Data Science Project 2

Sparse Uncorrelated Linear Discriminant Analysis

(ULDA & SULDA)

Submitted To:

Mr. Surendra Gupta

Submitted By:

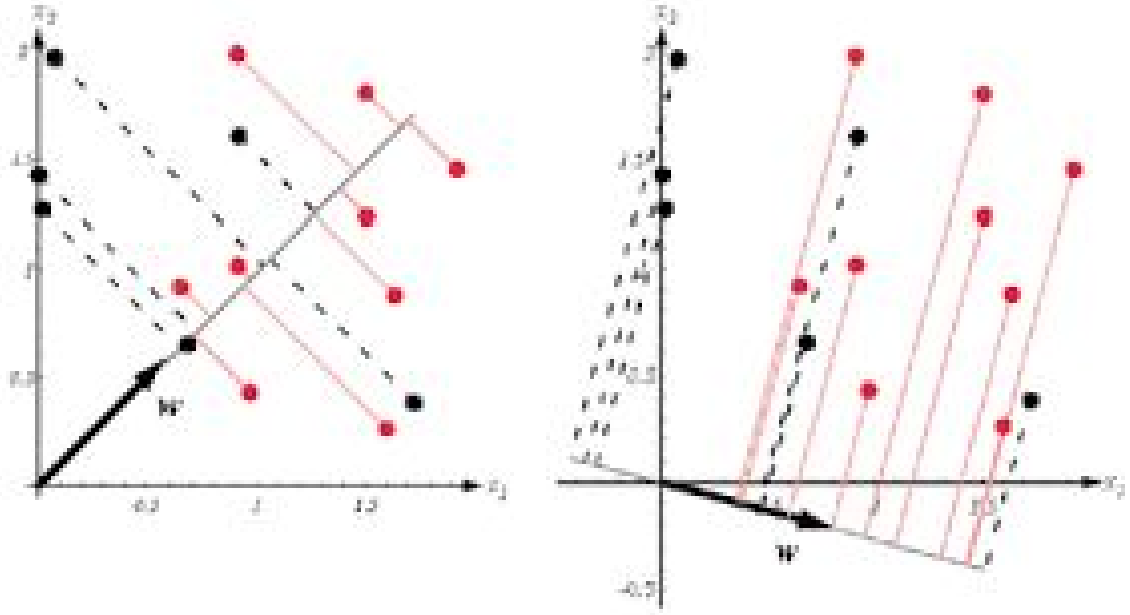
Mahak Jain
(0801CS171040)
Divyansh Joshi
(0801CS171023)

Contents

Introduction

Linear Discriminant Analysis (LDA):

Linear discriminant analysis is used as a tool for **classification, dimension reduction, and data visualization**. It has been around for quite some time now.
Despite its simplicity, LDA often produces robust, decent, and interpretable classification results. When tackling real-world classification problems, LDA is often the first and benchmarking method before other more complicated and flexible ones are employed.LDA provides class separability by drawing a decision region between the different classes.
LDA tries to maximize the ratio of the between-class variance and the within-class variance.



How is LDA different from PCA?

There are basically two techniques to reduce dimension of our dataset while maintaining the separation of classes, but they are different through following ways:

(i) PCA is an unsupervised algorithm. It ignores class labels altogether and aims to find the principal components that maximize variance in a given set of data. Linear Discriminant Analysis, on the other hand, is a supervised algorithm that finds the linear discriminants that will represent those axes which maximize separation between different classes.

(ii) Linear Discriminant Analysis often outperforms PCA in a multi-class classification task when the class labels are known. In some of these cases, however, PCA performs better. This is usually when the sample size for each class is relatively small. A good example is the comparisons between classification accuracies used in image recognition technology.

(ii) Many times, the two techniques are used together for dimensionality reduction. PCA is used first followed by LDA.

How LDA works?

(i) Calculate the separability between different classes. This is also known as between-class variance and is defined as the distance between the mean of different classes.

$$S_b = \sum_{i=1}^{g} N_i (\bar{x}_i - \bar{x})(\bar{x}_i - \bar{x})^T$$

Between Class Variable

(ii) Calculate the within-class variance. This is the distance between the mean and the sample of every class.

$$S_w = \sum_{i=1}^{g} (N_i - 1) S_i = \sum_{i=1}^{g} \sum_{j=1}^{N_i} (x_{i,j} - \bar{x}_i)(x_{i,j} - \bar{x}_i)^T$$

Within-Class Variable

(iii) Construct the lower-dimensional space that maximizes Step1 (between-class variance) and minimizes Step 2(within-class variance). In the equation below P is the lower-dimensional space projection. This is also known as Fisher's criterion.

$$P_{lda} = \arg\max_{P} \frac{\left| P^T S_b P \right|}{\left| P^T S_w P \right|}$$

Drawbacks of LDA:

One deficiency is that the classical LDA can not be applied directly to under- sampled problems, that is, the dimension of the data space is larger than the number of data samples, due to singularity of the scatter matrices; the other is the lack of sparsity in the LDA solution.

Although LDA is one of the most common data reduction techniques, it suffers from two main problems: the Small Sample Size (SSS) and linearity problems. In the next two subsections, these two problems will be explained, and some of the state-of-the-art solutions are highlighted.

Linearity problem LDA technique is used to find a linear transformation that discriminates between different classes. However, if the classes are non-linearly separable, LDA can not find a lower dimensional space. In

other words, LDA fails to find the LDA space when the discriminatory information are not in the means of classes.

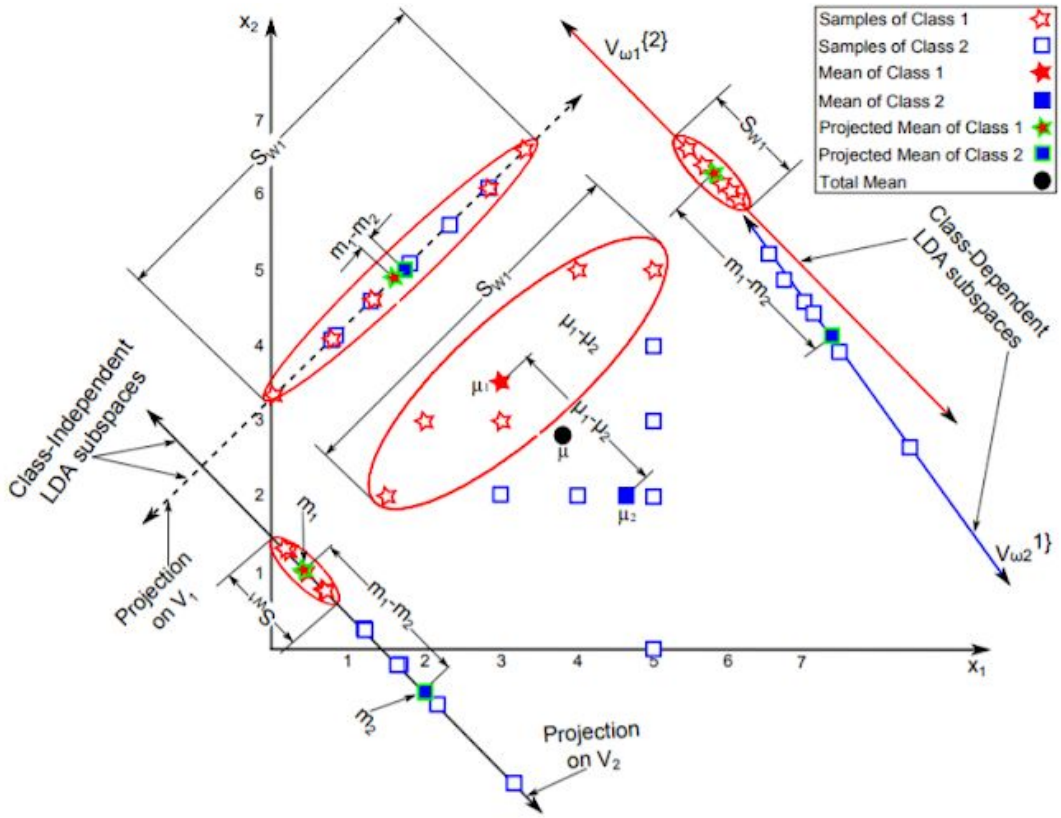
Linearity problem:

LDA technique is used to find a linear transformation that discriminates between different classes. However, if the classes are non-linearly separable, LDA can not find a lower dimensional space.

In other words, LDA fails to find the LDA space when the discriminatory information are not in the means of classes.

The discriminatory information does not exist in the mean, but in the variance of the data. This is because the means of the two classes are equal.

The mathematical interpretation for this problem is as follows: if the means of the classes are approximately equal, so the SB and W will be zero. Hence, the LDA space cannot be calculated.



An example of the two different methods of LDA methods.

- The blue and red lines represent the first and second eigenvectors of the class-dependent approach, respectively

- While the solid and dotted black lines represent the second and first eigenvectors of class-independent approach, respectively.

The second problem we encounter with LDA is:

Small Sample Size Problem:

Singularity, Small Sample Size (SSS), or undersampled problem is one of the big problems of LDA technique.

This problem results from high-dimensional pattern classification tasks or a low number of training samples available for each class compared with the dimensionality of the sample space.

Since all the other techniques use some kind of approximation.
the greedy algorithms ESLDA and GSLDA, the Penalized LDA (PLDA), and the Sparse Discriminant Analysis (SLDA).

Almost all existing sparse LDA algorithms introduce sparsity by adding 1 penalty (i.e., Lasso penalty) or its variants of the transformation matrix to objective functions, and thus, the computed sparse transformation is **not a solution of LDA but an approximation**.

Uncorrelated Linear Discriminant Analysis(ULDA):

Problems of LDA resolved by ULDA:

One deficiency is that the classical LDA can not be applied directly to undersampled problems, that is, the dimension of the data space is larger than the number of data samples, due to singularity of the scatter matrices.

To overcome the first problem, many extensions of the classical LDA have been proposed. These extensions include uncorrelated LDA (ULDA), regularized LDA, GSVD-based LDA (LDA/GSVD), and least squares LDA. Of these approaches, ULDA has an advantage over other approaches, that is, the feature vectors extracted by ULDA are mutually uncorrelated in the low-dimensional space.

This property is highly desirable for feature extraction in many applications in order to contain minimum redundancy.

Mathematical formulation:

In discriminant analysis, the between-class scatter matrix Sb, within-class scatter matrix Sw and total scatter matrix St are defined as:

$$S_b = \frac{1}{n}\sum_{i=1}^{k} n_i (c^{(i)} - c)(c^{(i)} - c)^T,$$

$$S_w = \frac{1}{n}\sum_{i=1}^{k} \sum_{a_j \in \mathcal{A}_i} (a_j - c^{(i)})(a_j - c^{(i)})^T,$$

$$S_t = \frac{1}{n}\sum_{j=1}^{n} (a_j - c)(a_j - c)^T,$$

Moreover let,

$$H_b = \frac{1}{\sqrt{n}}[\sqrt{n_1}(c^{(1)} - c) \ \cdots \ \sqrt{n_k}(c^{(k)} - c)] \in \mathbf{R}^{m \times k},$$

$$H_w = \frac{1}{\sqrt{n}}[A_1 - e^{(1)}e_1^T \ \cdots \ A_k - c^{(k)}e_k^T] \in \mathbf{R}^{m \times n},$$

$$H_t = \frac{1}{\sqrt{n}}[a_1 - c \ \cdots \ a_n - c] = A - ce^T \in \mathbf{R}^{m \times n},$$

Then the scatter matrices can be expressed as:

$$S_b = H_b H_b^T, \quad S_w = H_w H_w^T, \quad S_t = H_t H_t^T$$

Trace of the scatter matrices can be used to measure the quality of the class structure, where Trace(Sb) measures the distance between classes and Trace(Sw) measures the closeness of the data within the classes over all k classes.

In the low-dimensional space mapped by the linear transformation between-class, within class and total scatter matrices are of the forms:

$$S_b^L = G^T S_b G, \quad S_w^L = G^T S_w G, \quad S_t^L = G^T S_t G.$$

An optimal transformation GT should maximize Trace(S L b) and minimize Trace(S L w) simultaneously, which results in a common criterion for classical LDA:

$$G^* = \arg\max_G \{\mathrm{Trace}((S_t^L)^{-1} S_b^L)\}.$$

To deal with the singularity of St, many generalizations of classical LDA have been proposed. One popular generalization is the generalized ULDA

$$G^* = \arg \max_{G^T S_t G = I} \text{Trace}((S_t^L)^{(+)} S_b^L)$$
$$= \arg \max_{G^T S_t G = I} \text{Trace}(S_b^L),$$

Algorithm:

Algorithm 1: Uncorrelated LDA

Input: data matrix A

Output: transformation matrix G

1. Form three matrices H_b, H_w, and H_t as in Eq. (2);
2. Compute reduced SVD of H_t as $H_t = U_1 \Sigma_t V_1^T$;
3. $B \leftarrow \Sigma_t^{-1} U_1^T H_b$;
4. Compute SVD of B as $B = P \Sigma Q^T$; $q \leftarrow \text{rank}(B)$;
5. $X \leftarrow U_1 \Sigma_t^{-1} P$;
6. $G \leftarrow X_q$;

Here A is the input data matrix and SVD is defined as Singular Value Decomposition and can be easily found as from scipy.linalg import svd In python.

Sparse Uncorrelated Linear Discriminant Analysis (SULDA):

Problems of LDA resolved by SULDA:

Sparsity in the LDA solution is generally desirable for **high-dimensional data analysis** as it makes the inter- pretation of the extracted features much easier.

For LDA, each extracted feature in the transformed space is a linear combination of all the features of original data and the coefficients of such linear combination are generally nonzero, which makes the interpretation of the extracted features difficult.

Almost all existing sparse LDA algorithms introduce sparsity by adding l1 penalty or its variants of the transformation ma- trix to objective functions, and thus, the computed sparse transformation is not a solution of LDA but an approximation.

Sparse ULDA (SULDA) extracts mutually uncorrelated features and computes sparse LDA transformation, simultaneously.

SULDA is an improvement over ULDA where we find the sparse solution directly of the ULDA **solution matrix** by finding the l1- norm solution from all the solutions with minimum dimension (using Accelerated Linearized Bregman method).

Mathematical formulation:

Note that G is a minimum dimension solution of the optimization problem if and only if equality of (7)

> **Corollary 2.** *$G \in \mathbf{R}^{m\times l}$ is a minimum dimension solution of the optimization problem (3) if and only if $l = q$ and*
>
> $$G = (U_1\Sigma_t^{-1}P_1 + \mathcal{M}_2)\mathcal{Z}, \qquad (7)$$
>
> *where $\mathcal{M}_2 \in \mathbf{R}^{m\times q}$ is any matrix satisfying $\mathcal{M}_2^T U_1 = 0$ and $\mathcal{Z} \in \mathbf{R}^{q\times q}$ is orthogonal.*

holds, which is equivalent to

$$U_1^T G = \Sigma_t^{-1}P_1\mathcal{Z},\ \mathcal{Z}^T\mathcal{Z} = I.$$

The main idea of our sparse ULDA algorithm is to find the sparsest solution of ULDA from all G satisfying the above equation

A natural way to do this is to find a matrix G that minimizes the lo-norm (cardinality). Howev- er, lo-norm is non-convex and NP-hard. Therefore, in our sparse ULDA, we replace the lo-norm with its convex relaxation l1-norm, which results in the following problem

$$\begin{aligned} G^* = \arg\min_{G\in\mathbf{R}^{m\times q}} \quad & \|G\|_1 \\ s.t. \quad & U_1^T G = \Sigma_t^{-1}P_1\mathcal{Z},\ \mathcal{Z}^T\mathcal{Z} = I \end{aligned}$$

where $\|G\|_1$ is defined as $\|G\|_1 = \sum_{i=1}^{m}\sum_{j=1}^{q}|G_{ij}|$.

Here, Z ∈ Rq×q is **orthogonal**

When q = 1, the l1-minimization problem above equation is reduced to the basis pursuit problem which is then solved by **Accelerated Linearized Bregman method,** which is

$$\begin{aligned} x^{k+1} &= \delta \mathcal{S}_\mu(\tilde{v}^k), \\ v^{k+1} &= \tilde{v}^k - \tau \mathcal{A}^T(\mathcal{A}x^{k+1} - b), \ k \geq 0, \\ \tilde{v}^{k+1} &= \alpha_k v^{k+1} + (1-\alpha_k)v^k, \end{aligned}$$

where:

v ˜o = vo = τATb *, δ, μ and τ are positive parameters*

αk = (2k+3)/(k+3)

Sμ(·) is the componentwise k+3 soft-thresholding operator

$$\mathcal{S}_\mu(x) = \text{sign}(x) \odot \max\{|x| - \mu, 0\}.$$

Applying this Bregman's method for basic pursuit problem we get

$$\begin{aligned} G^{k+1} &= \delta \mathcal{S}_\mu(\tilde{V}^k), \\ V^{k+1} &= \tilde{V}^k - \tau U_1(U_1^T G^{k+1} - \Sigma_t^{-1} P_1 \mathcal{Z}), \qquad (12) \\ \tilde{V}^{k+1} &= \alpha_k V^{k+1} + (1-\alpha_k)V^k, \end{aligned}$$

Where $\tilde{V}^0 = V^0 = \tau U_1 \Sigma_t^{-1} P_1 \mathcal{Z}$

Algorithm:

Algorithm 1 Sparse ULDA (SULDA)

Input: data $A \in \mathbf{R}^{m \times n}$ and tolerance $\epsilon > 0$
Compute the reduced SVDs (4) and (5)
Let $\mathcal{Z} = I_q$, $\tilde{V}^0 = V^0 = \tau U_1 \Sigma_t^{-1} P_1 \mathcal{Z}$
repeat
  Compute G^{k+1} by (12)
  $error = \|U_1^T G^{k+1} - \Sigma_t^{-1} P_1 \mathcal{Z}\|_F$
until $error \leq \epsilon$
Output: G^{k+1}

(4) & (5) are,

$$H_t = U_1 \Sigma_t V_1^T, \tag{4}$$

where $U_1 \in \mathbf{R}^{m \times \gamma}$ and $V_1 \in \mathbf{R}^{n \times \gamma}$ are column orthogonal, and $\Sigma_t \in \mathbf{R}^{\gamma \times \gamma}$ is diagonal and nonsingular with $\gamma = rank(H_t) = rank(S_t)$. Next, let the reduced SVD of $\Sigma_t^{-1} U_1^T H_b$ be

$$\Sigma_t^{-1} U_1^T H_b = P_1 \Sigma_b Q_1^T, \tag{5}$$

where $P_1 \in \mathbf{R}^{\gamma \times q}$, $Q_1 \in \mathbf{R}^{k \times q}$ are column orthogonal, $\Sigma_b \in \mathbf{R}^{q \times q}$ is diagonal and nonsingular. Then $q = rank(H_b) = rank(S_b)$, and G is a solution of the optimization problem (3) if and only if $q \leq l \leq \gamma$ and

Examples:

Input:

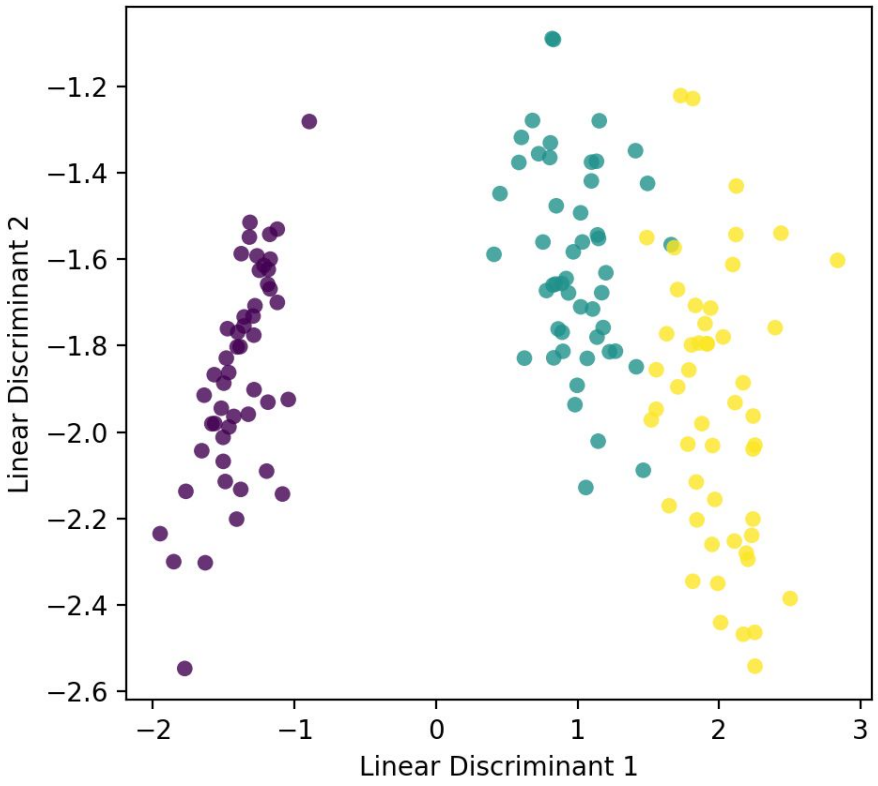
We used the famous Iris Dataset on our model.
The ***Iris* flower data set** or **Fisher's *Iris* data set** is a multivariate data set. It is sometimes called **Anderson's *Iris* data set** because Edgar Anderson collected the data to quantify the morphologic variation of *Iris* flowers of three related species.

It includes three iris species with 50 samples each as well as some properties about each flower. One flower species is linearly separable from the other two, but the other two are not linearly separable from each other.

Example:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
108	109	6.7	2.5	5.8	1.8	Iris-virginica
139	140	6.9	3.1	5.4	2.1	Iris-virginica
10	11	5.4	3.7	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
132	133	6.4	2.8	5.6	2.2	Iris-virginica
99	100	5.7	2.8	4.1	1.3	Iris-versicolor
140	141	6.7	3.1	5.6	2.4	Iris-virginica
1	2	4.9	3.0	1.4	0.2	Iris-setosa
107	108	7.3	2.9	6.3	1.8	Iris-virginica
42	43	4.4	3.2	1.3	0.2	Iris-setosa

Output:



Iris-Setosa, Iris-Versicolour and Iris-Virginica are all the possible three types of classification for our data and it is effectively distinguishable.

Hence, **the dimension is reduced without any approximations using ULDA and SULDA**.

Learning Outcome:

1. We developed SULDA, an efficient algorithm that performs sparse uncorrelated LDA, based on the characterization of solutions of generalized ULDA.

2. ULDA has the property that the features in the reduced space are uncorrelated

3. Based on the characterization we incorporate sparsity into the transformation matrix by selecting the solution with minimum l1-norm from all minimum dimension solutions of ULDA. The resulting l1-minimization problem is solved by the accelerated linearized Bregman method.

4. Different from existing sparse LDA algorithms, SULDA seeks a sparse solution directly from the solution set of ULDA. Thus, the computed sparse trans- formation is a solution of ULDA, instead of an approximation.

5. We learned that features extracted by SULDA are mutually uncorrelated, which ensures minimum redundancy in the low-dimensional space.

References:

[1] Delin Chu, Xiaowei Zhang, “Sparse Uncorrelated Linear Discriminant Analysis” in Department of Mathematics, National University of Singapore, Singapore 119076

[2] Jieping Ye, Tao Li, Tao Xiong, and Ravi Janardan, “Using Uncorrelated Discriminant Analysis for Tissue Classification with Gene Expression Data” in IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 1, no. 4, October-December 2004

[3] Jieping Ye, “Characterization of a Family of Algorithms for Generalized Discriminant Analysis on Undersampled Problems” in Journal of Machine Learning Research 6 (2005) 483–502, Minneapolis, Published 4/05

[4] Dalin Yuan a , Yizeng Liang a, ∗, Lunzhao Yi a , Qingsong Xu b , Olav M. Kvalheim c, “Uncorrelated linear discriminant analysis (ULDA): A powerful tool for exploration of metabolomics data” in Uncorrelated linear discriminant analysis (ULDA): A powerful tool for exploration of metabolomics data