Kernel-based Methods for Unsupervised Learning

LEAR project-team, INRIA

Zaid Harchaoui

Grenoble, July 30th 2010

Outline

1. Introduction
2. Kernel methods and feature space
3. Mean element and covariance operator
4. Kernel PCA
5. Kernel CCA
6. Spectral Clustering

Outline

1 Introduction

2 Kernel methods and feature space

3 Mean element and covariance operator

4 Kernel PCA

5 Kernel CCA

6 Spectral Clustering

Unsupervised learning

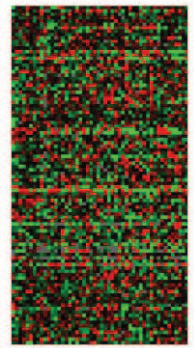
Dimension reduction

face images

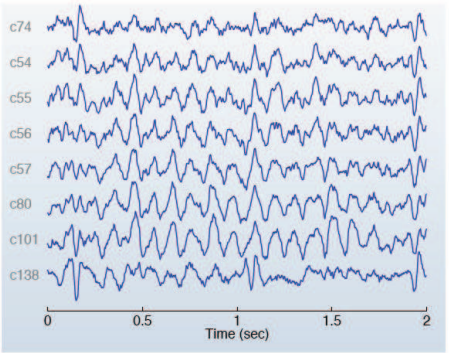
Zambian President Levy Mwanawasa has won a second term in office in an election his challenger Michael Sata accused him of rigging, official results showed on Monday.

According to media reports, a pair of hackers said on Saturday that the Firefox Web browser, commonly perceived as the safer and more customizable alternative to market leader Internet Explorer, is critically flawed. A presentation on the flaw was shown during the ToorCon hacker conference in San Diego.

documents



gene expression data



MEG readings

Unsupervised learning

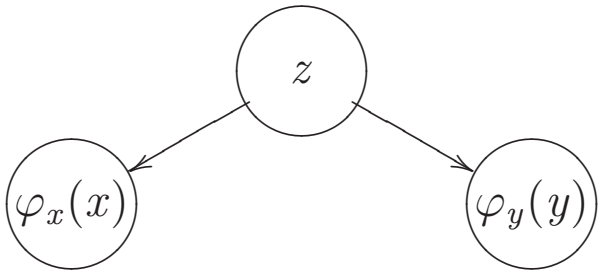
Dimension reduction

- Computational efficiency : space and time savings
- Statistical performance : fewer dimensions $\rightarrow$ regularization
- Visualization : discover underlying structure of the data

$\rightarrow$ PCA and KPCA

Unsupervised learning

Feature extraction



x:

y: "A view from Idyllwild, California, with pine trees and snow capped Marion Mountain under a blue sky."

Unsupervised learning

Feature extraction

- Multimodality : leverage the correlation between the modalities
- Statistical performance : take advantage of both views of the data
- Putting in relation : discover underlying relations between the modalities

$\rightarrow$ CCA and KCCA

Unsupervised learning

Clustering

Unsupervised learning

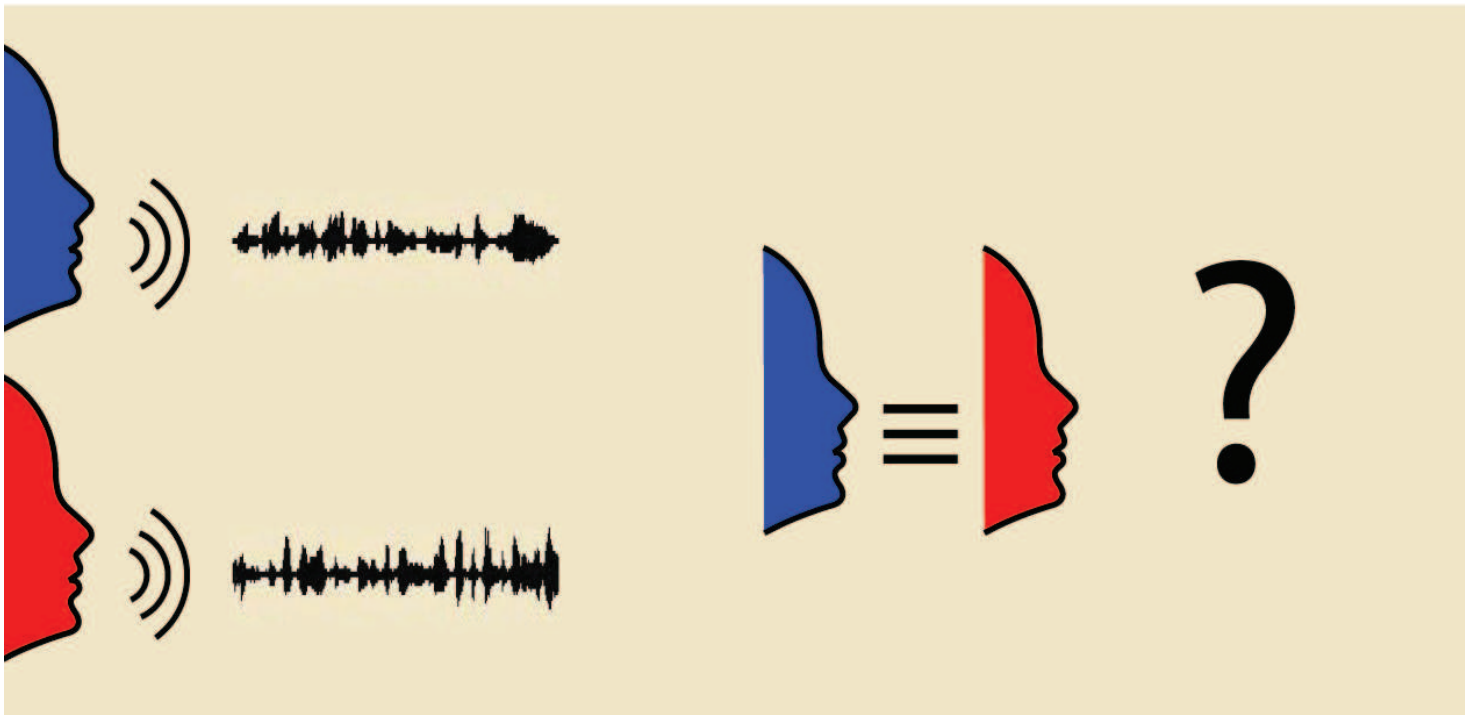
Clustering

- Semantics : grouping datapoints in meaningful clusters
- Statistical performance : intrinsic degrees of freedom of the data
- Visualization : discover groupings between datapoints

$\rightarrow$ spectral clustering, temporal segmentation, and regularized clustering (DIFFRAC)

Unsupervised learning

Detection problems



Unsupervised learning

Detection problems

- Balance risks : control detection rate with a guaranteed false alarm probability
- Power : detect differences not only in mean or covariance

$\rightarrow$ homogeneity testing, change detection

Outline

1 Introduction

2 Kernel methods and feature space

3 Mean element and covariance operator

4 Kernel PCA

5 Kernel CCA

6 Spectral Clustering

Kernel methods

Machine Learning methods taking $\mathbf{K} = [k(X_i, X_j)]_{i,j=1,\ldots,n}$ (Gram matrix as input for processing a sample $\{X_1, \ldots, X_n\}$, where $k(x, y)$ is a similarity measure between x and y defining a positive definite kernel.

Strengths of Kernel Methods

- Minimal assumptions on data types (vectors, strings, trees, graphs, etc.)
- Interpretation of $k(x, y)$ as a dot product $k(x, y) = \langle \phi(x), \phi(y) \rangle_{\mathcal{H}}$ in a reproducing kernel Hilbert space $\mathcal{H}$ where the observations are mapped *via* $[\phi : \mathcal{X} \to \mathcal{H}]$ the feature map $\phi(\bullet) = k(\bullet, \cdot)$

How does the feature space look like ?

Example : space of shapes of birds

How does the feature space look like ?

Feature map ?

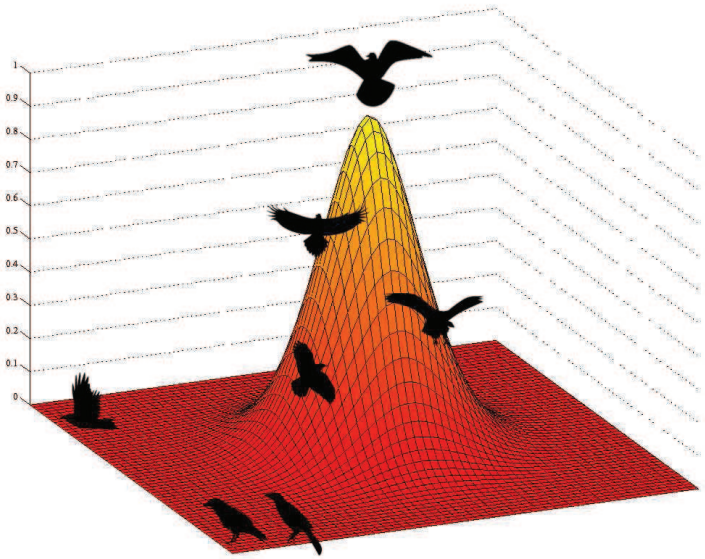
How does the feature map look like ?

$$k\left(\ \raisebox{-0.3ex}{  }\ ,\ \cdot\ \right)$$

How does the feature space look like ?

Feature map ?

The feature map is a function whose values span the whole range of shapes with varying magnitudes.



Examples of Kernels

Kernels on vectors

Polynomial	$k(\mathbf{x}, \mathbf{y}) = (c + (\mathbf{x}, \mathbf{y}))^d$
Laplace	$k(\mathbf{x}, \mathbf{y}) = \exp(-\|\mathbf{x} - \mathbf{y}\|_1/\sigma)$
RBF	$k(\mathbf{x}, \mathbf{y}) = \exp(-\|\mathbf{x} - \mathbf{y}\|_2^2/\sigma^2)$

Examples of Kernels

Kernels on histograms

Kernels built on top of divergence between probability distributions

$$\psi_{JD}(\theta,\theta') = h\left(\frac{\theta+\theta'}{2}\right) - \frac{h(\theta)+h(\theta')}{2},$$

$$\psi_{\chi^2}(\theta,\theta') = \sum_i \frac{(\theta_i-\theta'_i)^2}{\theta_i+\theta'_i}, \quad \psi_{TV}(\theta,\theta') = \sum_i |\theta_i-\theta'_i|,$$

$$\psi_{H_2}(\theta,\theta') = \sum_i |\sqrt{\theta_i}-\sqrt{\theta'_i}|^2, \quad \psi_{H_1}(\theta,\theta') = \sum_i |\sqrt{\theta_i}-\sqrt{\theta'_i}|.$$

$$k(\theta,\theta') = \exp(-\psi(\theta,\theta')/\sigma^2)\ .$$

The kernel jungle

Kernels on histograms

- Pyramid match kernels (Grauman and Darrell, 2005)
- Multiresolution (nested histograms) kernels (Cuturi, 2006)
- Walk and tree-walk kernels (Ramon & Gaertner, 2004 ; Harchaoui & Bach, 2007 ; Mahe et al., 2007)

Kernels from statistical generative models

- Mutual Information Kernels (Seeger, 2002)
- Fisher kernels (see Shawe-Taylor & Cristianini, 2004)

Other kernels

- Kernels of shapes and point coulds (Bach, 2007)
- Kernels on time series (Cuturi, 2007)

How does the feature space look like ?

Classical kernel trick

- Describes what happens to pairs of examples
- Focuses on the *pointwise* effect of the feature map on an example

"Remixed" kernel trick

- Describes what happens to a random sample from a probability distribution
- Focuses on the *global* effect of the feature map on a sample

Outline

1 Introduction

2 Kernel methods and feature space

3 Mean element and covariance operator

4 Kernel PCA

5 Kernel CCA

6 Spectral Clustering

Coordinate-free definitions of mean and covariance

Usual definitions

- need explicit basis to define quantities
  $\rightarrow$ tricky in high-dimensional/∞-dimensional feature spaces

Coordinate-free definitions

- define quantities through their projections along any direction
  $\rightarrow$ allow direct application of the *reproducing property*

Mean vector and mean element

Empirical mean element

Empirical mean vector $\hat{\mu}$ of $X_1, \ldots, X_m \sim \mathbb{P}$

$$\forall \mathbf{w} \in \mathcal{X},$$

$$(\hat{\mu}, \mathbf{w}) \stackrel{\text{def}}{=} \frac{1}{m} \sum_{\ell=1}^{m} (\mathbf{x}_\ell, \mathbf{w})$$

Empirical mean element $\hat{\mu}$ of $X_1, \ldots, X_m \sim \mathbb{P}$

$$\forall f \in \mathcal{H},$$

$$\langle \hat{\mu}, f \rangle_{\mathcal{H}} \stackrel{\text{def}}{=} \frac{1}{m} \sum_{\ell=1}^{m} \langle \phi(\mathbf{x}_\ell), f \rangle_{\mathcal{H}}$$

Mean vector and mean element

Empirical mean element

Empirical *mean element* $\hat{\mu}$ of $\mathbf{x}_1, \ldots, \mathbf{x}_m \sim \mathbb{P}$

$$\forall f \in \mathcal{H},$$

$$\langle \hat{\mu}, f \rangle_{\mathcal{H}} \stackrel{\text{def}}{=} \frac{1}{m} \sum_{\ell=1}^{m} \langle \phi(\mathbf{x}_\ell), f \rangle_{\mathcal{H}}$$

$$\langle \hat{\mu}, f \rangle_{\mathcal{H}} \stackrel{\text{def}}{=} \frac{1}{m} \sum_{\ell=1}^{m} \langle k(\mathbf{x}_\ell, \cdot), f \rangle_{\mathcal{H}}$$

$$\stackrel{\text{def}}{=} \frac{1}{m} \sum_{\ell=1}^{m} f(\mathbf{x}_\ell) \text{ (reproducing property)}$$

$$\stackrel{\text{def}}{=} \frac{1}{m} \sum_{\ell=1}^{m} \sum_{j=1}^{n} \alpha_j k(\mathbf{x}_j, \mathbf{x}_\ell) \text{ , if } f(\cdot) = \sum_{j=1}^{n} \alpha_j k(\mathbf{x}_j, \cdot)$$

Centering in feature space

Gram matrix

$\mathbf{K} = [k(X_i, X_j)]_{i,j=1,\ldots,n}$ of all evaluations of the kernel $k(\cdot, \cdot)$ on the sample $\mathbf{x}_1, \ldots, \mathbf{x}_n$.

Centering in feature space

To center all $\phi(\mathbf{x}_1), \ldots, \phi(\mathbf{x}_n)$ *simultaneously*, do

$$\mathbf{K} \leftarrow \tilde{\mathbf{K}} = \Pi \mathbf{K} \Pi \ ,$$

where

$$\Pi = \mathbf{I}_n - \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^T \ .$$

Covariance matrix and covariance operator

Empirical covariance operator

Empirical covariance matrix $\hat{\Sigma}$ of $\mathbf{x}_1, \ldots, \mathbf{x}_m \sim \mathbb{P}$

$$\forall \mathbf{w}, \mathbf{v} \in \mathcal{X},$$

$$(\mathbf{w}, \hat{\Sigma}\mathbf{v}) = \frac{1}{m} \sum_{\ell=1}^{m} (\mathbf{w}, \tilde{\mathbf{x}}_\ell)(\tilde{\mathbf{x}}_\ell, \mathbf{v})$$

$$\tilde{\mathbf{x}}_\ell = \mathbf{x}_\ell - \hat{\mu} \ .$$

Empirical covariance operator $\hat{\Sigma}$ of $\mathbf{x}_1, \ldots, \mathbf{x}_m \sim \mathbb{P}$

$$\forall f, g \in \mathcal{H},$$

$$\left\langle f, \hat{\Sigma} g \right\rangle = \frac{1}{m} \sum_{\ell=1}^{m} \left\langle f, \tilde{\phi}(\mathbf{x}_\ell) \right\rangle \left\langle \tilde{\phi}(\mathbf{x}_\ell), g \right\rangle$$

$$\tilde{\phi}(\mathbf{x}_\ell) = \phi(\mathbf{x}_\ell) - \hat{\mu} \ .$$

Covariance matrix and covariance operator

Covariance operator

Empirical covariance operator $\hat{\Sigma}$ of $\mathbf{x}_1, \ldots, \mathbf{x}_m \sim \mathbb{P}$

$$\forall f, g \in \mathcal{H},$$

$$\left\langle f, \hat{\Sigma} g \right\rangle = \frac{1}{m} \sum_{\ell=1}^{m} \left\langle f, \tilde{\phi}(\mathbf{x}_\ell) \right\rangle \left\langle \tilde{\phi}(\mathbf{x}_\ell), g \right\rangle$$

$$= \frac{1}{m} \sum_{\ell=1}^{m} \{ f(\mathbf{x}_\ell) - \langle \hat{\mu}, f \rangle_{\mathcal{H}} \} \{ f(\mathbf{x}_\ell) - \langle \hat{\mu}, g \rangle_{\mathcal{H}} \} \, .$$

Computing variance along a direction in feature space

Gram matrix

$\mathbf{K} = [k(X_i, X_j)]_{i,j=1,\ldots,n}$ of all evaluations of the kernel $k(\cdot, \cdot)$ on $x_1, \ldots, x_n$.

Covariance along two directions

$$\left\langle f, \hat{\Sigma} g \right\rangle = \frac{1}{m} \alpha^T \tilde{\mathbf{K}} \tilde{\mathbf{K}} \beta \ ,$$

where

$$f(\cdot) = \sum_{j=1}^{n} \alpha_j k(\mathbf{x}_j, \cdot) \ ,$$

$$g(\cdot) = \sum_{j=1}^{n} \beta_j k(\mathbf{x}_j, \cdot) \ .$$

Mean element and covariance operator

Population mean element and covariance operator

Population mean element μ and population covariance operator Σ of $\mathbf{x} \sim \mathbb{P}$

$$\langle \mu, f \rangle_{\mathcal{H}} \stackrel{\text{def}}{=} \mathbb{E}[f(\mathbf{x})] \,, \quad \forall f \in \mathcal{H}$$

$$\langle f, \Sigma g \rangle_{\mathcal{H}} \stackrel{\text{def}}{=} \mathrm{Cov}[f(\mathbf{x}), g(\mathbf{x})] \,, \quad \forall f, g \in \mathcal{H}$$

Empirical mean element and covariance operator

Empirical mean element $\hat{\mu}$ and empirical covariance operator $\hat{\Sigma}$ of $\mathbf{x}_1, \ldots, \mathbf{x}_m \sim \mathbb{P}$

$$\langle \hat{\mu}, f \rangle_{\mathcal{H}} \stackrel{\text{def}}{=} \frac{1}{m} \sum_{\ell=1}^{m} f(\mathbf{x}_\ell) \,, \quad \forall f \in \mathcal{H}$$

$$\left\langle f, \hat{\Sigma} g \right\rangle_{\mathcal{H}} \stackrel{\text{def}}{=} \frac{1}{m} \sum_{\ell=1}^{m} \{f(\mathbf{x}_\ell) - \langle \hat{\mu}, f \rangle_{\mathcal{H}}\}\{f(\mathbf{x}_\ell) - \langle \hat{\mu}, g \rangle_{\mathcal{H}}\} \quad \forall f, g \in \mathcal{H}$$

Some casual considerations before the real stuff

Supervised learning

- least-square regression, kernel ridge regression, multilayer-perceptron
  $\rightarrow$ tackled through (possibly a sequence of) linear of systems
- Operation \ in Matlab/Octave

Unsupervised learning

- (kernel) principal component analysis, (kernel) canonical correlation analysis, spectral clustering
  $\rightarrow$ tackled through (possibly a sequence of) eigenvalue problems
- Function eigs in Matlab/Octave

Outline

1 Introduction

2 Kernel methods and feature space

3 Mean element and covariance operator

4 Kernel PCA

5 Kernel CCA

6 Spectral Clustering

Kernel Principal Component Analysis (Schölkopf et al., 1998 ; Shawe-Taylor & Cristianini, 2004)

Principal Component Analysis (PCA)

A brief refresher

- Let $\mathbf{x}_1, \ldots, \mathbf{x}_n$ a dataset of points in $\mathbf{R}^d$
- PCA is a classical method in multivariate statistics to define a set of orthogonal directions, called *principal components*, that capture the maximum variance
- Projection along the first 2-3 principal components allows to visualize the dataset

Refresher on Principal Component Analysis

Computational aspects

- Maximum variance criterion corresponds to a Rayleigh quotient
- PCA boils down to an eigenvalue problem on the *centered* covariance matrix $\hat{\Sigma}$ of the dataset, *i.e.* the principal components $\mathbf{w}_1, \ldots, \mathbf{w}_d$ are the eigenvectors of $\hat{\Sigma}$ (assuming $n > d$)
- Computational complexity : $O(ndc)$ in time with a *Singular Value Decomposition* (SVD ; see `eigs` in Matlab/Octave), with n the number of points, d the dimension, c the number of principal components retained ; stochastic approximation version for nonstationary/large-scale datasets.

Variance along a direction and Rayleigh quotients

Variance along a direction

PCA seeks for directions $\mathbf{w}_1, \ldots, \mathbf{w}_c$ such that

$$\begin{aligned}
\mathbf{w}_j &= \operatorname{argmax}_{\mathbf{w} \in \mathbb{R}^d ; \mathbf{w}_j \perp \{\mathbf{w}_1, \ldots, \mathbf{w}_{j-1}\}} \operatorname{Var}_{\text{emp}} \frac{(\mathbf{w}, \mathbf{x})}{(\mathbf{w}, \mathbf{w})} \\
&= \operatorname{argmax}_{\mathbf{w} \in \mathbb{R}^d ; \mathbf{w}_j \perp \{\mathbf{w}_1, \ldots, \mathbf{w}_{j-1}\}} \frac{1}{m} \sum_{i=1}^{m} \frac{(\mathbf{w}, \mathbf{x}_i)^2}{(\mathbf{w}, \mathbf{w})} \\
&= \operatorname{argmax}_{\mathbf{w} \in \mathbb{R}^d ; \mathbf{w}_j \perp \{\mathbf{w}_1, \ldots, \mathbf{w}_{j-1}\}} \underbrace{\frac{(\mathbf{w}, \hat{\Sigma}\mathbf{w})}{(\mathbf{w}, \mathbf{w})}}_{\text{Rayleigh quotient}} .
\end{aligned}$$

Principal components $\mathbf{w}_1, \ldots, \mathbf{w}_c$ are the first c eigenvectors of $\hat{\Sigma}$.

Variance along a direction and Rayleigh quotients

Variance along a direction

KPCA seeks for directions $f_1, \ldots, f_c$ such that

$$
\begin{aligned}
f_j &= \text{argmax}_{f \in \mathcal{H}; f_j \perp \{f_1, \ldots, f_{j-1}\}} \operatorname{Var}_{\text{emp}} \frac{\langle f, \phi(\mathbf{x}) \rangle}{\langle f, f \rangle} \\
&= \text{argmax}_{f \in \mathcal{H}; f_j \perp \{f_1, \ldots, f_{j-1}\}} \frac{1}{m} \sum_{i=1}^{m} \frac{\langle f, \phi(\mathbf{x}_i) \rangle^2}{\langle f, f \rangle} \\
&= \text{argmax}_{f \in \mathcal{H}; f_j \perp \{f_1, \ldots, f_{j-1}\}} \underbrace{\frac{\left\langle f, \hat{\Sigma} f \right\rangle}{\langle f, f \rangle}}_{\text{Rayleigh quotient}} .
\end{aligned}
$$

Principal components $f_1, \ldots, f_c$ are the first c eigenvectors of $\hat{\Sigma}$. Is that it ?

Rescue theorems

Properties of covariance operators

RKHS Covariance operators are (Zwald et al., 2005, Harchaoui et al., 2008)

- self-adjoint (∞-dimensional counterpart of symmetric)
- positive
- trace-class

Consequence

The covariance operator $\hat{\Sigma}$ and the centered Gram matrix $\tilde{\mathbf{K}}$ share the same eigenvalues on the nonzero part of their spectra, and their eigenvectors are related by a simple relation.

Kernel Principal Component Analysis

KPCA algorithm

- Center the Gram matrix
- Performs an SVD on $\tilde{\mathbf{K}}$ to get the first c eigenvector/eigenvalue pairs $(e_j, \lambda_j)_{j=1,\ldots,c}$.
- Normalize the eigenvector $\tilde{e}_j \leftarrow e_j/\lambda_j$
- Projections onto the j-th eigenvectors is given by $\tilde{\mathbf{K}}\tilde{e}_j$

Computational aspects of KPCA

Computational aspects

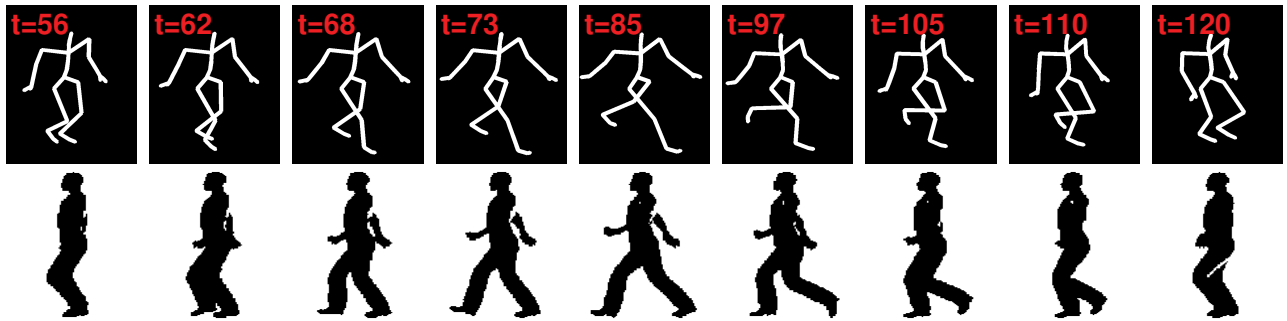
- Maximum variance in feature space corresponds to a Rayleigh quotient
- KPCA boils down to an eigenvalue problem involving the centered auto-covariance matrices $\tilde{\mathbf{K}}$
- Computational complexity : $O(cn^2)$ in time with a *Singular Value Decomposition* (SVD ; see `eigs` in Matlab/Octave), with n the number of points, c the number of principal components retained ; stochastic approximation version for nonstationary/large-scale datasets.

Low-dimensional representation with KPCA

Human body pose representation

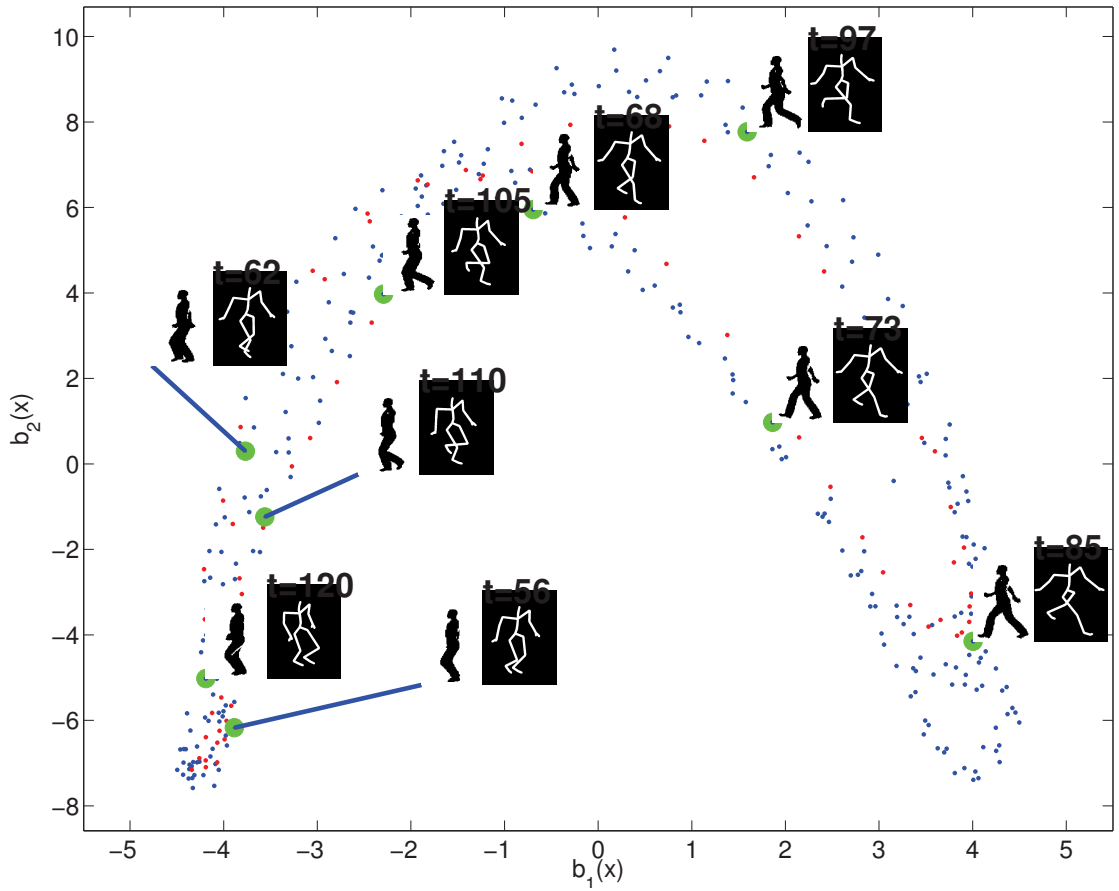
- Walking sequence of length 400 (containing about 3 walking cycles) obtained from the CMU Mocap database
- Data : silhouette images of size (160 100) taken at a side view

Human body pose representation (Kim & Pavlovic, 2008)



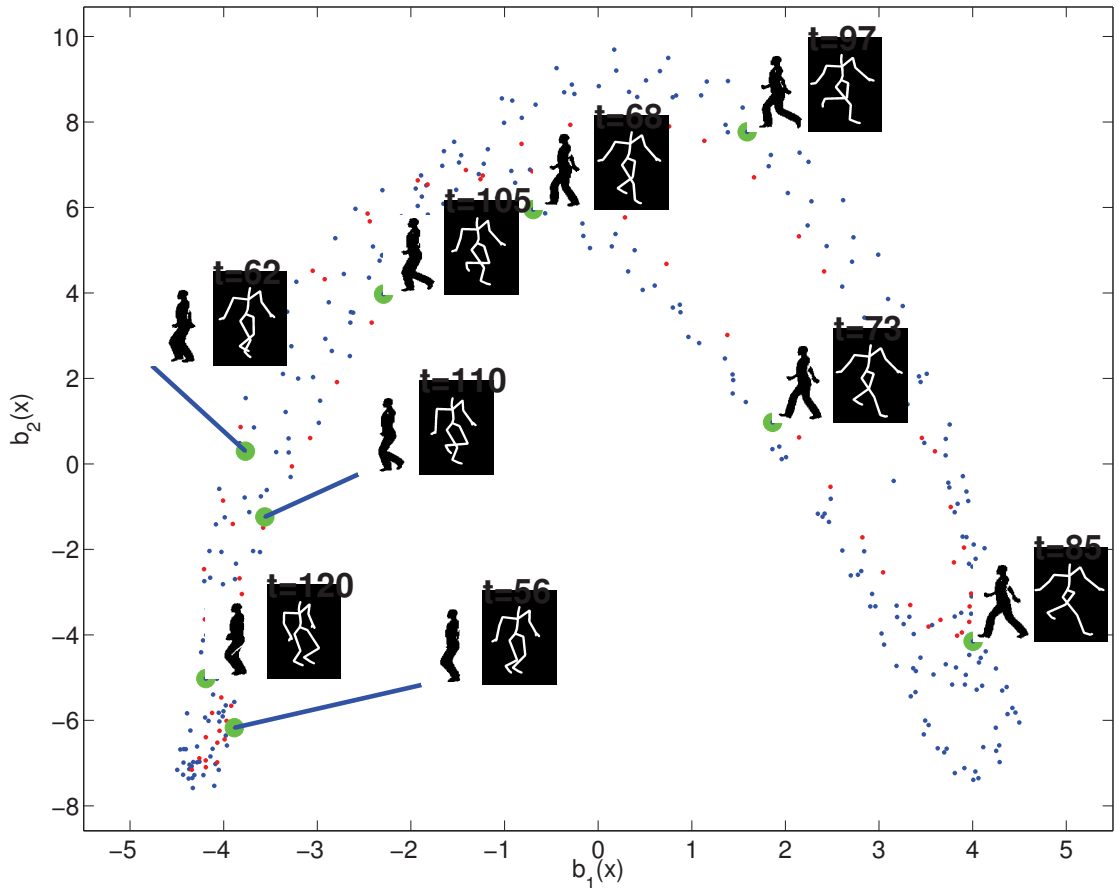
Low-dimensional representation with KPCA

Human body pose representation



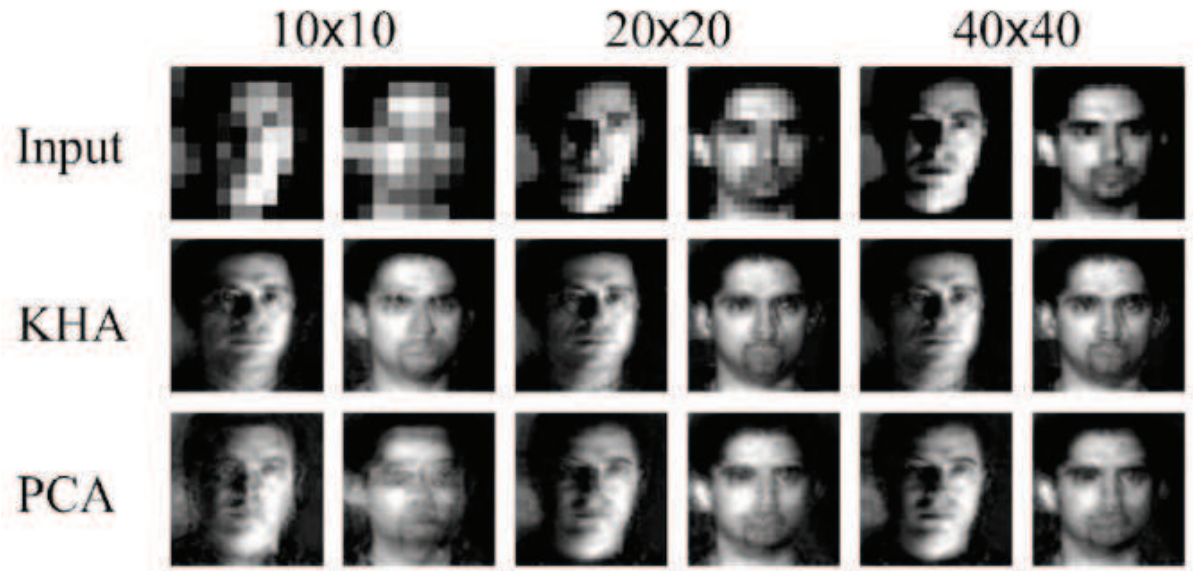
Low-dimensional representation with KPCA

Human body pose representation



Super-resoluton with KPCA (Kim et al., 2005)

Super-resolution



KPCA+n : unsupervised alignment (de la Torre & Nguyen, 2009)

Unsupervised alignment

KPCA + Rigid motion model

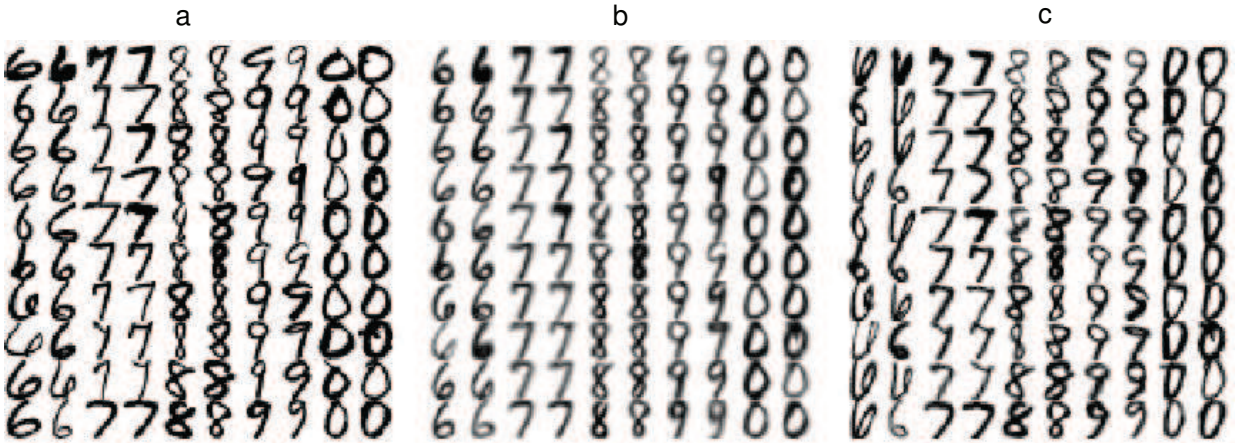


Figure: original USPS (a), Parametrized KPCA (b), annealing (c)

Applications

Popular

- Image denoising (digits, faces, etc.)
- Visualization of bioinformatics data (strings, proteins, etc.)
- Dimension-reduction of high-dimensional features (appearance, interest points, etc.)

Not so well-know property of KPCA

- Regularization in supervised learning can be enforced by projection $\rightarrow$ careful not to regularize twice !
- Useful in settings where ridge-regularization is impractical (Zwald et al., 2009 ; Harchaoui et al., 2009 ; Guillaumin et al., 2010)

Outline

1 Introduction

2 Kernel methods and feature space

3 Mean element and covariance operator

4 Kernel PCA

5 Kernel CCA

6 Spectral Clustering

Kernel Canonical Correlation Analysis (Shawe-Taylor & Cristianini, 2004)

Canonical Correlation Analysis (CCA)

A brief refresher

- Let $(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_n, \mathbf{y}_n)$ a dataset of points in $\mathbf{R}^d \times \mathbf{R}^p$, for which two *views* are available : the "x-view" and the "y-view"
- CCA is a classical method from multivariate statistics to define a set of pairs of orthogonal directions, called *canonical variates*, that capture the *maximum correlation* between the two views.
- Projection along the first 2-3 pairs of canonical variates resp. of "x-view" and the "y-view" allows to visualize the components dataset maximizing the correlation between the two views.

Refresher on Canonical Correlation Analysis

Computational aspects

- Maximum correlation criterion corresponds to a generalized Rayleigh quotient
- CCA boils down to a generalized eigenvalue problem involving the (centered) auto-covariance matrices $\hat{\Sigma}_{\mathbf{xx}}$ and $\hat{\Sigma}_{\mathbf{yy}}$ and on the (centered) cross-covariance matrix $\hat{\Sigma}_{\mathbf{xy}}$
- Computational complexity : $O(n(d+p)c)$ in time with a *Singular Value Decomposition* (SVD ; see `eigs` in Matlab/Octave), with n the number of points, d the dimension, c the number of canonical variates retained ; stochastic approximation version for nonstationary/large-scale datasets.

Cross-covariance matrix and cross-covariance operator

Empirical cross-covariance matrix

Empirical cross-covariance matrix $\hat{\Sigma}_{\mathbf{xy}}$ of $\mathbf{x}_1, \ldots, \mathbf{x}_m \sim \mathbb{P}_{\mathbf{x}}$ and $\mathbf{y}_1, \ldots, \mathbf{y}_m \sim \mathbb{P}_{\mathbf{y}}$

$$\forall \mathbf{w}, \mathbf{v} \in \mathcal{X}, \mathcal{Y}$$

$$(\mathbf{w}, \hat{\Sigma}_{\mathbf{xy}} \mathbf{v}) = \frac{1}{m} \sum_{\ell=1}^{m} (\mathbf{w}, \tilde{\mathbf{x}}_\ell)(\tilde{\mathbf{y}}_\ell, \mathbf{v})$$

$$\tilde{\mathbf{x}}_\ell = \mathbf{x}_\ell - \hat{\mu}_{\mathbf{x}}$$

$$\tilde{\mathbf{y}}_\ell = \mathbf{y}_\ell - \hat{\mu}_{\mathbf{y}} \ .$$

Empirical cross-covariance operator $\hat{\Sigma}_{\mathbf{xy}}$ of $\mathbf{x}_1, \ldots, \mathbf{x}_m \sim \mathbb{P}_{\mathbf{x}}$ and $\mathbf{y}_1, \ldots, \mathbf{y}_m \sim \mathbb{P}_{\mathbf{y}}$

$$\forall f, g \in \mathcal{F}, \mathcal{H}$$

$$\left\langle f, \hat{\Sigma}_{\mathbf{xy}} g \right\rangle = \frac{1}{m} \sum_{\ell=1}^{m} \left\langle f, \tilde{\phi}(\mathbf{x}_\ell) \right\rangle \left\langle \tilde{\psi}(\mathbf{y}_\ell), g \right\rangle$$

$$\tilde{\phi}(\mathbf{x}_\ell) = \phi(\mathbf{x}_\ell) - \hat{\mu}_{\mathbf{x}}$$

$$\tilde{\psi}(\mathbf{y}_\ell) = \psi(\mathbf{y}_\ell) - \hat{\mu}_{\mathbf{y}} \ .$$

Covariance along two directions and generalized Rayleigh quotients

Covariance along two directions

CCA seeks for directions $(\mathbf{w}_1, \mathbf{v}_1)$ such that [1]

$$(\mathbf{w}_1, \mathbf{v}_1) = \text{argmax}_{(\mathbf{w},\mathbf{v})\in\mathbb{R}^d\times\mathbb{R}^p} \frac{\text{Cov}((\mathbf{w},\mathbf{x}),(\mathbf{v},\mathbf{y}))}{\text{Var}^{1/2}((\mathbf{w},\mathbf{x})\text{Var}^{1/2}((\mathbf{v},\mathbf{y})}$$

$$= \text{argmax}_{(\mathbf{w},\mathbf{v})\in\mathbb{R}^d\times\mathbb{R}^p} \frac{(\mathbf{w}, \hat{\Sigma}_{\mathbf{xy}}\mathbf{v})}{(\mathbf{w}, \hat{\Sigma}_{\mathbf{xx}}\mathbf{w})^{1/2}(\mathbf{v}, \hat{\Sigma}_{\mathbf{yy}}\mathbf{v})^{1/2}} \ .$$

1. focus here on the first pair of canonical variates

Covariance along two directions and generalized Rayleigh quotients

Generalized Rayleigh quotient

Canonical variates $(\mathbf{w}_1, \mathbf{v}_1), \ldots, (\mathbf{w}_c, \mathbf{v}_c)$ are the first c pairs of vectors solutions of the generalized eigenvalue problem

$$\begin{bmatrix} \mathbf{0} & \hat{\Sigma}_{\mathbf{xy}} \\ \hat{\Sigma}_{\mathbf{xy}} & \mathbf{0} \end{bmatrix} \begin{pmatrix} \mathbf{w} \\ \mathbf{v} \end{pmatrix} = \rho \begin{bmatrix} \hat{\Sigma}_{\mathbf{xx}} & \mathbf{0} \\ \mathbf{0} & \hat{\Sigma}_{\mathbf{yy}} \end{bmatrix} \begin{pmatrix} \mathbf{w} \\ \mathbf{v} \end{pmatrix} .$$

Covariance along two directions and generalized Rayleigh quotients

Covariance along two directions

Kernel CCA seeks for directions (f_1, g_1) such that [2]

$$(f_1, g_1) = \text{argmax}_{(f,g)\in\mathcal{H}\times\mathcal{H}} \frac{\text{Cov}(\langle f, \phi(\mathbf{x})\rangle, \langle g, \psi(\mathbf{y})\rangle)}{\{\text{Var}\langle f, \phi(x)\rangle + \epsilon\langle f, f\rangle\}^{1/2}\{\text{Var}\langle g, \psi(x)\rangle + \epsilon\langle g, g\rangle\}^{1/2}}$$

$$= \text{argmax}_{(f,g)\in\mathcal{H}\times\mathcal{H}} \frac{\left\langle f, \hat{\Sigma}_{\mathbf{xy}} g\right\rangle}{\left\langle f, (\hat{\Sigma}_{\mathbf{xx}} + \frac{n\epsilon}{2})g\right\rangle^{1/2}\left\langle f, (\hat{\Sigma}_{\mathbf{yy}} + \frac{n\epsilon}{2})g\right\rangle^{1/2}} .$$

2. focus here on the first pair of canonical variates

Correlation along two directions

Generalized eigenvalue problem

Coefficients of canonical variates $(\alpha_1, \beta_1), \ldots, (\alpha_c, \beta_c)$ are the first c pairs of vectors solutions of the generalized eigenvalue problem

$$\begin{bmatrix} \mathbf{0} & \tilde{\mathbf{K}}_x \tilde{\mathbf{K}}_y \\ \tilde{\mathbf{K}}_x \tilde{\mathbf{K}}_y & \mathbf{0} \end{bmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \rho \begin{bmatrix} \tilde{\mathbf{K}}_x \tilde{\mathbf{K}}_x & \mathbf{0} \\ \mathbf{0} & \tilde{\mathbf{K}}_y \tilde{\mathbf{K}}_y \end{bmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} .$$

Computational aspects of KCCA

Computational aspects

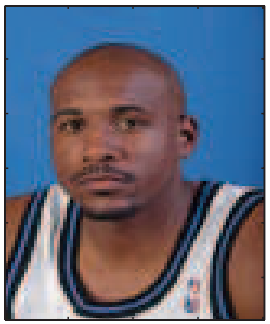
- Maximum correlation in feature space corresponds to a Rayleigh quotient
- KCCA boils down to a generalized eigenvalue problem involving the squared centered Gram matrices matrices $\tilde{\mathbf{K}_x}^2$ $\tilde{\mathbf{K}_y}^2$ and the product of the Gram matrices $\tilde{\mathbf{K}_x}\tilde{\mathbf{K}_y}$.
- Computational complexity : $O(cn^2)$ in time with a *Singular Value Decomposition* (SVD ; see `eigs` in Matlab/Octave), with n the number of points, c the number of principal components retained ; stochastic approximation version for nonstationary/large-scale datasets.

Multimedia content based image retrieval with KCCA

Multimedia

- Multimedia content $\rightarrow$ multi-view data
- images with text captions : text $\rightarrow$ "x"-view, image $\rightarrow$ "y"-view

Multimedia content based image retrieval (Hardoon et al, 2004)



I_1

I_2

I_3

Image	Label	Keywords
I_1	Sports	position college weight born lbs height guard
I_2	Aviation	na air convair wing
I_3	Paintball	check darkside force gog strike odt

Outline

1 Introduction

2 Kernel methods and feature space

3 Mean element and covariance operator

4 Kernel PCA

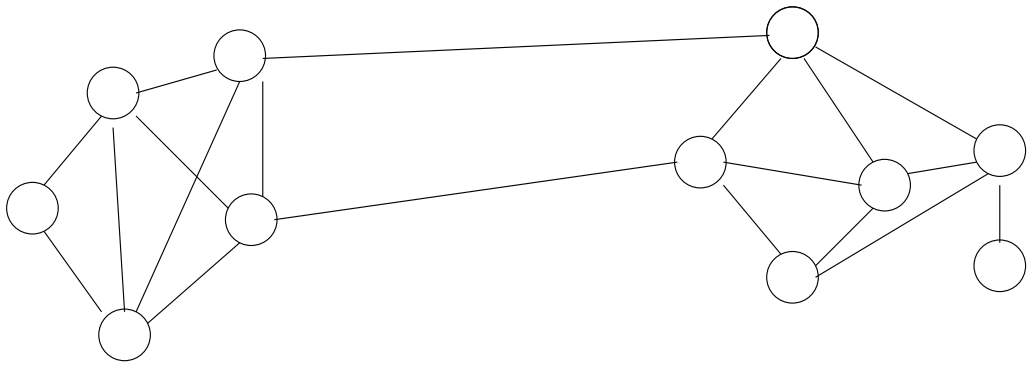
5 Kernel CCA

6 Spectral Clustering

Spectral clustering (von Luxburg, 2007)

Overview

- Let $\mathbf{x}_1, \ldots, \mathbf{x}_n$ a dataset of points in $\mathbf{R}^d$, along with pairwise similarities $s(\mathbf{x}_i, \mathbf{x}_j), 1 \leq i, j \leq n$.
- Build similarity graph, with data points as *vertices* and similarities as *edge lengths*
- Spectral clustering finds the best cut through the graph



Laplacian matrix and spectral clustering

Laplacian matrix

Spectral clustering relies on the spectrum of the Laplacian matrix $\mathbf{L}$

$$\mathbf{L} = \underbrace{\mathbf{D}}_{\text{degree matrix}} - \underbrace{\mathbf{S}}_{\text{similarity matrix}} ,$$

where

$$\mathbf{D} = \text{Diag}(\deg(\mathbf{x}_1), \ldots, \deg(\mathbf{x}_n))$$

$$\deg(\mathbf{x}_i) = \sum_{j=1}^{n} s(\mathbf{x}_i, \mathbf{x}_j) \ .$$

Laplacian matrix and the Laplace-Beltrami operator

Laplacian matrix

The Laplacian matrix measures the discrete variation of f along the graph

$$\forall f \in \mathbb{R}^d \, , f^T \mathbf{L} f = \frac{1}{2} \sum_{j=1}^{n} s(\mathbf{x}_i, \mathbf{x}_j)(f_i - f_j)^2 \, ,$$

$$f^T \mathbf{L} f \approx \frac{1}{2} \sum_{j=1}^{n} \frac{(f_i - f_j)^2}{d(\mathbf{x}_i, \mathbf{x}_j)^2} \, , \quad \text{if } s(\mathbf{x}_i, \mathbf{x}_j) \approx \frac{1}{d(\mathbf{x}_i, \mathbf{x}_j)^2} \, .$$

Laplacian operator

The Laplacian matrix is the discrete counterpart of the Laplace[3] operator

$$\forall f \in \mathbb{R}^d \, , \langle f, \Delta f \rangle = \int |\nabla f|^2 dx \, .$$

3. Laplace-Beltrami generalizes the Laplace operator to manifold data.

Rescue theorems

Properties of Laplacian operators

Laplacian matrices are (von Luxburg et al., 2005, Gine and Koltchinskii, 2008)

- symmetric
- positive definite
- smallest eigenvalue is 0, and associated eigenvector $\mathbf{1}$

Interpretation

- Multiplicity of eigenvalue 0 is the number of connected components of the graph $A_1, \ldots, A_k$
- Eigenspace spanned by the characteristic functions $\mathbf{1}_{A_1}, \ldots, \mathbf{1}_{A_k}$ on those components (so all eigenvectors are piecewise constants)

Normalization

Normalized graph Laplacians

Graph Laplacian matrices can be normalized in two ways [4]

$$\mathbf{L}_{rw} = D^{-1}L \qquad \text{random walk normalization ,}$$
$$\mathbf{L}_{sym} = D^{-1/2}LD^{-1/2} \qquad \text{symmetrized normalization .}$$

Interpretation

- $\mathbf{L}_{rw}$ and $\mathbf{L}_{sym}$ share similar spectral properties with Λ
- Normalized graph Laplacians are better understood theoretically and are consistent under general assumptions in large-sample settings
- Un-normalized ones are still used (!) despite their lack of consistency in some cases in large-sample settings.

4. Caution : eigenspace of $\mathbf{L}_{rw}$ spanned by the $\mathbf{1}_{A_1}, \ldots, \mathbf{1}_{A_k}$; eigenspace of $\mathbf{L}_{sym}$ spanned by the $D^{1/2}\mathbf{1}_{A_1}, \ldots, D^{1/2}\mathbf{1}_{A_k}$.

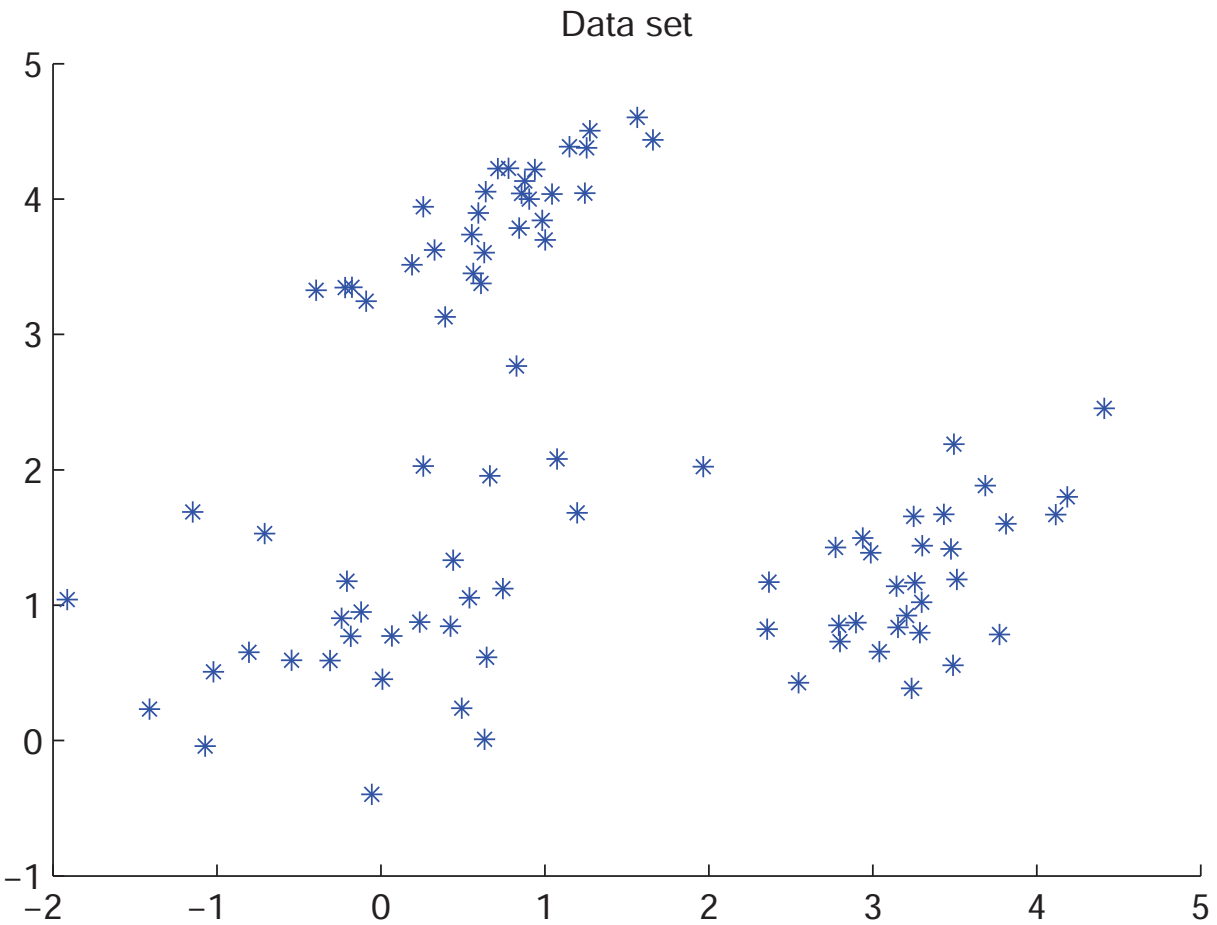
Spectral clustering

Spectral clustering algorithm

- Build similarity graph
- Performs an SVD on $\mathbf{L}_{rw}$ or $\mathbf{L}_{sym}$ to get the first k eigenvector/eigenvalue pairs $(v_j, \lambda_j)_{j=1,\ldots,c}$.
- Build the matrix $V = [v_1, \ldots, v_k]$ stacking the k eigenvectors as columns
- Launch your *favourite clustering algorithm* on the n rows of V

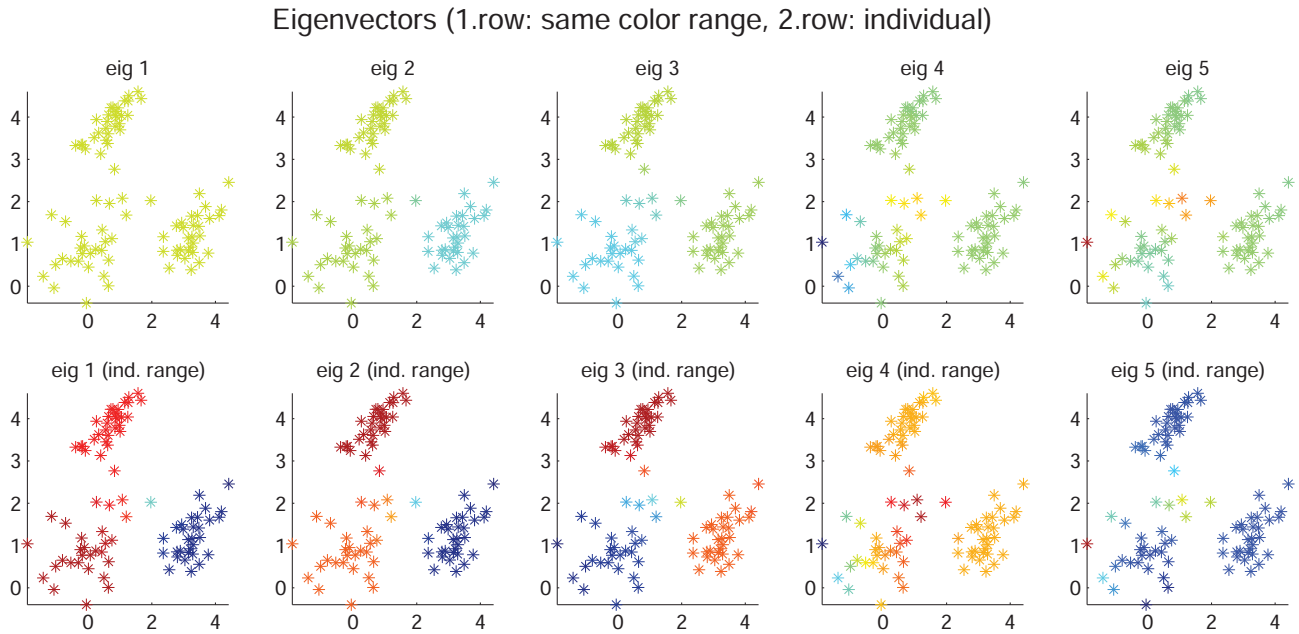
Example

2D example with 3 clusters



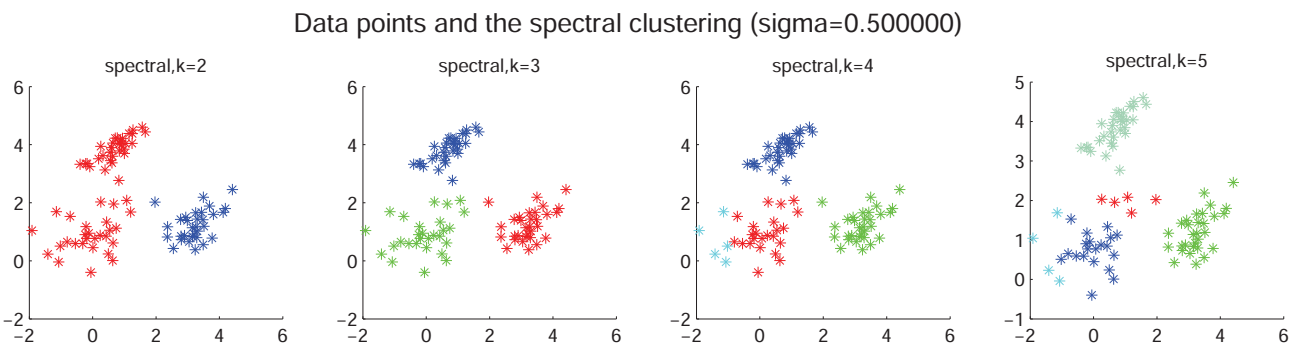
Example

Projections onto eigenvectors



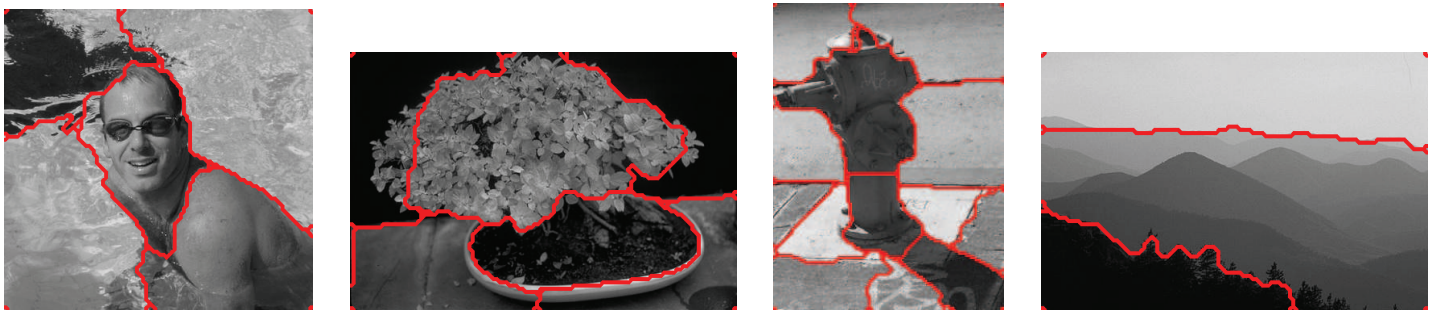
Example

Clustering obtained with k-means as the favourite clustering algorithm



Spectral clustering for image segmentation

Image segmentation algorithm



GrabCut and foreground extraction

Interactive foreground extraction algorithm

Kernel-based Methods for Unsupervised Learning

LEAR project-team, INRIA

Zaid Harchaoui

Grenoble, July 30th 2010

Outline

1 Discriminative clustering

2 Temporal Segmentation

Outline

1 Discriminative clustering

2 Temporal Segmentation

Summary

- **Discriminative clustering** = find labels that maximize linear separability
- **Multiclass square loss** for classification = cost function in closed form
- Optimization of the labels by **convex relaxation**
- Efficient optimization algorithm by **partial dualization**
- Application in **semi-supervised learning**

Classification with square loss

- n points $\mathbf{x}_1, \ldots, \mathbf{x}_n$ in $\mathbb{R}^d$, represented in a matrix $X \in \mathbb{R}^{n \times d}$.
- Labels = partitions of $\{1, \ldots, n\}$ into $k > 1$ clusters, represented by *indicator matrices*

$$y \in \{0,1\}^{n \times k} \text{ such that } y 1_k = 1_n$$

- Regularized linear regression problem of y given X :

$$J(y, X, \kappa) = \min_{\mathbf{w} \in \mathbb{R}^{d \times k},\ b \in \mathbb{R}^{1 \times k}} \frac{1}{n} \|y - X\mathbf{w} - 1_n b\|_F^2 + \kappa \ \mathrm{Tr}\ \mathbf{w}^\top \mathbf{w},$$

  - Multi-label classification problems with square loss functions
  - Solution in closed form (with $\Pi_n = I_n - \frac{1}{n} 1_n 1_n^\top$) :

$$\mathbf{w}^* = (X^\top \Pi_n X + n\kappa I_n)^{-1} X^\top \Pi_n y \quad \text{and} \quad b^* = \frac{1}{n} 1_n^\top (y - X\mathbf{w}^*)$$

Discriminative clustering cost

- **Discriminative clustering** consists in finding labels such that they lead to best linear separation by a discriminative classifier (Xu et al., 2004, 2005)
- Use square loss for multi-class classification
- Main advantages
  - minimizing the regularized cost in closed form
  - including a bias term by simply centering the data
- Optimal value equal to $J(y, X, \kappa) = \operatorname{Tr} yy^\top A(X, \kappa)$, where

$$A(X,\kappa) = \frac{1}{n}\Pi_n(I_n - X(X^\top \Pi_n X + n\kappa I)^{-1}X^\top)\Pi_n$$

Diffrac

- Optimization problem : minimize $\mathrm{Tr} yy^\top A(X, \kappa)$ with respect to y (indicator matrices)
- The cost function only involves the matrix $M = yy^\top \in \mathbb{R}^{n \times n}$ = k-class **equivalence matrix** $\in \{0, 1\}^{n \times n}$
- Convex outer approximation for M
  - M is positive semidefinite (denoted as $M \succcurlyeq 0$)
  - the diagonal of M is equal to 1_n (denoted as $\mathrm{diag}(M) = 1_n$)
  - if M corresponds to at most k clusters, we have $M \succcurlyeq \frac{1}{k} 1_n 1_n^\top$
- Convex set :

$$\mathcal{C}_k = \{M \in \mathbb{R}^{n \times n},\ M = M^\top,\ \mathrm{diag}(M) = 1_n,\ M \geqslant 0,\ M \succcurlyeq \tfrac{1}{k} 1_n 1_n^\top\}$$

Minimum cluster sizes

- Avoid trivial solution by imposing a minimum size λ_0 for each cluster, through :
  - **Row sums** : $M1_n \geqslant \lambda_0 1_n$ and $M1_n \leqslant (n-(k-1)\lambda_0)1_n$ (same constraint as Xu et al., 2005).
  - **Eigenvalues** : The sizes of the clusters are exactly the k largest eigenvalues of $M \Rightarrow$ constraint equivalent to $\sum_{i=1}^n 1_{\lambda_i(M) \geqslant \lambda_0} \geqslant k$, where $\lambda_1(M), \ldots, \lambda_n(M)$ are the n eigenvalues of M.
    - Non convex constraint
    - Relaxed as $\sum_{i=1}^n \phi_{\lambda_0}(\lambda_i(M)) \geqslant k$, where $\phi_{\lambda_0}(\kappa) = \min\{\kappa/\lambda_0, 1\}$
- **Final convex relaxation** : minimize $\mathrm{Tr} A(X,\kappa)M$ such that $M = M^\top,\ \mathrm{diag}(M) = 1_n,\ M \geqslant 0,\ M \succcurlyeq \frac{1}{k} 1_n 1_n^\top,$ $\sum_{i=1}^n \phi_{\lambda_0}(\lambda_i(M)) \geqslant k$

Comparison with K-means

- **DIFFRAC** ($\kappa = 0$) : minimize

$$\text{Tr } \Pi_n (I_n - X(X^\top \Pi_n X)^{-1} X^\top) \Pi_n y y^\top$$

- **K-Means** : minimize (Zha et al., 2002, Bach & Jordan, 2004)

$$\min_{\mu \in \mathbb{R}^{k \times d}} \|X - y\mu\|_F^2 = \text{Tr}(I_n - y(y^\top y)^{-1} y^\top)(\Pi_n X)(\Pi_n X)^\top$$

Kernels

- The matrix $A(X, \kappa)$ can be expressed only in terms of the Gram matrix $K = XX^\top$.

$$\boxed{A(K, \kappa) = \kappa \Pi_n (\widetilde{K} + n\kappa I_n)^{-1} \Pi_n}$$

  where $\widetilde{K} = \Pi_n K \Pi_n$ is the "centered Gram matrix" of the points X.
- Additional relaxation to kernel PCA :
  1. relaxing the constraints $M \succcurlyeq \frac{1}{k} 1_n 1_n^\top$ into $M \succcurlyeq 0$
  2. relaxing $\mathrm{diag}(M) = 1_n$ into $\mathrm{Tr} M = n$
  3. removing the constraint $M \geqslant 0$ and the constraints on the row sums.
- Important constraint : $\mathrm{diag}(M) = 1_n$

Optimization by partial dualization - I

- Optimization problem :

$$\min \mathrm{Tr} AM \quad \text{such that} \quad \begin{array}{l} M = M^\top,\ M \succcurlyeq 0,\ \mathrm{Tr} M = n \\ \Phi_{\lambda_0}(M) = \sum_{i=1}^n \phi_{\lambda_0}(\lambda_i(M)) \geqslant k \\ \mathrm{diag}(M) = 1_n \\ M 1_n \leqslant (n - (k-1)\lambda_0) 1_n,\ M 1_n \geqslant \lambda_0 1_n \\ M \geqslant 0 \\ M \succcurlyeq \frac{1_n 1_n^\top}{k} \end{array} \Big| \begin{array}{l} \beta \\ \\ \\ \\ \beta \end{array}$$

- Partial dualization of constraints
  - Kept constraints lead to simple **spectral problem**

Optimization by partial dualization - II

- Lagrangian equal to $\mathrm{Tr}B(\beta)M - b(\beta)$ with

$$B(\beta) = A + \mathrm{Diag}(\beta_1) - \tfrac{1}{2}(\beta_2 - \beta_3)1^\top - \tfrac{1}{2}1(\beta_2 - \beta_3)^\top - \beta_4 + \tfrac{1}{2}\frac{\beta_5\beta_5^\top}{\beta_6}$$

$$b(\beta) = \beta_1^\top 1 - (n - (k-1)\lambda_0)\beta_2^\top 1 + \lambda_0\beta_3^\top 1 + k\beta_6/2 + \beta_5^\top 1$$

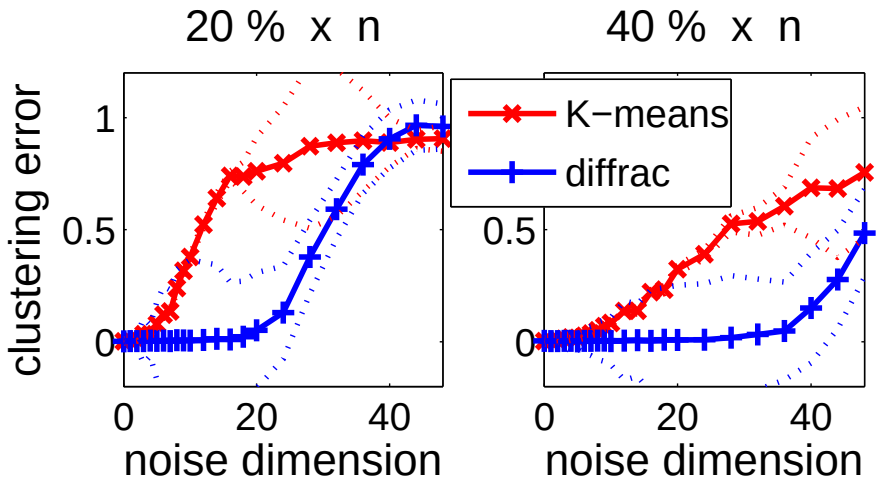
- Primal variable M, dual variables β_1, β_2, β_3, β_4, (β_5, β_6)
- **Dual problem** : $\max_{\beta} \left\{ \min_{M \succcurlyeq 0, \mathrm{Tr}M = n, \Phi_{\lambda_0}(M) \geqslant k} \mathrm{Tr}B(\beta)M - b(\beta) \right\}$
- Minimization with respect to M leads to convex non differentiable spectral function in β
- Maximization with respect to β by projected subgradient or projected gradient (after smoothing)

Computational complexity - Rounding

- Constant times the matrix-vector operation with the matrix A
- **Linear complexity** in the number n of data points.
- For linear kernels with dimension d : $O(d^2 n)$
- For general kernels : $O(n^3)$ or $O(m^2 n)$ using an incomplete Cholesky decomposition of rank m
- Rounding
  - After the convex optimization, we obtain a low-rank matrix $M \in \mathcal{C}_k$ which is pointwise nonnegative with unit diagonal
  - Spectral clustering algorithm on the matrix M (Ng & al., 2001)
  - NB : Diffrac works better than just doing spectral clustering on A or K !

Semi-supervised learning

- Equivalence matrices M allow simple inclusion of prior knowledge (Xu et al., 2004, De Bie and Cristianini, 2006)
- **"must-link"** constraints (positive constraints) : $M_{ij} = 1$
  - With a square loss $\Rightarrow$ equivalent to grouping into chuncks
- **"must-not-link"** constraints (negative constraints) : $M_{ij} = 0$



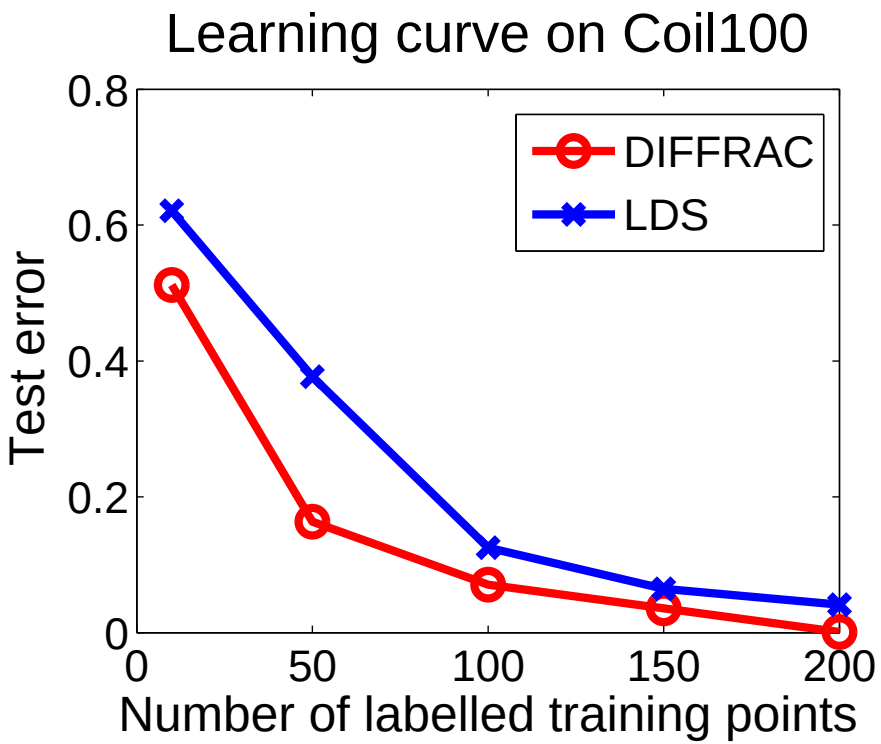
Simulations

- Clustering classification datasets
  - Performance measured by clustering error between 0 and $100(k-1)$
  - Comparison with K-means and RCA (Bar-Hillel et al., 2003)
  - Different amount of labelled data (0 to 40 %)

Dataset	K-means	Diffrac	RCA
Mnist-linear 0%	$\mathbf{5.6 \pm 0.1}$	6.0 ± 0.4	
Mnist-linear 20%	4.5 ± 0.3	3.6 ± 0.3	$\mathbf{3.0 \pm 0.2}$
Mnist-linear 40%	2.9 ± 0.3	2.2 ± 0.2	$\mathbf{1.8 \pm 0.4}$
Mnist-RBF 0%	5.6 ± 0.2	$\mathbf{4.9 \pm 0.2}$	
Mnist-RBF 20%	4.6 ± 0.0	$\mathbf{1.8 \pm 0.4}$	4.1 ± 0.2
Mnist-RBF 40%	4.9 ± 0.0	$\mathbf{0.9 \pm 0.1}$	2.9 ± 0.1
Isolet-linear 0%	$\mathbf{12.1 \pm 0.6}$	12.3 ± 0.3	
Isolet-linear 20%	10.5 ± 0.2	$\mathbf{7.8 \pm 0.8}$	9.5 ± 0.4
Isolet-linear 40%	9.2 ± 0.5	$\mathbf{3.7 \pm 0.2}$	7.0 ± 0.4
Isolet-RBF 0%	11.4 ± 0.4	$\mathbf{11.0 \pm 0.3}$	
Isolet-RBF 20%	10.6 ± 0.0	$\mathbf{7.5 \pm 0.5}$	7.8 ± 0.5
Isolet-RBF 40%	10.0 ± 0.0	$\mathbf{3.7 \pm 1.0}$	6.9 ± 0.6

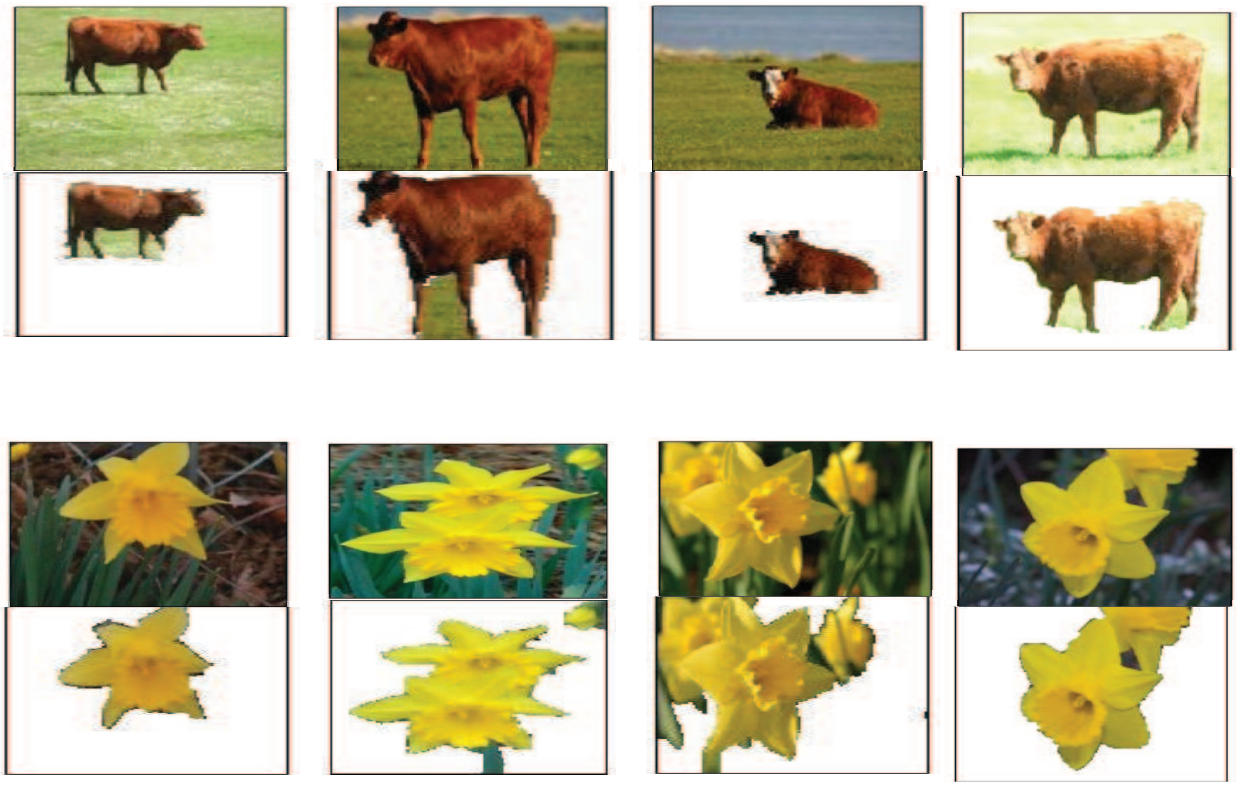
Simulations

- Semi-supervised classification
  - Diffrac "works" with any amount of supervision
  - Comparison with LDS (Chapelle & Zien, 2004)



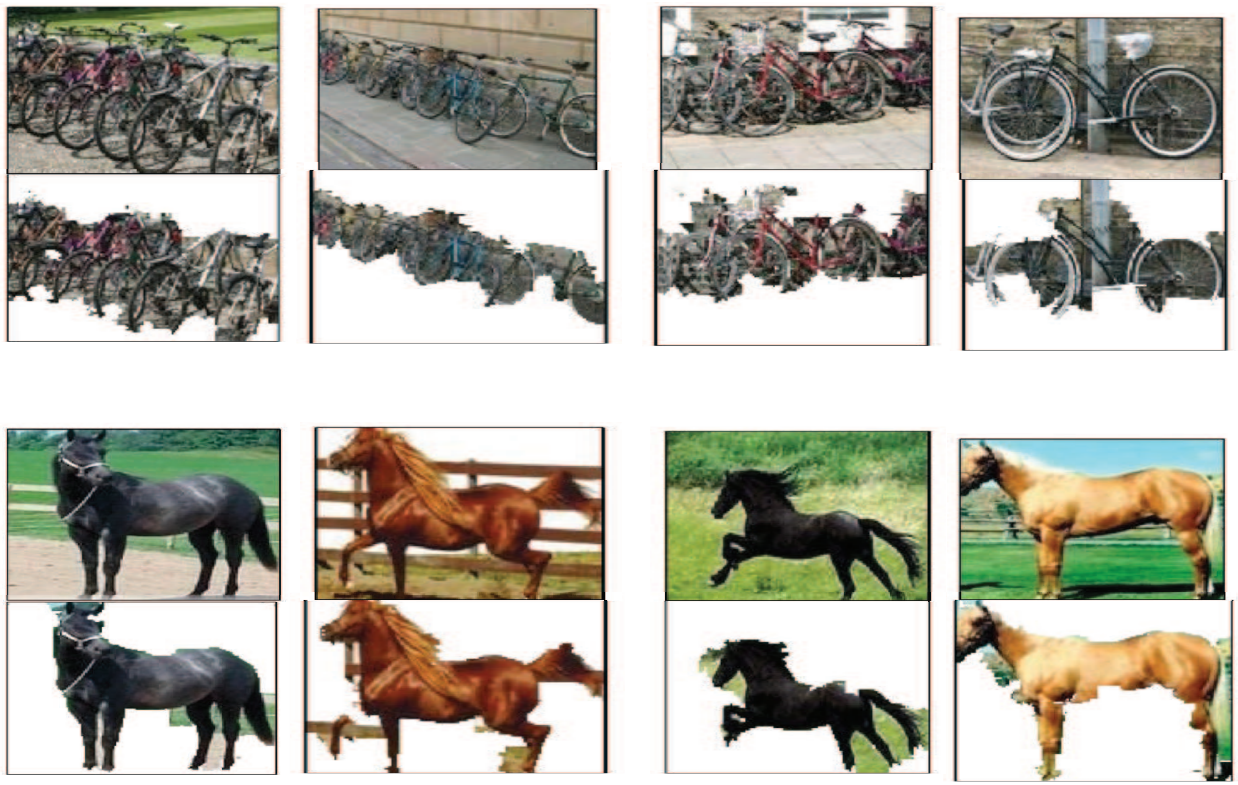
Extension to images co-segmentation (Joulin et al., 2010)

Natural images



Extension to images co-segmentation (Joulin et al., 2010)

Cycles and horses



Outline

1 Discriminative clustering

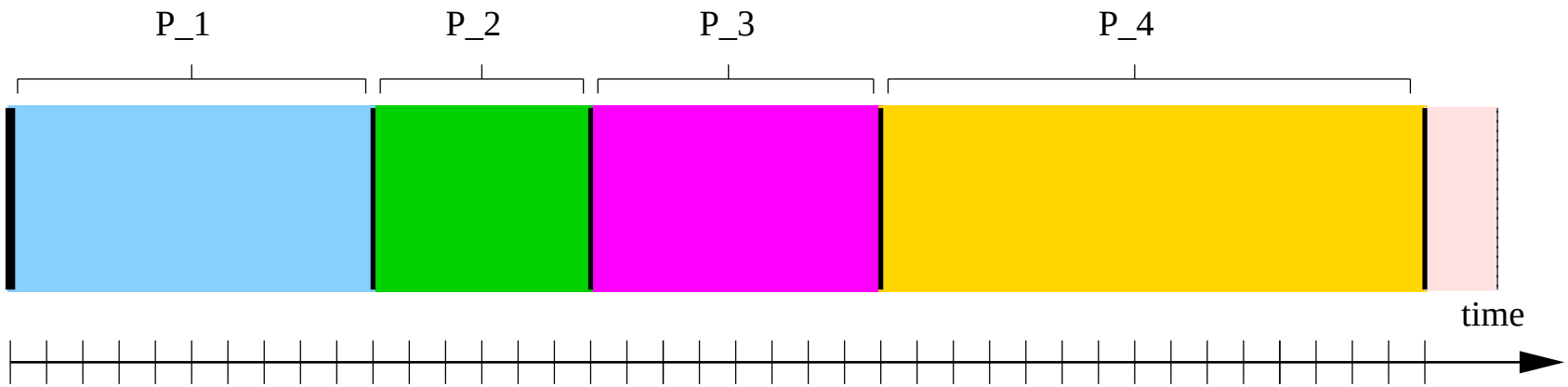
2 Temporal Segmentation

Temporal segmentation (clustering with temporal consistency)

Change-in-mean model

Time series of independent r.v. $\{Y_t\}_{t=1,\dots,n}$ such that

$$Y_t \overset{\mathcal{D}}{\sim} \mathcal{N}(\mu_k^\star, \sigma^2)\ , \quad t_{k-1}^\star + 1 \leq t \leq t_k^\star, \quad k = 1, \dots, K^\star + 1\ , \qquad (1)$$

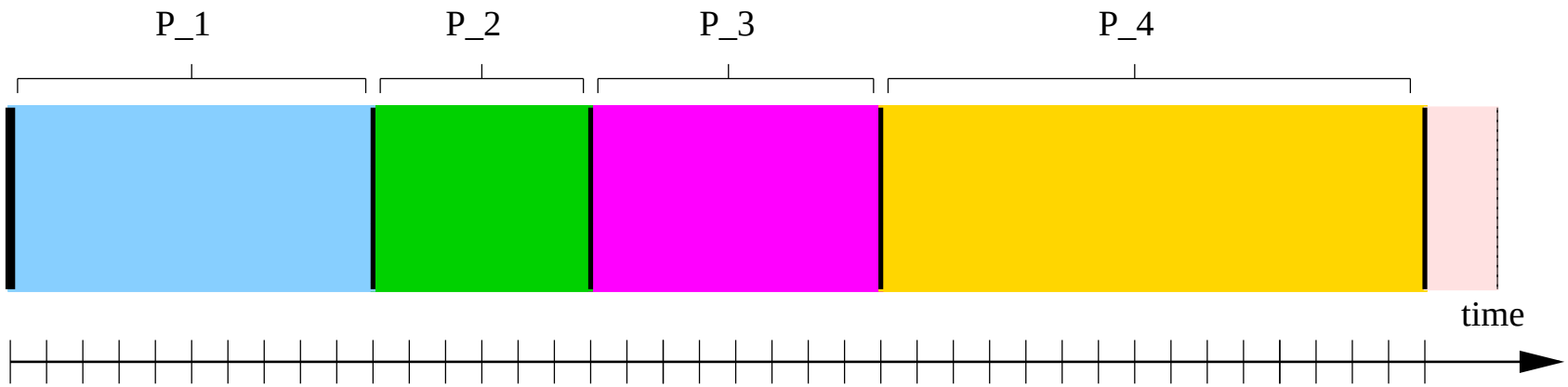


Temporal segmentation

Change-in-mean-element model

Time series of independent r.v. $\{Y_t\}_{t=1,\ldots,n}$ such that

$$\mathbb{E}[k(Y_t,\cdot)] = \mu_k^\star\ , \quad t_{k-1}^\star + 1 \leq t \leq t_k^\star, \quad k = 1,\ldots,K^\star+1\ .$$



Temporal segmentation with kernels

Classical least-squares formulation

$$\underset{t_1,\ldots,t_{K^\star}}{\text{Minimize}} \sum_{k=1}^{K^\star+1} \sum_{t=t_{k-1}+1}^{t_k} (Y_t - \overline{Y}(t_{k-1}, t_k))^2$$

Kernel-based version in $\mathcal{H}$

$$\underset{t_1,\ldots,t_{K^\star}}{\text{Minimize}} \sum_{k=1}^{K^\star+1} \sum_{t=t_{k-1}+1}^{t_k} \left\| k(Y_t, \cdot) - \hat{\mu}_{[t_{k-1}:t_k]} \right\|_{\mathcal{H}}^2$$

Massaging the objective function

Intra-segment scatter

$$\underset{t_1,\ldots,t_{K^\star}}{\text{Minimize}} \sum_{k=1}^{K-1} \hat{V}(Y_{t_k+1},\ldots,Y_{t_{k+1}})$$

with $\hat{V}(Y_{t+1},\ldots,Y_{t+s}) = \left\| k(Y_t,\cdot) - \hat{\mu}_{[t+1:t+s]} \right\|_{\mathcal{H}}^2$

Forward-backwrad recursions

Forward recursions

$$
\begin{aligned}
I_k(t) &= \operatorname*{Min}_{t_1,\ldots,t_{k-1};t_k=t} \sum_{k=1}^{K-1} \hat{V}(Y_{t_k+1},\ldots,Y_{t_{k+1}}) \\
&= \operatorname*{Min}_{t_{k-1};t_k=t} \operatorname*{Min}_{t_1,\ldots,t_{k-2}} \sum_{k=1}^{K-1} \hat{V}(Y_{t_k+1},\ldots,Y_{t_{k+1}}) \\
&= \operatorname*{Min}_{t_{k-1}} (I_{k-1}(t_{k-1}) + \hat{V}(Y_{t_{k-1}},\ldots,Y_t)) \ .
\end{aligned}
$$

Dynamic programming

Dynamic programming algorithm working on submatrices of the Gram matrix, leading to a time-complexity of $O(Kn^2)$.

Kernel-based Methods for Unsupervised Learning

LEAR project-team, INRIA

Zaid Harchaoui

Grenoble, July 30th 2010

Outline

1 Introduction

2 Homogeneity testing

3 Change-point Analysis

Outline

1 Introduction

2 Homogeneity testing

3 Change-point Analysis

Kernel methods

Machine Learning methods taking $\mathbf{K} = [k(X_i, X_j)]_{i,j=1,\ldots,n}$ (Gram matrix as input for processing a sample $\{X_1, \ldots, X_n\}$, where $k(x, y)$ is a similarity measure between x and y defining a positive definite kernel.

Strengths of Kernel Methods

- Minimal assumptions on data types (vectors, strings, trees, graphs, etc.)
- Interpretation of $k(x, y)$ as a dot product $k(x, y) = \langle \phi(x), \phi(y) \rangle_{\mathcal{H}}$ in a reproducing kernel Hilbert space $\mathcal{H}$ where the observations are mapped *via* $[\phi : \mathcal{X} \to \mathcal{H}]$ (feature map)

Mean element and covariance operator

Population mean element and covariance operator

Population mean element μ and population covariance operator Σ of $X \sim \mathbb{P}$

$$\langle \mu, f \rangle_{\mathcal{H}} \stackrel{\text{def}}{=} \mathbb{E}[f(X)] \,, \quad \forall f \in \mathcal{H}$$

$$\langle f, \Sigma g \rangle_{\mathcal{H}} \stackrel{\text{def}}{=} \mathrm{Cov}[f(X), g(X)] \,, \quad \forall f, g \in \mathcal{H}$$

Empirical mean element and covariance operator

Empirical mean element $\hat{\mu}$ and empirical covariance operator $\hat{\Sigma}$ of $X_1, \ldots, X_m \sim \mathbb{P}$

$$\langle \hat{\mu}, f \rangle_{\mathcal{H}} \stackrel{\text{def}}{=} \frac{1}{m} \sum_{\ell=1}^{m} f(X_\ell) \,, \quad \forall f \in \mathcal{H}$$

$$\left\langle f, \hat{\Sigma} g \right\rangle_{\mathcal{H}} \stackrel{\text{def}}{=} \frac{1}{m} \sum_{\ell=1}^{m} \{f(X_\ell) - \langle \hat{\mu}, f \rangle_{\mathcal{H}}\}\{f(X_\ell) - \langle \hat{\mu}, g \rangle_{\mathcal{H}}\} \quad \forall f, g \in \mathcal{H}$$

Test for homogeneity

Homogeneity of two samples

- Two samples $X_1^{(1)}, \ldots, X_{n_1}^{(1)} \sim \mathbb{P}^{(1)}$ and $X_1^{(2)}, \ldots, X_{n_2}^{(2)} \sim \mathbb{P}^{(2)}$ independent
- Problem : decide between

$$\mathbf{H}_0 : \quad \mathbb{P}^{(1)} = \mathbb{P}^{(2)}$$
$$\mathbf{H}_A : \quad \mathbb{P}^{(1)} \neq \mathbb{P}^{(2)}$$

Test statistic

Empirical mean elemnts $\hat{\mu}_1$ and $\hat{\mu}_2$, and empirical covariance operators $\hat{\Sigma}_1$ and $\hat{\Sigma}_2$ resp. $\{X_1^{(1)}, \ldots, X_{n_1}^{(1)}\}$ et $\{X_1^{(2)}, \ldots, X_{n_2}^{(2)}\}$

$$\{X_1^{(1)}, \ldots, X_{n_1}^{(1)}\} \hookrightarrow (\hat{\mu}_1, \hat{\Sigma}_1) \quad \text{and} \quad \{X_1^{(2)}, \ldots, X_{n_2}^{(2)}\} \hookrightarrow (\hat{\mu}_2, \hat{\Sigma}_2) \, .$$

Regularized Fisher ratio

$$\begin{aligned} &\mathrm{KFDR}_{n_1,n_2;\gamma}(X_1^{(1)}, \ldots, X_{n_1}^{(1)}; X_1^{(2)}, \ldots, X_{n_2}^{(2)}) \\ &\stackrel{\text{def}}{=} \frac{n_1 n_2}{n_1 + n_2} \left\| \Big(\underbrace{\frac{n_1}{n}\hat{\Sigma}_1 + \frac{n_2}{n}\hat{\Sigma}_2}_{\hat{\Sigma}_W} + \gamma \mathrm{I} \Big)^{-1/2} (\hat{\mu}_2 - \hat{\mu}_1) \right\|_{\mathcal{H}}^2 . \end{aligned}$$

Hotelling's T^2 : homogeneity of two normal probability distributions with different means and unknown covariance matrices

$$\frac{n_1 n_2}{n_1 + n_2} \left\| \Big(\frac{n_1}{n}\hat{\Sigma}_1 + \frac{n_2}{n}\hat{\Sigma}_2 \Big)^{-1/2} (\hat{\mu}_2 - \hat{\mu}_1) \right\|_{\mathfrak{R}^d}^2$$

Large-sample distribution under H_0 : regime $\gamma_n \equiv \gamma$

Proposition

Assume the kernel is bounded and that for $a = 1, 2$ the eigenvalues $\{\lambda_p(\Sigma_a)\}_{p\geq 1}$ satisfy $\sum_{p=1}^{\infty} \lambda_p^{1/2}(\Sigma_a) < \infty$. Assume also that $\mathbb{P}_1$ and $\mathbb{P}_2$ are equal *i.e.* $\mathbb{P}_1 = \mathbb{P}_2 = \mathbb{P}$, and that $\gamma_n \equiv \gamma > 0$. Then,

$$\frac{\mathrm{KFDR}_{n_1,n_2;\gamma} - d_{1,n_1,n_2;\gamma}(\hat{\Sigma}^W_{n_1,n_2})}{\sqrt{2}\, d_{2,n_1,n_2;\gamma}(\hat{\Sigma}^W_{n_1,n_2})} \xrightarrow{\mathcal{D}} \frac{1}{\sqrt{2}\, d_{2,n_1,n_2;\gamma}(\Sigma_W)} \sum_{p=1}^{\infty} \frac{\lambda_p(\Sigma_W)}{\lambda_p(\Sigma_W) + \gamma} (\underbrace{Z_p^2}_{\chi_1^2} - 1) \,,$$

Remarks

$$d_{1,n_1,n_2;\gamma}(\hat{\Sigma}_W) \stackrel{\mathrm{def}}{=} \mathrm{Tr}((\hat{\Sigma}_W + \gamma I)^{-1} \hat{\Sigma}_W) \qquad \text{recentering}$$

$$d_{2,n_1,n_2;\gamma}(\hat{\Sigma}_W) \stackrel{\mathrm{def}}{=} [\mathrm{Tr}((\hat{\Sigma}_W + \gamma I)^{-2} \hat{\Sigma}_W^2)]^{1/2} \qquad \text{renormalization}$$

Large-sample distribution under H_0 : regime $\gamma_n \to 0$

Proposition

Assume the kernel is bounded and that for $a = 1, 2$ the eigenvalues $\{\lambda_p(\Sigma_a)\}_{p\geq 1}$ satisfy $\sum_{p=1}^{\infty} \lambda_p^{1/2}(\Sigma_a) < \infty$. Assume in addition that $\mathbb{P}_1 = \mathbb{P}_2 = \mathbb{P}$, and that $\{\gamma_n\}$ is such that

$$\gamma_n + \frac{d_{1,n_1,n_2;\gamma}(\Sigma_W)}{d_{2,n_1,n_2;\gamma}(\Sigma_W)} \gamma_n^{-1} n^{-1/2} \to 0 \, .$$

Then,

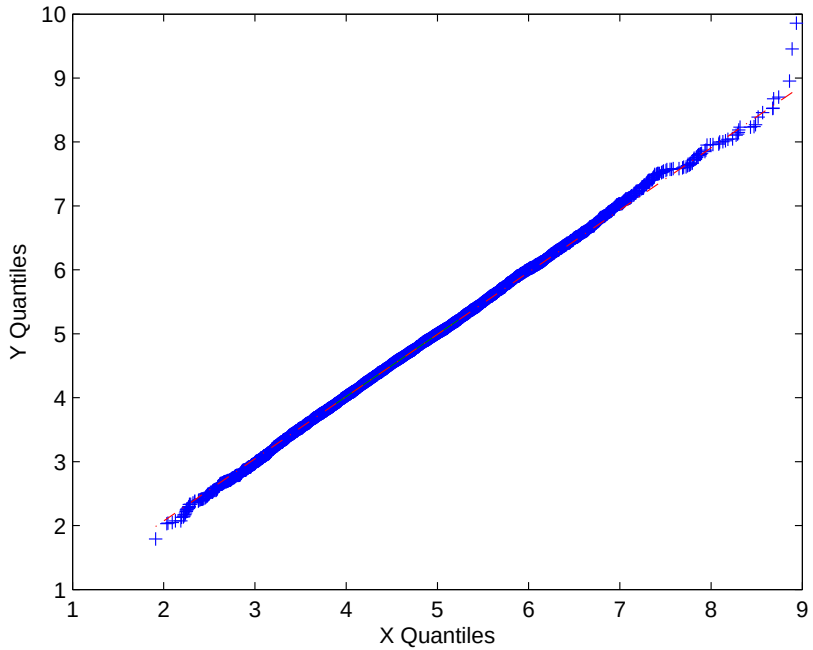
$$\frac{\mathsf{KFDR}_{n_1,n_2;\gamma_n} - d_{1,n_1,n_2;\gamma_n}(\hat{\Sigma}^W_{n_1,n_2})}{\sqrt{2}\, d_{2,n_1,n_2;\gamma_n}(\hat{\Sigma}^W_{n_1,n_2})} \xrightarrow{\mathcal{D}} \mathcal{N}(0,1) \, .$$

Remarks

- Typical situation $\gamma_n \to 0$ slower than $1/\sqrt{n}$
- Case $\lambda_p = p^{-2m}$: $d_{1,n_1,n_2;\gamma_n} \sim \gamma_n^{-1/2m}$ et $d_{2,n_1,n_2;\gamma_n} \sim \gamma_n^{-1/4m}$

Distribution under H_0

- Total sample size $n_1 + n_2 = 500$, Gaussian RBF kernel with $\sigma = 1$, $\mathbb{P}^{(1)} = \mathbb{P}^{(2)}$ normal probability distributions



Consistency in power

Proposition

Assume the kernel is bounded and that for $a = 1, 2$ the eigenvalues $\{\lambda_p(\Sigma_a)\}_{p\geq 1}$ satisfy $\sum_{p=1}^{\infty} \lambda_p^{1/2}(\Sigma_a) < \infty$, and that the RKHS $\mathcal{H}$ is dense in $L^2(\mathbb{P})$ for all $\mathbb{P}$. Let $\mathbb{P}_1$ and $\mathbb{P}_2$ two probability distributions such that $\mathbb{P}_2 \neq \mathbb{P}_1$. In both regimes ($\gamma_n \equiv \gamma$ and $\gamma_n \to 0$), for all $0 < \alpha < 1$

$$\mathbb{P}_{\mathbf{H}_A}\left(\frac{\mathrm{KFDR}_{n_1,n_2;\gamma_n} - d_{1,n_1,n_2;\gamma}(\hat{\Sigma}^W_{n_1,n_2})}{\sqrt{2}\, d_{2,n_1,n_2;\gamma}(\hat{\Sigma}^W_{n_1,n_2})} > c_{1-\alpha}\right) \to 1\,. \qquad (1)$$

Remarks

Universal density of the RKHS satisfied for translation-invariant kernels $k(x, y) = k(x - y)$ such as the Gaussian RBF kernel (Steinwart, 2006 ; Sriperumbudur et al., 2008).

Consistency against local alternatives

- Framework of local alternatives

$$\begin{aligned} \mathbf{H}_0 : &\quad \mathbb{P}_1 = \mathbb{P}_2^n \\ \mathbf{H}_A : &\quad \mathbb{P}_1 \neq \mathbb{P}_2^n \end{aligned}$$

where $\mathbb{P}_1$ and $\mathbb{P}_2^n$ get closer as $n \to \infty$, meaning that the χ^2-divergence

$$D_{\chi^2}(\mathbb{P}_1, \mathbb{P}_2^n) \leq \eta_n \ , \quad \text{as } n \to \infty \ .$$

Illustration : uniforme vs. uniform+high-frequency contamination with spline kernels

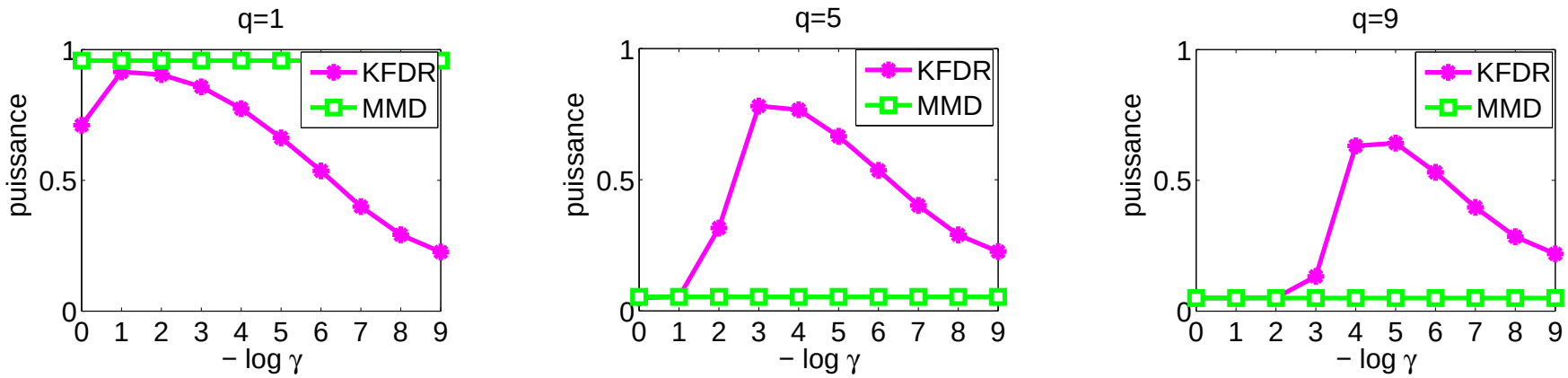


Figure: Comparison of change in power of KFDA versus MMD as $\gamma = 1, 10^{-1}, \ldots, 10^{-9}$, for local alternatives spanned by the q-ieme component (from left to right) with $q = 1, 5, 9$.

Computational aspects

Computation

$$\left\|(\hat{\Sigma}_W + \gamma_n \mathrm{I})^{-1/2}(\hat{\mu}_2 - \hat{\mu}_1)\right\|_{\mathcal{H}}^2$$
$$= \gamma^{-1}\left\{\mathbf{m}_n^T \mathbf{K}_n \mathbf{m}_n - n^{-1}\mathbf{m}_n^T \mathbf{K}_n \mathbf{N}_n (\gamma \mathrm{I} + n^{-1}\mathbf{N}_n \mathbf{K}_n \mathbf{N}_n)^{-1}\mathbf{N}_n \mathbf{K}_n \mathbf{m}_n\right\} .$$

$\mathbf{K}_n = [k(x_i, x_j)]_{i,j=1,\ldots,n}$ is the Gram matrix, $\mathbf{N}_n$ is that intra-class re-centering matrix (each block re-centers each sample), and $\mathbf{m}_n = (\mathbf{m}_{n,i})_{1 \leq i \leq n}$ stand for the "vector of mean difference" with $\mathbf{m}_{n,i} = -n_1^{-1}$ pour $i = 1, \ldots, n_1$ et $\mathbf{m}_{n,i} = n_2^{-1}$ for $i = n_1 + 1, \ldots, n_1 + n_2$

Computational complexity

$O((n_1 + n_2)^2)$ is space and $O((n_1 + n_2)^3)$ in time.

Application : speaker verification

- 8 speakers from the NIST evaluation 2004
- descriptors : MFCC

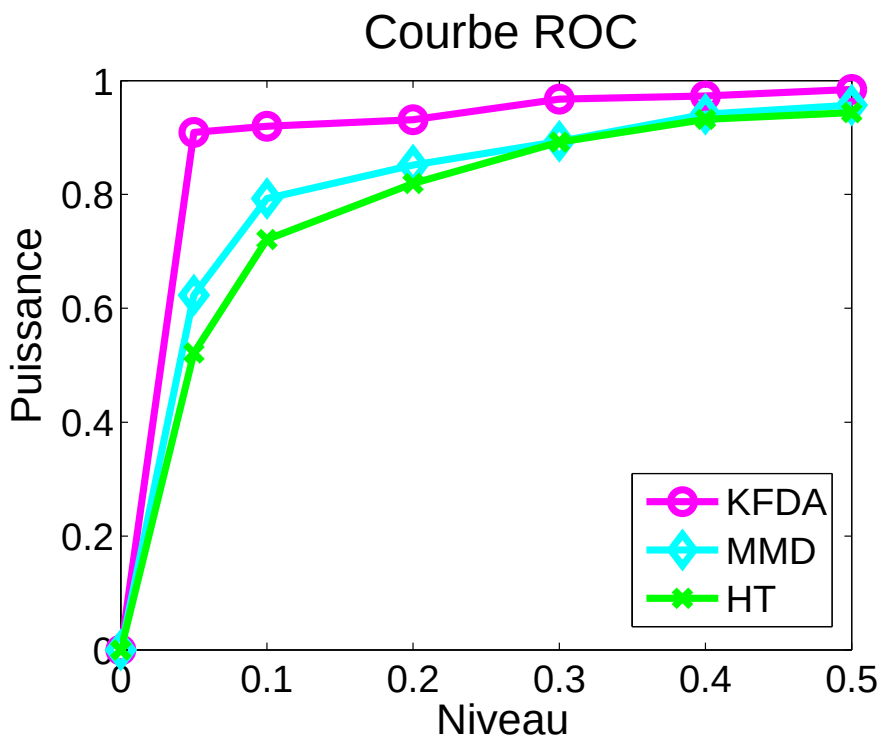


Figure: Comparison ROC curves for speaker verification

Application : audio segmentation

"Grand echiquier" TV-shows archives

- Semantic segmentation (coarse segmentation) : applause/film/music/*interview*
- Speaker segmentation (fine segmentation) : Coluche/J. Chancel/F.-R. Duchable/etc.

	Nb. of sections	Mean duration (sec.)
applause	84	14
film	29	155
music	38	194
speech	188	70
spk turns	962	6

Table: Data description

Experiments in audio segmentation

Experiences

- sliding-window along the signal
- super-descriptors cbuilt from cepstral coefficients
- comparison with unsupervised approaches MMD (Gretton et al., 2004), KCD (Desobry et al., 2005), and supervised HMM (Rabiner et al., 2007)

	Semantic seg.		Spk seg.	
	Precision	Recall	Precision	Recall
KFDR	0.72	0.63	0.89	0.90
MMD	0.71	0.58	0.76	0.73
KCD	0.65	0.63	0.78	0.74
HMM	0.73	0.65	0.93	0.96

Table: Precision and recall

Outline

1 Introduction

2 Homogeneity testing

3 Change-point Analysis

Change-point Analysis

Assumption

Time series $X_1, \ldots, X_n$ of independent observations

Change-point Problem

detection 1) Decide between

$$\mathbf{H}_0 : \quad \mathbb{P}_{X_1} = \cdots = \mathbb{P}_{X_\theta} = \cdots = \mathbb{P}_{X_n}$$

$$\mathbf{H}_A : \quad \text{there exists } 1 < k^\star < n \text{ such that}$$

$$\mathbb{P}_{X_1} = \cdots = \mathbb{P}_{X_{k^\star}} \neq \mathbb{P}_{X_{k^\star+1}} = \cdots = \mathbb{P}_{X_n} \ .$$

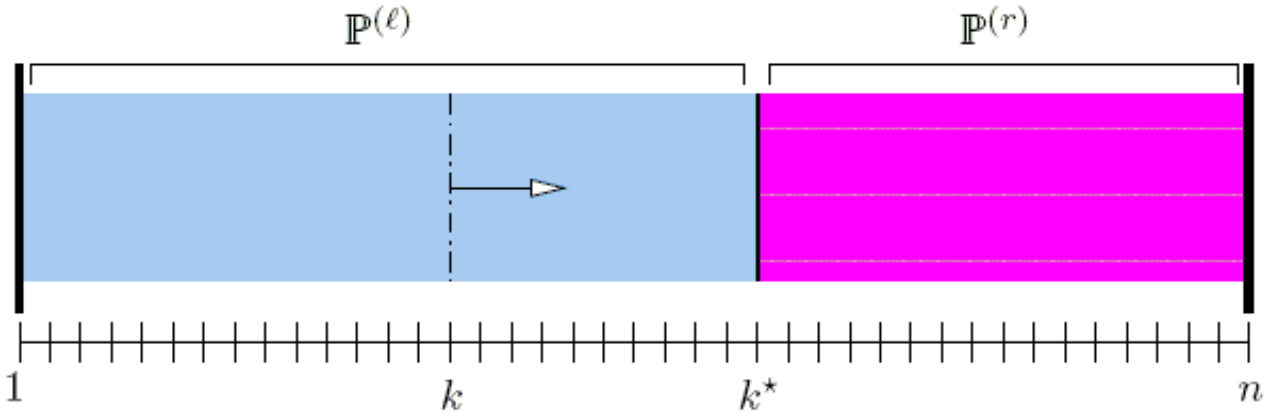
estimation 2) Estimate $k^\star$ from the sample $\{X_1, \ldots, X_n\}$ if $\mathbf{H}_A$ is true .

Change-point Analysis = Change-point Detection + Estimation

Running Maximum Strategy

Running Maximum Strategy for change-point detection

run along the series of observations $X_1, \ldots, X_n$, scanning all change-point candidates $k \in]1, n[$, in order to catch the true change-point instant $k^\star$, for which the segment *before change* and the segment *after change* have minimum homogeneity



Building block for the test statistic : finite-dimensional case

- Time series $X_1, \ldots, X_n \in \mathbf{R}^d$ of independent observations
- For any interval $[i, j] \subset \{2, \ldots, n-1\}$, define resp. the mean vector $\hat{\mu}_{i:j}$ and the covariance matrix $\hat{\Sigma}_{i:j}$.
- For any instant $k \in \{2, \ldots, n-1\}$,

$$T_{n,k}(X_1, \ldots, X_n) \stackrel{\text{def}}{=} \frac{k(n-k)}{n} \left\| \underbrace{\left(\frac{k}{n} \hat{\Sigma}_{1:k} + \frac{n-k}{n} \hat{\Sigma}_{k+1:n} \right)}_{\hat{\Sigma}^W_{n,k}}^{-1/2} (\hat{\mu}_{k+1:n} - \hat{\mu}_{1:k}) \right\|_2^2 .$$

- Null distribution

$$\max_{a_n < k < b_n} T_{n,k}(X_1, \ldots, X_n) \xrightarrow{\mathcal{D}} \max_{u < t < v} \frac{\sum_{p=1}^d \mathbf{B}_p^2(t)}{t(1-t)}$$

- Consistency in Power (see James, James, Siegmund, 1987)

Building block for the test statistic : kernelized case

- Time series $X_1, \ldots, X_n$ of independent observations
- For any interval $[i, j] \subset \{2, \ldots, n-1\}$, define for all $f, g \in \mathcal{H}$

$$\langle \hat{\mu}_{i:j}, f \rangle_{\mathcal{H}} \stackrel{\text{def}}{=} \frac{1}{j-i+1} \sum_{\ell=i}^{j} f(X_\ell)$$

$$\left\langle f, \hat{\Sigma}_{i:j} g \right\rangle_{\mathcal{H}} \stackrel{\text{def}}{=} \frac{1}{j-i+1} \sum_{\ell=i}^{j} \{f(X_\ell) - \langle \hat{\mu}_{i:j}, f \rangle_{\mathcal{H}}\}\{g(X_\ell) - \langle \hat{\mu}_{i:j}, g \rangle_{\mathcal{H}}\}$$

- For any instant $k \in \{2, \ldots, n-1\}$,

$\mathrm{KFDR}_{n,k;\gamma}(X_1, \ldots, X_n)$ (maximum) Kernel Fisher Discriminant Ratio

$$\stackrel{\text{def}}{=} \frac{k(n-k)}{n} \left\| \left(\underbrace{\frac{k}{n} \hat{\Sigma}_{1:k} + \frac{n-k}{n} \hat{\Sigma}_{k+1:n}}_{\hat{\Sigma}^W_{n,k}} + \gamma \mathrm{I} \right)^{-1/2} (\hat{\mu}_{k+1:n} - \hat{\mu}_{1:k}) \right\|^2_{\mathcal{H}}.$$

Kernel Change-point Analysis (KCpA)

KCpA Test statistic

$$T_{n;\gamma_n} = \max_{a_n<k<b_n} \frac{\mathsf{KFDR}_{n,k;\gamma_n} - d_{1,n,k;\gamma_n}(\hat{\Sigma}^W_{n,k})}{\sqrt{2}\, d_{2,n,k;\gamma_n}(\hat{\Sigma}^W_{n,k})}$$

with

$$d_{1,n,k;\gamma}(\hat{\Sigma}^W_{n,k}) \stackrel{\text{def}}{=} \mathrm{Tr}((\hat{\Sigma}^W_{n,k} + \gamma I)^{-1}\hat{\Sigma}^W_{n,k}) \qquad \text{recentering}$$

$$d_{2,n,k;\gamma}(\hat{\Sigma}^W_{n,k}) \stackrel{\text{def}}{=} [\mathrm{Tr}((\hat{\Sigma}^W_{n,k} + \gamma I)^{-2}(\hat{\Sigma}^W_{n,k})^2)]^{1/2} \qquad \text{rescaling}$$

Change-point Detection

$T_{n;\gamma_n} \leq t_{1-\alpha}$ no change occured

$T_{n;\gamma_n} > t_{1-\alpha}$ a change occured

with $t_{1-\alpha}$ the α-significance threshold.

Change-point Estimation

$$\hat{k}_n = \mathsf{argmax} \frac{\mathsf{KFDR}_{n,k;\gamma_n} - d_{1,n,k;\gamma_n}(\hat{\Sigma}^W_{n,k})}{\sqrt{2}\, d_{2,n,k;\gamma_n}(\hat{\Sigma}^W_{n,k})}$$

if a change has indeed occured ($\mathbf{H}_A$), and where $\hat{k}_n$ is the change-point estimator.

Limiting distribution under $H_0 : \gamma_n \to 0$ regime

Proposition

Assume that the kernel is bounded and that for $a = 1, 2$ the eigenvalues $\{\lambda_p(\Sigma_a)\}_{p\geq 1}$ of the covariance operatorΣ satisfy $\sum_{p=1}^{\infty} \lambda_p^{1/2}(\Sigma_a) < \infty$. Assume in addition $\mathbf{H}_0$, *i.e.* $\mathbb{P}_{X_i} = \mathbb{P}$ for all $1 \leq i \leq n$, and that $\{\gamma_n\}_{n\geq 1}$ is such that

$$\gamma_n + \frac{d_{1,n;\gamma_n}(\Sigma)}{d_{2,n;\gamma_n}(\Sigma)} \gamma_n^{-1} n^{-1/2} \to 0 \ ,$$

Then,

$$\max_{a_n < k < b_n} T_{n;\gamma_n}(k) \xrightarrow{\mathcal{D}} \sup_{u<t<v} \frac{\mathbf{B}(t)}{\sqrt{t(1-t)}} \ ,$$

where $a_n/n \to u > 0$ and $b_n/n \to v < 1$ as $n \to \infty$, and $\{\mathbf{B}_p(t)\}_t$ is a brownian bridge.

Remark

- Typically : $\gamma_n \to 0$ slower than $1/\sqrt{n}$
- Case $\lambda_p = p^{-2m}$: $d_{1,n_1,n_2;\gamma_n} \sim \gamma_n^{-1/2m}$ et $d_{2,n_1,n_2;\gamma_n} \sim \gamma_n^{-1/4m}$

Consistency in power

Proposition

Assume that the kernel is bounded and that for $a = 1, 2$ the eigenvalues $\{\lambda_p(\Sigma_a)\}_{p\geq 1}$ satisfy $\sum_{p=1}^{\infty} \lambda_p^{1/2}(\Sigma_a) < \infty$, and that the RKHS is dense in $L^2(\mathbb{P})$ for all $\mathbb{P}$, and $\mathbf{H}_A$, *i.e.* $u < \theta^\star < v$ with $u > 0$ and $v < 1$ such that $\mathbb{P}_{X_{\lfloor n\theta^\star \rfloor}} \neq \mathbb{P}_{X_{\lfloor n\theta^\star \rfloor + 1}}$ for all $1 \leq i \leq n$. Then, in either regularization scheme, for all $0 < \alpha < 1$,

$$\mathbb{P}_{\mathbf{H}_A}\left(\max_{a_n < k < b_n} \frac{\mathrm{KFDR}_{n,k;\gamma} - d_{1,n,k;\gamma}(\hat{\Sigma}^W_{n,k})}{\sqrt{2}\, d_{2,n,k;\gamma}(\hat{\Sigma}^W_{n,k})} > t_{1-\alpha} \right) \to 1\,, \quad \text{as } n \to \infty\,, \qquad (2)$$

where $a_n/n \to u > 0$ and $b_n/n \to v < 1$ as $n \to \infty$.

Remark

Universal density of RKHS satisfied for most translation-invariant kernels $k(x, y) = k(x - y)$, such as the gaussian kernel (Steinwart, 2006 ; Sriperumbudur et al., 2008).

Mental task segmentation : comparison with supervised methods

Dataset

- Data : 3 normal subjects during 4 non-feedback sessions
- 3 tasks : imagination of repetitive self-paced left hand movements or right hand movements, and generation of words beginning with the same random letter
- Features : based on Power Spectral Density

Experimental results

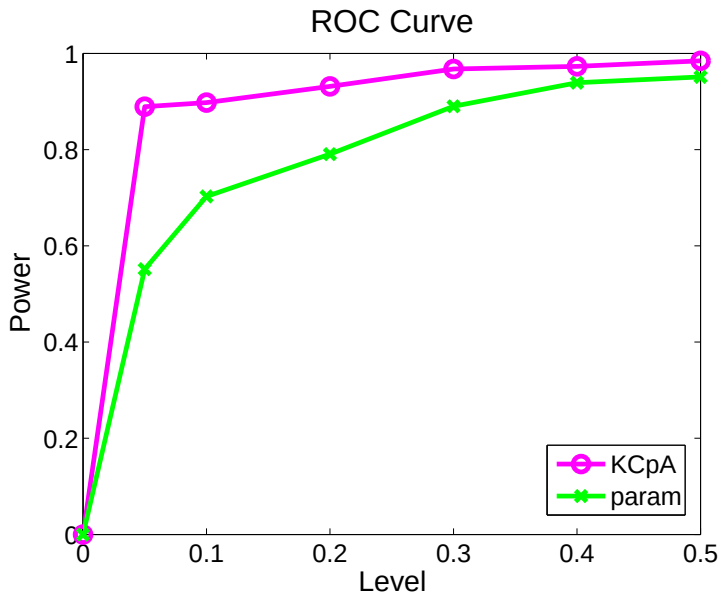
	Subject 1	Subject 2	Subject 3
KCpA	79%	74%	61%
SVM	76%	69%	60%

Mental task segmentation : comparison with unsupervised methods

Dataset

- Data : 3 normal subjects during 4 non-feedback sessions
- 3 tasks : imagination of repetitive self-paced left hand movements or right hand movements, and generation of words beginning with the same random letter
- Features : based on Power Spectral Density

Experimental results



Conclusion

Kernel learning and regularization

- Extension of mean element/covariance operator analysis to varying-kernel/multiple kernel settings
- Importance of regularization in unsupervised learning (see discriminative clustering and detection problems)

Computational efficiency

- efficient large-scale versions of kernel-based unsupervised learning algorithms
- low-rank approximation suited for particular unsupervised learning tasks