Designing Highly Interpretable Fuzzy Rule-Based Systems with Integration of Expert and Induced Knowledge

José M. Alonso
European Centre
for Soft Computing
Campus Universitario
de Mieres, 33600
Asturias, Spain
jose.alonso@softcomputing.es

Luis Magdalena
European Centre
for Soft Computing
luis.magdalena@softcomputing.es

Serge Guillaume
Cemagref Montpellier
BP 5095, 34033
Montpellier Cedex 5
France
serge.guillaume@montpellier.cemagref.fr

Abstract

This work describes a new methodology for fuzzy system modeling focused on maximizing the interpretability while keeping high accuracy. In order to get a good interpretability-accuracy trade-off, it considers the combination of both expert knowledge and knowledge extracted from data. Both types of knowledge are represented using the fuzzy logic formalism, in the form of linguistic variables and rules. The integration process is made carefully at both levels variables and rules, avoiding contradictions and/or redundancies. Results obtained in a well-known benchmark classification problem show the methodology ability to generate highly interpretable knowledge bases with a good accuracy, comparable to that achieved by other methods.

Keywords: Linguistic fuzzy modeling, interpretability-accuracy trade-off, expert-data integration.

1 Introduction

The interpretability, also called understandability, comprehensibility, intelligibility, or transparency, of fuzzy rule-based systems (FRBSs) is of prime importance. It is a desirable property for lots of applications, but it is an essential requirement for those with high human interaction such as decision support systems in medicine, robotics, etc.

The simplest way to get interpretable FRBSs consists in building them from expert knowledge. A domain expert is able to provide us with a global view of the system behavior, describing the most influential variables and using them in a few basic rules. However, dealing with complex systems the expert knowledge is not enough. The interaction among many variables is difficult to formalize by an expert. Fortunately, systems can also be built using induced knowledge, i.e., knowledge extracted from experimental data which are likely to give a good image of interaction between variables.

Since expert and induced knowledge convey complementary information (expert knowledge is usually general while induced knowledge is quite specific according to the available data) their combination seems a good choice. For instance, Browne et al. [7] explain how to fuse human knowledge-elicitation and data-mining in an industrial plant. The expert-data integration has been also used to yield clinical prediction rules in medical applications [17]. When integrating expert and induced knowledge there are two main policies:

- **FETD: First Expert, Then Data**. Build an expert knowledge base (KB), and then complete it using the knowledge extracted from experimental data.
- **FDTE: First Data, Then Expert**. Build an induced KB, and then look for an expert able to evaluate and refine it.

L. Magdalena, M. Ojeda-Aciego, J.L. Verdegay (eds): Proceedings of IPMU'08, pp. 682–689
Torremolinos (Málaga), June 22–27, 2008

Anyhow, beyond these two main options, there are infinitely many intermediate solutions which should be implemented as iterative procedures, where expert and data are integrated all along the process. Nevertheless, as far as we know there isn't a formal methodology explaining how to make such integration. Therefore, authors of this contribution have made a great effort in the last years to formalize a new methodology, in order to build FRBSs combining both expert and data while maximizing the interpretability of the final model. The entire process is made up of several blocks which have been already presented in other publications. The goal of this work is to give an overview of the full methodology [1], describing both the general framework and one of its possible implementations. Notice that it can be seen as a dynamic puzzle that comprises interchangeable pieces. Thus, depending on the selected pieces as well as on their internal carrying out, several implementations of the methodology are feasible.

The rest of the paper is structured as follows. Section 2 describes the new methodology. Section 3 explains the experiments made and the obtained results. Finally, section 4 offers some conclusions.

2 An overview of HILK

This contribution proposes a new methodology for building **H**ighly **I**nterpretable **L**inguistic **K**nowledge (HILK) bases on the fuzzy logic formalism.

The starting point is the cooperation framework introduced by Guillaume and Magdalena [13]. They proposed a general framework that includes two hierarchical steps. The first one was thoroughly described in the initial proposal; it consists in building a common fuzzy input space according to both data and expert knowledge. The second step, the integration of expert and induced fuzzy rule bases, was only introduced and it remained an open problem.

This work gives a global overview of the new proposed methodology, from the initial ideas sketched by Guillaume and Magdalena to their final development and test, extending the original proposal with new ideas and functionalities. The whole process is depicted in the figure 1. The global structure (partition and rule levels) has been enhanced adding a third level dedicated to improve the final accuracy-interpretability trade-off. Notice that the figure distinguishes between expert and data domains. Moreover, the entire process is carried out under expert supervision, keeping in mind the interpretability requirement, and following the FETD (first collecting expert knowledge and then adding induced one) policy at every level.

The rest of this section gives a few details about the three main blocks included in the proposal: *Partition Design*, *Rule Integration*, and *KB Improvement*.

2.1 Partition design

The readability of fuzzy partitioning is a prerequisite to build interpretable FRBSs. The use of linguistic variables favors the readability. However, linguistic constraints must be superimposed to the fuzzy partition definition in order to ensure their interpretability. In consequence, each system variable is described by a set of linguistic terms, modeled as fuzzy sets that form Strong Fuzzy Partitions (SFPs) [21]. This kind of partition satisfies all the semantic constraints (distinguishability, normalization, coverage, overlapping, etc.) demanded to be interpretable [9, 11].

The goal of this first block is to define the most influential variables, by means of such SFPs according to both expert knowledge and knowledge extracted from experimental data. Firstly, the expert can provide complete or partial information about the linguistic variables (*Expert Contribution*), and then the definition can be completed with information extracted from data (*Induced Partitions*). Of course, for a given variable, maximum information from the expert, in membership function definition, is desirable (but not always available). The integration is made at three levels:

- **Range.** On the one hand, the expert

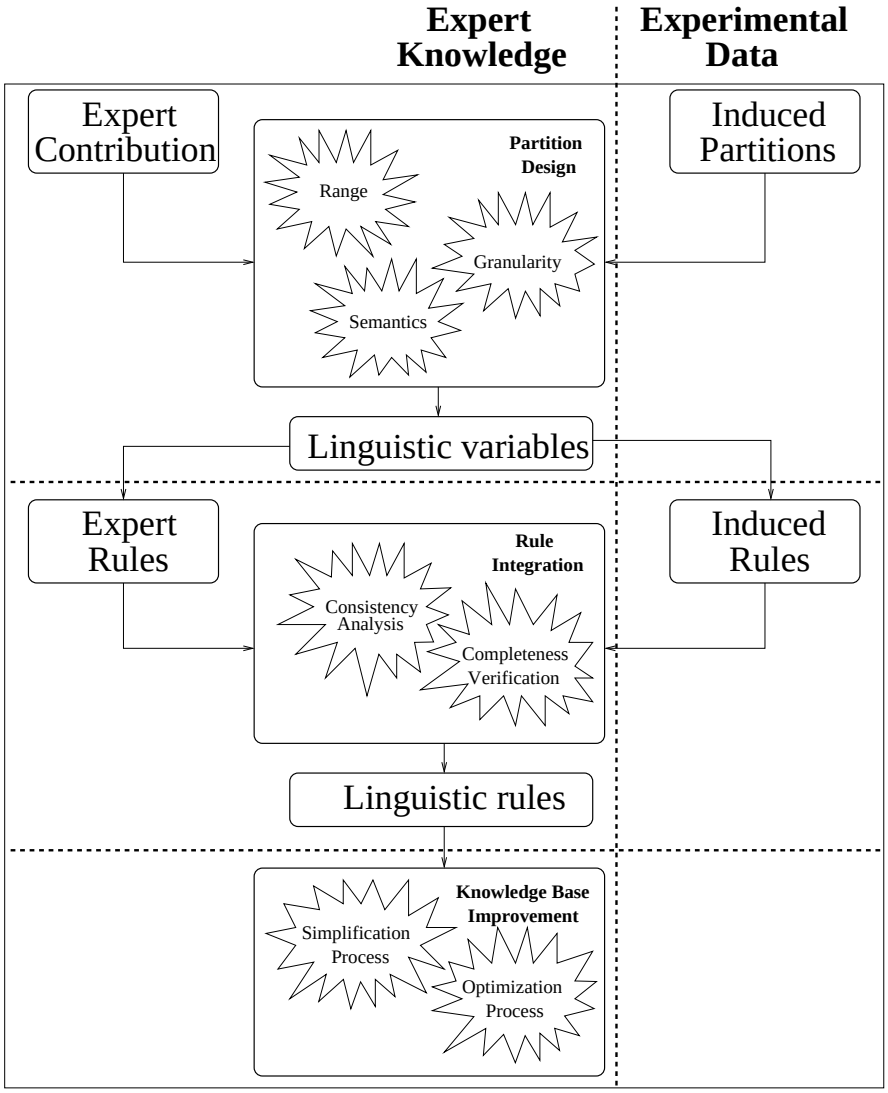


Figure 1: The whole proposed modeling process.

defines the universe of discourse (domain of interest and physical range). On the other hand, data range is automatically derived from the data distribution. Note that the interest range must be included into the physical one to be coherent. Furthermore, if data range differs significantly from the domain of interest given by the expert, the data set will not help to solve the problem.

- **Granularity**. How to find the best-suited number of linguistic terms (labels)? The first option consists in selecting the number of terms the experts need to express their reasoning. In case they are not very confident regarding the granularity, a default number (for instance five or seven) is proposed. Notice that this number will be used for building the initial fuzzy partitions, but it is likely to be reduced by the simplification process that will take place later, after rule integration.

- **Semantics**. Each label is characterized by a prototype, modal point, i.e., the most significant value of the fuzzy set center. This information can also be extracted from experimental data[1]. Once an agreement on the number of terms is

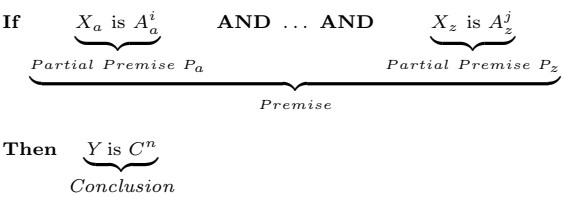
[1]In fact, generating fuzzy partitions from data involves defining the most appropriate shapes for the membership functions, determining the optimum number of linguistic terms in the fuzzy partitions,

reached, the final check is only semantic; it is limited to the modal point positions in the universe of discourse, in relation with their associated semantic labels: Are they possible prototypes according to expert knowledge? Are the differences with the expert-defined modal points acceptable?

The result of this first stage is the definition of a common universe for each of the variables involved in the problem, according to both expert knowledge and data distribution.

2.2 Rule base definition and integration

Once the fuzzy partitions have been designed, they can be used to define fuzzy rules. The considered rules were introduced by Mamdani [18] and they are of the form:



This is a Disjunctive Normal Form (DNF), a disjunction of conjunctions: Rule premises are made up of tuples (*input variable*, *linguistic term*) where X_a is the name of the input variable a, while A_a^i represents the label i of such variable. Notice that the absence of an input in a rule means that variable is not considered in the evaluation of that rule.

First, the expert is invited to make a description of the system behavior, expressing his/her system knowledge as linguistic rules (*Expert Rules*). In addition, rules can be induced from data[2] (*Induced Rules*). Then, the second integration phase (*Rule Integration*) is carried out. Thanks to the common universe previously defined both types of rules use the same linguistic terms defined by the same fuzzy sets. In consequence, rule comparison can be done at the linguistic level. However, since inconsistency and redundancy may appear during the integration of heterogeneous rule bases (composed of expert and induced rules), this integration stage must be made carefully, regarding the most important rule base (RB) features:

- **Consistency Analysis**. The goal is to detect potential conflicting rules [13] by means of a linguistic analysis. Then, a specific handling for these situations is proposed [1].
- **Completeness Verification**. According to [11], *Completeness means that for any possible input vector, at least one rule is fired, there is no inference breaking.* Merging two heterogeneous RBs may yield a unique RB able to manage areas where no knowledge was available in one of the original rule sets [1].

2.3 Knowledge base improvement

After rule integration, the entire KB is consistent and fully operative. The matter now is to evaluate and, if it is possible, to enhance its interpretability-accuracy trade-off. Hence, the first step consists in defining quality indices for measuring both properties [3]. Then, two procedures are run with the aim of getting better trade-off:

- **Simplification Process** [4]. The interpretability is increased without losing either consistency or accuracy, by reducing the number of rules, the premises by rule, and the number of labels, with a controlled loss of accuracy. The objective is to design incomplete, more general, rules while checking consistency and avoiding redundancy in the final RB. Building general rules (as expert rules usually are) makes the system more robust and more interpretable.

and/or locating the fuzzy sets into the universe of discourse. Many algorithms can be found in the specialized literature, for example: *Hierarchical Fuzzy Partitioning* (HFP) [12], and *K-means* [15].

[2]The process of generating rules from data is called rule induction. It aims to produce general statements from partial observations. Many methods are available in the literature [11, 16], but we are only interested in those ones which generate rules sharing the same fuzzy sets, for instance: *Fuzzy Decision Trees* (FDT) [6, 19], and *Wang and Mendel* (WM) [22].

- **Optimization Process** [2]. The accuracy gets better while keeping high interpretability. The procedure will not modify the linguistic description of variables and rules. It only affects the fuzzy partitions, becoming a membership function tuning constrained in order to maintain the SFP property. Many optimization strategies are possible, for instance Genetic Tuning [8].

3 Experimentation

As explained in previous section, many implementations of HILK are feasible depending on the blocks selected from the entire architecture. In addition, lots of algorithms can be used to develop each block. This section illustrates how to apply HILK on a well-known benchmark classification problem, the Wisconsin Breast Cancer Database (WBCD), freely available from the UCI (University of California, Irvine) machine-learning repository[3]. It consists of 683 samples (incomplete patterns with missing values are not taken into consideration) that involve 9 features obtained from fine needle aspirates, for two cancer states (benign or malignant). Hence, it is made up of 9 inputs and 1 output (2 classes).

Notice that expert knowledge about this problem is limited to the names of input variables and output classes. In consequence, these experiments are focused on the automatic generation of KBs from data, allowing a fair comparison with other popular methodologies like Naive Bayes (NB) [14] and C4.5 [20].

3.1 Description of the experiments

The experimentation is outlined as follows. Firstly, all variables involved in the problem are defined, setting a global semantics. Secondly, two different rule bases (RB1 and RB2) are induced. RB1 is made up of a small number of general rules while RB2 includes a larger number of specific rules. Then, both rule bases are merged becoming a unique one. Finally, KB interpretability (Simplification) is improved. Notice that this implementation of HILK disregards two blocks of the full architecture: *Completeness Verification* and *Optimization Process*.

Regarding the fuzzy partitioning we have generated SFPs of different sizes (3, 5 or 7 labels) and different types: HFP [12], K-means [15], and regular (uniform fuzzy partition defined in the range derived from the data distribution). Furthermore, with respect to rule definition two rule induction algorithms were considered: FDT [6, 19], and WM [22]. Moreover, the induction of the second rule set (RB2) can be obtained with either data selection (DS) or not. DS consists in generating a new reduced training set by removing from the whole one the samples managed by RB1. Notice that an item is considered as managed by a rule if its firestrength is higher than a threshold (set to 0.6, in this case).

Finally, bootstrapping was chosen as evaluation methodology because it can be used not only for estimating generalization error but also for estimating confidence bounds [10]. Although there are more sophisticated bootstrap methods, we have used one of the simplest ones. If we could repeat the same experiment thousands of times, then we could characterize it perfectly. However, this is not possible because each run of the experiment takes a long time. Therefore, we decided to repeat each experiment 30 times. Each time, the full data set is randomly divided, taking the 75% of samples as training set and the remainder as test set. Then, both training and test sets are used to build a KB. Thus, we got 30 KBs characterized by their quality indices. The bootstrap method consists in taking randomly 30 of these indices (the same index can be taken several times) and compute the average value. This procedure is repeated 1000 times. As a result, we estimate, in an inferential way, what we would obtain if the experiments were repeated 1000 times. Finally, the 1000 inferred values are ranked in decreasing order. After removing the 25 highest values and the 25 lowest ones, the maximum and minimum values of the remainder determine the confidence bounds of

[3] www.ics.uci.edu/~mlearn/MLSummary.html

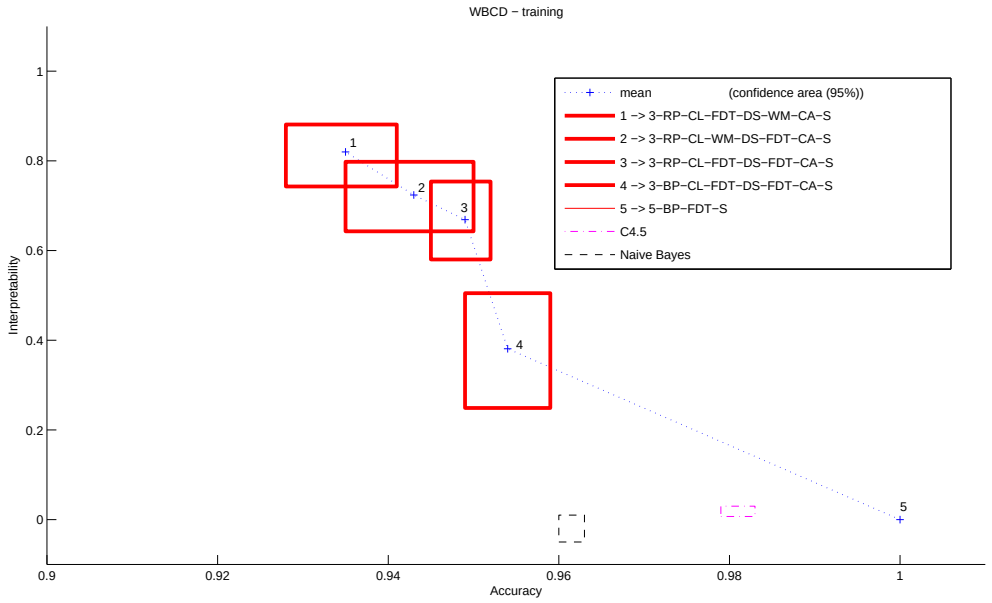


Figure 2: WBCD training Pareto front.

the experiment. Hence, it can be guaranteed that the 95% of times the experiment is repeated, the quality indices will be included into the confidence bounds.

3.2 Results and discussion

As result of the experimentation we get 36 different solutions and characterize each of them using bootstrapping. The best solutions, those non-dominated by other ones, form Pareto fronts like the ones plotted in figures 2 (training) and 3 (test). The interpretability index (vertical axis) is plotted versus the accuracy one (horizontal axis). Each Pareto solution is represented with a cross and framed by a rectangular box. The cross represents the average value while the rectangle determines the confidence area. A bold line emphasizes those solutions which appear in both training and test Pareto fronts at the same time. For comparison purpose, these graphics include two additional rectangle areas which represent the confidence areas for NB (dash line) and C4.5 (dot and dash line) methods.

The caption of the figures gives a name to each Pareto solution. That name describes the options involved in the generation of the KB. For instance, 3-RP-CL-WM-DS-FDT-CA-S, that produces simultaneously a good accuracy-interpretability trade-off regarding training and test, can be understood as follows. Firstly, regular partitions with three labels (3-RP) are built for each variable. Secondly, clustering techniques are applied in order to generate a small training set made up of cluster centroids which is used as training set by WM (CL-WM) in order to generate RB1. Then, a data selection process is used for generating a second training set which is used as training set by FDT (DS-FDT) in the generation of RB2. Then, the consistency of the entire KB (RB1 + RB2) is checked (CA). Finally, a simplification process (S) is applied to increase interpretability keeping accuracy and consistency.

To sum up, C4.5 gets high accuracy with respect to both training and test patterns. Of course, its interpretability is very poor. By the way, NB gets a robust solution with accuracy slightly higher than 96% over training and test, but its interpretability is extremely poor. Our methodology yields a set of highly interpretable and robust solutions with accuracy between 92% and 96% regarding both patterns. They produce accuracy

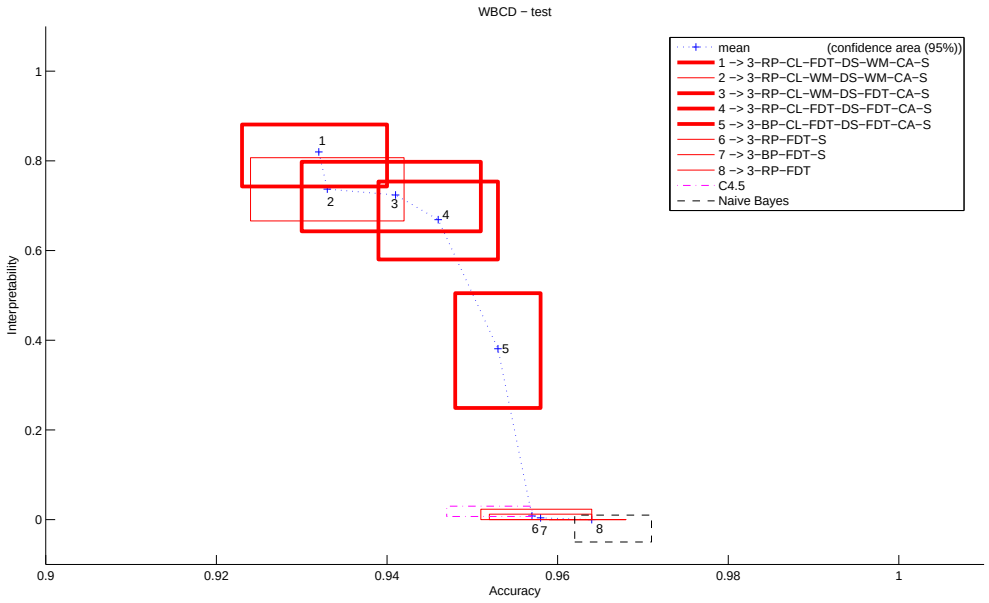


Figure 3: WBCD test Pareto front.

slightly lower than C4.5 with respect to training. However, C4.5 accuracy strongly decreases regarding test while our solutions keep almost the same accuracy in both cases.

4 Conclusions

This paper presents a new methodology (HILK) for building KBs with a good balance between accuracy and interpretability.

Two kinds of knowledge, expert knowledge and knowledge extracted from data are considered. However, regarding the evaluation of our methodology, it has been tested in a benchmark classification problem where the expert knowledge is quite reduced. As expert rules are not available, we have made the integration of two induced RBs generated using different induction techniques. The proposed approach leads not only to a good trade-off between accuracy and interpretability but also to a simultaneous improvement of both in some cases, where the final KB is more compact and transparent, but also more accurate than the initial one. Obtained results prove that the integration of two different rule bases can yield better accuracy-interpretability trade-off than the use of only one. In consequence, we can draw next conclusion: HILK can also be successfully applied even if there is not expert knowledge available relative to the problem under consideration.

Finally, notice that these experiments only show a small part of HILK power. They only regard induced knowledge, for an easy comparison with automatic methods, while our methodology is thought for solving problems where both expert knowledge and experimental data are available, and KB interpretability is of prime importance. It has been successfully applied in robotics for diagnosis of motion problems [5].

Acknowledgements

Although José M. Alonso is currently working with the European Centre for Soft Computing (ECSC), most of this work is based on his PhD dissertation [1] registered in the Technical University of Madrid (UPM)[4]. In addition, all results presented in this work were reached using the free software tool KBCT[5] which was designed and developed as an important part of the thesis.

[4]The full PDF document is freely available on: http://oa.upm.es/588/ (UPM Digital Archive).

[5]www.mat.upm.es/projects/advocate/kbct.htm

References

[1] J. M. Alonso. *Interpretable fuzzy systems modeling with cooperation between expert and induced knowledge (Modelado de sistemas borrosos interpretables con cooperación entre conocimiento experto e inducido).* PhD thesis, Technical University of Madrid (UPM), 2007.

[2] J. M. Alonso, O. Cordón, S. Guillaume, and L. Magdalena. Highly interpretable linguistic knowledge bases optimization: Genetic tuning versus solis-wetts. Looking for a good interpretability-accuracy trade-off. In *FUZZ-IEEE 2007*, pages 901–906.

[3] J. M. Alonso, S. Guillaume, and L. Magdalena. A hierarchical fuzzy system for assessing interpretability of linguistic knowledge bases in classification problems. In *IPMU 2006*, pages 348–355.

[4] J. M. Alonso, L. Magdalena, and S. Guillaume. Linguistic knowledge base simplification regarding accuracy and interpretability. *Mathware & Soft Computing*, XII(3):203–216, 2006.

[5] J. M. Alonso, L. Magdalena, S. Guillaume, M. A. Sotelo, L. M. Bergasa, M. Ocana, and R. Flores. Knowledge-based intelligent diagnosis of ground robot collision with non detectable obstacles. *Journal of Robotic & Intelligent Systems*, 48:539–566, 2007.

[6] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and regression trees.* Wadsworth International Group, Belmont CA, 1984.

[7] W. Browne, L. Yao, I. Postlethwaite, S. Lowes, and M. Mar. Knowledge-elicitation and data-mining: Fusing human and industrial plant information. *Engineering Applications of Artificial Intelligence*, 19(3):345–359, 2006.

[8] O. Cordón, F. Herrera, F. Hoffmann, and L. Magdalena. *Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases*, volume 19. Advances in Fuzzy Systems - Applications and Theory, World Scientific Publishing Co. Pte. Ltd., 2001.

[9] J. V. de Oliveira. Semantic constraints for membership function optimization. *IEEE Transactions on Systems, Man and Cybernetics. Part A, Systems and Humans*, 29(1):128–138, 1999.

[10] B. Efron and R. J. Tibshirani. *An Introduction to the Bootstrap.* London: Chapman & Hall, 1993.

[11] S. Guillaume. Designing fuzzy inference systems from data: An interpretability-oriented review. *IEEE Transactions on Fuzzy Systems*, 9(3):426–443, 2001.

[12] S. Guillaume and B. Charnomordic. Generating an interpretable family of fuzzy partitions. *IEEE Transactions on Fuzzy Systems*, 12 (3):324–335, 2004.

[13] S. Guillaume and L. Magdalena. Expert guided integration of induced knowledge into a fuzzy knowledge base. *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, 10(9):773–784, 2006.

[14] D. J. Hand and K. Yu. Idiot's Bayes - not so stupid after all? *International Statistical Review*, 69(3):385–399, 2001.

[15] J. A. Hartigan and M. Wong. A k-means clustering algorithm. *Applied Statistics*, 28:100–108, 1979.

[16] E. Hüllermeier. Fuzzy methods in machine learning and data mining: Status and prospects. *Fuzzy Sets and Systems*, 156:387–406, 2005.

[17] M. Kwiatkowska, N. T. Ayas, and F. Ryan. Evaluation of clinical prediction rules using a convergence of knowledge-driven and data-driven methods: A semio-fuzzy approach. *Data Mining VI: Data Mining, Text Mining And their Business Applications. Computational Mechanics. A. Zanasi (edt.)*, pages 411–420, 2005.

[18] E. H. Mamdani. Application of fuzzy logic to approximate reasoning using linguistic systems. *IEEE Transactions on Computers*, 26(12):1182–1191, 1977.

[19] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1:81–106, 1986.

[20] J. R. Quinlan. *C4.5: Programs for Machine Learning.* Morgan Kaufmann Publishers, San Mateo, CA, 1993.

[21] E. H. Ruspini. A new approach to clustering. *Information and Control*, 15(1):22–32, 1969.

[22] 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, 1992.