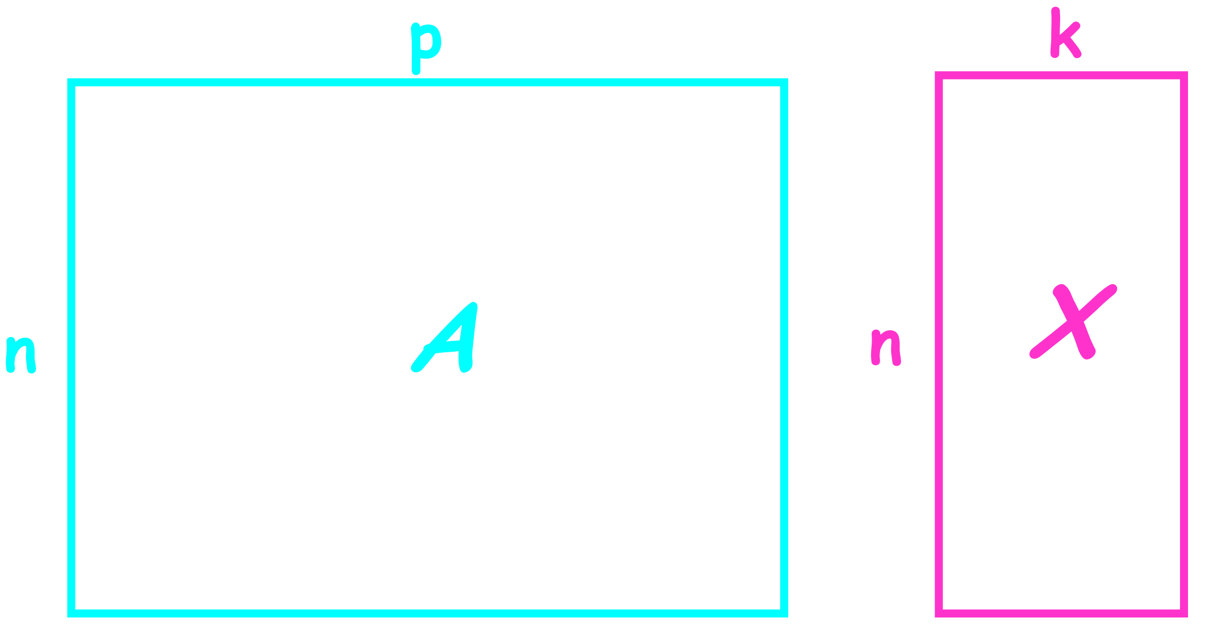
Dimensionality Reduction

Lingqiao Liu

What is dimensionality reduction?



Dimensionality Reduction: why?

- Extract underlying factors

	Disagree		Neutral		Agree
I am the life of the party.	○	○	○	○	○
I feel little concern for others.	○	○	○	○	○
I am always prepared.	○	○	○	○	○
I get stressed out easily.	○	○	○	○	○
I have a rich vocabulary.	○	○	○	○	○
I don't talk a lot.	○	○	○	○	○
I am interested in people.	○	○	○	○	○
I leave my belongings around.	○	○	○	○	○
I am relaxed most of the time.	○	○	○	○	○
I have difficulty understanding abstract ideas.	○	○	○	○	○
I feel comfortable around people.	○	○	○	○	○
I insult people.	○	○	○	○	○
I pay attention to details.	○	○	○	○	○
I worry about things.	○	○	○	○	○
I have a vivid imagination.	○	○	○	○	○

The five factors [edit]

A summary of the factors of the Big Five and their co

- **Openness to experience**: (*inventive/curious* vs intellectual curiosity, creativity and a preference f preference for a variety of activities over a strict r experience.
- **Conscientiousness**: (*efficient/organized* vs. *eas* spontaneous behavior.
- **Extraversion**: (*outgoing/energetic* vs. *solitary/re* talkativeness.
- **Agreeableness**: (*friendly/compassionate* vs. *an* one's trusting and helpful nature, and whether a p
- **Neuroticism**: (*sensitive/nervous* vs. *secure/conf* degree of emotional stability and impulse control

Dimensionality Reduction: why?

- Reduce data noise
  - Face recognition
  - Applied to image de-noising

(a) Noisy image (b) NL means (PSNR=32.90) (c) Local PCA (PSNR=33.70)

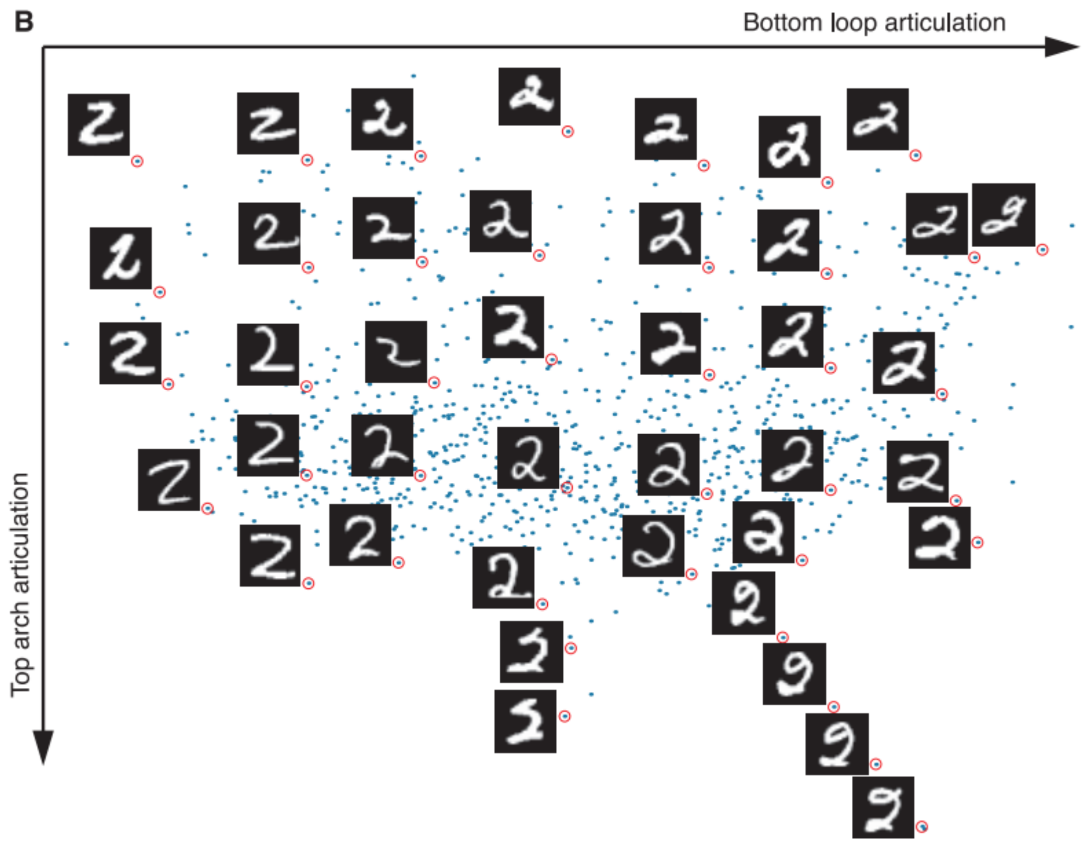
Image courtesy of Charles-Alban Deledalle, Joseph Salmon, Arnak Dalalyan; BMVC 2011
Image denoising with patch-based PCA: local versus global

Dimensionality Reduction: why?

- Reduce the number of model parameters
  - Avoid over-fitting
  - Reduce computational cost

Dimensionality Reduction: why?

- Visualization



Dimensionality Reduction

- General principle:
  - Preserve "useful" information in low dimensional data
- How to define "usefulness"?
  - Many
  - An active research direction in machine learning
- Taxonomy
  - Supervised or Unsupervised
  - Linear or nonlinear
- Commonly used methods:
  - PCA, LDA (linear discriminant analysis), and more.
- Feature Selection vs dimensionality reduction

Outline

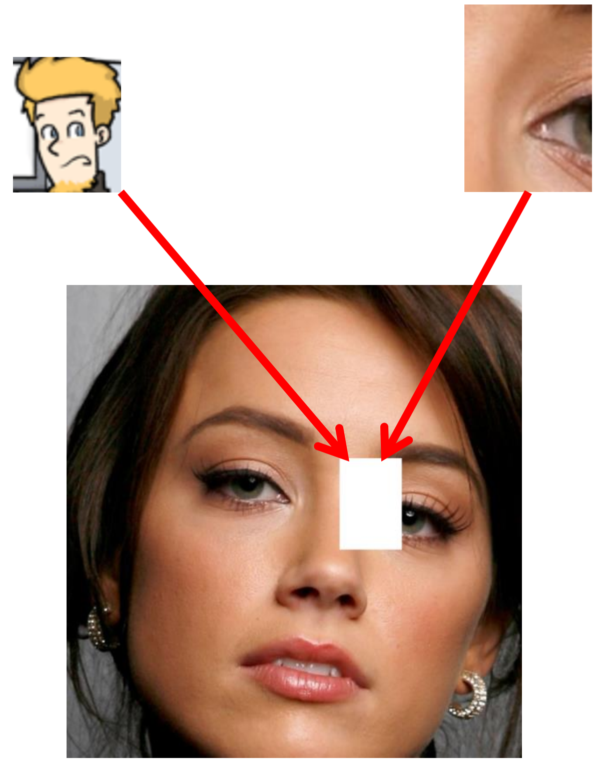
- Principal Component Analysis (PCA):
  - theoretic Part
  - Application: eigenface
- Linear Discriminant Analysis (LDA):
- Case study: character recognition
- Other dimensionality reduction methods

PCA explained: Two perspectives

- Data correlation and information redundancy
- Signal-noise ratio maximization

Data correlation & information redundancy

- I have a rich vocabulary.
- I don't talk a lot.
- I am interested in people.
- I leave my belongings around.
- I am relaxed most of the time.
- I have difficulty understanding abstract ideas.
- I feel comfortable around people.
- I insult people.
- I pay attention to details.
- I worry about things.
- I have a vivid imagination.

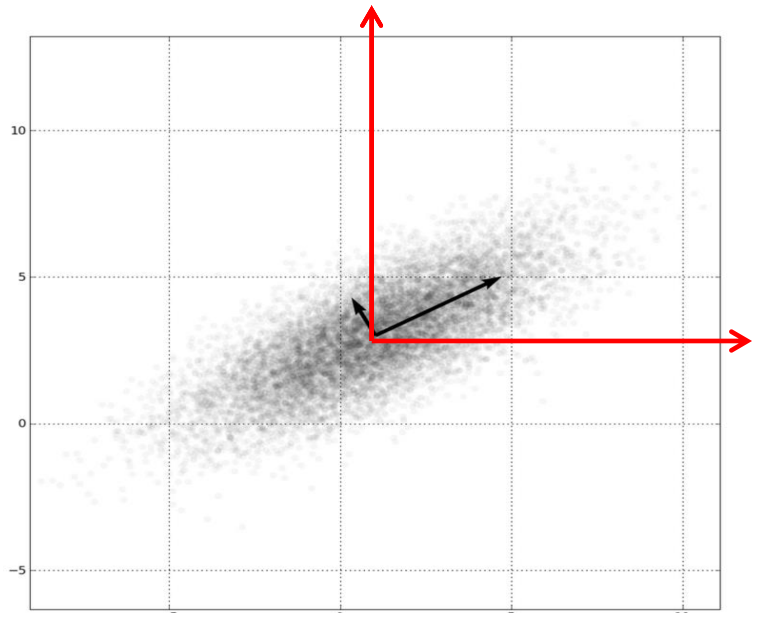


PCA explained: De-correlating data

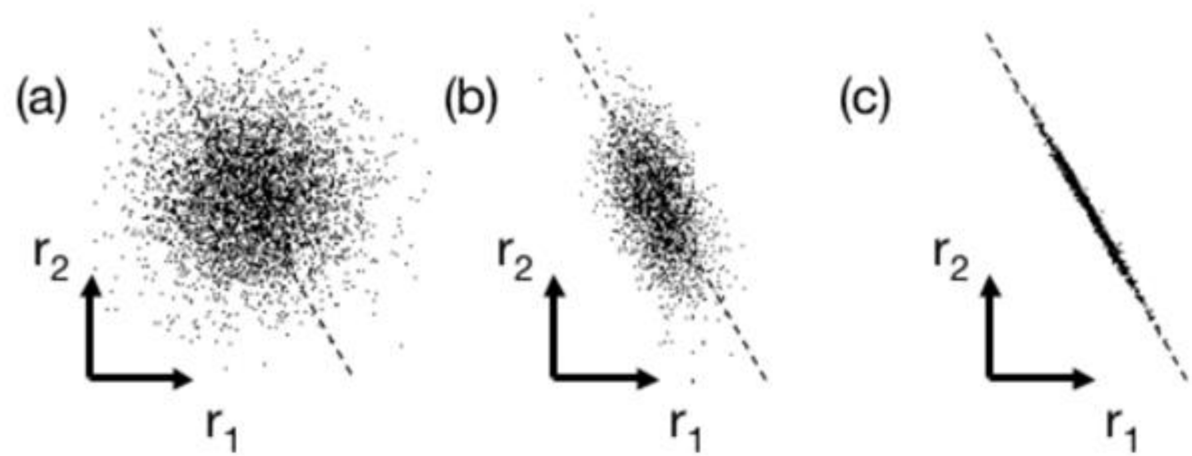
- Dependency vs. Correlation
  - Dependent is a stronger criterion
- Equivalent when data follows Gaussian distribution
- PCA only de-correlates data
  - One limitation of PCA
  - ICA, but it is more complicate

PCA explained: De-correlating data

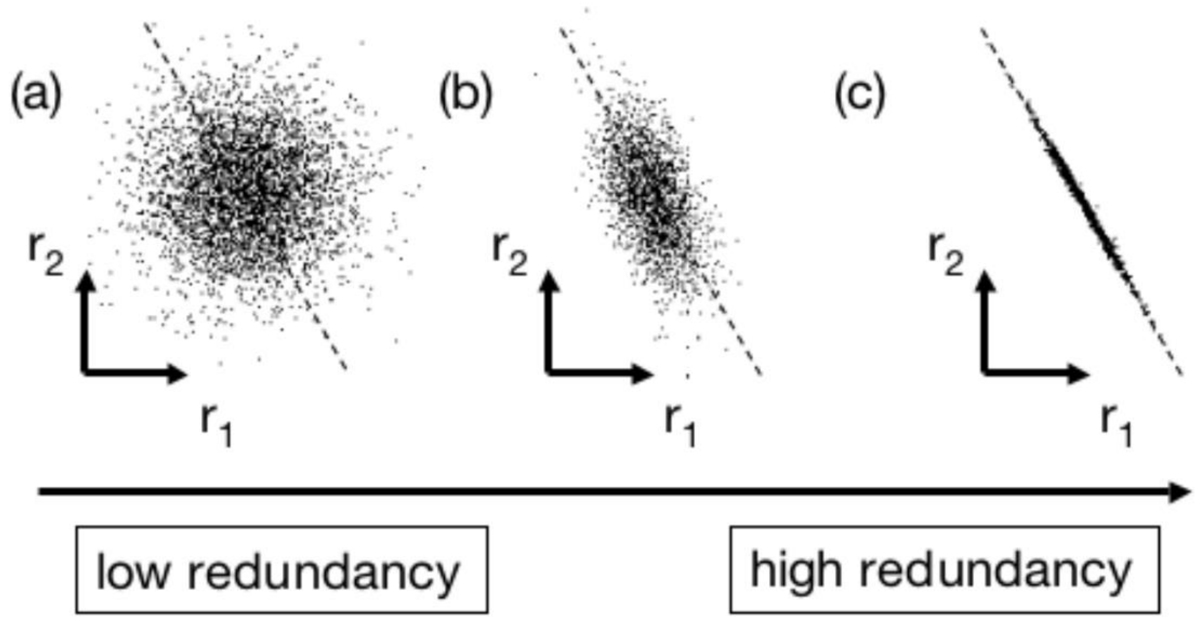
- Geometric interpretation of correlation



PCA explained: De-correlating data

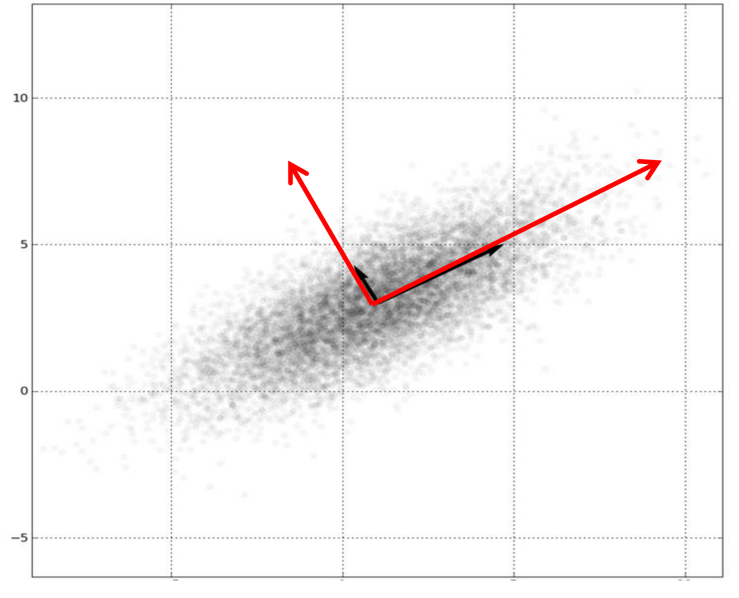


PCA explained: De-correlating data

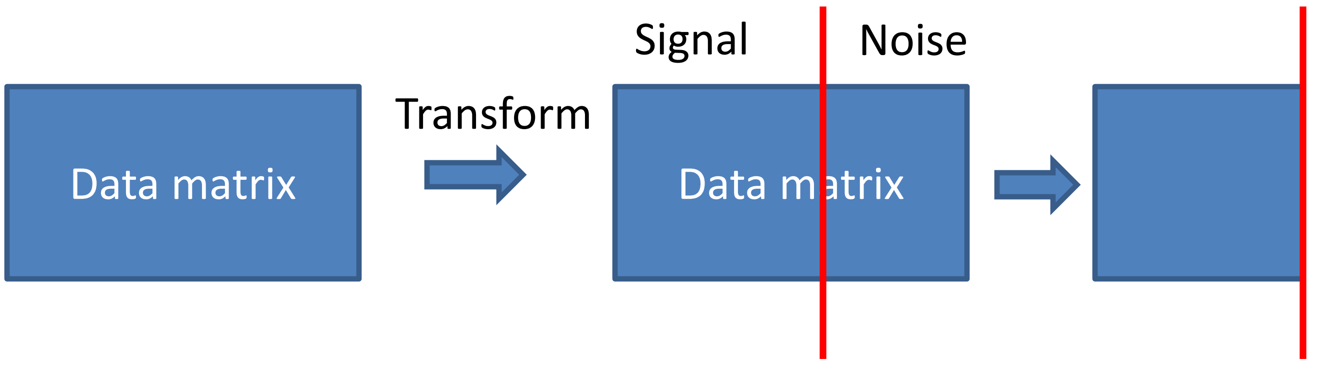


PCA explained: De-correlating data

- Correlation can be removed by rotating the data point or coordinate



PCA explained: SNR maximization

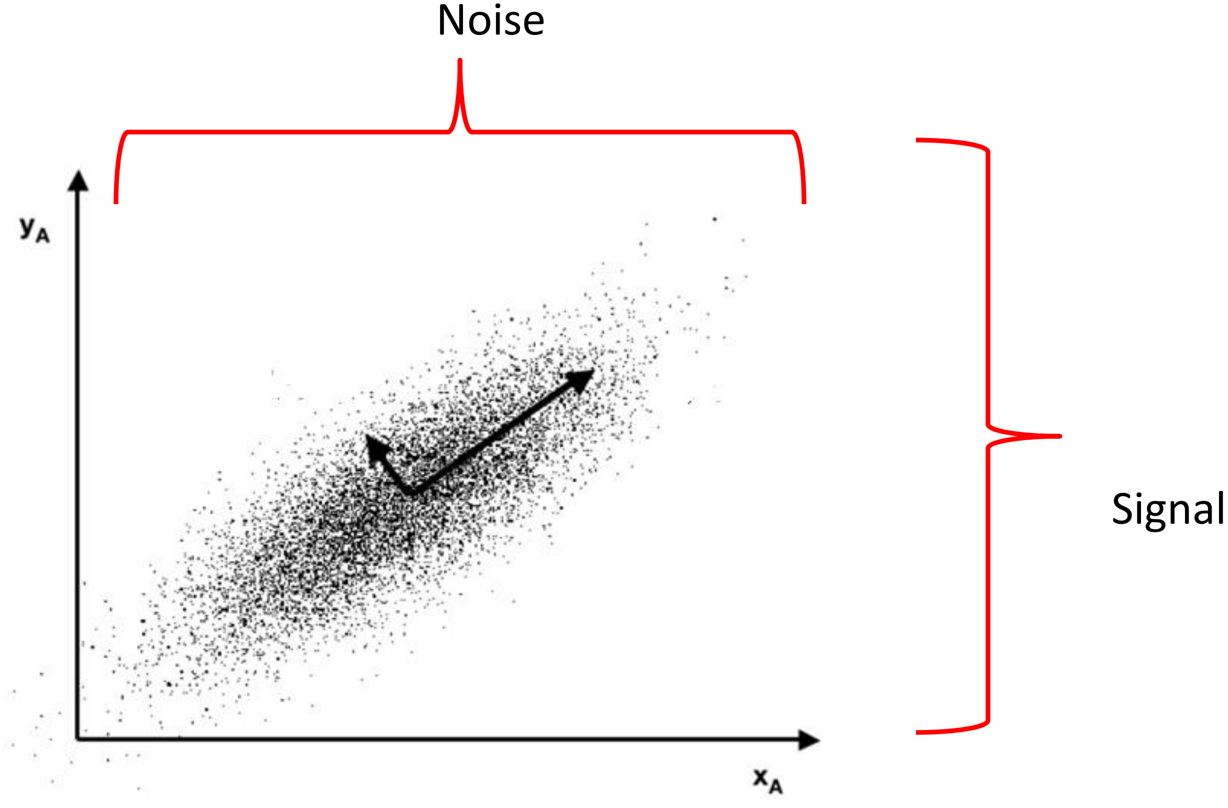


- Maximize

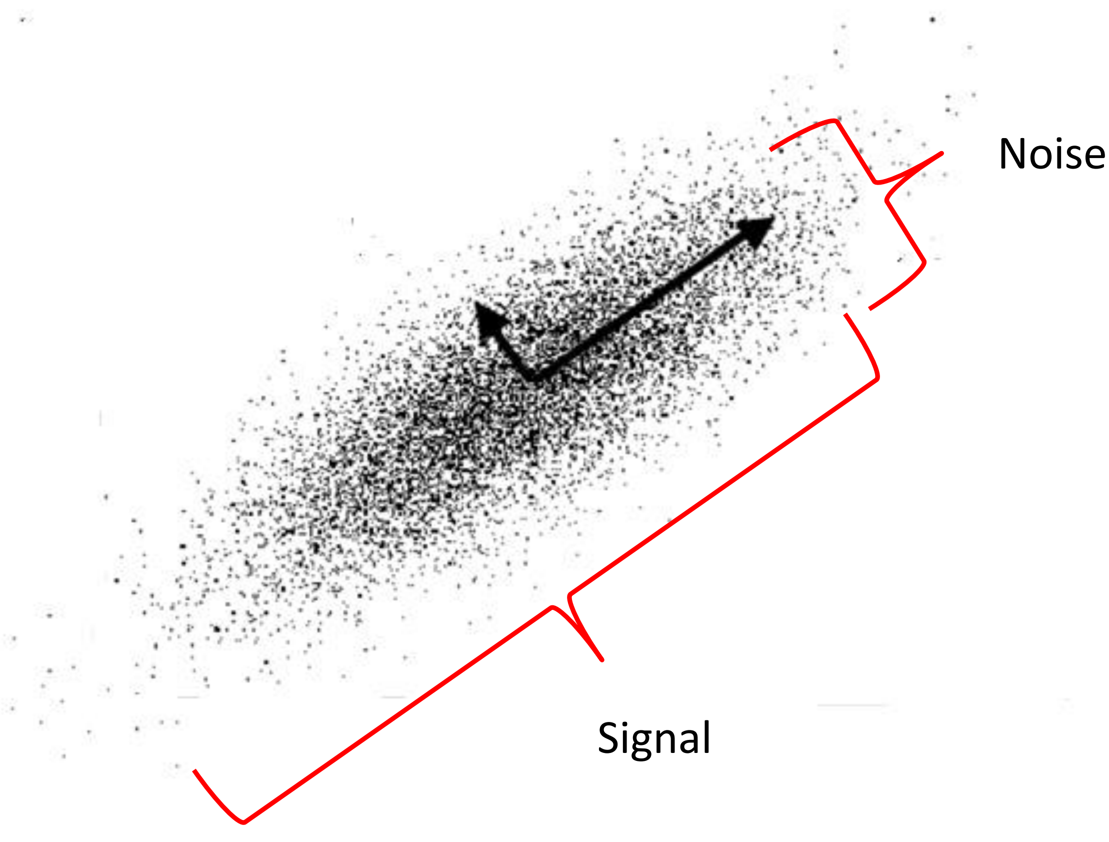
$$SNR = \frac{\sigma^2_{signal}}{\sigma^2_{noise}}.$$

PCA explained: SNR maximization

- Keep one signal dimension, discard one noisy dimension



PCA explained: SNR maximization



PCA explained

- Target
  - 1: Find a new coordinate system which makes different dimensions zero correlated
  - 2: Find a new coordinate system which aligns (top-k) largest variance
- Method
  - Rotate the data point or coordinate
- Mathematically speaking…
  - How to rotate?
  - How to express our criterion

Mathematatic Basics

- Mean, Variance, Covariance
- Matrix norm, trace,
- Orthogonal matrix, basis
- Eigen decomposition

Mathematematic Basics

- (Sample) Mean

$$\bar{X} = \frac{\sum_{i=1}^{n} X_i}{n}$$

- (Sample) Variance

$$s^2 = \frac{\sum_{i=1}^{n}(X_i - \bar{X})^2}{(n-1)}$$

- (Sample) Covariance

$$cov(X,Y) = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{(n-1)}$$

Mathematic Basics

- Covariance Matrix

$$C = \begin{pmatrix} cov(x,x) & cov(x,y) & cov(x,z) \\ cov(y,x) & cov(y,y) & cov(y,z) \\ cov(z,x) & cov(z,y) & cov(z,z) \end{pmatrix}$$

$$\mathbf{C} = \frac{1}{n-1}\sum_{i=1}^{n}(\mathbf{X}_i - \bar{\mathbf{X}})(\mathbf{X}_i - \bar{\mathbf{X}})^T$$

Mathematic Basics

- Frobenius norm

$$\|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2}$$

- Trace

$$\mathrm{tr}(A) = a_{11} + a_{22} + \cdots + a_{nn} = \sum_{i=1}^{n} a_{ii}$$

$$\mathrm{tr}(X^{\mathrm{T}}Y) = \mathrm{tr}(XY^{\mathrm{T}}) = \mathrm{tr}(Y^{\mathrm{T}}X) = \mathrm{tr}(YX^{\mathrm{T}})$$

$$\|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2} = \sqrt{\mathrm{trace}(A^*A)}$$

Mathematatic Basics

- Symmetric Matrix $\mathbf{A} = \mathbf{A}^T$
- Covariance matrix is symmetric

$$\mathbf{C} = \frac{1}{n-1}\mathbf{X}\mathbf{X}^T$$

$$\mathbf{C} = \mathbf{C}^T$$

Mathematatic Basics

- Orthogonal matrix

$$Q^{\mathrm{T}}Q = QQ^{\mathrm{T}} = I,$$

- Rotation effect

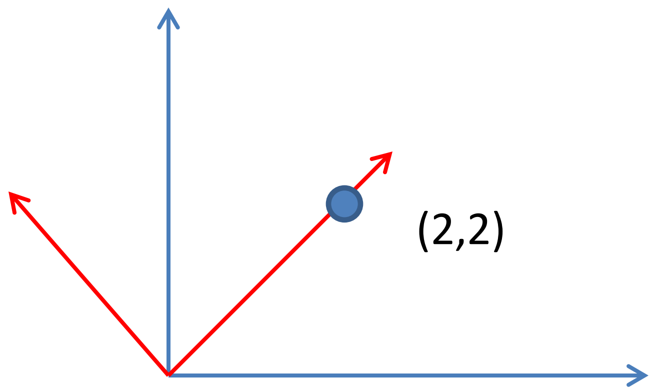
$$\|\mathbf{Qx}\|_F = \sqrt{\mathrm{trace}(\mathbf{x}^T\mathbf{Q}^T\mathbf{Qx})} = \sqrt{\mathrm{trace}(\mathbf{x}^T\mathbf{x})} = \|\mathbf{x}\|_F$$

$$\mathbf{x} = \mathbf{Q}^T\mathbf{Qx}$$

Mathematic Basics

- Relationship to coordinate system
  - A point = linear combination of bases
  - Combination weight = coordinate
- Each row (column) of Q = basis
  - Not unique
  - Relation to coordinate rotation
- New coordinate $\mathbf{Q}\mathbf{x}$

Mathematatic Basics



$$\begin{pmatrix}2\\2\end{pmatrix} = 2\begin{pmatrix}1\\0\end{pmatrix} + 2\begin{pmatrix}0\\1\end{pmatrix}$$

$$\begin{pmatrix}2\\2\end{pmatrix} = 2\sqrt{2}\begin{pmatrix}\frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}}\end{pmatrix} + 0\begin{pmatrix}\frac{1}{\sqrt{2}}\\ -\frac{1}{\sqrt{2}}\end{pmatrix}$$

New coordinate

$$\begin{pmatrix}2\sqrt{2}\\0\end{pmatrix} = \begin{pmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}}\end{pmatrix}\begin{pmatrix}2\\2\end{pmatrix}$$

Old coordinate

Mathematic Basics

- Rank of a matrix
  - Useful property

$$\begin{aligned}&\text{if} \quad \mathbf{A} = \mathbf{BC}\\&\text{rank}(\mathbf{A}) \leq \min\{\text{rank}(\mathbf{B}), \text{rank}(\mathbf{C})\}\end{aligned}$$

- Eigenvalue and Eigenvector

$$\mathbf{Au} = \lambda \mathbf{u}$$

- Properties
  - Multiple solutions
  - Scaling invariant
  - Relation to the rank of A

Mathematic Basics

Eigen-decomposition

- If $\mathbf{A}$ is symmetric

$$\mathbf{A} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T \quad Q^{\mathrm{T}}Q = QQ^{\mathrm{T}} = I,$$

PCA: solution

- Target 1: de-correlation

$$\mathbf{C}_X = \frac{1}{n-1}\mathbf{X}\mathbf{X}^T$$

$$\mathbf{Y} = \mathbf{P}\mathbf{X}$$

$$\begin{aligned}\mathbf{C}_Y &= \frac{1}{n-1}\mathbf{P}\mathbf{X}(\mathbf{P}\mathbf{X})^T \\ &= \frac{1}{n-1}\mathbf{P}\mathbf{X}\mathbf{X}^T\mathbf{P}^T \\ &= \frac{1}{n-1}\mathbf{P}\mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^{\mathbf{T}}\mathbf{P}^{\mathbf{T}}\end{aligned}$$

PCA: solution

$$\begin{aligned}\mathbf{C}_Y &= \frac{1}{n-1}\mathbf{PX}(\mathbf{PX})^T \\ &= \frac{1}{n-1}\mathbf{PXX}^T\mathbf{P}^T \\ &= \frac{1}{n-1}\mathbf{PQ\Lambda Q^T P^T}\end{aligned}$$

$$\text{if} \quad \mathbf{P} = \mathbf{Q}^T$$

$$\mathbf{C}_Y = \frac{1}{n-1}\mathbf{\Lambda}$$

PCA: solution

- Variance of each dimension

$$\begin{aligned}\mathrm{Var}(y_k) &= \frac{1}{n-1}\mathbf{P}_k\mathbf{X}\mathbf{X}^T\mathbf{P}_k^T \\ &= \frac{1}{n-1}\mathbf{p}_k\mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^{\mathbf{T}}\mathbf{p}_k^T \\ &= \frac{1}{n-1}\lambda_k\end{aligned}$$

- Rank dimensions according to their corresponding eigenvalues

PCA: algorithm

- 1. Subtract mean
- 2. Calculate the covariance matrix
- 3. Calculate eigenvectors and eigenvalues of the covariance matrix
- 4. Rank eigenvectors by its corresponding eigenvalues
- 4. Obtain P with its column vectors corresponding to the top k eigenvectors

PCA: MATLAB code

```

Mu = mean(fea);
fea = fea - repmat(Mu,[size(fea,1),1]);
Cov = fea'*fea;
[V,D] = eig(Cov);
[value,rank_idx] = sort(diag(D),'descend');
P = V(:,rank_idx(1:10));

```

PCA: reconstruction

- Reconstruct x

$$\hat{\mathbf{x}} = \hat{\mathbf{P}}^T \hat{\mathbf{P}} \mathbf{x}$$

- Derive PCA through minimizing the reconstruction error

$$\min_{\mathbf{P}} \ \|\mathbf{X} - \mathbf{P}^T \mathbf{P} \mathbf{X}\|_F^2$$
$$s.t. \ \mathbf{P}\mathbf{P}^T = \mathbf{I}$$

PCA: reconstruction

- Reighley Quotient

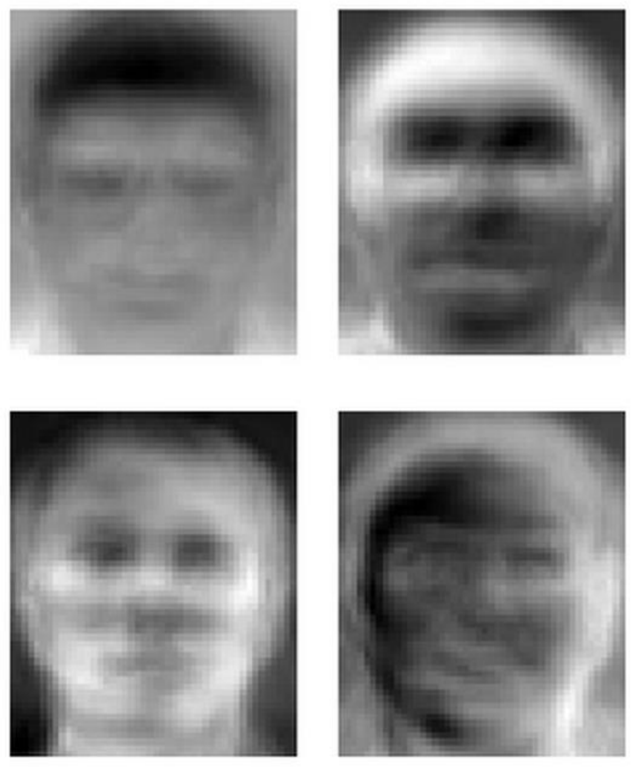
$$\max_{\mathbf{P}} \ \mathrm{trace}(\mathbf{P}\mathbf{X}\mathbf{X}^T\mathbf{P}^T)$$
$$s.t. \ \mathbf{P}\mathbf{P}^T = \mathbf{I}$$

- Solution = PCA

Application: Eigen-face method

- Sirovich and Kirby (1987) showed that PCA could be used on a collection of face images to form a set of basis features.
- Not only limited to face recognition
- Steps
  - Image as high-dimensional feature
  - PCA

Application: Eigen-face method



Some eigenfaces from AT&T Laboratories Cambridge

Application: Reconstruction

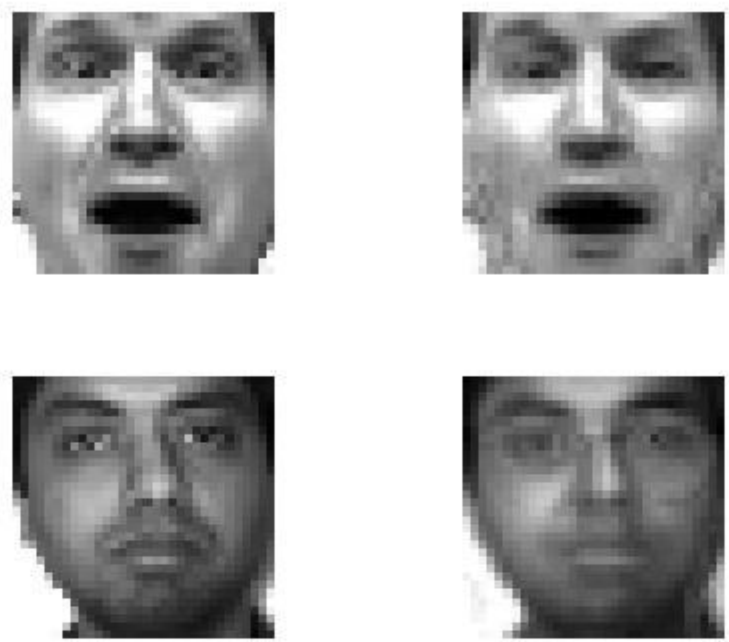
Reconstructed from top-2 eigenvectors

Application: Reconstruction

Reconstructed from top-15 eigenvectors

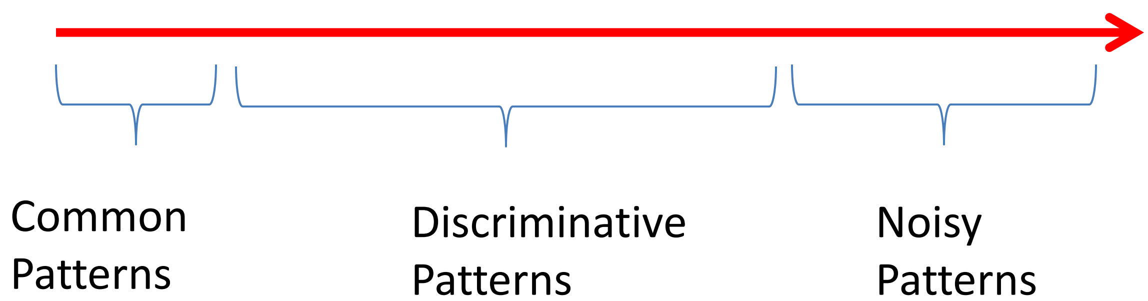
Application: Reconstruction

Reconstructed from top-40 eigenvectors



Application: Eigen-face method

- From large to small eigenvalues



high dimensionality issue

- For high-dimensional data $\mathbf{C}_X = \frac{1}{n-1}\mathbf{X}\mathbf{X}^T$ can be too large
- The number of samples is relatively small

$$d \gg N$$

$$\mathbf{X}^T\mathbf{X}\mathbf{v} = \lambda\mathbf{v}$$

$$\mathbf{X}\mathbf{X}^T(\mathbf{X}\mathbf{v}) = \lambda(\mathbf{X}\mathbf{v})$$

Define $\mathbf{u} = \mathbf{X}\mathbf{v}$

$$\mathbf{X}\mathbf{X}^T\mathbf{u} = \lambda\mathbf{u}$$

High dimensionality issue

- 1. Centralize data
- 2. Calculate the kernel matrix
- 3. Perform Eigen-decomposition on the kernel matrix and obtain its eigenvector $\mathbf{v}$
- 4. Obtain the Eigenvector of the covariance matrix by $\mathbf{u} = \mathbf{X}\mathbf{v}$
- Question? How many eigenvectors you can obtain in this way?

Discriminative dimensionality reduction

- General principle:
  - Preserve “useful” information in low dimensional data
  - PCA: measure ``usefulness’’ through reconstruction error or covariance structure.
  - Useful for reconstruction $\neq$ useful for classification
- General principle for discriminative dimensionality reduction
  - Preserve “discriminative” information in low dimensional data

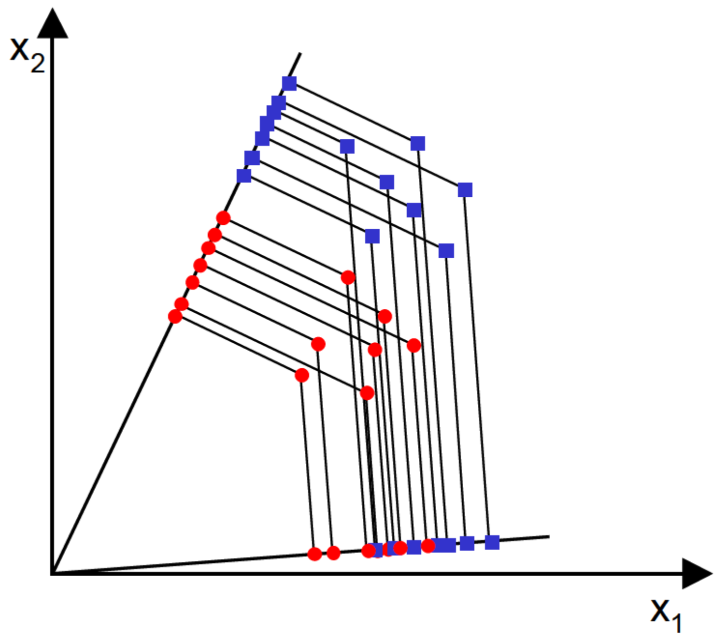
Linear Discriminant Analysis

- Linear Discriminant Analysis (LDA)
  - Discriminative dimensionality reduction
  - Linear dimensionality reduction $\mathbf{PX}$

- Supervised information
  - Class label
  - Data from the same class => Become close
  - Data from different classes => far from each other

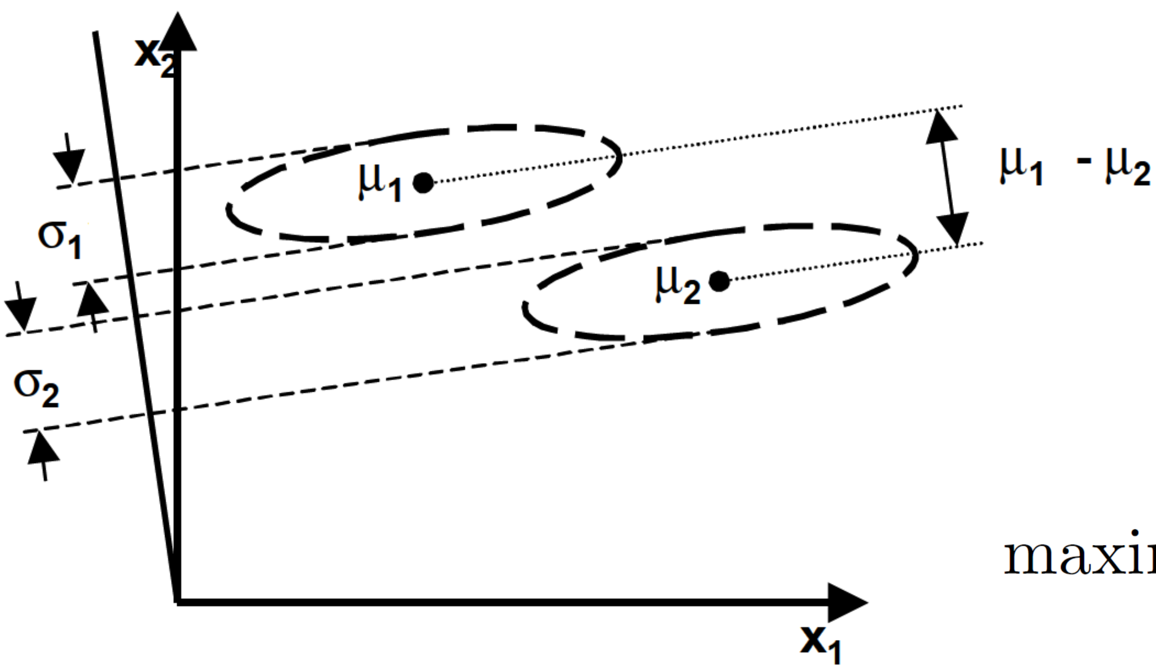
Linear Discriminant Analysis (LDA), Objective

- Two classes:



Linear Discriminant Analysis, Objective

- Two classes:



$$\text{maximize} \quad \frac{(\hat{\mu}_1 - \hat{\mu}_2)^2}{\hat{\sigma}_1^2 + \hat{\sigma}_2^2}$$

Linear Discriminant Analysis (LDA), Objective

- Mean after projection:

$$\hat{\mu} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{p}^T\mathbf{x}_i = \mathbf{p}^T\frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i = \mathbf{p}^T\mu$$

- Variance after projection:

$$\begin{aligned}
\hat{\sigma}^{\mathbf{2}} &= \frac{1}{N}\sum_{i=1}^{N}(\mathbf{p}^T\mathbf{x}_i - \mathbf{p}^T\mu)^2 \\
&= \frac{1}{N}\sum_{i=1}^{N}\mathbf{p}^T(\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T\mathbf{p} \\
&= \mathbf{p}^T\left(\frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T\right)\mathbf{p}
\end{aligned}$$

Linear Discriminant Analysis (LDA), Objective

- Mean distance

$$\left(\mathbf{p}^T\mu_1 - \mathbf{p}^T\mu_2\right)^2 = \mathbf{p}^T\left(\mu_1 - \mu_2\right)\left(\mu_1 - \mu_2\right)^T\mathbf{p}$$

- Between class Scatterness, within class Scatterness

$$\mathbf{S}_b = \left(\mu_1 - \mu_2\right)\left(\mu_1 - \mu_2\right)^T$$

$$\mathbf{S}_w = \sum_{j=1,2}\frac{1}{N_j}\sum_{i=1}^{N_j}(\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T$$

Linear Discriminant Analysis (LDA), Solution

- Objective

$$\text{maximize} \quad \frac{\mathbf{p}^T \mathbf{S}_b \mathbf{p}}{\mathbf{p}^T \mathbf{S}_w \mathbf{p}}$$

- Solution

$$\text{maximize} \quad \frac{\mathbf{p}^T \mathbf{S}_b \mathbf{p}}{\mathbf{p}^T \mathbf{S}_w \mathbf{p}} \quad \Longrightarrow \quad \begin{aligned} &\text{maximize} \quad \mathbf{p}^T \mathbf{S}_b \mathbf{p} \\ &\qquad s.t. \ \mathbf{p}^T \mathbf{S}_w \mathbf{p} = 1 \end{aligned}$$

$$L = \mathbf{p}^T \mathbf{S}_b \mathbf{p} - \lambda(\mathbf{p}^T \mathbf{S}_w \mathbf{p} - 1)$$

$$\frac{\partial L}{\partial \mathbf{p}} = 0 \quad \Rightarrow \quad \mathbf{S}_w^{-1} \mathbf{S}_b \mathbf{p} = \lambda \mathbf{p}$$

Linear Discriminant Analysis (LDA), Solution

- Eigenvectors corresponding to the largest eigenvalues of $\mathbf{S}_w^{-1}\mathbf{S}_b$
  - Why?
- Implementation details
  - What if $\mathbf{S}_w$ is not invertible?

Use $(\mathbf{S}_w + \lambda\mathbf{I})^{-1}$ instead

Linear Discriminant Analysis, Multi-class

- Generalized to multiple classes

$$\mathbf{S}_b = \sum_{i=1,j=1}^{C} (\mu_i - \mu_j)(\mu_i - \mu_j)^T = \sum_i (\mu_i - \mu)(\mu_i - \mu)^T$$

$$\mathbf{S}_w = \sum_{j=1}^{C} \sum_{i \in \mathcal{C}_j} (\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T$$

$$\text{maximize} \quad \frac{\operatorname{trace}\left(\mathbf{P}^T \mathbf{S}_b \mathbf{P}\right)}{\operatorname{trace}\left(\mathbf{P}^T \mathbf{S}_w \mathbf{P}\right)}$$

- Solution:
  - Top c eigenvectors of $\mathbf{S}_w^{-1}\mathbf{S}_b$
  - Discussion: how many projections you can have?

Case study

- Case study: character recognition

Other dimensionality reduction methods

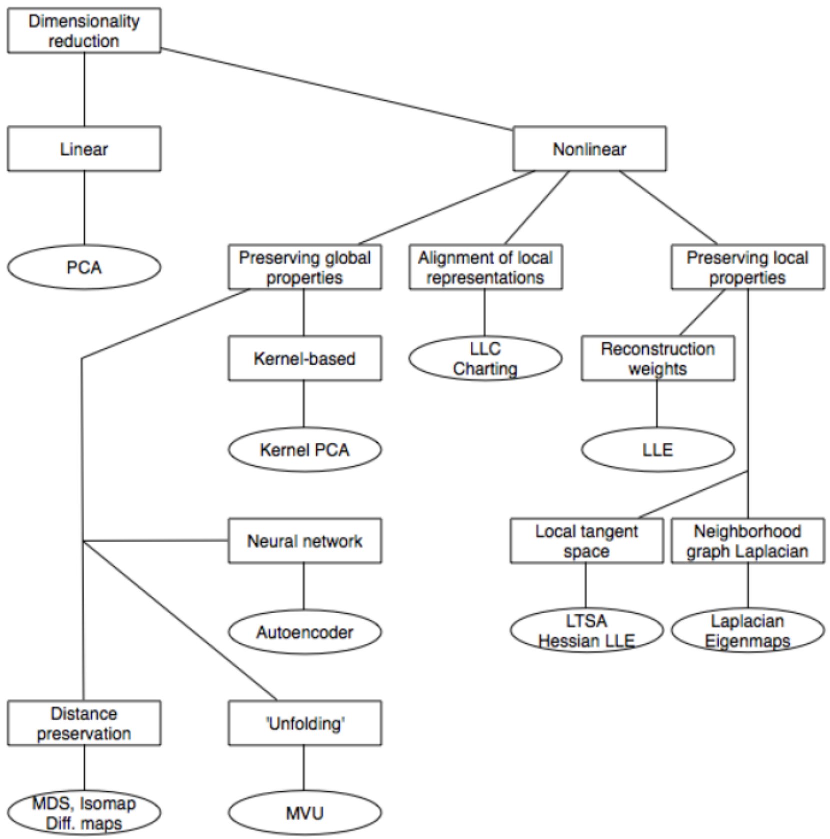


Fig. 1. Taxonomy of dimensionality reduction techniques.

Image courtesy of L.J.P. van der Maaten , E. O. Postma , H. J. van den Herik; **Dimensionality Reduction: A Comparative Review** 2008

Other dimensionality reduction methods

- Some popular idea:
  - Kernel trick, e.g. kernel PCA
  - Preserve localized information
    - Local similarity preserving, e.g. LLE
    - Local discriminative dimensionality reduction, e.g. LFDA
  - Nonlinear reconstruction through neural network
    - Auto-encoder

- How to develop a new dimensionality reduction method?
  - Define your own objective
  - Define your reduction function
  - Optimization