

Available online at

<http://www.TJMCS.com>

The Journal of Mathematics and Computer Science Vol. 4 No.3 (2012) 502 - 513

## Uni-norm Fuzzy Pattern Trees for Evolving Classification by Imperialist Competitive Algorithm

S. Rajaeipour<sup>1</sup>, G. Shojatalab<sup>2</sup>

<sup>1</sup>Department of Industrial Engineering, Shomal University of Amol, Iran, saeideh,  
rajaeipour@shomal.ac.ir

<sup>2</sup>Department of Industrial Engineering, Shomal University of Amol, Iran,  
shojatalabg@yahoo.ir

Received: February 2012, Revised: May 2012

Online Publication: July 2012

### Abstract

Fuzzy pattern trees induction was recently introduced as a novel machine learning method for classification. Roughly speaking, a pattern tree is a hierarchical, tree-like structure, whose inner nodes are marked with generalized fuzzy logical or arithmetic operators and whose leaf nodes are associated with fuzzy predicates on input attributes. Operators perform an important role in fuzzy pattern trees. These operators include arithmetic and logical operators. Unlike arithmetic operators, logical operators that were used in these trees are not parameterized. As arithmetic operators, we can choose weighted arithmetic mean and ordered weighted arithmetic mean. There are several families which contain the standard triangular norms and conorms as special cases. This way, we would implicitly select from an infinite number of operators, just like in the case of arithmetic operators. We develop this algorithm by proposing a method to using parameterized logical operators and tuning their parameters by imperialist competitive algorithm. In experimental studies, we compare our method to previous version of algorithm, showing that our method is significantly outperforms the previous method in terms of predictive accuracy and flexibility in operator selection.

**Keywords:** machine learning, classification, fuzzy operators, parameter tuning

### 1. Introduction

Fuzzy pattern tree (FPT) induction was recently introduced as a novel machine learning method for classification by Huang et al [1]. Independently, the same type of model structure was proposed in [2] under the name “fuzzy operator tree”. An alternative to the original algorithm for learning pattern trees, as proposed in [3], was developed by Senge and Hüllermeier. Besides, an FPT variant for

regression was introduced in [4]. Roughly speaking, a fuzzy pattern tree is a hierarchical, tree-like structure, whose inner nodes are marked with generalized (fuzzy) logical and arithmetic operators. It implements a recursive function that maps a combination of attribute values, entered in the leaf nodes, to a number in the unit interval, produced as an output by the root of the tree. The model class of fuzzy pattern trees is interesting for several reasons. Apart from some properties that make it appealing from a learning point of view (like a built-in feature selection mechanism and the possibility to guarantee monotonicity in certain attributes), FPTs are arguably attractive from an interpretation point of view. Generally, each tree can be considered as a kind of (generalized) logical description of a class [3]. In this regard, pattern trees can be considered as a viable alternative to classical fuzzy rule models. Compared to such models, the hierarchical structure of pattern trees further allows for a more compact representation and for trading off accuracy against model simplicity in a seamless manner [4]. Already due to the using limited number of nonparametric operators in fuzzy pattern trees, a wide variety of operators for aggregation of fuzzy predicates actually was ignored and this could have the impact on the performance and results of fuzzy pattern trees. As arithmetic operators, we can choose weighted arithmetic mean and ordered weighted arithmetic mean [5, 6]. These operators are parameterized, which means that we implicitly select from an infinite number of operators but the logical operators, t-norms and t-conorms [7], used in fuzzy pattern trees are not parameterized. There are several fuzzy aggregation families which contain the standard t-norms (t-conorms) as special cases.

Because of using parametric operators to aggregation, we need the optimal parameters for them. Motivated by these developments, we propose a new version of FPTs that their operators are flexible and by tuning parameters of them we can reach a wide range of fuzzy aggregators. To determine the optimal parameters, there are techniques which identify optimal parameters of aggregation operators.

The remainder of the paper is organized as follows: Fuzzy pattern trees are explained in Section 2, in which we also recall the basic algorithm for learning such trees. Section 3 introduces a method for optimal selection of parameters in parametric operators to be using in FPTs. The new procedure is evaluated empirically in Section 4. The paper ends with a summary and some concluding remarks in Section 5.

## 2. Fuzzy Pattern Trees Induction

In this section, we briefly recall the pattern tree model for classification and the original algorithm for learning such models from data; for further technical details, we refer to [1]. As already mentioned earlier, a fuzzy pattern tree is a hierarchical, tree-like structure. The inner nodes of an FPT are marked with generalized (fuzzy) operators, either logical or arithmetic, whereas the leaf nodes are associated with fuzzy predicates on input attributes. A pattern tree propagates information from the leaf to the root node: a node takes the values of its descendants as input, combines them using the respective operator, and submits the output to its predecessor. Thus, a pattern tree implements a recursive mapping producing outputs in the unit interval [3].

### 2.1. Tree Structure and Model Components

We proceed from the common setting of supervised learning and assume an attribute-value representation of instances, which means that an instance is a vector

$$x \in \mathbb{X} = \mathbb{X}_1 \times \mathbb{X}_2 \times \dots \times \mathbb{X}_m ,$$

Where  $\mathbb{X}_i$  is the domain of the  $i$ th attribute  $A_i$ . Each domain  $\mathbb{X}_i$  is discretized by means of a fuzzy partition that consists of  $n_i$  fuzzy subsets

$$F_{i,j} : \mathbb{X}_i \rightarrow [0,1] (j = 1, \dots, n_j) \tag{1}$$

such that  $\sum_{j=1}^{n_j} F_{i,j}(x) > 0$  for all  $x \in X_i$ . The  $F_{i,j}$  are often associated with linguistic labels such as “small” or “large”, in which case they are also referred to as fuzzy terms. Each instance is associated with a class label

$$y \in Y = \{y_1, y_2, \dots, y_k\}$$

A training example is a tuple  $(x, y) \in X \times Y$ .

Unlike decision trees [8], which assume an input at the root node and output a class prediction at each leaf, pattern trees process information in the reverse direction. The input of a pattern tree is entered at the leaf nodes [1]. More specifically, a leaf node is labeled by an attribute  $A_i$  and a fuzzy subset  $F_{i,j}$  of the corresponding domain  $X_i$ . Given an instance  $x = (x_1, x_2, \dots, x_m) \in X$  as an input, the node produces  $F_{i,j}(x_i)$  as an output, that is, the degree of membership of  $x_i$  in  $F_{i,j}$ . This degree of membership is then propagated to the parent node [3].

Internal nodes are labeled by generalized logical or arithmetic operators, including t-norms and t-conorms are extensively used to model logical operators *AND* and *OR*. The basic t-norm and t-conorm pairs that operate on two fuzzy membership values  $a$  and  $b$ ,  $a, b \in [0, 1]$ , are shown in Table 1.

Table 1: basic t-norms and t-conorms pairs

Name	T-Norm	T-Conorm
Minimum/Maximum	$\min\{a, b\}$	$\max\{a, b\}$
Algebraic AND/OR	$ab$	$a + b - ab$
Lukasiewicz AND/OR	$\max\{a + b - 1, 0\}$	$\min\{a + b, 1\}$
Einstein AND/OR	$\frac{ab}{2-(a+b-ab)}$	$\frac{a+b}{1+ab}$

Also, weighted and ordered weighted average provide a wide spectrum ranging from very strict, conjunctive over averaging to compensatory, disjunctive aggregation that are shown in Table 2 [1].

Table 2: weighted and ordered weighted average operators

Name	Averaging Operator
Weighted Average	$\lambda a + (1 - \lambda)b$
Ordered Weighted Average	$\lambda \max\{a, b\} + (1 - \lambda) \min\{a, b\}$

The results of the evaluations of internal nodes are propagated to the parents of these nodes in a recursive way. The output eventually produced by a pattern tree is given by the output of its root node. Fig. 1 shows some exemplary pattern trees. For further technical details, we refer to [1].

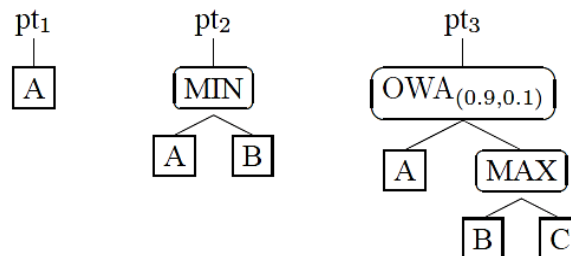


Fig. 1: exemplary pattern trees

A pattern tree classifier is a collection of pattern trees:

$$\{PT_i | i = 1, 2, \dots, k\},$$

where  $PT_i$  is the pattern tree associated with class  $y_i \in Y$ . Given a new instance  $x$  to be classified, a prediction is made in favor of the class whose tree produces the highest score:

$$\bar{y} = \arg \max_{y_i \in Y} PT_i(x) \tag{2}$$

A single tree  $PT_i$  can also be seen as a “fuzzy” selector of its class, hence the name pattern tree. Like a regular expression, a pattern tree selects instances belonging to its class, albeit in a fuzzy way. Eventually, the class is determined by the pattern tree that is most “confident” of being representative for the instance.

### 2.2. Learning Pattern Trees from Data

The use of the pattern tree model for classification requires an algorithm for learning such models from data, i.e., from a set of examples

$$T = \{(x^{(i)}, y^{(i)})\}_{i=1}^n \subset X \times Y$$

Following the original proposal of [1], pattern trees are built one by one, independently of each other. For each class, the induction method performs as follows:

- 1) Initialize with primitive pattern trees
- 2) Filter candidates by evaluation of their similarity to the target class
- 3) Check stopping criterion
- 4) Recombine candidates using fuzzy operators
- 5) Loop at step 2.

The basic algorithm for learning a pattern tree for binary classification is presented in pseudo-code in Fig. 2. It implements a beam search and maintains the  $B$  best models (trees) so far ( $B = 5$  is used as a default value). The algorithm starts by initializing the set of all primitive pattern trees  $P$ . A primitive tree is a tree that consists of only one node, labeled by a fuzzy term. Additionally, the first candidate set,  $C^0$ , is initialized by the  $B$  best primitive pattern trees, i.e., the trees being maximally similar to the target  $Y_0$  (see Equ.4)[3].

After initialization, the algorithm iterates over all candidate trees. Starting from line 11, it seeks to improve the currently selected candidate  $C_i^{t-1}$  in terms of performance. To this end, new candidates are created by tentatively replacing exactly one leaf node  $L$  (labeled by a fuzzy term) of  $C_i^{t-1}$  by a new subtree. This new subtree is a three-node pattern tree  $N = [L|\psi|R]$  that again contains  $L$  as one of its leaf nodes, now connected with another primitive tree  $R$  by means of an operator  $\psi$ . The new candidate tree is then evaluated by computing its performance. Having tried all possible replacements of all leaf nodes of the trees in  $C^t$ , the  $B$  best candidates are selected and passed to the next iteration, unless the termination criterion is fulfilled. More specifically, our algorithm stops if

$$Sim_{max}^t < (1 + \epsilon) Sim_{max}^{t-1} \tag{3}$$

i.e., if the relative improvement is smaller than  $\epsilon$ , where  $\epsilon = 0.0025$  by default [3].

To evaluate the performance of a pattern tree  $PT$ , we compare the output of our pattern tree for each training example  $(x^{(i)}, y^{(i)}) \in T$  to its respective target output [1].

$$Sim_{RMSE}(A_j, B) = 1 - \sqrt{\frac{\sum_{i=1}^n (A_j(x^{(i)}) - B(x^{(i)}))^2}{n}} \tag{4}$$

```

1: {Initialization}
2:  $P = \{A_{ij}\}, i = 1, \dots, n; j = 1, \dots, m$ 
3:  $C^0 = \arg \max_{P \in \mathcal{P}} [Sim(P, Y_0)]$ 
4:  $\varepsilon = 0.0025$ 
5:  $t = 0$  &  $T = 5$ 
6: {Induction}
7: {Loop on Iterations}
8: while true do
9:  $t = t + 1$ 
10:  $C^t = C^{t-1}$ 
11: {Loop on each candidate}
12: for all  $C_i^{t-1} \in C^{t-1}$  do
13:   {Loop on each leaf of the chosen candidate}
14:   for all  $l \in \text{leafs}(C_i^{t-1})$  do
15:     {Loop on each available operator  $\psi$ }
16:     for all  $\psi \in \Psi$  do
17:       {Loop on nearly each primitive pattern tree}
18:       for all  $P \in \mathcal{P} \setminus l$  do
19:          $C^t = C^t \cup \text{ReplaceLeaf}(C_i^{t-1}, l, \psi, P)$ 
20:       end for
21:     end for
22:   end for
23: end for
24:  $C^t = \arg \max_{C_i^t \in C^t} [Sim(C_i^t, Y_0)]$ 
25:  $Sim_{max}^t = \max_{C_i^t \in C^t} [Sim(C_i^t, Y_0)]$ 
26:  $Sim_{max}^{t-1} = \max_{C_i^{t-1} \in C^{t-1}} [Sim(C_i^{t-1}, Y_0)]$ 
27: if  $Sim_{max}^t < (1 + \varepsilon) Sim_{max}^{t-1}$  or  $t = T$  then
28:   break
29: end if
30: end while
31: return  $\arg \max_{C_i^t \in C^t} [Sim(C_i^t, Y_0)]$ 

```

Fig.2:pseudo-code of fuzzy pattern tree

### 3. Optimal Tuning of Parametric Operators

#### 3.1 Problem Formulation

Earlier, in fuzzy pattern trees we used nonparametric fuzzy logical operators, however a limited number of operators were used and a wide variety of fuzzy operators were not considered. The t-norm and t-conorm operators may be described as nonimpactive or impactive. A nonimpactive operator, such as Zadeh's [9] original union and intersection operators (Min and Max) is one where there is no interaction between grades of membership when they are combined. An impactive t-norm and t-conorm is one where a pair of weights is used to control the strength and weakness of the operators when combining membership grades. A number of impactive fuzzy inference operators based on the t-norm and t-conorm model have been proposed [10-12]. There are a number of factors which usually affect the selection of such operators. The first is the generic makeup of the domain. Continuous attributes are far more sensitive to fuzzification than discrete attributes. The second concerns how the operators are measured. These measurement properties have to be considered in relation with the scales of the membership functions used within the model. Other issues to consider are the accuracy of the fuzzy model, the degree of simplicity and the type of hardware implementation. This paper investigates a number of theoretical t-norm and t-conorms fuzzy inference operators within the context of fuzzy pattern trees.

Table 3: fuzzy prametric families

Family	t-norm	t-conorm	Ranges
Hamacher	$\frac{\mu_{\bar{A}}(x) \cdot \mu_{\bar{B}}(x)}{\gamma + (1 - \gamma)(\mu_{\bar{A}}(x) + \mu_{\bar{B}}(x) - \mu_{\bar{A}}(x) \cdot \mu_{\bar{B}}(x))}$	$\frac{(\gamma' - 1)\mu_{\bar{A}}(x) \cdot \mu_{\bar{B}}(x) + \mu_{\bar{A}}(x) + \mu_{\bar{B}}(x)}{1 + \gamma'\mu_{\bar{A}}(x) \cdot \mu_{\bar{B}}(x)}$	$\gamma \geq 0$ $\gamma \geq -1$
Yager	$1 - \min\{1, ((1 - \mu_{\bar{A}}(x))^p + (1 - \mu_{\bar{B}}(x))^p)^{1/p}\}$	$\min\{1, ((\mu_{\bar{A}}(x))^p + (\mu_{\bar{B}}(x))^p)^{1/p}\}$	$p \geq 1$
Dubois/ prade	$\frac{\mu_{\bar{A}}(x) \mu_{\bar{B}}(x)}{\max\{\mu_{\bar{A}}(x), \mu_{\bar{B}}(x), \alpha\}}$	$\frac{\mu_{\bar{A}}(x) + \mu_{\bar{B}}(x) - \mu_{\bar{A}}(x)\mu_{\bar{B}}(x) - \min\{\mu_{\bar{A}}(x), \mu_{\bar{B}}(x), (1 - \alpha)\}}{\max\{(1 - \mu_{\bar{A}}(x)), (1 - \mu_{\bar{B}}(x)), \alpha\}}$	$\alpha \in [0,1]$
Dombi	$\frac{1}{1 + \left(\left(\frac{1}{\mu_{\bar{A}}(x)} - 1\right)^w + \left(\frac{1}{\mu_{\bar{B}}(x)} - 1\right)^w\right)^{1/w}}$	$\frac{1}{1 + \left(\left(\frac{1}{\mu_{\bar{A}}(x)} - 1\right)^{-w} + \left(\frac{1}{\mu_{\bar{B}}(x)} - 1\right)^{-w}\right)^{-1/w}}$	$w \geq 1$

For the purposes of this paper, four fuzzy aggregation families of t-norm and t-conorm pairs (shown in table 3) have been selected to aggregate grades of membership generated by triangular membership functions for each attribute featured within the fuzzy pattern trees. These operators have been selected as a representative sample of common parameterized inference operators.

Zimmermann and Zysno[13]suggested the use of a linear combination of conjunctive and disjunctive operators

$$f(x_1, x_2) = w_1T(x_1, x_2) + w_2S(x_1, x_2) \tag{5}$$

$$w_1 + w_2 = 1 \quad w_1, w_2 \geq 0$$

This definition is consistent with the characteristics of uni-norms. The uni-norm is defined as a generalization of the t-norm and t-conorm. These operators allow for an identity element lying anywhere in the unit interval rather than at one or zero as in the case of t-norms and t-conorms, respectively. Various important properties of these uni-norms are investigated [23]. For having flexible and impactive operators that simultaneously are included of t-norm and t-conorm and average operator properties, we propose to use a convex combination of a t-norm and a t-conorm. Thus, the new operators are obtained from the operator families of table 3.

The t-norm and t-conorm that used in this new operator are parametric,so, for each one of these pairs there are two unknown parameters.Also, weight of each t-norm and t-conormin the convex combination of these operators add another parameter. So, we actually have three variables to be estimated.

Zimmermann [13]lists several criteria based on which aggregation operators can be selected. Among them, empirical fit is probably the most useful because it has a direct quantitative interpretation. In most cases, the problem of choice of the operator is translated into some sort of regression problem such as least squares fit. However, it is important to realize that fitting aggregation operators to data requires specialized regression techniques because of essential theoretical and semantical properties of these operators. The problem of fitting the parameters of aggregation operators to empiricaldata was examined by several authors [14-16].Consider the problem of fitting an aggregation operator  $f(x_1, \dots, x_n)$ to the empirical data. The empirical data consists of a list of pairs/triples/n tuples of membership values to be aggregated  $\{(x_1^k, \dots, x_n^k)\}_{k=1}^K$  and the corresponding compound membership values  $\{y^k\}_{k=1}^K$ , measured experimentally [17].

In our case, because of the binary classification, the aggregation operator is a2-place function  $f: [0,1]^2 \rightarrow [0,1]$ , nondecreasing in all arguments and satisfying  $f(\mathbf{0}) = 0$  and  $f(\mathbf{1}) = 1$ . This

function also is called general aggregation operator, and the foregoing properties constitute the minimal set of properties aggregation operators must satisfy. We will also require continuity of  $f$ , because this property is important from a practical point ofview. Additional properties define particular classes of aggregation operators[17].

Table 4: new uni-norm operators and their cost functions

Name	Operator	Cost function
Uninorm-Hamacher	$w_1 \left[ \frac{x_1 \cdot x_2}{\gamma + (1 - \gamma)(x_1 + x_2 - x_1 \cdot x_2)} \right] + w_2 \left[ \frac{(\gamma - 1)x_1 \cdot x_2 + x_1 + x_2}{1 + \gamma'x_1 \cdot x_2} \right]$	$\sum_{k=1}^K (w_1 \left[ \frac{x_1^k \cdot x_2^k}{\gamma + (1 - \gamma)(x_1^k + x_2^k - x_1^k \cdot x_2^k)} \right] + w_2 \left[ \frac{(\gamma' - 1)x_1^k \cdot x_2^k + x_1^k + x_2^k}{1 + \gamma'x_1^k \cdot x_2^k} \right] - Y^k)^2$
Uninorm-Yager		$\sum_{k=1}^K (w_1 \left[ 1 - \min (1, (1 - x_1^k)^{p_1} + (1 - x_2^k)^{p_1})^{\frac{1}{p_1}} \right] + w_2 \left[ \min (1, (x_1^k)^{p_2} + (x_2^k)^{p_2})^{\frac{1}{p_2}} \right] - Y^k)^2$
Uninorm-Dubois/Prade	$w_1 \left[ \frac{x_1 \cdot x_2}{\max\{x_1, x_2, \alpha_1\}} \right] + w_2 \left[ \frac{x_1 + x_2 - x_1 \cdot x_2 - \min\{x_1, x_2, (1 - \alpha_2)\}}{\max\{(1 - x_1), (1 - x_2), \alpha_2\}} \right]$	$\sum_{k=1}^K (w_1 \left[ \frac{x_1^k \cdot x_2^k}{\max\{x_1^k, x_2^k, \alpha_1\}} \right] + w_2 \left[ \frac{x_1^k + x_2^k - x_1^k \cdot x_2^k - \min\{x_1^k, x_2^k, (1 - \alpha_2)\}}{\max\{(1 - x_1^k), (1 - x_2^k), \alpha_2\}} \right] - Y^k)^2$
Uninorm-Dombi		$w_1 \sum_{k=1}^K \left( w_1 \left[ \frac{1}{1 + \left( \left( \frac{1}{x_1^k} - 1 \right)^{y_1} + \left( \frac{1}{x_2^k} - 1 \right)^{y_1} \right)^{1/y_1}} \right] + w_2 \left[ \frac{1}{1 + \left( \left( \frac{1}{x_1^k} - 1 \right)^{-y_2} + \left( \frac{1}{x_2^k} - 1 \right)^{-y_2} \right)^{-1/y_2}} \right] - Y^k \right)^2$

We formulate the problem as follows:

$$\text{minimize } \|f(x_1, x_2) - \mathbf{Y}\| \tag{6}$$

subject to  $f$  belonging to a given class of aggregation operators.  $\mathbf{Y}$  denotes the  $K$  vector of measured compound membership values[17].

The cost function used to seek the most optimal operator parameters and fuzzy spread around each tree node is determined by calculating and minimizing the difference between the amounts of operator and target data .The parameters of each operatorfor each fuzzy pattern tree is therefore determined using the following cost function.

$$\text{minimize } \sum_{k=1}^K (f(x_1^k, x_2^k) - Y^k)^2 \tag{7}$$

In this case, the problem has a nonlinear formula and also we face with some constraints on the problem. Multidimensional global optimization is a notoriously difficult problem and the answer cannot be simply achieved by regression methods[17]. The answer to that question can be found in another way. Evolutionary optimization algorithms are suitable option. There are many evolutionary optimization algorithms like genetic algorithm (GA) and ant colony algorithm and etc. In this case we use Imperialist Competitive Algorithm (ICA). This Algorithm is a new socio-politically motivated global search strategy that has recently been introduced for dealing with different optimization tasks[18]. This evolutionary optimization strategy has shown great performance in both convergence

rate and better global optima achievement [18]. New aggregation operators that we have proposed, can be created by uni-norm method and their cost functions can be seen in Table 4.

### 3.2 ICA Parameters

The search strategy of an ICA is defined by a number of optimization parameters. Initial experimental work was undertaken to determine suitable values for these parameters. Table 5 provides a summary of all ICA parameters used within evolutionary algorithm.

Table 5: Parameters used in the ICA algorithm

Variable	Value
Maximum Number of Iterations	100
Population Size	1000
Number of Empires/Imperialists	50
Selection Pressure	2
Assimilation Coefficient	2
Probability of Revolution	0.1
Revolution Rate	0.2
Colonies Mean Cost Coefficient	0.1

### 3.3 Optimization of parameter selection

For optimization of operator selection in fuzzy pattern trees, before the aggregation of two fuzzy membership degrees, the evolutionary optimization algorithm has been implemented. In this way, by optimizing the basic algorithm, three best parameters for the proposed operator are obtained. These parameters are located in operation and then membership degrees of each leaves of candidate trees and set of initial pattern trees are aggregated by this new operator. The proposed algorithm flowchart is shown in Fig 3.

## 4. Experiments

### 4.1 Data sets

For the purpose of this work, five real world data sets have been collected from the UCI [19] repository. The Iris data is a discrete outcome dataset consisting of 150 records and is used to detection of iris flowers. Data set of iris has 4 numeric attributes and 3 labels of classes. The cancer dataset, also with a discrete outcome, comprises of 699 records for classifying a given cell to benign or malignant cell types. This dataset has 9 numeric attributes. Glass data set investigates the type of a given glass and comprises of 214 records featuring 9 numeric attributes. The Crab data set is concerned with crab's sex (male or female) and has 1 nominal and 5 numerical attributes. Finally, wine data set comprises 178 records with 13 numeric valued attributes.



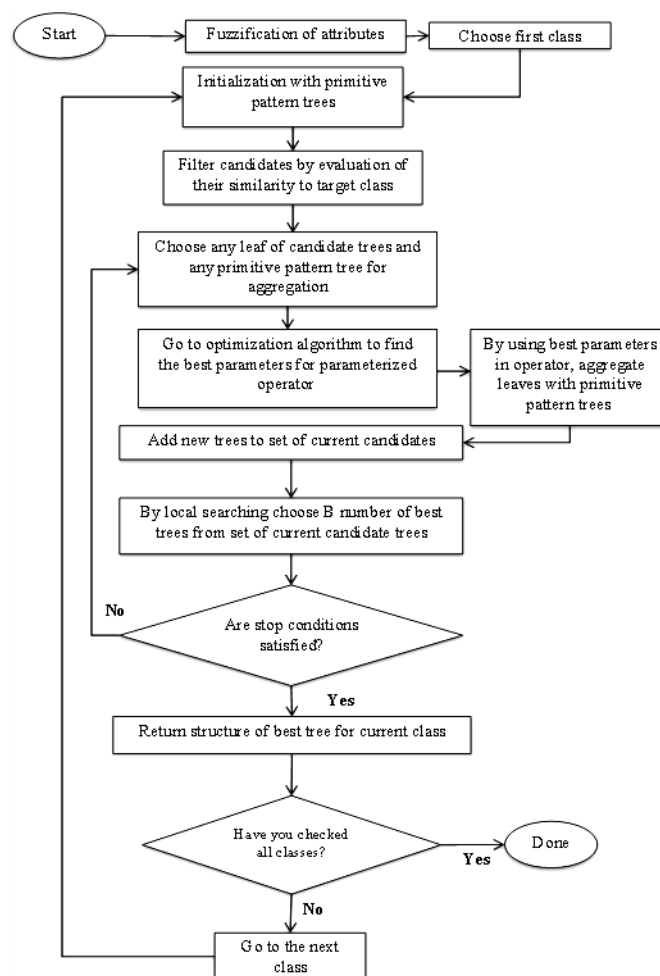


Fig.3:flowchart of proposed algorithm

#### 4.2 Method

We implemented new algorithms under the MATLAB 2010a [20]. This section presents a number of experimental studies that we conducted to get an idea of the performance of our optimized fuzzy pattern tree learner. We are interested in comparing our new algorithm with previous algorithm that was developed by Senge and Hüllermeier. The ICA algorithm merged with previous algorithm of fuzzy pattern tree and then for each proposed operators, the optimal parameters are found by ICA algorithm. Finally, the results for each of these new operators are investigated.

#### 4.3 Experimental Results

Fuzzy similarity values for each algorithm are shown in table 6 and Fig.4 shows the comparison between them. For each experiment set, stratified 10-fold cross validation was used. The training cases were partitioned into 10 equalized blocks with similar class distributions. In order to verify our proposed algorithm we use RMSE (root mean squared error), which has been obtained by averaging over 5 repetitions of a 10-fold cross validation (RMSE for iris dataset shown in Fig. 5). To test for statistical significance of the differences, and equalization of RMSE means of all algorithms (with their given operator), we used the Kruskal–Wallis test [21]. The test rejects the null hypothesis of equal performance at the 5% significance level (the p-value is 0.031). After that, we compared the methods in a pairwise way by means of a Post-Hoc Test [22]. The new algorithm together with Yager family has significant difference in performance among the other algorithms. But

pairwise comparison doesn't show significant difference between performances of previous algorithm with other new algorithms that have been developed by other parametric families. In other cases, the results show that other proposed method with Hamacher, Dubois/Prade and Dombi operators are fully competitive to previous algorithm in terms of predictive accuracy. Fig. 6 shows an example of a pattern tree for data set of wine that is developed by the new algorithm and Yager family parameters. The RMSE values for different algorithms is shown in Table 7, their ranking based on RMSE is indicated in parentheses.

Table 6: similarity values for each algorithm

Mean of similarities	Iris	Cancer	Glass	Crab	Wine	mean
Original	0.7577	0.8049	0.7356	0.5382	0.7094	0.70916
Uninorm -Hamacher	0.7301	0.8057	0.7567	0.5274	0.6865	0.70128
Uninorm - Yager	0.816	0.822	0.804	0.5547	0.8071	0.76076
Uninorm -DuboisPra	0.7604	0.8169	0.7239	0.5336	0.7302	0.713
Uninorm- Dombi	0.8058	0.7439	0.7448	0.5164	0.7316	0.7085

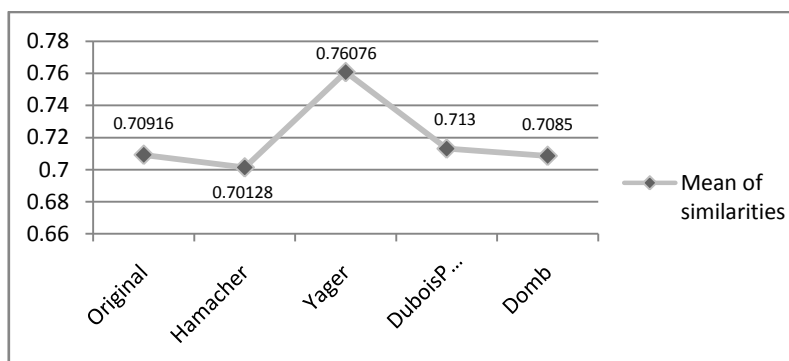


Fig 4: mean of similarity values for each algorithm

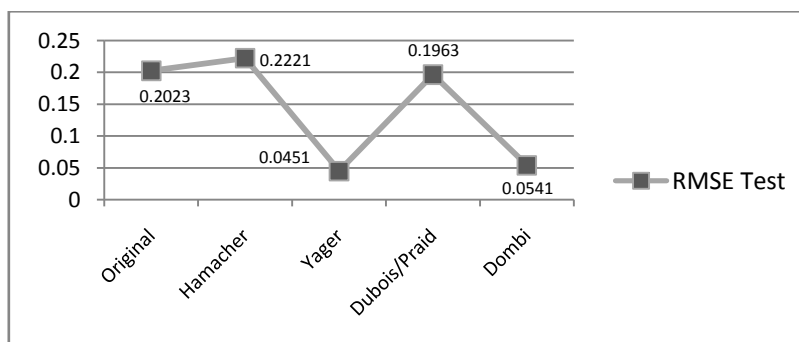


Fig 5: RMSE values for each algorithm for iris dataset

Table 7: RMSE values for different algorithms and their ranking

	Original	Hamacher	Yager	Dubois/prade	Dombi
<b>Iris</b>	0.2023(4)	0.2221(5)	0.0451(1)	0.1963(3)	0.0541(2)
<b>Cancer</b>	0.2186(4)	0.1771(3)	0.0389(1)	0.1498(2)	0.2954(5)
<b>Glass</b>	0.2671(4)	0.1721(2)	0.0693(1)	0.2706(5)	0.1947(3)
<b>Crab</b>	0.1430(2)	0.2053(4)	0.0622(1)	0.1799(3)	0.2408(5)
<b>Wine</b>	0.1925(4)	0.2485(5)	0.0696(1)	0.1588(3)	0.1189(2)
<b>Average rank</b>	3.6	3.8	1	3.2	3.4

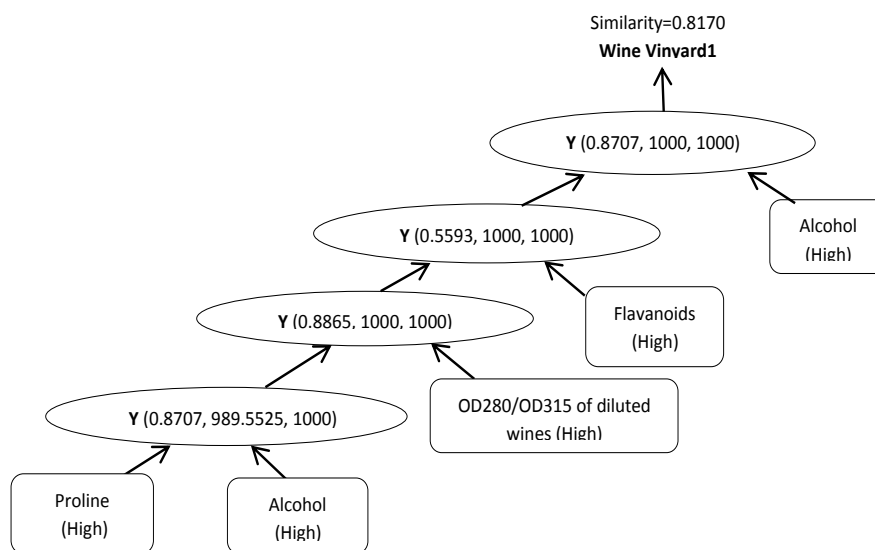


Fig. 5: an example of a pattern tree for data set of wine

## 5. Conclusions

This paper has investigated a number of theoretical proven parametric-t-norm and t-conorm-fuzzy inference operators and uses generalization of them as uni-norms within the context of fuzzy pattern trees and assessed their impact on the overall performance. A real encoded ICA was used to determine best parameters of uni-norm and simultaneously optimize strength controlling t-norm and t-conorm parameters for each specific operator.

Experimentally, we have shown that uni-norm fuzzy pattern tree (UFPT) with Yager family operators, significantly outperform the hitherto existing method in terms of predictive accuracy and, moreover, UFPT with other parametric family operators are at least competitive to previous algorithm.

In future work, we intend to generalize our current version of UFPT from binary to multi-class classification. Moreover, we are also interested in developing a uni-norm version of fuzzy pattern trees for regression.

## References:

1. *Pattern Trees Induction: A New Machine Learning Method*. Z. Huang, Tamas D. Gedeon, and Masoud Nikravesh. s.l. : IEEE TRANSACTIONS ON FUZZY SYSTEMS, 2008, Vols. 16, NO. 4. 1063-6706.
2. *Fuzzy operator trees for modeling rating functions*. Y. Yi, T. Fober, and E. Hullermeier. 4, s.l. : International Journal of Computational Intelligence and Applications, 2008, Vol. 8, pp. 413-428.
3. *Top-Down Induction of fuzzy Pattern Trees*. Senge, Robin and Hullermeier, Eyke. s.l. : IEEE Transactions On Fuzzy Systems, 2010, IEEE TRANSACTIONS ON FUZZY SYSTEMS.
4. *Pattern Trees for Regression and Fuzzy Systems Modeling*. Senge, Robin and Hullermeier, Eyke. Barcelona, Spain : World Congress on Computational, 2010. Proceedings WCCI 2010.
5. Sklar, B. Schweizer and A. Probabilistic Metric Spaces. North-Holland, New York : s.n., 1983.
6. *On ordered weighted averaging aggregation operators in multi criteria decision making*. Yager, R. 1, s.l. : IEEE Transactions on Systems, Man and Cybernetics, 1988, Vol. vol. 18, pp. 183-190.
7. *Triangular Norms*. E. P. Klement, R. Mesiar, and E. Pap. s.l. : Kluwer Academic Publishers, 2002.
8. *Decision trees and decision making*. Quinlan, J. R. 2, Man, Cybern. : IEEE Trans. Syst., Mar./Apr. 1990, Vol. 20, pp. 185-208.
9. *Fuzzy sets*. Zadeh, L. A. 3, s.l. : Inf. Control, 1965, Vol. 8. 338-353.
10. *A review of fuzzy set aggregation connectives*. Dubois D, Prade H. s.l. : Inform Sci, 1985, Vol. 36, pp. 85-121.
11. Klir G, Folger T. *Fuzzy sets, uncertainty, and information*. Singapore : Prentice Hall, 1992.

12. Li H-X, Yen VC. *Fuzzy sets and fuzzy decision-making*. Boca Raton : FL: CRC Press, 1995.
13. Zimmermann, Hans-Jürgen. *Fuzzy Set Theory and its Applications ,2ed edition*. Massachusetts : Kluwer Academic Publishers , 1992 .
14. *Latent connectives in human decision making*. Zimmermann H-J, Zysno P. s.l. : Fuzzy Set Syst, 1980, Vol. 4, pp. 37–51.
15. *Numerical construction of membership functions and aggregation operators from empirical data*. G, Beliakov. Paris : s.n., 2000. FUSION 2000. p. TuC4.
16. *Generalized means as model of compensative connectives*. Dyckhoff H, Pedrycz W. 1984, Fuzzy Set Syst, Vol. 14, pp. 143–154.
17. Beliakov, G. s.l. : International Journal Of Intelligent Systems, 2003, Vol. 18, pp. 903–923.
18. *Imperialist Competitive Algorithm: An Algorithm for Optimization Inspired by Imperialistic Competition*. Esmail Atashpaz Gargari & Caro Lucas. 2007. IEEE Congress on Evolutionary Computation (CEC 2007) . pp. 4661 - 4667.
19. Machine Learning Repository . [Online] UCI University. <http://archive.ics.uci.edu/ml/datasets.html>.
20. MATHWORKS. [Online] <http://www.mathworks.com/products/matlab/>.
21. *Use of ranks in one-criterion variance analysis*. Wallis, Kruskal and. 260, s.l. : American Statistical Association, Vol. 47, pp. 583–621.
22. *Pairwise multiple comparison procedures: A review*. Jaccard, J., Becker, M. A. and Wood, G. 3, s.l. : Psychological Bulletin, Vol. 96.
23. *Uninorm aggregation operators*. Yager, Ronald R. and Rybalov, Alexander, 1996, s.l. : Fuzzy Sets and Systems, Fuzzy Sets and Systems, pp. 111-120.