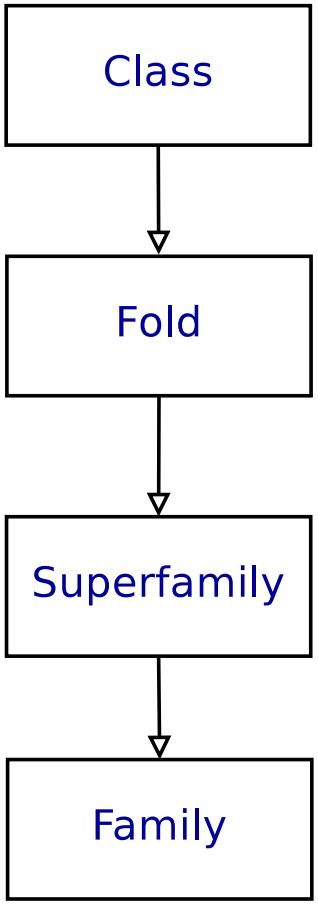
Kernels for Biological Sequences

Chloé-Agathe Azencott

Institute for Genomics and Bioinformatics
Bren School of Information and Computer Sciences
University of California, Irvine

Remote Protein Homology Detection

SCOP 1.53 Structural Classification of Proteins



Major structural similarity (shared secondary structures)

Probable common evolutionary origin (shared structural and functional features)

Clear common evolutionary origin (identical functionality and/or high sequence identity)

- 4 352 sequences grouped into 54 families
- → Detect homologies between proteins from the same super-family, but not necessarily the same family

G. Murzin, S. E. Brenner, T. Hubbard and C. Chothia.

SCOP: A Structural Classification of Proteins Database for the Investigation of Sequences and Structures.

J. Mol. Biol. 247:536-540, 1995

Approaches

- Local Alignment algorithms (Smith Waterman, BLAST, FASTA)
- Generative models (HMM, PSI-BLAST)
- Discriminative methods: SVMs

Local Alignment

Local Alignment

An alignment π of p positions between two sequences x and y is a pair of p-tuples:

$$\pi = ((\pi_1(1), \ldots, \pi_1(p)), (\pi_2(1), \ldots, \pi_2(p)))$$

such that

$$1 \leq \pi_1(1) < \pi_1(2) < \cdots < \pi_1(p) \leq |x|,$$

$$1 \leq \pi_2(1) < \pi_2(2) < \cdots < \pi_2(p) \leq |y|,$$

The $\pi_1(i)$th letter of x is aligned to the $\pi_2(i)$th letter of y
E.g.

$$\begin{cases} x = \texttt{GATCCAGG} \\ y = \texttt{GTTCAGT} \\ \pi = ((1,2,4,6),(1,3,4,5)) \end{cases}$$

```
G-ATCCAGG
GTT-C-AT-
```

Local Alignment Score

- substitution matrix $S \in \mathbb{R}^{\mathcal{A} \times \mathcal{A}}$
- gap penalty function $g : \mathbb{N} \to \mathbb{R}$, $g(0) = 0$

$$s_{S,g}(\pi) = \sum_{i=1}^{p} S(x_{\pi_1(i)}, y_{\pi_2(i)}) - \sum_{i=1}^{p-1} g(x_{\pi_1(i+1)} - x_{\pi_1(i)}) + g(x_{\pi_2(i+1)} - x_{\pi_2(i)})$$

Smith-Waterman score:

$$SW_{S,g}(x, y) = \max_{\pi \in \Pi(x, y)} s_{S,g}(\pi)$$

Smith and Waterman

$$M(i,0) = M(0,j) = 0$$

$$M(i,j) = \max \begin{cases} 0 & \\ M(i-1,j-1) + S(x_i, y_j) & \text{Match/Mismatch} \\ M(i-1,j) + g(x_i, -) & \text{Deletion} \\ M(i,j-1) + g(-, y_j) & \text{Insertion} \end{cases}$$

E.g: $S(x,y) = 2$ if $x = y$ and -1 otherwise

	-	G	A	C	T	A	G	T	T
-	0	0	0	0	0	0	0	0	0
G	0	2	1	0	0	0	2	1	0
A	0	1	4	3	2	2	1	1	0
G	0	2	3	3	2	1	4	3	2
A	0	1	4	3	2	4	3	3	2
G	0	2	3	3	2	3	6	5	4
T	0	1	2	2	5	4	5	8	7

Best alignment:

```
GACTAGTT
GA-GAGT-
```

T. Smith and M. Waterman.
Identification of Common Molecular Subsequences
J. Mol. Bio., 147:195-197.

Kernel Methods

(Positive Definite) Kernels

$k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ such that :

$$\forall (x, y) \in \mathcal{X} \times \mathcal{X},\ k(x, y) = k(y, x);$$

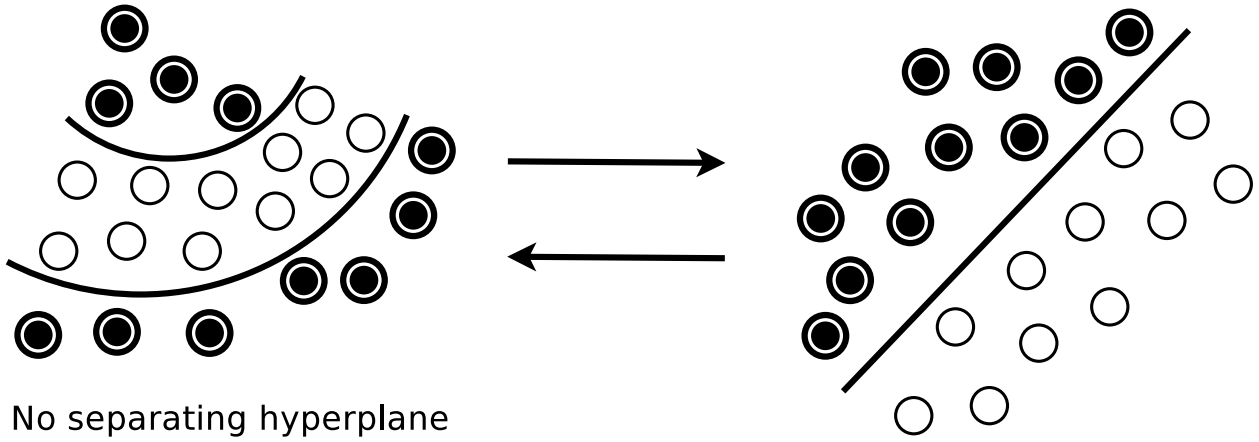
and

$$\forall n \in \mathbb{N}^*, \forall x_1, \ldots, x_n \in \mathcal{X}, \forall c_1, \ldots, c_n \in \mathbb{R},\ \sum_{i=1}^{n} \sum_{j=1}^{n} c_i c_j k(x_i, x_j) \geq 0$$

$\forall k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$,
$\exists (\mathcal{F}, \langle , \rangle)$ Hilbert space and $\phi : \mathcal{X} \rightarrow \mathcal{F}$ such that:
$\forall (x, y) \in \mathcal{X} \times \mathcal{X},\ k(x, y) = \langle \phi(x), \phi(y) \rangle$
$\mathcal{F}$ is the feature space.

Support Vector Machines

Linear separation in a feature space



No separating hyperplane
in the input space

Separating hyperplane
in the feature space

$$f(x) = \sum_{i=1}^{M} \alpha_i k(x, x_i) + b$$

String Kernels Based on Generative Models

Fisher's Kernel

- Learn a generative model per family (HMM)
- Fisher's score

$$U_x = \nabla_\theta \log P(x|H,\theta)$$

- Fisher's kernel

$$k(x,y) = \frac{1}{2}(U_x - U_y)^T F^{-1}(U_x - U_y)$$

- Fisher information matrix F: covariance matrix of the scores U_x (x sampled from $P(x|H,\theta)$)

$$F = \int_x U_x U_x^T P(x|H,\theta) dx$$

T. Jaakola, M. Diekhans and D. Haussler.
A Discriminative Framework for Detecting Remote Protein Homologies
Journal of Computational Biology, 7(1):95-114, 2000.

Pairwise Kernel (Liao)

- Family Pairwise Search: sequence vs. family comparison
- Pairwise score:

$$F_x = f_{x1}, f_{x2}, \ldots, f_{x_n}$$

where $x_1, \ldots, x_n$ are proteins in the family and

$$f_{x_i} = log(\text{pvalue}(SW(x, x_i)))$$

- Pairwise kernel

$$k(x, y) = \frac{\langle F_x, F_y \rangle}{\sqrt{\langle F_x, F_x \rangle \langle F_y, F_y \rangle}}$$

normalized to

$$K(x, y) = 1 + e^{-\frac{k(x,x)+k(y,y)-2k(x,y)}{2\sigma^2}}$$

L. Liao and W. S. Noble.

Combining Pairwise Sequence Similarity and Support Vector Machines for Detecting Remote Protein Evolutionary and Structural Relationships.

Journal of Computational Biology, 10(6): 857-868, 2003.

More Generic String Kernels

Lodhi's String Kernel

- α subsequence of x, $\alpha = x[i]$

$$i = (i_1, \ldots, i_{|\alpha|}) : \alpha_j = x_{i_j}$$

- Length of the subsequence u in x: $l(i) = i_{|\alpha|} - i_1 + 1$
- Feature vector

$$\phi_\alpha(x) = \sum_{i:\alpha=x[i]} \lambda^{l(i)}$$

  where $\lambda \leq 1$
- String kernel

$$K_n(x, y) = \sum_{\alpha \in \mathcal{A}^n} \langle \phi_\alpha(x), \phi_\alpha(y) \rangle$$

H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini and C. Watkins.
Text Classification Using String Kernels.
The Journal of Machine Learning Research, 2:419-444, 2002.

Spectral Kernels

- Define *feature vectors* that record the presence/absence (or number of occurrences) of particular substructures in a given structure
- $$\phi(x) = (\phi_s(x))_{s \text{ substructure}}$$

  where
  $$\phi_s(x) = \begin{cases} 1 & \text{if } s \text{ occurs in } x \\ 0 & \text{otherwise} \end{cases}$$
- Then, define a *similarity* between these feature vectors

Leslie's Spectrum Kernel

$$\phi_l(x) = (\phi_\alpha(x))_{\alpha \in \mathcal{A}^l}$$

$$k_l(x, y) = \langle \phi_l(x),\ \phi_l(y) \rangle$$

C. Leslie and E. Eskin.
The Spectrum Kernel: A String Kernel for SVM Protein Classification.
Proceedings of the Pacific Symposium on Biocomputing, 7:566-575, 2002.

Leslie's Mismatch Kernel

$$\phi_\beta^{mis}(\alpha) = \begin{cases} 1 & \text{if } \beta \text{ differs from } \alpha \text{ by at most } m \text{ mismatches} \\ 0 & \text{otherwise} \end{cases}$$

$$\phi_{l,m}^{mis}(x) = \sum_{\alpha \in l-sp(x)} \left(\phi_\beta^{mis}(\alpha) \right)_{\beta \in \mathcal{A}^l}$$

$$k_l : x, y \mapsto \left\langle \phi_{l,m}^{mis}(x), \phi_{l,m}^{mis}(y) \right\rangle$$

C. Leslie and R. Kuang

Fast Kernels for Inexact String Matching.

Learning Theory and Kernel Machines: COLT/Kernel, 114, 2001.

Leslie's Gappy Kernel

$$\phi_{\beta}^{gap}(\alpha) = \begin{cases} 1 & \text{if } \beta \text{ of occurs in } \alpha \text{ with at most } g \text{ gaps} \\ 0 & \text{otherwise} \end{cases}$$

$$\phi_{g,l}^{gap}(x) = \sum_{\alpha \in l-sp(x)} \left(\phi_{\beta}^{gap}(\alpha) \right)_{\beta \in \mathcal{A}^l}$$

$$k_l(x,y) = \left\langle \phi_{g,l}^{gap}(x), \phi_{g,l}^{gap}(y) \right\rangle$$

Leslie's Substitution Kernel

$$\phi_{\beta}^{sub}(\alpha) = \begin{cases} 1 & \text{if } \beta \text{ is such that } -\sum_{i=1}^{m} logP(a_i|b_i) < \sigma \\ 0 & \text{otherwise} \end{cases}$$

where $\alpha = (a_1 \ldots a_m)$ and $\beta = (b_1 \ldots b_m)$

$$\phi_{l,\sigma}^{sub}(x) = \sum_{\alpha \in l-sp(x)} \left(\phi_{\beta}^{sub}(\alpha)\right)_{\beta \in \mathcal{A}^l}$$

$$k_l(x,y) = \left\langle \phi_{l,\sigma}^{sub}(x),\ \phi_{l,\sigma}^{sub}(y) \right\rangle$$

Leslie's Wildcard Kernel

$$\phi^{wild}_{\beta}(\alpha) = \lambda^{j}$$

where α matches β containing j wildcards and $0 < \lambda \leq 1$

$$\phi^{wild}_{l,m,\sigma} : x \mapsto \sum_{\alpha \in l-sp(x)} (\phi^{wild}_{\beta}(\alpha))_{\beta \in \mathcal{W}}$$

$$k_l(x,y) = \left\langle \phi^{sub}_{l,m,\sigma}(x),\ \phi^{sub}_{l,m,\sigma}(y) \right\rangle$$

The Local Alignment Kernel

- Similarity between biological sequences: local alignment
- Smith-Waterman score is *not* a kernel in the general case
- → Define a kernel such that two strings are similar when they have many high-scoring local alignments

Convolution Kernel

$$k_1 * k_2(x,\ y) = \sum_{x_1 x_2 = x; y_1 y_2 = y} k_1(x_1,\ y_1) k_2(x_2,\ y_2)$$

If k1 and k2 are kernels, then k is a kernel

D. Haussler.

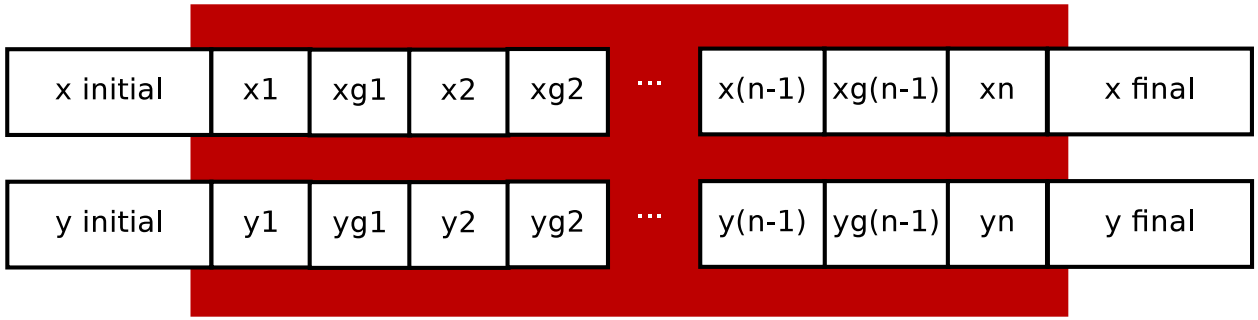
Convolution Kernels on Discrete Structures.
Technical Report, University of California, Santa Cruz, 1999.

Local Alignment Kernel

- $k_0(x,\ y) = 1$
- $k_a(x,\ y) = 0$ if $|x| \neq 1$ or $|y| \neq 1$,
- $k_a^{(\beta)}(x,\ y) = exp(\beta S(x,\ y))$ otherwise
- $k_g^{(\beta)}(x,\ y) = exp(\beta(g(|x|) + g(|y|)))$

Local alignment of x and y based on the local alignment of i residues:

$$k_0^{(\beta)} = k_0 \text{ and } k_i^{(\beta)} = k_i * (k_{a^{(\beta)}} * k_g^{(\beta)})^{i-1} * k_a * k_0$$



Local Alignment Kernel

Local alignment kernel:

$$k_{LA}^{(\beta)} = \sum_{i=0}^{\infty} k_i^{(\beta)}$$

Converges because finite number of non-zero terms

H. Saigo, J.-P. Vert, N. Ueda and T. Akutsu

Protein Homology Detection Using String Alignment Kernels

Bioinformatics, 20(11):1682-1689, 2004.

Local Alignment Kernel and Smith Waterman

$$k_{LA}^{(\beta)}(x,y) = \sum_{\pi \in \Pi(x,y)} exp(\beta s_{S,g}(x,y,\pi)))$$

$$SW(x,y) = \max_{\pi \in \Pi(x,\ y)} s_{S,g}(x,y,\pi)$$

$$\lim_{\beta \to \infty} \frac{1}{\beta} ln(k_{LA}^{(\beta)}(x,y)) = SW(x,y)$$

Computation by Dynamic Programming

If the gap penalty is affine

$$\begin{cases} g(0) = 0 \\ g(n) = d + e(n-1) \text{ if } n \geq 1 \end{cases}$$

then

$$k_{LA}^{(\beta)}(x, y) = 1 + X_2(|x|, |y|) + Y_2(|x|, |y|) + M(|x|, |y|)$$

$$M(i,0) = M(0,j) = X(i,0) = X(0,j) = Y(i,0) = Y(0,j) = 0$$
$$X_2(i,0) = X_2(0,j) = Y_2(i,0) = Y_2(0,j) = 0$$

$$\begin{cases} M(i,j) = e^{\beta S(x_i, y_j)}[1 + X(i-1,j-1) + Y(i-1,j-1) + M(i-1,j-1)] \\ X(i,j) = e^{\beta d} M(i-1,j) + e^{\beta e} X(i-1,j) \\ Y(i,j) = e^{\beta d}[M(i,j-1) + X(i,j-1)] + e^{\beta e} Y(i,j-1) \\ X_2(i,j) = M(i-1,j) + X_2(i-1,j) \\ Y_2(i,j) = M(i,j-1) + X_2(i,j-1) + Y_2(i,j-1) \end{cases}$$

Smith-Waterman: replace additions by max operation, take the log of the result devided by β.

Diagonal Dominance

$$k_{LA}^{(\beta)}(x,x) >> k_{LA}^{(\beta)}(x,y)$$

Correction:

$$\tilde{k}_{LA}^{(\beta)}(x,y) = \frac{1}{\beta} ln(k_{LA}^{(\beta)}(x,y))$$

- Not a kernel
- Correct to make positive definite over the training set
  - Eigenvalues
    substract the smallest negative eigenvalue of the training Gram matrix from the diagonal
  - Empirical Kernel Map

$$k_{LA-ekm}(x,y) = \sum_{i=1}^{n} \tilde{k}_{LA}^{(\beta)}(x,x_i)\tilde{k}_{LA}^{(\beta)}(y,x_i)$$

B. Schölkopf and A. J. Smola.
Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond.
MIT Press, Cambridge, MA, 2002.

Results

- substitution matrix: BLOSUM62
- gap penalty

$$\begin{cases} g(0) = 0 \\ g(n) = 11 + (n-1) \text{ if } n \geq 1 \end{cases}$$

- for the mismatch kernel: $l = 5$, $m = 1$

Performance of the Local Alignment Kernel

Kernel	Mean ROC	Mean ROC50	Mean mRFP
LA-eig ($\beta = +\infty$)	0.908	0.591	0.0654
LA-eig ($\beta = 1$)	0.912	0.612	0.0626
LA-eig ($\beta = 0.8$)	0.908	0.597	0.0679
LA-eig ($\beta = 0.5$)	**0.925**	**0.649**	0.0541
LA-eig ($\beta = 0.2$)	0.923	**0.661**	0.0637
LA-eig ($\beta = 0.1$)	0.868	0.429	0.111
LA-ekm ($\beta = +\infty$)	0.916	0.585	0.0580
LA-ekm ($\beta = 1$)	0.920	0.587	**0.0539**
LA-ekm ($\beta = 0.8$)	0.916	0.585	0.0592
LA-ekm ($\beta = 0.5$)	**0.929**	0.600	**0.0515**
LA-ekm ($\beta = 0.2$)	0.877	0.453	0.125
LA-ekm ($\beta = 0.1$)	0.596	0.052	0.500
Pairwise	0.896	0.464	0.0837
Mismatch	0.872	0.400	0.0837
Fisher	0.773	0.250	0.204

The LA-eig and LA-ekm kernels with $\beta = +\infty$ correspond to the SW score (modified to become positive definite on the set of proteins used to train the SVM). Bold numbers indicate the best results in each column.

Performance of the Local Alignment Kernel