Applied Soft Computing 37 (2015) 561–571

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

GPFIS-CLASS: A Genetic Fuzzy System based on Genetic Programming for classification problems

Adriano S. Koshiyama*, Marley M.B.R. Vellasco, Ricardo Tanscheit

Pontifical Catholic University of Rio de Janeiro, Rua Marquês de São Vicente, 255, Gávea, 38097 Rio de Janeiro, RJ, Brazil

A R T I C L E I N F O

Article history:
Received 28 May 2015
Received in revised form 2 August 2015
Accepted 31 August 2015
Available online 7 September 2015

Keywords:
Genetic Fuzzy System
Classification
Multi-Gene Genetic Programming

A B S T R A C T

Genetic Fuzzy Systems (GFSs) are models capable of integrating accuracy and high comprehensibility in their results. In the case of GFSs for classification, more emphasis has been given to improving the "Genetic" component instead of its "Fuzzy" counterpart. This paper focus on the Fuzzy Inference component to obtain a more accurate and interpretable system, presenting the so-called Genetic Programming Fuzzy Inference System for Classification (GPFIS-CLASS). This model is based on Multi-Gene Genetic Programming and aims to explore the elements of a Fuzzy Inference System. GPFIS-CLASS has the following features: (i) it builds fuzzy rules premises employing t-norm, t-conorm, negation and linguistic hedge operators; (ii) it associates to each rule premise a suitable consequent term; and (iii) it improves the aggregation process by using a weighted mean computed by restricted least squares. It has been evaluated in two sets of benchmarks, comprising a total of 45 datasets, and has been compared with eight different classifiers, six of them based on GFSs. The results obtained in both sets demonstrate that GPFIS-CLASS provides better results for most benchmark datasets.

1. Introduction

Genetic Fuzzy Systems (GFSs) [16,17,23,29] have been widely employed to solve classification [4,11,49], regression [2,5] and control [15,54] problems. The main feature that highlights GFSs in respect to other mathematical, statistical and artificial intelligence models is its capability of extracting knowledge from datasets or industrial plants and state it in linguistic terms with reasonable accuracy. This is provided by the bond between a Fuzzy Inference System (FIS) and a Genetic Based Meta-Heuristic (GBMH), which is based on Darwinian concepts of natural selection and genetic recombination. Therefore, a GFS provides fair accuracy and linguistic interpretation (FIS component) through the automatic learning of its parameters/rules (GBMH component), using information extracted from a dataset or a plant.

In GFSs literature, most works focus on developing or modifying methods in the GBMH component, such as:

- Modification of the Genetic Fuzzy Rule-Based Systems in the GBMH structure (codification, selection and evaluation) to generate fuzzy rule bases different from the standard Pittsburgh-style [29,36,44], such as Michigan [15,43], Genetic Cooperative-Competitive Learning (GCCL) [11,32,40,45] and Iterative Rule Learning (IRL) [18,26,28] approaches;
- Application of Multi-Objective Evolutionary Algorithms (MOEAs) to search for solutions that satisfy both accuracy and interpretability criteria in GFSs development [3,6,21,22,33];
- Employment of a GBMH to fine tune membership functions in the post-processing stage [49,50]; and
- Use of other Evolutionary Algorithms outside the GBMH scope (Particle Swarm and other Bio-inspired algorithms), in order to attain better results [13,25,37,42].

However, few researchers have focused on developments of the Fuzzy Inference component (e.g., operators such as negation, linguistic hedges, aggregation, etc.) to improve the performance obtained by the GFS. Additionally, there is a lack of works using Genetic Programming (GP) [39,47] as a GBMH for a GFS [16], despite its adequacy to problems that demand a non-fixed size codification – such as a fuzzy rule-based system. Therefore, to satisfy both conditions – FIS structure developments and application of Genetic Programming –, this work proposes a new GFS model: the Genetic Programming Fuzzy Inference System for Classification problems (GPFIS-CLASS). GPFIS-CLASS is based on Multi-Gene Genetic Programming (sometimes called Multi-Tree Genetic Programming) [24,52], a generalization of Koza-style GP [39,47] with

* Corresponding author. Tel.: +55 (21) 98571 4729.
E-mail address: as.koshiyama@gmail.com (A.S. Koshiyama).

http://dx.doi.org/10.1016/j.asoc.2015.08.055
1568-4946/

three main features: (i) it builds fuzzy rule premises by employing t-norm, t-conorm, negation and linguistic hedge operators; (ii)it associates a given premise to a suitable consequent term; and (iii) it improves the aggregation procedure by using operators other than the maximum t-conorm. GPFIS-CLASS should be seen as an improvement to the GPF-CLASS model [38]. This does not present reasonable results due to the lack of a suitable procedure for consequent term definition and membership degrees aggregation.

In order to assess its capability, GPFIS-CLASS is evaluated in two sets of experiments, extracted from Berlanga et al. [11] and Antonelli et al. [6] respectively. Briefly, the former [11] proposes a new GFS based on GP, in which each individual is encoded in a GCCL scheme for fuzzy rule base learning. This model was applied to 24 datasets and its results have been compared to those of four other GFSs, two of them based on GP. Antonelli et al. [6], on the other hand, presents a novel approach for learning concurrently the rule and data bases of fuzzy rule-based classifiers based on a multi-objective evolutionary approach. This system is evaluated in 24 datasets, mainly by comparing its results with those from two Evolutionary Fuzzy Systems (EFSs) and two other state-of-the-art classifiers. GPFIS-CLASS performance was then evaluated in most datasets; results have been compared to those provided by other GP-based and state-of-the-art EFSs models.

This paper has four additional sections. The next section describes the main concepts of Multi-Gene Genetic Programming. Section 2 presents the proposed GPFIS-CLASS model in five steps: fuzzification, fuzzy inference, decision, evaluation and selection & recombination. Section 4 deals with the model evaluation for different benchmark classification problems, and Section 5 concludes the work.

2. Multi-Gene Genetic Programming

Genetic Programming (GP) [39,47] belongs to the Evolutionary Computation field. Typically, it employs a population of individuals (or solutions), each of them denoted by a tree structure that codifies a mathematical equation, which describes the relationship between a set of input features X_j ($j = 1, \ldots, J$) and the output Y. Based on these ideas, Multi-Gene Genetic Programming (MGGP) [20,24,31,52] generalizes GP, as it denotes an individual as a set of tree structures, commonly called genes, that also receives the set of features X_j and tries to predict Y (Fig. 1).

Each individual is composed of D functions ($d = 1, \ldots, D$) that map X_j features to Y through user-defined mathematical operations. Those functions are composed of simple unary, binary or ternary operators, which have to satisfy some constraints detailed in [39]. It is easy to verify that, when $D = 1$, MGGP generates solutions similar to GP. In GP terminology, the X_j input features are included in the Terminal Set, while the mathematical operations (plus, minus, etc.) are inserted in the Function Set (or Mathematical Operations Set).

MGGP considers three recombination operators: a mutation and two different crossover operators. The mutation in MGGP is similar

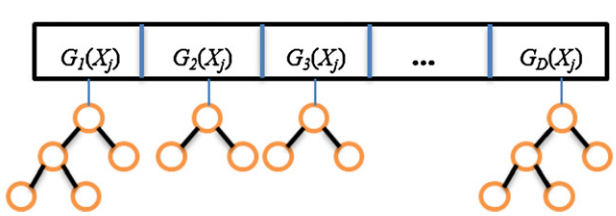

Fig. 1. Example of multi-gene individual.

to that in GP. As for crossover, the level at which the operation is performed must be specified: it is possible to apply crossover at high and low levels. The low level is the space where it is possible to manipulate the structures (Terminals and Mathematical Operations) of equations present in an individual. The high level, on the other hand, is the space where expressions can be manipulated in a macro way.

Fig. 2a presents a generic example of the application of a mutation operator to a multi-gene individual with five equations ($D = 5$), where the first tree has been randomly modified. Fig. 2b shows the low level crossover operation. As can be seen from Fig. 2(a) and (b), the mutation and low level crossover are similar to those performed in GP. The high-level crossover, on the other hand, differs from the standard GP crossover, as shown in the example presented in Fig. 2c. The dashed lines indicate that functions 2, 3 and 4 in individual 1 have been switched with functions 3, 4 and 5 from individual 2. The cutting point can be symmetric – the same number of equations is exchanged between individuals –, or asymmetric. Intuitively, high level crossover has a deeper effect on the output than low level crossover or mutation operators.

In general, the evolutionary process in MGGP differs from GP due to the addition of two parameters: maximum number of trees per individual and high level crossover rate. For the first parameter, a high value is usually employed in order not to obstruct the evolutionary process. On the other hand, the high level crossover rate, similar to other genetic operators rates, needs to be adjusted and its value is usually determined by experiments.

The next section presents the proposed GPFIS-CLASS model, which uses MGGP as its GBMH to evolve the premise term of each fuzzy rule in the FIS component.

3. GPFIS-CLASS model

GPFIS-CLASS is a typical Pittsburgh-type GFS [29], in which each individual represents a fuzzy rule base. Fig. 3 exhibits the main modules of the GPFIS-CLASS model.

Modeling begins by mapping crisp values into membership degrees of fuzzy sets (Fuzzification). Then, a fuzzy inference procedure is performed in three substeps: (i) ggeneration of fuzzy rule premises (Formulation); (ii) assignment of the best suited consequent term for each premise (Association) and (iii) aggregation of activated fuzzy rules (Aggregation). Finally, Decision, Evaluation and Selection & Recombination are performed. The following subsections provide a detailed description of each module of the GPFIS-CLASS model.

3.1. Fuzzification

In classification problems, the main source of information consists of a dataset containing n patterns $\mathbf{x}_i = [x_{i1}, x_{i2}, \ldots, x_{iJ}]$, each of them containing values of J features X_j ($i = 1, \ldots, n$ and $j = 1, \ldots, J$). The fuzzification step involves the association of fuzzy sets to each input feature. Therefore, each j-th feature has L associated fuzzy sets: $A_{lj} = \{(x_{ij}, \mu_{A_{lj}}(x_{ij})) | x_{ij} \in X_j\}$, where $\mu_{A_{lj}} : X_j \to [0, 1]$ is a membership function that assigns to each value x_{ij} a membership degree $\mu_{A_{lj}}(x_{ij})$ to the fuzzy set A_{lj} (see Fig. 4). Finally, each i-th pattern in the dataset belongs to a class C of K possible classes, i.e., $C \in \{1, 2, \ldots, k, \ldots, K\}$.

The specification of fuzzy sets involves the definition of three factors: (i) functional description (triangular, trapezoidal, etc.); (ii) support and granularity of membership functions $\mu_{A_{lj}}(x_{ij})$; and (iii) linguistic terms, in order to qualify the subspace defined by the membership function with an appropriate adjective. In theory, this should be specified by an expert. In practice, however, due to