Knowledge-Based Systems 89 (2015) 203–217

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Feature selection with redundancy-complementariness dispersion

Zhijun Chen [a,b,e], Chaozhong Wu [a,b], Yishi Zhang [c,f,*], Zhen Huang [d], Bin Ran [e], Ming Zhong [a,b], Nengchao Lyu [a,b]

[a] Intelligent Transport Systems Research Center, Wuhan University of Technology, Wuhan 430063, China
[b] Engineering Research Center for Transportation Safety, Ministry of Education, Wuhan 430063, China
[c] School of Management, Huazhong University of Science and Technology, Wuhan 430074, China
[d] School of Automation, Wuhan University of Technology, Wuhan 430063, China
[e] Department of Civil and Environment Engineering, University of Wisconsin-Madison, Madison, WI 53706, USA
[f] Wisconsin School of Business, University of Wisconsin-Madison, Madison, WI 53706, USA

ARTICLE INFO

Article history:
Received 2 February 2015
Received in revised form 18 May 2015
Accepted 12 July 2015
Available online 18 July 2015

Keywords:
Classification
Feature selection
Relevance
Redundancy
Pairwise approximation
Redundancy-complementariness dispersion

ABSTRACT

Feature selection has attracted significant attention in data mining and machine learning in the past decades. Many existing feature selection methods eliminate redundancy by measuring pairwise inter-correlation of features, whereas the complementariness of features and higher inter-correlation among more than two features are ignored. In this study, a modification item concerning feature complementariness is introduced in the evaluation criterion of features. Additionally, in order to identify the interference effect of already-selected False Positives (FPs), the redundancy-complementariness dispersion is also taken into account to adjust the measurement of pairwise inter-correlation of features. To illustrate the effectiveness of proposed method, classification experiments are applied with four frequently used classifiers on ten datasets. Classification results verify the superiority of proposed method compared with seven representative feature selection methods.

1. Introduction

With the fast development of the world, the dimensional and size of data is fast-growing in most kinds of fields which challenge the data mining and machine learning techniques. Feature selection is an important and useful approach that can effectively reduce the dimensionality of feature space while retaining a relatively high accuracy in representing the original data. Thus, it plays a fundamental role in many data mining and machine learning tasks, particularly in pattern recognition, knowledge discovery, information retrieval, computer vision, bioinformatics, and so forth. The effects of feature selection have been widely recognized for its abilities in facilitating data interpretation, reducing acquisition and storage requirements, increasing learning speeds, improving generalization performance, etc. [1]. Therefore, feature selection has attracted significant attention of more and more researchers [2–8].

Generally speaking, the feature selection methods can be divided into two types: Wrapper and filter. Wrapper methods depend on specific learning algorithms. Thus the performance of wrapper methods is affected by the selected learning methods. This may makes wrapper methods computationally expensive in learning, since they must train and test classifiers for each feature subset candidate. Conversely, filter methods do not rely on any learning schemes. Instead, it is only based on some classifier-irrelevant metrics, including Fisher score [9], χ^2-test [10], mutual information [11–14], Symmetrical Uncertainty(SU) [15], etc., to estimate the discrimination power of features. Recently, new criteria and techniques such as sparse logistic regression attract increasing attention (e.g. [16]) since they have potential ability to handle very high-dimensional datasets. In this study, we only focus on filter methods.

Filter methods can also divided into feature subset selection and feature ranking ones, with regard to their search strategy. The evaluation unit for subset selection methods is a set of features, thus the set with best discrimination power is trying to be discovered [17–19]. Nevertheless, to find the best feature subset, a total of $2^m - 1$ candidate subsets (where m is # features in the original data) are possible to be traversed for feature selection task cannot be solved optimally in polynomial-time unless **P** = **NP** [20].

* Corresponding author at: Wisconsin School of Business, University of Wisconsin-Madison, 975 University Avenue, Madison, WI 53706, USA.
E-mail addresses: chenzj556@gmail.com (Z. Chen), wucz@whut.edu.cn (C. Wu), zhang685@wisc.edu (Y. Zhang), h-zhen@whut.edu.cn (Z. Huang), bran@wisc.edu (B. Ran), mzhong@whut.edu.cn (M. Zhong), lvnengchao@163.com (N. Lyu).

http://dx.doi.org/10.1016/j.knosys.2015.07.004
0950-7051/

Thus it is computationally intractable in nowadays practice, particularly in the context of big data. Unlike subset methods, feature ranking methods individually take features as the evaluation units and rank them according to their discrimination power [21,22]. These methods usually employ heuristic search strategies such as forward search, backward search, and sequential floating search.

However, whatever feature ranking or feature subsets selection methods, there are two problems possibly leading to wrong rankings or lower capacity for classification. One is that neglecting feature interaction or dependence may lead to redundancy, as some feature selection methods like MIM [23] take the assumption of independence of features. For real-world datasets, particularly those high-dimensional ones, such strong assumption may produce results far from optimal. The other problem is that group capacity of features is usually ignored, since many methods only measure the relationship between two features [11,24,22]. For example, a feature that has low individual classification capacity but is highly dependent on other features may be overlooked and even misidentified as a redundant one by only measuring its pairwise relationship with other features. However, since it is highly dependent on other features, it is also possible that it contributes largely to the discrimination power of the subset consisting of such features. Thus, it should be evaluated as a salient feature and then selected. Since the dependence among features is related to both redundancy and complementariness, it is imperative to develop more precise correlation analysis in order to distinguish them effectively. To this end, we propose a novel feature selection algorithm which tries to modify the redundancy analysis applied in prior methods by introducing a modification item and a dynamic coefficient to effectively adjust redundancy-complementariness identification. The main contributions that distinguish our work form extant studies are listed as follows:

- Complementary correlation of features is explicitly separated from redundancy.
- Redundancy-complementariness dispersion is taken into account to adjust the measurement of pairwise intercorrelation of features.

The remainder of the paper is organized as follows: Section 2 reviews related work. Section 3 presents the Information theoretic metrics and evaluation criteria. A new feature selection method is included in Section 4. In Section 5, experimental study is conducted and the results are discussed. Finally, Section 6 concludes this study and proposes possible further work.

2. Related work

In recent decades, many kinds of feature selection methods have been studied. In general, there are two aims in these feature selection methods. One is to search the most class-relevant features, the other is to remove redundancy. Most feature selection algorithms can effectively find relevant features [25]. A well-known example is Relief, which is developed by Kira and Rendell [21]. The main idea of Relief is to rank features in terms of the weight corresponding to their ability to both discriminate instances with different class labels and cluster those with same class labels based on the distance between instances. However, Relief method may be ineffective since similar weights of two or more features cannot be removed by this method. In other words, this implies that redundant features cannot be identified. A typical and widely used extension of Relief is ReliefF [26], which is competent to the noisy and incomplete datasets. However, it is still unable to remove redundant features. Redundant features are considered to have negative effects on the accuracy and speed of classification methods, hence many feature selection methods are proposed to address this problem by statistic-based merics [22,27,17]. For example, Correlation based Feature Selection (CFS) algorithm proposed by Hall [27] adopts *cor* value to simultaneously measure a feature subset's correlation to the class and inter-correlation among features in it. CFS selects the subset which obtains the maximum *cor* value. However CFS does not designate specific search approaches, thus how to select feature subsets still remains to be a problem.

Minimum Redundancy and Maximum Relevance (mRMR) criterion and its variants [11,24,22] apply information theoretic metrics to separately measure class-relevance and pairwise correlation between features. A comprehensive score consisting of the two indices is applied to evaluate and select features. Fast Correlation Based Feature selection algorithm (FCBF) proposed by Yu and Liu [17] is another typical method that separately handles relevance and redundancy. FCBF utilizes Symmetrical Uncertainty (*SU*) as the metric to represent class-relevance and pairwise correlation. If the class-relevance of a feature is lower than that of another and the correlation between them, it would be identified as a redundant features and thus to be removed. Recently, an extension of FCBF, namely fast clustering-based feature selection algorithm (FAST), is proposed [28]. In this algorithm, features are firstly divided into clusters. Then for each cluster, an approximate Markov blanket based elimination strategy is applied to finally determine the selected feature subset. All of the above mentioned methods take pairwise correlation as the redundancy index and identify features with high such index to be redundant, while ignoring (1) complementary correlation between features (which we will discuss detailed in Section 3.2) and (2) correlation among more than two features, which still remain to be problems that impair the performance of feature selection.

Much effort has been made to tackle the former problem mentioned above [18,29–32,13–15,33]. Flueret [18] and Wang et al. [29] propose Conditional Mutual Information Maximization (CMIM) criterion for feature selection. CMIM harnesses Conditional Mutual Information (CMI) to measure the intensity of relevance and redundancy since CMI can implicitly identify complementary correlation between features, i.e. a large value of $CMI(F;C|\widetilde{F})$ implies (1) F is relevant to class C, and (2) F is highly complementary with $\widetilde{F}$, many information theoretic feature selection methods apply it to build up their evaluation criteria [34,31,30,35]. Algorithm based on Cumulate Conditional mutual information Minimization (CCM) criterion [13] is one of the typical algorithms that apply CMI to directly evaluate and select features. It generates candidate feature subset during the incremental step and eliminates redundancy during the shrinking step. Algorithms based on class-separability strategy extend the traditional usage of CMI in feature selection by measuring conditional mutual information between a feature and each class label [14]. Recentely, a feature selection framework basd on Data Anvelopment Analysis (DEA) is proposed [15]. Algorithm with this framework may apply MI and CMI as the evaluation indices to establish the feature evaluation system. Meanwhile, there are also several methods explicitly identifying redundancy and complementary correlation without CMI. Algorithms based on Joint Mutual Information (JMI) [32,36] take into account mutual information between a group of features and class. A typical algorithm taking JMI as metric can be found in [36], which applies JMI to measure mutual information between k features and class. Since the feature relevant to class and the one complementary to salient features will obtain high JMI values, they both will be identified as salient ones and thus is more possible to be selected. Although the above mentioned methods try to recognize complementariness from the pairwise correlation of features, measuring pairwise correlation is actually an approximation to

measuring the correlation among more than two features [37]. Under this circumstance, features that are strongly complementary to the certain selected feature(s) but not significantly correlated with the feature group are possible to be selected using such approximation, which will in turn intervene the later selection process.

3. Information theoretic metrics and evaluation criteria

3.1. Entropy, mutual information, and conditional mutual information

Entropy, mutual information, and conditional mutual information are the most frequently used metrics in feature selection method [38]. In this section, some essential information theoretic metrics used in our method will be described. The entropy, a fundamental unit of information, is used to quantify the uncertainty preset in the distribution of X, which is formed as [38]

$$H(X) = -\sum_{x\in X} p(x)\log p(x),$$

where $x \in X$ denotes the possible value assignments of X, $p(x)$ is the distribution of x (for convenience, we hereafter use the notation log to denote the base 2 logarithm instead of $\log_2$). According to the probability theory, one can use conditional entropy to quantify the uncertainty one variable conditioned on another one. The conditional entropy of X given Y is defined as [38]

$$H(X|Y) = -\sum_{y\in Y}\sum_{x\in X} p(xy)\log p(x|y),$$

Mutual Information (MI) between two random variables X and Y can be described as follows [38]

$$I(X;Y) = \sum_{x\in X}\sum_{y\in Y} p(xy)\log\frac{p(xy)}{p(x)p(y)},$$

where $x \in X$ and $y \in Y$ are the possible value assignments of X and Y, respectively. MI can be considered as the amount of information shared by two variables. In feature selection field, it is one of the most widely used metrics for measuring the correlation intensity of two features. Note that the MI is a symmetric merci, i.e. $I(X;Y) = I(Y;X)$. $I(X;Y) = 0$ implies that X and Y are statistically independent. Conditional mutual information (CMI), which is an extension of MI for measuring the conditional dependence between two random variables given the third, is defined as [38]

$$I(X;Y|Z) = \sum_{z\in Z} p(z)\sum_{x\in X}\sum_{y\in Y} p(xy|z)\log\frac{p(xy|z)}{p(x|z)p(y|z)}.$$

$I(X;Y|Z)$ can be interpreted as the information shared between X and Y given the value of a third variable (Z). MI and CMI can also be expressed with entropies as follows:

$$I(X;Y) = H(X) - H(X|Y)$$

and

$$I(X;Y|Z) = H(X|Z) - H(X|Y,Z).$$

3.2. Relevance, redundancy, and complementariness analysis

The motivation of using MI to solve feature selection problem is that a larger MI between the feature and the class should imply a potentially greater discrimination ability when using the feature. In addition, a commonly cited justification for using MI in feature selection is that MI can be used to write both an upper and lower bound on the Bayes error rate [39]. It can simply be applied as the criterion of a filter taking the form of

$$J(F) = I(F;C), \tag{1}$$

where $J(\cdot)$ denotes the evaluation criterion, F denotes a candidate feature and C denotes the class. Intuitively, the top m candidates which maximize $J(\cdot)$ could be selected, where m is a predefined number or decided by some stop criterion. In fact, this criterion takes the assumption that each feature is independent to all other features, which makes the criterion very efficient. However, such an assumption is so strong in practice that almost all the features may be mutually dependent to others, which makes the criterion shown in Eq. (1) be far from optimal. In general, it is widely recognized that a salient set of features should not only be individually relevant to class, but also should not be redundant to other features in the set. In order to identify redundancy, mRMR and its variants are proposed which can be generally formed as

$$J(F) = D(F) - R(F) \tag{2}$$

where $D(F)$ represents relevance between F and class C, $R(F)$ describes redundancy between F and the selected features in the subset $\mathbf{S}$. Usually, like in mRMR [11], $D(F)$ and $R(F)$ take the forms of MI. This criterion can efficiently find the features with high class-relevance and low dependence with respect to each other in $\mathbf{S}$. However, term "redundancy" not only implies that features are highly dependent to each other, but also indicates which one would be substitutable, i.e. their discrimination power would be significantly impaired when some other feature(s) are(is) given. From this viewpoint, only considering dependence between features is not enough to effectively identify redundancy. In other words, a feature which is dependent on another may not definitely imply to be redundant. Instead, the two features may complementary to each other, i.e. they would have stronger discriminatory power as a group (but may weak as individuals), particularly in microarray data analysis [40,41]. To this end, a complementary modification item is introduced as

$$J(F) = D(F) - (R(F) - M(F)) \tag{3}$$

where $M(F)$ is an item to identify complementary correlation between F and selected features in $\mathbf{S}$. In the context of MI, if $R(C)$ takes the form of $\sum_{F_s\in\mathbf{S}} I(F;F_s)$ (as in mRMR), $M(F)$ could thus be denoted as $\sum_{F_s\in\mathbf{S}} I(F;F_s|C)$, which represents the information shared between F and F_s given class C. In order to illustrate this, we first show the relationship between $R(F)$ and $M(F)$ as follows

$$\begin{aligned}
R(F) - M(F) &= I(F;F_s) - I(F;F_s|C)\\
&= \sum_{f_s\in F_s}\sum_{f\in F} p(ff_s)\log\frac{p(ff_s)}{p(f)p(f_s)}\\
&\quad - \sum_{c\in C} p(c)\sum_{f_s\in F_s}\sum_{f\in F} p(ff_s|c)\log\frac{p(ff_s|c)}{p(f|c)p(f_s|c)}\\
&= \sum_{c\in C}\sum_{f\in F}\sum_{f_s\in F_s} p(ff_sc)\log\left(\frac{p(ff_s)}{p(f)p(f_s)}\cdot\frac{p(f|c)p(f_s|c)}{p(ff_s|c)}\right)\\
&= \sum_{c\in C}\sum_{f\in F}\sum_{f_s\in F_s} p(ff_sc)\log\frac{p(ff_s)p(fc)p(f_sc)}{p(f)p(f_s)p(c)p(ff_sc)}\\
&= \sum_{c\in C}\sum_{f\in F}\sum_{f_s\in F_s} p(ff_sc)\log\left(\frac{p(fc)}{p(f)p(c)}\cdot\frac{p(ff_s)p(f_sc)}{p(ff_sc)p(f_s)}\right)\\
&= \sum_{f\in F}\sum_{c\in C} p(fc)\log\frac{p(fc)}{p(f)p(c)}\\
&\quad - \sum_{f_s\in F_s}\sum_{f\in F}\sum_{c\in C} p(ff_sc)\log\frac{p(fc|f_s)}{p(f|f_s)p(c|f_s)}\\
&= I(F;C) - I(F;C|F_s).
\end{aligned} \tag{4}$$

We now explain $R(F) - M(F)$ using Eq. (4), since the relationship between $I(F;C)$ and $I(F;C|F_s)$ is straightforward: If $I(F;C)$ is much

great than $I(F;C|F_s)$, the relevance between F and class C would become significantly weak after given the information of F_s. In other words, F is redundant to F_s. Conversely, if $I(F;C)$ is much small than $I(F;C|F_s)$, the relevance between F and class C would become significantly strong after given the information of F_s, i.e. F is complementary to F_s. Thus, $R(F)-M(F)$ could be applied to simultaneously measure redundancy and complementary correlation: When $R(F)-M(F)>0$, it captures the magnitude of redundancy between F and F_s; when $R(F)-M(F)<0$, it captures the magnitude of complementary correlation between F and F_s. In the context of MI, the following expression could be applied to be the evaluation criterion according to Eq. (4)

$$J(\cdot)=I(F;C)-Pair\_Cor(F;\mathbf{S}) \tag{5}$$

where $Pair\_Cor(F;\mathbf{S})$ takes the form of

$$Pair\_Cor(F;\mathbf{S})=\sum_{F_s\in\mathbf{S}}(I(F;F_s)-I(F;F_s|C)). \tag{6}$$

For the sake of convenience for the discussion in the following sections, we denote $cor(F;F_s)=I(F;F_s)-I(F;F_s|C)$ and thus Eq. (6) can be rewritten as

$$Pair\_Cor(F;\mathbf{S})=\sum_{F_s\in\mathbf{S}}cor(F;F_s). \tag{7}$$

It is noted that although Eq. (7) can measure both redundancy and complementary correlation, it is still a pairwise-based criterion since it only catches the relationship between two features. Criteria that only concern pairwise correlation among features is also called first-order approximation in literature [37]. We will further discuss the limitation of Eq. (7) in detail in the next section.

4. Feature selection with redundancy-complementariness dispersion

4.1. Interference effect of false positives

First-order approximation is a prevailing strategy that seems to bring the best trade-off between executional efficiency and the selected features' quality. Yet ignoring the group effect of features is still known to be suboptimal although taking the pairwise relevance effect into account. As mentioned before, feature selection methods that only handle individual relevance take the assumption of mutual independence among features. Similarly, first-order approximation in redundancy analysis only concentrates on individual redundancy. In other words, it takes the assumption that all the selected features are mutually independent. Since the first-order approximation only identify pairwise correlation, it is not able to take high inter-feature correlation into account, thus may misidentify and select actually-redundant features (i.e. False Positives, which is denoted as FPs hereafter in the paper; Similarly, we use the term True Positives (TPs) to denote the selected actually-salient features hereafter in the paper), which will in turn intervene the later selection process.

More specifically, only focusing on pairwise correlation may give chance to FPs to intervene the evaluation of candidates. Suppose the selected feature subset already contains FPs, the pairwise correlation between the candidate and each FP is an interference that prompts the candidate to be given unduly high status if such correlation is influential to the value of the evaluation criterion $J(\cdot)$. recall that the correlation between candidate and each selected features is denoted as $cor(F;F_s)$ where $F_s\in\mathbf{S}$ ($\mathbf{S}$ is the selected feature subset) and thus $Pair\_Cor(F;\mathbf{S})=\sum_{F_s\in\mathbf{S}}cor(F;F_s)$, the interference effect of FPs can be illustrated in two possible scenarios shown in Fig. 1 (a) and (b), where node in yellow, nodes in red, and nodes in green denote the candidate, FPs, and TPs, respectively. Distance between yellow node and any other node is in proportion to the strength of their pairwise correlation, e.g. a short distance corresponds to the complementary correlation, while long corresponds to the redundant correlation.

Scenario 1: FPs are close to the candidate. As shown in Fig. 1 (a), most of TPs are distant to the candidate, which implies that the candidate is more likely to be redundant rather than complementary to FPs (which corresponds to positive *cor* value in terms of Eq. (4)) and thus it is possibly a redundant feature. However, as FPs are very close to the candidate, they are more likely to be complementary and the corresponding *cor* value tend to be negative. Under this circumstance, the complementary correlation between candidate and FPs impairs the reliability of the estimation of $Pair\_Cor(F;\mathbf{S})$ and thus makes the candidate to be overestimated.

Scenario 2: FPs are distant to the candidate. Fig. 1(b) shows that most of TPs are close to the candidate. This implies that the candidate is more complementary to TPs and thus more likely to be a salient feature that should be selected. However, it is redundant to the distant FPs and the corresponding *cor* value tend to be positive, thus also impairs the reliability of the estimation of $Pair\_Cor(F;\mathbf{S})$ and makes the candidate to be underestimated.

Actually, the interference effect of FPs revealed in the above scenarios can be depicted by the dissimilarity of the selected features. That is, the magnitude of the interference effect of FPs depends on the extent of the dispersion of the correlation between candidate and the selected features. When a certain value of $Pair\_Cor(F;\mathbf{S})$ is given, the correlation between F and FPs in $\mathbf{S}$ more likely to be complementary corresponding to larger negative *cor* values would lead to the correlation between F and TPs in $\mathbf{S}$ more likely to be redundant corresponding to larger positive *cor* values, and vice versa. We call such dissimilarity as redundancy-complementariness dispersion. As a heuristic, we apply standard deviation of *cor* to capture such dispersion in order to possibly identify the interference effect of FPs, for standard deviation is always the best index for risk estimation and instability identification. The standard deviation of $cor(F;F_s)$ given the selected feature subset $\mathbf{S}$ takes the form of

$$\sigma(F;\mathbf{S})=\left(\frac{\sum_{F_s\in\mathbf{S}}(cor(F;F_s)-\mu(F;\mathbf{S}))^2}{|\mathbf{S}|}\right)^{\frac{1}{2}}, \tag{8}$$

where $\mu(F;\mathbf{S})$ is the mean value of $cor(F;F_s)$ calculated as

$$\mu(F;\mathbf{S})=\frac{Pair\_Cor(F;\mathbf{S})}{|\mathbf{S}|}. \tag{9}$$

Thus, the smaller the value of $\sigma(F;\mathbf{S})$, the less influential the interference effect of FPs. We try to find salient candidates not only with more complementariness and less redundancy, but also less redundancy-complementariness dispersion, i.e. a small value of $\sigma(F;\mathbf{S})$, to heuristically avoid the interference effect of FPs. To this end, we use $\sigma(F;\mathbf{S})$ to adjust the value of *Pair_Cor*. Recall that *Pair_Cor* simultaneously measures two types of correlation, i.e. redundancy (where the value of *Pair_Cor* is positive) and complementariness (where the value of *Pair_Cor* is negative). Taking this into account, we use the following criterion

$$J_{RCD}=D(F;C)-\phi(F;\mathbf{S})\cdot Pair\_Cor(F;\mathbf{S}) \tag{10}$$

where

$$\phi(F;\mathbf{S})=\begin{cases}1+\sigma(F;\mathbf{S}) & Pair\_Cor(F;\mathbf{S})\geqslant 0\\ 1-\sigma(F;\mathbf{S}) & Pair\_Cor(F;\mathbf{S})<0\end{cases} \tag{11}$$

to evaluate and select features among candidates. Note that $\phi(F;\mathbf{S})$ is defined piecewise for different types of correlation. Also, we use $1+\sigma(F;\mathbf{S})$ or $1-\sigma(F;\mathbf{S})$ rather than $\sigma(F;\mathbf{S})$ or $-\sigma(F;\mathbf{S})$ as the coefficient of $Pair\_Cor(F;\mathbf{S})$ in order to reduce the estimation bias of $\sigma(F;\mathbf{S})$ particularly when there are only a few features selected in $\mathbf{S}$.

4.2. Proposed method

Based on the above analysis, we propose our feature selection framework shown in Fig. 2. Different from traditional feature selection frameworks, proposed one extends traditional redundancy analysis to redundancy-complementariness analysis, and conducts dispersion analysis before feature evaluation. It not only considers class-relevance and pairwise inter-correlation of features, but also takes into account the effect of redundancy-complementariness dispersion. Similar to most of the feature selection methods, proposed method also applies the sequential forward search strategy to select features. That is, only one candidate features would be selected at each iteration.

We show the pseudo code of proposed algorithm in Algorithm 1.

Algorithm 1. RCDFS: **R**edundancy-**C**omplementariness **D**ispersion-based **F**eature **S**election

```
   Input: D /*dataset*/, F /*feature set*/, C /*class*/, δ
   /*expected # features to be selected*/
   Output: S /*selected feature subset*/
1  Initialize S ← ∅, k ← 1
2  repeat
3    foreach F_s ∈ F do
4      Relevance ← I(F; C)
5      Pair_Cor ← 0
6      foreach F_s ∈ S do
7        cor ← I(F; F_s) − I(F; F_s|C)
8        Pair_Cor ← Pair_Cor + cor
9      end
10     Calculate σ(F; S) according to Eq. (8)
11     if Pair_Cor ⩾ 0 then
12       φ ← 1 + σ(F; S)
13     else
14       φ ← 1 − σ(F; S)
15     end
16     J(F) ← Relevance − φ · Pair_Cor
17   end
18   S ← S ∪ {F̃} satisfying F̃ = arg max_{F∈F} J(F)
19   F ← F − {F̃}
20   k ← k + 1
21 until k ⩾ δ;
22 return S
```

4.3. Complexity analysis and a fast improvement of Algorithm 1

Algorithm 1 contains a 'repeat' loop and two 'for' loops and a calculation process of $\sigma(F;\mathbf{S})$ (line 11 in Algorithm 1) which takes at least $|\mathbf{S}|$ loops for the calculation. Thus, the worst iteration complexity of Algorithm 1 is $O(\delta\cdot|\mathbf{F}|^2)$, where δ is the predefined number of selected features. Taking into account the cost of (conditional) mutual information estimation ($O(|\mathbf{D}|+r)$) [13], the worst computational complexity of Algorithm 1 is $O(\delta\cdot(|\mathbf{D}|+r)\cdot|\mathbf{F}|^2)$, where $|\mathbf{D}|$ denotes the number of samples in the dataset and $r=\max_{F\in\mathbf{F}}|F|$ ($|F|$ is the number of values of F). Since there is only one candidate to be selected at the end of each iteration when traversing $F_s\in\mathbf{S}$, we only need to get the additional information of the newly-added feature rather than traversing $\mathbf{S}$ again. As for the calculation of $\sigma(F;\mathbf{S})$, we could apply an alternative formulation of variance, i.e. $Var(X)=E(X^2)-E^2(X)$ to make use of 'for' loops in Algorithm 1. That is, to get $\sigma(F;\mathbf{S})$, we have

$$\sigma(F;\mathbf{S})=\left(\frac{P-Q^2/|\mathbf{S}|}{|\mathbf{S}|}\right)^{\frac{1}{2}},$$

where we use P to record the summation of cor^2 and Q to record the summation of cor. Taking the above into account, we show the fast improvement of Algorithm 1 as Algorithm 2.

Algorithm 2. A fast implementation of RCDFS

```
   Input: D /*dataset*/, F /*feature set*/, C /*class*/, δ
   /*expected # features to be selected*/
   Output: S /*selected feature subset*/
1  Initialize S ← ∅, F_new ← ∅, Δ(F) ← 0 for
   ∀F ∈ F, Pair_Cor(F) ← 0 for ∀F ∈ F, k ← 0
2  foreach F ∈ F do
3    Relevance(F) ← I(F; C)
4  end
5  F_new ← F̃ satisfying F̃ = arg max_{F∈F} Relevance(F)
6  S ← S ∪ {F_new}
7  F ← F − {F_new}
8  k ← k + 1
9  repeat
10   foreach F ∈ F do
11     Relevance ← I(F; C)
12     cor ← I(F; F_new) − I(F; F_new|C)
13     Δ(F) ← Δ(F) + cor^2
14     Pair_Cor(F) ← Pair_Cor(F) + cor
15     σ(F; S) ← ( (Δ(F) − Pair_Cor(F)^2/|S|) / |S| )^{1/2}
16     if Pair_Cor(F) ⩾ 0 then
17       φ ← 1 + σ(F; S)
18     else
19       φ ← 1 − σ(F; S)
20     end
21     J(F) ← Relevance(F) − φ · Pair_Cor(F)
22   end
23   F_new ← F̃ satisfying F̃ = arg max_{F∈F} J(F)
24   S ← S ∪ {F_new}
25   F ← F − {F_new}
26   k ← k + 1
27 until k ⩾ δ
28 return S
```

By utilizing the additional information gained at the latest iteration, the worst iteration complexity of Algorithm 2 is reduced to $O(\delta\cdot|\mathbf{F}|)$ (accordingly, the worst computational complexity is reduced to $(O)(\delta\cdot(|\mathbf{D}|+r)\cdot|\mathbf{F}|)$), which is in the same level with that of mRMR and CMIM. Thus, we implement proposed method according to Algorithm 2 in the experiments to verify the performance of RCDFS.

5. Experiment study

In order to evaluate the performance and effectiveness of proposed method, the most representative and well-performed feature selection methods (CMIM [18], mRMR [11], FCBF [17], MIM [23], ReliefF [26], DEAFS [15], and *k*ASSI [36]) are used to compare with proposed algorithm. Brief reviews on above seven selected feature selection algorithms are described as follows:

- CMIM (Conditional Mutual Information Maximization) [18]: This well-known algorithm makes use of CMI to simultaneously measure class-relevance and inter-correlation of features, applying the following function

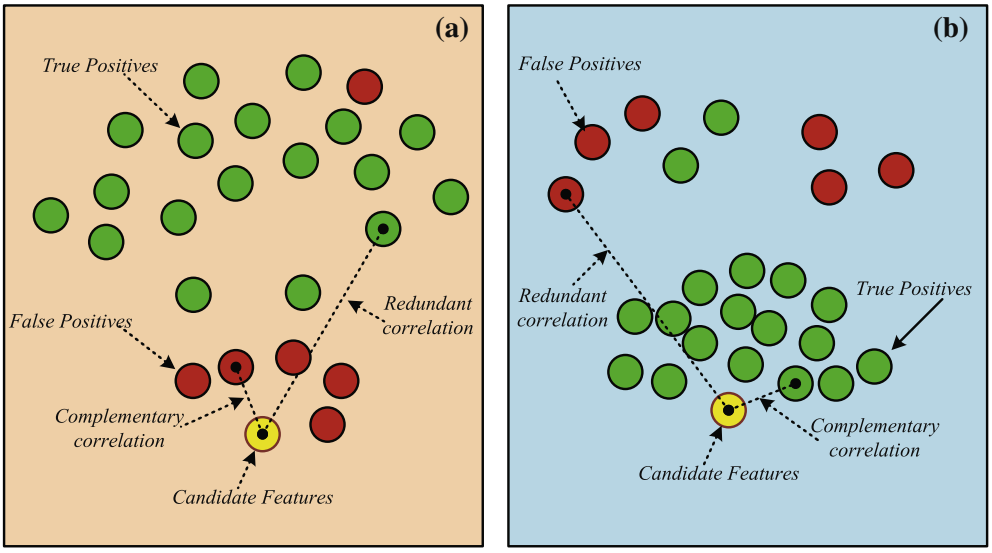

Fig. 1. Toy examples of interference effect of FPs.

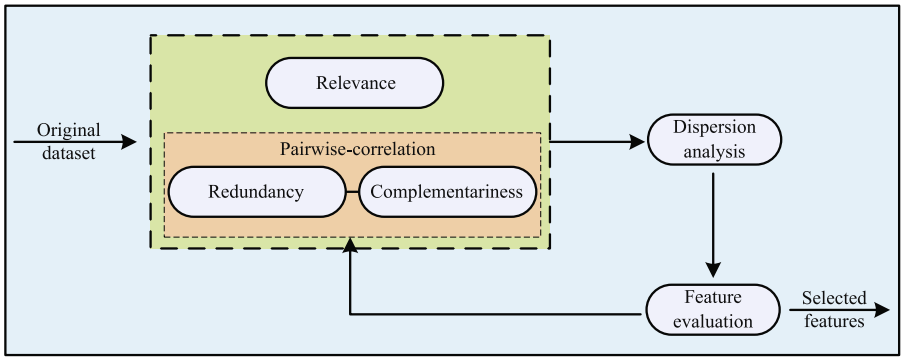

Fig. 2. A new framework for feature selection.

$$J(F) = \min_{\widetilde{F} \in \mathbf{S}} I(F; C|\widetilde{F})$$

as the evaluation criterion, taking the heuristic that $\widetilde{F}$ satisfying $\min_{\widetilde{F} \in \mathbf{S}} I(F; C|\widetilde{F})$ could best represent the conditioning set **S**.

- mRMR (minimum Redundancy and Maximum Relevance) [11]: It is a very famous feature selection algorithm that uses MI to measure class-relevance and pairwise dependence. It selects feature satisfying

$$J(F) = I(F; C) - \frac{1}{|\mathbf{S}|} \sum_{F_s \in \mathbf{S}} I(F; F_s)$$

in a greedy manner, where $I(F; C)$ measures the class-relevance of F and $\frac{1}{|\mathbf{S}|} \sum_{F_s \in \mathbf{S}} I(F; F_s)$ measures the average pairwise dependence between F and $F_s \in \mathbf{S}$. Note that we have already introduced it in Section 3.2.

- FCBF (Fast Correlation-Based Feature selection) [17]: In this algorithm, Symmetrical Uncertainty (SU) is used as the evaluation merci. It first ranks features in descending order. Then it eliminates redundant features in terms of an approximate Markov blanket criterion: If $SU(F_1; C) > SU(F_2; C)$ and $SU(F_1; C) > SU(F_1; F_2)$, F_2 is thus identified as a redundant feature of F_1 and thus would be eliminated. For this method, we set the predefined threshold $\gamma = 0$ as suggested by [17].
- MIM (Mutual Information Maximization) [23]: It is the most basic feature ranking algorithms based on mutual information that only concerns the class-relevance of features. We have also introduced it in Section 3.2. It applies

$$J(F) = I(F; C)$$

as the criterion to select the top m features with the highest value of $I(F; C)$. It is one of the most typical benchmark algorithms in the field of feature selection.

- ReliefF [26]: It is a well-known distance-based feature ranking method that searches nearest neighbors of samples for each class label and then weights features in terms of how well they differentiate samples for different class labels. As for the parameter settings, we use 5 neighbors and 30 instances throughout the experiments as suggested by [26].
- DEAFS [15]: It is a feature ranking algorithm based on Data Envelopment Analysis (DEA). It evaluates features according to more than one criterion (i.e. relevance, redundancy, conditional dependence, etc.), and applies a DEA-based evaluation approach to get the efficiency score of a feature by considering its class-relevance and conditional dependence to every other feature in the feature space. However, DEAFS will always generate a $|\mathbf{F}|^2$ matrix (where **F** denotes the feature size) to record the conditional dependence between every two features, it is thus not executable on large-scale datasets.

- *k*ASSI [36]: It is a feature ranking algorithm based on joint mutual information criterion. It aims at maximizing joint mutual information among features [36]:

$$F_{\mathbf{S}}^{k\text{ASSI}} = \arg\max_{F_{\mathbf{S}}\subset\mathbf{F}}\left\{\sum_{\mathbf{V}\subset\mathbf{S}:|\mathbf{V}|=k} I(F_{\mathbf{V}};Y)\right\}$$

  In our experiments, sequential forward search strategy is applied for *k*ASSI and k (i.e. the number of features in $|\mathbf{V}|$) is set as 2 to make *k*ASSI only consider pairwise correlation among features.

Weka (Waikato environment for knowledge analysis) [42] is chosen as the classification platform. Since FCBF, MIM, and ReliefF have already been integrated in Weka, we directly use them to generate datasets with their selected features before classification. CMIM, mRMR, DEAFS, *k*ASSI and the proposed method are implemented in Java and with Weka interfaces. All experiments are conducted on a 2.60 GHz CPU, 8 GB RAM personal computer with Windows 7.

5.1. Datasets

In order to validate the performance of the proposed method, ten frequently used datasets are applied in our experiments, where six of them (mushroom, kr-vs-kp, sonar, multiple features ka, DNA, and isolet5) are well known UCI datasets and the rest (Colon Tumor, BCR_ABL, Prostate Cancer, and Breast Cancer) are gene microarray datasets with high dimensionality (i.e. containing more than 2000 features). General information of these datasets are summarized in Table 1. For the continuous and mixed datasets, a supervised discretization method called Minimum Descriptive Length (MDL) method [43] is employed to discrete continuous features before feature selection and classification.

5.2. Classifiers and experimental settings

5.2.1. Classifiers

In our experiments, four famous and most frequently used classifiers – Naïve Bayesian Classifier (NBC) [44], Support Vector Machine (SVM) [45], k-Nearest Neighbor (*k*NN) [46] and C4.5 decision tree [47] are adopted to generate classification error rates on the datasets with selected features preprocessed by different feature selection methods. We set $k = 1$ for *k*NN and employ Gaussian RBF kernels for SVM.

5.2.2. Experimental settings

First, we show the classification results of the four classifiers on $1,\ldots,m$ selected features for each feature selection method, where m in our experiments is set to be $\min\{50,|\mathbf{F}|\}$. 10-fold cross validation is applied in this part. Note that the nature of the learning process of each classifier is different. Since we are interested in checking the quality of the selected features, independently from the type of classification rule applied, the average result of the four classifiers is thus reported.

In addition, we compare the best classification results for the eight feature selection methods among their selected features. That is, we check the average classification results for each feature selection method on the datasets with selected features ranging from 1 to $\min\{50,|\mathbf{F}|\}$, and report the best one. In order to achieve stable results, a $(M = 10)\times(N = 10)$-fold cross-validation is applied, i.e. 10-fold validation will be conduct ten times for each classifier on each dataset. Thus, a total of one hundred result samples (i.e. average results from four classifiers) can be collected where each sample is an average classification result of the four classifiers. Finally, the average of one hundred samples is reported in our paper. Wilcoxon rank-sum test is applied to determine the statistical significance of the difference of the results (where the significant level is set to be 0.05).

To further test the stability of the performance on different datasets, average classification results of different datasets, in ranges from 1 to 5, from 1 to 10, from 1 to 15, from 1 to 20, from 1 to 25, from 1 to 30, from 1 to 35, from 1 to 40,from 1 to 45, and from 1 to 50 selected features, are reported and analyzed respectively for each classifier and feature selection method. Friedman test is applied to analyze the statistical significance of the results. These ten average classification results have been considered to be the approximate transitory period to reach a stable performance for the datasets used.

At last, execution time of selected feature selection methods and proposed RCDFS are reported. For CMIM, mRMR, *k*ASSI, and RCDFS, we compare their runtime with respect to different sizes of selected features. Meanwhile, we also report their runtime on the feature size that corresponding to the best average classification results. For MIM, ReliefF, FCBF, and DEAFS, we only report the runtime of them on certain feature sizes corresponding to their best average classification results since they are all one-time ranking methods and their time complexity is irrelevant to the number of selected features.

Table 1
Description of datasets.

#	Name	# samples	# features	Type	# classes
1	Mushroom	8124	22	nominal	2
2	kr-vs-kp	3196	36	nominal	2
3	Sonar	208	60	nominal	2
4	Multiple features kahunen	2000	64	numeric	10
5	DNA	3186	180	nominal	3
6	isolet5	1559	617	mixed	26
7	Colon Tumor	62	2000	numeric	2
8	BCR_ABL	215	12,559	numeric	2
9	Prostate Cancer	102	12,601	numeric	2
10	Breast Cancer	78	24,482	numeric	2

5.3. Experimental results and discussion

Figs. 3–12 show the 10-fold cross-validation average classification error rates of the different types of classifiers (NBC, SVM, kNN, and C4.5) on the ten datasets to illustrate the effectiveness of proposed method RCDFS, where the consecutive numbers of selected features are described by X axis, and the average classification error rate is represented by Y axis. According to the results shown in Figs. 3–12, the superiority of RCDFS can be verified in the majority of cases. Particularly on seven datasets namely mushroom (Fig. 3), kr-vs-kp (Fig. 4), sonar (Fig. 5), DNA (Fig. 7), Colon Tumor (Fig. 9), BCR_ABL (Fig. 10), and Breast Cancer (Fig. 12), RCDFS significantly outperforms CMIM, mRMR, FCBF, MIM, ReliefF, DEAFS, and *k*ASSI (DEAFS cannot execute on gene microarray datasets because of its capability limitation mentioned previously). More precisely, RCDFS usually perform better at the beginning of feature selection process on several datasets such as sonar (Fig. 5) and Colon Tumor (Fig. 9). This is probably because the redundancy and complementariness are both considered by RCDFS, rather than only measuring pairwise redundancy like mRMR or ignoring redundancy like MIM, FCBF and ReliefF. For other datasets, e.g. BCR_ABL dataset, the classification error rate corresponding to RCDFS is almost the same as (or a little higher than) those to CMIM, mRMR, and *k*ASSI on the first seven features, whereas after the eighth feature being selected, RCDFS performs better (i.e. the

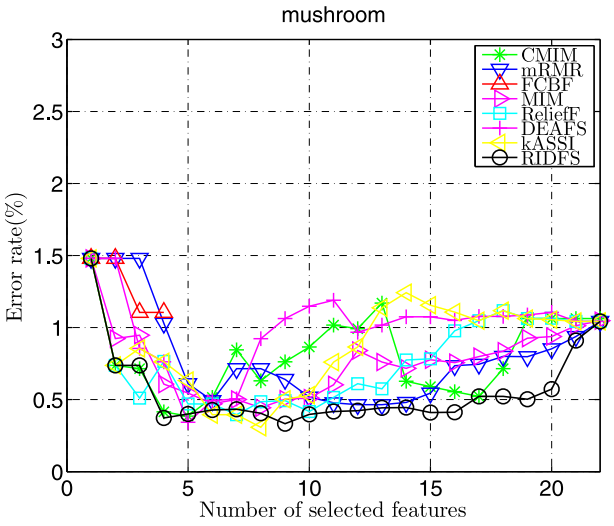

Fig. 3. Accuracy comparison with different number of selected features on mushroom.

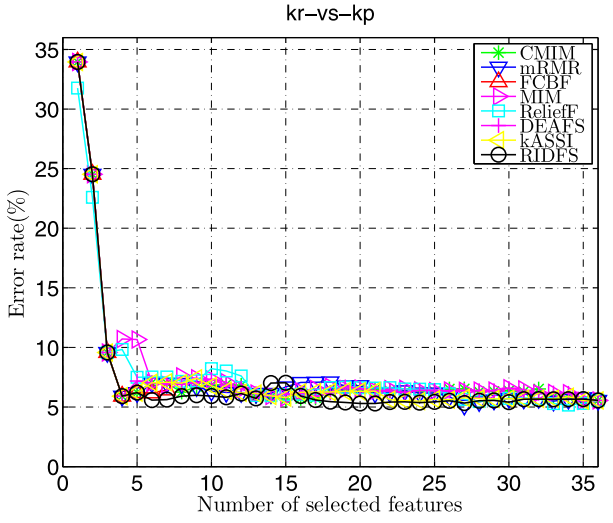

Fig. 4. Accuracy comparison with different number of selected features on kr-vs-kp.

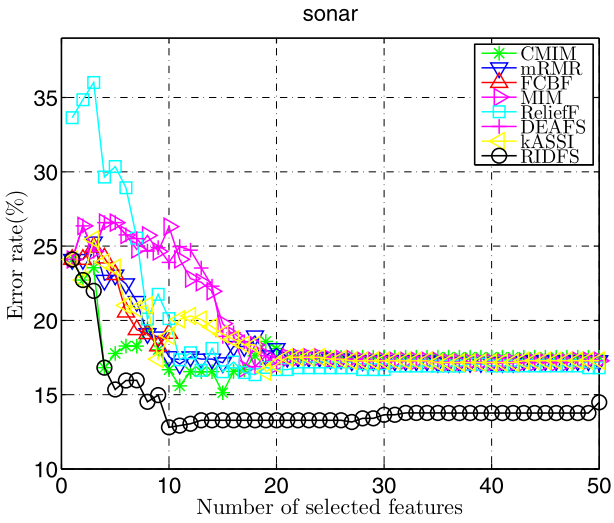

Fig. 5. Accuracy comparison with different number of selected features on sonar.

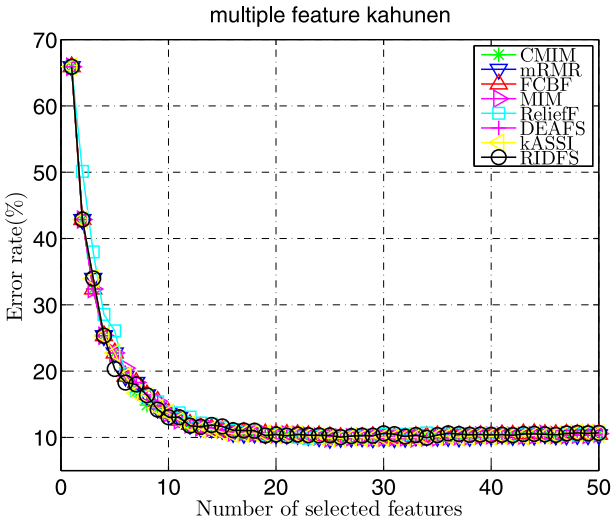

Fig. 6. Accuracy comparison with different number of selected features on multiple features Kahunen.

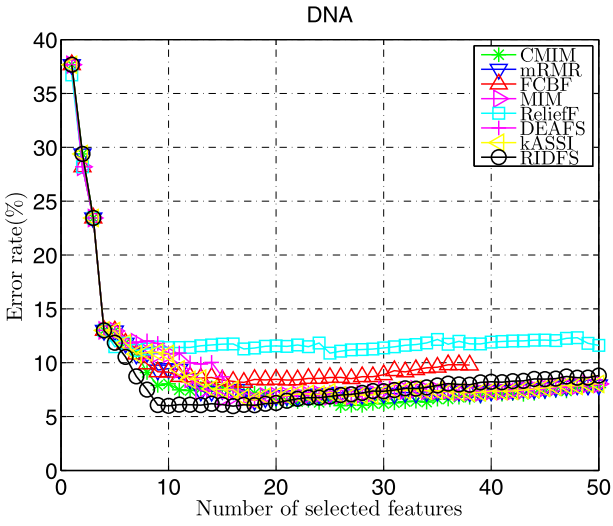

Fig. 7. Accuracy comparison with different number of selected features on DNA.

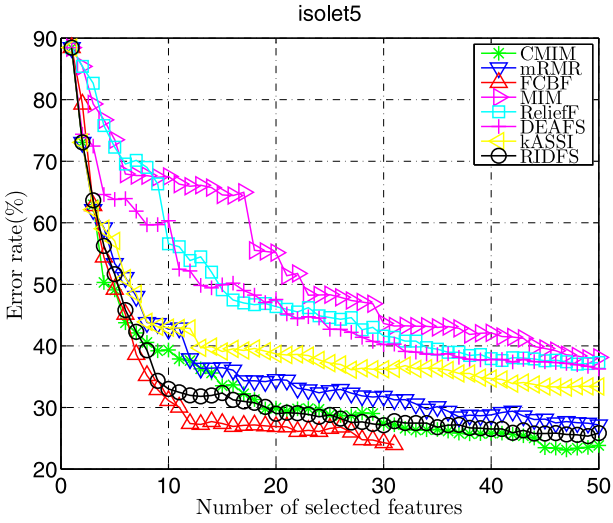

Fig. 8. Accuracy comparison with different number of selected features on isolet5.

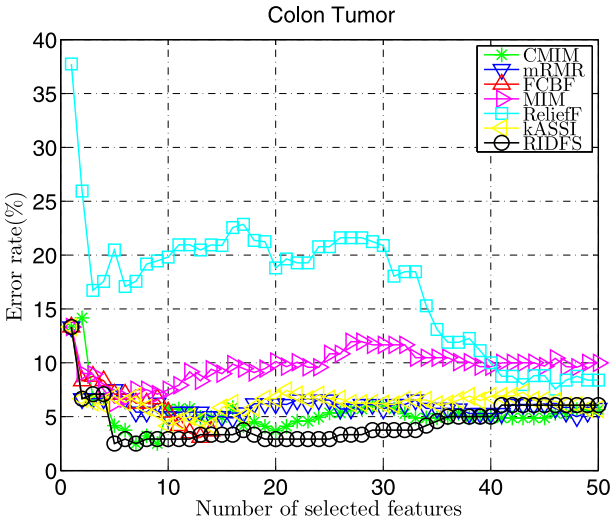

Fig. 9. Accuracy comparison with different number of selected features on Colon Tumor.

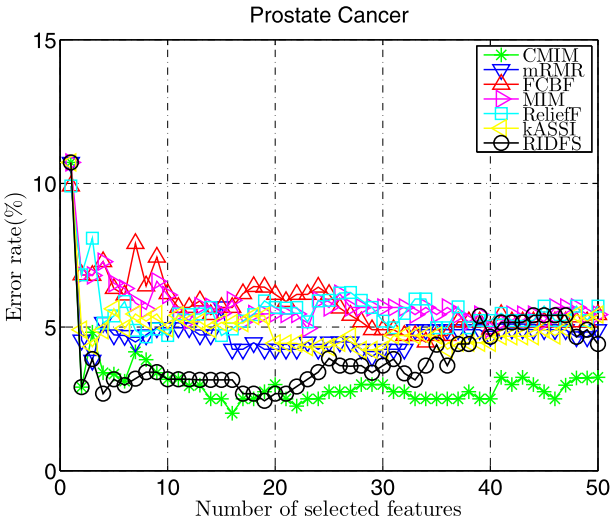

Fig. 11. Accuracy comparison with different number of selected features on Prostate Cancer.

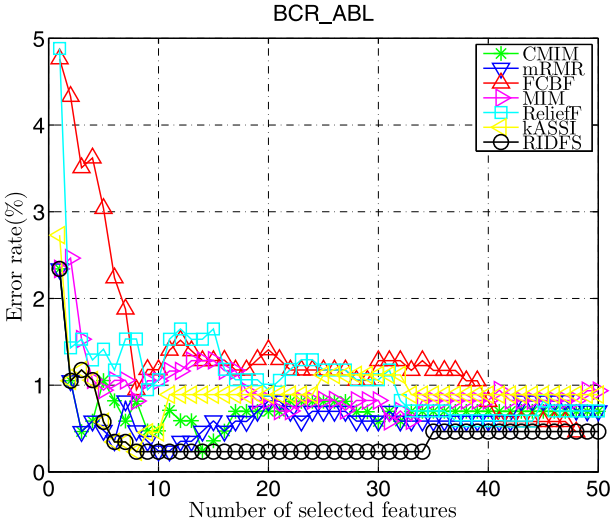

Fig. 10. Accuracy comparison with different number of selected features on BCR_ABL.

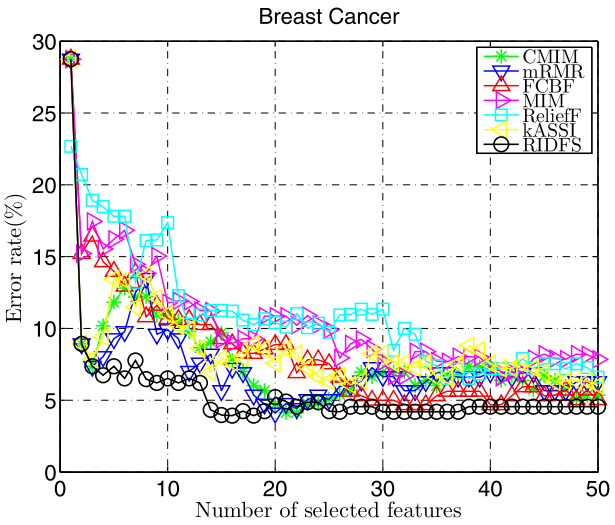

Fig. 12. Accuracy comparison with different number of selected features on Breast Cancer.

classification error rate is lower) than other methods and will be never exceeded. This is possibly due to the fact that the dispersion of redundancy-complementariness correlation becomes influential to feature evaluation process after several features being selected, i.e. the interference effect of FPs in the selected subset impairs the evaluation ability of the selected compared algorithms. On the whole, it can be seen that RCDFSs select less features corresponding to the lowest error rates than other methods (e.g. it corresponds to the best classification results only selecting five and sixteen features on Colon Tumor and DNA, respectively). It is also found that the performance of RCDFS is not always outstanding and sometimes inferior to CMIM (e.g. results shown in Fig. 11). This may also lie in the dispersion of the redundancy-complementariness correlation since there may exist alternative causes leading to high dispersion which cannot be captured by the variance between TPs and FPs.

Table 2 records the number of features selected by each feature selection algorithm corresponding to the best average classification results. We observe from the table that the average number of selected features of RCDFS (18.7) is smallest compared to other algorithms used in our experiment. This indicates the advantage of RCDFS: The best classification result can be obtained with a significantly small set of features.

Table 3 show the average classification error rates of NBC, SVM, kNN, and C4.5 on ten datasets over $(M = 10) \times (N = 10)$-fold cross validation, respectively. For each dataset, Wilcoxon test is conducted to evaluate the statistical significance of the difference between the two groups of result samples, i.e. groups of the result samples that corresponds to RCDFS and any other feature selection method. In Table 3, "Err" column records the average classification error rate of $(M = 10) \times (N = 10)$-fold cross-validation. "p-val" column records the p-value associated with Wilcoxon test, where p-value less than 0.05 indicates the statistical significance of the difference between the two average values. Notation "•"/"◦" are used to show that the classification error rate corresponding to the current feature selection method is significantly lower/higher

Table 2
Number of selected features corresponding to best average classification results.

Datesets	# Features							
	RCDFS	CMIM	mRMR	FCBF	MIM	ReliefF	DEAFS	*k*ASSI
Mushroom	9	5	12	3	8	7	5	8
kr-vs-kp	21	35	27	4	35	34	35	24
Sonar	10	15	11	9	17	18	18	19
Multiple feature kahunen	34	31	25	32	25	23	31	25
DNA	16	26	18	17	18	25	18	18
isolet5	49	47	50	31	49	48	50	47
Colon Tumor	5	7	15	13	5	46	N/A	10
BCR_ABL	8	14	10	48	31	42	N/A	8
Prostate Cancer	19	16	3	34	40	8	N/A	28
Breast Cancer	16	21	20	32	50	36	N/A	16
Avg.	18.7	21.7	19.1	22.3	27.8	28.7	26.2	20.3

Table 3
Best average classification error rates corresponding to the eight feature selection algorithms with NBC, SVM, kNN and C4.5, and the results of Wilocxon test.

# Dat.	RCDFS	CMIM		mRMR		FCBF		MIM		ReliefF		DEAFS		*k*ASSI	
	Err	Err	*p*-val	Err	*p*-val	Err	*p*-val	Err	*p*-val	Err	*p*-val	Err	*p*-val	Err	*p*-val
1	**0.32**	0.37	0.207	0.47	0.000[a]	23.26	0.000[a]	20.57	0.000[a]	0.39	0.000[a]	0.34	0.701	0.33	0.363
2	5.32	5.61	0.015[a]	**5.14**	0.231	5.91	0.000[a]	5.61	0.015[a]	5.21	0.449	5.61	0.015[a]	5.55	0.086
3	**14.05**	16.05	0.025[a]	17.44	0.000[a]	18.63	0.000[a]	17.38	0.001[a]	16.56	0.006[a]	17.05	0.003[a]	16.77	0.003[a]
4	10.05	9.89	0.317	**9.81**	0.143	10.08	0.874	**9.81**	0.143	9.83	0.144	10.00	0.935[a]	**9.81**	0.143
5	**5.98**	5.99	0.959	6.46	0.011[a]	8.16	0.000[a]	6.46	0.011[a]	10.79	0.000[a]	6.46	0.011[a]	6.46	0.011[a]
6	25.13	**23.19**	0.000[b]	27.06	0.000[a]	23.62	0.000[b]	37.89	0.000[a]	37.11	0.000[a]	36.03	0.000[a]	33.11	0.000[a]
7	3.32	2.83	0.105	4.02	0.715	**2.37**	0.103	7.26	0.001[a]	8.35	0.000[a]	N/A		19.99	0.000[a]
8	0.29	0.44	0.317	0.44	0.329	0.58	0.000[a]	0.92	0.000[a]	0.64	0.000[a]	N/A		**0.14**	0.001[b]
9	5.13	5.58	0.070	5.71	0.113	5.96	0.960	6.71	0.758	8.54	0.053	N/A		**4.72**	0.37
10	**3.66**	5.65	0.000[a]	4.52	0.106	4.86	0.021[a]	7.58	0.000[a]	8.63	0.000[a]	N/A		8.57	0.000[a]
Avg.	**7.35**	7.56		8.11		10.34		12.02		10.60		10.78		10.54	

[a] Statistical degradation at significant level of 0.05.
[b] Statistical improvement at significant level of 0.05.

Table 4
Average classification error rates of all datasets with NBC and results of Friedman test.

NB	$k = 5$	$k = 10$	$k = 15$	$k = 20$	$k = 25$	$k = 30$	$k = 35$	$k = 40$	$k = 45$	$k = 50$
RCDFS	**18.91**	**13.92**	**11.66**	**10.45**	**10.56**	**9.93**	**9.48**	**8.87**	**8.59**	**8.34**
CMIM	19.34	14.70	12.68	11.43	11.48	10.88	10.40	9.82	9.51	9.25
mRMR	19.42	15.32	13.23	12.07	12.30	11.73	11.32	10.88	10.57	10.29
FCBF	20.26	15.51	13.22	12.04	12.30	11.60	11.26	10.72	10.59	10.48
MIM	27.01	21.11	18.64	17.11	17.38	16.50	15.78	15.52	15.00	14.56
ReliefF	29.28	23.04	19.99	18.35	18.95	18.02	17.29	17.08	16.43	15.87
*k*ASSI	19.45	15.41	13.74	12.77	13.10	12.49	12.11	11.74	11.44	11.21
p-val	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	S	S	S	S	S	S	S	S	S	S

than that to proposed method (corresponding to "RCDFS" column) under the test. Bold value in each row indicates that it is the best result among eight feature selection methods. The average error rate of ten datasets is given in the last row.

As can be seen from Table 3, average classification error rates on ten datasets show that RCDFS outperforms other methods on mushroom, sonar, DNA, BCR_ABL, Prostate Cancer, and Breast Cancer. According to the values of "Err" given in the last row, the best one is obtained by our method (7.35) and the worst is by MIM (12.02). Also, the average error rate of CMIM (7.56) is better than other algorithms (mRMR (8.11), FCBF (10.34), ReliefF (10.60), DEAFS (10.78), and *k*ASSI (10.54)).

For further analysis, the diagram (Fig. 13) is applied to visualize the statistical significance of RCDFS comparing with the selected methods under four classifiers on ten datasets. The blue box in Fig. 13 describes that the classification error rate of RCDFS is significantly better than compared algorithm in current dataset. The yellow box represents that there is no significant difference between the results of RCDFS and compared algorithm. The red box implies that the classification error rate of RCDFS is significantly worse than compared algorithm. The white box with a crossing mark in it implies that Wilcoxon test is not conducted owing to capability limitation of the compared algorithm. Results shown in Fig. 13 indicate that RCDFS achieves better performance in most of datasets compared with selected feature selection algorithms.

Tables 4–7 show the statistical significance of average error using Friedman test under different classifiers on ten datasets. Results in column $k = 5$, 10, 15, 20, 25, 30, 35, 40, 45, and 50 represents the average classification error in ranges from 1 to 5, from 1 to 10, from 1 to 15, from 1 to 20, from 1 to 25, from 1 to 30, from 1 to 35, from 1 to 40,from 1 to 45, and from 1 to 50 features, respectively. Note that FCBF may select less features than other methods, e.g. it only selects 4 features on mushroom

Table 5
Average classification error rates of all datasets with SVM classifiers and results of Friedman test.

SVM	$k = 5$	$k = 10$	$k = 15$	$k = 20$	$k = 25$	$k = 30$	$k = 35$	$k = 40$	$k = 45$	$k = 50$
RCDFS	**18.83**	**13.65**	**11.41**	**10.15**	**10.34**	**9.72**	**9.32**	**9.34**	9.24	9.20
CMIM	18.87	14.24	12.14	10.75	10.85	10.12	9.57	9.37	**8.96**	**8.64**
mRMR	19.64	15.20	12.83	11.38	11.57	10.82	10.25	10.08	9.71	9.41
FCBF	20.81	15.80	13.49	12.20	12.46	11.70	11.29	10.78	10.59	10.43
MIM	27.28	21.15	18.53	16.60	16.79	15.61	14.67	14.75	14.06	13.49
ReliefF	29.10	22.17	18.68	16.52	16.86	15.82	14.92	15.09	14.40	13.87
*k*ASSI	19.29	15.34	13.14	11.84	12.20	11.51	10.99	10.99	10.60	10.28
p-val	0.003	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.001
	S	S	S	S	S	S	S	S	S	S

Table 6
Average classification error rates of all datasets with *k*NN classifiers and results of Friedman test.

*k*NN	$k = 5$	$k = 10$	$k = 15$	$k = 20$	$k = 25$	$k = 30$	$k = 35$	$k = 40$	$k = 45$	$k = 50$
RCDFS	**18.72**	**14.27**	**12.38**	**11.22**	**11.62**	**11.11**	**10.77**	**11.19**	**11.01**	**10.89**
CMIM	19.10	14.90	13.04	12.02	12.59	12.21	11.87	12.37	12.19	12.03
mRMR	19.56	15.61	13.68	12.64	13.22	12.64	12.21	12.63	12.38	12.17
FCBF	20.54	16.45	14.52	13.51	14.22	13.70	13.42	13.27	13.07	12.92
MIM	26.94	21.28	18.88	17.40	18.14	17.36	16.74	17.44	16.97	16.57
ReliefF	28.98	22.81	19.87	18.05	18.72	17.91	17.29	18.02	17.49	17.04
*k*ASSI	19.37	15.47	13.75	12.79	13.52	12.98	12.65	13.25	13.06	12.89
p-val	0.006	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	S	S	S	S	S	S	S	S	S	S

Table 7
Average classification error rates of all datasets with C4.5 classifiers and results of Friedman test.

C4.5	$k = 5$	$k = 10$	$k = 15$	$k = 20$	$k = 25$	$k = 30$	$k = 35$	$k = 40$	$k = 45$	$k = 50$
RCDFS	**19.23**	**15.50**	**13.84**	**12.91**	**13.83**	**13.50**	**13.27**	**14.22**	**14.13**	**14.08**
CMIM	19.50	16.48	15.04	14.12	15.10	14.78	14.53	15.51	15.40	15.29
mRMR	20.26	17.72	16.19	15.33	16.36	15.97	15.68	16.83	16.71	16.57
FCBF	21.34	17.91	16.57	16.05	17.40	17.18	17.07	17.57	17.56	17.57
MIM	27.47	22.76	20.84	19.34	20.37	19.57	18.91	20.13	19.77	19.52
ReliefF	29.65	24.22	21.95	20.68	21.94	21.32	20.69	21.90	21.29	20.81
*k*ASSI	20.15	17.39	16.07	15.33	16.49	16.16	15.92	17.19	17.08	16.96
p-val	0.019	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	S	S	S	S	S	S	S	S	S	S

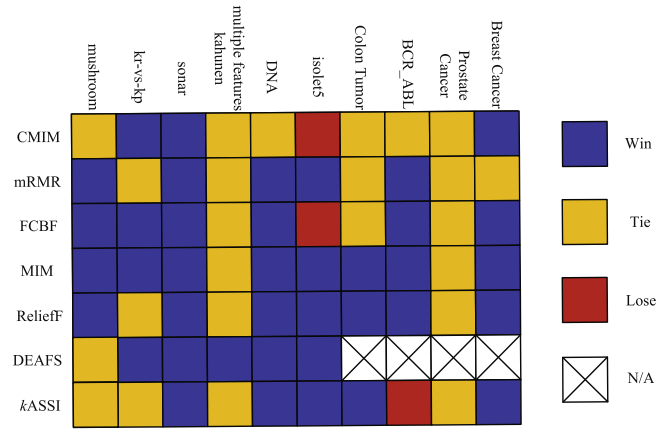

Fig. 13. Average classification error rate comparison between RCDFS and the selected methods on the selected ten datasets.

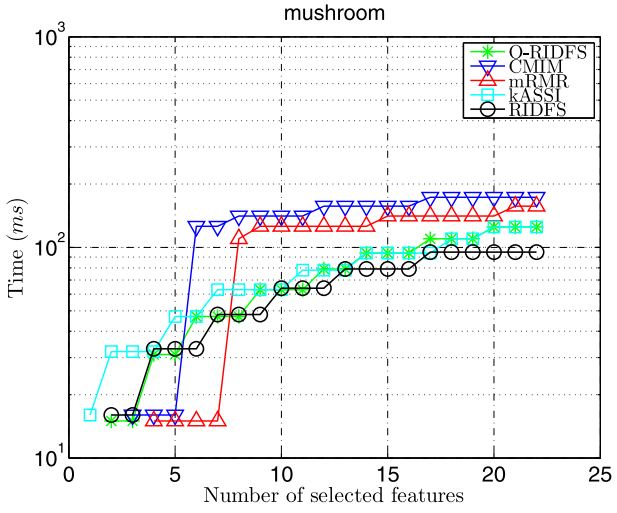

Fig. 14. Runtime comparison with different number of selected features on mushroom.

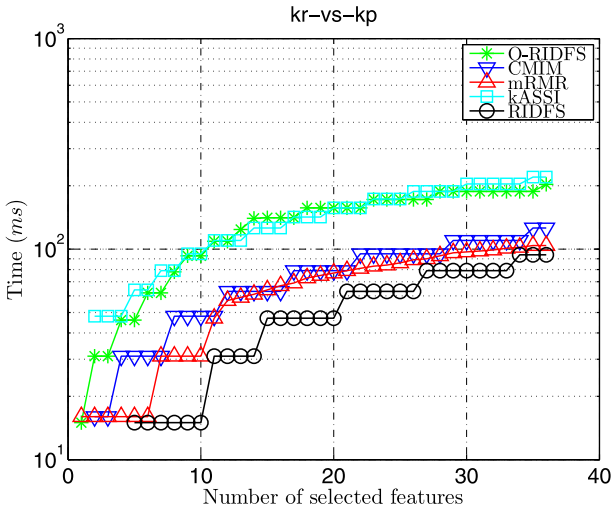

Fig. 15. Runtime comparison with different number of selected features on kr-vs-kp.

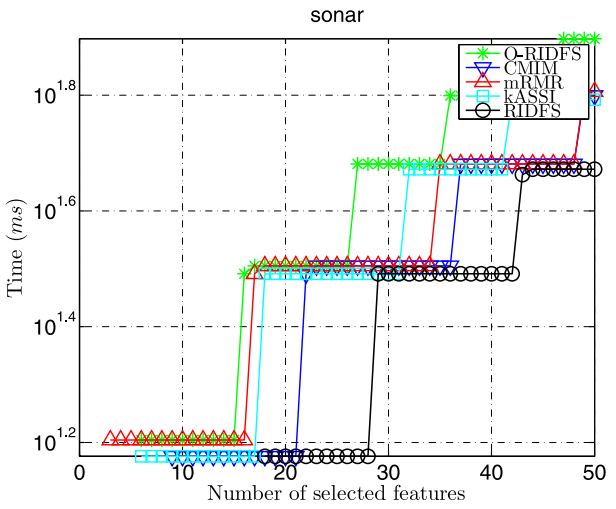

Fig. 16. Runtime comparison with different number of selected features on sonar.

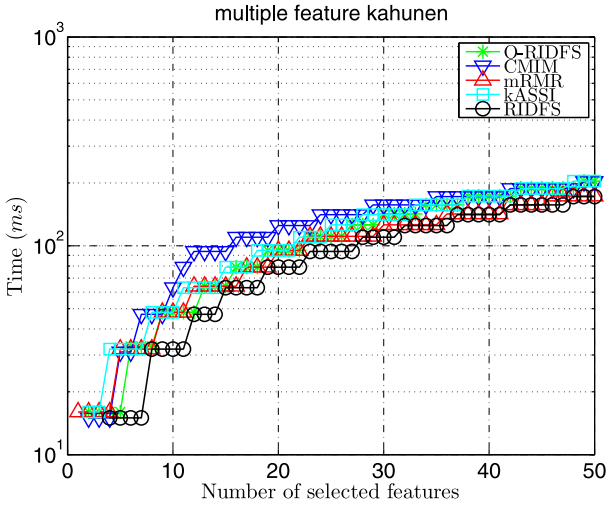

Fig. 17. Runtime comparison with different number of selected features on multiple features kahunen.

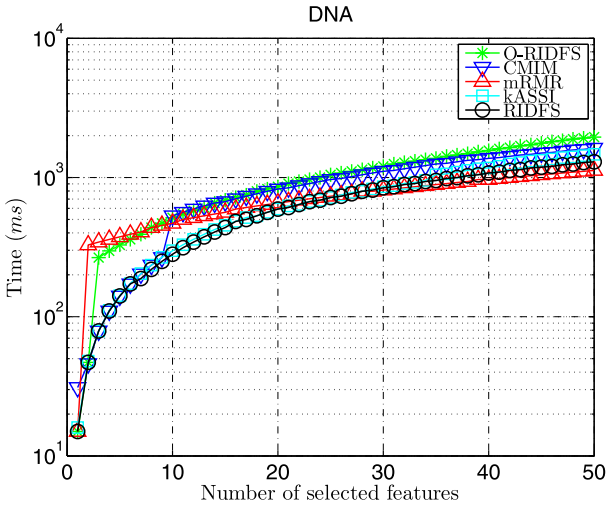

Fig. 18. Runtime comparison with different number of selected features on DNA.

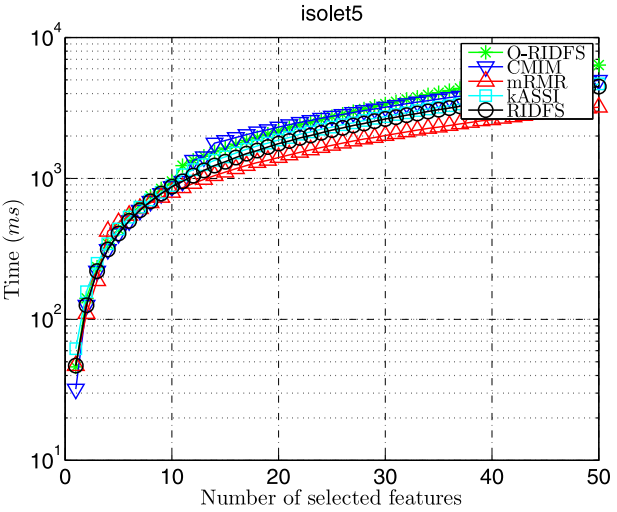

Fig. 19. Runtime comparison with different number of selected features on isolet5.

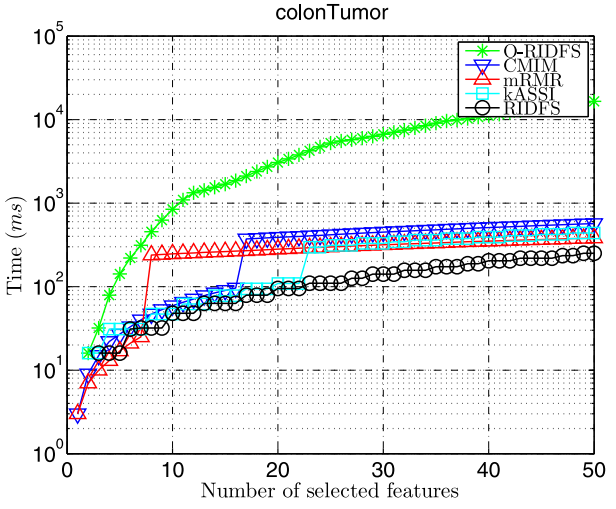

Fig. 20. Runtime comparison with different number of selected features on Colon Tumor.

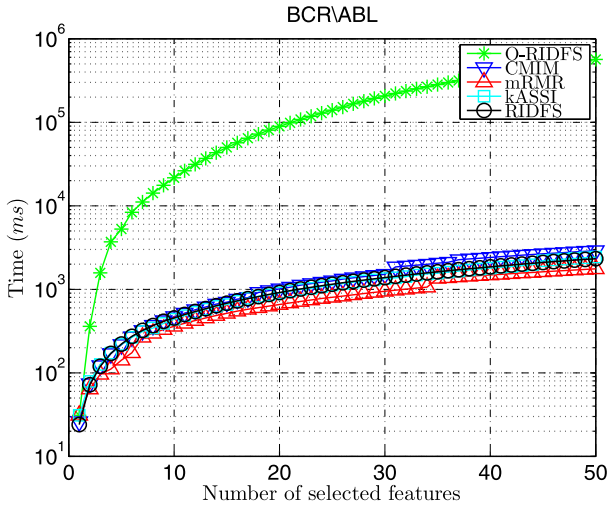

Fig. 21. Runtime comparison with different number of selected features on BCR_ABL.

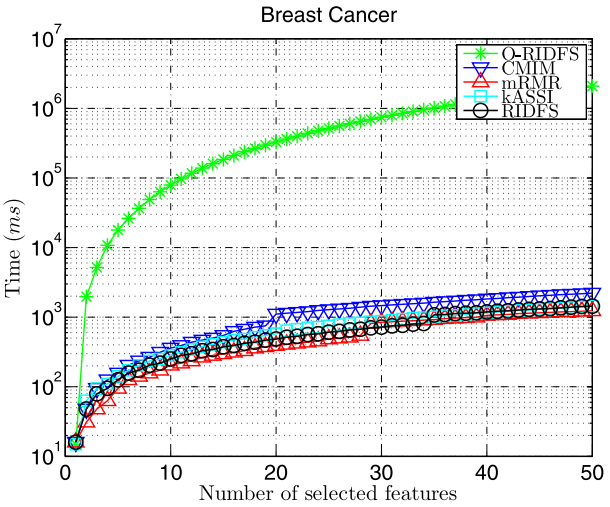

Fig. 23. Runtime comparison with different number of selected features on Breast Cancer.

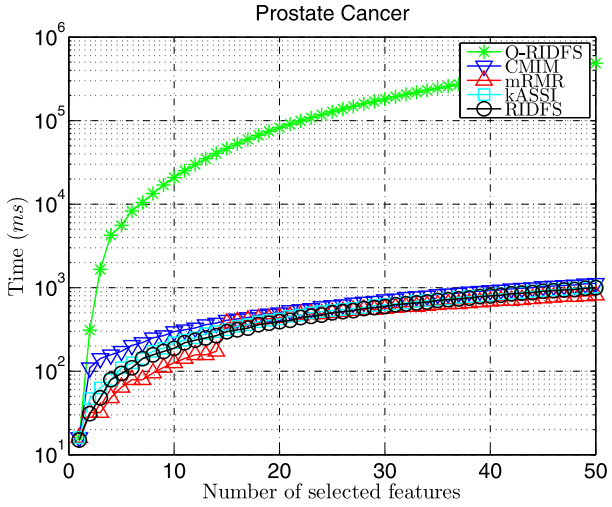

Fig. 22. Runtime comparison with different number of selected features on Prostate Cancer.

dataset, thus the average up to 4 features is described in row $k =$ 25, 30, 35, 40, 45, and 50. A very small p-val (i.e. p-val < 0.05) indicates the significant difference among the average values. In addition, we use S/N given in the last row of the tables to represent statistically significant/not significant difference among the average values under Friedman test with significant level 0.05. Bold value in each column shows the best classification result among seven feature selection methods (DEAFS is excluded since its capability limitation).

Table 4 shows that the average classification error rate of Naïve Bayesian Classifier (NBC) corresponding to RCDFS is lowest among all methods and p-val is smaller than 0.05. This indicates that the performance of RCDFS is best using NBC with the number of selected features in all ranges. Similar to NBC, the average error rate of SVM corresponding to RCDFS shown in Table 5 is also lowest with the number of selected features in most of the ranges. In addition, the CMIM is superior to other methods with SVM in the ranges of k = 1–45 and to 50. From Tables 6 and 7, the average error rates of kNN and C4.5 corresponding to RCDFS are both the lowest and the p-val is also smaller than 0.05, which verifies the effectiveness of RCDFS.

5.4. Runtime comparison and analysis

Figs. 14–23 show the runtime of four feature selection algorithms (RCDFS, CMIM, mRMR, kASSI) under different selected feature sizes. Recall that all features will be evaluated and ranked one time by MIM, ReliefF, FCBF, and DEAFS, thus we only report their runtime corresponding to the best average classification results in Table 8. To show the efficiency improvement of Algorithm 1, we implement RCDFS using both Algorithms 1 and 2, which are denoted as O-RCDFS (Original-RCDFS) and RCDFS, respectively. Results show that RCDFS, CMIM, mRMR, and kASSI execute in the same level of time and all significantly faster than O-RCDFS (particularly on gene microarray datasets), indicating the effectiveness of the fast implementation shown in Algorithm 2. More specifically, the runtime of RCDFS is faster than O-RCDFS, CMIM, mRMR, and kASSI on mushroom (Fig. 14), kr-vs-kp (Fig. 15), sonar (Fig. 16), multiple feature kahunen (Fig. 17), and Colon Tumor (Fig. 20). We notice that the runtime of RCDFS is slower than mRMR on DNA (Fig. 18), isolet5 (Fig. 19), BCR_ABL (Fig. 21), Prostate Cancer (Fig. 22), and Breast Cancer (Fig. 23), this is because mRMR never considers the complementary correlation among features and only two correlations (i.e. class-relevance and pairwise redundancy between features) are calculated.

Table 8 records the runtime of seven selected feature selection algorithms corresponding to their best average classification results shown in Table 3. Note that DEAFS is not executable on gene microarray datasets namely Colon Tumor, BCR_ABL, Prostate Cancer, and Breast Cancer owing to its capability limitation. Results show again that RCDFS, CMIM, mRMR, and kASSI are in the same level of runtime. We also notice that MIM, ReliefF, and FCBF run much faster than RCDFS, CMIM, mRMR, kASSI, and DEAFS (e.g. in some cases the running time of MIM and ReliefF is even lower than 10^{-2} ms) because they all contain constant iterations (one-time for MIM and ReliefF to calculate the evaluation scores of all features) whereas the number of iterations of RCDFS, CMIM, and mRMR is determined by the feature size.

Table 8
Runtime of eight feature selection algorithms corresponding to their best average classification results shown in Table 2.

Datesets	Execution time (ms)							
	RCDFS	CMIM	mRMR	FCBF	MIM	ReliefF	DEAFS	*k*ASSI
Mushroom	95	173	157	42	0.008	0.004	1036	125
kr-vs-kp	94	126	104	31	0.002	0.004	1058	220
Sonar	47	63	64	2	0.008	0.002	1242	62
Multiple feature kahunen	172	203	173	111	0.004	0.006	2057	204
DNA	1296	1624	1125	409	0.022	0.012	57,177	1469
isolet5	4482	4966	3202	770	0.074	0.038	4,431,154	4685
Colon Tumor	251	568	383	20	0.144	0.162	N/A	438
BCR_ABL	2325	2902	1758	205	1.058	1.022	N/A	2436
Prostate Cancer	999	1125	813	247	1.044	1.028	N/A	1015
Breast Cancer	1437	2217	1234	287	1.95	1.992	N/A	1437

6. Conclusions and future work

Relevance and redundancy are two important feature properties attracting much attention in the study of feature selection. Many algorithms eliminate redundancy by measuring pairwise inter-correlation between features, while they cannot identify the complementariness of features and the correlation among more than two features. Although the former problem can be effectively addressed by introducing a modification item, high inter-correlation of features still makes the result far from optimal. Specifically, pairwise approximation of high inter-correlation may misidentify and select FPs which will in turn impair the effectiveness of feature evaluation. In order to identify the interference effect of FPs, the redundancy-complementariness dispersion is taken into account in proposed method to adjust the measurement of pairwise inter-correlation of features. To illustrate the effectiveness of proposed method RCDFS, classification experiments are conducted with four frequently used classifiers on ten datasets.In the experiments, RCDFS is compared with seven representative feature selection methods namely CMIM, mRMR, FCBF, MIM, ReliefF, DEAFS and *k*ASSI. Classification results have been proven to perform satisfactorily of RCDFS. To verify the stability of RCDFS, Wilcoxon test as well as Friedman test are adopted to assess the statistical significance of the differences among the results of the feature selection method. According to the test results, RCDFS performs better than the selected methods in most of the cases.

Although the superiority of RCDFS has been verified in the experiments, there still remain challenges which are imperative to be solved in our future work. One is that how to properly set the weights of three objectives, i.e. coordinate relevance, redundancy-complementary, and dispersion of pairwise inter-correlation, is needed to be studied. Possible directions include multi-objective programming and multi-index evaluation techniques such as data envelopment analysis. Moreover, since there is no causal relationship between FPs and the dispersion of pairwise inter-correlation, only concerning such dispersion may not always be effective in feature evaluation. How to design more effective heuristics in the context of first-order approximation will be further studied. In addition, optimization techniques like sparse logistic regression [16] will be introduced to deal with high-dimensional data in our future work.

Acknowledgements

We thank the editors and two anonymous reviewers for their constructive comments and suggestions. This work is partially supported by the National Natural Science Foundation of China (51178364, 61104158, 51208401, and 61203236), National Key Technology Support Program (2014BAG01B0503), National Key Projects in the Science & Technology of China (2014BAG01B03), the Fundamental Research Funds for the Central Universities (2013-YB-011), the Doctorate Fellowship Foundation of Huazhong University of Science & Technology (D201177780), the Fundamental Research Funds for the Central Universities, HUST (CXY12Q044, CXY13Q035), the Graduates' Innovation Fund of Huazhong University of Science & Technology (HF-11-20-2013), and China Scholarship Council (201406950047 and 201406160046).

References

[1] I. Guyon, A. Elisseeff, An introduction to variable and features election, J. Mach. Learn. Res. 3 (2003) 1157–1182.
[2] S. Yang, L. Yuan, Y.-C. Lai, X. Shen, P. Wonka, J. Ye, Feature grouping and selection over an undirected graph, in: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, New York, NY, USA, 2012, pp. 922–930.
[3] J. Tang, H. Liu, An unsupervised feature selection framework for social media data, IEEE Trans. Knowl. Eng. 26 (12) (2014) 2914–2927.
[4] Z. Li, J. Liu, Y. Yang, X. Zhou, H. Lu, Clustering-guided sparse structural learning for unsupervised feature selection, IEEE Trans. Knowl. Eng. 26 (9) (2014) 2138–2150.
[5] S.H. Yang, B.G. Hu, Discriminative feature selection by nonparametric Bayes error minimization, IEEE Trans. Knowl. Eng. 24 (8) (2012) 1422–1434.
[6] F. Yang, K. Mao, G.K.K. Lee, W. Tang, Emphasizing minority class in LDA for feature subset selection on high-dimensional small-sized problems, IEEE Trans. Knowl. Eng. 27 (1) (2015) 88–101.
[7] K. Javed, H.A. Babri, M. Saeed, Feature selection based on class-dependent densities for high-dimensional binary data, IEEE Trans. Knowl. Eng. 24 (3) (2012) 465–477.
[8] Z. Zhao, L. Wang, H. Liu, J. Ye, On similarity preserving feature selection, IEEE Trans. Knowl. Eng. 25 (3) (2013) 619–632.
[9] T.S. Furey, N. Cristianini, N. Duffy, D.W. Bednarski, M. Schummer, D. Haussler, Support vector machine classification and validation of cancer tissue samples using microarray expression data, Bioinformatics 16 (10) (2000) 906–914.
[10] G. Qu, S. Hariri, M. Yousif, A new dependency and correlation analysis for features, IEEE Trans. Knowl. Data Eng. 17 (9) (2005) 1199–1207.
[11] H. Peng, F. Long, C. Ding, Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy, IEEE Trans. Pattern Anal. Mach. Intell. 27 (8) (2005) 1226–1238.
[12] J.J. Huang, Y.Z. Cai, X.M. Xu, A parameterless feature ranking algorithm based on MI, Neurocomputing 71 (2008) 1656–1668.
[13] Y. Zhang, Z. Zhang, Feature subset selection with cumulate conditional mutual information minimization, Expert Syst. Appl. 39 (2012) 6078–6088.
[14] Y. Zhang, S. Li, T. Wang, Z. Zhang, Divergence-based feature selection for separate classes, Neurocomputing 101 (2013) 32–42.
[15] Y. Zhang, A. Yang, C. Xiong, Z. Zhang, Feature selection using data envelopment analysis, Knowl.-Based Syst. 64 (2014) 70–80.
[16] M. Tan, I.W. Tsang, L. Wang, Minimax sparse logistic regression for very high-dimensional feature selection, IEEE Trans. Neur. Learn. Syst. 24 (10) (2013) 1609–1622.
[17] L. Yu, H. Liu, Efficient feature selection via analysis of relevance and redundancy, J. Mach. Learn. Res. 5 (2004) 1205–1224.
[18] F. Fleuret, Fast binary feature selection with conditional mutual information, J. Mach. Learn. Res. 5 (2004) 1531–1555.
[19] I. Tsamardinos, C. Aliferis, A. Statnikov, Local causal and markov blanket induction for causal discovery and feature selection for classification part i: algorithms and empirical evaluation, J. Mach. Learn. Res. 11 (2010) 171–234.
[20] A.A. Albrecht, Stochastic local search for the feature set problem, with applications to microarray data, Appl. Math. Comput. 183 (2) (2006) 1148–1164.
[21] K. Kira, L. Rendell, A practical approach to feature selection, in: Proceedings of the 9th International Workshop on Machine Learning, ML'92, Morgan Kaufmann, San Francisco, CA, USA, 1992, pp. 249–256.
[22] R. Battiti, Using mutual information for selecting features in supervised neural net learning, IEEE Trans. Neural Netw. 5 (4) (1994) 537–550.

[23] D.D. Lewis, Feature selection and feature extraction for text categorization, in: Proceedings of the Workshop on Speech and Natural Language, Association for Computational Linguistics Morristown, NJ, USA, 1992, pp. 212–217.
[24] C. Ding, H. Peng, Minimum redundancy feature selection from microarray gene expression data, in: Proceedings of the IEEE Computer Society Conference on Bioinformatics, CSB'03, IEEE Computer Society, Washington, DC, USA, 2003, pp. 523–528.
[25] G. Forman, An extensive empirical study of feature selection metrics for text classification, J. Mach. Learn. Res. 3 (2003) 1289–1305.
[26] M. Robnik-Sikonja, I. Kononenko, Theoretical and empirical analysis of relief and relieff, Mach. Learn. 53 (2003) 23–69.
[27] M.A. Hall, Correlation-based feature selection for discrete and numeric class machine learning, in: Proceedings of the 7th International Conference on Machine Learning, ICML'00, Morgan Kaufmann, Los Altos, CA, USA, 2000, pp. 359–366.
[28] Q. Song, J. Ni, G. Wang, A fast clustering-based feature subset selection algorithm for high-dimensional data, IEEE Trans. Knowl. Data Eng. 25 (1) (2013) 1–14.
[29] G. Wang, F.H. Lochovsky, Q. Yang, Feature selection with conditional mutual information maximin in text categorization, in: Proceedings of the 19th ACM International Conference on Information and Knowledge Management, CIKM'04, ACM Press, New York, NY, USA, 2004, pp. 342–349.
[30] I. Tsamardinos, C.F. Aliferis, A. Statnikov, Algorithms for large scale markov blanket discovery, in: Proceedings of the 16th International Florida Artificial Intelligence Research Society Conference, FLAIRS'03, AAAI Press, Menlo Park, CA, USA, 2003, pp. 376–381.
[31] A. Charnes, W.W. Cooper, E. Rhodes, Measuring the efficiency of decision making units, Euro. J. Operat. Res. 2 (1978) 429–444.
[32] H.H. Yang, J. Moody, Feature selection based on joint mutual information, in: Proceedings of International ICSC Symposium on Advances in Intelligent Data Analysis, 1999, pp. 22–25.
[33] N. Kwak, C.H. Choi, Input feature selection for classification problems, IEEE Trans. Neural Netw. 13 (1) (2002) 143–159.
[34] J.M. Sotoca, F. Pla, Supervised feature selection by clustering using conditional mutual information-based distances, Pattern Recog. 43 (6) (2010) 2068–2081.
[35] S. Yaramakala, D. Margaritis, Speculative markov blanket discovery for optimal feature selection, in: Proceedings of the 5th IEEE International Conference on Data Mining, ICDM'05, IEEE Computer Society Press, Washington, DC, USA, 2005, pp. 809–812.
[36] P.E. Meyer, C. Schretter, G. Bontempi, Information-theoretic feature selection in microarray data using variable complementarity, IEEE J. Select. Topics Signal Process. 2 (3) (2008) 261–274.
[37] G. Brown, A. Pocock, M.-J. Zhao, M. Luján, Conditional likelihood maximisation: a unifying framework for information theoretic feature selection, J. Mach. Learn. Res. 13 (2012) 27–66.
[38] T.M. Cover, J.A. Thomas, Elements of Information Theory, Wiley, NewYork, NY, USA, 1991.
[39] M. Hellman, J. Raviv, Probability of error, equivocation, and the Chernoff bound, IEEE Trans. Inf. Theory 16 (4) (1970) 368–372.
[40] A. Gyenesei, U. Wagner, S. Barkow-Oesterreicher, E. Stolte, R. Schlapbach, Mining co-regulated gene profiles for the detection of functional associations in gene expression data, Bioinformatics 23 (2007) 1927–1935.
[41] D. Ruano, G.R. Abecasis, B. Glaser, E.S. Lips, L.N. Cornelisse, A.P.H. de Jong, D.M. Evans, G.D. Smith, N.J. Timpson, A.B. Smit, P. Heutink, M. Verhage, D. Posthuma, Functional gene group analysis reveals a role of synaptic heterotrimeric g proteins in cognitive ability, Am. J. Human Genet. 86 (2010) 113–125.
[42] M. Shah, M. Marchand, J. Corbeil, Feature selection with conjunctions of decision stumps and learning from microarray data, IEEE Trans. Pattern Anal. Mach. Intell. 34 (1) (2011) 174–186.
[43] U.M. Fayyad, K.B. Irani, Multi-interval discretization of continuous valued attributes for classification learning, in: Proceedings of the 13th International Joint Conference on Artificial Intelligence, IJCAI'93, 1993, pp. 1022–1027.
[44] H.I. Witten, E. Frank, Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations, Morgan Kaufmann, San Francisco, CA, USA, 2000.
[45] N. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods, Cambridge University Press, Cambridge, UK, 2000.
[46] D. Aha, D. Kibler, Instance-based learning algorithms, Mach. Learn. 6 (1991) 37–66.
[47] R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufmann Publishers, San Mateo, CA, USA, 1993.