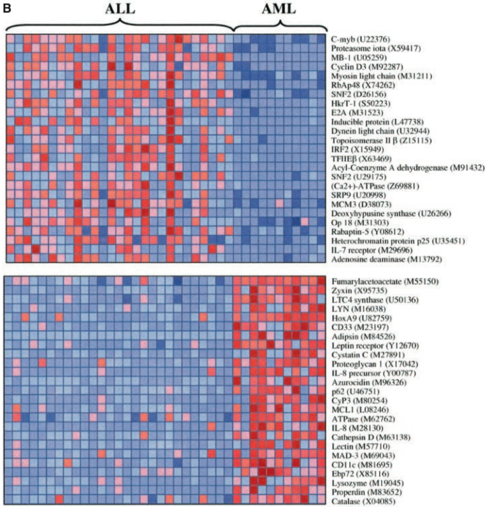
Some contributions of machine learning in bioinformatics

Jean-Philippe Vert
Jean-Philippe.Vert@mines-paristech.fr

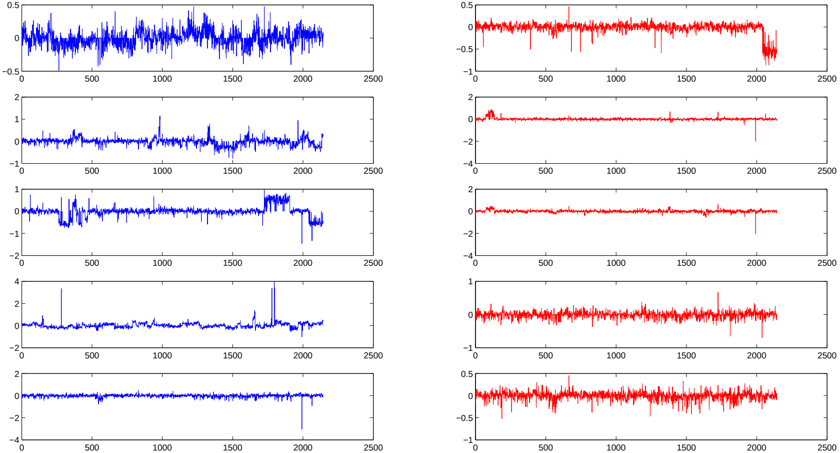
Mines ParisTech / Curie Institute / Inserm

ENS Paris, séminaire du Département d'informatique, Nov 24, 2009

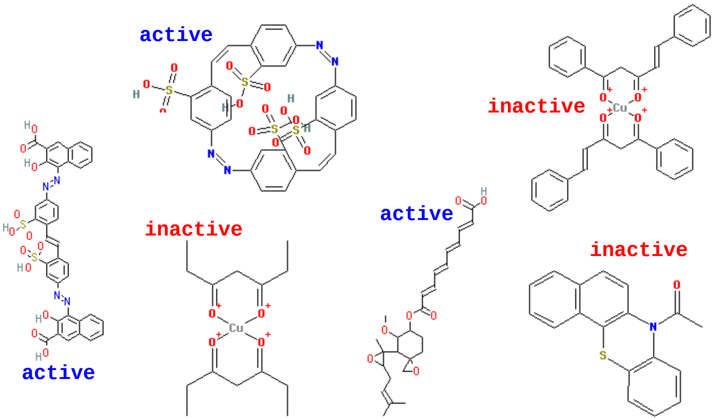
Cancer diagnosis



Cancer prognosis

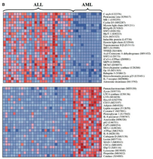


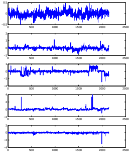
Virtual screening for drug discovery

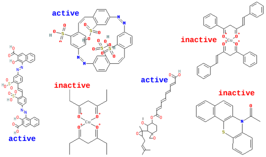


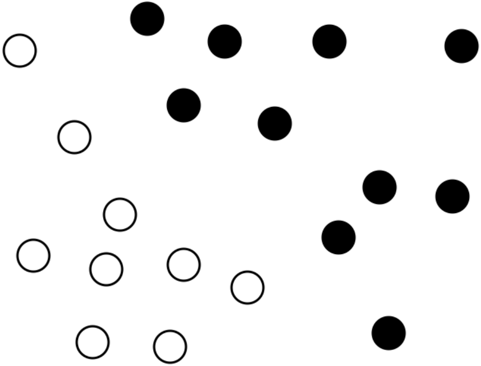
NCI AIDS screen results (from http://cactus.nci.nih.gov).

Pattern recognition, *aka* supervised classification

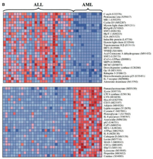


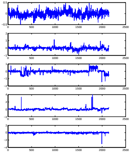


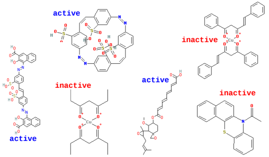


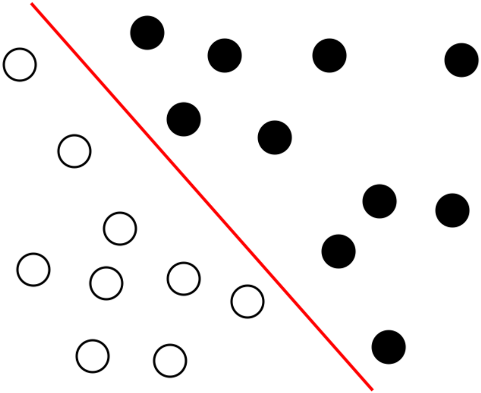


Pattern recognition, *aka* supervised classification

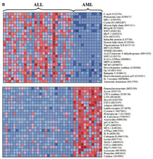


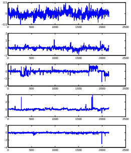


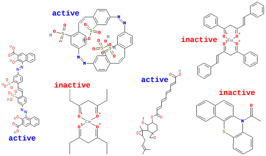


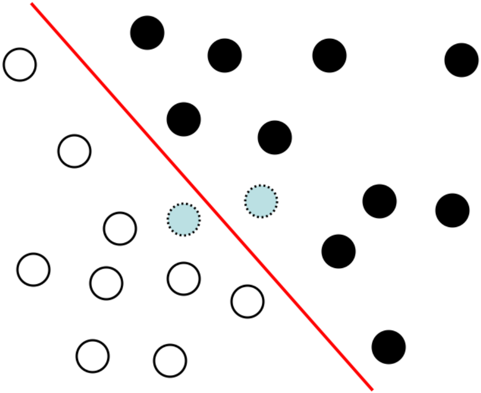


Pattern recognition, *aka* supervised classification

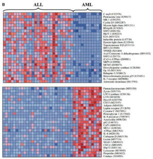


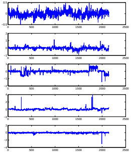


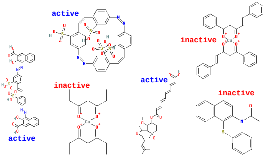


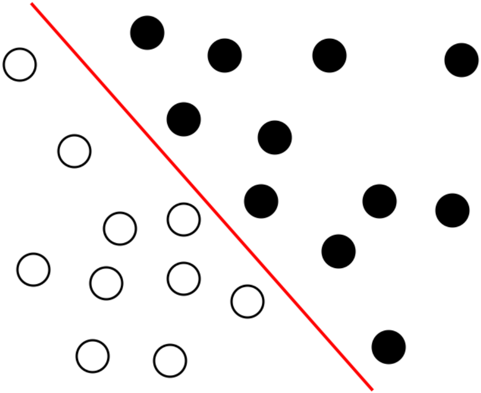


Pattern recognition, *aka* supervised classification

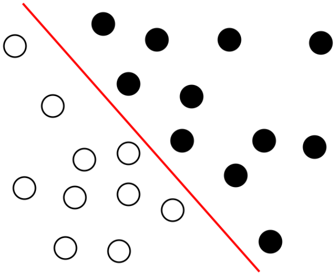








Pattern recognition, *aka* supervised classification



Challenges

- High dimension
- Few samples
- Structured data
- Heterogeneous data
- Prior knowledge
- Fast and scalable implementations
- Interpretable models

Formalization

The problem

- Given a set of training instances $(x_1, y_1), \ldots, (x_n, y_n)$, where $x_i \in \mathcal{X}$ are data and $y_i \in \mathcal{Y}$ are continuous or discrete variables of interest,
- Estimate a function

  $$y = f(x)$$

  where x is any new data to be labeled.
- f should be accurate and intepretable.

Linear classifiers

The model

- Each sample $x \in \mathcal{X}$ is represented by a vector of features (or descriptors, or patterns):

$$\Phi(x) = (\Phi_1(x), \dots, \Phi_p(x)) \in \mathbb{R}^p \, .$$

- Based on the training set we estimate a linear function:

$$f_\beta(x) = \sum_{i=1}^{p} \beta_i \Phi_i(x) = \beta^\top \Phi(x) \, .$$

Estimating linear classifiers

- For any candidate set of weights $\beta = (\beta_1, \ldots, \beta^p)$ we quantify how "good" the linear function f_β is on the training set with some empirical risk:

$$R(\beta) = \frac{1}{n}\sum_{i=1}^{n} l(f_\beta(x_i), y_i)\,.$$

- We choose the β that achieves the minimium empirical risk, subject to some constraint:

$$\Omega(\beta) \leq C\,.$$

- Equivalently we solve

$$\min_{\beta\in\mathbb{R}^p} \frac{1}{n}\sum_{i=1}^{n} l(f_\beta(x_i), y_i) + \lambda\Omega(\beta)\,.$$

Two related questions

$$f_\beta(x) = \sum_{i=1}^{p} \beta_i \Phi_i(x)$$

$$\min_{\beta \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^{n} l(f_\beta(x_i), y_i) + \lambda \Omega(\beta)$$

- How to design the features $\Phi(x)$?
- How to estimate the model β?

Outline

1. Cancer prognosis from DNA copy number variations

2. Diagnosis and prognosis from gene expression data

3. Virtual screening for drug discovery

4. Conclusion

Outline

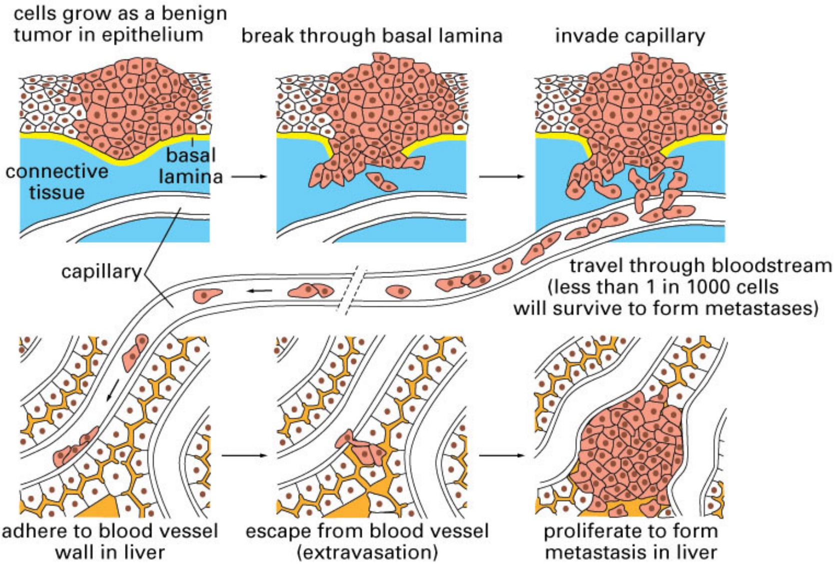
1. Cancer prognosis from DNA copy number variations

2. Diagnosis and prognosis from gene expression data

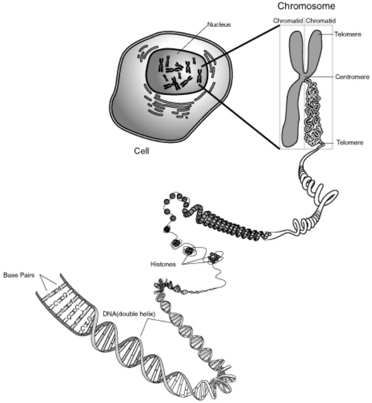
3. Virtual screening for drug discovery

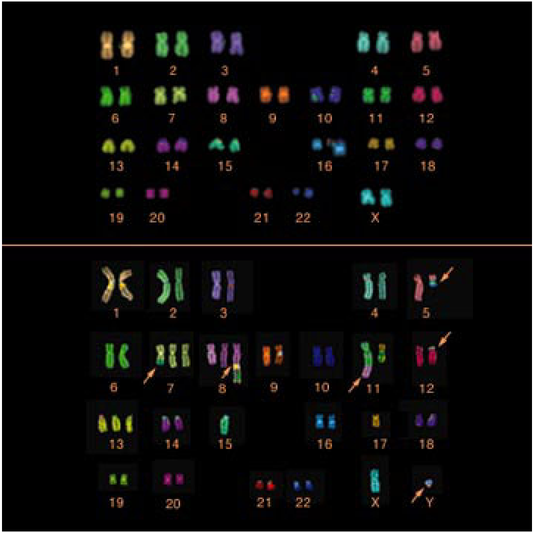
4. Conclusion

A simple view of cancer progression


cells grow as a benign tumor in epithelium
break through basal lamina
invade capillary
connective tissue
basal lamina
capillary
travel through bloodstream (less than 1 in 1000 cells will survive to form metastases)
adhere to blood vessel wall in liver
escape from blood vessel (extravasation)
proliferate to form metastasis in liver

Chromosomic aberrations in cancer

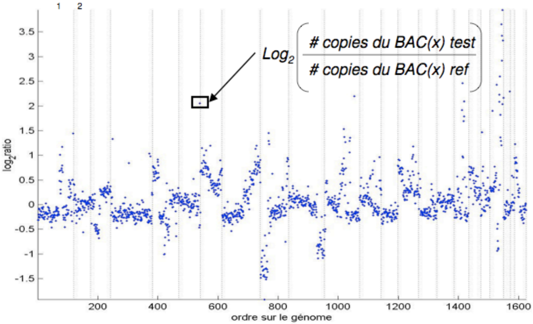




Comparative Genomic Hybridization (CGH)

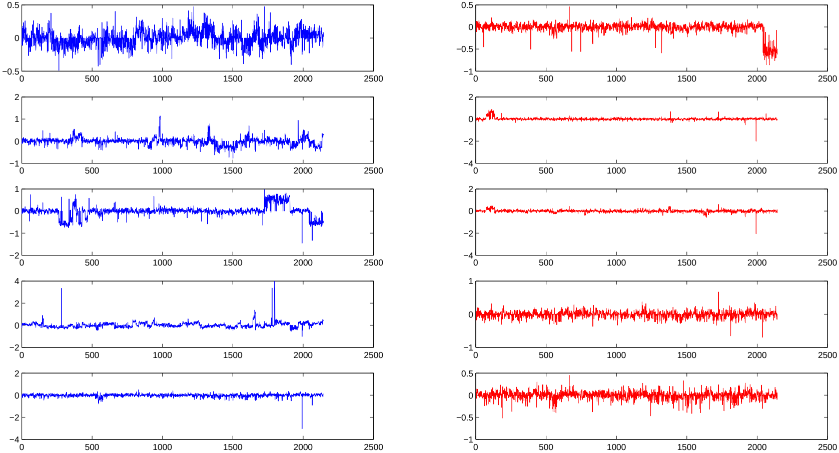
Motivation

- Comparative genomic hybridization (CGH) data measure the DNA copy number along the genome
- Very useful, in particular in cancer research
- Can we classify CGH arrays for diagnosis or prognosis purpose?



Jain et al. Genome research 2002 12:325-332

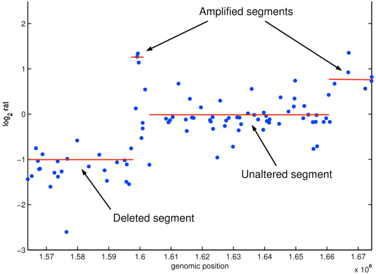
Aggressive vs non-aggressive melanoma



CGH array classification

Prior knowledge

- For a CGH profile $x \in \mathbb{R}^p$, we focus on linear classifiers, i.e., the sign of :
$$f_\beta(x) = \beta^\top x \,.$$
- We expect β to be
  - sparse : not all positions should be discriminative
  - piecewise constant : within a selected region, all probes should contribute equally



Promoting sparsity with the ℓ_1 penalty

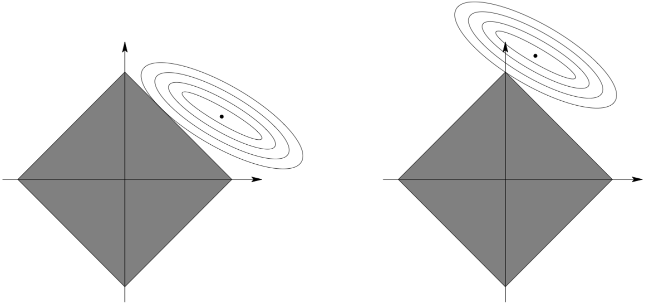
The ℓ_1 penalty (Tibshirani, 1996; Chen et al., 1998)

The solution of

$$\min_{\beta \in \mathbb{R}^p} R(\beta) + \lambda \sum_{i=1}^{p} |\beta_i|$$

is usually sparse.

Geometric interpretation with $p = 2$



Promoting piecewise constant profiles penalty

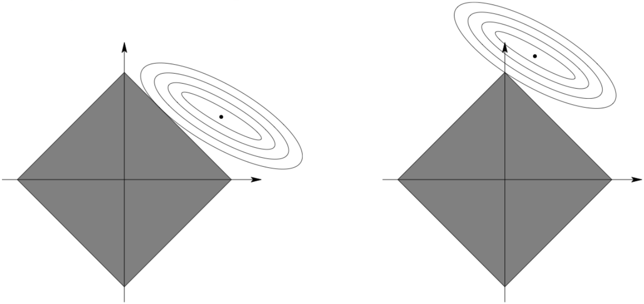
The variable fusion penalty (Land and Friedman, 1996)

The solution of

$$\min_{\beta \in \mathbb{R}^p} R(\beta) + \lambda \sum_{i=1}^{p-1} |\beta_{i+1} - \beta_i|$$

is usually piecewise constant.

Geometric interpretation with $p = 2$



A penalty for CGH array classification

The fused LASSO penalty (Tibshirani et al., 2005)

$$\Omega_{fusedlasso}(\beta) = \sum_i |\beta_i| + \sum_{i \sim j} |\beta_i - \beta_j| \, .$$

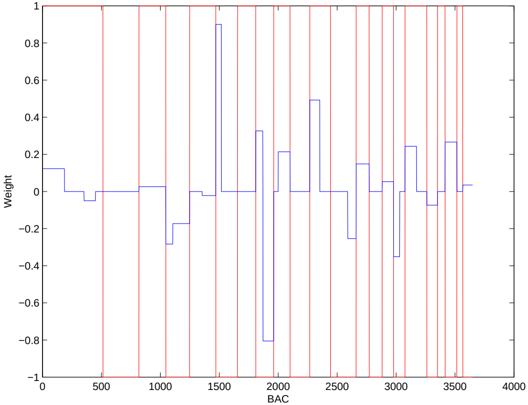
- First term leads to sparse solutions
- Second term leads to piecewise constant solutions

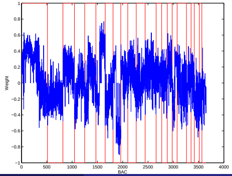
The fused SVM (Rapaport et al., 2008)

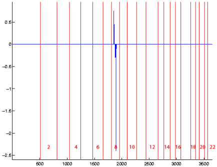
$$\min_{\beta \in \mathbb{R}^p} \sum_{i=1}^{n} \ell\left(y_i, \beta^\top x_i\right) + \lambda \sum_i |\beta_i| + \mu \sum_{i \sim j} |\beta_i - \beta_j| \, .$$

where ℓ is, e.g., the hinge loss $\ell(y,t) = max(1 - yt, 0)$. It is then a LP.

Application: predicting metastasis in melanoma



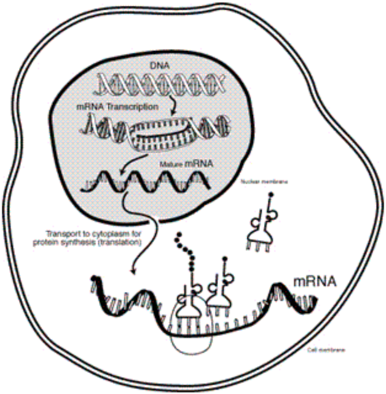


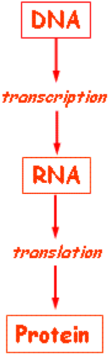


Outline

1. Cancer prognosis from DNA copy number variations
2. Diagnosis and prognosis from gene expression data
3. Virtual screening for drug discovery
4. Conclusion

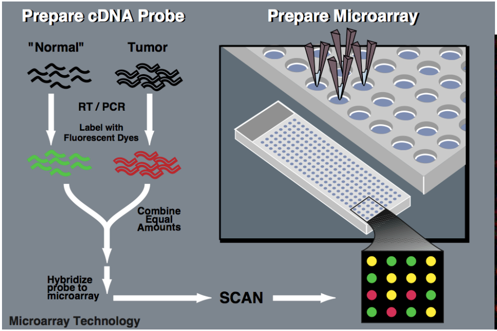
DNA → RNA → protein

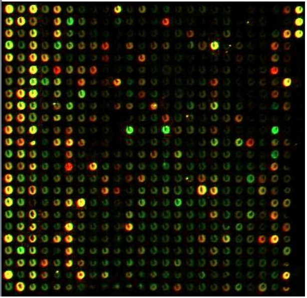



DNA
transcription
RNA
translation
Protein

- CGH shows the (static) DNA
- Cancer cells have also abnormal (dynamic) gene expression (= transcription)

Tissue profiling with DNA chips

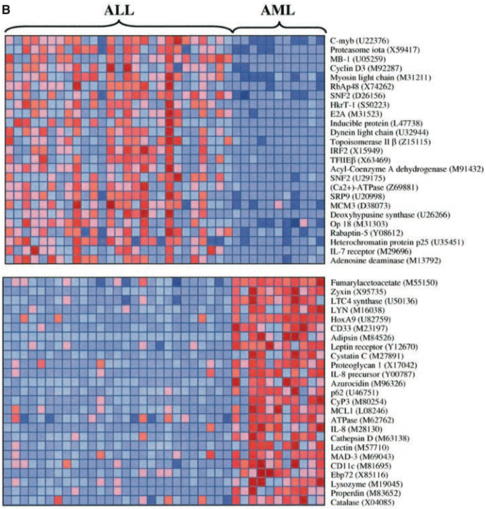




Data

- Gene expression measures for more than 10*k* genes
- Measured typically on less than 100 samples of two (or more) different classes (e.g., different tumors)

Tissue classification from microarray data



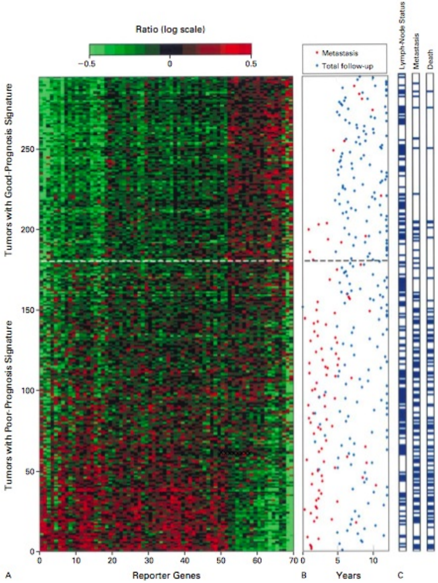
Goal

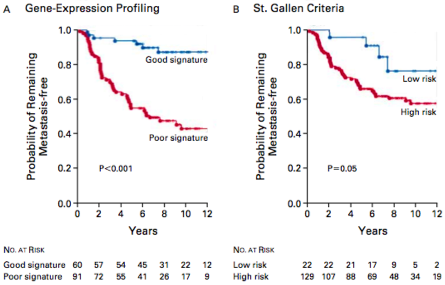
- Design a classifier to automatically assign a class to future samples from their expression profile
- Interpret biologically the differences between the classes

Difficulty

- Large dimension
- Few samples

Prognosis from microarray data (MAMMAPRINT)





No. at Risk

Good signature	60	57	54	45	31	22	12
Poor signature	91	72	55	41	26	17	9

No. at Risk

Low risk	22	22	21	17	9	5	2
High risk	129	107	88	69	48	34	19

Gene signature

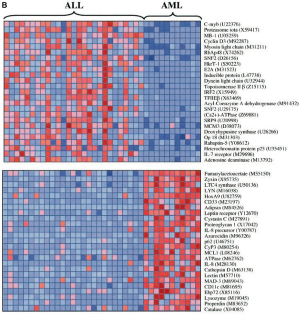
The idea

- We look for a limited set of genes that are sufficient for prediction.
- Equivalently, the linear classifier will be sparse

Motivations

- Bet on sparsity: we believe the "true" model is sparse.
- Interpretation: we will get a biological interpretation more easily by looking at the selected genes.
- Accuracy: by restricting the class of classifiers, we "increase the bias" but "decrease the variance". This should be helpful in large dimensions (it is better to estimate well a wrong model than estimate badly a good model).

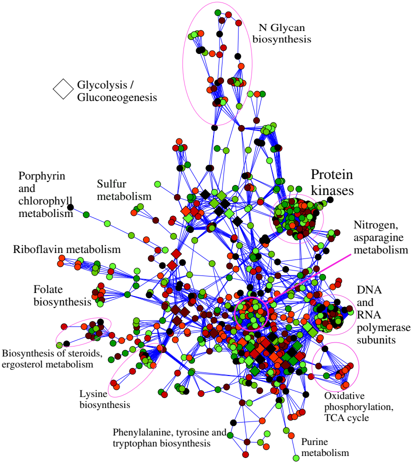
But...

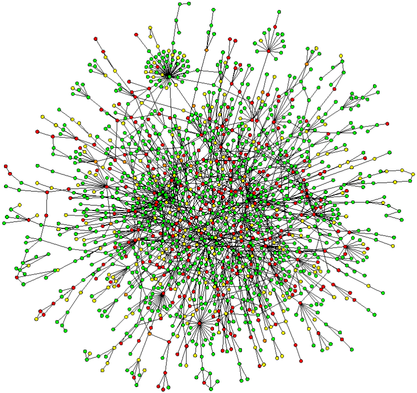


Challenging the idea of gene signature

- We often observe little stability in the genes selected...
- Is gene selection the most biologically relevant hypothesis?
- What about thinking instead of "pathways" or "modules" signatures?

Gene networks

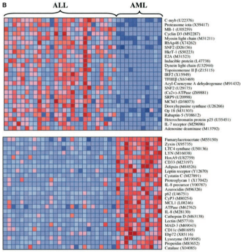

N Glycan biosynthesis
◇ Glycolysis / Gluconeogenesis
Porphyrin and chlorophyll metabolism
Sulfur metabolism
Protein kinases
Nitrogen, asparagine metabolism
Riboflavin metabolism
Folate biosynthesis
DNA and RNA polymerase subunits
Biosynthesis of steroids, ergosterol metabolism
Lysine biosynthesis
Oxidative phosphorylation, TCA cycle
Phenylalanine, tyrosine and tryptophan biosynthesis
Purine metabolism

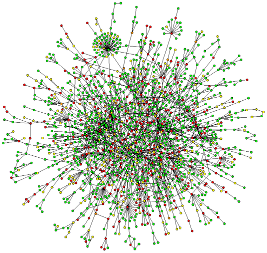


Gene networks and expression data

Motivation

- Basic biological functions usually involve the coordinated action of several proteins:
  - Formation of protein complexes
  - Activation of metabolic, signalling or regulatory pathways
- Many pathways and protein-protein interactions are already known
- Hypothesis: the weights of the classifier should be "coherent" with respect to this prior knowledge





Graph based penalty

Prior hypothesis

Genes near each other on the graph should have similar weigths.

Two solutions (Rapaport et al., 2007, 2008)

$$\Omega_{spectral}(\beta) = \sum_{i \sim j} (\beta_i - \beta_j)^2 \,,$$

$$\Omega_{graphfusion}(\beta) = \sum_{i \sim j} |\beta_i - \beta_j| + \sum_i |\beta_i| \,.$$

Graph based penalty

Prior hypothesis

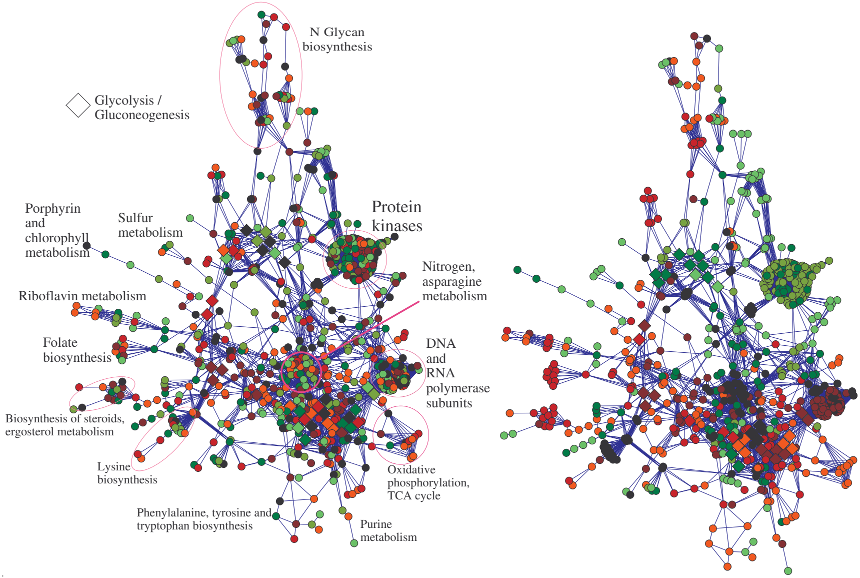
Genes near each other on the graph should have similar weigths.

Two solutions (Rapaport et al., 2007, 2008)

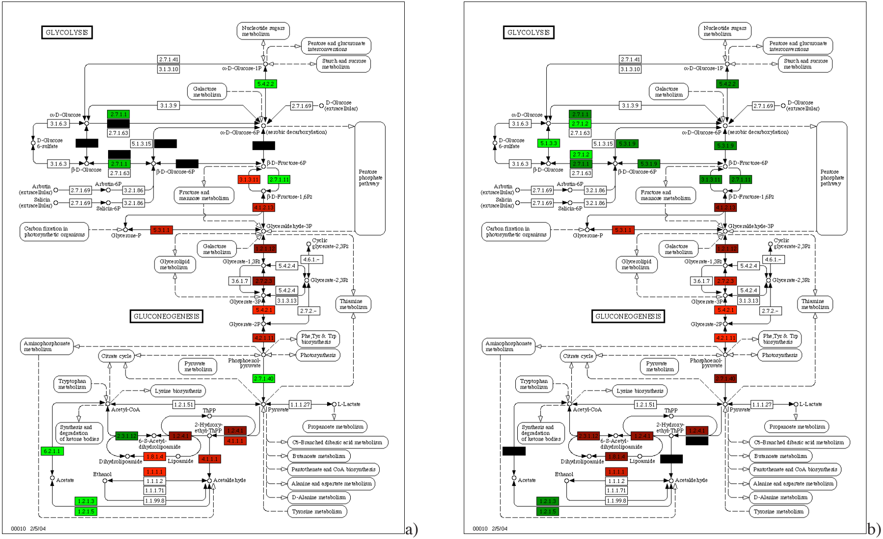
$$\Omega_{spectral}(\beta) = \sum_{i \sim j} (\beta_i - \beta_j)^2 \,,$$

$$\Omega_{graphfusion}(\beta) = \sum_{i \sim j} |\beta_i - \beta_j| + \sum_i |\beta_i| \,.$$

Classifiers


N Glycan biosynthesis
◇ Glycolysis / Gluconeogenesis
Porphyrin and chlorophyll metabolism
Sulfur metabolism
Protein kinases
Nitrogen, asparagine metabolism
Riboflavin metabolism
Folate biosynthesis
DNA and RNA polymerase subunits
Biosynthesis of steroids, ergosterol metabolism
Lysine biosynthesis
Oxidative phosphorylation, TCA cycle
Phenylalanine, tyrosine and tryptophan biosynthesis
Purine metabolism

Classifier


GLYCOLYSIS
GLUCONEOGENESIS
Nucleotide sugars metabolism
Pentose and glucuronate interconversions
Starch and sucrose metabolism
Galactose metabolism
Pentose phosphate pathway
Fructose and mannose metabolism
Carbon fixation in photosynthetic organisms
Glycerolipid metabolism
Thiamine metabolism
Aminophosphonate metabolism
Citrate cycle
Pyruvate metabolism
Phe,Tyr & Trp biosynthesis
Photosynthesis
Tryptophan metabolism
Lysine biosynthesis
Synthesis and degradation of ketone bodies
Propanoate metabolism
C5-Branched dibasic acid metabolism
Butanoate metabolism
Pantothenate and CoA biosynthesis
Alanine and aspartate metabolism
D-Alanine metabolism
Tyrosine metabolism
α-D-Glucose-1P
α-D-Glucose
D-Glucose (extracellular)
α-D-Glucose-6P
(aerobic decarboxylation)
D-Glucose 6-sulfate
β-D-Glucose
β-D-Glucose-6P
β-D-Fructose-6P
β-D-Fructose-1,6P2
Arbutin (extracellular)
Arbutin-6P
Salicin (extracellular)
Salicin-6P
Glycerone-P
Glyceraldehyde-3P
Glycerate-1,3P2
Cyclic glycerate-2,3P2
Glycerate-2,3P2
Glycerate-3P
Glycerate-2P
Phosphoenol-pyruvate
Pyruvate
L-Lactate
Acetyl-CoA
ThPP
2-Hydroxy-ethyl-ThPP
6-S-Acetyl-dihydrolipoamide
Dihydrolipoamide
Lipoamide
Acetate
Ethanol
Acetaldehyde
2.7.1.41
3.1.3.10
5.4.2.2
3.1.3.9
2.7.1.69
2.7.1.1
2.7.1.2
2.7.1.63
3.1.6.3
5.1.3.15
5.1.3.3
5.3.1.9
3.1.3.11
2.7.1.11
3.2.1.86
4.1.2.13
5.3.1.1
1.2.1.12
4.6.1.-
5.4.2.4
3.6.1.7
2.7.2.3
3.1.3.13
2.7.2.-
5.4.2.1
4.2.1.11
2.7.1.40
1.2.1.51
1.1.1.27
2.3.1.12
1.2.4.1
4.1.1.1
1.8.1.4
6.2.1.1
1.1.1.1
1.1.1.2
1.1.1.71
1.1.99.8
1.2.1.3
1.2.1.5
00010 2/5/04
a)
b)

Example: finding discriminant modules in gene networks

Prior hypothesis

Genes near each other on the graph should have non-zero weigths (i.e., the support of β should be made of a few connected components).

Two solutions?

$$\Omega_{intersection}(\beta) = \sum_{i\sim j} \sqrt{\beta_i^2 + \beta_j^2}\,,$$

$$\Omega_{union}(\beta) = \sup_{\alpha\in\mathbb{R}^p:\forall i\sim j,\|\alpha_i^2+\alpha_j^2\|\leq 1} \alpha^\top \beta\,.$$

Example: finding discriminant modules in gene networks

Prior hypothesis

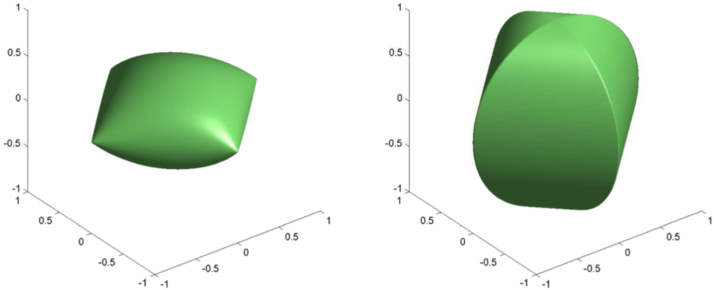
Genes near each other on the graph should have non-zero weigths (i.e., the support of β should be made of a few connected components).

Two solutions?

$$\Omega_{intersection}(\beta) = \sum_{i \sim j} \sqrt{\beta_i^2 + \beta_j^2} \,,$$

$$\Omega_{union}(\beta) = \sup_{\alpha \in \mathbb{R}^p : \forall i \sim j, \| \alpha_i^2 + \alpha_j^2 \| \leq 1} \alpha^\top \beta \,.$$

Example: finding discriminant modules in gene networks



Groups $(1,2)$ *and* $(2,3)$. *Left:* $\Omega_{intersection}(\beta)$. *Right:* $\Omega_{union}(\beta)$. *Vertical axis is* β_2.

Graph lasso vs kernel on graph

- Graph lasso:

$$\Omega_{\text{graph lasso}}(w) = \sum_{i \sim j} \sqrt{w_i^2 + w_j^2} \,.$$

constrains the sparsity, not the values

- Graph kernel

$$\Omega_{\text{graph kernel}}(w) = \sum_{i \sim j} (w_i - w_j)^2 \,.$$

constrains the values (smoothness), not the sparsity

Preliminary results

Breast cancer data

- Gene expression data for 8, 141 genes in 295 breast cancer tumors.
- Canonical pathways from MSigDB containing 639 groups of genes, 637 of which involve genes from our study.

METHOD	ℓ_1	$\Omega_{group}.$
ERROR	0.38 ± 0.04	0.36 ± 0.03
♯ PATH.	$148, 58, 183$	$6, 5, 78$
PROP. PATH.	$0.32, 0.14, 0.41$	$0.01, 0.01, 0.17$

- Graph on the genes.

METHOD	ℓ_1	$\Omega_{graph}(.)$
ERROR	0.39 ± 0.04	0.36 ± 0.01
AV. SIZE C.C.	$1.1, 1, 1.0$	$1.3, 1.4, 1.2$

Outline

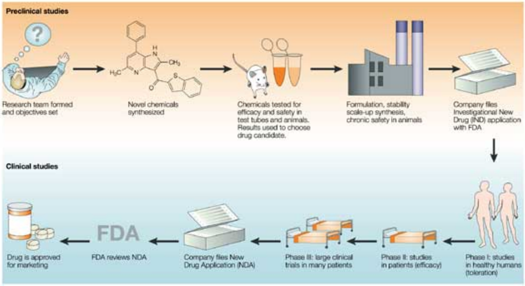
1 Cancer prognosis from DNA copy number variations

2 Diagnosis and prognosis from gene expression data

3 Virtual screening for drug discovery

4 Conclusion

Drug discovery



Preclinical studies

Research team formed and objectives set

Novel chemicals synthesized

Chemicals tested for efficacy and safety in test tubes and animals. Results used to choose drug candidate.

Formulation, stability scale-up synthesis, chronic safety in animals

Company files Investigational New Drug (IND) application with FDA

Clinical studies

Phase I: studies in healthy humans (toleration)

Phase II: studies in patients (efficacy)

Phase III: large clinical trials in many patients

Company files New Drug Application (NDA)

FDA reviews NDA

Drug is approved for marketing

Nature Reviews | Drug Discovery

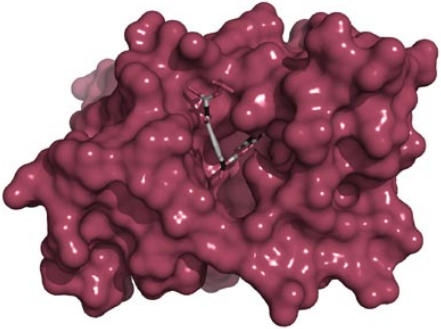
A long, expensive and risky process

- On average 15 years and $800 millions
- High attrition rate: for 10,000 molecules tested, 10 make it to clinicals, 1 to the market.
- >70% of the costs are wasted on failures

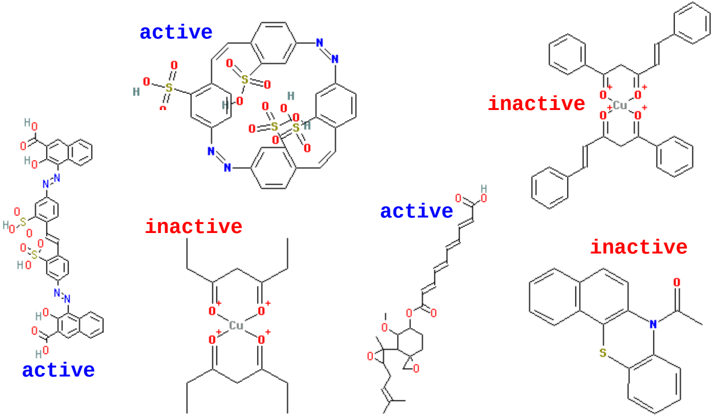
Computational approaches

The use of computers and computational methods permeates all aspects of drug discovery today, in particular for:

- Target identification
- Structure prediction, virtual screening (docking)
- Prediction of drug-likeliness of compounds



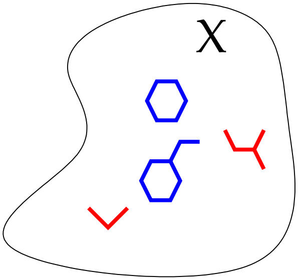
Example : ligand-Based Virtual Screening



NCI AIDS screen results (from http://cactus.nci.nih.gov).

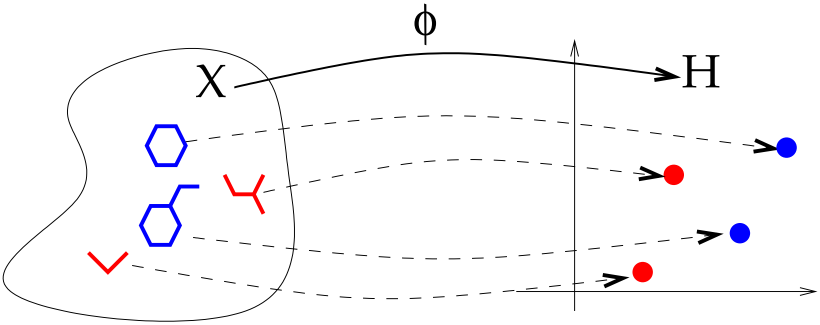
The machine learning approach

1. Represent explicitly each graph x by a vector of fixed dimension $\Phi(x) \in \mathbb{R}^p$.
2. Use an algorithm for regression or pattern recognition in $\mathbb{R}^p$.



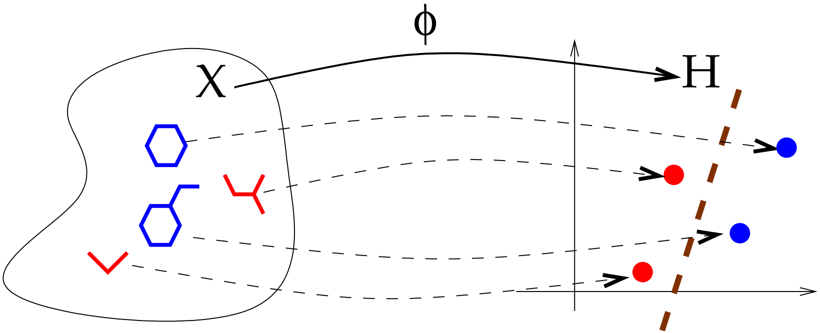
The machine learning approach

1. Represent explicitly each graph x by a vector of fixed dimension $\Phi(x) \in \mathbb{R}^p$.
2. Use an algorithm for regression or pattern recognition in $\mathbb{R}^p$.



The machine learning approach

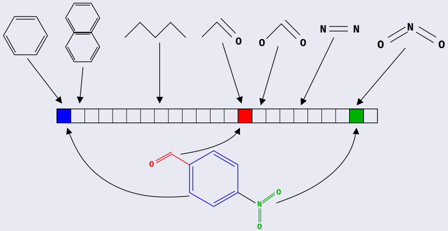
1. Represent explicitly each graph x by a vector of fixed dimension $\Phi(x) \in \mathbb{R}^p$.
2. Use an algorithm for regression or pattern recognition in $\mathbb{R}^p$.



Example

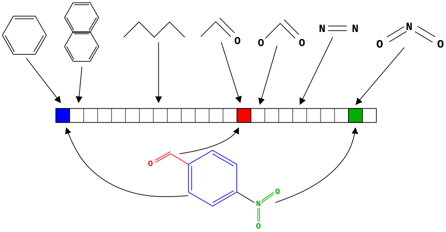
2D structural keys in chemoinformatics

- Index a molecule by a binary fingerprint defined by a limited set of pre-defined stuctures



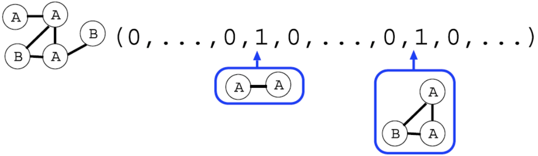
- Use a machine learning algorithms such as SVM, NN, PLS, decision tree, ...

Challenge: which descriptors (patterns)?



- Expressiveness: they should retain as much information as possible from the graph
- Computation : they should be fast to compute
- Large dimension of the vector representation: memory storage, speed, statistical issues

Indexing by all subgraphs?

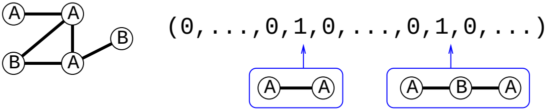


(0, . . . ,0,1,0, . . . ,0,1,0, . . .)

Theorem

Computing all subgraph occurrences is NP-hard.

Indexing by all paths?



Theorem

Computing all path occurrences is NP-hard.

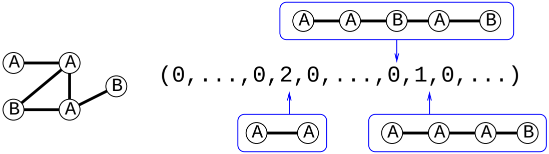
Indexing by what?

Substructure selection

We can imagine more limited sets of substuctures that lead to more computationnally efficient indexing (non-exhaustive list)

- substructures selected by domain knowledge (MDL fingerprint)
- all path up to length *k* (Openeye fingerprint, Nicholls 2005)
- all shortest paths (Borgwardt and Kriegel, 2005)
- all subgraphs up to *k* vertices (graphlet kernel, Sherashidze et al., 2009)
- all frequent subgraphs in the database (Helma et al., 2004)

Example : Indexing by all shortest paths

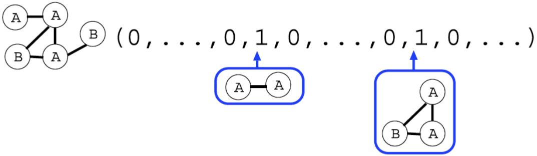


(0,...,0,2,0,...,0,1,0,...)

Properties (Borgwardt and Kriegel, 2005)

- There are $O(n^2)$ shortest paths.
- The vector of counts can be computed in $O(n^4)$ with the Floyd-Warshall algorithm.

Example : Indexing by all subgraphs up to *k* vertices

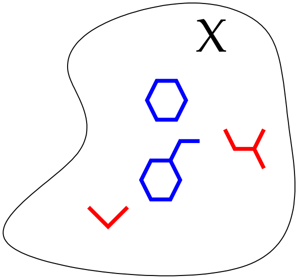


Properties (Shervashidze et al., 2009)

- Naive enumeration scales as $O(n^k)$.
- Enumeration of connected graphlets in $O(nd^{k-1})$ for graphs with degree $\leq d$ and $k \leq 5$.
- Randomly sample subgraphs if enumeration is infeasible.

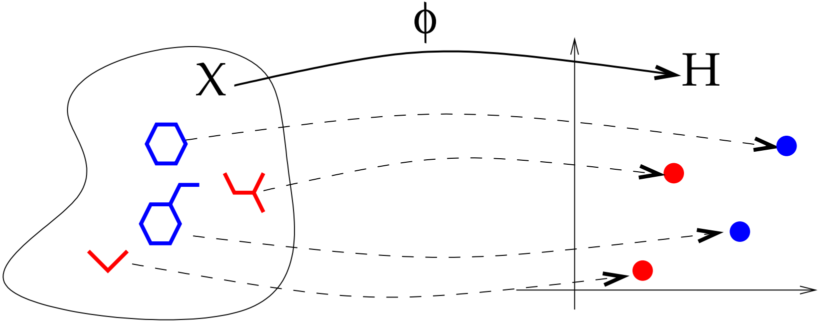
Graph kernels

1. Represent *implicitly* each graph x by a vector $\Phi(x) \in \mathcal{H}$ through the kernel
$$K(x, x') = \Phi(x)^\top \Phi(x') .$$
2. Use a kernel method for classification in $\mathcal{H}$.



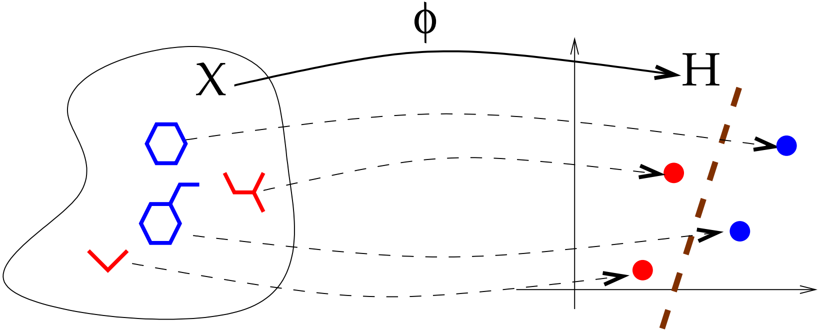
Graph kernels

1. Represent implicitly each graph x by a vector $\Phi(x) \in \mathcal{H}$ through the kernel
$$K(x,x') = \Phi(x)^\top \Phi(x') .$$
2. Use a kernel method for classification in $\mathcal{H}$.



Graph kernels

1. Represent implicitly each graph x by a vector $\Phi(x) \in \mathcal{H}$ through the kernel
$$K(x, x') = \Phi(x)^\top \Phi(x') .$$
2. Use a kernel method for classification in $\mathcal{H}$.

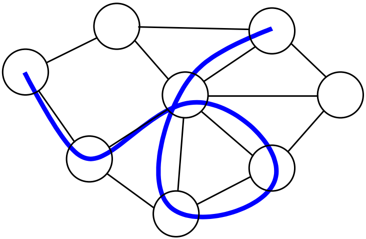


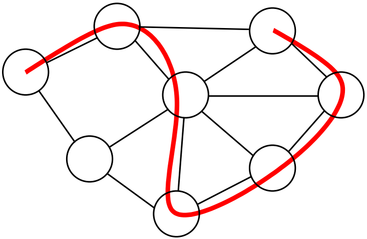
Does the "kernel trick" help?

Unfortunately...

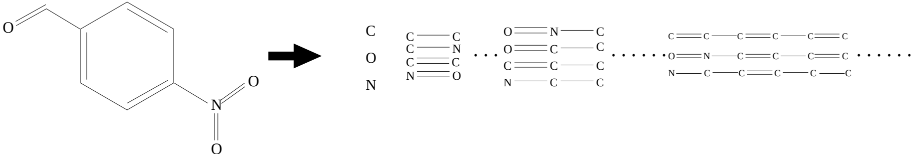
- It is intractable to compute complete graph kernels (which separate non-isomorphic graphs)
- It is intractable to compute the subgraph kernels (NP-hard).
- It is intractable to compute the path kernel (NP-hard).

Walks $\neq$ paths





2D walk kernel



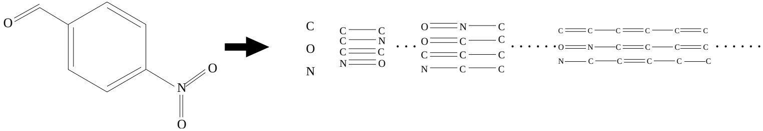
- $\phi_d(x)$ is the vector of counts of all walks of length d:

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

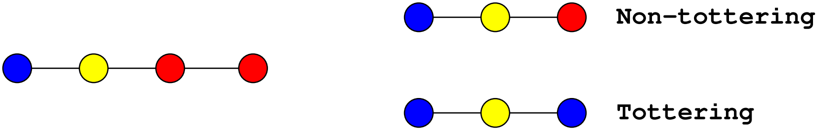
- The 2D fragment kernel is defined by

$$K_{walk}(x, x') = \sum_{d=1}^{\infty} \lambda_d \phi_d(x)^\top \phi_d(x') \, .$$

2D walk kernel in practice

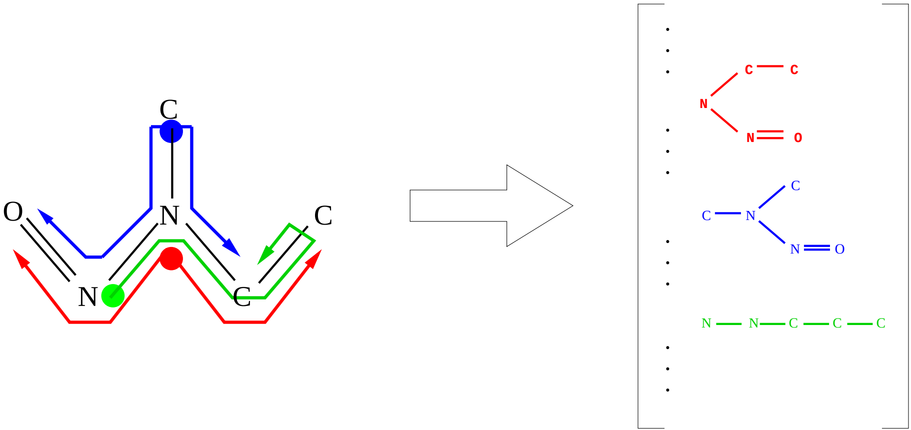


- K_{walk} can be computed efficiently for various weightings, although the feature space has infinite dimension.
- Selecting only walks with no backward moves ("non-tottering") can be done efficiently and improves performance.

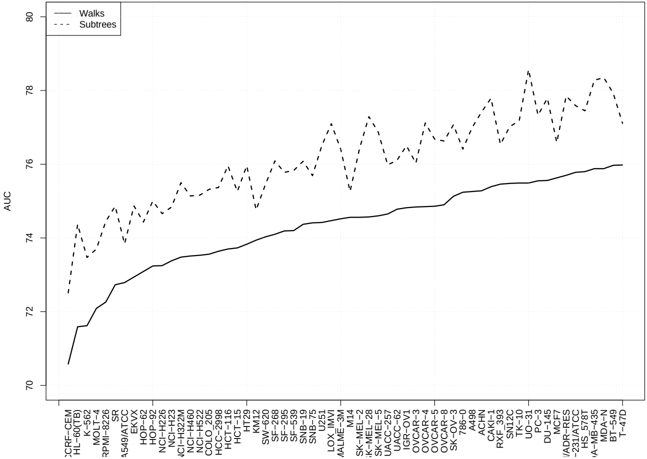

Non-tottering

Tottering

Extension: 2D subtree kernel

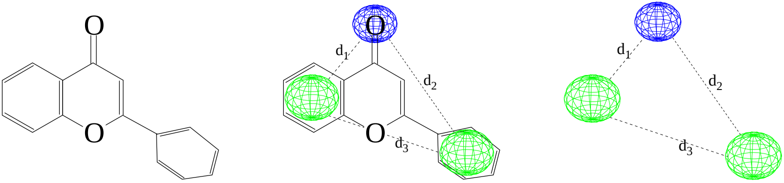


2D Subtree vs fragment kernels (Mahé and V, 2007)



Screening of inhibitors for 60 cancer cell lines (from Mahé and V., 2008)

Example: 3D pharmacophore kernel (Mahé et al., 2005)



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

Results (accuracy)

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**

Outline

1. Cancer prognosis from DNA copy number variations
2. Diagnosis and prognosis from gene expression data
3. Virtual screening for drug discovery
4. Conclusion

Conclusion

- Modern machine learning methods play an increasing role in bio- and chemo-informatics
- The development of dedicated method is increasingly important to overcome the challenges (few samples, high-dimension, structures..)
- This increasingly requires tight collaboration with domain experts