International Journal of Uncertainty,
Fuzziness and Knowledge-Based Systems
Vol. 15, No. 5 (2007) 615–624
© World Scientific Publishing Company

World Scientific
www.worldscientific.com

VISUALIZATION OF FUZZY CLASSIFIERS

FRANK REHM

Institute of Flight Guidance, German Aerospace Center,
Lilienthalplatz 7, Braunschweig, 38108, Germany
frank.rehm@dlr.de

FRANK KLAWONN

Department of Computer Science,
University of Applied Sciences Braunschweig/Wolfenbüttel,
Salzdahlumer Str. 46/48, Wolfenbüttel, 38302, Germany
f.klawonn@fh-wolfenbuettel.de

RUDOLF KRUSE

Faculty of Computer Science, University of Magdeburg,
Universitätsplatz 2, Magdeburg, 39106, Germany
kruse@iws.cs.uni-magdeburg.de

Received 2 January 2007
Revised 9 August 2007

This paper presents different techniques to visualize high-dimensional fuzzy rule bases in relation to the classified data. The degree of membership to influential rules can be visualized for an entire data set. This enables the observer to detect conflicting or error-prone rules as well as misclassified feature vectors. Results are shown on a benchmark data set and on a weather data set that is used to predict flight durations on a major European airport.

Keywords: Fuzzy classifiers; visualization; multidimensional scaling; fuzzy rules; air traffic management.

1. Introduction

Fuzzy rules are a powerful instrument to model classification issues. The strength of fuzzy rules is their simple interpretability and their easy extraction from data or generation by hand. Nevertheless, if the data set is high-dimensional in the feature space and the underlying data structure is rather complicated, a resulting rule system can be fairly complex.

We propose in this paper to use multidimensional scaling (MDS) to map a high-dimensional rule base on the plane.[1] With MDS and related methods one tries to find a low-dimensional representation of the data while preserving distances or — generally — dissimilarity between data as an objective function for such transformations. The idea is to determine dissimilarity between the single rules of the classifier and then to apply MDS in order to map the rule representatives. This concept has been successfully applied on comparable issues.[2,3] Additionally, classified feature vectors will be mapped while preserving membership degrees to the two rules that yield the highest response to the respective feature vector. We will apply this technique on a benchmark example to demonstrate various aspects and benefits of the proposed tool. Furthermore we show results from applying this method on weather data that is used to predict flight durations of arriving aircraft at Frankfurt Airport.

The rest of the paper is organized as follows: In Section 2 we recall Sammon's mapping as a common representative for multidimensional scaling. In Section 3 we describe the proposed method. In Section 4 we present our results. Finally we conclude with Section 5.

2. Multidimensional Scaling

Multidimensional scaling (MDS) is a method that estimates the coordinates of a set of n objects $Y = \{y_1, \ldots, y_n\} \subset \mathbb{R}^q$ in a feature space of specified (low) dimensionality ($q \ll p \quad \text{with} \quad q, p \in \mathbb{N}$) that come from data $X = \{x_1, \ldots, x_n\} \subset \mathbb{R}^p$ trying to preserve the distances and dissimilarity between pairs of objects respectively. Different ways of computing dissimilarity and various functions relating the dissimilarity to the actual data are commonly used. These dissimilarity values are usually stored in a dissimilarity matrix

$$D^x = \left(d^x_{ef}\right), \; d^x_{ef} = \|x_e - x_f\|, \; e, f = 1, \ldots, n \,. \tag{1}$$

The estimation of the coordinates will be carried out under the constraint that the error between the dissimilarity matrix D^x of the data set and the dissimilarity matrix $D^y = \left(d^y_{ef}\right), \; d^y_{ef} = \|y_e - y_f\|, \; e, f = 1, \ldots, n$ of the corresponding transformed data set will be minimized.

Thus, different error measures to be minimized can be considered, i.e. the absolute error, the relative error or a combination of both. A commonly used error measure, the so-called Sammon's mapping

$$E = \frac{1}{\sum\limits_{e=1}^{n} \sum\limits_{f=e+1}^{n} d^x_{ef}} \sum_{e=1}^{n} \sum_{f=e+1}^{n} \frac{\left(d^y_{ef} - d^x_{ef}\right)^2}{d^x_{ef}} \tag{2}$$

describes the absolute and the relative quadratic error.[4] To determine the transformed data set Y by means of minimizing error E a gradient descent method can be used. By means of this iterative method, the parameters y_k to be optimized, will be updated during each step proportional to the gradient of the error function E.

Calculating the gradient of the error function leads to

$$\frac{\partial E}{\partial y_k} = \frac{2}{\sum\limits_{e=1}^{n}\sum\limits_{f=e+1}^{n} d^x_{ef}} \sum_{f\neq k} \frac{d^y_{kf}-d^x_{kf}}{d^x_{kf}}\,\frac{y_k-y_f}{d^y_{kf}}. \tag{3}$$

After random initialization for each projected feature vector y_k a gradient descent is carried out and the dissimilarity values d^y_{ef} as well as the gradients $\frac{\partial E}{\partial y_k}$ will be recalculated again. The algorithm terminates when E becomes smaller than a certain threshold. A crucial step when applying MDS to visualize a fuzzy rule base is to determine the dissimilarity matrix. The following section discusses this issue.

3. Visualization and Classification of High-Dimensional Data

Instead of constructing an entire rule base by hand, one can automatically derive rules from data. Fuzzy rules are often obtained from fuzzy clusters by projecting the clusters to the coordinate spaces, but also various other techniques are commonly used. Typically, fuzzy rules describe an inference scheme:

$$\mathcal{R}: \quad \texttt{if} \quad \textit{antecedent} \quad \texttt{then} \quad \textit{consequent}$$

where the antecedent is described by the input variables:

$$x_1 \;\texttt{is}\; A^{(1)} \quad \texttt{and} \quad \ldots \quad \texttt{and} \quad x_\ell \;\texttt{is}\; A^{(\ell)}$$

and the consequent by a single output variable y `is` B^1. Input variables are defined by means of membership functions. A trapezoidal membership function is depicted in Figure 1. Output variables, as they are considered here, are always singletons. Singletons can be taken as the special forms of the trapezoidal function, where the four parameters $< a_i, b_i, c_i, d_i >$ are identical.

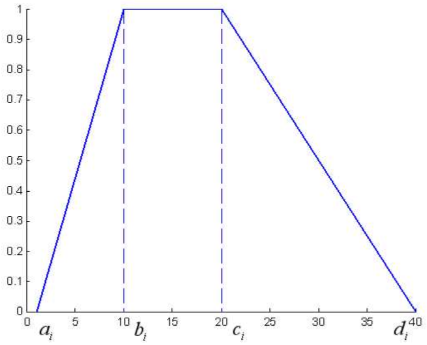


Fig. 1. A trapezoidal membership function.

[1] There are also other concepts of fuzzy rules, e.g. where the rule consequent is employed as a linear function of the input variables, which are not considered here.

Despite the good interpretability of single fuzzy rules, the analysis of an entire rule base can be a tedious task. Particularly if the data comprehend many attributes, i.e. the input data is high-dimensional, interpretation becomes difficult.

When generating fuzzy rules from data, different rule styles are commonly formed. For instance, an algorithm is proposed[5] that generates fuzzy rules from data, where each rule holds an independent membership function for each variable of the data. The algorithm forms trapezoidal membership functions which are defined by four parameters $< a_i, b_i, c_i, d_i >$ (see Figure 1). The rule's core region for attribute i is defined by parameter b_i and c_i. It describes the region of the membership function that is supported by training examples during the rule learning phase. The rule's support region for attribute i is defined by parameter a_i and d_i. The support region might be constrained as the figure shows, but also open to $\pm\infty$.

Having this rule style it is easy to derive rule center vectors (rule representatives) for every single rule. Such a center vector $v_{\mathcal{R}_e}$ can be determined by means of the core region's center for each attribute i of the rule

$$v_{\mathcal{R}_e}^{(i)} = b_{\mathcal{R}_e}^{(i)} + \frac{\left| b_{\mathcal{R}_e}^{(i)} - c_{\mathcal{R}_e}^{(i)} \right|}{2}. \tag{4}$$

As mentioned earlier, Sammon's mapping produces low-dimensional layouts based on a dissimilarity matrix. Dissimilarity of two rules $\mathcal{R}_e$ and $\mathcal{R}_f$ can be defined by means of the distance of the according pair of rule center vectors

$$d_{ef} = \| v_{\mathcal{R}_e} - v_{\mathcal{R}_f} \|.$$

The majority of rule construction algorithms tries to form rules that use a minimal number of variables to describe the classification task.[6,7] Then rules cannot be represented by means of center vectors and dissimilarity matrices cannot be defined over distances between rule representatives. Nevertheless, dissimilarity of rules can be defined anyhow.

Dissimilarity can be defined by comparing single membership functions of rule pairs. Starting with an initially zero valued dissimilarity matrix, dissimilarity of a rule pair can be then augmented if membership functions for the same variable do not overlap

$$w_{ef}^{(i)} = \begin{cases} 1 \text{ , if } A_{\mathcal{R}_e}^{(i)} \cap A_{\mathcal{R}_f}^{(i)} = \emptyset \\ 0 \text{ , otherwise} \end{cases} \tag{5}$$

$$d_{ef} = \sum_{i=1}^{\ell} w_{ef}^{(i)}. \tag{6}$$

Dissimilarity of a rule pair that is not overlapping (that means any of the membership functions are disjoint)

$$A_{\mathcal{R}_e}^{(i)} \cap A_{\mathcal{R}_f}^{(i)} = \emptyset \qquad , \forall i \tag{7}$$

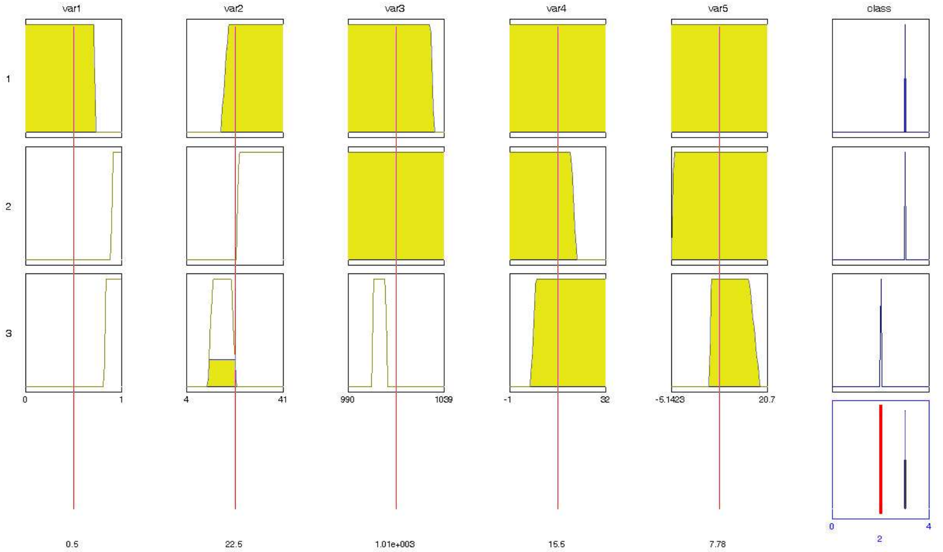


Fig. 2. An exemplary rule base with individual membership functions for each variable.

or that predicts different classes

$$B_{\mathcal{R}_e} \neq B_{\mathcal{R}_e} \tag{8}$$

can also be augmented. A suchlike derived dissimilarity matrix can be used in a straight forward manner with Sammon's mapping.

Figures 2 and 3, both show a rule base with individual membership functions for each variable and a rule base with a reduced number of membership variables, respectively. It can be easily seen in Figure 3 that rules $\mathcal{R}_3$ and $\mathcal{R}_4$ are similar to a certain degree since the membership functions for variable 2 are identical and the membership functions for variable 5 overlap partly. Further, both rules predict the same class in this example — namely class 3. Thus dissimilarity d_{34} will be fairly low for this rule pair. Contrary, rule $\mathcal{R}_8$ and rule $\mathcal{R}_9$ are quite dissimilar since several membership functions for identical variables do not or or almost not overlap. Moreover, both rules predict different classes. For rule pairs whose rules mainly cover different variables, dissimilarity can hardly be determined. Such a case can be observed with rule pair $(\mathcal{R}_4, \mathcal{R}_{11})$. Only variable 2 gives a direct hint for dissimilarity of this rule pair.

When learning classifiers based on high-dimensional data, the classifier itself will also be high-dimensional somehow. Thus its visualization including the visualization of the classified data is all but trivial. As discussed in the previous section, Sammon's mapping finds low-dimensional layouts trying to preserve a given dissimilarity matrix. As mentioned earlier, a dissimilarity matrix can be easily defined for arbitrary fuzzy rule bases. Thus, we can apply Sammon's mapping in order to find a 2-dimensional mapping of the rule base. Once the rule base is mapped on

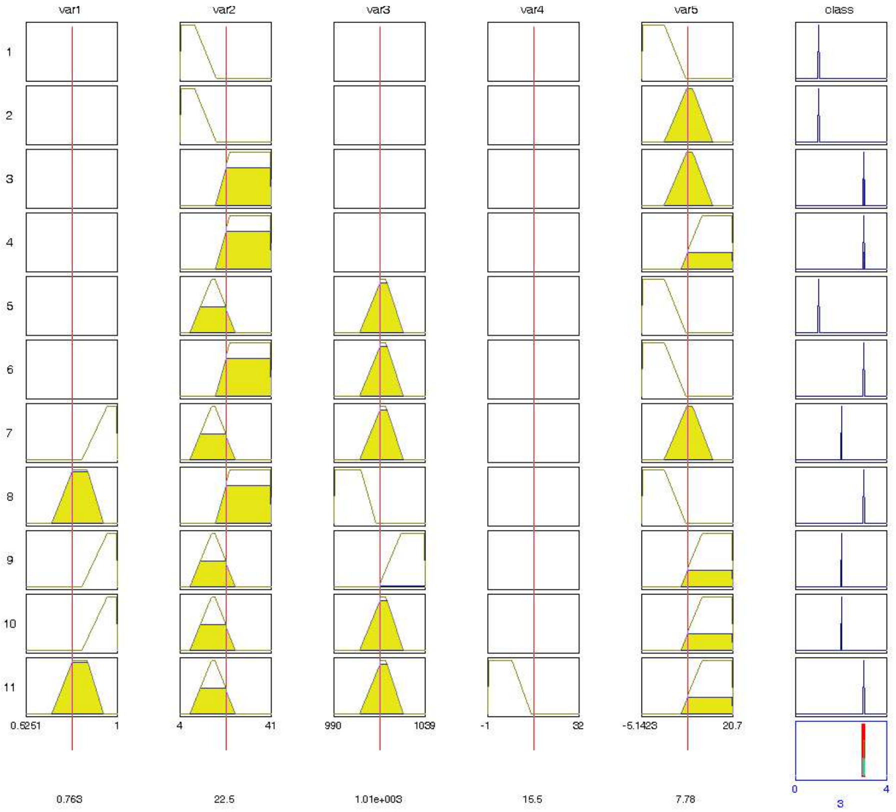


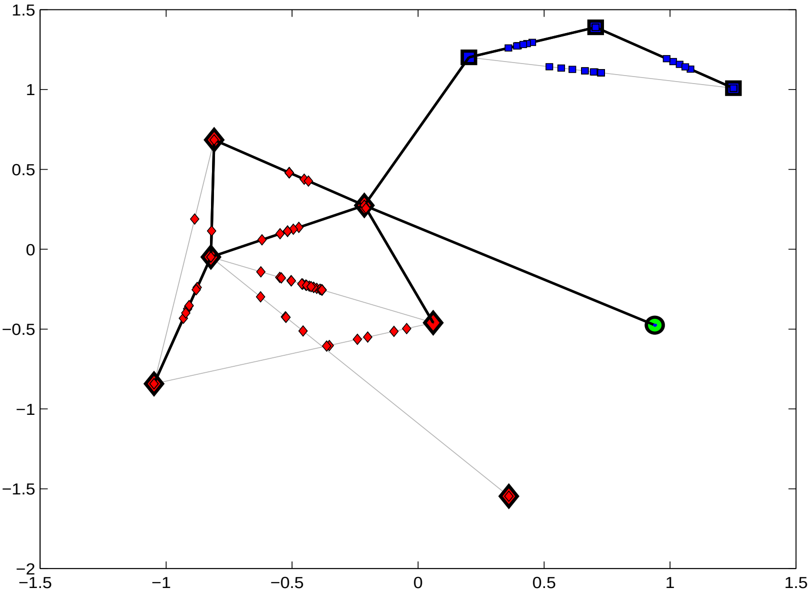
Fig. 3. An exemplary rule base with membership functions for a reduced number of variables.

the plane it remains to visualize the classified data. Due to dimension reduction we cannot preserve all characteristics of the data exactly. Indeed, it is even better to emphasize essential facts and to smooth dispensable information. In our example it is of no importance to preserve the data structure, e.g. the relative position of data points, but the degree of firing to the rules yielding the highest membership degree to the respective data point. We propose to place each single data point proportional to both rule representatives that yield the highest response.

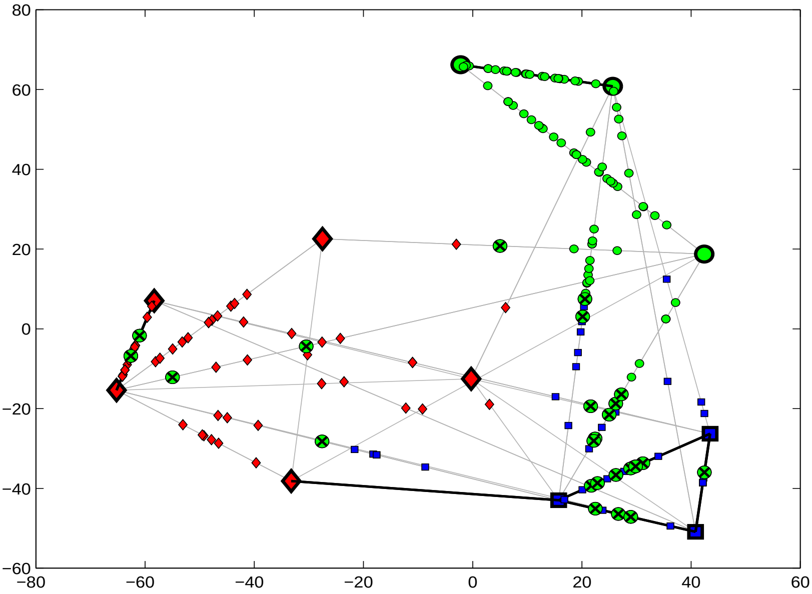
In the following section we will demonstrate our visualization tool on a benchmark example and on a practical example originating from an active industrial application as well.

4. Results

Figure 4(a) depicts the result of the proposed visualization tool on the Wine data set. The Wine data set results from a chemical analysis of wines grown in the



(a) 2D-Visualization of a rule classifier for the Wine data set.



(b) 2D-Visualization of a rule classifier for the Aviation data set.

Fig. 4. 2D-Visualization of rule classifiers.

same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines.

For this example the fuzzy rule learning algorithm as described in Ref. 5 was applied. Ten rules were obtained which classify the entire data set correctly. Rule center vectors are visualized by big squares, diamonds and circles (□, ◇, ○). Connections between rule center vectors (drawn by a bold line) indicate their neighborhood regarding the core region. Rules of the same class are visualized by the same symbol. Additionally, data objects are visualized by means of small symbols. A feature vector's membership to a certain rule can be identified by means of its symbol and its distance to rule center vectors.

The figure reveals some interesting facts. In consequence of placing vectors in the plane depending on their membership degree to the two rules that yield the highest response, classified feature vectors will be placed on an imaginary line that connects two rule center vectors. Note, feature vectors may not only be represented by neighboring rules corresponding to the core-based neighborhood definition whose neighborhood is visualized by lines in the figure. As the figure reveals, for some neighboring rules the data set contains no data that lie in the core regions of those rules. Two out of ten rules represent data that lie not in any of the core regions of these rules. If two rules yield similar membership degrees to a feature vector, it will be placed in the middle between these rule center vectors. Of course, the classification that will be done in such cases is not that confiding since the decision comes randomly if no further information is available.

The depicted classifier is almost ideal. Despite of one conflicting ◇-rule that overlaps with one □-rule and one ○-rule no misclassifications occur. There are also no rule pairs representing different classes that compete for the same data. Actually, the visualization tool can provide much more insight into classifiers that comprise problematic aspects. The following example will demonstrate some more prospects of this technique.

As a second example the visualization tool is applied on data that originates from an active industrial application. The data set describes the weather situation at Frankfurt Airport. Recent studies concerned the prediction of aircraft flight durations of arriving aircraft depending on the weather situation present at the airport.[8–10] Predicting the delay that an aircraft may have allows to retard flights on other airports that depart to the airport but also to coordinate ground activities such as baggage handling.

Figure 4(b) shows a visualization of the rule base from Figure 3 that is generated by the NEFCLASS fuzzy rule learner.[7] In this example the flight duration times were grouped into three classes: short flights (○), medium flights (□) and long flights (◇). The visualization reveals that the rule base contains some overlapping rules — rules that get non-zero membership degrees to a shared subset of data of the same class — but also two conflicting rules that get non-zero membership degrees to a shared subset of data of a different class. Misclassified data are visualized by circled cross (⊗). Note, misclassified medium flights and long flights are suppressed here in order to keep the visualization clear. Eye-catching are some misclassified short flights whose two highest membership degrees get rules that actually classify long flights. Investigations on the raw traffic data have shown that the flight durations for these flights where originally missing. These missing values were replaced by the mean value which obviously does not suitably reflect the reality. There are also some short flights that get high membership degrees to rules that classify medium and long flights. Despite of bad weather conditions and demanding traffic at the airport these aircraft could land very fast.

5. Conclusions

In this paper we have presented a visualization technique that provides a concise view to fuzzy rule bases and to the classified data on an integrated mapping. As an extension to earlier proposals, we also described new approaches to measure dissimilarity. By means of this new tool meaningful interpretations can be extracted. Neighboring, conflicting or overlapping rules can be found and outlier data can be identified. We demonstrated these aspects with two practical examples. It would be also interesting to see visualization of fuzzy systems on regression problems. While the winner-takes-all principle is applied on classification problems, the weighted average of all rules is determined with regression. Beside the research on further dissimilarity measures, visualization of fuzzy regressors will be a challenge of future work.

References

1. I. Borg and P. Groenen, *Modern Multidimensional Scaling: Theory and Applications* (Springer, Berlin, 1997).
2. F. Rehm, F. Klawonn and R. Kruse, Rule classification visualization of high-dimensional data, *Proceedings of the 11th International Conference on Information Processing and Management of Uncertainty in Knowledge-based Systems* (*IPMU 2006*), Paris, 2006, pp. 1944–1948.
3. T. R. Gabriel, K. Thiel and M. R. Berthold, Rule Visualization based on Multi-Dimensional Scaling, *IEEE International Conference on Fuzzy Systems*, Vancouver, Canada, 2006, pp. 66–71.
4. J. W. Sammon, A nonlinear mapping for data structure analysis, *IEEE Transactions on Computers* **18** (1969) 401–409.
5. M. R. Berthold, Mixed Fuzzy Rule Formation, *International Journal of Approximate Reasoning* (*IJAR*) **32** (2003) 67–84.
6. H. Genther and M. Glesner, Automatic generation of a fuzzy classification system using fuzzy clustering methods, *Proceedings of the ACM Symposium on Applied Computing* (*SAC 1994*), Phoenix, 1994, pp. 180–183.
7. D. Nauck and R. Kruse, Neuro-fuzzy classification with NEFCLASS, in *Operations Research Proceedings*, eds. A. Bachem, U. Derigs, D. Fischer, U. Leopold-Wildburger and R. Möhring (Springer, Berlin, 1995) pp. 294–299.
8. M-J. Lesot, F. Rehm, F. Klawonn and R. Kruse, Prediction of aircraft flight duration, *Proceedings of the 11th IFAC Symposium on Control in Transportation Systems*, Delft, 2006, pp. 107–112.

9. F. Rehm, F. Klawonn and R. Kruse, Single cluster visualization to optimize air traffic management, in *Advances in Data Analysis, Studies in Classification, Data Analysis, and Knowledge Organization*, eds. H.-J. Lenz and R. Decker (Springer, Berlin/Heidelberg, 2007), pp. 319–325.
10. F. Rehm and F. Klawonn, Learning methods for air traffic management, in *Symbolic and Quantitative Approaches to Reasoning with Uncertainty, Proceedings of the 8th European Conference*, ed. L. Godo (Springer, Berlin/Heidelberg, 2005), pp. 992–1001.