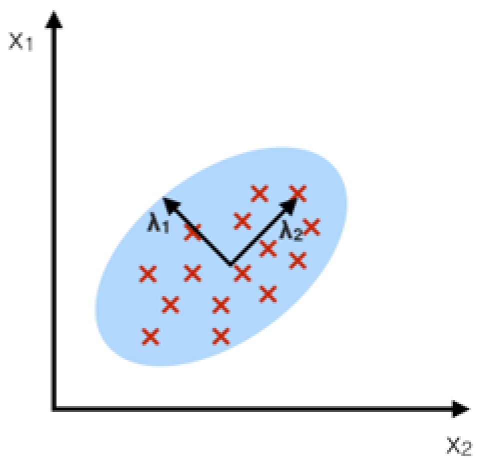
Linear Discriminant Analysis

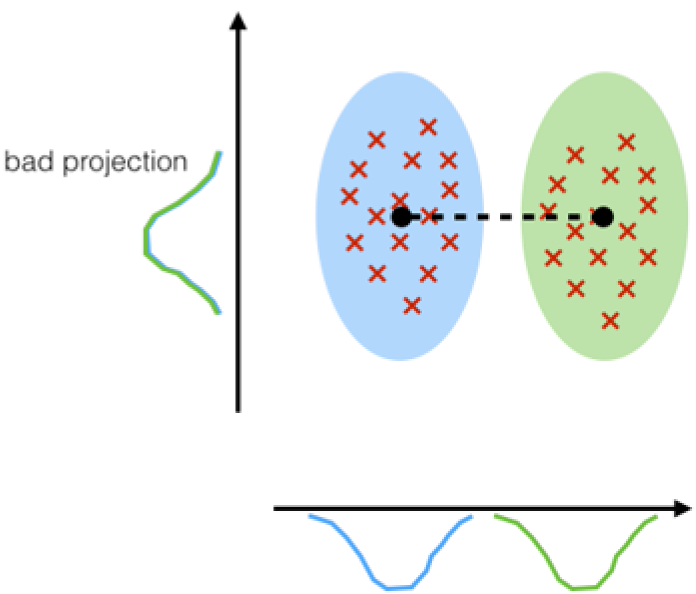
COEN140

Santa Clara University

PCA vs LDA

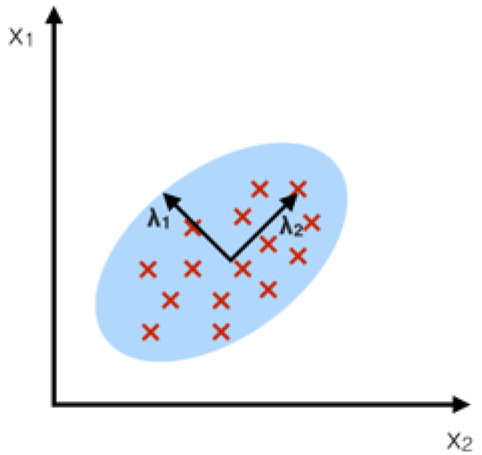
- PCA: select the component axis that maximizes data variance
- LDA: select the component axis to separate classes

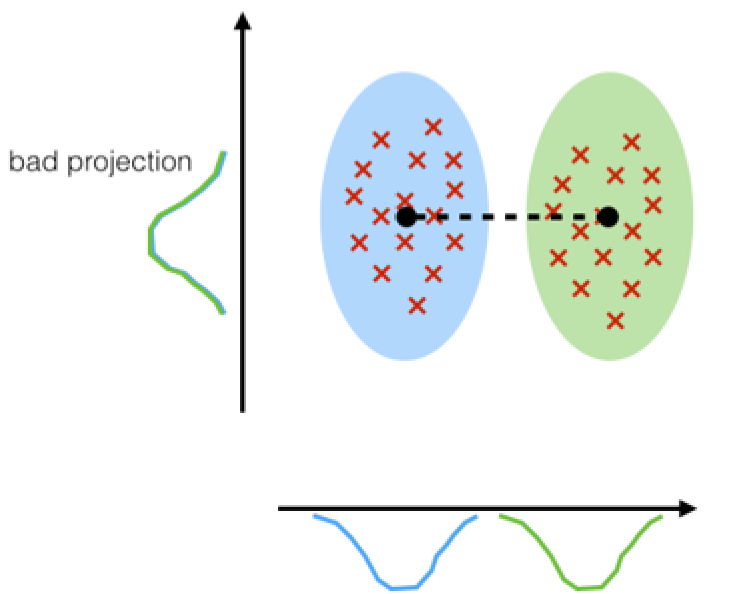




PCA vs LDA

- PCA: unsupervised learning
- LDA: supervised learning





Projection Direction

- Assume two classes
  - C_1: N_1 data samples
  - C_2: N_2 data samples
- Class mean
  - $\mathbf{m}_1 = \frac{1}{N_1} \sum_{n \in C_1} \mathbf{x}_n$
  - $\mathbf{m}_2 = \frac{1}{N_2} \sum_{n \in C_2} \mathbf{x}_n$

Projection Direction

- The mean of the projected data from class C_1

$$m_1 = \mathbf{w}^T \mathbf{m}_1$$

- The mean of the projected data from class C_2

$$m_2 = \mathbf{w}^T \mathbf{m}_2$$

Projection Direction

- A measure of the separation of the classes when projected onto $\mathbf{w}$

$$m_2 - m_1 = \mathbf{w}^T(\mathbf{m}_2 - \mathbf{m}_1)$$

- **Makes sense to maximize** $m_2 - m_1$

- The expression can be made arbitrarily large simply by increasing the magnitude of $\mathbf{w}$
- Constrain $\mathbf{w}$ to have unit length

$$\sum_i w_i^2 = 1$$

Projection Direction

- **Problem: to maximize** $m_2 - m_1$

$$\arg\max_{\mathbf{w}} \mathbf{w}^T(\mathbf{m}_2 - \mathbf{m}_1)$$
$$\text{Subject to } \mathbf{w}^T\mathbf{w} = 1$$

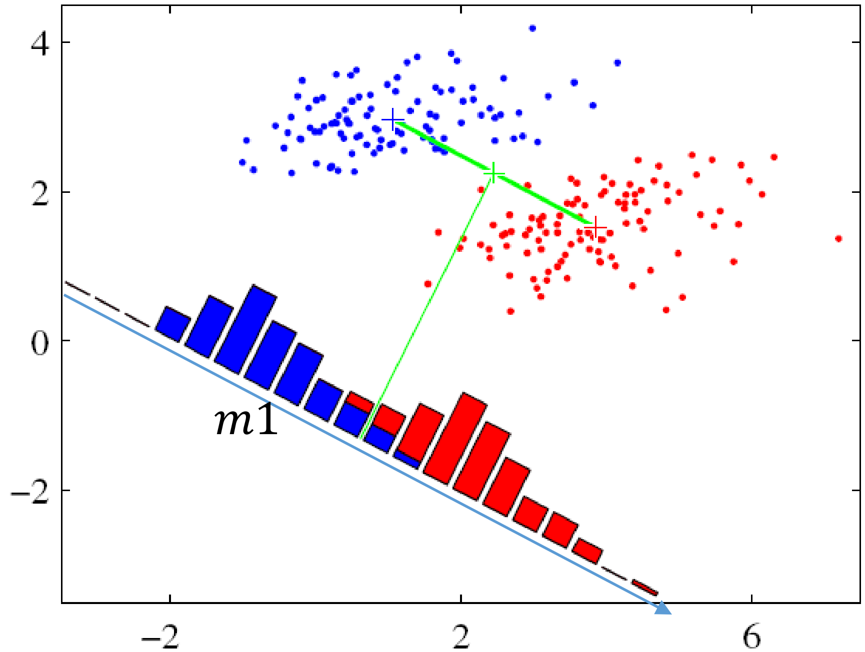
- Use the method of Lagrange multiplier, we find

$$\mathbf{w} \propto (\mathbf{m}_2 - \mathbf{m}_1)$$

- Derivation: LDA_notes.pdf

Projection Direction

- Result



- The mean in the projection space are well separated
- But the data points in the projection space still have big overlap.

Linear Discriminant Analysis (LDA)

- Also called "Fisher's Linear Discriminant".
- Maximize a function that will give a large separation **between** the projected class means, while giving a small variance **within** each class, thereby minimizing the class overlap.
- Within-class variance in the projection space

$$s_k^2 = \sum_{n \in C_k} (y_n - m_k)^2$$

$$y_n = \mathbf{w}^T \mathbf{x}_n, m_k = \mathbf{w}^T \mathbf{m}_k$$

Class index $k = 1, 2$

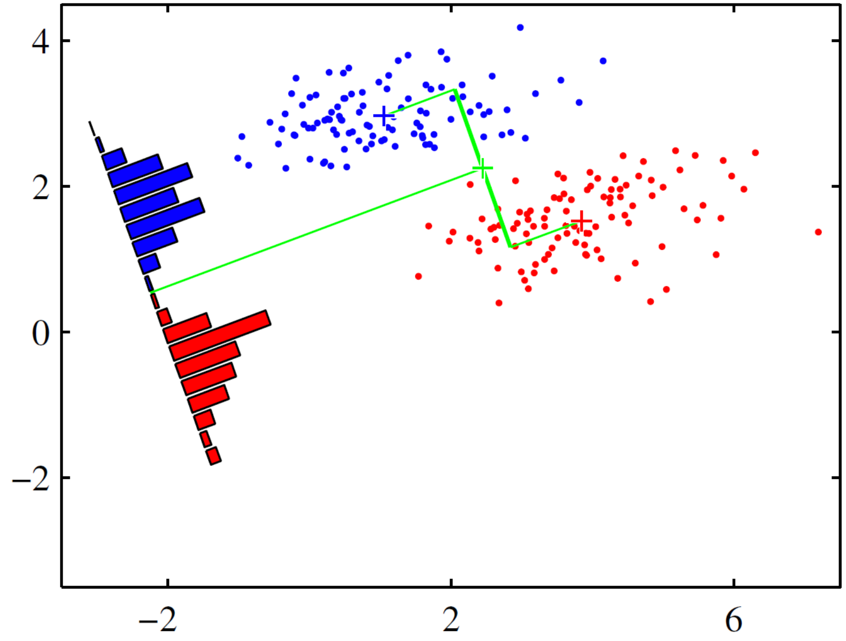
LDA: Two-Class

- Total within-class variance

$$s_1^2 + s_2^2$$

- Fisher criterion: to maximize

$$J(\mathbf{w}) = \frac{(m_2 - m_1)^2}{s_1^2 + s_2^2}$$



LDA: Two-Class

- Fisher criterion: to maximize

$$J(\mathbf{w}) = \frac{(m_2 \ - m_1 \ )^2}{s_1^2 + s_2^2} = \frac{\mathbf{w}^T \mathbf{S}_B \mathbf{w}}{\mathbf{w}^T \mathbf{S}_W \mathbf{w}}$$

- Between-class covariance matrix

$$\mathbf{S}_{\mathrm{B}} = (\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^{\mathrm{T}}$$

- Total within-class covariance matrix

$$\mathbf{S}_{\mathrm{W}} = \sum_{n \in \mathcal{C}_1} (\mathbf{x}_n - \mathbf{m}_1)(\mathbf{x}_n - \mathbf{m}_1)^{\mathrm{T}} + \sum_{n \in \mathcal{C}_2} (\mathbf{x}_n - \mathbf{m}_2)(\mathbf{x}_n - \mathbf{m}_2)^{\mathrm{T}}$$

LDA: Two-Class

- Fisher criterion: to maximize

$$J(\mathbf{w}) = \frac{\mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{B}}\mathbf{w}}{\mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{W}}\mathbf{w}}$$

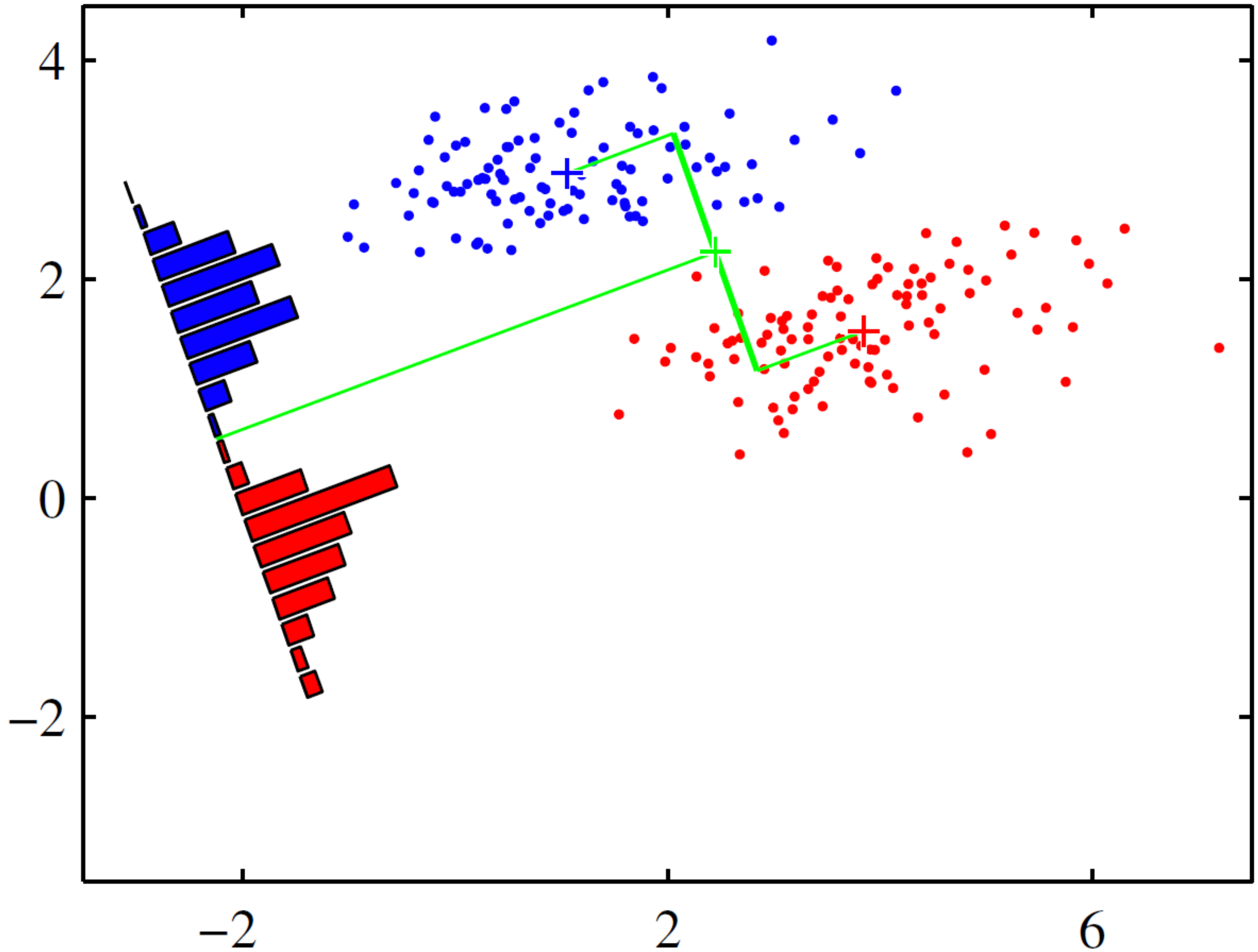
- Taking the derivative of $J(\mathbf{w})$ with respect to $\mathbf{w}$, and set it as $\mathbf{0}$. Solve for $\mathbf{w}$, we find

$$\mathbf{w} \propto \mathbf{S}_{\mathrm{W}}^{-1}(\mathbf{m}_2 - \mathbf{m}_1)$$

$\mathbf{S}_{\mathrm{W}}$ need to be rank-D

- Derivation: LDA_notes.pdf

LDA: Two-Class



LDA: Two-Class

- We know

$$\mathbf{S}_{\mathrm{W}} = \sum_{n \in \mathcal{C}_1} (\mathbf{x}_n - \mathbf{m}_1)(\mathbf{x}_n - \mathbf{m}_1)^{\mathrm{T}} + \sum_{n \in \mathcal{C}_2} (\mathbf{x}_n - \mathbf{m}_2)(\mathbf{x}_n - \mathbf{m}_2)^{\mathrm{T}}$$

- Assume $\mathbf{x}_n \in \mathbb{R}^D$
  - $\mathbf{S}_{\mathrm{W}}$: $D \times D$ matrix
  - Rank($\mathbf{S}_{\mathrm{W}}$)$\leq \min\{D, N_1 + N_2 - 2\}$

$$\mathbf{w} \propto \mathbf{S}_{\mathrm{W}}^{-1}(\mathbf{m}_2 - \mathbf{m}_1)$$

$\mathbf{S}_{\mathrm{W}}$ need to be rank-D

LDA: Multiple-Class

- $K > 2$ classes
- Dimensionality of the data sample $\mathbf{x}$: D

- Find a vector $\mathbf{w}$ to project the data sample $\mathbf{x}$

$$\mathrm{y} = \mathbf{w}^T \mathbf{x}$$

LDA: Multiple-Class

- Within-class covariance matrix

$$\mathbf{S}_W = \sum_{k=1}^{K} \sum_{n \in C_k} (\mathbf{x}_n - \mathbf{m}_k)(\mathbf{x}_n - \mathbf{m}_k)^T$$

Mean of class-k: $\mathbf{m}_k = \frac{1}{N_k} \sum_{n \in C_k} \mathbf{x}_n$

N_k: the number of data samples in class-k

LDA: Multiple-Class

- Between-class covariance matrix:

$$\mathbf{S}_B = \sum_{k=1}^{K} N_k(\mathbf{m}_k - \mathbf{m})(\mathbf{m}_k - \mathbf{m})^T$$

The mean of class-k: $\mathbf{m}_k = \frac{1}{N_k} \sum_{n \in C_k} \mathbf{x}_n$

The mean of all data samples: $\mathbf{m} = \frac{1}{N} \sum_{n=1}^{N} \mathbf{x}_n$

$$N = N_1 + N_2 + \cdots + N_K$$

LDA: Multiple-Class

- Maximize the following objective function w.r.t. $\mathbf{w}$

$$J(\boldsymbol{w}) = \frac{\mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{B}}\mathbf{w}}{\mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{W}}\mathbf{w}}$$

- Solution: $\mathbf{w}$ is given by the eigenvector of $\mathbf{S}_W^{-1}\mathbf{S}_B$, corresponding to the largest eigenvalue

$\mathbf{S}_{\mathrm{W}}$ need to be full-rank

- Derivation: LDA_notes.pdf

LDA: Multiple-Class

- Within-class covariance matrix

$$\mathbf{S}_W = \sum_{k=1}^{K} \sum_{n \in C_k} (\mathbf{x}_n - \mathbf{m}_k)(\mathbf{x}_n - \mathbf{m}_k)^T$$

$$\mathbf{m}_k = \frac{1}{N_k} \sum_{n \in C_k} \mathbf{x}_n$$

- rank($\mathbf{S}_\mathrm{W}$)$\leq \min\{D, N_1 + N_2 + \cdots + N_K - K\}$
- If $D > N_1 + N_2 + \cdots + N_K - K$,
  then $\mathbf{S}_\mathrm{W}$ is not invertible

LDA: Multiple-Class

- Maximize the following objective function w.r.t. $\mathbf{w}$

$$J(\boldsymbol{w}) = \frac{\mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{B}}\mathbf{w}}{\mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{W}}\mathbf{w}}$$

- If we want to find multiple projection vectors $[\mathbf{w}_1 \ , \mathbf{w}_2 \ , \dots , \mathbf{w}_d \ ]^T$, then these vectors are given by the top-d eigenvectors of $\mathbf{S}_W^{-1}\mathbf{S}_B$, corresponding to the d largest eigenvalues

What if $\mathbf{S}_W$ is not full rank?

- **Solution:**
- **Step 1:** reduce the dimension of data samples by PCA

  Use d_0 projection vectors
- $\mathbf{W}_{PCA} = [\mathbf{w}_1, \dots, \mathbf{w}_{d_0}]$: $D \times d_0$
- $\mathbf{Y} = \mathbf{W}_{PCA}^T \mathbf{X}$
  - $\mathbf{Y}$: $d_0 \times N$
  - $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_N]$: $D \times N$, columns are training data samples
  - $\mathbf{W}_{PCA}$: $D \times d_0$
  - $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_N]$: $d_0 \times N$, columns are the projected training data samples

What if $\mathbf{S}_W$ is not full rank?

- **Solution:**
- **Step 2:** then apply FLD/LDA to the reduced-dimensional data
- $\mathbf{Z} = \mathbf{W}_{FLD}^T \mathbf{Y}$
  - $\mathbf{Y}$: $d_0 \times N$
  - $\mathbf{Z}$: $d \times N$
  - $\mathbf{W}_{FLD} = [\mathbf{w}_1, \dots, \mathbf{w}_d]$: $d_0 \times d$
  - Note: $d \leq d_0$
- How to train $\mathbf{W}_{FLD}$? Use $\mathbf{Y}$ and the original class labels

Face Recognition Example

- 10 subjects
- Image size: 112x92, $D = 10304$
- Number of training samples per class:
- $N_k = 9, k = 1, 2, \ldots, 10$

- Dimensionality reduction
    - From D to d, $d = [1,2,3,6,10,20,30]$

- For FLD/LDA, first the data dimension is reduced to $d_0 = 40$ by PCA

Face Recognition Example

- Run 10 independent experiments
  - Each experiment has randomly chosen training images, and the test images are then automatically determined
- Result
  - Blue: FLD/LDA
  - Red: PCA
  - FLD performs better, especially for small d values