2014 IEEE International Conference on Data Mining

Noise-Resistant Unsupervised Feature Selection via Multi-Perspective Correlations

Hao Huang*, Shinjae Yoo†, Dantong Yu† and Hong Qin*
*Department of Computer Science, Stony Brook University
Email: haohuangcssbu@gmail.com, qin@cs.stonybrook.edu
†Computational Science Center, Brookhaven National Laboratory
Email: sjyoo@bnl.gov, dtyu@bnl.gov

***Abstract*—Unsupervised feature selection is an important issue for high dimensional dataset analysis. However popular methods are susceptible to noisy instances (observations) or noisy features. We propose a noise-resistant feature selection algorithm by capturing multi-perspective correlations. Our proposed approach, called Noise-Resistant Unsupervised Feature Selection (NRFS), is based on multi-perspective correlation that reflects the importance of feature with respect to noise-resistant representative instances and various global trends from spectral decomposition. In this way, the model concisely captures a wide variety of local patterns. Experimental results demonstrate the effectiveness of our algorithm.**

I. Introduction

Many real world applications have high dimensionality in their feature space. A larger number of features can be associated with expensive data collection cost, more difficulty in model interpretation, expensive computational cost, and sometimes decreased ability of generalization . These challenges are commonly referred to "the curse of dimensionality", and motivate a plethora of research to find a well representative feature subset and thereby reduce the number of features before actual machine learning and analysis. Many feature selection approaches have been developed [34] [29] [35] [7] [9] [31] [10]. In many applications, usually data has no label information, since it is too expensive or difficult to assign labels by experts. Therefore, it is important to develop an unsupervised approach which can perform feature selection task without labels. Compared with the supervised case, the unsupervised feature selection is much more challenging because of the lack of prior knowledge. In this paper, we focus on an unsupervised feature selection due to its broad applicability.

The goal of feature selection is to minimize information loss when removing the noise and redundancy in the feature space [33], therefore can achieve better 1) model interpretation, 2) computational efficiency, and 3) generalization ability. However, there are significant challenges associated with many existing unsupervised feature selection algorithms:

(1) Feature importance is usually more about a "local" conception than a "global" one. To obtain a better representative feature subsets, the feature impact associates with different low-rank embeddings or spectrums need to be considered [7]. Besides, the perspective of instances is also indispensable since some features may only have strong correlations with certain instances with respect to certain spectrums. Therefore it is necessary to design a feature selection algorithm based on such **multi-perspective correlation**.

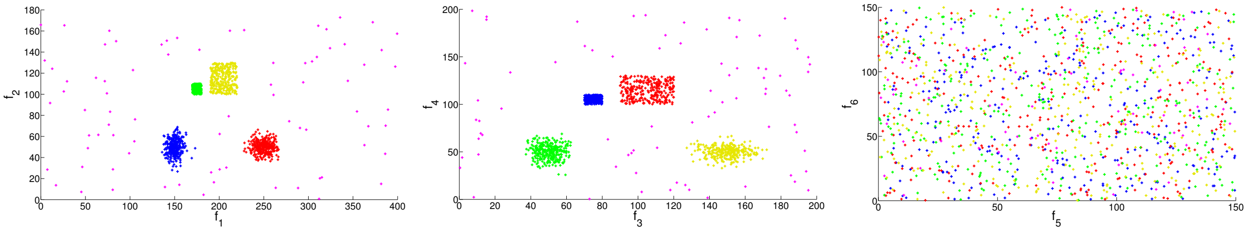
(2) Real world datasets contain many **noisy features** (such as f_5 and f_6 shown in Figure 1(c)). These noisy features have negative impacts and make it difficult to identity the informative features, especially for the existing unsupervised feature selection algorithms [7] [20] [15] [28].

(3) **Noisy observations/instances** (colored as purple in Figure 1(a) and 1(b)) are also very common in real world applications. When a dataset has a significant number of noisy instances, feature importance are hard to discover by most of the unsupervised feature selection algorithms [7] [28] [15] [27] due to that the weights of feature become influenced by noisy instances.

To solve these problems, our proposed method, called **Noise-Resistant Feature Selection (NRFS)**, designs a feature selection strategy based on multi-perspective correlation measurement which is effective and robust to both noisy observations and noisy features. By selecting representative instances via density distribution statistics, we reduce the occurrence of the noise observations. For each feature, we compute its local correlation with regard to the representative instances. Such local correlations are evaluated with respect to each global spectrum of data to find the informative features. Noisy features tend to have lower local correlations across all of the global spectrums compared to the informative ones, while the locally informative features tend to show a strong association to at least one global spectrum. We comprehensively considerate all correlation scores and obtain the informative feature subset. Our paper has the following contributions:

- Our proposed NRFS selects features **under local context** instead of global context. We build a set of similarity matrices, where each similarity matrix is constructed using a local feature subspace (each feature and its nearest neighbor features) (Section III-A). By doing this, we have a local perspective w.r.t. each instance and feature pair, and measure their local correlation with the global spectrums (Section III-B).
- In order to mitigate the influence of noisy instances, we propose the **Noise-Resistant Density-Preserving Sampling** (Section IV). It combines both anomaly detection [18] and Density-Preserving Sampling [4], and selects only representative instances from the original dataset. By only analyzing the feature impact

1550-4786/14 $31.00 © 2014 IEEE
DOI 10.1109/ICDM.2014.88



(a) Distribution on f_1 and f_2

(b) Distribution on f_3 and f_4

(c) Distribution on f_5 and f_6

Fig. 1. Synthetic dataset with four clusters (colored with red, blue, yellow and green respectively), each has 300 instances and 34 features. In addition, there are 80 noisy instances (colored as purple). Fig.1(a) shows the feature subspace of f_1 and f_2, where the blue and red clusters have a Gaussian distribution, while green and yellow clusters show a uniform distribution in a rectangle area. Fig.1(b) shows the feature subspace of f_3 and f_4, where blue and red clusters show uniform distribution in a rectangle area, while green and yellow clusters have a Gaussian distribution. The other 30 features are all noisy, for example f_5 and f_6 shown in Fig.1(c). Through the experimental results listed in Table I we can see that noisy instances can become a hurdle for feature selection, and noisy features, with their quantity even more than that of the informative (useful) ones, could be another issue.

TABLE I. CLUSTERING RESULTS OF SYNTHETIC DATASET IN FIG. 1. THE SIZE OF SELECTED FEATURE SUBSET IS 4 FOR ALL THE FIVE FEATURE SELECTION ALGORITHMS. WE RUN EACH ALGORITHM 30 TIMES ON THE DATASET WITH ALL INSTANCES (INCLUDING NORMAL AND NOISY INSTANCES), AND ALSO ON THE SUBSET OF THE NORMAL INSTANCES (WITHOUT ANY NOISY INSTANCE). WE REPORT THE AVERAGE NMI SCORE ONLY ON THE NORMAL INSTANCES.

Algorithms	K-means	NJW	SPEC[43]	LS[15]	MCFS[7]	NDFS[28]	NRFS (our algorithm)
NMI(all instances)	0.1571	0.1002	0.0017	0.0069	0.0032	0.3147	1.0000
NMI(normal instances)	0.2665	0.1097	0.0032	0.0071	0.0094	0.5004	1.0000

on these representatives, we have a noise-instance-resistant algorithm.

- Our proposed NRFS has a **more stable performance** in that it selects features by comprehensively considering multi-perspective correlation for each feature, each representative instance, and each global spectrum (Section III-C).
- Our proposed NRFS combines all these contributions in a **well-organized framework** (Section V-A), to deliver a more robust feature selection algorithm, as shown in our systematic benchmark evaluation (Section VI).

A. Related Work

He et al. [15] proposed Laplacian Score (LS) which is one of the earliest work to seek features with respect to the manifold structure. It uses a nearest neighbor graph to model the local geometric structure of the data and selects those features which are smoothest on the manifold graph [7]. Similarly, Spectral Feature Selection (SPEC) [43] obtains the feature importance by estimating the feature consistency with the spectrum of a matrix derived from a similarity matrix on the whole feature space. Jiang et al. pointed out the untrustworthiness of the similarity matrix due to noise, and designed Eigenvalue Sensitive Criteria (EVSC) [20] which evaluates the feature importance by measuring the change of graph Laplacian's eigenvalues. Although these methods could find features that are related to the manifold structure to some degree, they cannot necessarily discriminate the feature importance because they are only based on the global context without local view and noise resistance.

Recently many algorithms perform feature selection simultaneously during the model building process [44]. In their work, the embedded modeling usually treats feature selection as a part of training process. The feature importance is obtained by optimizing the objective function of the learning model. The method in [39] puts a l_0-norm constraint into the proposed objective function to achieve sparse and efficient solution. l_1-norm has been used in [40] and Multi-Cluster Feature Selection (MCFS) [7] to recover the global distribution pattern on either similarity or dimensionality on the manifold space. Algorithms in [41] [16] and Nonnegative Discriminative Feature Selection (NDFS) [28] use $l_{2,1}$-norm regularization to achieve similar objectives. Although these methods are effective and robust to some degree, they only focus on the global feature importance by measuring how much each feature can preserve the global distribution pattern on the low-rank embedding dimensions (eigenvectors). Therefore they cannot reveal the local correspondence between each feature-instance pair.

In general, the aforementioned unsupervised feature selection algorithms conduct feature selection globally by producing a common feature subset across all instances at the same time. This, however, might fail to deal with real world noisy datasets in practice, where feature selection becomes challenging in the presence of noisy observations, and where the local intrinsic property of data plays more important role [26]. Li et al. proposed the Localized Feature Selection algorithms [25] [26] which tend to find the optimal feature subsets for each cluster. But these algorithms are either based on K-means or Bayesian variational learning, and not practically robust to real world datasets due to the lack of manifold awareness and noise effect mitigation.

Although projected clustering [1], subspace clustering [13] [24] and co-clustering algorithms [5] [8] can detect local structure through simultaneously clustering on instances and features of a dataset, they cannot provide the relative importance value of each feature. Secondly, finding the correct subspace to define a suitable group of objects is a difficult problem, since cluster objects may reside in arbitrarily oriented, affine subspaces [24]. In addition, most of subspace clustering methods are formulated only for a mixture of linear manifolds and do not work well in the presence of nonlinear manifolds [13].

B. Motivation

We illustrate our motivation using a synthetic noisy dataset with 1280 instances and 34 features in Figure 1. The dataset contains noise in both instance space and feature space. It has four clusters, each cluster contains 300 instances and colored with red, blue, yellow and green respectively. We also added 80 noisy instances which are colored with purple. On the other hand, only the first four features are significantly important: the subspace of f_1 and f_2 in Figure 1(a) shows that the blue and red clusters have a Gaussian distribution, while green and yellow clusters have a uniform distribution in the rectangle area; the subspace of f_3 and f_4 (Figure 1(b)) shows that the blue and red clusters have a uniform distribution in the rectangle area, while green and yellow clusters have a Gaussian distribution. Except these four features, all the other 30 features show noisy distribution, such as f_5 and f_6 shown in Figure 1(c).

There are two characteristics about this synthetic dataset: 1) it has a certain amount of noisy instances that cannot be neglected (corresponds to challenge 1 in Section I). 2) The dataset contains more noisy features than useful features (30 $v.s.$ 4, which corresponds to challenge 2 in Section I). These two characteristics exist in many real world datasets, such as microarray or text datasets.

These two characteristics make the popular unsupervised feature selection algorithms to be difficult to handle. In Table I, we reveal the challenges of the other popular feature selection algorithms. We evaluate K-means clustering results on the selected four-feature subspace from a few popular feature selection algorithms (SPEC [43], Laplacian Scores (LS) [15], MCFS [7] and NDFS [28]). From Table I, we can see that if the noisy observations are filtered out, all the baseline algorithms have better performance (although only slightly better for some algorithms), which indicates that the noisy instances lower the performance. Among the four popular feature selection algorithms, NDFS has the most noticeable improvement after filtering out the noisy observations, since it performs a joint and iterative learning between cluster labels and feature selection matrix that optimizes the objective functions [28]. However, NDFS, as well as the other existing algorithms, still suffers a lot from noisy features and observations.

We here design an advanced unsupervised feature selection algorithm which not only reduces noisy instance effects (challenge 1), but also effectively filter out the noisy features (challenge 2).

II. Notations and Background

We use $X_{**} \in R^{n \times m}$ to denote a high-dimensional dataset with n instances and m features. The corresponding global similarity matrix $W_{**} \in R^{n \times n}$ can be constructed to represent the relationship among instances considering the whole feature space. Gaussian similarity is one of the most generally used options for constructing W_{**}:

$$W_{ij}^{(GAU)} = exp(- \parallel X_{i*} - X_{j*} \parallel^2 /(2\sigma^2)), \tag{1}$$

where σ controls the width of neighborhood [30]. For some datasets with nonuniform sizes such as text datasets we tend to use cosine similarity:

$$W_{ij}^{(COS)} = \frac{X_{i*} \cdot X_{j*}}{\parallel X_{i*} \parallel_2 \cdot \parallel X_{j*} \parallel_2}. \tag{2}$$

The degree matrix D_{**} on W_{**} is defined by $D_{ij} = \sum_{k=1}^{n} W_{ik}$ if $i = j$, and 0 otherwise. Given W_{**} and the corresponding D_{**}, the Laplacian matrix $\mathcal{L}_{**}$ and symmetric normalized Laplacian matrix L_{**}^{sym} are defined as:

$$\mathcal{L} = D - W, \tag{3}$$

$$L^{sym} = D^{-1/2} \mathcal{L} D^{-1/2}. \tag{4}$$

From L_{**}^{sym} we can compute the eigenvectors $Y_{**} \in R^{n \times c}$ ($c \ll m$) which in theory provide the manifold structure of the high-dimensional dataset X_{**} [30]. By carefully setting the value of c, the first c eigenvectors reveal the global distribution pattern of X_{**}. In practice c is usually set as the number of clusters [32].

In 2010, Cai et al. proposed a method called Multi-Cluster Feature Selection (MCFS) [7]. They measured the importance of each feature w.r.t. each column of Y_{**} which corresponds to the contribution of each feature for differentiating clusters [7] by minimizing the following equation:

$$min_{a_{k*}} \parallel Y_{*k} - X a_{k*} \parallel^2 + \beta \mid a_{k*} \mid, \tag{5}$$

where Y_{*k} is the k-th column/eigenvector in Y_{**}, a_{k*} is a $m \times 1$ vector and β is a parameter controls the a_{k*}'s approximation speed to zero. For each feature f_j, they defined the feature importance as:

$$MCFS(f_j) = max_k \mid a_{kj} \mid, \tag{6}$$

where a_{kj} is the j-th element of vector a_{k*}.

III. Multi-perspective Unsupervised Feature Selection

The notion of correlation is essential since it allows us to discover signals with similar patterns and, consequently for feature selection applications, discover each feature contribution to the global spectrums. In this section we consider the correlation among features and global spectrums, and exhibit two important properties:

- The effect of each feature may change over different instances or global spectrums. In this case, a single and static score for each feature regardless of different instances and spectrums would be misleading. It is desirable to have a notion of multi-perspective correlation that evolves with each instance, each feature and each global spectrum.
- The second property is that some informative features w.r.t. certain instance subset exhibit strong but fairly complex, non-linear correlations with global spectrums. Traditional linear measures, such as [7] are less effective in capturing these non-linear relationships. Here we seek a powerful model that can capture such correlations on certain dataset applications.

We introduce a powerful model that can capture multi-perspective correlations inside the high dimensional dataset. It

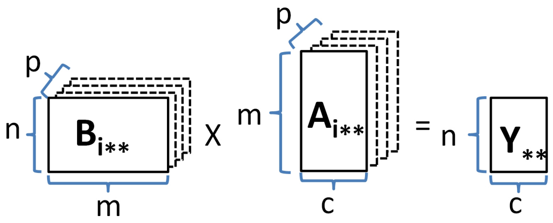


Fig. 2. Multi-layers of matrix (cube) used in our algorithm. Each layer shows a case of Equation 7 with a similarity matrix B_{i**}, coefficient matrix A_{i**} and global spectrums Y_{**}. Equation 7 shows how to construct A_{i**} which represents the multi-perspective correlations [1].

starts with global spectrum derivation and make the spectrums as regression target. Then the association score is measured by comparing the correlation between each global spectrum and each feature on certain instance (representatives). Higher value of association score means higher possibility that the corresponding feature is an informative feature with respect to the related global spectrum.

A. Constructing Similarity Cube

To learn a model of comprehensive feature weighting, we learn from multiple representative instances simultaneously, since each representative instance usually only provides "strong feedback" to a subset of features. We will explain how to choose representative instance in Section IV. To obtain the perspective from each instance representative, we acquire the similarity information between the representative and all the other instances within each local feature subspace. In this way, the influence of each feature to the neighborhood of each representative can be revealed.

Specifically, for each representative instance x_i ($x \leq q$, where p is the number of representative instances) and each feature f_j ($j \leq m$), we construct x_i's similarity vector B_{i*j} (to all instances), which is a $1 \times n$ vector based on the q neighboring features of f_j (including f_j). Using f_j's q neighborhood instead of only f_j itself can generate more stable and informative similarity distribution for each x_i. For those applications with a large feature size, we use fast approximate k-nearest neighborhood search [12] to obtain the neighbors of each feature. After we extract q neighbors for each feature, we construct the corresponding similarity matrix (on the representative instances) within this feature subspace. Therefore for each feature f_j and each representative instance x_i, we obtain a $1 \times n$ similarity vector B_{i*j}. So we have a $p \times n \times m$ three dimensional cube B_{***} shown in Figure 2, where p is the number of representative instances, m is the number of features and n is the number of total instances.

In practice, for those Gaussian distributed dataset we use Gaussian kernel (Equation 1) to reveal the non-linear correlation between global spectrums and original features.

Each B_{i**} shows x_i's similarity with all the instances within each local feature subspace. Next subsection explains, by learning the correlation of these local information to the global spectrums Y_{**}, we can measure how much each feature contributes to the global spectrums for each representative instance. The more it contributes, the more important the corresponding feature is.

B. Learning Coefficient Cube

On the other hand, different instances (representatives) may have very different feature preferences. To qualify these preferences, we here resort to a regression procedure, which typically requires learning from the low-rank model, or global spectrums on instance space, in order to measure the feature contribution across representative instances to different spectrums.

Intuitively we want to extract the "key information" locally contained in the similarity cube B_{***} and measure how close they are to the global spectrums. This is where the spectral decomposition Y_{**} helps. Here Y_{**} is set as the regression target that consists of the first c global spectrums. These spectrums capture the key aperiodic and oscillatory trends that explain the largest fraction of the data variance. Thus, we only consider the low-rank subspace spanned by the first c global spectrums/eigenvectors. Specifically, we compare the feature impact for each representative instance on this low-rank subspace, and extract the correlation score.

For each representative instance x_i in Cube B_{***}, there is one $n \times m$ similarity layer B_{i**}, which contains x_i's information related to all n instances and all m features. Given B_{i**}, we propose the following equation to characterize the correlation between each feature and each global spectrum from the perspective of x_i, i.e. A_{i**}:

$$B_{i**} \times A_{i**} = Y_{**}, \qquad i = 1, 2, ..., p. \tag{7}$$

Equation 7, shown in Figure 2, is a simple regression problem. In practice we solve it with the following ridge regression equation:

$$argmin_{A_{i**}} \|B_{i**} \times A_{i**} - Y_{**}\|^2 + \lambda \|A_{i**}\|^2. \tag{8}$$

which can be solved by using Moore-Penrose pseudoinverse [6]. A_{i**} is a $m \times c$ matrix which represents the coefficients to reconstruct Y_{**} given B_{i**} [1]. This equation is to find the matrix factorization that has minimal reconstruction error on Y_{**}. Because the layer/perspective is independent to each other, more advanced techniques such as Lasso regression would not be necessary. The advantage of using pseudoinverse here is that it is a relatively simple and non-iterative method, and the weights/coefficients can be solved analytically.

The coefficient matrix A_{i**} is of interest because it reflects the correlation between the pattern of the corresponding feature in B_{i**} and the global spectrum Y_{**}. When the value of such coefficients, or interdependence scores are high, the contribution of the corresponding features to the global spectrums are high. These measures can also help us to filter out the noisy features since they tend to have very low correlation with the low-rank embeddings of the whole dataset.

In particular, A_{i*k} provides the correlations of all the features to the global spectrum Y_{*k} with respect to the representative instance x_i. Therefore, for each representative instance x_i, we obtain a $m \times c$ coefficient matrix. The final coefficient

[1] In practice we added one column-vector $\mathbf{1} \in R^n$ in B_{i**} which plays a role of intercept.

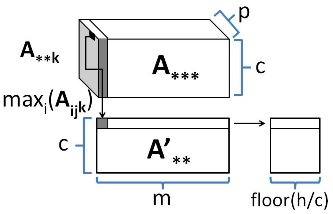


Fig. 3. The selection of feature subset based on the coefficient cube A_{***} (Section III-C).

cube A_{***} is $p \times m \times c$ (Figure 2). The three dimensional cube A_{***} provides a multi-perspective model of different feature weighting across all the representative instances and global spectrums. Therefore, it provides a comprehensive "platform" for an informative feature selection.

C. Feature Selection with Coefficient Cube

Based on the coefficient cube A_{***}, we now select feature subset in a more comprehensive way compared with the other existing methods.

(1) First of all, we need to make all the coefficient measures have the same sign. The coefficients generated from Equation 8 usually have mixed positive and negative values, while the extremes of both sides show a strong correlation. In our algorithm we take the absolute value of coefficient (similar to Equation 6). Also since the "localized" feature selection may result in different value ranges of coefficient, each coefficient vector A_{i*k} should thereby be properly normalized. In our implementation, we use L_2-normalization for each A_{i*k}, therefore the above processing could be represented as:

$$A_{ijk} = |A_{ijk}| / \sqrt{(\sum_{g} |A_{igk}|^2)}, \tag{9}$$

Now the higher the coefficient value is, the more important the feature is to the corresponding pair of representative instance and global spectrum.

(2) We then select the feature subset based on the normalized A_{***}. To preserve the global spectrums with a small amount of observed features, we select representative features from the perspective of each global spectrum. Suppose we need to select no more than h features (usually $h > c$), then $\lfloor h/c \rfloor$ features are chosen for each global spectrum, where c is the number of global spectrums.

In the coefficient cube A_{***}, each global spectrum Y_{*k} corresponds to a $p \times m$ matrix A_{**k}. The first dimension p correlates with the number of representative instances, while the second dimension m corresponds to the number of original features. To study how much a global spectrum values each feature, we need to compress this $p \times m$ matrix A_{**k} into a $1 \times m$ vector A'_{*k}, in which each value A'_{jk} is the weight of feature f_j w.r.t. the corresponding global spectrum Y_{*k}. As shown in Figure 3, we choose the maximum along all the representative instances:

$$A'_{jk} = max_i \{A_{ijk}\}. \tag{10}$$

Now we have a $m \times c$ correlation matrix A'_{**} which shows the relation of features and global spectrums.

(3) For each global spectrum we select $\lfloor h/c \rfloor$ features. Every time when we select w.r.t. A'_{*k}, we choose the $\lfloor h/c \rfloor$ features with the highest coefficient value. And set the elements in the same positions but on the unprocessed columns as 0, in order to avoid duplicate features. Finally we successfully choose $\lfloor h/c \rfloor \times c$ features out of the original feature space.

IV. Noise-Resistant and Density-Preserving Sampling

This section introduces how to select representative instances by our proposed noise-resistant density-preserving sampling. It consists of two components: outlier removal and density-preserving sampling to fulfill the needs of our proposed feature selection algorithm.

A. Noisy Observation Removal

The first step is to remove noisy observations. Here we assume noisy observations are those instances with small neighborhood density, which also called outliers or anomalies. We resort to anomaly detection algorithms [2] [17] [18], which distinguish normal instances from a small portion of abnormal instances (noisy observations). Particularly we apply FDD (Fermi Density Descriptor) [18] due to its effectiveness and stability. It measures the average probability of a fermion appearing at a specific location (corresponds to each instance in high-dimensional coordinates) in the "polarized" manifold space. The computed probability provides the value of anomalousness for each instance. By choosing the stable energy distribution function, FDD steadily distinguishes anomalies from normal instances. In our algorithm, we sort all instances in the descending order of their anomalousness, and remove the first 10% instances. We assume that the majority of the noisy observations are removed after we apply this approach.

B. Density-Preserving Sampling

The second step is down-sampling. Many sampling methods have been proposed [21] [11]. But most of them are stochastic and their sampling results vary significantly from one repetition to another. There is no guarantee that the sample results are inclusively representing the original dataset [4]. In this paper, we adopt a more intelligent sampling approach aiming to produce representative splits with minimum duplications. We use the newly appeared density-preserving sampling (DPS) [4] to eliminate the need for repeating an error estimation procedure by dividing available data into subsets that are guaranteed to represent the input data.

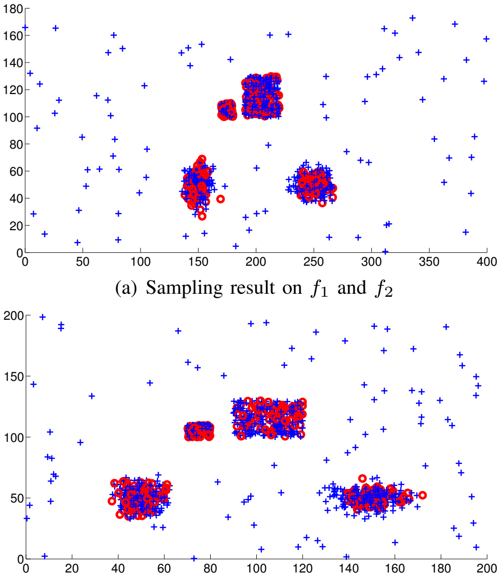
The idea of DPS is inspired by the concept of correntropy which is a nonparametric similarity measurement between two random variables. Since correntropy can be used to measure similarity, it can also be used to measure the quality of a sample to preserve representatives of the whole dataset [4]. DPS uses correntropy as an optimization criterion, guiding the sampling process to split a given dataset into two or more maximally representative subsets. In their paper, Budka et al. proposed correntropy-inspired similarity index (CiSI) between two random variables (datasets) X and Y:

$$CiSI(X,Y) \approx \frac{1}{n} \sum_{i \in (1 \dots n)} G(x_i - y_j, 2\sigma^2 I), \quad i,j = argmin_{i,j} \|x_i - y_j\|, j \in J_{avail}, \tag{11}$$

where $G(x_i - y_j, 2\sigma^2 I)$ denotes a Gaussian kernel centered at $(x_i - y_j)$ to avoid the ordering effect, $\sigma^2 I$ is a diagonal covariance matrix of the Gaussian kernel, $\|\cdot\|$ denotes the Euclidean norm, and the set J_{avail} contains the indices of y which have not yet been used, and it ensures that each y_k is used only once [4]. Since a Gaussian kernel peaks at the 0 Euclidean distance regardless the value of σ, CiSI provides a σ-independent iterative binary procedure to split dataset into subsets X and Y. It selects instances z_i and z_j from dataset Z at each step such that the following equation holds:

$$i,j = argmin_{i,j} \|z_i - z_j\|. \tag{12}$$

Subsequently, z_i and z_j are added into X and Y to maximize CiSI(X,Y). The procedure can be iteratively applied to split X or Y furthermore to get a small enough sample size.



(a) Sampling result on f_1 and f_2

(b) Sampling result on f_3 and f_4

Fig. 4. Sampling result of synthetic dataset in Figure 1. Instances marked with red circles are one of the 25% sampling subsets after noisy instance removal.

Note that the density-preserving sampling is not guaranteed to remove noisy observations/instances. We have to combine both noisy observation detection and density-preserving sampling to obtain the final informative representative instances.

The main property of the above sampling strategy is to produce only representatives while excluding noisy observations. The down-sampling also reduces the running time complexity as shown in Section VI-C. In Figure 4, we show the effect of our sampling strategy with a 25% sample size (means $p = 0.9n/4$ after removing $0.1n$ noisy observations), which demonstrates that the proposed sampling strategy is not only noise-resistant, but also selects representatives with density-preserving.

It is worth noting that given proper normalization, the above sampling strategy can be also applied on text datasets.

V. Noise-Resistant Feature Selection and Theoretical Connections

A. *Noise-Resistant Feature Selection*

In this section, we propose the integrated framework that documents the whole process of NRFS. Let X_{**} be the dataset matrix of size $n \times m$ where n is the number of instances and m is the number of features. Algorithm 1 describes NRFS step by step.

```
Algorithm 1: NRFS($X_{**}$, $h$, $\sigma$ (if use Gaussian kernel), $p$, $q$)
  Input: Input data $X_{**} \in R^{n \times m}$; $h$ is the #selected
         features; $\sigma$ is the Gaussian scaling parameter; $p$
         is the # representative instances; $q$ is the size of
         local feature subspaces.
  Output: Selected feature subset.
1 Construct similarity matrix $W_{**}$ using Equation 2, or
  Equation 1 with $\sigma$ (Section II);
2 Construct symmetric normalized Laplacian matrix $L^{sym}_{**}$
  using Equation 3 and 4 (Section II);
3 Compute generalized eigenvectors $Y_{**}$ (Section II);
4 Remove noisy observations using anomaly detection
  algorithm (Section IV-A);
5 Down sample the remaining dataset to $p$ representative
  instances (Section IV-B) ;
6 Construct cube $B_{***}$ for each sample instance and each
  local feature subspace with $q$ (Section III-A);
7 Learn the coefficient cube $A_{***}$ (Section III-B);
8 Obtain the final feature subset (Section III-C)
```

Through Step $1-3$, we obtain the global spectrum Y_{**} (Section II) as our later regression target. We simply use all instances (including normal and noisy observations) to construct Y_{**}, in that we need to stably rebuild the low-rank embeddings. However it is both sensitive and useless to detect the local correlation w.r.t. noisy instances between features and global spectrums. We thereby remove noisy observations and only focus on the informative representatives, by applying Step 4 and 5 which constitute the Noise-Resistant Density-Preserving Sampling (Section IV). On the other hand, noisy features can be filtered out based on their values of the coefficients in Step $6-8$ (Section III). Here the noisy features are coincident with the low correlation values between the global spectrums and local perspective of the representative instances.

Regarding computational complexity, NRFS is dominated by the eigendecomposition (that gives Y_{**}) which takes $O(n^3)$ and pseudoinverse in Equation 8 that takes $O(p(mn^2 + n^3))$. However, the pseudo inverse can be done parallelly for different representative instance layer.

We run NRFS 30 times on the synthetic dataset in Figure 1 with $p = 288$ and $q = 1$. Each time the four selected features are always f_1, f_3, f_2, f_4 which generate the highest K-means clustering result $NMI = 1$.

TABLE II. STATISTICS OF EXPERIMENTAL DATASETS.

	Dataset	#instances	#features	#clusters
1	11Tumors	174	12534	11
2	Leukemia2	72	11225	3
3	BrainTumor2	50	10368	4
4	Lung	181	12533	3
5	RCV1-4Classes	1200	11370	4
6	Reuter21578A	1000	18933	5
7	20NewsgroupA	800	11269	4
8	20NewsgroupB	800	11217	4

B. Connections with Other Techniques

Our proposed NRFS has close connection with recommendation techniques, of which one popular approach for characterizing the multi-user personalization problem is collaborative modeling [22] [42]. In collaborative modeling, users provide feedback on an absolute scale and the model integrates these feedback and obtain final results. Most of these approaches are motivated by the intuition that even though users have different preferences, many users share preference with other users. Therefore the integrated result can be stable and informative. Similarly, our NRFS treats representative instances and global spectrums as two different kinds of "users". Each of them has its own perspective (feedback) of feature importance. The coefficient cube A_{***} of our NRFS (Section III-B) reveals the two different perspectives to each feature.

On the other hand, different from the target of collaborative modeling, our NRFS tries to locally weight features with multi-perspective correlations. This step is closely related to matrix factorization [23] and fuzzy feature weighting [38] [19]. Our proposed NRFS learns from a low-dimensional latent model Y_{**} which reliably characterize the space of the "user's" dominative yet diverse preferences. It computes a factorization that has a minimal reconstruction error on the latent-variable matrix Y_{**}. Finally, instead of assigning a global importance for each feature, NRFS weights feature according to different perspectives, namely, different global spectrums Y_{*k}. Therefore it is a more comprehensive strategy compared with the other feature selection algorithms.

VI. EXPERIMENTAL ANALYSIS

A. Experimental Setup

Datasets and Preprocessing. To demonstrate the performance of our proposed method, we evaluate our algorithm on four microarray datasets and four text datasets (statistics are summarized in Table II).

The microarray datasets were mainly produced by oligonucleotide based technology [36]. We took the advantage of all available information in order to increase the number of categories or diagnoses for outcome variable, as described in [36]. In summary, the ten microarray datasets have 3-11 distinct diagnostic categories, 50-181 patients (instances) and about $10,000$-$13,000$ genes (features). In the preprocessing phase, we relied on the following three commonly used steps: 1) $base$-10 logarithm [5], 2) standard quantile normalization [3] over multiple chips, and 3) double centering [5] for background correction.

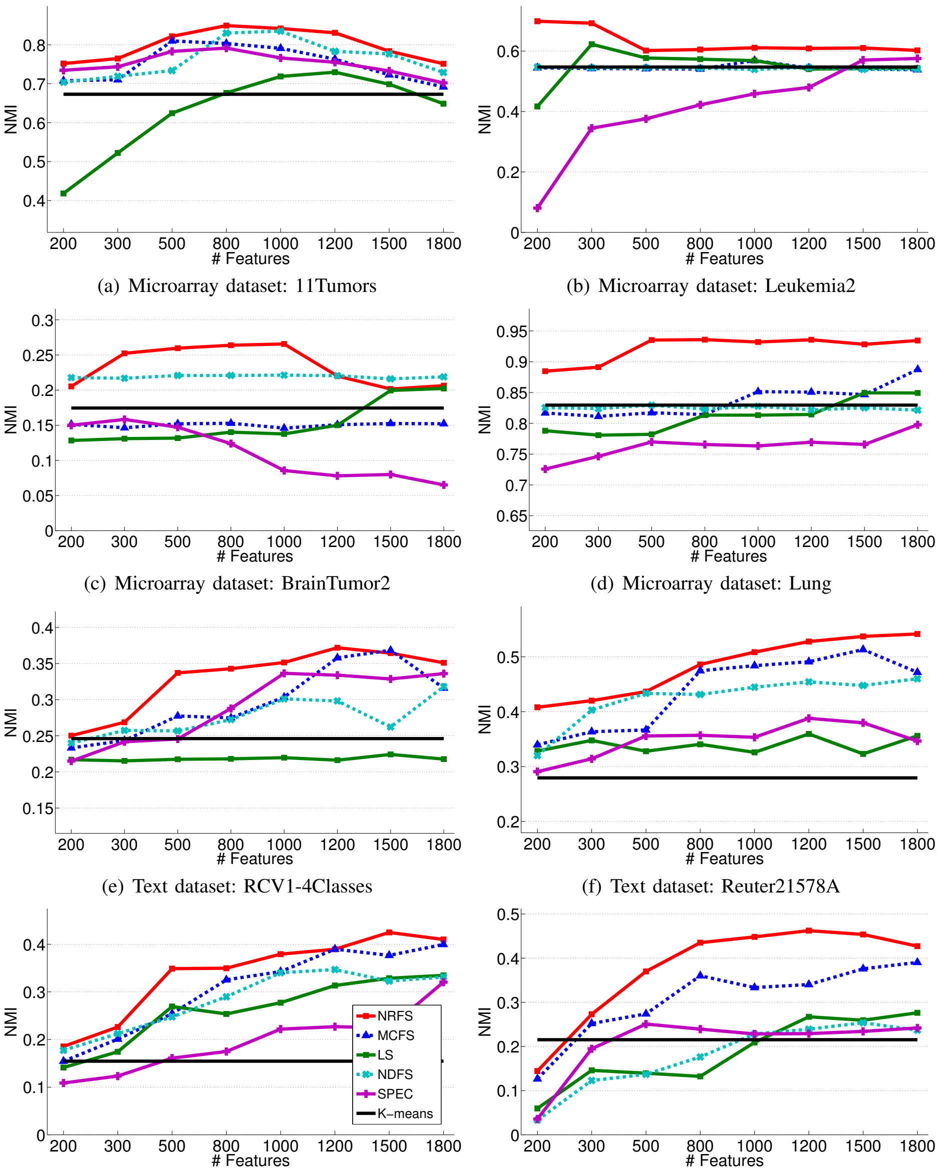
All the four text datasets we used came from large and popularly used datasets: 20Newsgroups, Reuters21578 and RCV1. The original 20Newsgroups has $18,846$ documents (instances) and $26,214$ words (features). 20NewsgroupA has 800 documents, namely 200 documents from four categories: alt. atheism, comp. graphics, rec. autos, and sci. med. 20NewsgroupB has 800 documents and four categories: comp. windows, rec. motorcycles, sci. space, and talk. religion. misc, and each of them takes 200 documents. Note that there is no repetitive category in the above two datasets. The origin Reuters21578 has $8,293$ documents and $18,933$ words. We select 200 documents from each of the first five clusters. The origin RCV1 is a dataset contains $810,000$ documents. In order to obtain a smaller dataset, we choose samples from only four categories: "C15", "ECAT", "GCAT" and "MCAT", with 300 documents from each category. Our text data preprocessing steps include 1) removing stop words; 2) applying stemming to the remaining words; 3) applying tf-idf transformation; 4) applying the l_2-norm normalization on document; 5) applying bi-normalization to the data matrix as in [8].

Baselines and Evaluation Metric. We choose four state-of-the-art competitors to show the outperformance of our proposed NRFS: Laplacian Score (LS) [15]; Spectral Feature selection (SPEC) [43]; Multi-Cluster Feature Selection (MCFS) [7]; and Nonnegative Discriminative Feature Selection (NDFS) [28].

It would be the best to evaluate feature selection results based on ground truth of feature importance. However, in real world application, we cannot easily find such ground truth because: 1) it is highly subjective to select candidate features because there are many similar features/terms, and 2) feature selection is an intermediate step for the rest of data analysis pipeline. However, even though we don't have the ground truth for feature importance, we do have the ground truth of cluster labels to indirectly evaluate the quality of feature selection, by comparing clustering performance of the feature-reduced dataset.

In our experiment, we evaluate the feature selection algorithms by performing K-means clustering on the selected feature space. To give a more general perspective, we also test K-means clustering (WCSS [14]) without any feature selection. Normalized Mutual Information (NMI) is used as our only evaluation metric among all being described because most of clustering algorithm papers make use of NMI as their primary evaluation metric. The detailed definition of NMI can be found in [37].

Parameters. The number of selected features are set as $\{$ 200, 300, 500, 800, 1000, 1200, 1500, 1800 $\}$. For the similarity function used in the microarray dataset experiments, we use Gaussian similarity (Equation 1). We need to construct similarity matrices with both local feature subspace and the whole feature space. Here we adopt an adaptive width of neighborhood σ for each local feature subspace, instead of a fixed value. In our implementation, we assign σ to be the average Euclidean distance of each instance to its K' nearest neighbor, where K' is the average size of clusters ($K' = round(n/c)$). For text datasets, cosine similarity (Equation 2) is a reasonable choice to compare texts with different sizes. For all the kNN based similarity methods $k = 5$, where k specifies the size of neighborhood. The number of eigenvectors c is set as the number of instance clusters, which assume to be already known [32] [7].



(a) Microarray dataset: 11Tumors

(b) Microarray dataset: Leukemia2

(c) Microarray dataset: BrainTumor2

(d) Microarray dataset: Lung

(e) Text dataset: RCV1-4Classes

(f) Text dataset: Reuter21578A

(g) Text dataset: 20NewsgroupA

(h) Text dataset: 20NewsgroupB

Fig. 5. Comparison of feature selection performance. Results are evaluate by K-means clustering on the selected feature subset using NMI score. It shows that our proposed NRFS (in red) outperforms the other competitors.

Especially, for MCFS, we keep $min\{M, n\}$ non-zero entries in each eigenvector when trying to select M features. For NDFS, we set $\alpha = 1e-006$, $\beta = 1e-006$ and $\gamma = 10^8$. We follow the suggestions in [7] [28] to set default values for these parameters.

Our proposed algorithm NRFS has two specific parameters: the sampling rate p and the number of neighbors q for each feature. We set p according to DPS [4] with $level = 2$ and pick one out of four sampling subsets), and $q = 50$ which is appropriate for maintaining stable performance and alleviating noise effects adaptively. We also test the performance stability of NRFS across different size of feature subspace q later.

To guarantee a fair comparison, for each size of feature subsets, we run every algorithm 30 times and record the average NMI in Figure 5. Whenever we get the reduced feature subspace, we apply the K-means clustering (the version with minimizing within-cluster sum of square (WCSS) [14]), with 100 inner loops and 100 outer loops.

B. Overall Algorithm Performance Analysis

Figure 5 documents the performance of a few feature selection algorithms, including our proposed NRFS and K-means clustering on the whole feature space. The experiments are measured by NMI derived from the K-means clustering on feature subspaces generated by the feature selection algorithms. The experimental results offer the following observations:

(1) Generally speaking, NRFS results on text datasets showed an "improving" trend as the feature size increases, i.e. NRFS started with a suboptimal performance for text datasets when the size of feature subset is small (eg. 200, 300), and surpassed the other algorithms when the size increases. The reason is that the number of informative features/words in text datasets is usually much higher (e.g. hundreds) than those in microarray datasets (e.g. dozens). For microarray dataset, a small number of informative features contain sufficient information to achieve a good clustering quality. However, for text datasets, if a feature subset is too small, it cannot provide enough descriptive capability to differentiate different document categories.

(2) Second, feature selection algorithms help to obtain a refined description of the feature space. Compared with the K-means clustering on the whole feature space, most of the five feature selection algorithms have better performance in their reduced feature space. In particular, our proposed NRFS, has more than 25% for the microarray datasets and $180\% \sim 200\%$ improvement for the text datasets in average.

(3) Our proposed NRFS outperforms not only the similarity-based methods such as LS and SPEC, but also regression-based methods such as MCFS and NDFS in terms of average NMI. Moreover, NRFS shows more stable performance as the number of feature change. Our NRFS outperforms MCFS, the second best algorithm, by a margin of more than 10% for microarray datasets and 25% for text datasets in average. It confirms that our proposed NRFS algorithm is capable to find better representative feature subsets by detecting and taking advantage of multi-perspective correlation.

(4) MCFS [7] and NDFS [28], to some extent, are capable to exploit discriminative information among different features, which result in more accurate result than LS [15] and SPEC [43].

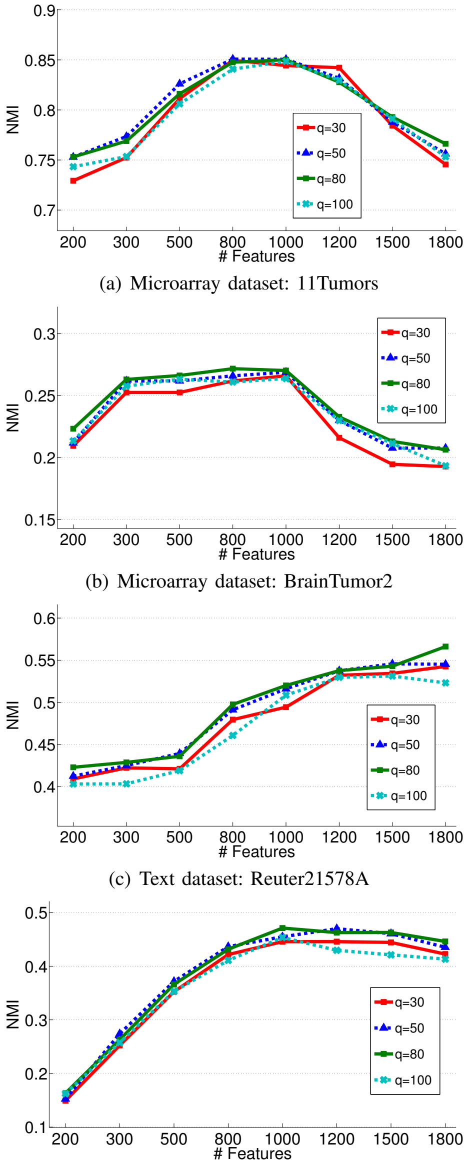
We conduct experiments with controlled size of feature neighborhood q to examine the NRFS's stability. The datasets used in the experiments are 11Tumors, BrainTumor2, Reuter21578A and 20NewsgroupB with $q = [30, 50, 80, 100]$. As shown in Figure 6, our proposed NRFS consistently shows a robust performance across different q.

C. Comparison of Time Complexity

Figure 7 shows the comparison results of time complexity among the six algorithms including two versions of NRFS: NRFS-1 is with noise-resistant and density-preserving sampling, while NRFS-2 uses the full instance space without representative selection. With the help of our sampling strategy, NRFS-1 is 57% faster than NRFS-2. Moreover, NRFS-1 has comparable running time with MCFS, but it is more than 2.5 times faster than NDFS. Although SPEC and LS are more efficient, their effectiveness shown in Figure 5 is actually much worse than our proposed NRFS.

VII. Conclusion

This paper has proposed an unsupervised feature selection algorithm called Noise-Resistant Feature Selection (NRFS). It has two main advantages: firstly, NRFS is a collaborative feature selection algorithm based on multi-perspective correlation, in that it probes the feature effect via the local perspective of representative instances and global spectrums, and thereby effectively distinguishes diverse and yet informative features from the remaining ones. Secondly, NRFS applies noise-resistant and density-preserving sampling to improve its efficiency while reducing the negative affect incurred by noisy instances. Compared with existing algorithms, our proposed NRFS demonstrates much more stable and better performance in the experiments on microarray and text datasets.



(a) Microarray dataset: 11Tumors

(b) Microarray dataset: BrainTumor2

(c) Text dataset: Reuter21578A

(d) Text dataset: 20NewsgroupB

Fig. 6. Performance stability of NRFS across different size of feature neighborhood q.

VIII. Acknowledgements

We gratefully thank all the anonymous reviewers for constructive suggestions toward paper improvement. This research

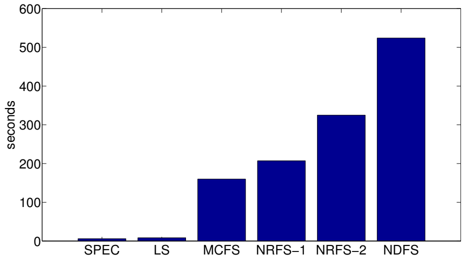


Fig. 7. Comparison of time complexity. NRFS-1 is NRFS with our sampling strategy, while NRFS-2 is NRFS without any sampling.

is supported in part by National Science Foundation of USA (No. IIS-0949467, IIS-1047715, and IIS-1049448), and National Natural Science Foundation of China (No. 61190120, 61190121, and 61190125). It is also supported by United States Departmentof Energy, Grant No. DE-SC0003361, funded through the American Recovery and Reinvestment Act of 2009, and BSA/DOE Prime Contract (DE-AC02-98CH10886) to Brookhaven National Laboratory.

References

[1] C. C. Aggarwal, J. Han, J. Wang, and P. S. Yu. A framework for projected clustering of high dimensional data streams.

[2] D. Barbara, C. Domeniconi, and J. P. Rogers. Detecting outliers using transduction and statistical testing. *SIGKDD*, pages 55–64, 2006.

[3] B. Bolstad. Probe level quantile normalization of high density oligonucleotide array data. *Unpublished manuscript*, 2001.

[4] M. Budka and G. Bogdan. Density-preserving sampling: Robust and efficient alternative to cross-validation for error estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013.

[5] H. Cho and I. S. Dhillon. Coclustering of human cancer microarrays using minimum sum-squared residue coclustering. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 5(3):385–400, 2008.

[6] P. Courrieu. Fast computation of moore-penrose inverse matrices. *Nueural Information Processing-Letters and Review*.

[7] C. Deng, C. Zhang, and X. He. Unsupervised feature selection for multi-cluster data. *SIGKDD*, pages 333–342, 2010.

[8] I. S. Dhillon. Co-clustering documents and words using bipartite spectral graph partitioning. *SIGKDD*.

[9] J. G. Dy. Unsupervised feature selection. *Computational Methods of Feature Selection*, pages 19–39, 2008.

[10] J. G. Dy and C. E. Brodley. Feature selection for unsupervised learning. *The Journal of Machine Learning Research*.

[11] B. Efron and R. Tibshirani. An introduction to the bootstrap. *Chapman Hall*, 1993.

[12] V. Garcia, E. Debreuve, and M. Barlaud. Fast k nearest neighbor search using gpu. *Computer Vision and Pattern Recognition Workshops*, 2008.

[13] A. Goh and R. Vidal. Segmenting motions of different types by unsupervised manifold clustering. *CVPR*.

[14] J. A. Hartigan and M. A. Wong. Algorithm as 136: a k-means clustering algorithm. *Applied Statistics*, 28(1):100–108, 1978.

[15] X. He, D. Cai, and P. Niyogi. Laplacian score for feature selection. *Advances in Neural Information Processing Systems*, 2006.

[16] C. Hou, F. Nie, D. Yi, and Y. Wu. Feature selection via joint embedding learning and sparse regression. *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence*, pages 1324–1329, 2011.

[17] H. Huang, H. Qin, S. Yoo, and D. Yu. Local anomaly descriptor: a robust unsupervised algorithm for anomaly detection based on diffusion space. *CIKM*, pages 405–414, 2012.

[18] H. Huang, H. Qin, S. Yoo, and D. Yu. A new anomaly detection algorithm based on quantum mechanics. *ICDM*, 2012.

[19] W. Hung, M. Yang, and D. Chen. Bootstrapping approach to feature-weight selection in fuzzy c-means algorithms with an application in color image segmentation. *Pattern Recognition Letters*.

[20] Y. Jiang and J. Ren. Eigenvalue sensitive feature selection. *Proceedings of the International Conference on Machine Learning*, 2011.

[21] R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation andmodel selection. *IJCAI*.

[22] Y. Koren and R. Bell. Advances in collaborative filtering. *Recommender Systems Handbook*.

[23] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*.

[24] H. P. Kriegel, P. Kroger, and A. Zimek. Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering. *TKDD*.

[25] Y. Li, M. Dong, and J. Hua. Localized feature selection for clustering. *Pattern Recognition Letters*, pages 10–18, 2008.

[26] Y. Li, M. Dong, and J. Hua. Simultaneous localized feature selection and model detection for gaussian mixtures. *Pattern Analysis and Machine Intelligence*, pages 953–960, 2009.

[27] Z. Li, J. Liu, S. Chen, and X. Tang. Noise robust spectral clustering. pages 1–8, 2007.

[28] Z. Li, Y. Yang, J. Liu, X. Zhou, and H. Lu. Unsupervised feature selection using nonnegative spectral analysis. *AAAI*, 2012.

[29] H. Liu and L. Yu. Toward integrating feature selection algorithms for classification and clustering. *IEEE Knowledge and Data Engineering*, 17(4):491–502, 2005.

[30] U. V. Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

[31] P. Mitra, C. A. Murthy, and Sankar K. Pal. Unsupervised feature selection using feature similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3).

[32] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. *Advances in Neural Information Processing Systems*, 14:846–856, 2002.

[33] S. K. Pal and P. Mitra. Pattern recognition algorithms for data mining. *Chapman and Hall/CRC*, 2004.

[34] H. Peng, F. Long, and C. Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *Pattern Analysis and Machine Intelligence*, 27(8):1226–1238, 2010.

[35] Y. Saeys, I. Inza, and P. Larranaga. A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23(19):2507–2517, 2007.

[36] A. Statnikov, C. F. Aliferis, I. Tsamardinos, D. Hardin, and S. Levy. Gene expression model selector. *http://www.gems-system.org/*, 2005.

[37] A. Strehl and J. Ghosh. Cluster ensembles ł a knowledge reuse framework for combining multiple partitions. *J. Mach. Learn. Res.*, pages 583–617, 2003.

[38] X. Wang, Y. Wang, and L. Wang. Improving fuzzy c-means clustering based on feature-weight learning. *Pattern Recognition Letters*.

[39] J. Weston, A. Elisseeff, B. Scholkopf, and M. Tipping. Use of the zero norm with linear models and kernel methods. *The Journal of Machine Learning Research*, 2003.

[40] D. M. Witten and R. Tibshirani. A framework for feature selection in clustering. *Journal of the American Statistical Association*, pages 713–726, 2010.

[41] Y. Yang, H. T. Shen, Z. Ma, Z. Huang, and X. Zhou. l2,1-norm regularized discriminative feature selection for unsupervised learning. *IJCAI*, pages 1589–1594, 2011.

[42] Y. Yue, C. Wang, K. El-Arini, and C. Guestrin. Personalized collaborative clustering. *WWW*.

[43] Z. Zhao and H. Liu. Spectral feature selection for supervised and unsupervised learning. *Proceedings of the 24th international conference on Machine learning*.

[44] X. Zhu, Z. Huang, Y. Yang, H. T. Shen, C. Xu, and J. Luo. Self-taught dimensionality reduction on the high-dimensional small-sized data. *Pattern Recognition*, 2012.