symmetry

MDPI

Article

Feature Selection with Conditional Mutual Information Considering Feature Interaction

Jun Liang [1,2,*], Liang Hou [2], Zhenhua Luan [1,2] and Weiping Huang [2]

[1] State Key Lab of Nuclear Power Safety Monitoring Technology and Equipment, Shenzhen 518124, China
[2] State Key Lab of Industrial Control Technology, College of Control Science and Engineering, Zhejiang University, Hangzhou 310027, China
* Correspondence: jliang@zju.edu.cn

Received: 30 May 2019; Accepted: 25 June 2019; Published: 2 July 2019

Abstract: Feature interaction is a newly proposed feature relevance relationship, but the unintentional removal of interactive features can result in poor classification performance for this relationship. However, traditional feature selection algorithms mainly focus on detecting relevant and redundant features while interactive features are usually ignored. To deal with this problem, feature relevance, feature redundancy and feature interaction are redefined based on information theory. Then a new feature selection algorithm named CMIFSI (Conditional Mutual Information based Feature Selection considering Interaction) is proposed in this paper, which makes use of conditional mutual information to estimate feature redundancy and interaction, respectively. To verify the effectiveness of our algorithm, empirical experiments are conducted to compare it with other several representative feature selection algorithms. The results on both synthetic and benchmark datasets indicate that our algorithm achieves better results than other methods in most cases. Further, it highlights the necessity of dealing with feature interaction.

Keywords: feature selection; conditional mutual information; feature interaction; classification; computer engineering

1. Introduction

In an era of growing data complexity and volume, high dimensional data brings a huge challenge for data processing, as it increases the computational complexity in computer engineering. Feature selection is a widely used technique to address this issue. Theoretically, the more features are used, the more information is provided, however this is not always true in practical experience. Excessive features not only bring high computation complexity, but also cause the learning algorithm to over-fit the training data. Since feature selection could provide many advantages, such as avoiding over-fitting, resisting noise, reducing computation complexity and increasing predictive accuracy, it has attracted increasing interest in the field of machine learning and a large amount of feature selection algorithms have been proposed during recent years.

Feature selection could be broadly categorized into three types, i.e., wrapper, filter, and embedded methods according to whether the selection algorithm is independent of the specified learning algorithm [1]. Wrapper methods use a predetermined classifier to evaluate the candidate feature subset. Therefore, they usually achieve a higher predictive accuracy than other methods, like some heuristic algorithms that excessively depend on hyper-parameters, with a heavy computational burden and a high risk of being overly specific to the classifier. One of the typical wrapper methods is shown in reference [2]. For the embedded methods, feature selection is integrated into the training process for a given learning algorithm. They are less computationally expensive, but need strict model structure assumptions. In contrast, filter methods are independent of learning algorithms because

they involve defining a heuristic evaluation criterion to provide a proxy measure of the classification accuracy. Compared with wrapper and embedded methods, due to the computational efficiency and generalization ability, filter methods are gaining more interest and many contributions have been made in feature selection since 2008 [3]. Filter methods could be further divided according to different kinds of evaluation criterions, such as distance, information, dependency and consistency [4]. Among these evaluation criterions, the information metric has gained more attention and is more comprehensively studied because of its ability to quantify the nonlinear relevance among features and classes.

Traditional feature selection algorithms mainly focus on the removing of irrelevant and redundant features. Irrelevant features provide no useful information and redundant features provide overlapped information about the selected features. However, feature interaction is usually ignored. Feature interaction was first proposed by Jakulin, et al. [5] and some recent research has pointed out its effect on classification. Interactive features could provide more information when combined together than the sum of information provided individually. Unintentional removal of interactive features would result in poor classification performance. An extreme example of feature interaction is the XOR problem. Suppose we defined label C based on two features f1, f2, C= f1$\oplus$f2, then each feature is independent of the label C and provides no information about the class individually. However, these two features completely determine the class together.

Wrapper methods could deal with feature interaction implicitly to some extent. However, the heavy computational burden makes wrapper methods intractable for large scale classification tasks. Some newly proposed filter methods have considered feature interaction [6–8]. However, it's still a challenge for most filter methods to handle interaction and more work is needed on an explicit treatment of this issue. These challenges include sensitivity to data noise and data transformation [9].

Many feature selection algorithms have been proposed and widely used. Genetic Algorithm (GA) is a heuristic algorithm with global optimization. However, "pre-mature" outcomes can occur with expected hyper-parameters. The Symmetric Uncertainty (SU) algorithm assumes that the evaluated feature is independent of other features and reflects only the single feature and category. The Relief algorithm takes samples randomly while the number of samples greatly affects the results. Correlation-based feature selection (CFS) is a filter method that selects features by measuring the correlation between features and categories and the redundancy between different features, but its result may not be the global optimum. The Minimum-Redundancy Maximum-Relevance (MRMR) method searches for the most closely related features with objective category, or a subset of features that are least redundant. It can meticulously characterize feature correlation and redundancy weights. Conditional Mutual Information Maximization (CMIM) uses conditional mutual information to measure distance, which makes a tradeoff between the predictive power of the candidate feature and its independence from previously selected features. However, it may be difficult to calculate the multidimensional probability density in high dimensional space. Those methods have achieved good performance in some cases. However, they ignored the significance of feature interaction. Feature interaction is very significant and can be used in many fields like object detection and recognition, and neurocomputing and so on. Reference [10] integrated feature interaction into their proposed linear regression model to capture the nonlinear property of data. Reference [11] proposed a method to remove relevant features by considering the feature interaction and reducing the weakly relevant features.

In this paper, a new feature selection method based on conditional mutual information (named CMIFSI) is proposed. Firstly, some basic information-theoretic concepts and related work are reviewed, then a new information metric is proposed to evaluate the redundancy and interaction of candidate features. With the aid of this metric, CMIFSI could restrain the redundant features and redress interactive ones in the feature ranking process. To verify its performance, CMIFSI is compared with several of the state-of-the-art feature selection methods mentioned above.

2. Basic Information-Theoretic Concepts

In this section, we give a brief introduction to information-theoretic concepts, followed by a summary of applications used for feature selection. Information theory was initially developed by Shannon to deal with communication problems, and entropy is the key measure. Because of its capability to quantify the uncertainty of random variables and the amount of information shared by different random variables, information theory has also been widely applied to feature selection [12].

Let X be a random variable with m discrete values and $p(x_i)$ represents the probability of x_i, x_i is the i-th value of X, then its uncertainty measured by entropy $H(X)$ is defined as

$$H(X) = -\sum_{i=1}^{m} p(x_i) \log p(x_i) \tag{1}$$

It's worth noting that entropy doesn't depend on actual values but just the probability distribution of discrete values. Then the joint entropy $H(X, Y)$ of X and Y, a random variable with n discrete values is defined as

$$H(X, Y) = -\sum_{i=1}^{m} \sum_{j=1}^{n} p(x_i, y_j) \log p(x_i, y_j) \tag{2}$$

When $p(x_i, y_j)$ is the joint distribution probabilities of x_i and y_i, and variable Y is known, y_i is the j-th of Y, then the reserved uncertainty of X is measured by conditional entropy $H(X|Y)$ which is defined as

$$H(X|Y) = -\sum_{i=1}^{m} \sum_{j=1}^{n} p(x_i, y_j) \log p(x_i|y_j) \tag{3}$$

where $p(x_i|y_j)$ is the posterior probabilities of X given Y. And it could be proven that

$$H(X|Y) = H(X, Y) - H(Y) \tag{4}$$

To quantify the information shared by two random variables X and Y, a new concept termed as mutual information (MI) is defined as

$$I(X; Y) = \sum_{i=1}^{m} \sum_{j=1}^{n} p(x_i, y_j) \log \frac{p(x_i|y_j)}{p(x_i)} \tag{5}$$

MI could quantify the relevance between variables, whether liner or nonlinear, and plays a key role in feature selection based on information metric.

Additionally, the MI and the entropy could be related by the following formula

$$I(X; Y) = H(X) - H(X|Y) \tag{6}$$

In addition, conditional mutual information (CMI) of X and Y when given a new random variable Z is defined as

$$I(X; Y|Z) = H(X|Z) - H(X|Y, Z) \tag{7}$$

CMI represents the quantity of information shared by X and Y when Z is known. It implies Y brings information about X which is not already contained in Z.

3. Related Work

Evaluation criterion is the key role in filter methods, which is intended to measure how potentially useful a feature or feature subset should be when used in a classifier. The general evaluation criterion of feature selection based on information metric could be represented as

$$J(f) = I(C;f) - g(C,S,f) \tag{8}$$

where f is a candidate feature, S is the selected feature subset, C is the class vector that evaluates the candidate feature f and $g(C,S,f)$ is a deviated function which is used to penalize or compensate the first part, i.e., $I(C;f)$. Different feature selection methods were proposed by designing modified evaluation criterions according to Equation (8).

A simple method termed as Mutual Information Maximization (MIM) is proposed in [13], which simplifies Equation (8) by removing the deviated function

$$J(f) = I(C;f) \tag{9}$$

Since mutual information tends to favor features with more discrete values, a normalized mutual information criterion named symmetrical uncertainty (SU) [14] is then introduced into the feature selection.

$$J(f) = \frac{2I(C;f)}{H(C) + H(f)} \tag{10}$$

where $H(C)$ and $H(f)$ is defined as Equation (1), $I(C;f)$ is defined as Equation (6). This criterion compensates mutual information's bias towards features with more discrete values and restricts its value to the range of [0,1].

In general, it is widely accepted that an optimal feature set should not only be relevant with the class individually, but also consider feature redundancy. Therefore, other modified criterions have been proposed to pursue the "relevancy-redundancy" goal.

Battiti [15] proposed the Mutual Information Feature Selection (MIFS) criterion:

$$J(f) = I(C;f) - \beta \sum_{f_i \in S} I(f;f_i) \tag{11}$$

This criterion uses mutual information to identify the relevant features, and a penalty to ensure low redundancy within selected features. β is a configurable parameter to determine the trade-off between relevance and redundancy. However, β is set experimentally, which results in unstable performance.

A Minimum-Redundancy Maximum-Relevance (MRMR) criterion was proposed by Peng et al. [16].

$$J(f) = I(C;f) - \frac{1}{|S|} \sum_{f_i \in S} I(f;f_i) \tag{12}$$

where $|S|$ is the number of features in selected feature subset S In this criterion, the deviated function $g(C,S,f) = \frac{1}{|S|} \sum_{f_i \in S} I(f;f_i)$ acts as a penalty to feature redundancy.

Another similar criterion is called Joint Mutual Information (JMI) [17].

$$J(f) = \sum_{f_i \in S} I(f, f_i; C) \tag{13}$$

This criterion could be re-written in the form of Equation (8) by using some relatively simple manipulations.

$$J(f) = I(C;f) - \frac{1}{|S|} \sum_{f_i \in S} [I(f;f_i) - I(f;f_i|C)] \tag{14}$$

In this criterion, $I(f;f_i) - I(f;f_i|C)$ represents the amount of information about C shared by f and f_i. Therefore, the second part of this criterion is another modified deviated function to penalize feature redundancy.

Fleuret [18] proposed the Conditional Mutual Information Maximization (CMIM) criterion

$$J(f) = \min_{f_i \in S}[I(C;f|f_i)] \tag{15}$$

This criterion could also be re-written in the form of Equation (8)

$$J(f) = I(C;f) - \min_{f_i \in S}[I(C;f) - I(C;f|f_i)] \tag{16}$$

Actually, the initial form of this criterion is $J(f) = I(C;f|S)$, since $I(C;f|S)$ is difficult to calculate, it should be approximated by some simplified form. When only taking feature redundancy into consideration, the following inequality is established

$$I(C;f|S) \leq \underset{S_i \in S}{I}(C;f|S_i) \leq \underset{f_i \in S}{I}(C;f|f_i) \tag{17}$$

Therefore, we could estimate $I(C;f|S)$ by using the minimum value, i.e.,

$$I(C;f|S) \approx \min_{f_i \in S}[I(C;f|f_i)] \tag{18}$$

Many other criterions based on information metric have also been proposed, such as FCBF [19], AMIFS [20], CMIFS [21]. Reviewing these criterions, it is easy to find that almost all of these information based criterions focus on selecting relevant features and penalizing feature redundancy by a deviated function, while feature interaction is ignored. As stated above, feature interaction does exist and unintentional ignoring of this feature interaction may result in poor classification performance. Therefore an appropriate deviated function in Equation (8) should not only penalize feature redundancy but also compensate for feature interaction. After taking feature interaction into account, many of the presented criterions would be ill-considered or even improper. Taking CMIM as an example, the inequality (17) would be not tenable once feature interaction is considered, then the final criterion $\min[I(C;f|f_i)]$ would be improper as well. However, little work has been conducted to deal with feature interaction using the information metric.

4. Some Definitions about Feature Relationships

In this section, we first present some classic definitions of feature relevance and redundancy, then provide our formal definitions of feature irrelevance, redundancy and interaction based on information theory.

John et al. [22] classifies features into three disjoint categories, namely, strong relevance, weak relevance and irrelevant features. Then Yu and Liu [18] proposed the definition of redundancy base on the concept of Markov blanket.

Let F be a full set of features, f_i a feature and $S_i = F - \{f_i\}$, C the class vector. These definitions are as follows.

Definition 1. *(Strong relevance) A feature fi is strong relevant if and only if*

$$P(C|F) \neq P(C|S_i) \tag{19}$$

Definition 2. *(Weak relevance) A feature fi is weak relevant if and only if*

$$\begin{gathered} P(C|F) = P(C|S_i) \\ \exists S_i' \subset S_i, \text{ such that } P(C|f_i, S_i') \neq P(C|S_i') \end{gathered} \tag{20}$$

Corollary 1. *(Irrelevance) A feature f_i is irrelevant if and only if*

$$\forall S_i' \subset S_i,\ P(C|f_i, S_i') = P(C|S_i') \tag{21}$$

Definition 3. *(Markov blanket) Given a feature f_i, let $M_i \subset F(f_i \notin M_i)$, M_i is said to be a Markov blanket for f_i if and only if*

$$P(F - M_i - \{f_i\}, C|f_i, M_i) = P(F - M_i - \{f_i\}, C|M_i) \tag{22}$$

Definition 4. *(Redundancy) A feature f_i is redundant if and only if it's weakly relevant and has a Markov blanket M_i within F.*

It's important to note that strong relevance, weak relevance and irrelevance are paratactic, while redundancy is a part of weak relevance. The objective of feature selection is to select an optimal or relatively suboptimal feature subset which contains the strongly relevant and weakly relevant but non-redundant features. However, the above definitions for relevance and redundancy rely on a whole probability distribution, which is intractable to guide feature selection. Moreover, feature interaction has not been defined specifically either.

In the following sections, we will provide some definitions for relevance, redundancy, irrelevance and interaction based on information theory. It's worth noting that the following definitions are based on a candidate feature and a selected feature subset, which is different from the above definitions but could be directly used to guide feature selection especially those using greedy search strategy.

Let S be the subset of features which has been selected, f_i is a candidate feature, C is the class vector.

Definition 5. *(Irrelevance) Feature f_i is irrelevant if and only if*

$$I(f_i; C) = I(f_i; C|S) = 0 \tag{23}$$

According to definition 5, an irrelevant feature f_i couldn't provide any information about C.

Definition 6. *(Strong redundancy) Feature f_i is strongly redundant with S if and only if*

$$I(f_i; C) > I(f_i; C|S) = 0 \tag{24}$$

Where $I(f_i; C) > 0$ suggests that f_i could provide some information about C individually, while $I(f_i; C|S) = 0$ indicates that f_i provides no more information when S is given. Therefore, the information provided by f_i has already been contained in the selected feature subset S, thus f_i should not be selected.

Definition 7. *(Weak redundancy) Feature f_i is weakly redundant with S if and only if*

$$I(f_i; C) > I(f_i; C|S) > 0 \tag{25}$$

According to definition 7, $I(f_i; C) > I(f_i; C|S)$ means the information about C provided by f_i would decrease when S is given, i.e., f_i and S share some information about C. But f_i is still useful and may be selected since it could provide more information even S is known according to $I(f_i; C|S) > 0$.

Definition 8. *(Independent relevance) Feature f_i is independently relevant with S if and only if*

$$I(f_i; C) = I(f_i; C|S) > 0 \tag{26}$$

According to definition 8, the information provided by an independently relevant feature has totally not been contained in S, therefore an independently relevant feature should be selected.

Definition 9. *(Interaction) Feature f_i is interactive with S if and only if*

$$I(f_i; C|S) > I(f_i; C) \geq 0 \tag{27}$$

According to definition 9, $I(f_i;C|S) > I(f_i;C)$ suggests that the information provided by f_i would increase when S is given. Thus feature f_i and feature subset S have a synergy.

In the process of feature selection, when a feature subset is selected, a candidate feature belonging to weak redundancy, independent relevance and interaction should be selected, while irrelevant or strong redundant features would be eliminated.

5. A New Feature Smethod Considering Feature Interaction

The main goal of feature selection is to select a small number of features that can carry as much information as possible. When using information metric, the objective function is defined as max$I(S;C)$, where S is the selected feature subset. Based on this objective function, the evaluation criterion for a candidate feature fi in a greedy search strategy could be represented as [23]

$$J(f_i) = I(C;f_i|S) \tag{28}$$

Re-write this criterion in the form of Equation (8)

$$J_{CMI}(f_i) = I(C;f_i) + [I(f_i;C|S) - I(f_i;C)] \tag{29}$$

To directly calculate JCMI(fi), we need to compute the complex joint probability, which would be computationally intractable. To address this issue, we would like to evaluate JCMI(f_i) by using some approximation technique without the involvement of complex joint probability.

The second part in JCMI (f_i) (termed as $DF = I(f_i;C|S) - I(f_i;C)$) acts as a deviated function, to penalized or compensate the first part. Therefore, a proper approximation of DF should consider both redundancy and interaction. One feasible method is to consider redundancy and interaction severally.

For a candidate feature f_i, the selected S could be divided into three kinds of subsets, which are redundant, independently relevant and interactive with fi respectively, denoted as S_{redu}, S_{inde} and S_{inte}.

The redundancy between S and f_i could be represented by the subset with the highest redundancy degree, denoted as

$$S_{redu} = \arg\min_{S_{redu} \subset S} [I(f_i;C|S_{redu}) - I(f_i;C)] \tag{30}$$

Similarly, the interaction between S and f_i could be represented by the subset with the highest interaction degree, denoted as

$$S_{inte} = \arg\max_{S_{inte} \subset S} [I(f_i;C|S_{inte}) - I(f_i;C)] \tag{31}$$

In addition, according to Definition 8, $I(f_i;C) = I(f_i;C|S_{inde})$, so independently relevant subsets S_{inde} doesn't influence the selection of candidate feature f_i and could be ignored. Therefore, the deviated function DF could be replaced by

$$DF = [I(f_i;C|S_{redu}) - I(f_i;C)] + [I(f_i;C|S_{inte}) - I(f_i;C)] \tag{32}$$

For features in S_{redu}, more features would intensify redundancy, thus

$$I(f_i;C|S_{redu}) \leq \underset{f_j \in S_{redu}}{I} (f_i;C|f_j) \tag{33}$$

We estimate $I(f_i;C|S_{redu})$ by their minimum value, i.e., $I(f_i;C|S_{redu}) \approx \min \underset{f_j \in S_{redu}}{I} (f_i;C|f_j)$.

Similarly, for features in S_{inte}, more features would intensify interaction

$$I(f_i;C|S_{inte}) \geq \underset{f_j \in S_{inte}}{I} (f_i;C|f_j) \tag{34}$$

We estimate $I(f_i; C|S_{inte})$ by their maximum value, i.e., $I(f_i; C|S_{inte}) \approx \max_{f_j \in S_{inte}} I(f_i; C|f_j)$.

Therefore, DF could be approximated by

$$DF \approx [\min_{f_j \in S_{redu}} I(f_i; C|f_j) - I(f_i; C)] + [\max_{f_j \in S_{inte}} I(f_i; C|f_j) - I(f_i; C)] \tag{35}$$

Since the subsets S_{redu}, S_{inte} are all implicit, we should generalize the above DF into the whole subset S. It's easy to prove that $\min_{f_j \in S_{redu}} I(f_i; C|f_j) = \min_{f_j \in S} I(f_i; C|f_j)$. Therefore, the first part of DF is denoted as $\min[[\min_{f_j \in S} I(f_i; C|f_j) - I(f_i; C)], 0]$, comparing with zero in case of $S_{redu} = \varnothing$. Similarly, the second part of DF is denoted as $\max[[\max_{f_j \in S} I(f_i; C|f_j) - I(f_i; C)], 0]$.

Finally, the deviated function DF is represented as

$$DF = \min[[\min_{f_j \in S} I(f_i; C|f_j) - I(f_i; C)], 0] + \max[[\max_{f_j \in S} I(f_i; C|f_j) - I(f_i; C)], 0] \tag{36}$$

And the evaluation criterion of a candidate f_i is defined as

$$\begin{aligned} J_{CMI}(f_i) &= I(C; f_i) + DF \\ &= I(C; f_i) + \min[[\min_{f_j \in S} I(f_i; C|f_j) - I(f_i; C)], 0] + \max[[\max_{f_j \in S} I(f_i; C|f_j) - I(f_i; C)], 0], \end{aligned} \tag{37}$$

It's important to note that this new evaluation criterion makes use of a similar idea with CMIM, except taking feature interaction into consideration. And it would degenerate to the CMIM criterion once interaction is ignored.

With this newly defined evaluation criterion, a new feature selection algorithm termed as CMIFSI is proposed in this paper, the details of the Algorithm 1 are as follows

Algorithm 1 CMIFSI algorithm

Input: A training dataset D with a full feature set $F = \{f_1, f_2, \ldots f_n\}$ and class vector C
A predefined threshold K
Output: The selected feature sequence
1. Initialize parameters: the selected feature subset $S = \varnothing$, $k = 0$, deviated function $DF(f_i, S) = 0$ for all candidate features;
2. for $i = 1$ to n do
3. Calculate $I(f_i;C)$
4. end
5. While $k < K$ do
6. For each candidate feature $f_i \in F$ do
7. Update $DF(f_i, S)$ according to Equation (36)
8. Calculate the evaluation value
$J_{CMI}(f_i) = I(f_i;C) + DF(f_i, S)$
9. End
10. Select the feature f_j with the largest $J_{CMI}(f_i)$
$S = S \cup f_j$
$F = F - f_j$
11. $k = k + 1$
12. end

As shown in Algorithm 1, a sequential forward search strategy was adopted in this algorithm and the procedure was terminated by a predefined threshold K. The low K value can achieve low computation complexity but may lose many effective features that are useful, while a high K value can achieve better classification accuracy but also entails high computation complexity. Actually,

when the threshold is exceeded, the classification accuracy doesn't increase much but the computation complexity still increases. As a result, an appropriate K value is needed. The features are ranked in descending order according to the new evaluation criterion.

Now we analyze the time complexity of our algorithm. Suppose the total number of candidate feature is n, and the predefined threshold is K. When k features have been selected, the complexity of updating the deviated function is O(n-k) for each while loop. Therefore, the complexity for K selected features is n + (n − 1) + (n − 2)+ . . . + (n − K + 1) = nK − 1/2(K2 − K), namely its time complexity is O(nK). In the worst case, when all candidate features are selected, i.e., K = n, the time complexity is O(n2).

6. Experiments and Results

In this section, we empirically evaluate the effectiveness of the proposed algorithm by comparing it with some other representative feature selection algorithms using both synthetic and benchmark datasets. The experiment setup is described in Section 6.1, while the results and discussion on synthetic and benchmark datasets are shown in Sections 6.2 and 6.3, respectively.

6.1. Experiment Setup

To evaluate the effectiveness of a feature selection algorithm, a simple and direct criterion is the similarity degree between the selected subset and the optimal subset, but it can only be measured using synthetic data whose optimal subset is known beforehand. For real-world data like some benchmark datasets, such prior knowledge in unavailable and we usually use the predictive accuracy on selected subset of features as an indirect measure.

Six representative feature selection algorithms (GA, SU, Relief [24], CFS [25], MRMR, CMIM) were selected to compare with CMIFSI using both synthetic and benchmark datasets. All of these methods could effectively identify irrelevant features and some of them (e.g. CFS, MRMR, CMIM) could detect redundant features as well.

In the experiment on benchmark datasets, two different learning algorithms (C4.5 and SVM) were used to evaluate the predictive accuracy on the selected feature subsets. This meant we could verify whether the performance of our new algorithm would be limited to the specified learning algorithm.

The experiment is mainly conducted in the WEKA (WEKA is a software and can be available at http://www.cs.waikato.ac.nz/~{}ml) environment with the default settings. Some of these feature selection algorithms and learning algorithms (like SU, Relief, CFS, C4.5) could be found in the WEKA environment, other algorithms were implemented in MATLAB. To achieve impartial results, five-fold validation and ten-fold cross validation were adopted for each step in selecting features and verifying the classification capability, respectively.

6.2. Experiment on Synthetic Datasets

6.2.1. Synthetic Datasets

In this section, five synthetic datasets were employed to evaluate the effectiveness of different feature selection algorithms. These synthetic datasets were generated by the data generation RDG1 in a WEKA environment. The description of these five datasets is as follows:

Data1

There are 100 instances and 10 Boolean features denoted as a_0,a_1,a_2 . . . ,a_9 with 2 classes. Six of these ten features are irrelevant and the target concept was defined as

$$c_1 = \bar{a}_5 \vee (\bar{a}_1 \wedge \bar{a}_6).$$

Data2

There are 100 instances and 10 Boolean features denoted as a_0,a_1,a_2 ... ,a_9 with 2 classes. Five of the ten features are irrelevant and the target concept was defined as

$$c_1 = (a_0 \wedge a_1 \wedge \overline{a}_5) \vee (a_8 \wedge a_0 \wedge \overline{a}_6) \vee \overline{a}_0.$$

Data3

There are 100 instances and 10 Boolean features denoted as a_0,a_1,a_2 ... ,a_9 with 2 classes. Six of the ten features are irrelevant and the target concept was defined as

$$c_1 = \overline{a}_5 \vee (\overline{a}_1 \wedge \overline{a}_6 \wedge \overline{a}_8).$$

Data4

There are 200 instances and 15 Boolean features denoted as a_0,a_1,a_2 ... ,a_{14} with 3 classes. Eleven of the fifteen features are irrelevant and the target concept was defined as

$$\begin{aligned} c_0 &= \overline{a}_2 \vee (a_2 \wedge a_{12} \wedge \overline{a}_8) \\ c_1 &= a_8 \vee (a_2 \wedge \overline{a}_{12} \wedge \overline{a}_8) \\ c_2 &= \textit{therest} \end{aligned}$$

Data5

There are 100 instances and 10 Boolean features denoted as a_0,a_1,a_2 ... ,a_9 with 2 classes. Six of the ten features are irrelevant and the target concept was defined as

$$c_1 = (\overline{a}_1 \wedge \overline{a}_5) \oplus (\overline{a}_0 \wedge \overline{a}_6).$$

Features absent in the definitions of the target concepts are redundant or irrelevant, and the relevant and interactive features in each synthetic dataset are shown in Table 1.

Table 1. Relevant and interactive features of the four synthetic datasets.

Dataset	Relevant Features	Interactive Features
Data1	a_1,a_5,a_6	(a_1,a_6)
Data2	a_0,a_1,a_5,a_6,a_8	(a_0,a_1,a_5),(a_0,a_6,a_8)
Data3	a_1,a_5,a_6,a_8	(a_1,a_6,a_8)
Data4	a_2,a_8,a_{12},a_{13}	(a_2,a_8,a_{12})
Data5	a_0,a_1,a_5,a_6	(a_1,a_5),(a_0,a_6)

6.2.2. Results on Synthetic Datasets

Feature selection methods could be divided into two types: subset selection and feature ranking [1]. Subset selection preserves relevant features and removes as much irrelevant and redundant features as possible, while feature ranking ranks features in a descending order according to specific evaluation criterions and the number of selected features is predefined. In this experiment, GA, SU, Relief, MRMR, CMIM, CMIFSI belonged to feature ranking and CFS was a kind of subset selection method. The feature selection/ranking results are shown in Table 2 with no threshold predefined. The bold values in entries represent features belong to the optimal subset and the notation" *" denotes correct selected/ranked features.

It can be seen that when comparing with other feature selection algorithms, CMIFSI achieves the best performance. For data1, data2 and data3, CMIFSI ranks the optimal subset in the top, which means the feature ranking result is more accurate than the other results. For data4 and data5, all feature selection algorithms failed to obtain the correct results, but CMIFSI still performed better than other algorithms since its feature ranking sequence is more similar to the optimal subset. This is mainly

because feature interaction really exists in these datasets (as shown in Table 1), but all algorithms except for CMIFSI just focus on detecting irrelevant and redundant features while feature interaction is ignored, which results in poor performance. Among the other feature selection algorithms, CMIM and Relief performed with approximately suboptimal results on all the four synthetic datasets, and other methods failed to obtain satisfying results.

Table 2. Feature selection results on synthetic datasets.

Algorithm	**Data1**	**Data2**
GA	$\boldsymbol{a_1}$,$\boldsymbol{a_2}$,a_5,a_0,$\boldsymbol{a_6}$,a_9,a_4,a_7,a_3,a_8	$\boldsymbol{a_8}$,$\boldsymbol{a_0}$,$\boldsymbol{a_6}$,$\boldsymbol{a_5}$,a_7,a_3,a_2,$\boldsymbol{a_1}$,a_9,a_4
SU	$\boldsymbol{a_1}$,$\boldsymbol{a_5}$,a_2,a_0,a_7,a_9,a_8,a_4,a_3,$\boldsymbol{a_6}$	$\boldsymbol{a_8}$,$\boldsymbol{a_0}$,$\boldsymbol{a_6}$,a_7,$\boldsymbol{a_1}$,$\boldsymbol{a_5}$,a_4,a_3,a_2,a_9
Relief	$\boldsymbol{a_1}$,$\boldsymbol{a_5}$,$\boldsymbol{a_6}$,a_2,a_7,a_7,a_4,a_9,a_0,a_3*	$\boldsymbol{a_8}$,$\boldsymbol{a_0}$,$\boldsymbol{a_6}$,$\boldsymbol{a_1}$,$\boldsymbol{a_5}$,a_7,a_2,a_3,a_4,a_9*
CFS	$\boldsymbol{a_1}$,a_2,$\boldsymbol{a_5}$	$\boldsymbol{a_0}$,$\boldsymbol{a_6}$,a_7,$\boldsymbol{a_8}$
MRMR	$\boldsymbol{a_1}$,a_0,a_7,a_8,$\boldsymbol{a_6}$,$\boldsymbol{a_5}$,a_2,a_9,a_4,a_3	$\boldsymbol{a_8}$,$\boldsymbol{a_6}$,$\boldsymbol{a_5}$,a_3,a_2,$\boldsymbol{a_0}$,a_7,$\boldsymbol{a_1}$,a_9,a_4
CMIM	$\boldsymbol{a_1}$,$\boldsymbol{a_6}$,$\boldsymbol{a_5}$,a_2,a_7,a_9,a_0,a_8,a_3,a_4*	$\boldsymbol{a_8}$,$\boldsymbol{a_0}$,$\boldsymbol{a_6}$,$\boldsymbol{a_5}$,a_7,$\boldsymbol{a_1}$,a_4,a_3,a_9,a_2
CMIFSI	$\boldsymbol{a_1}$,$\boldsymbol{a_6}$,$\boldsymbol{a_5}$,a_4,a_2,a_9,a_7,a_8,a_3,a_0*	$\boldsymbol{a_8}$,$\boldsymbol{a_0}$,$\boldsymbol{a_1}$,$\boldsymbol{a_6}$,$\boldsymbol{a_5}$,a_7,a_4,a_2,a_9,a_3*
Optimal subset	a_1,a_5,a_6	a_0,a_1,a_5,a_6,a_8
Algorithm	**Data3**	**Data4**
GA	$\boldsymbol{a_5}$,$\boldsymbol{a_6}$,a_9,$\boldsymbol{a_1}$,a_3,a_2,a_0,a_7,$\boldsymbol{a_8}$,a_4	$\boldsymbol{a_2}$,$\boldsymbol{a_{13}}$,a_{11},$\boldsymbol{a_8}$,a_1,a_{14},a_5,$\boldsymbol{a_{12}}$,a_0,a_6,a_{10},a_7,a_3,a_4,a_9
SU	$\boldsymbol{a_5}$,$\boldsymbol{a_1}$,$\boldsymbol{a_6}$,a_7,a_0,a_3,a_4,a_9,$\boldsymbol{a_8}$,a_2	$\boldsymbol{a_2}$,$\boldsymbol{a_8}$,$\boldsymbol{a_{13}}$,a_{10},a_6,a_{11},a_0,a_1,a_{14},a_4,a_7,a_5,a_3,a_9,$\boldsymbol{a_{12}}$
Relief	$\boldsymbol{a_5}$,$\boldsymbol{a_1}$,$\boldsymbol{a_6}$,a_2,a_0,a_9,a_4,$\boldsymbol{a_8}$,a_7,a_3	$\boldsymbol{a_2}$,$\boldsymbol{a_8}$,$\boldsymbol{a_{13}}$,a_{11},a_{14},a_{10},a_7,a_4,$\boldsymbol{a_{12}}$,a_1,a_6,a_3,a_5,a_0,a_9
CFS	a_0,$\boldsymbol{a_1}$,$\boldsymbol{a_5}$,a_7	a_0,$\boldsymbol{a_2}$,$\boldsymbol{a_8}$,a_{10},a_{11},$\boldsymbol{a_{13}}$,a_{14}
MRMR	$\boldsymbol{a_5}$,$\boldsymbol{a_6}$,a_3,a_2,a_9,$\boldsymbol{a_1}$,a_0,a_4,a_7,$\boldsymbol{a_8}$	$\boldsymbol{a_2}$,$\boldsymbol{a_{13}}$,a_{11},a_1,$\boldsymbol{a_8}$,a_7,a_5,$\boldsymbol{a_{12}}$,a_{10},a_6,a_0,a_{14},a_4,a_3,a_9
CMIM	$\boldsymbol{a_5}$,$\boldsymbol{a_6}$,$\boldsymbol{a_1}$,a_0,a_7,a_2,a_4,$\boldsymbol{a_8}$,a_9,a_3	$\boldsymbol{a_2}$,$\boldsymbol{a_8}$,$\boldsymbol{a_{13}}$,a_4,a_{10},a_1,a_0,a_{14},a_{11},a_6,$\boldsymbol{a_{12}}$,a_3,a_5,a_7,a_9
CMIFSI	$\boldsymbol{a_5}$,$\boldsymbol{a_6}$,$\boldsymbol{a_1}$,$\boldsymbol{a_8}$,a_2,a_4,a_0,a_9,a_7,a_3*	$\boldsymbol{a_2}$,$\boldsymbol{a_8}$,$\boldsymbol{a_{13}}$,a_4,$\boldsymbol{a_{12}}$,a_{11},a_5,a_1,a_{14},a_{10},a_3,a_0,a_6,a_9,a_7
Optimal subset	a_1,a_5,a_6,a_8	a_2,a_8,a_{12},a_{13}
Algorithm	**Data5**	
GA	$\boldsymbol{a_1}$,$\boldsymbol{a_6}$,a_9,$\boldsymbol{a_5}$,a_3,a_2,a_8,a_4,$\boldsymbol{a_0}$,a_7	
SU	$\boldsymbol{a_1}$,$\boldsymbol{a_5}$,$\boldsymbol{a_6}$,a_2,a_9,$\boldsymbol{a_0}$,a_7,a_3,a_4,a_8	
Relief	$\boldsymbol{a_5}$,$\boldsymbol{a_1}$,$\boldsymbol{a_6}$,a_2,a_9,a_4,a_8,$\boldsymbol{a_0}$,a_7,a_3	
CFS	$\boldsymbol{a_6}$,a_2,$\boldsymbol{a_1}$,$\boldsymbol{a_5}$	
MRMR	$\boldsymbol{a_1}$,$\boldsymbol{a_6}$,a_2,a_9,$\boldsymbol{a_5}$,a_3,$\boldsymbol{a_0}$,a_8,a_7,a_4	
CMIM	$\boldsymbol{a_5}$,$\boldsymbol{a_6}$,$\boldsymbol{a_1}$,a_9,$\boldsymbol{a_0}$,a_2,a_4,a_7,a_8,a_3	
CMIFSI	$\boldsymbol{a_5}$,$\boldsymbol{a_6}$,$\boldsymbol{a_1}$,a_2,$\boldsymbol{a_0}$,a_9,a_4,a_8,a_7,a_3	
Optimal subset	a_0,a_1,a_5,a_6	

The above results and discussion demonstrate the necessity of taking feature interaction into consideration in feature selection. For datasets that involved interactive features, most of the traditional feature selection algorithms would fail to achieve an optimal result. Take data4 for example, a_{12} is a relevant feature in the optimal subset which is interactive with a_2, a_8, traditional algorithms remove it or rank it at the back of the feature sequence mainly because its correlation with the class is low individually and its interaction with other features is ignored. However, during the selection process of CMIFSI, once a_2 and a_8 are selected, the evaluation criterion value of a_{12} would be compensated because of its interaction with a_2, a_8, which would increase its probability of being selected. Therefore, the result of CMIFSI is found to be superior to others.

6.3. Experiment on Benchmark Datasets

6.3.1. Benchmark Datasets

Ten datasets from the UCI Machine Learning Repository [26] are adopted in our simulation experiments. These datasets contain various numbers of features, instances, and classes, as shown in Table 3. At the same time, the distribution of each class in terms of number of instances is shown as Figure 1.

Data preprocess was applied before feature selection. Missing values were replaced by the most frequently used values and means for nominal and numeric features, respectively. For algorithms based on information metric, the MDL discretization method was applied to transform the numerical features into discrete ones. Algorithms conducted in WEKA are set with default parameters. For SVM, the grid searching method was adopted to obtain its relatively optimal parameters. In addition, we selected the top K features that produce the highest accuracy and limited their maximum to 20 (used in GA, SU, Relief MRMR, CMIM and CMIFSI), since the objective of feature selection is to reduce the original feature dimension.

Table 3. Summary of UCI benchmark datasets.

No.	Datasets	Features	Instances	Classes
1	Wine	13	178	3
2	Kr-vs-kp	36	3196	2
3	SPECTF-heart	44	267	2
4	Zoo	16	101	7
5	Credit Approval	15	690	2
6	Optical Recognition of Handwritten Digits	64	1797	10
7	Contraceptive Method Choice	9	1473	3
8	Congressional Voting Records	16	435	2
9	Waveform	21	5000	3
10	Waveform+noise	40	5000	3

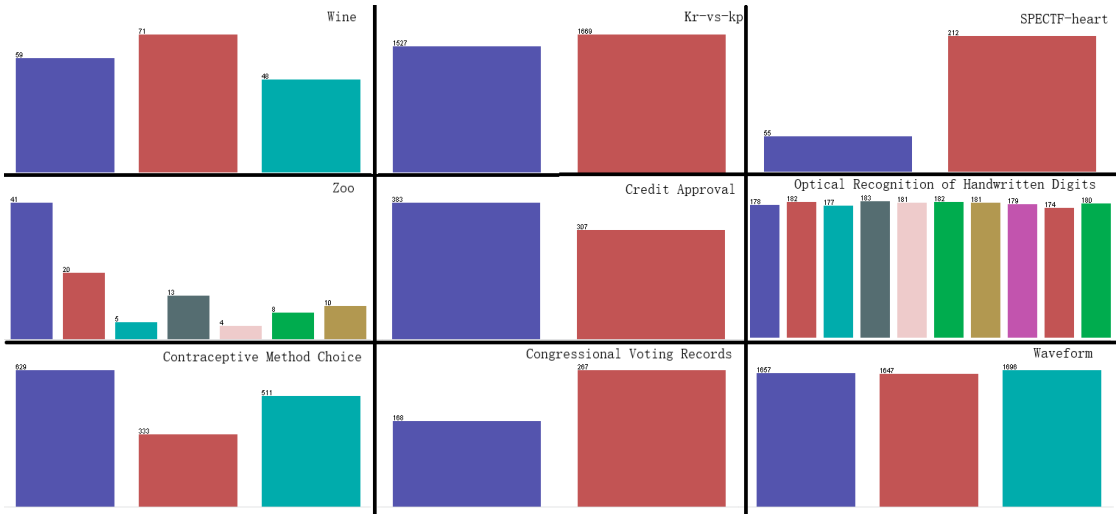

Figure 1. Distribution of each class in terms of number of instances.

6.3.2. Results on Benchmark Datasets

Tables 4 and 5 record the number of features selected by different feature selection algorithms using C4.5 and SVM, respectively. It is shown that all these feature selection algorithms achieve reduction of dimensionality by selecting only a portion of the original features. Furthermore, CMIFSI tends to obtain smaller feature subsets than those of other feature selection algorithms. AVE is the average of the same selected features of the same dataset. From the table, we can see that in most cases CMIFSI outperform the other algorithms.

Tables 6 and 7 show the 10-fold cross-validation accuracies of C4.5 and SVM respectively, where "Unselected" depicts the accuracies on datasets with original features. The bold values indicate the highest accuracies among these six feature selection algorithms using the same classifier. Notation"*" denotes the highest accuracies in each dataset corresponding to a specific classifier, including "Unselected". The last row "W/T/L" in each table summarizes the wins/ties/losses in accuracy over all datasets by comparing various feature sets with those selected by CMIFSI. At the same time,

we computed the confidence intervals of all our results to evaluate the algorithms. Here, we used the bootstrapping method with 1000 as the sampling number and a 95% confidence level. The results are shown in Tables 8 and 9.

Table 4. Number of features selected by different feature selection algorithms using C4.5.

No.	C4.5								
	Total	**GA**	**SU**	**Relief**	**CFS**	**MRMR**	**CMIM**	**CMIFSI**	**AVE.**
1	13	6	3	5	11	3	3	3	4.86
2	36	15	13	15	7	14	19	19	14.57
3	44	3	2	1	21	1	2	3	4.71
4	16	10	14	9	9	10	4	4	8.57
5	15	8	6	9	6	4	3	7	6.14
6	64	20	15	20	38	17	20	20	21.43
7	9	8	7	7	8	8	7	4	7.0
8	16	12	13	9	5	12	10	6	9.57
9	21	17	17	16	16	17	16	9	15.43
10	40	16	13	11	15	16	11	13	13.57

Table 5. Number of features selected by different feature selection algorithms using SVM.

No.	SVM								
	Total	**GA**	**SU**	**Relief**	**CFS**	**MRMR**	**CMIM**	**CMIFSI**	**AVE.**
1	13	2	5	8	11	9	6	8	7.0
2	36	13	13	20	7	12	15	15	13.57
3	44	15	18	5	21	14	8	19	14.29
4	16	9	9	8	9	7	9	5	8.0
5	15	6	5	1	6	6	4	3	4.43
6	64	20	19	13	38	17	11	11	18.43
7	9	8	5	5	8	9	3	7	6.43
8	16	6	2	1	5	3	5	4	3.71
9	21	19	19	18	16	18	20	19	18.43
10	40	20	20	19	15	18	17	16	18.29

Table 6. Accuracy of selected features using C4.5.

No.	C4.5							
	Unselected	**GA**	**SU**	**Relief**	**CFS**	**MRMR**	**CMIM**	**CMIFSI**
1	93.80	92.67	96.07	94.38	93.80	**97.19***	**97.19***	**97.19***
2	99.31*	95.48	96.62	97.70	94.09	96.65	96.53	**97.74**
3	74.90	79.40	79.78	79.40	77.90	79.40	79.78	**80.15***
4	92.08	**95.05***	**95.05***	**95.05***	93.07	**95.05***	94.06	94.06
5	84.93	86.24	86.09	86.81	86.81	**86.96***	85.65	85.94
6	87.42	86.16	**87.48***	86.94	87.42	87.20	86.92	87.26
7	53.22	55.53	55.53	55.53	54.58	45.96	55.53	**56.21***
8	96.32	95.86	96.32	96.32	94.94	**96.78***	96.32	96.32
9	75.94	76.82	77.04	76.92	76.76	77.04	77.16	**77.22***
10	75.08	77.40	77.82	**77.92***	77.30	76.58	77.82	77.82
Ave.	83.30	84.06	84.78	84.71	83.67	83.88	84.70	**84.99**
W/T/L	1/1/8	2/0/8	3/2/5	3/2/5	2/0/8	3/1/6	0/4/6	

Table 7. Accuracy on selected features using SVM.

No.	SVM							
	Unselected	**GA**	**SU**	**Relief**	**CFS**	**MRMR**	**CMIM**	**CMIFSI**
1	94.44	97.85	97.22	98.89	95.56	98.89	98.33	**99.40***
2	94.91	95.03	95.68	**97.80***	94.87	96.21	95.03	96.94
3	79.78	82.57	83.70	83.33	81.48	82.22	82.59	**84.81***
4	93.07	97.27	94.54	96.36	88.18	97.27	**98.18***	97.27
5	85.51	87.79	88.10	87.82	88.50	87.39	87.10	**88.55***
6	95.11*	88.28	**90.78**	88.39	88.28	89.44	88.00	88.00
7	51.28	53.85	54.32	53.85	54.79	50.81	54.46	**55.34***
8	96.09	94.31	95.90	95.45	93.86	96.59	96.59	**97.27***
9	84.08	85.59	86.22	85.70	85.20	86.12	**86.56***	86.24
10	86.02	86.12	86.18	86.12	85.50	**86.42***	85.82	86.18
Ave.	86.03	86.87	87.26	87.37	85.62	87.14	87.27	**88.00**
W/T/L	1/0/9	1/1/8	1/1/8	1/0/9	1/0/9	2/1/7	2/2/6	

Table 8. Confidence intervals on selected features using C4.5.

No.	C4.5							
	Unselected	**GA**	**SU**	**Relief**	**CFS**	**MRMR**	**CMIM**	**CMIFSI**
1	[93.41,94.35]	[92.04,92.96]	[95.73,96.70]	[93.50,94.46]	[93.20,94.16]	[96.77,97.70]	[96.53,97.70]	[96.84,97.86]
2	[98.62,99.58]	[94.92,95.81]	[95.95,97.00]	[97.55,98.50]	[93.49,94.36]	[95.59,96.62]	[95.81,96.76]	[97.68,98.66]
3	[74.45,75.44]	[78.76,79.71]	[79.84,80.90]	[79.10,80.08]	[77.59,78.66]	[78.78,79.80]	[77.96,79.80]	[79.68,80.65]
4	[91.92,92.86]	[94.48,95.46]	[94.55,95.55]	[94.68,95.59]	[92.53,93.52]	[94.19,95.23]	[93.58,94.61]	[93.56,94.51]
5	[84.03,85.05]	[85.80,86.73]	[85,58,86.40]	[86.50,87.55]	[86.04,87.17]	[86.84,87.72]	[85.13,86.10]	[85.14,86.00]
6	[86.87,87.73]	[85.57,86.60]	[87.20,88.09]	[86.40,87.49]	[87.05,87.92]	[86.71,87.74]	[86.53,87.44]	[86.36,87.40]
7	[52.47,53.93]	[54.29,55.75]	[55.06,56.35]	[55.13,56.60]	[53.82,55.66]	[45.45,46.71]	[53.85,55.38]	[55.34,56.33]
8	[96.32,96.74]	[95.56,96.37]	[95.81,96.72]	[95.56,96.38]	[94.08,94.87]	[96.16,96.95]	[96.06,96.87]	[96.16,97.05]
9	[75.19,76.16]	[76.63,77.52]	[76.67,77.56]	[76.58,77.51]	[76.33,77.30]	[76.51,77.47]	[76.45,77.31]	[77.09,78.09]
10	[74.58,75.58]	[77.01,77.97]	[76.92,77.70]	[77.86,78.77]	[76.97,77.86]	[75.96,76,91]	[77.10,78.06]	[77.27,78.15]

Table 9. Confidence intervals on selected features using SVM.

No.	SVM							
	Unselected	**GA**	**SU**	**Relief**	**CFS**	**MRMR**	**CMIM**	**CMIFSI**
1	[93.95,94.87]	[97.71,98.52]	[96.80,97.64]	[98.59,99.35]	[95.14,95.95]	[98.48,99.27]	[97.61,98.36]	[98.47,99.52]
2	[94.40,95.27]	[94.81,95.60]	[95.14,95.92]	[97.50,98.32]	[94.22,94.96]	[95.35,96.23]	[94.46,95.31]	[96.83,97.60]
3	[78.68,79.49]	[82.39,83.31]	[83.05,83.84]	[82.94,83.73]	[80.81,81.55]	[81.98,82.88]	[82.50,83.39]	[84.32,85.13]
4	[92.38,93.33]	[96.71,97.42]	[94.31,95.20]	[95.56,96.35]	[87.56,88.47]	[96.77,97.61]	[96.78,98.55]	[96.88,97.77]
5	[85.26,86.08]	[87.52,88.27]	[87,58,88.46]	[87.01,87.85]	[88.22,89.14]	[86.77,87.52]	[86.49,87.38]	[88.56,89.40]
6	[94.68,95.60]	[87.69,88.44]	[90.39,91.22]	[87.87,88.77]	[87.88,88.65]	[89.07,89.98]	[87.66,88.46]	[87.20,88.03]
7	[51.24,52.02]	[53.41,54.22]	[53.74,54.59]	[53.63,54.46]	[54.34,55.17]	[50.53,51.38]	[54.02,54.79]	[55.18,55.98]
8	[95.82,96.60]	[93.64,94.47]	[95.34,96.21]	[95.30,96.18]	[93.60,94.38]	[96.13,96.97]	[96.37,97.23]	[96.53,97.28]
9	[83.79,84.59]	[85.13,86.00]	[85.43,86.40]	[85.06,85.80]	[84.89,85.80]	[85.77,86.63]	[86.18,86.98]	[86.10,86.90]
10	[85.69,86.58]	[86.33,87.13]	[86.68,87.60]	[87.45,88.21]	[84.86,85.69]	[86.94,87.77]	[86.80,87.62]	[87.71,88.56]

The results in Tables 6 and 7 show that CMIFSI tends to outperform other feature selection algorithms on these ten benchmark datasets, regardless of whether C4.5 or SVM is used. The proposed method gets the highest classification accuracy on five datasets of ten. The average classification accuracies of CMIFSI are higher than others in both tables. From the view of "W/T/L", CMIFSI is also relatively superior to other selectors. More specifically, CMIFSI obtains 5 maximal classification accuracies (denoted by bold value) over ten datasets whether use C4.5 or SVM.

CMIFSI and CMIM are similar to some extent, since they apply the same method to evaluate feature redundancy, except that the feature interaction is ignored by CMIM. Actually, CMIFSI could be regarded as a modification to the original CMIM. Therefore, it's worth comparing these two algorithms to verify whether the modification to CMIM is worthwhile. The results in Tables 6 and 7 show that CMIFSI outperforms CMIM in almost all of the datasets. In contrast, MRMR, which uses a different

way to evaluate feature redundancy, achieves higher classification accuracies than CMIFSI in some cases, even though its average accuracy is lower than CMIM and CMIFSI in both tables.

Apart from comparison among feature selectors, it's interesting to find that for some cases such as the 6th dataset using C4.5 and SVM, the accuracies based on feature selection tend to be lower than the ones based on the original features. This doesn't mean that feature selection deteriorated the classification performance. Actually it mainly resulted from a limitation in this experiment. Since we restricted the maximal number of features selected to 20, when the optimal feature subset consisted of more than 20 features, feature selector resulted in poor performance.

To further evaluate the effectiveness of our new algorithm, another experiment was conducted to compare different feature ranking algorithms (GA, SU, Relief, MRMR, CMIM and CMIFSI) by adding features for learning one by one in the order that the features are ranked. This experiment was applied to two datasets with more than 20 features (i.e., kr-vs-kp and waveform). These two datasets with different number of selected features were tested on both C4.5 and SVM, and the average classification accuracies of these features are shown in Figure 2.

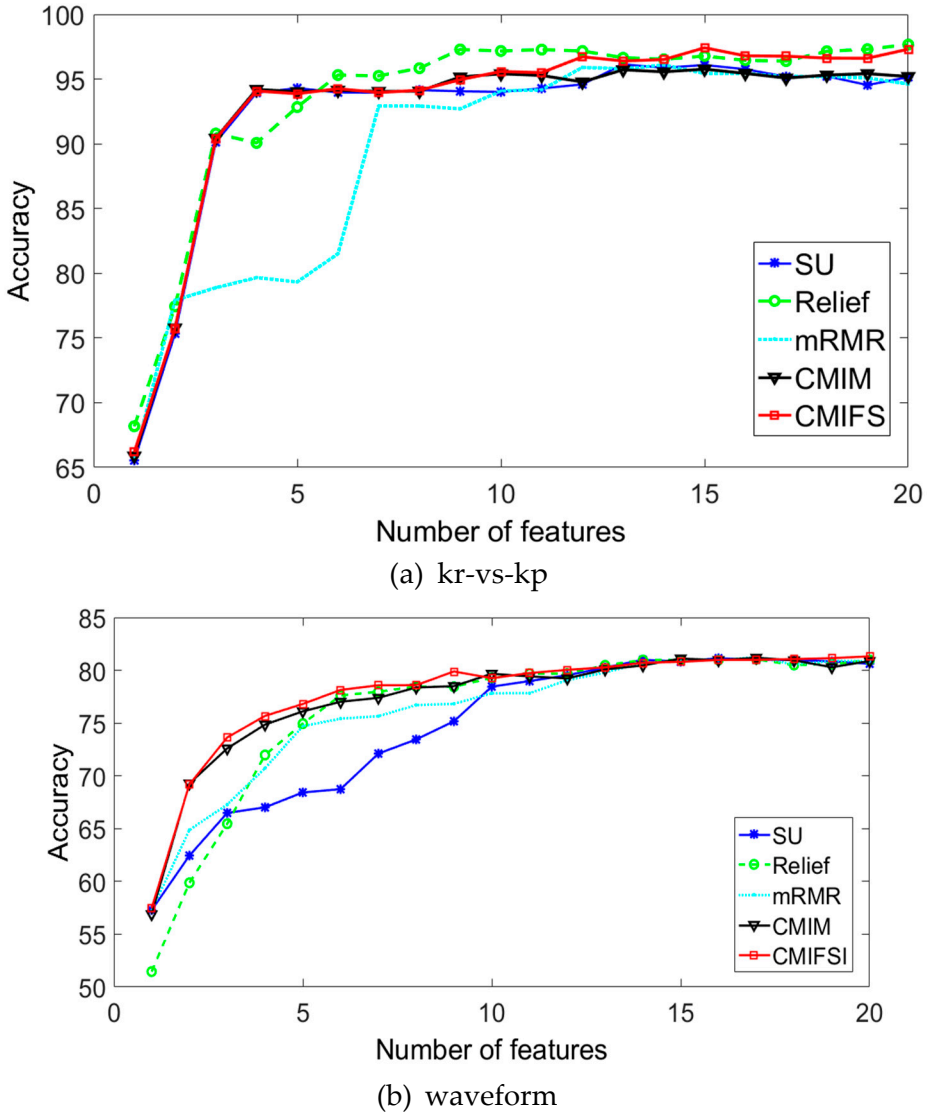

Figure 2. Average accuracy with different number of selected features using different selector.

The result in Figure 2a shows that Relief seems to outperform other selectors in the majority of cases for dataset "kr-vs-kp". The result may be confusing since Relief does not consider feature

redundancy and interaction. A similar result was obtained by reference [27] on this dataset and this may be due to the properties inherent to this dataset. For other selectors which are all based on information metric, CMIFSI achieves the best performance, which is comparable to Relief to some extent. And the superiority of CMIFSI increases as more features are added for learning. For dataset "waveform" (shown in Figure 2b), CMIFSI outperforms other selectors in most cases. For instance, all plots of CMIFSI are higher than others when feature number is less than ten especially in the number of 9. As more features are added, all selectors tend to perform comparably. It is worth noting that with the increase of the number of features, it more sources may needed for calculation as part of the process of prediction.

7. Discussion and Conclusions

In this paper, several feature selection algorithms based on information metric were reviewed in the framework of a general evaluation criterion. Then feature interaction was introduced and some new definitions were proposed to better analyze feature relationships. The state-of-the art methods like CMIM based on conditional mutual information are not rigorous enough to select features, because they don't consider the relationship between features beyond relevance and redundancy. To address the drawback that most of these traditional feature selection methods ignore feature interaction, a new algorithm CMIFSI based on conditional mutual information was proposed to take interaction into consideration. The main idea of CMIFSI is to penalize feature redundancy and compensate feature interaction in the evaluation criterion. Experiments were conducted to compare CMIFSI with other 6 up-to-data feature selection algorithms on both synthetic and benchmark datasets, and the results showed that CMIFSI works well and outperforms other algorithms in most cases. Many features are related to each other, so if we only select features by considering relevant or redundant features while ignoring feature interactions, some feature interactions may cause bad performance on the results. Therefore, the methods that consider feature interaction perform better on some datasets than the methods that do not. Based on the mutual information methods, we proposed exploiting feature interaction to capture more information, making the classifier more efficient for prediction.

Further work is still needed to improve the performance stability of this new algorithm. Furthermore, while CMIFSI adopts an approximation method to estimate feature redundancy and interaction, other new methods are called for to better handle feature interaction. It is still a challenging task to deal with feature interaction.

Author Contributions: Methodology, L.H.; software, Z.L.; writing—original draft preparation, W.H.; writing—review and editing, L.H.; supervision, J.L.

Funding: This research was funded by the National Natural Science Foundation of China (U1664264,U1509203).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Guyon, I.; Elisseeff, A. An Introduction to Variable and Feature Selection. *J. Mach. Learning Res.* **2002**, *3*, 1157–1182.
2. Maldonado, S.; Weber, R. A wrapper method for feature selection using support vector machines. *Inf. Sci.* **2009**, *179*, 2208–2217. [CrossRef]
3. Bolón-Canedo, V.; Sánchez-Maroño, N.; Alonso-Betanzos, A. Recent advances and emerging challenges of feature selection in the context of big data. *Knowledge-Based Syst.* **2015**, *86*, 33–45. [CrossRef]
4. Dash, M.; Liu, H. Feature Selection for Classification. *Intel. Data Anal.* **1997**, *1*, 131–156. [CrossRef]
5. Jakulin, A.; Bratko, I. Testing the Significance of Attribute Interactions. In Proceedings of the twenty-first international conference on Machine learning, Banff, AB, Canada, 4–8 July 2004.
6. Zhao, Z.; Liu, H. Searching for interacting features in subset selection. *Intel. Data Anal.* **2009**, *13*, 207–228. [CrossRef]

7. Zeng, Z.; Zhang, H.; Zhang, R.; Yin, C. A novel feature selection method considering feature interaction. *Pattern Recogn.* **2015**, *48*, 2656–2666. [CrossRef]
8. Tao, H.; Hou, C.; Nie, F.; Jiao, Y.; Yi, D. Effective Discriminative Feature Selection with Nontrivial Solution. *IEEE Trans. Neural Netw. Learning Syst.* **2016**, *27*, 3013–3017. [CrossRef] [PubMed]
9. Murthy, C.A.; Chanda, B. Generation of Compound Features based on feature Interaction for Classification. *Expert Syst. Appl.* **2018**, *108*, 61–73.
10. Chang, X.; Ma, Z.; Lin, M.; Yang, Y.; Hauptmann, A.G. Feature Interaction Augmented Sparse Learning for Fast Kinect Motion Detection. *IEEE Trans. Image Process.* **2017**, *26*, 3911–3920. [CrossRef] [PubMed]
11. Yin, Y.; Zhao, Y.; Zhang, B.; Li, C.; Guo, S. Enhancing ELM by Markov Boundary Based Feature Selection. *Neurocomputing* **2017**, *261*, 57–69. [CrossRef]
12. Brown, G.; Pocock, A.; Zhao, M.J.; Luján, M. Conditional Likelihood Maximisation: A Unifying Framework for Information Theoretic Feature Selection. *J. Mach. Learning Res.* **2012**, *13*, 27–66.
13. Lewis, D.D. Feature Selection and Feature Extraction for Text Categorization. In Proceedings of the workshop on Speech and Natural Language, New York, NY, USA, 23–26 February 1992; pp. 212–217.
14. Press, W. Numerical Recipes in FORTRAN: The Art of Scientific Computing. Available online: https://doi.org/10.5860/choice.30-5638a (accessed on 29 May 2019).
15. Battiti, R. Using mutual information for selecting features in supervised neural net learning. *IEEE Trans. Neural Netw.* **1994**, *5*, 537–550. [CrossRef] [PubMed]
16. Peng, H.; Long, F.; Ding, C. Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans. Pattern Anal. Mach. Intel.* **2005**, *27*, 1226–1238. [CrossRef] [PubMed]
17. Yang, H.H.; Moody, J. Data visualization and feature selection: New algorithms for non-gaussian data. *Advances in Neural Information Processing Systems*. 2000, pp. 687–693. Available online: https://www.researchgate.net/publication/2460722 (accessed on 29 May 2019).
18. Fleuret, F. Fast Binary Feature Selection with Conditional Mutual Information. *J. Mach. Learn. Res.* **2004**, *5*, 1531–1555.
19. Yu, L.; Liu, H. Efficient Feature Selection via Analysis of Relevance and Redundancy. *J. Mach. Learn. Res.* **2004**, *5*, 1205–1224.
20. Tesmer, M.; Estevez, P.A. AMIFS: Adaptive feature selection by using mutual information. In Proceedings of the 2004 IEEE International Joint Conference on Neural Networks, Budapest, Hungary, 25–29 July 2004; pp. 303–308.
21. Cheng, G.; Qin, Z.; Feng, C.; Wang, Y.; Li, F. Conditional Mutual Information-Based Feature Selection Analyzing for Synergy and Redundancy. *Etri J.* **2011**, *33*, 210–218. [CrossRef]
22. John, G.H.; Kohavi, R.; Pfleger, K. Irrelevant Features and the Subset Selection Problem. In Proceedings of the Machine Learning Proceedings, New Brunswick, NJ, USA, 10–13 July 1994; pp. 121–129.
23. Wang, G.; Lochovsky, F.H. Feature selection with conditional mutual information maximin in text categorization. In Proceedings of the 13th ACM international conference on Information and knowledge management, Washington, DC, USA, 8–13 November 2004; pp. 342–349.
24. Kira, K.; Rendell, L.A. A Practical Approach to Feature Selection. In Proceedings of the Ninth International Workshop on Machine Learning, Aberdeen, Scotland, 1–3 July 1992; pp. 249–256.
25. Hall, M.A. Correlation-based Feature Selection for Discrete and Numeric Class Machine Learning. In Proceedings of the Seventeenth International Conference on Machine Learning, Stanford, CA, USA, 29 June–2 July 2000; pp. 359–366.
26. UCI Repository of Machine Learning Databases. Available online: http://www.ics.uci.edu/~{}mlearn/MLRepository.html (accessed on 29 May 2019).
27. Liu, H.; Sun, J.; Liu, L.; Zhang, H. Feature selection with dynamic mutual information. *Pattern Recognit.* **2009**, *42*, 1330–1339. [CrossRef]