**Universidad Veracruzana**
**Instituto de Investigaciones en Inteligencia Artificial**

# Genetic Projective Discriminant Analysis

Protocolo de tesis.

Presenta:
**Víctor David García Medina**

Asesor:
**Hector Gabriel Acosta Mesa**
Coasesores:
**Adriana Laura Lopez Lobato**
**Efrén Mezura Montes**

Xalapa, Veracruz, México　　Enero 2025

# Contents

# 1 Introduction

Machine learning is a branch of Artificial intelligence that aims to create programs capable of generalizing behavior from information provided in the form of examples, these examples are datasets. Machine learning can be categorized into the following types: supervised learning (classification) which uses labeled dataset to train algorithms to recognize patterns, unsupervised learning (clustering) that uses unlabeled data to discover insights without guidance, and reinforcement learning that uses trial and error to teach an agent to take actions in an environment to maximize rewards. A dataset can be interpreted as a matrix where each row represents an observation and each column represents a feature. In general, a dataset can contain irrelevant and redundant features, which increase the computational cost for both storage and processing, ultimately affecting model performance. Additionally, theoretical issues arise when a dataset has more dimensions than observations, all these characteristics are known as the curse of dimensionality. To handle data effectively, it is necessary to reduce its dimensionality. Dimensionality reduction refers to the transformation of high-dimensional data into a lower-dimensional representation, retaining the essential structure and minimizing the number of parameters needed to describe the data's observed attributes.

Dimensionality reduction is crucial in a variety of fields as it not only decreases dimensionality but also eliminates undesirable attributes in high-dimensional data. This process is achieved using various statistical methods, such as Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), and Singular Value Decomposition (SVD). The techniques for dimensionality reduction can be broadly categorized into two main approaches: feature dimension reduction and feature selection. The key difference between these approaches is that feature selection identifies a subset of the original features, while feature extraction generates new features based on the original data. Figure 1 presents a taxonomy tree with some examples of these methods.

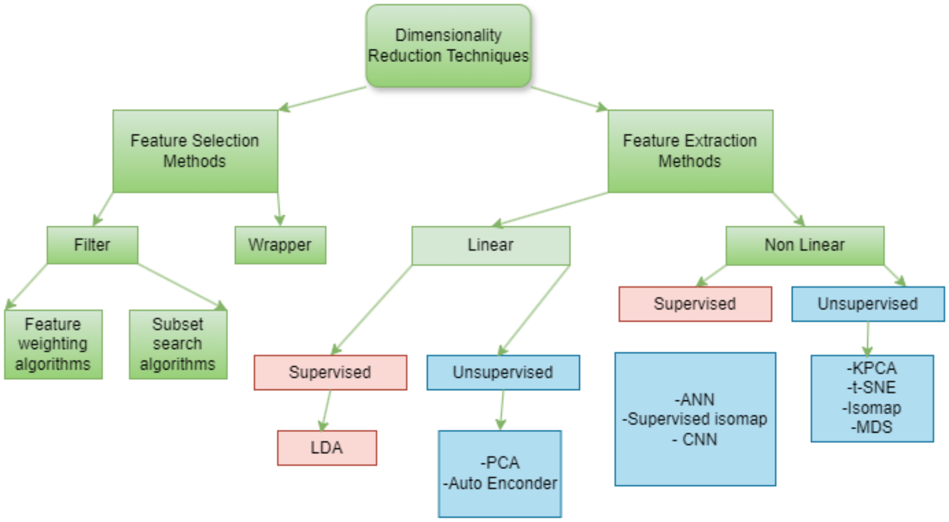


Figure 1: Dimensionality Reduction techniques taxonomy

In machine learning, dimensionality reduction is a crucial step to improve the efficiency of classification algorithms and facilitate the visualization of complex data, Linear Discriminant Analysis (LDA) and Principal Component Analysis (PCA) are the most popular techniques for this purpose. These techniques have been used for a variety of purposes, including facial recognition [1], discrimination of osteoarthritic cartilage from healthy cartilage [2], and classification of chemical processes [3]. Fisher Discriminant Analysis (FDA), proposed by R.A. Fisher [4], is a technique that has been used for this purpose as it provides an effective way to project a high-dimensional dataset into a subspace while preserving the ability to differentiate between predefined classes. It has been used in face recognition [5, 6] and spoken language identification [7], however its main drawback is that it assumes Gaussian distributions with the same covariance matrix for the classes which may not be suitable for more complex problems where decision boundaries are nonlinear [8, 9].There is a problem called the Small Sample Size (SSS) problem which occurs when the number of dimensions in a database significantly exceeds the number of observations, which can lead to unreliable or unstable results in learning algorithms. In the case of LDA, this problem often leads to poor solutions or even the impossibility of obtaining results due to the occurrence of complex eigenvalues. There is an alternative called Kernel Discriminant Analysis (KDA) that extends FDA to nonlinear separability by using the kernel trick to implicitly project data into a higher dimensional feature space. This improves its ability to deal with problems with nonlinearly separable classes although it leads to a new problem which is to disregard the explicit mapping, compromising the interpretability of the process [10]. This method is also used in face recognition to explore the geometry of colors distribution [6].

Genetic algorithms have been widely used to solve complex optimization problems. These iterative algorithms emulate the process of natural evolution to find optimal solutions by selection, crossover and mutation techniques [11].

Given the potential of genetic algorithms to explore and optimize complex solution spaces, the idea of integrating them with LDA has been contemplated in different fields of artificial intelligence: In the article [12] a genetic algorithm that uses the Fisher ratio and the Bhattacharyya distance as fitness function is used as a feature extraction technique on images in the HSI space to reduce dimensionality. A similar idea is presented in [13] where an evolutionary algorithm is used in audio files to reduce the dimensionality using PCA. A more traditional approach was proposed in [14] where individuals are vectors that represent linear transformations (flattened matrices) and the fitness function is the Fisher criterion, all the operators modify the vectors using recombination, bit flip and swap. This method is known as Genetic Algorithm Linear Discriminant Analysis (GLDA).

The first proposals mentioned work adequately although they are only applicable to very specific databases, while the GLDA algorithm keeps the main idea but lacks a crucial feature: The solution is a transformation that keeps the dimension of the original space and doesn't guarantee zero covariance between the principal axes.

## 2 Definition of the Problem

Let $D$ be a dataset with $k$ observations, represented as:

$$D = \{(x,l) \mid x \in \mathbb{R}^n \text{ and } l \in \{1,\ldots,m\}\}$$

where each vector $x$ is an observation and $l$ is its corresponding label. If we ignore the labels, $D$ can be interpreted as a matrix on $\mathbb{R}^{k\times n}$. Now, consider a projection matrix $P \in \mathbb{R}^{n\times p}$, where $p < n$, and define the transformed dataset $D'$ as $D'_{k\times p} = D_{k\times n} \times P_{n\times p}$. This transformation reduces the dimensionality of the dataset, resulting in fewer features than the original.

This dimensionality reduction raises key questions:

1. Does $D'$ preserve the same separability properties as $D$ with respect to the class labels?

2. Under what conditions does the incorporation of label information ensure that the reduced dataset $D'$ retains the essential discriminatory features of $D$?

As mentioned in the introduction, PCA and LDA are the most popular methods for dimensionality reduction. Both methods utilize a projection onto a reduced space: PCA achieves dimensionality reduction by maximizing variability, while LDA maximizes likelihood while maintaining class separability. These properties directly address the questions posed above, projecting to reduce variability might get the classes mixed, see Figure 2. To avoid this problem there exist the Fisher's criterion designed to maximize separability between classes while minimizing the spread within each class. It is mathematically expressed as:

$$J(w) = \frac{w^T S_B w}{w^T S_W w}$$

where:

- $w$ is the projection vector.

- $S_B$ (between-class scatter matrix) quantifies the variance between the means of different classes:

  $$S_B = \sum_{i=1}^{c} N_i (\mu_i - \mu)(\mu_i - \mu)^T,$$

  where $c$ is the number of classes, $N_i$ is the number of samples in class $i$, $\mu_i$ is the mean vector of class $i$, and $\mu$ is the overall mean vector.

- $S_W$ (within-class scatter matrix) quantifies the variance within each class:

  $$S_W = \sum_{i=1}^{c} \sum_{x \in \text{class } i} (x - \mu_i)(x - \mu_i)^T,$$

  where $x$ represents data points in class $i$.

A **larger** $J(w)$ indicates that the projection vector $w$ leads to better class separation, LDA optimizes $J(w)$ to find the projection $w$ that best separates the classes. The Fisher criterion is especially effective when the data are approximately Gaussian and linearly separable.

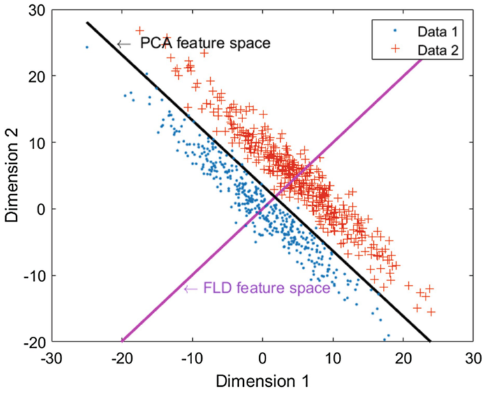


Figure 2: Diferent projections using PCA and FLD (particular case of LDA) [15]

For the general case of more than two classes, the **Fisher criterion** is extended to handle $c$ classes by generalizing the between-class and within-class scatter matrices. The goal remains the same: maximize class separability while minimizing within-class variance.

The criterion can be expressed as:

$$J(W) = \frac{\det(W^T S_B W)}{\det(W^T S_W W)},$$

where:

- $W$ is a matrix whose columns are the projection vectors.
- $S_B$ (between-class scatter matrix) is defined as:

  $$S_B = \sum_{i=1}^{c} N_i(\mu_i - \mu)(\mu_i - \mu)^T,$$

  where:

  - $c$ is the number of classes.
  - $N_i$ is the number of samples in class $i$.
  - $\mu_i$ is the mean vector of class $i$.
  - $\mu$ is the overall mean vector:

    $$\mu = \frac{1}{N}\sum_{i=1}^{c} N_i \mu_i,$$

    where $N$ is the total number of samples.

- $S_W$ (within-class scatter matrix) is defined as:

  $$S_W = \sum_{i=1}^{c} \sum_{x \in \text{class } i} (x-\mu_i)(x-\mu_i)^T,$$

  where $x$ represents a data point in class $i$.

The optimal projection $W$ is obtained by solving the generalized eigenvalue problem:

$$S_B W = \lambda S_W W,$$

where $\lambda$ is a diagonal matrix of eigenvalues. The columns of $W$ correspond to the eigenvectors associated with the largest eigenvalues. The interpretation of this idea is

- The projection $W$ maps the high-dimensional data into a lower-dimensional subspace (at most $c-1$ dimensions) where the class separability is maximized.
- A higher $J(W)$ value indicates better separability among the $c$ classes.

Once the eigenvectors are found, the ones with the hightest eigenvalues are selected to reduce dimension. The final number of eigenvectors is either the number of classes minus one or the number of dimensions minus one. This generalized Fisher criterion is the key point of Linear Discriminant Analysis (LDA) to reduce dimensionality while preserving discriminative information.

In the article [14], the authors present an alternative to maximize this function using a genetic approach called Genetic Linear Discriminant Analysis (GLDA), parameters are not specified and the details of the algorithm are the following:

- **Representation** Vectors that represent a flattened matrix, the length of these vectors is $n^2$ where $n$ is the number of features of the dataset
- **Selection operator** Binary tournament selection
- **Recombination** One Point Crossover with a fixed probability $\mu$, this is, a random crossover point in the range of the individual length, the portions of the individual lying to the right side of the crossover point in two parents are exchanged to produce two offspring
- **Mutation operators**

  - Random point mutation: A chromosome is selected at random, one bit is changed with probability $0.25\mu$
  - Swap mutation: A chromosome is selected at random and two bits are selected randomly, these bits are swapped with probability $0.25\mu$
  - Creep mutation: Add a small random value to one bit of the selected chromosome with $0.3\mu$ probability.
  - Scramble: A chromosome is selected at random and the values are reconfigured with $0.2\mu$ probability.

- **Fitness Function** Fisher Criterion
- **Termination criteria** fixed number of iterations

So far, the idea is clear: to find a linear transformation $W$ such that the Fisher criterion is maximized. However, this approach relies on certain assumptions, such as the classes following a normal distribution and having identical (ideally) covariance matrices. In practice, many datasets fail to meet these requirements. Furthermore, the method encounters challenges in the presence of the small sample size problem, where the number of features exceeds the number of observations. This imbalance can lead to singular covariance matrices and result in complex eigenvalues, making the computation of the transformation $W$ problematic, and GLDA might not be suitable alternative for this cases since it uses the same principies behind LDA.

# 3 Propossal

Design and implement an alternative to GLDA by considering orthonormal matrices that, starting from a set of attributes, finds a reduced-dimensional basis in which the projection of two (or more) classes are linearly separable using the Silhouette index as fitness function.

# 4 Justification

The GLDA method retains the original number of dimensions by representing individuals as squared matrices, leaving the selection of features to discard at the discretion of the user. Moreover, since GLDA relies on the Fisher criterion, the solution it provides can be identical to that of traditional LDA. This limitation is particularly significant in datasets where the number of features exceeds the number of observations, a scenario that LDA does not handle effectively or has no results.

# 5 Hypothesis

The design of a genetic algorithm that uses individuals capable of reducing dimensionality while preserving orthogonality, evaluated by a fitness function that maximizes the separability between classes, will improve the interpretation of the transformed space. In addition, the use of these individuals will mitigate the problems associated with databases with more dimensions than observations, where LDA often fail and GLDA does not consider. Finally, by incorporating the silhouette index as an evaluation metric, more interpretable and bounded results will be obtained, avoiding extreme values derived from Fisher's criterion and ensuring a better representation of the separability of classes.

# 6 Objectives

## 6.1 General Objective

Design and implement a genetic algorithm capable of reducing the dimensions of a database to 2 or 3, preserving or increasing the separability between classes. The algorithm should match or exceed the results obtained by LDA and GLDA, offering an effective solution for Small Sample Size problem scenarios, where the number of dimensions significantly exceeds the number of records. This algorithm will be refered as the Genetic Projective Discriminant Analysis (GPDA)

## 6.2 Specific Objectives

- Implement the algorithm for the 2 and 3 dimensional cases
- Compare results with LDA using different types of databases where normal distribution, identical covariance matrices and more dimensions than observations might or might not be the case
- Compare the results obtained in [14] using the same metrics with the same databases

# 7 Methodology

## 7.1 The silhouette index

The **silhouette index** is a measure used to evaluate the quality of clustering results. It provides an assessment of how well each data point lies within its assigned cluster and how distinct clusters are from one another. The silhouette index combines cohesion (how close data points in the same cluster are) and separation (how far apart clusters are). Since this index also measures separability, this will be used as the fitness function for the proposal. Mathematically, the silhouette value $s(i)$ for a data point $i$ is defined as:

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))},$$

where:

- $a(i)$ is the average distance from $i$ to all other points in the same cluster (intra-cluster distance).
- $b(i)$ is the minimum average distance from $i$ to points in other clusters (inter-cluster distance).

The silhouette value $s(i)$ lies in the range $[-1, 1]$, with the following interpretations:

- $s(i) \approx 1$: The data point is well-clustered, as it is much closer to points in its own cluster than to points in other clusters.
- $s(i) \approx 0$: The data point lies on or near the boundary between two clusters.
- $s(i) \approx -1$: The data point may be assigned to the wrong cluster, as it is closer to points in another cluster than to those in its assigned cluster.

To evaluate an entire clustering, the **average silhouette index** is calculated:

$$S = \frac{1}{N} \sum_{i=1}^{N} s(i),$$

where $N$ is the total number of data points. This number is interpreted as follows

- A high average silhouette score (close to 1) indicates that the clustering structure is well-defined, with distinct and cohesive clusters.
- A low or negative average silhouette score suggests poor clustering, with overlapping or improperly assigned clusters.

The silhouette index is widely used in clustering analysis due to its simplicity and interpretability, making it a valuable tool for comparing different clustering solutions or validating the results of an algorithm.

## 7.2 Genetic Algorithm Projective Discriminant Analysis

This section outlines the design and implementation of GPDA to optimize matrix configurations that enhance class separability in reduced dimensional spaces. The parameters chosen for the genetic algorithm, such as population size, number of generations, and rates of crossover and mutation will be in the experiments section.

### 7.2.1 Individuals and Selection

The initial population is generated with random matrices $A_{n\times 3}$ whose values are between -8 and 8, and the following repair process is performed to ensure orthonormality: let $A = [v^1, v^2, v^3]$ be a matrix, where $v_n^i$ represents the $n$-th coordinate in the $i$-th column. To ensure that the columns are orthogonal, the substitutions indicated in Equations (1), (2) and (3) are made.

$$v_n^2 = \frac{-\sum_{i=1}^{n} v_i^1 \cdot v_i^3}{v_n^1} \tag{1}$$

$$v_n^3 = \frac{-\sum_{i=1}^{n-1} v_i^2 v_i^3}{v_n^2} \tag{2}$$

$$v_{n-1}^3 = \frac{v_n^2 \sum_{i=1}^{n-2} v_i^1 v^3 i - v_n^1 \sum_{i=1}^{n-2} v_i^2 v_i^3}{v_{n-1}^2 v_n^1 - v_{n-1}^1 v_n^2} \tag{3}$$

As soon as these values are obtained, each vector $v^i$ is normalized. This is for the case where the projected space has 3 dimensions, for the 2 dimensional case we consider the Gram Schmidth ortonormalization process.

These equations are the result of calculating the dot product of each pair of distinct columns, equaling to 0, and solving the system of equations to satisfy the condition $v^i \cdot v^j = 0$ for $i \neq j$.

Selection will be Stochastich Universal Sampling (SUS), this is a method that allows all individuals, including those with lower fitness, to have a chance of being selected thereby helping to maintain genetic diversity within the population. Conceptually, it is equivalent to performing roulette with $\gamma$ slots. Markers are placed corresponding to the number of individuals to be selected and the roulette is spun only once. The individuals where the markers land are then selected [11], finally the survival criteria used is elitism, the offspring is included in the population and only the best survive to keep the same population size.

### 7.2.2 Crossover and Mutation

Once the parents are selected a single-point crossover is performed on each column vector of the individuals, it should be noted that there is no guarantee that the resulting new individuals will be viable solutions. Therefore, it is necessary to perform the **repair process** described by the Equations (1), (2) and (3).

Mutation is performed by randomly selecting a rotation axis, then generating a rotation matrix $R_{n\times n}$ along that axis with a random angle between 0 and $\frac{\pi}{5}$ to obtain the mutated individual $A' = A \times R$. Fig. 3 illustrates the structure of the matrix $R$ for the 3-dimensional case. The 2-dimensional mutation is almost identical, but there is no need to choose an axis, and the angle is chosen between $-\frac{\pi}{5}$ and $\frac{\pi}{5}$.

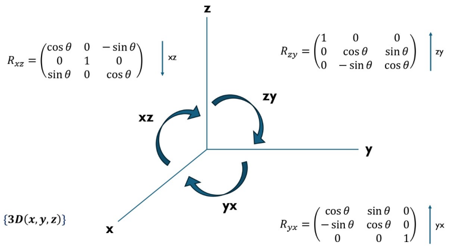

$R_{xz} = \begin{pmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix}$
xz
z
$R_{zy} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{pmatrix}$
zy
zy
xz
y
yx
$\{3D(x, y, z)\}$
x
$R_{yx} = \begin{pmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$
yx

Figure 3: Rotation matrixes $R$ depending on the selected axis (3-dimensional case).

Once the initial parameters are specified, the proposed method is described as pseudocode in the algorithm 1.

**Algorithm 1** GPDA

```
Input: initial population of pobsize orthonormal matrices
Output: Best individual found as the solution
while number of generations not reached do
    Select $\gamma$ parents using Stochastic Universal Sampling (SUS)
    foreach pair of parents do
        Perform single-point crossover on each column vector of the matrices
        Generate offspring
    foreach offspring generated do
        Repair the column vectors to maintain orthonormality
        Apply mutation with a certain probability by multiplying with a rotation matrix
        around a randomly selected axis with a random angle between 0 and $\pi/5$.
    Include the mutated offspring in the population
    Select individuals based on fitness to maintain a constant population size
return the best individual found as the solution
```

### 7.3 Datasets

Since the second objective involves comparing the results of LDA, GLDA and GPDA, different datasets with different properties are considered, see table 1:

| Database | Normal distribution | Identical covariance matrices | More observations than features | Expected outcome of LDA |
|---|---|---|---|---|
| 1 | Yes | Yes | Yes | Good projection |
| 2 | Yes | Yes | No | No eigenvalues/eigenvectors |
| 3 | Yes | No | Yes | Bad projection |
| 4 | Yes | No | No | No eigenvalues/eigenvectors |
| 5 | No | Yes | Yes | Bad projection |
| 6 | No | Yes | No | No eigenvalues/eigenvectors |
| 7 | No | No | Yes | Bad projection |
| 8 | No | No | No | No eigenvalues/eigenvectors |

Table 1: Comparison of LDA performance expected under different dataset conditions.

It has been challenging to find eight datasets that meet these specific requirements. Therefore, synthetic datasets were generated to conduct most of the experiments. In addition for datasets 7 and 8, where LDA fails to provide results, two suitable real-world datasets were identified: the Breast Cancer Wisconsin dataset and a normalized version of the Golub leukemia dataset.

The process to generate datasests **1** and **2** was to establish a suitable number of observations with respect to the dimensions, so the more observations than features condition of the reference table was met, three points were generated in a Euclidean space to form an equilateral triangle with sides of length 5 that function as the centroids of each of the classes. A symmetric matrix of dimensions according to the number of features was generated, with random values between 0 and 1, and the main diagonal was set to 1 to ensure that it represents a covariance matrix. The classes were generated from multivariate normal distributions with means at the vertices of the equilateral triangle and covariances according to the previously defined matrix, this process guarantees the presence of some intersection between the classes, maintaining moderate separability. For data sets **3** and **4** the process was similar but different covariance matrices were generated for each class.

For datasets **4** and **5** a number of dimensions and classes consistent with the other databases was established, the number of observations was modified to meet the condition of the “More observations than features” column. A symmetric random covariance matrix with values between 0 and 1 and main diagonal equal to 1 was generated, this matrix was used to guarantee identical covariances between classes. Three groups of samples were generated from a uniform distribution. The class means were 3 points forming an equilateral triangle of side 0.3 for database 5 and 0.15 for database 6, these points were used to center the distributions, maintaining a moderate level of overlap between classes. To ensure identical covariances between classes, each set of samples was transformed using the Cholesky decomposition of the generated covariance matrix.

Finally for datasets **7** and **8** three centroids were generated that form an equilateral triangle with sides of length 3, these points served as centers for the generation of three groups of samples distributed around them. The samples of each class were generated with different geometric distributions:

- **Class1**: Points uniformly distributed over the surface of a sphere of radius 1.5 centered on the first centroid.
- **Class2**: Random points within a sphere of radius 2 centered on the second centroid.
- **Class3**: Uniformly distributed points near the surface of a sphere of radius 1.5 centered on the third centroid, with a slight radial variation.

Low-magnitude Gaussian noise was added to all generated samples in order to introduce a slight overlap between classes.

Now to make a fair comparision between GLDA and GPDA three datasets used in [14] will be consider: Iris, Diabetes and Ionosphere, all of them are available in the UCI Machine Learning repository.

## 8 Experiments and Results

Since dimensionality reduction has been considered for 2- and 3-dimensional subspaces, two algorithm configurations were employed which, although sharing the same selection strategy and fitness function, can be considered as different variants. Ideally, the optimal parameters for both versions would be expected to be the same. However, when performing a grid search using databases 1 and 8, which represent extreme cases in terms of difficulty, some differences in the parameters obtained were observed.

To validate these differences, the nonparametric hypothesis test Wilcoxon Rank Sum was applied. The final parameters selected for each case were as follows:

Table 2: Parameters for 2D and 3D.

| **Parameters** | **3D** | **2D** |
|---|---|---|
| Number of generations | 2000 | 1500 |
| Number of parents | 20 | 15 |
| Population size | 250 | 150 |
| Crossover rate | 1.0 | 1.0 |
| Mutation rate | 0.09 | 0.09 |

These results reflect the need to adjust the parameters according to the dimensionality of the projected subspace, highlighting the differences in computational requirements between the 2D and 3D configurations. For the GLDA the parameters used were: 150 individuals, 10000 generations, 0.7 crossover rate, notice that mutation probabilities depend on this rate and all the experiments were conducted using the same parameters. The process was conducted using a diverse set of databases including:

- The 8 databases listed in table 1
- 3 of the databases used in the article [14]
- 2 representative examples of small sampling problem.

For each database, different versions with different feature space transformations were generated:

1. Original version, no transformation.
2. Projection using Linear Discriminant Analysis.
3. Projection to 2 dimensions using the proposed method (GPDA-2D)
4. Projection to 3 dimensions using the proposed method (GPDA-3D)
5. Projection using Genetic Linear Discriminant Analysis (GLDA)

Two main metrics will be applied to each of these versions for comparison:

- Silhouette index: to asses the quality of class separability yn the projected space
- Linear classifier accuracy: Obtained using a linear discriminant that searches for the optimal hyperplane that separates two classes

In the case of the versions generated with genetic algorithms, the following procedure was followed:

- 30 independent runs were performed using the parameters previously specified
- Among the results obtained, the run corresponding to the median fitness was selected as a representative of the algorithm's behavior.

This approach ensures that comparisons between methods are consistent and that the results reflect bot the quality of the projections and their impact on the classification. The results of applying this methodology on each data set are shown in the following subsections.

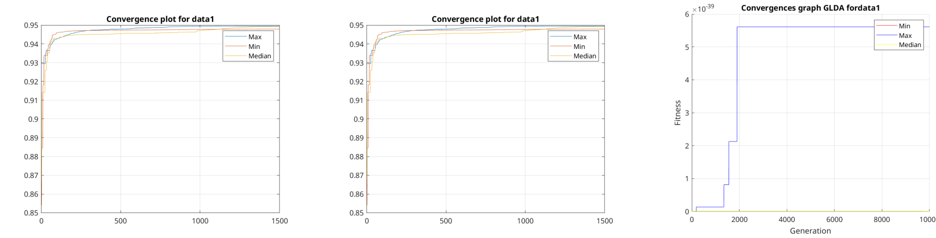
**At this stage of the work, initial analyses have been performed on the 13 selected data sets. The results obtained so far provide a basis for assessing the feasibility of the proposed approach and its performance in different scenarios. However, these results should be considered preliminary, as the analysis is still ongoing and both the methods and evaluation criteria are being refined. In the coming weeks, the detailed analysis is expected to be completed, incorporating additional refinements and cross-validations to strengthen the conclusions.**

## 8.1 Data1

This is a synthetic data set with ideal conditions for LDA: normal distribution and identical covariance matrices, with 3 balanced classes and 150 records. F The statistics of running the algorithms 30 times using the parameters mentioned can be found in table 3.

| Method | Mean | Std Dev | Max Fitness | Min Fitness | Median Fitness |
|---|---|---|---|---|---|
| GPDA 2D | 0.949 | $4 \times 10^{-4}$ | 0.949 | 0.947 | 0.949 |
| GPDA 3D | 0.933 | $3 \times 10^{-3}$ | 0.937 | 0.924 | 0.934 |
| GLDA | $1.46 \times 10^{-13}$ | $1 \times 10^{-39}$ | $5.6 \times 10^{-39}$ | $1.46 \times 10^{-45}$ | $9.66 \times 10^{-43}$ |

Table 3: Statistics for data1



((a)) Convergence for 30 runs

((b)) Scatter plot for the 2D case

((c)) Convergence plot GLDA

Figure 4: Convergence graphs for 30 runs, Maximum, Minimum and Median convergence graphs are shown

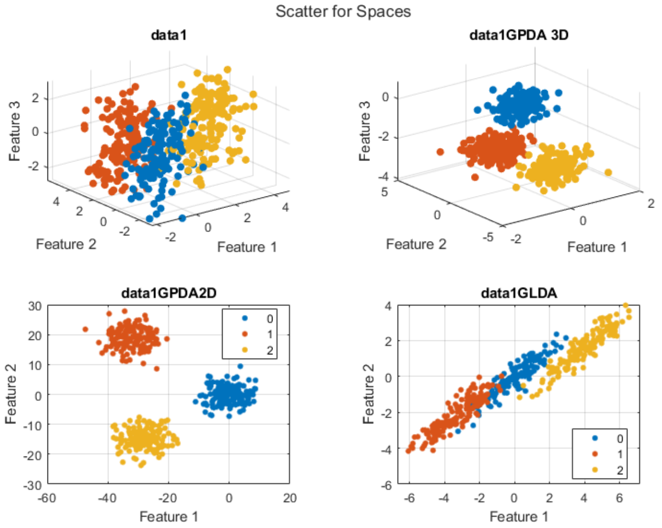


Figure 5: Projected spaces scatter plots for Data 1

## 8.2 Data 2

Synthetic database with 110 features, 30 records and 3 classes as described in table 1.

| Method | Mean | Std Dev | Max Fitness | Min Fitness | Median Fitness |
|---|---|---|---|---|---|
| GPDA 2D | 0.949 | $4 \times 10^{-4}$ | 0.949 | 0.947 | 0.949 |
| GPDA 3D | 0.933 | $3 \times 10^{-3}$ | 0.937 | 0.924 | 0.934 |
| GLDA | $1.46 \times 10^{-13}$ | $1 \times 10^{-39}$ | $5.6 \times 10^{-39}$ | $1.46 \times 10^{-45}$ | $9.66 \times 10^{-43}$ |

Table 4: Statistics for data2

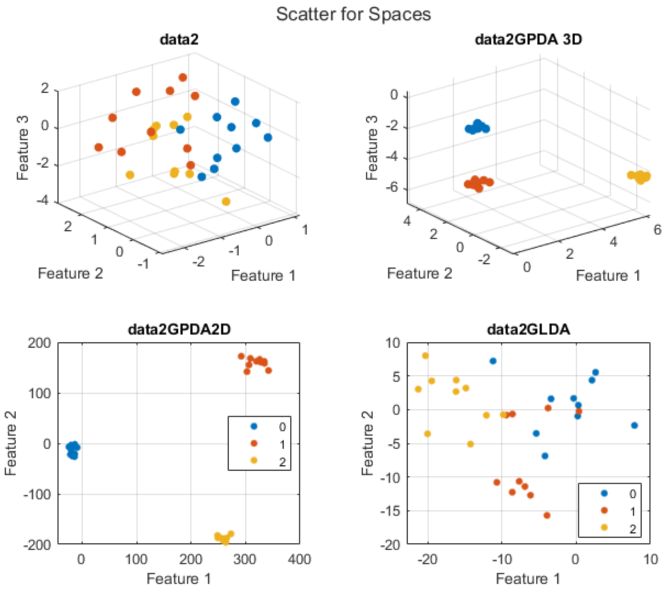
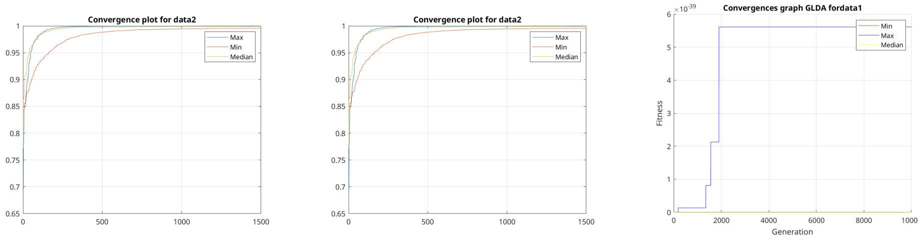


Figure 6: Projected spaces scatter plots for Data 2



((a)) Convergence for 30 runs

((b)) Scatter plot for the 2D case

((c)) Convergence plot GLDA

Figure 7: Convergence graphs for 30 runs, Maximum, Minimum and Median convergence graphs are shown

### 8.3 Data 3

Synthetic database with 8 features, 450 records and 3 classes as described in table 1.

| Method | Mean | Std Dev | Max Fitness | Min Fitness | Median Fitness |
|---|---|---|---|---|---|
| GPDA 2D | 0.82 | $3 \times 10^{-3}$ | 0.823 | 0.814 | 0.821 |
| GPDA 3D | 0.789 | $6 \times 10^{-3}$ | 0.797 | 0.778 | 0.789 |
| GLDA | $1.17 \times 10^{-72}$ | $5.67 \times 10^{-72}$ | $3.11 \times 10^{-71}$ | $2.62 \times 10^{-78}$ | $1.82 \times 10^{-75}$ |

Table 5: Statistics for data3

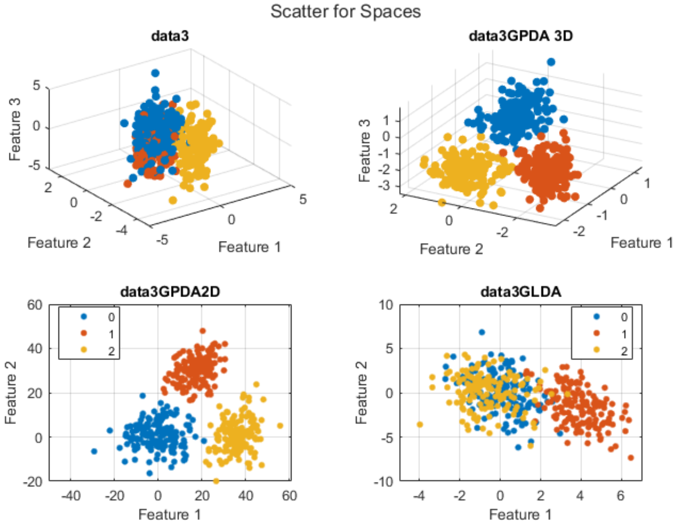
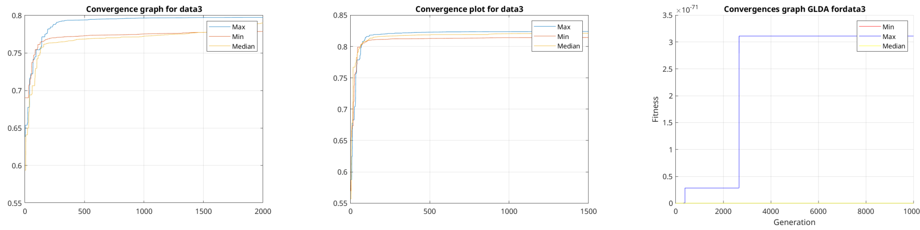


Figure 8: Projected spaces scatter plots for Data 3



((a)) Convergence for 30 runs ((b)) Scatter plot for the 2D case ((c)) Convergence plot GLDA

Figure 9: Convergence graphs for 30 runs, Maximum, Minimum and Median convergence graphs are shown

## 8.4 Data 4

Synthetic database with 110 features, 45 records and 3 classes as described in table 1.

| Method | Mean | Std Dev | Max Fitness | Min Fitness | Median Fitness |
|---|---|---|---|---|---|
| GPDA 2D | 0.985 | $4 \times 10^{-3}$ | 0.991 | 0.976 | 0.987 |
| GPDA 3D | 0.975 | $3 \times 10^{-3}$ | 0.981 | 0.968 | 0.975 |
| GLDA | 0 | 0 | 0 | 0 | 0 |

Table 6: Statistics for data4

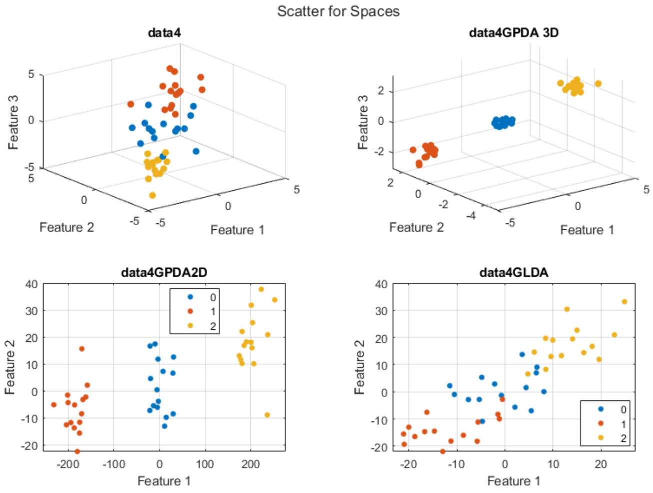
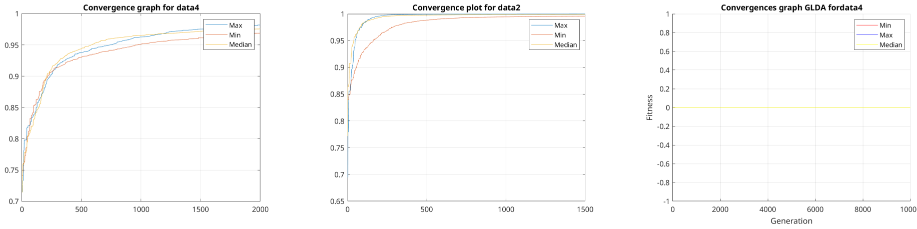


Figure 10: Projected spaces scatter plots for Data 4



((a)) Convergence for 30 runs

((b)) Scatter plot for the 2D case

((c)) Convergence plot GLDA

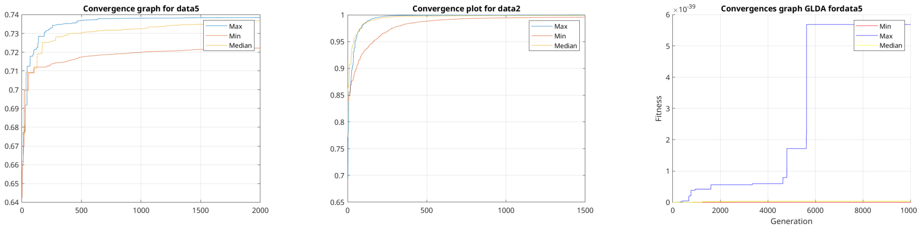
Figure 11: Convergence graphs for 30 runs, Maximum, Minimum and Median convergence graphs are shown

## 8.5 Data 5

Synthetic database with 6 features, 450 records and 3 classes as described in table 1.

| Method | Mean | Std Dev | Max Fitness | Min Fitness | Median Fitness |
|---|---|---|---|---|---|
| GPDA 2D | 0.77 | $1 \times 10^{-3}$ | 0.772 | 0.765 | 0.77 |
| GPDA 3D | 0.733 | $5 \times 10^{-3}$ | 0.738 | 0.722 | 0.736 |
| GLDA | $7 \times 10^{-40}$ | $1.49 \times 10^{-39}$ | $5.6 \times 10^{-39}$ | $1.46 \times 10^{-45}$ | $9.66 \times 10^{-43}$ |

Table 7: Statistics for data5



((a)) Convergence for 30 runs

((b)) Scatter plot for the 2D case

((c)) Convergence plot GLDA

Figure 12: Convergence graphs for 30 runs, Maximum, Minimum and Median convergence graphs are shown

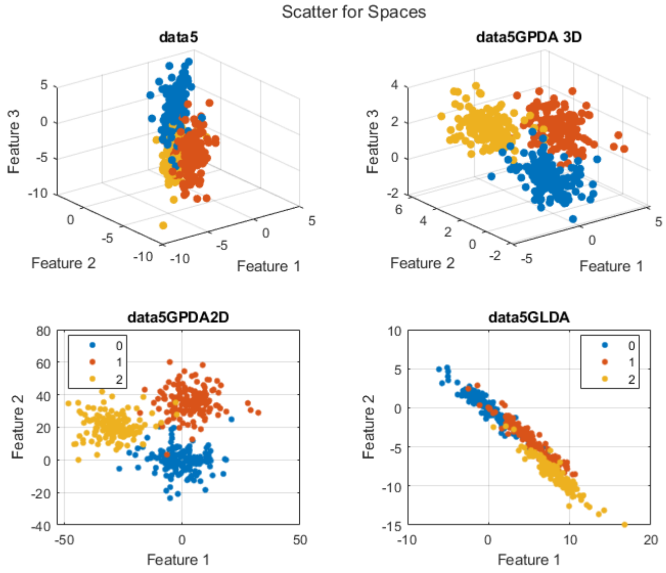


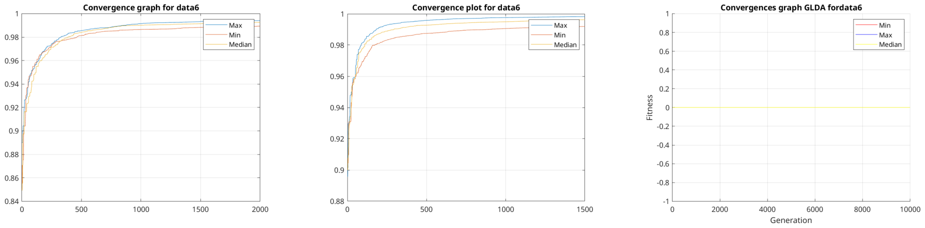
Figure 13: Projected spaces scatter plots for Data 5

## 8.6 Data 6

Synthetic database with 110 features, 45 records and 3 classes as described in table 1.

| Method | Mean | Std Dev | Max Fitness | Min Fitness | Median Fitness |
|---|---|---|---|---|---|
| GPDA 2D | 0.996 | $1 \times 10^{-3}$ | 0.998 | 0.991 | 0.996 |
| GPDA 3D | 0.932 | $1 \times 10^{-3}$ | 0.994 | 0.989 | 0.992 |
| GLDA | 0 | 0 | 0 | 0 | 0 |

Table 8: Statistics for data6



((a)) Convergence for 30 runs

((b)) Scatter plot for the 2D case

((c)) Convergence plot GLDA

Figure 14: Convergence graphs for 30 runs, Maximum, Minimum and Median convergence graphs are shown

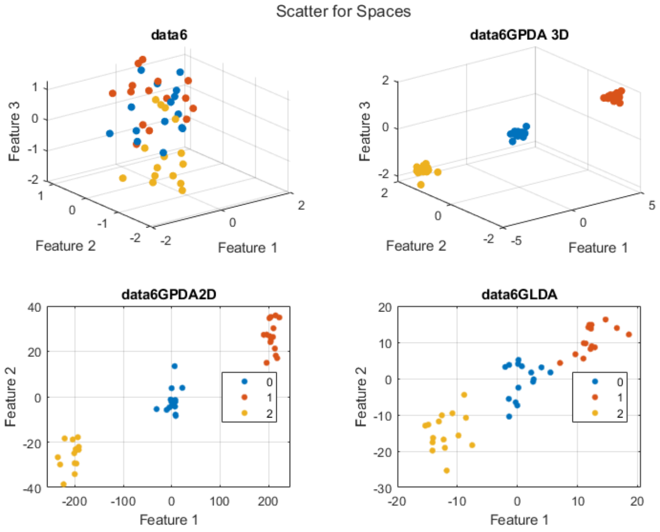


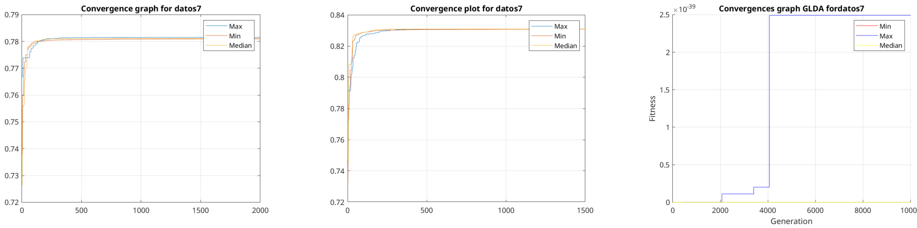
Figure 15: Projected spaces scatter plots for Data 6

## 8.7 Data 7

Synthetic database with 6 features, 450 records and 3 classes as described in table 1.

| Method | Mean | Std Dev | Max Fitness | Min Fitness | Median Fitness |
|---|---|---|---|---|---|
| GPDA 2D | 0.83 | $3.2 \times 10^{-5}$ | 0.83 | 0.83 | 0.83 |
| GPDA 3D | 0.78 | $2 \times 10^{-4}$ | 0.781 | 0.78 | 0.781 |
| GLDA | $9.2 \times 10^{-41}$ | $4.5 \times 10^{-40}$ | $2.48 \times 10^{-39}$ | $2.2 \times 10^{-47}$ | $1 \times 10^{-44}$ |

Table 9: Statistics for data7



((a)) Convergence for 30 runs

((b)) Scatter plot for the 2D case

((c)) Convergence plot GLDA

Figure 16: Convergence graphs for 30 runs, Maximum, Minimum and Median convergence graphs are shown

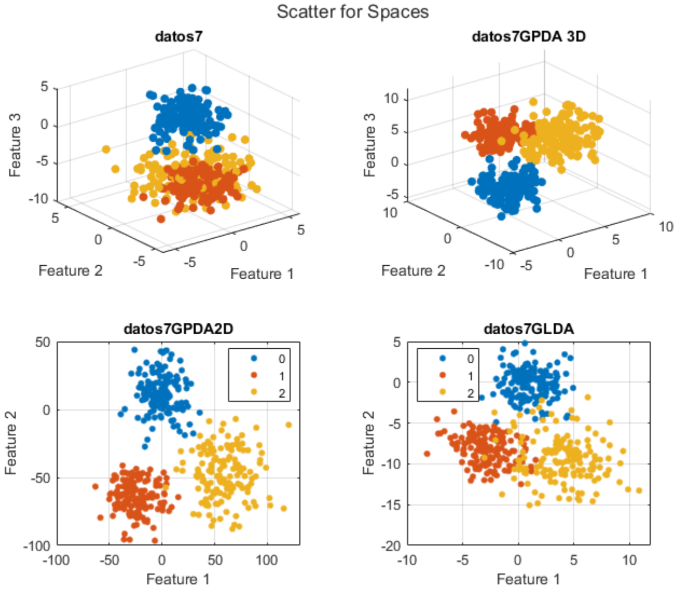


Figure 17: Projected spaces scatter plots for Data 7

## 8.8 Data 8

Synthetic database with 110 features, 45 records and 3 classes as described in table 1.

| Method | Mean | Std Dev | Max Fitness | Min Fitness | Median Fitness |
|---|---|---|---|---|---|
| GPDA 2D | 0.998 | $6 \times 10^{-4}$ | 0.999 | 0.996 | 0.998 |
| GPDA 3D | 0.997 | $2 \times 10^{-4}$ | 0.998 | 0.997 | 0.997 |
| GLDA | 0 | 0 | 0 | 0 | 0 |

Table 10: Statistics for data8

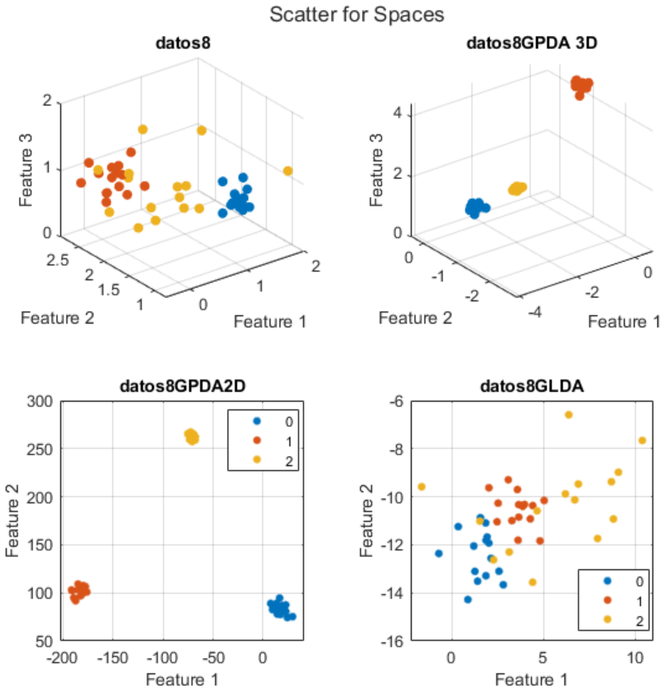
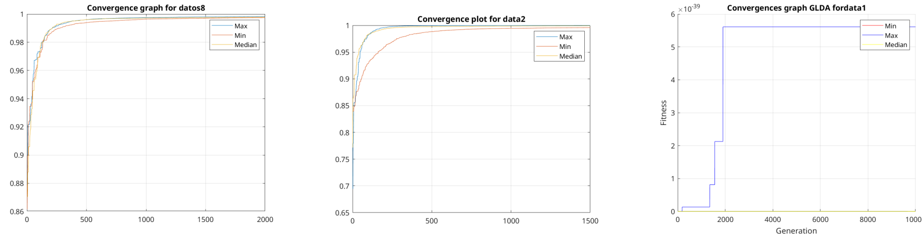


Figure 18: Projected spaces scatter plots for Data 8



((a)) Convergence for 30 runs

((b)) Scatter plot for the 2D case

((c)) Convergence plot GLDA

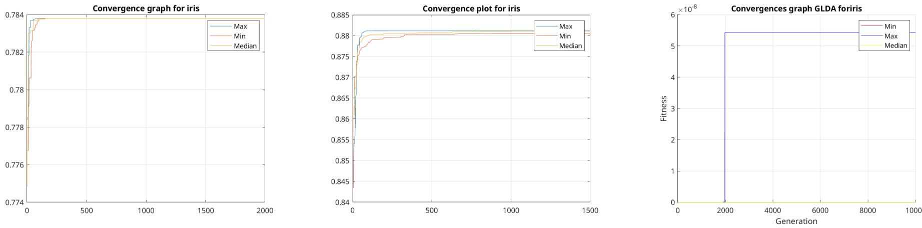
Figure 19: Convergence graphs for 30 runs, Maximum, Minimum and Median convergence graphs are shown

## 8.9 Iris

This is one of the earliest datasets used in the literature on classification methods and widely used in statistics and machine learning. The data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant. One class is linearly separable from the other 2; the latter are not linearly separable from each other.

| Method | Mean | Std Dev | Max Fitness | Min Fitness | Median Fitness |
|---|---|---|---|---|---|
| GPDA 2D | 0.905 | $4 \times 10^{-4}$ | 0.905 | 0.903 | 0.905 |
| GPDA 3D | 0.782 | $7.4 \times 10^{-6}$ | 0.782 | 0.782 | 0.782 |
| GLDA | $1.8 \times -9$ | $9.9 \times 10^{-9}$ | $5.4 \times 10^{-8}$ | $2.06 \times 10^{-17}$ | $2.06 \times 10^{-13}$ |

Table 11: Statistics for data1



((a)) Convergence for 30 runs

((b)) Scatter plot for the 2D case

((c)) Convergence plot GLDA

Figure 20: Convergence graphs for 30 runs, Maximum, Minimum and Median convergence graphs are shown

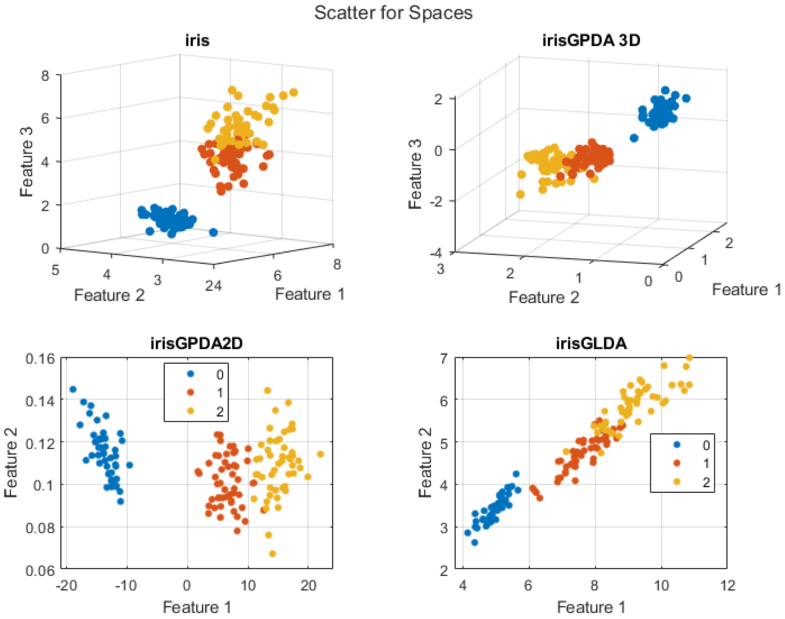


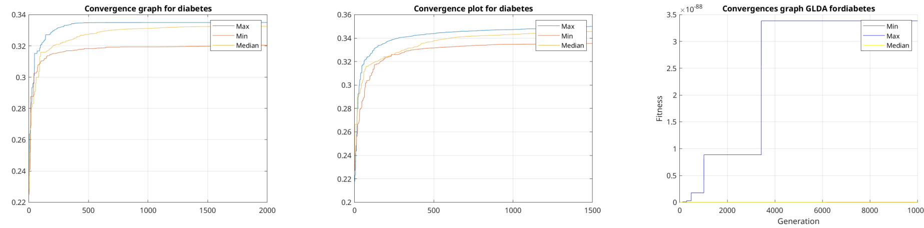
Figure 21: Scatter plots for projections

### 8.10 Diabetes

This dataset is originally from the National Institute of Diabetes and Digestive and Kidney Diseases. The objective is to predict based on diagnostic measurements whether a patient has diabetes.

| **Method** | **Mean** | **Std Dev** | **Max Fitness** | **Min Fitness** | **Median Fitness** |
|---|---|---|---|---|---|
| GPDA 2D | 0.344 | $4 \times 10^{-3}$ | 0.35 | 0.335 | 0.345 |
| GPDA 3D | 0.331 | $3 \times 10^{-3}$ | 0.334 | 0.32 | 0.332 |
| GLDA | $2.1 \times 10^{-89}$ | $6.6 \times 10^{-89}$ | $3.3 \times 10^{-88}$ | $3.7 \times 10^{-94}$ | $1.9 \times 10^{-91}$ |

Table 12: Statistics for Iris



((a)) Convergence for 30 runs ((b)) Scatter plot for the 2D case ((c)) Convergence plot GLDA

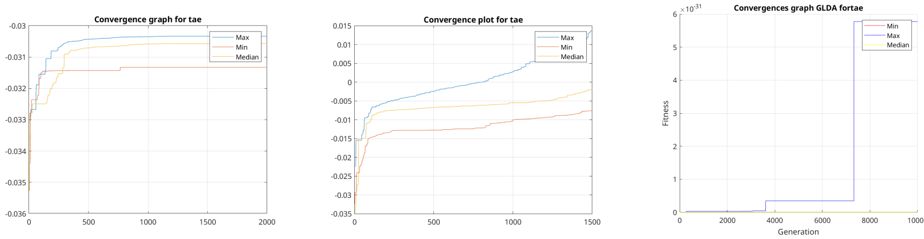
Figure 22: Convergence graphs for 30 runs, Maximum, Minimum and Median convergence graphs are shown

## 8.11 Teaching Assistant Evaluation

The data consist of evaluations of teaching performance over three regular semesters and two summer semesters of 151 teaching assistant (TA) assignments at the Statistics Department of the University of Wisconsin-Madison. The scores were divided into 3 roughly equal-sized categories ("low", "medium", and "high") to form the class variable.

| Method | Mean | Std Dev | Max Fitness | Min Fitness | Median Fitness |
|---|---|---|---|---|---|
| GPDA 2D | 0 | 0 | 0 | 0 | 0 |
| GPDA 3D | | | | | |
| GLDA | $2.1 \times -32$ | $1 \times 10^{-31}$ | $5.7 \times 10^{-31}$ | $2.1 \times 10^{-37}$ | $3.8 \times 10^{-35}$ |

Table 13: Statistics for T.A.E.



((a)) Convergence for 30 runs ((b)) Scatter plot for the 2D case ((c)) Convergence plot GLDA

Figure 23: Convergence graphs for 30 runs, Maximum, Minimum and Median convergence graphs are shown

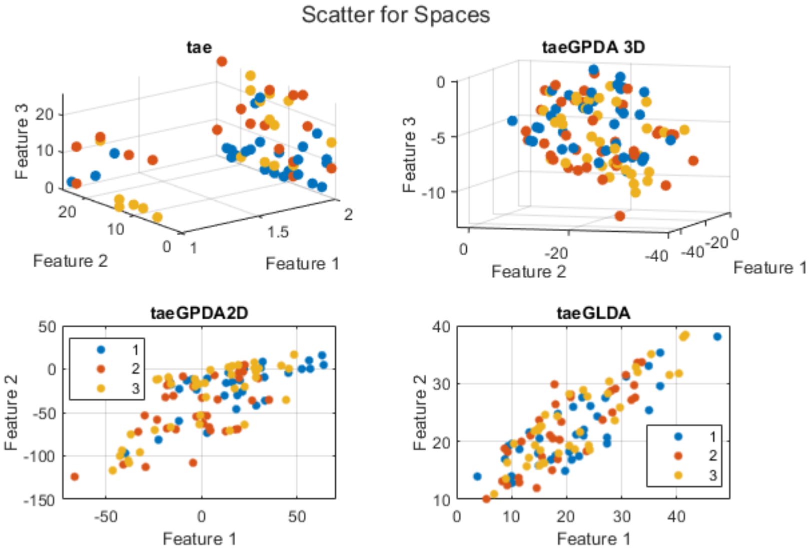


Figure 24: Rrojected spaces for T.A.E.

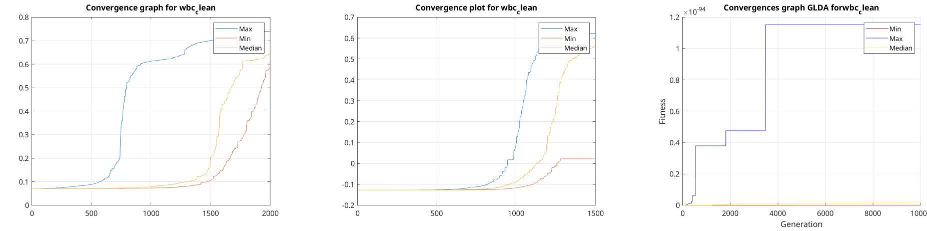
## 8.12 Wisconsin Breast Cancer

Features are computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. They describe characteristics of the cell nuclei present in the image. A few of the images can be found at http://www.cs.wisc.edu/ street/images/

Separating plane described above was obtained using Multisurface Method-Tree (MSM-T) [K. P. Bennett, "Decision Tree Construction Via Linear Programming." Proceedings of the 4th Midwest Artificial Intelligence and Cognitive Science Society, pp. 97-101, 1992], a classification method which uses linear programming to construct a decision tree.

| Method | Mean | Std Dev | Max Fitness | Min Fitness | Median Fitness |
|---|---|---|---|---|---|
| GPDA 2D | 0.949 | $4 \times 10^{-4}$ | 0.949 | 0.947 | 0.949 |
| GPDA 3D | 0.933 | $3 \times 10^{-3}$ | 0.937 | 0.924 | 0.934 |
| GLDA | $1.46 \times -13$ | $1 \times 10^{-39}$ | $5.6 \times 10^{-39}$ | $1.46 \times 10^{-45}$ | $9.66 \times 10^{-43}$ |

Table 14: Statistics for Wisconsin Breast Cancer dataset



((a)) Convergence for 30 runs

((b)) Scatter plot for the 2D case

((c)) Convergence plot GLDA

Figure 25: Convergence graphs for 30 runs, Maximum, Minimum and Median convergence graphs are shown

# 9 Following Work

The previous section details the databases used and the preliminary results obtained by applying the genetic algorithms. The next step in this work will be to determine the projected space using Linear Discriminant Analysis (LDA) and calculate the established metrics to compare the results. As part of future progress, a Table 15 will be completed to analyze and document the differences and similarities between the methods evaluated for each database.

| | Silhouette index | Discriminant Accuracy |
|---|---|---|
| Original | | |
| GPDA 2D | | |
| GPDA 3D | | |
| GLDA | | |
| LDA | | |

Table 15: Metrics for diferent transformations

# 10 Conclusions

The preliminary results obtained with the GPDA proposal are encouraging, since they show a good separability capacity between classes, with behaviors comparable to those expected when applying LDA and GLDA. In particular, superior performance in terms of separability was observed in databases 2, 6 and 8, which present the small sample size (SSS) problem. This behavior reinforces the hypothesis that optimization based on the silhouette index, instead of Fisher's criterion, is a viable alternative to address problems related to singular covariance matrices or complex eigenvalues, scenarios where LDA and GLDA tend to fail or show unsatisfactory results.

However, these results, although promising, should be interpreted with caution, since visualizations using scatter plots are not a sufficiently rigorous metric for evaluating dimensionality reduction methods, even when it is a preprocessing technique. Therefore, as part of future work, we contemplate the incorporation of a linear discriminant to identify the hyperplane or straight line that best separates each pair of classes present in the databases. The classification accuracy obtained after this evaluation will provide a more robust quantitative metric to validate the effectiveness of the proposed method.

In addition, the Golub database, known for its challenging problem regards sample size (SSS), was considered. However, the results obtained so far with this database have not been consistent enough to be presented in this report, suggesting the need for further analysis or additional adjustments in the methodology to deal with this type of more complex databases.

# References

[1] Lu, Juwei, K.N. Plataniotis, and A.N. Venetsanopoulos: *Face recognition using lda-based algorithms*. IEEE Transactions on Neural Networks, 14(1):195–200, 2003.

[2] Mao, Zhi Hua, Jian Hua Yin, Xue Xi Zhang, Xiao Wang, and Yang Xia: *Discrimination of healthy and osteoarthritic articular cartilage by fourier transform infrared imaging and fisher&#x2019;s discriminant analysis*. Biomed. Opt. Express, 7(2):448–453, Feb 2016. `https://opg.optica.org/boe/abstract.cfm?URI=boe-7-2-448`.

[3] Chiang, Leo H, Evan L Russell, and Richard D Braatz: *Fault diagnosis in chemical processes using fisher discriminant analysis, discriminant partial least squares, and principal component analysis*. Chemometrics and Intelligent Laboratory Systems, 50(2):243–252, 2000, ISSN 0169-7439. `https://www.sciencedirect.com/science/article/pii/S0169743999000611`.

[4] Fisher, R. A.: *The statistical utilization of multiple measurements*. Annals of Eugenics, 8:376–386, 1938.

[5] Belhumeur, P.N., J.P. Hespanha, and D.J. Kriegman: *Eigenfaces vs. fisherfaces: recognition using class specific linear projection*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(7):711–720, 1997.

[6] Liu, Qingshan, Xiaoou Tang, Hanqing Lu, and Songde Ma: *Face recognition using kernel scatter-difference-based discriminant analysis*. IEEE Transactions on Neural Networks, 17(4):1081–1085, 2006.

[7] Shen, Peng, Xugang Lu, Lemao Liu, and Hisashi Kawai: *Local fisher discriminant analysis for spoken language identification*. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5825–5829, 2016.

[8] Hastie, Trevor, Robert Tibshirani, and Jerome Friedman: *The Elements of Statistical Learning*. Springer Series in Statistics, 2009.

[9] James, Gareth, Daniela Witten, Trevor Hastie, Robert Tibshirani, *et al.*: *An introduction to statistical learning*, volume 112. Springer, 2013.

[10] Ghojogh, Benyamin, Fakhri Karray, and Mark Crowley: *Fisher and kernel fisher discriminant analysis: Tutorial*, 2022. `https://arxiv.org/abs/1906.09436`.

[11] Eiben, A. E. and J. E. Smith: *Introduction to Evolutionary Computing*. Springer, 2nd edition, 2003.

[12] Cui, Minshan, Saurabh Prasad, Majid Mahrooghy, Lori M. Bruce, and James Aanstoos: *Genetic algorithms and linear discriminant analysis based dimensionality reduction for remotely sensed image analysis*. In *2011 IEEE International Geoscience and Remote Sensing Symposium*, pages 2373–2376, 2011.

[13] Gharsellaoui, Soumaya, Sid Ahmed Selouani, and Mohammed Sidi Yakoub: *Linear discriminant differential evolution for feature selection in emotional speech recognition.* In *INTERSPEECH*, pages 3297–3301, 2019.

[14] Mohammadi, Mehdi, Bijan Raahemi, Ahmad Akbari, Babak Nassersharif, and Hossein Moeinzadeh: *Improving linear discriminant analysis with artificial immune system-based evolutionary algorithms*. Information Sciences, 189:219–232, 2012, ISSN 0020-0255. `https://www.sciencedirect.com/science/article/pii/S0020025511006177`.

[15] Zhao, Haitao, Zhihui Lai, Henry Leung, and Xianyi Zhang: *Linear Discriminant Analysis*, pages 71–85. Springer International Publishing, Cham, 2020, ISBN 978-3-030-40794-0. `https://doi.org/10.1007/978-3-030-40794-0_5`.

[16] Duda, Richard O, Peter E Hart, and David G Stork: *Pattern Classification*. John Wiley  Sons, 2001.

[17] Schölkopf, Bernhard, Alexander Smola, and Klaus Robert Müller: *Kernel principal component analysis*. In Gerstner, Wulfram, Alain Germond, Martin Hasler, and Jean Daniel Nicoud (editors): *Artificial Neural Networks — ICANN'97*, pages 583–588, Berlin, Heidelberg, 1997. Springer Berlin Heidelberg, ISBN 978-3-540-69620-9.