On Rule Generation Approaches for Genetic Fuzzy Systems

Marcos E. Cintra[1], Maria C. Monard[1], Heloisa A. Camargo[2]
Trevor P. Martin[3], Andrei Majidian[3]

[1]University of Sao Paulo (USP) – Mathematics and Computer Science Institute
CEP13.566-590, São Carlos - SP, Brazil

[2]Computer Science Department – Federal University of Sao Carlos
Rod. Washington Luís, Km 235, PO Box 676, 13.565-905, São Carlos - SP

[3]Department of Engineering Mathematics, University of Bristol, BS8 1UB, UK
BT Innovate and Design, Adastral Park, Ipswich, IP5 3RE, UK

{cintra, mcmonard}@icmc.usp.br, heloisa@dc.ufscar.br

trevor.martin@bris.ac.uk, andrei.majidian@bt.com

Abstract. *Genetic Fuzzy Systems have been researched for two decades and a considerable number of approaches have been proposed in the literature. Depending on the strategy used by the genetic algorithm, the generation of candidate rules is required to form the search space of the genetic algorithm. Specifically for the generation of these rules, proposals in the literature include the exhaustive generation of rules, the use of selection criteria over rules generated exhaustively, such as support, confidence, and degree of coverage, among others. This paper describes some of these methods and present their advantages and disadvantages in order to provide the reader with relevant information when deciding which method to use. A method for rule extraction with competitive advantages based on formal concept analysis is also proposed and preliminary results are discussed in detail. These results show evidence that our proposal is suitable for the task of forming a search space in terms of number of rules and processing time.*

1. Introduction

Fuzzy systems can be described as systems with variables based on the fuzzy logic. They have been successfully applied for the solution of problems in many areas, including pattern classification, optimization, and control of processes [Dumitrescu et al. 2000].

In this work, we focus on the genetic process of the automatic definition of fuzzy systems, specifically on those known as Rule Based Fuzzy Systems (RBFS), which usually have two main components: a Knowledge Base (KB) and an Inference Mechanism (IM). The KB comprises the Fuzzy Rule Base (FRB), *i.e.*, a set of fuzzy rules that represents a given problem, and the Fuzzy Data Base (FDB), which contains the definitions of the fuzzy sets related to the linguistic variables used in the FRB. The IM is responsible for carrying out the required computation that uses inferences to derive the output (or conclusion) of the system, based on both, the KB and the input to the system.

The special term Genetic Fuzzy System (GFS) was coined by the community to refer to fuzzy systems that use a genetic algorithm to create or adjust one or more of their components [Cordon et al. 2004]. Specifically, the classification of GFSs, according to [Herrera 2008], takes into account if the goal is: i) the genetic tuning of an existing knowledge base, or ii) the genetic learning of components of the KB. This work is focused on the genetic learning of the rule base.

Regarding the generation of rules to form the search space of a Genetic Algorithm (GA), in [Ishibuchi and Yamamoto 2004] the authors proposed an approach based on the rules confidence and support to preselect rules. A predefined number of rules with 0, 1, 2, and 3 antecedent conjunctions were generated for the wine dataset [Frank and Asuncion 2010] and used with a GA. A similar approach, named DOC-BASED, is proposed in [Cintra et al. 2007], in which, after the exhaustive generation of possible rules, a subset of them is selected to form the search space of a GA according to the degree of coverage of the rules. However, since the task of generating all possible rule combinations has exponential complexity, depending on the number of fuzzy variables and sets, the number of possible rules can be very large, interfering with the codification of the chromosomes, and overloading the whole genetic learning process. Thus, for datasets described by many features, a preselection of the most relevant ones is essential.

In this work, we describe some relevant approaches for the generation of the genetic search space proposed in the literature. We also present a proposal for the use of Formal Concept Analysis (FCA) as a competitive alternative to the described methods, including preliminary experiments and results that support our proposal. FCA is a theory on data analysis which identifies conceptual structures among data sets.

The remainder of this paper is described as follows. Section 2 presents the basic concepts of GFS. Section 3 introduces some of the existing approaches in the literature for the generation of the search space for a fuzzy genetic system. Section 4 introduces the theory of FCA and its use for the extraction of rules as well as a proposal for the use of FCA to extract fuzzy rules to be used as the search space of a GA, together with some preliminary experiments and results. The conclusions and future work are presented in Section 5.

2. Genetic Fuzzy Systems

Accordingly to [Herrera 2008], GFSs can be classified as:

1. Genetic tuning: if there exists a KB, a genetic tuning process for improving the RBFS performance is applied without changing the existing FRB;
2. Genetic learning: a component of the KB is learnt, either the FRB, FDB, or both. An adaptive inference mechanism can also be included in the process.

The genetic tuning can be further divided according to the taxonomy provided in Figure 1(a), extracted and adapted from [Herrera 2008], into: i) genetic tuning of KB parameters; ii) genetic adaptive inference engine. Similarly, the genetic learning can also be further divided into: i) genetic KB learning; and ii) genetic learning of DB components and inference engine parameters.

Our work is related to the genetic learning of the FRB, which is a part of the KB. Figure 1(b) provides the complete classification of the genetic KB learning, according to [Herrera 2008], which is also listed with some references next.

2.1 Genetic rule learning with a predefined FDB [Thrift 1991];

2.2 Genetic rule selection with a priori rule extraction [Cintra and Camargo 2007];
2.3 Genetic FDB learning [Cordon et al. 2001];
2.4 Simultaneous genetic learning of KB components [Homaifar and McCormick 1995].

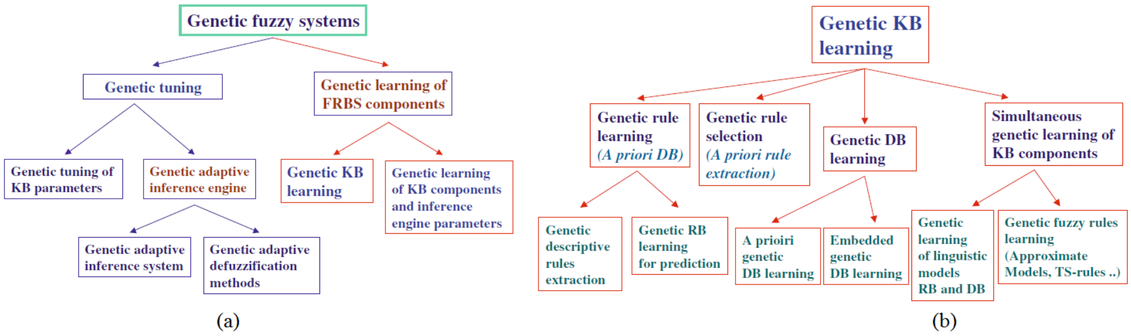

Figure 1. Genetic Fuzzy Systems Classification [Herrera 2008].

More specifically, our work focuses on methods to form the search space of a GA in order to build a GFS, which is part of item 2.2 from the previous list. Next Section surveys some of the key existing approaches for this task, *i.e.*, the generation of the search space for a GFS.

3. Methods for Search Space Rule Generation

In this section we describe some of the most frequently used methods in the literature for the generation of genetic search spaces, presenting and discussing their pros and cons.

3.1. Use of Heuristic Criteria

In [Ishibuchi and Murata 1995], the authors propose the generation of classification rules to form the search space of a GA using the following steps:

1. generation of all possible rule antecedent combinations;
2. calculation of the degree of certainty of each antecedent combination with each possible class using a set of training examples;
3. defining the consequent of each rule antecedent combination as the class with highest degree of certainty.

This approach suits low-dimensional domains, but it is not scalable to larger domains. Thus, in [Ishibuchi and Yamamoto 2004], the authors propose the extensive generation of rule antecedent and class calculation using the degree of certainty with preselection of candidate rules as a more feasible alternative to larger domains. The criteria used to preselect rules are the confidence and support measures, as defined in the data mining context for association rules.

Similarly to the approach in [Ishibuchi and Yamamoto 2004], in [Cintra et al. 2007] it is proposed the generation of the search space by extensively generating all possible rules combining all attributes, then calculating their degree of coverage as a criterium to preselect a subset of them. This way, all rules were composed by a combination of valid linguistic values for each attribute. The Degree of Coverage (DoC) was used as an indication of the classification power of these rules.

In both approaches described previously the main issue is related to the exponential complexity of the exhaustive generation of all possible rules. While these approaches are feasible for domains described by a small number of attributes combined with a reduced set of linguistic values for each attribute, they are not directly scalable for larger domains or when attributes are described by many linguistic values. Another issue is the computational cost required to calculate the degree of certainty and degree of coverage of these rules when combined with the substantial computational effort and processing time required by GAs.

Regarding the advantages of these methods, and others based on heuristic criteria, we can state the following:

1. Since these approaches preselect only relevant rules (according to criteria related to their classification power), the search space can be substantially reduced;
2. Because the rules are previously selected according to their classification power, the search process of the GA is reduced in terms of time;
3. With the preselection of rules, each rule in the search space has an index that is used in the chromosomes, thus, the chromosome codification is simplified (each position of the chromosome has an index to a rule or a special value to represent the absence of a rule). This way, the whole genetic process is optimized saving processing time;
4. Although the approach is not scalable to larger domains, or when a large number of fuzzy terms is used, one might argue that it is, nevertheless, totally feasible for a large number of real domains producing good results. Also, the use of a feature subset selection algorithm reduces the computational cost and time required by this approach, making it scalable to larger domains.

3.2. Association Rules Extraction

Another option for the generation of the rules to form the search space of a GA is to use an association rule extraction algorithm, such as Apriori [Agrawal and Srikant 1994]. Given a set of items (or attributes, for classification problems) and a set of representative examples, the idea is to find rules based on associated items. The support and confidence values of these rules are usually considered for their evaluation. Although association rules are not concerned with supervised domains, association rule algorithms can be used to extract associative classification rules, or simply classification rules, from a set of examples, by simply fixing the consequent as the class attribute .

In the literature it is possible to find proposals to generate fuzzy rule-based classifiers using fuzzy association rules, such as [Hu et al. 2002, Pach et al. 2008]. In [Hu et al. 2002], the authors propose the generation of large fuzzy grids from training examples by fuzzy partitioning each attribute; these grids are then use to generate fuzzy association rules for classification. In [Pach et al. 2008] the authors use the Fuzzy A Priori algorithm to search for frequent fuzzy item sets to form classification rules. The set of these rules is then pruned using the complexity, importance, and generality measures of the rules, forming a fuzzy classifier.

While approaches for the generation of classifiers using fuzzy association rules are abundant in the literature, to the best of our knowledge, the closest proposal using association rules and GAs to a GFS can be found in [Ishibuchi et al. 2006] in which the authors propose the use of association rule mining to define classifiers using a technique to search for Pareto-optimal rule sets. The authors first mine all possible classification rules using a minimum support and confidence values. These rules are then used by an evolutionary multiobjective optimization algorithm to search for Pareto-optimal rule sets. The three objectives used are: i) number of correctly classified training patterns; ii) number of selected rules (number of rules in the classifier); and iii) total number of antecedent conditions over the selected rules in the classifiers. It is important to notice that the rules generated in this paper are not fuzzy, but classic association rules.

An advantage of this approach based on association rules is related to the fact that the support of these rules can be used in a selection process, discarding rules with very low support, which would not help improve the final rule base, speeding up the genetic search process, and contributing to a better interpretability of the final FRB.

The disadvantages might include the fact that extracting all possible rules is an exponentially complex task. Furthermore, the number of attributes to be included in the extraction process must be defined previously and, thus, the total number of extracted rules might not be sufficient to form the search space, creating a dilemma: the number of combined attributes to be used *versus* the number of possible extracted rules.

Next, we briefly introduce the topic of Formal Concept Analysis (FCA) and present our proposed approach for the generation of the genetic search space using FCA.

4. Formal Concept Analysis

FCA is a mathematical technique for extracting concepts and structure from data introduced in [Wille 1982] which is becoming increasingly popular. Next, we present the basic definitions of FCA.

FCA transforms a formal context into a concept lattice. A formal context is a representation of the relation between objects and their attributes. The basic data structure in FCA is the context, which is normally represented in a table form where the columns represent the attributes and the rows represent the objects. The table contains 1 (true) in cell (i, j) if object i has attribute j, and 0 (false) otherwise. Formally, a context is a triple $k = (G, M, I)$, where G is a set of objects, M is a set of attributes, and I is a binary relation $I \subseteq G \times M$. Given a set of objects $A \subseteq G$, the shared image of A in M is defined as:

$$A^{\uparrow} := \{m \in M | (g, m) \in I \;\forall\; g \in A\} \tag{1}$$

Similarly, given a set of attributes $B \subseteq M$, the shared image of B in G is:

$$B^{\downarrow} := \{g \in G | (g, m) \in I \;\forall\; m \in B\} \tag{2}$$

The pair $(A, B) \in G \times M$ is a formal concept of (G, M, I) if and only if $A \subseteq G, B \subseteq M$ and $A = B \downarrow$, $B = A \uparrow$. A is called the extent of the concept and B is called the intent of the concept [Wille 1982]. In other words, Equation 1 defines the collection of all attributes shared by all objects from A, and Equation 2 defines the collection of all objects sharing all the attributes from B.

In traditional FCA, the relation is binary, although multi-valued contexts are much more common than binary-valued ones. For attributes that can take a range of values, the idea of "conceptual scaling" that transforms a many-valued attribute (e.g. a number) into a symbolic attribute can be used. For example, an attribute such as "height in centimetres", given an integer or real value between 0 and 200, could be transformed into attributes "height-less-than-50", "height-from-50-to-99", etc. These derived attributes have true/false values and can thus be treated within the FCA framework.

A toy example is illustrated in Table 1, which presents an attribute $\times$ value table with values for name, age, sex, and hair colour of six people.

Table 1. Toy example of a Formal Context.

Name	Age	Sex	Hair Colour
Andy	48	M	Black
Lina	29	F	Black
Mark	23	M	Brown
Martina	46	F	Blonde
Mike	18	M	Brown
Suzy	17	F	Blonde

In order to create the formal context, once FCA only admits binary attributes, attribute Age is discretized into three attributes, Age ≤ 20, > 20 & ≤ 30, and > 30. Since attributes Sex and Hair Colour already present nominal values, these values are used to create single attributes. Table 2 is the resulting table after the transformation.

Table 2. Scaled Version of the Previous Formal Context.

	Age			Sex		Hair Colour		
Name	≤ 20	> 20 & ≤ 30	> 30	Male	Female	Black	Blonde	Brown
Andy			⊛	⊛		⊛		
Lina		⊛			⊛	⊛		
Mark		⊛		⊛				⊛
Martina			⊛		⊛		⊛	
Mike	⊛			⊛				⊛
Suzy	⊛				⊛		⊛	

Using the formal context, it is possible to generate a conceptual lattice that presents the information in a nice visual way. Figure 2 shows the generated conceptual lattice of the information in Table 2. In the lattice structure, formal concepts are represented by the nodes. Attributes are noted slightly above nodes while objects are noted slightly under nodes. Notice that the positioning of the nodes can be arranged in a variety of ways. In the lattice presented in Figure 2, the nodes were arranged in order to minimize intersections, thus, attributes are not displayed in the order they are shown in the formal context. Another option would be to arrange the nodes respecting the order of the attributes or the order of the examples. In order to retrieve extensions, one must simply trace all paths leading down from the node. To retrieve intentions, on the other hand, one must trace all paths leading up from the node. For example, the intention of the formal concept represented by the node named *Mark* is: > 20 & ≤ 30, *Brown*, and *Male*. The extension of the formal concept represented by the node named *Blonde* is: *Martina* and *Suzy*.

As stated previously, the transformation of continuous attributes into binary ones is commonly called *scaling* by the FCA community. The fuzzy theory has also contributed

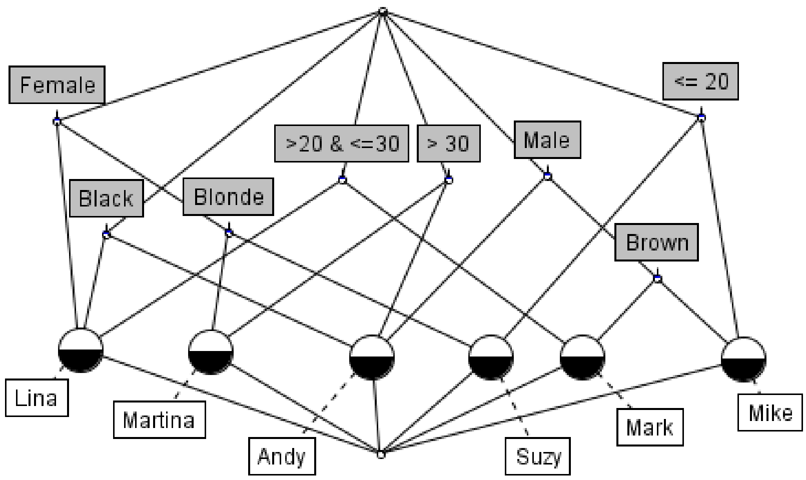

Figure 2. Conceptual lattice.

to this task. In [Wolff 2002] the authors present an introduction to the use of the fuzzy theory for the task of FCA scaling. Some of the advantages of using the fuzzy theory for FCA scaling include:

- the partitioning assume linguistic values, which are easily interpretable by humans, such as *young, old, tall, short*, etc. This way, the formal concepts extracted will convey this interpretability characteristic, which is highly desirable;
- unnatural divisions are avoided, such as the division in Table 2 for the 30 year-old people that are borderlines but must be place in only one category. The fuzzy logic easily avoids this problem with the use of the membership degrees that allow one object to belong to different categories with different degrees of membership;
- it is a natural choice when the interest in FCA is for the extraction of fuzzy rules, since these rules will reflect the fuzzy data base that will also be used by the inference mechanism of the induced classifier.

Next, we present a proposal on the use of FCA for the extraction of fuzzy rules.

4.1. Proposal for Extracting Fuzzy Rules from a Formal Context

As discussed in the previous sections, a formal context is the base for the extraction of formal concepts. These formal concepts can be seen as associations between attributes based on the existence of objects sharing these attributes. It is also important to notice that in the process of extracting formal concepts, their support is calculated automatically. In the literature it is possible to find various algorithms to extract formal concepts from a lattice. Worth mentioning are the NextClosure algorithm which works by finding neighbouring concepts [Ganter 2002], and the one proposed in [Krajca et al. 2010], which has a parallel search process.

Our proposal is based on the fuzzy definition of a problem in an attribute $\times$ value table to create a formal context and then obtain the classification rules. Considering a general dataset for classification purposes with n examples and m attributes, this fuzzy scaling procedure is performed with the following steps:

1. Define the fuzzy database, *i.e.*, the partitions that will define the fuzzification of the continuous attributes;
2. Create a binary attribute defined by each fuzzy set of each continuous attribute and each value of each discrete attribute;
3. Calculate the membership degree of the input values for each example in each binary attribute (notice that this step is only required for continuous attributes, once discrete ones will be automatically defined as true of false);
4. Define a minimum value A_{min} to guide the scaling of the real values so that if the membership degree of a certain value for a particular fuzzy set is equal or higher than A_{min}, the corresponding attribute is set to true in the formal context;
5. Use an algorithm to extract formal concepts to extract all existing classification rules from the formal context;
6. Define a minimum support value to select a subset of the extracted rules;
7. Use the final rule set as the search space of a GA to induce a fuzzy system.

Please, notice that due to space, the last step of the proposal, the use of a GA to induce a fuzzy system, is not explained in details here.

In order to reduce the number of possible formal concepts, the fuzzy sets defining each attribute can be evenly distributed in the partition, so the maximum possible membership degree in the intersections is 0.5. This way, if A_{min} is set to 0.5, for each original attribute only one of its binary attributes will be activated. Notice that ties must also be handled, thus, in our implementation we used a random variable to activate one of 2 tied fuzzy sets.

An important issue regarding the extraction of the formal concepts when using a binary fuzzification of the formal context is related to the increase in the number of attributes. This total number of attributes will be equal to the sum of all fuzzy sets describing each attribute, each value of each discrete attribute, and the number of classes. This increase in the number of attributes leads to an increase in the total number of formal concepts that can be extracted. Nevertheless, it is important to notice that for our purpose, the total number of formal concepts is much larger than the number of formal concepts we are interested in extracting: since we want to extract classification rules, we are only interested in extracting formal concepts that have a class in their intention.

Regarding the total number of formal concepts existing in a formal context, this number can be estimated using the Metropolis-Hastings algorithm for sampling formal concepts described in [Boley et al. 2010].

The process of extracting formal concepts and, consequently, classification rules can be done by adapting any algorithm for the extraction of formal concepts. In our experiments we used the NextClosure algorithm described in [Ganter 2002]. Since NextClosure can be used to extract formal concepts by analysing either the attributes or the objects, when it uses the attributes, it is called *Next Intent*, and it is called *Next Extent* when it uses the objects. This way, the word *element* is adopted instead of attributes or objects. Regarding the modification of the NextClosure algorithm to extract only formal concepts including a class, the process is quite direct: once the algorithm looks for neighbouring concepts, we just need to check whether a found concept has a class or not. If it does, then it is stored, otherwise, the process carries on looking for the next neighbouring concept.

Another interesting possibility for the restriction of formal concepts is the evaluation of their support, which, for a formal context, is just the number of objects a given formal concept shares divided by the number of all objects. This way, considering the balancing of the classes, formal concepts with low support could be discarded, reducing the number of extracted formal concepts.

4.2. Preliminary Experiments

Our main goals when carrying on the experiments were threefold: i) to check whether the resulting rule set would contain a suitable number of rules to form the search space of a GA; ii) to evaluate the time taken to extract the formal concepts; and iii) to evaluate the idea of using the support of the rules for a preselecting process.

To evaluate the proposed modification of the NextClosure algorithm we used 10 datasets from the UCI - Machine Learning Repository [Frank and Asuncion 2010] in order to analyse the number of concepts. Table 3 summarizes the dataset characteristics giving the total number of examples (Examples); total number of features (Features), including the number of continuous and discrete features in brackets; number of classes (Classes); the majority error (ME); and the number of fuzzy sets for each of the attributes (FS) predefined using the Fuzzy–DBD algorithm [Cintra et al. 2009]. Examples with missing values were removed from the datasets.

Table 3. General characteristics of the datasets.

Dataset	Examples	Features			Classes	ME	FS
Credit	653	15	6	9	2	45.33	2
Cylinder	277	32	19	13	2	35.74	2
Dermatology	358	34	33	1	6	68.99	2
Diabetes	769	8	8	0	2	34.90	2
Glass	220	9	9	0	7	65.46	7
Heart	270	13	13	0	2	44.44	2
Ionosphere	351	34	34	0	2	35.90	3
Iris	150	4	4	0	3	66.60	3
Vehicle	846	18	18	0	4	74.23	2
Wine	178	13	13	0	3	59.74	3

Table 4 presents the total number of formal concepts (TNFC), the total number of formal concepts with a class (FCwC), *i.e.*, the total number of classification rules, and the percentage it represents of the total number of formal concepts. In order to allow further comparisons, Table 4 also presents the total number of formal concepts for 50%, 20%, 10%, and 5% support values and the percentage they represent of the total number of formal concepts with a class (FCwC). The bigger the support, the smaller the number of rules. Support values greater than 50% were not considered as the number of rules was not enough to form the search space of a GA.

Regarding our first goal (the suitability of the rule set extracted in terms of number), we verified that an appropriate number of rules was extracted (FCwC). Our verification takes into consideration an estimated number of rules to populate 50 chromosomes,

Table 4. Extracted Formal Concepts Information

Dataset	TNFC	FCwC		50%		20%		10%		5%	
Credit	20,083	9,843	49.01	4	0.04	825	8.38	2,410	24.48	4,235	43.03
Cylinder	7,041,110	1,944,271	27.61	884	0.05	113,474	5.84	546,265	28.10	1,236,470	63.60
Dermatology	21,896,570	312,177	1.43	0	0.00	553	0.18	38,141	12.22	140,502	45.01
Diabetes	2,172	1,279	58.89	32	2.50	308	24.08	669	52.31	758	59.27
Glass	4,054	1,863	45.95	0	0.00	38	2.04	301	16.16	670	35.96
Heart	81,935	36,942	45.09	9	0.02	1,648	4.46	8,342	22.58	19,854	53.74
Ionosphere	102,641,179	2,076,229	2.02	12	0.00	1,187,827	57.21	1,197,307	57.67	1,649,406	79.44
Iris	93	65	69.89	0	0.00	11	16.92	27	41.54	41	63.08
Segmentation	10,785	1,437	13.32	0	0.00	0	0.00	162	11.27	54	3.76
Vehicle	86,918	28,979	33.34	0	0.00	91	0.31	1,780	6.14	6,625	22.86
Wine	21,000	9,802	46.68	0	0.00	423	4.32	3,338	34.05	6,676	68.11

which was the total population used in previous experiments with the DOC-BASED algorithm [Cintra et al. 2007]. Considering that the smallest rule set was obtained with the Iris dataset (60 rules), for this specific dataset, the total number of rules in each chromosome must be carefully defined in order to avoid lack of diversity in the population. The number or rules obtained for the remaining datasets can be considered appropriate.

Table 5 shows the time taken, in minutes, for the extraction of the formal concepts and calculation of their support. The process was executed in an Intel® Core™2 Duo T7250 (2.00GHz, 2MB L2 Cache, 800MHz FSB) machine. The time taken to extract the rules can also be considered appropriate for this approach to be used together with a genetic algorithm search process. The whole process took a matter of seconds to finish for all datasets but Cylinder, Dermatology, and Ionosphere, due to the total number of attributes and examples for these databases. It is also important to notice that our experiments were carried out with the NextClosure algorithm [Ganter 2002], but a faster algorithm was recently proposed which has a parallel search process [Krajca et al. 2010].

Table 5. Time (in minutes) taken to extract all formal concepts.

Dataset	Time	Dataset	Time
Credit	0.50	Heart	0.55
Cylinder	126.00	Ionosphere	144.63
Dermatology	168.73	Iris	0.02
Diabetes	0.05	Vehicle	1.90
Glass	0.12	Wine	0.18

If the support is taken into consideration for the selection of formal concepts including a class, it is possible to reduce even more the number of extracted formal concepts, giving the user more flexibility to decide on the number of extracted rules. The only issue one has to bear in mind when using the support in order to reduce the number of extracted rules is related to classes with few examples, which, in turn, will result in relative low support. In fact, as stated previously, in [Ishibuchi and Yamamoto 2004], the authors use the support, confidence, and the product of these two measures to select rules. These measures can also be investigated as future work.

Next, we present the conclusions and future work.

5. Conclusions and Future Work

The field of genetic fuzzy systems is the one with some of the most promising results in the area of fuzzy systems and many new approaches have been proposed in the literature. Specifically for the genetic definition of the rule base, a possible approach is the genetic rule selection, with the previous definition of the fuzzy database and the generation of fuzzy rules to form the search space of the genetic selection process. In this paper we presented some approaches found in the literature for this task of forming the search space of a genetic algorithm.

We also presented a new proposal and preliminary experiments and results on the use of formal concept analysis for this task. Formal concept analysis can be considered a new area of research, with applications in various domains and with an increasing interest due to its visual benefits and powerful mathematical basis.

The preliminary results show that our proposal is suitable for the task of forming the search space of a genetic algorithm in terms of the number of rules extracted, processing time, and also the use of the support measure to preselect a reduced number of rules if necessary.

As future work, we intend to adopt and use the DoC-BASED method, proposed in [Cintra et al. 2007, Cintra and Camargo 2007], to generate fuzzy rule bases using the proposal presented here to form the search space. This method includes the number of rules and the accuracy of the rule base in the fitness calculation in order to induce a rule base with high accuracy and interpretability rates. We intend to compare the formal concept extraction time taken by the NextClosure algorithm [Ganter 2002] and the parallel approach proposed in [Krajca et al. 2010]. We also intend to empirically evaluate the use of the support and confidence measures, and the product of them, to select rules.

Acknowledgements

The authors would like to thank CNPq - Conselho Nacional de Desenvolvimento Científico e Tecnológico and the São Paulo State Research Foundation - FAPESP for the financial support.

References

Agrawal, R. and Srikant, R. (1994). Fast algorithms for mining association rules in large databases. In *Proceedings of the 20th International Conference on Very Large Data Bases*, VLDB '94, pages 487–499, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Boley, M., Grosskreutz, H., and Gärtner, T. (2010). Formal concept sampling for counting and threshold–free local pattern mining. In *Proc. of the SIAM Int. Conf. on Data Mining (SDM 2010)*, pages 177–188. SIAM.

Cintra, M. E. and Camargo, H. A. (2007). Fuzzy rules generation using genetic algorithms with self-adaptive selection. *IEEE International Conference on Information Reuse and Integration - IRI*, pages 261–266.

Cintra, M. E., Camargo, H. A., and Martin, T. P. (2009). Optimising the fuzzy granulation of attribute domains. *International Fuzzy Systems Association World Conference*, pages 742–747.

Cintra, M. E., Camargo, H. A., R. Hruschka Jr, E., and Nicoletti, M. C. (2007). Fuzzy rule base generation through genetic algorithms and Bayesian classifiers - a comparative approach. *International Conference on Intelligent Systems Design and Application - ISDA*, 1:315–320.

Cordon, O., Gomide, F., Herrera, F., Hoffmann, F., and Magdalena, L. (2004). Special issue on genetic fuzzy systems.

Cordon, O., Herrera, F., and Villar, P. (2001). Generating the knowledge base of a fuzzy-based system by the genetic learning of the data base. *IEEE Transactions on Fuzzy Systems*, 9(4):667–674.

Dumitrescu, D., Lazzerini, B., and Jair, L. (2000). *Fuzzy Sets and Their Application to Clustering and Training*. Int. Series on Computational Intelligence. CBC Press.

Frank, A. and Asuncion, A. (2010). UCI machine learning repository.

Ganter, B. (2002). Formal concept analysis: Algorithmic aspects. Technical report, Technische Universitat Dresden.

Herrera, F. (2008). Genetic fuzzy systems: taxonomy, current research trends and prospects. *Evolutionary Intelligence*, 1(1):27–46.

Homaifar, A. and McCormick (1995). Simultaneous design of membership functions and rule sets for fuzzy controller using genetic algorithms. *IEEE Transactions on. Fuzzy Systems*, 3(2):129–139.

Hu, Y.-C., Chen, R.-S., and Tzeng, G.-H. (2002). Mining fuzzy association rules for classification problems. *Computing and Industrial Engineering*, 43:735–750.

Ishibuchi, H., Kuwajima, I., and Nojima, Y. (2006). Multiobjective association rule mining. In *Proc. of PPSN Workshop on Multiobjective Problem Solving from Nature*, pages 12–24.

Ishibuchi, H. and Murata, T. (1995). Selecting linguistic classification rules by two-objective genetic algorithms. In *Proceedings of the 1995 IEEE Int. Conf. on Systems, Man and Cybernetics*, pages 1410–1415. IEEE.

Ishibuchi, H. and Yamamoto, T. (2004). Fuzzy rule selection by multi-objective genetic local search algorithms and rule evaluation measures in data mining. *Fuzzy Sets and Systems*, 141:59–88.

Krajca, P., Outrata, J., and Vychodil, V. (2010). Parallel algorithm for computing fixpoints of galois connections. *Annals of Mathematics and Artificial Intelligence*, 59:257–272.

Pach, F. P., Gyenesei, A., and Abonyi, J. (2008). Compact fuzzy association rule-based classifier. *Expert Syst. Appl.*, pages 2406–2416.

Thrift, P. (1991). Fuzzy logic synthesis with genetic algorithms. In Belew, R. K. and Booker, L. B., editors, *Proceedings of the 4th International Conference on Genetic Algorithms*, pages 509–513. Morgan Kaufmann.

Wille, R. (1982). Restructuring lattice theory: An approach based on hierarchies of concepts. In *Ordered Setsand in I. Rivals (Ed.)*, volume 23, pages 445–470.

Wolff, K. E. (2002). Concepts in fuzzy scaling theory: order and granularity. *Fuzzy Sets and Systems*, 132(1):63–75.