# Machine learning for ligand-based virtual screening and chemogenomics

Jean-Philippe Vert

Institut Curie - INSERM U900 - Mines ParisTech

*In silico discovery of molecular probes and drug-like compounds: Success & Challenges*
*INSERM workshop, Saint-Raphaël, France, March 25, 2010*

1909 100 ans 2009
institutCurie
Ensemble, prenons le cancer de vitesse.

Inserm
Institut national
de la santé et de la recherche médicale

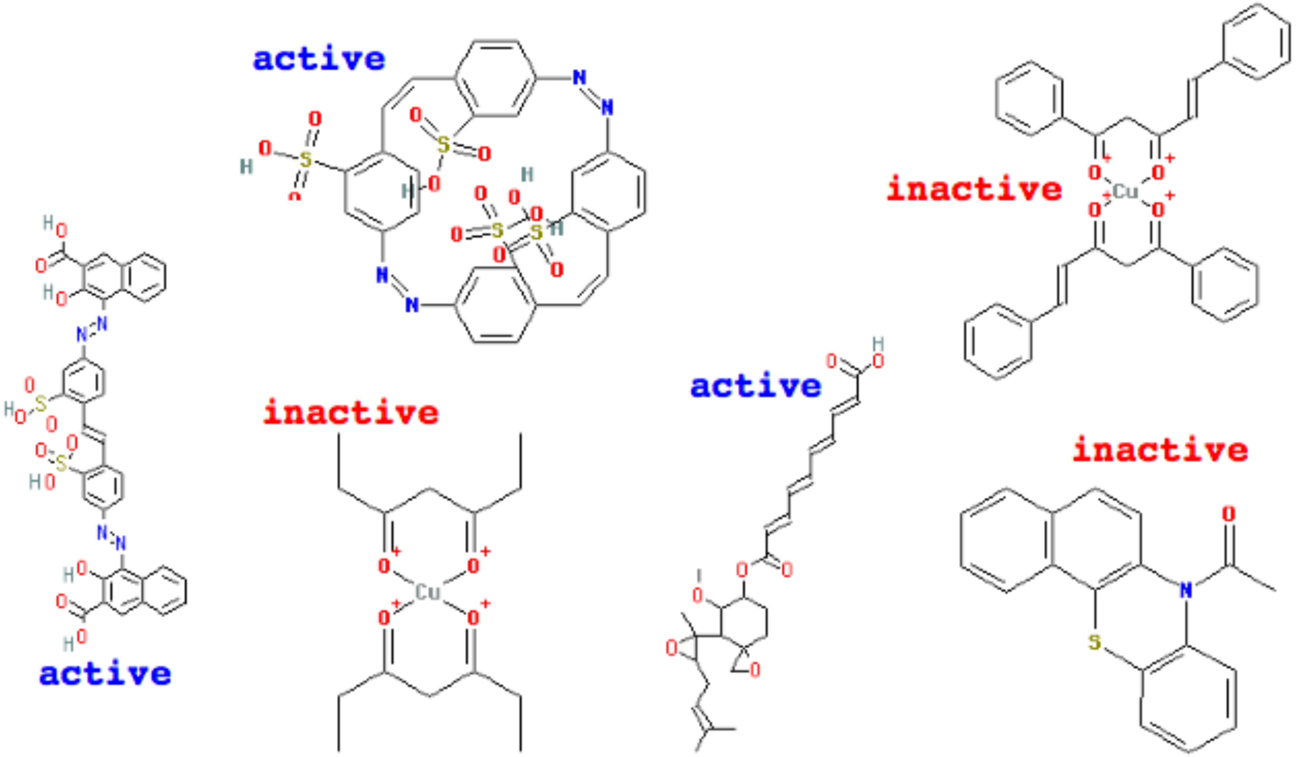
MINES ParisTech

# Outline

1. Machine learning for ligand-based virtual screening
2. 2D kernels
3. 3D kernels
4. Towards *in silico* chemogenomics

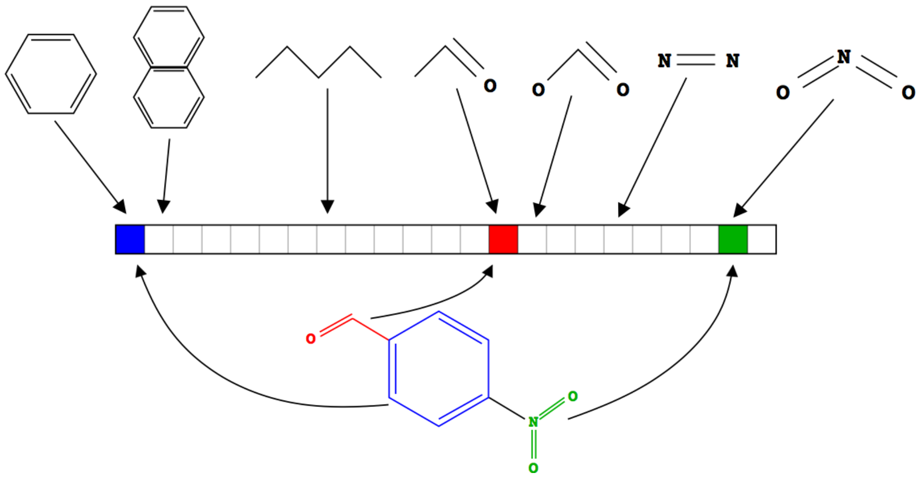
# Machine learning for ligand-based virtual screening

# Ligand-based virtual screening / QSAR

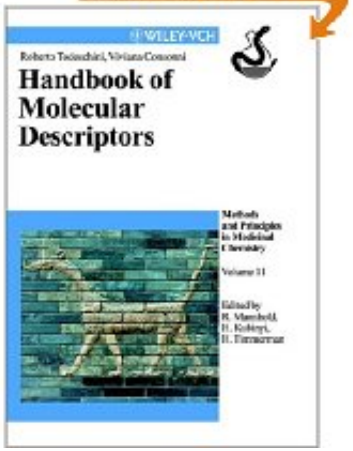


*From http://cactus.nci.nih.gov*

# Represent each molecule as a vector…



Cliquez pour Feuilleter!



WILEY-VCH

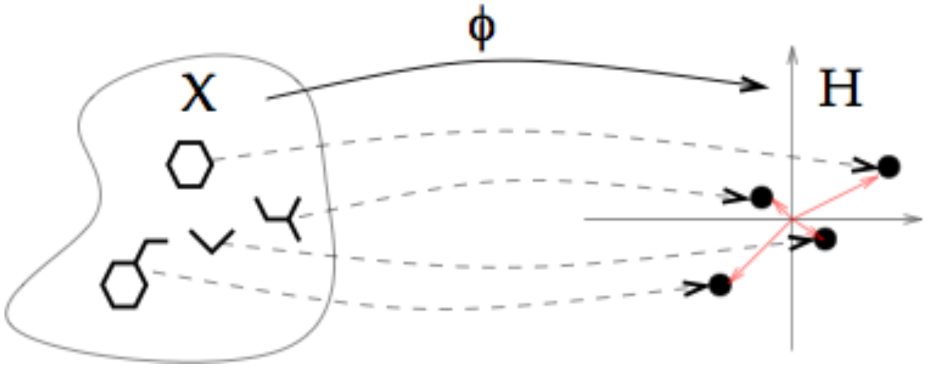
Roberto Todeschini, Viviana Consonni

Handbook of Molecular Descriptors

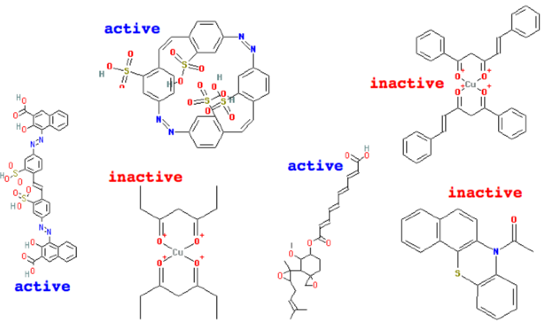
Methods and Principles in Medicinal Chemistry

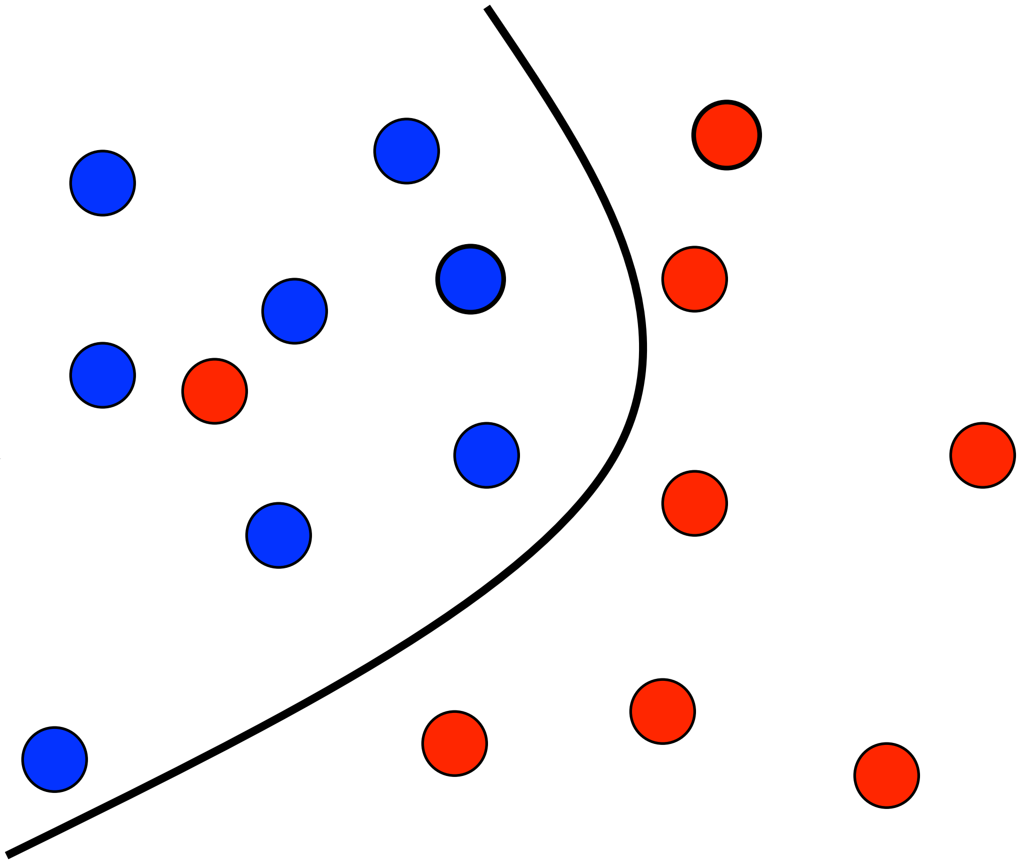
Volume 11

Edited by R. Mannhold, H. Kubinyi, H. Timmerman



# …and discriminate with machine learning





-LDA
-PLS
-Neural network
-Decision trees
-Nearest neighbour
-SVM, …

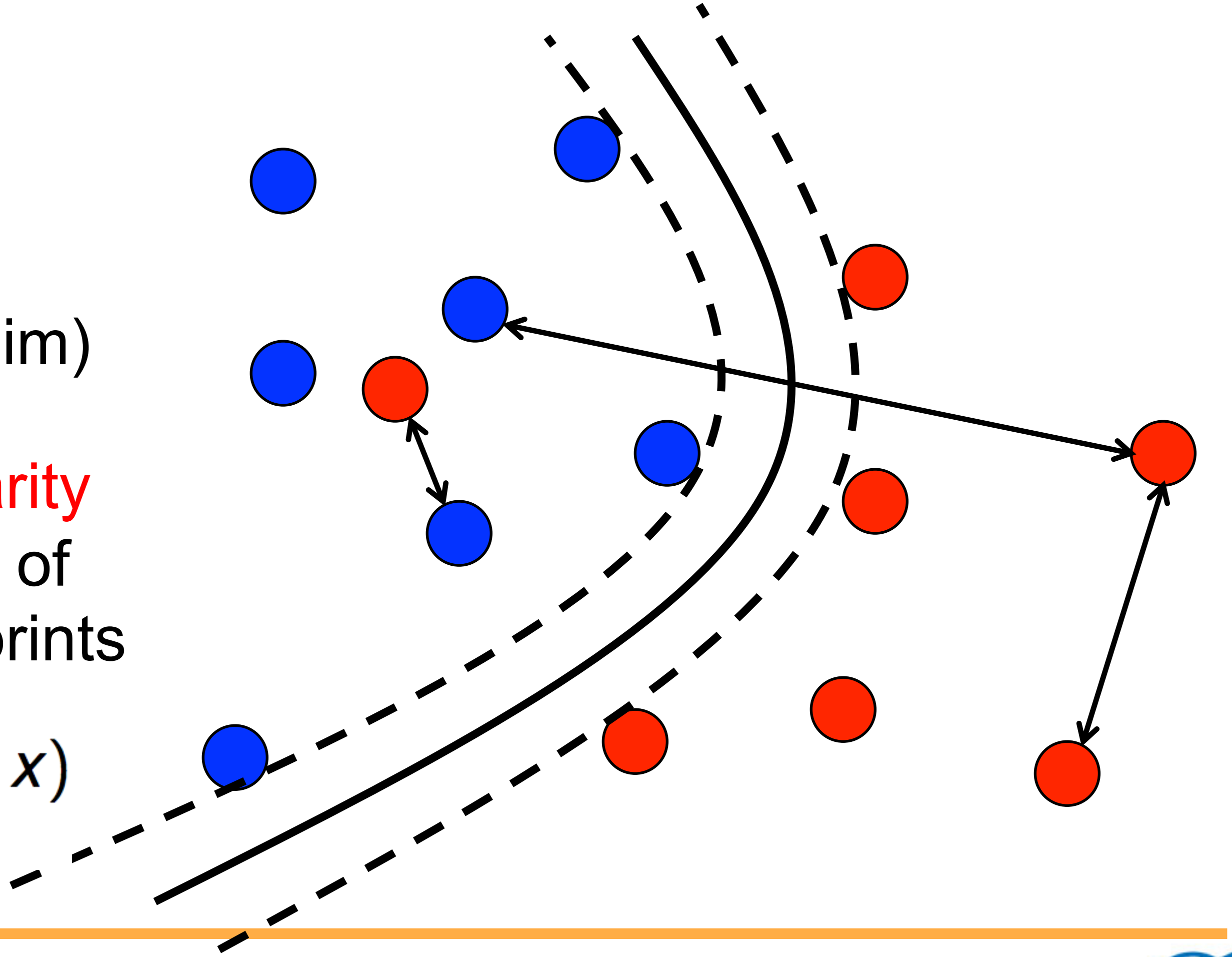
# Support Vector Machine (SVM)

- Nonlinear

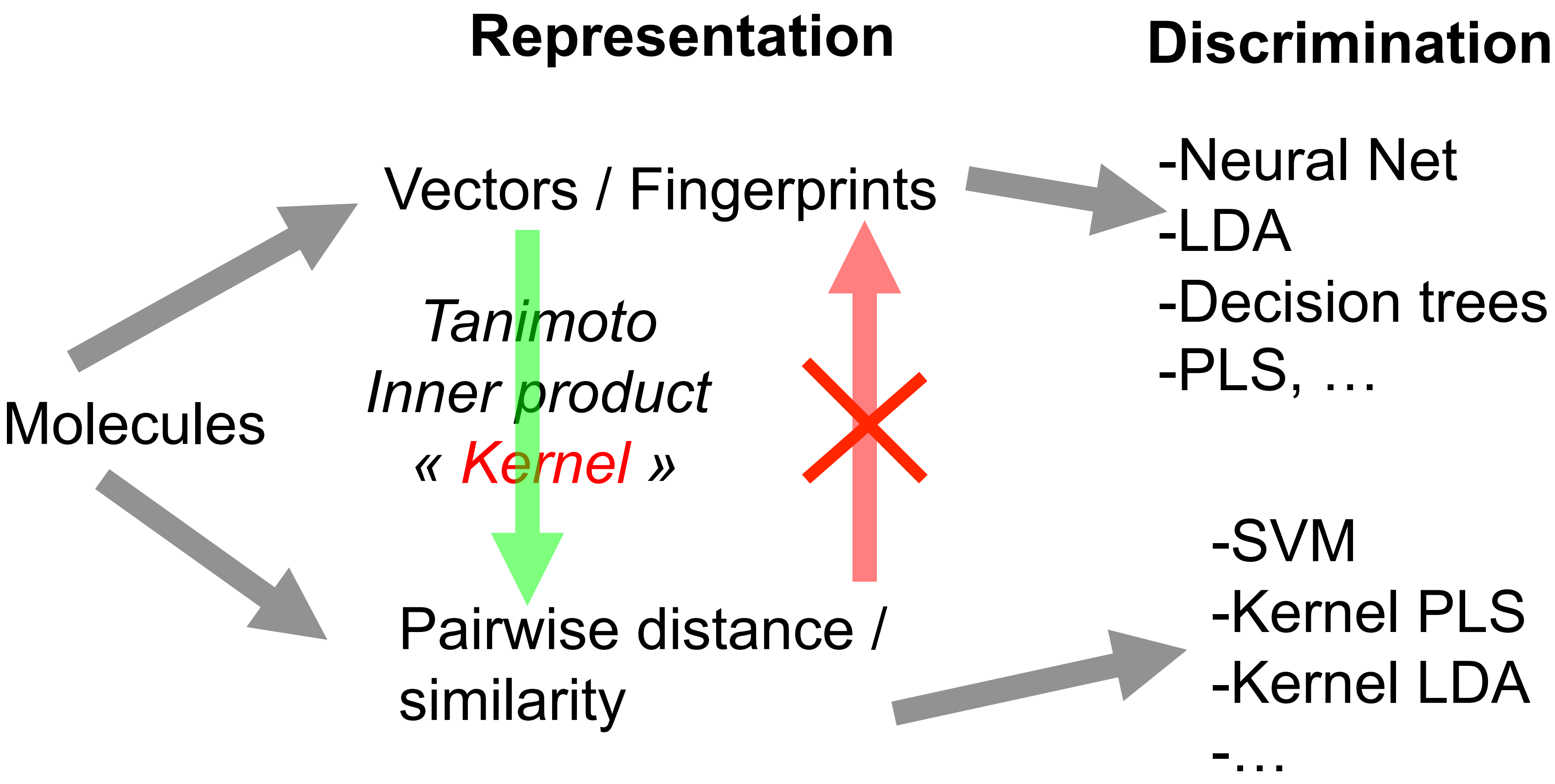
-Large margin
(useful in high dim)

- Need pairwise distance / similarity as input instead of vectors / fingerprints

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



# From descriptors to similarities



**Representation**

**Discrimination**

Molecules

Vectors / Fingerprints

*Tanimoto*

*Inner product*

« *Kernel* »

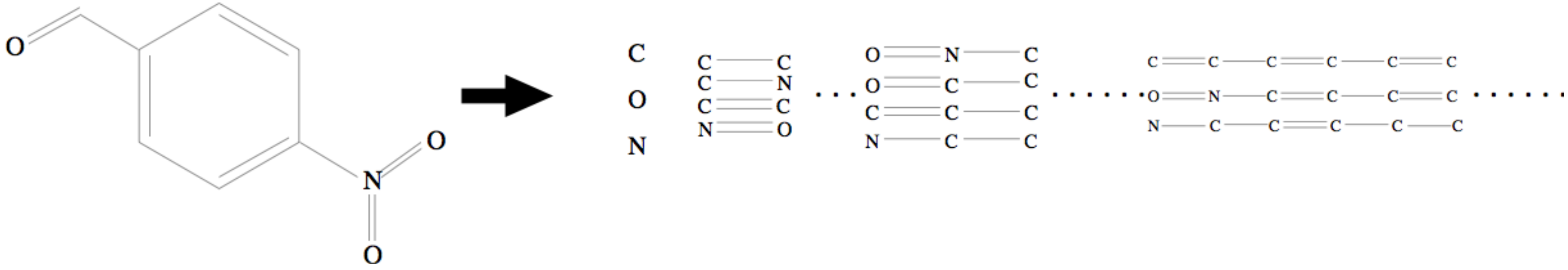
Pairwise distance / similarity

-Neural Net
-LDA
-Decision trees
-PLS, …

-SVM
-Kernel PLS
-Kernel LDA
-…

# 2D kernels

# 2D fragment kernels (walks)



- For any $d > 0$ let $\phi_d(x)$ be the vector of counts of all fragments of length $d$:

$$\phi_1(x) = (\quad \texttt{\#(C),\#(O),\#(N),}\ \ \ldots)^\top$$
$$\phi_2(x) = (\quad \texttt{\#(C-C),\#(C=O),\#(C-N),}\ \ \ldots)^\top \quad \text{etc...}$$

- The 2D fingerprint kernel is defined, for $\lambda < 1$, by

$$K_{2D}(x, x') = \sum_{d=1}^{\infty} \lambda(d) \phi_d(x)^\top \phi_d(x') \, .$$

*Kashima et al. (2003), Gärtner et al. (2003)*

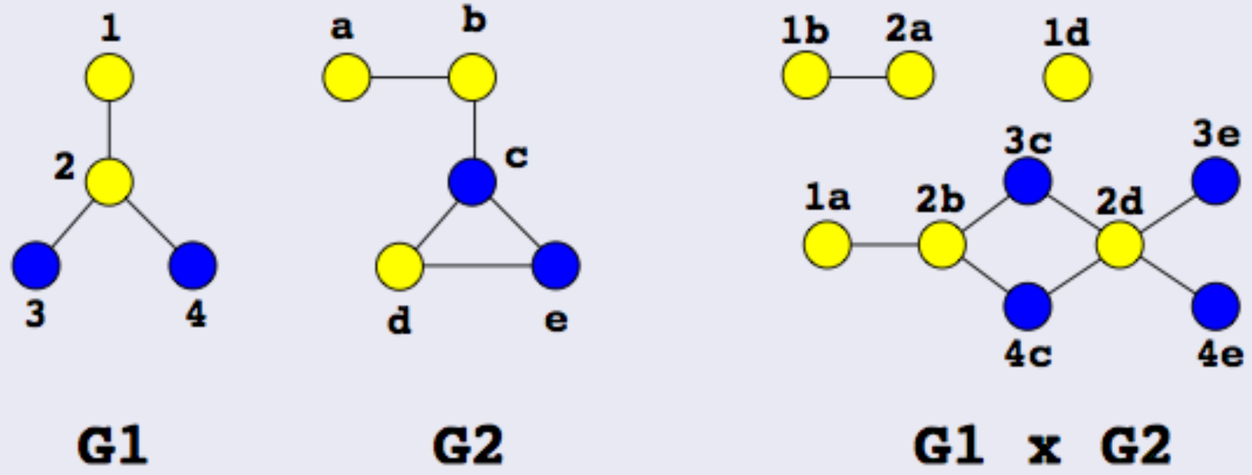
# Properties of the 2D fragment kernel

- Corresponds to a fingerprint of infinite size
- Can be computed efficiently in O(|x| ^3 |x'|^3) (much faster in practice)
- Solves the problem of clashes and memory storage (fingerprints are not computed explicitly)

*Kashima et al. (2003), Gärtner et al. (2003)*

# 2D kernel computational trick

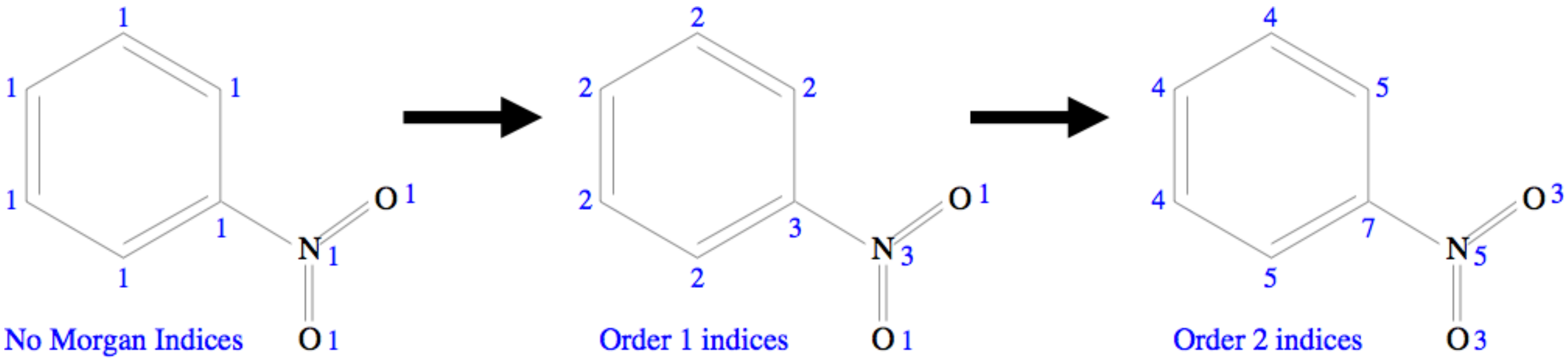
- Rephrase the kernel computation as that of counting the number of walks on a graph (the product graph)



- The infinite counting can be factorized

$$\lambda A + \lambda^2 A^2 + \lambda^3 A^3 + \ldots = (I - \lambda A)^{-1} - I.$$

# Extension 1: label enrichment

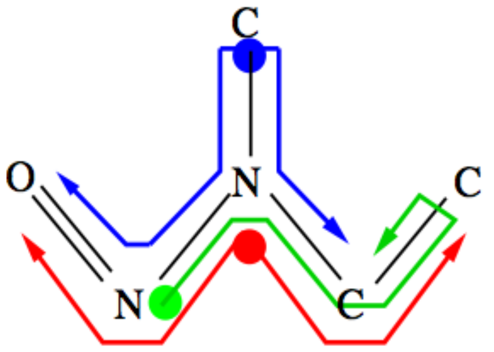


-Increases the expressiveness of the kernel
-Faster computation with more labels
-Other relabeling schemes are possible (pharmacophores)

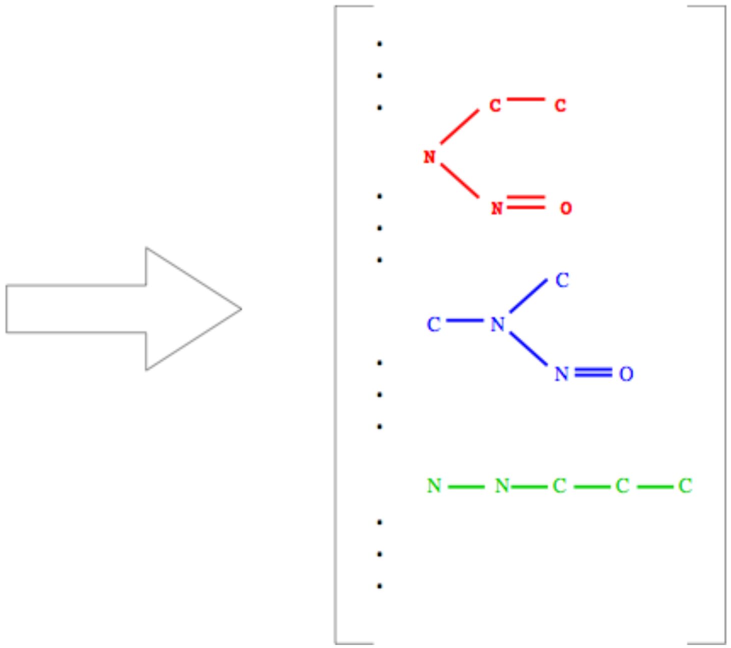
*Mahé et al. (2005)*

# Extension 2: subtree patterns

« All subtree patterns »



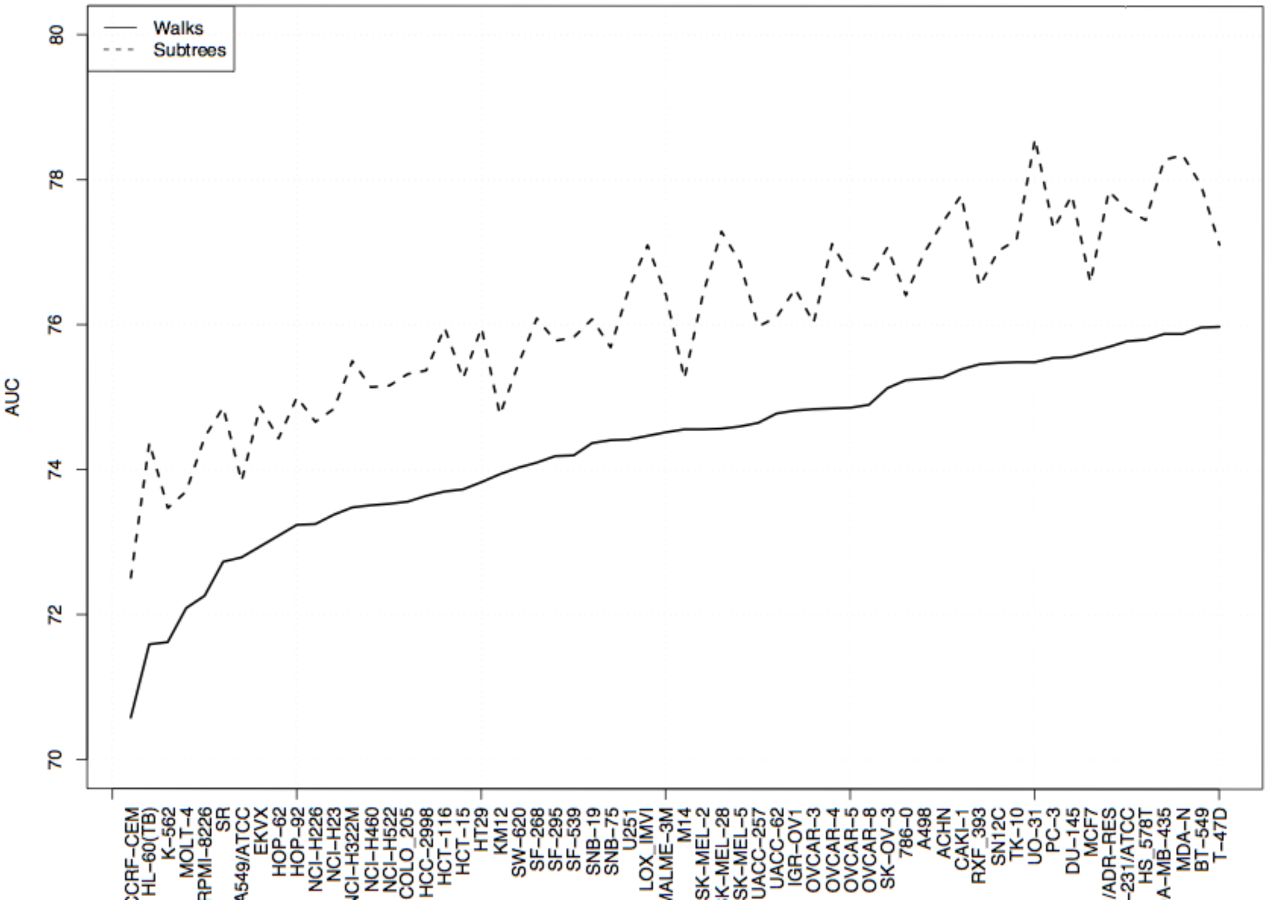
Mahé and V., *Mach. Learn*, 2009.



$$\mathcal{T}(v,n+1) = \sum_{R\subset\mathcal{N}(v)} \prod_{v'\in R} \lambda_t(v,v')\mathcal{T}(v',n)$$

*Ramon et al. (2004), Mahé & V. (2009)*

# 2D subtree vs walk kernel

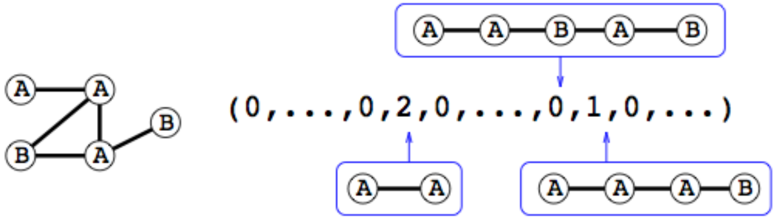


*NCI 60 dataset*

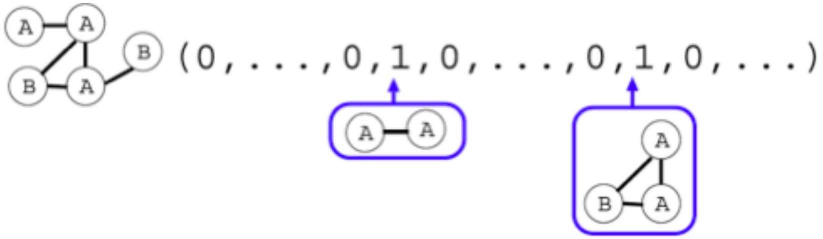
*Mahé & V. (2009)*

# Other 2D kernels

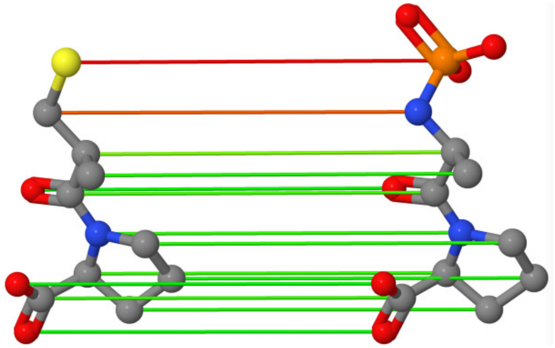
- Indexing by all **shortest paths** *(Borgwardt & Kriegel 2005)*



- Indexing by all **small subgraphs** *(Shervashidze et al. 2009)*

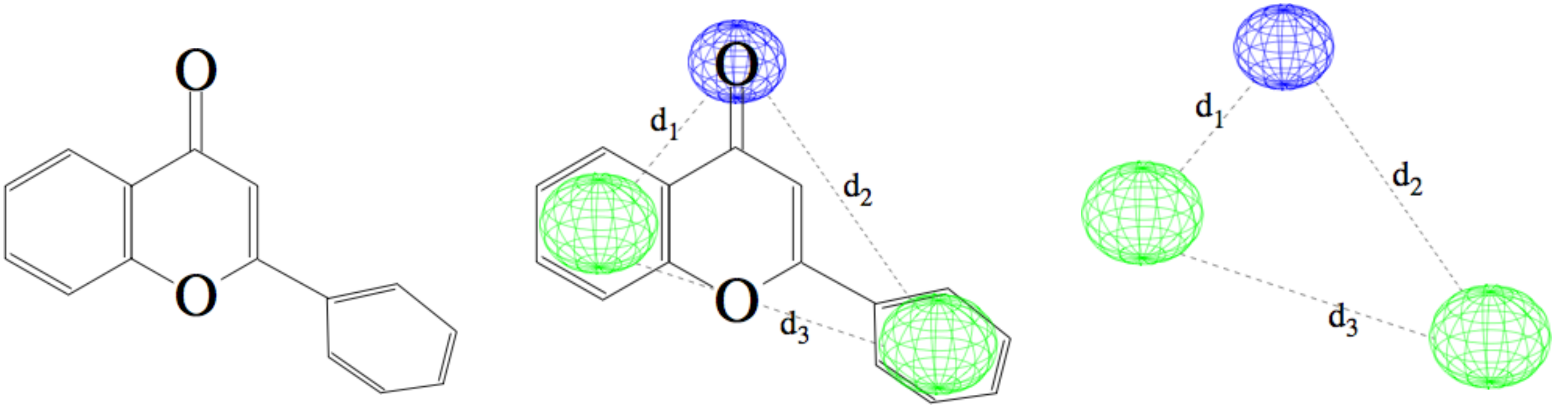


- **Optimal assignment** kernel *(Fröhlich et al. 2005)*



# 3D pharmacophore kernel

# 3-point pharmacophores



A set of 3 atoms, and 3 inter-atom distances:

$$\mathcal{T} = \{((x_1, x_2, x_3), (d_1, d_2, d_3)), x_i \in \{\text{atom types}\}; d_i \in \mathbb{R}\}$$

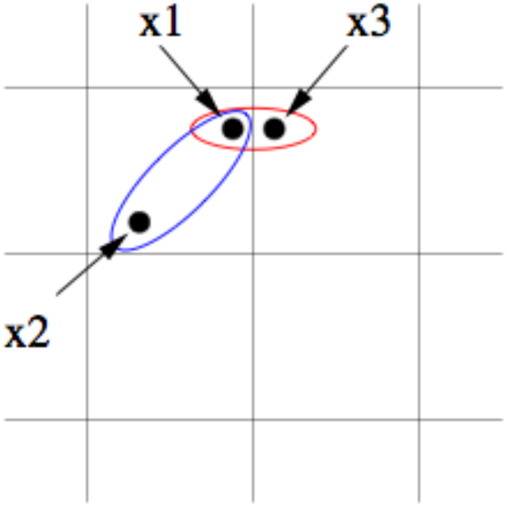
Mahé et al., *J. Chem. Inf. Model.*, 2006.

# 3D fingerprint kernel

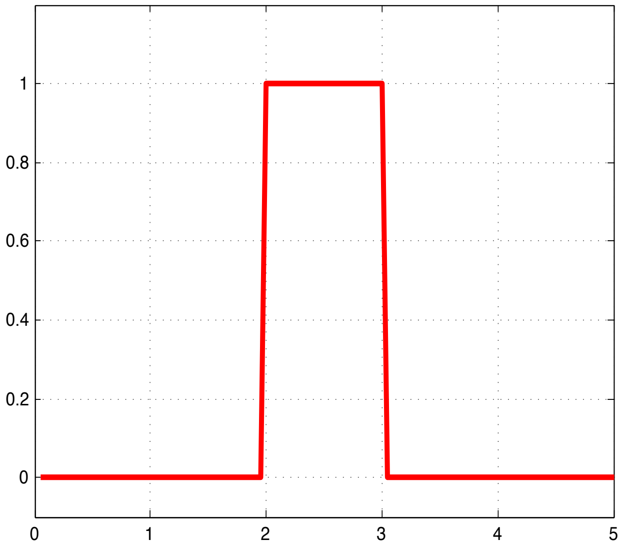
1. Discretize the space of pharmacophores $\mathcal{T}$ (e.g., 6 atoms or groups of atoms, 6-7 distance bins) into a finite set $\mathcal{T}_d$
2. Count the number of occurrences $\phi_t(x)$ of each pharmacophore bin $t$ in a given molecule $x$, to form a pharmacophore fingerprint.

A simple 3D kernel is the inner product of pharmacophore fingerprints:

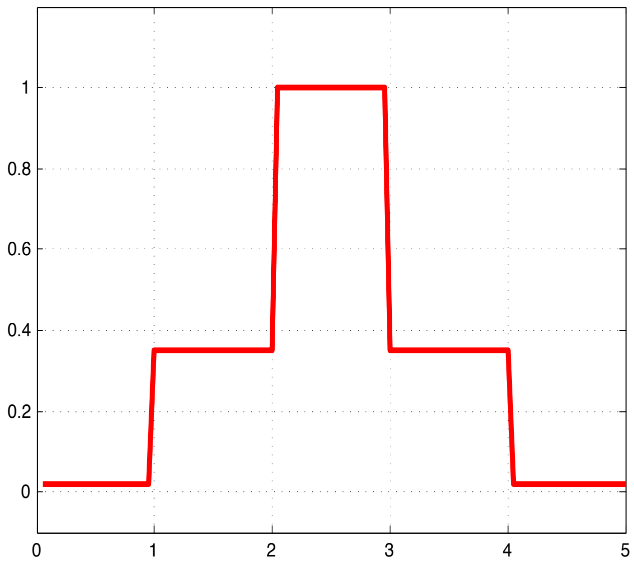
$$K(x, x') = \sum_{t \in \mathcal{T}_d} \phi_t(x)\phi_t(x') \, .$$



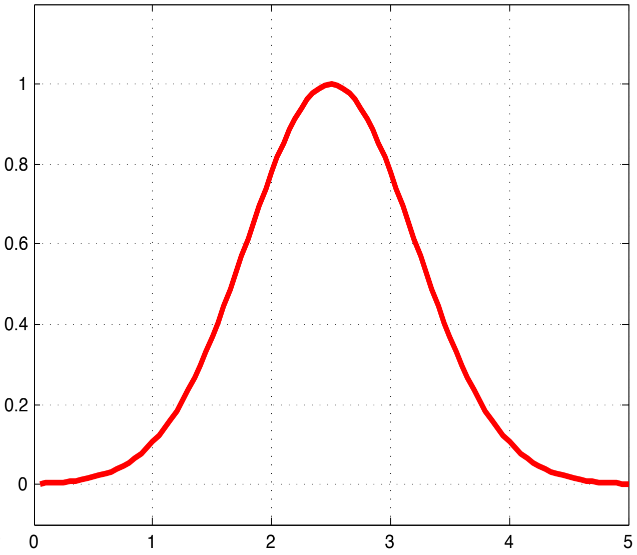
# Removing discretization artifacts



3D Fingerprint



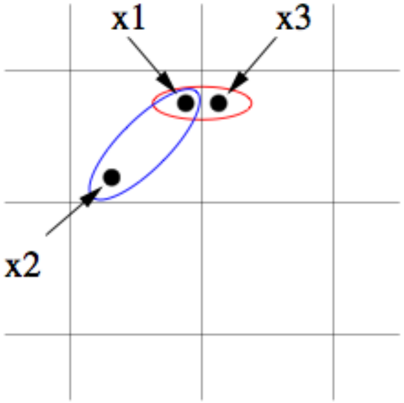
3D Fuzzy Fingerprint



3D Kernel

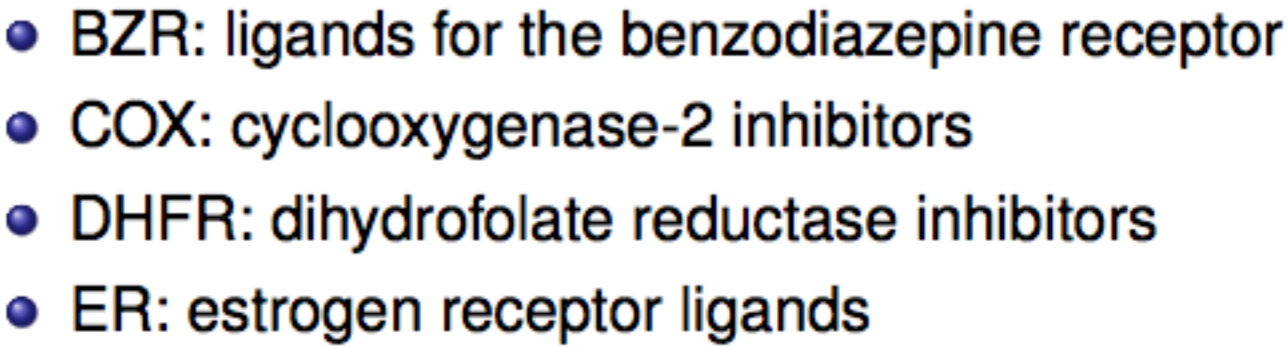
# From the fingerprint kernel to the pharmacophore kernel

$$
\begin{aligned}
K(x,y) &= \sum_{t \in \mathcal{T}_d} \phi_t(x)\phi_t(y) \\
&= \sum_{t \in \mathcal{T}_d} \Big( \sum_{p_x \in \mathcal{P}(x)} \mathbf{1}(\mathrm{bin}(\mathbf{p_x}) = \mathbf{t}) \Big) \Big( \sum_{p_y \in \mathcal{P}(y)} \mathbf{1}(\mathrm{bin}(\mathbf{p_y}) = \mathbf{t}) \Big) \\
&= \sum_{p_x \in \mathcal{P}(x)} \sum_{p_y \in \mathcal{P}(y)} \mathbf{1}(\mathrm{bin}(\mathbf{p_x}) = \mathrm{bin}(\mathbf{p_y}))
\end{aligned}
$$

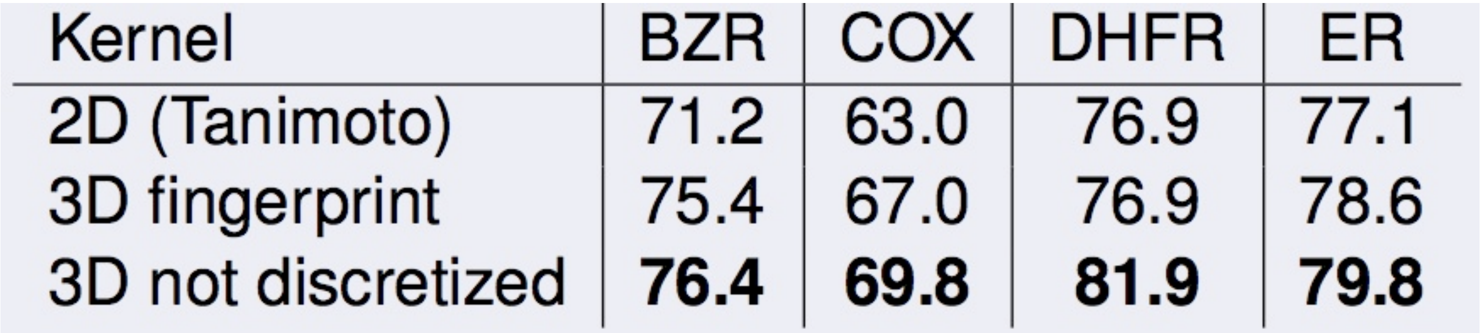


$$
K(x,y) = \sum_{p_x \in \mathcal{P}(x)} \sum_{p_y \in \mathcal{P}(y)} \exp\left( -\gamma ||p_x - p_y||^2 \right)
$$

# Experiments



- BZR: ligands for the benzodiazepine receptor
- COX: cyclooxygenase-2 inhibitors
- DHFR: dihydrofolate reductase inhibitors
- ER: estrogen receptor ligands

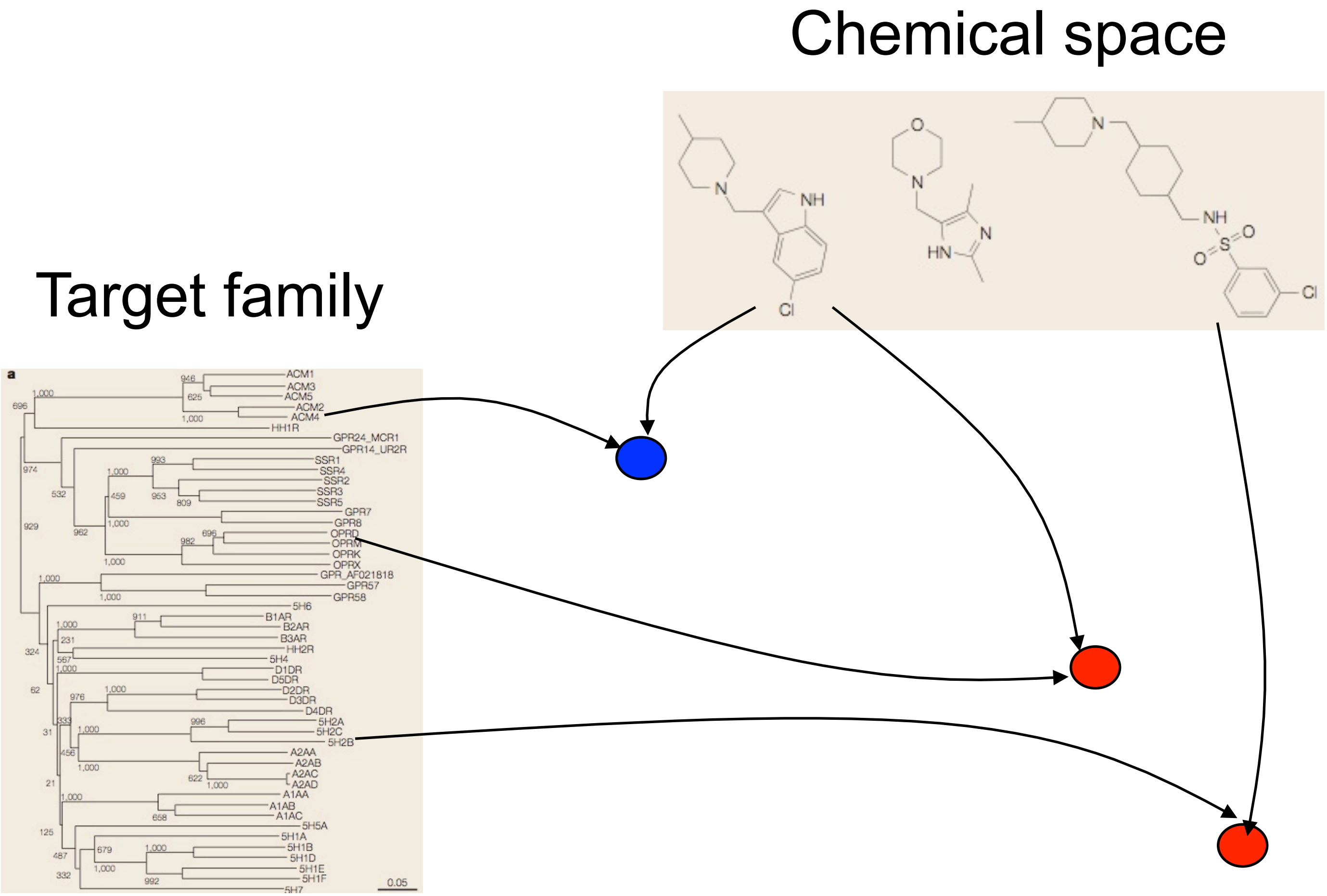


| Kernel | BZR | COX | DHFR | ER |
|---|---|---|---|---|
| 2D (Tanimoto) | 71.2 | 63.0 | 76.9 | 77.1 |
| 3D fingerprint | 75.4 | 67.0 | 76.9 | 78.6 |
| 3D not discretized | **76.4** | **69.8** | **81.9** | **79.8** |

Mahé et al., *J. Chem. Inf. Model.*, 2006.

# Towards *in silico* chemogenomics

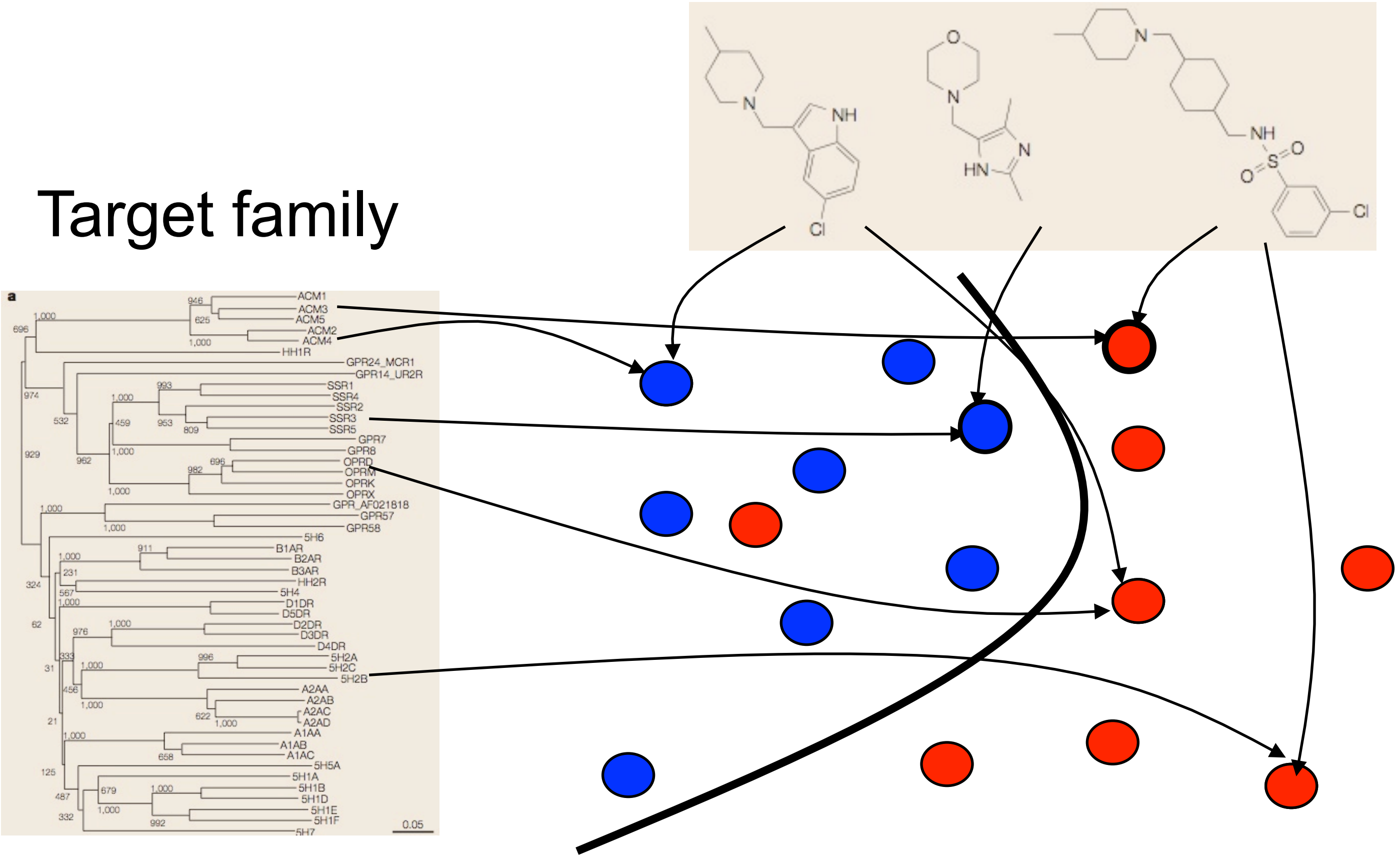
# Chemogenomics



# *In silico* Chemogenomics

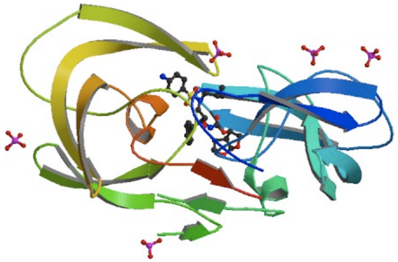
Chemical space

Target family

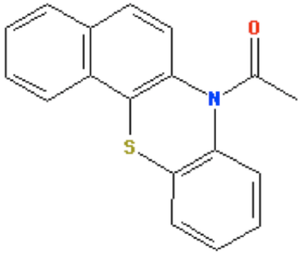


# Fingerprint for a (target,molecule) pair?

t=

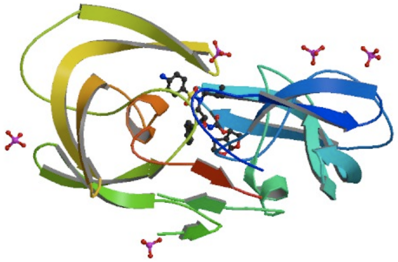


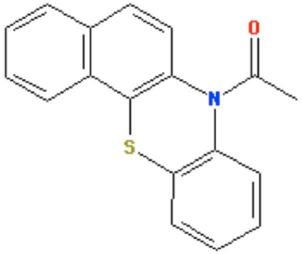
c =



$$\underbrace{\Phi_{tar}(t) = \begin{cases}\text{-Sequence}\\ \text{-Structure}\\ \text{-Evolution}\\ \text{-Expression}\\ \text{-…}\end{cases} \qquad \Phi_{lig}(c) = \begin{cases}\text{-2D}\\ \text{-3D}\\ \text{-Pharmacophore}\\ \text{-MW, logP, …}\end{cases}}_{\Phi(c,t) \,=\, ???}$$

# Fingerprint for a (target,molecule) pair?

T= 

c = 

$$\Phi_{tar}(t) = \begin{cases} \text{-Sequence} \\ \text{-Structure} \\ \text{-Evolution} \\ \text{-Expression} \\ \text{-…} \end{cases}$$

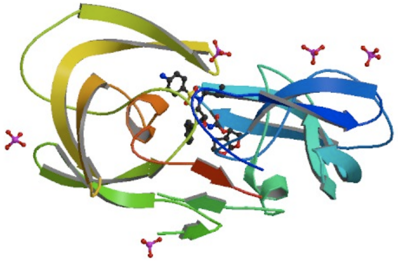
$$\Phi_{lig}(c) = \begin{cases} \text{-2D} \\ \text{-3D} \\ \text{-Pharmacophore} \\ \text{-logP, …} \end{cases}$$

$$\Phi(c,t) = \Phi_{lig}(c) \otimes \Phi_{tar}(t)$$

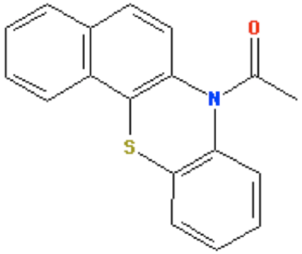
$10^6$ $10^3$ $10^3$

# Similarity for (target,molecule) pairs

t=



c =



$$K_{target}(t,t') =$$

-Sequence
-Structure
-Evolution
-Expression
-…

$$K_{ligand}(c,c') =$$

-2D
-3D
-Pharmacophore
-logP, …

$$K\left((c,t),(c',t')\right) = K_{target}(t,t') \times K_{ligand}(c,c')$$

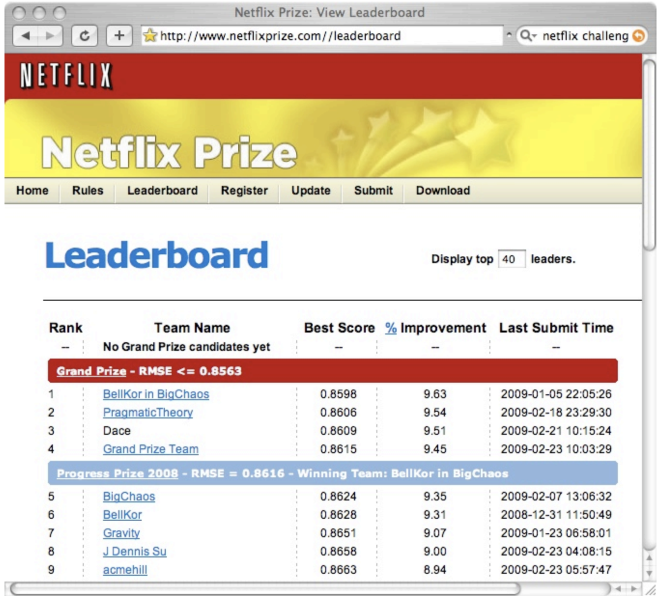
# Summary: SVM for chemogenomics

1. Choose a kernel (similarity) for targets
2. Choose a kernel (similarity) for ligands
3. Train a SVM model with the product kernel for (target/ligand) pairs

# Important remark

- New methods are being actively developed in machine learning for learning over pairs

- « Collaborative filtering », « transfer learning », « multitask learning », « MMMF », « pairwise SVM », etc…



Leaderboard

Display top 40 leaders.

| Rank | Team Name | Best Score | % Improvement | Last Submit Time |
|---|---|---|---|---|
| -- | No Grand Prize candidates yet | -- | -- | -- |
| Grand Prize - RMSE <= 0.8563 | | | | |
| 1 | BellKor in BigChaos | 0.8598 | 9.63 | 2009-01-05 22:05:26 |
| 2 | PragmaticTheory | 0.8606 | 9.54 | 2009-02-18 23:29:30 |
| 3 | Dace | 0.8609 | 9.51 | 2009-02-21 10:15:24 |
| 4 | Grand Prize Team | 0.8615 | 9.45 | 2009-02-23 10:03:29 |
| Progress Prize 2008 - RMSE = 0.8616 - Winning Team: BellKor in BigChaos | | | | |
| 5 | BigChaos | 0.8624 | 9.35 | 2009-02-07 13:06:32 |
| 6 | BellKor | 0.8628 | 9.31 | 2008-12-31 11:50:49 |
| 7 | Gravity | 0.8651 | 9.07 | 2009-01-23 06:58:01 |
| 8 | J Dennis Su | 0.8658 | 9.00 | 2009-02-23 04:08:15 |
| 9 | acmehill | 0.8663 | 8.94 | 2009-02-23 05:57:47 |

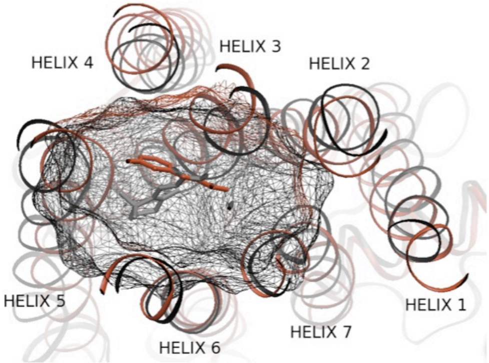
*37k registered teams from 180 countries!*

# Application: virtual screening of GPCR

**Data**: GLIDA database filtered for drug-like compounds

- 2446 ligands
- 80 GPCR
- 4051 interactions
- *4051 negative interactions generated randomly*

**Ligand similarity**
-2D Tanimoto
-3D pharmacophore

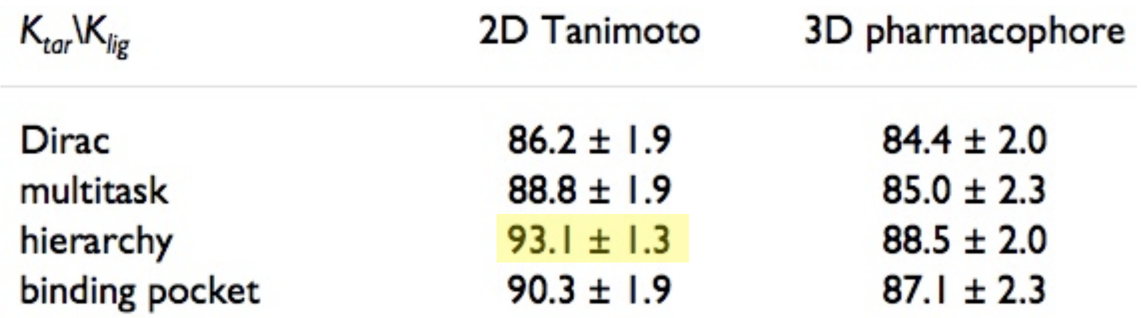


**Target similarities**
-0/1 Dirac (no similarity)
-Multitask (uniform similarity)
-GLIDA's hierarchy similarity
-Binding pocket similarity (31 AA)

(Jacob et al., *BMC Bioinformatics*, 2008)

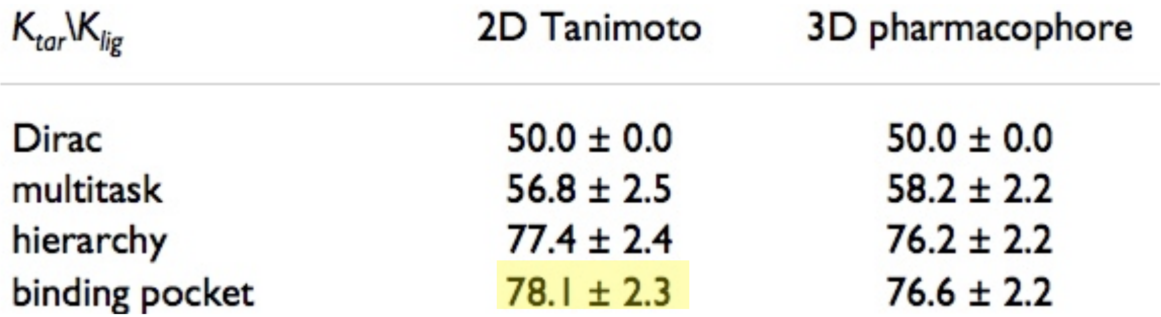
# Results (mean accuracy over GPCRs)

5-fold cross-validation



| $K_{tar}\backslash K_{lig}$ | 2D Tanimoto | 3D pharmacophore |
|---|---|---|
| Dirac | 86.2 ± 1.9 | 84.4 ± 2.0 |
| multitask | 88.8 ± 1.9 | 85.0 ± 2.3 |
| hierarchy | 93.1 ± 1.3 | 88.5 ± 2.0 |
| binding pocket | 90.3 ± 1.9 | 87.1 ± 2.3 |

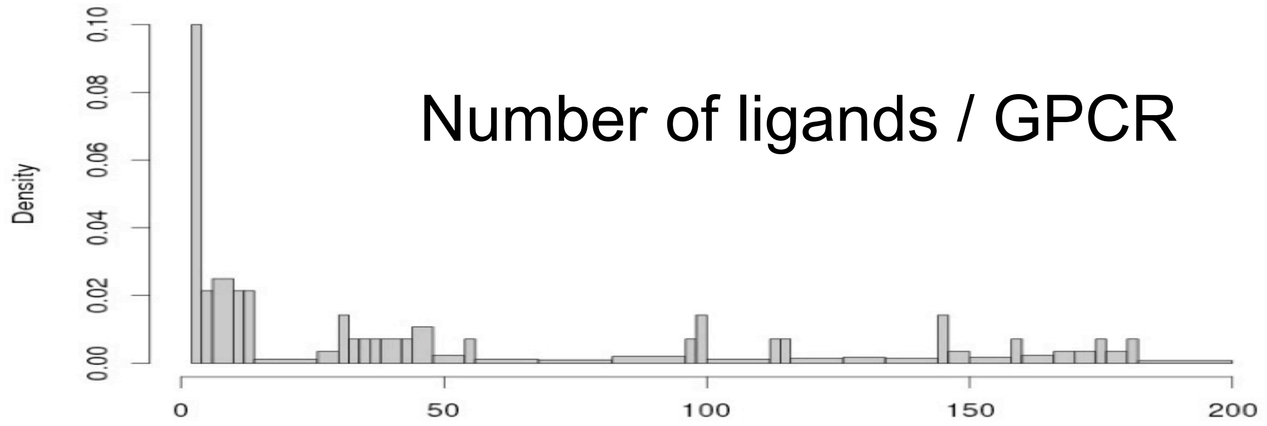
Orphan GPCRs setup



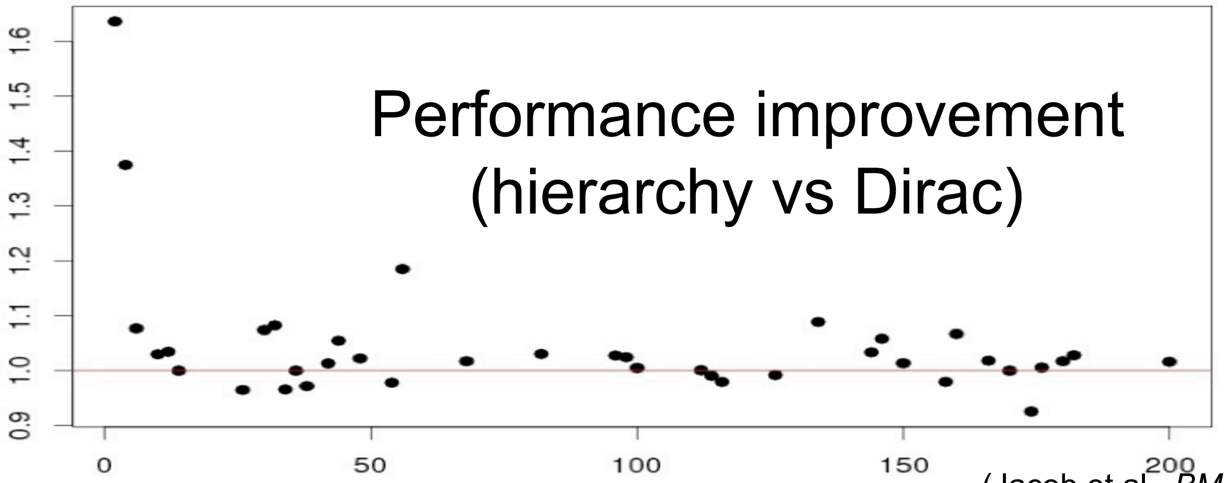
| $K_{tar}\backslash K_{lig}$ | 2D Tanimoto | 3D pharmacophore |
|---|---|---|
| Dirac | 50.0 ± 0.0 | 50.0 ± 0.0 |
| multitask | 56.8 ± 2.5 | 58.2 ± 2.2 |
| hierarchy | 77.4 ± 2.4 | 76.2 ± 2.2 |
| binding pocket | 78.1 ± 2.3 | 76.6 ± 2.2 |

(Jacob et al., *BMC Bioinformatics*, 2008)

# Influence of the number of known ligands



Number of ligands / GPCR



Performance improvement (hierarchy vs Dirac)

(Jacob et al., *BMC Bioinformatics*, 2008)

# Screening of enzymes, GPCRs, ion channels

**Data**: KEGG BRITE database, redundancy removed

**Enzymes**

-675 targets

-524 molecules

-1218 interactions

-1218 negatives

**GPCRs**

-100 targets

-219 molecules

-399 interactions

-399 negatives

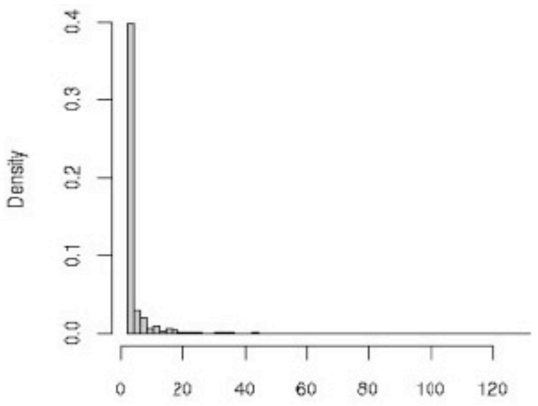
**Ion channels**

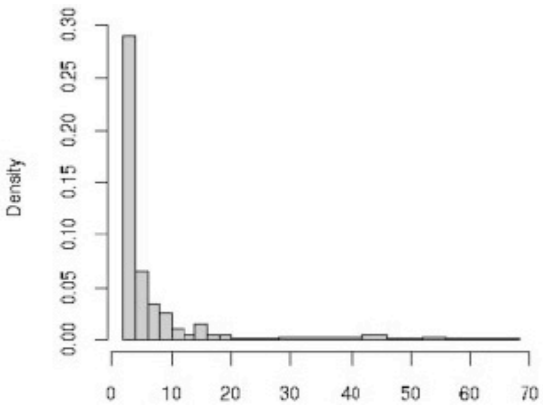
-114 targets

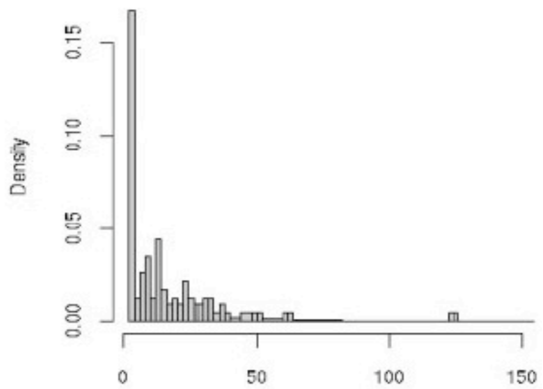
-462 molecules

-1165 interactions

-1165 negatives



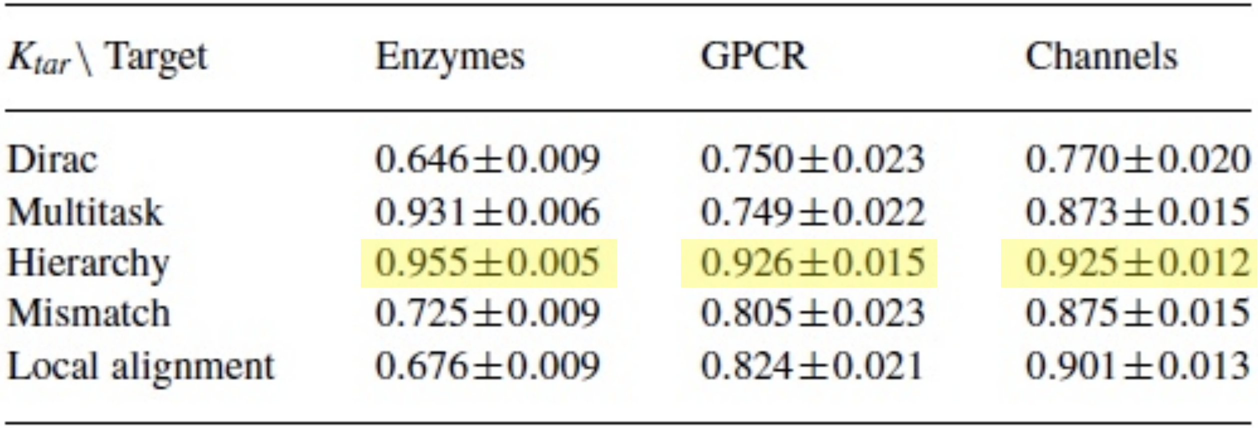




(Jacob and V., *Bioinformatics*, 2008)

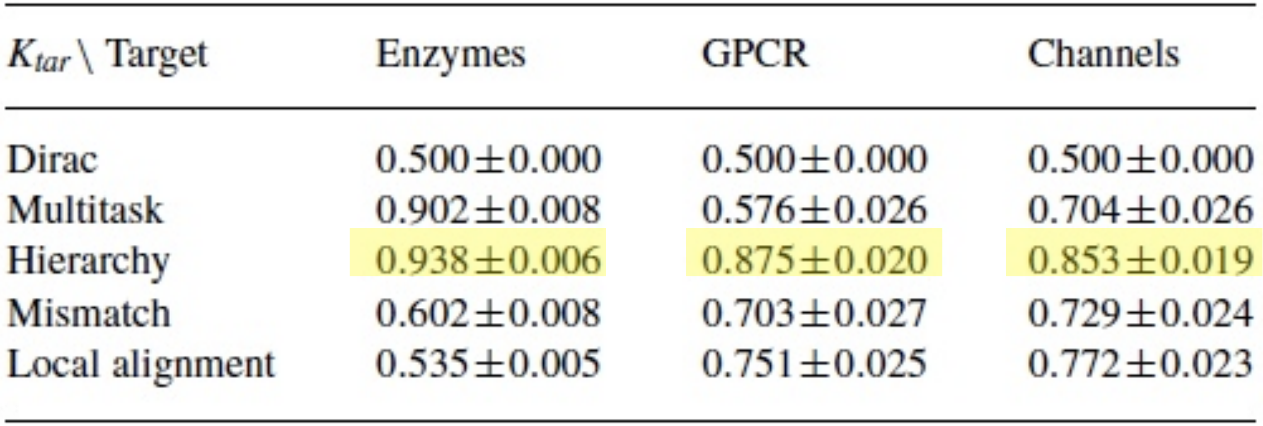
# Results (mean AUC)

10-fold CV



| $K_{tar}$ \ Target | Enzymes | GPCR | Channels |
|---|---|---|---|
| Dirac | 0.646±0.009 | 0.750±0.023 | 0.770±0.020 |
| Multitask | 0.931±0.006 | 0.749±0.022 | 0.873±0.015 |
| Hierarchy | 0.955±0.005 | 0.926±0.015 | 0.925±0.012 |
| Mismatch | 0.725±0.009 | 0.805±0.023 | 0.875±0.015 |
| Local alignment | 0.676±0.009 | 0.824±0.021 | 0.901±0.013 |

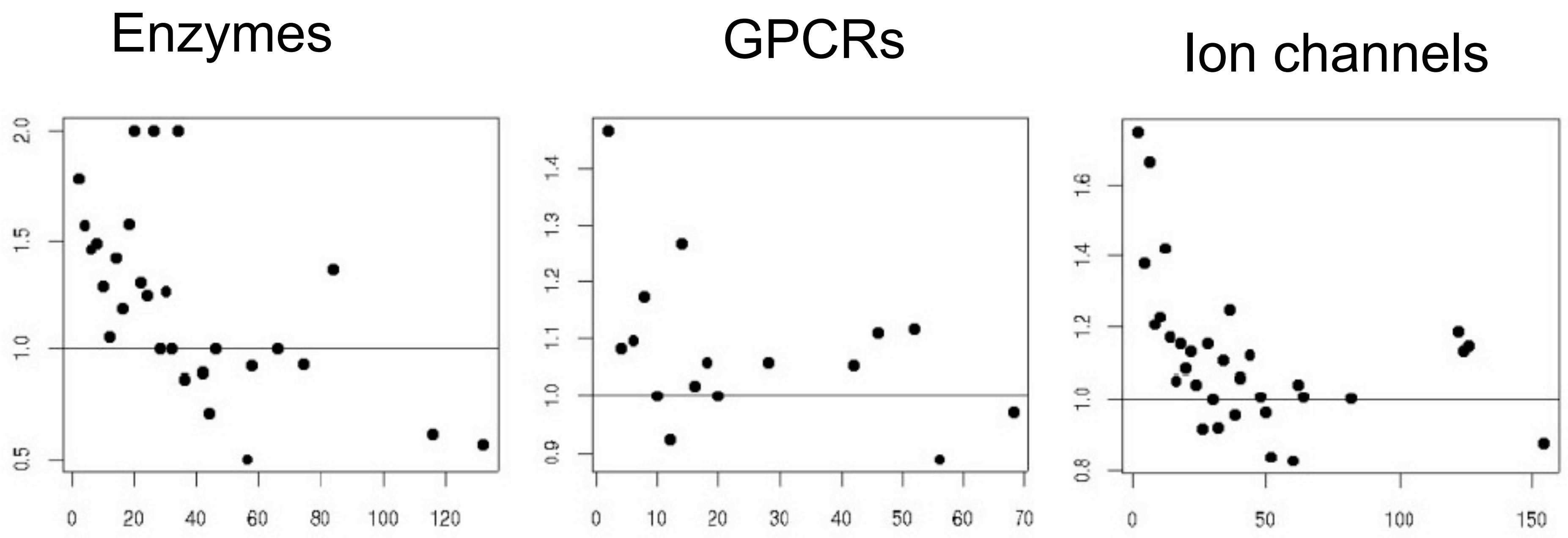
Orphan setting



| $K_{tar}$ \ Target | Enzymes | GPCR | Channels |
|---|---|---|---|
| Dirac | 0.500±0.000 | 0.500±0.000 | 0.500±0.000 |
| Multitask | 0.902±0.008 | 0.576±0.026 | 0.704±0.026 |
| Hierarchy | 0.938±0.006 | 0.875±0.020 | 0.853±0.019 |
| Mismatch | 0.602±0.008 | 0.703±0.027 | 0.729±0.024 |
| Local alignment | 0.535±0.005 | 0.751±0.025 | 0.772±0.023 |

(Jacob and V., *Bioinformatics*, 2008)

# Influence of the number of known ligands



Relative improvement : hierarchy vs Dirac

(Jacob and V., *Bioinformatics*, 2008)

# Conclusion

- SVM offer state-of-the-art performance in many chemo- and bio-informatics applications
- The kernel trick is useful to
  - Work implicitly with **many features** without computing them (*2D fragment kernels)*
  - Work with **similarity measures** that cannot be derived from descriptors (*optimal alignment kernel)*
  - Relax the need for **discretization** (*3D pharmacophore kernel)*
  - Work in a **product space** (*chemogenomics*)
- Promising direction:
  - Multiple kernel learning
  - Collaborative filtering in product space

# Thank you !

***Collaborators:***
***P. Mahé, L. Jacob, V. Stoven, B. Hoffmann***

*References* :
http://cbio.ensmp.fr/~jvert

*Open-source kernels for chemoinformatics:*
http://chemcpp.sourceforge.net

Ensemble, prenons le cancer de vitesse.

Institut national
de la santé et de la recherche médicale