Linear Discriminent Analysis

Arpit Patel-AU1741074
Ratnam Parikh-AU1741036
Hardi Trivedi-AU1741092
Jainam Ajmera-AU1741051
Manav Shah-AU1741042

October 1, 2018

Abstract

The main aim of the project is to build library function for LDA which can be implemented on hardware. An input of 32x32 binary image matrix is considered here, accordingly two classes are created to separate pixel values of 0 and 1. Here the concept of within-class distance and between-class distance is taken into consideration. The final output here are all the eigenvectors of the matrix $S_w^{-1}S_B$. These eigenvectors can be further used for applications for image compression, pattern recognition etc.

Keywords:

1. Linear Discriments
2. Eigen Values
3. Eigen Vectors
4. Dimension Reduction
5. Classified

1 Introduction:

Definition Of LDA:
The goal of the LDA technique is to project the original data matrix onto lower dimensional space. In order to do so, three steps have to be performed.

1. To calculate the separability between different classes, which is called between-class variance.

2. To calculate the distance between mean and samples of each class, which is called within-class variance.

3. To construct Eigenvalues and Eigenvectors.

4. To construct Projection matrix from eigenvectors.

2 Literature Review:

LDA can be performed by two approaches. The two approaches are:

1. Class-Dependent Transformation

2. Class-Independent Transformation

2.1 Class-Dependent Transformation

This type of approach involves maximizing the ratio of between class variance to within class variance. The main objective is to maximize this ratio so that adequate class separability is obtained. The class-specific type approach involves using two optimizing criteria for transforming the data sets independently.

2.2 Class-Independent Transformation

This approach involves maximizing the ratio of overall variance to within class variance. This approach uses only one optimizing criterion to transform the data setsand hence all data points irrespective of their class identity are transformed using this transform. this type of LDA, each class is considered as a separate class against all other classes.

The mathematical operations needed for LDA:
Input : Dataset with C number of classes.
1:Solving for within-class variance matrix:

$$S_W = \frac{1}{N-1}\sum_{i=1}^{C} S_i$$

$$\text{where } S_i = \sum_{x\in\omega_i}(x-\mu_i)(x-\mu_i)^T$$

$$\text{and } \mu_i = \frac{1}{N_i}\sum_{x\in\omega_i} x$$

2:Solving for between-class variance matrix:

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

where $\mu = \frac{1}{N} \sum_{forallx} x = \frac{1}{N} \sum_{forallx} N_i\mu_i$

3:Generating EigenValues and EigenVectors

$$S_w^{-1} S_B w_i = \lambda_i w_i$$

where λ_i = Eigen Values of corresponding class and w_i = contains eigen vectors according to eigen values

4:Constructing Projection Matrix

W = eigenvector matrix of w_i

$$y = W^T x$$

where y is the lower dimensional feature space.

3 Comparision

Eigen Values In-Built	Eigen Values Project
4.7619	4.76194
0	0

Eigen Vectors In-Built
$w = \begin{bmatrix} 1.0000 & -0.0066 \\ 0.0057 & 1.0000 \end{bmatrix}$

Eigen Vectors Project
$w = \begin{bmatrix} 0.9999838 & -0.0065672 \\ 0.0056999 & 0.9999784 \end{bmatrix}$

4 Conclusion:

After the first simulation we observe that error between in-built code and our code is neglible or zero. This is because we just have two classes(0 and1) and thus we get only two eigenvectors corresponding to these classes. Another observation is that out of these vectors one with the maximum eigenvalue separates the data into two classes more precisely thus fulfilling the motive of LDA technique.

References

[1] Linear Discriminant Analysis - A Brief Tutorial, S. Balakrishnama, A. Ganapathiraju.

`https://www.isip.piconepress.com/publications/reports/1998/isip/lda/lda_theory.pdf`

[2] Linear discriminant analysis: A detailed tutorial, Alaa Tharwat, Tarek Gaber, Abdelhameed Ibrahim and Aboul Ella Hassanien.

`https://www.researchgate.net/publication/316994943_Linear_discriminant_analysis_A_detailed_tutorial`

[3] A Tutorial on Data Reduction, Shireen Elhabian and Aly A. Farag University of Louisville, CVIP Lab September 2009

[4] Feature Selection

https://www.scss.tcd.ie/Gerard.Lacey/Gerard_Lacey_Homepage/CS4053_Course_files/Lecture

[5] Face Recognition By Linear Discriminant Analysis, Suman Kumar Bhattacharyya, Kumar Rahul

http://interscience.in/IJCNS/_Vol2Iss2/31-35.pdf

[6] Animation Geometry Compression Using the Linear Discriminant Analysis, A.R.Vamisdhar, P.K.Bora, Sanjib Das

https://ieeexplore.ieee.org/document/4756113/authorsauthors

[7] Linear Discriminant Analysis, By Sebastian Raschka

https://sebastianraschka.com/Articles/2014_python_lda.html

[8] The QR Algorithm for Finding Eigenvectors, By Eric Mikida

https://cse.buffalo.edu/faculty/miller/Courses/CSE633/Eric-Mikida-Fall-2011.pdf

[9] Matlab Documentation

https://in.mathworks.com/help/matlab/