# Adaptive Dimension Reduction Using Discriminant Analysis and $K$-means Clustering

**Chris Ding** CHQDING@LBL.GOV
Lawrence Berkeley National Laboratory, University of California, Berkeley, CA 94720

**Tao Li** TAOILI@CS.FIU.EDU
School of Computer Science, Florida International University, Miami, FL 33199

## Abstract

We combine linear discriminant analysis (LDA) and $K$-means clustering into a coherent framework to adaptively select the most discriminative subspace. We use $K$-means clustering to generate class labels and use LDA to do subspace selection. The clustering process is thus integrated with the subspace selection process and the data are then simultaneously clustered while the feature subspaces are selected. We show the rich structure of the general LDA-Km framework by examining its variants and their relationships to earlier approaches. Extensive experimental results on real-world datasets show the effectiveness of our approach.

## 1. Introduction

In many application domains, such as information retrieval, computational biology, and image processing, the data dimension is usually very high. Developing effective clustering methods for high dimensional datasets is a challenging problem due to the *curse of dimensionality*. It has been shown that in a high dimensional space, the distance between every pair of points is almost the same for a wide variety of data distributions and distance functions (Beyer et al., 1999).

There are many approaches to address this problem. The simplest approach is dimension reduction techniques, including principal component analysis (PCA) (Duda et al., 2000; Jolliffe, 2002) and random projections (Dasgupta, 2000). In these methods, dimension reduction is carried out as a preprocessing step and is decoupled from the clustering process: once the subspace dimensions are selected, they stay fixed during the clustering process. An extension of this approach is the adaptive dimension reduction approach (Ding et al., 2002; Li et al., 2004) where the subspace is adaptively adjusted and integrated with the clustering process.

A different approach is called subspace clustering (see a survey (Parsons et al., 2004)) where the focus is on selecting a small number of original dimensions (features) in some unsupervised way so that clusters become more obvious in this subspace. Focusing on the original features (dimensions) has the advantage of easy implementation on a database. However, the rigidity of original dimension do not have enough flexibility to handle clusters which extends along a mixture of directions.

Subspace clustering and adaptive dimension reduction attempt to find the subspace where clusters are most well-separated or well defined in some way. This is different from co-clustering (simultaneously clustering the features and data points) (Dhillon, 2001; Zha et al., 2001; Banerjee et al., 2004) and biclustering (Cheng & Church, 2000) (which essentially find blocks in a rectangle data matrix).

If we restrict the subspace to be linear combinations of original features, the subspace obtained in linear discriminant analysis (LDA) is perhaps the best subspace to do data clustering, because in LDA subspace, clusters are well separated. LDA is a very well developed theory (Hastie et al., 2001), and is getting renewed interest (De la Torre & Kanade, 2006; Ye & Xiong, 2006; Park & Howland, 2004) with the growth of matrix-based approaches in machine learning. In (De la Torre & Kanade, 2006), a matrix factorization is proposed that, after one matrix factor is eliminated, the two remaining matrix factors can be viewed as the projection directions in a LDA variant and cluster indicators respectively. They are solved in an alternative fashion using LDA and a soft-clustering (see §5.3), similar to adaptive dimension reduction.

In this paper, we further develop the adaptive dimension reduction approach by explicitly combining LDA and $K$-means clustering in a coherent way. Our contributions are the following: (a) We show that LDA and $K$-means cluster-

Appearing in *Proceedings of the 24$^{th}$ International Conference on Machine Learning*, Corvallis, OR, 2007. Copyright 2007 by the author(s)/owner(s).

ing are optimizing the same objective function, i.e., they both minimize the within-class scatter matrix and maximize the between-class scatter matrix. (b) Based on the above theoretical analysis, we show that the objective function for LDA provides a natural generalization which combine LDA and $K$-means clustering together. in the $K$-means clustering. (c) This adaptive dimension reduction optimization problem is then solved by standard and well-established LDA and $K$-means clustering algorithms. (d) We show that this new approach reduces to earlier approaches under various restrictions. Overall, our new approach provides a generalization that has a solid theoretical foundation, extremely clear and simple to implement, and recover earlier approaches as special cases.

Our adaptive dimension reduction approach can be intuitively viewed as an unsupervised LDA. We use $K$-means clustering to generate class labels and use LDA to do subspace selection. The clustering process is thus integrated with the subspace selection process and the data are then simultaneously clustered while the feature subspaces are selected. We make effective use of cluster membership as the bridge connecting the clusters discovered in the subspace and those defined in the full space. With this connection, clusters are discovered in the low dimensional subspace to avoid the curse of dimensionality, while the subspace are adaptively re-adjusted for global optimality.

The rest of the paper is organized as follows: Section 2 present theoretical analysis and introduces the LDA-guided K-means clustering; Section 3 proposes the LDA-Km learning framework by combining LDA and K-means clustering; Section 4 examines two variants of the LDA-Km algorithm; Section 5 explores the relations of the LDA-Km framework to other earlier approaches; Section 6 presents our experimental results; and finally Section 7 concludes.

## 2. LDA and K-means

Consider a set of input data vectors $X = (\mathbf{x}_1, \cdots, \mathbf{x}_n)$ in high dimensional space. For simplicity, the data is centered in the preprocessing step, so that $\bar{\mathbf{x}} = \sum_i \mathbf{x}_i/n = 0$. The standard $K$-means clustering is to minimize the clustering objective function

$$\min_H J_K,\; J_K = \sum_k \sum_{i\in C_k} ||\mathbf{x}_i - \mathbf{m}_k||^2 \tag{1}$$

where the matrix $H = \{0,1\}^{n\times K}$ is the cluster indicator: $H_{ik} = 1$ if $\mathbf{x}_i$ belongs to the $k$-th cluster, and $H_{ik} = 0$ otherwise. We use Tr $M$ to denote the trace of matrix $M$.

In linear discriminant analysis (LDA), we use the total scatter, between-class scatter and within-class scatter matrices:

$$S_t = \sum_{i=1}^n \mathbf{x}_i\mathbf{x}_i^T,\; S_b = \sum_k n_k \mathbf{m}_k \mathbf{m}_k^T, \tag{2}$$

$$S_w = \sum_k \sum_{i\in C_k} (\mathbf{x}_i - \mathbf{m}_k)(\mathbf{x}_i - \mathbf{m}_k)^T \tag{3}$$

It is well-known that $S_t = S_w + S_b$. It is clear that the K-means clustering objective function is

$$J_K = \text{Tr}\, S_w = \text{Tr}\, (S_t - S_b)$$

Therefore, $K$-means clustering minimizes the within-class scatter matrix $S_w$, or maximizes the between-class scatter matrix $S_b$ since Tr $S_t$ is a constant.

On the other hand, given class labels as specified by the indicator matrix $H$, the LDA directions $U$ are determined by

$$\max_U \text{Tr} \frac{U^T S_b U}{U^T S_w U} \tag{4}$$

which can be interpreted as

$$\min_U \text{Tr}(U^T S_w U) \quad \text{and} \quad \max_U \text{Tr}(U^T S_b U) \tag{5}$$

Indeed another LDA objective function is

$$\max_U \frac{\text{Tr} U^T S_b U}{\text{Tr} U^T S_w U}. \tag{6}$$

Thus LDA has very similar properties as $K$-means clustering: minimizing within-class scatter $S_w$ and/or maximizing between-class scatter $S_b$ .

LDA is widely used to select the subspace (feature space) which has the maximal discriminant power. However, LDA is a supervised learning method which requires we know the class label for each data point before-hand.

Since LDA and $K$-means clustering both minimizes $S_w$ and maximize $S_b$, there should be ways to combine them into a single framework. In this paper, we propose to combine them into a single framework: we use $K$-means clustering to generate class labels and use LDA to do subspace selection. The final results of this learning process is that the data are clustered while the feature subspaces are selected simultaneously.

## 3. Adaptive Subspace Selection Using LDA and $K$-means Clustering

We consider clustering in the subspace and adaptively improving the subspace selected simultaneously (Ding et al., 2002). The key motivation is that the initially chosen subspace may not be the optimal subspace, which we defined as the subspace spanned by the cluster centroids. For clustering purpose, all dimensions orthogonal to this subspace are *irrelevant*. This is because the distance computation

$$||\mathbf{x}_i - \mathbf{m}_k||^2 = \sum_{j=1}^r [\mathbf{x}_i(j) - \mathbf{m}_k(j)]^2 + \sum_{j=r+1}^p [\mathbf{x}_i(j) - \mathbf{m}_k(j)]^2$$

Assuming the cluster centroid subspace $C$ span the first $r$ dimensions. Dimensions $r+1, \cdots, p$ are orthogonal to the centroid subspace and We call them as irrelevant dimensions. Distances of components in irrelevant dimensions are independent of the clustering and is merely an additive constant. Subtracting this irrelevant constants makes the clustering more well defined: clusters are more well separated in the relevant subspace.

Our main goal is to find the most discriminative subspace in a unsupervised manner. Our framework is to optimize the LDA objective function

$$\max_{U,H} \mathrm{Tr} \frac{U^T S_b U}{U^T S_w U} \tag{7}$$

We propose an procedure that alternatively optimizes $H$ and $U$. We call this algorithm as LDA-guided adaptive subspace $K$-means clustering, or LDA-Km algorithm for short.

**LDA-Km(1)**. Solve for $H$ while fixing $U$. In this case, we solve

$$\begin{aligned} \max_H \frac{\mathrm{Tr}\, U^T S_b U}{\mathrm{Tr}\, U^T S_w U} &= \frac{\mathrm{Tr}\, U^T (S_t - S_w) U}{\mathrm{Tr}\, U^T S_w U} \\ &= \frac{\mathrm{Tr}\, U^T S_t U}{\mathrm{Tr}\, U^T S_w U} - 1. \end{aligned}$$

Since $\mathrm{Tr}\, U^T S_t U$ is independent of $H$, this leads to

$$\begin{aligned} \min_H \mathrm{Tr}\, U^T S_w U &= \mathrm{Tr} \sum_k \sum_{i \in C_k} U^T (\mathbf{x}_i - \mathbf{m}_k)(\mathbf{x}_i - \mathbf{m}_k)^T U \\ &= \sum_k \sum_{i \in C_k} ||U^T \mathbf{x}_i - U^T \mathbf{m}_k||^2 \end{aligned} \tag{8}$$

This is precisely the $K$-means clustering in the subspace $Y = U^T X$. Once the solution for $H$ are obtained, we can compute the within and between cluster scatter matrices $S_w, S_b$.

For completeness, these computation can also be done using matrix notations. Given $H$, we can obtain the cluster centroids $M = (\mathbf{m}_1, \cdots, \mathbf{m}_k)$ as $M = XH(H^T H)^{-1}$. Then

$$S_w = (X - MH^T)(X - MH^T)^T,$$

$$S_b = MH^T HM^T.$$

**LDA-Km(2)**. Solve for $U$ while fixing $H$. $U$ is given by the standard LDA procedure.

Combining LDA-Km(1) and LDA-Km(2), we see that this algorithm essentially does the following:

$$\begin{array}{cc} & \text{(Initialize U)} \\ \rightarrow & \text{(K-means in subspace)} \\ \rightarrow & \text{(LDA in original space)} \\ \rightarrow & \text{(K-means in subspace)} \\ \rightarrow & \cdots \end{array}$$

The Algorithm for solving LDA-guided adaptive subspace $K$-means clustering of can be summarized as follows:

**Algorithm 1** Adaptive LDA-guided K-means Clustering

```
Step 1:  Set $K =$ number of clusters,
         Set $d = K - 1$ the dimension of the subspace.
Step 2:  Compute PCA on $X$ to obtain initial $U$.
Step 3:  Do step LDA-Km(1) to obtain $H$.
Step 4:  Do step LDA-Km(2) to obtain $U$.
Step 5:  Go to step 3 until convergence.
```

A unique feature in this approach is switching between the subspace (for clustering) and the original space (for LDA). The cluster centroids $U^T \mathbf{m}_k$ obtained in the subspace can not uniquely projected back to the original space. In fact, there are infinite number of points in the original space can be projected onto a single point in the subspace (e.g., in 3D, all points along z-axis are projected onto the origin in x-y plane). The cluster indicator $H$ enables us to uniquely connect the two spaces. For example, we compute the centroid for cluster 1 by simply averaging the data points belonging to cluster 1 in the subspace using cluster membership $H$. With this connection, clusters are discovered in the low dimensional subspace to avoid the curse of dimensionality and are adaptively re-adjusted for global optimality.

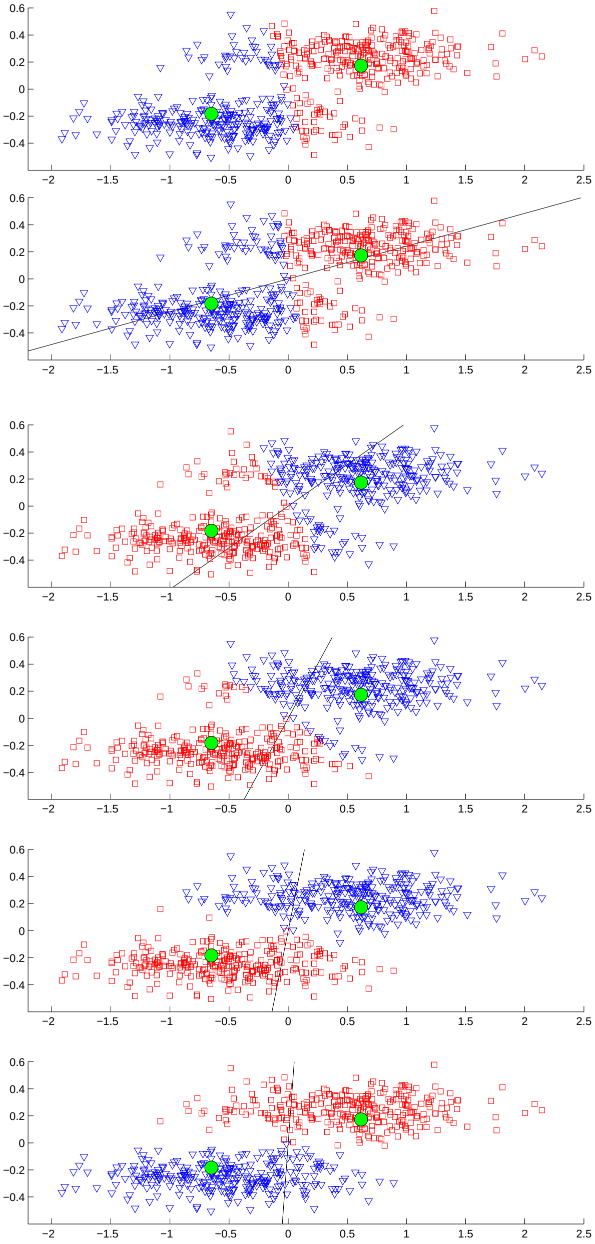
**Computational complexity**. Generally speaking, the algorithm is equivalent to $t$ (LDA + $K$-means clustering), where $t \simeq 10$ is the number of iterations of the algorithm to converge. Thus the computational complexity of the algorithm is $O(pnt)$ for $K$-means clustering and $O(p^2 nt)$ for LDA clustering where $p$ is the dimension of data, $n$ is number of data points.

### 3.1. Extension to nonlinear case using kernels

The basis idea of LDA is to transform data into a new space/subspace where clusters become most well-separated. The best linear transformation is LDA. To deal with nonlinear transformation, we turn to kernels and implement the nonlinear transformation as linear transformation. This is achieved by the mapping to a higher dimension space, much like the mapping in Support Vector Machines. It is well-known that both $K$-means clustering and LDA can be extend to nonlinear kernels. Our adaptive dimension reduction using LDA and $K$-means clustering can be similarly extended to nonlinear kernels.

### 3.2. An illustrative example

Below we give an example of 2D dataset with 600 data points. The top panel shows $K$-means clustering in the full space. The next panel shows the results of $K$-means clustering in PCA subspace. The next 4 panels show iterations of the LDA-Km algorithm, starting with $U$=PCA subspace. The line indicates the direction of $U$. One can see that the LDA-Km algorithm, starting from PCA subspace, adaptively adjusted the subspace, and converge to the correct optimal subspace.



## 4. Two Variants of the LDA-Km algorithm

In this section, we describe two variants of the LDA-Km algorithm. In step LDA-Km(2), we compute $U$ using full LDA procedure. We may consider two variants using either one of the two sub-parts of LDA in Eq.(5).

### 4.1. LDA-Km-B: Using Between-Cluster Scatter $S_b$

In this variant, we use only the between-cluster scatter $S_b$ in step LDA-Km(2). The algorithm procedure is described below:

**LDA-Km-B(1)**. Solve for $H$ while fixing $U$. Do $K$-means clustering on $Y = U^T X$.

**LDA-Km-B(2)**. Solve for $U$ while fixing $H$. $U$ is given by $d$ eigenvectors associated with the $d$ largest eigenvalues of the between-cluster scatter matrix $S_b$.

This LDA-Km-B variant of the LDA-Km algorithm can be cast in the following optimization framework

$$\max_{U,H} \mathrm{Tr} U^T S_b U \tag{9}$$

The proof is the following:

1. Given $U$, we solve for $H$ by maximizing

   $$\begin{aligned} \mathrm{Tr}\, U^T S_b U &= \mathrm{Tr}\, U^T (S_t - S_w) U \\ &= \mathrm{Tr}(U^T S U - U^T S_w U). \end{aligned}$$

   Since $U^T SU$ is constant given $U$, we minimize $\mathrm{Tr}\, U^T S_w U$, which, by Eq.(8), is exactly $K$-means clustering in the subspace $Y = U^T X$.

2. Given $H$, $U$ are given by LDA-Km-B(2).

### 4.2. LDA-Km-W: Using Within-Cluster Scatter $S_w$

In this variant, we use only the within-cluster scatter $S_w$ in step LDA-Km(2). The algorithm procedure is described below:

**LDA-Km-W(1)**. Solve for $H$ while fixing $U$. Do $K$-means clustering on $Y = U^T X$.

**LDA-Km-W(2)**. Solve for $U$ while fixing $H$. $U$ is given by $d$ eigenvectors associated with the $d$ smallest eigenvalues of the with-cluster scatter matrix $S_w$.

It is easy to see the LDA-Km-W variant of the LDA-Km algorithm [consisting of LDA-Km(1) and LDA-Km(2W)] can be cast in the following optimization framework

$$\max_{U,H} \mathrm{Tr} U^T S_w U \tag{10}$$

The proof is the following:

1. Given $U$, we solve for $H$ by minimizing Tr $U^T S_w U$ which, by Eq.(8), is the $K$-means clustering in the subspace $Y = U^T X$.

2. Given $H$, $U$ are given by LDA-Km-W(2).

## 5. Relationships to Earlier Approaches

We discuss the relation of the LDA-Km to earlier work(Ding et al., 2002; Li et al., 2004; De la Torre & Kanade, 2006).

We show that LDA-Km algorithm reduces to the *adaptive dimension reduction* (ADM) algorithm (Ding et al., 2002), where we only optimize the between-class scatter (rather than the full LDA). LDA-Km algorithm reduces to the *adaptive subspace iteration* algorithm (Li et al., 2004), where we only optimize the within-class scatter (rather than the full LDA). We also recap the discriminative cluster analysis (De la Torre & Kanade, 2006) and show it is equivalent to a variant of LDA.

### 5.1. Adaptive Dimension Reduction (ADM)

ADM begins with the observation that the PCA subspace is not necessarily the best subspace to perform data clustering and considers the best subspace to be the centroid subspace (spanned by cluster centroids). It proceeds iteratively to find the best subspace using the ADM algorithm described in Algorithm 2:

**Algorithm 2** Adaptive Dimensional Reduction (ADM) Algorithm

```
(ADM-0): Initialize the directions $U$ using PCA.
(ADM-1): Do $K$-means clustering in the subspace $Y = U^T X$.
(ADM-2): Using obtained cluster indicator $H$ to construct
         cluster centroids $C = (\mathbf{c}_1, \cdots, \mathbf{c}_K)$ in the
         original space. Do SVD of $C$: $C = U \Sigma V^T$.
         Use $U$ as the new subspace directions.
(ADM-3): Go to (ADM1) and repeat until convergence.
```

Now we show that if we replace $C$ by $\tilde{C} = (\sqrt{n_1}\ \mathbf{c}_1, \cdots, \sqrt{n_K}\ \mathbf{c}_K)$ in Step (ADM-2), ADM is identical to LDA-Km-B. This is because $U$ (the left singular vectors of $\tilde{C}$) are eigenvectors of

$$\tilde{C}\tilde{C}^T = n_1 \mathbf{c}_1 \mathbf{c}_1^T + \cdots + n_K \mathbf{c}_K \mathbf{c}_K^T = S_b.$$

Thus ADM with weighted centroid matrix $\tilde{C}$ is identical to LDA-Km-B. In general, $(\sqrt{n_1}, \cdots, \sqrt{n_K})$ do not differ very much, and so ADM is effectively equivalent to LDA-Km-B.

Experiments show that ADM can adaptively modifies the subspace to fit the data distribution; this happens when either the natural clusters in the data are close to spherical Gaussians or natural clusters are well separated. The distances between subspaces are measured by the principal angle and be efficiently computed using SVD.

However, when natural clusters in the data are far away from spherical distributions, such as the case shown in the Figure in Section 3.2, standard $K$-means clustering is no longer a good model for the data in *the full space*. ADM, starting from PCA subspace, and does not seem to converge to a subspace where the natural clusters become more separated (e.g., see the Figure in Section 3.2). In this case, starting from PCA subspace, ADM converges to a local solution which is in general close to the $K$-means solution.

Thus the challenge becomes: for datasets where natural clusters are far away from spherical Gaussians, how to modify the subspace adaptively to converge to the subspace where clusters are most separable? This subspace is clearly the LDA subspace. LDA-Km algorithm is developed along this direction. In the Figure in Section 3.2, we see LDA-Km has the ability to find the appropriate LDA subspace starting from PCA subspace.

### 5.2. Adaptive Subspace Iteration (ASI)

In ADM above, we deal explicitly with the between-cluster scatter matrix $S_b$. In ASI, we deal implicitly with the within-cluster scatter matrix $S_w$. ASI is proposed in (Li et al., 2004) to optimize the following objective function

$$\min_{C,H,U} ||U^T X - CH^T||^2 \tag{11}$$

In the initial study, $U, H$ are restricted to $\{0,1\}$, and $C$ is always set to

$$C = \arg\min_{C} ||U^T X - CH^T||^2 = U^T XH(H^T H)^{-1}.$$

$H, U$ are solved by an Iterative Feature and Data (IFD) clustering algorithm (Li & Ma, 2004).

The ASI factorization is interesting for several reasons. First, assuming $Y = U^T X, C, H$ are nonnegative. Then $Y \approx CH^T$ is a nonnegative matrix factorization (NMF). which is obtained by the optimization

$$\min_{C,H} ||Y - CH^T||^2, \quad \text{s.t.}\ C \geq 0,\ H \geq 0, H^T H = I, \tag{12}$$

By a theorem (Ding et al., 2005), the NMF of Eq.(12) is equivalent to a relaxed $K$-means clustering (Ding & He, 2004; Zha et al., 2002), the NMF of $C = (\mathbf{c}_1, \cdots, \mathbf{c}_K)$ contains the cluster centroids, and $H$ are cluster indicator. In fact, let $H = \{0,1\}$ be the cluster indicator, the $K$-means clustering of $Y = (\mathbf{y}_1, \cdots, \mathbf{y}_n)$, $J = \sum_k \sum_{i \in C_k} ||\mathbf{y}_i - \mathbf{c}_k||^2 = ||Y - CH^T||^2$ Clearly, let $U$ be the new subspace directions (the projection matrix), the data points in the new subspace are $\mathbf{y}_i = U\mathbf{x}_i$, or $Y = (\mathbf{y}_1, \cdots, \mathbf{y}_n) = UX$. Eq.(11) is just the $K$-means clustering in the subspace.

Second, we show that the objective function of ASI factorization [cf. Eq.(11) ] is identical to the objective function of LDA-Km-W [cf. Eq.(10)]. Clearly, considering Eq.(8), Eq.(10) can be written as

$$\text{Tr}\, U^T S_w U = ||U^T (X - MH^T)||^2. \tag{13}$$

Let $C = U^TM$ be the cluster centroids in the subspace, we can write Eq.(13) as Eq.(11).

### 5.3. Discriminative Cluster Analysis

In (De la Torre & Kanade, 2006), they propose to optimize

$$\min_{H,V,U} ||(HH^T)^{-1/2}(H^T - VU^TX)||^2. \tag{14}$$

using our notation $U, H$, where $V$ is a new matrix factor. After eliminate $V$, this becomes

$$\max_{H,U} \mathrm{Tr}\ (UXX^TU)^{-1}(U^TXH(HH)^{-1}H^TX^TU). \tag{15}$$

They solve this alternatively using soft clustering and a variant of LDA in a fairly involved way. This can be shown to be equivalent to

$$\max_{H,U} \mathrm{Tr}\frac{U^TS_bU}{U^TS_tU}, \tag{16}$$

which is similar to the standard LDA objective of Eq.(4). In LDA objective, the denominator is $U^TS_wU$; this "data sphering" step is a crucial step in LDA (Hastie et al., 2001). However, in Eq.(16) the denominator is $U^TS_tU$ rather than $U^TS_bU$. In other word, this variant of LDA of Eq.(16) is not a full LDA, in contrast to our LDA+*K*-means approach.

Since $S_t$ does not depends on class labels, the denominator $U^TS_tU$ can be ignore at first order approximation. The maximization of the nominator is essentially identical to the ADM of §5.1.

## 6. Experimental Results

### 6.1. Dataset Descriptions

We use a wide range of datasets in our experiments as summarized in Table 1. The number of classes ranges from 2 to 20, the number of samples ranges from 47 to 8280, and the number of dimensions ranges from 4 to 1000. In addition, these datasets represent applications from different domains such as information retrieval, gene expression data and pattern recognition. We anticipate they would provide us with enough insights on our approach.

The descriptions of these datasets are as follows.

- Eight datasets including Digits, Glass, Ionosphere, Iris, Protein, Soybean, Wine, and Zoo are from UCI data repository.
- Other datasets including CSTR, Log, Reuters, WebACE, WebKB4, WebKB are standard text datasets that has been frequently used in document clustering. We give brief descriptions of them below. The documents are represented as the term vectors using vector space model. These document datasets are pre-processed (removing the stop words and unnecessary tags and headers) using rainbow package (McCallum, 1996).
    - CSTR is the dataset of the abstracts of technical reports (TRs) published in the Department of Computer Science at a research university between 1991 and 2002. The dataset contains 476 abstracts, which are divided into four research areas: Natural Language Processing(NLP), Robotics/Vision, Systems, and Theory.
    - The Log dataset contains 1367 text messages of system log from different desktop machines describing the status of computer components. These messages are divided into 8 different situations.
    - The Reuters dataset is a subset of the Reuters-21578 Text Categorization Test collection containing the 10 most frequent categories among the 135 topics.
    - The WebACE dataset contains 2340 documents consisting of news articles from 20 different topics in October 1997 collected in WebACE project (Han et al., 1998).
    - The WebKB dataset contains webpages gathered from university computer science departments. There are about 8280 documents and they are divided into 7 categories: student, faculty, staff, course, project, department and other.
    - The WebKB4 dataset is the subset of WebKB associating with four most populous entity-representing categories, i.e., student, faculty, course and project.

| Datasets | # Samples | # Dimensions | # Class |
|---|---|---|---|
| CSTR | 475 | 1000 | 4 |
| Digits | 7494 | 16 | 10 |
| Glass | 214 | 9 | 7 |
| Ionosphere | 351 | 34 | 2 |
| Iris | 150 | 4 | 3 |
| Protein | 116 | 20 | 6 |
| Log | 1367 | 200 | 8 |
| Reuters | 2900 | 1000 | 10 |
| Soybean | 47 | 35 | 4 |
| WebACE | 2340 | 1000 | 20 |
| WebKB4 | 4199 | 1000 | 4 |
| WebKB | 8280 | 1000 | 7 |
| Wine | 178 | 13 | 3 |
| Zoo | 101 | 18 | 7 |

*Table 1.* Dataset Descriptions.

| | Kmeans | PCA+Kmeans | LDA-Km |
|---|---|---|---|
| Digits | 0.706 | 0.768 | 0.772 |
| Glass | 0.472 | 0.453 | 0.510 |
| Ionosphere | 0.710 | 0.710 | 0.712 |
| Iris | 0.893 | 0.887 | 0.980 |
| Protein | 0.483 | 0.526 | 0.595 |
| Soybean | 0.681 | 0.723 | 0.766 |
| Wine | 0.702 | 0.702 | 0.826 |
| Zoo | 0.762 | 0.792 | 0.842 |

*Table 2.* Clustering accuracy table on UCI datasets. The results are obtained by averaging 5 trials. PCA+Kmeans denotes PCA-based clustering algorithm.

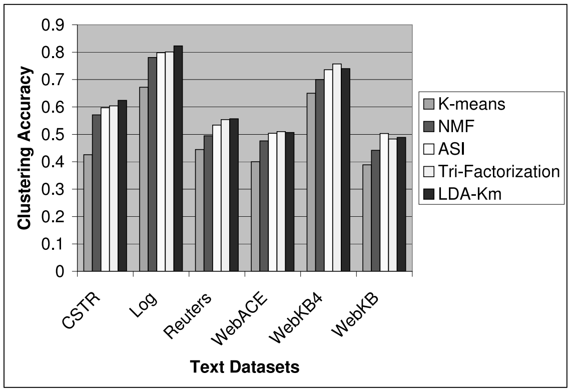


*Figure 1.* Clustering accuracy comparison on text datasets.

### 6.2. Results Analysis

All the above datasets have labels. We view the labels of the datasets as the objective knowledge on the structure of the datasets. We use accuracy as the clustering performance measure. Accuracy discovers the one-to-one relationship between clusters and classes and measures the extent to which each cluster contains data points from the corresponding class and it has been used as performance measures for clustering analysis. Accuracy can be described as:

$$Accuracy = Max(\sum_{C_k,L_m} T(C_k,L_m))/n, \qquad (17)$$

where $n$ is the number of data points, $C_k$ denotes the $k$-th cluster, and $L_m$ is the $m$-th class. $T(C_k,L_m)$ is the number of data points that belong to class $m$ are assigned to cluster $k$. Accuracy is then computed as the maximum sum of $T(C_k,L_m)$ for all pairs of clusters and classes, and these pairs have no overlaps.

On the eight datasets from UCI data repository, we compare our LDA-Km algorithm with standard K-means algorithm. We also compare it with PCA-based clustering algorithm: PCA is first applied to reduce the data dimension followed by K-means clustering. Table 2 shows the experimental results.

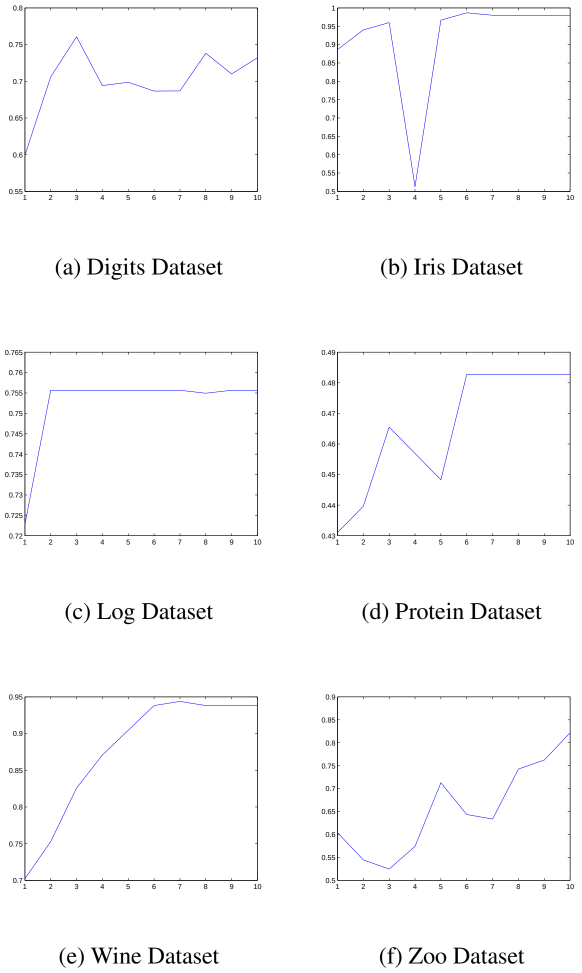
On the text datasets, we compare our subspace clustering algorithm with the following algorithms: (i) standard K-means algorithm; (ii) Non-negative Matrix Factorization (NMF) method (Lee & Seung, 2001); (iii) Tri-Factorization Method (Ding et al., 2006); and (iv) Adaptive Subspace Clustering (ASI). The results are shown in Figure 1. Note that Tri-Factorization method is based on the decomposition $X \approx FS_tG^T$ where the orthogonality of $F^TF = I, G^G = I$ is imposed to ensure $F,G$ can be viewed as cluster indicators for rows and columns. It gives a good framework for simultaneously clustering the rows and columns of $X$.

We note that on all the UCI datasets, LDA-Km clustering has the best clustering accuracies. On many datasets (e.g., Iris, Glass, Protein, soybean, wine, zoo), LDA-Km yields improvements over K-means and PCA-based clustering. On text datasets, the subspace clustering algorithm has the best accuracy on CSTR, Log and Reuters datasets. In summary, our subspace clustering is always either the winner or very close to the winner. This shows that LDA-Km clustering is viable and competitive. The subspace clustering is able to perform the subspace selection and data reduction at the same time, thus offering the capability of discovering subspace structures and yielding good clustering performances.

To get more insights on our approach, Figure 2 plots the clustering accuracy across iterations of one trial of the LDA-Km algorithm on several datasets. We observe that LDA-Km clustering is able to adaptively perform subspace section for global optimality and thus generally leads to better clustering performance. Note that LDA-Km is also able to discover clusters in the low dimensional subspace to overcome the curse of dimensionality.

## 7. Conclusions

This paper proposes LDA-Km clustering, an adaptive dimension reduction framework by combining discriminant analysis and *K*-means clustering. In the proposed framework, *K*-means clustering is used to generate class labels and linear discriminant analysis (LDA) is utilized to perform subspace selection. The clustering process is thus integrated with the subspace selection process and the learning algorithm performs data clustering and subspace selection simultaneously. We also examine two variants of the LDA-Km algorithm and discuss its relations to other earlier approaches. Encouraging experimental results are obtained showing the effectiveness of our approach.



(a) Digits Dataset

(b) Iris Dataset

(c) Log Dataset

(d) Protein Dataset

(e) Wine Dataset

(f) Zoo Dataset

*Figure 2.* Clustering accuracy evolution as a function of iterations.

## Acknowledgment

C. Ding is partially supported by the US Dept of Energy, Office of Science. T. Li is partially supported by NSF CAREER Award IIS-0546280 and NIH/NIGMS S06 GM008205. We thank Fernando De la Torre for useful discussions.

## References

Banerjee, A., Dhillon, I., Ghosh, J., Merugu, S., & Modha, D. (2004). A generalized maximum entropy approach to bregman co-clustering and matrix approximation. *KDD'04*.

Beyer, K. S., Goldstein, J., Ramakrishnan, R., & Shaft, U. (1999). When is nearest neighbor meaningful? *Proceedings of 7th International Conference on Database Theory(ICDT'99)* (pp. 217–235). Springer.

Cheng, Y., & Church, G. (2000). Biclustering of expression data. *Proc. Int'l Symp. Mol. Bio (ISMB)*, 93–103.

Dasgupta, S. (2000). Experiments with random projection. *Proc. 16th Conf. Uncertainty in Artificial Intelligence (UAI 2000)*.

De la Torre, F., & Kanade, T. (2006). Discriminative cluster analysis. *ICML'06*.

Dhillon, I. S. (2001). Co-clustering documents and words using bipartite spectral graph partitioning. *KDD'01*.

Ding, C., & He, X. (2004). K-means clustering and principal component analysis. *ICML'04*.

Ding, C., He, X., & Simon, H. (2005). On the equivalence of nonnegative matrix factorization and spectral clustering. *Proc. SIAM Data Mining Conf.*

Ding, C., He, X., Zha, H., & Simon, H. (2002). Adaptive dimension reduction for clustering high dimensional data. *Proc. IEEE Int'l Conf. Data Mining*.

Ding, C., Li, T., Peng, W., & Park, H. (2006). Orthogonal nonnegative matrix tri-factorizations for clustering. *KDD'06*, 126–135.

Duda, R. O., Hart, P. E., & Stork, D. G. (2000). *Pattern classification, 2nd ed.* Wiley.

Han, E.-H., Boley, D., Gini, M., Gross, R., Hastings, K., Karypis, G., Kumar, V., Mobasher, B., & Moore, J. (1998). WebACE: A web agent for document categorization and exploration. *Proceedings of the 2nd International Conference on Autonomous Agents (Agents'98)*.

Hastie, T., Tibshirani, R., & Friedman, J. (2001). *Elements of statistical learning*. Springer Verlag.

Jolliffe, I. (2002). *Principal component analysis*. Springer. 2nd edition.

Lee, D., & Seung, H. (2001). Algorithms for non-negative matrix factorization. *NIPS 13*.

Li, T., & Ma, S. (2004). IFD: Iterative feature and data clustering. *Pro. SIAM Int'l conf. on Data Mining (SDM 2004)* (pp. 472–476).

Li, T., Ma, S., & Ogihara, M. (2004). Document clustering via adaptive subspace iteration. *SIGIR'04* (pp. 218–225).

McCallum, A. K. (1996). Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering. http://www.cs.cmu.edu/ mccallum/bow.

Park, H., & Howland, P. (2004). Generalizing discriminant analysis using the generalized singular value decomposition. *IEEE. Trans. on Pattern Analysis and Machine Intelligence*, *26*, 995 – 1006.

Parsons, L., Haque, E., & Liu, H. (2004). Subspace clustering for high dimensional data: a review. *SIGKDD Explorations*, *6*, 90–105.

Ye, J., & Xiong, T. (2006). Null space versus orthogonal linear discriminant analysis. *ICML'06*.

Zha, H., Ding, C., Gu, M., He, X., & Simon, H. (2002). Spectral relaxation for K-means clustering. *NIPS 14*, 1057–1064.

Zha, H., He, X., Ding, C., Gu, M., & Simon, H. (2001). Bipartite graph partitioning and data clustering. *Proc. Int'l Conf. Information and Knowledge Management (CIKM 2001)*.