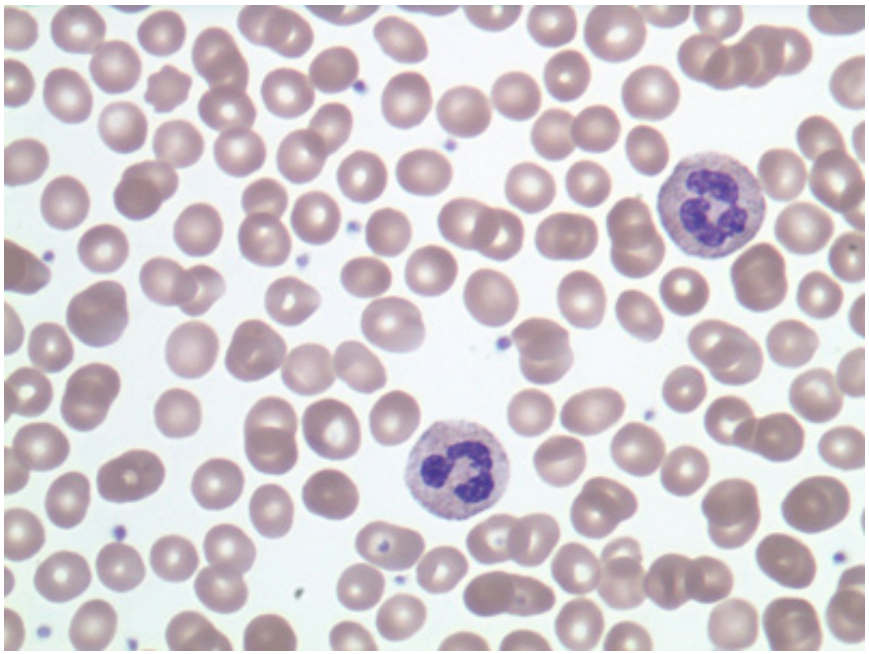
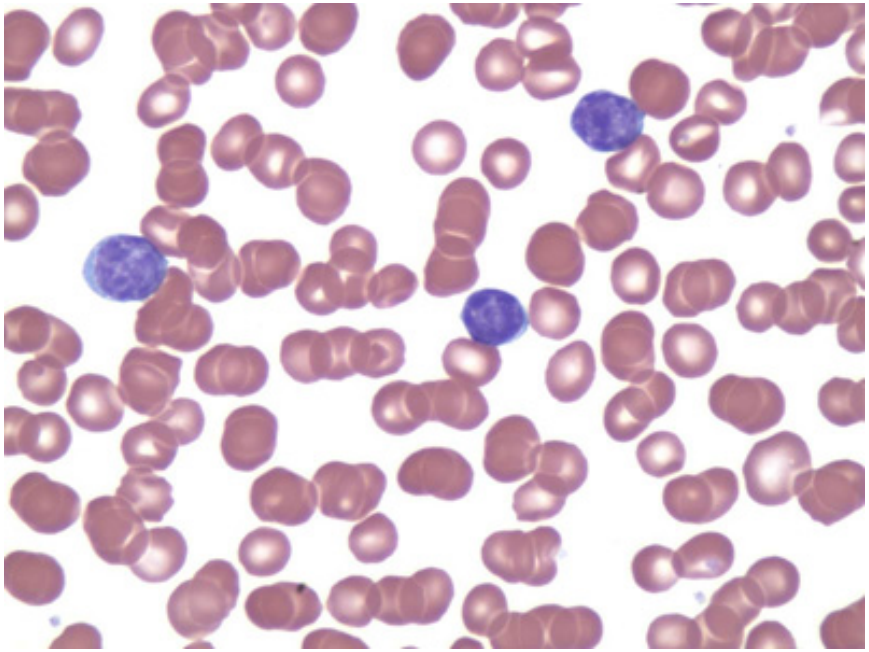
Fast sparse methods for genomics data

Jean-Philippe Vert

Chalmers University, April 16, 2013

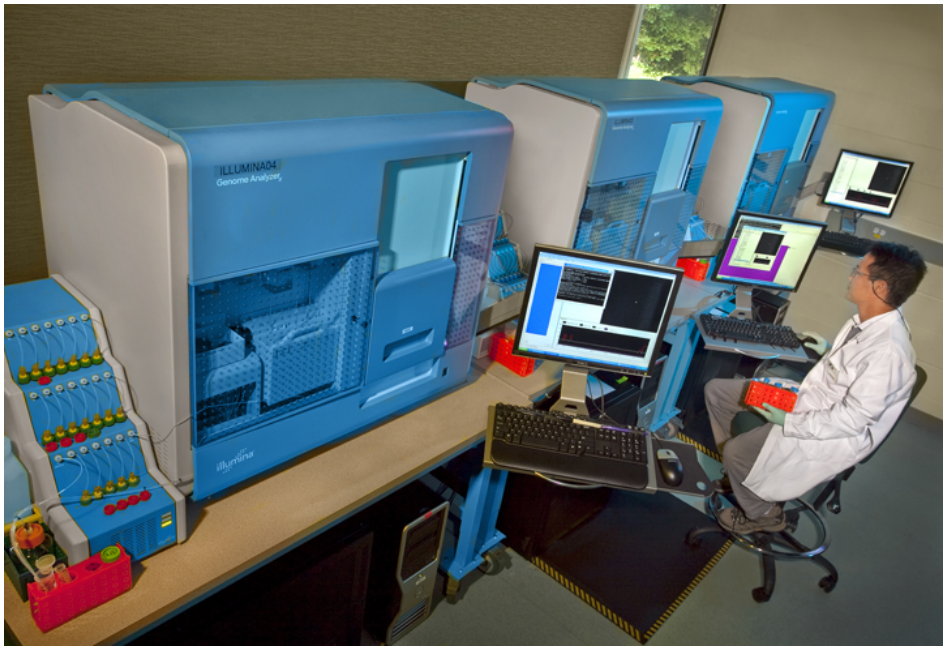
Normal vs cancer cells



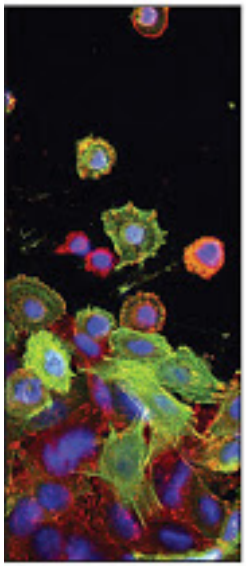


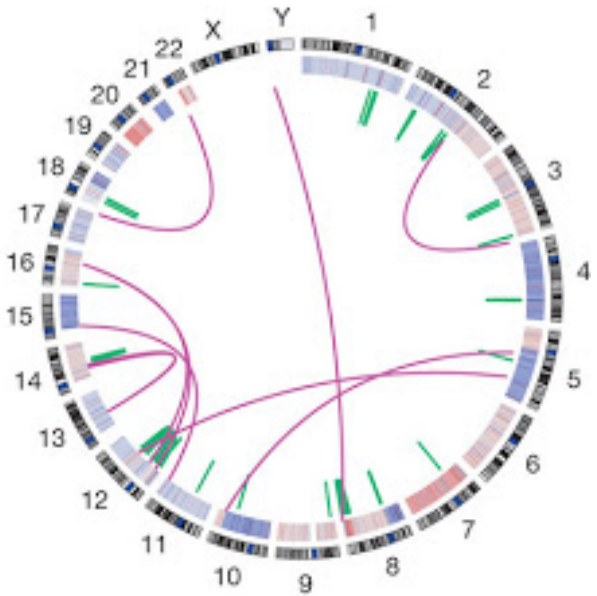
What goes wrong?
How to treat?

Biology is now quantitative, "high-throughput"

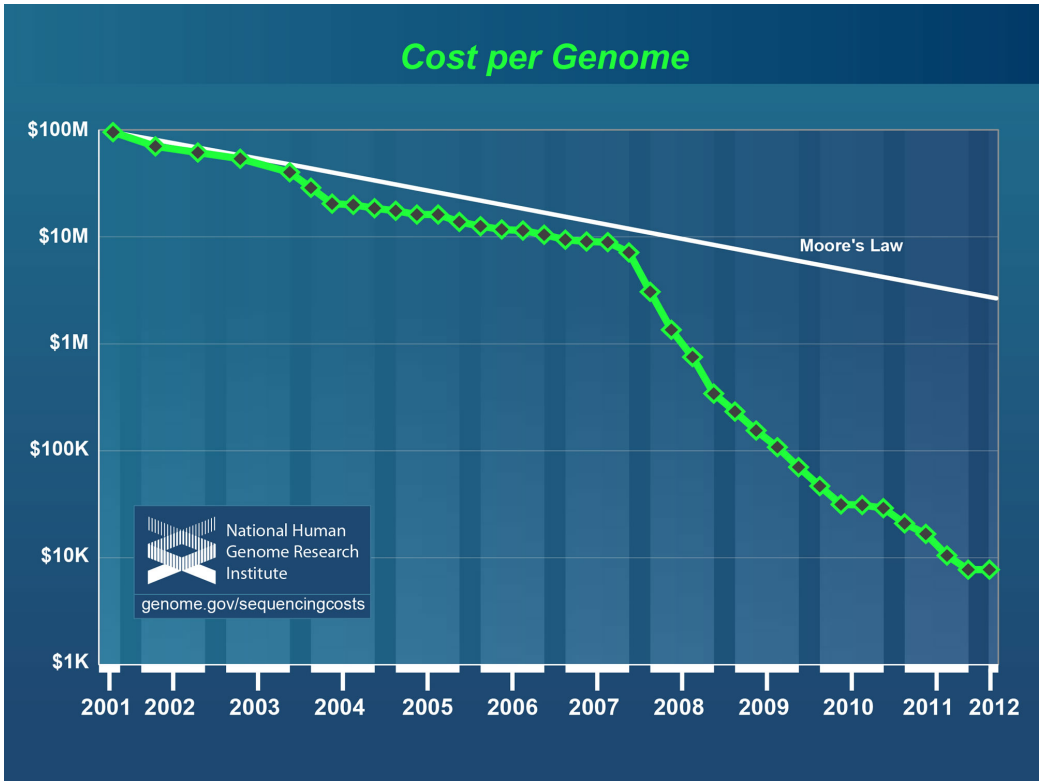


DOE Joint Genome Institute





Big data in biology



- "The $1,000 genome, the $1 million interpretation" (B. Kopf)
- High-dimensional, heterogeneous, structured data. "Large p"
- `http://aws.amazon.com/1000genomes/`

In this talk

$$\min_w R(w) + \lambda \Omega(w)$$

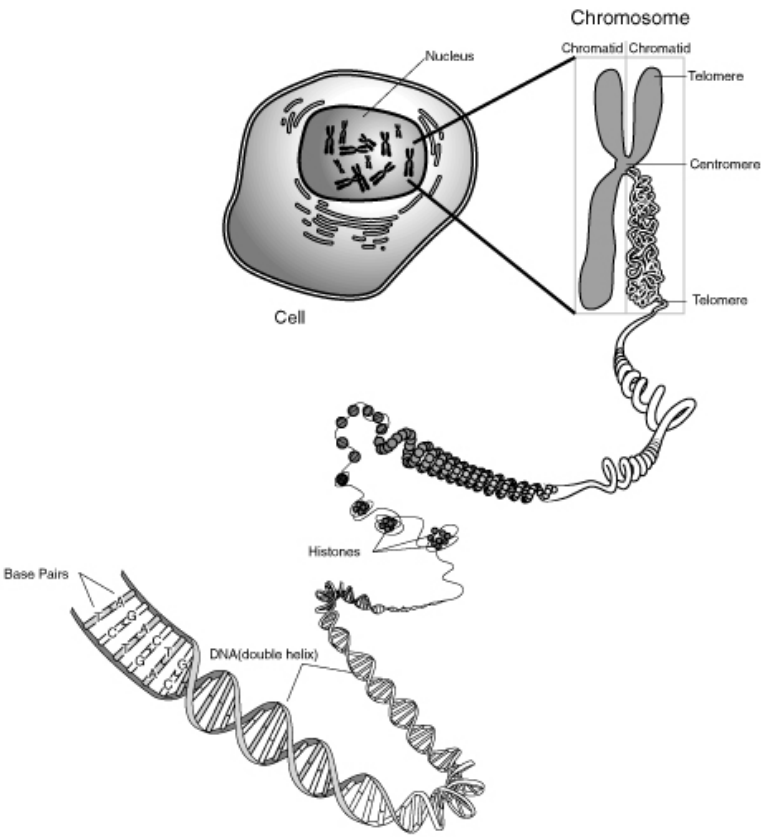
where:

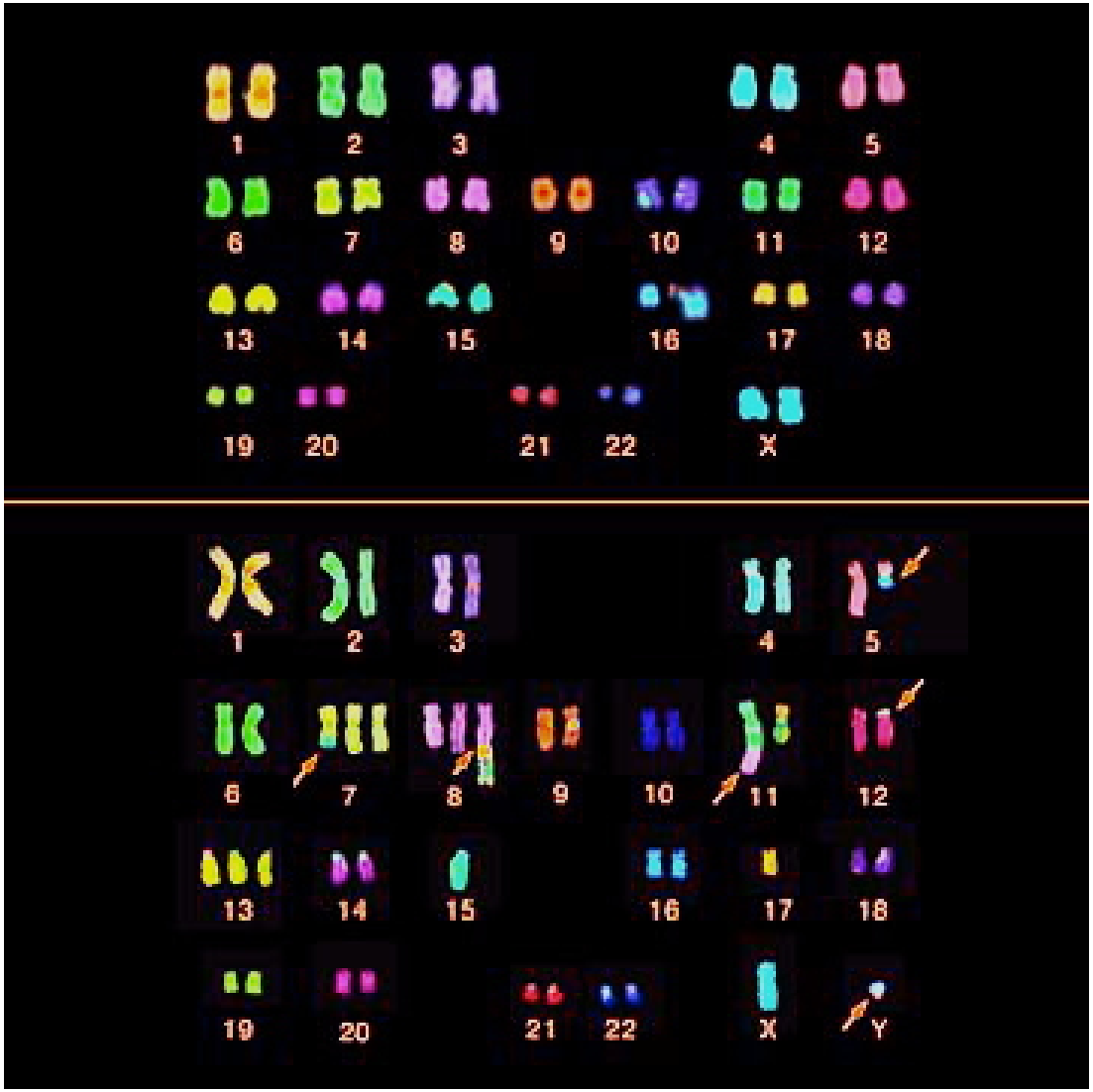
- w is the hypothesis we want to infer from data
- $R(w)$ is a smooth convex "fitness" function
- $\Omega(w)$ is a non-smooth convex penalty, which favors particular solution

1. Mapping DNA breakpoints in cancer genomes
2. Isoform detection from RNA-seq data
3. Inference of gene regulatory networks

Outline

Chromosomic aberrations in cancer

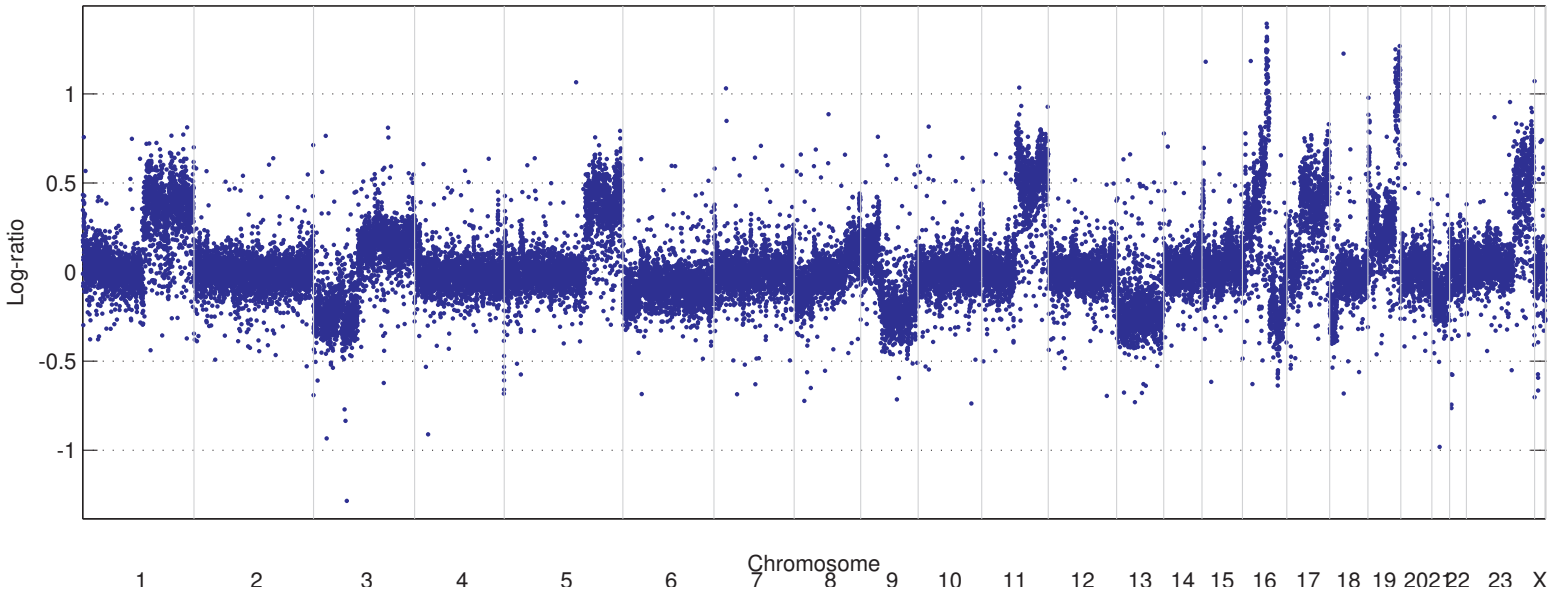




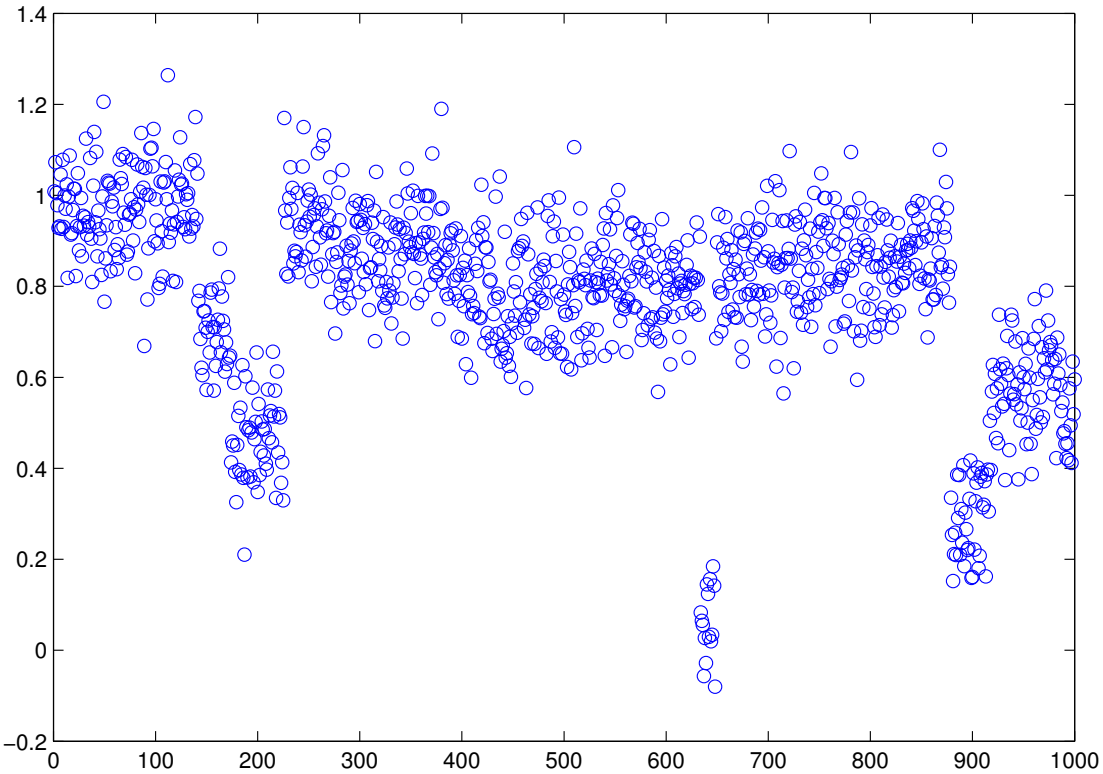
Comparative Genomic Hybridization (CGH)

Motivation

- Comparative genomic hybridization (CGH) data measure the DNA copy number along the genome
- Very useful, in particular in cancer research to observe systematically variants in DNA content

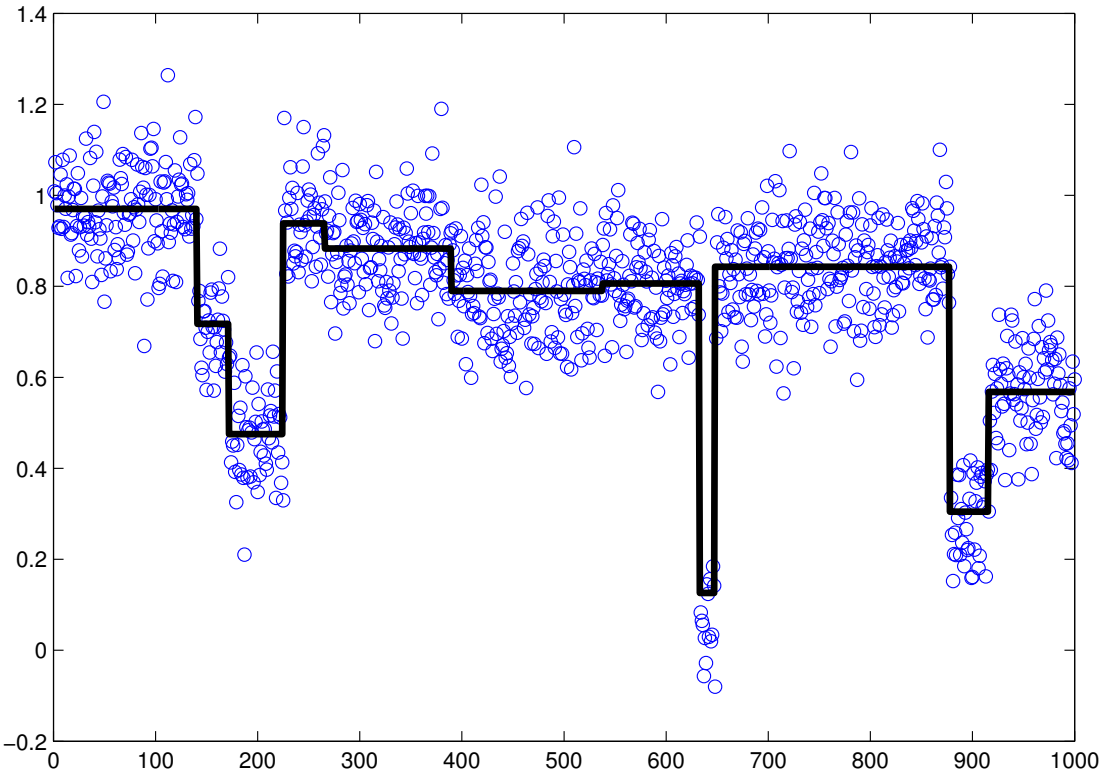


Can we identify breakpoints and "smooth" each profile?



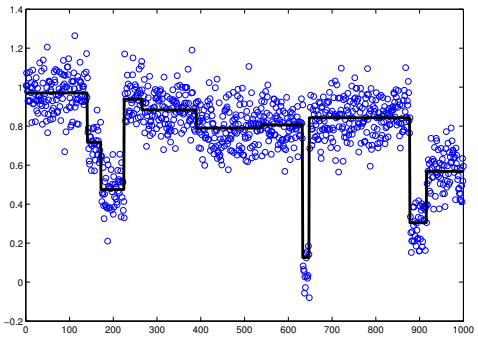
- A classical multiple change-point detection problem
- Should scale to lengths of order $10^6 \sim 10^9$

Can we identify breakpoints and "smooth" each profile?



- A classical multiple change-point detection problem
- Should scale to lengths of order $10^6 \sim 10^9$

An optimal solution

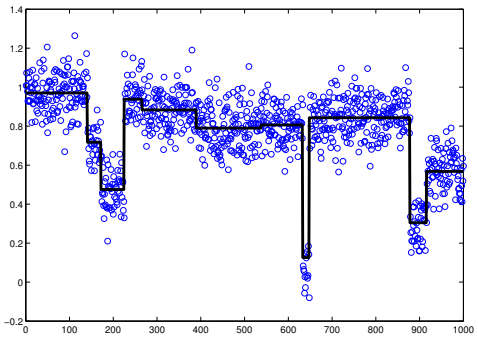


- For a signal $Y \in \mathbb{R}^p$, define an optimal approximation $\beta \in \mathbb{R}^p$ with k breakpoints as the solution of

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

- 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

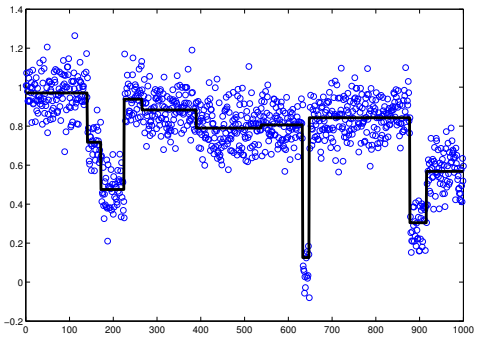


- For a signal $Y \in \mathbb{R}^p$, define an optimal approximation $\beta \in \mathbb{R}^p$ with k breakpoints as the solution of

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

- 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

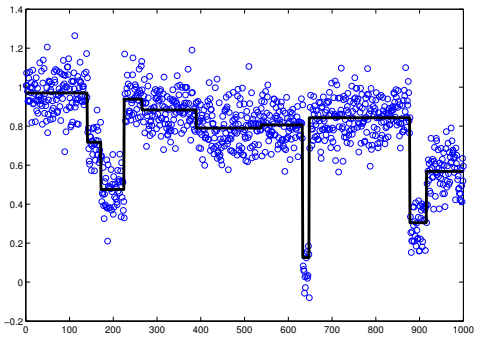


- For a signal $Y \in \mathbb{R}^p$, define an optimal approximation $\beta \in \mathbb{R}^p$ with k breakpoints as the solution of

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

- 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



- For a signal $Y \in \mathbb{R}^p$, define an optimal approximation $\beta \in \mathbb{R}^p$ with k breakpoints as the solution of

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

- 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

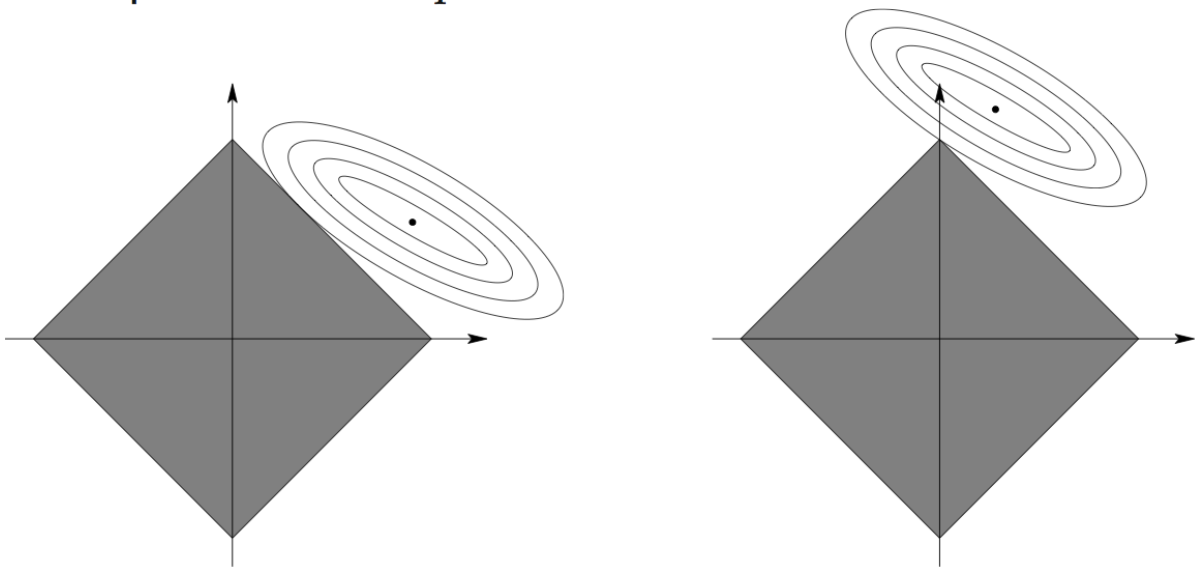
The ℓ_1 penalty (Tibshirani, 1996; Chen et al., 1998)

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 $p = 2$



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:

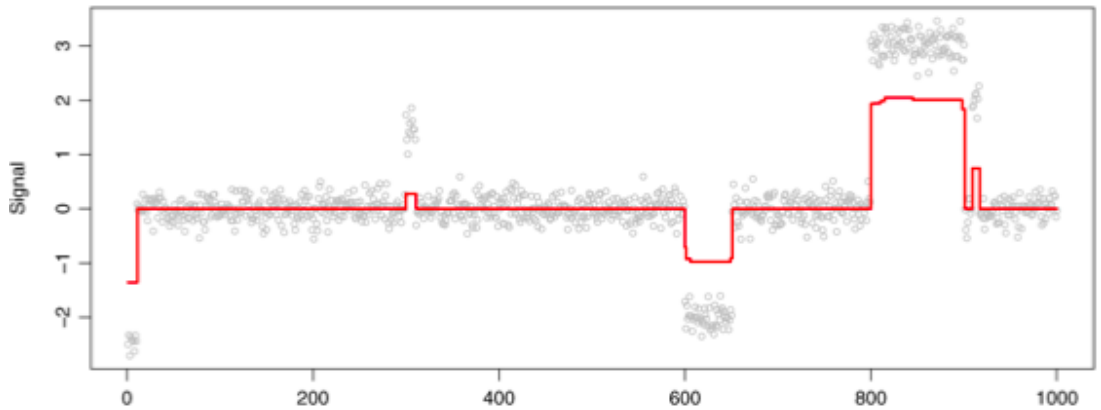
- Change of variable $u_i = \beta_{i+1} - \beta_i$, $u_0 = \beta_1$
- We obtain a Lasso problem in $u \in \mathbb{R}^{p-1}$
- u sparse means β piecewise constant

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)
- 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)

Solving TV signal approximator in $O(p \ln k)$

Theorem (V. and Bleakley, 2010; see also Hoefling, 2009)

TV signal approximator performs "greedy" dichotomic segmentation

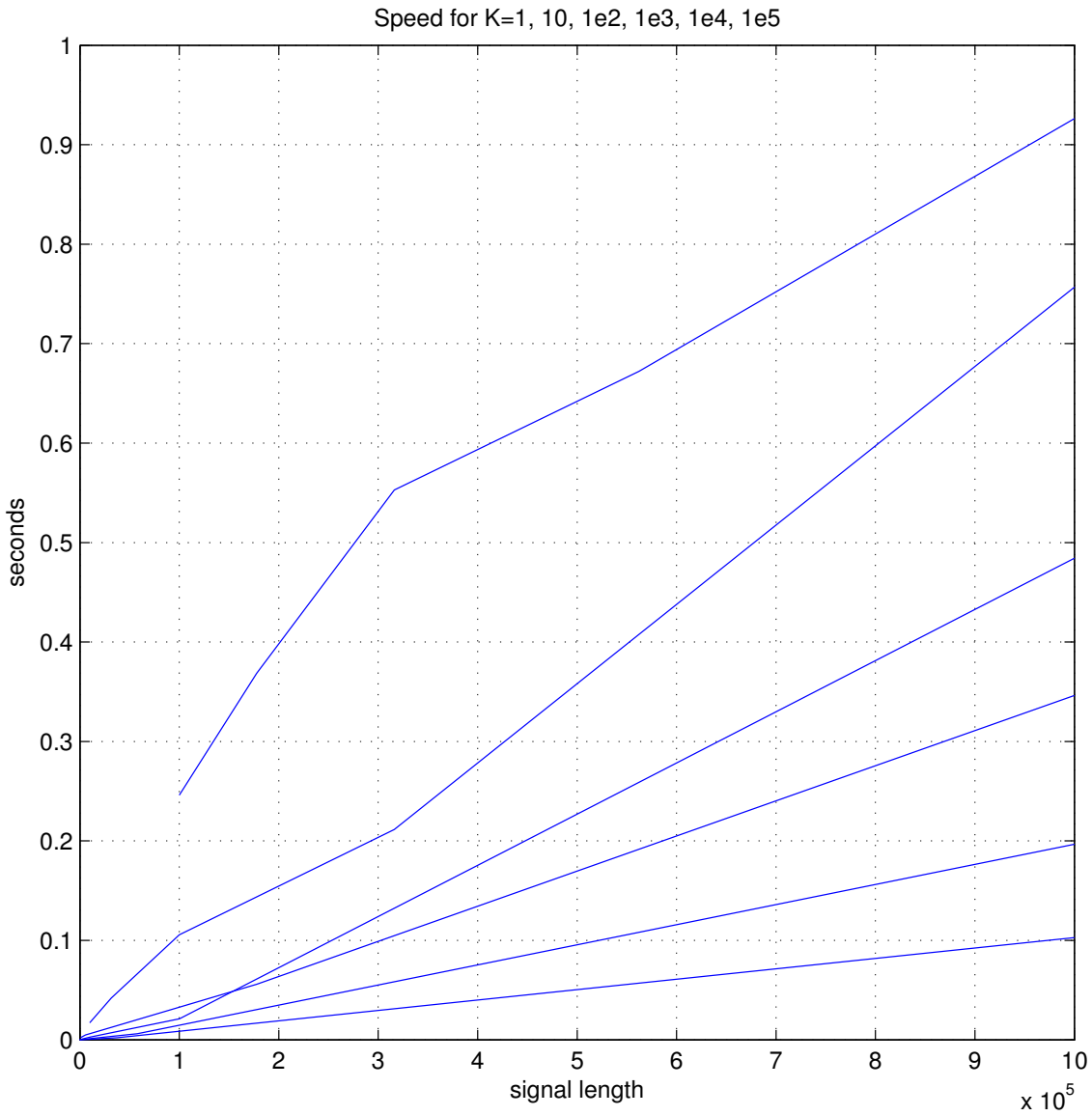
Algorithm 1 Greedy dichotomic segmentation

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

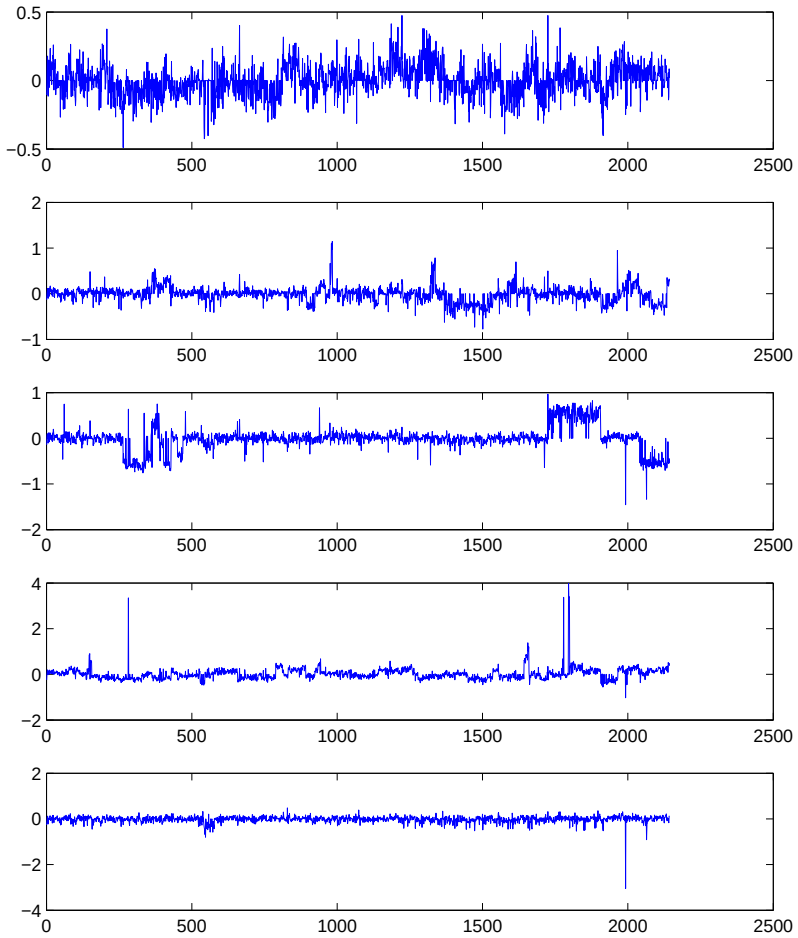
I_0 represents the interval $[1, n]$
$\mathcal{P} = \{I_0\}$
for $i = 1$ to k **do**
  $I^* \leftarrow \arg\max_{I \in \mathcal{P}} \gamma(I^*)$
  $\mathcal{P} \leftarrow \mathcal{P} \setminus \{I^*\}$
  $\mathcal{P} \leftarrow \mathcal{P} \cup \{I_L(I^*), I_R(I^*)\}$
end for
return $\mathcal{P}$

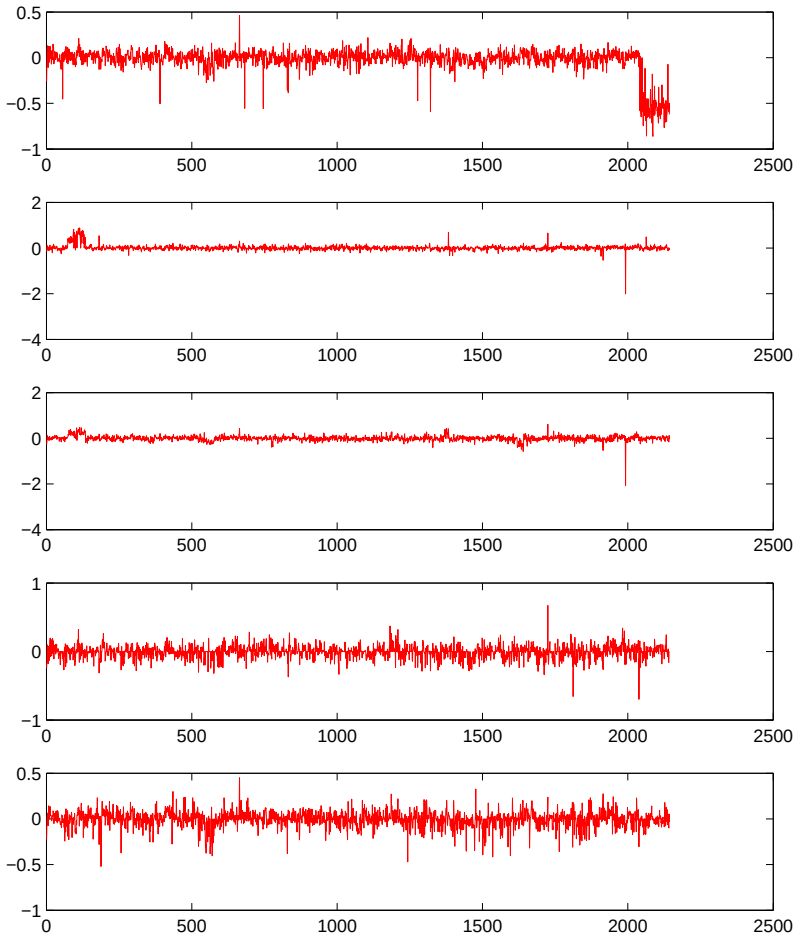
Apparently greedy algorithm finds the global optimum!

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



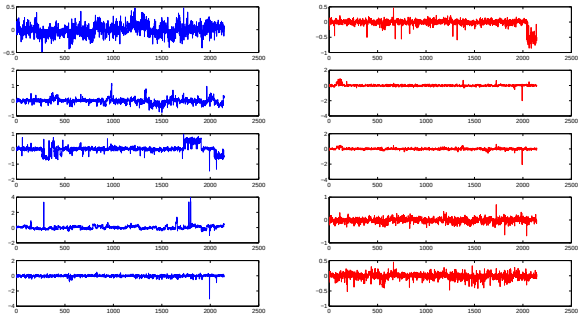
Extension 1: linear discrimination / regression





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

Fused lasso for supervised classification

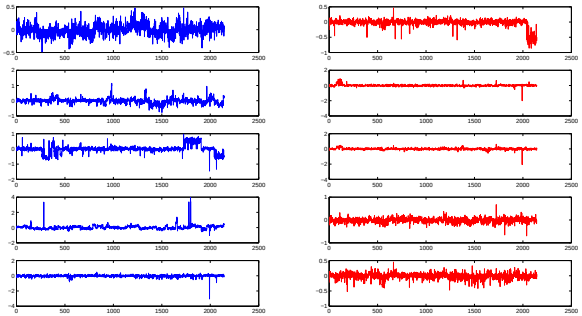


- Idea: find a linear predictor $f(Y) = \beta^\top Y$ that best discriminates the aggressive vs non-aggressive samples, subject to the constraints that it should be sparse and piecewise constant
- Mathematically:

$$\min_{\beta \in \mathbb{R}^p} R(\beta) + \lambda_1 \| \beta \|_1 + \lambda_2 \| \beta \|_{TV}$$

- Computationnally: proximal methods

Fused lasso for supervised classification

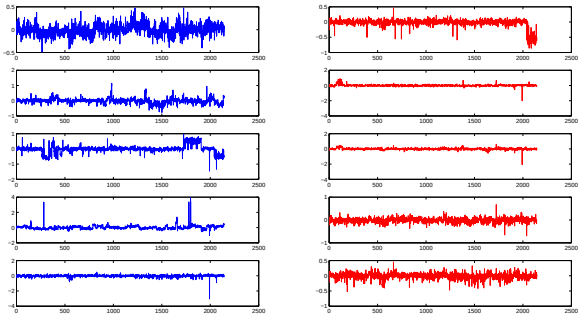


- Idea: find a linear predictor $f(Y) = \beta^\top Y$ that best discriminates the aggressive vs non-aggressive samples, subject to the constraints that it should be sparse and piecewise constant
- Mathematically:

$$\min_{\beta \in \mathbb{R}^p} R(\beta) + \lambda_1 \| \beta \|_1 + \lambda_2 \| \beta \|_{TV}$$

- Computationnally: proximal methods

Fused lasso for supervised classification

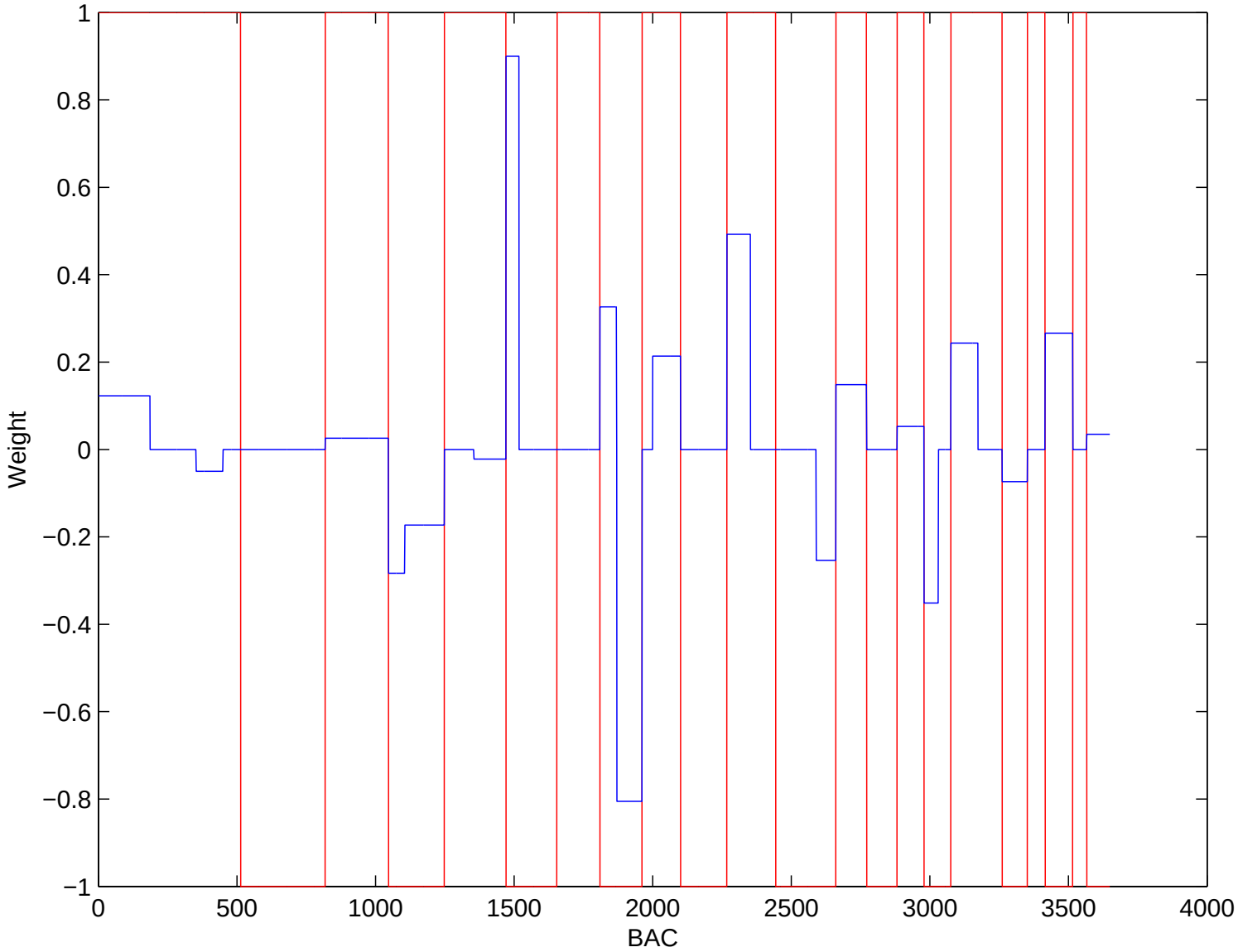


- Idea: find a linear predictor $f(Y) = \beta^\top Y$ that best discriminates the aggressive vs non-aggressive samples, subject to the constraints that it should be sparse and piecewise constant
- Mathematically:

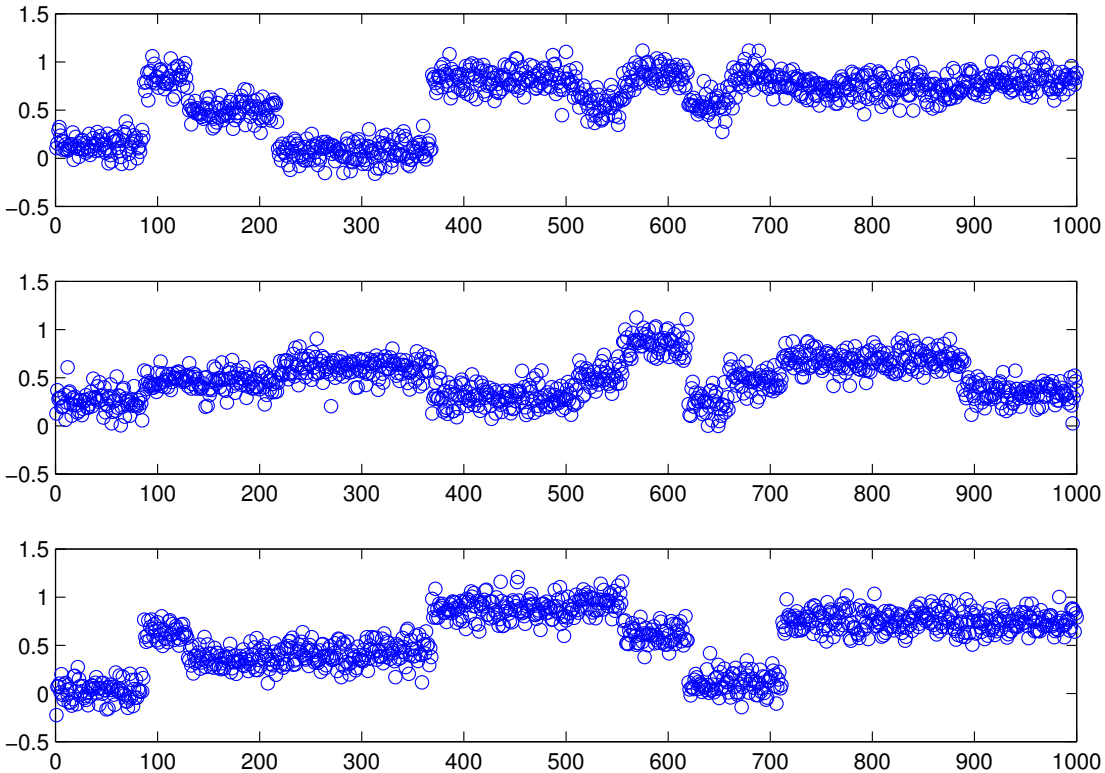
$$\min_{\beta \in \mathbb{R}^p} R(\beta) + \lambda_1 \| \beta \|_1 + \lambda_2 \| \beta \|_{TV}$$

- Computationnally: proximal methods

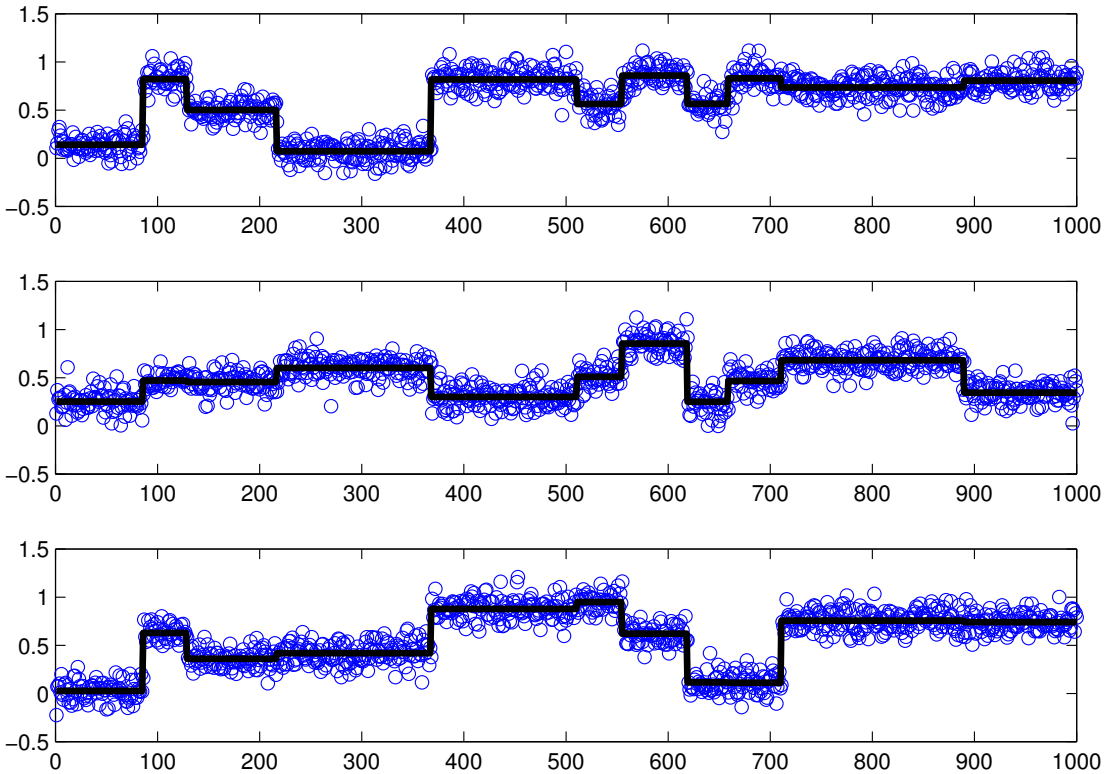
Prognosis in melanoma (Rapaport et al., 2008)



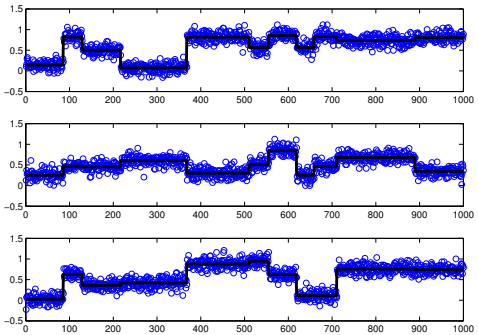
Extension 2: finding multiple change points shared by several profiles



Extension 2: finding multiple change points shared by several profiles



"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

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

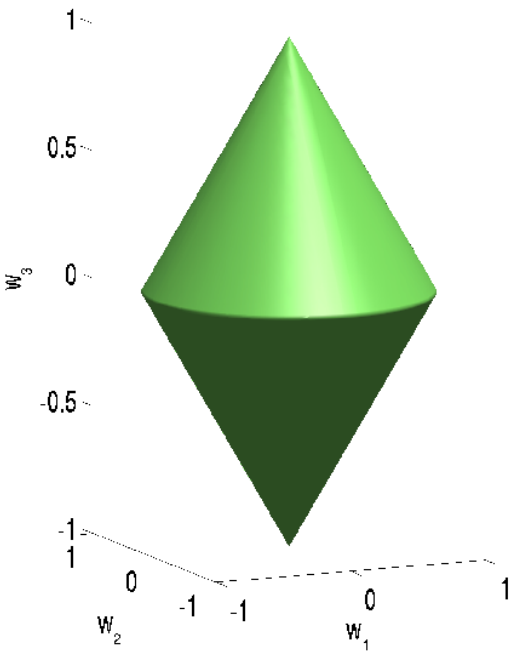
- DP finds the solution in $O(p^2 kn)$ in time and $O(p^2)$ in memory
- But: does not scale to $p = 10^6 \sim 10^9$...

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*:

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



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

GFLseg (Bleakley and V., 2011)

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

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

GFLseg = Group Fused Lasso segmentation

Questions

- Practice: can we solve it efficiently?
- Theory: does it recover the correct segmentation?

GFLseg (Bleakley and V., 2011)

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

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

GFLseg = Group Fused Lasso segmentation

Questions

- Practice: can we solve it efficiently?
- Theory: does it recover the correct segmentation?

GFLseg 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... (from Zaid)

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

Speed trial

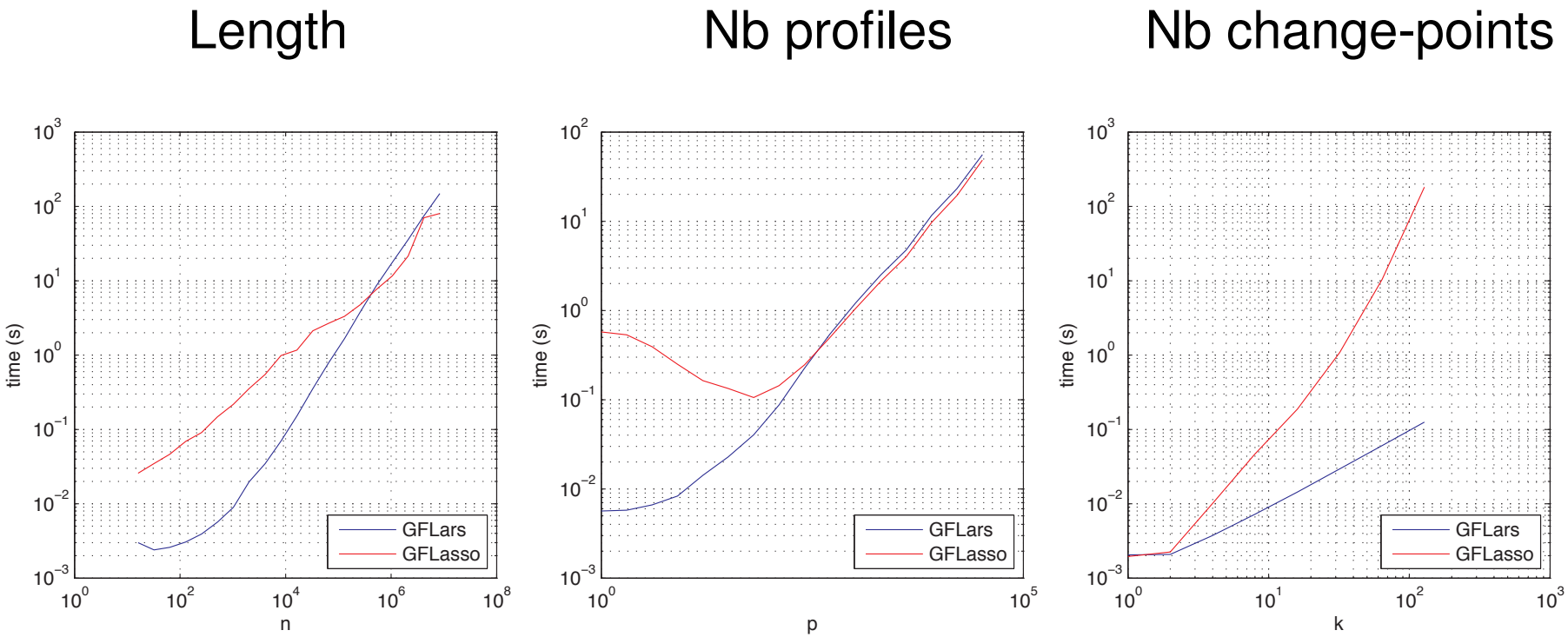
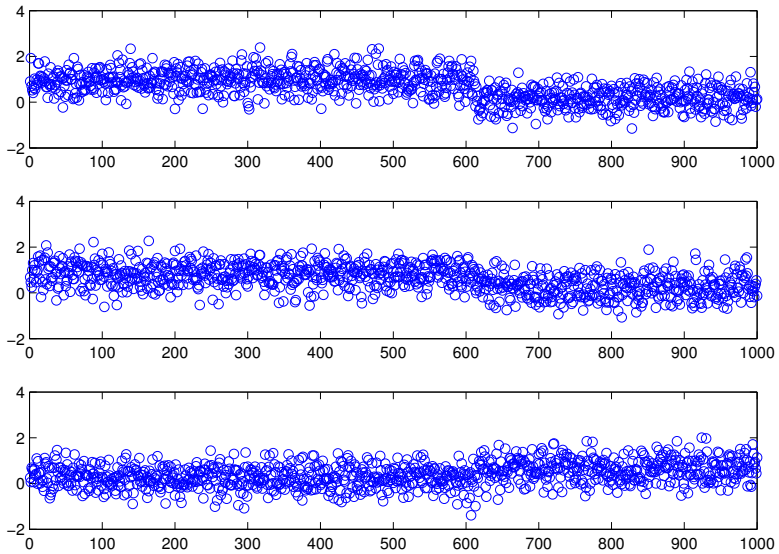


Figure 2: **Speed trials for group fused LARS (top row) and Lasso (bottom row).** *Left column:* varying n, with fixed $p = 10$ and $k = 10$; *center column:* varying p, with fixed $n = 1000$ and $k = 10$; *right column:* varying k, with fixed $n = 1000$ and $p = 10$. Figure axes are log-log. Results are averaged over 100 trials.

Consistency

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 $G_u = \max_i G_i$

Consistency for a single change-point

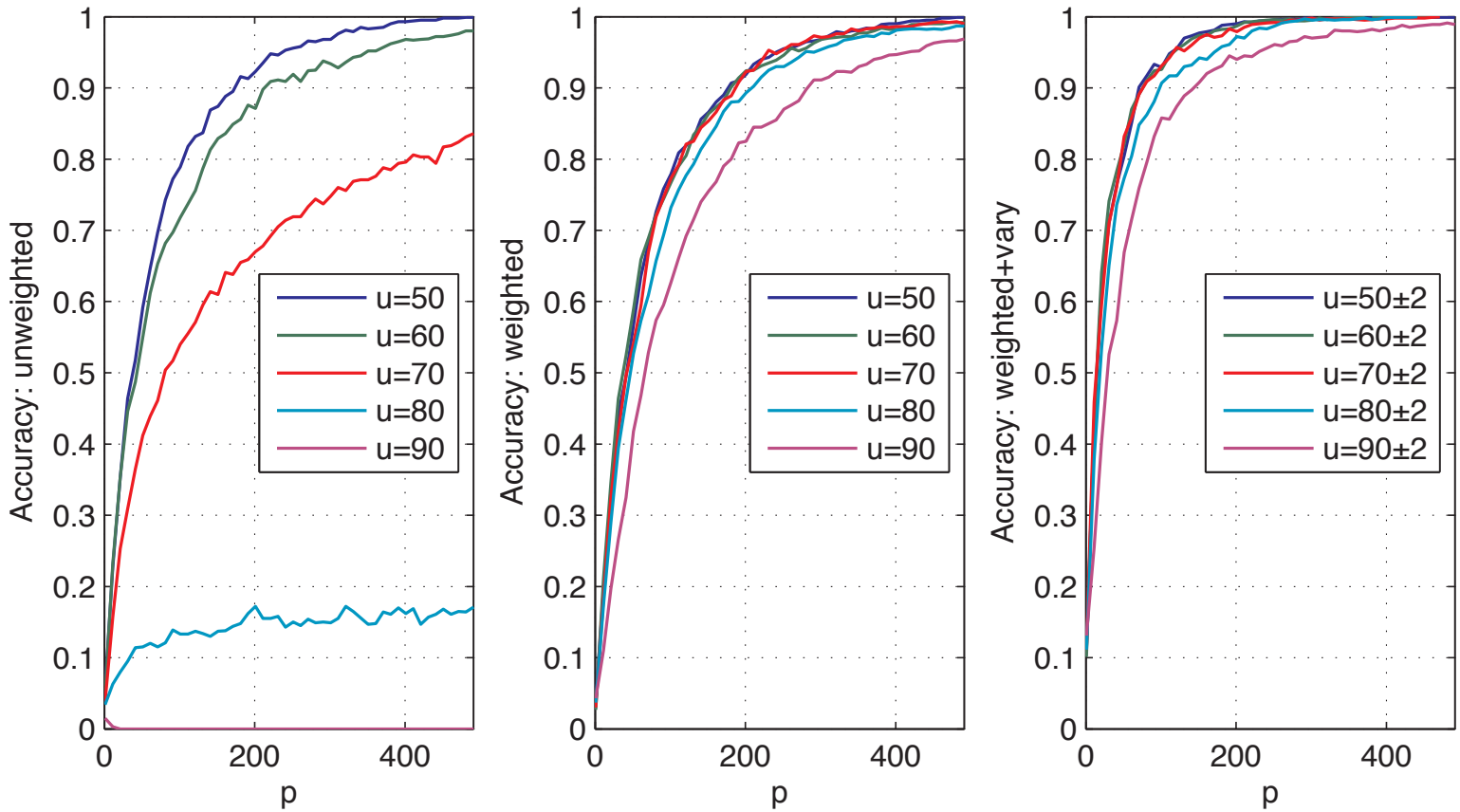


Figure 3: **Single change-point accuracy for the group fused Lasso.** Accuracy as a function of the number of profiles p when the change-point is placed in a variety of positions $u = 50$ to $u = 90$ (left and centre plots, resp. unweighted and weighted group fused Lasso), or: $u = 50 \pm 2$ to $u = 90 \pm 2$ (right plot, weighted with varying change-point location), for a signal of length 100.

Estimation of several change-points

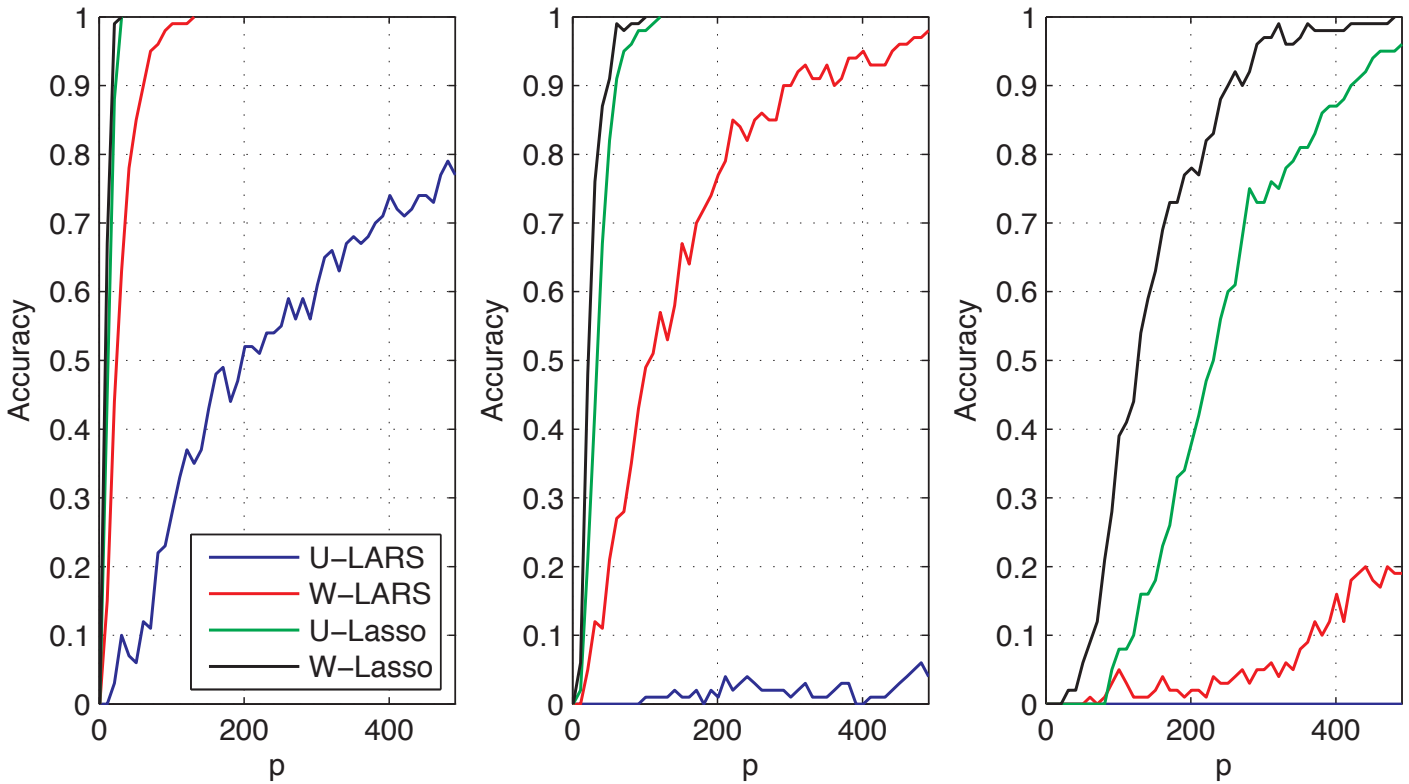
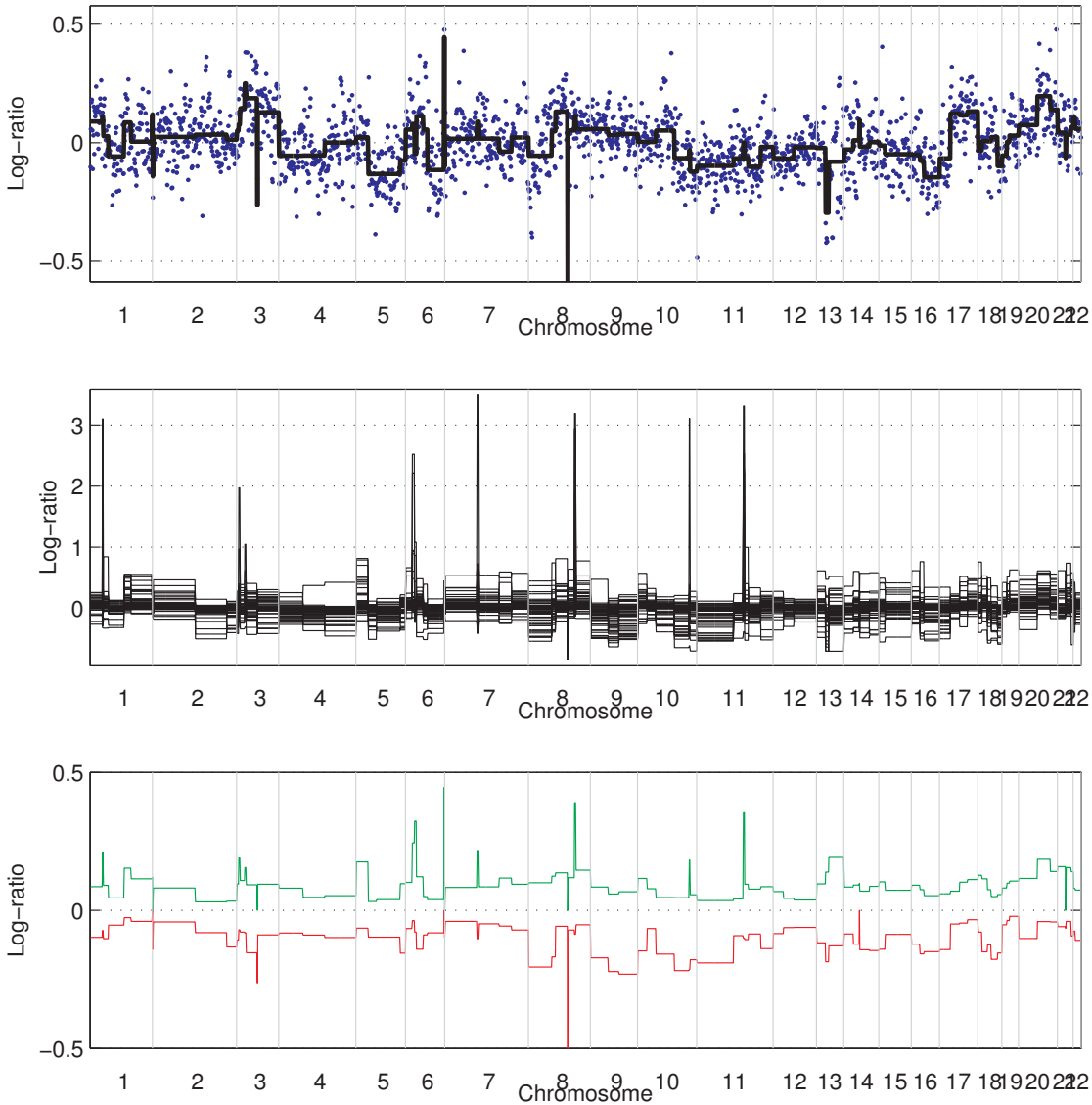


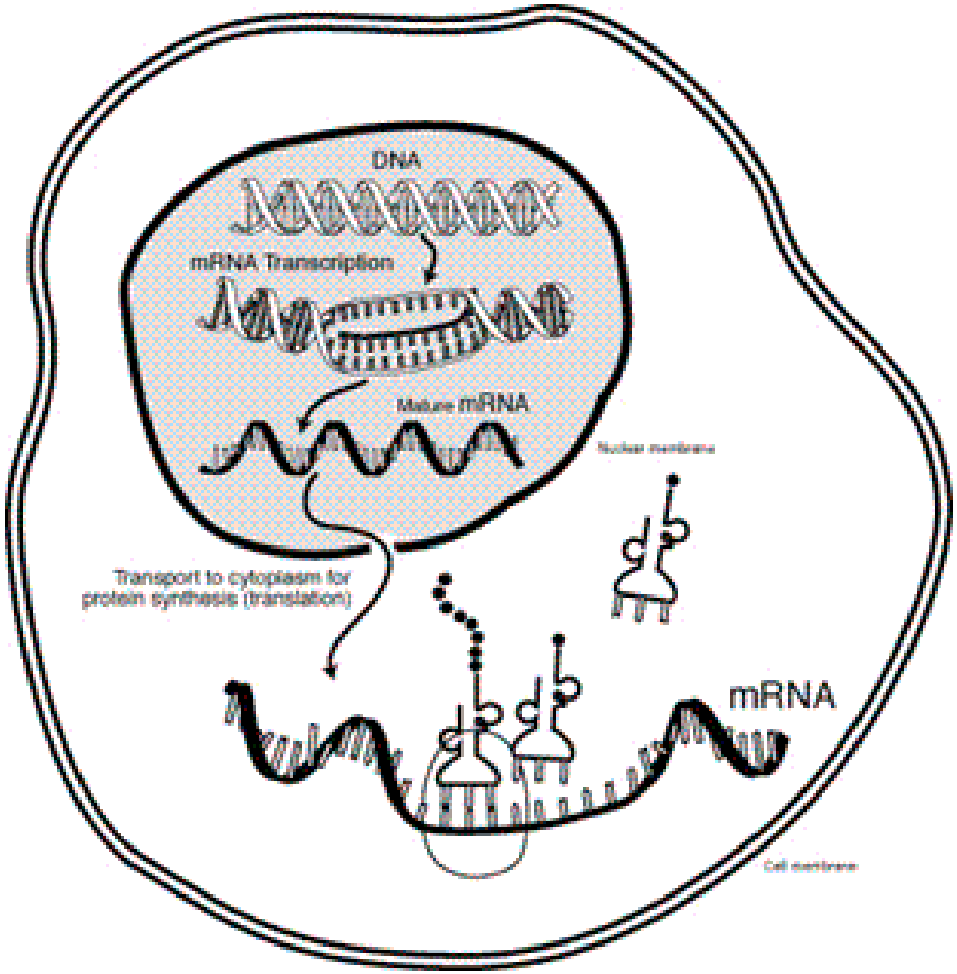
Figure 4: **Multiple change-point accuracy.** Accuracy as a function of the number of profiles p when change-points are placed at the nine positions $\{10, 20, \ldots, 90\}$ and the variance σ^2 of the centered Gaussian noise is either 0.05 (left), 0.2 (center) and 1 (right). The profile length is 100.

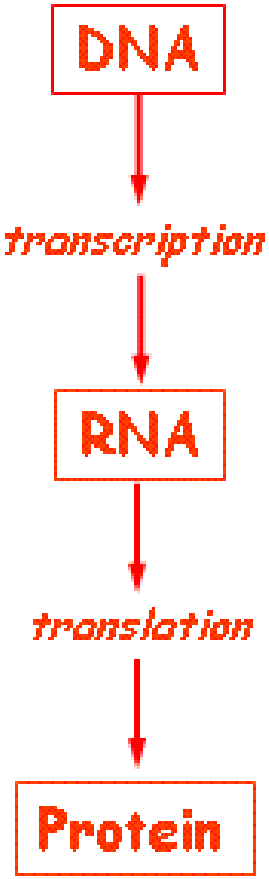
Application: detection of frequent abnormalities



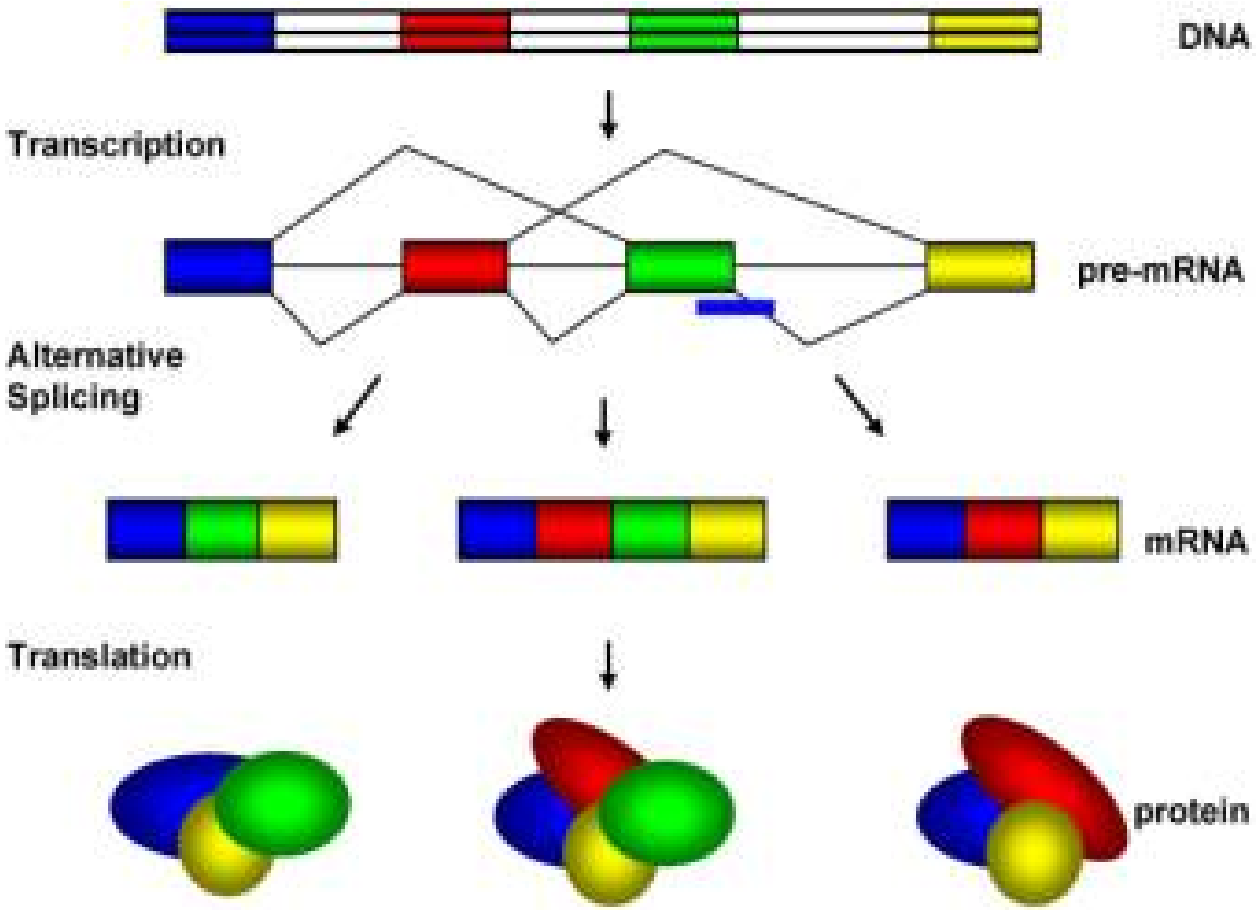
Outline

Central dogma

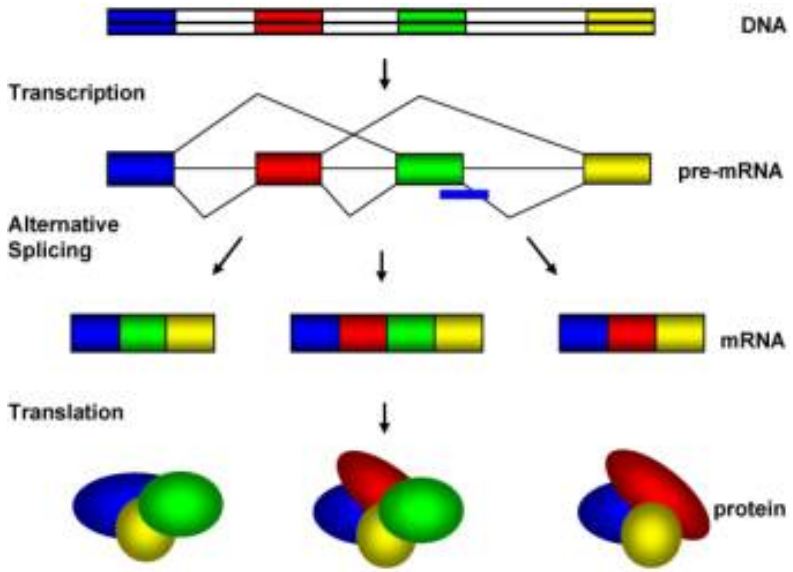




Alternative splicing: 1 gene = many proteins



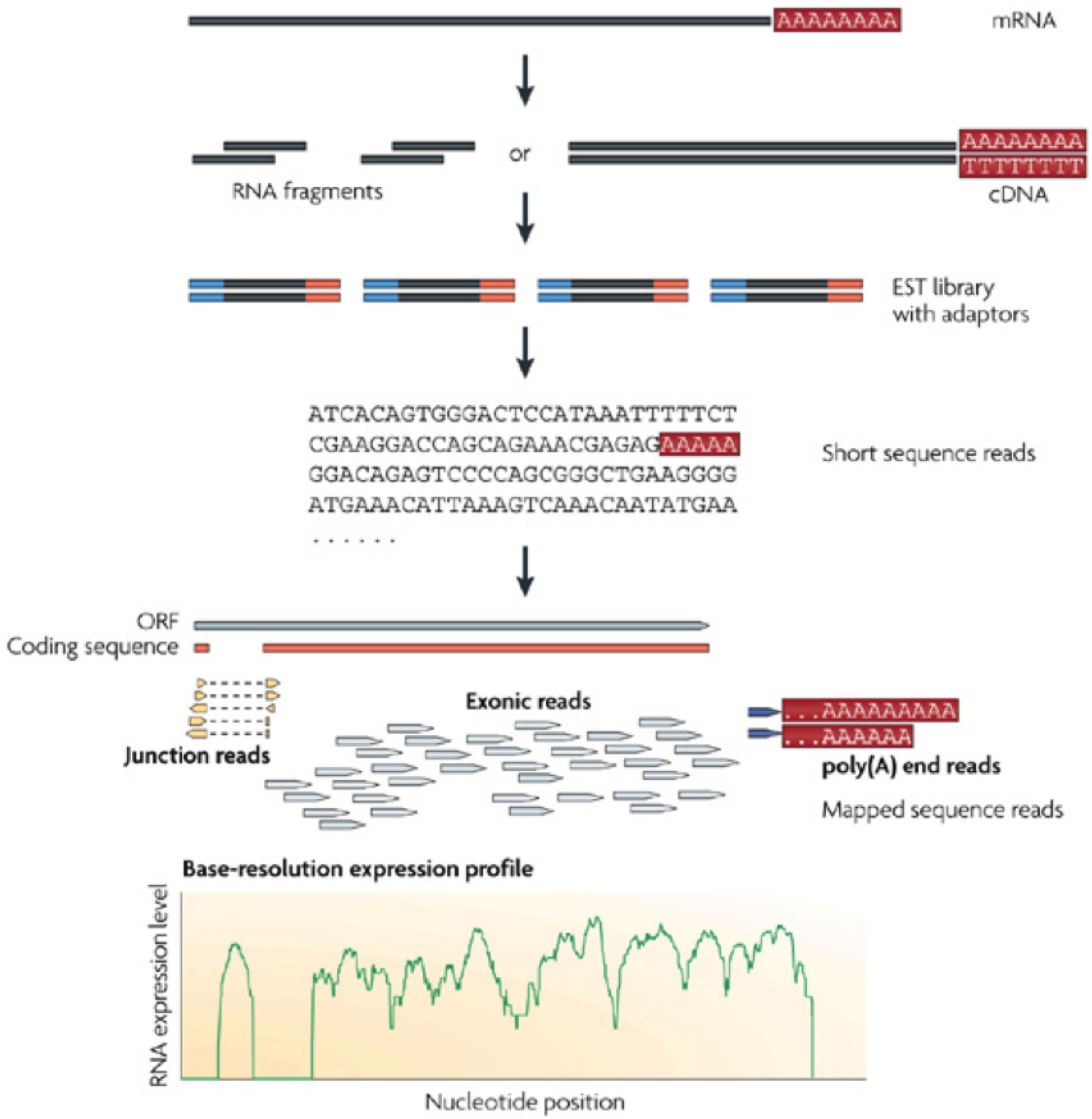
The isoform identification and quantification problem



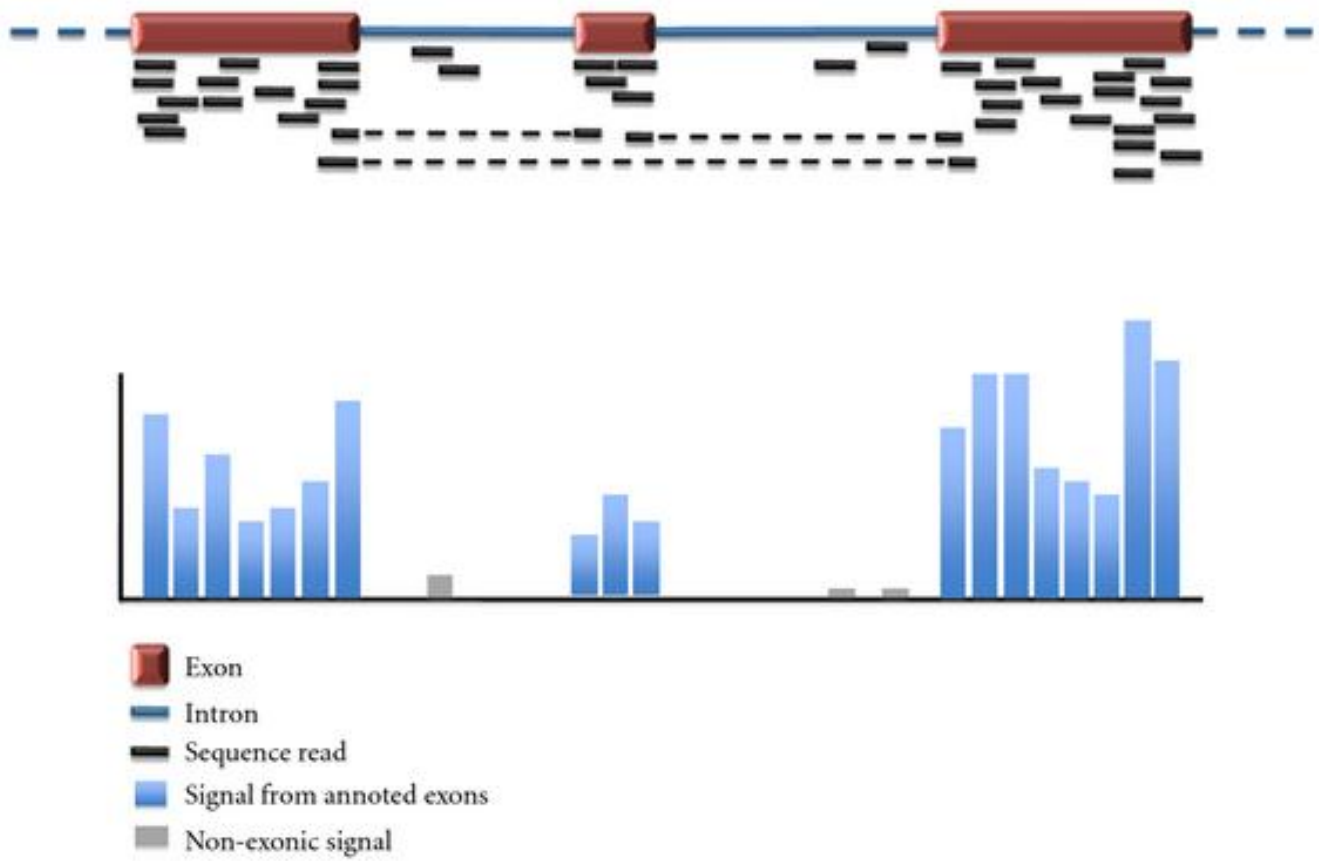
Given a biological sample (e.g., cancer tissue), can we:

1. identify the isoform(s) of each gene present in the sample?
2. quantify their abundance?

RNA-seq measures mRNA abundance by sequencing short fragments

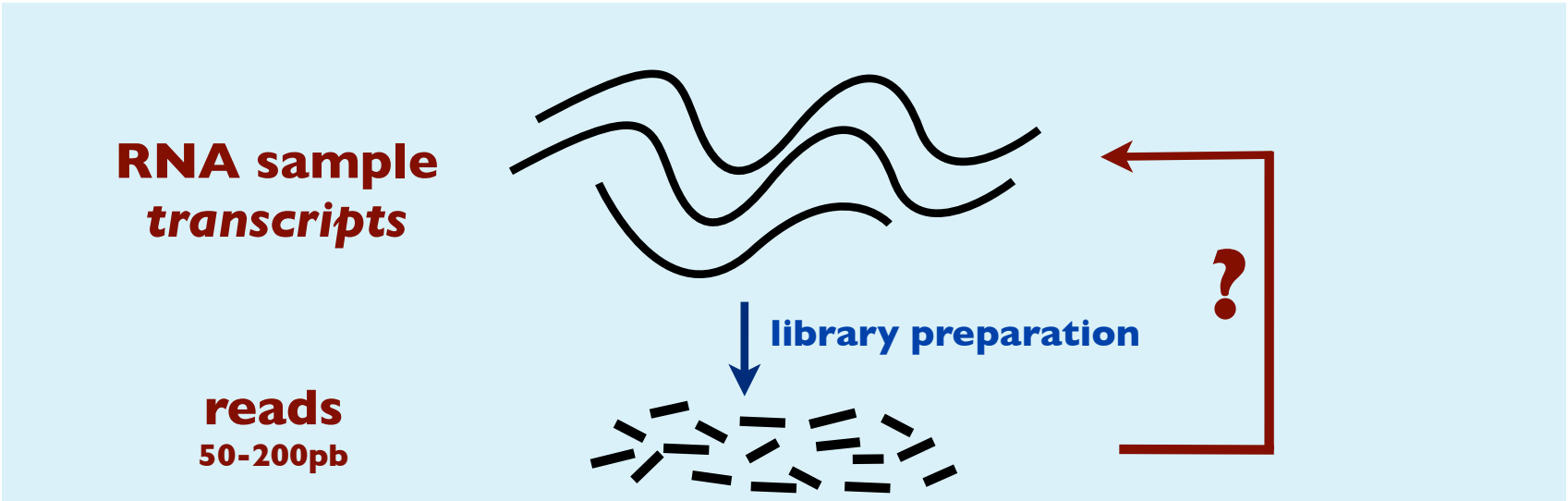


RNA-seq and alternative splicing



(Costa et al., 2011)

From RNA-seq to isoforms



Transcripts Quantification using annotations

- RQuant (Bohnert et al. 2009)
- FluxCapacitor (Montgomery et al. 2010)
- IsoEM (Nicolae et al. 2011)
- eXpress (Roberts et al. 2013)

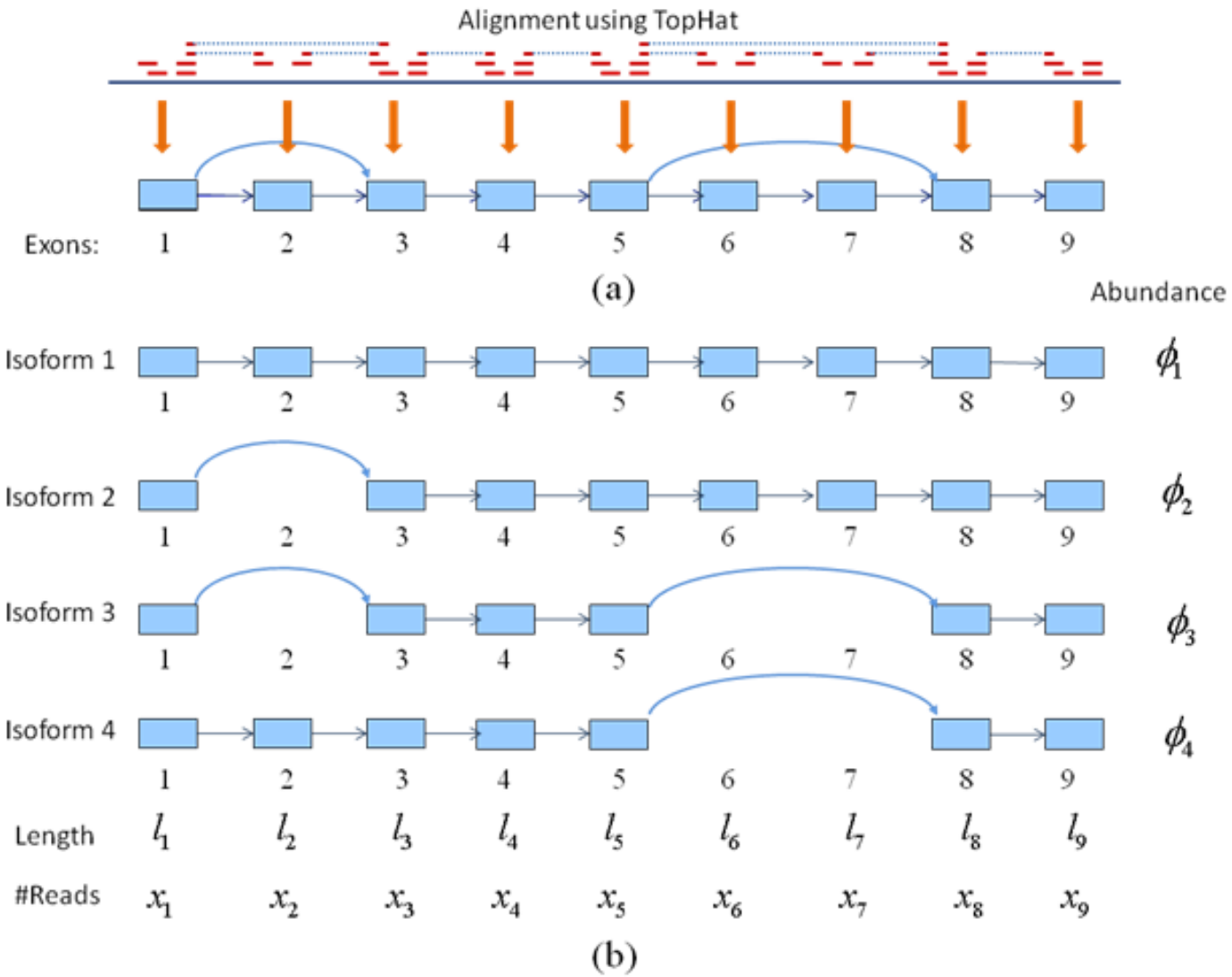
De Novo approaches

- OASES (Schultz et al. 2012)
- Trinity (Grabherr et al. 2011)
- Kissplice (Sacomoto et al. 2012)

Genome-based Transcripts Reconstruction

- Scripture (Guttman et al. 2010)
- Cufflinks (Trapnell et al. 2010)
- IsoLasso (Li et al. 2011a)
- NSMAP (Xia et al. 2011)
- SLIDE (Li et al. 2011b)
- iReckon (Mezlini et al. 2012)
- **FlipFlop**

The isoform deconvolution problem



(Xia et al., 2011)

More formally

e exons
c candidate isoforms (up to $2^e - 1$)
$\phi \in \mathbb{R}_+^c$ the vector of abundance of isoforms (unknown!)
U binary matrix:

$$
\begin{array}{c}
 \\
\mathit{isoform}_1 \\
\mathit{isoform}_2 \\
\vdots \\
\mathit{isoform}_c
\end{array}
\begin{array}{c}
\begin{array}{ccccccc}
\mathit{exon}_1 & \cdots & \mathit{exon}_e & \mathit{junction}_{1,2} & \cdots & \mathit{junction}_{e_1,e}
\end{array} \\
\left(\begin{array}{ccccccc}
1 & \cdots & 1 & 1 & \cdots & 1 \\
1 & \cdots & 0 & 1 & \cdots & 0 \\
 & \cdots & & & \cdots & \\
0 & \cdots & 1 & 0 & \cdots & 0
\end{array}\right)
\end{array}
$$

$U^\top \phi$ the abundance of each exon/junction.

Goal: estimate ϕ from the observed reads on each exon/junction

Isoform deconvolution with the Lasso

Estimate ϕ sparse by solving:

$$\min_{\phi \in \mathbb{R}_+^c} R(U^\top \phi) + \lambda \| \phi \|_1$$

- IsoLasso (Li et al., 2011)
- NSMAP (Xia et al., 2011)
- SLIDE (Li et al., 2011)

Works well BUT computationally challenging to enumerate all candidate isoforms (up to 2^e) for large genes!

Fast isoform deconvolution with the Lasso

Theorem (Bernard, Mairal, Jacob and V., 2012)

The isoform deconvolution problem

$$\min_{\phi \in \mathbb{R}_+^c} R(U^\top \phi) + \lambda \| \phi \|_1$$

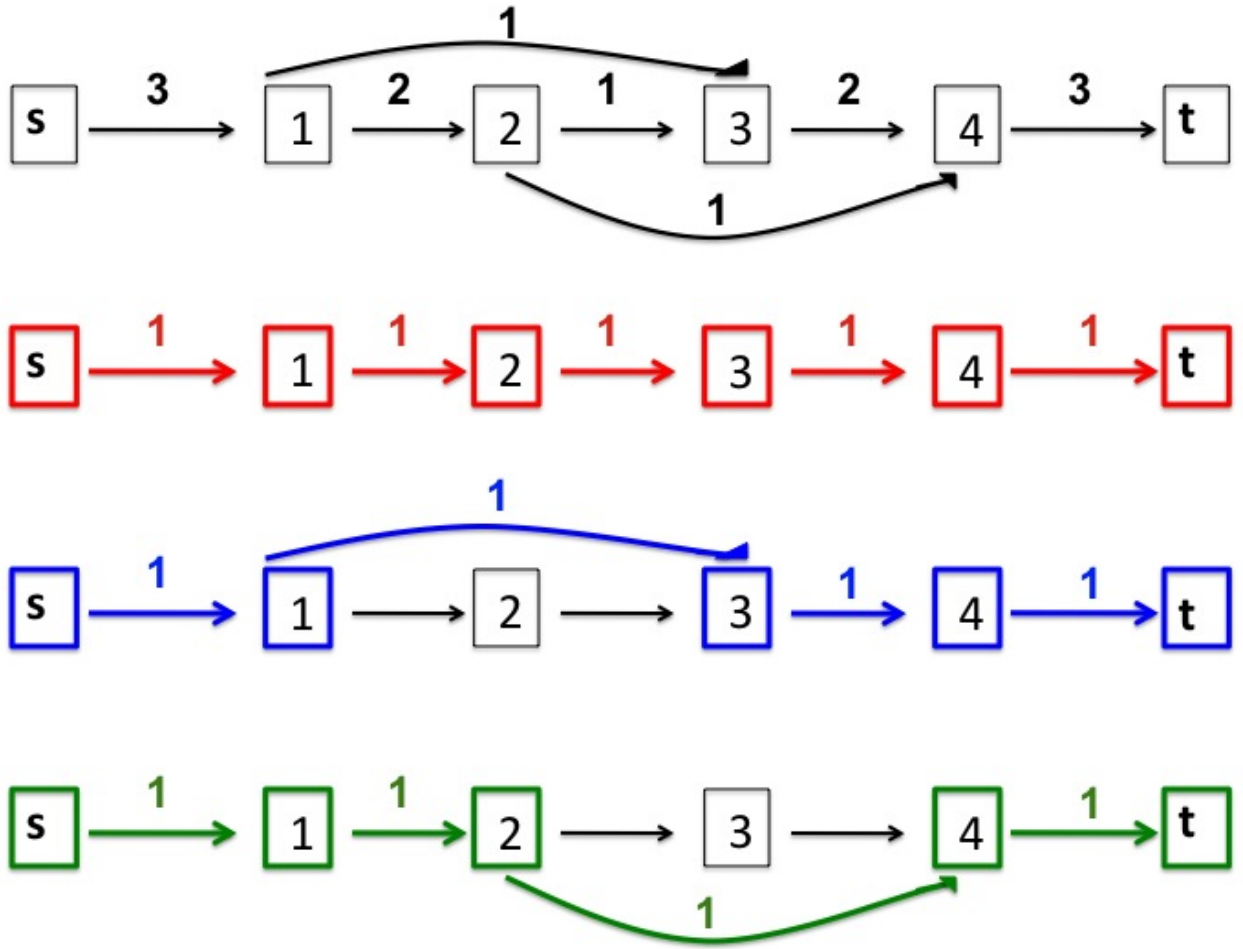
can be solved in polynomial time in the number of exon.

Key ideas

1. $U^\top \phi$ corresponds to a flow on the graph
2. Reformulation as a convex cost flow problem (Mairal and Yu, 2012)
3. Recover isoforms by flow decomposition algorithm

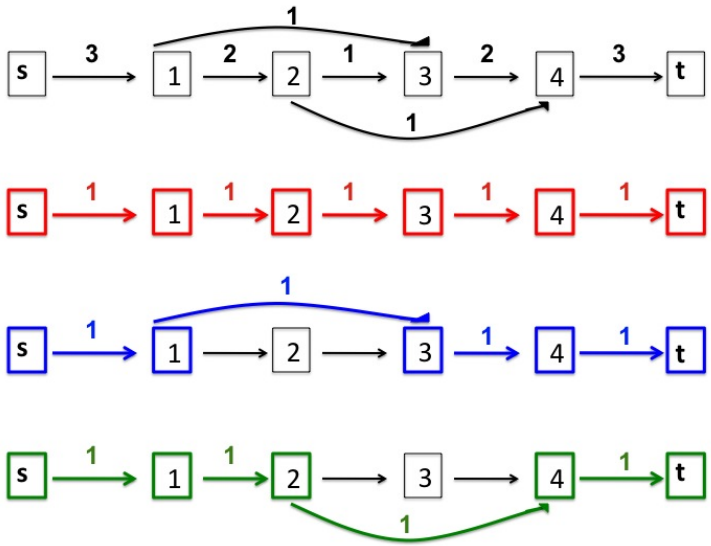
"Feature selection on an exponential number of features in polynomial time"

From isoforms to flows



- Isoforms are paths
- Linear combinations of isoforms are flows

Isoform deconvolution as convex cost flow problem



$$\min_{\phi \in \mathbb{R}_+^c} R(U^\top \phi) + \lambda \| \phi \|_1$$

is equivalent to

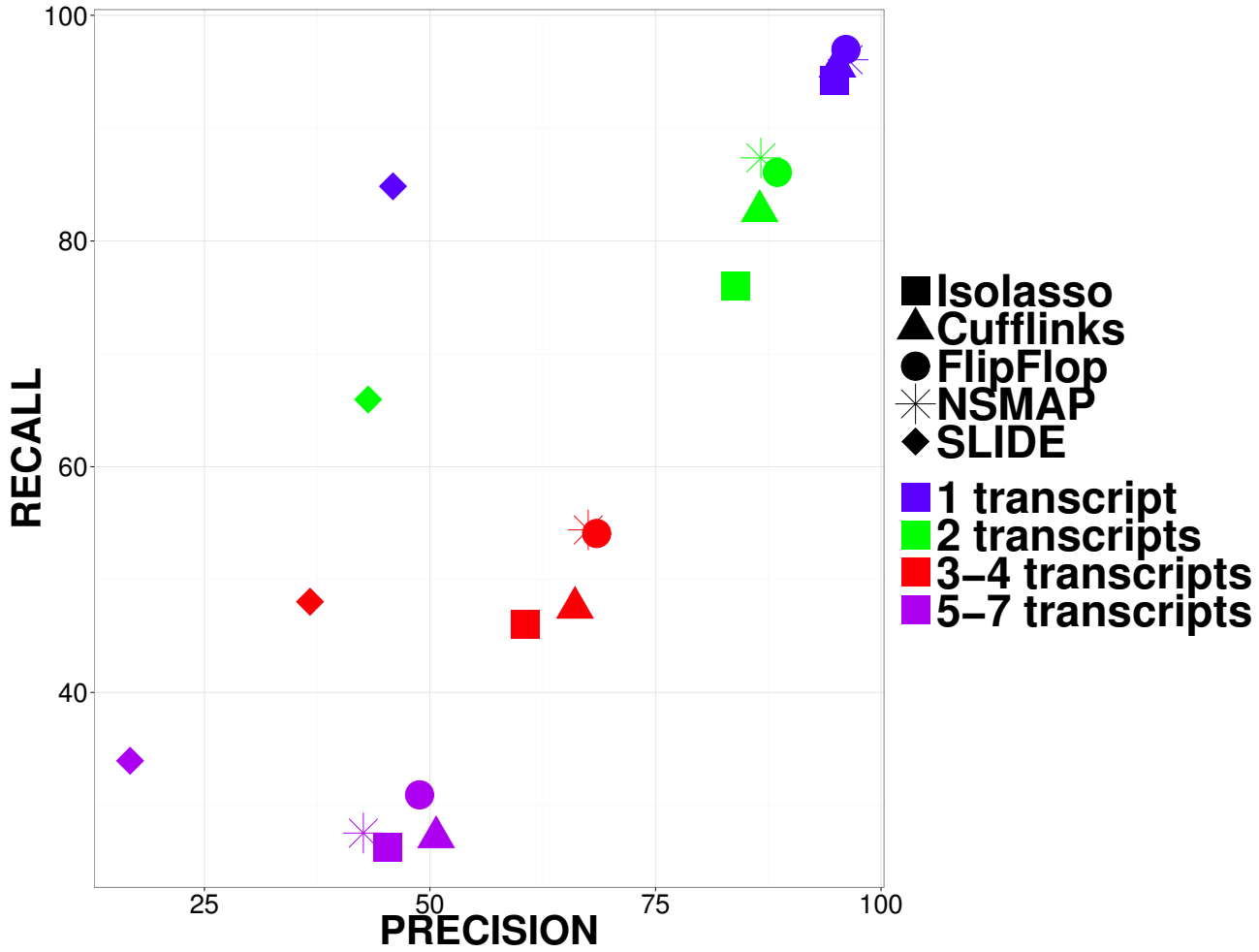
$$\min_{\text{f flow}} R(f) + \lambda f_t$$

FlipFlop (Bernard, Jacob, Mairal and V., 2012)

$$\min_{\phi \in \mathbb{R}_+^c} R(U^\top \phi) + \lambda \| \phi \|_1$$

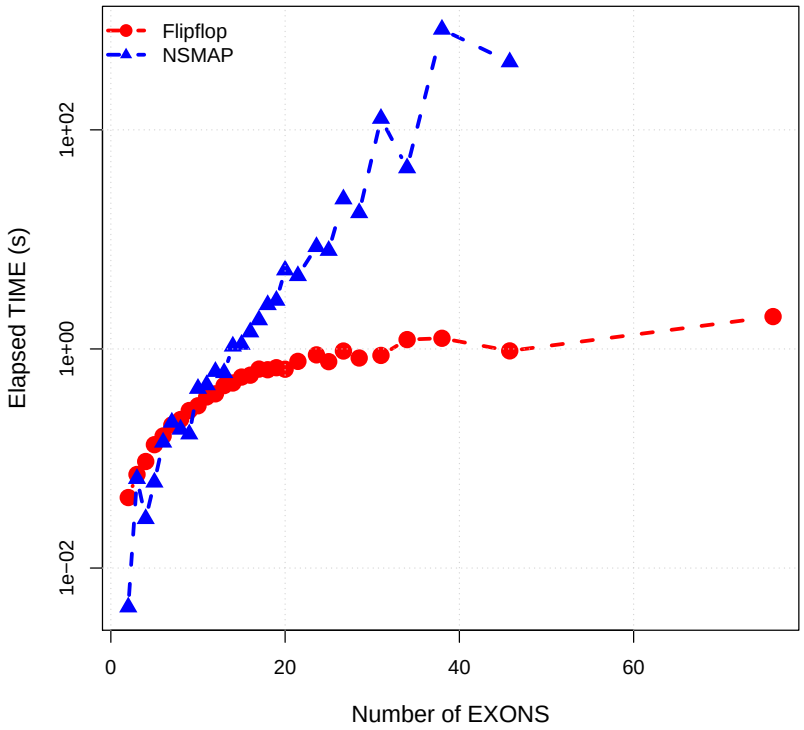
- Cufflink : *a priori* selection of isoforms (minimum graph cover)
- IsoLasso : pre-filtering of candidate isoforms using various heuristics
- NSMAP, SLIDE : limit the maximum number of exons
- FlipFlop : exact optimization without pre-filtering in polynomial time

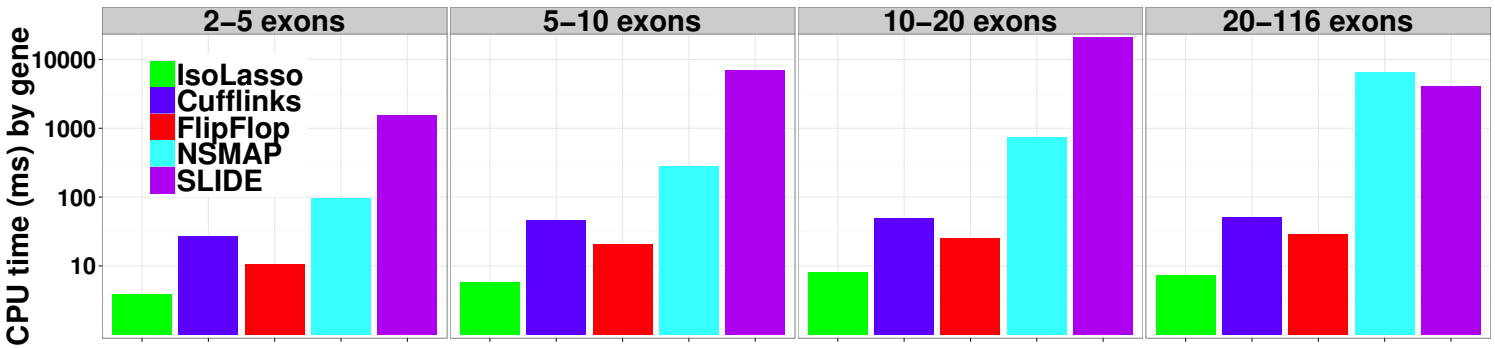
Performance in isoform identification



Simulated data (hg19, 1137 genes on chr1, 1million 75 bp single-end reads by transcript levels).

Speed trial


Flipflop
NSMAP
Elapsed TIME (s)
1e−02
1e+00
1e+02
0
20
40
60
Number of EXONS


2–5 exons
5–10 exons
10–20 exons
20–116 exons
CPU time (ms) by gene
10000
1000
100
10
IsoLasso
Cufflinks
FlipFlop
NSMAP
SLIDE

Outline

Gene expression

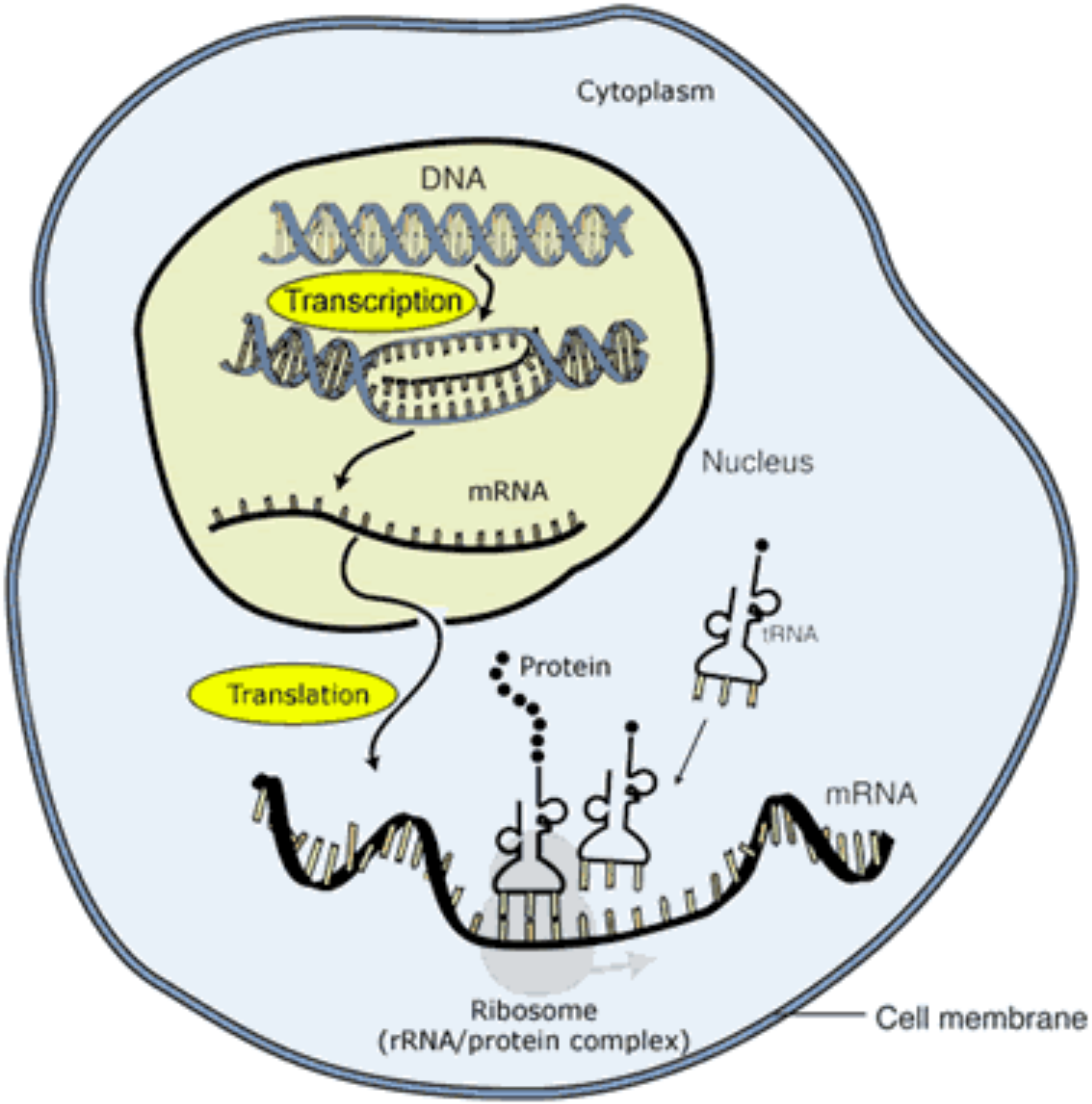
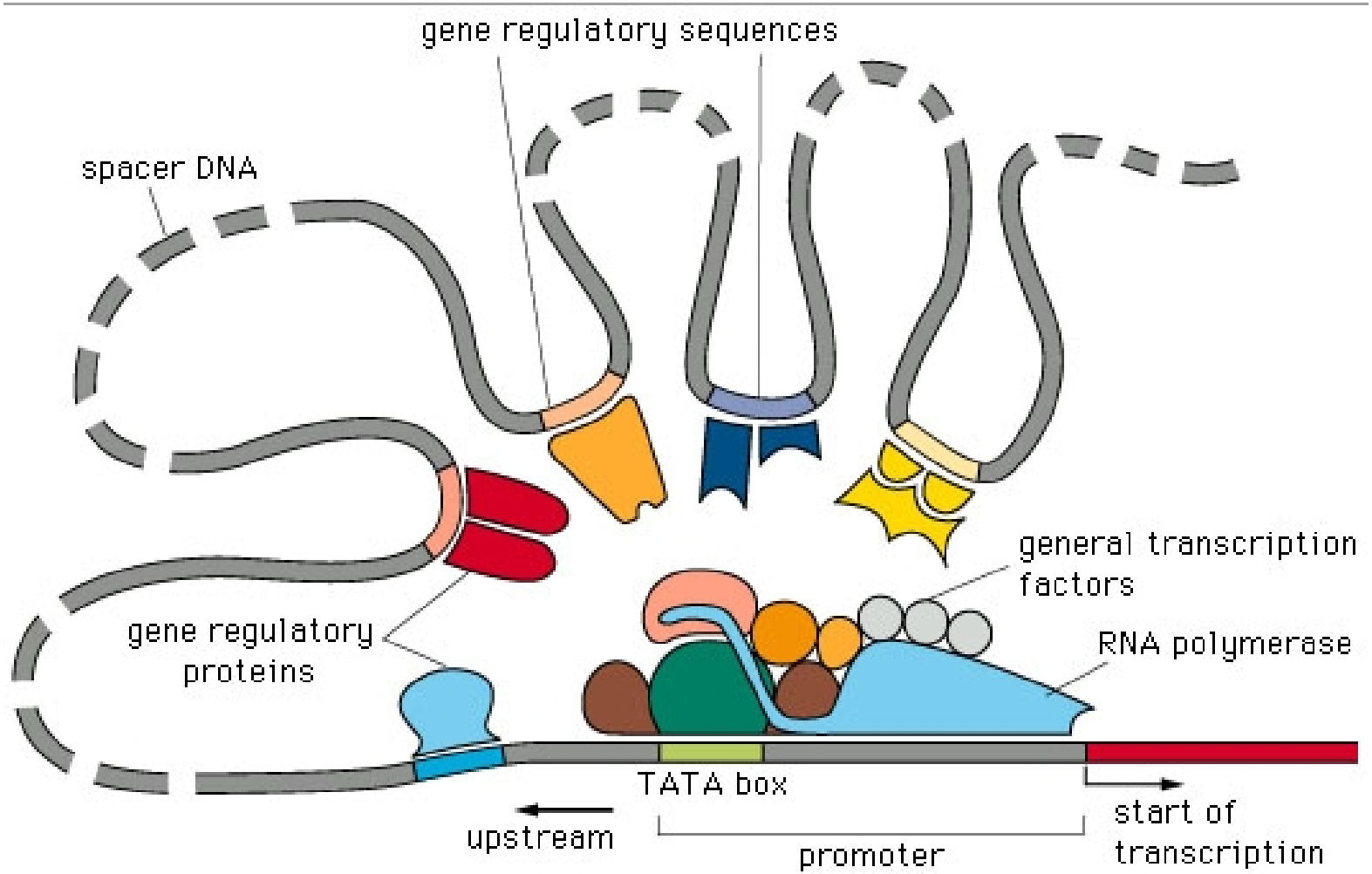
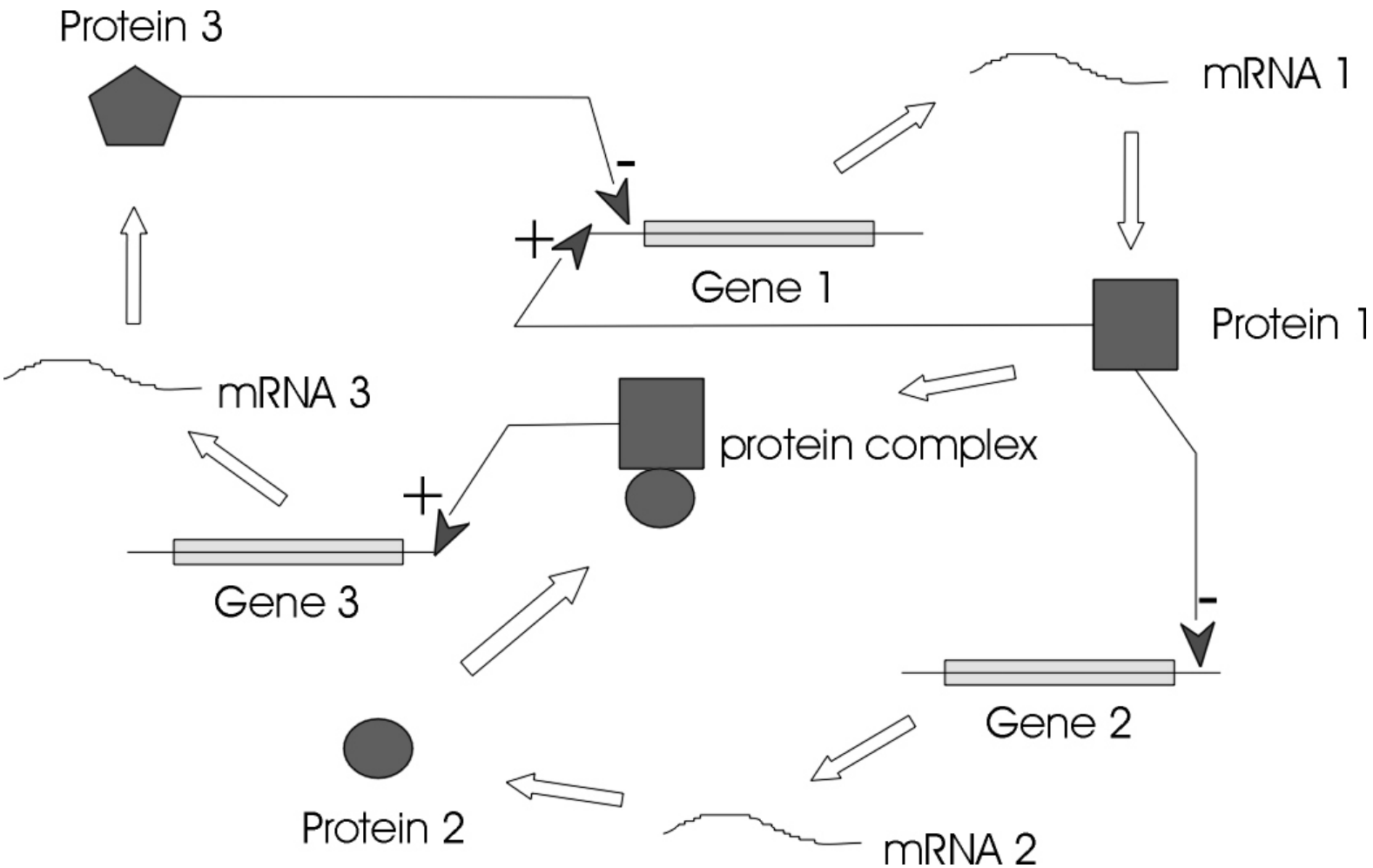


Image adapted from: National Human Genome Research Institute.

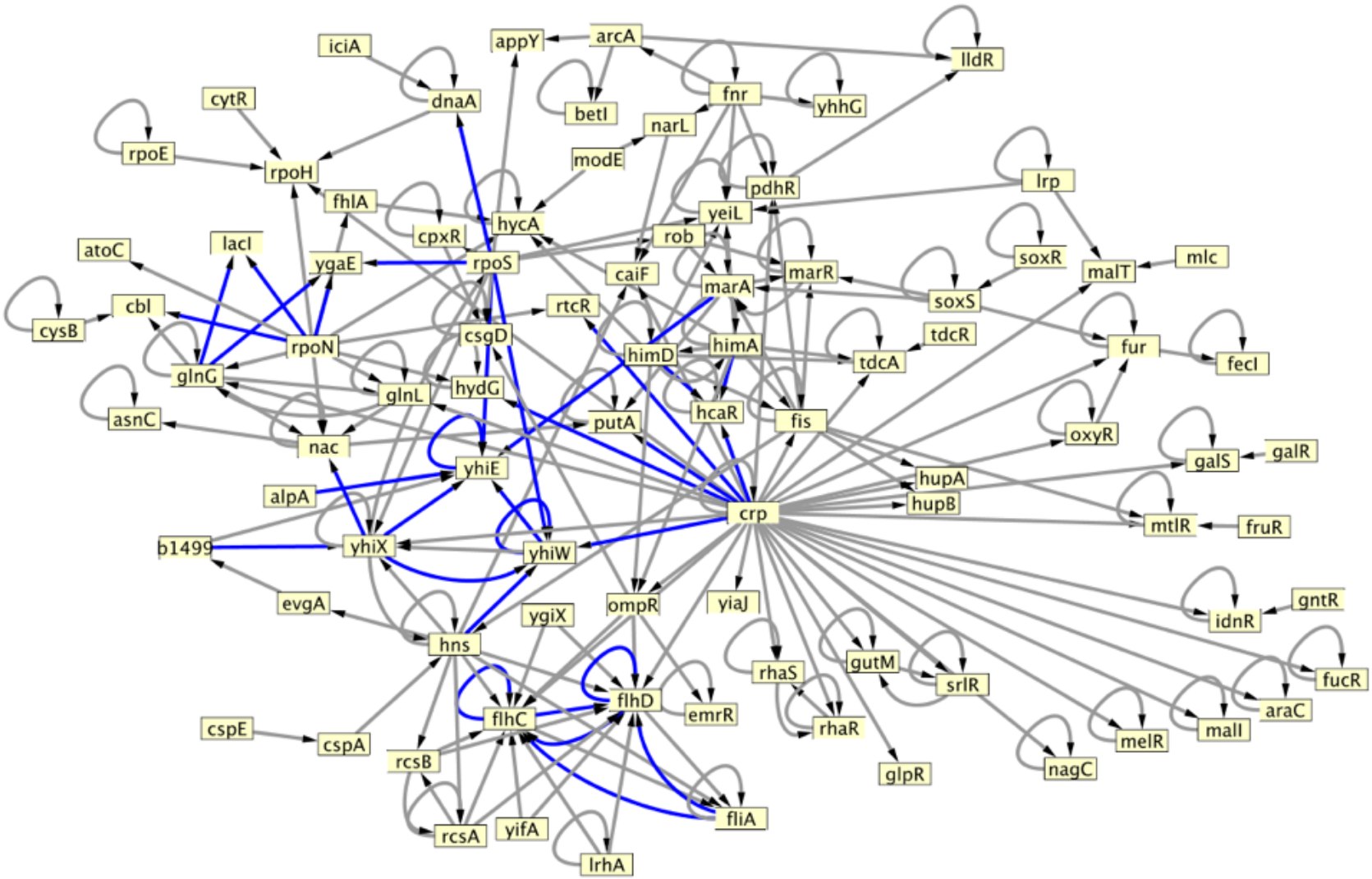
Gene expression regulation



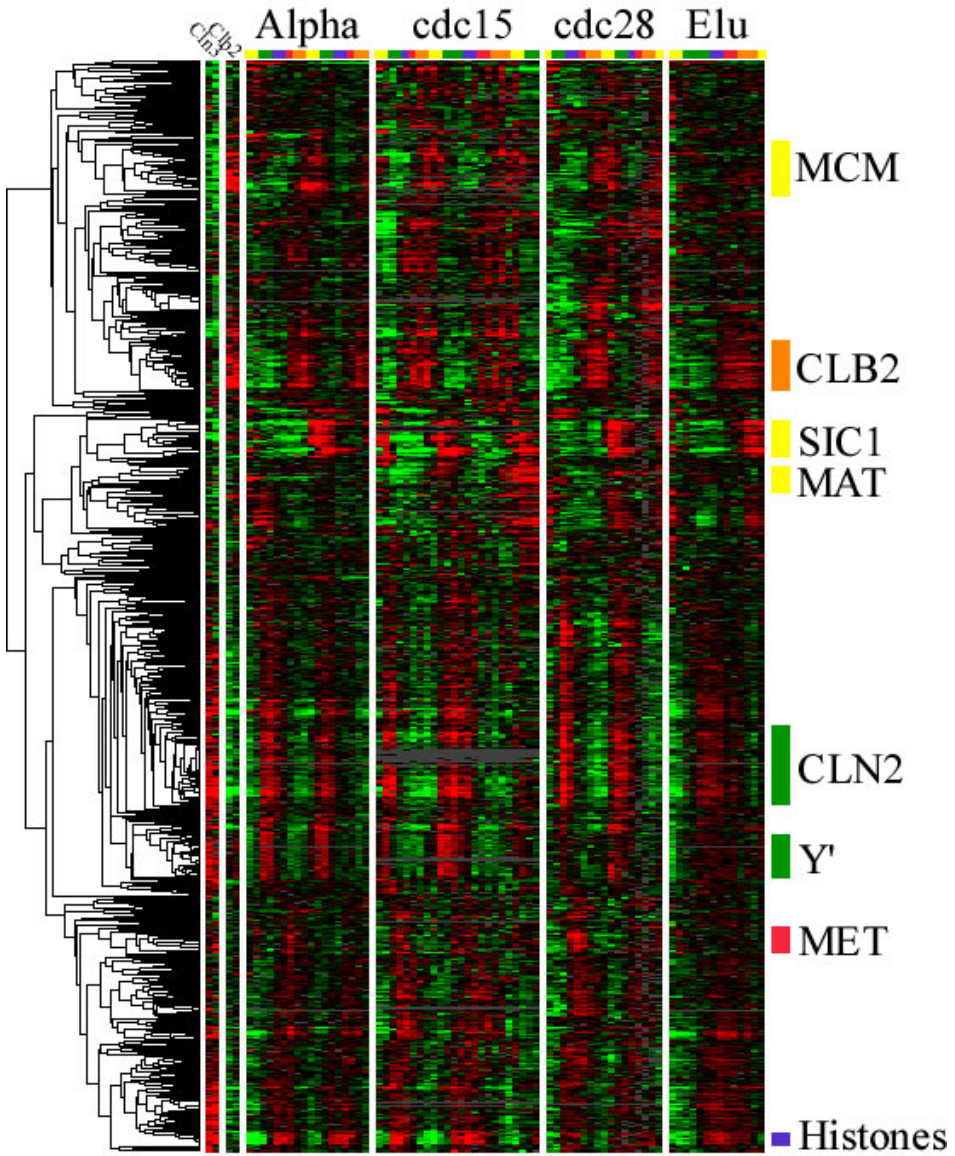
Gene regulatory network



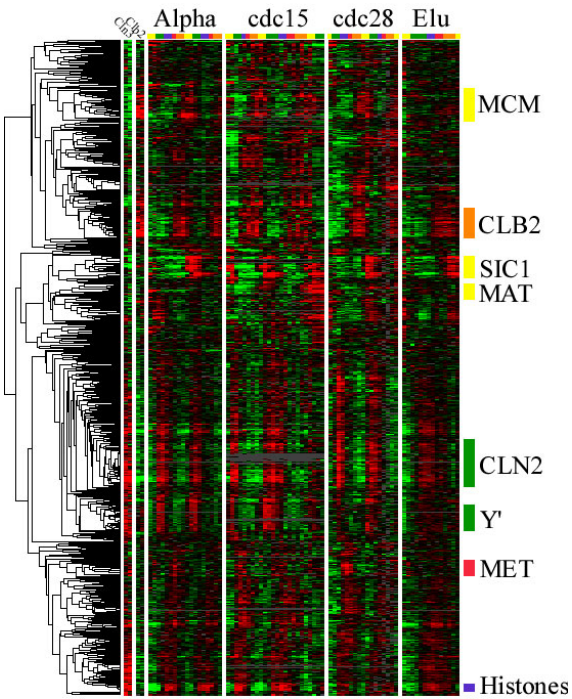
Gene regulatory network of E. coli

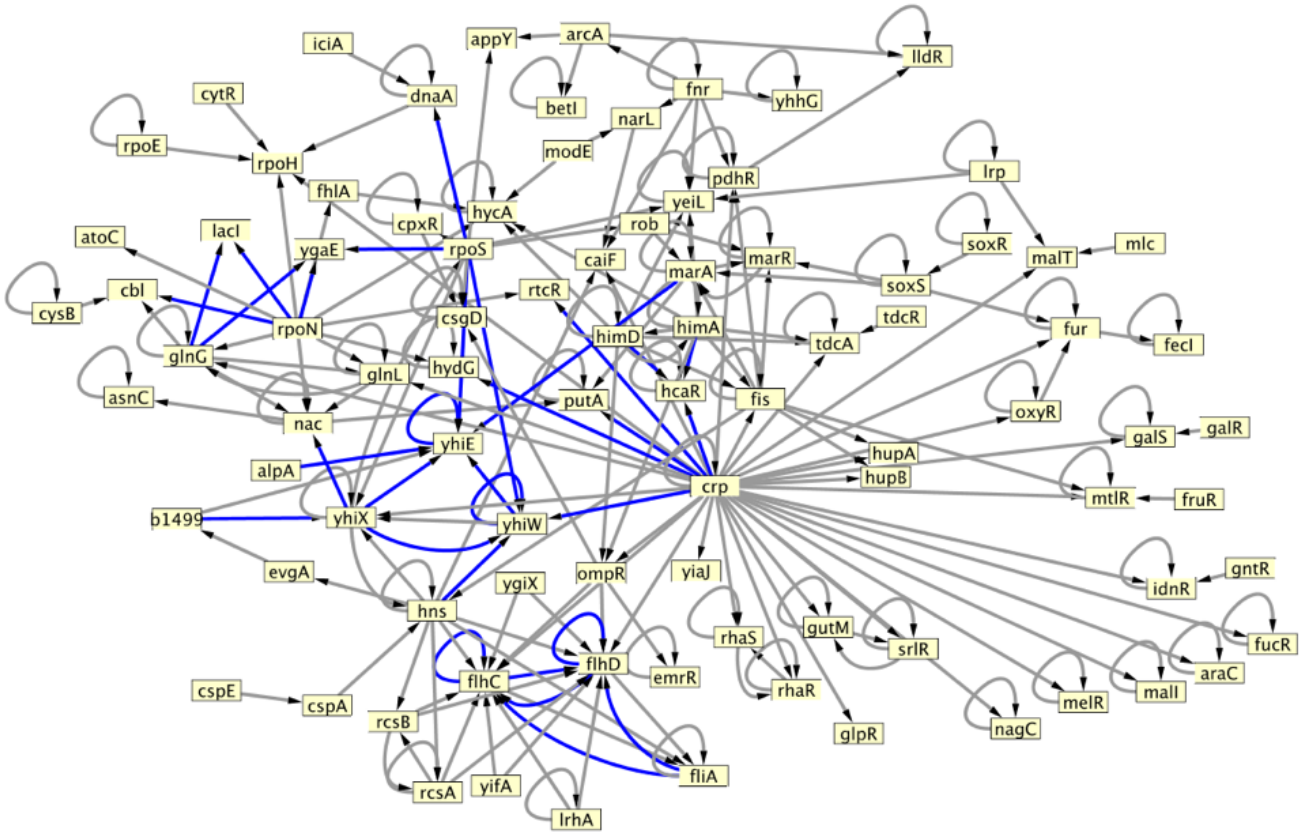


Gene expression data



Reconstruction of gene regulatory network from expression data

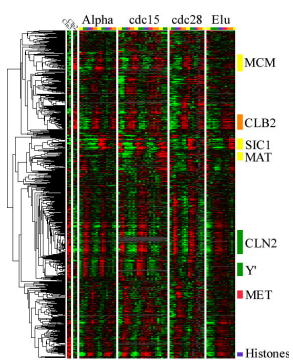


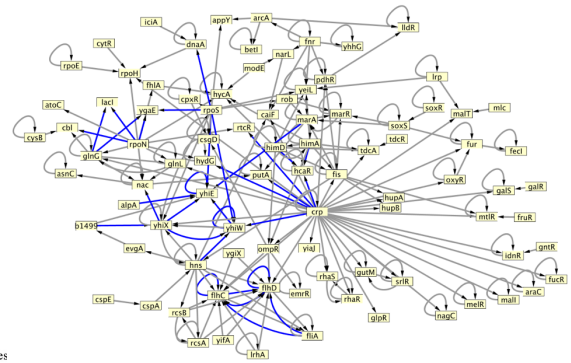


De novo inference

The problem

Given a set of gene expressions, infer the regulations.





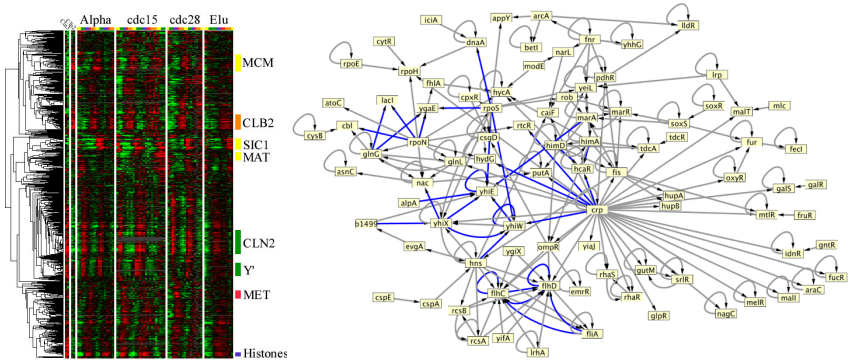
How?

- Connect "similar genes": correlation, mutual-information...
- Model-based approaches: dynamic systems, boolean networks, state-space models, Bayesian networks
- Sparse regression: regulators as the smallest set of TF necessary to predict the expression of the target (GENIE, TIGRESS...)

De novo inference

The problem

Given a set of gene expressions, infer the regulations.



How?

- Connect "similar genes": correlation, mutual-information...
- Model-based approaches: dynamic systems, boolean networks, state-space models, Bayesian networks
- Sparse regression: regulators as the smallest set of TF necessary to predict the expression of the target (GENIE, TIGRESS...)

Predicting regulation by sparse regression

- Let $Y \in \mathbb{R}^n$ the expression of a gene, and $X_1, \ldots, X_p \in \mathbb{R}^n$ the expression of all TFs. We look for a model

$$Y = \sum_{i=1}^{p} \beta_i X_i + \text{noise}$$

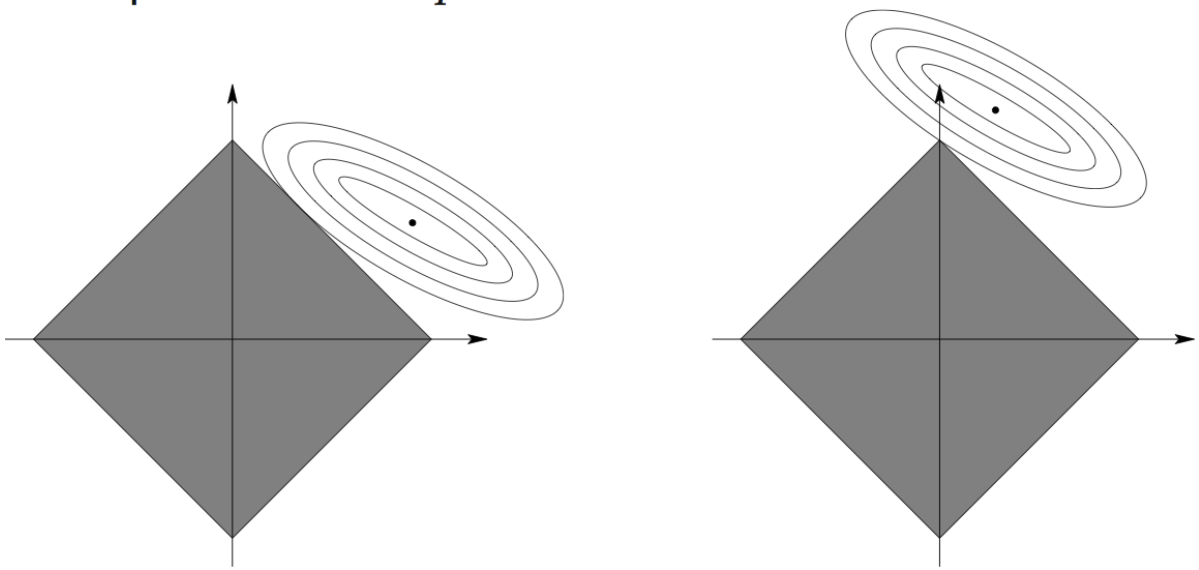
  where β is sparse, i.e., only a few β_i are non-zero.
- We can estimate the sparse regression model from a matrix of expression data.
- Non-zero β_i's correspond to predicted regulators.

Feature selection with the lasso

$$\min_{\beta \in \mathbb{R}^p} \| Y - X\beta \|^2 + \lambda \| \beta \|_1 \qquad \text{where} \| \beta \|_1 = \sum_{i=1}^{p} |\beta_i|$$

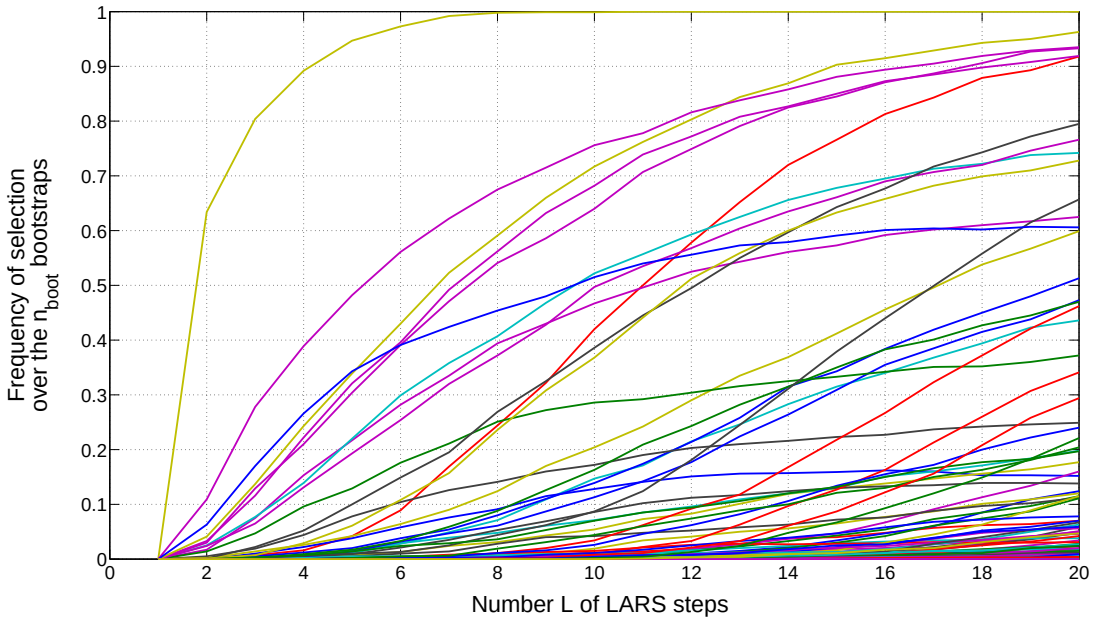
- No explicit solution, but this is just a quadratic program (Tibshirani, 1996; Chen et al., 1998).
- Efficient solution with the LARS (Efron et al., 2004)
- When t is not too large, the solution will usually be sparse

Geometric interpretation with $p = 2$

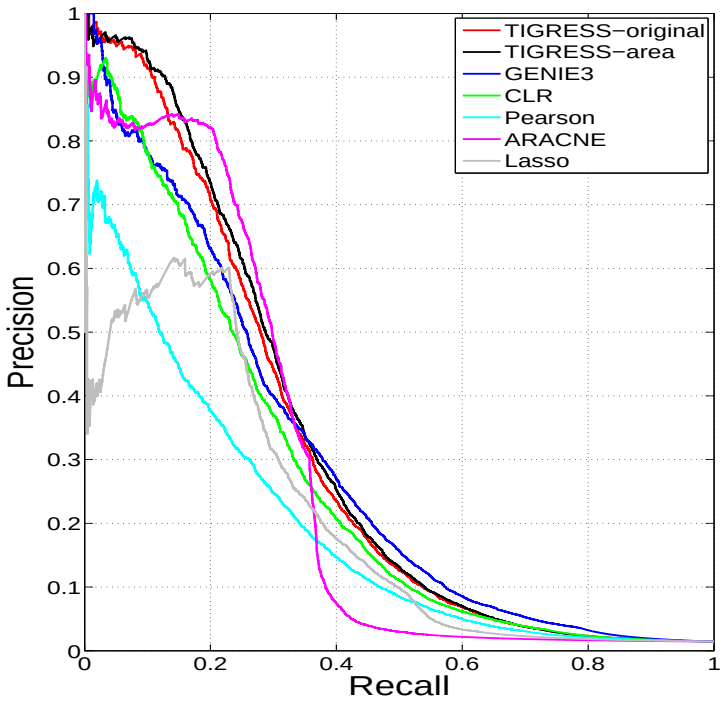


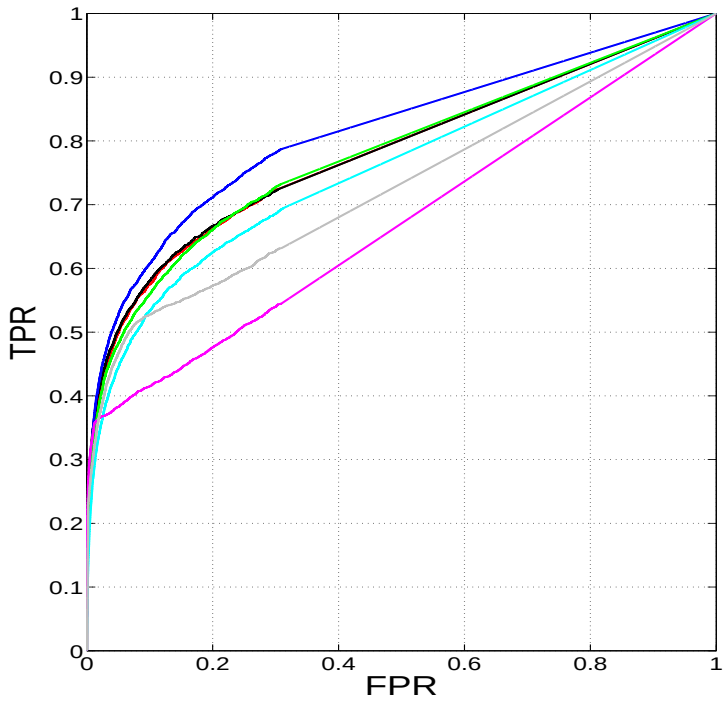
TIGRESS (Haury, Mordelet, Vera-Licona and V., 2012)

- For $t = 1$ to T do
  - Bootstrap a random sample S_t from the training set
  - Randomly reweight each feature (uniform on $[\alpha, 1]$)
  - Select L features with the Lasso
- The score of a feature is the number of times it was selected among the T repeats (Meinshausen and Bühlmann, 2010).
- Rank features (TF-TG interactions) by decreasing area under the score curve



Performance





TPR

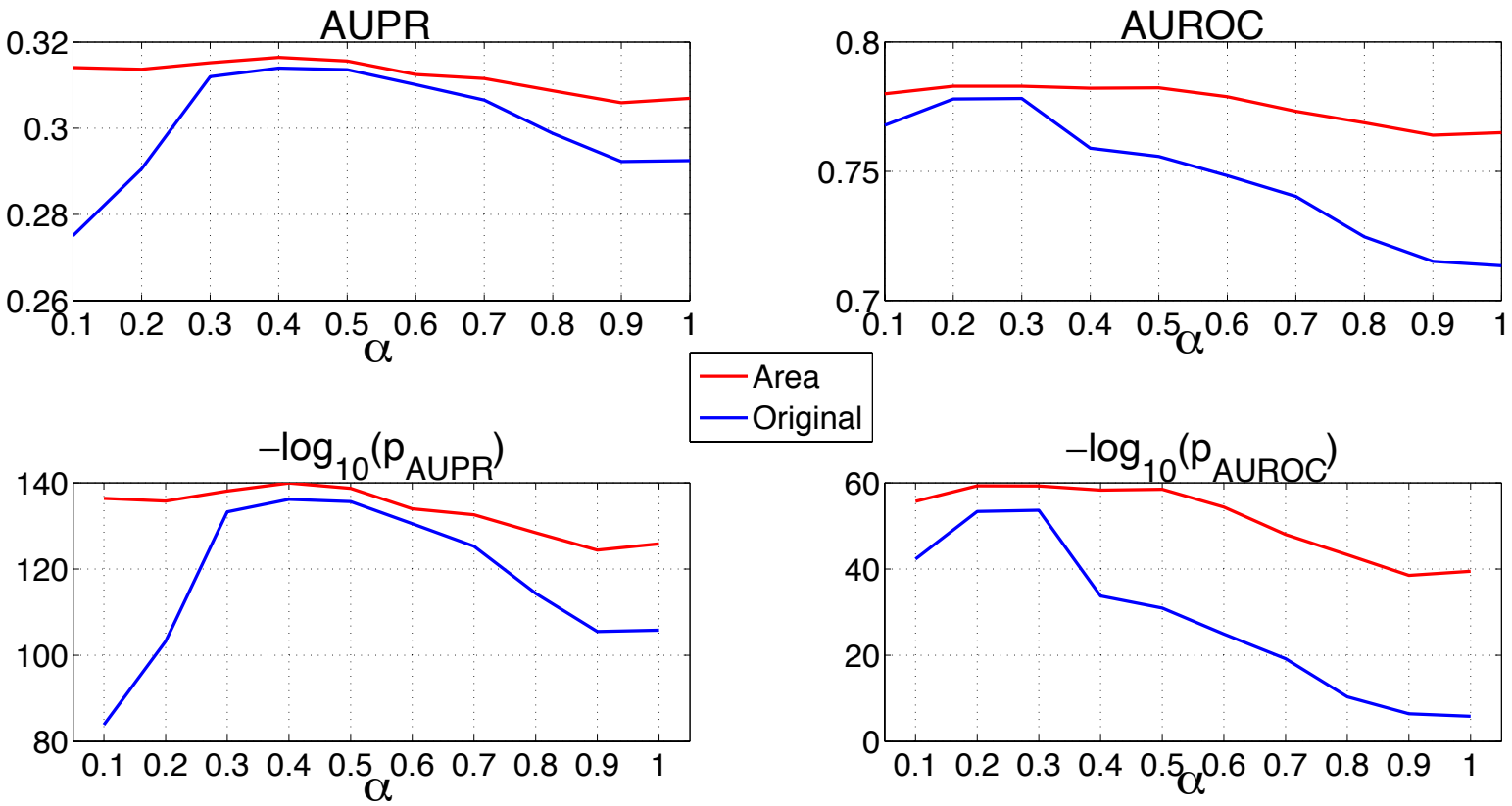
DREAM5: GENIE and TIGRESS ranked 1st and 2nd out or 29 on the *in silico* challenge

TIGRESS vs ...

Algorithm	**AUPR**	p_{AUPR}	**AUROC**	p_{AUROC}
TIGRESS	0.3152	8.01e-139	0.7829	5.43e-60
GENIE3	0.2915	2.91e-105	0.8155	2.30e-107
CLR	0.2654	1.82e-73	0.7817	1.41e-58
Pearson	0.1887	3.71e-13	0.7568	1.44e-32
ARACNE	0.2758	1.73e-85	0.6715	9.82e-01
Lasso	0.2079	1.38e-23	0.7280	1.06e-12

Table: AUPR, AUROC and p-values obtained by several methods on the *in silico* dataset.

Influence of α and scoring method

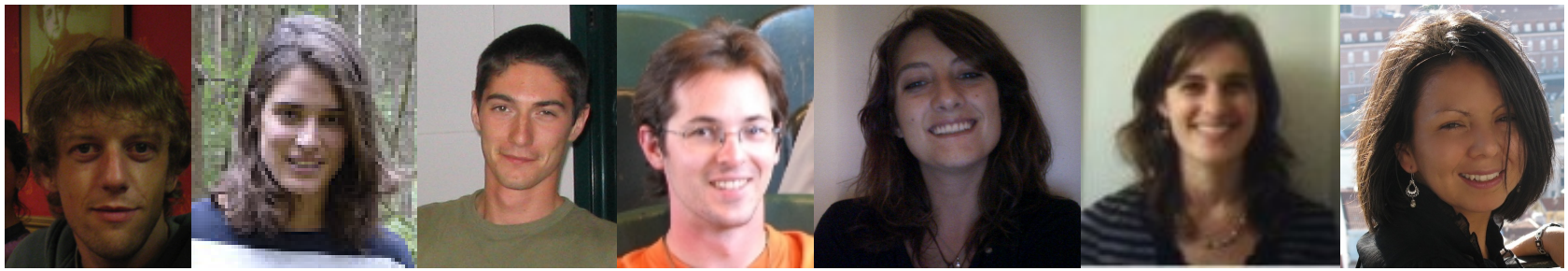


DREAM5 in silico network.

Conclusions

- Convex sparsity-inducing penalties as a way to incorporate prior knowledge
- Specific implementations for specific problems:
  - greedy dichotomic segmentation for fused lasso
  - fast group Lasso for joint segmentation
  - network flow optimization of lasso over the paths of a graph
- Often, feature selection is consistent (although we pay a price when features are very correlated), stability selection may help
- Numerous applications in bioinformatics and beyond!

Thanks!



Kevin Bleakley (INRIA), Elsa Bernard (ParisTech/Institut Curie), Laurent Jacob (UC Berkeley/CNRS), Julien Mairal (UC Berkeley/INRIA), Anne-Claire Haury, Fantine Mordelet (Duke), Paola Vera-Licona (Institut Curie)