Including prior knowledge in machine learning for genomic data

Jean-Philippe Vert

Mines ParisTech / Curie Institute / Inserm

StatLearn workshop, Grenoble, March 17, 2011
Outline

1. Motivations
2. Finding multiple change-points in a single profile
3. Finding multiple change-points shared by many signals
4. Supervised classification of genomic profiles
5. Learning molecular classifiers with network information
6. Conclusion
1 Motivations
2 Finding multiple change-points in a single profile
3 Finding multiple change-points shared by many signals
4 Supervised classification of genomic profiles
5 Learning molecular classifiers with network information
6 Conclusion
Chromosomal aberrations in cancer
Comparative Genomic Hybridization (CGH)

\[ \log_2 \left( \frac{\text{# copies du BAC(x) test}}{\text{# copies du BAC(x) ref}} \right) \]

Jain et al. Genome research 2002 12:325-332
Can we identify breakpoints and "smooth" each profile?
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.
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)
Can we identify the cancer subtype? (diagnosis)

J.P Vert (ParisTech)
Can we predict the future evolution? (prognosis)

A Gene-Expression Profiling

B St. Gallen Criteria

NO. AT RISK
Good signature 50 57 54 45 31 22 12
Poor signature 91 72 55 41 26 17 9

NO. AT RISK
Low risk 22 22 21 17 9 5 2
High risk 120 107 88 60 48 34 19

P<0.001

P=0.05
Many problems...
Data are high-dimensional, but "structured"
Classification accuracy is not all, interpretation is necessary (pattern discovery)
A general strategy

$$\min R(\beta) + \lambda \Omega(\beta)$$
1. Motivations

2. Finding multiple change-points in a single profile

3. Finding multiple change-points shared by many signals

4. Supervised classification of genomic profiles

5. Learning molecular classifiers with network information

6. Conclusion
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.
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

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

- 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

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

- This is an optimization problem over the \( \binom{p}{k} \) partitions...

- Dynamic programming finds the solution in \( O(p^2k) \) 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

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

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 \( O(p^2k) \) in time and \( O(p^2) \) in memory.

But: does not scale to \( p = 10^6 \sim 10^9 \)...
Promoting sparsity with the $\ell_1$ penalty

The $\ell_1$ 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.
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 $\beta$ 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

\[ \min_{\beta \in \mathbb{R}^p} \| Y - \beta \|_2^2 \quad \text{such that} \quad \sum_{i=1}^{p-1} |\beta_{i+1} - \beta_i| \leq \mu \]

- QP with sparse linear constraints in \( O(p^2) \) \(-\rightarrow 135 \text{ min for } p = 10^5 \) (Tibshirani and Wang, 2008)
- Coordinate descent-like method \( O(p) \)? \(-\rightarrow 3s \text{ s for } p = 10^5 \) (Friedman et al., 2007)
- For all \( \mu \) with the LARS in \( O(pK) \) (Harchaoui and Levy-Leduc, 2008)
- For all \( \mu \) in \( O(p \ln p) \) (Hoefling, 2009)
- For the first \( K \) change-points in \( O(p \ln K) \) (Bleakley and V., 2010)
Speed trial: 2 s. for $K = 100$, $p = 10^7$
A fast method for multiple change-point detection
An embedded method that boils down to a dichotomic wrapper method (very different from dynamic programming)
Outline

1. Motivations
2. Finding multiple change-points in a single profile
3. Finding multiple change-points shared by many signals
4. Supervised classification of genomic profiles
5. Learning molecular classifiers with network information
6. Conclusion
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.
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.
Define the "optimal" piecewise constant approximation \( \hat{U} \in \mathbb{R}^{p \times n} \) of \( Y \) as the solution of

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

- DP finds the solution in \( O(p^2kn) \) 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 $\ell_1/\ell_2$-norm induces sparse solutions *at the group level*:

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

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

$$= \sqrt{w_1^2 + w_2^2} + \sqrt{w_3^2}$$
TV approximator for many signals

- Replace

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

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)?
Make the change of variables:

\[
\gamma = U_{1,\bullet},
\]

\[
\beta_{i,\bullet} = w_i \left( U_{i+1,\bullet} - U_{i,\bullet} \right) \quad \text{for } i = 1, \ldots, p - 1.
\]

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

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

where \( \tilde{Y} \) is the centered signal matrix and \( \tilde{X} \) 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 $\bar{X}_{\bullet,A}^\top \bar{X}_{\bullet,B}$ 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 $C = \left(\bar{X}_{\bullet,A}^\top \bar{X}_{\bullet,A}\right)^{-1} R$ 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 $\sigma^2$

Does the TV approximator correctly estimate the first change-point as $p$ increases?
Consistency of the unweighted TV approximator

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

**Theorem**

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

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

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

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

**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
The first change-point \( \hat{i} \) found by TV approximator maximizes 
\[ F_i = \| \hat{c}_{i,:} \|^2 , \]
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 \( \chi^2 \) 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?

\[ \rho = 100, \, k = 10, \, \beta^2 = 1, \, \sigma^2 \in \{0.05; 0.2; 1\} \]
Outline

1. Motivations
2. Finding multiple change-points in a single profile
3. Finding multiple change-points shared by many signals
4. Supervised classification of genomic profiles
5. Learning molecular classifiers with network information
6. Conclusion
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 $f : \mathbb{R}^p \rightarrow [-1, 1]$
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 \textit{sparse} solutions
- Second term leads to \textit{piecewise constant} solutions
Fused lasso for supervised classification (Rapaport et al., 2008)

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

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

Implementation

- When \(\ell\) is the hinge loss (fused SVM), this is a linear program -> up to \(p = 10^3 \sim 10^4\)
- When \(\ell\) 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)

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

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

Implementation

- When \( \ell \) is the hinge loss (fused SVM), this is a linear program -> up to \( p = 10^3 \sim 10^4 \)
- When \( \ell \) is convex and smooth (logistic, quadratic), efficient implementation with proximal methods -> up to \( p = 10^8 \sim 10^9 \)
Example: predicting metastasis in melanoma
1. Motivations

2. Finding multiple change-points in a single profile

3. Finding multiple change-points shared by many signals

4. Supervised classification of genomic profiles

5. Learning molecular classifiers with network information

6. Conclusion
Molecular diagnosis / prognosis / theragnosis

J.P. Vert (ParisTech)

Prior knowledge in ML

StatLearn 43 / 68
Gene networks

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

An idea (Rapaport et al., 2007)

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

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

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

An idea (Rapaport et al., 2007)

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

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

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

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

The function \( f(x) = \beta^T x \) where \( \beta \) is solution of

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

is equal to \( g(x) = \gamma^T \Phi(x) \) where \( \gamma \) is solution of

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

and where

\[
\Phi(x)^T \Phi(x') = x^T K_G x'
\]

for \( K_G = L^* \), the pseudo-inverse of the graph Laplacian.
Spectral analysis of gene expression profiles using gene networks

Fig. 5. [Diagram of gene networks and pathways]

The work was supported by the grant ACI-IMPBIO-2004-47 of the French Ministry for Research and New Technologies.
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 \( \beta \) 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 \( \ell_1/\ell_2 \)-norm induces sparse solutions *at the group level*:

\[
\Omega_{\text{group}}(w) = \sum_g \| w_g \|_2
\]

\[
\Omega(w_1, w_2, w_3) = \|(w_1, w_2)\|_2 + \|w_3\|_2
\]
What if a gene belongs to several groups?

**Issue of using the group-lasso**
- $\Omega_{\text{group}}(w) = \sum_g \|w_g\|_2$ sets groups to 0.
- One variable is selected $\iff$ 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 $v_g$:

$$\min_{w,v} L(w) + \lambda \sum_{g \in G} \|v_g\|_2$$

$$w = \sum_{g \in G} v_g$$

$\text{supp}(v_g) \subseteq g$.

Properties

- Resulting support is a *union* of groups in $G$.
- 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

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

with

\[
\Omega_{overlap}(w) \triangleq \left\{ \begin{array}{l}
\min_v \sum_{g \in G} \|v_g\|_2 \\
w = \sum_{g \in G} v_g \\
\text{supp}(v_g) \subseteq g.
\end{array} \right.
\]

(*)

Property

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

Balls for $\Omega^G_{\text{group}} (\cdot)$ (middle) and $\Omega^G_{\text{overlap}} (\cdot)$ (right) for the groups $G = \{\{1, 2\}, \{2, 3\}\}$ where $w_2$ is represented as the vertical coordinate. Left: group-lasso ($G = \{\{1, 2\}, \{3\}\}$), for comparison.
Consistency in group support (Jacob et al., 2009)

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

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 \( (\hat{v}_g)_{g \in G} \) such that
  \[
  \{ g \in G | \hat{v}_g \neq 0 \} = \{ g \in G | \tilde{v}_g \neq 0 \}.
  \]
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
\[
\bar{w} = \sum_g \bar{v}_g \quad \text{and} \quad \Omega^G_{\text{overlap}} (\bar{w}) = \sum \|\bar{v}_g\|_2.
\]
Consider the regularized empirical risk minimization problem
\[
L(w) + \lambda \Omega^G_{\text{overlap}} (w).
\]
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 \( (\hat{v}_g)_{g \in G} \) such that
\[
\{ g \in G | \hat{v}_g \neq 0 \} = \{ g \in G | \bar{v}_g \neq 0 \}.
\]
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\textsuperscript{th} and 5\textsuperscript{th} groups.
- Learn from 100 training points.

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

Two solutions

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

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

<table>
<thead>
<tr>
<th>METHOD</th>
<th>$\ell_1$ (error)</th>
<th>$\Omega_{\text{overlap}}^G$ (mean)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERROR</td>
<td>0.38 ± 0.04</td>
<td>0.36 ± 0.03</td>
</tr>
<tr>
<td>MEAN # PATH</td>
<td>130</td>
<td>30</td>
</tr>
</tbody>
</table>

Graph on the genes.

<table>
<thead>
<tr>
<th>METHOD</th>
<th>$\ell_1$ (error)</th>
<th>$\Omega_{\text{graph}}$ (mean)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERROR</td>
<td>0.39 ± 0.04</td>
<td>0.36 ± 0.01</td>
</tr>
<tr>
<td>AV. SIZE C.C.</td>
<td>1.03</td>
<td>1.30</td>
</tr>
</tbody>
</table>
Lasso signature

![Diagram of Lasso signature with nodes and connections]

Prior knowledge in ML

StatLearn 65 / 68
Graph Lasso signature

Prior knowledge in ML
Outline

1. Motivations
2. Finding multiple change-points in a single profile
3. Finding multiple change-points shared by many signals
4. Supervised classification of genomic profiles
5. Learning molecular classifiers with network information
6. Conclusion
Conclusions

- Feature / pattern selection in high dimension is central for many applications.
- Convex sparsity-inducing penalties or positive definite kernels are promising.
- Success stories remain limited on real data...
- Need to adjust the complexity of the model to the data available.