Seventh IEEE International Conference on Data Mining - Workshops

Experimental Comparison of Feature Subset Selection Methods

Chulmin Yun and Jihoon Yang
Department of Computer Science, Sogang University, Seoul, Korea.
{cmyun, yangjh}@sogang.ac.kr

Abstract

In the field of machine learning and pattern recognition, feature subset selection is an important area, where many approaches have been proposed. In this paper, we choose some feature selection algorithms and analyze their performance using various datasets from public domain. We measured the number of reduced features and the improvement of learning performance with chosen feature selection methods, then evaluated and compared each method on the basis of these measurements.

1. Introduction

Feature selection is a topic that concerns selecting a subset of features, among the full features, that shows the best performance in classification accuracy [1]. The importance of feature selection in machine learning and pattern recognition comes from its ability of improving learning performance. This means that, through feature selection, we can reduce the cost of learning by reducing the number of features for learning, and provide a better learning accuracy compared to using full feature set.

Many feature selection algorithms have been proposed and discussed for many years. However, the problem of finding the optimal feature selection method for a certain data type or learning algorithm still remains to be a very basic, yet difficult problem. In order to find a solution of this problem, we made a selection of feature selection algorithms which were proposed and discussed recently, then compared each of them through a set of experiments in this paper.

The remainder of this paper is organized into five sections. In section 2 we briefly explain some feature selection methods. Section 3 explains each feature selection algorithms we chose and used. In section 4 and 5, we present our experimental setup and show the results. In the last section, we conclude our study and propose future work.

2. Characteristics of feature subset selection

Ideally, feature selection methods should choose the optimal feature subset from a candidate set to describe the target conceptions of a learning system. The following aspects must be considered in the process of feature selection [2].

2.1 Starting point

From the set of full features, first we must determine the starting point in feature space, which in turn influences the direction of search. The search for feature subsets can start with no features. Or, it can start with all (full) features. The first approach is called *forward selection*, and second is known as *backward elimination* [2]. We can also start the search from not only the two points explained above. A set with only half the number of the full feature set or a set with a random number of features can be good, as well.

2.2 Search strategy

Theoretically, the best subset of features can be found by evaluating all the possible subsets, which is known as exhaustive search. But an exhaustive search of the feature space needs to search all of 2^n possible subsets of n features, so it is almost always impractical when we meet large number of features. Therefore, we have to consider a more realistic and practical approach. Several search procedures that are easier to implement have been developed, but they are not guaranteed to find the optimal subset of features. These search procedures differ in their computational cost and the

† This work was supported by grant No. R01-2004-000-10689-0 from the Basic Research Program of the Korea Science & Engineering Foundation

0-7695-3019-2/07 $25.00 © 2007 IEEE
DOI 10.1109/ICDMW.2007.77

optimality of the subsets they find.

2.3 Subset evaluation

After generating subsets of candidate features, we need to evaluate them. One approach is, called the *filter method,* using some metric function measuring a feature's ability to discriminate the classes in data. Another approach is the *wrapper method* that uses a learning algorithm as the metric function. For each generated subset, it evaluates its goodness (e.g. classification accuracy) by applying the learning algorithm to the data [2].

2.4 Stopping criteria

Finally, we must decide the criteria for halting (stopping) the search [2]. For example, we can stop adding or removing features when none of the alternatives improves the estimate of classification accuracy, or we can stop when the number of selected features reaches a pre-determined threshold [3]. We can then choose the best subset among the candidates we have encountered during the search.

3. Feature selection methods in experiment

We chose eight feature subset selection methods proposed in the literature. We tried to choose the methods that are most up-to-date, and maintain as much variety in the approaches that the processes take as possible. All methods are available in public domain.

3.1 mRMR

The mutual information of two random variables is a quantity that measures the mutual dependence of the two variables. The mRMR method uses the mutual information between a feature and a class as *relevance* of the feature for the class [4]. And mRMR also uses the mutual information between features as *redundancy* of each feature.

The criterion combining above two conditions is called "*Minimal-Redundancy and Maximal-Relevance*" (mRMR). The mRMR measure has the following form to optimize D and R simultaneously :

$$\max \Phi(D,R), \quad \Phi = D - R$$

Where D and R means *relevance* and *redundancy* of each feature.

3.2 I-RELIEF

RELIEF is a well-known feature-weighting (ranking) approach that first introduced by Kira and Rendell [5]. The basic idea is to measure the relevance of features in the neighborhoods around target samples. For each target sample, RELIEF finds the nearest sample in feature space of the same category, called the "hit" sample, then measures the distance between the target and hit samples. It also finds the nearest sample of the other category, called the "miss" sample, and then does the same work. RELIEF uses the difference between those measured distances as the weight of target feature.

This basic algorithm was extended into several variants. In our experiments, we chose the iterative approach of RELIEF (I-RELIEF) which reduces the bias of original RELIEF [6].

3.3 CMIM

The main goal of *Conditional Mutual Information Maximization* (CMIM) is to select a feature subset that carries maximum relevance to the target class by using conditional mutual information [7]. It works by the following iterative scheme. $v(k)$ stands for the feature number of the k th feature in selected feature subset $\{X_{v(1)},...X_{v(l)}\}$ (full features in dataset are shown $\{X_1, X_2,...X_n\}$) :

$$v(1) = \arg\max_n \hat{I}(Y; X_n)$$

$$v(k+1) = \arg\max_n \underbrace{\{\min_{l \le k} I(Y; X_n \mid X_{v(l)})\}}_{s(n,k)}$$

$I(Y; X_n \mid X_{v(l)})$ is the conditional mutual information between target class Y and feature X_n when feature $X_{v(l)}$ was already chosen. By taking the feature X_n with the maximum score $s(n,k)$, we ensure that the new feature is both more informative and different than the preceding ones, at least in terms of predicting Y. (The current version of CMIM requires that both the feature values and output classes be binary.)

3.4 Correlation Coefficient

The correlation coefficient evaluates how well an individual feature contributes to the separation of classes [8][9]. For a feature *i*, the ranking criteria is

$$c_i = \left| \frac{\mu_i(+) - \mu_i(-)}{\sigma_i(+) + \sigma_i(-)} \right|$$

where μ_i and σ_i are the mean and standard deviation of the feature expression values of feature *i* for all the samples of class (+) or class (–). Each feature is ranked by this value and the top ones are selected.

3.5 BW-ratio

The BW-ratio uses the ratio of between-group to within-group sums of squares for each feature, and selects features with the largest BW-ratios [10]. For a feature j , the ratio is

$$\frac{BSS(j)}{WSS(j)}=\frac{\sum_i\sum_k I(y_i=k)(\overline{x_{kj}}-\overline{x_j})^2}{\sum_i\sum_k I(y_i=k)(x_{ij}-\overline{x_{kj}})^2}$$

where $I()$ is the indicator function, $\overline{x_j}$ denotes the average value of feature j across all the training set and $\overline{x_{kj}}$ denotes the average value of feature j for class k .

3.6 INTERACT

INTERACT algorithm considers feature interaction [11]: A single feature can be considered irrelevant based its correlation with the class, but when combined with other features, it might become relevant. INTERACT finds interacting features by backward elimination with measurement of Consistency Contribution (C-contribution). C-contribution of a feature is an indicator about how significantly the elimination of that feature will affect consistency. (i.e., C-contribution of an irrelevant feature is zero.) Using backward elimination, INTERACT starts with the full feature set and successively eliminates features one at a time based on their C-contributions: If C-contribution of a feature is less than the threshold δ (predefined sufficiently small value), that feature is removed from feature set.

3.7 Genetic Algorithm

Genetic Algorithm (GA) is a well-known randomized approach. It is a particular class of evolutionary algorithms that makes use of techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover.

In feature selection problems, each feature subset is represented by a binary string [12]. 1 of N th bit means that the feature set contained feature X_n . A fitness function is a particular type of objective function that quantifies the optimality of a solution in a genetic algorithm. We use the wrapper approach in experiment. That is, we will measure fitness function by the accuracy of learning algorithms.

3.8 SVM-RFE

SVM-RFE (Recursive Feature Elimination) is a wrapper method which performs backward feature elimination [13]. The idea is to find the *m* features which lead to the largest margin of class separation, and uses the weight vector *w* as a ranking criterion. The recursive elimination procedure of SVM-RFE is implemented as follows:

```
1. Start: ranked feature set R = []; selected feature subset S = [1,...,d];
2. Repeat until all features are ranked:
    a) Train a linear SVM with features in set S as input variables;
    b) Compute the weight vector;
    c) Compute the ranking scores for features in set S : $c_i = (w_i)^2$
    d) Find the feature with the smallest ranking score : $e = \arg\min_i(c_i)$
    e) Update: R=[e,R], S=S-[e];
3. Output: Ranked feature list R.
```

The algorithm can be generalized to remove more than one feature per step for speed up.

3.9 PAM

PAM (Prediction Analysis of Microarrays) is a method which uses the nearest shrunken centroid methodology [14]. This method computes a standardized centroid for each class and determine d_{ik} , which is a t -statistics for feature i comparing class k to the overall centroid

$$d_{ik}=\frac{\bar{x}_{ik}-\bar{x}_i}{s_i}$$

where $\bar{x}_{ik}$ is the mean expression value in class k for feature i (actually it is *i*th component of the centroid for class k), and $\bar{x}_i$ is the *i*th component of the overall centroid, and s_i is the within-class standard deviation for feature i .

Then this method shrinks each d_{ik} toward zero, and the shrinkage it uses is called *soft thresholding*. Each d_{ik} is reduced by an amount Δ and is set to zero if its absolute value is less than zero. For a feature i, if d_{ik} is shrunken to zero for all classes k, then the centroid for feature i is $\bar{x}_i$, the same for all classes. It means that feature i does not contribute to the nearest-centroid classification, so it can be eliminated from the feature set. As the shrinkage parameter Δ is increased, many of the features are eliminated.

4. Experiments

4.1 Dataset

The seven data sets we used are summarized in Table 1. The first two data sets, lung cancer and leukemia, are well-known microarray data. And the other five data sets are from UCI machine learning archive [15]. All seven data sets are available on-line.

The two microarray data sets were directly used for continuous feature selection experiment. And the five of UCI data sets were preprocessed for discrete feature selection. Each feature variable was discretized into three states. We separated the range of each feature variable into three subsets of same fixed width. It assigns 1 if it is in the first subset, 2 if the second subset, and 3 if otherwise.

Also, in order to experiment with CMIM algorithm, we changed Arrhythmia, Sonar, and Ionosphere dataset into binary type. Each feature variable was discretized by comparison with the mean: As, the value, it takes 1 if it is larger than the mean value and 0 if otherwise. Classes of Arrhythmia dataset were discretized, as well: It takes 1 if it is normal, 0 if otherwise.

Note that the methods, Correlation Coefficient (CC), BW-ratio and SVM-RFE only operate on two-class datasets. Therefore, we exclude Image Segmentation and Waveform in the experiment of these methods.

Table 1. Datasets used in experiment

	# of Classes	# of Features	# of Samples
Lung	2	3916	32
Leukemia	2	12533	38
Arrhythmia	16 (2)	279	452
Sonar	2	60	208
Ionosphere	2	34	351
Image Segmentation	7	19	2310
Waveform	3	21	5000

4.2 Experimental setup

We used WEKA [16] to measure the performance of each feature selection algorithm. WEKA is a well-known machine learning tool based on JAVA. And we evaluated selected feature subsets using two learning algorithms – Naïve Bayes (NB) and LIBSVM algorithm (LIBSVM). We evaluated feature subsets using 10-fold Cross-Validation (CV) for five UCI data sets and using Leave-One-Out Cross-Validation (LOOCV) for two microarray data sets.

5. Results

5.1 Number of selected features

First, we consider the number of features reduced by feature selection methods. Reducing the number of features of dataset is important because it can decrease the complexity and reduce the learning time. Table 2 shows the number of features selected by each method, compared with full feature sets. As we can see clearly, all methods were successful in reducing the number of features.

5.2 Accuracy of classification

We measured the classification accuracy of datasets with full features first using the learning algorithms by 10-fold CV and LOOCV. Then we reorganized the datasets using selected (reduced) features by each method and evaluated each method by the same process. We measured the accuracy of those reorganized datasets, and calculated the difference of accuracies between the reorganized datasets and the full-featured datasets. Table 3 shows us the result. Each value in table shows increase or decrease of learning accuracy compared with full feature set. (+) means the best change (largest increase), and (-) means the worst change (least increase, or largest decrease).

Almost all of the methods produced better performance with reduced feature set than full features. In particular, mRMR method performed even better than other methods for most datasets with two learning algorithms. Also, INTERACT worked as good as mRMR with NB learning algorithm.

On the other hand, I-RELIEF method shows poor performance for some discrete data. This is because of the characteristics of the method. Relief methods use the distance of neighbor samples, so in discrete cases, the weight of each feature might have been biased.

GA also produced poor results. In contrast with the result of I-RELIEF, GA shows poor performance in microarray data sets.

Table 2. Number of selected features by feature selection methods

	mRMR	I-Relief	CC	BW-ratio	INTERACT	GA	SVMRFE	PAM
Lung	20	30	10	10	40	345	100	10
Leukemia	60	90	10	10	37	98	100	20
Arrhythmia	50	60	60	60	13	25	60	40
Sonar	40	30	5	5	5	9	20	5
Ionosphere	10	5	30	18	12	8	25	5
Segmentation	6	12	-	-	9	6	-	18
Waveform	15	12	-	-	19	15	-	15

Table 3. Change of learning accuracy compared with full feature set

NB								
	mRMR	**I-Relief**	**CC**	**BW-ratio**	**INTERACT**	**GA**	**SVMRFE**	**PAM**
Lung	3.13(+)	-3.13	3.13(+)	3.13(+)	3.13(+)	-6.25(-)	2.79	3.13(+)
Leukemia	5.26(+)	2.63	5.26(+)	5.26(+)	5.26(+)	-5.26(-)	0.41	5.26(+)
Arrhythmia	11.28(+)	6.28	4.20	3.76(-)	9.73	5.75	5.46	10.13
Sonar	1.92	5.76(+)	5.76(+)	5.76(+)	5.76(+)	1.92	1.92	3.85
Ionosphere	4.55	-6.26(-)	0.28	-4.84	5.68(+)	4.27	3.13	4.05
Segmentation	2.55	0.86(-)	-	-	4.37(+)	2.90	-	0.97
Waveform	0.04	-3.88(-)	-	-	0.16(+)	0.04	-	0.04
LIBSVM								
	mRMR	**I-Relief**	**CC**	**BW-ratio**	**INTERACT**	**GA**	**SVMRFE**	**PAM**
Lung	0	-3.12(-)	-1.97	-0.93	0	-3.12(-)	-0.34	0
Leukemia	24.39(+)	16.50	21.76	19.13	19.17	-1.91(-)	22.17	24.39(+)
Arrhythmia	11.17(+)	8.48	5.30	4.20(-)	10.84	8.84	9.89	10.57
Sonar	1.92(+)	-1.92	0	0.48	-0.96	-3.36(-)	1.44	0.96
Ionosphere	0	-2.56(-)	-0.56	0	-1.13	-1.13	0	-1.13
Segmentation	1.34	1.34	-	-	1.94(+)	-1.47(-)	-	0.30
Waveform	-0.38	-5.64(-)	-	-	-0.06(+)	-0.38	-	-0.18

5.3 Result of binary case

To compare performance of CMIM method with other methods, we made binary datasets from the three UCI datasets, Arrhythmia, Sonar and Ionosphere. We conducted the same experiments with those datasets. As expected, all methods reduced the number of features (Table 4). However, the learning performance with selected features did not improve in all methods. While CMIM and mRMR produced improved learning performance, other methods did not. Especially I-RELIEF method produced incomparably lower learning performance than others. This might be due to the bias problem, which we mentioned in the previous result with discrete datasets. We showed the result of learning performance with binary case in Table 5.

6. Conclusion and future work

We compared nine different feature selection methods which were newly proposed and tested with public data. Most of the methods have shown improved performance in terms of the feature size and the classification accuracy. In particular, the result of mRMR method demonstrated the most powerful and stable performance over all the other methods we considered. Meanwhile, some methods such as I-RELIEF were unstable, dependent on the data or the learning algorithm.

For future work we plan to fine-tune the settings of our experiment, and include comparison of execution time of each method. This is because each method needs to be compared with its results obtained by the best parameter setting, as well as with its actual running time for a fair comparison. Finally, we are planning to develop a novel feature subset selection method based on this experimental study.

7. References

[1] A. Jain and D. Zongker, "Feature Selection: Evaluation, Application, and Small Sample Performance", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1997, Vol. 19, No. 2, pp. 153-158.

Table 4. Number of selected features by feature selection methods in binary case

	mRMR	I-Relief	CC	BW-ratio	INTERACT	GA	SVMRFE	PAM	CMIM
Arrhythmia	65	60	50	70	12	38	85	45	55
Sonar	5	27	40	10	9	7	55	5	5
Ionosphere	5	11	25	5	10	7	25	5	10

Table 5. Change of learning accuracy compared with full feature set in binary case

NB									
	mRMR	**I-Relief**	**CC**	**BW-ratio**	**INTERACT**	**GA**	**SVMRFE**	**PAM**	**CMIM**
Arrhythmia	1.76	-12.21(-)	1.76	-0.22	2.65	-2.43	-0.90	-0.31	2.87(+)
Sonar	7.21(+)	-1.44	1.92	0.96	0	-2.40(-)	-1.44	-2.40	7.21(+)
Ionosphere	2.56(+)	-15.38(-)	0.85	0.85	0.85	2.56(+)	0.28	2.56(+)	0.85
LIBSVM									
	mRMR	**I-Relief**	**CC**	**BW-ratio**	**INTERACT**	**GA**	**SVMRFE**	**PAM**	**CMIM**
Arrhythmia	0.78(+)	-10.20(-)	-1.76	-1.76	0	-0.63	-0.25	-0.22	0.22
Sonar	3.37(+)	0.48(-)	2.40	1.9	2.40	3.32	3.32	2.32	3.36
Ionosphere	2.28	-5.12(-)	1.69	1.99	1.70	1.99	2.27	2.27	2.56(+)

[2] A.L. Blum and P. Langley, "Selection of Relevant Features and Examples in Machine Learning", *Artificial Intelligence,* 1997, Vol. 97, No. 1-2, pp. 245-271

[3] H. Liu and L. Yu, "Toward Integrating Feature Selection Algorithms for Classification and Clustering", *IEEE Transactions on Knowledge and Data Engineering,* 2005, Vol. 17, No. 4, pp. 491-502

[4] H. Peng, F. Long and C. Ding, "Feature Selection Based on Mutual Information : Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2005, Vol. 27, No. 8, pp. 1226-1238

[5] K.Kira and L.A. Rendell, "A Practical Approach to Feature Selection", *Proceedings of the 9th International Workshop*

[6] Y. Sun and J. Li, "Iterative RELIEF for Feature Weighting", *Proceedings of the 23rd International Conference on Machine Learning,* 2006, pp. 913-920

[7] F. Fleuret, "Fast binary feature selection with conditional mutual information", *Journal of Machine Learning Research,* 2004, vol. 5, pp. 1531-1555

[8] I. Guyon and A. Elisseeff, "An Introduction to Variable and Feature Selection", *Journal of Machine Learning Research,* 2003, Vol. 3, pp. 1157-1182

[9] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield and E. S. Lander, "Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring", *Science,* 1999, Vol. 286, No. 5439, pp. 531-537

[10] S. Dudoit, J. Fridlyand and T. P. Speed, "Comparison of Discrimination Methods for the Classification of Tumors Using Gene Expression Data", *Journal of the American Statistical Association,* 2002, Vol. 97, No. 457, pp. 77-87

[11] Z. Zhao and H. Liu, "Searching for Interacting Features", *Proceedings of International Joint Conference on Artificial Intelligence,* 2007, pp. 1156-1161

[12] M.J Martin-Bautista and M-A Vila, "A Survey of Genetic Feature Selection in Mining Issues", *Proceedings of the Congress on Evolutionary Computation,* 1999, Vol. 2, pp. 1314-1321

[13] I. Guyon, J. Weston, S. Barnhill and V. Vapnik, "Gene Selection for Cancer Classification using Support Vector Machines", *Machine Learning,* 2002, Vol. 46, No. 1-3, pp. 389-422

[14] R.Tibshirani, T.Hastie, B.Narasimhan and G.Chu, "Diagnosis of Multiple Cancer Types by Shrunken Centroids of Gene Expression", *Proceedings of the National Academy of Sciences of the United States of America,* 2002, Vol. 99, No. 10, pp. 6567-6572

[15] UCI Machine Learning Repository, http://www.ics.uci.edu/~mlearn/MLRepository.html

[16] WEKA, http://www.cs.waikato.ac.nz/~ml/index.html