Stability Selection for High-Dimensional Data

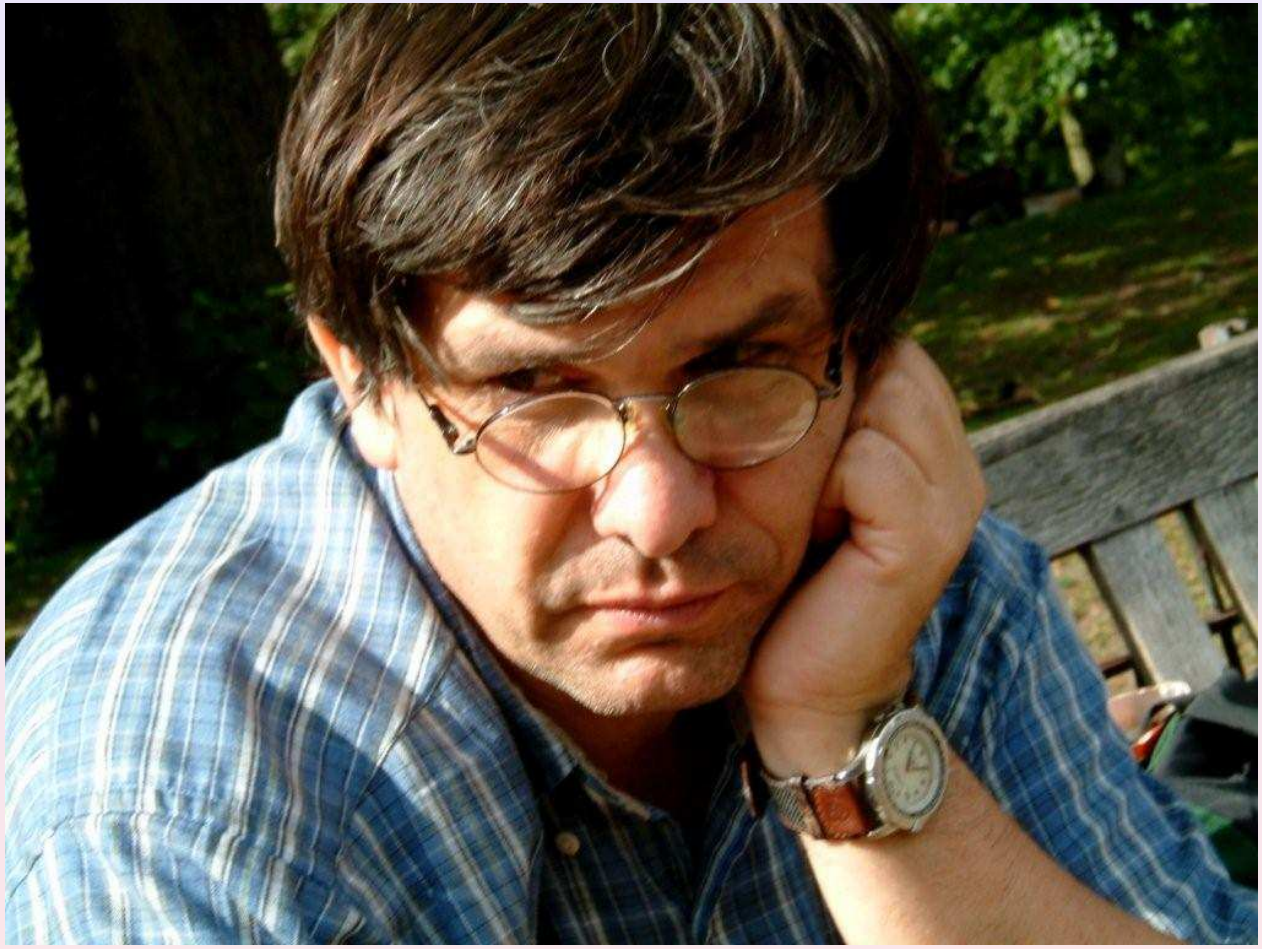
Peter Bühlmann
joint work with Nicolai Meinshausen

Seminar für Statistik, ETH Zürich

December 2008

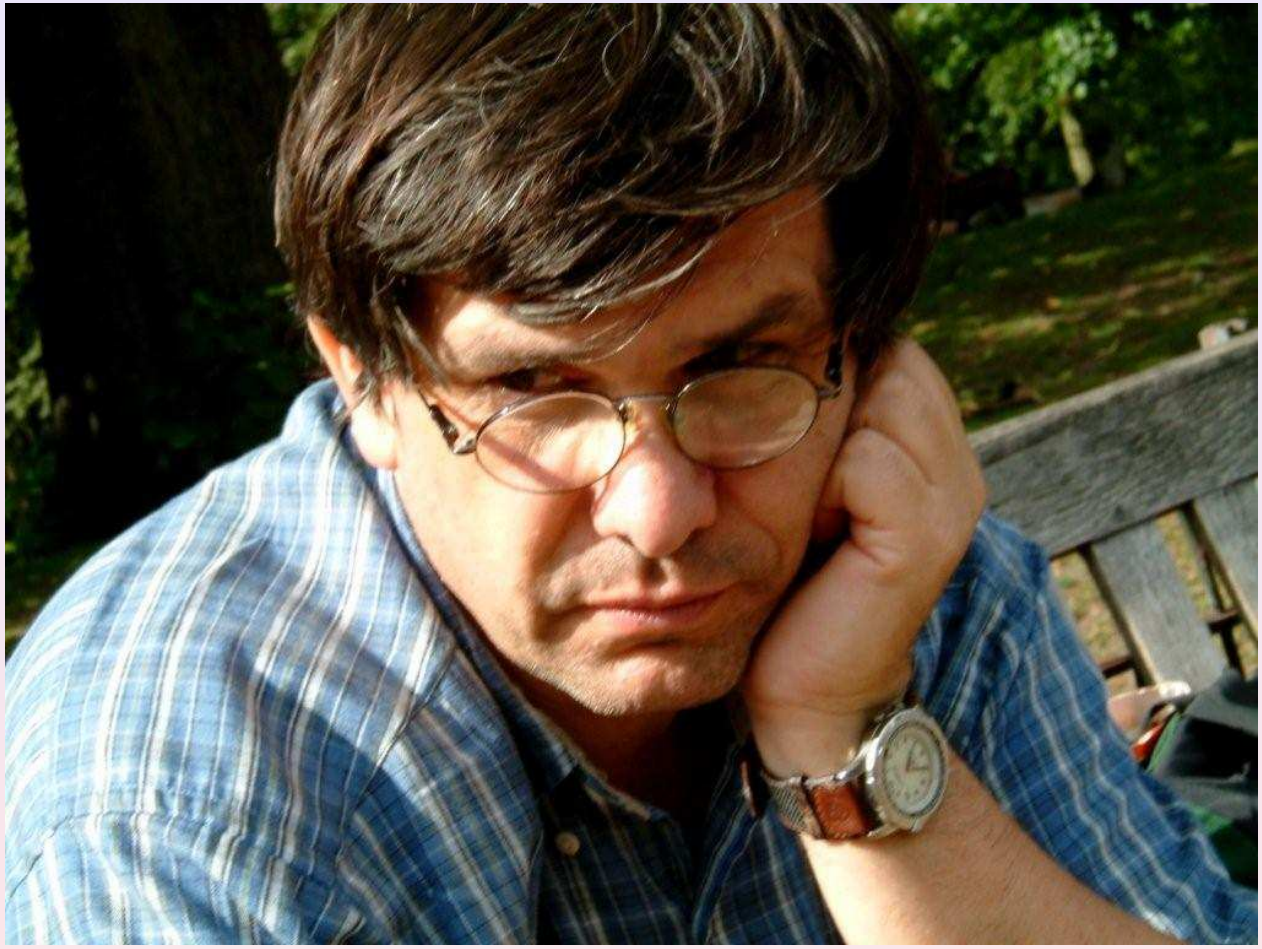
a highly stable object

he will not so easily change his mind...



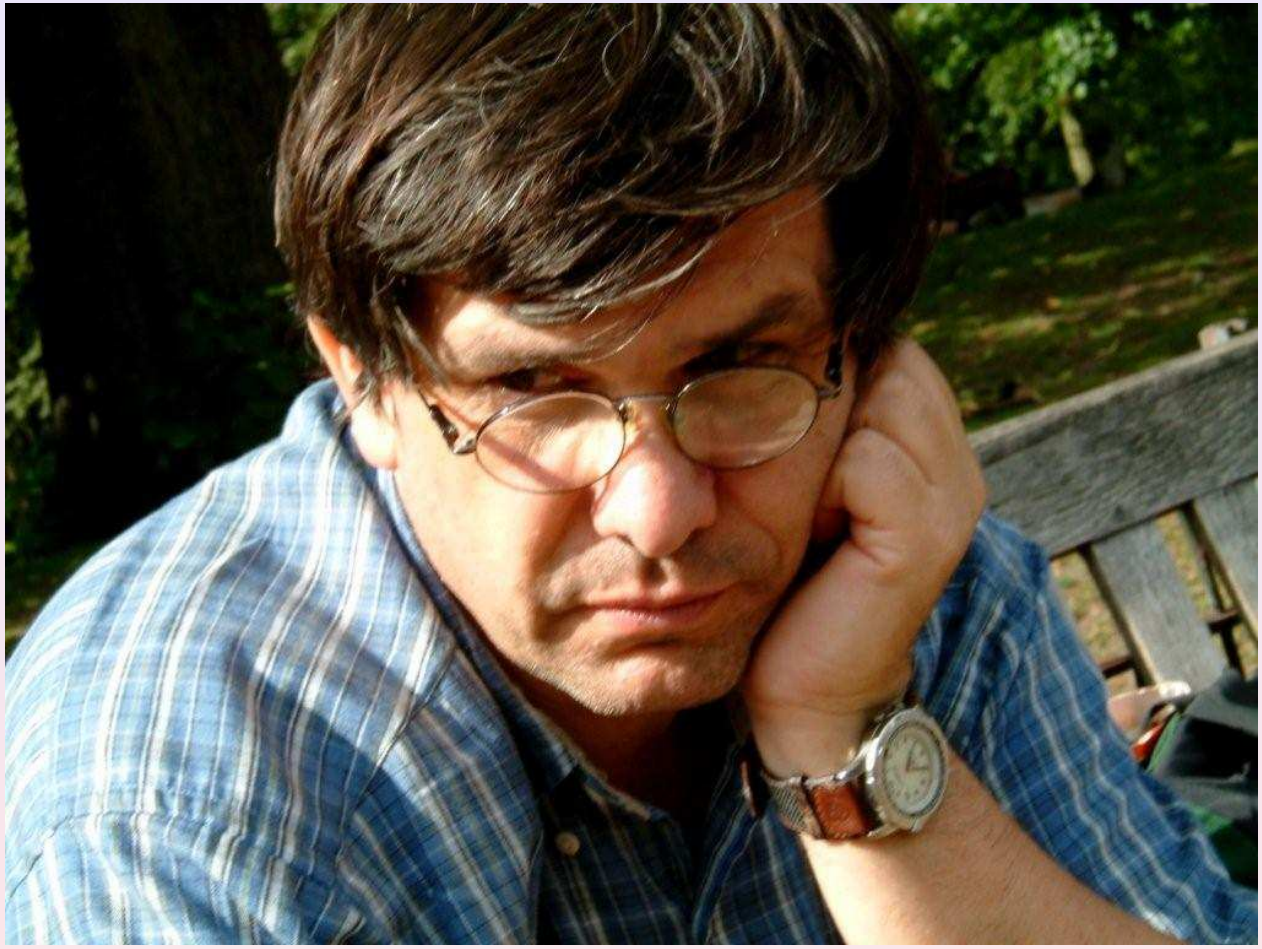
Congratulations to the Humboldt Prize!

a highly stable object
he will not so easily change his mind...



Congratulations to the Humboldt Prize!

a highly stable object
he will not so easily change his mind...



Congratulations to the Humboldt Prize!

- “better” detection of variables/components/factors
- how many variables/components

$\rightsquigarrow$ regularized estimation for discrete structures in high-dimensional settings
(which is more difficult than prediction)

Motif regression

for finding HIF1α transcription factor binding sites in DNA seq.

$Y_i \in \mathbb{R}$: univariate response measuring binding intensity on coarse DNA segment i (from CHIP-chip experiments)

$X_i = (X_i^{(1)}, \ldots, X_i^{(p)}) \in \mathbb{R}^p$:

$X_i^{(j)}$ = abundance score of candidate motif j in DNA segment i

variable selection in additive model $Y_i = \sum_{j=1}^{p} f_j(X_i^{(j)}) + \varepsilon_i$

$i = 1, \ldots, n = 287, \; p = 196$

$\rightsquigarrow$ a reasonable algorithm selects 26 covariates
i.e. 26 interesting candidate motifs
and hence report these findings to the biologists...

really?
do we trust our selection algorithm?
how stable are the findings?

Motif regression

for finding HIF1α transcription factor binding sites in DNA seq.

$Y_i \in \mathbb{R}$: univariate response measuring binding intensity on coarse DNA segment i (from CHIP-chip experiments)

$X_i = (X_i^{(1)}, \ldots, X_i^{(p)}) \in \mathbb{R}^p$:

$X_i^{(j)}$ = abundance score of candidate motif j in DNA segment i

variable selection in additive model $Y_i = \sum_{j=1}^{p} f_j(X_i^{(j)}) + \varepsilon_i$

$i = 1, \ldots, n = 287, \; p = 196$

$\rightsquigarrow$ a reasonable algorithm selects 26 covariates
i.e. 26 interesting candidate motifs
and hence report these findings to the biologists...

really?

do we trust our selection algorithm?

how stable are the findings?

Motif regression

for finding HIF1α transcription factor binding sites in DNA seq.

$Y_i \in \mathbb{R}$: univariate response measuring binding intensity on coarse DNA segment i (from CHIP-chip experiments)

$X_i = (X_i^{(1)}, \ldots, X_i^{(p)}) \in \mathbb{R}^p$:

$X_i^{(j)}$ = abundance score of candidate motif j in DNA segment i

variable selection in additive model $Y_i = \sum_{j=1}^{p} f_j(X_i^{(j)}) + \varepsilon_i$

$i = 1, \ldots, n = 287, \; p = 196$

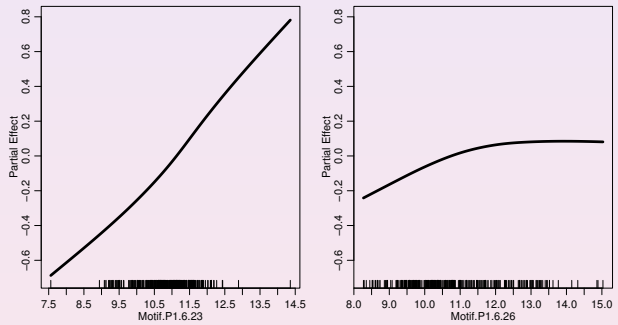
$\rightsquigarrow$ a reasonable algorithm selects 26 covariates
i.e. 26 interesting candidate motifs
and hence report these findings to the biologists...

really?
do we trust our selection algorithm?
how stable are the findings?

one simple idea: subsampling or bootstrapping

applied to HIF1α transcription factor binding site problem (Müller, Meier, PB & Ricci, 2009):

- sparse additive modeling with $n = 287$, $p = 196$
- keep the variables which are selected in more than $\pi_{\text{thr}} \cdot 100\%$ subsampling runs
- report these variables $\rightsquigarrow$ only 2 "stable" findings ($\neq 26$)



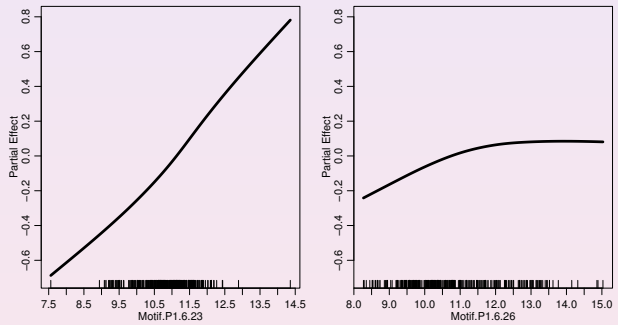
right panel: variable corresponds to a true, known motif
left panel: good additional support for relevance (nearness to transcriptional start-site of important genes, ...)
ongoing biological validation with Ricci lab (ETH Zurich)

how to choose the threshold π_{thr}?

one simple idea: subsampling or bootstrapping

applied to HIF1α transcription factor binding site problem (Müller, Meier, PB & Ricci, 2009):

- sparse additive modeling with $n = 287$, $p = 196$
- keep the variables which are selected in more than $\pi_{\text{thr}} \cdot 100\%$ subsampling runs
- report these variables $\rightsquigarrow$ only 2 "stable" findings ($\neq 26$)



right panel: variable corresponds to a true, known motif
left panel: good additional support for relevance (nearness to transcriptional start-site of important genes, ...)
ongoing biological validation with Ricci lab (ETH Zurich)

how to choose the threshold π_{thr}?

- subsampling/bootstrapping in the context of selection yields better results
- our approach is mathematically justified:
  we will be able to achieve finite sample control for the number of false positives; or of familywise error rates

$\rightsquigarrow$ marriage of high-dimensional selection algorithms and error control

Stability selection

consider (first) linear model setting

$$Y_i = (\beta_0) + \sum_{j=1}^{p} \beta_j X_i^{(j)} + \varepsilon_i, \ i = 1, \ldots, n \ (\ll p)$$

set of active variables: $S = \{j; \ \beta_j \neq 0\}$

variable selection procedure:

$$\hat{S}^\lambda \subseteq \{1, \ldots, p\},$$

λ a tuning parameter

prime example: Lasso (Tibshirani, 1996)

$$\hat{\beta}^\lambda = \text{argmin}_\beta (n^{-1} \| Y - X\beta \|^2 + \underbrace{\lambda}_{\geq 0} \underbrace{\|\beta\|_1}_{\sum_{j=1}^{p} |\beta_j|})$$

$$\hat{S}^\lambda = \{j; \ \hat{\beta}_j^\lambda \neq 0\} \quad \text{(no significance testing)}$$

subsampling:

- draw sub-sample of size $[n/2]$ without replacement, denoted by $I^* \subseteq \{1, \ldots, n\}$, $|I^*| = [n/2]$
- run the selection algorithm $\hat{S}^\lambda(I^*)$ on I^*
- do these steps many times and compute the relative selection frequencies

$$\hat{\Pi}_j^\lambda = P^*(j \in \hat{S}^\lambda(I^*)),\ j = 1, \ldots, p$$

P^* is w.r.t. finite number of sub-sampling (and maybe other sources of randomness if a randomized selection algorithm is invoked)

could also use bootstrap sampling with replacement...

subsampling:

- draw sub-sample of size $[n/2]$ without replacement, denoted by $I^* \subseteq \{1, \ldots, n\}$, $|I^*| = [n/2]$
- run the selection algorithm $\hat{S}^\lambda(I^*)$ on I^*
- do these steps many times and compute the relative selection frequencies

$$\hat{\Pi}_j^\lambda = P^*(j \in \hat{S}^\lambda(I^*)),\ j = 1, \ldots, p$$

P^* is w.r.t. finite number of sub-sampling (and maybe other sources of randomness if a randomized selection algorithm is invoked)

could also use bootstrap sampling with replacement...

typically, we consider many regularization parameters:

$$\{\hat{S}^\lambda;\ \lambda \in \Lambda\}$$

Λ can be discrete (this is what you compute), a singleton or continuous

stability selection:

$$\hat{S}^{\text{stable}} = \{j;\ \max_{\lambda\in\Lambda} \hat{\Pi}_j^\lambda \geq \pi_{\text{thr}}\}$$

choice of π_{thr}?

How to choose the threshold π_{thr}?

Let

$$\hat{S}^{\Lambda} = \cup_{\lambda \in \Lambda} \hat{S}^{\lambda} \approx \hat{S}^{\lambda_{min}}$$

$q_{\Lambda} = E^*[|\hat{S}^{\Lambda}(I^*)|]$ which is easily computable

denote by $V = |S^C \cap \hat{S}^{\text{stable}}|$ = number of false positives

Theorem (Meinshausen & PB, 2008)
main assumption: exchangeability condition
in addition: $\hat{S}^{\lambda}$ has to be better than "random guessing"
Then:

$$E(V) \leq \frac{1}{2\pi_{\text{thr}} - 1} \frac{q_{\Lambda}^2}{p}$$

i.e. finite sample control, even if $p \gg n$
$\rightsquigarrow$ choose threshold π_{thr} to control e.g. $E[V] \leq 1$ or $P[V > 0] \leq \alpha$

note the generality of the Theorem...

- it works for any method which is better than “random guessing”
- it works not only in regression but also in “any” discrete structure estimation problem (whenever there is a include/exclude decision)
  $\rightsquigarrow$ variable selection, graphical modeling, clustering, ...

and hence there must be a fairly strong condition...
Exchangeability condition:
the distribution of $\{I_{\{j \in \hat{S}^\lambda\}};\ j \in S^C\}$ is exchangeable
note: only some requirement for noise variables

for specific problems, we can prove finite sample control under weaker assumptions ...

note the generality of the Theorem...

- it works for any method which is better than “random guessing”
- it works not only in regression but also in “any” discrete structure estimation problem (whenever there is a include/exclude decision)
  $\rightsquigarrow$ variable selection, graphical modeling, clustering, ...

and hence there must be a fairly strong condition...
Exchangeability condition:
the distribution of $\{I_{\{j \in \hat{S}^\lambda\}};\ j \in S^C\}$ is exchangeable
note: only some requirement for noise variables

for specific problems, we can prove finite sample control under weaker assumptions ...

Example:

linear model with random design

$\{X^{(j)};\ j \in S^C\}$ are exchangeable,

e.g. $\mathrm{Corr}(X^{(j)}, X^{(k)}) = \rho$ for all $j, k \in S^C$

Some numerical experiments

Variable selection in linear models using Lasso

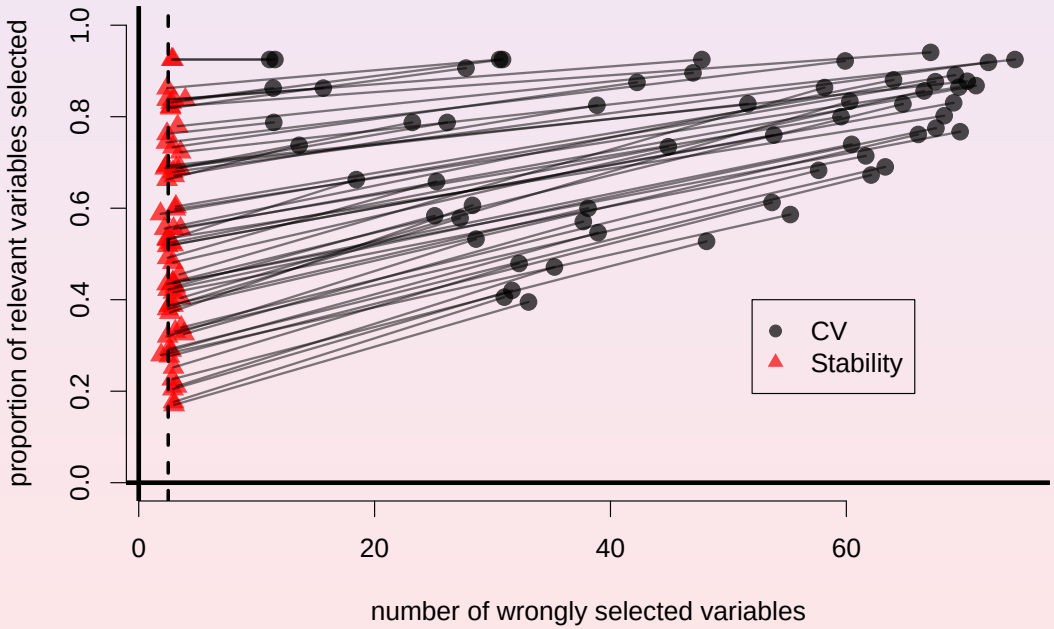
a range of scenarios:

$p = 660$ with design from a real data set about motif regression

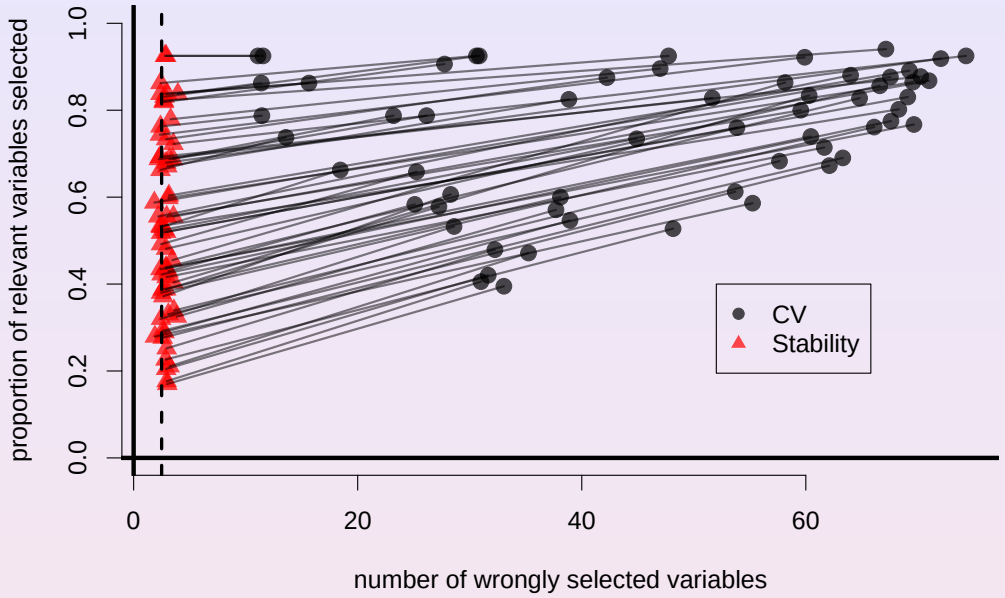
$n \in \{450, 750\}$, sparsity $s \in \{4, 8, 12, \ldots, 40\}$

signal to noise ratio (SNR) $\in \{0.25, 1, 4\}$

control for $E[V] = 2.5$



control for $E[V] = 2.5$



stability selection yields:

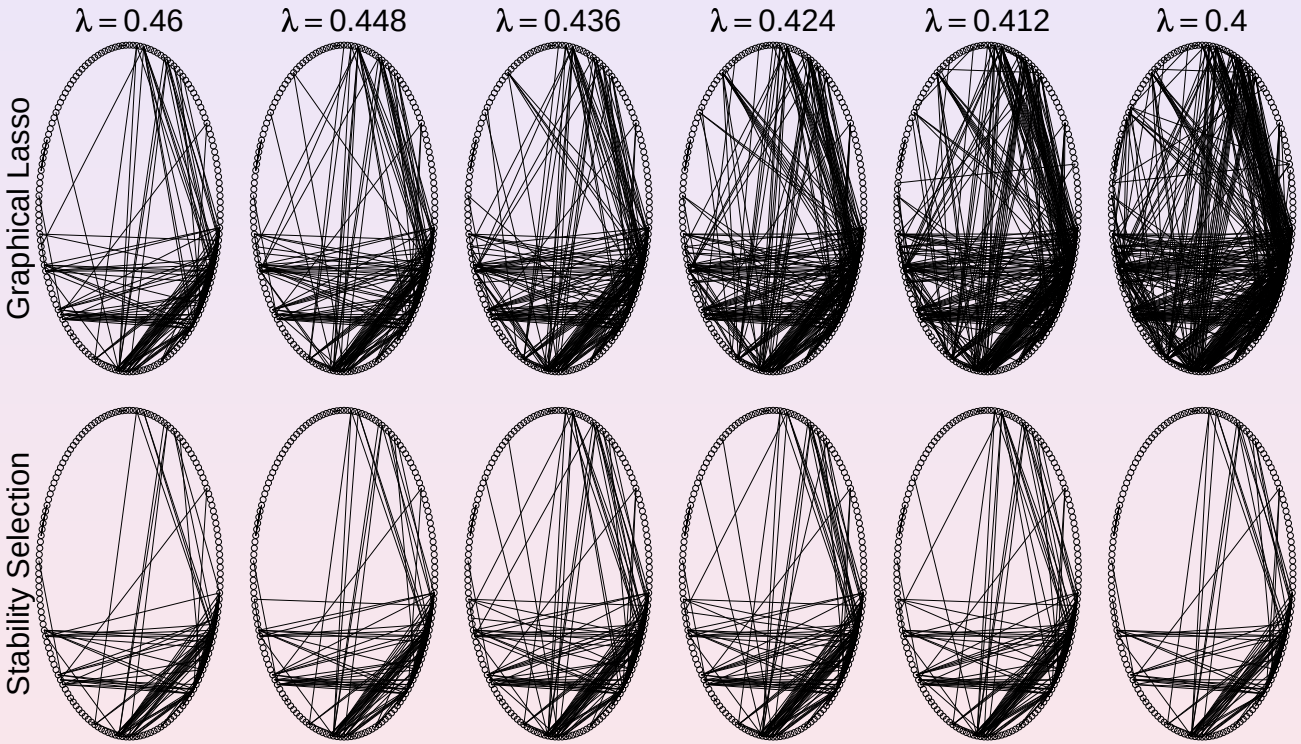
- accurate control (as proved in theory)
- drastic reduction of false positives in comparison to CV-tuned solution
- not much loss in terms of power (true positives)

Graphical modeling using GLasso

infer conditional independence graph using ℓ_1-penalization

(Rothman, Bickel, Levina & Zhu, 2008; Friedman, Hastie & Tibshirani, 2008)

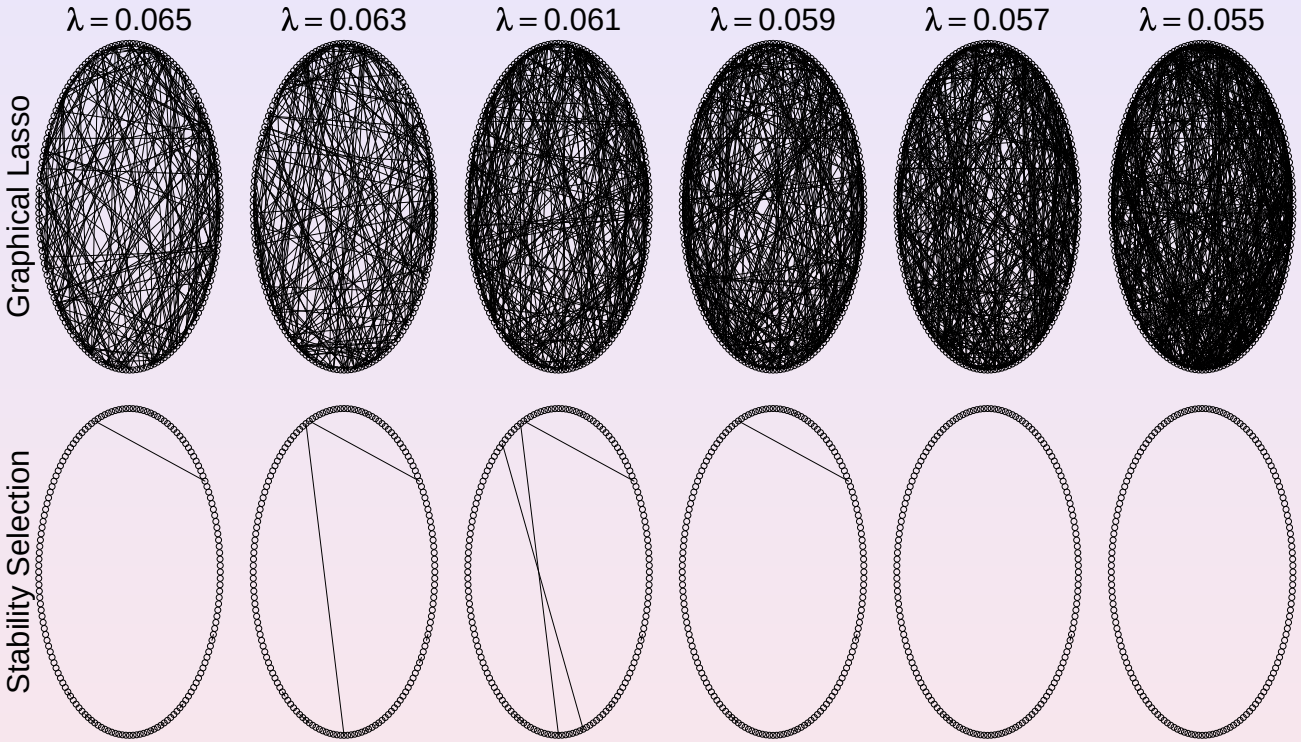
varying the regularization parameter ($p = 160,\ n = 115$)



with stability selection: choice of initial λ-tuning parameter does not matter much (as proved by our theory)

just need to fix the finite-sample control

varying the regularization parameter for the null-case

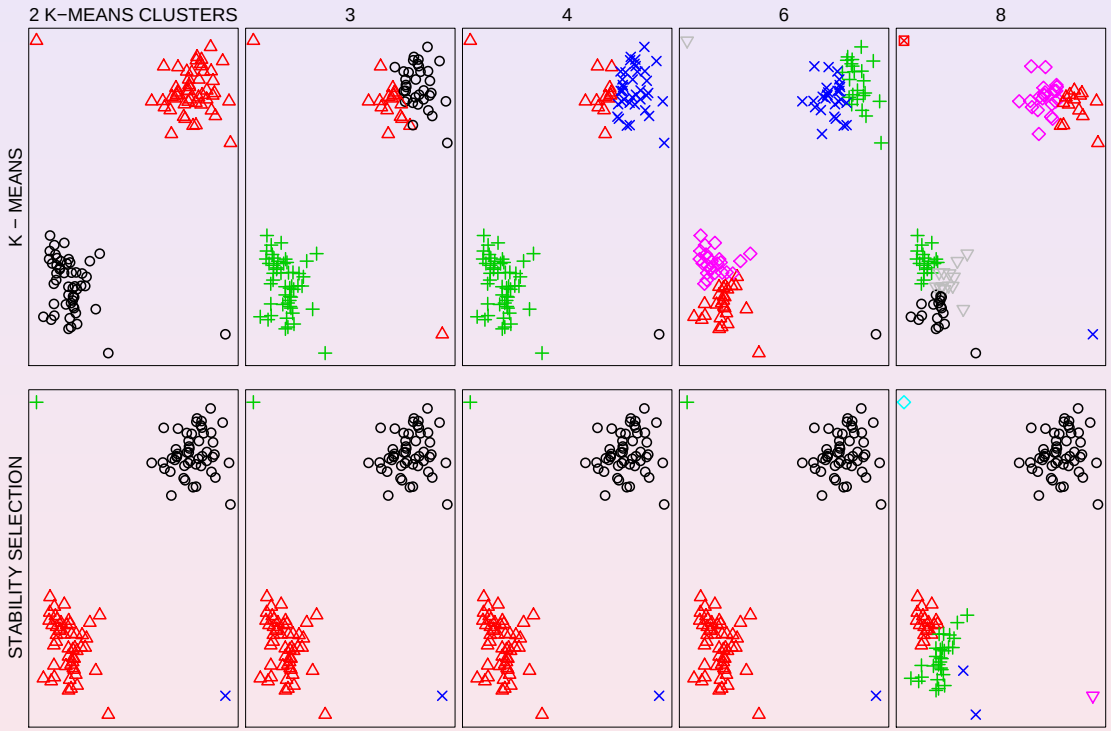


with stability selection: the number of false positives are indeed controlled (as proved by our theory)

Clustering

truth: 2 clusters plus 2 “outlier-singletons” = 4 clusters

varying the number of clusters in K-means



with stability selection: for reasonable size of K’s, we can find the true structure

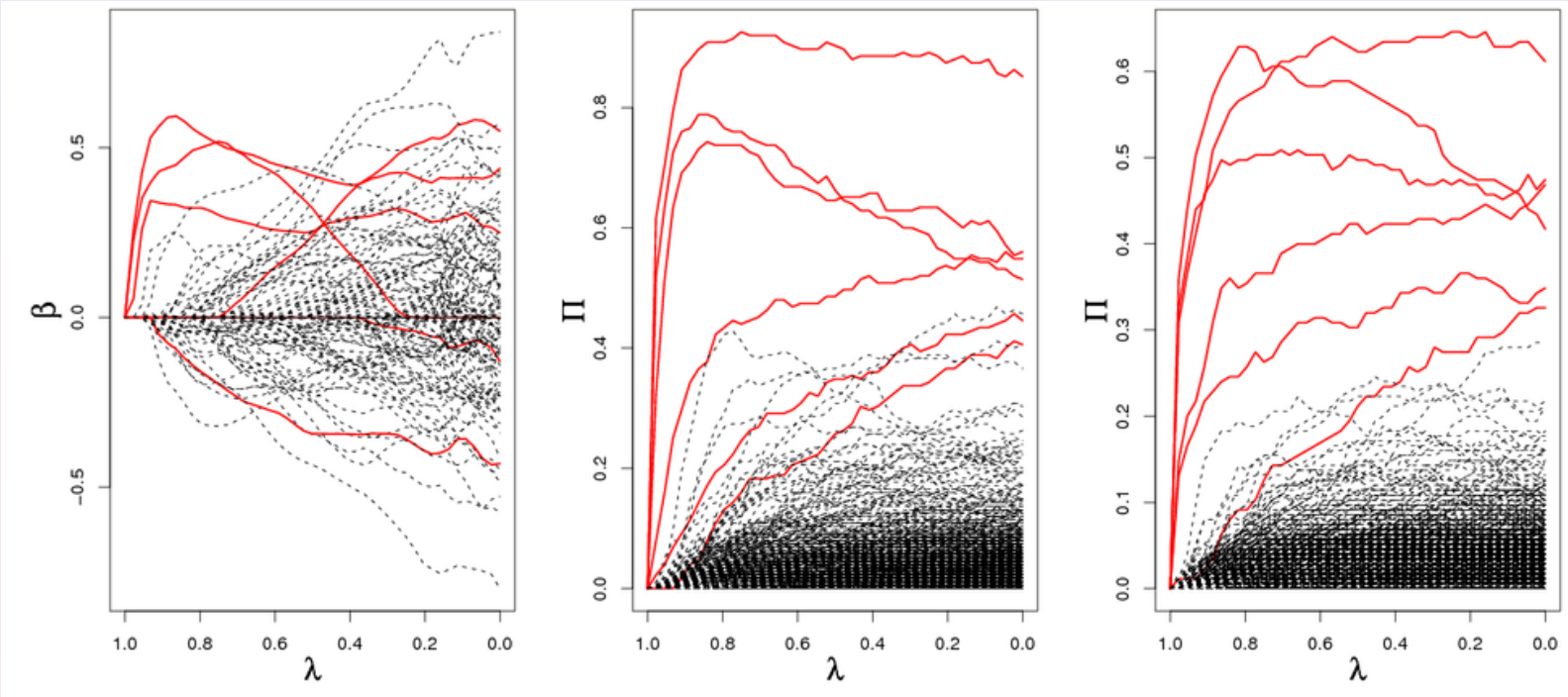
The Lasso and its corresponding stability path

Riboflavin production in Bacillus Subtilis

$p = 4088$ gene expressions, $n = 115$

sparsity s "=" 6 (6 "true" genes; all other variables permuted)

Lasso stab.sel. stab.sel. rand.Lasso



with stability selection: the 4-6 "true" variables are sticking out much more clearly from noise covariates

stability selection cannot be reproduced by simply selecting the right penalty with Lasso

stability selection provides a fundamentally new solution

Improved variable selection: additional randomization

consider the Lasso for linear models

variable selection consistency $P[\hat{S}^\lambda = S] \to 1$ $(n \to \infty)$
$\Leftrightarrow$
neighborhood stability condition holds
(Meinshausen & PB, 2004)
$\Leftrightarrow$
irrepresentable condition holds (Zhao & Yu, 2006)
$\max_{j \in S^c} |\text{sign}(\beta_S)^T (X_S^T X_S)^{-1} X_S^T X_j| \leq 1 - \gamma$

i.e. restrictive condition on the design

assuming irrepresentable condition:
variable selection consistency with stability selection holds as well (no surprise)
from a practical view: insensitivity w.r.t. choice of λ
and an "objective choice" for regularization (choice of π_{thr})

Improved variable selection: additional randomization

consider the Lasso for linear models

variable selection consistency $P[\hat{S}^\lambda = S] \to 1$ $(n \to \infty)$
$\Leftrightarrow$
neighborhood stability condition holds
(Meinshausen & PB, 2004)
$\Leftrightarrow$
irrepresentable condition holds (Zhao & Yu, 2006)
$\max_{j \in S^c} |\text{sign}(\beta_S)^T (X_S^T X_S)^{-1} X_S^T X_j| \leq 1 - \gamma$

i.e. restrictive condition on the design

assuming irrepresentable condition:
variable selection consistency with stability selection holds as well (no surprise)
from a practical view: insensitivity w.r.t. choice of λ
and an "objective choice" for regularization (choice of π_{thr})

from a theory and practical perspective:
we can improve by additional randomization

Randomized Lasso (Meinshausen & PB, 2008)

$W_1, \ldots W_p$ i.i.d. r.v. $\in [\alpha, 1]$
$\alpha \in (0, 1]$ is called the weakness parameter
e.g. two-values: $P(W_j = \alpha) = p_w,\ P(W_j = 1) = 1 - p_w$

$$\hat{\beta}^{\lambda, W} = \text{argmin}_\beta (n^{-1} \| Y - X\beta \|^2 + \lambda \sum_{j=1}^{p} \frac{|\beta_j|}{W_j})$$

i.e. a weighted Lasso aka adaptive Lasso (Zou, 2006)
but here the weights are totally random
$\rightsquigarrow$ non-sensical procedure as a stand-alone!

but in connection with stability selection, this improves upon Lasso

Randomized Lasso (Meinshausen & PB, 2008)

$W_1, \ldots W_p$ i.i.d. r.v. $\in [\alpha, 1]$
$\alpha \in (0, 1]$ is called the weakness parameter
e.g. two-values: $P(W_j = \alpha) = p_w,\ P(W_j = 1) = 1 - p_w$

$$\hat{\beta}^{\lambda, W} = \text{argmin}_\beta (n^{-1} \| Y - X\beta \|^2 + \lambda \sum_{j=1}^{p} \frac{|\beta_j|}{W_j})$$

i.e. a weighted Lasso aka adaptive Lasso (Zou, 2006)
but here the weights are totally random
$\rightsquigarrow$ non-sensical procedure as a stand-alone!

but in connection with stability selection, this improves upon Lasso

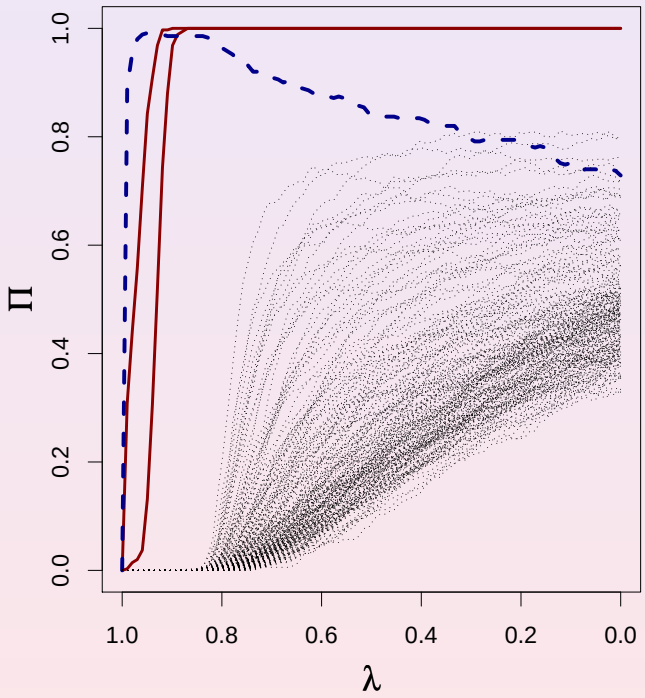
artificial example where irrepresentable condition fails to hold

2 active variables (in red)

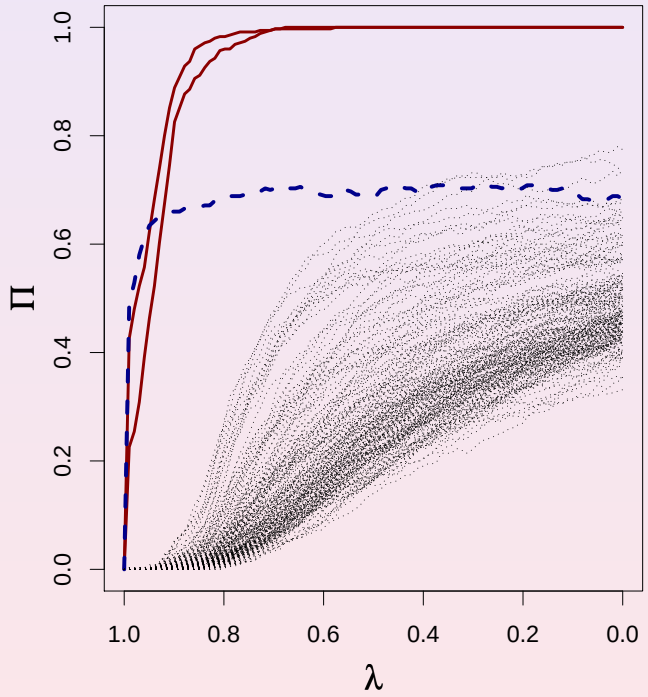
1 variable which violates IR condition (in blue)

197 other noise variables (in black); n=200

stab. sel. with Lasso



stab. sel. with rand. Lasso ($\alpha = 0.2$)



stability selection with randomized Lasso:

recovers the true set of variables

for more realistic examples:

3 – 70% fewer false positives using stability selection with randomized Lasso

Theorem (Meinshausen & PB, 2008)
Consider the randomized Lasso with weakness $\alpha \in (0, 1)$
Assumptions:

- fixed design satisfying a sparse eigenvalue condition somewhat stronger than in
  Bickel, Ritov & Tsybakov, 2007; (for $\|\hat{\beta} - \beta\|_1 \to 0$)
  but weaker than necessary and sufficient "neighborhood stability"/"irrepresentable condition" for Lasso var. select.
- $\Lambda = \{\lambda; \lambda \geq \lambda_{min}\}$: $\lambda_{min} \asymp s\sqrt{\log(p)/n}$
- $\min\{|\beta_j|;\ \beta_j \neq 0\} \geq Cs^{3/2}\lambda_{min}$
  not as tight as it could be
- $p \gg n$ is allowed

Then, for suitable π_{thr},

$$P(\hat{S}^{\text{stable}}(\pi_{\text{thr}}) = S) \geq 1 - \frac{5}{p} \to 1 \quad (n \ll p \to \infty)$$

Why does it work...

Randomized Lasso = Lasso for re-scaled covariates:
$X_W^{(j)} \leftarrow W_j X^{(j)}$

- condition number of re-scaled design becomes worse as $\alpha \to 0 \rightsquigarrow$ respect lower bound $\alpha_{low} \leq \alpha$

  for such $\alpha \geq \alpha_{low}$: relevant variables are still chosen
  and they will be selected in a stable way

- irrepresentable condition is more likely to hold for small α

  irrepresentable condition is "sometimes" (on set $\mathcal{B}$) fulfilled
  under randomly sampled weights
  (even when it is not fulfilled for the original data)
  $\rightsquigarrow$ noise covariates are "sometimes" (on set $\mathcal{B}$)
  not selected
  $P_W(\mathcal{B})$ is not too small ($\geq p_w(1-p_w)^{Cs^2}$)
  and hence the noise covariates cannot be stably selected

Conclusions

Stability Selection is applicable in a wide range of problems, and it is mainly useful in high-dimensional problems

- stability selection is very generic for finite-sample error control when estimating discrete structures,
  e.g. variable selection, graphical modeling, clustering
- additional randomization can help, e.g. Randomized Lasso

  we proved a similar result on randomized weak orthogonal matching pursuit
  randomized weakening (e.g. randomly high penalty) seems to be useful in greater generality (aka Random Forests)