Rule Chains for Visualizing Evolving Fuzzy Rule-Based Systems

Sascha Henzgen, Marc Strickert and Eyke Hüllermeier

Abstract Evolving fuzzy systems are data-driven fuzzy (rule-based) systems supporting an incremental model adaptation in dynamically changing environments; typically, such models are learned on a continuous stream of data in an online manner. This paper advocates the use of visualization techniques in order to help a user gain insight into the process of model evolution. More specifically, *rule chains* are introduced as a novel visualization technique for the inspection of evolving Takagi-Sugeno-Kang (TSK) fuzzy systems. To show the usefulness of this techniques, we illustrate its application in the context of learning from data streams with temporal concept drift.

1 Introduction

Visualization has become an important tool in data-driven research fields, such as machine learning and data mining. Apart from the visualization of data objects, data relationships, and aggregated information content [4], the visualization of models learned from data has recently attracted increasing attention in the field of computational intelligence [1, 2, 8, 9]. Going beyond the presentation of static models, this paper suggests the use of visualization techniques for tracking *evolving* models, that is, models that are learned and adapted in an online manner on a continuous stream of data [3, 10].

More specifically, we developed an interactive visualization tool that allows for monitoring evolving TSK fuzzy inference systems in real time. So-called *rule chains* are proposed for visualizing changes of a rule system between two consecutive time points. Experimentally, we show that characteristic patterns emerge in a rule chain visualization when applied to evolving rule systems learned from data streams exhibiting concept drift.

Department of Mathematics and Computer Science, Philipps University Marburg, Germany, {henzgen,strickert,eyke}@mathematik.uni-marburg.de

This paper is structured as follows. Before introducing our rule chain visualization technique in Section 3, we revisit the underlying TSK fuzzy rule-based systems in Section 2. In Section 4, an experimental study with synthetic data is presented, showing the usefulness of our visualization technique for analyzing the evolution of fuzzy models learned on data streams with concept drift. Conclusions are drawn in Section 5.

2 TSK Fuzzy Rule-based Systems

In this work, we consider first order Takagi-Sugeno-Kang (TSK) fuzzy rule-based systems $\mathfrak{R} = \{R_1, \ldots, R_N\}$ implementing maps of the form

$$f : \mathbb{R}^p \to \mathbb{R},\, \mathbf{x} \mapsto \sum_{i=1}^{N} l_i(\mathbf{x}) \cdot \Psi_i(\mathbf{x}) \enspace . \tag{1}$$

Here, the antecedent of each rule R_i is modeled as a conjunction of p fuzzy sets with Gaussian membership function, each characterized by its center $c_{i,j}$ and width $\sigma_{i,j}$; for a given input vector $\mathbf{x} = (x_1, \ldots, x_p) \in \mathbb{R}^p$, the relevance ("firing strength") of an antecedent part is evaluated as follows:

$$\Psi_i(\mathbf{x}) = \frac{\exp\left[-\frac{1}{2}\sum_{j=1}^{p}\left((x_j - c_{i,j})^2/\sigma_{i,j}^2\right)\right]}{\sum_{k=1}^{|\mathfrak{R}|}\exp\left[-\frac{1}{2}\sum_{j=1}^{p}\left((x_j - c_{k,j})^2/\sigma_{k,j}^2\right)\right]} \tag{2}$$

The conclusion of R_i is a linear function specified by a $(p+1)$-dimensional weight vector $\mathbf{w}_i = (w_{i,0}, w_{i,1}, \ldots, w_{i,p})$:

$$l_i(\mathbf{x}) = w_{i,0} + w_{i,1}x_1 + w_{i,2}x_2 + \ldots + w_{i,p}x_p \tag{3}$$

Learning TSK models of that kind on a continuous (and potentially unbounded) stream of data $(\mathbf{z}^{(1)}, \mathbf{z}^{(2)}, \mathbf{z}^{(3)}, \ldots)$ in the form of input/output tuples $\mathbf{z}^{(t)} = (\mathbf{x}^{(t)}, y^{(t)})$ essentially means applying a learning algorithm $\mathcal{A}$ that adapts the current rule model after each newly observed example. Thus, starting with a model $\mathfrak{R}^{(0)}$ at time $t = 0$, a corresponding sequence of models $(\mathfrak{R}^{(0)}, \mathfrak{R}^{(1)}, \mathfrak{R}^{(2)}, \ldots)$ is produced, where $\mathfrak{R}^{(t)} = \mathcal{A}(\mathfrak{R}^{(t-1)}, \mathbf{z}^{(t)})$ is obtained by modifying certain rules in $\mathfrak{R}^{(t-1)}$, by merging different rules into a single new rule, or by creating a new rule from scratch. For a proper handling of rules, it is important that all rules have a unique label, no matter whether freshly generated or originating from a merging process. A rule $R^{(t)}$ in $\mathfrak{R}^{(t)}$ which evolved from a rule $R_i^{(t-1)}$ in $\mathfrak{R}^{(t-1)}$ with label i keeps that label. The history $\mathfrak{H}_i$ is the set of all rules labeled with i.

3 Visualizing Model Evolution with Rule Chains

In this section, we introduce our idea of rule chains for visualizing important aspects of the evolution of fuzzy rule-based systems. Essentially, a rule chain seeks to capture the changes of a single rule between two consecutive time points. An important prerequisite for visualizing such changes is the availability of meaningful measures of similarity or distance between fuzzy rules. Therefore, prior to explaining our visualization technique, we introduce measures of this kind. Since an independent treatment of the antecedent parts and the conclusion parts of fuzzy rules is informative for different aspects of the visualization, we measure the similarity of both parts separately.

3.1 Similarity and distance measures for fuzzy rules

3.1.1 Antecedence similarity

Recall that the antecedence part M_k of a rule R_k is a conjunction of fuzzy sets $\mu_{k,i}$ with normalized Gaussian membership function, one for each input variable x_i (cf. Section 2). We define the similarity between two antecedence parts M_k and M_l of two rules R_k and R_l by

$$\mathfrak{S}(M_k, M_l) = \min\Big(\mathfrak{s}(\mu_{k,1}, \mu_{l,1}), \mathfrak{s}(\mu_{k,2}, \mu_{l,2}), \ldots, \mathfrak{s}(\mu_{k,p}, \mu_{l,p})\Big) \ , \tag{4}$$

where $\mathfrak{s}(\mu_{k,i}, \mu_{l,i})$ is a standard similarity between fuzzy sets, namely the size of their intersection (pointwise minimum of membership degrees) normalized by the size of the larger of the two:

$$\mathfrak{s}(\mu_{k,i}, \mu_{l,i}) = \frac{|\mu_{k,i} \cap \mu_{l,i}|}{\max(|\mu_{k,i}|, |\mu_{l,i}|)} \tag{5}$$

The size $|\mu|$ of a Gaussian fuzzy set μ is defined by the area under the membership function. Thus, its computation comes down to solving an integration problem for which no closed-form solution exists. Therefore, we exploit the connection between the cumulative distribution function of the normal distribution and the error function $\mathrm{erf}(z) = 2/\sqrt{\pi}\int_0^z e^{-\tau^2} d\tau$, for which efficient numerical implementations exists. Thus, the area under a normalized Gaussian function with mean c and standard deviation σ can be efficiently determined as follows:

$$FN_{c,\sigma}(z) = \frac{\sigma\sqrt{2\pi}}{2}\left(1 + \mathrm{erf}\left(\frac{z-c}{\sigma\sqrt{2}}\right)\right) \tag{6}$$

3.1.2 Distance between the rule centers

Another similarity measure based on rule antecedence parts is the distance between rule centers, where the center of a rule R_k is defined as

$$c_k = (c_{k,1}, c_{k,2}, \ldots, c_{k,p}) \ , \tag{7}$$

with $c_{k,i}$ the center of the i^{th} fuzzy set in the antecedence of the k^{th} rule. By using the Euclidean metric, the distance between two rules is

$$\mathfrak{D}(R_k, R_l) = \|c_k - c_l\| \ . \tag{8}$$

For the purpose of visualization, it is desirable to have the measure normalized to the range $[0, 1]$. In rule chains, the computation of distances is restricted to adjacent rules $R_i^{(t)}$ and $R_i^{(t+1)}$ from the same history $\mathfrak{H}_i$ (the history of the i^{th} rule). Therefore, the normalization is done as follows:

$$\mathfrak{DN}\left(R_i^{(t)}, R_i^{(t+1)}\right) = \frac{\mathfrak{D}\left(R_i^{(t)}, R_i^{(t+1)}\right)}{\max_{(R_i^{(\tau)}, R_i^{(\tau+1)}) \in \mathfrak{H}_i \times \mathfrak{H}_i} \mathfrak{D}\left(R_i^{(\tau)}, R_i^{(\tau+1)}\right)} \tag{9}$$

3.1.3 Angle similarity

The angle similarity is a measure of similarity between the conclusion parts of two rules. The conclusion part (3) of a TSK fuzzy rule R_i defines a p-dimensional hyperplane $H_i = \{(x_1, x_2, \ldots, x_n, l_i(\mathbf{x})) \,|\, \mathbf{x} \in \mathbb{R}^n\}$ with normal vector $\mathbf{v}_i = (w_{i,1}, w_{i,2}, \ldots, w_{i,p}, -1)$. Using this normal vector, we can compute the angle between two conclusions H_i and H_j by

$$\alpha = \arccos\left(\frac{\mathbf{v}_i \cdot \mathbf{v}_j}{\|\mathbf{v}_i\| \cdot \|\mathbf{v}_j\|}\right) \ , \tag{10}$$

and finally their angle similarity [6, 7] as

$$\mathfrak{S}_\alpha(H_i, H_j) = \begin{cases} 1 - \frac{2}{\pi}\alpha & \text{if } \alpha \leq \frac{\pi}{2} \\ \frac{2}{\pi}\left(\alpha - \frac{\pi}{2}\right) & \text{else} \end{cases} \ . \tag{11}$$

3.1.4 Conclusion intercept difference

In addition to the angle similarity of two conclusions, we also consider the difference in their y-intercept, which is not captured by this similarity. Although this difference might indeed be negligible on a global scale, one should keep in mind that the influence of a conclusion is localized by the rule antecedence; and locally, the constant term may clearly make a difference. Therefore, we define the conclusion y-intercept difference by

$$\mathfrak{CY}\left(R_i^{(t)}, R_i^{(t+1)}\right) = \frac{\left|w_{i,0}^{(t)} - w_{i,0}^{(t+1)}\right|}{\max_{(R_i^{(\tau)}, R_i^{(\tau+1)}) \in \mathfrak{H}_i \times \mathfrak{H}_i} \left|w_{i,0}^{(\tau)} - w_{i,0}^{(\tau+1)}\right|} , \qquad (12)$$

with $w_{i,0}$ denoting the y-intercept of the conclusion of the rule R_i.

3.2 Rule chains

We propose rule chains as an adequate means for visualizing specific aspects of the evolution of a fuzzy rule-based system. A rule history $\mathfrak{H}$ is visualized as a horizontal pearl chain, where every pearl represents a rule at a certain time point. The first time point is located at the left and the most recent time point at the right end. Pearls along a chain are connected with different types of links. Focusing on a measure of interest, links between adjacent pearls can represent the corresponding similarity or distance between consecutive rules. As illustrated in Figure 1, there are three types of links that correspond, respectively, to the measures introduced in Section 3.1.

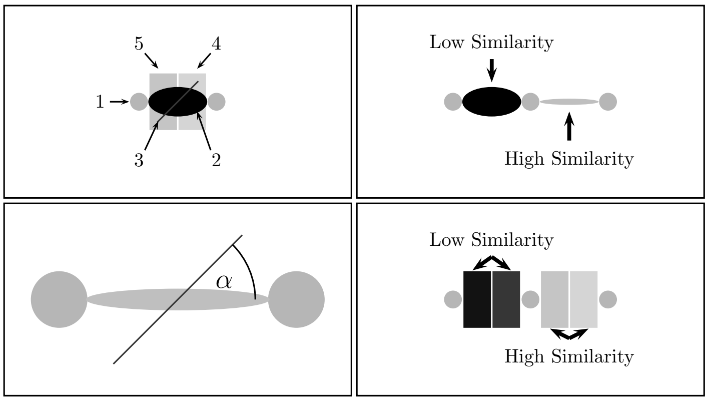

Fig. 1 The upper left panel shows all five elements of a rule chain. The elements pointed at by the numbered arrows are: (1) rule pearl, (2) antecedence similarity link, (3) angle line, (4) angle similarity block, (5) conclusion shift block. The top right panel describes the meaning of different heights and color intensities of ellipsoid links. The bottom right panel describes the meaning of different color intensities of rectangular links. Generally, both types of links can be used for visualizing any given similarity or distance measure. The lower left panel shows the rotated line segment for visualizing the angle between the conclusions of two adjacent rules.

In addition to the horizontal time point position, the vertical position of a pearl determines the history of the associated rule. The antecedence similarity link corresponds to the antecedence similarity measure or the Euclidean distance of centers. The degree of similarity is connected to the color intensity and shape of the ellipse. The lower the similarity, the bigger and darker is the ellipse. The angle similarity block visualizes the angle similarity, and the conclusion shift block displays differences of y-intercepts of the related rule conclusions. The color intensity of both blocks is the higher, the lower the similarity and the higher the difference is, respectively. Thus, the three types of links highlight different aspects of change between two rules. The angle line is mathematically positively rotated by α against the horizon to display the angle between two temporally adjacent conclusions.

For visualizing the evolving system as a whole, the horizontal rule chains are stacked (Figure 2). A single rule chain gives a quick overview of a rule's lifetime and development, and stacking them provides a holistic view on the development of the whole rule chain system.

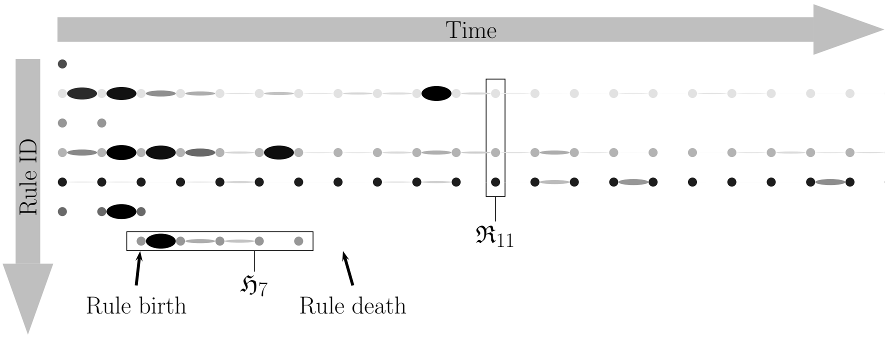

Fig. 2 This figure shows a cut-out from a rule chain system. All rules, symbolized by pearls, belonging to the same rule system are aligned vertically. Rules belonging to the same rule history are organized in horizontal chains. The pearl links are associated with similarity measures. Here, only the antecedence similarity links (ellipses) are shown.

3.3 Concept drift detection with rule chains

An important requirement for evolving (fuzzy) systems is a quick discovery and adequate reaction to so-called *concept drift* [3]. Roughly speaking, a concept drift is a (gradual) change of the data-generating process in the course of time, that is, a change of the probability distribution P that generates input/output tuples emitted by the data stream.

Rule chains provide an adequate (visual) means for discovering concept drift and monitoring the fuzzy system's reaction to this drift. In fact, as a reaction to a change of the data generating process, the fuzzy system is expected to adapt a possibly large number of rules (both, antecedence and/or conclusion parts). Likewise, new rules will typically be created and existing ones will be deleted.

Changes of that kind naturally produce observable patterns in the visualized rule chain system. More precisely, a simultaneous change of the antecedence or conclusion parts of many rules will produce noticeable vertical lines in the rule chain system (Figure 3). Such patterns can be amplified by defining thresholds for the minimal change to be drawn. The simultaneous appearance or disappearance of many rules produces long vertical edges in the displayed rule chains. This enables the user to recognize potential concept drifts in the data. More correctly, since we are visualizing the model and not the data evolution, the observation of such patterns should only be taken as an indication (and not as a proof) of a possible concept drift.

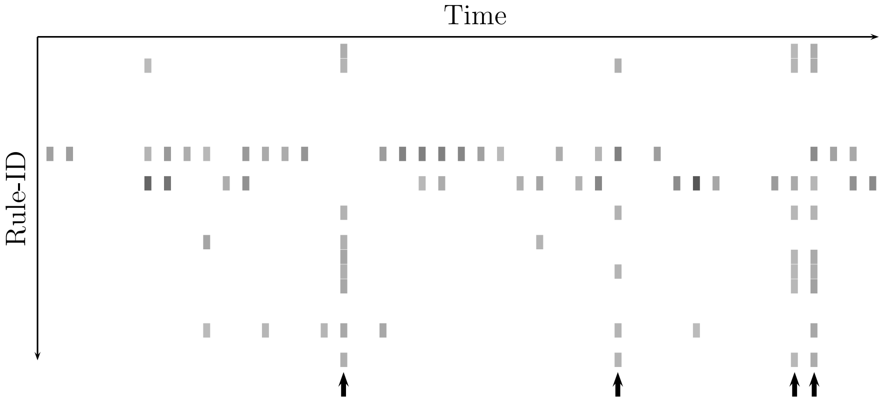

Fig. 3 This figure highlights patterns in the rule chains which indicate significant changes in the model and, therefore, suggest a possible concept drift in the data. The bold arrows are marking time points at which an abrupt change of the conclusion parts of many rules can be observed as vertical lines.

4 Illustration

This section presents an illustration of our rule chain visualization tool that is meant as a first proof of concept. For this purpose, we make use of synthetic data, which allows for conducting controlled experiments and, since the "ground truth" is known for this data, for judging the plausibility of the results—properties that are obviously not offered by real data. Moreover, while our visualization tool is in principle independent of the learning algorithm, we used FLEXFIS [5] as a concrete implementation and set the forgetting factor to 0.9. This FLEXFIS specific factor controls the forgetting of the inverse hessian matrix during the recursive weighted least squares optimization of the rule consequents.

To generate a data stream, we concatenate a number of synthetic datasets. Every dataset is associated with a time point determined by the position p of the dataset in the concatination. An instance of a dataset consists of a two-dimensional input and a real-valued output. For the initial dataset, three normally distributed clusters C_1, C_2, C_3 are generated. Every cluster C_i is characterized by its center $\mathbf{c}_i = (c_{i,1}, c_{i,2})$, its width $\boldsymbol{\sigma}_i = (\sigma_{i,1}, \sigma_{i,2})$ and the size n_i. The output y associated with an instance $\mathbf{x} = (x_1, x_2)$ depends on the cluster C_i and is determined by a linear function:

$$y = f_i(\mathbf{x}) = w_{i,0} + w_{i,1}(x_1 - c_{i,1}) + w_{i,2}(x_2 - c_{i,2}) \ . \qquad (13)$$

Thus, a dataset can be characterized by $(p, (\mathbf{c}_i, \boldsymbol{\sigma}_i, n_i, \mathbf{w}_i)_{i=1}^{m})$, where p is the position of the dataset in the stream and m is the cluster number. For the whole stream, only a few datasets lying on so-called anchor positions are explicitly declared. The datasets lying between two anchor positions are interpolated by a linear function (i.e., the parameters $\mathbf{c}, \boldsymbol{\sigma}, n, \mathbf{w}$ characterizing a data set are convex combinations of the corresponding parameters of the left and right anchor datasets). The characteristics of the data stream used in our study are summarized in Table 1.

Table 1 Specification of the data stream. Cluster size of each cluster at any time is $n = 1000$. The cluster width is $\sigma_1 = 2$ and $\sigma_2 = 2$ for all clusters.

	C_1		C_2		C_3		C_1,C_3			C_2		
p	c_1	c_2	c_1	c_2	c_1	c_2	w_0	w_1	w_2	w_0	w_1	w_2
0	10	50	50	50	90	50	0	10	0	0	10	0
3	10	50	50	50	90	50	0	10	0	0	10	0
6	10	50	50	50	90	50	-10	0	0	10	0	0
12	10	50	50	50	90	50	-10	0	0	10	0	0
16	10	-100	50	150	90	-100	-10	0	0	10	0	0
26	10	-100	50	150	90	-100	-10	0	0	10	0	0
30	10	-100	50	150	90	-100	-10	0	10	10	0	10

The data stream contains three concept drifts with two stable episodes in-between, namely 6–12 and 16–26. These episodes allow the learner to re-stabilize the model. At the beginning, all three clusters are well separated and located next to each other along the first dimension. Moreover, the three regression functions share the same coefficients. Somewhat surprisingly, this initial configuration lets FLEXFIS generate more than only the expected three rules and leads to rich patterns during model formation.

The first drift is caused by a change of the regression values (between anchor positions 3 and 6, i.e., time steps 3000 and 6000). In Figure 4, this drift phase lies between marks A and B. At mark 1, a reaction of FLEXFIS is clearly seen. The second drift (between anchor positions 12 and 16) is caused by a movement of the clusters, which are all moving in different directions. The period of this drift is delimited by marks C and D. The visible reaction of FLEXFIS is between marks 2 and 3. Here, a change in the antecedent parts of the rules is seen, which confirms the expectation from the experimental design. The last concept drift (between anchor positions 26 and 30) is again caused by a drift in the regression values. It starts at mark E and ends at F. A change in the rule system caused by this drift can be seen from mark 4 onwards. After the third drift, the rule system does not reach a stable stage until the end of the experiment. All three concept drifts cause an adaptation of the rule system by the learning algorithm, resulting in clearly visible patterns in the rule chain visualization.

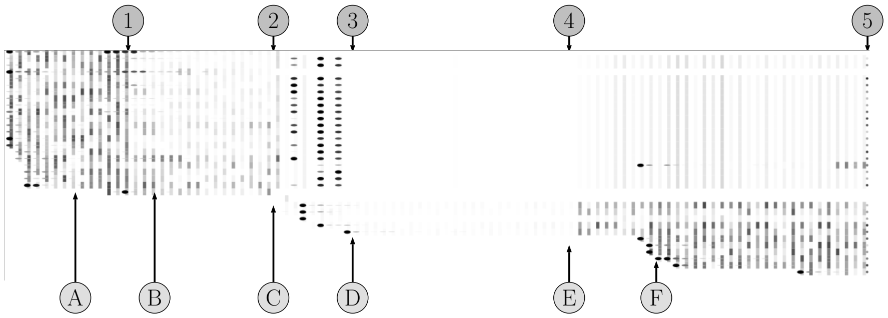

Fig. 4 This figure shows the result of a visualization of an evolving fuzzy rule-based system using stacked rule chains. There are three concept drifts in the underlying data stream, the first between A and B, the second between C and D, and the third between E and F. The numbers mark the time points where a change in the rule system, as a reaction to the concept drift, becomes visible.

5 Conclusion

We proposed stacked pearl chains for visually tracking the development of rules in evolving TSK fuzzy rule-based systems. Not only the life span of individual rules can be assessed by looking at the rule history, also demographic changes of the evolving rule system become visible. Temporal rule chaining is accomplished by visualizing similarities between adjacent rule antecedents and conclusions. Basically, the overlap between evolved rules or their center proximities is being displayed, as well as the angle between their conclusions.

With an example application using synthetic data, we highlighted several benefits of our visualization tool: the size of the rule system (number of rules), being a good indicator of model complexity, can be grasped quickly and monitored over time quite easily. Moreover, vertical patterns clearly indicate concept drifts and corresponding reactions of the evolving model.

Rule chain visualization as introduced in this paper is part of an ongoing software project for interactive monitoring of evolving TSK model evolution. This way, insights into model dynamics can be attained, which is interesting for both end-users who receive early feedback about setting model-specific parameters and developers who like to study convergence or adaptation properties of their models. Future work is focusing on the display of rule similarities and evolving rules at attribute level, global model state characteristics, and interactive control of evolving fuzzy system models. Eventually, our tool is supposed to provide the user access to complex states in time-adaptive machine learning methods beyond evolving TSK models.

Acknowledgments

This project is supported by the German Research Foundation (DFG). We also thank the LOEWE Center for Synthetic Microbiology (SYNMIKRO), Marburg, for financial support.

References

1. JM Alonso, O. Cordón, A. Quirin, and L. Magdalena. Analyzing interpretability of fuzzy rule-based systems by means of fuzzy inference-grams. In *World Congress on Soft Computing*, 2011.
2. T.R. Gabriel, K. Thiel, and M.R. Berthold. Rule visualization based on multi-dimensional scaling. In *Fuzzy Systems, 2006 IEEE International Conference on*, pages 66–71. IEEE, 2006.
3. J. Gama. A survey on learning from data streams: current and future trends. *Progress in Artificial Intelligence*, 1(1):45–55, 2012.
4. Daniel A. Keim, Jörn Kohlhammer, Geoffrey Ellis, and Florian Mansmann. *Mastering The Information Age - Solving Problems with Visual Analytics.* Eurographics, 2010.
5. E. Lughofer. FLEXFIS: A robust incremental learning approach for evolving Takagi–Sugeno fuzzy models. *Fuzzy Systems, IEEE Transactions on*, 16(6):1393–1410, 2008.
6. E. Lughofer and E. Hüllermeier. On-line redundancy deletion in evolving fuzzy regression models using a fuzzy inclusion measure. In S. Galichet, J. Montero, and G. Mauris, editors, *Proc. Eusflat–2011, 7th Int. Conf. of the European Soc. for Fuzzy Logic and Technology*, pages 380–387, 2011.
7. Edwin Lughofer, Jean-Luc Bouchot, and Ammar Shaker. On-line elimination of local redundancies in evolving fuzzy systems. *Evolving Systems*, 2(3):165–187, 2011.
8. Gabriele Peters, Kerstin Bunte, Marc Strickert, Michael Biehl, and Thomas Villmann. Visualization of processes in self-learning systems. *Tenth Annual International Conference on Privacy, Security and Trust (TSOS)*, pages 244–249, 2012.
9. F. Rehm, F. Klawonn, and R. Kruse. Rule classification visualization of high-dimensional data. In *Proc. of the 11th Int. Conf. on Information Processing and Management of Uncertainty in Knowledge-based Systems (IPMU 2006)*, 2006.
10. Moamar Sayed-Mouchaweh and Edwin Lughofer. *Learning in non-stationary environments.* Springer, 2012.