

# Genetic Lineal Discriminant Analysis GLDA

Víctor-David García-Medina<sup>1</sup>, Hector-Gabriel Acosta-Mesa<sup>1</sup>, Adriana-Laura López-Lobato<sup>1</sup>, and Efrén Mezura-Montes<sup>1</sup>

<sup>1</sup>Artificial Intelligence Research Institute, University of Veracruz  
Campus Sur, Calle Paseo, Lote II, Sección No 112, Nuevo Xalapa, 91097  
Xalapa-Enríquez Veracruz México  
`zs23000641@estudiantes.uv.mx`

**Abstract.** Linear Discriminant Analysis (LDA) is a technique used for dimensionality reduction that facilitates class separability in a projected subspace. In this paper a variant of LDA is presented, employing a genetic algorithm to explore the space of linear transformations. This approach seeks to find a projection matrix that not only reduces the dimensionality but also improves the linear separability of the classes avoiding the sensitivity to the natural distribution of the data and the presence of outliers. Two fitness functions are proposed to guide this genetic algorithm: the first one based on the maximization of the inter-cluster distance and the minimization of the intra-cluster distance, and the second one based on the accuracy obtained through the application of Fisher’s method for linear discriminant analysis (FDA).

**Keywords:** Genetic Algorithm · Evolutionary computation · Artificial intelligence · Machine Learning · Linear Discriminant Analysis

## 1 Introduction

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 employed for a variety of purposes, including facial recognition [9], discrimination of osteoarthritic cartilage from healthy cartilage [10], and classification of chemical processes [2]. 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 [1, 8] and spoken language identification [12], 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 [6, 7]. 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 [5]. This method is also used in face recognition to explore the geometry of colors distribution [8].

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 [3].

Given the potential of genetic algorithms to explore and optimize complex solution spaces, the idea of integrating them with KDA arose. This integration aims to overcome some of the limitations of standard KDA, especially in scenarios where the data structure is highly complex or nonlinear. By combining the robustness of KDA in feature projection with the global search capability of genetic algorithms, it is expected to develop a method less sensitive to distribution and variance constraints for dimensionality reduction and classification.

The remaining paper is structured into five sections. In section 2, a formal formulation of the problem is presented. Section 3 describes the structure of the genetic algorithm. Section 4 is dedicated to the experiments and the results obtained applying the proposed model to four different databases. Sections 5 and 6 offers detailed discussion and conclusions.

## 2 Formulation of the problem

The primary objective of this study is to represent data in a lower-dimensional space while maintaining the ability to distinguish between classes, which is crucial for effective classification. To achieve this, this work seeks to identify a transformation matrix  $A$  that optimizes the data. This transformation is essential for enhancing the performance of machine learning algorithms on complex datasets by reducing dimensionality and emphasizing features that are crucial for distinguishing between classes.

To address this challenge, it is proposed a methodological framework that not only seeks to find such matrix  $A$  but also ensures that the transformed space maximizes the discriminatory power between different classes.

Let's consider a database composed of  $m$  records, each with  $l$  variables and a label, forming a matrix  $B_{m \times (l+1)}$ . This database undergoes a polynomial expansion, a process that generates new features by computing all polynomial combinations of the original variables up to a certain degree. As a result, the expanded matrix becomes  $B'_{m \times (k+1)}$ , where  $l < k$ , reflecting the increased number of features. The challenge in dimensionality reduction lies in transforming this matrix  $B'$  into a new matrix  $D_{m \times (n+1)}$  with  $n < l$  and the labels remaining unchanged. The objective is to guarantee that this new representation preserves class separability, ensuring that the points corresponding to each row in  $D$ , along with their labels, are linearly separable. The key challenge is to select or transform features in a way that maximizes class distinction while maintaining the structural integrity of the original data. The objective is to identify a matrix  $A_{m \times n}$  that transforms the data  $B'$  to obtain the new matrix  $D$  with dimensionality reduction, i.e.  $B' \times A = D$ . It should be noted that the disadvantage of

using KDA is avoided as a result of this transformation, due to the fact that the mapping is explicit and the index of each record will remain the same.

This matrix  $A$  must be orthonormal so its columns represents an orthonormal basis of the subspace to be projected. Three vectors of dimension  $m$  are considered to form  $A$ . The fitness function  $f(A)$  will evaluate the separability of the data according to two approaches:

1.  $f(A) = \frac{eC}{iC}$ , where  $iC$  represents the intra-cluster distance, calculated as the average distance between the centroid of the cluster and its other elements, and  $eC$  represents the inter-cluster distance, calculated as the distance between the centroids. For multiple classes, the average of  $iC$  and  $eC$  will be considered.
2.  $f(A) = \text{Accuracy}_{FDA}(B \times A)$ , which is the accuracy obtained when applying FDA to the transformed database  $B$  using  $A$ .

The first of these functions is inspired by the basic principle of FDA: maximize distance between classes and minimize distance between elements of the same class [11]. For the first function the maximum value is unknown while for the second function the maximum value is 100. The following section outlines the process for identifying the required matrix  $A$  using a genetic algorithm.

### 3 Genetic Linear Discriminant Analysis algorithm

The genetic algorithm (GA) plays a crucial role in this study, since is the methodology selected for exploring potential solutions to the problem of optimizing the transformation matrix  $A$ . This section outlines the design and implementation of GLDA 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, are justified based on preliminary experiments and literature precedents.

The initial population is generated with random matrices 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  $v_m^i$  represents the  $m$ -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.

$$v_m^2 = \frac{-\sum_{i=1}^m v_i^1 \cdot v_i^3}{v_m^1} \quad (1)$$

$$v_m^3 = \frac{-\sum_{i=1}^{m-1} v_i^2 v_i^3}{v_m^2} \quad (2)$$

$$v_{m-1}^3 = \frac{v_m^2 \sum_{i=1}^{m-2} v_i^1 v_i^3 - v_m^1 \sum_{i=1}^{m-2} v_i^2 v_i^3}{v_{m-1}^2 v_m^1 - v_{m-1}^1 v_m^2} \quad (3)$$

As soon as these values are obtained, each vector  $v^i$  is normalized.

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  $i \neq j$ . Once the initial population has been established, the algorithm is run as described in the pseudo-code shown in Algorithm 1.

---

**Algorithm 1** GLDA

---

**Input:** initial population of *pobsize* orthonormal matrices

**Output:** Best individual found as the solution

```

1 while number of generations not reached do
2   Select  $\gamma$  parents using Stochastic Universal Sampling (SUS)
3   foreach pair of parents do
4     Perform single-point crossover on each column vector of the matrices
5     Generate offspring
6   foreach offspring generated do
7     Repair the column vectors to maintain orthonormality
8     Apply mutation with a certain probability by multiplying with a rotation
9     matrix around a randomly selected axis with a random angle between 0 and
10     $\pi/5$ .
11  Include the mutated offspring in the population
12  Select individuals based on fitness to maintain a constant population size
13 return the best individual found as the solution

```

---

Stochastic Universal Sampling (SUS) 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 [3].

Once the parents have been selected, they are crossed to generate offspring. 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  $R_{m \times m}$  along that axis with a random angle between 0 and  $\frac{\pi}{5}$  to obtain the mutated individual  $A' = A \times R$ . Fig. 1 illustrates the structure of the matrix  $R$  for the 3-dimensional case.

This genetic algorithm explores the solution space, searching for the transformation matrix with the best class separability under the criteria defined by the fitness function.

## 4 Experiments and results

This section presents an overview of the algorithm's performance, showcasing its efficacy in identifying optimal transformations that significantly enhance data classification tasks.

Diagram illustrating rotation matrices  $R$  for a 3D coordinate system  $\{3D(x, y, z)\}$ . The axes are labeled  $x$ ,  $y$ , and  $z$ . Rotations are indicated by curved arrows around each axis.

- Rotation around the  $z$ -axis:  $R_{xz} = \begin{pmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{pmatrix}$
- Rotation around the  $y$ -axis:  $R_{xy} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & \sin \theta \\ 0 & -\sin \theta & \cos \theta \end{pmatrix}$
- Rotation around the  $x$ -axis:  $R_{yx} = \begin{pmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$

Fig. 1: Rotation matrixes  $R$  depending on the selected axis (3-dimensional case).

Four datasets are used in these experiments: two synthetic datasets, the Iris dataset, and the RGB values of an image, where the classes are extracted from a binary mask. The scatter plots of these datasets are shown in the Fig. 2.

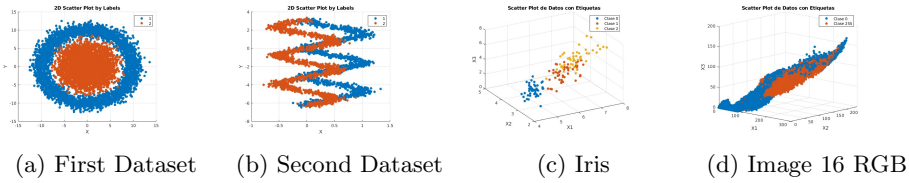


Fig. 2: Scatter plots for the diferent datasets

To assess the efficacy of the genetic algorithm, 30 runs were conducted to evaluate performance through statistical analysis.

The following subsections present the results of executing the two algorithms with the two fitness functions described in section 2.

#### 4.1 Dataset (a)

The 2-dimensional points were generated as follows: Class 1 consists of 3500 random points generated within a circle of radius 6.5, and Class 2 consists of 3500 points generated on a circle of radius 10. Both sets of points include Gaussian noise in their coordinates. For this dataset, the parameters used were: polynomial expansion of degree 2, initial population of 50 individuals, 15 parents, 35% mutation probability, 100% crossover probability, 100 generations for the first fitness function and 500 generations for the second one. The results of executing and recording the fitness of the best individual for each function are shown in Tables 1 and 2.

Table 1: Statistics for Fitness 1

Statistics	Value
Max value (Ex9)	4.7204
Min value (Ex24)	4.5937
Median (Ex20)	4.7182
Mean	4.7089
Standard deviation	0.025631

The convergence plots and projected data with the best individual for each fitness function are presented in Figures 3 and 4.

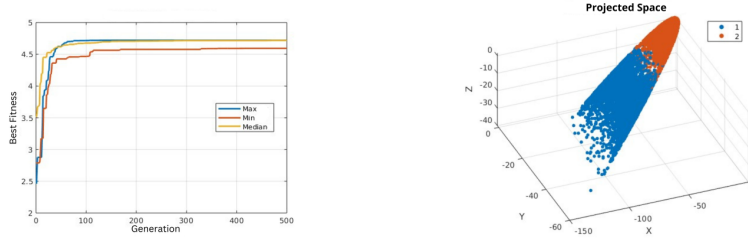


Fig. 3: Convergence plots and space resulting from projecting the best individual of the best run with the fitness function 1

Table 2: Statistics for fitness 2

Statistics	Value
Max value (Ex1)	99.357
Min value (Ex 12)	99.314
Median (Ex 1)	99.357
Mean	99.354
Standard deviation	0.008

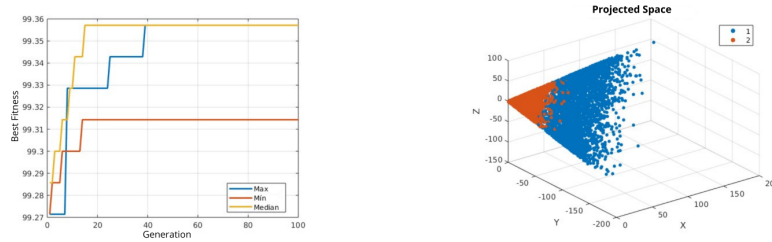


Fig. 4: Convergence plots and space resulting from projecting the best individual of the best run with the fitness function 2

## 4.2 Dataset (b)

This dataset consists of random points: for the first class, 3500 points are distributed along the curve  $\cos(t + 1) \sin(t^2)$ , and for the second class, 3500 points are distributed along the curve  $\cos^2(t)$ , with  $t \in [-2\pi, \pi]$  and Gaussian noise added to the  $y$  coordinate. For this dataset, the parameters used were: initial population of 50 individuals, 100 generations, 15 parents, 35% mutation probability, and 100% crossover probability.

The results obtained with this dataset are shown in Tables 3 and 4. Please refer to Figures 5 and 6 for the convergence plots and projected data with the best individual for each fitness function.

Table 3: Statistics for Fitness 1

Statistics	Value
Max value (Ex 18)	0.693
Min value (Ex 17)	0.3634
Median (Ex 26)	0.5032
Mean	0.510
Standard deviation	0.0732

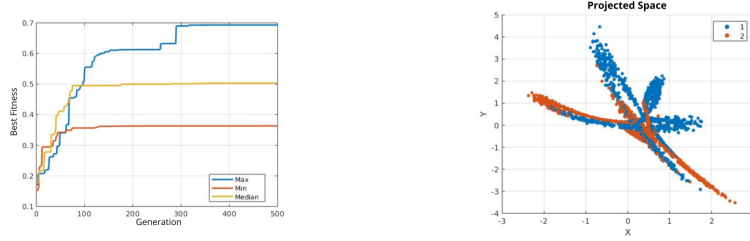


Fig. 5: Statistics, convergence plots and space resulting from projecting the best individual of the best run with the fitness function 1

Table 4: Statistics for fitness 2

Statistics	Value
Max value (Ex 17)	72.96
Min value (Ex 26)	71.36
Median (E 19)	71.93
Mean	72.03
Standard deviation	0.43

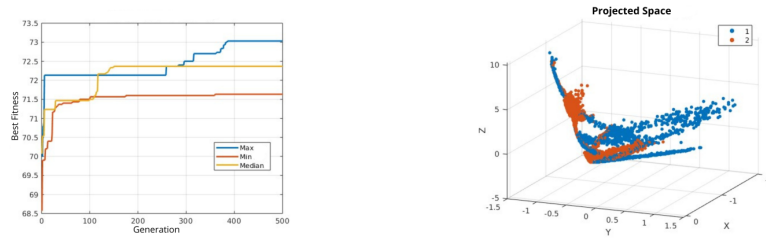


Fig. 6: Convergence plots and space resulting from projecting the best individual of the best run with the fitness function 2

### 4.3 Iris dataset

The iris dataset<sup>1</sup> comprises information about 3 distinct classes of iris plant, exhibiting 4 features and 50 instances of each class. For this dataset, the parameters used were: initial population of 50 individuals, 15 parents, 35% mutation

<sup>1</sup> <https://archive.ics.uci.edu/dataset/53/iris>

probability, 100% crossover probability, 100 generations for fitness 1 and 500 generations for fitness 2. The statistics from the 30 runs are shown in Tables 5 and 6. Figures 7 and 8 illustrate the convergence plots and projected data for the optimal individual for each fitness function.

Table 5: Statistics for fitness 1

Statistics	Value
Max value (Ex 25)	8.87
Min value (Ex 12)	8.439
Median (Ex 7)	8.541
Mean	8.57
Standard deviation	0.09

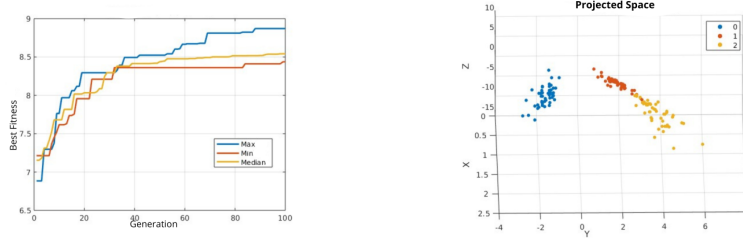


Fig. 7: Convergence plots and space resulting from projecting the best individual of the best run with the fitness function 1

Table 6: Statistics for fitness 2

Statistics	Value
Max value (Ex 1)	99.33
Min value (Ex 2)	98.66
Median (Ex1)	99.33
Meean	99.11
Standard deviation	0.3

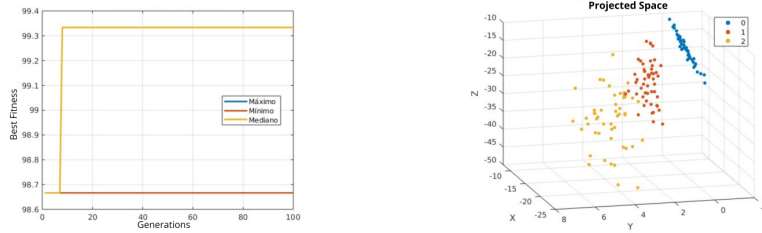


Fig. 8: Convergence plots and space resulting from projecting the best individual of the best run with the fitness function 2



#### 4.4 Dataset obtained from an RGB image

In these experiments, image 16 with an eyeball and its binary mask, which represent pixels denoting a vein, from the database “Retinal Vessel Segmentation”<sup>2</sup> were selected for analysis. The database was obtained by extracting the RGB values of each pixel from the image and concatenating them with the class from the binary mask, see Fig. 9.

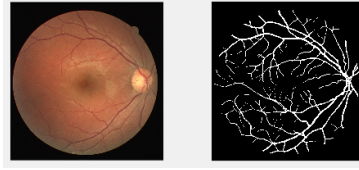


Fig. 9: Image and binary mask

The parameters used for these experiments are: 100 individuals, 25 parents, 100% crossover probability 35% mutation probability, 100 generations for fitness 1 and 500 generations for fitness 2. The results are shown in Tables 7 and 8. In Figures 10 and 11 the convergence plots and projected data with the best individual, for each fitness function, are presented.

Table 7: Statistics for fitness 1

Statistic	Value
Max value (Ex 23)	1.47
Min value (Ex 5)	1.33
Median (Ex 26)	1.44
Mean	1.434
Standard Deviation	0.03

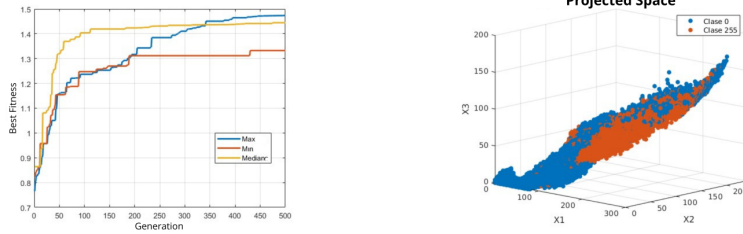


Fig. 10: Convergence plots and space resulting from projecting the best individual of the best run with the fitness function 1

<sup>2</sup> <https://drive.grand-challenge.org/DRIVE/>

Table 8: Statistics for fitness 2

Statistic	Value
Max value (Ex 3)	61.37
Min value (Ex 15)	61.29
Median (Ex 5)	61.3
Mean	61.31
Standard Deviation	0.02

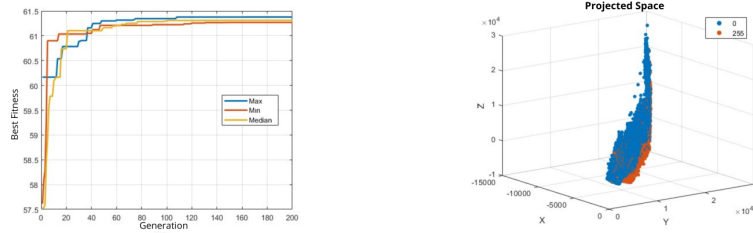


Fig. 11: Convergence plots and space resulting from projecting the best individual of the best run with the fitness function 2

## 5 Discussion

For this study, all datasets were subjected to a polynomial expansion of degree 2. This modification naturally increases the dimensionality of the space which facilitates the algorithm's ability to identify the most suitable projection for achieving effective separability. To evaluate the effectiveness of this approach, the accuracy obtained using FDA, which also employed a polynomial expansion of degree 2, was compared with the results derived from transforming the space using the best individual from the median run of each algorithm. The results are shown in Table 9.

Table 9: Comparison of results

	Fitness 1	Fitness 2	FDA
DB (a)	99.2	<b>99.35</b>	99.28
DB (b)	67.6	<b>72.9</b>	67.9
Iris	96.6	<b>99.33</b>	98.66
Image16	60.71	61.3	<b>61.4</b>

Figure 12 shows the subspace resulting from using FDA for the different datasets. For dataset (a), it can be observed that all three methods perform similarly well with accuracy values close to 99%. This indicates an excellent capacity for class separation for this particular dataset, where all methodologies prove to be effective.

In contrast, dataset (b) shows significant variability in the results. Fitness 1 achieves an accuracy of 67.6%, while Fitness 2 significantly improves this result

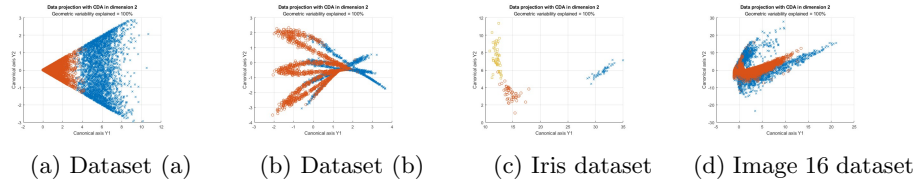


Fig. 12: Projected spaces resulting when applying FDA to datasets.

with 72.9%. However, FDA returns a value close to that obtained by Fitness 1 at 67.9%. This may suggest that, although the genetic algorithm with Fitness 2 configuration is more robust for this dataset, the FDA method does not fully capture the distinguishing features between classes, similar to Fitness 1.

The Iris dataset, known for its well-separated class characteristics, once again shows high results across all methods with an outstanding 99.33% for Fitness 2, followed closely by FDA and Fitness 1. This reinforces the idea that both genetic methods and linear analysis are suitable for datasets with clearly distinctive features.

Finally, for Image16, all three methods report significantly lower accuracy values compared to the other datasets with FDA achieving the highest accuracy at 61.4%. This may indicate that both the genetic algorithm configurations and FDA struggle to achieve clear separation in data that possibly presents greater overlap or less distinct class boundaries.

## 6 Conclusions

This paper presents a variant of LDA that employs a genetic algorithm to explore the space of linear transformations. This approach is designed to identify a projection matrix that not only reduces dimensionality but also enhances the linear separability of the classes. The datasets selected for this study were strategically chosen to cover a broad spectrum of scenarios: from situations where class separation is relatively easy, dataset (a), to more challenging contexts, dataset (b). Additionally, the classic Iris dataset results and analysis in a real-world dataset, data set from Image 16, were included to provide a comprehensive evaluation of the proposed algorithm under varied conditions.

The implemented algorithm demonstrated improvements in most cases compared to the FDA method, especially in the more challenging datasets. However, for Image16, the results were very similar to those obtained by FDA, indicating that the algorithm maintains its effectiveness even in scenarios with more complex and realistic data characteristics.

Despite the encouraging results, there are identified areas for improvement. For example, the polynomial expansion, which currently distributes data in a conic form could be revisited to explore new ways of relating variables. Additionally, while crossover preserves properties of the original orthonormal matrix vectors, there is still room for innovation in methods that increase the genetic diversity of the algorithm.

A promising line of research involves implementing a differential evolutionary alternative. This variant could benefit from the algebraic structure of matrices as it has a better geometric interpretation than combinatorial methods incorporating a rotation mutation with progressively decreasing angles in each generation. This adjustment would allow for a more detailed and precise exploration of the solution space, potentially leading to improvements in the algorithm's ability to adapt and optimize under different conditions.

## References

- [1] P.N. Belhumeur, 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.
- [2] Leo H Chiang, 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.
- [3] A. E. Eiben and J. E. Smith. *Introduction to Evolutionary Computing*. Springer, 2 edition, 2003.
- [4] R. A. Fisher. The statistical utilization of multiple measurements. *Annals of Eugenics*, 8:376–386, 1938.
- [5] Benyamin Ghogh, Fakhri Karay, and Mark Crowley. Fisher and kernel fisher discriminant analysis: Tutorial, 2022.
- [6] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics, 2009.
- [7] Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, et al. *An introduction to statistical learning*, volume 112. Springer, 2013.
- [8] Qingshan Liu, 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.
- [9] Juwei Lu, K.N. Plataniotis, and A.N. Venetsanopoulos. Face recognition using lda-based algorithms. *IEEE Transactions on Neural Networks*, 14(1):195–200, 2003.
- [10] Zhi-Hua Mao, 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's discriminant analysis. *Biomed. Opt. Express*, 7(2):448–453, Feb 2016.
- [11] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Kernel principal component analysis. In Wulfram Gerstner, Alain Germond, Martin Hasler, and Jean-Daniel Nicoud, editors, *Artificial Neural Networks — ICANN'97*, pages 583–588, Berlin, Heidelberg, 1997. Springer Berlin Heidelberg.
- [12] Peng Shen, Xugang Lu, Lema 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.