Clustering association rules to build beliefs and discover unexpected patterns

Reference:
Bui Thi Danh, Meysman Pieter, Laukens Kris.- Clustering association rules to build beliefs and discover unexpected patterns
Applied intelligence - ISSN 1573-7497 - Dordrecht, Springer, 2020, 12 p.
Full text (Publisher's DOI): https://doi.org/10.1007/S10489-020-01651-1
To cite this reference: https://hdl.handle.net/10067/1661920151162165141

Noname manuscript No.
(will be inserted by the editor)

Clustering association rules to build beliefs and discover unexpected patterns

Danh Bui-Thi · Pieter Meysman · Kris Laukens

Received: date / Accepted: date

Abstract Interesting pattern discovery is an important topic in data mining research. Many different definitions have been proposed to describe whether a pattern is interesting. Among these many definitions, unexpectedness has shown to be a highly promising measure. Mining unexpected patterns allows one to identify a failing in prior knowledge and may suggest an aspect of the data that deserves further investigation.

Unexpected patterns are typically mined using belief-driven methods, but these require an established belief system. Prior studies have manually built their own partial belief systems to apply their method, but these remain laborious to create. In this study, we propose a novel approach that is able to automatically detect beliefs from data, which can in turn be used to reveal unexpected patterns. Central to this approach is a clustering-based method in which clusters represent beliefs and outliers are potential unexpected patterns. We also propose a pattern representation that captures the semantic relation between patterns rather than the lexical difference. An experimental evaluation on different datasets and a comparison to some other methods demonstrate the effectiveness of the proposed method, as well as the relevance of the discovered patterns.

Keywords Unexpected pattern mining · Pattern clustering · Belief system · association rule mining

1 Introduction

Association rule mining is a popular approach in data mining that can discover interesting patterns hidden in a database. It is widely used in many application domains, such as market basket analysis, web usage mining, intrusion detection and bioinformatics [28]. Compared to other approaches, the patterns or rules produced

Supported by Universiteit Antwerpen under BOF docpro grant.

Danh Bui-Thi, Pieter Meysman, Kris Laukens
Adrem Data Lab, Universiteit Antwerpen, Belgium
E-mail: {danh.bui-thi,pieter.meysman, kris.laukens}@uantwerpen.be

from association rule mining are easy to interpret because they are defined as a relationship $X \rightarrow Y$, where X, Y are each an item or a set of items occurring in the data. However, one of the major drawbacks of association rule mining is that it produces a large number of patterns that include both trivial and non-trivial relationships. Downstream analysis still too often consists of manual trudging through long lists of patterns numbering in the thousands. Therefore, prioritizing those patterns that are most likely to be interesting is a central problem in data mining research.

Unexpectedness is one of the nine criteria put forward to describe the relevance of a pattern for a given research question [17]. Unexpectedness is an important criterion as it suggests the patterns that contradict users' expectations or existing knowledge. These patterns allow us to identify failings in previous knowledge and may reveal an aspect of the data that needs further investigation. As a result, there are several studies tackle with mining unexpectedness from different kinds of data such as transaction data [25,29,31], sequential data [24], and recently are attributed graphs [5,6,22]. The unexpected patterns investigated in these studies are diverse as well: they could be unexpected transactions [30], unexpected associate rules [25,29,31], unexpected sequences [24] or unexpected sub-graphs [5,6,22].

In this study, we focus on detecting unexpected association rules from transaction datasets. We present a novel method to discover unexpected patterns based on beliefs that are automatically derived from the data. Our method utilizes a clustering-based framework where rule clusters play the role of a belief system in the mining process. To this end, we integrate DBSCAN, a density-based clustering algorithm, into the framework. Clustering of association rules has been previously explored [23,18,21] but unique to our proposed clustering procedure is that patterns are represented as feature vectors. The feature vectors are designed so that they can capture both semantic and lexical relationships between patterns. Therefore, these feature vectors not only facilitate processing but account for logical relationships such as contradiction or similarity in the distance between the rules. This focuses the way the relationship between rules are defined, which in turn improves rule clusters which can be considered as major beliefs within the dataset. Identified outliers from the clustering process can then be prioritized as candidate unexpected patterns. The implementation of our model can be found at `https://github.com/banhdzui/UnexpectedPatternMining`.

2 Related works

Unexpected pattern mining methods can be divided into *objective measures* and *subjective measures*. Objective measures are only based on pattern structure and underlying data while subjective measures include users' knowledge in term of belief system and an interestingness measure that estimates the unexpectedness of patterns to the beliefs.

In subjective measure based studies [29,31,25] the belief systems were constructed manually using domain knowledge. They are represented as first-order logic formulations which allow the system to easily verify the unexpectedness of patterns against the beliefs. The exact manner in which unexpectedness is defined differs among these studies: Silberschatz et al.[31] considered the relative difference in belief degree of the system with and without the pattern; Padmanabhan et al.[29] proposed a matching function in terms of logical contradiction while Liu et al.[25] employed lexical comparison of the pattern content. In 2005, Jaroszewicz et al. [20] proposed to represent domain knowledge as a Bayesian network instead of first-order logic and estimate unexpectedness as an absolute difference between the pattern's probability inferred from the network and from the observation. Recently, De Bie T. [11] proposed a Maximum Entropy model to formalize prior knowledge. The strategy was based on considering prior knowledge as constraints on a probabilistic model that represents the uncertainty in the data.

The largest drawback of the subjective measures is the difficulty in building a "good" belief system. Interviewing domain experts is a commonly proposed approach but is extremely hard in practice because of two reasons. First, experts may have trouble enumerating their beliefs; they often make vague and contradictory statements, which are hard to translate into logic. Second, there may be too many beliefs and it is not trivial to know which of these beliefs are really important and worth including in the belief system.

Compared to the subjective measures, objective measures can avoid the need to construct a belief system. Dong and Li [13] argued that unexpectedness could be interpreted in the statistical sense. They considered a significant variation in confidence between a rule and its neighbors as unexpectedness. However, their neighbor distance was defined in term of the lexical difference, which does not reflect true logical relationships between rules. We believe that the logical relationships also play an important role in estimating rule distance, thus we include this information into rule feature vectors which in turn effects the distance measure. In 2016, Chang et al. [7], proposed a hybrid way for mining unexpected patterns, which is specific for a case study of endometriosis data.

In addition, there are some shared concepts between our study with exception rule mining and exceptional model mining. Exception rules [8,9] can be considered as negative forms of strong association rules. For example, $X \rightarrow \sim Y, \sim X \rightarrow Y$ and $\sim X \rightarrow \sim Y$ are negative forms of the rule $X \rightarrow Y$. In other studies [19,32], the authors defined exception rules as those that represent a change of conclusion caused by adding an extra condition to common sense rules. The rule pairs in form $\{A \rightarrow X, AB \rightarrow X'\}$ are here evaluated with different metrics to judge whether $AB \rightarrow X'$ is exceptional. In 2005, Suzuki et al. [33] argued that there are 11 categories of exception rule based on rule triplets $\{y \rightarrow x, z \rightarrow x, yz \rightarrow x\}$. They proposed an algorithm that performs a depth-first search for triplets of literals a, b, c and leave an algorithm for a transaction dataset for future work. Meanwhile, exceptional model mining (EMM) is a supervised local pattern mining framework, where several target attributes are pre-selected and a model over these attributes is chosen to be the target concept. EMM aims to find interesting subgroups where the target model behaves differently from the complement of these subgroups or whole dataset [15]. These target models can be correlation, association, classifier, regression or Bayesian network. Their research focus on describing how much the relationship between the target attributes changes when they are placed in the context of those subgroups. [26] is an extension of [15] where they discover exceptional relationship in term of association rules instead of item-sets. In this study, we focus on unexpectedness as when patterns' conclusions change depending on the specific context. Thus when one outcome from a rule is expected given the bulk of the dataset, but another outcome is returned.

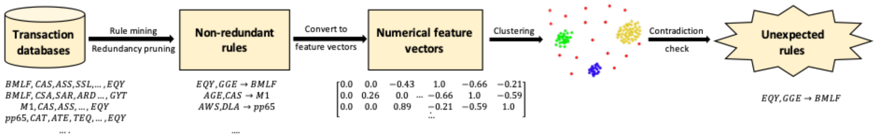

Fig. 1 Unexpected pattern discovery workflow with example.

Procedure 1 Redundancy checking

```
Input: A rule $R : X \to Y$
Output: $True$ if the rule is redundant; otherwise is $False$
1: for $k = 0$ to $|X|$ do
2:     for each $k$-item subset $X'$ of $X$ do
3:         if any item $\alpha \in X \setminus X'$ and $P(\alpha|X') = 1$ then
4:             return True
5:         end if
6:     end for
7: end for
8: Repeat from step 1 to step 7 for Y
9: return False
```

3 Methods

Figure 1 illustrates our method for detecting unexpected rules from data. It starts with mining association rules from the transaction database, followed by pruning the redundant rules. Non-redundant rules are then converted into numerical feature vectors before clustering. The non-redundant rules are next categorized into two groups: beliefs and potential candidates for unexpected rules. In the end, a contradiction check is applied to reveal the true unexpected rules from the candidates.

3.1 Pruning redundant rules

Redundant rules can present a major challenge in association rule mining. Several previous studies have defined prior assumptions that form the basis of custom pruning frameworks [4].

For this study, the redundancy pruning was inspired by the following case. Given a rule $X, \alpha \to Y$ in which α is a ubiquitous item in the database or α always occurs wherever X occurs in a database, means $P(\alpha|X) = 1$, due to an underlying relationship. Rules with such items do not provide any new knowledge about the data compared to $X \to Y$. Furthermore such rules are longer and hence harder to interpret. These types of redundant rules can be filtered based on their conditional probability $P(\alpha|S), S \subseteq X \setminus \{\alpha\}$. We thus apply Procedure 1 during association rule mining to prune redundant rules.

3.2 Rule Representation

We represent rules as numerical feature vectors. This representation is the basis for calculating distance between the rules, and enables subsequent rule clustering.

$x_1^{(l)}$	$x_2^{(l)}$	...	$x_{k_l}^{(l)}$	$x_1^{(r)}$	$x_2^{(r)}$	...	$x_{k_r}^{(r)}$

Fig. 2 An illustration of the feature vector structure. It consists of two parts: antecedent (or LHS) features and consequent (or RHS) features.

An association rule is defined by two itemsets: one in the left hand side named antecedent or LHS, and one in the right hand side, named consequent or RHS. The feature vector is constructed by concatenating features for the LHS and the RHS of the rule. Each feature corresponds to a unique item which potentially occurs at corresponding sides. Let $x_i^{(l)}, x_j^{(r)}$ be the i^{th} feature of LHS and the j^{th} feature of RHS; k_l, k_r are feature vector length for LHS and RHS respectively. Figure 2 illustrates the structure of the complete feature vector. In the case of unsupervised pattern mining, $k_l = k_r = |I|$, as any item can occur on either side of the rule. However, in the case of classification rule mining, only decision items occur in the RHS thus $k_l + k_r = |I|$.

A straight-forward representation of the feature vector of a rule would be a binary encoding that represents the absence or presence of items in the rule. However, the difference between two binary features vectors only addresses the difference in the lexical aspect. Consider the following examples from a hypothetical medical dataset that contains "*sick*″ and "*healthy*″ patients:

Example 1: $X \rightarrow$ "*sick*″ and $X \rightarrow$ "*healthy*″. The pair of rules differ only in two items: "*sick*″ and "*healthy*″, but these items are contrary each other. Thus, the distance of these rules should be greater than rules that differ in two arbitrary non-exclusive items.

Example 2: $X, \alpha \rightarrow$ "*sick*″ and $X, \beta \rightarrow$ "*sick*″, where α and β are positively correlated with each other. This means that α is likely to occur wherever β occurs and vice versa. Thus, these rules can be considered as highly similar and should have a small distance.

As these examples show, a binary representation of item presence is not sufficient to distinguish rules in an interestingness context. Therefore we propose a new feature value domain in the range of $[-1, 1]$, that represents how much a given feature is related to the LHS or RHS of the rule. A negative value implies a contradiction while a positive value implies similarity. More specifically, we define each feature value as a relationship score between the feature and the most correlated item in the rule.

This relationship score between a feature and an item is defined as the Spearman rank-order correlation coefficient of their relative occurrences in transaction data. These coefficients can then be interpreted as a similarity or a contradiction, depending on their sign. This interpretation is based on the assumption that items occurring together might be similar while items that appear in different transactions might be contradictory. To remove trivial relationships, we only keep the coefficients with a *p_value* ≤ 0.05, and the other coefficients are set to 0. Small correlation coefficients indicate little if any correlation. We can set a magnitude threshold, which is dataset-dependent, as more transactions allow minor relationships to be significant. Here the empirical correlation coefficient threshold was set to 0.1.

The steps of extracting feature values for a rule are described in Procedure 2, with M as the correlation matrix between items in the data after removing

$age = 30..39$	$age = 40..49$	..	$rec = y$	$rec = n$
-0.279	1.0	..	1.0	-1.0

Fig. 3 Feature vector of the rule "$age = 40..49, tumor\_size = 10..14 \rightarrow recurrence = no$" from the Breast Cancer dataset. The first row is the list of features while the second row contains corresponding feature values. Here the LHS contains an item which is highly similar to "$age = 40..49$" but contradictory to "$age = 30..39$". Meanwhile its RHS is "$recurrence = yes$", then the value of feature "$rec = yes$" is 1.0 while "$rec = no$" feature is -1.0 (contradiction).

Procedure 2 Computing feature values for a rule.

```
Input: A rule $rule$, a correlation matrix $M$
Output: Feature vector of $rule$ : $\phi(rule) = (x_1^{(l)}, .., x_{k_l}^{(l)}, x_1^{(r)}, .., x_{k_r}^{(r)})$
1: Initialize $\phi(rule) \leftarrow \vec{0}$, $L \leftarrow$ LHS items, $R \leftarrow$ RHS items
2: for $i = 0$ to $|L|$ do
3:     $\alpha \leftarrow$ item associated to the feature $x_i^{(l)}$
4:     $x_i^{(l)} \leftarrow M(\alpha, \beta)$ s.t $\beta \in L, \forall \beta' \in L, |M(\alpha, \beta)| \geq |M(\alpha, \beta')|$
5: end for
6: Repeat from step 2 to step 7 for RHS features but with RHS items
```

insignificant correlation coefficients. For each LHS feature, its value is the correlation coefficients of the feature and the LHS item that shows the highest (negative or positive) correlation to that feature. The same procedure is followed for the RHS features using the RHS items. Figure 3 illustrates the feature vector for the rule "$age = 40..49, tumor\_size = 10..14 \rightarrow recurrence = no$" from the Breast Cancer dataset [14]. By comparing these feature vectors, we can judge how the rules are related to each other. This is useful for subsequently grouping the rules into clusters and discovering unexpected rules.

3.3 Rule clustering and Contradiction check Function

To detect unexpectedness, we cluster the association rules and subsequently check for belief contradictions. Prior to clustering, a Principal Component Analysis (PCA), is applied on the feature vectors representing the association rules. This allows us to reduce the clustering computational complexity on large scale rule sets. The clustering is done by the DBSCAN algorithm [16] using the Euclidean distance on the proposed numerical feature vectors. DBSCAN classifies rules as either core points, reachable points or outliers. The rules that are core or reachable points are considered as beliefs because they are supported by several similar neighbors. The outliers are candidate unexpected rules. To determine if they are unexpected or not, these outliers are finally compared to the beliefs by a contradiction check function .

Definition 1 A rule $X \rightarrow Y$ is unexpected with respect to a belief $X' \rightarrow Y'$ on the dataset D if the following conditions hold:

- Y and Y' logically contradict each other.
- X and X' co-occur in a substantial subset of transactions in D. A subset is substantial if it satisfies a user-defined support threshold.
- X and X' are similar to each other (defined by the Cosine similarity).

Procedure 3 Discover unexpected rules

```
Input:
    Set of rules $Rules$, $minsup$
    Similarity threshold for LHS $\delta_1 \in (0, 1]$
    Contradiction threshold for RHS $\delta_2 \in [-1, 0)$
    DBSCAN parameter $minPts$, $maxEps$, $epsStep$
    A small proportion $\epsilon$
Output: Set of unexpected rules $U$
    {Estimate optimal $eps$}
$L_\epsilon \leftarrow \epsilon|Rules|$
$bestEps \leftarrow maxEps$
while $bestEps > 0$ do
    $clusters, outliers \leftarrow dbscan(Rules, eps, minPts)$
    if exists two clusters $C_1, C_2 \in clusters$ s.t.
    $|C_1| \geq L_\epsilon$ and $|C_2| \geq L_\epsilon$ and $C_1$ is contradict $C_2$ then
        $bestEps \leftarrow bestEps - epsStep$
    else
        break
    end if
end while
{Contradiction check function}
for each rule $R : X \rightarrow Y$ in $outliers$ do
    for each cluster $C$ in $clusters$ do
        if exists $R' : X' \rightarrow Y'$ in cluster $C$ s.t $sim(\phi(R)^{(l)}, \phi(R')^{(l)}) > \delta_1$ and
        $sim(\phi(R)^{(r)}, \phi(R')^{(r)}) \leq \delta_2$ and $sup(XX') \geq minsup$ then
            $U \leftarrow U \cup \{R\}$
        end if
    end for
end for
```

By combining the above conditions, we obtain the following knowledge base: $\{X \rightarrow\sim Y', X' \rightarrow Y', X, X'\}$. The knowledge base is un-satisfiable which can be proven by resolution. Because $X' \rightarrow Y'$ is a belief, the existence of $X \rightarrow Y$ becomes unexpected.

To estimate the contradiction or similarity between two itemsets, we proposed to compute the Cosine similarity of their feature vectors. If the Cosine value exceeds a given threshold, we can conclude that these itemsets are contrary or similar each other. We denote a similarity threshold for LHS as δ_1 and a contradiction threshold for RHS as δ_2, where δ_1 needs to be positive while δ_2 needs to be negative for an unexpected rule to occur.

The DBSCAN algorithm used to identify the candidate unexpected rules requires two parameters, *minPts* and *eps*. *minPts* is the minimum number of data points required to form a dense region and *eps* is the maximum distance a data point and the nearest data point can have to be considered as part of the same cluster. The choice of these parameter values directly impacts the belief system and the identified outliers. However a belief system should not contain any contradictory rules. This can be used to reject specific belief system solutions that are invalid. This observation can thus be used as a 'no-contradiction constraint' on the *eps* parameter for the DBSCAN clustering. This constraint specifies that there should be no contradiction allowed between clusters, or as a parameter that can be tuned, no contradiction between clusters that consist of a large number of rules. The maximum size of a contradicting cluster is $L_\epsilon = \epsilon|R|$ rules, where R is the set of all rules and ϵ is a small fraction. We then reduce the *eps* value steadily until DBSCAN finds a set of clusters that do not break the no-contradiction constraint.

The complete framework for detecting unexpected rules is described in Procedure 3. This procedure requires the following parameters: DBSCAN parameter *minPts*; an initial value for *eps maxEps*; a reducing step for *eps epsStep*; a small proportion ϵ for the non-contradiction constraint and two thresholds δ_1, δ_2 for the contradiction check function. After unexpected rules are discovered from the database, we can rank them according to the similarity in LHS between unexpected rules and their contrary beliefs.

4 Experiments and Discussion

In this section, we present experiments on six different datasets to test our method in pruning redundancy, constructing the belief systems and discovering the unexpected rules. These datasets are classification data which allows us to evaluate performance of the unexpected rules. At first, we generate rules from data and show the result on redundancy pruning. Next is the evaluation of the unexpected rules generated by our method and the comparison with two other methods by of Hussain et al.[19] and by Suzuki et al. [32]. Finally, we show some examples of unexpected rules obtained from the data and their interpretation.

Most of our datasets are obtained from UCI repository [14], including: *Adult, Breast cancer, Credit approval, Smtp (KDDCup99)* and *Pima*. Also included is a T-cell receptor (TCR) dataset, an example of a higher dimensional dataset derived from Dash et al.[10]. The Smtp(KDDCup99) is a subset of KDD CUP 1999 network intrusion data, which consists of transactions that have *service = smtp*. Similar to [34], we do not use all features from the Smtp data. We restricted the number of features to 9 (*count, dst_bytes, flag, src_bytes, logged_in, diff_srv_rate, dst_host_srv_serror_rate, same_srv_rate, dst_host_serror_rate*), based on the cross-validated information gain in the training dataset. Most of these datasets are imbalanced, i.e. class distributions are not uniform among the classes. In the Adult dataset, for example, the label "$\leq 50K$" occurs in 75.92% of the transactions while "$> 50K$" only in 24.08%. Such datasets potentially contain abnormal patterns hidden in minority classes that are not easy to discover. The properties of the datasets and the used parameters for association rule mining are summarized in Table 1. Except *Adult* and *Smtp(KDDCup99)* where training and test data are available, we randomly split every dataset into 8 : 2 for training and testing, respectively.

Prior to association rule mining, we convert continuous attributes into categorical attributes and then consider each pair of *attribute–value* as a transaction item. We used the Apriori algorithm [2] with low minimum support and high minimum confidence to generate pre-mined rules. Apriori is one of the most well-known algorithms in association mining. In this step, we can replace Apriori by other association mining algorithms such as FP-Growth [35], ECLAT [36] and LCM [37] which have been proposed to improve the mining process in terms of running time [38]. As this step is not the main objective of this study, we leave the exploration of these algorithms onto mining unexpectedness for our near future work. High minimum confidence is to guarantee the quality of the beliefs while low minimum support is to reveal unexpected rules. Spearman rank-order correlation coefficients between items are computed by *spearmanr* function of SciPy library.

All experiments run on an Intel Core i7 machine with 16GB of RAM running at 2.8GHz. The experimental results are described in the following sub-sections.

Dataset	**train**	**test**	**#attributes**	**#class**	**Minority class**
Adult (UCI)	32561	16281	14	2	0.241
BreastCancer (UCI)	240	46	9	2	0.296
CreditApproval(UCI)	598	92	15	2	0.446
Pima (UCI)	630	138	6	2	0.346
Smtp (KDDCup99)	96554	8268	9	2	0.0122
TCR (Dash et al.)	333	79	822	3	0.156

Table 1 A summary of experimental datasets and the used parameters for association rule mining.

Dataset	**minsup**	**minconf**	**#rules**	**#nonredundant**	**Ratio ∂**
Adult	2E-3	0.80	536723	278864	1.925
BreastCancer	1E-2	0.80	4586	1687	2.718
Credit	6E-2	0.90	438714	14076	**31.168**
Pima	4E-3	0.80	550	387	1.421
Smtp	5E-5	0.90	113077	15044	7.516
TCR	1E-2	0.80	11388	327	**34.825**

Table 2 Redundancy ratio ∂ detected by our method on the set of rules generated by the Apriori algorithm. The results on TCR and Credit approval data show high redundancy. This can be explained by strong correlation between items (attribute-value pairs) in the data. This creates many item combinations with high support without any information gain. The other datasets have much lower redundancy ratios (i.e. below 10.0) but the number of removed rules is still remarkable, such as Adult, Smtp.

4.1 Redundant rule pruning

To evaluate the performance of pruning redundant rules, we used the *redundancy ratio* ∂ [1] as defined by equation 1. The higher *redundant ratio* ∂ is, the more redundancy exists in the generated rules.

$$\partial = \frac{\text{Total rules generated } (T)}{\text{Non-redundant rules } (E)} \tag{1}$$

Table 2 shows the *redundancy ratio* which was obtained by our method on a set of rules generated by the Apriori algorithm for each dataset.

4.2 Unexpected rules evaluation

To evaluate the quality of unexpected rules, denoted as UnexpRules, that are automatically generated in our framework, we examine their contribution to the

Dataset	ϵ	**eps**	**#outliers**	**#UnexpRules**
Adult	2E-4	1.4	308	33
BreastCancer	0.0	0.8	1315	39
Credit	5E-3	1.2	4000	13
Pima	0.0	1.0	247	43
Smtp	5E-3	0.8	3427	80
TCR	0.0	1.8	33	3

Table 3 ϵ and the outcomes from the cluster-based framework for each dataset. We used $\epsilon > 0$ for the datasets with large rule set, such as Adult, Credit and Smtp. In these cases, the rules easily collect enough *minPts* neighbors to form clusters even with a very small *eps*. Thus, instead of using a strict non-contradictory constraint, we limit the clusters containing some contradictory factors to be small in size.

performance of classifiers on the minority class. The basic idea behind this evaluation is that classifiers are learning and capturing knowledge hidden in data. Therefore these classifiers can be considered as black-box belief systems. If rules improve the performance of classifiers, that means these rules are meaningful but unexpected to the classifier based belief system. In other words, they are actually true unexpected patterns.

The evaluation procedure is as follows. A classifier and UnexpRules are generated from the same training data. An independent testing dataset is then used to validate performance of the classifier with and without using UnexpRules. In the case of using UnexpRules, the samples satisfying the LHS of UnexpRules must follow the prediction of UnexpRules instead of the classifier's. In this study, we used the F1 measure to validate the classifiers' performance. The two classifier models deployed in our experiments are Support Vector Machines (SVM, polynomial kernel with $degree = 3$) and Random Forest (RF, $numberoftrees = 20$), from the python scikit-learn package. These models are not chosen to represent the state-of-the-art in the field of unbalanced data classification, but are only used as a stand-in for those beliefs that can be easily learned from the data.

The UnexpRules set was obtained using $minPts = 3$, $maxEps = 2.0$, $epsStep = 0.1$ and $\delta_1 = 0.0$. The values of other parameters, ϵ and δ_2, depended on the dataset. In most cases, $\delta_2 = -1$. If more than two prediction classes were present the correlation coefficients among these classes cannot reach -1.0, thus $\delta_2 = -0.5$ was used. Table 3 reports ϵ and the summary of the cluster based framework outcomes while Figure 4 shows the contribution of generated UnexpRules to SVM and Random Forest over six datasets.

In regards to running time, the DBSCAN algorithm takes only a few seconds for small rule sets but it can take up to several hours for larger scale sets. e.g. Adult data which consists of more than 27×10^4 rules. The computational complexity for detecting UnexpRules from clusters is $O(n^2)$ for a rule set with size n, as the procedure 3 requires comparisons between all outliers and belief rules.

We also examine the influence of the threshold δ_1 on our framework. The threshold δ_1 is used as the criterion for the similarity between the LHS of an unexpected rule and a belief. In addition, it impacts the search for the optimal *eps* for the DBSCAN algorithm and thus effects the quality of both the belief system and UnexpRules. Figure 5 illustrates the effect of δ_1 on the clustering result. The figure shows that when the δ_1 increases, the clusters grow and the number of outliers decreases. The fact is that δ_1 influences the *eps* through relaxing or

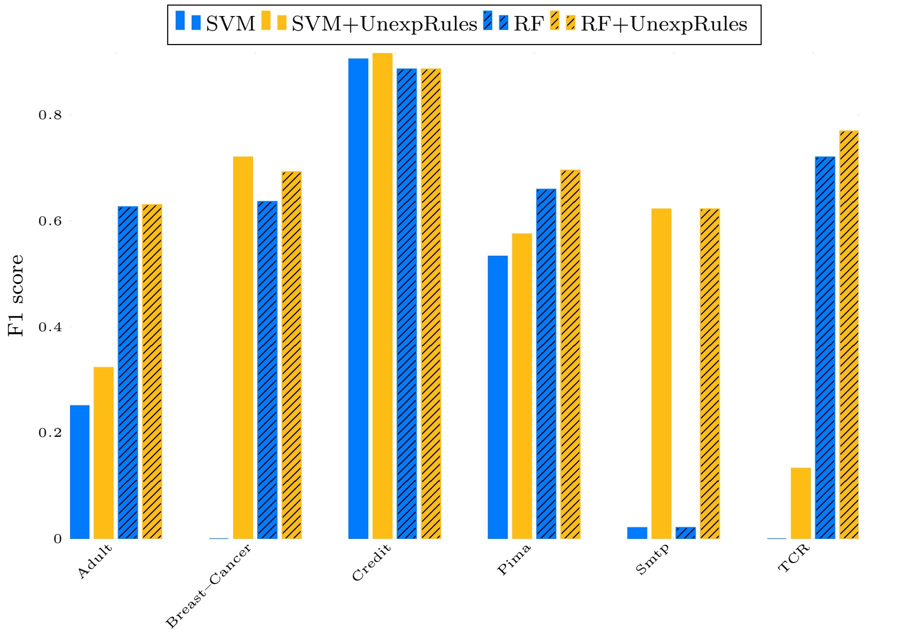

Fig. 4 F1 score on minority class for the SVM and RF, with and without using UnexpRules on testing data. Overall the obtained UnexpRules improve the performance of classifiers on the minority class. This indicates that these rules are unexpected and actionable. The largest improvement can be found in the case of *Smtp* dataset where the F1-score on the minority class increases from 0.021 to 0.621. This dataset in comparison to the others is extremely imbalanced, only 1.22% in training data correspond to the “attack” class compared to 98.78% of “non-attack” class. This could prevent the classifier to learn knowledge related to “attack” class and the UnexpRules are helpful in this case.

restricting the no-contradiction constraint among clusters. The *eps* in its turn will guide the clustering process. Figure 6 reports the F1 score variation over four datasets when δ_1 increases from 0.0 to 1.0. For each dataset, the maximum improvement is reported at $\delta_1 = 0.0$ and decreases as δ_1 reaches 1.0.

Given the conceptual similarities between the unexpected patterns described here and the prior known exception rules, the two were compared for the different datasets. Table 4 reports the performance of exception rules based on the approach from Hussain et al [19] and Suzuki et al.[32] and the unexpected rules found by our method. Hussain et al. [19] generated exception rules by pairing common sense rules and ranks the rules with a relative interesting measure. Only exception rules with high relative score were kept to determine performance. As this method was designed to work on binary datasets, only those with two classes were used in this comparison. Suzuki et al. [32] introduced 5 constraints that an exception rule pair $\{A \rightarrow X, AB \rightarrow X'\}$ has to satisfy so that the rule $AB \rightarrow X'$ is considered as an exception. These constraints require following user-defined thresholds: $\theta_1^S, \theta_1^F, \theta_2^S, \theta_2^F, \theta_2^I$ which define the bounds for five derived scores related to $Pr(A), Pr(X|A), Pr(AB), Pr(X'|AB)$ and $Pr(X'|B)$ respectively and δ for significant level of rules. Based on the discussion in their study and our experimental examination, we choose $\theta_1^F = 0.5, \theta_2^F = 0.75, \theta_2^I = 0.5$ and $\delta = 0.9$. The

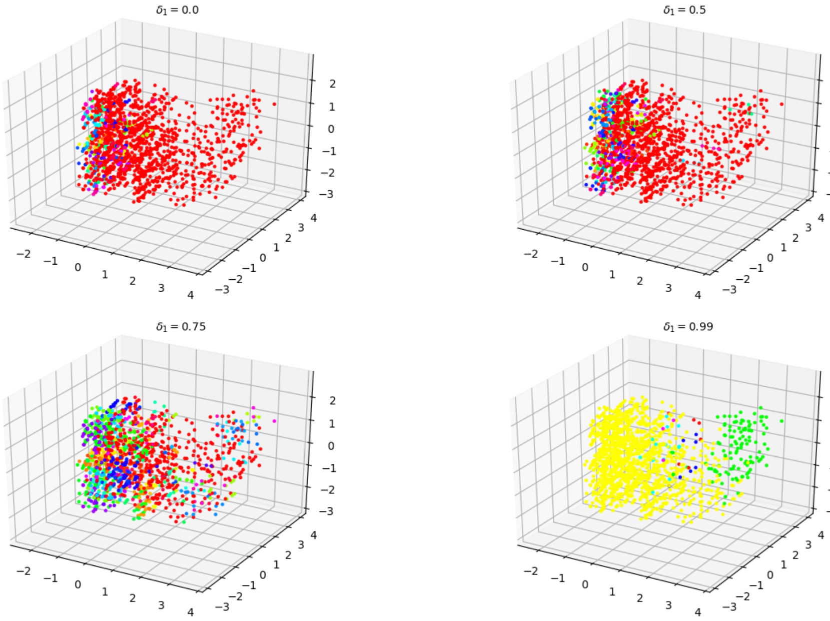

Fig. 5 The effect of δ_1 onto the clustering result of *Breast cancer* dataset. The red points are outliers while the others are beliefs. Data points in the same cluster are in the same color. The axes are three first principal components of the rules.

		Minority class			
Method	**Adult**	**BreastCancer**	**Credit**	**Pima**	**Smtp**
SVM	0.251	0.0	0.905	0.533	0.021
SVM+Hussain et al.	0.266	0.133	0.864	0.533	0.626
SVM+Suzuki et al.	0.297	0.235	0.902	0.552	**0.684**
SVM+UnexpRules	**0.323**	**0.72**	**0.916**	**0.575**	0.622
		Majority class			
Method	**Adult**	**BreastCancer**	**Credit**	**Pima**	**Smtp**
SVM	0.88	0.82	0.92	0.825	0.561
SVM+Hussain et al.	0.859	0.831	0.875	0.825	0.699
SVM+Suzuki et al.	0.883	0.827	0.921	**0.83**	**0.719**
SVM+UnexpRules	**0.884**	**0.895**	**0.931**	0.826	0.693

Table 4 F1 score on minority and majority classes when doing prediction on testing data with SVM and SVM plus unexpected rules generated from Hussain et al.[19], Suzuki et al.[32] and our method.

other thresholds depend on experimental datasets. In general, rules from three mentioned methods improved the F1 score on minority classes. However, the Hussain et al. approach lowers the F1 score for the majority classes of two datasets, Adult and Credit, while the others did not. The Suzuki et al. approach gave the significant improvement on Smtp data while UnexpRules did on Breast Cancer. Overall, our method gave the best score on most of the experimental datasets. This difference in performance was caused by the different types of patterns these methods mined. The Hussain et al. and the Suzuki et al. approaches suggest the rules $AB \rightarrow X'$ as exceptional when $A \rightarrow X$ is high in both support and confi-

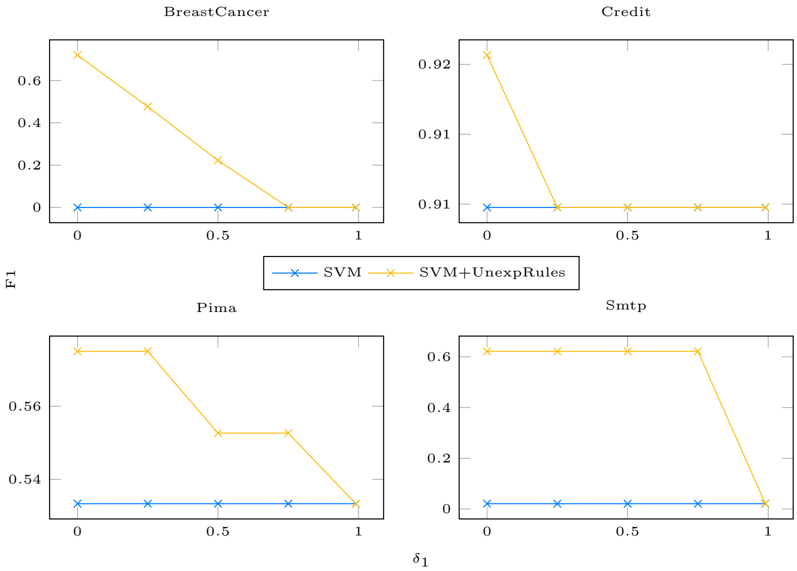

Fig. 6 F1 score on minority class of SVM with and without UnexpRules on four datasets with δ_1 varying from 0.0 to 1.0. This value signifies the minimal similarity between the LHS between an unexpected rule and a belief. None of the datasets have any UnexpRules at $\delta_1 = 1.0$.

dence and $B \rightarrow X'$ is low in confidence. Meanwhile, we focused on $B \rightarrow X'$ that is high in confidence and used AB as a reference for unexpectedness. The rules from our method are therefore more concise and more reliable.

4.3 Unexpected rules and their interpretation

To evaluate the validity and interpretability of both the beliefs and the unexpected rules, we take a closer look at some examples from the TCR, Credit approval and Breast cancer datasets.

TCR dataset: the rules represent relationships between the T-cell receptor beta-chain CDR3 amino acid sequence and the antigen that is targeted by the T-cell. The CDR3 beta-chains consist of 12 to 17 amino acids. The dataset contains human T-cell receptor data for three antigens, namely BMLF, M1 and pp65. The classification goal is to predict for the CDR3 amino acid sequence which antigen it is associated with, which has been recently tackled with a RF [12] as per the black-box belief system presented here. Here each CDR3 amino acid sequence was encoded as a set of 3-grams: one 3-gram is a sequence of three adjacent amino acids. Table 5 lists the unexpected rules generated from this dataset. From the table, we can see that contradiction surrounds two 3-grams EQY and GGE. The proposed method found the belief that EQY is associated with the $M1$ antigen. This belief matches observation made about this study in the past, the EQY 3-gram is derived from the TRBJ2-7*01 gene, which can in turn for this dataset be associated with the $M1$ antigen [27]. Thus our approach captured a belief that had been formulated by experts working with this data. However, the appearance

	Rules	conf(%)	supp(%)	Rank
Belief	$EQY \rightarrow antigen = M1$	0.894	0.354	
UnexpRule	EQY, $\mathbf{GGE} \rightarrow antigen = BMLF$	1.0	0.012	0.812
Reference	EQY, GGE		0.012	
Belief	ASS ,$\mathbf{EQY} \rightarrow antigen = M1$	0.911	0.339	
UnexpRule	ASS, $\mathbf{GGE} \rightarrow antigen = BMLF$	1.0	0.012	0.735
Reference	ASS ,EQY, GGE		0.012	
Belief	$\mathbf{EQY} \rightarrow antigen = M1$	0.894	0.354	
UnexpRule	$\mathbf{GGE} \rightarrow antigen = BMLF$	0.8	0.012	0.0264
Reference	EQY, GGE		0.012	

Table 5 Unexpected rule examples of the TCR dataset. $Reference$ is the LHS co-occurrence of belief and unexpected rule in data.

	Rules	**conf**	**sup**	**Rank**
Belief	$a_8 < 7$, $\mathbf{a_9 = t}$, $\mathbf{a_{10} = t}$,$\mathbf{a_{12} = f}$, $a_{13} = g$ $\rightarrow approval = yes$	0.901	0.137	
UnexpRule	$\mathbf{a_6 = ff}$, $a_8 < 7$, $a_{13} = g \rightarrow approval = no$	0.905	0.064	0.485
Reference	$a_6 = ff$, $a_8 < 7$, $a_9 = t$, $a_{10} = t$, $a_{12} = f$,$a_{13} = g$		0.0067	
Belief	$a_8 < 7$, $\mathbf{a_9 = t}$, $\mathbf{a_{10}} = t$, $\mathbf{a_{12} = f}$, $a_{15} < 25k \rightarrow approval = yes$	0.9	0.135	
UnexpRule	$\mathbf{a_6 = ff}$, $a_8 < 7$, $a_{15} < 25k \rightarrow approval = no$	0.905	0.064	0.425
Reference	$a_6 = ff$, $a_8 < 7$, $a_9 = t$, $a_{10} = t$, $a_{12} = f$,$a_{15} < 25k$		0.0067	

Table 6 Unexpected rule examples from the Credit approval dataset. $Reference$ is the LHS co-occurrence of belief and unexpected rule in data.

of GGE breaks the rule and it turns the conclusion to the $BMLF$ antigen. This is highly unexpected as the 3-gram GGE can also be derived from the same TRBJ2-7*01 gene. However in this instance the physicochemical properties of the T-cell receptor allow it to be associated with the $BMLF$ antigen. This is a departure from what has been previously observed, and is a good lead for further investigation.

Credit approval dataset: the rules represent associations between customer information and credit approval decision. Table 6 shows 2 of 6 unexpected rules discovered by our method. One belief says that if customers have the following properties: $a_8 < 7$, $a_9 = t$, $a_{10} = t$,$a_{12} = f$, $a_{13} = g$, they will be approved for credit with high confidence. However, there are some transactions in database, about 0.67%, showing customers that, even though they satisfy these properties, if their profile also includes the property $a_6 = ff$, will not be approved for credit with a higher confidence.

Breast cancer dataset: the mined rules correspond to the relationship between 9 attributes that describe women that have successfully undergone breast cancer treatment and if their cancer recurred at a later date. Table 7 shows two of the unexpected rules found from this dataset. In the first example, the belief says that *"if menopause=ge40, node-caps = no and irradiat=no then recurrence=no"*. However, when *inv-nodes=3..5* is added into the condition, cancer recurrence happens with a confidence of 100%. This contradiction can be interpreted as follows: normally we do not expect a recurrence in a woman who is menopausal after 40 years of age, who did not originally receive radiation treatment and had no

	Rules	**conf**	**sup**	**Rank**
Belief	$menopause = ge40$, $node - caps = no$, $irradiat = no \to recurrence = no$	0.823	0.233	
UnexpRule	$menopause = ge40$, $\mathbf{inv - nodes = 3..5}$, $node - caps = no$, $irradiat = no \to$ $recurrence = yes$	1.0	0.0167	0.878
Reference	$menopause = ge40, inv - nodes =$ $3 - 5, node - caps = no, irradiat = no$		0.0167	
Belief	$menopause = ge40$, $node - caps = no$, **breast=left** $\to recurrence = no$	0.833	0.146	
UnexpRule	$menopause = ge40$, $\mathbf{inv - nodes = 3..5}$, $node - caps = no$, $\mathbf{breast - quad = left - low} \to$ $recurrence = yes$	1.0	0.0167	0.768
Reference	$menopause = ge40$, $inv - nodes = 3..5$, $node - caps = no$, $breast = left$, $breast - quad = left - low$		0.0125	

Table 7 Unexpected rule examples related to the class attribute on the Breast cancer data. *Reference* is the LHS co-occurrence of belief and unexpected rule in data.

	Rules	**conf**	**sup**	**Rank**
Belief	$menopause = ge40$, $breast - quad = left - low \to$ $irradiat = no$	0.812	0.146	
UnexpRule	$menopause = ge40$, $\mathbf{node - cap = unknown}$, $breast - quad = left - low \to$ $irradiat = yes$	1.0	0.0125	0.854
Reference	$menopause = ge40$, $node - cap = unknown$, $breast - quad = left - low$		0.0125	
Belief	$age = 60 - 69$, $irradiat = no \to$ $inv - node = 0 - 2$	0.857	0.120	
UnexpRule	$age = 60 - 69$, $irradiat = no$, **inv-nodes=6-8** $\to node - cap = yes$	1.0	0.0125	0.808
Reference	$age = 60 - 69$, $irradiat = no$, **inv-nodes=6-8**		0.0125	

Table 8 Unexpected rule examples related to the other attributes on the Breast cancer data. *Reference* is the LHS co-occurrence of belief and unexpected rule in data.

so-called node caps, except if she had 3 to 5 auxiliary lymph nodes that contain metastatic breast cancer visible with histological examination. This follows the belief that if the cancer was not severe enough to warrant radiation treatment, it will likely have been entirely eliminated from the first treatment. However the contradiction is that some of these women already had auxiliary lymph node metastasis, which makes the cancer much harder to remove entirely.

Thus far only the class attribute has been considered as the unexpected outcome of the rule, however the approach is equally valid for other attributes. Table 8 shows two examples of such patterns. A belief describes that *"if patients go through menopause since 40 year old, and their tumors occur at the left low quad of breast then radiation treatment is no"*. Unexpectedly, if those patients have *node-cap* unknown then they do receive radiation treatment. In the second example, we find a belief that claims *"if patients whose age is from 60 to 69 and they do not get any radiation treatment then the number of auxiliary lymph nodes is less than 2"*. Unexpectedly, several cases in the data set with the same initial conditions, have auxiliary lymph nodes from 6 to 8. In addition, it adds the outcome *node-cap is yes* which does not occur in the case of $inv - node = 0 - 2$. In this manner, the approach is able to find novel outcomes that do not match the beliefs and warrant further study.

5 Conclusion

Unexpectedness is considered as one of the nine relevant criteria to evaluate pattern interestingness. It identifies a failing in previous knowledge or may reveal an aspect of data that needs further study. In this work, we developed a clustering-based method to identify unexpected rules from different datasets. An advantage of our method is that it can discover unexpected rules without the need to build a belief system manually. We evaluated the quality of unexpected rules based on the performance of classifiers. The experimental results show that the discovered unexpected rules are actionable and have great potential for imbalanced datasets which appear in many real world applications such as fault detection, fraud detection, and medical diagnosis.

To facilitate rules clustering, we proposed to represent association rules as numerical feature vectors. We have extended this representation to also summarize and visualize the association rules and their relationships. This could be useful for the development of a system in which users are able to examine and select interesting rules interactively. However, such representation requires high memory capacity when the number of unique items in the data set is large. Moreover, the clustering process scales poorly on large rule collections. Various options exist that are likely to speed up this aspect of the algorithm, such as using an approximate nearest neighbor search, but these are beyond the scope of this study. We also proposed a way for pruning redundant rules based on association between items. This approach holds great promise for integration with other interestingness discovery approaches to form a complete pattern prioritizing work-flow.

References

1. Aggarwal CC, Yu PS (2001) A new approach to online generation of association rules. TKDE 13:527–540
2. Agrawal R, Ramakrishnan Srikant (1994) Fast algorithms for mining association rules. In: Proceedings of International Conference on Very Large Databases, pp 487–499
3. Anglin P M, Gencay R (1996) Semiparametric estimation of a hedonic price function. Journal of Applied Econometrics 11: 633–648
4. Ashrafi MZ, Taniar D, Smith K (2004) A new approach of eliminating redundant association rules. In: Database and Expert Systems Applications. Springer Berlin Heidelberg, Berlin, Heidelberg, pp 465–474
5. Bendimerad A, Plantevit M, Robardet C (2018) Mining exceptional closed patterns in attributed graphs. Knowledge and Information Systems 56:1–25
6. Bendimerad AA, Plantevit M, Robardet C (2016) Unsupervised exceptional attributed sub-graph mining in urban data. In: Proceedings of IEEE International Conference on Data Mining, pp 21–30

7. Chang M-Y, Chiang R-D, Wu S-J, Chan C-H (2016) Mining unexpected patterns using decision trees and interestingness measures: a case study of endometriosis. Soft Computing 20:3991–4003
8. Daly O, Taniar D (2004) Exception rules mining based on negative association rules. In: Computational Science and Its Applications. Springer Berlin Heidelberg, Berlin, Heidelberg, pp 543–552
9. Taniar D, Rahayu W, Lee V, Daly O (2008) Exception rules in association rule mining. Applied Mathematics and Computation 205:735–750
10. Dash P, Fiore-Gartland AJ, Hertz T, Wang GC, Sharma S, Souquette A, Crawford JC, Clemens EB, Nguyen T-H-O, Kedzierska K, La Gruta NL, Bradley P, Thomas PG (2017) Quantifiable predictive features define epitope-specific T cell receptor repertoires. Nature 547:89–93
11. De Bie T (2011) Maximum entropy models and subjective interestingness: an application to tiles in binary databases. Data Mining and Knowledge Discovery 23:407–446
12. De Neuter N, Bittremieux W, Beirnaert C, Cuypers B, Mrzic A, Moris P, Suls A, Van Tendeloo V, Ogunjimi B, Laukens K, Meysman P (2018) On the feasibility of mining CD8+ T cell receptor patterns underlying immunogenic peptide recognition. Immunogenetics 70:159–168
13. Dong G, Li J (1998) Interestingness of discovered association rules in terms of neighborhood based unexpectedness. In: Research and Development in Knowledge Discovery and Data Mining. Springer Berlin Heidelberg, pp 72–86
14. Dua D, Karra Taniskidou E (2017). UCI Machine learning repository. Irvine, CA: University of California, School of Information and Computer Science
15. Duivesteijn W, Feelders A J, Knobbe A (2016) Exceptional Model Mining: Supervised descriptive local pattern mining with complex target concepts. Data Min Knowl Disc 30:4798
16. Ester M, Kriegel H-P, Xu X (1996) A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In: Proceedings of International Conference on Knowledge Discovery and Data Mining, pp 226–231
17. Geng L, Hamilton HJ (2006) Interestingness measures for data mining: A survey. ACM Computing Surveys 38
18. Gupta GK, Strehl A, Ghosh J (1999) Distance based clustering of association rules. In: Intelligent Engineering Systems Through Artificial Neural Networks. ASME Press, pp 759–764
19. Hussain F, Liu H, Suzuki E, Lu H (2000) Exception rule mining with a relative interestingness measure. In: Knowledge Discovery and Data Mining. Current Issues and New Applications. Springer Berlin Heidelberg, Berlin, Heidelberg, pp 86–97
20. Jaroszewicz S, Scheffer T (2005) Fast discovery of unexpected patterns in data, relative to a Bayesian network. In: Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp 118–127
21. Jorge A (2004) Hierarchical clustering for thematic browsing and summarization of large sets of association rules. In: Proceedings of SIAM International Conference on Data Mining, pp 178–187
22. Kaytoue M, Plantevit M, Zimmermann A, Bendimerad A, Robardet C (2017) Exceptional contextual subgraph mining. Machine Learning 106:1171–1211
23. Lent B, Swami A, Widom J (1997) Clustering association rules. In: Proceedings of International Conference on Data Engineering, pp 220–231
24. Li H, Laurent A, Poncelet P (2007) Mining unexpected sequential patterns and rules. Laboratoire dInformatique de Robotique et de Microlectronique de Montpellier
25. Liu B, Hsu W, Chen S (1997) Using general impressions to analyze discovered classification rules. In: Proceedings of International Conference on Knowledge and Data Mining, pp 31–36
26. Luna J M, Pechenizkiy M, Ventura S (2016) Mining exceptional relationships with grammar-guided genetic programming. Knowl Inf Syst 47:571594
27. Meysman P, De Neuter N, Gielis S, Bui Thi D, Ogunjimi B, Laukens K (2018) On the viability of unsupervised T-cell receptor sequence clustering for epitope preference. Bioinformatics
28. Naulaerts S, Meysman P, Bittremieux W, et al (2015) A primer to frequent itemset mining for bioinformatics. Briefings in Bioinformatics 16:216–231
29. Padmanabhan B, Tuzhilin A (1998) A belief-driven method for discovering unexpected patterns. In: Proceedings of International Conference on Knowledge Discovery and Data Mining, pp 94–100

30. Roel B, Jilles V, Arno Siebes (2017) Efficiently discovering unexpected pattern-co-occurrences. In: Proceedings of SIAM International Conference on Data Mining, pp 126–134
31. Silberschatz A, Tuzhilin A (1995) On subjective measures of interestingness in knowledge discovery. In: Proceedings of International Conference on Knowledge Discovery and Data Mining, pp 275–281
32. Suzuki E (2002) Undirected discovery of interesting exception rules. International Journal of Pattern Recognition and Artificial Intelligence 16:1065–1086
33. Suzuki E, ytkow JM (2005) Unified algorithm for undirected discovery of exception rules. International Journal of Intelligent Systems 20:673691
34. Williams G, Baxter R, Hongxing He, Hawkins S, Gu L (2002) A comparative study of RNN for outlier detection in data mining. In: Proceedings of IEEE International Conference on Data Mining, pp 709–712
35. Han J, Pei H, Yin Y (2000) Mining Frequent Patterns without Candidate Generation. SIGMOD Rec.:1–12
36. Zaki MJ (2000) Scalable algorithms for association mining. IEEE Transactions on Knowledge and Data Engineering, 12(3):372–390
37. Uno T, Kiyomi M, Arimura H (2004) LCM version 2: Efficient Mining Algorithms for Frequent/Closed/Maximal Itemsets. In: Proceedings of the IEEE ICDM workshop on frequent itemset mining implementations.
38. Luna JM, FournierViger P, Ventura S (2019) Frequent itemset mining: A 25 years review. WIREs Data Mining Knowl Discov 9