Data Mining with Fuzzy Methods: Status and Perspectives

Rudolf Kruse, Detlef Nauck, and Christian Borgelt

Department of Knowledge Processing and Language Engineering
Otto-von-Guericke-University of Magdeburg
Universitätsplatz 2, D-39106 Magdeburg, Germany

E-mail: kruse@iws.cs.uni-magdeburg.de

Abstract: Data mining is the central step in a process called knowledge discovery in databases, namely the step in which modeling techniques are applied. Several research areas like statistics, artificial intelligence, machine learning, and soft computing have contributed to its arsenal of methods. In this paper, however, we focus on fuzzy methods for rule learning, information fusion, and dependency analysis. In our opinion fuzzy approaches can play an important role in data mining, because they provide comprehensible results (although this goal is often neglected—maybe because it is sometimes hard to achieve with other methods). In addition, the approaches studied in data mining have mainly been oriented at highly structured and precise data. However, we expect that the analysis of more complex heterogeneous information source like texts, images, rule bases etc. will become more important in the near future. Therefore we give an outlook on information mining, which we see as an extension of data mining to treat complex heterogeneous information sources, and argue that fuzzy systems are useful in meeting the challenges of information mining.

Keywords: data mining, fuzzy system, information mining, neuro-fuzzy systems

1 Introduction: Data Mining

Due to modern information technology, which produces ever more powerful computers every year, it is possible today to collect, store, transfer, and combine huge amounts of data at very low costs. Thus an ever-increasing number of companies and scientific and governmental institutions can afford to build up large archives of documents and other data like numbers, tables, images, and sounds. However, exploiting the information contained in these archives in an intelligent way turns out to be fairly difficult. In contrast to the abundance of data there is a lack of tools that can transform these data into useful information and knowledge. Although a user often has a vague understanding of his data and their meaning and can usually formulate hypotheses and guess dependencies, he rarely knows: where to find the "interesting" or "relevant" pieces of information, whether these pieces of information support his hypotheses and models, whether (other) interesting phenomena are hidden in the data, which methods are best suited to find the needed pieces of information in a fast and reliable way, and how the data can be translated into human notions that are appropriate for the context in which they are needed.

In reply to these challenges a new area of research has emerged, which has been named "knowledge discovery in databases" or "data mining". In [13] we read

> Knowledge discovery in databases (KDD) is a research area that considers the analysis of large databases in order to identify valid, useful, meaningful, unknown, and unexpected relationships.

Often data mining is restricted to the application of discovery and modeling techniques within the KDD process. It is an interdisciplinary field that employs methods from statistics, soft computing, artificial intelligence and machine learning. Usually data mining is defined by a set of tasks [13, 27], which include at least segmentation (e.g. what kind of customers does a company have?), classification (e.g. is this person a prospective customer?), concept description (e.g. what attributes describe a prospective customer?), prediction (e.g. what value will the stock index have tomorrow?), deviation analysis (e.g. why has the behaviour of customers changed?), and dependency analysis (e.g. how does marketing influence customer behaviour?)

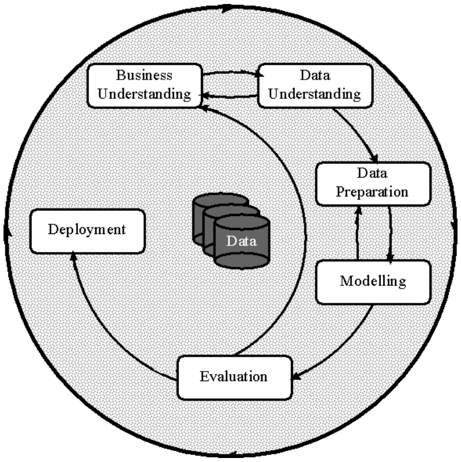

Figure 1: The CRISP-DM Model.

Data mining is of an exploratory nature and can also be seen as exploratory data analysis with a special focus on large data collections. It is quite possible that the questions we want to answer with data mining methods are not clear from the beginning. During the analysis process new questions my arise and we may have to repeat it several times, possibly applying different methods each time.

Some well-known analysis methods and tools that are used in data mining are, for example, statistics (regression analysis, discriminant analysis etc.), time series analysis, decision trees, (fuzzy) cluster analysis, neural networks, inductive logic programming, and association rules. In this paper we concentrate on fuzzy methods in data mining and show where and how they can be used. A survey of commercial data mining tools can be found, for instance, in [18].

Fuzzy set theory provides excellent means to model the "fuzzy" boundaries of linguistic terms by introducing gradual memberships. In contrast to classical set theory, in which an object or a case either is a member of a given set (defined, e.g., by some property) or not, fuzzy set theory makes it possible that an object or a case belongs to a set only to a certain degree, thus modeling the penumbra of the linguistic term describing the property that defines the set [24]. Interpretations of membership degrees include *similarity*, *preference*, and *uncertainty* [11]: They can state how similar an object or case is to a prototypical one, they can indicate preferences between suboptimal solutions to a problem, or they can model uncertainty about the true situation, if this situation is described in imprecise terms. In general, due to their closeness to human reasoning, solutions obtained using fuzzy approaches are easy to understand and to apply. Due to these strengths, fuzzy systems are the method of choice, if linguistic, vague, or imprecise information has to be modeled [25].

Data mining is concerned with the analysis of large but homogeneous data. Our information-oriented world, however, demands that we also automate the analysis of more complex information sources like texts, images, audio and video streams, rule bases provided by experts etc. We suggest to use the term *information mining* to describe a process analogous to KDD and data mining but focused on complex and heterogeneous sources of information. In section 5 we give an outlook on information mining.

2 Fuzzy Sets in Data Mining

The research in knowledge discovery in databases and data mining has lead to a large number of suggestions for a general model of the knowledge discovery process. A recent suggestion for such a model, which can be expected to have considerable impact, since it is backed by several large companies like NCR and DaimlerChrysler, is the CRISP-DM model (CRoss Industry Standard Process for Data Mining) [9].

The basic structure of this process model is depicted in figure 1. The circle indicates that data mining is essentially a circular process, in which the evaluation of the results can trigger a re-execution of the data preparation and model generation steps. In this process, fuzzy set methods can profitably be applied in several phases:

The *business understanding* and *data understanding* phases are usually strongly human centered and only little automation can be achieved here. These phases serve mainly to define the goals of the knowledge discovery project, to estimate its potential benefit, and to identify and collect the necessary data. In addition, background domain knowledge and meta knowledge about the data is gathered. In these phases, fuzzy set methods can be used to formulate, for instance, the background domain knowledge in vague terms, but still in a form that can be used in a subsequent modeling phase. Furthermore, fuzzy database queries are useful to find the data needed and to check whether it may be useful to take additional, related data into account.

In the *data preparation* step, the gathered data is cleaned, transformed and maybe properly scaled to produce the input for the modeling techniques. In this step fuzzy methods may, for example, be used to detect outliers, e.g., by *fuzzy clustering* the data [3, 21] and then finding those data points that are far away from the cluster prototypes.

The *modeling* phase, in which models are constructed from the data in order, for instance, to predict future developments or to build classifiers, can, of course, benefit most from fuzzy data analysis approaches. These approaches can be divided into two classes. The first class, *fuzzy data* analysis, consists of approaches that analyze fuzzy data—data derived from imprecise measurement instruments or from the descriptions of human domain experts [6]. An example from our own research is the induction of *possibilistic graphical models* from data which complements the induction of the well-known probabilistic graphical models. The second class, fuzzy *data analysis*, consists of methods that use fuzzy techniques to structure and analyze crisp data, for instance, *fuzzy clustering* for data segmentation and rule generation and *neuro-fuzzy systems* for rule generation.

In the *evaluation* phase, in which the results are tested and their quality assessed, the usefulness of fuzzy modeling methods becomes most obvious. Since they yield interpretable systems, they can easily be checked for plausibility against the intuition and expectations of human experts. In addition, the results can provide new insights into the domain under consideration, in contrast to, e.g., pure neural networks, which are black boxes.

To illustrate the usefulness of fuzzy data analysis approaches, in the following sections we discuss generating fuzzy rules from data, information fusion in fuzzy systems and learning possibilistic graphical models in a little more detail.

3 Learning Fuzzy Rule Bases

One possible application of fuzzy systems in data mining is the induction of fuzzy rules in order to interpret the underlying data linguistically. To describe a fuzzy system completely we need to determine a rule base (structure) and fuzzy partitions (parameters) for all variables. The data driven induction of fuzzy systems by simple heuristics based on local computations is usually called *neuro-fuzzy* [28]. If we apply such techniques, we must be aware of the trade-off between precision and interpretability. A solution is not only judged by its accuracy, but also—if not primarily—by its simplicity and readability: A user of a fuzzy system must be able to comprehend the rule base.

Important points for the interpretability of a fuzzy system are that there are only few fuzzy rules in the rule base. Each rule should use only a few variables and the variables should be partitioned by few meaningful fuzzy sets. It is also important that no linguistic label is represented by more than one fuzzy set.

There are several ways to induce the structure of a fuzzy system. Cluster-oriented and hyperbox-oriented approaches to fuzzy rule learning create rules and fuzzy sets at the same time. Structure-oriented approaches need initial fuzzy partitions to create a rule base [29].

Cluster-oriented rule learning approaches are based on fuzzy cluster analysis [3, 21], i.e., the learning process is unsupervised. Hyperbox-oriented approaches use a supervised learning algorithm that tries to cover the training data by overlapping hyperboxes [2]. Fuzzy rules are created in both approaches by projection of clusters or hyperboxes. The main problem of both approaches is that each generated fuzzy rule uses individual membership functions and thus the rule base is hard to interpret. Cluster-oriented approaches additionally suffer from a loss of information and can only determine an appropriate number of rules if they are iterated with different fixed rule base sizes.

Structure-oriented approaches avoid all these drawbacks, because they do not search for (hyper-ellipsoidal or hyper-rectangular) clusters in the data space. By providing (initial) fuzzy sets before fuzzy rules are created the data space is structured by a multidimensional fuzzy grid. A rule base is created by selecting those grid cells that contain data. This can be done in a single pass through the training data. This way of learning fuzzy rules was suggested in [35]. Extended versions were used in the the neuro-fuzzy classification system NEFCLASS [28]. NEFCLASS uses a performances measure for the detected fuzzy rules. Thus the size of the rule base can be determined automatically by adding rules ordered by their performance until all training data is covered. The performance measure is also used to compute the best consequent for each rule.

The number of fuzzy rules can also be restricted by including only the best rules in the rule base. It is also possible to use pruning methods to reduce the number of rules and the number of variables used by the rules. In order to obtain meaningful fuzzy partitions, it is better to create rule bases by structure-oriented learning than by cluster-oriented or by hyperbox-oriented rule learning. The latter two approaches create individual fuzzy sets for each rule and thus provide less interpretable solutions. Structure-oriented methods allow the user to provide appropriate fuzzy partitions in advance such that all rules share the same fuzzy sets. Thus the induced rule base can be interpreted well.

After the rule base of a fuzzy system has been generated, we must usually train the membership function in order to improve the performance. In NEFCLASS, for example, the fuzzy sets are tuned by a simple backpropagation-like procedure. The algorithm does not use gradient-descent, because the degree of fulfillment of a fuzzy rule is determined by the minimum and non-continuous membership function may be used. Instead a simple heuristics is used that results in shifting the fuzzy sets and in enlarging or reducing their support.

The main idea of NEFCLASS is to create comprehensible fuzzy classifiers, by ensuring that fuzzy sets cannot be modified arbitrarily during learning. Constraints can be applied in order to make sure that the fuzzy sets still fit their linguistic labels after learning. For the sake of interpretability we do not want adjacent fuzzy sets to exchange positions, we want the fuzzy sets to overlap appropriately etc.

The most recent JAVA implementation of the NEFCLASS approach to generate fuzzy classifiers from data uses structure-oriented fuzzy rule learning and can automatically determine the number of rules. The training data can contain missing values, which are handled without imputation. The data can also consist of both numeric *and* symbolic attributes. The fuzzy set learning is constrained in order to obtain interpretable solutions. After training automatic pruning strategies can be applied to reduce the size of the rule base and to further enhance the interpretability of the fuzzy classifier. The tool is called NEFCLASS-J and can be obtained at `http://fuzzy.cs.uni-magdeburg.de`.

This tool can also be used in the emerging area of information fusion. *Information fusion* refers to the acquisition, processing, and merging of information originating from multiple sources to provide a better insight and understanding of the phenomena under consideration. There are several levels of information fusion. Fusion may take place at the level of data acquisition, data pre-processing, data or knowledge representation, or at the model or decision making level. On lower levels, where raw data is involved, the term (sensor) *data fusion* is preferred. For a conceptual and comparative study of fusion strategies for expert opinions in various calculi of uncertainty see [10, 12, 17].

NEFCLASS and neuro-fuzzy systems in general can be used to integrate expert knowledge in form of fuzzy rules and information obtained from data. If prior knowledge about the classification problem is available, then the rule base of the fuzzy classifier can be initialized with suitable fuzzy rules before rule learning is invoked to complete the rule base. If the algorithm creates a rule from data that contradicts with an expert rule then three options are available:

- always prefer expert rule,
- always prefer the learned rule,
- select the rule with the larger performance value.

In NEFCLASS we determine the performance of all rules over the training data and in case of contradiction the better rule prevails. This reflects fusion of expert opinions and observations.

Because NEFCLASS is able to resolve conflicts between rules based on rule performance, it is also able to fuse expert opinions on fuzzy rule level. If rule bases from different experts are available, they can be entered as prior knowledge. They will be fused into one rule base and contradictions are resolved automatically by deleting from each pair of contradicting rules the rule with lower performance.

After all contradictions between expert rules and rules learned from data were resolved, usually not all rules can be included into the rule base, because its size is limited by some criterion. In this case we must decide whether to include expert rules in any case, or to include rules by descending performances values. For an application in the context of stock prediction that is based on another neuro-fuzzy approach, which is implemented in the neural network tool SENN, see [34].

The decision on that option depends on the trust we have in the experts knowledge and in the training data. A mixed approach can be used, e.g. include the best expert rules and then use the best learned rules to complete the rule base.

A similar decision must be made, when the rule base is pruned after training, i.e. is it acceptable to remove an expert rule during pruning, or must such rules remain in the rule base. In NEFCLASS expert rules and rules induced from data are not treated differently.

4 Dependency Analysis

Since reasoning in multi-dimensional domains tends to be infeasible in the domains as a whole—and the more so, if uncertainty and imprecision are involved—decomposition techniques, that reduce the reasoning process to computations in lower-dimensional subspaces, have become very popular. In the field of graphical modeling [23, 36], *decomposition* is based on dependence and independence relations between the attributes or variables that are used to describe the domain under consideration. The structure of these dependence and independence relations are represented as a graph (hence the name graphical models), in which each node stands for an attribute and each edge for a direct dependence between two attributes. The precise set of dependence and (conditional) independence statements that hold in the modeled domain can be read from the graph using simple graph theoretic criteria, for instance, *d*-separation, if the graph is a directed one, or simple separation, if the graph is undirected.

The conditional independence graph (as it is also called) is, however, only the *qualitative* or *structural component* of a graphical model. To do reasoning, it has to be enhanced by a *quantitative component* that provides confidence information about the different points of the underlying domain. This information can often be represented as a distribution function on the underlying domain, for example, a probability distribution, a possibility distribution, a mass distribution etc. W.r.t. this quantitative component, the conditional independence graph describes a *factorization* of the distribution function on the domain as a whole into conditional or marginal distribution functions on lower-dimensional subspaces.

Graphical models make reasoning much more efficient, because propagating the evidential information about the values of some attributes to the unobserved ones and computing the marginal distributions for the unobserved attributes can be implemented by locally communicating node and edge processors in the conditional independence graph.

Using graphical models to facilitate reasoning in multi-dimensional domains has originated in the probabilistic setting. Bayesian networks [30], which are based on directed conditional independence graphs, and Markov networks [26], which are based on undirected graphs, are the most prominent examples. However, this approach has also been generalized to be usable with other uncertainty calculi [32], for instance, in the so-called valuation-based networks [33] and has been implemented, for example, in PULCINELLA [31]. Due to their connection to fuzzy systems, which in the past have successfully been applied to solve control problems, and due to their ability to deal not only with uncertainty but also with imprecision, recently possibilistic networks also gained some attention. They can be based on the context-model interpretation of a degree of possibility, which focuses on imprecision [14], and have been implemented, for example, in POSSINFER [16, 24].

For some time the standard approach to construct a graphical model has been to let a human domain expert specify the dependency structure of the considered domain. This provided the conditional independence graph. Then the human domain expert had to estimate the necessary conditional or marginal distribution functions, which then formed the quantitative component of the graphical model. This approach, however, can be tedious and time consuming, especially, if the domain under consideration is large. In addition, it may be impossible to carry it out, if no or only vague knowledge is available about the dependence and independence relations that hold in the domain to be modeled. Therefore recent research has concentrated on learning graphical models from databases of sample cases.

Due to the origin of graphical modeling research in probabilistic reasoning (see above), the most widely known methods are, of course, learning algorithms for Bayesian or Markov networks [7, 8, 20]. However, these approaches—as probabilistic approaches do in general—suffer from certain deficiencies, if imprecise information, understood as set-valued data, has to be taken into account. For this reason learning possibilistic networks from data is a noteworthy alternative, the theory of which can be developed in close analogy to the probabilistic case [15, 16, 4, 5, 22]. These methods can be used to do dependency analysis, even if the data to analyze is highly imprecise and thus offer interesting perspectives for future research.

5 Perspective: Information Mining

Although the standard definition of knowledge discovery and data mining [13] only speaks of discovery in *data*, thus not restricting the type and the organization of the data to work on, it has to be admitted that research up to now concentrated on highly structured data. Usually a minimal requirement is relational data. Most methods (e.g. classical methods like decision trees and neural networks) even demand as input a single uniform table, i.e., a set of tuples of attribute values. It is obvious, however, that this paradigm is hardly adequate for mining image or sound data or even textual descriptions, since it is inappropriate to see such data as, say, tuples of picture elements. Although such data can often be treated successfully by transforming them into structured tables using feature extraction, it is not hard to see that methods are needed which yield, for example, descriptions of what an image depicts, and other methods which can make use of such descriptions e.g. for retrieval purposes.

Another important point to be made is the following: The fact that pure neural networks are often seen as data mining methods, although their learning result (matrices of numbers) is hardly interpretable, shows that in contrast to the standard definition the goal of *understandable* patterns is often neglected. Of course, there are applications where comprehensible results are not needed and, for example, the prediction accuracy of a classifier is the only criterion of success. Therefore interpretable results should not be seen as a *conditio sine qua non*. However, our own experience—gathered in several cooperations with industry—is that modern technologies are accepted more readily, if the methods applied are easy to understand and the results can be checked against human intuition. In addition, if we want to gain insight into a domain, training, for instance, a neural network is not of much help.

Therefore we suggest to concentrate on *information mining*, which we see as an extension of data mining and which can be defined in analogy to the KDD definition given in [13] as follows:

> Information mining is the non-trivial process of identifying valid, novel, potentially useful, and *understandable* patterns in *heterogeneous information sources*.

The term *information* is thus meant to indicate two things: In the first place, it points out that the heterogeneous sources to mine can already provide *information*, understood as expert background knowledge, textual descriptions, images and sounds etc., and not only raw data. Secondly, it emphasizes that the results must be *comprehensible* ("must provide a user with information"), so that a user can check their plausibility and can get insight into the domain the data comes from.

For research this results in the challenges

- to develop theories and scalable techniques that can extract knowledge from large, dynamic, multi-relational, and multi-medial information sources,
- to close the semantic gap between structured data and human notions and concepts, i.e., to be able to translate computer representations into human notions and concepts and vice versa.

The goal of fuzzy systems has always been to model human expert knowledge and to produce systems that are easy to understand. Therefore we expect fuzzy systems technology to play a prominent role in the quest to meet these challenges. In the following we try to point out how fuzzy techniques can help to do information mining.

6 Conclusions

In knowledge discovery and data mining as it is, there is a tendency to focus on purely data-driven approaches in a first step. More model-based approaches are only used in the refinement phases (which in industry are often not necessary, because the first successful approach wins—and the winner takes all). However, to arrive at truly useful results, we must take background knowledge and, in general, non-numeric information into account and we must concentrate on comprehensible models.

The complexity of the learning task, obviously, leads to a problem: When learning from information, one must choose between (often quantitative) methods that achieve good performance and (often qualitative) models that explains what is going on to a user. This is another good example of Zadeh's principle of the incompatibility between precision and meaning. Of course, precision and high performance are important goals. However, in the most successful fuzzy applications in industry such as intelligent control and pattern classification, the introduction of fuzzy sets was motivated by the need for more human-friendly computerized devices that help a user to formulate his knowledge and to clarify, to process, to retrieve, and to exploit the available information in a most simple way. In order to achieve this user-friendliness, often certain (limited) reductions in performance and solution quality are accepted.

So the question is: What is a good solution from the point of view of a user in the field of information mining? Of course, correctness, completeness, and efficiency are important, but in order to manage systems that are more and more complex, there is a constantly growing demand to keep the solutions conceptually simple and understandable. This calls for a formal theory of utility in which the simplicity of a system is taken into account. Unfortunately such a theory is extremely hard to come by, because for complex domains it is difficult to measure the degree of simplicity and it is even more difficult to assess the gain achieved by making a system simpler. Nevertheless, this is a lasting challenge for the fuzzy community to meet.

References

[1] S.K. Andersen, K.G. Olesen, F.V. Jensen, and F. Jensen. HUGIN — A Shell for Building Bayesian Belief Universes for Expert Systems. *Proc. 11th Int. J. Conf. on Artificial Intelligence (IJCAI'89, Detroit, MI, USA)*, 1080–1085. Morgan Kaufman, San Mateo, CA, USA 1989

[2] M. Berthold and K.-P. Huber. Constructing Fuzzy Graphs from Examples. *Int. J. of Intelligent Data Analysis*, 3(1), 1999. Electronic journal (http://www.elsevier.com/locate/ida)

[3] J.C. Bezdek, J.M. Keller, R. Krishnapuram, and N. Pal. *Fuzzy Models and Algorithms for Pattern Recognition and Image Processing*. The Handbooks on Fuzzy Sets. Kluwer, Dordrecht, Netherlands 1999

[4] C. Borgelt and R. Kruse. Evaluation Measures for Learning Probabilistic and Possibilistic Networks. *Proc. 6th IEEE Int. Conf. on Fuzzy Systems (FUZZ-IEEE'97, Barcelona, Spain)*, 669–676. IEEE Press, Piscataway, NJ, USA 1997

[5] C. Borgelt, J. Gebhardt, and R. Kruse. Chapter F1.2: Inference Methods. In: E. Ruspini, P. Bonissone, and W. Pedrycz, eds. *Handbook of Fuzzy Computation*. Institute of Physics Publishing Ltd., Philadelphia, PA, USA 1998

[6] C. Borgelt, J. Gebhardt, and R. Kruse. Fuzzy Methoden in der Datenanalyse. In: R. Seising, ed. *Fuzzy Theorie und Stochastik — Modelle und Anwendungen in der Diskussion*, 370–386. Vieweg, Braunschweig, Germany 1999

[7] C.K. Chow and C.N. Liu. Approximating Discrete Probability Distributions with Dependence Trees. *IEEE Trans. on Information Theory* 14(3) 462–467. IEEE Press, Piscataway, NJ, USA 1968

[8] G. Cooper and E. Herskovits. A Bayesian Method for the Induction of Probabilistic Networks from Data. *Machine Learning* **9**:309–347. Kluwer, Dordrecht, Netherlands 1992

[9] P. Chapman, J. Clinton, T. Khabaza, T. Reinartz, and R. Wirth. *The CRISP-DM Process Model*, 1999 Available from `http://www.ncr.dk/CRISP/`.

[10] R.M. Cooke. *Experts in Uncertainty*. Oxford University Press, Oxford, United Kingdom 1991

[11] D. Dubois, H. Prade, and R.R. Yager. Information Engineering and Fuzzy Logic. *Proc. 5th IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'96, New Orleans, LA, USA)*, 1525–1531. IEEE Press, Piscataway, NJ, USA 1996

[12] D. Dubois, H. Prade, and R.R. Yager. Merging Fuzzy Information. In: J.C. Bezdek, D. Dudois, and H. Prade, eds. *Approximate Reasoning and Fuzzy Information Systems*, The Handbooks of Fuzzy Sets, chapter 6, 335–402. Kluwer, Dordrecht, Netherlands 1999 (to appear)

[13] U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, eds. *Advances in Knowledge Discovery and Data Mining*. MIT Press, Menlo Park, CA, USA 1996

[14] J. Gebhardt and R. Kruse. The Context Model — An Integrating View of Vagueness and Uncertainty. *Int. Journal of Approximate Reasoning* 9:283–314. North-Hollan, Amsterdam, Netherlands 1993

[15] J. Gebhardt and R. Kruse. Learning Possibilistic Networks from Data. *Proc. 5th Int. Workshop on Artificial Intelligence and Statistics*, 233–244. Fort Lauderdale, FL, USA 1995

[16] J. Gebhardt and R. Kruse. Tightest Hypertree Decompositions of Multivariate Possibility Distributions. *Proc. Int. Conf. on Information Processing and Management of Uncertainty in Knowledge-based Systems (IPMU'96)*, 923–927. Granada, Spain 1996

[17] J. Gebhardt and R. Kruse. Parallel Combination of Information Sources. In: D. Gabbay and P. Smets, eds. *Belief Change*, Vol. 3 of *Handbook of Defeasible Reasoning and Uncertainty Management Systems*, 329–375. Kluwer, Dordrecht, Netherlands 1998

[18] P. Gentsch. *Data Mining Tools: Vergleich marktgängiger Tools.* WHU Koblenz, Germany 1999

[19] D. Heckerman. *Probabilistic Similarity Networks.* MIT Press, Cambridge, MA, USA 1991

[20] D. Heckerman, D. Geiger, and D.M. Chickering. Learning Bayesian Networks: The Combination of Knowledge and Statistical Data. *Machine Learning* 20:197–243, Kluwer, Dordrecht, Netherlands 1995

[21] F. Höppner, F. Klawonn, R. Kruse, and T. Runkler. *Fuzzy Cluster Analysis.* J. Wiley & Sons, Chichester, United Kingdom 1999

[22] R. Kruse and C. Borgelt. Data Mining with Graphical Models. *Proc. Computer Science for Environmental Protection (12th Int. Symp. "Informatik fr den Umweltschutz", Bremen 1998)*, Vol. 1:17–30. Metropolis-Verlag, Marburg, Germany 1998

[23] R. Kruse, E. Schwecke, and J. Heinsohn. *Uncertainty and Vagueness in Knowledge-based Systems: Numerical Methods.* Series: Artificial Intelligence, Springer, Berlin, Germany 1991

[24] R. Kruse, J. Gebhardt, and F. Klawonn. *Foundations of Fuzzy Systems.* J. Wiley & Sons, Chichester, United Kingdom 1994

[25] R. Kruse, C. Borgelt, and D. Nauck. Fuzzy Data Analysis: Challenges and Perspectives. *Proc. 8th IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'99, Seoul, Korea).* IEEE Press, Piscataway, NJ, USA 1999 (to appear)

[26] S.L. Lauritzen and D.J. Spiegelhalter. Local Computations with Probabilities on Graphical Structures and Their Application to Expert Systems. *Journal of the Royal Stat. Soc., Series B* **2**(50):157–224. Blackwell, Oxford, United Kingdom 1988

[27] G. Nakhaeizadeh. Wissensentdeckung in Datenbanken und Data Mining: Ein Überblick. In: G. Nakhaeizadeh, ed. *Data Mining: Theoretische Aspekte und Anwendungen*, 1–33. Physica-Verlag, Heidelberg, Germany 1998

[28] D. Nauck, F. Klawonn, and R. Kruse. *Foundations of Neuro-Fuzzy Systems.* J. Wiley & Sons, Chichester, United Kingdom 1997

[29] D. Nauck and R. Kruse. Chapter D.2: Neuro-fuzzy Systems. In: E. Ruspini, P. Bonissone, and W. Pedrycz, eds. *Handbook of Fuzzy Computation.* Institute of Physics Publishing Ltd., Philadelphia, PA, USA 1998

[30] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference (2nd edition).* Morgan Kaufmann, San Mateo, CA, USA 1992

[31] A. Saffiotti and E. Umkehrer. PULCINELLA: A General Tool for Propagating Uncertainty in Valuation Networks. *Proc. 7th Conf. on Uncertainty in Artificial Intelligence*, 323–331. Morgan Kaufmann, San Mateo, CA, USA 1991

[32] G. Shafer and P.P. Shenoy. *Local Computations in Hypertrees*, Working paper 201. School of Business, University of Kansas, Lawrence, KS, USA 1988

[33] P.P. Shenoy. Valuation-based Systems: A Framework for Managing Ancertainty in Expert Systems. In: L.A. Zadeh and J. Kacprzyk, eds. *Fuzzy Logic for the Management of Uncertainty*, 83–104. J. Wiley & Sons, New York, NY, USA 1992

[34] S. Siekmann, J. Gebhardt, and R. Kruse. Information Fusion in the Context of Stock Index Prediction. *Proc. European Conference on Symbolic and Quantitative Approaches to Uncertainty (ECSQARU'99, London, United Kingdom).* Springer, Heidelberg, Germany 1999 (to appear)

[35] L.X. Wang and J.M. Mendel. Generating Fuzzy Rules by Learning from Examples. *IEEE Trans. on Systems, Man, and Cybernetics*, 22(6):1414–1427. IEEE Press, Piscataway, NJ, USA 1992

[36] J. Whittaker. *Graphical Models in Applied Multivariate Statistics.* J. Wiley & Sons, Chichester, United Kingdom 1990