Information Sciences xxx (2014) xxx–xxx

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

# Feature selection with SVD entropy: Some modification and extension

Monami Banerjee, Nikhil R. Pal *

*Electronics and Communication Sciences Unit, Indian Statistical Institute, Calcutta 700 108, India*

ARTICLE INFO

*Article history:*
Received 21 November 2012
Received in revised form 25 November 2013
Accepted 24 December 2013
Available online xxxx

*Keywords:*
Feature selection
Singular Value Decomposition
Entropy

ABSTRACT

Many approaches have been developed for dimensionality reduction. These approaches can broadly be categorized into supervised and unsupervised methods. In case of supervised dimensionality reduction, for any input vector the target value is known, which can be a class label also. In a supervised approach, our objective is to select a subset of features that has adequate discriminating power to predict the target value. This target value for an input vector is absent in case of an unsupervised approach. In an unsupervised scheme, we mainly try to find a subset that can capture the inherent "structure" of the data, such as the neighborhood relation or the cluster structure. In this work, we first study a Singular Value Decomposition (SVD) based unsupervised feature selection approach proposed by Varshavsky et al. Then we propose a modification of this method to improve its performance. An SVD-entropy based supervised feature selection algorithm is also developed in this paper. Performance evaluation of the algorithms is done on altogether 13 benchmark and one Synthetic data sets. The quality of the selected features is assessed using three indices: Sammon's Error (SE), Cluster Preservation Index (CPI) and misclassification error (MCE) using a 1-Nearest Neighbor (1-NN) classifier. Besides showing the improvement of the modified unsupervised scheme over the existing one, we have also made a comparative study of the modified unsupervised and the proposed supervised algorithms with one well-known unsupervised and two popular supervised feature selection methods respectively. Our results reveal the effectiveness of the proposed algorithms in selecting relevant features.



## 1. Introduction

Dimensionality reduction is frequently used as a pre-processing step in data mining. Selecting a smaller number of features carries a significant role in applications involving hundreds or thousands of features. Bioinformatic applications such as gene expression analysis involving a huge number of features and comparatively smaller number of samples [9] lead to the known problem of "curse of the dimensionality". Therefore, selection of a small number of discriminative genes from thousands of genes becomes essential for designing successful diagnostic classification systems [7,13,27,28,37,38].

Among the features present in a data set, there could be features which are derogatory and degrade the performance of a classifier or there could be features whose presence in the data set does not affect the performance of a classifier at all. There could even be some correlated set of features and selection of just a few of them might be sufficient for the classifier. In brief, besides relevant features, there might be derogatory features, indifferent features, and redundant (dependent) ones. Removal

* Corresponding author.
*E-mail addresses:* monamie.b@gmail.com (M. Banerjee), nikhil@isical.ac.in (N.R. Pal).

0020-0255/$ - see front matter © 2014 Published by Elsevier Inc.
http://dx.doi.org/10.1016/j.ins.2013.12.029

of these features not only makes the learning task easier, by reducing computational constraint but also often improves the performance of the classifier.

Dimensionality reduction can be done mainly in two ways: either by selecting a subset of the original feature set or by extracting a lower number of features preserving the characteristics of the original higher dimensional data set. In feature selection we limit ourselves to selection of a particular subset of the original features, while (linear/nonlinear) combination of features appears in the later case.

The feature selection problem can be formulated as follows: Given a data set $X \subseteq \mathbb{R}^{n\times p}$ we have to select $r$ "important" features such that some criterion is optimized, where $r < p$. The notion of importance usually depends on the target application. One such criterion can be the preservation of topology of the data set or the performance of a classifier in the reduced dimension. According to Kohavi and John [20], feature selection schemes can be broadly classified into two groups, Wrapper Models and Filter Models. Wrapper methods have a well-specified objective function, which should be optimized through selection of a subset of features, whereas, filter models use some underlying properties of the features.

Dimensionality reduction methods can also be broadly categorized into two families *Supervised* and *Unsupervised*. Supervised approaches require a target or output value associated with each data point (e.g. the class label) while unsupervised approaches do not demand such target values. Feature selection using supervised approaches is easier because the goal is to select a small subset of features so that the problem of prediction/classification can be done satisfactorily. For unsupervised methods, since there is no target application, some task independent criterion should be used to select features. There are many methods under both supervised and unsupervised families [20,31,6,12,10,36,5].

In this work, we have studied an existing SVD-entropy (Singular Value Decomposition entropy) based unsupervised feature selection technique, proposed by Varshavsky et al. [39] and have modified it to improve its performance. In addition, a comparison with another well-known unsupervised algorithm, $\ell_{2,1}$-*norm Regularized Discriminative Feature Selection* (UDFS) [40] is also given. We have then extended this SVD-entropy based approach to a supervised framework. A comparative study of this supervised method with two existing supervised algorithms, *Multi-Class SVM-Recursive Feature Elimination* (MSVM-RFE) [41] and *Redundancy Constrained Feature Selection* (RCFS), [42] proves its effectiveness.

The rest of the work is organized as follows. In the next section we describe some existing feature selection approaches. In Section 3, descriptions of the indices used in this work to assess the selected features are given. Some preliminary discussions on singular values and singular vectors are included in Section 4. The SVD-entropy based unsupervised feature selection method [39] is discussed and analyzed in Section 5. The modified unsupervised approach is elaborated in Section 5.1. The computational complexity of this scheme is analyzed in Section 6, while Section 7 reports some experimental results. Besides showing the improvement of the modified unsupervised algorithm over the existing one, in Section 8 we have also made a comparative study of it with UDFS [40]. In Section 9 our proposed supervised scheme is explained. Some experimental results on the supervised method and its comparisons with MSVM-RFE [41] and RCFS [42] are given in Sections 10 and 11 respectively. Some conclusive notes are mentioned in Section 12.

## 2. Some existing feature selection techniques

For a given data set, let $F$ be a set of features and $G$ be some subset of $F$. In general, the goal of any feature selection technique can be formalized as selecting a minimum subset $G$ such that $P(C|G)$ is equal or as close as possible to $P(C|F)$, where $P(C|G)$ and $P(C|F)$ are the respective probability distributions of different classes ($C$) given the feature values in $G$ and $F$ [21]. We believe that the constraint of the minimum subset should be relaxed to accommodate some redundancy so that any system designed using the selected features can tolerate some measurement error in features. In presence of hundreds of features, researchers have noticed that usually a large number of features are not informative because they are either irrelevant or redundant with respect to the class concept or may carry "negative" information (derogatory in nature). By removing such non-essential features not only the performance of the system is improved but also the learning task becomes simpler.

Devaney and Ram [6] proposed an unsupervised feature ranking scheme. Based on the work of Gluck and Corter [12], they formulated a measure called Category utility to measure the goodness of a feature subset. A feature ranking method using Fisher's measure [10] for categorical data has been proposed by Talavera [36]. This algorithm is based on the assumption that features, which have higher dependencies with other features, are going to be more relevant from clustering point of view. Hence they ranked features based on a measure of dependency [36]. Pena et al. [29] proposed a similar feature selection scheme. They considered features, having low correlation with other features, as irrelevant and they assigned a relevance measure to each feature. Authors pre-processed the training data by selecting only features having relevance score higher than a threshold. Then the Conditional Gaussian Networks (CGN) were used in selection of features. Dash et al. [5] proposed an entropy based unsupervised scheme for both categorical and numerical data. This method iteratively rejected the least important feature based on a sequential backward selection scheme.

Saxena et al. [31] proposed a Genetic Algorithm based feature selection scheme using Sammon's stress/error as fitness function. Then the data set in the reduced dimension was evaluated by using classification (1-Nearest Neighbour) and clustering ($K$-means) techniques. He et al. [15] proposed a feature selection scheme based on *Laplacian Score*, which can be incorporated in both supervised and unsupervised frameworks.

The unsupervised feature selection scheme proposed by Hong et al. [17] first uses a cluster ensemble method to combine different clustering solutions obtained by different clustering algorithms on the data set. Then it tries to select a subset of

Q1 Please cite this article in press as: M. Banerjee, N.R. Pal, Feature selection with SVD entropy: Some modification and extension, Inform. Sci. (2014), http://dx.doi.org/10.1016/j.ins.2013.12.029

features for which the resultant clustering in the reduced dimension has the highest similarity with the ensembled clustering.

Boutsidis et al. [3] have also proposed an unsupervised feature selection algorithm. Given a data set $X_{[n\times p]}$, first the top $l$ right singular vectors ($V_l$s) are computed, where $l$ is a constant. For each feature $i$, the (normalized) leveraged score $\gamma_i$ [4,24] is computed as the square of the Euclidean norm of the $i$th row as follows: $\gamma_i = ||(V_l)_{(i)}||^2/l$.

A sampling parameter $r$ is chosen as $r = \Theta(l\log(l/\epsilon)/\epsilon^2)$. Then for $r$ i.i.d random trials, the $i$th feature is kept with probability $\gamma_i$. The $i$th feature is then multiplied by $(r\gamma_i)^{-\frac{1}{2}}$. The $n \times r$ matrix is returned as the re-scaled reduced data set. Mao [25] proposed an unsupervised feature selection approach based on selecting important variables using principal components.

A Support Vector Machine (SVM) based feature selection algorithm is proposed by Guyon et al. [14]. Their method is a backward elimination technique which starts with all features and eliminates one feature at a time. The decision function of an SVM is $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b$ where $\mathbf{w} = [w_1, w_2, \ldots, w_p]^T$ is the weight vector and $b$ is a scalar, $p$ is the number of features. They rank the features based on the square of the weights ($w_i$s) associated with each feature, the higher the weight value, the better the rank. The feature with the smallest rank is rejected. This process is repeated until the set contains the desired number of features.

This two-class scheme of Guyon et al. [14] is extended in a $k$-class ($k \geqslant 2$) framework by Zhou et al. [41]. They call it Multi-Class SVM-RFE (MSVM-RFE). They train $k$ binary SVMs based on *One versus All* (OVA) strategy. From these $k$ classifiers we get $k$ weight vectors $\mathbf{w}_j = [w_{j1}, w_{j2}, \ldots, w_{jp}]^T$; $j = 1, \ldots, k$. The ranking criteria for the $i$th feature is taken as the sum of the squares of the $i$th coefficient of $k$ weight vectors, $\sum_{j=1}^{k}(w_{ji})^2$. As before, after eliminating the smallest ranked feature, the process is repeated until the desired number of features is reached.

Zhou et al. [42] proposed a redundancy constrained feature selection approach. As trace based methods cannot take into account the redundancy between features [42], they have formulated it as a constrained 0–1 linear fractional program (LFP). Given a set of training samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ with $p$ features and the desired number of features, $r$, as the input, an hierarchical agglomerative clustering of features is used to generate $r$ clusters of features. Then from each of this $p - r + 1$ levels of hierarchy, $r$ features are selected using linear fractional programming. For each feature $i$, the usual between-class scatter ($f_i$) and the total-class scatter ($g_i$) are calculated. For each level of the hierarchical clustering, a 0–1 linear fractional optimization problem is formulated as follows:

$$\boldsymbol{\omega}^* = \underset{\boldsymbol{\omega}}{\text{argmax}}\{(\mathbf{f}^T\boldsymbol{\omega})/(\mathbf{g}^T\boldsymbol{\omega})\} \tag{1}$$

subject to the following constraints. $\omega \in [0,1]^n$; $\mathbf{1}^T\boldsymbol{\omega} = r$; and $\sum_{k=1}^{p_i}\omega_{i_k} \leqslant \gamma_i,\ \forall i$ clusters. where $p_i$ is the size of the $i$th cluster. $\gamma_i$ is taken as 1 enforcing that at most one feature could be selected from each cluster. This constraint is to control redundancy between features. They selected $r$ features from each of $p - r + 1$ levels by Dinkelbach's algorithm for linear programming [8,26,33]. Dinkelbach's algorithm is based on the totally unimodular (TUM) condition in linear programming [32]. Among the all feature subsets of size $r$, the one whose ten-fold cross validation error with SVM classifier is the minimum, is taken as the best feature set. We note that if the features are centered and standardized, then $g_i = 1;\ i = 1, \ldots, p$.

In recent years, various sparsity induced feature selection schemes have been proposed by researchers [19,40,22,23]. Yang et al. [40] proposed an $\ell_{2,1}$-norm regularized unsupervised feature selection method (UDFS). Given a data set $X$ with $n$ data points, a set of $m$ local neighbors of each point is obtained. The local discriminative score $DS_i$ of $i$th data point is defined as

$$DS_i = Trace\left[\left(S_t^i + \lambda I\right)^{-1} S_b^i\right]$$

where $S_t^i$ and $S_b^i$ are respectively the local total and between scatter matrices, which are computed using the $k$-nearest neighbors of the $i$th data points. In order to make the matrix invertible $\lambda I$ is added. Authors have assumed a linear classifier $W$ that classifies each data point to a class. A higher value of $DS_i$ indicates a better discriminating property of $W$ for the $i$th point. To select features the following objective function is minimized:

$$\sum_i \left\{Trace\left[G_{(i)}^T H_{k+1} G_{(i)}\right] - DS_i\right\} + \gamma||W||_{2,1}$$

where $G_{(i)}$ is a set of $k$ classifiers of the form $G_i = W^T\mathbf{x}_i$, $H_{k+1}$ is a centering matrix and $W$ is of size $p \times k$ where $p$ and $k$ are the number of features and classes respectively. The objective function can be further simplified as follows:

$$\min_{W^TW=I} Trace(W^TMW) + \gamma||W||_{2,1}$$

where $M$ is defined using the data matrix, the local neighborhood data matrix and the centering matrix [40]. Here $||W||_{2,1}$ is the $\ell_{2,1}$-norm of $W$. The features are ranked in non-increasing order based on $||W||_{2,1}$. The top $r$ features are selected to form the reduced dimensional space.

Another unsupervised $\ell_{2,1}$ norm based feature selection scheme is proposed by Li et al. [23]. In this work spectral clustering criterion [34] is used to form the objective function. Like the work in [40] here also the features are ranked based on the $\ell_{2,1}$ norm of the feature selection matrix. These norm based methods are expected to reduce the redundancy in



Q1 Please cite this article in press as: M. Banerjee, N.R. Pal, Feature selection with SVD entropy: Some modification and extension, Inform. Sci. (2014), http://dx.doi.org/10.1016/j.ins.2013.12.029

the selected features. However, the authors did not claim this property, and our experiments also showed that they are not very effective in removing redundancy.

We have made a comparative study of the proposed unsupervised scheme with the $\ell_{2,1}$-*norm Regularized Discriminative Feature Selection* algorithm (UDFS) [40] in Section 8. In Section 11, the performance of the proposed supervised method is compared with that of the *Multi-Class SVM-Recursive Feature Elimination* (MSVM-RFE) [41] and the *Redundancy Constrained Feature Selection* (RCFS) [42] algorithms.

## 3. Assessment of the selected features

Several indices can be used to measure the goodness of the selected features. There are broadly two categories of indices: indices those depend on the class labels and indices those do not depend on the labels.

When the class labels are not known, Sammon's Error can be used as a useful index to check the topology preservation of the data set. Several Cluster Preserving Indices like Adjusted Rand Index [18], Normalized Mutual Information [35] can also be used to check the preservation of the cluster structure in the reduced dimension. In this investigation we have used the Cluster Preserving Index proposed in [31]. If the data set is labeled, irrespective of the selection method, supervised or unsupervised we can use a 1-NN (Nearest Neighbor) or any other classifier to evaluate the selected features. However, if the features selected using an unsupervised method cannot do a good job of classification, it does not mean that the feature selection method is a poor one, because class structure may be quite different from cluster structure. If the cluster structure or the neighborhood relation between data points are preserved in the reduced space, then the performance of 1-NN or any other distance based classifier using the selected features should be comparable with that of using all features. Here, in this work, 1-NN classifier with ten-fold cross validation is used.

We have used three performance indices, which are discussed in the following subsections.

### 3.1. Sammon's Error (SE)

Q2 Let, $X \subseteq \mathbb{R}^{n\times p}$ be the data set, i.e. $X$ has $n$ data samples, each of dimension $p$ and $Y$ be the data set in the reduced dimension $r$. If $d^*_{ij}$ is the Euclidean distance between data samples $\mathbf{x_i}$ and $\mathbf{x_j}$ where $\mathbf{x_i}, \mathbf{x_j} \in X$ and $d_{ij}$ is the Euclidean distance between samples $\mathbf{y_i}$ and $\mathbf{y_j}$ where $\mathbf{y_i}, \mathbf{y_j} \in Y$, then the Sammon's Error (SE) [30] can be written as:

$$SE = \frac{1}{\sum_{i<j} d^*_{ij}} \sum_{i<j} \frac{(d^*_{ij} - d_{ij})^2}{d^*_{ij}}.$$

The Sammon's Error measures the extent of preservation of interpoint distances in the reduced dimension. The lower the value, the better is the preservation.

### 3.2. Cluster Preserving Index [31]

In this work, to measure the preservation of cluster structure in the reduced dimension, we have considered Cluster Preserving Index (CPI) discussed in [31]. If the selected features are good in the sense of preserving cluster structure then in both original and reduced dimension the cluster structure would be similar. In order to get the clustering in the original dimension, we apply Fuzzy $C$-Means (FCM) on the original data set with number of clusters equals to $k$. We initialize the fuzzy partition matrix randomly for the original data set. Let the final fuzzy partition matrix be $U$. In the reduced dimension, we initialize the fuzzy partition matrix with $U$ and then apply FCM to get the final partition in the reduced dimension. This is likely to reduce the effect of initialization on the final clustering results. As a result of this, the final partition in the reduced dimension is likely to be similar to that in the original dimension, if the cluster structure remains more or less the same in the reduced dimension. Then we create a confusion matrix $C$, where each $ij$th entry of the matrix denotes the number of data points in the $i$th cluster in the original dimension which are placed in the $j$th cluster in the reduced dimension. Then the clusters are re-labeled and the confusion matrix is realigned using a realignment algorithm described in Fig. 2. The sum ($T$) of the off diagonal entries of the realigned confusion matrix is the number of points misplaced between the two partitions. Let the number of data points be $n$. We compute the percentage of points misplaced between the two partitions as $M = 100\ \frac{T}{n}$. The CPI is defined as $(100 - M)$. The algorithm for computing the Cluster Preserving Index is given in Fig. 1. Since all our data sets are labeled, we use $k$ = number of classes. Note that, for a given data set this may not necessarily be the best choice for the number of clusters. But that does not cause a problem as we are trying to compare two partitions, one using all features, another using a set of selected features, and in both cases we use the same number of clusters.

### 3.3. Misclassification error

When the class labels for a data set are known, the goodness of the selected feature set can also be measured by calculating the misclassification error (MCE). As mentioned earlier, here the 1-NN classifier with ten-fold cross validation is used to compute MCE.

Q1 Please cite this article in press as: M. Banerjee, N.R. Pal, Feature selection with SVD entropy: Some modification and extension, Inform. Sci. (2014), http://dx.doi.org/10.1016/j.ins.2013.12.029

**Input:** **The Original data set $X$, The Reduced data set $Y$, number of clusters $k$.**
**Output:** **Cluster Preserving Index (CPI).**

Step 1: Apply FCM (Fuzzy C-Means) on the data set $X$ with number of clusters $k$. Let the final fuzzy partition matrix be $U$.
Step 2: Take the reduced data set $Y$. Apply FCM on $Y$ with initialization of the fuzzy partition matrix with $U$ in order to reduce the effect of initialization.
Step 3: Compute the confusion matrix $C = [c_{ij}]_{k\times k}$ where $c_{ij}$ denotes the number of points in the $i^{th}$ cluster in the original data set that are placed in the $j^{th}$ cluster in the reduced data set.
Step 4: Use *Realignment* algorithm (described in Figure 2) to relabel the centers and realign the confusion matrix.
Step 5: Let $T$ be the sum of the off diagonal elements of the realigned matrix. Compute CPI = $1 - 100\,\frac{T}{n}$, where $n$ = Number of data points in $|X|$.

**Fig. 1.** Cluster Preserving Index algorithm.

**1) Construct the Confusion Matrix $C$ of dimension $k \times k$:**
for $i = 1 : n$
  $l$ = cluster label of $\mathbf{x_i} \in X$.
  $m$ = cluster label of $\mathbf{y_i} \in Y$.
  $c_{lm} = c_{lm} + 1$.
end

**2) Normalize the Confusion matrix $C$ to generate matrix $\bar{C}$**
for $i = 1 : k$
  for $j = 1 : k$
    $\bar{c}_{ij} = \frac{c_{ij}}{\sum_{l=1}^{k} c_{il}}$
  end
end

**3) The realigned matrix $R$ is computed from $C$ using $\bar{C}$**
for $i = 1 : k$
  for the $i^{th}$ row of $\bar{C}$ select the column, *maxcol*, with the maximum value.
  Copy the $maxcol^{th}$ column of $C$ into the $i^{th}$ column of $R_{k\times k}$, i.e.,
  for $j = 1 : k$
    $r_{ji} = c_{j,maxcol}$
  end
  Replace all entries in the $maxcol^{th}$ column of $\bar{C}$ by $-1$ i.e.,
  for $j = 1 : k$
    $\bar{c}_{j,maxcol} = -1$
  end
end

**Fig. 2.** Realignment algorithm.

**1-NN classifier:** We are given a training data set $X_{TR}$ and test data set $X_{TS}$. A data sample $\mathbf{v} \in X_{TS}$ is assigned to the same class as of its closest neighbor $\mathbf{u} \in X_{TR}$, i.e. $d_{\mathbf{u},\mathbf{v}} \leqslant d_{\bar{\mathbf{u}},\mathbf{v}}, \forall \bar{\mathbf{u}} \in X_{TR}$. $d_{\mathbf{u},\mathbf{v}}$ is a dissimilarity measure between $\mathbf{u}$ and $\mathbf{v}$, which is taken as the Euclidean distance.

**Ten-fold cross validation:** The data set $X$ is randomly partitioned into ten nearly equal subsamples, $S_1, S_2, \ldots, S_{10}$. Where,

- $\cup_{i=1}^{10} S_i = X$.
- $S_i \neq \emptyset,\ i = 1, \ldots, 10$.
- $S_i \cap S_j = \emptyset,\ i, j = 1, \ldots, 10\ i \neq j$.

Union of nine subsets are used for training and the remaining one for testing. In this way each of the 10 subsets is used for testing. The entire process is repeated ten times and the average misclassification over these ten iterations is reported.

## 4. Singular values and singular vectors

We shall be encountering the term *Singular Values* frequently in this paper. Hence, for the sake of completeness we briefly discuss what are singular values and singular vectors of a matrix and how are they computed [16].

Q1 Please cite this article in press as: M. Banerjee, N.R. Pal, Feature selection with SVD entropy: Some modification and extension, Inform. Sci. (2014), http://dx.doi.org/10.1016/j.ins.2013.12.029

Consider a matrix $X \subseteq \mathbb{R}^{n\times p}$ with $n$ vectors in $p$-dimension. We can represent the $i$th row vector as $\mathbf{x}_i$, where $i = 1,\ldots,n$. Intuitively we shall define the direction of the first singular vector (the first principle direction) to be that direction in which the vectors $\mathbf{x}_1,\ldots,\mathbf{x}_n$ point more than in any other direction. Take an unit vector $\mathbf{u}$ along the direction, in which we want to measure alignment of the data vectors. The scalar product of the vector $\mathbf{x}_i$ with $\mathbf{u}$, say $(\mathbf{u},\mathbf{x}_i)$, denotes how much $\mathbf{x}_i$ points in the direction of $\mathbf{u}$. Let $\mathbf{z}$ be the $n$ dimensional vector, defined by $((\mathbf{u},\mathbf{x}_1),\ldots,(\mathbf{u},\mathbf{x}_n))$. Then the length of $\mathbf{z}$ measures the extent these $n$ vectors $\mathbf{x}_1,\ldots,\mathbf{x}_n$ point in the direction of $\mathbf{u}$. Now find the unit vector $\mathbf{u}_1$, for which the vector $\mathbf{z}_1 = X\mathbf{u}_1$ will have the maximum length. Then, the length $s_1 = \|\mathbf{z}_1\|$ is the first singular value and $\mathbf{u}_1$ is called the corresponding singular direction.

Suppose we have already chosen $i-1$ singular directions $\mathbf{u}_1,\ldots,\mathbf{u}_{i-1}$, where $i \leqslant \text{rank}(X)$. Then we select an unit vector $\mathbf{u}_i$, orthogonal to $\mathbf{u}_1,\ldots,\mathbf{u}_{i-1}$, in such a way that the length of $\mathbf{z}_i = X\mathbf{u}_i$ is the maximum. If this length $s_i = \|\mathbf{z}_i\|$ is not null, we call $s_i$ the $i$th singular value and $\mathbf{u}_i$ the respective singular direction. A non-null vector proportional to $\mathbf{u}_i$ will be called a singular vector corresponding to $s_i$. If $m = \text{rank}(X)$, then there will be $m$ non-zero singular values, $s_1,\ldots,s_m$. Moreover, for any vector $\mathbf{u}$, $X\mathbf{u} = \mathbf{0}$, if $\mathbf{u}$ is orthogonal to $\mathbf{u}_1,\ldots,\mathbf{u}_m$.

Now, $\|\mathbf{z}\|^2 = \|X\mathbf{u}\|^2 = (X\mathbf{u},X\mathbf{u}) = (X^TX\mathbf{u},\mathbf{u})$. This means that $s_1^2, s_2^2,\ldots$, are the eigenvalues of $XX^T$ and $\mathbf{u}_1,\mathbf{u}_2,\ldots$, are the corresponding eigenvectors. As, $\mathbf{u}_i$ and $\mathbf{u}_j$ are orthogonal $\forall j \neq i$, we have $(\mathbf{u}_i,\mathbf{u}_j) = 0$. Consequently $(X\mathbf{u}_i, X\mathbf{u}_j) = (\mathbf{z}_i,\mathbf{z}_j) = 0$, which suggests that the vectors $\mathbf{z}_1,\mathbf{z}_2,\ldots$, are also mutually orthogonal.

Let, $U$ and $Z$ be the matrices whose column vectors are $\mathbf{u}_1,\ldots,\mathbf{u}_m$ and $\mathbf{z}_1,\ldots,\mathbf{z}_m$ respectively. Then,

$$XU = Z. \tag{2}$$

As, $s_1,\ldots,s_m$ are lengths of the vectors $\mathbf{z}_1,\ldots,\mathbf{z}_m$, we can find a set of unit vectors $\mathbf{v}_1,\ldots,\mathbf{v}_m$, such that $\mathbf{z}_i = s_i\mathbf{v}_i,\ \forall i = 1,\ldots,m$. So, we may write $Z = V\Lambda$, where $V$ is the matrix with column vectors $\mathbf{v}_1,\ldots,\mathbf{v}_m$ and $\Lambda$ is a diagonal matrix having $s_1,\ldots,s_m$ as its diagonal elements. Then using Eq. (2) we have,

$$XU = V\Lambda. \tag{3}$$

Suppose $p = n = m$. Then $U$ and $V$ are orthogonal matrices and $U^T$ and $V^T$ are their inverses. Hence from Eq. (3), the matrix $X$ can be expressed in the form $X = V\Lambda U^T$. This result is valid even if $p, n$ and $m$ are not equal [16]. The vectors $\mathbf{v}_1,\mathbf{v}_2,\ldots$, are the singular vectors of $X^T$.

## 5. An SVD-entropy based feature selection technique [39]

Let us first define some notations, which will be used in this work. $X$ denotes a data matrix of size $n \times p$. If clarity demands, we shall use $X_{n\times p}$ to indicate the size of the data matrix $X$. $X^{-i}$ denotes a data matrix of size $n \times (p-1)$, which is generated by dropping the $i$th column of $X$. Similarly, $X^{-i,-j}$ denotes a data matrix of size $n \times (p-2)$, which is generated by dropping the $i$th and $j$th columns of $X$. On the other hand, $X^i$ is a data matrix of size $n \times 1$ that is obtained by considering only the $i$th column of $X$ and $X^{i,j}$ is a data matrix of size $n \times 2$, obtained by taking projection on the $i$th and $j$th dimensions.

Alter et al. [1] defined the SVD-entropy (Singular Value Decomposition based entropy) of a data set $X \subseteq \mathbb{R}^{n\times p}$ as the Shannon's entropy of normalized eigenvalues of the $n \times n$ matrix $XX^T$.

Let the singular values of the matrix $X$ be denoted by $s_j$. $s_j^2$s are then the eigenvalues of the $n \times n$ matrix $XX^T$. Let us define the normalized eigenvalues as,

$$V_j = s_j^2 \Big/ \sum_k s_k^2. \tag{4}$$

In [1], the SVD-entropy of the data set $X$ was defined as,

$$E(X) = -\frac{1}{log(N)} \sum_{j=1}^{N} V_j\ log(V_j). \tag{5}$$

Note that, $N$ is the rank of the matrix $X$. Varshavsky et al. [39] used this SVD based data set entropy to estimate the relevance of each feature. They defined the contribution of the $i$th feature, $i = 1,\cdots,p$, to the entropy, i.e. $CE_i$, by a leave-one-out method as follows:

$$CE_i = E(X) - E(X^{-i}). \tag{6}$$

Let the average and the standard deviation of all $CE_i$ values be $c$ and $s$ respectively. Then features are partitioned into three groups:

- $CE_i > c + s$: features with high contribution.
- $c - s < CE_i < c + s$: features with average contribution.
- $CE_i < c - s$: features with low (usually negative) contribution.



Q1 Please cite this article in press as: M. Banerjee, N.R. Pal, Feature selection with SVD entropy: Some modification and extension, Inform. Sci. (2014), http://dx.doi.org/10.1016/j.ins.2013.12.029

In [39], only the features in the first group (i.e. the features with high contribution) are considered relevant. And the number of features in this group is taken as the *optimum* number of features to be selected. Let this optimum number of features be $r0$.

The selection of features based on *CE* values can be done in three different ways.

- **Simple Ranking (SR) method:** Select $r0$ features according to the highest rank in terms of the CE values.
- **Forward Selection (FS) method:** Two implementations are considered.
  - **FS1:** Choose the first feature according to the highest CE value. Choose among the remaining features the one which, together with the first feature, produces a 2-feature set with the highest entropy. Suppose, we have already selected t features. Continue this process over the remaining $p - t$ features to choose the $(t + 1)$th feature according to the maximum entropy. The process stops when $r0$ features are selected.
  - **FS2:** Choose the first feature as before. Recalculate the CE values of the remaining set of size $p - 1$ and select the second feature according to the highest CE value. Continue the same way until $r0$ features are selected.
- **Backward Elimination (BE) method:** Eliminate the feature with the lowest CE value. Recalculate the CE values and iteratively eliminate the lowest one until $r0$ features remain.

According to Varshavsky et al. [39], SR is the fastest one and BE is the most cumbersome one for large number of features. FS1 chooses features which exhibit high original CE-scores; hence the selected set will have a large intersection with that by SR. It has also been claimed that, though SR can include redundant features, i.e. features with high correlation, such redundancy is reduced by FS1.

### 5.1. A modified unsupervised feature filtering technique

In this section, based on the definition of SVD-entropy of a data set, as given in [1], we have modified the definition of $CE_i$, i.e. the contribution of the $i$th feature to the entropy, in Eq. (6). Before stating the modification, we will comment in brief on the use of Singular Value Decomposition in dimensionality reduction.

A high singular value of a data set denotes that there is a high variance in the data along a certain direction. If a normalized eigenvalue, $V_j$ is large, then by considering only the component of the data set along the direction of the corresponding singular vector, the underlying structure of the data set will be well preserved. The alignment of the data set in a particular direction is a resultant effect of all features. Some of the features have more influence on this alignment than the others. If we leave out a feature with a high influence, the resultant variance in that direction, i.e. the singular value will reduce significantly. For high singular values, if we can identify features with greater influence, then by selecting only those features, the underlying characteristics of the data set can be maintained to a great extent.

As discussed in Section 5, SVD-entropy of a data set can be obtained as the Shannon's entropy of the normalized eigenvalues, following Eq. (5). This entropy varies between 0 and 1. $E = 0$ corresponds to an ordered and redundant data set that can be explained by a single eigenvector (problem of rank 1), and $E = 1$ represents a disordered and random data set in which all eigenvalues (and eigenvectors) are equally expressed.

We want to identify those features, which help to make the data set an ordered one, and in absence of which the data set become disordered. Hence, we modify the definition of $CE_i$ with $mCE_i$ as,

$$mCE_i = E(X^{-i}) - E(X). \tag{7}$$

Now, if the $i$th feature can represent the underlying structure of the data set well, then it must have a large influence on high singular values. In absence of this $i$th feature, these singular values will reduce significantly, making the distribution of the normalized eigenvalues more uniform, as the eigenvalues are now more equally expressed. Hence, $E(X)$ is expected to be much less than $E(X^{-i})$, where $X^{-i}$ is the data matrix of size $n \times (p - 1)$, which is obtained by removing the $i$th feature of $X$. Hence the $i$th feature is considered to be better than the $j$th feature, if $mCE_i > mCE_j$, following Eq. (7).

Now, let $c$ and $s$ represent the mean and standard deviation of all $mCE_i$ values respectively. Then, the optimum number of features ($r1$) based on the modified $CE_i$ score, $mCE_i$, is the number of features with $mCE_i > c + s$. The four modified variants of this unsupervised algorithm, i.e. mSR, mFS1, mFS2 and mBE can then be obtained in a similar way as SR, FS1, FS2 and BE respectively. In other words, in the methods SR, FS1, FS2 and BE (as described in Section 5), if we replace $CE_i$ by $mCE_i$, then we get methods mSR, mFS1, mFS2 and mBE, respectively. Like FS1, mFS1 can also reduce redundancy.

## 6. Computational complexity analysis

Note that, we have modified the *CE* definition only, so the complexities of the four new methods remain the same as those of respective old methods. For clarity, the detailed complexities of these four proposed algorithms are given.

The computation of all singular values for a dense matrix $A$ requires $\mathcal{O}(\varrho_A^3)$ operations, where $\varrho_A$ is the rank of $A$ [2]. For the given data matrix $X$ of size $n \times p$, the first step of all four methods is the same. In this step we compute $p$ *mCE* values, which requires computation of order $p\mathcal{O}\left(\varrho_{X_{n\times p}}^3\right)$.

Q1 Please cite this article in press as: M. Banerjee, N.R. Pal, Feature selection with SVD entropy: Some modification and extension, Inform. Sci. (2014), http://dx.doi.org/10.1016/j.ins.2013.12.029

**mSR**: After the first step, mSR needs to find out top $r1$ features with largest *mCE* values. With an efficient sorting algorithm, it requires altogether $\mathcal{O}(\min(pr1, p\log p)) + p\mathcal{O}\left(\varrho^3_{X_{n\times p}}\right)$ operations.

**mFS1**: This algorithm choses the first feature with the highest *mCE* value, which requires $p\mathcal{O}\left(\varrho^3_{X_{n\times p}}\right) + (p-1)$ computations. After the first step, in any $t$th step we want to find out the $t$th important feature. This needs to compute SVD-entropy of $(p-t+1)$ data sets of size $n \times t$. Thus the $t$th step requires the following amount of operations:

$$\left\{ \overbrace{(p-t+1)\ \mathcal{O}\left(\varrho^3_{X_{n\times t}}\right)}^{\text{Eigen value computation}} + \underbrace{(p-t)}_{\text{Comparisons}} \right\}, \quad \text{where } t = 2, \cdots, r1.$$

So, total computations for mFS1 to select $r1$ features is,

$$p\ \mathcal{O}\left(\varrho^3_{X_{n\times p}}\right) + \sum_{t=2}^{r1}\left\{(p-t+1)\ \mathcal{O}\left(\varrho^3_{X_{n\times t}}\right)\right\} + \left\{p\times r1 - \frac{r1\ (r1+1)}{2}\right\}.$$

**mFS2**: In the $t$th step, this method computes all *mCE* values of a $(p-t+1)$ dimensional data matrix, $X_{n\times(p-t+1)}$ and selects the $t$th important feature as the feature with the highest *mCE* value. The computational overhead for the $t$th step is,

$$\left\{ \overbrace{(p-t+1)\ \mathcal{O}\left(\varrho^3_{X_{n\times(p-t+1)}}\right)}^{\text{mCE value computation}} + \underbrace{(p-t)}_{\text{Comparisons}} \right\}, \quad \text{where } t = 1, \cdots, r1.$$

So, the total computational cost to select $r1$ features is,

$$\sum_{t=1}^{r1}\left\{ \overbrace{(p-t+1)\ \mathcal{O}\left(\varrho^3_{X_{n\times(p-t+1)}}\right)}^{\text{mCE value computation}} + \underbrace{(p-t)}_{\text{Comparisons}} \right\}.$$

**mBE**: This method eliminates $(p-r1)$ less important features to retain $r1$ features. In the $t$th step, among the $(p-t+1)$ remaining features, the feature with the lowest *mCE* value is discarded. Hence, the amount of computation required in the $t$th step of *mFS*2 and *mBE* is the same. But number of iterations needed in *mBE* to select $r1$ features is $(p-r1)$. So, the computational complexity of *mBE* is,

$$\sum_{t=1}^{p-r1}\left\{ \overbrace{(p-t+1)\ \mathcal{O}\left(\varrho^3_{X_{n\times(p-t+1)}}\right)}^{\text{mCE value computation}} + \underbrace{(p-t)}_{\text{Comparisons}} \right\}.$$

mSR is the fastest method and mFS2 or mBE are computationally the most expensive ones. If $p > 2r1$, then mBE is the most expensive method.

## 7. Experimental results

We have evaluated our methods on 14 data sets. The specification of each data set is given in Table 1. There are 12 real data sets, an augmented version of Iris data and one Synthetic data.

Detailed description of the real data sets can be found in [43,44]. The WBC data set has 16 instances with missing values, hence these instances are deleted. Feature 2 of the Ionosphere data has all zero values, so it is an indifferent feature. According to [43], number of instances in 4 classes of the vehicle data are 240, 240, 240 and 226. This description does not match the actual data set available at [43]. Colon and Leukemia data sets are preprocessed following [11].

Our method evaluates features based on their contribution to entropy (*mCE*) values. This measure heavily depends on how the eigenvalues are distributed. Eigenvalues are sensitive to the relative scaling of the original variables. So, we use the $z$ score normalization of the data to re-scale each coordinate to have unit variance, which ensures that every attribute is treated on the same scale.

The Synthetic data set has 300 points, each of dimensions 7. These points are distributed in 3 spherical clusters. The clusters are formed in 3$D$ and then 4 additional features are added to each point as follows: The centers of the three clusters in 3$D$ are $(0,0,0)$, $(0,15,0)$ and $(0,0,15)$. Note that, feature 1 does not add any discriminating power – it is a redundant feature. Now we add two features 4th and 5th, where 4th is highly correlated with 2nd, while 5th is highly correlated with 3rd. Two random features, with values ranging from $-5$ to 5, are also added as 6th and 7th features. Hence, (2nd or 4th) and (3rd or 5th) features are the important ones. The 1st feature is redundant and 6th and 7th features are derogatory.

Please cite this article in press as: M. Banerjee, N.R. Pal, Feature selection with SVD entropy: Some modification and extension, Inform. Sci. (2014), http://dx.doi.org/10.1016/j.ins.2013.12.029

**Table 1**
Description of data sets used.

| Serial number | Name of the data set | No. of features | No. of classes | Data set size (with distribution) |
|---|---|---|---|---|
| 1 | Synthetic | 7 | 3 | 300(103 + 85 + 112) |
| 2 | Iris | 3 | 4 | 150(50 + 50 + 50) |
| 2_1 | (Augmented) Iris | 3 | 6 | 150(50 + 50 + 50) |
| 3 | WBC | 2 | 9 | 699(458 + 241) |
| 4 | Glass | 6 | 9 | 214(70 + 17 + 76 + 13 + 9 + 29) |
| 5 | Wine | 3 | 13 | 178(59 + 72 + 47) |
| 6 | Ionosphere | 2 | 34 | 351(225 + 126) |
| 7 | Sonar | 2 | 60 | 208(97 + 111) |
| 8 | Vehicle | 4 | 18 | 846(212 + 217 + 218 + 199) |
| 9 | Liver | 2 | 6 | 345(145 + 200) |
| 10 | Thyroid | 3 | 5 | 215(150 + 35 + 30) |
| 11 | SPECT Heart | 2 | 22 | 267(55 + 212) |
| 12 | Colon | 2 | 2000 | 62(40 + 22) |
| 13 | Leukemia | 2 | 7129 | 72(47 + 25) |

We have added two additional features (5th and 6th) with the Iris data set to form the (Augmented) Iris data. The 5th and 6th features are highly correlated to the 3rd and 4th features respectively. The 5th and 6th features have higher variance than the 3rd and 4th features.

As mentioned in Section 5.1, the optimum number of features ($r1$), i.e. the dimension of the reduced data set is taken as the number of features with *mCE* values greater than $c + s$, where $c$ and $s$ denote the mean and standard deviation of all *mCE* values respectively. *We note here that, for Iris, (Augmented) Iris and Synthetic data sets the number of features with mCE values higher than $c + s$ are 0.* Even with the original *CE* definition (Eq. (6)), the optimum number ($r0$) for the Thyroid data set is 0. This happens because these data sets have features with *mCE* (or *CE*) values much lower than the mean ($c$), hence creating a large standard deviation ($s$). But there is no feature, whose *mCE* (or *CE*) value is that much higher than $c$. Thus the optimum number of features, following the principle adopted in [39] are 0 for these data sets. Since this principle does not work even for simple well structured data sets, it may not be a good principle to adapt. However, even when there is no feature with *mCE* value $> c + s$, we can rank features based on them and that is what we have done.

In Table 2, we summarize the results for the Synthetic data using the four proposed unsupervised methods, mSR, mFS1, mFS2 and mBE. In Table 2 (as well as in other tables), for any given method the column named "Feature No." lists the indices of the features in order of the ranks given in column 1 of the Table. The $i$th row of the result for a method shows values of the three indices as obtained by the dataset in the reduced dimension with the first $i$ selected features. As an example, the result of the algorithm mSR is discussed in detail. From the second column we can see that mSR has selected feature 3 as the first feature, feature 2 as the second feature and so on. The 1st row of this table denotes SE, CPI and MCE values for the reduced dataset with only the first selected feature, i.e. feature 3 for mSR. Similarly, the 4th row represents these indices using the first 4 selected features, i.e. features 3, 2, 5 and 4, listed in rows 1–4.

From the results on the Synthetic data in Table 2, it can be seen that except mFS1 other three methods are able to identify the important features. The poor performance of mFS1 is not because it can control linear dependency. If we consider only the 2nd and 3rd features, the Synthetic data will form three circular plates. So, it is evident that these two most important

**Table 2**
Results of the Synthetic data set with the mCE definition.

| Ranks | mSR | | | | mFS1 | | | |
|---|---|---|---|---|---|---|---|---|
| | Feature no. | SE | CPI | MCE | Feature no. | SE | CPI | MCE |
| 1 | **3** | 0.35 | 71.00 | 30.27 | **3** | 0.35 | 71.00 | 30.27 |
| 2 | **2** | 0.14 | 100.00 | 0.00 | 6 | 0.28 | 70.67 | 28.47 |
| 3 | **5** | 0.06 | 100.00 | 0.00 | 7 | 0.23 | 70.33 | 30.53 |
| 4 | **4** | 0.03 | 100.00 | 0.00 | 1 | 0.20 | 70.00 | 33.67 |
| 5 | 1 | 0.01 | 100.00 | 0.00 | **4** | 0.06 | 100.00 | 0.00 |
| 6 | 6 | 0.00 | 100.00 | 0.00 | **5** | 0.02 | 100.00 | 0.00 |
| 7 | 7 | 0.00 | 100.00 | 0.00 | **2** | 0.00 | 100.00 | 0.00 |
| | mFS2 | | | | mBE | | | |
| 1 | **3** | 0.35 | 71.00 | 30.27 | **3** | 0.35 | 71.00 | 30.27 |
| 2 | **2** | 0.14 | 100.00 | 0.00 | **5** | 0.23 | 71.00 | 27.63 |
| 3 | **5** | 0.06 | 100.00 | 0.00 | **2** | 0.06 | 100.00 | 0.00 |
| 4 | 1 | 0.04 | 100.00 | 0.00 | **4** | 0.03 | 100.00 | 0.00 |
| 5 | 6 | 0.03 | 100.00 | 0.00 | 1 | 0.01 | 100.00 | 0.00 |
| 6 | **4** | 0.00 | 100.00 | 0.00 | 6 | 0.00 | 100.00 | 0.00 |
| 7 | 7 | 0.00 | 100.00 | 0.00 | 7 | 0.00 | 100.00 | 0.00 |



Q1 Please cite this article in press as: M. Banerjee, N.R. Pal, Feature selection with SVD entropy: Some modification and extension, Inform. Sci. (2014), http://dx.doi.org/10.1016/j.ins.2013.12.029

**Table 3**
Results of the Iris data set with the mCE definition.

| Ranks | mSR | | | | mFS1 | | | |
|---|---|---|---|---|---|---|---|---|
| | Feature no. | SE | CPI | MCE | Feature no. | SE | CPI | MCE |
| 1 | 3 | 0.08 | 94.67 | 6.40 | 3 | 0.08 | 94.67 | 6.40 |
| 2 | 4 | 0.04 | 93.33 | 3.67 | 2 | 0.04 | 95.33 | 8.27 |
| 3 | 1 | 0.01 | 98.67 | 4.27 | 1 | 0.01 | 98.00 | 6.47 |
| 4 | 2 | 0.00 | 100.00 | 4.01 | 4 | 0.00 | 100.00 | 4.01 |
| | mFS2 | | | | mBE | | | |
| 1 | 3 | 0.08 | 94.67 | 6.40 | 4 | 0.45 | 89.33 | 5.67 |
| 2 | 4 | 0.04 | 93.33 | 3.67 | 3 | 0.04 | 93.33 | 3.67 |
| 3 | 1 | 0.01 | 98.67 | 4.27 | 1 | 0.01 | 98.67 | 4.27 |
| 4 | 2 | 0.00 | 100.00 | 4.01 | 2 | 0.00 | 100.00 | 4.01 |

**Table 4**
Results of the (Augmented) Iris data set with the mCE definition.

| Ranks | mSR | | | | mFS1 | | | |
|---|---|---|---|---|---|---|---|---|
| | Feature no. | SE | CPI | MCE | Feature no. | SE | CPI | MCE |
| 1 | 3 | 0.68 | 96.67 | 7.07 | 3 | 0.68 | 96.67 | 7.07 |
| 2 | 5 | 0.02 | 97.33 | 7.93 | 2 | 0.66 | 96.00 | 7.07 |
| 3 | 4 | 0.02 | 97.33 | 3.87 | 1 | 0.63 | 92.00 | 6.60 |
| 4 | 6 | 0.00 | 100.00 | 3.13 | 6 | 0.31 | 96.67 | 3.27 |
| 5 | 1 | 0.00 | 100.00 | 3.27 | 4 | 0.31 | 96.67 | 3.73 |
| 6 | 2 | 0.00 | 100.00 | 3.73 | 5 | 0.00 | 100.00 | 3.73 |
| | mFS2 | | | | mBE | | | |
| 1 | 3 | 0.68 | 96.67 | 7.07 | 5 | 0.02 | 97.33 | 6.80 |
| 2 | 5 | 0.02 | 97.33 | 7.93 | 3 | 0.02 | 97.33 | 7.93 |
| 3 | 4 | 0.02 | 97.33 | 3.87 | 4 | 0.02 | 97.33 | 3.87 |
| 4 | 6 | 0.00 | 100.00 | 3.13 | 6 | 0.00 | 100.00 | 3.13 |
| 5 | 1 | 0.00 | 100.00 | 3.27 | 1 | 0.00 | 100.00 | 3.27 |
| 6 | 2 | 0.00 | 100.00 | 3.73 | 2 | 0.00 | 100.00 | 3.73 |

features are nonlinearly dependent to each other. Thus, we see that mFS1 has excluded a feature with high nonlinear dependency with the selected feature, but that has not helped in this case.

The results on Iris data are shown in Table 3. For Iris data set, the 3rd and 4th features are the two most important features in terms of discriminating power. From Table 3 we can see that all the four unsupervised approaches have performed well. As claimed in Section 5.1, mFS1 has indeed excluded highly correlated features as redundant. This is evident from the order in which features are selected by mFS1. mSR, mFS2 and mBE have selected 3rd and 4th features as the most important two, whereas mFS1 has ranked features 2 and 4 as the second and fourth features respectively. This is because correlation between the 3rd and 2nd features is the minimum (−0.42), and it is the maximum between the 3rd and 4th features (0.96).

The result for the (Augmented) Iris data is depicted in Table 4. From this result, we see that all methods, mSR, mFS1, mFS2 and mBE have performed well. But unlike the other three, mFS1 after selecting the 3rd feature, has selected the highly correlated features 5, 4 and 6 with lower priorities than the less correlated features 2 and 1. For all methods but mFS1, the top 2–3 features result in very good values for SE, CPI and MCE.

The Ionosphere data set has 351 instances in 34 dimensions. Among these 34 features, the 2nd feature has all 0 values. The optimum number of features ($r1$) for the Ionosphere data is 8. For this data set, in Table 5 we show results with selected features up to the optimum number of features. In the Ionosphere data set the 2nd feature is redundant, still it has been chosen by mSR, mFS2 and mBE as one of the important features. Apparently after choosing the 2nd feature none of the three indices has undergone any changes. The reason for selecting the 2nd feature can be explained as follows.

As the 2nd feature contains all zero values, among 34 singular values of the Ionosphere data, one singular value must be equal to zero. The 2nd feature has no influence on any of the singular values. So, the 33 singular values, obtained after removing the 2nd feature, are the same as the previous non-zero singular values of the original data set. Hence, the value of $-\sum_{j=1}^{N} V_j \, log(V_j)$ in Eq. (5) is the same for both the original data set (say, $X$) and the data set obtained after removing the 2nd feature (say, $X^{-2}$). But the values of $N$, for $X$ and $X^{-2}$ are 34 and 33 respectively. As $-\sum_{j=1}^{N} V_j \, log(V_j)$ is always a non-negative quantity, from Eq. (5), we have $E(X^{-2}) > E(X)$, where $X^{-2}$ is the 33 dimensional data set, generated after removing the 2nd feature of the Ionosphere data. Because of this, though the 2nd feature is redundant, it will have a small but positive *mCE* value, as per Eq. (7). Now depending on the *mCE* values of other features the 2nd feature has been selected by mSR, mFS2 and mBE. Even mFS1 has chosen it as the 29th feature, when it should have been the feature with the least priority. This clearly



Q1 Please cite this article in press as: M. Banerjee, N.R. Pal, Feature selection with SVD entropy: Some modification and extension, Inform. Sci. (2014), http://dx.doi.org/10.1016/j.ins.2013.12.029

**Table 5**
Results of the ionosphere data set with the mCE definition.

| Ranks | mSR | | | | mFS1 | | | |
|---|---|---|---|---|---|---|---|---|
| | Feature no. | SE | CPI | MCE | Feature no. | SE | CPI | MCE |
| 1 | 15 | 0.69 | 92.59 | 24.25 | 15 | 0.69 | 92.59 | 24.25 |
| 2 | 21 | 0.56 | 93.73 | 20.17 | 32 | 0.56 | 92.59 | 20.63 |
| 3 | 17 | 0.47 | 92.88 | 16.44 | 1 | 0.53 | 92.31 | 20.83 |
| 4 | 13 | 0.41 | 93.45 | 15.81 | 20 | 0.46 | 92.31 | 17.44 |
| 5 | 19 | 0.36 | 94.87 | 17.86 | 12 | 0.41 | 91.17 | 18.03 |
| 6 | 23 | 0.31 | 96.58 | 15.70 | 3 | 0.37 | 92.59 | 16.70 |
| **7** | **2** | **0.31** | **96.58** | **15.70** | 24 | 0.33 | 91.74 | 12.22 |
| 8 | 11 | 0.28 | 97.44 | 17.04 | 4 | 0.30 | 92.59 | 10.40 |
| | All Features | 0.00 | 100.00 | 13.22 | All Features | 0.00 | 100.00 | 13.22 |
| | mFS2 | | | | mBE | | | |
| 1 | 15 | 0.69 | 92.59 | 24.25 | 15 | 0.69 | 92.59 | 24.25 |
| **2** | 21 | 0.56 | 93.73 | 20.17 | **2** | **0.69** | **92.59** | **24.25** |
| 3 | 17 | 0.47 | 92.88 | 16.44 | 13 | 0.57 | 91.17 | 22.65 |
| 4 | 19 | 0.41 | 92.59 | 16.07 | 21 | 0.48 | 92.88 | 19.40 |
| 5 | 23 | 0.35 | 96.58 | 15.90 | 17 | 0.41 | 93.45 | 15.93 |
| **6** | **2** | **0.35** | **96.58** | **15.90** | 11 | 0.37 | 94.59 | 17.78 |
| 7 | 13 | 0.31 | 96.58 | 15.58 | 19 | 0.32 | 94.87 | 17.72 |
| 8 | 33 | 0.27 | 98.29 | 17.52 | 9 | 0.29 | 95.16 | 15.47 |
| | All Features | 0.00 | 100.00 | 13.22 | All Features | 0.00 | 100.00 | 13.22 |

indicates a very weak point of this algorithm. *Note that this analysis is equally applicable to the original method with CE. It may also select indifferent features.*

Now, let us replace this second feature of the Ionosphere data with random values following $N(0, 0.01)$ distribution. The findings using the modified methods on this new version of the Ionosphere data are stated next (data not shown). The mSR and mBE ranked the 2nd feature as the least important one and mFS2 selects it as the 4th last feature. But surprisingly, mFS1 has selected it as the 2nd important one! We looked at the correlation values and found that the 2nd feature has the least correlation with the 15th feature, which is the most important feature selected by all the 4 methods. Hence, this behavior of mFS1 though surprising, may be justified by the inherent redundancy removal property of mFS1.

We have discussed the results of the proposed unsupervised framework on a few data sets. Besides its ability to select the relevant features, some limitations of this framework have also been mentioned. The effectiveness of this proposed scheme can be better understood from the next section, where in addition to demonstrating the improvement achieved by the *mCE* definition over the *CE* definition [39], we have also made a comparison with the UDFS algorithm [40].

## 8. Comparison

As mentioned earlier, the optimum number of features ($r1$) is 0 for Iris, (Augmented) Iris, and Synthetic data sets. In order to do the comparison, we take the value of $r1$ for Iris, (Augmented) Iris, and Synthetic data as 1, 2, and 3 respectively, in an ad

**Table 6**
Comparative results of SR and mSR.

| Data sets | Opt. no. of features ($r1$) | SR | | | mSR | | |
|---|---|---|---|---|---|---|---|
| | | SE | CPI | MCE | SE | CPI | MCE |
| Synthetic | 0 (3) | 0.50 | 41.67 | 65.90 | **0.06** | **100.00** | **0.00** |
| Iris | 0 (1) | 0.69 | 46.27 | 55.33 | **0.08** | **94.67** | **6.40** |
| (Augmented) Iris | 0 (2) | 0.81 | 84.67 | 28.80 | **0.02** | **97.33** | **7.93** |
| WBC | 2 | 0.44 | 86.09 | 22.01 | **0.35** | **92.39** | **6.63** |
| Glass | 2 | 0.73 | 31.00 | 61.07 | **0.37** | **50.81** | **56.45** |
| Wine | 3 | **0.92** | 44.96 | 41.40 | 0.99 | **55.62** | **22.70** |
| Ionosphere | 8 | 0.36 | 64.10 | **14.84** | **0.28** | **97.44** | 17.04 |
| Sonar | 8 | 0.97 | 49.17 | 41.20 | **0.32** | **85.58** | **24.47** |
| Vehicle | 3 | 0.88 | 29.73 | 62.66 | **0.03** | **98.70** | **44.57** |
| Liver | 2 | 0.45 | 44.93 | 45.54 | **0.43** | **89.86** | **41.33** |
| Thyroid | 1 | **0.59** | 35.38 | 20.51 | **0.59** | **40.76** | **11.77** |
| SPECT Heart | 4 | 0.45 | 63.30 | 31.65 | **0.40** | **71.52** | **27.45** |
| Colon | 179 | **0.33** | 88.71 | **26.13** | 0.44 | **93.55** | 27.42 |
| Leukemia | 547 | **0.35** | 84.95 | 29.41 | 0.42 | **91.16** | **5.88** |

Q1 Please cite this article in press as: M. Banerjee, N.R. Pal, Feature selection with SVD entropy: Some modification and extension, Inform. Sci. (2014), http://dx.doi.org/10.1016/j.ins.2013.12.029

**Table 7**
Comparative results of FS1 and mFS1.

| Data sets | Opt. no. of features ($r1$) | FS1 | | | mFS1 | | |
|---|---|---|---|---|---|---|---|
| | | SE | CPI | MCE | SE | CPI | MCE |
| Synthetic | 0 (3) | **0.23** | **70.33** | **30.53** | **0.23** | **70.33** | **30.53** |
| Iris | 0 (1) | 0.69 | 49.91 | 52.53 | **0.08** | **94.67** | **6.40** |
| (Augmented) Iris | 0 (2) | 0.81 | 84.67 | 28.60 | **0.66** | **96.00** | **7.07** |
| WBC | 2 | **0.40** | **91.36** | 10.29 | 0.45 | 89.60 | **7.79** |
| Glass | 2 | 0.73 | 30.83 | 58.27 | **0.29** | **63.08** | **50.37** |
| Wine | 3 | **0.92** | 42.55 | 32.02 | 0.99 | **64.61** | **8.54** |
| Ionosphere | 8 | 0.32 | 91.74 | 10.48 | **0.30** | **92.59** | **10.40** |
| Sonar | 8 | 0.50 | 61.06 | 29.23 | **0.42** | **88.94** | **20.58** |
| Vehicle | 3 | 0.88 | 29.65 | 62.87 | **0.67** | **96.34** | **49.05** |
| Liver | 2 | 0.45 | 50.00 | 46.99 | **0.37** | **65.51** | **45.16** |
| Thyroid | 1 | **0.59** | 40.05 | 19.86 | **0.59** | **40.64** | **11.49** |
| SPECT Heart | 4 | 0.38 | 68.91 | 31.39 | **0.37** | **83.15** | **20.79** |
| Colon | 179 | **0.33** | **93.55** | 25.97 | **0.33** | **93.55** | **22.10** |
| Leukemia | 547 | **0.36** | 85.05 | **17.65** | **0.36** | **87.59** | 20.59 |

**Table 8**
Comparative results of FS2 and mFS2.

| Data sets | Opt. no. of features ($r1$) | FS2 | | | mFS2 | | |
|---|---|---|---|---|---|---|---|
| | | SE | CPI | MCE | SE | CPI | MCE |
| Synthetic | 0 (3) | 0.50 | 41.67 | 66.73 | **0.06** | **100.00** | **0.00** |
| Iris | 0 (1) | 0.69 | 46.09 | 53.67 | **0.08** | **94.67** | **6.40** |
| (Augmented) Iris | 0 (2) | 0.81 | 84.67 | 28.67 | **0.02** | **97.33** | **7.93** |
| WBC | 2 | 0.44 | 86.09 | 17.13 | **0.35** | **92.39** | **7.86** |
| Glass | 2 | 0.73 | 30.06 | 59.25 | **0.37** | **46.23** | **57.06** |
| Wine | 3 | 0.91 | 44.48 | 41.63 | **0.00** | **100.00** | **27.64** |
| Ionosphere | 8 | 0.35 | 51.41 | **16.52** | **0.27** | **98.29** | 17.52 |
| Sonar | 8 | 0.98 | 51.15 | 48.94 | **0.33** | **88.46** | **19.81** |
| Vehicle | 3 | 0.88 | 29.53 | 62.59 | **0.03** | **98.70** | **44.42** |
| Liver | 2 | 0.45 | 46.96 | 45.28 | **0.16** | **96.23** | **42.03** |
| Thyroid | 1 | **0.59** | 34.98 | 21.44 | **0.59** | **46.42** | **11.35** |
| SPECT Heart | 4 | 0.45 | 63.46 | 33.33 | **0.34** | **83.91** | **22.70** |
| Colon | 179 | **0.32** | 87.10 | 25.16 | 0.44 | **93.55** | **25.00** |
| Leukemia | 547 | **0.34** | 85.28 | 26.47 | 0.40 | **94.44** | **2.94** |

**Table 9**
Comparative results of BE and mBE.

| Data sets | Opt. no. of features ($r1$) | BE | | | mBE | | |
|---|---|---|---|---|---|---|---|
| | | SE | CPI | MCE | SE | CPI | MCE |
| Synthetic | 0 (3) | 0.29 | 66.00 | 36.83 | **0.06** | **100.00** | **0.00** |
| Iris | 0 (1) | 0.69 | 47.16 | 53.67 | **0.45** | **89.33** | **5.67** |
| (Augmented) Iris | 0 (2) | 0.81 | 84.67 | 28.60 | **0.02** | **97.33** | **8.00** |
| WBC | 2 | 0.44 | 86.09 | 16.68 | **0.35** | **92.39** | **6.79** |
| Glass | 2 | **0.27** | 46.36 | **45.19** | 0.37 | **51.14** | 57.34 |
| Wine | 3 | **0.98** | 40.41 | **15.62** | 0.99 | **55.62** | 22.47 |
| Ionosphere | 8 | 0.32 | 91.17 | **9.86** | **0.29** | **95.16** | 15.47 |
| Sonar | 8 | 0.47 | 59.13 | **29.76** | **0.27** | **86.63** | 31.44 |
| Vehicle | 3 | 0.87 | 32.24 | 66.90 | **0.03** | **98.70** | **45.22** |
| Liver | 2 | 0.45 | 48.99 | 46.20 | **0.43** | **89.86** | **41.16** |
| Thyroid | 1 | **0.59** | 37.29 | 22.79 | 0.88 | **43.71** | **21.49** |
| SPECT Heart | 4 | **0.40** | **72.23** | 33.18 | **0.40** | 71.66 | **25.88** |
| Colon | 179 | **0.33** | 93.55 | **25.65** | 0.43 | 91.94 | 34.35 |
| Leukemia | 547 | **0.34** | **83.47** | **26.47** | 0.46 | 78.98 | **26.47** |



hoc manner. In order to show the effectiveness of the mCE definition over the old CE definition, we have demonstrated the comparison between the old and modified unsupervised methods in Tables 6–9. The best results are shown in bold face. The results in these tables are self explanatory. For most of data sets, all four versions of the proposed method have performed

Q1 Please cite this article in press as: M. Banerjee, N.R. Pal, Feature selection with SVD entropy: Some modification and extension, Inform. Sci. (2014), http://dx.doi.org/10.1016/j.ins.2013.12.029

**Table 10**
Comparative results of UDFS and mSR.

| Data sets | Opt. no. of features ($r1$) | UDFS | | | mSR | | |
|---|---|---|---|---|---|---|---|
| | | SE | CPI | MCE | SE | CPI | MCE |
| Synthetic | 0 (3) | 0.10 | **100.00** | **0.00** | **0.06** | **100.00** | **0.00** |
| Iris | 0 (1) | 0.45 | 89.33 | 7.93 | **0.08** | **94.67** | **6.40** |
| (Augmented) Iris | 0 (2) | 0.66 | 96.00 | 8.87 | **0.02** | **97.33** | **7.93** |
| WBC | 2 | 0.45 | 89.60 | 14.42 | **0.35** | **92.39** | **6.63** |
| Glass | 2 | 0.73 | 30.79 | 61.50 | **0.37** | **50.81** | **56.45** |
| Wine | 3 | 1.00 | 44.23 | 24.49 | **0.99** | **55.62** | **22.70** |
| Ionosphere | 8 | 0.29 | 96.01 | **13.11** | **0.28** | **97.44** | 17.04 |
| Sonar | 8 | 0.33 | 53.51 | 30.67 | **0.32** | **85.58** | **24.47** |
| Vehicle | 3 | 0.68 | 97.28 | 46.34 | **0.03** | **98.70** | **44.57** |
| Liver | 2 | 0.65 | 88.12 | 43.42 | **0.43** | **89.86** | **41.33** |
| Thyroid | 1 | **0.59** | 33.18 | 21.67 | **0.59** | **40.76** | **11.77** |
| SPECT Heart | 4 | 0.44 | 70.44 | 33.11 | **0.40** | **71.52** | **27.45** |
| Colon | 179 | **0.31** | 88.71 | **24.19** | 0.44 | **93.55** | 27.42 |
| Leukemia | 547 | **0.20** | 89.35 | **5.88** | 0.42 | **91.16** | **5.88** |

significantly better than the respective approach given be Varshavsky et al. [39], in terms of the three performance indices. These results demonstrate the effectiveness and superiority of the modified *mCE* definition.

Now, we compare the performance of mSR and UDFS [40]. We have chosen mSR based on two reasons, (i) both UDFS and mSR cannot remove redundancy between the selected features and ii) among the four versions, mSR is computationally most efficient as well as consistently well performed. From the results in Table 10, we can conclude that mSR outperforms UDFS for most of the data sets. We know that mFS2 and mBE also do not exclude redundancy. So, one might want to look at their comparative analysis too. In order to address this, we look at the results of mFS2, mBE and UDFS from Tables 8–10. From these tables, we can see that mFS2 and BE also perform better than UDFS. So in conclusion, our methods not only perform better than the old definition, but they also outperform the well-known method, UDFS [40]. Unlike the other three, mFS1 can exclude redundancy which gives it an additional advantage.

## 9. Proposed supervised approach

We now propose a supervised version of the SVD-entropy method which can be used for selecting features when class information is known. Let $X \subseteq \mathbb{R}^{n\times p}$ be a data set with $n$ instances in $p$ dimension. From the discussion in Section 5 we know that, $E(X)$ in (5) is the SVD-entropy of the data set $X$. Contribution of the $i$th feature to the entropy, $mCE_i$, $\forall i = 1,\ldots,p$, can be estimated using Eq. (7). If the $i$th feature can represent the data set ($X$) well, it will have a greater influence on the *large* singular values of $X$, and hence resulting in a high $mCE_i$ score.

Let us divide the feature set into two disjoint subsets, $P_1$ and $P_2$. If the contribution of a feature $i$ to the entropy for data set ($X$), is positive ($mCE_i > 0$), then it belongs to $P_1$, else to $P_2$. When can the $mCE_i$ value for a feature $i$ be negative? If the $i$th feature has some influence on the low singular values and small or negligible influence on the high ones, then the distribution of the eigenvalues, obtained after removing this feature is steeper than the earlier distribution, i.e. $E(X^{-i}) < E(X)$, where $X^{-i}$ is the data matrix of size $n \times (p-1)$, obtained by dropping the $i$th column of $X$. Hence, from Eq. (7), for the $i$th feature, we have $mCE_i < 0$.

The features with influence only on low singular values cannot represent a data set better than the features with greater influence on high singular values. From the earlier discussion, we can say that, any feature $i \in P_1$ should have a better rank than a feature $j \in P_2$. In the unsupervised approach, mSR, this is done automatically, as features with smaller *mCE* scores are selected with lower priorities.

Let, the data set $X$ consists of $k$ classes. The $j$th class has $n_j$ number of data points, i.e. $\sum_{j=1}^{k} n_j = n$. Let, $X_j$ be the set of data points corresponding to the $j$th class, i.e. $X_j \subseteq \mathbb{R}^{n_j\times p}$. If the feature $i$ can distinguish the $j$th class from other classes, it is expected to have the following two properties. First, this $i$th feature should have little effect on the underlying structure of the data set $X_j$, as the $i$th feature is supposed to have similar values for all points in $X_j$, and thus should not have much influence on its singular values. So, $mCE_i$ for the data part $X_j$, say $mCE_i(X_j)$ should be low. Second, it should also be able to represent the underlying structure of the complete data set ($X$) well. Hence, from our earlier discussion we have, $mCE_i$ for $X$, say $mCE_i(X)$ would be high. Thus, we can say that the feature $i_1$ is a better discriminator for class $j$ than the feature $i_2$, if $(mCE_{i_1}(X) - mCE_{i_1}(X_j)) > (mCE_{i_2}(X) - mCE_{i_2}(X_j))$, when both $i_1$ and $i_2$ belong to the same subset $P$, where $P \in \{P_1, P_2\}$. Let, for the $i$th feature and the $j$th class,

$$mCE_{i,j} = mCE_i(X) - mCE_i(X_j), \quad \forall i = 1,\cdots,p \quad \text{and} \quad \forall j = 1,\cdots,k. \tag{8}$$

For the $j$th class we rank the features based on their $mCE_{i,j}$ values (for each subset $P$, $P \in \{P_1, P_2\}$, separately). In other words, if $mCE_{i_1,j} > mCE_{i_2,j}$, then for the $j$th class the feature $i_1$ is of higher priority than the feature $i_2$, given $i_1$ and $i_2$ belong to the

Please cite this article in press as: M. Banerjee, N.R. Pal, Feature selection with SVD entropy: Some modification and extension, Inform. Sci. (2014), http://dx.doi.org/10.1016/j.ins.2013.12.029

same subset. To obtain the overall ranking of features for the entire data set ($X$), from these class wise rankings, we adopt the following approach.

For the $i$th feature, we find the maximum $mCE_{i,j}$ value, $\forall j = 1, \ldots, k$. Let, this maximum value be $M\_mCE_i$. Now, rank all features in $P, P \in \{P_1, P_2\}$, with respect to their $M\_mCE_i$ values, where $i \in P$ and a feature with higher $M\_mCE_i$ value will have a better rank. And ultimately, any feature $i \in P_1$ will have a higher rank than any feature $j \in P_2$. The pseudo-code for this Simple Ranking based supervised approach (sSR) is shown in Fig. 3.

In this work, we also develop a Forward Selection based supervised algorithm (sFS1), following the mFS1 framework. This algorithm proceeds as follows. The first feature is chosen to be the top feature selected by sSR. Let this feature be $f_1$. Like mFS1, we want that the second feature, $f_2$ together with the first feature, $f_1$ to have a high SVD-entropy, $E(X^{f_1,f_2})$, where $f_2 \neq f_1$. In addition, we also want this entropy value to increase within a class; in other words, we want an increase in $E\left(X_j^{f_1,f_2}\right)$ for the $j$th class, where $X_j^{f_1,f_2}$ is the projection of $X_j$ on the selected 2-feature space. So unlike mFS1, here we select the second feature as the $k$th feature, where $k = \underset{i \neq f_1}{\operatorname{argmax}} \left\{ E(X^{f_1,i}) + \max_{j=1}^{k} E\left(X_j^{f_1,i}\right) \right\}$.

Suppose, in step $t$, we have already selected $t$ features and want to select the $(t+1)$th feature. So, together with these $t$ selected features, $f_1, \ldots, f_t$, we shall add each of the remaining $(p-t)$ features, one at a time, to form a $(t+1)$ features set. Then, we chose the $(t+1)$th important feature as $\underset{i \notin \{f_1, \ldots, f_t\}}{\operatorname{argmax}} \left\{ E(X^{f_1,\ldots,f_t,i}) + \max_{j=1}^{k} E\left(X_j^{f_1,\ldots,f_t,i}\right) \right\}$. Again all features in $P_1$ will be selected before any feature in $P_2$. Repeat this process until $r1$ features are selected. As we try to maximize the SVD-entropy of the reduced data set in each step, like mFS1, sFS1 can also remove redundancy. Fig. 4 shows the pseudo-code of this algorithm.

The computational complexities of sSR and sFS1 are similar to their respective unsupervised counterparts, as analyzed in Section 6.

## 10. Experimental results

Here we discuss the results of our proposed supervised schemes, sSR and sFS1, on some data sets in detail. As these are supervised methods, we have taken MCE as a performance measure to assess the selected features. Both of these methods are based on the $mCE$ definition, so the optimum number of features is taken as $r1$. As mentioned in Section 7, if $r1$ is zero for a data set, we have chosen it in an ad hoc manner. The organizations of the tables are similar to those in Section 7.

For Synthetic data set, ranking of features with sSR and sFS1 and the corresponding $MCE$ values are given in Table 11. Here, the optimum number of features ($r1$) is taken as 3. Table 11 reveals that with the first 3 features, sSR yields 100% classification accuracy, while sFS1 achieves 100% accuracy with just the top two features. As far as classification accuracy is concerned, both of these methods perform equally good. But, the feature subsets selected by these two methods are different. Here, sSR selects feature 4 as the second important feature, but sFS1 selects feature 3 with rank 2. It is because the feature pairs {2,4} and {3,5} are highly correlated (Pearson's correlation coefficient is practically 1.00) and sFS1 tries to exclude redundant features. Then one can ask, why sFS1 has selected feature 4 as the third feature? To address this issue, we look at the $mCE$ values of the features. We find that features 2, 3, 4 and 5 belong to partition $P_1$ with positive $mCE$ and the rest are in $P_2$. As all features in $P_1$ should precede the features in $P_2$, sFS1 had only two choices, either feature 4 or feature 5 to be selected as the third feature. Now as mentioned earlier, each of them is highly correlated with one of the already selected features. Hence, the ranking of sFS1 is justified.

The result for Iris data is given in Table 12. Here the optimum number of features is 1. We know that features 3 and 4 are the two most discriminative features of this data. Both sSR and sFS1 rank feature 3 in the first position. Pearson's correlation coefficient between features 3 and 4 is very high (0.96). So, when sSR selects feature 4 as the second feature, to avoid redundancy, sFS1 chooses feature 1 instead.

**Input: Data Set** $X \subseteq \mathbb{R}^{n \times p}$
**Output: Overall Ranking of Features.**

**Compute M_mCE$_i$ values $\forall i = 1, \cdots, p$:**
for $i = 1$ *to* $p$
    $M\_mCE_i = \max_{j=1}^{k} mCE_{i,j}$
end

**Ranking of features for each partition:**
    Sort $M\_mCE_i$ values, $\forall i \in P_1$, in non-increasing order.
    Sort $M\_mCE_i$ values, $\forall i \in P_2$, in non-increasing order.
**Overall ranking of features =** The ranking of features obtained from $P_1$;
    Followed by,
    The ranking of features obtained from $P_2$.

**Fig. 3.** Supervised Simple Ranking (*sSR*) algorithm.

Q1 Please cite this article in press as: M. Banerjee, N.R. Pal, Feature selection with SVD entropy: Some modification and extension, Inform. Sci. (2014), http://dx.doi.org/10.1016/j.ins.2013.12.029

**Input: Data Set** $X \subseteq \mathbb{R}^{n\times p}$
**Output:** $\mathcal{F}$ **Containing Top** $r1$ **Features.**

$\mathcal{F} = \emptyset$.
Select the first feature $f_1$ following $sSR$. $\mathcal{F} = \mathcal{F} \cup f_1$.

for $t = 1$ *to* $(r1 - 1)$
  if $t < |P_1|$, $P = P_1$. Else $P = P_2$.
  $\forall i \in P$ and $i \notin \mathcal{F}$,
  Compute $E_i = \left\{ E\left(X^{f_1,\cdots,f_t,i}\right) + \max_{j=1}^{k} E\left(X_j^{f_1,\cdots,f_t,i}\right) \right\}$.
  $f_{t+1} = \underset{i}{\operatorname{argmax}}\, E_i$. $\mathcal{F} = \mathcal{F} \cup f_{t+1}$.
end

Return $\mathcal{F}$.

**Fig. 4.** Supervised Forward Selection (*sFS*1) algorithm.

**Table 11**
Results of the synthetic data set with sSR and sFS1.

| Ranks | sSR | | sFS1 | |
|---|---|---|---|---|
| | Feature no. | MCE | Feature no. | MCE |
| 1 | 2 | 34.33 | 2 | 34.33 |
| 2 | 4 | 37.83 | 3 | 0 |
| 3 | 5 | 0 | 4 | 0 |
| 4 | 3 | 0 | 5 | 0 |
| 5 | 1 | 0 | 7 | 0 |
| 6 | 6 | 0 | 1 | 0 |
| 7 | 7 | 0 | 6 | 0 |

**Table 12**
Results of the Iris data set with sSR and sFS1.

| | sSR | | sFS1 | |
|---|---|---|---|---|
| | Feature no. | MCE | Feature no. | MCE |
| 1 | 3 | 6.67 | 3 | 6.67 |
| 2 | 4 | 3.67 | 1 | 7.73 |
| 3 | 1 | 4.27 | 4 | 4.27 |
| 4 | 2 | 4.01 | 2 | 4.01 |

**Table 13**
Results of the ionosphere data set with sSR and sFS1.

| Ranks | sSR | | sFS1 | |
|---|---|---|---|---|
| | Feature no. | MCE | Feature no. | MCE |
| 1 | 5 | 21.25 | 5 | 21.25 |
| 2 | 17 | 17.86 | 20 | 15.01 |
| 3 | 19 | 14.16 | 8 | 10.63 |
| 4 | 33 | 13.16 | 29 | 9.82 |
| 5 | 15 | 13.07 | 22 | 11.62 |
| 6 | 25 | 12.82 | 19 | 10.74 |
| 7 | 21 | 14.18 | 33 | 10.34 |
| 8 | 13 | 12.99 | 27 | 9.72 |
| 34 | All Features | 13.22 | All Features | 13.22 |

548 For Ionosphere data set, the optimum number of features ($r1$) is 8. So by looking at Table 13, we find that using $r1$ features
549 sSR and sFS1 gives 12.67% and 9.57% misclassification errors respectively, which is even better than using all the features. We
550 can also see that, both of these algorithms do not select the indifferent 2nd feature among the important ones. To inspect the
551 performance of sFS1 in terms of redundancy removal, we calculate the maximum correlation between all pairs of features in
552 the selected $r1$ features. For sFS1 this value is 0.58, whereas in case of sSR it is 0.83, which is significantly high.

**Table 14**
Comparative results with MSVM-RFE and RCFS.

| Data sets | Opt. no. of features ($r1$) | MCE | | MCE | |
|---|---|---|---|---|---|
| | | MSVM-RFE | sSR | RCFS | sFS1 |
| Synthetic | 0 (3) | **0.00** | **0.00** | **0.00** | **0.00** |
| Iris | 0 (1) | **6.67** | **6.67** | **6.40** | **6.40** |
| (Augmented) Iris | 0 (2) | 6.54 | **6.33** | 4.14 | **3.13** |
| WBC | 2 | 8.40 | **6.94** | 7.67 | **7.47** |
| Glass | 2 | **47.29** | 55.14 | **44.78** | 49.63 |
| Wine | 3 | 26.27 | **25.22** | 25.28 | **9.49** |
| Ionosphere | 8 | 12.56 | **10.45** | 13.11 | **9.63** |
| Sonar | 8 | **21.47** | 25.19 | 23.72 | **17.60** |
| Vehicle | 3 | **41.94** | 45.22 | **43.38** | 50.14 |
| Liver | 2 | 42.98 | **41.65** | 43.25 | **41.77** |
| Thyroid | 1 | 20.97 | **11.53** | **11.53** | **11.53** |
| SPECT Heart | 4 | 27.43 | **22.21** | 24.94 | **21.69** |
| Colon | 179 | 18.16 | **17.74** | **18.73** | 25.65 |
| Leukemia | 547 | **1.18** | 5.88 | **1.76** | 5.88 |

Thus, the performance of sSR and sFS1 on these three data sets, not only portrays the effectiveness of these two schemes but also demonstrates the ability of sFS1 to eliminate redundancy among the selected features.

## 11. Comparison

In this section, we evaluate the effectiveness of our proposed supervised methods by comparing them with two well-known supervised algorithms *Multi-Class SVM-Recursive Feature Elimination* (MSVM-RFE) [41] and *Redundancy Constrained Feature Selection* (RCFS) [42]. As sSR does not exclude redundancy like MSVM-RFE, we have compared its performance with MSVM-RFE in Table 14. Whereas, both RCFS and sFS1 try to remove redundancy among the selected features. So a comparative study between these two methods is also given in the same table.

The comparison between sSR and MSVM-RFE portrays the effectiveness of our method over MSVM-RFE for most of the data sets. The sFS1 algorithm performs better than RCFS for 7 out of 14 data sets in terms of MCE and they select similar features (perform the same) for 3 data sets. If we compare sFS1 with MSVM-RFE, we find that sFS1 outperforms for 10 out of the 14 data sets. So both with or without redundancy removal, our proposed supervised methods are very effective compared to these state-of-the-art methods.

## 12. Conclusion

We have analyzed the unsupervised feature selection approach proposed by Varshavsky et al. [39]. We have pointed out some limitations of this approach and have proposed a modification to improve its performance. However, SVD entropy based methods has a limitation. These methods may not be able to discard even indifferent features having a constant value. For example, in case of the Ionosphere data, the second feature has all zero values, but it has been selected by mSR, mFS2 and mBE with a high rank. The optimum numbers of features, for all methods are taken as the number of features suggested by the scheme in [39]. Although this has given a common platform for comparison, it has not been a good choice for some well-behaved data sets: Iris, (Augmented) Iris and Synthetic data. Our experimental results have demonstrated the utility of the modified definition, *mCE*. We compared the proposed unsupervised algorithm with one well-known scheme UDFS [40]. Our method is found to outperform UDFS for most of the data sets.

We have extended this scheme to a supervised framework. Two supervised methods, sSR and sFS1, have been proposed. The usefulness of these two methods and the ability of sFS1 to exclude redundancy have been demonstrated through experimental results. The performance of sSR has been compared with a well-known supervised algorithm MSVM-RFE [41]. We also made a comparative study between sFS1 and a state-of-the-art method RCFS [42], as both of them try to remove redundancy among the selected features. These comparisons have established the effectiveness and superiority of the proposed supervised methods.

## References

[1] O. Alter, P.O. Brown, D. Botstein, Singular value decomposition for genome-wide expression data processing and modeling, Proc. Natl. Acad. Sci. (PNAS) USA 97 (18) (2000) 10101–10106.
[2] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, D. Sorensen, LAPACK Users' Guide, third ed., Society for Industrial and Applied Mathematics, 1999.
[3] Christos Boutsidis, Michael W. Mahoney, Petros Drineas, Unsupervised feature selection for the *k*-means clustering problem, in: NIPS, 2009, pp. 153–161.
[4] S. Chatterjee, A.S. Hadi, Influential observations, high leverage points, and outliers in linear regression, Stat. Sci. 1 (1986) 379–393.

Please cite this article in press as: M. Banerjee, N.R. Pal, Feature selection with SVD entropy: Some modification and extension, Inform. Sci. (2014), http://dx.doi.org/10.1016/j.ins.2013.12.029

[5] M. Dash, H. Liu, J. Yao, Dimensionality reduction for unsupervised data, in: Proceedings of 19th IEEE International Conference on Tools with AI, ICTAI, 1997.
[6] M. Devaney, A. Ram, Efficient feature selection in conceptual clustering, in: Proceedings of Machine Learning: Fourteenth International Conference, Nashville, TN, 1997.
[7] C. Ding, H. Peng, Minimum redundancy feature selection from microarray gene expression data, in: Proceedings of the Computational Systems Bioinformatics Conference, 2003, pp. 523–529.
[8] Werner Dinkelbach, On nonlinear fractional programming, Manage. Sci. 13 (7) (1967) 492–498.
[9] E.R. Dougherty, Small sample issue for microarray-based classification, Comparat. Funct. Genom. 2 (2001) 28–34.
[10] F. Douglas, Knowledge acquisition via incremental conceptual clustering, Mach. Learn. 2 (2) (1987) 139–172.
[11] S. Dudoit, J. Fridlyand, T.P. Speed, Comparison of discrimination methods for the classification of tumors using gene expression data, J. Am. Stat. Assoc. 97 (457) (2002) 77–87.
[12] M.A. Gluck, J.E. Corter, Information uncertainty and the utility of categories, in: Proceedings of the Seventh Annual Conference of the Cognitive Science Society, Lawrence Erlbaum Associates, Irvine, CA, 1985, pp. 283–287.
[13] T.R. Golub, Molecular classification of cancer: class discovery and class prediction by gene expression monitoring, Science 286 (1999) 531–537.
[14] Isabelle Guyon, Jason Weston, Stephen Barnhill, Vladimir Vapnik, Gene selection for cancer classification using support vector machines, Mach. Learn. 46 (1–3) (2002) 389–422.
[15] Xiaofei He, Deng Cai, Partha Niyogi, Laplacian score for feature selection, in: NIPS, MIT Press, 2005.
[16] Magnus R. Hestenes, Inversion of matrices by biorthogonalization and related results, J. Soc. Indust. Appl. Math. 6 (1) (1958) 51–90.
[17] Yi Hong, Sam Kwong, Yuchou Chang, Qingsheng Ren, Unsupervised feature selection using clustering ensembles and population based incremental learning algorithm, Pattern Recogn. 41 (9) (2008) 2742–2756.
[18] L. Hubert, P. Arabie, Comparing partitions, J. Classification (1985) 193–218.
[19] Rodolphe Jenatton, Jean-Yves Audibert, Francis Bach, Structured variable selection with sparsity-inducing norms, J. Mach. Learn. Res. 12 (2011) 2777–2824.
[20] P. Kohavi, G.H. John, Wrappers for feature subset selection, Artif. Intell. (1997) 273–324.
[21] D. Koller, M. Sahami, Toward optimal feature selection, in: Proceedings of the Thirteenth International Conference on Machine Learning, 1996, pp. 284–292.
[22] Honglak Lee, Alexis Battle, Rajat Raina, Andrew Y. Ng, Efficient sparse coding algorithms, Advances in Neural Information Processing Systems, vol. 19, MIT Press, 2007, pp. 801–808.
[23] Zechao Li, Yi Yang, Jing Liu, Xiaofang Zhou, Hanqing Lu, Unsupervised feature selection using nonnegative spectral analysis, in: AAAI, 2012.
[24] M.W. Mahoney, P. Drineas, Cur matrix decompositions for improved data analysis, in: Proceedings of the National Academy of Sciences, 2009, pp. 697–702.
[25] K.Z. Mao, Identifying critical variables of principal components for unsupervised feature selection, IEEE Trans. Syst. Man Cybernet. Part B 35 (2) (2005) 339–344.
[26] Tomomi Matsui, Yasufumi Saruwatari, Maiko Shigeno, An analysis of dinkelbach's algorithm for 0–1 fractional programming problems. Technical report, Dept. Math. Eng. Inf. Phys., Univ. Tokyo, Japan, 1992.
[27] N.R. Pal, K. Aguan, A. Sharma, S. Amari, Discovering biomarkers from gene expression data for predicting cancer subgroups using neural networks and relational fuzzy clustering, BMC Bioinform. 8 (2007) 5.
[28] Nikhil R. Pal, A fuzzy rule based approach to identify biomarkers for diagnostic classification of cancers, in: FUZZ-IEEE, 2007, pp. 1–6.
[29] J.M. Pena, J.A. Lozano, P. Larranga, I. Iwza, Dimensionality reduction in unsupervised learning of conditional gaussian networks, IEEE Trans. Pattern Anal. Mach. Intell. 23 (6) (2001).
[30] J.W. Sammon, A nonlinear mapping for data structure analysis, IEEE Trans. Comput. C-18 (1969) 401–409.
[31] A. Saxena, N.R. Pal, M. Vora, Evolutionary methods for unsupervised feature selection using Sammon's stress function, Fuzzy Inform. Eng. 2 (3) (2010) 229–247.
[32] Alexander Schrijver, Theory of Linear and Integer Programming, John Wiley & Sons, Chichester, 1986.
[33] Chunhua Shen, Hongdong Li, Michael J. Brooks, Supervised dimensionality reduction via sequential semidefinite programming, Pattern Recogn. 41 (12) (2008) 3644–3652.
[34] Jianbo Shi, Jitendra Malik, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (1997) 888–905.
[35] C. Studholme, D.L.G. Hill, D.J. Hawkes, An overlap invariant entropy measure of 3d medical image alignment, Pattern Recogn. (1999) 71–86.
[36] L. Talavera, Dependency-based feature selection for clustering symbolic data, Intell. Data Anal. 4 (2000) 19–28.
[37] Yu-Shuen Tsai, Chin-Teng Lin, George C. Tseng, I-Fang Chung, Nikhil R. Pal, Discovery of dominant and dormant genes from expression data using a novel generalization of snr for multi-class problems, BMC Bioinform. 9 (2008) 425.
[38] Yu-Shuen Tsai, Kripamoy Aguan, Nikhil R. Pal, I-Fang Chung, Identification of single- and multiple-class specific signature genes from gene expression profiles by group marker index, PLoS ONE 6 (2011) e24259.
[39] R. Varshavsky, A. Gottlieb, M. Linial, D. Horn, Novel unsupervised feature filtering of biological data, Bioinformatics 22 (14) (2006) e507–e513.
[40] Yi Yang, Heng Tao Shen, Zhigang Ma, Zi Huang, Xiaofang Zhou, $\ell_{2,1}$-norm regularized discriminative feature selection for unsupervised learning, Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence, vol. 2, AAAI Press, 2011, pp. 1589–1594.
[41] Xin Zhou, David P. Tuck, MSVM-RFE: extensions of SVM-RFE for multiclass gene selection on dna microarray data, Bioinformatics 23 (9) (2007) 1106–1114.
[42] L. Zhou, L. Wang, C. Shen, Feature selection with redundancy-constrained class separability, IEEE Trans. Neural Networks 21 (5) (2010) 853–858.
[43] http://archive.ics.uci.edu/ml/datasets.html.
[44] http://www.ntu.edu.sg/home/elhchen/data.htm.



Q1 Please cite this article in press as: M. Banerjee, N.R. Pal, Feature selection with SVD entropy: Some modification and extension, Inform. Sci. (2014), http://dx.doi.org/10.1016/j.ins.2013.12.029