A Discriminant Analysis for Undersampled Data

Matthew Robards[1] **Junbin Gao**[2] **Philip Charlton**[3]

[1] School of Wine and Food Science
Charles Sturt University,
Wagga Wagga, NSW 2678,
Email: mrobards@csu.edu.au

[2] School of Accounting and Computer Science
Charles Sturt University,
Bathurst, NSW 2795,
Email: jbgao@csu.edu.au

[3] School of Computing and Mathematics
Charles Sturt University,
Wagga Wagga, NSW 2678,
Email: pcharlton@csu.edu.au

Abstract

One of the inherent problems in pattern recognition is the undersampled data problem, also known as *the curse of dimensionality reduction*. In this paper a new algorithm called pairwise discriminant analysis (PDA) is proposed for pattern recognition. PDA, like linear discriminant analysis (LDA), performs dimensionality reduction and clustering, without suffering from undersampled data to the same extent as LDA.

Keywords: Linear Discriminant Analysis, Pattern Recognition, Dimensionality Reduction.

1 Introduction

Dimensionality reduction (DR) is one of the important steps in many advanced applications such as exploratory data analysis and manifold learning. Pattern recognition, including recognition of faces and handwriting, is a problem often solved by the use of DR and classification algorithms. Other areas where it has been successfully applied include robotics (Ham et al. 2005), information retrieval (He et al. 2004), biometrics (Raytchev et al. 2006, Mekuz et al. 2005), and bioinformatics (Teodoro et al. 2002, Okun et al. 2005). The main goal of DR is to find a corresponding mapping in a much lower dimensional space, of an input data set, without incurring significant information loss. The low dimensional representation can be used in subsequent procedures such as classification, pattern recognition, and so on.

As with general machine learning problems, DR algorithms and their design can be categorized into two major groups: unsupervised DR algorithms and supervised DR algorithms. Principal Component Analysis (PCA), one of the original classical algorithms, along with many modern algorithms like locality preserving projections (LPP) (He et al. 2004) and ISOMAP (Tenenbaum et al. 2000) are well known unsupervised algorithms. This means that they perform their operations giving no consideration to class labels, or more simply, they don't "care" about the type of data they are reducing. In contrast to these unsupervised methods, the famous Linear Discriminant Analysis (LDA) or Fisher Analysis is one of most successful supervised DR algorithms – that is, LDA uses data label information in conducting DR.

The past twenty years have seen the development of numerous new DR algorithms and criterion, see (van der Maaten et al. 2007). Based on their careful observation and analysis, Guo et al. (2007) proposed a unified framework called Twin Measure Embedding (TME) which aims to interpret the design of the majority of DR algorithms in a unified manner. Guo et al. (2007) have also introduced a new kind of DR algorithm, called Twin Kernel Embedding (TKE). It is also noticed here that many successful algorithms have made use of kernel machine learning, see (Schölkopf & Smola 2002).

Two of the most widely known algorithms, PCA and LDA, can be contrasted as being unsupervised and supervised respectively. PCA has been successfully applied to problems such as fault detection and classification (Yue & Tomoyasu 2004), hyperspectral imagery (Wang et al. 2003), and multikey searching (Lee & Chin 1976). LDA on the other hand is primarily used for clustering and classification. Generally speaking the conventional LDA algorithm has the advantages of reasonable motivation in principle and the simplicity in form. It is formulated by specifying a ratio of measure criteria for the between-class scatter matrix and the within-class scatter matrix. The two most popular such measures are the trace criterion and the determinant criterion, see (Duda et al. 2001, Zhuang & Dai 2007).

When the data are well sampled — that is a large sample from each class exists — LDA performs clustering and classification very well. When the data are undersampled, however — that is only a handful of points exist from each class — LDA is unable to perform the required clustering. This phenomenon is also known as the "small sample size problem" (SSS) in (Raudys & Jain 1991). This appears quite often in high-dimensional pattern recognition tasks, where the number of available samples is smaller than the dimensionality of the samples. Recently Zhuang & Dai (2007) proposed a new LDA criterion by using the inverse Fisher criterion with the determinant measure for scatter matrices. This partly solves the problem but it still suffers from the problem of singular between-class scatter matrices. This paper will focus on LDA and its intrinsic limitation — namely the undersampled data problem — and attempt to address

it by way of a new algorithm.

A new supervised algorithm called pairwise discriminant analysis (PDA) is introduced. A major advantage of PDA over LDA is that PDA does not suffer from the curse of undersampled data, a constraint which is well documented as the main drawback of LDA (Ye & Li 2005). Like linear discriminant analysis, PDA attempts to minimise the within-class scatter whilst maximising the between-class scatter. In PDA, however, scatter is determined by a sum of pairwise distances rather than the variance matrix as done in all the previously mentioned methods. Also distinguishing PDA from LDA is that the new algorithm does not project the data values towards a reference point such as the centroids of the classes. Rather, it minimises the distance between all the vectors of the same class, while maximising the distance between each vector and those not in its class. We should note here that a similar algorithm has been investigated in (Zhao et al. 2007) in which the mean centres are used as reference points.

PDA can be used in conjunction with other algorithms such as LPP (He & Niyogi 2004) and local learning projections (LLP) (Wu et al. 2007) to improve the classification rate.

In the next section, we shall introduce the necessary notation and review the conventional PCA and LDA algorithms. Then in section 3, we derive the new PDA algorithm. In section 4, we present several of the experimental results to evaluate the presented method and compare it with the LDA algorithm. Finally, in section 5, we present our conclusions.

2 Linear Discriminant Analysis

2.1 Notation

In this paper we use bold small letters for column vectors and capital letters for a matrix. Let $\mathbf{x}_i^k$ be the i-th datum in the k-th class of K different classes. The number of data vectors in the k-th class is n_k. Let $N = \sum_{k=1}^{K} n_k$ be the total number of data elements and let $\mathbf{X} = \{\mathbf{x}_1^1, \ldots, \mathbf{x}_{n_1}^1, \ldots, \mathbf{x}_1^K, \ldots, \mathbf{x}_{n_K}^K\}$ be the given dataset of N vectors in a high dimensional Euclidean space $\mathbb{R}^d$. Denote the matrix whose columns consist of the data vectors $\mathbf{x}_i^k$ as $\mathbf{X}$. That is, $\mathbf{X} = [\mathbf{x}_1^1, \ldots, \mathbf{x}_{n_1}^1, \ldots, \mathbf{x}_1^K, \ldots, \mathbf{x}_{n_K}^K] \in \mathbf{R}^{d \times N}$. If we don't want to distinguish the class of the data, we simply write $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]$.

One of the objectives of DR algorithms is to find a suitable mapping ϕ which maps each high dimensional vector $\mathbf{x}$ to a lower dimensional vector $\mathbf{f} = \phi(\mathbf{x}) \in \mathbb{R}^{d'}$ where $d' << d$. In linear algorithms like the LDA the desired mapping is a linear transformation P such that $\mathbf{f} = P^T\mathbf{x}$ where P is a matrix of size $d \times d'$.

As suggested by its name, linear discriminant analysis (LDA) is a linear supervised DR algorithm. It takes a matrix of input data $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N]$, and similar to PCA finds the covariance. Unlike PCA however, LDA uses three different covariances known as the within-class variance, between-class variance, and the total variance. The within-class variance is minimised in order to cluster points of the same type together, and the between-class variance is maximised in order to separate the classes from each other.

LDA takes input data $\mathbf{X}$ which is partitioned into K classes. The scatter matrices are then defined. The between-class scatter matrix is given by

$$S_b = \frac{1}{N}\sum_{k=1}^{K} n_k(\mu_k - \mu)(\mu_k - \mu)^T \qquad (1)$$

where μ_k is defined by Ye & Li (2005) as the centroid of the k-th class. Essentially this is the mean data value in the k-th class. μ is defined as the global centroid of the training data $\mathbf{X}$, N as the number of training data points, and n_k as the number of points in k-th class.

The within-class scatter matrix is defined as

$$S_w = \frac{1}{N}\sum_{k=1}^{K}\sum_{x \in A_k}(x - \mu_k)(x - \mu_k)^T \qquad (2)$$

where $A_k = \{\mathbf{x}_1^k, \ldots, \mathbf{x}_{n_k}^k\}$ is the set of data in class k. So it is easy to see that the matrix S_b provides a measure of *scatter* from the centroid of each class to the global centroid, whereas S_w provides a measure of *scatter* from each point $\mathbf{x}_i^k$ to the centroid of its class. Now the LDA algorithm aims to find a projection matrix P which maximizes the ratio of S_b to S_w in an appropriate measure. According to (D. L. Swets 1996) this is achieved by setting the columns of P to the eigenvectors corresponding to the d' largest eigenvalues of $S_w^{-1}S_b$

One limitation, suggested by Ye and Li (Ye & Li 2005), is that classical LDA requires at least one of the scatter matrices to be nonsingular, a condition not always satisfied with many real life data sets. This is caused by the undersampled problem. Samples of data in which a low number of points exist in a high dimensional space often result in nonsingular scatter matrices. This means that LDA cannot perform a dimension reduction on the data, a problem which often occurs in such areas as facial recognition and text analysis. Modifications, such as regularized LDA and PCA + LDA detailed below, have been suggested to address the problem.

2.2 PCA + LDA

Zhuang & Dai (2007) suggest that taking the optimal transformation matrix,

$$P_{\text{opt}}^T = P_{LDA}^T \dot{P}_{PCA}^T \qquad (3)$$

where P_{LDA}^T is the transformation matrix obtained by the LDA procedure and P_{PCA}^T is that of PCA, will in general address the problem of singularity in the within-class scatter matrix. In order to ensure non-singularity however, they suggest a resultant dimensionality from the PCA step of $d' \leq N - K - c$ where c is usually equal to 1. Note here N is the number of input data points and K is the number of classes.

A drawback of this method is that PCA might consider certain information insignificant in calculation of the principal components, and thus it is lost. This insignificant information to PCA however may be quite significant for classification in the LDA step. Thus vital information can be lost using this method, and optimal classification may not be achieved.

2.3 Regularized LDA

To resolve the undersampled problem Multilevel PCA followed by Regularized LDA is utilised and can be applied to facial recognition (Lin & Tang 2006). According to (Ye & Li 2005), the method of regularized LDA for dealing with singularity in S_w simply adds a multiple of the identity matrix. ie.

$$S_w' = S_w + \sigma I_{d \times d} \qquad (4)$$

They claim that this new scatter matrix is positive definite, and thus non singular. A drawback to this method however, is the difficulty of calculating an optimal value for σ.

3 Pairwise Discriminant Analysis (PDA)

Since LDA encounters the problem of the scatter matrix being singular, we instead aim to use the sum of pairwise distances as the measure of scatter. This idea is inspired by the example of the TKE (Guo et al. 2006, 2007) where the concept of pairwise measure is adopted to maintain the similarity measure between the data pairs.

In the LDA algorithm family, researchers mainly focus on the covariance matrices and their scatter measure criteria. When the dataset is undersampled compared with the higher dimension, the singularity of the scatter matrix often happens. To avoid using any scatter matrix we adopt a scatter measure using the sum of all pairwise distances.

3.1 Derivation of the Algorithm

In the following, the PDA process is outlined to reduce the dimensionality of a dataset $\mathbf{X} \in \mathbf{R}^{d\times N}$ given class labels $\{1, 2, \ldots, K\}$. As is done in the LDA algorithm family, the data $\mathbf{x}_i \in \mathbf{R}^d$ is to be reduced in dimension by a transformation P, to $\mathbf{f}_i \in \mathbf{R}^{d'}$ where d' is significantly smaller than d.

Let us define the within-class distance scatter by

$$\sum_{i=1}^{N} \sum_{\mathbf{x}_j \in \text{class}(\mathbf{x}_i)} ||\mathbf{f}_i - \mathbf{f}_j||^2. \tag{5}$$

Expression (5) gives the total sum of the distance squared between transformed data pairs which are in the same class. It should be noted that the within-class distance scatter may also be defined as

$$S = \sum_{k=1}^{K} \sum_{\mathbf{x}_i, \mathbf{x}_j \in A_k} \|\mathbf{f}_i - \mathbf{f}_j\|^2. \tag{6}$$

However it is easily seen that this is simply twice the value of (5). We prefer expression (5) in terms of simple notation and coding.

Similarly, we can work out the sum of the distance between the projected data pairs which are not in the same class. We call it the between-class distance scatter, i.e.,

$$\sum_{i=1}^{N} \sum_{\mathbf{x}_j \notin \text{class}(\mathbf{x}_i)} ||\mathbf{f}_i - \mathbf{f}_j||^2. \tag{7}$$

In terms of clustering and classification it is natural to minimise the within-class distance scatter and maximise the between-class distance scatter, as with the LDA algorithm. That is, after a transformation by way of P we hope the projected data within a class should be close together while data in different classes should be widely separated. This goal can be achieved if the within-class distance scatter is minimized and the between-class distance scatter is maximized, so we have two separate objectives:

$$\begin{cases} \min \sum_{i=1}^{N} \sum_{\mathbf{x}_j \in \text{class}(\mathbf{x}_i)} ||\mathbf{f}_i - \mathbf{f}_j||^2, \\ \max \sum_{i=1}^{N} \sum_{\mathbf{x}_j \notin \text{class}(\mathbf{x}_i)} ||\mathbf{f}_i - \mathbf{f}_j||^2. \end{cases} \tag{8}$$

Since we cannot always achieve both of these goals simultaneously, we compromise by forming a weighted difference of the two objectives and minimizing the single objective function

$$\sum_{i=1}^{N} \left(\sum_{\mathbf{x}_j \in \text{class}(\mathbf{x}_i)} ||\mathbf{f}_i - \mathbf{f}_j||^2 - \lambda \sum_{\mathbf{x}_j \notin \text{class}(\mathbf{x}_i)} ||\mathbf{f}_i - \mathbf{f}_j||^2 \right). \tag{9}$$

Here λ is imposed as a weight scalar to provide balance between the within-class and between-class distance scatters. In other words, we are basically assigning an importance to the between-class distance scatter. This effectively gives us control over the level of clustering we wish to perform. Since $\mathbf{f}_i = P^T\mathbf{x}_i$, (9) can be written as

$$\sum_{i=1}^{N} \left(\sum_{\mathbf{x}_j \in \text{class}(\mathbf{x}_i)} ||P^T\mathbf{x}_i - P^T\mathbf{x}_j||^2 - \lambda \sum_{\mathbf{x}_j \notin \text{class}(\mathbf{x}_i)} ||P^T\mathbf{x}_i - P^T\mathbf{x}_j||^2 \right). \tag{10}$$

The objective function (10) can then be minimized as follows. We first recognise that

$$\begin{aligned} ||P^T\mathbf{x}_i - P^T\mathbf{x}_j||^2 &= ||P^T(\mathbf{x}_i - \mathbf{x}_j)||^2 \\ &= (\mathbf{x}_i - \mathbf{x}_j)^T PP^T(\mathbf{x}_i - \mathbf{x}_j). \end{aligned} \tag{11}$$

Therefore the within-class distance scatter (5) becomes

$$\begin{aligned} \text{Tr}(\sum_{i=1}^{N} &[\mathbf{x}_i - \mathbf{x}_{i_1}, \mathbf{x}_i - \mathbf{x}_{i_2}, \ldots, \mathbf{x}_i - \mathbf{x}_{i_k}]^T \\ &PP^T[\mathbf{x}_i - \mathbf{x}_{i_1}, \mathbf{x}_i - \mathbf{x}_{i_2}, \ldots, \mathbf{x}_i - \mathbf{x}_{i_k}]) \end{aligned} \tag{12}$$

where $\mathbf{x}_{i_k}$'s are in the same class as $\mathbf{x}_i$. Let $S_{w_i} = [\mathbf{x}_i - \mathbf{x}_{i_1}, \mathbf{x}_i - \mathbf{x}_{i_2}, \ldots, \mathbf{x}_i - \mathbf{x}_{i_k}]$ which allows us to write expression (5) as

$$\text{Tr}(PP^T \sum_{i=1}^{N} S_{w_i} S_{w_i}^T) \tag{13}$$

which, upon formulating $A = \sum_{i=1}^{N} S_{w_i} S_{w_i}^T$, we can write as

$$\text{Tr}(PP^T A). \tag{14}$$

To handle the second part of our objective function, formulated in (7), we define $S_{b_i} = [\mathbf{x}_i - \mathbf{x}_{i_1}, \mathbf{x}_i - \mathbf{x}_{i_2}, \ldots, \mathbf{x}_i - \mathbf{x}_{i_m}]$, where each $\mathbf{x}_{i_m}$ is not in the same class as $\mathbf{x}_i$. Then we construct B similar to A, such that $B = \sum_{i=1}^{N} S_{b_i} S_{b_i}^T$. Therefore (7) becomes

$$\text{Tr}(PP^T B) \tag{15}$$

and our objective function is simply

$$\text{Tr}(PP^T(A - \lambda B)). \tag{16}$$

Obviously there exists a trivial solution $P = 0$ to the above problem. At this point we need to apply a constraint to P to avoid trivial solutions. We may consider many different constraints. For example, we can require that the columns of P be linearly independent unit vectors.

However, in this paper, we would like to use the constraint $P^T P = I$ which gives the following optimization problem:

$$\min_P \text{Tr}(P^T(A - \lambda B)P) \tag{17}$$

$$\text{Subject to:} \quad P^T P = I_{d' \times d'} \tag{18}$$

The problem (17)–(18) can be solved by the following procedure: Let $P^* \in \mathbb{R}^{d \times d'}$ denote the matrix whose columns consist of d' eigenvectors associated with the d' smallest eigenvalues of $d \times d$ matrix $A - \lambda B$. Then P^* is a solution for both the problems (17)–(18). Of course, when P^* is a solution of (17)–(18), then PR, for any orthogonal matrix $R \in \mathbb{R}^{d' \times d'}$, is also a solution.

Once we have found out the transformation matrix $P = [\mathbf{p}_1, \ldots, \mathbf{p}_{d'}]$, for any data point $\mathbf{x}_i$ in the higher dimensional space its projected lower dimensional data can be calculated as $P^T \mathbf{x}_i$. The low dimensional space is spanned by the columns of P.

3.2 The Computer Algorithm

The PDA algorithm was implemented in Matlab so that it could be tested with image datasets. Pseudocode for the three major procedures of the algorithm is presented below.

Procedure *findA*
Input: Data matrix X, vector of class variables G.
Output: A.
Comments: This procedure finds A as defined in §3.1.

1. Initialise A to be a $d \times d$ matrix of zeroes.
2. For $i = 1$ to N,
   - 2.1 Find all vectors $\mathbf{x}_{i_k}$ in the class of $\mathbf{x}_i$.
   - 2.2 $S_{w_i} \leftarrow [\mathbf{x}_i - \mathbf{x}_{i_1}, \mathbf{x}_i - \mathbf{x}_{i_2}, \ldots, \mathbf{x}_i - \mathbf{x}_{i_k}]$.
   - 2.3 $A \leftarrow A + S_{w_i} S_{w_i}^T$.

Procedure *findB*
Input: Data matrix X, vector of class variables G.
Output: B.
Comments: This procedure finds B as defined in §3.1.

1. Initialise B to be a $d \times d$ matrix of zeroes.
2. For $i = 1$ to N,
   - 2.1 Find all vectors $\mathbf{x}_{i_m}$ that are not in the class of $\mathbf{x}_i$.
   - 2.2 $S_{b_i} \leftarrow [\mathbf{x}_i - \mathbf{x}_{i_1}, \mathbf{x}_i - \mathbf{x}_{i_2}, \ldots, \mathbf{x}_i - \mathbf{x}_{i_m}]$.
   - 2.3 $B \leftarrow B + S_{b_i} S_{b_i}^T$.

Procedure *PDA*
Input: Data matrix X, vector of class variables G, λ, and number of output dimensions d'.
Output: Transformation P.
Comments: This is the main procedure for PDA.

1. Assign A using the *findA* procedure.
2. Assign B using the *findB* procedure.
3. Find eigenvalues and eigenvectors of $A - \lambda B$.
4. Assign columns of P to be the d' eigenvectors corresponding to the d' smallest eigenvalues of $A - \lambda B$.

3.3 Further Considerations

As pointed out in the introduction, our algorithm has an intrinsic link with LDA in the sense that the scatter is measured by using distances in PDA while it is measured by the covariances in LDA. A similar approach can be applied in the PCA formulation by maximising distance scatter, however we have noted that this idea has been implemented in the so-called Maximum Variance Unfolding (Weinberger & Saul 2006).

Like the conventional LDA, the present formulation of the within-class distance scatter does not take the size of the classes into account. Thus the algorithm may suffer on datasets with unequal class sizes. However, this could be handled by using averaging of the within-class distance. Since the number of pairs of vectors (excluding self-pairs) in class k is $n_k(n_k-1)/2$, the within-class distance scatter with averaging is

$$\sum_{k=1}^{K} \frac{1}{n_k(n_k-1)} \sum_{\mathbf{x}_i, \mathbf{x}_j \in A_k} \|\mathbf{f}_i - \mathbf{f}_j\|^2. \tag{19}$$

Furthermore, we may include the distance information of the original data in the scatter definition. For example, for each pair of original data $\mathbf{x}_i$ and $\mathbf{x}_j$ we may impose on $\|\mathbf{f}_i - \mathbf{f}_j\|^2$ a weight factor depending on the distance $\|\mathbf{x}_i - \mathbf{x}_j\|$ such that if $\|\mathbf{x}_i - \mathbf{x}_j\|$ is small, the $\mathbf{f}_i$ and $\mathbf{f}_j$ should be close in the projected space. The strategies used in (Tenenbaum et al. 2000, Roweis & Saul 2000) can be adopted in our case.

4 Experimental Results

In this section we illustrate the utility and properties of the proposed method given in the previous section. We will test the proposed algorithm against the conventional LDA algorithm to demonstrate the robustness of the proposed method in the case of undersampled scenarios.

One of the most well studied datasets for testing clustering and classification algorithms is the USPS handwritten postcode digits. This dataset can be obtained from
`http://www.cs.toronto.edu/~roweis/data.html`.
This dataset contains 1100 handwritten samples of each digit 0–9. Each digit is scanned as an image of 16×16 pixels, resulting in a 256-dimensional vector for each digit. We grouped the data into ten classes labeled by digit (with class 10 corresponding to the digit 0).

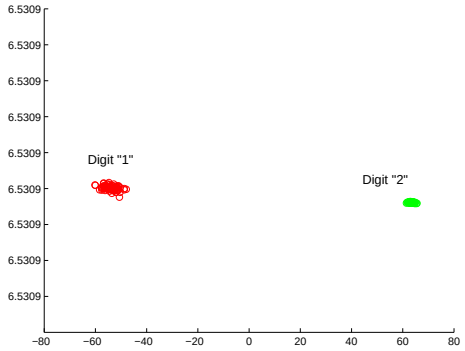
In the results below we projected the 256-dimensional space of digit images onto a 3-dimensional space ie. our dimension parameters are

$d = 256$ and $d' = 3$. For ease of presentation, the results are further projected onto the plane by suppressing the third coordinate.

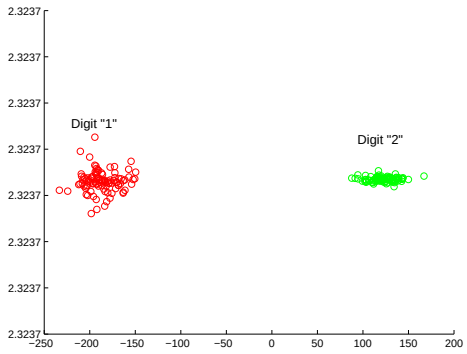
4.1 Determining λ Parameter

The parameter λ in (17) controls the trade-off between the within-class and between-class scatters. In practical applications, the λ has to be carefully tuned. In this experiment, we only demonstrate how different values of λ affect the clustering.

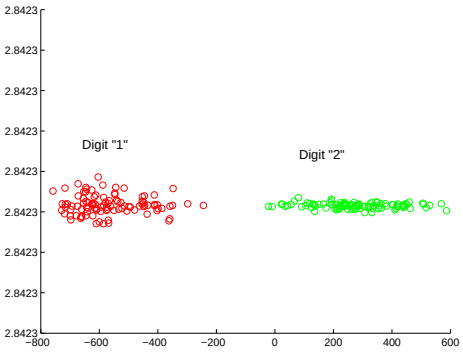
Figure 1 shows the results for different values of λ, where 100 points were taken from two classes, those for the digits 1 and 2. We do not show the output for the LDA algorithm, as it failed to cluster with such undersampled data. The λ value can be seen to control the balance between favoring within-class distance scatter and favoring between-class distance scatter. With a small λ value (see Figure 1(a)) each class is collapsing into a point while with larger λ values (see Figure 1(d)) the classes are widely scattered. This demonstrates the effect of the parameter λ, however finding a method to discover the ideal λ value is still an outstanding problem.



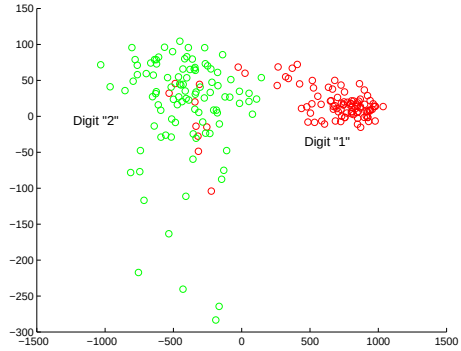
(a) $\lambda = 0.001$



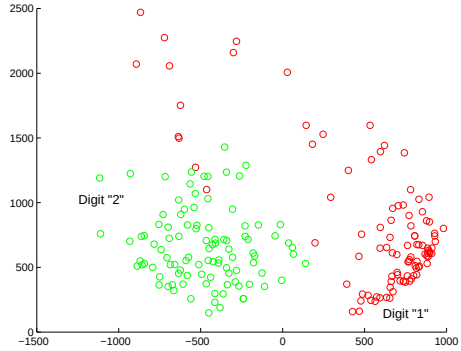
(b) $\lambda = 0.01$



(c) $\lambda = 0.1$



(d) $\lambda = 1$



(e) $\lambda = 10$

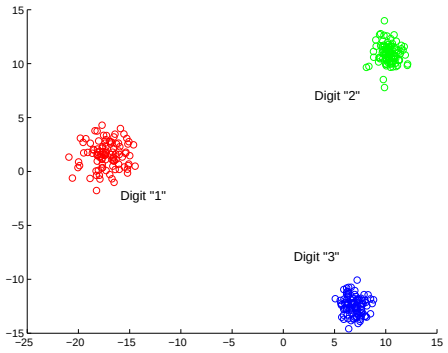
Figure 1: 200 digits taken across 2 classes clustered using PDA with $\lambda = 0.001$ (a) then $\lambda = 0.01$ (b) $\lambda = 0.1$ (c) $\lambda = 1$ (d) and $\lambda = 10$ (e).

4.2 Clustering With PDA

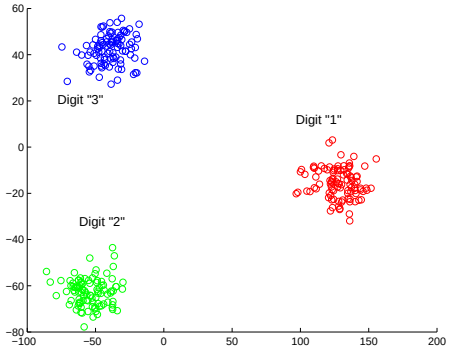
4.2.1 Well Sampled Data

When data is well sampled, as shown below, PDA and LDA both perform clustering proficiently. Taking 300 points from across 3 classes of the USPS handwritten digits dataset, the clustering with LDA and PDA is comparable (see Figure 2). The ideal λ value used in obtaining these results was found by multiple testing.

An important point to note here is that different images exhibit different levels of similarity. For example, the above clustering was performed on the handwritten digits 1, 2 and 3. Sometimes digits are more alike – for example, many classification techniques have difficulty distinguishing the digits 3, 6 and 8. Figure 3 shows a comparison between the results of clustering performed on the digits 3, 6 and 8 by PDA and LDA.

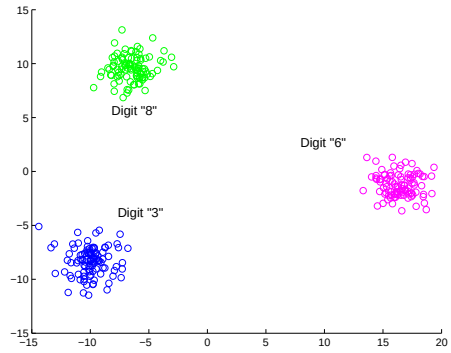


(a) LDA

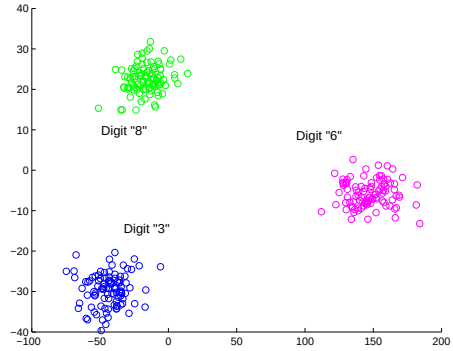


(b) PDA

Figure 2: 300 digits taken across 3 classes clustered using LDA (a) then PDA (b) with $\lambda = 0.01$



(a) LDA



(b) PDA

Figure 3: 100 images of each number 3, 6, 8, clustered using LDA (a) then PDA (b) with $\lambda = 0.01$

As with the real life dataset, LDA and PDA are fairly comparable when applied to a synthetic dataset. In Figure 4, a comparison is made between LDA and PDA when performing clustering on a well sampled collection of data from the synthetic dataset. This synthetic data, like the handwritten digits, comprises 1100 vectors. These vectors, in 256 dimensions, are divided into 10 classes, each of which has a random normal distribution. The classes are each slightly overlapping.

4.2.2 Undersampled Data

As we have seen, both LDA and PDA are comparable in the cases of well sampled data collections. In the following two experiments, we performed PDA on undersampled datasets to test its robustness in clustering. First we use the same synthetic dataset as in the previous section.

Figure 5 shows the clustering performed by PDA on 51 synthetic data points. The LDA output is also given, however it should be noted that this is a trivial solution where all the data entries of the same class have been projected to the same point. LDA cannot perform clustering on 51 data points as it considers this to be undersampled. This provides our motivation for choosing such a number of data and we are given a good indication of PDA's capabilities when LDA cannot perform clustering.

To further test the extent of PDA's capabilities in clustering, a sample of 10 points was taken from each class. PDA performs consistently, despite the data being extremely undersampled which is evidenced in Figure 6. Here only 10 points have been taken from each class, in both the real life dataset and the synthetic dataset. In these cases LDA failed to cluster the data. For this reason no results are displayed for LDA, as only a trivial solution would be displayed like that in Figure 5(a).

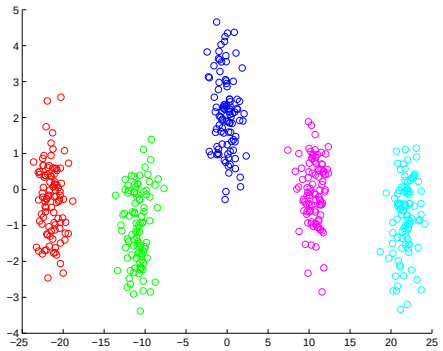
4.3 Classification With PDA

PDA is here compared to LDA for classification capabilities. Table 1 illustrates the classification rates for LDA and PDA. The column headed n_k should be noted. This is the number of points in each class that were used in finding the transformation matrix P. Once P is determined through training, it can be applied to points where the class is unknown *a priori*. To classify these unknown data points, they were projected into the d'-dimensional space using the projection P. We then classified each point as being in the class to whose centroid it was closest. The classification rate represents the fraction of unknown data points which were correctly classified by each method.

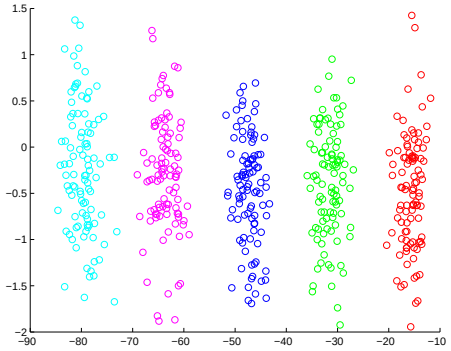
It should be noted for the more poorly classified sets in LDA's case, the projections appeared to be random. This was due to a trivial projection where all training points of the same class were projected to the same point.

5 Conclusions

This paper introduces the pairwise discriminant analysis algorithm PDA and compares its performance with LDA. Our results suggest that with undersampled data, LDA is not able to perform classification well. PDA on the other hand is very capable when it comes to clustering and classification in cases where the data are undersampled. This is evidenced by the consistently higher classification rate for PDA over



(a) LDA



(b) PDA

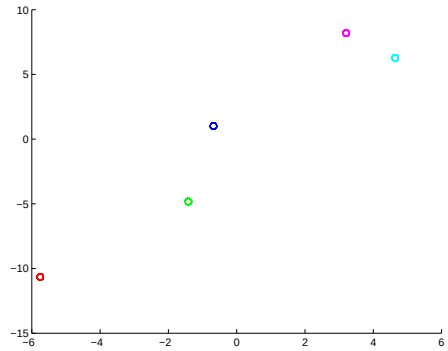
Figure 4: 500 digits taken across 5 classes of synthetic data clustered using LDA (a) then PDA (b) with $\lambda = 0.01$

LDA. When well-sampled data are used, for example 600 data points, LDA performs marginally better at classification than PDA. The experimental results supporting this fact have been omitted here, however, as the current paper is only concerned about pattern recognition in undersampled data. It is very rare that a real life dataset will be well-sampled.

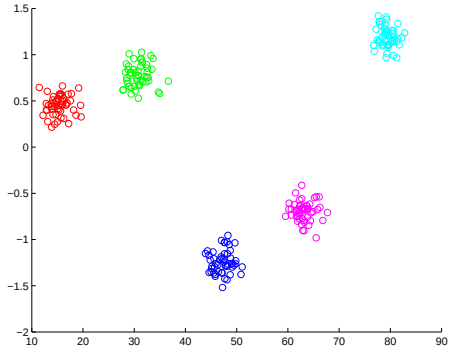
Future directions for this research will investigate algorithms such as LLP (Wu et al. 2007), LPP (He & Niyogi 2004) and TKE (Guo et al. 2006, 2007) in depth. Further work will be directed at using these algorithms in conjunction with PDA. PDA will also be further investigated to improve its clustering capabilities.

	n_k	LDA classification rate (%)	PDA classification rate (%)
2 classes (digits 1 & 2)	10	56.5	89.5
	50	62.0	96.5
	100	54	97.5
3 classes (digits 1, 2 & 3	10	49.3	90.3
	50	48.3	93.3
	100	77.7	94.6

Table 1: Classification of the handwritten digits dataset using PDA and LDA, where n_k is the number points from each class used in training.



(a) LDA



(b) PDA

Figure 5: 51 digits taken across 5 classes clustered using LDA (a) and PDA (b) with $\lambda = 0.01$. Note the trivial solution for LDA.

References

D. L. Swets, J. J. W. (1996), 'Using discriminant eigenfeatures for image retrieval', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **18**(8), 831–836.

Duda, R., Hart, P. & Stork, D. (2001), *Pattern Classification*, 2nd edn, John Wiley and Sons, New York.

Guo, Y., Gao, J. & Kwan, P. W. (2006), Visualization of non-vectorial data using twin kernel embedding, *in* 'International Workshop on Integrating AI and Data Mining', pp. 11–17.

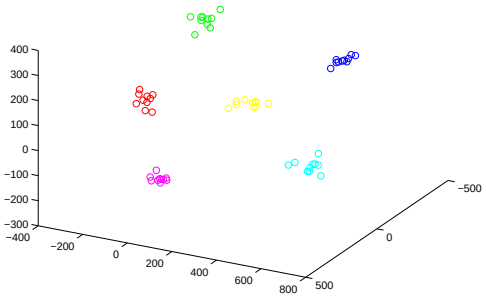
Guo, Y., Gao, J. & Kwan, P. W. (2007), 'Twin measure embedding: A unified framework for visualizing non-vectorial data in low dimensional spaces', *submitted to IEEE Trans on Pattern Recognition and Machine Intelligence* .

Ham, J., Lin, Y. & Lee, D. D. (2005), Learning nonlinear appearance manifolds for robot localization, *in* 'IEEE/RSJ International Conference on Intelligent Robots and Systems', pp. 2971– 2976.

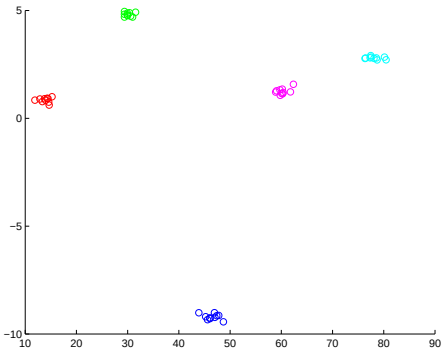
He, X., Ma, W. & Zhang, H. (2004), Learning an image manifold for retrieval, *in* 'ACM conference on Multimedia 2004', New York, pp. 17–23.

He, X. & Niyogi, P. (2004), Locality preserving projections, *in* S. Thrun, L. Saul & B. Schölkopf, eds, 'Advances in Neural Information Processing Systems 16', MIT Press, Cambridge, MA.

Lee, R. C. T. & Chin, Y. H. (1976), 'Application of principal component analysis to multikey searching', *IEEE Transactions on Software Engineering* **2**(3), 185–193.



(a) Handwritten digits



(b) Synthetic dataset

Figure 6: 10 digits taken from each of 6 classes in the handwritten dataset, clustered using PDA. LDA is unable to perform this clustering, as the data is too undersampled (a). Then 10 digits taken from each of 5 classes in the synthetic dataset, clustered using PDA. Note that LDA was not able to perform clustering on 51 points from each class in the synthetic data (b). In each case $\lambda = 0.01$

Lin, D. & Tang, X. (2006), Recognize high resolution faces: From macrocosm to microcosm, *in* 'Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition', pp. 1335–1362.

Mekuz, N., Bauckhage, C. & Tsotsos, J. K. (2005), Face recognition with weighted locally linear embedding, *in* 'Proceedings of the 2nd Canadian Conference on Computer and Robot Vision', pp. 290–296.

Okun, O., Priisalu, H. & Alves, A. (2005), Fast nonnegative dimensionality reduction for protein fold recognition., *in* 'ECML', pp. 665–672.

Raudys, S. & Jain, A. (1991), 'Small sample size effects in statistical pattern recognition: recommendations for practitioners', *IEEE Trans Pattern Analysis and Machine Intelligence* **31**, 252–264.

Raytchev, B., Yoda, I. & Sakaue, K. (2006), Multi-view face recognition by nonlinear dimensionality reduction and generalized linear models, *in* 'FGR '06: Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition (FGR06)', Washington, DC, USA, pp. 625–630.

Roweis, S. T. & Saul, L. K. (2000), 'Nonlinear dimensionality reduction by locally linear embedding', *Science* **290**(22), 2323–2326.

Schölkopf, B. & Smola, A. (2002), *Learning with Kernels*, The MIT Press, Cambridge, Massachusetts.

Tenenbaum, J. B., de Silva, V. & Langford, J. C. (2000), 'A global geometric framework for nonlinear dimensionality reduction', *Science* **290**(22), 2319–2323.

Teodoro, M. L., Jr, G. N. P. & Kavraki, L. E. (2002), A dimensionality reduction approach to modeling protein flexibility, *in* 'International Conference on Computational Molecular Biology (RECOMB)', pp. 299–308.
URL: *http://citeseer.ist.psu.edu/teodoro02dimensionality.html*

van der Maaten, L., Postma, E. & van den Herik, H. (2007), *Dimensionality Reduction: A Comparative Review.*
URL: *http://www.cs.unimaas.nl/l.vandermaaten/dr/DR_draft.pdf*

Wang, C., Menenti, M. & Li, Z. (2003), Modified principal component analysis (MPCA) for feature selection of hyperspectral imagery, *in* 'Proceedings of IEEE International Conference on Geoscience and Remote Sensing Symposum', Vol. 6, pp. 3781–3783.

Weinberger, K. Q. & Saul, L. K. (2006), An introduction to nonlinear dimensionality reduction by maximum variance unfolding, *in* 'Proceedings of the National Conference on Artificial Intelligence (AAAI)', Boston MA.

Wu, M., Yu, K., Yu, S. & Schölkopf, B. (2007), Local learning projections, *in* 'Proceedings of the 24th International Conference on Machine Learning', Vol. 227, pp. 1039–1046.

Ye, J. & Li, Q. (2005), 'A two stage linear discriminant analysis via qr-decomposition', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **27**(6), 929–941.

Yue, H. H. & Tomoyasu, M. (2004), Weighted principal component analysis and its applications to improve FDC performance, *in* 'Proceedings of 43rd IEEE Conference on CDC', Vol. 4, pp. 4262–4267.

Zhao, D., Lin, Z., Xiao, R. & Tang, X. (2007), Linear Laplacian discrimination for feature extraction, *in* 'IEEE Conf. on Computer Vision and Pattern Recognition (CVPR'07)'.

Zhuang, X. S. & Dai, D. Q. (2007), 'Improved discriminate analysis for high-dimensional data and ts application to face recognition', *Pattern Recognition* **40**(5), 1570–1578.