# I-mRMR: Incremental Max-Relevance, and Min-Redundancy Feature Selection

**Abstract**. An incremental method of feature selection based on mutual information, called incremental Max-Relevance, and Min-Redundancy (I-mRMR), is presented. I-mRMR is an incremental version of Max-Relevance, and Min-Redundancy feature selection (mRMR), which is used to handle streaming data or large-scale data. First, Incremental Key Instance Set is proposed which composes of the non-distinguished instances by the historical selected features. Second, an incremental feature selection algorithm is designed in which the Incremental Key Instance Set, replacing of all the seen instances so far, is used in the process of adding representative features. Since the Key Instance Set is far less than the whole instances, the incremental feature selection by using this key set avoid redundant computation and save computation time and space. Finally, the experimental results show that I-mRMR could significantly or even dramatically reduce the time of feature selection with an acceptable classification accuracy. The main advantage of I-mRMR is that it makes full use of the historical information, reduce the training scale greatly, and save training time.

**Index Terms**：Feature selection, Incremental algorithm, normalized mutual information, min-redundancy, max-relevance.

1. **Introduction**

In recent years, we encounter databases in which the issue of data is too big to load in the memory or the data are streaming collected over time. Storing and processing all data might be computationally costly and impractical. To deal with this issue, incremental algorithms have become feasible and effective tool in machine learning and data mining techniques [6-10].

Incremental learning is a promising approach to refreshing data mining results, which utilizes previously saved results or data structures to avoid the expense of re-computation [12–17]. Incremental feature selection (sometimes also called online feature selection) is one important part of “incremental machine learning” fields [50][19]. The main idea of incremental feature selection is that only part of the data are to be considered at one time and the results are subsequently combined. Thus incremental feature selection technique makes full use of the historical information, reduce the training scale greatly, and save training time [5].

Feature selection based on mutual information ,as a known way of feature selection, have been deeply studied [] .Since mutual information (MI) [references ?] is a good tool to measure the correlation and redundancy among features. As a pioneer, Battiti [using] proposes a greedy selection method called MIFS based on mutual information between inputs and outputs. Considering MIFS does not work well in nonlinear problems, Kwak and Choi [Input] proposed an improved feature selection method MIFS-U which is feasible and effective on nonlinear applications. However, both Battiti and Kwak’s methods omit the redundancy among features, only relevance among features and labels are considered. Peng et al. [?] then proposed a heuristic "Max-Relevance and Min-Redundancy" framework for feature selection. In[] it is pointed that mRMR criterion is equal to max-dependency. Furthermore, Pablo and Tesmer [] proposed an updated feature selection method, called normalized mutual information features selection. This method updated the Max-Relevance and Min-Redundancy criterion and numerical experiments demonstrates that it is faster and better than MIFS, MIFS-U. However, there exist one common limitation among the mentioned methods. That is, most of them could only be applied to static data. When new instances are arriving successively, these methods have to be re-computed on the updated datasets.

To select features on streaming datasets, some incremental feature selection algorithms have been proposed based on entropy or its generalization [Liang, 10, 25, 28, 51]. Whereas max-relevance and min-redundancy based MI is not considered yet. Now, it is promising to design an incremental feature selection method based on max-relevance and min-redundancy.

In this paper, we propose an incremental feature selection algorithm, called Incremental Max-Relevance, and Min-Redundancy Feature Selection (I-mRMR). First, Incremental Key Instance Set is proposed which is composed of part of instances not distinguished by historical selected features. An incremental algorithm is then proposed based on this Incremental Key Instance Set. Finally, the main advantage of I-mRMR is that it makes full use of the historical selected features, reduce the training scale greatly, and save training time.

The remainders of this paper are organized as follows. Section 2 reviews MI and mRMR based on normalized mutual information. Section 3 introduces the concept of Incremental Key Instance Set and presents the incremental feature selection algorithm I-mRMR. In Section 4, ten UCI Datasets and two extremely high dimensional datasets are employed to illustrate the effectiveness and efficiency of I-mRMR. Section 5 concludes this paper.

1. **Preliminaries**

In this section, MI and mRMR are reviewed. For more detailed information about them, please kindly refer to [1,3,10,11,12,13,14,15,17]. Since we will compare it with GIARC algorithm in the experimental part, we then will briefly introduce it.

* 1. **notation description**

Given a dataset of original instances . Here is a matrix with is the number of original instances and is the number of all features, is a row vector representing -th instance in . is the index set of selected feature subset. denotes the complementary set of S. is a column vector representing the -th feature. represents a vector of under feature subset S(i=1,…,n), is a column vector representing the label feature in U. Here is the label for the i-th instance in U(i=1,…,n).

* 1. **Mutual information**

MI is often used to measure the relevance of two random variables, which is defined as follows.

Formally, given two discrete random variables X and , MI between X and with a joint probability mass function and marginal probabilities and is defined as follows:

The larger the value of MI, the larger the relevance of X and is. Considering features as variables, the larger the value of MI, the larger the relevance among features is, the more instances with different labels could be distinguished.

Just as described in [17], MI can be written as the following equation:

Where is the information entropy, is the conditional entropy. It is easy to derive the range of mutual information .

MI is often used to describe the relevance degree between features and label [3,5,6,7,8,9]. It is preferential to select the features with the maximum MI to add to the feature subset candidate. Since every feature could be seen as a variable, we use the notation of feature to replace the notion of variables when no confusion arises in the following of this paper.

* 1. **Max-Relevance and Min-Redundancy**

Max-Relevance is to find the feature that satisfies the following formula:

, where

By the Max-Relevance criterion, only the relevance between the features and labels are considered, whereas the relevance among the features is not considered. Thus there may exist great redundancy among the selected features. As a result, it is necessary to make the redundancy among the selected features as small as possible.

, where

The above two criteria are combined, called "Max-Relevance and Min-Redundancy", and defined as follows.

Suppose that the feature subset candidate we have selected so far is , and indicates that features have been selected. And then the feature with the maximum value of is selected. The incremental feature selection algorithm optimizes the following formula:

.

Based on this idea, a series of feature selection methods based on max-relevance and min-redundancy have be done, please kindly refer to [1,3,14,16] . Among these works, normalized mutual information based mRMR is the state-to-art work, and then in the following we briefly review it, based on which, we design an incremental feature selection method.

* 1. **Normalized mutual information feature selection**

Since the criteria of the relevance between and are not uniform, the normalized mutual information is then used as the evaluation criterion to measure the degree of feature relevance.

The normalized mutual information between the feature and the feature is then defined as follows.

Therefore, "Max-Relevance and Min-Redundancy" criterion is rewritten as follows:

.

The algorithm based on normalized mutual information feature selection is formulated in Algorithm 1. In this paper, we briefly notate this algorithm mRMR when no confusion arises.

|  |
| --- |
| **Algorithm 1**: Normalized Mutual Information Feature Selection(mRMR) |
| Input: U, the given number of selected features K,  Output: the set containing the selected features |
| Step 1: Compute for every .  Step 2: Select the first feature .  Step 3: Set , .  Step 4: While do.  {  Compute for all pairs,with and if it not available;  Select the feature which maximizes measures ;  Set ; .  }  Step 5: Return . |

**2.5 A group incremental algorithm for reduct computation (GIARC)**

To select useful features from a dynamically increasing data set,Liang[] proposed a group incremental algorithm for reduct computation in the framework of rough set theory, and this algorithm employs information entropy to measure the feature significance. When a group new instances coming, the new Shannon’s conditional entropy as follow:

, where

1. The Proposed Incremental Algorithm

When some new instances arriving, the classical feature selection algorithm has to be recomputed on all the seen data so far, which is really time and computation consuming. To solve this problem, we propose an incremental algorithm based on the "Max-Relevance and Min-Redundancy" criterion. The key idea of our proposed method is to update and maintain the previously selected feature subset by finding the features more representative for discriminating the new instances from its current surrounding.

3.1 Problem Definition

When some new instances, represented by (where represents the number of newly added instances), are added to , is the label for the j-th instance in ,j=1,…,m. The selected feature subset has to be updated from U to . The traditional method is directly to recompute the feature selection method on all seen instances to obtain the updated feature subset . It is very time and space consuming and many redundant computations are conducted. Therefore, it is necessary to reduce the amount of computation by using some incremental mechanisms. And we refer to feature selection method based incremental mechanism as incremental feature selection.

An incremental feature selection algorithm is then proposed based on normalized mutual information. The purpose of this algorithm is to dynamically find the selected feature subset .

3.2 Incremental Key Instance Set

To incrementally update the selected feature subset , it is necessary to find the features more representative for discriminating the new instances from its current surrounding. That is to say, it is fundamental to find those instances which are key to select the new representative features.

In the following we propose a concept called Incremental Key Instance Set which composes of part of the seen instances so far which are undistinguished by the original feature subset .

**Definition 1.** Given , , , then the “Incremental Key Set” of , denoted by , is defined as follows.

Incremental Key Instance Set composes of such instances which have the same feature values on but the different labels, which means that the features in could not distinguish the new instances from its current surrounding and then some new features should be added. plays the key role to find the new features.

Usually, the conditional entropy is used to measure the significance of . Here the feature subset functions as a random variable. means that the selected feature subset could definitely distinguish all the instances so far. The larger is, some more instances could not be distinguished by .

Theorem 1: For the selected features subset S, we have on , which implies

The selected features subset can completely separate datasets other than Incremental Key Instance Set .

A function that measures the significance of the feature according to the criterion of the "Max-Relevance and Min-Redundancy" is then proposed based on Increment Key Instance Set. Since Increment Key Instance Set is far less than all the seen instances so far, it greatly reducing the computation, time and space consumption when updating feature selection.

**Definition 2.** Given, , and , for every and , the significance degree of with respect to and on is defined as follows.

.

3.3 Incremental feature selection

In this subsection, we present the incremental feature selection algorithm when a set of new instances arriving. I-mRMR is designed in algorithm 2.

|  |
| --- |
| **Algorithm 2**. An incremental algorithm for feature selection based on Max-Relevance, and Min-Redundancy (I-mRMR) |
| Input: , , , , ,  Output: on . |
| Step 1: compute .  Step 2: If , go to Step 6, else go to Step 3.  Step 3: Compute ,H(Y) on .  If ,go to Step 6;  Else go to Step 4.  Step 4: While do.  {  For every ,Compute on ;  Select ;  ,  Update on .  }  Step 5:  Step 6: Return . |

3.4 Scalability Analysis

The time complexity of the algorithm is shown in Table 1. The main step of I-mRMR is to find the Incremental Key Instance Set , which gradually decreases the computation of choosing new representative features.

From Table 1, it is to see that When the dimension of the datasets is high, the time complexity of I-mRMR is . Comparing with the time complexity of mRMR, I-mRMR spends less time because is far smaller than 。In the following numerical experimental part, we experimentally demosntrate the efficiency of I-mRMR.

Table 1: Time complexity of the algorithm

|  |  |
| --- | --- |
| Algorithm | Time Complexity |
| mRMR |  |
| I-mRMR |  |

4. Numerical Experiments

In this section, we conduct some numerical experiments to evaluate the proposed algorithm, I-mRMR, on ten datasets from UCI and two extremely high dimensional datasets from KDD cup, seen in Table $2\&8$. The Max-Relevance and Min-Redundancy feature selection based on normalized mutual information, denoted by mRMR [1], as the classical non-incremental feature selection algorithm, is compared with I-mRMR. Also, an entropy-based incremental feature selection algorithm GIARC, as an incremental feature selection algorithm, is compared with our proposed algorithm [2]. Because GIARC and I-mRMR are comparable as both of their measures of feature relevance are generalized by information entropy.

4.1 Experimental Setup

All the experiments have been conducted on computer with CentOS release 6.5(Final), Westmere E56xx/L56xx/X56xx(Nehalem-C) and 8GB memory. The programming language is Python. The detail experimental setting are presented as follows.

(1) Since our algorithm is only valid for discrete data, fuzzy-c-means is used to discretize those continuous data sets.

(2) Every dataset is divided into six parts equally, the first part is used as the original data set , and remaining parts as the newly added dataset , are added one by one.

(3) All the experimental comparison is demonstrated from four indices: running time, global speedup ratio, local speedup ratio and classification accuracy.

Global speedup ratio:

Where denotes the running time of mRMR on the seen instances so far, denotes the running time of I-mRMR on the seen instances so far. When the dataset is divided into six parts, represents the sum of six times running time of mRMR, where each time the dataset is updated when some new instances arriving.

Local speedup ratio:

When the dataset is divided into six parts, the local speedup ratio is the ratio of the running time of mRMR on the whole dataset to the running time of I-mRMR when the last part arriving.

(4) To show the effectiveness of I-mRMR, SVM and KNN are used to evaluate the classification performance. And 5-fold cross validation is used in classification evaluation.

4.2 Experimental : Evaluation on UCI

To test the performance of I-mRMR, some experimental comparison and analyses are conducted on ten UCI datasets, seen in Table 2. Among the datasets in Table 2, 'madelon', 'arcene' are used in the NIPS 2003 Feature selection challenge. The dataset 'colon' is frequently studied as public microarray datasets [35].

Table 2: The description of the selected datasets from UCI

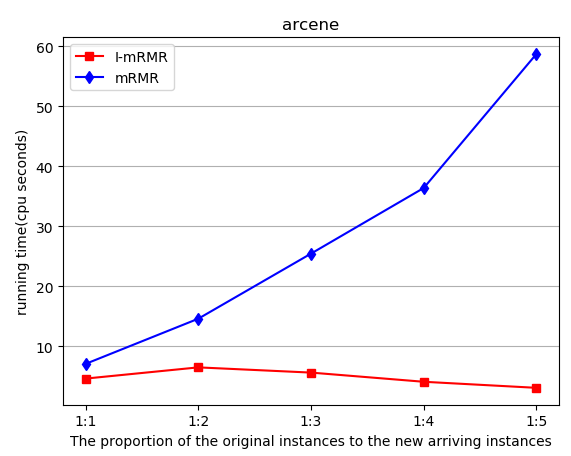
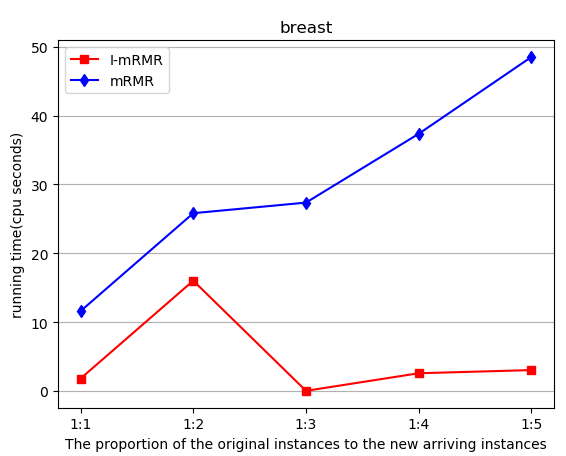
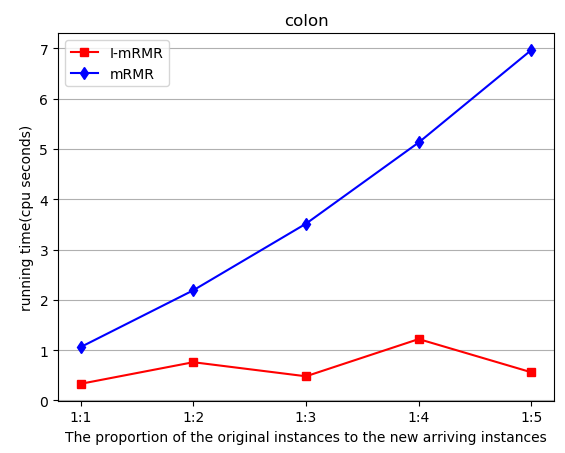
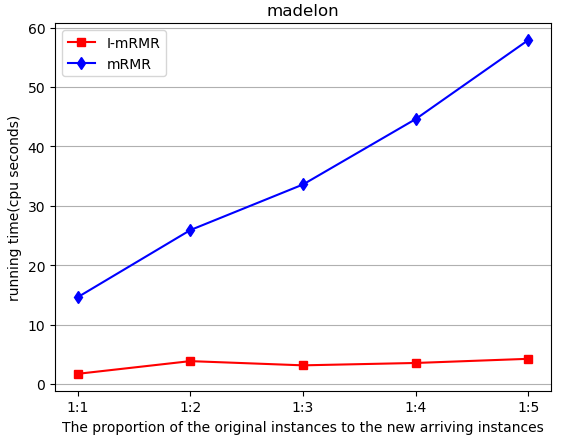
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Dataset | Instances | Features | Data Type | Classes |
| madelon | 2000 | 500 | continuous | 2 |
| colon | 62 | 2000 | continuous | 2 |
| breast | 84 | 9216 | continuous | 5 |
| arcene | 100 | 10000 | continuous | 2 |
| Gene9 | 203 | 12600 | continuous | 5 |
| TCGA\_PANCAN | 801 | 16383 | continuous | 5 |
| Ad | 2359 | 1558 | continuous | 2 |
| FPS5 | 3600 | 3208 | continuous | 6 |
| FPS7 | 3600 | 4813 | continuous | 6 |
| Gisette | 7000 | 5000 | continuous | 2 |

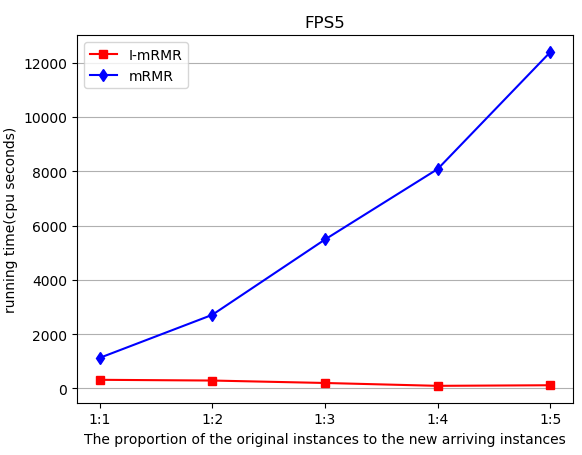
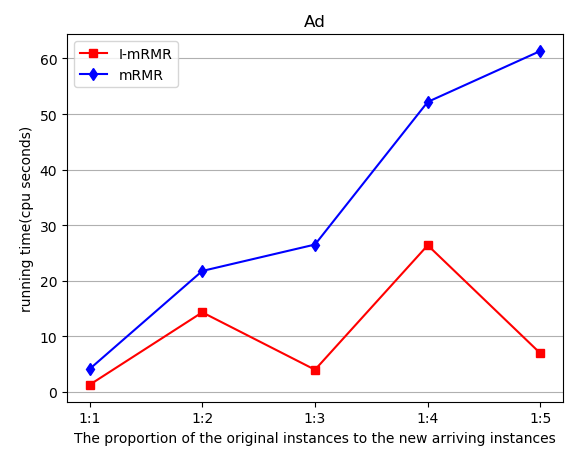
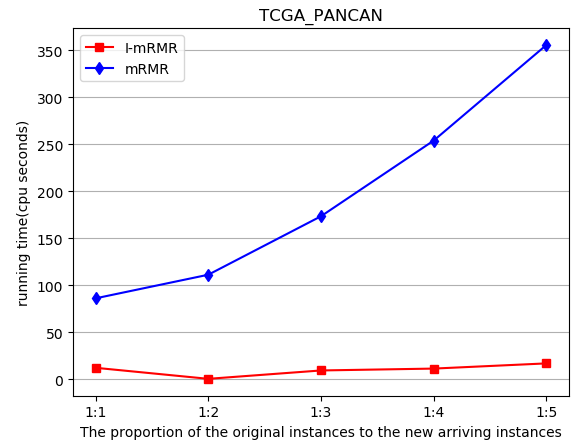
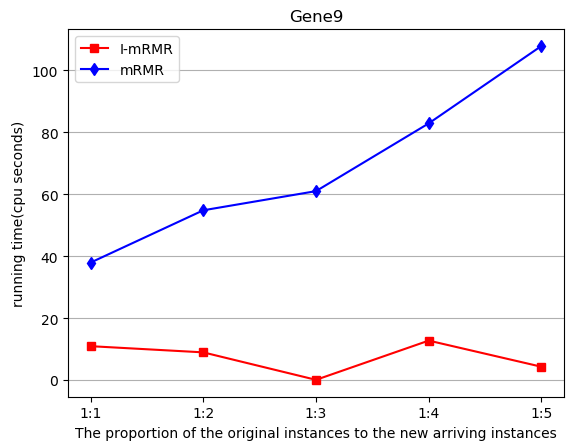
4.2.1 Compared with mRMR

In this part, I-mRMR and mRMR are compared. Both of them are feature selection methods based on normalized mutual information of "Max-Relevance and Min-Redundancy" criterion. One main difference between them is that I-mRMR is an incremental feature selection algorithm, whereas mRMR is a non-incremental feature selection algorithm. Another obvious difference of them lies in the different stop criteria. The stopping criterion of I-mRMR is that the mutual information of the selected feature subset is equal to the information entropy of the label feature Y, whereas the stop criterion of mRMR is to select top k features.

1. The Comparison of the running time

In this part, we demonstrate the running time of I-mRMR and mRMR when instances successively arriving and then graph them in Figure 1.





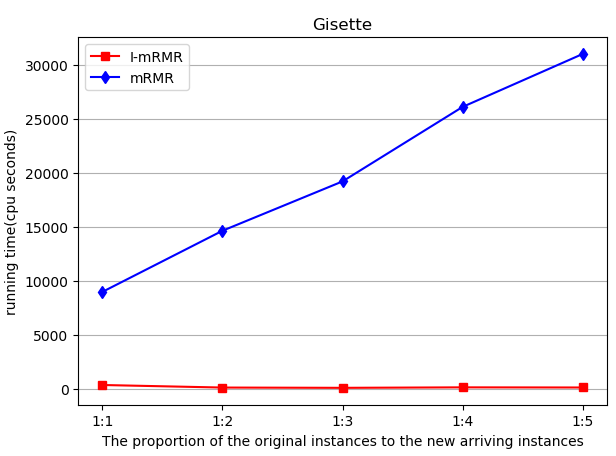
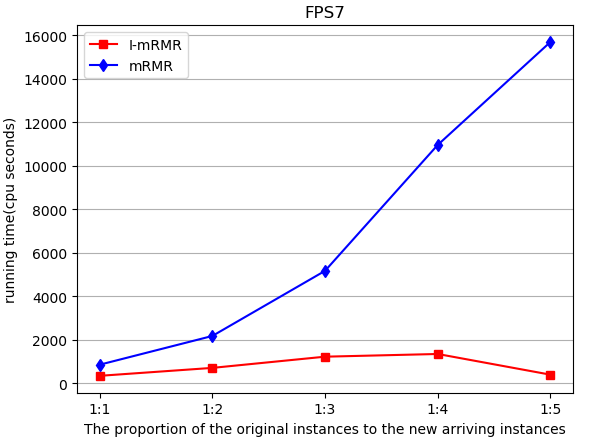


Figure 1: The running time of I-mRMR and mRMR with instances successively arriving

Figure 1 clearly demonstrates that the running time of I-mRMR changes slightly, whereas the running time of mRMR increases significantly with the instances successively arriving. This shows that I-mRMR works efficiently on streaming instances, whereas mRMR works more and more less-efficiently.

To further illustrate the time superiority of I-mRMR, the global speedup ratio is then presented, seen in Tables 3&4.

Table 3: The global speedup ratio of mRMR and I-mRMR

|  |  |  |  |
| --- | --- | --- | --- |
| Dataset | mRMR | I-mRMR | Global accelerate ratio |
| madelon | 184.32s | 32.49s | 5.67 |
| colon | 19.08s | 3.79s | 5.03 |
| breast | 156.96s | 29.52s | 5.31 |
| arcene | 144.37s | 29.73s | 4.85 |
| Gene9 | 377.17s | 69.47s | 5.43 |
| TCGA\_PANCAN | 1026.37s | 121.77s | 8.42 |
| Ad | 168.14s | 58.17s | 2.89 |
| FPS5 | 30092(8hr21m32s) | 1654(27m34s) | 18.19 |
| FPS7 | 35092(9hr44m52s) | 4501(1hr25m1s) | 7.79 |
| Gisette | 103161(28hr39m21s) | 10801(3hr1s) | 9.55 |
| Average | 17420(4hr50m6s) | 1730(28m50s) | 7.31 |

Table 3 shows that the total time of mRMR is obviously or even significantly higher than that of I-mRMR, especially on the datasets with high number of instances. This is because when some new instances arriving mRMR has to be recomputed on the whole seen instances so far, which is really time consuming. Take 'Gisette' as an example, we present the detail running time of I-mRMR and mRMR in Table 4. Table 4 clearly demonstrates that mRMR is really time consuming and conducts much redundant computation.

Table 4: the detail running time of Gisette when new instances arriving successively

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Algorithm | | 1st new instances arriving | 2st new instances arriving | 3st new instances arriving | 4st new instances arriving | 5st new instances arriving |
| I-mRMR | Selected Feature No. | 69 | 73 | 74 | 76 | 76 |
| running time | 337s | 100.85s | 74.6s | 119s | 105.6s |
| mRMR | Selected Feature No. | 69 | 73 | 74 | 76 | 76 |
| running time | 8954(2hr29m14s) | 14617(4hr3m23s) | 19188(5hr19m48s) | 26088(7hr14m48s) | 30991(8hr36m31s) |

Furthermore, Table 5 demonstrates the time superiority of I-mRMR from the aspect of local speedup ratio. From Table 5 we observe that I-mRMR is significantly or even dramatically faster than mRMR. This is because I-mRMR only consider part of instances which are not distinguished by the previous selected features, whereas mRMR computes on the whole seen instances so far. Take three datasets with high number of instances 'FPS5', 'FPS7' and 'Gisette' as examples. We find that the running time of I-mRMR when the 5th part of new instances arriving is far less than that of mRMR. This is because I-mRMR find that the previously selected features are enough to distinguish the new arriving instances and no additive feature need to be added, which dramatically saves the running time and avoids the redundant computation.

Table 5: The local speedup ratio of mRMR and I-mRMR

|  |  |  |  |
| --- | --- | --- | --- |
| Dataset | mRMR | I-mRMR | Local speedup ratio |
| madelon | 57.92s | 4.23s | 13.69 |
| colon | 19.08s | 0.56 | 34.07 |
| breast | 48.53s | 3.02s | 16.06 |
| arcene | 58.7s | 3.05s | 19.24 |
| Gene9 | 107.9s | 4.25s | 25.38 |
| TCGA\_PANCAN | 355.8s | 16.74s | 21.25 |
| Ad | 61.36s | 6.88s | 8.92 |
| FPS5 | 12392(3hr26m32s) | 119s | 104 |
| FPS7 | 15690(4hr21m30s) | 398s | 39.4 |
| Gisette | 30991(8hr36m31s) | 105.6s | 293.5 |
| Average | 5978(1hr39m38s) | 66s | 57.5 |

1. The Comparison of the classification accuracy

In this part, to illustrate the effectiveness of I-MRMR, we compare the classification performance of I-mRMR and mRMR. For fairness, the number of the features selected by mRMR is set as same as the number of selected features by I-mRMR. The Classification accuracies of mRMR and I-mRMR are summarized in Table 6.

Table 6: The classification accuracy of mRMR and I-mRMR

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Dataset | Original | | | mRMR | | | I-mRMR | | |
| Feature No. | Accuracy | | Selected Feature No. | Accuracy | | Selected Feature No. | Accuracy | |
| SVM | KNN | SVM | KNN | SVM | KNN |
| madelon | 500 | 0.573 | **0.585** | 15 | **0.614** | 0.577 | 15 | 0.581 | 0.567 |
| colon | 2000 | 0.803 | 0.791 | 6 | **0.835** | **0.834** | 6 | 0.831 | 0.832 |
| breast | 9216 | 0.829 | 0.746 | 7 | **0.865** | **0.893** | 7 | 0.777 | 0.754 |
| arcene | 10000 | 0.679 | 0.649 | 7 | 0.699 | 0.661 | 7 | **0.699** | **0.667** |
| Gene9 | 12600 | 0.800 | 0.803 | 8 | 0.889 | 0.876 | 8 | **0.906** | **0.895** |
| TCGA-PANCAN | 16383 | **0.996** | **0.998** | 8 | 0.976 | 0.983 | 8 | 0.985 | 0.976 |
| Ad | 1558 | 0.839 | 0.947 | 7 | 0.868 | 0.924 | 7 | **0.943** | **0.947** |
| FPS5 | 3208 | 0.748 | 0.535 | 82 | **0.797** | 0.721 | 82 | 0.793 | **0.744** |
| FPS7 | 4812 | **0.816** | 0.596 | 79 | 0.777 | 0.678 | 79 | 0.784 | **0.726** |
| Gisette | 5000 | 0.896 | **0.963** | 76 | 0.943 | 0.932 | 76 | **0.946** | 0.951 |
| Average | 6456.9 | 0.798 | 0.761 | 29.5 | 0.826 | 0.808 | 29.5 | 0.825 | 0.806 |

Several observations can be drawn from the results in Table 6. First, I-mRMR and mRMR have the same number of selected features. This ensures the comparability of these two algorithms. Second, we find that I-mRMR has the similar or sometime slightly lower classification accuracy than mRMR. Whereas, I-mRMR has a better classification performance with dramatical dimension reduction than the original datasets. All these indicate that I-mRMR has an acceptable and effective classification performance.

4.2.2 Compared with GIARC

In this part, we compare I-mRMR with GIARC. One of the main similarities is that both of them utilize the measure generalized by information entropy to select the representative feature, another similarity is that both of them are incremental feature selection methods. As a result, it is necessary to compare their performance.

For fairness, the cost of de-redundancy of GIARC is not counted. That is to say, the real running time of GIARC is higher than presented in this part.

In the following, the total running time of these two algorithms are compared and graphed in Figure 2. Here, the total running time means the sum of time running mRMR/I-mRMR six times, where each time the dataset is updated when some new instances arriving.

1. The comparison of total running time

The total running time of I-mRMR compared with GIARC and graphed in Figure 2.

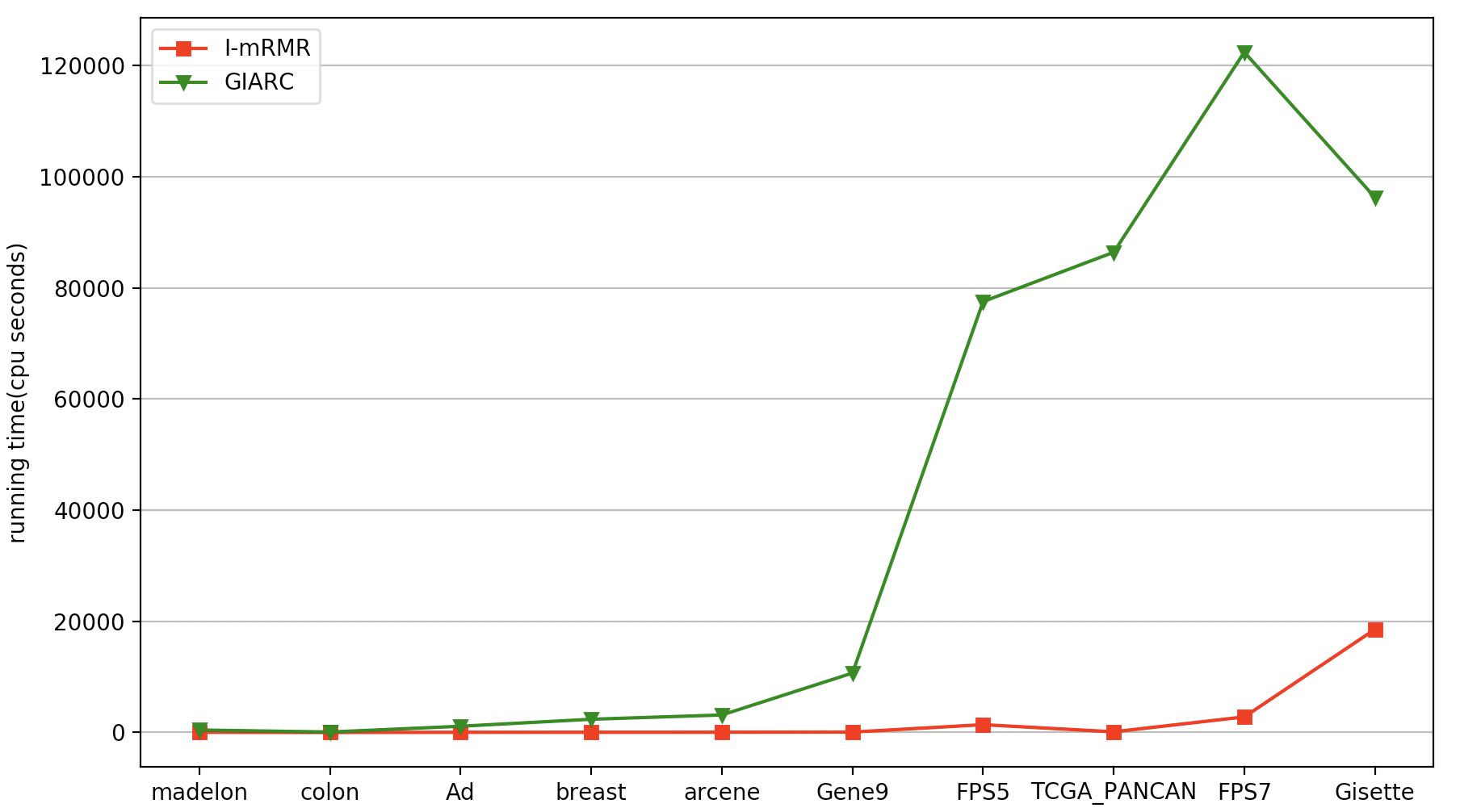


Figure 2: the total running time of I-mRMR and GIARC

The classification accuracy of features selected by I-mRMR and GIARC is shown in Table 7.

Table 7: Classification accuracy of I-mRMR and GIARC

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Data Set | I-mRMR | | | GIARC | | |
| Selected Feature No. | Accuracy | | Selected Feature No. | Accuracy | |
| SVM | KNN | SVM | KNN |
| madelon | 15 | **0.581** | 0.567 | 9 | 0.507 | 0.568 |
| colon | 6 | **0.831** | **0.832** | 4 | 0.823 | 0.76 |
| breast | 7 | **0.777** | **0.754** | 5 | 0.5 | 0.488 |
| arcene | 7 | **0.699** | **0.667** | 6 | 0.698 | 0.662 |
| Gene9 | 8 | **0.906** | **0.895** | 6 | 0.716 | 0.814 |
| TCGA-PANCAN | 8 | **0.985** | **0.976** | 8 | 0.918 | 0.899 |
| Ad | 7 | **0.943** | **0.947** | 5 | 0.914 | 0.914 |
| FPS5 | 82 | **0.793** | **0.744** | 42 | 0.603 | 0.575 |
| FPS7 | 79 | **0.784** | **0.726** | 32 | 0.502 | 0.491 |
| Gisette | 76 | **0.946** | **0.951** | 10 | 0.884 | 0.865 |
| Average | 29.5 | **0.825** | **0.806** | 12.7 | 0.707 | 0.704 |

According to the experimental results of this part, we can see from Figure 2 that GIARC consumes more time than I-mRMR, especially on the datasets with both high number of instances and high dimensionality, e.g. FPS5, FPS7, TCGA\\_PAVCAN and Gisette. In addition, it is easy from Table 7 to see that the classification accuracy of I-mRMR on all the datasets is obviously higher than that of GIARC. Although GIARC could select some more features to increase its classification accuracy by adjusting its threshold, it makes GIARC cost some more time and is far time consuming than I-mRMR. It is easy to get that I-mRMR is more feasible and effective.

4.3 Experimental : Evaluation on High dimensional Data Set

In this section, we conduct numerical experiments on two extremely high dimensional datasets, seen in Table 8, Here 'PEMS' comes from the California Department of Transportation PEMS website (www.pems.dot.gov), and 'thrombin' comes from KDD Cup 2001 competition.

Table 8: Summary of the two high dimensional datasets.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Dataset | Instances | Features | Data Type | Classes |
| PEMS\_train267 | 267 | 138672 | Continuous | 7 |
| thrombin | 1909 | 139351 | Discrete | 2 |

1. The comparison of the running time

First, we compare the running time of mRMR and I-mRMR on the two extremely high dimension datasets. Because GIARC spends too much time on these datasets, we have to terminate them when they run more than one week. The trends of the running time of I-mRMR and mRMR with the instances successively arriving are graphed in Figure 3 and the global/local speedup ratio of them are presented in Table 9&10.

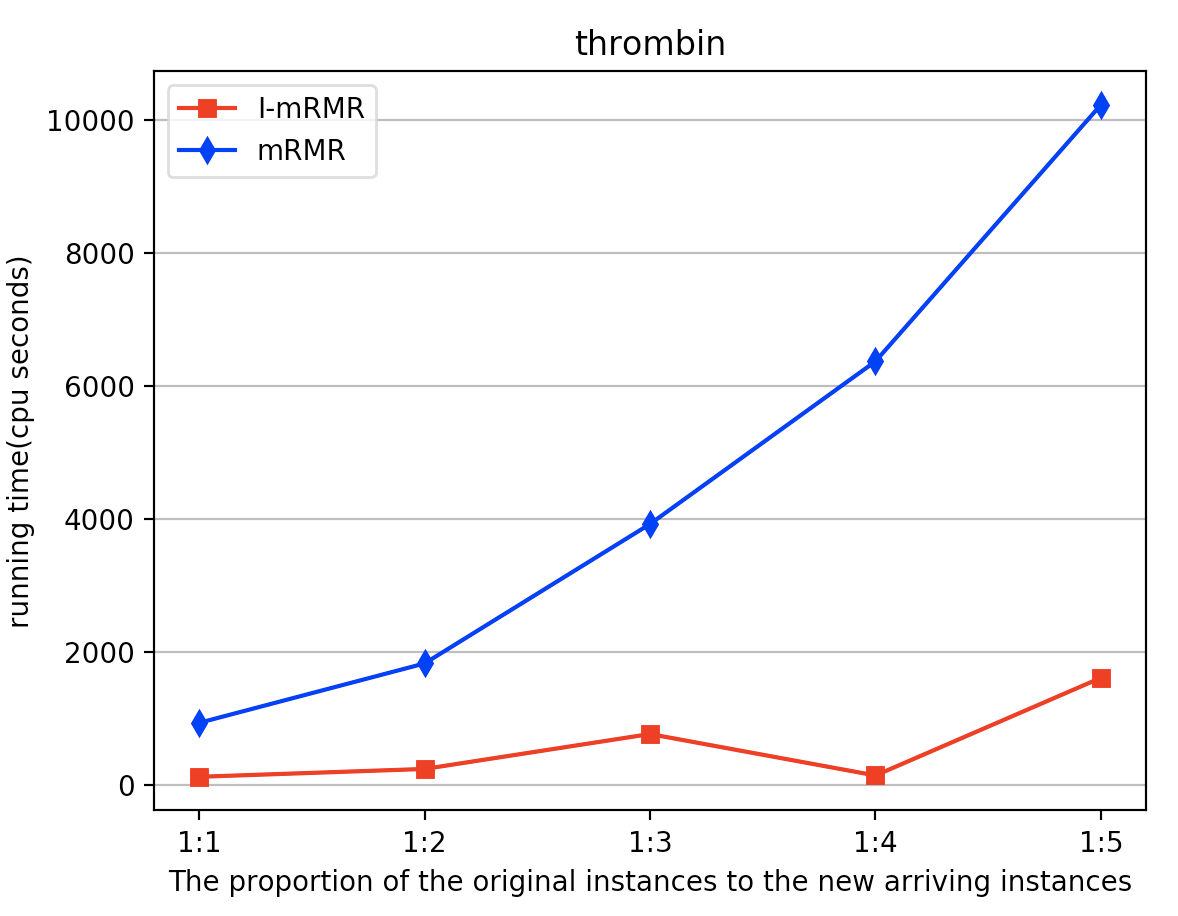
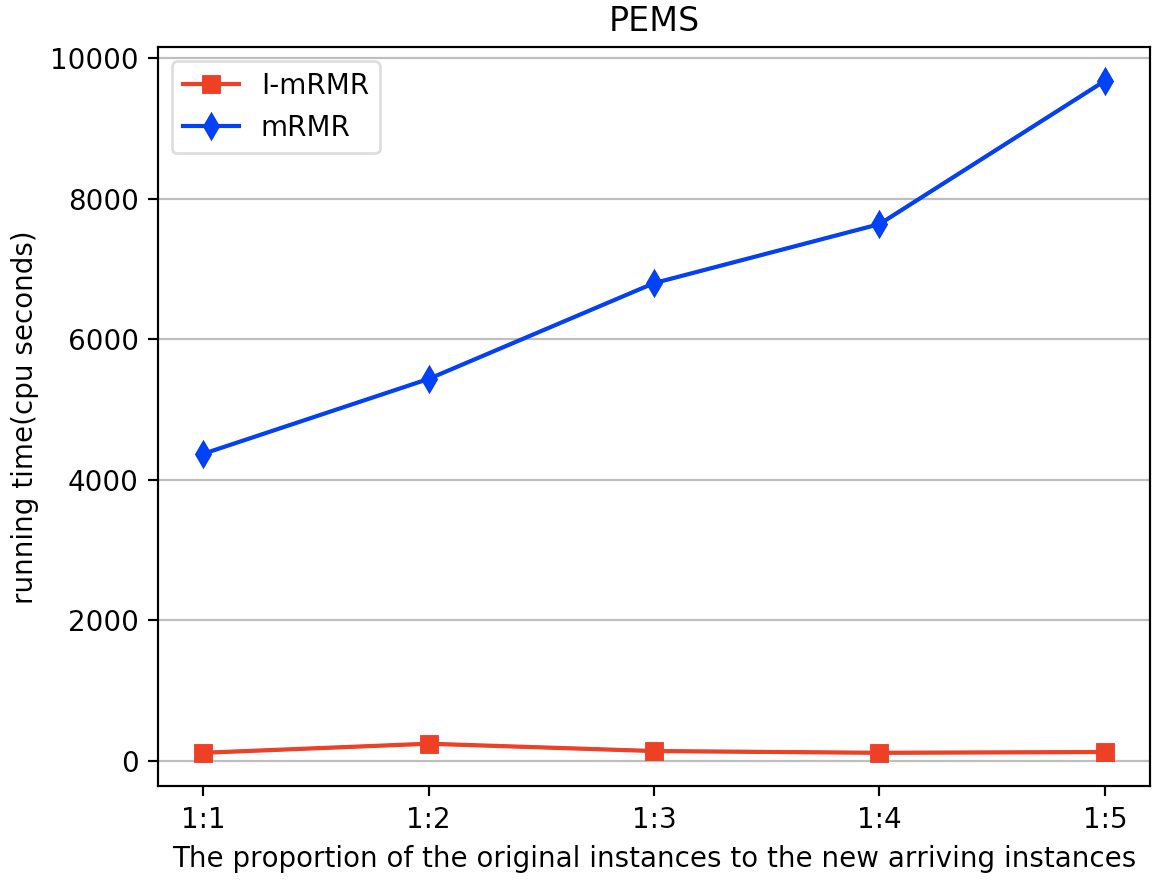


Figure 3: The trends of the running time of I-mRMR and mRMR with the instances successively arriving

Table 9: The Global speedup ratio of I-mRMR and mRMR

|  |  |  |  |
| --- | --- | --- | --- |
| Dataset | mRMR | I-mRMR | Global speedup Ratio |
| PEMS | 7172(1hr59m28s) | 1287s(21m27s) | 5.57 |
| thrombin | 7680(2hr40m) | 1872(31m12s) | 4.1 |
| Average | 7426(2hr3m46s) | 1579.5(26m19.5s) | 4.84 |

Table 10: Local Speedup Ratio of mRMR and I-mRMR

|  |  |  |  |
| --- | --- | --- | --- |
| Dataset | mRMR | I-mRMR | Local speedup Ratio |
| PEMS | 2130(35m30s) | 187s | 11.34 |
| thrombin | 2949(49m9s) | 85s | 34.7 |
| Average | 2539.5(42m19.5s) | 136s | 23.02 |

Several observations could be obtained from the above figures and tables. First, from Figure 3 we find that mRMR is more time consuming than I-mRMR with the instances successively arriving. Second, Table 9&10 globally and locally demonstrate that I-mRMR works significantly more efficient than mRMR on the datasets with extremely high dimensionality.

1. The comparison of classification accuracy

The classification accuracy of I-mRMR and mRMR on extremely high-dimensional datasets is shown in Table 11.

Table 11: The classification accuracy of I-mRMR and mRMR

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Data Set | Feature No. | Accuracy | | mRMR | | | I-mRMR | | |
| Feature No. | Accuracy | | Feature No. | Accuracy | |
| SVM | KNN | SVM | KNN | SVM | KNN |
| PEMS | 138672 | **0.869** | 0.787 | 10 | 0.829 | 0.865 | 10 | 0.855 | **0.904** |
| thrombin | 139351 | 0.978 | 0.982 | 5 | 0.978 | 0.973 | 5 | 0.**983** | **0.985** |
| Average | 139011 | **0.924** | 0.885 | 7.5 | 0.904 | 0.919 | 7.5 | 0.919 | **0.945** |

Table 11 shows the classification accuracy of mRMR and I-mRMR on the two high-dimensional data sets. The average classification accuracy of I-mRMR is similar or sometimes even higher than the average classification accuracy of mRMR. Therefore, these results verify that the proposed incremental feature selection algorithm is feasible and efficient on the extremely high-dimensional datasets with an acceptable classification accuracy.

In summary, this numerical experimental part demonstrates that I-mRMR is more efficient than mRMR with the acceptable classification performance.

1. Conclusions

In this paper, we propose an incremental feature selection algorithm I-mRMR based on max-relevance and min-redundancy criterion. When a new set of instances is arriving, not all seen instances so far are necessary to update the feature selection results. Actually, just an Incremental Key Instance Set, which is composed of the instances undistinguished by historical selected features, is key to update the feature subset. As a result, I-mRMR is designed by using Incremental Key Instance Set, which dramatically improve the efficiency of feature selection on streaming instances. By numerical experiments, we demonstrate that the proposed incremental algorithm is significantly faster than the classical algorithm mRMR not only in the global speedup ratio but also in the local speedup ratio. Furthermore, on the extremely high-dimensional dataset, we experimentally demonstrate that our proposed feature selection algorithm I-mRMR is obviously more efficient than mRMR with an acceptable classification accuracy.

**Acknowledgements**

This work is supported by National Key Research & Develop Plan (No.2016YFB1000702), National Key R&D Program of China(2017YFB1400700), National Basic Research Program of China (973) (No.2014CB340402), National High Technology Research and Development Program of China (863) (No.2014AA015204) and NSFC under the grant No. 61532021, 61772536, 61772537, 61732006, 61702522 and NSSFC (No.12\&ZD220), and the Fundamental Research Funds for the Central Universities, and the Research Funds of Renmin University of China(15XNLQ06). It was partially done when the authors worked in SA Center for Big Data Research in RUC. This Center is funded by a Chinese National 111 Project Attracting. This work is also supported by the Macao Science and Technology Development Fund (081/2015/A3).

**References**

[1] Tesmer M, Perez C A, Zurada J M. Normalized mutual information feature selection[J]. IEEE Transactions on Neural Networks, 2009, 20(2):189.

[2] Liang J, Wang F, Dang C, et al. A group incremental approach to feature selection applying rough set technique[J]. IEEE Trans. Knowl. Data Eng., 2014, 26(2): 294-308.

[3] Peng H, Long F, Ding C. Feature Selection Based on Mutual Information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy[J]. IEEE Trans Pattern Anal Mach Intell, 2005, 27(8):1226-1238.

[4] Ding C, Peng H. Minimum redundancy feature selection from microarray gene expression data.[J]. Journal of Bioinformatics & Computational Biology, 2005, 3(02):185-205.

[5] Doquire G, Verleysen M. Feature selection with mutual information for uncertain data[C]//International Conference on Data Warehousing and Knowledge Discovery. Springer, Berlin, Heidelberg, 2011: 330-341.

[6] François D, Rossi F, Wertz V, et al. Resampling methods for parameter-free and robust feature selection with mutual information[J]. Neurocomputing, 2007, 70(7-9): 1276-1288.

[7] Battiti R. Using mutual information for selecting features in supervised neural net learning[J]. IEEE Transactions on neural networks, 1994, 5(4): 537-550.

[8] Rossi F, Lendasse A, François D, et al. Mutual information for the selection of relevant variables in spectrometric nonlinear modelling[J]. Chemometrics and intelligent laboratory systems, 2006, 80(2): 215-226.

[9] Huang J, Cai Y, Xu X. A wrapper for feature selection based on mutual information[C]//Pattern Recognition, 2006. ICPR 2006. 18th International Conference on. IEEE, 2006, 2: 618-621.

[10] Bell D A, Wang H. A formalism for relevance and its application in feature subset selection[J]. Machine learning, 2000, 41(2): 175-195.

[11] Kojadinovic I. Relevance measures for subset variable selection in regression problems based on k-additive mutual information[J]. Computational Statistics & Data Analysis, 2005, 49(4): 1205-1227.

[12] Kwak N, Choi C H. Input feature selection for classification problems[J]. IEEE transactions on neural networks, 2002, 13(1): 143-159.

[13] Yu L, Liu H. Efficient feature selection via analysis of relevance and redundancy[J]. Journal of machine learning research, 2004, 5(Oct): 1205-1224.

[14] Gao Y F, Li B Q, Cai Y D, et al. Prediction of active sites of enzymes by maximum relevance minimum redundancy (mRMR) feature selection[J]. Molecular BioSystems, 2013, 9(1): 61-69.

[15] Vergara J R, Estévez P A. A review of feature selection methods based on mutual information[J]. Neural computing and applications, 2014, 24(1): 175-186.

[16] Chandrashekar G, Sahin F. A survey on feature selection methods[J]. Computers & Electrical Engineering, 2014, 40(1): 16-28.

[17] Cover T M, Thomas J A. Elements of information theory[M]. John Wiley & Sons, 2012.

[35] L. Yu, C. Ding and S. Loscalzo. (2008) Stable feature selection via dense feature groups. KDD’08, 803-811