Selecting Feature Subset via Constraint Association Rules*

Guangtao Wang and Qinbao Song

Dept. of Computer Science and Technology
Xi'an Jiaotong University, China

Abstract. In this paper, a novel feature selection algorithm FEAST is proposed based on association rule mining. The proposed algorithm first mines association rules from a data set; then, it identifies the relevant and interactive feature values with the constraint association rules whose consequent is the target concept, and detects the redundant feature values with constraint association rules whose consequent and antecedent are both single feature value. After that, it eliminates the redundant feature values, and obtains the feature subset by mapping the relevant feature values to corresponding features. The efficiency and effectiveness of FEAST are tested upon both synthetic and real world data sets, and the classification results of the three different types of classifiers (including Naive Bayes, C4.5 and PART) with the other four representative feature subset selection algorithms (including CFS, FCBF, INTERACT and associative-based FSBAR) were compared. The results on synthetic data sets show that FEAST can effectively identify irrelevant and redundant features while reserving interactive ones. The results on the real world data sets show that FEAST outperformed other feature subset selection algorithms in terms of average classification accuracy and Win/Draw/Loss record.

Keywords: Feature subset selection, association rule, feature interaction.

1 Introduction

Feature subset selection is an important research issue in the domains of machine learning and data mining. Its purpose is to help the learning algorithm focus on those aspects of the data most useful for analysis and future prediction. Generally, feature subset selection is the process of identifying and removing as many irrelevant and redundant features as possible. As irrelevant features do not contribute to the predictive accuracy [13], and redundant features do not contribute to getting a better predictor for that the most information they provide is already present in other feature(s) [28], thus many feature subset

* This work is supported by the National Natural Science Foundation of China under grant 61070006.

P.-N. Tan et al. (Eds.): PAKDD 2012, Part II, LNAI 7302, pp. 304–321, 2012.
© Springer-Verlag Berlin Heidelberg 2012

selection algorithms have been proposed to handle the irrelevant features or/and redundant features.

However, feature interaction is not a negligible issue in practice [12]. For example, suppose $F_1 \oplus F_2 = Y$, where F_1 and F_2 are two boolean variables, Y represents the target concept, and $\oplus$ represents the *xor* operation. F_1 and F_2 are irrelevant with Y when we consider their discrimination abilities for Y separately, but they become very relevant when we combine them together. Therefore, removing the interactive features will lead to poor predictive accuracy. Thus a feature subset selection algorithm should consist of eliminating the irrelevant and redundant features while taking the feature interaction into consideration. Unfortunately, to our knowledge, only a few algorithms can deal with this situation [12,29].

Association rule mining can discover interesting associations among data items [15], it has been used to build classifiers which show better classification accuracy compared with the other types of classifiers [2,10,19]. Especially, it also has been employed for feature selection recently by Xie et al. [26]. However, Xie et al. only focus on relevant features and do not consider redundant and interactive features.

An association rule is an expression of $A \Rightarrow C$, where A (Antecedent) and C (Consequent) are itemsets. If we view A as the feature(s) and C as the feature(s)/the target concept, association rules can reveal the dependencies between either feature(s) and feature(s) or feature(s) and the target concept. Therefore, it is reasonable and desirable to devise an association rule mining based method to choose feature subset.

In this paper, we propose a Feature subset sElection Algorithm based on aSsociaTion rule mining (FEAST), which can eliminate the irrelevant and redundant features while taking the feature interaction into consideration. Moreover, FEAST uses association as the measure to evaluate the relativity between feature(s) and the target concept, which is quite different from the traditional measures, such as the consistency measure [4,20,29], the dependence measure [9,27], the distance measure [18,21] and the information theory measure [17,23]. The association measure evaluates irrelevant, redundant and interactive features in a uniform way, it is at least a potential alternatives for feature subset selection. The experimental results on the synthetic and real world data sets show the effectiveness of the proposed algorithm.

The rest of the paper is organized as follows: In Section 2, we introduce the related work. In Section 3 we describe some preliminaries. In Section 4, we present the new feature subset selection algorithm FEAST. In Section 5, we provide the experimental results. Finally, in Section 6, we summarize our work and draw some conclusions.

2 Related Work

Feature subset selection has been an active research topic since 1970's, and a great deal of research has been published.

Of the existing research work, most feature selection algorithms can effectively identify the irrelevant features based on different evaluation functions. But not all of them can eliminate the redundant features and take the feature interaction into consideration [3]. Thus, the existing feature selection algorithms can generally be grouped into several categories according to whether or not they can deal with irrelevant features, redundant features and the feature interaction.

Traditionally, feature subset selection research has focused on searching for relevant features. Feature weighting/ranking algorithms [8] weigh features individually and rank them based on their relevance to the target concept. Unfortunately, they are incapable of removing redundant features. Such as well-known Relief and its extension Relief-F [18].

Moreover, along with irrelevant features, redundant features also affect the speed and accuracy of learning algorithms and thus should be eliminated as well [16]. CFS [9], FCBF [27] and CMIM [5] are examples that take into consideration the redundant features. However, they do not handle the feature interaction [29].

Feature interaction has been drawing more attention in recent years. There can be two-way, three-way or complex multi-way interactions among features [7]. Jakulin and Bratko [12] use interaction gain as a heuristic to detect feature interaction. Their algorithms can detect 2-way (one feature and the class) and 3-way (two features and the class) interactions. Zhao and Liu [29] demonstrate that feature interactions can be implicitly handled by a carefully designed feature evaluation metric and a search strategy with a specially designed data structure.

Recently, association rules have been used for feature selection. Xie et al. [26] propose an association rule-based feature selection algorithm FSBAR. Unfortunately, it just detects relevant features and does not handle redundant and interactive features. In contrast, our algorithm aims to eliminate the irrelevant and redundant features, and takes the multi-way feature interactions into consideration, hence it is quite different from these algorithms above.

3 Preliminaries

3.1 Strong, Classification and Atomic Association Rules

Association rule mining searches for interesting relationships among items in a data set D. Let $I = \{i_1, i_2, \cdots, i_k\}$ be a set of items, an association rule is an implication of form $A \Rightarrow B$, where $A \subset I$, $B \subset I$, and $A \cap B = \phi$.

The support and confidence are two important measures of a rule's interestingness.

1. The support of rule $A \Rightarrow B$ is the percentage of instances in D that contain both A and B, denoted as $\text{Support}(A \Rightarrow B) = P(A \cup B)$; this measure reflects the rule's usefulness whose value range is $(0, 100\%]$.
2. The confidence of rule $A \Rightarrow B$ is the percentage value that shows how frequently B occurs among all the instances containing A. It is denoted as $\text{Confidence}(A \Rightarrow B) = P(B|A)$; this measure reflects the rule's certainty whose value range is $(0, 100\%]$.

Typically, association rules are considered interesting if they satisfy minimum support threshold (*minSupp*) and minimum confidence threshold (*minConf*). The *minSupp* and *minConf* can be set by users or domain experts. Based on these two thresholds, strong association rule (SAR) can be defined as follow.

Definition 1. *Strong association rule (*SAR*). A rule r of form $A \Rightarrow C$ is a strong association rule if and only if:*

$$Supp(r) > minSupp \wedge Conf(r) > minConf. \tag{1}$$

Where *Supp(r)* and *Conf(r)* represent the support and confidence of the association rule *r*, respectively.

For the sake of introducing classification association rule (CAR) and atomic association rule (AAR), we first give the concepts of feature value itemset (FVIS) and target value itemset (TVIS).

Let $D = \{d_1, d_2, \cdots, d_n\}$ be a data set of n instances, $F = \{F_1, F_2, \cdots, F_m\}$ be the feature space of D with m features, where F_i is the domain of ith feature and Y be the target concept. The instance d_i of D can be denoted as a tuple (X_i, y_i), where $X_i \in F_1 \times F_2 \times \cdots \times F_m$, and $y_i \in Y$. Then the feature value itemset $\text{FVIS} = \bigcup_{i=1}^{m} F_i$ containing all possible feature values, and the target value item set $\text{TVIS} = Y$.

With the definitions of FVIS and TVIS, classification association rule (CAR) and atomic association rule (AAR) are defined as follows.

Definition 2. *Classification association rule (*CAR*). A rule r of form $A \Rightarrow C$ is a classification association rule if and only if:*

$$r \text{ is a SAR } \wedge A \subseteq \text{FVIS } \wedge C \subseteq \text{TVIS} \wedge \mid C \mid = 1. \tag{2}$$

Here, $|X|$ denotes the cardinality of set X. All CARs constitute *classification association rule set* (CARset).

Definition 3. *Atomic association rule (*AAR*). A rule r of form $A \Rightarrow C$ is an atomic association rule if and only if:*

$$r \text{ is a SAR} \wedge \mid A \mid = 1 \wedge \mid C \mid = 1. \tag{3}$$

All AARs excluding atomic classification rules constitute *atomic association rule set* (AARset). Here, an atomic classification rule is an AAR whose consequent is the target concept value.

3.2 Definitions of Relevant, Redundant and Interactive Features

To define the relevant, redundant features and feature interaction based on constraint association rules (i.e., classification and atomic association rules), we firstly give the definitions of relevant feature value, redundant feature value and feature value interaction based on association rules.

Definition 4. *Relevant feature value (*RelFV*). A specific value f_{ij} of feature F_i is relevant to the target concept Y if and only if:*

$$\exists r \in \text{CARset}, f_{ij} \in r.Ante. \tag{4}$$

*Otherwise, f_{ij} is an irrelevant feature value (*iRelFV*).*

Where f_{ij} denotes the jth ($1 \leq j \leq |F_i|$) value of feature F_i, and $r.Ante$ represents the antecedent of rule r. The same notations are employed in the following definitions.

From Definition 4 we can know that, the feature values appeared in the antecedent of a rule $r \in$ CARset are relevant feature values; on the other hand, the feature values never appeared in the antecedent of any rule $r \in$ CARset are irrelevant feature values.

We know that classification association rules have been extensively employed in classification [2,10,19], and these classifiers usually possess preferable classification accuracy. This indicates that the rules in CARset can be used to effectively explore the relationship between features and target concept. The feature values appeared in the antecedents of CARs are necessary and related to the target concept. Thus, it is reasonable to identify the relevant feature values by Definition 4.

However, the feature values appeared in a rule's antecedent maybe redundant. That is, two closely-correlated feature values will be simultaneously appearing in the rule's antecedent. This is because that the association rules are generated based on frequent itemset mining (FIM) [24], but FIM cannot detect the redundant items (i.e., feature values) since that, for a given feature value, if it is frequent and selected into a frequent itemset, then the value being redundant to it will be frequent and selected into an itemset as well. To handle this problem, the redundant feature value is defined as follow.

Definition 5. *Redundant feature value (*RedFV*). A specific value f of a feature value set (*FVset*) is redundant if and only if:*

$$\exists r \in \text{AARset}, (\{f\} = r.Cons) \wedge (r.Ante \subseteq \text{FVset}). \tag{5}$$

Where $r.Ante$ and $r.Cons$ represent the antecedent and consequent of rule r, respectively.

From Definition 5 we can know that, of a given feature value set, a feature value is redundant when it appeared in the consequent of a rule in AARset and the rule's antecedent is in the given feature value set as well.

As we known, for a redundant feature value, the information it provides is already present in other feature value. This indicates that it is closely related to and can be replaced by other feature value. What's more, atomic association rule can be used to explore the correlation between two feature values. Thus, Definition 5 based on AAR can be used to detect redundant value.

It is noticed that Definition 5 only shows the two-way value redundancy (the redundancy between two values). Of course, there might exist multi-way feature

value redundancy (the redundancy among multiple feature values). However, detecting all the multi-way value redundancy is a combination explosion problem since we need to list all possible combinations. This is impracticable even when the feature space is of a middle size. Therefore, we just focus on the two-way redundancy in this paper.

Suppose FVset $= \{f_1, f_2, \cdots, f_k\}$ is a feature value set with k feature values. It is a value-assignment set of a feature set Fset with k features, that is, each member of FVset corresponds to exactly a value of the feature of Fset. Let ($A \subset$ FVset) $\neq \phi$ and $B =$ FVset $- A$, y be a value of the target Y, $Conf(r)$ be the confidence of an association rule r, and r_F, r_A and r_B be the CARs of FVset $\Rightarrow \{Y = y\}$, $A \Rightarrow \{Y = y\}$ and $B \Rightarrow \{Y = y\}$, respectively. Then, the interactive feature value can be defined as follow.

Definition 6. *k-th feature value interaction. The k feature values in* FVset *are said to interact with each other if and only if:*

$$Conf(r_F) > Conf(r_A) \ \wedge \ Conf(r_F) > Conf(r_B). \tag{6}$$

The confidence of an association rule shows how well the rule's antecedent describes its consequent. The higher confidence means the stronger description ability. In Definition 6, the confidence of rule r_F is greater than those of rules r_A and r_B. This means that although either feature value set A or B is not helpful in describing the target concept, FVset $= A \cup B$ works well in describing the target concept. In this case, feature value sets A and B are said to interact with each other.

According to Definition 2, the classification association rules usually have high confidence since their confidence should be at least greater than *minConf*. This implies that all the rules with high confidence are included in CARset. In Definition 6, it is impossible that r_A or r_B is a CAR but r_F is not a CAR, since $Conf(r_F)$ is greater than both $Conf(r_A)$ and $Conf(r_B)$. Therefore, the antecedents of rules in CARset will contain all possible feature value interactions according to Definition 6. That is, the feature value interaction can be reserved by the rules in CARset.

Based on the definitions of relevant feature value (RelFV), redundant feature value (RedFV) and feature value interaction, relevant feature, redundant feature and feature interaction are defined as follows.

Definition 7. *Relevant feature (*RelFea*). Feature F_i is relevant to the target concept Y if and only if:*

$$\exists f_{ij} \in F_i, \{f_{ij} \mid f_{ij} \textit{ is a } \text{RelFV}\} \neq \phi. \tag{7}$$

*Otherwise, F_i is an irrelevant feature (*iRelFea*).*

Definition 7 shows that a feature is relevant when at least one of its values is a relevant feature value. On the other hand, for an irrelevant feature, all its values are irrelevant.

Definition 8. *Redundant Feature (*RedFea*). Feature F_i is redundant if and only if:*

$$\forall f_{ij} \in F_i, \{f_{ij} \mid f_{ij} \textit{ is a } \text{RedFV } \textit{or an } \text{iRelFV}\} \neq \phi. \tag{8}$$

Definition 8 indicates that a feature is redundant due to two reasons: (i) each value of this feature is a redundant feature value; (ii) some values of this feature are redundant while others are irrelevant. As irrelevant values provide no information about the target concept and redundant values provide the information which is present by the other values, they are all useless in describing the target concept. This is consistent with the property of the classical definition of redundant feature [28].

Definition 9. *Feature interaction. Let* Fset $= \{F_1, F_2, \cdots, F_k\}$ *be a feature subset with k features, and* VAset *be its value-assignment sets. Features* $F_1, F_2, \cdots, F_k$ *are said to interact with each other if and only if:*

$$\exists \text{fset} \in \text{VAset}, \{\text{fset } \textit{is a } \text{FVset } \textit{with k-th feature value interaction}\} \neq \phi. \tag{9}$$

As we known, there is an intrinsic relationship between a feature and its values, and the properties of a feature subset can be studied by its value-assignment. Thus, for a given feature subset, it is reasonable that the feature interaction among this feature subset could be implied and studied by that among its value-assignment. Inspired by this, Definition 9 based on feature value interaction is proposed to identify feature interaction.

4 Feature Subset Selection Algorithm

Based on the definitions of relevant feature, redundant feature and feature interaction, we propose a novel feature subset selection algorithm FEAST, which searches for relevant features while taking into consideration redundant features and feature interaction.

4.1 FEAST Algorithm

The algorithm FEAST consists of four steps: i) *Association rule mining*, ii) *Relevant feature value set discovery*, iii) *Redundant feature value elimination* and iv) *Feature subset identification*.

1) *Association rule mining*
   Constraint association rules are mined from the given data set based on the predetermined thresholds *minSupp* and *minConf*. These rules include classification association rules and atomic association rules. After this step, classification association rule set (CARset) and atomic association rule set (AARset) are obtained.
2) *Relevant feature value set discovery*
   By collecting the antecedents of rules in CARset together, initial relevant feature value set (RFVset), which reserves the feature value interactions, is achieved according to Definition 4 and Definition 6.

3) *Redundant feature value elimination*
A feature value is redundant means that the information it provides is already present in another feature value. This indicates the redundant value is implied by another value. In this paper, atomic association rule is employed to identify this kind of implication relation. The higher the confidence of an atomic association rule is, the stronger the implication. This means that the AARs with higher confidence could be used to identify and eliminate redundant values firstly.

For a given AAR $r \in$ AARset with the highest confidence, the feature value in r's consequent is identified redundant and eliminated from current RFVset. Meanwhile, according to Definition 5, a feature value in the consequent of an AAR is redundant iff the feature value of the AAR's antecedent is in the current RFVset. Therefore, after eliminating r's consequent from RFVset, AARset should be updated by removing r and the rules whose antecedents are equal to r's consequent.

4) *Feature subset identification*
After eliminating redundant feature values, there are no irrelevant and redundant values in RFVset. Meanwhile, step 2 shows that RFVset includes all feature value interactions based on which the feature interactions are defined (see details in Definition 9). Thus, according to Definition 7, by mapping the feature values in RFVset to the corresponding features, the final feature subset is identified, which not only retains relevant features and excludes irrelevant and redundant features, but also takes feature interaction into consideration.

Algorithm 1 shows the pseudo-code description of FEAST. Of the input parameters, *minSupp* and *minConf* are used as the constraint conditions to achieve strong association rule SAR (Definition 1).

The pseudo-code of FEAST includes four parts, in part 1 (lines 1-2), classification association rule set CARset and atomic association rule set AARset are mined by function *FP_growth* [11] on the given data set D according to *minSupp* and *minConf*. In part 2 (lines 3-4), the union of the antecedents of the association rules in CARset constitutes the relevant feature value set RFVset. Part 3 (lines 5-13) is used to eliminate the redundant feature values in RFVset, where function *Sort* sorts the rules in AARset in descending order of rule's confidence. Firstly, the first rule (i.e. the rule with the highest confidence) r is chosen and removed from AARset. Then if its antecedent is a subset of the current RFVset, the value in r's consequent is eliminated from RFVset; meanwhile, the rules whose antecedents are equal to its consequent are removed from AARset. This process repeats until that AARset is empty. Part 4 (lines 14-17) achieves the selected feature subset S according to the feature values in RFVset.

Time Complexity Analysis. In part 1, the CARset and AARset are mined by function *FP_growth*. Since the time consumption of FP-growth is closely related to the value of *minSupp* [11], the time complexity of this part can be represented as *O(f(minSupp, D))*, where *f(minSupp, D)* is a function of *minSupp* and D which increases with the decrease of *minSupp*/increase of the size of D. For part

Algorithm 1. FEAST

```
inputs : D - the given data set;
         minSupp - the support threshold;
         minConf - the confidence threshold.
output: S - selected feature subset.
//– Part 1 : Association rule mining –
S = φ; RFVset = φ;//RFVset- relevant feature value set;
[CARset, AARset ] = FP_growth (D, minSupp, minConf);
//– Part 2 : Relevant feature value set discovery –
for each r ∈ CARset do
    RFVset = RFVset ∪ r.Antecedent;
//– Part 3: Redundant feature value elimination –
Sort (AARset); //sort rules in descending order of rule's confidence
while AARset ≠ φ do
    r = the first rule in AARset;
    AARset = AARset − {r};
    if r.Antecedent ⊂ RFVset then
        RFVset = RFVset − r.Consequent;
        for each r′ ∈ AARset do
            if r′.Antecedent == r.Consequent then
                AARset = AARset − {r′};
//– Part 4: Feature subset identification –
for each feature value val ∈ RFVset do
    if val ∈ value set of feature F then
        S = S ∪ {F};
return S
```

2, once a CAR is generated by FP-growth, its antecedent could be merged into RFVset meanwhile, so the consumed time of this part can be ignored. For part 3, since its main time consummation is the process of sorting the rules in AARset, the time complexity of this part is $O(V \cdot \log V)$ (by quick sort), where V is the number of rules in AARset. The time complexity of part 4 is $O(K)$ where K is the number of feature values in the final RFVset whose maximum value is the number of all possible feature values in D.

Consequently, the time complexity of FEAST is $O(f(minSupp, D) + O(V \cdot \log V) + O(K)$. Since part 1 is the major time consumer in the worst case, the efficiency of FEAST depends largely on that of association rule mining.

5 Experimental Results and Analysis

In this section, we empirically evaluate the performance of FEAST, and present the experimental results compared with the other four representative feature selection algorithms upon both synthetic and real world data sets.

5.1 Benchmark Data Sets

Synthetic Data Sets. In order to directly evaluate how well FEAST deals with irrelevant, redundant features and feature interaction, five synthetic data sets with all the irrelevant, redundant and interactive features being known are employed.

The first two data sets synData1 and synData2 are generated by the data generation tool RDG1 of the data mining toolkit WEKA[1]. The other three data sets about MONK's problems are available from UCI Machine Learning Repository [1]. The five data sets are described as follows.

1) synData1. There are 100 instances and 10 boolean features $a_0, a_1, \cdots, a_9$. The target concept c is defined by $c = (a_0 \wedge a_1 \wedge \overline{a_5}) \vee (a_0 \wedge \overline{a_1} \wedge a_6 \wedge a_8) \vee (a_0 \wedge a_1 \wedge a_5 \wedge a_8) \vee (\overline{a_0} \wedge a_1 \wedge a_5 \wedge \overline{a_8}) \vee (a_5 \wedge a_6 \wedge a_8) \vee (a_0 \wedge \overline{a_1})$.
2) synData2. There are 100 instances, 11 boolean features denoted as $a_0, a_1, \cdots, a_9$ and a redundant feature r that is the copy of a_5. The target concept c is defined by $c = \overline{a_5} \vee (\overline{a_1} \wedge \overline{a_6} \wedge \overline{a_8})$.
3) MONK1. There are 432 instances and 6 features $a_1, a_2, \cdots, a_6$. The target concept c is defined by $c = (a_1 = a_2) \vee (a_5 = 1)$.
4) MONK2. There are 432 instances and 6 features $a_1, a_2, \cdots, a_6$. The target concept c is defined by exactly two of $\{a_1 = 1, a_2 = 1, \cdots, a_6 = 1\}$.
5) MONK3. There are 432 instances and 6 features $a_1, a_2, \cdots, a_6$. The target concept c is defined by $c = (a_5 = 3 \wedge a_4 = 1) \vee (a_5 \neq 4 \wedge a_2 \neq 3)$. 5% class noise was added to the training set.

For each data set, the features appearing in the definition of the target concept are all relevant, while the absent features are either redundant or irrelevant. The conjunctive terms in the target concept's definition imply feature interactions.

Real World Data Sets. 14 extensively used real world data sets, which are available from from UC Irvine Machine Learning Repository [1], are employed. Table 1 summarizes the 14 data sets in terms of number of features (denoted as F), the number of instances (denoted as I), the number of target concept values (denoted as T). The sizes of data sets vary from 57 to 20,000 instances, and the total number of original features is up to 240. Note that for the data sets containing continuous-value features, if needed, we apply the MDL discretization method (available in WEKA).

Table 1. Summary of the 14 real world data sets

Data set	F	I	T	Data set	F	I	T
heart-c	11	303	5	autos	22	205	7
cleve	12	303	2	mushroom	22	8124	2
austra	14	690	2	colic-orig	23	368	2
labor	14	57	2	flags	26	194	6
letter	15	20000	26	molecular	57	106	2
primary-tumor	17	339	22	splice	60	3190	3
lymph	18	148	4	mfeat-pixel	240	2000	10

5.2 Experimental Setup

1) Four representative feature selection algorithms were selected to be compared with FEAST.

These algorithms include two well-known and frequently-used CFS [9] and FCBF [27]. They can effectively identify irrelevant features while taking consideration of the redundant features.

[1] `http://www.cs.waikato.ac.nz/ml/weka/`

To further study the performance of FEAST in terms of handling feature interaction, an algorithm INTERACT [29], which is specifically proposed to address the feature interaction, is selected as one benchmark algorithm.

Moreover, since our proposed FEAST is an association-rule-based feature selection algorithm, a latest association-rule-based feature selection algorithm FSBAR [26] is selected as well.

The parameters of these algorithms (including FEAST) were determined by the cross-validation strategy.

2) Classification accuracy over selected feature subset is extensively used as a measure to evaluate the performance of the feature selection algorithm in feature selection literature. This is due to the fact that the relevant features of real world data sets are usually not known in advance, and we can not directly evaluate how good a feature selection algorithm is by the features selected.

However, different classification algorithms have different biases, and a feature subset selection algorithm may be more suitable for some classification algorithms than others. With this in mind, three different types of well-known classification algorithms including probability-based Naive Bayes [14], decision tree-based C4.5 [22] and rule-based PART [6] were selected.

In order to make best use of the data set and get stable results, the classification accuracies before and after feature selection were obtained by a 5×10-fold cross-validation procedure. That is, for a given data set, each feature selection algorithm and each classifier were repeatedly performed on the data set with 10-fold cross-validation by five times.

3) All the experiments were conducted in the WEKA environment [25].

5.3 Results on the Synthetic Data Sets

Table 2 shows the feature subsets selected by the five feature subset selection algorithms on the five synthetic data sets. In this table, '_' indicates a missing relevant feature, and the letter in bold type indicates an irrelevant or a redundant feature selected by mistake. The last row "Relevant features" reports the actual relevant features of each data set.

Table 2. Features selected by the five algorithms on the synthetic data sets

FSS algorithm	synData1	synData2	MONK1	MONK2	MONK3
CFS	$a_0, \_, a_5, a_6, a_8$	$\boldsymbol{a_0}, a_1, a_5, \_, \boldsymbol{a_7}, \_, \boldsymbol{r}$	$\_, \_, a_5$	$\_, \_, \_, \_, a_5, \_$	$a_2, \_, \_$
FCBF	$a_0, \_, a_5, a_6, a_8$	$\boldsymbol{a_0}, a_1, a_5, \_, \boldsymbol{a_7}, \_$	$\_, \_, a_5$	$a_1, a_2, a_3, a_4, a_5, a_6$	a_2, a_4, a_5
FSBAR	$a_0, a_1, \boldsymbol{a_3}, a_5, a_6, a_8$	$\boldsymbol{a_0}, a_1, a_5, a_6, a_8$	$\_, \_, a_5$	$a_1, \_, \_, \_, \_, \_$	$a_2, \_, a_5$
INTERACT	a_0, a_1, a_5, a_6, a_8	$a_1, \boldsymbol{a_3}, \boldsymbol{a_4}, a_5, a_6, \boldsymbol{a_7}, \_$	a_1, a_2, a_5	$a_1, a_2, a_3, a_4, a_5, a_6$	a_2, a_4, a_5
FEAST	a_0, a_1, a_5, a_6, a_8	a_1, a_5, a_6, a_8	a_1, a_2, a_5	$a_1, a_2, a_3, a_4, a_5, a_6$	a_2, a_4, a_5
Relevant features	a_0, a_1, a_5, a_6, a_8	a_1, a_5, a_6, a_8	a_1, a_2, a_5	$a_1, a_2, a_3, a_4, a_5, a_6$	a_2, a_4, a_5

From Table 2, we observe that: (i) Only algorithm FEAST removes all irrelevant features while reserving all relevant features for all the five data sets. The other algorithms identify the irrelevant on some but not all data sets. (ii) Except algorithm CFS, all other four algorithms can identify and remove the redundant feature r in the data set "synData2". (iii) Only algorithm FEAST

reserves all the interactive features on all the five data sets. INTERACT works well on all the data sets except for "synData2". The other algorithms identify all the interactive features on some but not all the data sets.

5.4 Results on the Real World Data Sets

In this section, we present the comparison results of FEAST with other feature subset selection algorithms in terms of (i) the classification accuracies after feature subset selection; (ii) the proportion of selected features; and (iii) the runtime.

Here, the proportion of selected features is the ratio of the number of features selected by a feature selection algorithm to the original number of features of a data set.

What's more, we also provide the sensitivity analysis results of the support and confidence thresholds on the proposed algorithm FEAST.

Classification Accuracy Comparison. Table 3 records the classification accuracies of Naive Bayes, C4.5 and PART with the five feature subset selection algorithms, and the Win/Draw/Loss records, which are the numbers of data sets where the classification accuracy of the given classifier obtained with FEAST is greater than/equal to/lower than that with the compared feature selection algorithm.

Table 3. Accuracies of Naive Bayes, C4.5 and PART with different feature selection algorithms

Data Set	Naive Bayes						C4.5						PART					
	FEAST	CFS	FCBF	INTERACT	FSBAR	ORG	FEAST	CFS	FCBF	INTERACT	FSBAR	ORG	FEAST	CFS	FCBF	INTERACT	FSBAR	ORG
heart-c	83.46	84.43	84.43	82.90	82.18	84.44	79.80	79.80	79.80	78.88	80.86	78.79	82.13	81.12	81.12	79.80	82.84	78.85
cleve	84.22	84.86	84.86	83.50	82.51	83.85	79.60	79.20	79.20	78.75	78.22	77.90	83.19	80.58	80.58	81.72	78.88	79.57
austra	87.68	85.51	87.10	87.48	86.52	85.22	86.46	85.51	86.52	86.46	87.10	86.70	86.38	85.51	85.07	86.00	86.23	85.80
labor	90.00	89.33	89.33	90.18	89.47	91.67	84.33	80.67	80.67	91.58	85.96	73.68	88.00	80.67	84.00	85.96	85.96	80.67
letter	74.48	73.03	74.48	74.55	NA	74.04	78.98	79.17	79.14	79.08	NA	78.82	81.45	81.41	80.90	81.05	NA	80.69
primary-tumor	47.48	45.70	46.00	49.68	43.95	50.13	43.65	41.56	42.47	41.12	42.18	41.00	43.35	45.39	40.12	40.53	43.07	40.70
lymph	83.62	81.67	80.24	83.24	83.78	83.67	77.62	75.71	70.81	73.51	74.32	78.33	81.71	77.14	78.90	76.08	75.00	79.67
autos	77.95	77.40	69.21	78.15	59.51	71.64	77.98	75.55	67.31	76.98	73.17	83.81	78.95	79.45	67.29	75.90	74.63	78.00
mushroom	95.59	98.52	98.52	98.92	98.92	95.83	100.00	98.52	99.02	100.00	100.00	100.00	100.00	98.52	99.02	100.00	100.00	100.00
colic-orig	83.95	81.52	84.25	70.22	83.15	70.40	85.84	81.52	81.52	66.30	85.33	85.03	85.84	81.52	81.24	66.30	84.24	64.11
flags	79.89	73.13	75.18	70.82	70.1	73.21	71.74	72.24	71.63	70.72	69.59	71.18	70.16	70.58	72.1	66.19	67.53	64.92
molecular	97.18	93.27	95.27	94.53	92.45	90.27	80.91	83.82	82.82	81.70	83.96	80.82	85.82	86.73	84.82	86.98	86.79	82.82
splice	96.24	92.48	96.14	96.13	91.85	95.36	94.54	92.70	94.48	94.31	92.95	94.36	92.76	92.07	93.39	92.93	92.57	92.51
mfeat-pixel	90.95	93.00	91.15	90.45	NA	93.30	77.40	79.60	77.80	80.20	NA	78.65	83.00	84.15	80.95	82.25	NA	82.00
Average	83.76	82.42	82.58	82.20	80.37	81.64	79.92	78.97	78.08	78.54	79.47	79.22	81.62	80.35	79.25	78.69	79.81	77.88
W/D/L	-	10/0/4	8/1/5	9/0/5	10/0/2	8/0/6	-	9/1/4	9/1/4	8/2/4	7/1/4	9/1/4	-	9/0/5	12/0/2	11/0/2	9/1/2	13/1/0

* In this table, "ORG" denotes original data sets, "W/D/L" represents "Win/Draw/Loss", and "NA" means the algorithm is not available.

From Table 3 we observe that:

1) For Naive Bayes, (i) compared to the original data set, the average accuracy of Naive Bayes is improved by all the algorithms except FSBAR; (ii) FEAST outperforms other algorithms in terms of average accuracy, it improves the average accuracy by 2.29% averagely; (iii) FEAST outperforms other algorithms in terms of Win/Draw/Loss record, it wins other algorithms for 9.25 out of 14 data sets on average, while losses only 4 out of 14 on average.
2) For C4.5, (i) compared to the original data set, the average accuracy of C4.5 is improved only by the FEAST and FSBAR, but FSBAR were not available on two data sets due to its high time complexity; (ii) FEAST outperforms other algorithms in terms of average accuracy, it improves the average accuracy

by 1.47% averagely; (iii) FEAST outperforms other algorithms in terms of Win/Draw/Loss record, it wins other algorithms for 8.25 out of 14 data sets on average, while losses only 4 out of 14 on average.

3) For PART, (i) compared to the original data set, the average accuracy of PART is improved by all algorithms; (ii) FEAST outperforms other algorithms in terms of average accuracy, it improves the average accuracy by 2.64% averagely; (iii) FEAST outperforms other algorithms in terms of Win/Draw/Loss record, it wins other algorithms for 10.25 out of 14 data sets on average, while losses only 2.75 out of 14 on average.

Table 4. Proportion (%) of selected features for different feature selection algorithms

Data set	FEAST	CFS	FCBF	INTERACT	FSBAR
heart-c	81.82	54.55	54.55	90.91	90.91
cleve	83.33	50.00	50.00	83.33	83.33
austra	50.00	50.00	50.00	92.86	64.29
labor	57.14	50.00	42.86	50.00	50.00
letter	80.00	73.33	73.33	80.00	NA
primary-tumor	47.06	70.59	64.71	94.12	52.94
lymph	44.44	55.56	44.44	55.56	55.56
autos	27.27	22.73	18.18	27.27	54.55
mushroom	36.36	18.18	18.18	27.27	36.36
colic-orig	26.09	8.70	8.70	21.74	30.43
flags	26.92	11.54	15.38	38.46	57.69
molecular	22.81	10.53	10.53	10.53	12.28
splice	31.67	10.00	36.67	38.33	11.67
mfeat-pixel	48.75	42.92	11.25	14.58	NA
Average	47.40	37.76	35.63	51.78	50.00

Table 5. Runtime (ms) for different feature selection algorithms

Data set	FEAST	CFS	FCBF	INTERACT	FSBAR
heart-c	20	22	20	144	215
cleve	20	76	72	63	412
austra	45	80	83	74	1432
labor	16	62	60	62	18
letter	1190	678	558	5333	NA
primary-tumor	624	74	81	64	228
lymph	51	74	69	89	6786
autos	2216	78	72	82	29206
mushroom	223	238	215	405	242803
colic-orig	31	74	81	88	582
flags	319	22	42	58	2649
molecular	79	82	77	66	631
splice	1890	126	42	889	57435
mfeat-pixel	4250	7287	1696	4514	NA
Average	783.86	640.93	226.29	852.21	28533.08

Proportion of Selected Features Comparison. The reduction on the number of features is an important metric used to evaluate feature subset selection algorithms. This can be measured through the proportion of features selected by the feature selection algorithms.

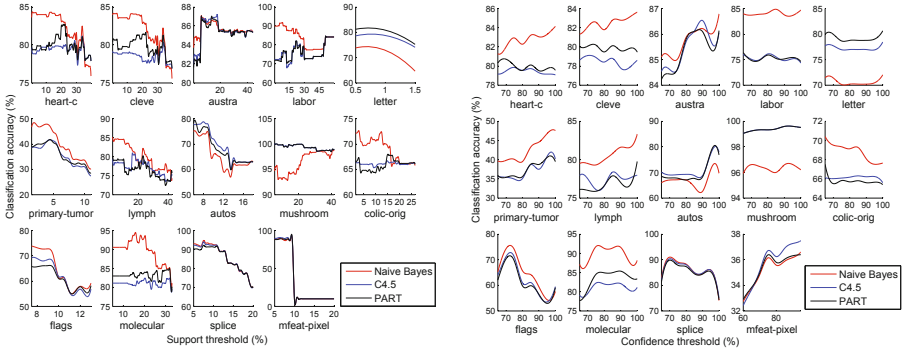
Table 4 presents the proportion of features selected by each of the five feature selection algorithms over the 14 data sets. From this table we observe that: i) All the feature subset selection algorithms could significantly reduce the number of features on average. FCBF ranks 1 with proportion of selected features 35.63%, and INTERACT ranks last with 51.78%. ii) FEAST outperforms algorithms INTERACT and FSBAR in reducing the number of features.

Runtime Comparison. Table 5 records the runtime of each feature subset selection algorithm upon the 14 data sets. From it we observe that (i) the average runtime of different algorithms is varying greatly, FCBF ranks 1 with 226.29 ms, and FSBAR ranks last with 28533.08 ms. (ii) FEAST is faster than INTERACT and FSBAR. Compared with the associative-based algorithm FSBAR, FEAST is much more efficient since it generates association rules by FP-growth algorithm which is more efficient than the Apriori algorithm used in FSBAR.

To summarize, the proposed algorithm FEAST outperformed other feature subset selection algorithms on the 14 UCI data sets in terms of average classification accuracy and Win/Draw/Loss record, and the runtime and the reduction rate are acceptable.

Sensitivity Analysis of the Support and Confidence Thresholds. Support threshold and confidence threshold are two important parameters in the proposed algorithm FEAST. To study how they affect the performance of FEAST, in this part, we give the sensitivity analysis of these two parameters on FEAST in terms of classification accuracy, proportion of selected features and runtime, respectively.

Classification Accuracy. Fig. 1 shows sensitivity analysis results of the support and confidence thresholds on the classification accuracies of the three classifiers with respect to our proposed algorithm FEAST.



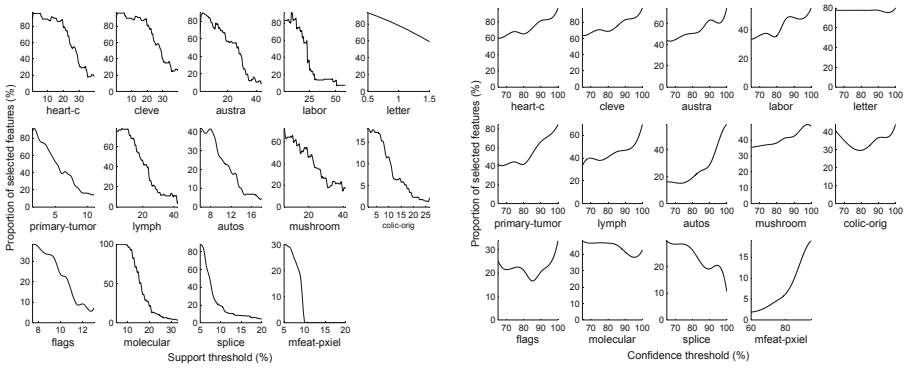
(a) Accuracies vs. Support threshold (b) Accuracies vs. Confidence threshold

Fig. 1. Classification accuracies of the three classifiers with FEAST vs. different thresholds

From Fig. 1(a) and 1(b) we observe that (i) for a given data set, the classification accuracy varying trends of the three classifiers w.r.t FEAST are very similar for either the given support thresholds or the given confidence thresholds. This reveals that the FEAST has no bias for a special classifier, i.e. the results obtained by FEAST are generally suitable. (ii) The classification accuracy varies with both the support and confidence thresholds, and the thresholds corresponding to the highest classification accuracy are different for different data sets. For example, in Fig. 1(a), the support threshold corresponding to the highest classification accuracy is about 10% for "austra", while less than 5% for "colic-orig". In Fig. 1(b), the confidence threshold corresponding to the highest classification accuracy is greater than 95% for "autos", while about 70% for "splice". This implies that both support and confidence thresholds affect the feature subset selected by FEAST, and the best thresholds are different for different data sets.

Proportion of Selected Features. Fig. 2 shows sensitivity analysis results of the support and confidence thresholds on the proportion of features selected by the proposed algorithm FEAST.

From Fig. 2(a) we observe that for all the 14 data sets, with the increment of the support threshold, the proportion of the selected features decreases. The



(a) Proportion of selected features vs. Support threshold

(b) Proportion of selected features vs. Confidence threshold

Fig. 2. Proportion of features selected by FEAST vs. different thresholds

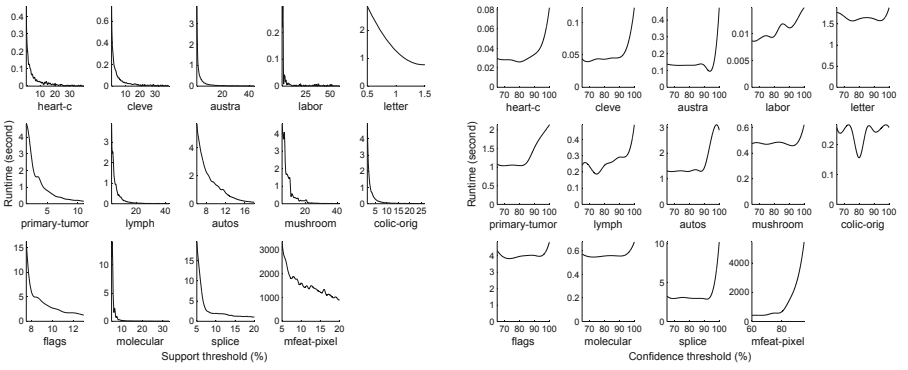
reason is that with the increment of the support threshold, the number of the frequent itemsets decreases. At the same time, FEAST chooses feature subset from itemsets that are at least frequent, thus the number of the selected features deceases, and the proportion of the selected features decreases as well. We also observe that although the proportion of the selected features deceases with the increment of the support threshold, for the different data sets, the decrement extents are varying. Therefore, we should choose different support thresholds for the different data sets.

From Fig. 2(b) we observe that with the increment of the confidence threshold, the proportion of selected features either increases or decreases. The reason is that for a given confidence threshold, there are many support thresholds with varying values. Further, for the different confidence thresholds, the varying ranges of the support thresholds are different. This means the corresponding numbers of the frequent itemsets and further the proportions of selected features are different as well. This reveals that both the support and confidence thresholds are affected by data set characteristics and we should select different thresholds for different data sets.

Runtime Fig. 3 shows the sensitivity analysis results of the support and confidence thresholds on the runtime of our proposed algorithm FEAST.

From Fig. 3(a) we observe that for all the data sets, the runtime of FEAST decreases when the support threshold increases. This is because with the increment of the support threshold, the number of the frequent itemsets is decreased. So the time spending on mining the frequent itemsets is decreased as well. At the same time, FEAST chooses the feature subset from the itemsets that are at least frequent, thus the time consumed in the feature subset identification is also deceased.

From Fig. 3(b) we observe that the runtime of FEAST can increase, decrease and fluctuate when the confidence threshold increases. The reason is that for a given confidence threshold, there are many support thresholds with varying



(a) Runtime vs. Support threshold (b) Runtime vs. Confidence threshold

Fig. 3. Runtime of FEAST vs. different thresholds

values. Further, for the different confidence thresholds, the varying ranges of the support thresholds are different. This means that the corresponding numbers of the frequent itemsets, and further the numbers of selected features are different as well. Thus, the time used to mine frequent itemsets and to identify feature subset is varying.

To summarize, the performance of the proposed algorithm FEAST is directly affected by the selection of these two input-parameters: support and confidence thresholds. However, the appropriate thresholds for different data sets would be different. That is, there are no specific support and confidence thresholds which are the best choice for all the data sets. We should pick up different thresholds for different data sets.

6 Conclusion

In this paper, we have presented a novel constraint association rule based feature selection algorithm FEAST. We have also compared FEAST with the other four representative feature selection algorithms, including two well-known algorithms CFS and FCBF, the algorithm INTERACT aiming at solving feature interaction, and an associative-rule-based algorithm FSBAR, upon both the five synthetic data sets and the 14 UCI data sets. The results on the synthetic data sets show that FEAST can identify relevant features and remove redundant ones while reserving feature interaction. The results on the real world data sets show that our proposed algorithm FEAST can reduce the number of features and outperforms all the other four feature selection algorithms in terms of the average accuracy improvement and the Win/Draw/Loss records of all the three different types of classifiers Naive Bayes, C4.5 and PART.

We have also conducted a sensitivity analysis of support and confidence thresholds to FEAST. The results show that the support and confidence thresholds play a fundamental role in the proposed algorithm. Moreover, for different data sets,

the appropriate thresholds could be different. Therefore, for further research, we plan to explore how to recommend the support and confidence thresholds for FEAST according to data set characteristics.

References

1. Asuncion, A., Newman, D.J.: UCI machine learning repository (2007), http://archive.ics.uci.edu/ml/
2. Chen, G., Liu, H., Yu, L., Wei, Q., Zhang, X.: A new approach to classification based on association rule mining. Decision Support Systems 42(2), 674–689 (2006)
3. Dash, M., Liu, H.: Feature selection for classification. Intelligent Data Analysis 1(3), 131–156 (1997)
4. Dash, M., Liu, H.: Consistency-based search in feature selection. Artificial Intelligence 151(1-2), 155–176 (2003)
5. Fleuret, F.: Fast binary feature selection with conditional mutual information. Journal of Machine Learning Research 5, 1531–1555 (2004)
6. Frank, E., Witten, I.H.: Generating accurate rule sets without global optimization. In: Proceedings of the Fifteenth International Conference on Machine Learning, pp. 144–151. Morgan Kaufmann Publishers Inc. (1998)
7. Gheyas, I.A., Smith, L.S.: Feature subset selection in large dimensionality domains. Pattern Recognition 43(1), 5–13 (2010)
8. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. Journal of Machine Learning Research 3, 1157–1182 (2003)
9. Hall, M.A.: Correlation-based feature selection for discrete and numeric class machine learning. In: Proceedings of the Seventeenth International Conference on Machine Learning, pp. 359–366. Morgan Kaufmann Publishers Inc. (2000)
10. Han, J.: CPAR: Classification based on predictive association rules. In: Proceedings of the Third SIAM International Conference on Data Mining, vol. 3, pp. 331–335. Society for Industrial & Applied (2003)
11. Han, J., Pei, J., Yin, Y., Mao, R.: Mining frequent patterns without candidate generation: A frequent-pattern tree approach. Data Mining and Knowledge Discovery 8(1), 53–87 (2004)
12. Jakulin, A., Bratko, I.: Testing the significance of attribute interactions. In: Proceedings of the 21st International Conference on Machine learning, pp. 409–416. ACM (2004)
13. John, G.H., Kohavi, R., Pfleger, K.: Irrelevant features and the subset selection problem. In: Proceedings of the 11th International Conference on Machine Learning, vol. 129, pp. 121–129. Citeseer (1994)
14. John, G.H., Langley, P.: Estimating continuous distributions in bayesian classifiers. In: Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence, vol. 1, pp. 338–345. Citeseer (1995)
15. Klemettinen, M., Mannila, H., Ronkainen, P., Toivonen, H., Verkamo, A.I.: Finding interesting rules from large sets of discovered association rules. In: Proceedings of the 3rd International Conference on Information and Knowledge Management, pp. 401–407. ACM (1994)
16. Kohavi, R., John, G.H.: Wrappers for feature subset selection. Artificial Intelligence 97(1-2), 273–324 (1997)
17. Koller, D., Sahami, M.: Toward optimal feature selection. In: Proceedings of International Conference on Machine Learning, pp. 284–292. Citeseer (1996)

18. Kononenko, I.: Estimating Attributes: Analysis and Extensions of RELIEF. In: Bergadano, F., De Raedt, L. (eds.) ECML 1994. LNCS, vol. 784, pp. 171–182. Springer, Heidelberg (1994)
19. Li, W., Han, J., Pei, J.: CMAR: Accurate and efficient classification based on multiple class-association rules. In: Proceedings of IEEE International Conference on Data Mining, pp. 369–376. IEEE Computer Society (2001)
20. Liu, H., Setiono, R.: A probabilistic approach to feature selection-a filter solution. In: Proceedings of the 13rd International Conference of Machine learning. Morgan Kaufmann Pub. (1996)
21. Park, H., Kwon, H.C.: Extended relief algorithms in instance-based feature filtering. In: Proceedings of the 6th International Conference on Advanced Language Processing and Web Information Technology (ALPIT 2007), pp. 123–128. IEEE Computer Society (2007)
22. Quinlan, J.R.: C4.5: programs for machine learning. Morgan Kaufmann Publishers Inc., San Francisco (1993)
23. Scanlon, P., Potamianos, G., Libal, V., Chu, S.M.: Mutual information based visual feature selection for lipreading. In: Processings of the 8th International Conference on Spoken Language, pp. 857–860. Citeseer (2004)
24. Tan, P.N., Steinbach, M., Kumar, V.: Introduction to data mining. Pearson Addison Wesley, Boston (2006)
25. Witten, I.H., Frank, E.: Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann Pub. (2005)
26. Xie, J., Wu, J., Qian, Q.: Feature selection algorithm based on association rules mining method. In: Proceedings of 8th IEEE/ACIS International Conference on Computer and Information Science, pp. 357–362. IEEE (2009)
27. Yu, L., Liu, H.: Feature selection for high-dimensional data: A fast correlation-based filter solution. In: Proceedings of 20th International Conference on Machine Leaning, vol. 20, pp. 856–863 (2003)
28. Yu, L., Liu, H.: Efficient feature selection via analysis of relevance and redundancy. Journal of Machine Learning Research 5, 1205–1224 (2004)
29. Zhao, Z., Liu, H.: Searching for interacting features in subset selection. Intelligent Data Analysis 13(2), 207–228 (2009)