ELSEVIER

Cognitive Systems Research 3 (2002) 125–144

Cognitive Systems RESEARCH

www.elsevier.com/locate/cogsys

A neuro-fuzzy network to generate human-understandable knowledge from data

Action editor: Paolo Frasconi

Giovanna Castellano, Anna Maria Fanelli*, Corrado Mencar

Dipartimento di Informatica, Università degli Studi di Bari, V. Orabona, 4, 70126 Bari, Italy

Received 15 December 2000; accepted 24 August 2001

Abstract

Neuro-fuzzy networks have been successfully applied to extract knowledge from data in the form of fuzzy rules. However, one drawback with the neuro-fuzzy approach is that the fuzzy rules induced by the learning process are not necessarily understandable. The lack of readability is essentially due to the high dimensionality of the parameter space that leads to excessive flexibility in the modification of parameters during learning. In this paper, to obtain readable knowledge from data, we propose a new neuro-fuzzy model and its learning algorithm that works in a parameter space with reduced dimensionality. The dimensionality of the new parameter space is necessary and sufficient to generate human-understandable fuzzy rules, in the sense formally defined by a set of properties. The learning procedure is based on a gradient descent technique and the proposed model is general enough to be applied to other neuro-fuzzy architectures. Simulation studies on a benchmark and a real-life problem are carried out to embody the idea of the paper.

Keywords: Neuro-fuzzy network; Fuzzy knowledge base; Fuzzy rule extraction; Fuzzy rule interpretability

1. Introduction

Neuro-Fuzzy models have been developed with the aim of integrating the learning capability of neural networks with the representational power of fuzzy inference systems, thus producing learning machines capable of acquiring knowledge from data and representing it in form of fuzzy rules (Jang & Sun, 1995; Jang, 1993; Nauck, Klawonn & Kruse, 1997; Brown & Harris, 1994; Zurada & Lozowski, 1996). However, the interpretability of fuzzy knowledge acquired by a neuro-fuzzy system may be heavily compromised by the learning phase of the network, if no special attention is paid during data-based rule generation and adaptation.

The requirement of interpretability is particularly felt when neuro-fuzzy systems are applied to real-world problems (Nauck, 1995; Halgamuge & Glesner, 1994) such as decision support in medicine, finance, commerce and other applications. In such application areas the knowledge about the behavior of the decision system should be transparent and physically sound so as to meet the cognitive capacity of human beings and to mimic the way they perform

*Corresponding author. Tel.: +39-080-544-3285; fax: +39-080-544-3196.

E-mail address: fanelli@di.uniba.it (A.M. Fanelli).

1389-0417/02/$ – see front matter
PII: S1389-0417(01)00055-9

high-decision processes. As a consequence, the lack of interpretability often makes neuro-fuzzy models less useful than classical fuzzy inference systems (Pedrycz & Gomide, 1998; Cios, Pedrycz & Swiniarski, 1998; Ross, 1997), where the knowledge base is manually built and learning techniques are not adopted.

Since interpretability itself is a fuzzy and subjective concept, it is hard to find an explicit and exhaustive list of properties that, when violated, make the fuzzy rule base to loose its readability. Some important aspects pertaining the interpretability of fuzzy rules have been discussed in (Lofti, Handerson & Toi, 1996; Jin, Von Seelen & Sendhoff, 1998; Jin, Von Seelen & Sendhoff, 2000), while a comprehensive set of properties that fuzzy sets should verify to preserve interpretability is postulated in (Pedrycz & Gomide, 1998; de Oliveira, 1999). However, to date, there is no well-established definition for interpretability of a fuzzy rule base. Furthermore, even with a clear definition of readability, the preservation of readability during rule extraction and adaptation requires either reducing the degrees of freedom of the neuro-fuzzy model or using a constrained learning method which penalizes all solutions which are not readable (Bersini & Bontempi, 1997). Hence, the development of learning methods to induce understandable fuzzy rules from data is an important research issue.

Several approaches have been proposed to obtain interpretable knowledge by neuro-fuzzy learning (Jin et al., 1998; Nauck, Nauck & Kruse, 1996; Nauck & Kruse, 1997; Lozowski & Zurada, 2000; Marin-Blazquez, Shen & Gomez-Skarmeta, 2000; Setnes, Babuska & Verbuggen, 1998a; Chow, Altug & Trussell, 1999a; Chow, Altug & Trussell, 1999b).

In (de Oliveira, 1999), the learning process is constrained to respect some properties that make fuzzy rules human-understandable. Such constraint is realized by means of regularization theory: the cost function to be minimized during training is composed of the Mean Squared Error (MSE), as usual, in addition to a penalty function, which is the mathematical counterpart of the properties that fuzzy rules have to satisfy. In (Jin et al., 1998; Jin, Von Seelen & Sendhoff, 1999) the authors proposed completeness and consistency indices for a fuzzy rule base that are treated as a means of regularization by incorporating them into the cost function of an evolution algorithm to generate an interpretable fuzzy rule base. The rule base is converted into a RBF network and refined through a regularization algorithm called Adaptive Weight Sharing that guarantees interpretability and compactness of the final rules. On the overall, this approach turns out to be flexible and gives promising results in handling high-dimensional problems. However, the approaches based on regularization have the drawback of introducing more hyper-parameters – the regularizing parameters – for which no efficient method exists to determine the optimal values, except by trial-and-error. Some mathematical techniques have been proposed, as in (Bengio, 2000; Craven & Wabba, 1979), but they are computationally intensive.

In (Chow et al., 1999a,b), the authors propose a set of transformations to project the parameter space of a neuro-fuzzy network into a subspace where a number of properties (more stringent than those adopted in (de Oliveira, 1999)) are satisfied. This projection is applied at each iteration of the learning algorithm, resulting in a high computational cost.

In other works, interpretability of fuzzy systems from the view point of membership functions is discussed. In (Setnes et al., 1998a; Setnes, Babuska, Kaymak & van Nauta Lemke, 1998b), similar fuzzy membership functions are merged so that the resulting fuzzy partitions are interpretable, while in (Lofti et al., 1996) a constraint is imposed to the location of membership functions during learning. In (Nauck, Nauck & Kruse, 1996; Nauck and Kruse, 1997), the authors propose NEFCLASS, an approach that creates fuzzy systems from data by applying an heuristic data-driven learning algorithm that constraints the modifications of fuzzy set parameters to take the semantical properties of the underlying fuzzy system into account. However, a good interpretation of the learning result cannot always be guaranteed, especially for high-dimensional problems. Hence, in (Nauck and Kruse, 1999) the NEFCLASS algorithm is added with interactive strategies for pruning rules and variables so as to improve readability. This approach provides good results, but it results in a long interactive process that cannot extract automatically rules from data but requires the ability of the user to supervise and interpret the learning procedure in all its stages.

This paper proposes an approach to extract automatically fuzzy rules by learning from data, with the

main objective to obtain human-readable fuzzy knowledge base. A new neuro-fuzzy model and its learning algorithm is developed that works in a parameter space with reduced dimensionality with respect to the space of all the free parameters of the model. The dimensionality of the new parameter space is necessary and sufficient to generate human-understandable fuzzy rules, in the sense formally defined by a set of properties. Once the new parameter space is defined, the learning algorithm performs simple gradient descent with no additional constraint in the parameter modifications. The proposed model is general enough to implement different types of fuzzy rules, since its structure depends only on the form of the rule antecedents and does not depend on the form of the rule consequents. In this work, the proposed model has been defined to implement a zero-order Takagi–Sugeno (TS) fuzzy model (Sugeno & Kang, 1988; Takagi & Sugeno, 1985). However, our model can be easily adapted to embody other neuro-fuzzy architectures, such as ANFIS (Jang, 1993), Lin and Lee network (Lin & Lee, 1991), Neuro-Fuzzy Classifiers (Castellano & Fanelli, 2000a; Castellano, Fanelli & Mencar, 2000), and Multistage Fuzzy Neural Networks (Chung & Duan, 2000; Wang, 1999).

The paper is organized as follows. Section 2 gives a set of formal properties of a Fuzzy Knowledge Base (FKB) that must be satisfied to ensure readability. Section 3 focuses on the dimensionality of the parameter space of a readable FKB. Section 4 describes the proposed neuro-fuzzy architecture and its learning algorithm for the extraction of a FKB. Section 5 reports some experimental results, which support the theoretical framework, and Section 6 ends the paper with some conclusive remarks.

2. Interpretable fuzzy knowledge base

In this section, we first describe the Fuzzy Knowledge Base (FKB) and the input space fuzzy partition adopted. Then, we formalize the properties that must be satisfied in order to assure interpretability.

2.1. Fuzzy knowledge base

The rule base schema adopted in this paper is the following:

$$\text{RULE } r\text{: IF } x_1 \text{ IS } A^1_{g(r,1)} \text{ AND } x_2 \text{ IS } A^2_{g(r,2)} \text{ AND} \cdots \text{AND } x_n \text{ IS } A^n_{g(r,n)},$$
$$r = 1, \ldots, R,$$
$$\text{THEN } [\mathit{CONSEQUENCE}] \tag{1}$$

where n is the number of inputs and R is the total number of rules. The symbols $A^i_{g(r,i)}$ denote input fuzzy sets with membership function $\mu_{A^i_{g(r,i)}}$. The function g:$\{1, 2, \ldots, R\} \times \{1, 2, \ldots, n\} \to \mathbb{N}$ is used to share the same fuzzy sets in different rules. For a given rule r and an input i, the index $g(r, i)$ represents the fuzzy set of the i-th input variable used in the r-th rule. It is not necessary to give an analytical definition of the function g, since it can be easily implemented by a R-by-n matrix, automatically generated by a combinatorial algorithm. Fig. 1 illustrates a simple example of such function.

In (1) the form of the rule consequence is left undefined to include all the varieties of fuzzy rules having the antecedents in the form described in (1). Here, we consider a zero-order Takagi–Sugeno fuzzy model, which has the following rule base schema:

Fuzzy sets for variable x_1:
$\{Light, Heavy\} = \{ A^1_1, A^1_2 \}$
Fuzzy sets for variable x_2:
$\{Low, Medium, High\} \equiv \{ A^2_1, A^2_2, A^2_3 \}$

Example of rule base:
RULE 1: IF x_1 IS Light AND x_2 IS Medium THEN …
RULE 2: IF x_1 IS Heavy AND x_2 IS Low THEN …
RULE 3: IF x_1 IS Light AND x_2 IS High THEN …
RULE 4: IF x_1 IS Light AND x_2 IS Low THEN …

Extension of function g:

i \ r	1	2	3	4
1	1	2	1	1
2	2	1	3	1

Fig. 1. An example of function g.

RULE r: IF x_1 IS $A^1_{g(r,1)}$ AND $\cdots$ AND x_n IS $A^n_{g(r,n)}$

THEN $y_1 = k_{r1}, y_2 = k_{r2}, \ldots, y_m = k_{rm}$ (2)

where k_{rj} are fuzzy singleton defined on the m output variables y_j. Using singleton fuzzification, the product operator as t-norm for rule inference and center average defuzzification, the inferred j-th crisp output value for any input vector $\boldsymbol{x}$ is calculated as

$$y_j := \frac{\sum_{r=1}^{R} \mu_r(\boldsymbol{x}) k_{rj}}{\sum_{r=1}^{R} \mu_r(\boldsymbol{x})} \tag{3}$$

where $\boldsymbol{x} := (x_1, x_2, \ldots, x_n)$ and

$$\mu_r(\boldsymbol{x}) := \prod_{i=1}^{n} \mu_{A^i_{g(r,i)}}(x_i) \tag{4}$$

is the truth value of the r-th fuzzy rule.

2.2. Fuzzy partition of the input space

The partition of the input space plays a key role in the definition of a human-understandable FKB. The concept of *Frame Of Cognition* (FOC), defined in (Pedrycz & Gomide, 1998), is useful to formalize the properties of a fuzzy partition of an input domain. A FOC is a family of fuzzy sets that model a particular aspect of the world, also called *Universe Of Discourse* (UOD). Such collection of fuzzy sets must respect two main properties in order to be called FOC:

- *Coverage*: For each element x of the UOD there exists at least one fuzzy set A of the FOC for which $\mu_A(x) > 0$. Usually, a more stringent condition, called, *ε-coverage* is requested: $\mu_A(x) \geq \varepsilon$, for a fixed ε, (e.g. $\frac{1}{2}$). This condition assures that each element of the UOD is sufficiently represented.
- *Semantic soundness*: The fuzzy sets defined in the FOC must be linguistically interpretable. Such condition translates in a set of properties formally defined in the next section. According to such properties, each fuzzy set of the FOC must be normal and unimodal, all the fuzzy sets should be sufficiently disjointed and in a number 'psychologically' justified (empirically, 7 ± 2).

Unfortunately, these two properties are satisfied by only few techniques that perform a fuzzy partition of the input space. The simplest technique that produces a fuzzy partition of the input space considered as a FOC is the so-called *grid partition*. With this method, the domains of the input variables are partitioned into a specified number of fuzzy sets. The rule base is then established to cover the input space by using all possible combinations of input fuzzy sets as multivariate fuzzy sets describing the rule antecedents. This formulation of multivariate fuzzy sets results in a lattice partition of the input space, with the advantage that very interpretable fuzzy sets can be generated. A clear drawback of this approach is that the number of rules grows exponentially with the number of inputs. Conversely, other techniques, such as fuzzy clustering-based methods (Bezdek, 1981) can produce flexible fuzzy partitioning with a number of rules that grows linearly with the number of inputs, but they ignore the lattice partition of the input space, resulting in a FKB that cannot be easily interpreted.

In this work, we chose the grid partition technique since it is simple and offers the most comprehensible FOC that can be derived from an input space. However, other fuzzy partition methods that provide complete coverage of the input space may be used as well (Brown & Harris, 1994; Abe & Lan, 1995). We represent fuzzy sets by Gaussian membership functions, i.e. each fuzzy set A^i_h is characterized by the membership function

$$\mu_{A^i_h}(x_i) := \exp\left(-\frac{(x_i - \omega_{hi})^2}{2\sigma^2_{hi}} \right) \tag{5}$$

where ω_{hi} and σ_{hi} are the center and the width of the Gaussian function, respectively.

To guarantee the two properties above-mentioned, the centers and the widths of membership functions must be properly determined. More formally, let $\mathbf{X}$ be the input space. We assume that $\mathbf{X}$ is a n-dimensional Cartesian product among n intervals, that is,

$$\mathbf{X} := \bigtimes_{i=1}^{n} X_i, \quad X_i := [m_i, M_i] \subset \Re, \tag{6}$$

where X_i the ith axis of the input space $\mathbf{X}$, $m_i := \inf X_i$ and $M_i := \sup X_i$. For each axis X_i, a set Φ_i of K_i fuzzy sets is defined as follows:

$$\Phi_i := \{A_1^i, A_2^i, \ldots, A_{K_i}^i\} \quad (7)$$

The centers ω_{hi} are imposed to be equally spaced in the interval $[m_i, M_i]$, i.e. they are calculated as follows:

$$\omega_{hi} := (h-1)\frac{M_i - m_i}{K_i - 1} + m_i,$$
$$h = 1, 2, \ldots, K_i, i = 1, 2, \ldots, n. \quad (8)$$

In order to guarantee the, ε-coverage, for a given, $\varepsilon \in (0,1)$, the widths σ_{hi} are calculated as follows:

$$\sigma_{hi} := \frac{M_i - m_i}{2(K_i - 1)\sqrt{-2\ln\varepsilon}},$$
$$h = 1, 2, \ldots, K_i, i = 1, 2, \ldots, n. \quad (9)$$

Fig. 2 illustrates an example of fuzzy partition of an axis of the input space into three fuzzy sets. Here a 0.33-coverage is guaranteed.

2.3. Formal properties

In this subsection, the properties that must be satisfied to make a FKB human-understandable are mathematically formalized. Such properties are taken from (Pedrycz & Gomide, 1998; de Oliveira, 1999; Chow et al., 1999a).

For each set Φ_i the following properties must hold:

1. *Unimodality and normality:* Every fuzzy set of Φ_i must have only one element with maximum membership value equal to 1:

$$\forall A \in \Phi_i \exists | x \in X_i \ni' \mu_A(x) = 1.$$

This constraint helps to associate a linguistic label to each fuzzy set.

2. *Convexity:* Each fuzzy set of Φ_i must be convex, that is,

$$\forall A \in \Phi_i \forall x,y,z \in X_i : x \le y \le z \rightarrow \min\{\mu_A(x), \mu_A(z)\} \le \mu_A(y).$$

Although non-convex fuzzy sets could be interpretable (indeed convexity is not required in the definition of FOC), convex fuzzy sets are easier to understand.

3. *Coverage:* Any input must belong at least to one fuzzy set, with a membership value not smaller than a prefixed threshold value ε: $\exists \varepsilon \in (0,1) \forall i \forall x \in X_i \exists A \in \Phi_i \ni' \mu_A(x) \ge \varepsilon$. This constraint guarantees that each element of the input space is sufficiently represented by some fuzzy set (i.e. by a linguistic term). The concept of 'sufficiently' depends from the application and is formalized by the threshold ε (usually fixed to 0.5).
4. *Leftmost membership function:* The leftmost fuzzy set of Φ_i should assume its maximum membership value in m_i, that is: $\mu_{A_1^i}(m_i) = 1$. In this way, linguistic terms like 'low', 'small', etc., are easily modeled, although such constraint is not necessary in the definition of FOC.
5. *Rightmost membership function:* The rightmost fuzzy set of Φ_i should assume its maximum membership value in M_i, that is: $\mu_{A_{K_i}^i}(M_i) = 1$. With this constraint, it is easier to associate linguistic labels like 'high', 'tall', etc., to the rightmost fuzzy set.
6. *Disjunction:* Each fuzzy set in Φ_i must be sufficiently disjointed from the other fuzzy sets, i.e. they should not overlap too much. In this way the fuzzy sets are linguistically meaningful. The disjunction property can be formalized in several ways. In (Chow et al., 1999a), the definition of *overlap* is given as the relative measure of the support of the intersection between two fuzzy sets. To satisfy the disjunction property, such overlap must be limited in an interval that must be carefully chosen. In addition, the overlap measure cannot be directly applied to membership functions with infinite support (such as Gaussian functions) but requires a re-definition using α-

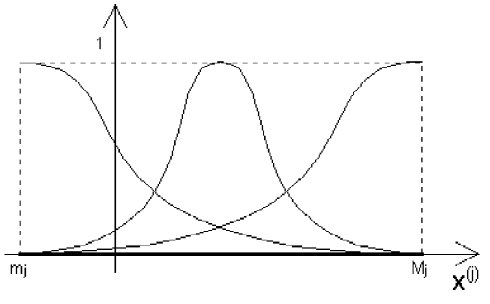


Fig. 2. Example of input partition into three fuzzy sets.

cuts. In this work we use the *possibility* measure, given in (Zadeh, 1978) and (Dubois & Prade, 1980), to compare fuzzy sets as defined by Gaussian membership functions since it quantifies the extent to which two fuzzy sets overlap (Pedrycz & Gomide, 1998). Unlike overlap measures, the possibility measure can be applied to any type of fuzzy sets. Precisely, the possibility measure of a fuzzy A with respect to fuzzy set B, defined as

$$\mathrm{Poss}(A, B) := \sup_x \min\{\mu_A(x), \mu_B(x)\}.$$

To obtain disjoint fuzzy sets, the possibility of each pair of fuzzy sets in Φ_i must be lower than a specified threshold π (usually $\pi = 0.5$): $\exists \pi \in (0,1) \forall i \forall A,B \in \Phi_i$: $\mathrm{Poss}(A, B) \leq \pi$. The use of possibility measure instead of overlap measure makes it easy to verify the disjunction property during learning since a single threshold needs to be set.

It can be easily observed that a grid partition of the input space, together with the use of Gaussian membership functions, satisfies all the properties above enumerated. Indeed Gaussian membership functions are unimodal, normal and convex, hence properties and are satisfied;

The formulas (8) and (9) guarantee the ε-coverage – hence property 3 – for any chosen $\varepsilon \in (0,1)$. Moreover the distribution of the centers guarantees the validity of properties 4 and 5;

When using Gaussian functions, the possibility measure between two adjacent fuzzy sets is equal to ε. Moreover, because of the strict monotonicity (in descending sense) of the membership value with respect to the distance from the center of the function, it is easy to derive that the possibility measure between two non-adjacent fuzzy sets is smaller than the fixed ε.

3. Parameter space of interpretable FKB

A Fuzzy Knowledge Base is characterized by several free parameters, defining the position and the width of each fuzzy set. The set of all possible values that parameters can assume, called *parameter space*, is usually highly dimensional. Usually, neuro-fuzzy approaches modify fuzzy set parameters in order to adapt fuzzy rules to the available data by a learning process. If the learning process is not constrained, fuzzy sets that do not respect the properties defined in Section 2.3 may be induced. In Fig. 3, an example of this lack of readability is graphically illustrated. The problem is that the training process modifies parameters in the whole parameter space, while only a small subset of this space corresponds to fuzzy sets that satisfy the properties given in the previous section. Here, we define a parameter space with reduced dimensionality and show that such dimensionality is necessary and sufficient to generate human-understandable fuzzy sets, in the sense formally defined by the properties given in Section 2.3.

Let Ω be the parameter space of the centers and the widths of the fuzzy sets used in the antecedents of the fuzzy rules. It is easy to observe that

$$\Omega \subseteq \Re^{D_1},$$
$$\text{where} \quad D_1 := 2 \sum_{i=1}^{n} K_i. \tag{10}$$

Let $\Omega^* \subseteq \Omega$ be the subspace of parameters for which properties 1–6 defined in Section 2.3 are satisfied (assuming $\pi = \varepsilon$). Now we prove that the dimensionality of Ω^* is smaller than D_1. To demonstrate this property, we will define a parametric hyper-surface that describes Ω^*, and show that the domain of such hyper-surface has D_2 dimension, with $D_2 < D_1$.

For sake of simplicity, we will consider only one axis of the input space, denoted by X_i. Since grid

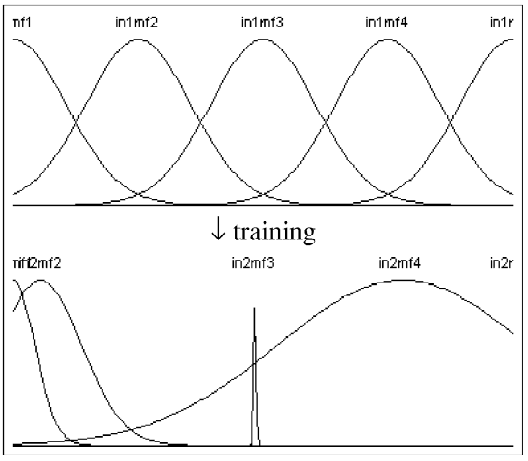


Fig. 3. Effects of training on the fuzzy sets of the FKB.

partition is used to build the FKB, this does not reduce the generality of the proof, because the fuzzy sets of each axis are independent. The following definition is given:

$$\Omega_i := \{[\omega_1, \omega_2, \ldots, \omega_{K_1}, \sigma_1, \sigma_2, \ldots, \sigma_{K_i}] \in \Re^{2K_i} | \forall h: \sigma_h > 0 \wedge \omega_h \leq \omega_{h+1}\}. \quad (11)$$

Every vector in Ω_i is directly related to a vector of membership functions with centers ω_h and widths σ_h, denoted by

$$[\mu_{A_1^i}, \mu_{A_2^i}, \ldots, \mu_{A_{K_i}^i}]. \quad (12)$$

Such relation is bijective, so we will use the parameters or membership functions interchangeably.

Let Ω_i^* be the subset of Ω_i for which properties 1–6 defined in Section 2.3 hold. The following set is defined:

$$T_i := \{[t_1, t_2, \ldots, t_{K_i-1}] \in \Re^{K_i-1} | m_i < t_1 < t_2 < \cdots < t_{K_i-1} < M_i\} \quad (13)$$

where the index i denoting the specific input axis has been dropped from $t_1, t_2, \ldots, t_{K_i-1}$ for ease of notation.

The sets Ω_i^* and T_i are related. Indeed, the following statement is true:

Lemma 1. *For each pair of adjacent Gaussian membership functions associated to a vector of Ω_i^*, there exists a unique point, between their centers, for which the membership values are equal. Formally:*

$$\forall h \in \{1,2, \ldots, K_i - 1\} \exists | t_h \in X_i \ni' (\mu_{A_h^i}(t_h) = \varepsilon = \mu_{A_{h+1}^i}(t_h)) \wedge (\omega_h < t_h < \omega_{h+1}). \quad (14)$$

The proof is outlined in Appendix A.

By collecting all the t_h in an ordered vector, a unique element of T_i is obtained. It should be noted that for a parameter vector in Ω_i^*, all the centers are distinct (see proof of Lemma 1), so all the t_h are distinct too. Fig. 4 illustrates an example of relation between a vector in Ω_i^* and a vector in T_i.

Moreover, starting from a vector in T_i, it is possible to construct a vector in Ω_i^*, as stated by the following lemma:

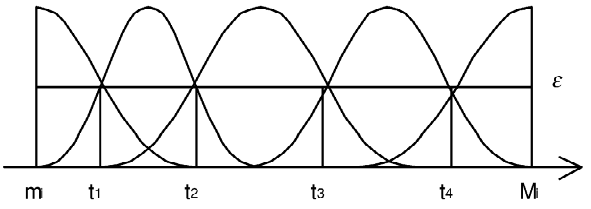


Fig. 4. Example of relation between Ω_i^* and T_i.

Lemma 2. *Given a vector $\mathbf{t} = [t_1, t_2, \ldots, t_{K_i-1}] \in T_i$, there exist a unique vector*

$$\mathbf{w} = [\omega_1, \omega_2, \ldots, \omega_{K_i}, \sigma_1, \sigma_2, \ldots, \sigma_{K_i}] \in \Omega_i^*$$

such that

$$\forall h \in \{1, 2, \ldots, K_i - 1\}: \mu_{A_h^i}(t_h) = \mu_{A_{h+1}^i}(t_h) = \varepsilon.$$

The proof of this lemma is outlined in Appendix B.

Lemmas 1 and 2 imply that there exists a bijective mapping between Ω_i^* and T_i. Such result can be directly extended to the entire input space $\mathbf{X}$, i.e. there exists a bijective mapping $\Gamma: \mathbf{T} \rightarrow \Omega^*$, where $\mathbf{T} := \times_{i=1}^{n} T_i$. The space $\mathbf{T}$ has D_2 dimensions, where

$$D_2 := \sum_{i=1}^{n} (K_i - 1). \quad (15)$$

As a consequence, the subspace Ω^* of readable fuzzy sets has D_2 dimensions. From the definition of D_1, given in Eq. (10), it can be observed that $D_2 < D_1$. Summarizing, the space Ω^* is a low dimensional hyper-surface (described by the function Γ in the domain $\mathbf{T}$) contained in a higher dimensional space Ω.

4. The neuro-fuzzy model and its learning algorithm

In this section, we propose a new neuro-fuzzy model that is able to keep valid during learning all the properties that formalize an 'understandable' FKB. Specifically, we develop a new neuro-fuzzy network architecture that is able to provide a fuzzy rule base composed of fuzzy sets in Ω^*. To achieve this, the proposed architecture uses $\mathbf{T}$ as parameter space of the antecedent part of the fuzzy rules (the parameter space of the consequence part depends on the particular FIS model, as explained in Section 1).

To find the optimal parameter vector in **T**, a learning algorithm based on the gradient-descent technique is also defined.

4.1. Neuro-fuzzy architecture

The proposed neuro-fuzzy network has a 5-layer feed-forward architecture (not fully connected), with the following layers:

- *Input Layer*. It simply spreads the input signals to the Membership Layer's neurons, jumping the second layer:
- *T-Layer*. The nodes of such layer are grouped in n blocks, each corresponding to one input variable. The i-th block is made of:
  - a fixed (non-adjustable) neuron that always fires the value m_i;
  - $K_i - 1$ adjustable neurons that fire the values $[t_{1i}, t_{2i}, \ldots, t_{K_i-1,i}] \in T_i$;
  - A fixed neuron tha always fires M_i.
- *Membership Layer*. Each neuron of this layer is connected with an input neuron and a pair of consecutive neurons of the T-Layer. These neurons compute membership values of each input to each fuzzy set of the corresponding axis. Given an input index i, the transfer function of a neuron of such layer is calculated as follows:

$$\mu_{A_h^i}(x_i) := \begin{cases} \exp\left(-\dfrac{(x_i - m_i)^2}{2\delta(2m_i - t_{1i}, t_{1i})^2} \right), & h = 1, \\ \exp\left(-\dfrac{(x_i - \gamma(t_{h-1,i}, t_{hi}))^2}{2\delta(t_{h-1,i}, t_{hi})^2} \right), & 1 < h < K_i, \\ \exp\left(-\dfrac{(x_i - M_i)^2}{2\delta(t_{K_i-1,i}, 2M_i - t_{K_i-1,i})^2} \right), & h = K_i \end{cases}$$

(functions γ and δ are defined in (B.2) and (B.3) in Appendix B). In this way we embody the function Γ in the network architecture. It is noteworthy to observe that these neurons are *fixed*, that is they do not have free parameters, because the centers and the widths are calculated by functions γ and δ, which are the basic components of Γ. In this way the understandability of fuzzy rules is maintained.

- *Rule Layer*. The neurons of this layer compute the truth value of each rule according to Eq. (4), hence they use the transfer functions:

$$\mu_r(\boldsymbol{x}) := \prod_{i=1}^{n} \mu_{A^i_{g(r,i)}}(x_i).$$

The function g is implemented by the connections between the Membership Layer and the Rule Layer. Full connection between the Rule Layer and the Membership Layer is requested when grid partition of the input space is adopted. The neurons of this layer have no free parameters.

- *Output Layer*. This layer is fully connected with the previous Rule Layer. Such connections are weighted by the free parameters k_{rj}. The output of each neuron is determined by the function y_j:

$$y_j := \frac{\sum_{r=1}^{R} \mu_r(\boldsymbol{x}) k_{rj}}{\sum_{r=1}^{R} \mu_r(\boldsymbol{x})}.$$

Fig. 5 illustrates the architecture of the neuro-fuzzy network in the case of a three-inputs zero-order TS fuzzy model.

4.2. Learning

The learning algorithm defined for the proposed neuro-fuzzy network is based the gradient descent technique. Given a *training set*

$$TS := \left\{ \langle \boldsymbol{x}^{(1)}, \boldsymbol{d}^{(1)} \rangle, \langle \boldsymbol{x}^{(2)}, \boldsymbol{d}^{(2)} \rangle, \ldots, \langle \boldsymbol{x}^{(P)}, \boldsymbol{d}^{(P)} \rangle \right\} \subset \Re^n \times \Re^m \tag{16}$$

the objective of training is to modify the free parameters of the network in order to minimize the following error function:

$$E := \frac{1}{2P} \sum_{p=1}^{P} \| \boldsymbol{y}^{(p)} - \boldsymbol{d}^{(p)} \|^2 \tag{17}$$

where $\boldsymbol{y}^{(p)} := [y_1^{(p)}, y_2^{(p)}, \ldots, y_m^{(p)}]$ is the output of the neuro-fuzzy network when the input $\boldsymbol{x}^{(p)} := [x_1^{(p)}, x_2^{(p)}, \ldots, x_n^{(p)}]$ is applied, and $\boldsymbol{d}^{(p)} := [d_1^{(p)}, d_2^{(p)}, \ldots, d_m^{(p)}]$ is the desired output. To minimize Eq. (17) each free parameter ζ. is iteratively updated according to the following formula:

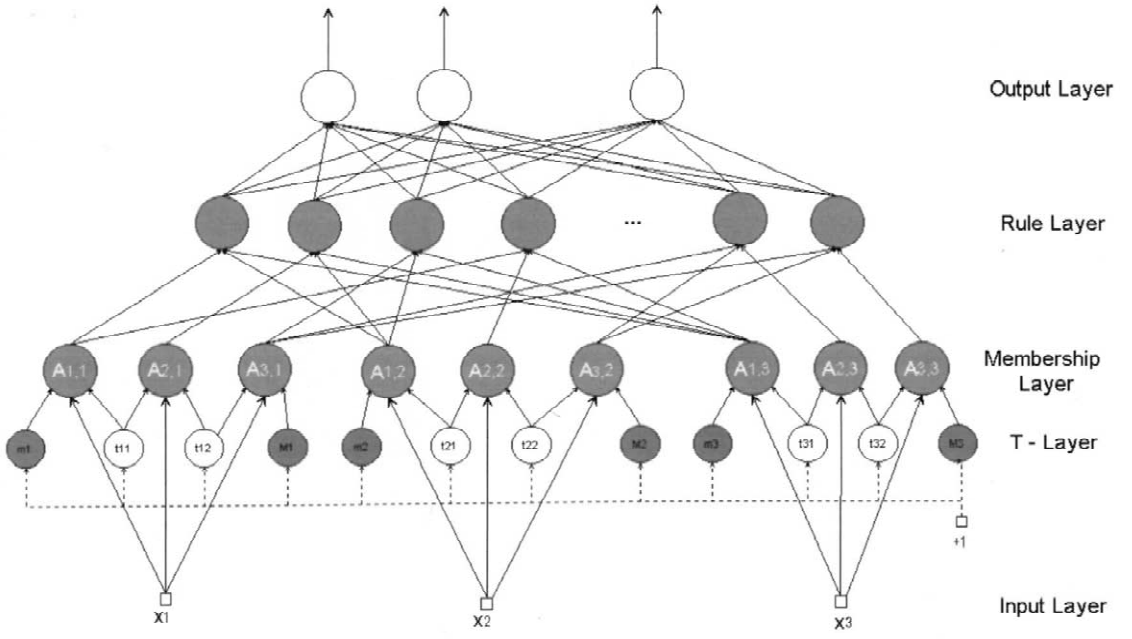


Fig. 5. Architecture of the proposed neuro-fuzzy network in the case of three-inputs zero-order TS fuzzy model. Gray-filled circles correspond to non-parametric neurons, while white circles correspond to neurons with at least one free parameter.

$$\Delta\zeta = -\eta\frac{\partial E}{\partial \zeta} \tag{18}$$

where η is the *learning rate* and E is rewritten as

$$E = \frac{1}{P}\sum_{p=1}^{P} E^{(p)} \tag{19}$$

where

$$E^{(p)} := \frac{1}{2}\sum_{j=1}^{m} (e_j^{(p)})^2 \tag{20}$$

and

$$e_j^{(p)} := y_j^{(p)} - d_j^{(p)}. \tag{21}$$

In our case, two types of parameters must be adapted, that is the consequent parameters k_{rj} and the parameters t_{hi} of the T-Layer.

The derivatives of $E^{(p)}$ with respect to k_{rj} are determined as follows:

$$\frac{\partial E^{(p)}}{\partial k_{rj}} = \frac{\partial E^{(p)}}{\partial e_j}\frac{\partial e_j}{\partial y_j}\frac{\partial y_j}{\partial k_{rj}} = e_j^{(p)} \cdot \frac{\mu_r(\boldsymbol{x}^{(p)})}{\sum_{s=1}^{R}\mu_s(\boldsymbol{x}^{(p)})}$$

$$= \frac{(y_j^{(p)} - d_j^{(p)})\mu_r(\boldsymbol{x}^{(p)})}{\sum_{s=1}^{R}\mu_s(\boldsymbol{x}^{(p)})}. \tag{22}$$

In order to calculate the derivatives of $E^{(p)}$ respect to t_{hi}, the following definition is given:

$$\mu(x, \gamma, \delta) := \exp\left(-\frac{(x-\gamma)^2}{2\delta^2}\right). \tag{23}$$

With this definition, the transfer functions of the Membership Layer can be rewritten as

$$\mu_{A_h^i}(x_i) = \begin{cases} \mu(x, m_i, \delta(2m_i - t_{1i}, t_{1i})), & h = 1, \\ \mu(x, \gamma(t_{h-1,i}, t_{hi}), \delta(t_{h-1,i}, t_{hi})), & 1 < h < K_i - 1, \\ \mu(x, M_i, \delta(t_{K_i-1,i}, 2M_i - t_{K_i-1,i})), & h = K_i. \end{cases} \tag{24}$$

Then, we have

$$\frac{\partial E^{(p)}}{\partial t_{hi}} = \sum_{j=1}^{m}\frac{\partial E^{(p)}}{\partial e_j}\frac{\partial e_j}{\partial y_j}\sum_{k=h}^{h+1}\sum_{\{r:g(r,i)=k\}}\frac{\partial y_j}{\partial \mu_r}\frac{\partial \mu_r}{\partial \mu_{A_k^i}}\frac{\partial \mu_{A_k^i}}{\partial t_{hi}}. \tag{25}$$

The derivation of $\partial\mu_{A_k^i}/\partial t_{hi}$ is given in Appendix C.

4.3. Choice of the learning rate

When the learning procedure is applied to train the proposed neuro-fuzzy network, an important question arises on the order of the elements $t_{1i}, t_{2,i}, \ldots, t_{K_i-1,i}$. Fixed an index $i \in \{1, 2, \ldots, n\}$, the definition of T_i described in Eq. (13), imposes the following relationship:

$$m_i < t_{1i} < t_{2i} < \cdots < t_{K_i-1,i} < M_i. \tag{26}$$

The respect of such constraint, which is fundamental to guarantee understandability of fuzzy rules, depends on the value of the learning rate. In this section we prove that a choice of a small learning rate avoids any trouble. Moreover, too large learning rates could be automatically made smaller in order to avoid a violation of the previous constraint. The demonstration is given for an input index i, but such index is omitted to lighten the notation.

In order to give an evaluation of the learning rate which guarantees the satisfaction of constraint (26), we consider the succession of instances of the free parameters $t_h(\tau)$, where τ is the training epoch. Suppose that initially the constraint (26) is respected, that is,

$$\boldsymbol{t}(0) := [t_1(0), t_2(0), \ldots, t_{K_i-1}(0)] \in T. \tag{27}$$

Such supposition is verified if grid partition is applied on the input space. Consider now $h \in \{1, 2, \ldots, K_i - 2\}$ and a training iteration step τ. In the next training step $\tau + 1$, the following statement is true:

$$[t_h(\tau+1), t_{h+1}(\tau+1)] = [t_h(\tau), t_{h+1}(\tau)] - \eta(\tau)\left[\frac{\partial E}{\partial t_h}(\boldsymbol{t}(\tau)), \frac{\partial E}{\partial t_{h+1}}(\boldsymbol{t}(\tau))\right]. \tag{28}$$

In the successive iteration, we have

$$\Delta_h(\tau+1) = \Delta_h(\tau) + \eta(\tau)\left(\frac{\partial E}{\partial t_h}(\boldsymbol{t}(\tau)) - \frac{\partial E}{\partial t_{h+1}}(\boldsymbol{t}(\tau))\right) \tag{29}$$

where $\Delta_h(\tau) := t_{h+1}(\tau) - t_h(\tau)$.

The following property is true:

$$\Delta_h(\tau+1) < \Delta_h(\tau) \leftrightarrow \frac{\partial E}{\partial t_h}(\boldsymbol{t}(\tau)) < \frac{\partial E}{\partial t_{h+1}}(\boldsymbol{t}(\tau)). \tag{30}$$

For a given h and a training step τ, the constraint (26) is respected when $\Delta_h(\tau) > 0$. Suppose that after the τ-th iteration, $\Delta_h(\tau) > 0$. The domain constraint is verified for any positive learning rate when $\Delta_h(\tau + 1) \geq \Delta_h(\tau)$, but if this inequality is not verified, then we must consider only positive learning rates for which $\Delta_h(\tau + 1) > 0$. Assume that $\Delta_h(\tau + 1) < \Delta_h(\tau)$. Then we have

$$\begin{aligned}
&\Delta_h(\tau+1) > 0 \\
&\leftrightarrow \underbrace{\Delta_h(\tau)}_{>0} + \underbrace{\eta(\tau)}_{>0}\underbrace{\left(\frac{\partial E}{\partial t_h}(\boldsymbol{t}(\tau)) - \frac{\partial E}{\partial t_{h+1}}(\boldsymbol{t}(\tau))\right)}_{<0} > 0 \\
&\leftrightarrow \eta(\tau) < \frac{\Delta_h(\tau)}{\frac{\partial E}{\partial t_{h+1}}(\boldsymbol{t}(\tau)) - \frac{\partial E}{\partial t_h}(\boldsymbol{t}(\tau))}.
\end{aligned} \tag{31}$$

The previous relation guarantees that, given an index h, $t_h(\tau + 1) < t_{h+1}(\tau + 1)$. However, the constraint defined in (26) requires also the following relationships:

$$(m_i < t_1) \wedge (t_{K_i-1} < M_i). \tag{32}$$

In the same way we have derived the upper bounds given in Eq. (31), it can be proved that inequalities in Eq. (32) are satisfied when

$$\eta(\tau) < \frac{\Delta_0(\tau)}{\frac{\partial E}{\partial t_1}(\boldsymbol{t}(\tau))}$$

and

$$\eta(\tau) < \frac{\Delta_{K_i}(\tau)}{-\frac{\partial E}{\partial t_{K_i-1}}(\boldsymbol{t}(\tau))} \tag{33}$$

where

$$\Delta_0(\tau) := t_1(\tau) - m_i \text{ and } \Delta_{K_i}(\tau) := M_i - t_{K_i-1}. \tag{34}$$

From Eqs. (31) and (33), the following general statement is true:

$$\forall \tau \geq 0: \mathbf{t}(\tau) \in \boldsymbol{T} \Rightarrow \left(\mathbf{t}(\tau+1) \in \mathbf{T} \leftrightarrow \eta(\tau) < \min \left\{ \frac{\Delta_h(\tau)}{\frac{\partial E}{\partial t_{h+1}}(\mathbf{t}(\tau)) - \frac{\partial E}{\partial t_h}(\mathbf{t}(\tau))} \;\middle|\; \begin{array}{l} \Delta_h(\tau+1) < \Delta_h(\tau) \\ h = 0,1,\ldots,K_i - 1 \end{array} \right\} \right). \tag{35}$$

The previous statement guarantees the existence of a learning rate for which constraint (26) is respected. From a practical point of view, if the learning rate in a training step is too large and generates non-valid configurations of free parameters, then a new one can be chosen, for instance by halving the learning rate, and verifying the adequacy of the new value.

In addition, an analysis of the magnitude of the learning rate can be performed. Firstly, from Eq. (25) we observe that the derivatives $\partial E / \partial t_h$ never go to infinity, since

$$\lim_{t_h \to \pm\infty} \partial E / \partial t_h \propto \lim_{t_h \to \pm\infty} \sum_k \partial \mu_{A_k^i} / \partial t_{hi} = 0$$

and $\partial E / \partial t_h$ is continue with respect to t_h except in t_{h+1}. It is therefore possible to find a value $M > 0$ such that

$$\forall h \in \{1, 2, \ldots, K_i - 1\}: \left| \frac{\partial E}{\partial t_h} \right| < M. \tag{36}$$

From Eqs. (35) and (36) we can deduce that the learning rate can be chosen in such a way that

$$\eta(n) < \frac{\min\{\Delta_h(\tau) | h = 1, 2, \ldots, K_i - 1\}}{2M}. \tag{37}$$

Though the previous relation does not provide a precise evaluation of the learning rate, we can roughly say that when the free parameters t_h are well distanced, like in the initial training epochs, it is not difficult to choose good values of the learning rate, for which the convergence is not slow and the domain constraint is still guaranteed. Moreover, we can prove that the choice of the learning rate becomes less important when two free parameters come closer. The following lemmas are useful for this purpose:

$$\forall h \in \{1, 2, \ldots, K_i - 1\}: \lim_{t_h \to t_{h+1}} \frac{\partial E}{\partial t_h} = 0.$$

Since the considered architecture is feed-forward, the derivative of the cost function E with respect to the free parameter t_h can be expressed in terms of summation of the derivatives $\partial \mu_{A_k^i} / \partial t_k$, for some set of indices k. It is therefore sufficient to prove the following statement:

$$\forall k \in \{1, 2, \ldots, K_i - 1\}: \lim_{t_k \to t_{k+1}} \frac{\partial \mu_{A_k^i}}{\partial t_k} = 0. \tag{38}$$

The proof of the previous statement is straightforward, if Eqs. (C.1)–(C.4) in Appendix C are considered. Using Landau notation, it can be observed that

$$\frac{\partial \mu_{A_k^i}}{\partial t_k} \sim \frac{O(\mathrm{e}^{-(t_k - t_{k+1})^{-2}})}{O((t_k - t_{k+1})^3)} \underset{t_k \to t_{k+1}}{\to} 0 \tag{39}$$

$$\forall h \in \{1, 2, \ldots, K_i - 2\}: \lim_{t_h \to t_{h+1}} \left(\frac{\partial E}{\partial t_{h+1}} - \frac{\partial E}{\partial t_h} \right) = 0.$$

The proof is straightforward.

Corollary. *The derivatives of the cost function respect to the free parameters t_h are infinitesimal with order greater than one, hence:*

$$\forall h \in \{1, 2, \ldots, K_i - 2\}: \quad \frac{\partial E}{\partial t_{h+1}} - \frac{\partial E}{\partial t_h} \sim o(t_h - t_{h+1}). \tag{40}$$

The corollary implies that

$$\forall \varepsilon > 0 \exists \delta > 0 \ni' \forall t_h: 0 < t_{h+1} - t_h < \delta \rightarrow \left| \frac{\partial E}{\partial t_{h+1}} - \frac{\partial E}{\partial t_h} \right| < \varepsilon (t_{h+1} - t_h) = \varepsilon \Delta_h. \tag{41}$$

Therefore, fixed an ε positive but arbitrarily small, there exists a positive value δ such that

$$0 < \Delta_h < \delta \rightarrow \frac{\Delta_h}{\left| \frac{\partial E}{\partial t_{h+1}} - \frac{\partial E}{\partial t_h} \right|} > \frac{1}{\varepsilon}. \tag{42}$$

Comparing Eq. (31) with Eq. (42) we can deduce that, the smaller is the value $\Delta_h(\tau)$ for some h, the higher is the upper bound for the learning rate to verify the constraint (26). In conclusion it is possible

to affirm that the choice of a 'good' learning rate is more important in the early stages of the training, in order to allow a graduate learning without violating the domain constraint. During the training, if a non-increasing learning rate is adopted, the possibility of violating constraint (26) becomes harder and harder.

5. Simulation results

To demonstrate our approach to extract human-understandable fuzzy knowledge base from data, simulations on a well-known identification problem of a non-linear system (Narendra & Parthasarathy, 1990) and a real-world example from medicine (Wolberg & Mangasarian, 1990) have been carried out. The results are compared with other methods, whenever possible.

5.1. A simple example

The goal of this first simulation is to show how the proposed approach can extract a fuzzy rule base from data and how this rule base turns out to be interpretable and accurate as well. A very simple example concerning the identification of a non-linear system has been considered. The results were compared with those obtained by an ANFIS network (Jang, 1993) implementing a zero-order TSK fuzzy model. The ANFIS code was taken from the Matlab©Fuzzy Toolbox.

The system to be identified is a static non-linear system with two inputs and a single output. The input/output relation of such system is described by

$$y = \left(1 + x_1^{-2} + x_2^{-1.5}\right)^2, \quad 1 \le x_1, x_2 \le 5. \tag{43}$$

A three-dimensional I/O graph of this non-linear system is depicted in Fig. 6.

The training set was obtained by computing the function (43) on 50 pairs (x_1, x_2) randomly taken in $[1, 5] \times [1, 5]$. Each input domain was normalized in the interval $[-1, 1]$ and then partitioned into 5 fuzzy sets using the grid partition technique. The resulting membership functions, plotted in Fig. 7, provide a FKB of 25 fuzzy rules, with all the consequent values initialized to zero. This initial FKB was embedded into the proposed neuro-fuzzy network and into the ANFIS network to establish the structure and initial parameters. Then, both the networks were trained for 5000 epochs, with learning rate fixed to 0.01 in each epoch. The standard MSE was used as cost function during learning.

Fig. 8 compares the fuzzy partitions obtained after the learning process for the two architectures. It can be seen that the fuzzy sets generated by our approach are much more readable than those obtained by ANFIS. Also, as it can be seen from Fig. 9, the

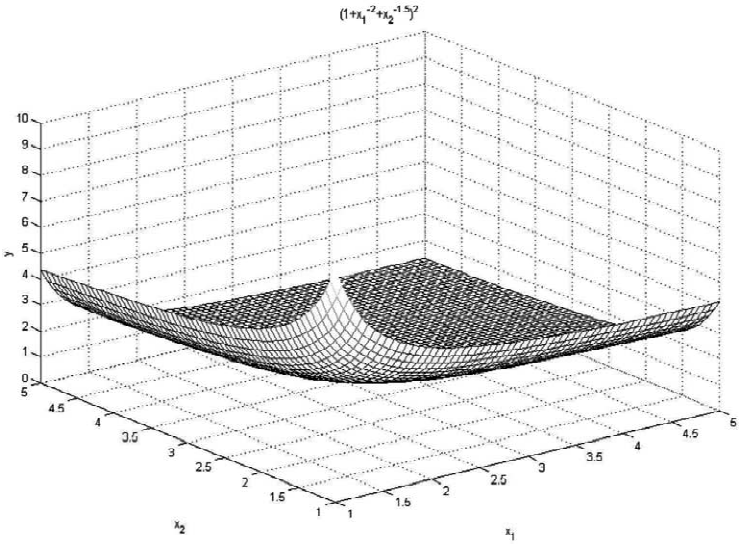


Fig. 6. Output surface of the non-linear system.

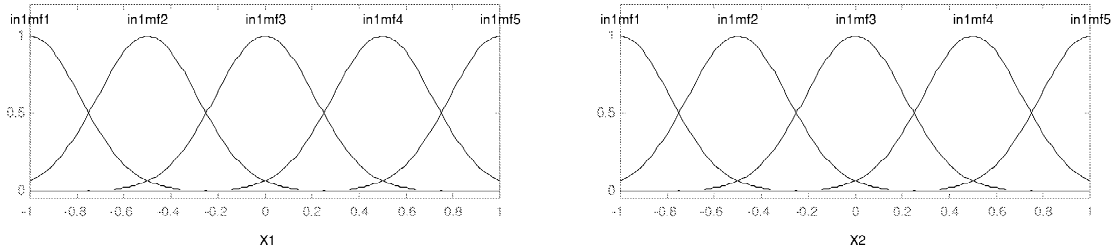
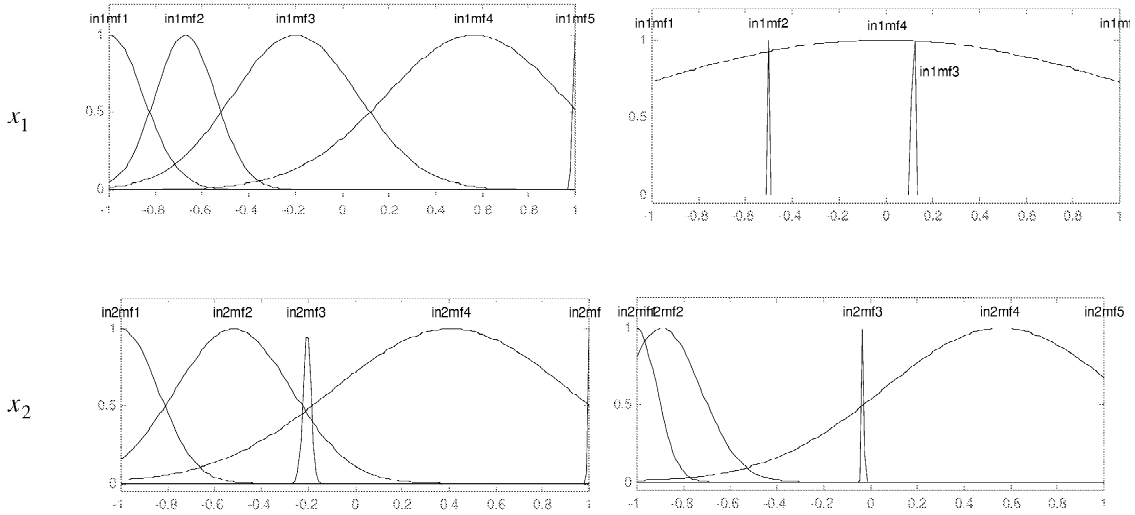


Fig. 7. Initial fuzzy partition of the two input variables.



(a) (b)

Fig. 8. Fuzzy partition of the two input domains obtained after learning in the case of the proposed neuro-fuzzy network (a) and the ANFIS network (b).

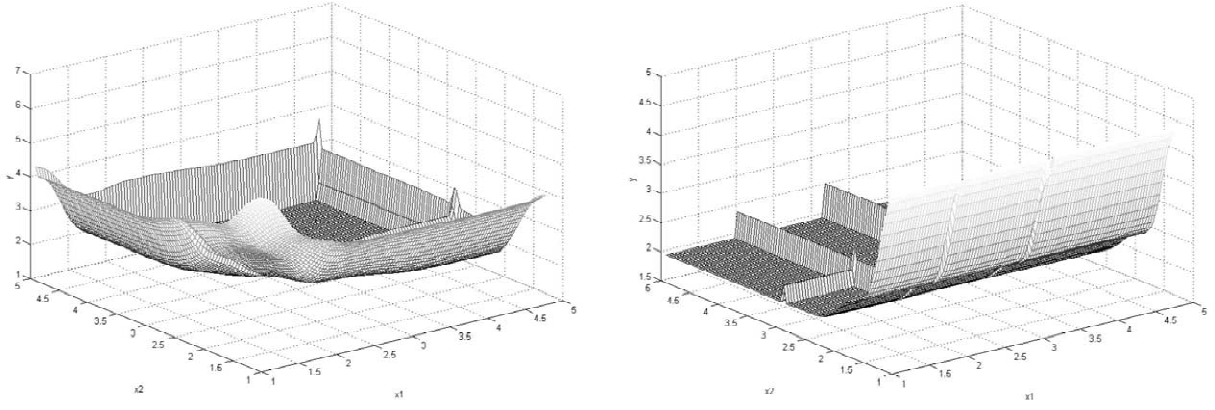


Fig. 9. Output surface of the FKB extracted by the proposed neuro-fuzzy network (a) and the ANFIS network (b).

output surface provided by the FKB extracted with the proposed approach approximates quite well the desired input/output mapping, while the approximation provided by the FKB generated by ANFIS network is rather poor. Moreover, as shown in Fig. 10, the trend of the MSE in the case of our learning algorithm is smoother in comparison to the ANFIS learning algorithm, providing a final MSE of 0.0053 which is lower than the final MSE (0.0301) achieved in the ANFIS case.

Our approach overcomes also other fuzzy approaches in terms of accuracy. For example, the Sugeno–Yasukawa model (Sugeno & Yasukawa, 1993) and the fuzzy model in (Huang & Chu, 1999) provide for the same data set, an MSE of 0.079 and 0.022, respectively. Comparison in terms of interpretability was not possible since no semantic issue is addressed by such fuzzy modeling methods.

In conclusion, through this example, we have illustrated how the proposed approach is able to extract a FKB with interpretable fuzzy sets and with a good approximation ability.

5.2. A real-word example

To assess the effectiveness of the proposed approach, a more realistic example, with higher dimensionality, was considered to provide an idea of the network behavior in practice. The example is the Wisconsin Breast Cancer (WBC) data set, provided by W.H. Wolberg from the University of Wisconsin Hospitals, Madison (Wolberg & Mangasarian, 1990). The data set contains 699 cases belonging to one of two classes (benign: 458 cases, or malignant: 241 cases). Each case is represented by an ID number and nine attributes (x_1: clump thickness; x_2: uniformity of cell size; x_3: uniformity of cell shape; x_4: marginal adhesion; x_5: single epithelial cell size; x_6: bare nuclei; x_7: bland chromatin; x_8: normal nucleoli; x_9: mitoses). All attribute values are integers from the domain $\{1, \ldots, 10\}$. There are 16 cases with missing values. Since our model cannot yet deal with missing values, we used only the complete 683 cases: 444 in class ‘benign’ and 239 in class ‘malignant’.

To cope with the high-dimensionality of this data set and avoid generation of a high number of rules by grid partition, the number of input variables was reduced by applying a feature selection algorithm that we have developed in (Castellano & Fanelli, 2000b). After the feature selection process, we find that the most significant attributes are x_1 (clump thickness), x_3 (uniformity of cell shape) and x_6 (bare nuclei), while the less significant attributes are x_2 and x_5, both related to cell size, as also stated in (Duch, Adamczak & Grabczewski, 2001). Hence we use the three selected features and define two initial fuzzy sets for each variable. By this, we created a fuzzy partition of eight rules with null consequents, which was used to establish the structure and initial parameters of our neuro-fuzzy network.

The data set was split randomly in a training set of 342 cases and a test set of 341 cases, so that each set contains roughly the same number of patterns for each class. After 100 epochs of our learning algorithm, we obtained very distinguishable fuzzy sets

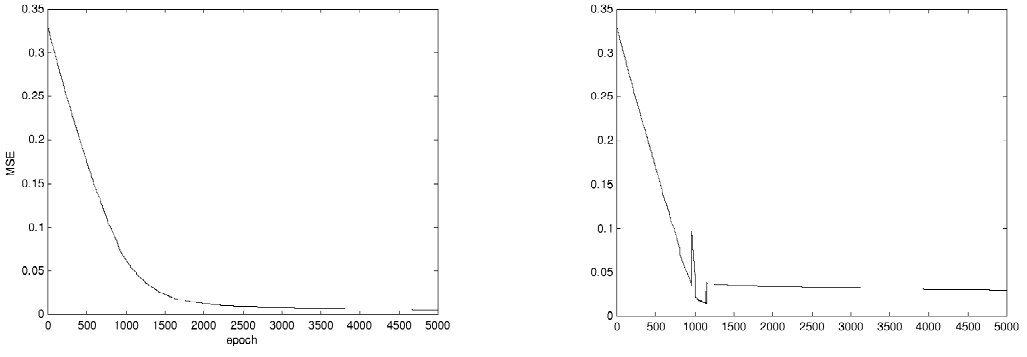
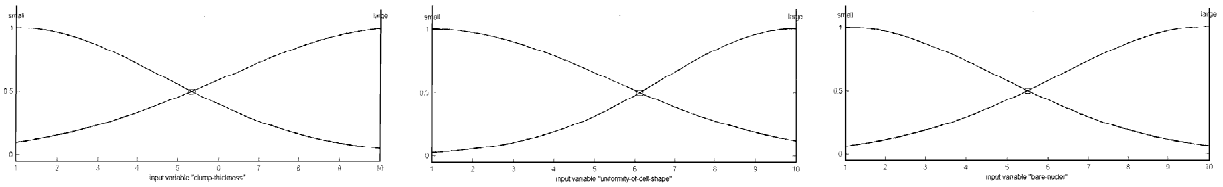


Fig. 10. Trend of the MSE during learning in the case of the proposed network (a) and the ANFIS network (b).



Small: ω=1.026, σ=3.848 Large: ω=9.93, σ=3.88	Small: ω=1.024, σ=3.844 Large: ω=9.934, σ=3.889	Small: ω=0.982, σ=3.804 Large: ω=9.904, σ=3.911

Fig. 11. The final membership functions and their parameters.

and a FKB with a classification rate of 95.32% on the training set (16 errors), 96.48% on the test set (12 errors), and 95.90% (28 errors) on the whole dataset. For each variable, the two fuzzy sets (which are labeled 'small' and 'large') are represented by very distinct membership functions which overlap with neighbors at membership degree 0.5, and so they are nicely interpretable (see Fig. 11).

Since in application areas like medicine not only the accuracy but also the rule simplicity and comprehensibility is important, the extracted FKB was simplified via our pruning algorithm (Castellano & Fanelli, 1996) that automatically reduces the number of rules while completely preserving the accuracy and the interpretability of fuzzy sets. After rule simplification, there are only four fuzzy rules in the rule base with unchanged fuzzy sets and accuracy with respect to the eight-rule base. Figs. 12 and 13 show the final four rules in a graphic and a textual form.

To evaluate the effectiveness of such results, they were compared with those obtained by the NEFCLASS neuro-fuzzy system, also applied to this dataset in (Nauck & Kruse, 1999), under our experimental setting (i.e. removing 16 cases with missing values and partitioning the data set into 342

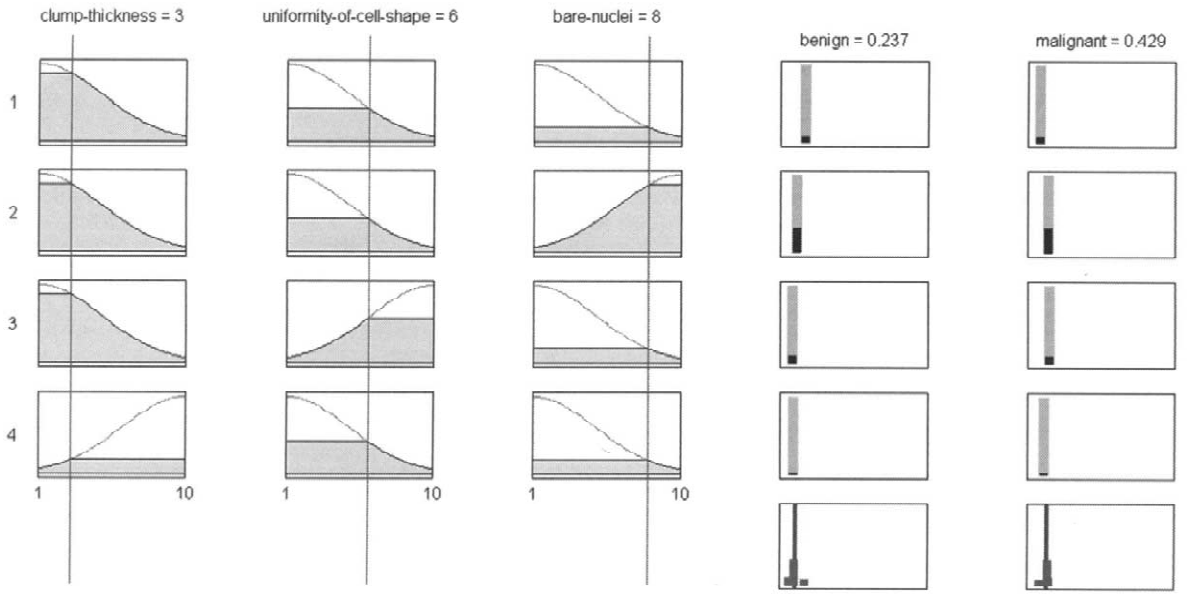


Fig. 12. The 4-rule FKB obtained by our approach for the WBC data set.

Rule 1: IF (clump thickness IS *small*) AND (uniformity of cell shape IS *small*) AND (bare nuclei IS *small*)
THEN Patient belongs to class 'benign' with degree 0.67 and to class 'malignant' with degree 0.11
Rule 2: IF (clump thickness IS *small*) AND (uniformity of cell shape IS *small*) AND (bare nuclei IS *large*)
THEN Patient belongs to class 'benign' with degree 0.022 and to class 'malignant' with degree 0.47
Rule 3: IF (clump thickness IS *small*) AND (uniformity of cell shape IS *large*) AND (bare nuclei IS *small*)
THEN Patient belongs to class 'benign' with degree 0.00 and to class 'malignant' with degree 0.54
Rule 4: IF (clump thickness IS *large*) AND (uniformity of cell shape IS *small*) AND (bare nuclei IS *small*)
THEN Patient belongs to class 'benign' with degree 0.081 and to class 'malignant' with degree 0.32

Fig. 13. The 4-rule FKB obtained by our approach for the WBC data set.

samples for training and 341 for testing). Using 4 rules and 'best-per-class' rule learning (that can be regarded as a kind of pruning strategy), NEFCLASS achieves 8 errors on the training set (97.66% correct), 18 errors on the test set (94.72% correct) and 26 errors (96.2% correct) on the whole set. Despite the slightly better accuracy of NEFCLASS on the whole dataset, it should be noted that in our case higher accuracy on the test set (generalization ability) is achieved with even a very small number of input variables with respect to the 9 features used by NEFCLASS, thus resulting in a more simple and interpretable rule base. It should be noted that our results comes from the application of automatic procedures, both for learning and simplification, that do not require human intervention unlike the NEFCLASS system.

In addition, to obtain a more feasible estimate of the classification error, we carried out a 10-fold cross validation. The data set was randomly split into 10 equally sized parts without changing the class frequency. Each part was used as test set for the FKB extracted from the remaining data. The estimate of the classification accuracy was then computed as average of classification errors of all 10 FKB on their test set. The mean error on the 10 test sets was 96.08% and the average number of rules was 4.2 (ranging from 3 to 7 rules). A comparison with other neural, fuzzy and traditional classifiers developed for the same dataset is summarized in Table 1. It can be seen that the estimated classification of our FKB is comparable with most of the considered models. Indeed, most of the modeling methods reported in Table 1 pursue only accuracy as ultimate goal and take no care about the interpretability of the knowledge representation.

6. Conclusions

Comprehensibility of knowledge extracted from data is a very attractive feature for a neuro-fuzzy

Table 1
Comparing our approach for the WBC data set to some other approaches

Method	Accuracy	Reference
IncNet	97.1	Jankowski & Kadirkamanathan (1997)
k-NN	97.0±0.12	Duch, Adamczak & Grabczewski (2001)
Fisher LDA	96.8	Ster & Dobnikar (1996)
MLP + backprop	96.7	Ster & Dobnikar (1996)
LVQ	96.6	Ster & Dobnikar (1996)
Bayes (pairwise dependent)	96.6	Ster & Dobnikar (1996)
Naïve Bayes	96.4	Ster & Dobnikar (1996)
DB-CART	96.2	Shand & Breiman (1996)
LDA	96.0	Ster & Dobnikar (1996)
LFC, ASI, ASR dec. trees	94.4–95.6	Ster & Dobnikar (1996)
CART (dec. tree)	93.5	Shand & Breiman (1996)
Quadratic DA	34.5	Ster & Dobnikar (1996)
FSM, 12 fuzzy rules	96.5	Duch et al. (2001)
SSV, 3 crisp rules	96.3±0.2	Duch et al. (2001)
NEFCLASS-X, 2 fuzzy rules using 5–6 variables	95.06	Nauck & Kruse (1999)
our approach, 4 (4.2) fuzzy rules using 3 variables	96.08	Our result

approach, since it establishes a bridge between the so-called symbolic reasoning paradigm, that provides explicit knowledge representation, and the sub-symbolic paradigm, where systems like neural networks discover automatically knowledge from data. However, a fuzzy knowledge base that is precise and interpretable as well can hardly be found by a completely automatic learning process. Our work aims to make a step further to achieve this objective.

A new neuro-fuzzy architecture and its learning algorithm have been proposed that is able to acquire knowledge from data in the form of fuzzy rules easily interpretable by humans. Interpretability of fuzzy sets is preserved during learning by allowing the free parameters to vary in a parameter subspace containing only configurations satisfying a set of formal properties.

A simple benchmark and a real-world example have been considered to illustrate the key features of the proposed model and its related learning algorithm. The given examples highlight that the proposed approach can be an effective technique for knowledge extraction, providing fuzzy knowledge bases with accuracy comparable and often significantly better than those of other state-of-the-art models. Also, simulation results confirm the essential feature of the proposed neuro-fuzzy architecture, that is the ability to produce final rule bases that are also interpretable since they contain well distinct fuzzy sets. On the overall, the reported results indicate that our approach is a valid tool to automatically extract fuzzy rules from data providing a good balance between accuracy and readability.

Further extensions of the proposed model may concern the use of different *t*-norms in the rule inference mechanism and the use of different measures to evaluate the degree of overlapping between fuzzy sets. Finally, since the proposed approach focuses only on interpretability of rule antecedents, future work is aimed to extend the proposed architecture so as to deal with interpretability of consequents too.

Appendix A

Proof of Lemma 1. Let t_h be the value of X_i such that $(\mu_{A_h^i}(t_h) = \varepsilon) \wedge (t_h > \omega_h)$, with $h \in \{1, 2, \ldots, K_i - 1\}$. The existence of t_h is guaranteed by the properties of Gaussian membership functions – continuity and monotonicity with respect to the distance from center. We consider the value $t_h' \in X_i \ni' (\mu_{A_{h+1}^i}(t_h') = \varepsilon) \wedge (t_h' < \omega_{h+1})$. The existence of t_h' is also guaranteed. Now we prove that $t_h = t_h'$. Indeed, if $t_h < t_h'$ then the middle point $t_h'' := (t_h + t_h'/2)$ is different from both t_h and t_h'. In particular, we have $\omega_h < t_h < t_h'' < t_h' < \omega_{h+1}$, hence $(\mu_{A_h^i}(t_h'') < \varepsilon) \wedge (\mu_{A_{h+1}^i}(t_h'') < \varepsilon)$. If this condition holds, then ε-coverage is not guaranteed.

Suppose, ad absurdum, that there is a fuzzy set for which the membership of t_h'' is greater or equal to ε, that is: $\exists k \in \{1, 2, \ldots, K_i - 1\} \backslash \{h, h+1\} \ni' \mu_{A_k^i}(t_h'') \geq \varepsilon$. Two cases must be considered. If $k < h$ then: $\omega_k \leq \omega_h$. Because of the normality and continuity of the membership functions, it follows that

$$(\exists \zeta \in [\omega_h, t_h''] \ni' \mu_{A_h^i}(\xi) = \mu_{A_k^i}(\xi) > \varepsilon)$$
$$\Rightarrow (\exists \xi \in [\omega_h, t_h''] \ni' \min\{\mu_{A_h^i}(\xi), \mu_{A_k^i}(\xi)\} > \varepsilon).$$

In this case, $\mathrm{Poss}(A_h^i, A_k^i) > \varepsilon$, hence the disjunction property is not respected. Similar considerations can be made for the case $k > h + 1$.

In conclusion, we can say that $t_h \geq t_h'$. But, if $t_h > t_h'$, again $\mathrm{Poss}(A_{h+1}^i, A_k^i) > \varepsilon$ and so the disjunction property is not respected. This is absurd, since the fuzzy sets considered have been generated by a parameter vector belonging to Ω_i^*. Since the existence of t_h and t_h' is guaranteed, then $t_h = t_h'$.

The unicity of t_h is established by the monotonicity of Gaussian membership functions for values greater than the center. □

Appendix B

Proof of Lemma 2. The proof of the lemma consists in the formulation of a functional transformation Γ: $\Gamma_i \rightarrow \Omega_i$ such that $\forall \boldsymbol{t} \in T_i$: $\Gamma(\boldsymbol{t}) \in \Omega_i^*$.

The function Γ is defined as follows:

$$\Gamma(t_1, t_2, t_3, \ldots, t_{K_1-2}, t_{K_i-1}) :=$$
$$\begin{bmatrix} m_i, \gamma(t_1, t_2), \gamma(t_2, t_3), \ldots, \gamma(t_{K_i-2}, t_{K_i-1}), M_i, \\ \delta(2m_i - t_1, t_1), \delta(t_1, t_2), \delta(t_2, t_3), \ldots, \delta(t_{K_i-2}, t_{K_i-1}), \delta(t_{K_i-1}, 2M_i - t_{K_i-1}) \end{bmatrix} \quad \text{(B.1)}$$

where

$$\gamma(x, y) := \frac{x + y}{2}, \quad \text{(B.2)}$$

$$\delta(x, y) := \frac{y - x}{2\sqrt{-2\ln\varepsilon}}. \tag{B.3}$$

It is easy to verify that

$$\forall x < y < z: \gamma(x, y) < \gamma(y, z) \tag{B.4}$$

$$\forall x < y: \delta(x, y) > 0. \tag{B.5}$$

Because of the validity of Eqs. (B.4) and (B.5), it is possible to state that $\Gamma(\boldsymbol{t}) \in \Omega_i$. Let $[A_1^i, A_2^i, \ldots, A_{K_i}^i]$ be the vector of fuzzy sets induced by $\Gamma(\boldsymbol{t}) = \boldsymbol{w} = [\omega_1, \omega_2, \ldots, \omega_{K_i}, \sigma_1, \sigma_2, \ldots, \sigma_{K_i}] \in \Omega_i$. Now we demonstrate that properties 1–5 defined in Section 2.3 are satisfied (assuming $\pi = \varepsilon$).

Unimodality, *normality* and *convexity* are always satisfied because the membership functions are assumed to be Gaussian. This property is valid independently on the function Γ.

Coverage. The input space X_i can be partitioned into $K_i - 1$ intervals, defined as

$$I_h := \begin{cases} [\omega_h, \omega_{h+1}), & 1 \le h < K_i - 1, \\ [\omega_h, \omega_{h+1}], & h = K_i - 1. \end{cases}$$

It is easy to observe that such intervals yield a partition of the input space, that is,

$$h \neq k \rightarrow (I_h \cap I_k = \varnothing) \wedge \left(X_i = \bigcup_{h=1}^{K_i - 1} I_h \right).$$

Therefore each input can be assigned to just one interval, i.e. $\forall x \in X_i \exists h \ni' x \in I_h$. Also, in every interval I_h there are at least two fuzzy sets that intersect. In particular, for a given h, the fuzzy sets A_h^i and A_{h+1}^i can be considered. The ε-cut of the two fuzzy sets are calculated as follows:

$$\forall 1 \le h \le K_i - 1:$$

$$\begin{cases} [A_h^i]_\varepsilon \cap I_h = \{x \in X_i | \mu_{A_h^i}(x) \ge \varepsilon\} \cap I_h = [\omega_h, \omega_h + \sigma_h\sqrt{-2\ln\varepsilon}], \\ [A_{h+1}^i]_\varepsilon \cap I_h = \{x \in X_i | \mu_{A_{h+1}^i}(x) \ge \varepsilon\} \cap I_h = [\omega_{h+1} - \sigma_{h+1}\sqrt{-2\ln\varepsilon}, \omega_{h+1}), \end{cases}$$

The intersection points of the two intervals are all the points x such that

$$\begin{cases} \omega_h \le x \le \omega_h + \sigma_h\sqrt{-2\ln\varepsilon}, \\ \omega_{h+1} - \sigma_{h+1}\sqrt{-2\ln\varepsilon} \le x < \omega_{h+1}. \end{cases} \tag{B.6}$$

For ease of notation, we define

$$t_0 := 2m_i - t_1,$$
$$t_{K_i} := 2M_i - t_{K_i - 1}.$$

It is easy to observe that $\gamma(t_0, t_1) = m_i$ and $\gamma(t_{K_i - 1}, t_{K_i}) = M$. Hence, the inequality system (B.6) can be rewritten as

$$\begin{cases} \gamma(t_{h-1}, t_h) \le x \le \gamma(t_{h-1}, t_h) + \delta(t_{h-1}, t_h)\sqrt{-2\ln\varepsilon}, \\ \gamma(t_h, t_{h+1}) - \delta(t_h, t_{h+1})\sqrt{-2\ln\varepsilon} \le x < \gamma(t_h, t_{h+1}). \end{cases}$$

Expanding the definitions of γ and δ, and making some simplifications, we have

$$\begin{cases} \dfrac{t_{h-1} + t_h}{2} \le x \le t_h, \\ t_h \le x < \dfrac{t_h + t_{h+1}}{2}. \end{cases}$$

Since $t_{h-1} < t_h < t_{h+1}$, there is only one point of intersection between A_h^i and A_{h+1}^i and I_h, and it is $x = t_h$. Since this point belongs to the ε-cuts of the two fuzzy sets, its membership value is not less than ε. Moreover, the union of the two ε-cut coincides with the entire interval I_h. Hence we can state that ε-coverage is guaranteed in each I_h and consequently in X_i.

Leftmost and rightmost membership functions are guaranteed, since the center of the first fuzzy set is imposed to be m_i and the center of the last fuzzy set is M_i.

Disjunction. As seen in the previous point, for each interval I_h the ε-cuts of the fuzzy sets A_h^i and A_{h+1}^i intersect only in the point $x = t_h$ where the membership value is

$$\mu_{A_h^i}(t_h) = \mu_{A_{h+1}^i}(t_h) = \exp\left(- \frac{\left(t_h - \dfrac{t_h + t_{h-1}}{2} \right)^2}{2\left(\dfrac{t_h - t_{h-1}}{2\sqrt{-2\ln\varepsilon}} \right)^2} \right)$$
$$= \exp(\ln\varepsilon) = \varepsilon.$$

Since this is the unique intersection point, it is true that $\forall x \in I_h$: $\min\{\mu_{A_h^i}(x), \mu_{A_{h+1}^i}(x)\} \le \varepsilon$. Moreover, for any other k-th fuzzy set, we have that

$$\begin{cases} k < h \Rightarrow \omega_k < \omega_h \wedge \forall x \in I_h: x \ge \omega_h = \dfrac{t_h + t_{h-1}}{2} > t_k \Rightarrow \forall x \in I_h: \mu_{A_k^i}(x) < \mu_{A_k^i}(t_k) = \varepsilon, \\ k > h + 1 \Rightarrow \omega_k > \omega_{h+1} \wedge \forall x \in I_h: x < t_{k-1} \Rightarrow \forall x \in I_h: \mu_{A_k^i}(x) < \mu_{A_k^i}(t_{k-1}) = \varepsilon. \end{cases}$$

Generalizing the property, we have $\forall h \neq k \forall x \in I_h$: $\sup_{x \in I_h} \min\{\mu_{A_h^i}(x), \mu_{A_k^i}(x)\} \le \varepsilon$. This statement can be easily generalized to the entire input space X_i, so $\forall h \neq k$: $\mathrm{Poss}(A_h^i, A_k^i) \le \varepsilon$ and the disjunction property is satisfied. □

Appendix C

In order to calculate the derivatives $\partial\mu_{A_k^i}/\partial t_{hi}$, several cases must be considered:

Case 1: $h = k = 1$

$$\frac{\partial\mu_{A_1^i}}{\partial t_{1i}} = \frac{\partial\mu(x_i,m_i,\delta)}{\partial\delta}\left(-\frac{\partial\delta(a,t_{1i})}{\partial a} + \frac{\partial\delta(a,t_{1i})}{\partial t_{1i}}\right)\Bigg|_{\substack{\delta=\delta(2m_i-t_1,t_1)\\ a=2m_i-t_1}} = \frac{\mu_{A_1^i}(x_i)}{\sqrt{-2\ln\varepsilon}}\cdot\frac{(x_i-m_i)^2}{\delta(2m_i-t_{1i})^3}. \tag{C.1}$$

Case 2: $1 < h \le K_i - 1,\ k = h$

$$\frac{\partial\mu_{A_k^i}}{\partial t_{hi}} = \frac{\partial\mu_{A_h^i}}{\partial t_{hi}} = \left[\frac{\partial\mu(x_i,\gamma,\delta)}{\delta\gamma}\frac{\partial\gamma(t_{h-1,i},t_{hi})}{\partial t_{hi}} + \frac{\partial\mu(x_i,\gamma,\delta)}{\partial\delta}\frac{\partial\delta(t_{h-1,i},t_{hi})}{\partial t_{hi}}\right]_{\substack{\gamma=\gamma(t_{h-1,i},t_{hi})\\ \delta=\delta(t_{h-1,i},t_{hi})}} = \mu_{A_h^i}(x_i)\frac{x-\gamma(t_{h-1,i},t_{hi})}{\delta(t_{h-1,i},t_{hi})^2}\cdot\frac{1}{2} + \mu_{A_h^i}(x_i)\frac{(x-\gamma(t_{h-1,i},t_{hi}))^2}{\delta(t_{h-1,i},t_{hi})^3}\frac{1}{2\sqrt{-2\ln\varepsilon}}. \tag{C.2}$$

Case 3: $1 \le h < K_i - 1,\ k = h+1$

$$\frac{\partial\mu_{A_k^i}}{\partial t_{hi}} = \frac{\partial\mu_{A_{h+1}^i}}{\partial t_{hi}} = \left[\frac{\partial\mu(x_i,\gamma,\delta)}{\partial\gamma}\frac{\partial\gamma(t_{h,i},t_{h+1,i})}{\partial t_{hi}} + \frac{\partial\mu(x_i,\gamma,\delta)}{\partial\delta}\frac{\partial\delta(t_{h,i},t_{h+1,i})}{\partial t_{hi}}\right]_{\substack{\gamma=\gamma(t_{h,i},t_{h+1,i})\\ \delta=\delta(t_{h,i},t_{h+1,i})}} = \mu_{A_{h+1}^i}(x_i)\frac{x-\gamma(t_{h,i},t_{h+1,i})}{\delta(t_{h,i},t_{h+1,i})^2}\cdot\frac{1}{2} - \mu_{A_{h+1}^i}(x_i)\frac{(x-\gamma(t_{h,i},t_{h+1,i}))^2}{\delta(t_{h,i},t_{h+1,i})^3}\frac{1}{2\sqrt{-2\ln\varepsilon}}. \tag{C.3}$$

Case 4: $h = K_i - 1,\ k = K_i$

$$\frac{\partial\mu_{A_k^i}}{\partial t_{hi}} = \frac{\partial\mu_{A_{K_i}^i}}{\partial t_{K_i-1,i}} = \frac{\partial\mu(x,M_i,\delta)}{\partial\delta}\left(\frac{\partial\delta(t_{K_i-1},b)}{\partial t_{K_i-1}} + \frac{\partial\delta(t_{K_i-1},b)}{\partial b}\frac{\partial b}{\partial t_{K_i-1}}\right)\Bigg|_{\substack{\delta=\delta(t_{K_i-1},2M_i-t_{K_i-1})\\ b=2M_i-t_{K_i-1}}} = \mu_{A_{K_i}^i}\frac{(x-M_i)^2}{\delta(t_{K_i-1},2M_i-t_{K_i-1})^3}\left(-\frac{1}{\sqrt{-2\ln\varepsilon}}\right). \tag{C.4}$$

For efficiency pursuits, such formulas can be rearranged in the back-propagation algorithm in order to avoid calculation redundancies.

References

Abe, S., & Lan, M. S. (1995). A method for fuzzy rule extraction directly from numerical data and its application to pattern classification. *IEEE Trans. on Fuzzy Systems., 3*, 18–28.

Bengio, Y. (2000). Gradient-based optimization of hyper-parameters. *Neural Computation*, 12(8).

Bersini, H., & Bontempi, G. (1997). Now comes the time to defuzzify neuro-fuzzy models. *Fuzzy Sets and Systems*, *90*, 161–169.

Bezdek, J. C. (1981). *Pattern recognition with fuzzy objective function algorithms*. New York: Plenum.

Brown, M., & Harris, C. J. (1994). *Neurofuzzy adaptive modeling and control*. Hemel Hempstead: Prentice Hall.

Castellano, G., & Fanelli, A. M. (1996). Simplifying a neuro-fuzzy model. *Neural Processing Letters*, *4*(6), 75–81.

Castellano, G., & Fanelli, A. M. (2000a). Fuzzy classifiers acquired from data. In Mohammadian, M. (Ed.), *New frontiers in computational intelligence and its applications*. IOS Press, pp. 31–41.

Castellano, G., & Fanelli, A. M. (2000b). Variable selection using neural network models. *Neurocomputing*, *31*(14), 1–13.

Castellano, G., Fanelli, A.M., & Mencar, C. (2000). A new empirical risk functional for a neuro-fuzzy classifier. *Proceedings of ESIT 2000*, Aachen, pp. 429–436.

Chow, M. -Y., Altug, S., & Trussell, H. J. (1999a). Heuristic constraints enforcement for training of an knowledge extraction from a fuzzy/neural architecture—Part I: Foundation. *IEEE Trans. on Fuzzy Systems*, *7*(2), 143–150.

Chow, M. -Y., Altug, S., & Trussell, H. J. (1999b). Heuristic constraints enforcement for training of an knowledge extraction from a fuzzy/neural architecture—Part II: Implementation and application. *IEEE Trans. on Fuzzy Systems*, *7*(2), 151–159.

Chung, F.-L., & Duan, J.-C. (2000). On multistage fuzzy neural network modeling. *IEEE Trans. on Fuzzy Systems*, 8(2).

Cios, K. J., Pedrycz, W., & Swiniarski, R. W. (1998). *Data mining*.

Methods for knowledge discovery. Kluwer Academic Press.

Craven, P., & Wabba, G. (1979). Smoothing noisy data with spline functions: Estimating the correct degree of smoothing by the method of generalized cross-validation. *Numerische Mathematik*, *31*, 377–403.

de Oliveira, J. V. (1999). Towards neuro-linguistic modeling: constraints for optimization of membership functions. *Fuzzy Sets and Systems*, *106*, 357–380.

Dubois, D., & Prade, H. (1980). *Fuzzy sets and systems: theory and applications*. New York: Academic Press.

Duch, W., Adamczak, R., & Grabczewski, K. (2001). A new methodology of extraction, optimization and application of crisp and fuzzy logical rules. *IEEE Transactions on Neural Networks*, *12*(2), 277–306.

Halgamuge, S. K., & Glesner, M. (1994). Neural network in designing fuzzy systems for real-world applications. *Fuzzy Sets and Systems*, *65*, 1–12.

Huang, Y. P., & Chu, H. C. (1999). Simplifying fuzzy modeling by both gray relational analysis and data transformation methods. *Fuzzy Sets and Systems*, *104*, 183–197.

Jang, J.-S.R., & Sun, C.-T., (1995). Neuro-fuzzy modeling and control. *Proc. of the IEEE*, 83(3).

Jang, J. -S. R. (1993). ANFIS: Adaptive-network-based fuzzy inference system. *IEEE Trans. on Systems, Man and Cybernetics*, *23*(3), 665–685.

Jankowski N., & Kadirkamanathan, V. (1997). Statistical control of RBF-like networks for classification. *Proc. 7th Int. Conf. Artificial Neural Networks*, Lausanne, Switzerland, 1997, pp. 385–390.

Jin, Y.C., Von Seelen, W., & Sendhoff, B. (1998). An approach to rule-based knowledge extraction. *Proc. IEEE Int. Conf. Fuzzy Syst*., Anchorage, AK, pp. 1188–1193.

Jin, Y., Von Seelen, W., & Sendhoff, B. (1999). On generating FC^3 Fuzzy rule systems from data using evolution strategies. *IEEE Trans. on Systems, Man and Cybernetics—Part B*, *29*(6), 829–845.

Jin, Y., Von Seelen, W., & Sendhoff, B. (2000). Fuzzy modeling of high-dimensional systems: complexity reduction and interpretability improvement. *IEEE Trans. on Fuzzy Systems*, *8*(2), 212–221.

Lin, C. -T., & Lee, C. S. G. (1991). Neural-network based fuzzy logic control and decision system. *IEEE Trans. on Computers*, *40*(12), 1320–1336.

Lofti, A., Handerson, H. C., & Toi, A. C. (1996). Interpretation preservation of adaptive fuzzy inference systems. *Int. J. Approxim. Reason*., 15(4).

Lozowski, A., & Zurada, J. M. (2000). Extraction of linguistic rules from data via neural networks and fuzzy approximation. In Cloete, J., & Zurada, J. M. (Eds.), *Knowledge-based neurocomputing*. Cambridge, Massachusetts: The MIT Press.

Marin-Blazquez, J. G., Shen, Q., & Gomez-Skarmeta, A. F. (2000). From approximative to descriptive models. *Proc. of FUZZ-IEEE*, San Antonio, Texas.

Narendra, K. S., & Parthasarathy, K. (1990). Identification and control of dynamical systems using neural networks. *IEEE Trans. on Neural Networks*, *1*, 4–27.

Nauch, D., Klawonn, F., & Kruse, R. (1997). *Foundations of neuro-fuzzy systems*. Chichester: Wiley.

Nauck, D., Nauck, U., & Kruse, R. (1996). Generating classification rules with the neuro-fuzzy system NEFCLASS. *Proc. Biennal Conf. North Amer. Fuzzy Inform. Processing Soc (NAFIPS '96)*, Berkeley, CA.

Nauck, D. (1995). Beyond neuro-fuzzy: perspectives and directions. *Proc. of EUFIT '95*, Aachen, pp. 1159–1164.

Nauck, D., & Kruse, R. (1997). A neuro-fuzzy method to learn fuzzy classification rules from data. *Fuzzy Sets and Systems*, *89*, 277–288.

Nauck, D., & Kruse, R. (1999). Obtaining interpretable fuzzy classification rules from medical data. *Artificial Intelligence in Medicine*, *16*, 149–169.

Pedrycz, W., & Gomide, F. (1998). *An introduction to fuzzy sets. Analysis and design*. The MIT Press.

Ross, T. J. (1997). *Fuzzy logic with engineering applications*. McGraw Hill Inc.

Setnes, M., Babuska, R., & Verbuggen, H. B. (1998a). Rule-based modelling: precision and transparency. *IEEE Trans. on Systems, Man and Cybernetics—Part C: Appl. Rev*., *28*(1), 165–169.

Setnes, M., Babuska, R., Kaymak, U., & van Nauta Lemke, H. R. (1998b). Similarity measures in fuzzy rule base simplification. *IEEE Trans. on Systems, Man and Cybernetics—Part B*, *28*, 376–386.

Shand, N., & Breiman, M. (1996). Distribution based trees are more accurate. *Proc. Int. Conf. Neural Inform. Processing*, Vol. 1, Hong Kong, pp. 133–138.

Ster, B., & Dobnikar, A. (1996). Neural networks in medical diagnosis: comparison with other methods. In A. Bulsari (Ed.), *Proc. Int. Conf. EANN '96*, pp. 427–430.

Sugeno, M., & Kang, G. T. (1988). Structure identification of fuzzy model. *Fuzzy Sets and Systems*, *28*, 15–33.

Sugeno, M., & Yasukawa, T. (1993). A fuzzy-logic-based approach to qualitative modeling. *IEEE Trans. on Fuzzy Systems*, *1*, 7–31.

Takagi, T., & Sugeno, M. (1985). Fuzzy identification of systems and its applications to modeling and control. *IEEE Trans. on Systems, Man and Cybernetics*, *15*, 195–214.

Wang, L.-X. (1999). Analysis and design of hierarchical fuzzy systems. *IEEE Trans. on Fuzzy Systems*, 7(5).

Wolberg, W., & Mangasarian, O. (1990). Multisurface method of pattern separation for medical diagnosis applied to breast cytology. *Proc. Natl. Acad. Sci*., *87*, 9193–9196.

Zadeh, L. A. (1978). Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, *1*, 3–28.

Zurada, J.M., & Lozowski, A. (1996). Generating linguisitic rules from data using neuro-fuzzy framework. *Proc. of the fourth International Conference on Soft Computing (IIZUKA '96)*, Iizuka, Fukuoda, Japan, pp. 618–621.