Manifold Learning for Signal and Image Analysis Lecture 5: A Brief Introduction to Kernel Methods

Radu Horaud
INRIA Grenoble Rhone-Alpes, France
Radu.Horaud@inria.fr
http://perception.inrialpes.fr/

Outline of Lecture 5

- Linear regression in "feature space"
- Kernel construction and characterization of the feature space.
- The kernel (Gram) matrix
- The covariance matrix in feature space
- Feature-space computations
- Kernal PCA

Material for This Lecture

- C. Bishop. Pattern Analysis and Machine Learning (chapters 6 and 12).
- J. Shawe-Taylor & N. Cristianini. Kernel Methods in Pattern Analysis (chapters 2, 3, 5 and 6).

The Kernel Function

- Consider a data set: $\mathbf{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n] \in \mathbb{R}^D$
- **Definition of a kernel function**: consider a *nonlinear feature space* mapping: $\boldsymbol{\phi} : \boldsymbol{x} \rightarrow \boldsymbol{\phi}(\boldsymbol{x})$, with $\boldsymbol{\phi}(\boldsymbol{x}) \in \mathbb{R}^M$. A kernel satisfies:

$$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{\phi}(\boldsymbol{x}_i)^\top \boldsymbol{\phi}(\boldsymbol{x}_j) = \langle \boldsymbol{\phi}(\boldsymbol{x}_i), \boldsymbol{\phi}(\boldsymbol{x}_j) \rangle$$

- The main principle of *kernel methods* is to interpret the kernel function as an *inner product* in feature space and to design algorithms without making explicit the function $\boldsymbol{\phi}$.
- This extends many algorithms by making use of the *kernel trick* or *kernel substitution*.
- For example, we can extend basic algorithms, such as PCA and LDA in feature space, namely *kernel PCA* and *kernel Fisher discriminant*, etc.

Linear Regression in Feature Space

- Replace the standard regression problem with:

$$y = \sum_{m=1}^{M} w_m \phi_m(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x})$$

- The parameters $w_1, \ldots, w_M$ can be estimated from a training set of pairs $(y_j, \boldsymbol{x}_j)$ by minimizing the following criterion:

$$J(\boldsymbol{w}) = \frac{1}{2} \left(\sum_{j=1}^{n} (\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}_j) - y_j)^2 + \lambda \boldsymbol{w}^\top \boldsymbol{w} \right)$$

Least-square Solution

- By taking the derivatives of J with respect to $\boldsymbol{w}$ and setting them to zero, we obtain the following solution:

$$\boldsymbol{w} = -\frac{1}{\lambda}\sum_{j=1}^{n}(\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}_j) - y_j)\boldsymbol{\phi}(\boldsymbol{x}_j)$$

- Let $a_j = -\frac{1}{\lambda}(\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}_j) - y_j)$ be the j-th entry of a vector $\boldsymbol{a} \in \mathbb{R}^n$.
- Let $\boldsymbol{\Phi} = [\boldsymbol{\phi}(\boldsymbol{x}_1) \dots \boldsymbol{\phi}(\boldsymbol{x}_j) \dots \boldsymbol{\phi}(\boldsymbol{x}_n)]$ be a $M \times n$ data matrix in feature space.
- Hence:

$$\boldsymbol{w} = \boldsymbol{\Phi}\boldsymbol{a}$$

- We will use $\boldsymbol{a}$ instead of $\boldsymbol{w}$.

Dual representation

- Substitute $\boldsymbol{w} = \boldsymbol{\Phi}\boldsymbol{a}$ in $J(\boldsymbol{w})$. We obtain:

$$J(\boldsymbol{a}) = \frac{1}{2}\left(\boldsymbol{a}^\top \boldsymbol{\Phi}^\top \boldsymbol{\Phi}\boldsymbol{\Phi}^\top \boldsymbol{\Phi}\boldsymbol{a} - \boldsymbol{a}^\top \boldsymbol{\Phi}^\top \boldsymbol{\Phi}\boldsymbol{y} + \lambda \boldsymbol{a}^\top \boldsymbol{\Phi}^\top \boldsymbol{\Phi}\boldsymbol{a}\right)$$

- The $n \times n$ matrix:

$$\mathbf{K} = \boldsymbol{\Phi}^\top \boldsymbol{\Phi}$$

  is a Gram matrix in feature space (it will be referred to as a kernel matrix), with entries:

$$\kappa_{ij} = \langle \boldsymbol{\phi}(\boldsymbol{x}_i), \boldsymbol{\phi}(\boldsymbol{x}_j) \rangle = \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

Solution in Feature Space

- We obtain a new expression for $J(\boldsymbol{w})$ as a function of vector $\boldsymbol{a}$ and of the Gram matrix:

$$J(\boldsymbol{a}) = \frac{1}{2}\left(\boldsymbol{a}^\top \mathbf{K}^\top \mathbf{K}\boldsymbol{a} - \boldsymbol{a}^\top \mathbf{K}\boldsymbol{y} + \lambda \boldsymbol{a}^\top \mathbf{K}\boldsymbol{a}\right)$$

- The solution is obtained by setting the gradient of J with respect to $\boldsymbol{a}$ to zero:

$$\boldsymbol{a} = (\mathbf{K} + \lambda \mathbf{I})^{-1}\boldsymbol{y}$$

which always has an inverse.

Back to Linear Regression

- The linear regression model allows to predict the output y from a new input $\boldsymbol{x}$:

$$y = \sum_{m=1}^{M} w_m \phi_m(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x})$$

- By substitution this becomes:

$$y = \boldsymbol{a}^\top \boldsymbol{\Phi}^\top \boldsymbol{\phi}(\boldsymbol{x}) = \sum_{j=1}^{n} a_j \kappa(\boldsymbol{x}_j, \boldsymbol{x})$$

- The mapping ϕ is not required!

Discussion

- The dual representation allows the solution to be expressed entirely in terms of the kernel function;
- Inversion of a $M \times M$ matrix (dimension of the feature space) is replaced by inversion of a $n \times n$ matrix (number of points in the training set).
- It avoids computations in feature space when M is very large.
- The feature-space is a vector space equipped with an inner-product – metric space;
- This means that there is a strong similarity between feature-space methods and MDS (only the pairwise inner-product between data points are needed to construct algorithms).

Constructing Kernels

- A valid kernel function is such that the associated Gram matrix is symmetric positive semidefinite.
- The simplest kernel corresponds to $\boldsymbol{\phi}(\boldsymbol{x}) = \boldsymbol{x}$, or

$$\kappa(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{x}^\top \boldsymbol{x}'$$

- Valid kernels:

$$c\kappa(\boldsymbol{x}, \boldsymbol{x}') \text{ with } c > 0; \;\; f(\boldsymbol{x})\kappa(\boldsymbol{x}, \boldsymbol{x}')f(\boldsymbol{x}'); \;\; \exp(\kappa(\boldsymbol{x}, \boldsymbol{x}'));$$
$$\kappa_1(\boldsymbol{x}, \boldsymbol{x}') + \kappa_2(\boldsymbol{x}, \boldsymbol{x}'); \;\; \kappa_1(\boldsymbol{x}, \boldsymbol{x}')\kappa_2(\boldsymbol{x}, \boldsymbol{x}');$$
$$\kappa(\boldsymbol{\phi}(\boldsymbol{x}), \boldsymbol{\phi}(\boldsymbol{x}')); \;\; \boldsymbol{x}^\top \mathbf{A} \boldsymbol{x}' \text{ with } \mathbf{A} \succeq 0.$$

Kernel Normalization

- $\boldsymbol{x} \rightarrow \phi(\boldsymbol{x})/\|\phi(\boldsymbol{x}\|$ which yields:

$$
\begin{aligned}
\hat{\kappa}(\boldsymbol{x}, \boldsymbol{x}') &= \frac{\kappa(\boldsymbol{x}, \boldsymbol{x}')}{\sqrt{\kappa(\boldsymbol{x}, \boldsymbol{x})\kappa(\boldsymbol{x}', \boldsymbol{x}')}} \\
&= \kappa(\boldsymbol{x}, \boldsymbol{x})^{-1/2}\kappa(\boldsymbol{x}, \boldsymbol{x}')\kappa(\boldsymbol{x}', \boldsymbol{x}')^{-1/2}
\end{aligned}
$$

The Gaussian Kernel

$$\kappa(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{2\sigma^2}\right)$$

- $\|\boldsymbol{x} - \boldsymbol{x}'\|^2 = \boldsymbol{x}^\top \boldsymbol{x} + \boldsymbol{x}'^\top \boldsymbol{x}' - 2\boldsymbol{x}^\top \boldsymbol{x}'$
- Let $f(\boldsymbol{x}) = \exp(-\boldsymbol{x}^\top \boldsymbol{x}/2\sigma^2) = \frac{1}{\sqrt{\exp(\boldsymbol{x}^\top \boldsymbol{x}/\sigma^2)}}$
- The Gaussian kernel writes:

$$\begin{aligned} \kappa(\boldsymbol{x}, \boldsymbol{x}') &= f(\boldsymbol{x}) \exp(\boldsymbol{x}^\top \boldsymbol{x}'/\sigma^2) f(\boldsymbol{x}') \\ &= \frac{\exp(\boldsymbol{x}^\top \boldsymbol{x}'/\sigma^2)}{\sqrt{\exp(\boldsymbol{x}^\top \boldsymbol{x}/\sigma^2) \exp(\boldsymbol{x}'^\top \boldsymbol{x}'/\sigma^2)}} \end{aligned}$$

- This is also known as the *basis radial function* (BRF) kernel.

Mercer Kernel (In Brief!)

- Let X be a compact subset of $\mathbb{R}^D$. Suppose that k is a continuous and symmetric function such that the integral operator is positive

$$\int_{X \times X} \kappa(\boldsymbol{x}, \boldsymbol{x}') f(\boldsymbol{x}) f(\boldsymbol{x}') d\boldsymbol{x} d\boldsymbol{x}' \geq 0$$

  for all $f \in L_2(X)$. *An L_2 function is a function that is square integrable.*

- We can expand $\kappa(\boldsymbol{x}, \boldsymbol{x}')$ in a uniformly convergent series in terms of functions $\{\phi_i\}_{i=1}^{\infty}$ satisfying $\langle \phi_i, \phi_j \rangle = \delta_{ij}$

$$\kappa(\boldsymbol{x}, \boldsymbol{x}') = \sum_{i=1}^{\infty} \phi_i(\boldsymbol{x}) \phi_i(\boldsymbol{x}')$$

Inner-product Space

- A vector space is an inner-product space if there exists a real-valued symmetric bilinear map that satisfies:

$$\langle \boldsymbol{x}, \boldsymbol{x} \rangle \ \geq \ 0$$

- The inner product is *strict* if: $\langle \boldsymbol{x}, \boldsymbol{x} \rangle = 0$ iff $\boldsymbol{x} = 0$.
- A strict inner product allows to define a norm of a vector $\|\boldsymbol{x}\|_2 = \sqrt{\langle \boldsymbol{x}, \boldsymbol{x} \rangle}$ and an associated *metric* or distance $\|\boldsymbol{x} - \boldsymbol{x}'\|_2$.
- *A vector space with a metric is known as a* **metric space**.
- The feature space is a metric space, equipped with the strict inner product.

The Gram/Kernel Matrix

- The $n \times n$ matrix:

$$\mathbf{K} = \mathbf{\Phi}^\top \mathbf{\Phi}$$

is a Gram matrix in feature space, with entries:

$$\kappa_{ij} = \langle \boldsymbol{\phi}(\boldsymbol{x}_i), \boldsymbol{\phi}(\boldsymbol{x}_j) \rangle = \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

- We remind that $\mathbf{\Phi} = [\boldsymbol{\phi}(\boldsymbol{x}_1) \dots \boldsymbol{\phi}(\boldsymbol{x}_j) \dots \boldsymbol{\phi}(\boldsymbol{x}_n)]$ is the $M \times n$ data matrix.
- It is symmetric, positive, semidefinite:

$$\boldsymbol{x}^\top \mathbf{K} \boldsymbol{x} = \boldsymbol{x}^\top \mathbf{\Phi}^\top \mathbf{\Phi} \boldsymbol{x} = \|\mathbf{\Phi} \boldsymbol{x}\|_2^2$$

- This matrix was studied in Lecture #1 within the context of MDS. Here we have a generalization because each entry is a kernel function which is more general hat the dot-product of MDS.

Spectral Decomposition of the Kernel Matrix

- Let $(\lambda_1, \boldsymbol{v}_1), \dots, (\lambda_n, \boldsymbol{v}_n)$ be the eigenvalue-eigenvector pairs of a Kernel matrix. It can be written as:

$$\mathbf{K} = \sum_{k=1}^{n} \lambda_k \boldsymbol{v}_k \boldsymbol{v}_k^\top$$

- Each matrix entry can be written as:

$$\kappa_{ij} = \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sum_{k=1}^{n} \lambda_k v_{ik} v_{jk} = \langle \boldsymbol{\phi}(\boldsymbol{x}_i), \boldsymbol{\phi}(\boldsymbol{x}_j) \rangle$$

- with $\boldsymbol{\phi}(\boldsymbol{x}_i) = (\sqrt{\lambda_1} v_{i1}, \dots \sqrt{\lambda_k} v_{ik}, \dots \sqrt{\lambda_n} v_{in})^\top$.
- Therefore, we can think of the eigenvectors as defining a feature space.

Feature-space Computations

- The norm of a feature-space vector:

$$\|\boldsymbol{\phi}(\boldsymbol{x})\|_2^2 = \langle \boldsymbol{\phi}(\boldsymbol{x}), \boldsymbol{\phi}(\boldsymbol{x}) \rangle = \kappa(\boldsymbol{x}, \boldsymbol{x})$$

- The norm of a linear combination:

$$\left\| \sum_{i=1}^{n} \alpha_i \boldsymbol{\phi}(\boldsymbol{x}_i) \right\|^2 = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

- Distance between two feature-space vectors:

$$\|\boldsymbol{\phi}(\boldsymbol{x}_i) - \boldsymbol{\phi}(\boldsymbol{x}_j)\|_2^2 = \kappa(\boldsymbol{x}_i, \boldsymbol{x}_i) + \kappa(\boldsymbol{x}_j, \boldsymbol{x}_j) - 2\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

Center of Mass

- Notation: $\overline{\boldsymbol{\phi}} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{\phi}(\boldsymbol{x}_i)$
- There is no explicit dual representation for this point. Moreover, it is not the image of a "valid" data point.
- Norm, distance from a point, and expected distance:

$$
\begin{aligned}
\|\overline{\boldsymbol{\phi}}\|^2 &= \sum_{i=1}^{n}\sum_{j=1}^{n}\frac{1}{n}\frac{1}{n}\kappa(\boldsymbol{x}_i,\boldsymbol{x}_j) = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\kappa(\boldsymbol{x}_i,\boldsymbol{x}_j) \\
\|\boldsymbol{\phi}(\boldsymbol{x}) - \overline{\boldsymbol{\phi}}\|^2 &= \langle\boldsymbol{\phi}(\boldsymbol{x}),\boldsymbol{\phi}(\boldsymbol{x})\rangle + \langle\overline{\boldsymbol{\phi}},\overline{\boldsymbol{\phi}}\rangle - 2\langle\boldsymbol{\phi}(\boldsymbol{x}),\overline{\boldsymbol{\phi}}\rangle \\
&= \kappa(\boldsymbol{x},\boldsymbol{x}) + \frac{1}{n^2}\sum_{i,j=1}^{n}\kappa(\boldsymbol{x}_i,\boldsymbol{x}_j) - \frac{2}{n}\sum_{i=1}^{n}\kappa(\boldsymbol{x},\boldsymbol{x}_i) \\
\frac{1}{n}\sum_{k=1}^{n}\|\boldsymbol{\phi}(\boldsymbol{x}_k) - \overline{\boldsymbol{\phi}}\|^2 &= \frac{1}{n}\sum_{k=1}^{n}\kappa(\boldsymbol{x}_k,\boldsymbol{x}_k) - \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\kappa(\boldsymbol{x}_i,\boldsymbol{x}_j)
\end{aligned}
$$

The Kernel Matrix of Centered Data

- In feature-space the centered data writes:

$$\hat{\boldsymbol{\phi}}(\boldsymbol{x}) = \boldsymbol{\phi}(\boldsymbol{x}) - \overline{\boldsymbol{\phi}}$$

- The corresponding entry of the associated kernel matrix writes:

$$\begin{aligned}\hat{\kappa}(\boldsymbol{x}, \boldsymbol{x}') &= \langle \boldsymbol{\phi}(\boldsymbol{x}) - \overline{\boldsymbol{\phi}}, \boldsymbol{\phi}(\boldsymbol{x}') - \overline{\boldsymbol{\phi}} \rangle \\ &= \kappa(\boldsymbol{x}, \boldsymbol{x}') - \frac{1}{n} \sum_{i=1}^{n} \left(\kappa(\boldsymbol{x}, \boldsymbol{x}_i) - \kappa(\boldsymbol{x}', \boldsymbol{x}_i) \right) + \frac{1}{n^2} \sum_{i,j=1}^{n} \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j\end{aligned}$$

- In matrix form:

$$\widehat{\mathbf{K}} = \mathbf{K} - \frac{1}{n} \sum_{i=1}^{n} \left(\mathbb{1}\mathbb{1}^\top \mathbf{K} + \mathbf{K} \mathbb{1}\mathbb{1}^\top \right) + \frac{1}{n^2} (\mathbb{1}^\top \mathbf{K} \mathbb{1}) \mathbb{1}\mathbb{1}^\top$$

The Spread of the Data

- The $M \times n$ data matrix in feature space:

$$\mathbf{\Phi} = [\boldsymbol{\phi}(\boldsymbol{x}_1) \dots \boldsymbol{\phi}(\boldsymbol{x}_j) \dots \boldsymbol{\phi}(\boldsymbol{x}_n)]$$

- The covariance matrix *for centered data* is an $M \times M$ matrix:

$$\mathbf{C} = \frac{1}{n} \mathbf{\Phi} \mathbf{\Phi}^\top$$

- Each entry of this matrix is:

$$c_{st} = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{\phi}(\boldsymbol{x}_i)_s \; \boldsymbol{\phi}(\boldsymbol{x}_i)_t$$

The Projected Variance

- For centered data, the variance along a vector $\boldsymbol{v}$ writes:

$$\sigma_{\boldsymbol{v}}^2 = \frac{1}{n}\boldsymbol{v}^\top \boldsymbol{\Phi}\boldsymbol{\Phi}^\top \boldsymbol{v}$$

- If the data are not centered:

$$\sigma_{\boldsymbol{v}}^2 = \frac{1}{n}\boldsymbol{v}^\top \boldsymbol{\Phi}\boldsymbol{\Phi}^\top \boldsymbol{v} - \left(\frac{1}{n}\boldsymbol{v}^\top \boldsymbol{\Phi}\mathbb{1}\right)^2$$

Dual Representation of the Projected Variance

- Let's write $\boldsymbol{v}$ as a combination of the feature-space points: $\boldsymbol{v} = \sum_{i=1}^{n} \alpha_i \boldsymbol{\phi}(\boldsymbol{x}_i) = \boldsymbol{\Phi}\boldsymbol{\alpha}$.
- By substitution in the formula of the projected variance, we obtain:

$$
\begin{aligned}
\sigma_{\boldsymbol{v}}^2 &= \frac{1}{n}\boldsymbol{\alpha}^\top \boldsymbol{\Phi}^\top \boldsymbol{\Phi}\boldsymbol{\Phi}^\top \boldsymbol{\Phi}\boldsymbol{\alpha} - \left(\frac{1}{n}\boldsymbol{\alpha}^\top \boldsymbol{\Phi}^\top \boldsymbol{\Phi}\mathbb{1}\right)^2 \\
&= \frac{1}{n}\boldsymbol{\alpha}^\top \mathbf{K}^2 \boldsymbol{\alpha} - \left(\frac{1}{n}\boldsymbol{\alpha}^\top \mathbf{K}\mathbb{1}\right)^2
\end{aligned}
$$

Eigendecomposition of Covariance and Kernel Matrices

- For centred data we have:

$$\begin{aligned}\mathbf{C} &= \frac{1}{n}\mathbf{\Phi}\mathbf{\Phi}^\top \text{ with } \{(\mu_i, \boldsymbol{u}_i)\}_{i=1}^M \\ \mathbf{K} &= \mathbf{\Phi}^\top\mathbf{\Phi} \text{ with } \{(\lambda_i, \boldsymbol{v}_i)\}_{i=1}^n\end{aligned}$$

- By premultiplication of $\mathbf{\Phi}\mathbf{\Phi}^\top\boldsymbol{u} = n\mu\boldsymbol{u}$ with $\mathbf{\Phi}^\top$ we obtain:

$$\boldsymbol{v} = \mathbf{\Phi}^\top\boldsymbol{u} \text{ and } \lambda = n\mu$$

- From which we obtain: $\|\boldsymbol{v}\|^2 = \boldsymbol{u}^\top\mathbf{\Phi}\mathbf{\Phi}^\top\boldsymbol{u} = n\mu = \lambda$
- The normalized eigenvector of the kernel matrix is:

$$\boldsymbol{v} = \lambda^{-1/2}\mathbf{\Phi}^\top\boldsymbol{u}$$

- There is a similar dual expression:

$$\boldsymbol{u} = \lambda^{-1/2}\mathbf{\Phi}\boldsymbol{v}$$

Traces

- The traces are related by:

$$\mathsf{tr}(\mathbf{C}) = \frac{1}{n}\mathsf{tr}(\mathbf{K})$$

- The trace of the kernel matrix:

$$\mathsf{tr}(\mathbf{K}) = \sum_{i=1}^{n} \kappa(\boldsymbol{x}_i, \boldsymbol{x}_i)$$

- The total variance in feature-space:

$$\sum_{i=1}^{M} \mu_i = \frac{1}{n} \sum_{i=1}^{n} \kappa(\boldsymbol{x}_i, \boldsymbol{x}_i)$$

- This can be used to estimate the dimension $m \ll M$ of the reduced feature space.

Covariance Eigenvectors in Feature-space

- The eigenvectors of the covariance matrix:

$$\mathbf{U} = \left[\begin{array}{cccc} \lambda_1^{-1/2}\boldsymbol{\Phi}\boldsymbol{v}_1 & \dots & \lambda_k^{-1/2}\boldsymbol{\Phi}\boldsymbol{v}_k & \dots \end{array} \right] = \boldsymbol{\Phi}\mathbf{V}\boldsymbol{\Lambda}^{-1/2}$$

- Each eigenvector:

$$\boldsymbol{u}_k = \lambda_k^{-1/2}\boldsymbol{\Phi}\boldsymbol{v}_k = \lambda_k^{-1/2}\sum_{i=1}^{n} v_{ik}\boldsymbol{\phi}(\boldsymbol{x}_i)$$

- Let: $\boldsymbol{\beta}_k = \lambda_k^{-1/2}\boldsymbol{v}_k = (\lambda_k^{-1/2}v_{1k}\dots\lambda_k^{-1/2}v_{ik}\dots\lambda_k^{-1/2}v_{nk})$
- Hence:

$$\boldsymbol{u}_k = \sum_{i=1}^{n} \beta_{ik}\boldsymbol{\phi}(\boldsymbol{x}_i)$$

Projection of a Data Point on a Principal Direction

- Let's project a data point in feature space $\boldsymbol{\phi}(\boldsymbol{x})$ onto an eigenvector of the covariance matrix:

$$
\begin{aligned}
\boldsymbol{u}_k^\top \boldsymbol{\phi}(\boldsymbol{x}) &= \langle \boldsymbol{u}_k, \boldsymbol{\phi}(\boldsymbol{x}) \rangle \\
&= \sum_{i=1}^{n} \beta_{ik} \langle \boldsymbol{\phi}(\boldsymbol{x}_i), \boldsymbol{\phi}(\boldsymbol{x}) \rangle \\
&= \sum_{i=1}^{n} \beta_{ik} \kappa(\boldsymbol{x}_i, \boldsymbol{x})
\end{aligned}
$$

- Let $\mathbf{U}$ be the $M \times m$ matrix formed with $m \ll M$ eigenvectors of $\mathbf{C}$. A feature point can be mapped in the eigenspace of $\mathbf{C}$ with:

$$
\tilde{\boldsymbol{\phi}}(\boldsymbol{x}) = \mathbf{U}^\top \boldsymbol{\phi}(\boldsymbol{x})
$$

The Kernel PCA method

- Build the centered kernel matrix associated with a data set $\mathbf{X}$ and a kernel $\kappa(\boldsymbol{x}, \boldsymbol{x}')$:

$$\widehat{\mathbf{K}} = \mathbf{K} - \frac{1}{n} \sum_{i=1}^{n} \left(\mathbb{1}\mathbb{1}^\top \mathbf{K} + \mathbf{K}\mathbb{1}\mathbb{1}^\top \right) + \frac{1}{n^2} (\mathbb{1}^\top \mathbf{K} \mathbb{1}) \mathbb{1}\mathbb{1}^\top$$

- Compute the eigen-decomposition of this matrix and retain the K largest eigenvalue-eigenvector pairs:

$$\widehat{\mathbf{K}} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^\top$$

- Compute the vectors: $\boldsymbol{\beta}_k = \lambda_k^{-1/2} \boldsymbol{v}_k, \; k = 1 \dots K$
- Project the feature-space data onto the space spanned by the eigenvectors of the feature-space covariance:

$$\tilde{\boldsymbol{\phi}}(\boldsymbol{x}) = \mathbf{U}^\top \boldsymbol{\phi}(\boldsymbol{x}) \;\text{ with }\; \boldsymbol{u}_k^\top \boldsymbol{\phi}(\boldsymbol{x}) = \sum_{i=1}^{n} \beta_{ik} \kappa(\boldsymbol{x}_i, \boldsymbol{x})$$

Additional Topics of Interest

- Kernel K-means clustering
  http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.79.2967&rep=rep1&type=pdf
- Kernel Fisher discriminant analysis
  http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=788121
- Diffusion and exponential kernels (next lecture)