AAAI 2011

Size Adaptive Selection of Most Informative Features *

Si Liu[1], **Hairong Liu**[2], **Longin Jan Latecki**[4], **Shuicheng Yan**[2], **Changsheng Xu**[1,3], **Hanqing Lu**[1]
[1] National Lab of Pattern Recognition, Institute of Automation, Chinese Academy of Science
[2] Department of Electrical and Computer Engineering, National University of Singapore
[3] China-Singapore Institute of Digital Media
[4] Department of Computer and Information Sciences, Temple University, Philadelphia, USA

Abstract

In this paper, we propose a novel method to select the most informative subset of features, which has little redundancy and very strong discriminating power. Our proposed approach automatically determines the optimal number of features and selects the best subset accordingly by maximizing the average pairwise informativeness, thus has obvious advantage over traditional filter methods. By relaxing the essential combinatorial optimization problem into the standard quadratic programming problem, the most informative feature subset can be obtained efficiently, and a strategy to dynamically compute the redundancy between feature pairs further greatly accelerates our method through avoiding unnecessary computations of mutual information. As shown by the extensive experiments, the proposed method can successfully select the most informative subset of features, and the obtained classification results significantly outperform the state-of-the-art results on most test datasets.

Introduction

Many applications, such as text processing, gene expression array analysis, and combinatorial chemistry, are characterized by high dimensional data, but usually only a small subset of features is really important. Feature selection (Guyon and Elisseeff 2003; Jain and Zongker 1997) is thus preferred. Feature selection can enhance subsequent classifiers's generalization capability and remarkably speed up learning and classification process. Moreover, it improves model interpretability and significantly reduces storage requirements.

Among all feature selection methods, information theoretic filter (ITF) has received much attention due to its close relationship with Bayes error rate by Fano's inequation (Fano 1961), and mutual information (Shannon 1948) is the most frequently used criterion for ITF methods. For two random variables X and Y, their mutual information is denoted by $I(X;Y)$. If X is a feature vector and Y is its corresponding label vector, then $I(X;Y)$ reflects feature X's informativeness; if both X and Y are feature vectors, $I(X;Y)$ then measures the redundancy between the two features. Our proposed method belongs to the ITF method, and it defines a mutual information related statistical criterion to rank features. In mathematical form, for a dataset with N features denoted as $\mathcal{X} = \{X_1, \ldots, X_N\}$, the goal of our method is to select a Most Informative Subset (MIS) of features. We denote the selected MIS of size n as $S = \{X_{m(1)}, \ldots, X_{m(n)}\}$, where $m(\cdot)$ is a mapping function from the MIS index to the index of the original N features.

Existing ITF methods have two widely acknowledged problems. First, the number of selected features need to be specified in advance. In real applications, it is hard to estimate the number of useful features before the feature selection process. A common strategy is to use the wrapper method (Boull'e 2007), which determines the useful feature subset by a built-in classifier. However, the built-in classifier will severely slowdown the training process and result in the selected feature subset dependent on particular classifier setting. Second, all traditional ITF methods mine the useful feature subset in a greedy/incremental way (Brown 2009): an empty feature pool is constructed first, then features are added into the pool one by one until the user-defined number is reached. The basic assumption is *the best features till now are among the best subset forever*. However, this assumption can be easily violated. Some methods try to handle this problem, such as Plus-l-TakeAway-r and its extension Sequential Floating Search (Jain and Zongker 1997). However, these methods only partially solve this problem and bring in additional parameters.

In this paper, we tackle the above mentioned problems from a global perspective. In the proposed approach all informative features are selected jointly and simultaneously. We require the selected features in S to be *jointly* informative. Measuring joint informativeness involves high-order correlation between the set S and the label vector Y. However, directly estimating such a high-order correlation is difficult due to the scarcity of training data in most cases and is computationally intractable. To balance the computational cost and effectiveness of feature selection, only up to second-order correlation is considered. As mentioned above, the correlation of a feature pair $\{X_i, X_j\}$ is estimated through

*This work was done when Si Liu was intern at NUS.

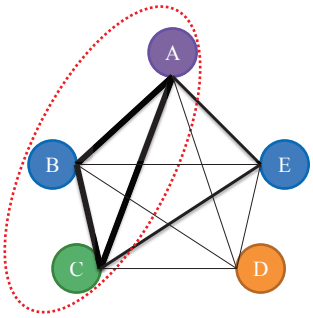


Figure 1: Feature graph. Each node indicates a feature, and each edge represents the informativeness of a feature pair. The selected most informative features are enclosed with the red ellipse.

mutual information , and it is defined as

$$W_{i,j} = \frac{1}{2} I\left(X_i; Y\right) + \frac{1}{2} I\left(X_j; Y\right) - I\left(X_i; X_j\right). \quad (1)$$

$W_{i,j}$ is large if and only if both $I\left(X_i; Y\right)$ and $I\left(X_j; Y\right)$ are large (indicating both X_i and X_j are informative themselves with respect to the class labels Y) and $I\left(X_i; X_j\right)$ is small (indicating X_i and X_j are not redundant). Therefore, the problem we consider can be expressed as selecting $n \leq N$ features such that the average pairwise informativeness is maximized, that is,

$$\begin{aligned} \max_{t,n} \quad & Q_{t,n} = \frac{1}{n^2} t^T W t, \\ s.t. \quad & t_i \in \{0,1\}, |t| = n, \end{aligned} \quad (2)$$

where t is an N-dimensional indicator vector such that $t_i = 1$ if i-th feature is selected and $t_i = 0$ otherwise. $|t|$ denotes ℓ_1-norm of vector t and requiring the ℓ_1-norm of t to be n aims to constrain the MIS size to be n. The division by $1/n^2$ in the objective function ensures that it is the average pairwise informativeness that is maximized. For N features, there are $N\left(N-1\right)/2$ feature pairs, therefore the computation of all mutual information $I(X_i, X_j)$ is usually time-prohibitive. However, in our method, only a very small subset of pairwise informativeness calculation is required, which greatly reduces the computational burden and makes our proposed method very efficient.

Formulation (2) has an intuitive geometric explanation as shown in Figure 1. There we have 5 nodes, A, B, C, D and E, with each node corresponding to a feature. A, B, C and E are discriminative features, but B and E have high redundancy, and B is slightly more discriminative than E. D is an irrelevant feature with almost no discriminating power. In Figure 1, redundant features are filled with the same color, such as blue nodes B and E. The edge thickness is proportional to the value of pairwise informativeness. Therefore, the edges from irrelevant feature D to other features are all thin. The selected subset of informative features is marked with the red ellipse, where both the redundant feature E and the irrelevant feature D are not included. Even though both B and E are informative features, they are not pairwise-informative because of the high redundancy between them. This is why only one of them is selected. Intuitively speaking, the formulation (2) is equivalent to searching for a dense subgraph (Gibson, Kumar, and Tomkins 1999) defined as the subgraph with the largest average edge weight.

In summary, our contributions can be summarized as follows. (I) We propose to select the most informative features by maximizing features' average pairwise informativeness. Different from other methods, the size of the optimal subset is determined automatically in the proposed method. (II) We relax the combinational feature subset selection task to a constrained quadratic optimization problem, and propose an iterative solution whose convergence is guaranteed. (III) Although many pairwise informativeness values are involved in the objective function, only a small percentage of them (usually $< 1\%$ in large datasets) need to be calculated, which greatly speeds up the optimization. (IV) According to the extracted most informative feature subset, we can rank other features and get a complete feature ranking list, as what traditional feature selection methods achieve.

Problem Relaxation

Based on (1), we construct a feature informativeness matrix $W = (W_{ij})$ and set $W_{ii} = 0$, i.e., all diagonal entries of W are set to zero. Since it is difficult to solve (2) due to the binary constraint on the indicator vector t, our goal is to relax this constraint. We first replace vector t by $s = \frac{t}{n}$. Then the formulation (2) is equivalent to

$$\max_{s,n} \quad Q_{s,n} = s^T W s, \;\; s.t. \;\; s_i \in \left\{0, \frac{1}{n}\right\}, \; |s| = 1. \quad (3)$$

Since each coordinate s_i of s is nonnegative, $|s| = 1$ is equivalent to $\sum_{i=1}^{N} s_i = 1$. By relaxing s_i to be within the range of $[0, 1]$, we obtain the final formulation of the feature selection problem:

$$\max_{s} \quad Q_s = s^T W s, \;\; s.t. \quad s \in \Delta^N, \quad (4)$$

where $\Delta^N = \{s \mid s_i \geqslant 0, \forall i \text{ and } \sum_{i=1}^{N} s_i = 1\}$ is the standard simplex in the N-dimensional Euclidean space. By relaxing Eq. (2) to Eq. (4), the maximum over original two variables, t and n, is replaced with the maximum over a single variable s. Once the solution s^* of (4) is obtained, we can easily recover the number of the selected features n and the index of the selected features in MIS: a feature X_i is selected if and only if $s_i^* > 0$. Consequently, the number of selected features n is determined by the number of positive coordinates of s^*.

Since $W_{ij} = \frac{1}{2} I\left(X_i; Y\right) + \frac{1}{2} I\left(X_j; Y\right) - I\left(X_i; X_j\right)$ for $i \neq j$, the objective function (2) can be expanded as:

$$\begin{aligned} & Q_{t,n} = \max_{t,n} \frac{1}{n^2} t^T W t \\ & = \max_{t,n} \frac{1}{n^2} \left((n-1) \sum_{t_i \neq 0} I\left(X_i; Y\right) - \sum_{t_i \neq 0} \sum_{t_j \neq 0, j \neq i} I\left(X_i; X_j\right) \right) \\ & \approx \max_{t,n} \frac{1}{n\left(n-1\right)} \left((n-1) \sum_{t_i \neq 0} I\left(X_i; Y\right) - \sum_{t_i \neq 0} \sum_{t_j \neq 0, j \neq i} I\left(X_i; X_j\right) \right) \\ & = \max_{t,n} \frac{1}{n} \sum_{t_i \neq 0} I\left(X_i; Y\right) - \frac{1}{n\left(n-1\right)} \sum_{t_i \neq 0} \sum_{t_j \neq 0, j \neq i} I\left(X_i; X_j\right) \end{aligned} \quad (5)$$

The first term in the last row of Eq. (5) is the average informativeness of each feature, and thus describes the discriminating power of the selected feature subset. The second term is the average redundancy between each feature pair, which is minimized for the compactness of the final feature subset.

Pairwise Optimization

In this section, we first analyze the properties of the maximizer s^* in (4), which are critical for algorithm design, and then introduce our algorithm to calculate s^*.

Since (4) is a constrained optimization problem, by adding Lagrangian multipliers λ and $\beta_1, \cdots \beta_N$ with $\beta_i \geqslant 0$ for all $i = 1, \cdots, N$, we obtain its Lagrangian function:

$$L(s, \lambda, \beta) = Q_s - \lambda \left(\sum_{i=1}^{N} s_i - 1 \right) + \sum_{i=1}^{N} \beta_i s_i. \quad (6)$$

Any local maximizer s^* must satisfy the Karush-Kuhn-Tucker (KKT) condition (Kuhn and Tucker 1951), i.e., the first-order necessary conditions for local optimality. That is,

$$\begin{cases} (Ws^*)_i - \lambda + \beta_i = 0; \\ \sum_i s_i^* \beta_i = 0. \end{cases} \quad (7)$$

Since s_i^* and β_i are both nonnegative, $\sum_i s_i^* \beta_i = 0$ is equivalent to say that if $s_i^* > 0$, then $\beta_i = 0$. Hence, the KKT conditions can be rewritten as:

$$(Ws^*)_i \begin{cases} \leqslant \lambda, & s_i^* = 0; \\ = \lambda, & 0 < s_i^* \leqslant 1. \end{cases} \quad (8)$$

We define the *reward* of feature X_i as $r_i(s) = (Ws^*)_i$. According to Eq. (8), there exists a constant λ such that the rewards of all selected features are equal to λ and the rewards of unselected features are not larger than λ. Higher reward indicates more informative feature. According to the value of s, all features X fall into three disjoint subsets, $P_1(s) = \{X_i | s_i = 0\}$, $P_2(s) = \{X_i | s_i \in (0, 1)\}$ and $P_3(s) = \{X_i | s_i = 1\}$. The set of the variables s_i which are smaller than 1 is $U = P_1(s) \cup P_2(s)$ and the set of nonzero variables is $V = P_2(s) \cup P_3(s)$.

If the objective function can be improved, to ensure the solution inside a simplex, our strategy is to add a constant value $0 < \alpha < 1$ to one variable belonging to U and at the same time, subtract α from some variable in V. According to KKT condition, if s^* is the optimal solution, then $r_i(s^*) \leqslant r_j(s^*), \forall i \in U, \forall j \in V$. On the contrary, if $\exists i \in U, \exists j \in V, r_i(s) > r_j(s)$, then s is not the solution. In fact, in such case, we can increase s_i and decrease s_j to increase $Q(s)$ by

$$s' = \begin{cases} s_l, & l \neq i, l \neq j; \\ s_l + \alpha, & l = i; \\ s_l - \alpha, & l = j. \end{cases} \quad (9)$$

Our goal is to find α such that $Q(s') - Q(s) > 0$. Since

$$Q(s') - Q(s) = (W_{ii} + W_{jj} - 2W_{ij}) \alpha^2 + 2(r_i(s) - r_j(s)) \alpha, \quad (10)$$

Algorithm 1 Size Adaptive Selection of Most Informative Features (SASMIF)

1: **Input:** Set $\mathcal{X}$ of all features and an initialization $s(0)$.
2: **while** s is not a local maximizer **do**
3: Check all entries of $W(i,j)$, where $j \in \{k | s_k(t) \neq 0\}$ and $i = 1, \ldots, N$. If $W(i,j)$ has not been calculated, then calculate it.
4: Compute the reward $r_i(s)$ for each feature X_i based on W;
5: Compute $P_1(s), P_2(s), P_3(s), U$ and V;
6: Find the feature X_i with the largest reward in U and X_j with the smallest reward in V;
7: Compute α by formula (11) and update $s(t)$ by formula (9) to obtain $s(t+1)$;
8: **end while**
9: **Output**: The selected feature subset corresponding to the non-zero elements of s and unselected features are ranked by $r_i(s)$.

which is a quadratic function of α, it is sufficient to set

$$\alpha = \begin{cases} \min(s_j, 1 - s_i), & f \geqslant 0; \\ \min\left(s_j, 1 - s_i, -\frac{d}{f}\right), & f < 0, \end{cases} \quad (11)$$

with $f = W_{ii} + W_{jj} - 2W_{ij}$ and $d = r_i(s) - r_j(s)$.

Since (4) is non-convex and usually has many local maximizers, we propose a heuristic initialization strategy: $s(0)$ is an N-dimensional vector, whose i-th element is one and others are zeros. Here, i corresponds to the feature with the highest mutual information $I(X_i; Y)$. In the proposed algorithm, we iterate (9) until $r_i(s) \leqslant r_j(s), \forall i \in U, \forall j \in V$. The algorithm is summarized in Algorithm 1. Intuitively, Algorithm 1 iteratively chooses the "best" feature in U and the "worst" feature in V and then updates their corresponding components of s. Hence in each iteration, we only need to consider two components of s, which makes both the update of rewards and the update of $s(t)$ very efficient. As $Q_s(s(t))$ increases, the number of candidate pairs for the operation in (9) decreases quickly, thus $Q_s(s)$ converges to a local maximum quickly. Suppose the number of iteration is T, and the computational complexity of each iteration is proportional to the number of edges E in a very sparse feature graph, so the total computational complexity is $O(TE)$.

Dynamic Edge Calculation When the feature dimension is large, estimating all entries in W becomes time consuming. Note that it is unnecessary to calculate every entry of W beforehand. Actually, W only affects features' reward computing. According to the definition of reward, in the t-th round, feature X_i's reward $r_i(s(t)) = (Ws(t-1))_i$ can be interpreted as the weighted pairwise informativeness between feature X_i and the selected feature set in the $(t-1)$-th round. The weighting coefficient is $s(t-1)$. In most cases, the selected feature set is very small, only the entries of columns of W need to be computed that correspond to nonzero entries in vector $s(t-1)$. Additionally, to avoid multiple unnecessary calculations, the value of W_{ij} will be calculated only when it has never been computed before.

Table 1: Datasets used in our experiments.

dataset	b_c	hep	ion	spa	LYM	B_T	leu	L_C	14_T
fea-num	10	19	34	57	4026	5920	5327	12600	15009
sam-num	683	80	351	4601	96	90	72	203	308

Complete Feature Ranking SASMIF focuses on mining a compact MIS which is composed of $\{X_j \left| s_{\mathrm{j}}^* \neq 0\right.\}$. The features inside the MIS are both highly discriminative themselves and with the minimum level of mutual redundancy. Based on MIS, we can calculate the rewards of each feature X_u inside unselected features set $U = \{X_u \left| s_{\mathrm{u}}^* = 0\right.\}$ as:

$$r_u\left(s^*\right) = \left(Ws^*\right)_u = \sum_{s_j^* \neq 0} W_{uj} s_j^* = \sum_{X_j \in \mathrm{MIS}} W_{uj} s_j^*, \quad (12)$$

which summarizes the pairwise informativeness between feature X_u and each feature inside the MIS. Higher reward indicates more informative feature. Therefore, $r_u\left(s^*\right)$ is used as a natural measure to rank remaining features $X_u \in U$. Consequently, we can obtain a complete feature ranking list, which starts from the size of MIS and ends at any user-specified fixed number.

Experimental Evaluation

In this section, we compare SASMIF with other state-of-the-art methods for feature selection on many real datasets.

Experimental Settings

We compare the proposed SASMIF with seven other baseline methods: mRMR (Peng, Long, and Ding 2005), MIFS (Battiti 1994), ReliefF (Kira and Rendell 1992), MIM (Brown 2009), Pearson's correlation (PC), JMI (Yang and Moody 1999) and CIFE (Lin and Tang 2006). All baseline methods belong to filter methods. We do not compare the proposed SASMIF with supervised methods, such as wrapper (Zhang 2008) or embedded (Bach 2008). The results of all methods are estimated by leave-one-out cross validated errors when using an SVM classifier with the features selected by these methods. We run linear SVM with LIBSVM (Chang and Lin 2001). The SVM regularization parameter is set to 1. To calculate the mutual information, we first quantify each feature X_i into 3-bin discrete variable $\tilde{X}_i$ as Peng did (Peng, Long, and Ding 2005). The quantitative thresholds are $\bar{X}_i - 0.5\sigma\left(X_i\right)$ and $\bar{X}_i + 0.5\sigma\left(X_i\right)$, where $\bar{X}_i$ and $\sigma\left(X_i\right)$ are the mean and standard variation of X_i respectively. Then the mutual information is calculated as $I\left(X_i; Y\right) =$

$$\sum\nolimits_{y \in Y} \sum\nolimits_{x \in \tilde{X}_i} p\left(x, y\right) \log\left(p\left(x, y\right) / \left(p_1\left(x\right) p_2\left(y\right)\right)\right),$$

where $p\left(x, y\right)$ is the joint probability distribution function of X and Y, and $p_1\left(x\right)$ and $p_2\left(y\right)$ are the marginal probability distribution functions of X and Y respectively. $I\left(X_i; X_j\right)$ can be estimated similarly.

Datasets

We run experiments on 9 datasets as shown in Table 1: breast-cancer (abbr. b_c), hepatitis (hep), ionosphere (ion), spambase (spa), LYM, Brain_Tumor1 (B_T), Leukemia1 (leu), LungCancer (L_C) and 14_Tumors (14_T). All of the 9 datasets are publicly available, b_c, hep, ion and spa are from the UCI repository[1]; LYM is microarray gene expression data sets[2]; B_T, leu, L_C and 14_T are cancer diagnosis datasets[3]. Table 1 lists the total number of original features (denoted by fea-num) and the sample numbers (represented by sam-num) in each dataset. Note that in many datasets, the number of features is much larger than the number of the data points, which makes the feature selection task challenging. Among these data sets, 4 are relatively small with less than 100 features; the remaining 5 datasets have above 4000 features each. Datasets with such wide dimension ranges serve as a good platform for a comprehensive evaluation.

Results and Analysis

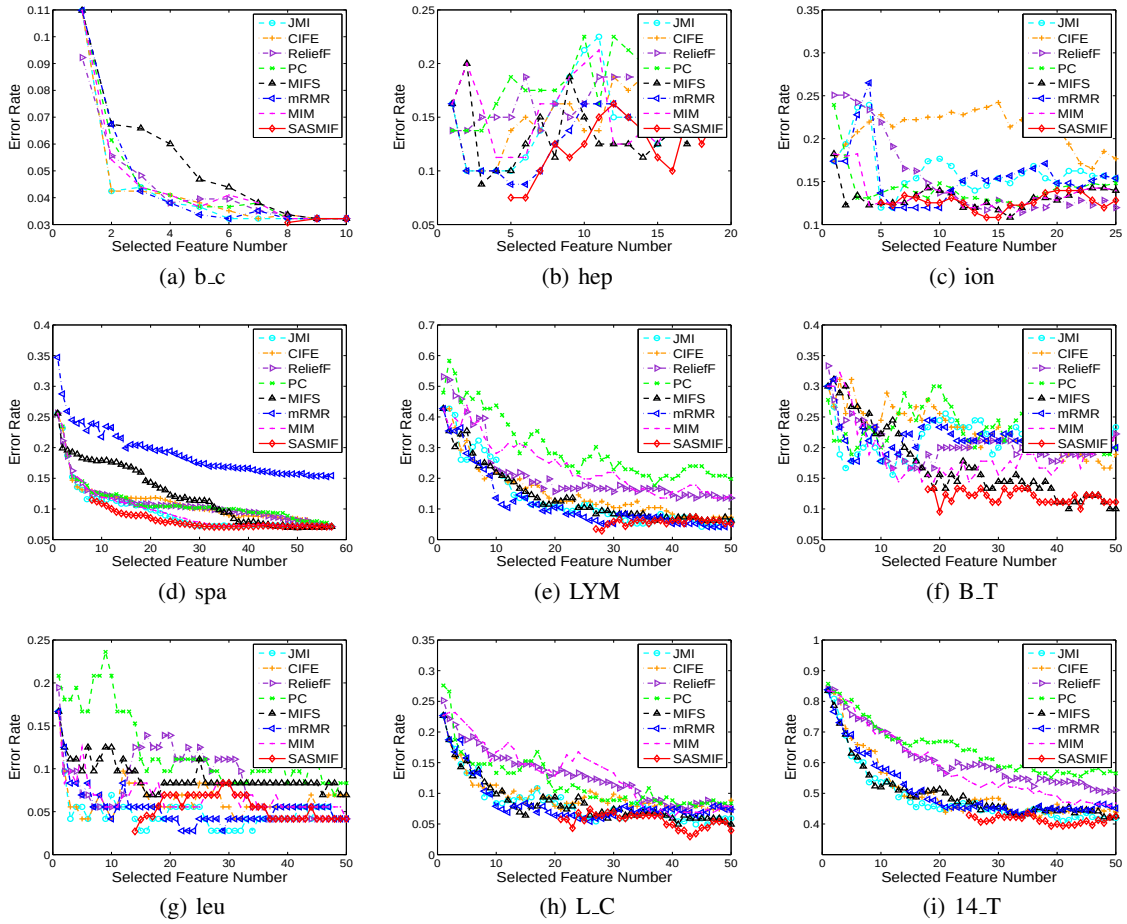
Figure 2 illustrates the error rate as a function of the number of features selected. The red lines with diamond markers is the proposed SASMIF. Traditional filtering methods cannot automatically determine the number of useful features, thus exhaustively guess the number from 1 to a pre-defined fixed number; however, our proposed SASMIF automatically determines the size of MIS and thus start from this number. As Figure 2 shows, in most cases, our method achieves the best results at the MIS number, and also outperform the best results of other baseline methods, such as in Figure 2 (a, b, e, g), which verifies that our method can automatically determines the most informative feature subset. In some cases, adding some top ranked features outside MIS can further improve the results, such as in Figure 2 (d), this is because our proposed method just selects most informative features, and may miss some features that are a little informative. In such case, the feature ranking list obtained by our method provides a natural order to add more features.

For clear comparison, we summarize the classification error rates of different methods in Table 2. In the last row, the error rate of SASMIF and the automatically determined MIS number are reported. To make a fair comparison, suppose that the MIS number is n, for each baseline method, we measure three error rates around n (including $n-1$, n, and $n+1$), and take the minimum of the three as the baseline performance. For each dataset, the best result of all methods is emphasized in bold. As demonstrated in the table, on $8/9$ datasets, SASMIF achieves better performance around MIS number. As shown in the last column of Table 2, our algorithm reaches the lowest average error rate across different datasets. The results further verify that SASMIF can select more informative feature subset than baselines if the number of selected features is around MIS number. The improvement mainly derives from the dynamic feature selection mechanism, i.e., SASMIF iteratively increases and decreases feature selection probabilities until reaching a steady state, while existing ITF methods only add features.

[1] http://www.ics.uci.edu/ mlearn/MLRepository.html

[2] http://penglab.janelia.org/proj/mRMR/index.htm

[3] http://www.gems-system.org/



(a) b_c (b) hep (c) ion

(d) spa (e) LYM (f) B_T

(g) leu (h) L_C (i) 14_T

Figure 2: The error rate of different algorithms with respect to the number of features selected.

Based on the mined MIS, we rank remaining features. The best results of all methods are shown in Table 3. In the table, the error rate is shown first and the number of features selected when reaching the best performance is reported in the following bracket. The best result for each dataset is highlighted in bold. Overall, SASMIF reaches the best result on $8/9$ datasets. Even though MIFS gets the best performance with 50 features on spa dataset, SASMIF achieves comparable result with much smaller number of features, i.e., only 33 features. What's more, it is apparent from the last column that average error rate of SASMIF is also much lower than other 7 baselines. The relative improvement from the best baseline to SASMIF is shown in the last row, in terms of error rate, which validates that SASMIF can select features that substantially reduce the error rate over the state-of-the-art approaches. The improvement is particularly obvious for the dataset LYM and L_C, where the error rate have been reduced by 31.4% and 40.0%, respectively.

In order to illustrate the computational benefits of the dynamic edge calculation strategy, we report the percentage of computed entries of W for large database in Table 4. We can find that in most datasets, we calculate less than one percent of entries of W. For example, 14_Tumors (14_T) has 15009 features, and consequently, W is of size 15009×15009, which is extremely large. Thus, the fact that only 0.36% elements of W are calculated means significant saving in time of computation.

Table 4: Percentage of entries of matrix W that were calculated.

Dataset	LYM	B_T	leu	L_C	14_T
Ratio(%)	1.77	0.90	0.81	0.47	0.36

Conclusion

We have proposed a method to automatically select the most informative feature subset. The number of selected features is automatically determined, depending on the data and label distribution. And only a small portion of mutual information is dynamically calculated, which makes the proposed method very efficient. Experimental results showed that the obtained most informative feature subset is compact yet sufficient for classification. Even though sometimes the classification results of the selected features are not the optimal, they are close to optimum with much fewer features. The mined MIS also lays a good foundation for a complete feature ranking as demonstrated by the experimental results. Until now, only up to second-order relations of

Table 2: Performance comparison of error rate (in %) at MIS number of different feature selection algorithms on different datasets. The best results are highlighted in bold.

dataset	b_c	hep	ion	spa	LYM	B_T	leu	L_C	14_T	avg
JMI	3.22	10.0	**12.0**	11.6	9.38	22.2	2.78	7.89	45.5	13.8
CIFE	3.22	10.0	21.4	12.6	11.5	24.4	8.33	7.39	44.2	15.9
ReliefF	3.22	15	19.0	12.6	16.7	16.7	9.72	13.3	58.5	18.3
PC	3.22	16.3	13.1	12.6	26.0	26.7	11.1	10.3	64.0	20.4
MIFS	3.22	10.0	12.3	17.8	8.33	13.3	8.33	7.39	46.5	14.1
mRMR	3.22	8.75	**12.0**	22.8	5.21	23.3	5.56	6.40	45.1	14.7
MIM	3.22	11.3	12.3	12.4	19.8	14.4	6.94	12.8	52.9	16.2
SASMIF	**3.07(8)**	**7.5(5)**	12.5(5)	**11.5(8)**	**3.48(27)**	**13.2(18)**	**2.70(14)**	**5.75(21)**	**42.9(25)**	**11.4**

Table 3: Performance comparison of minimum error rates (in %) of different feature selection algorithms on different datasets. The best results of each dataset are highlighted in bold. 'Improve' shows the improvements obtained by SASMIF in comparison the best baseline methods as percentage of error rate reduction.

dataset	b_c	hep	ion	spa	LYM	B_T	leu	L_C	14_T	avg
JMI	3.22(6)	10.0(2)	12.0(5)	7.11(57)	4.17(45)	15.6(12)	2.78(15)	4.92(44)	41.0(41)	11.2
CIFE	3.22(7)	10.0(2)	16.5(23)	7.08(55)	5.21(46)	16.7(48)	4.17(5)	6.90(42)	42.2(37)	12.4
ReliefF	3.22(9)	13.8(1)	11.4(17)	7.11(57)	13.5(41)	16.7(12)	4.17(37)	7.39(42)	50.3(48)	14.2
PC	3.22(8)	13.8(1)	12.0(17)	7.11(57)	17.7(37)	18.9(5)	6.94(48)	7.88(41)	55.2(39)	15.9
MIFS	3.22(9)	8.75(3)	**10.8(16)**	**6.89(50)**	6.25(37)	10.0(42)	6.84(16)	4.93(41)	41.9(49)	11.1
mRMR	3.22(6)	8.75(5)	12.0(6)	15.3(56)	4.17(46)	17.8(5)	2.78(22)	5.41(87)	43.2(34)	12.5
MIM	3.22(8)	11.3(4)	**10.8(16)**	7.11(57)	13.5(38)	14.4(13)	4.17(50)	5.91(37)	45.8(49)	12.9
SASMIF	**3.07(8)**	**7.50(5)**	**10.8(14)**	7.04(33)	**2.86(28)**	**9.52(20)**	**2.70(14)**	**2.95(43)**	**39.3(39)**	**9.52**
Improve	4.65%	14.3%	5.26%	-2.18%	31.4%	4.80%	2.88%	40.0%	4.15%	14.2%

features and corresponding labels are considered, while intuitively, higher-order relations may improve the feature selection. Our future work will focus on efficient representation of high-order relations among features, such as representation by hypergraphs.

Acknowledgments

This research is done for CSIDM Project No. CSIDM-200803 partially funded by a grant from the National Research Foundation (NRF) administered by the Media Development Authority (MDA) of Singapore. The work was also partially supported by NSF Grant IIS-0812118 and AFOSR Grant FA9550-09-1-0207.

References

Bach, F. 2008. Exploring large feature spaces with hierarchical multiple kernel learning. In *NIPS*.

Battiti, R. 1994. Using mutual information for selecting features in supervised neural net learning. *TNN* 5:537–550.

Boull'e, M. 2007. Compression-based averaging of selective naive bayes classifiers. *JMLR* 1659–1685.

Brown, G. 2009. A new perspective for information theoretic feature selection. In *AISTATS*.

Chang, C.-C., and Lin, C.-J. 2001. *LIBSVM: a library for support vector machines*.

Fano, R. 1961. Transmission of information: Statistical theory of communications.

Gibson, D.; Kumar, R.; and Tomkins, A. 1999. Discovering large dense subgraphs in massive graphs. In *VLDB*, 721–732.

Guyon, I., and Elisseeff, A. 2003. An introduction to variable and feature selection. *JMLR* 3:1157–1182.

Jain, A., and Zongker, D. 1997. Feature selection: Evaluation, application, and small sample performance. *TPAMI* 19:153–158.

Kira, K., and Rendell, L. A. 1992. A practical approach to feature selection. In *ML 1992*, 249–256.

Kuhn, H., and Tucker, A. 1951. Nonlinear programming. In Proceedings of 2nd Berkeley Symposium on Mathematical Statistics and Probabilistics.

Lin, D., and Tang, X. 2006. Conditional infomax learning: An integrated framework for feature extraction and fusion. In *ECCV*.

Peng, H.; Long, F.; and Ding, C. 2005. Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. *TPAMI* 27:1226–1238.

Shannon, C. 1948. A mathematical theory of communication. *Bell Syst. Tech. J* 27(3):379–423.

Yang, H., and Moody, J. 1999. Data visualization and feature selection: New algorithms for nongaussian data. In *NIPS*.

Zhang, T. 2008. Multi-stage convex relaxation for learning with sparse regularization. In *NIPS*.