

**homework #3**

**Linear Discriminant Analysis**

**EE 8993: Fundamentals of Speech Recognition**

July 23, 1998

*submitted to:*

Dr. Joseph Picone

*submitted by:*

Suresh Balakrishnama

Institute for Signal and Information Processing  
Department of Electrical and Computer Engineering  
Mississippi State University  
MS 39762, USA  
Email: balakris@isip.msstate.edu



## 1. INTRODUCTION

There are many possible techniques for classification of data. Principle Component Analysis (PCA) and Linear Discriminant Analysis (LDA) are two commonly used techniques for data classification and dimensionality reduction. Linear Discriminant Analysis easily handles the case where the within-class frequencies are unequal and their performances has been examined on randomly generated test data. This method maximizes the ratio of between-class variance to the within-class variance in any particular data set thereby guaranteeing maximal separability. The use of Linear Discriminant Analysis for data classification is applied to classification problem in speech recognition. We decided to implement an algorithm for LDA in hopes of providing better classification compared to Principle Components Analysis. The prime difference between LDA and PCA is that PCA does more of feature classification and LDA does data classification. In PCA, the shape and location of the original data sets changes when transformed to a different space whereas LDA doesn't change the location but only tries to provide more class separability and draw a decision region between the given classes. This method also helps to better understand the distribution of the feature data.

### Case I:

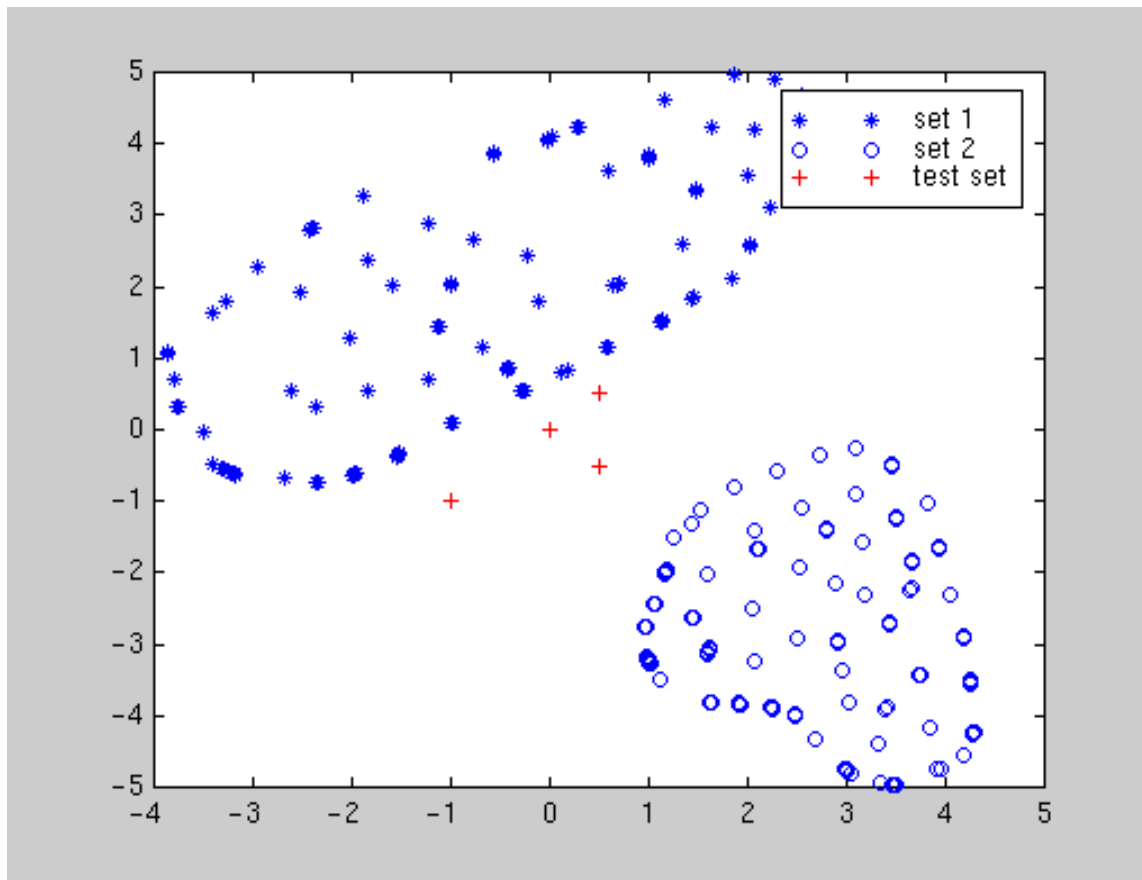


Figure 1. Figure showing data sets and test vectors for case I in original space

## 2. Problem Description

The data sets, set1 and set2 were given in linear space each consisting of 100 points. The test vectors were also given which are to be classified belonging to either set1 or set2. The condition given was that each data set 1 should have a mean of (-2,2) and data set2 should have a mean of (2, -2). The objective is to transform the data sets into a new space and using the euclidean distance classify the test vectors to one of the two sets. There are two ways of approaches in this technique: Class dependent method and Class independent method. Both approaches were applied to this problem.

Data point	Coordinates
x1	-1,-1
x2	0, 0
x3	0.5, 0.5
x4	0.5, -0.5

Table 1: Test vectors considered for the problem

## 3. Implementation and Results

This classification problem was solved using matlab package. The data sets were created using xmgr, a graphical tool in Unix environment. The steps adopted to solve this problem are as follows:

1. Data set1 and set2 were created in such a way that the mean of set1  $\mu_1$  was (2,-2) and mean of set 2  $\mu_2$  was (-2,2). The data obtained from xmgr was adjusted in matlab code to get the required mean. The four test vectors were defined and all these points were plotted in a graph which pictorially demonstrates three points belonging to both sets.
2. Compute the mean of each data set and mean of entire data set. Let  $\mu_1$  and  $\mu_2$  be the mean of set 1 and set 2 respectively and  $\mu_3$  be mean of entire data, which is obtained by merging set 1 and set 2, is given by Equation 1.

$$\mu_3 = p_1 \times \mu_1 + p_2 \times \mu_2 \quad (1)$$

where  $p_1$  and  $p_2$  are the apriori probabilities of the classes. In the case of this simple two class problem, the probability factor is assumed to be 0.5.

3. In LDA, within-class and between-class scatter are used to formulate criteria for class separability. Within-class scatter is the expected covariance of each of the classes. The scatter measures are computed using Equations 2 and 3.

$$S_w = \sum_j p_j \times (cov_j) \quad (2)$$

Therefore, for the two-class problem,

$$S_w = 0.5 \times cov_1 + 0.5 \times cov_2 \quad (3)$$

All the covariance matrices are symmetric. Let  $cov_1$  and  $cov_2$  be the covariance of set 1 and set 2 respectively. Covariance matrix is computed using the following equation.

$$cov_j = (\mathbf{x}_j - \mu_j)(\mathbf{x}_j - \mu_j)^T \quad (4)$$

The between-class scatter is computed using the following equation.

$$S_b = \sum_j (\mu_j - \mu_3) \times (\mu_j - \mu_3)^T \quad (5)$$

Note that  $S_b$  can be thought of as the covariance of data set whose members are the mean vectors of each class. As defined earlier, the optimizing criterion in LDA is the ratio of between-class scatter to the within-class scatter. The solution obtained by maximizing this criterion defines the axes of the transformed space. However for the class-dependent transform the optimizing criterion is computed using equation (6). It should be noted that if the LDA is a class dependent type, for  $L$ -class  $L$  separate optimizing criterion are required for each class. The optimizing factors in case of class dependent type are computed as

$$criterion_j = inv(cov_j) \times S_b \quad (6)$$

For the class independent transform, the optimizing criterion is computed as

$$criterion = inv(S_w) \times S_b \quad (7)$$

	Cov	$S_b$	$S_w$	Optimizing criterion
set 1	$\begin{bmatrix} 3.5271 & 2.1607 \\ 2.1607 & 2.8932 \end{bmatrix}$	$\begin{bmatrix} 4.0001 & -4.00 \\ -4.00 & 4.0001 \end{bmatrix}$	$\begin{bmatrix} 2.3284 & 0.9778 \\ 0.9778 & 2.2503 \end{bmatrix}$	$\begin{bmatrix} 3.6519 & -3.651 \\ -4.109 & 4.109 \end{bmatrix}$

	Cov	$S_b$	$S_w$	Optimizing criterion
set 2	$\begin{bmatrix} 1.1298 & -0.250 \\ -0.250 & 1.6074 \end{bmatrix}$	$\begin{bmatrix} 4.0001 & -4.00 \\ -4.00 & 4.0001 \end{bmatrix}$	$\begin{bmatrix} 2.3284 & 0.9778 \\ 0.9778 & 2.2503 \end{bmatrix}$	$\begin{bmatrix} 3.1621 & -3.162 \\ -2.0851 & 2.085 \end{bmatrix}$

Table 2: Covariance, between class scatter, within class scatter and optimizing criterion for class dependent type

	Cov	$S_b$	$S_w$	Optimizing criterion
set 1	$\begin{bmatrix} 3.5271 & 2.1607 \\ 2.1607 & 2.8932 \end{bmatrix}$	$\begin{bmatrix} 4.000 & -4.00 \\ -4.00 & 4.000 \end{bmatrix}$	$\begin{bmatrix} 2.3284 & 0.9778 \\ 0.9778 & 2.2503 \end{bmatrix}$	$\begin{bmatrix} 3.0145 & -3.01 \\ -3.08 & 3.0874 \end{bmatrix}$
set 2	$\begin{bmatrix} 1.1298 & -0.205 \\ -0.205 & 1.6074 \end{bmatrix}$	$\begin{bmatrix} 4.000 & -4.00 \\ -4.00 & 4.000 \end{bmatrix}$	$\begin{bmatrix} 2.3284 & 0.9778 \\ 0.9778 & 2.2503 \end{bmatrix}$	$\begin{bmatrix} 3.0145 & -3.01 \\ -3.01 & 3.0874 \end{bmatrix}$

Table 3: Covariance, between class scatter, within class scatter and optimizing criterion for class independent type

4. By definition, an eigen vector of a transformation represents a 1-D invariant subspace of the vector space in which the transformation is applied. A set of these eigen vectors whose corresponding eigen values are non-zero are all linearly independent and are invariant under the transformation. Thus any vector space can be represented in terms of linear combinations of the eigen vectors. A linear dependency between features is indicated by a zero eigen value. To obtain a non-redundant set of features all eigen vectors corresponding to non-zero eigen values only are considered and the ones corresponding to zero eigen values are neglected. In the case of LDA, the transformations are found as the eigen vector matrix of the different criteria defined in Equations 6 and 7.

	Eigen values	Eigen vectors
set 1	$\begin{bmatrix} 0 & 0 \\ 0 & 6.1020 \end{bmatrix}$	$\begin{bmatrix} -0.7071 & 0.6986 \\ -0.7071 & -0.7155 \end{bmatrix}$

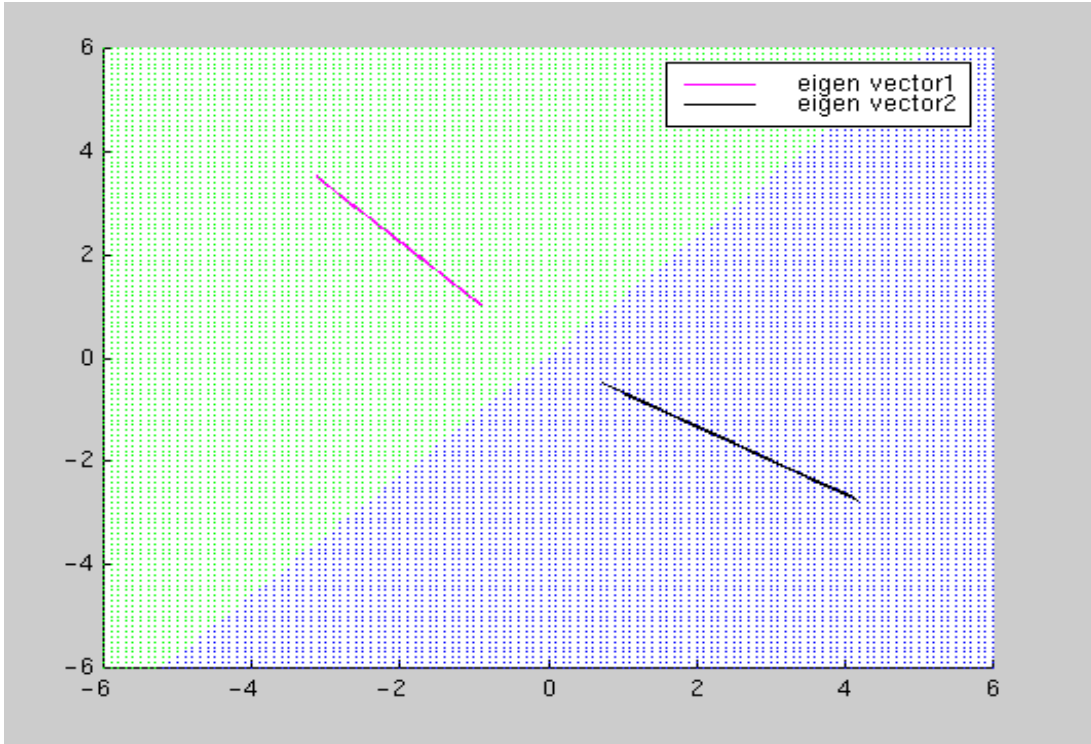


Figure 2. Figure for eigen vector direction in class dependent type

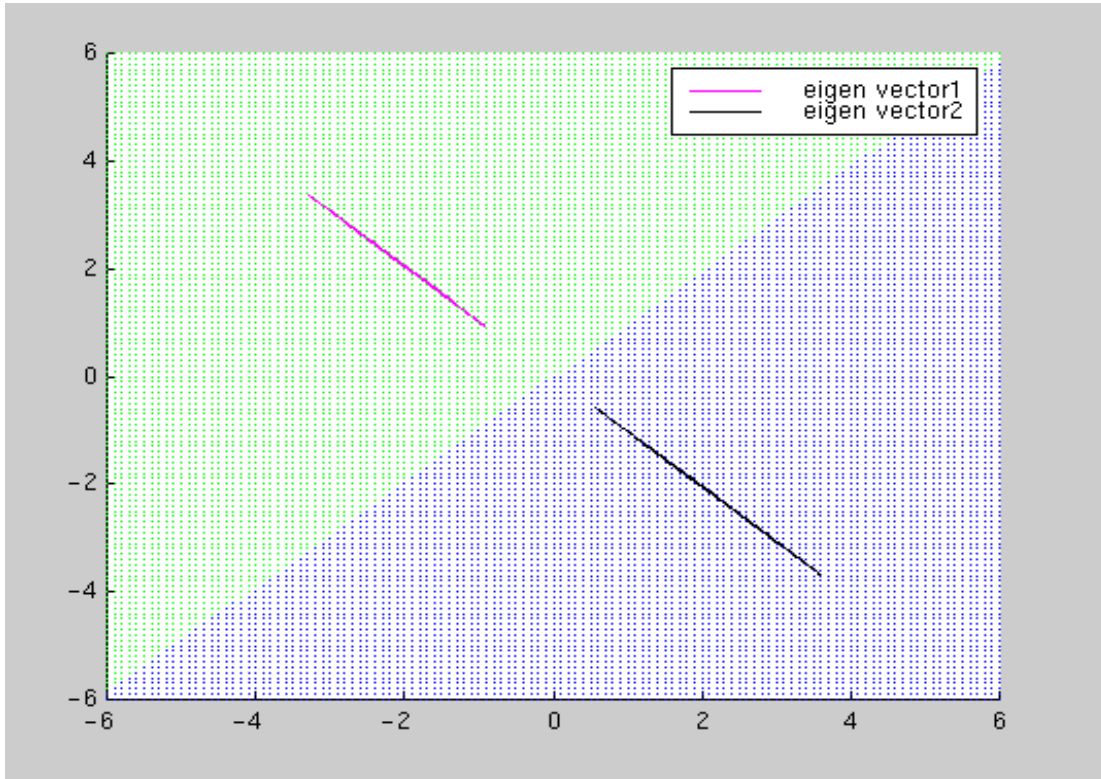


Figure 3. Figure for eigen vector direction in class independent type

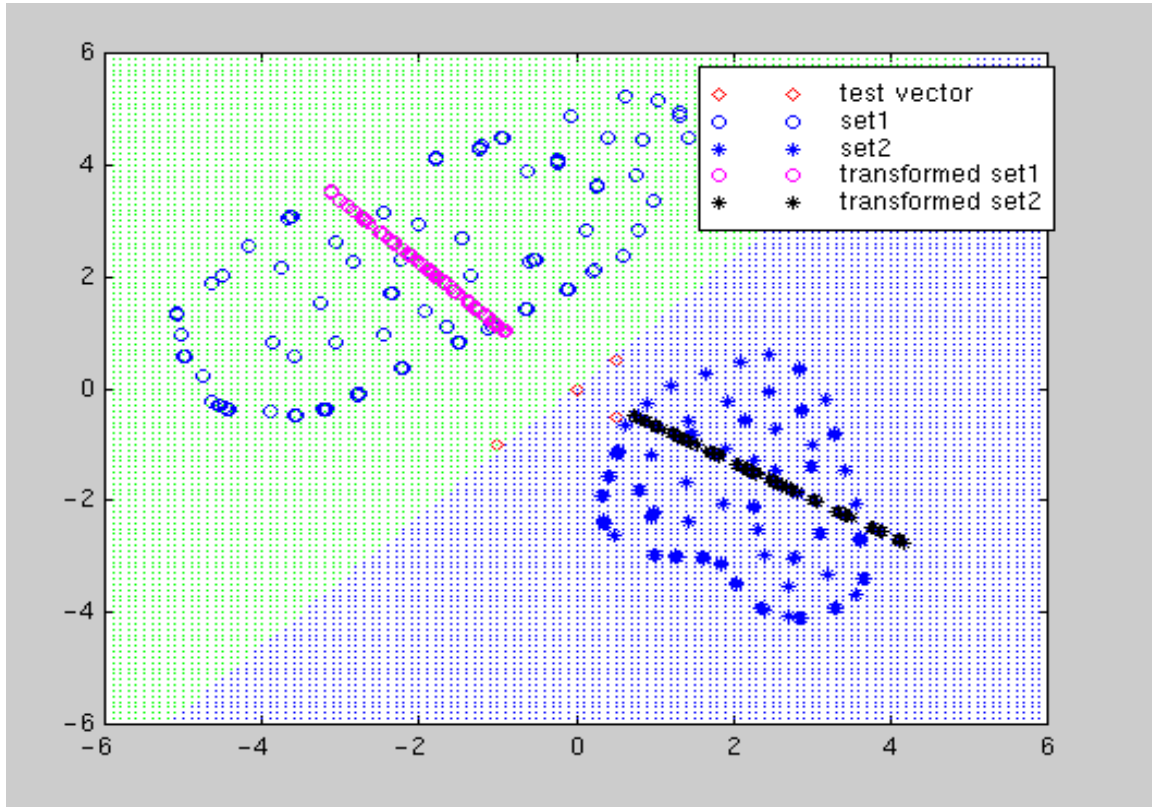


Figure 4. Data sets in original space and transformed space along with the transformation axis for class dependent LDA of a 2-class problem

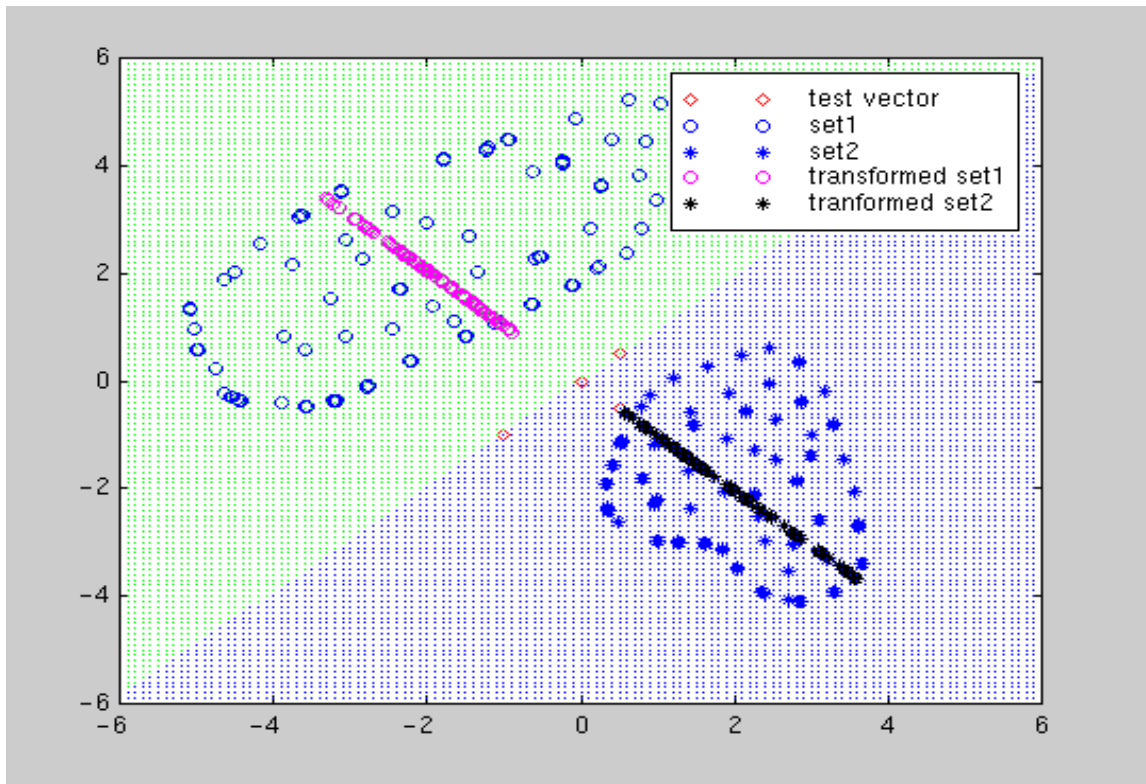


Figure 5. Data sets in original space and transformed space along with the transformation axis for class independent LDA of a 2-class problem

	Eigen values	Eigen vectors
set 2	$\begin{bmatrix} 5.2472 & 0 \\ 0 & 0.0 \end{bmatrix}$	$\begin{bmatrix} 0.8348 & 0.7071 \\ -0.5505 & 0.7071 \end{bmatrix}$

Table 4: Eigen values and eigen vectors of data sets for class dependent type

	Eigen values	Eigen vectors
set 1	$\begin{bmatrix} 0 & 0 \\ 0 & 6.1020 \end{bmatrix}$	$\begin{bmatrix} -0.7071 & 0.6986 \\ -0.7071 & -0.7155 \end{bmatrix}$
set 2	$\begin{bmatrix} 0 & 0 \\ 0 & 6.1020 \end{bmatrix}$	$\begin{bmatrix} -0.7071 & 0.6986 \\ -0.7071 & -0.7155 \end{bmatrix}$

Table 5: Eigen values and eigen vectors of data sets for class independent type

6. For any  $L$ -class problem we would always have  $L-1$  non-zero eigen values. This is attributed to the constraints on the mean vectors of the classes in Equation 1. The eigen vectors corresponding to non-zero eigen values for the definition of the transformation.

For our 2-class example, Figures 2 and 3 show the direction of the significant eigen vector along which there is maximum discrimination information. Having obtained the transformation matrices, we transform the data sets using the single LDA transform or the class specific transforms which ever the case may be. From the figures it can be observed that, transforming the entire data set to one axis provides definite boundaries to classify the data. The decision region in the transformed space is a solid line separating the transformed data sets thus,

For the class dependent LDA,

$$transformed\_set\_j = transform\_j^T \times set\_j \quad (8)$$

For the class independent LDA,

$$transformed\_set = transform\_spec^T \times data\_set^T \quad (9)$$

Similarly the test vectors are transformed and are classified using the euclidean distance of the test vectors from each class mean.



The two Figures 4 and 5 clearly illustrate the theory of Linear Discriminant Analysis applied to a 2-class problem. The original data sets are shown and the same data sets after transformation are also illustrated. It is quite clear from these figures that transformation provides a boundary for proper classification. In this example the classes were properly defined but cases where there is overlap between classes, obtaining a decision region in original space will be very difficult and in such cases transformation proves to be very essential. Transformation along largest eigen vector axis is the best transformation.

7. Once the transformations are completed using the LDA transforms, Euclidean distance or RMS distance is used to classify data points. Euclidean distance is computed using Equation 10 where  $\mu_{ntrans}$  is the mean of the transformed data set,  $n$  is the class index and  $\mathbf{x}$  is the test vector. Thus for  $n$  classes,  $n$  euclidean distances are obtained for each test point.

$$dist_n = (transform\_n\_spec)^T \times \mathbf{x} - \mu_{ntrans} \quad (10)$$

	set 1	set 2	classification
x1	2.8452	2.8113	set 2
x2	2.8283	2.8282	set 2
x3	2.8198	2.8367	set 1
x4	3.5353	2.1212	set 2

Table 6: Euclidean distance of test vectors from data sets in transformed space for class dependent type

	set 1	set 2	classification
x1	2.8452	2.8113	set 2
x2	2.8283	2.8282	set 2
x3	2.8198	2.8367	set 1
x4	3.5353	2.1212	set 2

Table 7: Euclidean distance of test vectors from data sets in transformed space for class independent type

8. The smallest Euclidean distance among the  $n$  distances classifies the test vector as belonging to class  $n$ .

**Case II:**

Parallel ellipses: Class dependent and class independent LDA approaches are applied to a different type of data which are parallel ellipses as shown in figure 6. The variances of both data sets are just the same. LDA fails to provide a proper classification for this type of data. The main reasons are that both data sets have same mean, same variance and are parallel to each other. Figure 7 and 8 clearly show that LDA fails to obtain a definite decision region separating both data sets in original space hence becomes further complex to draw a line in transformed space.

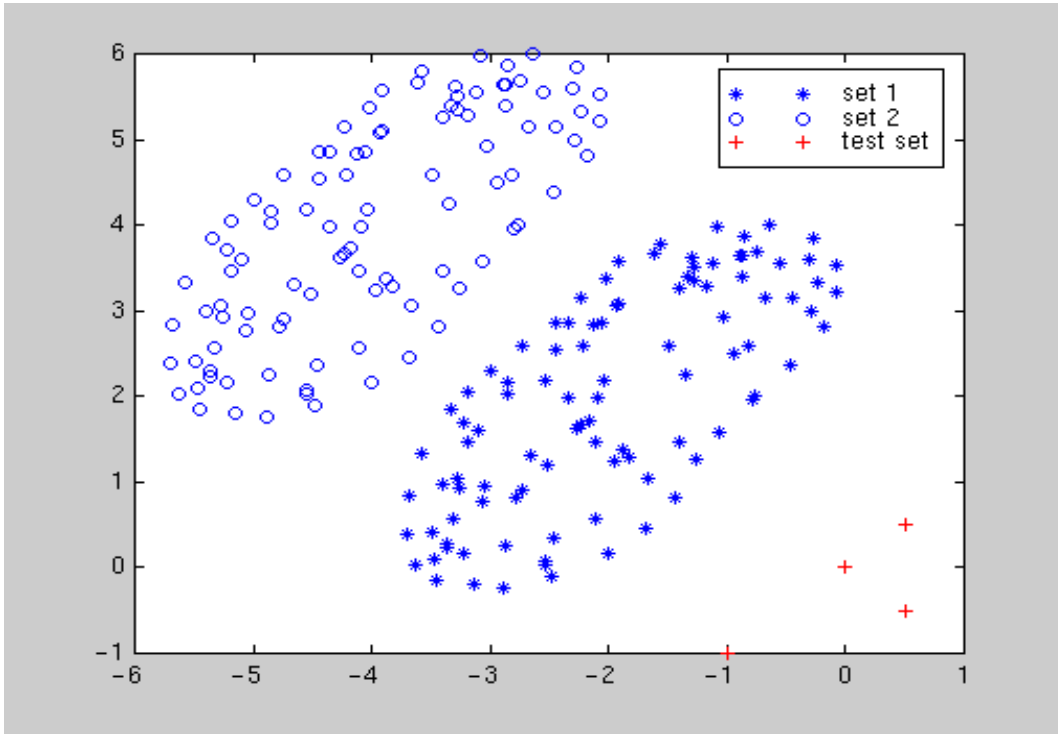


Figure 6. Figure showing data sets and test vectors for case II in original space

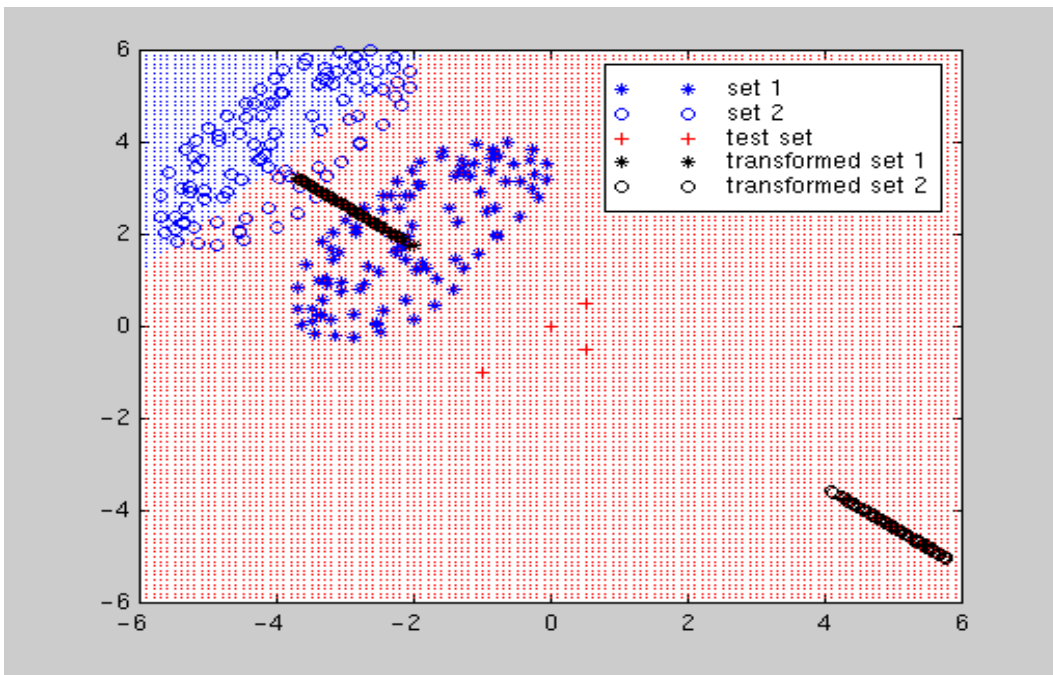


Figure 7. Data sets for case II in original space and transformed space along with the transformation axis for class dependent LDA

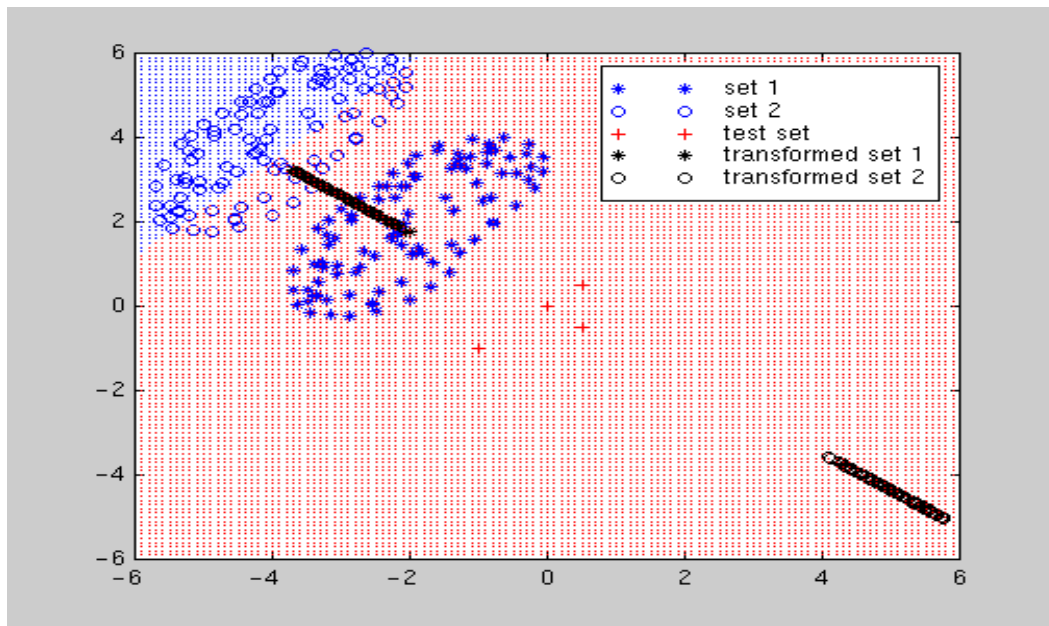


Figure 8. Data sets for case II in original space and transformed space along with the transformation axis for class independent LDA

### Case III:

Overlapping sets: The third case for this problem are overlapping sets, each stretching at an angle of  $45^\circ$  to the axes as shown in figure 8. Both sets are perpendicular to each other. Both class specific LDA and class independent LDA are applied to both data sets to find the decision regions. One of the problem encountered in this case is that there exist some data points which are common to both sets and would not help in obtaining a perfect decision region. Figure 9 and 10 shows plots of the two data sets in the original space and data sets in the transformed space along with the decision regions using class specific LDA and class independent LDA. The class specific LDA does a better job compared to class independent type because the data sets are transformed separately into the transformed space which enables to obtain different separate decision regions for each half of the data sets.

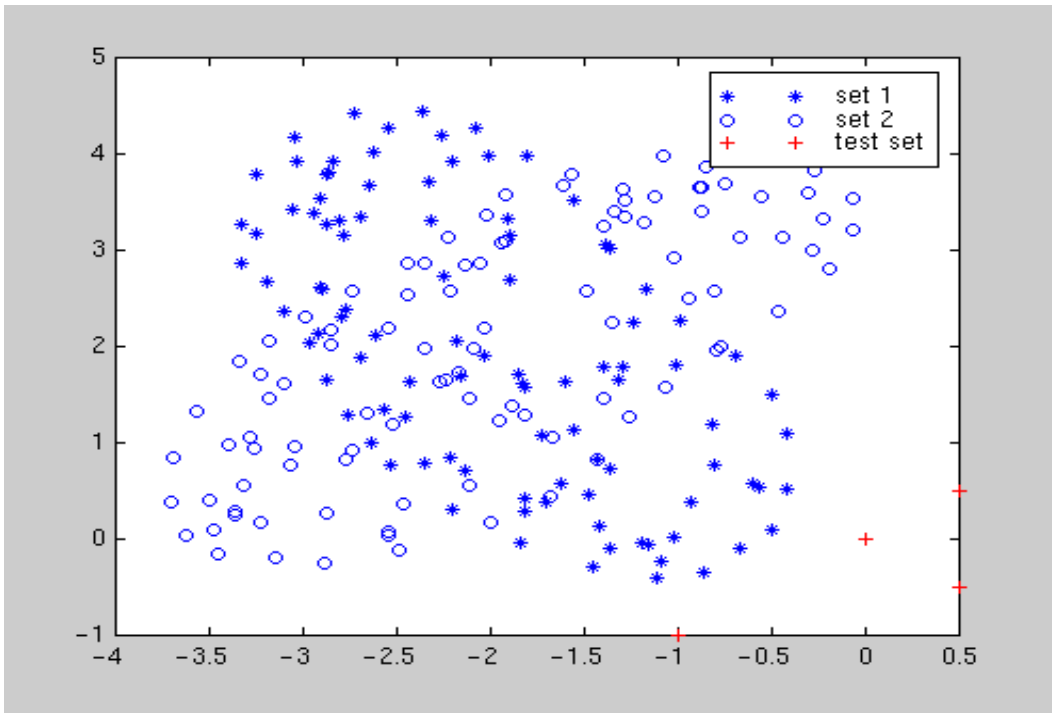


Figure 9. Figure showing data sets and test vectors for case III in original space

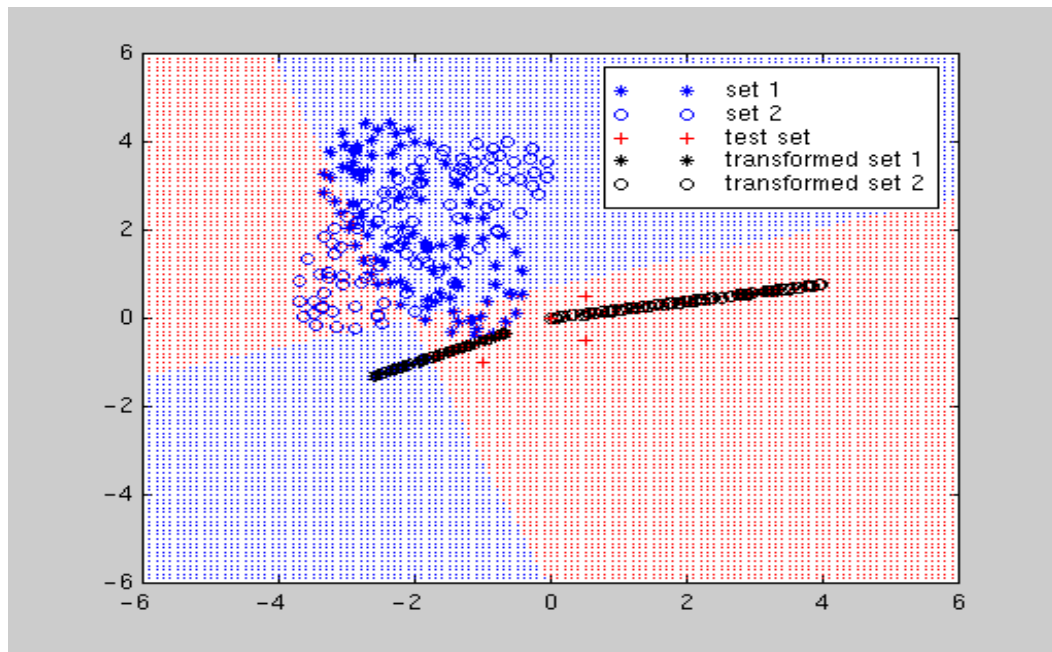


Figure 10. Data sets for case III in original space and transformed space along with the transformation axis for class dependent LDA

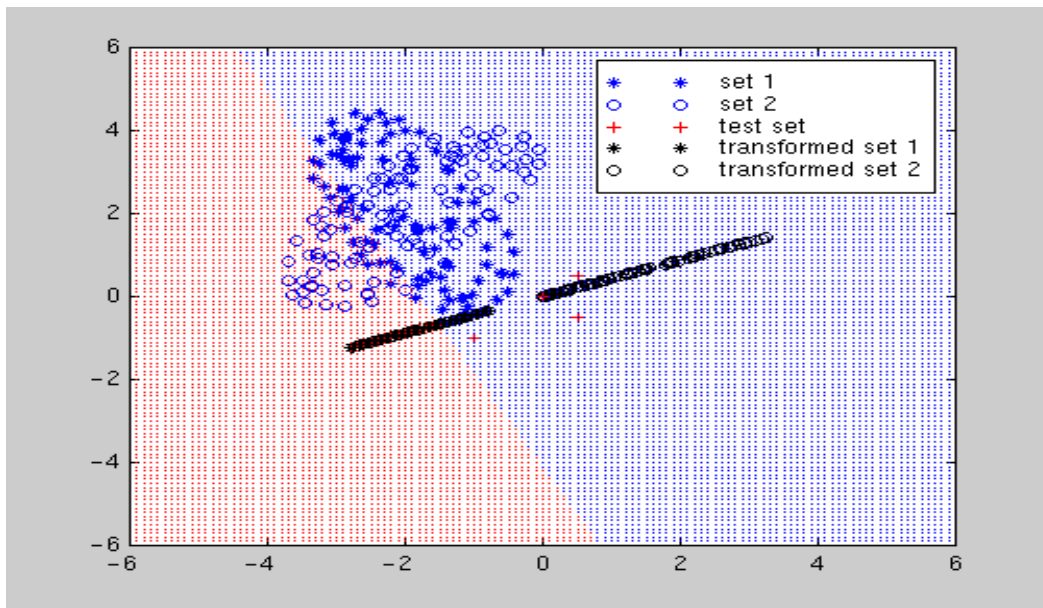


Figure 11. Data sets for case III in original space and transformed space along with the transformation axis for class independent LDA

#### 4. CONCLUSIONS

We have presented the theory and implementation of LDA as a classification technique. Two approaches to LDA, namely, class independent and class dependent, have been implemented for the given problem. The choice of the type of LDA depends on the data set and the goals of the classification problem. If generalization is of importance, the class independent transformation is preferred. However, if good discrimination is what is aimed for, the class dependent type should be the first choice.

#### 5. SOFTWARE

All Matlab code written for this project is available for public from our website at [www.isip.msstate.edu](http://www.isip.msstate.edu)

#### 6. REFERENCES

- [1] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Academic Press, San Diego, California, 1990.