

# Increasing stability and interpretability of gene expression signatures

Prediction of breast cancer outcome

Anne-Claire Haury   Laurent Jacob   Jean-Philippe Vert

Center For Computational Biology  $\in$  Mines Paristech/Institut Curie/INSERM U900

SMPGD Marseille - January 14, 2010

# Outline

- 1 Motivation
  - Gene expression signatures
  - Mathematical tools for model selection
- 2 Stabilizing the signature
  - Main procedure
  - Scoring
- 3 Results
- 4 Conclusion and Perspectives

# Outline

## 1 Motivation

- Gene expression signatures
- Mathematical tools for model selection

## 2 Stabilizing the signature

- Main procedure
- Scoring

## 3 Results

## 4 Conclusion and Perspectives

# SIGNATURES AS A PROGNOSTIC TOOL

- **Signature**: list of genes sufficient to predict response (e.g. metastasis vs no metastasis)
- Should involve **few genes**
- Should be **robust** to perturbations of the data and, more importantly, **stable** across datasets

# INSTABILITY OF SIGNATURES FOR BREAST CANCER OUTCOME

- Many proposals through literature, e.g. *Van't Veer et al., 2002*; *Van de Vijver et al., 2002*; *Wang et al. 2005*
- However: **very few overlap** between them, if any
- Moreover: lists of genes may be hard to interpret

# PROPOSAL : GRAPHICAL PRIOR

- Consider a **graph** with PPI + coregulation information (*Chuang et al., 2007*)



- **Assumption** : genes close on the graph build perturbed components
- Consider **groups of genes** from this graph (e.g. edges, connected components, etc.)

# Outline

## 1 Motivation

- Gene expression signatures
- **Mathematical tools for model selection**

## 2 Stabilizing the signature

- Main procedure
- Scoring

## 3 Results

## 4 Conclusion and Perspectives

# MODEL SELECTION FRAMEWORK

- **INPUTS:**

- $n$  examples (e.g. microarrays)
- $p$  variables (e.g. genes)
- $X : n \times p$  design matrix (e.g. gene expression dataset)
- $Y : n \times 1$  binary response vector (e.g. phenotype to predict)

- **OUTPUTS** (that we hope for):

- Relevant features for discriminating against the two possible phenotype's status, i.e. **good accuracy**
- **Stable** signature both across inner perturbations of a dataset and many datasets
- Genes **connected** on the graph

# L1-PENALIZED CLASSIFIERS

- **Lasso** : selects *genes* (*Tibshirani, 1996*)

$$\beta^{Lasso} = \arg \min_{\beta \in \mathbb{R}^p} \sum_{i=1}^n L(x_i \beta, y_i) + \lambda \|\beta\|_1$$

- **Group Lasso** (*Yuan & Lin, 2006*): implies *group sparsity* for groups of covariates that form a partition of  $\{1 \dots p\}$
- **Overlapping group Lasso** (*Jacob et al., 2009*): selects a union of *potentially overlapping* groups of covariates (e.g. gene pathways).
- **Graph Lasso**: uses groups induced by the graph (e.g. edges, connected components)

# L1-PENALIZED CLASSIFIERS

- **Lasso** : selects *genes* (Tibshirani, 1996)

$$\beta^{Lasso} = \arg \min_{\beta \in \mathbb{R}^p} \sum_{i=1}^n L(x_i \beta, y_i) + \lambda \|\beta\|_1$$

- **Group Lasso** (Yuan & Lin, 2006): implies *group sparsity* for groups of covariates that form a partition of  $\{1 \dots p\}$
- **Overlapping group Lasso** (Jacob et al., 2009): selects a union of *potentially overlapping* groups of covariates (e.g. gene pathways).
- **Graph Lasso**: uses groups induced by the graph (e.g. edges, connected components)

# L1-PENALIZED CLASSIFIERS

- **Lasso** : selects *genes* (*Tibshirani, 1996*)

$$\beta^{Lasso} = \arg \min_{\beta \in \mathbb{R}^p} \sum_{i=1}^n L(x_i \beta, y_i) + \lambda \|\beta\|_1$$

- **Group Lasso** (*Yuan & Lin, 2006*): implies *group sparsity* for groups of covariates that form a partition of  $\{1 \dots p\}$
- **Overlapping group Lasso** (*Jacob et al., 2009*): selects a union of *potentially overlapping* groups of covariates (e.g. gene pathways).
- **Graph Lasso**: uses groups induced by the graph (e.g. edges, connected components)

# L1-PENALIZED CLASSIFIERS

- **Lasso** : selects *genes* (Tibshirani, 1996)

$$\beta^{Lasso} = \arg \min_{\beta \in \mathbb{R}^p} \sum_{i=1}^n L(x_i \beta, y_i) + \lambda \|\beta\|_1$$

- **Group Lasso** (Yuan & Lin, 2006): implies *group sparsity* for groups of covariates that form a partition of  $\{1 \dots p\}$
- **Overlapping group Lasso** (Jacob et al., 2009): selects a union of *potentially overlapping* groups of covariates (e.g. gene pathways).
- **Graph Lasso**: uses groups induced by the graph (e.g. edges, connected components)

# PROPERTIES OF LASSO-LIKE ALGORITHMS

- **Advantages :**

- Do well when the number of features greatly exceeds the sample size, i.e.  $p \gg n$
- Relatively easy to implement. Quite fast to run.

- **Drawbacks :**

- Dependency on a parameter  $\lambda$  *to choose*: tradeoff between accuracy and no overfitting
- Bad behaviour in the presence of *too correlated* features : false positives and false negatives. Also implies great instability.

# PROPERTIES OF LASSO-LIKE ALGORITHMS

- **Advantages :**

- Do well when the number of features greatly exceeds the sample size, i.e.  $p \gg n$
- Relatively easy to implement. Quite fast to run.

- **Drawbacks :**

- Dependency on a parameter  $\lambda$  *to choose*: tradeoff between accuracy and no overfitting
- Bad behaviour in the presence of *too correlated* features : false positives and false negatives. Also implies great instability.

# EXAMPLE

- Groups 1 and 2 are very correlated
- The Group Lasso algorithm might choose one or the other at random
- **Scenario 1:** Both are relevant. But only one will be selected.
- **Scenario 2:** Group 1 is relevant, group 2 is noise. Roughly 50% probability that only group 2 is selected

# Outline

- 1 Motivation
  - Gene expression signatures
  - Mathematical tools for model selection
- 2 **Stabilizing the signature**
  - **Main procedure**
  - Scoring
- 3 Results
- 4 Conclusion and Perspectives

# TAKE ADVANTAGE OF RANDOMIZATION

Basis: *Meinshausen & Buehlmann, 2009* : Stability Selection.

- Simulate different datasets by perturbing the data, i.e. do a 100 times as follows
  - 1 Randomly choose  $n/2$  examples from the data (without replacement)
  - 2 Run the whole path of the graph lasso
  - 3 Store the selected groups
- When done: for each  $\lambda$  compute each group's **selection frequency**, i.e. get something like:

Groups	$\lambda_1$ (the largest)	.....	$\lambda_L$ (the smallest)
1	0.25	.....	0.6
...	...	.....	.....
p	0.65	.....	0.96

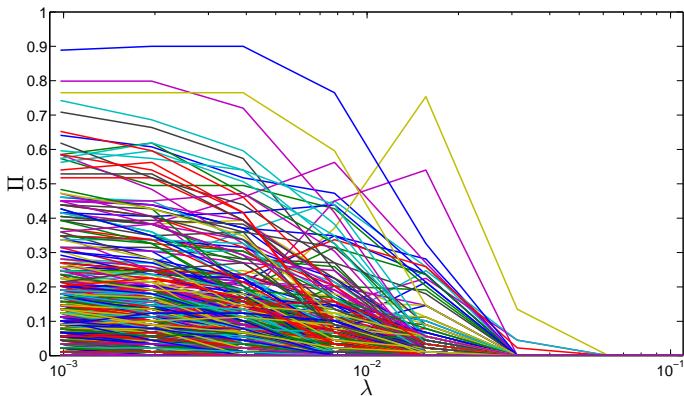
# TAKE ADVANTAGE OF RANDOMIZATION

Basis: *Meinshausen & Buehlmann, 2009* : Stability Selection.

- Simulate different datasets by perturbing the data, i.e. do a 100 times as follows
  - 1 Randomly choose  $n/2$  examples from the data (without replacement)
  - 2 Run the whole path of the graph lasso
  - 3 Store the selected groups
- When done: for each  $\lambda$  compute each group's **selection frequency**, i.e. get something like:

Groups	$\lambda_1$ (the largest)	.....	$\lambda_L$ (the smallest)
1	0.25	.....	0.6
...	...	.....	.....
p	0.65	.....	0.96

# GRAPHICAL ILLUSTRATION

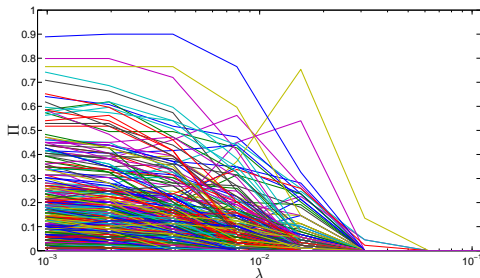


# Outline

- 1 Motivation
  - Gene expression signatures
  - Mathematical tools for model selection
- 2 Stabilizing the signature
  - Main procedure
  - Scoring
- 3 Results
- 4 Conclusion and Perspectives

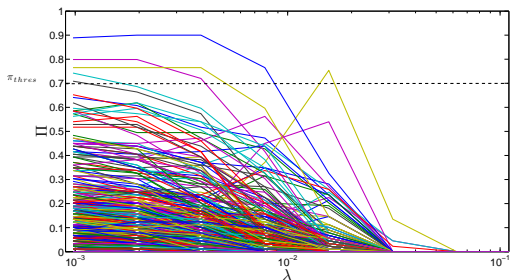
# SCORING THE GROUPS

- Initial scoring proposed in *Meinshausen & Buehlmann, 2009* : threshold.



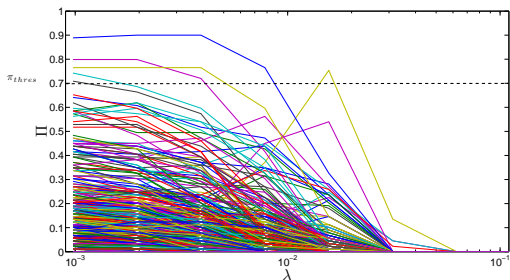
# SCORING THE GROUPS

- Initial scoring proposed in *Meinshausen & Buehlmann, 2009* : threshold.
- However : hard to choose a grid for  $\lambda$ .



# SCORING THE GROUPS

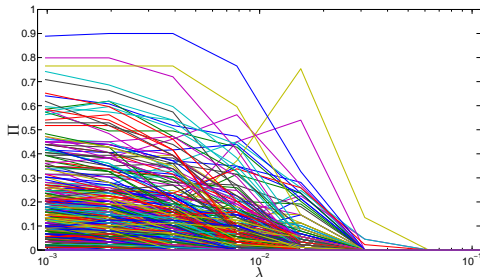
- Initial scoring proposed in *Meinshausen & Buehlmann, 2009* : threshold.
- However : hard to choose a grid for  $\lambda$ .



# SCORING THE GROUPS

For each  $\lambda$  we compute the frequency ratio of each group; we then keep the maximum value over the grid for each group, i.e. the score vector is defined as

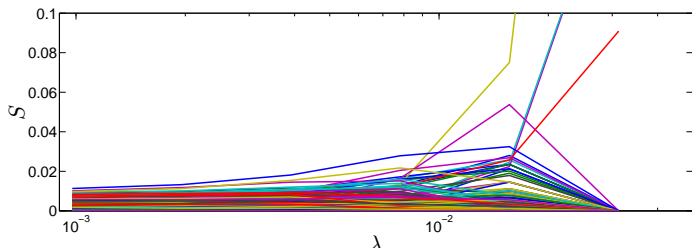
$$\forall j \in \text{Groups}, S_j = \max_{\lambda} \frac{p(j \in \text{Solution}|\lambda)}{\sum_j p(j \in \text{Solution}|\lambda)}$$



# SCORING THE GROUPS

For each  $\lambda$  we compute the frequency ratio of each group; we then keep the maximum value over the grid for each group, i.e. the score vector is defined as

$$\forall j \in \text{Groups}, S_j = \max_{\lambda} \frac{p(j \in \text{Solution}|\lambda)}{\sum_j p(j \in \text{Solution}|\lambda)}$$



# DATA AND OBJECTIVES

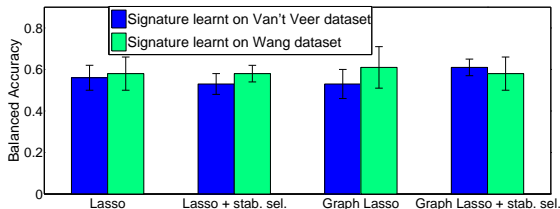
- Data:
  - Van't Veer dataset : 295 tumors, 78 metastatic, 8141 genes
  - Wang dataset : 286 tumors, 106 metastatic, 8141 genes
  - Graph (*Chuang et al., 2007*), 8141 nodes, 57235 edges
- Algorithms to compare:
  - Lasso
  - Graph Lasso (edges)
  - Lasso + stability selection
  - Graph Lasso + stability selection

# ACCURACY

- For a signature of 60 genes: balanced accuracy on Van't Veer data, five fold CV

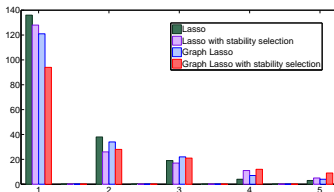
	No Stability selection	Stability Selection
Lasso	$0.61 \pm 0.03$	$0.57 \pm 0.02$
Graph Lasso	$0.62 \pm 0.02$	$0.58 \pm 0.03$

- Accuracy when tested on Wang data:

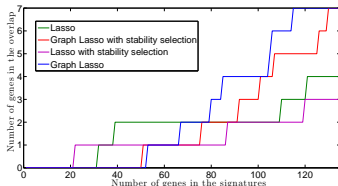


# STABILITY

- Inner stability:

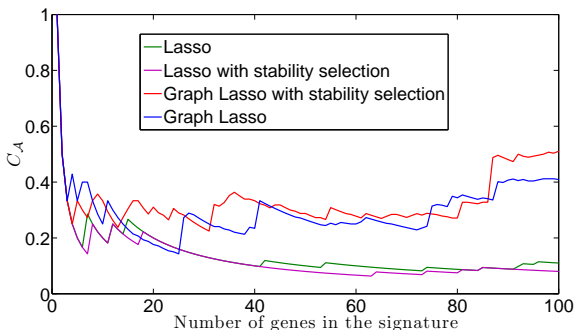


- Stability across datasets:

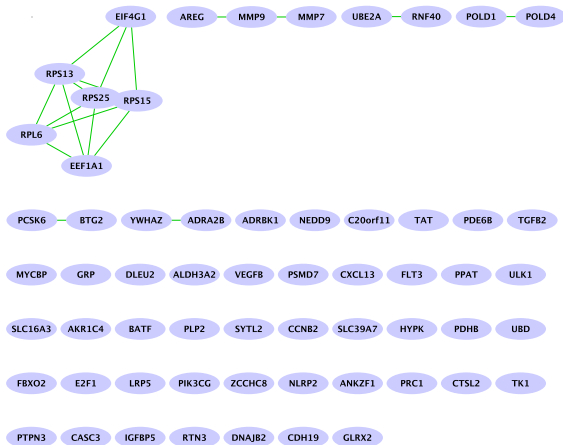


# CONNECTIVITY

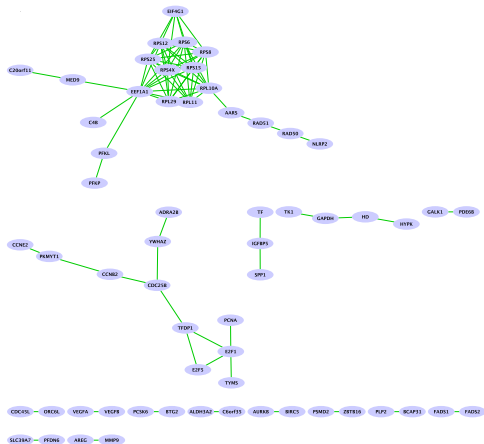
$$C_A = \frac{\text{Size of the largest connected component}}{\text{Number of genes selected}}$$



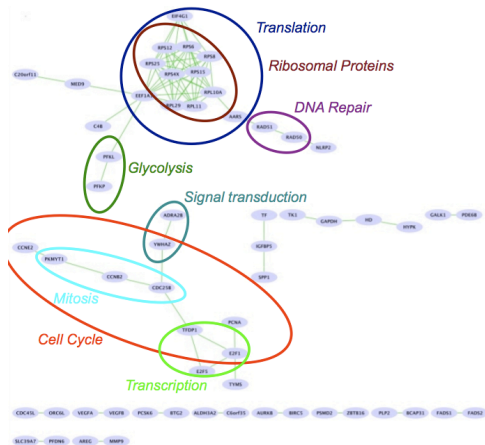
# SIGNATURE OBTAINED FROM LASSO



# SIGNATURE OBTAINED FROM GRAPH LASSO WITH STABILITY SELECTION



# SIGNATURE OBTAINED FROM GRAPH LASSO WITH STABILITY SELECTION



# CONCLUSION

- Selecting groups from a graph instead of genes:
  - Adds relevant biological information to the model
  - Increases **connectivity** and hence **interpretability** of the signature
  - Drawback: may become computationally more demanding (larger groups)
- Using stability selection:
  - Improves **stability** of the signature within a given dataset
  - Drawback: hard to know how many genes should be in the signature
- Neither of these methods or their combination change the accuracy

# PERSPECTIVES

- Take **subtypes of tumors** into account : need more data
- Related project (with F. Reyal): build a larger Breast Cancer dataset
- Try different groups/ Larger groups
- Compare biological processes involved instead of genes