Genetic Programming for Optimizing Fuzzy Gradual Pattern Discovery

Sarra Ayouni[1,2] Sadok Ben Yahia[2] Anne Laurent[1] Pascal Poncelet[1]

[1]LIRMM - Univ. Montpellier 2 - FRANCE
[2]Faculty of Sciences of Tunis - TUNISIA

Abstract

Gradual patterns refer to frequent patterns describing correlations between variables which evolution is linked. For instance, the gradual pattern the older, the higher the salary means that the age and the salary increase/decrease simultaneously between two persons. This co-evolution can be found either as increasing together or evolving oppositely (e.g., the more cars, the less bus tickets). Several approaches were proposed to mine such patterns. These approaches differ depending on the way they count how frequent a pattern is, or depending on their efficiency both for memory and time consumption. The approaches can also differ depending on the way the attributes are treated, i.e. if they are considered as monotonically growing within the range of values or if they are considered as a fuzzy partition. For instance, the pattern the closer the age of an employee to 46, the higher his/her income is called to be a fuzzy gradual pattern. The challenge is then to retrieve the fuzzy sets (e.g. almost 46) that allow to mine the most relevant fuzzy gradual patterns. In fact, it is not an easy task to know in advance these appropriate fuzzy sets. In this paper, we focus on how genetic programming can be used in this context.

Keywords: Fuzzy Data Mining, Gradual Patterns, Genetic Programming.

1. Introduction

Fuzzy data mining has been extensively studied these last decades. It allows to cross scalable algorithms, such as frequent pattern mining, with approaches handling the imperfection of real world data. In this context, the literature covers varied topics such as fuzzy association rule mining [4], fuzzy sequential pattern mining [3], fuzzy graph and tree mining [13, 7], etc.

These approaches are very interesting when considering applications such as biology where the data are imperfect and where the treatment must be soften. In many cases, crisp comparisons may indeed lead to non informative, non relevant, or even false results. However, fuzzy sets can avoid some undesirable threshold effects by allowing "soft" rather than "harsh" boundaries of intervals. Furthermore, fuzzy association rules are very appealing from a knowledge representational point of view: fuzzy set theory features an interesting capability to bridge the gap between quantitative patterns and qualitative knowledge structures expressible in terms of natural language. Thus, association rules discovered in a database might be presented in a linguistic and, hence, comprehensible and user-friendly way.

On the other hand, mining for gradual patterns helps understanding the relations between attributes. The challenge is to retrieve all correlations that hold at a minimal ratio of the data. This kind of patterns is for instance the more gene x is expressed, the more gene y is expressed. It has been shown that properties like monotonicity hold on these patterns. Even if the task of mining such patterns is complex, very efficient algorithms can be designed, especially when considering multi-core architectures [15].

However, these patterns only hold when the correlation is linearly set on the full range of data values, while it could be interesting to point out that some values attract a particular knowledge. In this respect, we consider the problem of extracting gradual patterns like the more gene x is expressed at rate *almost* r, the more gene y is expressed at rate *almost* s. The value representing the rate is considered as being a soft interval in order to convey the idea of a fuzzy region. By considering these kinds of fuzzy intervals, it is then possible to extract patterns like the more the age of a woman is within the range $[25 - 35]$, the higher the fertility.

In recent work, we have proposed to mine fuzzy gradual patterns [14]. In this paper, we propose to extend this approach by defining how genetic algorithms may help focus on the best intervals to describe such patterns. Indeed, we discuss how genetic algorithms must be parameterized in order to extract the best fuzzy gradual patterns that can be hidden in quantitatives datasets. In particular, we discuss the encoding mechanism, the fitness function and the genetic operators that will be applied in the proposed genetic process.

The paper is organized as follows. Section 2 recalls the basics of fuzzy gradual patterns, while Section 3 presents our proposition on how to define the steps of the genetic algorithm for extracting appropriate fuzzy intervals leading to relevant fuzzy grad-

ual patterns. Section 4 concludes and discusses the main perspectives of this work.

2. Fuzzy Gradual Patterns

We consider a dataset having the schema (X_1, ..., X_m) and containing a set of rows (m-tuples) in $\text{dom}(X_1)\times$, ..., $\times$ $\text{dom}(X_m)$.

The problem of mining for frequent gradual patterns has been studied from varied points of view [1, 5, 6, 9, 10, 11, 2]. Such patterns are sets of gradual items.

A gradual item is defined as a pair of an attribute and a variation $* \in \{\leq, \geq\}$. Let A be an attribute, then the gradual item $(A, \geq)$ means that the attribute A is increasing.

A gradual pattern or gradual itemset is defined as a non-empty set of several gradual items. The main point is then to study the methods to define the extent to which a gradual pattern holds. Gradual rules have also been described in the literature, requiring causality to be explicitly defined.

According to [?], the gradual dependence $A \Rightarrow B$ holds in a database $\mathcal{D}$ if $\forall$ o=(x, y) and o'=(x', y') $\in \mathcal{D}$, $A(x) < A(x')$ implies $B(y) < B(y')$. It should be noted that this definition does not take into account any comparison between data rows.

A new definition of gradual dependencies was proposed in [12] using fuzzy association rules. The authors take into account the variation strength in the degree of fulfilment of an imprecise property by different objects. Hence, a gradual dependency holds in a database $\mathcal{D}$ if $\forall$ o=(x, y) and o'=(x', y') $\in \mathcal{D}$, $v_{*1}(A(x), A(x'))$ implies $v_{*2}(B(y), B(y'))$, where v_* is a variation degree of an attribute between two different rows. In both propositions [?] and [12], the authors propose to build a modified data set $\mathcal{D}'$ that contains as many rows as there are pairs of distinct objects in the initial data set $\mathcal{D}$ in order to identify such gradual dependencies.

Another definition was proposed in [5]. As pointed out in the introduction, the support of a gradual itemset $A_1^{*_1}, \ldots, A_p^{*_p}$ represents the extent to which the gradual itemset holds within the dataset.

The support can be defined as the maximal number of rows $\{r_1, \ldots, r_l\}$ for which there exists a permutation π such that $\forall$ $j \in [1, l-1]$, $\forall$ $k \in [1, p]$, it holds $A_k(r_{\pi_j}) *_k A_k(r_{\pi_{j+1}})$. More formally, denoting $\mathcal{L}$ the set of all such sets of rows the support of a gradual itemset is defined as follows. Let s=$A_1^{*_1}, \ldots, A_p^{*_p}$ be a gradual itemset, we have:

$$supp(s) = \frac{\max_{L_i \in \mathcal{L}} |L_i|}{|\mathcal{D}|}$$

The authors propose a heuristic to compute this support for gradual itemsets, in a levelwise process that considers itemsets of increasing lengths. The authors in [6] propose the Grite method [1] which is an efficient method based on the precedence graph and binary matrices for data representation.

In [11], the authors proposed a novel definition of the support of gradual patterns based on rank correlation (concordant and discordant pairs of rows from the data). The authors use the same algorithm and data binary representation as cited above.

In the case of fuzzy data, the X_i attributes are fuzzy linguistic variables associated to linguistic values defined through fuzzy sets. For instance, let us consider an attribute corresponding to the speed of vehicles. In the crisp case, it contains the numerical values of the measured speeds. Whereas, in the fuzzy case, it will be associated to linguistic variables, e.g., “slow", “normal", and “fast". These linguistic variables or modalities are defined by their membership degrees (according to a membership function) indicating the extent to which their speeds belong to each modality.

It can be interpreted by “the more A". Here, the attribute A can be crisp or fuzzy. For instance (Speed, $\geq$) is a crisp gradual item, as well as (Fast Speed, $\geq$) which is a fuzzy gradual item.

In [14], one fuzzy modality (either triangular, trapezoidal or gaussian) is built for every attribute, based on the median or average value from the domain. However, this does not guarantee that the fuzzy set extracted from this method is appropriate for extracting relevant fuzzy gradual patterns. Indeed fuzzy gradual patterns can be hidden some where between the two extremes of the minimum and maximum values of attribute without being all around the median/average value. We aim in this paper at automatically finding the most appropriate fuzzy membership functions ensuring the extraction of relevant fuzzy gradual patterns. The automatic definition of these fuzzy modalities can be considered as an optimization problem. Evolutionary algorithms and particularly Genetic algorithms have been proven to be efficient for solving such problems.We thus propose to consider genetic programming in order to enhance the definition of the fuzzy sets.

3. Genetic Algorithms: Application to Fuzzy Gradual Patterns

Genetic algorithms requires a starting population containing individuals described by their chromosomes. At each step of the algorithm, some genetic operations (i.e., evaluation, selection, crossover and mutation) are applied on these individuals and a new population will be created in order to enhance its ability to fit a fitness function.

In this section, we discuss how genetic algorithms can help extracting some of the most relevant fuzzy gradual patterns. We thus introduce a representation of what could be a chromosome in this context,

[1] Grite stands for GRadual ITemset Extraction.

and how they could be mixed when a new population is built.

In our context, one individual is one fuzzy gradual itemset. We aim at enhancing a set of frequent fuzzy gradual patterns that could be output to the user.

3.1. Individual Representation

Every gradual pattern is defined over the set of attributes X_1, ..., X_i, ..., X_m. For the sake of simplicity, we focus here on trapezoidal functions, but our approach would be the same on triangular, gaussian or other shapes. These shapes can be easily mixed within chromosomes without changing our proposition.

Every function is thus given by a 4-tuple corresponding to the four values a_i, b_i, c_i, d_i of the limits of the intervals for the kernel and support of the fuzzy subset. Note that the fuzzy membership function is not necessarily symmetric.

An individual is a set of chromosomes representing m fuzzy itemsets, where the fuzzy sets correspond to 4-tuples fuzzy trapezoidal functions defined over the m attributes and where there are two possible variations corresponding to increasing or decreasing ($* \in \{\leq, \geq\}$).

If an attribute does not occur in the gradual pattern, then the corresponding slot in the chromosome is set to empty, as shown on Figures 1 and 2.

3.2. Initial population

The first individuals of the initial population can be generated by considering the method proposed in [14]. Indeed, each chromosome of the individual represents the whole domain of the corresponding attribute. The following individuals encode fuzzy membership functions obtained by random lateral variation of previous ones. This ensures to start with enough diversity in the initial population. The size of the population is one of the parameters of the system. We denote it by σ.

3.3. Genetic operators

At each step of the algorithm, the population is mixed in order to build a new (better) one. The chromosomes of different individuals of the population are crossed in order to mix the fuzzy membership functions and to try and extract better patterns. The crossover operator consists in randomly taking two individuals, called parents, and generating new individuals : for each chromosome the parameters of the corresponding membership function is either inherited from one of the parents.

The mutation operator is a mechanism that guarantees the diversity of the population. Mutation can be used in order to change one individual. Such a mutation will randomly affect some attribute values w.r.t a mutation rate MR. The change is operated on the membership function definition, by considering a shift, or a transformation of the shape (e.g., stretching). The change can also be operated on the variation direction, switching from an increasing to decreasing variation, or conversely.

It must however be computed by keeping some constraints on. For instance when considering trapezoidal functions, the following property must always hold for every attribute i: $a_i \leq b_i \leq c_i \leq d_i$.

As stated above, every individual corresponds to a fuzzy gradual pattern. Starting from it, it is thus possible to retrieve within the dataset all the data matching it. Depending on the size of the data matching it, the support will be high or low.

As we consider that it is better to consider few gradual patterns, but whose support is high, rather than keeping many individuals. Some individuals may thus be discarded, corresponding to gradual patterns having a very low support.

3.4. Fitness Function

Defining the fitness function is difficult as there is no unique solution to the problem we explore, as defining what is a relevant and optimal gradual pattern is impossible.

We argue in this paper that a fuzzy gradual pattern is relevant if its support is high and its length (number of attributes) is high as well. A population of fuzzy gradual patterns is thus interesting if the gradual patterns it contains fit these criteria.

Let us denote the population by $\mathcal{I} = \{I_1, \ldots, I_\sigma\}$. Each individual I_i corresponds to a gradual pattern whose support is denoted by $support(I_i)$. We denote by $Length(I_i)$ the number of non-empty gradual itemsets within I_i.

We consider a local fitness for every individual defined as:

$$L\_Fitness(I_i) = -support(I_i) * log\left(\frac{1}{Length(I_i)}\right)$$

This definition is used to evaluate the individuals of the population by discarding those having too low fitness regarding the other ones or regarding the target.

The global fitness of a population is then defined as the average of the local fitness values.

$$G\_Fitness(\mathcal{I}) = \left(\frac{1}{\sigma}\right) * \sum_{i=1}^{\sigma} L\_Fitness(I_i)$$

The global fitness will be used as a stopping condition of the algorithm. In fact, the loop of chromosome generations is terminated when certain conditions are met. In our proposal there are two criteria either the number of generations has reached a maximum number ($NbrGen$), or if the global fitness has not changed for a certain number of generations.

X_1	...	X_i	...	X_m
$\emptyset$ or $(a_1, b_1, c_1, d_1), *$	...	$\emptyset$ or $(a_i, b_i, c_i, d_i), *$	...	$\emptyset$ or $(a_m, b_m, c_m, d_m), *$

Table 1: Individual - General Case

X_1	...	X_i	...	X_m
$\emptyset$	...	$(25, 26, 43, 58), \leq$	...	$(5, 12, 24, 43), \geq$

Table 2: Individual - Example

X_1	...	X_5	...	X_m
$(25, 26, 43, 58), \leq$	...	$\emptyset$	...	$(5, 12, 24, 43), \geq$

Crossed With

X_1	...	X_5	...	X_m
$\emptyset$	...	$(14, 16, 23, 38), \geq$	...	$(38, 43, 72, 87), \geq$

After X_5, leading to

X_1	...	X_5	...	X_m
$(25, 26, 43, 58), \leq$	...	$\emptyset$	...	$(38, 43, 72, 87), \geq$

And

X_1	...	X_i	...	X_m
$\emptyset$	...	$(14, 16, 23, 38), \geq$	...	$(5, 12, 24, 43), \geq$

Table 3: Crossing Chromosomes

3.5. Algorithm

According to the above description, the proposed algorithm for mining both fuzzy modalities and fuzzy gradual patterns is described in algorithm 1. The inputs are the dataset (D), the minimum support ($MinSup$), the population size ($PopSize$), the number of trials or generations ($NbrGen$), the fraction of individuals that will be crossed over (CF), the mutation rate (MR).

The pseudo-code and the notations used in our approach are, respectively presented in table 4 and algorithm 1.

FGPs	: Set of Fuzzy Gradual patterns.
FMs	: Set of Fuzzy membership functions.
D	: The input dataset.
PopSize	: The size of each generated population.
NBrGen	: Number of generations.
CF	: The fraction of individuals to be crossed over.
MR	: The mutation rate.

Table 4: Notations used in the algorithm

3.6. Experiments

Our approach has been tested on a real dataset which describes data about wine [8] (13 columns) where the class attribute has been removed. We have set the default parameters of the genetic algorithm to 100 individuals, 50 generations, 40 of mutations and 50 of crossovers. They have been chosen experimentally so as to produce non redondant individuals from a run of the system to another one. For a $MinSup$ value equal to 80% we have extracted 467 FGPs having an average size equal to $9,28$. And for a $MinSup$ set to 100% we have extracted 335 FGPs.

4. Conclusion and Perspectives

In this paper, we propose a new genetic-based approach for retrieving relevant fuzzy sets in order to mine fuzzy gradual patterns. We discuss how genetic programming can help, how the main genetic operators can be defined and how relevance must be set in order to extract relevant patterns. Our approach has the merit to be independent or not of the human expert's intervention during the mining process

Many perspectives are linked to this work. Other methods and statistics may be used. The goal may also be changed in order, for instance, to build the

Algorithm 1: Genetic-based process for mining FGPs

Input: D, $MinSup$, $PopSize$, $NbrGen$, CF, MR, $OptFitness$.

Output: FGPs, FMs.
begin

Step1: Generate the initial population of size $PopSize$ at random

Step2: Evaluate each individual I_i in the population Pop as follows:

foreach ($I_i \in Pop$) do
- Transform the quantitative values of every attribute in D into fuzzy sets using the corresponding fuzzy modality represented by the chromosomes of the individual.

- Calculate the $support(I_i)$ w.r.t the $MinSup$ value by counting the tuples in D matching the FGP_i represented by I_i.

- Set the $L\_Fitness(I_i)$ to $(-support(I_i) * log\left(\frac{1}{Length(I_i)}\right))$.

- Increment the the global fitness $G\_Fitness$ of the population Pop.

Step 3: Create a new population as follows:

- Apply the crossover operator on the CF fraction of individuals of the previous population.
- Apply the mutation operator on MR individuals of the population.
- Insert the new individuals into the population.
- Select the best individuals in the population (by the elitism mechanism).

Step4: Go to Step2 to evaluate the population.

Step5: Repeat Step3 and Step4 until satisfying the termination criterion which is either reaching the $NbrGen$ or global fitness has not changed for a certain number of generation.

Step6: Gather the sets of the best individuals in $FGPs$ and the corresponding membership functions in FMs

end

fuzzy membership function that allows to retrieve the highest number of rules, or the most discriminant rules for classifying data. Moreover, We will also explore how to work on a whole fuzzy partition instead of focusing on a single fuzzy membership function on every attribute.

References

[1] F. Berzal, J.-C. Cubero, D. Sanchez, M.-A. Vila, and J. M. Serrano. An alternative approach to discover gradual dependencies. Int. Journal of Uncertainty, Fuzziness and Knowledge-Based Systems (IJUFKS), 15(5):559–570, 2007.

[2] B. Bouchon-Meunier, A. Laurent, M.-J. Lesot, and M. Rifqi. Strengthening fuzzy gradual rules through "all the more" clauses. In Proceedings of IEEE World Congress on Computational Intelligence - WCCI'2010, Barcelona, Spain, 2010.

[3] A. Laurent C. Fiot and M. Teisseire. From crispness to fuzziness: Three algorithms for soft sequential pattern mining. International Journal IEEE Transactions on Fuzzy Systems, 2007.

[4] A.W. Fu C.M. Kuok and M.H. Wong. Mining fuzzy association rules in databases. SIGMOD Record, 27(1):41–46, 1998.

[5] L. Di Jorio, A. Laurent, and M. Teisseire. Fast extraction of gradual association rules: A heuristic based method. In Proceedings of the IEEE/ACM Int. Conf. on Soft computing as Transdisciplinary Science and Technology, CSTST'08, Cergy, France, 2008.

[6] L. Di Jorio, A. Laurent, and M. Teisseire. Mining frequent gradual itemsets from large databases. In Proceedings of the Int. Conf. on Intelligent Data Analysis, IDA'09, Lyon France, 2009.

[7] P. Poncelet F. Del Razo Lopez, A. Laurent and M. Teisseire. FTMnodes: fuzzy tree mining based on partial inclusion. Fuzzy Sets and Systems, 160:2224–2240, 2009.

[8] A. Frank and A. Asuncion. UCI machine learning repository, 2010.

[9] E. Hüllermeier. Implication-based fuzzy association rules. In Proceedings of the Int. Conf. on Principles of Data Mining and Knowledge Discovery (PKDD'01), pages 241–252, Freiburg, Germany, 2001.

[10] E. Hüllermeier. Association rules for expressing gradual dependencies. In Proceedings of the 6th European Conf. on Principles of Data Mining and Knowledge Discovery, PKDD'02, pages 200–211. Springer-Verlag, 2002.

[11] A. Laurent, M.-J. Lesot, and M. Rifqi. Graank: Exploiting rank correlations for extracting gradual dependencies. In Proceedings of Eighth International Conference on Flexible Query

Answering Systems (FQAS'09), Roskilde, Denmark, 2009.

[12] C. Molina, J.M. Serrano, D. Sánchez, and M.A. Vila. Measuring variation strength in gradual dependencies. In Proceedings of the 5th International Conference EUSFLAT'2007, Ostrava, Czech Republic, 2007.

[13] M. Viviani P. Ceravolo and M. C. Nocerino. Knowledge extraction from semi-structured data based on fuzzy techniques. International Journal of Knowledge-Based Intelligent Engineering Systems, 2004.

[14] A. Laurent S. Ayouni, S. Ben Yahia and P. Poncelet. Fuzzy gradual patterns: What fuzzy modality for what result? In Proceedings of the International Conference on Soft Computing and Pattern Recognition (SoCPaR'10), Cergy, France, 2010.

[15] A. Laurent T.D. Thac Do and A. Termier. PGLCM: efficient parallel mining of closed frequent gradual itemsets. In Proceedings of the 10th IEEE International Conference on Data Mining (ICDM), Sydney, Australia, 2010.