Mach Learn (2015) 98:157–180
DOI 10.1007/s10994-013-5337-8

Unsupervised feature selection with ensemble learning

Haytham Elghazel · Alex Aussem

Received: 25 May 2012 / Accepted: 26 February 2013 / Published online: 3 April 2013

Abstract In this paper, we show that the way internal estimates are used to measure variable importance in Random Forests are also applicable to feature selection in unsupervised learning. We propose a new method called Random Cluster Ensemble (RCE for short), that estimates the out-of-bag feature importance from an ensemble of partitions. Each partition is constructed using a different bootstrap sample and a random subset of the features. We provide empirical results on nineteen benchmark data sets indicating that RCE, boosted with a recursive feature elimination scheme (RFE) (Guyon and Elisseeff, Journal of Machine Learning Research, 3:1157–1182, 2003), can lead to significant improvement in terms of clustering accuracy, over several state-of-the-art supervised and unsupervised algorithms, with a very limited subset of features. The method shows promise to deal with very large domains. All results, datasets and algorithms are available on line (http://perso.univ-lyon1.fr/haytham.elghazel/RCE.zip).

Keywords Unsupervised learning · Feature selection · Ensemble methods · Random forest

1 Introduction

Feature selection is an essential component of quantitative modeling, data-driven construction of decision support models or even computer-assisted discovery. The identification of relevant subsets of random variables among thousands of potentially irrelevant and redundant variables is a challenging topic of pattern recognition research that has attracted much attention over the last few years (Hua et al. 2009; Ghaemi et al. 2009; Morais and Aussem 2010; Saeys et al. 2007; Tuv et al. 2009). In supervised learning, feature

Editors: Emmanuel Müller, Ira Assent, Stephan Günnemann, Thomas Seidl, Jennifer Dy.

H. Elghazel (✉) · A. Aussem
University of Lyon, 69622 Lyon, France
e-mail: haytham.elghazel@univ-lyon1.fr

A. Aussem
e-mail: alexandre.aussem@univ-lyon1.fr

H. Elghazel · A. Aussem
LIRIS, UMR 5205, University of Lyon 1, Villeurbanne, France

selection algorithms maximize some function of predictive accuracy. The relevant subsets of variables at those that conjunctively prove useful to construct an efficient classifier from data. It enables the classification model to achieve good or even better solutions than with the whole set of features. But in unsupervised learning, we are not given class labels. It becomes unclear which features we should keep as there are no obvious criteria to guide the search. Intuitively, all features are not equally important. Some of the features may be redundant, some may be irrelevant, and some can even misguide clustering results. Broadly speaking, the feature selection in unsupervised learning aims at finding relevant subsets of variables that produce "natural groupings" by grouping "similar" objects together based on some similarity measure (Dy and Brodley 2004). Reducing the number of features increases comprehensibility and ameliorates the problem that some unsupervised learning algorithms break down with high dimensional data.

Databases have increased many fold in recent years. Important recent problems (i.e., DNA data in biology) often have the property that there are hundreds or thousands of variables, with each one containing only a small amount of information. A single clustering model is known to produce very bad groupings as the learning algorithms break down with high dimensional data. Clustering ensemble is an effective solution to overcome the dimensionality problem and to improve the robustness of the clustering (Fred and Jain 2002, 2005; Strehl and Ghosh 2002; Topchy et al. 2005; Ghaemi et al. 2009). The idea is to combine the results of multiple clusterings into a single data partition without accessing the original features. The strategy follows a split-and-merge approach: (1) construct a diverse and accurate ensemble committee of clusterings, and (2) combine the clustering results of the committee using a consensus function. Although considerable attention has been given on the problem of constructing accurate and diverse ensemble committee of clusterings, little attention has been given to exploiting the multiple clusterings of the ensemble with a view to identify and remove the irrelevant features.

The framework pursued in this article attempts to bridge the gap between supervised and unsupervised feature selection approaches in ensemble learning. The way internal estimates are used to measure variable importance in the Random Forests (RF) paradigm (Breiman 2001) have been influential in our thinking. In this study, we show that these ideas are also applicable to unsupervised feature selection. We extend the RF paradigm to unlabeled data by introducing a clustering ensemble termed as RCE (for Random Cluster Ensemble). RCE combines both data resampling (*bagging*) and random selection of features (*random subspaces*) strategies for generating an ensemble of component clusterings. A combination of these two main strategies for producing clustering ensembles leads to exploration of distinct views of inter-pattern relationships. Many approaches can be used to combine the multiple obtained partitions (Ghaemi et al. 2009). For sake of simplicity, we use the *evidence accumulation technique* proposed in Fred and Jain (2005) in our experiments. The method consists of taking the co-occurrences of pairs of patterns in the same cluster as votes for their association. The co-association matrix of patterns represents a new similarity measure between patterns. The final (consensus) clustering is obtained by running a traditional average-link hierarchical agglomerative algorithm on this matrix. Once the consensus clustering is obtained, we select the relevant features locally in each final cluster based on the RF-based out-of-bag importance measure discussed by Breiman (2001). Finally, RCE is boosted with a more standard recursive feature elimination scheme (Guyon and Elisseeff 2003) which recursively removes the features with the lowest importance measure. It is important to mention that RCE is not aiming at discovering multiple solutions. It should be viewed as another feature weighting method for subspace clustering, as the Entropy weighted k-means for subspace clustering algorithm (EWKM) (Hong et al. 2008a) and the feature group weighting k-means for subspace clustering algorithm (FGKM) (Jing et al. 2007). Therefore, we believe

our contribution is closely related to the topic "summarizing multiple clustering solutions" as our method outputs a consensus clustering by aggregating many partitions constructed on random subspaces.

Empirical results on nineteen UCI labeled data sets will be presented to address the following questions: (1) Is our feature selection for unsupervised learning algorithm better than clustering on all features? (2) Is it competitive with other state-of-the-art supervised and unsupervised feature selection methods? (3) How does the performance vary as more irrelevant variables are included in the feature set?

The rest of the paper is organized as follow: Section 2 reviews recent studies on unsupervised feature selection and consensus clustering methods. Section 3 introduces the RCE framework and describes how variable importance used in RF can be extended in unsupervised context by using ensemble clustering diversity with RCE. Experiments using relevant benchmarks data sets are presented in Sect. 4. Finally, we would like to mention that this paper is an extension of our earlier work that appeared in Elghazel and Aussem (2010). In this paper, RCE is boosted with the Recursive Feature Elimination (RFE) scheme to improve the final feature ranking and additional experiments are conducted to support the relevance of our approach.

2 Background

This section provides an overview of the methods proposed so far for feature selection in the unsupervised learning setting and discusses the major consensus clustering methods that appeared in the literature.

2.1 Unsupervised feature selection

The problem of unsupervised feature selection has attracted a great deal of interest recently. Like in supervised FS, the methods can be divided into three categories, depending on how they interact with the clustering algorithm: *wrapper*, *embedded* and *filter* approaches.

Wrapper methods perform a search in the space of feature subsets, guided by the outcome of the clustering model, by wrapping the unsupervised feature selection process around a clustering algorithm. Typically, a criterion is firstly defined for evaluating the quality of a candidate feature subset. Wrapper approaches aim to identify a feature subset such that the clustering algorithm trained on this feature subset achieve the optimal value of the predefined criterion, such as the normalized scatter separability (for k-means) (Dy and Brodley 2004) or the normalized likelihood (for EM clustering) (Dy and Brodley 2004) or the DB-index (Morita et al. 2003). Another example is given by the algorithm CEFS described in Hong et al. (2008a). It searches for a subset of all features such that the clustering algorithm trained on this feature subset can achieve the most similar clustering solution to the one obtained by an ensemble learning algorithm. RF has been also extended to unlabeled data leading to unsupervised learning (Breiman and Cutler 2003; Shi and Horvath 2006). The approach in RF is to consider the original data as class 1 and to create a synthetic second class of the same size that will be labeled as class 2. The synthetic second class is created by sampling at random from the univariate distributions of the original data. Thus, class 2 has the distribution of independent random variables, each one having the same univariate distribution as the corresponding variable in the original data. Class 2 thus destroys the dependency structure in the original data. This artificial two-class problem is run through RF. If the oob misclassification rate in the two-class problem is significantly lower than 50 % then the dependencies are playing an important role, otherwise the dependencies do not have a large role and not much discrimination is taking place. Formulating it as a two class problem has a number of payoffs (e.g., missing value imputation outlier detection etc.).

But the most important payoff—in our context—is the possibility of estimating the variable importance.

A major shortcoming of the aforementioned methods is that they output a single feature subset. To remedy this situation, Yuanhong et al. (2008) proposed a localized feature selection algorithm for clustering. The proposed method first computes adjusted and normalized scatter separability values for each individual cluster and then proceeds to a sequential backward search to search for the optimal feature subsets for each cluster. However, the method has high computational overhead that prohibits its use with high-dimensional data.

A number of other studies are based on the *embedded formalism* for weighting the variables (Frigui and Nasraoui 2004; Huang et al. 2005; Grozavu et al. 2009). In these approaches, the search for an optimal subset of features is built into the clustering construction making these techniques specific to a given learning algorithm, not to mention the problem known as the "curse of dimensionality" that arises when dealing with high-dimensional data spaces. A number of subspace clustering methods are dedicated to high dimension data sets, as for instance the Entropy weighted k-means for subspace clustering algorithm (EWKM) (Hong et al. 2008a) and the feature group weighting k-means for subspace clustering algorithm (FGKM) (Jing et al. 2007). Co-clustering methods may also be viewed as unsupervised feature selection appraoches, see for instance Wang et al. (2011), Gullo et al. (2012), Kluger et al. (2003).

Finally, *filter methods* discover the relevant and redundant features through analyzing the correlation and dependence among features without involving any clustering algorithms (Mitra et al. 2002; Dash et al. 2002). The most common filter strategies are based on feature ranking. In this context, two opposite strategies have been proposed in the literature: those that aim at the removal of redundant features and those that focus on the removal of irrelevant features. Recently, Hong et al. (2008b) proposed a consensus unsupervised feature ranking approach that combines multiple rankings of the full set of features into a single consensus one. The ranking of features is obtained using their relevance measured by the linear correlation coefficient and symmetrical uncertainty. Unfortunately, the authors only report experimental results on very low-dimensional data sets.

2.2 Consensus clustering

Consensus clustering has emerged as a powerful method to improve both the robustness and the stability of unsupervised classification solutions. As in the supervised setting, consensus clustering comprises two phases: the production of multiple clusterings and their combination. The way the individual clusterings are combined is often referred to as the consensus function. Methods for constructing ensembles include: manipulation of the samples such as resampling (*bagging*) (Dudoit and Fridlyand 2003) or *random subspaces* techniques (Strehl and Ghosh 2002; Topchy et al. 2005); injection of some randomness into the learning algorithm (Fred and Jain 2002, 2005); applying different clustering algorithms (Strehl and Ghosh 2002) or their relaxed versions (Topchy et al. 2005). Resampling methods build a group of clustering models based on bootstrapped replicates of the dataset; the consensus partition is obtained using a consensus function over the set of partitions of each single clustering model. Strehl and Ghosh (2002) proposed an Object Distributed Clustering (ODC) scenario for which individual clusterers have a limited selection of the object population but have access to all of the features. They define the optimal combined clustering as the one maximizing the average mutual information with all other individual clustering and introduced three heuristic (CSPA, HGPA and MCLA) to solve this problem. Dudoit and Fridlyand (2003) used bagging to improve accuracy of clustering in reducing the variability of PAM

(Partitioning Around Medoids) results. Two bagged clustering procedures were proposed. In the first approach, the clustering procedure is repeatedly applied to each bootstrap sample and the final partition is obtained by plurality voting. The second bagging procedure forms a new similarity matrix by recording, for each pair of observations, the proportion of times they are clustered together in the bootstrap clusters. This new similarity matrix is then used as the input to a clustering procedure.

The random subspace method is another source of clustering diversity that provides different views of the data, thereby improving the quality of unsupervised classification solutions. Topchy et al. (2005) showed that the combination of clusterings in projected random 1-dimensional subspaces, using an average-link consensus function, outperforms the combination of clusterings in the original multidimensional space. Similarly, Strehl and Ghosh (2002) proposed a method called Feature-Distributed Clustering to combine a set of clusterings obtained from partial views of the data. The same clustering algorithm is used to form the committee. The cluster labels are afterwards retrieved and combined using one of three heuristic mentioned above to form the consensus clustering.

3 RCE: Random Cluster Ensemble

In this section, we discuss our unsupervised Random Cluster Ensemble (RCE) algorithm aimed at estimating the feature's importance.

3.1 Constructing and combining multiple clustering solutions

Formulating the unsupervised learning problem as an ensemble learning problem has a number of payoffs: missing values can be replaced effectively, outliers can be found, and more importantly, the way variable importance is estimated in RF can be transposed rather easily in the unsupervised framework, provided some care is taken over the way in which the importance values are estimated, as we will see. We propose to combine both bagging and random subspaces for producing an ensemble of component clusterings (Elghazel and Aussem 2010). Our approach operates as follows: a new data set is drawn with replacement from the original data set; then m features are randomly selected from the entire feature set (leading to a partial view of the bootstrap data set) and a clustering solution is obtained by running a "base" clustering algorithm on the selected features. The same steps are repeated T times. This yields T partitions, where T is the size of ensemble committee. There are many reasons for using bagging in tandem with random feature subspaces. First, bagging can be used to give estimates of both the *variable importance* and the *example proximities* that will serve to build the final consensus clustering from the ensemble of clusterings. Second, this technique for combining many weak learners in an attempt to produce a strong learner has proven to be very effective in supervised ensemble learning. Third, clustering in the projected subspace not only reduces the computational burden, but also allows us to mitigate the curse of dimensionality. We give a brief outline of the key steps involved: given a specific data set $\mathcal{D} = \{x_1, \ldots, x_n\}$ with M input variables, form T bootstrap data sets $\mathcal{D}_k$ ($k \in \{1, \ldots, T\}$) in a random feature subspace (a mD view of the bootstrap data set where $m = \sqrt{M}$), and construct several clusterings $\mathbf{C}^k = h(x, \mathcal{D}_k)$ using a method h. The way these $\mathbf{C}^k$ ($k \in \{1, \ldots, T\}$) should be combined together is called the cluster ensemble problem. Several approaches have been introduced in the literature to solve the cluster ensemble problem. In this study, we adopt the average-link consensus function proposed in Fred and Jain (2005), based on co-association values between data. The proximity between pair of cases simply counts the fraction of clusters shared by these objects in the initial partitions. It is worth noting that numerous similarity-based clustering algorithms could also be applied

to the proximity (or similarity) matrix to obtain the final partition. Interestingly, the overall framework is sufficiently generic to allow any feasible consensus clustering procedure.

3.2 Out-of-bag estimates to measure variable importance

As described in Dy and Brodley (2004), the goal of unsupervised feature selection is to find the smallest feature subset that best uncovers "interesting" clusters from data according to a predefined criterion. However, it is often difficult to define a good criterion for evaluating the "interestingness" of obtained clusters and to understand the interaction of variables that is providing the groupings. As mentioned before, many criteria have been proposed in this context. However, all these criteria are only effective for some data sets and may be ineffective for other data sets. In this study, we introduce the RCE algorithm to address the problem of feature relevance evaluation. In RCE, bagging is used in tandem with feature subspace to give ongoing estimates of the feature importance of the combined ensemble of clusterings. These estimates are done out-of-bag, exactly as done in RF. For any given clustering in RCE there is a subset of the learning set left aside during learning, because each clustering was grown only on a bootstrap sample. These subsets, called out-of-bag (oob for short), can be used to give unbiased measures of feature importance. RCE estimates the relevance of features entering the clustering in the following way. After each clustering is constructed, one at a time, each feature $f = 1, 2, \ldots, M$ is shuffled (randomly permuted) in the oob examples and the oob data are re-assigned into clusters. At the end of the run, the oob cluster assignments for x with the fth variable noised up is compared with the original cluster assignment of x.[1] Intuitively, irrelevant features will not change the classification of x when altered in this way. The relative difference in classification between the original and shuffled data sets is therefore related to the relevance of the shuffled feature. More formally, the average number of times the variable-f-permuted oob example x is misclassified divided by the number of clusterings in the ensemble T is the importance score for variable f for this example. Finally, the importance of the fth variable for a given cluster (local) in the final consensus clustering is calculated as the sum of all the importance values over all the patterns that fall into this particular cluster.

Our feature importance measure has several advantages when compared to other existing unsupervised feature selection algorithms: First, as mentioned in Hong et al. (2008a), most existing unsupervised feature selection algorithms are dimensionality-biased. For example, if the scatter separability based feature selection algorithm is adopted, high-dimensional feature subsets are selected more easily (Dy and Brodley 2004; Hong et al. 2008a). The problem should be circumvented as our feature ranking approach operates on low-dimensional feature spaces. Second, as our approach leverages different clustering solutions to measure feature importance, it is expected to improve the robustness and stability of the feature subset compared to feature selection algorithms based on a single clustering method. Third, the estimates of variable importance are obtained from the oob examples only. As note by Breiman, it should therefore be as accurate as using a test set of the same size as the data set. Therefore, using the oob error estimate removes the need for a set aside test set. In each bootstrap data set, about one-third of the instances are left out. Therefore, the oob estimates are based on combining only about one-third as many clustering models as in the ongoing main combination. Finally, the local feature selection allows us to characterize each individual cluster. The clusters can therefore be more easily understood and interpreted by human experts.

[1] An oob instance x is assigned to the closest cluster, where the distance from an instance to one cluster C is given by the distance between the instance and the centroid of C.

input : A data set $\mathcal{D} = \{x_1, \ldots, x_n\}$ with M input features $\mathcal{F} = \{f_1, \ldots, f_M\}$, h: a base clustering procedure, T: number of committee members
output: A list of estimated local feature's importance $imp(f)$

Initialize an $n \times M$ matrix $\mathbf{I}$ to zero;
Initialize an $n \times n$ co-association matrix $\mathbf{A}$ to zero;
for $k \leftarrow 1$ **to** T **do**
  $\mathcal{D}_k = \{x_k^1, \ldots, x_k^n\} \leftarrow$ form the kth bootstrap sample from $\mathcal{D}$;
  $\mathcal{D}_{oob} = \{x_{oob}^1, \ldots, x_{oob}^{n_{oob}}\} \leftarrow \mathcal{D} \setminus \mathcal{D}_k$;
  $\mathcal{F}_k = \{f_k^1, \ldots, f_k^m\} \leftarrow$ randomly select $m = \sqrt{M}$ features from $\mathcal{F}$;
  Project $\mathcal{D}_k$ and $\mathcal{D}_{oob}$ onto feature subset $\mathcal{F}_k$ where $\mathcal{D}_k = \mathcal{D}_{k_{|\mathcal{F}_k}}$ and $\mathcal{D}_{oob} = \mathcal{D}_{oob_{|\mathcal{F}_k}}$;
  $\mathbf{C}^k \leftarrow$ partition the data set $\mathcal{D}_k$ using h;
  for *each pair of observations* $(x_k^i, x_k^j) \in \mathcal{D}_k \times \mathcal{D}_k$ **do**
    if $\mathbf{C}^k(x_k^i) = \mathbf{C}^k(x_k^j)$ **then**
      $\mathbf{A}(x_k^i, x_k^j) \leftarrow \mathbf{A}(x_k^i, x_k^j) + \frac{1}{T}$
    end
  end
  Classify each $x_{oob}^i \in \mathcal{D}_{oob}$ into $\mathbf{C}^k$ and obtain its label $\mathbf{C}^k(x_{oob}^i)$;
  for *each feature* $f \in \mathcal{F}_k$ **do**
    randomly permute the values of f in oob data $\mathcal{D}_{oob}$;
    re-assign each $x_{oob}^i \in \mathcal{D}_{oob}$ to a new label $\mathbf{C}_{new}^k(x_{oob}^i)$;
    if $\mathbf{C}_{new}^k(x_{oob}^i) \neq \mathbf{C}^k(x_{oob}^i)$ **then**
      $\mathbf{I}(x_{oob}^i, f) \leftarrow \mathbf{I}(x_{oob}^i, f) + \frac{1}{T}$
    end
  end
end
$\mathbf{C} \leftarrow$ cluster the n original observations in $\mathcal{D}$ on the basis of the co-association matrix $\mathbf{A}$ using the average-link hierarchical agglomerative algorithm;
for *each feature* $f \in \mathcal{F}$ *and each cluster* c_k *in* $\mathbf{C}$ **do**
  $imp(f, c_k) \leftarrow \sum_{i=1; x_i \in c_k}^{n} \mathbf{I}(x_i, v)$;
end

Algorithm 1: The RCE algorithm

Let $\mathcal{D} = \{x_1, \ldots, x_n\}$ be the data set and let M be the total number of input features $\mathcal{F} = \{f_1, \ldots, f_M\}$. The overall proposed RCE framework for unsupervised feature selection is summarized in Algorithm 1. It is worth mentioning that various methods can be employed to select the most important local variables in view of their importance estimates, including: (1) statistical tests (e.g., the Scree test Cattell 1966), (2) selecting a predefined percentage or number of variables (Saeys et al. 2008), (3) selecting the features whose importance exceeds a user-defined cutoff-threshold.

3.3 Complexity of RCE

In order to analyze the computational complexity of RCE, we identify three phases: (1) Constructing multiple clustering solutions on random subspaces, (2) Out-of-bag estimates to measure variable importance, and (3) ranking and selection of the best features. The first

step depends on the "base" clustering algorithm that is adopted. Most of the time is spent on computing vector distances. One such operation costs $O(\sqrt{M})$. Here, the number of bootstrap data set is T, the number of clusters is K, and the number of patterns is n. At each iteration, the overall complexity is $O(nKT\sqrt{M})$ for both k-means and SOM. It is worth noting that the random cluster ensemble construction process can easily be parallelized. In second phase, $\sqrt{M}$ features are shuffled (randomly permuted) in the T oob samples and the $O(n)$ oob data are re-assigned into clusters after computing the component-wise differences with the centroids. The complexity of the second phase is $O(nKT\sqrt{M})$. The *evidence accumulation technique* proposed in Fred and Jain (2005) is then used to combine the multiple obtained partitions. The method consists of taking the co-occurrences of pairs of patterns in the same cluster as votes for their association. The co-association matrix takes $O(n^2T)$ to compute. The final (consensus) clustering is obtained by running a traditional average-link hierarchical agglomerative algorithm on this matrix. Average-link clustering merges in each iteration the pair of clusters with the highest cohesion. We first compute all $O(n^2)$ similarities for the singleton clusters, and sort them for each cluster. Each iteration thus takes $O(n\log(n))$. Overall, the time complexity of average-link clustering is $O(n^2\log(n))$. Once the dendogram and the final consensus clustering is obtained, the Scree test (Cattell 1966) selects the top-ranked features locally in each final cluster based on the RF-based oob importance measure in $O(KM\log(M))$. Overall, the computational complexity is $O(n^2\log(n) + KM\log(M))$ which makes RCE scalable to high-dimensional data.

4 Experiments

In order to empirically test the proposed non-supervised feature selection method as implemented by our generic RCE algorithm, we ran a number of experiments on several UCI data sets (Blake and Merz 1998) and compared RCE against several state-of-art methods. The evaluation of the performance of RCE was conducted as follows: (A) quality of the selected features using *k-means* as the "base" clustering algorithm, (B) quality of the selected features using *Self-Organizing Map* (SOM) (Kohonen 2001) as the "base" clustering algorithm, (C) quality of selected features with RCE "boosted" by Recursive Feature Elimination (RCE-RFE for short) using *k-means* as the "base" clustering, and (D) robustness of RCE-RFE in the presence of many noisy features.

In experiments A and B, we used the Scree test to select the "optimal" number of features in view of their importance (see Cattell 1966 for details). The variables are ordered by importance, and the importance is plotted against the variable number. The important variables are the ones above the "elbow" in the plot. It's called a scree test because the graph usually looks a bit like where a cliff meets the plain. The Scree tells us where the cliff stops and the plain begins. The Scree test was applied locally in each cluster of the consensus partition obtained by RCE and the union of these features was taken as the overall selected feature subset.

In experiments C and D, the performance of RCE was boosted using the Recursive Feature Elimination (RFE) scheme (Guyon and Elisseeff 2003). We chose the RFE scheme for its simplicity and efficiency despite the computational expense incurred in running RCE several times. The RFE selection method (Guyon and Elisseeff 2003) is basically a recursive process that ranks features according to some measure of importance in decreasing order. At each iteration, the last feature importances (and hopefully the less relevant) is removed. Another possibility is to remove a percentage of the least important features each time in

Table 1 The data sets used in Sects. 4.1, 4.2, 4.3 and 4.4

Data sets	#instances	#features	#labels	Section
Breast Tissue	106	9	6	[4.1, 4.3]
Glass	214	9	7	[4.1, 4.1]
Ionosphere	351	34	2	[4.1, 4.3]
Isolet	1559	617	26	[4.1, 4.3]
Leukemia	73	7129	2	[4.1, 4.3]
Lung	32	56	3	[4.1, 4.3]
Madelon	2600	500	2	[4.1, 4.3]
Multiple Features	2000	216	10	[4.1, 4.3]
Ovarian	54	1536	2	[4.1, 4.3]
Parkinson	195	22	2	[4.1, 4.3]
Pima	768	8	2	[4.1, 4.3]
Promoters	106	57	2	[4.1, 4.3]
Robot	88	90	4	[4.1, 4.3]
Segmentation	2310	19	7	[4.1, 4.3]
Soybean	266	35	15	[4.1, 4.3]
Spect	267	22	2	[4.1, 4.3]
Wdbc	569	30	2	[4.1, 4.3]
Wine	178	13	3	[4.1, 4.3]
Wisconsin	699	9	2	[4.1, 4.3]
Wave	5000	40	3	[4.2]
Wdbc	569	30	2	[4.2]
Spamb	4601	57	2	[4.2]
Madelon	2600	500	2	[4.2]
Isolet	1559	617	26	[4.2]
Iris	150	4	3	[4.4]

order to speed up the process. The recursion is necessary because the feature ranking may change substantially during the stepwise elimination process (in particular for large data sets and also highly correlated features). We stopped the RCE scheme once the same number of features as the competing methods was obtained in order to make fair comparisons.

Nineteen benchmark (UCI) labelled data sets (Blake and Merz 1998) were selected in experiments A and C to test the performance of RCE. They are described in Table 1. We selected these datasets as they are either well understood in terms of feature relevance (e.g., Wave and Iris), or they contain thousands features with comparatively much smaller sample size (e.g., *Ovarian* Schummer et al. 1999 and *Leukemia* Golub et al. 1999) and are thus good candidates for feature selection. Most of these data sets have already been used by other authors for testing the performance of their unsupervised feature selection algorithms (Hong et al. 2008a, 2008b; Grozavu et al. 2009; Morita et al. 2003; Dy and Brodley 2004). Note that, in experiments B, we had no other choice than using the same data sets used in Grozavu et al. (2009) for comparison purposes as their code (methods *lwd-SOM* and *lwo-SOM*) is not freely available. Also, the performance measures reported in Sect. 4.2 are directly taken from Grozavu et al. (2009).

The standard principle to assess the quality of our unsupervised feature selection is to monitor the quality of the clustering by computing an external criterion that evaluates how well the clustering matches the known class labels. The quality of the clustering on a validation test set is indicative of whether the selected features are relevant. For sake of convenience, let us recall the definition of the two external criteria of clustering quality used in our experiments: the Normalized Mutual Information index (NMI) (Meila 2005) and the Purity index.

The normalized mutual information (NMI) between two partitions P^a and P^b is may be written, after simplification, as:

$$\mathrm{NMI}(P^a, P^b) = \frac{\sum_{i,j} n_{ij} \log\left(\frac{n_{ij}^{ab} n}{n_i^a n_j^b}\right)}{\sqrt{\left(\sum_i n_i^a \log\left(\frac{n_i^a}{n}\right)\right) \times \left(\sum_j n_j^b \log\left(\frac{n_i^b}{n}\right)\right)}}$$

with n_{ij}^{ab} denoting the number of shared patterns between clusters C_i^a and C_j^b in partitions P^a and P^b respectively.

To compute purity, each cluster is assigned to the class which is most frequent in the cluster, and then the accuracy of this assignment is measured by counting the number of correctly assigned samples and dividing by N. Formally:

$$\mathrm{purity}(\Omega, \mathbb{C}) = \frac{1}{N} \sum_k \max_j |\omega_k \cap c_j|$$

where $\Omega = \{\omega_1, \omega_2, \ldots, \omega_K\}$ is the set of clusters and $\mathbb{C} = \{c_1, c_2, \ldots, c_J\}$ is the set of classes. We interpret ω_k as the set of samples in ω_k and c_j as the set of samples in c_j. Alternatively, we may view the clustering as a series of decisions, one for each of the $\binom{N}{2}$ pairs of data samples, and we assign two samples to the same cluster if and only if they are similar. A true positive (TP) decision assigns two similar samples to the same cluster, a true negative (TN) decision assigns two dissimilar samples to different clusters.

In the sequel, either the NMI Index or the Purity will be taken as our accuracy measure, depending on context.

4.1 RCE using k-means as the base clustering algorithm

First, the k-means clustering algorithm was adopted as the "base" clustering algorithm. The latter is known to be unstable and often yields to a good ensemble committee of multiple clustering solutions with many diversities (Fred and Jain 2005; Hong et al. 2008a). The size of the clustering ensemble, r, was arbitrary set to 200 in our experiments. In these experiments, the number and the quality of the features selected by RCE was studied and compared with those obtained by several state-of-the-art unsupervised FS algorithms, namely the DB-index wrapper unsupervised FS algorithm (Morita et al. 2003) (DBI), the clustering ensemble guided FS algorithm (CEFS) (Hong et al. 2008a), the unsupervised RF feature selection (URF) algorithm (Breiman and Cutler 2003),[2] the Entropy weighted k-means for subspace clustering algorithm (EWKM) (Hong et al. 2008a), the feature group weighting k-means for subspace clustering algorithm (FGKM) (Jing et al. 2007)[3] and the spectral bicluster algo-

[2] The number of trees in the RF classifier was set to 500.

[3] For fair comparisons, the same experimental settings in Chen et al. (2012) was adopted here for EWKM and FGKM, i.e., $\lambda = 5$ for EWKM and $\lambda = 6$, $\eta = 30$ for FGKM. The R "weightedKmeans" package is used for the implementation of both methods.

rithm (SBI) (Kluger et al. 2003).[4] To make fair comparisons, the same experimental protocol in Hong et al. (2008a) was adopted here and the quality assessment was based on the *NMI rate*. Then, the number of clusters was set to the true number of labels.

We used the Scree test to select the "optimal" number of features with RCE, URF, EWKM and FGKM. CEFS and DBI use their own heuristic to select the features. The selected feature subset for SBI approach is given by the union of local features obtained in each bicluster. The quality of the selected feature subset obtained from each approach on the 19 data sets in Table 1 was evaluated by running k-means on this feature subset and the clustering accuracy was calculated by the NMI Index. The algorithm was also ran on the original feature set (all features). Results were averaged over 20 independent runs as k-means is very unstable. The average NMI and the standard deviations are shown in Tables 2 and 3.

In this study, we adopt the methodology proposed by Demsar (2006) for the comparison of several classifiers over multiple datasets. In this methodology, the non-parametric Friedman test is used to evaluate the rejection of the hypothesis that all the classifiers perform equally well for a given risk level. It ranks the algorithms for each data set separately, the best performing algorithm getting the rank of 1, the second best rank 2 etc. It then checks whether the measured average ranks are significantly different from the mean rank (here 4.5) expected under the null-hypothesis. From Tables 2 and 3, we observe that the results using RCE dominate those obtained using the EWKM, FGKM, DBI, SBI, URF, CEFS and using the original set of features (column "All Features"). RCE ranks the first with rank 2.34 on average as shown in the last row. CEFS, FGKM and 'all features' seem to perform equally well while DBI, SBI, EWKM and URF perform the worst. The Friedman test reveals statistically significant differences ($p < 0.0005$) in accuracy for the partitions obtained with the selected features. As we intend to compare our single control method, RCE, with seven other methods, we compute the critical difference with the Bonferroni-Dunn test (see Demsar 2006 for details). The performance of RCE against the seven other classifiers is significantly different if the corresponding average ranks differ by at least 1.95 at $p = 0.1$. So the post-hoc test reveals statistically significant improvements in accuracy for RCE over FGKM (and EWKM, URF, DBI, SBI) as $4.47 - 2.34 > 1.95$ at $p = 0.1$, while we did not detect any significant improvement between the group of algorithms RCE, CEFS and 'all features'. Nevertheless, the high computational complexity of CEFS is a major drawback when one wants to select the best features among a large number of features. Table 4 reports the runtime on four representative data sets. As observed, the RCE is significantly faster than RCE, up to 500 on Leukemia.

For sake of completeness, we also give the significance *t-test* results at the bottom row of the Tables 2 and 3. The analogous trend between RCE and other features selection methods can be observed and it is important to note in this regard that feature selection with RCE never significantly degraded accuracy in any of the datasets tested. A closer inspection of the accuracy values reported in Table 2 shows that employing 'all features' is simply inadequate for obtaining good accuracies in all cases (e.g., Ovarian and Promoters). Finally, the significant gain in accuracy (expressed as the NMI index) on these data sets confirms the ability of the RCE approach to improve the quality of the clustering and to generate meaningful clusters.

Also instructive is the fact that, in high dimensional feature space, RCE (using the Scree test) can make dramatic reductions in the feature space and consequently improve running time performance as well. It is reasonable to conjecture that the better the variable importances are estimated, the more effective will be the Scree test (i.e., the estimation of the

[4]The R "biclust" Package is used for the implementation of this method.

Table 2 Experimental results using k-means, averaged other 20 independent runs, on 19 test data sets (the numbers in the brackets represent the optimal number of features). Bottom rows of the table present average rank of NMI mean used in the computation of the Friedman test and the Win-Tie-Loss comparisons between RCE against other approaches after t-tests at 95 % significance level

Data sets	All features	FGKM
Breast Tissu	$51.18 \pm 1.38^{+}$	$55.69 \pm 4.22(6)^{=}$
Glass	$32.80 \pm 3.61^{+}$	$28.00 \pm 2.58(6)^{+}$
Ionosphere	$12.62 \pm 2.37^{=}$	$12.95 \pm 2.44(32)^{=}$
Isolet	$69.83 \pm 1.74^{-}$	$69.83 \pm 1.74(617)^{-}$
Leukemia	$6.85 \pm 5.00^{+}$	$6.85 \pm 5.00(7129)^{+}$
Lung	$22.51 \pm 5.58^{+}$	$16.28 \pm 6.82(25)^{+}$
Madelon	$0.36 \pm 0.86^{+}$	$1.11 \pm 1.38(495)^{+}$
Multiple	$67.64 \pm 3.93^{=}$	$67.64 \pm 3.93(216)^{=}$
Ovarian	$51.46 \pm 11.25^{+}$	$52.73 \pm 11.24(735)^{+}$
Parkinson	$23.35 \pm 0.19^{+}$	$18.13 \pm 4.32(21)^{+}$
Pima	$2.97 \pm 0.00^{-}$	$1.39 \pm 0.86(6)^{-}$
Promoters	$8.65 \pm 6.98^{+}$	$9.80 \pm 10.45(54)^{+}$
Robot	$31.20 \pm 8.34^{+}$	$24.23 \pm 2.32(44)^{+}$
Segmentation	$60.73 \pm 1.71^{=}$	$62.34 \pm 2.21(17)^{=}$
Soybean	$65.87 \pm 1.84^{+}$	$66.59 \pm 2.57(30)^{+}$
Spect	$10.72 \pm 1.58^{-}$	$2.99 \pm 0.00(3)^{+}$
Wdbc	$62.32 \pm 0.00^{-}$	$62.92 \pm 0.00(14)^{-}$
Wine	$81.99 \pm 8.28^{=}$	$78.51 \pm 0.57(5)^{+}$
Wisconsin	$73.61 \pm 0.00^{-}$	$72.24 \pm 0.00(6)^{+}$
Average rank	4.05	4.47
(Win/Tie/Loss)	(10/4/5)	(12/4/3)
Data sets	URF	DBI
Breast Tissu	$54.78 \pm 2.02(5)^{+}$	$27.46 \pm 1.70(1)^{+}$
Glass	$33.57 \pm 2.44(5)^{+}$	$18.25 \pm 1.47(2)^{+}$
Ionosphere	$12.38 \pm 3.14(30)^{=}$	$7.93 \pm 0.00(3)^{+}$
Isolet	$58.39 \pm 0.29(1)^{+}$	$71.38 \pm 1.13(278)^{-}$
Leukemia	$7.91 \pm 5.33(3431)^{+}$	$9.01 \pm 3.44(3510)^{+}$
Lung	$1.56 \pm 0.50(1)^{+}$	$21.25 \pm 9.33(7)^{+}$
Madelon	$0.34 \pm 0.81(19)^{+}$	$1.87 \pm 0.05(152)^{+}$
Multiple	$67.29 \pm 3.42(209)^{+}$	$67.97 \pm 2.16(97)^{=}$
Ovarian	$49.27 \pm 15.49(1374)^{+}$	$54.84 \pm 2.38(714)^{+}$
Parkinson	$17.31 \pm 0.11(13)^{+}$	$25.58 \pm 0.00(1)^{+}$
Pima	$1.89 \pm 1.02(3)^{-}$	$0.07 \pm 0.02(1)^{-}$
Promoters	$18.19 \pm 0.00(1)^{+}$	$17.66 \pm 9.37(2)^{+}$
Robot	$31.52 \pm 7.10(87)^{+}$	$34.41 \pm 9.24(28)^{+}$
Segmentation	$48.42 \pm 0.85(3)^{+}$	$53.42 \pm 3.26(5)^{+}$
Soybean	$68.56 \pm 2.57(11)^{=}$	$55.42 \pm 1.63(7)^{+}$
Spect	$4.80 \pm 1.52(18)^{+}$	$7.92 \pm 0.00(1)^{+}$
Wdbc	$61.47 \pm 0.00(12)^{+}$	$52.77 \pm 0.39(1)^{+}$
Wine	$79.04 \pm 1.59(4)^{=}$	$39.91 \pm 0.24(1)^{+}$
Wisconsin	$57.37 \pm 1.00(2)^{+}$	$50.56 \pm 0.00(1)^{+}$
Average rank	5.42	5.16
(Win/Tie/Loss)	(15/3/1)	(16/1/2)

$^{+}$ RCE is significantly better after t-tests at 95 % significance level

$^{-}$ RCE is significantly worse after t-tests at 95 % significance level

Table 3 Experimental results using k-means, averaged other 20 independent runs, on 19 test data sets (the numbers in the brackets represent the optimal number of features). Bottom rows of the table present average rank of NMI mean used in the computation of the Friedman test and the Win-Tie-Loss comparisons between RCE against other approaches after t-tests at 95 % significance level

Data sets	EWKM	CEFS
Breast Tissu	$51.18 \pm 1.38(9)^{+}$	$53.81 \pm 2.18(7)^{+}$
Glass	$32.80 \pm 3.61(9)^{+}$	$28.44 \pm 3.05(4)^{+}$
Ionosphere	$11.46 \pm 2.40(27)^{+}$	$14.00 \pm 0.00(22)^{-}$
Isolet	$69.83 \pm 1.74(617)^{-}$	$69.80 \pm 1.30(331)^{-}$
Leukemia	$6.85 \pm 5.00(7129)^{+}$	$7.57 \pm 3.51(3561)^{+}$
Lung	$22.51 \pm 5.58(56)^{+}$	$25.81 \pm 3.60(19)^{+}$
Madelon	$0.47 \pm 0.95(498)^{+}$	$1.85 \pm 0.05(267)^{+}$
Multiple	$67.64 \pm 3.93(216)^{=}$	$68.83 \pm 2.75(112)^{=}$
Ovarian	$51.46 \pm 11.25(1536)^{+}$	$50.33 \pm 5.90(754)^{+}$
Parkinson	$22.48 \pm 4.04(21)^{+}$	$21.57 \pm 0.95(9)^{+}$
Pima	$0.01 \pm 0.00(5)^{=}$	$0.55 \pm 0.55(4)^{-}$
Promoters	$5.52 \pm 7.59(2)^{+}$	$33.88 \pm 9.59(21)^{=}$
Robot	$29.71 \pm 11.72(89)^{+}$	$37.00 \pm 10.93(39)^{+}$
Segmentation	$60.73 \pm 1.71(19)^{=}$	$60.80 \pm 1.89(13)^{=}$
Soybean	$65.87 \pm 1.84(35)^{+}$	$62.65 \pm 3.07(21)^{+}$
Spect	$10.75 \pm 1.28(18)^{-}$	$4.99 \pm 1.19(15)^{+}$
Wdbc	$60.73 \pm 0.91(28)^{+}$	$50.42 \pm 0.50(9)^{+}$
Wine	$77.99 \pm 1.95(10)^{+}$	$79.97 \pm 7.67(10)^{=}$
Wisconsin	$71.60 \pm 0.00(8)^{=}$	$73.55 \pm 0.00(6)^{-}$
Average rank	5.16	3.95
(Win/Tie/Loss)	(13/4/2)	(11/4/4)
Data sets	SBI	RCE
Breast Tissu	$51.18 \pm 1.38(9)^{+}$	$57.19 \pm 2.14(3)$
Glass	$25.63 \pm 2.46(6)^{+}$	$36.73 \pm 2.21(4)$
Ionosphere	$12.62 \pm 2.37(34)^{=}$	$12.93 \pm 0.25(7)$
Isolet	$69.83 \pm 1.74(617)^{-}$	$66.91 \pm 0.20(3)$
Leukemia	$6.85 \pm 5.00(7129)^{+}$	$10.69 \pm 3.10(479)$
Lung	$22.51 \pm 5.58(56)^{+}$	$29.05 \pm 3.64(3)$
Madelon	$0.03 \pm 0.04(248)^{+}$	$3.14 \pm 0.00(4)$
Multiple	$67.88 \pm 4.16(215)^{=}$	$68.75 \pm 0.94(125)$
Ovarian	$51.46 \pm 11.25(1536)^{+}$	$68.28 \pm 1.50(1)$
Parkinson	$23.92 \pm 4.12(20)^{+}$	$28.67 \pm 0.87(2)$
Pima	$2.28 \pm 0.00(6)^{-}$	$0.01 \pm 0.01(2)$
Promoters	$3.44 \pm 4.84(46)^{+}$	$36.05 \pm 5.70(14)$
Robot	$20.19 \pm 4.29(75)^{+}$	$46.42 \pm 1.92(11)$
Segmentation	$60.33 \pm 1.20(17)^{+}$	$61.72 \pm 1.40(9)$
Soybean	$60.19 \pm 1.31(4)^{+}$	$69.02 \pm 1.84(15)$
Spect	$10.49 \pm 1.49(5)^{=}$	$8.99 \pm 1.26(3)$
Wdbc	$60.14 \pm 0.39(27)^{+}$	$62.15 \pm 0.00(5)$
Wine	$68.46 \pm 1.00(6)^{+}$	$79.06 \pm 0.47(6)$
Wisconsin	$65.71 \pm 0.00(5)^{+}$	$72.42 \pm 0.00(6)$
Average rank	5.5	2.34
(Win/Tie/Loss)	(14/3/2)	

$^{+}$ RCE is significantly better after t-tests at 95 % significance level

$^{-}$ RCE is significantly worse after t-tests at 95 % significance level

Table 4 Runtime (in seconds) comparison of RCE and CEFS on four representative data sets

Data sets	CEFS	RCE
Ionosphere	11.66	3.41
Isolet	10523.47	123.58
Leukemia	7539.03	12.98
Multiple Features	8251.70	49.57

Table 5 Experimental results using the SOM on a test data set. The Purity rate and the number of selected features in parenthesis are shown. Results for *lwd-SOM* and *lwo-SOM* are taken from Grozavu et al. (2009). *Bold* highlights the best results over all five algorithms

Data sets	All features	lwd-SOM	lwo-SOM	Unsupervised RF	RCE
Wave	61.00(3)	53.47(3)	54.16(3)	55.74(3)	**66.24**(3)
Wdbc	84.71(2)	62.74(9)	86.82(9)	84.71(2)	**88.40**(2)
Spamb	63.51(2)	61.03(2)	64.13(2)	65.07(2)	**66.64**(2)
Madelon	55.89(2)	52.42(2)	53.47(2)	52.35(2)	**58.78**(2)
Isolet	50.67(26)	52.42(13)	52.61(13)	21.36(26)	**55.68**(30)

cutting point where the cliff stops and the plain begins in the graph). In large dimensions, RCE reduced the feature set down be nearly $1/15$ its original size on Leukemia for instance, and by nearly $1/7$ compared to CEFS without any expense in terms of accuracy (RCE still performs best). More surprising, is the fact that RCE ended up with a single feature on Ovarian compared to 754 features with CEFS and RCE still performs best. Clearly, RCE method shows promise for scaling to larger domains.

4.2 RCE using the SOM as the base clustering algorithm

In this section, the SOM (Kohonen 2001) was applied to generate the partitions for the combination. For the map clustering, we used the ward-link hierarchical classification. Again, the size of the clustering ensemble r was set to 200. To evaluate the quality of our feature selection procedure, we compared the clusters found with the SOM on the features selected by RCE to the clusters returned by (1) the SOM on all the features, (2) the SOM on the features selected by the unsupervised RF algorithm (Breiman and Cutler 2003) and (3) two recently embedded unsupervised feature selection methods based on the SOM, called *lwd-SOM* and *lwo-SOM* (Grozavu et al. 2009) in which the feature weights are adjusted in the course of the learning process. The results presented here for *lwd-SOM* and *lwo-SOM* are directly taken from Grozavu et al. (2009) as their code is not freely available. The comparison is restricted to the five data sets that were used in their study, namely: (*Wave*, *Wdbc*, *Spamb*, *Madelon* and *Isolet*). As in Grozavu et al. (2009), the Davies Bouldin index was used to choose the optimal number of clusters and the quality assessment was based on the *Purity rate* as the *NMI index* was not given in their work.

Accuracy results are reported in Table 5. As may be observed, RCE clearly outperforms *lwd-SOM* and *lwo-SOM* by a noticeable margin in respect of the *Purity rate*, on all four data sets. RCE is significantly better then all four approaches ($p < 0.05$) according to the two-tailed sign test (insufficient number of data sets for the Freidman test) (Demsar 2006). These experiments with another standard "base" clustering algorithm confirm the effectiveness of the RCE framework for unsupervised learning, and the usefulness of the consensus clustering combined with oob feature importance. As may be observed, the purity of the

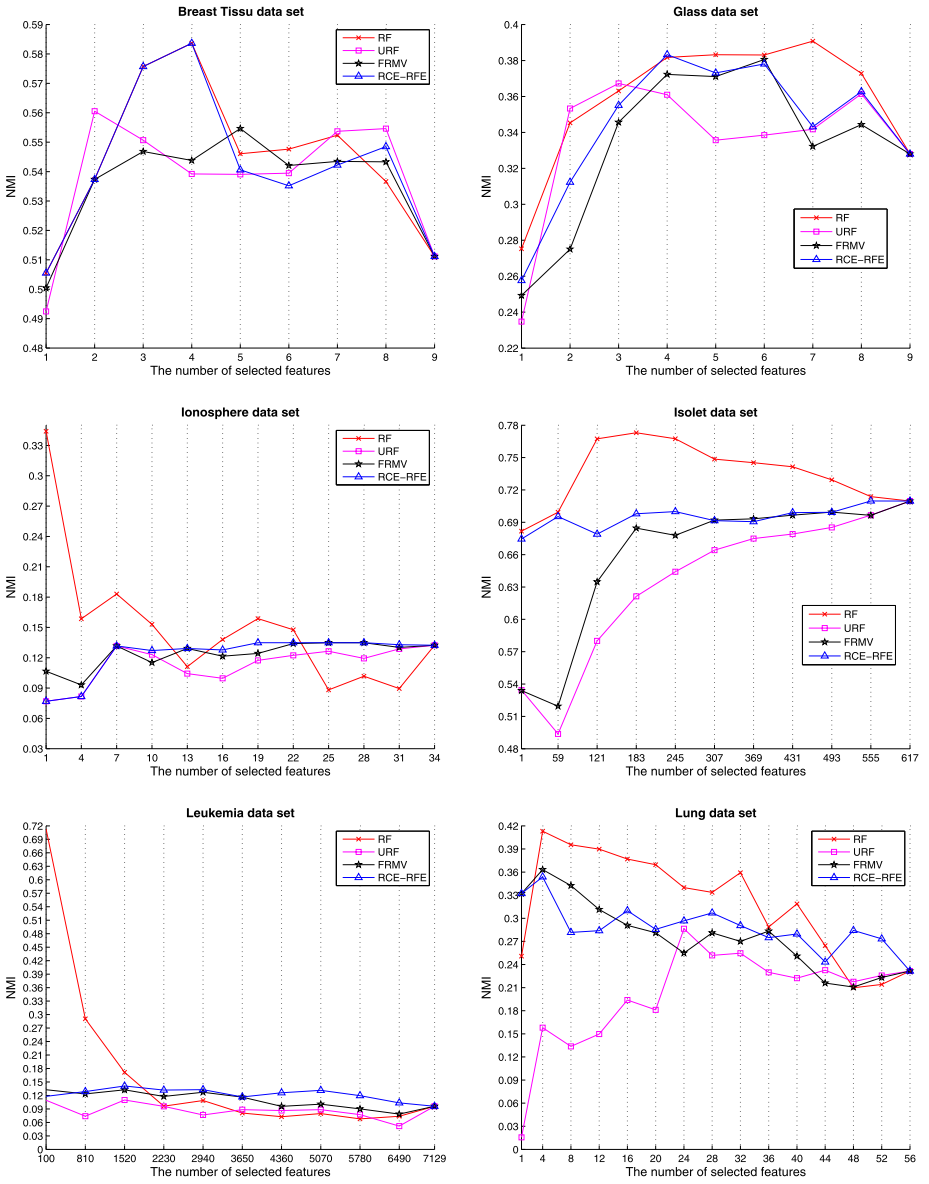

Fig. 1 Clustering accuracies (NMI) as a function of the features ranked in decreasing order of importance with RCE, RCE-RFE, FRMV and RF

SOM clustering on the features selected by RCE was always better than those obtained by *lwd-SOM*, *lwo-SOM* and *unsupervised RF*.

4.3 RCE-RFE using k-means as the base clustering algorithm

In this section, we boosted the performance of RCE using the RFE strategy, as discussed before, and monitored the behavior of the clustering against the features in decreasing order

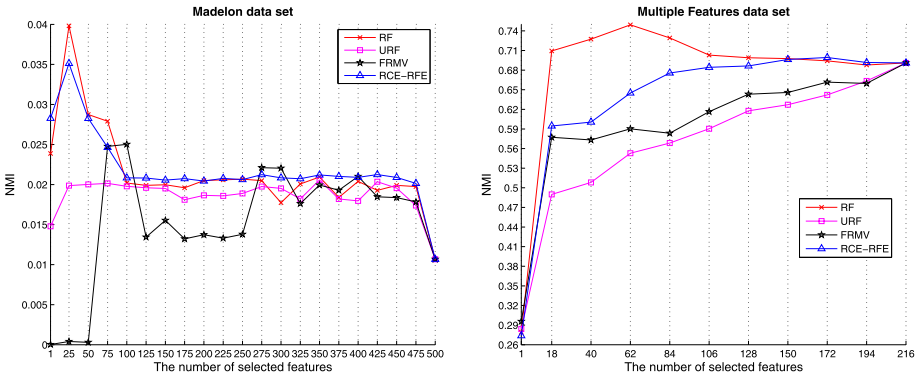

Fig. 1 (*Continued*)

of importance. We applied again the *k-means* clustering algorithm as the "base" clustering algorithm. The number of clusters was set to the true number of labels and the size of clustering ensemble T was set to 200 again. The data sets used in these experiments are exactly those used in Sect. 4.1. The quality of the feature ranking given by RCE-RFE was compared against three two unsupervised feature ranking algorithms and one gold standard supervised feature ranking algorithm:

1. Unsupervised feature ranking from multiple views (FRMV) based on symmetrical uncertainty (Hong et al. 2008b). The number of feature rankings for FRMV was equal to 100 as suggested by their paper.
2. Unsupervised Random Forest (URF) (Breiman and Cutler 2003).
3. Random Forest (RF) (Breiman 2001) taken as our gold standard ensemble supervised feature ranking approach. The number of trees was set to 500.

To make fair comparisons, the same experimental approach (protocol and evaluation measure) in Hong et al. (2008a, 2008b) was adopted here. We plotted the NMI values of the above four approaches against the features in decreasing order of importance. The NMI Index was averaged over 20 independent runs. Experimental results are reported in Figs. 1, 2 and 3. As may be seen, RCE-RFE works consistently better than the other two unsupervised feature ranking algorithms URF and FRMV. A closer inspection of the plots reveals that the accuracy of k-means on the features selected by RCE-RFE generally increases swiftly at the beginning (the number of selected feature is small) and slows down afterwards. This suggests that RCE-RFE ranks the most relevant features first. We also computed the average gain in clustering accuracy of RCE-RFE versus URF and FRMV computed as the number of features is varied from 1 to 33 % of the total features to get another point of view. Results are depicted in Table 6. As may be observed, the RCE-RFE generally dominates URF and FRMV. The Friedman test and the Bonferroni-Dunn post-hoc test reveal statistically significant improvements ($p < 0.05$) in NMI for RCE-RFE.

Another interesting phenomenon observed from Figs. 1, 2 and 3 is that RCE-RFE and the supervised feature ranking approach (RF) have a tendency to work comparatively well on most data sets, except Ionosphere, Isolet, Leukemia, Pima, Multiple Features. More importantly, RCE-RFE performs very well with very few features as shown for instance on Segmentation. On this data set, RCE-RFE identified thee features $\{f_2, f_{12}, f_{17}\}$ that yield meaningful clusters with an NMI of 65.01 %, while supervised RF yields only 37.21 % for the same number of features. The same tendency can also be observed with Ovarian, Spect

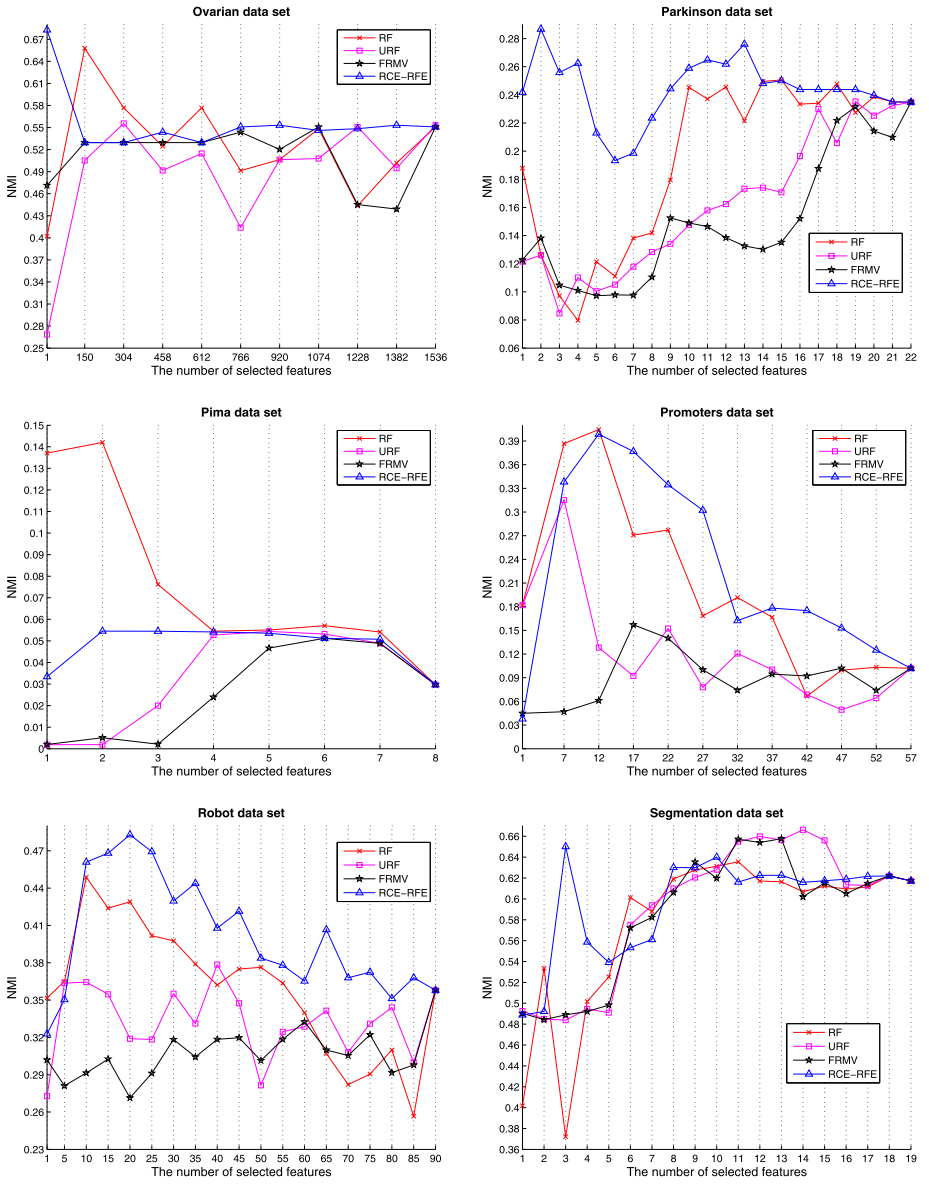

Fig. 2 Clustering accuracies (NMI) as a function of the features ranked in decreasing order of importance with RCE, RCE-RFE, FRMV and RF

and Segmentation. With Ovarian, the feature $\{f_9\}$ is the most important feature selected by RCE-RFE, and yields a clustering of 68.28 % which is far better off the 40.20 % accuracy obtained with the most important feature selected by supervised RF. The same remark holds for Robot, Parkinson and Spect data sets.

We also compare RCE-RFE against the Clustering Ensemble guided FS algorithm (CEFS) (Hong et al. 2008a) and RCE. In CEFS, the underlying principle is to search for

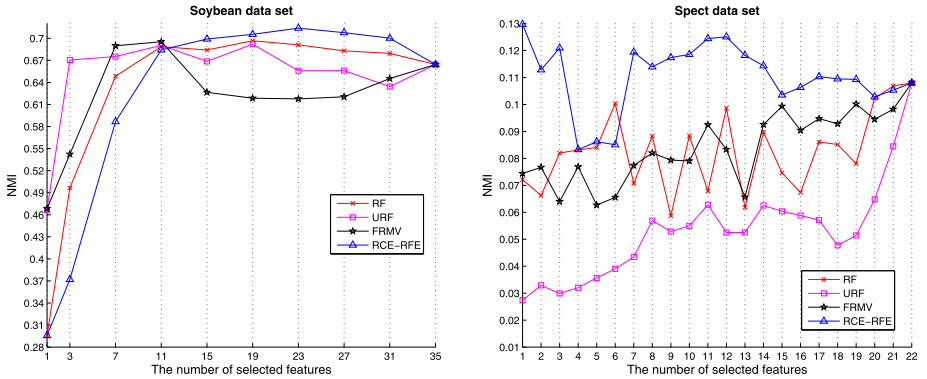

Fig. 2 (*Continued*)

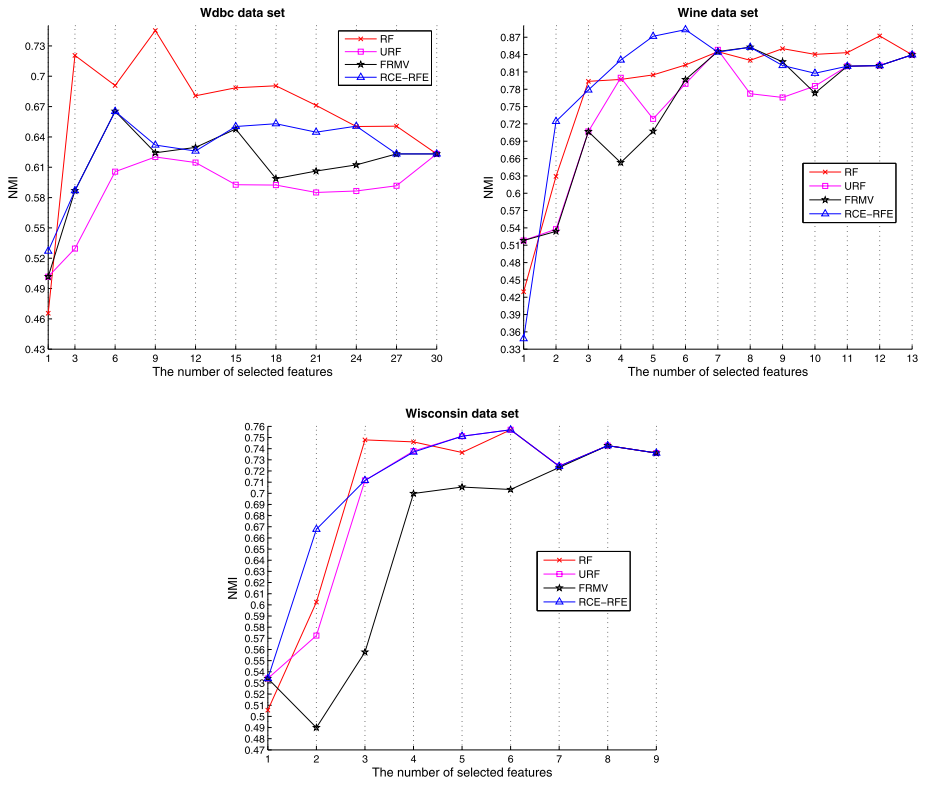

Fig. 3 Clustering accuracies (NMI) as a function of the features ranked in decreasing order of importance with RCE, RCE-RFE, FRMV and RF

the feature subset such that the clustering algorithm trained on this feature subset achieves the most similar clustering solution to the one obtained by an ensemble learning algorithm. It returns directly a subset of features. The Scree Test was used to selected the best features output by RCE. We used the parameter settings used in Hong et al. (2008a) in these ex-

Table 6 Comparison of the average gain in NMI between RCE-RFE versus URF and FRMV computed as the number of features is varied from 1 to 33 % of the total features

Data sets	URF	FRMV
Breast Tissue	+0.0148	+0.0185
Glass	−0.0020	+0.0165
Ionosphere	−0.0200	−0.0080
Isolet	+0.0919	+0.0935
Leukemia	+0.0274	−0.0023
Lung	+0.1570	−0.0130
Madelon	+0.0064	+0.0141
Multiple Features	+0.0698	+0.0194
Ovarian	+0.1161	+0.0564
Parkinson	+0.1266	+0.1275
Pima	+0.0375	+0.0444
Promoters	+0.0971	+0.2075
Robot	+0.1026	+0.1574
Segmentation	+0.0324	+0.0335
Soybean	−0.1171	−0.0909
Spect	+0.0710	+0.0343
Wdbc	+0.0804	+0.0373
Wine	+0.0298	+0.0675
Wisconsin	+0.0318	+0.1106
Average	+0.0502	+0.0486

Table 7 Comparison of NMI values (%) of RCE-RFE and CEFS on different data sets, averaged over 20 independent runs. The number of selected features is determined by CEFS unsupervised feature selection algorithm. *Bold* highlights the best results over the two algorithms

Data sets	Number of features	CEFS	RCE-RFE
Breast Tissue	7	53.81	**54.22**
Glass	4	28.44	**38.33**
Ionosphere	22	**14.00**	13.49
Isolet	331	69.80	**69.83**
Leukemia	3561	7.57	**11.80**
Lung	19	25.81	**29.63**
Madelon	267	1.85	**2.12**
Multiple Features	112	68.83	**69.79**
Ovarian	754	50.33	**54.84**
Parkinson	9	21.57	**24.44**
Pima	4	0.55	**5.42**
Promoters	21	33.88	**38.23**
Robot	39	37.00	**39.98**
Segmentation	13	60.80	**62.27**
Soybean	21	62.65	**70.82**
Spect	15	4.99	**10.35**
Wdbc	9	50.42	**63.20**
Wine	10	79.97	**80.78**
Wisconsin	6	73.55	**75.69**

periments. In order to make fair comparisons, we compared RCE-RFE and CEFS based on the same number of features returned by CEFS. The results are shown in Tables 7. Similarly, we compared RCE-RFE and RCE based on the same number of features returned by

Table 8 Comparison of NMI values (%) of RCE-RFE and RCE on different data sets, averaged 20 independent runs. The number of selected features is determined by RCE unsupervised feature selection algorithm. *Bold* highlights the best results over the two algorithms

Data sets	Number of features	RCE	RCE-RFE
Breast Tissue	3	57.19	**57.57**
Glass	4	36.73	**38.33**
Ionosphere	7	12.93	12.93
Isolet	3	66.91	**69.05**
Leukemia	479	10.69	**11.89**
Lung	3	29.05	**30.85**
Madelon	4	3.14	**3.88**
Multiple Features	125	68.75	**69.11**
Ovarian	1	68.28	68.28
Parkinson	2	28.67	28.67
Pima	2	0.01	**5.45**
Promoters	14	36.05	**38.47**
Robot	11	46.42	**46.65**
Segmentation	9	61.72	**62.98**
Soybean	15	69.02	**69.90**
Spect	3	8.99	**12.10**
Wdbc	5	62.15	**65.67**
Wine	6	79.06	**88.31**
Wisconsin	6	72.42	**75.69**

RCE and the Scree test. Results are shown in Table 8. In both cases, a two-tailed sign test reveals statistically significant improvements ($p < 0.0001$) in accuracy for RCE with the selected features. The improvement obtained with RCE-RFE over RCE shows that the relative importance of each feature varies as the stepwise elimination process progresses. This is particularly true for high dimensional data sets such as Isolet, Leukemia, Lung, Promoters and Spect.

4.4 Effect of noisy features on RCE-RFE performances

In this Section, we investigated the robustness of RCE-RFE as many irrelevant features are added to the original feature set. The k-means algorithm was used again as the base clustering algorithm. We consider the well known Iris dataset as it is well understood in terms of feature relevance. This dataset has three classes, 150 instances, and 4 features. As reported in Yuanhong et al. (2008), Hong et al. (2008a) and illustrated in Fig. 4, among these 4 features, features f_3 and f_4 suffice to differentiate the three original classes very well. We conducted several experiments on this data set in order to study the impact of adding noisy features on the performance of RCE-RFE. We first performed a feature ranking on the original data set; then $10^i (i \in \{1, 2, 3, 4, 5\})$ normally distributed variables with mean 0 and variance 1 were added sequentially to the feature set and RCE-RFE was ran again.

To gauge the practical relevance of RCE-RFE feature selection method in high dimension, the quality of the two most important features were evaluate at each time step i. The NMI index of the clustering obtained by k-means was indicative of the quality of selected features. The performance of RCE-RFE was compared to (1) Unsupervised feature ranking from multiple views (FRMV) based on symmetrical uncertainty, (2) Unsupervised Random Forest (URF), and (3) Random Forest (RF). The clustering accuracy is plotted against the

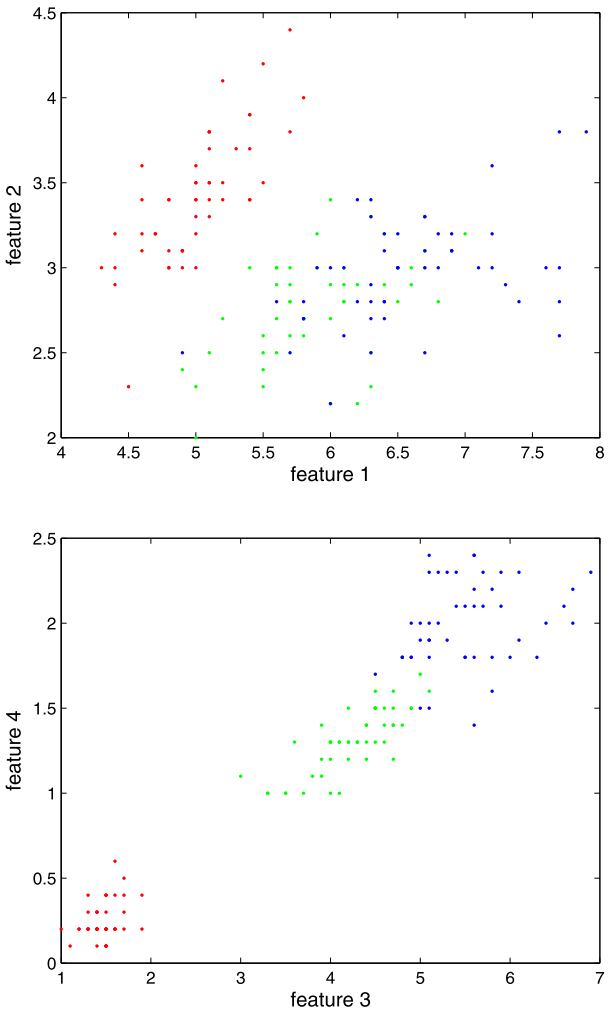

Fig. 4 Scatterplots on iris data using features 1 and 2 (*top*), and using features 3 and 4 (*bottom*). Data from different classes are marked with *different colors* (Color figure online)

number of noisy variables, up to 100000 features, in Fig. 5 for each method. As may be seen, the performance of FRMV and URF deteriorated markedly with an increasing number of noisy features. Surprisingly, RCE-RFE performed as well as the supervised RF feature ranking that has access to the class labels, on this data set. Indeed, we found that RCE-RFE was able to remove all irrelevant features and ranked first the two most relevant features $\{f_3, f_4\}$ that yields a meaningful clustering with a NMI index of 86.42 %. Therefore, RCE-RFE shows promise to deal with very large domains. In our opinion, this is the most interesting result of this study.

5 Conclusion

In this paper, we proposed a generic framework called RCE for estimating the feature importance in unsupervised learning, using an ensemble of clustering algorithms. We assessed

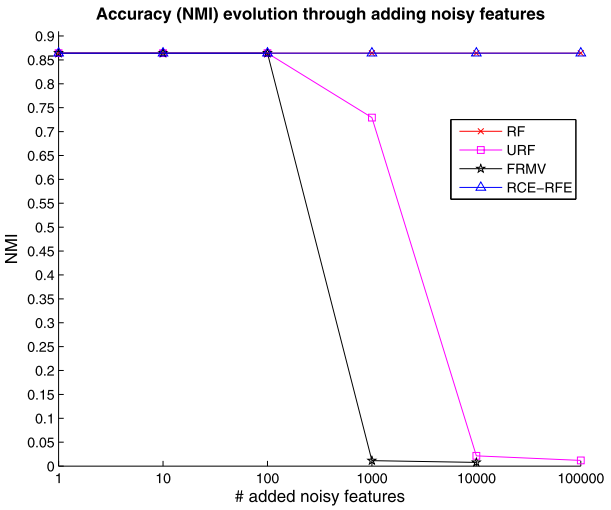

Fig. 5 Clustering accuracy as a function of the number of irrelevant variables in the input

the accuracy of the feature selection procedure on nineteen UCI data sets and compared its effectiveness against powerful unsupervised and supervised feature selection methods. The significant gain in accuracy on these data sets, expressed as the NMI index and the purity index, confirmed the ability of RCE to generate meaningful clusters with a very few features. Also instructive was the fact that, in high dimensional feature space, RCE reduced the feature set down by nearly $1/100$ its original size without any expense in terms of clustering accuracy. RCE "wrapped around" with the Recursive Filter Elimination scheme was shown to significantly outperform several unsupervised feature selection procedures that appeared recently in the literature. More importantly, RCE-RFE was shown to perform as well as Random Forest on the Iris data set to which we increasely added up to 100000 noisy features. The method shows promise to deal with very large domains. Future substantiation through more experiments on biological databases containing several thousands of variables are currently being undertaken.

While the emphasis in this paper was on estimating feature importance in unsupervised learning, it is worth mentioning that the idea underlying the permutation-based out-of-bag feature importance measure may also be extended to semi-supervised learning feature importance evaluation as shown recently (Bellal et al. 2012; Barkia et al. 2011).

References

Barkia, H., Elghazel, H., & Aussem, A. (2011). Semi-supervised feature importance evaluation with ensemble learning. In *11th IEEE international conference on data mining, ICDM'11*, Vancouver (pp. 31–40).

Bellal, F., Elghazel, H., & Aussem, A. (2012). A semi-supervised feature ranking method with ensemble learning. *Pattern Recognition Letters*, *33*(10), 1426–1432.

Blake, C., & Merz, C. (1998). *Uci repository of machine learning databases*.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32.

Breiman, L., & Cutler, A. (2003). *Random forests manual v4.0*. Technical report, UC Berkeley. http://oz.berkeley.edu/users/breiman/Using_random_forests_v4.0.pdf.

Cattell, R. B. (1966). The scree test for the number of factors. *Multivariate Behavioral Research*, *2*, 245–276.

Chen, X., Ye, Y., Xu, X., & Huang, J. Z. (2012). A feature group weighting method for subspace clustering of high-dimensional data. *Pattern Recognition*, *45*(1), 434–446.

Dash, M., Choi, K., Scheuermann, P., & Liu, H. (2002). Feature selection for clustering-a filter solution. In *IEEE international conference on data mining* (pp. 115–122).

Demsar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, *7*, 1–30.

Dudoit, S., & Fridlyand, J. (2003). Bagging to improve the accuracy of a clustering procedure. *Bioinformatics*, *19*(9), 1090–1099.

Dy, J., & Brodley, C. (2004). Feature selection for unsupervised learning. *Journal of Machine Learning Research*, *5*, 845–889.

Elghazel, H., & Aussem, A. (2010). Feature selection for unsupervised learning using random cluster ensembles. In *IEEE International Conference on Data Mining* (pp. 168–175).

Fred, A., & Jain, A. (2002). Data clustering using evidence accumulation. In *16th international conference on pattern recognition* (pp. 276–280).

Fred, A., & Jain, A. (2005). Combining multiple clusterings using evidence accumulation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *27*(6), 835–850.

Frigui, H., & Nasraoui, O. (2004). Unsupervised learning of prototypes and attribute weights. *Pattern Recognition*, *37*(3), 567–581.

Ghaemi, R., Sulaiman, N., Ibrahim, H., & Mustapha, N. (2009). A survey: clustering ensembles techniques. In *Engineering and technology* (Vol. 38). Singapore: World Scientific.

Golub, T., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J., & Coller, H. (1999). Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, *286*, 531–537.

Grozavu, N., Bennani, Y., & Lebbah, M. (2009). From variable weighting to cluster characterization in topographic unsupervised learning. In *IEEE international joint conference on neural network* (pp. 1005–1010).

Gullo, F., Talukder, A. K. M. K., Luke, S., Domeniconi, C., & Tagarelli, A. (2012). Multiobjective optimization of co-clustering ensembles. In *Genetic and evolutionary computation conference, GECCO'12* (pp. 1495–1496).

Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, *3*, 1157–1182.

Hong, Y., Kwong, S., Chang, Y., & Qingsheng, R. (2008a). Unsupervised feature selection using clustering ensembles and population based incremental learning algorithm. *Pattern Recognition*, *41*(9), 2742–2756.

Hong, Y., Kwong, S., Chang, Y., & Ren, Q. (2008b). Consensus unsupervised feature ranking from multiple views. *Pattern Recognition Letters*, *29*(5), 595–602.

Hua, J., Tembe, W., & Dougherty, E. (2009). Performance of feature-selection methods in the classification of high-dimension data. *Pattern Recognition*, *42*, 409–424.

Huang, J., Ng, M., Rong, H., & Li, Z. (2005). Automated variable weighting.in k-means type clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *27*(5), 657–668.

Jing, L., Ng, M. K., & Huang, J. Z. (2007). An entropy weighting k-means algorithm for subspace clustering of high-dimensional sparse data. *IEEE Transactions on Knowledge and Data Engineering*, *19*(8), 1026–1041.

Kluger, Y., Basri, R., Chang, J. T., & Gerstein, M. (2003). Spectral biclustering of microarray cancer data: co-clustering genes and conditions. *Genome Research*, *13*, 703–716.

Kohonen, T. (2001). *Self-organizing maps* (Vol. *30*). Berlin: Springer.

Meila, M. (2005). Comparing clusterings: an axiomatic view. In *Proceedings of the twenty-second international conference on machine learning (ICML 2005)*, Bonn, Germany, August 7–11, 2005 (pp. 577–584).

Mitra, P., Murthy, A., & Pal, S. (2002). Unsupervised feature selection using feature similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*(3), 301–312.

Morais, S. R. D., & Aussem, A. (2010). A novel Markov boundary based feature subset selection algorithm. *Neurocomputing*, *73*(4–6), 578–584.

Morita, M., Sabourin, R., Bortolozzi, F., & Suen, C. (2003). Unsupervised feature selection using multiobjective genetic algorithms for handwritten word recognition. In *International conference on document analysis and recognition* (pp. 666–670).

Saeys, Y., Inza, I., & Larrañaga, P. (2007). A review of feature selection techniques in bioinformatics. *Bioinformatics*, *23*, 95–116.

Saeys, Y., Abeel, T., & de Peer, Y. V. (2008). Robust feature selection using ensemble feature selection techniques. In *ECML PKDD* (pp. 313–325).

Schummer, M., Ng, W. V., & Bumgarnerd, R. E. (1999). Comparative hybridization of an array of 21,500 ovarian cdnas for the discovery of genes overexpressed in ovarian carcinomas. *Gene*, *238*(2), 375–385.

Shi, T., & Horvath, S. (2006). Unsupervised learning with random forest predictors. *Journal of Computational and Graphical Statistics*, *15*(1), 118–138.

Strehl, A., & Ghosh, J. (2002). Cluster ensembles-a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, *3*, 583–617.

Topchy, A., Jain, A., & Punch, W. (2005). Clustering ensembles: models of consensus and weak partitions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *27*(12), 1866–1881.

Tuv, E., Borisov, A., Runger, G. C., & Torkkola, K. (2009). Feature selection with ensembles, artificial variables, and redundancy elimination. *Journal of Machine Learning Research*, *10*, 1341–1366.

Wang, P., Laskey, K. B., Domeniconi, C., & Jordan, M. (2011). Eleventh SIAM international conference on data mining. In *SDM 2011* (pp. 331–342).

Yuanhong, L., Ming, D., & Jing, H. (2008). Localized feature selection for clustering. *Pattern Recognition Letters*, *29*(1), 10–18.