Support vector machine prediction of signal peptide cleavage site using a new class of kernels for strings

Jean-Philippe Vert

Bioinformatics Center, Kyoto University, Japan
Outline

1. SVM and kernel methods
2. New kernels for bioinformatics
3. Example: signal peptide cleavage site prediction
Part 1

SVM and kernel methods
Support vector machines

- Objects to classified $x$ mapped to a feature space
- Largest margin separating hyperplane in the feature space
The kernel trick

- Implicit definition of $x \rightarrow \Phi(x)$ through the kernel:

$$K(x, y) \overset{def}{=} \langle \Phi(x), \Phi(y) \rangle$$
The kernel trick

- Implicit definition of $x \rightarrow \Phi(x)$ through the kernel:

\[ K(x, y) \overset{def}{=} \langle \Phi(x), \Phi(y) \rangle \]

- Simple kernels can represent complex $\Phi$
The kernel trick

- Implicit definition of $x \rightarrow \Phi(x)$ through the kernel:

$$K(x, y) \overset{def}{=} <\Phi(x), \Phi(y)>$$

- Simple kernels can represent complex $\Phi$

- For a given kernel, not only SVM but also clustering, PCA, ICA... possible in the feature space $= \text{kernel methods}$
Kernel examples

- “Classical” kernels: polynomial, Gaussian, sigmoid...
Kernel examples

• “Classical” kernels: polynomial, Gaussian, sigmoid… but the objects $x$ must be vectors
Kernel examples

- “Classical” kernels: polynomial, Gaussian, sigmoid... but the objects $x$ must be vectors

- “Exotic” kernels for strings:
Kernel examples

- “Classical” kernels: polynomial, Gaussian, sigmoid... but the objects $x$ must be vectors

- “Exotic” kernels for strings:
  - Fisher kernel (Jaakkola and Haussler 98)
Kernel examples

- “Classical” kernels: polynomial, Gaussian, sigmoid… but the objects $x$ must be vectors

- “Exotic” kernels for strings:
  - Fisher kernel (Jaakkola and Haussler 98)
  - Convolution kernels (Haussler 99, Watkins 99)
Kernel examples

- “Classical” kernels: polynomial, Gaussian, sigmoid… but the objects $x$ must be vectors

- “Exotic” kernels for strings:
  - Fisher kernel (Jaakkoola and Haussler 98)
  - Convolution kernels (Haussler 99, Watkins 99)
  - Kernel for translation initiation site (Zien et al. 00)
Kernel examples

- “Classical” kernels: polynomial, Gaussian, sigmoid… but the objects $x$ must be vectors
- “Exotic” kernels for strings:
  - Fisher kernel (Jaakkola and Haussler 98)
  - Convolution kernels (Haussler 99, Watkins 99)
  - Kernel for translation initiation site (Zien et al. 00)
  - String kernel (Lodhi et al. 00)
Kernel engineering

Use prior knowledge to build the geometry of the feature space through $K(.,.)$
Part 2

New kernels for bioinformatics
The problem

- $\mathcal{X}$ a set of objects
The problem

- $\mathcal{X}$ a set of objects
- $p(x)$ a probability distribution on $\mathcal{X}$
The problem

- $\mathcal{X}$ a set of objects
- $p(x)$ a probability distribution on $\mathcal{X}$
- How to build $K(x, y)$ from $p(x)$?
Product kernel

\[ K_{\text{prod}}(x, y) = p(x)p(y) \]
Product kernel

\[ K_{prod}(x, y) = p(x)p(y) \]
Product kernel

$$K_{prod}(x, y) = p(x)p(y)$$

SVM = Bayesian classifier
Diagonal kernel

\[ K_{diag}(x, y) = p(x) \delta(x, y) \]
Diagonal kernel

\[ K_{diag}(x, y) = p(x) \delta(x, y) \]
Diagonal kernel

\[ K_{\text{diag}}(x, y) = p(x)\delta(x, y) \]

No learning
Interpolated kernel

If objects are composite: $x = (x_1, x_2)$:

$$K(x, y) = K_{\text{diag}}(x_1, y_1) K_{\text{prod}}(x_2, y_2)$$
Interpolated kernel

If objects are composite: \( x = (x_1, x_2) \):

\[
K(x, y) = K_{\text{diag}}(x_1, y_1)K_{\text{prod}}(x_2, y_2) \\
= p(x_1)\delta(x_1, y_1) \times p(x_2|x_1)p(y_2|y_1)
\]
General interpolated kernel

- Composite objects $x = (x_1, \ldots, x_n)$
General interpolated kernel

• Composite objects $x = (x_1, \ldots, x_n)$

• A list of index subsets: $\mathcal{V} = \{I_1, \ldots, I_v\}$ where $I_i \subset \{1, \ldots, n\}$
General interpolated kernel

- Composite objects $x = (x_1, \ldots, x_n)$

- A list of index subsets: $\mathcal{V} = \{I_1, \ldots, I_v\}$ where $I_i \subset \{1, \ldots, n\}$

- Interpolated kernel:

$$K_\mathcal{V}(x, y) = \frac{1}{|\mathcal{V}|} \sum_{I \in \mathcal{V}} K_{\text{diag}}(x_I, y_I) K_{\text{prod}}(x_{I^c}, y_{I^c})$$
Rare common subparts

For a given $p(x)$ and $p(y)$, we have:

$$K_{\mathcal{V}}(x, y) = K_{\text{prod}}(x, y) \times \frac{1}{|\mathcal{V}|} \sum_{I \in \mathcal{V}} \frac{\delta(x_I, y_I)}{p(x_I)}$$
Rare common subparts

For a given $p(x)$ and $p(y)$, we have:

$$K_V(x, y) = K_{prod}(x, y) \times \frac{1}{|V|} \sum_{I \in V} \frac{\delta(x_I, y_I)}{p(x_I)}$$

$x$ and $y$ get closer in the feature space when they share rare common subparts
Implementation

- Factorization for particular choices of $p(.)$ and $\mathcal{V}$
Implementation

- Factorization for particular choices of \( p(.) \) and \( \mathcal{V} \)
- Example:
  \[ \mathcal{V} = \mathcal{P}(\{1, \ldots, n\}) \]  the set of all subsets:  \( |\mathcal{V}| = 2^n \)
Implementation

- Factorization for particular choices of \( p(.) \) and \( \mathcal{V} \)
- Example:
  - \( \mathcal{V} = \mathcal{P} \left( \{1, \ldots, n\} \right) \) the set of all subsets: \(|\mathcal{V}| = 2^n\)
  - product distribution \( p(x) = \prod_{j=1}^{n} p_j(x_j) \).
Implementation

- Factorization for particular choices of $p(.)$ and $\mathcal{V}$

- Example:
  - $\mathcal{V} = \mathcal{P}(\{1, \ldots, n\})$ the set of all subsets: $|\mathcal{V}| = 2^n$
  - Product distribution $p(x) = \prod_{j=1}^{n} p_j(x_j)$.
  - Implementation in $O(n)$ because

$$\sum_{I \in \mathcal{V}} (\ldots) = \prod_{i=1}^{n} (\ldots)$$
Part 3

Application:
SVM prediction of signal peptide cleavage site
Secretory pathway

- Nascent protein
- ER
- Golgi
- Signal peptide
- mRNA
- Cell surface (secreted)
- Lysosome
- Plasma membrane
- Nucleus
- Chloroplast
- Mitochondrion
- Peroxisome
- Cytosole
## Signal peptides

<table>
<thead>
<tr>
<th>Protein</th>
<th>-1</th>
<th>+1</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td><strong>MKANAKTI</strong>IIGMIALAISHTAMA</td>
<td>EE...</td>
</tr>
<tr>
<td>(2)</td>
<td><strong>MKQSTIALALLPLLFTPVTKA</strong></td>
<td>RT...</td>
</tr>
<tr>
<td>(3)</td>
<td><strong>MKATKLVLGAVILGSTLLAG</strong></td>
<td>CS...</td>
</tr>
</tbody>
</table>

### Signal peptides

<table>
<thead>
<tr>
<th>Protein</th>
<th>-1</th>
<th>+1</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>MKANAKTIIGMIALAISHTAMA</td>
<td>EE...</td>
</tr>
<tr>
<td>(2)</td>
<td>MKQSTIALALLPLLFTPVTKA</td>
<td>RT...</td>
</tr>
<tr>
<td>(3)</td>
<td>MKATKLVLGAVALGSTLLAG</td>
<td>CS...</td>
</tr>
</tbody>
</table>


- **6-12** hydrophobic residues (in yellow)
- **(-3,-1)**: small uncharged residues


**Experiment**

- Challenge: classification of aminoacids windows, positive if cleavage occurs between -1 and +1:

\[ [x_{-8}, x_{-7}, \ldots, x_{-1}, x_1, x_2] \]
Experiment

- Challenge: classification of aminoacids windows, positive if cleavage occurs between -1 and +1:

\[
[ x_{-8}, x_{-7}, \ldots, x_{-1}, x_1, x_2 ]
\]

- 1,418 positive examples, 65,216 negative examples
Experiment

- Challenge: classification of aminoacids windows, positive if cleavage occurs between -1 and +1:
  \[ [x_{-8}, x_{-7}, \ldots, x_{-1}, x_1, x_2] \]

- 1,418 positive examples, 65,216 negative examples

- Computation of a weight matrix:
  SVM + \( K_{\text{prod}} \) (naive Bayes) vs SVM + \( K_{\text{interpolated}} \)
Result: ROC curves

- Interpolated Kernel
- Product Kernel (Bayes)
Conclusion
Conclusion

- An other way to derive a kernel from a probability distribution
Conclusion

- An other way to derive a kernel from a probability distribution
- Useful when objects can be compared by comparing subparts
Conclusion

• An other way to derive a kernel from a probability distribution

• Useful when objects can be compared by comparing subparts

• Encouraging result on real-world application’ “how to improve a weight matrix based classifier”
Conclusion

- An other way to derive a kernel from a probability distribution
- Useful when objects can be compared by comparing subparts
- Encouraging result on real-world application’ “how to improve a weight matrix based classifier”
- Future work: more application-specific kernels
Acknowledgement

• Minoru Kanehisa

• Applied Biosystems for the travel grant