Classification of gene expression data with gene networks

Jean-Philippe Vert
Jean-Philippe.Vert@ensmp.fr

Centre for Computational Biology
Ecole des Mines de Paris, ParisTech

Journée “Stats et génome”, Université de Nice, April 10th, 2007

Outline

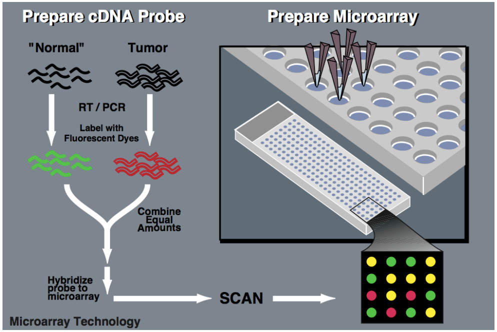
Outline

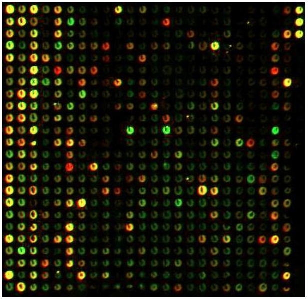
Outline

Outline

Outline

Tissue profiling with DNA chips

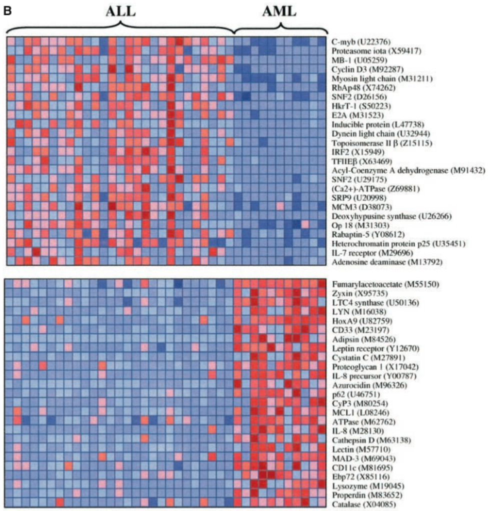




Data

- 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

Linear classifiers

The approach

- Each sample is represented by a vector $x = (x_1, \ldots, x_p)$ where $p > 10^5$ is the number of probes
- Classification: given the set of labeled sample, learn a linear decision function:

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

  that is positive for one class, negative for the other
- Interpretation: the weight β_i quantifies the influence of gene i for the classification

Linear classifiers estimation

Empirical risk minimization

Estimate the weights β_i by minimizing an empirical error on the training set:

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

where $l(y, f(x))$ is a loss function.

Pitfalls

- Statistics does not apply (?): 100 samples in 10^5 dimensions!
- It is necessary to reduce the complexity of the problem with prior knowledge.

Linear classifiers estimation

Empirical risk minimization

Estimate the weights β_i by minimizing an empirical error on the training set:

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

where $l(y, f(x))$ is a loss function.

Pitfalls

- Statistics does not apply (?): 100 samples in 10^5 dimensions!
- It is necessary to reduce the complexity of the problem with prior knowledge.

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 β

$$\| \beta \|_2^2 = \sum_{i=1}^{p} \beta_i^2 ,$$

(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 (< 100) whose expression are enough for classification. Interpretation is then about the selected genes. Examples:

- Greedy feature selection (T-tests, ...)
- Contrain the norm of β: LASSO penalty ($\| \beta \|_1 = \sum_{i=1}^{p} | \beta_i |$), elastic net penalty ($\| \beta \|_1 + \| \beta \|_2$), ...)

Pros

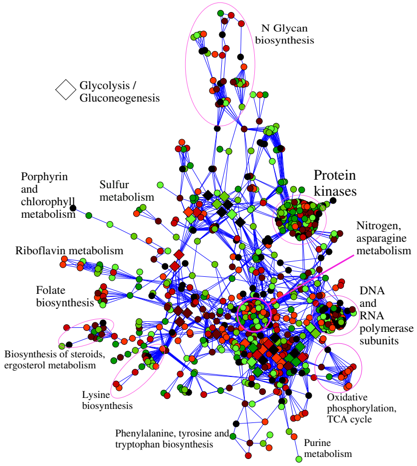
- Good performance in classification
- Biomarker selection
- Interpretability

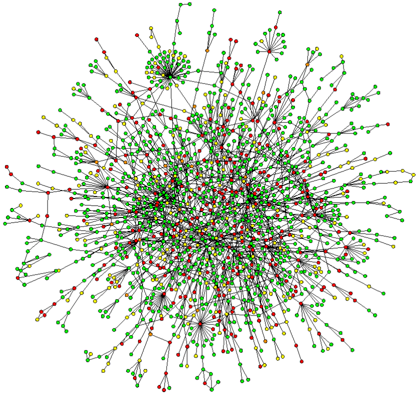
Cons

- The gene selection process is usually not robust
- No use of prior biological knowledge

Outline

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

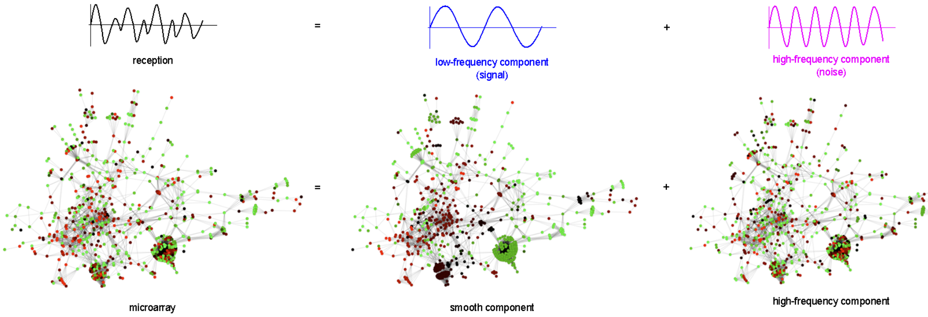


Gene network interpretation

Motivation

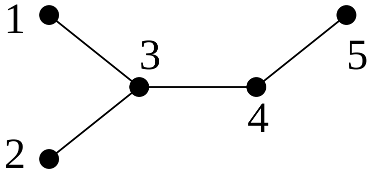
- Basic biological functions usually involve the coordinated action of several proteins:
  - Formation of protein complexes
  - Activation of metabolic, signalling or regulatory pathways
- Many pathways and protein-protein interactions are already known
- Hypothesis: the weights of the classifier should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

The 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

Notations

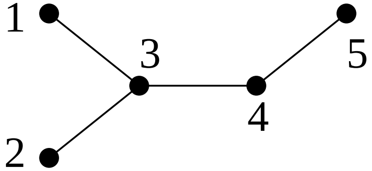


$$A = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}, \qquad D = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Graph Laplacian

Definition

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



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

Properties of the Laplacian

Lemma

Let $L = D - A$ be the Laplacian of the graph:

- *For any $f : \mathcal{X} \to \mathbb{R}$,*

$$f^\top L f = \sum_{i \sim j} \left(f(\mathbf{x}_i) - f(\mathbf{x}_j) \right)^2$$

- *L is a symmetric positive semi-definite matrix*
- *0 is an eigenvalue with multiplicity equal to the number of connected components.*

Proof: link between $\Omega(f)$ and L

$$\begin{aligned}
\sum_{i\sim j}\left(f\left(\mathbf{x}_i\right)-f\left(\mathbf{x}_j\right)\right)^2 &= \sum_{i\sim j}\left(f\left(\mathbf{x}_i\right)^2+f\left(\mathbf{x}_j\right)^2-2f\left(\mathbf{x}_i\right)f\left(\mathbf{x}_j\right)\right)\\
&= \sum_{i=1}^{m} D_{i,i} f\left(\mathbf{x}_i\right)^2 - 2\sum_{i\sim j} f\left(\mathbf{x}_i\right)f\left(\mathbf{x}_j\right)\\
&= f^\top D f - f^\top A f\\
&= f^\top L f
\end{aligned}$$

Proof: eigenstructure of L

- L is symmetric because A and D are symmetric.
- For any $f \in \mathbb{R}^m$, $f^\top L f \geq 0$, therefore the (real-valued) eigenvalues of L are ≥ 0 : L is therefore positive semi-definite.
- f is an eigenvector associated to eigenvalue 0
  iff $f^\top L f = 0$
  iff $\sum_{i \sim j} \left(f(\mathbf{x}_i) - f\left(\mathbf{x}_j\right)\right)^2 = 0$,
  iff $f(\mathbf{x}_i) = f\left(\mathbf{x}_j\right)$ when $i \sim j$,
  iff f is constant (because the graph is connected).

Fourier basis

Definition

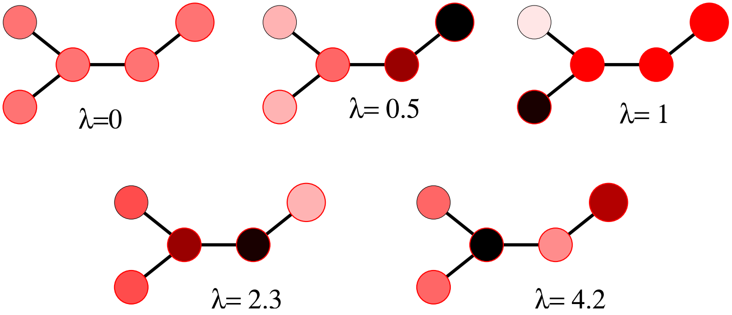
- The eigenvectors $e_1, \ldots, e_n$ of L with eigenvalues $0 = \lambda_1 \leq \ldots \leq \lambda_n$ form a basis called Fourier basis
- For any $f : V \to \mathbb{R}$, the Fourier transform of f is the vector $\hat{f} \in \mathbb{R}^n$ defined by:

$$\hat{f}_i = f^\top e_i \,, \quad i = 1, \ldots, n.$$

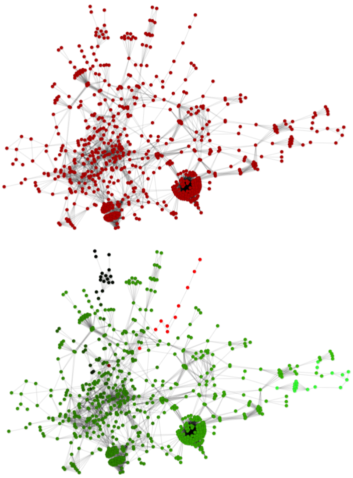
- Obviously the inverse Fourier formula holds:

$$f = \sum_{i=1}^{n} \hat{f}_i e_i \,.$$

Fourier basis



Fourier basis



Smoothing operator

Definition

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

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

Smoothing operators

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:

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

Smoothing operators

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:

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

Smoothing operators

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:

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

Supervised classification and regression

Working with smoothed profiles

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

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

- Applying these algorithms on the smooth profiles means solving:

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

Smooth solution

Lemma

This is equivalent to:

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

hence the linear classifier v is smooth.

Proof

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

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

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

Smooth solution

Lemma

This is equivalent to:

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

hence the linear classifier v is smooth.

Proof

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

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

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

Kernel methods

Smoothing kernel

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

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

with

$$
K_\phi = \sum_{i=1}^{n} \phi(\lambda_i)^2 e_i e_i^\top \,.
$$

Examples

- 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:

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

Examples

- 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:

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

Outline

Data

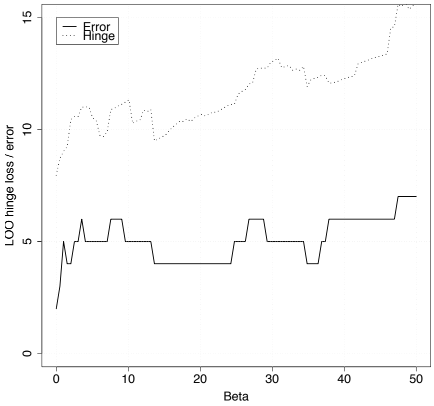
Expression

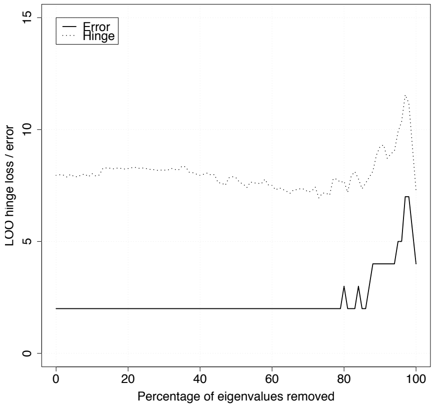
- Study the effect of low irradiation doses on the yeast
- 12 non irradiated vs 6 irradiated
- Which pathways are involved in the response at the transcriptomic level?

Graph

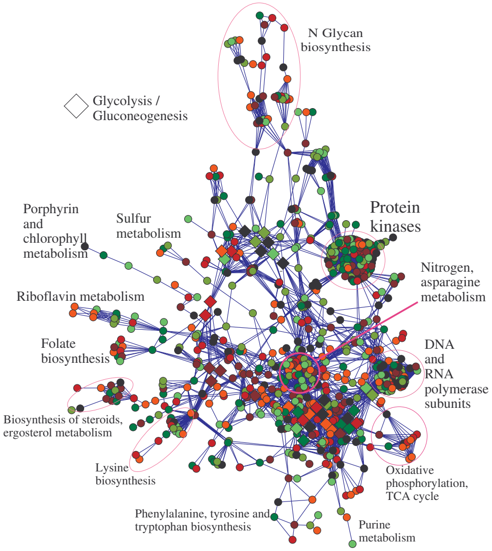
- KEGG database of metabolic pathways
- Two genes are connected is they code for enzymes that catalyze successive reactions in a pathway (metabolic gene network).
- 737 genes, 4694 vertices.

Classification performance

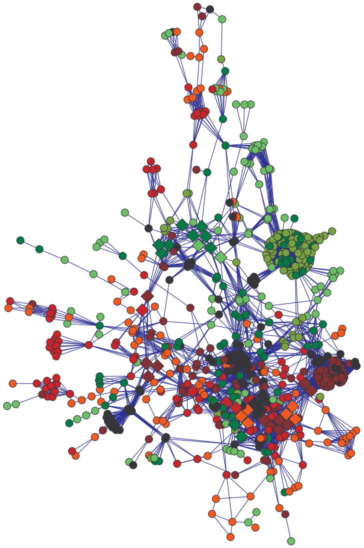




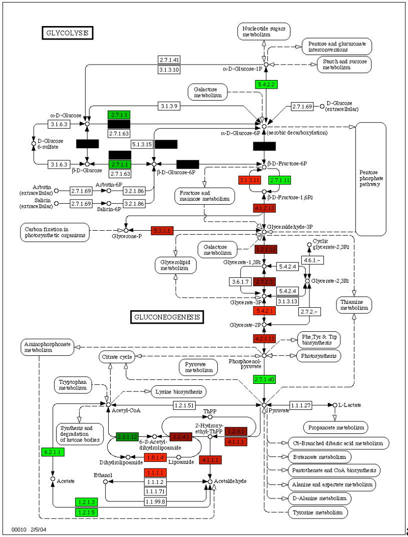
Classifier



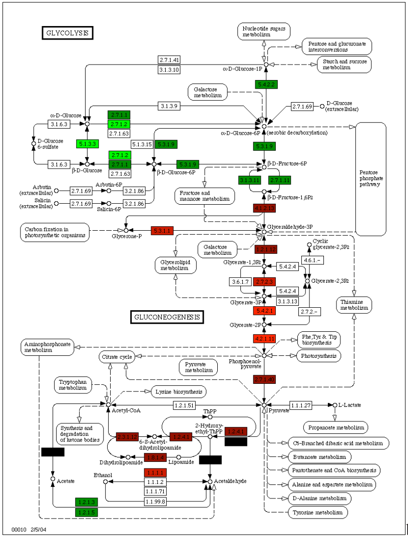
N Glycan biosynthesis
Glycolysis / Gluconeogenesis
Porphyrin and chlorophyll metabolism
Sulfur metabolism
Protein kinases
Nitrogen, asparagine metabolism
Riboflavin metabolism
DNA and RNA polymerase subunits
Folate biosynthesis
Biosynthesis of steroids, ergosterol metabolism
Lysine biosynthesis
Oxidative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



Classifier



a)



b)

Outline

Conclusion

- Use the gene graph to encode prior knowledge about the classifier.
- Prior knowledge is always needed to classify few examples in large dimensions (sometimes implicitly)
- Future work: validation of the method on more data, other formulations, directed graphs...

Acknowledgements

KernelChip project (ACI IMPBIO)

- Franck Rapaport (Ecole des Mines and Curie Institute)
- Emmanuel Barillot (Curie Institute)
- Andrei Zynoviev (Curie Institute)
- Marie Dutreix (Curie Institute)

Reference

F. Rapaport, A. Zynoviev, M. Dutreix, E. Barillot and J.-P. Vert, Classification of microarray data using gene networks, *BMC Bioinformatics* 8:35, 2007.