

# Including prior knowledge in shrinkage classifiers for genomic data

Jean-Philippe Vert

Jean-Philippe.Vert@mines-paristech.fr

Mines ParisTech / Institut Curie / Inserm

Statistical Advances in Genome-scale Data Analysis workshop,  
Ascona, Switzerland, May 3-8, 2009.

- 1 Supervised classification of genomic data
- 2 Classification of array CGH data
- 3 Classification of expression data using gene networks
- 4 Conclusion

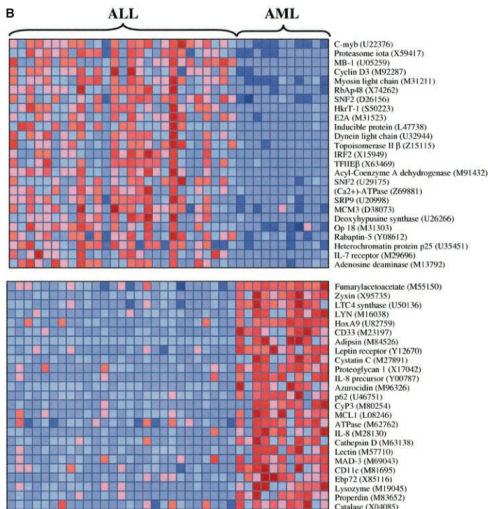
- 1 Supervised classification of genomic data
- 2 Classification of array CGH data
- 3 Classification of expression data using gene networks
- 4 Conclusion

- 1 Supervised classification of genomic data
- 2 Classification of array CGH data
- 3 Classification of expression data using gene networks
- 4 Conclusion

- 1 Supervised classification of genomic data
- 2 Classification of array CGH data
- 3 Classification of expression data using gene networks
- 4 Conclusion

- 1 Supervised classification of genomic data
- 2 Classification of array CGH data
- 3 Classification of expression data using gene networks
- 4 Conclusion

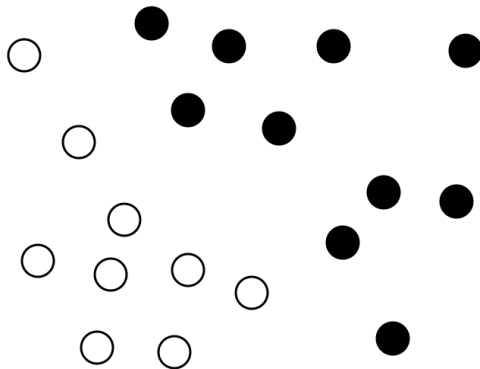
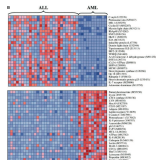
# Motivation



## Goal

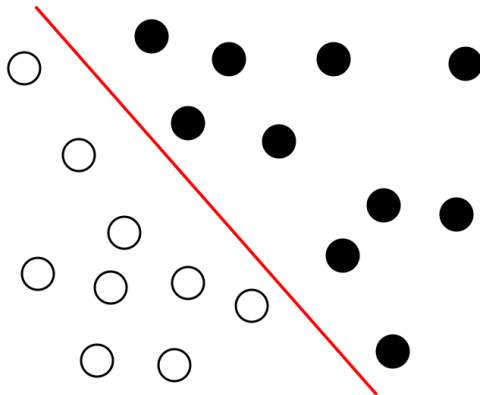
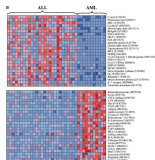
- Design a **classifier** to automatically assign a class to future samples from their expression profile
- **Interpret** biologically the differences between the classes

# Pattern recognition, *aka* supervised classification

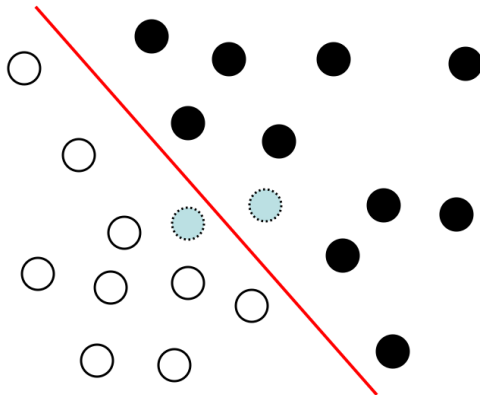
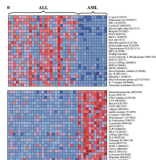




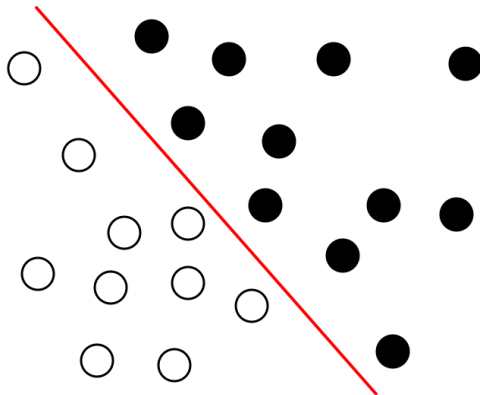
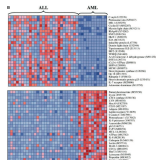
# Pattern recognition, *aka* supervised classification



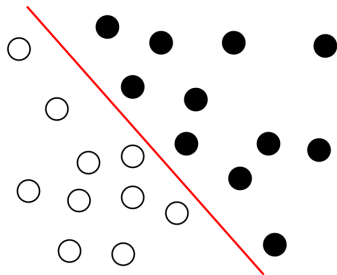
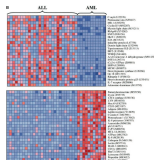
# Pattern recognition, *aka* supervised classification



# 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

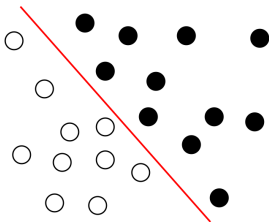
# Linear classifiers

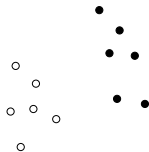
## The model

- Each sample is represented by a vector  $x = (x_1, \dots, x_p)$
- **Goal**: estimate a linear function:

$$f_{\beta}(x) = \sum_{i=1}^p \beta_i x_i + \beta_0 .$$

- **Interpretability**: the weight  $\beta_i$  quantifies the influence of feature  $i$  (but...)





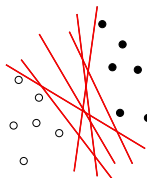
## Training the model

- Minimize an **empirical risk** on the training samples:

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

- ... subject to some **constraint** on  $\beta$ , e.g.:

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



## Training the model

- Minimize an **empirical risk** on the training samples:

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

- ... subject to some **constraint** on  $\beta$ , e.g.:

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

# Example : Norm Constraints

## The approach

A common method in statistics to learn with few samples in high dimension is to **constrain the Euclidean norm of  $\beta$**

$$\Omega_{\text{ridge}}(\beta) = \|\beta\|_2^2 = \sum_{i=1}^p \beta_i^2,$$

(ridge regression, support vector machines...)

### Pros

- Good performance in classification

### Cons

- Limited interpretation (small weights)
- No prior biological knowledge



# Example : Feature Selection

## The approach

Constrain most weights to be 0, i.e., **select a few genes** whose expression are sufficient for classification.

$$\Omega_{\text{Best subset selection}}(\beta) = \|\beta\|_0 = \sum_{i=1}^p \mathbf{1}(\beta_i > 0).$$

This is usually a NP-hard problem, many greedy variants have been proposed (filter methods, wrapper methods)

### Pros

- Good performance
- **Biomarker** selection
- Interpretability

### Cons

- NP-hard
- Gene selection **not robust**
- No use of prior knowledge

# Example : Sparsity inducing convex priors

## The approach

Constrain most weights to be 0 through a convex non-differentiable penalty:

$$\Omega_{\text{LASSO}}(\beta) = \|\beta\|_1 = \sum_{i=1}^p |\beta_i| .$$

- Greedy feature selection (T-tests, ...) Several variants exist, e.g., **elastic net** penalty ( $\|\beta\|_1 + \|\beta\|_2$ ), ... )

## Pros

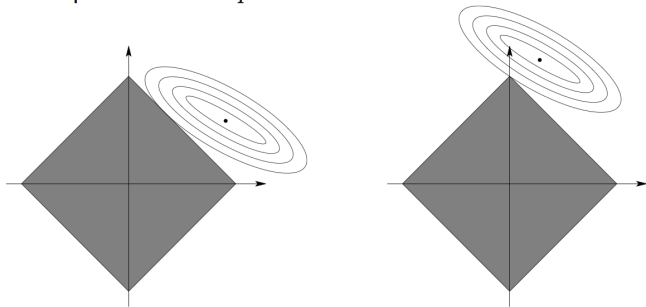
- Good performance
- **Biomarker** selection
- Interpretability

## Cons

- Gene selection **not robust**
- No use of prior knowledge

# Why LASSO leads to sparse solutions

Geometric interpretation with  $p = 2$



## The idea

- If we have a specific **prior knowledge** about the “correct” weights, it can be included in  $\Omega$  in the constraint:

Minimize  $R_{emp}(\beta)$  subject to  $\Omega(\beta) \leq C$ .

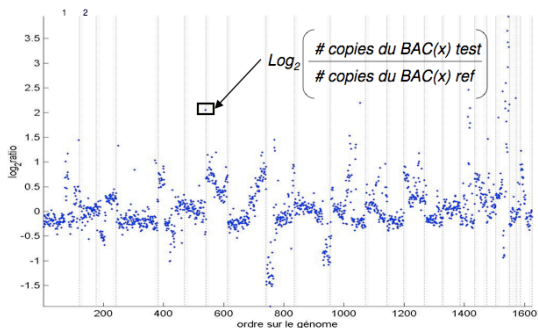
- If we design a **convex** function  $\Omega$ , then the algorithm boils down to a convex optimization problem (usually **easy to solve**).
- Similar to priors in Bayesian statistics

- 1 Supervised classification of genomic data
- 2 Classification of array CGH data**
- 3 Classification of expression data using gene networks
- 4 Conclusion

# 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

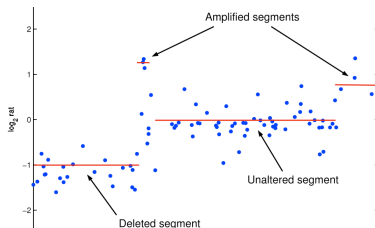
# Classification of array CGH

## Prior knowledge

- Let  $\mathbf{x}$  be a CGH profile
- We focus on linear classifiers, i.e., the sign of :

$$f(\mathbf{x}) = \mathbf{x}^\top \beta.$$

- We expect  $\beta$  to be
  - **sparse** : only a few positions should be discriminative
  - **piecewise constant** : within a region, all probes should contribute equally



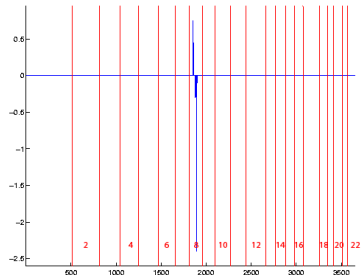
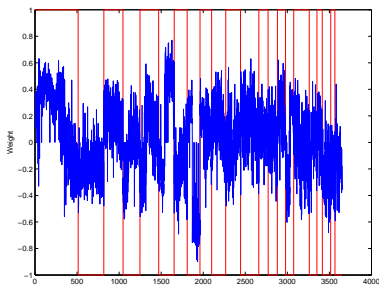
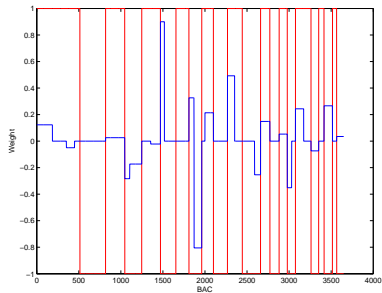
## The fused LASSO penalty (Tibshirani et al., 2005)

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

- First term leads to **sparse** solutions
- Second term leads to **piecewise constant** solutions
- Combined with a hinge loss leads to a **fused SVM** (Rapaport et al., 2008);

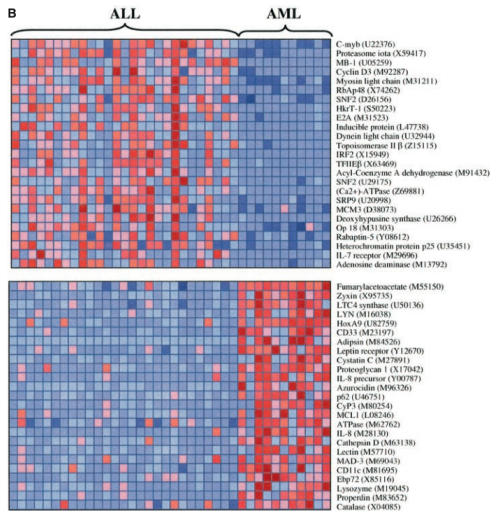


# Application: metastasis prognosis in melanoma

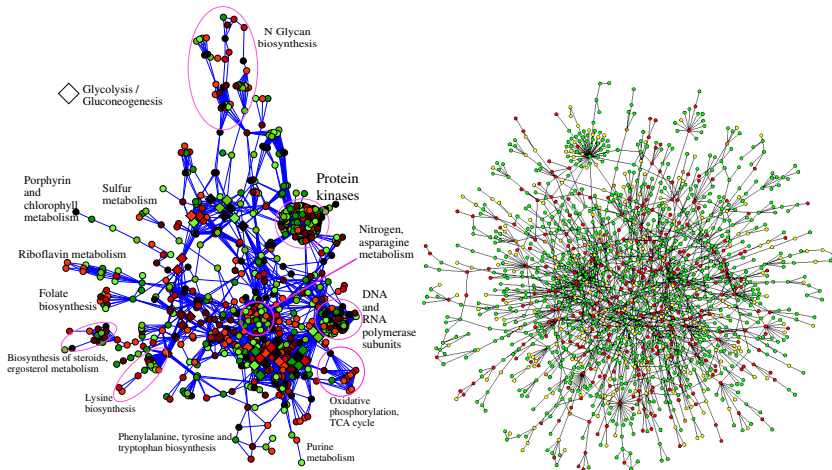


- 1 Supervised classification of genomic data
- 2 Classification of array CGH data
- 3 Classification of expression data using gene networks**
- 4 Conclusion

# Tissue classification from microarray data



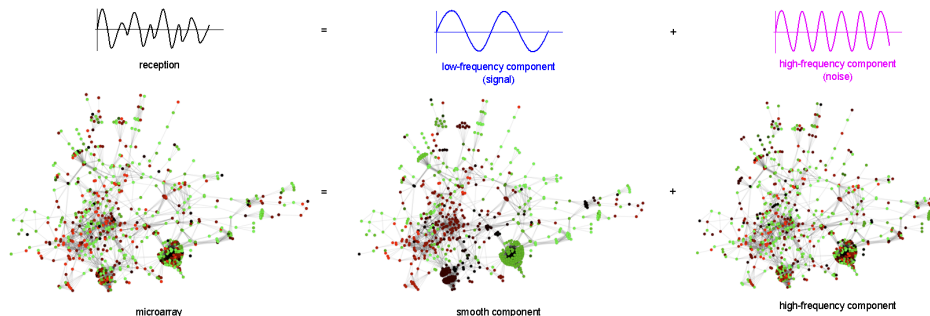
# Gene networks



## 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**

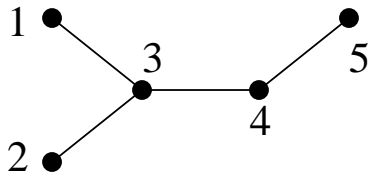
# An idea



- 1 Use the gene network to extract the “important information” in gene expression profiles by **Fourier analysis** on the graph
- 2 Learn a linear classifier on the **smooth components**

## Definition

The Laplacian of the graph is the matrix  $L = D - A$ .



$$L = D - A = \begin{pmatrix} 1 & 0 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ -1 & -1 & 3 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

- $L$  is positive semidefinite
- The **eigenvectors**  $\mathbf{e}_1, \dots, \mathbf{e}_n$  of  $L$  with eigenvalues  $0 = \lambda_1 \leq \dots \leq \lambda_n$  form a basis called **Fourier basis**
- For any  $f : V \rightarrow \mathbb{R}$ , the **Fourier transform** of  $f$  is the vector  $\hat{f} \in \mathbb{R}^n$  defined by:

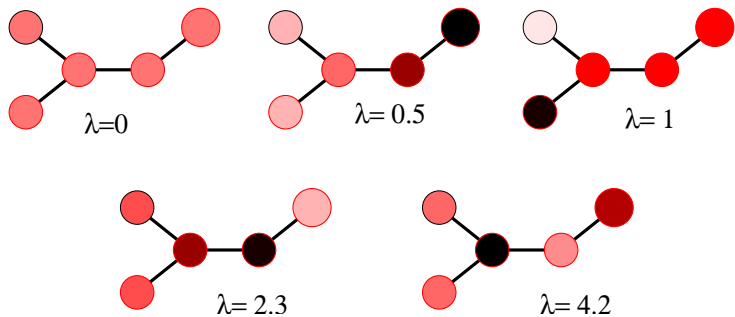
$$\hat{f}_i = f^\top \mathbf{e}_i, \quad i = 1, \dots, n.$$

- The **inverse Fourier formula** holds:

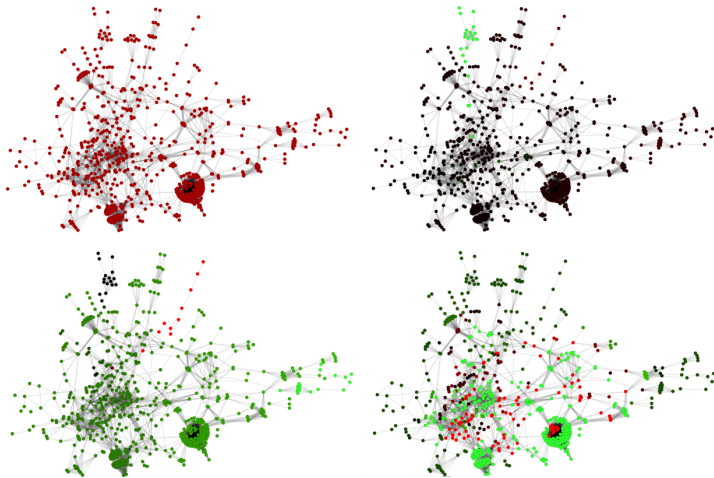
$$f = \sum_{i=1}^n \hat{f}_i \mathbf{e}_i.$$



# Fourier basis



# Fourier basis



## Definition

- Let  $\phi : \mathbb{R}^+ \rightarrow \mathbb{R}^+$  be **non-increasing**.
- A smoothing operator  $S_\phi$  transform a function  $f : V \rightarrow \mathbb{R}$  into a smoothed version:

$$S_\phi(f) = \sum_{i=1}^n \hat{f}_i \phi(\lambda_i) e_i.$$

## Examples

- Identity operator ( $S_\phi(f) = f$ ):

$$\phi(\lambda) = 1, \quad \forall \lambda$$

- Low-pass filter:

$$\phi(\lambda) = \begin{cases} 1 & \text{if } \lambda \leq \lambda^*, \\ 0 & \text{otherwise.} \end{cases}$$

- Attenuation of high frequencies:

$$\phi(\lambda) = \exp(-\beta\lambda).$$

## Examples

- Identity operator ( $S_\phi(f) = f$ ):

$$\phi(\lambda) = 1, \quad \forall \lambda$$

- Low-pass filter:

$$\phi(\lambda) = \begin{cases} 1 & \text{if } \lambda \leq \lambda^*, \\ 0 & \text{otherwise.} \end{cases}$$

- Attenuation of high frequencies:

$$\phi(\lambda) = \exp(-\beta\lambda).$$

## Examples

- Identity operator ( $S_\phi(f) = f$ ):

$$\phi(\lambda) = 1, \quad \forall \lambda$$

- Low-pass filter:

$$\phi(\lambda) = \begin{cases} 1 & \text{if } \lambda \leq \lambda^*, \\ 0 & \text{otherwise.} \end{cases}$$

- Attenuation of high frequencies:

$$\phi(\lambda) = \exp(-\beta\lambda).$$

## Working with smoothed profiles

- Classical methods for linear classification and regression with a ridge penalty solve:

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

- Applying these algorithms on the smooth profiles means solving:

$$\min_{\beta \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^n l(\beta^\top \mathbf{S}_\phi(f_i), y_i) + \lambda \beta^\top \beta.$$

## Lemma

This is equivalent to:

$$\min_{v \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^n l(v^\top f_i, y_i) + \lambda \sum_{i=1}^p \frac{\hat{v}_i^2}{\phi(\lambda_i)},$$

hence the linear classifier  $v$  is **smooth**.

## Proof

- Let  $v = \sum_{i=1}^n \phi(\lambda_i) e_i e_i^\top \beta$ , then

$$\beta^\top \mathcal{S}_\phi(f_i) = \beta^\top \sum_{i=1}^n \hat{f}_i \phi(\lambda_i) e_i = f_i^\top v.$$

- Then  $\hat{v}_i = \phi(\lambda_i) \hat{\beta}_i$  and  $\beta^\top \beta = \sum_{i=1}^n \frac{\hat{v}_i^2}{\phi(\lambda_i)^2}$ .



## Lemma

This is equivalent to:

$$\min_{v \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^n l(v^\top f_i, y_i) + \lambda \sum_{i=1}^p \frac{\hat{v}_i^2}{\phi(\lambda_i)},$$

hence the linear classifier  $v$  is **smooth**.

## Proof

- Let  $v = \sum_{i=1}^n \phi(\lambda_i) e_i e_i^\top \beta$ , then

$$\beta^\top S_\phi(f_i) = \beta^\top \sum_{i=1}^n \hat{f}_i \phi(\lambda_i) e_i = f_i^\top v.$$

- Then  $\hat{v}_i = \phi(\lambda_i) \hat{\beta}_i$  and  $\beta^\top \beta = \sum_{i=1}^n \frac{\hat{v}_i^2}{\phi(\lambda_i)^2}$ .

## Smoothing kernel

Kernel methods (SVM, kernel ridge regression..) only need the **inner product** between smooth profiles:

$$\begin{aligned}K(f, g) &= S_\phi(f)^\top S_\phi(g) \\&= \sum_{i=1}^n \hat{f}_i \hat{g}_i \phi(\lambda_i)^2 \\&= f^\top \left( \sum_{i=1}^n \phi(\lambda_i)^2 \mathbf{e}_i \mathbf{e}_i^\top \right) g \\&= f^\top K_\phi g,\end{aligned}\tag{1}$$

with

$$K_\phi = \sum_{i=1}^n \phi(\lambda_i)^2 \mathbf{e}_i \mathbf{e}_i^\top.$$

- For  $\phi(\lambda) = \exp(-t\lambda)$ , we recover the **diffusion kernel**:

$$K_\phi = \exp_M(-2tL).$$

- For  $\phi(\lambda) = 1/\sqrt{1+\lambda}$ , we obtain

$$K_\phi = (L + I)^{-1},$$

and the penalization is:

$$\sum_{i=1}^n \frac{\hat{v}_i^2}{\phi(\lambda_i)} = \mathbf{v}^\top (L + I) \mathbf{v} = \|\mathbf{v}\|_2^2 + \sum_{i \sim j} (v_i - v_j)^2.$$

- For  $\phi(\lambda) = \exp(-t\lambda)$ , we recover the **diffusion kernel**:

$$K_\phi = \exp_M(-2tL).$$

- For  $\phi(\lambda) = 1/\sqrt{1+\lambda}$ , we obtain

$$K_\phi = (L + I)^{-1},$$

and the penalization is:

$$\sum_{i=1}^n \frac{\hat{v}_i^2}{\phi(\lambda_i)} = \mathbf{v}^\top (L + I) \mathbf{v} = \|\mathbf{v}\|_2^2 + \sum_{i \sim j} (v_i - v_j)^2.$$

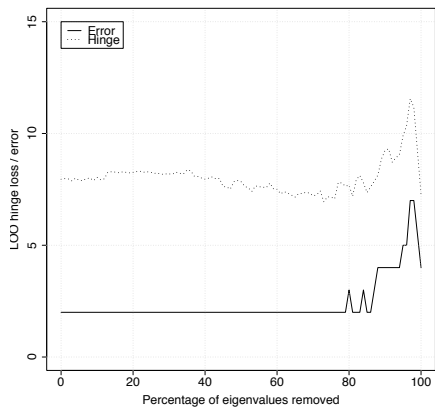
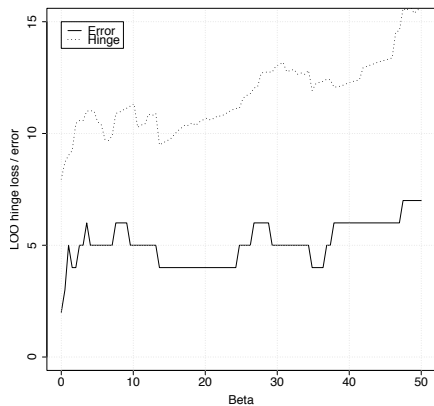
## Expression

- Study the effect of low irradiation doses on the yeast
- 12 non irradiated vs 6 irradiated
- Which pathways are involved in the response at the transcriptomic level?

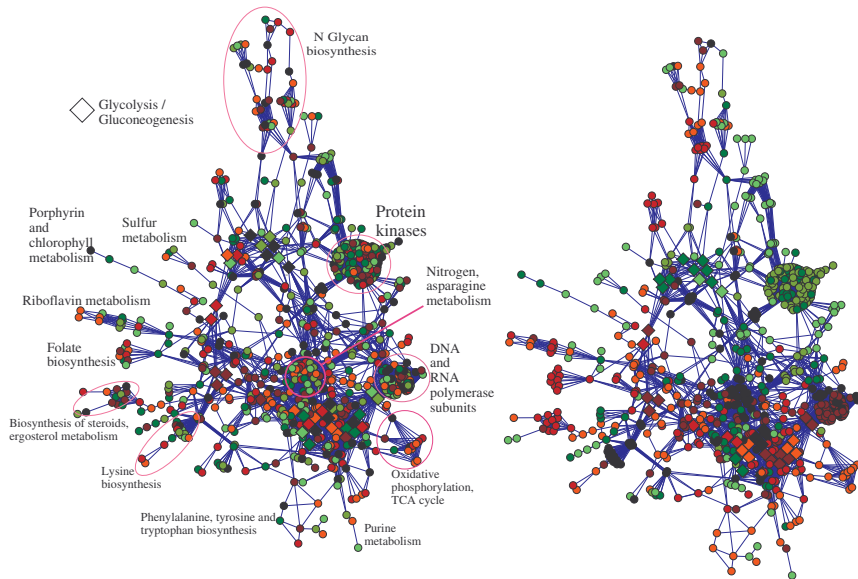
## Graph

- KEGG database of metabolic pathways
- Two genes are connected if they code for enzymes that catalyze successive reactions in a pathway (**metabolic gene network**).
- 737 genes, 4694 vertices.

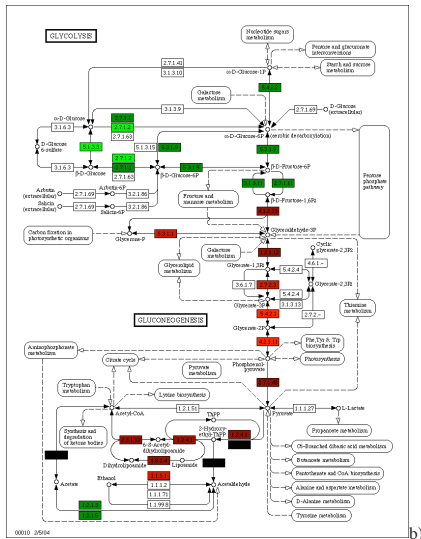
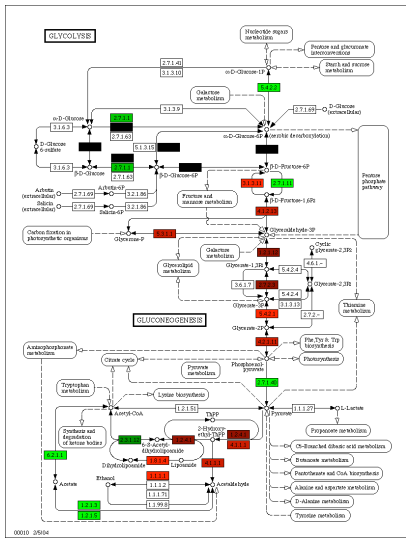
# Classification performance



# Classifier



# Classifier





## Prior hypothesis

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

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

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

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

## Prior hypothesis

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

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

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

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

## Prior hypothesis

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

## Graph Lasso (Jacob et al., 2009)

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

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

## Prior hypothesis

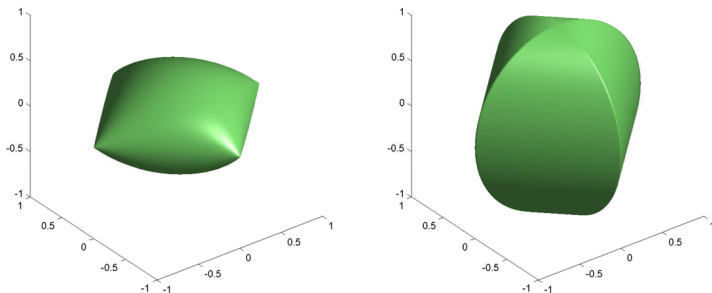
Genes near each other on the graph should have non-zero weights (i.e., the support of  $\beta$  should be made of a few connected components).

## Graph Lasso (Jacob et al., 2009)

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

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

# Example: finding discriminant modules in gene networks

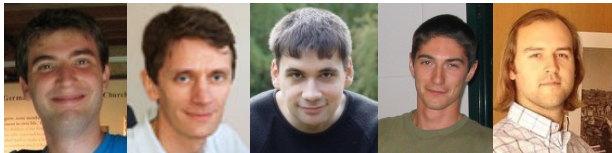


*Groups (1, 2) and (2, 3). Left:  $\Omega_{\text{intersection}}(\beta)$ . Right:  $\Omega_{\text{union}}(\beta)$ . Vertical axis is  $\beta_2$ .*

- 1 Supervised classification of genomic data
- 2 Classification of array CGH data
- 3 Classification of expression data using gene networks
- 4 Conclusion**

- Modern machine learning methods for regression / classification lend themselves well to the **integration of prior knowledge** in the penalization / regularization function.
- Several **computationally efficient** approaches (structured LASSO, kernels...)
- Natural extension to **data integration**

# People I need to thank



- Franck Rapaport, Emmanuel Barillot, Andrei Zynoviev, Laurent Jacob (Institut Curie / Mines ParisTech)
- Guillaume Obozinski (UC Berkeley / INRIA)