NEURO–FUZZY CLASSIFICATION INITIALIZED BY FUZZY CLUSTERING

Detlef Nauck and Frank Klawonn
Department of Computer Science, Technical University of Braunschweig
Bueltenweg 74 – 75, D–38106 Braunschweig, Germany
Tel. +49.531.391–3155, Fax: +49.531.391–5936
E–Mail: D.Nauck@tu-bs.de, URL: www.cs.tu-bs.de/∼nauck

Abstract

In this paper we discuss how a neuro–fuzzy classifier can be initialized by rules generated by fuzzy clustering. The neuro–fuzzy classifier NEFCLASS can learn fuzzy classification rules completely from data. The learning algorithm for fuzzy sets can be constrained in order to obtain interpretable classifiers. However, fuzzy clustering provides more sophisticated rule learning procedures. We show that the learning process of NEFCLASS produces better results, if it is initialized by fuzzy clustering.

1 Introduction

The determination of fuzzy rules from data is currently a growing research topic. Fuzzy rules are used e.g. to build fuzzy controllers, fuzzy classifiers, or they are needed to support decision making processes. Classification and decision support with the help of fuzzy rules is the background for this paper. In a lot of commercial areas there are huge unstructured data collections (e.g. banking, insurance, medical treatment etc.), and there is a need to transfer data into information. Currently we observe new trends in data management like data warehousing and data mining. Several data bases of an organization are merged into a large data base, which represents several dimensions (views) of the data at the same time to support efficient data analysis without the need of computing complex joins. This so–called data warehouse also stores results of often needed calculations. On top of a data warehouse data mining tools are used to analyze the data, and to transform it into information that supports decision making.

Several well known methods are used in the process of data mining, like statistics, machine learning, neural networks and fuzzy data analysis. In this paper we consider techniques for deriving fuzzy rules from data. The advantage of fuzzy rules for decision making lies in their interpretability. Decision makers in industry are usually not statisticians, mathematicians, or AI experts. So it is important, that the results of the data analysis process is presented in a form that can be easily grasped by non–experts. The linguistic representation of fuzzy rules supports this approach.

In the following we will consider the computation of fuzzy classification rules from data. For deriving fuzzy rules, e.g. fuzzy clustering and neuro–fuzzy techniques can be used. Fuzzy clustering tends to produce rules which are hard to interpret, because fuzzy sets are produced individually for each rule by projection of clusters. Neuro–fuzzy methods train an initial classifier by changing the fuzzy sets under given constraints, and so they can produce fuzzy sets that can be more easily labeled with linguistic expression. On the other hand the rule generation capabilities of neuro–fuzzy approaches are not as sophisticated as fuzzy clustering approaches. We will therefore present a combination of both approaches. In the following sections we will shortly review our neuro–fuzzy approach NEFCLASS [Nauck and Kruse, 1995] and some techniques for generating rules by fuzzy clustering [Klawonn and Kruse, 1997]. Then we show how to initialize a NEFCLASS system with rules from fuzzy clustering, and present some results.

This paper appears in: Proc. Fourth European Congress on Intelligent Techniques and Soft Computing (EUFIT96), Aachen 1996.

2 Creating a Fuzzy Classifier by Supervised Learning

For the creation of a fuzzy classifier by supervised learning we use our neuro–fuzzy model NEFCLASS (NEuro–Fuzzy CLASSification). A neuro–fuzzy system is usually a fuzzy system that uses a learning algorithm which is derived from neural network theory. The changes computed by the learning algorithm are based on local information only, and the changes are also carried out locally. The fuzzy system is usually viewed as a special 3–layer feed–forward network architecture, where the units of the second layer represent the fuzzy rules. The fuzzy sets are represented as *fuzzy weights* on the connections from the input to the hidden layer, and in case of a fuzzy controller, also from the hidden to the output units.

NEFCLASS is used to derive fuzzy rules from a set of data that can be separated in different crisp classes. The fuzzy rules describing the data are of the form:
R : **if** x_1 is μ_1 **and** x_2 is μ_2 **and** ... **and** x_n is μ_n **then** the pattern $(x_1, x_2, \ldots, x_n)$ belongs to class i, where $\mu_1, \ldots, \mu_n$ are fuzzy sets. The task of the NEFCLASS model is to discover these rules and to learn the shape of the membership functions to determine the correct class or category of a given input pattern.

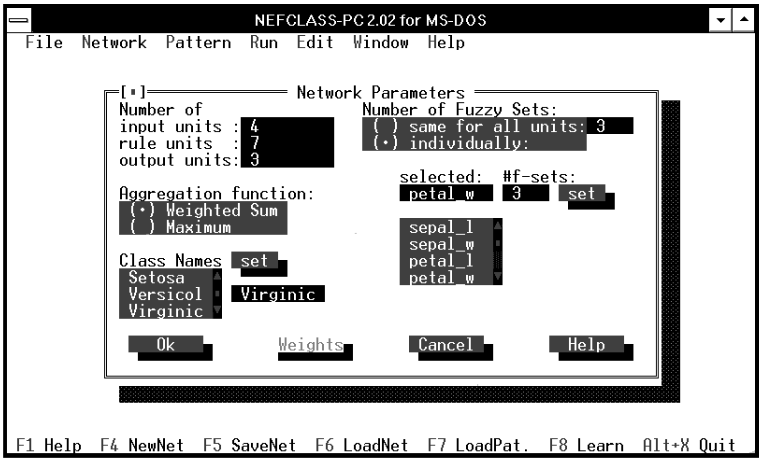

Figure 1: Definition of a NEFCLASS network in NEFCLASS–PC

A NEFCLASS system can be build from partial knowledge about the patterns, and can be then refined by learning, or it can be created from scratch by learning. A user has to define a number of initial fuzzy sets partitioning the domains of the input features, and must specify a value for k, i.e. the maximum number of rule nodes that may be created in the hidden layer (Fig. 1). The model has been implemented, and it is freely available on the Internet (see our WWW site). The tool helps the user to interpret the learning result (Fig. 2), and is mainly used as an interactive data analysis tool. The result of one training process is interpreted and used to obtain refined results by further training processes (e.g. using less variables or fuzzy sets).

The learning algorithm for NEFCLASS is described completely in [Nauck et al., 1996a; Nauck et al., 1996b]. In the following we will only give a short overview:

- **Initialization:** For each feature there is an input unit, and for each class there is one output unit. For each input unit an initial fuzzy partitioning is specified (e.g. a number of equally distributed triangular membership functions).
- **Rule Learning:** NEFCLASS starts without rules, and inserts fuzzy rules into the system during a first run through the training data. In a second run the rules are evaluated, and only the best r rules are kept, where r is given by the user. It is also possible to keep the best rules per class.
- **Fuzzy Set Learning:** For training the membership function a simple backpropagation–like heuristic is used. Depending on the output error for each rule unit a decision is made, whether the activation value has to be higher or lower. Each rule unit then changes its membership functions by changing their support. The user can specify several constraints such that the changes of the membership functions do not change the semantics of the underlying fuzzy model.
- **Fixed Output Weights:** For semantical reasons the weights to the output units are not learned.

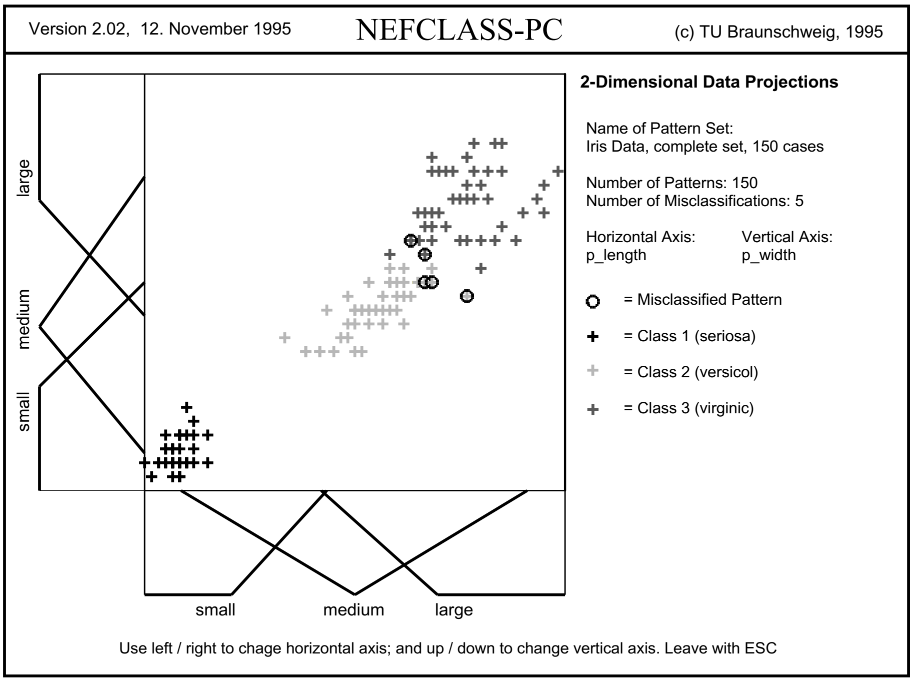

Figure 2: Two–dimensional projection of a data set (Iris data) and learned fuzzy partitionings in NEFCLASS–PC

If semantics are not an issue, these weights can also be trained, e.g. to obtain exact output values. However, in our applications we refrain from it.

The rule learning strategies of NEFCLASS work very well for many classification tasks even without using prior knowledge. For very complex tasks however, it is sometimes useful to include prior knowledge into the system, e.g. by specifying a partial rule base, which is then completed and tuned by the learning procedures. Fuzzy clustering algorithms provide more sophisticated rule learning strategies as they are included in the NEFCLASS learning algorithms. Therefore they are a method to initialize a NEFCLASS system.

3 Creating Fuzzy Classification Rules by Fuzzy Clustering

Nearly all fuzzy clustering algorithms try to find an adequate prototype for each fuzzy cluster and suitable membership degrees for the data to each cluster. Usually, the cluster algorithm aims at minimizing the objective function

$$J(X, U, v) \;=\; \sum_{i=1}^{c} \sum_{k=1}^{s} (u_{ik})^m d^2(\mathbf{v}_i, \mathbf{p}_k) \tag{1}$$

under the constraints that each cluster must not be completely empty, and usually that for every element the sum over its membership degrees to all clusters equals 1. $X = \{\mathbf{p}_1, \ldots, \mathbf{p}_s\} \subseteq \mathbb{R}^n$ is the data set, c is the number of fuzzy clusters, $u_{ik} \in [0,1]$ is the membership degree of datum $\mathbf{p}_k$ to cluster i, $\mathbf{v}_i \in \mathbb{R}^n$ is the prototype for cluster i, and $d(\mathbf{v}_i, \mathbf{p}_k)$ is the distance between prototype $\mathbf{v}_i$ and datum $\mathbf{p}_k$. The parameter $1 < m$ is called fuzziness index. For $m \to 1$ the clusters tend to be crisp, i.e. either $u_{ik} \to 1$ or $u_{ik} \to 0$, for $m \to \infty$ we have $u_{ik} \to 1/c$. Usually $m = 2$ is chosen.

The most simple fuzzy clustering algorithm is the fuzzy c–means (FCM) (see e.g. [Bezdek, 1981]) where the distance d is simply the Euclidean distance. It searches for spherical clusters of approximately the same size. Gustafson and Kessel [Gustafson and Kessel, 1979] and Gath and Geva [Gath and Geva, 1989] designed fuzzy clustering methods that are looking for hyper–ellipsoidal clusters of varying size. We refer to the corresponding algorithms by GK and GG, respectively, In both cases, in addition to the prototypes $\mathbf{v}_i$ and the membership degrees u_{ik}, for each cluster i a (positive definite) covariance matrix C_i is calculated. Based on this matrix a transformed Euclidean distance is computed allowing for hyper–ellipsoidal clusters. In any case for each cluster a matrix inversion has to be computed in every iteration step so that these algorithms have a much higher computational complexity than the FCM.

To generate fuzzy rules from fuzzy cluster analysis, it is desirable to have algorithms that search for axes–parallel hyper–ellipsoids only. The fuzzy sets of the antecedent of a rule are induced by projecting the data of a fuzzy cluster into each single dimension. This way we obtain discrete fuzzy sets which are approximated by a triangular or trapezoidal membership function. This procedure causes a loss of information because the Cartesian product of the induced membership functions does not reproduce the fuzzy cluster exactly. This loss of information is strongest in the case of arbitrarily oriented hyper–ellipsoids. Therefore we use modifications of GK and GG as they are described in [Klawonn and Kruse, 1997]. These algorithms look for axes–parallel hyper–ellipsoids only, and are also much simpler and faster than the original GK and GG.

Given a data set for which the correct classification is known, the fuzzy clustering algorithm is started with the number of clusters equal to the number of classes. To each cluster the class of the prototype or the class of the datum with the highest membership degree is assigned. For each cluster in which the rate of misclassifications exceeds a given upper bound, a new prototype in the neighborhood of the original prototype of the cluster is introduced, and the fuzzy clustering algorithm is applied again with the now increased number of clusters. This procedure is iterated, until the number of misclassifications is small enough for each cluster. By this the number of rules is kept as small as possible.

After the fuzzy clustering procedure has terminated the classification rules are constructed by projecting each cluster. Unfortunately the resulting fuzzy rule base is usually not easy to interpret, because the fuzzy sets are induced individually for each rule. For each feature there will be c usually different fuzzy sets. Some of these fuzzy sets may be similar, yet they are not identical. For a good interpretation it is necessary to have a fuzzy partitioning of (usually less than c) fuzzy sets, where each clearly represents a linguistic concept. It is therefore useful to provide a fuzzy partitioning for each feature, and map the induced fuzzy sets to fuzzy sets of these partitionings. This reinterpretation of the fuzzy rule base will usually result in a loss of classification performance. Therefore the rules should be used to initialize a NEFCLASS system that enhances the performance again by training.

4 NEFCLASS initialized by Fuzzy Clustering

In this section we show some results using fuzzy clustering to initialize a NEFCLASS system. We selected the "Wisconsin Breast Cancer" data set [Wolberg and Mangasarian, 1990] for demonstrating the benefits of combining fuzzy clustering with a neuro–fuzzy classifier. We used the our tool FCLUSTER (see Fig. 3) to derive fuzzy rules with the modified GK algorithm. The fuzzy sets were approximated by trapezoidal membership functions.

In Table 1 the classification results are shown (see also [Nauck et al., 1996a; Nauck et al., 1996b]). The first two rows show the performances of fuzzy clustering (FC) and NEFCLASS (using best per class rule learning) without combination. The percentage of correct classified pattern is not satisfactory in these two cases. For the third trial we re–interpreted the fuzzy rules produced by FC. We initialized each NEFCLASS input variable with a fuzzy partitioning of equally distributed triangular membership function labeled *small, medium* and *large.* Then we mapped each fuzzy set of the FC–generated rules (manually) to the most similar fuzzy set of the respective partitioning. After initializing the NEFCLASS system with these three rules, and training it for 80 epochs we received the result shown in the third row of Table 1.

After interpreting the fuzzy sets of the learning result we started another run where we used only 2 fuzzy sets for each feature. The fuzzy set labeled *medium* was almost in variables covered by one of its neighboring fuzzy sets. Therefore we concluded that 2 fuzzy set are sufficient for each variable. This time NEFCLASS was able to find 4 rules that produced an even better classification result. This result is based on the findings obtained from the third trial, and therefore it also benefits from the combination with fuzzy clustering.

5 Conclusions

We have shown that a combination of fuzzy clustering and neuro–fuzzy classification can produce better classification results, than either alone. To initialize NEFCLASS with rules obtained from fuzzy clustering, the fuzzy sets obtained by projecting the clusters must be mapped to fuzzy sets used by NEFCLASS. Currently an automatic mapping is implemented into NEFCLASS. Instead of using one of the available

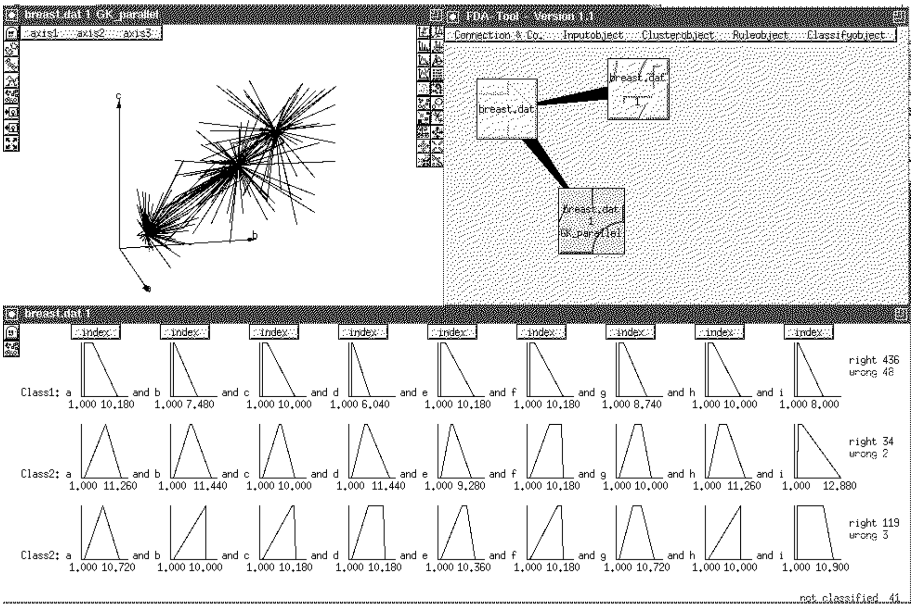

Figure 3: Fuzzy rules as they are induced by the modified GK algorithm of the tool FCLUSTER ("Wisconsin Breast Cancer" data)

Trial	Method	No. of rules	Result (% correct)
1	FC (modified GK)	3	94 (86.2%)
2	NEFCLASS	4	135 (80.4%)
3	FC + NEFCLASS	3	50 (92.7%)
4	NEFCLASS (based on trial 3)	4	24 (96.5%)

Table 1: Classification results for the Wisconsin–Breast–Cancer data

similarity measures for fuzzy sets, we think in terms of the learning procedure. A FC fuzzy set is mapped to that NEFCLASS fuzzy set that has to be changed least by the learning procedure to become identical to the FC fuzzy set.

References

J. C. Bezdek (1981). *Pattern Recognition with Fuzzy Objective Function Algorithms.* Plenum Press, New York.

I. Gath and A. Geva (1989). *Unsupervised Optimal Fuzzy Clustering.* IEEE Trans. Pattern Analysis and Machine Intelligence, 11:773–781.

D. Gustafson and W. Kessel (1979). *Fuzzy Clustering with a Fuzzy Covariance Matrix.* In *Proc. IEEE CDC*, pages 761–766, San Diego.

F. Klawonn and R. Kruse (1997). *Constructing a Fuzzy Controller from Data.* Fuzzy Sets and Systems, 85(1).

D. Nauck, F. Klawonn and R. Kruse (1996a). *Neuronale Netze und Fuzzy–Systeme, 2. erweiterte Auflage.* Vieweg, Wiesbaden.

D. Nauck and R. Kruse (1995). *NEFCLASS – A Neuro–Fuzzy Approach for the Classification of Data.* In K. George, J. H. Carrol, E. Deaton, D. Oppenheim and J. Hightower, eds.: *Applied Computing 1995. Proc. of the 1995 ACM Symposium on Applied Computing, Nashville, Feb. 26–28*, pages 461–465. ACM Press, New York.

D. Nauck, U. Nauck and R. Kruse (1996b). *Generating Classification Rules with the Neuro–Fuzzy System NEFCLASS*. In *Proc. Biennial Conference of the North American Fuzzy Information Processing Society NAFIPS'96*, Berkeley.

W. Wolberg and O. Mangasarian (1990). *Multisurface Method of Pattern Separation for Medical Diagnosis Applied to Breast Cytology*. Proc. National Academy of Sciences, 87:9193–9196.