International Journal of Computational Intelligence Systems, Vol. 8, No. 6 (2015) 1178-1191

DisCoSet: Discovery of Contrast Sets to Reduce Dimensionality and Improve Classification

Zaher Al Aghbari
Department of Computer Science, University of Sharjah
Sharjah, United Arab Emirates
E-mail: zaher@sharjah.ac.ae

Imran N. Junejo
Department of Computer Science, University of Sharjah
Sharjah, United Arab Emirates
E-mail: ijunejo@sharjah.ac.ae

Received 30 October 2014

Accepted 13 October 2015

Abstract

Traditionally, contrast set mining aims at finding a set of rules that best distinguish the instances of different user-defined groups. Contrast sets are conjunctions of attribute-value pairs that are significantly more frequent in one group than in other groups. Typically, these contrast sets are extracted from categorical data or discretized numerical data. Existing methods of rule-based contrast sets require some user-defined thresholds to select the contrast sets. In this paper, we propose a greedy algorithm, called DisCoSet, to find incrementally a minimum set of local features that best distinguishes a class from other classes without resorting to discretization. The discovered contrast sets reduce the dimensionality of the feature vectors considerably and improve the classification accuracy significantly. We show that the proposed algorithm reduces the dimensionality of class instances by 40%-97% of the original length and yet improves classification accuracy by 10%-24% using different types of datasets.

Keywords: Contrast sets, dimensionality reduction, classification, information retrieval, data mining.

1. Introduction

Mining contrast sets is relatively a new data mining task that aims at finding the distinguishing characteristics of one group from other groups. Generally, researchers in various fields of sciences have proposed different methods to compare between groups. For example, in statistics, a broad range of statistical methods was developed to compare between groups.[1] In data mining, discriminating rules have been used to compare different groups of data.[2] Also, in time series, distinguishing patterns are used to detect groups of sequences.[3] These scientific endeavors aim at answering fundamental questions in different disciplines. For example, in image databases, the question might be, which image features best distinguish images of one class from others? In computer vision, particularly in action recognition, where the time factor is taken into consideration, the question might be, what is the smallest difference between the start and the end of an action sequence? Alternatively, the question might be, what is the smallest set of patterns that distinguish one action sequence from other action sequences? In medical applications, the question could be, which symptoms best distinguish patients of a certain disease from patients of other diseases?

In most existing works, the task of finding the contrast sets is roughly viewed as a variant of association rule mining, see Refs 3-4. The found contrast sets are extracted from categorical or discretized data[5]. Contrast sets were first defined by Bay and Pazzani[2] as conjunctions of attribute-value pairs that are significantly large, or more frequent, in one group than in the other. That is, the attribute-value distribution across groups differ meaningfully.[5] Typically, association rule mining discovers rules that describe the current state of the dataset, while contrast set mining identifies rules that distinguish one group, or class, from others, see Refs 5-6. In association rule mining, the rules having support greater than a certain threshold are selected; however, in contrast set mining, only the rules that represent substantial differences in the underlying probability distributions are selected. Such rules help users distinguish the different groups and understand their fundamental differences.

Following the work of Novak and Webb[5], the works on finding discriminating features of groups, or classes, can be divided into three categories: contrast set mining[4], emerging patterns[8], and subgroup discovery.[9] These categories are defined as:

- Contrast set mining: conjunctions of attribute-value pairs that differ meaningfully in their distribution across groups.
- Emerging pattern mining: itemsets whose support value change significantly from one class to another. These itemsets are extracted from time-stamped datasets to capture future trends.
- Subgroup discovery: Population subgroups that are statistically interesting with respect to the property of interest.

All of these categories share the same objective, which is to find a set of discriminating features of each user-defined group, or class, see Refs 5 and 7. Different issues were tackled by researchers in each of these three categories. In contrast set mining, statistical issues were addressed to reduce the detection of false sets. While researchers in emerging pattern mining focus on using the discovered patterns for classification. In subgroup discovery, techniques to identify a small set of features that have high coverage with respect to the property of interest were investigated.

While discovering the contrast sets, we aim at obtaining the smallest set of features in every class that best distinguishes it from other classes. Such a minimum set of features should appropriately differentiate one class from the other classes and should be easily interpreted by end users. The work closest to the proposed method is contrast set mining. However, contrast set mining mostly deals with categorical and discretized datasets. There are two main disadvantages of discretizing continuous data. First, it causes loss of information as values falling in the same range become indistinguishable and thus small differences in feature value become unnoticeable. Second, it causes very small changes in feature values close to the discretization border unjustifiably large as they belong to two different discrete values.

In this paper, we propose a greedy method, called DisCoSet (**Dis**covery of **Co**ntrast **Set**s), to find contrast sets of multiple classes/groups from numerical datasets. For each class, DisCoSet extracts a set of features that has the highest distinguishing power from other classes. The extracted set of feature is much smaller in length than the original feature vectors, and yet improves the classification of unseen instances. That is, if each of the original vectors of class C_h contains s features (attributes), $C_h = (a_1, a_2, \ldots, a_s)$, then DisCoSet finds the contrast set of this class $CS(C_h) = (a_j, a_k, \ldots, a_m)$, which is a local subset of the original feature set. The features in the contrast set are considered the attributes that distinguish C_h from other classes with high accuracy. The extracted contrast set of each class is local to the class and could be different in terms of length and distinguishing power from the contrast sets of other classes.

The optimal contrast set of every class can be easily found by checking all permutations of features. However, this straightforward approach is prohibitively expensive due to its exponential complexity. Unlike previous works that generate huge number of candidate contrast sets to select the optimal ones, DisCoSet prunes the search space while searching for the contrast sets. Thus, DisCoSet aims at optimizing the tradeoff between finding the contrast sets from numerical datasets and the classification accuracy. That is, DisCoSet selects only the features that improve the discriminating power of the contrast set of a class, which in turn improves the classification accuracy of unseen instances of the class. The contributions of the proposed method are:

- A greedy algorithm, called DisCoSet, to discover the contrast sets from numerical datasets without discretization.
- Contrast sets of reduced lengths (reduction in number of features by 40%-97%).
- Improved classification accuracy of unseen instances by 10%-24% as compared to using the original feature vectors.
- A pruning mechanism, which is embedded in DisCoSet, to reduce the search space for contrast sets.

The rest of the paper is organized as follows: Section 2 surveys the related work. In Section 3, we introduce some terminology and definitions. The proposed greedy algorithm, DisCoSet, is presented in Section 4. Section 5 discusses the experimental results. Finally, we conclude the paper in Section 6.

2. Related Work

The discovery of major differences among classes of data is an important goal in data mining. Following the work of Novak and Webb[5], we divide the related work into three categories.

Contrast Set Mining: The problem of contrast set mining was first introduced by Bay and Pazzani.[2] The authors proposed an algorithm, called STUCCO (Search and Testing for Understandable Consistent Contrasts), which is based on the Max-Miner rule discovery algorithm.[10] STUCCO estimates the interestingness of a contrast set by computing the statistical significance using X^2 test with a Bonferroni correction to control the false positives, i.e. finding only, but not necessarily all, significant contrast sets. Webb et al.[11] argue that contrast set mining is a special case of a more general rule discovery task, such as the Apriori association rule algorithm.[12] That is, a contrast set can be interpreted as an antecedent of a rule and the group as the consequent: $contrastSet \rightarrow Group_i$. In their algorithm, called Magnum Opus, they employ a heuristic approach to reduce the number of contrast sets in the search space.

A different approach, called CIGAR (Contrasting Grouped Association Rules), by Hilderman and Pechham[13] employs additional constrains and uses different statistical tests including a test for minimum support to find the rules. He et al.[14] find actionable rules from defined clusters of customer datasets by applying a contrast set mining algorithm. On the other hand, Minaei-Bidgoli et al.[15] proposed an algorithm to mine contrast rules on a web-based educational systems by using small minimum support to detect interesting rules that would have otherwise been ignored. Lin and Keogh[16] extended the idea of contrast set mining to handle time series data. The above-mentioned algorithms require setting domain specific parameters like the minimum support threshold between classes. Furthermore, these algorithms work only on categorical or discretized numerical data, which is not the case in real world applications.[5]

Emerging Patterns Mining: The datasets used by algorithms in this category are usually sequences or time-stamped data, which constitute the main difference between emerging patterns and contrast sets. The objective of emerging patterns mining algorithms is to capture future trends as defined by Dong and Li.[17] Other works, see Refs. 18-19 and 21, used the discovered emerging patterns in classification. Li et al.[22] utilized the discovered emerging patterns to classify acute lymphoblastic leukemia on microarray data. However, Song et al.[23] made use of the found emerging patterns to mine customer behaviors.

Fan and Ramamohanarao[18] proposed an approach to discover interesting patterns, while Soulet et al.[20] proposed to produce condensed representation of the emerging patterns. Another related algorithm, called ConSGapMiner, utilizes a depth-first search tree to enumerate candidate patterns and then prune the ones that do not satisfy certain constrains. Due to the nature of the data structure used, ConSGapMiner algorithm is inefficient.[25] To mine substrings, Chan et al.[26] proposed an algorithm based on the idea of emerging patterns. However, this algorithm only supports exact substring match. Patterns from software behaviors are mined by Lo et al.[27] to distinguish events that cause software failures. On the other hand, Deng and Zaiane[25] employ a suffix tree based algorithm and a sliding window matching mechanism to find the emerging patterns.

Subgroup Discovery: Subgroup discovery was defined by Wrobel[28] as the discovery of statistically interesting subgroup with respect to the property of interest. That is, given the property of interest, the distribution of this property in the subgroup is significantly different from the distribution in the entire dataset. Atzmuller and Puppe[29] proposed a fast algorithm for exhaustive subgroup discovery that exploits background knowledge. To produce smaller subgroups with higher coverage, compared to peer

algorithms, Kavsek and Lavrac[30] proposed the Apriori-SD algorithm. Xiang et al.[31] introduced an efficient sparse group hard thresholding algorithm to find subgroups. On the other hand, Xu et al.[32] proposed Gradient Boosted Feature Selection method to find reliable features.

Relational subgroup discovery algorithm is proposed by Wrobel[28] to detect subgroups in multi-relational datasets. Klosgen and May[33] proposed to discover subgroups in spatial databases. Genetic algorithms along with fuzzy systems are employed by Del Jesus et al.[34] to discover subgroups. A two-phase algorithm proposed by Langohr et al.[36] is used to find enriched gene sets that are specific for virus infected samples of a specific time point or a specific phenotype.

The above categories borrow ideas from feature selection algorithms, see Refs. 37 and 38, which select the relevant features from a dataset. There are two main heuristic approaches in the literature to select automatically the relevant subset of features: filter approach and wrapper approach.

The filter approach tries to find a subset of features independently of the inductive algorithm that will use this subset in classification. This is achieved by applying some statistics to select strong relevant features and filter out the weak relevant ones before executing the classification algorithm. In contrast, wrapper approach searches for a subset of features using cross-validation and compares the performance of the classification algorithm with each tested subset in order to select the best possible one. Although the wrapper approach achieves better classification performance as compared to filter approach, it is computationally expensive.[37] The filter approach emphasizes the discovery of relevant features that maximize the classification accuracy, while the wrapper approach searches for relevant features that minimize the classification error[39].

3. Preliminaries and Definitions

In this section, we introduce the notations and the definitions used in the paper.

Let a labeled database D be a set of feature vectors that represent objects. An object can be an image, an action in a video, or symptoms of a patient. A vector is set of attributes (features) that describe the properties of the object and each attribute is represented by a numerical value (integer or real). For example, if the object is an image, then the image could be represented by its color histogram vector, where the attributes are the different colors represented by their RGB values. Given user-defined classes ($C_1, C_2, \ldots, C_k$), each class C_g determines a partition (a subset of the vectors) in the database, $C_g \subseteq D$. These partitions are non-overlapping, that is $C_d \cap C_e = \emptyset$: $\forall_{d \neq e}$.

Definition 1. *Contrast set*: Given a labeled database D of vectors, $D = (v_1, v_2, \ldots, v_n)$, where each $v_j = (a_{j1}, a_{j2}, \ldots, a_{js})$ is a set of features, and the user-defined classes ($C_1, C_2, \ldots, C_k$), which are the non-overlapping partitions of D, a contrast set CS of class C_g, $CS(C_g)$, is the conjunction of features that distinguishes C_g from other classes. That is, $CS(C_g)$ should classify unseen instances of C_g with high accuracy.

$$CS(C_g) = (a_j, a_k, \ldots, a_m) \subseteq (a_1, a_2, \ldots, a_s) \quad (1)$$

The additive discriminating power of a feature is used as a criterion for determining whether the feature should be added to the contrast set of a class or not. That is, we measure the contribution of a candidate feature when added to the set in terms of classification accuracy of unseen instances. The feature that positively contributes the most to the classification accuracy is selected.

Definition 2. *Selection criteria*: Let P_i be the discriminating power (class-wise classification accuracy of class C_g against the rest of the classes) of the current set of features in $CS(C_g)$. A candidate feature a_r is added to $CS(C_g)$ *only* if the discriminating power P_{i+1} of the updated contrast set $CS'(C_g)$ is greater than P_i and greater than the discriminating power of the set caused by adding any other features.

$$P_{i+1}\,[CS'(C_g) = CS(C_g) \cup a_r\,] > P_i\,[CS(C_g)] \quad (2)$$

$$P_{i+1}\,[CS(C_g) \cup a_r] > P_{i+1}\,[\,CS(C_g) \cup a_t], \forall t \neq r \quad (3)$$

In our paper, the discriminating power of a contrast set is measured by the classification accuracy of unseen instances of a certain class, C_g, against the rest of the classes. Note that the empirical classification accuracy is measured class-wise. For each class C_g, when measuring the classification accuracy, the database is first partitioned into two partitions: one partition contains the feature vectors of C_g and the other partition contains the feature vectors of the other classes. Then,

the classification accuracy is measured as the average classification of all unseen instances of C_g. Therefore, our selection criteria might result in different number of features for each class.

Definition 3. *Contrast set length*: the length of a contrast set of a class is the number of features in $CS(C_g)$.

$$Len_{CS(Cg)} = COUNT(CS(C_g)) \leq s \quad (4)$$

The function *COUNT*() simply counts the number of features in a contrast set C_g. The contrast set length is smaller than, or in the worst case equal to, the length of the original vectors, s. Furthermore, the length of the contrast set of the different classes may not necessarily be the same. This is because the number of features to be added to the contrast set of a class is based on whether the feature is positively contributing to the discriminating power of the class or not.

4. Contrast Set Discovery

Given a labeled training database D, a test database T and user-defined classes (C_1, C_2, ..., C_k), the best possible contrast sets that result in the highest discriminating power of each class can be extracted by exploring all possible combinations of features. Such an exploration task is prohibitively expensive due to the exponential number of feature combinations.

The fundamental complexity of this problem cannot be changed[4]; however, in this paper we propose a heuristic greedy algorithm DisCoSet that reduces the search space to discover efficiently the contrast sets. The discovered contrast sets are much smaller in length as compared to the original feature vectors and at the same time, they demonstrate higher discriminating power as well. The discovered contrast sets are local to their classes. Thus, every class has its own local contrast set as opposed to the other methods that find one global contrast set for all classes.

DisCoSet draws ideas from the feature selection algorithms, see Refs. 37-38, which select the relevant global features from a dataset; however, DisCoSet discovers the contrast sets of features that are local to every class. The benefit of feature selections is to discard the irrelevant features and perform the classification task only on the selected subset of features. As a result, the running time cost of the system is reduced.

Algorithm: DisCoSet

Input: labeled training database $D = (v_1, v_2, ..., v_n)$, where each $v_j = (a_{j1}, a_{j2}, ..., a_{js})$ is a set of features (attributes), user-defined classes (C_1, C_2, ..., C_k) and a test dataset of labeled instances T.

Output: Contrast sets *CS[i];* one contrast set per class

```
1.  for each Ci,  i: 1 to k
2.     CS_accuracy[] = 0
3.     CS(Ci) = ∅
4.     len = 0
5.     repeat
6.           bestFeature = 0
7.           bestFeatureAcc = 0
8.           f = vj
9.           for each feature ajh ∈ vj, where h: 1 to s
10.               CS(Ci)' = CS(Ci) ∪ ajh
11.               f = f - ajh
12.               train classifier with π⟨CS'(Ci)⟩ (D)
13.               test classifier with π⟨CS'(Ci)⟩ (T)
14.               compute curFeatureAcc using f
15.               if curFeatueAcc > bestFeatureAcc
16.                     bestFeatureAcc = curFeatureAcc
17.                     bestFeature = ajh
18.           len = len + 1
19.           CS_accuracy[len] = bestFeatureAcc
20.           CS(Ci) = CS(Ci) ∪ bestFeature
21.    Until{CS_accuracy[len]<=CS_accuracy[len1]}
22.    CS[i] = CS(Ci)
```

A wrapper approach is used by DisCoSet to test and select the interesting features. We perform cross-validation and compare the performance of the classification algorithm with each tested subset in order to select the features with most discriminating power.

In the algorithm described above, for every class C_i, the DisCoSet algorithm starts the search by initializing some variables to help in discovering the contrast set of every class (see lines 1-4). The *CS_accuracy[]* array is used to keep the intermediate classification accuracies of the different lengths of the discovered contrast set. $CS(C_i)$ is used to keep the contrast set of C_i and the current length of $CS(C_i)$ is kept in *len*. In each iteration of the *for* loop between line 5 and line 22, only the feature that improves the classification accuracy the most is added to the contrast set $CS(C_i)$. The

bestFeature and *bestFeatureAcc* variables (lines 6-7) are used to track the feature (attribute) that improves the accuracy the most for the current length of the contrast set and the accuracy value, respectively. The *repeat-until* loop (lines 5-21) determines which set of features to add to the current contrast set. The *for* loop (lines 9-17) determines which feature from the remaining unselected features should be added to the current contrast set. That is, the *for* loop first tentatively adds to the current contrast set one feature that is not already selected (see lines 9-11). Then, the classifier is trained and tested with only the vertically projected training dataset, $\pi_{\langle CS'(Ci)\rangle}(D)$, and test dataset, $\pi_{\langle CS'(Ci)\rangle}(T)$, respectively, on the current contrast set (see lines 12-13). These projections are tentative datasets. The total classification accuracy *curFeatureAcc* resulting from adding the current feature is computed (see line 14). Then, in lines 15-17, the algorithm keeps the highest accuracy computed so far in *curFeatureAcc* and the feature that resulted in the highest accuracy so far in *bestFeature*. After determining the feature that gives the highest accuracy, this feature is added to the contrast set and the length of the contrast set is incremented. This process of adding features to the contrast set continues until adding a new feature to the current contrast set reduces the total classification accuracy. This algorithm discovers a single contrast set for each class (see line 22).

To clarify the above steps, let us say we are finding the contrast set of class# 1 (C_1). For example, if features number 3, 11, 8 have been selected in previous iterations of the *repeat until* loop, then $CS(C_1)$ = (3, 11, 8). In the current iteration, each of the features that are not in $CS(C_1)$ will be tested by adding them individually to $CS(C_1)$. Say in the current iteration of the *for* loop, feature 21 is being tested; thus, $CS(C_1)'$ = (3, 11, 8, 21). Then, a vertical projection of the training, $\pi_{\langle CS'(Ci)\rangle}(D)$, and test ,$\pi_{\langle CS'(Ci)\rangle}(T)$, datasets on these four features is applied. That is, in the current projected datasets, each vector only contains the corresponding four features (3, 11, 8, 21). The classifier is trained using the current projected training dataset and then the total classification accuracy *curFeatureAcc* is computed on the current projected test dataset. In the next iteration of the *for* loop, say feature 22 will be tested, therefore $CS(C_1)'$ = (3, 11, 8, 22) and the process repeats to find the total classification accuracy *curFeatureAcc,* and so on. At the end of the *for* loop, the feature that improves the total classification the most is added to the contrast set. Then, in the next iteration of the repeat until loop, a contrast set of length 5 is tested, and so on. The process to find the contrast set for C_1 stops if none of the tested features improves the total accuracy, and thus the features that are already added to $CS(C_1)$ constitute the contrast set for C_1. These steps are repeated when finding the contrast set of class# 2 (C_2), and so on. This class-wise classification is the reason why the contrast sets of different classes could be different in length and features.

As mentioned above, a naïve approach generates and tests all possible combinations to discover the best possible contrast sets. However, this would be computationally prohibitive and impractical. The proposed DisCoSet algorithm avoids the generation and evaluation of such huge number of candidate sets. Let the number of user defined classes be k in a database of n instances (feature vectors) and the number of features in the original vector space be s, then the maximum number of candidate contrast sets for every class is $s+(s-1)+(s-2)+\ldots+1 = s(s-1)/2$, which gives a complexity of $O(s^2)$. However, the DisCoSet algorithm employs an early stopping mechanism that reduces the value s as follows. During each iteration, the algorithm may stop testing and adding new features to the contrast set if adding a new feature causes a reduction in the classification accuracy (see line 20 of the DisCoSet algorithm). Thus the total number of comparisons to discover all contrast set of all k classes is $k*n*s*(s-1)/2$, which gives a complexity of $O(kns^2)$.

Classifying Unseen instances: Given an unseen instance v_j, if we assume the length of the feature vector is s, then the unseen instance feature vector is represented by $v_j = (a_{j1}, a_{j2}, \ldots, a_{js})$, which is a set of features (attributes). To classify v_j into one of the k user-defined classes ($C_1, C_2, \ldots, C_k$), assuming that the contrast sets of the k classes, $CS(C_1), CS(C_2), \ldots, CS(C_k)$, have been computed, a vertical projection of v_j on each of the contrast sets of the k classes is calculated $\pi_{\langle CS(Ci)\rangle}(v_j)$. The k vertical projections of v_j are inputted into the k classifiers. The unseen instance is then classified to the class that gives the best classification accuracy out of the k classes.

5. Experimental Results

In this section, we discuss the results of discovering contrast sets obtained by the proposed algorithm. We implemented the algorithm using Python 2.7 and executed it on a Windows 7 environment with Intel Quad Core™ i7-2600, 3.4 GHz CPU, 64-bit processor and 4 GB of memory.

5.1. *Datasets*

DisCoSet was applied on different datasets with varying number of attributes and number of classes:

- **Image** dataset used by Al Aghbari[40] in which each image is of size 256 × 256 and stored as portable pixmap (ppm) type image. The collection of 420 `images` contains 12 different classes (`Beach, Garden, Desert, Snow, Sunset, Rose, Banana, Tomato, Copper, Tiger, Wood, Gorilla`). These images, which are used in the experiments, were taken from the Corel image collection at UCI Machine Learning Repository.
- 3D motion capture (`mocap`) action data from the **CMU** dataset (http://mocap.cs.cmu.edu). Trajectories of 13 reference points on the human body were projected to six cameras with pre-defined orientations with respect to the human body. We have used 164 sequences in total corresponding to 12 action classes (`golf, walkturn, fjump, flystroke, jjack, jump, cartwheel, drink, kick, walk, bend, run`).
- **Musk** dataset describes a set of 92 molecules of which 47 are judged by human experts to be musks and the remaining 45 molecules are judged to be non-musks. This dataset is available from (http://archive.ics.uci.edu/ml/datasets.html).
- **Optical Recognition of Handwritten Digits** dataset describes normalized bitmaps of handwritten digits from a preprinted form. From the handwriting of 43 people, 30 contributed to the training set and different 13 to the test set. This dataset is available from (http://archive.ics.uci.edu/ml/datasets.html).
- **Breast Cancer Wisconsin (Prognostic)** dataset describes follow-up data, where each record is for one breast cancer case. The dataset includes only those cases exhibiting invasive breast cancer and no evidence of distant metastases at the time of diagnosis. This dataset is available from (http://archive.ics.uci.edu/ml/datasets.html).
- **SPECTF** dataset that describes cardiac Single Proton Emission Computed Tomography images. The database is a summary the original SPECT images, where each image is represented by a 44 continuous feature pattern. The dataset is from (http://archive.ics.uci.edu/ml/datasets.html).
- **Libras Movements** dataset, which contains 15 classes of 24 instances each, where each class represents a hand movement type in Libras. Each movement is mapped in a representation with 90 features. The dataset is from (http://archive.ics.uci.edu/ml/datasets.html).
- **Hill Valley** dataset, which contains records, where each record contains 100 points on a two-dimensional graph. When plotted in order (from 1 through 100) as the Y co-ordinate, the points will create either a Hill or a Valley. The dataset is from (http://archive.ics.uci.edu/ml/datasets.html).

The characteristics of the datasets are shown in Table 1. Training DB column shows the number of training instances used to train the classifier and the Test DB column shows the number of test instances used to extract the contrast sets. The Features column show the number of features (attributes) in each tuple, and the Classes column shows the number of classes in the dataset. For these experiments, we show the result using the SVM classifier. However, other classifiers exhibit similar results.

As mentioned in Section 4, using the group of features that constitute the contrast set of a class $CS(C_i)$, the training and test datasets are vertically projected on $CS(C_i)$. The SVM is re-trained using this vertically projected training dataset. Consequently, the accuracy of $CS(C_i)$ is computed by classifying the vertically projected test dataset using the trained SVM. The same process is repeated for every class. In Table 1, we avoided the problem of overfitting in the training sets by using stratified sampling to create the training sets. The stratification approach ensures that all classes are well represented proportionally in the training dataset. We used the Linear SVM classifier that implements "one-vs-the-rest" multi-class strategy. That is, a single classifier is trained per class to distinguish that class from all other classes. Classification is then performed by using each binary classifier, and choosing the class that gives the highest accuracy.

Table 1. Characteristics of the datasets.

Dataset	Training DB	Test DB	Features	Classes
Images dataset	304	115	64	12
3D mocap dataset	164	164	32	12
Musk dataset	376	100	166	2
Optical Recognition of Handwritten Digits	3823	1797	64	10
Breast Cancer Wisc.	469	100	32	2
SPECTF	187	80	44	2
Libras Movements	210	150	90	15
Hill Valley	606	606	101	2

5.2. *Contrast Sets Discovery*

DisCoSet extracts a local contrast set for each class in the input dataset. As discussed in the previous section, the contrast set of one class could be different from the contrast sets of other classes in terms of selected features, length and discriminating power. For example, Table 2 shows the discovered contrast sets of the **Image dataset**. Notice that the discovered sets of features for every class are different from other classes in terms of selected features and length.

Fig. 1 compares the lengths of the discovered contrast sets with the length of the original feature vector, which is 64 features. Some classes require only few features that give the highest classification accuracy for unseen instances of that class. For example, the lengths of the contrast sets of `Desert` and `Copper` classes are 2 and 3 features respectively, which are found to be enough to give the maximum possible classification accuracy of 100% and 95%, respectively. This is expected as the images of `Desert` and `Copper` classes contain two, or three, dominant colors that distinguish instances of these classes and the existence of other colors in the image does not positively contribute to the discriminating power of these classes. On the other hand, the contrast sets of the `Tomato` and `Banana` classes are relatively longer, which are 26 and 35 features, respectively. Due to the existence of many colors in the images of the `Tomato` and `Banana` classes, the maximum discriminating power of these classes were reached after the inclusion of all the representative features in the classes. However, the average length of contrast sets of all classes of the **Image dataset** that are discovered by DisCoSet is computed to be 12.2 features.

Table 2. Discovered contrast sets of the Image dataset.

#	Class	Contrast Set
1	Beach	36, 41, 10, 57, 21
2	Garden	9, 0, 6, 2, 4
3	Desert	22, 27
4	Snow	0, 1, 5, 43, 12, 2, 59, 44, 17, 25, 28, 14, 6, 54, 57, 31, 8
5	Sunset	0, 1, 43, 2, 4, 26, 5, 12, 6, 24, 18
6	Rose	19, 63, 2, 6
7	Banana	15, 3, 1, 5, 0, 4, 9, 6, 8, 16, 29, 28, 14, 24, 35, 42, 17, 18, 13, 19, 41, 30, 44, 12, 38, 40, 50, 37, 31, 48, 36, 43, 33, 46, 34
8	Tomato	3, 6, 12, 1, 18, 13, 8, 15, 20, 5, 22, 9, 10, 21, 25, 44, 45, 11, 40, 37, 47, 38, 17, 33, 23, 32
9	Copper	43, 49, 15
10	Tiger	0, 1, 3, 2, 4, 5, 6, 7, 8, 9, 52, 22
11	Wood	33, 6, 0, 23, 1, 4, 11, 15, 16, 9, 2, 32
12	Gorilla	0, 1, 2, 3, 4, 6, 5, 7, 12, 9, 25, 14, 11

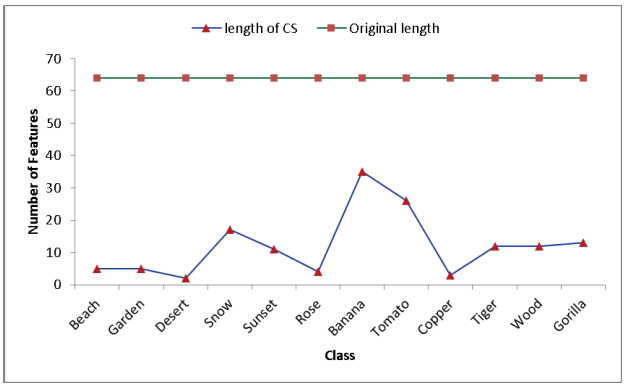

Fig. 1. The relative length of the discovered contrast sets from the Image dataset to the length of the original feature vector.

The DisCoSet algorithm was applied on the **CMU** mocap dataset and the resulting lengths of discovered contrast set are as shown in Fig. 2. In the dataset, 13 joints on the human body are tracked for an entire action sequence. Trajectory of each joint, considered as a time series, is converted to SAX representation of length 32, see Ref. 41. The features here, therefore, are the SAX representation of the original time series. As can be seen in Fig. 2, the average length of the contrast set is now 7.3. Some of the actions, such as `golf swing`, `walkturn`, and `drink` need the contrast set of only length 1 to achieve 100% accuracy. This is understandable, as some of these actions are very

distinct from the other actions in the dataset, hence only a very short contrast set is sufficient to correctly classify the action class. Other sequences, such as `fjump` (forward jump) and `jjack` (jumping jack), require more features to distinguish between other classes of activities such as `jump`.

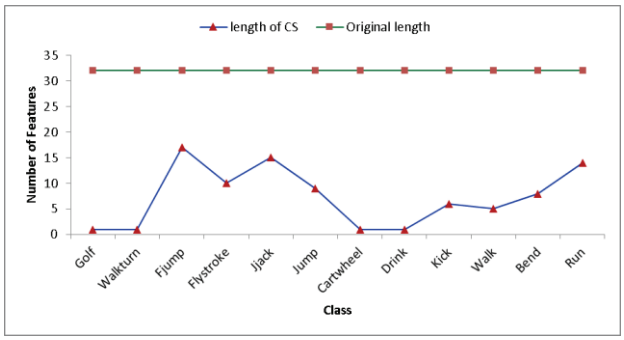

Fig. 2. The relative length of the discovered contrast sets from the 3D mocap action dataset to the length of the original feature vector.

Similarly, we applied DisCoSet on the Musk, Optical Recognition of Handwritten Digits and Breast Cancer Wisconsin (Prognostic) datasets. The lengths of the discovered contrast sets of these datasets are shown in Tables 3, 4 and 5, respectively. For the Musk dataset, seven features out of the 166 features in the original feature vectors are discovered to distinguish the Musk class with 86% accuracy. Note that adding more features to the contrast set do not improve its discriminating power. However, only two features are enough to identify the Non-Musk instances with 100% accuracy. In Table 4, we note that only two features are enough to give 100% accuracy to the handwritten digits of class Nine. On the other hand, a longer contrast set (45 features) is needed to distinguish the instances of class One with an accuracy of 98%, which is due to the existence of the vertical lines in most digits. As shown in Table 5, the Breast Cancer Wisconsin (Prognostic) dataset requires two features to distinguish the Non-recurrent class instances with a 100% accuracy and 12 features to identify the instances of the Recurrent class with a 96% accuracy.

Table 3. Lengths of the discovered contrast sets of the Musk dataset.

Class	Length of Contrast set	Length of Original Vector
Musk	7	166
Non-Musk	2	166

Table 4. Lengths of the discovered contrast sets of the Optical Recognition of Handwritten Digits dataset.

Class	Length of Contrast set	Length of Original Vector
Zero	18	64
One	45	64
Two	34	64
Three	12	64
Four	14	64
Five	44	64
Six	39	64
Seven	42	64
Eight	41	64
Nine	2	64

Table 5. Lengths of the discovered contrast sets of the Breast Cancer Wisconsin (Prognostic) dataset.

Class	Length of Contrast set	Length of Original Vector
Recurrent	12	30
Non-recurrent	2	30

Additional experiments are performed on three datasets (SPECTF dataset, the Libras Movements dataset and the Hill Valley dataset) to determine the length of the discovered contrast sets as shown in Table 6. The length reduction results of the discovered contrast sets were consistent with those of the previous datasets.

Table 6. Average lengths of the discovered contrast sets of the SPECTF, the Libras Movements and the Hill Valley datasets.

Dataset	Average Length of Contrast set	Length of Original Vector
SPECTF	6	44
Libras Movements	38	90
Hill Valley	37	101

From the above discussion, we can conclude that the DisCosSet discovers high discriminating contrast sets and at the same time minimizes their lengths. Previous approaches that rely on extracting rules as contrast sets, end up with a huge number of rules in every contrast set, negatively affecting the efficiency of the classifier.

5.3. *Classification Accuracy of Contrast Sets*

The first objective of discovering the contrast sets is to find attribute-value pairs that are significantly more frequent, and thus have more distinguishing power, in one class than in the other classes. Thus, DisCoSet discovers a minimum set of features (attributes) that can distinguish the respective classes from others. As a result, the discovered contrast set of a class is theoretically smaller, or equal, in length as compared to the original vectors. Experimentally we have shown that the lengths of the discovered contrast sets are much smaller than the original feature vectors (see Fig. 2). Furthermore, a discovered contrast set is more representative and thus is more efficient in classifying unseen instances. The second objective of discovering contrast sets is to improve the classification accuracy.

To demonstrate the effectiveness of the proposed DisCoSet algorithm, we compared the classification accuracy of the discovered contrast sets with the accuracies of two other methods that minimize the number features by computing the best global features of a dataset: (1) the Recursive feature elimination with cross-validation (RFECV) method, see Refs. 42-43, and (2) the method proposed by Al Aghbari.[40] The accuracy comparison is shown in Fig. 3. The RFECV method recursively tunes the number of features selected using cross-validation. Using the Image DB, RFECV selected 22 features as the global minimum number of features for all classes. On the other hand, Al Aghbari's method[40] converts the 256-color feature vectors of the images into SAX representation, which reduces the dimensionality from 256 colors to 64. For a fair comparison, we implemented both algorithms (RFECV method, see Refs. 42-43 and the method proposed by Al Aghbari[40]) and ran them using the same dataset (ImageDB) and in the same computing environment as the proposed DisCoSet algorithm. From Fig. 3, it is clear that the classification accuracies of the contrast sets discovered by the proposed DisCoSet algorithm are higher than those of the RFECV and Al Aghbari[40] methods for all classes. In Table 7, it is shown that the length, or dimensionality, of the discovered contrast sets for the Image DB is on the average 12.2 features, which is lower than those of RFECV and Al Aghbari[40] methods, which are 22 and 64 features, respectively. That is, the DisCoSet was able to reduce the dimensionality significantly compared to the RFECV and Al Aghbari[40] methods and at the same time, DisCoSet has the highest classification accuracy as compared to the other two methods.

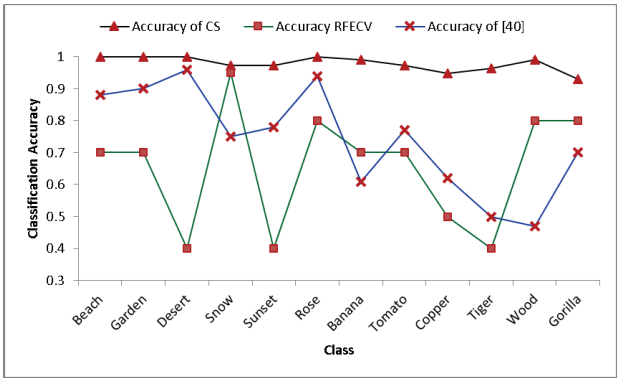

Fig. 3. Comparison of the classification accuracy of the discovered contrast sets with the classification accuracy of the RFECV and Al Aghbari[40] methods.

Table 7. Lengths of the discovered contrast sets of the Breast Cancer Wisconsin (Prognostic) dataset.

Method	Average # Features	Classification Accuracy
Contrast Sets	12.2	0.98
RFECV	22	0.66
Method in Ref.40	64	0.74

As can be seen in Fig. 4, we compare the accuracy obtained by Junejo and Al Aghbari[41] with and without using the DisCoSet. As shown in Table 8, the overall accuracy reported by Junejo and Al Aghbari[41] is 97.4%, whereas with DisCoSet, the accuracy improves to 98.6%. Recognition accuracy for some of the classes, such a `fjump`, `jjack`, `cartwheels`, have now improved to 100%.

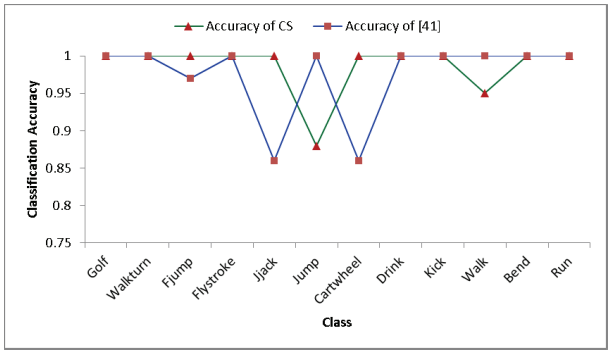

Fig. 4. Comparison of the classification accuracy of the discovered contrast sets with the classification accuracy of the method proposed by Junejo and Al Aghbari.[41]

Table 8. Comparison of the average classification accuracy between DisCoSet and the method by Junejo and Al Aghbari.[41]

Method	Average # Features	Classification Accuracy
Contrast Sets	7.3	0.986
method in Ref. 41	32	0.974

To compare the classification accuracy of the discovered contrast sets with the accuracy of the full feature vectors, we conducted an experiment on eight different datasets, as shown in Fig. 5, and measured the overall classification accuracy of each dataset. For each class C_i in a dataset D, we compute the accuracy, A_{Ci}, using Eq. (5).

$$A_{C_i} = \frac{n_{ci}}{N_i} \tag{5}$$

Where n_{ci} is the number of correctly classified instances (true positives and true negatives) and N_i is the number of all instances in D. Then, we use Eq. (6) to compute the overall classification accuracy of the dataset.

$$Overall\ Accuray = \frac{\sum_{i=1}^{k} A_{C_i}}{k} \tag{6}$$

Where k is the number of classes in D. As shown in Fig. 5, the overall classification accuracy of the dataset when using the contrast sets is higher than the overall classification accuracy when using the whole feature vectors of the instances. We note that the highest accuracy gain in the Libras Movement dataset (about 35%) and the second highest are in the image DB and Musk datasets (about 24%), while the lowest accuracy gain (about 10%) is in the Optical Recognition of Handwritten Digits datasets. Such significant improvements in classification accuracies in all datasets are paralleled by efficient classification due to the reduced number of features. Fig. 6 shows a comparison between the original lengths of the feature vectors and the lengths of the contrast sets. The largest reduction of about 97% in the number of features occurred for the Musk dataset, followed by the next largest reduction of about 86% for the SPECTF dataset. The smallest reduction in the number of features of about 43% occurred for the Optical Recognition of Handwritten Digits dataset.

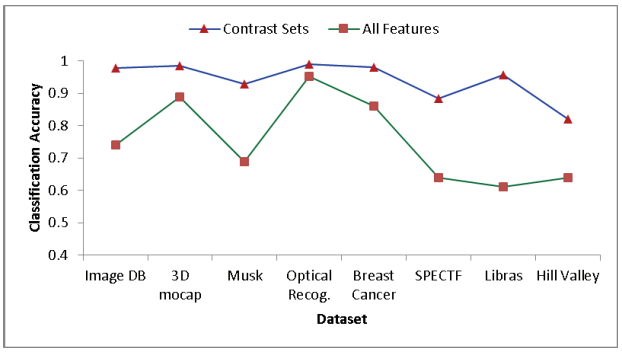

Fig. 5. Comparison of the overall classification accuracy of the discovered contrast sets with that of the whole feature vector of five different datasets.

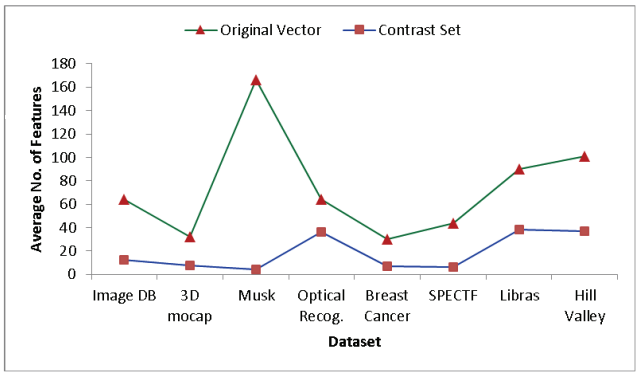

Fig. 6. Comparison of the average length of the discovered contrast sets with the original lengths of eight datasets.

To demonstrate the efficiency of the discovered contrast sets, we compare the execution times of classifying the Test DBs while using the original feature vectors (see Table 1) versus using the discovered contrast sets.

Using the linear SVM classifier, the time saving results are shown in Fig. 7. Note that the saving in execution time using the discovered contrast sets is in the range of 28%-81% of the time taken by the original feature vectors. The results of these experiments prove the efficiency of the contrast sets in classification.

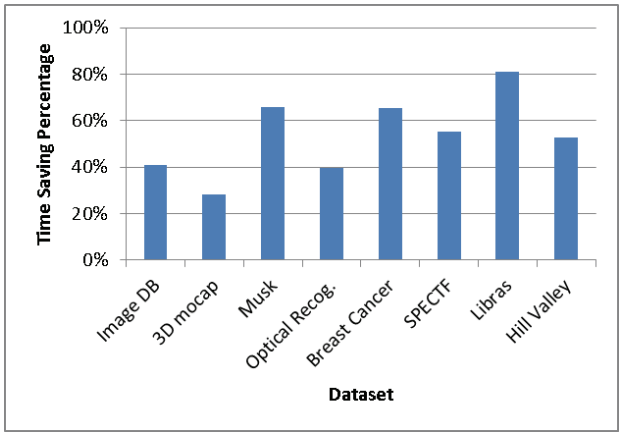

Fig. 7. Comparison of the classification execution time using the discovered contrast sets with the original feature vectors.

Table 9. The time taken to mine the contrast sets of training datasets vs. the time taken to classify the test dataset (see Table 1).

	Mining Time (seconds)	Classification Time (seconds)
Image DB	184.12	0.02570
3D mocap	16.49	0.00011
Musk	245.49	0.06556
Optical Recog.	310.07	0.39023
Breast Cancer	9.27	0.03207
SPECTF	5.87	0.00926
Libras	6.56	0.53192
Hill Valley	162.83	0.05161

Table 9 shows the time taken to discover the contrast sets of features using the training datasets described in Table 1 and it shows the time taken to classify the instances of the test datasets in Table 1. The mining time and classification time of each dataset is dependent on the dataset size, number of classes, and number of features. The mining phase is conducted offline; however, the classification phase is conducted online.

6. Conclusion

In this paper, we present a novel technique to find the set of local features that is much smaller than the original set of features and yet best distinguishes one class from other classes. These sets are extracted from numerical data, without resorting to discretization. In addition, without requiring user-defined threshold like other methods, we show that the proposed method greatly reduces the dimensionality of the feature set and significantly improves the accuracy results. We demonstrate results on various datasets, and show that the proposed method reduces the dimensionality by 40%-97% of the original feature vector length and improves classification to by 10%-24%. We believe these encouraging results demonstrate the practicality and the applicability of the proposed method.

The advantages of the proposed algorithm can be summarized as follows: reduces dimensionality of original feature vectors, improves the classification accuracy, and speeds up the classification process. On the other hand, computing the contrast set during the training phase is computationally expensive, however this phase is usually performed offline. Speeding up the training phase, is one of our future goals.

References

1. T. F. Liao, “Statistical Group Comparison”, Wiley’s Series in Probability and Statistics, 2002.
2. S. D. Bay, M. J. Pazzani, “Detecting Change in Categorical Data: Mining Contrast Sets”. In proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. August 15-18, 1999 San Diego, CA. 302-306.
3. J. Lin, E. Keogh, “Group SAX: Extending the notion of contrast sets to time series and multimedia data. In Proceedings of the 10th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD-06), pages 284–296, 2006.
4. T. Menzies, Y. Hu, “Data Mining for Very Busy People”. IEEE Computer, October, 2003, pp. 18-25.
5. P. K. Novak, N. Lavrac, G. I. Webb, “Supervised Descriptive Rule Discovery: A Unifying Survey of Contrast Set, Emerging Pattern and Subgroup Mining”, Journal of Machine Learning Research, Vol. 10, Feb. 2009, pp. 377-403.
6. S. D. Bay, “Multivariate Discretization of Continuous Variables for Set Mining”, In proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Boston, MA. Aug 20-23, 2000.

7. P. Kralj, N. Lavrac, D. Gamberger, A. Krstacic, "Contrast Set Mining through Subgroup Discovery Applied to Brain Ischaemina Data", PAKKD 2007.
8. G. Dong and J. Li, "Efficient mining of emerging patterns: Discovering trends and differences", In Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-99), pages 43–52, 1999.
9. S. Wrobel, "An algorithm for multi-relational discovery of subgroups", In Proceedings of the 1st European Conference on Principles of Data Mining and Knowledge Discovery (PKDD-97), pages 78–87, 1997.
10. R. J. Bayardo, "Efficiently Mining Long Patterns from Databases", In Proceedings of the ACM International Conference on Management of Data (SIGMOD), 1988, pp. 85-93.
11. G. I. Webb, S. Butler, D. Newlands, "On Detecting Differences between Groups", In the Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining, 2003, pp. 256-265, NY, USA.
12. R. Agrawal, R. Srikantand, "Fast algorithms for mining association rules in large databases", In Proceedings of the 20th International Conference on Very Large Data Bases, VLDB, pages 487-499, Santiago, Chile, September 1994.
13. R. J. Hilderman, T. Pechham, "A Statistically Sound Alternative Approach to Mining Contrast Sets", In Proceedings of the Australian Data Mining Conference, 2005, pp. 157-172, Australia.
14. Z. He, X. Xu, S. Deng, "Mining Cluster-Defining Actionable Rules", In Proceedings of the National Database Conference (NDBC), 2004.
15. B. Minaei-Bidgoli, P-N. Tan, W. F. Punch, "Mining Interesting Contrast Rules for a Web-based Educational System", in proceedings of International Conference on Machine Learning Application. Louisville, KY. Dec 16-18, 2004.
16. J. Lin and E. Keogh. "Group SAX: Extending the notion of contrast sets to time series and multimedia data", In Proceedings of the 10th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD-06), pages 284–296, 2006.
17. G. Dong and J. Li, "Efficient mining of emerging patterns: Discovering trends and differences", In Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-99), pages 43–52, 1999.
18. H. Fan and K. Ramamohanarao, " Efficiently mining interesting emerging patterns", In Proceeding of the 4th International Conference on Web-Age Information Management (WAIM), pages 189–201, 2003.
19. H. Fan, M. Fan, K. Ramamohanarao, M. Liu, "Further improving emerging pattern based classifiers via bagging", In Proceedings of the 10th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD-06), pages 91–96, 2006.
20. A. Soulet, B. Crmilleux, F. Rioult, "Condensed representation of emerging patterns", In Proceedings of the 8th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD), pages 127–132, 2004.
21. J. Li, G. Dong, and K. Ramamohanarao, "Instance-based classification by emerging patterns", In Proceedings of the 14th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD), pages 191–200, 2000.
22. J. Li, H. Liu, J. R. Downing, A. E-J. Yeoh, and L. Wong, "Simple rules underlying gene expression profiles of more than six subtypes of acute lymphoblastic leukemia (ALL) patients", Bioinformatics, vol. 19, no.1, pp. 71–78, 2003.
23. H. S. Song, J. K. Kimb, S. H. Kima, "Mining the change of customer behavior in an internet shopping mall", Expert Systems with Applications, vol. 21, no. 3, pp. 157–168, 2001.
24. X. Ji, J. Bailey, G. Dong, "Mining minimal distinguishing sub-sequence patterns with gap constraints", Knowledge Information Systems, vol. 11, no. 3, pp.259-286, 2007.
25. K. Deng, O. Zaiane, "Contrasting Sequence Groups by Emerging Sequences", Discovery Science, 2009.
26. S. Chan, B. Kao, C. L. Yip, M. Tang, "Mining emerging sub-strings", In Database Systems for Advanced Applications (DASFAA), page 119, 2003.
27. D. Lo, H. Cheng, J. Han, S-C Khoo, "Classification of soft-ware behaviors for failure detection: A discriminative pattern mining approach", In Knowledge Discovery and Data Mining Conference (KDD), 2009.
28. S. Wrobel, "An algorithm for multi-relational discovery of subgroups", In Proceedings of the 1st European Conference on Principles of Data Mining and Knowledge Discovery (PKDD), pp. 78–87, 1997.
29. M. Atzmuller, F. Puppe, "SD-Map - a fast algorithm for exhaustive subgroup discovery", In Proceedings of the 10th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD), pages 6–17, 2006.
30. B. Kavsek, N. Lavrac, "APRIORI-SD: Adapting association rule learning to subgroup discovery", Applied Artificial Intelligence, vol. 20, no. 7, pp. 543–583, 2006.
31. S. Xiang, T. Yang, J. Ye, "Simultaneous feature and feature group selection through hard thresholding", In proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 532-541, 2014.
32. Z. Xu, G. Huang, K. Weinberger, A. Zheng, "Gradient boosted feature selection", In proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 522-531, 2014.
33. W. Klosgen, M. May, "Spatial subgroup mining integrated in an object-relational spatial database", In Proceedings of the 6th European Conference on

Principles and Practice of Knowledge Discovery in Databases (PKDD), pp. 275–286, 2002.
34. M. J. del Jesus, P. Gonzalez, F. Herrera, M. Mesonero, "Evolutionary fuzzy rule induction process for subgroup discovery: A case study in marketing", IEEE Transactions on Fuzzy Systems, vol. 15, no. 4, pp. 578–592, 2007.
35. P. Kralj, N. Lavrac, D. Gamberger, A. Krstacic, "Contrast set mining through subgroup discovery applied to brain ischaemia data", In Proceedings of the 11th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining, (PAKDD), pp. 579-586, 2007b.
36. L. Langohr, V. Podpecan, M. petek, I. Mozeric, K. Gruden, N. Lavrac, H. Toivonen, "Contrast Subgroup Discovery", The Computer Journal, 2012.
37. I. Guyon, A. Elisseeff, "Overfitting in Making Comparisons Between Variable Selection Methods", Journal of Machine Learning Research , vol. 3, pp. 1371-1382, 2003.
38. Z. Al Aghbari, "Classification of Categorical and Numerical data on Selected Subset of Features", Bayesian Networks, Sciyo Publisher, ISBN: 978-953-7619-X-X, Oct. 2010.
39. Y. Liu, J. R. Kender, "Sort-Merge Feature Selection for Video Data", SIAM Data Mining Conference (SDM), 2003, San Francisco, USA.
40. Z. Al Aghbari, "Effective Image Mining by Representing Color Histograms as Time Series" the International Journal of Advanced Computational Intelligence and Intelligent Informatics, vol. 13, no.2, pp.109-114, 2009.
41. I. N. Junejo, Z. Al Aghbari, "Using SAX Representation for Human Action Recognition" Elsevier International Journal of Visual Communication and Image Representation, Vol. 23, no. 6, 2012, pp. 853-861.
42. H. Pang, S. L. George, K. Hui, T. Tong, "Gene Selection Using Iterative Feature Elimination Random Forests for Survival Outcomes", IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 9, no. 5, pp. 1422-1431, 2012.
43. X. Lina, F. Yanga, L. Zhoub, P. Yinb, H. Kongb, W. Xingc, X. Lub, L. Jiad, Q. Wanga, G. Xu, "A support vector machine-recursive feature elimination feature selection method based on artificial contrast variables and mutual information", Elsevier Journal of Chromatography B, 2012.