ESANN'2007 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2007, d-side publi., ISBN 2-930307-07-2.

How to process uncertainty in machine learning?

Barbara Hammer [1] and Thomas Villmann[2]

1- Clausthal University of Technology, Institute of Computer Science,
Clausthal-Zellerfeld, Germany

2- University of Leipzig, Medical Department, Leipzig, Germany

Abstract. Uncertainty is a popular phenomenon in machine learning and a variety of methods to model uncertainty at different levels has been developed. The aim of this paper is to motivate the merits and problems when dealing with uncertainty in machine learning and to give an overview about methodologies which fall under the framework of neuro-fuzzy methods, in particular fuzzy-clustering on the one side and fuzzy inference systems on the other side.

1 Areas dealing with uncertainty

Uncertainty and fuzziness are popular phenomena in many application areas such as medicine (medical diagnosis is often not crisp but there exist various degrees of illness e.g. for psychical diseases such as phobia), image processing (areas at object borders or at overlapping regions can seldom uniquely be classified), linguistics (terms such as 'high' or 'small' are context dependent), etc. Therefore, uncertainty almost automatically occurs in any application of machine learning.

Different types of uncertainty can be observed: (i) Input data are subject to noise, outliers, and errors. A machine learning method has to deal with this type of fuzzy information, showing robustness with respect to such disturbances. Thereby, input noise can have a positive effect on the generalization behavior of the machine learning method since the method is forced to develop some form of invariance and to abstract from the noise. (ii) Output decisions should be accompanied by a measure which allows to judge the certainty or belief of the output. This is particularly important in critical domains such as clinical diagnosis, in safety critical areas, or in semiautomatic systems where human expert knowledge is accompanied by automatic inference. (iii) Representation of information within a machine learning system is distributed and fuzzy. This is the standard situation for classical neural network models and it is difficult to assign a crisp meaning to specific parts of a neural model. For a deeper insight into the network behavior, an interpretation of the fuzzy information in the internal representation is required.

According to these different locations and goals of fuzzy information, a variety of different models exist which allow machine learning to deal with insecure information as input, output, or internal representation.

1.1 Fuzzy-representations

Fuzzy and probability

Uncertainty is closely connected to probability, which (directly or indirectly) constitutes the formal framework for machine learning and neural network models. Specifying the probability of an event also gives information about how certain we are, that this event occurs. However, for probability theory, we need to specify the probability of all events and applying Bayes rule of inference requires a prior, which is often not known and therefore chosen based on general principals e.g. as uniform distribution.

The Dempster-Shafer theory of evidence [11, 43] constitutes a generalization of probability theory which allows a formalization of belief, evidence and possibility. In this theory, it is possible to assign a mass (or probability) $m(S)$ to every power set of a domain X, expressing the evidence that an element of $S \subset X$ occurs, but the same is not believed for a proper subset of S. One can think of such a universe as a set-valued stochastic variable, called a possibilistic set. The belief (or necessity) of a set S is given by the sum of the masses of all subsets of S, whereas the plausibility (or possibility) of S is given by the difference of 1 and the sum of all sets which have a zero intersection with S. These two quantities span an interval which indicates limits for the probability of S: the belief gives the minimum evidence supporting S, whereas the plausibility gives the maximum possible value which is not contradicted by alternative evidence.

Fuzzy sets and operations

The Dempster-Shafer-theory of evidence can serve as a formal background for fuzzy sets. A fuzzy set A over the domain X is given by its characteristic function $\chi_A : X \to [0, 1]$, which can possess degrees of fulfillment in the whole interval from 0 to 1 in contrast to crisp sets which characteristic function maps to $\{0, 1\}$. Every possibilistic set induces a fuzzy set by assigning $X \ni x \mapsto \sum_{x \in S} m(S)$, the sum of masses of sets which contain x. Unlike a probability distribution, this can be any function with codomain in $[0, 1]$, it need not be normalized.

It is possible to extend standard operations of sets to fuzzy sets such that crisp values are preserved [23, 25]. The intersection is modeled by a t-norm which is an operator $\top : [0, 1]^2 \to [0, 1]$ which is associative, commutative, monotonic in every argument, and which fulfills $\top(x, 1) = x$ for every x, e.g. the classical operator min as proposed by Zadeh. The dual, the union, is given by a t-conorm, which is associative, commutative, monotonic in every argument, and which fulfills $\top(x, 0) = x$ for every x, e.g. the classical operator max as proposed by Zadeh. The complement of a set is usually realized by the function $1 - x$. These choices, however, necessarily yield to extensions of crisp set operations for which no longer all classical laws hold such as the law of contradiction ($A \cap A^C = \emptyset$) or the law of excluded third ($A \cup A^C = X$), which are usually not fulfilled for fuzzy sets and reasonable choices of the t-norm and t-conorm.

Fuzzy logic

Similar to these extensions of set operations, one can extend crisp 0/1-valued logic to fuzzy values [23, 25]. The logical 'and' is realized by a t-norm, the 'or' by a t-conorm, negation by the function $1 - x$. For the implication, different reasonable choices have been proposed. The S-implication identifies $a \to b$ and $\neg a \vee b$. R-implication relies on the observation that a and $a \to b$ yield b, thus, if one knows the degree of fulfillment of a and b, one has to built some form of inverse of the and-operator to obtain the degree of fulfillment of $a \to b$. This is achieved by the so-called residuum of the t-norm. The R-implication is defined as the residuum of a and b with respect to a t-norm. The T-implication uses a t-norm to evaluate the implication. Note that this is no longer an extension of the classical implication, however, it is often used in fuzzy inference and fuzzy control.

These basic operations can be extended to fuzzy inference systems in different ways. There exists a variety of fuzzy logic calculi (usually restricted to boolean logic) [23, 25]. Some of these calculi possess an underlying semantic and it can be shown that they are correct and complete with respect to this semantic, such as fuzzy resolution or possibilistic inference. Thereby, depending on the calculus, valid models can be obtained by repeated inference as a fixed point of the operator which represents logical inference.

One of the most popular inference mechanisms is the compositional rule of inference: This is not accompanied by an underlying semantic. Rather, the rule is applied only once to map a fuzzy set over a domain X to a fuzzy set over a domain Y by means of an inference rule with precedent concerning X and antecedent concerning Y. More precisely, given a fuzzy set A over X and an inference rule $A' \to B'$, it is possible to infer B on Y with $\chi_B(y) = \sup_x \top\{\chi_A(x), \chi_{A' \to B'}(x, y)\}$ where $\chi_{A' \to B'}$ corresponds to the fuzzy set which describes the inference and $\top$ is a t-norm. This allows to transfer a fuzzy set on X which is similar to the precedent A' to a conclusion over Y. The compositional rule of inference is often applied for control or function approximation to map input values in X to corresponding outputs in Y by means of if-then rules.

Fuzzy arithmetic

A fuzzy number fuzzifies a real number b. Often, triangle sets (a, b, c) which yield 0 outside the interval (a, c), and which consist of two linear pieces which map b to 1, or generalizations thereof which substitute the two linear pieces by a monotonically increasing resp. decreasing function are considered. It is possible to extend any function on crisp sets to fuzzy sets by means of the so-called extension principle [23, 25]. Given a function $f : X \to Y$, the extension $\hat{f}$ maps fuzzy sets A over X to sets B over Y by means of the equality $\chi_B(y) = \sup\{\chi_A(x) \mid f(x) = y\}$. This way, simple arithmetic operations, but also more complex functions can be transferred to fuzzy sets.

Usually, it is complicated to compute the extension of a function for a given concrete scenario unless X is finite, such that the value $\chi_{\hat{f}(A)}(y)$ can be determined by extensive search over X. For the general situation, the alternative representation of fuzzy sets via so-called (strict) α-cuts allows to reduce the computation of the extension to interval arithmetic. The strict α-cut of a fuzzy set A is given by $[A]_\alpha = \{x \mid \chi_A(x) > \alpha\}$. For smooth fuzzy functions over the real numbers, this is an interval resp. a union of intervals. It is possible to recover the underlying fuzzy set by the equality $\chi_A(x) = \sup\{\alpha \mid x \in [A]_\alpha\}$. It holds $[\hat{f}(A)]_\alpha = f([A]_\alpha)$ for $\alpha > 0$ and every function f, therefore strict α-cuts constitute a convenient way to compute the extension of a function f.

Fuzzy information

Which type of fuzzy information is available in machine learning? Naturally, a variety of probabilistic frameworks exist such as Bayesian belief networks for causal reasoning [39]. These approaches represent a specific type of fuzzy information according to the laws of probability theory. The background in statistics and probability yields a very powerful theory, at the same time, often quite complex inference and complex specifications are necessary. Depending on the setting, it offers an elegant general framework to model complex information. As an example, the approach presented in [15] proposes a methodology to visualize and data mine tree structures by means of an extension of self-organizing maps to a generative probabilistic model. Thereby, each lattice point generates a mode of the data, whereby the generation process (the noise model in the original generative topographic mapping [5]) is given by hidden Markov tree models. This point of view offers a general and convenient interface to complex data structures.

For classification models in machine learning such as the support vector machine or feedforward networks, there is often a real value available which is mapped to a crisp output class by means of a threshold, such as the activation of the output neuron. This can be transferred to a probability of the classification after possible normalization. Thus, probability or fuzzy information of clustering can directly be obtained in these cases. Some models extend this process towards a richer fuzzy information such as the belief and the possibility. The approach [16] trains feedforward neural networks directly for a given classification task, whereby the activation of the network is chosen as possibility. Belief values are obtained thereof by means of the λ-complementation $\chi_A(x) = (1 - \chi_{A^c}(x))/(1 + \lambda \cdot \chi_{A^c}(x))$. The value λ influences the size of the outcome, and it is determined based on the number of misclassification errors for the classes. These two values give additional information in the form of an upper and lower bound of the class probability. Training, however, requires only crisp class information. Belief and possibility are obtained as additional output information based on the geometric form and the number of misclassifications of the trained model.

Some models explicitely deal with fuzzy class labels provided within the input set such as [48, 49, 50]. This opens the possibility to individually model the belief of the respective situation independent of assumptions posed by the algorithm. The question occurs how fuzzy information can be gained in practice. For this purpose, specific interfaces have to be designed which allow humans to easily specify the amount of fuzziness of a decision. The approach [51] deals with fuzzy information within image segmentation. Thereby, an interactive editor has been developed which allows humans to specify the borders of crisp segments as well as the borders of fuzzy segments, whereby automatic interpolation is done in between. In [33], it is tested in how far the belief of a human decision can be detected from features generated by an eye tracking system. This study opens the way towards the automatic generation of degrees of fuzziness by various human-computer interfaces.

The question of how fuzzy information learned by a computer system is represented is closely connected to this topic. A fuzzy classification with only two classes can easily be represented, but the situation is more complex for a large number of classes. Here, the classes as well as the respective fuzzy value have to be represented in such a way that they can be distinguished by a human. The contribution [51] deals with the case that fuzzy class values are assigned to every pixel within a classified image. Since it is hard to find an unbiased and easy to distinguish color for more than a few classes, the approach proposes an embedding of the fuzzy vectors into the RGB space by means of multidimensional scaling.

1.2 Applications

Applications of neuro-fuzzy systems of various form are widespread. One important area of application concerns causal reasoning, where fuzzy inference systems are used. This way, decision support can be applied e.g. in medicine, stock trading, or credit risk evaluation [7, 54]. A fuzzy inference system can be interpreted as function approximation mapping input assignments to output assignments, therefore, fuzzy inference is also applied to scientific explanation, natural language understanding, isolate faulty components in electronics, infer dynamical systems, etc. [7]. Fuzzy inference is closely connected to fuzzy logic control which builds a closed loop control around a control signal which is realized by means of a fuzzy inference. Fuzzy control has been applied in a variety of settings including among industrial applications toy examples such as the inverted pendulum [53] or control of scientific algorithms such as a computational fluid dynamics algorithm [42]. Since neuro-fuzzy control constitutes an important model for real-time applications various hardware realizations have been proposed such as [2].

Another important class of neuro-fuzzy models is fuzzy clustering. This has been applied for all kinds of pattern recognition tasks such as robotics space representation [1] or face recognition [16, 31, 30].

2 Models dealing with uncertainty

The term 'neuro-fuzzy model' covers diverse approaches which differ in the type of fuzzy information within the network as well as the kind of the fuzzy theory which is integrated in the network. Neural network models constitute very powerful mechanisms with good learning and generalization issues, however, most models have a black box character and it is hard to assign a meaning to single constituents of a network. On the contrary, fuzzy systems assign a meaning to single fuzzy values and are easy to understand by humans, but it is difficult to train fuzzy models. Thus, the properties of fuzzy systems and neural networks are complementary whereby they share the basic units, real values. Therefore, a combination of both approaches such that efficient adaptability is combined with good interpretability seems promising.

One can identify two main directions of neuro-fuzzy systems: models which are close to pattern recognition methods and which can be centered around fuzzy clustering on the one hand and models which rely on fuzzy logic inference, including neuro-fuzzy control as a prime example on the other hand. Besides, a variety of further proposals exist. These include, for example, a fuzzification of neural network functions by means of the extension principle such that functions on fuzzy sets arise. This has been applied to RBF networks in [56] where the universal approximation property of these models is also discussed. In [30] a fuzzy support vector machine which can better deal with outliers and noise is proposed. Basically, the slack variable ξ_i of pattern i is multiplied by a fuzzifier which accounts for a less sensitivity with respect to points with small fuzzy values. The fuzziness is thereby determined during training based on geometric principles. The approach [21] yields better results by computing the fuzzifier based on distances in the feature space.

2.1 Clustering and pattern recognition

Basic fuzzy clustering

The basic fuzzy-c-means clustering algorithm was introduced by Bezdek as a generalization of standard vector quantization resp. crisp c-means clustering as follows [4]: given data points $\boldsymbol{x}^i$, c clusters are defined by prototypes $\boldsymbol{w}^j$ which represent the cluster centers. Data points can be assigned to the respective closest prototype, or, alternatively, a fuzzy value for the assignment to cluster j can be based on the distance: $\chi_j(\boldsymbol{x}^i) = \|\boldsymbol{x}^i - \boldsymbol{d}^j\|^{-1/(d-1)} / \sum_k \|\boldsymbol{x}^i - \boldsymbol{d}^k\|^{-1/(d-1)}$. Cluster centers fulfill the equation $\boldsymbol{w}^j = \sum_i \chi_j(\boldsymbol{x}^i)^d \boldsymbol{x}^i / \sum_i \chi_j(\boldsymbol{x}^i)^d$. $d > 1$ constitutes the fuzzifier, which is usually chosen as 2. Fuzzy-c-means learning determines weights and assignments based on given training data by iterative optimization of these values, using the above formulas. These formulas can be derived from the cost function which is minimized by fuzzy-c-means: $\sum_{i,j} \chi_j(\boldsymbol{x}^i)^d \|\boldsymbol{x}^i - \boldsymbol{w}^j\|^2$ with constraint $\sum_j \chi_j(\boldsymbol{x}^i) = 1$. In an EM like scheme, fuzzy assignments are optimized for fixed weights and vice versa.

There exist quite a few publications which consider the convergence property of this algorithm, such as [17] and references therein. This basic algorithm has been widely applied e.g. in [31] for face recognition, whereby the fuzziness of the trained classification constitutes an important feature for further processing which preserves more information than a crisp clustering.

Extensions

Basic fuzzy-c-means has been extended in several ways: the algorithm itself aims at probabilistic clustering in the sense that the sum of fuzzy membership functions taken over the prototypes adds up to 1 for every pattern. Thus, outliers are not identified but assigned to the clusters. Possibilistic clustering weakens the restriction $\sum_j \chi_j(\boldsymbol{x}^i) = 1$ and substitutes it by a different regularization term. The algorithm of Krishnapuram/Keller adds a term which punishes the deviation of the fuzzy assignments from 1. However, this only yields a local valid cost term since identical prototype locations are not prevented.

Davé proposes an alternative which explicitely introduces a 'cluster of outliers' with fixed quantization costs [10]. In [36] possibilistic clustering as a controlled mixture of the proposal of Krishnapuram/Keller and the basic algorithm is investigated.

Further variants concern more general metrics such as a full adaptive matrix [14, 18] or more general cluster structures such as varieties or principle components [19]. In addition, several extensions of basic-c-means deal with specific data types such as time series [12] or efficient implementations such as hierarchical versions [20].

Fuzzy neural clustering

Variants which extend crisp assignments to fuzzy memberships based on the distance of data points and prototypes have also been proposed for classical neural clustering variants such as fuzzy learning vector quantization, which is a fuzzification of online (unsupervised) vector quantization [9] or fuzzy self organizing maps which extend the popular self organizing map by fuzzy assignments [3]. Also adaptive resonance networks have been extended to fuzzy memberships. The comparison of data and prototypes is done using t-norm and t-conorm, such that rectangular shapes of the clusters result [46]. Fuzzy art has been applied to, among others, robot navigation [1], robot control [8], and medical image classification [47].

Fuzzy labeled neural clustering

A different extension of classical neural clustering algorithms has been proposed in [48, 49, 50]. Here the self-organizing map (SOM) resp. neural gas networks (NG) are extended such that they can incorporate and learn fuzzy label information. For SOM, this allows a visualization of data statistics with respect to prior label information, such that the important aspects of data are accelerated for convenient human inspection. Both approaches, supervised fuzzy neural gas and supervised fuzzy self organizing map are based on an extension of the cost function of the formalism (thereby considering the self-organizing map in the variant of Heskes) by an additional term which compares the label information of data and prototypes. Thus, a very general and uniform approach is obtained this way.

Hence, a rich variety of neuro-fuzzy clustering mechanisms exists. Most of these algorithms constitute prototype-based models which are derived from a cost function related to the quantization error which is optimized by clustering.

2.2 Symbolic approaches

In contract to these pattern-based approaches, symbolic approaches assign a meaning to neurons by means of an identification with fuzzy-logic principles. Basic constituents are neurons which implement a specific t-norm, t-conorm, or alternative logic primitives. Thereby, smooth parameterization or generalization e.g. in the form of aggregation has also been proposed [38].

Neuro-fuzzy control

Neuro-fuzzy control implements classical fuzzy control by means of a network topology. The two control systems which are mostly considered are the Mamdani control and the Takagi/Sugeno control. Both controllers are described by if-then rules of the form: if x_1 is A_1 and x_2 is A_2 and ... then y is B, whereby x_i are input variables (observations) and y is the output (control signal). A_i are fuzzy sets. For Mamdani control, B is also a fuzzy set, whereas it is given by an explicit real-valued function $f(x_1, x_2, \ldots)$ for the Takagi/Sugeno controller. Given a signal, the control signal is computed by an application of the compositional rule of inference followed by defuzzification for the Mamdani controller. For both controllers, the degree of fulfillment of the precedent is computed as $\alpha = \top\{\chi_{A_1}(x_1), \chi_{A_2}(x_2), \ldots\}$, $\top$ being a t-norm. This yields the fuzzy output $\top\{\alpha, \chi_B(y)\}$ for the Mamdani controller, which is aggregated and defuzzified by means of the center of gravity. For Takagi/Sugeno control, α serves as a weighting factor for the output function $f(x_1, x_2, \ldots)$ and these values are aggregated for all fuzzy rules.

Note that this principle leads to one acyclic application of the fuzzy inference system resp. neural network representing the fuzzy system to obtain the control signal. Besides control, this method can also be used for function approximation, mapping inputs x to desired outputs y. It has been shown for a variety of fuzzy controllers that they constitute universal approximators [32].

One of the most popular neuro-fuzzy networks is the ANFIS model [40] which directly implements a Takagi/Sugeno control. It uses RBF neurons in the first layer to implement the fuzzy sets A_i, the product t-norm, and a number of linear layers to compute the output function, summation and averaging. Overall, the network function is differentiable such that training can be done using standard backpropagation. An alternative, NEFCON, implements a form of Mamdani control [34]. Thereby, connections are assigned fuzzy sets and neurons compute t-norms resp. aggregation and defuzzification. That means, no standard neurons are used in this model and depending on the choice of these generalized connections and neurons, the function is not necessarily differentiable with respect to the weights. Correspondingly, the training algorithm relies on heuristics.

For control, the property of stability is essential to guarantee correct behavior of the model. A variety of methods to guarantee stability for Takagi/Sugeno control has been developed, mostly based on some Lyapunov function, see [13] for an overview.

Variants

The question occurs how parameters and rules can automatically be obtained for neuro-fuzzy models. Training a fixed architecture is done by standard backpropagation (if the function is differentiable), Hebbian learning, or alternatives such as SVM like methods which include regularization [28]. The network structure, i.e. the rules can be obtained by metaheuristics such as genetic algorithms [27, 35] or appropriate initialization based on given data such as fuzzy clustering [37] or Bayesian inference [22]. Note that these methods can also directly be used in data mining in order to extract fuzzy rules from given data.

Another Takagi/Sugeno approach is based on SOM and Kohonen's learning vector quantization (LVQ) [52], whereby the prototypes here describe fuzzy rules, which are composed of fuzzy sets. The fuzzy sets define an area in the input space, where each fuzzy rule fires. First, the SOM is trained to detect the fuzzy sets roughly. A subsequent modified LVQ fine tunes the fuzzy sets and, parallely optimizes the output for each rule.

Neural networks have not only been applied to learn the control but also to approximate the underlying process which has to be controlled if this is not available analytically, such as in the case of a permanent magnet linear synchronous motor [29]. Interestingly, these type neuro-fuzzy networks can be applied for function approximation and, thus, also clustering as proposed in [26], so-called min-max clustering. The approach [44] proposes a transductive variant of the ANFIS model which considers only the k-nearest neighbors for the output of a data point. This has the benefit that a very simple control consisting of only one rule and a linear output function is sufficient for these local patches. Further variants such as hierarchical neuro-fuzzy control [6] or recurrent neuro-fuzzy networks which also include network outputs through time delay units [57] have been proposed.

Fuzzy associative memory

Naturally, one step logical inference as occurs in fuzzy control can also directly be used for decision support and fuzzy inference, as presented e.g. in [55], where rules are optimized by means of particle swarm optimization. Fuzzy associative memories are very similar with respect to the function [45]. These models associate a given input x to an output y by means of fuzzy-logical operations. They combine neurons which implement t-norm resp. t-conorm as a realization of logical 'and' and 'or'. In [45] the logical implication is realized as R-implication which guides learning. Interestingly, convergence of association can be guaranteed also when iterating the process due to the monotonicity of the involved fuzzy-logical operators t-norm and t-conorm.

Fuzzy inference systems

Neuro-fuzzy logic reasoning directly transfers fuzzy logic concepts into a neural network model, the activation of single neurons encoding degrees of fulfillment of fuzzy logic terms. The difference to neuro-fuzzy control and simple inference consists in the fact, that complex reasoning systems include iterative or cyclic processes which mimic the iterated or recursive application of inference steps. Thereby, non-classical effects such as additive positive interaction of evidence for a given fact can easily be modeled. The convergence of such reasoning systems either transfers from the convergence property of the underlying fuzzy logic system, or alternative guarantees have to be found. A popular possibility to guarantee convergence consists in the connection to an energy function which is minimized by consecutive logical inference steps such as proposed in [41].

Overall, a variety of neuro-fuzzy systems which are based on some form of symbolic fuzzy logic exists. Apart from their success in practice, these models can serve as an interface towards human understandable interpretation of neural networks.

3 Conclusions

We have presented a variety of formalisms in fuzzy logic and their combination with neural networks. These models differ in the type of fuzzy theory which is integrated into the neural system ranging from a direct fuzzification by means of the extension principle to fuzzy clustering and fuzzy control systems. These diverse directions indicate the widespread techniques which exist in this domain which open the way towards interesting industrial and scientific applications.

References

[1] R. Araujo, Prune-able fuzzy ART neural architecture for robot map learning and navigation in dynamic environments, *IEEE TNN* 17(5):1235-1249, 2006.

[2] K. Basterretxea, J.M. Tarela, I. del Campo, G. Bosque, An experimental study on nonlinear function computation for neural/fuzzy hardware design, *IEEE TNN* 17(5): 266-283, 2007.

[3] J.C. Bezdek, E.C.K. Tsao, N.R. Pal, Fuzzy Kohonen clustering networks, In Proc. IEEE Int. Conf. on Fuzzy Systems, pages 1035-1043, IEEE, 1992.

[4] J. Bezdek, S. Pal, *Fuzzy models for pattern recognition*, IEEE, Piscataway, NJ, 1992.

[5] C.M. Bishop, M.Svensen, C.K.I. Williams, GTM: The generative topographic mapping, *Neural Computation* 10:215-234, 1998.

[6] K. Cagatay Bayindir, M. Ugras Kumar, M Türmay, Hierarchical neuro-fuzzy current control for a shunt active power filter, *Neural Computing & Applications* 15(3-4): 223-238, 2006.

[7] E. Castillo, J.M. Guitierrez, A.S. Hadi, *Expert Systems and Probabilistic Neural Networks*, Springer, 1997.

[8] A. Chatterjee, K. Watanabe, An optimized Takagi-Sugeno type neuro-fuzzy system for modeling robot manipulators, *Neural Computing & Applications* 15(1): 55-61, 2006.

[9] F.-L. Chung, T. Lee, Fuzzy learning vector quantization, In Proc. IJCNN-93, volume III, pages 2739-2742, IEEE, 1993.

[10] R.N. Davé. Characterization and detection of noise in clustering. *PRL*, 12:657-664, 1991.

[11] A.P. Dempster, A generalization of Bayesian inference, *Journal of the Royal Statistical Society* B, 30:205-247, 1968.

[12] P. D'Urso, Fuzzy clustering for data time arrays with inlier and outlier time trajectories, *IEEE Tans. Fuzzy Syst.* 13(5):583-604, 2005.

[13] G. Feng, A survey on analysis and design of model-based fuzzy control systems, *IEEE Trans. Fuzzy Sys.* 14(5):676-697, 2006.

[14] I. Gath, A.B. Geva. Unsupervised Optimal Fuzzy Clustering. *TPAMI*, 11(7):773-781, 1989.

[15] N. Gianniotis, P. Tino, Visualisation of tree-structured data through generative probabilistic modelling, in this volume.

[16] K.E. Graves, R. Nagarajah, Uncertainty estimation using fuzzy measures for multiclass classification, *IEEE TNN* 18(1):128-140, 2007.

[17] L. Gröll, J. Jäkel, A new convergence proof of fuzzy-c-means, *IEEE Trans. Fuzzy Syst.* 13(5):717-720, 2005.

[18] D.E. Gustafson, W.C. Kessel. Fuzzy Clustering with a Fuzzy Covariance Matrix. In Proc. of the IEEE Conference on Decision and Control, pages 761–766, 1979.

[19] K. Honda, H. Ichihashi, Regularized linear fuzzy clustering and probabilistic PCA mixture models, *IEEE Trans. Fuzzy Syst.* 13(4):508-516, 2005.

[20] Y.-J. Horng, S.-M. Cheng, Y.-C. Chang, C.-H. Lee, A new method for fuzzy information retrieval based on fuzzy hierarchical clustering and fuzzy inference techniques, *IEEE Trans. Fuzzy Syst.* 13(2):216-228, 2005.

[21] X. Jiang, Z. Yi, J.C. Lv, Fuzzy SVM with a new fuzzy membership function, *Neural Computing and Applications* 15(3-4):268-276, 2006.

[22] J. Kim, Y. Suga, S. Won, A new approach to fuzzy modeling and nonlinear dynamic systems with noise: relevance vector learning mechansism, *IEEE Trans. Fyzzy Syst.* 14(2): 222-231, 2006.

[23] G.J. Klir and B. Yuan, Fuzzy Sets and Fuzzy Logic - Theory and Applications, Prentice Hall, 1995.

[24] R. Krishnapuram, J. M. Keller. The Possibilistic c-Means Algorithm: Insights and Recommendations. *IEEE Trans.Fuzzy Sst.*, 4(3):385–393, 1996

[25] R. Kruse, J. Gebhardt, und F. Klawonn, Foundations of Fuzzy Systems Wiley, 1994.

[26] M. Kumar, R. Stoll, N. Stoll, A min-max approach to fuzzy clustering, estimation, and identification, *IEEE Trans. Fuzzy Syst.* 14(2): 248-262, 2006.

[27] G. Leng, T.M. McGinnity, G. Prasad, Design of self-organizing fuzzy neural networks based on genetic algorithms, IEEE Trans. Fuzzy Systems, 14(6):755-766, 2006.

[28] C.-T. Lin, C.-M. Yen, S.-F. Liang, J.-F. Chung, N. Kumar, Support-vector-based fuzzy neural network for pattern classification, *IEEE Trans. Fuzzy Syst.* 14(1):31-41, 2006.

[29] F.-J. Lin, P.-H- Shen, Adaptive fuzzy-neural-network control for a DSP-based permanent magnet linear synchronous motor servo drive, *IEEE Trans. Fuzzy Syst.* 14(4):481-495, 2006.

[30] Y.-H. Liu, Y.-T. Chen, Face recognition using total margin-based adaptive fuzzy support vector machines, *IEEE TNN* 18(1):178-192, 2007.

[31] J. Lu, X. Yuan, T. Yahagi, A method of face recognition based on fuzzy c-means clustering and associated sub-NNs, *IEEE TNN* 18(1): 150-160, 2007.

[32] S. Mitra, Y. Hayashi, Neuro-fuzzy rule generation: survey in soft computing framework, *IEEE TNN* 11(3):748-468, 2000.

[33] M. Nakayama, Y. Takahashi, An estimation of response certainty using features of eye-movements, in this volume.

[34] D. Nauck. Building Neural Fuzzy Controllers with NEFCON-I. In R. Kruse, J. Gebhardt und R. Palm, eds, Fuzzy Systems in Computer Science, Series Artificial intelligence, 141-151. Vieweg, 1994.

[35] S.-K. Oh, W. Pedrycz, H.-S. Park, Genetically optimized fuzzy polynomial neural networks, *IEEE Trans. Fuzzy Syst.* 14(1):125-144, 2006.

[36] N.R. Pal, K. Pal, J.M. Keller, J.C. Bezdek, A possibilistic fuzzy-c-means clustering algorithm, *IEEE Trans. Fuzzy Syst.* 13(4):517-530, 2005.

[37] M. Panella, A.S. Gallo, An input-output clustering approach to the synthesis of ANFIS networks, *IEEE Trans. Fuzzy Syst.* 13(1):69-81, 2005.

[38] W. Pedrycz, Logic-based fuzzy neurocomputing with unineurons, *IEEE Trans. Fuzzy Syst.* 14(6): 860-873, 2006.

[39] I. Rish et al., Adaptive diagnosis in distributed systems, *IEEE TNN* 16(5):1088-1109, 2005.

[40] J.-S. Roger Jang, ANFIS: Adaptive-network-based fuzzy inference system, *IEEE Trans. Syst., Man, and Cybernetics* 23:665-685, 1993.

[41] L.B. Romdhane, A softmin-based neural model for causal reasoning, *IEEE TNN* 17(3):732-744, 2006.

[42] J.Ryoo, Z. Dragojlovic, D.A. Kaminski, Control of convergence in a computational fluid dynamics simulation using ANFIS, *IEEE Trans. Fuzzy Syst.* 13(1):42-47, 2005.

[43] G. Shafer, A Mathematical Theory of Evidence, Princeton University Press, 1976.

[44] Q. Song, N.K. Kasabov, NFI: a neuro-fuzzy inference method for transductive reasoning, *IEEE Trans. Fuzzy Syst.* 13(6):799-808, 2005.

[45] P. Sussner, M.E. Valle, Implicative fuzzy associative memories, *IEEE Trans. Fuzzy Syst.* 14(6):793-807, 2006.

[46] M. Taghi, V. Baghmishesh, P. Nikola, A fast simplifies fuzzy ARTMAP network, *Neural Processing Letters* 17(3):273-316, 2003.

[47] B. Vigdor, B. Lerner, Accurate and fast off and inline fuzzy ARTMAP-based image classification with application to genetic abnormality diagnosis, *IEEE TNN* 17(5):1288-1300, 2006.

[48] T. Villmann, B. Hammer, F.-M. Schleif, T. Geweniger, W. Herrmann. Fuzzy classification by fuzzy labeled neural gas. *Neural Networks* 19:772-779, 2006.

[49] T. Villmann, F.-M. Schleif, B. Hammer. Prototype-based fuzzy classification with local relevance for proteomics. *Neurocomputing* 69(16-18):2425-2428, 2006.

[50] T. Villmann, U. Seiffert, F.-M. Schleif, C. Brüß, T.Geweniger, and B. Hammer. Fuzzy labeled self-organizing map with label-adjusted prototypes. In F. Schwenker and S. Marinai, eds., ANNPR 2006, LNAI 4087, pages 46-56. Springer, 2006.

[51] T. Villmann, M. Strickert, C. Brüß F.-M. Schleif, U. Seiffert, Visualiation of fuzzy information in fuzzy-clustering for image segmentation using MDS, in this volume.

[52] Petri Vuorimaa. Fuzzy self-organizing map. *Fuzzy Sets and Systems*, 66(2):223–231, 1994.

[53] R.-J. Wai, L.-J. Chang, Stabilizing and tracking control of nonlinear dual-axis inverted-pendulum system using fuzzy neural network, *IEEE Trans. Fuzzy Syst.* 14(1), 145-168, 2006.

[54] Y. Wang, S. Wang, K.K. Lai, A new fuzzy support vector machine to evaluate credit risk, *IEEE rans. Fuzzy Syst.* 13(6):820-831, 2005.

[55] B. Ye, C.Z.Zhu, Y.J.Cao, Adaptive extended fuzzy basis function network, *Neural Computing and Applications* 197-206, 2007.

[56] D. Zhang, L.-F. Deng, K.Y. Cai, A. So, Fuzzy nonlinear regression with fuzzified radial basis function networks, *IEEE Trans. Fuzzy Syst.* 13(6):742-760, 2005.

[57] J. Zhang, Modeling and optimal control of batch processes using resurrent neuro-fuzzy networks, *IEEE Trans. Fuzzy Syst.* 13(4):417-427, 2005.