DISCRIMINANT ANALYSIS METHODS

Soheil Kolouri

Center for Bioimage Informatics (CBI)
Biomedical Engineering Department
Carnegie Mellon University

Introduction

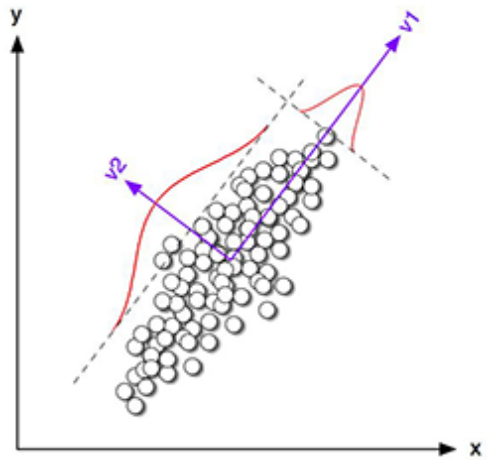
- Dimensionality reduction is a crucial concept in machine learning and data classification.
- The most famous example of dimensionality reduction is Principal Component Analysis(PCA):

Introduction

- Dimensionality reduction is a crucial concept in machine learning and data classification.
- The most famous example of dimensionality reduction is Principal Component Analysis(PCA):
  - Is an unsupervised method, so it doesn't include label information.
  - Searches for the directions the data have the largest variance.
  - There are difficulty issues with the number of principal components to choose.

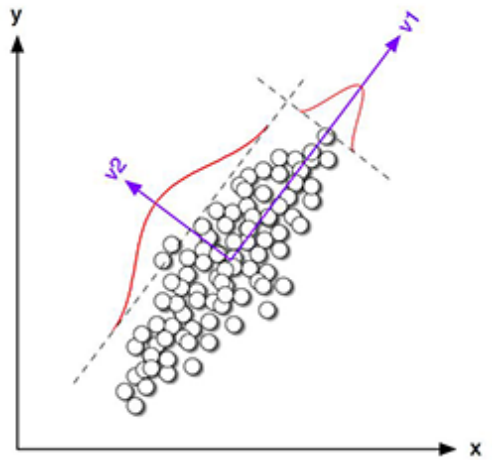
Introduction

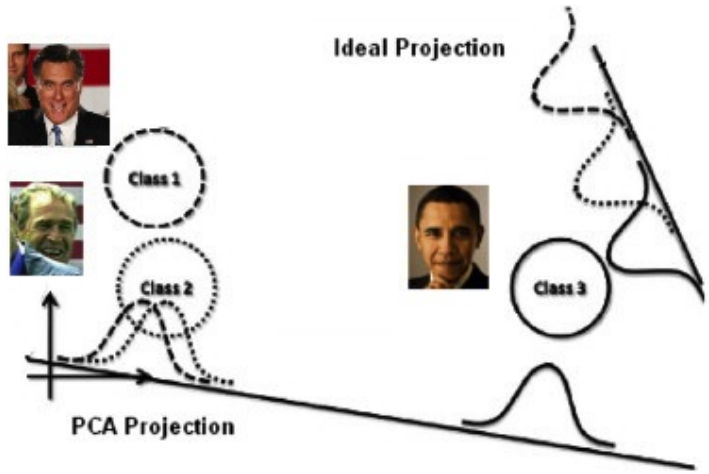
- Dimensionality reduction is a crucial concept in machine learning and data classification.
- The most famous example of dimensionality reduction is Principal Component Analysis(PCA):
  - Is an unsupervised method, so it doesn't include label information.
  - Searches for the directions the data have the largest variance.
  - There are difficulty issues with the number of principal components to choose.



Introduction

- Dimensionality reduction is a crucial concept in machine learning and data classification.
- The most famous example of dimensionality reduction is Principal Component Analysis(PCA):
  - Is an unsupervised method, so it doesn't include label information.
  - Searches for the directions the data have the largest variance.
  - There are difficulty issues with the number of principal components to choose.





Introduction

- discriminant analysis methods can be good candidates to address such problems.
  - These methods are supervised, so they include label information.
  - The goal is to find directions on which the data is best separable.
- One of the very well-known discriminant analysis method is the Linear Discriminant Analysis (LDA).

Introduction

- discriminant analysis methods can be good candidates to address such problems.
  - These methods are supervised, so they include label information.
  - The goal is to find directions on which the data is best separable.
- One of the very well-known discriminant analysis method is the Linear Discriminant Analysis (LDA).

LDA:

- Suppose that we have N number of data points $\mathbf{x}_i$, for $i = 1, \ldots, N$, which belong to c known classes $L_1, \ldots, L_c$.
- Then, the question is, how to utilize the label information to find informative directions?

Introduction

- discriminant analysis methods can be good candidates to address such problems.
  - These methods are supervised, so they include label information.
  - The goal is to find directions on which the data is best separable.
- One of the very well-known discriminant analysis method is the Linear Discriminant Analysis (LDA).

LDA:

- Suppose that we have N number of data points $\mathbf{x}_i$, for $i = 1, \ldots, N$, which belong to c known classes $L_1, \ldots, L_c$.
- Then, the question is, how to utilize the label information to find informative directions?
- LDA answers this question and suggests to find directions which maximize the following equation

$$\mathbf{w}^* = \underset{\mathbf{w}}{\operatorname{argmax}}\left(\frac{\mathbf{w}^T S_b \mathbf{w}}{\mathbf{w}^T S_w \mathbf{w}}\right)$$

LDA:

- Lets first define between class scatter matrix, S_b, and within class scatter matrix, S_w:

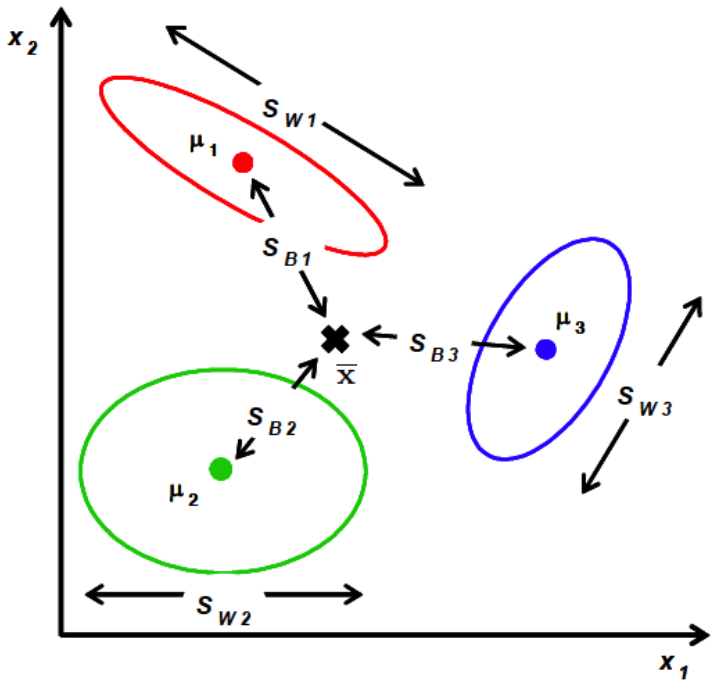
$$S_b = \frac{1}{N}\sum_{i=1}^{c} l_i S_{bi}$$

$$S_{bi} = (\boldsymbol{\mu}_i - \overline{\mathbf{x}})(\boldsymbol{\mu}_i - \overline{\mathbf{x}})^T$$

where l_i is the number of data points in class i and we have $\sum_{i=1}^{c} l_i = N$, $\boldsymbol{\mu}_i$ is the mean point of the i'th class, and $\overline{\mathbf{x}}$ is the mean of all data points.

$$S_w = \frac{1}{N}\sum_{i=1}^{c} S_{wi}$$

$$S_{wi} = \sum_{j \in L_i} (\mathbf{x}_j - \boldsymbol{\mu}_i)(\mathbf{x}_j - \boldsymbol{\mu}_i)^T$$



- Note that

$$trace(S_{bi}) = ||\boldsymbol{\mu}_i - \overline{\mathbf{x}}||_2^2$$

$$trace(S_{wi}) = \sum_{j \in L_i} ||\mathbf{x}_j - \boldsymbol{\mu}_i||_2^2$$

LDA:

- For S_b we can write,

$$\text{trace}(S_b) = \frac{1}{N}\sum_{i=1}^{c} l_i ||\boldsymbol{\mu}_i - \overline{\mathbf{x}}||_2^2$$

which is a measure for the distances of the center (mean) of classes from the center (mean) of the whole data set.

- For S_w we can write

$$\text{trace}(S_w) = \frac{1}{N}\sum_{i=1}^{c}\sum_{j\in L_i} ||\mathbf{x}_j - \boldsymbol{\mu}_i||_2^2$$

which corresponds to the average of the variances of all classes.

- Now assume that we project our data to a lower dimension using $\boldsymbol{W}^T\mathbf{x}$ (Projecting the data into the column space of $\boldsymbol{W}$). then the same scatter matrices can be defined in lower dimension:

$$\begin{aligned} S_b^W &= \boldsymbol{W}^T S_b \boldsymbol{W} \\ S_w^W &= \boldsymbol{W}^T S_w \boldsymbol{W} \end{aligned}$$

LDA:

- LDA aims to maximize the between class distance and minimize the within class distant in the dimensionality-reduced space.
- This can be achieved by minimizing $trace(S_w^W)$ and maximizing $trace(S_b^W)$. Which can be formulated as

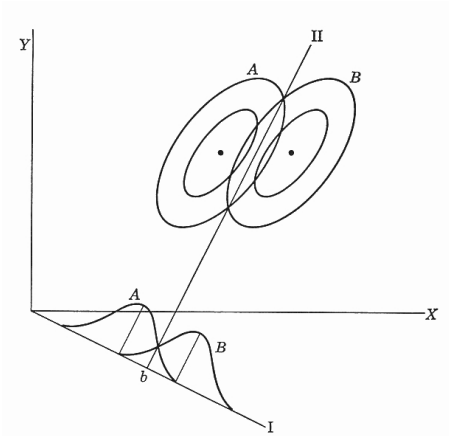
$$\boldsymbol{W}^* = \underset{\boldsymbol{W}}{\operatorname{argmax}} \left(trace((S_w^W)^{-1} S_b^W)\right)$$

- Above equation can be written in the form of the generalized Rayleigh quotient (generalized eigenvalue problem) and be solved by

$$S_w^{-1} S_b \mathbf{w}_i^* = \lambda_i \mathbf{w}_i^*$$

- where $\boldsymbol{W}^* = [\mathbf{w}_1^*, \ldots, \mathbf{w}_{c-1}^*]$.

- The figure on the right shows the LDA direction for the case of $c = 2$ classes (binary classification).



Limitations of LDA:

- LDA produces at most $c-1$ discriminant vectors. Proof:

$$rank(S_w^{-1}S_b) = min[rank(S_w^{-1}), rank(S_b)] = rank(S_b)$$

Note that for a large training data we expect S_w to be full rank and therefore invertible, and the rank constraint in this problem comes from S_b. S_b can be written as follows,

$$S_b = \begin{bmatrix} (\boldsymbol{\mu}_1 - \overline{\mathbf{x}}) & (\boldsymbol{\mu}_2 - \overline{\mathbf{x}}) & \dots & (\boldsymbol{\mu}_c - \overline{\mathbf{x}}) \end{bmatrix} \begin{bmatrix} \frac{l_1}{N} & 0 & \cdots & 0 \\ 0 & \frac{l_2}{N} & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & \frac{l_c}{N} \end{bmatrix} \begin{bmatrix} (\boldsymbol{\mu}_1 - \overline{\mathbf{x}})^T \\ (\boldsymbol{\mu}_2 - \overline{\mathbf{x}})^T \\ \vdots \\ (\boldsymbol{\mu}_c - \overline{\mathbf{x}})^T \end{bmatrix}$$

Therefore

$$rank(S_b) = rank(\begin{bmatrix} (\boldsymbol{\mu}_1 - \overline{\mathbf{x}}) & (\boldsymbol{\mu}_2 - \overline{\mathbf{x}}) & \dots & (\boldsymbol{\mu}_c - \overline{\mathbf{x}}) \end{bmatrix})$$

Also note that

$$\sum_{i=1}^{c} l_i(\boldsymbol{\mu}_i - \overline{\mathbf{x}}) = 0$$

Limitations of LDA:

- LDA produces at most $c-1$ discriminant vectors. Proof:

$$rank(S_w^{-1}S_b) = min[rank(S_w^{-1}), rank(S_b)] = rank(S_b)$$

Note that for a large training data we expect S_w to be full rank and therefore invertible, and the rank constraint in this problem comes from S_b. S_b can be written as follows,

$$S_b = \begin{bmatrix} (\boldsymbol{\mu}_1 - \overline{\mathbf{x}}) & (\boldsymbol{\mu}_2 - \overline{\mathbf{x}}) & \dots & (\boldsymbol{\mu}_c - \overline{\mathbf{x}}) \end{bmatrix} \begin{bmatrix} \frac{l_1}{N} & 0 & \cdots & 0 \\ 0 & \frac{l_2}{N} & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & \frac{l_c}{N} \end{bmatrix} \begin{bmatrix} (\boldsymbol{\mu}_1 - \overline{\mathbf{x}})^T \\ (\boldsymbol{\mu}_2 - \overline{\mathbf{x}})^T \\ \vdots \\ (\boldsymbol{\mu}_c - \overline{\mathbf{x}})^T \end{bmatrix}$$

Therefore

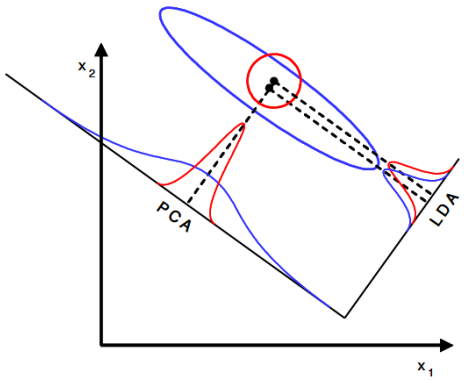
$$rank(S_b) = rank(\begin{bmatrix} (\boldsymbol{\mu}_1 - \overline{\mathbf{x}}) & (\boldsymbol{\mu}_2 - \overline{\mathbf{x}}) & \dots & (\boldsymbol{\mu}_c - \overline{\mathbf{x}}) \end{bmatrix})$$

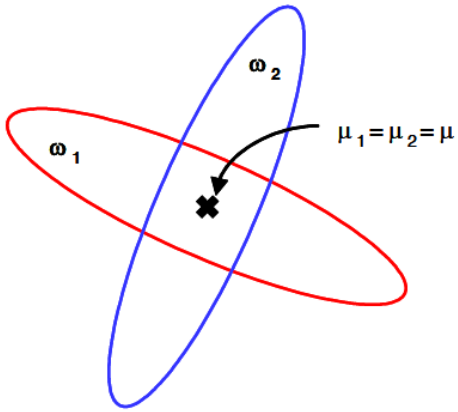
Also note that

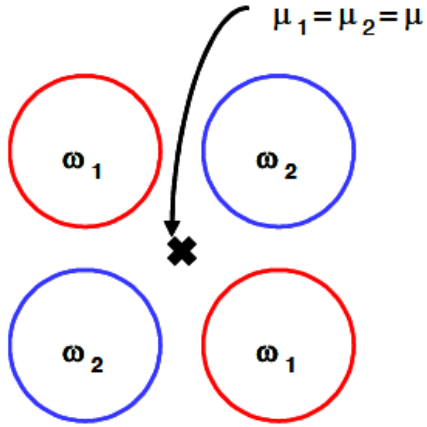
$$\sum_{i=1}^{c} l_i(\boldsymbol{\mu}_i - \overline{\mathbf{x}}) = 0 \;\Rightarrow\; rank(S_b) \leq c-1$$

Limitations of LDA:

- LDA will fail if the discriminatory information is not in the mean but in the variance of the data







- LDA is not stable for problems with small sample size. For small number of samples S_w may become singular and therefore non-invertible. (This problem was initially addressed by adding a diagonal matrix, αI with very small α, to S_w. The geometry of this method is explained in the Penalized LDA paper.)

Limitations of LDA:

- For a binary classification problem where $c = 2$, LDA provides just one discriminant direction.
- In order to achieve additional discriminant directions, an intuitive approach is to take the first discriminant direction out of data (Project the data onto the largest subspace orthogonal to the first discriminant) and then take the LDA of projected data to find the second direction. For the second step this can be done as follows,

$$\mathbf{x}_i^2 = \mathbf{x}_i^1 - \frac{\mathbf{w}_1^T \mathbf{x}_i^1}{\mathbf{w}_1^T \mathbf{w}_1} \mathbf{w}_1$$

- Calculate the scatter matrices, $S_{b,2}$ and $S_{w,2}$ based on the projected data and find $\mathbf{w}_2$ solving the following,

$$S_{w,2}^{-1} S_{b,2} \mathbf{w}_2 = \lambda \mathbf{w}_2$$

- These steps can be continued until adequate number of discriminant directions is achieved.

MMC:

- If some distance metric is used to measure the dissimilarity, we would hope that a pattern is close to those in the same class but far from those in different classes.
- Therefore a good feature extractor should maximize the distances between classes after the transformation. In general we can write,

$$J(\boldsymbol{W}) = \frac{1}{2}\sum_{i=1}^{c}\sum_{j=1}^{c} l_i l_j d_{\boldsymbol{W}}(L_i, L_j)$$

  where $d_{\boldsymbol{W}}(L_i, L_j)$ is the distance between projections of i'th and j'th classes onto the lower dimension spanned by $\boldsymbol{W}$.
- One may use the distance between mean vectors as the distance between classes

$$d_{\boldsymbol{W}}(L_i, L_j) = d_{\boldsymbol{W}}(\boldsymbol{\mu}_i, \boldsymbol{\mu}_j)$$

- However, similar to the LDA, the means are not enough to separate the data. Even if the distance between the mean vectors is large, it might not be easy to separate two classes that have the large spread and overlap.

$$d_{\boldsymbol{W}}(L_i, L_j) = d_{\boldsymbol{W}}(\boldsymbol{\mu}_i, \boldsymbol{\mu}_j) - S_{\boldsymbol{W}}(L_i) - S_{\boldsymbol{W}}(L_j)$$

MMC:

- But we have already defined the measures for the distances between the projected mean of classes and the scatter of their projections.

$$J(\boldsymbol{W}) = trace(S_b^W - S_w^W) = trace(\boldsymbol{W}^T(S_b - S_w)\boldsymbol{W})$$

- Without losing generality we can write $\boldsymbol{W} = [\mathbf{w}_1, \ldots, \mathbf{w}_M]$ and assume that $||\mathbf{w}_i||_2^2 = 1$. therefore we will have an eigenvalue problem

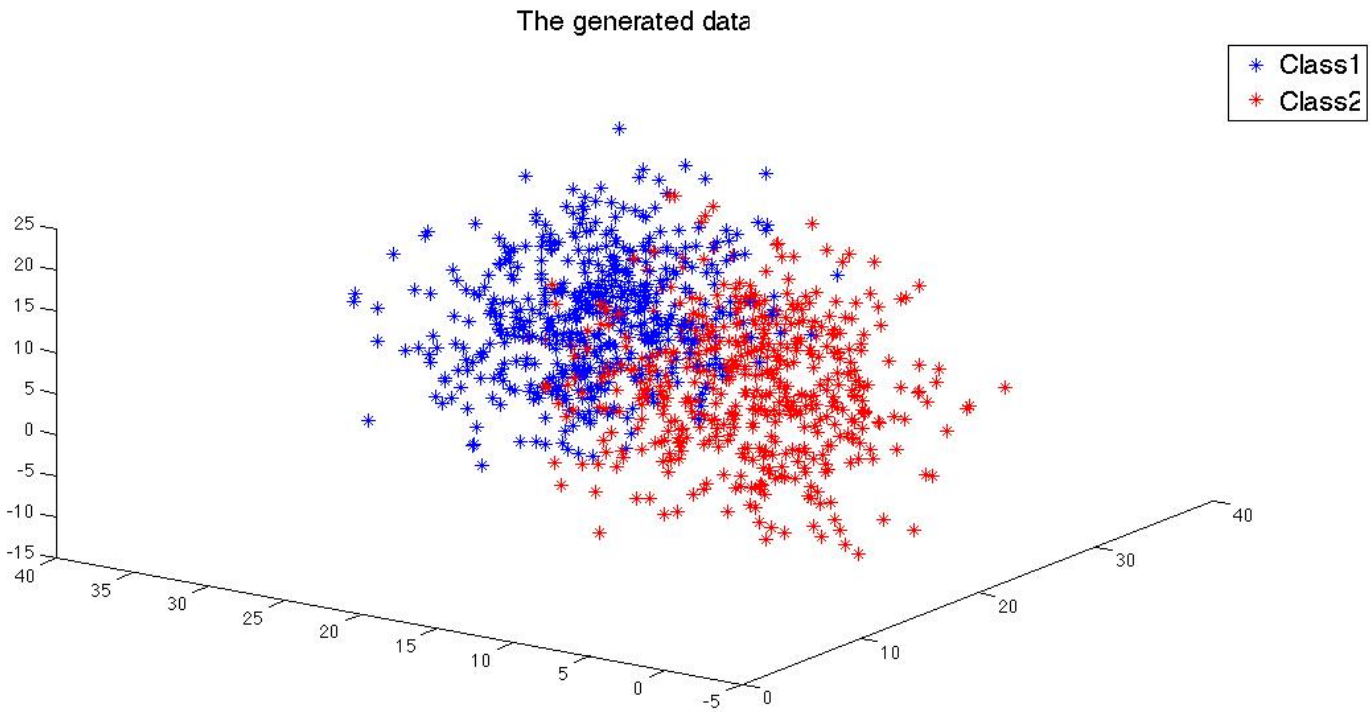
$$\mathbf{w}^* = \underset{\mathbf{w}}{\text{argmax}}\left(\frac{\mathbf{w}^T(S_b - S_w)\mathbf{w}}{\mathbf{w}^T\mathbf{w}}\right) \Rightarrow (S_b - S_w)\mathbf{w} = \lambda\mathbf{w}$$

- Also note that the low rank constraint ('c-1' discriminant dimensions) in LDA, caused by S_b, is lifted for MMC formulation.

$$rank(S_b - S_w) \leq rank(S_b) + rank(S_w)$$

Simulated Data:

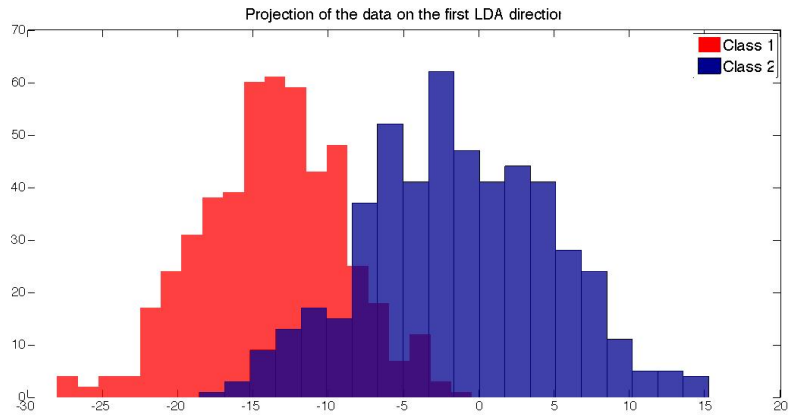
- The data set looks as follows,



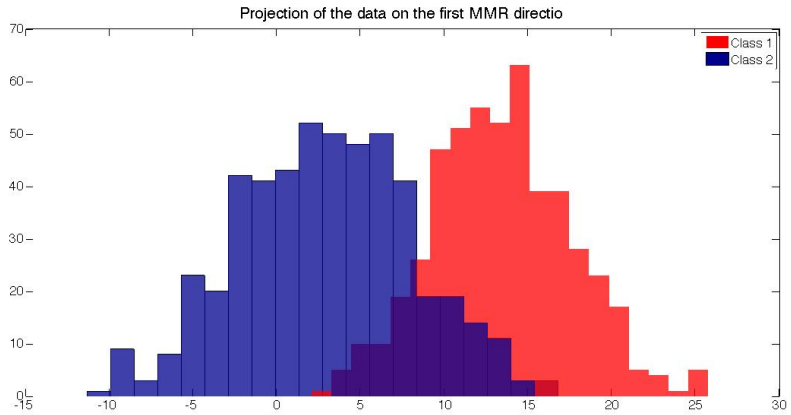
Results:

- The projected data on the first direction of Stepwise-LDA and MMC,

Stepwise-LDA



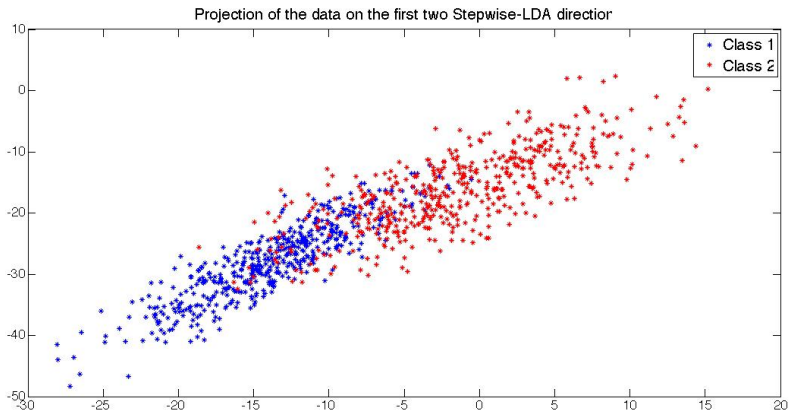
MMC



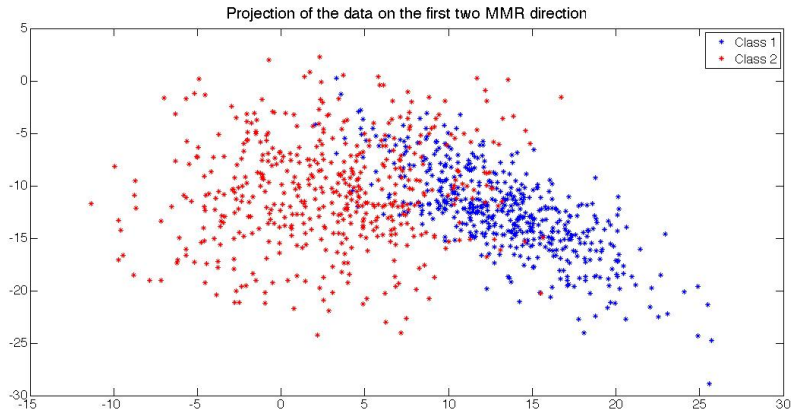
Results:

- The projected data on the first two directions of Stepwise-LDA and MMC,

Stepwise-LDA



MMC



Thank You!