A fast algorithm for structured gene selection

MLSB 2010

Sofia Mosci
DISI, Università degli Studi di Genova

Joint work with
A. Verri(1), S. Villa(1), and L. Rosasco(2)
1 - Universita' di Genova 2-IIT-MIT

Motivation

Gene selection problem

extracting a predictive model depending on a small subset of genes

many variable selection algorithms are available (filters wrappers and embedded)

- low accuracy
- low stability
- low interpretability

Motivation

Gene selection problem

extracting a predictive model depending on a small subset of genes

many variable selection algorithms are available (filters wrappers and embedded)

- low accuracy
- low stability
- low interpretability

Motivation

Strong prior is often available!

Genes must be selected according to groups defined a priori.

Examples of groups:

- GO
- KEGG
- ad hoc grouping

Group lasso

References:

Lanckriet et al.'04, Meier et al. '06, Yuan-Lin '06, Bach '08,...

Group lasso drawback: groups must be a partition of the genes

Motivation

Group lasso with overlap (Jacob, Obozinski and Vert '09)

genes must be selected group-wise according to groups defined a priori.
Like group lasso but groups may overlap.

Advantages:

- higher stability
- higher accuracy
- higher interpretability

Disadvantages:

- implementability

Motivation

Goal

to develop a scalable approach to group lasso with overlap

Plan:

- Proximal methods for Sparsity based regularization
- Group lasso with overlap: the initial approach
- Group lasso with overlap: our projection algorithm
- Experiments

Sparsity Based Regularization

General sparsity prior: variables are organized in separate, nested or possibly overlapping groups.

Given a training set $(x_i, y_i)_{i=1}^n$, with $x_i \in \mathbb{R}^d$, consider

$$\operatorname*{argmin}_{\beta \in \mathbb{R}^d} \left\{ \underbrace{\frac{1}{n}\|X\beta - y\|^2}_{data\ \ term} + \underbrace{2\tau\Omega(\beta)}_{penalty\ \ term} \right\}$$

where

- $[X]_{i,j} = (x_i)_j$
- $\Omega : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$, encodes the sparsity prior, and is convex and one-homogeneous ($\Omega(\lambda\beta) = \lambda\Omega(\beta)$, $\forall \beta \in \mathbb{R}^d$ and $\lambda \in \mathbb{R}^+$).

A Proximal Algorithm

Require: $\tau, \sigma > 0$
Initialize: $\beta^0 = 0$
while `convergence not reached` **do**
  $p := p + 1$

$$\beta^p = \quad prox_{\frac{\tau}{\sigma}\Omega} \quad \left(\beta^{p-1} - \frac{1}{n\sigma} X^T (X\beta^{p-1} - y)\right)$$

end while
return β^p

References:

- Lions-Mercier('79), Passty ('76), Tseng (90s), Chen-Rockafellar('89), Eckstein ('89), Combettes-Wajs('05)
- Duchi and Singer '09, Jenatton et al. '10, Mosci et al. '10 for machine learning

A Proximal Algorithm

Require: $\tau, \sigma > 0$
Initialize: $\beta^0 = 0$
while `convergence not reached` **do**
$p := p + 1$

$$\beta^p = prox_{\frac{\tau}{\sigma}\Omega}\left(\beta^{p-1} - \underbrace{\frac{1}{n\sigma}X^T(X\beta^{p-1} - y))}_{\text{gradient of the data term}}\right)$$

end while
return β^p

References:

- Lions-Mercier('79), Passty ('76), Tseng (90s), Chen-Rockafellar('89), Eckstein ('89), Combettes-Wajs('05)
- Duchi and Singer '09, Jenatton et al. '10, Mosci et al. '10 for machine learning

A Proximal Algorithm

Require: $\tau, \sigma > 0$
Initialize: $\beta^0 = 0$
while `convergence not reached` **do**

$p := p + 1$

$$\beta^p = \underbrace{prox_{\frac{\tau}{\sigma}\Omega}}_{\textcolor{red}{proximity\ operator\ of\ \Omega}} \left(\beta^{p-1} - \underbrace{\frac{1}{n\sigma} X^T (X\beta^{p-1} - y))}_{gradient\ of\ the\ data\ term} \right)$$

end while
return β^p

References:

- Lions-Mercier('79), Passty ('76), Tseng (90s), Chen-Rockafellar('89), Eckstein ('89), Combettes-Wajs('05)
- Duchi and Singer '09, Jenatton et al. '10, Mosci et al. '10 for machine learning

Iterative soft-thresholding for the lasso

Prior: the relevant variables are a subset of the total variables

$$\underset{\beta \in \mathbb{R}^d}{\operatorname{argmin}} \frac{1}{n} \|X\beta - y\|^2 + 2\tau \|\beta\|_1$$

Require: $\tau, \sigma > 0$
Initialize: $\beta^0 = 0$
while `convergence not reached` **do**
  $p := p + 1$

$$\beta^p = \mathbf{S}_{\frac{\tau}{\sigma}} \left(\beta^{p-1} - \frac{1}{n\sigma} X^T (X\beta^{p-1} - y) \right)$$

end while
return β^p

where $\mathbf{S}$ is the soft-thresholding operator: $\mathbf{S}_\lambda(\beta^j) := (|\beta^j| - \lambda)_+ sign(\beta^j)$

References:
Daubechies et al. '04, Combettes '05, Figuereido et al. '07

Iterative soft-thresholding for group lasso

Prior: the relevant variables are union of a subset of the B groups given a priori, $\{G_r\}_{r=1}^{B}$, that make a block partition of $\{1,\ldots,d\}$

$$\operatorname*{argmin}_{\beta\in\mathbb{R}^d} \frac{1}{n}\|X\beta - y\|^2 + \underbrace{2\tau\sum_{r=1}^{M}\sqrt{\sum_{j\in G_r}\beta_j^2}}_{\Omega}$$

$$\beta^p = \tilde{\mathbf{S}}_{\frac{\tau}{\sigma}}\left(\beta^{p-1} - \frac{1}{n\sigma}X^T(X\beta^{p-1} - y)\right)$$

where $\tilde{\mathbf{S}}$ is the group-wise soft-thresholding operator:

$$\tilde{\mathbf{S}}_\lambda(\beta_k) = (\|\beta_k\|_k - \lambda)_+ \frac{\beta_k}{\|\beta_k\|_k}$$

Group Lasso with overlap

Prior: the relevant variables are the union of a small subset of the B groups given a priori, $\mathcal{G} = \{G_r\}_{r=1}^{B}$ with $G_r \subset \{1, \ldots, d\}$
Like Group Lasso but groups may overlap

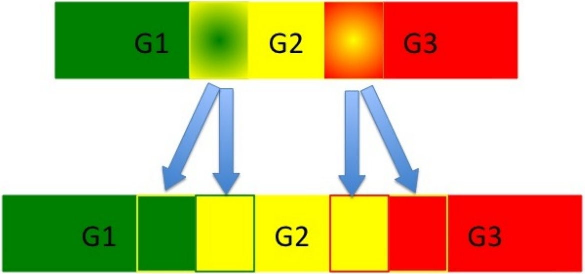
$$\underset{\beta \in \mathbb{R}^d}{\operatorname{argmin}} \left\{ \frac{1}{n} \|X\beta - y\|^2 + 2\tau \Omega^{\mathcal{G}}_{\text{overlap}}(\beta) \right\},$$

$$\Omega^{\mathcal{G}}_{\text{overlap}}(\beta) = \inf_{\substack{(v_1,\ldots,v_M),\, v_r \in \mathbb{R}^d, \\ \operatorname{supp}(v_r) \subset G_r,\, \sum_{r=1}^{M} v_r = \beta}} \sum_{r=1}^{M} \|v_r\|.$$

Reference:
Jacob, Obozinski and Vert, *Group Lasso with Overlap and Graph Lasso*, ICML 2009

Group Lasso with overlap: the replication approach

A simple implementation is obtained by replicating variables belonging to more than one group, and using any algorithm for standard group lasso (e.g iterative group-wise soft-thresholding).



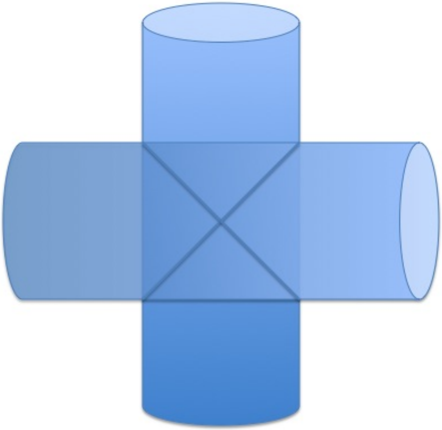
Drawback: as the degree of overlap increases the dimensionality increases and the computational burden may become very high!

Group Lasso with overlap: projection algorithm

$$prox_{\tau\Omega^{\mathcal{G}}_{\text{overlap}}} = I - \pi_{\tau K}$$

K is the intersection of cylinders centered in a coordinate subspace.

Group Lasso with overlap: projection algorithm



Group Lasso with overlap: projection algorithm

$$prox_{\tau\Omega^{\mathcal{G}}_{\text{overlap}}} = I - \pi_{\tau K}$$

K is the intersection of cylinders centered in a coordinate subspace.

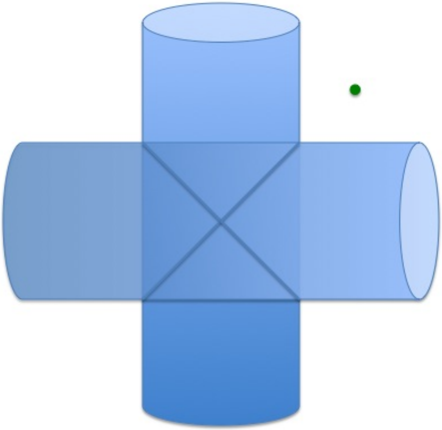
Only a (small) subset of the cylinders are active.

For a given $\beta \in \mathbb{R}^d$, the projection onto τK is given by

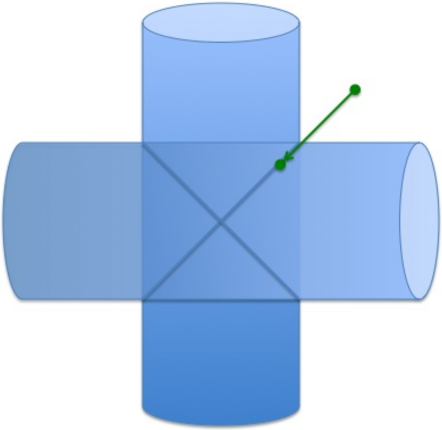
$$\begin{array}{ll} \textit{argmin} & \|v - \beta\|^2 \\ \textit{s.t.} & v \in \mathbb{R}^d, \|v\|_G \leq \tau \textit{ per } G \in \hat{\mathcal{G}}. \end{array}$$

where $\hat{\mathcal{G}} := \{G \in \mathcal{G},\ \|\beta\|_G > \tau\}$ is the set of active groups .

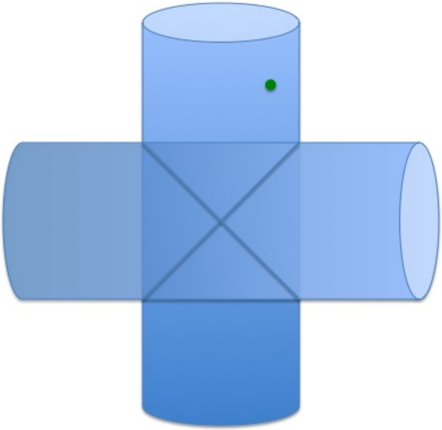
Group Lasso with overlap: projection algorithm



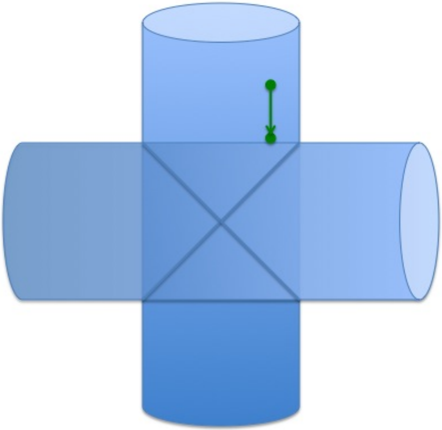
Group Lasso with overlap: projection algorithm



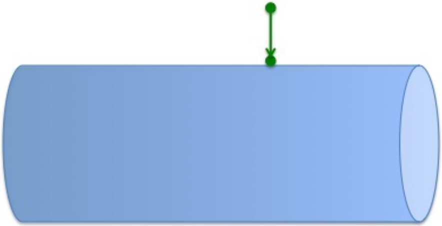
Group Lasso with overlap: projection algorithm



Group Lasso with overlap: projection algorithm



Group Lasso with overlap: projection algorithm



Group Lasso with overlap: projection algorithm

$$prox_{\tau\Omega^{\mathcal{G}}_{\text{overlap}}} = I - \pi_{\tau K}$$

K is the intersection of cylinders centered in a coordinate subspace.

Only a (small) subset of the cylinders are active.

For a given $\beta \in \mathbb{R}^d$, the projection onto τK is given by

$$\begin{array}{ll} argmin & \|v - \beta\|^2 \\ s.t. & v \in \mathbb{R}^d, \|v\|_G \leq \tau \text{ per } G \in \hat{\mathcal{G}}. \end{array}$$

where $\hat{\mathcal{G}} := \{G \in \mathcal{G}, \ \|\beta\|_G > \tau\}$ is the set of active groups .

The projection can be computed by solving the dual problem in $\mathbb{R}^{\hat{B}}$

$$\lambda^* = \text{argmax}_{\lambda \in \mathbb{R}^{\hat{B}}_+} \sum_{j=1}^d \frac{-w_j^2}{1+\sum_{r=1}^{\hat{B}} \mathbf{1}(j \in \hat{G}_r)\lambda_r} - \sum_{r=1}^{\hat{B}} \lambda_r \tau^2,$$

Gradient step

$$w = \beta^{p-1} - \frac{1}{\sigma} X^T (X\beta^{p-1} - y)$$

Projection

- find the set of active groups $\hat{\mathcal{G}} := \{\hat{G}_1, \ldots, \hat{G}_{\hat{B}}\}$
- compute λ^* solution of the dual problem associated to the reduced projection:

$$\lambda^* = \operatorname{argmax}_{\lambda \in \mathbb{R}_+^{\hat{B}}} \sum_{j=1}^{d} \frac{-w_j^2}{1+\sum_{r=1}^{\hat{B}} \mathbf{1}(j \in \hat{G}_r)\lambda_r} - \sum_{r=1}^{\hat{B}} \lambda_r \tau^2,$$

- $\beta_j^p = w_j - \frac{w_j}{(1+\sum_{r=1}^{\hat{B}} \mathbf{1}(j \in \hat{G}_r)\lambda_r^*)} \qquad$ *for* $j = 1, \ldots, d$

Overall convergence is still guaranteed!

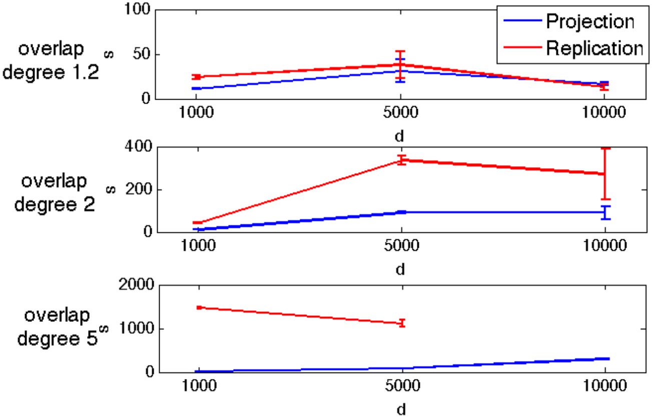
For more details see the forthcoming paper:
Mosci, Verri, Villa and Rosasco, *A primal-dual algorithm for group ℓ_1 regularization with overlapping groups* , NIPS 2010.

Experiments: projection vs duplication

3 relevant groups (with 20% overlap) for a total of 240 variables

for $k > 3$, G_k is built by drawing 100 indices from $[1, \ldots, d]$
$n = 2400$

Running time (in seconds) vs d



Experiments: microarray data

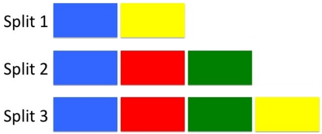
Microarray experiment presented in Jacob, Obozinski and Vert '09 on breast cancer (Van de Vijver et al. '01)

- 8141 genes
- 295 tumors
- 637 gene groups (Subramanian et al. 2005).
- 3-fold cross validation

	Replication	**Projection**
loss:	logistic	square
prediction error:	0.36 ± 0.03	0.30 ± 0.06
# of selected pathways:	6, 5 and 78	2, 3 and 4
computing time:	–	850s

Frequency of selected groups

for the Projection algorithm



Concluding Remarks

I have presented an iterative procedure for solving the group lasso with overlap regularization problem that

- is based on proximal methods and an ad hoc lemma
- is convergent
- is fast and can deal with large data sets

(code available at:
`www.disi.unige.it/person/MosciS/CODE/Prox.html`)

I have not discussed:

- accelerations of the basic schemes
  - Continuation Methods (Hale, Yin and Zhang '08)
  - Adaptive Step size
  - Nesterov Method, linear $\longrightarrow$ quadratic convergence!
    (Nesterov '83, Guler '91, Beck and Teboulle '09)
- other loss functions

Concluding Remarks

I have presented an iterative procedure for solving the group lasso with overlap regularization problem that

- is based on proximal methods and an ad hoc lemma
- is convergent
- is fast and can deal with large data sets

(code available at:
`www.disi.unige.it/person/MosciS/CODE/Prox.html`)

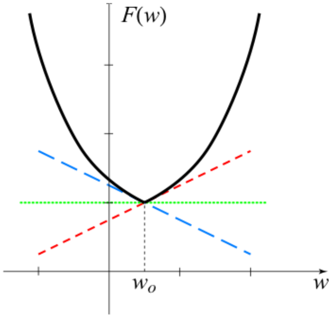
I have not discussed:

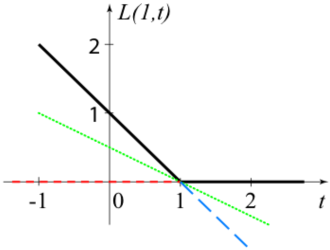
- accelerations of the basic schemes
  - Continuation Methods (Hale, Yin and Zhang '08)
  - Adaptive Step size
  - Nesterov Method, linear $\longrightarrow$ quadratic convergence!
    (Nesterov '83, Guler '91, Beck and Teboulle '09)
- other loss functions

Experimental protocol

- For each (τ, λ):
  - Variable **selection** via Sparse Learning Algorithm with parameter τ on training set
  - **Regression** via Regularized Least Squares(RLS) on training set with parameter λ on selected variables
  - Error estimation on **validation** set (hold-out or cross-validation)
- **Minimization** of the validation error $\rightarrow (\tau_{\mathrm{opt}}, \lambda_{\mathrm{opt}})$
- Error estimation on **test** set

Proximal Operator





For one-homogeneous functionals we have the following result

Let K denote the subdifferential of Ω, $\partial\Omega(0)$, at the origin (which is a convex and closed subset of $\mathbb{R}^d$. For any $\lambda \in \mathbb{R}^+$ we let $\pi_{\lambda K}: \mathbb{R}^d \to \mathbb{R}^d$ be the projection on $\lambda K \subset \mathbb{R}^d$.
Then

$$prox_{\lambda\Omega} = (I - \pi_{\lambda K})$$