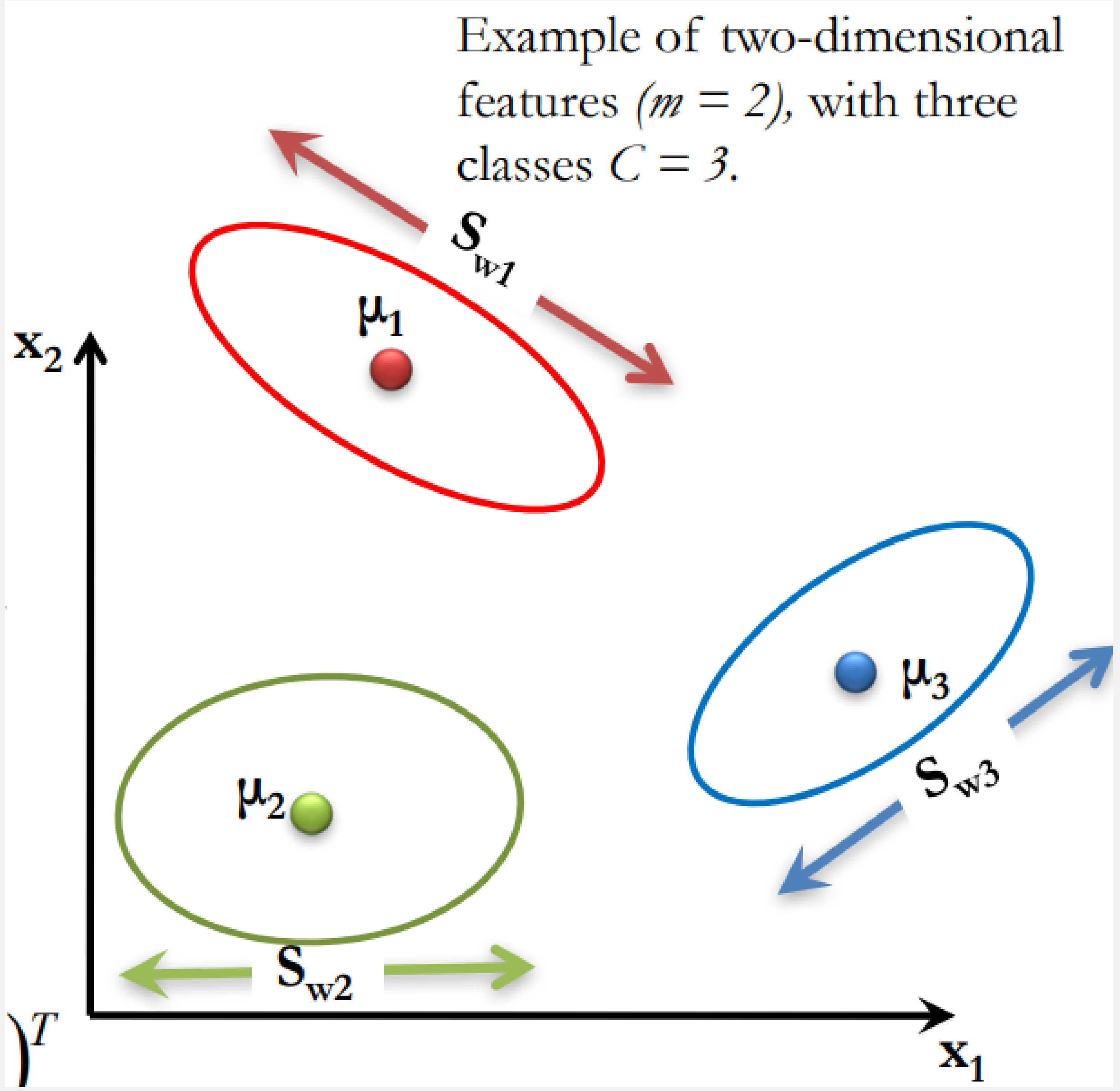
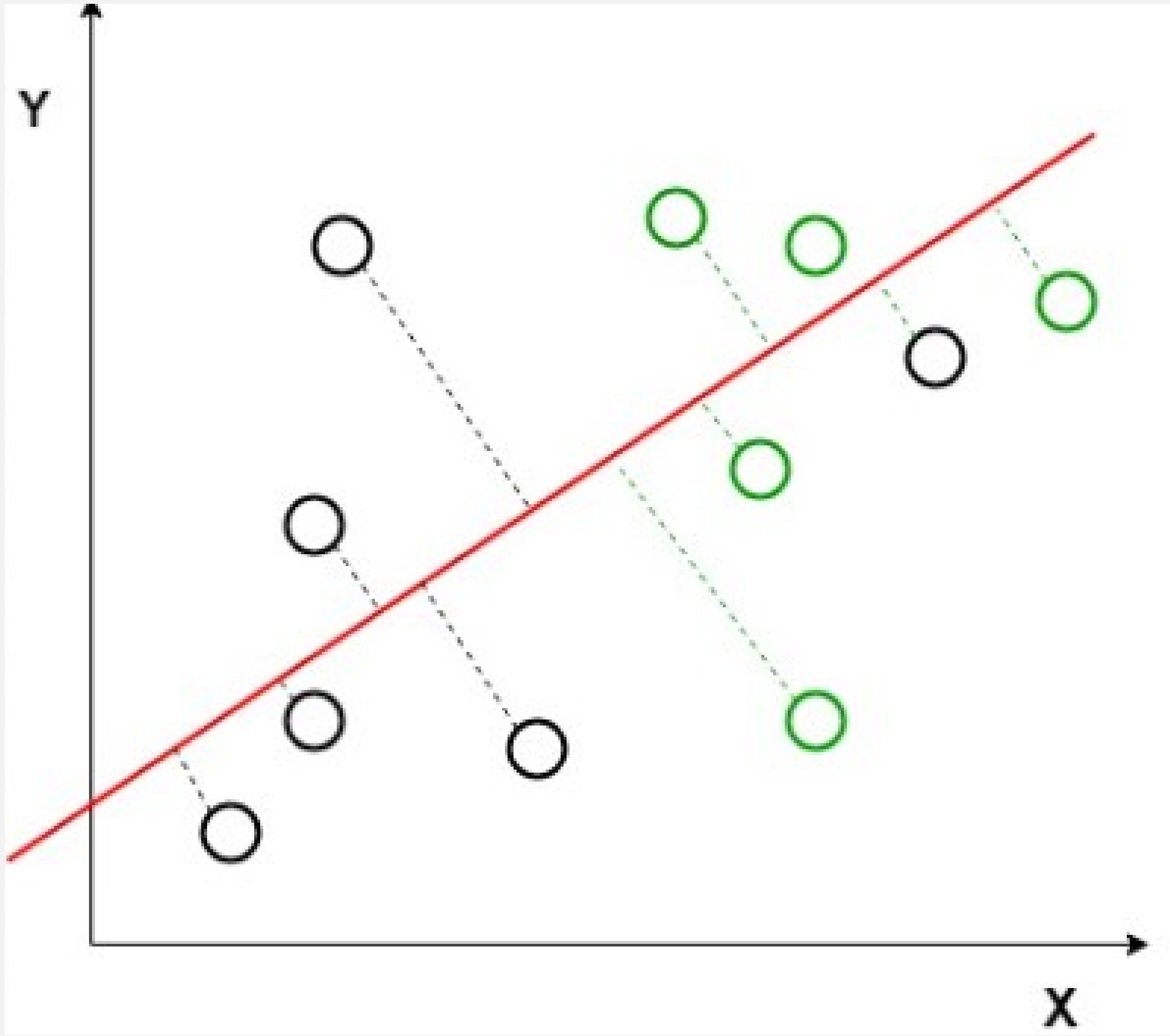
LINEAR DISCRIMINANT ANALYSIS (LDA)


Example of two-dimensional features ($m = 2$), with three classes $C = 3$.

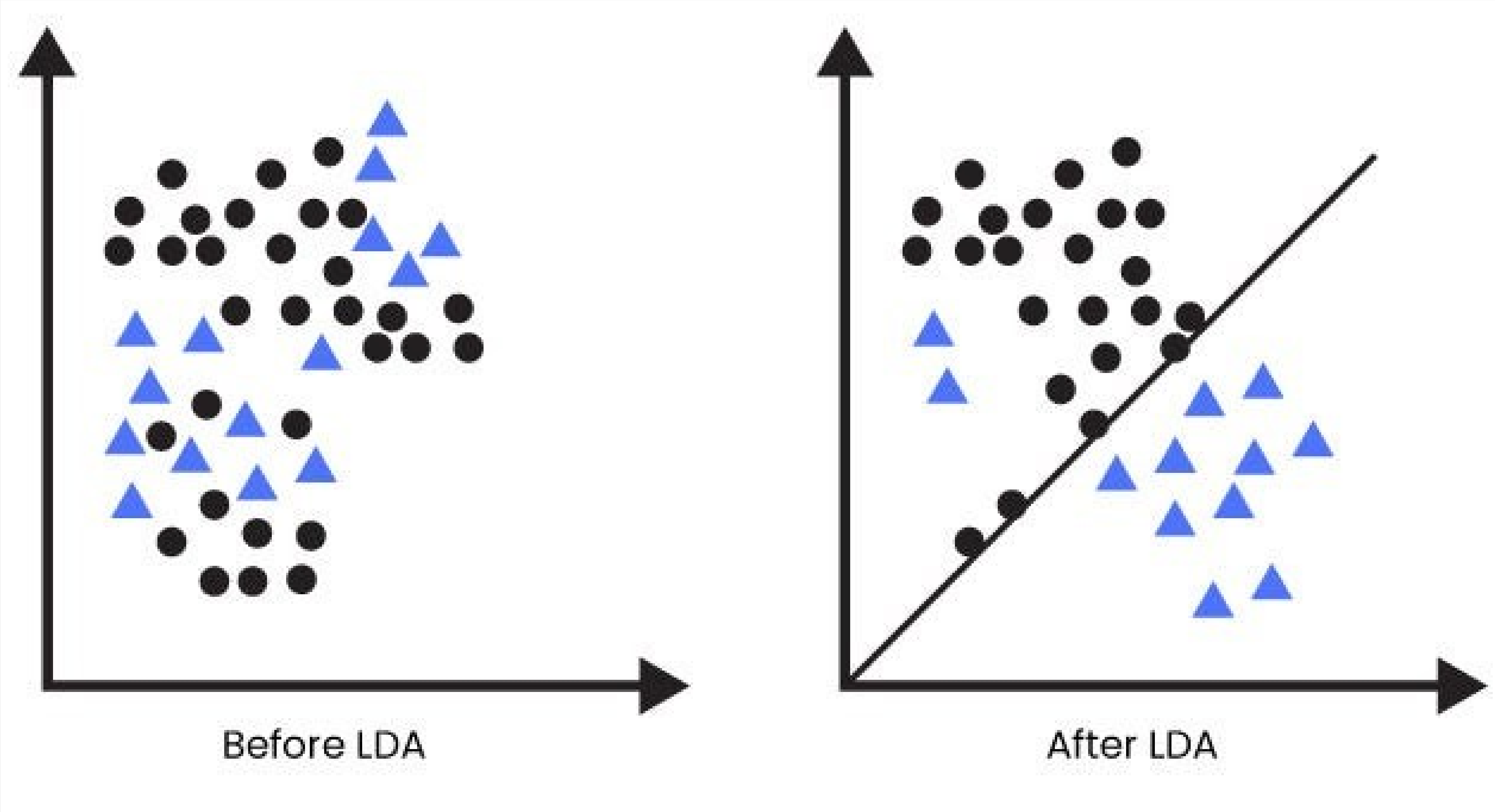


Summer 2024

@IIT Roorkee

INTRODUCTION

- Linear Discriminant Analysis (LDA) is a statistical and machine learning technique used for dimensionality reduction, classification, and pattern recognition.
- It is primarily employed when you have labeled data (supervised learning) and want to separate different classes based on the features
- It aims to project the data in such a way that maximizes the separation between the classes while minimizing the variance within each class.



KEY CONCEPTS OF LDA

LDA is based on the concept of finding a linear combination of features that best separates two or more classes of objects or events. It essentially looks for the axes (linear discriminants) in the feature space that provide the greatest separation between the different classes.

- Dimensionality Reduction : LDA can reduce the dimensionality of the dataset. It finds a lower-dimensional space (e.g., projecting data from a 2D space to a 3D space) where the different classes are better separated.
- Linear Separability : LDA assumes that the different classes can be linearly separated. This means that the boundary between classes is a straight line (in two dimensions), a plane (in three dimensions), or a hyperplane in higher dimensions
- Maximizing Between-Class Separation: LDA works by maximizing the distance between the means of different classes (between-class variance) while minimizing the variance within each class (within-class variance)

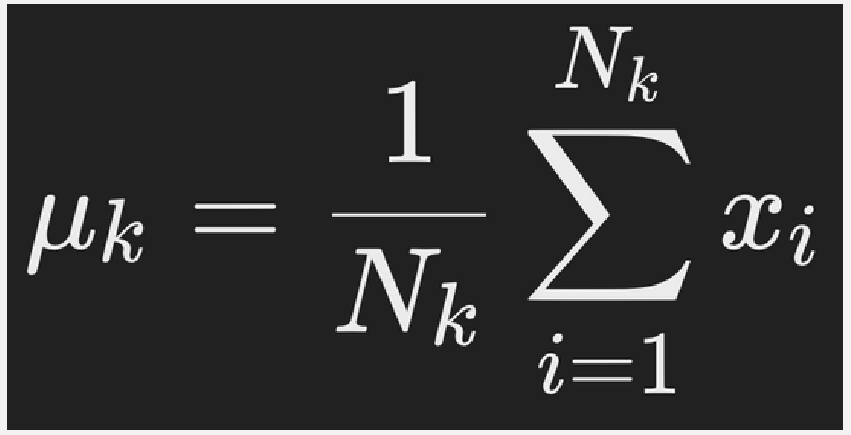
Key Mathematical Concepts in LDA

Let's assume we have a dataset X with N samples, each having d features, and the data is classified into C classes. We will represent the data as follows:

- Xi∈R^(N×d) where each row is a sample and each column is a feature.
- y∈{1,2,...,C} represents the class labels for each sample.

MEAN VECTORS:

For each class k, we calculate the mean vector μk, which represents the average position of the data points in that class in the feature space:

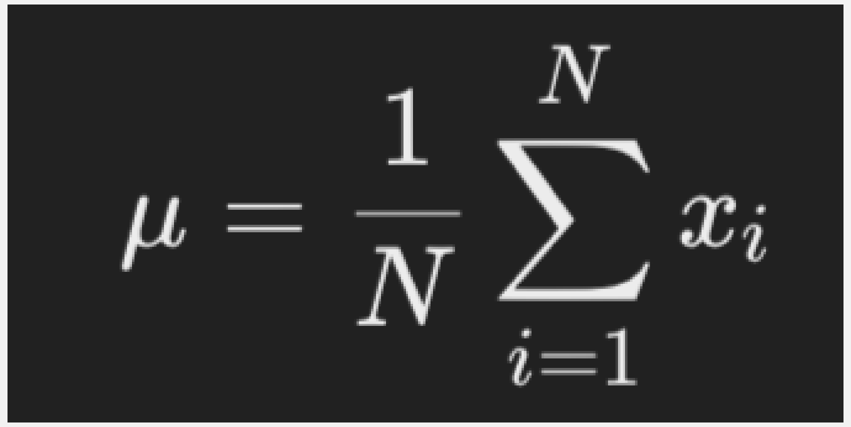


$$\mu_k = \frac{1}{N_k} \sum_{i=1}^{N_k} x_i$$

Where:

- Nk is the number of samples in class k.
- xi is a feature vector for sample i in class k.
- μk is the mean vector for class k.

The overall mean of the dataset, considering all classes, is:



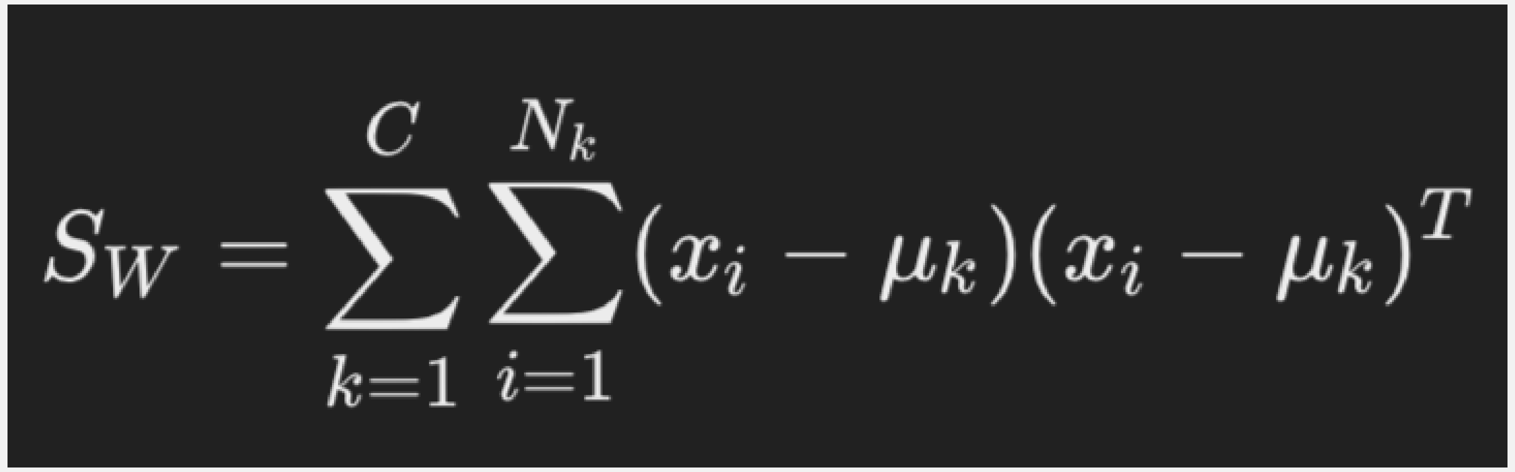
$$\mu = \frac{1}{N} \sum_{i=1}^{N} x_i$$

Where

- N is the total number of samples across all classes.

2.Scatter Matrices:

Within-class scatter matrix (SW): Measures the spread (covariance) of points within each class. The within-class scatter matrix is calculated by summing up the scatter within each class. For class k, the scatter of points within that class is:

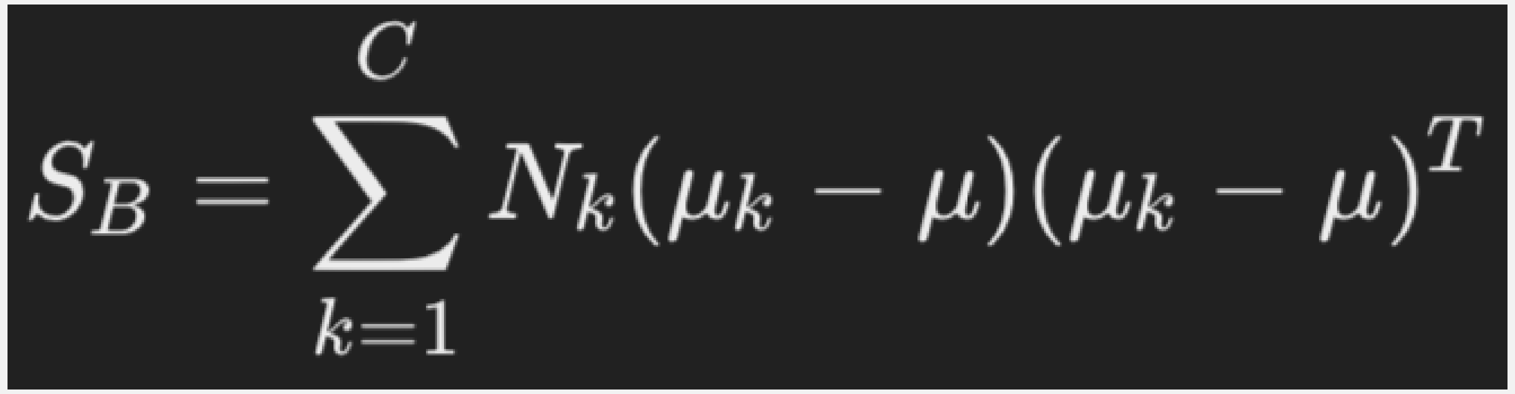


$$S_W = \sum_{k=1}^{C} \sum_{i=1}^{N_k} (x_i - \mu_k)(x_i - \mu_k)^T$$

Where:

- C is the number of classes,
- Nk is the number of samples in class k
- (xi−μk) is the deviation of each sample from the mean of its class.

Between-class scatter matrix (SB): Measures the separation between the means of different classes.The between-class scatter matrix captures how the mean of each class differs from the overall mean:



$$S_B = \sum_{k=1}^{C} N_k (\mu_k - \mu)(\mu_k - \mu)^T$$

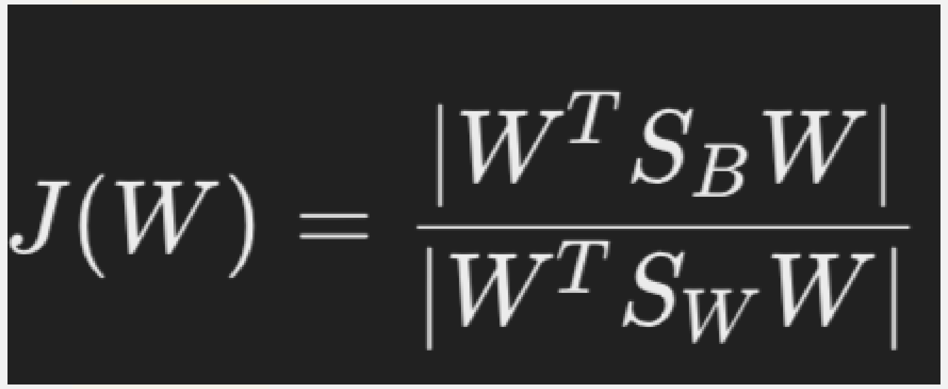
Where:

- Nk is the number of samples in class k,
- μk is the mean vector of class k,
- μ is the overall mean vector of the dataset.

The idea is to maximize SB (separation between classes) and minimize SW (spread within classes).

3. Optimization Objective

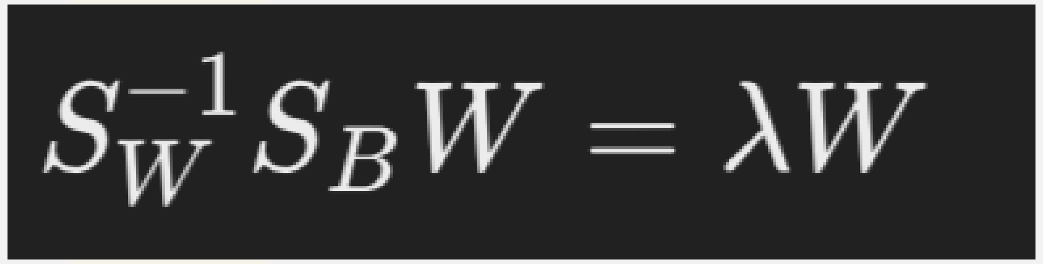
To find the optimal projection that maximizes the separation between the classes, we solve for the projection matrix W that maximizes the following criterion:



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

- W` is the projection matrix (the linear transformation we are looking for).
- The objective is to maximize the ratio of the between-class scatter to the within-class scatter.

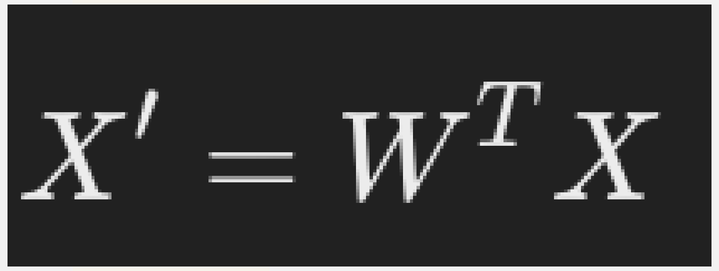
This leads to the following generalized eigenvalue problem:



$$S_W^{-1} S_B W = \lambda W$$

- Where λ represents the eigenvalues.

- Solve the eigenvalue problem to get the eigenvalues and corresponding eigenvectors of SW–1SB.
- The eigenvectors corresponding to the largest eigenvalues define the directions in which the classes are best separated. These eigenvectors form the columns of the projection matrix W.
- Once W is computed, we project the original data X onto the new subspace using:



$$X' = W^T X$$

The new data X′ is the lower-dimensional representation of the original data with maximum class separability.

Applications of LDA

- Face Recognition: LDA is used for dimensionality reduction before classifying faces in face recognition systems.

- Text Classification: It helps categorize documents or emails into different classes like spam or non-spam.

- Medical Diagnosis: LDA is applied to predict the category of diseases based on patient data.

- Credit Scoring: Banks use LDA for distinguishing between good and bad credit risks.

- Image Classification: It is used to classify objects in images by reducing feature dimensions.

Limitations of LDA

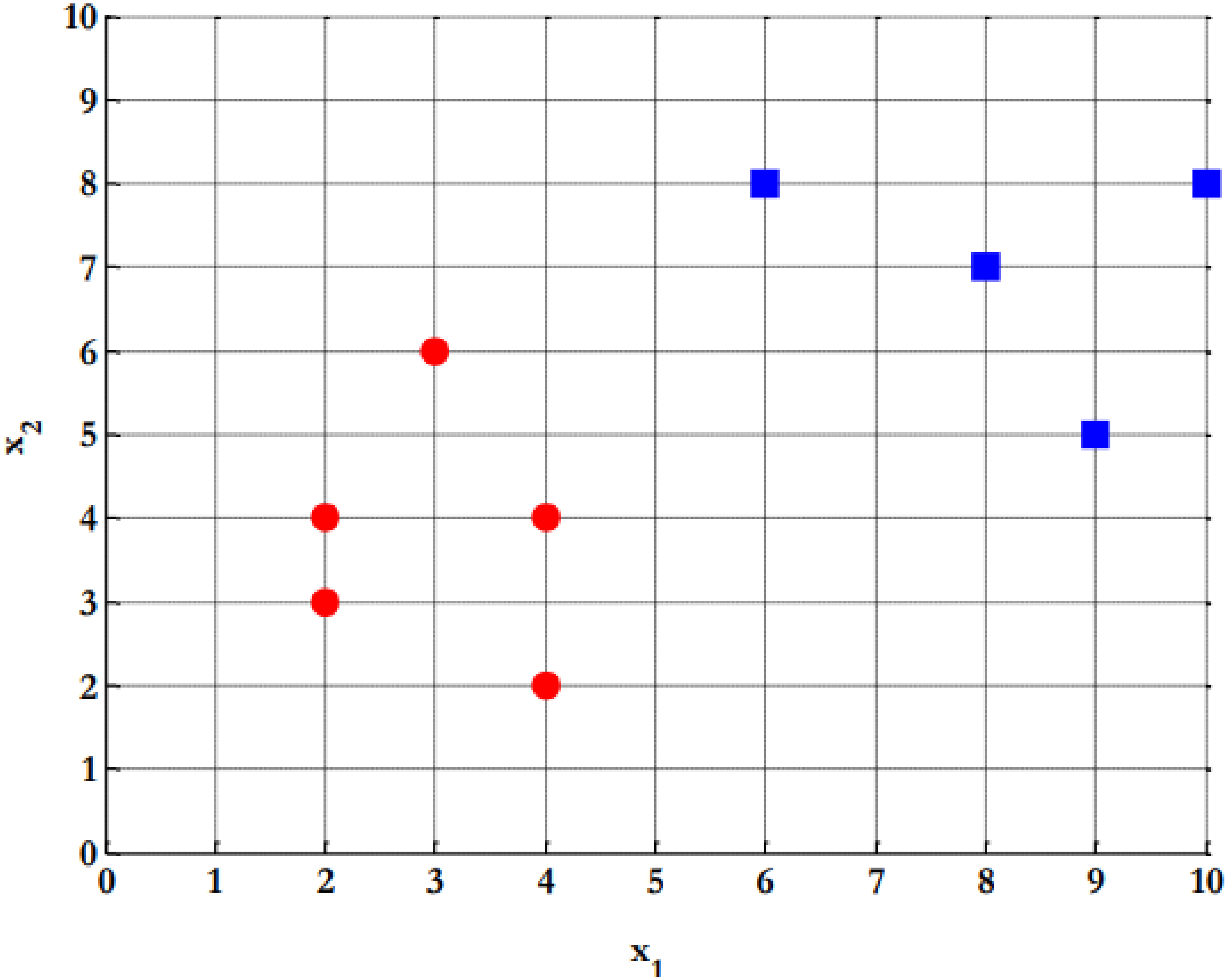
- Assumes Linearly Separable Data: LDA works best when data is linearly separable. It may perform poorly when the decision boundary is nonlinear.

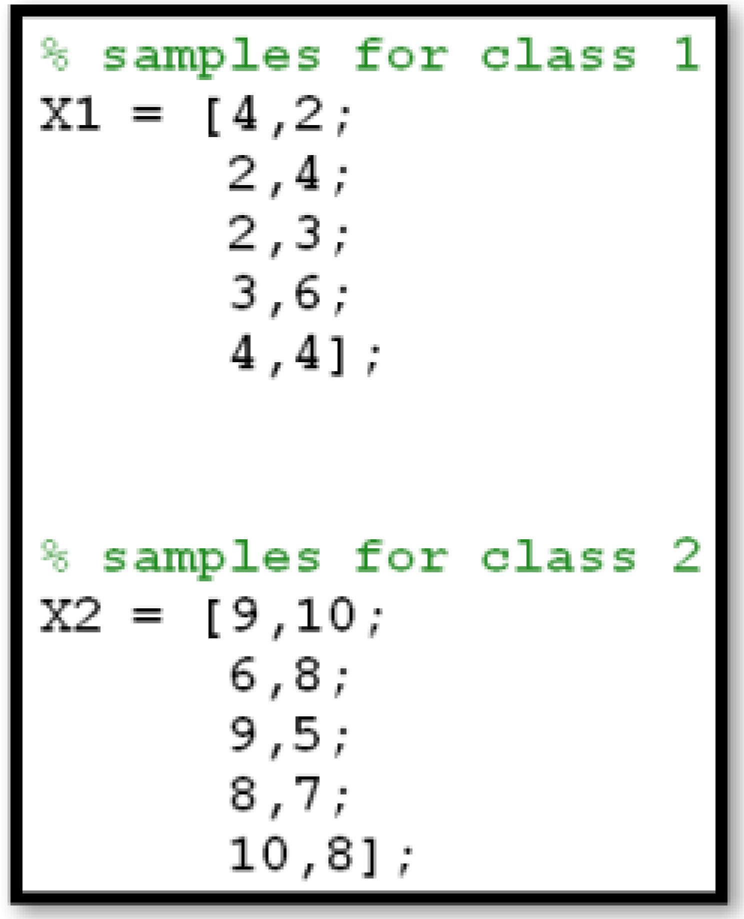
- Normality Assumption: LDA assumes that the features follow a Gaussian distribution, which may not always hold true in practice.

- Equal Covariance Matrices: It assumes that the covariance matrices of the classes are equal, which may not be the case in all applications.

LDA … Two Classes - Example

- Compute the Linear Discriminant projection for the following two-dimensional dataset.
  - Samples for class $\boldsymbol{\omega_1}$: $\mathbf{X_1}$=(x_1,x_2)={(4,2),(2,4),(2,3),(3,6),(4,4)}
  - Sample for class $\boldsymbol{\omega_2}$: $\mathbf{X_2}$=(x_1,x_2)={(9,10),(6,8),(9,5),(8,7),(10,8)}





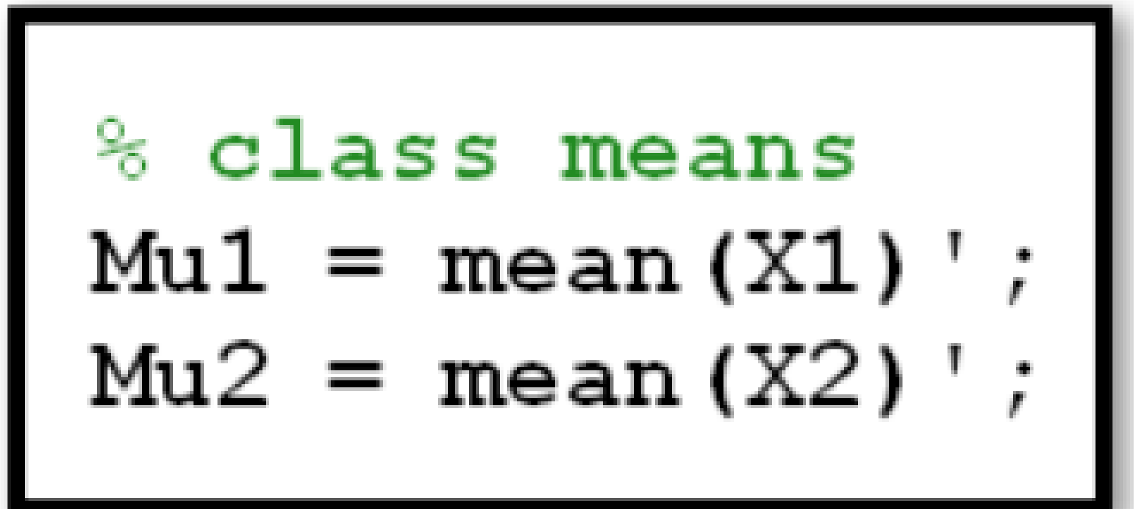
```
% samples for class 1
X1 = [4,2;
      2,4;
      2,3;
      3,6;
      4,4];


% samples for class 2
X2 = [9,10;
      6,8;
      9,5;
      8,7;
      10,8];
```

- The classes mean are :

$$\mu_1 = \frac{1}{N_1}\sum_{x\in\omega_1} x = \frac{1}{5}\left[\begin{pmatrix}4\\2\end{pmatrix}+\begin{pmatrix}2\\4\end{pmatrix}+\begin{pmatrix}2\\3\end{pmatrix}+\begin{pmatrix}3\\6\end{pmatrix}+\begin{pmatrix}4\\4\end{pmatrix}\right]=\begin{pmatrix}3\\3.8\end{pmatrix}$$

$$\mu_2 = \frac{1}{N_2}\sum_{x\in\omega_2} x = \frac{1}{5}\left[\begin{pmatrix}9\\10\end{pmatrix}+\begin{pmatrix}6\\8\end{pmatrix}+\begin{pmatrix}9\\5\end{pmatrix}+\begin{pmatrix}8\\7\end{pmatrix}+\begin{pmatrix}10\\8\end{pmatrix}\right]=\begin{pmatrix}8.4\\7.6\end{pmatrix}$$



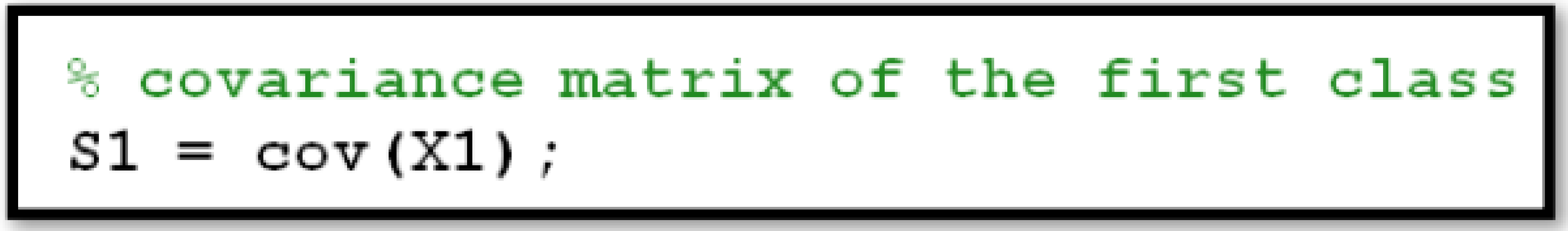
```
% class means
Mu1 = mean(X1)';
Mu2 = mean(X2)';
```

- Covariance matrix of the first class:

$$S_1 = \sum_{x \in \omega_1} (x-\mu_1)(x-\mu_1)^T = \left[\begin{pmatrix}4\\2\end{pmatrix} - \begin{pmatrix}3\\3.8\end{pmatrix}\right]^2 + \left[\begin{pmatrix}2\\4\end{pmatrix} - \begin{pmatrix}3\\3.8\end{pmatrix}\right]^2$$

$$+ \left[\begin{pmatrix}2\\3\end{pmatrix} - \begin{pmatrix}3\\3.8\end{pmatrix}\right]^2 + \left[\begin{pmatrix}3\\6\end{pmatrix} - \begin{pmatrix}3\\3.8\end{pmatrix}\right]^2 + \left[\begin{pmatrix}4\\4\end{pmatrix} - \begin{pmatrix}3\\3.8\end{pmatrix}\right]^2$$

$$= \begin{pmatrix} 1 & -0.25 \\ -0.25 & 2.2 \end{pmatrix}$$



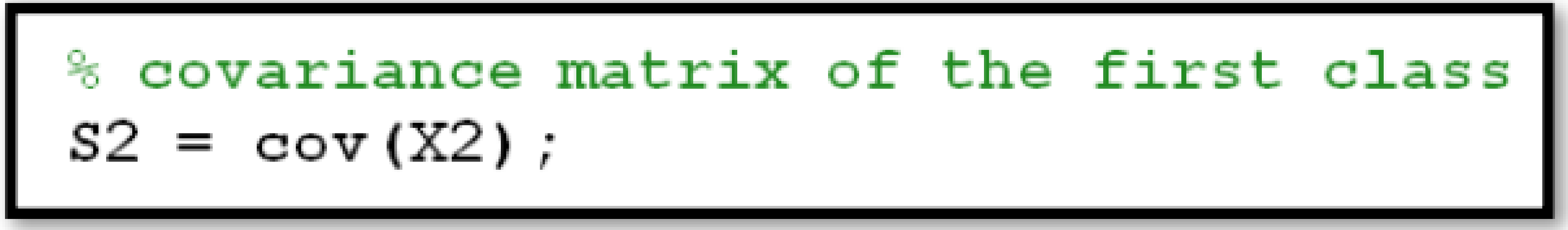
```
% covariance matrix of the first class
S1 = cov(X1);
```

- Covariance matrix of the second class:

$$S_2 = \sum_{x \in \omega_2} (x-\mu_2)(x-\mu_2)^T = \left[\begin{pmatrix}9\\10\end{pmatrix}-\begin{pmatrix}8.4\\7.6\end{pmatrix}\right]^2 + \left[\begin{pmatrix}6\\8\end{pmatrix}-\begin{pmatrix}8.4\\7.6\end{pmatrix}\right]^2$$

$$+\left[\begin{pmatrix}9\\5\end{pmatrix}-\begin{pmatrix}8.4\\7.6\end{pmatrix}\right]^2 + \left[\begin{pmatrix}8\\7\end{pmatrix}-\begin{pmatrix}8.4\\7.6\end{pmatrix}\right]^2 + \left[\begin{pmatrix}10\\8\end{pmatrix}-\begin{pmatrix}8.4\\7.6\end{pmatrix}\right]^2$$

$$= \begin{pmatrix}2.3 & -0.05\\ -0.05 & 3.3\end{pmatrix}$$

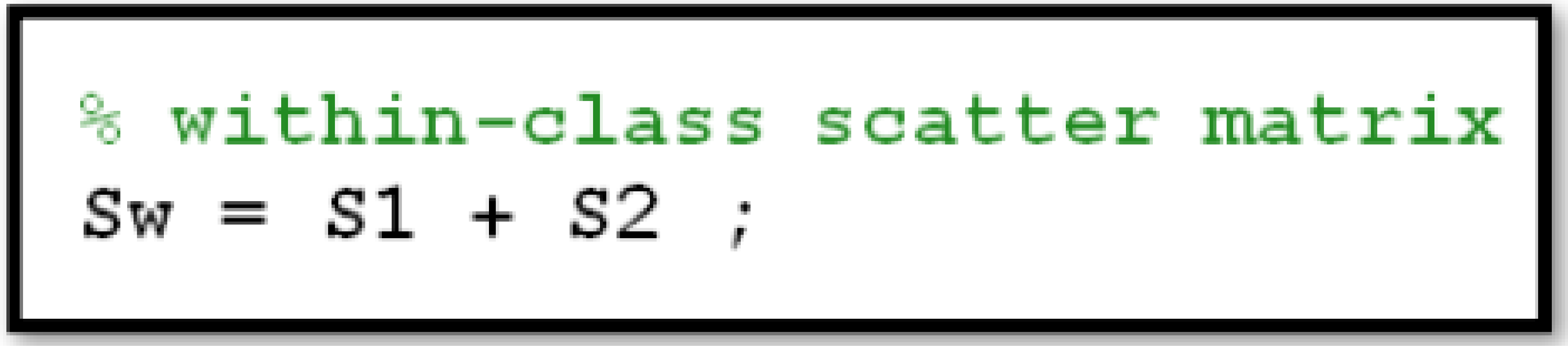


```
% covariance matrix of the first class
S2 = cov(X2);
```

- Within-class scatter matrix:

$$S_w = S_1 + S_2 = \begin{pmatrix} 1 & -0.25 \\ -0.25 & 2.2 \end{pmatrix} + \begin{pmatrix} 2.3 & -0.05 \\ -0.05 & 3.3 \end{pmatrix}$$

$$= \begin{pmatrix} 3.3 & -0.3 \\ -0.3 & 5.5 \end{pmatrix}$$



```
% within-class scatter matrix
Sw = S1 + S2 ;
```

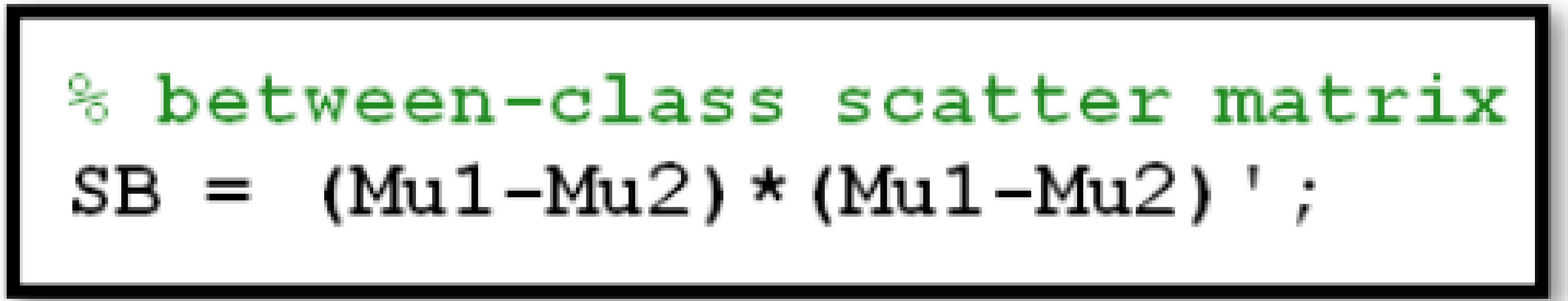
- Between-class scatter matrix:

$$S_B = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$$

$$= \left[\begin{pmatrix} 3 \\ 3.8 \end{pmatrix} - \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix}\right]\left[\begin{pmatrix} 3 \\ 3.8 \end{pmatrix} - \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix}\right]^T$$

$$= \begin{pmatrix} -5.4 \\ -3.8 \end{pmatrix}\begin{pmatrix} -5.4 & -3.8 \end{pmatrix}$$

$$= \begin{pmatrix} 29.16 & 20.52 \\ 20.52 & 14.44 \end{pmatrix}$$



```
% between-class scatter matrix
SB = (Mu1-Mu2)*(Mu1-Mu2)';
```

- The LDA projection is then obtained as the solution of the generalized eigen value problem $S_W^{-1} S_B w = \lambda w$

$$\Rightarrow \left| S_W^{-1} S_B - \lambda I \right| = 0$$

$$\Rightarrow \left| \begin{pmatrix} 3.3 & -0.3 \\ -0.3 & 5.5 \end{pmatrix}^{-1} \begin{pmatrix} 29.16 & 20.52 \\ 20.52 & 14.44 \end{pmatrix} - \lambda \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right| = 0$$

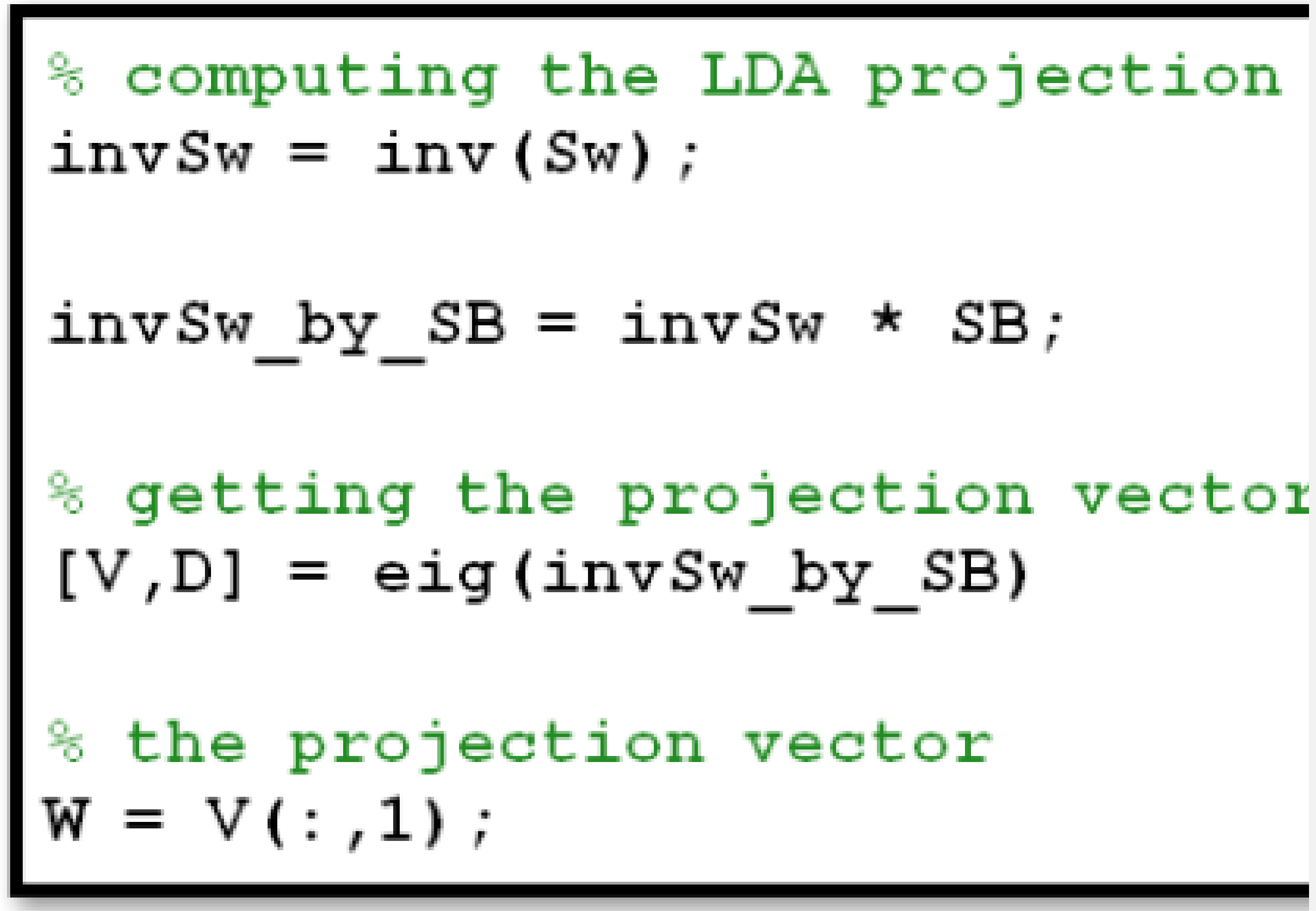
$$\Rightarrow \left| \begin{pmatrix} 0.3045 & 0.0166 \\ 0.0166 & 0.1827 \end{pmatrix} \begin{pmatrix} 29.16 & 20.52 \\ 20.52 & 14.44 \end{pmatrix} - \lambda \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right| = 0$$

$$\Rightarrow \left| \begin{pmatrix} 9.2213 - \lambda & 6.489 \\ 4.2339 & 2.9794 - \lambda \end{pmatrix} \right|$$

$$= (9.2213 - \lambda)(2.9794 - \lambda) - 6.489 \times 4.2339 = 0$$

$$\Rightarrow \lambda^2 - 12.2007\lambda = 0 \Rightarrow \lambda(\lambda - 12.2007) = 0$$

$$\Rightarrow \lambda_1 = 0, \lambda_2 = 12.2007$$



```
% computing the LDA projection
invSw = inv(Sw);

invSw_by_SB = invSw * SB;

% getting the projection vector
[V,D] = eig(invSw_by_SB)

% the projection vector
W = V(:,1);
```

- Hence

$$\begin{pmatrix} 9.2213 & 6.489 \\ 4.2339 & 2.9794 \end{pmatrix} w_1 = \underbrace{0}_{\lambda_1} \begin{pmatrix} w_1 \\ w_2 \end{pmatrix}$$

and

$$\begin{pmatrix} 9.2213 & 6.489 \\ 4.2339 & 2.9794 \end{pmatrix} w_2 = \boxed{\underbrace{12.2007}_{\lambda_2}} \begin{pmatrix} w_1 \\ w_2 \end{pmatrix}$$

Thus;

$$w_1 = \begin{pmatrix} -0.5755 \\ 0.8178 \end{pmatrix} \quad and \quad \boxed{w_2 = \begin{pmatrix} 0.9088 \\ 0.4173 \end{pmatrix} = w^*}$$

- The optimal projection is the one that given maximum $\lambda = J(w)$

Or directly;

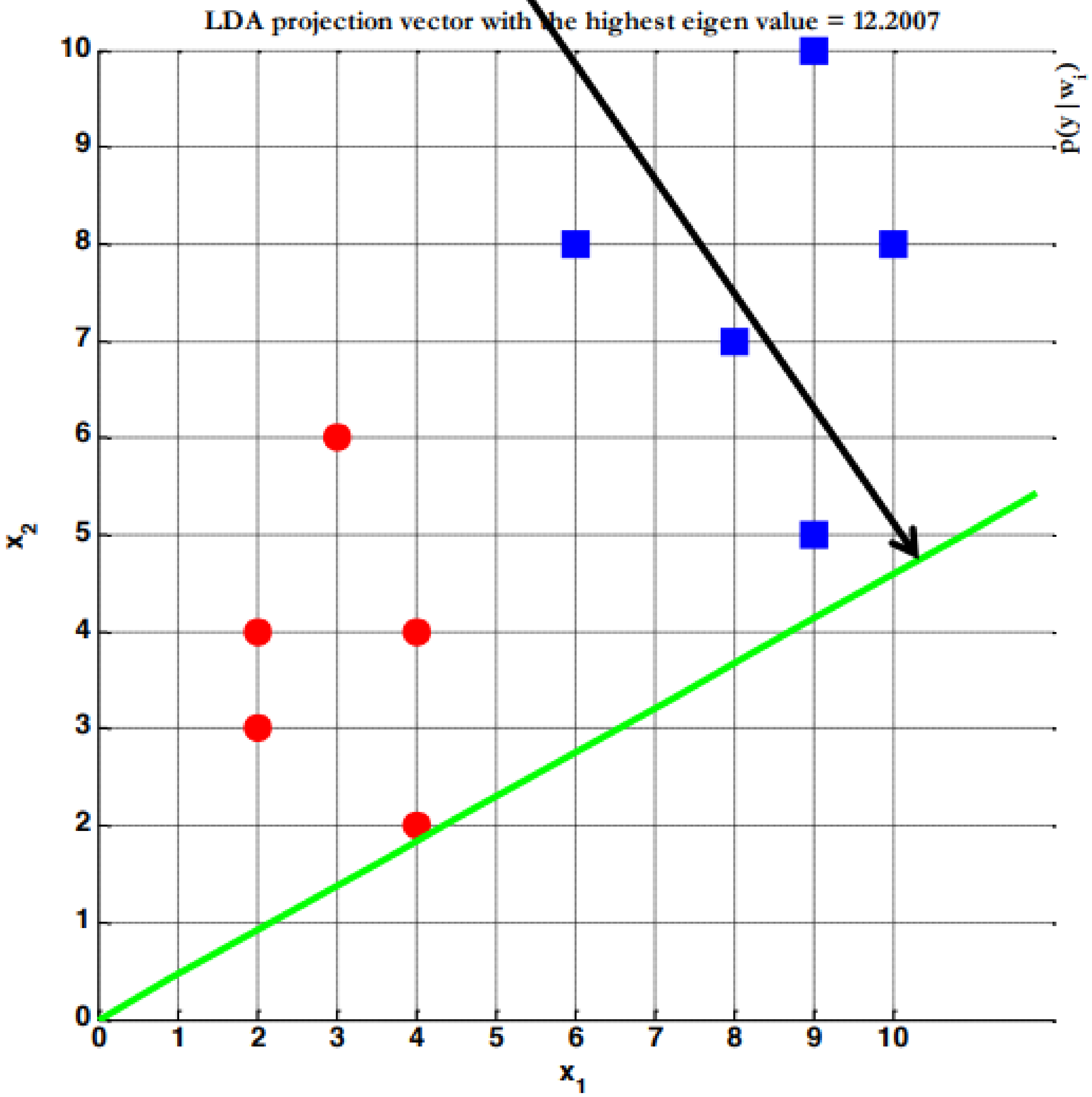
$$w^* = S_W^{-1}(\mu_1 - \mu_2) = \begin{pmatrix} 3.3 & -0.3 \\ -0.3 & 5.5 \end{pmatrix}^{-1} \left[\begin{pmatrix} 3 \\ 3.8 \end{pmatrix} - \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix} \right]$$

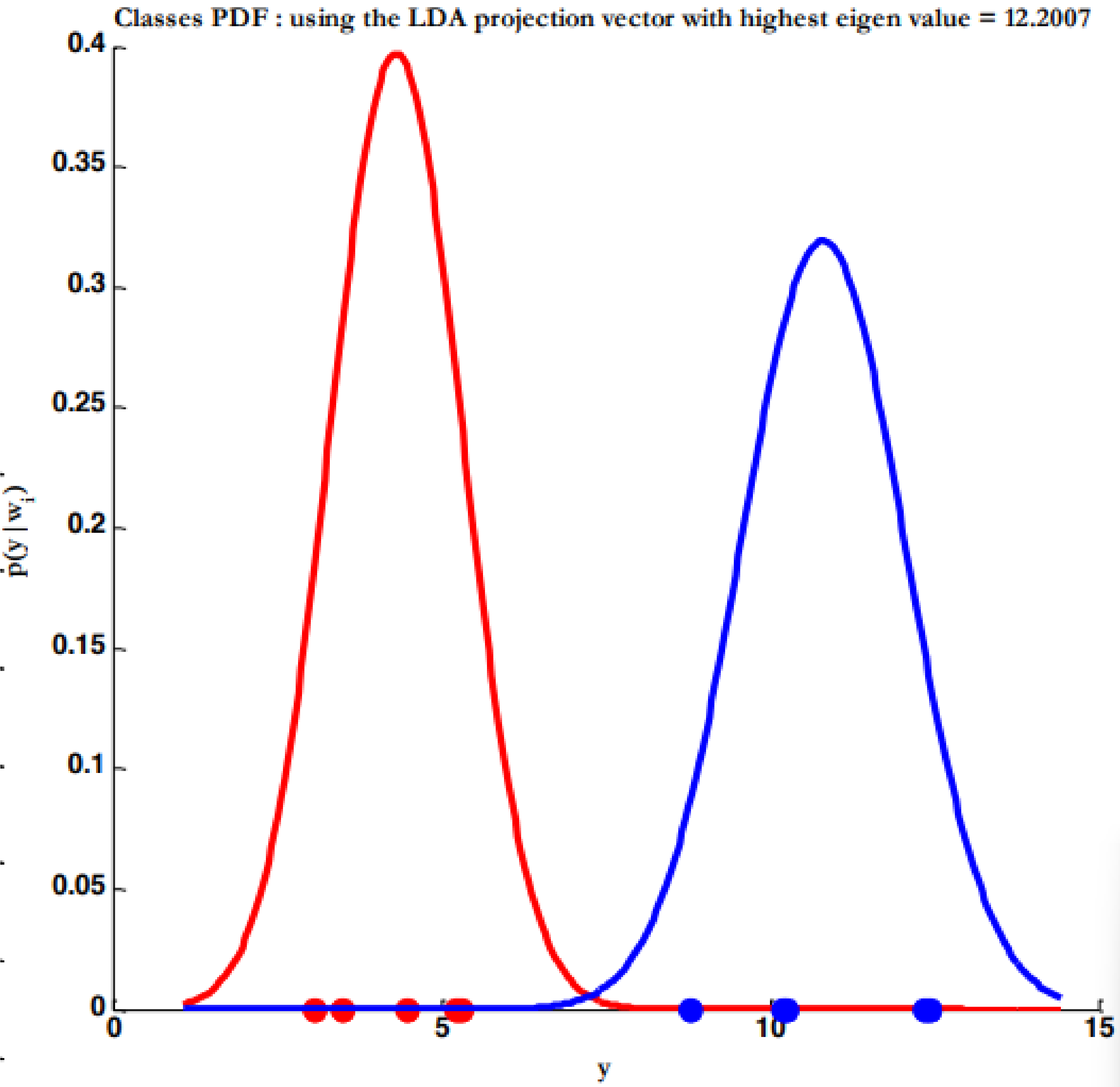
$$= \begin{pmatrix} 0.3045 & 0.0166 \\ 0.0166 & 0.1827 \end{pmatrix} \begin{pmatrix} -5.4 \\ -3.8 \end{pmatrix}$$

$$= \begin{pmatrix} 0.9088 \\ 0.4173 \end{pmatrix}$$

LDA - Projection

The projection vector corresponding to the **highest** eigen value

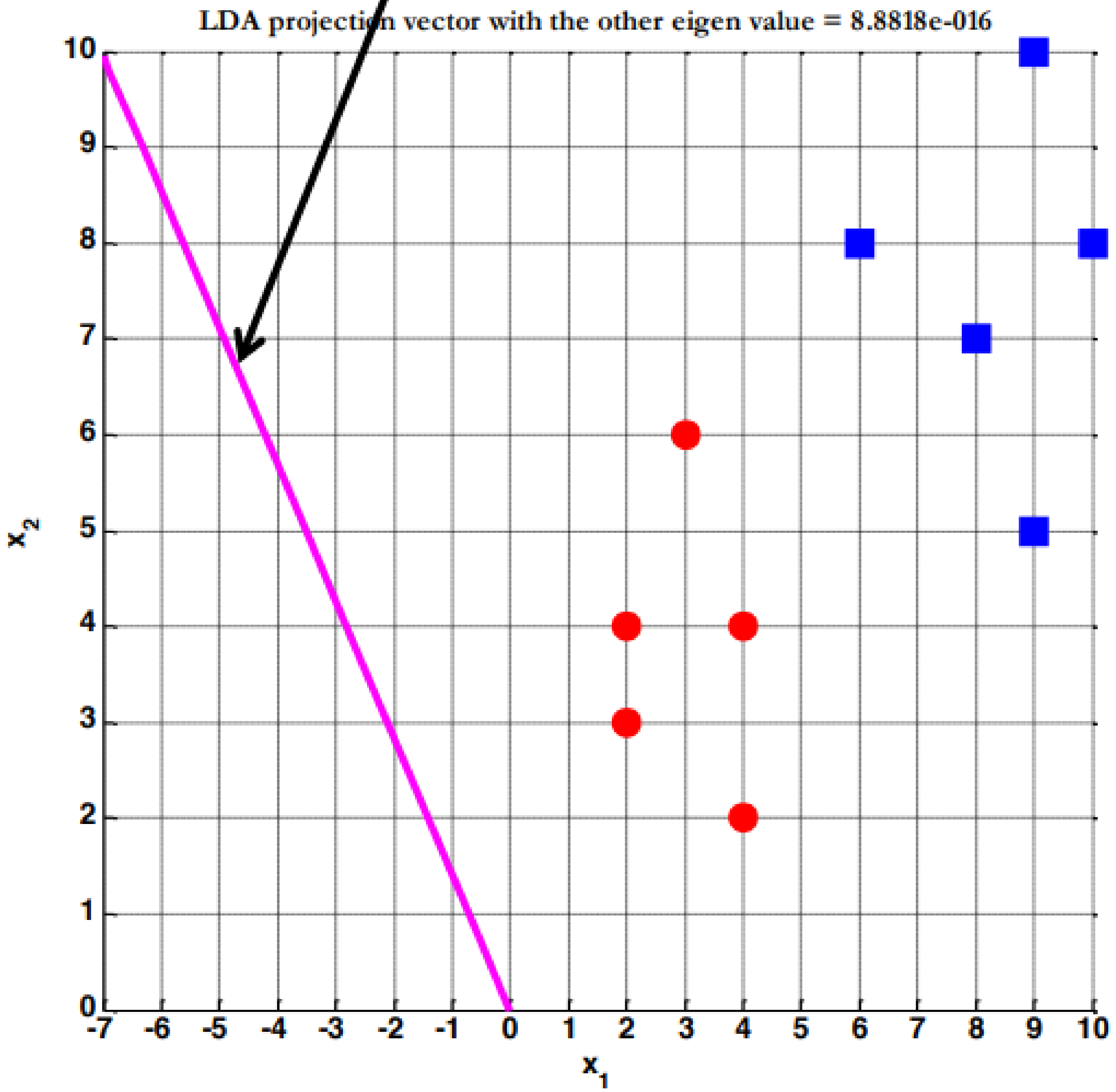


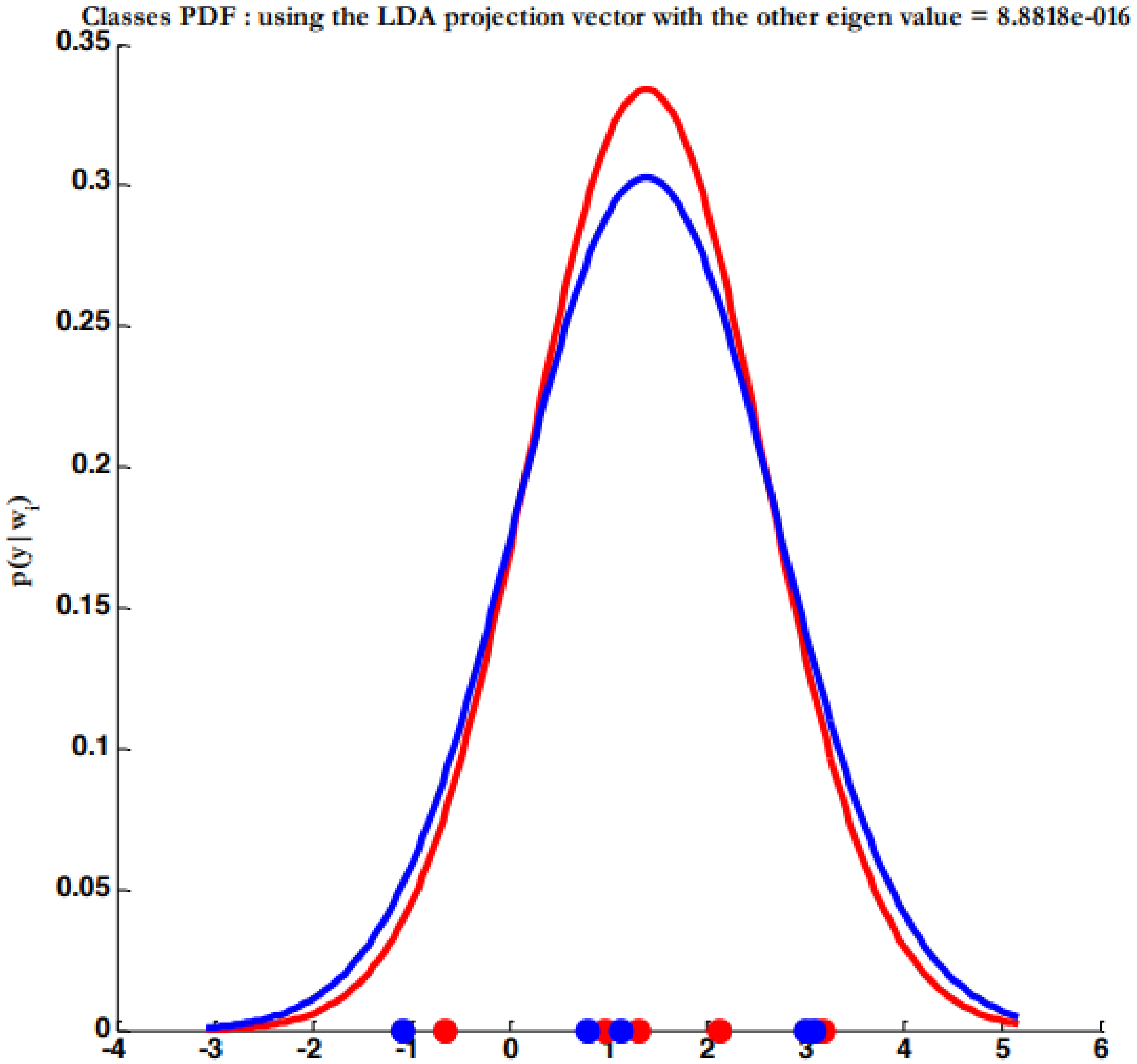


Using this vector leads to **good separability** between the two classes

LDA - Projection

The projection vector corresponding to the **smallest** eigen value


LDA projection vector with the other eigen value = 8.8818e-016


Classes PDF : using the LDA projection vector with the other eigen value = 8.8818e-016

Using this vector leads to **bad separability** between the two classes