Prog Artif Intell
DOI 10.1007/s13748-017-0131-7

CrossMark

REGULAR PAPER

Improvement of subgroup descriptions in noisy data by detecting exceptions

Pedro González[1] · Ángel Miguel García-Vico[1] · Cristóbal José Carmona[2,3] · María José del Jesus[1]

Author Proof

Received: 9 March 2017 / Accepted: 30 May 2017
© Springer-Verlag Berlin Heidelberg 2017

Abstract The presence of noise in datasets to which data mining techniques are applied can greatly reduce the quality and interest of the knowledge extracted. Subgroup discovery is a supervised descriptive rule discovery technique which is not exempt from this problem. The aim of this paper is to improve the descriptions of subgroups previously obtained by any subgroup discovery algorithm in noisy datasets. This is achieved using the post-processing approach of the MEFES algorithm, that first detects exceptions in the input subgroups and then includes those exceptions in the descriptions. The experiments performed in noisy datasets show the suitability of the proposal to improve the quality of the results.

Keywords Subgroup discovery · Exceptions · Noisy data · MEFES

✉ Pedro González
pglez@ujaen.es

Ángel Miguel García-Vico
agvico@ujaen.es

Cristóbal José Carmona
cjcarmona@ubu.es

María José del Jesus
mjjesus@ujaen.es

1 Department of Computer Science, University of Jaen, 23071 Jaén, Spain

2 Department of Civil Engineering, University of Burgos, 09006 Burgos, Spain

3 Leicester School of Pharmacy, De Montfort University, LE1 9BH Leicester, UK

1 Introduction

Subgroup discovery (SD) [8,19] is an interesting task within data mining that allows the extraction of novel and interesting knowledge about subgroups of the data whose behaviour with respect to a variable of interest is significantly different from that of the whole dataset.

Different factors influence the quality of the subgroups obtained by SD algorithms such as missing values, noise, and so on. These problems can affect the interpretations, the decisions taken and the models created from the data, as well as the performance of the system. In particular, the presence of noise in datasets on which data mining techniques are applied can greatly reduce the quality and interest of the knowledge extracted and worsen the accuracy.

Studies on the impact of noise in data mining tasks have traditionally focused on predictive data mining, with little attention has been paid to its impact in descriptive data mining. In addition, the usual approach is the use of noise filtering methods [23] as a pre-processing step to identify and eliminate noisy instances, but they usually can not produce data with characteristics similar to those of the original data [38]. In this way, it would be interesting to explore approaches different to the use of filters for dealing with noisy data in SD. A particular consequence of noise in SD is the appearance of exceptions within the models generated. The detection and description of these exceptions caused by noise could be a good starting point to improve the results of SD algorithms in noisy environments.

The aim of this paper is to improve the descriptions of the subgroups in noisy environments by using exceptions, rather than using a pre-processing method to filter noise in SD. According to this, a methodology is proposed that involves obtaining SD rules (using any SD algorithm, both evolutionary and non-evolutionary) to later detect exceptions

uncorrected proof

Journal: **13748** Article No.: **0131** MS Code: **PRAI-D-17-00017.1** ☐ TYPESET ☐ DISK ☐ LE ☐ CP Disp.:**2017/6/5** Pages: **10** Layout: **Large**

in those rules in datasets with noise, in order to increase the level of description of the rules. This is done using the MEFES [6] post-processing algorithm that, applied to the results of a SD algorithm, allows to detect exceptions in the rules that describe the subgroups, and then obtain modified rules that include the exceptions. These exceptions could correspond to noisy values or outliers. The advantage of this approach is that experts can analyse the exceptions detected and determine whether they correspond to outliers (obtaining interesting knowledge) or noise. Our hypothesis is that this methodology, that works well in datasets without noise [6], will work particularly well with noisy data.

A complete experimental study is developed with datasets with noise in order to verify the applicability of the post-processing mechanism, and check if it is a good alternative to the use of noise elimination or mitigation approaches.

The remaining of the paper is organised as follows. Section 2 introduces the concept of SD and its main properties, and Sect. 3 describes the problem of the presence of noise in data mining and SD. Section 4 describes the post-processing proposal to improve the results of the SD algorithms in datasets with noise. Section 5 describes the experiments carried out, comparing the results of SD algorithms with those obtained after applying MEFES algorithm, and analysing whether this reduces the impact of noise on the quality of the results. Finally, Sect. 6 presents some concluding remarks.

2 Subgroup discovery

SD is a data mining technique which attempts to obtain a set of independent rules with a good compromise between generality-precision and with high levels of interest. This concept was initially introduced by Kloesgen [24] and Wrobel [35], and formally defined by Siebes using the name Data Surveying for the discovery of interesting subgroups [31]. It can be defined as [36]:

> *In subgroup discovery, we assume we are given a so-called population of individuals (objects, customers, ...) and a property of those individuals we are interested in. The task of subgroup discovery is then to discover the subgroups of the population that are statistically "most interesting", i.e. are as large as possible and have the most unusual statistical (distributional) characteristics with respect to the property of interest.*

SD is receiving an special interest throughout the community in these years due to the capacity to describe problems from a different perspectives than traditional descriptive inductions such as applications in medicine [5,9,11], e-learning [28,29], industry [6,10,21], amongst others.

Knowledge is represented by rules in SD. A rule (R) can be defined as:

$$R : Cond \ \rightarrow \ TargetVar$$

where $TargetVar$ is a value for the variable of interest (target variable) for the subgroup discovery task and $Cond$ is commonly a conjunction of features (attribute-value pairs) which is able to describe an unusual statistical distribution with respect to the $TargetVar$.

The main elements defining the SD approaches and the quality measures used are described below.

2.1 Main elements of subgroup discovery approaches

Different elements must be considered to apply an SD algorithm [3]:

- *Type of the target variable*. Different types of target variable can be used: binary, nominal or numeric. Different analyses can be applied for each type considering the target variable as a dimension of the reality to study.
- *Description language*. The representation of the subgroups must be suitable to obtain interesting rules. These rules must be simple and therefore are represented as attribute-value pairs in conjunctive or disjunctive normal form in general. Furthermore, the values of the variables can be represented as positive and/or negative, through fuzzy logic, or through the use of inequality or equality and so on.
- *Quality measures*. These are a key factor for the extraction of knowledge because the interest obtained depends directly on them. Furthermore, quality measures provide the expert with the importance and interest of the subgroups obtained. Different quality measures have been presented in the specialised bibliography [14,19,24–26].
- *Search strategy*. This is very important, since the dimension of the search space has an exponential relation to the number of features and values considered. Different strategies have been used up to the moment, for example beam search, evolutionary algorithms and search in multi-relational spaces. The algorithms implemented, their search strategies and applications can be observed in [19].

2.2 Quality measures in subgroup discovery

SD algorithms seek to obtain simple and interpretable subgroups, being desirable to cover most of the examples of the property of interest. According to this definition and the study of the different quality measures used in the literature presented in [8], three guidelines are proposed in order to establish the type of measure more suitable to analyse the quality of the subgroups obtained by any SD algorithm:

Author Proof

uncorrected proof

- *Interpretability*. The idea is to obtain few rules containing a low number of variables in order to help the expert to understand and use the extracted knowledge. The algorithm must obtain a low number of rules with a low number of variables because the algorithms look for simple and interpretable subgroups through partial relations. Therefore, we propose the use of the number of variables and rules for this guideline
- *Relation sensitivity-confidence*. An SD algorithm should obtain results with good precision, where most of the covered examples belong to the value of the analysed target variable, i.e. the algorithm must achieve the best possible relation between sensitivity and confidence. Both quality measures are essential to provide subgroups to experts covering as many correctly described examples as possible. The balance between both quality measures is difficult to reach by the algorithms due to the decrease that a measure undergoes when trying to increase the other. Both quality measures (Sensitivity and Confidence) must be considered for this guideline.
- *Novelty*. An SD model should contribute new knowledge about the problem, providing the experts with information that describes unusual and interesting behaviour within the data. This objective could be measured with a wide number of quality measures such as novelty, interest or significance, amongst others. Nevertheless, it is important to emphasise the use of unusualness to measure this objective because it contributes with generality and confidence to the problem. Moreover, this quality measure is widely used in the specialised literature. Therefore, despite the large number of quality measures within this category, we propose the use of unusualness.

After that, the aim of an SD algorithm is to find a good balance between these three guidelines, since this leads to a good performance in a large number of quality measures used in SD, and not only in those used in the search process.

3 Influence of noise in data mining and subgroup discovery

Noise is a real problem which is usually found in data. Such is its influence in the construction of a model that it can lead to reduce system performance in terms of classification accuracy, time in building and/or size of the model [39]. In fact, the quality of any dataset is determined by a large number of components [33]. Two of these are the source of the data and the input of the data, which are inherently subject to error. Errors in real-world datasets are therefore common and action must be taken to mitigate their consequences [37].

Two different types of noise are generally distinguished [4]:

- *Noisy attributes*, which are erroneous values of the attributes of the dataset. Several causes induce noise in an attribute such as labelling process, data entry errors, absence of attributes and so on.
- *Noisy classes*, that occurs when the instance belongs to the incorrect class; it can be caused by the same properties mentioned previously.

Noise in a dataset can be found both in the target variable or class as in the attributes, where the quality of the attributes indicates how well attributes characterise the instances, and the quality of the class labels represents whether the class of each instance is correctly assigned. However, the noisy classes only have sense in the training file and nowadays there are techniques to reduce it in a good way [4]. On the other hand, the noise in the attributes is more present in the real data and its handling is more difficult [38]. Noisy attributes include erroneous attribute values, missing or unknown attribute values or incomplete attributes, amongst others.

When applying data mining techniques, if noise is present in the training cases, this means that even low levels of noisy attributes can cause common cases to overwhelm rare cases. On the other hand, if noise is present in the test, the cases will be misclassified because the noise corrupted the test instance by making it to look like another class or because the incorrect classification of a case was learned during the training step.

Traditionally, problems arising from the presence of noise in classification has received special attention throughout the literature. However, this problem has not been widely analysed from the descriptive point of view. Specifically, this lack of analysis is also present in SD. The only approximation with an analysis about the presence of noise in data for SD can be observed in [27], where the behaviour of EFSs in SD is analysed and different noise filters are applied in order to improve the results. However, this analysis lacks of interesting information for the experts about the noise filtered.

This leads us to believe that we would obtain more interesting knowledge using an alternative approach to filtering in problems with noise in SD, such as the one presented in the next section.

4 The use of MEFES with noisy datasets for subgroup discovery

The problems derived from the use of filtering techniques as a pre-processing with datasets with noise make us think of the search for alternative strategies to handle data with noise in SD. Perhaps we can take advantage of the fact that several factors as missing values, outliers or the noise cause rare cases in the dataset, i.e. these types of data cause small groups of instances which correspond to another class. These

Author Proof
uncorrected proof

incorrectly described instances can be described as exceptions [32]. If we are able to detect these exceptions, we could determine if they correspond to noise or another situation, and the description of the subgroups could be improved by incorporating this knowledge into the rules.

For this purpose, the post-processing algorithm MEFES [6] can be used. MEFES is a multi-objective evolutionary fuzzy system for the detection of exceptions in subgroups which extracts modified subgroups in a post-processing stage, improving the results obtained by any SD algorithm. The main purpose of the algorithm is to find out exceptions associated for each subgroup, representing incorrectly described examples within the subgroup - examples within the subgroup with a different value of the target variable. The modified subgroups are formed by the initial subgroups and their exceptions. This way, based on the concept of the exceptions, an improvement of data mining algorithms can be focused as a search process of exceptions within the data described by any data mining model. In this way, knowledge extracted from a problem in a noisy environment can be improved.

The following scheme summarises the operation of MEFES:

1. Starts from a set of initial subgroups (R) obtained by any SD algorithm.
2. Search for exceptions associated to each subgroup.
3. Generate modified subgroups (R') formed by the initial subgroups and their exceptions associated.
4. Evaluate the modified subgroups.

So, the objective is to introduce a methodology which consist of the following steps:

- Use an SD algorithm, both evolutionary and non-evolutionary, to obtain subgroups in a dataset with noise.
- Apply the post-processing algorithm MEFES to the rules obtained to detect exceptions in those rules which could be caused by noise.
- Include these exceptions in the original rules to obtain modified rules in order to increase the level of description on the subgroups.

Our hypothesis is that, as MEFES works well in datasets without noise [6], the proposed methodology will work particularly well with noisy data, since noise can cause exceptions to appear in the rules, and that exceptions would be detected and the rules modified accordingly, so improving the results. In this sense, this methodology can become a good alternative to the use of pre-processing filtering methods, by providing interesting knowledge to the experts.

In addition, the use of MEFES as a post-processing stage provides:

- an improvement of the accuracy of the SD algorithms, because possible errors of the model in the description of examples are fixed; and
- new knowledge to the experts, because new spaces in the data with unusual behaviour are delimited.

Although the MEFES algorithm is described in detail in [6], the most important features are summarised below. MEFES is based on the NSGA-II approach [12], a multi-objective evolutionary algorithm where the objective vectors used are: sensitivity and confidence. The use of these quality measures as objectives provides the algorithm an improvement in quality measures such as precision and other specific measures for SD as unusualness.

Amongst its main operators, it is interesting to remark the use of these specific operators to keep the purpose of the algorithm:

- Oriented initialisation. It generates a population with individuals which contain amongst their properties the same values that the initial subgroup together with new values for the remaining attributes. Afterwards, this new operator generates part of the population with biased individuals and the rest are generated randomly.
- Oriented mutation. It is a new operator derived from the standard mutation [18]. In this case, the modification is related to the values of the initial subgroup which must be kept in the individual, i.e. the values of the new individual corresponding to those of the initial subgroup cannot be modified.
- Oriented re-initialisation based on coverage. A verification on the Pareto to see whether evolves or not is performed before to obtain the main population of the next generation. It is considered that the Pareto evolves if it covers at least one example of the dataset not covered by the Pareto of the previous generation. If the Pareto does not evolve a re-initialisation of the population is performed but this initialisation keep the non-repeated individuals of the Pareto front and all new individuals keep the same values of the initial subgroup.

According to this, the modified subgroup is described by the expression:

$$R_i' : \text{IF } Cond_i \text{ AND } \overline{Exc_i} \text{ THEN } TargetVar \tag{1}$$

where $Cond_i$ represents the condition for R_i and Exc_i represents conditions for associated exceptions to the rule R_i.

Examples of SD rules modified by the MEFES algorithm including exceptions in the descriptions of the subgroups can be find in [6]. In spite of that, en example is described below to facilitate understanding. Let us suppose we have applied an SD algorithm to discover subgroups for the well-known

Journal: 13748 Article No.: 0131 MS Code: PRAI-D-17-00017.1 TYPESET DISK LE CP Disp.:2017/6/5 Pages: 10 Layout: Large

Author Proof
uncorrected proof

Author Proof

uncorrected proof

IRIS dataset, obtaining the following rule:

$$R_1 : \text{IF } PetalWidth = \text{“}High\text{''} \text{ THEN } Class = Iris - virginica$$

Once applied the MEFES post-processing algorithm to the SD rule, the modified rule obtained might look like the following:

$$\begin{aligned} &R'_1 : \text{IF } (PetalWidth = \text{“}High\text{''} \text{ AND} \\ &\quad \text{NOT(} \\ &\quad (PetalLength = \text{“}Low\text{''} \text{ AND } SepalWidth = \text{“}Medium\text{''}) \text{ OR} \\ &\quad (PetalLength = \text{“}Low\text{''} \text{ AND } SepalLength = \text{“}Low\text{''}))) \\ &\quad \text{THEN } Class = Iris - virginica \end{aligned}$$

In order to analyse this type of rules, modified quality measures for SD have to be defined because the evaluation of the subgroups with exceptions must be performed considering the examples covered by the initial subgroup without the examples covered by its associated exceptions. Below are defined the modified quality measures used for the evaluation of the modified subgroups:

– Unusualness of a subgroup with exceptions:

$$Unus'(R'_i) = \left(\frac{TP_{R'_i}}{(TP+FP)_{R'_i}} - \frac{(TP+FN)_{R_i}}{N} \right) \cdot \frac{(TP+FP)_{R'_i}}{N} \quad (2)$$

where $TP_{R'_i} = TP_{R_i} - FP_{Exc_i}$, TP_{R_i} are the number of correctly described examples of the rule, FP_{Exc_i} are the number of incorrectly described examples for the set of associated exceptions to the rule, $(TP+FP)_{R'_i} = (TP+FP)_{R_i} - (TP+FP)_{Exc_i}$, $(TP+FP)_{R_i}$ are the number of examples covered by the rule, $(TP+FP)_{Exc_i}$ are the examples covered by the set of associated exceptions to the initial rule, $(TP+FN)_{R_i}$ are the number of examples for values of the target variable, and N is the total number of examples.

– Sensitivity of a subgroup with exceptions:

$$Sens'(R'_i) = \frac{TP_{R'_i}}{(TP+FN)_{R_i}} \quad (3)$$

– Fuzzy confidence of a subgroup with exceptions:

$$Cnf'(R'_i) = \frac{\sum_{E^k \in E / E^k \in TargetVar} APC(E^k, R'_i)}{\sum_{E^k \in E} APC(E^k, R'_i)} \quad (4)$$

where $APC(E^k, R'_i) = APC(E^k, R_i) - APC(E^k, Exc_i)$.

Table 1 Properties of the datasets used from the KEEL repository

Name	n_v	$TargetVar$	n_s
Balance	4	3	625
Heart	13	2	270
Iris	4	3	150
Monk-2	6	2	432
Nursery	8	5	12, 960
Penbased	16	10	10, 992
Pima	8	2	768
Shuttle	9	7	2175
Spambase	57	2	4597
Wdbc	30	2	569
German	20	2	1000
Ionosphere	33	2	351
Magic	10	2	1902
New-thyroid	5	3	215
Page-blocks	10	5	5472
Phoneme	5	2	5404
Segment	19	7	2310
Sonar	60	2	208
Thyroid	21	3	7200
Zoo	16	7	101

5 Experimentation

This section describes the details of the experimental study carried out to analyse the improvement of the results when applying MEFES post-processing algorithm on the knowledge generated by some of the most outstanding subgroup discovery algorithms in a noisy environment. The experimental study is divided in different subsections to clarify the approach proposed. First, the experimental framework used is described, including the datasets used, the process to induce noise in the original datasets, and the methodology employed to perform the experiments. Then, it is analysed if the results of the SD algorithms worsen when the level of noise is increased. Once the worsening of the results is verified, the MEFES post-processing algorithm is applied to the results of the selected SD algorithms to analyse if the new results improve the original ones.

5.1 Experimental framework

The experimental study uses 20 datasets from the KEEL [1,2] repository.[1] Table 1 shows the properties of these datasets, including $Name$, number of variables (n_v), number of values of the target variable ($TargetVar$) and number of instances (n_s) of each dataset.

[1] http://www.keel.es.

Author Proof

Table 2 Parameters used in the algorithms

Algorithm	Parameter
Apriori-SD	Minimum support = 0.03, minimum confidence = 0.6, number of rules = 5
SDIGA	Population size = 100, evaluations = 10,000, crossover probability = 0.60, mutation probability = 0.01, minimum confidence = 0.6, representation of the rule = canonical, linguistic labels = 3, objective1 = sensitivity, objective2 = unusualness
NMEEF-SD	Population size = 50, evaluations = 10,000, crossover probability = 0.60, mutation probability = 0.1, minimum confidence = 0.6, representation of the rule = canonical, linguistic labels = 3, objective1 = sensitivity, objective2 = unusualness
MEFES	Population size = 50, evaluations = 10,000, crossover probability = 0.60, mutation probability = 0.1, re-initialisation based on coverage with 90% of biased, minimum confidence = 0.80, representation of the rule = canonical, linguistic labels = 3

It is important to remark that one of the algorithms used in the experiments, Apriori-SD [22], is not able to handle large datasets, i.e. with more than 20 variables. This problem is illustrated in [6] with a experimental study based on a feature selection process. Therefore, the experimental study for Apriori-SD is carried out with only 15 of the datasets, those with up to 20 variables.

In order to analyse the impact of the noise on the different datasets used for the SD task it is necessary to control the noise level. Therefore, manual mechanisms are used to add noise in data. Starting from the previously mentioned datasets from the KEEL repository, new datasets with noise are generated by adding noise on both the training and test partitions. The presence of noise in both the training and test partitions allows us to observe how noise affects the accuracy of the models generated.

Noise is introduced in datasets through a random attribute noise scheme [40], where certain percentage of values of each attribute of the datasets are substituted with wrong (noisy) values, consistent with the hypothesis that interactions between attributes are weak [39]. The percentages of noisy values introduced determines de level of noise, i.e. a dataset with a noise level of 10% indicates that 10% of the attribute values of the dataset have been replaced by corrupt values. For the experiments, datasets with noise levels of 5% and 10 % have been generated. The noise introduced in each attribute has a low correlation with the noise introduced in the others. In addition, noise is only introduced with numerical attributes. The noisy values are assigned through a random value between the minimum and maximum of the domain of the attribute, following a uniform distribution.

The experiments have been carried out using some of the most representative algorithms for SD, both classical, such as Apriori-SD [22], and based on EFSs, such as SDIGA [20] and NMEEF-SD [7]. After that, the post-processing algorithm MEFES [6] has been applied to the results of the previous algorithms. The parameters used in the experimental study for the different algorithms can be observed in Table 2.

In the experiments for the different algorithms, Apriori-SD [22], SDIGA [20], and NMEEF-SD [7], and the application of MEFES [6] on the rules generated by these algorithms, the results presented in the different tables are obtained by means of five-fold cross-validation. In this way, datasets are divided into 5 partitions with equal number of instances but maintaining the class ratio in each one. The training stage is performed with four partitions, obtaining a set of subgroups, and the remaining partition is used to evaluate the quality of this set of subgroups. This procedure is repeated five times, using for the evaluation a different partition each time. Finally, the results shown are the average results of the five repetitions of the evaluation process. Therefore, quality measures presented in the result tables are the average results of all the rule sets in the different datasets analysed: unusualness ($UNUS$), sensitivity ($SENS$) and fuzzy confidence ($FCNF$). The quality measures used for Apriori+MEFES, SDIGA-MEFES and NMEEF-SD+MEFES are $UNUS'$, $SENS'$, and $FCNF'$, but are represented with the same acronyms in order to avoid confusion.

In order to complete the experimental study, analysing whether there are significant differences between the results of the algorithms Apriori-SD, SDIGA and NMEEF-SD with respect to the application of the MEFES algorithm to their results, a statistical comparison is performed. In [13,17] a set of simple, safe and robust nonparametric tests for statistical comparisons of classifiers are recommended. According to that, the Wilcoxon signed-ranks test [30,34] is selected in this analysis to make the comparison. A complete description of the Wilcoxon signed-ranks test and other nonparametric tests for pairwise and multiple comparisons, together with software for their use is available in [15,16] and on the complementary website.[2]

[2] http://sci2s.ugr.es/sicidm/.

Journal: 13748 Article No.: 0131 MS Code: PRAI-D-17-00017.1 TYPESET DISK LE CP Disp.:2017/6/5 Pages: 10 Layout: Large

Table 3 Average results with different levels of noise

Algorithm	%Noise	*UNUS*	% ↓	*SENS*	% ↓	*FCNF*	% ↓
Apriori-SD	0	0.064		0.548		0.68	
	5	0.059	7.8	0.518	5.5	0.655	3.8
	10	0.056	12.5	0.513	6.4	0.650	4.6
SDIGA	0	0.049		0.774		0.596	
	5	0.034	30.6	0.727	6.1	0.563	5.5
	10	0.030	38.8	0.691	10.7	0.547	8.2
NMEEF-SD	0	0.094		0.907		0.796	
	5	0.082	12.8	0.875	3.5	0.761	4.4
	10	0.069	26.6	0.825	9.0	0.726	8.8

Author Proof

5.2 Impact of noise in SD algorithms

An analysis showing the impact of noise in evolutionary fuzzy systems for SD can be seen in [27]. However, we consider necessary to include a classical SD algorithm in order to obtain a more general view on the impact of noise in the results of the SD algorithms. Hence, in this study both classical and evolutionary algorithms for SD are analysed with respect to their behaviour in a noisy environment. To do so, Apriori-SD, SDIGA and NMEEF-SD have been run with both the original datasets and datasets with different noise levels to check how the presence of noise affects the results of these algorithms. Table 3 shows the average results of the different quality measures for Apriori-SD, SDIGA and NMEEF-SD with different levels of noise, and the percentages of decrease when noise is introduced respect to the original datasets (0% noise). Different levels of noise in data are employed in this experimental study; specifically, we use 5% and 10% of noise induced. The complete results on the different datasets for each algorithm are available in the website.[3]

These experiments show that the results are worse when noise is introduced for both classical and evolutionary SD algorithms. In fact, these results become even worse as more noise is introduced in the datasets. In particular, the quality measure for SD that deteriorates the most in these algorithms is the unusualness. Sensitivity and confidence are also worsen, but reaching only 10% of decrease. This means that some of the quality measures suffer significant deterioration when the datasets have noise, making it interesting to work towards the reduction of the impact of noise on the results of SD algorithms.

5.3 Impact of noise using the approach proposed

A comparison of the results of the SD algorithms (Apriori-SD, SDIGA, and NMEEF-SD) and those obtained after applying the approach proposed (that implies the application of the algorithm MEFES to the results of the SD algorithms, and called Apriori-SD+MEFES, SDIGA+MEFES and NMEEF-SD+MEFES) is presented in Table 4. In this table, the level of noise ($\%Noise$), $Algorithm$ and results of the quality measures explained above are shown. The complete results obtained for each algorithm in the different datasets are available in the Website http://simidat.ujaen.es/papers/SD-Noisy.

As can be observed, in this experimental study are employed different levels of noise in data in order to analyse the quality of the post-processing approach for SD algorithms in noisy environments (5 and 10% of noise induced). In general, there is a relative loss of quality in measures when the level of noise is increased, i.e. there is a loss of

Table 4 Average results with different levels of noise and the post-processing algorithm MEFES

%Noise	Algorithm	*UNUS*	*SENS*	*FCNF*
0	Apriori-SD	0.064	0.548	0.681
	Apriori-SD + MEFES	0.070	0.509	0.723
	SDIGA	0.049	0.774	0.596
	SDIGA + MEFES	0.051	0.715	0.603
	NMEEF-SD	0.094	0.907	0.796
	NMEEF-SD + MEFES	0.099	0.894	0.817
5	Apriori-SD	0.059	0.518	0.655
	Apriori-SD + MEFES	0.064	0.468	0.692
	SDIGA	0.034	0.727	0.563
	SDIGA + MEFES	0.037	0.666	0.572
	NMEEF-SD	0.082	0.875	0.761
	NMEEF-SD + MEFES	0.086	0.855	0.779
10	Apriori-SD	0.056	0.513	0.650
	Apriori-SD + MEFES	0.061	0.468	0.692
	SDIGA	0.030	0.691	0.547
	SDIGA + MEFES	0.033	0.647	0.564
	NMEEF-SD	0.069	0.825	0.726
	NMEEF-SD + MEFES	0.073	0.804	0.745

[3] http://simidat.ujaen.es/papers/SD-Noisy.

uncorrected proof

Journal: 13748 Article No.: 0131 MS Code: PRAI-D-17-00017.1 ☐ TYPESET ☐ DISK ☐ LE ☐ CP Disp.:2017/6/5 Pages: 10 Layout: Large

Author Proof

Table 5 Wilcoxon test for the comparison SD Algorithm + MEFES versus SD Algorithm

%Noise	Algorithm		$R+$	$R-$	p val	$Hypothesis$
0	Apriori-SD	$UNUS$	114	6	0.002	Rejected by Apriori-SD + MEFES
		$SENS$	0	120	0.001	Rejected by Apriori-SD
		$FCNF$	109	11	0.005	Rejected by Apriori-SD + MEFES
	SDIGA	$UNUS$	135	36	0.033	Rejected by SDIGA-MEFES
		$SENS$	0	153	0.001	Rejected by SDIGA
		$FCNF$	135	36	0.033	Rejected by SDIGA-MEFES
	NMEEF-SD	$UNUS$	90	15	0.019	Rejected by NMEEF-SD + MEFES
		$SENS$	0	120	0.001	Rejected by NMEEF-SD
		$FCNF$	105	0	0.001	Rejected by NMEEF-SD + MEFES
5	Apriori-SD	$UNUS$	92.5	27.5	0.065	Rejected by Apriori-SD + MEFES
		$SENS$	0	120	0.001	Rejected by Apriori-SD
		$FCNF$	105	15	0.011	Rejected by Apriori-SD + MEFES
	SDIGA	$UNUS$	175	56	0.384	Non-rejected
		$SENS$	0	231	0.000	Rejected by SDIGA
		$FCNF$	205	26	0.001	Rejected by SDIGA-MEFES
	NMEEF-SD	$UNUS$	167	43	0.021	Rejected by NMEEF-SD + MEFES
		$SENS$	0	171	0.000	Rejected by NMEEF-SD
		$FCNF$	192	18	0.001	Rejected by NMEEF-SD + MEFES
10	Apriori-SD	$UNUS$	91	29	0.078	Rejected by Apriori-SD + MEFES
		$SENS$	0	120	0.001	Rejected by Apriori-SD
		$FCNF$	110	10	0.005	Rejected by Apriori-SD + MEFES
	SDIGA	$UNUS$	186	45	0.013	Rejected by SDIGA-MEFES
		$SENS$	0	210	0.000	Rejected by SDIGA
		$FCNF$	219	12	0.000	Rejected by SDIGA-MEFES
	NMEEF-SD	$UNUS$	157	33	0.013	Rejected by NMEEF-SD + MEFES
		$SENS$	0	210	0.000	Rejected by NMEEF-SD
		$FCNF$	199	11	0.000	Rejected by NMEEF-SD + MEFES

quality between results obtained by a dataset with a concrete noise level with respect to the case without added noise. The analysis for each quality measure is explained below:

- Unusualness. MEFES improves the results of Apriori-SD, SDIGA, and NMEEF-SD independently of the level of noise as can be observed in Table 4.
- Sensitivity. This quality measures can never be improved by MEFES because it quantifies the ratio of examples per target variable of the original subgroup. Despite this, the loss is directly related to the level of noise.
- Confidence. This measure has a short relative loss between the dataset without noise and that with a level of 10%. In all the cases, the results after applying MEFES improve those of the original SD algorithm.

To complete these statements, a statistical study for the quality measures of unusualness, sensitivity and fuzzy confidence has been performed. These quality measures are analysed independently through the Wilcoxon test. The results of this test will show the existence or not of significant differences between the algorithms for each measure. A confidence level of $\alpha = 0.1$ is used in all the experiments. Table 5 presents the results, including the noise level ($\%Noise$) employed, the name of the $Algorithm$, the different quality measures ($UNUS$, $SENS$, $FCNF$), the positive range ($R+$), the negative range ($R-$), the correspondent p-value, and the $Hypothesis$.

The results of the statistical tests with different levels of noise determine that the algorithms Apriori-SD and NMEEF-SD with the post-processing algorithm MEFES obtain the best results with significant differences with respect to the original SD algorithms in unusualness and fuzzy confidence. In the case of the algorithm SDIGA, the post-processing algorithm MEFES allows to obtain better results with significant differences in fuzzy confidence but not in unusualness. As expected, it is also confirmed that there are significant differences in favour of the original SD algorithms in terms of the sensitivity measure. In summary, the results support that in noisy environments, the application of the post-processing algorithm MEFES allows to improve the results regarding novelty and confidence.

Uncorrected proof

Author Proof

6 Conclusions

In this paper, we have analysed the influence of the noise on SD algorithms. Specifically, the analysis has been performed with some of the most outstanding classical and evolutionary algorithms for SD, Apriori-SD, SDIGA and NMEEF-SD. To do so, different levels of noise (5 and 10%) were introduced into the original datasets.

The experimental study shows a loss of quality of the results obtained when noise is introduced in the datasets. In this way, the appearance of noise in real-world datasets could lead to a loss of quality in subgroups obtained for any approach employed. The identification of these data in real-world data is a key factor in order to improve the results.

This contribution proposes the use of a post-processing algorithm called MEFES in order to search for exceptions within subgroups obtained for any SD algorithm from the literature, i.e. the application of MEFES in subgroups obtained previously allows the detection of exceptions with bad descriptions within the original subgroups. Considering the original and modified subgroup, an expert could determine the elements corresponding to the noise and delete them from the data, or treat them in some way, for example. Therefore, the idea of this contribution is not delete the noise but rather consider a subgroup such as an independent problem, to palliate the possible noise within the subgroup and to improve the description of the original subgroup.

The experimental study has been carried out in three of the most relevant algorithms within SD, Apriori-SD, SDIGA, and NMEEF-SD, with different features. Apriori-SD is a modification for the SD task of the widely known Apriori algorithm for association rules, SDIGA is monoobjective evolutionary fuzzy system for SD and NMEEF-SD is a multi-objective evolutionary fuzzy system based on the NSGA-II approach [12]. In these algorithms, the behaviour after the post-processing stage is satisfactory because the quality of the original subgroups extracted is improved, even with different levels of noise.

Acknowledgements This paper was supported by the Spanish Ministry of Economy and Competitiveness under Project TIN2015-68454-R (FEDER Founds).

References

1. Alcalá-Fdez, J., Fernández, A., Luengo, J., Derrac, J., García, S., Sánchez, L., Herrera, F.: KEEL data-mining software tool: data set repository, integration of algorithms and experimental analysis framework. J. Mult. Valued Logic Soft Comput. **17**(2–3), 255–287 (2011)
2. Alcalá-Fdez, J., Sánchez, L., García, S., del Jesus, M., Ventura, S., Garrell, J., Otero, J., Romero, C., Bacardit, J., Rivas, V., Fernández, J., Herrera, F.: KEEL: a software tool to assess evolutionary algorithms for data mining problems. Soft. Comput. **13**(3), 307–318 (2009)
3. Atzmueller, M., Puppe, F., Buscher, H.P.: Towards knowledge-intensive subgroup discovery. In: Proceedings of the Lernen–Wissensentdeckung-Adaptivität-Fachgruppe Maschinelles Lernen, pp. 111–117 (2004)
4. Brodley, C., Friedl, M.: Identifying mislabeled training data. J. Artif. Intell. **11**, 131–167 (1999)
5. Carmona, C.J., Chrysostomou, C., Seker, H., del Jesus, M.J.: Fuzzy rules for describing subgroups from influenza a virus using a multi-objective evolutionary algorithm. Appl. Soft Comput. **13**(8), 3439–3448 (2013)
6. Carmona, C.J., González, P., García-Domingo, B., del Jesus, M.J., Aguilera, J.: MEFES: an evolutionary proposal for the detection of exceptions in subgroup discovery. An application to concentrating photovoltaic technology. Knowl. Based Syst. **54**, 73–85 (2013)
7. Carmona, C.J., González, P., del Jesus, M.J., Herrera, F.: NMEEF-SD: non-dominated multi-objective evolutionary algorithm for extracting fuzzy rules in subgroup discovery. IEEE Trans. Fuzzy Syst. **18**(5), 958–970 (2010)
8. Carmona, C.J., González, P., del Jesus, M.J., Herrera, F.: Overview on evolutionary subgroup discovery: analysis of the suitability and potential of the search performed by evolutionary algorithms. WIREs Data Min. Knowl. Discov. **4**(2), 87–103 (2014)
9. Carmona, C.J., González, P., del Jesus, M.J., Navío, M., Jiménez, L.: Evolutionary fuzzy rule extraction for subgroup discovery in a Psychiatric Emergency Department. Soft. Comput. **15**(12), 2435–2448 (2011)
10. Carmona, C.J., Ramírez-Gallego, S., Torres, F., Bernal, E., del Jesus, M.J., García, S.: Web usage mining to improve the design of an e-commerce website: OrOliveSur.com. Expert Syst. Appl. **39**, 11243–11249 (2012)
11. Carmona, C.J., Ruiz-Rodado, V., del Jesus, M.J., Weber, A., Grootveld, M., González, P., Elizondo, D.: A fuzzy genetic programming-based algorithm for subgroup discovery and the application to one problem of pathogenesis of acute sore throat conditions in humans. Inf. Sci. **298**, 180–197 (2015)
12. Deb, K., Pratap, A., Agrawal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
13. Demsar, J.: Statistical comparisons of classifiers over multiple data sets. J. Mach. Learn. Res. **7**, 1–30 (2006)
14. Gamberger, D., Lavrac, N.: Active subgroup mining: a case study in coronary heart disease risk group detection. Artif. Intell. Med. **28**(1), 27–57 (2003)
15. García, S., Fernández, A., Luengo, J., Herrera, F.: Study of statistical techniques and performance measures for genetics-based machine learning: accuracy and interpretability. Soft. Comput. **13**(10), 959–977 (2009)
16. García, S., Fernández, A., Luengo, J., Herrera, F.: Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental Analysis of Power. Inf. Sci. **180**, 2044–2064 (2010)
17. García, S., Herrera, F.: An extension on "statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons. J. Mach. Learn. Res. **9**, 2677–2694 (2008)
18. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley Longman Publishing Co., Inc., Reading (1989)
19. Herrera, F., Carmona, C.J., González, P., del Jesus, M.J.: An overview on subgroup discovery: foundations and applications. Knowl. Inf. Syst. **29**(3), 495–525 (2011)
20. del Jesus, M.J., González, P., Herrera, F., Mesonero, M.: Evolutionary fuzzy rule induction process for subgroup discovery: a case study in marketing. IEEE Trans. Fuzzy Syst. **15**(4), 578–592 (2007)

uncorrected proof

Journal: **13748** Article No.: **0131** MS Code: **PRAI-D-17-00017.1** ☐ TYPESET ☐ DISK ☐ LE ☐ CP Disp.:**2017/6/5** Pages: **10** Layout: **Large**

Author Proof

21. Jin, N., Flach, P.A., Wilcox, T., Sellman, R., Thumim, J., Knobbe, A.J.: Subgroup discovery in smart electricity meter data. IEEE Trans. Ind. Inf. **10**(2), 1327–1336 (2014)
22. Kavsek, B., Lavrac, N.: APRIORI-SD: adapting association rule learning to subgroup discovery. Appl. Artif. Intell. **20**, 543–583 (2006)
23. Khoshgoftaar, T.M., Rebours, P.: Improving software quality prediction by noise filtering techniques. J. Comput. Sci. Technol. **22**(3), 387–396 (2007). doi:10.1007/s11390-007-9054-2
24. Kloesgen, W.: Advances in knowledge discovery and data mining, chap. Explora: A Multipattern and Multistrategy Discovery Assistant, pp. 249–271. American Association for Artificial Intelligence (1996)
25. Kloesgen, W., Zytkow, J. (eds.): Handbook of Data Mining and Knowledge Discovery. Oxford University Press Inc, New York (2002)
26. Lavrac, N., Cestnik, B., Gamberger, D., Flach, P.A.: Decision support through subgroup discovery: three case studies and the lessons learned. Mach. Learn. **57**(1–2), 115–143 (2004)
27. Luengo, J., García-Vico, A.M., Pérez-Godoy, M.D., Carmona, C.J.: The influence of noise on the evolutionary fuzzy systems for subgroup discovery. Soft. Comput. **20**(11), 4313–4330 (2016). doi:10.1007/s00500-016-2300-1
28. Noaman, A.Y., Luna, J.M., Ragab, A.H.M., Ventura, S.: Recommending degree studies according to students? Attitudes in high school by means of subgroup discovery. Int. J. Comput. Intell. Syst. **9**(6), 1101–1117 (2016)
29. Poitras, E.G., Lajoie, S.P., Doleck, T., Jarrel, A.: Subgroup discovery with user interaction data: an empirically guided approach to improving intelligent tutoring systems. Educ. Technol. Soc. **19**(2), 204–214 (2016)
30. Sheskin, D.: Handbook of Parametric and Nonparametric Statistical Procedures, 2nd edn. Chapman and Hall, London (2006)
31. Siebes, A.: Data surveying: foundations of an inductive query language. In: Proceedings of the 1st International Conference on Knowledge Discovery and Data Mining, pp. 269–274. AAAI Press, Palo Alto (1995)
32. Suzuki, E.: Data mining methods for discovering interesting exceptions from an unsupervised table. J. Univers. Comput. Sci. **12**(6), 627–653 (2006)
33. Wang, R.Y., Storey, V.C., Firth, C.P.: A framework for analysis of data quality research. IEEE Trans. Knowl. Data Eng. **7**(4), 623–640 (1995). doi:10.1109/69.404034
34. Wilcoxon, F.: Individual comparisons by ranking methods. Biometrics **1**, 80–83 (1945)
35. Wrobel, S.: An algorithm for multi-relational discovery of subgroups. In: Proceedings of the 1st European Symposium on Principles of Data Mining and Knowledge Discovery, LNAI, Vol. 1263, pp. 78–87. Springer, Berlin (1997)
36. Wrobel, S.: Relational Data Mining, chap. Inductive Logic Programming for Knowledge Discovery in Databases. Springer, Berlin (2001)
37. Wu, X.: Knowledge Acquisition from Databases. Ablex Publishing Corp, Norwood (1996)
38. Wu, X., Zhu, X.: Mining with noise knowledge: error-aware data mining. IEEE Trans. Syst. Man Cybern. Part A **38**(4), 917–932 (2008)
39. Zhu, X., Wu, X.: Class noise vs. attribute noise: a quantitative study. Artif. Intell. Rev. **22**(3), 177–210 (2004)
40. Zhu, X., Wu, X., Yang, Y.: Error detection and impactsensitive instance ranking in noisy datasets. In: Proceedings of the 19th National conference on Artificial Intelligence, pp. 378–383. AAAI Press, Palo Alto (2004)

uncorrected proof

Journal: **13748** Article No.: **0131** MS Code: **PRAI-D-17-00017.1** ☐ TYPESET ☐ DISK ☐ LE ☐ CP Disp.:**2017/6/5** Pages: **10** Layout: **Large**