Neural Processing Letters (2020) 51:1237–1263
https://doi.org/10.1007/s11063-019-10144-3

A Feature Selection Algorithm Based on Equal Interval Division and Minimal-Redundancy–Maximal-Relevance

Xiangyuan Gu[1] · Jichang Guo[1] · Lijun Xiao[1] · Tao Ming[1] · Chongyi Li[2]

Published online: 4 November 2019

Abstract
Minimal-redundancy–maximal-relevance (mRMR) algorithm is a typical feature selection algorithm. To select the feature which has minimal redundancy with the selected features and maximal relevance with the class label, the objective function of mRMR subtracts the average value of mutual information between features from mutual information between features and the class label, and selects the feature with the maximum difference. However, the problem is that the feature with the maximum difference is not always the feature with minimal redundancy maximal relevance. To solve the problem, the objective function of mRMR is first analyzed and a constraint condition that determines whether the objective function can guarantee the effectiveness of the selected features is achieved. Then, for the case where the objective function is not accurate, an idea of equal interval division is proposed and combined with ranking to process the interval of mutual information between features and the class label, and that of the average value of mutual information between features. Finally, a feature selection algorithm based on equal interval division and minimal-redundancy–maximal-relevance (EID–mRMR) is proposed. To validate the performance of EID–mRMR, we compare it with several incremental feature selection algorithms based on mutual information and other feature selection algorithms. Experimental results demonstrate that the EID–mRMR algorithm can achieve better feature selection performance.

Keywords Minimal-redundancy–maximal-relevance · Equal interval division · Mutual information · Feature selection

1 Introduction

With the explosive growth of information, dimension of feature set increases and it can cause the curse of dimensionality. Therefore, it is necessary to reduce the dimension of feature set [1–3]. Dimensionality reduction methods involve feature extraction and feature selection [4]. Feature extraction is a way that transforms the original features into a new space and

✉ Jichang Guo
jcguo@tju.edu.cn

1 School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China

2 The Department of Computer Science, City University of Hong Kong, Hong Kong 999077, China

takes the transformed features as the final features, while feature selection selects a subset of the original features. Compared with feature extraction, feature selection has advantages in the interpretation of data [5]. Therefore, feature selection has a wide range of applications, such as text processing [6,7], underwater objects recognition and classification [8,9],network anomaly detecting [10], information retrieval [11], image classification [12,13] and microarray data classification [14].

The metrics adopted in feature selection include distance, mutual information and consistency. Compared with other metrics, mutual information can measure the relationship between variables and it has the invariance under space transformations [15]. Hence, many feature selection algorithms based on mutual information are proposed, such as [16,17]. Among these algorithms, mutual information maximisation (MIM) algorithm [18] is a basic algorithm. However, it does not perform well due to only considering mutual information between features and the class label.

To overcome the shortcoming of MIM, some algorithms that employ mutual information between features and the class label to describe relevance and adopt mutual information between features to describe redundancy are proposed. Among them, minimal-redundancy–maximal-relevance (mRMR) algorithm [19] is a typical algorithm. In order to select the feature that has minimal redundancy with the selected features and maximal relevance with the class label, the average value of mutual information between each candidate feature and all the selected features is subtracted from mutual information between each candidate feature and the class label, and the feature with the maximum difference is selected. Since the feature with the maximum difference does not mean that the feature has minimal redundancy maximal relevance, the objective function of mRMR has a limitation.

Aiming at solving the existing problems of mRMR, some feature selection algorithms have been proposed. Since mRMR had the problem that mutual information biases toward multivalued features, normalization operation was used. Ultimately, NMIFS algorithm was proposed in [15]. Mutual information between each candidate feature and the class label, and the average value of mutual information between each candidate feature and all the selected features were processed by an optimization algorithm known as NSGA-II. Finally, MIFS-ND algorithm was presented in [20]. Combining mRMR with the idea of optimization, feature selection was investigated in [21]. mRMR was combined with ReliefF algorithm, and a two-stage feature selection algorithm was proposed in [22]. Combining mRMR with a particle swarm optimization algorithm, a maximum relevance minimum redundancy PSO algorithm was presented in [23]. In [15,20–23], the aforementioned limitation of the objective function of mRMR has not been handled properly.

In view of the problem that the objective function of mRMR has a limitation, this paper first analyzes the objective function of mRMR and achieves a condition that the objective function can guarantee the effectiveness of selected features. Then, for the case where the objective function cannot guarantee the effectiveness of selected features, the interval of mutual information between each candidate feature and the class label, and that of the average value of mutual information between each candidate feature and all the selected features are divided equally, and then the subintervals are ranked. Finally, a feature selection algorithm based on equal interval division and minimal-redundancy–maximal-relevance (EID–mRMR) is proposed.

The rest of this paper is organized as follows. Section 2 analyzes some feature selection algorithms based on mutual information. The EID–mRMR algorithm is proposed in Sect. 3. Section 4 presents and discusses experimental results. Conclusions and future work are presented in Sect. 5.

2 Related Work

In this paper, we only analyze mutual information of discrete random variables. Assuming Y and Z are two discrete random variables, $p(y)$ is the probability density function of Y, $p(z)$ is the probability density function of Z, and $p(y, z)$ is the joint probability density function of Y and Z. Mutual information is utilized to quantify the information that two random variables share. Mutual information $I(Y; Z)$ can be defined as

$$I(Y; Z) = \sum_{y \in Y} \sum_{z \in Z} p(y, z) \log \frac{p(y, z)}{p(y)p(z)}. \tag{1}$$

The higher mutual information values means that the two random variables share more information.

MIM is a feature selection algorithm based on mutual information, and its objective function is expressed as

$$MIM = \arg\max_{f_i \in X} [I(c; f_i)] \tag{2}$$

where X is the candidate features set, f_i is a candidate feature and c is the class label. MIM calculates mutual information between each candidate feature and the class label. Then, it ranks features in descending order according to the values, and selects some features with larger values. The algorithm does not yield good results due to ignoring feature interactions.

To overcome the shortcoming of MIM, some feature selection algorithms based on relevance and redundancy are proposed [19,24]. Objective functions of these algorithms are different, while their feature selection processes are same. The process is presented as follows. It first calculates mutual information between features and the class label, and selects the feature that has the maximum value. Then, it loops to select the feature that complies with the objective function in a forward search way. The loop ends when a specified number of features are selected. Obviously, objective functions are the key of these algorithms. Combined with the objective functions, these algorithms are analyzed.

$$MIFS = \arg\max_{f_i \in X} \left[I(c; f_i) - \beta \sum_{f_s \in S} I(f_s; f_i) \right]. \tag{3}$$

Equation (3) is the objective function of mutual information based feature selection (MIFS) algorithm [24]. S is the selected feature set and f_s is a selected feature. MIFS uses a parameter β to adjust mutual information $I(c; f_i)$ and mutual information between all the selected features and f_i. When β is set to zero, this algorithm is MIM.

mRMR [19] uses the reciprocal of the number of selected features to replace the parameter β, solving the problem of uncertain parameter. For selecting the feature that has minimal redundancy with the selected features and maximal relevance with the class label, mRMR subtracts the average value of mutual information between all the selected features and f_i from $I(c; f_i)$, and selects the feature with the maximum difference. The objective function of mRMR is expressed as

$$mRMR = \arg\max_{f_i \in X} \left[I(c; f_i) - \frac{1}{|S|} \sum_{f_s \in S} I(f_s; f_i) \right] \tag{4}$$

where $|S|$ is the number of selected features. However, the feature satisfying Eq. (4) is not always the feature with minimal redundancy maximal relevance. Therefore, the objective function of mRMR has a limitation.

$$MIFS\text{-}ND = \arg\max_{f_i \in X} [C_d - F_d]. \tag{5}$$

Combining mRMR with an optimization algorithm NSGA-II, MIFS-ND algorithm [20] was proposed. MIFS-ND first selects the feature that has the maximum mutual information value with the class label. Then, it calculates $I(c; f_i)$ and the average value of mutual information between all the selected features and each candidate feature. Following that, it processes them by NSGA-II and achieves the domination count C_d and the dominated count F_d for each feature. As shown in [20], the domination count of a candidate feature represents the number of features that it dominates for mutual information between the candidate feature and the class label. The dominated count of a candidate feature represents the number of features that it dominates for the average value of mutual information between the candidate feature and all the selected features. Finally, the feature satisfying Eq. (5) is selected. Following the above steps, it loops to select features until a specified number of features are selected. Compared with the range of $I(c; f_i)$ and that of the average value of mutual information between all the selected features and f_i, the range of C_d and that of F_d are greater. However, since C_d and F_d are not correlated to the difference between mutual information between the class label and different candidate features, and the difference between the average values of mutual information between the selected features and different candidate features, MIFS-ND cannot effectively handle the problem that the limitation existed in the objective function of mRMR.

3 The Proposed Feature Selection Algorithm

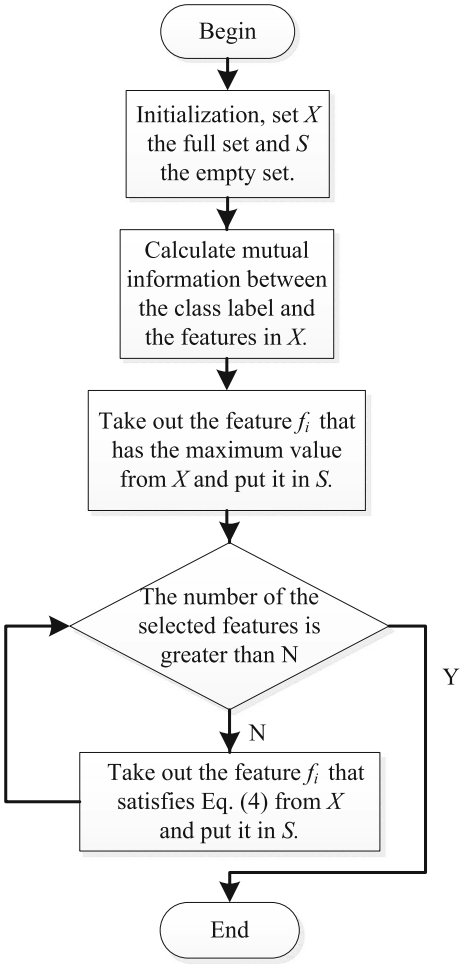
This section first achieves a condition that tests whether the objective function can guarantee the performance of selected features. Then, equal interval division is proposed and combined with ranking to deal with the case where the objective function cannot guarantee the performance of selected features. Following that, the proposed algorithm EID–mRMR is presented. Finally, an example is adopted to analyze the first two features selected by mRMR, MIFS-ND and EID–mRMR.

3.1 A Validation Condition

mRMR is a feature selection algorithm based on relevance and redundancy. Flow chart of mRMR is presented in Fig. 1. In the flow chart, the candidate feature set X and the selected feature set S are initialized. Then, mutual information between features and the class label is calculated, and the feature with the maximum value is selected; it loops to select the feature that complies with Eq. (4) in a forward search way until a specified number of features N are selected.

The objective function presented in Eq. (4) is the key of mRMR. mRMR utilizes $I(c; f_i)$ to describe relevance and adopts $I(f_s; f_i)$ to describe redundancy. In order to select the feature that has minimal redundancy with the selected features and maximal relevance with the class label, mRMR subtracts the average value of mutual information between all the selected features and each candidate feature from $I(c; f_i)$, and selects the feature with the maximum difference. The feature that has minimal redundancy with the selected features

Fig. 1 Flow chart of mRMR



and maximal relevance with the class label can satisfy Eq. (4), not vice versa. Therefore, Eq. (4) has a limitation. The feature selected by Eq. (4) is analyzed. It is necessary to present Eq. (6) at first.

$$J(f_i) = I(c; f_i) - \frac{1}{|S|} \sum_{f_s \in S} I(f_s; f_i) \tag{6}$$

Equation (4) is a special case of Eq. (6) attaining the maximum value. Calculate Eq. (6), if the maximum value is far greater than the secondary maximum value, and then select the feature with the maximum value. If not, the advantage of using Eq. (4) to select features is not obvious. To simplify the calculation, a condition that the difference between the maximum value and the secondary maximum value of Eq. (6) is greater than a fixed value P is adopted to test whether using Eq. (4) to select features. If the difference between the maximum value and the secondary maximum value is greater than P, Eq. (4) is used to select features; otherwise, an idea of equal interval division and ranking is adopted to select features. We will give a detailed description of the idea of equal interval division and ranking in the next section.

3.2 Equal Interval Division and Ranking

For the situation where Eq. (4) cannot guarantee the effectiveness of selected features, to guarantee the feature with minimal redundancy maximal relevance being selected, the interval of $I(c; f_i)$ and that of the average value of mutual information between all the selected features and f_i are divided equally, and then the subintervals are ranked. The number of dataset's features is taken as the number of subintervals. The concrete practices are presented as follows: determine the maximum value and the minimum value of $I(c; f_i)$ and the average value of mutual information between all the selected features and f_i as interval values, and then the interval values are divided equally. Following that, the subintervals are numbered from 1 to the number of dataset's features, and the numbers are taken as the ordinal values of the values in the subintervals.

FFMI $(f_s; f_i)$ is the value that the interval of $\sum_{f_s \in S} I(f_s; f_i)/|S|$ is processed by equal division and ranking. CFMI $(c; f_i)$ is the value that the interval of $I(c; f_i)$ is processed by equal division and ranking. The process of computing FFMI $(f_s; f_i)$ is shown in Algorithm 1 and that of computing CFMI $(c; f_i)$ is shown in Algorithm 2.

Algorithm 1 Compute FFMI $(f_s; f_i)$;

Input: M : the number of dataset's features, $|S|$: the number of features in S, $\sum_{f_s \in S} I(f_s; f_i)/|S|$.
Output: FFMI $(f_s; f_i)$.
1: a = min [$\sum_{f_s \in S} I(f_s; f_i)/|S|$], b = max [$\sum_{f_s \in S} I(f_s; f_i)/|S|$];
2: c = (b-a) / (M-1);
3: **for** $f_i \in X$ **do**
4: **for** j = 1: M **do**
5: **if** $\sum_{f_s \in S} I(f_s; f_i)/|S| \geq$ a - c + j*c **then**
6: **if** $\sum_{f_s \in S} I(f_s; f_i)/|S| <$ a + j*c **then**
7: FFMI $(f_s; f_i) = j$;
8: **end if**
9: **end if**
10: **end for**
11: **end for**

Algorithm 2 Compute CFMI $(c; f_i)$;

Input: M : the number of dataset's features, $I(c; f_i)$.
Output: CFMI $(c; f_i)$.
1: a = min [$I(c; f_i)$], b = max [$I(c; f_i)$];
2: c = (b-a) / (M-1);
3: **for** $f_i \in X$ **do**
4: **for** j = 1: M **do**
5: **if** $I(c; f_i) \geq$ a - c + j*c **then**
6: **if** $I(c; f_i) <$ a + j*c **then**
7: CFMI $(c; f_i) = j$;
8: **end if**
9: **end if**
10: **end for**
11: **end for**

For the situation where Eq. (4) cannot guarantee the effectiveness of selected features, equal interval division and ranking are adopted to process $I(c; f_i)$ and $\sum_{f_s \in S} I(f_s; f_i)/|S|$, and CFMI $(c; f_i)$ and FFMI $(f_s; f_i)$ are attained. Then, Eq. (7) is presented.

$$EID{-}mRMR = \arg\max_{f_i \in X} [\text{CFMI}(c; f_i) - \text{FFMI}(f_s; f_i)] \tag{7}$$

Equation (7) is the objective function of the proposed algorithm EID–mRMR. CFMI $(c; f_i)$ is utilized to describe relevance and FFMI $(f_s; f_i)$ is adopted to describe redundancy. If there are some features satisfying Eq. (7), the feature that maximizes Eq. (6) is selected from these features satisfying Eq. (7).

Equations (4), (5) and (7) are all proposed to select the features with minimal redundancy maximal relevance. Therefore, it is necessary to compare Eq. (7) with Eqs. (4) and (5). Before comparisons, Eqs. (8) and (9) are presented.

$$J(f_i) = C_d - F_d \tag{8}$$

$$J(f_i) = \text{CFMI}(c; f_i) - \text{FFMI}(f_s; f_i) \tag{9}$$

Equation (5) is a special case of Eq. (8) attaining the maximum value and Eq. (7) is that of Eq. (9) attaining the maximum value. Equation (9) is compared with Eqs. (6) and (8). The first part of Eqs. (6), (8) and (9) is adopted to describe relevance, and the second part is adopted to describe redundancy. With different candidate features, the range of C_d and that of F_d are related to the number of candidate features, and the range of CFMI $(c; f_i)$ and that of FFMI $(f_s; f_i)$ are related to the number of dataset's features. While the range of $I(c; f_i)$ and that of the average value of mutual information between all the selected features and f_i are not related to the number of candidate features and that of dataset's features, and they are smaller than the range of C_d, that of F_d, that of CFMI $(c; f_i)$ and that of FFMI $(f_s; f_i)$. Further, since the number of dataset's features is greater than that of candidate features, the range of C_d and that of F_d are smaller than the range of CFMI $(c; f_i)$ and that of FFMI $(f_s; f_i)$. Therefore, Eq. (9) has greater range of relevance and range of redundancy than Eqs. (6) and (8), and Eq. (8) has greater range of relevance and range of redundancy than Eq. (6).

Further, different from Eq. (8), since Eq. (9) adopts equal interval division and ranking to process $I(c; f_i)$ and the average value of mutual information between all the selected features and f_i, CFMI $(c; f_i)$ and FFMI $(f_s; f_i)$ can guarantee the values in the same subinterval have the same priority. Therefore, compared with Eqs. (4) and (5), Eq. (7) has more advantages in selecting the feature with minimal redundancy maximal relevance.

3.3 Algorithmic Implementation

With the above validation condition and Eq. (7), EID–mRMR is shown in Algorithm 3.

EID–mRMR consists of two parts: in the first part (lines 1–7), the candidate feature set X and the selected feature set S are initialized. Then, $I(c; f_i)$ is calculated, and the feature with the maximum value is selected; in the second part (lines 8–34), the average value of mutual information between all the selected features and f_i is calculated. Then, the difference between the maximum and the secondary maximum of Eq. (6) is calculated, if the difference is greater than a fixed value P, select the feature that satisfies Eq. (4); otherwise, calculate CFMI $(c; f_i)$, FFMI $(f_s; f_i)$ and the difference between CFMI $(c; f_i)$ and FFMI $(f_s; f_i)$. Following that, test whether the number of features with the maximum difference between CFMI $(c; f_i)$ and FFMI $(f_s; f_i)$ is more than 1, if it is no less than 2, select the feature that maximizes Eq. (6) from these features; otherwise, select the feature with the maximum

Algorithm 3 EID–mRMR: a feature selection algorithm based on equal interval division and minimal-redundancy–maximal-relevance;

Input: M: the number of dataset's features, Q: the number of features to be selected, P: a fixed value.
Output: S: the selected features.

initialize $S = \emptyset$ and $X = \{f_1, f_2, \ldots, f_M\}$;
for $f_i \in X$ **do**
  compute mutual information $I(c; f_i)$;
end for
find the feature f_k by maximizing mutual information with c, $f_k \in X$;
$S = S \cup \{f_k\}$;
$X = X - \{f_k\}$;
while $|S| \leq$ Q **do**
  for $f_i \in X$ **do**
    for $f_s \in S$ **do**
      compute mutual information $I(f_s; f_i)$;
    end for
    compute $\sum_{f_s \in S} I(f_s; f_i)/|S|$;
  end for
  compute Eq.(6);
  if the difference between the maximum and the secondary maximum of Eq.(6) is greater than P **then**;
    find the feature f_l that maximizes Eq.(6), $f_l \in X$;
    $S = S \cup \{f_l\}$;
    $X = X - \{f_l\}$;
  else
    compute FFMI $(f_s; f_i)$;
    compute CFMI $(c; f_i)$;
    find the number of features that satisfy Eq.(7);
    if the number of features is more than 1 **then**
      find the feature f_m that maximizes Eq.(6) from the features satisfying the above condition, $f_m \in X$;
      $S = S \cup \{f_m\}$;
      $X = X - \{f_m\}$;
    else
      find the feature f_n that satisfies Eq.(7), $f_n \in X$;
      $S = S \cup \{f_n\}$;
      $X = X - \{f_n\}$;
    end if
  end if
end while

difference between CFMI $(c; f_i)$ and FFMI $(f_s; f_i)$. Following the above steps, it loops to select features. The loop ends when a specified number of features are selected.

3.4 An Example

For better understanding the idea of EID–mRMR, an example is presented. Since the first selected feature is the maximum mutual information value with the class label and it is different from the other selected features. The second selected feature and other selected features satisfy the objective function of the algorithm. Therefore, an example is adopted to analyze the first two features selected by mRMR, MIFS-ND and EID–mRMR. A dataset with 7 features is used, and the feature f_7 has the maximum mutual information value with c. Dataset description is presented in Table 1.

Since the feature f_7 has the maximum mutual information value with c, mRMR, EID–mRMR and MIFS-ND first select f_7. Since f_7 is selected and the number of selected feature

Table 1 Dataset description

Features f_i	Mutual information $I(c; f_i)$	Mutual information $I(f_7; f_i)$	$I(c; f_i)$ - $I(f_7; f_i)$
f_1	0.90	0.09	0.81
f_2	0.89	0.07	0.82
f_3	0.86	0.06	0.80
f_4	0.78	0.05	0.73
f_5	0.66	0.03	0.63
f_6	0.60	0.04	0.56

Table 2 Domination count of a feature

Features f_i	Domination count C_d	Dominated count F_d	C_d–F_d
f_1	5	5	0
f_2	4	4	0
f_3	3	3	0
f_4	2	2	0
f_5	1	0	1
f_6	0	1	− 1

is 1, the average value of mutual information between all the selected features and the candidate feature f_i is $I(f_7; f_i)$. The range of $I(c; f_i)$ is [0.60, 0.90] and that of $I(f_7; f_i)$ is [0.03, 0.09]. Considering that the feature f_2 has the maximum difference between mutual information, mRMR selects f_2.

MIFS-ND adopts NSGA-II to process mutual information $I(c; f_i)$ and $I(f_7; f_i)$, and achieves the domination count C_d and the dominated count F_d. F_d is subtracted from C_d and the difference is attained. Domination count of a feature is shown in Table 2.

As shown in Table 2, because f_5 has the maximum difference, MIFS-ND selects f_5. Since the number of the candidate features is 6, the range of C_d and that of F_d are both [0, 5]. Therefore, compared with the range of $I(c; f_i)$ and that of $I(f_7; f_i)$, the range of C_d and that of F_d are greater. Further, although $I(c; f_2)$ is not far greater than $I(c; f_1)$, f_1 and f_2 have different C_d values. We can know that C_d and F_d are not correlated to the difference between mutual information between the class label and different candidate features, and the difference between the average values of mutual information between all the selected features and different candidate features.

f_2 has the maximum difference between $I(c; f_2)$ and $I(f_7; f_2)$. f_1 has the secondary maximum difference between $I(c; f_1)$ and $I(f_7; f_1)$. Considering that the maximum difference is not far greater than the secondary maximum difference, the objective function of mRMR cannot guarantee the effectiveness of the selected features. We adopt equal interval division and ranking to process $I(c; f_i)$ and $I(f_7; f_i)$. Since the dataset has 7 features, with the idea of equal interval division, the interval of $I(c; f_i)$ is divided into [0.60, 0.65), [0.65, 0.70), [0.70, 0.75), [0.75, 0.80), [0.80, 0.85), [0.85, 0.90), [0.90, 0.95) and that of $I(f_7; f_i)$ is divided into [0.03, 0.04), [0.04, 0.05), [0.05, 0.06), [0.06, 0.07), [0.07, 0.08), [0.08, 0.09), [0.09, 0.10). The ordinal values CFMI $(c; f_i)$ and FFMI $(f_s; f_i)$ are calculated. Ordinal value of a feature is shown in Table 3.

Table 3 Ordinal value of a feature

Features f_i	Ordinal value CFMI $(c; f_i)$	Ordinal value FFMI $(f_7; f_i)$	CFMI $(c; f_i)$– FFMI $(f_7; f_i)$
f_1	7	7	0
f_2	6	5	1
f_3	6	4	2
f_4	4	3	1
f_5	2	1	1
f_6	1	2	− 1

In order to understand EID–mRMR, $I(c; f_1)$ and $I(f_7; f_1)$ are adopted to analyze the ordinal values with EID–mRMR. Since $I(c; f_1)$ is 0.90, the value is in the subinterval of [0.90, 0.95) and the number is 7, hence the ordinal value of $I(c; f_1)$ is 7. Considering that $I(f_7; f_1)$ is 0.09, the value is in the subinterval of [0.09, 0.10) and the number is 7, hence the ordinal value of $I(f_7; f_1)$ is 7.

As shown in Table 3, since f_3 has the maximum difference between the ordinal values, EID–mRMR selects f_3 as the second selected feature. Since the dataset has 7 features, the range of CFMI $(c; f_i)$ and that of FFMI $(f_s; f_i)$ are both [1, 7]. Compared with the range of $I(c; f_i)$, that of $I(f_7; f_i)$, that of C_d and that of F_d, the range of CFMI $(c; f_i)$ and that of FFMI $(f_s; f_i)$ are greater. Therefore, EID–mRMR has greater range of relevance and range of redundancy than mRMR and MIFS-ND. Further, CFMI $(c; f_i)$ and FFMI $(f_s; f_i)$ can guarantee the features in the same subinterval have the same priority. Therefore, EID–mRMR has more advantages in selecting the feature with minimal redundancy maximal relevance. Compared with f_2 and f_5, it is more appropriate to select f_3.

4 Experimental Results

To validate the effectiveness of EID–mRMR, mRMR and MIFS-ND, other five incremental MI-based feature selection algorithms and other four feature selection algorithms are adopted for performance comparisons.

4.1 The Datasets and Experimental Settings

The datasets presented in Table 4 are from UCI machine learning repository [25] and ASU feature selection datasets [26]. The number of selected features is 50 for all the datasets. The minimum description length discretization method is adopted to transform the numerical features into discrete ones [27] and it is only used for feature selection. Three popular classifiers, J48, IB1 and Naive Bayes are utilized. The classifiers' parameters are set to WEKA's [28] default values. ASU feature selection software package [29] is adopted.

To validate the effectiveness of EID–mRMR, MIFS, mRMR, MIFS-ND, NMIFS [15], MIFS-U [30], MIFS-CR [31] and CMI [32] are adopted for performance comparisons. NMIFS, MIFS-U, MIFS-CR and CMI are four feature selection algorithms based on relevance and redundancy, and their objective functions are presented as follows:

$$NMIFS = \arg\max_{f_i \in X} \left[I(c; f_i) - \frac{1}{|S|} \sum_{f_s \in S} \frac{I(f_s; f_i)}{\min\{H(f_i), H(f_s)\}} \right] \tag{10}$$

Table 4 Summary of datasets in the experiment

Datasets	Instances	Features	Classes	Source
Spambase	4601	57	2	UCI
Synthetic_control	600	60	6	UCI
Mfeat_fou	2000	76	10	UCI
Movement_libras	360	90	15	UCI
Musk	476	166	2	UCI
Mfeat_fac	2000	216	10	UCI
Mfeat_pix	2000	240	10	UCI
Semeion	1593	256	10	UCI
ORL	400	1024	40	ASU
COIL20	1440	1024	20	ASU
gisette	7000	5000	2	ASU
orlraws10P	100	10,304	10	ASU

$$MIFS\text{-}U = \arg\max_{f_i \in X} \left[I(c; f_i) - \beta \sum_{f_s \in S} \frac{I(c; f_s)}{H(f_s)} I(f_s; f_i) \right] \tag{11}$$

$$MIFS\text{-}CR = \arg\max_{f_i \in X} \left\{ I(c; f_i) - \frac{1}{2} \sum_{f_s \in S} \left[\frac{I(c; f_s)}{H(f_s)} + \frac{I(c; f_i)}{H(f_i)} \right] I(f_s; f_i) \right\} \tag{12}$$

$$CMI = \arg\max_{f_i \in X} \left[I(c; f_i) - \frac{H(f_i/c)}{H(f_i)} \sum_{f_s \in S} \frac{I(c; f_s) I(f_s; f_i)}{H(f_s) H(c)} \right] \tag{13}$$

where $H(f_s)$ is the entropy of f_s, $H(f_i)$ is the entropy of f_i, $H(c)$ is the entropy of c, $H(f_i/c)$ is the conditional entropy.

In the experiment, the β value of MIFS is set to 0.5 and that of MIFS-U is set to 1.

Except for seven incremental MI-based algorithms, Relief-F [33], Fisher [34], QPFS [35] and SPEC-CMI [36] are compared with EID–mRMR. In QPFS and SPEC-CMI, before feature selection, all the features are normalized to the range $[-1, 1]$ and five equal-size bins is adopted to transform the numerical features into discrete ones.

As shown in [37], for reducing the influence of randomness on the final results, ten times of tenfold cross-validation are used, and the mean value and standard deviation of ten results are taken as the final results. To determine whether the effectiveness of experimental results is significant, a one-sided paired t-test at 5% significance level is carried out.

4.2 Experimental Results and Analysis

4.2.1 With Different *P* Values

In this section, the influence of P value on EID–mRMR is analyzed. The P value is set to 0.02, 0.03, 0.04, 0.05 and 0.06 respectively. The average performance of EID–mRMR with different P values when using J48, IB1 and Naive Bayes is presented in Table 5.

As shown in Table 5, when the P value is 0.05 or 0.06, it achieves the greatest Avg. value. At the same time, with different P values, the differences between the Avg. values

Table 5 Average performance of EID–mRMR with different P values

Datasets	0.02	0.03	0.04	0.05	0.06
Spambase	86.97	86.91	86.83	86.49	86.10
Synthetic_control	92.28	92.28	92.26	92.26	92.26
Mfeat_fou	77.14	77.10	77.06	77.06	77.02
Movement_libras	66.59	66.83	66.89	66.89	66.91
Musk	80.79	80.80	80.80	80.80	80.80
Mfeat_fac	89.64	89.66	89.73	89.70	89.71
Mfeat_pix	80.19	80.33	80.48	80.59	80.59
Semeion	70.28	70.97	71.28	71.61	71.77
ORL	68.12	68.43	68.57	68.68	68.78
COIL20	93.33	93.28	93.32	93.37	93.38
gisette	91.10	91.19	91.26	91.31	91.31
orlraws10P	87.96	87.84	87.93	88.03	88.12
Avg.	82.04	82.13	82.20	82.23	82.23

are very small. Therefore, the P value has a relatively small impact on the performance of EID–mRMR. In the experiment, we find that it takes more time for feature selection with the P value increasing. Therefore, in comparison with the incremental MI-based Algorithms and other feature selection algorithms, the P value of EID–mRMR is set to 0.05.

4.2.2 Comparison with Incremental MI-Based Algorithms

The section compares EID–mRMR with other incremental MI-based algorithms. Margins between the performance of EID–mRMR and other algorithms when using J48, IB1 and Naive Bayes are presented in Tables 6, 7 and 8. The values in the Avg. row are the mean value and standard deviation of the above twelve values. The values in the Win/Tie/Loss (W/T/L) row are the results obtained with one-sided paired t-test, among the values, the first value is the number of datasets that EID–mRMR is significantly superior to the other algorithms, the second value is the number of datasets that EID–mRMR performs equally with the other algorithms and the third value is the number of datasets that EID–mRMR is significantly inferior to the other algorithms. Classification accuracy of the optimal features selected by EID–mRMR and other algorithms when using J48, IB1 and Naive Bayes are presented in Tables 9, 10 and 11. The running time of EID–mRMR and other algorithms that select 50 features is shown in Table 12. Average performance comparisons of algorithms with the three classifiers are shown in Fig. 2.

As shown in Table 6, in Movement_libras, Mfeat_pix, Semeion, COIL20 and orlraws10P, EID–mRMR is significantly better than mRMR. From the values in the Avg. and W/T/L row, EID–mRMR performs better than mRMR when using J48. Compared with other algorithms, MIFS-U, MIFS-ND and EID–mRMR achieve better results.

In Table 7, the Avg. values indicate that MIFS-U, MIFS-CR and EID–mRMR yield better results. The values in the W/T/L row suggest that mRMR, NMIFS, MIFS-U and EID–mRMR can obtain better feature selection performance. Different from Table 6, the differences between EID–mRMR and the other algorithms increase except for MIFS-U.

In Table 8, in the terms of the Avg. values, compared with other algorithms, MIFS-U, MIFS-CR and EID–mRMR achieve better results. For the W/T/L values, MIFS-U, CMI

Table 6 Margins (mean ± SD) (%) between the performance of EID–mRMR and other algorithms when using J48

Datasets	EID–mRMR	MIFS	MIFS-U	mRMR	NMIFS	CMI	MIFS-CR	MIFS-ND
Spambase	91.94 ± 0.07	91.03 ± 0.07	91.98 ± 0.11	92.00 ± 0.10	91.99 ± 0.08	91.76 ± 0.05	92.09 ± 0.12	90.39 ± 0.12
Synthetic_control	89.89 ± 0.51	89.37 ± 0.56	87.79 ± 0.72	88.78 ± 0.70	84.74 ± 0.32	87.31 ± 0.55	89.99 ± 0.53	90.48 ± 0.57
Mfeat_fou	74.93 ± 0.45	70.92 ± 0.40	73.67 ± 0.41	75.30 ± 0.50	75.23 ± 0.49	75.26 ± 0.56	73.31 ± 0.49	61.88 ± 0.39
Movement_libras	63.74 ± 0.82	62.04 ± 1.78	61.46 ± 1.47	59.87 ± 1.33	57.90 ± 1.07	59.88 ± 1.16	63.50 ± 1.17	63.06 ± 0.93
Musk	80.69 ± 0.69	78.87 ± 0.81	80.45 ± 0.35	80.65 ± 0.77	80.02 ± 0.83	80.36 ± 0.85	79.98 ± 0.57	80.19 ± 0.95
Mfeat_fac	85.96 ± 0.30	85.49 ± 0.51	85.80 ± 0.43	85.13 ± 0.34	83.57 ± 0.31	85.17 ± 0.34	85.44 ± 0.44	85.14 ± 0.20
Mfeat_pix	74.51 ± 0.45	72.94 ± 0.57	73.90 ± 0.41	73.05 ± 0.36	70.32 ± 0.41	71.25 ± 0.39	73.79 ± 0.53	72.98 ± 0.28
Semeion	69.89 ± 0.50	69.53 ± 0.36	69.31 ± 0.45	62.86 ± 0.19	63.74 ± 0.25	63.87 ± 0.22	69.70 ± 0.40	69.33 ± 0.43
ORL	51.65 ± 1.95	49.31 ± 2.19	51.18 ± 1.64	52.39 ± 1.19	50.21 ± 1.23	50.60 ± 1.36	50.19 ± 1.33	48.99 ± 1.24
COIL20	89.56 ± 0.36	88.01 ± 0.40	87.49 ± 0.73	88.53 ± 0.34	84.86 ± 0.30	87.48 ± 0.35	86.16 ± 0.55	88.59 ± 0.34
gisette	92.42 ± 0.07	91.18 ± 0.22	92.19 ± 0.07	92.06 ± 0.06	91.34 ± 0.10	92.35 ± 0.10	91.64 ± 0.13	89.36 ± 0.16
orlraws10P	78.08 ± 1.91	73.53 ± 2.84	72.43 ± 2.49	74.09 ± 2.74	70.45 ± 2.76	70.20 ± 2.75	73.01 ± 2.44	77.17 ± 2.61
Avg.	78.61 ± 12.54	76.85 ± 13.00	77.30 ± 12.72	77.06 ± 13.24	75.36 ± 13.22	76.29 ± 13.44	77.40 ± 12.67	76.46 ± 13.50
W/T/L	–	12/0/0	8/4/0	8/2/2	10/1/1	8/3/1	8/3/1	9/2/1

Table 7 Margins (mean ± SD) (%) between the performance of EID–mRMR and other algorithms when using IB1

Datasets	EID–mRMR	MIFS	MIFS-U	mRMR	NMIFS	CMI	MIFS-CR	MIFS-ND
Spambase	84.43 ± 0.15	78.48 ± 0.30	85.17 ± 0.18	84.74 ± 0.12	86.22 ± 0.13	83.78 ± 0.23	83.94 ± 0.12	81.72 ± 0.17
Synthetic_control	92.66 ± 0.28	88.05 ± 0.46	88.65 ± 0.53	90.17 ± 0.27	88.35 ± 0.38	89.75 ± 0.45	88.20 ± 0.44	92.32 ± 0.33
Mfeat_fou	79.62 ± 0.20	75.00 ± 0.26	77.29 ± 0.21	80.12 ± 0.20	80.14 ± 0.19	80.12 ± 0.22	77.91 ± 0.25	67.38 ± 0.39
Movement_libras	80.75 ± 1.03	80.11 ± 0.76	80.57 ± 0.78	76.65 ± 1.20	74.32 ± 1.09	75.47 ± 1.09	80.21 ± 0.90	79.82 ± 1.01
Musk	82.28 ± 0.74	81.91 ± 0.91	81.72 ± 0.76	81.95 ± 0.64	80.18 ± 0.61	82.25 ± 0.68	80.89 ± 0.52	82.29 ± 0.88
Mfeat_fac	92.26 ± 0.14	91.51 ± 0.19	91.93 ± 0.12	91.51 ± 0.10	90.04 ± 0.13	91.42 ± 0.08	91.80 ± 0.17	91.45 ± 0.13
Mfeat_pix	81.36 ± 0.29	79.80 ± 0.29	80.65 ± 0.24	73.72 ± 0.12	67.76 ± 0.20	73.69 ± 0.11	79.91 ± 0.28	80.97 ± 0.22
Semeion	71.20 ± 0.37	70.90 ± 0.23	70.91 ± 0.32	58.84 ± 0.32	60.32 ± 0.24	60.74 ± 0.23	71.02 ± 0.30	69.39 ± 0.44
ORL	81.54 ± 0.80	80.29 ± 0.87	82.02 ± 1.00	77.61 ± 0.87	75.21 ± 0.75	75.59 ± 0.59	80.28 ± 1.01	81.48 ± 0.68
COIL20	96.09 ± 0.09	95.21 ± 0.18	95.06 ± 0.20	95.45 ± 0.14	91.92 ± 0.14	93.26 ± 0.10	94.98 ± 0.12	95.52 ± 0.09
gisette	91.32 ± 0.24	87.89 ± 0.16	90.86 ± 0.12	91.21 ± 0.11	90.22 ± 0.12	91.00 ± 0.12	90.51 ± 0.10	86.56 ± 0.20
orlraws10P	92.60 ± 1.21	84.17 ± 1.53	87.19 ± 2.80	90.97 ± 0.80	88.92 ± 1.41	85.75 ± 1.50	84.78 ± 1.40	88.47 ± 1.66
Avg.	85.51 ± 7.38	82.78 ± 6.94	84.34 ± 6.83	82.75 ± 10.27	81.13 ± 10.06	81.90 ± 9.45	83.70 ± 6.75	83.11 ± 8.56
W/T/L	–	11/1/0	9/2/1	9/1/2	10/0/2	10/1/1	11/1/0	10/2/0

Table 8 Margins (mean ± SD) (%) between the performance of EID–mRMR and other algorithms when using Naive Bayes

Datasets	EID–mRMR	MIFS	MIFS-U	mRMR	NMIFS	CMI	MIFS-CR	MIFS-ND
Spambase	83.09 ± 0.25	69.43 ± 0.33	83.03 ± 0.33	85.26 ± 0.08	84.24 ± 0.05	77.80 ± 0.11	76.60 ± 0.17	74.24 ± 0.22
Synthetic_control	94.24 ± 0.20	93.91 ± 0.18	87.90 ± 0.56	89.29 ± 0.29	84.26 ± 0.32	89.89 ± 0.35	93.95 ± 0.29	93.08 ± 0.40
Mfeat_fou	76.64 ± 0.24	73.28 ± 0.10	75.74 ± 0.25	76.39 ± 0.17	76.36 ± 0.16	76.43 ± 0.18	75.82 ± 0.30	65.11 ± 0.28
Movement_libras	56.18 ± 1.49	53.60 ± 1.39	51.46 ± 0.90	50.55 ± 1.10	46.25 ± 1.00	49.47 ± 1.21	52.79 ± 1.28	55.42 ± 1.18
Musk	79.43 ± 0.53	76.96 ± 0.76	78.90 ± 0.61	79.51 ± 0.40	79.07 ± 0.54	79.68 ± 0.58	79.31 ± 0.76	79.13 ± 0.60
Mfeat_fac	90.90 ± 0.10	89.70 ± 0.10	90.43 ± 0.17	88.43 ± 0.11	85.19 ± 0.20	88.88 ± 0.10	90.06 ± 0.21	90.15 ± 0.16
Mfeat_pix	85.89 ± 0.12	85.75 ± 0.18	85.94 ± 0.21	79.07 ± 0.17	74.53 ± 0.20	78.40 ± 0.12	85.60 ± 0.14	85.53 ± 0.10
Semeion	73.76 ± 0.39	73.71 ± 0.28	73.46 ± 0.19	63.53 ± 0.10	64.42 ± 0.11	65.17 ± 0.11	73.20 ± 0.24	73.00 ± 0.25
ORL	72.86 ± 1.20	71.94 ± 0.99	72.08 ± 0.80	61.44 ± 0.72	54.88 ± 0.83	58.19 ± 0.69	67.73 ± 2.02	69.95 ± 0.85
COIL20	94.45 ± 0.13	91.99 ± 0.26	91.14 ± 0.36	92.16 ± 0.17	83.91 ± 0.26	89.28 ± 0.23	90.17 ± 0.37	92.95 ± 0.20
gisette	90.18 ± 0.09	90.45 ± 0.36	91.31 ± 0.09	88.26 ± 0.05	87.48 ± 0.05	91.30 ± 0.07	90.91 ± 0.05	82.31 ± 0.77
orlraws10P	93.40 ± 1.42	84.99 ± 2.34	82.87 ± 1.94	91.52 ± 1.25	79.43 ± 1.98	78.48 ± 1.72	83.59 ± 2.56	89.20 ± 1.54
Avg.	82.59 ± 11.47	79.64 ± 11.93	80.36 ± 11.34	78.78 ± 13.53	75.00 ± 13.11	76.91 ± 13.23	79.98 ± 11.79	79.17 ± 11.92
W/T/L	–	10/1/1	9/1/2	10/1/1	11/0/1	10/0/2	10/1/1	11/1/0

and EID–mRMR perform better than the other algorithms. Moreover, the advantages that EID–mRMR is superior to the other algorithms increase in both the Avg. and W/T/L values.

As shown in Tables 9, 10 and 11, we can see that EID–mRMR can achieve better results in the majority of datasets. Take Table 11 for example, the number of datasets that EID–mRMR performs better than the other seven algorithms is 7, while that of any of the other algorithms is superior to EID–mRMR is no more than 2. Furthermore, EID–mRMR can obtain the greatest Avg. values with all the three classifiers.

In Table 12, EID–mRMR takes a fair amount of time with the other seven algorithms in the majority of datasets except for gisette and orlraws10p. Since features of gisette and orlraws10p are high dimensional and they need to be processed by the method of equal interval division and ranking for more times, EID–mRMR is more time-consuming in these two datasets and it obtains greater Avg. values than other algorithms.

As shown in Fig. 2, in some datasets, such as Spambase and Mfeat_fou, the selected features account for a large proportion of these datasets. After classification accuracy of the features selected by the majority of algorithms reaches the maximum value, it decreases to some extent. While in most of datasets, due to selected features occupying a certain proportion of datasets, the average performance of the features selected by most of algorithms significantly increases in the beginning. Then, it increases slowly and varies gently after arriving at a certain extent. Considering that the average value of the 50 features selected by the algorithms is taken as the final result, the percentage of selected features accounting for the datasets has less impact on the final result. Therefore, it is suitable for selecting 50 features. In the average performance comparisons of algorithms, EID–mRMR can achieve better effectiveness in the majority of datasets. However, the other algorithms do not perform well in some datasets; for example, mRMR cannot handle well in Synthetic_control, Movement_libras, Mfeat_pix, Semeion, ORL and gisette. MIFS-ND does not yield good results in Spambase, Mfeat_fou and gisette. As a whole, compared with the other seven feature selection algorithms, EID–mRMR can achieve better results.

4.2.3 Comparison with Other Feature Selection Algorithms

The section compares EID–mRMR with Relief-F, Fisher, QPFS and SPEC-CMI. Margins between the performance of EID–mRMR and other algorithms when using J48, IB1 and Naive Bayes are presented in Tables 13, 14 and 15. Classification accuracy of the optimal features selected by EID–mRMR and other algorithms when using J48, IB1 and Naive Bayes are presented in Tables 16, 17 and 18.

As shown in Table 13, EID–mRMR and SPEC-CMI are superior to Relief-F, Fisher and QPFS in Spambase and Mfeat_fou. EID–mRMR and QPFS are superior to Relief-F, Fisher and SPEC-CMI in Mfeat_fou and Movement_libras. In most of datasets, EID–mRMR performs better than Relief-F, Fisher, QPFS and SPEC-CMI. From the values in the Avg. and W/T/L row, EID–mRMR can achieve better results than the other four algorithms.

In Table 14, SPEC-CMI performs better than EID–mRMR in Spambase and Mfeat_fou. In the majority of datasets, EID–mRMR yields better results than the other four algorithms. The values in the Avg. row and the W/T/L row suggest that EID–mRMR can obtain better feature selection effectiveness. Different from Table 13, the advantages that EID–mRMR is superior to the other algorithms increase in both the Avg. and W/T/L values.

As shown in Table 15, Fisher and SPEC-CMI achieve better results than EID–mRMR in Spambase. In most of datasets, EID–mRMR can achieve better feature selection performance than the other four algorithms. Different from Tables 13 and 14, the differences between EID–mRMR and the other four algorithms increase in the Avg. and W/T/L values.

Table 9 Classification accuracy (%) of the optimal features selected by EID–mRMR and other algorithms when using J48

Datasets	EID–mRMR	MIFS	MIFS-U	mRMR	NMIFS	CMI	MIFS-CR	MIFS-ND
Spambase	93.03 (49)	92.55 (50)	93.12 (23)	93.06 (49)	93.09 (17)	92.83 (34)	**93.18** (48)	92.95 (50)
Synthetic_control	92.87 (50)	92.85 (30)	93.17 (49)	93.55 (47)	93.15 (44)	92.37 (48)	92.88 (46)	**93.63** (37)
Mfeat_fou	77.09 (32)	73.87 (8)	76.53 (9)	**77.83** (11)	77.74 (10)	77.81 (11)	75.36 (8)	72.67 (48)
Movement_libras	**68.50** (29)	67.11 (45)	66.94 (50)	65.53 (44)	65.31 (46)	67.94 (42)	67.78 (41)	68.28 (34)
Musk	**83.41** (48)	80.63 (44)	83.26 (49)	83.13 (50)	83.36 (45)	82.97 (21)	82.57 (30)	82.68 (23)
Mfeat_fac	89.03 (43)	88.30 (22)	88.67 (22)	88.86 (43)	87.82 (50)	**89.43** (50)	88.08 (14)	88.67 (49)
Mfeat_pix	**78.81** (49)	76.36 (40)	77.55 (24)	78.27 (48)	77.82 (48)	78.26 (50)	77.65 (27)	77.38 (47)
Semeion	76.75 (50)	75.98 (46)	76.80 (50)	71.59 (49)	71.76 (48)	74.64 (50)	**77.23** (50)	76.17 (40)
ORL	55.95 (50)	52.27 (23)	55.23 (37)	57.30 (44)	56.78 (49)	**58.18** (49)	54.48 (50)	52.85 (37)
COIL20	**92.81** (46)	90.62 (50)	90.46 (49)	91.35 (42)	89.17 (50)	92.72 (50)	89.17 (43)	92.33 (45)
gisette	**94.21** (46)	91.95 (23)	93.28 (50)	93.44 (49)	93.24 (50)	93.78 (50)	93.23 (50)	90.82 (50)
orlraws10P	**81.40** (31)	75.20 (9)	75.60 (13)	76.90 (5)	72.70 (12)	74.80 (43)	74.30 (34)	80.00 (38)
Avg.	**81.99**	79.81	80.88	80.90	80.16	81.31	80.49	80.70

Bold indicates that the best results can be achieved

Table 10 Classification accuracy (%) of the optimal features selected by EID–mRMR and other algorithms when using IB1

Datasets	EID–mRMR	MIFS	MIFS-U	mRMR	NMIFS	CMI	MIFS-CR	MIFS-ND
Spambase	90.70 (50)	87.99 (50)	90.76 (48)	90.70 (50)	**90.84** (50)	89.02 (50)	89.01 (50)	90.57 (50)
Synthetic_control	97.58 (49)	96.07 (47)	97.40 (50)	**98.20** (50)	98.15 (50)	97.38 (48)	97.25 (50)	97.38 (46)
Mfeat_fou	83.15 (23)	78.34 (8)	81.82 (12)	84.05 (14)	84.16 (12)	**84.24** (11)	82.05 (14)	79.75 (50)
Movement_libras	85.03 (37)	84.31 (10)	**85.08** (12)	82.44 (50)	82.61 (49)	84.78 (44)	84.64 (15)	84.00 (28)
Musk	**86.18** (50)	85.28 (49)	84.77 (23)	85.29 (50)	83.82 (50)	85.31 (42)	84.26 (45)	85.74 (34)
Mfeat_fac	**96.48** (30)	95.74 (50)	95.98 (49)	96.24 (45)	94.87 (49)	96.38 (28)	95.76 (49)	96.37 (35)
Mfeat_pix	**93.94** (50)	91.32 (49)	92.64 (50)	89.35 (50)	88.17 (50)	92.56 (50)	91.61 (50)	93.82 (50)
Semeion	**86.40** (47)	85.53 (50)	86.06 (50)	77.18 (50)	77.89 (50)	82.45 (50)	86.23 (50)	85.44 (50)
ORL	**93.03** (49)	90.47 (29)	92.00 (50)	88.40 (49)	87.32 (50)	88.27 (50)	91.20 (50)	92.65 (50)
COIL20	99.87 (46)	99.06 (48)	99.05 (49)	99.43 (50)	97.91 (50)	98.61 (50)	98.90 (45)	**99.88** (50)
gisette	**94.32** (50)	90.49 (14)	93.30 (47)	93.49 (36)	92.70 (50)	93.87 (50)	93.36 (48)	89.22 (50)
orlraws10P	**97.20** (41)	88.10 (16)	93.10 (49)	94.90 (42)	93.10 (46)	93.40 (45)	91.20 (47)	93.90 (33)
Avg.	**91.99**	89.39	91.00	89.97	89.30	90.52	90.46	90.73

Bold indicates that the best results can be achieved

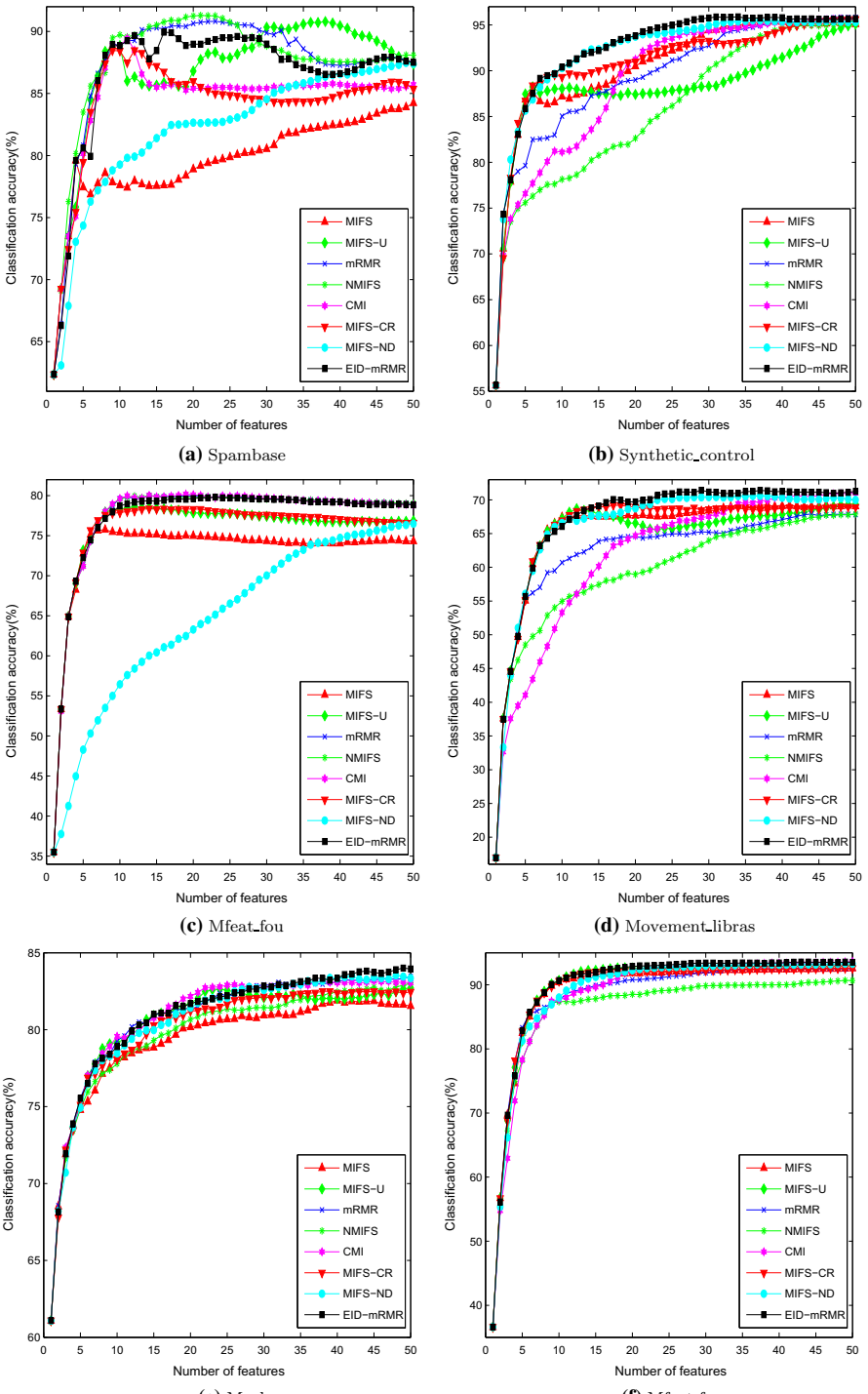
Table 11 Classification accuracy (%) of the optimal features selected by EID–mRMR and other algorithms when using Naive Bayes

Datasets	EID–mRMR	MIFS	MIFS-U	mRMR	NMIFS	CMI	MIFS-CR	MIFS-ND
Spambase	90.26 (12)	86.42 (4)	89.21 (9)	**91.29** (23)	90.98 (22)	89.78 (11)	88.95 (9)	79.04 (49)
Synthetic_control	97.98 (31)	**98.23** (34)	94.73 (50)	95.98 (46)	95.55 (50)	97.33 (45)	96.93 (28)	96.28 (36)
Mfeat_fou	**79.34** (21)	75.42 (16)	78.34 (17)	79.11 (25)	79.08 (25)	79.28 (20)	78.37 (16)	77.12 (50)
Movement_libras	**61.03** (37)	58.06 (35)	56.89 (12)	56.36 (44)	56.06 (50)	60.42 (50)	57.22 (15)	60.25 (36)
Musk	82.77 (50)	80.04 (40)	80.85 (22)	82.34 (41)	82.08 (50)	82.27 (47)	82.00 (36)	**82.98** (50)
Mfeat_fac	**95.48** (49)	94.12 (50)	94.58 (50)	92.84 (37)	89.37 (49)	95.24 (50)	94.13 (41)	95.08 (47)
Mfeat_pix	92.78 (50)	92.41 (50)	92.87 (46)	87.96 (50)	86.75 (50)	91.20 (50)	92.17 (35)	**93.03** (47)
Semeion	**83.97** (50)	83.69 (50)	83.93 (50)	74.55 (50)	75.22 (50)	80.34 (50)	83.45 (49)	83.10 (49)
ORL	**88.05** (50)	82.60 (31)	83.95 (42)	74.62 (49)	70.55 (50)	78.05 (50)	81.38 (48)	84.87 (50)
COIL20	**99.12** (50)	96.11 (50)	95.67 (46)	96.87 (50)	90.23 (50)	96.01 (50)	94.48 (49)	98.37 (50)
gisette	91.76 (50)	91.72 (12)	93.59 (45)	88.83 (33)	88.36 (44)	**93.92** (50)	93.61 (49)	88.46 (12)
orlraws10P	**98.20** (41)	87.70 (13)	87.50 (16)	95.90 (48)	83.50 (46)	86.10 (50)	89.80 (50)	95.70 (49)
Avg.	**88.40**	85.54	86.01	84.72	82.31	85.83	86.04	86.19

Bold indicates that the best results can be achieved

Table 12 Running time (s) of EID–mRMR and other algorithms

Datasets	EID–mRMR	MIFS	MIFS-U	mRMR	NMIFS	CMI	MIFS-CR	MIFS-ND
Spambase	0.67	0.63	0.64	0.65	0.67	0.74	0.71	0.66
Synthetic_control	0.19	0.16	0.16	0.18	0.19	0.20	0.19	0.18
Mfeat_fou	0.59	0.54	0.55	0.57	0.58	0.64	0.62	0.58
Movement_libras	0.21	0.16	0.16	0.20	0.21	0.23	0.21	0.20
Musk	0.38	0.26	0.25	0.35	0.37	0.41	0.38	0.35
Mfeat_fac	1.02	0.84	0.82	0.96	0.99	1.17	1.10	0.97
Mfeat_pix	0.65	0.42	0.42	0.56	0.59	0.82	0.72	0.56
Semeion	0.61	0.39	0.37	0.53	0.57	0.76	0.68	0.54
ORL	3.46	1.74	1.69	2.37	2.53	2.90	2.54	2.39
COIL20	6.85	4.92	4.87	5.62	5.80	6.61	6.03	5.66
gisette	65.54	38.35	37.93	42.22	43.21	57.91	54.34	42.20
orlraws10P	130.21	12.51	12.02	19.05	20.57	22.13	19.63	19.12
Avg.	210.38	60.92	59.88	73.26	76.28	94.52	87.15	73.41

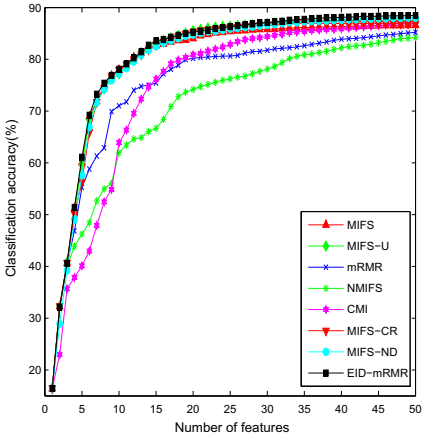


(a) Spambase **(b)** Synthetic_control

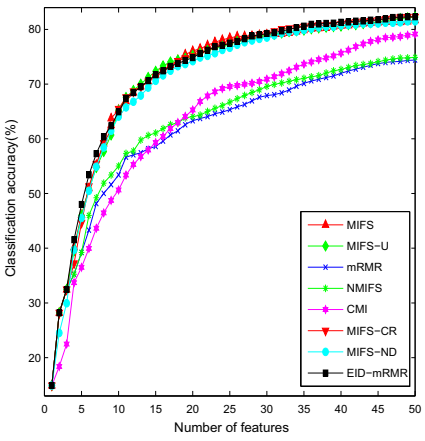
(c) Mfeat_fou **(d)** Movement_libras

(e) Musk **(f)** Mfeat_fac

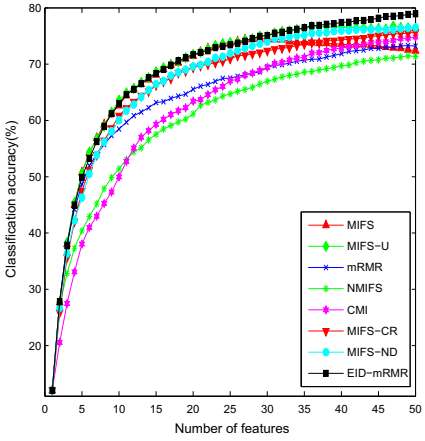
Fig. 2 Average performance comparisons of algorithms with the three classifiers



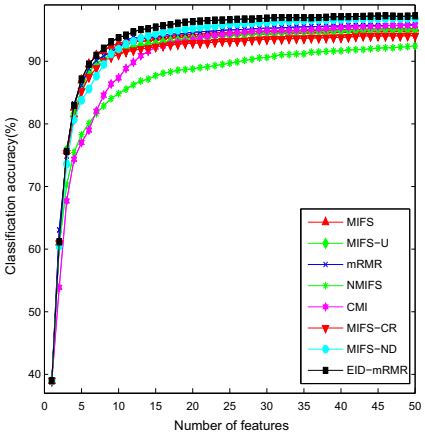
(g) Mfeat_pix



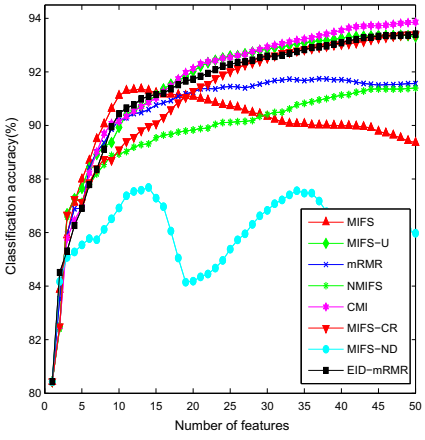
(h) Semeion



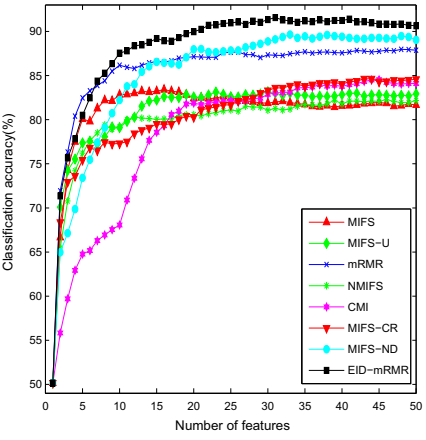
(i) ORL



(j) COIL20



(k) gisette



(l) orlraws10P

Fig. 2 continued

Table 13 Margins (mean ± SD) (%) between the performance of EID–mRMR and other algorithms when using J48

Datasets	EID–mRMR	Relief-F	Fisher	QPFS	SPEC-CMI
Spambase	91.94 ± 0.07	88.35 ± 0.11	91.22 ± 0.13	91.47 ± 0.11	91.98 ± 0.07
Synthetic_control	89.89 ± 0.51	81.66 ± 0.55	82.68 ± 0.35	89.38 ± 0.57	86.53 ± 0.76
Mfeat_fou	74.93 ± 0.45	75.07 ± 0.44	74.92 ± 0.42	75.10 ± 0.51	75.30 ± 0.47
Movement_libras	63.74 ± 0.82	49.76 ± 0.78	47.68 ± 0.61	63.58 ± 1.19	58.73 ± 1.34
Musk	80.69 ± 0.69	74.64 ± 0.59	75.56 ± 0.40	77.30 ± 0.67	78.27 ± 0.60
Mfeat_fac	85.96 ± 0.30	78.04 ± 0.22	81.60 ± 0.17	84.15 ± 0.10	82.72 ± 0.20
Mfeat_pix	74.51 ± 0.45	64.58 ± 0.27	63.07 ± 0.22	71.16 ± 0.43	65.56 ± 0.77
Semeion	69.89 ± 0.50	57.53 ± 0.17	56.52 ± 0.18	64.35 ± 0.23	57.26 ± 1.22
ORL	51.65 ± 1.95	46.93 ± 0.89	42.72 ± 0.90	43.81 ± 0.90	38.47 ± 0.65
COIL20	89.56 ± 0.36	69.62 ± 0.43	62.38 ± 0.28	87.86 ± 0.27	81.71 ± 0.37
gisette	92.42 ± 0.07	92.10 ± 0.09	90.26 ± 0.07	90.08 ± 0.06	90.27 ± 0.07
orlraws10P	78.08 ± 1.91	64.49 ± 3.15	53.74 ± 3.31	61.91 ± 2.12	47.09 ± 4.47
Avg.	78.61 ± 12.54	70.23 ± 14.24	68.53 ± 16.46	75.01 ± 14.66	71.16 ± 17.52
W/T/L	–	11/0/1	11/1/0	10/1/1	10/1/1

Table 14 Margins (mean ± SD) (%) between the performance of EID–mRMR and other algorithms when using IB1

Datasets	EID–mRMR	Relief-F	Fisher	QPFS	SPEC-CMI
Spambase	84.43 ± 0.15	83.84 ± 0.15	83.79 ± 0.15	82.40 ± 0.10	85.53 ± 0.10
Synthetic_control	92.66 ± 0.28	86.31 ± 0.19	86.79 ± 0.23	90.28 ± 0.23	87.67 ± 0.44
Mfeat_fou	79.62 ± 0.20	79.59 ± 0.23	79.64 ± 0.21	79.71 ± 0.22	80.09 ± 0.19
Movement_libras	80.75 ± 1.03	65.18 ± 0.65	58.49 ± 0.74	79.75 ± 0.85	75.36 ± 0.80
Musk	82.28 ± 0.74	77.05 ± 0.50	78.64 ± 0.48	79.58 ± 0.49	80.24 ± 0.41
Mfeat_fac	92.26 ± 0.14	81.34 ± 0.16	88.34 ± 0.11	90.93 ± 0.10	89.02 ± 0.13
Mfeat_pix	81.36 ± 0.29	61.09 ± 0.14	58.67 ± 0.22	74.87 ± 0.08	69.92 ± 0.61
Semeion	71.20 ± 0.37	54.22 ± 0.27	51.73 ± 0.27	60.81 ± 0.15	58.08 ± 0.91
ORL	81.54 ± 0.80	68.52 ± 0.51	67.46 ± 0.66	75.72 ± 1.07	61.54 ± 1.06
COIL20	96.09 ± 0.09	75.60 ± 0.20	65.30 ± 0.15	93.49 ± 0.11	91.26 ± 0.17
gisette	91.32 ± 0.24	91.32 ± 0.04	89.27 ± 0.08	88.91 ± 0.08	89.13 ± 0.07
orlraws10P	92.60 ± 1.21	74.97 ± 0.69	69.11 ± 2.54	70.43 ± 0.93	64.42 ± 2.35
Avg.	85.51 ± 7.38	74.92 ± 10.88	73.10 ± 12.99	80.57 ± 9.50	77.69 ± 11.69
W/T/L	–	10/2/0	11/1/0	11/0/1	10/0/2

In Tables 16, 17 and 18, we can see that EID–mRMR performs better than the other four algorithms in the majority of datasets. Take Table 16 for example, the number of datasets that EID–mRMR can obtain better results than the other four algorithms is 7, while that of datasets that EID–mRMR is inferior to any one of the other algorithms is no more than 5. Furthermore, EID–mRMR achieves the greatest Avg. values with all the three classifiers.

Table 15 Margins (mean ± SD) (%) between the performance of EID–mRMR and other algorithms when using Naive Bayes

Datasets	EID–mRMR	Relief-F	Fisher	QPFS	SPEC-CMI
Spambase	83.09 ± 0.25	72.08 ± 0.06	83.81 ± 0.07	79.66 ± 0.10	85.58 ± 0.09
Synthetic_control	94.24 ± 0.20	80.08 ± 0.33	81.58 ± 0.23	92.53 ± 0.27	89.50 ± 0.41
Mfeat_fou	76.64 ± 0.24	75.86 ± 0.16	75.86 ± 0.19	76.27 ± 0.17	76.18 ± 0.17
Movement_libras	56.18 ± 1.49	38.31 ± 0.60	34.52 ± 0.43	52.08 ± 1.71	42.38 ± 1.25
Musk	79.43 ± 0.53	73.71 ± 1.08	75.52 ± 0.76	74.61 ± 0.51	76.67 ± 0.47
Mfeat_fac	90.90 ± 0.10	75.33 ± 0.18	81.09 ± 0.18	88.72 ± 0.13	84.03 ± 0.18
Mfeat_pix	85.89 ± 0.12	67.27 ± 0.20	65.65 ± 0.19	79.80 ± 0.14	75.70 ± 0.54
Semeion	73.76 ± 0.39	57.23 ± 0.14	56.22 ± 0.08	66.61 ± 0.12	59.18 ± 0.69
ORL	72.86 ± 1.20	44.80 ± 0.97	36.71 ± 0.82	51.74 ± 0.71	34.80 ± 1.49
COIL20	94.45 ± 0.13	66.10 ± 0.19	60.71 ± 0.22	90.35 ± 0.15	81.49 ± 0.29
gisette	90.18 ± 0.09	85.22 ± 0.02	85.89 ± 0.02	85.67 ± 0.02	85.86 ± 0.03
orlraws10P	93.40 ± 1.42	68.06 ± 3.18	58.92 ± 3.26	70.56 ± 0.97	43.89 ± 3.11
Avg.	82.59 ± 11.47	67.00 ± 13.93	66.37 ± 17.56	75.72 ± 13.62	69.61 ± 19.35
W/T/L	–	12/0/0	11/0/1	12/0/0	11/0/1

Table 16 Classification accuracy (%) of the optimal features selected by EID–mRMR and other algorithms when using J48

Datasets	EID–mRMR	Relief-F	Fisher	QPFS	SPEC-CMI
Spambase	93.03 (49)	93.00 (47)	93.04 (30)	**93.13** (22)	92.93 (45)
Synthetic_control	92.87 (50)	93.32 (50)	91.60 (50)	93.33 (30)	**93.77** (45)
Mfeat_fou	77.09 (32)	78.11 (10)	**78.19** (11)	78.11 (16)	77.97 (10)
Movement_libras	68.50 (29)	64.00 (49)	59.56 (50)	**68.92** (28)	67.39 (50)
Musk	**83.41** (48)	79.54 (50)	81.19 (48)	80.36 (43)	81.60 (46)
Mfeat_fac	**89.03** (43)	87.27 (50)	87.45 (47)	88.62 (50)	87.44 (50)
Mfeat_pix	**78.81** (49)	77.36 (50)	74.02 (50)	78.00 (44)	75.77 (50)
Semeion	**76.75** (50)	70.92 (50)	68.41 (48)	74.80 (48)	70.24 (50)
ORL	**55.95** (50)	55.03 (50)	53.92 (50)	51.17 (43)	46.88 (47)
COIL20	92.81 (46)	77.81 (49)	90.66 (47)	**93.20** (50)	87.25 (50)
gisette	**94.21** (46)	94.17 (50)	91.68 (50)	92.15 (50)	92.33 (50)
orlraws10P	**81.40** (31)	68.30 (27)	65.70 (50)	65.90 (3)	52.60 (50)
Avg.	**81.99**	78.24	77.95	79.81	77.18

Bold indicates that the best results can be achieved

5 Conclusions and Future Work

This paper first analyzes the objective function of mRMR. Then, for the case where the objective function cannot guarantee the effectiveness of selected features, an idea of equal interval division and ranking is proposed to process mutual information between the class label and features, and the average value of mutual information between features. Ultimately, EID–mRMR is proposed. To verify the performance, we apply EID–mRMR to three classifiers, eight UCI datasets and four ASU datasets, and compare results with those from seven

Table 17 Classification accuracy (%) of the optimal features selected by EID–mRMR and other algorithms when using IB1

Datasets	EID–mRMR	Relief-F	Fisher	QPFS	SPEC-CMI
Spambase	90.70 (50)	90.80 (49)	90.11 (45)	89.60 (39)	**90.88** (45)
Synthetic_control	97.58 (49)	97.80 (50)	97.08 (50)	97.77 (42)	**98.27** (44)
Mfeat_fou	83.15 (23)	83.61 (22)	83.46 (31)	83.70 (12)	**84.05** (14)
Movement_libras	85.03 (37)	82.94 (50)	76.11 (50)	**85.67** (31)	83.94 (49)
Musk	**86.18** (50)	82.68 (50)	85.40 (50)	83.40 (50)	84.05 (33)
Mfeat_fac	96.48 (30)	94.19 (50)	95.21 (47)	**96.57** (50)	94.92 (50)
Mfeat_pix	**93.94** (50)	84.83 (50)	80.99 (50)	93.64 (50)	89.40 (50)
Semeion	**86.40** (47)	77.96 (50)	73.28 (50)	80.92 (50)	79.34 (50)
ORL	**93.03** (49)	83.27 (50)	86.83 (50)	90.67 (49)	78.58 (50)
COIL20	**99.87** (46)	85.71 (50)	96.26 (50)	99.74 (50)	98.73 (50)
gisette	94.32 (50)	**94.71** (49)	91.92 (45)	92.47 (50)	92.36 (50)
orlraws10P	**97.20** (41)	81.60 (49)	87.20 (50)	77.50 (49)	75.60 (48)
Avg.	**91.99**	86.68	86.99	89.30	87.51

Bold indicates that the best results can be achieved

Table 18 Classification accuracy (%) of the optimal features selected by EID–mRMR and other algorithms when using Naive Bayes

Datasets	EID–mRMR	Relief-F	Fisher	QPFS	SPEC-CMI
Spambase	90.26 (12)	77.14 (49)	**90.69** (29)	89.50 (8)	90.68 (29)
Synthetic_control	**97.98** (31)	94.77 (50)	94.40 (50)	96.98 (38)	96.92 (45)
Mfeat_fou	79.34 (21)	**79.53** (22)	78.93 (28)	79.37 (21)	79.13 (20)
Movement_libras	**61.03** (37)	52.78 (49)	49.64 (50)	59.03 (42)	55.83 (50)
Musk	**82.77** (50)	80.20 (50)	78.88 (36)	78.83 (50)	79.61 (50)
Mfeat_fac	**95.48** (49)	88.63 (50)	87.82 (50)	94.77 (50)	89.02 (48)
Mfeat_pix	**92.78** (50)	83.00 (50)	81.92 (50)	92.16 (50)	89.04 (50)
Semeion	**83.97** (50)	75.14 (50)	71.80 (50)	79.86 (50)	75.75 (50)
ORL	**88.05** (50)	62.65 (50)	55.08 (50)	75.63 (50)	55.77 (50)
COIL20	**99.12** (50)	74.42 (50)	91.06 (50)	98.29 (49)	89.45 (50)
gisette	**91.76** (50)	86.34 (29)	86.74 (45)	86.85 (50)	86.85 (50)
orlraws10P	**98.20** (41)	72.10 (50)	76.70 (50)	75.40 (35)	61.60 (50)
Avg.	**88.40**	77.23	78.64	83.89	79.14

Bold indicates that the best results can be achieved

incremental MI-based algorithms and other four feature selection algorithms. Experimental results validate that EID–mRMR can achieve better feature selection effectiveness in the majority of datasets.

Considering that EID–mRMR currently only adopts mutual information, and it does not utilize three-way interaction information or higher dimensional mutual information [38–46], it is likely to degrade the performance due to missing some useful information. In the next stage, we will investigate how to combine three-way interaction information with the idea of equal interval division and ranking.

Acknowledgements This work was supported by the National Natural Science Foundation of China (61771334).

References

1. Tan MK, Tsang IW, Wang L (2014) Towards ultrahigh dimensional feature selection for big data. J Mach Learn Res 15:1371–1429
2. Cataron A, Andonie R (2010) Energy supervised relevance neural gas for feature ranking. Neural Process Lett 32(1):59–73
3. Borja SP, Veronica BC, Amparo AB (2017) Testing different ensemble configurations for feature selection. Neural Process Lett 46:1–24
4. Tiwari S, Singh B, Kaur M (2017) An approach for feature selection using local searching and global optimization techniques. Neural Comput Appl 28(10):2915–2930
5. Wang JZ, Wu LS, Kong J, Li YX, Zhang BX (2013) Maximum weight and minimum redundancy: a novel framework for feature subset selection. Pattern Recognit 46(6):1616–1627
6. Shang CX, Li M, Feng SZ, Jiang QS, Fan JP (2013) Feature selection via maximizing global information gain for text classification. Knowl-Based Syst 54:298–309
7. Tang B, Kay S, He HB (2016) Toward optimal feature selection in naive Bayes for text categorization. IEEE Trans Knowl Data Eng 28(9):2508–2521
8. Fei T, Kraus D, Zoubir AM (2012) A hybrid relevance measure for feature selection and its application to underwater objects recognition. In: International conference on image processing, pp 97–100
9. Fei T, Kraus D, Zoubir AM (2015) Contributions to automatic target recognition systems for underwater mine classification. IEEE Trans Geosci Remote Sens 53(1):505–518
10. Zhang F, Chan PPK, Biggio B, Yeung DS, Roli F (2016) Adversarial feature selection against evasion attacks. IEEE Trans Cybern 46(3):766–777
11. Veronica BC, Noelia SM, Amparo AB (2013) A review of feature selection methods on synthetic data. Knowl Inf Syst 34(3):483–519
12. Jia XP, Kuo BC, Crawford MM (2013) Feature mining for hyperspectral image classification. Proc IEEE 101(3):676–697
13. Lin CH, Chen HY, Wu YS (2014) Study of image retrieval and classification based on adaptive features using genetic algorithm feature selection. Expert Syst Appl 41(15):6611–6621
14. Zhao YH, Wang GR, Yin Y, Li Y, Wang ZH (2016) Improving ELM-based microarray data classification by diversified sequence features selection. Neural Comput Appl 27(1):155–166
15. Estevez PA, Tesmer M, Perez CA, Zurada JM (2009) Normalized mutual information feature selection. IEEE Trans Neural Netw 20(2):189–201
16. Brown G, Pocock A, Zhao MJ, Lujan M (2012) Conditional likelihood maximisation: a unifying framework for information theoretic feature selection. J Mach Learn Res 13:27–66
17. Vergara JR, Estevez PA (2014) A review of feature selection methods based on mutual information. Neural Comput Appl 24(1):175–186
18. Lewis DD (1992) Feature selection and feature extraction for text categorization. In: Proceedings of the workshop on speech and natural language, pp 212–217
19. Peng HC, Long FH, Ding C (2005) Feature selection based on mutual information: criteria of max-dependency, max-relevance and min-redundancy. IEEE Trans Pattern Anal Mach Intell 27(8):1226–1238
20. Hoque N, Bhattacharyya DK, Kalita JK (2014) MIFS-ND: a mutual information-based feature selection method. Expert Syst Appl 41(14):6371–6385
21. Han M, Ren WJ (2015) Global mutual information-based feature selection approach using single-objective and multi-objective optimization. Neurocomputing 168:47–54
22. Zhang Y, Ding C, Li T (2008) Gene selection algorithm by combining reliefF and mRMR. BMC Genom 9(2):1–10
23. Unler A, Murat A, Chinnam RB (2011) mr^2PSO: a maximum relevance minimum redundancy feature selection method based on swarm intelligence for support vector machine classification. Inf Sci 181(20):4625–4641
24. Battiti R (1994) Using mutual information for selecting features in supervised neural net learning. IEEE Trans Neural Netw 5(4):537–550
25. UCI machine learning repository. http://archive.ics.uci.edu/ml. Accessed 6 Mar 2018
26. ASU feature selection datasets. http://featureselection.asu.edu/datasets/. Accessed 6 Mar 2018
27. Fayyad U, Irani K (1993) Multi-interval discretization of continuous-valued attributes for classification learning. In: Proceedings of joint conference on artificial intelligence, pp 1022–1027

28. Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH (2009) The weka data mining software: an update. ACM SIGKDD Explor Newsl 11(1):10–18
29. Zhao Z, Morstatter F, Sharma S, Alelyani S, Anand A, Liu H (2010) Advancing feature selection research. In: ASU feature selection repository, pp 1–28
30. Kwak N, Choi CH (2002) Input feature selection for classification problems. IEEE Trans Neural Netw 13(1):143–159
31. Wang ZC, Li MQ, Li JZ (2015) A multi-objective evolutionary algorithm for feature selection based on mutual information with a new redundancy measure. Inf Sci 307:73–88
32. Foithong S, Pinngern O, Attachoo B (2012) Feature subset selection wrapper based on mutual information and rough sets. Expert Syst Appl 39(1):574–584
33. Kononenko I (1994) Estimating attributes: analysis and extension of RELIEF. In: Proceedings of European conference on machine learning, pp 171–182
34. Duda RO, Hart PE, Stork DG (2001) Pattern classification. Wiley-Interscience Publication, New York
35. Rodriguez-Lujan I, Huerta R, Elkan C, Cruz CS (2010) Quadratic programming feature selection. J Mach Learn Res 11:1491–1516
36. Nguyen XV, Chan J, Romano S, Bailey J (2014) Effective global approaches for mutual information based feature selection. In: Proceedings of ACM SIGKDD international conference on knowledge discovery and data mining, pp 512–521
37. Herman G, Zhang B, Wang Y, Ye GT, Chen F (2013) Mutual information-based method for selecting informative feature sets. Pattern Recognit 46(12):3315–3327
38. Fleuret F (2004) Fast binary feature selection with conditional mutual information. J Mach Learn Res 5:1531–1555
39. Sun X, Liu YH, Xu MT, Chen HL, Han JW, Wang KH (2013) Feature selection using dynamic weights for classification. Knowl-Based Syst 37:541–549
40. Bennasar M, Hicks Y, Setchi R (2015) Feature selection using joint mutual information maximisation. Expert Syst Appl 42(22):8520–8532
41. Zeng ZL, Zhang HJ, Zhang R, Yin CX (2015) A novel feature selection method considering feature interaction. Pattern Recognit 48(8):2656–2666
42. Shishkin A, Bezzubtseva AA, Drutsa A, Shishkov I, Gladkikh E, Gusev G, Serdyukov P (2016) Efficient high order interaction aware feature selection based on conditional mutual information. In: Proceedings of the conference on advances in neural information processing systems, pp 4637–4645
43. Vinh NX, Zhou S, Chan J, Bailey J (2016) Can high-order dependencies improve mutual information based feature selection. Pattern Recognit 53:46–58
44. Wang J, Wei JM, Yang ZL, Wang SQ (2017) Feature selection by maximizing independent classification information. IEEE Trans Knowl Data Eng 29(4):828–841
45. Gao WF, Hu L, Zhang P (2018) Class-specific mutual information variation for feature selection. Pattern Recognit 79:328–339
46. Gao WF, Hu L, Zhang P, He JL (2018) Feature selection considering the composition of feature relevancy. Pattern Recognit Lett 112:70–74