Journal of Advances in Computer Research
Quarterly pISSN: 2345-606x eISSN: 2345-6078
Sari Branch, Islamic Azad University, Sari, I.R.Iran
(Vol. 8, No. 3, August 2017), Pages: 43-54
www.jacr.iausari.ac.ir

FFS: An F-DBSCAN Clustering- Based Feature Selection for Classification Data

Nasim Eshaghi✉[1], Ali Aghagolzadeh[2]

1) Department of Computer Engineering, Rouzbahan University, Sari, Iran.
2) Faculty of Electrical and Computer Engineering, Babol Noshirvani University of Technology, Babol, Iran

n.eshaghi@rouzbahan.ac.ir; aghagol@nit.ac.ir

Received: 2016/07/23; Accepted: 2016/09/07

Abstract

Feature selection is an important step in most classification problems to select an optimal subset of features to increase the learning accuracy and reduce the computational time. In this paper we proposed a new feature clustering based method to perform feature selection (FFS) in classification problems. The FFS algorithm works in two steps. In the first step, features are divided into clusters by using F-DBSCAN method. A novel F-DBSCAN clustering method used mutual information for measuring dependencies between features. In the second step, the most representative feature is selected from each cluster by a new criterion function. This allows us to consider the possible dependency on the target class and the redundancy between the selected features in each cluster. The experimental results on different datasets show that the proposed algorithm is more effective for feature selection in classification problems. Compared with the other methods, the average classification accuracy of C4.5, KNN and Naïve Bayes are improved using FFS by 8.05, 8.36 and 4.63 percent, respectively. Also, the results demonstrate that the FFS algorithm produces small subsets of features with very high classification rate.

Keywords: *Feature Selection; Mutual Information; Feature Clustering*

1. Introduction

In the past years in applications of machine learning and pattern recognition, the domain of features has been extended. Thus, feature selection is known to be an important preprocessing task for many pattern recognition applications, such as classification and clustering [1]. In datasets all of the features are not useful for classification, since dependent and redundant features provide no extra information about classes and sometimes some features are even noisy. Therefore, feature selection methods are used to determine an optimal feature subset [2]. Suitable feature subset may bring lots of advantages such as improving learning accuracy, avoiding over-fitting, distinguishing key features with unimportant ones and enhancing learning comprehensibility [3]. Due to these benefits, feature selection algorithm has wide applications, such as text classification [4], disease recognition [5] and bioinformatics [6].

Feature selection methods are usually classified into four categories namely: filter, wrapper, embedded and hybrid methods [7], [8]. Filter methods evaluate the importance of features by using feature ranking techniques as the principle criteria. A suitable ranking criterion is used to measure the quality of features. So far, many evaluation

criteria are designed, such as correlation and mutual information. Also, a threshold value is used which can remove features with less important [9]. Wrapper methods evaluate the goodness of a subset feature with a specific learning algorithm. The performance is usually measured in terms of the classification accuracy obtained on testing data. Wrapper methods can give high accuracy of the learning algorithms however the computational complexity is large [10]. Embedded methods want to reduce the computation time taken up for reclassifying different subsets which is done in wrapper methods. They perform feature selection in the process of training and are usually specific to given a learning algorithm [9]. The hybrid methods are a combination of filter and wrapper methods and attempt to take the advantages of filter and wrapper methods. They mainly focus on combining filter and wrapper methods to achieve the best possible performance with a particular learning algorithm and similar time complexity of filter methods [8].

Several feature selection algorithms [11], [12], [13] have been proposed for classification. The main idea of these algorithms is to select an optimal set of features by removing those of redundant or irrelevant features. Most feature selection algorithms use several criteria to evaluate feature subsets or a population-based optimization methods such as particle swarm optimization, [14] ant colony optimization,[15] gravitation search algorithm and genetic algorithms [16], [17]. Recently, some methods used clustering techniques for feature selection [18], [19].This methods partition the original feature set into some distinct subsets or clusters so that the features within a cluster are more correlated with each other whereas features in different clusters are less correlated [20], [21].

In cluster analysis, DBSCAN (Density-Based Spatial Clustering of Applications with Noise) method has been well studied and used in many applications. DBSCAN is a density based clustering containing large amount of data which has noise and outliers. Clusters are regarded as regions in which the objects are dense, and are separated by regions of low object density or noise [22], [23]. DBSCAN clustering algorithm has become popular rapidly and applied in different fields of science.

In our study, we apply DBSCAN clustering methods for clustering of features. In this paper, we propose a new feature selection algorithm based on F-DBSCAN feature clustering. First we partition features into clusters by using F-DBSCAN clustering method. Since mutual information is good at determining how much information is shared by two features, we use it to measure similarity of features in F-DBSCAN algorithm, so features in each cluster are more dependent to each other. Then, we use a special metric to choose the most representative features that are strongly related to target classes. The aim is to choose an optimal subset of features thereby reducing the number of features involved in the clustering process.

This paper is organized as follows. Some related works of feature selection are reviewed in section 2. The proposed method for feature selection and some corresponding definitions and algorithms are stated in section 3. The experiment of evaluation with different feature selection algorithms are shown in section 4. Conclusions are finally given in section 5.

2. Related Works

During past years, various feature selection algorithms have been witnessed. Here only the latest methods will be discussed briefly. The common approach for these

algorithms is to search for an optimal set of features that provides good classification result. Many feature selection algorithms use information theoretic measure such as entropy, mutual information and information gain to evaluate the feature-class relevance as well as feature-feature redundancy [24], [25]. Peng et al.[26] introduce a mutual information based feature selection method called mRMR that combines minimal-redundancy-maximal-relevance and wrappers. FCBF [27] is a fast filter method which can identify relevant features as well as redundancy among relevant features without pairwise correlation analysis. Hoque et al.[7] introduce a greedy feature selection method using mutual information called MIFS-ND. This method uses both feature-feature mutual information and feature-class mutual information to select a subset of features which are strongly relevant but non-redundant.

Feature clustering methods make features cluster together rather than instances. In this case, instances distance metric is replaced by feature similarity. Song et al. [8] introduce a clustering-based feature subset selection algorithm called FAST. The method consists of two steps. In the first step, features are divided into clusters by using graph-theoric clustering methods. In the second step, the most representative feature that is strongly related to target classes is selected from each cluster to form the final subsets of features. Au et al.[19] use a new method to cluster features for feature selection called ACA. ACA employs an information measure to evaluate the interdependence between features and groups the features that are dependent on each other into clusters. Liu et al.[20] propose a new filter feature selection method by using clustering with information metric. In this method, each feature is considered as a data point cluster with between-cluster and within-cluster distances. Maji [28] introduce a new quantitive measure, based on mutual information, to compute the similarity between features. The proposed measure incorporates the information of sample categories while measuring the similarity between features. As shown in recent researches, using a feature clustering-based method for the general feature selection problem is effective approach. Quite different from these clustering-based algorithms, our proposed FFS algorithm uses F-DBSCAN method to cluster features.

3. The Proposed Method

The proposed method uses information theory and F-DBSCAN clustering algorithm to select a subset of relevant features. In this method, an information-theoretic concepts based measure is utilized as the feature redundancy and relevance metric. Given an initial set of d features, $F=\{F_1,F_2,\ldots F_d\}$, and class label C, the proposed algorithm is to find out F' with m features, that minimizes the redundancy among features and maximizes the relevance between the feature set and class label. The proposed algorithm logically consists of two steps: 1) features clustering, 2) selecting representative features for each cluster.

3.1 Redundancy and Relevance Measure

The symmetric uncertainty (SU) is adopted to measure the redundancy among features and relevance between the feature set and the class label. It is calculated based on entropies of two features or feature-class and the mutual information between them. Given random variables X and Y, is defined as follows:

$$SU(X,Y) = \frac{2I(X,Y)}{H(X)+H(Y)} \quad (1)$$

Where the H(x) is the entropy of a random variable X. Suppose p(x) is the probability for all values of X, H(X) is defined by

$$H(x) = -\sum_{x \in X} p(x)\log_2 p(x) \quad (2)$$

I(X,Y) is the mutual information between variable X and Y, and is defined with the joint probability p(x,y) and prior probability p(x), p(y) as

$$I(x,y) = \sum_{x \in X}\sum_{y \in Y} p(x,y)\log_2\left(\frac{p(x,y)}{p(x)\,p(y)}\right) \quad (3)$$

Features in feature set F and target C both are treated as random variables. The symmetric uncertainty (SU) normalizes its value to the range [0, 1]. A value 1 of SU(X, Y) indicates that X and Y are strongly interdependent. On the other hand, the value 0 reveals that they are independent.

Therefore, some features holding great SU values with each other are considered as redundant. They can be replaced by one of them. If some feature has a great SU value with the target, it is relevant to the target.

3.2 Feature Clustering

It note that the process of feature selection works like data clustering, where each cluster contains the features, which are more relevant to classes and irrelevant to each other. Thus, we use the idea of clustering to serve for the purpose of feature selection. To this end, the F-DBSCAN clustering algorithm is applied to group the features into different clusters based on the features similarities. Also it can detect noisy features and remove them from the list of features. So the features within a cluster are more correlated with each other, whereas features in different clusters are less correlated.

To group features $F_1, F_2, \ldots, F_d$ into clusters, we build our information-theoric feature clustering algorithm by converting the popular DBSCAN into F-DBSCAN algorithm. Their differences are in data representation and distance measurement. For the individual clustering point, it is feature in our method, not data point. Also, we use mutual information instead of Euclidean distance measure. This algorithm requires the specification of two input parameters. The input parameters are the radius of cluster (Eps) and minimum required features inside the cluster (MinF). The basic idea in F-DBSCAN algorithm is as follows:

Difinition1. The **neighborhood** of a feature f_i, denoted as $N_e(f_i)$, is defined by

$$N_e(f_i) = \{ f_j \in D \,||\, MI(f_i, f_j) \leq eps \}$$

Difinition2. A feature is a **core feature** if it has more than a specified number of features (MinF) within Eps. These are features that are at the interior or a cluster.

Definition3. A **border feature** has fewer than MinF within Eps, but is in the neighborhood of a core feature.

Definition4. A **noise feature** is any feature that is neither a core feature nor a border feature.

Definition5. A feature f_j is directly density-reachable from a feature f_i when $f_j \in N_e(f_i)$

Definition6. A feature f_i is density-reachable if there is a chain of features f_1, f_2, ..., f_d=f_i such that f_j is directly density-reachable from f_{j+1}.

Algorithm 1. F-DBSCAN

Function F-DBSCAN (Dataset D, Eps , MinF)
For each $f_i \in$ D **do**
If f_i is not yet classified **then**
  If f_i is a core feature **then**
    Collect all features density-reachable from f_i and
    Assign them to a new cluster.
Else
Assign f_i to noise.

3.3 Representative Features Selection

In this subsection, a new feature evaluation function is proposed for measuring representative features. Designing a feature evaluation function to measure the quality of candidate features is an important issue in the feature selection process. Most of the existing evaluation functions consider a feature with the largest dependency on the target class as a representative feature, but not take into consideration the redundancy between the selected features. Hence a new evaluation function is defined here that combines maximum relevance and the minimum redundancy together for measuring candidate features.

After features clustering, in the second step, we select the representatives from each cluster so that the other features that are dependent to the representatives are eliminated. In order to select the representatives, we calculate an evaluation function for all features within a cluster. The evaluation function of the candidate feature f_i is defined by:

$$J(f_i) = \frac{SU(f_i, C)}{avg(SU(f_i, F)) + std(SU(f_i, F))} \tag{4}$$

where $avg(SU(f_i, F))$ and $std(SU(f_i, F))$ are the average and standard deviation of the SU between f_i and features within a other clusters.

Definition7. A feature $F_i \in C=\{F_1, F_2, ..., F_m\}$ ($m< |C|$) is a representative feature of the cluster C if and only if,

$F_i = \arg\max_{F_j \in C} J(F_j)$.

Then the features with greatest J value are selected as representative from each cluster. The representative features that have high relevance with the target classes and low redundancy will be selected in the feature selection process. In this way, a reasonable subset of features can be selected.

The details of the FFS algorithm are shown in Algorithm 2.

Algorithm 2. FFS algorithm

Input: d, the number of features, dataset D, F={f_1,f_2,…,f_d}, the set of features, , Eps , MinF
Output: F', an optimal subset of features
Steps:
[cluster]=F-DBSCAN(Dataset D,Eps,MinF)
Cand(F)←0
Count ←1
While Count ≤ k do
ClusterF← find(cluster==Count)
For all features ∈ ClusterF **do**
RepF ← argmax J(f)

Cand(F)←cand(F) ∪ {RepF}
End
End
Return features set F'

3.4 Example

An example shows how the algorithm works at feature selection. Let $F=\{F_1,F_2,F_3,F_4,F_5,F_6,F_7,F_8\}$ be a set of eight features. First, for a feature $F_i \in F$ we compute feature-feature mutual information with every feature $F_j \in F(i\neq j)$. Suppose every mutual information values shown in Fig.1(a). Next, in order to cluster the features we apply F-DBSCAN algorithm by Eps=1.08 and MinF=1. Here, $MI(F_1,F_j)(j=2,\ldots,8)$ is greater than 1.08 so we consider F_1 as a cluster. We also observed that $MI(F_2,F_4),MI(F_3,F_4),MI(F_4,F_5),MI(F_5,F_6),MI(F_4,F_8)$ is smaller than 1.08 and $MI(F_7,F_j)(j=1,\ldots,6)$ is greater than 1.08. As shown in Fig.1(b), it finds that the optimal number of clusters is three. F-DBSCAN identifies three clusters of features: $\{F_1\},\{F_2,F_3,F_4,F_5,F_6,F_8\}and\{F_7\}$. Then we compute evaluation function J(F) for every feature in each cluster and select feature with maximum value of J. consider the scenario shown in Fig1(c). Here, features $\{F_1, F_4, F_7\}$ have the maximum value of J. Hence features $\{F_1, F_4, F_7\}$ will be selected from F.

3.5 Classifiers

For evaluating the classification performance of the reduced feature subset, we used the different classifiers. In this subsection we provide a brief introduction to three classifiers which can be used for feature selection. Three of the most widely used and successful methods of classification are C4.5 decision trees (DT), K-nearest neighbor and Naïve Bayes (NB) learning.

Decision tree learning algorithm C4.5 builds decision tree by a set of training data using entropy based criterion. Thus, each feature of data can be used to make a decision that splits the data into two smaller subsets. Decision trees are also believed to be quite fast at any rate, several orders of the magnitude faster than the neural networks and SVMs.

K-nearest neighbor is an instance-based classification method that has been an effective approach for classification problems. It classifies samples based on closest training samples in the feature space. A sample is classified by a majority vote of its k-neighbors [25].

The Naïve Bayes classifier is one of the oldest classifiers. It is the simplest form of Bayesian network, in which all attributes are independent given the value of the class variable. The major advantage of the Naïve Bayes classifier is its short computational time for training. Naïve Bayes and the KNN can be easily used as incremental learners whereas rule algorithms cannot.

Also, contrary to KNN, NB and decision trees are considered resistant to noise because their pruning strategies avoid over-fitting the data in general and noisy data in particular.

Classification accuracy is the ratio of true positive and false positive to the total sample. In our experiment it is calculated by:

$$CA=\frac{|Correct-result|}{|Instance|} \tag{5}$$

where $|Correct-result|$ represents the number of correct classification result and

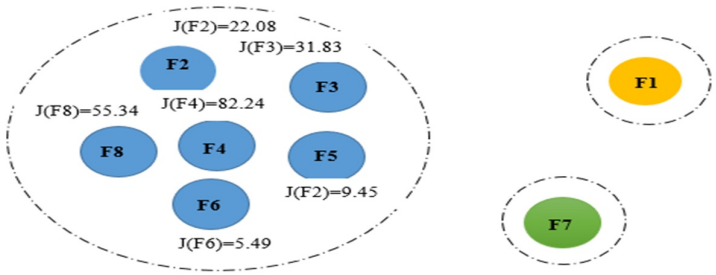
$|Instance|$ is the number of instances in a dataset.

4. Experiments and Results

Experiments were carried out on a workstation with 6GB main memory, 2GHZ Intel® core(TM)i7 cpu and 64-bit windows 10 operating system. All experiments of this section were done over 10 datasets taken from UCI dataset repository that are noise free. [29]. Descriptions of the datasets are given in table1. To evaluate the performance of our algorithm we use three well know classifiers, namely, Decision Trees, K-nearest neighbor (KNN), Naïve Bayes (NB) after feature selection.

	F1	F2	F3	F4	F5	F6	F7
F2	1.47						
F3	1.32	1.19					
F4	1.26	0.89	1.05				
F5	1.22	1.11	0.94	1.04			
F6	1.52	1.36	1.27	1.24	1.09		
F7	1.61	1.41	1.26	1.30	1.12	1.62	
F8	1.45	1.20	1.19	0.89	1.15	1.44	1.62

(a) Feature-Feature MI (features similarities in F-DBSCAN)



(b) Features clustering

(c) Candidate features

Figure 1. An example of FFS algorithm steps

Table 1.Description of the datasets

Data ID	Dataset	#Instances	#Attributes	#Classes
1	spamebase	4601	57	2
2	Coil2000	5822	85	2

3	*wine*	*178*	*13*	*3*
4	*hepatitis*	*366*	*34*	*6*
5	*satimage*	*155*	*19*	*2*
6	*lymphography*	*148*	*18*	*4*
7	*Xd6*	*461*	*9*	*2*
8	*Waveform-40*	*5000*	*40*	*3*
9	*Image segmentation*	*2310*	*19*	*7*
10	*Statlog(suttle)*	*58000*	*9*	*7*

The proposed algorithm is compared with four standard feature selection algorithms, namely mRMR, FCBF, ReliefF and MIM. mRMR use the maximum relevance and minimum redundancy criterion to select a feature from the original feature set. FCBF is a fast filter method which can identify relevant features as well as redundancy among relevant features without pairwise correlation analysis. ReliefF extends Relief, similar to Relief, ReliefF randomly samples a number of instances from training set and updates the relevance estimation of each feature based on the difference between the selected instance and the two nearest instances of the same and opposite classes. MIM by picking features which maximize their mutual information with the class to predict conditional to any feature already picked, it ensures the selection of features which are both individually informative and two-by-two weakly dependent.

4.1 Classification Accuracy

In what follows, the classification performance of the proposed algorithm, along with a comparison with other four algorithms, is demonstrated on the data sets using the classification accuracy of the three classifiers: C4.5, K-NN, and Naïve Bayes.

Tables 2, 3, and 4 show the classification accuracies of the three different types of classifiers on the 10 datasets after each feature selection algorithm is performed, respectively.

Table 2.Classification accuracy of C4.5 with five feature selection algorithms

Dataset	***FFS***	***mRMR***	***FCBF***	***ReiliefF***	***MIM***
spamebase	85.54%	**90.43%**	81.63%	84.89%	84.89%
Coil2000	**94.07%**	76.95%	84.45%	68.15%	68.15%
wine	**88.57%**	82.85%	74.28%	85.71%	71.42%
hepatitis	**67.74%**	48.38%	54.83%	45.16%	35.48%
satimage	**83.42%**	58.17%	80.04%	83.29%	54.45%
lymphography	**89.47%**	68.42%	52.63%	73.68%	78.95%
Xd6	**67.39%**	**67.39%**	66.30%	66.30%	66.30%
Waveform-40	**65%**	63.33%	61.67%	58.33%	**65%**
Image segmentation	88.63%	**91.88%**	87.26%	88.96%	**91.88%**
Statlog(suttel)	**99.72%**	**99.72%**	97.21%	94.55%	96.87%
average	**82.95%**	74.75%	74.03%	74.90%	71.34%

Table 2 shows the classification accuracy of C4.5. From it we observe that the proposed feature selection gives good classification accuracy on most datasets. In addition, from the average classification accuracies presented in the last row of table, it can be seen that the average accuracy value for all datasets is equal to 74.75% for mRMR, 74.03% for FCBF, 74.90% for ReliefF, and 71.34% for MIM, compared with 82.95% for FFS. FFS ranks 1 with a margin of 8.05% to the second best accuracy. From this, we can observe that the proposed algorithm is clearly superior to others on most of

the datasets. With the experimental results presented in Tables 3 and 4 by the K-nearest neighbor and Naïve Bayes classifiers, respectively, we can see that the proposed algorithm FFS shows similar patterns to that of C4.5 classifier. But for some dataset FFS is not the best; for example, in spamebase dataset we can observe that FFS ranks 2 to the best accuracy of Decision Tree and KNN of mRMR. However, the Eps value used for this dataset may not be the optimal value.

Table 3. Classification accuracy of K-NN with five feature selection algorithms

Dataset	*FFS*	*mRMR*	*FCBF*	*ReiliefF*	*MIM*
Spamebase	74.13%	**80.54%**	73.47%	71.63%	72.06%
Coil2000	**94.07%**	39.13%	71.63%	39.45%	42.93%
wine	**82.85%**	77.14%	71.14%	74.28%	68.57%
hepatitis	**67.74%**	54.83%	54.83%	48.38%	45.16%
satimage	**81.84%**	54.90%	80.60%	83.43%	56.46%
lymphography	**78.95%**	73.68%	47.37%	68.42%	63.16%
Xd6	**67.39%**	**67.39%**	66.30%	66.30%	56.52%
Waveform-40	66.67%	70%	71.67%	51.67%	**75%**
Image segmentation	91.23%	91.23%	88.96%	79.87%	**95.45%**
Statlog (suttel)	**99.72%**	98.04%	95.09%	93.11%	94.86%
average	**80.46%**	70.68%	72.10%	67.65%	67.01%

Table 4. Classification accuracy of Naïve Bayes with five feature selection algorithms

Dataset	*FFS*	*mRMR*	*FCBF*	*ReiliefF*	*MIM*
Spamebase	62.93%	**82.82%**	75.10%	81.95%	81.73%
Coil2000	**94.07%**	65.86%	76.52%	60.10%	64.67%
wine	**94.28%**	85.71%	85.71%	91.42%	85.71%
hepatitis	**67.74%**	51.61%	64.51%	38.70%	41.93%
satimage	76.09%	59.30%	76.32%	**77.90%**	58.17%
lymphography	**71.54%**	64.37%	51.26%	65.62%	61.29%
Xd6	**67.39%**	**67.39%**	66.30%	66.30%	66.30%
Waveform-40	70%	75%	76.67%	46.67%	**80%**
Image segmentation	**75.58%**	66.67%	64.28%	58.67%	67.85%
Statlog(suttel)	**82.71%**	80.25%	79.36%	81.54%	80.17%
average	**76.23%**	69.88%	71.60%	66.88%	68.78%

4.2 Proportion of Selected Features

The proportion of selected features is the ratio of the number of features selected by a feature selection algorithm to the original number of features of a dataset. Fig 4 shows the proportion of the proposed feature selection algorithm (FFS) for each dataset. It can observe that FFS is able to select less than 45% of features for feature selection in different datasets. The results indicate that the average proportion of selected features is 24.40% and the proposed algorithm obtained the best classification accuracy of C4.5, K-NN and NB by 82.95%, 80.46% and 76.23%, respectively. Fig 2 shows Average accuracy of 5 methods of decision tree. Note that, in average of all cases, the proposed method gets the best results for C4.5 classifier. Generally the algorithm achieves significant reduction of dimensionality by selecting only a small portion of the original features. Also, it is shown that with the increase in size of the original feature sets for different datasets, the cardinality for optimal feature subset identified by FFS also increases. Therefore, when the size of feature set varies from 9 to 85 (as shown in Table 1) the cardinality of the optimal feature subsets varies from 1 to 16. Fig 3 establishes this fact for 10 datasets.

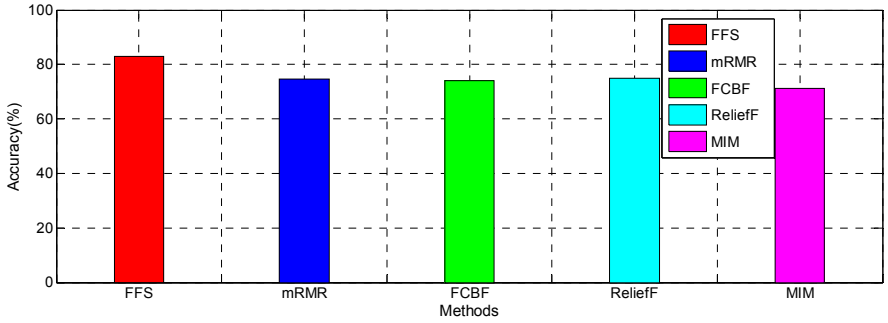


Figure 2. Average accuracy of 5 methods of decision tree

5. Conclusions

In this paper, we have presented a novel supervised feature selection method, which includes feature clustering and representative feature selection. The proposed algorithm can select candidate features that have high relevance to the class and low redundancy among the selected features. We have compared the performance of the proposed algorithm with some existing methods. The experiment results show that our method is effective for feature selection. Also, results on the various datasets show that the FFS method performs consistently well for the different sets of features chosen with an accuracy of 65-99.72%. F-DBSCAN algorithm is sensitive to its input parameters and it is not easy to determine the optimal value of them in practice. For the future work, we will extend the proposed method to choose F-DBSCAN parameters automatically by using optimization algorithms.

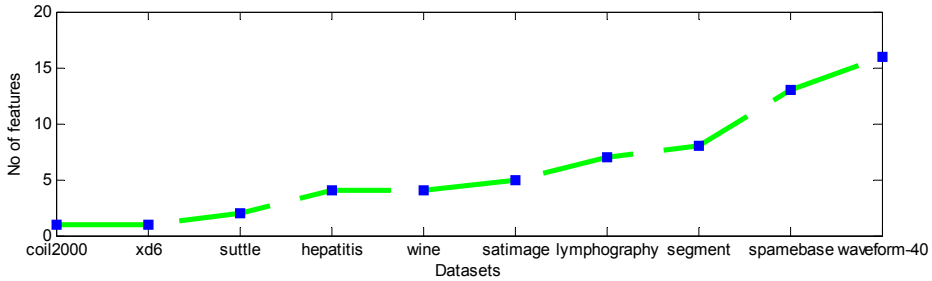


Figure 3. Optimal range of the size of feature subsets for different datasets

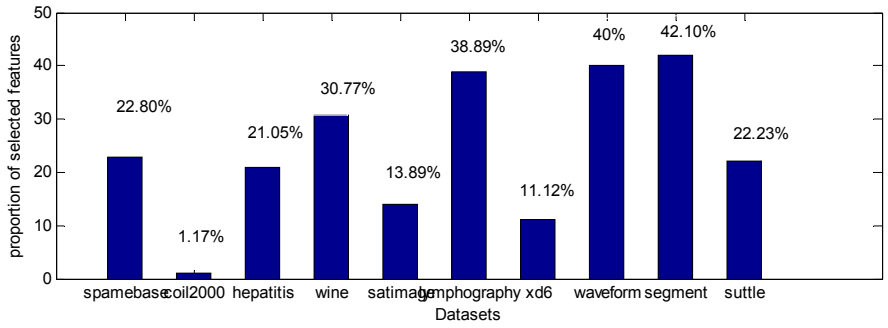


Figure 4. Proportion of selected features for different datasets

References

[1] Wang, De, Feiping Nie, and Heng Huang. "Feature Selection via Global Redundancy Minimization." Knowledge and Data Engineering, IEEE Transactions on 27.10 (2015): 2743-2755.

[2] Shang, Ronghua, et al. "Self-representation based dual-graph regularized feature selection clustering." Neurocomputing 171 (2016): 1242-1253.

[3] Guyon, Isabelle, and André Elisseeff. "An introduction to variable and feature selection." The Journal of Machine Learning Research 3 (2003): 1157-1182.

[4] Bakus, Jan, and Mohamed S. Kamel. "Higher order feature selection for text classification." Knowledge and Information Systems 9.4 (2006): 468-491.

[5] Fung, Glenn, and Jonathan Stoeckel. "SVM feature selection for classification of SPECT images of Alzheimer's disease using spatial information." Knowledge and Information Systems 11.2 (2007): 243-258.

[6] Saeys, Yvan, Iñaki Inza, and Pedro Larrañaga. "A review of feature selection techniques in bioinformatics." bioinformatics 23.19 (2007): 2507-2517.

[7] Hoque, N., D. K. Bhattacharyya, and Jugal K. Kalita. "MIFS-ND: a mutual information-based feature selection method." Expert Systems with Applications 41.14 (2014): 6371-6385.

[8] Song, Qinbao, Jingjie Ni, and Guangtao Wang. "A fast clustering-based feature subset selection algorithm for high-dimensional data." Knowledge and Data Engineering, IEEE Transactions on 25.1 (2013): 1-14.

[9] Chandrashekar, Girish, and Ferat Sahin. "A survey on feature selection methods." Computers & Electrical Engineering 40.1 (2014): 16-28.

[10] Vergara, Jorge R., and Pablo A. Estévez. "A review of feature selection methods based on mutual information." Neural Computing and Applications 24.1 (2014): 175-186.

[11] Xue, Bing, Mengjie Zhang, and Will N. Browne. "Particle swarm optimization for feature selection in classification: A multi-objective approach." Cybernetics, IEEE Transactions on 43.6 (2013): 1656-1671.

[12] Doquire, Gauthier, and Michel Verleysen. "Mutual information-based feature selection for multilabel classification." Neurocomputing 122 (2013): 148-155.

[13] Novovičová, Jana, and Antonin Malik. "Information-theoretic feature selection algorithms for text classification." Neural Networks, 2005. IJCNN'05. Prc ∃E International Joint Conference on. Vol. 5. IEEE, 2005.

[14] Deepthi, P. S., and Sabu M. Thampi. "PSO based feature selection for clustering gene expression data." Signal Processing, Informatics, Communication and Energy Systems (SPICES), 2015 IEEE International Conference on. IEEE, 2015.

[15] Aghdam, Mehdi Hosseinzadeh, Nasser Ghasem-Aghaee, and Mohammad Ehsan Basiri. "Text feature selection using ant colony optimization." Expert systems with applications 36.3 (2009): 6843-6853.

[16] Saroj, Jyoti. "Multi-objective genetic algorithm approach to feature subset optimization." Proc. of IEEE Int'l Advance Computing Conf.(IACC). 2014.

[17] Bu, Hualong, Shangzhi Zheng, and Jing Xia. "Genetic algorithm based Semi-feature selection method." Bioinformatics, Systems Biology and Intelligent Computing, 2009. IJCBS'09. International Joint Conference on. IEEE, 2009.

[18] Şeref, Onur, et al. "Information-theoretic feature selection with discrete k-median clustering." Annals of Operations Research (2012): 1-26.

[19] Au, Wai-Ho, et al. "Attribute clustering for grouping, selection, and classification of gene expression data." IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB) 2.2 (2005): 83-101.

[20] Liu, Huawen, et al. "A new feature selection method based on clustering." Fuzzy Systems and Knowledge Discovery (FSKD), 2011 Eighth International Conference on. Vol. 2. IEEE, 2011.

[21] Jiang, Jung-Yi, Yao-Lung Su, and Shie-Jue Lee. "MIKM: A mutual information-based K-medoids approach for feature selection." Machine Learning and Cybernetics (ICMLC), 2011 International Conference on. Vol. 1. IEEE, 2011.

[22] Kisilevich, Slava, Florian Mansmann, and Daniel Keim. "P-DBSCAN: a density based clustering algorithm for exploration and analysis of attractive areas using collections of geo-tagged photos." Proceedings of the 1st international conference and exhibition on computing for geospatial research & application. ACM, 2010.

[23] Karami, Amin, and Ronnie Johansson. "Choosing dbscan parameters automatically using differential evolution." International Journal of Computer Applications 91.7 (2014).

[24] Qian, Wenbin, and Wenhao Shu. "Mutual information criterion for feature selection from incomplete data." Neurocomputing 168 (2015): 210-220.

[25] Sotoca, José Martínez, and Filiberto Pla. "Supervised feature selection by clustering using conditional mutual information-based distances." Pattern Recognition 43.6 (2010): 2068-2081.

[26] Peng, Hanchuan, Fuhui Long, and Chris Ding. "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy." Pattern Analysis and Machine Intelligence, IEEE Transactions on 27.8 (2005): 1226-1238.

[27] Yu, Lei, and Huan Liu. "Feature selection for high-dimensional data: A fast correlation-based filter solution." *ICML*. Vol. 3. 2003.

[28] Maji, Pradipta. "Mutual information-based supervised attribute clustering for microarray sample classification." Knowledge and Data Engineering, IEEE Transactions on 24.1 (2012): 127-140.

[29] Lichman, Moshe. "UCI machine learning repository." University of California, Irvine, School of Information and Computer Sciences (2013).