Big Data Analysis
(MA60306)

Bibhas Adhikari

Spring 2022-23, IIT Kharagpur

Lecture 13
March 1, 2023

Linear discriminant analysis

Background

→ PCA is unsupervised i.e. blind to training labels

→ PCS focuses on variance which need not be always relevant

Linear discriminant analysis

Background

→ PCA is unsupervised i.e. blind to training labels

→ PCS focuses on variance which need not be always relevant

Linear Discriminant Analysis (LDA)

→ is a dimensionality reduction technique in machine learning to solve more than two-class classification problems

→ is also known as Normal Discriminant Analysis (NDA) or Discriminant Function Analysis (DFA) (why!!)

Linear discriminant analysis

Background

→ PCA is unsupervised i.e. blind to training labels

→ PCS focuses on variance which need not be always relevant

Linear Discriminant Analysis (LDA)

→ is a dimensionality reduction technique in machine learning to solve more than two-class classification problems

→ is also known as Normal Discriminant Analysis (NDA) or Discriminant Function Analysis (DFA) (why!!)

→ Uses - Face Recognition, medical data analysis, customer identification etc.

LDA

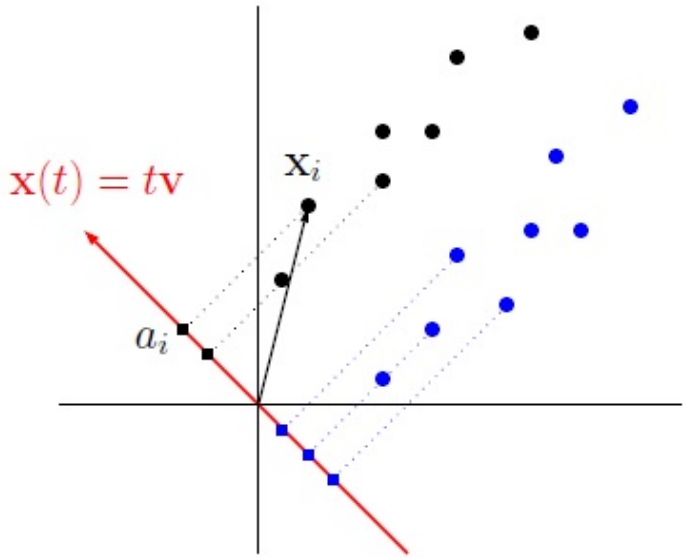
Two-class problem Let $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \in \mathbb{R}^d$ be a given data set consisting of two classes Π_1, Π_2 with n_1 and n_2 number of points respectively. Then find a unit vector that 'best' discriminates between the classes.

LDA

Two-class problem Let $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \in \mathbb{R}^d$ be a given data set consisting of two classes Π_1, Π_2 with n_1 and n_2 number of points respectively. Then find a unit vector that 'best' discriminates between the classes.

Let $\mathbf{v}$ be the direction. The orthogonal projections of the points are

$$a_i = \mathbf{v}^T \mathbf{x}_i, \ 1 \leq i \leq n$$



LDA

Naive idea The separation between the two classes can be measured by the distance between the two class means:

$$\text{measure of separation:} \quad |\mu_1 - \mu_2|$$

where

$$\mu_1 = \frac{1}{n_1} \sum_{\mathbf{x}_i \in \Pi_1} a_i = \frac{1}{n_1} \sum_{\mathbf{x} \in \Pi_1} \mathbf{v}^T \mathbf{x}_i = \mathbf{v}^T \cdot \frac{1}{n_1} \sum_{\mathbf{x}_i \in \Pi_i} \mathbf{x}_i = \mathbf{v}^T \mathbf{m}_1$$

LDA

Naive idea The separation between the two classes can be measured by the distance between the two class means:

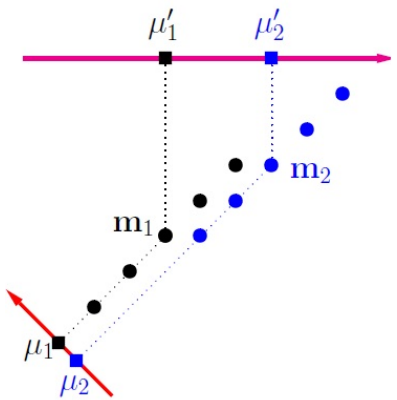
$$\text{measure of separation:} \quad |\mu_1 - \mu_2|$$

where

$$\mu_1 = \frac{1}{n_1} \sum_{\mathbf{x}_i \in \Pi_1} a_i = \frac{1}{n_1} \sum_{\mathbf{x} \in \Pi_1} \mathbf{v}^T \mathbf{x}_i = \mathbf{v}^T \cdot \frac{1}{n_1} \sum_{\mathbf{x}_i \in \Pi_i} \mathbf{x}_i = \mathbf{v}^T \mathbf{m}_1$$

Similarly,

$$\mu_2 = \mathbf{v}^T \mathbf{m}_2, \ \mathbf{m}_2 = \frac{1}{n_2} \sum_{\mathbf{x}_i \in \Pi_2} \mathbf{x}_i$$



LDA

Thus the problem is:

$$\max_{\|\mathbf{v}\|=1} |\mu_1 - \mu_2|$$

where

$$\mu_j = \mathbf{v}^T \mathbf{m}_j, \; j = 1, 2.$$

LDA

Thus the problem is:

$$\max_{\|\mathbf{v}\|=1} |\mu_1 - \mu_2|$$

where

$$\mu_j = \mathbf{v}^T \mathbf{m}_j,\ j = 1, 2.$$

Further, we should pay attention to the variances of the projected classes:

$$s_1^2 = \sum_{\mathbf{x}_i \in \Pi_1} (a_i - \mu_i)^2,\ s_2^2 = \sum_{\mathbf{x}_i \in \Pi_2} (a_i - \mu_2)^2$$

LDA

Thus the problem is:

$$\max_{\|\mathbf{v}\|=1} |\mu_1 - \mu_2|$$

where

$$\mu_j = \mathbf{v}^T \mathbf{m}_j, \ j = 1, 2.$$

Further, we should pay attention to the variances of the projected classes:

$$s_1^2 = \sum_{\mathbf{x}_i \in \Pi_1} (a_i - \mu_i)^2, \ s_2^2 = \sum_{\mathbf{x}_i \in \Pi_2} (a_i - \mu_2)^2$$

Thus modified problem is:

$$\max_{\|\mathbf{v}\|=1} \frac{(\mu_1 - \mu_2)^2}{s_1^2 + s_2^2},$$

where the optimal $\mathbf{v}$ should be be such that $(\mu_1 - \mu_2)^2$ large and s_1^2, s_2^2 both small.

LDA

Now

$$\begin{aligned}(\mu_1 - \mu_2)^2 &= (\mathbf{v}^T\mathbf{m}_1 - \mathbf{v}^T\mathbf{m}_2)^2 \\ &= (\mathbf{v}^T(\mathbf{m}_1 - \mathbf{m}_2))^2 \\ &= \mathbf{v}^T(\mathbf{m}_1 - \mathbf{m}_2)\cdot(\mathbf{m}_1 - \mathbf{m}_2)^T\mathbf{v} \\ &= \mathbf{v}^T S_b \mathbf{v}, \text{ where}\end{aligned}$$

$$S_b = (\mathbf{m}_1 - \mathbf{m}_2)\cdot(\mathbf{m}_1 - \mathbf{m}_2)^T \in \mathbb{R}^{d\times d}$$

is called the **b**etween-class scatter matrix.

LDA

Now

$$\begin{aligned}(\mu_1 - \mu_2)^2 &= (\mathbf{v}^T\mathbf{m}_1 - \mathbf{v}^T\mathbf{m}_2)^2 \\ &= (\mathbf{v}^T(\mathbf{m}_1 - \mathbf{m}_2))^2 \\ &= \mathbf{v}^T(\mathbf{m}_1 - \mathbf{m}_2)\cdot(\mathbf{m}_1 - \mathbf{m}_2)^T\mathbf{v} \\ &= \mathbf{v}^T S_b \mathbf{v}, \text{ where}\end{aligned}$$

$$S_b = (\mathbf{m}_1 - \mathbf{m}_2)\cdot(\mathbf{m}_1 - \mathbf{m}_2)^T \in \mathbb{R}^{d\times d}$$

is called the **b**etween-class scatter matrix.

Note The matrix S_b is symmetric positive semi-definite with rank one matrix.

LDA

Further, for each class Π_j, $j = 1, 2$, the variance of the projection onto $\mathbf{v}$ is

$$\begin{aligned} s_j^2 &= \sum_{\mathbf{x}_i \in \Pi_j} (a_i - \mu_j)^2 = \sum_{\mathbf{x}_i \in \Pi_j} (\mathbf{v}^T \mathbf{x}_i - \mathbf{v}^T \mathbf{m}_j)^2 \\ &= \sum_{\mathbf{x}_i \in \Pi_j} \mathbf{v}^T (\mathbf{x}_i - \mathbf{m}_j)(\mathbf{x}_i - \mathbf{m}_j)^T \mathbf{v} \\ &= \mathbf{v}^T \left(\sum_{\mathbf{x}_i \in \Pi_j} (\mathbf{x}_i - \mathbf{m}_j)(\mathbf{x}_i - \mathbf{m}_j)^T \right) \mathbf{v} \\ &= \mathbf{v}^T S_j \mathbf{v}, \text{ where} \end{aligned}$$

$S_j = \sum_{\mathbf{x}_i \in \Pi_j} (\mathbf{x}_i - \mathbf{m}_j)(\mathbf{x}_i - \mathbf{m}_j)^T \in \mathbb{R}^{d \times d}$ is called the **w**ithin-class scatter matrix for class j.

LDA

Then the total within-class scatter of the two classes in the project space is

$$s_1^2 + s_2^2 = \mathbf{v}^T S_1 \mathbf{v} + \mathbf{v} S_2 \mathbf{v} = \mathbf{v}^T (S_1 + S_2)\mathbf{v} = \mathbf{v}^T S_w \mathbf{v},$$

where

$$S_w = S_1 + S_2 = \sum_{\mathbf{x}_i \in \Pi_1} (\mathbf{x}_i - \mathbf{m}_1)(\mathbf{x}_i - \mathbf{m}_1)^T + \sum_{\mathbf{x}_i \in \Pi_2} (\mathbf{x}_i - \mathbf{m}_2)(\mathbf{x}_i - \mathbf{m}_2)^T$$

is called the total within-class scatter matrix of the original data

LDA

Then the total within-class scatter of the two classes in the project space is

$$s_1^2 + s_2^2 = \mathbf{v}^T S_1 \mathbf{v} + \mathbf{v} S_2 \mathbf{v} = \mathbf{v}^T (S_1 + S_2)\mathbf{v} = \mathbf{v}^T S_w \mathbf{v},$$

where

$$S_w = S_1 + S_2 = \sum_{\mathbf{x}_i \in \Pi_1} (\mathbf{x}_i - \mathbf{m}_1)(\mathbf{x}_i - \mathbf{m}_1)^T + \sum_{\mathbf{x}_i \in \Pi_2} (\mathbf{x}_i - \mathbf{m}_2)(\mathbf{x}_i - \mathbf{m}_2)^T$$

is called the total within-class scatter matrix of the original data

Therefore we arrive at the optimization problem

$$\max_{\|\mathbf{v}\|=1} \frac{\mathbf{v}^T S_b \mathbf{v}}{\mathbf{v}^T S_w \mathbf{v}}$$

LDA

Theorem Suppose S_w is nonsingular. Then the maximizer of the problem is given by the largest eigenvector $\mathbf{v}_1$ of $S_w^{-1} S_b$

LDA

Theorem Suppose S_w is nonsingular. Then the maximizer of the problem is given by the largest eigenvector $\mathbf{v}_1$ of $S_w^{-1} S_b$

Note The rank of $S_w^{-1} S_b$ = rank of $S_b = 1$, hence there is only one nonzero eigenvalue (positive!!) can be found. It represents the largest amount of separation between the two classes along any single direction.

LDA

Theorem Suppose S_w is nonsingular. Then the maximizer of the problem is given by the largest eigenvector $\mathbf{v}_1$ of $S_w^{-1} S_b$

Note The rank of $S_w^{-1} S_b$ = rank of $S_b = 1$, hence there is only one nonzero eigenvalue (positive!!) can be found. It represents the largest amount of separation between the two classes along any single direction.

Question What happens if S_w is not invertible?

LDA

Theorem Suppose S_w is nonsingular. Then the maximizer of the problem is given by the largest eigenvector $\mathbf{v}_1$ of $S_w^{-1} S_b$

Note The rank of $S_w^{-1} S_b$ = rank of $S_b = 1$, hence there is only one nonzero eigenvalue (positive!!) can be found. It represents the largest amount of separation between the two classes along any single direction.

Question What happens if S_w is not invertible?
Question What is generalized eigenvalue problem?

LDA

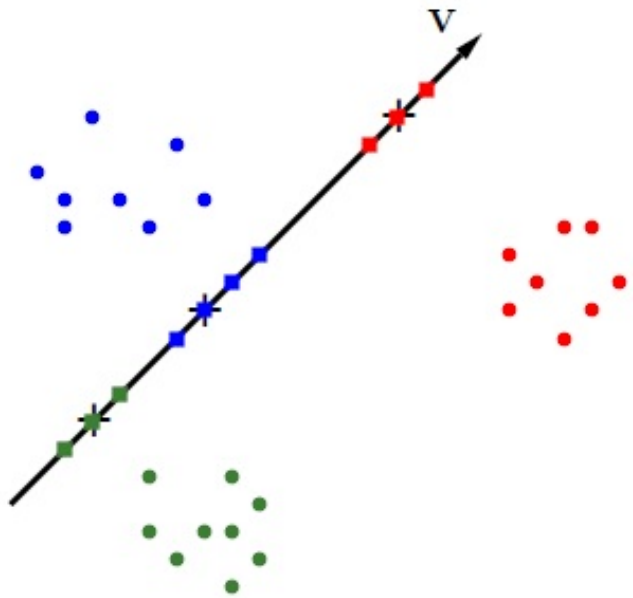
Multiclass problem When there are more than 2 classes, what is the most discriminatory direction?

LDA

Multiclass problem When there are more than 2 classes, what is the most discriminatory direction?

Intuition The optimal direction $\mathbf{v}$ should project the different classes such that

- △ each class is as dense as possible
- △ the centroids of the classes are as far as possible



LDA

Assume that there are c classes and a class Π_j contains n_j data points. Then for any unit vector $\mathbf{v}$, the tightness of the projected classes of the training data is described by the total within-class scatter:

$$\sum_{j=1}^{c} s_j^2 = \sum \mathbf{v}^T S_j \mathbf{v} = \mathbf{v}^T \left(\sum_j S_j \right) \mathbf{v} = \mathbf{v}^T S_w \mathbf{v},$$

where

$$S_j = \sum_{\mathbf{x} \in \Pi_j} (\mathbf{x} - \mathbf{m}_j)(\mathbf{x} - \mathbf{m}_j)^T$$

and $S_w = \sum S_j$ is the total within-class scatter matrix

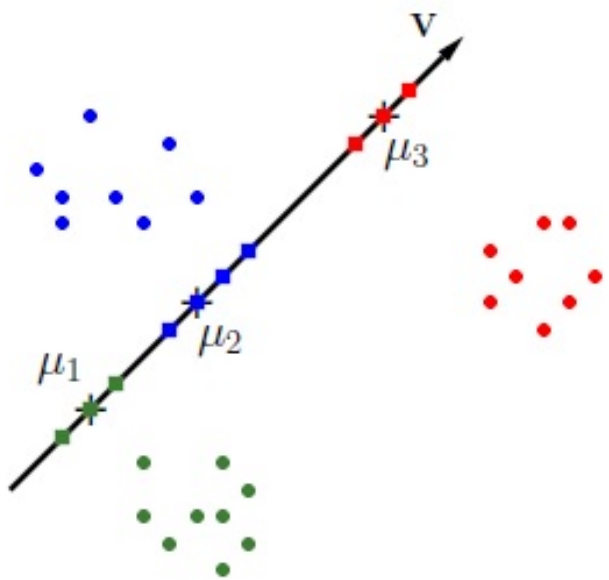
LDA

To make the class centroids in the project space as far from each other as possible, we can maximize the variance of these centroids set $\{\mu_1, \ldots, \mu_c\}$:

$$\sum_{j=1}^{c}(\mu_j - \overline{\mu})^2 = \frac{1}{c}\sum_{j<l}(\mu_j - \mu_l)^2,$$

where

$$\overline{\mu} = \frac{1}{c}\sum_{j=1}^{c}\mu_j$$



LDA

Indeed, we use a weighted mean of the projected centroids to define the between-class scatter:

$$\sum_{j=1}^{c} n_j(\mu_j - \mu)^2, \text{ where } \mu = \frac{1}{n}\sum_{j=1}^{c} n_j\mu_j$$

since the weighted mean μ is the projection of the global centroid $\mathbf{m}$ on the training data onto $\mathbf{v}$:

$$\mathbf{v}^T\mathbf{m} = \mathbf{v}^T\left(\frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i\right) = \mathbf{v}^T\left(\frac{1}{n}\sum_{j=1}^{c} n_j\mathbf{m}_j\right) = \frac{1}{n}\sum_{j=1}^{c} n_j\mu_j = \mu$$

LDA

Indeed, we use a weighted mean of the projected centroids to define the between-class scatter:

$$\sum_{j=1}^{c} n_j(\mu_j - \mu)^2, \text{ where } \mu = \frac{1}{n}\sum_{j=1}^{c} n_j\mu_j$$

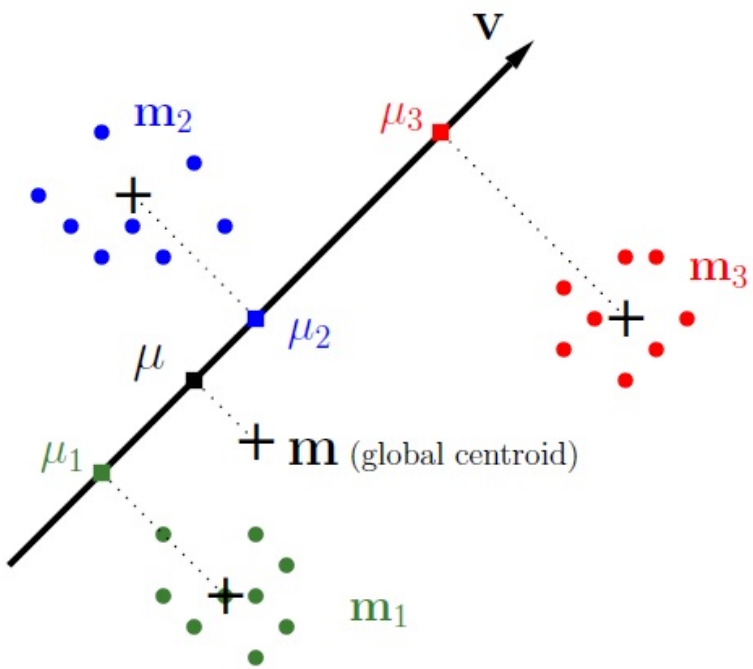
since the weighted mean μ is the projection of the global centroid $\mathbf{m}$ on the training data onto $\mathbf{v}$:

$$\mathbf{v}^T\mathbf{m} = \mathbf{v}^T\left(\frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i\right) = \mathbf{v}^T\left(\frac{1}{n}\sum_{j=1}^{c} n_j\mathbf{m}_j\right) = \frac{1}{n}\sum_{j=1}^{c} n_j\mu_j = \mu$$

Note Note that the simple mean does not have such a geometric interpretation:

$$\overline{\mu} = \frac{1}{c}\sum_{j=1}^{c}\mu_j = \frac{1}{c}\sum_{j=1}^{c}\mathbf{v}^T\mathbf{m}_j = \mathbf{v}^T\left(\frac{1}{c}\sum_{j=1}^{c}\mathbf{m}_j\right)$$

LDA


v
$\mathbf{m}_2$
μ_3
$\mathbf{m}_3$
μ_2
μ
$\mathbf{m}$ (global centroid)
μ_1
$\mathbf{m}_1$

LDA

Then the between-class scatter in the projection space is:

$$\begin{aligned}
\sum_{j=1}^{c} n_j(\mu_j - \mu)^2 &= \sum n_j(\mathbf{v}^T(\mathbf{m}_j - \mathbf{m}))^2 \\
&= \sum n_j \mathbf{v}^T(\mathbf{m}_j - \mathbf{m})(\mathbf{m}_j - \mathbf{m})^T\mathbf{v} \\
&= \mathbf{v}^T\left(\sum n_j(\mathbf{m}_j - \mathbf{m})(\mathbf{m}_j - \mathbf{m})^T\right)\mathbf{v} \\
&= \mathbf{v}^T S_b \mathbf{v}
\end{aligned}$$

LDA

Then the between-class scatter in the projection space is:

$$\begin{aligned}
\sum_{j=1}^{c} n_j(\mu_j - \mu)^2 &= \sum n_j(\mathbf{v}^T(\mathbf{m}_j - \mathbf{m}))^2 \\
&= \sum n_j \mathbf{v}^T(\mathbf{m}_j - \mathbf{m})(\mathbf{m}_j - \mathbf{m})^T\mathbf{v} \\
&= \mathbf{v}^T\left(\sum n_j(\mathbf{m}_j - \mathbf{m})(\mathbf{m}_j - \mathbf{m})^T\right)\mathbf{v} \\
&= \mathbf{v}^T S_b \mathbf{v}
\end{aligned}$$

Thus the optimization problem becomes

$$\max_{\|\mathbf{v}\|=1} \frac{\mathbf{v}^T S_b \mathbf{v}}{\mathbf{v}^T S_w \mathbf{v}}$$

LDA

Observation When $c = 2$,

$$\sum_{j=1}^{2} n_j(\mu_j - \mu)^2 = \frac{n_1 n_2}{n}(\mu_1 - \mu_2)^2, \text{ where } \mu = \frac{1}{n}(n_1\mu_1 + n_2\mu_2)$$

and

$$\sum_{j=1}^{2} n_j(\mathbf{m}_j - \mathbf{m})(\mathbf{m}_j - \mathbf{m})^T = \frac{n_1 n_2}{n}(\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^T$$

where $\mathbf{m} = \frac{1}{n}(n_1\mathbf{m}_1 + n_2\mathbf{m}_2)$

LDA

Observation When $c = 2$,

$$\sum_{j=1}^{2} n_j(\mu_j - \mu)^2 = \frac{n_1 n_2}{n}(\mu_1 - \mu_2)^2, \text{ where } \mu = \frac{1}{n}(n_1\mu_1 + n_2\mu_2)$$

and

$$\sum_{j=1}^{2} n_j(\mathbf{m}_j - \mathbf{m})(\mathbf{m}_j - \mathbf{m})^T = \frac{n_1 n_2}{n}(\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^T$$

where $\mathbf{m} = \frac{1}{n}(n_1\mathbf{m}_1 + n_2\mathbf{m}_2)$

Thus multiclass LDA $\sum n_j(\mu_j - \mu)^2 / \sum s_j^2$ is a generalization of the two-class LDA $(\mu_1 - \mu_2)^2/(s_1^2 + s_2^2)$

LDA

Finding the optimizer for

$$\max_{\|\mathbf{v}\|=1} \frac{\mathbf{v}^T S_b \mathbf{v}}{\mathbf{v}^T S_w \mathbf{v}}$$

can be obtained by finding the generalized eigenvalue problem

$$S_b \mathbf{v}_1 = \lambda_1 S_w \mathbf{v}_1$$

LDA

Finding the optimizer for

$$\max_{\|\mathbf{v}\|=1} \frac{\mathbf{v}^T S_b \mathbf{v}}{\mathbf{v}^T S_w \mathbf{v}}$$

can be obtained by finding the generalized eigenvalue problem

$$S_b \mathbf{v}_1 = \lambda_1 S_w \mathbf{v}_1$$

However if S_w is invertible then the directions can be found by solving the eigenvalue-eigenvector problem:

$$S_w^{-1} S_b \mathbf{v} = \lambda \mathbf{v}.$$

LDA

Note that

$$
\begin{aligned}
S_b &= \sum n_i(\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T \\
&= [\sqrt{n_1}(\mathbf{m}_1 - \mathbf{m}) \dots \sqrt{n_c}(\mathbf{m}_c - \mathbf{m})] \begin{bmatrix} \sqrt{n_1}(\mathbf{m}_1 - \mathbf{m})^T \\ \vdots \\ \sqrt{n_c}(\mathbf{m}_c - \mathbf{m})^T \end{bmatrix}
\end{aligned}
$$

LDA

Note that

$$\begin{aligned} S_b &= \sum n_i(\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T \\ &= [\sqrt{n_1}(\mathbf{m}_1 - \mathbf{m}) \ldots \sqrt{n_c}(\mathbf{m}_c - \mathbf{m})] \begin{bmatrix} \sqrt{n_1}(\mathbf{m}_1 - \mathbf{m})^T \\ \vdots \\ \sqrt{n_c}(\mathbf{m}_c - \mathbf{m})^T \end{bmatrix} \end{aligned}$$

Further

$$\begin{aligned} & \sqrt{n_1} \cdot \sqrt{n_1}(\mathbf{m}_1 - \mathbf{m}) + \ldots + \sqrt{n_c} \cdot \sqrt{n_c}(\mathbf{m}_c - \mathbf{m}) \\ = \; & (n_1\mathbf{m}_1 + \ldots + n_c\mathbf{m}_c) - (n_1 + \ldots + n_c)\mathbf{m} \\ = \; & n\mathbf{m} - n\mathbf{m} = 0 \end{aligned}$$

and hence the vectors $\{\sqrt{n_1}(\mathbf{m}_1 - \mathbf{m}), \ldots, \sqrt{n_c}(\mathbf{m}_c - \mathbf{m})\}$ is linearly dependent.

LDA

Note that

$$\begin{aligned} S_b &= \sum n_i(\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T \\ &= [\sqrt{n_1}(\mathbf{m}_1 - \mathbf{m}) \ldots \sqrt{n_c}(\mathbf{m}_c - \mathbf{m})] \begin{bmatrix} \sqrt{n_1}(\mathbf{m}_1 - \mathbf{m})^T \\ \vdots \\ \sqrt{n_c}(\mathbf{m}_c - \mathbf{m})^T \end{bmatrix} \end{aligned}$$

Further

$$\begin{aligned} & \sqrt{n_1} \cdot \sqrt{n_1}(\mathbf{m}_1 - \mathbf{m}) + \ldots + \sqrt{n_c} \cdot \sqrt{n_c}(\mathbf{m}_c - \mathbf{m}) \\ = \; & (n_1\mathbf{m}_1 + \ldots + n_c\mathbf{m}_c) - (n_1 + \ldots + n_c)\mathbf{m} \\ = \; & n\mathbf{m} - n\mathbf{m} = 0 \end{aligned}$$

and hence the vectors $\{\sqrt{n_1}(\mathbf{m}_1 - \mathbf{m}), \ldots, \sqrt{n_c}(\mathbf{m}_c - \mathbf{m})\}$ is linearly dependent.
Thus rank$(S_b) \leq c - 1$ and there can be at most $c - 1$ discriminatory directions.

LDA

LDA Algorithm Input: the data matrix $X \in \mathbb{R}^{n \times d}$ with c classes
Output: At most $c-1$ discriminatory directions and projections of X onto them

1. Compute

$$S_w = \sum_{j=1}^{c} \sum_{\mathbf{x} \in \Pi_j} (\mathbf{x} - \mathbf{m}_j)(\mathbf{x} - \mathbf{m}_j)^T, \; S_b = \sum_{j=1}^{c} n_j (\mathbf{m}_j - \mathbf{m})(\mathbf{m}_j - \mathbf{m})^T$$

LDA

LDA Algorithm Input: the data matrix $X \in \mathbb{R}^{n \times d}$ with c classes
Output: At most $c-1$ discriminatory directions and projections of X onto them

1. Compute

$$S_w = \sum_{j=1}^{c} \sum_{\mathbf{x} \in \Pi_j} (\mathbf{x} - \mathbf{m}_j)(\mathbf{x} - \mathbf{m}_j)^T, \; S_b = \sum_{j=1}^{c} n_j (\mathbf{m}_j - \mathbf{m})(\mathbf{m}_j - \mathbf{m})^T$$

2. Solve the generalized eigenvalue problem $S - b\mathbf{v} = \lambda S_w \mathbf{v}$ to find all eigenvectors $V_k = [\mathbf{v}_1 \; \ldots \; \mathbf{v}_k], k \leq c-1$

LDA

LDA Algorithm Input: the data matrix $X \in \mathbb{R}^{n \times d}$ with c classes
Output: At most $c-1$ discriminatory directions and projections of X onto them

1. Compute

$$S_w = \sum_{j=1}^{c} \sum_{\mathbf{x} \in \Pi_j} (\mathbf{x} - \mathbf{m}_j)(\mathbf{x} - \mathbf{m}_j)^T, \; S_b = \sum_{j=1}^{c} n_j (\mathbf{m}_j - \mathbf{m})(\mathbf{m}_j - \mathbf{m})^T$$

2. Solve the generalized eigenvalue problem $S - b\mathbf{v} = \lambda S_w \mathbf{v}$ to find all eigenvectors $V_k = [\mathbf{v}_1 \; \ldots \; \mathbf{v}_k], k \leq c-1$
3. Project the data X onto them $Y = X \cdot V_k \in \mathbb{R}^{n \times k}$