Data Compression

Compressing data
via dimensionality reduction

Potential Benefits

- Summarize & Visualize
- Reduce storage size
- Speedup learning algorithm
- Prevent overfitting

Types of Data Compression

Feature Selection

Select the most important features

Feature Extraction

Transform data into a new feature subspace

Types of Data Compression

Feature Selection

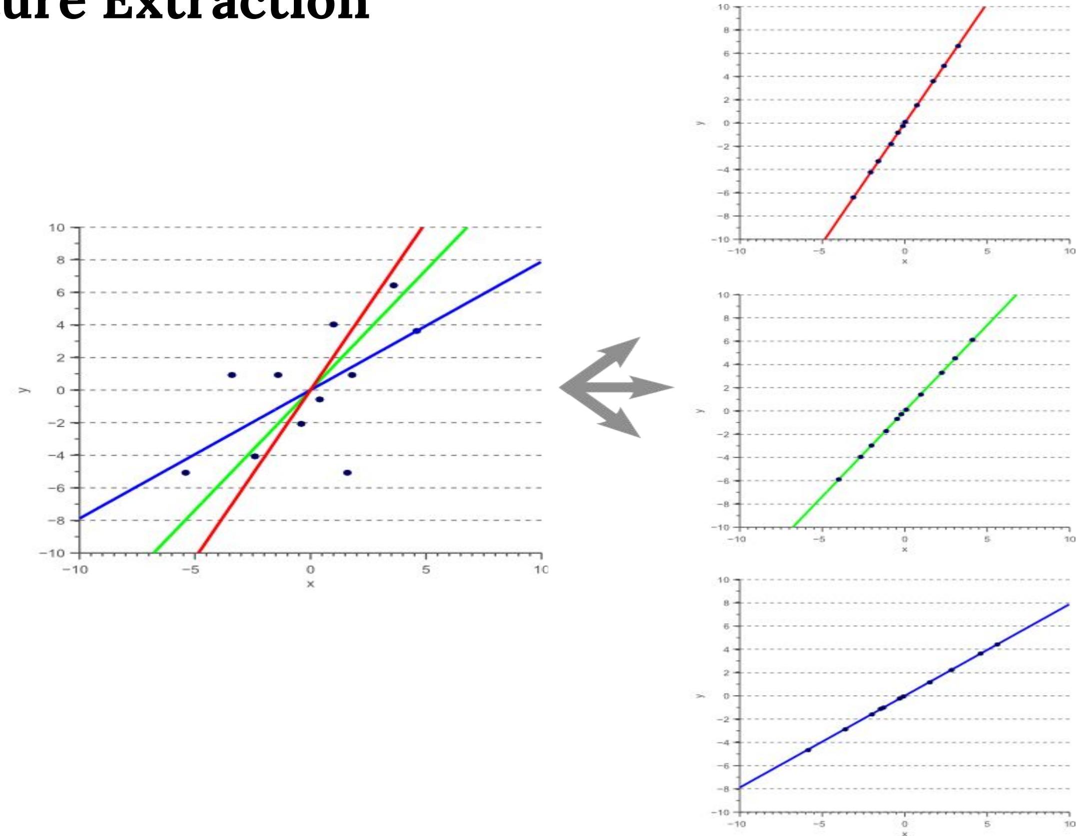
Select the most important features

Feature Extraction

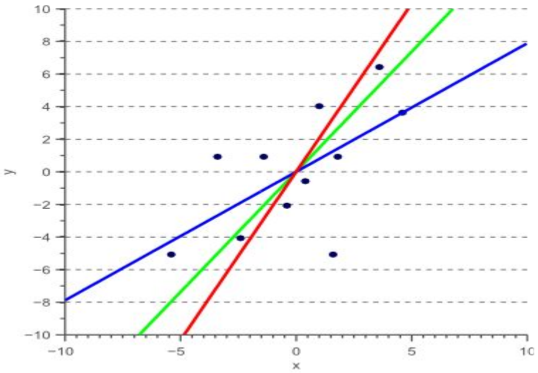
Transform data into a new feature subspace

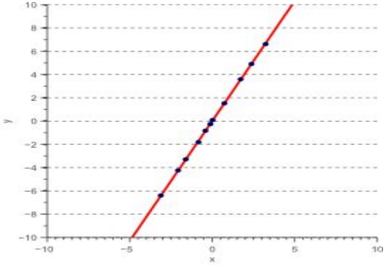
TODAY

Feature Extraction

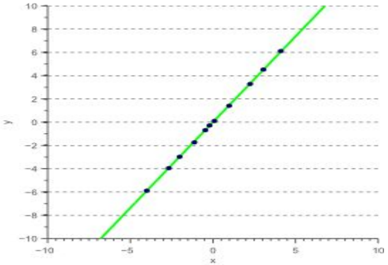


Feature Extraction

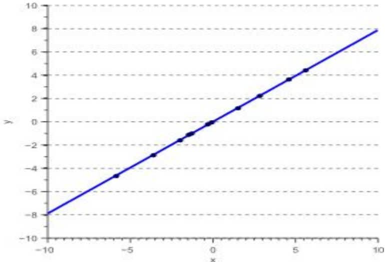




$$W = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$



$$W = \begin{bmatrix} 1.4 \\ 2 \end{bmatrix}$$

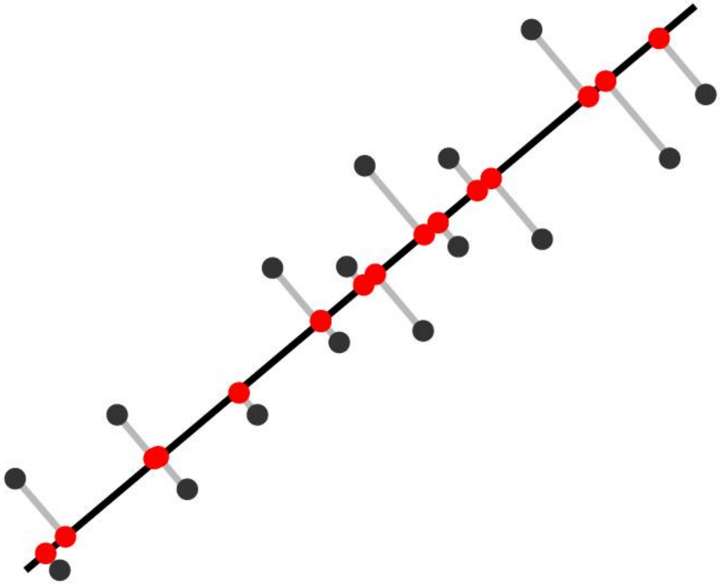


$$W = \begin{bmatrix} 2.5 \\ 2 \end{bmatrix}$$

Types of Feature Extraction

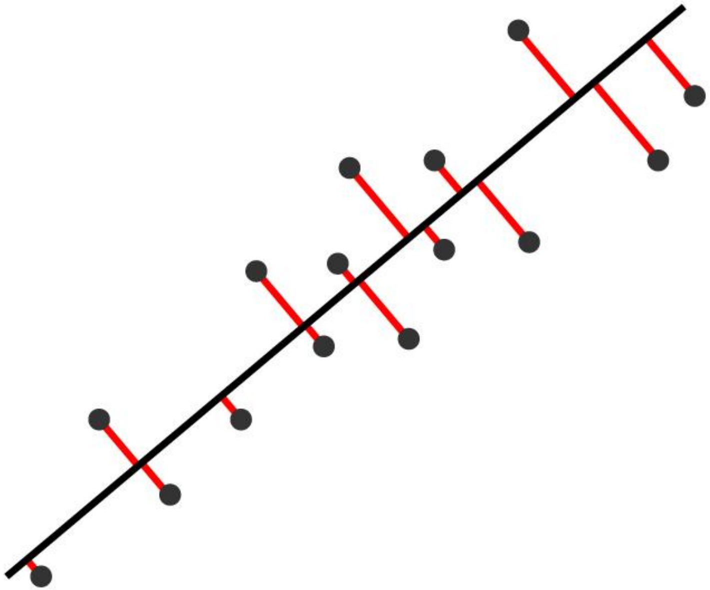
	Linear Transformation	Non-Linear Transformation
Unsupervised	PCA	KPCA
Supervised	LDA	KDA

Principal Component Analysis (PCA)



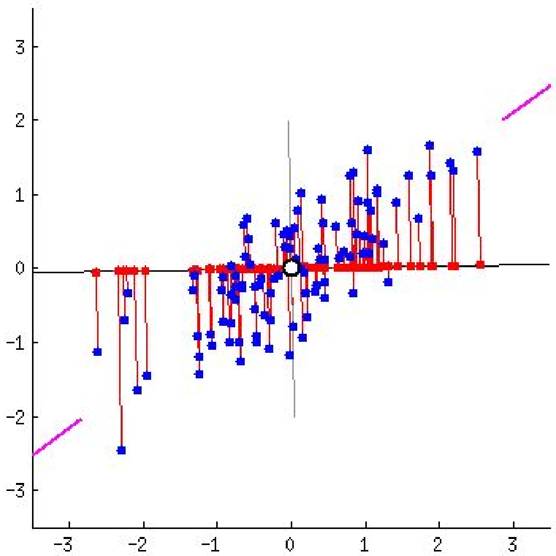
Principal Component

New feature axe that maximizes variance



Principal Component

New feature axe that minimize reconstruction error



Maximize Variance

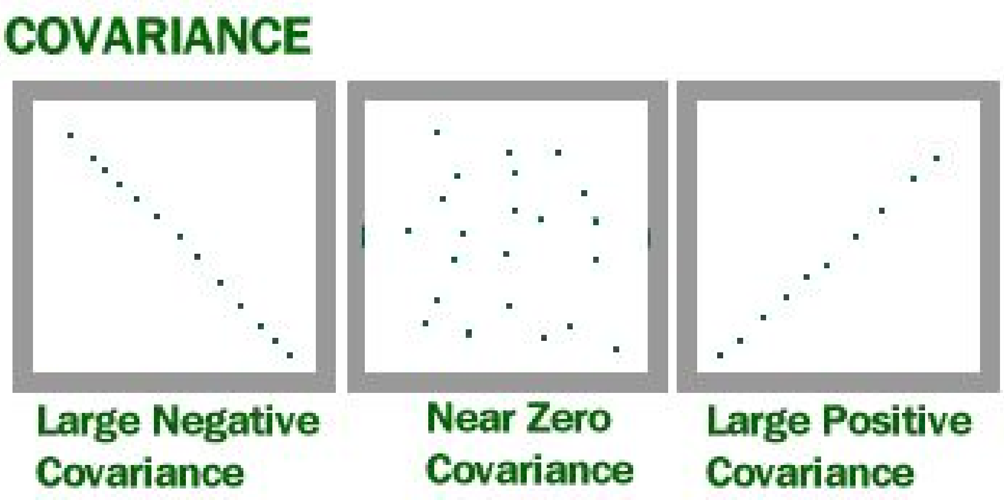
<=>

Minimize Reconstruction Error

Basic Math Review

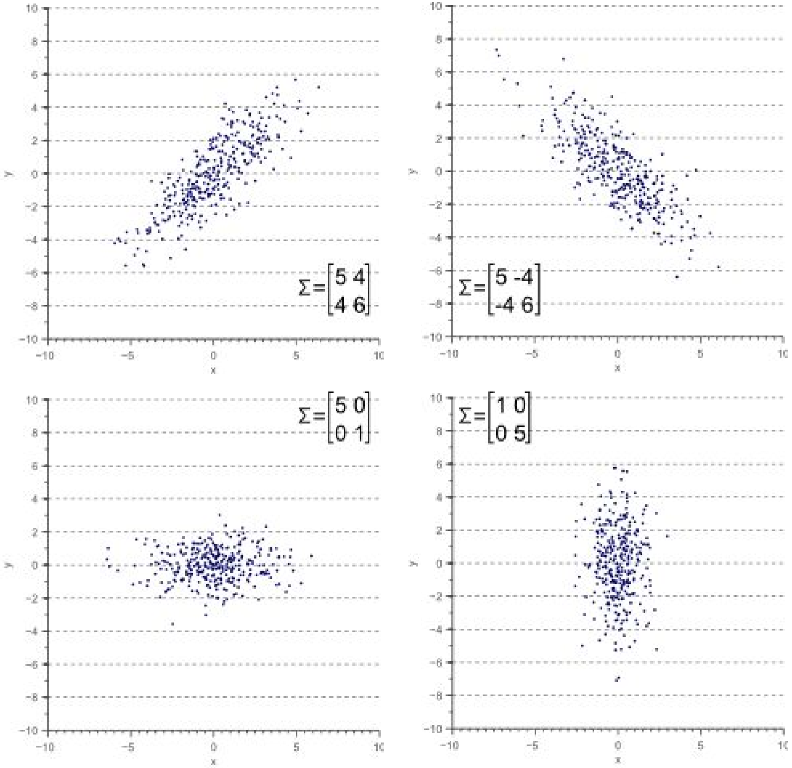
(Before starting finding Principal Component)

Covariance



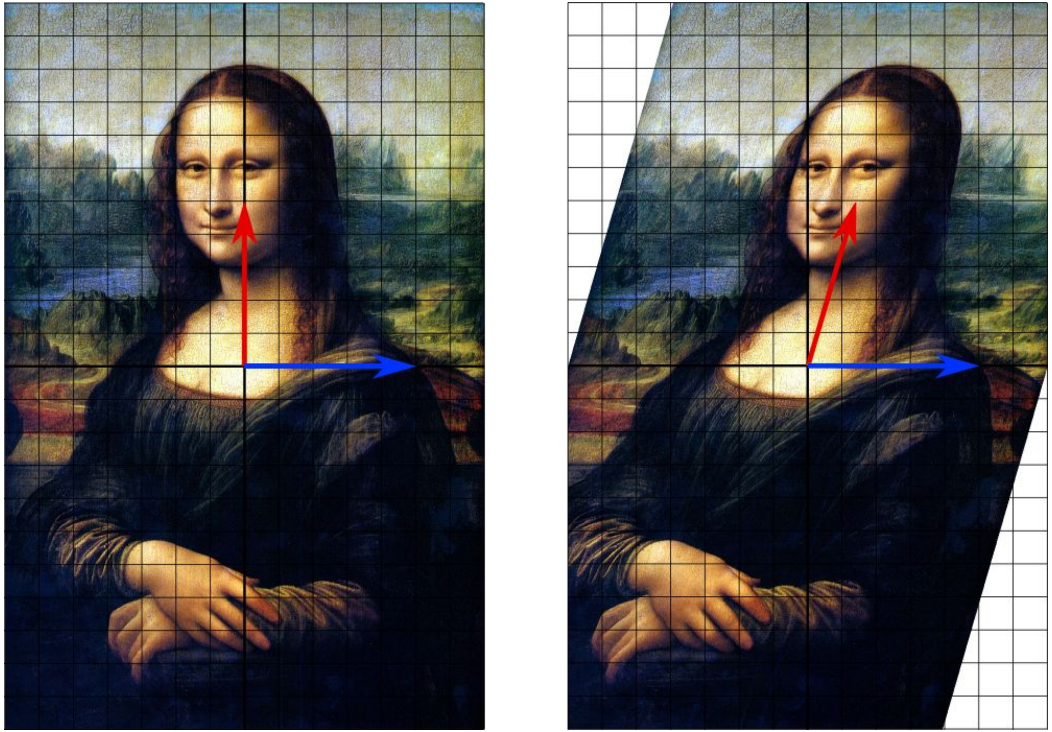
$$\sigma(x, y) = \frac{\sum (x-\bar{x})(y-\bar{y})}{(n-1)}$$

Covariance Matrix



$$\Sigma = \begin{bmatrix} \sigma(x,x) & \sigma(x,y) \\ \sigma(y,x) & \sigma(y,y) \end{bmatrix}$$

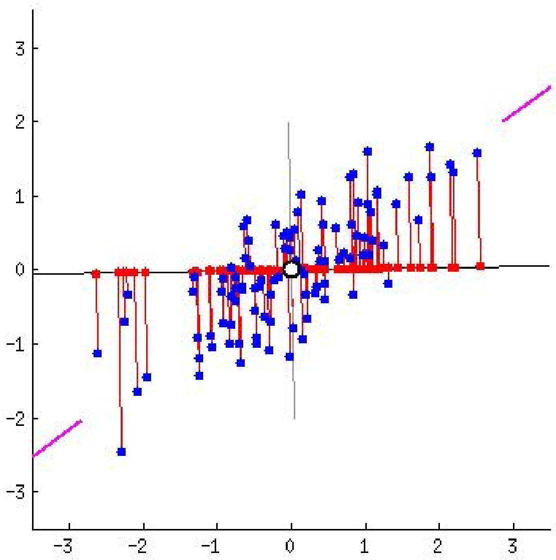
Eigenvectors & Eigenvalues



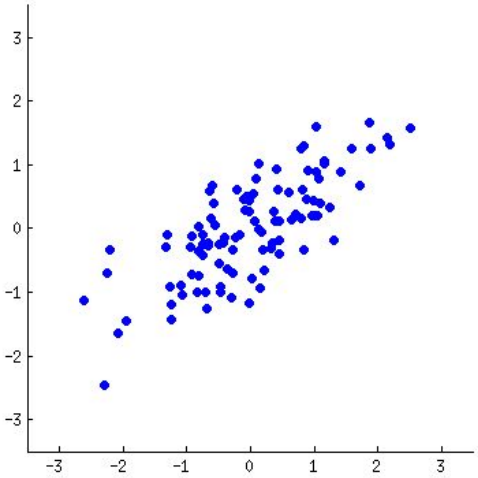
$$A\mathbf{v} = \lambda\mathbf{v}$$

Let’s Find Principal Component!

Example: 2 features -> 1 feature

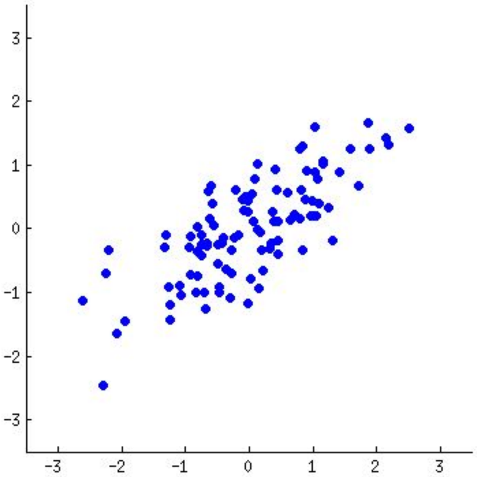


Construct the covariance matrix



$$\begin{bmatrix} 1.07 & 0.63 \\ 0.63 & 0.64 \end{bmatrix}$$

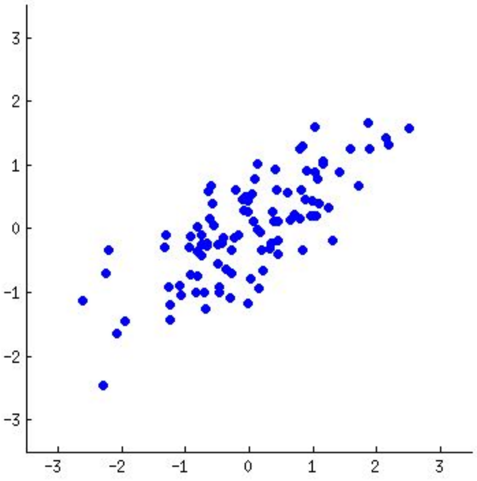
Find eigenvalues & eigenvectors from the covariance matrix



eigenvectors: $\begin{bmatrix} 0.81 \\ 0.58 \end{bmatrix}$ $\begin{bmatrix} -0.58 \\ 0.81 \end{bmatrix}$

eigenvalues: 1.52 0.19

Select the eigenvector with largest eigenvalue

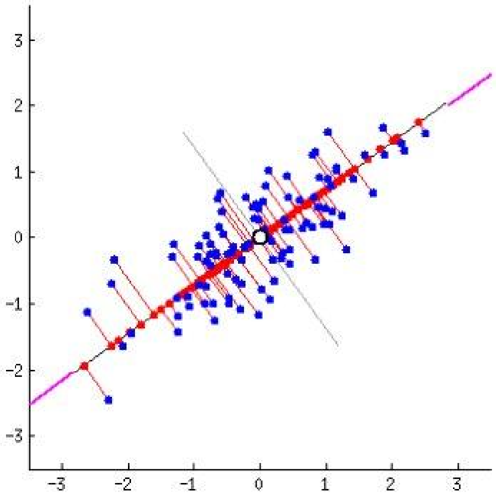


PC

$$\textit{eigenvectors:} \quad \boxed{\begin{bmatrix} 0.81 \\ 0.58 \end{bmatrix}} \quad \begin{bmatrix} -0.58 \\ 0.81 \end{bmatrix}$$

$$\textit{eigenvalues:} \quad 1.52 \quad\quad 0.19$$

Transform the data

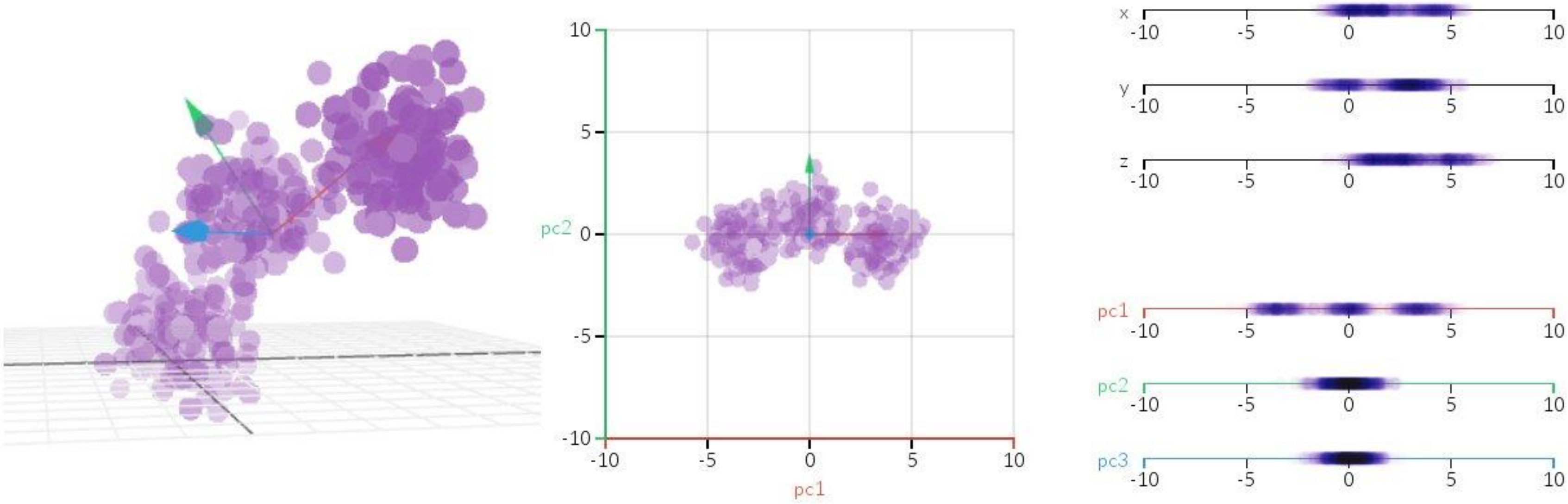


$$x' = [x \quad y]\begin{bmatrix}0.81\\0.58\end{bmatrix} = 0.81x + 0.58y$$

Proof:
https://goo.gl/STgLp4

Principal Component Analysis in General

d features -> k features



[Principal Component Analysis explained visually]()

Target: find the d × k transformation matrix

$$\boldsymbol{x} = [x_1, x_2, \ldots, x_d], \quad \boldsymbol{x} \in \mathbb{R}^{d}$$

$$\downarrow \boldsymbol{xW}, \quad \boldsymbol{W} \in \mathbb{R}^{d \times k}$$

$$\mathbf{z} = [z_1, z_2, \ldots, z_k], \quad \mathbf{z} \in \mathbb{R}^{k}$$

1. Standardize the dataset

$$x_{std}^{(i)} = \frac{x^{(i)} - \mu_x}{\sigma_x}$$

2. Construct the covariance matrix

$$\Sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1d} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{d1} & \sigma_{d2} & \cdots & \sigma_{dd} \end{bmatrix}$$

$$\sigma_{jk} = \frac{1}{(n-1)} \sum_{i=1}^{n} x_j^{(i)} x_k^{(i)}$$

2. Construct the covariance matrix

$$\Sigma = \sum_{i=1}^{n} (x^{(i)})^T x^{(i)}$$

** Each x is a 1 x d vector*

3. Find eigenvalues & eigenvectors from covariance matrix

eigenvectors: $v_1 \quad v_2 \quad \cdots \quad v_d$

eigenvalues: $\lambda_1 \quad \lambda_2 \quad \cdots \quad \lambda_d$

4. Select eigenvectors correspond to largest k eigenvalues

$$\begin{array}{lcccccc} \textit{eigenvectors:} & v_1 & \boxed{v_2} & \boxed{} & \cdots & v_d \\ \textit{eigenvalues:} & \lambda_1 & \lambda_2 & & \cdots & \lambda_d \\ & & \text{PC1} & \text{PCK} & \text{PC3} & \text{PC2} \end{array}$$

5. Construct the transformation matrix from the top k eigenvectors

$$W = \begin{bmatrix} PC_1 & PC_2 & \cdots & PC_k \end{bmatrix}$$

6. Transform the data

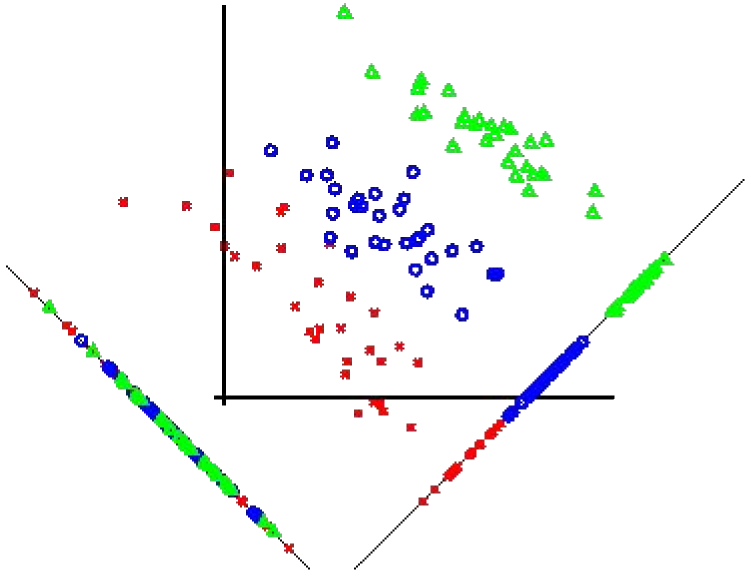
$$\boldsymbol{x} = [x_1, x_2, \ldots, x_d], \quad \boldsymbol{x} \in \mathbb{R}^{\boldsymbol{d}}$$

$$\downarrow \boldsymbol{xW}, \quad \boldsymbol{W} \in \mathbb{R}^{\boldsymbol{d \times k}}$$

$$\mathbf{z} = [z_1, z_2, \ldots, z_k], \quad \mathbf{z} \in \mathbb{R}^{\boldsymbol{k}}$$

PCA Application Example

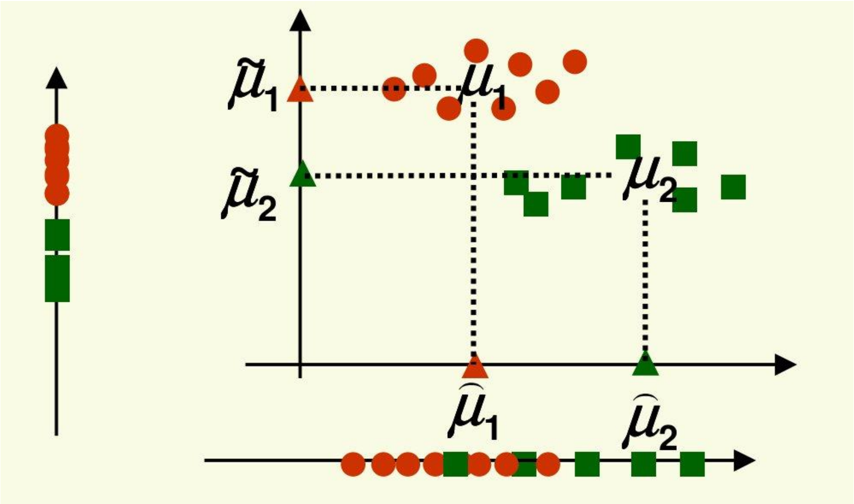
Linear Discriminant Analysis (LDA)



Linear Discriminant Analysis

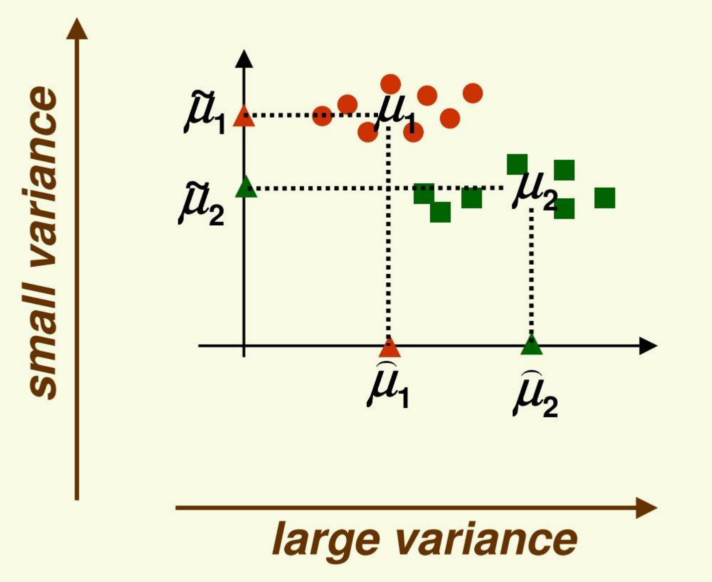
Maximizes class separability

LDA Objectives



1. Maximize **distance between means**

LDA Objectives



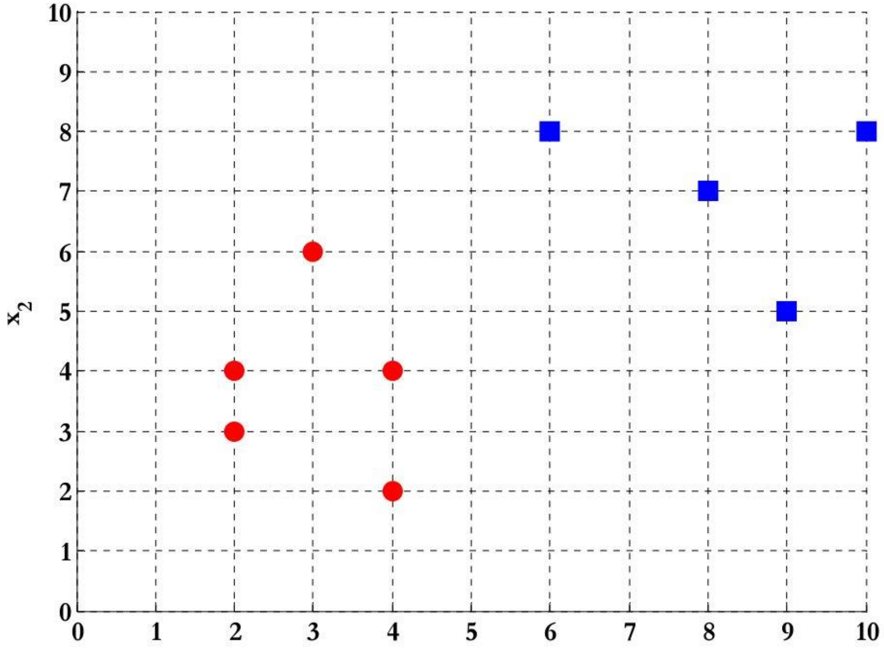
2. Minimize in-class scatter

LDA Objectives

$$\text{Maximize} \quad \left(\frac{\text{distance between means}}{\text{in-class scatter}} \right)$$

Two-classes LDA Example

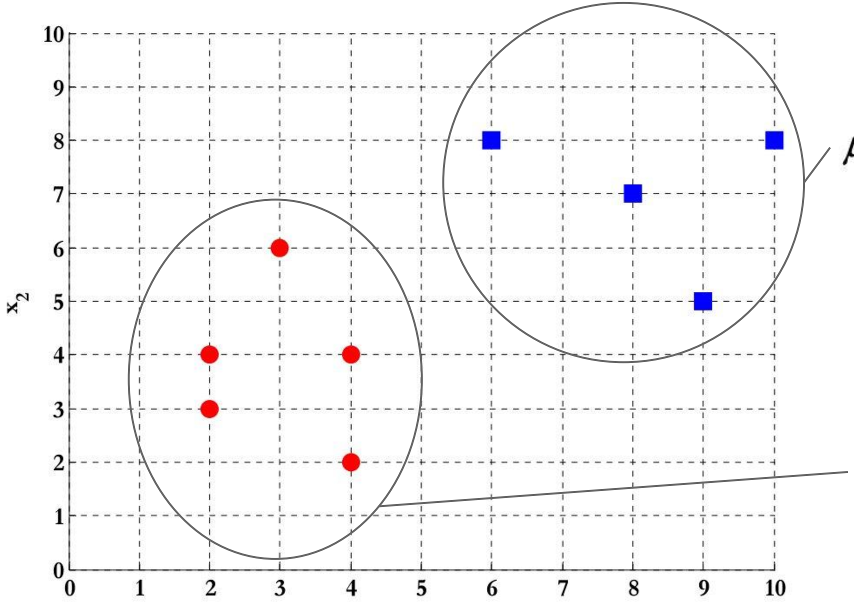
Example Dataset



– 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 $\boldsymbol{\omega_2}$: $\mathbf{X_2}=(x_1,x_2)=\{(9,10),(6,8),(9,5),(8,7),(10,8)\}$

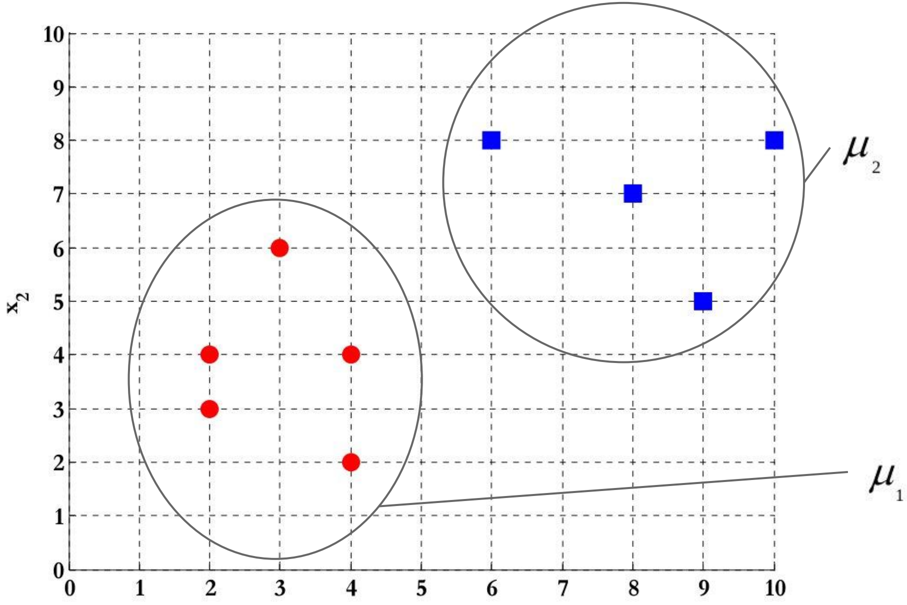
Calculate mean vector of each class



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

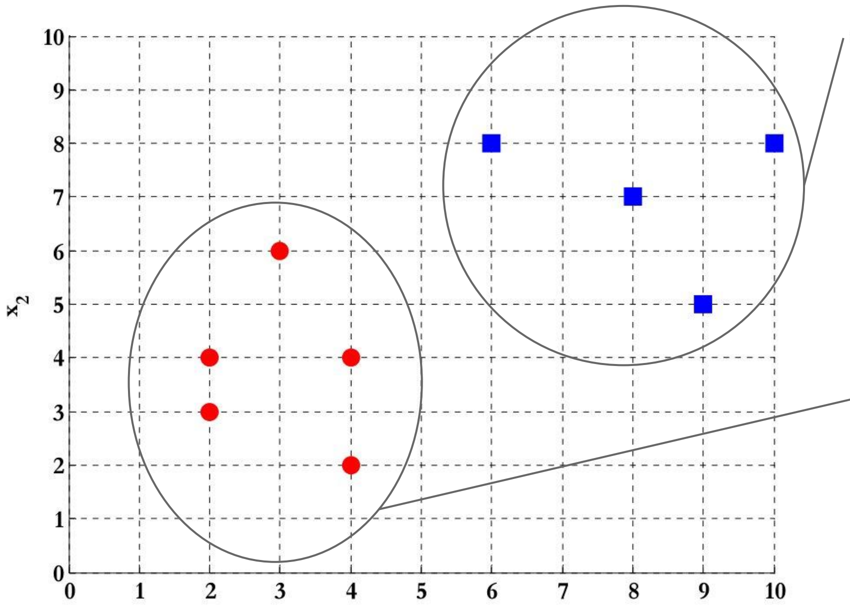
$$\mu_1 = \frac{1}{N_1}\sum_{x\in\omega_1} x = \frac{1}{5}\left[\begin{pmatrix}4\\2\end{pmatrix}+\begin{pmatrix}2\\4\end{pmatrix}+\begin{pmatrix}2\\3\end{pmatrix}+\begin{pmatrix}3\\6\end{pmatrix}+\begin{pmatrix}4\\4\end{pmatrix}\right]=\begin{pmatrix}3\\3.8\end{pmatrix}$$

Calculate between-class matrix



$$S_B = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$$
$$= \left[\begin{pmatrix} 3 \\ 3.8 \end{pmatrix} - \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix}\right]\left[\begin{pmatrix} 3 \\ 3.8 \end{pmatrix} - \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix}\right]^T$$
$$= \begin{pmatrix} -5.4 \\ -3.8 \end{pmatrix}\begin{pmatrix} -5.4 & -3.8 \end{pmatrix}$$
$$= \begin{pmatrix} 29.16 & 20.52 \\ 20.52 & 14.44 \end{pmatrix}$$

Calculate scatter matrix of each class



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

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

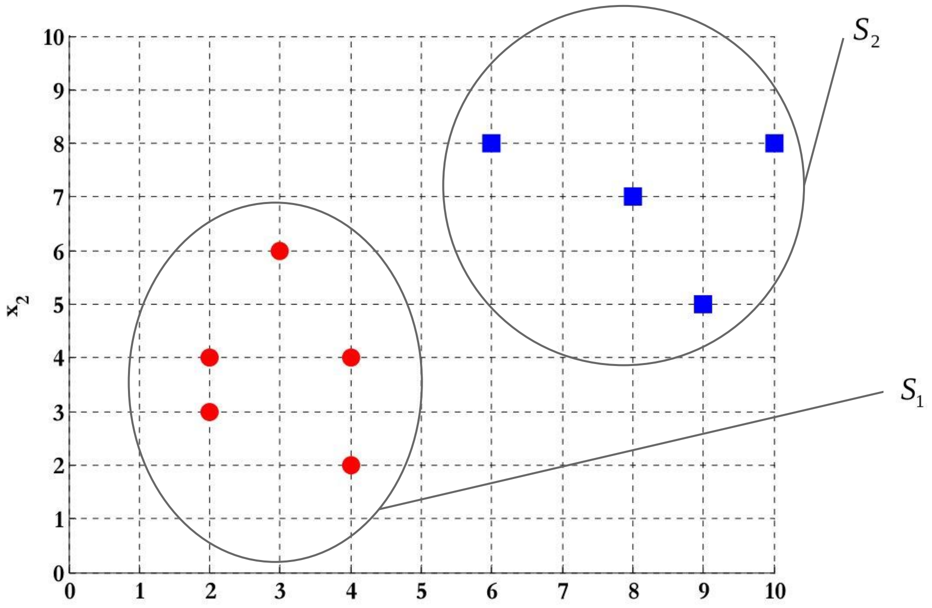
$$= \begin{pmatrix} 2.3 & -0.05 \\ -0.05 & 3.3 \end{pmatrix}$$

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

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

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

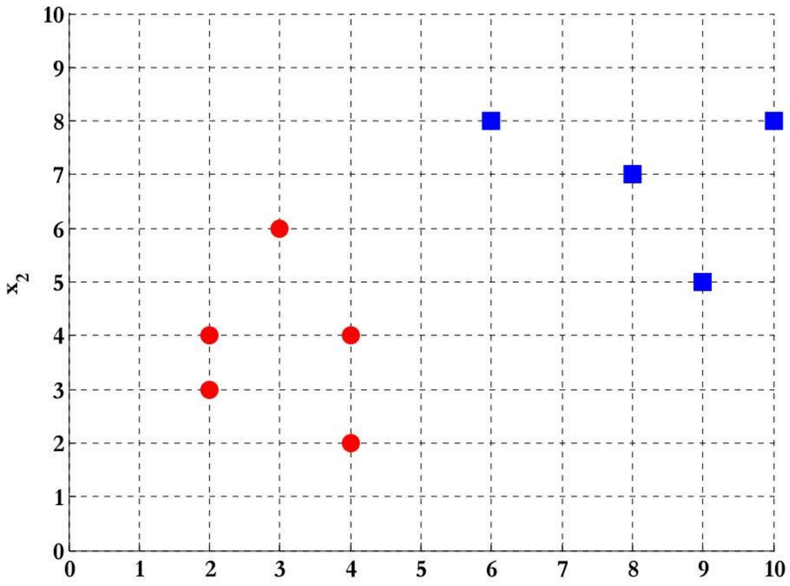
Calculate within-class matrix by summing up scatter matrices



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

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

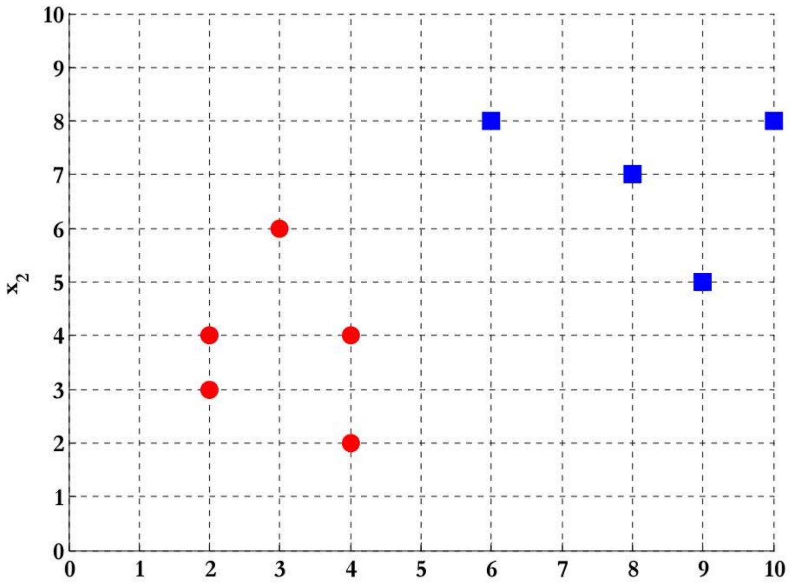
Find eigenvalues & eigenvectors of $S_W^{-1} S_B$



$$\textit{eigenvectors:} \quad \begin{bmatrix} -0.58 \\ 0.82 \end{bmatrix} \quad \begin{bmatrix} 0.91 \\ 0.42 \end{bmatrix}$$

$$\textit{eigenvalues:} \quad 0 \quad 12.2$$

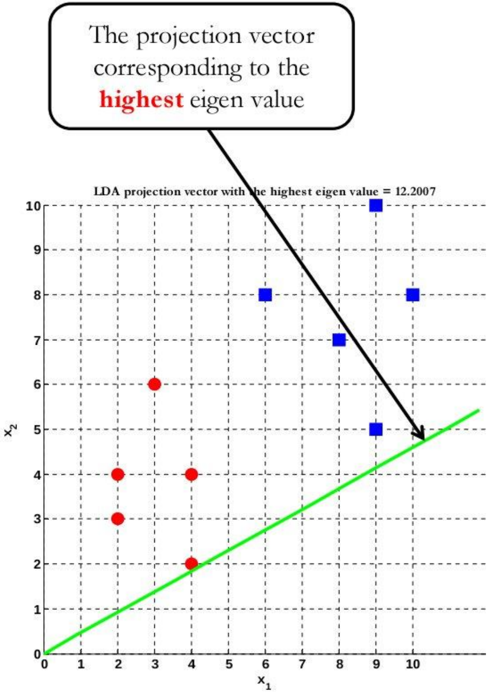
Select the eigenvector with largest eigenvalue



eigenvectors: $\begin{bmatrix} -0.58 \\ 0.82 \end{bmatrix}$ $\boxed{\begin{bmatrix} 0.91 \\ 0.42 \end{bmatrix}}$

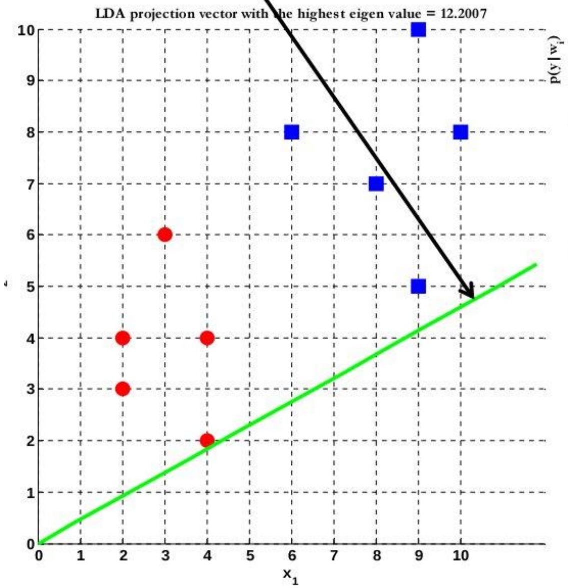
eigenvalues: 0 12.2

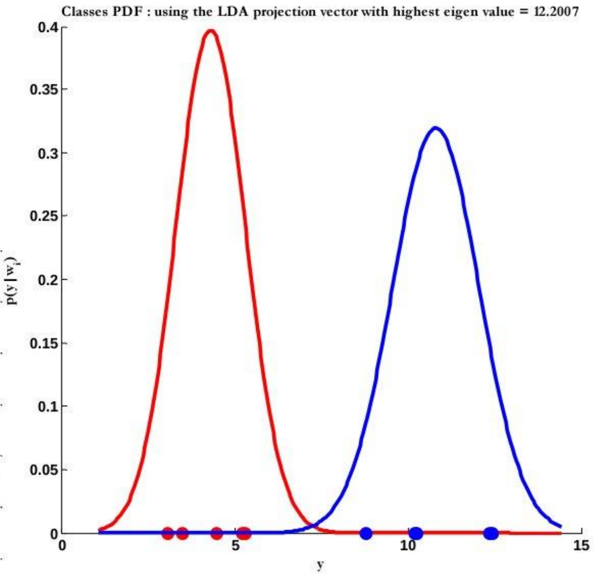
Transform the data



$$x' = \begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} 0.91 \\ 0.42 \end{bmatrix} = 0.91x_1 + 0.42x_2$$

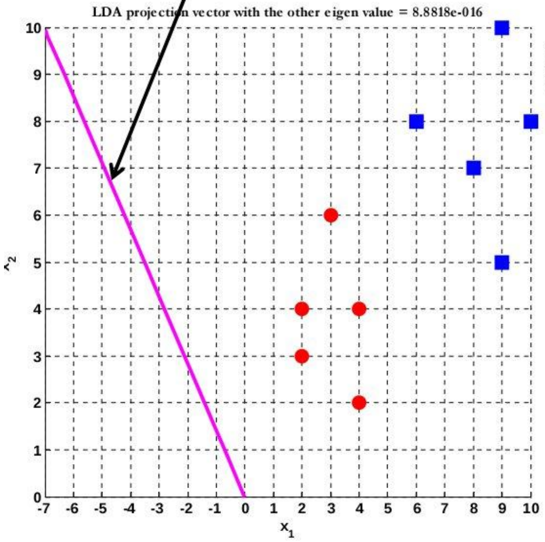
The projection vector corresponding to the **highest** eigen value

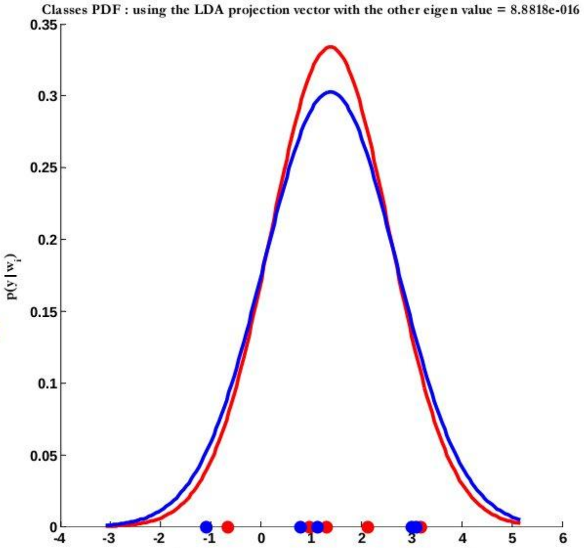




Using this vector leads to **good separability** between the two classes

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





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

Proof:
https://goo.gl/4hUajb

Linear Discriminant Analysis in General

Objective:
For C-classes dataset,
d features -> k features

Target:
find the d $\times$ k transformation matrix

1. Standardize the dataset
2. Calculate mean vector of each class
3. Construct between within-class matrix S_w and between-class matrix S_B
4. Find eigenvalues & eigenvectors of $S_W^{-1} S_B$
5. Select eigenvectors correspond to largest k eigenvalues and construct transformation matrix
6. Transform the data

PCA vs LDA

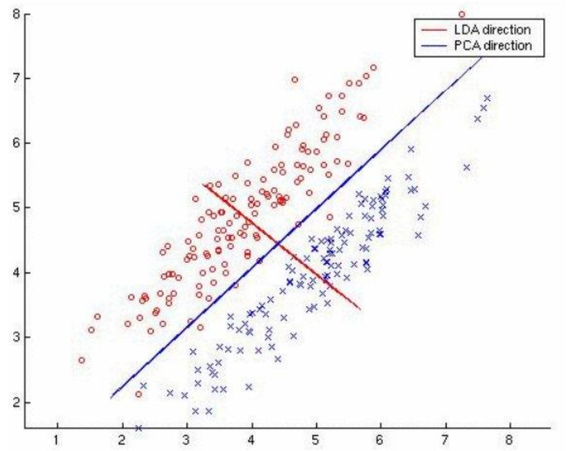
LDA

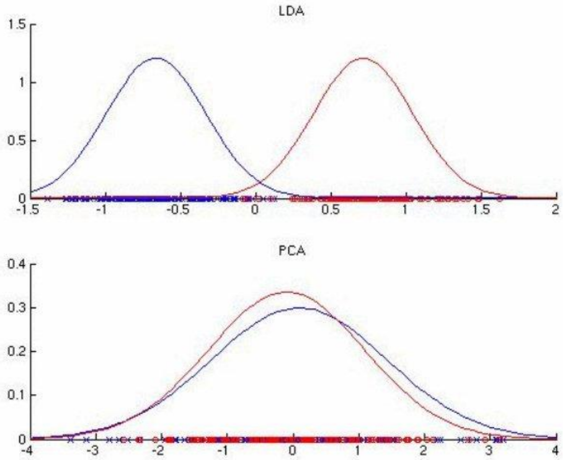
- Only most C-1 features to transformation for C-classes
- May not be good for non-normal distribution data

PCA

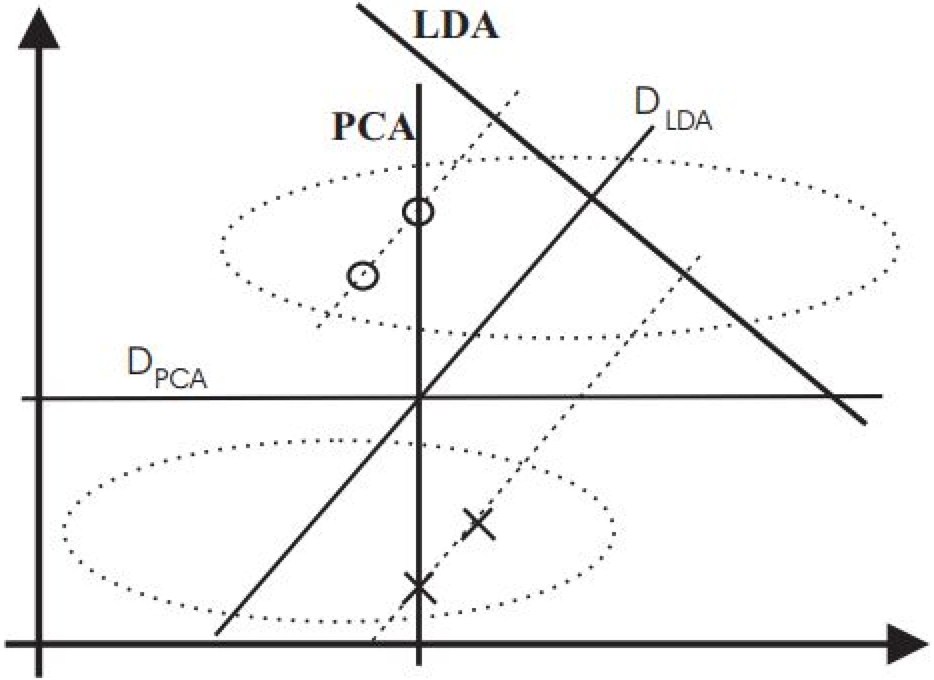
- Not optimal for classification

PCA vs LDA





PCA vs LDA



Paper Study

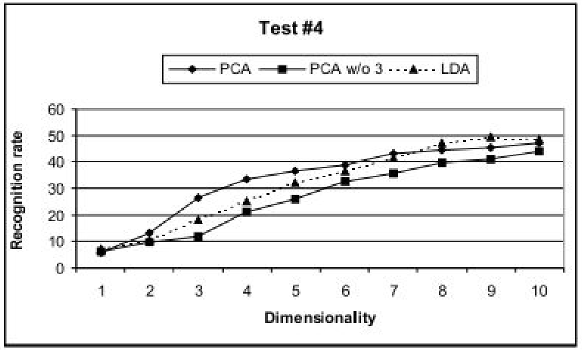
Martínez, A. M., & Kak, A. C. (2001). Pca versus lda. IEEE transactions on pattern analysis and machine intelligence, 23(2), 228-233.

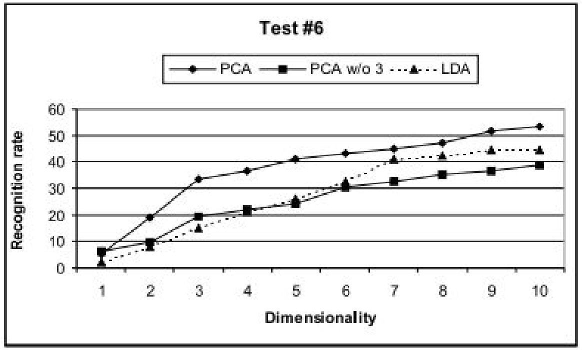
Paper Study

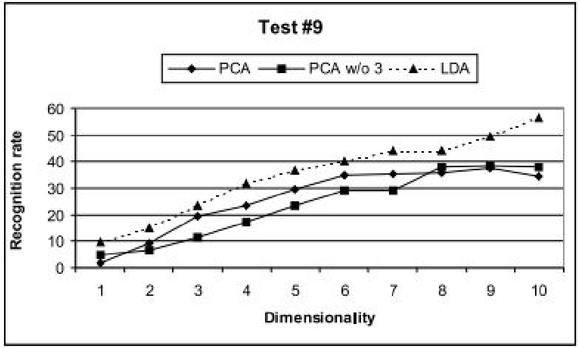
(a) (b) (c) (d) (e) (f) (g)
(h) (i) (j) (k) (l) (m)
(n) (o) (p) (q) (r) (s) (t)
(u) (v) (w) (x) (y) (z)

AR Face Database

Paper Study

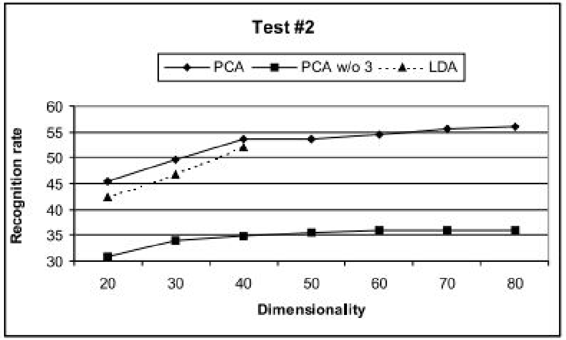


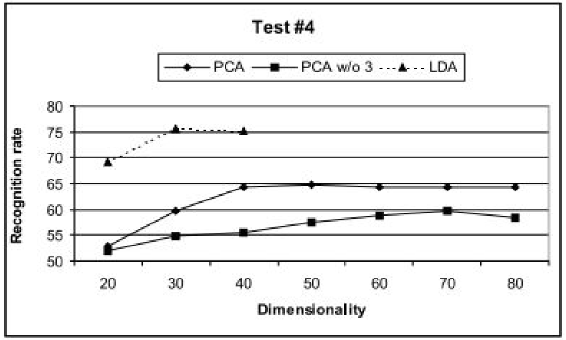


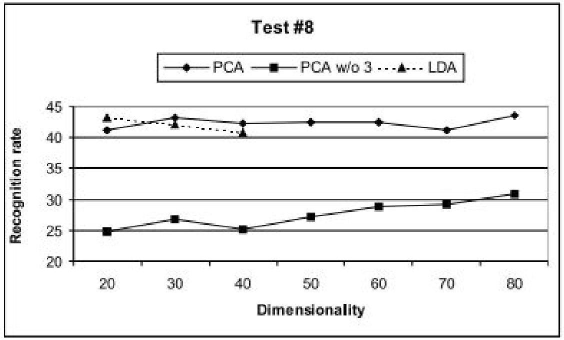


Small Training Data Sets

Paper Study







Small Training Data Sets

Paper Study

$Method$	$f=1$	$f=2$	$f=3$	$f=4$	$f=5$	$f=6$	$f=7$	$f=8$	$f=9$	$f=10$
PCA	6	9	13	9	9	9	7	4	4	3
PCA w/o 3	4	1	0	0	0	0	0	0	0	0
LDA	11	11	8	12	12	12	14	17	17	18

$Method$	$f=20$	$f=30$	$f=40$
PCA	3	2	2
PCA w/o 3	0	0	0
LDA	18	19	19

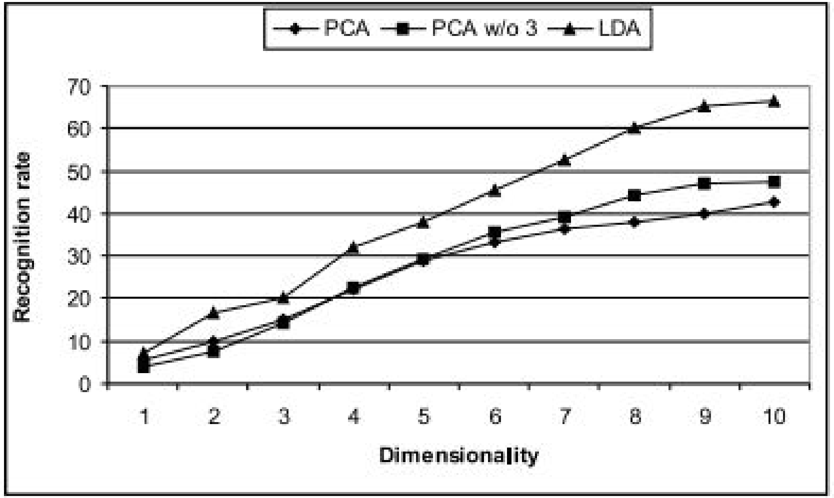
Small Training Data Sets

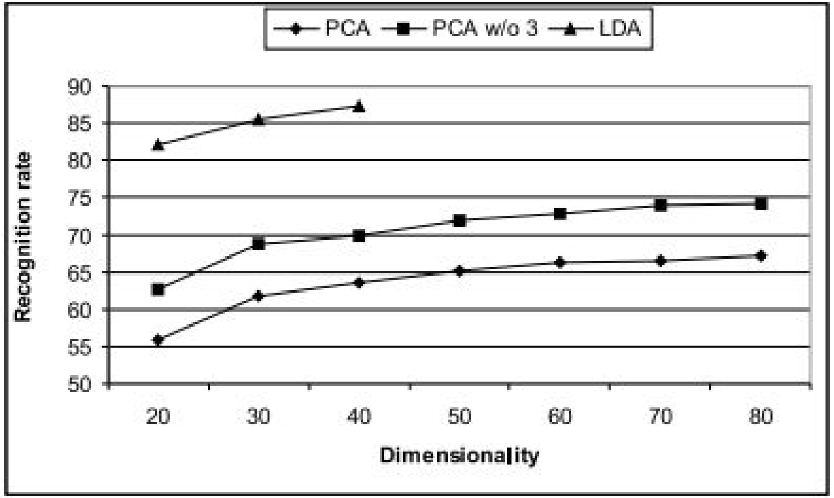
Paper Study

Method	PCA		LDA	
f	10	80	10	40
accuracy	**28%-58%**	**44%-75%**	**31%-68%**	**41%-82%**

Small Training Data Sets

Paper Study





Large Training Data Sets