

$$b_i := y_i - a_i x_i \qquad \text{Formula 56}$$

For the calculation of membership degrees for quantiles, the input is a list of points with one point for every quantile k. The first coordinate is the average of the attribute values in k. The second coordinate is the inductive degree of membership $\mu_{y'}$ in target class y, given $Z(i)$ is in quantile k, for example derived using the NLR function.

$$
\begin{aligned}
y_k &:= \mu_{y'}(k) \\
x_k &:= avg\{Z(i) \mid Q_n^Z(i) = k\}
\end{aligned}
\qquad \text{Formula 57}
$$

Finally, a continuous, piecewise affine membership function can be calculated, truncated below 0 and above 1, and is composed of straight lines for every quantile $k = 1, \dots, n-1; n \geq 2$ of the numeric variable Z (Formula 58).

$$
\mu_y(x) :=
\begin{cases}
0 & \mid a_1 x + b_1 \leq 0 \vee a_{n-1}x + b_{n-1} \leq 0 \\
a_1 x + b_1 & \mid x \leq x_2 \\
\vdots & \vdots \\
a_k x + b_k & \mid x_k < x \leq x_{k+1} \\
\vdots & \vdots \\
a_{n-1}x + b_{n-1} & \mid x > x_{n-1} \\
1 & \mid a_1 x + b_1 \geq 1 \vee a_{n-1}x + b_{n-1} \geq 1
\end{cases}
\qquad \text{Formula 58}
$$

The number of quantiles can be optimized, so that the correlation of the membership function with the target variable is optimal, as illustrated in Figure 10.
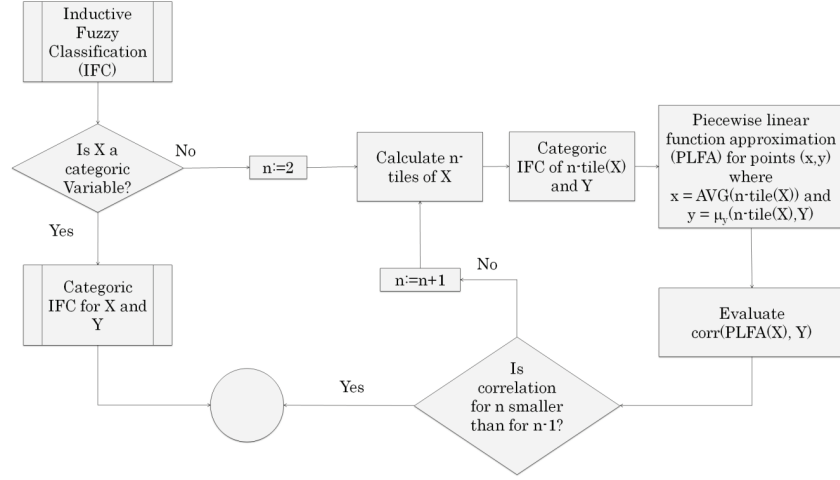
Figure 10: Computation of membership functions for numerical variables.

2.4.2 Multivariate Membership Function Induction

As shown in Figure 11, the proposed process for inducing a multivariate inductive fuzzy class consists of preparing the data, inducing univariate membership functions for the attributes, transforming the attribute values into univariate target membership degrees, classifying individuals by aggregating the fuzzified attributes into a multivariate fuzzy classification, and evaluating the predictive performance of the resulting model.
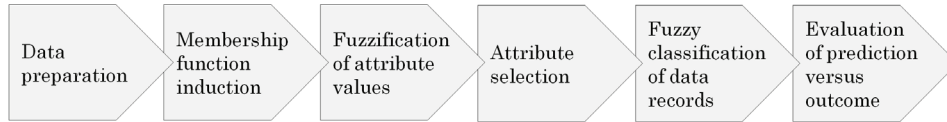
Figure 11: Proposed method for multivariate IFC.

The idea of the process is to develop a fuzzy classification that ranks the inductive membership of individuals, i, in the target class y gradually. This fuzzy classification will assign individuals an inductive membership degree to the predictive inductive fuzzy class y' using the multivariate model $\mu_{y'}$. The higher the inductive degree of membership $\mu_{y'}(i)$ of an individual in y', the greater the degree of inductive support for class membership in the target class y.

In order to accomplish this, a training set is prepared from source data, and the relevant attributes are selected using an interestingness measure. Then, for every attribute X_k, a membership function, $\mu_{y'}^k: \mathrm{dom}(X_k) \rightarrow [0,1]$, is defined. Each $\mu_{y'}^k$ is induced from the data such that the degree of membership of an attribute value $X_k(i)$ in the inductive fuzzy class y' is proportional to the degree of support for the inference that $i \in y$. After that, in the univariate classification step, each variable, X_k, is fuzzified using $\mu_{y'}^k$. The multivariate fuzzy classification step consists of aggregating the fuzzified attributes into one multivariate model, $\mu_{y'}$, of data elements that represents the membership function of individual i in y'. This inductive fuzzy class corresponds to an IFC that can be used for predictive ranking of data elements. The last step of the process is model evaluation through analyzing the prediction performance of the ranking. Comparing the forecasts with the real class memberships in a test set does this. In the following paragraphs, every step of the IFC process is described in detail.

In order to analyze the data, combining data from various sources into a single coherent matrix composes a training set and a test set. All possibly relevant attributes are merged into one table structure. The class label Y for the target variable has to be defined, calculated, and added to the dataset. The class label is restricted to the Zadehan domain, as defined in the previous section. For multiclass predictions, the proposed process can be applied iteratively.

Intuitively, the aim is to assign to every individual a membership degree in the inductive fuzzy class y'. As explained in Section 2.3.1, this degree indicates support for the inference that an individual is a member of the target class y. The membership function for y' will be derived as an aggregation of inductively fuzzified attributes. In order to accomplish this, for each attribute, a univariate membership function in the target class is computed, as described in the previous section.

Once the membership functions have been induced, the attributes can be fuzzified by application of the membership function to the actual attribute values. In order to do so, each variable, X_k, is transformed into an inductive degree of membership in the target class. The process of mapping analytic variables into the interval $[0, 1]$ is an attribute fuzzification. The resulting values can be considered a membership degree to a fuzzy set. If this membership function indicates a degree of support for an inductive inference, it is called an *inductive attribute fuzzification* (IAF), and this transformation is denoted by the symbol $\rightsquigarrow$ in Formula 59.

$$X_k(\mathrm{i}) \rightsquigarrow \mu_{\mathrm{y}'}\big(X_k(\mathrm{i}) \big) \qquad\qquad \text{Formula 59}$$

The most relevant attributes are selected before the IFC core process takes place. The proposed method for attribute selection is a ranking of the Pearson correlation coefficients (Formula 38) between the inductively fuzzified analytic variables and the (Zadehan) target class indicator Y. Thus, for every attribute, X_k, the relevance regarding target y is defined as the correlation of its inductive fuzzification with the target variable (see Section 3.1.1).

In order to obtain a multivariate membership function for individuals i derived from their fuzzified attribute values $\mu_{y'}(X_k(i))$, their attribute value membership degrees are aggregated. This corresponds to a multivariate fuzzy classification of individuals. Consequently, the individual's multivariate membership function $\mu_{y'}: U \rightarrow [0,1]$ to the inductive fuzzy target class y' is defined as an aggregation, $aggr$, of the membership degrees of n attributes, $X_k, k = 1, 2, \ldots, n$ (Formula 60).

$$\mu_{y'}(i) := aggr\left(\mu_{y'}(X_1(i)), \ldots, \ \mu_{y'}(X_n(i))\right) \qquad \text{Formula 60}$$

By combining the inductively fuzzified attributes into a multivariate fuzzy class of individuals, a multivariate predictive model, $\mu_{y'}$, is obtained from the training set. This corresponds to a classification of individuals by the fuzzy proposition "i is likely a member of y," for which the truth value is defined by an aggregation of the truth values of fuzzy propositions about the individual's attributes, $X_k(i)$ is y'. This model can be used for IFC of unlabeled data for predictive ranking. Applying an alpha cutoff, $\{i \mid \mu_{y'}(i) \geq \alpha\}$, an $\alpha \in [0,1]$ leads to a binary classifier.

There are different possibilities for calculating the aggregation, $aggr$. Simpler methods use an average of the attribute membership degrees, logical conjunction (minimum, algebraic product), or logical disjunction (maximum, algebraic sum). More sophisticated methods involve the supervised calculation of a multivariate model. In this thesis, normalized or cutoff linear regression, logistic regression, and regression trees are considered. These different aggregation methods were tested as a parameter in the meta-induction experiment described in Section 4.2 in order to find an optimal configuration.

Finally, in order to evaluate predictive performance, the classifier is applied to a hold-out test set, and the predictions $\mu_{y'}(i)$ are compared with the actual target variable (i). The correlation between the prediction and the target, $corr(\mu_{y'}, Y)$, can be used to compare the performance of different IFC models.

3 Analytics & Marketing

3.1 Analytics

Analytics is "the method of logical data analysis" (merriam-webster.com, 2012a). According to Zimmermann (1997), data analysis is the "search for structure in data". The more data is available, the more complex it becomes to find relevant information. Consequently, organizations and individuals analyze their data in order to gain useful insights. *Business analytics* is defined as "a broad category of applications and techniques for gathering, storing, analyzing and providing access to data to help enterprise users make better business and strategic decisions" (Turban, Aronson, Liang, & Sharda, 2007, p. 256).

The ability of enterprises to analyze the potentially infinite space of available data—their capacity of business analytics—is a major competitive advantage. Companies that use analytics as key strategies are called "analytics competitors" by Davenport (2006, p. 3). They can differentiate themselves through a better customer understanding in a time when products and technologies are becoming more and more comparable. Analytics competitors apply predictive modeling to a wide range of fields, such as customer relationship management, supply chain management, pricing, and marketing. Their top management understands and advocates that most business functions can benefit from quantitative optimization. In fact, business analytics can be applied to almost any area that concerns an enterprise:

o Customer relationship management (CRM): Prediction of the most appropriate customer relationship activity.

o Web analytics: Optimization of the website according to click stream analysis.

o Compliance: Prediction of illegal behavior such as fraud or money laundering.

o Risk management: Prediction of credit worthiness.

o Strategic management: Visualization of customer profiles for product or market strategies.

o Marketing: Prediction of customer product affinity.

Kohavi, Rothleder, and Simoudis (2002) identified comprehensibility and integration as the driving forces of emerging trends in business analytics. Today, good business analytics is either comprehensible *or* integrated: Human decision makers need

interpretable, visual, or textual models in order to understand and apply analytic insights in their daily business. Contrarily, information systems demand machine readable, integrated automated interfaces from analytic applications to operational systems in order to apply scorings or classifications in automated processes.

In order to enhance the data basis for analytics, large data pools, such as data warehouses, have been built. Today, the problem is no longer the lack of data, but the abundance of it. *Data mining* (in the broader sense) is "the process of discovering patterns in data". In a more narrow sense, those "structural patterns" help not only to explain the data but also to make "nontrivial predictions on new data" (Witten & Frank, 2005, p. 5). The common aspect of the two definitions is the discovery of patterns in data. In contrast, the second definition emphasizes the inductive aspect of data mining, namely prediction. Data mining, in its broader sense, means any form of discovery of knowledge in data, whereas in the narrower sense, it means model induction for prediction using statistics and machine learning. One aim of data mining is the induction of models for datasets. If applied correctly, models not only describe the past, but also help to forecast the future. Successful data mining is a practical demonstration of inductive logic. There are two categories of machine learning: non-supervised and supervised learning. The former extracts models such as clusters or association rules from data without labels. The latter induces a model for labeled data, representing the relationship between the data and the label. Continuous labels lead to regression models, and categorical ones lead to classification models or classifiers.

Descriptive data analysis presents data as is, that is, it describes the data without generalization or conclusion. It is completely deductive, because the numbers are true without need for inductive support. In a business context, this is often called reporting. A report is an understandable tabular or graphical representation of relevant data. Usually, methods of classification and aggregation are applied to consolidate data into meaningful information. Frequently applied techniques for descriptive data analysis are deductive classification, aggregation, and grouping. For deductive classification, data records are classified according to an a priori predicate. Therefore, the class membership is known in advance. Aggregation is the calculation of a scalar value from a vector or set, such as sums, averages, or extrema. Often, these aggregations are grouped by one or more variables. In such cases, the aggregated values are calculated for each combination of values in the grouping variables For example, the average cost and benefit per customer segment could be an interesting data description for many companies. An important part of descriptive data analysis is the presentation of the results. The raw output of an analysis is usually

a matrix or table, but human decision makers are usually managers that want to decide quickly and intuitively based on insights that can be understood. Therefore, visualization of data descriptions can help increase the understanding of analytic insights.

Data descriptions might indicate certain conclusions or inferences that cannot be directly deduced from the data. For example, if twice as many women than men have bought a certain product, an inference could be that women are more likely to buy this product in the future. The question, therefore, is, what kind of inductive inferences can be made from the given data descriptions, and how high is the degree of support in the data for this kind of inductive inference? There are two important inductive techniques in data analysis: attribute selection and prediction. In data abundance, not only the number of available records is incredibly large but also the number of available attributes. Automated attribute selection applies statistical and machine learning techniques for a ranking of association between attributes and a prediction target, and thus filters out the most relevant attributes for inductive inferences regarding a target class. Prediction means inference of an unknown feature (a target variable) from known features (the analytic variables). This can be accomplished to a certain degree of likelihood using a model induced by methods from statistics or machine learning. There are two kinds of predictions: inductive classification and regression. Inductive classification means predicting a categorical target variable, and regression means predicting a numerical target variable. Fuzzy classification is a special case in which the prediction consists of a numeric value indicating the gradual membership in a category.

Insights from analytics are traditionally presented to human decision makers as tables and graphics. Today, more and more analytic results are loaded automatically from analytic systems into operational systems, a concept that is called *integrated analytics*. Those systems can automate certain decision processes or display analytic insights to users of operational systems. Marketing decisions can be automated, such as choosing an individualized advertisement message in the online channel (Kaufmann & Meier, 2009). The process of integrated analytics can be described in five steps. First, analytic data is collected from different sources. Second, a predictive model is induced from the data, either in a supervised form using a target variable or in an unsupervised form of clustering or association analysis. Third, a prediction, classification, or score is assigned to the original data based on the induced model. Fourth, these predictions are transmitted to the operational systems where they are applied to decision support. Finally, outcomes of decision support—for example, sales decisions and

actions—are fed back into the analytic data pool for meta-analysis and iterative optimization.

Application of fuzzy logic techniques to analytics is called *fuzzy data analysis*. This permits gradual relationships between data and predictions. The application of fuzzy logic to analytics is appropriate when there is fuzziness in the data, the prediction target, or in the association between the two (Zimmermann H. J., 1997). The advantage of fuzzy classification methods is that the class boundary is not sharp: Patterns detected by fuzzy data mining provide soft partitions of the original data. Furthermore, an object can belong to several classes with different degrees of membership (Zimmermann H. J., 2000). Two main advantages of fuzzy logic techniques pointed out by Hüllermeier (2005) are the elimination of certain threshold effects because of gradual soft data partitioning, and the comprehensibility and interpretability of resulting models containing linguistic terms. Hüllermeier discusses possible fields of application of fuzzy set theory to data mining: Fuzzy cluster analysis partitions a set of objects into several fuzzy sets with similar properties in an unsupervised manner. Learning of fuzzy rules computes decision rules with fuzzy propositions. Fuzzy decision trees are a special case of fuzzy rules in which every node of the tree partitions the data into a fuzzy subset with an optimal distinction criterion. Fuzzy association analysis computes association rules between fuzzy restrictions on variables. Fuzzy classification partitions sharp data into fuzzy sets according to a classification predicate (Meier, Schindler, & Werro, 2008). If this predicate is inferred by induction, the process is called inductive fuzzy classification, or IFC (Kaufmann & Meier, 2009).

In business analytics, scoring methods are used for ranking data records according to desirable features. Multivariate methods based on multiple attributes, for instance linear or logistic regression, can predict targets such as customer profitability, product affinity, or credit worthiness. The scores are used in daily decision making. Record scoring based on data in an information system is an inference about unknown target variables, which is not deductive. Rather, it is a form of inductive inference. This introduces fuzziness into decision support because there is only some degree of support for the hypothesis that a given record belongs to a certain class. In fact, statistical models can increase the likelihood of correct inductive classification. Accordingly, for every scored record, there is fuzziness in the membership to the target class. Consequently, fuzzy logic is the appropriate tool for reasoning about those fuzzy target classes. The solution is to compute a continuous membership function mapping from the data into the fuzzy target class to which every record has a degree of membership. Ranking or scoring models yield continuous predictions instead of crisp

classifications. Those models can serve as target membership functions. Thus, scoring methods correspond to an IFC.

The proposed IFC method provides a means for computing inductive membership functions for a target class. These membership functions can be visualized. Furthermore, they can be used for inductive fuzzification of attributes, which can be applied for attribute selection and for improving predictive models. Thus, the proposed IFC methods can be applied in the following areas of data analysis:

o Selection: Attributes can be scored by the correlation of automatically generated inductive membership degrees with a Boolean or Zadehan target class in order to select relevant attributes.

o Visualization: Induced membership functions can serve as human-readable diagrams of association between analytic and target variables using a plot of automatically generated membership functions.

o Prediction: Datasets used for prediction can be inductively fuzzified by transforming the original data into inductive target membership degrees, which can improve the correlation of predictions with the actual class membership in existing statistical methods.

3.1.1   Selection

In practice, there is an abundance of data available. The problem is to find relevant attributes for a given data mining task. Often, thousands of variables or more are available. Most machine learning algorithms are not suited for such a great number of inputs. Also, human decision makers need to know which of those customer attributes are relevant for their decisions. Therefore, the most relevant attributes need to be selected before they can be used for visualization or prediction.

In order to find relevant attributes for predicting a target class y, an attribute selection method can be derived from membership functions induced by IFC, using the method proposed in Section 2.4. All possible analytic variables can be ranked by the correlation coefficient of their NLR with the target variable. For every attribute, X_k, the membership function in the predictive target class y', denoted by $\mu_y(X_k)$, is computed. The membership in target y is indicated by a Zadehan variable Y, which is a variable with a domain of gradual truth values in the interval between 0 and 1. Then, the Pearson sample correlation coefficients between the NLR-based membership degrees and the Zadehan values of Y are calculated. Thus, the relevance of attribute X_k regarding target y is defined as the correlation of its inductive

fuzzification with the target variable (Formula 61). The most relevant variables are those with the highest correlations.

$$relevance(X_k) := corr(\mu_{y'}(X_k), Y)$$

The fuzzification of analytic variables prior to attribute relevance ranking has the advantage that all types of analytic variables—linguistic, categorical, Boolean, Zadehan, and numeric variables—can be ranked using the same measure. Choosing correlation as a measure of association with the target has the advantage that it is a standard aggregate in SQL, and thus readily available as a well-performing precompiled function in major database systems.

To illustrate the proposed method, the attributes of the German credit data are ranked regarding the target class of customers with a good credit rating (Table 2). Those attributes have been inductively fuzzified with NLRs. The Pearson coefficient of correlation between the resulting membership degrees and the Boolean target variable *good credit rating* has been calculated. One can see that, for a credit rating, checking status, credit history, or duration of customer relationship are quite correlated attributes, whereas for example, the number of existing credits is less relevant.

This kind of attribute selection can be used as an input for visualization and prediction. When a target class is defined, the relevant correlated variables can be identified. A visualization of five to ten most relevant variables gives good insights about associations in the data.

Table 2
Example of an Attribute Ranking regarding NLR/Target Correlation[1]

Attribute	Correlation of NLR with target
Checking status	0.379304635
Credit history	0.253024544
Duration	0.235924759
Purpose	0.190281416
Savings status	0.189813415
Credit amount	0.184522282
Housing	0.173987212
Property magnitude	0.157937117
Age	0.136176815

[1] http://archive.ics.uci.edu/ml/datasets/Statlog+(German+Credit+Data)

Employment	0.128605942
Personal status	0.093525855
Installment commitment	0.091375548
Other payment plans	0.079139824
Foreign worker	0.077735317
Job	0.061875096
Other parties	0.060046535
Own telephone	0.051481771
Existing credits	0.042703728
Residence since	0.034584104
Num. dependents	0.007769441

3.1.2 Visualization

The IFC-NLR method can be applied to create visualizations of associations between an analytic variable (a factor) and a class (a target). A visualization of variable association is a human-readable presentation that allows the reader to see the distribution of target likelihood in tabular form or as a graph.

Using IFC for visualization, the table consists of relations between factor values and normalized target likelihood ratios, and the graph results in a plot of the inductive membership function. For categorical factors, the graph is a bar chart. For numerical factors, the membership function is plotted as a continuous line. The advantage of this method is that the notion of membership of a factor X in a target Y is semantically clear and intuitively understandable by readers. Furthermore, the semantics of the NLR function is mathematically clearly defined.

As an example, for two variables from the German credit dataset, checking status and duration, the inductive fuzzy membership function in the target class of customers with a good credit rating can be visualized as shown in Figure 12. This graphic can be interpreted as follows: Customers without checking accounts or with a balance of more than $200 are more likely to have good credit histories than not. Customers with a negative balance are more likely to have bad credit ratings. For the duration of the credit (in months), the shorter the duration is, the higher the likelihood of good credit history. Credits with durations of more than 19 months have higher likelihoods of being associated with bad credit histories, and accordingly, the NLR is less than 0.5.

Inductive fuzzy membership in the target "good credit rating"
for variable "Checking status"

Inductive fuzzy membership in the target "good credit rating"
for variable "Duration"
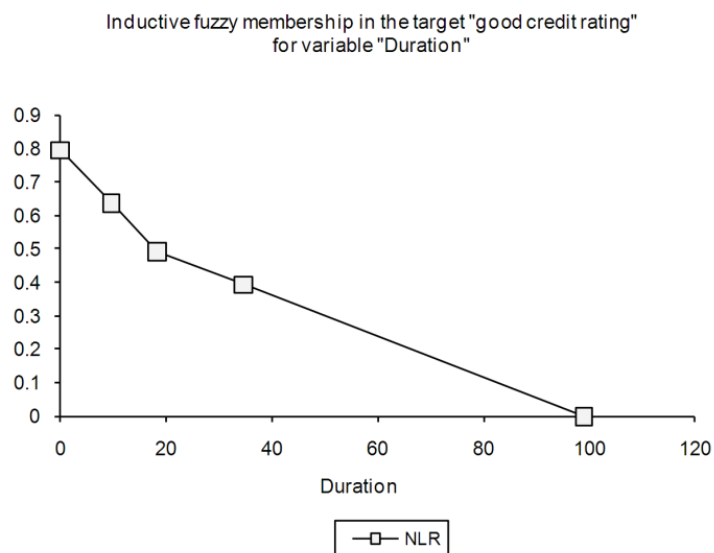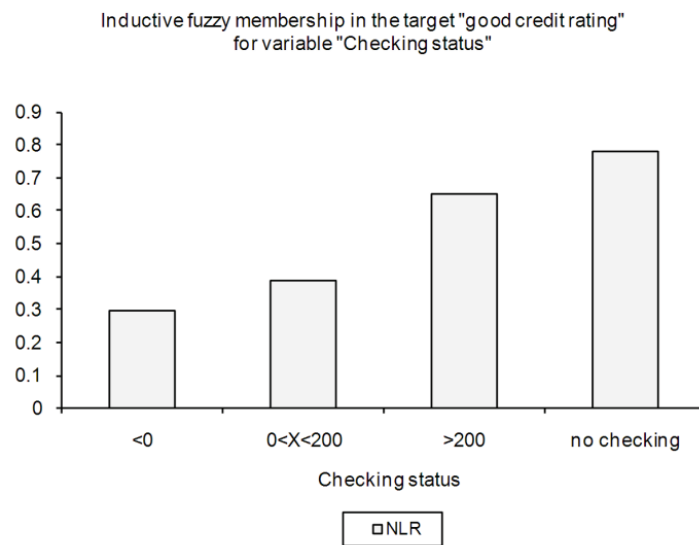
Figure 12: Visualization of relevant attributes and their association with the target as an inductive membership function.

Adapted from "Fuzzy Target Groups in Analytic Customer Relationship Management," by M. Kaufmann and C. Graf, 2012, In A. Meier and L. Donzé (Eds.), Fuzzy Methods for Customer Relationship Management and Marketing, p. 1731 Copyright 2012 by Publisher.

3.1.3 Prediction

A method for the application of likelihood-based IFC to prediction has been introduced by Kaufmann and Meier (2009). The basic idea is to create a multivariate inductive model for target class membership with a combination of inductively fuzzified attributes derived by membership function induction (see Section 2.4). The proposed approach for application of likelihood-based IFC to prediction consists of a univariate inductive fuzzification of analytic variables prior to a multivariate aggregation. This has the advantage that non-linear associations between analytic variables and target membership can be modeled by an appropriate membership function. As presented in Figure 13, the following steps are applied in order to derive an inductive membership degree of individual *i* in the prediction *y'* for class *y* based on the individual's attributes:

o A: The raw data consists of sharp attribute values.

o B: An inductive definition of the membership function for the attribute values in the predictive fuzzy class *y'* is calculated using the previously described IFC-NLC methods.

o C: The attribute values are fuzzified using the derived membership functions from step B. This step is called *inductive attribute fuzzification* (IAF), defined as *s*upervised univariate fuzzy classification of attribute values.

o D: After that, the dataset consists of fuzzified attribute values in the interval [0,1], indicating the inductive support for class membership.

o E: The several fuzzified analytic variables are aggregated into a membership degree of individuals in the predictive class. This can be a simple conjunction or disjunction, a fuzzy rule set, or a statistical model derived by supervised machine learning algorithms such as logistic or linear regression.

o F: This results in a multivariate membership function that outputs an inductive membership degree for individual *i* in class *y*, which represents the resulting prediction.

It is proposed to preprocess analytic data with IFC methods in order to improve prediction accuracy. More specifically, attributes used for data mining can be transformed into inductive membership degrees in the target class, which can improve the performance of existing data mining methods. The basic idea is to create a multivariate model for a target variable with a combination of inductively fuzzified attributes derived by membership function induction.
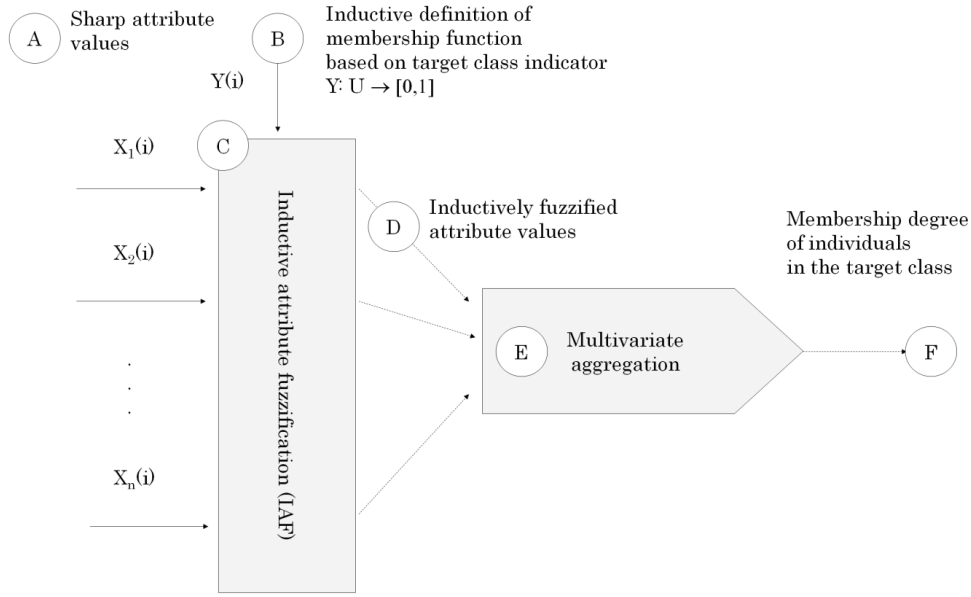
Figure 13: Proposed schema for multivariate IFC for prediction.

Empirical tests (Section 4.2) have shown that using the proposed method of IAF significantly improves the average correlation between prediction and target. For binary target variables, a combination of IAF with an NLR, and the subsequent application of a logistic regression, turned out to be optimal. This configuration is illustrated in Figure 14.

For numerical targets, a linear fuzzification (LF) of the target, IAF using NLD, and calculation of a regression tree have turned out to be optimal. This configuration is illustrated in Figure 15.

However, this improvement by IAF can be shown only in average prediction correlation. There are instances of data in which the application of IAF lowers the predictive performance. Therefore, an IAF is worth a try, but it has to be tested whether it really improves the prediction or not in the given data domain. IAF provides a tool for fine-tuning predictive modeling, but answering the question about the best classification algorithm in a specific context takes place in the data mining process in which the algorithm with the best results is selected (Küsters, 2001).
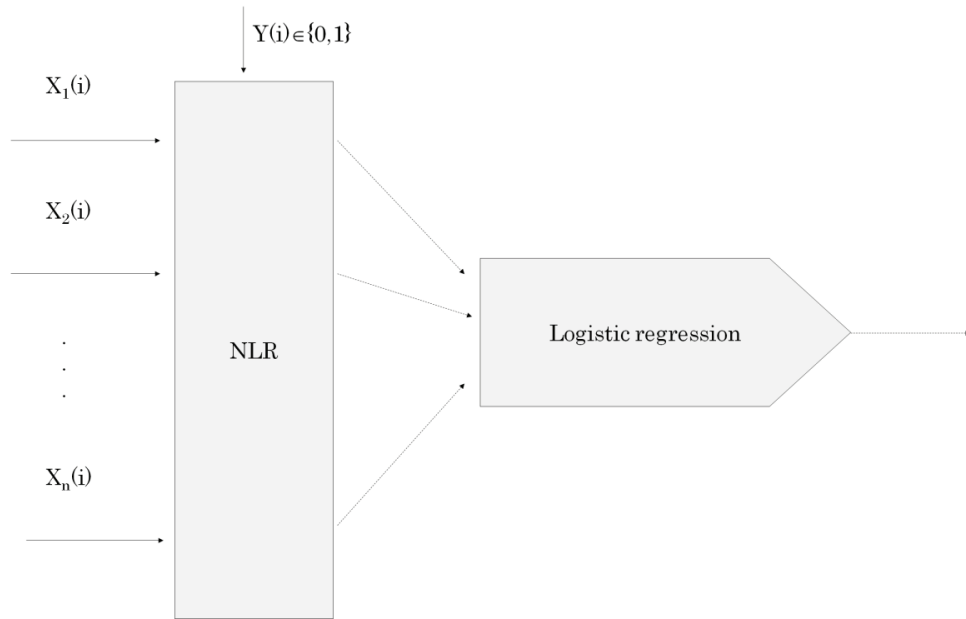
Figure 14: IFC prediction with binary target classes based on categorical attributes.
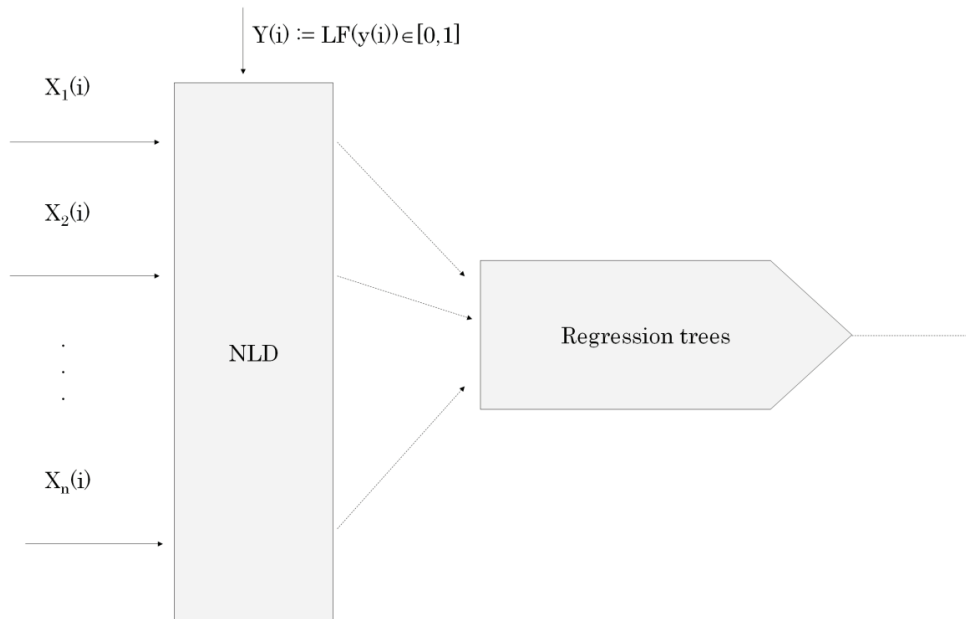
Figure 15: IFC prediction with Zadehan target classes based on numerical attributes.

3.2 Marketing Analytics

"Analytics competitors understand that most business functions—even those, like marketing, that have historically depended on art rather than science—can be improved with sophisticated quantitative techniques." (Davenport, 2006, p. 4)

The application of the method of data analysis to marketing is called *marketing analytics.* This can optimize the marketing mix and individual marketing on multiple channels. An important challenge in marketing is its financial accountability: making costs and benefits of marketing decisions explicit. Spais and Veloutsou (2005) proposed that marketing analytics could increase the accountability of marketing decisions. They point out the need of incorporating marketing analytics in daily marketing decision making, which is a conceptual shift for marketers to work with mathematical tools. Furthermore, they suggest that the problem of fuzziness in consumer information should be addressed by fuzzy logic techniques.

Individual marketing is capable of accounting for all costs and benefits of a marketing campaign to daily CRM decisions. Because all customers, advertisements, and sales are recorded in information systems, the benefit of marketing activities can be measured. For example, in an individual marketing campaign, the sales ratio of targeted customers can be compared to average or random customers, and the sales increase can be directly linked to the corresponding marketing campaigns.

The application of IFC methods to data analysis can be used in marketing analytics. Specifically, customer and product analytics, target selection, and integrated analytics are proposed as possible applications for the methods that have been developed in this thesis. Using IFC methods for selection, visualization, and prediction can support these four fields in marketing analytics. The benefits of IFC include automated generation of graphics with a linguistic interpretation, clear model semantics for visualization by membership function induction, and a possibly better target selection because of optimized predictive models using IAF. As shown in Figure 16, general application of IFC to the field of analytics for selection, visualization, and prediction can be applied to marketing analytics, specifically. Visualizations of induced membership functions of attributes in marketing targets provide fuzzy customer and product profiles. Furthermore, MFI allows fuzzy definitions of target groups based on customer characteristics. Finally, scoring methods can represent membership functions to fuzzy target groups defined as fuzzy sets of customers to which every individual customer has a degree of

membership. These can be integrated into automated analytics processes for individual marketing.
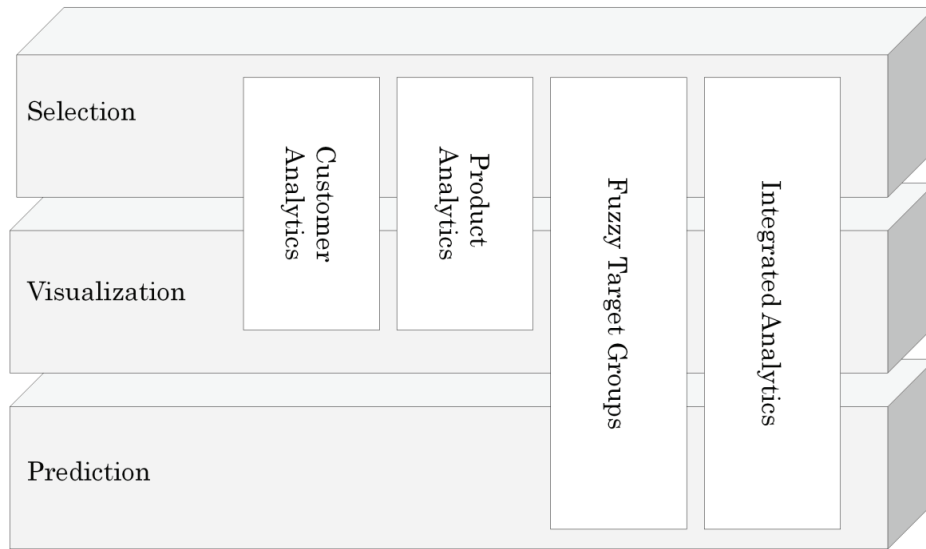
Figure 16: Proposed areas of application of IFC to marketing analytics.

3.2.1   Customer Analytics

The aim of customer analytics is to make associations between customer characteristics and target classes of interesting customers with desirable features understandable to marketing decision makers. The question is, which customer features are associated with the target? A customer report visualizes the likelihood of target class membership given different values of an attribute in order to show relevant features that distinguish the target class from the rest of the customers. In the context of CRM, there are several target customer classes that are interesting for profiling:

o   Product affinity: Customers who have bought a given product.

o   Profitability: Customers who are profitable to the enterprise.

o   Recency: Customers who are active buyers because they have recently bought a product.

o   Frequency: Customers who buy frequently.

o   Monetary value: Customers who generate a high turnover.

o   Loyalty: Customers who have used products or services for a long period of time.

A customer profile for the mentioned target customer classes should answer the question, what distinguishes customers in this class from other customers? Thus, the association between class membership and customer attributes is analyzed. The customer attributes that can be evaluated for target likelihood include all of the aforementioned characteristics, and additionally, socio-demographic data, geographic data, contract data, and transaction behavior recorded in operational computer systems, such as CRM interactions and contacts in direct marketing channels.

If one looks more closely at the target customer classes, one can see that, often, these classes are fuzzy. For example, to generate a "high turnover" is a fuzzy proposition, and the corresponding customer class is not sharply defined. Of course, one could discretize the class using a sharp boundary, but this does not reflect reality in an appropriate manner. Therefore, it is proposed to visualize fuzzy customer classes as fuzzy sets with membership functions.

In order to generate customer profiles based on a target class and the relevant customer attributes, the method of MFI by NLRs presented in Section 2.4.1 can be applied. It is proposed to select the relevant customer attributes first, using the method previously described. After that, the relevant attributes are called *profile variables*. For a profile variable X, the values $x_k \in \text{dom}(X)$ are called profile characteristics. They are assigned an inductive membership degree for the predictive target class y' defined by $\mu_{y'}(x_k) := NLR(y \mid x_k)$. A plot of the corresponding membership function represents a visual fuzzy customer profile for variable X, as illustrated by Figure 17.

Two instances of a fuzzy customer profile using NLRs are shown in Figure 17, in which the customers of investment funds products of a financial service provider are profiled by customer segment and customer age. Interpretation of the graphs suggests that customers above the age of 50 and customers in the segments *Gold* and *Premium* are more likely to buy investment funds than the average customer.

A fuzzy customer profile derived with the IFC-NLR method creates a visual image of the customer class to be analyzed by showing degrees of membership of customer features in the target characteristic. Marketing managers can easily interpret the resulting reports.
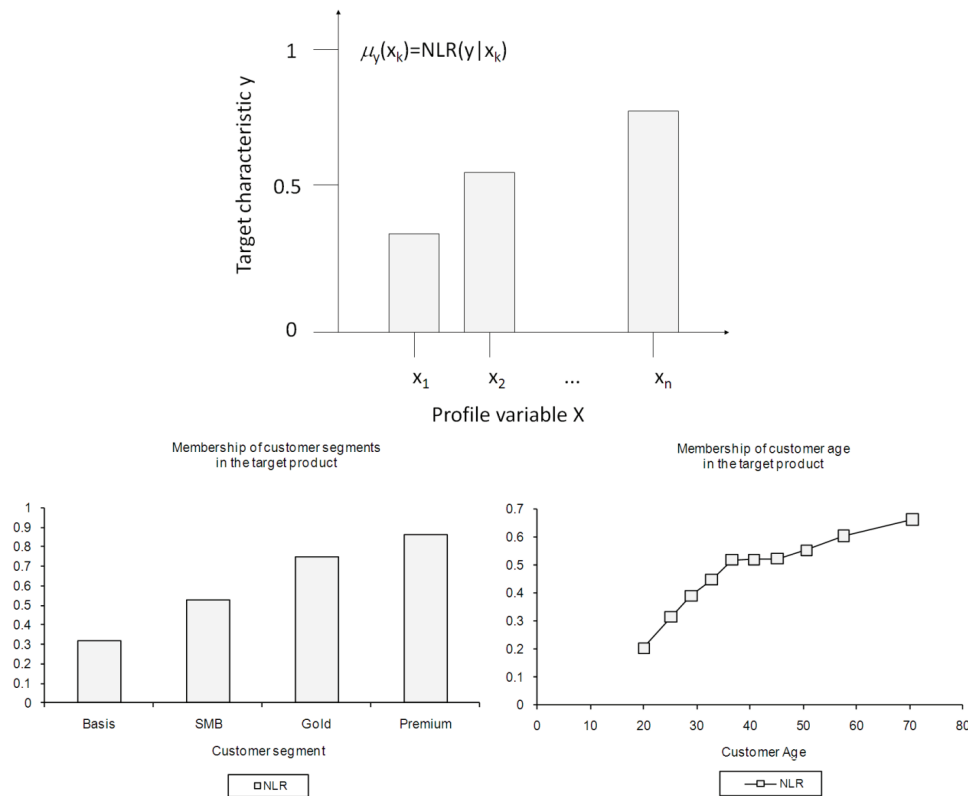
Figure 17: Schema of a fuzzy customer profile based on IFC-NLR, together with two examples of a fuzzy customer profile.

Adapted from "Fuzzy Target Groups in Analytic Customer Relationship Management," by M. Kaufmann and C. Graf, 2012, In A. Meier and L. Donzé (Eds.), Fuzzy Methods for Customer Relationship Management and Marketing, p. 173. Copyright 2012 by Publisher.

3.2.2 Product Analytics

Marketing target groups can be defined by typical characteristics of existing product users. This method is based on analysis of existing customer data in the information systems of a company, which is an instance of secondary market research. The data of customers that are product users (the test group) is compared to the data of customers that do not use the product (the control group). The attributes that separate test and control group most significantly are used for defining the target group of potential customers. These attributes are the most selective ones. Accordingly, the target group is derived inductively by similarity to the set of existing product users. As an example, a customer profile

could show that customers between 30 and 40 years of age in urban areas are twice as likely to be product users as other customers. In such a case, the target group for this particular product could be defined accordingly.

Analytic target group definitions are based on a set of customer characteristics. However, these characteristics differ in relevance. Thus, the set of relevant customer characteristics for a product target group is a *fuzzy set* because the boundary between relevant and non-relevant characteristics is gradual. The corresponding membership degree can be precisiated by a relevance or selectivity metric. It is proposed that an NLR be used as a measure of selectivity.

The likelihood of product usage, U_P (the notation is for product P), given that a customer record, i, in the existing customer database d has characteristic c, can be computed accurately as a sampled conditional probability, $p(c|U_P)_d$, defined as the number of product users that have this feature divided by the total number of product users (Formula 62). In this formula, $c(i)$ and $U_P(i)$ are Boolean truth values that indicate presence or absence of a customer characteristic and the usage of the product.

$$p(c \,|\, U_P)_d := \frac{|\{i \in d \,|\, c(i) \wedge U_P(i)\}|}{|\{i \in d \,|\, U_P(i)\}|}$$
<div align="right">Formula 62</div>

This product usage likelihood can be compared to the likelihood, $p(c \,|\, \neg U_P)_d$, of the opposite hypothesis that a customer is not a product user, given characteristic c, calculated as the number of non-users of the product that have characteristic c divided by the total number of non-users (Formula 63).

$$p(c \,|\, \neg U_P)_d := \frac{|\{i \in d \,|\, c(i) \wedge \neg U_P(i)\}|}{|\{i \in d \,|\, \neg U_P(i)\}|}$$
<div align="right">*Formula* 63</div>

Thus, the selectivity of a customer characteristic c can be expressed by the ratio between the likelihood of product usage, given c, and likelihood of product non-usage, given c (Formula 64). This ratio can be normalized as formalized in Formula 65.

$$LR(U_P \,|\, c)_d := \frac{p(c \,|\, U_P)_d}{p(c \,|\, \neg U_P)_d}$$
<div align="right">*Formula* 64</div>

$$NLR(U_P \mid c)_d := \frac{p(c \mid U_P)_d}{p(c \mid U_P)_d + p(c \mid \neg U_P)_d} \qquad \textit{Formula 65}$$

Finally, the selectivity of characteristic *c*, expressed as an NLR, represents an inductive degree of membership of this characteristic in the fuzzy target group definition for product *P*. An inductive fuzzy target group definition, *t*, based on analysis of database *d* is a fuzzy set of customer features and its membership function is defined by the corresponding *NLR* (Formula 66).

$$\mu_t(c)_d := NLR(u_P \mid c)_d \qquad \textit{Formula 66}$$

A customer characteristic may or may not be defined by a granular attribute value. Characteristics can also be computed by functional combination of several basic attribute values. Furthermore, the customer characteristic indicator *c(i)* can indicate a fuzzy truth value in the interval [0,1] if the corresponding characteristic *c* is a fuzzy proposition such as "high turnover." In that case, the definition of the NLR for fuzzy truth values from Section 2.4.1 can be applied. However, the aim of analytic target group definition is to find or construct optimal defining target customer characteristic indicators with the highest possible degree of membership in the target group definition. These indicators can be constructed by logical connections or functions of granular customer characteristics.

This kind of analytic target group definition is a conceptual one, suited for presentation to human decision makers, because it is intuitively understandable. It results in a ranking of customer characteristics that define typical product customers. However, for integrated analytics in analytic CRM, a scoring approach is more promising because it yields better response rates, although the resulting models may be less comprehensible.

3.2.3 Fuzzy Target Groups

Contemporary information technology provides the means to individualize marketing campaigns. Each customer is targeted not only directly but also individually with an advertisement message, decision, or activity. The behavior of an organization toward an individual customer is analytically customized according to the customer's classification, and customers with different characteristics are targeted with different behaviors. Individual customers are assigned a score for different CRM targets based on a predictive multivariate model. IFC

methods can be applied to customer data for individual marketing in order to calculate a *predictive scoring model* for product usage. This model can be applied on the data to score customers for their product affinity, which corresponds to an IFC of customers in which the predictive model is a multivariate membership function.

An aim of customer analytics is the application of prediction to the selection of a target group with likelihood of belonging to a given class of customers. For example, potential buyers or credit-worthy customers can be selected by applying data analysis to the customer database. This is done either by *segmentation* or *scoring* (Sousa, Kaymak, & Madeira, 2002). By applying a segmentation approach, sharp sets of customers with similar characteristics are calculated. Those segments have a given size. For individual marketing, a scoring approach is more promising, in which every customer is assigned a score predicting a likelihood of response. The score is calculated by application of a predictive multivariate model with numeric output, such as neural networks, linear regression, logistic regression, or regression trees. When a numeric score can be normalized, it represents a membership function to a fuzzy set of customers with a high response likelihood—a fuzzy target group—to which every customer has a degree of membership. This scoring process can be applied for every product. Thus, for every individual customer, the degree of membership to all possible cross-selling target groups is known, and in direct customer contact, the customer can be assigned the advertisement message with the highest score.

There are different methods for fuzzy customer scoring. For example, fuzzy clustering for product affinity scoring was presented by Setnes, Kaymak, and van Nauta Lemke (1998), in which the reason for using fuzzy systems instead of neural networks is declared as the comprehensibility of the model. Kaufmann and Meier (2009) evaluated a supervised fuzzy classification approach for prediction using a combination of NLRs with an algebraic disjunction. In this thesis, a synthesis of probabilistic modeling and approximate reasoning applying fuzzy set theory is proposed for prediction (as previously explained), using univariate inductive membership functions for improving the target correlations of logistic regressions or regression trees. However, all inductive scoring methods that yield a numeric value representing response likelihood can be normalized in order to represent an inductive membership function to a fuzzy set of customers, and can, therefore, be categorized as IFC.

A classical sharp CRM target group $T \subset C$ is a subset of all customers C defined by a target group definition t: $T := \{i \in C \mid t(i)\}$. In the case of scoring methods for target selection, the output of the model application is numeric and can be normalized in the interval $[0,1]$ to

represent a membership function. In that case, the target group is a fuzzy set T' and the (normalized) customer score is a membership function, $\mu_{T'}:C\rightarrow[0,1]$. This score is defined by a predictive model, M, as a multivariate combination of n customer attributes, $X_1, \ldots, X_n$, as shown in Formula 67.

$$\mu_{T'}(i) := M(X_1(i),...,X_n(i)) \qquad \textit{Formula 67}$$

Because T' is a fuzzy set, it has to be defuzzified when a campaign requires a binary decision. For example, the decision about contacting a customer by mail has to be sharp. An alpha cutoff of the fuzzy set allows the definition of individual marketing target groups of optimal size regarding budget and response rate, and leads to a sharp target group T_α, as formalized in Formula 68.

$$T_\alpha := \{i \in C \mid \mu_{T'}(i) \geq \alpha\} \qquad \text{Formula 68}$$

3.2.4   Integrated Analytics

For individual marketing, the fuzzy target group membership degrees are predictive scores that are processed via the CRM system, which dispatches them to the distribution channel systems for mapping customers to individualized advertisement messages. In all computer supported channels with direct customer contact, inbound or outbound, as soon as the customer is identified, a mapping can be calculated to the product to which the customer has the highest score. Based on that mapping, the advertisement message is chosen. In the online channel, logged-in customers are displayed individual advertisement banners. Customers are sent individual letters with product advertisement. In the call center and in personal sales, if the customer can be identified, the agent can try cross-selling for the next best product.

In *analytic customer relationship management* (aCRM), customer data is analyzed in order to improve customer interactions in multiple channels (Turban et al., 2007). This can be applied to individual marketing. Target groups for aCRM campaigns are derived analytically and individually by application of classification and regression models to individual customer data records. The aim is to increase campaign response rates with statistical methods. Cross-selling campaigns analyze product affinity of customers given their features. *Churn*

(change and turn) campaigns target customers with a low loyalty prediction in order to re-gain their trust. Figure 18 illustrates a schematic information system that can enable aCRM processes.

In contemporary enterprises, due to intensive application of computing machinery and electronic databases in business processes, there exist large amounts of possibly useful customer data in various software applications that can serve as data sources for aCRM. The heterogeneity of data storage implies a necessity for data integration before they can be analyzed. A system that automates this task is called a *customer data warehouse* (Inmon, 2005). Based on this integrated data pool, analytics provides inductions of predictive models for desirable customer classes, such as product purchase, profitability, or loyalty. If these models output a gradual degree of membership to the target classes, their application to the customer database can be called an IFC. These membership degrees are defined by a prediction of class membership for individual customers, sometimes called *lead detection*. The corresponding fuzzy sets of customers can be called fuzzy target groups. These analytic results, called *leads* by aCRM managers, are transferred into the operational CRM application, where their presentation to human decision makers provides decision support for direct customer contact. The utilization of analytic data output (leads) takes place in individual marketing channels with direct customer contact—for instance, web presence, mailing, call center, or personal sales. The results of aCRM campaigns, such as sales, are electronically collected and fed back into the customer data warehouse in order to improve future aCRM campaigns by meta-analytics. This mechanism is called a *closed loop*. Generally, aCRM is an instance of a business intelligence process (Gluchowski, Gabriel, & Dittmar, 2008), which involves data sourcing, integration, analytics, presentation, and utilization. Furthermore, it is also an instance of integrated analytics, in which analytics are integrated into operational systems with feedback mechanisms.

In comparison to mass marketing, in individual marketing, every customer is provided with the advertisement that fits best. Especially when customer databases, such as a data warehouse, are present, individual scoring is possible based on available data: The customer's target membership score is used to assign an individual advertisement message to each customer. These messages are delivered electronically via customer relationship management application to individual marketing channels. Campaign target groups are individualized. All customers in the target groups are known individually and are contacted directly.
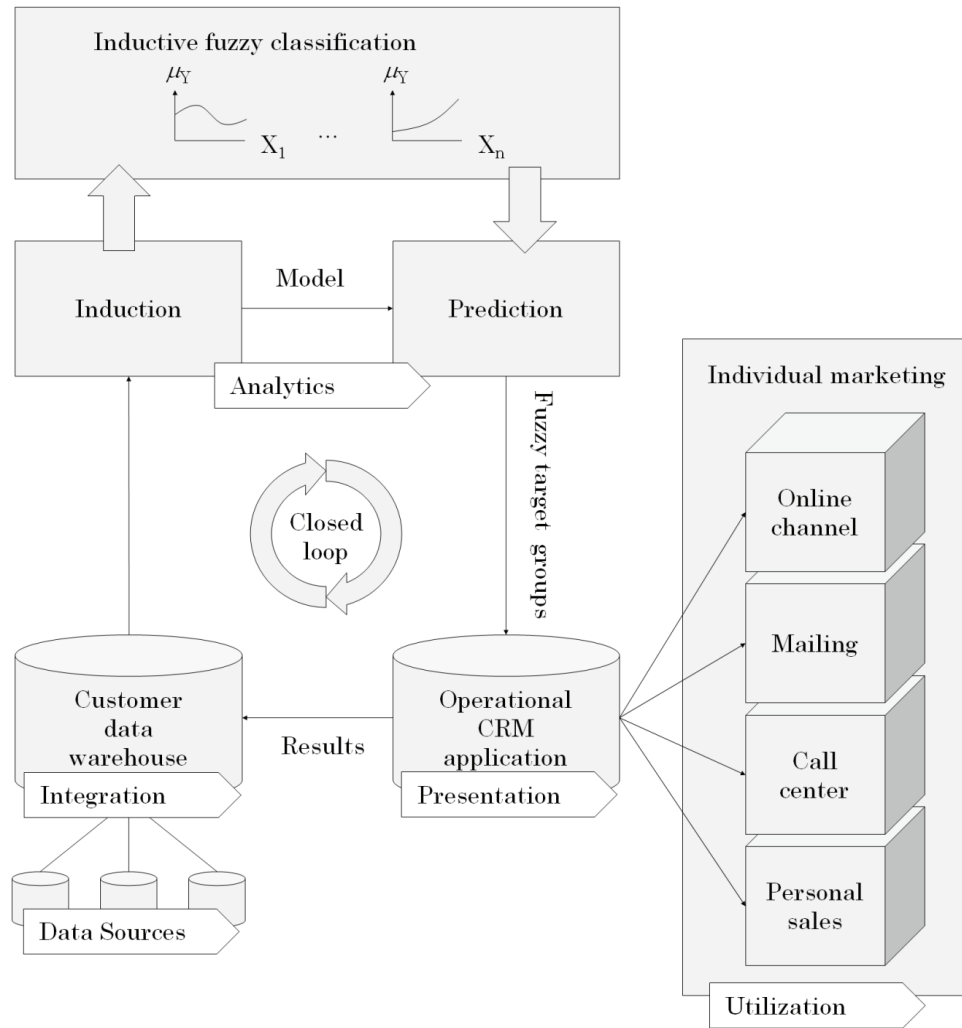
Figure 18: Inductive fuzzy classification for individual marketing.

3.2.5 Case Study: IFC in Online Marketing at PostFinance

The following case study shows the application of the proposed methods in practice, from a point of view of information systems and methodology. Membership function induction is applied to a real-world online marketing campaign. A comparison with crisp classification and random selection is made.

PostFinance Inc.[2] is a profitable business unit of Swiss Post. Its activities contribute significantly to the financial services market in Switzerland. PostFinance is an analytic enterprise feeding business processes with information gained from predictive analytics.

The online channel of PostFinance supports individual advertisement banners for logged-in customers. Based on the customer score, a customer is assigned the advertisement message for the product with the highest response likelihood computed for that particular customer. By clicking on the banner, the customer has the opportunity to order the product directly.

The marketing process of PostFinance uses analytical target groups and predictive scores on product affinity for individual marketing. Target groups for online marketing campaigns are selected using inductive classification on the customer data warehouse. Figure 19 shows the process from customer data to individualized online marketing. Target groups are defined in the data warehouse. This is done using a predictive model (e.g., logistic regression) or a crisp classification (e.g., customers above the age of 50 with a balance higher than CHF 10,000). A dataset is generated that assigns an individual advertisement message to every single customer based on his or her target group. This dataset is loaded into the online platform, where it controls the online advertisements. In the online channel, for every customer that logs in, the individual advertisement message is mapped according to the previous target group selection.

A case study was conducted with PostFinance in 2008 in order to test IFC in a real online marketing campaign. Inductive fuzzy classification was applied in a PostFinance online marketing campaign promoting investment funds. The aim of the prediction was to forecast customers with an enhanced likelihood of buying investments funds. The resulting fuzzy classification yielded a fuzzy target group for an individual marketing campaign in which every customer had a gradual degree of membership.

---

[2] http://www.postfinance.ch
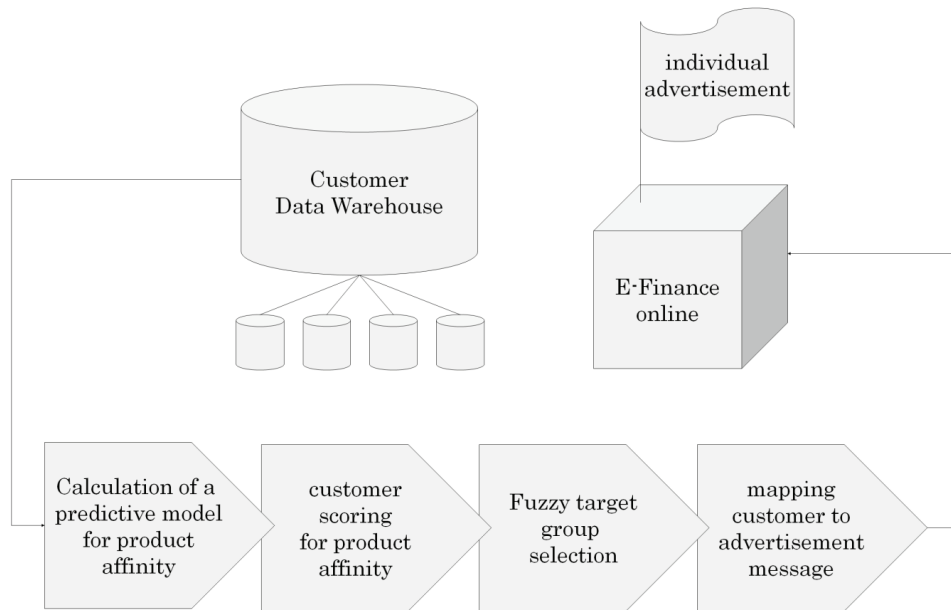
Figure 19: Analytics applied to individual marketing in the online channel at PostFinance.

Adapted from " An Inductive Fuzzy Classification Approach applied to Individual Marketing.," by M. Kaufmann and A. Meier, 2009, In *Proceedings of the 28th North American Fuzzy Information Processing Society Annual Conference*, p. 4. Copyright 2009 by Publisher.

The same advertisement message was shown to three groups of customers: a test group of 5,000 customers with the highest degree of membership to a fuzzy target group, a control group of 5,000 customers selected by classical crisp target group definition, and a second control group with randomly selected customers. The hypothesis was that a gradual scoring approach (an inductive fuzzy customer classification) would provide better response rates than a crisp Boolean target selection (a segmentation approach), and that a fuzzy target group selection using typical characteristics of product users would classify customers more softly and eliminate certain threshold effects of Boolean classification because attributes can be compensated for with a multivariate gradual approach. It was tested whether customers in a fuzzy target group selected with a scoring approach had a higher response rate than customers in a crisp target group selected with a classical segmentation approach. Furthermore, the two groups were compared with a random selection of customers. In the following, it is shown how the data mining methodology presented in Section 2.4 has been applied to a real-world direct marketing campaign.

86

In order to prepare a test set for model induction, a sample customer dataset was selected from the customer data warehouse. Because of the excellent quality of the cleaned and integrated data in the data warehouse, the data preparation step was accomplished by a simple database query. The dataset contained anonymous customer numbers associated with customer specific attributes. As the target variable, the class label was set to 1 if the customer had investment funds in his product portfolio and to 0 if else. Mutual information of the dependent variables with the target variable was chosen as a ranking method. The following attributes were selected as relevant:

o   Customer segment (CS; 0.047 bit),

o   Number of products (NP; 0.036 bit),

o   Overall balance (OB; 0.035 bit),

o   Loyalty (L; 0.021 bit),

o   Customer group (CG; 0.016 bit),

o   Balance on private account (BP; 0.014 bit), and

o   Age (A; 0.013 bit).

Table 3
Conditional Probabilities and NLRs for a Categorical Attribute

$X1$: customer segment	$Y=1$	$Y=0$	$p(X1 \mid Y=1)$	$p(X1 \mid Y=0)$	$NLR_{Y=1}(X)$
Basis	11'455	308'406	0.38	0.82	0.32
SMB	249	2'917	0.01	0.01	0.52
Gold	12'666	54'173	0.43	0.14	0.75
Premium	5'432	10'843	0.18	0.03	0.86
Total	29'802	376'339			

Using the method presented in Section 2.4, for each of the relevant attributes, the fuzzy restriction corresponding to the likelihood of having investment funds was induced. In the following section, the induction processes for a categorical and a continuous attribute are described in detail.

As the first example, in the domain of the categorical attribute *customer segment* (X_1), there are four values: *Basis*, *SMB*, *Gold*, and *Premium*. For customers who have not yet bought investment funds, the aim is to define a degree of membership in a fuzzy restriction on the customer segment domain in order to classify them for their likelihood to buy that product in the future.

The frequencies and conditional probabilities are presented in Table 3. The first column indicates the customer segment. Column Y=1

contains the number of customers of each segment who have bought investment funds. Column Y=0 contains the number of customers of each segment who have not bought that product. According to Formula 69, the membership degree of segment *Basis* in the fuzzy restriction y_1 was induced as follows:

$$\mu_{y_1}("Basis") = \frac{p(X_1 = "Basis" \mid Y = 1)}{p(X_1 = "Basis" \mid Y = 1) + p(X_1 = "Basis" \mid Y = 0)}$$

Formula 69

$$= \frac{0.38}{0.38 + 0.82} = 0.32 = NLR_{Y=1}("Basis")$$

The other degrees of the membership were induced analogically. The resulting values are shown in column NLR of Table 3. The corresponding membership function is illustrated in Figure 20.
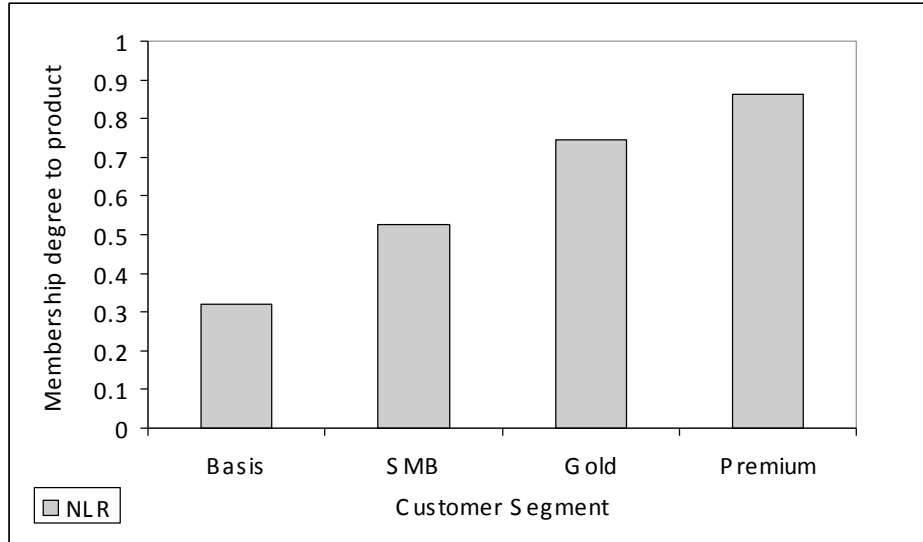
Figure 20: Membership function induction for a categorical attribute.

Adapted from " An Inductive Fuzzy Classification Approach applied to Individual Marketing.," by M. Kaufmann and A. Meier, 2009, In *Proceedings of the 28th North American Fuzzy Information Processing Society Annual Conference*, p. 4. Copyright 2009 by Publisher.

As a second example, for the continuous attribute *overall balance*, the membership function was induced in the following way: First, the NLR for deciles of the attribute's domain was calculated analogically to the previous example, represented by grey squares in Figure 21. Then, a function, $A / (1 + \exp(B - C \ln(x + 1))) + D$, was fitted by optimizing the parameters A to D using the method of John (1998). The resulting membership function for the customer overall balance is shown as a line in Figure 21.
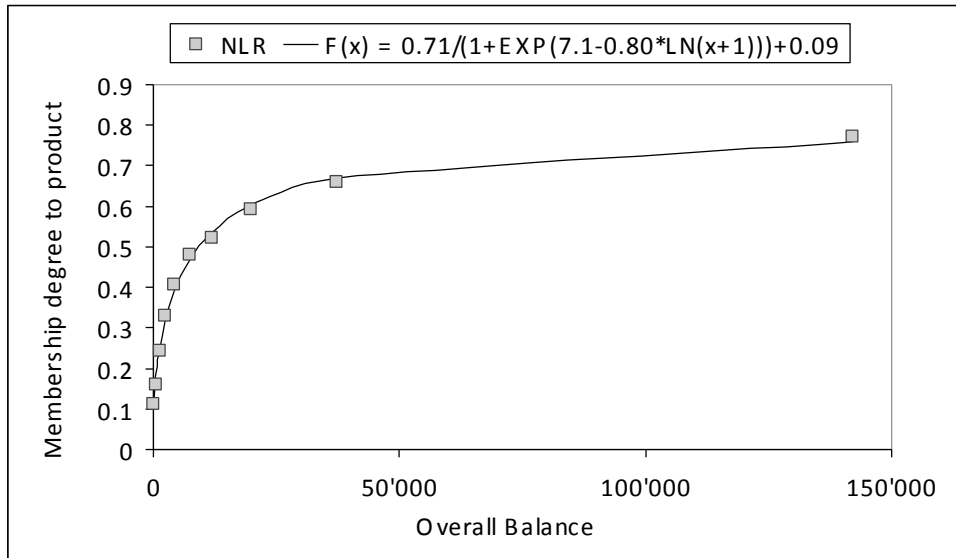
Figure 21: Membership function induction for a continuous attribute.

Adapted from " An Inductive Fuzzy Classification Approach applied to Individual Marketing.," by M. Kaufmann and A. Meier, 2009, In *Proceedings of the 28th North American Fuzzy Information Processing Society Annual Conference*, p. 5. Copyright 2009 by Publisher.

Every relevant attribute of the original dataset was transformed to a fuzzy membership degree using SQL directly in the database. Categorical variables were transformed using a case differentiation statement. For example, the attribute customer segment was transformed using the following SQL command:

```
select case
when customer_segment = 'Basis' then 0.32
when customer_segment = 'SMB' then 0.52
when customer_segment = 'Gold' then 0.75
when customer_segment = 'Premium' then 0.86
end as fuzzy_customer_segment
from ads_funds
```

Continuous variables were transformed using a function expression. For example, the attribute *overall balance* was fuzzified using the following SQL expression:

```
select
0.71/(1+EXP(7.1-0.8*LN(overall_balance + 1)))+0.09
as fuzzy_overall_balance
from ads_funds
```

The individual fuzzy attribute domain restrictions were then aggregated to a multivariate fuzzy class of customers. This was done using an SQL statement implementing the gamma operator (Zimmermann & Zisno, 1980) defined in Formula 70.

$$
\begin{aligned}
&\Gamma_\gamma(\mu_{y_1}(e_j), \ldots, \mu_{y_n}(e_j)) \\
&:= \mu_{\bigcup_i y_i}(e_j)^\gamma \cdot \mu_{\bigcap_i y_i}(e_j)^{1-\gamma} \\
&:= (1 - \prod_i 1 - \mu_{y_i}(x_{ij}))^\gamma \cdot (\prod_i \mu_{y_i}(x_{ij}))^{1-\gamma}
\end{aligned}
\qquad \text{Formula 70}
$$

Table 4
Optimization of the Gamma Parameter for Multivariate Fuzzy Aggregation

	I(Y;Y')	I(Y;Y'=1)	LR(Y'=1)
Gamma = 0	0.018486	0.4357629	6.7459743
Gamma = 0.5	0.0185358	0.4359879	6.7488694
Gamma = 1	0.0209042	0.4787567	7.2359172

In order to define the gamma parameter, different performance measures were calculated. As shown in Table 4, a gamma of 1, corresponding to an algebraic disjunction or full compensation, was most successful. Thus, the multivariate fuzzy classification was performed using the following SQL statement:

```
Select
(1-
(1-fuzzy_number_of_products)
* (1-fuzzy_customer_segment)
* (1-fuzzy_overall_balance)
```

```
* (1-fuzzy_loyalty)
* (1-fuzzy_age)
* (1-fuzzy_balance_on_private_account)
* (1-fuzzy_customer_group)
) as multivariate_fuzzy_classification
from f_ads_funds
```

As a result, a fuzzy class of customers was calculated, for whom the degree of membership indicates the product affinity for investment funds and represents a product affinity score. This can be used in individual marketing for defining target groups for different products.

In order to test the resulting fuzzy classifier, a pilot marketing campaign was performed using the resulting fuzzy target group. First, a target group of 5,000 customers with the highest membership degree was selected from the fuzzy class using an alpha cutoff (Test Group 1). Second, as a comparison, 5,000 other customers were selected using a crisp classification (Test Group 2), using the following crisp constraints:

```
Select case when
Customer segment in ('Gold', 'Premium', 'SMB')
And Customer_group = 50 Plus
And Loyalty > 14
And Number_of_products > 1
And Age between 35 and 75
And Balance on private account > 3000
And Overall_balance > 20000
Then 1 else 0 end as multivariate_crisp_classification
From ads_funds
```

This conventional target group selection used the same seven customer attributes as the fuzzy classification, but the classification was done using a crisp constraints predicate. Third, 5,000 customers were selected randomly (Test Group 3) in order to compare inductive classification to random selection.

To each of those 15,000 customers, an online advertisement for investment funds was shown. After the marketing campaign, the product selling ratio during three months was measured. The results are shown in Table 5. The column Y=1 indicates the number of customers who have bought the product. The product selling ratio for the target group selected by IFC was the most effective. This individual marketing campaign using inductive fuzzy target group selection

showed that fuzzy classification can lead to a higher product selling ratio than crisp classification or random selection. In the case study, IFC predictions of product affinity were more accurate than those of the crisp classification rules on exactly the same attributes. The conclusion is that, in comparison to crisp classification, fuzzy classification has an advantageous feature that leads to better response rates because it eliminates certain threshold effects by compensation between attributes.

Table 5
Resulting Product Selling Ratios per Target Group

Test group	Y=1	Y=0	Sales Rate
1: Fuzzy classification	31	4939	0.63%
2: Crisp classification	15	5037	0.30%
3: Random selection	10	5016	0.20%

4 Prototyping & Evaluation

Three software prototypes were programmed as proofs of concept of the technological aspects of the proposed methods for automated MFI in fuzzy data analysis. Master's students developed two of them, iFCQL and IFC-Filter for Weka. The author developed the inductive fuzzy classification language IFCL. This implementation also allows experiments on the implemented methods for experimental evaluation. Using a meta-induction approach, the designed method was applied for prediction in several real-world datasets in order to analyze characteristics and optimal parameters of the constructed methodology and to compare it to conventional predictive approaches. Classical inductive statistical methods were applied to gain insights about induction by IFC.

4.1   Software Prototypes

4.1.1   IFCQL

The first attempt of prototyping software for IFC was developed in a master's thesis by Mayer (2010). The basic idea was an extension of the existing fuzzy classification and query language FCQL, for which a prototype interpreter was built by Werro (2008).

In order to derive fuzzy membership functions directly from the underlying data, the FCQL language was extended with commands to induce membership functions in order to classify data based on them. For example, it was intended to describe MFI using the syntax "induce fuzzy membership of <dependent variable> in <target variable> from <relation>." For more detail on the iFCQL syntax, see the report by Mayer (2010).

As shown in Figure 22, the architecture of iFCQL reflects the language-oriented design approach. The user can enter commands in iFCQL, which are translated into SQL, or he or she can enter SQL in the command shell, which is transmitted directly to the database by a database connector. Using language instead of a graphical user interface, it is possible to save and reload scripts for data mining processes—this was the intention. Nevertheless, the implementation of the iFCQL language interpreter, consisting of lexer, parser, evaluator, and SQL generator, turned out to be a complex task that is not directly associated with IFC.

The original idea was to develop a tool to support the proposed data mining methodology of this thesis (Section 2.4). Thus, there was one statement class for each of the six steps: data preparation (audit),

MFI, fuzzification of attribute values, attribute selection, fuzzy classification of data records, and model evaluation. Eventually, only three syntax elements for the main tasks were implemented, namely MFI, univariate classification of attributes, and multivariate classification of records.
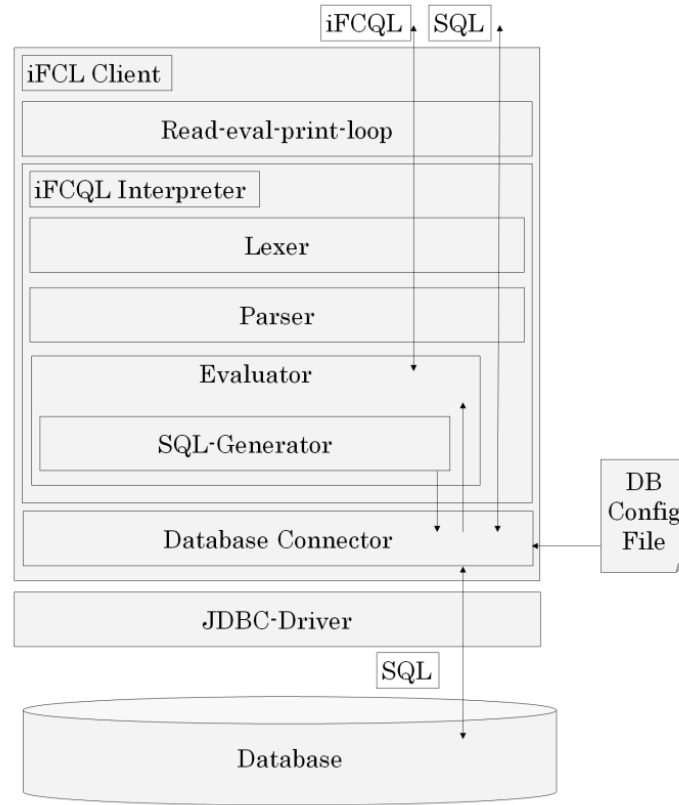
Figure 22: Architecture of the iFCQL interpreter.

Adapted from "Implementation of an Inductive Fuzzy Classification Interpreter," (master's thesis), by P. Mayer, 2010, p. 59. Copyright 2010 by Mayer, University of Fribourg, Switzerland.

4.1.2   IFCL

This section presents an implementation of a tool that enables the computation of inductive membership functions based on the methodology proposed in Section 2.4. The inductive fuzzy classification language (IFCL) is a research prototype. Its goal is to show the

94

feasibility of implementing the data mining methodology based on IFC. It is a markup language designed for simplicity of description and parsing; therefore, the development of the language interpreter was kept simple. In addition to supporting the basic steps of the proposed method, the scope is to support automated experiments for benchmarking the predictive performance of the proposed methodology. This meta-induction experiment and its results are presented in Section 4.2
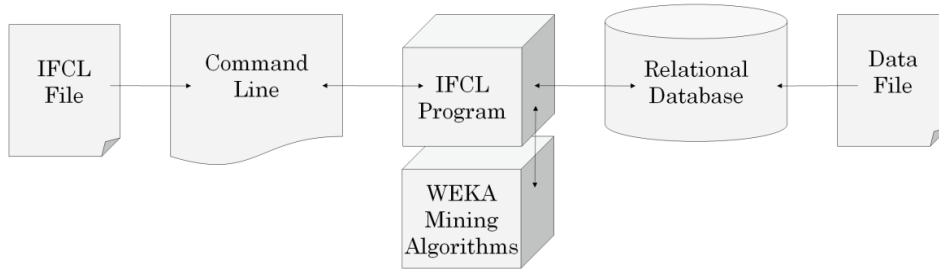
Figure 23: Inductive fuzzy classification language (IFCL) system architecture.

As shown in Figure 23, the system architecture of IFCL consists of an IFCL file, a command line shell, the IFCL program itself, Weka regression algorithms, a relational database, and a data file. In the IFCL file, the data mining process steps are described in the IFCL language. In a command line shell, the IFCL program is invoked, and the corresponding IFCL file is passed to the program. There is no graphical user interface because the research concentrated on evaluating automated inductive inferences. The IFCL actions representing steps of the data mining process are interpreted by the IFCL program, which translates them into SQL queries to the relational database. For the supervised multivariate aggregation functions linear regression, logistic regression and M5P regression trees, open source implementations of mining algorithms from the Weka machine learning workbench (Hall et al. 2009) are accessed by the IFCL program. The IFC process steps that have been translated to SQL are executed on a database server. Analytic data is loaded into the database via a data file in comma-separated value format (CSV). The IFCL language supports a simple load functionality that creates the necessary database tables based on a data description and loads records from a file into database tables. The data file contains the data to be analyzed, and the IFCL program provides the means to compute predictive models based on the data in the form of membership functions.

The functionality of the software encompasses all steps of the proposed data mining process for IFC as presented in Section 2.4.2. This

includes data preparation, univariate MFI, fuzzification of attribute values, attribute selection, multivariate aggregation, data classification, prediction, and evaluation of predictions. This prototype provides the possibility

o   to prepare data using arbitrary SQL statements,

o   to induce membership functions with all thirteen methods proposed in Table 1 in Section 2.4.1,

o   to fuzzify attribute values using the induced membership functions directly in the database,

o   to rank attribute relevance by sorting correlation between membership degrees and target class indicators,

o   to aggregate many inductively fuzzified attributes into a fuzzy class membership for data records using eight different aggregation methods shown in Table 6,

o   to assign membership degrees by fuzzy classification to data records in the database,

o   and to evaluate predictive models with correlation analysis.

All of these steps can be programmed in IFCL-syntax. IFCL is fully automatable using scripting files. All models are displayed and stored in SQL syntax. Therefore, it provides repeatability of experiments and comprehensibility of models with respect to their database implementation as queries. Details on the syntax and application of IFCL can be found in the appendix (Section A.4) To summarize, the functionality of IFCL encompasses the following points:

o   Executing IFCL script files in batch mode,

o   Connecting to the database,

o   Dropping a database table,

o   Executing an SQL script,

o   Loading data into the database,

o   Inducing a membership function,

o   Classification of data,

o   Aggregating multiple variables,

o   Evaluating predictions,

o   Data preparation, and

o   Attribute selection.

Steps of the data mining process in Section 2.4.2 are invoked in so called IFCL actions. A sequence of actions is listed in IFCL files. IFC-processes can be described in a tree-like structure of IFCL files. There are nodes and leaves of the IFCL tree. An IFCL file is either a node as a list of calls to other IFCL files or a leaf containing actual execution sequences. IFCL leaves consist of a database connection followed by a sequence of IFCL actions. An IFCL file can call the execution of several other predefined IFCL files in batch mode using the action *execifcl*. In order to do so, one needs to indicate the correct operation system file path of the corresponding IFCL file, either as an absolute path, or a path relative to the location of the process that invoked the IFCL program.

IFCL can connect to a database with the action *connect*. This is necessary for operation and has to be done at the beginning of an IFCL file leaf. One needs to know the hostname of the database server, the service identifier, the port number, a valid username, and a password. The user must be granted the rights to read tables. For data classifications, the user needs grants to create tables. The action *drop* allows the deletion of an existing database table. The IFCL program tests whether the indicated table exists, and executes a drop command to the database if and only if the table exists in the database. This is useful in preventing an exception because of an attempt to drop a non-existing table. IFCL allows executing SQL statements in the database with the action *execsql*. A single SQL command can be indicated directly within the IFCL syntax. Scripts (statement sequences) can be called from a separate SQL script file. The file system path has to be indicated in order to call an execution of an SQL file.

In order to load analytic data into the database server, there are two possibilities. First, the SQL*Loader can be called, which is highly parameterized. Second, a simpler load mechanism can be invoked involving less code. The SQL*Loader has the advantage that it is extremely configurable by a control file. The drawback is that there is a large amount of code that needs to be written for simpler load tasks. Therefore, for simpler loads that do not involve transformation, the IFCL load utility is a faster method to load data into the database. The call to the SQL*Loader can be made from an IFCL file using the action *sqlldr*. In order to load data, one needs a data file containing the analytic data and an SQL*Loader control file for the configuration of the load process. Furthermore, one can indicate two SQL*Loader parameters, namely skip and errors. (For more details, consult the SQL*Loader documentation.) For simpler data loads, the IFCL program provides a data loading utility with the action *load*. The advantage is that the load action automatically creates or recreates a table for the data load based on an enumeration of attributes and their type, where

the type is either n (numerical) or c (categorical). In the enumerations, the number of the attribute and the number of its corresponding type must be equal: Type 3 corresponds to Attribute 3. The attribute names will correspond to table columns, so their syntax is syntactically restricted to valid column names. The load action must also be provided a path to a file containing the load data, and a delimiter that separates data cells in the text file. White spaces and semicolons cannot be used as delimiters because they are syntactical elements of the IFCL language, and an escape mechanism has not been implemented.

In order to induce a univariate membership function for a single attribute to the target class, the IFCL action *inducemf* can be used. A database table is indicated on which the MFI will be based. One specific table column is defined as a target variable. As analytic variables, a specific table column can be defined, a list of columns can be enumerated, or all columns can be chosen for MFI. One of several different methods for MFI can be chosen (see Section 2.4.1). The output containing the induced membership function in SQL syntax is written to the indicated output file. This SQL classification template will be used later for univariate fuzzification of input variables. When more than one column is chosen as analytic variables, some of those variables can be skipped. The IFCL action will skip columns if they are enumerated in the corresponding action parameter. This means that those columns are not considered for MFI and they are also not copied into the fuzzified table. Alternatively, some columns can be left out by indicating the IFCL action to leave columns in the corresponding action parameters the way they are. This means that, for those columns, no membership function is induced, but those columns are copied into the fuzzified table in their original state. This can be useful to incorporate original key columns in a fuzzified table.

In order to fuzzify the input variables using the membership functions obtained, the action *classify* can be invoked using the previously generated SQL classification template file. The classification action can also apply multivariate fuzzy classification templates obtained by the IFCL action *aggregatemv*, which aggregates multiple variables into a multivariate fuzzy class. Template files generated by the IFCL actions *inducemf* and *aggregatemv* contain an SQL query in which the input and output tables are parameterized. A data classification applies these two parameters (classified table and output table) to this query template stored in the template file. Data is read from an existing table, the classified table, and transformed using the induced membership functions, and the resulting transformation is written into the output table.

In order to aggregate multiple variables into a multivariate fuzzy classification, the corresponding aggregation function can be computed

using the IFCL action *aggregatemv*. A base table is indicated on which the aggregation takes place. This is usually a transformed table containing inductively fuzzified attributes derived with the actions inducemf and classify, but for Weka regression algorithms (linreg, logreg, regtree) the aggregatemv action can also be applied to the original data. Again, the analytic variables and the target variable are indicated, which usually correspond to those variables used for MFI. As aggregation operator, either an unsupervised aggregation, such as a minimum (min), maximum (max), algebraic product (ap), algebraic sum (as), or average (avg), can be chosen or a supervised aggregation, such as a linear regression (linreg), a logistic regression (logreg), or a regression tree (regtree), can be calculated. The column that contains the aggregated membership value for each row must be given a column alias. The resulting aggregation function is written as a query template to an output file for which the path is defined in the action definition. This query template represents the multivariate model as a membership function in the target class. It can be used for data classification and prediction with the classify action.

In order to evaluate correlations between columns, the IFCL action *evaluate* can be applied. This is useful to evaluate predictive performance, but also for attribute ranking and selection. Using IFCL, an attribute selection can be accomplished by ranking the inductively fuzzified analytic variables by their correlations with the target variable. In order to do this, for all columns of the table, a membership function to the target is induced in a training set with the *inducemf* action and the columns are transformed into a membership degree with the *classify* action. Then, the correlations of all fuzzified variables and the target variable can be evaluated with the *evaluate* action.

4.1.3   IFC-Filter for Weka

The IFC-NLR data mining methodology introduced by Kaufmann and Meier (2009) has been implemented by Graf (2010) as a supervised attribute filter in the Weka machine learning workbench (Hall et al., 2009). The aim of this implementation is the application of IFC-NLR in a typical data mining process. This IFC-Filter allows the evaluation of the algorithm.

Weka can be used for data mining in order to create predictive models based on customer data. The IFC-Filter can facilitate visualization of the association between customer characteristics and the target class, and it can improve predictive performance of customer scoring by transforming attribute values into inductive fuzzy membership degrees, as previously explained. Thus, it can perform MFI

and IAF. As shown in Figure 24, the Weka software provides data access to relational databases (RDB), comma separated values (CSV), and attribute relation file format (ARFF); classification and regression; and predictions. The IFC-Filter implements the functionality for MFI, IAF, and visualizations based on the calculated membership functions.

Figure 24: Software architecture of the IFC-Filter.

The IFC-Filter transforms sharp input data into membership degrees that indicate the inductive support for the conclusion that the data record belongs to the target class. In order to do so, first, membership functions are induced from data and optionally displayed to the screen. Then, these functions are applied to fuzzify the original attributes. Visualization and prediction based on the concepts of MFI and IAF are two main uses for the IFC-Filter software in the data mining process.

In order to visualize associations between variables, inductive membership functions can be plotted with the method IFC-NLR described earlier. Thus, for every analytic variable, a function mapping from the variable's domain into a degree of membership in the inductive fuzzy target class is displayed graphically. This plot gives intuitive

100

insights about associations between attribute values and the target class.

For prediction, the transformation of crisp attribute values from a data source into inductive membership degrees can enhance the performance of existing classification algorithms. The IFC-Filter transforms the original attribute values into inductive membership indicating target class likelihood based on the original value. After that, a classical prediction algorithm, such as logistic regression, can be applied to the transformed data and to the original data in order to compare the performance of IAF. It is possible and likely that IAF improves prediction. When that is the case, IAF data transformation can be applied to huge data volumes in relational databases using the SQL code generated by the IFC-Filter. The software is available for download[3] in source and binary format for researchers and practitioners for experimenting and practicing IFC.

Figure 25: Knowledge flow with IFC-Filters.

Adapted from "Fuzzy Target Groups in Analytic Customer Relationship Management," by M. Kaufmann and C. Graf, 2012, In A. Meier and L. Donzé (Eds.), Fuzzy Methods for Customer Relationship Management and Marketing, p. 186. Copyright 2012 by Publisher.

As proposed in the section on application of IFC to analytics (Section 3.1) the IFC-Filter can visualize membership functions that indicate target membership likelihoods. Figure 25 shows two screenshots of the membership function plots for the variables *Duration* and *Checking status* from the German Credit dataset. The IFC-Filter has the possibility to activate a frame containing a graphical

---

[3] http://diuf.unifr.ch/is/ifc (accessed 11.2010)

illustration of the resulting membership functions. Each analytical variable is represented by a tab in this frame. The presentation of numerical analytical variables differs slightly from that of categorical analytical variables, because it presents a continuous membership function.

As shown in Figure 26, the illustration of numerical analytical variables consists of four fields. The first field is a table containing the NLRs with the corresponding quantiles and average quantile values (AQVs). The second field shows a histogram containing the NLRs and their corresponding AQVs. The third field shows the membership function of the analytical variable. The fourth field shows the membership function in SQL syntax for this particular analytical variable, which can be used directly in a relational database for fuzzy classification of variables.

The illustration of categorical analytical variables can be reduced to three fields. The first field is a table containing the NLR with the corresponding quantile and average value of the quantile. The second field shows a histogram containing the NLRs corresponding to the categorical values. The third field shows the membership function in SQL syntax.

An additional tab, the SQL panel, contains the membership functions of all analytic variables in SQL. It displays a concatenation of the membership functions in SQL syntax for all analytical variables that have been input for MFI by the IFC-Filter. This database script can be applied in a database in order to transform large database tables into inductive degrees of membership in a target class. This can improve the predictive quality of multivariate models.

Inductive membership functions for target class { x | GOOD_CREDIT_RATING (x) = 1 }

CHECKING_STATUS	CREDIT_HISTORY	PURPOSE	SAVINGS_ACCOUNTBONDS	PRESENT_EMPLOYMENT_SINCE	PERSONAL_STATUS_AND_SEX	OTHER_DEBTORS__GUARANTORS	
INSTALLMENT_RATE_IN_PERCENTAGE		PRESENT_RESIDENCE_SINCE		AGE_IN_YEARS	NUMBER_OF_EXISTING_CREDITS_AT	NUMBER_OF_PEOPLE_BEING_LIABLE	SQL panel
PROPERTY		OTHER_INSTALLMENT_PLANS		HOUSING	JOB	TELEPHONE	FOREIGN_WORKER	DURATION	CREDIT_AMOUNT

Quantile	Average qu...	NLR
1.0	8.15	0.733050847...
2.0	12.555	0.5625
3.0	18.505	0.470930232...
4.0	24.655	0.488207547...
5.0	40.65	0.317073170...

NLR and average values of the quantile

Inductive membership function of DURATION in target class { x | GOOD_CREDIT_RATING (x) = 1 }

Membership function in SQL syntax

```
case
when DURATION < 8.15 then DURATION * -0.03871755901421729 + 1.048598953423498
when 8.15 <= DURATION < 12.555 then DURATION * -0.03871755901421729 + 1.048598953423498
when 12.555 <= DURATION < 18.505 then DURATION * -0.015389876880984943 + 0.755719904240766
when 18.505 <= DURATION < 24.655 then DURATION * 0.0028093194490523355 + 0.41894377615342615
when 24.655 <= DURATION < 40.65 then DURATION * -0.010699242040519169 + 0.7519973596788116
when DURATION >= 40.65 then DURATION * -0.010699242040519169 + 0.7519973596788116
end
```

Inductive membership functions for target class { x | GOOD_CREDIT_RATING (x) = 1 }

INSTALLMENT_RATE_IN_PERCENTAGE		PRESENT_RESIDENCE_SINCE		AGE_IN_YEARS	NUMBER_OF_EXISTING_CREDITS_AT	NUMBER_OF_PEOPLE_BEING_LIABLE	SQL panel
PROPERTY		OTHER_INSTALLMENT_PLANS		HOUSING	JOB	TELEPHONE	FOREIGN_WORKER	DURATION	CREDIT_AMOUNT
CHECKING_STATUS	CREDIT_HISTORY	PURPOSE	SAVINGS_ACCOUNTBONDS	PRESENT_EMPLOYMENT_SINCE	PERSONAL_STATUS_AND_SEX	OTHER_DEBTORS__GUARANTORS	

Value	NLR
...<0DM	0.30616740088105726
...>=200DM	0.6000000000000001
0<=...<200DM	0.40097799511002447
no_checking_account	0.7642752562225474

Inductive membership degrees of CHECKING_STATUS in target class { x | GOOD_CREDIT_RATING (x) = 1}

Membership function in SQL syntax

```
case
when CHECKING_STATUS = ' ...<0DM' then 0.30616740088105726
when CHECKING_STATUS = ' ...>=200DM' then 0.6000000000000000
when CHECKING_STATUS = ' 0<=...<200DM' then 0.40097799511002447
when CHECKING_STATUS = ' no_checking_account' then 0.7642752562225474
end
```
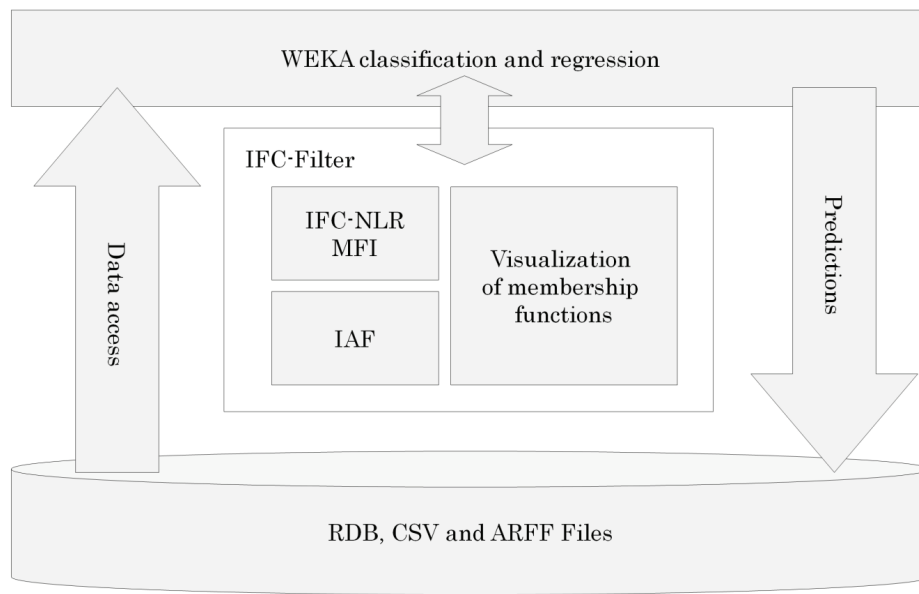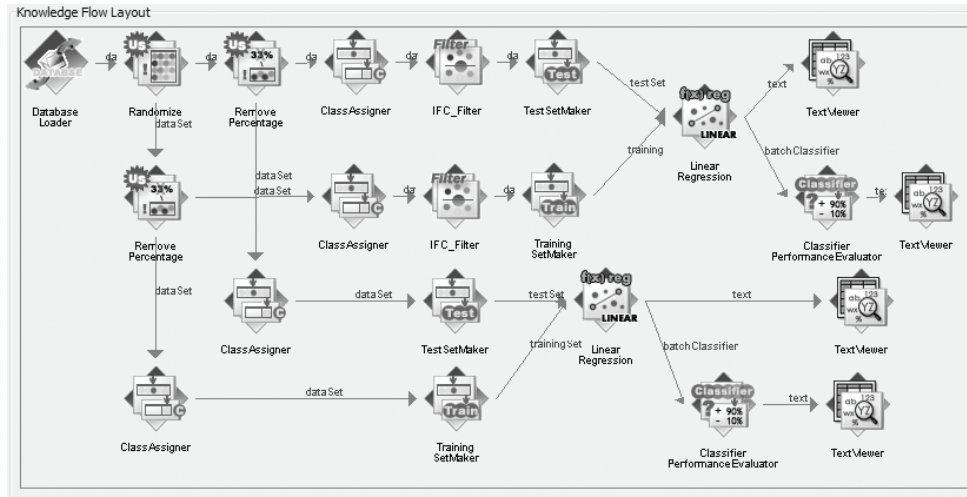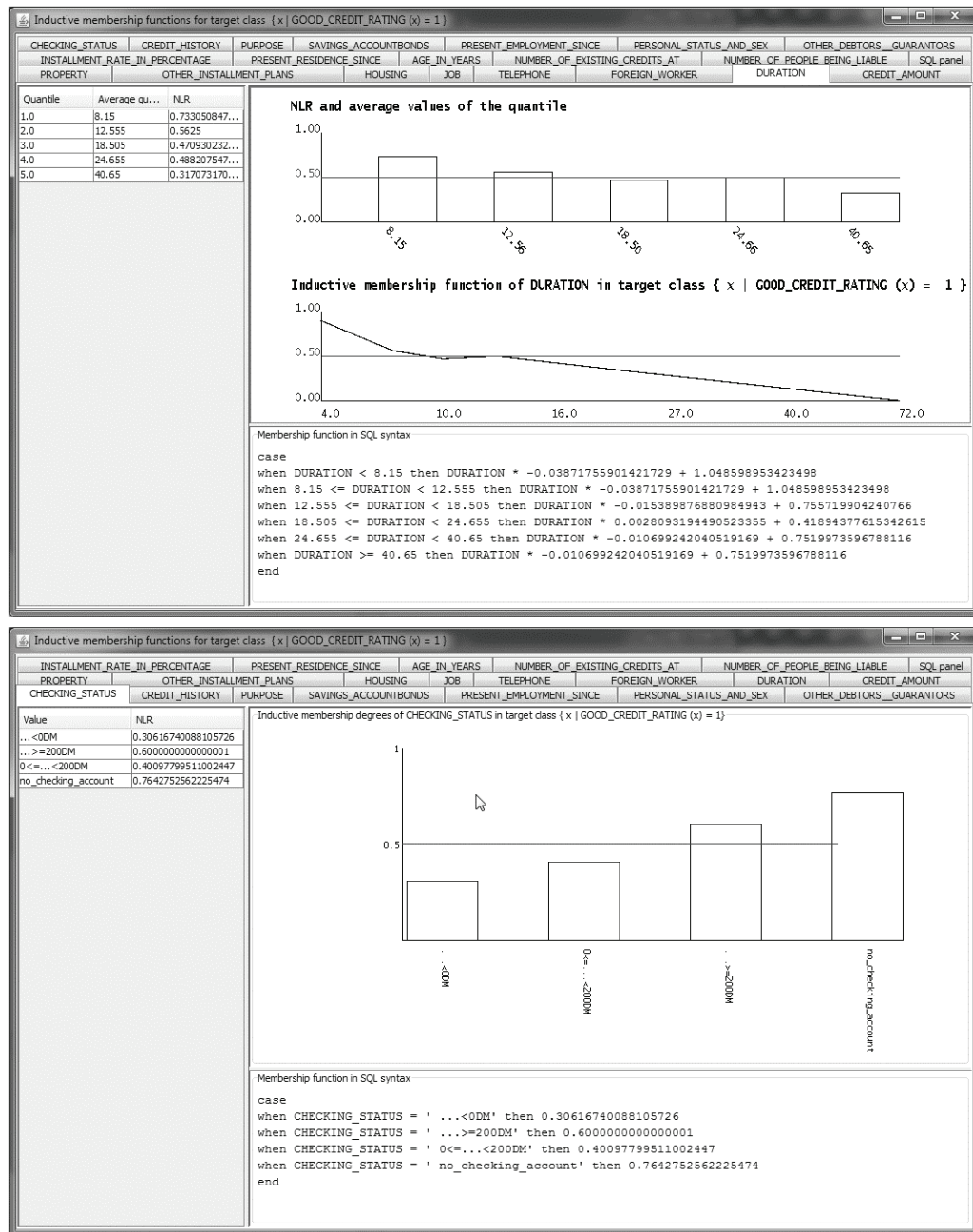
Figure 26: Visualization of membership functions with the IFC-Filter.

Adapted from "Fuzzy Target Groups in Analytic Customer Relationship Management," by M. Kaufmann and C. Graf, 2012, In A. Meier and L. Donzé (Eds.), Fuzzy Methods for Customer Relationship Management and Marketing, p. 188. Copyright 2012 by Publisher.

4.2   Empirical Tests

This section describes the experiments that have been conducted using the IFCL prototype (Section 4.1.2). Sixty real-world datasets (see the appendix in Section A.1) with categorical and numerical target attributes have been used to build predictive models with different parameters. The results of these experiments allow the answering of the following research questions: Which is a suitable MFI method for the application of fuzzy classification for prediction? Which aggregation operators are suitable for the multivariate combination of fuzzy classes? Can it be statistically supported that this method of IFC leads to higher predictive performance?

In order to find the answers to these research questions, the aim is to inductively infer conclusions about inductive methods, which is called a meta-induction scheme. Thus, quantitative statistical methods are applied to the data on the performance of IFC using different parameters gained by running the automated prediction experiments.

4.2.1   Experiment Design

The aim of these experiments was to determine the optimal parameters for a maximal predictive performance in order to gain insights about IFC. Using the scripting functionality of the IFCL prototype, a systematic benchmarking of predictive performance of different IFC algorithms was conducted. From the UCI Machine Learning Repository,[4] Weka numeric datasets archive,[5] and Weka UCI datasets archive,[6] 60 statistical datasets were used for systematic automated prediction experiments using different parameters. These datasets are listed in Table 9 and Table 10 in Section A.1. They contain real-world data from different processes that can be used for testing machine learning algorithms. They have been chosen by their tabular structure and by the type of the prediction target. Apart from that, they were chosen randomly.

For categorical target classes, a Boolean target variable was extracted with a simple predicate (see Table 9, column Target). For numerical target classes, a Zadehan target variable was generated using inductive target fuzzification using both linear and percentile fuzzification (see Table 10).

---

[4] http://archive.ics.uci.edu/ml/ (accessed 03.2010)

[5] http://sourceforge.net/projects/WEKA/files/datasets/datasets-numeric.jar/datasets-numeric.jar/download (accessed 08.2010)

[6] http://sourceforge.net/projects/WEKA/files/datasets/datasets-UCI.jar/datasets-UCI.jar/download (accessed 08.2010)

The experimental parameters included 13 methods for IAF (Table 1), two types of base target classes (numerical or categorical), two types of inductive target fuzzification for numerical target variables (ITF; Formula 43 and Formula 44), and eight methods for multivariate aggregation (Table 6). As benchmarks, three state of the art prediction methods (regression trees, linear and logistic regression) were applied for prediction without prior IAF.

Table 6
Different Aggregation Methods used for Prediction Experiments

Abbreviation	Description
logreg	Logistic regression
regtree	Regression trees
linreg	Linear regression
avg	Average membership degree
min	Minimum membership degree
ap	Algebraic product
as	Algebraic sum
max	Maximum membership degree

In order to test predictive performance, for every combination of dataset, IAF method, ITF method, and aggregation method, the correlation of the prediction with the actual class value was computed. This has been done using a repeated hold-out cross-validation method: For every parameter combination, the correlation result was calculated 10 times with different random splits of 66.7% training data versus 33.3% test data, and the results were averaged.

Prediction quality of model instances was measured with the Pearson correlation coefficient (Weisstein, 2010b) between the (Boolean or Zadehan) predictive and target variables in order to measure the proximity of predictions and targets in two-dimensional space. Significance of prediction correlation for experiment parameters θ_1 and θ_2 was estimated with a non-parametric statistical test, the one-sided Wilcoxon Signed-Rank (WSR) test (Weisstein, 2012a), where the null hypothesis, H_0, supposes that the average prediction correlation, $\gamma(\theta_1) := avg(corr(Y',Y) \mid \theta_1)$, of experiments with parameter θ_1 is smaller or equal to $\gamma(\theta_2)$, where Y is the target variable, Y' is the predictive model output, and both Y and Y' are Boolean or Zadehan variables with a range of truth values within $[0,1]$. The statistical test hypothesis is formalized by H_0: $\gamma(\theta_1) \leq \gamma(\theta_2)$. If the corresponding p-value returned by WSR is significantly small, it is likely that θ_1 performs better than θ_2.

In addition to their statistical meaningfulness, the reason for using these measures, Pearson correlation and WSR, is that both are

precompiled functions in certain database systems, and thus computed very efficiently for large numbers of experiments.

4.2.2 Results

This subsection describes the results of the meta-inductive prediction experiments. It describes the findings about the best performing methods for MFI and multivariate class aggregation. Additionally, it evaluates which dataset parameters correlate with the performance improvement gained by IAF.

In order to evaluate the best performing aggregation method, the average correlation between prediction and actual value of the target variable has been calculated for every aggregation method for categorical as well as numerical targets over all datasets and MFI methods. In Figure 27, the result of this evaluation for categorical variables is shown. Clearly, logistic regression is the best aggregation method for binary targets with an average correlation of 0.71.[7]

	logreg	regtree	linreg	avg	min	ap	as	max
☐ Average Correlation	0.71	0.69	0.69	0.64	0.53	0.52	0.45	0.45
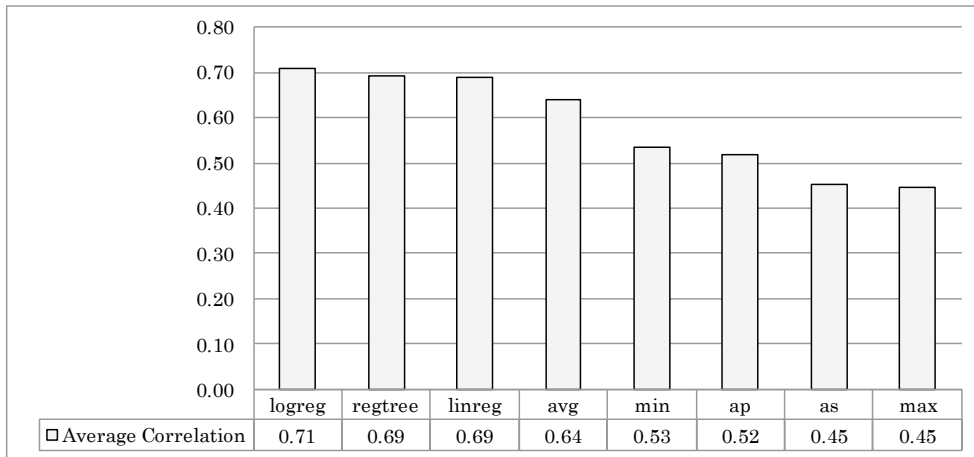
Figure 27: Average prediction correlation for different aggregation methods for binary targets.

A Wilcoxon Signed Rank (WSR)[8] test returned a p-value of 0.0115 for the hypothesis that the average correlation between prediction and target for logistic regression is smaller than or equal to that of regression trees. This means that, with a high probability, the predictive performance of logistic regression is higher than that of regression tree. The p-value for the difference in performance between logistic and linear regression is even smaller: 0.00105. Therefore,

---

[7] See Query 2 in Chapter A
[8] See Query 3 and data in Table 11 in Chapter A

logistic regression performs significantly better as an aggregation method for prediction than the other methods.

As illustrated by Figure 28, the best performing method of multivariate class aggregation for numerical targets is regression trees with an average correlation of 0.66.[9] The WSR significance test[10] showed, with a p-value of 1.30152E-06, that the average correlation of regression trees is smaller than or equal to the average correlation of linear regression. Again, this result is, statistically, very significant.

	regtree	linreg	avg	logreg	ap	max	min	as
□ Average Correlation	0.66	0.58	0.52	0.52	0.45	0.42	0.41	0.39
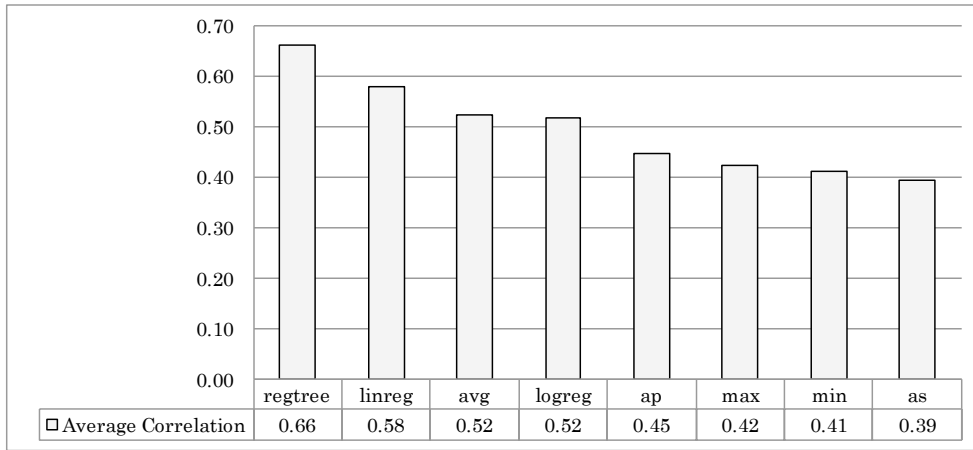
Figure 28: Average prediction correlation for different aggregation methods for fuzzified numerical targets.

In order to determine the best performing combination of membership induction method for IAF and multivariate aggregation method, the average correlation between prediction and actual value of the target variable has been calculated for every combination. As shown in Figure 29, the best combination for binary target variables is logistic regression, with an NLR as MFI method for IAF, with an average prediction correlation of 0.71.[11] The best benchmark method without IAF is regression trees on rank 24 of 107 with an average prediction correlation of 0.699. The top 23 combinations all involved a form of IAF. The best performing method was to combine an IAF using NLR with a logistic regression.

The WSR significance test returned a p-value of 0.1800[12] for the hypothesis that the average correlation between prediction and target for logistic regression with NLR is smaller than or equal to that of logistic regression with IAF using NLD. This means that, for logistic

---

[9] See Query 7 in Chapter A

[10] See Query 8 and data in Table 12 in Chapter A

[11] See Query 4 in Chapter A

[12] See Query 5 and data in Table 14 in Chapter A

regression with IAF, NLR does not perform significantly better than NLD. However, the WSR showed, with a p-value of 0.04888, that a logistic regression with IAF using NLR performs equally or less well compared to regression trees without IAF. Consequently, a significant performance improvement of logistic regression models can be achieved, on average, with IAF.

	logreg(nlr): Rank 1	logreg(nld): Rank 2	logreg(cp): Rank 3	logreg(npd): Rank 4	logreg(npdu) : Rank 5	regtree(no IAF): Rank 24	logreg(no IAF): Rank 30	linreg(no IAF): Rank 44
☐ Average Correlation	0.712	0.710	0.710	0.709	0.708	0.699	0.688	0.669

Figure 29: Average prediction correlation for combinations of membership function induction and aggregation methods for binary targets.

	Linear	Percentile
☐ Average Correlation	0.69	0.64
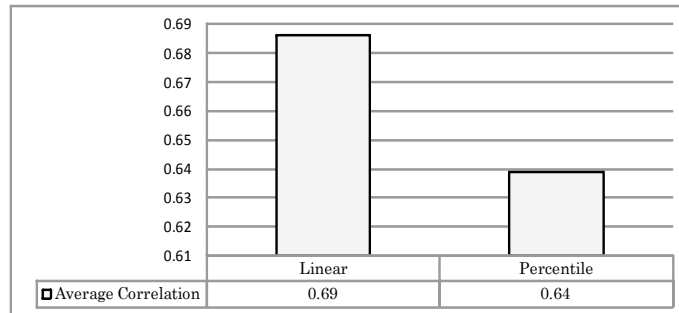
Figure 30: Average prediction correlation of predictions with linear versus percentile ITF, given regression trees as the aggregation method.

In order to apply MFI methods from Section 2.4, numerical target variables need to be fuzzified. As proposed in this section, two approaches for inductive target fuzzification (ITF) have been proposed. Both were tested and compared in the experiments. In Figure 30, one can see that the average correlation between prediction and target was much higher for linear ITF.[13] The WSR significance test[14] returned a p-value of 0.0000289. Thus, linear target fuzzification performs significantly better for predicting fuzzy classes than percentile target fuzzification.

---

[13] See Query 9 in Chapter A
[14] See Query 10 and data in Table 13 in Chapter A

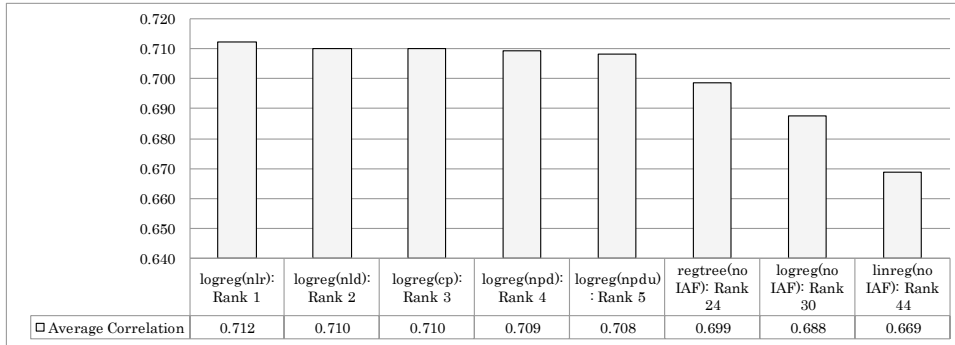|	regtree(nld): Rank 1	regtree(nldu): Rank 2	regtree(nc): Rank 3	regtree(npdu): Rank 4	regtree(npr): Rank 5	regtree(no IAF): Rank 14	linreg(no IAF): Rank 25	logreg(no IAF): Rank 38
▢ Average Correlation	0.697	0.695	0.690	0.690	0.689	0.621	0.591	0.569

Figure 31: Average prediction correlation for combinations of membership function induction and aggregation methods for fuzzified numerical targets.
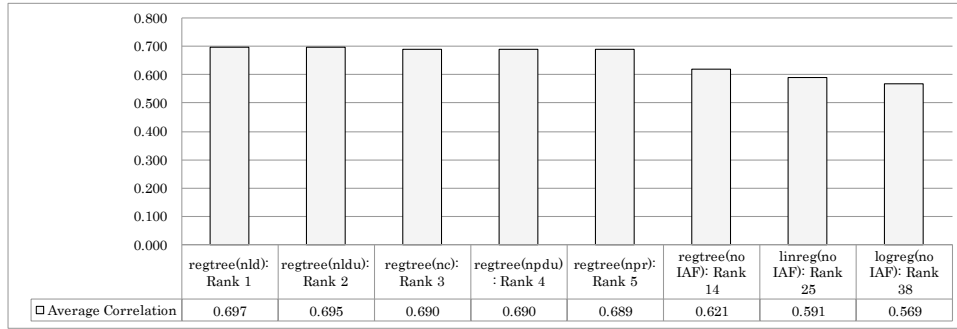
In Figure 31, the results for experiments with combinations of different IAF and aggregation methods for data with numerical targets[15] are shown, given linear ITF of numerical target variables. In this case, IAF with NLD, together with a supervised aggregation using Weka M5P regression trees (regtree), is the best method with an average correlation of 0.697. The difference between this combination and the benchmark prediction using regression trees without IAF is significant with a WSR p-value[16] of 0.0026.

There are several parameters that distinguish datasets. These include number of variables, number and percentage of categorical or numerical variables, number of rows, ratio between number of rows and number of columns, and the linearity of the prediction target. This linearity was measured by the average correlation of the predictions of a linear regression benchmark model, Y_{LR}, with the target variable Y: $Linearity := avg(corr(Y_{LR}, Y))$. The question is whether any of these parameters correlate significantly with the improvement of predictive performance of regression models with IAF.

In order to answer this question for binary target predictions, correlation and level of significance of correlation were calculated, using a non-parametric correlation test, between the parameters and the relative improvement of the predictive performance of logistic regression if the analytic variables are fuzzified using NLR versus logistic regression without IAF.[17] As the statistical test, Spearman's rho (Weisstein, 2012b), with a z-score significance test was applied, which exists as a precompiled function in the database.[18]

---

[15] See Query 11 in Chapter A

[16] See Query 12 and data in Table 15 in Chapter A

[17] See Query 14 in Chapter A

[18] http://docs.oracle.com/cd/B28359_01/server.111/b28286/functions029.htm (6/2012)

Table 7

Correlation and p-Value of Dataset Parameters and Improvement of Logistic Regression with IAF for Binary Target Variables

Parameter	Correlation with Improvement with IAF	p-value
Number of categorical variables	0.335516011	0.069899788
Number of numeric variables	-0.097871137	0.60688689
Percentage of categorical variables	0.206624854	0.273290146
Percentage of numeric variables	-0.206624854	0.273290146
Number of columns	0.228400654	0.224758897
Average correlation of the prediction of linear regression	*-0.50567297*	*0.004362976*
Number of rows (n)	0.089665147	0.637498617
Ratio number of rows / number of columns	-0.001557286	0.993483625
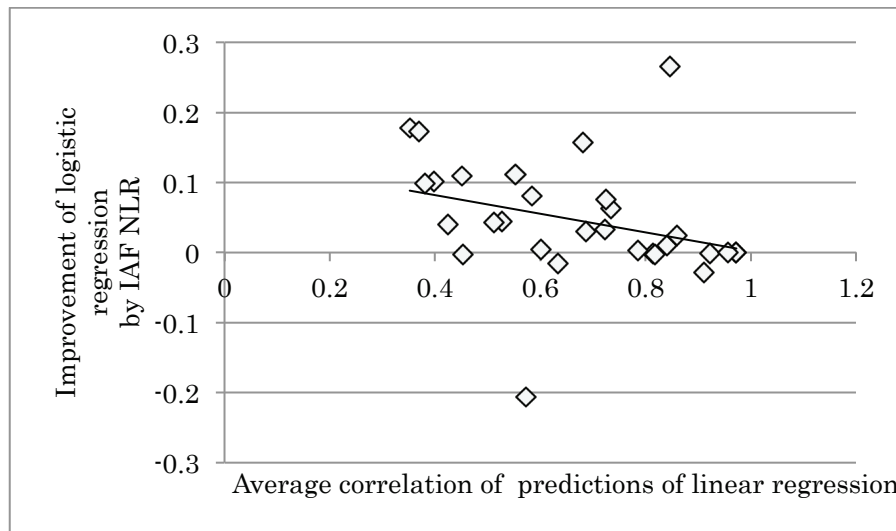
Figure 32: Relationship between target linearity and IAF benefit for binary target variables.

As shown in Table 7, the correlation between target linearity and improvement by NLR fuzzification is negative and significant ($p < 0.005$). This means that the less linear the connection between the analytic variables and the target variable is, the more a Boolean prediction using logistic regression can be improved with an IAF using NLRs. In Figure 32, this association is visualized in a scatter plot.[19]

---

[19] See Query 15 and data in Table 16 in Chapter A

For fuzzified numerical prediction targets, Table 8 shows similar results.[20] The improvement of regression trees with IAF NLD correlates negatively and significantly with the linearity of association between analytic and target variables. The lower the performance of linear regression, the more a predictive regression tree model can benefit from IAF using NLD. This association is illustrated[21] in Figure 33.

Table 8

Correlation and p-Value of Dataset Parameters and Improvement of Regression Trees with IAF for Zadean Target Variables

Parameter	Correlation w/ Improvement by IAF	p-value
Number of categorical variables	0.121461217	0.522578264
Number of numeric variables	0.03610839	0.84975457
Percentage of categorical variables	0.036497846	0.848152597
Percentage of numeric variables	-0.036497846	0.848152597
Number of columns	0.195910119	0.299481536
Average correlation of the prediction of linear regression	*-0.583537264*	*0.000712151*
Number of rows (n)	0.081432863	0.668807278
Ratio number of rows / number of columns	-0.043381535	0.81994093
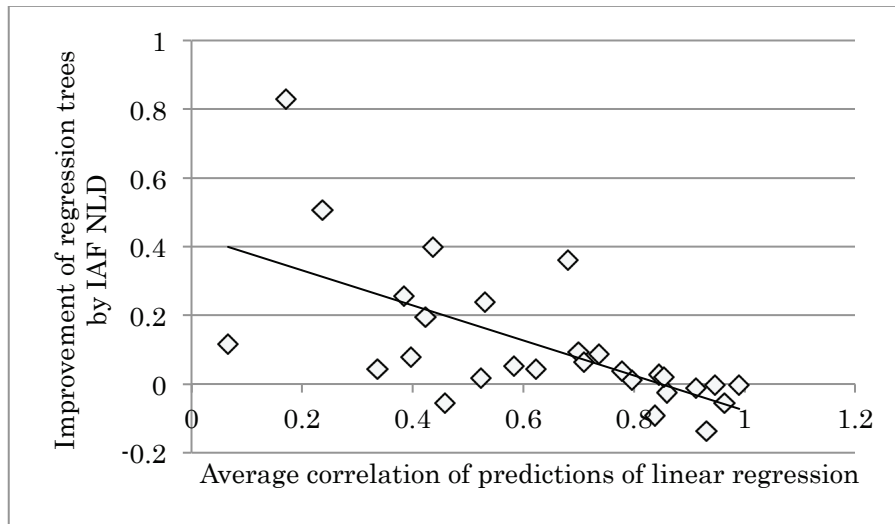
Figure 33: Relationship between target linearity and IAF benefit for fuzzified numerical target variables.

---

[20] See Query 14 in Chapter A
[21] See Query 15 and data in Table 17 in Chapter A
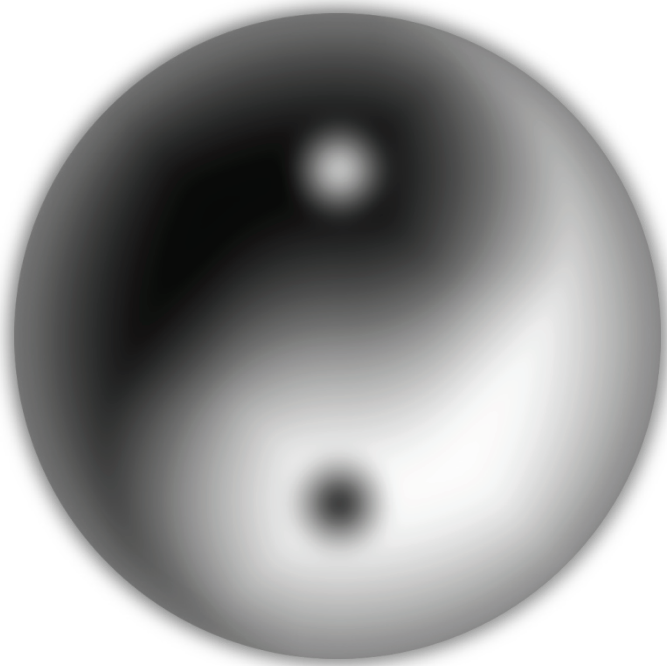
"Every species is vague, every term goes cloudy at its edges, and so in my way of thinking, relentless logic is only another name for stupidity—for a sort of intellectual pigheadedness. If you push a philosophical or metaphysical enquiry through a series of valid syllogisms—never committing any generally recognized fallacy—you nevertheless leave behind you at each step a certain rubbing and marginal loss of objective truth and you get deflections that are difficult to trace, at each phase in the process. Every species waggles about in its definition, every tool is a little loose in its handle, every scale has its individual." (H. G. Wells, 1908)

5 Precisiating Fuzziness by Induction

5.1 Summary of Contribution

A methodology for IFC has been introduced. This method allows automated derivation of membership functions from data. The idea is to translate sampled probabilities into fuzzy restrictions. Thus, a conceptual switch from the probabilistic to the possibilistic view is applied in order to reason about the data in terms of fuzzy logic. Business applications of MFI have been proposed in the area of marketing analytics in the fields of customer and product analytics, fuzzy target groups, and integrated analytics for individual marketing. A prototype inductive fuzzy classification language (IFCL) has been implemented and applied in experiments with 60 datasets with categorical and numerical target variables in order to evaluate performance improvements by application of the proposed methodology. Seven questions (Section 1.1) have guided the research conducted for this thesis. The following list summarizes the answers that this thesis proposes:

o   What is the theoretical basis of inductive fuzzy classification (IFC) and what is its relation to inductive logic?

> Inductive logic is defined as "a system of evidential support that extends deductive logic to less-than-certain inferences" (Hawthorne, 2008, ""Inductive Logic," paragraph 1). Fuzzy classes are fuzzy sets (Zadeh 1965) defined by propositional functions (Russell, 1919) with a gradual truth-value (i.e., mapping to fuzzy propositions; Zadeh 1975). Their membership functions are thus inductive if their defining propositional functions are based on less-than-certain inferences (Section 2.4). The inductive fuzzy class $y' = \{\, i \in U \mid i$ is likely a member of $y\,\}$ is defined by a predictive model, μ, for membership in class y, where "likely" is a fuzzy set of hypotheses. The truth function of the fuzzy propositional function $L(i,y) := "i$ is likely a member of $y"$ is a fuzzy restriction (Zadeh, 1975a) on U defined by $\mu_{y'}$.

o   How can membership functions be derived inductively from data?

> Membership functions can be derived using ratios or differences of sampled conditional probabilities. Different formulas have been proposed in Section 2.4.1 and tested in Section 4.2 Normalized differences and ratios of likelihoods have shown to provide optimal membership degrees. For numeric variables, a continuous membership function can be defined by piecewise

linear function approximation. The numeric variable is discretized using quantiles and a linear function is approximated to the average values and the membership degrees for every adjacent pair of quantiles. For numerical target variables, normalization leads to a linear fuzzification with a membership degree to a fuzzy set that can be used to calculate likelihoods of fuzzy hypotheses.

o   How can a business enterprise apply IFC in practice?

Inductive fuzzy classes can support marketing analytics by providing selection, visualization, and prediction in the fields of customer analytics, product analytics, fuzzy target selection, and integrated analytics for individual marketing. These techniques improve decision support for marketing by generating visual, intuitive profiles and improved predictive models, which can enhance sales (Section 3.2). A case study shows a real-world application with a financial service provider (Section 3.2.5).

o   How can the proposed methods be implemented in a computer program?

The IFCL prototype uses a database server for data storage, access, and computation (Section 4.1.2). The analytic process is configured in a text file using a synthetic language called IFCL. This language allows definition of all computations necessary for MFI, IAF, target fuzzification, attribute selection, supervised multivariate aggregation, fuzzy classification of data using predictive models, and model evaluation. Additionally, the IFCL prototype allows experiments on the proposed methods for MFI in batch mode. Furthermore, student works have shown implementations of IFC with graphical user interfaces.

o   How is IFC optimally applied for prediction?

First, data is prepared and relevant attributes are selected. Then, each attribute is inductively fuzzified, so that the membership degree indicates the degree of membership in the target class. Finally, those multiple membership degrees are aggregated into a single membership degree with a multivariate regression, which represents the inductive class prediction. Optimally, for binary targets, attributes are fuzzified with NLRs and aggregated using logistic regression. For numerical targets, the target attribute is fuzzified using a linear normalization, the dependent attributes are fuzzified using NLD and then aggregated using regression trees (Section 3.1.3).

o Which aggregation methods are optimal for the multivariate combination of fuzzy classes for prediction?

> In the experimental phase, product, algebraic sum, minimum, maximum, average, logistic regression, linear regression, and regression trees have been tested for their predictive performance using the prototype software IFCL. For the aggregation of fuzzified attributes into a single prediction value for data records, the experiments showed that, on average, a supervised aggregation using multivariate regression models (logistic regression for Boolean targets and regression trees for Zadehan targets) are optimal (Section 4.2.2).

o Can it be statistically supported that the proposed method of IFC improves predictive performance?

> Inductive attribute fuzzification using NLRs significantly improved average prediction results of classical regression models (Section 4.2.2). In fact, this improvement significantly and negatively correlated with the linearity of association between the data and the target class. Thus, the less linear this association is, demonstrated by a smaller correlation of the predictions of a linear regression model with the actual values, the more a multivariate model can be improved by transforming attributes into an inductive membership degree using the methods proposed in this thesis.

5.2 Discussion of Results

Human perception and cognition is blurred by uncertainty and fuzziness. Concepts in our minds cluster perceptions into categories that cannot always be sharply distinguished. In some cases, binary classifications are feasible, such as detecting the presence or absence of light. Nevertheless, in many cases, our classification relies on fuzzy terms such as *bright*. Some concepts are gradual and can be ordered by their degree. In our previous example, brightness is a perception that is ordinal because some visions are brighter than others. By mapping an ordinal scale to Zadehan truth values, a membership function can be defined in order to clearly *precisiate* the semantics of fuzzy words, concepts, or terms, in the sense of Zadeh (2008). In fact, fuzzy set theory has not yet been recognized enough for its most valuable contribution: It provides a tool for assigning *precise definitions* to fuzzy statements! In Section 2.2.1, the sorites paradox, an age-old philosophical puzzle, was easily resolved by application of fuzzy set theory; such is the power of Zadehan logic.

Conditioned by fuzziness of cognition, the human mind infers and reasons based on fuzzy ideas, and often, successfully so. Induction is an uncertain type of reasoning, and thus classifies inferences into a fuzzy set: the fuzzy class of inductively supported conclusions. On one hand, induction is always uncertain, and even the most unlikely event can actually happen, even if induction indicates the contrary, such as the observation of a black swan (Taleb, 2007). On the other hand, how can we learn if not from past experience? Good inductive inferences are reliable more often than not. Furthermore, gaining insights by evaluating likelihoods of hypotheses using objective data is a cornerstone of empiricism. Thus, induction and its support measure, the likelihood ratio, can amplify past experience by leading to better and more reliable inductive inferences. While many concepts are fuzzy, induction provides a tool for precisiation.

The transformation of likelihood ratios into membership functions to fuzzy sets by normalization is a tool for reasoning about uncertainty by amplification of past experience, where the fuzziness of class membership is precisiated by likelihood in the data. Accordingly, MFI with NLRs applies the law of likelihood and fuzzy set theory to make inductive inferences and to represent them as membership functions to fuzzy classes.

There are three main advantages of applying likelihood-based MFI in analytics. First, a membership function makes hidden meaningful associations between attributes in the data explicit. It allows visualizing these associations in order to provide insight into automated inductive inferences. Second, a two-dimensional plot of these membership functions is easily interpretable by humans. Those visualizations are precisiations of inductive associations. Third, these membership functions are models that allow for a better quality of predictions and thus more accurate decisions.

Likelihood-based precisiation of fuzziness can be applied to marketing analytics. Data analysis on customers, products, and transactions can be applied for customer relationship management (CRM) and individual marketing. This kind of analytic CRM provides decision support to company representatives in marketing channels with direct customer contact for offers, underwriting, sales, communications, gifts, and benefits.

For the analytics process, it is proposed to classify customers inductively and fuzzily in order to target individual customers with the right relationship activities. First, the customers are classified by assigning them to different targets (classes), such as product offers or promotional gifts. Second, this classification is done inductively by analyzing present data as evidence for or against target membership based on customer characteristics. Third, the assignment of individual

customers to targets is fuzzy if the resulting classes have a gradual membership function. Finally, those inductive membership degrees to fuzzy targets provide decision support by indicating the likelihood of desired response in customer activities based on customer characteristics. These membership functions can be used in analytics to select the most important characteristics regarding a target, visualize the relationship between characteristics and a target, and predict target membership for individual customers.

The proposed method for MFI is based on calculating likelihoods of target membership in existing data. These likelihoods are turned into a fuzzy set membership function by comparison and normalization using an NLR or an NLD. This algorithm has been implemented in the Weka machine learning workbench as a supervised attribute filter, which inductively fuzzifies sharp data and visualizes the induced membership functions. This software can be downloaded for experimental purposes.

Sharp binary classification and segmentation lead to anomalies of threshold, and fuzzy classification provides a solution with gradual distinctions that can be applied to CRM, as pointed out by Werro (2008). Gradual customer partitioning has the advantage that the size of target groups can be varied by choosing a threshold according to conditions such as budget or profitability. Of course, scoring methods for CRM are state of the art; additionally, the present approach suggests that fuzzy logic is the appropriate tool for reasoning about CRM targets because they are essentially fuzzy concepts. Scoring provides a means for precisiation of this fuzziness, which provides numerical membership degrees of customers in fuzzy target groups.

The advantages of MFI for fuzzy classification of customers are efficiency and precision. First, membership functions to CRM targets can be derived automatically, which is an efficient way of definition. And second, those inferred membership functions are precise because they indicate an objective, measurable degree of likelihood for target membership. Thus, the semantics of the membership function is clearly grounded.

5.3   Outlook and Further Research

The first point of further research is further development of the IFCL software. This prototype has fulfilled its purpose in answering the research questions of this thesis, but it lacks many features to make it suitable for productive use. Further development of this software could address the following points:

o refactoring of the source code with a clear object oriented design methodology,

o adding a graphical user interface,

o redesign of certain syntactical elements that have turned out to be inconvenient,

o adding a syntax checker for the IFCL language,

o adding support for all databases with a JDBC interface,

o handling null values in the data,

o and allowing multiclass predictions.

As a second research direction, the field of application of IFC will be diversified. So far, applications have been proposed and evaluated in the area of marketing. Providing applications for web semantics analytics and biomedical analytics is envisioned.

Third, the logical framework of fuzzy classification by induction is intended to be generalized. *"Inductive approximate reasoning"* can be defined as a methodology for the application of logic with gradual truth-values using less-than-certain inferences. Also, the epistemological problem of induction will be tackled, because the likelihood ratio as a measure of support is distorted for large values of $p(\overline{H} \cap \overline{E})/p(H)$, which is exemplified by the paradox of the raven (Vickers, 2009): Is the conclusion that a Raven (E) is black (H) truly supported by countless observations of white clouds ($\overline{H} \cap \overline{E}$)? At least, it increases the likelihood ratio by decreasing $p(E|\overline{H})$; perhaps, there might be other useful measures of inductive support.

The most important suggestion for further research is the investigation of multidimensional MFI. This thesis is concentrated on induction of membership functions for single variables and their multivariate combination. As such, not all possible combinations of all variables are evaluated (see Section 2.2.2). In a multivariate model, a given attribute value can be assigned the same weight for every combination with values of other variables, for example, in a linear regression. In a multidimensional model, the same attribute value has different weights in different contexts, because every vector of a possible attribute value combination is assigned an individual weight or membership degree. Therefore, it might be interesting for future research to investigate the properties of multidimensional MFI. Instead of a multivariate combination of univariate membership functions, a multidimensional membership function assigns a membership degree to vectors of attribute value combinations, and the influence of an attribute value is not only dependent on itself, but also on the context of the other attribute's values. Thus, predictive performance could be

improved because the interdependency of attributes can be modeled. This could resolve the "watermelon paradox" stated by Elkan (1993, p. 703), because the inductive membership function to the concept of *watermelon* is measured in two dimensions, with "red inside" as a first coordinate and "green outside" as a second.

The following research questions can be of interest: How are multidimensional membership functions induced from data? How are multidimensional membership degrees computed for vectors of categorical values, for vectors of numerical values, and for vectors of numerical and categorical values? How are continuous functions derived from those discrete degrees? Can multidimensional membership functions improve prediction? In what cases is it advisable to work with multidimensional membership functions?

A possibility is to use n-linear interpolation using n-simplexes (Danielsson & Malmquist, 1992), which are hyper-triangles or generalizations of triangles in multidimensional space. Using simplexes, for every n-tuple of possible dimension value combinations, a corresponding likelihood measure (such as an NLR) could be sampled. Then, n different membership degree samples would represent the edges of a simplex. This hyper-triangle would represent a piece of the continuous multidimensional membership function, which could be approximated by combining simplexes to piecewise linear hyper surfaces.

6 Literature References

Backus, J. W., Bauer, F. L., Green, J., Katz, C., McCarthy, J., Perlis, A. J., et al. (1960). Report on the algorithmic language ALGOL 60. (P. Naur, Ed.) *Communications of the Association for Computing Machinery , 3* (5), pp. 299-314.

Bellmann, R. E., & Zadeh, L. A. (1977). Local and Fuzzy Logics. In J. M. Dunn, & G. Epstein (Eds.), *Modern Uses of Multiple-Valued Logic* (pp. 283-335). Kluwer Academic Publishers.

Boole, G. (1847). *The Matematical Analysis of Logic, being an Essay towards a Calculus of Deductive Reasoning.* London: George Bell.

Carrol, L. (1916). *Alice's Adventures in Wonderland.* New York: Sam'l Gabriel Sons & Company.

Chmielecki, A. (1998). What is Information? *Twentieth World Congress of Philosophy.* Boston, Massachusetts USA.

Danielsson, R., & Malmquist, G. (1992). Multi-Dimensional Simplex Interpolation - An Approach to Local Models for Prediction. *Chemometrics and Intelligent Laboratory Systems , 14*, pp. 115-128.

Davenport, T. H. (2006). Competing on Analytics. *Harvard Business Review .*

Del Amo, A., Montero, J., & Cutello, V. (1999). On the Principles of Fuzzy Classification. *Proc. 18th North American Fuzzy Information Processing Society Annual Conf,* (pp. 675 – 679).

Dianhui, W., Dillon, T., & Chang, E. J. (2001). A Data Mining Approach for Fuzzy Classification Rule Generation. *IFSA World Congress and 20th NAFIPS International Conference*, (pp. 2960-2964).

Dubois, D. J., & Prade, H. (1980). *Fuzzy Sets and Systems. Theory and Applications.* New York: Academic Press.

Elkan, C. (1993). The Paradoxical Success of Fuzzy Logic. *Proceedings of the Eleventh National Conference on Artificial Intelligence*, (pp. 698–703). Washington D.C.

Glubrecht, J.-M., Oberschelp, A., & Todt, G. (1983). *Klassenlogik.* Mannheim: Wissenschaftsverlag.

Gluchowski, P., Gabriel, R., & Dittmar, C. (2008). *Management Support Systeme und Business Intelligence.* Berlin: Springer Verlag.

Graf, C. (2010). *Erweiterung des Data-Mining-Softwarepakets WEKA um induktive unscharfe Klassifikation (Master's Thesis).* Department of Informatics, University of Fribourg, Switzerland.

Greene, B. (2011). *The Hidden Reality. Parallel Universes And The Deep Laws Of The Cosmos.* New York: Random House.

Hüllermeier, E. (2005). Fuzzy Methods in Machine Learning and Data Mining: Status and Prospects. *Fuzzy Sets and Systems* , pp. 387-406.

Hájek, A. (2009). Interpretations of Probability. In E. N. Zalta (Ed.), *Stanford Encyclopedia of Philosophy.* Stanford: The Metaphysics Research Lab, Stanford University.

Hajek, P. (2006). Fuzzy Logic. In E. N. Zalta (Ed.), *Stanford Encyclopedia of Philosophy.* Stanford: The Metaphysics Research Lab, Stanford University.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA Data Mining Software: An Update. *SIGKDD Explorations , 11* (1).

Hawthorne, J. (2008). Inductive Logic. In E. N. Zalta (Ed.), *Stanford Encyclopedia of Philosophy.* Stanford: The Metaphysics Research Lab, Stanford University.

Hu, Y., Chen, R., & Tzeng, G. (2003). Finding Fuzzy Classification Rules using Data Mining Techniques. *Pattern recognition letters , 24* (1-3), pp. 509-514.

Hyde, D. (2008). Sorites Paradox. In E. N. Zalta (Ed.), *The Stanford Encyclopedia of Philosophy.* Stanford : The Metaphysics Research Lab, Stanford University.

Inmon, W. H. (2005). *Building the Data Warehouse* (fourth ed.). New York: Wiley Publishing.

John, E. G. (1998). Simplified Curve Fitting Using Speadsheet Add-ins. *International Journal of Engineering Education , 14* (5), pp. 375-380.

Joyce, J. (2003). Bayes' Theorem. In E. N. Zalta (Ed.), *Stanford Encyclopedia of Philosophy.* Stanford: The Metaphysics Research Lab, Stanford University.

Küsters, U. (2001). Data Mining Methoden: Einordung und Überblick. In H. Hippner, U. Küsters, M. Meyer, & K. Wilde (Eds.), *Handbuch Data Mining im Marketing - Knowledge Discovery in Marketing Databases* (pp. 95-130). Wiesbaden: vieweg GABLER.

Kaniza, G. (1976). Subjective contours. *Scientific American* (234), pp. 48-52.

Kaufmann, M. A. (2009). An Inductive Approach to Fuzzy Marketing Analytics. In M. H. Hamza (Ed.), *Proceedings of the 13th IASTED international conference on Artificial Intelligence and Soft Computing.* Palma, Spain.

Kaufmann, M. A. (2008). Inductive Fuzzy Classification. *Thesis Proposal* . Internal Working Paper, University of Fribourg, Switzerland.

Kaufmann, M. A., & Graf, C. (2012). Fuzzy Target Groups in Analytic Customer Relationship Management. In A. Meier, & L. Donzé (Eds.), *Fuzzy Methods for Customer Relationship Management and Marketing - Applications and Classification.* London: IGI Global.

Kaufmann, M. A., & Meier, A. (2009). An Inductive Fuzzy Classification Approach Applied To Individual Marketing. *Proc. 28th North American Fuzzy Information Processing Society Annual Conf.*

Kohavi, R., Rothleder, N. J., & Simoudis, E. (2002). Emerging Trends in Business Analytics. *Communications of the ACM , 45* (8), pp. 45-48.

Mayer, P. (2010). *Implementation of an Inductive Fuzzy Classification Interpreter. Master's Thesis.* University of Fribourg.

Meier, A., Schindler, G., & Werro, N. (2008). Fuzzy classification on relational databases. In M. Galindo (Ed.), *Handbook of Research*

on Fuzzy Information Processing in Databases (Vol. II, pp. 586-614). London: Information Science Reference.

merriam-webster.com. (2012a). *Analytics.* Retrieved from Merriam Webster Online Dictionnary: http://www.merriam-webster.com/dictionary/analytics

merriam-webster.com. (2012b). *Heap.* Retrieved from Merriam Webster Online Dictionnary: http://www.merriam-webster.com/dictionary/heap

Mill, J. S. (1843). *A System of Logic.* London: John W. Parker, West Strand.

Oberschelp, A. (1994). *Allgemeine Mengenlehre.* Mannheim: Wissenschaftsverlag.

Oesterle, H., Becker, J., Frank, U., Hess, T., Karagiannis, D., Krcmar, H., et al. (2010). Memorandum zur gestaltungsorientierten Wirtschaftsinformatik. *Multikonferenz Wirtschaftsinformatik (MKWI).* Göttingen.

Precht, R. D. (2007). *Wer bin ich - und wenn ja wie viele? Eine philosophische Reise.* München: Goldmann.

Roubos, J. A., Setnes, M., & Abonyi, J. (2003). Learning Fuzzy Classification Rules from Labeled Data. *Information Sciences—Informatics and Computer Science: An International Journal, Volume 150* (Issue 1-2).

Russell, B. (1919). *Introduction to Mathematical Philosophy.* London: George Allen & Unwin, Ltd.

Setnes, M., Kaymak, H., & van Nauta Lemke, H. R. (1998). Fuzzy Target Selection in Direct Marketing. *Proc. of the IEEE/IAFE/INFORMS Conf. On Computational Intelligence for Financial Engineering (CIFEr).*

Shannon, C. (1948). A Mathematical Theory of Communication. *The Bell Systems Technical Journal* (Vol. 27), pp. 379–423, 623–656.

Sorensen, R. (2008). Vagueness. In E. N. Zalta (Ed.), *The Stanford Encyclopedia of Philosophy.* Stanford: The Metaphysics Research Lab, Stanford University.

Sousa, J. M., Kaymak, U., & Madeira, S. (2002). A Comparative Study of Fuzzy Target Selection Methods in Direct Marketing. *Proceedings of the 2002 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE '02).*

Spais, G., & Veloutsou, C. (2005). Marketing Analytics: Managing Incomplete Information in Consumer Markets and the Contribution of Mathematics to the Accountability of Marketing Decisions. *Southeuropean Review of Business Finance & Accounting* (3), pp. 127-150.

Taleb, N. N. (2007). *The Black Swan. The Impact of the Highly Improbable.* New York: Random House.

Turban, E., Aronson, J. E., Liang, T.-P., & Sharda, R. (2007). *Decision Support and Business Intelligence Systems.* New Jersey: Pearson Education.

Vickers, J. (2009). The Problem of Induction. In E. N. Zalta (Ed.), *Stanford Encyclopedia of Philosophy.* Stanford: The Metaphysics Research Lab, Stanford University.

Wang, L., & Mendel, J. (1992). Generating Fuzzy Rules by Learning from Examples. *IEEE Transactions on Systems, Man and Cybernetics* (Issue 6), pp. 1414 – 1427.

Weisstein, E. W. (2010a). *Conditional Probability.* Retrieved from MathWorld--A Wolfram Web Resource: http://mathworld.wolfram.com/ConditionalProbability.html

Weisstein, E. W. (2010b). *Correlation Coefficient.* Retrieved from MathWorld--A Wolfram Web Resource: http://mathworld.wolfram.com/CorrelationCoefficient.html

Weisstein, E. W. (2012b). *Spearman Rank Correlation Coefficient.* From MathWorld--A Wolfram Web Resource: http://mathworld.wolfram.com/SpearmanRankCorrelationCoefficient.html

Weisstein, E. W. (2012a). *Wilcoxon Signed Rank Test.* From MathWorld--A Wolfram Web Resource.: http://mathworld.wolfram.com/WilcoxonSignedRankTest.html

Wells, H. G. (1908). *First and Last Things; A Confession Of Faith And Rule Of Life.* New York: G. P. Putnam's Sons.

Werro, N. (2008). *Fuzzy Classification of Online Customers.* PhD. Dissertation, University of Fribourg, Switzerland.

Witten, I. H., & Frank, E. (2005). *Data Mining, Practical Machine Learning Tools and Techniques* (Second ed.). San Francisco: Morgan Kaufmann Publishers.

Zadeh, L. A. (1965). Fuzzy Sets. *Information and Control* (8), pp. 338–353.

Zadeh, L. A. (1975a). Calculus of Fuzzy Restrictions. In L. A. Zadeh, K.-S. Fu, K. Tanaka, & M. Shimura (Eds.), *Fuzzy sets and Their Applications to Cognitive and Decision Processes.* New York: Academic Press.

Zadeh, L. A. (1975b). The Concept of a Linguistic Variable and its Application to Approximate Reasoning. *Information Sciences , I* (8), pp. 199–251.

Zadeh, L. A. (2008). Is There a Need for Fuzzy Logic. *Information Sciences , 178* (13).

Zimmermann, H. J. (1997). Fuzzy Data Analysis. In O. Kaynak, L. A. Zadeh, B. Turksen, & I. J. Rudas (Eds.), *Computational Intelligence: Soft Computing and Fuzzy-Neuro Integration with Applications.* Berlin: Springer-Verlag.

Zimmermann, H. J. (2000). *Practical Applications of Fuzzy Technologies.* Berlin: Springer-Verlag.

Zimmermann, H. J., & Zisno, P. (1980). Latent Connectives in Human Decision Making. *Fuzzy Sets and Systems , 37-51.*

A. Appendix

A.1. Datasets used in the Experiments

Table 9
Data Sets with Categorical Target Variable, Transformed into a Binary Target

Name	Target	Filter[22]	Records	Link (accessed 08.2010)
abalone	rings <= 9	No	2743	http://archive.ics.uci.edu/ml/datasets/Abalone
adult	class = ' >50K'	No	21623	http://archive.ics.uci.edu/ml/datasets/Adult
anneal	class <> '3'	yes	597	http://archive.ics.uci.edu/ml/datasets/Annealing
balance	class = 'L'	No	404	http://archive.ics.uci.edu/ml/datasets/Balance+Scale
bands	band_type = 'band'	Yes	237	http://archive.ics.uci.edu/ml/datasets/Cylinder+Bands
breastw	class = 'benign'	No	437	http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)
car	acceptable <> 'unacc'	No	1145	http://archive.ics.uci.edu/ml/datasets/Car+Evaluation
cmc	contraceptive_method <> '1'	No	1002	http://archive.ics.uci.edu/ml/datasets/Contraceptive+Method+Choice
credit	a16 = '+'	No	431	http://archive.ics.uci.edu/ml/datasets/Credit+Approval
creditg	class = 'good'	No	679	http://archive.ics.uci.edu/ml/datasets/Statlog+(German+Credit+Data)
diabetes	class <> 'tested_positive'	No	489	http://archive.ics.uci.edu/ml/datasets/Pima+Indians+Diabetes
diagnosis	inflammation = 'yes'	No	87	http://archive.ics.uci.edu/ml/datasets/Acute+Inflammation
glass	type in ('"build wind float"', '"build wind non-float"', '"vehic wind float"', '"vehic wind non-float"')	No	157	http://archive.ics.uci.edu/ml/datasets/Glass+Identification
heart	y= '2'	No	188	http://archive.ics.uci.edu/ml/datasets/Statlog+(Heart)
hepatitis	class = 'LIVE'	No	59	http://archive.ics.uci.edu/ml/datasets/Hepatitis
horse	outcome = '1'	Yes	124	http://archive.ics.uci.edu/ml/d

---

[22] See Section A.1.1 and A.1.2

				atasets/Horse+Colic
hypothyroid	class <> 'negative'	Yes	1800	http://archive.ics.uci.edu/ml/datasets/Thyroid+Disease
internetads	y=1	No	2153	http://archive.ics.uci.edu/ml/datasets/Internet+Advertisements
ionosphere	class = 'g'	No	230	http://archive.ics.uci.edu/ml/datasets/Ionosphere
iris	class = 'Iris-setosa'	No	99	http://archive.ics.uci.edu/ml/datasets/Iris
letter	class = 'F'	No	13221	http://archive.ics.uci.edu/ml/datasets/Letter+Recognition
lymph	class = 'metastases'	No	99	http://archive.ics.uci.edu/ml/datasets/Lymphography
segment	class = 'sky'	No	1490	http://archive.ics.uci.edu/ml/datasets/Statlog+(Image+Segmentation)
sonar	class = 'Mine'	No	143	http://archive.ics.uci.edu/ml/datasets/Connectionist+Bench+(Sonar,+Mines+vs.+Rocks)
spectf	y=1	Yes	177	http://archive.ics.uci.edu/ml/datasets/SPECTF+Heart
vehicle	class = 'saab'	No	552	http://archive.ics.uci.edu/ml/datasets/Statlog+(Vehicle+Silhouettes)
waveform	class <> 1	No	3342	http://archive.ics.uci.edu/ml/datasets/Waveform+Database+Generator+(Version+2)
wine	class = 2	No	110	http://archive.ics.uci.edu/ml/datasets/Wine
wisconsin	diagnosis = 'B'	Yes	391	http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)
yeast	cls = 'CYT'	Yes	973	http://archive.ics.uci.edu/ml/datasets/Yeast

Table 10

Data Sets with Numerical Target Variable, Transformed into a Fuzzy Proposition $\mu_\uparrow$
(see Formula 43 and Formula 44)

Name	Target	Filter[23]	Records	Link (accessed 08.2010)
auto93	$\mu_\uparrow$(price)	Yes	138	http://www.amstat.org/public ations/jse/v1n1/datasets.lock. html
autohorse	$\mu_\uparrow$(horsepower)	No	65	http://archive.ics.uci.edu/ml/d atasets/Automobile
baskball	$\mu_{\uparrow-}$ (points_per_minut e)	No	62	http://lib.stat.cmu.edu/datase ts/smoothmeth
bodyfat	$\mu_\uparrow$(bodyfat)	No	163	http://lib.stat.cmu.edu/datase ts/bodyfat
bolts	$\mu_\uparrow$(t20bolt)	No	30	http://lib.stat.cmu.edu/datase ts/bolts
breasttumor	$\mu_\uparrow$(tumor_size)	No	181	http://archive.ics.uci.edu/ml/ machine-learning-databases/breast-cancer/
cholestrol	$\mu_\uparrow$ (chol)	No	203	http://archive.ics.uci.edu/ml/d atasets/Heart+Disease
cloud	$\mu_\cdot$(te)	No	68	http://lib.stat.cmu.edu/datase ts/cloud
communities	$\mu_{\uparrow-}$ (violentcrimesperp op)	Yes	1363	http://archive.ics.uci.edu/ml/d atasets/Communities+and+C rime
cpu	$\mu_\uparrow$(prp)	No	135	http://archive.ics.uci.edu/ml/ machine-learning-databases/cpu-performance/
elusage	$\mu_\uparrow$(average_electric ity_usage)	No	35	http://lib.stat.cmu.edu/datase ts/smoothmeth
fishcatch	$\mu_\uparrow$(wheight)	No	97	http://www.amstat.org/public ations/jse/datasets/fishcatch. dat http://www.amstat.org/public ations/jse/datasets/fishcatch.t xt
forestfires	$\mu_\uparrow$(area)	No	346	http://archive.ics.uci.edu/ml/d atasets/Forest+Fires
fruitfly	$\mu_\uparrow$(longevity)	No	79	http://www.amstat.org/public ations/jse/datasets/fruitfly.da t http://www.amstat.org/public ations/jse/datasets/fruitfly.txt
housing	$\mu_\uparrow$(price)	No	348	http://archive.ics.uci.edu/ml/d atasets/Housing
lowbwt	$\mu_\uparrow$(bwt)	No	124	http://www.umass.edu/statda ta/statdata/data/lowbwt.txt http://www.umass.edu/statda

---

131

				ta/statdata/data/lowbwt.dat
meta	μ_r(error_rate)	No	348	http://archive.ics.uci.edu/ml/datasets/Meta-data
parkinsons	μ_r(motor_UPDRS)	Yes	1172	http://archive.ics.uci.edu/ml/datasets/Parkinsons
pbc	μ_r(survival_time)	Yes	114	http://lib.stat.cmu.edu/datasets/pbc
pharynx	μ_r(days_survived)	Yes	117	http://www.umass.edu/statdata/statdata/data/pharynx.txt http://www.umass.edu/statdata/statdata/data/pharynx.dat
pollution	μ_r(mort)	No	39	http://lib.stat.cmu.edu/datasets/pollution
quake	μ_r(richter)	No	1463	http://lib.stat.cmu.edu/datasets/smoothmeth
sensory	μ_r(score)	No	391	http://lib.stat.cmu.edu/datasets/sensory
servo	μ_r(servo_rise_time)	No	108	http://archive.ics.uci.edu/ml/datasets/Servo
sleep	μ_r(total_sleep)	No	41	http://lib.stat.cmu.edu/datasets/sleep
slump	μ_r(slump_cm)	Yes	75	http://archive.ics.uci.edu/ml/datasets/Concrete+Slump+Test
strike	μ_r(volume)	No	416	http://lib.stat.cmu.edu/datasets/strikes
veteran	μ(survival)	No	88	http://lib.stat.cmu.edu/datasets/veteran
wineqred	μ_r(quality)	No	1058	http://archive.ics.uci.edu/ml/datasets/Wine+Quality
wineqwhite	μ_r(quality)	No	3247	http://archive.ics.uci.edu/ml/datasets/Wine+Quality

A.1.1. Attribute Selection Filters

Data set	Selected attributes in prediction experiments
anneal	surface_quality, family, ferro, thick, formability, chrom, condition, steel, non_ageing, phos, surface_finish, bf, enamelability, width, blue_bright_varn_clean, strength, carbon, shape, cbond, bw_me, exptl, lustre, bl
auto93	Type, City_MPG, Highway_MPG, Air_Bags_standard, Drive_train_type, Number_of_cylinders, Engine_size, RPM, Engine_revolutions_per_mile, Manual_transmission_available, Fuel_tank_capacity, Passenger_capacity, Length, Wheelbase, Width, U_turn_space, Rear_seat_room, Luggage_capacity, Weight, Domestic
bands	grain_screened, ink_color, proof_on_ctd_ink, blade_mfg, cylinder_division , paper_type, ink_type, direct_steam, solvent_type, type_on_cylinder , press_type, press , unit_number , cylinder_size, paper_mill_location, plating_tank, proof_cut, viscosity, caliper, ink_temperature, humifity, roughness, blade_pressure, varnish_pct, press_speed, ink_pct, solvent_pct, ESA_Voltage, ESA_Amperage , wax, hardener, roller_durometer, current_density_num , anode_space_ratio, chrome_content
communities	pctilleg, pctkids2par, pctfam2par, racepctwhite, numilleg, pctyoungkids2par, pctteen2par, racepctblack, pctwinvinc, pctwpubasst, numunderpov, pctpopunderpov, femalepctdiv, pctpersdensehous, totalpctdiv, pctpersownoccup, pctunemployed, malepctdivorce, pcthousnophone, pctvacantboarded, pctnothsgrad, medfaminc, pcthousownocc, housvacant, pcthousless3br, pctless9thgrade, medincome, blackpercap, pctlarghousefam, numinshelters, numstreet, percapinc, agepct16t24, population, numburban, pctwofullplumb, pcthousoccup, lemaspctofficdrugun, pctemploy, mednumbr, pctoccupmgmtprof, pctimmigrec10, numimmig, pctbsormore, medrentpcthousinc, pctoccupmanu, pctimmigrec8, pctwwage, agepct12t29, pctlarghouseoccup, hisppercap, popdens, pctimmigrecent, pctimmigrec5
horse	Age, surgery, mucous_membranes, to_number(pulse) as pulse, capillary_refill_time, to_number(packed_cell_volume) as packed_cell_volume, peristalsis, pain, abdominal_distension, temperature_of_extremities, to_number(rectal_temperature)
hypothyroid	on_thyroxine, query_on_thyroxine, on_antithyroid_medication, sick, pregnant, thyroid_surgery, I131_treatment, query_hypothyroid, query_hyperthyroid, lithium, goitre, tumor, hypopituitary, psych, to_number(replace(TSH, '?', null)) as TSH, to_number(replace(T3, '?', null)) as T3, to_number(replace(TT4, '?', null)) as TT4, to_number(replace(T4U, '?', null)) as T4U, to_number(replace(FTI, '?', null)) as FTI, referral_source
parkinsons	subject_no, age, sex, test_time, Jitter_prc, Jitter_abs, Jitter_rap, Jitter_PPQ5, Jitter_DDP, Shimmer, Shimmer_dB, Shimmer_APQ3, Shimmer_APQ5, Shimmer_APQ11, Shimmer_DDA, NHR, HNR, RPDE, DFA, PPE

pbc	Z1, Z2, Z3, Z4, Z5, Z6, Z7, Z8, Z9, Z10, Z11, Z12, Z13, Z14, Z15, Z16, Z17
pharynx	Inst, sex, Treatment, Grade, Age, Condition, Site, T, N, Status
slump	Cement, Slag, Fly_ash, Water, SP, Coarse_Aggr, Fine_Agg
spectf	f12s, f22r, f10s, f9s, f16s, f3s, f5s, f8s, f5r, f19s, f2s, f13s, f15s, f20s
wisconsin	radius, texture, perimeter, area, smoothness, compactness, concavity, concave_points, symmetry, fractal_dimension
yeast	mcg, gvh, alm, mit, erl, pox, vac, nuc

A.1.2. Record Selection Filters

Data set	Selection criteria
horse	surgery <> '?' and mucous_membranes <> '?' and pulse <> '?' and capillary_refill_time <> '?' and packed_cell_volume <> '?' and peristalsis <> '?' and pain <> '?' and abdominal_distension <> '?' and temperature_of_extremities <> '?' and rectal_temperature <> '?'
hypothyroid	TSH <> '?' and T3 <> '?' and TT4 <> '?' and T4U <> '?' and FTI <> '?'

A.2. Database Queries

A.2.1. Database Queries for Experiments with Sharp Target Variables

Query 1: View for evaluation of experiment base data

```
create table evaluation_basis3 as
select
dataset, method_tmp as aggregation, ifc,
holdout_no,
correlation,
auc,mae,rmse,
c_s, c_s_sig, c_k, c_k_sig
from
(
  select
  ifcl_file,dataset,case when ifc is null then 'noifc' else ifc
end as ifc,
  correlation,
  percent_rank() over(partition by dataset order by correlation)
as r,
  --correlation as r,
  auc,mae,rmse,
  replace(
  replace(regexp_replace(replace(ifcl_file,
'./metainduction/'||dataset||'/'||ifc||'/', ''), '[[:digit:]]+_',
''),'.ifcl')
  , './metainduction/'||dataset||'/', '') as method_tmp,
  holdout_no,
  c_s, c_s_sig, c_k, c_k_sig
  from
  (
      select
      ifcl_file,dataset,correlation,auc,mae,rmse,
      replace(replace(ifc_method_tmp,
'./metainduction/'||dataset||'/', ''), '/', '') as ifc,
      holdout_no, c_s, c_s_sig, c_k, c_k_sig
      from
```

```
        (
          SELECT
          ifcl_file,correlation,auc,mae,rmse,
          replace(replace(REGEXP_SUBSTR(ifcl_file,
'./metainduction/[[:alnum:]]+/'),
'./metainduction/', ''), '/', '') as dataset,
          REGEXP_SUBSTR(ifcl_file,
'./metainduction/([[:alnum:]]+)/([[:alpha:]])+/')  as
ifc_method_tmp,
          holdout_no, c_s, c_s_sig, c_k, c_k_sig
          from evaluations_repeated
        )
    )
)
where method_tmp not like '%numeric%'
and method_tmp not like '%ct%'

create index ix1 on evaluation_basis3(dataset)
create index ix2 on evaluation_basis3(ifc)
```

Query 2: Average correlation per aggregation method

```
select aggregation, avg(correlation)
from evaluation_basis3
where ifc <> 'noifc'
group by aggregation
order by avg(correlation) desc
```

Query 3: Test of significance in difference between logistic regression, linear regression and regression tree as aggregation methods

```
select stats_wsr_test(corr_logreg, corr_regtree, 'ONE_SIDED_SIG'),
stats_wsr_test(corr_logreg, corr_linreg, 'ONE_SIDED_SIG')
from
(
    select
    dataset,
```

136

```
    sum(case when aggregation = 'logreg' then correlation else 0
end) /
    sum(case when aggregation = 'logreg' then 1 else 0 end) as
corr_logreg,
    sum(case when aggregation = 'regtree' then correlation else 0
end) /
    sum(case when aggregation = 'regtree' then 1 else 0 end) as
corr_regtree,
    sum(case when aggregation = 'linreg' then correlation else 0
end) /
    sum(case when aggregation = 'linreg' then 1 else 0 end) as
corr_linreg
    from evaluation_basis3
    where ifc <> 'noifc'
group by dataset
)
```

Query 4: Average correlation per induction and aggregation method

```
select
ifc, aggregation, avg(correlation)
from evaluation_basis3
group by ifc, aggregation
order by avg(correlation) desc
```

Query 5: Test of significance of difference between logistic regression using NLR-IAF, NLD-IAF and regression trees without IAF

```
select
stats_wsr_test(corr_logreg_nlr, corr_regtree_no_iaf,
'ONE_SIDED_SIG'),
stats_wsr_test(corr_logreg_nlr, corr_logreg_nld, 'ONE_SIDED_SIG')
from
(
    select
    dataset,
sum(case when ifc||'.'||aggregation = 'nlr.logreg' then
correlation
```

```
else 0 end)
/ sum(case when ifc||'.'||aggregation = 'nlr.logreg' then 1 else 0
end)
as corr_logreg_nlr,
     sum(case when ifc||'.'||aggregation = 'nld.logreg' then
correlation
else 0 end)
/ sum(case when ifc||'.'||aggregation = 'nld.logreg' then 1 else 0
end)
as corr_logreg_nld,
     sum(case when ifc||'.'||aggregation = 'noifc.regtree' then
correlation
else 0 end)
/ sum(case when ifc||'.'||aggregation = 'noifc.regtree' then 1
else 0 end)
as corr_regtree_no_iaf
     from evaluation_basis3
     where dataset is not null
     group by dataset )
```

A.2.2. Database Queries for Experiments with Numerical Target Variables

Query 6: View for evaluation of experiment base data

```
create table evaluation_basis_num as
select
dataset, method_tmp as aggregation, ifc, fuzzification,
holdout_no,
correlation,
auc,mae,rmse,
c_s, c_s_sig, c_k, c_k_sig
from
(
  select
  ifcl_file,dataset,
  nvl(replace(substr(ifc, 1, regexp_instr(ifc, '._l|_p')),'_',
''), 'noifc') as ifc,
```

```sql
  replace(substr(ifc, regexp_instr(ifc, '._l|_p')+1, length(ifc)),
'_', '') as fuzzification,
  correlation,
  percent_rank() over(partition by dataset order by correlation)
as r,
  --correlation as r,
  auc,mae,rmse,
  replace(
  replace(regexp_replace(replace(ifcl_file,
'./metainduction_num/'||dataset||'/'||ifc||'/', ''),
'[[:digit:]]+_', ''),'.ifcl')
  ,  './metainduction_num/'||dataset||'/', '') as method_tmp,
  holdout_no,
  c_s, c_s_sig, c_k, c_k_sig
   from
  (
      select
      ifcl_file,dataset,correlation,auc,mae,rmse,
      replace(replace(ifc_method_tmp,
'./metainduction_num/'||dataset||'/', ''), '/', '') as ifc,
      holdout_no, c_s, c_s_sig, c_k, c_k_sig
      from
      (
        SELECT
        ifcl_file,correlation,auc,mae,rmse,
        replace(replace(REGEXP_SUBSTR(ifcl_file,
'./metainduction_num/[[:alnum:]]+/'),
'./metainduction_num/', ''), '/', '') as dataset,
        REGEXP_SUBSTR(ifcl_file,
'./metainduction_num/([[:alnum:]]+)/([[:print:]])+/')  as
ifc_method_tmp,
        holdout_no, c_s, c_s_sig, c_k, c_k_sig
        from evaluations_repeated_num
      )
  )
)


create index ix3 on evaluation_basis_num(dataset)
```

```
create index ix4 on evaluation_basis_num(ifc)
create index ix5 on evaluation_basis_num(fuzzification)
```

Query 7: Average correlation per aggregation method, given linear ITF

```
select aggregation, avg(correlation)
from evaluation_basis_num
where ifc <> 'noifc'
group by aggregation
order by avg(correlation) desc
```

Query 8: Test of significance in difference between regression tree, linear regression and average

```
select
stats_wsr_test(corr_regtree, corr_avg, 'ONE_SIDED_SIG'),
stats_wsr_test(corr_regtree, corr_linreg, 'ONE_SIDED_SIG')
from
(
    select
    dataset,
    sum(case when aggregation = 'avg' then correlation else 0
end) /
    sum(case when aggregation = 'avg' then 1 else 0 end) as
corr_avg,
    sum(case when aggregation = 'regtree' then correlation else 0
end) /
    sum(case when aggregation = 'regtree' then 1 else 0 end) as
corr_regtree,
    sum(case when aggregation = 'linreg' then correlation else 0
end) /
    sum(case when aggregation = 'linreg' then 1 else 0 end) as
corr_linreg
    from evaluation_basis_num
    where ifc <> 'noifc'
group by dataset
)
```

Query 9: Average correlation per target fuzzification method

```
select
fuzzification, avg(correlation)
from evaluation_basis_num
where aggregation = 'regtree'
group by fuzzification
order by avg(correlation) desc
```

Query 10: Test of significance between target fuzzification methods

```
select
stats_wsr_test(corr_l, corr_p, 'ONE_SIDED_SIG')
from
(
    select
    dataset,
    sum(case when fuzzification = 'l' then correlation else 0
end) /
    sum(case when fuzzification = 'l' then 1 else 0 end) as
corr_l,
    sum(case when fuzzification = 'p' then correlation else 0
end) /
    sum(case when fuzzification = 'p' then 1 else 0 end) as
corr_p
    from evaluation_basis_num
    where aggregation = 'regtree'
  and ifc <> 'noifc'
group by dataset
)
```

Query 11: Average correlation per induction and aggregation method (given linear target fuzzification)

```
select
ifc, aggregation, avg(correlation)
from evaluation_basis_num
where (fuzzification = 'l' or fuzzification is null)
and aggregation <> 'logreg_p'
group by ifc, aggregation
```

```
order by avg(correlation) desc
```

Query 12: Test of significance of difference between regression trees using NLD-IAF , NLDU-IAF and regression trees without IAF

```
select
stats_wsr_test(corr_regtree_nld, corr_regtree_no_iaf,
'ONE_SIDED_SIG'),
stats_wsr_test(corr_regtree_nld, corr_regtree_nldu,
'ONE_SIDED_SIG'),
stats_wsr_test(corr_regtree_nld, corr_regtree_nlr,
'ONE_SIDED_SIG')
from
(
    select
    dataset,
sum(case when ifc||'.'||aggregation = 'nld.regtree' then
correlation
else 0 end)
/ sum(case when ifc||'.'||aggregation = 'nld.regtree' then 1 else
0 end)
as corr_regtree_nld,
    sum(case when ifc||'.'||aggregation = 'nldu.regtree' then
correlation
else 0 end)
/ sum(case when ifc||'.'||aggregation = 'nldu.regtree' then 1 else
0 end)
as corr_regtree_nldu,
sum(case when ifc||'.'||aggregation = 'npr.regtree' then
correlation
else 0 end)
/ sum(case when ifc||'.'||aggregation = 'npr.regtree' then 1 else
0 end)
as corr_regtree_nlr,
    sum(case when ifc||'.'||aggregation = 'noifc.regtree' then
correlation
else 0 end)
/ sum(case when ifc||'.'||aggregation = 'noifc.regtree' then 1
else 0 end)
```

```
as corr_regtree_no_iaf
    from evaluation_basis_num
    where dataset is not null
    group by dataset
)
```

A.2.3. Database Queries for Both Types of Experiments

Query 13: Table with data parameters for each dataset

```
-- Execute 00_load_all.bat first
create or replace view data_params as
select * from
( -- improvement
  select dataset, (corr_logreg_nlr-
corr_logreg_no_iaf)/corr_logreg_no_iaf as improvement_by_iaf
  from
  (
    select
    dataset,
    sum(case when ifc||'.'||aggregation = 'nlr.logreg' then
correlation
    else 0 end)
    / sum(case when ifc||'.'||aggregation = 'nlr.logreg' then 1
else 0 end)
    as corr_logreg_nlr,
    sum(case when ifc||'.'||aggregation = 'noifc.regtree' then
correlation
    else 0 end)
    / sum(case when ifc||'.'||aggregation = 'noifc.regtree' then 1
else 0 end)
    as corr_regtree_no_iaf,
    sum(case when ifc||'.'||aggregation = 'noifc.logreg' then
correlation
    else 0 end)
    / sum(case when ifc||'.'||aggregation = 'noifc.logreg' then 1
else 0 end)
    as corr_logreg_no_iaf
```

```
    from evaluation_basis3
    where dataset is not null
    group by dataset
  )
  union
  select
  dataset, (corr_regtree_nld -
corr_regtree_no_iaf)/corr_regtree_no_iaf as improvement_by_iaf
  from
  (
  select
    dataset,
    sum(case when ifc||'.'||aggregation = 'nld.regtree' then
correlation
    else 0 end)
    / sum(case when ifc||'.'||aggregation = 'nld.regtree' then 1
else 0 end)
    as corr_regtree_nld,
    sum(case when ifc||'.'||aggregation = 'noifc.regtree' then
correlation
    else 0 end)
    / sum(case when ifc||'.'||aggregation = 'noifc.regtree' then 1
else 0 end)
    as corr_regtree_no_iaf
    from evaluation_basis_num
    where dataset is not null
    group by dataset
  )
)
join
( -- column count
  select
  replace(lower(table_name), '_tr', '') as dataset,
  sum(case when data_type in ('CHAR', 'VARCHAR2') then 1 else 0
end) as ncatcols,
  sum(case when data_type in ('NUMBER', 'FLOAT') then 1 else 0
end) as nnumcols,
```

```
   sum(case when data_type in ('CHAR', 'VARCHAR2') then 1 else 0
end)/count(*) as pcatcols,
   sum(case when data_type in ('NUMBER', 'FLOAT') then 1 else 0
end)/count(*) as pnumcols,
   count(*) as ncols
   from
   sys.all_tab_columns
   where table_name in
   ('ABALONE_TR', 'ADULT_TR', 'ANNEAL_TR', 'BALANCE_TR',
'BANDS_TR', 'BREASTW_TR', 'CAR_TR', 'CMC_TR', 'CREDIT_TR',
'CREDITG_TR', 'DIABETES_TR', 'DIAGNOSIS_TR', 'GLASS_TR',
'HEART_TR', 'HEPATITIS_TR', 'HORSE_TR', 'HYPOTHYROID_TR',
'INTERNETADS_TR', 'IONOSPHERE_TR', 'IRIS_TR', 'LETTER_TR',
'LYMPH_TR', 'SEGMENT_TR', 'SONAR_TR', 'SPECTF_TR', 'VEHICLE_TR',
'WAVEFORM_TR', 'WINE_TR', 'WISCONSIN_TR', 'YEAST_TR', 'AUTO93_TR',
'AUTOHORSE_TR', 'BASKBALL_TR', 'BODYFAT_TR', 'BOLTS_TR',
'BREASTTUMOR_TR', 'CHOLESTROL_TR', 'CLOUD_TR', 'COMMUNITIES_TR',
'CPU_TR', 'ELUSAGE_TR', 'FISHCATCH_TR', 'FORESTFIRES_TR',
'FRUITFLY_TR', 'HOUSING_TR', 'LOWBWT_TR', 'META_TR',
'PARKINSONS_TR', 'PBC_TR', 'PHARYNX_TR', 'POLLUTION_TR',
'QUAKE_TR', 'SENSORY_TR', 'SERVO_TR', 'SLEEP_TR', 'SLUMP_TR',
'STRIKE_TR', 'VETERAN_TR', 'WINEQRED_TR', 'WINEQWHITE_TR')
   and column_name <> 'IS_TRAINING'
   and column_name <> 'Y'
   group by table_name
)
using (dataset)
join
( -- correlation of linear regression
   select dataset, avg(correlation) as avg_corr_linreg from
evaluation_basis3
   where ifc = 'noifc'
   and aggregation = 'linreg'
   group by dataset
   union
   select dataset, avg(correlation) from evaluation_basis_num
   where ifc = 'noifc'
   and aggregation = 'linreg'
```

```
  group by dataset
)
using (dataset) join
( -- row count
  select 'abalone' as dataset, 'c' as target_type, count(*) as
nrows from abalone_tr
  union select 'abalone', 'c', count(*) from abalone_tr
  union select 'adult', 'c', count(*) from adult_tr
  union select 'anneal', 'c', count(*) from anneal_tr
  union select 'balance', 'c', count(*) from balance_tr
  union select 'bands', 'c', count(*) from bands_tr
  union select 'breastw', 'c', count(*) from breastw_tr
  union select 'car', 'c', count(*) from car_tr
  union select 'cmc', 'c', count(*) from cmc_tr
  union select 'credit', 'c', count(*) from credit_tr
  union select 'creditg', 'c', count(*) from creditg_tr
  union select 'diabetes', 'c', count(*) from diabetes_tr
  union select 'diagnosis', 'c', count(*) from diagnosis_tr
  union select 'glass', 'c', count(*) from glass_tr
  union select 'heart', 'c', count(*) from heart_tr
  union select 'hepatitis', 'c', count(*) from hepatitis_tr
  union select 'horse', 'c', count(*) from horse_tr
  union select 'hypothyroid', 'c', count(*) from hypothyroid_tr
  union select 'internetads', 'c', count(*) from internetads_tr
  union select 'ionosphere', 'c', count(*) from ionosphere_tr
  union select 'iris', 'c', count(*) from iris_tr
  union select 'letter', 'c', count(*) from letter_tr
  union select 'lymph', 'c', count(*) from lymph_tr
  union select 'segment', 'c', count(*) from segment_tr
  union select 'sonar', 'c', count(*) from sonar_tr
  union select 'spectf', 'c', count(*) from spectf_tr
  union select 'vehicle', 'c', count(*) from vehicle_tr
  union select 'waveform', 'c', count(*) from waveform_tr
  union select 'wine', 'c', count(*) from wine_tr
  union select 'wisconsin', 'c', count(*) from wisconsin_tr
  union select 'yeast', 'c', count(*) from yeast_tr
  union select 'auto93', 'n', count(*) from auto93_tr
  union select 'autohorse', 'n', count(*) from autohorse_tr
```

```
  union select 'baskball', 'n', count(*) from baskball_tr
  union select 'bodyfat', 'n', count(*) from bodyfat_tr
  union select 'bolts', 'n', count(*) from bolts_tr
  union select 'breasttumor', 'n', count(*) from breasttumor_tr
  union select 'cholestrol', 'n', count(*) from cholestrol_tr
  union select 'cloud', 'n', count(*) from cloud_tr
  union select 'communities', 'n', count(*) from communities_tr
  union select 'cpu', 'n', count(*) from cpu_tr
  union select 'elusage', 'n', count(*) from elusage_tr
  union select 'fishcatch', 'n', count(*) from fishcatch_tr
  union select 'forestfires', 'n', count(*) from forestfires_tr
  union select 'fruitfly', 'n', count(*) from fruitfly_tr
  union select 'housing', 'n', count(*) from housing_tr
  union select 'lowbwt', 'n', count(*) from lowbwt_tr
  union select 'meta', 'n', count(*) from meta_tr
  union select 'parkinsons', 'n', count(*) from parkinsons_tr
  union select 'pbc', 'n', count(*) from pbc_tr
  union select 'pharynx', 'n', count(*) from pharynx_tr
  union select 'pollution', 'n', count(*) from pollution_tr
  union select 'quake', 'n', count(*) from quake_tr
  union select 'sensory', 'n', count(*) from sensory_tr
  union select 'servo', 'n', count(*) from servo_tr
  union select 'sleep', 'n', count(*) from sleep_tr
  union select 'slump', 'n', count(*) from slump_tr
  union select 'strike', 'n', count(*) from strike_tr
  union select 'veteran', 'n', count(*) from veteran_tr
  union select 'wineqred', 'n', count(*) from wineqred_tr
  union select 'wineqwhite', 'n', count(*) from wineqwhite_tr
)
using (dataset)
```

Query 14: Correlation of parameters with the performance improvement by IFC

```
select
target_type,
CORR_S(improvement_by_iaf, ncatcols) c_ncatcols,
CORR_S(improvement_by_iaf, ncatcols, 'TWO_SIDED_SIG') p_ncatcols,
CORR_S(improvement_by_iaf, nnumcols) c_nnumcols,
```

```
CORR_S(improvement_by_iaf, nnumcols, 'TWO_SIDED_SIG') p_nnumcols,
CORR_S(improvement_by_iaf, pcatcols) c_pcatcols,
CORR_S(improvement_by_iaf, pcatcols, 'TWO_SIDED_SIG') p_pcatcols,
CORR_S(improvement_by_iaf, pnumcols) c_pnumcols,
CORR_S(improvement_by_iaf, pnumcols, 'TWO_SIDED_SIG') p_pnumcols,
CORR_S(improvement_by_iaf, ncols) c_ncols,
CORR_S(improvement_by_iaf, ncols, 'TWO_SIDED_SIG') p_ncols,
CORR_S(improvement_by_iaf, avg_corr_linreg) c_avg_corr_linreg,
CORR_S(improvement_by_iaf, avg_corr_linreg, 'TWO_SIDED_SIG')
p_avg_corr_linreg,
CORR_S(improvement_by_iaf, nrows) c_nrows,
CORR_S(improvement_by_iaf, nrows, 'TWO_SIDED_SIG') p_nrows,
CORR_S(improvement_by_iaf, nrows/ncols) c_nrowscols,
CORR_S(improvement_by_iaf, nrows/ncols, 'TWO_SIDED_SIG')
p_nrowscols
from data_params
group by target_type
```

Query 15: Table for scatter plot visualization of association between linearity of data and improvement by IAF

```
select dataset, target_type, avg_corr_linreg, improvement_by_iaf
from data_params order by target_type, dataset
```

A.3. Experimental Results Tables

Table 11

Average Prediction Correlations per Dataset/Aggregation Method for Binary Targets

Dataset	Avg. Corr. Log. Reg.	Avg. Corr. Reg. Tree	Avg. Corr. Lin. Reg.
abalone	0.638687626	0.651733317	0.636539418
adult	0.659169773	0.651350849	0.636197273
anneal	0.823367865	0.846689248	0.797382729
balance	0.920732292	0.791055945	0.840604813
bands	0.454571882	0.351967704	0.426261177
breastw	0.939085094	0.927238012	0.931453636
car	0.914349826	0.933901714	0.811581886
cmc	0.425228906	0.421732164	0.418979581
credit	0.76498662	0.757839955	0.767073216
creditg	0.457163162	0.426897568	0.437699783
diabetes	0.54510302	0.533295481	0.546568895
diagnosis	1	0.868887353	0.965967697
glass	0.794456151	0.774937317	0.802028743
heart	0.695053211	0.643374806	0.684389592
hepatitis	0.291646106	0.383819768	0.388031234
horse	0.572088107	0.529928212	0.555148535
hypothyroid	0.792125757	0.892843929	0.79338345
internetads	0.805987731	0.753679747	0.78294501
ionosphere	0.819557055	0.799538551	0.798060482
iris	1	0.999193202	0.991252264
letter	0.626368894	0.798441855	0.480644051
lymph	0.695768921	0.672304979	0.726334152
segment	0.99999997	0.997855955	0.992554443
sonar	0.558775153	0.554680801	0.578090696
spectf	0.501761759	0.447468223	0.463867849
vehicle	0.475892051	0.460322532	0.437534698
waveform	0.82107583	0.785686835	0.74716555
wine	0.918745813	0.861574492	0.914019581
wisconsin	0.908323537	0.872676408	0.88191778
yeast	0.421658925	0.424627552	0.423187415

Table 12

Average Prediction Correlations per Dataset and Aggregation Method for Numerical Target Variables

Dataset	Avg. Corr. Avg.	Avg. Corr. Reg. Tree	Avg. Corr. Lin. Reg.
auto93	0.691201303	0.857445115	0.738746817
autohorse	0.892653343	0.926796658	0.930059659
baskball	0.536398611	0.638183226	0.539943795
bodyfat	0.744776961	0.980938537	0.977226072
bolts	0.79763048	0.894265558	0.905782285
breasttumor	0.231971819	0.380396489	0.258283305
cholestrol	0.176404133	0.345649097	0.163348437
cloud	0.8020314	0.891982414	0.865600118
communities	0.731280609	0.79667882	0.780258252
cpu	0.788972056	0.814152614	0.794450568
elusage	0.865618561	0.897333607	0.872277672
fishcatch	0.845272306	0.918434123	0.894956413
forestfires	0.018454772	0.167736369	0.014733488
fruitfly	-0.131668697	0.165425495	-0.067547186
housing	0.687542408	0.908550517	0.86615543
lowbwt	0.741678073	0.810166446	0.788121932
meta	0.27658433	0.448502931	0.353513754
parkinsons	0.326735958	0.881670195	0.605425938
pbc	0.475841958	0.67048775	0.487261393
pharynx	0.606093213	0.775843245	0.683939249
pollution	0.656915329	0.811606345	0.655832652
quake	0.057471089	0.081629626	0.058700741
sensory	0.290356213	0.400912946	0.337693928
servo	0.703894008	0.828999856	0.780728183
sleep	0.713524339	0.753362622	0.601491837
slump	0.422059811	0.668746482	0.500413959
strike	0.382845386	0.485543157	0.418259534
veteran	0.358292594	0.483294002	0.413181002
wineqred	0.544798097	0.615254318	0.595187199
wineqwhite	0.507972143	0.577186397	0.543279329

Table 13:
Average Prediction Correlations for IAF on Data with Numerical Target Variables using
Linear versus Percentile ITF

Dataset	Linear ITF	Percentile ITF
auto93	0.88105593	0.833834299
autohorse	0.946780795	0.90681252
baskball	0.635879956	0.640486496
bodyfat	0.984034462	0.977842612
bolts	0.921790457	0.866740659
breasttumor	0.385000153	0.375792824
cholestrol	0.365144244	0.32615395
cloud	0.926867976	0.857096852
communities	0.82334782	0.770009819
cpu	0.937630747	0.690674481
elusage	0.905249904	0.88941731
fishcatch	0.94873926	0.888128986
forestfires	0.210580299	0.124892439
fruitfly	0.157932749	0.172918241
housing	0.940547453	0.87655358
lowbwt	0.814805887	0.805527006
meta	0.577117175	0.319888686
parkinsons	0.880604806	0.882735584
pbc	0.666064677	0.674910823
pharynx	0.785763664	0.765922825
pollution	0.807044092	0.816168598
quake	0.078567569	0.084691683
sensory	0.399599899	0.402225994
servo	0.872101166	0.785898546
sleep	0.783933708	0.722791535
slump	0.708012949	0.629480015
strike	0.522569102	0.448517211
veteran	0.515124253	0.45146375
wineqred	0.619064343	0.611444293
wineqwhite	0.58210665	0.572266143

Table 14

Average Prediction Correlations per Dataset and Three Different Combinations of
Supervised Aggregation and IAF for Binary Target Variables

Dataset	Avg. Corr. Log. Reg. + IAF NLR	Avg. Corr. Log. Reg. + IAF NLD	Avg. Corr. Reg. Tr w/o IAF
abalone	0.633651826	0.636990551	0.64867844
adult	0.662989858	0.656921233	0.608645742
anneal	0.826555309	0.823932942	0.853821973
balance	0.923393427	0.924777928	0.79058825
bands	0.461862401	0.450021541	0.343296921
breastw	0.939701709	0.939464426	0.930009743
car	0.91660187	0.913644582	0.948927087
cmc	0.42385694	0.423400481	0.455736376
credit	0.768277547	0.763557368	0.751852807
creditg	0.464104961	0.469843758	0.393045324
diabetes	0.556749014	0.545362638	0.50644607
diagnosis	1	1	0.944533837
glass	0.802605598	0.804760434	0.824622632
heart	0.694785041	0.70493683	0.657426735
hepatitis	0.374975313	0.311577565	0.339608818
horse	0.569485166	0.55213211	0.541218353
hypothyroid	0.80477987	0.800429276	0.898883448
internetads	0.807436554	0.808116466	0.798458728
ionosphere	0.809199099	0.819279121	0.764848148
iris	1	1	0.996991022
letter	0.6349395	0.62902598	0.727029396
lymph	0.68823764	0.690663197	0.727999682
segment	0.999999995	0.999999953	0.997593306
sonar	0.559774797	0.557996241	0.496422658
spectf	0.505452116	0.510651243	0.473619771
vehicle	0.46898602	0.496373681	0.559857093
waveform	0.826959479	0.821165897	0.783760649
wine	0.912450784	0.923247357	0.90726124
wisconsin	0.912735058	0.907890135	0.873368639
yeast	0.420860794	0.420696658	0.412990274

Table 15
Average Prediction Correlations per Dataset and Three Different Combinations of
Supervised Aggregation and IAF for Numerical Target Variables

Dataset	Avg. Corr. Reg. Tr. + IAF NLD	Avg. Corr. Reg. Tr. + IAF NLDU	Avg. Corr. Reg. Tr. w/o IAF
auto93	0.851949884	0.854702482	0.783694291
autohorse	0.934671172	0.932528555	0.939116035
baskball	0.651189815	0.65121036	0.623578896
bodyfat	0.989455545	0.990353116	0.991221665
bolts	0.899445255	0.90182103	0.921987087
breasttumor	0.379942005	0.37994241	0.252316629
cholestrol	0.354768905	0.346985441	0.193892889
cloud	0.901964456	0.89860454	0.913436117
communities	0.798599026	0.796617813	0.791015749
cpu	0.843304898	0.842528113	0.97677025
elusage	0.905475647	0.905995831	0.881926102
fishcatch	0.932313534	0.933532791	0.985895292
forestfires	0.175728178	0.169928128	0.014991533
fruitfly	0.212102028	0.212108602	-0.064702557
housing	0.918270044	0.917557553	0.900012141
lowbwt	0.813006747	0.812914092	0.783555356
meta	0.462534319	0.454644049	0.368499477
parkinsons	0.887066413	0.881144444	0.941203949
pbc	0.66081537	0.660799138	0.534079402
pharynx	0.773127351	0.773127987	0.708101866
pollution	0.829522663	0.831729118	0.60923147
quake	0.082057392	0.08047578	0.073515546
sensory	0.404010864	0.404010206	0.386991775
servo	0.82787911	0.827815554	0.912721346
sleep	0.77696476	0.764053467	0.730795112
slump	0.690106281	0.689333045	0.493530846
strike	0.496745411	0.496578249	0.415429314
veteran	0.471382971	0.471232266	0.437471553
wineqred	0.611102154	0.615205884	0.580172859
wineqwhite	0.572707345	0.573805738	0.563459737

Table 16

Relationship between Target Linearity and Improvement of Logistic Regression by IAF
NLR for Binary Target Variables

Dataset	Target linearity	Improvement by IAF NLR
abalone	0.632486147	-0.015038624
adult	0.601697632	0.003616993
anneal	0.847487701	0.265333889
balance	0.839927779	0.010051891
bands	0.397808335	0.101084917
breastw	0.92253246	-0.001225938
car	0.814076541	-0.001610122
cmc	0.351769834	0.17744701
credit	0.733383933	0.063102998
creditg	0.425527893	0.040954514
diabetes	0.527451425	0.044424251
diagnosis	0.972462485	-5.2E-16
glass	0.817993073	-0.003667589
heart	0.6866888	0.030462835
hepatitis	0.381693585	0.098467765
horse	0.552106327	0.111811784
hypothyroid	0.584747606	0.080805492
internetads	0.784842676	0.00291351
ionosphere	0.681485184	0.15769256
iris	0.97214843	2.97338E-06
letter	0.45185967	0.10947387
lymph	0.723280478	0.033367981
segment	0.956800746	-4.89814E-09
sonar	0.512394902	0.043510527
spectf	0.452480291	-0.002580917
vehicle	0.57214217	-0.206318239
waveform	0.724956824	0.075158502
wine	0.910709707	-0.029160901

Table 17
Relationship between Target Linearity and Improvement of Regression Trees by IAF
NLD for Numerical Target Variables

Dataset	Target linearity	Improvement by IAF NLD
autohorse	0.946606415	-0.004733029
auto93	0.736992093	0.087094666
baskball	0.623578896	0.044278148
bodyfat	0.990358409	-0.001781761
bolts	0.85950705	-0.024449184
breasttumor	0.236969021	0.505814364
cholestrol	0.170467119	0.82971591
cloud	0.912861146	-0.0125588
communities	0.797109195	0.009586759
cpu	0.930542139	-0.136639452
elusage	0.845600615	0.026702402
fishcatch	0.963623075	-0.054348325
forestfires	0.073146275	10.72182832
fruitfly	-0.064702557	-4.278108926
housing	0.854820529	0.020286285
lowbwt	0.778262746	0.037586867
meta	0.383398892	0.25518311
parkinsons	0.459022397	-0.057519453
pbc	0.530953313	0.237297988
pharynx	0.699754036	0.091830693
pollution	0.679583809	0.361588664
quake	0.066085663	0.116191016
sensory	0.336532333	0.043977907
servo	0.836899785	-0.092955244
sleep	0.709322872	0.063177281
slump	0.436362228	0.398304251
strike	0.42286715	0.195739913
veteran	0.397186311	0.077516853
wineqred	0.583219599	0.053310483
wineqwhite	0.52352684	0.01641219

A.4. IFCL Syntax and Application

A.4.1. Grammar Notation

The syntax will be described with the aid of meta-linguistic formulae in the Backus-Naur Form (BNF) as proposed by Backus, et al. (1960):

`<...>`	Meta-linguistic variables: nodes of the grammar tree
`::=`	Definition of syntax element placeholders
`...	...`	Alternative choice of syntax elements
`<character>`	A symbol
`<string>`	Any sequence of symbols
`<integer>`	A sequence of digits (0-9)
`<decimal>`	A sequence of digits (0-9) containing one decimal point
`<empty>`	An empty set of symbols: no symbol at all

All symbols that are not inside of less than / greater than signs (`<...>`) except the symbols `::=` and `|` are used literally in the IFCL language. Different meta-linguistic variable declarations are separated by an empty line. Comments can be made in IFCL files as a sequence of symbols enclosed by the character #. In fact, comments are filtered out before parsing, so they are not part of the language itself.

A.4.2. IFCL File Structure

```
<IFCL file> ::=
<IFCL file node> <IFCL file leaf>


<IFCL file node> ::=
<IFCL file execution call> <IFCL file node>
| <empty>


<IFCL file leaf> ::=
<connect to database>  <IFCL leaf action sequence>


<IFCL leaf action sequence> ::=
<IFCL leaf action> <IFCL action sequence> | <empty>


<IFCL leaf action> ::=
| <drop database table>
| <execute sql>
| <load database>
```

```
| <induce membership function>
| <aggregate multiple variables>
| <data classification>
| <evaluate correlations>
```

A.4.3.   Executing IFCL Files in Batch Mode

o   Syntax

```
<IFCL file execution call> ::= @execifcl: file == <ifcl file
path>;


<ifcl file path> ::= <string>
```

o   Example

```
@execifcl: file == ./metainduction/abalone/load.ifcl;
```

A.4.4.   Connecting to the Database

o   Syntax

```
<connect to database> ::=
@connect:
hostname == <database host name>  ;
SID  == <database service identifier> ;
port  == <database server port number> ;
username  == <user name> ;
password  == <password> ;


<database host name> ::= <string>


<database service identifier>::= <string>


<database server port number> ::= <integer>


<user name> ::= <string>


<password> ::= <string>
```

o   Example

```
@connect:
HostName   == localhost;
```

```
SID        == XE;
Port       == 1521;
UserName   == ME;
Password   == passwort;
```

A.4.5.   Droping a Database Table

o   Syntax

```
<drop database table> ::= @drop: table == <table name>;


<table name>::= <string>
```

o   Example

```
@drop: table == abalone_te_nlr_regtree
```

A.4.6.   Executing an SQL Script

o   Syntax

```
<execute sql> ::= @execsql: <sql>


<sql> ::=
file == <sql script file path> ;
| command == <sql statement> ;


<sql script file path> ::= <string>


<sql statement> ::= <string>
```

o   Example

```
@execsql: file == metainduction/abalone/01_sql/table_split.sql;
```

A.4.7.   Loading Data into The Database

o   Syntax

```
<load database> ::= <SQL*Loader> | <IFCL loader>


<SQL*Loader>::=
@sqlldr:
data == <path to data file> ;
control == <path to sqlldr control file> ;
skip == <number of rows to skip>;
```

```
errors == <number of errors to accept>;

<path to data file> ::= <string>

<path to sqlldr control file> ::= <string>

<number of rows to skip> ::= <integer>

<number of errors to accept> ::= <integer>
```

o Example
```
@sqlldr:
data == metainduction/abalone/00_data/data.csv;
control == metainduction/abalone/00_data/sqlldr.ctl;
skip == 0;
errors == 10000;
```

o Syntax
```
<IFCL loader> ::=
@load:
file ==  <path to data file> ;
table == <table to be created and loaded> ;
delimiter == <data delimiting character> ;
attributes == <attribute list> ;
types == <type list>;

<path to data file> ::= <string>

<data delimiting character> ::= <character>

<attribute list> ::= <attribute> <attribute list> | <empty>

<attribute> ::= <string>

<type list> ::= <type> <type list> | <empty>

<type> ::= n | c
```

o Example

```
@load:
file == ./metainduction/balance/00_data/00_data.csv;
table == balance_load;
delimiter == ,;
attributes == left_weight, left_distance, right_weight,
right_distance, class;
types == n, n, n, n, c;
```

A.4.8.   Inducing a Membership Function

o Syntax

```
<induce membership function> ::=
@inducemf:
table == <table for membership function induction> ;
target variable == <table column> ;
analytic variables == <column> | <column list> | <all columns>
induction == <induction type> ;
output file == <path to output file for sql classification
template> ;
<optional skipped columns>
<optional left out columns>

<table for membership function induction> ::= <string>

<column> ::= <string>

<all columns> ::= *

<column list> ::=  <column> <column list or empty>

<column list or empty> = <column>  <column list or empty> |
<empty>

<induction type> ::= l | nlr | nlru | nld | nldu | cp | npr | npru
| npd | npdu | iff | nc | mm

<path to output file for sql classification template> ::= <string>
```

```
<optional skipped columns> ::=
skip columns == <column list> ;


<optional left out columns> ::=
let columns == <column list> ;
```

o   Example
```
@inducemf:
table == abalone_tr;
target variable == y;
analytic variables == *;
induction == cp;
output file == ./metainduction/abalone/02_output/cp.sql.template;
```

A.4.9.   Classification of Data

o   Syntax
```
<data classification> ::=
@classify:
classified table == <table to be classified> ;
template file == <path to sql classification template file> ;
output table == <table for storing resulting classification> ;


<table to be classified> ::= <string>


<path to sql classification template file> ::= <string>


<table for storing resulting classification> ::= <string>
```

o   Example
```
@classify:
classified table == abalone_te;
template file == ./metainduction/regtree.sql.template;
output table == abalone_te_rt;
```

A.4.10. Aggregating Multiple Variables

o   Syntax
```
<aggregate multiple variables> ::=
@aggregatemv:
```

```
table == <table to be aggregated> ;
analytic variables == <column> | <column list> | <all columns> ;
aggregation operator == <aggretation operator>;
target variable == <table column>;
output file == <path to output file>;
column alias == <name of column containing aggregated membership
value>;


<table to be aggregated> ::= <string>


<aggretation operator> ::= linreg | logreg | regtree | min | max |
ap | as | avg


<name of column containing aggregated membership value> ::=
<string>
```

o   Example
```
@aggregatemv:
table == abalone_tr_s;
analytic variables == *;
aggregation operator == linreg;
target variable == y;
output file ==
./metainduction/abalone/02_output/linreg.sql.template;
column alias == mfc;
```

A.4.11. Evaluating Predictions

o   Syntax
```
<evaluate correlations> ::=
@evaluate:
table == <table containing resulting prediction> ;
<analytic variables> ::= <column> | <column list> | <all columns>
;
target variable == <table column>;
<optional result table>
<optional output file>


<optional result table> ::=
```

```
result table == <table for storing evaluation results> | <empty> ;

<table for storing evaluation results> ::= <string>

<optional output file> ::=
output file == <file for storing evaluation results> | <empty> ;

<file for storing evaluation results> ::= <string>
```

o   Example

```
@evaluate:
table == abalone_te_rt;
target variable == y;
analytic variable == mfc;
result table == evaluations;
output file == metainduction/anneal/attreval.csv;
```

A.4.12. Data Preparation

The following example IFCL code shows how to prepare datasets for
training and evaluation using the IFCL `load` action. It shows also how
to split training and test data randomly.

o   Example:

```
@connect:
HostName   == localhost;
SID        == XE;
Port       == 1521;
UserName   == ME;
Password   == passwort;

@load:
file == ./metainduction_num/sensory/00_data/00_data.csv;
table == sensory_load;
delimiter == ,;
attributes ==
Occasion,Judges,Interval,Sittings,Position,Squares,Rows_,Columns,
Halfplot,Trellis,Method,Score;
types == c,c,c,c,c,c,c,c,c,c,c,n;
null == ?;
```

```
@drop: table == sensory_split;


@execsql: command ==
create table sensory_split as
select
Occasion,Judges,Interval,Sittings,Position,Squares,Rows_,Columns,
Halfplot,Trellis,Method,Score as y,
# percent rank fuzzification #
percent_rank() over (order by score) as y_p,
# linear fuzzification #
(score - min(score) over()) / (max(score) over()-min(score)
over()) as y_l,
case when dbms_random.value <= 0.667 then 1 else 0 end as
is_training
# random split of data for training and test sets #
from sensory_load;


@drop: table == sensory_tr;


# create table with training set #
@execsql: command ==
create table sensory_tr as
select * from sensory_split where is_training = 1;


@drop: table == sensory_te;


# create table with test set #
@execsql: command ==
create table sensory_te as
select * from sensory_split where is_training = 0;
```

A.4.13. Attribute Selection

The following example IFCL code shows how to select relevant
attributes from a relation regarding a target attribute.

o   Example:

```
@connect:
HostName   == localhost;
```

```
SID        == XE;
Port       == 1521;
UserName   == ME;
Password   == passwort;

@inducemf: # membership function induction #
table == sensory_tr;
target variable == y;
analytic variables == *;
induction == nlr;
output file ==
./metainduction_num/sensory/02_output/nlr.sql.template;

@classify: # inductive attribute fuzzification #
classified table == sensory_tr;
template file ==
./metainduction_num/sensory/02_output/nlr.sql.template;
output table == sensory_tr_nlr;

@drop: table == attribute_selection;

@evaluate: # ranking of correlation of fuzzified attribute values
and target #
table == sensory_tr_nlr;
target variable == y;
analytic variables == *;
output file == metainduction_num/sensory/attreval.csv;
result table == attribute_selection;
```

A.5. Curriculum Vitae

Michael Alexander Kaufmann
from Recherswil SO
born on September 12th, 1978:

1993 – 1998:	College of Economics, Bern-Neufeld
1998 – 1999:	International Studies in Los Angeles, California
1999 – 2003:	Bachelor of Science in Computer Science, Fribourg
2003 – 2005:	Master of Science in Computer Science, Fribourg
2005 – 2008:	Data Warehouse Analyst at PostFinance, Bern
2008 – 2011:	Data Architect at Swiss Mobiliar Insurance, Bern
2011 – 2012:	Consultant at Nationale Suisse Insurance, Basel
2008 – 2012:	PhD in Computer Science, Fribourg

A.6. Key Terms and Definitions

IFC	Inductive fuzzy classification: Assigning individuals to fuzzy sets for which membership function is generated from data so that the membership degrees provide support for an inductive inference.
ITF	Inductive target fuzzification: Transformation of a numerical target variable into a membership degree to a fuzzy class.
MFI	Membership function induction: Derivation of an inductive mapping from attribute values to membership degrees based on available data.
IFCL	Inductive fuzzy classification language: Software prototype implementing IFC.
NLR	Normalized likelihood ratio: A proposed formula for MFI.
NLD	Normalized likelihood difference: Another proposed formula for MFI.
IAF	Inductive attribute fuzzification: Transformation of attribute values into membership degrees using induced membership functions.
Zadehan Variable	A variable with a range of $[0,1]$ representing truth values, analogous to Boolean variables with a range of $\{0,1\}$.
Zadehan Logic	A framework for reasoning with gradual truth values in the interval between 0 and 1.