Discriminant Analysis for Unsupervised Feature Selection

Jiliang Tang* Xia Hu* Huiji Gao* Huan Liu*

Abstract

Feature selection has been proven to be efficient in preparing high dimensional data for data mining and machine learning. As most data is unlabeled, unsupervised feature selection has attracted more and more attention in recent years. Discriminant analysis has been proven to be a powerful technique to select discriminative features for supervised feature selection. To apply discriminant analysis, we usually need label information which is absent for unlabeled data. This gap makes it challenging to apply discriminant analysis for unsupervised feature selection. In this paper, we investigate how to exploit discriminant analysis in unsupervised scenarios to select discriminative features. We introduce the concept of pseudo labels, which enable discriminant analysis on unlabeled data, propose a novel unsupervised feature selection framework DisUFS which incorporates learning discriminative features with generating pseudo labels, and develop an effective algorithm for DisUFS. Experimental results on different types of real-world data demonstrate the effectiveness of the proposed framework DisUFS.

1 Introduction

High-dimensional data is common in many real-world applications such as data mining, machine learning, computer vision and image processing. Data with high dimensionality not only significantly increases the time and memory requirements of algorithms, but also can degenerate their performance due to the curse of dimensionality and the existence of irrelevant dimensions [1]. Feature selection, selecting a subset of most relevant features for a compact and accurate presentation, is proven to be an effective and efficient way to handle high-dimensional data [2, 3, 1].

Based on whether the training data is labeled or not, feature selection methods are broadly divided into supervised methods and unsupervised methods. Supervised feature selection selects features with the capability to distinguish samples from different classes [4, 5, 6, 7, 8]. As most data is unlabeled, unsupervised feature selection attracts increasing attention in recent years [9, 10, 11]. For unsupervised feature selection, the definition of relevance of features becomes unclear due to the lack of label information [12, 13, 14], hence, it is a less constrained problem and particularly difficult [10, 3, 15, 16].

Without label information to define feature relevance, a number of alternative criteria have been proposed for unsupervised feature selection such as data variance, data similarity [10, 15, 17] and data separability [18, 14]. However, most existing criteria for unsupervised feature selection neglects the discriminative information of features, which has been demonstrated to be important in data analysis [16]. Discriminant analysis plays a crucial role in supervised feature selection [19]. It aims to select discriminative features such that within-class distance is as small as possible while between-class distance is as large as possible [5]. Previous studies showed that algorithms based on discriminant analysis such as Fisherscore [5] and Linear Discriminant Feature Selection [20] can select discriminative features for classification and are the state-of-the-art supervised feature selection algorithms [15, 21, 22]. The work of discriminant analysis is often associated with the availability of label information, which is unavailable for unlabeled data. This gap makes performing discriminant analysis on unlabeled data challenging.

In this paper, we investigate how to employ discriminant analysis for unsupervised feature selection to select discriminative features and propose a novel feature selection framework DisUFS which can select a set of discriminative features simultaneously for unlabeled data. Our contributions are summarized as,

- Introducing the concept of pseudo labels to enable us to perform discriminant analysis on unlabeled data;
- Proposing an unsupervised feature selection framework DisUFS which combines learning discriminative features and generating pseudo-labels;
- Developing an efficient algorithm to address the optimization problem of DisUFS; and
- Evaluating the proposed framework DisUFS systematically on various types of real-world datasets to understand the working of DisUFS.

The rest of this paper is organized as follows. Our unsupervised feature selection framework based on discriminant analysis DisUFS is introduced in Section 2. In Section 3, an alternating optimization method is developed to optimize the proposed framework DisUFS. Empirical evaluation

*Computer Science and Engineering, Arizona State University, Tempe, AZ. {jiliang.tang, xia.hu, huiji.gao, huan.liu}@asu.edu

is presented in Section 4 with discussion. In Section 5, we briefly review related work. The conclusion and future work are presented in Section 6.

2 The Proposed Unsupervised Feature Selection Framework

Let $\mathcal{F} = \{f_1, f_2, \ldots, f_m\}$ be the set of features where m is the number of features and $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n] \in \mathbb{R}^{m\times n}$ be a set of unlabeled data where n is the number of data points. For simplicity, we assume that the data points in $\mathbf{X}$ are centered, that is, $\sum_{i=1}^{n} \mathbf{x}_i = 0$, which can be realized as $\mathbf{X} = \mathbf{XP}$ where $\mathbf{P} = \mathbf{I}_n - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^\top$.

To address the challenges presented by the lack of label information for discriminant analysis, we introduce the concept of pseudo labels. In detail, we assume that these n unlabeled data points can be assigned with k pseudo labels $\mathcal{C} = \{c_1, c_2, \ldots, c_k\}$. Let $\mathbf{H} \in \mathbb{R}^{n\times k}$ be pseudo labels indicator matrix where $\mathbf{H}(i,j) = 1$ if $\mathbf{x}_i$ is assigned the j-th label, zero otherwise. We further define $\mathbf{Y}$ as the weighted pseudo label indicator matrix, which can be obtained from $\mathbf{H}$ as below,

$$\mathbf{Y} = \mathbf{H}(\mathbf{H}^\top\mathbf{H})^{-\frac{1}{2}},$$

where the i-th column of $\mathbf{Y}$ is given by

$$\mathbf{Y}(i,:) = \pi(\overbrace{0,\ldots,0}^{n-h_i},\overbrace{1,\ldots,1}^{h_i})/\sqrt{h_i}, \tag{2.1}$$

where h_i is the number of data points with the i-th pseudo label c_i and $\pi(\cdot)$ is the permutation function.

With pseudo labels, we can apply discriminant analysis to unlabeled data. Denote $\mu_i = \sum_{\mathbf{x}\in c_i}\mathbf{x}/h_i$ as the mean vector of the i-th pseudo label c_i. We first define within-cluster scatter, between-cluster scatter and total scatter matrices based on pseudo labels as

$$\begin{aligned}
\mathbf{S}_w &= \sum_{i=1}^{k}\sum_{\mathbf{x}_j\in c_i}(\mathbf{x}_j-\mu_i)(\mathbf{x}_j-\mu_i)^\top,\\
\mathbf{S}_b &= \sum_{i=1}^{k} h_i\mu_i\mu_i^\top = \mathbf{XYY}^\top\mathbf{X}^\top,\\
\mathbf{S}_t &= \mathbf{XX}^\top,
\end{aligned}$$

where $Tr(\mathbf{S}_w)$ captures the within-cluster distance, and $Tr(\mathbf{S}_b)$ captures the between-cluster distance. It is easy to verify that

$$\mathbf{S}_t = \mathbf{S}_w + \mathbf{S}_b. \tag{2.2}$$

Then linear discriminant analysis aims to obtain a linear transformation $\mathbf{W} \in \mathbb{R}^{m\times b}$ that projects $\mathbf{X}$ from m-dimensional space to d-dimensional space ($b < m$) such that the within-cluster distance is minimized while the between-cluster distance is maximized as

$$\max Tr\big((\mathbf{W}^\top\mathbf{S}_t\mathbf{W})^{-1}\mathbf{W}^\top\mathbf{S}_b\mathbf{W}\big). \tag{2.3}$$

The introduction of pseudo labels allows us to perform discriminant analysis on unlabeled data as in Eq. (2.3) and then our unsupervised feature selection framework DisUFS based on discriminant analysis can be formulated as the following optimization problem,

$$\begin{aligned}
\max_{\mathbf{W},\mathbf{Y}} & Tr\big((\mathbf{W}^\top\mathbf{XX}^\top\mathbf{W})^{-1}\mathbf{W}^\top\mathbf{XYY}^\top\mathbf{X}^\top\mathbf{W}\big)\\
& - \alpha\|\mathbf{W}\|_{2,1},\\
s.t.\ & \mathbf{y}_i = \pi(\overbrace{0,\ldots,0}^{n-h_i},\overbrace{1,\ldots,1}^{h_i})/\sqrt{h_i},\\
& \sum_{i=1}^{k} h_i = n
\end{aligned} \tag{2.4}$$

where $\|\mathbf{W}\|_{2,1}$ is the $\ell_{2,1}$-norm of $\mathbf{W}$, which is defined as follows:

$$\|\mathbf{W}\|_{2,1} = \sum_{i=1}^{m}\sqrt{\sum_{j=1}^{k}\mathbf{W}^2(i,j)} = \sum_{i=1}^{m}\|\mathbf{W}(i,:)\|_2. \tag{2.5}$$

The term $\|\mathbf{W}\|_{2,1}$ in Eq. (2.4) is introduced to control the capacity of $\mathbf{W}$ and also ensure that $\mathbf{W}$ is sparse in rows. Since each row of $\mathbf{W}$ corresponds to a feature in $\mathcal{F}$, the term $\|\mathbf{W}\|_{2,1}$ makes $\mathbf{W}$ particularly suitable for feature selection. In detail, we can rank features $f_i|_{i=1}^m$ according to $\|\mathbf{W}(i,:)\|_2|_{i=1}^m$ in descending order and select top-K ranked features where K is the number of features we want to select. The parameter α is introduced to control the sparsity of $\mathbf{W}$.

The significance of the introduction of pseudo labels to the proposed framework DisUFS is two-fold. First, with pseudo labels, we can perform discriminant analysis on unlabeled data. Second, with pseudo labels, we can do unsupervised feature selection in an supervised manner. However, the introduction of pseudo labels also brings about new challenges to optimize Eq. (2.4). In the following section, we will introduce an optimization algorithm to seek an optimal solution for Eq. (2.4).

3 An Optimization Method for DisUFS

The optimization problem of DisUFS mixes $\ell_{2,1}$-norm optimization on $\mathbf{W}$ with integer programming on $\mathbf{Y}$ and it is difficult to address $\ell_{2,1}$-norm optimization and integer programming simultaneously. We note that $\ell_{2,1}$-norm optimization on $\mathbf{W}$ and integer programming on $\mathbf{Y}$ are decoupled if we optimize $\mathbf{W}$ and $\mathbf{Y}$ separately. This observation motivates us to adopt an alternating optimization to solve this problem, which works well for a number of practical optimization problems [23]. Under this scheme, we update $\mathbf{W}$ and $\mathbf{Y}$ in an alternating manner.

3.1 Given $\mathbf{Y}$, Computing $\mathbf{W}$ When $\mathbf{Y}$ is fixed, $\mathbf{W}$ is obtained by the following $\ell_{2,1}$-norm optimization problem,

$$\max_{\mathbf{W}} Tr\big((\mathbf{W}^\top\mathbf{X}\mathbf{X}^\top\mathbf{W})^{-1}\mathbf{W}^\top\mathbf{X}\mathbf{Y}\mathbf{Y}^\top\mathbf{X}^\top\mathbf{W}\big) - \alpha\|\mathbf{W}\|_{2,1}. \tag{3.6}$$

Recently there have been many methods proposed to solve the $\ell_{2,1}$-norm optimization problem [24, 8, 25]. However, the problem in Eq. (3.6) is different from existing ones due to the term of $(\mathbf{W}^\top\mathbf{X}\mathbf{X}^\top\mathbf{W})^{-1}$. Directly solving Eq. (3.6) is difficult; thus, we will introduce an algorithm to solve Eq. (3.6) indirectly with the following theorem.

THEOREM 3.1. *Maximizing*

$$Tr\big((\mathbf{W}^\top\mathbf{X}\mathbf{X}^\top\mathbf{W})^{-1}\mathbf{W}^\top\mathbf{X}\mathbf{Y}\mathbf{Y}^\top\mathbf{X}^\top\mathbf{W}\big) \tag{3.7}$$

is equivalent to minimizing the following problem:

$$\|\mathbf{X}^\top\mathbf{B} - \mathbf{G}\|_F^2, \tag{3.8}$$

under the condition: rank($\mathbf{S}_t$) = rank($\mathbf{S}_b$) + rank($\mathbf{S}_w$), where $\mathbf{B} \in \mathbb{R}^{m\times k}$ and $\mathbf{G}$ is a special pseudo label indicator matrix as follows:

$$\mathbf{G}(i,k) = \begin{cases} \sqrt{\frac{n}{h_k}} - \sqrt{\frac{h_k}{n}} & \text{if } \mathbf{x}_i \in c_k, \\ -\sqrt{\frac{h_k}{n}} & \text{otherwise} \end{cases} \tag{3.9}$$

In addition, the optimal solution of Eq. (3.7) $\mathbf{W}$ and the optimal solution of Eq. (3.8) $\mathbf{B}$ have the following relation,

$$\mathbf{B}\mathbf{Q} = [\mathbf{W}, \mathbf{0}], \tag{3.10}$$

where $\mathbf{Q}$ is a orthogonal matrix, i.e., $\mathbf{Q}^\top\mathbf{Q} = \mathbf{Q}\mathbf{Q}^\top = \mathbf{I}$.

Proof. The detailed proof process is similar to that of equivalence between linear discriminant analysis and multi-label least square in [26]. Note that the condition in Theorem 3.1 is usually satisfied for high-dimensional data [26], which is usually the case in feature selection problem. □

Next we will find an equivalent formulation for Eq. (3.6) with the help of Theorem 3.1, and we begin with the following lemma.

Lemma 3.1 If $\mathbf{B}\mathbf{Q} = [\mathbf{W}, \mathbf{0}]$ and $\mathbf{Q}$ is an orthogonal matrix, then

$$\|\mathbf{B}(i,:)\|_2 = \|\mathbf{W}(i,:)\|_2. \tag{3.11}$$

Proof. Setting $\mathbf{C} = \mathbf{B}\mathbf{Q}$ and $\mathbf{D} = [\mathbf{W}, \mathbf{0}]$, we have

$$\begin{aligned}\|\mathbf{C}(i,:)\|_2 &= \|\mathbf{B}(i,:)\mathbf{Q}\|_2 \\ &= \sqrt{\mathbf{B}(i,:)\mathbf{Q}\mathbf{Q}^\top\mathbf{B}^\top(i,:)} = \|\mathbf{B}(i,:)\|_2,\end{aligned}$$

and $\|\mathbf{D}(i,:)\|_2 = \|\mathbf{W}(i,:)\|_2$. Since $\|\mathbf{C}(i,:)\|_2 = \|\mathbf{D}(i,:)\|_2$, we can obtain that $\|\mathbf{B}(i,:)\|_2 = \|\mathbf{W}(i,:)\|_2$, which completes the proof. □

Lemma 3.1 indicates that the $\ell_{2,1}$-norm of $\mathbf{B}$ is equal to that of $\mathbf{W}$. With Theorem 3.1 and Lemma 3.1, the maximization problem in Eq. (3.6) is equivalent to the following minimization problem:

$$\min_{\mathbf{B}} \|\mathbf{X}^\top\mathbf{B} - \mathbf{G}\|_F^2 + \alpha\|\mathbf{B}\|_{2,1}, \tag{3.12}$$

where the optimal solutions of $\mathbf{B}$ and $\mathbf{W}$ have the relation $\mathbf{B}\mathbf{Q} = [\mathbf{W}, \mathbf{0}]$.

The $\ell_{2,1}$-norm minimization problem in Eq. (3.12) is well studied [24, 8, 25] and in this paper, we adopt the optimization method in [8, 25] to obtain $\mathbf{B}$ as shown in the following theorem.

THEOREM 3.2. *$\mathbf{B}$ in Eq. (3.12) can be updated by*

$$\mathbf{B} \to (\mathbf{X}\mathbf{X}^\top + \alpha\Omega)^{-1}\mathbf{X}\mathbf{G}, \tag{3.13}$$

which can monotonically reduce the objective value. Ω is a diagonal matrix and its i-th diagonal element is defined as

$$\Omega(i,i) = \frac{1}{2\|\mathbf{B}(i,:)\|_2}. \tag{3.14}$$

Proof. Using $\mathcal{L}_W$ to denote the objective function of Eq. (3.12), we take the derivative of $\mathcal{L}_W$,

$$\frac{\partial\mathcal{L}_W}{\partial\mathbf{W}} = 2\mathbf{X}\mathbf{X}^\top\mathbf{B} - 2\mathbf{X}\mathbf{G} + 2\alpha\Omega\mathbf{B}. \tag{3.15}$$

$\mathbf{X}\mathbf{X}^\top$ is a semi-positive definite matrix and therefore $\mathbf{X}\mathbf{X}^\top + \alpha\Omega$ is a positive definite matrix. Setting the derivative to zero, we can obtain the update rule in Eq. (3.16),

$$\mathbf{B} = (\mathbf{X}\mathbf{X}^\top + \alpha\Omega)^{-1}\mathbf{X}\mathbf{G}, \tag{3.16}$$

Similar to [8, 25], we can prove that the update rule in Eq. (3.13) monotonically reduces the objective value of Eq. (3.12), which completes the proof.

According to Theorem 3.2, we can obtain an optimal solution of $\mathbf{B}$. However, we still do not know the optimal solution of $\mathbf{W}$ from $\mathbf{B}$ since we do not know the specific form of the orthogonal matrix $\mathbf{Q}$, which is difficult to obtain [26]. Actually $\mathbf{W}$ plays two roles in our framework - selecting features and computing $\mathbf{Y}$. Lemma 3.1 indicates that the $\ell_{2,1}$-norm of $\mathbf{B}$ is equal to that of $\mathbf{W}$. Therefore, $\mathbf{B}$ can replace $\mathbf{W}$ in terms of selecting features. If $\mathbf{B}$ can also replace $\mathbf{W}$ to compute $\mathbf{Y}$, we can use $\mathbf{B}$ to replace $\mathbf{W}$ in our framework, which significantly reduces the difficulty of optimizing our framework. In the following subsection, we will demonstrate that $\mathbf{B}$ can also replace $\mathbf{W}$ to compute pseudo label matrix $\mathbf{Y}$.

3.2 Given W, Computing Y When $\mathbf{W}$ is fixed, $\mathbf{Y}$ is obtained via solving the following problem,

$$\begin{aligned}\max_{\mathbf{Y}} & Tr\big((\mathbf{W}^\top\mathbf{X}\mathbf{X}^\top\mathbf{W})^{-1}\mathbf{W}^\top\mathbf{X}\mathbf{Y}\mathbf{Y}^\top\mathbf{X}^\top\mathbf{W}\big)\\ s.t.\ & \mathbf{y}_i = \pi(\overbrace{0,\ldots,0}^{n-h_i},\overbrace{1,\ldots,1}^{h_i})/\sqrt{h_i},\\ & \sum_{i=1}^{k} h_i = n\end{aligned} \tag{3.17}$$

Next we will show that $\mathbf{B}$ can replace $\mathbf{W}$ in Eq. (3.17) to compute $\mathbf{Y}$ with the following lemma.

Lemma 3.2 If $\mathbf{B}\mathbf{Q} = [\mathbf{W}, \mathbf{0}]$ and $\mathbf{Q}$ is a orthogonal matrix, then

$$Tr\big((\mathbf{W}^\top\mathbf{X}\mathbf{X}^\top\mathbf{W})^{-1}\mathbf{W}^\top\mathbf{X}\mathbf{Y}\mathbf{Y}^\top\mathbf{X}^\top\mathbf{W}\big)$$

is equivalent to

$$Tr\big((\mathbf{B}^\top\mathbf{X}\mathbf{X}^\top\mathbf{B})^{-1}\mathbf{B}^\top\mathbf{X}\mathbf{Y}\mathbf{Y}^\top\mathbf{X}^\top\mathbf{B}\big)$$

Proof. It is easy to verify that

$$\begin{aligned}&Tr\big((\mathbf{W}^\top\mathbf{X}\mathbf{X}^\top\mathbf{W})^{-1}\mathbf{W}^\top\mathbf{X}\mathbf{Y}\mathbf{Y}^\top\mathbf{X}^\top\mathbf{W}\big)\\ &= Tr\big(((\mathbf{B}\mathbf{Q})^\top\mathbf{X}\mathbf{X}^\top\mathbf{B}\mathbf{Q})^{-1}(\mathbf{B}\mathbf{Q})^\top\mathbf{X}\mathbf{Y}\mathbf{Y}^\top\mathbf{X}^\top\mathbf{B}\mathbf{Q}\big),\end{aligned}$$

we have

$$\begin{aligned}((\mathbf{B}\mathbf{Q})^\top\mathbf{X}\mathbf{X}^\top\mathbf{B}\mathbf{Q})^{-1} &= (\mathbf{Q}^\top\mathbf{B}^\top\mathbf{X}\mathbf{X}^\top\mathbf{B}\mathbf{Q})^{-1}\\ &= \mathbf{Q}^{-1}(\mathbf{B}^\top\mathbf{X}\mathbf{X}^\top\mathbf{B})^{-1}(\mathbf{Q}^\top)^{-1}.\end{aligned} \tag{3.18}$$

Since $\mathbf{Q}$ is a orthogonal matrix, we have,

$$\mathbf{Q}^{-1} = \mathbf{Q}^\top,\ \ (\mathbf{Q}^\top)^{-1} = \mathbf{Q}. \tag{3.19}$$

With Eq. (3.18) and Eq. (3.19), we can obtain,

$$\begin{aligned}&Tr\big(((\mathbf{B}\mathbf{Q})^\top\mathbf{X}\mathbf{X}^\top\mathbf{B}\mathbf{Q})^{-1}(\mathbf{B}\mathbf{Q})^\top\mathbf{X}\mathbf{Y}\mathbf{Y}^\top\mathbf{X}^\top\mathbf{B}\mathbf{Q}\big)\\ &= Tr\big(\mathbf{Q}^\top(\mathbf{B}^\top\mathbf{X}\mathbf{X}^\top\mathbf{B})\mathbf{Q}(\mathbf{B}\mathbf{Q})^\top\mathbf{X}\mathbf{Y}\mathbf{Y}^\top\mathbf{X}^\top\mathbf{B}\mathbf{Q}\big)\\ &= Tr\big((\mathbf{B}^\top\mathbf{X}\mathbf{X}^\top\mathbf{B})\mathbf{B}^\top\mathbf{X}\mathbf{Y}\mathbf{Y}^\top\mathbf{X}^\top\mathbf{B}\big)\end{aligned} \tag{3.20}$$

which completes the proof. □

Lemma 3.2 indicates that $\mathbf{B}$ can replace $\mathbf{W}$ to compute $\mathbf{Y}$. Therefore, Eq. (3.17) can be rewritten as the following optimization problem,

$$\begin{aligned}\max_{\mathbf{Y}} & Tr\big((\mathbf{B}^\top\mathbf{X}\mathbf{X}^\top\mathbf{B})^{-1}\mathbf{B}^\top\mathbf{X}\mathbf{Y}\mathbf{Y}^\top\mathbf{X}^\top\mathbf{B}\big)\\ s.t.\ & \mathbf{y}_i = \pi(\overbrace{0,\ldots,0}^{n-h_i},\overbrace{1,\ldots,1}^{h_i})/\sqrt{h_i},\\ & \sum_{i=1}^{k} h_i = n\end{aligned} \tag{3.21}$$

where $\mathbf{B}$ is learnt by Theorem 3.2.

We develop the following theorem to solve the integer programming problem in Eq. (3.21).

THEOREM 3.3. *The optimal $\mathbf{Y}$ can be computed by solving a kernel K-means problem with $\mathbf{X}^\top\mathbf{B}(\mathbf{B}^\top\mathbf{X}\mathbf{X}^\top\mathbf{B})^{-1}\mathbf{B}^\top\mathbf{X}$ as the kernel Gram matrix.*

Proof. Since $Tr(\mathbf{A}\mathbf{C}) = Tr(\mathbf{C}\mathbf{A})$ for any two matrices $\mathbf{A}$ and $\mathbf{C}$, Eq. (3.21) can be reformed as

$$\begin{aligned}\max_{\mathbf{Y}} & Tr\big(\mathbf{Y}^\top\mathbf{X}^\top\mathbf{B}(\mathbf{B}^\top\mathbf{X}\mathbf{X}^\top\mathbf{B})^{-1}\mathbf{B}^\top\mathbf{X}\mathbf{Y}\big)\\ s.t.\ & \mathbf{y}_i = \pi(\overbrace{0,\ldots,0}^{n-h_i},\overbrace{1,\ldots,1}^{h_i})/\sqrt{h_i},\\ & \sum_{i=1}^{k} h_i = n\end{aligned} \tag{3.22}$$

It is easy to verify that $\mathbf{X}^\top\mathbf{B}(\mathbf{B}^\top\mathbf{X}\mathbf{X}^\top\mathbf{B})^{-1}\mathbf{B}^\top\mathbf{X}$ is a semi-definite matrix, which can be a kernel Gram matrix. According to [27, 28, 29], the optimal $\mathbf{Y}$ can be obtained via solving a kernel K-means problem with $\mathbf{X}^\top\mathbf{B}(\mathbf{B}^\top\mathbf{X}\mathbf{X}^\top\mathbf{B})^{-1}\mathbf{B}^\top\mathbf{X}$ as the kernel Gram matrix, which completes the proof □

3.3 The Proposed Algorithm Lemma 3.1 indicates that the $\ell_{2,1}$-norm of $\mathbf{B}$ is equal to that of $\mathbf{W}$, while Lemma 3.2 indicates that $\mathbf{B}$ can replace $\mathbf{W}$ to compute $\mathbf{Y}$. These two lemmas suggest that $\mathbf{B}$ can replace $\mathbf{W}$ for DisUFS. Instead of computing $\mathbf{W}$ and $\mathbf{Y}$, it is much easier to compute $\mathbf{B}$ and $\mathbf{Y}$. We develop an alternating optimization method for DisUFS via alternatively solving the following two optimization problems for $\mathbf{B}$ and $\mathbf{Y}$, respectively.

- The optimization problem for $\mathbf{B}$ is

$$\min_{\mathbf{B}} \|\mathbf{X}^\top\mathbf{B} - \mathbf{G}\|_F^2 + \alpha\|\mathbf{B}\|_{2,1}. \tag{3.23}$$

- The optimization problem for $\mathbf{Y}$ is

$$\begin{aligned}\max_{\mathbf{Y}} & Tr\big(\mathbf{Y}^\top\mathbf{X}^\top\mathbf{B}(\mathbf{B}^\top\mathbf{X}\mathbf{X}^\top\mathbf{B})^{-1}\mathbf{B}^\top\mathbf{X}\mathbf{Y}\big)\\ s.t.\ & \mathbf{y}_i = \pi(\overbrace{0,\ldots,0}^{n-h_i},\overbrace{1,\ldots,1}^{h_i})/\sqrt{h_i},\\ & \sum_{i=1}^{k} h_i = n\end{aligned} \tag{3.24}$$

Theorem 3.2 and Theorem 3.3 provide updating rules for $\mathbf{B}$ and $\mathbf{Y}$ respectively, and the detailed optimization algorithm for DisUFS is presented in Algorithm 1.

We briefly review Algorithm 1. In line 4, we construct $\mathbf{G}$ from $\mathbf{Y}$ according to Eq. (3.9). Based on Theorem 3.2, we update $\mathbf{B}$ in line 5 and construct Ω in line 6. In line 7, we update $\mathbf{Y}$ according to Theorem 3.3. Originally the importance of the i-th feature is indicated by $\|\mathbf{W}(i,:)\|_2$. However, with Lemma 3.1, we have $\|\mathbf{B}(i,:)\|_2 = \|\mathbf{W}(i,:)\|_2$. Therefore, in line 9, we rank features in descending order according to $\|\mathbf{B}(i,:)\|_2$.

Algorithm 1 The Proposed Unsupervised Feature Selection Framework - DisUFS

Input: $\mathbf{X}$, the number of pseudo labels k, α, and the number of features to select K

Output: K most relevant features

1: Initialize $\mathbf{Y}$ via performing k-means on $\mathbf{X}$
2: Initialize Ω as an identity matrix
3: **while** Not convergent **do**
4: Construct $\mathbf{G}$ from $\mathbf{Y}$
5: Update $\mathbf{B}$: $\mathbf{B} \leftarrow (\mathbf{X}\mathbf{X}^\top + \alpha\Omega)^{-1}\mathbf{X}\mathbf{G}$
6: Update the diagonal matrix Ω, where the i-th diagonal element is $\frac{1}{2\|\mathbf{B}(i,:)\|_2}$
7: Update $\mathbf{Y}$ via solving a kernel K-means problem with $\mathbf{X}^\top\mathbf{B}(\mathbf{B}^\top\mathbf{X}\mathbf{X}^\top\mathbf{B})^{-1}\mathbf{B}^\top\mathbf{X}$ as the kernel Gram matrix
8: **end while**
9: Sort each feature according to $\|\mathbf{B}(i,:)\|_2$ in **descending** order and select the top-K ranked ones;

Time Complexity : The most time-consuming operations for Algorithm 1 are to update $\mathbf{B}$ in the line 5 and $\mathbf{Y}$ in the line 7.

- $\mathbf{B} \leftarrow (\mathbf{X}\mathbf{X}^\top + \alpha\Omega)^{-1}\mathbf{X}\mathbf{G}$ can be efficiently obtained by solving the linear equation $(\mathbf{X}\mathbf{X}^\top + \alpha\Omega)\mathbf{B} = \mathbf{X}\mathbf{G}$, which needs $O(km^2)$.
- To obtain $\mathbf{Y}$, we need to solve a kernel K-means problem with $\mathbf{X}^\top\mathbf{B}(\mathbf{B}^\top\mathbf{X}\mathbf{X}^\top\mathbf{B})^{-1}\mathbf{B}^\top\mathbf{X}$ as the kernel Gram matrix, which takes $O(kmn + km^2)$.

In summary, the total time complexity of Algorithm 1 is $\#iterations * O(km(m+n))$.

4 Experiments

In this section, we conduct experiments to evaluate the effectiveness of DisUFS. After introducing experimental settings, we compare DisUFS with the state-of-the-art unsupervised feature selection methods. Further experiments are designed to investigate the effects of parameters (the number of pseudo labels k and α) on DisUFS.

4.1 Experimental Settings We choose four benchmark data sets of different types, e.g. image data (PIX10P and PIE10P) and microarray data (CLL-SUB-111 and TOX-171), to test the performance of unsupervised feature selection[1]. Some statistics of these datasets are shown in Table 1.

Following the common way to evaluate unsupervised feature selection algorithms, we assess DisUFS in terms of clustering performance [15, 16]. In detail, we first apply unsupervised feature selection algorithms to select features and then perform k-means with the selected features. Since k-means often converges to local minima, we repeat each experiment 10 times and report the average performance.

Table 1: Statistics of the Data Sets

Datasets	Size	# of Features	# of Classes
PIX10P	100	10,000	10
PIE10P	210	2,420	10
CLL-SUB-111	111	11,340	3
TOX-171	171	5,748	4

Two commonly used metrics, *accuracy* and *normalized mutual information* (NMI), are employed to evaluate the quality of clusters[2]. How to determine the optimal number of selected features is still an open problem [25] thus we vary the numbers of selected features as $\{20, 50, 70, 100, 120, 150, 170, 200, 250, 300\}$.

4.2 Quality of Selected Features We compare DisUFS with the following three representative unsupervised feature selection algorithms:

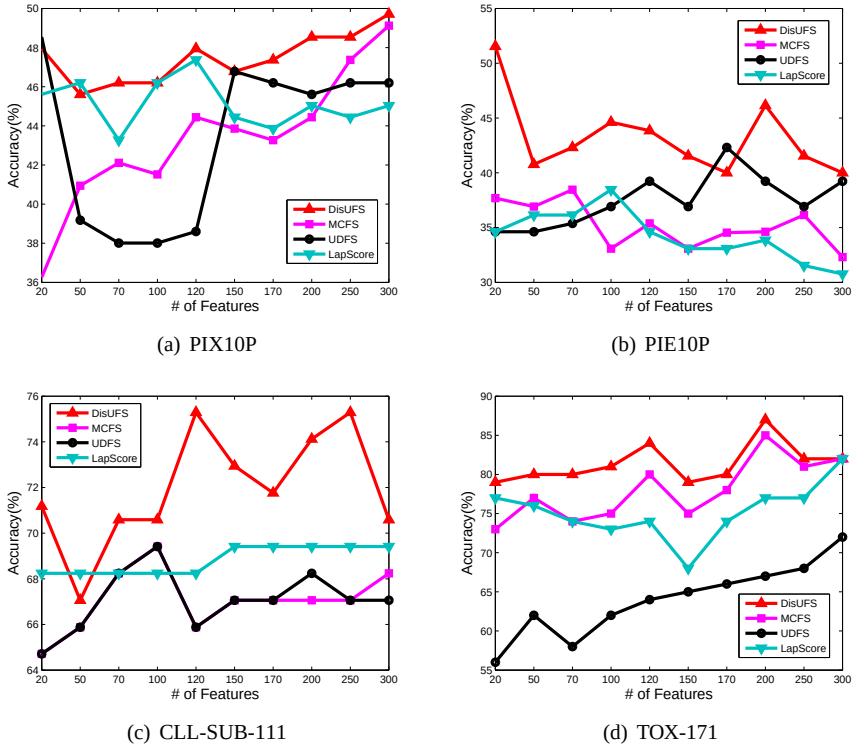
- UDFS [16] selects features in batch mode by simultaneously exploiting local discriminative information and feature correlation;
- MCFS [14] selects features using spectral regression with ℓ_1-norm regularization;
- Laplacian Score [10] evaluates the importance of a feature through its power of locality preservation.

MCFS and Laplacian Score are not based on discriminate analysis, while UDFS and DisUFS are based on discriminate analysis. The major differences between UDFS and DisUFS are two-fold. First, UDFS exploits the local discriminative information while DisUFS performs global discriminant analysis with the help of pseudo labels. Second, the optimization problems for UDFS and DisUFS are very different. The parameters in all methods are determined via cross-validation. For DisUFS, we set the number of pseudo labels k to 20 in PIX10P and PIE10P, while we set k to 25 in CLL-SUB-11 and TOX-171. More details about the effects of parameters on DisUFS will be discussed in the later subsections. The comparison results in terms of accuracy and NMI are shown in Figures 1 and 2, respectively. We make the following observations,

- With the increase of the number of selected features, clustering performance trends to first increase and then degrade. LapScore, MCFS and UDFS obtain comparable results on all data sets.

[1] These data sets are publicly available from http:// featureselection.asu.edu/datasets.php.

[2] We use the source code from http://www.zjucadcg.cn/ dengcai/Data/Clustering.html.



(a) PIX10P (b) PIE10P

(c) CLL-SUB-111 (d) TOX-171

Figure 1: Comparison of Different Unsupervised Feature Selection Algorithms w.r.t. Accuracy.

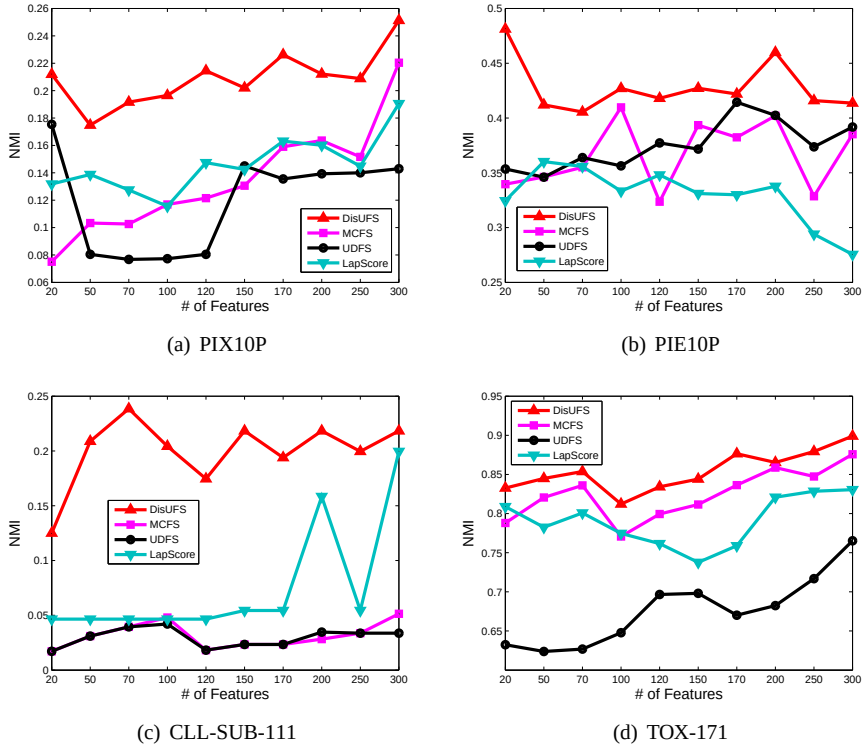
- DisUFS consistently obtains better performance than UDFS. We perform a t-test on the results of DisUFS and UDFS, and the testing results suggest that the improvement of DisUFS over UDFS is significant. These results indicate that global discriminant analysis with pseudo labels is more likely to select discriminative features than local discriminant analysis.
- Most of the time, DisUFS outperforms all the baseline methods. Compared to the best performance of baseline methods, on average DisUFS obtains 7.78% relative improvement in terms of accuracy. Similarly, a t-test is performed to investigate the significance and all results suggest the improvement of DisUFS is significant. These results further demonstrate the capability of discriminant analysis for unsupervised feature selection with pseudo labels.

In summary, DisUS performing discriminant analysis with pseudo labels can improve unsupervised feature selection performance in terms of clustering. In the following subsections, we will investigate the impact of the number of pseudo labels on DisUFS in detail.

4.3 Impact of Numbers of Pseudo labels In this subsection, we investigate the effect of the number of pseudo labels (k) on the proposed framework DisUFS. We vary the numbers of pseudo labels as $\{3, 5, 7, 10, 15, 20, 25, 30, 35, 40\}$. The performance variation with respect to k and the number of selected features is depicted in Figure 3. Note that we only show the performance in terms of accuracy since we have similar observations in terms of NMI.

Most of the time, with the increase of k, the performance first increases gradually, reaches its peak value and then degrades. When k is too small, pseudo labels cannot fully capture the cluster structure of the data, while DisUFS will overfit the data with a large number of pseudo labels. This pattern can be used to determine the optimal value of k. We also note that the best performance is achieved when the number of pseudo labels is larger than the actual number of classes. For each dataset, the performance is not sensitive to k when k is in a certain region such as k is from 10 to 30 in CLL-SUB-11.

4.4 Impact of α The parameter α, controlling the row sparsity of $\mathbf{W}$, plays an important role in DisUFS for feature selection. We investigate the effect of α by analyzing how changes of α affect the performance of DisUFS. We vary the value of α as $\{0.1, 0.5, 1, 10, 100, 500, 1e3, 5e3, 7e3, 1e4\}$. The performance variance w.r.t. α and the numbers of selected features is demonstrated in Figure 4. We only



(a) PIX10P (b) PIE10P

(c) CLL-SUB-111 (d) TOX-171

Figure 2: Comparison of Different Unsupervised Feature Selection Algorithms in terms of NMI.

show the results in terms of accuracy since we have similar observations in terms of NMI.

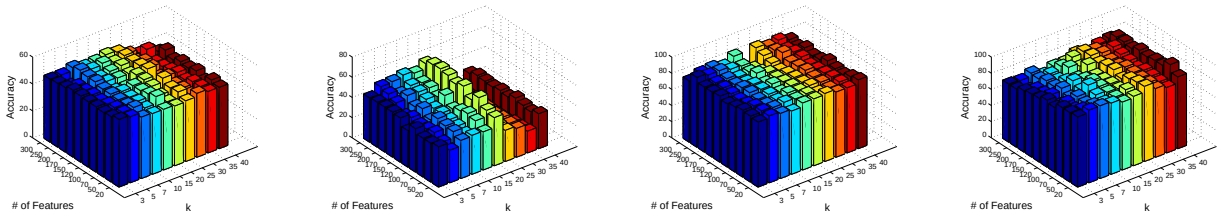
With the increase of α, we observe a similar pattern to that with the increase of the number of pseudo labels k - performance first increases and then decreases, demonstrating the capability of $\ell_{2,1}$-norm in feature selection. We also note that between α and the number of selected features, DisUFS is more sensitive to the number of selected features.

5 Related Work

Feature selection can be roughly categorized into supervised or unsupervised methods based on the training data being labeled or unlabeled [3, 30]. Supervised methods can be further divided into filter models [4, 7] and wrapper models [31, 32]. Filter models separates feature selection from classifier learning and the bias of a learning algorithm does not interact with the bias of a feature selection algorithm [4], while wrapper models adopt the performance of a predetermined learning algorithms to assess the quality of selected features and can be egregiously expensive to run for data with a large number of features [5, 6]. Since most data in the real-world is unlabeled, more and more attention is paid on unsupervised feature selection [9, 10, 11]. Without class labels, unsupervised feature selection [10, 3, 15] is a less constrained search problem and depending on clustering quality measures [12, 13], and can eventuate many equally valid feature subsets. With high-dimensional data, it is likely to find many sets of features that seem equally good without considering additional constraints.

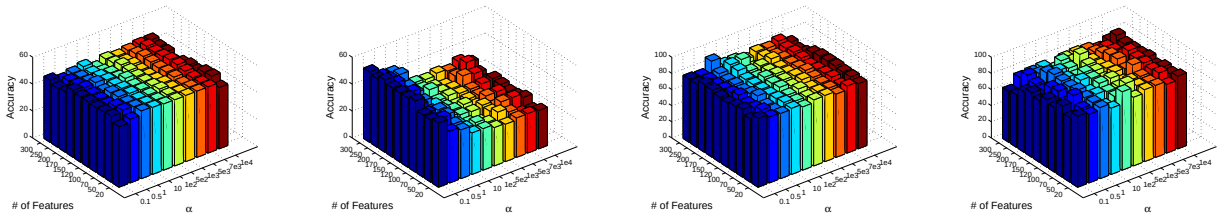
With label information, discriminant analysis is broadly adopted by supervised feature selection methods. Fisher score is one of the most popular methods in this family. Its key idea is to find a subset of features such that with the new representation, the distances between instances in the same class are as small as possible, while the distances between instances in different classes are as large as possible [5]. Sparsity regularization, such as the $\ell_{2,1}$-norm of a matrix [33], has been widely investigated and applied to feature selection [34, 24, 8, 35] in dimensionality reduction. Discriminant analysis with spare learning attracts increasing attention in supervised feature selection. In [20], the authors proposed a sparse linear discriminant feature selection framework (LDFS), which is equivalent to solve the following problem,

$$\max_{\mathbf{W}} Tr((\mathbf{W}^\top \mathbf{S}_t \mathbf{W})^{-1}(\mathbf{W}^\top \mathbf{S}_b \mathbf{W})) - \mu \sum_{i=1}^{b} \|\mathbf{W}(i,:)\|_\infty$$



(a) ACC in PIX10P (b) ACC in PIE10P (c) ACC in CLL-SUB-111 (d) ACC in TOX-171

Figure 3: Number of Features vs Number of Pseudo labels k



(a) ACC in PIX10P (b) ACC in PIE10P (c) ACC in CLL-SUB-111 (d) ACC in TOX-171

Figure 4: Number of Features vs α

where $\sum_{i=1}^{b} \|\mathbf{W}(i,:)\|_\infty$ is the ℓ_1/ℓ_∞ norm of $\mathbf{W}$. The structured sparse transformation matrix $\mathbf{W}$ allows LDFS to achieve feature selection. In [22], an alternative formulation based on discriminant analysis is proposed, which is equivalent to solve the following problem,,

$$\min_{\mathbf{W}} \frac{1}{2}\|\mathbf{X}^\top \mathbf{W} - \mathbf{H}\|_F^2 + \mu\|\mathbf{W}\|_{2,1}$$

where $\mathbf{H}$ is a weighted class label matrix, and details about the definition of $\mathbf{H}$ can be found in [22].

To apply discriminant analysis, we usually need label information which is absent for unlabeled data. This gap makes it challenging to apply discriminant analysis for unsupervised feature selection. In [16], Local discriminative analysis is performed for unsupervised feature selection UDFS and it defines local discriminative score to evaluate the levels of within class scatter and between-class scatter in a local manner, which is substantially different from our proposed framework DisUFS - (1) DisUFS performs global discriminant analysis with pseudo-labels, while UDFS makes use of local discriminant analysis; and (2) DisUFS combines generating pseudo-labels and selecting discriminative features into a coherent framework, while UDFS assumes the class label of input instances can be predicted by a linear classifier and predefines a linear classifier.

6 Conclusion

Discriminant analysis is widely adopted to select discriminative features for supervised feature selection. Due to the lack of label information, it is much more difficult to perform discriminant analysis for unsupervised feature selection. In this paper, we propose a novel unsupervised feature selection framework DisUFS which can select a set of discriminative features simultaneously. To tackle the difficulty presented by the lack of label information, we introduce the concept of pseudo-labels, which allows us to perform discriminant analysis on unlabeled data. We combine learning discriminative features and generating pseudo-labels into a coherent framework. The optimization problem for DisUFS mixes $\ell_{2,1}$-norm optimization with integer programming and we develop an alternating optimization method for DisUFS. Experiments are conducted on various types of real-world datasets and the results show that our proposed framework outperforms the state-of-the-art unsupervised feature selection methods.

There are several interesting directions to investigate in the future. First, the proposed optimization algorithm can only find a local optimal solution for DisUFS and we will study optimization algorithms to seek a global solution for DisUFS. Second, we would like to seek a method to determine parameters of DisUFS automatically.

Acknowledgments

We thank the anonymous reviewers for their useful comments. The work is, in part, supported by NSF (#IIS-1217466).

References

[1] H. Liu and H. Motoda, *Computational methods of feature selection*. Chapman & Hall, 2008.

[2] G. John, R. Kohavi, and K. Pfleger, "Irrelevant feature and the subset selection problem," in *ICML*, 1994.

[3] J. Dy and C. Brodley, "Feature selection for unsupervised learning." *Journal of Machine Learning Research*, 2004.

[4] M. Hall, "Correlation-based feature selection for discrete and numeric class machine learning," in *ICML*, 2000.

[5] R. Duda, P. Hart, D. Stork *et al.*, *Pattern classification*. wiley New York, 2001, vol. 2.

[6] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Machine learning*, 2002.

[7] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy," *TPAMI*, 2005.

[8] F. Nie, H. Huang, X. Cai, and C. Ding, "Efficient and robust feature selection via joint l21-norms minimization." NIPS, 2010.

[9] L. Wolf and A. Shashua, "Feature selection for unsupervised and supervised inference: the emergence of sparsity in a weighted-based approach," *Journal of Machine Learning Research*, 2005.

[10] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," *NIPS*,2006.

[11] C. Constantinopoulos, M. Titsias, and A. Likas, "Bayesian feature and model selection for gaussian mixture models," *TPAMI*, 2006.

[12] J. G. Dy, C. E. Brodley, A. C. Kak, L. S. Broderick, and A. M. Aisen, "Unsupervised feature selection applied to content-based retrieval of lung images," *TPAMI*, 2003.

[13] J. G. Dy and C. E. Brodley, "Visualization and interactive feature selection for unsupervised data," in *KDD*, 2000.

[14] D. Cai, C. Zhang, and X. He, "Unsupervised feature selection for multi-cluster data," in *KDD*,2010.

[15] Z. Zhao and H. Liu, "Spectral feature selection for supervised and unsupervised learning," in *ICML*, 2007.

[16] Y. Yang, H. Shen, Z. Ma, Z. Huang, and X. Zhou, "L21-norm regularized discriminative feature selection for unsupervised learning," in *IJCAI*, 2011.

[17] F. Nie, S. Xiang, Y. Jia, C. Zhang, and S. Yan, "Trace ratio criterion for feature selection," in *AAAI*, vol. 2, 2008, pp. 671–676.

[18] Y. Li, M. Dong, and J. Hua, "Localized feature selection for clustering," *Pattern Recognition Letters*, vol. 29, no. 1, pp. 10–18, 2008.

[19] K. Fukunaga, *Introduction to statistical pattern recognition*. Academic Pr, 1990.

[20] M. Masaeli, G. Fung, and J. Dy, "From transformation-based dimensionality reduction to feature selection," in *Int. Conf. on Machine Learning*, 2010.

[21] Z. Zhao, L. Wang, and H. Liu, "Efficient spectral feature selection with minimum redundancy," in *Proceedings of the Twenty-4th AAAI Conference on Artificial Intelligence (AAAI)*, 2010.

[22] Q. Gu, Z. Li, and J. Han, "Generalized fisher score for feature selection," *arXiv preprint arXiv:1202.3725*, 2012.

[23] J. Bezdek and R. Hathaway, "Some notes on alternating optimization," *Advances in Soft Computing‡AFSS 2002*, pp. 187–195, 2002.

[24] J. Liu, S. Ji, and J. Ye, "Multi-task feature learning via efficient l 2, 1-norm minimization," in *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*. AUAI Press, 2009, pp. 339–348.

[25] J. Tang and H. Liu, "Feature selection with linked data in social media," in *SIAM International Conference on Data Mining*, 2012.

[26] J. Ye, S. Ji, and J. Chen, "Learning the kernel matrix in discriminant analysis via quadratically constrained quadratic programming," in *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2007, pp. 854–863.

[27] I. Dhillon, Y. Guan, and B. Kulis, "A unified view of kernel k-means, spectral clustering and graph cuts," in *UTCS Technical Report # TR-04-25*, 2004.

[28] C. Ding and T. Li, "Adaptive dimension reduction using discriminant analysis and k-means clustering," in *ICML*, 2007.

[29] J. Ye, Z. Zhao, and M. Wu, "Discriminative k-means for clustering." Advances in Neural Information Processing Systems, 2007.

[30] H. Liu and L. Yu, "Toward integrating feature selection algorithms for classification and clustering," *IEEE Trans. on Knowledge and Data Engineering*, vol. 17, no. 3, pp. 1–12, 2005.

[31] J. G. Dy and C. E. Brodley, "Feature subset selection and order identification for unsupervised learning," in *Proceedings of the Seventeenth International Conference on Machine Learning*, 2000, pp. 247–254.

[32] Y. Kim, W. Street, and F. Menczer, "Feature selection for unsupervised learning via evolutionary search," in *KDD*, 2000, pp. 365–369.

[33] C. Ding, D. Zhou, X. He, and H. Zha, "R 1-pca: rotational invariant l 1-norm principal component analysis for robust subspace factorization," in *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006, pp. 281–288.

[34] A. Argyriou, T. Evgeniou, and M. Pontil, "Multi-task feature learning," *Advances in Neural Information Processing Systems*, vol. 19, p. 41, 2007.

[35] J. Tang and H. Liu, "Unsupervised feature selection for linked social media data," in *ACM SIGKDD international conference on Knowledge discovery and data mining*, 2012.