Kernel PCA and related methods

Johan Suykens

KU Leuven, ESAT-STADIUS
Kasteelpark Arenberg 10
B-3001 Leuven (Heverlee), Belgium
Email: johan.suykens@esat.kuleuven.be
http://www.esat.kuleuven.be/stadius

Lecture 9

Contents

Classical PCA formulation (1)

- Given data $\{x_k\}_{k=1}^N$ with $x_k \in \mathbb{R}^n$ (zero mean)

- Find projected variables $w^T x_k$ with maximal variance

$$\max_w \mathrm{Var}(w^T x) \quad = \quad \mathrm{Cov}(w^T x, w^T x) \simeq \frac{1}{N} \sum_{k=1}^{N} (w^T x_k)^2 = w^T C\, w$$

  where $C = (1/N) \sum_{k=1}^N x_k x_k^T$. Take constraint $w^T w = 1$.

- Constrained optimization: Lagrangian $\mathcal{L}(w; \lambda) = \frac{1}{2} w^T C w - \lambda (w^T w - 1)$ with Lagrange multiplier λ.

- Eigenvalue problem

$$C w = \lambda w$$

  with $C = C^T \geq 0$, obtained from $\partial \mathcal{L} / \partial w = 0$, $\partial \mathcal{L} / \partial \lambda = 0$.

Classical PCA formulation (2)

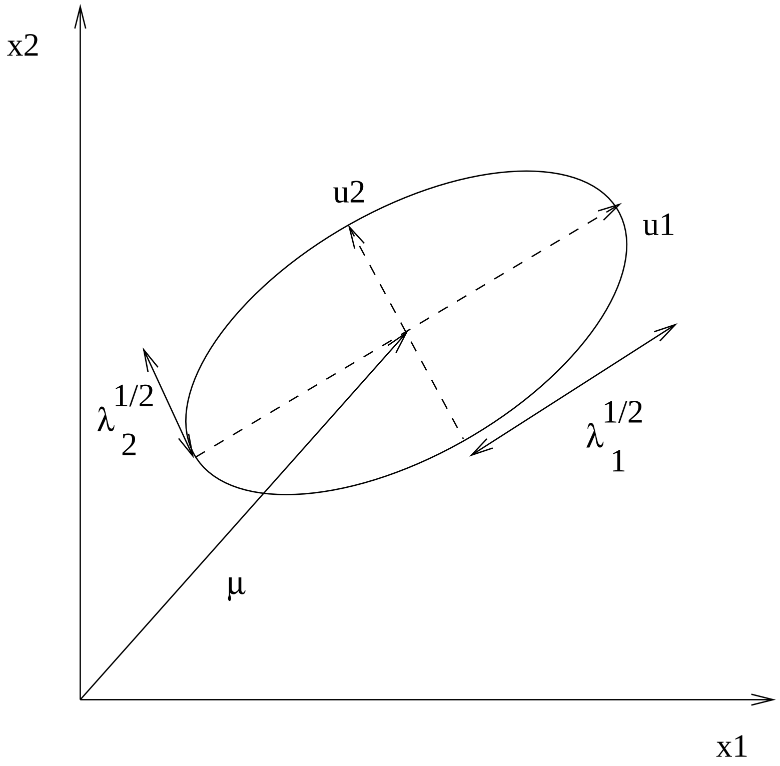


Illustration of an eigenvalue decomposition

$$Cu = \lambda u$$

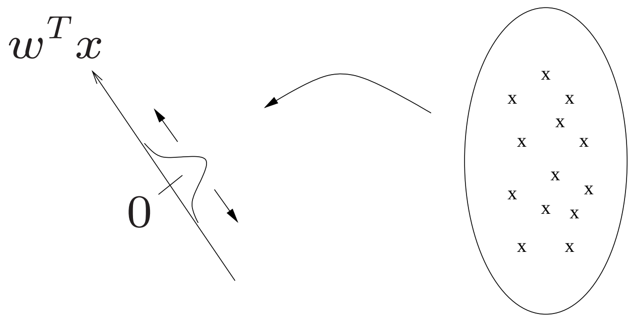
PCA analysis as a one-class modelling problem (1)

- One-class with target value zero:

$$\max_w \sum_{k=1}^{N} (0 - w^T x_k)^2$$

- Score variables: $z = w^T x$

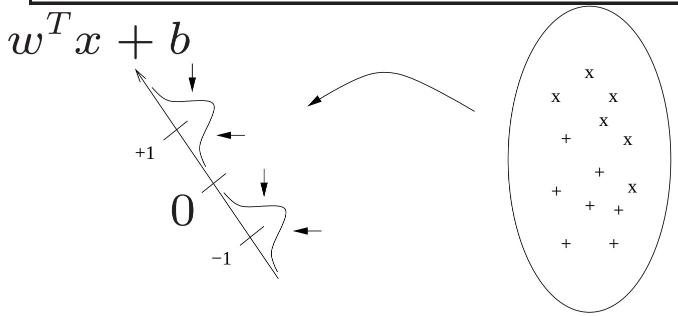
- Illustration:



Target space *Input space*

PCA analysis as a one-class modelling problem (2)

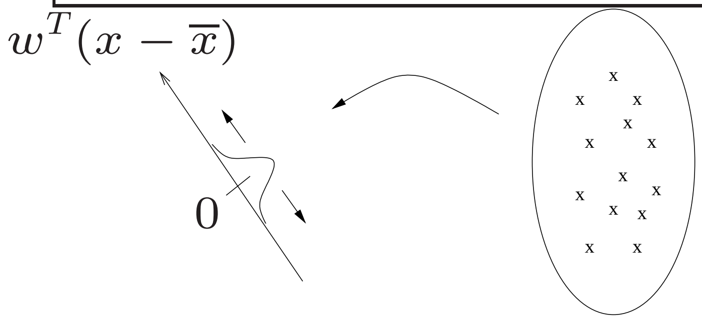
LS-SVM interpretation to FDA



Target space *Input space*

Minimize within class scatter

LS-SVM interpretation to PCA



Target space *Input space*

Find direction with maximal variance

LS-SVM formulation to linear PCA (1)

- Primal problem:

$$\boxed{\mathrm{P}}: \quad \max_{w,e} J_{\mathrm{P}}(w,e) = \gamma \frac{1}{2} \sum_{k=1}^{N} e_k^2 - \frac{1}{2} w^T w$$

$$\text{subject to} \quad e_k = w^T x_k, \; k = 1, ..., N$$

- Lagrangian $\mathcal{L}(w,e;\alpha) = \gamma \frac{1}{2} \sum_{k=1}^{N} e_k^2 - \frac{1}{2} w^T w - \sum_{k=1}^{N} \alpha_k \left(e_k - w^T x_k \right)$

- Conditions for optimality

$$\begin{cases} \frac{\partial \mathcal{L}}{\partial w} = 0 & \rightarrow \quad w = \sum_{k=1}^{N} \alpha_k x_k \\ \frac{\partial \mathcal{L}}{\partial e_k} = 0 & \rightarrow \quad \alpha_k = \gamma e_k, \qquad k = 1, ..., N \\ \frac{\partial \mathcal{L}}{\partial \alpha_k} = 0 & \rightarrow \quad e_k - w^T x_k = 0, \quad k = 1, ..., N \end{cases}$$

LS-SVM formulation to linear PCA (2)

- Elimination of variables e, w gives

$$\frac{1}{\gamma}\alpha_k - \sum_{l=1}^{N} \alpha_l x_l^T x_k = 0 \; , \quad k = 1, ..., N$$

- After defining $\lambda = 1/\gamma$ one obtains the eigenvalue problem

$$\boxed{\mathrm{D}} : \quad \text{solve in } \alpha :$$

$$\begin{bmatrix} x_1^T x_1 & \dots & x_1^T x_N \\ \vdots & & \vdots \\ x_N^T x_1 & \dots & x_N^T x_N \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_N \end{bmatrix} = \lambda \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_N \end{bmatrix}$$

as the dual problem (quantization in terms of $\lambda = 1/\gamma$).

LS-SVM formulation to linear PCA (3)

- Score variables become

$$z(x) = w^T x = \sum_{l=1}^{N} \alpha_l x_l^T x$$

- Optimal solution corresponding to largest eigenvalue

$$\sum_{k=1}^{N} (w^T x_k)^2 = \sum_{k=1}^{N} e_k^2 = \sum_{k=1}^{N} \frac{1}{\gamma^2} \alpha_k^2 = \lambda_{max}^2$$

  where $\sum_{k=1}^{N} \alpha_k^2 = 1$ for the normalized eigenvector.

- Many data: better solve primal problem
  Many inputs: better solve dual problem

Formulation with bias term (1)

- Usually: apply PCA analysis to *centered data* and consider

$$\max_w \sum_{k=1}^{N} [w^T (x_k - \hat{\mu}_x)]^2 \text{ where } \hat{\mu}_x = \frac{1}{N} \sum_{k=1}^{N} x_k.$$

- Bias term formulation: score variables $z(x) = w^T x + b$ and objective

$$\max_{w,b} \sum_{k=1}^{N} [0 - (w^T x_k + b)]^2$$

- Primal optimization problem

$$\boxed{\mathrm{P}} : \quad \begin{array}{rl} \displaystyle\max_{w,b,e} J_{\mathrm{P}}(w,e) = & \displaystyle\gamma \frac{1}{2} \sum_{k=1}^{N} e_k^2 - \frac{1}{2} w^T w \\ \text{subject to} & e_k = w^T x_k + b, \quad k = 1, ..., N \end{array}$$

Formulation with bias term (2)

- Conditions for optimality

$$
\begin{cases}
\frac{\partial \mathcal{L}}{\partial w} = 0 & \rightarrow \quad w = \sum_{k=1}^{N} \alpha_k x_k \\
\frac{\partial \mathcal{L}}{\partial e_k} = 0 & \rightarrow \quad \alpha_k = \gamma e_k, \qquad\qquad k = 1, ..., N \\
\frac{\partial \mathcal{L}}{\partial b} = 0 & \rightarrow \quad \sum_{k=1}^{N} \alpha_k = 0 \\
\frac{\partial \mathcal{L}}{\partial \alpha_k} = 0 & \rightarrow \quad e_k - w^T x_k - b = 0, \quad k = 1, ..., N
\end{cases}
$$

- Applying $\sum_{k=1}^{N} \alpha_k = 0$ yields

$$
b = -\frac{1}{N} \sum_{k=1}^{N} \sum_{l=1}^{N} \alpha_l x_l^T x_k.
$$

Formulation with bias term (3)

- By defining $\lambda = 1/\gamma$ one obtains the dual problem

$\boxed{\mathrm{D}}$: solve in α :

$$\left[\begin{array}{ccc} (x_1 - \hat{\mu}_x)^T (x_1 - \hat{\mu}_x) & \ldots & (x_1 - \hat{\mu}_x)^T (x_N - \hat{\mu}_x) \\ \vdots & & \vdots \\ (x_N - \hat{\mu}_x)^T (x_1 - \hat{\mu}_x) & \ldots & (x_N - \hat{\mu}_x)^T (x_N - \hat{\mu}_x) \end{array} \right] \left[\begin{array}{c} \alpha_1 \\ \vdots \\ \alpha_N \end{array} \right] = \lambda \left[\begin{array}{c} \alpha_1 \\ \vdots \\ \alpha_N \end{array} \right]$$

which is an eigenvalue decomposition of the centered Gram matrix

$$\Omega_c \alpha = \lambda \alpha$$

with $\Omega_c = M_c \Omega M_c$ where $M_c = I - 1_v 1_v^T / N$, $1_v = [1; 1; ...; 1]$ and $\Omega_{kl} = x_k^T x_l$ for $k, l = 1, ..., N$.

- Score variables: $z(x) = w^T x + b = \sum_{l=1}^{N} \alpha_l x_l^T x + b$.

Reconstruction problem for linear PCA (1)

- Reconstruction error:

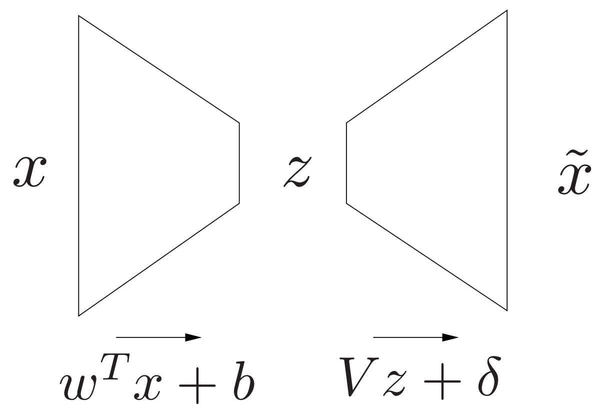
$$\min \sum_{k=1}^{N} \|x_k - \tilde{x}_k\|_2^2$$

where $\tilde{x}_k$ are variables reconstructed from the score variables, with

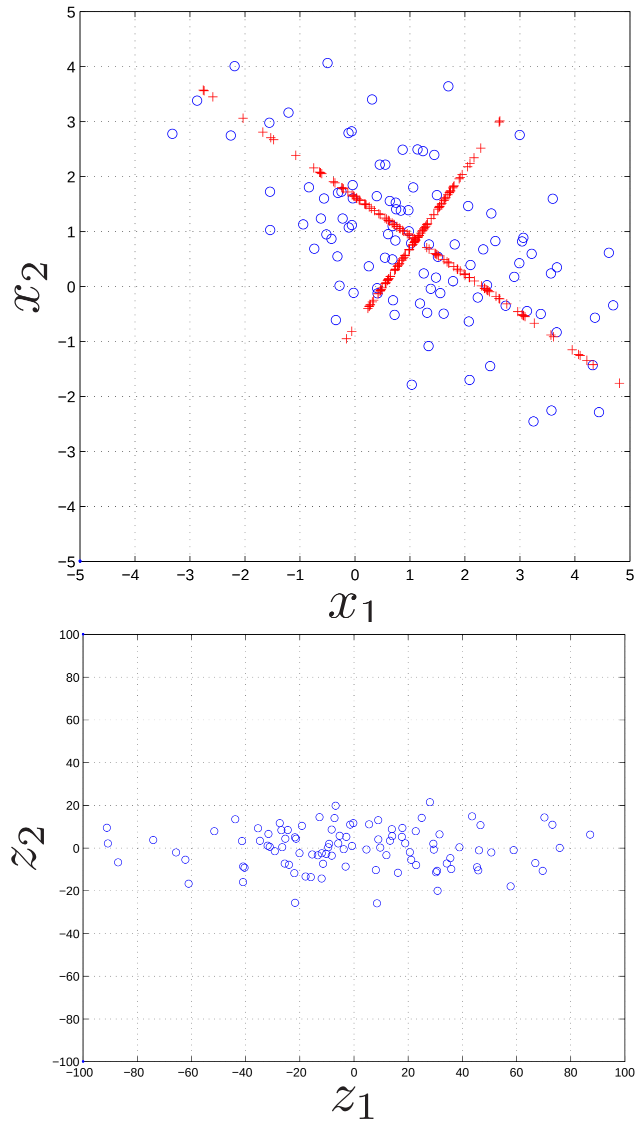
$$\tilde{x} = Vz + \delta$$

Hence $\min_{V,\delta} \sum_{k=1}^{N} \|x_k - (Vz_k + \delta)\|_2^2$.

- Information bottleneck:



Reconstruction problem for linear PCA (2)



LS-SVM approach to kernel PCA (1)

- Create nonlinear version of the method by

  - Mapping input space to a high dimensional feature space
  - Applying the kernel trick

  (kernel PCA - Schölkopf *et al.*; LS-SVM approach - Suykens *et al.*, 2002)

- Primal optimization problem:

$$\boxed{\mathrm{P}}: \quad \max_{w,e} J_{\mathrm{P}}(w,e) = \gamma \frac{1}{2} \sum_{k=1}^{N} e_k^2 - \frac{1}{2} w^T w$$
$$\text{subject to} \quad e_k = w^T(\varphi(x_k) - \hat{\mu}_{\varphi}), \; k = 1, ..., N.$$

- Lagrangian

$$\mathcal{L}(w,e;\alpha) = \gamma \frac{1}{2} \sum_{k=1}^{N} e_k^2 - \frac{1}{2} w^T w - \sum_{k=1}^{N} \alpha_k \left(e_k - w^T(\varphi(x_k) - \hat{\mu}_{\varphi}) \right)$$

LS-SVM approach to kernel PCA (2)

- Conditions for optimality

$$\begin{cases} \dfrac{\partial \mathcal{L}}{\partial w} = 0 & \rightarrow \quad w = \displaystyle\sum_{k=1}^{N} \alpha_k (\varphi(x_k) - \hat{\mu}_\varphi) \\ \dfrac{\partial \mathcal{L}}{\partial e_k} = 0 & \rightarrow \quad \alpha_k = \gamma e_k, \qquad\qquad\qquad\qquad\qquad k = 1, ..., N \\ \dfrac{\partial \mathcal{L}}{\partial \alpha_k} = 0 & \rightarrow \quad e_k - w^T (\varphi(x_k) - \hat{\mu}_\varphi) = 0, \quad k = 1, ..., N. \end{cases}$$

- By elimination of the variables e, w and defining $\lambda = 1/\gamma$ one obtains

$$\boxed{\mathrm{D}} : \quad \text{solve in } \alpha : \qquad \Omega_c \alpha = \lambda \alpha$$

with

$$\Omega_c = \begin{bmatrix} (\varphi(x_1) - \hat{\mu}_\varphi)^T (\varphi(x_1) - \hat{\mu}_\varphi) & \dots & (\varphi(x_1) - \hat{\mu}_\varphi)^T (\varphi(x_N) - \hat{\mu}_\varphi) \\ \vdots & & \vdots \\ (\varphi(x_N) - \hat{\mu}_\varphi)^T (\varphi(x_1) - \hat{\mu}_\varphi) & \dots & (\varphi(x_N) - \hat{\mu}_\varphi)^T (\varphi(x_N) - \hat{\mu}_\varphi) \end{bmatrix}$$

LS-SVM approach to kernel PCA (3)

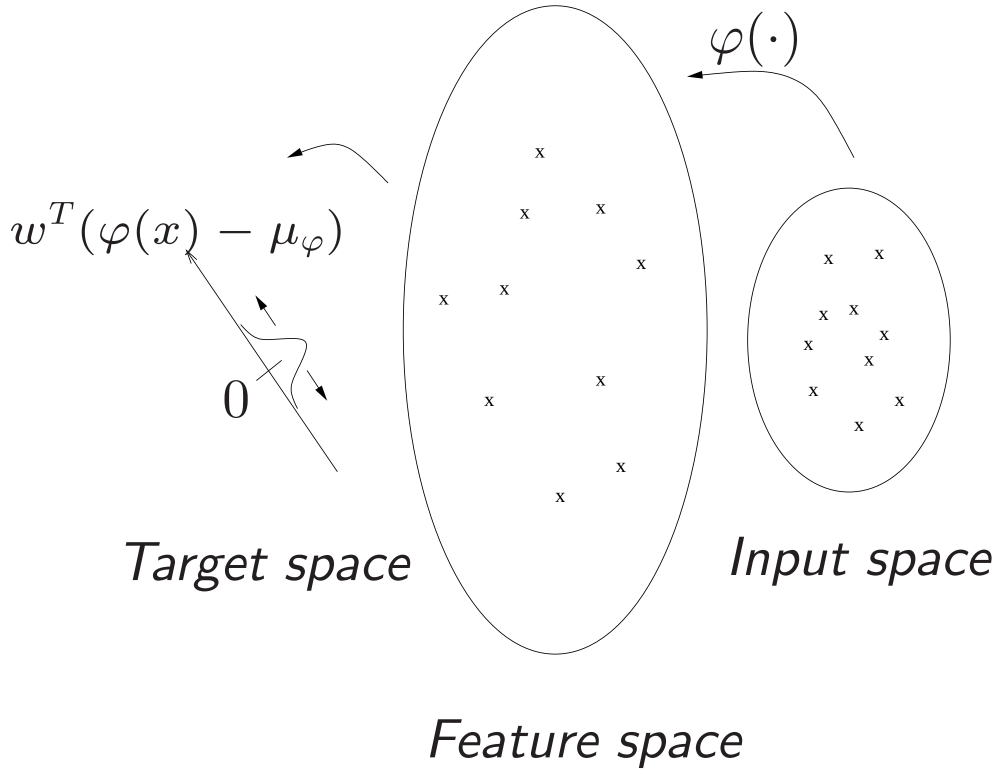
- Elements of the centered kernel matrix

$$\Omega_{c,kl} = (\varphi(x_k) - \hat{\mu}_\varphi)^T (\varphi(x_l) - \hat{\mu}_\varphi) \,, \qquad k, l = 1, ..., N$$

- Score variables

$$\begin{aligned} z(x) &= w^T \left(\varphi(x) - \hat{\mu}_\varphi\right) \\ &= \sum_{l=1}^{N} \alpha_l \left(\varphi(x_l) - \hat{\mu}_\varphi\right)^T \left(\varphi(x) - \hat{\mu}_\varphi\right) \\ &= \sum_{l=1}^{N} \alpha_l \Bigg(K(x_l, x) - \tfrac{1}{N} \sum_{r=1}^{N} K(x_r, x) - \tfrac{1}{N} \sum_{r=1}^{N} K(x_r, x_l) + \\ &\qquad \frac{1}{N^2} \sum_{r=1}^{N} \sum_{s=1}^{N} K(x_r, x_s) \Bigg) . \end{aligned}$$

LS-SVM approach to kernel PCA (4)



Find direction with maximal variance

Example: denoising by kernel PCA (1)

- For the nonlinear PCA case the number of score variables n_s can be larger than the dimension of the input space n. One selects then as few score variables as possible and minimize the reconstruction error. In this form of nonlinear PCA the mappings are nonlinear.

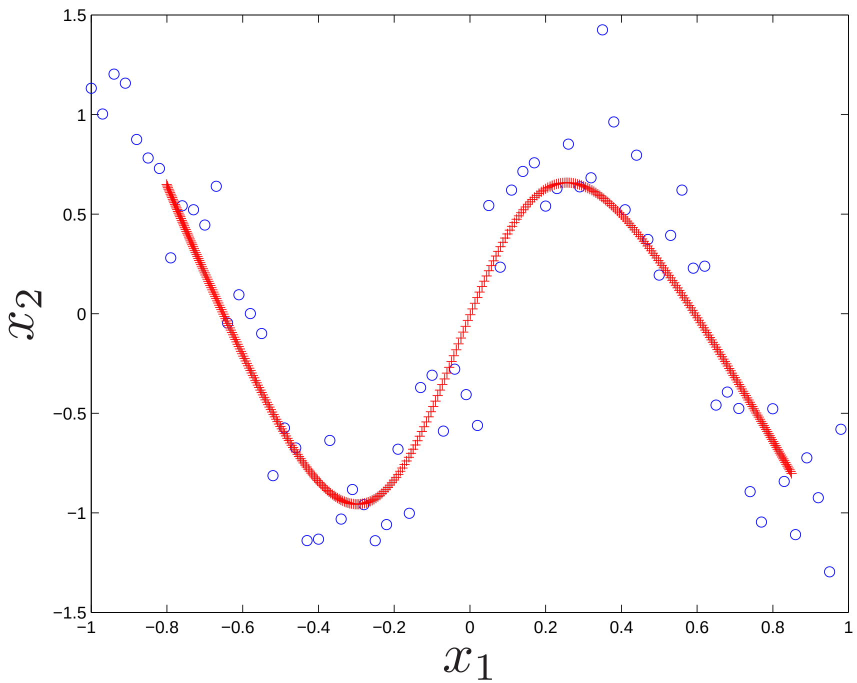
- The mapping from the score variables to the reconstructed input variables is done as

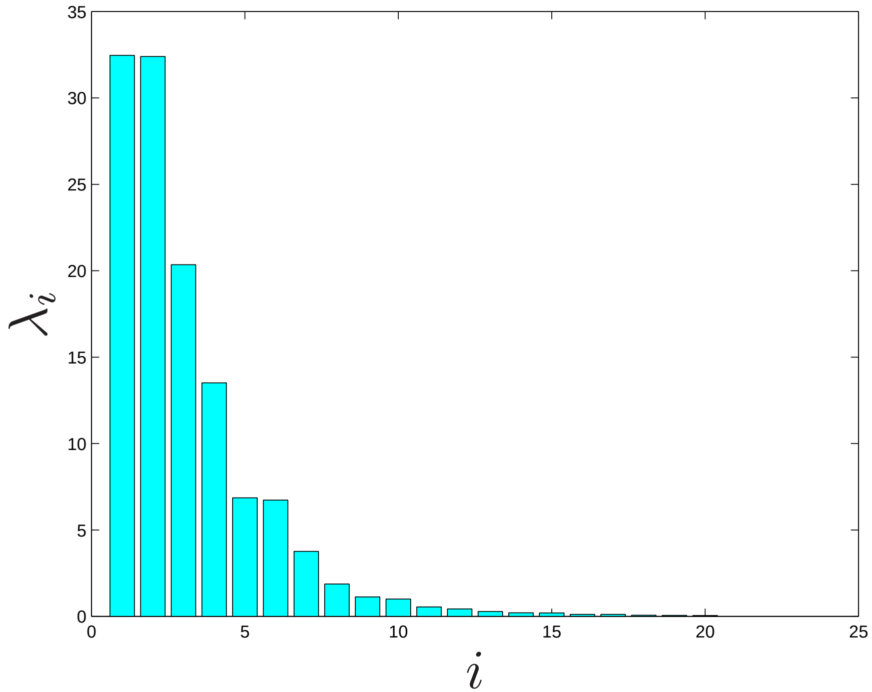
$$\tilde{x} = h(z)$$

such that one minimizes the reconstruction error

$$\min \sum_{k=1}^{N} \|x_k - h(z_k)\|_2^2$$

Example: denoising by kernel PCA (2)





Example: Denoising a noisy sine function. For the nonlinear mapping h an MLP with one hidden layer has been taken which was trained by Bayesian learning.

Example: denoising by kernel PCA (3)

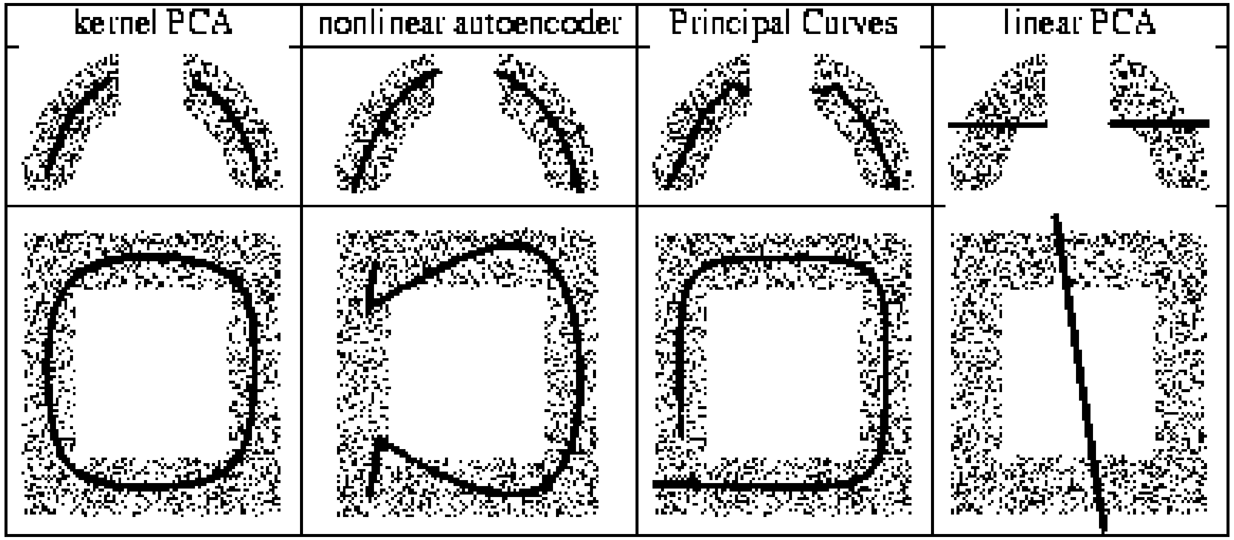


Figure 1: De-noising in 2-d (see text). Depicted are the data set (small points) and its de-noised version (big points, joining up to solid lines). For linear PCA, we used one component for reconstruction, as using two components, reconstruction is perfect and thus does not de-noise. Note that all algorithms except for our approach have problems in capturing the circular structure in the bottom example.

Schölkopf B., Mika S., Burges C., Knirsch P., Müller K.-R., Rätsch G., Smola A., Input space vs. feature space in kernel-based methods, IEEE Transactions on Neural Networks, 10(5), 1000-1017, 1999.

Example: denoising by kernel PCA (4)

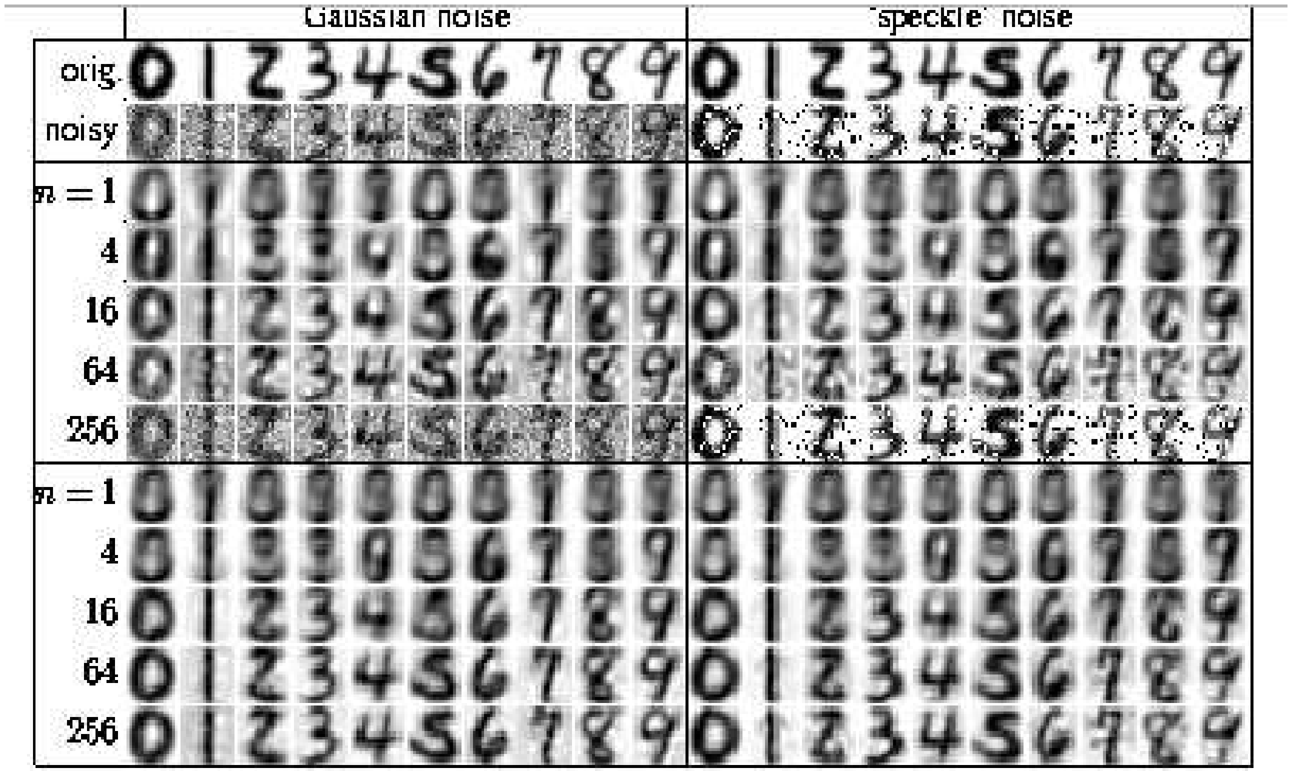


Figure 1: De-Noising of USPS data (see text). The left half shows: *top:* the first occurrence of each digit in the test set, *second row:* the upper digit with additive Gaussian noise ($\sigma = 0.5$), *following five rows:* the reconstruction for linear PCA using $n = 1, 4, 16, 64, 256$ components, and, *last five rows:* the results of our approach using the same number of components. In the right half we show the same but for 'speckle' noise with probability $p = 0.4$.

Density estimation by kernel PCA (1)

- A link between kernel PCA and orthogonal series density estimation has been established [Girolami, 2002].

- One can then take the scores resulting from kernel PCA as basis functions for a density estimator. Therefore, one considers the eigenvalue decomposition of the centered kernel matrix

$$\Omega_c U = U \tilde{\Lambda}$$

where $\tilde{\Lambda} = \mathrm{diag}([\tilde{\lambda}_1; ...; \tilde{\lambda}_N])$ contains the eigenvalues and $U = [u_1 ... u_N] \in \mathbb{R}^{N \times N}$ the corresponding eigenvectors.

Density estimation by kernel PCA (2)

- This can be used in order to estimate the eigenfunctions $\phi_i(x)$ and eigenvalues λ_i for the integral equation (Karhunen-Loeve expansion)

$$\int K(x, x')\phi_i(x)p(x)dx = \lambda_i \phi_i(x')$$

with the estimates (Nyström method)

$$\hat{\lambda}_i = \frac{1}{N}\tilde{\lambda}_i, \;\; \hat{\phi}_i(x_k) = \sqrt{N} u_{ki}, \;\; \hat{\phi}_i(x') = \frac{\sqrt{N}}{\tilde{\lambda}_i} \sum_{k=1}^{N} u_{ki} K(x_k, x')$$

where u_{ki} denotes the ki-th entry of the matrix U.

Density estimation by kernel PCA (3)

- Using the eigenvectors as finite sample estimates of the corresponding eigenfunctions, the truncated estimate of the probability density function at point x' is given by

$$\begin{aligned} \hat{p}_M(x') &= \frac{1}{N} 1_v^T \sum_{i=1}^{M} \sqrt{\tilde{\lambda}_i} u_i \sum_{k=1}^{N} \frac{1}{\sqrt{\tilde{\lambda}_i}} u_{ki} K(x_k, x') \\ &= \frac{1}{N} 1_v^T U_M U_M^T \theta(x') \end{aligned}$$

  where $\theta(x') = [K(x', x_1); K(x', x_2); ...; K(x', x_N)]$, $1_v = [1; 1; \ ...; 1]$ and $U_M \in \mathbb{R}^{N \times M}$ is the matrix with eigenvectors of Ω_c consisting of the eigenvectors corresponding to the M largest eigenvalues. (Note: normalizations are done for the kernel K).

- For the case of $M = N$ this reduces to the well-known Parzen window density estimator $p(x') = \frac{1}{N} 1_v^T \theta(x')$.

Density estimation by kernel PCA (4)

- Cutoff value for determination of the value of M:

$$(\frac{1}{N} 1_v^T u_i)^2 > \frac{2N}{1+N}.$$

An estimate for the overall integrated square truncation error $\sum_{i=M+1}^{\infty} c_i^2$ is given by

$$c_i^2 \simeq \tilde{\lambda}_i (\frac{1}{N} 1_v^T u_i)^2.$$

This can also be related to the quadratic Renyi entropy

$$H_R = -\log \int p(x)^2 dx$$

Density estimation by kernel PCA (5)

- One can show that

$$\int \hat{p}(x)^2 dx = \sum_{i=1}^{N} \tilde{\lambda}_i (\frac{1}{N} 1_v^T u_i)^2.$$

Large contributions to the entropy come from components that have small values of $\tilde{\lambda}_i(\frac{1}{N}1_v^T u_i)^2$ and are related to elements with little or no structure, caused by observation noise or diffuse regions in the data. Large values of $\tilde{\lambda}_i(\frac{1}{N}1_v^T u_i)^2$ on the other hand indicate regions of high density or compactness.

- More generally (beyond RBF kernels):

$$\int \hat{p}(x)^2 dx = \frac{1}{N^2} 1_v^T K 1_v$$

Canonical Correlation Analysis

- CCA analysis has applications e.g. in system identification, signal processing, bioinformatics and textmining.
- Objective: find a maximal correlation between the projected variables $z_x = w^T x$ and $z_y = v^T y$ where $x \in \mathbb{R}^{n_x}, y \in \mathbb{R}^{n_y}$ (zero mean).
- Maximize the correlation coefficient

$$\max_{w,v} \rho = \frac{\mathcal{E}[z_x z_y]}{\sqrt{\mathcal{E}[z_x z_x]}\sqrt{\mathcal{E}[z_y z_y]}} = \frac{w^T C_{\mathrm{xy}} v}{\sqrt{w^T C_{\mathrm{xx}} w}\sqrt{v^T C_{\mathrm{yy}} v}}$$

with $C_{\mathrm{xx}} = \mathcal{E}[xx^T]$, $C_{\mathrm{yy}} = \mathcal{E}[yy^T]$, $C_{\mathrm{xy}} = \mathcal{E}[xy^T]$. This is formulated as the constrained optimization problem

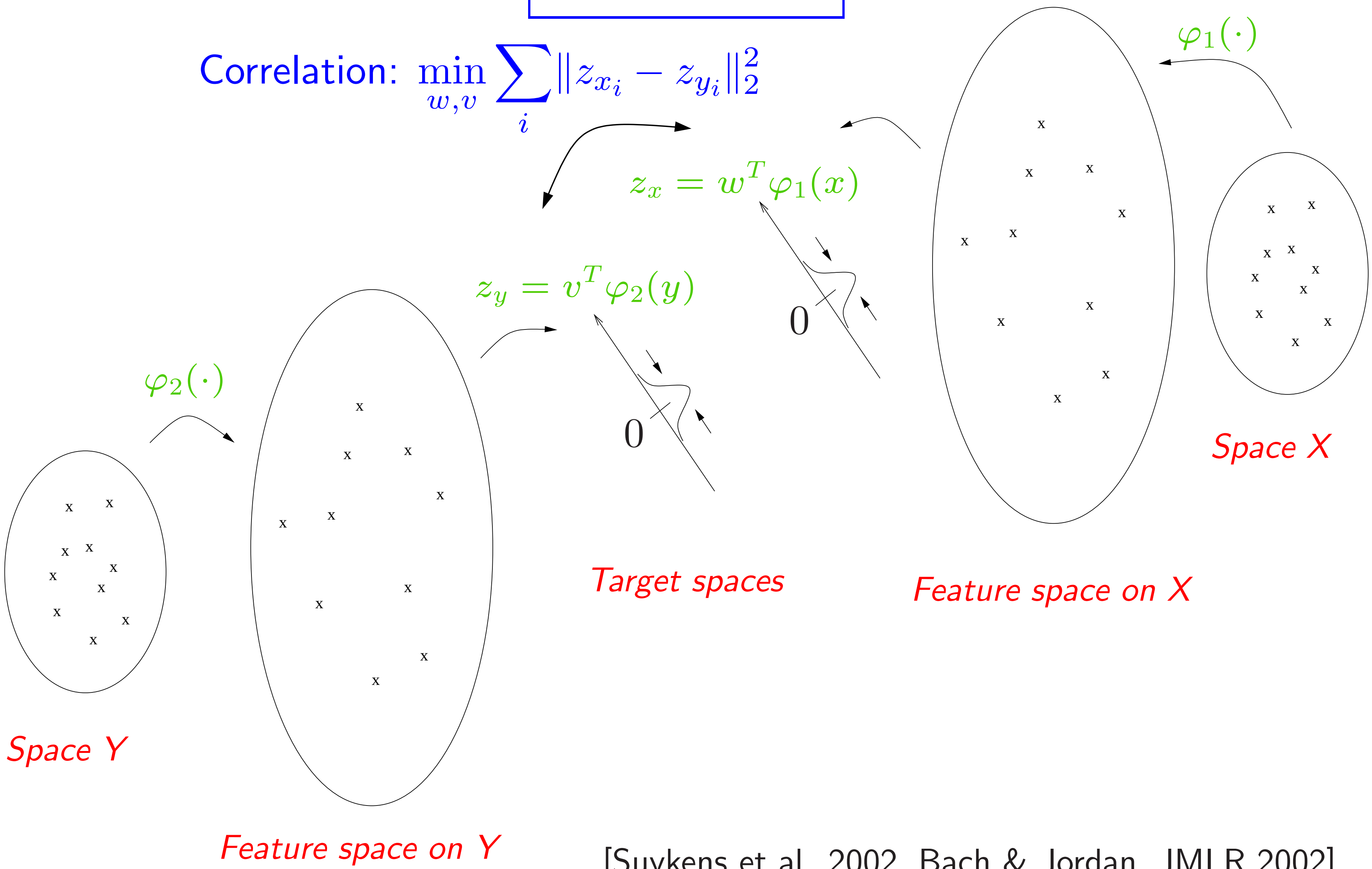
$$\max_{w,v} w^T C_{\mathrm{xy}} v \;\; \text{s.t.} \;\; w^T C_{\mathrm{xx}} w = 1 \;\; \text{and} \;\; v^T C_{\mathrm{yy}} v = 1$$

which leads to the generalized eigenvalue problem

$$C_{\mathrm{xy}} v = \eta \, C_{\mathrm{xx}} w, \;\; C_{\mathrm{yx}} w = \nu \, C_{\mathrm{yy}} v.$$

Kernel CCA

Correlation: $\min_{w,v} \sum_i \|z_{x_i} - z_{y_i}\|_2^2$



[Suykens et al. 2002, Bach & Jordan, JMLR 2002]

LS-SVM formulation to Kernel CCA

- Score variables: $z_x = w^T(\varphi_1(x) - \hat{\mu}_{\varphi_1}), z_y = v^T(\varphi_2(y) - \hat{\mu}_{\varphi_2})$

Feature maps φ_1, φ_2, kernels $K_1(x_i, x_j) = \varphi_1(x_i)^T \varphi_1(x_j), K_2(y_i, y_j) = \varphi_2(y_i)^T \varphi_2(y_j)$

- Primal problem: (Kernel PLS case: $\nu_1 = 0, \nu_2 = 0$ [Hoegaerts et al., 2004])

$$\max_{w,v,e,r} \quad \gamma \sum_{i=1}^{N} e_i r_i - \nu_1 \frac{1}{2} \sum_{i=1}^{N} e_i^2 - \nu_2 \frac{1}{2} \sum_{i=1}^{N} r_i^2 - \frac{1}{2} w^T w - \frac{1}{2} v^T v$$

$$\text{subject to} \quad e_i = w^T(\varphi_1(x_i) - \hat{\mu}_{\varphi_1}), \quad r_i = v^T(\varphi_2(y_i) - \hat{\mu}_{\varphi_2}), \ \forall i$$

with $\hat{\mu}_{\varphi_1} = (1/N)\sum_{i=1}^N \varphi_1(x_i)$, $\hat{\mu}_{\varphi_2} = (1/N)\sum_{i=1}^N \varphi_2(y_i)$.

- Dual problem: generalized eigenvalue problem [Suykens et al. 2002]

$$\begin{bmatrix} 0 & \Omega_{c,2} \\ \Omega_{c,1} & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \lambda \begin{bmatrix} \nu_1 \Omega_{c,1} + I & 0 \\ 0 & \nu_2 \Omega_{c,2} + I \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix}, \lambda = 1/\gamma$$

with $\Omega_{c,1_{ij}} = (\varphi_1(x_i) - \hat{\mu}_{\varphi_1})^T(\varphi_1(x_j) - \hat{\mu}_{\varphi_1}), \Omega_{c,2_{ij}} = (\varphi_2(y_i) - \hat{\mu}_{\varphi_2})^T(\varphi_2(y_j) - \hat{\mu}_{\varphi_2})$

Obtaining solution from Lagrangian

- Lagrangian $\mathcal{L}(w, v, e, r; \alpha, \beta) = \gamma \sum_{i=1}^N e_i r_i - \nu_1 \frac{1}{2} \sum_{i=1}^N e_i^2 - \nu_2 \frac{1}{2} \sum_{i=1}^N r_i^2$
$-\frac{1}{2} w^T w - \frac{1}{2} v^T v - \sum_{i=1}^N \alpha_i [e_i - w^T (\varphi_1(x_i) - \hat{\mu}_{\varphi_1})] - \sum_{i=1}^N \beta_i [r_i - v^T (\varphi_2(y_i) - \hat{\mu}_{\varphi_2})]$

- Conditions for optimality (eliminate w, v, e, r)

$$
\left\{
\begin{array}{lll}
\frac{\partial \mathcal{L}}{\partial w} = 0 & \rightarrow & w = \sum_{i=1}^N \alpha_i (\varphi_1(x_i) - \hat{\mu}_{\varphi_1}) \\
\frac{\partial \mathcal{L}}{\partial v} = 0 & \rightarrow & v = \sum_{i=1}^N \beta_i (\varphi_2(y_i) - \hat{\mu}_{\varphi_2}) \\
\frac{\partial \mathcal{L}}{\partial e_i} = 0 & \rightarrow & \gamma v^T (\varphi_2(y_i) - \hat{\mu}_{\varphi_2}) = \nu_1 w^T (\varphi_1(x_i) - \hat{\mu}_{\varphi_1}) + \alpha_i \\
 & & \qquad i = 1, \ldots, N \\
\frac{\partial \mathcal{L}}{\partial r_i} = 0 & \rightarrow & \gamma w^T (\varphi_1(x_i) - \hat{\mu}_{\varphi_1}) = \nu_2 v^T (\varphi_2(y_i) - \hat{\mu}_{\varphi_2}) + \beta_i \\
 & & \qquad i = 1, \ldots, N \\
\frac{\partial \mathcal{L}}{\partial \alpha_i} = 0 & \rightarrow & e_i = w^T (\varphi_1(x_i) - \hat{\mu}_{\varphi_1}) \\
 & & \qquad i = 1, \ldots, N \\
\frac{\partial \mathcal{L}}{\partial \beta_i} = 0 & \rightarrow & r_i = v^T (\varphi_2(y_i) - \hat{\mu}_{\varphi_2}) \\
 & & \qquad i = 1, \ldots, N
\end{array}
\right.
$$

Kernel CCA applications - textmining

- A. Vinokourov, J. Shawe-Taylor, N. Cristianini, Inferring a semantic representation of text via cross-language correlation analysis, NIPS 2002.

- Learning a semantic representation of a text document from data in a cross-lingual setting.

- Corpus of unlabelled paired documents. Each pair is formed by an English document and its French translation.

- Learning correlation between the two spaces by kernel CCA and bag-of-words approach.

- Certain patterns of English words that relate to a specific meaning correlate with patterns of French words with the same meaning across the corpus.

Kernel CCA applications - bioinformatics

- [Vert & Kanehisa, Bioinformatics 2003]:
  For kernels related to spaces X and Y

  K_1 : graph from gene network
  K_2 : gene expression profiles

  Study correlation between gene network and set of profiles
  Able to extract biologically relevant expression patterns and pathways with related activity.

- [Yamanishi et al., Bioinformatics 2003]:
  Extract correlated gene clusters from multiple genomic data. Successfully tested on the ability to recognize operons in the *Escherichia coli* genome, from the comparison of three data sets:

  1. functional relationships between genes in metabolic pathways
  2. geometrical relationships along the chromosome
  3. co-expression relationships as observed by gene expression data

Kernel CCA applications - system identification (1)

- Given I/O data, estimate parameter vector θ of the nonlinear state space model:

$$\left\{ \begin{array}{rcl} x_{k+1} & = & f(x_k, u_k; \theta) \\ y_k & = & g(x_k, u_k; \theta) \end{array} \right.$$

- Conceptually 2 steps:
  - Step 1: estimate state vector sequence $\{\hat{x}_k\}$ from the I/O data
  - Step 2: given I/O data and $\{\hat{x}_k\}$, solve a set of nonlinear equations to estimate θ.

- Estimation of state vector sequence using **Kernel CCA** [Verdult et al., MTNS 2004]

Kernel CCA applications - system identification (2)

- **Kernel CCA**: primal formulation [Suykens et al., 2002]

$$\min_{w,v,b,d,e,r} w^T w + v^T v + \nu \sum_i (e_i - r_i)^2 \text{ s.t. } \begin{cases} e_i &= w^T \varphi_1(x_i) + b, \forall i \\ r_i &= v^T \varphi_2(z_i) + d, \forall i \end{cases}$$

  - Data $\{x_i\}$: **past** *of time-series*
  - Data $\{z_i\}$: **future** *of time-series*
  - **State vector sequence from kernel CCA**
  - System order estimate from kernel CCA

- Dual problem: generalized eigenvalue problem