On feature selection and pattern detection in high dimension

Jean-Philippe Vert

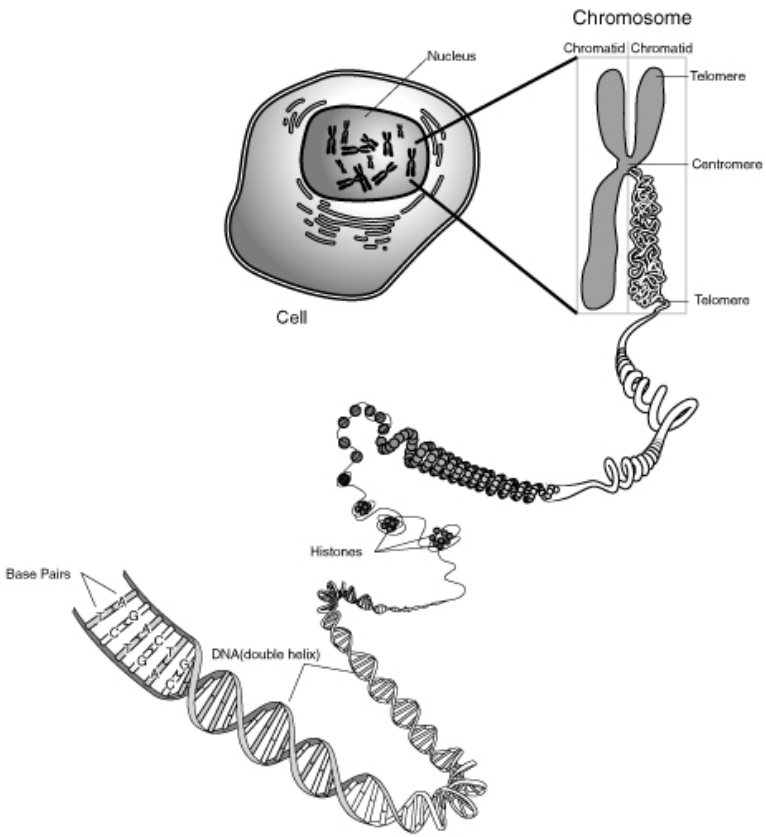
Mines ParisTech / Curie Institute / Inserm

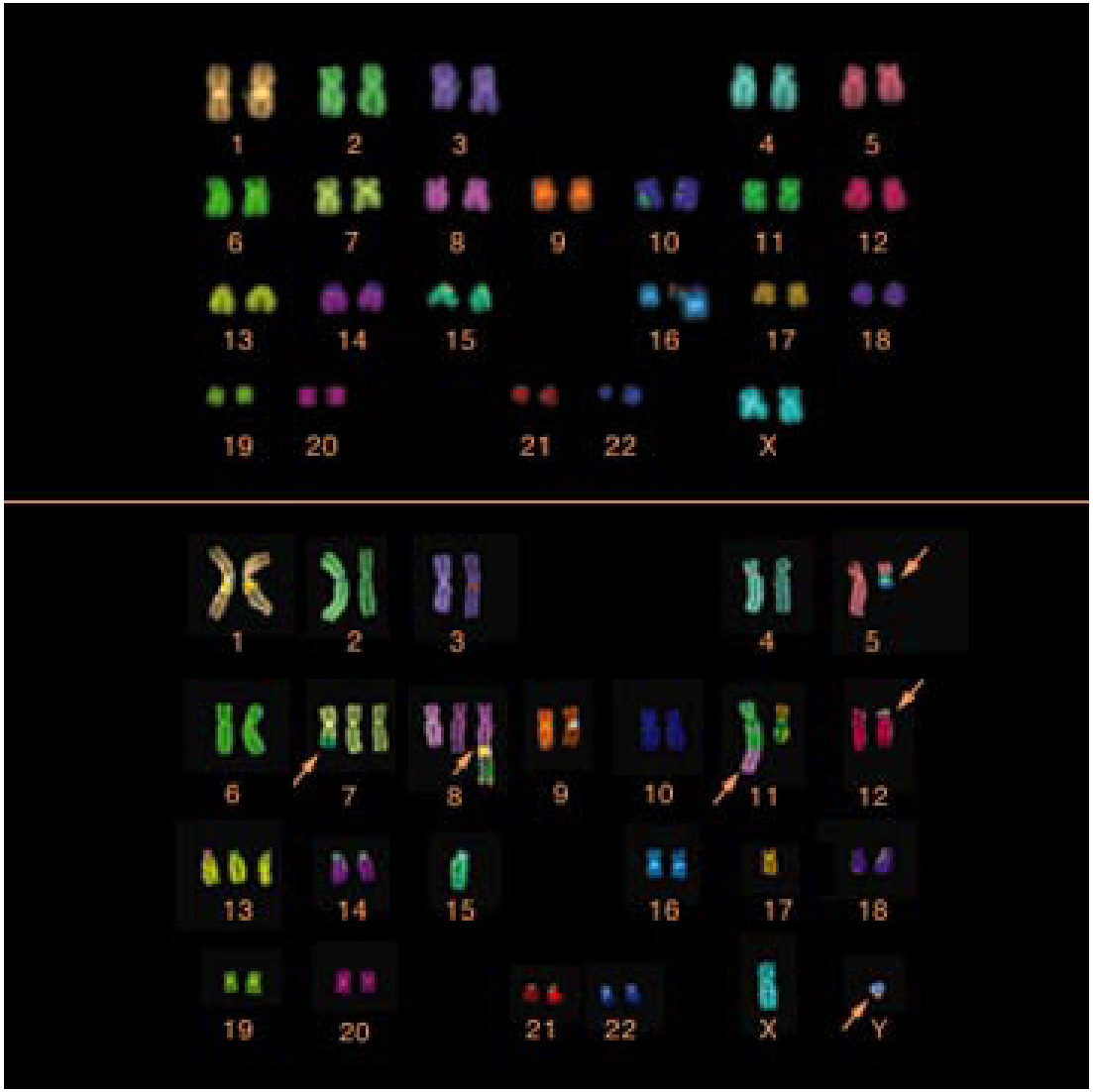
"Point de vue" seminar, Paris 7, Jan 17, 2010.

Outline

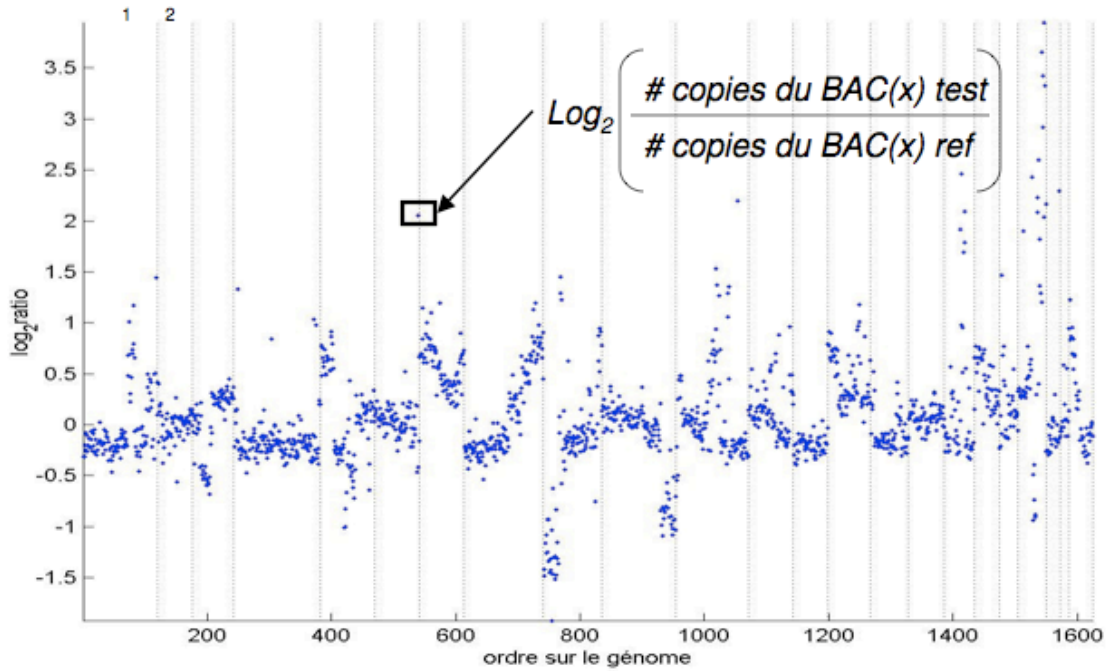
Outline

Chromosomic aberrations in cancer



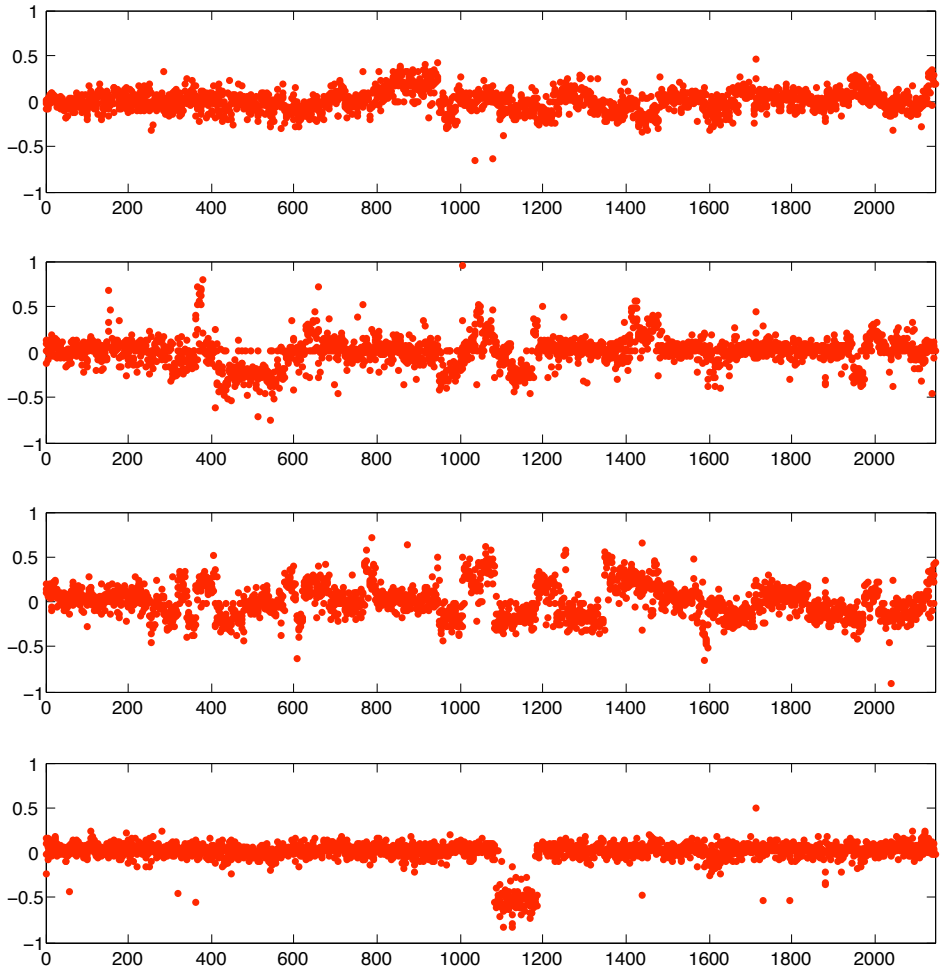


Comparative Genomic Hybridization (CGH)



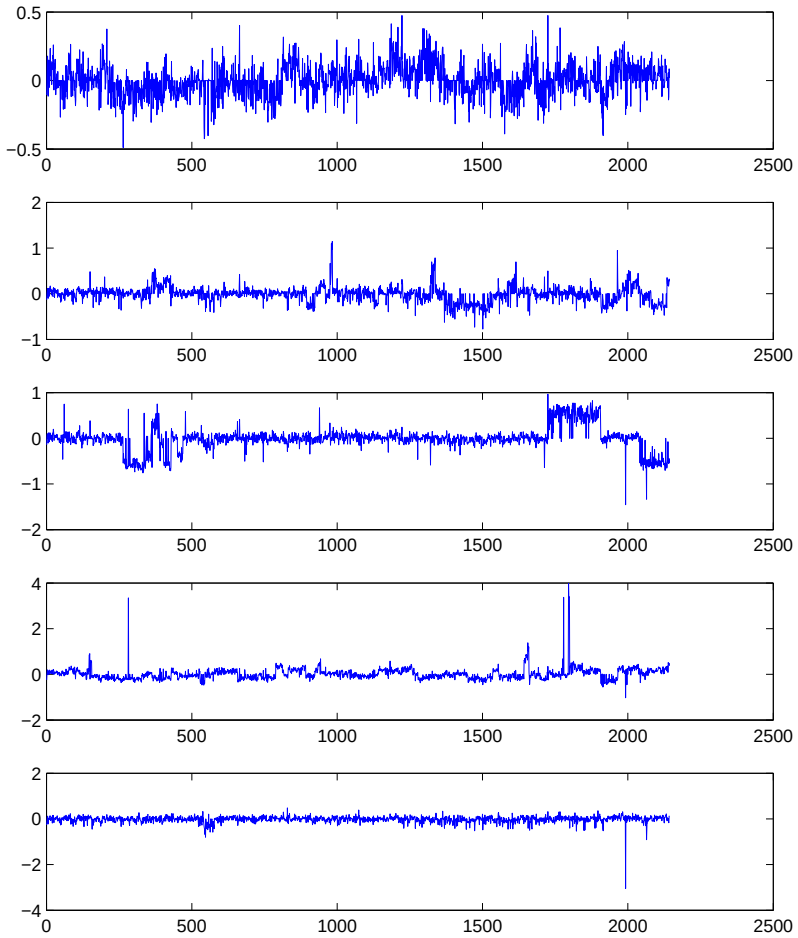
Jain et al. Genome research 2002 12:325-332

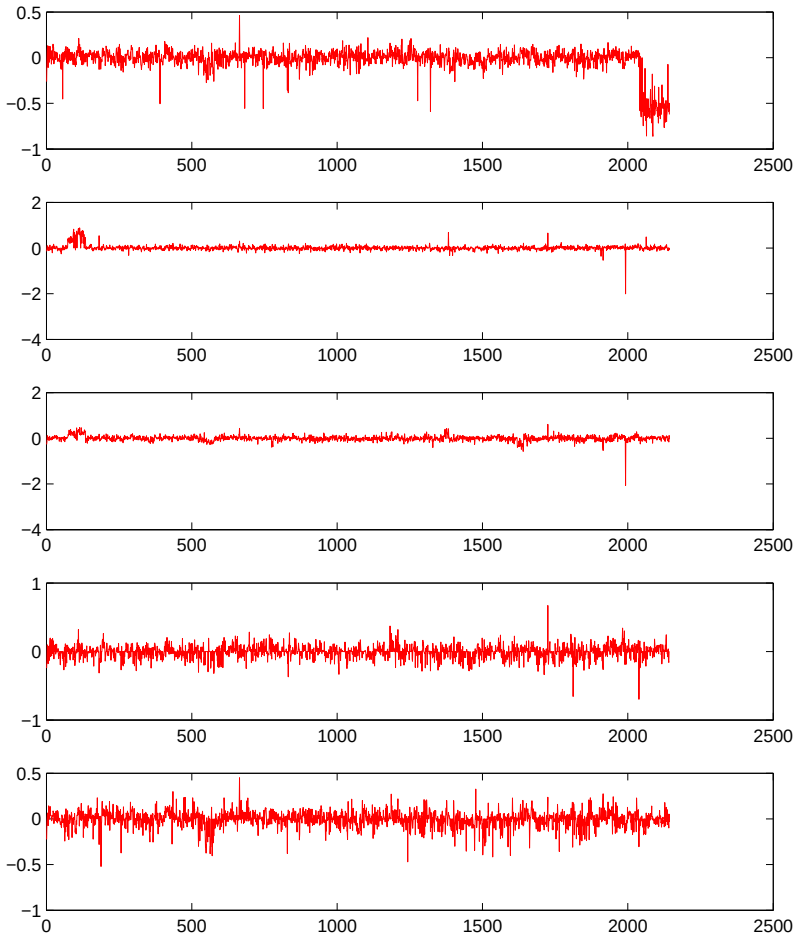
Can we detect frequent breakpoints?



A collection of bladder tumour copy number profiles.

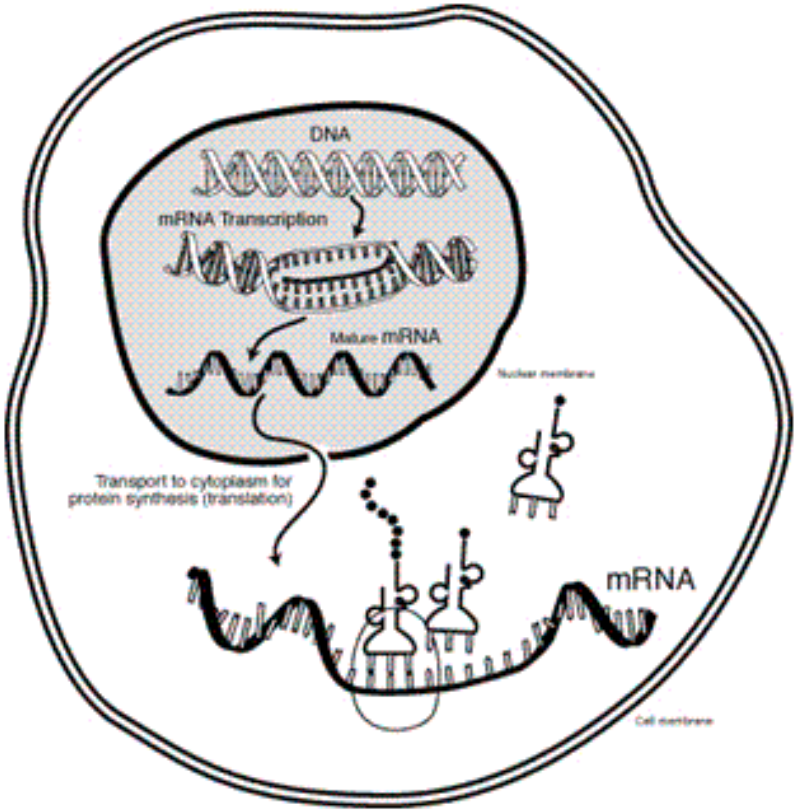
Can we detect discriminative patterns?

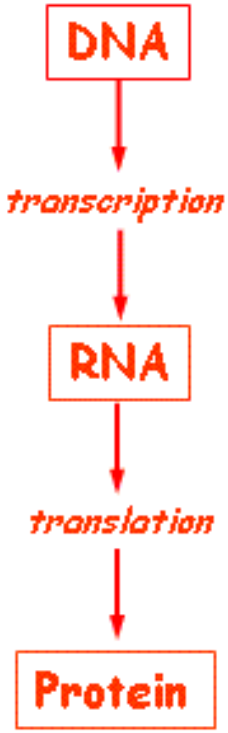




Aggressive (left) vs non-aggressive (right) melanoma.

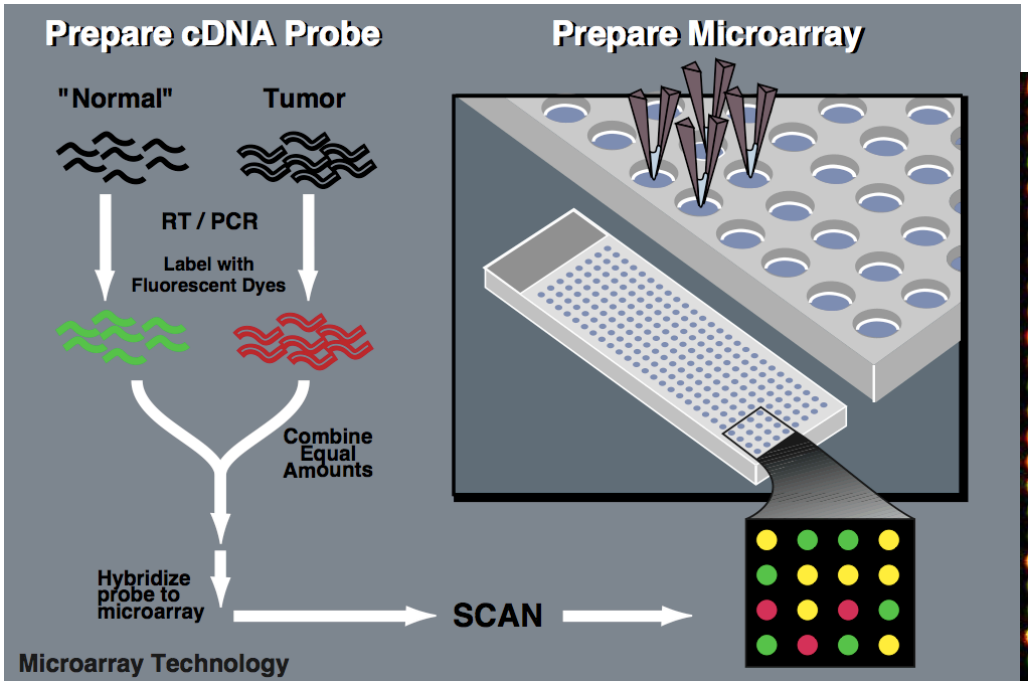
DNA → RNA → protein

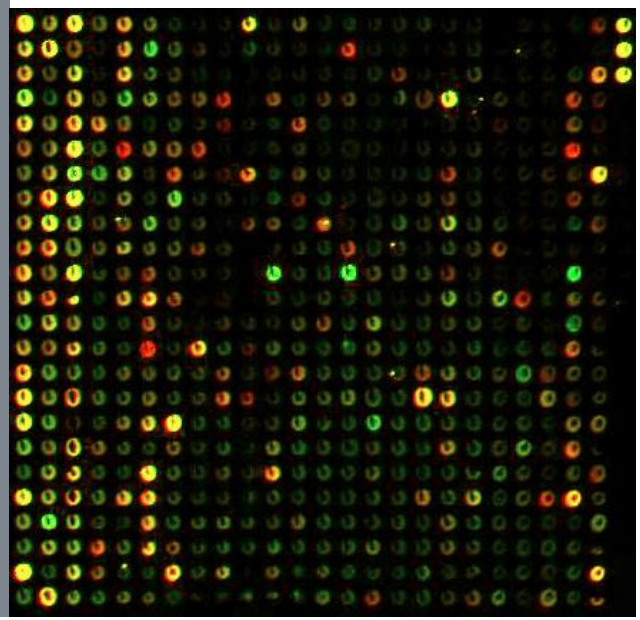




- CGH shows the (static) DNA
- Cancer cells have also abnormal (dynamic) gene expression (= transcription)

Tissue profiling with DNA chips

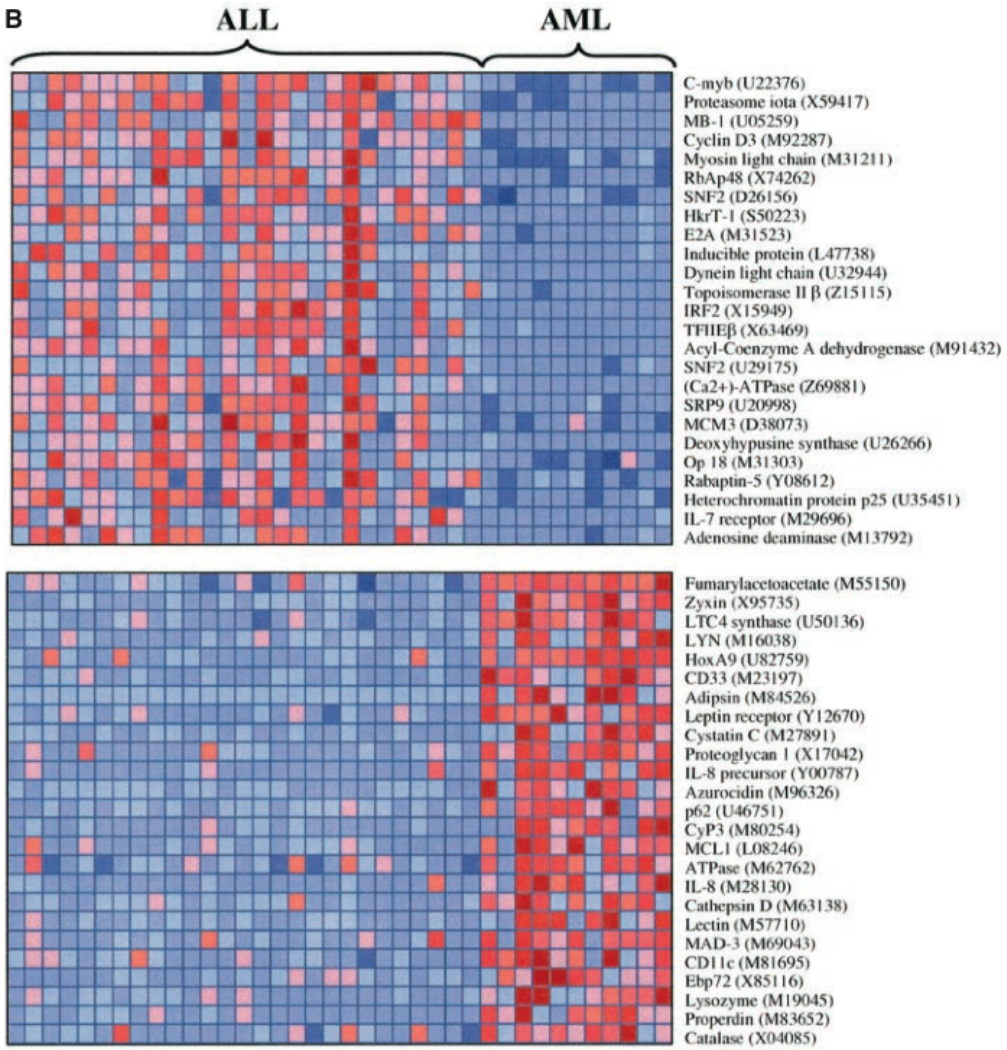




Data

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

Tissue classification from microarray data



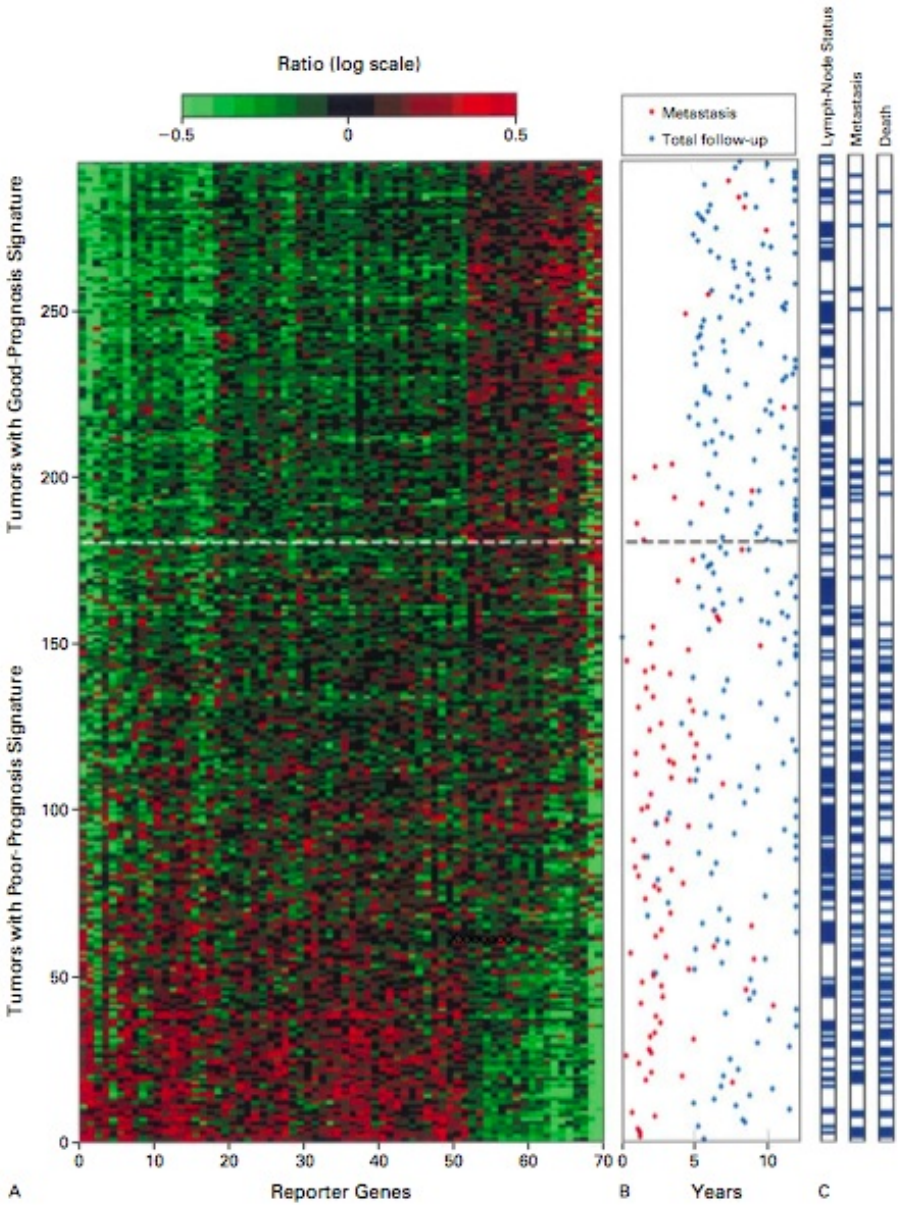
Goal

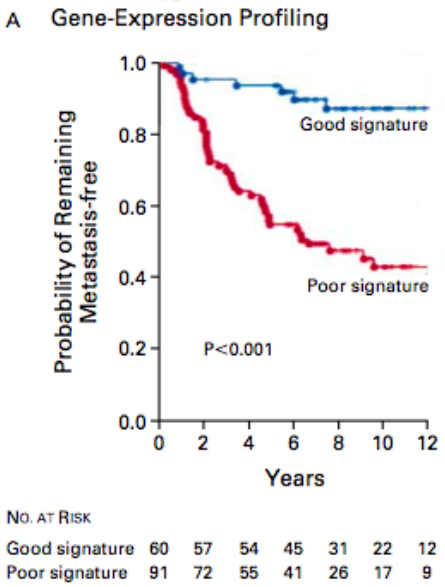
- Design a classifier to automatically assign a class to future samples from their expression profile
- Interpret biologically the differences between the classes

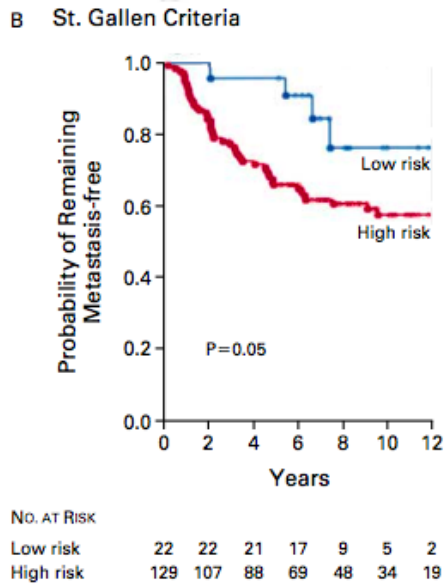
Difficulty

- Large dimension
- Few samples

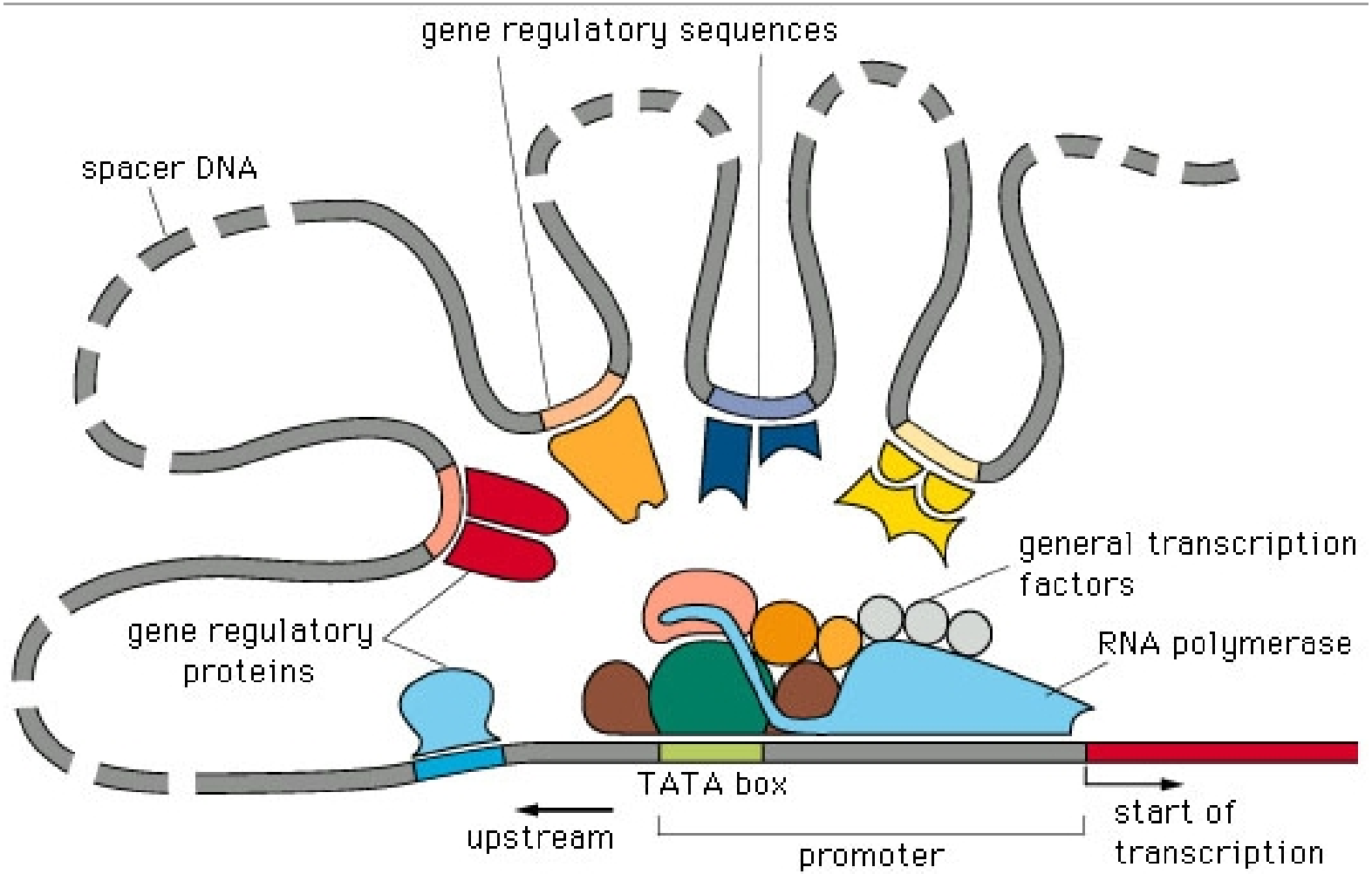
Can we detect predictive molecular signatures?



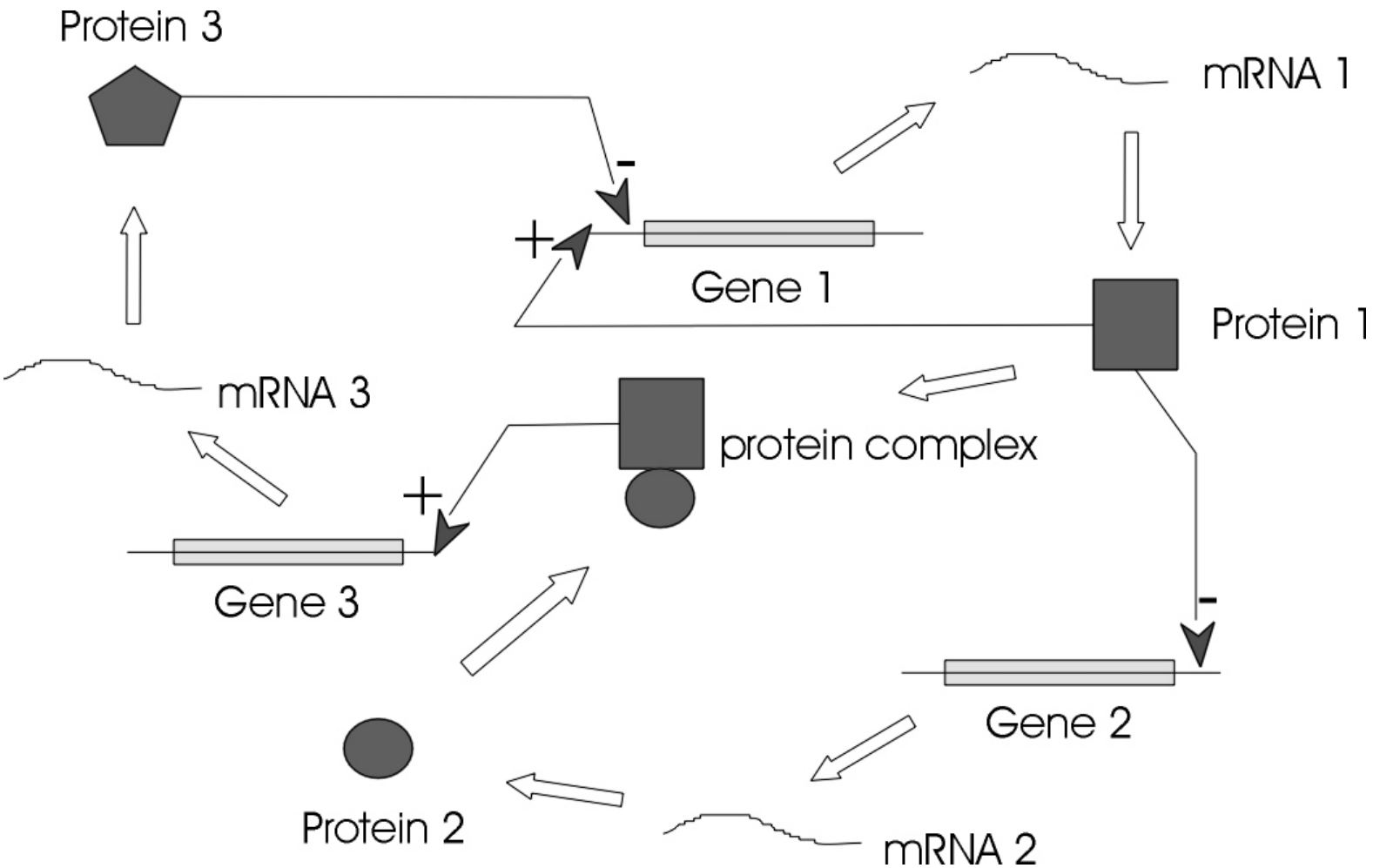




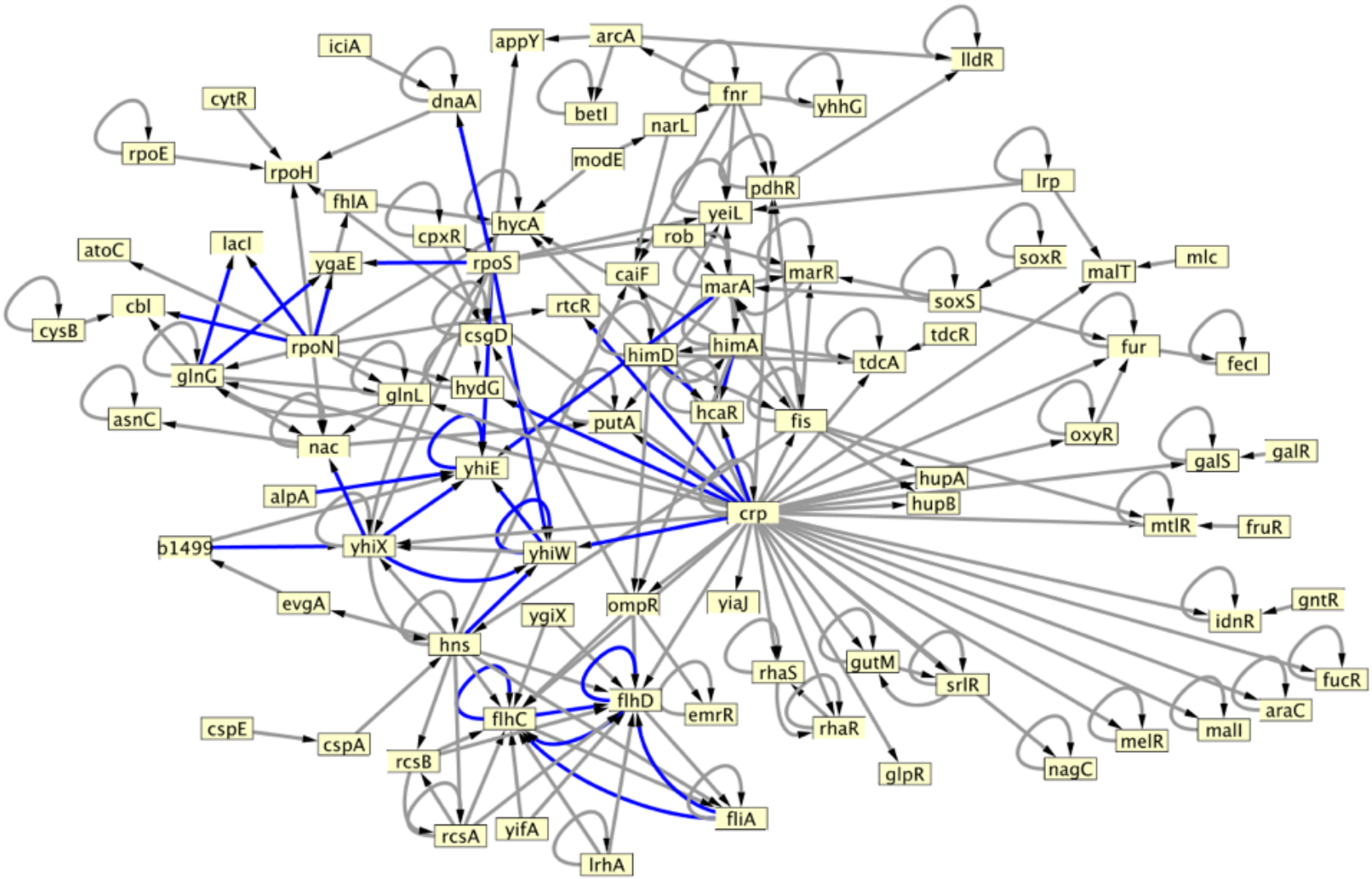
Gene expression regulation



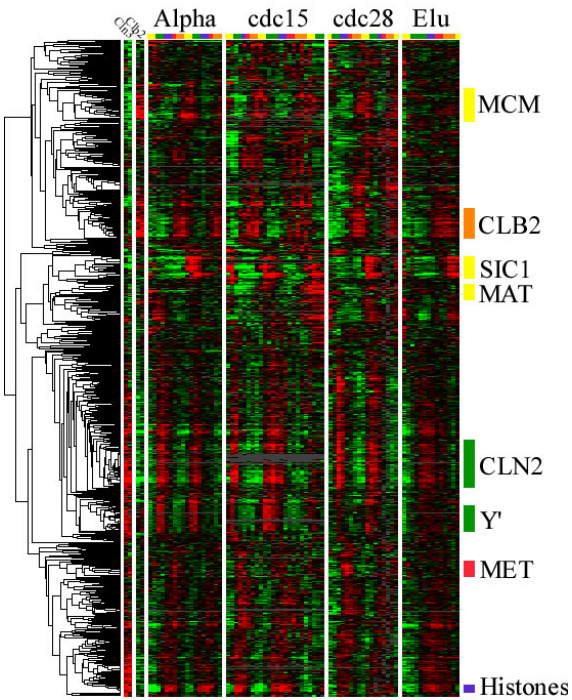
Gene regulatory network

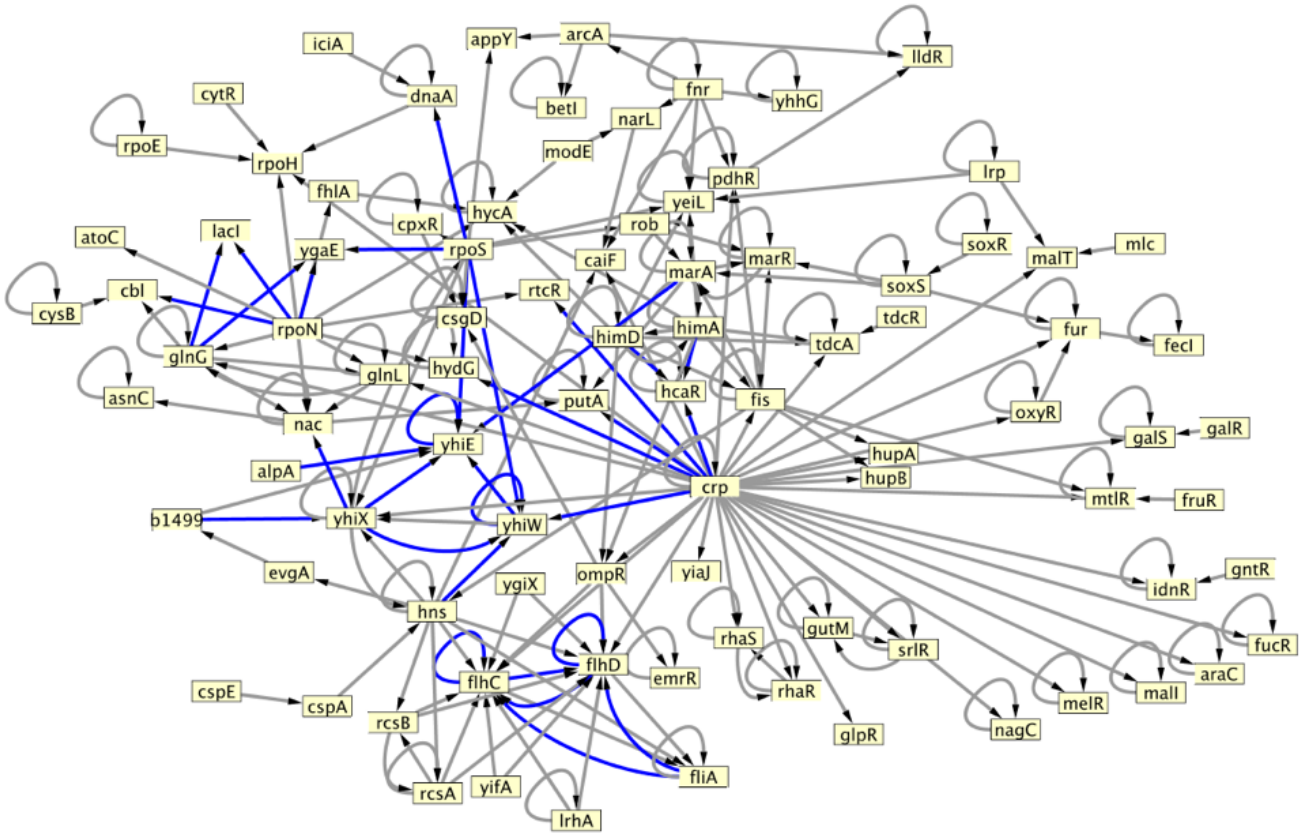


Gene regulatory network (GRN) of E. coli

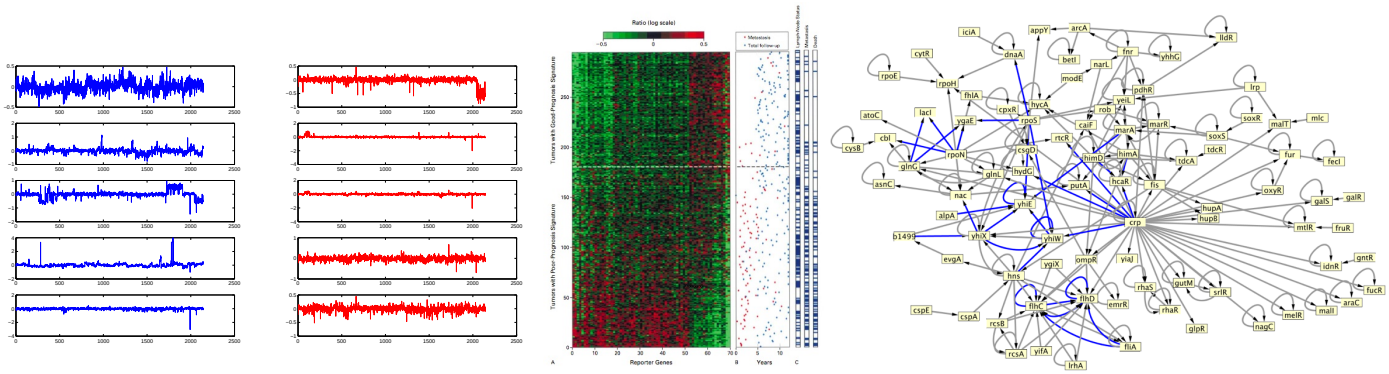


Can we reconstruct the GRN from expression data?





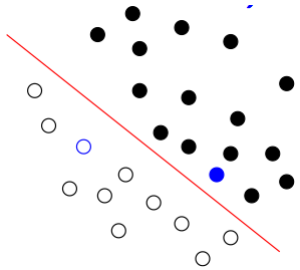
Summary

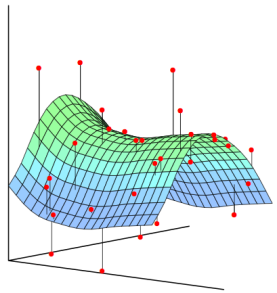


- Many problems...
- Classification accuracy is not all, interpretation is necessary
- Common topic: detect predictive variables / patterns
- Need for efficient and scalable algorithms

Outline

Classification and regression





Input

- $\mathcal{X}$ the space of patterns (typically, $\mathcal{X} = \mathbb{R}^p$)
- $\mathcal{Y}$ the space of response or labels
  - Classification or pattern recognition : $\mathcal{Y} = \{-1, 1\}$
  - Regression : $\mathcal{Y} = \mathbb{R}$
- $\mathcal{S} = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ a training set in $(\mathcal{X} \times \mathcal{Y})^n$

Output

- A function $f : \mathcal{X} \to \mathcal{Y}$ to predict the output associated to any new pattern $x \in \mathcal{X}$ by $f(x)$

Feature selection

Estimate a function $f(x)$ that only depends on a subset $S \subset [1, p]$ of the variables.

Why?

- Statistics: a way to control the complexity of the search space, can improve accuracy by reducing the estimation error. Especially relevant in high dimension, and if we believe that there exist good sparse models.
- Interpretation: the selected variables in S are interesting to understand the physical/biological structure of the problem, and suggest further investigations
- Practical: a small set S can lead to cheap implementations of the predictor, e.g., dedicated chips for prognosis.

Feature selection

Estimate a function $f(x)$ that only depends on a subset $S \subset [1, p]$ of the variables.

Why?

- Statistics: a way to control the complexity of the search space, can improve accuracy by reducing the estimation error. Especially relevant in high dimension, and if we believe that there exist good sparse models.
- Interpretation: the selected variables in S are interesting to understand the physical/biological structure of the problem, and suggest further investigations
- Practical: a small set S can lead to cheap implementations of the predictor, e.g., dedicated chips for prognosis.

Best subset selection

- In best subset selection, we must solve the problem:

$$\min R(f_\beta) \quad \text{s.t.} \quad \| \beta \|_0 \leq k$$

  for $k = 1, \ldots, p$, and R is an empirical risk.
- The state-of-the-art is branch-and-bound optimization, known as *leaps and bound* for least squares (Furnival and Wilson, 1974).
- This is usually a NP-hard problem, feasible for p as large as 30 or 40

Efficient feature selection

To work with more variables, we must use different methods. The state-of-the-art is split among

1. Filter methods : the predictors are preprocessed and ranked from the most relevant to the less relevant. The subsets are then obtained from this list, starting from the top.
2. Wrapper method: here the feature selection is iterative, and uses a learning algorithm in the inner loop
3. Embedded methods : here the feature selection is part of the learning algorithm itself

Additionnally, ensemble feature learning has been proposed as a useful meta-method for feature selection.

Filter methods

- Associate a score $S(i)$ to each feature i, then rank the features by decreasing score.
- Many scores / criteria can be used
  - Loss of the ERM trained on a single feature
  - Statistical tests (Fisher, T-test)
  - Other performance criteria of the ERM restricted to a single feature (AUC, ...)
  - Information theoretical criteria (mutual information...)

Pros

Simple, scalable, good empirical success

Cons

- Selection of redundant features
- Some variables useless alone can become useful together

Filter methods

- Associate a score $S(i)$ to each feature i, then rank the features by decreasing score.
- Many scores / criteria can be used
  - Loss of the ERM trained on a single feature
  - Statistical tests (Fisher, T-test)
  - Other performance criteria of the ERM restricted to a single feature (AUC, ...)
  - Information theoretical criteria (mutual information...)

Pros

Simple, scalable, good empirical success

Cons

- Selection of redundant features
- Some variables useless alone can become useful together

Filter methods

- Associate a score $S(i)$ to each feature i, then rank the features by decreasing score.
- Many scores / criteria can be used
  - Loss of the ERM trained on a single feature
  - Statistical tests (Fisher, T-test)
  - Other performance criteria of the ERM restricted to a single feature (AUC, ...)
  - Information theoretical criteria (mutual information...)

Pros

Simple, scalable, good empirical success

Cons

- Selection of redundant features
- Some variables useless alone can become useful together

Wrapper methods

The idea

- A greedy approach to

$$\min R^n(f_\beta) \quad \text{s.t.} \quad \| \beta \|_0 \leq k$$

- For a given set of seleted features, we know how to minimize $R^n(f)$
- We iteratively try to find a good set of features, by adding/removing features which contribute most to decrease the risk (using ERM as an internal loop)

Two flavors of wrapper methods

Forward stepwise selection

- Start from no features
- Sequentially add into the model the feature that most improves the fit

Backward stepwise selection (if n>p)

- Start from all features
- Sequentially removes from the model the feature that least degrades the fit

Other variants

Hybrid stepwise selection strategies that consider both forward and backward moves at each stage, and make the "best" move

Two flavors of wrapper methods

Forward stepwise selection

- Start from no features
- Sequentially add into the model the feature that most improves the fit

Backward stepwise selection (if n>p)

- Start from all features
- Sequentially removes from the model the feature that least degrades the fit

Other variants

Hybrid stepwise selection strategies that consider both forward and backward moves at each stage, and make the "best" move

Two flavors of wrapper methods

Forward stepwise selection

- Start from no features
- Sequentially add into the model the feature that most improves the fit

Backward stepwise selection (if n>p)

- Start from all features
- Sequentially removes from the model the feature that least degrades the fit

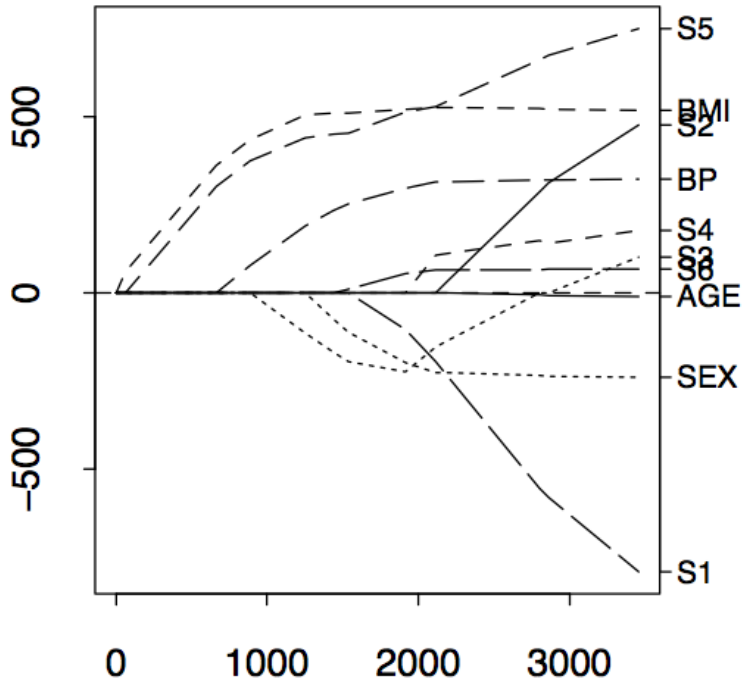
Other variants

Hybrid stepwise selection strategies that consider both forward and backward moves at each stage, and make the "best" move

Embedded methods

- Decision trees
- Sparsity-inducing convex penalties, e.g.

$$\min R(f_\beta) \quad \text{s.t.} \quad \| \beta \|_1 \leq k$$



Ensemble feature selection

1. For $t = 1, \ldots, T$, randomly subsample samples and/or variables
2. For each t, select a subset of variables S_t
3. Aggregate all S_t to obtain the final list of variables

Examples:

- Random forests
- Stability selection

Ensemble feature selection

1. For $t = 1, \ldots, T$, randomly subsample samples and/or variables
2. For each t, select a subset of variables S_t
3. Aggregate all S_t to obtain the final list of variables

Examples:

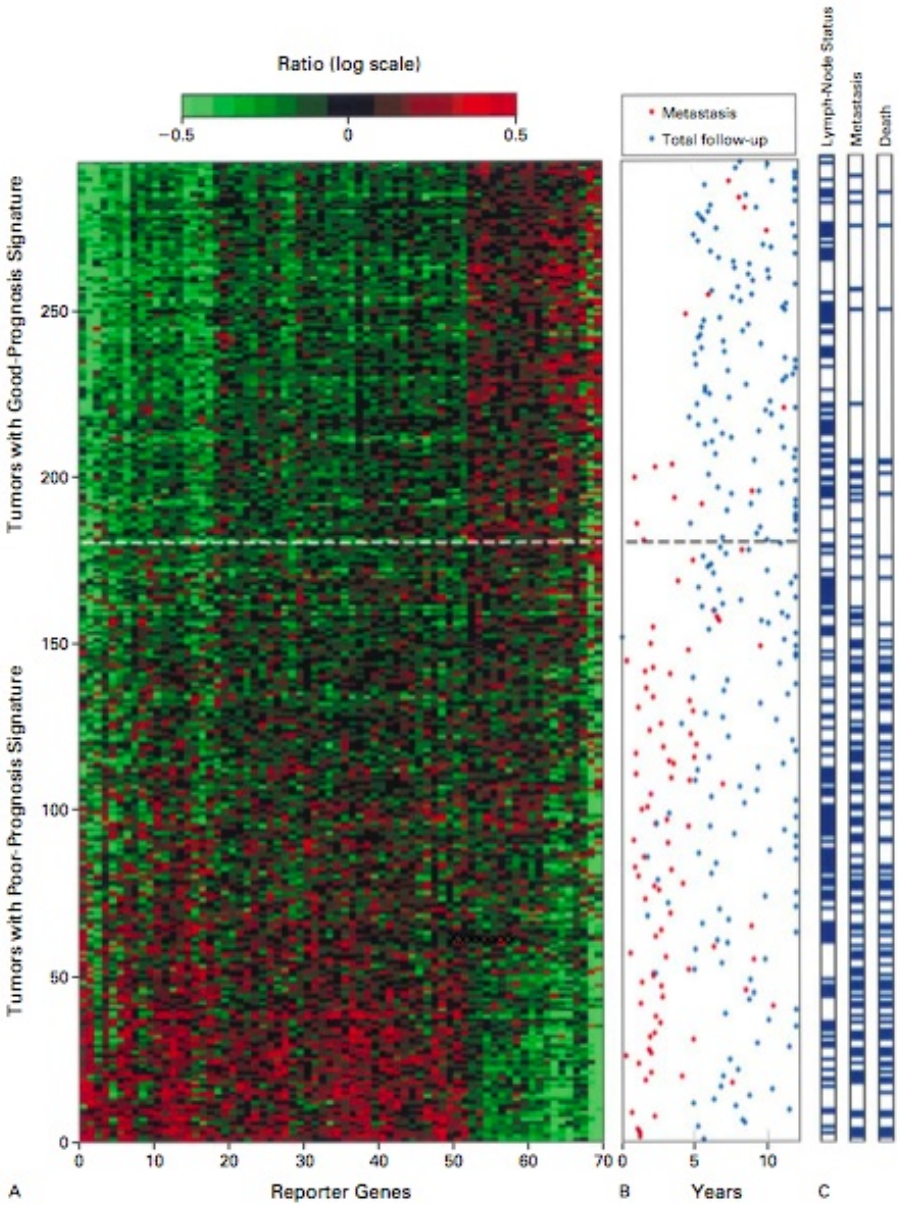
- Random forests
- Stability selection

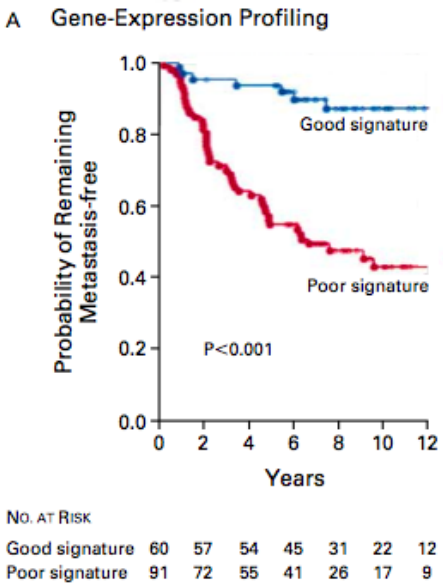
Summary

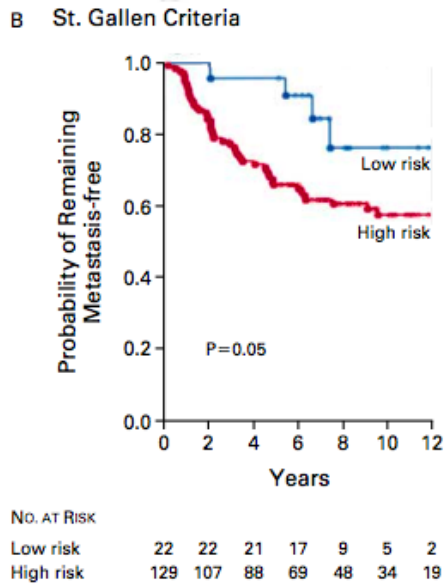
- A well-studied problem, with many solutions that vary in computational complexity and theoretical guarantees
- Feature selection in the "small n large p" setting has been studied a lot recently, mostly for embedded methods (lasso...) and largely motivated by applications in biology
- Ensemble feature selection has been put forward recently (stability selection, bolasso...), but limited theoretical results and validations
- The theoretical validity of different methods on real data is often hard to check

Outline

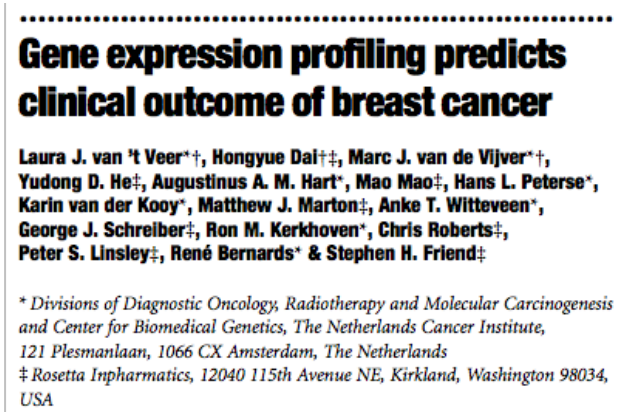
Prognostic molecular signatures







Prognostic molecular signatures



Gene expression profiling predicts clinical outcome of breast cancer

Laura J. van 't Veer*†, Hongyue Dai†‡, Marc J. van de Vijver*†, Yudong D. He‡, Augustinus A. M. Hart*, Mao Mao‡, Hans L. Peterse*, Karin van der Kooy*, Matthew J. Marton‡, Anke T. Witteveen*, George J. Schreiber‡, Ron M. Kerkhoven*, Chris Roberts‡, Peter S. Linsley‡, René Bernards* & Stephen H. Friend‡

* *Divisions of Diagnostic Oncology, Radiotherapy and Molecular Carcinogenesis and Center for Biomedical Genetics, The Netherlands Cancer Institute, 121 Plesmanlaan, 1066 CX Amsterdam, The Netherlands*
‡ *Rosetta Inpharmatics, 12040 115th Avenue NE, Kirkland, Washington 98034, USA*

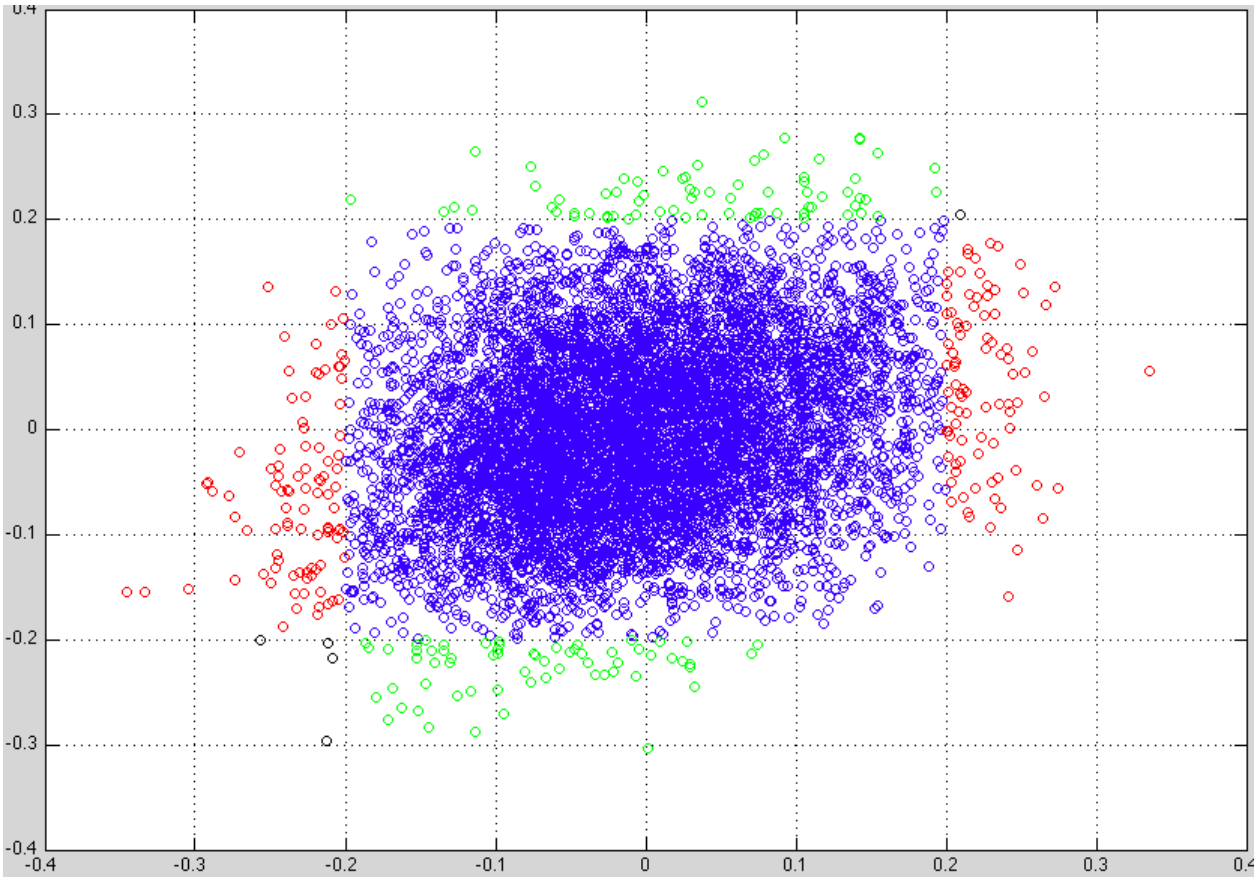
Gene-expression profiles to predict distant metastasis of lymph-node-negative primary breast cancer

Yixin Wang, Jan G M Klijn, Yi Zhang, Anieta M Sieuwerts, Maxime P Look, Fei Yang, Dmitri Talantov, Mieke Timmermans, Marion E Meijer-van Gelder, Jack Yu, Tim Jatkoe, Els M J J Berns, David Atkins, John A Foekens

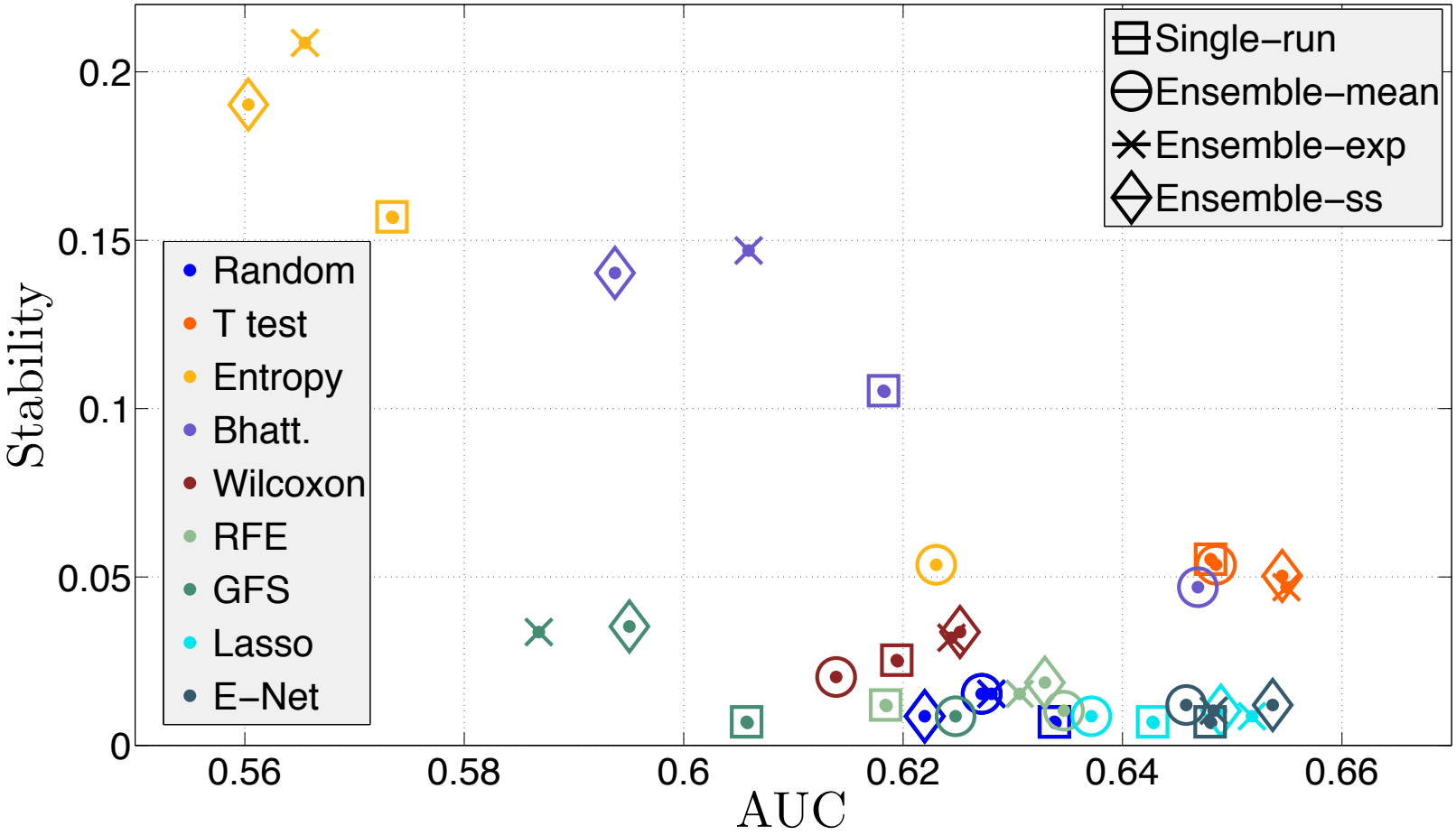
- Two signatures in clinical trial for breast cancer (70 and 76 genes)
- Only 3 genes in common... Why?
  - Different cohorts of patients?
  - Different technologies and experimental protocols?
  - Different algorithm for feature selection?
  - Other?

Unstability of molecular signatures

- Wang dataset: $n = 286$, $p = 8141$
- Pearson correlation with the output on 2 random subsamples of 143 samples:



Comparison of feature selection methods



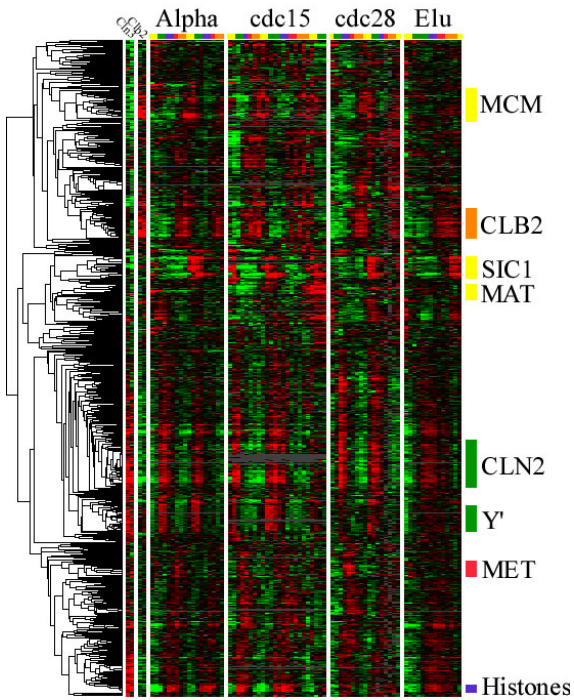
Haury et al. (2011)

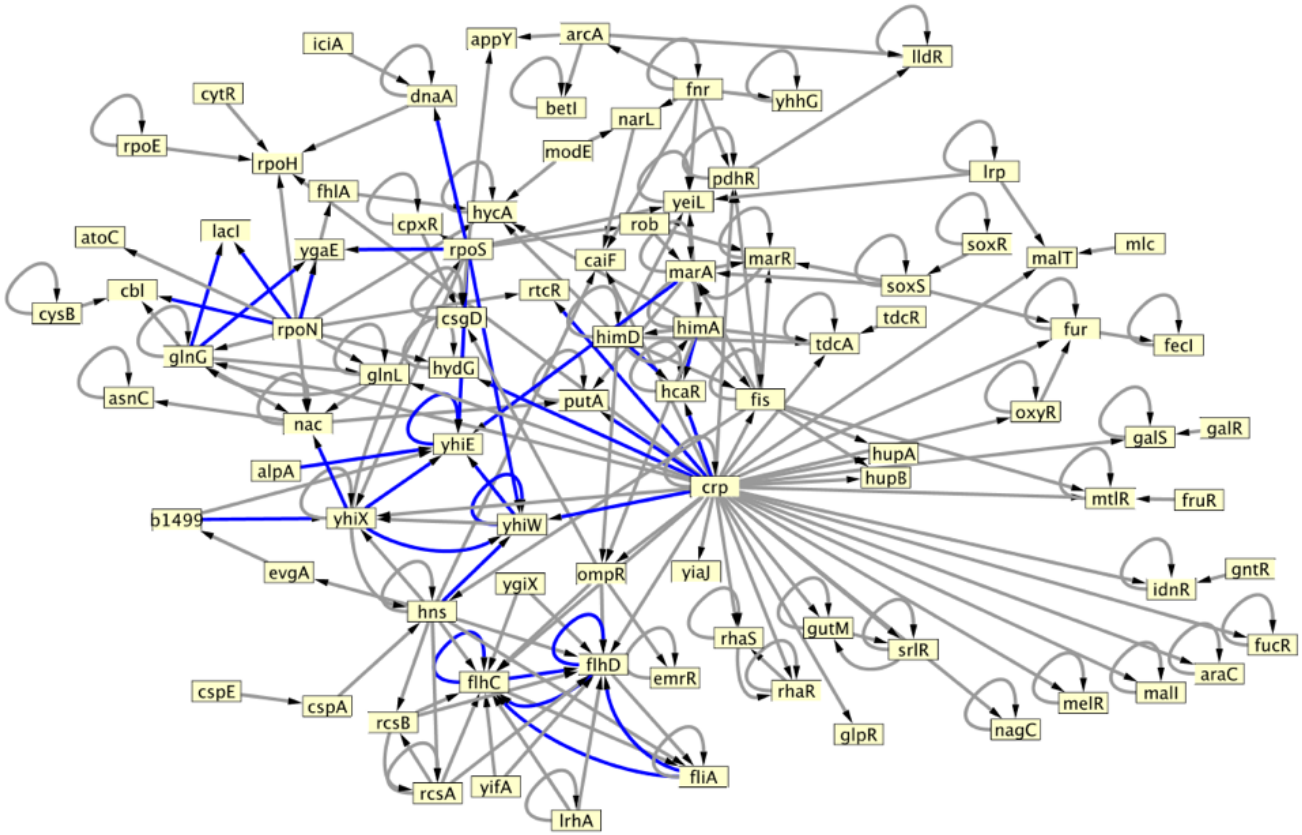
Lessons learned...

- Difficult problem!
- Unstability mostly due to statistical issues
- Filter methods (t-test) current method of choice
- Ensemble feature selection not really useful

Outline

The problem

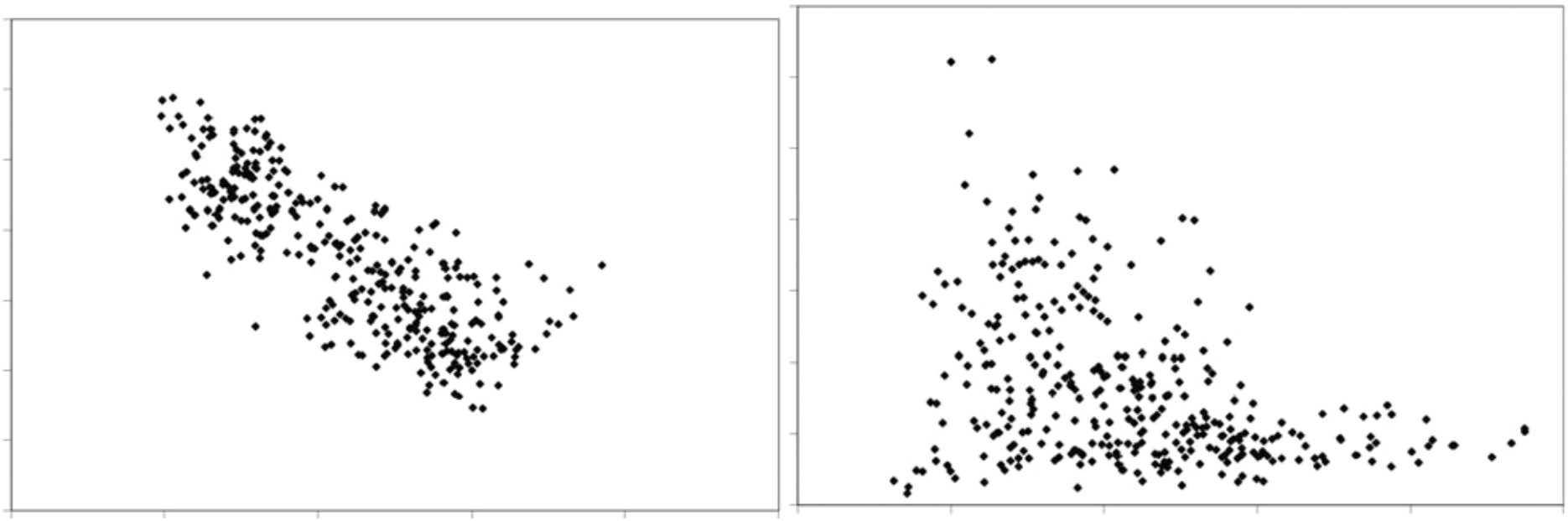




Predict the GRN from a matrix $X \in \mathbb{R}^{n \times p}$ of expression data.

Predict regulations between "dependent" genes

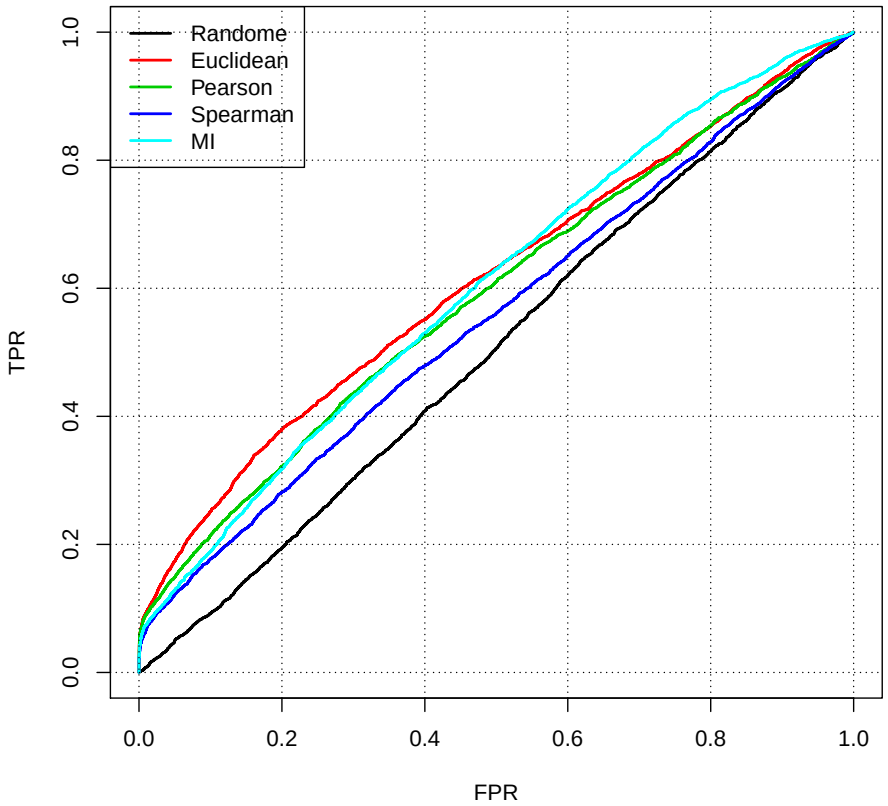
If *A* regulates *B*, we expect their expressions to be dependent across experiments

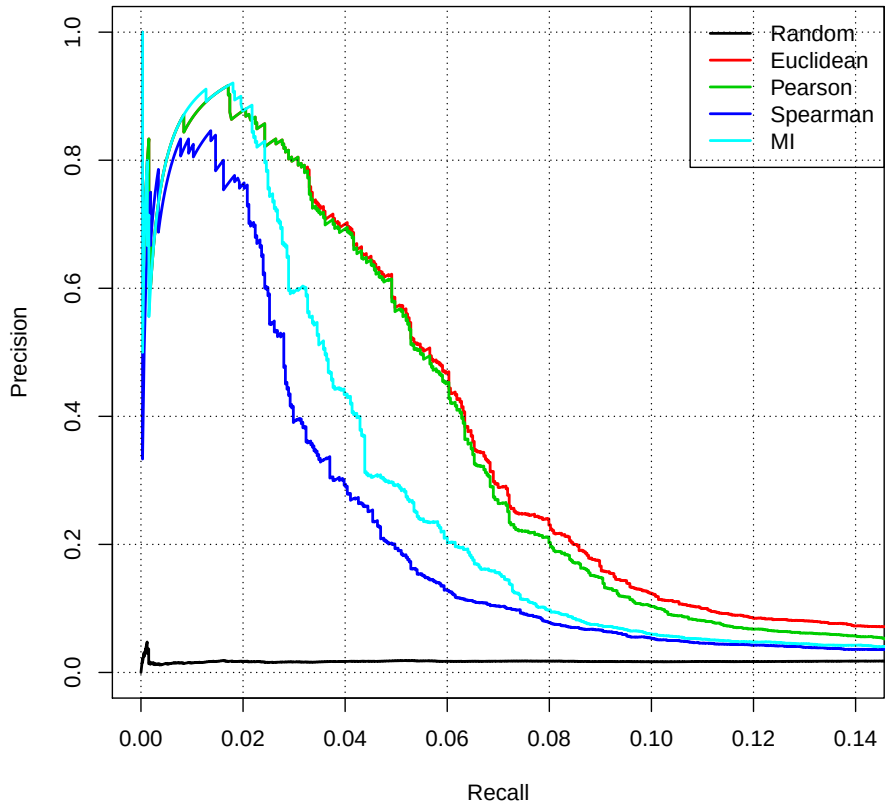


Detect the dependency by various measures, e.g., Euclidean distance, correlation, mutual information...

Validation

Application: E coli regulatory network : 154 TF targeting 1164 genes through 3293 regulations





GRN inference by feature selection

- The dynamic equation of the mRNA concentration of a gene is of the form:

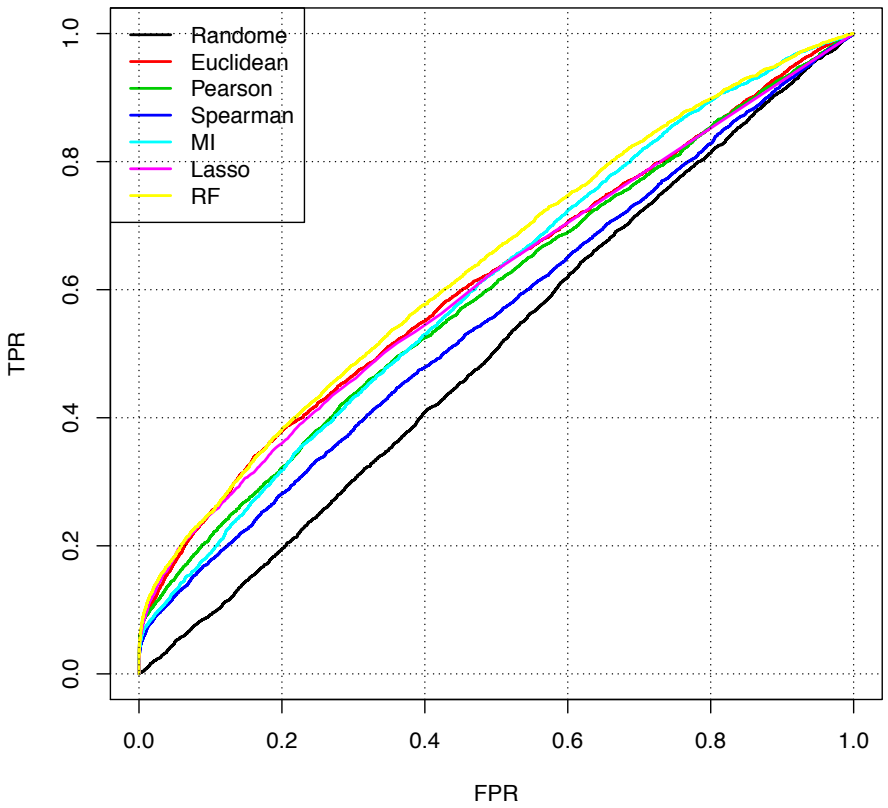
$$\frac{dX}{dt} = f(X, R)$$

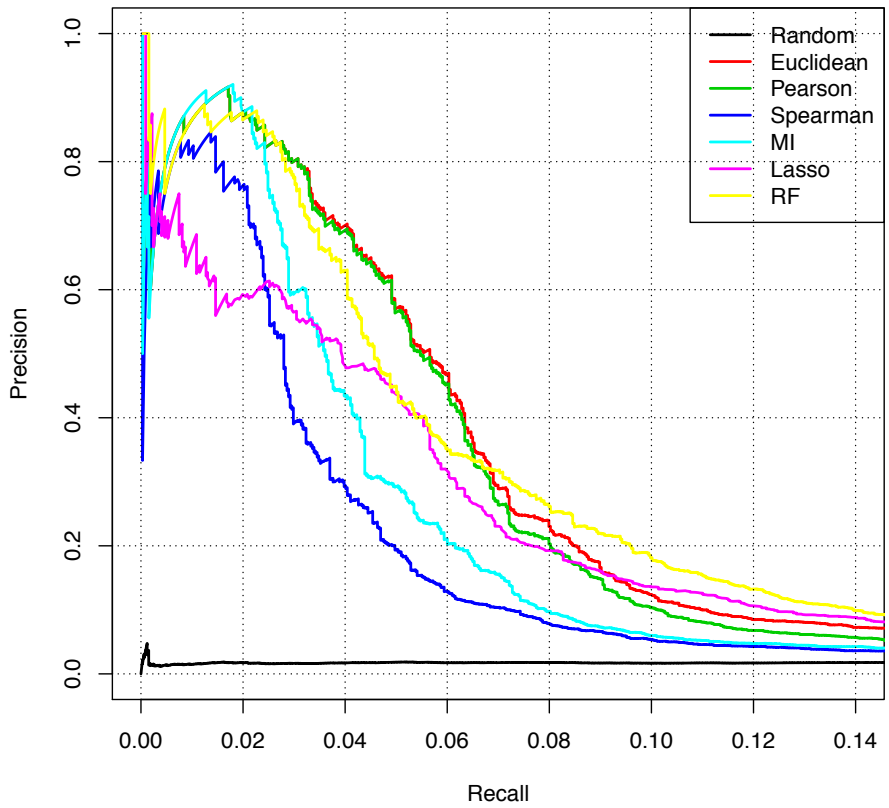
  where R represent the set of concentrations of transcription factors that regulate X.
- At steady state, $dX/dt = 0 = f(X, R)$
- If we linearize $f(X, R) = 0$ we get linear relation of the form

$$X = \sum_{i \in R} \beta_i X_i$$

- This suggests to look for transcription factors whose expression is sufficient to explain the expression of X across different experiments.

Validation





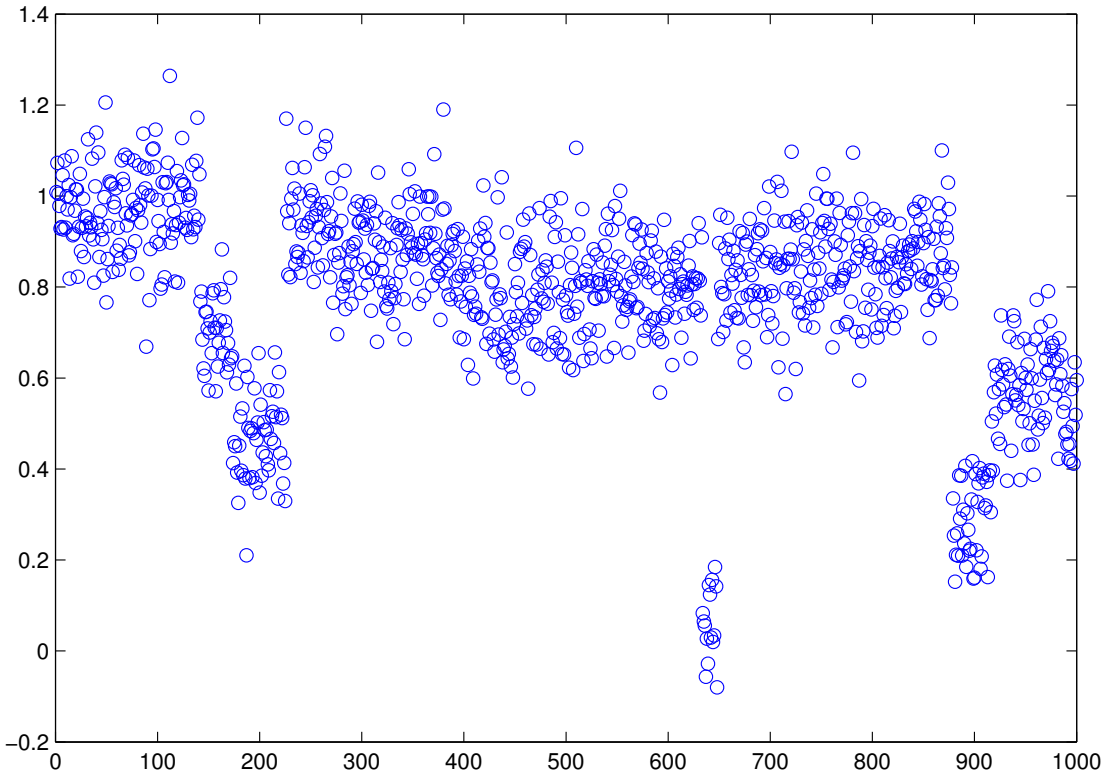
RF (Huynh-Thu et al., 2010) and Lasso+stability selection (Haury et al., 2011) ranked 1st and 2nd at the 2010 DREAM5 in silico network inference challence

Lessons learned...

- Again, very difficult problem! (recall around 10% in the best case...)
- State-of-the-art express GRN network as feature selection (perhaps not the best idea?)
- Ensemble feature selection seems to work best

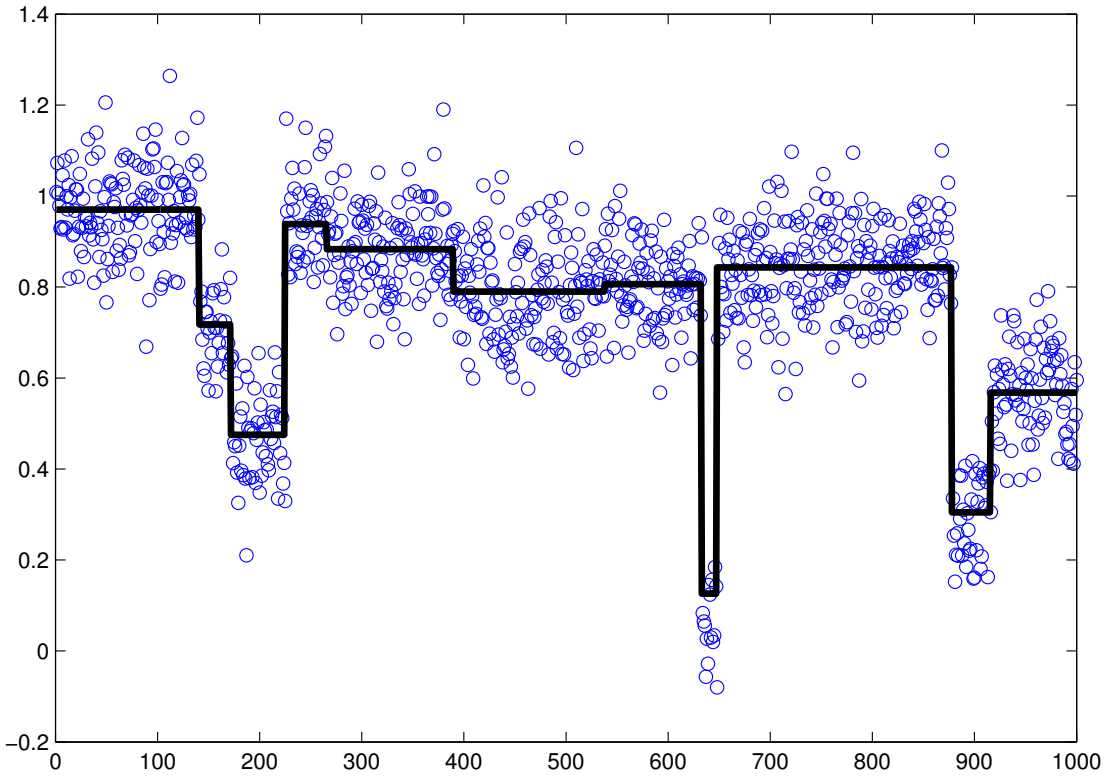
Outline

The problem



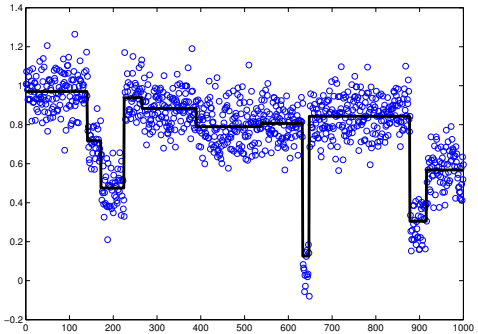
- Let $Y \in \mathbb{R}^p$ the signal
- We want to find a piecewise constant approximation $\hat{U} \in \mathbb{R}^p$ with at most k change-points.

The problem



- Let $Y \in \mathbb{R}^p$ the signal
- We want to find a piecewise constant approximation $\hat{U} \in \mathbb{R}^p$ with at most k change-points.

An optimal solution?

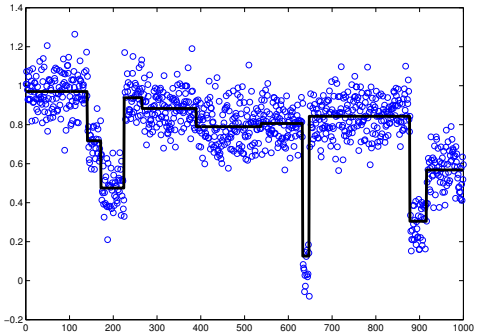


- We can define an "optimal" piecewise constant approximation $\hat{U} \in \mathbb{R}^p$ as the solution of

$$\min_{U \in \mathbb{R}^p} \| Y - U \|^2 \quad \text{such that} \quad \sum_{i=1}^{p-1} \mathbf{1}\left(U_{i+1} \neq U_i\right) \leq k$$

- This is an optimization problem over the $\binom{p}{k}$ partitions...
- Dynamic programming finds the solution in $O(p^2 k)$ in time and $O(p^2)$ in memory
- But: does not scale to $p = 10^6 \sim 10^9$...

An optimal solution?

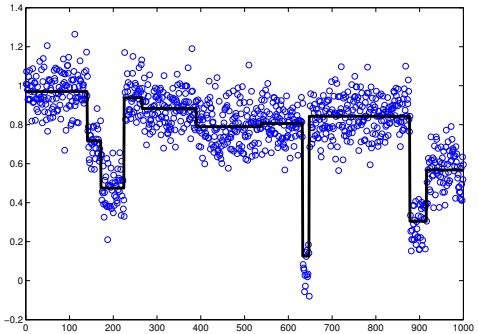


- We can define an "optimal" piecewise constant approximation $\hat{U} \in \mathbb{R}^p$ as the solution of

$$\min_{U \in \mathbb{R}^p} \| Y - U \|^2 \quad \text{such that} \quad \sum_{i=1}^{p-1} \mathbf{1}(U_{i+1} \neq U_i) \leq k$$

- This is an optimization problem over the $\binom{p}{k}$ partitions...
- Dynamic programming finds the solution in $O(p^2 k)$ in time and $O(p^2)$ in memory
- But: does not scale to $p = 10^6 \sim 10^9$...

An optimal solution?

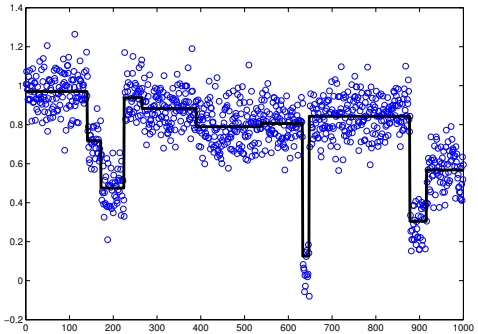


- We can define an "optimal" piecewise constant approximation $\hat{U} \in \mathbb{R}^p$ as the solution of

$$\min_{U \in \mathbb{R}^p} \| Y - U \|^2 \quad \text{such that} \quad \sum_{i=1}^{p-1} \mathbf{1}(U_{i+1} \neq U_i) \leq k$$

- This is an optimization problem over the $\binom{p}{k}$ partitions...
- Dynamic programming finds the solution in $O(p^2 k)$ in time and $O(p^2)$ in memory
- But: does not scale to $p = 10^6 \sim 10^9$...

An optimal solution?



- We can define an "optimal" piecewise constant approximation $\hat{U} \in \mathbb{R}^p$ as the solution of

$$\min_{U \in \mathbb{R}^p} \| Y - U \|^2 \quad \text{such that} \quad \sum_{i=1}^{p-1} \mathbf{1}(U_{i+1} \neq U_i) \leq k$$

- This is an optimization problem over the $\binom{p}{k}$ partitions...
- Dynamic programming finds the solution in $O(p^2 k)$ in time and $O(p^2)$ in memory
- But: does not scale to $p = 10^6 \sim 10^9$...

Promoting sparsity with the ℓ_1 penalty

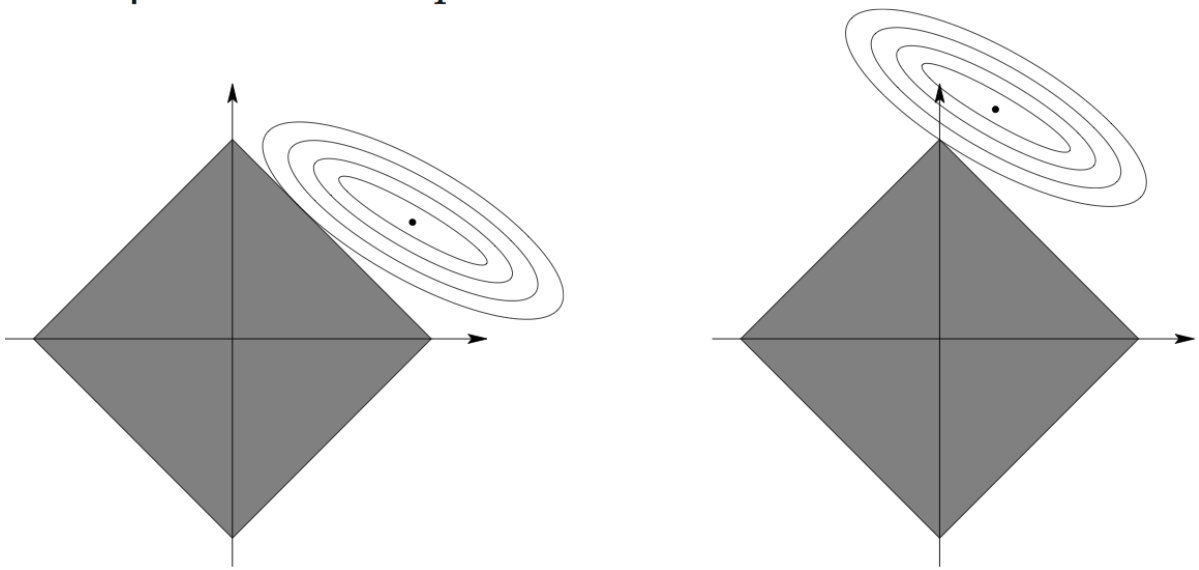
The ℓ_1 penalty (Tibshirani, 1996; Chen et al., 1998)

If $R(\beta)$ is convex and "smooth", the solution of

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

is usually sparse.

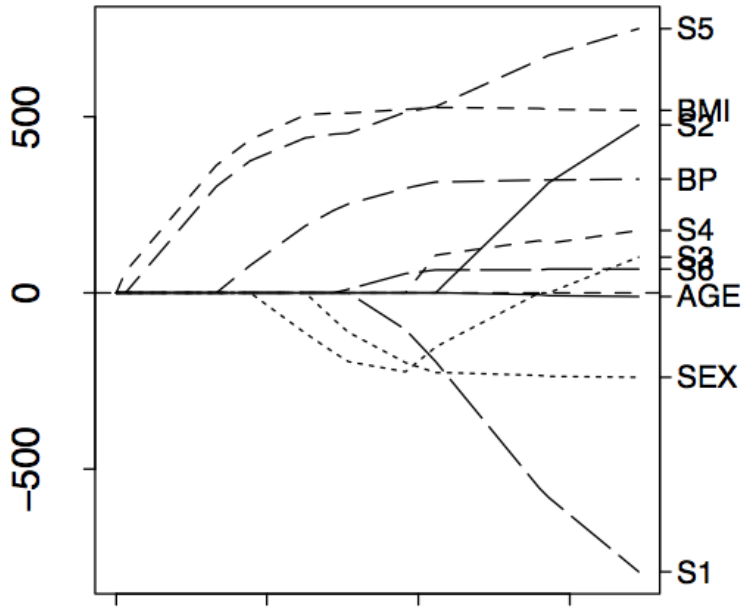
Geometric interpretation with $p = 2$



Efficienty computation of the regularization path

$$\min_{\beta \in \mathbb{R}^p} \| Y - X\beta \|^2 + \lambda \sum_{i=1}^{p} | \beta_i | \qquad (1)$$

- No explicit solution, but this is just a quadratic program.
- LARS (Efron et al., 2004) provides a fast algorithm to compute the solution for all λ's simultaneously (regularization path)



Promoting piecewise constant profiles penalty

The total variation / variable fusion penalty

If $R(\beta)$ is convex and "smooth", 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 (Rudin et al., 1992; Land and Friedman, 1996).

Proof:

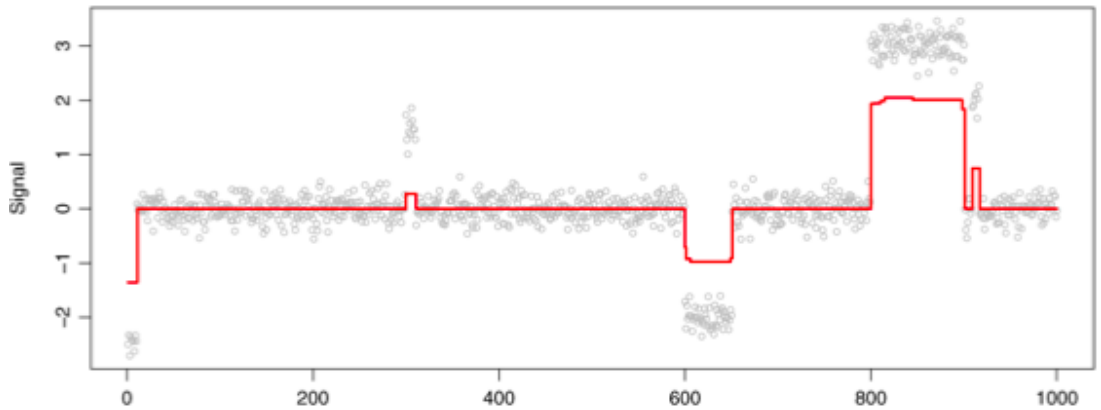
- Change of variable $u_i = \beta_{i+1} - \beta_i$, $u_0 = \beta_1$
- We obtain a Lasso problem in $u \in \mathbb{R}^{p-1}$
- u sparse means β piecewise constant

TV signal approximator

$$\min_{\beta \in \mathbb{R}^p} \| Y - \beta \|^2 \quad \text{such that} \quad \sum_{i=1}^{p-1} | \beta_{i+1} - \beta_i | \leq \mu$$

Adding additional constraints does not change the change-points:

- $\sum_{i=1}^{p} | \beta_i | \leq \nu$ (Tibshirani et al., 2005; Tibshirani and Wang, 2008)
- $\sum_{i=1}^{p} \beta_i^2 \leq \nu$ (Mairal et al. 2010)



Solving TV signal approximator

$$\min_{\beta \in \mathbb{R}^p} \| Y - \beta \|^2 \quad \text{such that} \quad \sum_{i=1}^{p-1} | \beta_{i+1} - \beta_i | \leq \mu$$

- QP with sparse linear constraints in $O(p^2)$ -> 135 min for $p = 10^5$ (Tibshirani and Wang, 2008)
- Coordinate descent-like method $O(p)$? -> 3s s for $p = 10^5$ (Friedman et al., 2007)
- For all μ with the LARS in $O(pK)$ (Harchaoui and Levy-Leduc, 2008)
- For all μ in $O(p \ln p)$ (Hoefling, 2009)
- For the first K change-points in $O(p \ln K)$ (Bleakley and V., 2010)

Greedy dichotomic segmentation

Require: k number of intervals, $\gamma(I)$ gain function to split an interval I into $I_L(I), I_R(I)$

I_0 represents the interval $[1, p]$
$\mathcal{P} = \{I_0\}$
for $i = 1$ to k **do**
  $I^* \leftarrow \arg\max_{I \in \mathcal{P}} \gamma(I^*)$
  $\mathcal{P} \leftarrow \mathcal{P} \setminus \{I^*\}$
  $\mathcal{P} \leftarrow \mathcal{P} \cup \{I_L(I^*), I_R(I^*)\}$
end for
return $\mathcal{P}$

From greedy segmentation to TV approximator

Theorem

TV approximator is a greedy dichotomic segmentation.

Consequences:

- Good: very fast methods for TV approximator
- Good: we can analyze this greedy method by expressing the solution as the global minimum of an objective function
- Bad: TV approximator is no more than a greedy method...

From greedy segmentation to TV approximator

Theorem

TV approximator is a greedy dichotomic segmentation.

Consequences:

- Good: very fast methods for TV approximator
- Good: we can analyze this greedy method by expressing the solution as the global minimum of an objective function
- Bad: TV approximator is no more than a greedy method...

Technical details

- Represent an interval $[u+1, v]$ by a quadruplet $I = (u, v, \sigma_u, \sigma_v)$ where $\sigma_u, \sigma_v \in \{-1, 0, 1\}$
- Let $F_u = \sum_{i=1}^{u} Y_u$, and for $u < k < v$, $\sigma \in \{-1, 1\}$

$$f_I(k, \sigma) = \begin{cases} \sigma A_k / 2 & \text{if } \sigma_u = \sigma_v \neq 0\,, \\ A_k / (\sigma - B_k) & \text{otherwise}\ , \end{cases}$$

where

$$A_k = -F_k + \frac{(v-k)\,F_u + (k-u)\,F_v}{v-u}\,,$$
$$B_k = \frac{(v-k)\,\sigma_u + (k-u)\,\sigma_v}{v-u}\,.$$

Technical details (cont.)

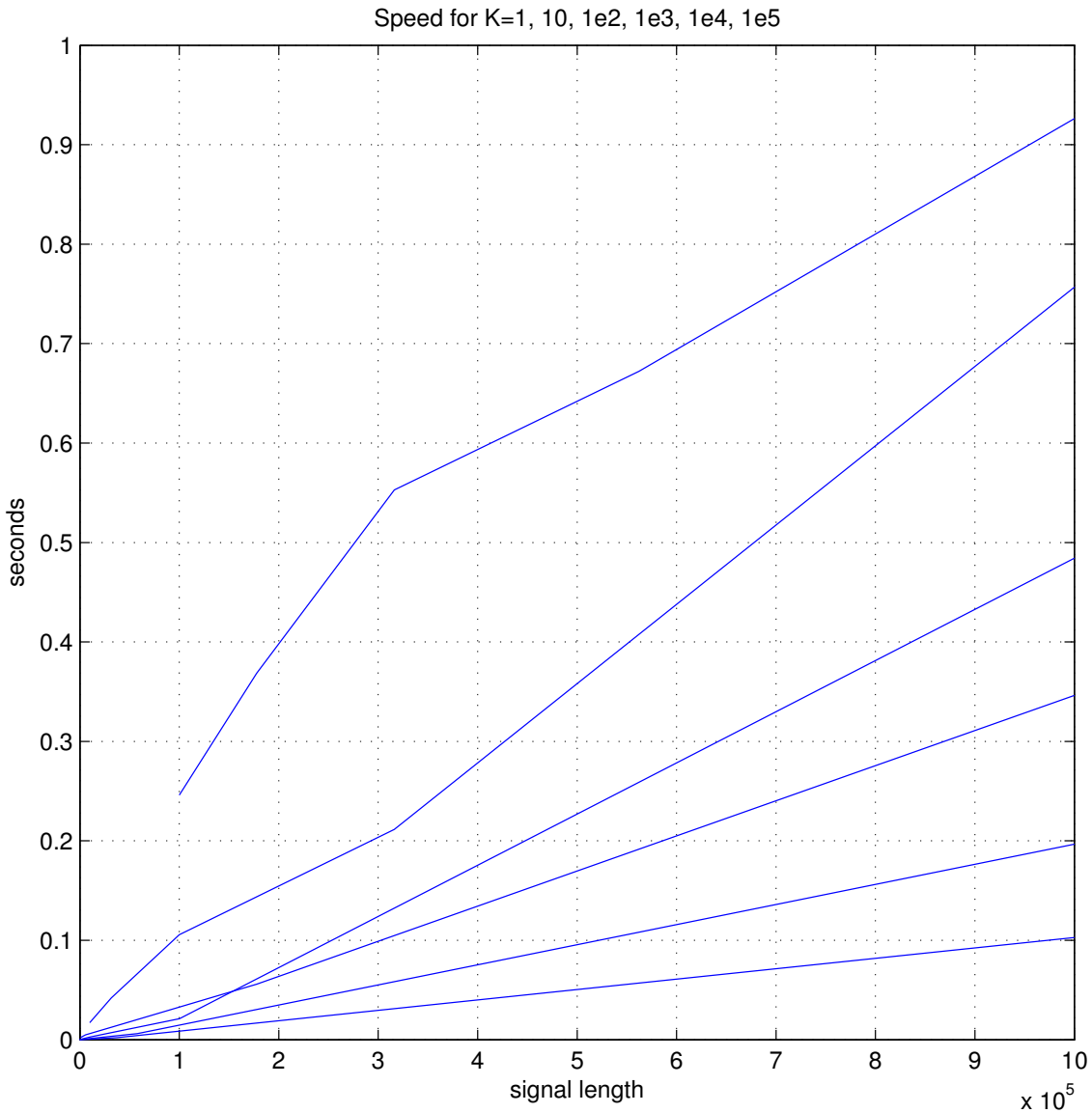
Then the functions $\gamma(I)$, $I_L(I)$ and $I_R(I)$ are respectively given by:

$$\begin{aligned}
\gamma(I) &= \max_{k \in [u+1, v-1], \sigma \in \{-1,1\}} f_I(k, \sigma) \,, \\
(k^*, \sigma^*) &= \operatorname*{argmax}_{k \in [u+1, v-1], \sigma \in \{-1,1\}} f_I(k, \sigma) \,, \\
I_L(I) &= (u, k^*, \sigma_u, \sigma^*) \,, \\
I_R(I) &= (k^*, v, \sigma^*, \sigma_v) \,.
\end{aligned}$$

Proof (sketch)

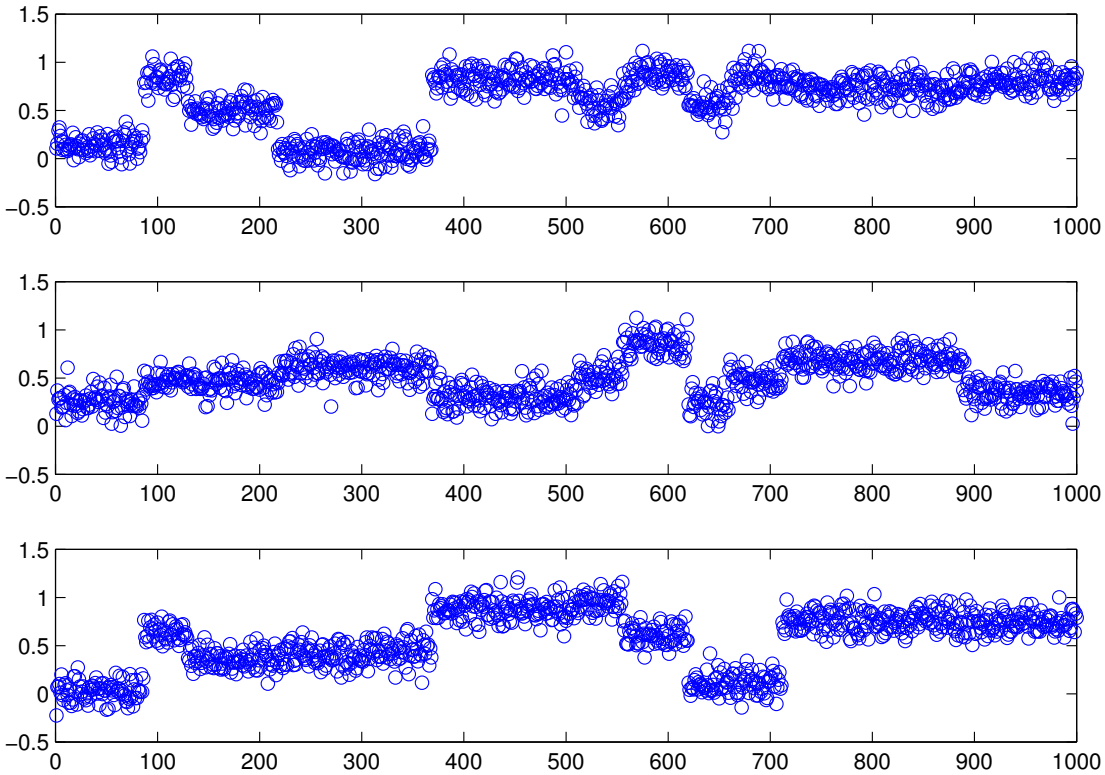
- Homotopy method (LARS)
- Similar to Harchaoui and Levy-Leduc (2008), removing superfluous computations
- The next breakpoint in a segment, and the μ where it appears, is independent of events in other segments

Speed trial : 2 s. for $K = 100$, $p = 10^7$



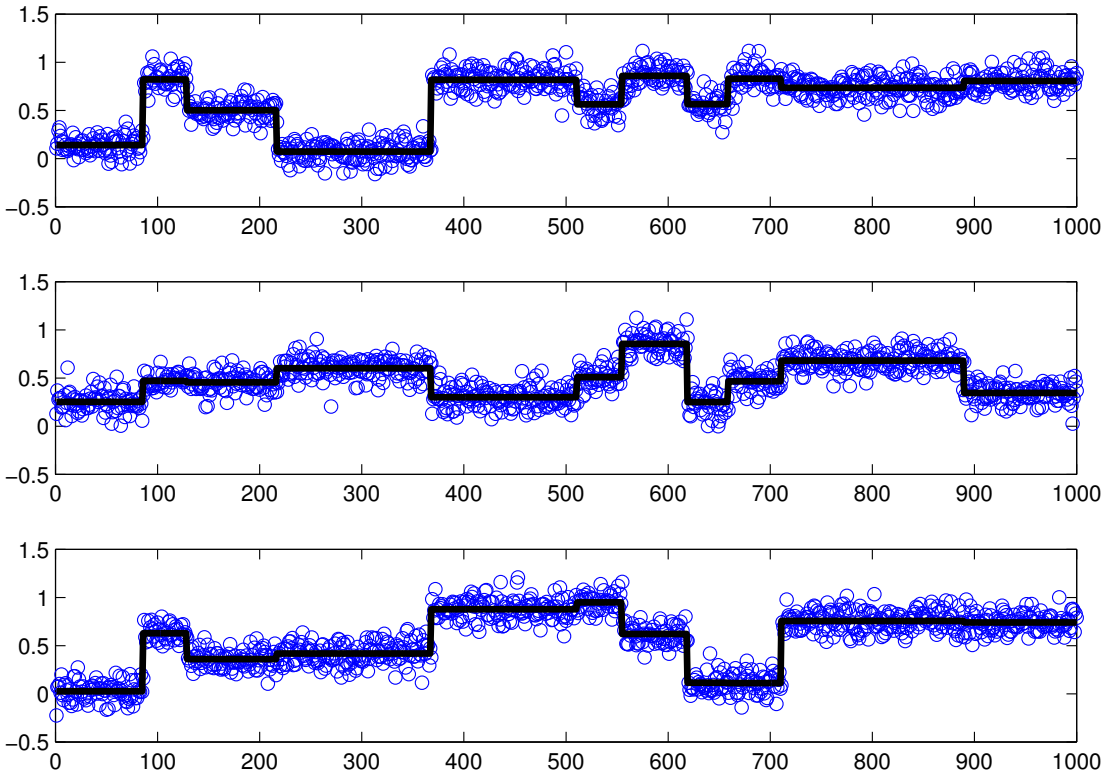
Outline

The problem



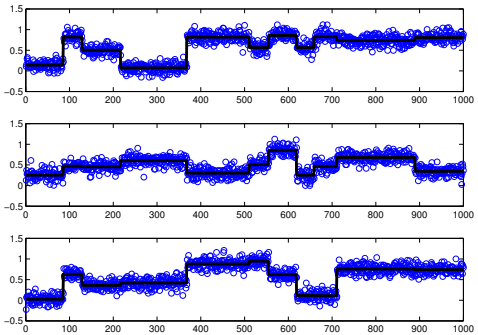
- Let $Y \in \mathbb{R}^{p \times n}$ the n signals of length p
- We want to find a piecewise constant approximation $\hat{U} \in \mathbb{R}^{p \times n}$ with at most k change-points.

The problem



- Let $Y \in \mathbb{R}^{p \times n}$ the n signals of length p
- We want to find a piecewise constant approximation $\hat{U} \in \mathbb{R}^{p \times n}$ with at most k change-points.

"Optimal" segmentation by dynamic programming



- Define the "optimal" piecewise constant approximation $\hat{U} \in \mathbb{R}^{p \times n}$ of Y as the solution of

$$\min_{U \in \mathbb{R}^{p \times n}} \| Y - U \|^2 \quad \text{such that} \quad \sum_{i=1}^{p-1} \mathbf{1}\left(U_{i+1,\bullet} \neq U_{i,\bullet}\right) \leq k$$

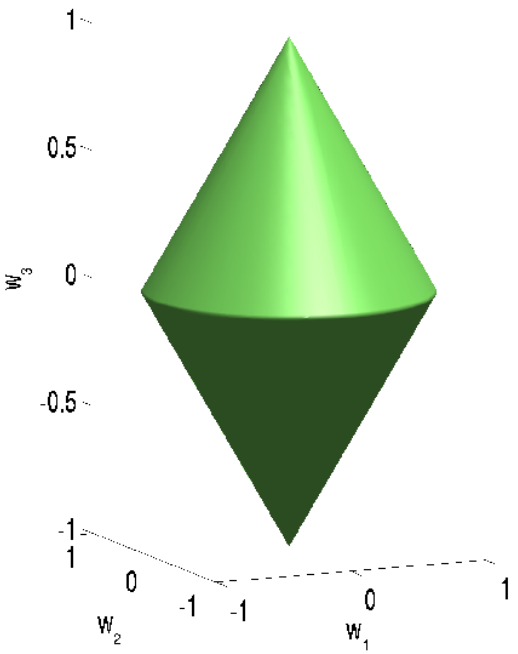
- DP finds the solution in $O(p^2 kn)$ in time and $O(p^2)$ in memory
- But: does not scale to $p = 10^6 \sim 10^9$...

Selecting pre-defined groups of variables

Group lasso (Yuan & Lin, 2006)

If groups of covariates are likely to be selected together, the ℓ_1/ℓ_2-norm induces sparse solutions *at the group level*:

$$\Omega_{group}(w) = \sum_g \|w_g\|_2$$



$$\begin{aligned}\Omega(w_1, w_2, w_3) &= \|(w_1, w_2)\|_2 + \|w_3\|_2 \\ &= \sqrt{w_1^2 + w_2^2} + \sqrt{w_3^2}\end{aligned}$$

TV approximator for many signals

- Replace

$$\min_{U \in \mathbb{R}^{p \times n}} \| Y - U \|^2 \quad \text{such that} \quad \sum_{i=1}^{p-1} \mathbf{1} \left(U_{i+1,\bullet} \neq U_{i,\bullet} \right) \leq k$$

by

$$\min_{U \in \mathbb{R}^{p \times n}} \| Y - U \|^2 \quad \text{such that} \quad \sum_{i=1}^{p-1} w_i \| U_{i+1,\bullet} - U_{i,\bullet} \| \leq \mu$$

Questions

- Practice: can we solve it efficiently?
- Theory: does it benefit from increasing p (for n fixed)?

TV approximator as a group Lasso problem

- Make the change of variables:

$$\begin{aligned} \gamma &= U_{1,\bullet}\,, \\ \beta_{i,\bullet} &= w_i \left(U_{i+1,\bullet} - U_{i,\bullet}\right) \quad \text{for } i = 1, \ldots, p-1\,. \end{aligned}$$

- TV approximator is then equivalent to the following group Lasso problem (Yuan and Lin, 2006):

$$\min_{\beta \in \mathbb{R}^{(p-1)\times n}} \| \bar{Y} - \bar{X}\beta \|^2 + \lambda \sum_{i=1}^{p-1} \| \beta_{i,\bullet} \|\,,$$

where $\bar{Y}$ is the centered signal matrix and $\bar{X}$ is a particular $(p-1) \times (p-1)$ design matrix.

TV approximator implementation

$$\min_{\beta \in \mathbb{R}^{(p-1)\times n}} \| \bar{Y} - \bar{X}\beta \|^2 + \lambda \sum_{i=1}^{p-1} \| \beta_{i,\bullet} \| ,$$

Theorem

The TV approximator can be solved efficiently:

- approximately with the group LARS in $O(npk)$ in time and $O(np)$ in memory
- exactly with a block coordinate descent + active set method in $O(np)$ in memory

Proof: computational tricks...

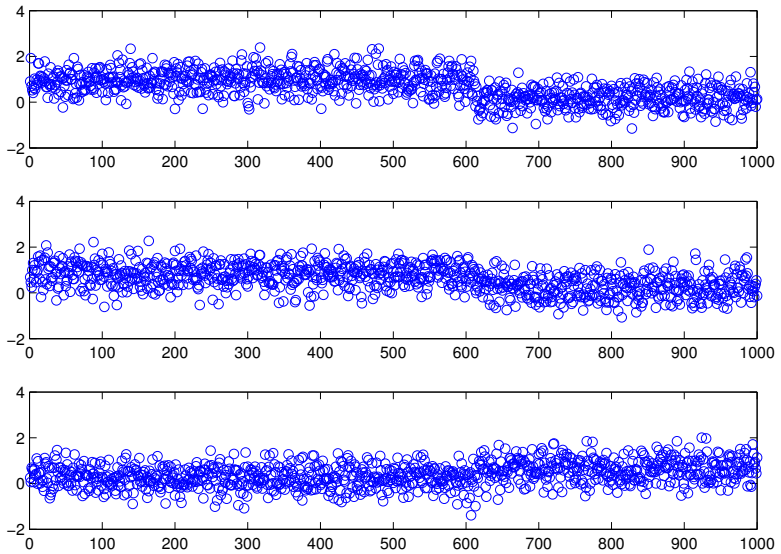
Although $\bar{X}$ is $(p-1) \times (p-1)$:

- For any $R \in \mathbb{R}^{p \times n}$, we can compute $C = \bar{X}^\top R$ in $O(np)$ operations and memory
- For any two subset of indices $A = (a_1, \ldots, a_{|A|})$ and $B = (b_1, \ldots, b_{|B|})$ in $[1, p-1]$, we can compute $\bar{X}^\top_{\bullet,A} \bar{X}_{\bullet,B}$ in $O(|A||B|)$ in time and memory
- For any $A = (a_1, \ldots, a_{|A|})$, set of distinct indices with $1 \leq a_1 < \ldots < a_{|A|} \leq p-1$, and for any $|A| \times n$ matrix R, we can compute $C = \left(\bar{X}^\top_{\bullet,A} \bar{X}_{\bullet,A}\right)^{-1} R$ in $O(|A|n)$ in time and memory

Consistency for a single change-point

Suppose a single change-point:

- at position $u = \alpha p$
- with increments $(\beta_i)_{i=1,\ldots,n}$ s.t. $\bar{\beta}^2 = \lim_{k\to\infty} \frac{1}{n} \sum_{i=1}^{n} \beta_i^2$
- corrupted by i.i.d. Gaussian noise of variance σ^2



Does the TV approximator correctly estimate the first change-point as p increases?

Consistency of the unweighted TV approximator

$$\min_{U \in \mathbb{R}^{p \times n}} \| Y - U \|^2 \quad \text{such that} \quad \sum_{i=1}^{p-1} \| U_{i+1,\bullet} - U_{i,\bullet} \| \leq \mu$$

Theorem

The unweighted TV approximator finds the correct change-point with probability tending to 1 *(resp. 0) as* $n \to +\infty$ *if* $\sigma^2 < \tilde{\sigma}_\alpha^2$ *(resp.* $\sigma^2 > \tilde{\sigma}_\alpha^2$*), where*

$$\tilde{\sigma}_\alpha^2 = p\bar{\beta}^2 \frac{(1-\alpha)^2(\alpha - \frac{1}{2p})}{\alpha - \frac{1}{2} - \frac{1}{2p}} .$$

- correct estimation on $[p\epsilon, p(1-\epsilon)]$ with $\epsilon = \sqrt{\frac{\sigma^2}{2p\bar{\beta}^2}} + o(p^{-1/2})$.
- wrong estimation near the boundaries

Consistency of the weighted TV approximator

$$\min_{U \in \mathbb{R}^{p \times n}} \| Y - U \|^2 \quad \text{such that} \quad \sum_{i=1}^{p-1} w_i \| U_{i+1,\bullet} - U_{i,\bullet} \| \leq \mu$$

Theorem

The weighted TV approximator with weights

$$\forall i \in [1, p-1] \,, \quad w_i = \sqrt{\frac{i(p-i)}{p}}$$

correctly finds the first change-point with probability tending to 1 *as* $n \to +\infty$.

- we see the benefit of increasing n
- we see the benefit of adding weights to the TV penalty

Proof sketch

- The first change-point $\hat{i}$ found by TV approximator maximizes $F_i = \| \hat{c}_{i,\bullet} \|^2$, where

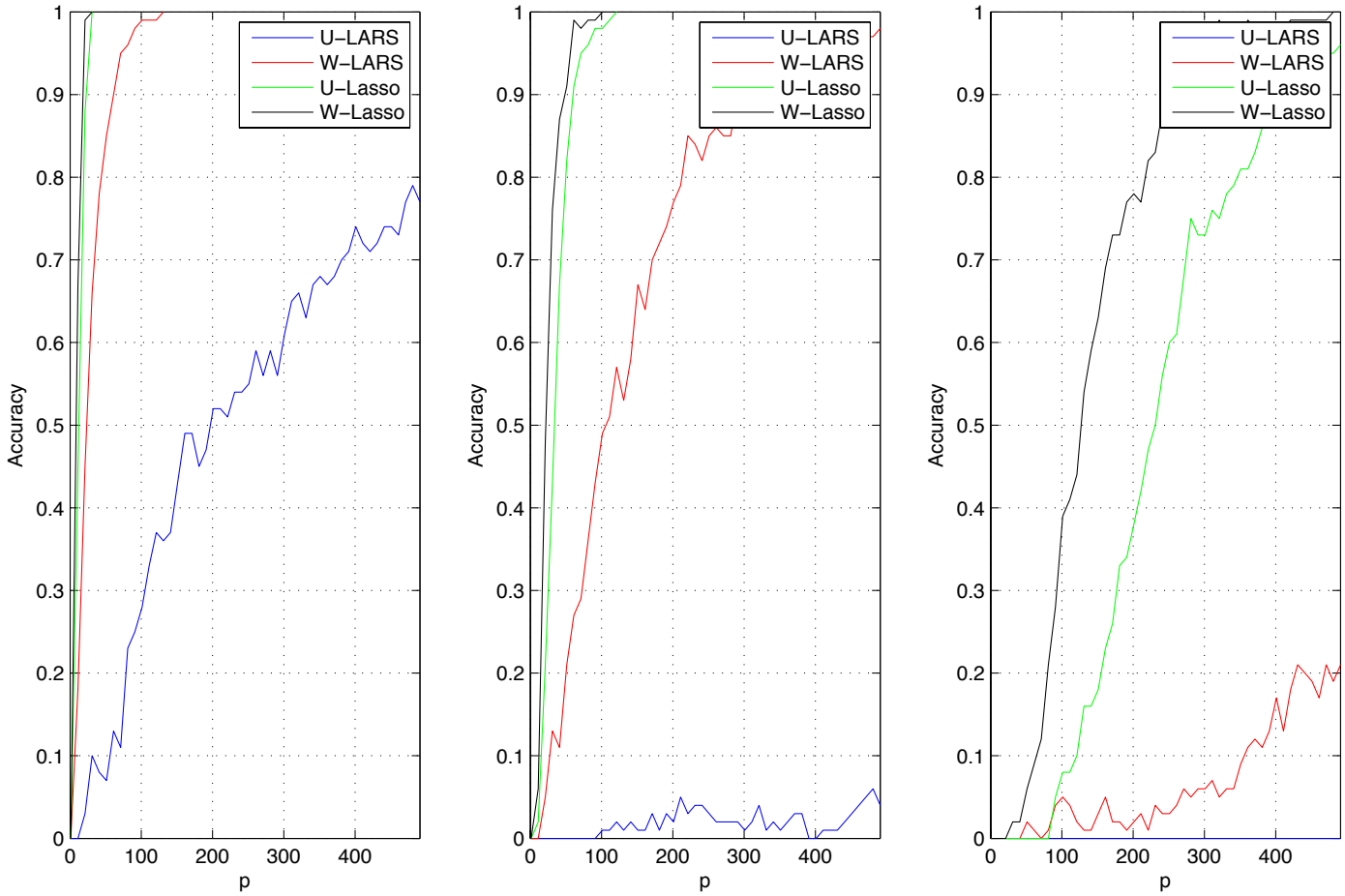
$$\hat{c} = \bar{X}^\top \bar{Y} = \bar{X}^\top \bar{X} \beta^* + \bar{X}^\top W \,.$$

- $\hat{c}$ is Gaussian, and F_i is follows a non-central χ^2 distribution with

$$G_i = \frac{EF_i}{p} = \frac{i(p-i)}{pw_i^2}\sigma^2 + \frac{\bar{\beta}^2}{w_i^2 w_u^2 p^2} \times \begin{cases} i^2 (p-u)^2 & \text{if } i \leq u \,, \\ u^2 (p-i)^2 & \text{otherwise.} \end{cases}$$

- We then just check when $G_u = \max_i G_i$

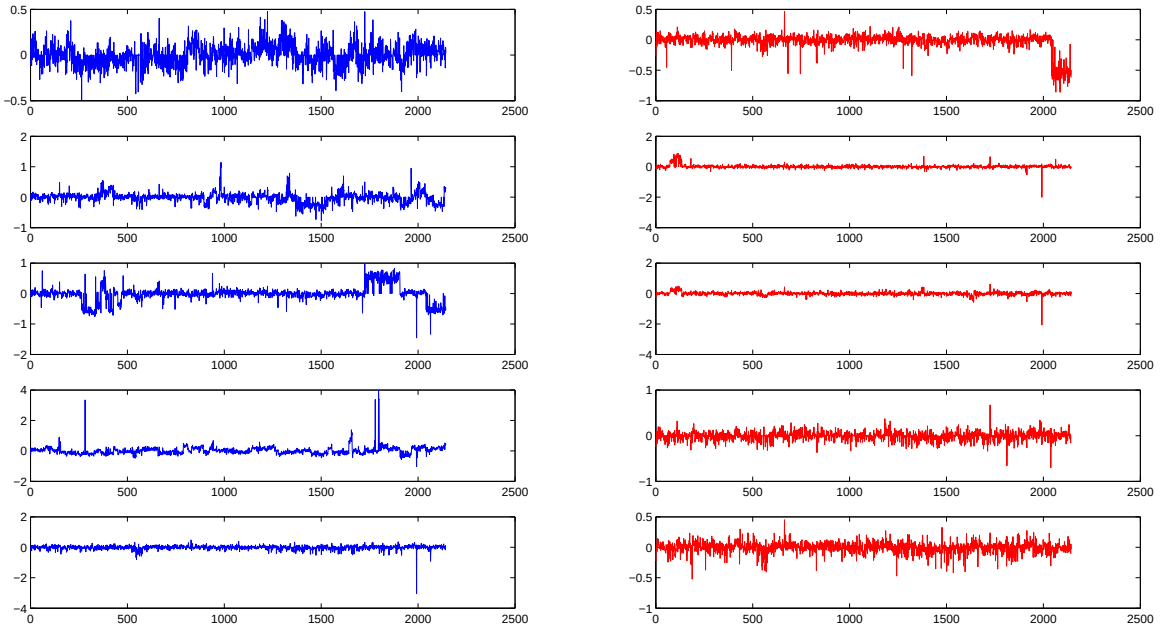
Consistent estimation of more change-points?



$p = 100, k = 10, \bar{\beta}^2 = 1, \sigma^2 \in \{0.05; 0.2; 1\}$

Outline

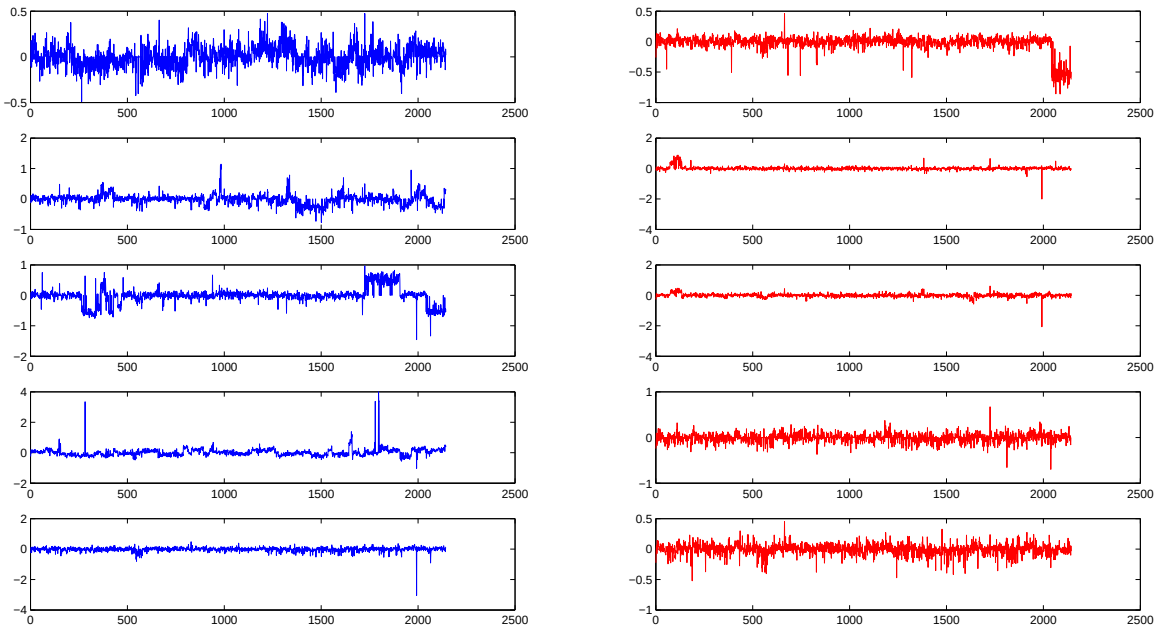
The problem



- $x_1, \ldots, x_n \in \mathbb{R}^p$ the n profiles of length p
- $y_1, \ldots, y_n \in [-1, 1]$ the labels
- We want to learn a function $f : \mathbb{R}^p \longrightarrow [-1, 1]$

Prior knowledge

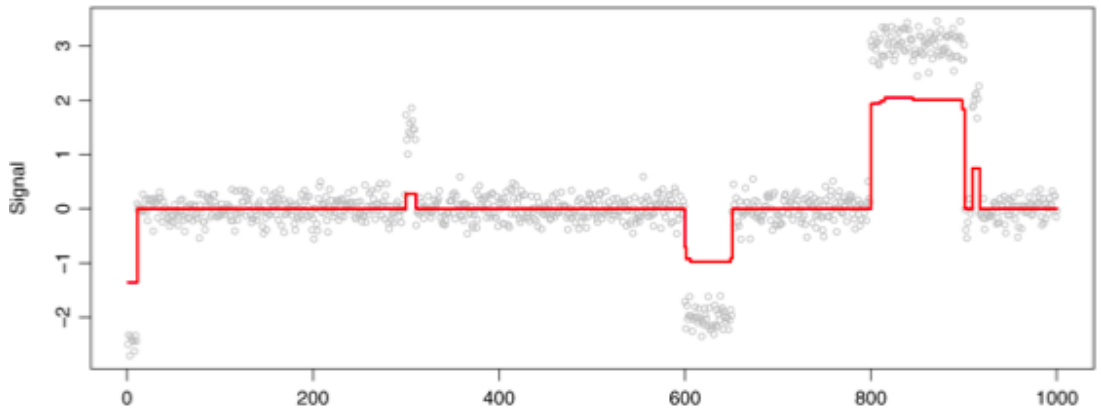
- Sparsity : not all positions should be discriminative, and we want to identify the predictive region (presence of oncogenes or tumor suppressor genes?)
- Piecewise constant : within a selected region, all probes should contribute equally



Fused Lasso signal approximator (Tibshirani et al., 2005)

$$\min_{\beta \in \mathbb{R}^p} \sum_{i=1}^{p} (y_i - \beta_i)^2 + \lambda_1 \sum_{i=1}^{p} |\beta_i| + \lambda_2 \sum_{i=1}^{p-1} |\beta_{i+1} - \beta_i| \,.$$

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



Fused lasso for supervised classification (Rapaport et al., 2008)

$$\min_{\beta \in \mathbb{R}^p} \sum_{i=1}^{n} \ell\left(y_i, \beta^\top x_i\right) + \lambda_1 \sum_{i=1}^{p} |\beta_i| + \lambda_2 \sum_{i=1}^{p-1} |\beta_{i+1} - \beta_i| \,.$$

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

Implementation

- When ℓ is the hinge loss (fused SVM), this is a linear program -> up to $p = 10^3 \sim 10^4$
- When ℓ is convex and smooth (logistic, quadratic), efficient implementation with proximal methods -> up to $p = 10^8 \sim 10^9$

Fused lasso for supervised classification (Rapaport et al., 2008)

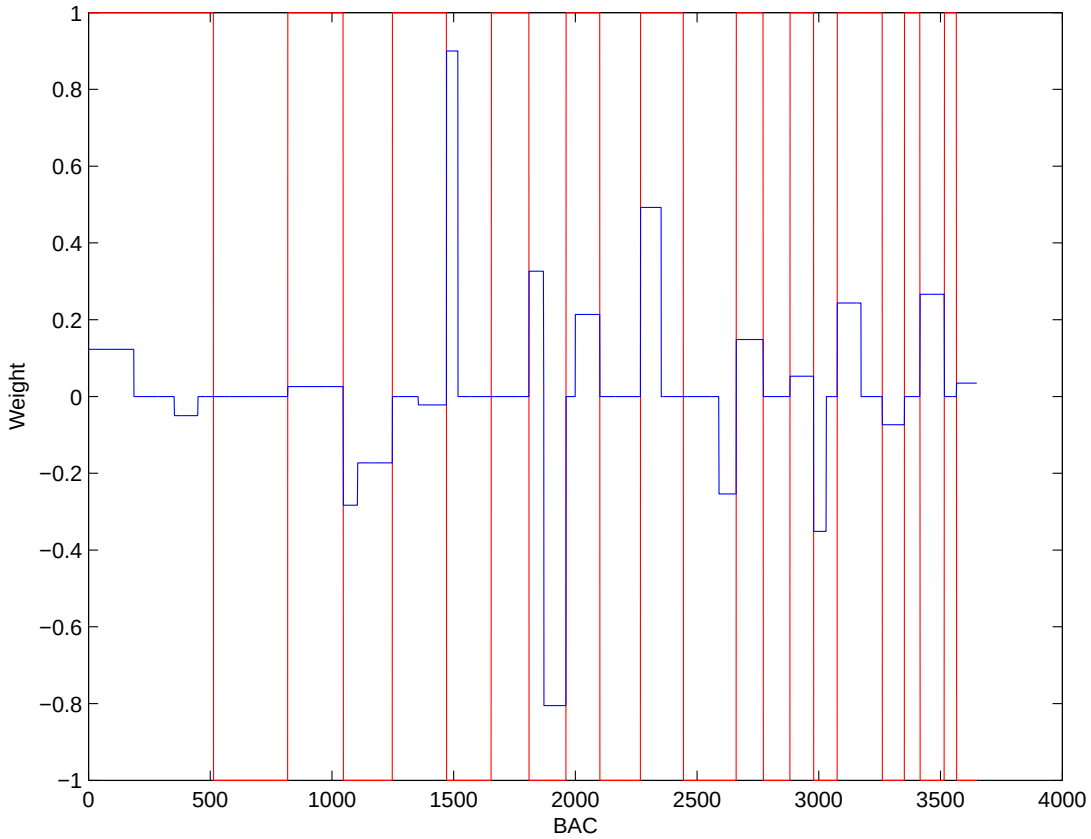
$$\min_{\beta \in \mathbb{R}^p} \sum_{i=1}^{n} \ell\left(y_i, \beta^\top x_i\right) + \lambda_1 \sum_{i=1}^{p} |\beta_i| + \lambda_2 \sum_{i=1}^{p-1} |\beta_{i+1} - \beta_i| \,.$$

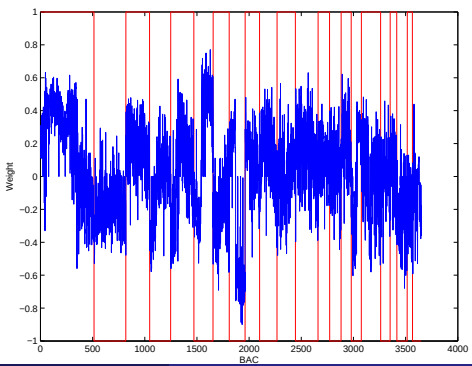
where ℓ is, e.g., the hinge loss $\ell(y,t) = max(1 - yt, 0)$.

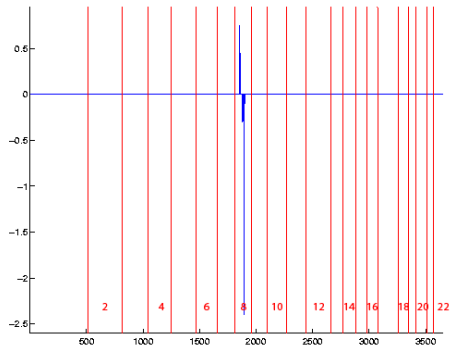
Implementation

- When ℓ is the hinge loss (fused SVM), this is a linear program -> up to $p = 10^3 \sim 10^4$
- When ℓ is convex and smooth (logistic, quadratic), efficient implementation with proximal methods -> up to $p = 10^8 \sim 10^9$

Example: predicting metastasis in melanoma

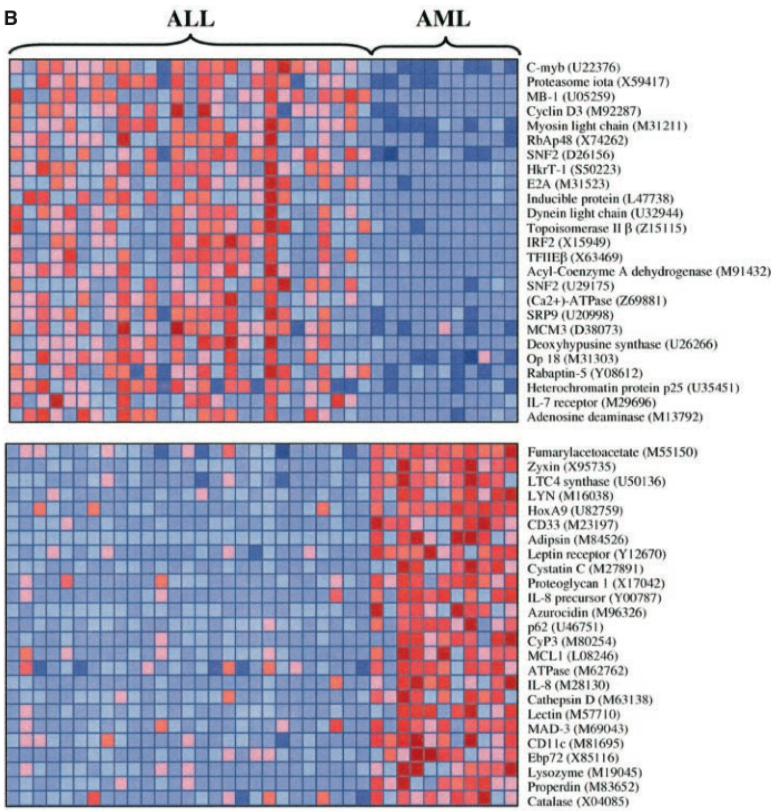


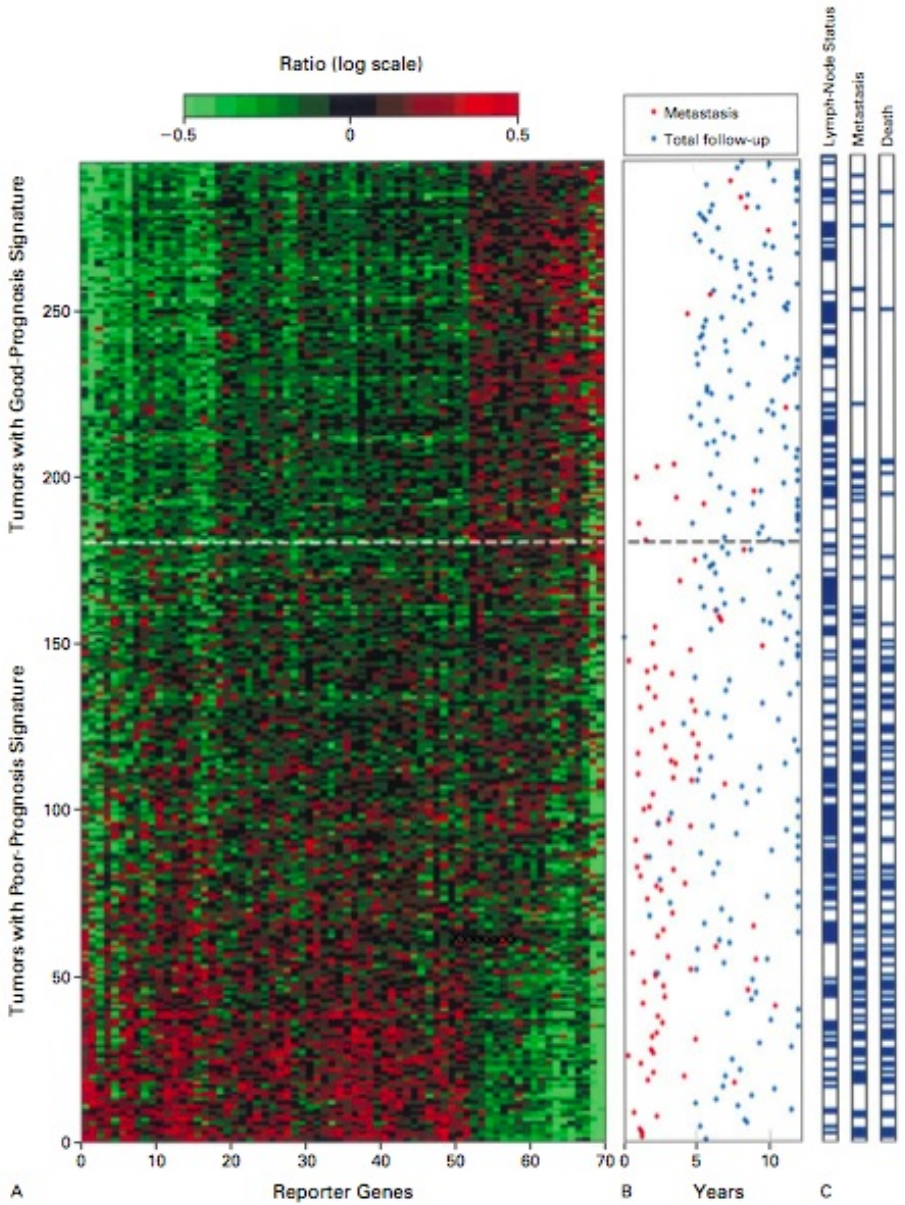




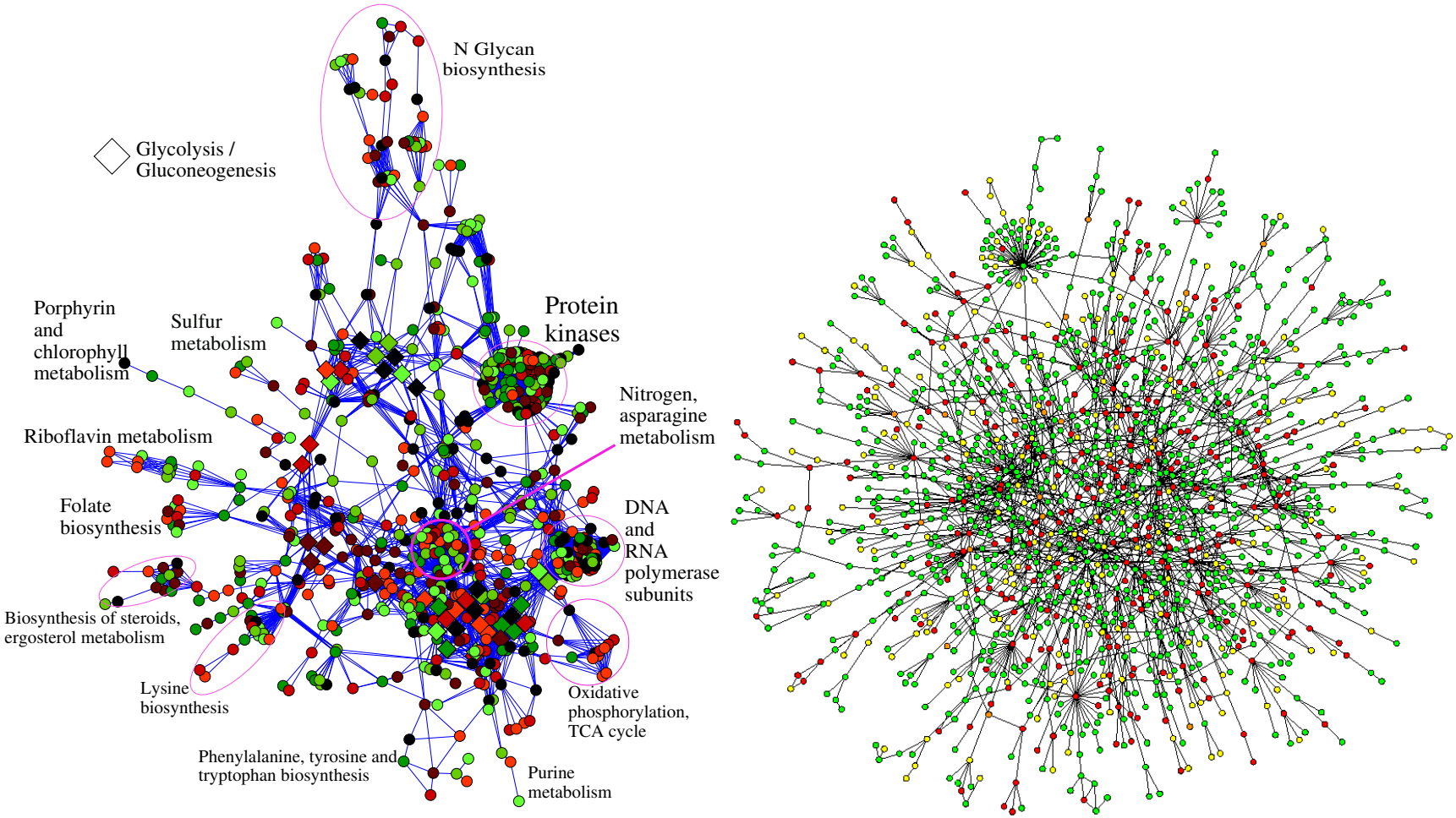
Outline

Molecular diagnosis / prognosis / theragnosis





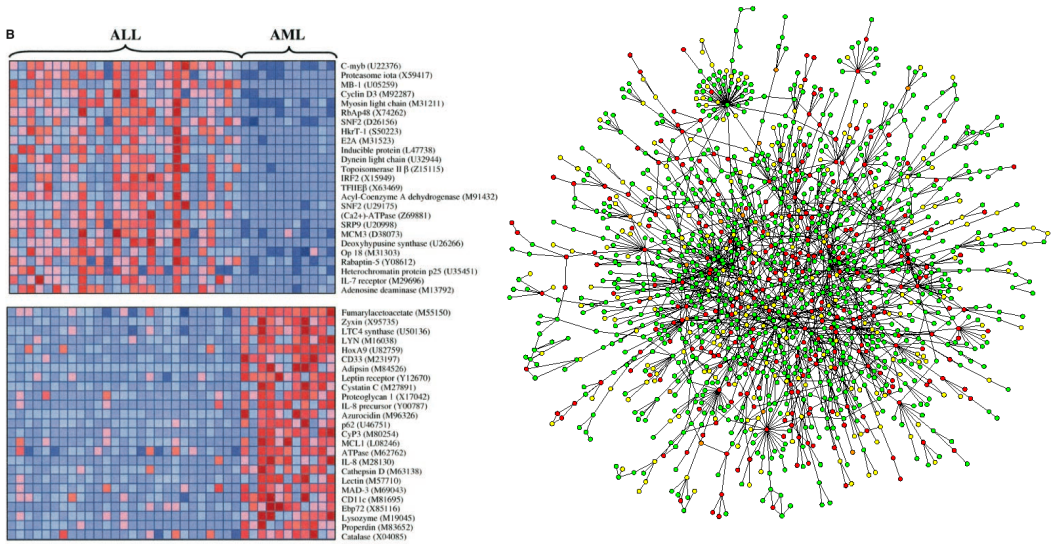
Gene networks



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

$$\min_{\beta} R(\beta) + \lambda \Omega_G(\beta)$$

Hypothesis

We would like to design penalties $\Omega_G(\beta)$ to promote one of the following hypothesis:

- Hypothesis 1: genes near each other on the graph should have similar weights (but we do not try to select only a few genes), i.e., the classifier should be smooth on the graph
- Hypothesis 2: genes selected in the signature should be connected to each other, or be in a few known functional groups, without necessarily having similar weights.

Graph based penalty

Prior hypothesis

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

An idea (Rapaport et al., 2007)

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

$$\min_{\beta \in \mathbb{R}^p} R(\beta) + \lambda \sum_{i \sim j} (\beta_i - \beta_j)^2 \,.$$

Graph based penalty

Prior hypothesis

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

An idea (Rapaport et al., 2007)

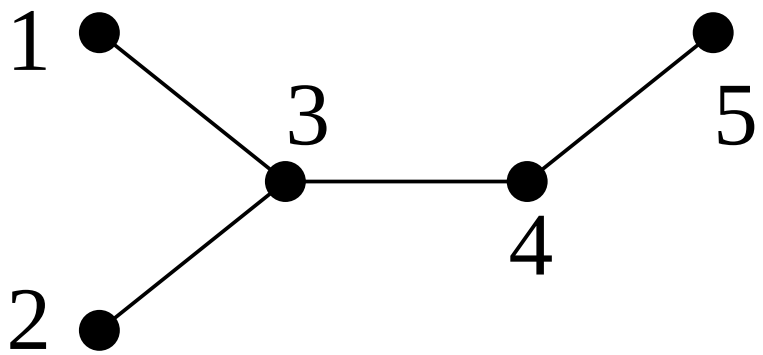
$$\Omega_{spectral}(\beta) = \sum_{i \sim j} (\beta_i - \beta_j)^2 \,,$$

$$\min_{\beta \in \mathbb{R}^p} R(\beta) + \lambda \sum_{i \sim j} (\beta_i - \beta_j)^2 \,.$$

Graph Laplacian

Definition

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



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

Spectral penalty as a kernel

Theorem

The function $f(x) = \beta^\top x$ where b is solution of

$$\min_{\beta \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^{n} l\left(\beta^\top x_i, y_i\right) + \lambda \sum_{i \sim j} \left(\beta_i - \beta_j\right)^2$$

is equal to $g(x) = \gamma^\top \Phi(x)$ where γ is solution of

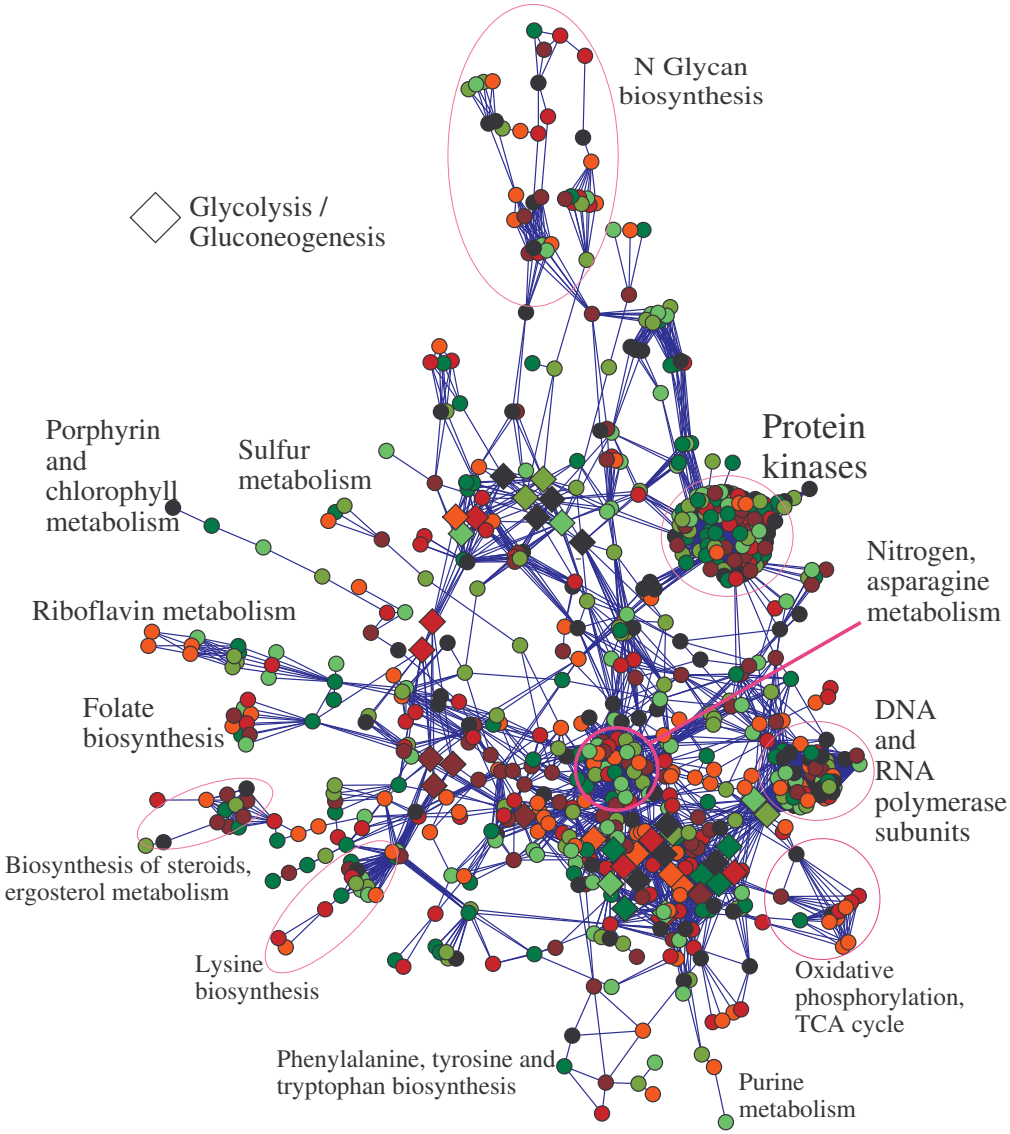
$$\min_{\gamma \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^{n} l\left(\gamma^\top \Phi(x_i), y_i\right) + \lambda \gamma^\top \gamma \,,$$

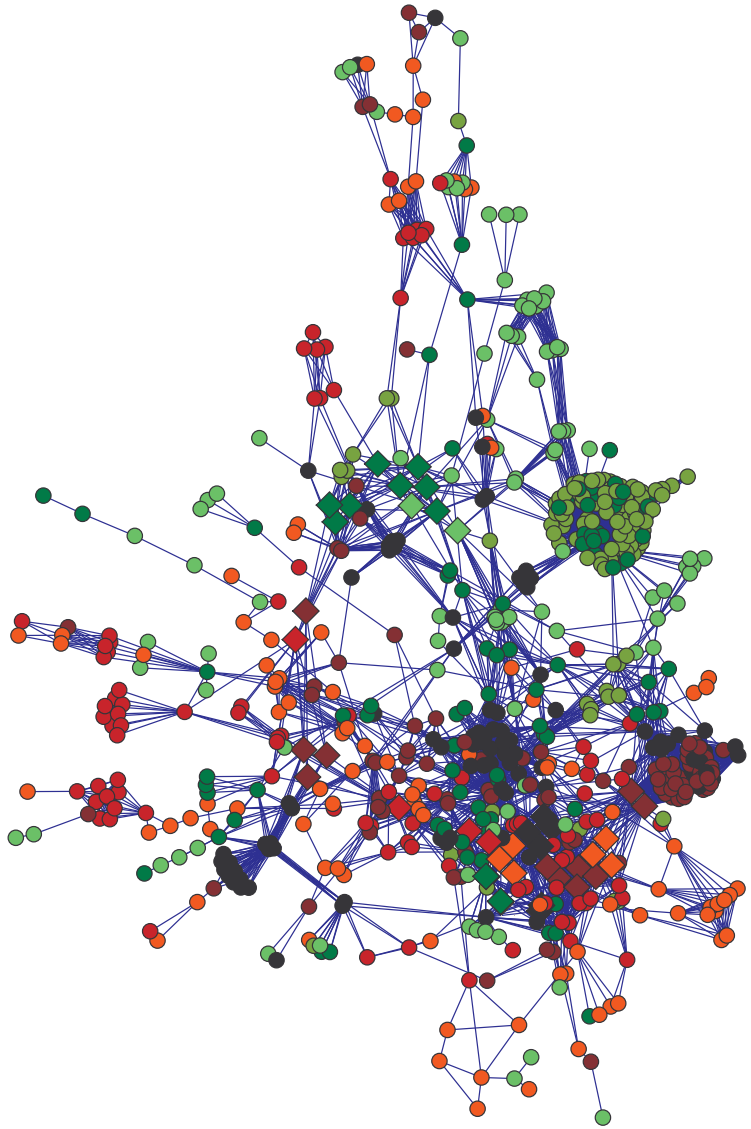
and where

$$\Phi(x)^\top \Phi(x') = x^\top K_G x'$$

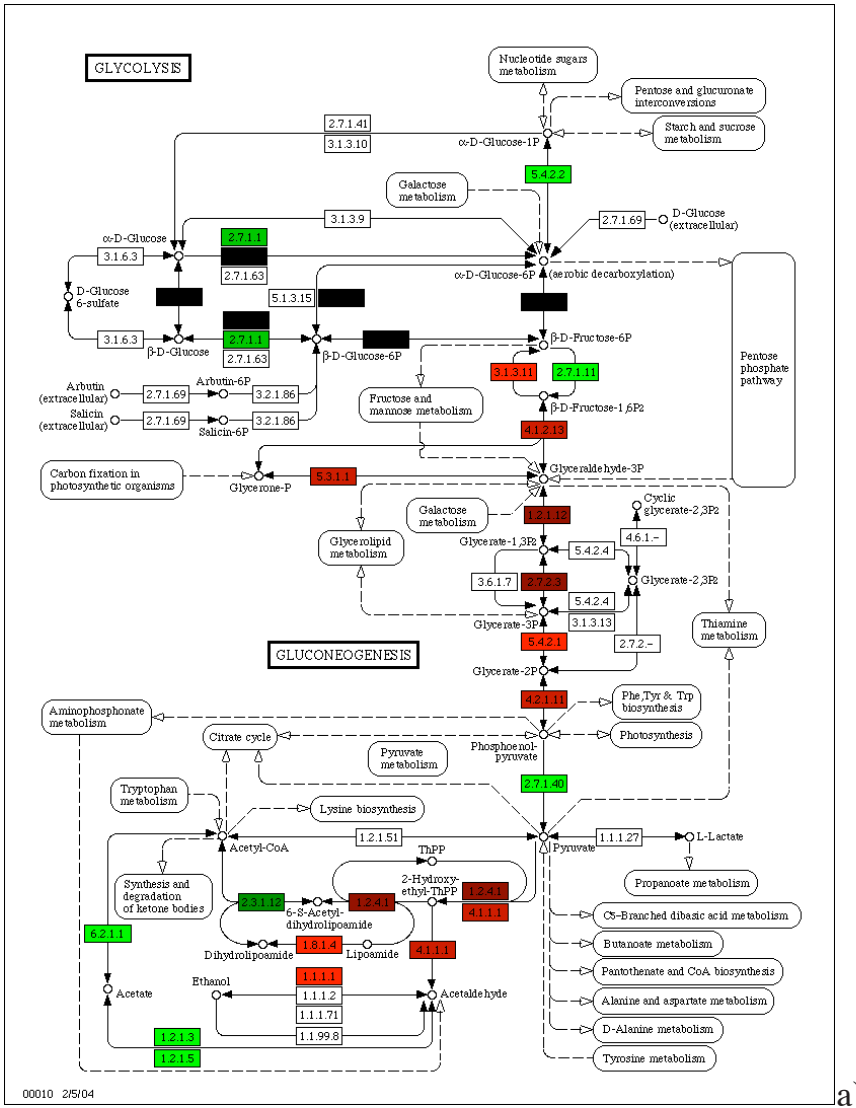
for $K_G = L^*$, the pseudo-inverse of the graph Laplacian.

Classifiers

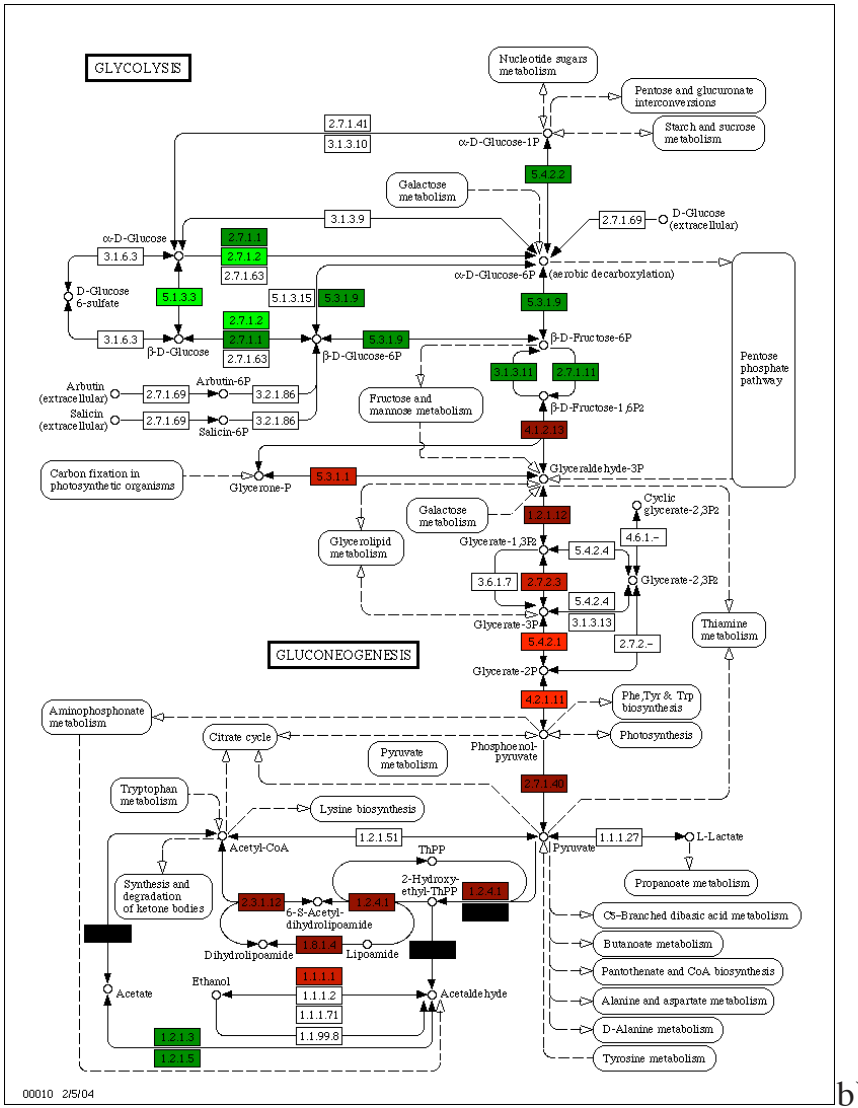




Classifier



a)



b)

Other penalties with kernels

$$\Phi(x)^\top \Phi(x') = x^\top K_G x'$$

with:

- $K_G = (c + L)^{-}1$ leads to

$$\Omega(\beta) = c \sum_{i=1}^{p} \beta_i^2 + \sum_{i \sim j} (\beta_i - \beta_j)^2 \ .$$

- The diffusion kernel:

$$K_G = \exp_M(-2tL) \, .$$

penalizes high frequencies of β in the Fourier domain.

Other penalties without kernels

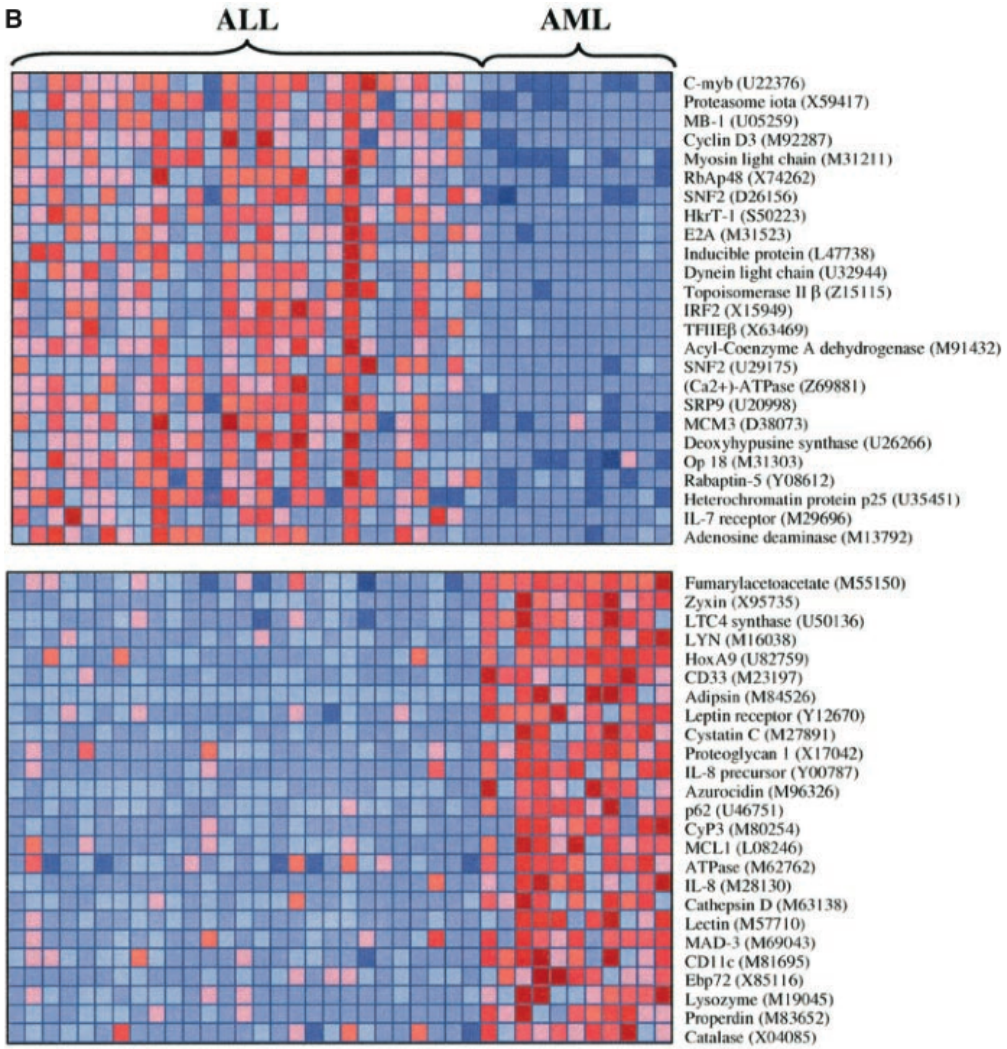
- Gene selection + Piecewise constant on the graph

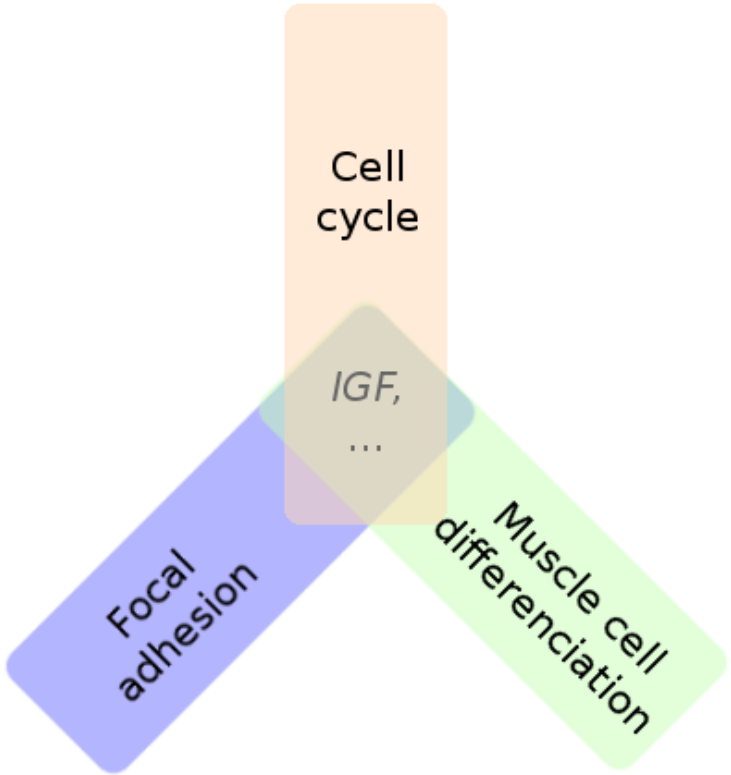
$$\Omega(\beta) = \sum_{i \sim j} |\beta_i - \beta_j| + \sum_{i=1}^{p} |\beta_i|$$

- Gene selection + smooth on the graph

$$\Omega(\beta) = \sum_{i \sim j} (\beta_i - \beta_j)^2 + \sum_{i=1}^{p} |\beta_i|$$

How to select jointly genes belonging to predefined pathways?



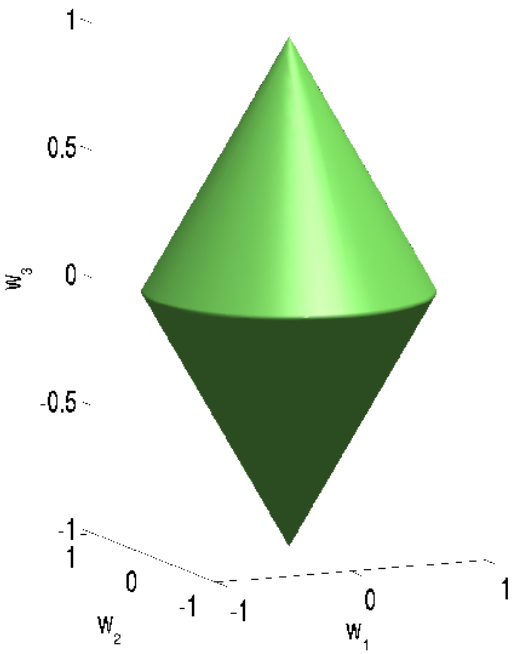


Selecting pre-defined groups of variables

Group lasso (Yuan & Lin, 2006)

If groups of covariates are likely to be selected together, the ℓ_1/ℓ_2-norm induces sparse solutions *at the group level*:

$$\Omega_{group}(w) = \sum_g \|w_g\|_2$$

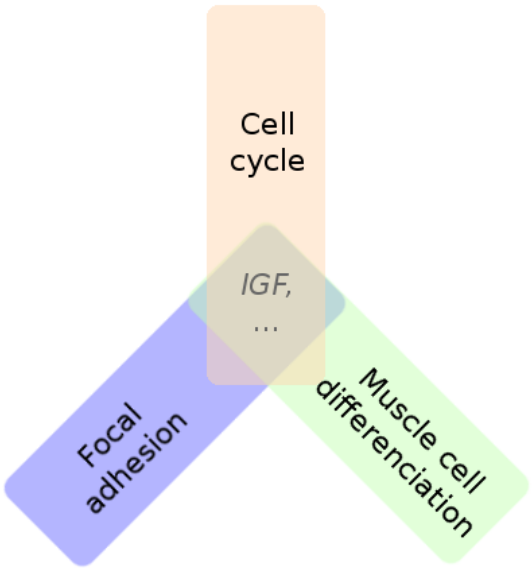


$$\Omega(w_1, w_2, w_3) = \|(w_1, w_2)\|_2 + \|w_3\|_2$$

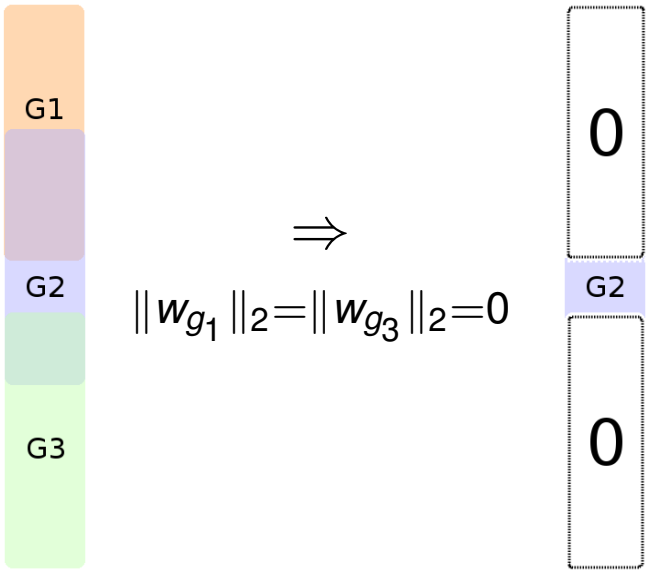
What if a gene belongs to several groups?

Issue of using the group-lasso

- $\Omega_{group}(w) = \sum_g \|w_g\|_2$ sets groups to 0.
- One variable is selected $\Leftrightarrow$ all the groups to which it belongs are selected.



IGF selection $\Rightarrow$ selection of unwanted groups



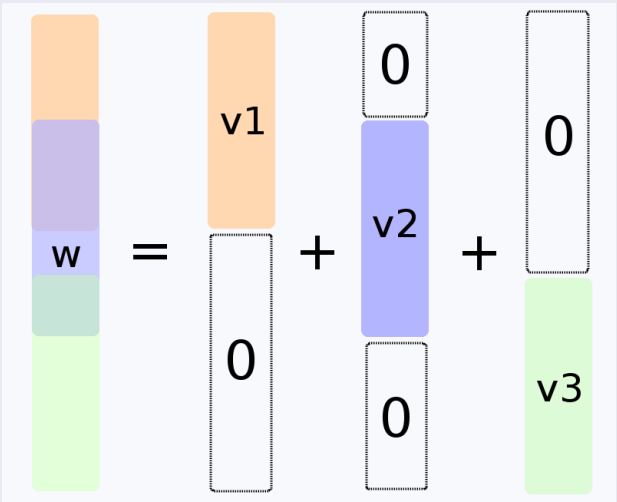
Removal of *any* group containing a gene $\Rightarrow$ the weight of the gene is 0.

Overlap norm (Jacob et al., 2009)

An idea

Introduce latent variables v_g:

$$\begin{cases} \min_{w,v} L(w) + \lambda \sum_{g \in \mathcal{G}} \|v_g\|_2 \\ w = \sum_{g \in \mathcal{G}} v_g \\ \text{supp}\,(v_g) \subseteq g. \end{cases}$$



Properties

- Resulting support is a *union* of groups in $\mathcal{G}$.
- Possible to select one variable without selecting all the groups containing it.
- Equivalent to group lasso when there is no overlap

A new norm

Overlap norm

$$\begin{cases} \min_{w,v} L(w) + \lambda \sum_{g \in \mathcal{G}} \|v_g\|_2 \\ w = \sum_{g \in \mathcal{G}} v_g \\ \text{supp}\,(v_g) \subseteq g. \end{cases} = \min_w L(w) + \lambda \Omega_{overlap}(w)$$

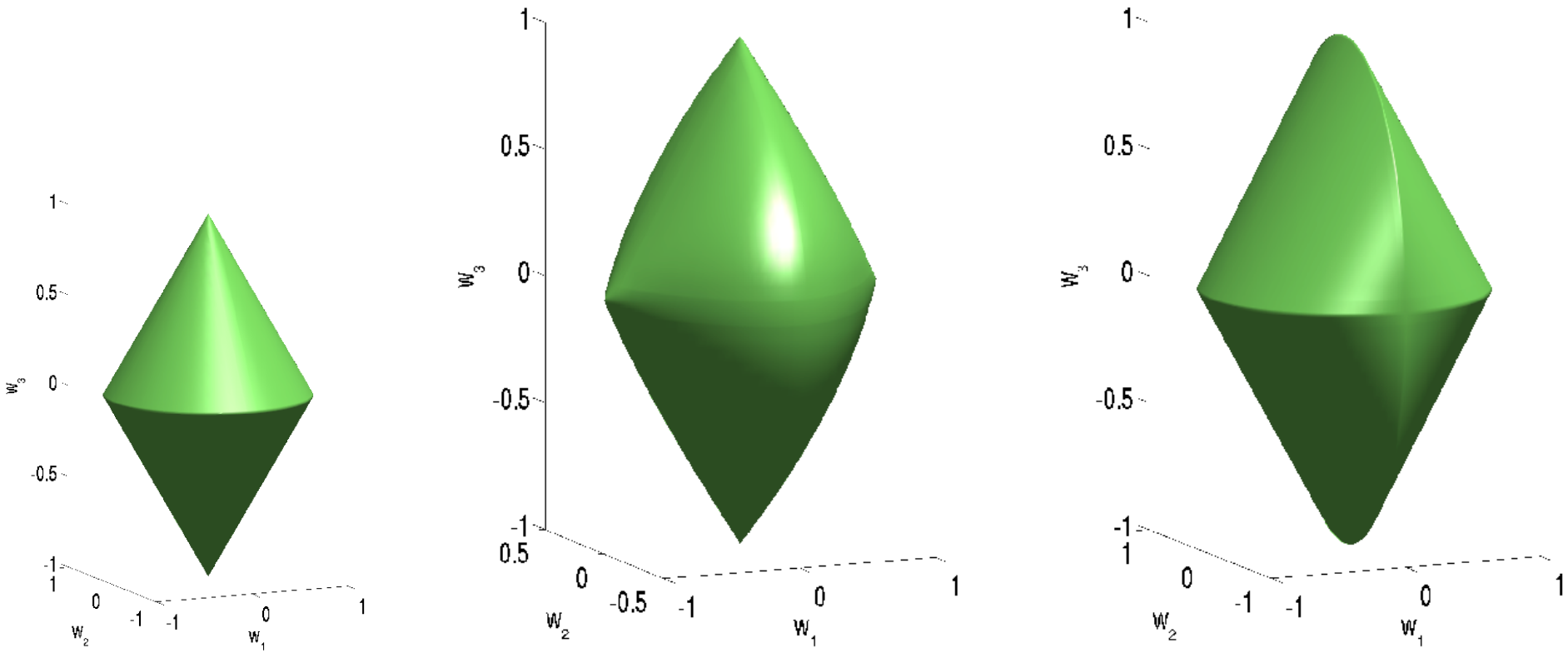
with

$$\Omega_{overlap}(w) \triangleq \begin{cases} \min_v \sum_{g \in \mathcal{G}} \|v_g\|_2 \\ w = \sum_{g \in \mathcal{G}} v_g \\ \text{supp}\,(v_g) \subseteq g. \end{cases} \quad (*)$$

Property

- $\Omega_{overlap}(w)$ is a norm of w.
- $\Omega_{overlap}(.)$ associates to w a specific (not necessarily unique) decomposition $(v_g)_{g \in \mathcal{G}}$ which is the argmin of $(*)$.

Overlap and group unity balls



Balls for $\Omega^{\mathcal{G}}_{\text{group}}(\cdot)$ (middle) and $\Omega^{\mathcal{G}}_{\text{overlap}}(\cdot)$ (right) for the groups $\mathcal{G} = \{\{1,2\},\{2,3\}\}$ where w_2 is represented as the vertical coordinate. Left: group-lasso ($\mathcal{G} = \{\{1,2\},\{3\}\}$), for comparison.

Theoretical results

Consistency in group support (Jacob et al., 2009)

- Let $\bar{w}$ be the true parameter vector.
- Assume that there exists a unique decomposition $\bar{v}_g$ such that $\bar{w} = \sum_g \bar{v}_g$ and $\Omega^{\mathcal{G}}_{\text{overlap}}(\bar{w}) = \sum \|\bar{v}_g\|_2$.
- Consider the regularized empirical risk minimization problem $L(w) + \lambda \Omega^{\mathcal{G}}_{\text{overlap}}(w)$.

Then

- under appropriate mutual incoherence conditions on X,
- as $n \to \infty$,
- with very high probability,

the optimal solution $\hat{w}$ admits a unique decomposition $(\hat{v}_g)_{g \in \mathcal{G}}$ such that

$$\left\{ g \in \mathcal{G} | \hat{v}_g \neq 0 \right\} = \left\{ g \in \mathcal{G} | \bar{v}_g \neq 0 \right\}.$$

Theoretical results

Consistency in group support (Jacob et al., 2009)

- Let $\bar{w}$ be the true parameter vector.
- Assume that there exists a unique decomposition $\bar{v}_g$ such that $\bar{w} = \sum_g \bar{v}_g$ and $\Omega^{\mathcal{G}}_{\text{overlap}}(\bar{w}) = \sum \|\bar{v}_g\|_2$.
- Consider the regularized empirical risk minimization problem $L(w) + \lambda \Omega^{\mathcal{G}}_{\text{overlap}}(w)$.

Then

- under appropriate mutual incoherence conditions on X,
- as $n \to \infty$,
- with very high probability,

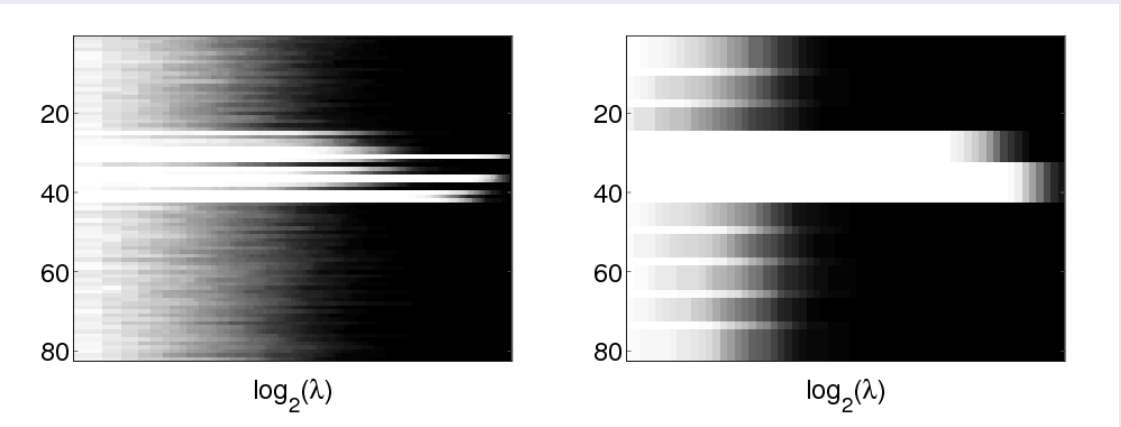
the optimal solution $\hat{w}$ admits a unique decomposition $(\hat{v}_g)_{g \in \mathcal{G}}$ such that

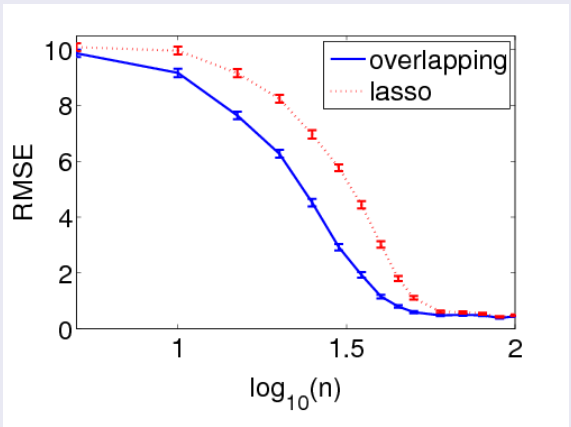
$$\left\{ g \in \mathcal{G} | \hat{v}_g \neq 0 \right\} = \left\{ g \in \mathcal{G} | \bar{v}_g \neq 0 \right\}.$$

Experiments

Synthetic data: overlapping groups

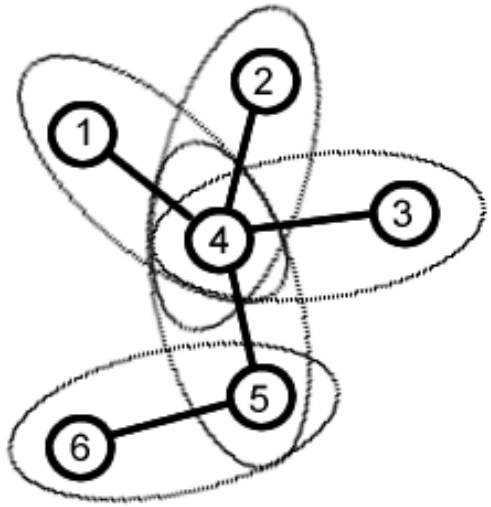
- 10 groups of 10 variables with 2 variables of overlap between two successive groups :$\{1, \ldots, 10\}, \{9, \ldots, 18\}, \ldots, \{73, \ldots, 82\}$.
- Support: union of 4*th* and 5*th* groups.
- Learn from 100 training points.





Frequency of selection of each variable with the lasso (left) and $\Omega^{\mathcal{G}}_{\text{overlap}}(.)$ (middle), comparison of the RMSE of both methods (right).

Graph lasso



Two solutions

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

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

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

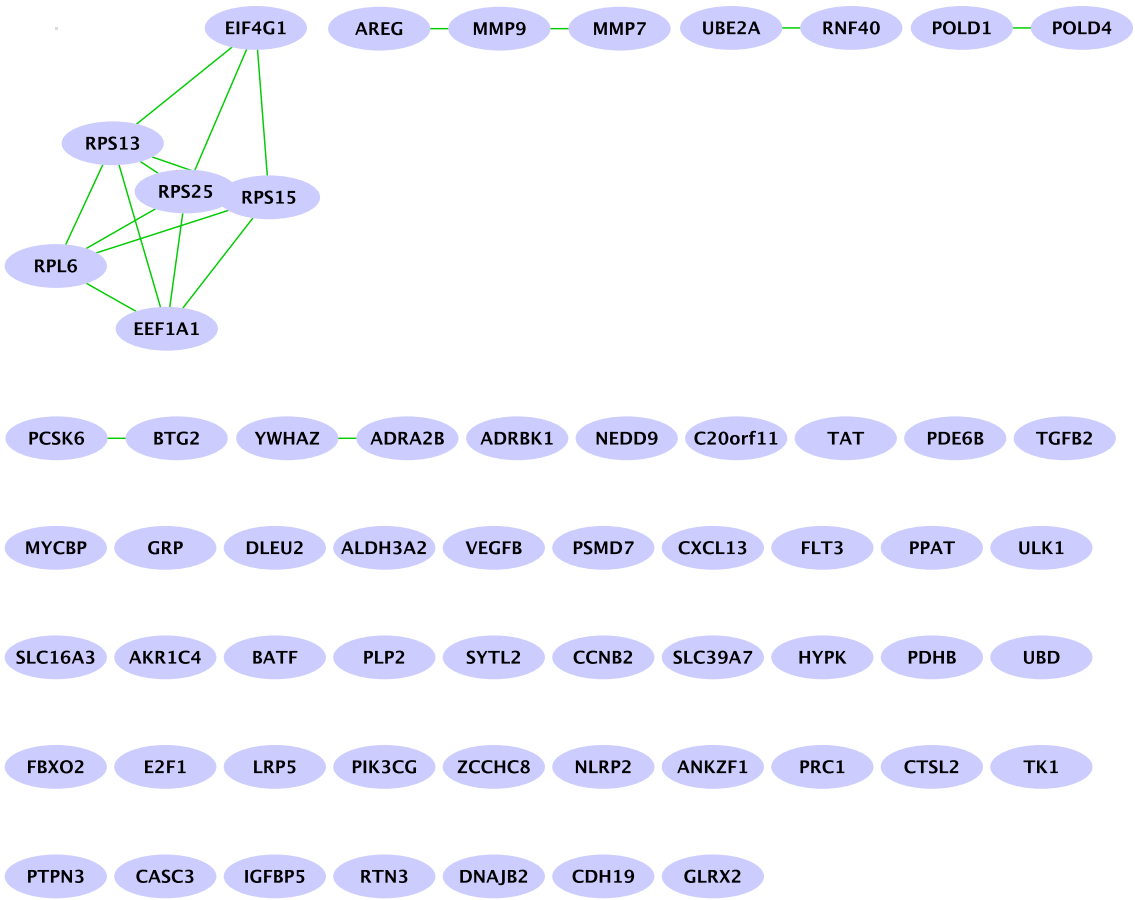
- Gene expression data for 8,141 genes in 295 breast cancer tumors.
- Canonical pathways from MSigDB containing 639 groups of genes, 637 of which involve genes from our study.

METHOD	ℓ_1	$\Omega^{\mathcal{G}}_{\text{OVERLAP}}(.)$
ERROR	0.38 ± 0.04	0.36 ± 0.03
MEAN $\sharp$ PATH.	130	30

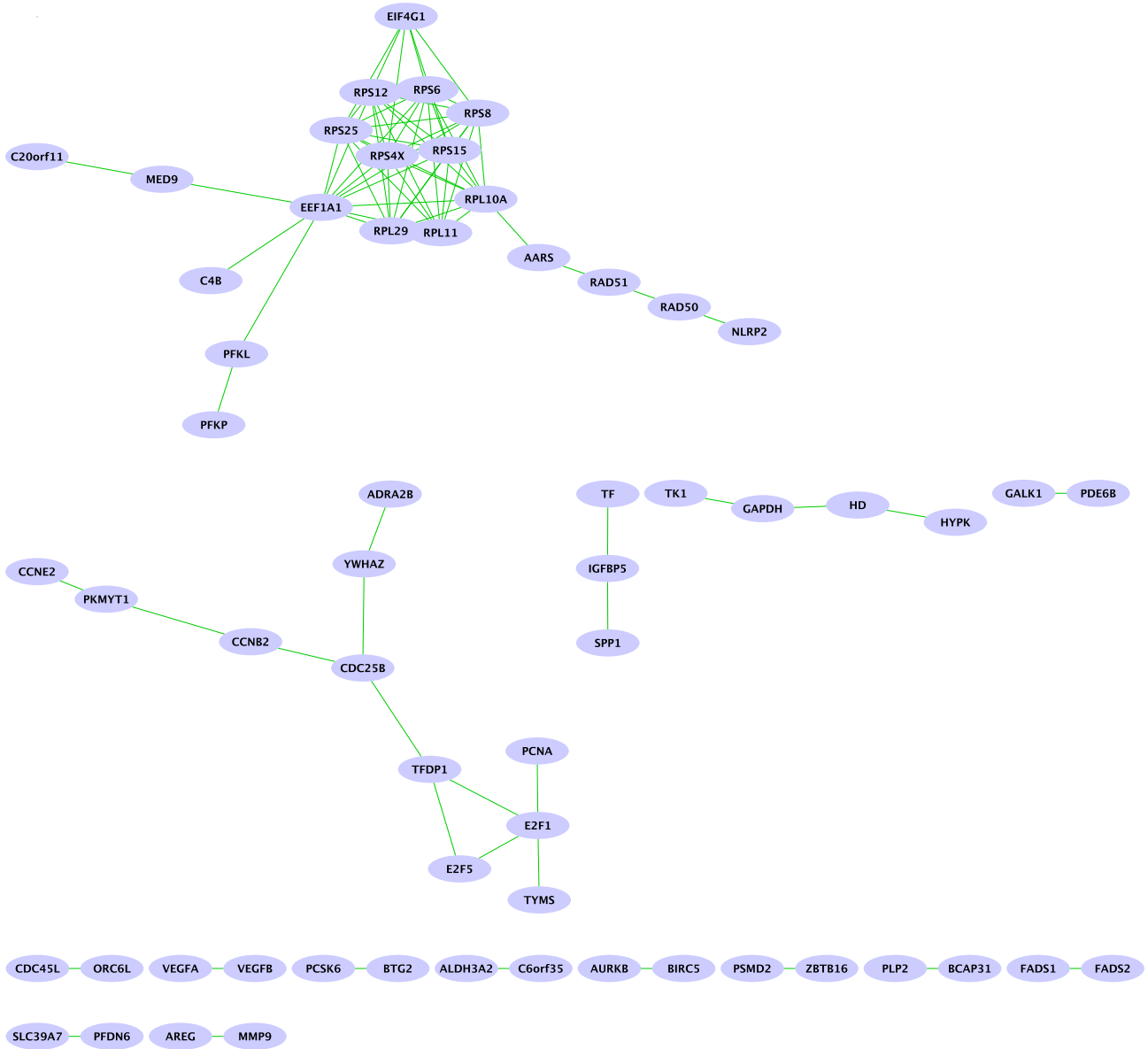
- Graph on the genes.

METHOD	ℓ_1	$\Omega_{graph}(.)$
ERROR	0.39 ± 0.04	0.36 ± 0.01
AV. SIZE C.C.	1.03	1.30

Lasso signature



Graph Lasso signature



Outline

Conclusions

- Feature / pattern selection in high dimension is central for many applications
- People excited about embedded methods (convex optimization), ensemble methods... but many disappointing results when tested on real data
  - Filter methods not so bad
  - Ensemble learning useful?
- Need for more theory to explain practical observations, suggest new methods
- Structured sparsity / pattern discovery is a promising direction
- Need to adjust the difficulty of the inference problem to the data available