

Fuzziness as a recognition problem: using decision tree learning algorithms for inducing fuzzy membership functions

O. Nykänen

Department of Mathematics, Tampere University of Technology, Finland

Abstract

In this article we establish a new method for inducing fuzzy set membership degrees based on empirical training data. The approach is founded on the notion of Redundant Decision Trees (RDT), a generalisation of regular crisp Decision Trees (DT). RDTs suffice in capturing the attribute tests required for recognising crisp concepts, from which the related fuzzy concepts may be unambiguously derived. Potential applications of this method include categorisation and the semiautomatic construction and the statistical evaluation of fuzzy concepts. In addition, since the definition of the membership degrees is effectively based on a robust DT machine learning algorithm, the induced fuzzy membership functions generalise. Thus, with certain assumptions, they output sensible membership degrees of previously unseen objects. In addition to introducing and analysing the basic definitions and algorithms, we briefly evaluate their applicability with examples and present some remarks concerning the scope of the approach.

Keywords: fuzzy sets, decision trees, machine learning, empirical categorisation.

1 Introduction

Many of the practical problems of implementing intelligent systems are related to recognition and qualification problems. In short, the recognition problems make it difficult to evaluate the state of the world, while the qualification problems, partly because of the recognition problems, make it hard to define the circumstances under which a given action is guaranteed to work [12].

These problems become concrete when implementing fuzzy systems based on empirical data sets. By definition, fuzzy systems tolerate imprecision by operating with fuzzy concepts (sets). Roughly speaking, there exists two



complementary approaches for implementing fuzzy systems: *fuzzy expert systems*, and *fuzzy control systems*. Fuzzy expert systems operate via chaining rules, applying a set of rules whose thresholds get satisfied in evaluations. Fuzzy control systems operate in terms of fuzzy sets: they combine several fuzzy rules to output conclusions as transformed fuzzy sets. For brevity, we will next only consider fuzzy expert systems, i.e. treat fuzzy sets as logical truth functions [14].

Consider the following linguistic rule: IF $m.x$ is A AND $m.y$ is *very*(B) THEN *breaks*(m). In applications, evaluating the membership degree of A($m.x$) is effectively a recognition problem. The inference agent must first verify that the type of x (perhaps a complex object) is indeed comparable with A, and then evaluate the membership value of x in terms of A. Alternatively, this problem may be perceived as a data mining problem, related to empirical categorisation.

In the Classical case, all fuzzy concepts (A and B) are defined analytically. In practice, however, it may turn out that no analytical definition is available (or sensible), there exists only a limited amount of crisp examples, or the concept B seems to be poorly compatible with the concept A (but to what extent?).

In this article, we introduce a new method for empirically inducing fuzzy membership functions, based on Decision Tree (DT) learning algorithms. In short, the introduction of Machine Learning (ML) algorithms allows analysing and implementing fuzzy systems in terms of empirical recognition problems. The basic idea is inducing fuzzy membership functions from modified decision trees (forests) which are trained to recognise the crisp concepts demonstrating the two ends of the related bipolar fuzzy concepts. The definition of fuzzy membership degrees is ultimately based on the information revealed by the attribute tests.

The contributions of this article are threefold: First, establishing a new approach for inducing fuzzy membership functions based on examples, second, demonstrating and evaluating the definitions and the algorithms in applications, and third, (briefly) assessing the generalisation performance of the approach.

This work is founded on the research and the development of fuzzy sets, fuzzy logic, fuzzy logic systems, and the related Soft Computing and Neuro-Fuzzy methods (see, e.g., [14, 15, 7, 4, 3]). When compared to fuzzy decision trees in general [6], the main benefits of our approach include founding the definitions on redundant decision trees in the context of concept learning and considering the generalisation performance, thus establishing a well-defined and a transparent basis for implementing reactive fuzzy agents (see, e.g., [11, 10]).

The rest of this article is organised as follows: The basic foundations are outlined in Section 2. The method of inducing fuzzy membership functions is presented in Section 3. Section 4 evaluates the approach with concrete examples, and finally, Section 5 concludes the article with some remarks.

2 Inducing fuzzy sets upon examples

Fuzzy sets (concepts) generalise the notion of crisp sets (concepts) by replacing characteristic functions $\chi_A: X \rightarrow \{0,1\}$ with membership functions $\mu_A: X \rightarrow [0,1]$. The definition of a membership function may be established either analytically or based on empirical data. A seemingly natural inductive definition originates



from providing two crisp sets, D^+ and D^- called the positive and the negative examples (the training data), from which the concept is to be *induced*.

In general, the task is thus to derive an *algorithm* T which, when presented an instance $x \in A$ (defined over X), functionally induces the desired membership degree of some concept α , i.e. $T(x, D^+, D^-) = \mu_\alpha(x)$. (In the crisp case, this corresponds to $A = \{x \in X : T(x, D^+, D^-)\}$.) Note that not all concepts can be presented this way and the approach of using two sets D^+ and D^- , is simply pragmatic: in many applications, (crisp) positive and negative examples are the cheapest to produce. Ideally, the examples are assumed to establish the two ends of a bipolar concept.

Inducing concepts has traditionally been a task of Machine Learning (ML) and data mining. Few results are worth mentioning here. First, there is no such thing as bias-free learning (domain characteristics must be known *a priori*) and second, no learner demonstrates better-than-chance performance over arbitrary domain (the conservation law for generalisation performance) [5, 13].

The task of fuzzy concept learning can be fruitfully associated with fuzzy (logic) systems. A Fuzzy Logic System (FLS) provides a non-linear mapping between two systems in a rule-based form (see, e.g., [7, 4]). Figure 1 presents the typical components of a Fuzzy Logic System.

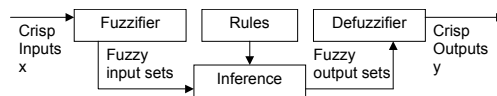


Figure 1: Fuzzy logic system $y=f(x)$ (modified from [7]).

In particular, fuzzy inference and control involves two steps which ground the fuzzy rules into applications: fuzzification and defuzzification. These transform the crisp signals into fuzzy sets, and vice versa. From this perspective, we may treat T as a *fuzzifier* or, more precisely, a *recognition algorithm*. This interpretation is natural in many applications. In its most restricted form, T transforms the crisp input x directly into a membership degree, $\mu_\alpha(x)$ for some concept α . In the general case, T establishes a density function for α .

Typical reasons for working with induced fuzzy sets in applications is seeking "data-to-model" methods, the fact that only empirical data is available, or that the concepts are supposed to adapt based on future observations. The latter motivation is particularly relevant in the context of adaptive systems.

3 Algorithms for learning fuzzy concepts

Next we establish and briefly analyse the basic definitions of inducing fuzzy concepts based on redundant decision trees.

3.1 Membership degrees based on decision trees

Consider the following modified decision tree learning algorithm recognising crisp concepts:



Algorithm 1 (Redundant Decision Tree Learner (RDTL)) *We call a Redundant Decision Tree Learner any entropy-based decision tree learning algorithm DTL (that induces a decision tree from the training data of positive and negative examples, outputting 1 for Yes and 0 for No) with the following modifications:*

1. *Assume an inductive DT learning algorithm DTL that uses information gain (entropy) as a measure for choosing attribute tests to the decision tree.*
2. *Modify DTL so that instead of choosing a single attribute a with the best information gain, T chooses a set of attributes $\{a_i\}$, including all of the attributes with the best information gain.*
3. *Modify DTL so that instead of (recursively) adding a single subtree to the decision tree by the attribute a , it adds one subtree for each attribute a_i .*

A popular example of an entropy-based DTL is the ID3 and its modifications [5, 12, 6]. The following experiments of this article utilise our own implementation of a RDT learner which is based on the modified ID3 algorithm called DECISION-TREE-LEARNING, found in [12].

It is easy to see that the RDTL and the DTL are identical in terms of input-output behaviour, i.e. they induce exactly the same concepts:

Proposition 1 (Unique Redundancy Property) *Let T_1 be a decision tree induced by an entropy-based decision tree and T_2 a related redundant decision tree. Trivially, $\text{Dom}(T_1) = \text{Dom}(T_2)$; denote this set by X . Then*

1. *For each $x \in X$, $T_1(x) = T_2(x)$.*
2. *As a tree, T_2 is uniquely defined by T_1 : for each T_1 there is exactly one T_2 .*

Proof. Follows from Algorithm 1 (the second part is true effectively because there are no choices left to the algorithm).

Note that in practice, inducing the smallest, best-performing decision tree(s) is an intractable problem (approximated by pruning etc.). This means that in applications, the decision trees may depend on the particular implementation.

The rationale behind the algorithm 1 becomes clear in the context of inducing fuzzy membership degrees.

Definition 1 (Fuzzy Membership Degree by RDTL) *Let T be a redundant decision tree induced by the algorithm 1 and x an input attribute vector, $x \in X = A_1 \times A_2 \times \dots \times A_m$. Let $\{\Gamma_k\} = \{\Gamma_k \subset T : \Gamma_k(x) = 0\}$ be a finite set that denotes all the branches of T that output a negative classification. Let $p > 0$ be the length of the longest branch of $\{\Gamma_k\}$. If $T(x) = 0$, let s be the length of the shortest branch $\Gamma \in \{\Gamma_k\}$ so that $\Gamma(x) = 0$.*

Define the membership function induced by T , $\mu_T : X \rightarrow L$, $L \subset [0, 1]$, as follows: If $T(x) = 1$, define $\mu_T(x) = 1$, otherwise define $\mu_T(x) = (s-1)/p$.

In the following examples, if the decision tree is unable to output a prediction (due encountering a previously unseen attribute value), we simply set $\mu_T(x)=0.5$. (This is not the best strategy, however.)



The added redundancy of Algorithm 1 ensures that Definition 1 is well-defined, i.e. the membership degree is unambiguous (proposition 1). It is easy to see that the definition genuinely generalises the crisp concept in terms of Classical reasoning:

Proposition 2 (Subsumption Property) *Let $\mu_T(x)$ be a membership degree induced by a redundant decision tree T . Then $\mu_T(x) = 1$ if and only if $T(x) = 1$. Otherwise $0 \leq \mu_T(x) < 1$.*

Proof. Follows straightforwardly from Definition 1.

The asymmetric definition of the fuzzy membership degrees is in fact guided by the subsumption property in applications. The property is useful both in inference behaviour in the limit (with crisp values) and in composing and evaluating training data (the ends of the bipolar target concept can be intuitively defined). It is easy to see that the subsumption property nevertheless allows, e.g., modifying the formula of $\mu_T = f(s, p)$ by composition or using the information gains etc.

Intuitively, since the attribute tests are organised according to the best information gain, Definition 1 resembles the game of Twenty Questions: The more attribute tests are required to differentiate an object from the concept defined by a decision tree, the larger the fuzzy membership degree (when < 1). From this perspective, the redundancy of decision trees implements a sort of rotation symmetry (invariance), demonstrating the fact that some aspects of the objects (attributes) are equal when establishing membership degrees.

3.2 Membership degrees based on decision trees

Of course, there are other ways to define empirical, attribute-based membership functions. A well-known alternative is to measure the distance between objects and sets in terms of (mis)matching attributes:

Definition 2 (Direct Distance) *Let $D = \{e_1, e_2, \dots, e_n\} \subset X$ be the set of examples and define $\Delta(x, D) = \min_{k=1,2,\dots,n} \{ \sum_{j=1}^m \delta(x_j, e_k^j) \}$ where m is the number of the predictive attributes, i.e. the dimension of x .*

We say that $\Delta : X \times D$ is the direct distance between $x \in X$ and (the concept denoted by) the set D .

In practice, the direct distance is the number of the unequal attribute values between an object x and the most similar instance in D . As suspected, the distance may be used for defining a fuzzy membership degrees.

Definition 3 (Fuzzy Membership degree by a Direct Distance) *Let $D^+ = \{e_1, e_2, \dots, e_n\} \subset X$ be a set of positive examples, $m > 0$ the number of predictive attributes in D^+ , and $\Delta_1 : X \times D^+$ a direct distance.*

We say that $\mu_d(x, D^+) = 1 - \Delta_1(x, D^+) / m$ is the fuzzy membership function in X , induced from D^+ by the direct distance.



Note that Definition 3 neglects the negative examples. For brevity, we will not consider further definitions here. Note that from the perspective of machine learning, Definition 3 establishes a sort of similarity-based (default) learner.

3.3 Relationship with fuzzy clustering

So far we have considered inducing fuzzy concepts (sets) upon examples. This process may also be interpreted in terms of (fuzzy) clustering.

Assume a data set X with n_c clusters. A *fuzzy clustering algorithm* associates each $x_i \in X$ with a vector $[a_1, a_2, \dots, a_{n_c}]^T_i$ of membership degrees for the optimal n_c clusters. The fuzzy clusters are thus induced from the data, using n_c as a parameter value. Technically, fuzzy clustering may take place in a metric space where a special kind of norm function establishes a measure of distance (or difference). The basic fuzzy clustering algorithm is the probabilistic c-means algorithm which generalises the crisp k-means algorithm [2].

In this context, using decision trees for inducing fuzzy membership degrees may be considered a special kind of conceptual fuzzy clustering method. An alternative is to consider it as special kind hierarchical clustering method.

Regardless of the point of view, the main difference is that the RDTL algorithm is a *supervised* learning algorithm. It establishes a single cluster via the training data $D = (x_i^T, c_i)_{i=1,2,\dots,n}$ which is *a priori* classified by crisp binary labels c_i . Given an object x_i , the distance between x_i and the target concept (which can be loosely described as a cluster) is computed indirectly, from the structure of the induced decision tree. Note that the approach does not restrict the number of cluster definitions in X , but each cluster C_k (concept) must be characterised by its own training data (the two sets D_k^- and D_k^+).

4 Assessing the empirical performance

Next we will establish a simple performance measure based on linear correlation and consider case studies demonstrating both the use of the induction algorithms and the process of evaluating their (generalisation) performance.

4.1 A performance measure based on correlation

An intuitive performance measure originates from evaluating the induced membership degrees in a well-understood domain.

Definition 3 (Reference Data Matrix) Let $D = (e_i^T c_i)_{i=1,2,\dots,n}$ be the matrix of the available learning data where $e_i^T \in A_1 \times A_2 \times \dots \times A_m$ and $c_i \in \{0, 1\}$. Assume an ideal membership function $\mu_\beta : E \rightarrow L, L \subset [0, 1]$ (of some target concept β) and denote $B = (e_i^T \beta_i)_{i=1,2,\dots,n}$ where $\beta_i = \mu_\beta(e_i^T)$.

We call a reference data matrix a data matrix $R = (e_i^T c'_i), c'_i \in [0, 1]$ that faithfully approximates B .

In other words, when testing the algorithms, the reference data matrix allows assessing the quality of the outputted membership degrees. Of course, in



production systems, reference data matrices are not usually available (if they were, they could be transformed into membership degrees without RDTLs).

For simplicity, we shall use the Pearson's correlation to investigate the relationship between the induced membership degrees and their reference values. Note that we thus assume that both C and C' include data on interval scales. For assessing the generalisation performance, we will divide the training data and the reference data matrices into (DTL) training and validation data subsets

4.2 Tests with UCI data sets: basic properties

We tested the definitions with several data sets with known goal attributes (used as reference values c_i). During the training phase, the goal attributes (denoting several concept classes) were transformed into binary identifiers, thus establishing the required two sets of the positive and the negative examples.

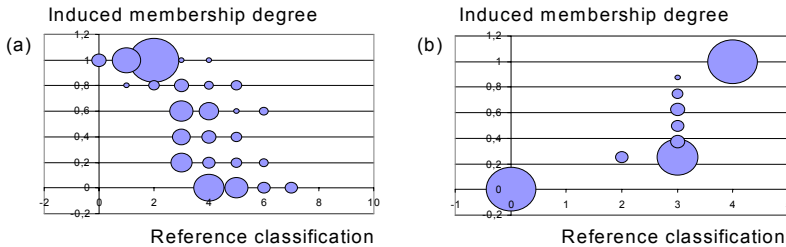


Figure 2: Bubble plots (c_i, μ_i) with membership degrees μ_i induced from the UCI ML repository data sets (a) *cpu-performance* and (b) *nursery*.

Figure 2 depicts two tests with UCI machine learning repository data sets [1]. The first plot is based on the UCI *cpu-performance* archive. It uses 209 instances with 5 predictive attributes and a goal attribute (8 crisp classes), all naively reduced and discretised by performing $\text{floor}(\log_2(v))$ to each value v . A crisp concept of "a modest performance cpu" (an informal add-on label) was compiled according to the Definition 1 ($\text{goal} < 3$). The resulting correlation coefficient of the induced membership degrees and the reference goal attributes is -0.79.

The second plot is based on the UCI *nursery* archive. It uses 12960 instances with 7 predictive attributes and a goal attribute (4 crisp classes). No discretisation was required since attributes encoded classes. A crisp concept of "spec_prior" was compiled according to the Definition 1. The resulting correlation coefficient of the induced membership degrees and the reference goal attributes is 0.87.

The induced degrees correlate with the reference values as expected. The granularity of the membership degrees (Y axis) is roughly determined by the number of the predictive attributes and the concept structure (and is rather low).

4.3 Fuzzifying a crisp attribute: generalisation performance

Let us then consider a more detailed description of a case study that involves the generalisation aspect. Assume we know a set of descriptive attributes (e_i^T) from a

group of users, related to a particular application domain. Further, assume that the users provide additional binary attributes (c_i), e.g., by means of answering questions or making choices which are suitable for classifying the users.

Our task is fuzzifying the binary attributes into fuzzy membership degrees (e.g. in order to recognise three fuzzy user classes). In addition, assuming that the additional attributes produce additional costs, we would like to manage with as few attributes (c_i) as possible. The underlying assumption is that the descriptive data is rich enough and statistically adequate. (Roughly speaking, as a parameter set, (e^T_i) suffices in defining (c_i) in the above sense.)

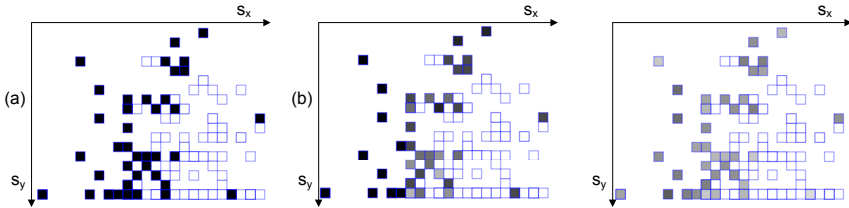


Figure 3: (PCA-based) crisp and fuzzy maps based on (a) the crisp classification, (b) the RDTL algorithm, and (c) the direct distance when using all of the available training data.

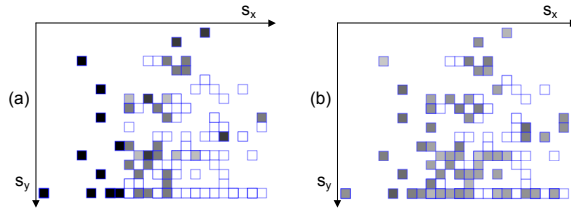


Figure 4: Generalisation performance of the (a) the RDTL algorithm and (b) the direct distance, when using only 2/3 of the available training data.

Next we will consider solving this task by using the empirical induction methods discussed so far. We will see that this approach provides two kinds of advantages. First, the induced membership degrees establish the fuzzy concept (subject to logical modifiers) without explicit modelling. Second, if the data (e^T_i) is descriptive enough, the definition generalises to unseen examples. In other words, membership degrees can also be estimated for users whose descriptive attributes (e^T_i) are known, but whose additional attributes (c_i) are not known.

As a part of an evaluation project of Web-based learning systems, a student survey was compiled at the Tampere University of Technology, Hypermedia laboratory. After attending to various university courses including Web-based modules, 150 students answered to a set of (multiple choice) questions.

In the experiment, the descriptive attributes (e^T_i) included 14 values from 11 questionnaire questions and three summation variables, resulting from the pre-processing of 17 (semantically overlapping) background questions with Principle

Component Analysis (PCA) with manual selection. In addition, a binary question (c_i) was compiled ("Would you participate in a Web-based course in the Future (Yes/No)?"). Figure 3a demonstrates the distribution of the binary questions portrayed as a student map. The co-ordinate values ($s_x \sim$ "general attitude towards Web-based learning", $s_y \sim$ "IT skills") result from the PCA analysis.

Figure 3a clearly shows that there exists a strong correlation between the attributes s_x and s_y , and the binary question c . Note that this actually suggests that in the future, given (e_i^T), there is little sense in explicitly asking (c_i).

Figures 3b and 3c show the fuzzification results after applying the RDTL fuzzification algorithm (μ_T by Definition 1) and the membership degree based on the direct distance (μ_d by Definition 3). Both methods seem to recognise the fuzzy concept C ("a student interested in Web-based courses in the future").

Figures 4a and 4b show the fuzzification results in terms of the generalisation performance, when using only 2/3 of the available training data (a random experiment). When comparing figures 4a and 4b it seems that μ_T generalises better than μ_d (which diverges). This becomes evident when analysing the performance in terms of the reference data matrix (for simplicity, with values s_x).

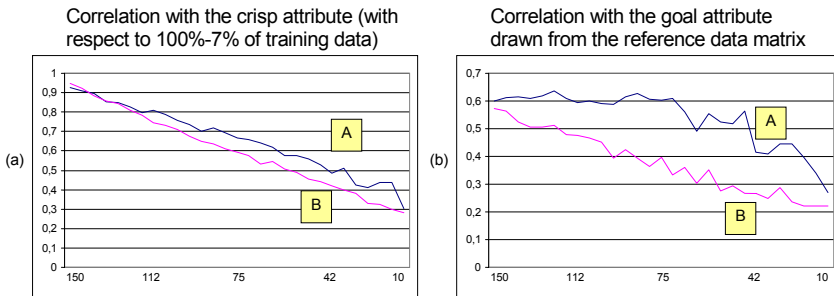


Figure 5: Correlation graphs demonstrating the generalisation performance with the validation data set, using the RDTL algorithm (A) and the direct distance (B).

The four *correlation curves* in Figure 5 show the generalisation performance in terms of the training data and the *a priori* known reference data. The X axis shows the training set size, starting from the whole set of the available training data ($N = 150$). The values of the Y axis consist of the arithmetic means of the correlation coefficients after 10 trials with the validation data set, using random training data sets consisting of X instances.

The RDTL algorithm seems to outperform the direct distance with the reference data (curve A in Figure 5b). The performance stays maximal until using only half of the available training data. As suspected, the reason for this is the generalisation performance of the underlying DT learner (which in this case seems to capture the crisp concept with some 75 training instances).

These results seem to indicate that the RDTL algorithm and the related definitions for inducing fuzzy membership degrees outperform the method of

inducing membership degrees based on the direct attribute distance. In particular, the RDTL algorithm seems to be able to learn and to recognise the fuzzy concept C , at least within the given application. As a consequence, it is thus possible to reasonably estimate the membership degree of C for a student k without explicitly asking c_k in the future, assuming that the archived training data is adequate and the domain characteristics remain unaltered.

5 Discussion and conclusions

Applying decision tree learning algorithms is implicitly associated with the assumption that the training data contains regularity that can be captured by the training data, and presented by decision trees (a finite set of Boolean functions). The generalisation performance of decision trees is guaranteed only statistically on a restricted domain. In brief, any limitation of decision trees is potentially a limitation when inducing fuzzy membership functions. The additional memory and computing requirements (redundancy, parallel searches), however, are not significant. In principle, decision trees are also transparent (unlike, e.g., neural networks) which allows explaining the outputted classifications.

Due to the lack of space, we have not analysed the generalisation performance in detail. Roughly speaking, the generalisation performance of the membership degrees follows the generalisation performance of the underlying decision trees. With certain assumptions (a restricted, "well-behaving" domain), it can be shown that the generalisation performance of the induced membership function (μ_T), due to the errors of the crisp predictions (T), is better than change.

In this article we have demonstrated a new method for inducing fuzzy membership functions upon examples, using a modified decision tree learning algorithm. This approach allows a model-free definition of fuzzy concepts which may extend to previously unseen data. As a consequence, the induced fuzzy sets can also be characterised statistically, in terms of the properties of the related training data sets and the performance of the decision trees (learning).

We are currently investigating a strategy for incorporating the presented induction algorithms as new module in a rule system called CWM [9]. Within this framework, we anticipate to implement adaptive and reactive decision-support systems that effectively combine the sub-symbolic (induction of concepts) and the symbolic (e.g. writing of the rules) information processing paradigms. The basic motivation for this work is enabling users to define their private fuzzy concepts by example, supported by a common set of fuzzy concepts, induced (mined) from the related context and the known data sources.

References

- [1] Blake, C.L. & Merz, C.J., UCI Repository of machine learning databases. Irvine, CA: University of California, Department of Information and Computer Science, 1998. Online. <http://www.ics.uci.edu/~mllearn/MLRepository.html>



- [2] Höppner, F., Klawonn, F., Kruse R. & Runkler, T., *Fuzzy Cluster Analysis*, John Wiley, 1999.
- [3] Jang, J-S.R., Sun, C-T. & Mizutani, E., *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*, Prentice Hall, 1996.
- [4] Kosko, B., Fuzziness vs. Probability. *Int. Journal of General Systems*, Vol. 17, 1990.
- [5] Mitchell, T., *Machine Learning*. McGraw-Hill, 1997.
- [6] Mitra, S., Konwar, K.M. & Sankar, K.P., Fuzzy Decision Tree, Linguistic Rules and Fuzzy Knowledge-Based Network: Generation and Evaluation. *IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews*, Vol. 32, No. 4, November 2002.
- [7] Mendel, J., *Uncertain Rule-Based Fuzzy Logic Systems*. Prentice Hall, 2001.
- [8] Murthy, S.K., *On Growing Better Decision Trees from Data*. Doctoral dissertation, University of Maryland, 1997.
- [9] Nykänen, O., Home page of the Decision Tree module for the CWM. Tampere University of Technology, Department of Mathematics, Hypermedia Laboratory, 2003. Online. http://matriisi.ee.tut.fi/~onykane/2004/cwm_dt/.
- [10] Nykänen, O., (to appear), An Approach for Integrating Statistical Decision-Support Data with Fuzzy Action Rules. *Proc. of the ICALT 2004*, Joensuu, Finland, 2004.
- [11] Nykänen, O., (to appear), On Implementing Fuzzy Semantic Web Agents with the CWM Rule System. *Proc. of the IADIS International Conference WWW/Internet 2004*, Madrid, Spain, 2004.
- [12] Russel, S. & Norvig, P., *Artificial Intelligence: A Modern Approach*. Prentice-Hall, USA, 1995.
- [13] Schaffer, C.A., Conservation Law for Generalization Performance. *Proc. of the Int. Conf. on ML*, San Mateo, Ca., Morgan Kaufmann, 1994.
- [14] Turunen, E., *Mathematics Behind Fuzzy Logic*. Advances in Soft Computing. Physica-Verlag Heidelberg, 1999.
- [15] Yager, R.R. & Zadeh, L.A., *Fuzzy Sets, Neural Networks, and Soft Computing*, Van Nostrand Reinhold, New York, 1994.

