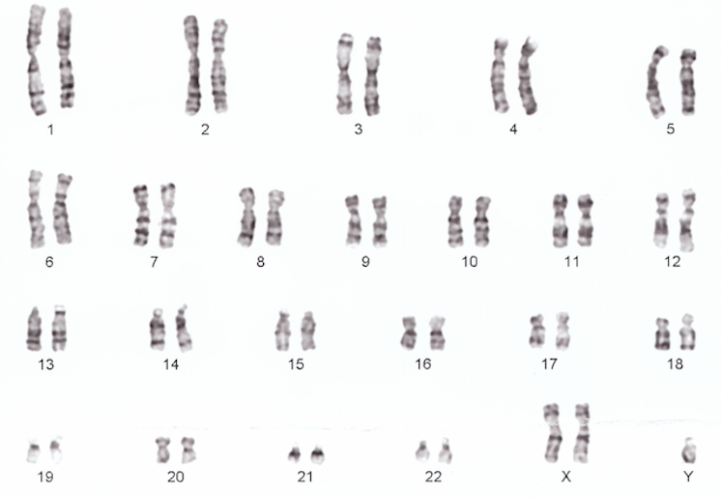
Learning from genomic data: efficient representations and algorithms

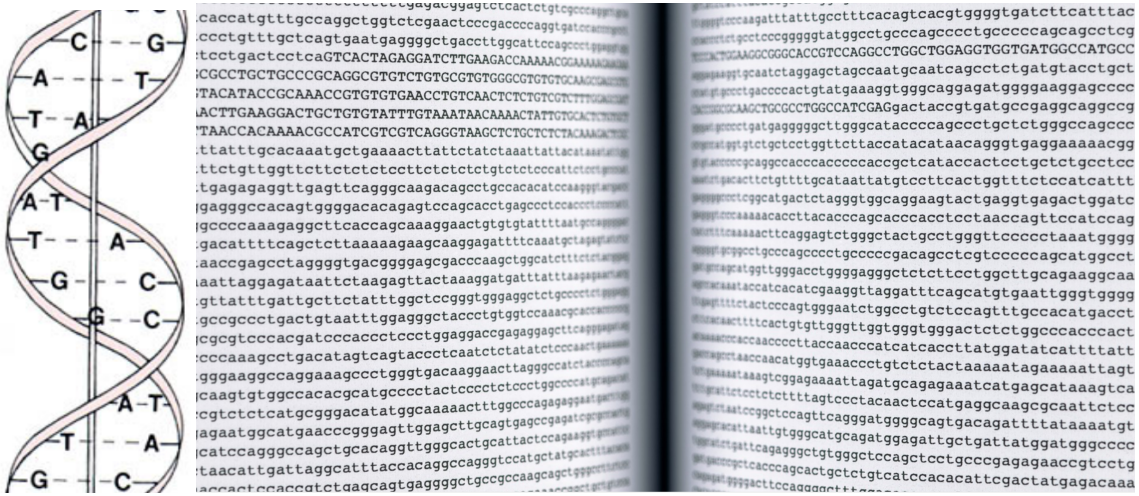
Marine Le Morvan
Supervised by **Jean-Philippe Vert** & **Andrei Zinoyev**

CBIO - Mines Paristech, INSERM U900 - Curie institute, Paris, France

July 3rd, 2018

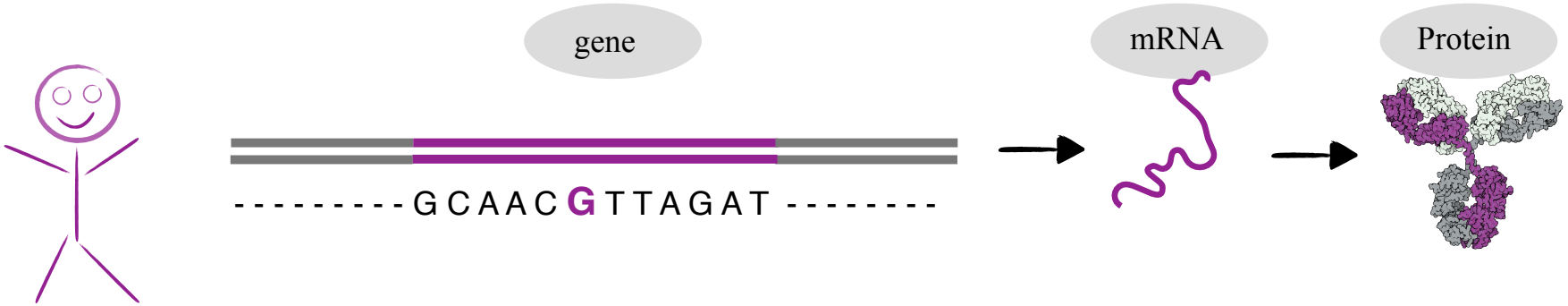
The human genome





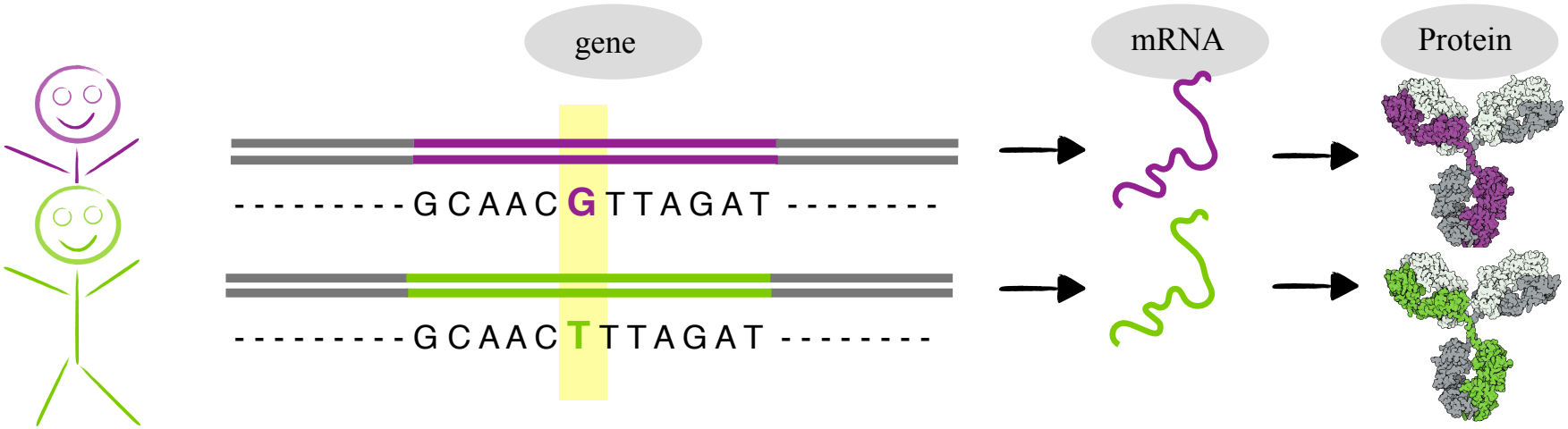
- A total of 3 billions nucleotides (*Les Misérables* - V. Hugo - $\times 1000$).
- First version of the human reference genome completed in 2003.
- Today, a whole human genome can be sequenced in a day for 1000$.
- That makes large scale sequencing efforts affordable.

Nucleotide variations and mutations

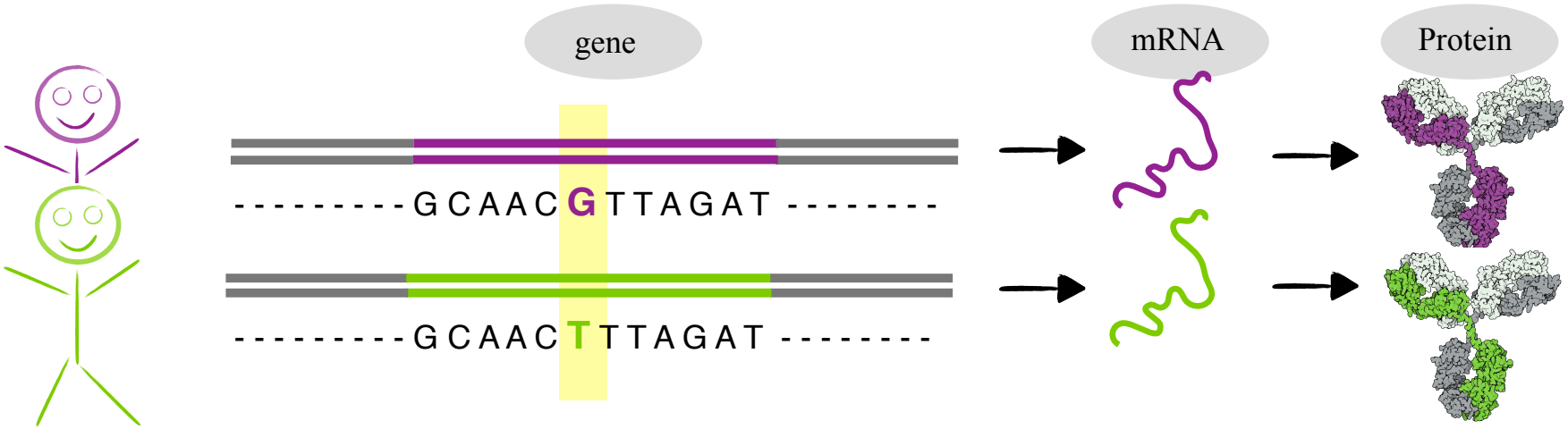


Nucleotide variations and mutations

Nucleotide variations and mutations



Nucleotide variations and mutations

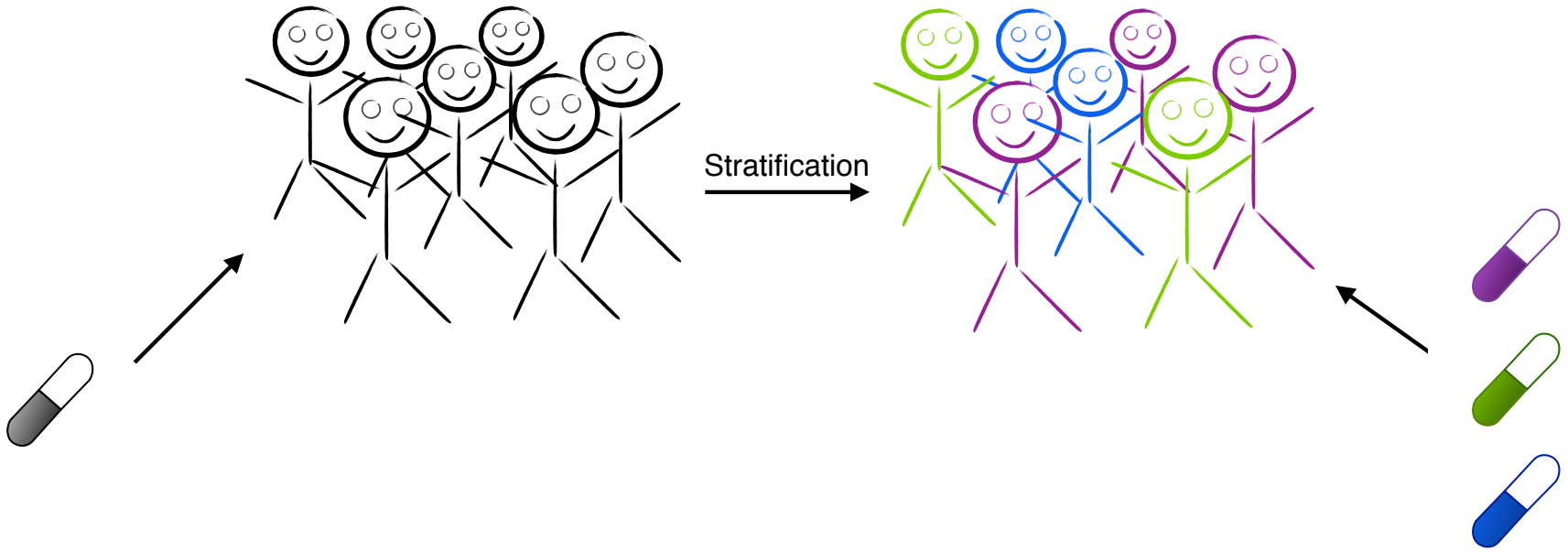


- Single Nucleotide Polymorphism (SNP):
  - ✓ Variation compared to a reference genome in typically $> 1\%$ of the population.
- Germline mutation:
  - ✓ Variation compared to a reference genome in typically $< 1\%$ of the population.
- Somatic mutation:
  - ✓ Variation compared to one's germline cells. Appears during one's lifetime and is not present in all cells.
  - ✓ Somatic mutations play an important role in the onset of many cancers.

Somatic mutations in cancer

- Large endeavours have set out to sequence many cancer genomes during the passed decade.

- Questions:
  - ✓ Patient stratification
  - ✓ Prognosis: survival prediction, risk of metastasis ...



Trait prediction from SNPs data

	X_1	-	-	-	X_2	-	-	-	X_3	-	-	-	-	-	X_4	-	-	-	-	X_5	-	-	-	-	-	-		y
$P_1 \ldots$	A	T	C	G	C	T	G	A	A	T	A	C	G	G	C	T	C	G	A	A	A	T	C	G	G	A	...	✓
$P_2 \ldots$	T	T	C	G	G	T	G	A	G	T	A	C	G	G	T	T	C	G	A	A	A	T	C	G	G	A	...	✗
$P_3 \ldots$	A	T	C	G	C	T	G	A	A	T	A	C	G	G	T	T	C	G	A	A	A	T	C	G	G	A	...	✗
$P_4 \ldots$	T	T	C	G	C	T	G	A	G	T	A	C	G	G	C	T	C	G	A	C	A	T	C	G	G	A	...	✓
$P_5 \ldots$	T	T	C	G	C	T	G	A	G	T	A	C	G	G	C	T	C	G	A	C	A	T	C	G	G	A	...	✓

Trait prediction from SNPs data

	X_1	-	-	-	X_2	-	-	-	X_3	-	-	-	-	-	X_4	-	-	-	-	X_5	-	-	-	-	-	-		y
$P_1 \ldots$	A	T	C	G	C	T	G	A	A	T	A	C	G	G	C	T	C	G	A	A	A	T	C	G	G	A	$\ldots$	✓
$P_2 \ldots$	T	T	C	G	G	T	G	A	G	T	A	C	G	G	T	T	C	G	A	A	A	T	C	G	G	A	$\ldots$	✗
$P_3 \ldots$	A	T	C	G	C	T	G	A	A	T	A	C	G	G	T	T	C	G	A	A	A	T	C	G	G	A	$\ldots$	✗
$P_4 \ldots$	T	T	C	G	C	T	G	A	G	T	A	C	G	G	C	T	C	G	A	C	A	T	C	G	G	A	$\ldots$	✓
$P_5 \ldots$	T	T	C	G	C	T	G	A	G	T	A	C	G	G	C	T	C	G	A	C	A	T	C	G	G	A	$\ldots$	✓

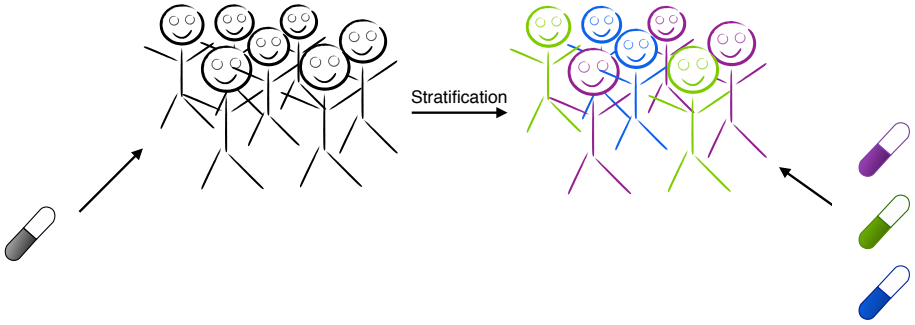
Trait prediction from SNPs data

	X_1	-	-	-	X_2	-	-	-	X_3	-	-	-	-	-	X_4	-	-	-	-	X_5	-	-	-	-	-	-		y
$P_1 \ldots$	A	T	C	G	C	T	G	A	A	T	A	C	G	G	C	T	C	G	A	A	A	T	C	G	G	A	$\ldots$	✓
$P_2 \ldots$	T	T	C	G	G	T	G	A	G	T	A	C	G	G	T	T	C	G	A	A	A	T	C	G	G	A	$\ldots$	✗
$P_3 \ldots$	A	T	C	G	C	T	G	A	A	T	A	C	G	G	T	T	C	G	A	A	A	T	C	G	G	A	$\ldots$	✗
$P_4 \ldots$	T	T	C	G	C	T	G	A	G	T	A	C	G	G	C	T	C	G	A	C	A	T	C	G	G	A	$\ldots$	✓
$P_5 \ldots$	T	T	C	G	C	T	G	A	G	T	A	C	G	G	C	T	C	G	A	C	A	T	C	G	G	A	$\ldots$	✓

- Questions: Breast cancer risk? Response to treatment? Pharmacokinetics? ...
- Polygenic Risk Score (PRS): $PRS_i = \sum_k \boldsymbol{w}_k \boldsymbol{X}_{ik} I(P_k \leq 5.10^{-8})$
- Difficulties:
  - ✓ High Dimension ($\approx$ 1 million SNPs)
  - ✓ Population structure (linkage disequilibrium)
  - ✓ Confounding factors
  - ✓ Gene-gene interactions
  - ✓ Gene environment interactions

Part 1: NetNorM

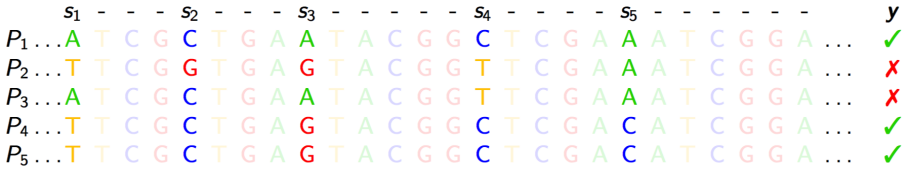
Somatic mutations



- A new representation of somatic mutation profiles,
- based on gene networks,
- to improve patients stratification and survival prediction.

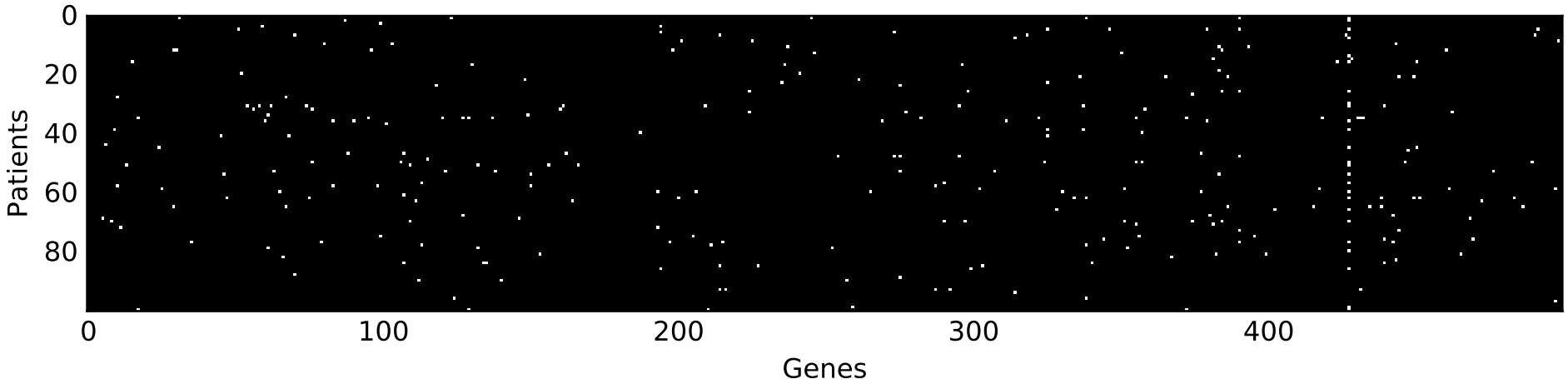
Part 2: WHInter

SNPs



- Taking into account gene-gene interactions in polygenic risk scores.
- A computational challenge.

Somatic mutations



The raw data:

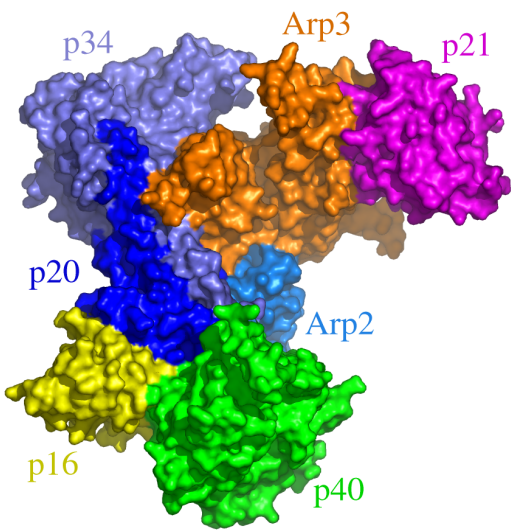
- Binary mutation profiles where a 1 stands for the presence of one (or more) mutation in a given gene for a given patient
- yield poor survival prediction performances,
- are not well suited for patient stratification.

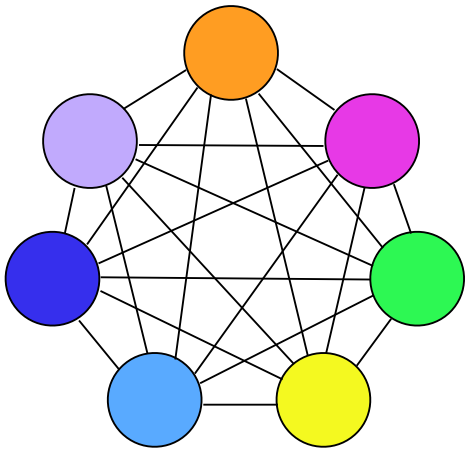
Challenges:

- High dimension (around $\approx 20,000$ genes).
- Low mutation frequency.
- Patients share few mutations in common.

Gene-gene interaction networks

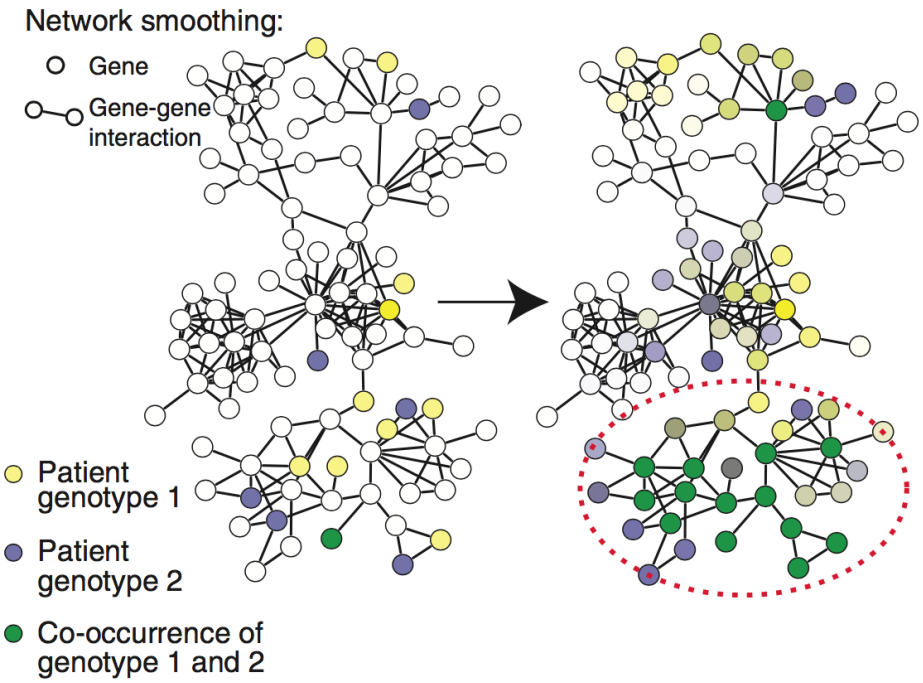
- An idea is to use protein-protein interaction networks to create an overlap between patients.
- Many types of interactions recorded:
  - ✓ Complexes and physical interactions
  - ✓ Biochemical reactions (phosphorylation, . . .)
  - ✓ Catalysis
  - ✓ Regulatory interactions
  - ✓ . . .





- Hypothesis: if two mutations in different genes are close on the gene network, they may cause similar downstream effects.

Previous work: Network-based stratification (NBS)



[*Hofree et al. 2013*]

- **Assumption**

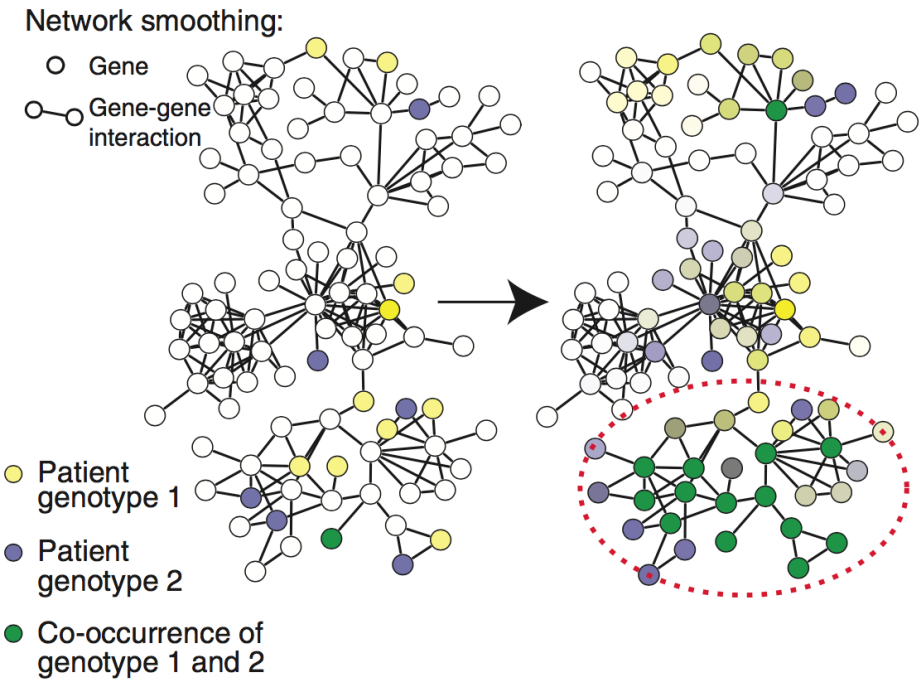
  Even if two tumors have no mutations in common, the same subnetworks may be affected.

- **Method**

  1. Network smoothing.
     Diffusion process.
     Each mutation profile (row of the mutation matrix) is smoothed independently.

  3. Non-Negative matrix factorisation (NMF).

Previous work: Network-based stratification (NBS)



[*Hofree et al. 2013*]

- **Assumption**

  Even if two tumors have no mutations in common, the same subnetworks may be affected.

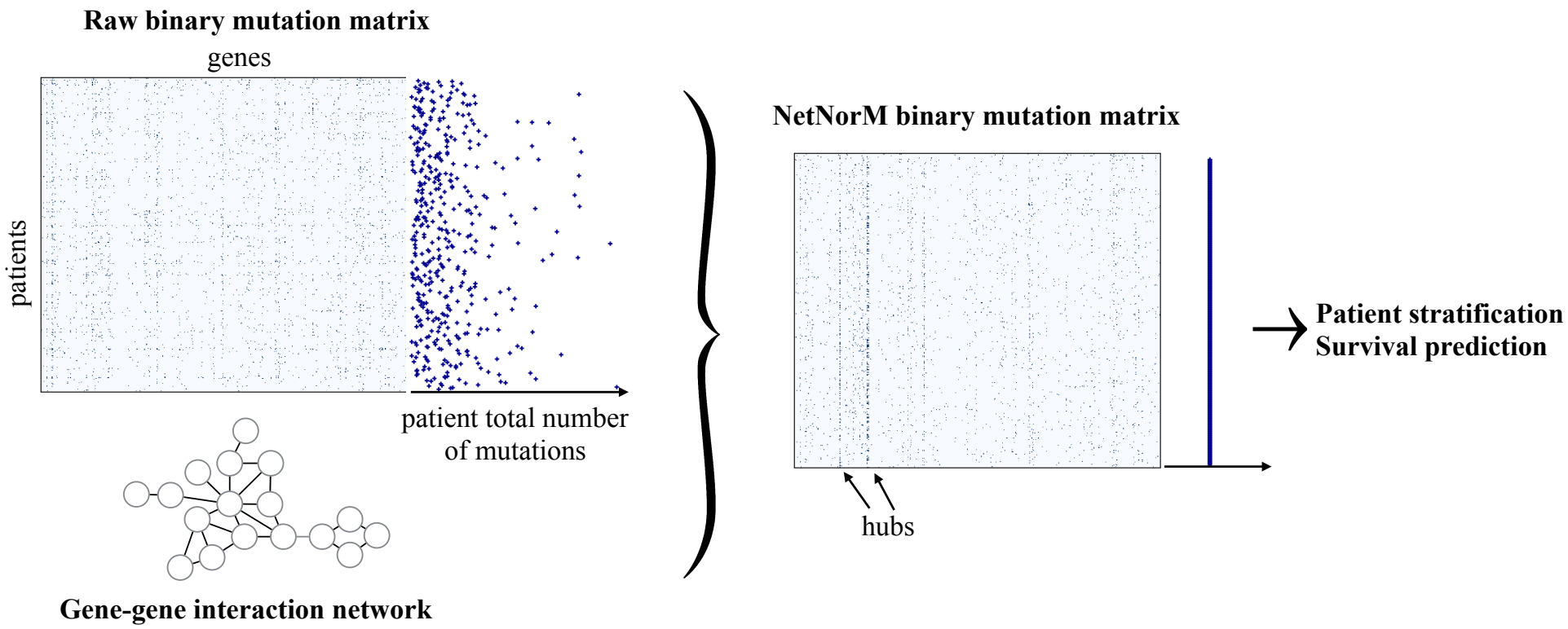
- **Method**

  1. Network smoothing.
     Diffusion process.
     Each mutation profile (row of the mutation matrix) is smoothed independently.
  2. Quantile normalisation (QN)
     The ith smallest value of all samples (patients) is set to the median of all i^{th} smallest values across samples.
  3. Non-Negative matrix factorisation (NMF).

Overview of NetNorM - 1/2

- Quantile normalisation:
  - ✓ has no obvious biological motivation.
  - ✓ it modifies the smoothed mutation profiles so that the interpretation in terms of shared mutated subnetworks is not so straightforward after QN.
  - ✓ QN is crucial for NBS to work
- We propose NetNorM a new representation of mutation profiles:
  - ✓ inspired from the crucial role of QN in NBS,
  - ✓ and try to identify and predictive signals created.
- We compare the different representations of mutations (raw binary, NBS, NetNorM) for two tasks:
  - ✓ survival prediction,
  - ✓ patient stratification.

Overview of NetNorM - 1/2

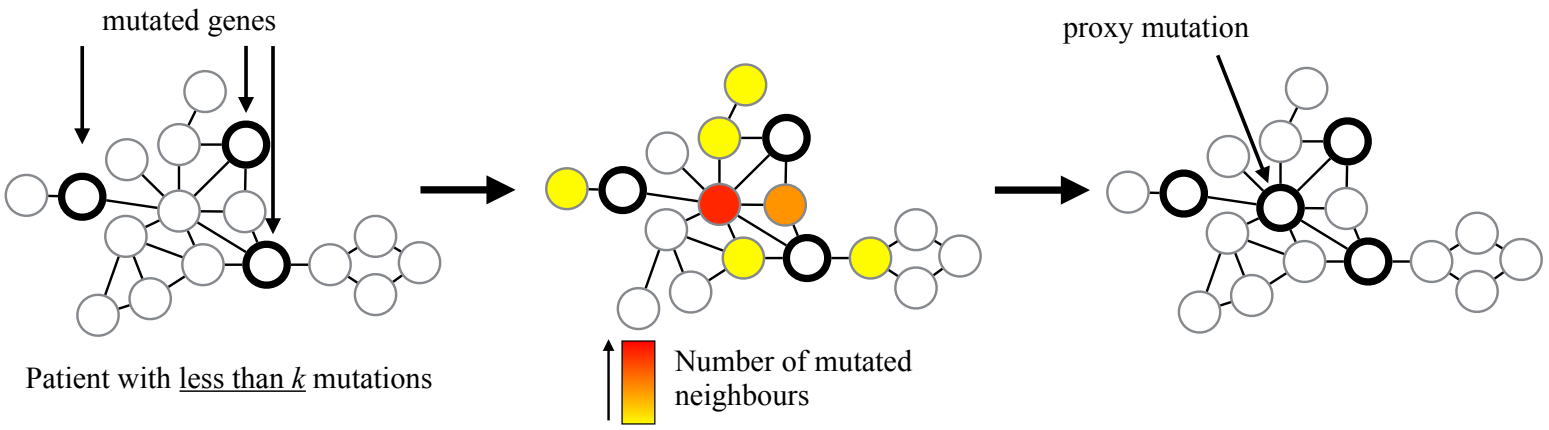


NetNorM replaces $\boldsymbol{x} \in \{0,1\}^p$ by a representation with more information shared between samples $\phi(\boldsymbol{x}) \in \mathcal{H}$ where $\mathcal{H} = \left\{ \boldsymbol{x} \in \{0,1\}^p : \sum_{i=1}^{p} \boldsymbol{x}_i = k \right\}$ and relies on a gene network to remove/add mutations. k is a parameter chosen by cross-validation.

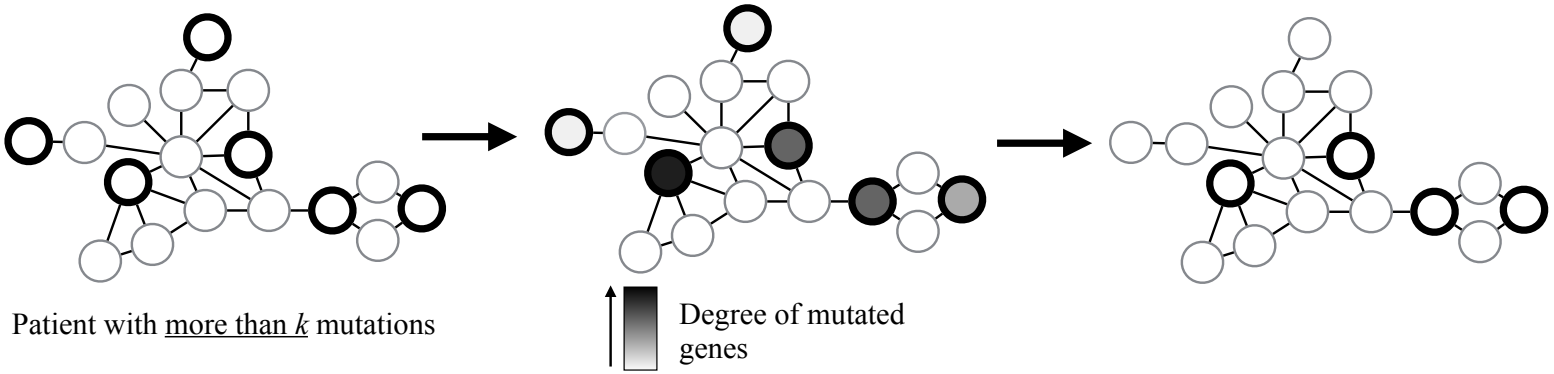
Overview of NetNorM - 2/2

Toy example with $k = 4$: (in reality, k is around of few 10s to a few 100s)

1. **Add** mutations to patients with fewer than k mutations.



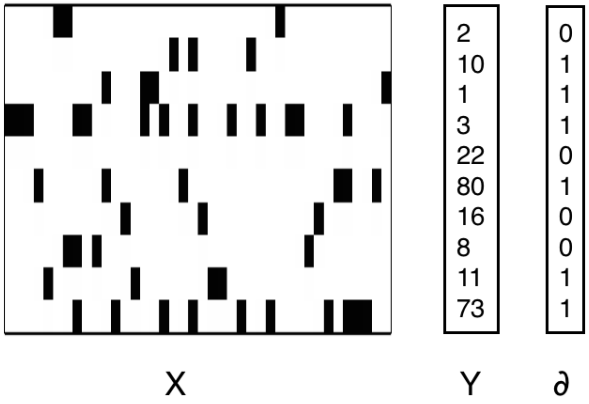
2. **Remove** mutations from patients with more than k mutations.



Large-scale efforts to collect exome somatic mutation profiles

Data used in this study:

- 3,378 samples with survival information (somatic mutations in exomes - silent mutations removed)
- from 8 cancer types
- downloaded from TCGA and cBioPortal.

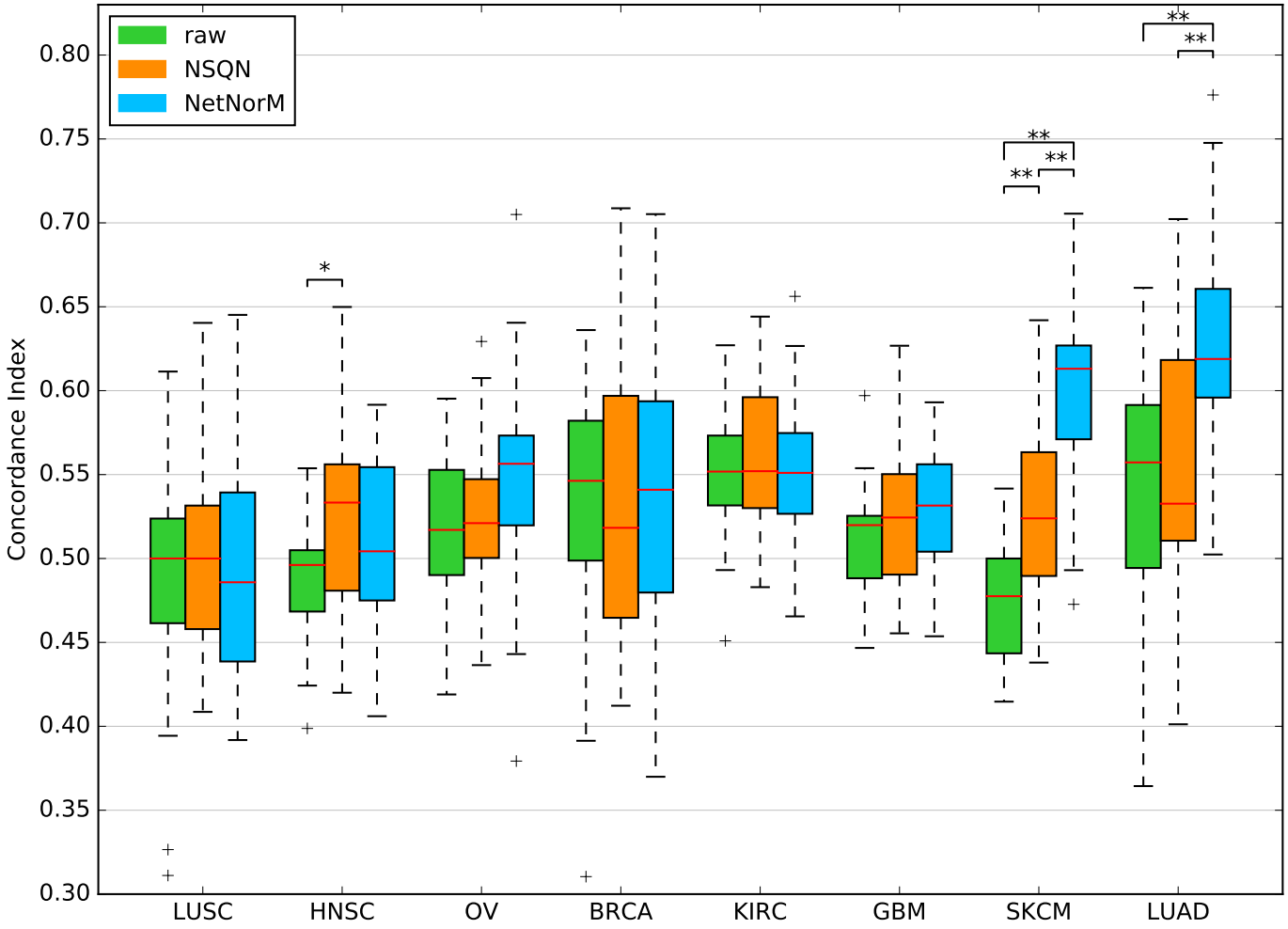


- ✓ $\boldsymbol{X}$: mutation matrix
- ✓ $\boldsymbol{y}$: months of survival since diagnosis
- ✓ $\boldsymbol{\delta}$: censoring status (1: deceased, 0: alive)

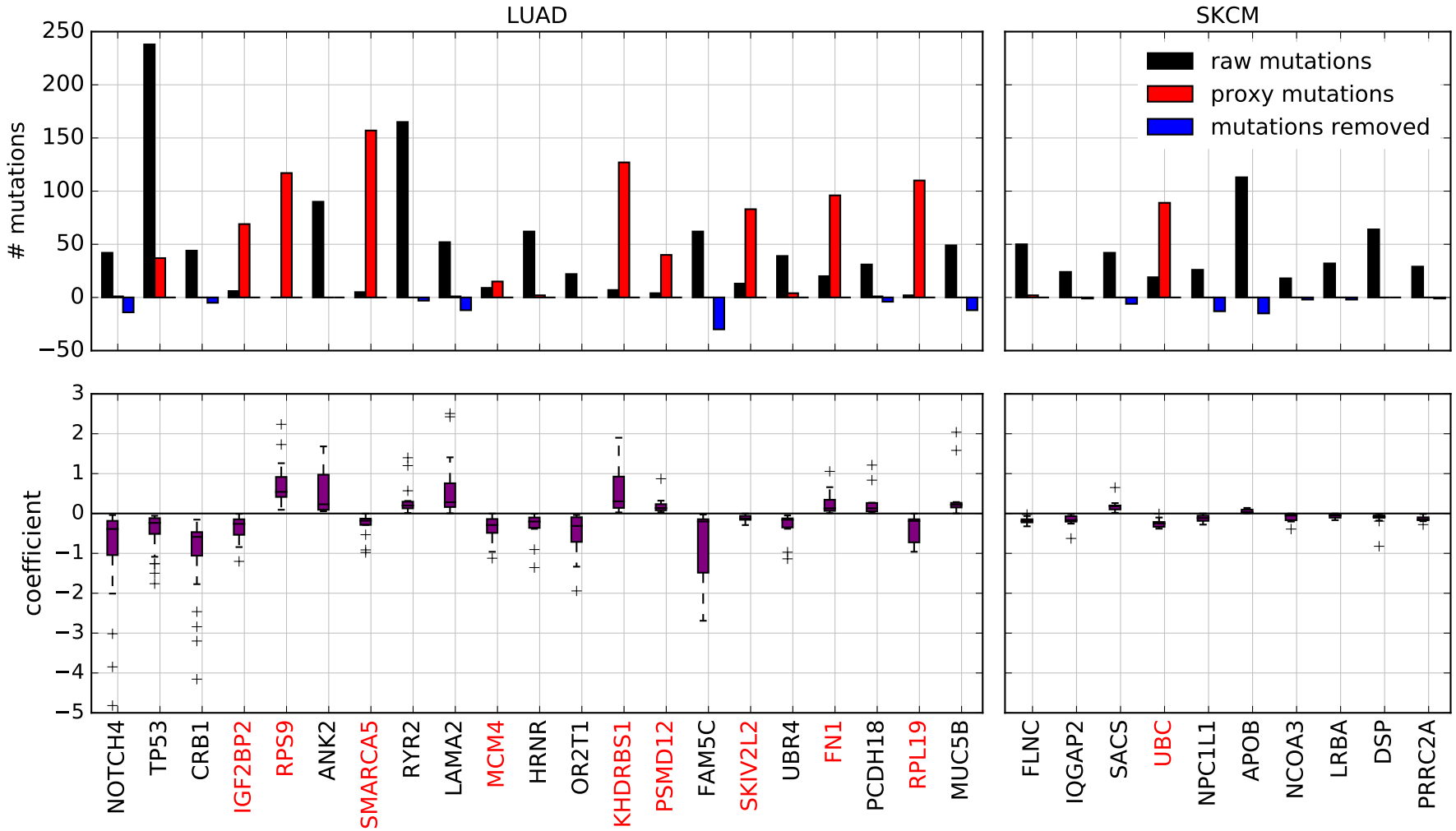
Cancer type	Patients	Genes
LUAD (Lung adenocarcinoma)	430	20 596
SKCM (Skin cutaneous melanoma)	307	17 461
GBM (Glioblastoma multiform)	265	14 748
BRCA (Breast invasive carcinoma)	945	16 806
KIRC (Kidney renal clear cell carcinoma)	411	10 608
HNSC (Head & Neck squam. cell carcinoma)	388	17 022
LUSC (Lung squamous cell carcinoma)	169	13 589
OV (Ovarian serous cystadenocarcinoma)	363	10 192

Comparison of survival prediction performances

- ✓ We assume $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{w}$
- ✓ Sparse survival SVM [*Van Belle et al. 2007*]
- ✓ 4×5-fold cross-validation
- ✓ Gene network: Pathway Commons.

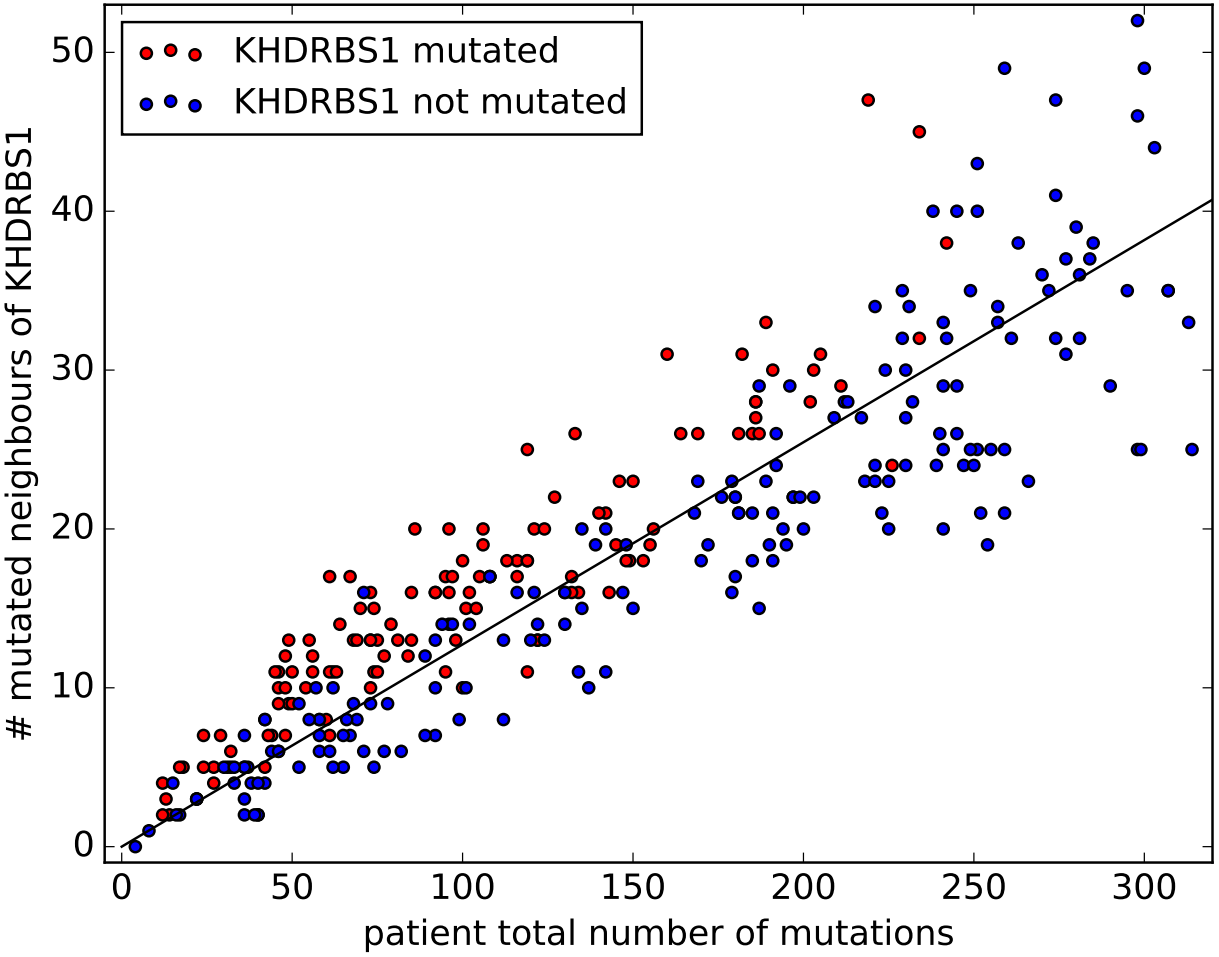


Genes frequently selected in survival prediction models



Genes selected at least 10 times out of 20 folds

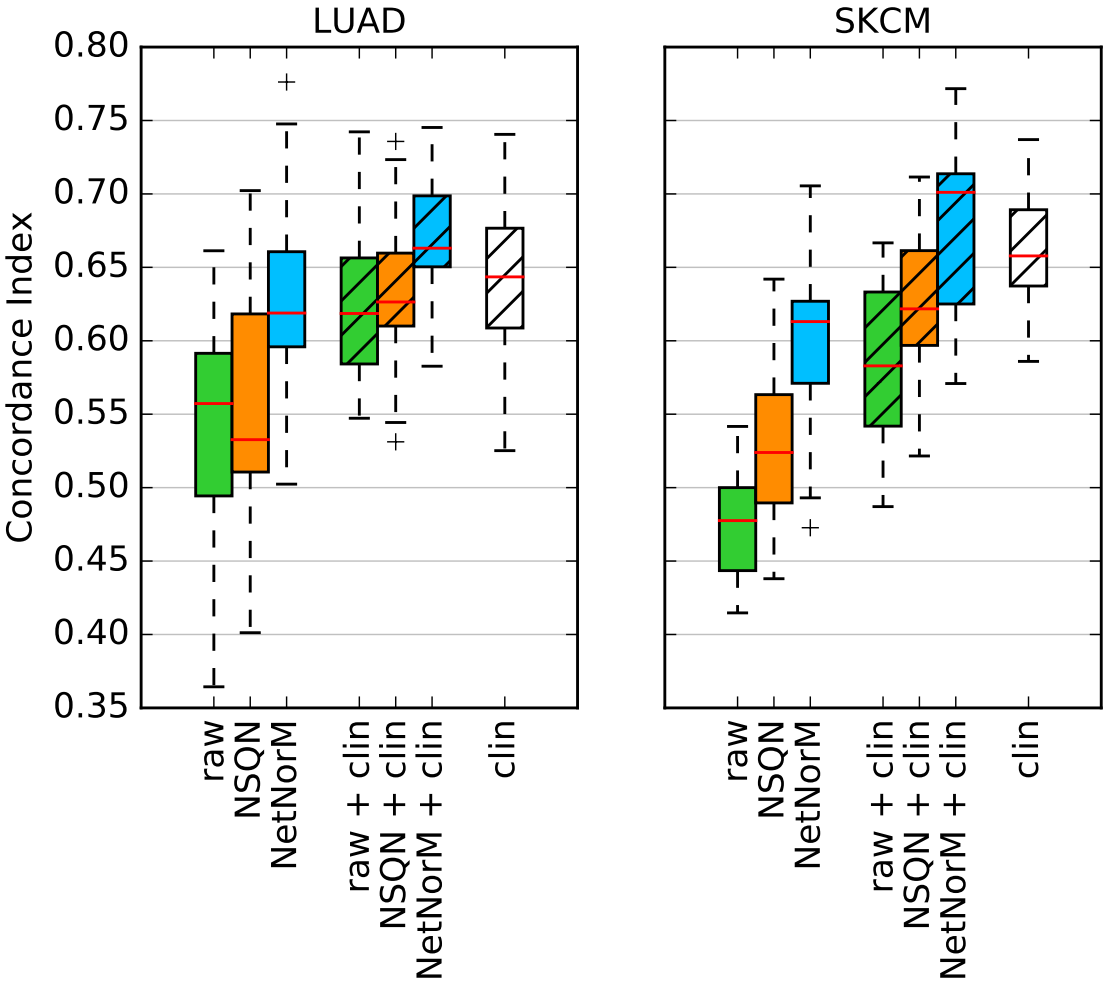
Proxy mutations encode local and global mutational burden



Mutations in KHDRBS1 are almost only proxy mutations.

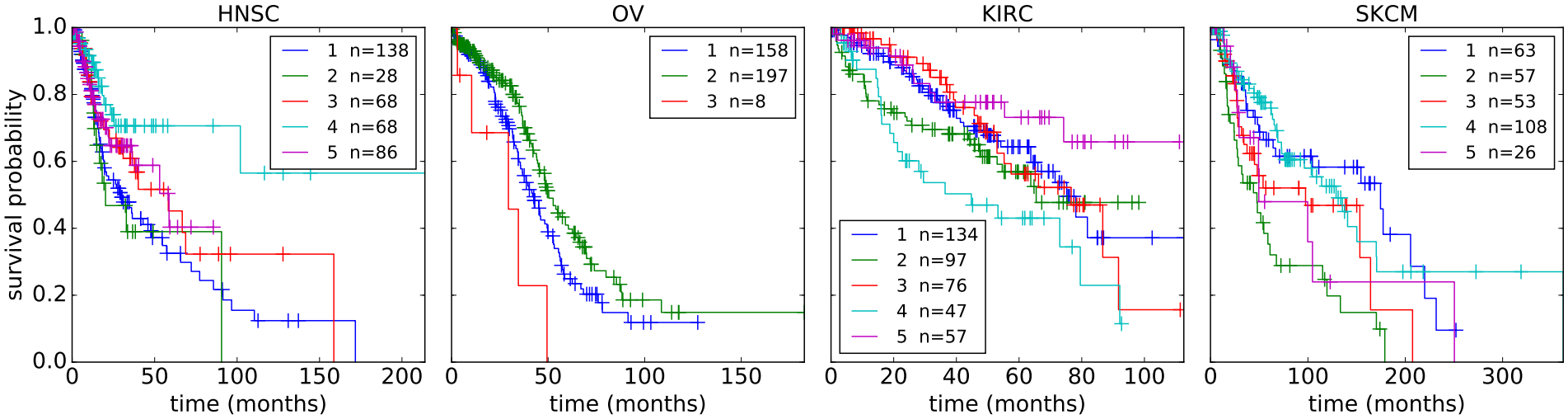
Using mutations and clinical data together

- ✓ Models are learned on mutations and clinical data separately and subsequently averaged.
- ✓ Clinical data alone outperforms mutation data alone.
- ✓ There is information in mutation data, as captured by NetNorM, that allows to improve on clinical data alone.



Unsupervised patient stratification

- Unsupervised patient stratification:
  - ✓ With NMF + consensus clustering.
  - ✓ Number of clusters tested vary from 2 to 6.
  - ✓ The logrank test (case > 2 subgroups) tests whether or not there is at least one subgroup whose survival distribution is different from the others.

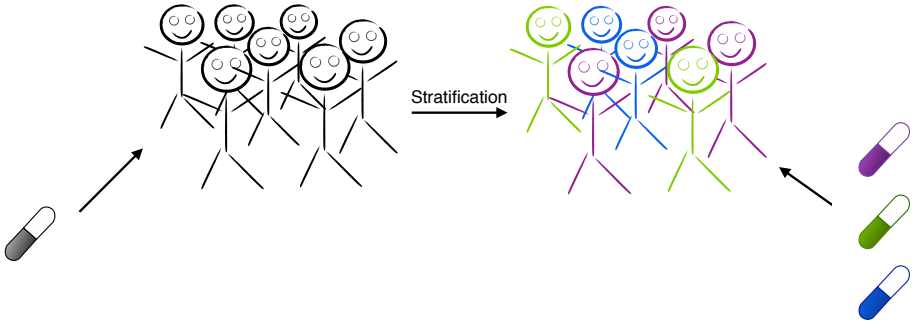


Conclusion

- Somatic mutation profiles are challenging because:
  - ✓ Low mutation frequency.
  - ✓ Few shared mutations among patients.
  - ✓ Large variability in the total number of mutations.
- Network smoothing/local averaging sometimes helps
  - ✓ but with current methods, looking at direct neighbours is good enough.
- Normalising for the total number of mutations is important
  - ✓ with NSQN or NetNorM.
  - ✓ NetNorM creates a signal related to local and global mutational burden.

Part 1: NetNorM

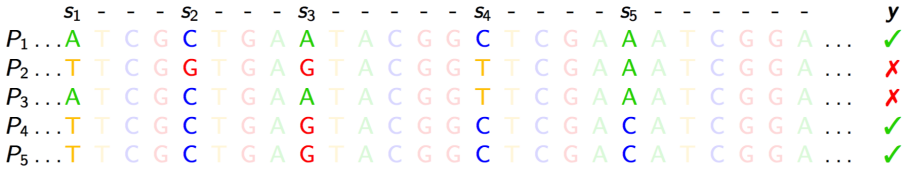
Somatic mutations



- A new representation of somatic mutation profiles,
- based on gene networks,
- to improve patients stratification and survival prediction.

Part 2: WHInter

SNPs



- Taking into account gene-gene interactions in polygenic risk scores.
- A computational challenge.

Back to the SNPS

	X_1	-	-	-	X_2	-	-	-	X_3	-	-	-	-	-	X_4	-	-	-	-	X_5	-	-	-	-	-	-		y
$P_1 \ldots$	A	T	C	G	C	T	G	A	A	T	A	C	G	G	C	T	C	G	A	A	A	T	C	G	G	A	...	✓
$P_2 \ldots$	T	T	C	G	G	T	G	A	G	T	A	C	G	G	T	T	C	G	A	A	A	T	C	G	G	A	...	✗
$P_3 \ldots$	A	T	C	G	C	T	G	A	A	T	A	C	G	G	T	T	C	G	A	A	A	T	C	G	G	A	...	✗
$P_4 \ldots$	T	T	C	G	C	T	G	A	G	T	A	C	G	G	C	T	C	G	A	C	A	T	C	G	G	A	...	✓
$P_5 \ldots$	T	T	C	G	C	T	G	A	G	T	A	C	G	G	C	T	C	G	A	C	A	T	C	G	G	A	...	✓

Back to the SNPS

	X_1	-	-	-	X_2	-	-	-	X_3	-	-	-	-	-	X_4	-	-	-	-	X_5	-	-	-	-	-	-		y
$P_1 \ldots$	A	T	C	G	C	T	G	A	A	T	A	C	G	G	C	T	C	G	A	A	A	T	C	G	G	A	$\ldots$	✓
$P_2 \ldots$	T	T	C	G	G	T	G	A	G	T	A	C	G	G	T	T	C	G	A	A	A	T	C	G	G	A	$\ldots$	✗
$P_3 \ldots$	A	T	C	G	C	T	G	A	A	T	A	C	G	G	T	T	C	G	A	A	A	T	C	G	G	A	$\ldots$	✗
$P_4 \ldots$	T	T	C	G	C	T	G	A	G	T	A	C	G	G	C	T	C	G	A	C	A	T	C	G	G	A	$\ldots$	✓
$P_5 \ldots$	T	T	C	G	C	T	G	A	G	T	A	C	G	G	C	T	C	G	A	C	A	T	C	G	G	A	$\ldots$	✓

The LASSO:

$$\boldsymbol{y} \approx \boldsymbol{X}\boldsymbol{w}^*$$

where

$$\boldsymbol{w}^* \leftarrow \underset{\boldsymbol{w} \in \mathbb{R}^p}{\operatorname{argmin}} \; \frac{1}{n} \underbrace{\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}\|_2^2}_{\text{data fitting term}} + \underbrace{\lambda \|\boldsymbol{w}\|_1}_{\text{sparsity inducing penalty}}$$

$$\boldsymbol{X} = \underbrace{\begin{pmatrix} | & | & & | \\ \boldsymbol{x}_1 & \boldsymbol{x}_2 & \ldots & \boldsymbol{x}_p \\ | & | & & | \end{pmatrix}}_{\in [\![0,1]\!]^{n \times p}}$$

Back to the SNPS

	X_1	-	-	-	X_2	-	-	-	X_3	-	-	-	-	-	X_4	-	-	-	-	X_5	-	-	-	-	-	-		y
$P_1 \ldots$	1	T	C	G	0	T	G	A	1	T	A	C	G	G	0	T	C	G	A	0	A	T	C	G	G	A	...	✓
$P_2 \ldots$	0	T	C	G	1	T	G	A	0	T	A	C	G	G	1	T	C	G	A	0	A	T	C	G	G	A	...	✗
$P_3 \ldots$	1	T	C	G	0	T	G	A	1	T	A	C	G	G	1	T	C	G	A	0	A	T	C	G	G	A	...	✗
$P_4 \ldots$	0	T	C	G	0	T	G	A	0	T	A	C	G	G	0	T	C	G	A	1	A	T	C	G	G	A	...	✓
$P_5 \ldots$	0	T	C	G	0	T	G	A	0	T	A	C	G	G	0	T	C	G	A	1	A	T	C	G	G	A	...	✓

The LASSO:

$$\boldsymbol{y} \approx \boldsymbol{X}\boldsymbol{w}^*$$

where

$$\boldsymbol{w}^* \leftarrow \underset{\boldsymbol{w} \in \mathbb{R}^p}{\operatorname{argmin}} \frac{1}{n} \underbrace{\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}\|_2^2}_{\text{data fitting term}} + \underbrace{\lambda \|\boldsymbol{w}\|_1}_{\text{sparsity inducing penalty}}$$

$$\boldsymbol{X} = \underbrace{\begin{pmatrix} | & | & & | \\ \boldsymbol{X}_1 & \boldsymbol{X}_2 & \ldots & \boldsymbol{X}_p \\ | & | & & | \end{pmatrix}}_{\in [\![0,1]\!]^{n \times p}}$$

Back to the SNPS

	X_1	- - -	X_2	- - -	X_3	- - - - - -	X_4	- - - -	X_5	- - - - - -		y
P_1	...1	T C G	0	T G A	1	T A C G G	0	T C G A	0	A T C G G A	...	✓
P_2	...0	T C G	1	T G A	0	T A C G G	1	T C G A	0	A T C G G A	...	✗
P_3	...1	T C G	0	T G A	1	T A C G G	1	T C G A	0	A T C G G A	...	✗
P_4	...0	T C G	0	T G A	0	T A C G G	0	T C G A	1	A T C G G A	...	✓
P_5	...0	T C G	0	T G A	0	T A C G G	0	T C G A	1	A T C G G A	...	✓

The LASSO with pairwise interactions:

$$\boldsymbol{y} \approx \boldsymbol{Z}\boldsymbol{w}^*$$

where

$$\boldsymbol{w}^* \leftarrow \underset{\boldsymbol{w}\in\mathbb{R}^D}{\operatorname{argmin}} \frac{1}{n} \underbrace{\|\boldsymbol{y} - \boldsymbol{Z}\boldsymbol{w}\|_2^2}_{\text{data fitting term}} + \underbrace{\lambda \|\boldsymbol{w}\|_1}_{\text{sparsity inducing penalty}}$$

$$\boldsymbol{Z} = \underbrace{\left(\begin{array}{cccc|cccc} | & | & & | & | & | & & | \\ \boldsymbol{X}_1 & \boldsymbol{X}_2 & \dots & \boldsymbol{X}_p & \boldsymbol{X}_1\boldsymbol{X}_1 & \boldsymbol{X}_1\boldsymbol{X}_2 & \dots & \boldsymbol{X}_p\boldsymbol{X}_p \\ | & | & & | & | & | & & | \end{array}\right)}_{\in [\![0,1]\!]^{n\times D}}$$

where $D = \frac{p(p+1)}{2}$.

Back to the SNPS

	X_1	- - -	X_2	- - -	X_3	- - - - - -	X_4	- - - -	X_5	- - - - - -		y
$P_1 \ldots$	1	T C G	0	T G A	1	T A C G G	0	T C G A	0	A T C G G A	...	✓
$P_2 \ldots$	0	T C G	1	T G A	0	T A C G G	1	T C G A	0	A T C G G A	...	✗
$P_3 \ldots$	1	T C G	0	T G A	1	T A C G G	1	T C G A	0	A T C G G A	...	✗
$P_4 \ldots$	0	T C G	0	T G A	0	T A C G G	0	T C G A	1	A T C G G A	...	✓
$P_5 \ldots$	0	T C G	0	T G A	0	T A C G G	0	T C G A	1	A T C G G A	...	✓

The LASSO with pairwise interactions:

$$\boldsymbol{y} \approx \boldsymbol{Z}\boldsymbol{w}^*$$

$$\boldsymbol{Z} = \underbrace{\left(\begin{array}{cccc|cccc} | & | & & | & | & | & & | \\ \boldsymbol{X}_1 & \boldsymbol{X}_2 & \ldots & \boldsymbol{X}_p & \boldsymbol{X}_1\boldsymbol{X}_1 & \boldsymbol{X}_1\boldsymbol{X}_2 & \ldots & \boldsymbol{X}_p\boldsymbol{X}_p \\ | & | & & | & | & | & & | \end{array}\right)}_{\in [\![0,1]\!]^{n \times D}}$$

where

$$\boldsymbol{w}^* \leftarrow \underset{\boldsymbol{w} \in \mathbb{R}^D}{\operatorname{argmin}} \; \frac{1}{n} \underbrace{\|\boldsymbol{y} - \boldsymbol{Z}\boldsymbol{w}\|_2^2}_{\text{data fitting term}} + \underbrace{\lambda \|\boldsymbol{w}\|_1}_{\text{sparsity inducing penalty}}$$

where $D = \frac{p(p+1)}{2}$.

If $p = 100,000$, then $D \approx 5 \times 10^9$. Classical LASSO solvers will be too slow.
We propose a solver that provides an optimal solution to problems of such size in a reasonable time.

Working set and screening strategies

Safe screening rules

- Safe screening rules:

  Given a primal-dual feasible solution, identify features which are guaranteed not to belong to the optimal support.

- Safe Pattern Pruning (SPP) [*Nakagawa et al. 2016*]:

  Applies safe rules to speed-up sparse linear model estimation with higher-order interactions.

- Main drawback:
  Safe screening rules and consequently SPP is too conservative.

[*El Ghaoui et al. 2012; Fercoq et al. 2015*], ...

Working set strategies

A simple working set algorithm

Input: $\boldsymbol{Z} \in \{0,1\}^{n \times D}, \boldsymbol{y} \in \mathbb{R}^n, \lambda > 0$
Output: $\boldsymbol{w}^*, b^*$

1: Set $\boldsymbol{\phi} \leftarrow \boldsymbol{y}$, $\mathcal{W} = \emptyset$.
2: **while** true **do**
3: $\quad \mathcal{W}' = \left\{ i \in [\![D]\!] : \left| \boldsymbol{Z}_i^\top \boldsymbol{\phi} \right| \geq \lambda \right\}$
4: $\quad$ **if** $\max_{i \in \mathcal{W}'} \left| \boldsymbol{Z}_i^\top \boldsymbol{\phi} \right| \leq \lambda$ **then** Break **else** $\mathcal{W} \leftarrow \mathcal{W}'$
5: $\quad \boldsymbol{w}^*_{\mathcal{W}}, b^* \leftarrow \underset{\boldsymbol{w}_{\mathcal{W}}, b}{\text{argmin}} \| \boldsymbol{y} - \boldsymbol{Z}_{\mathcal{W}} \boldsymbol{w}_{\mathcal{W}} - b\mathbf{1}_n \|_2^2 + \lambda \| w_{\mathcal{W}} \|_1$
6: $\quad \boldsymbol{\phi} \leftarrow \boldsymbol{y} - \boldsymbol{Z}_{\mathcal{W}} \boldsymbol{w}^*_{\mathcal{W}} - b^* \mathbf{1}_n$.
7: **end while**

- When D is too big, the computation of the working set is too expensive.
- WHInter is a working set algorithm where line 3 is accelerated.

[*Friedman et al. 2010; Johnson and Guestrin 2015; Massias et al. 2018*], ...

Outline

WHInter

- Key ideas for a fast delineation of the working set
  - Branch bound
  - Maximum Inner Product Search (MIPS)
- Experimental results
  - Simulations
  - Preliminary results on real data

Outline

WHInter

- Key ideas for a fast delineation of the working set
  - Branch bound
  - Maximum Inner Product Search (MIPS)
- Experimental results
  - Simulations
  - Preliminary results on real data

Branch upper bound

- The working set update reads:

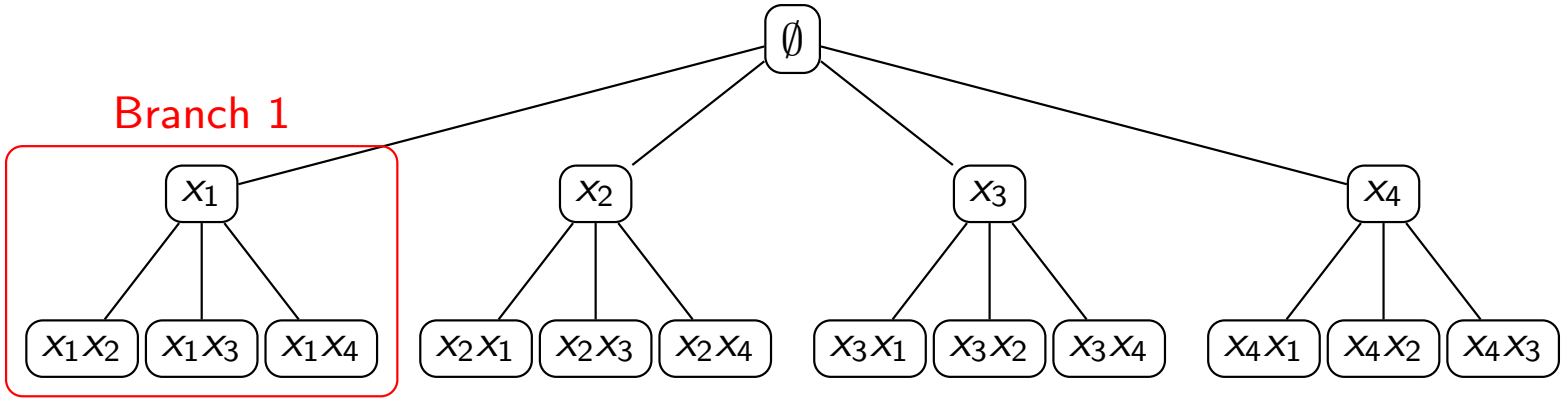
$$\mathcal{W}' = \left\{ i \in [\![D]\!] : \left| \boldsymbol{Z}_i^\top \phi \right| \geq \lambda \right\} .$$

with ϕ the current residual.
Scales as $O(p^2)$.

- Idea: Find an upper bound B_j s.t.:

$$\max_{k \in [\![p]\!] : \tau(j,k) \notin \mathcal{W}} \left| \boldsymbol{Z}_{\tau(j,k)}^\top \phi \right| \leq B_j .$$

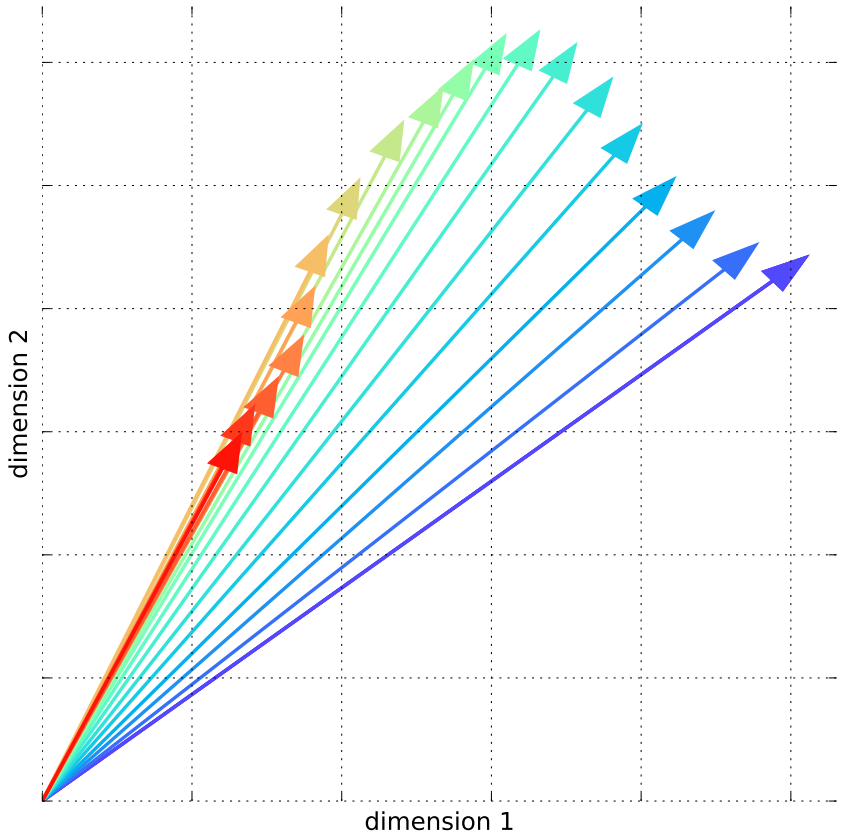
If $B_j < \lambda$, then branch j does not contain any feature that belongs to the working set and is not already in it.



- Let $\tau(k,j)$ be the index of the feature $\boldsymbol{X}_j\boldsymbol{X}_k$ in the expanded matrix $\boldsymbol{Z}$ such that $\boldsymbol{Z}_{\tau(j,k)} = \boldsymbol{Z}_{\tau(k,j)} := \boldsymbol{X}_j\boldsymbol{X}_k = \boldsymbol{X}_k\boldsymbol{X}_j$.

Branch upper bound: geometrical intuition

To compute the branch bound, we propose to leverage the relationship between residuals along a regularisation path (or optimisation path).



Sequence of residuals ϕ obtained along the optimisation path (starts with the purple residual and ends wit the red one).

Branch upper bound: derivation

- Let $\mathbf{\Phi}_j^{ref}$ be a *reference residual* chosen for branch j.
  Let $\boldsymbol{m}_j^{ref} = \max_{k \in [\![p]\!] : \tau(j,k) \notin \mathcal{W}} \left| \boldsymbol{Z}_{\tau(j,k)}^T \mathbf{\Phi}_j^{ref} \right|$. We propose the following bound:

$$
\begin{aligned}
\boldsymbol{m}_j &\stackrel{def}{=} \max_{k \in [\![p]\!] : \tau(j,k) \notin \mathcal{W}} \left| \boldsymbol{Z}_{\tau(j,k)}^\top \phi \right| \\
&\leq \max_{k \in [\![p]\!] : \tau(j,k) \notin \mathcal{W}} \left| \boldsymbol{Z}_{\tau(j,k)}^T \mathbf{\Phi}_j^{ref} \right| + \max_{k \in [\![p]\!] : \tau(j,k) \notin \mathcal{W}} \left| \boldsymbol{Z}_{\tau(j,k)}^T \left(\phi - \mathbf{\Phi}_j^{ref} \right) \right| \\
&\leq \boldsymbol{m}_j^{ref} + max \left(\sum_{i : \phi_i > \mathbf{\Phi}_{ij}^{ref}} \boldsymbol{X}_{ij} \left(\phi_i - \mathbf{\Phi}_{ij}^{ref} \right), - \sum_{i : \phi_i < \mathbf{\Phi}_{ij}^{ref}} \boldsymbol{X}_{ij} \left(\phi_i - \mathbf{\Phi}_{ij}^{ref} \right) \right) \\
&\stackrel{def}{=} \eta \ (\boldsymbol{X}_j, \mathbf{\Phi}_j^{ref}, \phi, \boldsymbol{m}_j^{ref})
\end{aligned}
$$

- We choose $\mathbf{\Phi}_j^{ref}$ as the last residual for which branch j could not be pruned.
- $\boldsymbol{m}_j^{ref}$ needs to be updated each time $\mathbf{\Phi}_j^{ref}$ is updated.

Branch upper bound: derivation

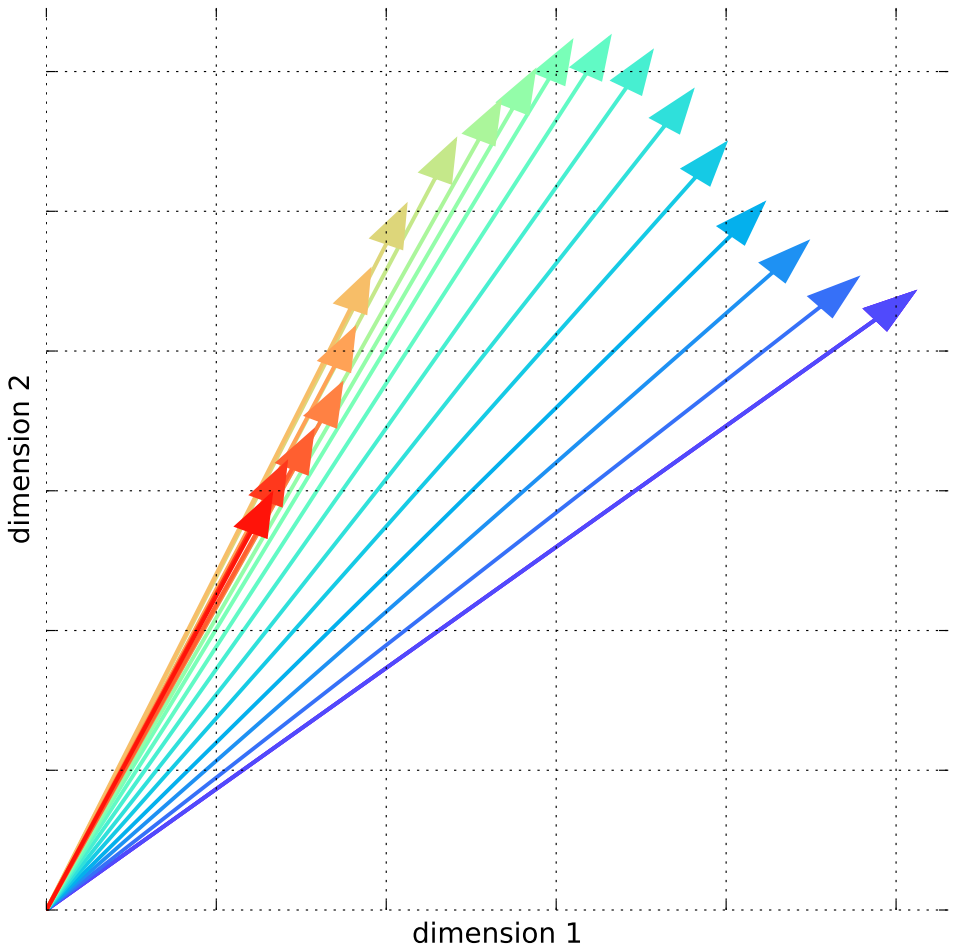
- Let $\mathbf{\Phi}_j^{ref}$ be a *reference residual* chosen for branch j.
  Let $\boldsymbol{m}_j^{ref} = \max_{k \in [\![p]\!] : \tau(j,k) \notin \mathcal{W}} \left| \boldsymbol{Z}_{\tau(j,k)}^T \mathbf{\Phi}_j^{ref} \right|$. We propose the following bound:

$$
\begin{aligned}
\boldsymbol{m}_j &\stackrel{def}{=} \max_{k \in [\![p]\!] : \tau(j,k) \notin \mathcal{W}} \left| \boldsymbol{Z}_{\tau(j,k)}^\top \phi \right| \\
&\leq \max_{k \in [\![p]\!] : \tau(j,k) \notin \mathcal{W}} |\alpha| \left| \boldsymbol{Z}_{\tau(j,k)}^T \mathbf{\Phi}_j^{ref} \right| + \max_{k \in [\![p]\!] : \tau(j,k) \notin \mathcal{W}} \left| \boldsymbol{Z}_{\tau(j,k)}^T \left(\phi - \alpha \mathbf{\Phi}_j^{ref} \right) \right| \\
&\leq |\alpha| \, \boldsymbol{m}_j^{ref} + max \left(\sum_{i : \phi_i > \alpha \mathbf{\Phi}_{ij}^{ref}} \boldsymbol{X}_{ij} \left(\phi_i - \alpha \mathbf{\Phi}_{ij}^{ref} \right), - \sum_{i : \phi_i < \alpha \mathbf{\Phi}_{ij}^{ref}} \boldsymbol{X}_{ij} \left(\phi_i - \alpha \mathbf{\Phi}_{ij}^{ref} \right) \right) \\
&\stackrel{def}{=} \eta_\alpha(\boldsymbol{X}_j, \mathbf{\Phi}_j^{ref}, \phi, \boldsymbol{m}_j^{ref})
\end{aligned}
$$

- We choose $\mathbf{\Phi}_j^{ref}$ as the last residual for which branch j could not be pruned.
- $\boldsymbol{m}_j^{ref}$ needs to be updated each time $\mathbf{\Phi}_j^{ref}$ is updated.

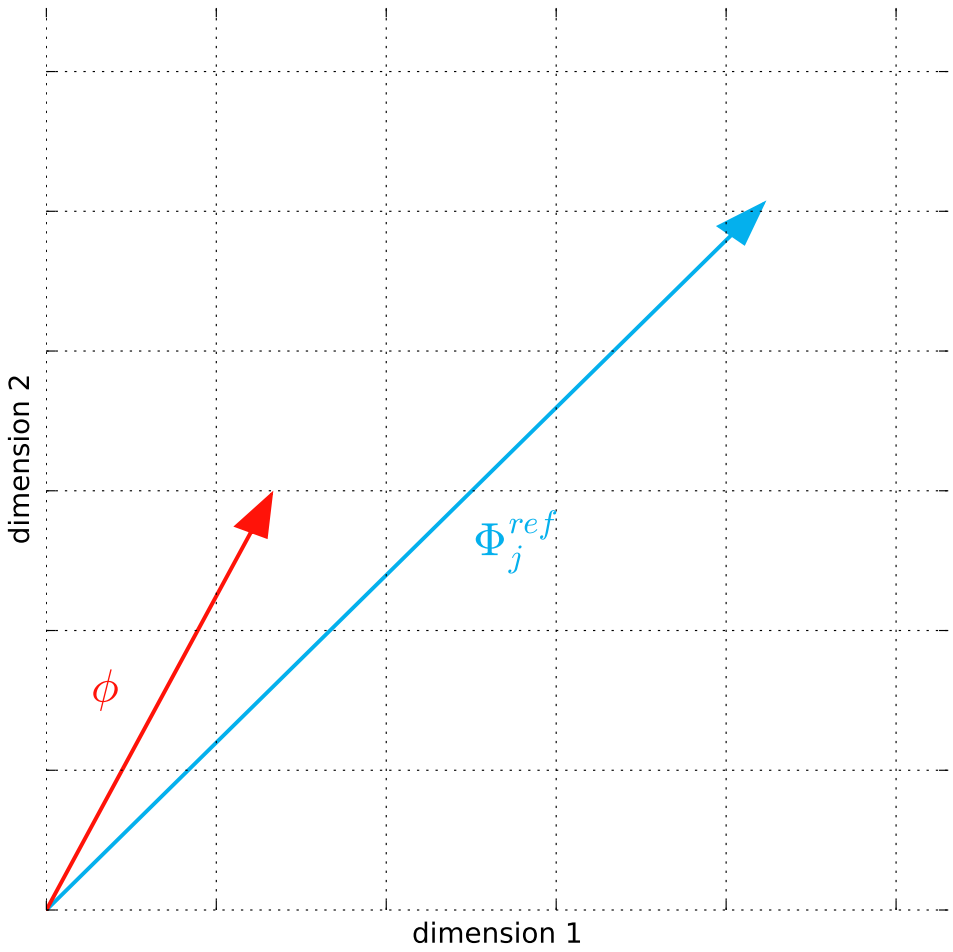
Branch upper bound: role of the parameter alpha

$$\eta_\alpha(\dots) = |\alpha|\, \boldsymbol{m}_j^{ref} + max\left(\sum_{i:\phi_i > \alpha \boldsymbol{\Phi}_{ij}^{ref}} \boldsymbol{X}_{ij}\left(\phi_i - \alpha \boldsymbol{\Phi}_{ij}^{ref}\right), - \sum_{i:\phi_i < \alpha \boldsymbol{\Phi}_{ij}^{ref}} \boldsymbol{X}_{ij}\left(\phi_i - \alpha \boldsymbol{\Phi}_{ij}^{ref}\right)\right)$$



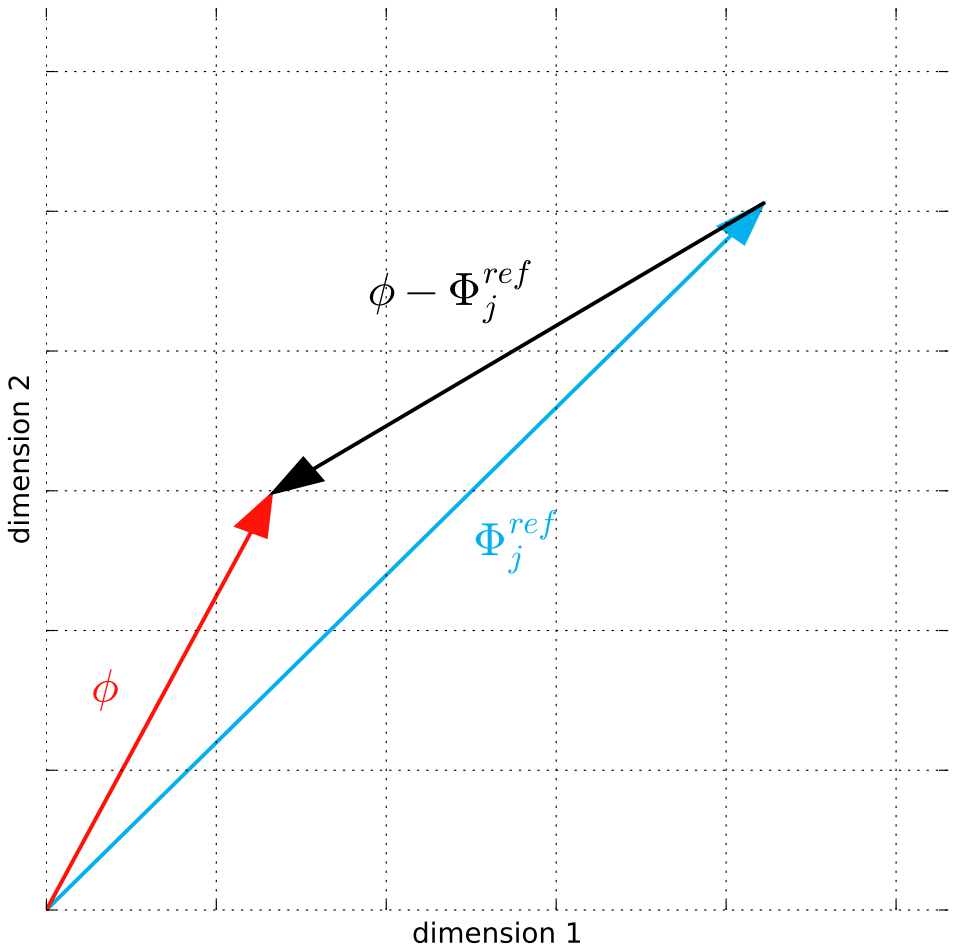
Branch upper bound: role of the parameter alpha

$$\eta_\alpha(\dots) = |\alpha|\, \boldsymbol{m}_j^{ref} + max\left(\sum_{i:\phi_i > \alpha \boldsymbol{\Phi}_{ij}^{ref}} \boldsymbol{X}_{ij}\left(\phi_i - \alpha \boldsymbol{\Phi}_{ij}^{ref}\right), - \sum_{i:\phi_i < \alpha \boldsymbol{\Phi}_{ij}^{ref}} \boldsymbol{X}_{ij}\left(\phi_i - \alpha \boldsymbol{\Phi}_{ij}^{ref}\right)\right)$$



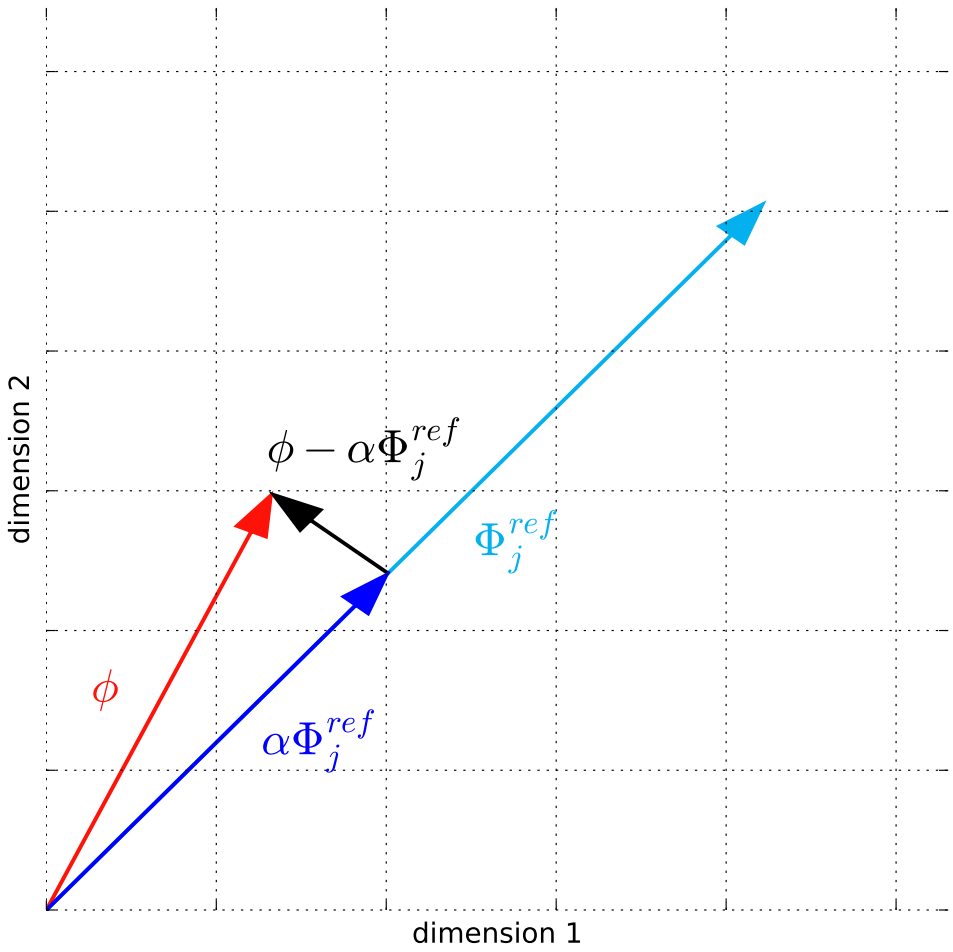
Branch upper bound: role of the parameter alpha

$$\eta_\alpha(\dots) = |\alpha|\, \boldsymbol{m}_j^{ref} + max\left(\sum_{i:\phi_i > \alpha \boldsymbol{\Phi}_{ij}^{ref}} \boldsymbol{X}_{ij}\left(\phi_i - \alpha \boldsymbol{\Phi}_{ij}^{ref}\right), - \sum_{i:\phi_i < \alpha \boldsymbol{\Phi}_{ij}^{ref}} \boldsymbol{X}_{ij}\left(\phi_i - \alpha \boldsymbol{\Phi}_{ij}^{ref}\right)\right)$$



Branch upper bound: role of the parameter alpha

$$\eta_\alpha(\dots) = |\alpha|\, \boldsymbol{m}_j^{ref} + max\left(\sum_{i:\phi_i > \alpha \boldsymbol{\Phi}_{ij}^{ref}} \boldsymbol{X}_{ij} \left(\phi_i - \alpha \boldsymbol{\Phi}_{ij}^{ref}\right), - \sum_{i:\phi_i < \alpha \boldsymbol{\Phi}_{ij}^{ref}} \boldsymbol{X}_{ij} \left(\phi_i - \alpha \boldsymbol{\Phi}_{ij}^{ref}\right) \right)$$

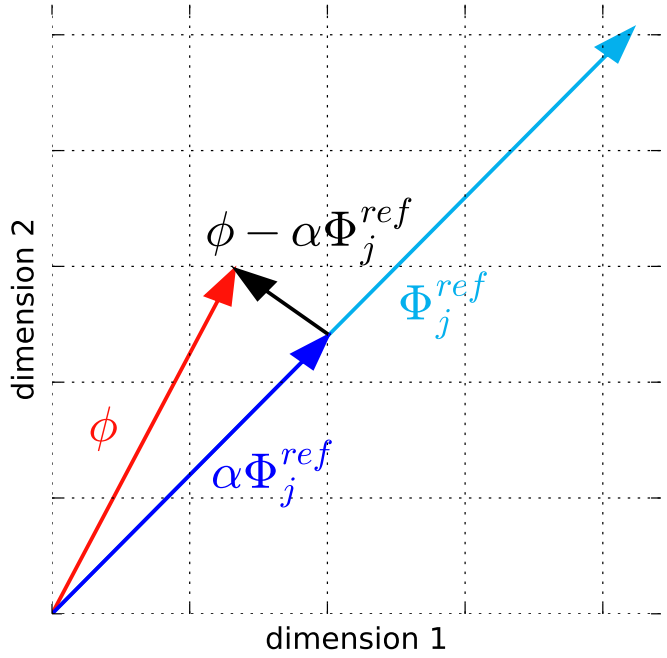


Branch upper bound: choice of the parameter alpha

$$\eta_\alpha(\dots) = |\alpha|\, \boldsymbol{m}_j^{ref} + max\left(\sum_{i:\phi_i > \alpha \boldsymbol{\Phi}_{ij}^{ref}} \boldsymbol{X}_{ij}\left(\phi_i - \alpha \boldsymbol{\Phi}_{ij}^{ref}\right), - \sum_{i:\phi_i < \alpha \boldsymbol{\Phi}_{ij}^{ref}} \boldsymbol{X}_{ij}\left(\phi_i - \alpha \boldsymbol{\Phi}_{ij}^{ref}\right)\right)$$

How to choose α?

- Option 1: $\eta_{min} = \min_{\alpha \in \mathbb{R}} \eta_\alpha$
  - ✓ η is a piecewise continuous function which is convex in α.
  - ✓ η can be minimised in $\mathcal{O}(n_j \log n_j)$ operations.
- Option 2: $\eta_{\alpha_{\ell 2}}$ with $\alpha_{\ell_2} = \frac{\phi^\top \left(\boldsymbol{\Phi}_j^{ref} \odot \boldsymbol{X}_j\right)}{\|\boldsymbol{\Phi}_j^{ref} \odot \boldsymbol{X}_j\|_2^2}$.
  - ✓ $\alpha_{\ell 2}$ minimizes $\|\left(\phi - \alpha \boldsymbol{\Phi}_j^{ref}\right) \odot \boldsymbol{X}_j\|_2^2$.
  - ✓ $\alpha_{\ell 2}$ can be obtained in $\mathcal{O}(n_j)$ operations.

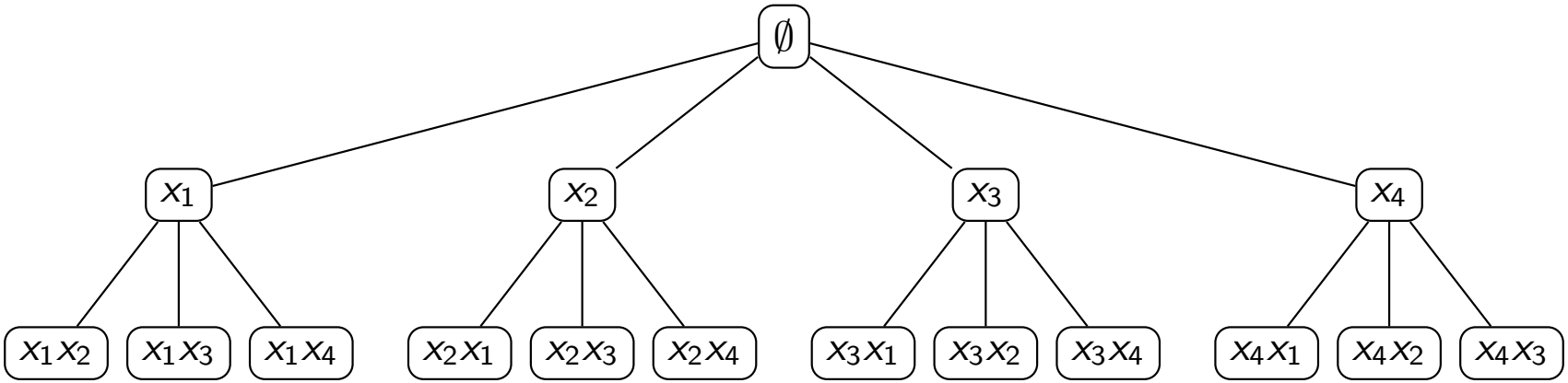


Outline

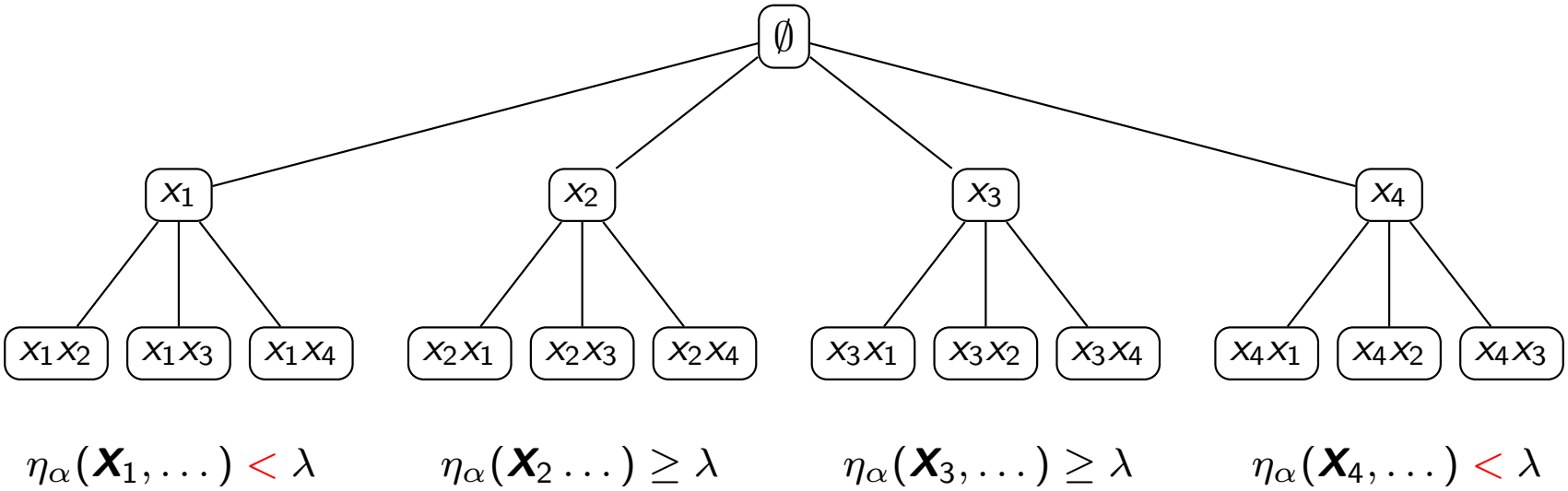
WHInter

- Key ideas for a fast delineation of the working set
  - Branch bound
  - Maximum Inner Product Search (MIPS)
- Experimental results
  - Simulations
  - Preliminary results on real data

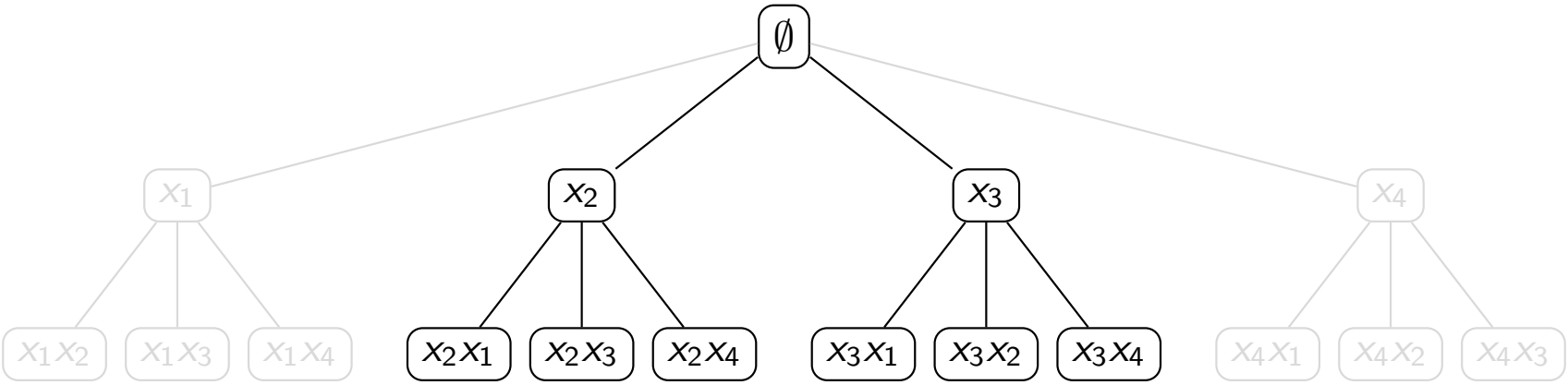
Maximum Inner Product Search (MIPS)



Maximum Inner Product Search (MIPS)

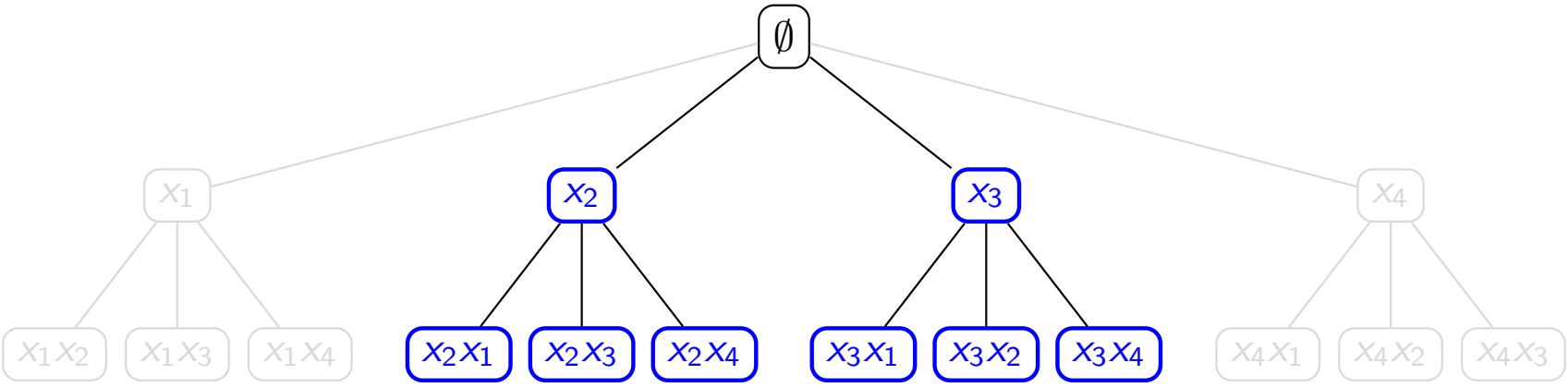


Maximum Inner Product Search (MIPS)



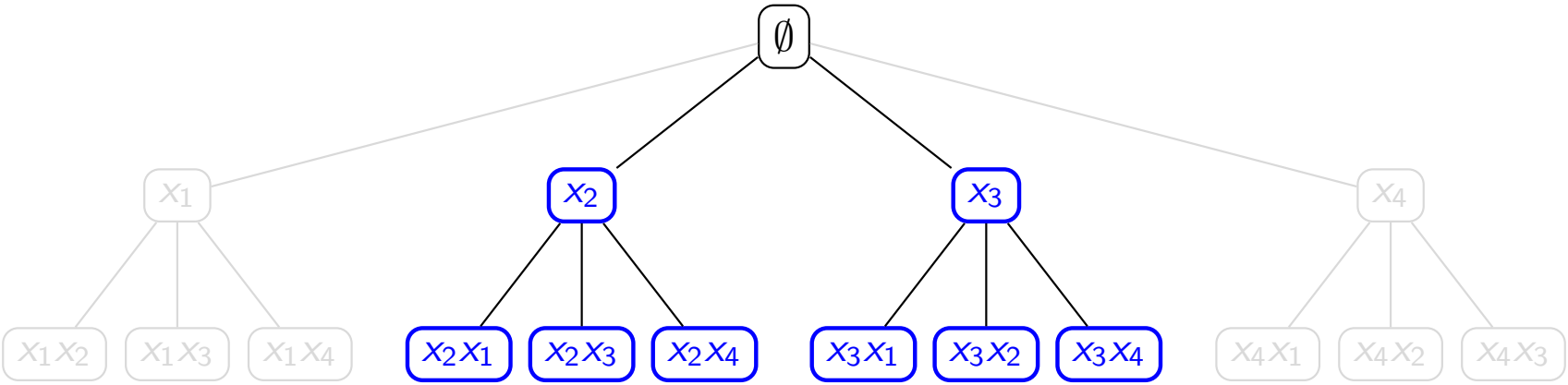
$\eta_\alpha(\boldsymbol{X}_1, \dots) < \lambda$ $\quad$ $\eta_\alpha(\boldsymbol{X}_2 \dots) \geq \lambda$ $\quad$ $\eta_\alpha(\boldsymbol{X}_3, \dots) \geq \lambda$ $\quad$ $\eta_\alpha(\boldsymbol{X}_4, \dots) < \lambda$

Maximum Inner Product Search (MIPS)



$\eta_\alpha(\boldsymbol{X}_1, \dots) < \lambda$ $\quad$ $\eta_\alpha(\boldsymbol{X}_2 \dots) \geq \lambda$ $\quad$ $\eta_\alpha(\boldsymbol{X}_3, \dots) \geq \lambda$ $\quad$ $\eta_\alpha(\boldsymbol{X}_4, \dots) < \lambda$

Maximum Inner Product Search (MIPS)



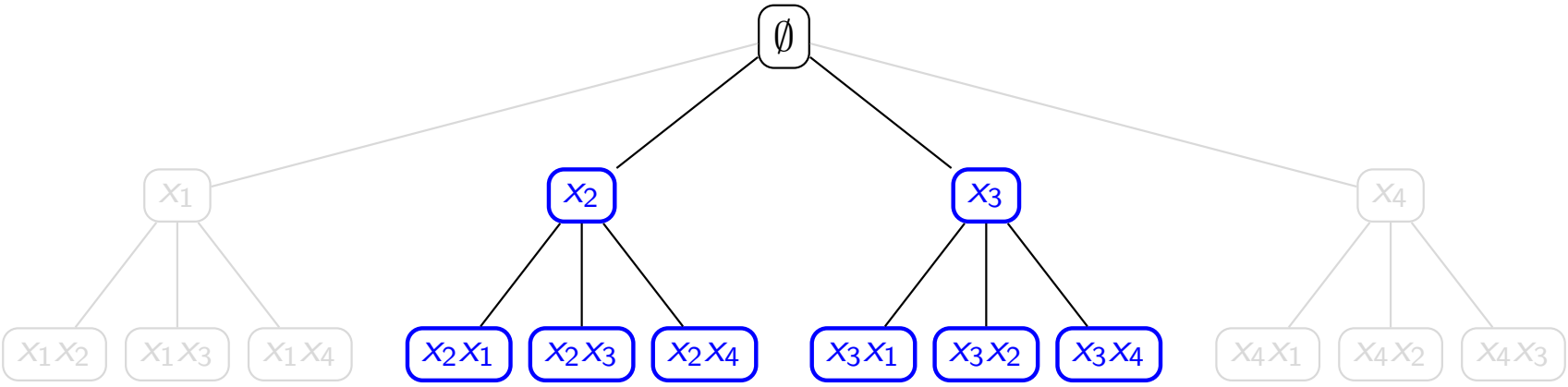
$\eta_\alpha(\boldsymbol{X}_1, \dots) < \lambda$ $\quad$ $\eta_\alpha(\boldsymbol{X}_2 \dots) \geq \lambda$ $\quad$ $\eta_\alpha(\boldsymbol{X}_3, \dots) \geq \lambda$ $\quad$ $\eta_\alpha(\boldsymbol{X}_4, \dots) < \lambda$

We need to scan all features in the set of branches $\mathcal{V}$ that cannot be pruned to:

- ✓ check if they belong to the working set.
- ✓ update $\boldsymbol{m}_{\mathcal{V}}^{ref}$.

These two updates are variants of a Maximum Inner Product Search problem.

Maximum Inner Product Search (MIPS)



$\eta_\alpha(\boldsymbol{X}_1, \dots) < \lambda$ $\quad$ $\eta_\alpha(\boldsymbol{X}_2 \dots) \geq \lambda$ $\quad$ $\eta_\alpha(\boldsymbol{X}_3, \dots) \geq \lambda$ $\quad$ $\eta_\alpha(\boldsymbol{X}_4, \dots) < \lambda$

We need to scan all features in the set of branches $\mathcal{V}$ that cannot be pruned to:

- ✓ check if they belong to the working set.
- ✓ update $\boldsymbol{m}_{\mathcal{V}}^{ref}$.

These two updates are variants of a Maximum Inner Product Search problem.

Let $\boldsymbol{D} \in \mathbb{R}^{n \times p}$ be a set of p vectors and let $\boldsymbol{q} \in \mathbb{R}^n$ be a query vector. The MIPS problem reads:

$$\max_{j \in [\![1,p]\!]} q^\top \boldsymbol{D}_j \,.$$

Previous work has focused on how to solve the MIPS efficiently, for example [*Shrivastava and Li 2014; Teflioudi and Gemulla 2016*]. We use a simple inverted index based approach (Term-At-A-Time) adapted to our case.

Outline

WHInter

- Key ideas for a fast delineation of the working set
  - Branch bound
  - Maximum Inner Product Search (MIPS)
- Experimental results
  - Simulations
  - Preliminary results on real data

Outline

WHInter

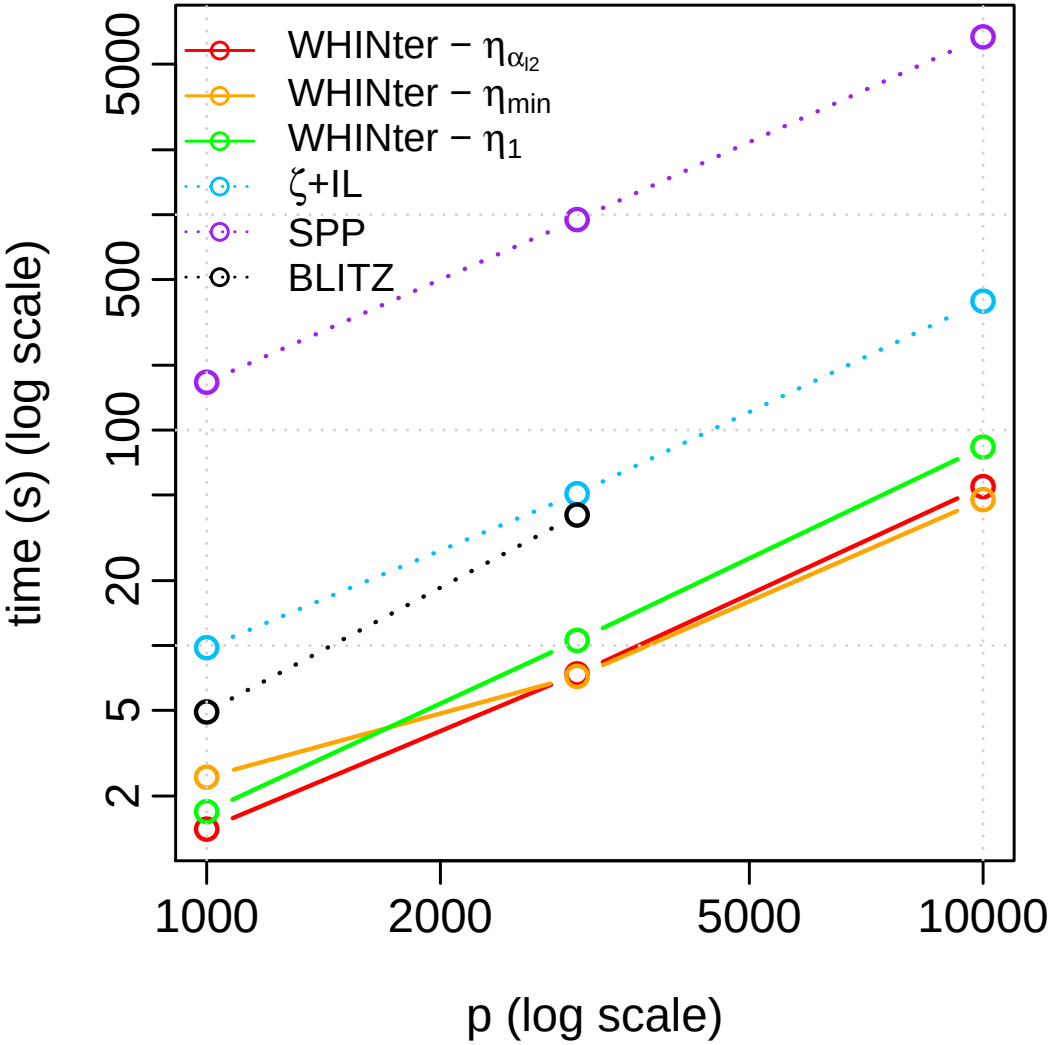
- Key ideas for a fast delineation of the working set
  - Branch bound
  - Maximum Inner Product Search (MIPS)
- Experimental results
  - Simulations
  - Preliminary results on real data

LASSO Simulations

- $\boldsymbol{X} \in \{0,1\}^{n \times p}$ where $\boldsymbol{X}_{ik} \sim \text{Bern}\,(q_k)$, and $q_k \sim \text{Unif}\,(0.1, 0.5)$.
- Randomly pick $\mathcal{S} \subset [\![D]\!]$ with $|\mathcal{S}| = 100$.
- $\boldsymbol{y} = \boldsymbol{Z}_{\mathcal{S}} \boldsymbol{w}^*_{\mathcal{S}}$ where $\boldsymbol{w}^*_{\mathcal{S}} \sim \mathcal{N}(\boldsymbol{0}_{|\mathcal{S}|}, I_{|\mathcal{S}|})$
- Take 100 values of λ logarithmically spaced in $[\lambda_{max}, 0.01\lambda_{max}]$.
- Algorithm stopped as soon as 150 features or more are selected in the model.

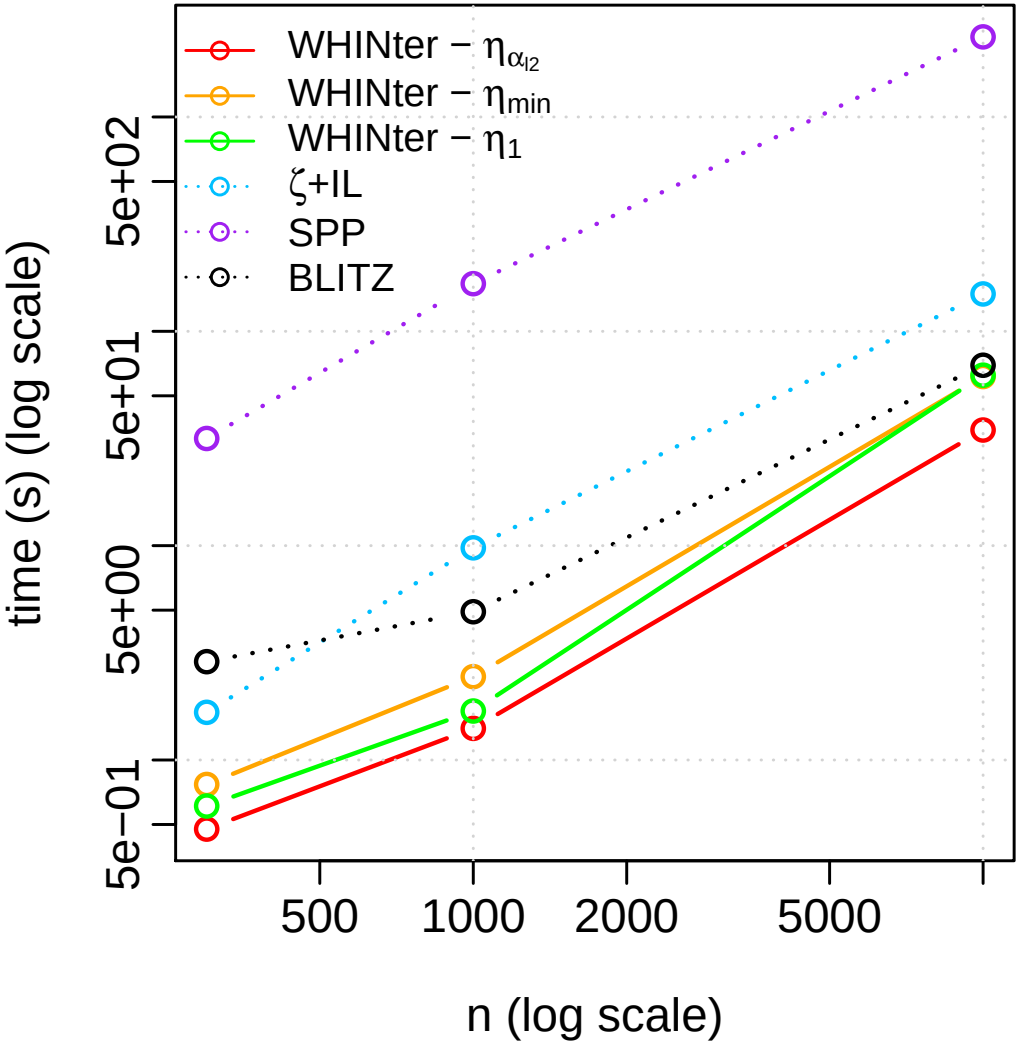
LASSO Simulations

$n = 1000,\ p \in \{1000, 3000, 10000\}$.



LASSO Simulations

$p = 1000,\ n \in \{300, 1000, 10000\}$.



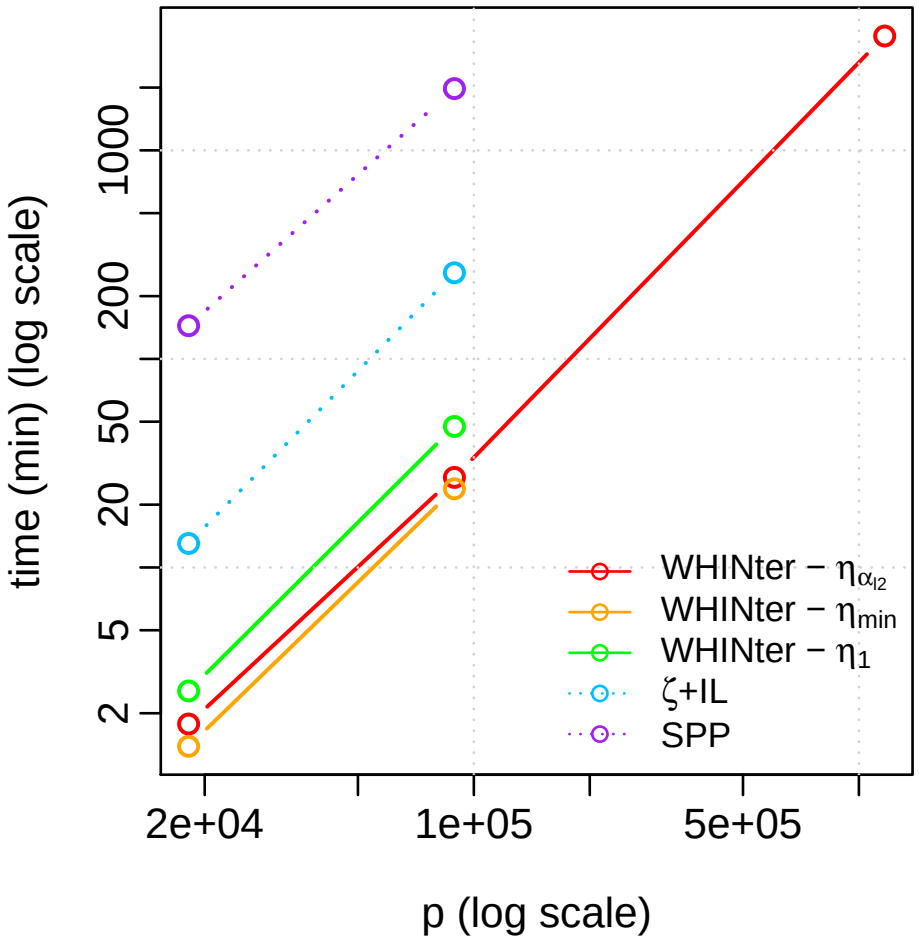
Outline

WHInter

- Key ideas for a fast delineation of the working set
  - Branch bound
  - Maximum Inner Product Search (MIPS)
- Experimental results
  - Simulations
  - Preliminary results on real data

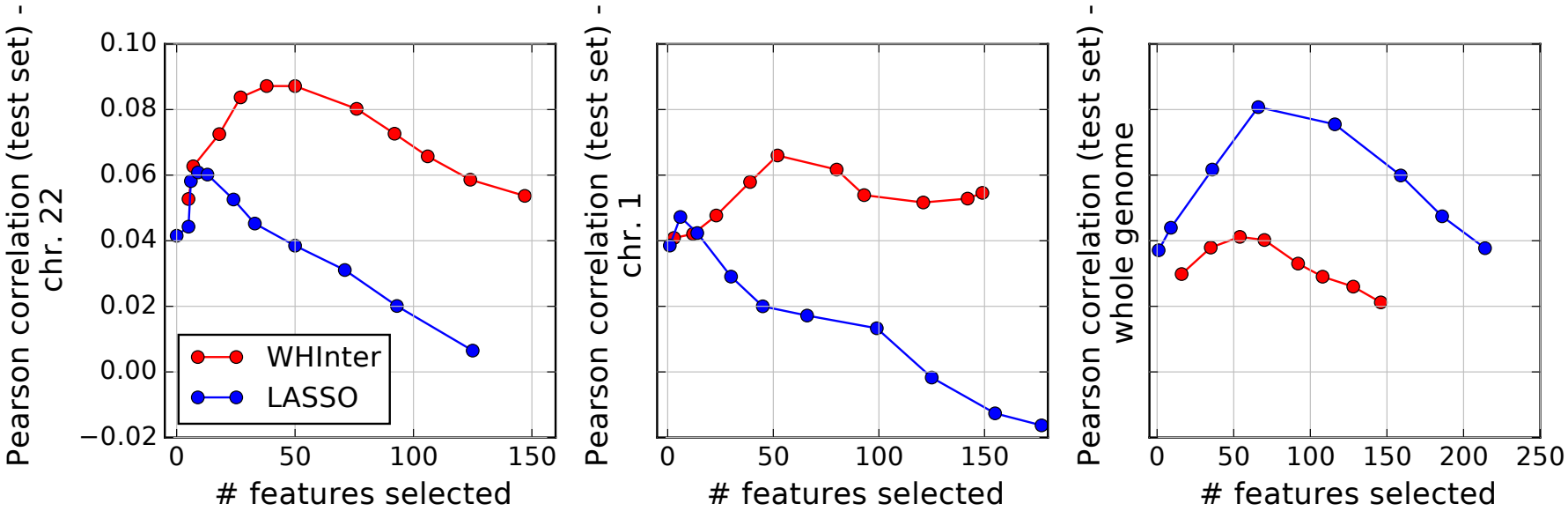
Results on Dream 8 toxicogenetics data - Scalability

- 884 lymphoblastoid cell lines: $n_{train} = 620$, $n_{test} = 264$.
- We consider the SNPs from:
  - ✓ chromosome 22 ($p = 18,168$)
  - ✓ chromosome 1 ($p = 89,027$)
  - ✓ all chromosomes ($p = 1,166,836$)
- The response $\boldsymbol{y}$ is the cytotoxicity (EC10) of a chemical compound (phenanthroline).
- Correction for population structure applied as in *Price et al. 2006*.



Results on Dream 8 toxicogenetics data - Predictive performance

- Preliminary results regarding the predictive performance of the LASSO with or without interactions.



- Interactions between SNPs seem to carry useful predictive signal, which can be practically captured by WHInter, at least on separated chromosomes.
- The poor performance of WHInter on all chromosomes illustrate the difficulty to learn when there are two many noise "junk" variables.

Conclusion

- Two words on Suquan:
  - ✓ Jointly learns a normalisation scheme and the weights of a linear model for HD genomic data.
  - ✓ Applied with success to cancer relapse prediction from gene expression data.
- High dimensionality is a core challenge in genomics:
  - ✓ Somatic mutations: a few pathogenic alterations with little overlap.
  - ✓ SNPs: a large number of variants with weak signal. Computational challenge to look for potential stronger signals in interactions.
- As sequencing costs continue to decrease, exciting days ahead to make clinical innovations possible!
  - ✓ Patient stratification and choice of treatment, clinical management, ...
  - ✓ Trait prediction and screening management, choice of dosage, ...

List of publications:

Marine Le Morvan, Andrei Zinovyev and Jean-Philippe Vert (2017). "NetNorM: Capturing cancer-relevant information in somatic exome mutation data with gene networks for cancer stratification and prognosis". In: PLoS Comput. Biol. 13.6, e1005573

Marine Le Morvan and Jean-Philippe Vert (2018). "WHInter: A Working set algorithm for High-dimensional sparse second order interaction models". In ArXiv e-prints. arXiv: 1802.05980 (accepted to ICML 2018)

Marine Le Morvan and Jean-Philippe Vert (2018). "Supervised Quantile Normalisation". In ArXiv e-prints. arXiv: 1706.00244

Thanks!

El Ghaoui, Laurent, Vivian Viallon, and Tarek Rabbani (2012). "Safe feature elimination in sparse supervised learning". In: *Pacific J. Optim.* 8.4, pp. 667–698.

Fercoq, Olivier, Alexandre Gramfort, and Joseph Salmon (2015). "Mind the Duality Gap: Safer Rules for the Lasso". In: *Proc. 32nd Int. Conf. Mach. Learn.* Pp. 333–342.

Friedman, Jerome, Trevor Hastie, and Robert Tibshirani (2010). "Regularization Paths for Generalized Linear Models via Coordinate Descent". In: *J. Stat. Softw.* 33.1, pp. 1–22.

Hofree, Matan, John P Shen, Hannah Carter, Andrew Gross, and Trey Ideker (2013). "Network-based stratification of tumor mutations". In: *Nat. Methods* 10.11, p. 1108.

Johnson, Tyler and Carlos Guestrin (2015). "Blitz: A Principled Meta-Algorithm for Scaling Sparse Optimization". In: *Proc. 32nd Int. Conf. Mach. Learn.* Pp. 1171–1179.

Massias, Mathurin, Alexandre Gramfort, and Joseph Salmon (2018). "Dual Extrapolation for Faster Lasso Solvers". In: *ArXiv e-prints*. arXiv: 1802.07481.

Nakagawa, Kazuya, Shinya Suzumura, Masayuki Karasuyama, Koji Tsuda, and Ichiro Takeuchi (2016). "Safe Pattern Pruning: An Efficient Approach for Predictive Pattern Mining". In: *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.* Pp. 1785–1794.

Price, Alkes L., Nick J. Patterson, Robert M. Plenge, Michael E. Weinblatt, Nancy A. Shadick, and David Reich (2006). "Principal components analysis corrects for stratification in genome-wide association studies". In: *Nat. Genet.* 38.8, pp. 904–909.

Shrivastava, Anshumali and Ping Li (2014). "Asymmetric LSH (ALSH) for sublinear time maximum inner product search (MIPS)". In: *Adv. Neural Inf. Process. Syst.* Pp. 2321–2329.

Teflioudi, Christina and Rainer Gemulla (2016). "Exact and Approximate Maximum Inner Product Search with LEMP". In: *ACM Trans. Database Syst.* 42.1, 5:1–5:49.

Van Belle, Vanya, Kristiaan Pelckmans, J.A.K. Suykens, and Sabine Van Huffel (2007). "Support vector machines for survival analysis". In: *Proc. 3rd Int. Conf. Comput. Intell. Med. Healthc.* Pp. 1–8.

Supplementaries

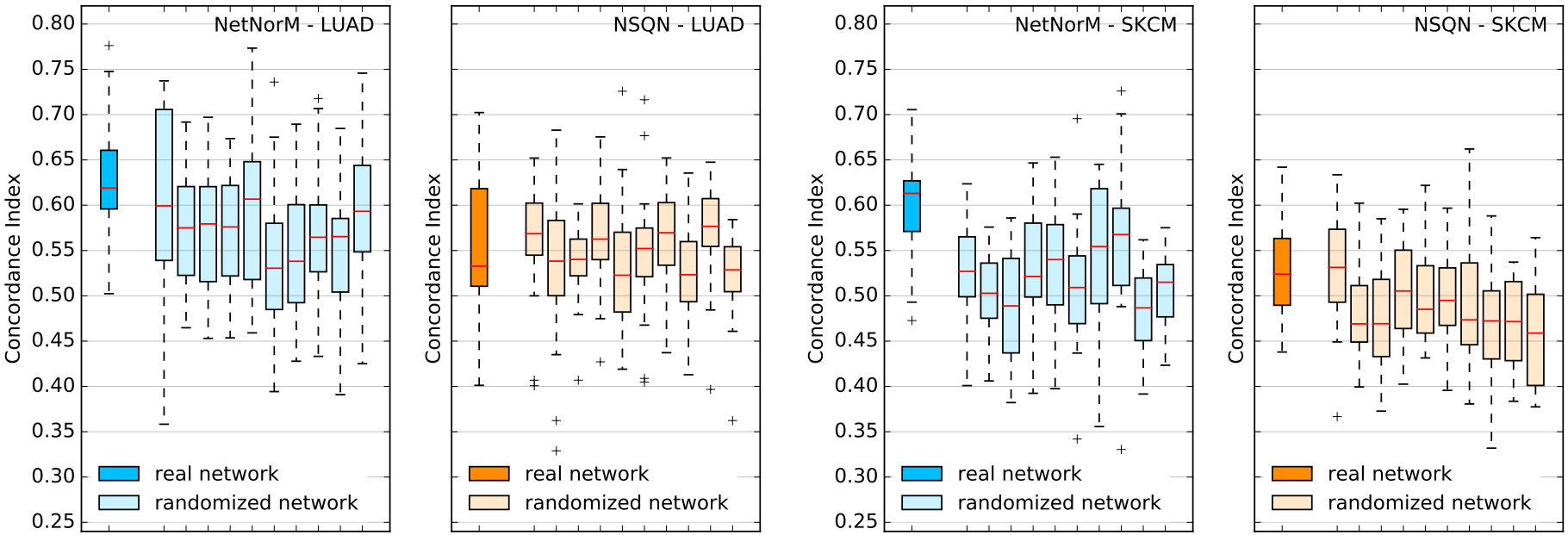
NetNorm

WHinter

Suquan

Randomised networks decrease survival prediction performance

10 randomised versions of Pathway Commons are generated by shuffling node labels.



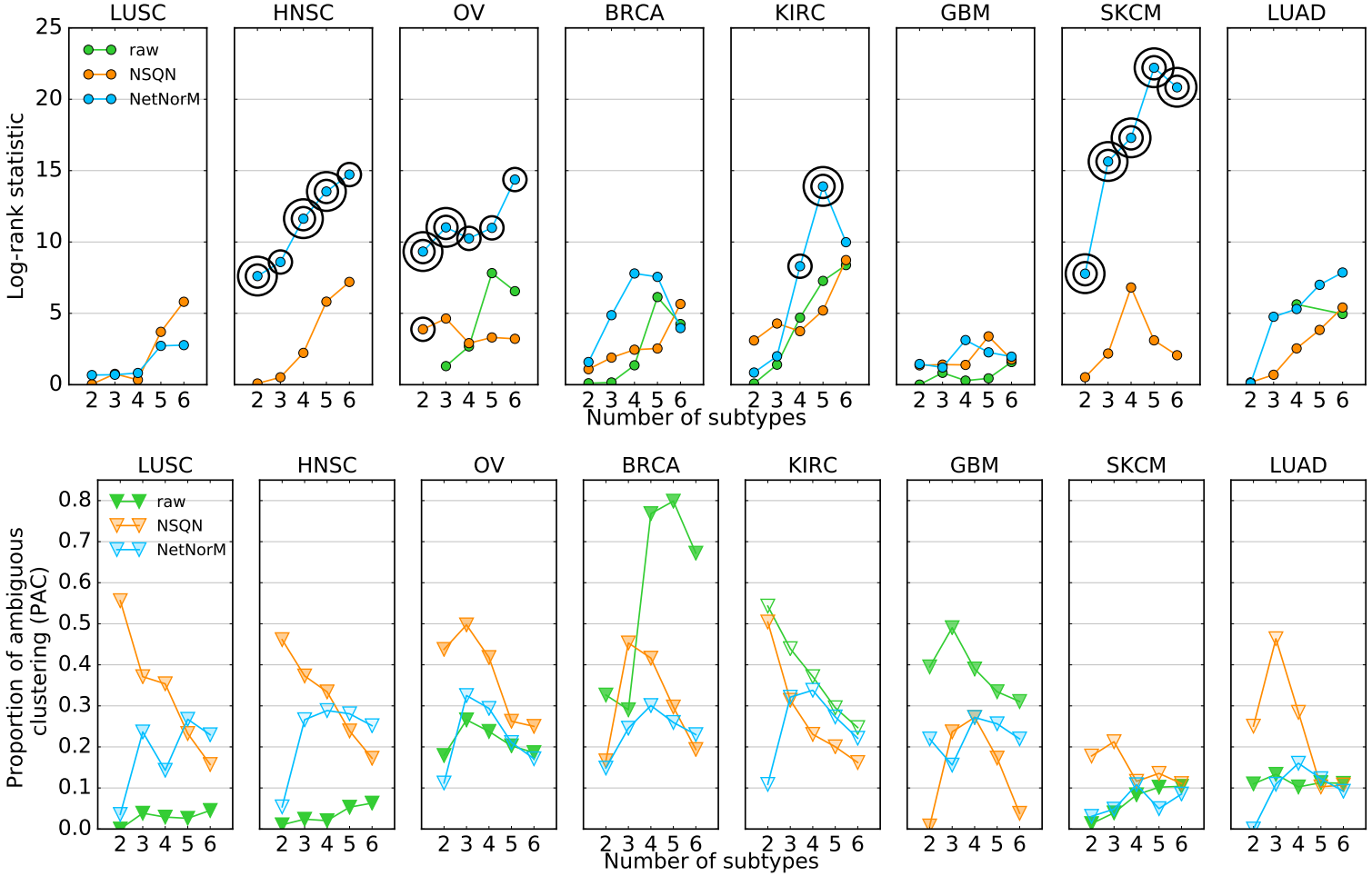
A Welsh t-test was performed to compare the performances obtained with randomised networks to that obtained with the real network.

	NSQN	NetNorM
LUAD	0.65	1.4×10^{-3}
SKCM	1×10^{-2}	1×10^{-5}

p-values

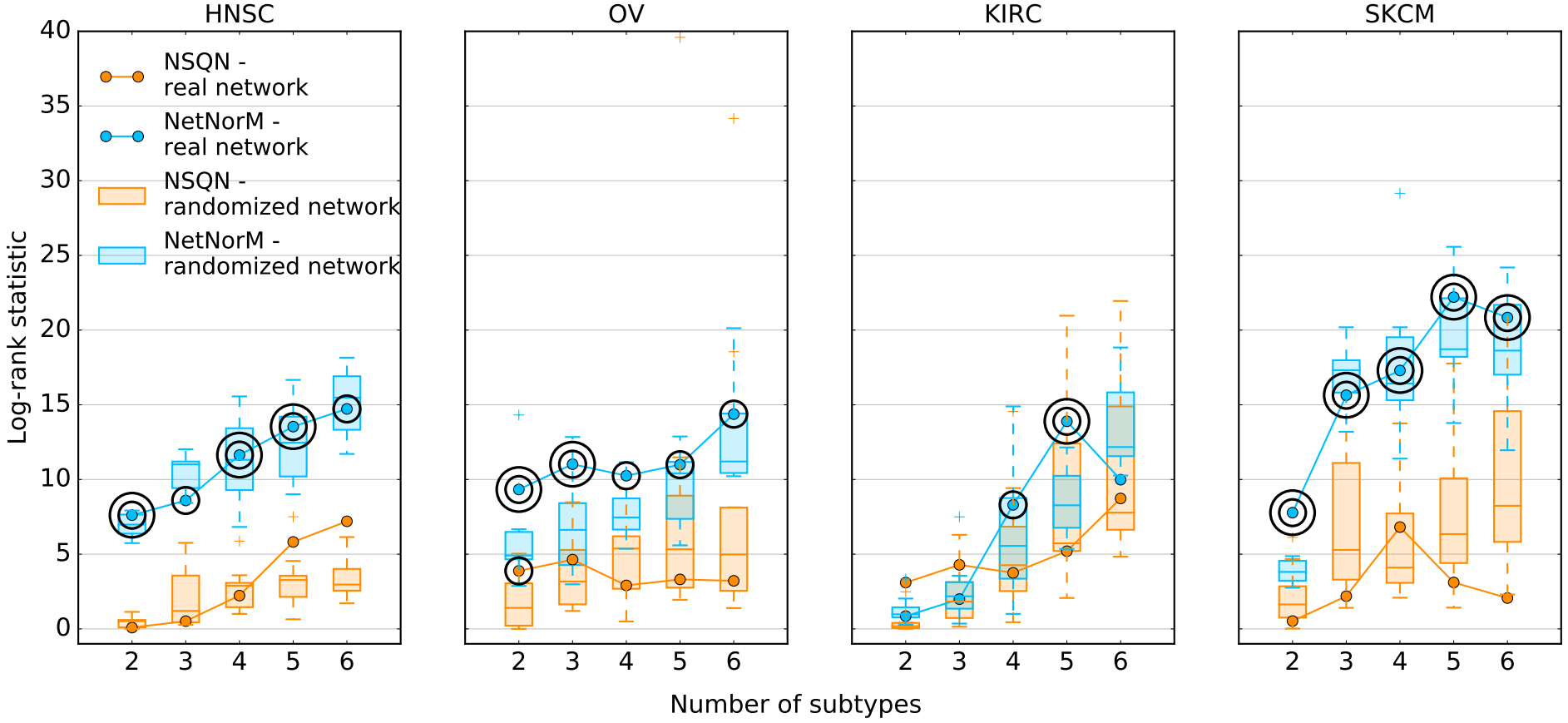
Patient Stratifications

(a) Association of patient subtypes with survival time



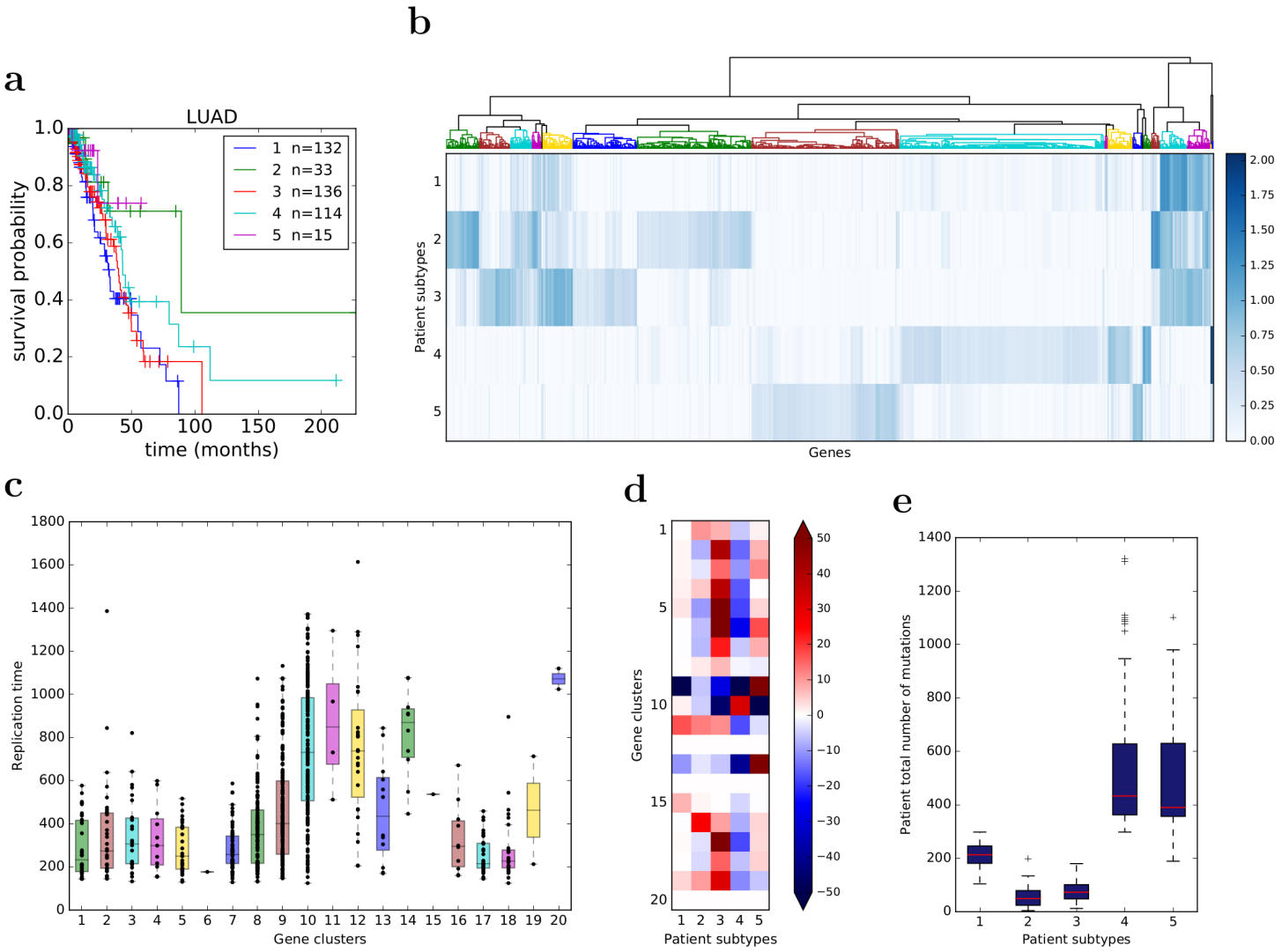
(b) Evaluation of the clustering stability as measured by the proportion of ambiguous clustering (PAC).

Patient Stratifications



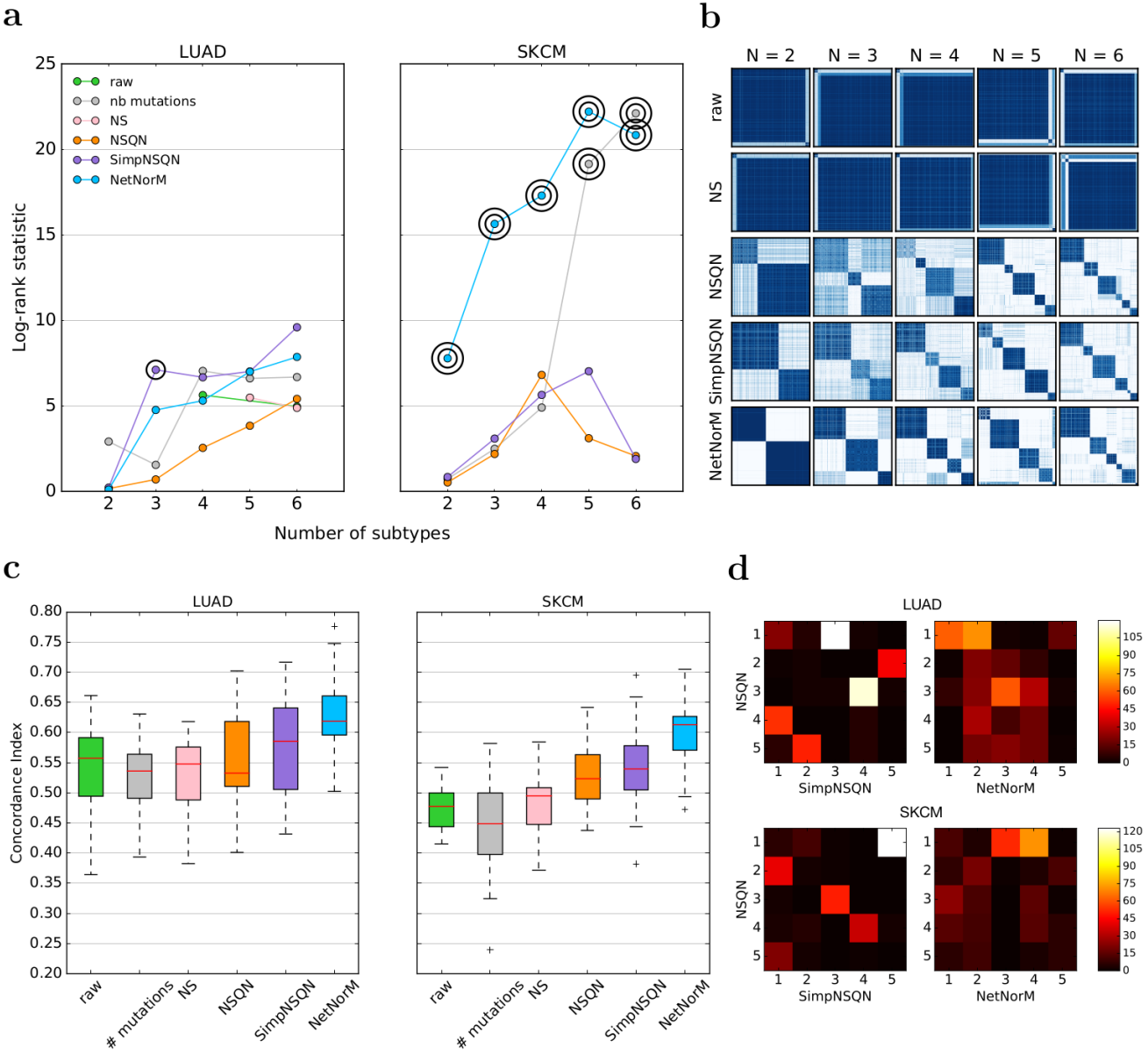
Log-rank statistic

Patient Stratifications

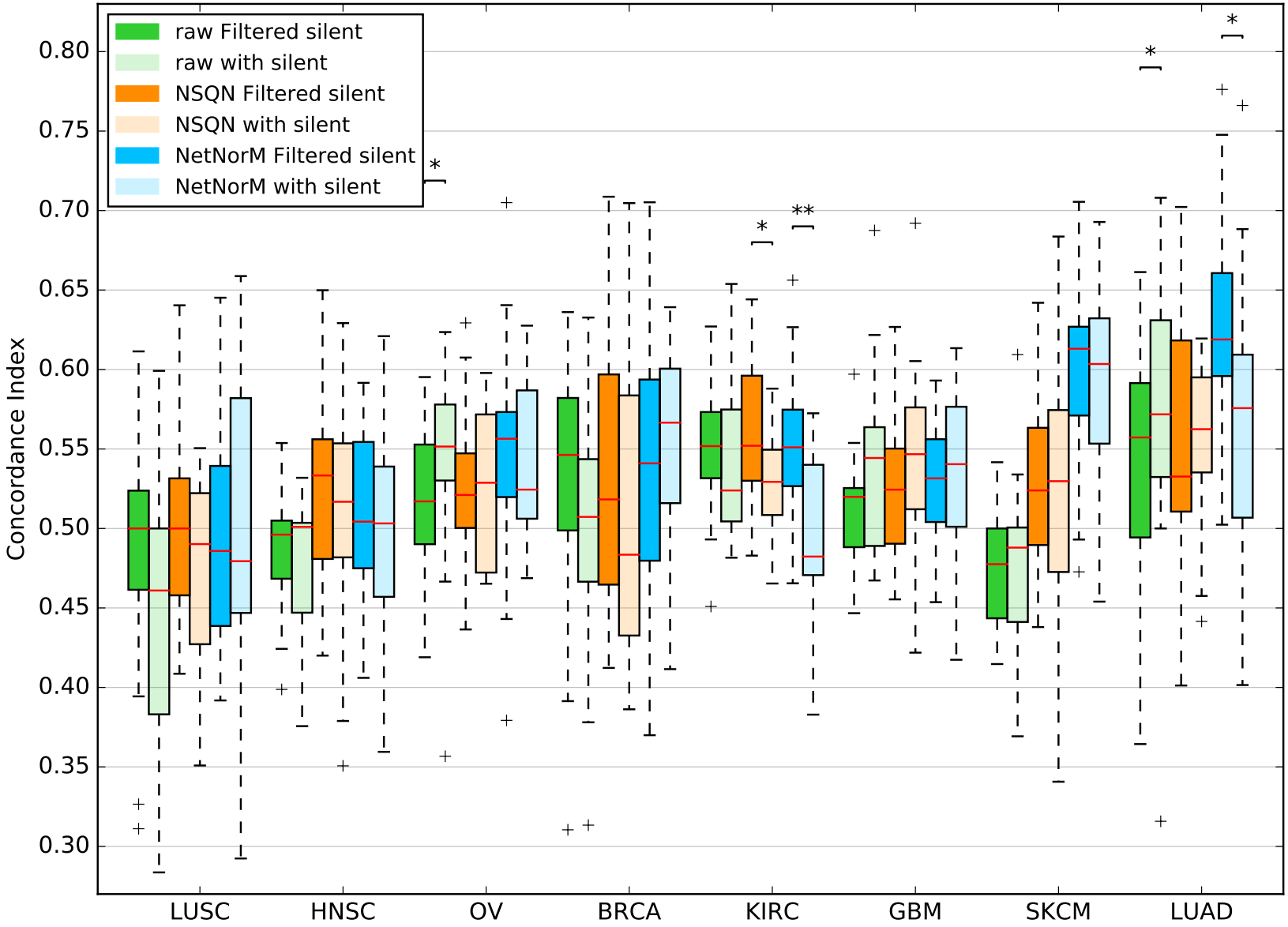


Characterisation of LUAD patient subtypes obtained with NetNorM

NSQN and NetNorM performances levers



Effect of silent mutations



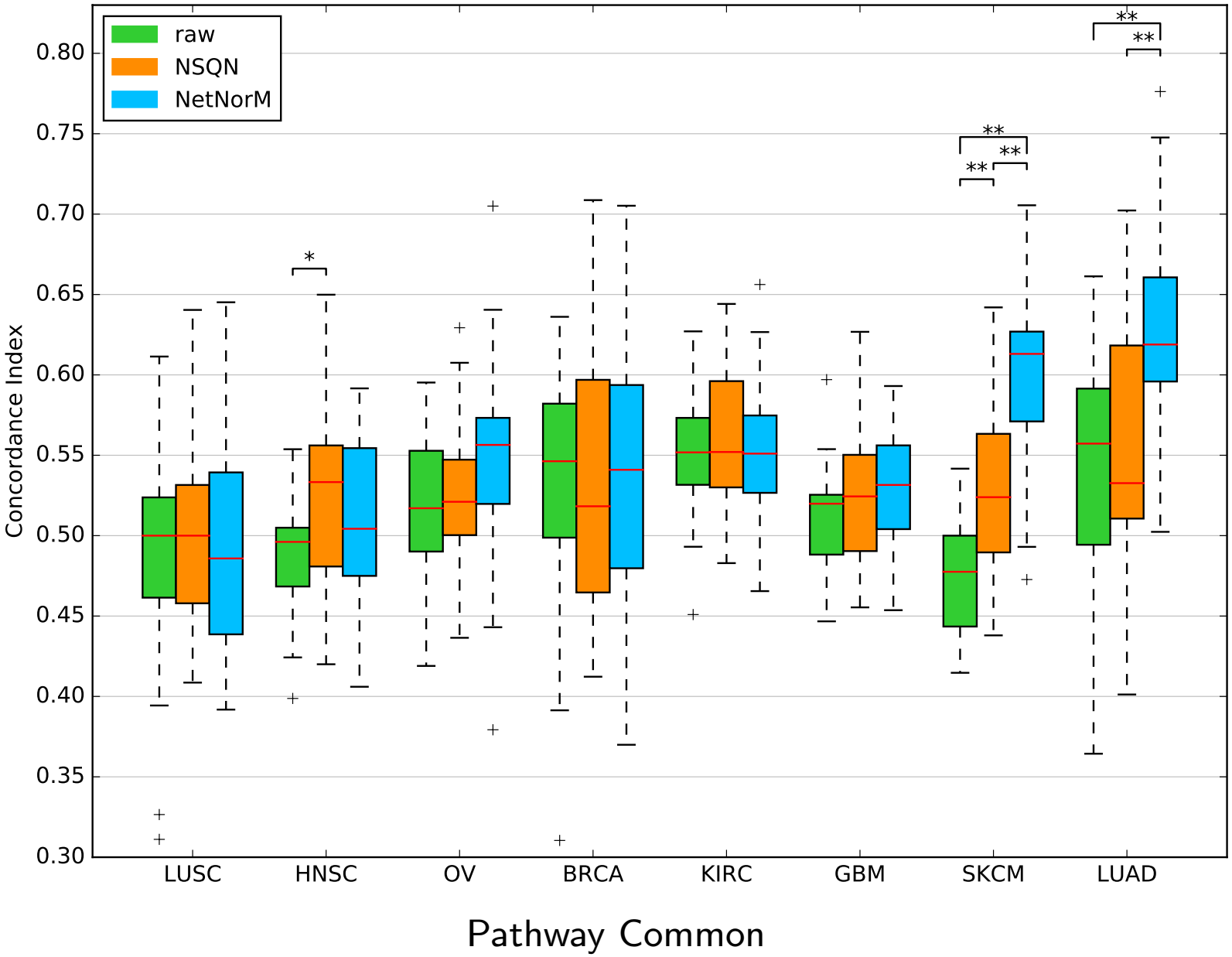
Effect of silent mutations on the survival predictive power

Gene selection

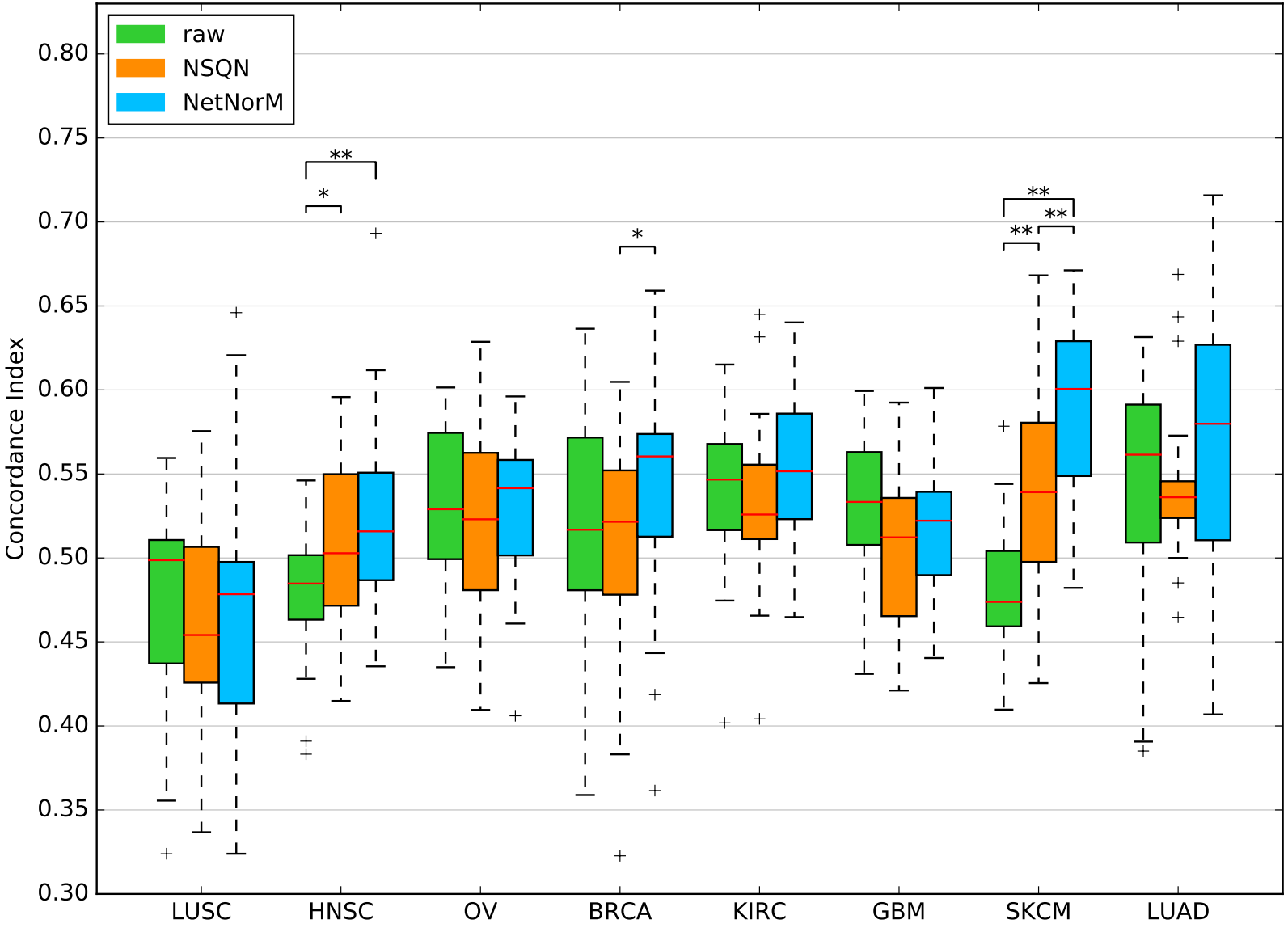
LUSC		HNSC		OV		BRCA		KIRC		GBM		SKCM		LUAD	
TTN	6	TP53	17	TTN	19	TP53	19	BAP1	19	TP53	10	PCDHGC5	10	ANK2	4
COL11A1	3	CACNA2D1	1	BRCA2	1	TTN	1	PBRM1	1	IDH1	6	FLNC	5	RYR2	4
FAM5C	3	MUC16	1							ITSN2	2	COL3A1	2	CRB1	4
PCDHAC2	3	NEB	1							PLEC	1	PCDHB5	1	TP53	3
ANK2	3									EDA	1	SCN11A	1	LAMA2	2
TP53	1											KIAA1217	1	HMCN1	1
RP1	1													USH2A	1
														LARP1	1

Table S1. Summary of the genes selected when only one gene is used to predict survival. For each gene the number of folds (out of 20 folds) where the gene is selected is indicated.

Effect of the gene network

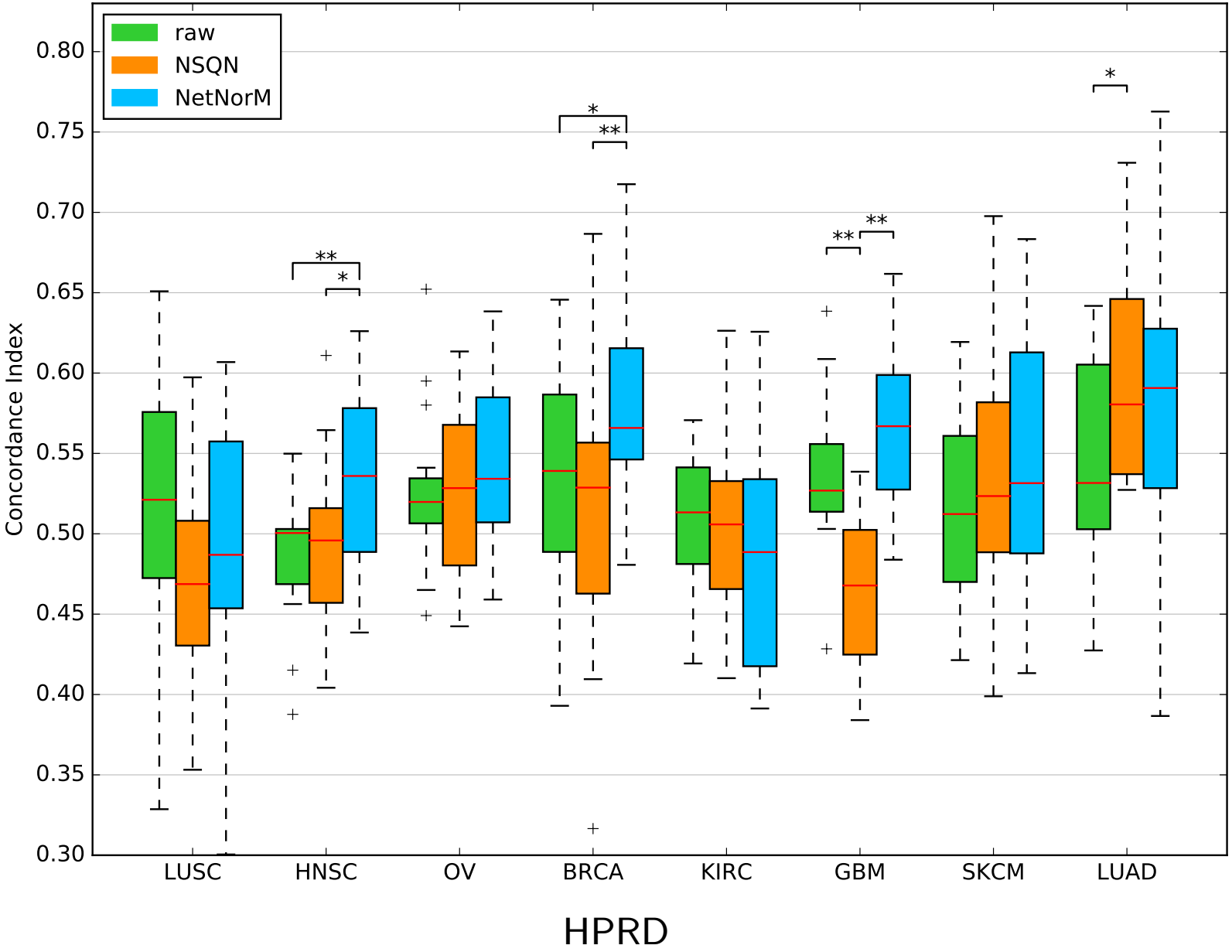


Effect of the gene network

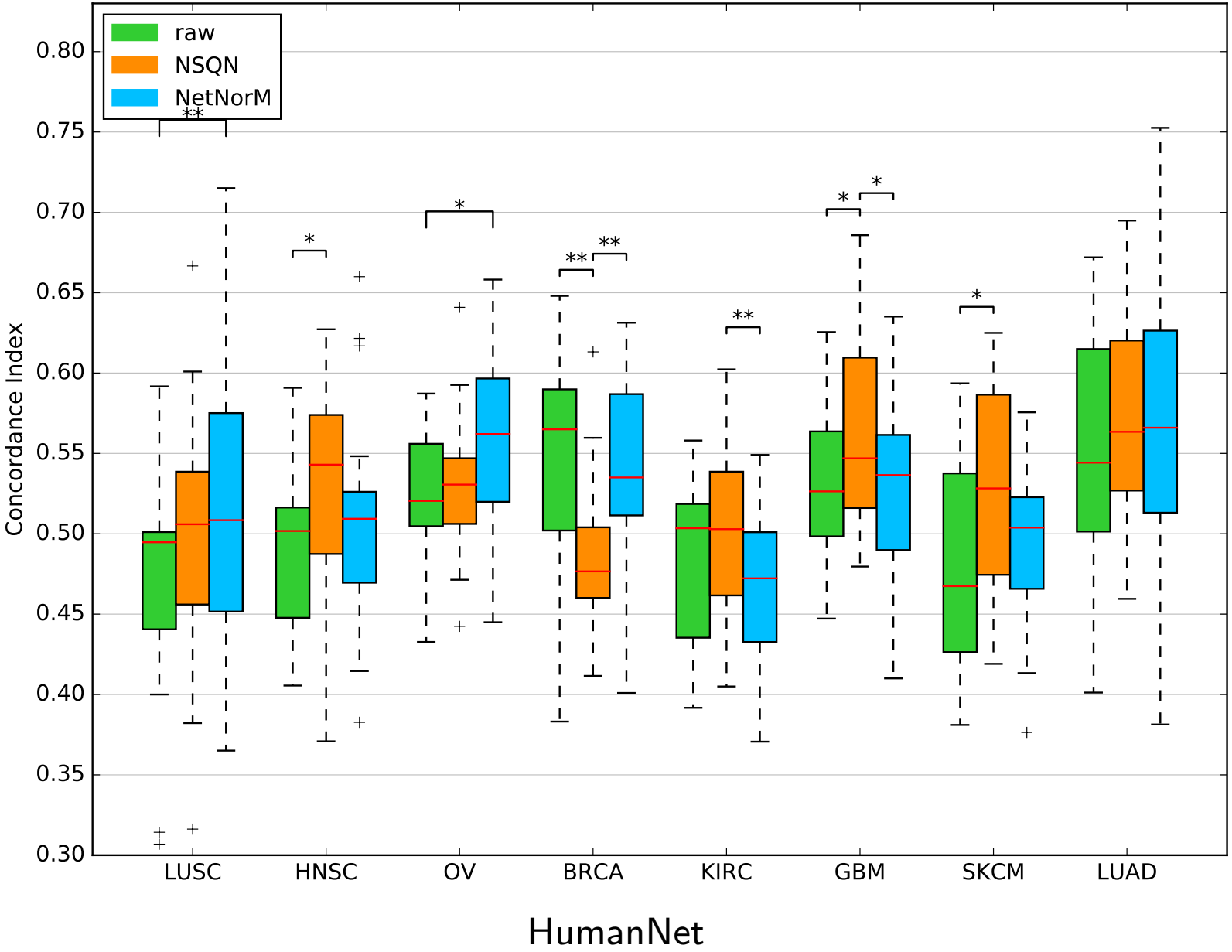


BioGRID

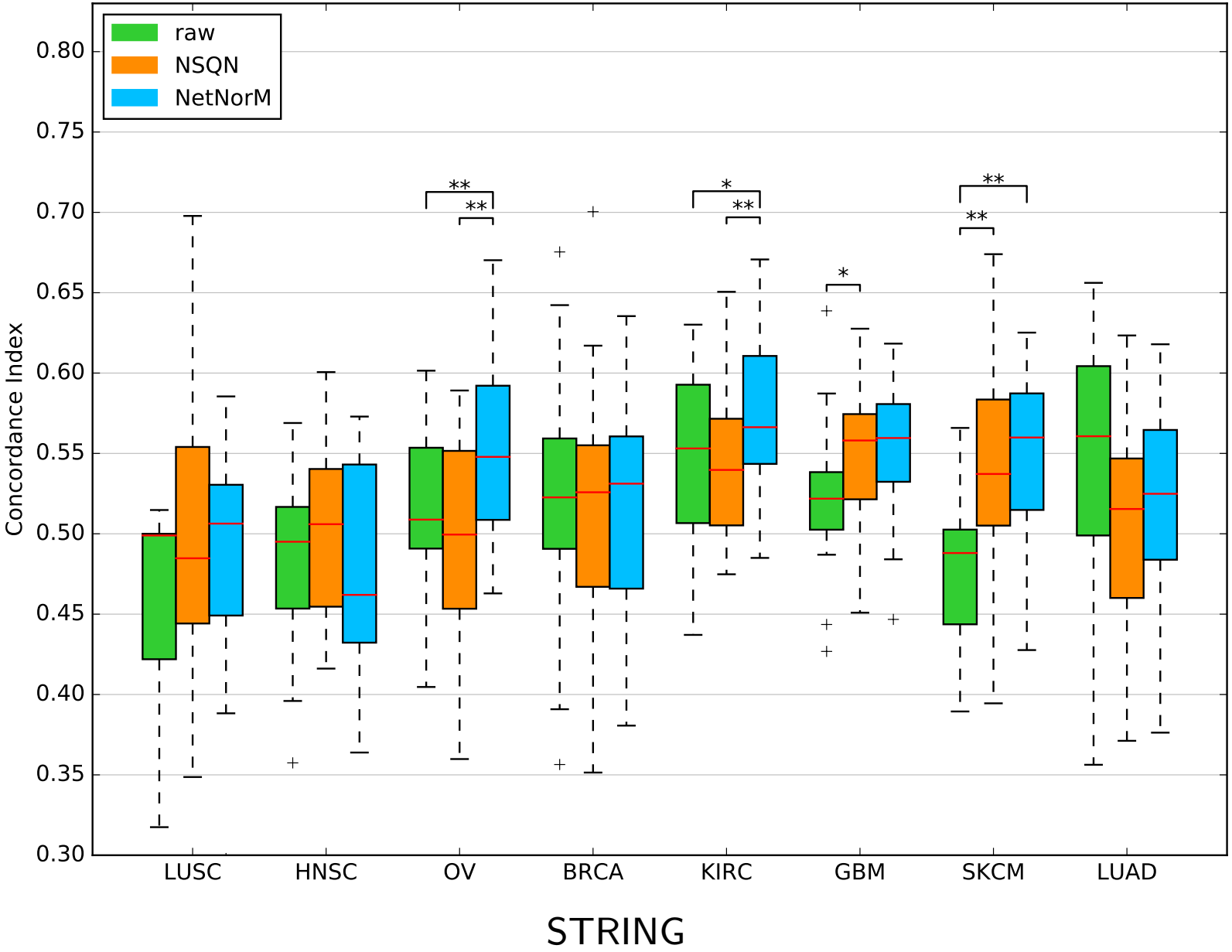
Effect of the gene network



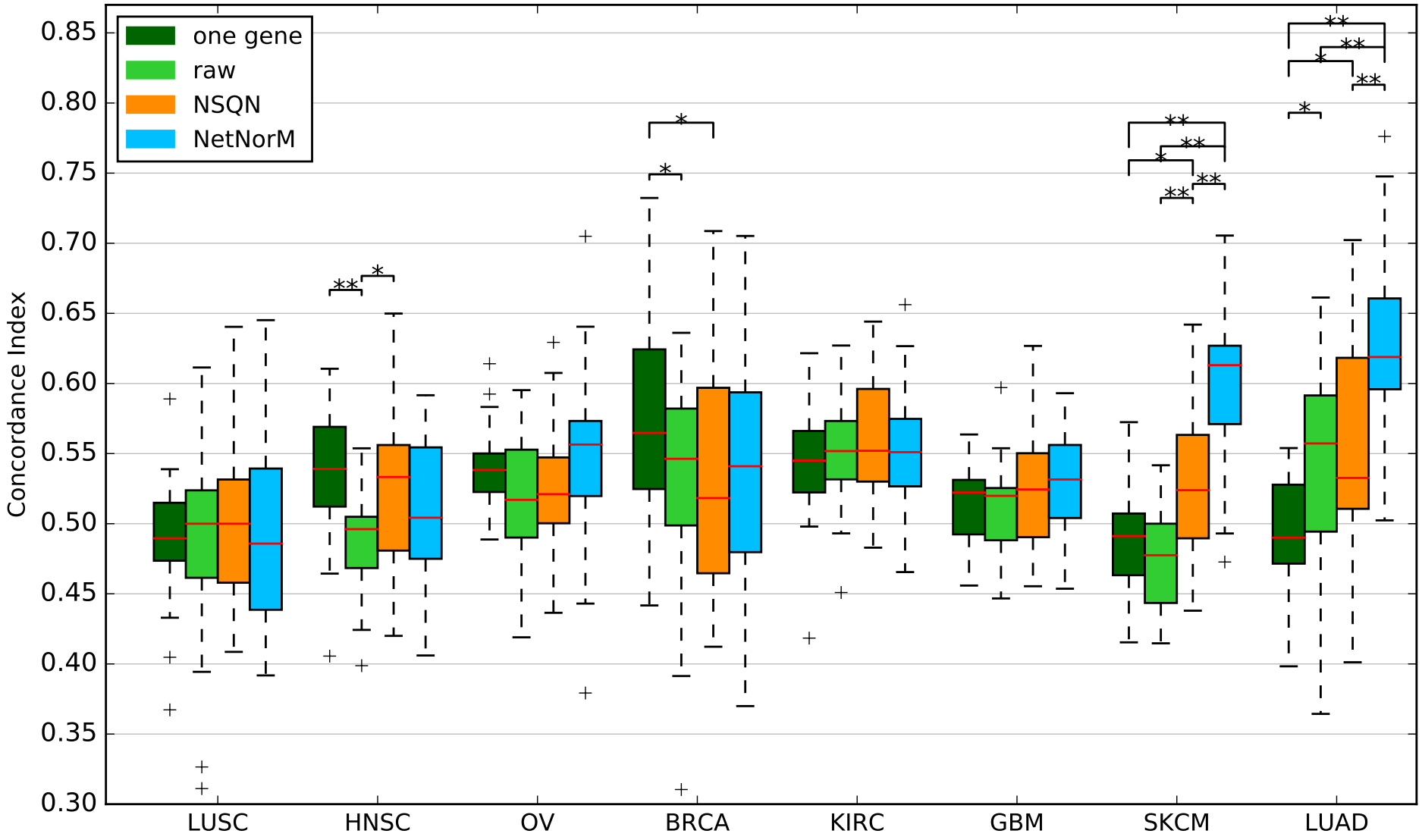
Effect of the gene network



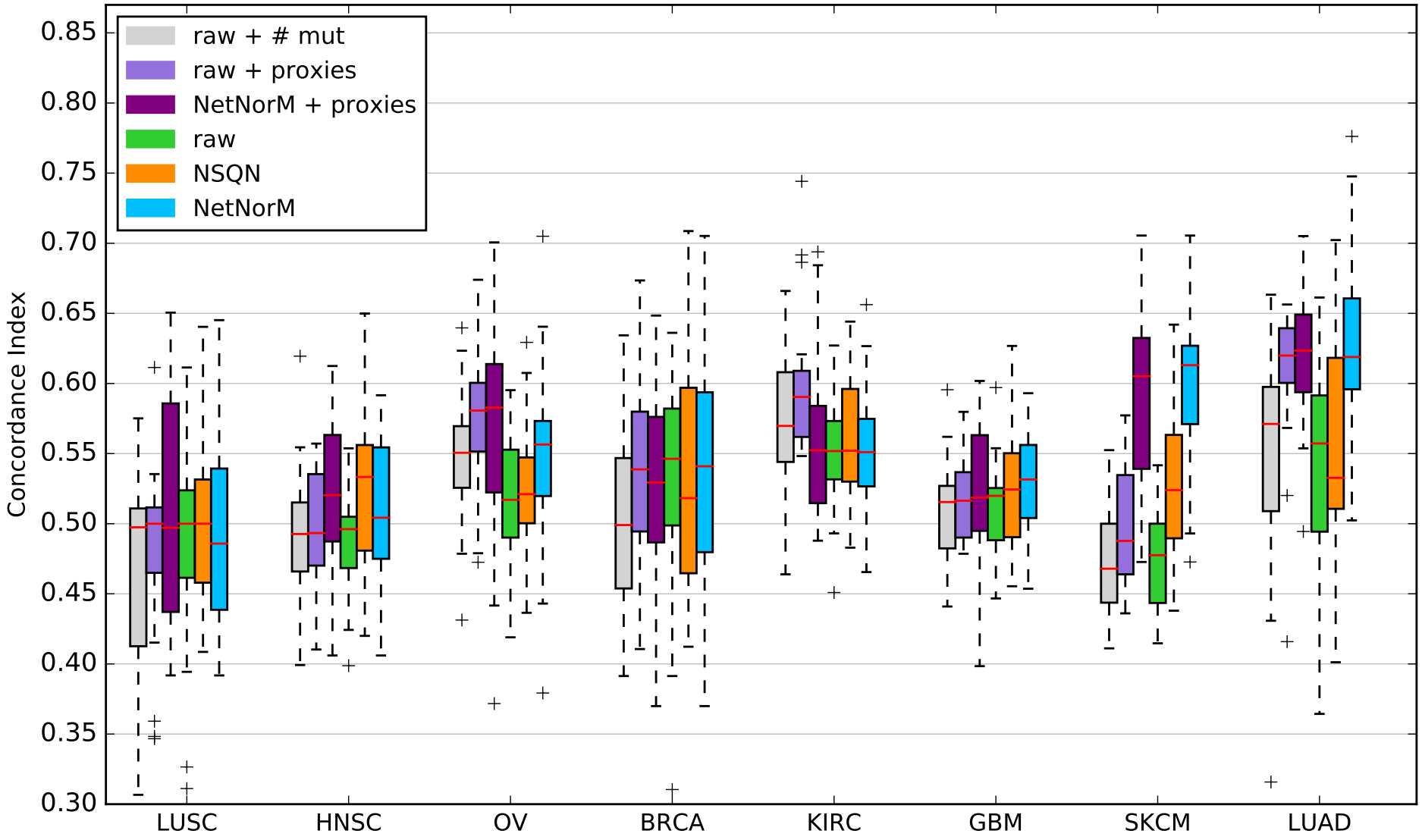
Effect of the gene network



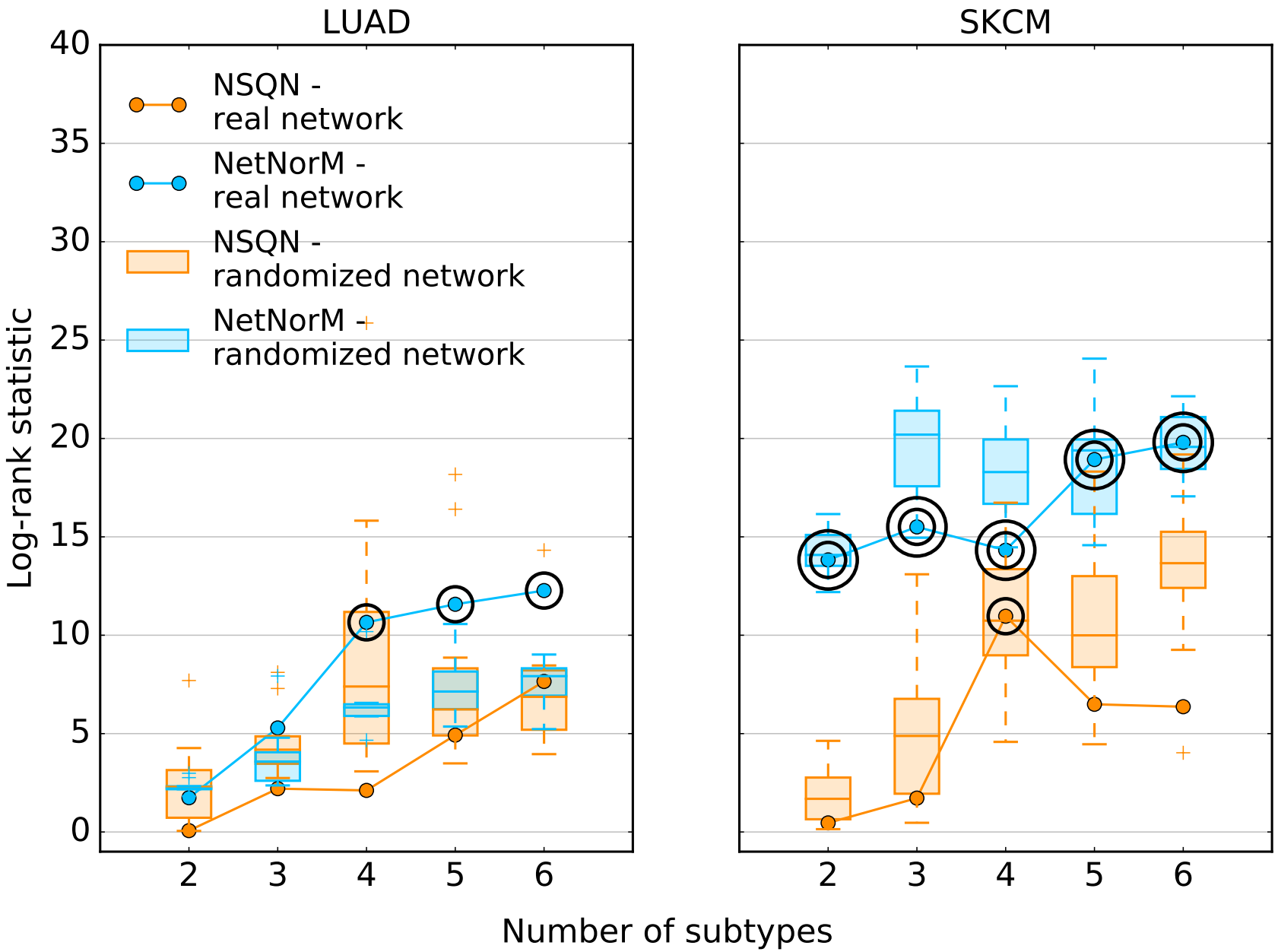
Survival predictive power: mutation and gene selection



Survival predictive prower: preprocessing steps



Randomized network



Randomized network

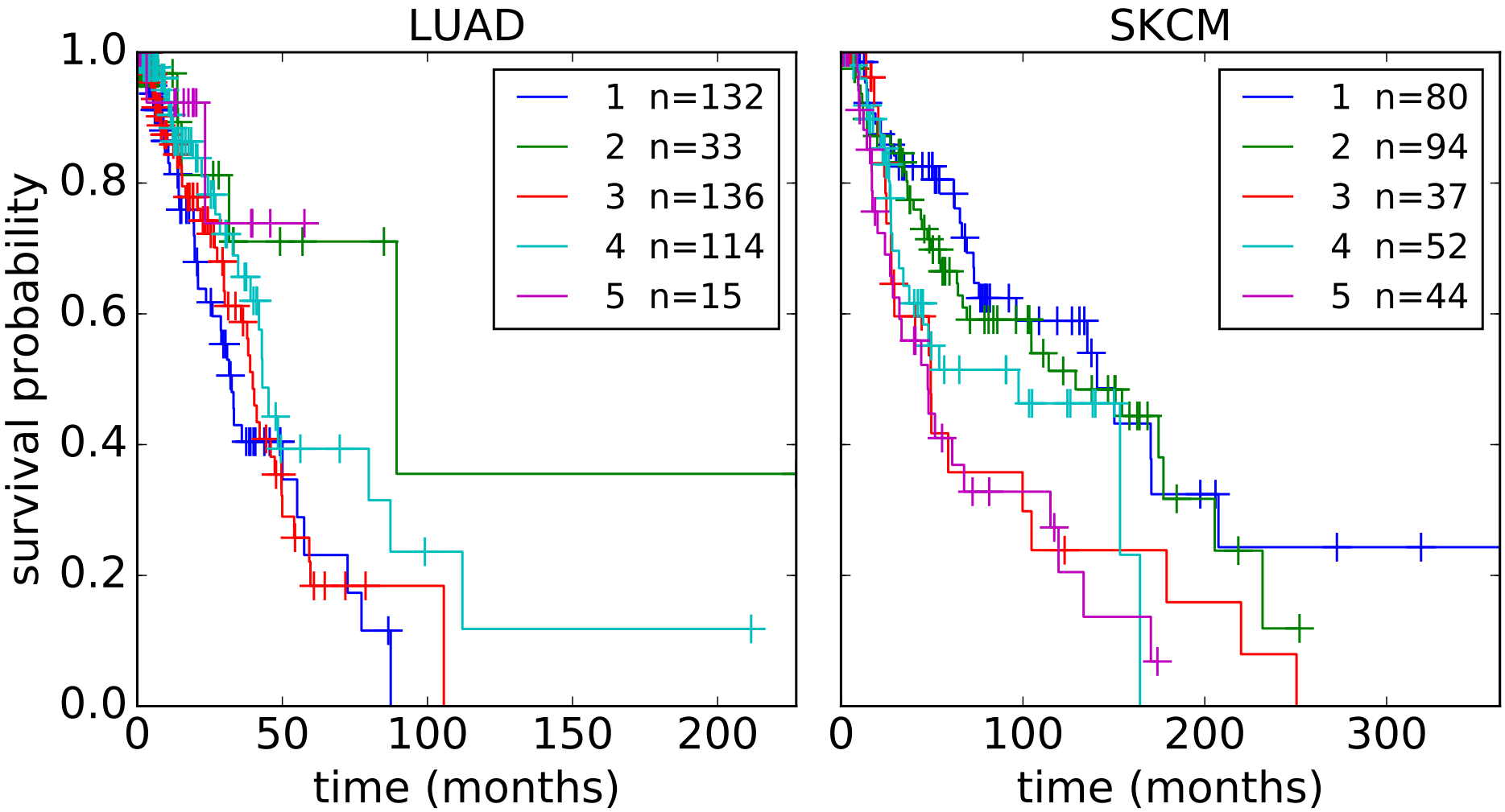
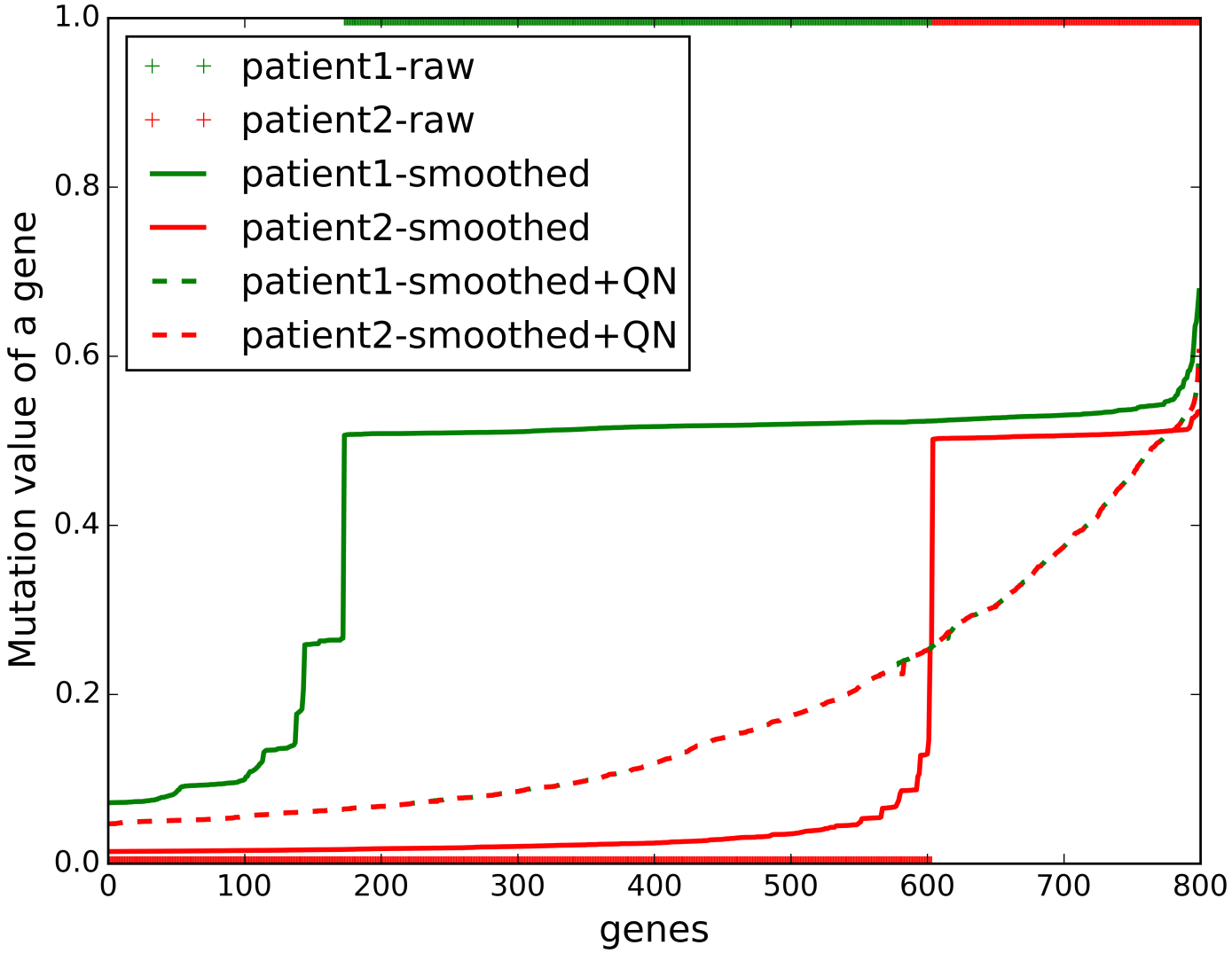
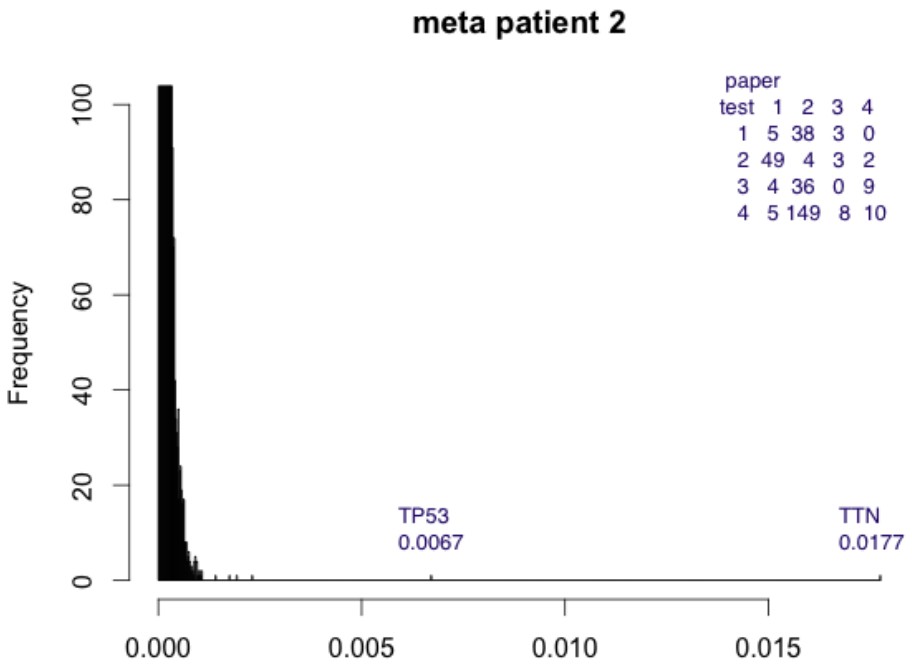


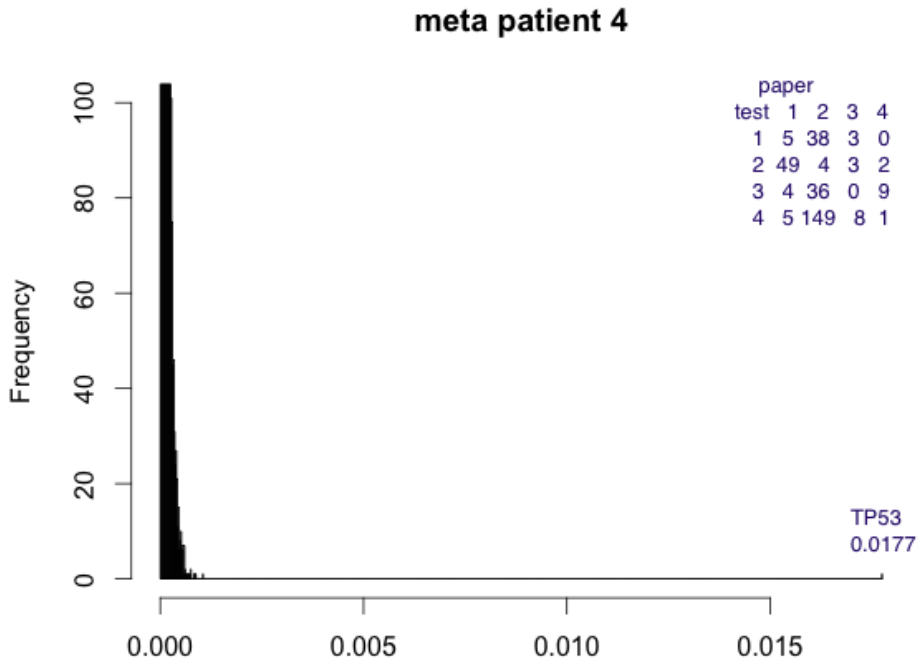
Illustration of quantile normalisation after network smoothing



Reproducing Hofree et al. results for ovarian cancer



Metapatient corresponding to Hofree's cluster 1



Metapatient corresponding to Hofree's cluster 2

- The **2 main clusters** seem to be mainly **driven by mutations in TP53 and TTN**.

	TTN: 0	TTN: 1
TP53: 0	1	6
TP53: 1	0	56

Contingency table for Hofree cluster 1

	TTN: 0	TTN: 1
TP53: 0	33	1
TP53: 1	186	7

Contingency table for Hofree cluster 2

Supplementaries

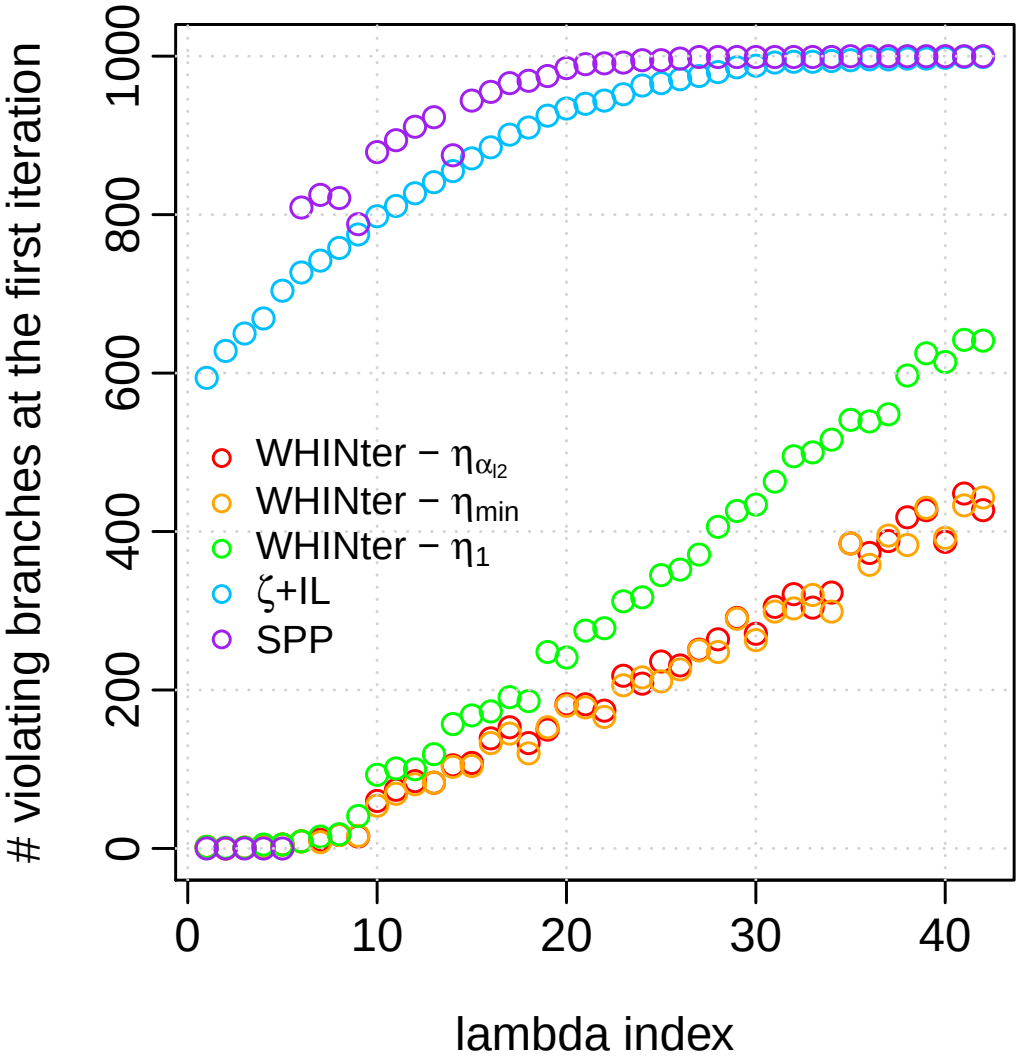
NetNorm

WHinter

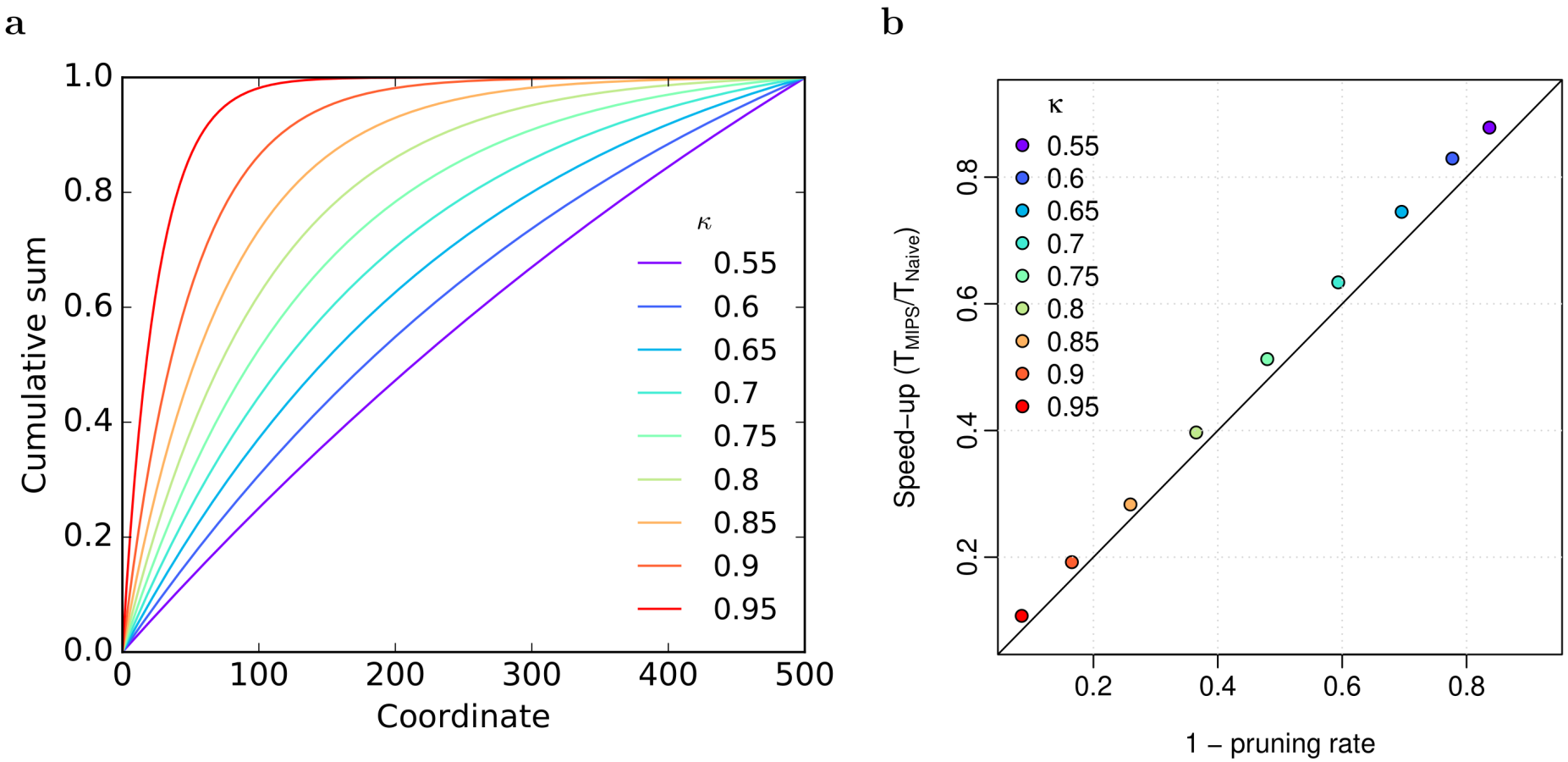
Suquan

LASSO Simulations

$n = 1000,\ p = 10000.$

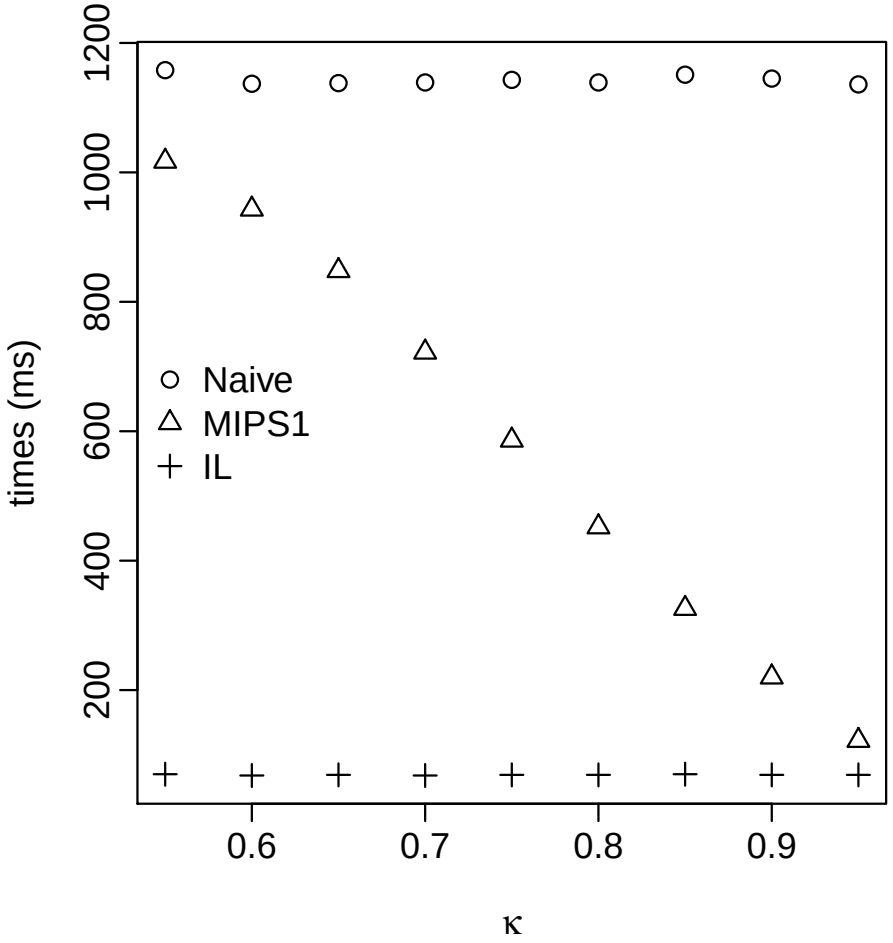


Performance of the MIPS



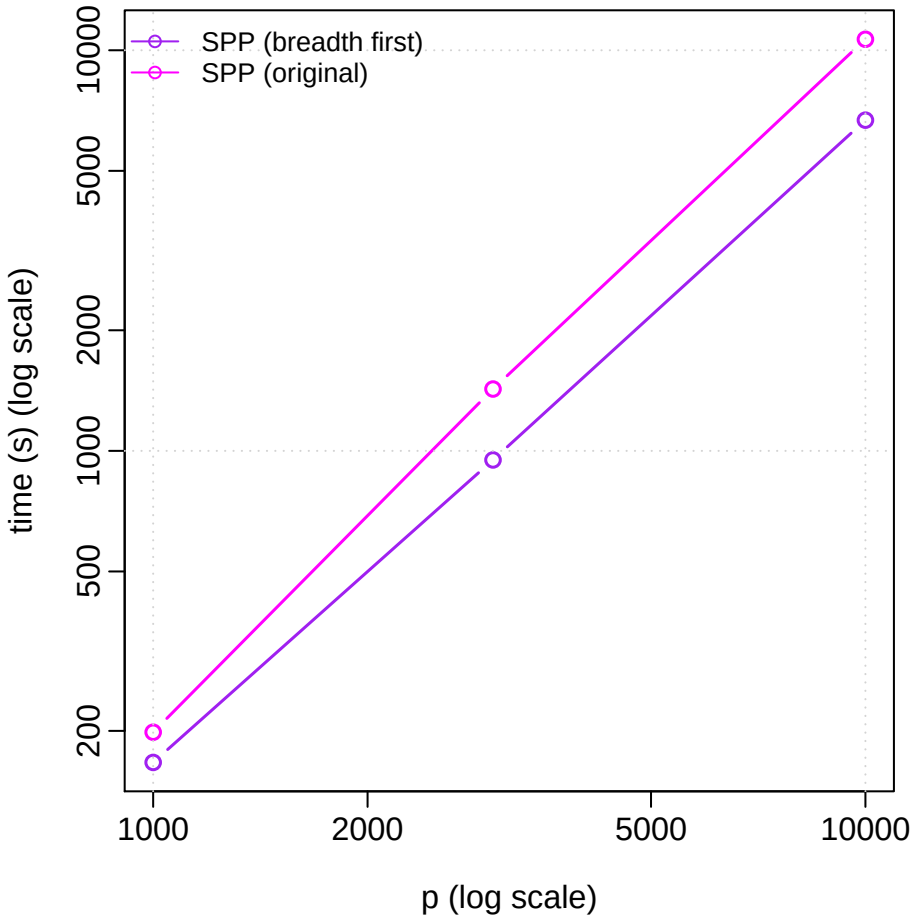
Performances of MIPS on simulated data

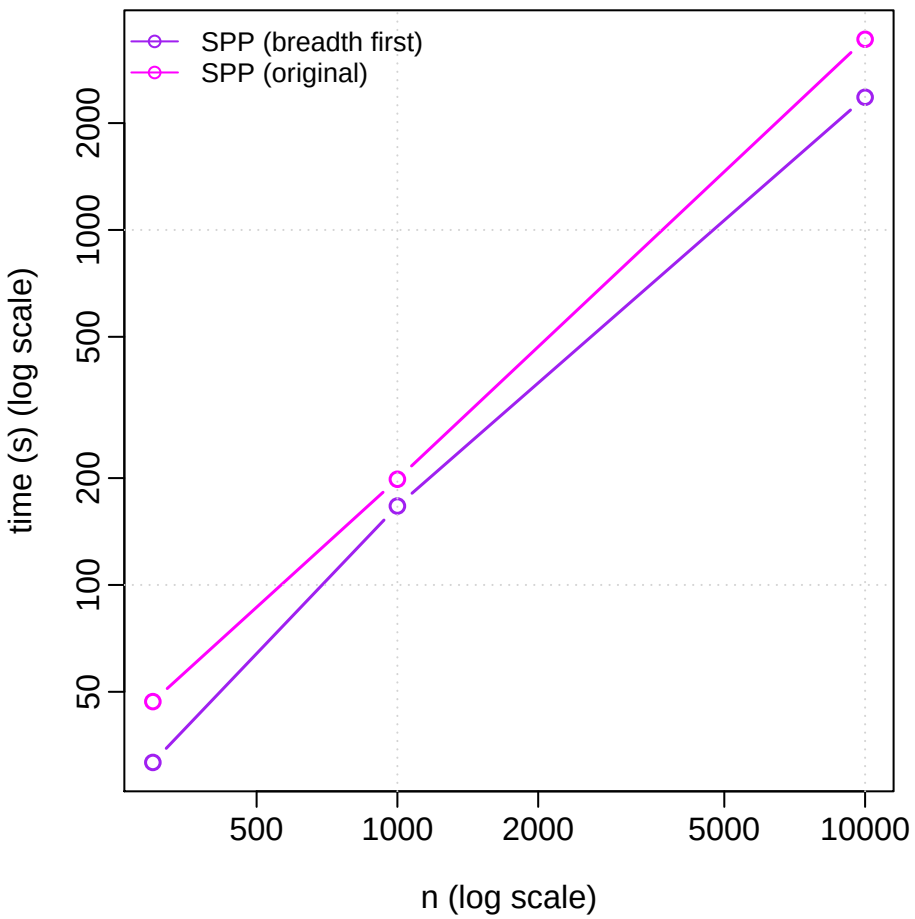
Performance of the MIPS



Performance comparisons on simulated data for MIPS, IL and naive MIPS

Safe Pattern Pruning





SPP performances on simulated data for an entire regularisation path

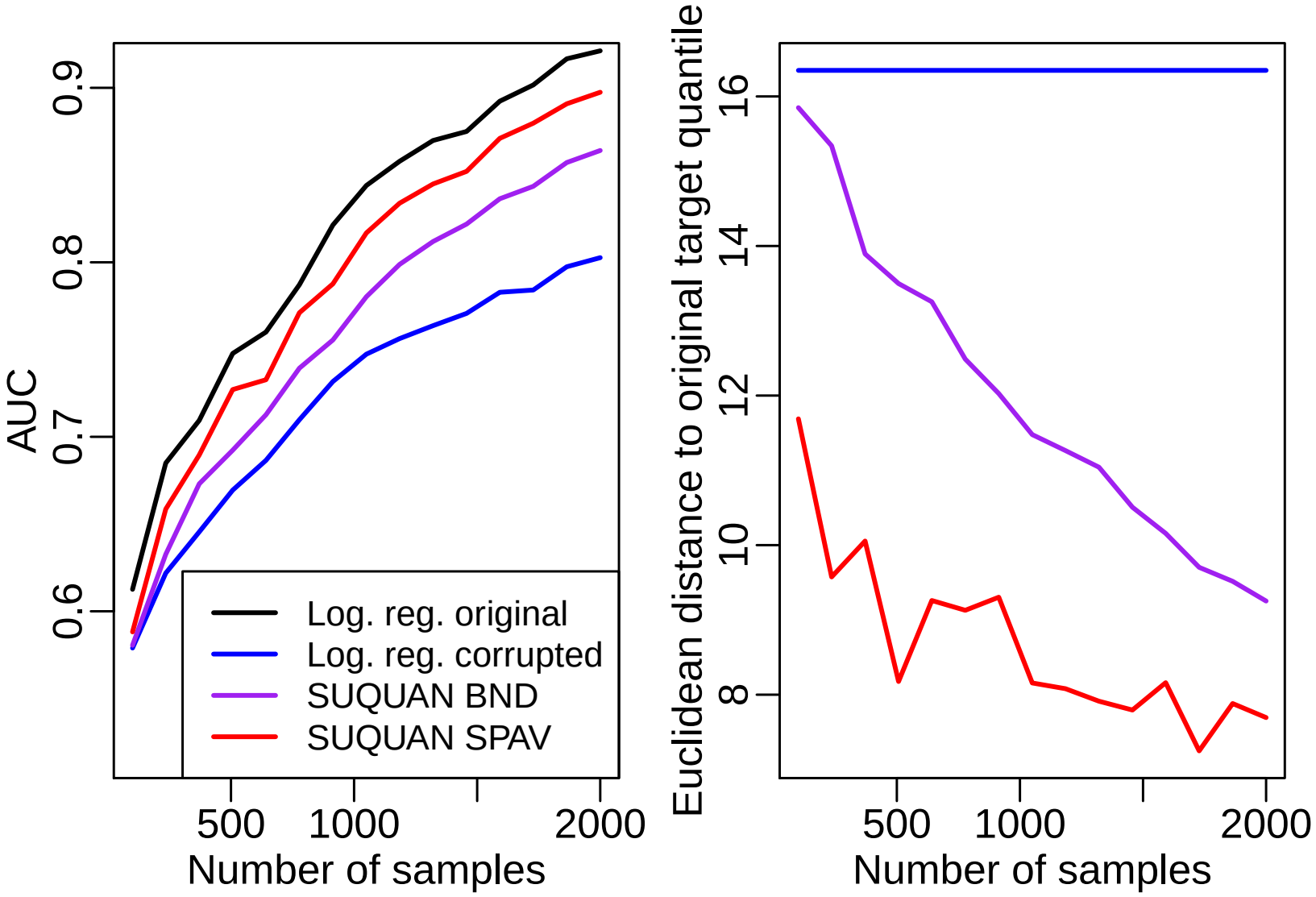
Supplementaries

NetNorm

WHinter

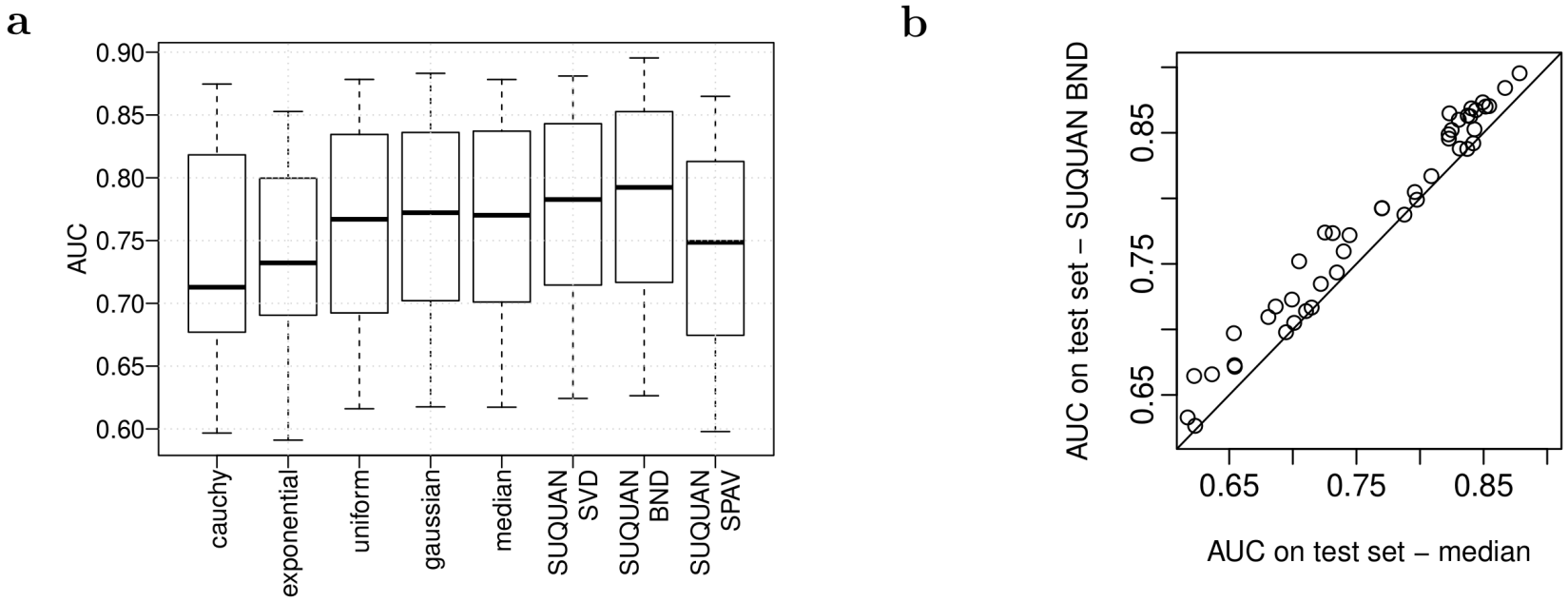
Suquan

Suquan: simulated datasets



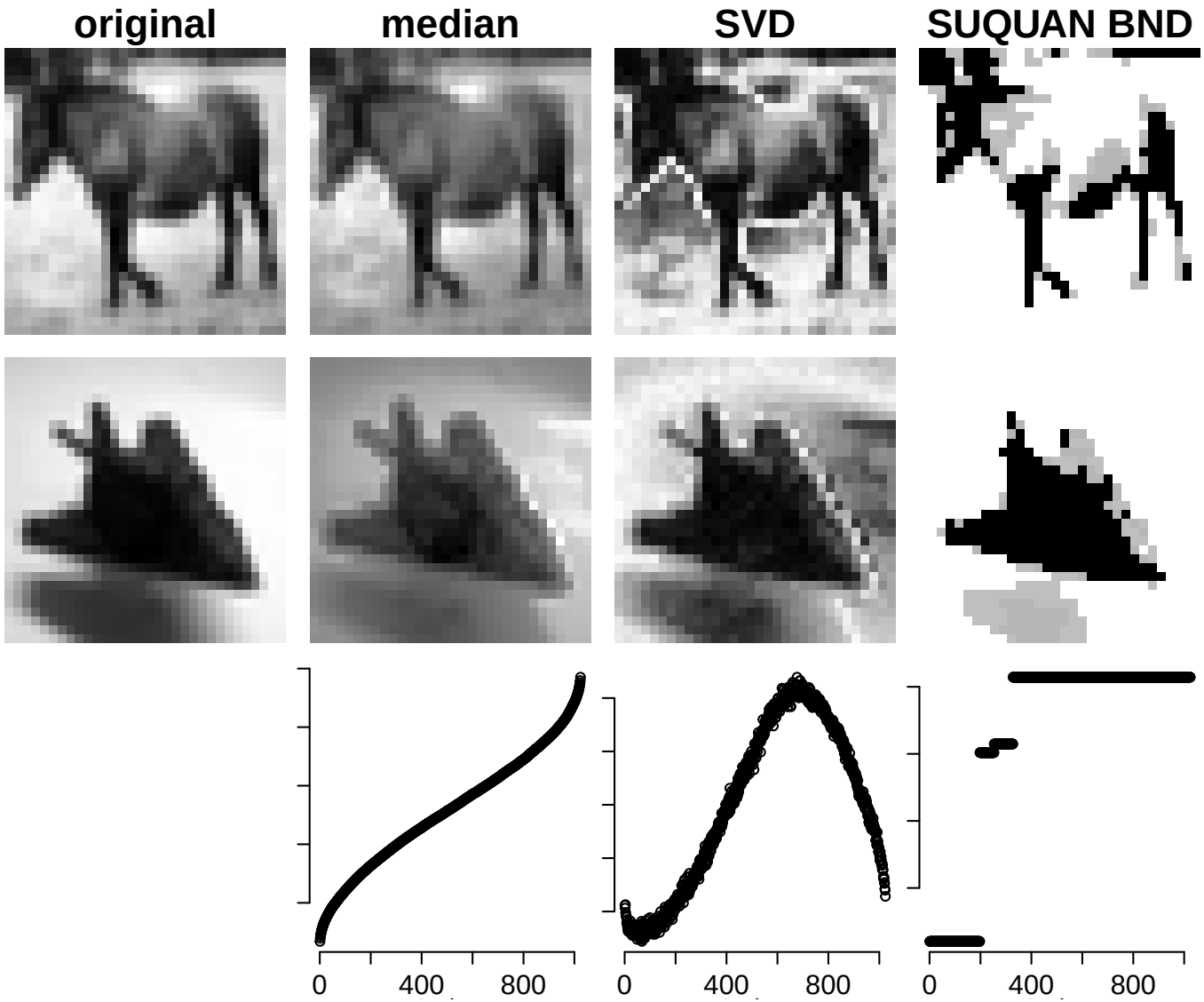
Performance on simulated data.

Suquan: performance on CIFAR-10



Performance on CIFAR-10 data.

Suquan: normalized image



Normalized images in CIFAR-10.

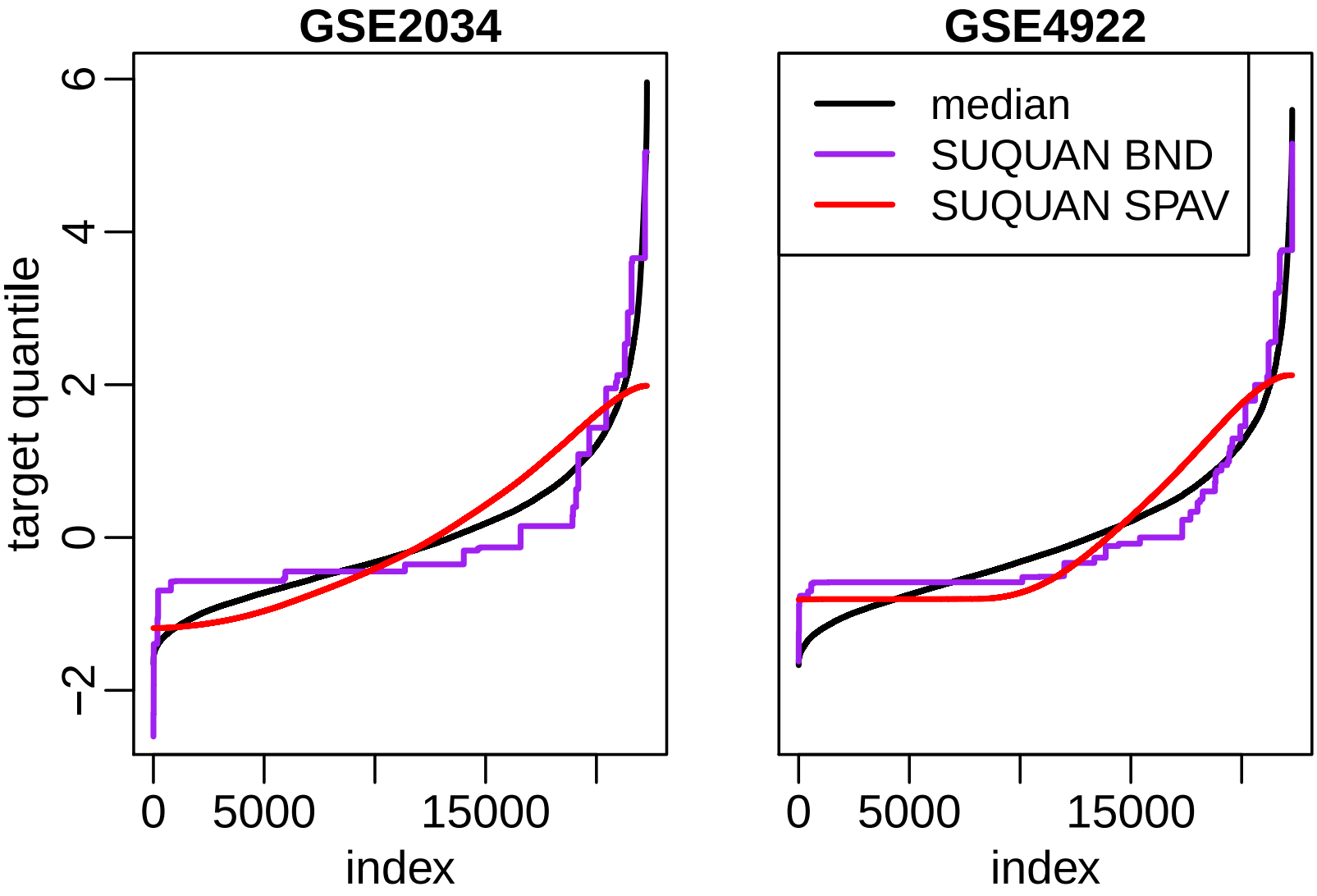
Suquan: performance on gene expression

Table 2 – AUC for SUQUAN and logistic regression with various data normalisation procedures applied to four gene expression datasets.

	LOGISTIC REGRESSION							SUQUAN		
	RAW	RMA	CAUCHY	EXP.	UNIF.	GAUS.	MEDIAN	SVD	BND	SPAV
GSE1456	65.94	68.73	59.56	68.86	68.72	69.00	69.06	57.60	71.44	69.60
GSE2034	74.52	75.42	61.91	74.53	75.22	76.45	74.92	52.61	70.50	76.11
GSE2990	57.01	60.43	54.72	61.25	56.25	58.66	59.72	52.51	59.22	59.94
GSE4922	58.52	58.86	55.24	58.81	55.66	60.01	59.18	52.39	61.82	61.41
AVERAGE	64.00	65.86	57.86	65.86	63.96	66.03	65.72	53.78	65.75	**66.77**

AUC for SUQUAN and logistic regression with various data normalisation applied to gene expression prediction.

Suquan: learned quantiles normalization



target quantiles learned for two gene expression datasets.