Kernel methods in computational biology

Jean-Philippe Vert

Ecole des Mines de Paris, France
Jean-Philippe.Vert@mines.org

SABRES seminar, Universite Bretagne Sud, Vannes, April 4, 2003
Outline

1. About kernels
2. What you can do with a kernel
3. Kernelizing the proteome
4. Application: comparison of a protein network and expression data
Part 1

Kernels
Definition

- Let $\mathcal{X}$ be a set (e.g., discrete)

- A kernel is a mapping $K : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ which is:
  - symmetric: $K(x, y) = K(y, x)$,
  - positive semi-definite: $\sum_{i,j} a_i a_j K(x_i, x_j) \geq 0$ for all $a_i \in \mathbb{R}$ and $x_i \in \mathcal{X}$
Example

• Suppose $\mathcal{X} = \mathbb{R}^d$. Then the following is a valid kernel:

$$K(\vec{x}, \vec{y}) = \vec{x} \cdot \vec{y}$$

• Indeed:

  * $\vec{x} \cdot \vec{y} = \vec{y} \cdot \vec{x}$
  * $\sum_{i,j} a_i a_j \vec{x}_i \cdot \vec{x}_j = \left\| \sum_i a_i \vec{x}_i \right\|^2 \geq 0$
Example: kernel in feature space

\[ K(g_i, g_j) \overset{def}{=} \vec{\Phi}(g_i) \cdot \vec{\Phi}(g_j) \]
All kernels are inner product

- If \( K(.,.) \) is a kernel, then there exists a Hilbert space \( \mathcal{H} \) and a mapping \( \Phi : \mathcal{X} \to \mathcal{H} \) such that:

\[
K(x, y) = \langle \Phi(x), \Phi(y) \rangle_{\mathcal{H}}.
\]

- Proof: by diagonalizing the kernel operator
- Second proof: by explicitly constructing such a \( \mathcal{H} \)
RKHS

- A reproducible kernel Hilbert space (RKHS) is a Hilbert space, subset of $\mathbb{R}^X$, defined as the completion of:

$$\text{span}\left\{K(x, .), s \in X\right\}.$$

- The inner product between two elements $f = \sum_i a_i K(x_i, .)$ and $g = \sum_i b_i K(x_i, .)$ is defined by:

$$< f, g >_H = \sum_{i,j} a_i b_j K(x_i, x_i).$$
Let $\Phi : \mathcal{X} \to \mathcal{H}$ defined by $\Phi(x) = K(x, \cdot)$. Then:

$$K(x, y) = \langle \Phi(x), \Phi(y) \rangle_{\mathcal{H}} = \langle K(x, \cdot), K(y, \cdot) \rangle_{\mathcal{H}}$$

For any $x \in \mathcal{X}$ and $f \in \mathcal{H}$, the following holds:

$$\langle f, K(x, \cdot) \rangle_{\mathcal{H}} = f(x).$$
RKHS (3)

- We have seen that a kernel $K$ defines a Hilbert structure on (a subset of) $\mathcal{X}^\mathbb{R}$

- **Conversely**: let $\mathcal{H}$ be a Hilbert space, subset of $\mathcal{X}^\mathbb{R}$, such that for any $x \in \mathcal{X}$ the evaluation functional $f \in \mathcal{H} \rightarrow f(x)$ be continuous

- Then there exists a kernel $K$ such that $\mathcal{H}$ be its associated RKHS.
Let $\mathcal{H}$ be a RKHS with kernel $K$, and $(x_1, \ldots, x_N) \in \mathcal{X}^N$. Then the solution of:

$$
\min_{f \in \mathcal{H}} \sum_{i=1}^{N} c(x_i, f(x_i)) + \lambda \|f\|_\mathcal{H}^2
$$

where $c : \mathcal{X} \times \mathbb{R} \to \mathbb{R}$, can always be written in the form:

$$
f(x) = \sum_{i=1}^{n} a_i K(x_i, x).
$$
Example

For a Gaussian kernel:

$$K(x, y) = \exp \left( -\frac{\|x - y\|^2}{2\sigma^2} \right),$$

the norm in RKHS is:

$$\|f\|_H^2 = \frac{1}{2\pi\sigma^2} \int |\hat{f}(\omega)|^2 \exp \left( \frac{\sigma^2 \|\omega\|^2}{2} \right) d\omega.$$
What can you do with a kernel
Overview

Let $K(x, y)$ be a given kernel. Then is it possible to perform various algorithms implicitly in the feature space (thanks to the representer theorem), such as:

- Compute the distance between points
- Principal component analysis (PCA)
- Canonical correlation analysis (CCA)
- Classification by Support vector machines (SVM)
Compute the distance between objects

\[ d(g_1, g_2)^2 = \| \Phi(g_1) - \Phi(g_2) \|^2 \]

\[ = (\Phi(g_1) - \Phi(g_2)) \cdot (\Phi(g_1) - \Phi(g_2)) \]

\[ = \Phi(g_1) \cdot \Phi(g_1) + \Phi(g_2) \cdot \Phi(g_2) - 2\Phi(g_1) \cdot \Phi(g_2) \]

\[ d(g_1, g_2)^2 = K(g_1, g_1) + K(g_2, g_2) - 2K(g_1, g_2) \]
Distance to the center of mass

Center of mass: \( \vec{m} = \frac{1}{N} \sum_{i=1}^{N} \vec{\Phi}(g_i) \), hence:

\[
\|\vec{\Phi}(g_1) - \vec{m}\|^2 = \vec{\Phi}(g_1) \cdot \vec{\Phi}(g_1) - 2 \vec{\Phi}(g_1) \cdot \vec{m} + \vec{m} \cdot \vec{m}
\]

\[
= K(g_1, g_1) - \frac{2}{N} \sum_{i=1}^{N} K(g_1, g_i) + \frac{1}{N^2} \sum_{i,j=1}^{N} K(g_i, g_j)
\]
Principal component analysis

It is equivalent to find the eigenvectors of

\[ K = \left( \Phi(g_i) \cdot \Phi(g_j) \right)_{i,j=1\ldots N} \]

\[ = \left( K(g_i, g_j) \right)_{i,j=1\ldots N} \]

Useful to project the objects on small-dimensional spaces (feature extraction).
$K_1$ and $K_2$ are two kernels for the same objects. CCA can be performed by solving the following generalized eigenvalue problem:

$$
\begin{pmatrix}
0 & K_1 K_2 \\
K_2 K_1 & 0
\end{pmatrix}
\vec{\xi} = \rho
\begin{pmatrix}
K_2^2 & 0 \\
0 & K_1^2
\end{pmatrix}
\vec{\xi}
$$

Useful to find correlations between different representations of the same objects (ex: genes, ...)

**Canonical correlation analysis**

[Diagram showing correlation analysis through CCA1 and CCA2 axes]

**CCA1**

**CCA2**
Classification: support vector machines (SVM)

Find a linear boundary with large margin and few errors

\[
\begin{align*}
\max_{\alpha} & \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j K(g_i, g_j) \\
\forall i = 1, \ldots, n & \quad 0 \leq \alpha_i \leq C \\
\sum_{i=1}^{n} \alpha_i y_i & = 0
\end{align*}
\]
Examples: SVM in bioinformatics

- **Gene functional classification from microarray**: Brown et al. (2000), Pavlidis et al. (2001)

- **Tissue classification from microarray**: Mukherje et al. (1999), Furey et al. (2000), Guyon et al. (2001)

- **Protein family prediction from sequence**: Jaakkoolo et al. (1998)

- **Protein secondary structure prediction**: Hua et al. (2001)

- **Protein subcellular localization prediction from sequence**: Hua et al. (2001)
Summary

- Once a kernel $K(x, y)$ is given, several analysis can be performed implicitly in the feature space.
- These methods are considered currently among the most powerful on many real-world problems.
- Modularity: each kernel can work with each method.
Part 3

Kernelizing the proteome
What is a gene

- a DNA sequence?
- a primary, secondary or 3D structure of a protein?
- an expression profile?
- a node in a regulatory or interaction network?
- a promoter region?
- a phylogenetic profile?
- ...

Kernel for sequences

- spectrum kernel (Eskin et al., 2002)
- Fisher kernel (Jaakkola et al., 1999)
- Pair HMM kernels (Haussler, 1999)
- Very good results for remote homology detection
Kernels for expression profiles

An expression profile is a vector $\vec{\Phi}(x)$:

- **Linear kernel**: $K(x, y) = \vec{\Phi}(x) \cdot \vec{\Phi}(y)$.

- **Polynomial kernel**: $K(x, y) = \left(\vec{\Phi}(x) \cdot \vec{\Phi}(y) + 1\right)^d$.

- **Gaussian kernel**: $K(x, y) = \exp\left(\frac{||\vec{\Phi}(x) - \vec{\Phi}(y)||^2}{2\sigma^2}\right)$. 
Diffusion kernel for the nodes of a graph (Kandor, 2001)

- Let $G$ a graph with vertices $\mathcal{X}$.
- Let $L = A - D$ be the Laplacian matrix of the graph.
- For any $\lambda > 0$, the following is a valid kernel

$$K = \exp(-\lambda L)$$
Example (1)

\[
L = \begin{pmatrix}
-1 & 0 & 1 & 0 & 0 \\
0 & -1 & 1 & 0 & 0 \\
1 & 1 & -3 & 1 & 0 \\
0 & 0 & 1 & -2 & 1 \\
0 & 0 & 0 & 0 & 1
\end{pmatrix}
\]
Example (2)

\[ K = \exp(-L) = \begin{pmatrix}
0.49 & 0.12 & 0.23 & 0.10 & 0.03 \\
0.12 & 0.49 & 0.23 & 0.10 & 0.03 \\
0.23 & 0.23 & 0.24 & 0.17 & 0.10 \\
0.10 & 0.10 & 0.17 & 0.31 & 0.30 \\
0.03 & 0.03 & 0.10 & 0.30 & 0.52 \\
\end{pmatrix} \]
More kernels

- Information diffusion kernels (Lafferty and Lebanon, 2002) for probability densities
- Kernels on finite groups (Kondor)
- Kernels for 3D structures
Operations on kernels

- The space of kernels is a closed convex cone (closed under addition, pointwise limit, multiplication by a positive scalar).
- Closed under product and tensor product.
- Linear combinations can be optimized by semi-definite programming (SDP).
- A kernel is a covariance function which defines a Gaussian process. The information geometry of Gaussian process defines a natural geometry on kernels.
Summary

- Kernels can be built from a priori knowledge, or obtained by various operations from initial kernels.

- A kernel can be thought of as a measure of similarity; this can be useful to make new kernels for any given type of data.

- Kernel engineering and kernel optimization is an active field of research currently.
Application: comparing a protein network and expression data
Genes encode proteins which can catalyse chemical reactions

Nicotinamide Mononucleotide Adenylyltransferase With Bound Nad+
Chemical reactions are often parts of pathways

From http://www.genome.ad.jp/kegg/pathway
Microarray technology monitors RNA quantity

(From Spellman et al., 1998)
Comparing gene expression and protein network

Gene network

Expression profiles

Are there “correlations”? 
Pattern of expression

- In yellow: a candidate pattern, and the correlation coefficient with each gene profile.
The correlation function with interesting patterns should vary smoothly on the graph.
Pattern relevance

- Interesting patterns involve many genes
- The projection of profiles onto an interesting pattern should capture a lot of variations among profiles
- Relevant patterns can be found by PCA
Problem

Find patterns of expression which are simultaneously

- smooth
- relevant
Pattern relevance

- Let $e(x)$ the profile of gene $x$

- Let $K_1(x, y) = e(x) \cdot e(y)$ be the linear kernel, with RKHS $H_1$.

- The norm $\| \cdot \|_{H_1}$ is a relevance functional: the relevance of $f \in H_1$ increases when the following decreases:

$$\frac{\| f \|_{H_1}}{\| f \|_{L_2}}$$
Pattern smoothness

- Let $K_2(x, y)$ be the diffusion kernel obtained from the gene network, with RKHS $H_2$.

- It can be considered as a discretized version of a Gaussian kernel (solving the heat equation with the graph Laplacian).

- The norm $\| \cdot \|_{H_2}$ is a smoothness functional: the smoother a function $f : X \rightarrow \mathbb{R}$, the larger the function:

$$\frac{\| f \|_{H_1}}{\| f \|_{L_2}}$$
Problem reformulation

Find a linear function $f_1$ and a function $f_2$ such that:

- $f_1$ be relevant: $\frac{\|f_1\|_{L^2}}{\|f_1\|_{H_1}}$ be large

- $f_2$ be smooth: $\frac{\|f_2\|_{L^2}}{\|f_2\|_{H_2}}$ be large

- $f_1$ and $f_2$ be correlated: 

$$\frac{f_1 \cdot f_2}{\|f_1\|_{L^2} \|f_2\|_{L^2}}$$

be large
Problem reformulation (2)

The three goals can be combined in the following problem:

\[
\max_{f_1, f_2} \frac{f_1 \cdot f_2}{\left(\|f_1\|^2_{L^2} + \delta \|f_1\|^2_{H^1}\right)^{1/2} \left(\|f_2\|^2_{L^2} + \delta \|f_2\|^2_{H^2}\right)^{1/2}}
\]

where the parameter \( \delta \) controls the trade-off between relevance/smoothness on the one hand, correlation on the other hand.
Solving the problem

This formulation is equivalent to a generalized form of CCA (Kernel-CCA, Bach and Jordan, 2002), which is equivalent to the following generalized eigenvector problem

\[
\begin{pmatrix}
0 & K_1 K_2 \\
K_2 K_1 & 0
\end{pmatrix}
\begin{pmatrix}
\alpha \\
\beta
\end{pmatrix}
= \rho \begin{pmatrix}
K_1^2 + \delta K_1 & 0 \\
0 & K_2^2 + \delta K_2
\end{pmatrix}
\begin{pmatrix}
\alpha \\
\beta
\end{pmatrix}
\]
Summary

Diffusion kernel

Kernel CCA

Linear kernel
Data

- **Gene network**: two genes are linked if the catalyze successive reactions in the KEGG database.

- **Expression profiles**: 18 time series measures for the 6,000 genes of yeast, during two cell cycles.
First pattern of expression

Expression

Time
Related metabolic pathways

50 genes with highest $s_2 - s_1$ belong to:

- Oxidative phosphorylation (10 genes)
- Citrate cycle (7)
- Purine metabolism (6)
- Glycerolipid metabolism (6)
- Sulfur metabolism (5)
- Selenoaminoacid metabolism (4), etc...
Related genes
Related genes
Related genes
Opposite pattern

Expression vs. Time
Related genes

- RNA polymerase (11 genes)
- Pyrimidine metabolism (10)
- Aminoacyl-tRNA biosynthesis (7)
- Urea cycle and metabolism of amino groups (3)
- Oxidative phosphorylation (3)
- ATP synthesis (3), etc...
Related genes
Related genes
Related genes
Conclusion
Conclusion

- There is an urgent need for formalisms and computational tools to integrate heterogeneous data.

- **Kernel methods** offer such a framework.

- Few conceptual relationships between genes, but computational efficiency.

- Machine learning and kernel methods are currently **boosted by biology**.