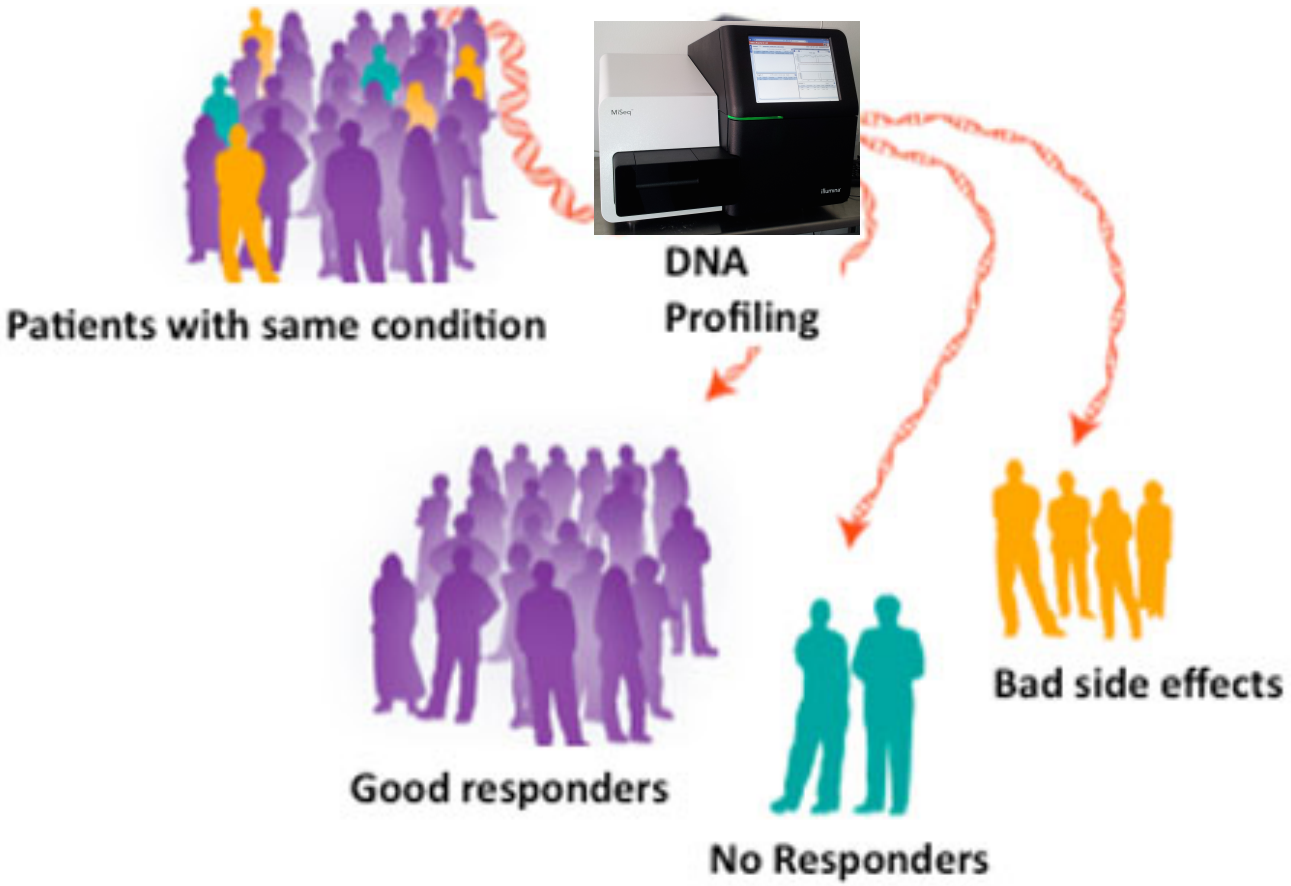
Patient stratification and cancer prognosis from molecular profiles

Jean-Philippe Vert

Memorial Sloan-Kettering Cancer Center, New-York, May 5, 2016

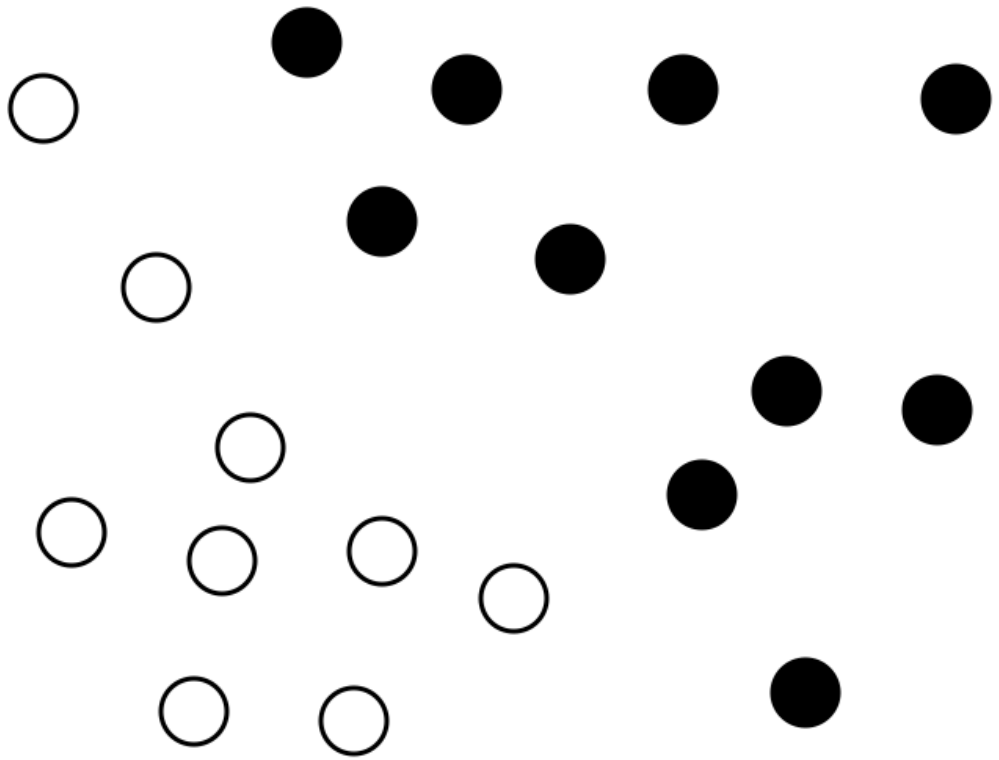
Motivation



- Diagnosis
- Prognosis
- Drug response prediction / personalized treatment optimization

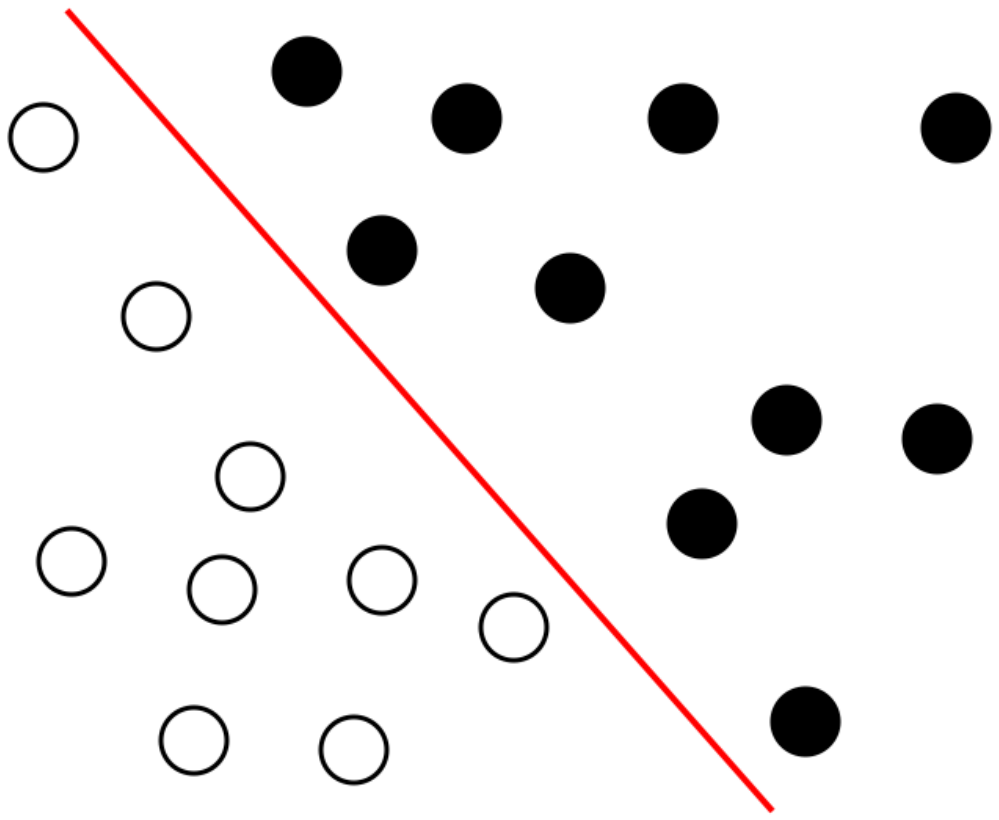
Learning from data (EASY case)

$n(= 19)$ patients $>>$ $p(= 2)$ genes



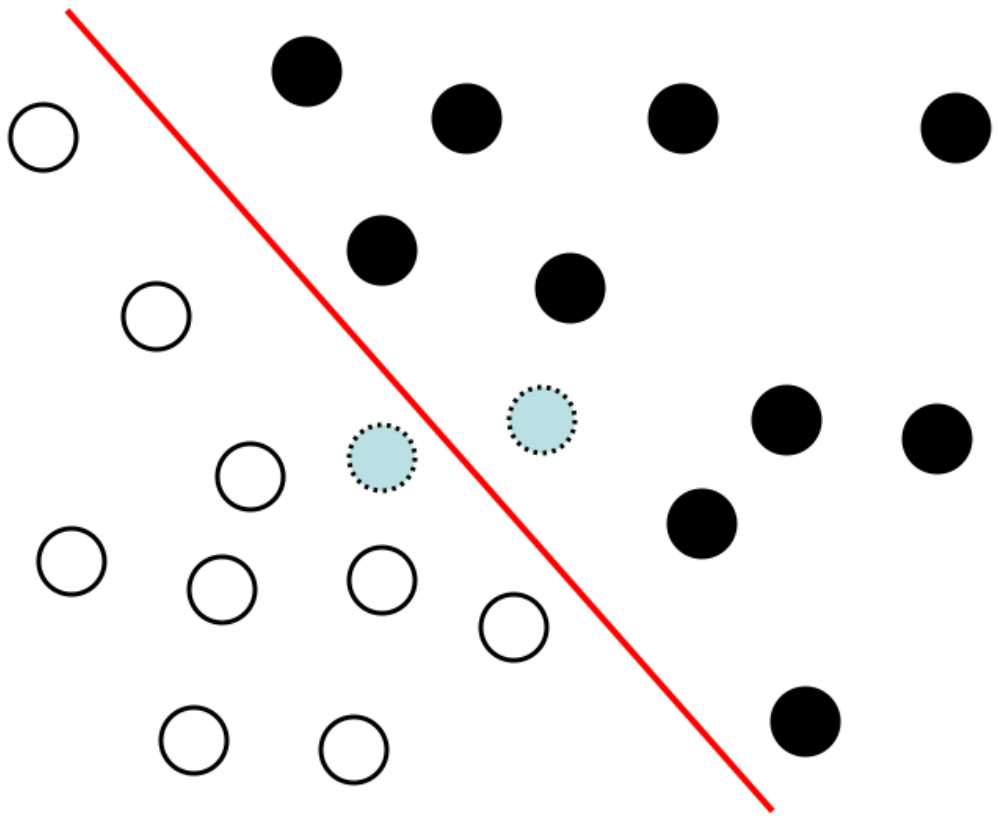
Learning from data (EASY case)

$n(= 19)$ patients $>>$ $p(= 2)$ genes



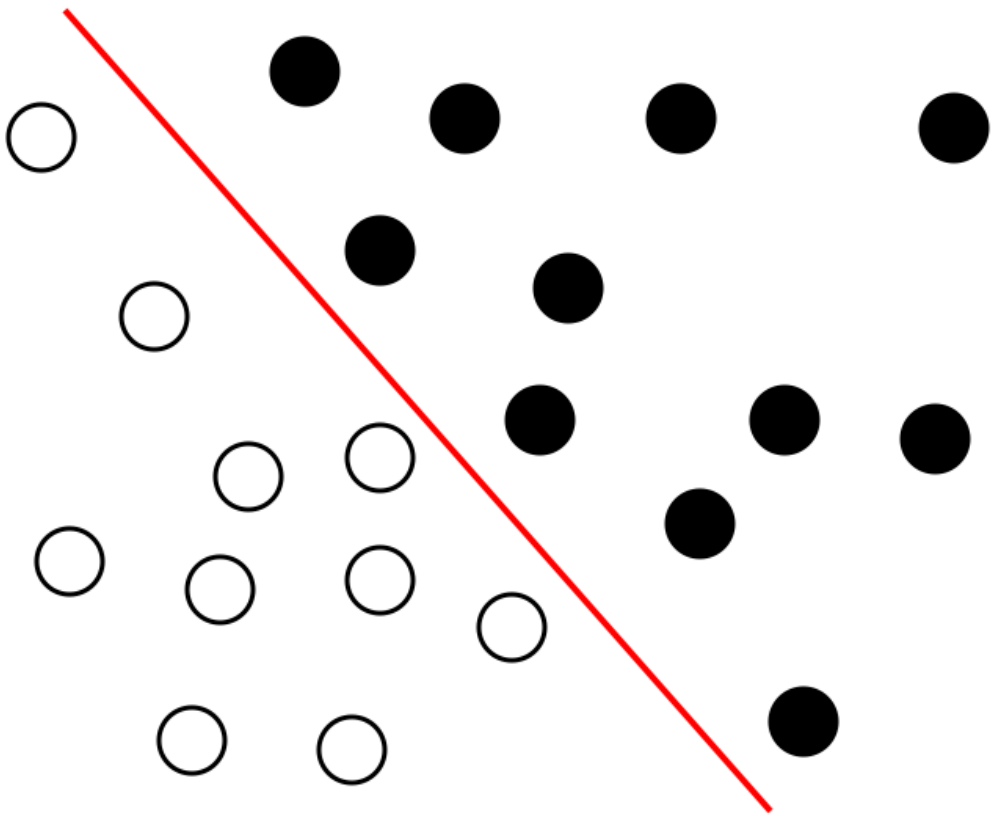
Learning from data (EASY case)

$n(= 19)$ patients $>>$ $p(= 2)$ genes

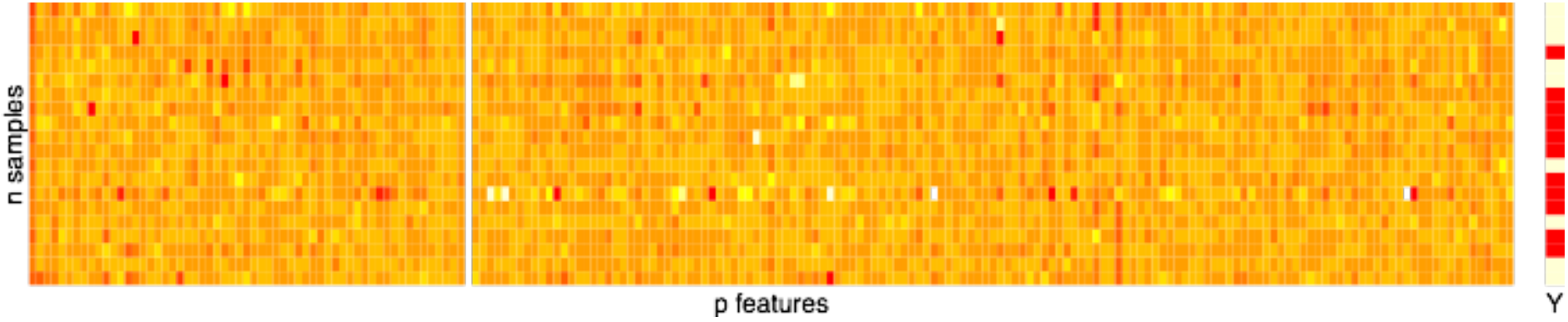


Learning from data (EASY case)

$n(= 19)$ patients $>>$ $p(= 2)$ genes



*-omics challenge: $n << p$



- $n = 10^2 \sim 10^4$ (patients)
- $p = 10^4 \sim 10^7$ (genes, mutations, copy number, ...)
- Data of various nature (continuous, discrete, structured, ...)
- Data of variable quality (technical/batch variations, noise, ...)

Consequences: Accuracy drops, biomarker selection unstable

Can we replace the high-dimensional profile of a sample by a "simpler" representation, more amenable to statistical learning?

Outline

1. Patient stratification from somatic mutations using gene network
2. Supervised quantile normalization
3. Learning from rankings through pairwise comparisons
4. Conclusion

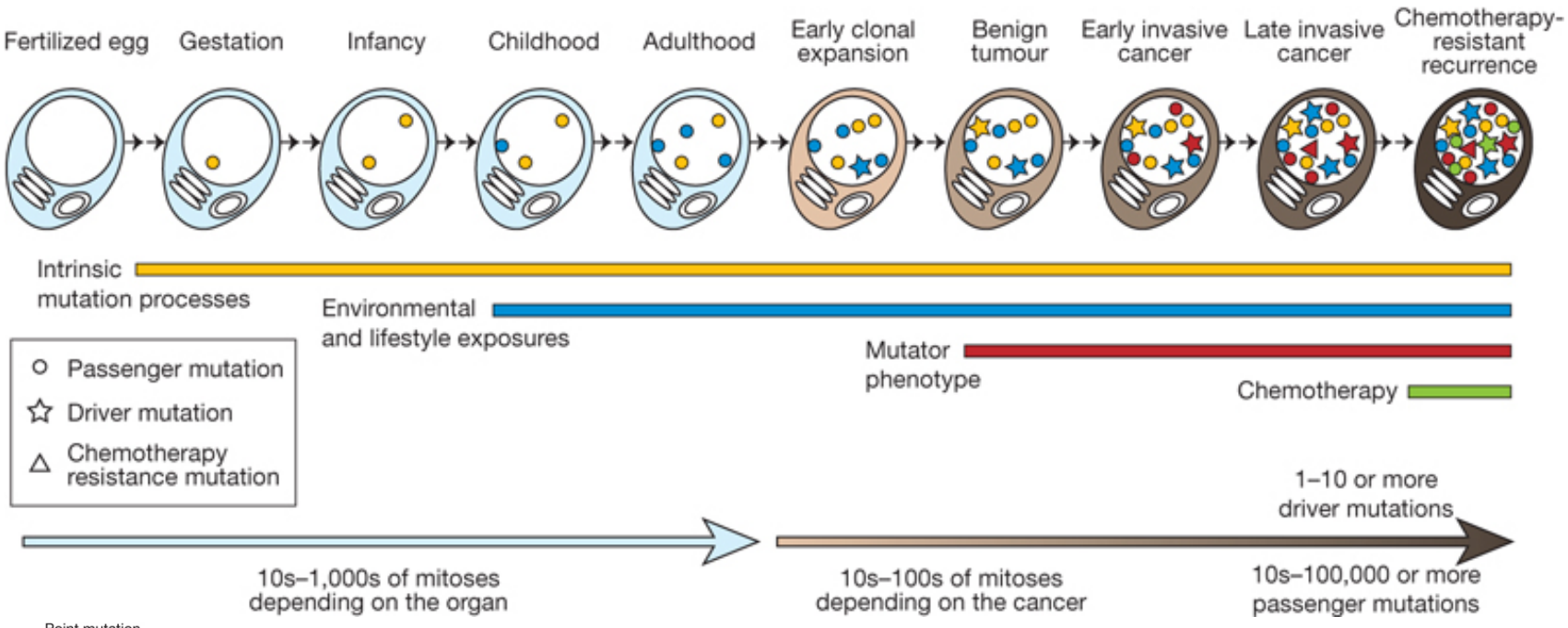
Outline

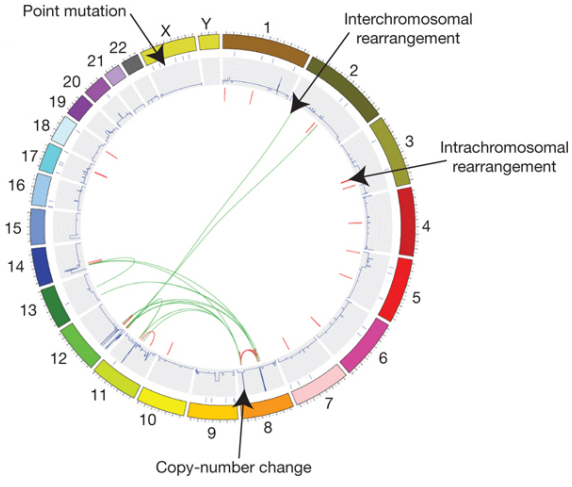
1. Patient stratification from somatic mutations using gene network
2. Supervised quantile normalization
3. Learning from rankings through pairwise comparisons
4. Conclusion

Joint work with

Marine Le Morvan

Somatic mutations in cancer





Stratton et al., Nature 458:719-24, 2009.

Large-scale efforts to collect somatic mutations profiles

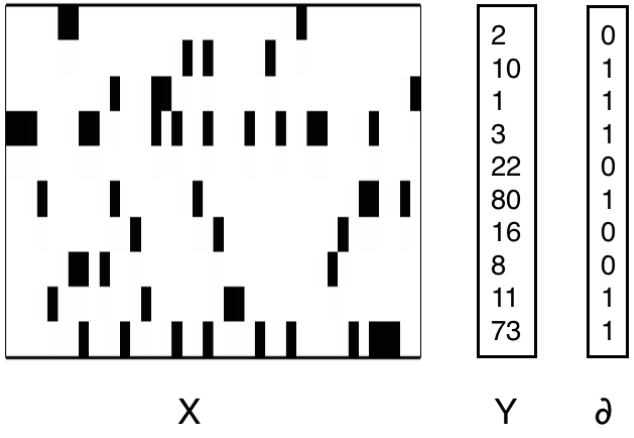
Data used in this study:

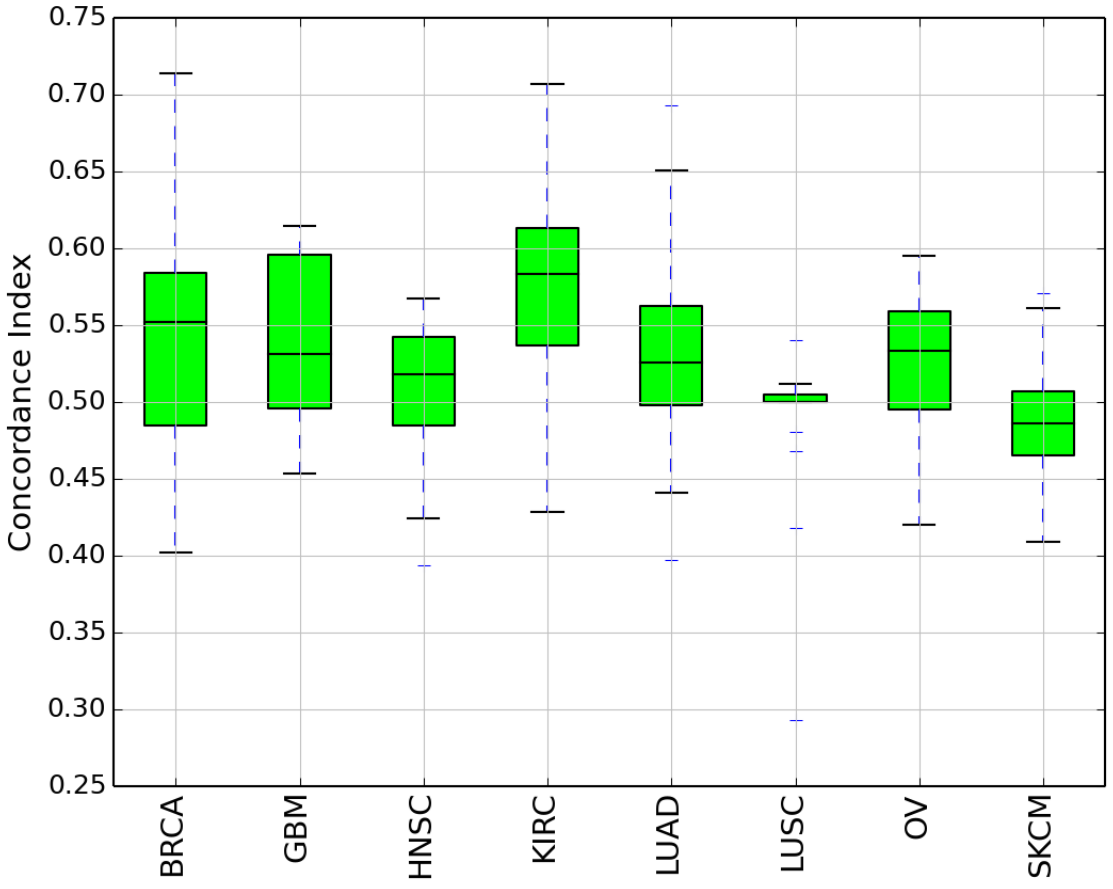
- 3,378 samples with survival information
- from 8 cancer types
- downloaded from the TCGA / cBioPortal portals.

Cancer type	Patients	Genes
LUAD (Lung adenocarcinoma)	430	20 596
SKCM (Skin cutaneous melanoma)	307	17 463
GBM (Glioblastoma multiforme)	265	14 750
BRCA (Breast invasive carcinoma)	945	16 806
KIRC (Kidney renal clear cell carcinoma)	411	10 609
HNSC (Head and Neck squamous cell carcinoma)	388	17 022
LUSC (Lung squamous cell carcinoma)	169	13 590
OV (Ovarian serous cystadenocarcinoma)	363	10 195

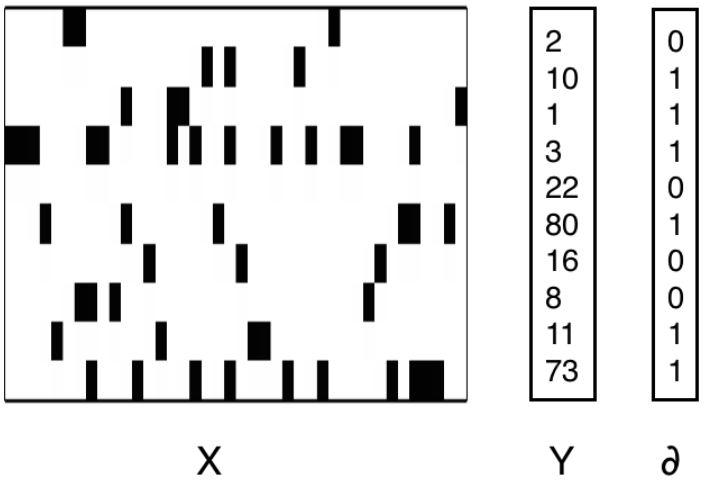
Survival prediction from raw mutation profiles

- Each patient is a binary vector: each gene is mutated (1) or not (2)
- Silent mutations are removed
- Survival model estimated with sparse survival SVM
- Results on 5-fold cross-validation repeated 4 times



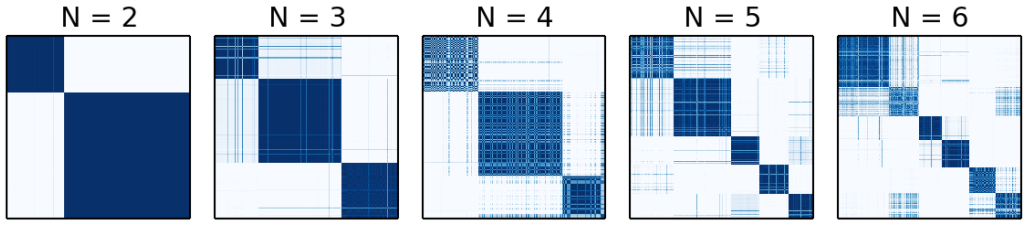


Patient stratification (unsupervised) from raw mutation profiles

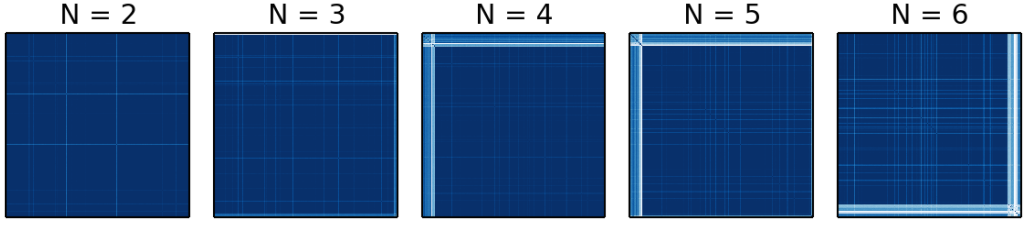


✓ Non-Negative matrix factorisation (NMF)

✓ Desired behaviour:

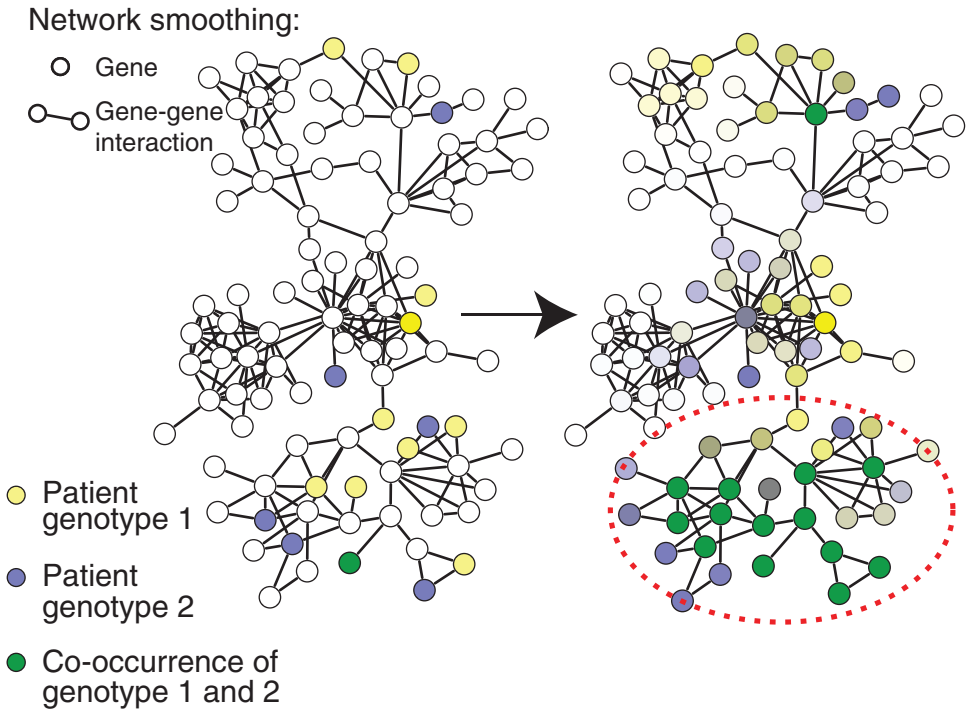


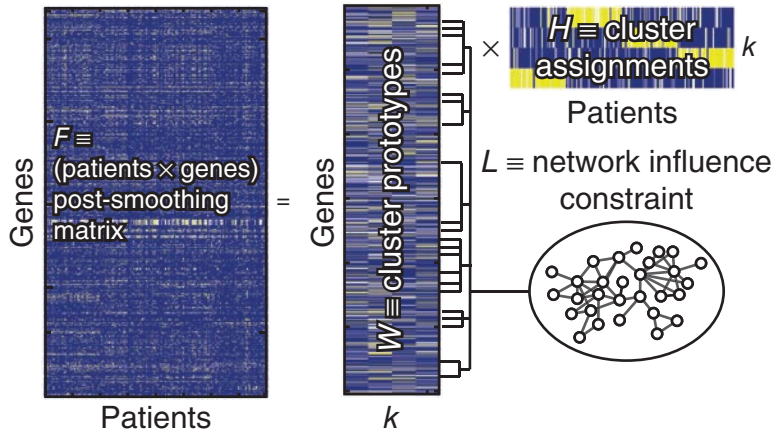
✓ Observed behaviour:

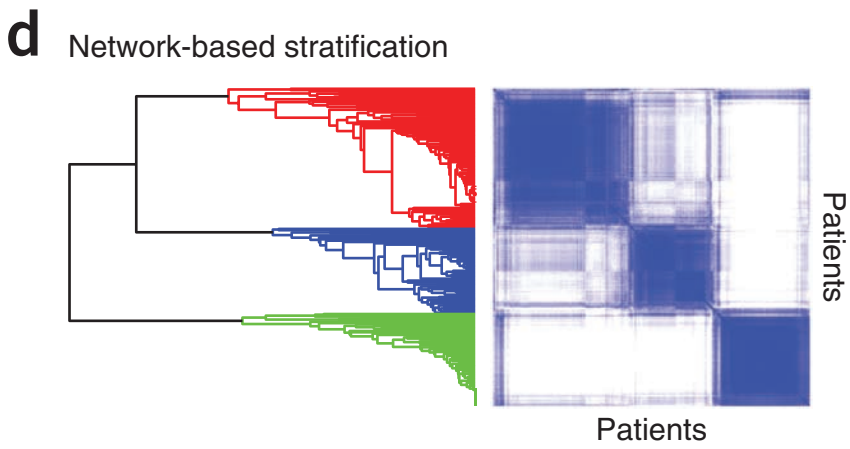


Patients share very few mutated genes!

Network-based stratification (NBS)

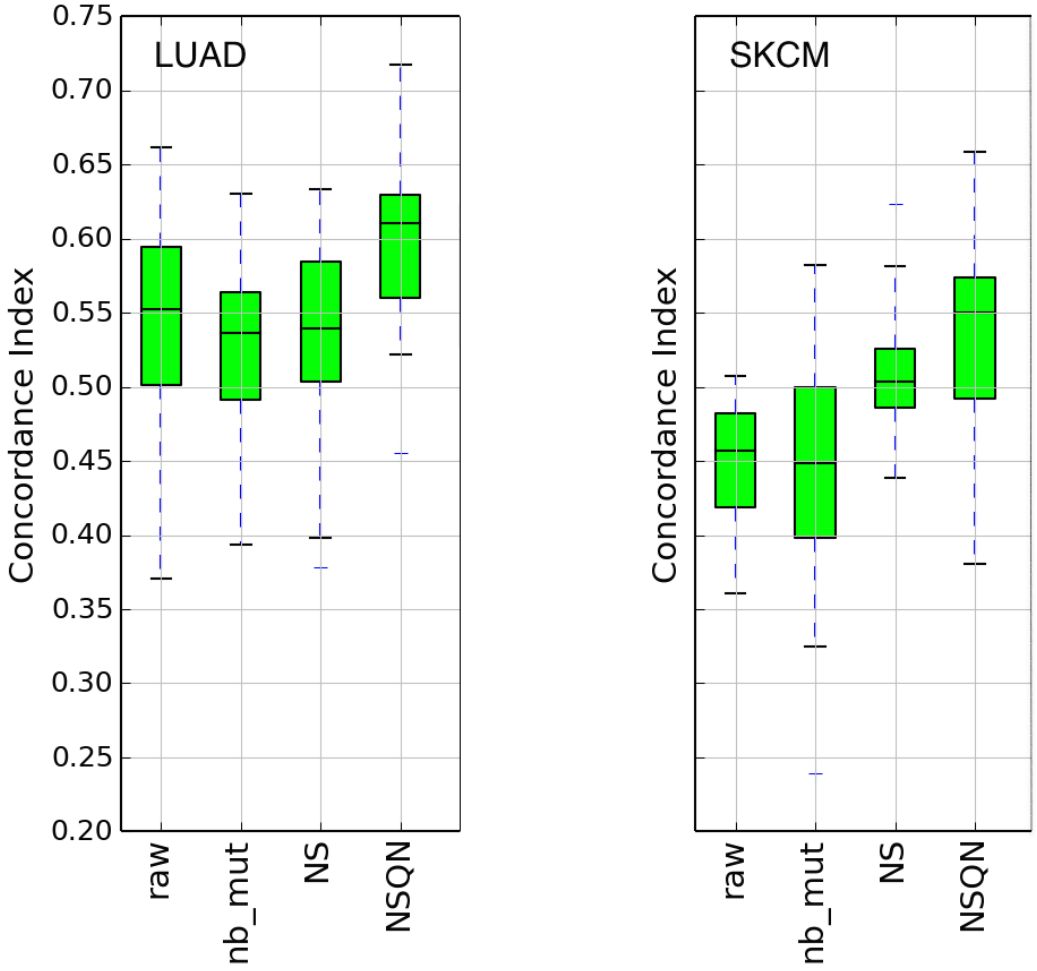






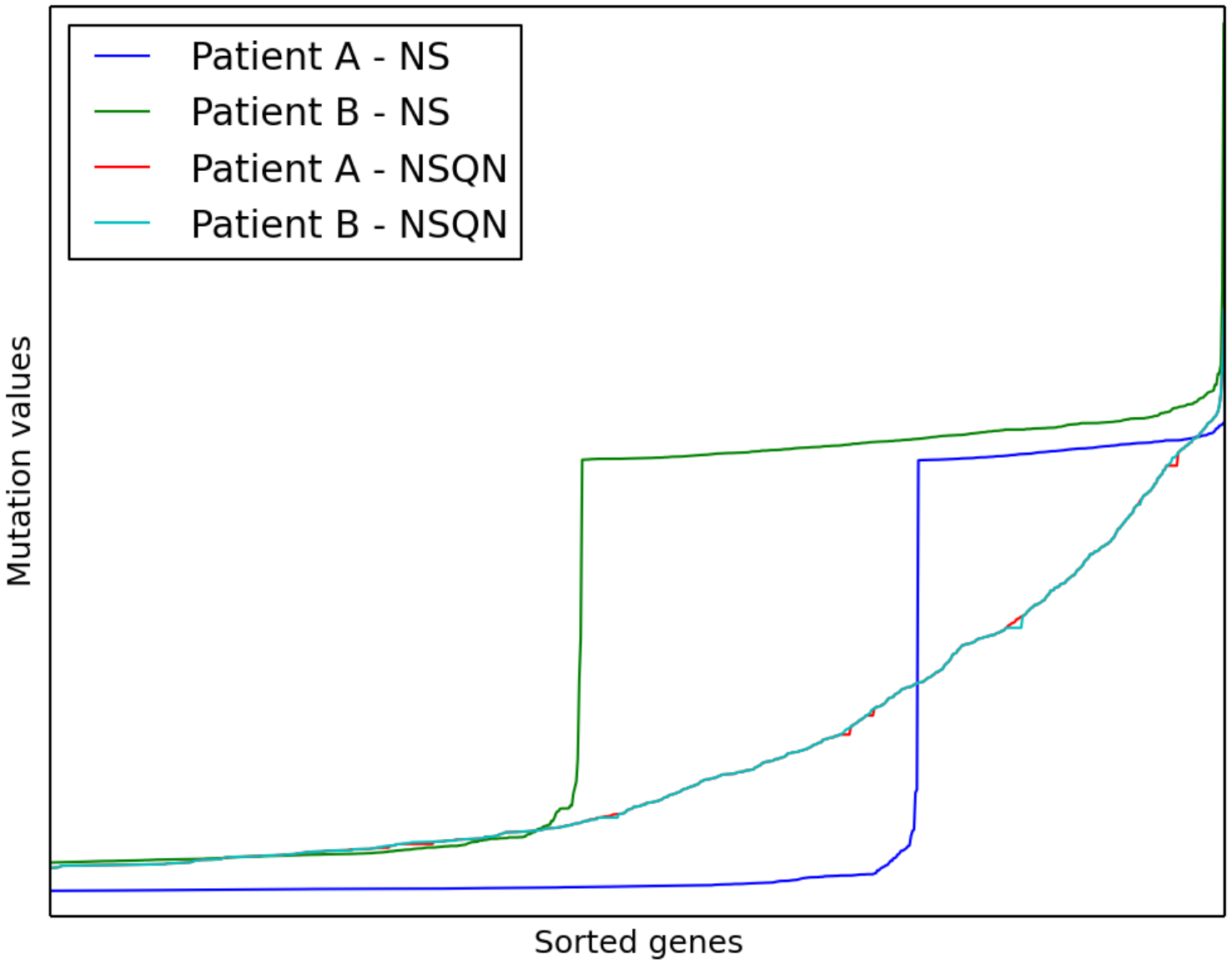
Hofree et al., Nat. Methods, 10:1108-15, 2013.

NBS representation helps to predict survival



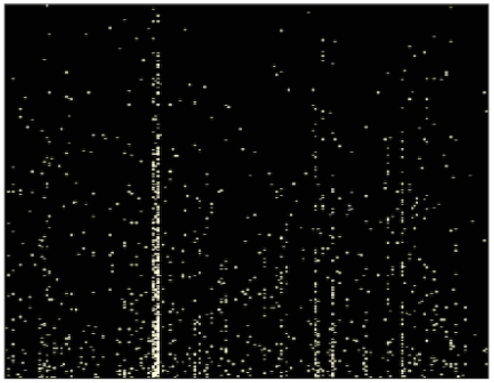
- NS = Network Smoothing
- QN = Quantile normalization
- NBS = NS+QN

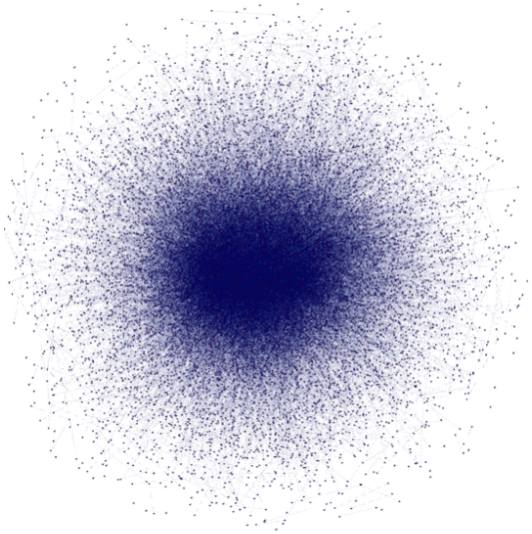
Importance of Quantile Normalization (QN) on NBS



NetNorM: a simplified NSQN

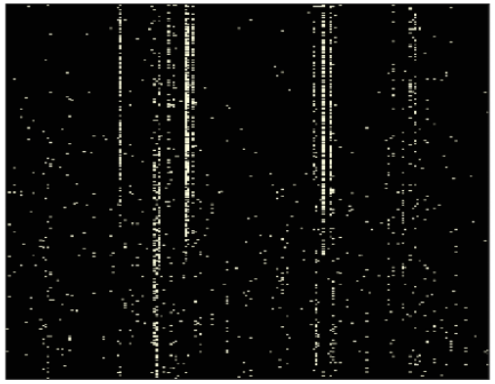
Somatic mutation matrix





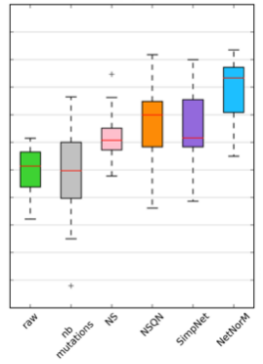
Gene-gene interaction network

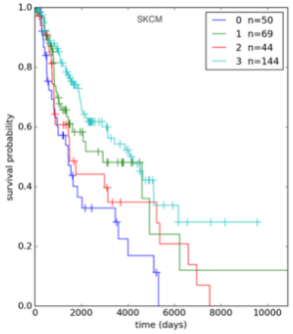
NetNorM



New representation of the mutation matrix

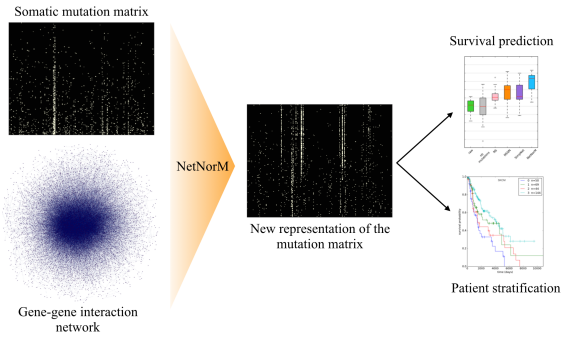
Survival prediction





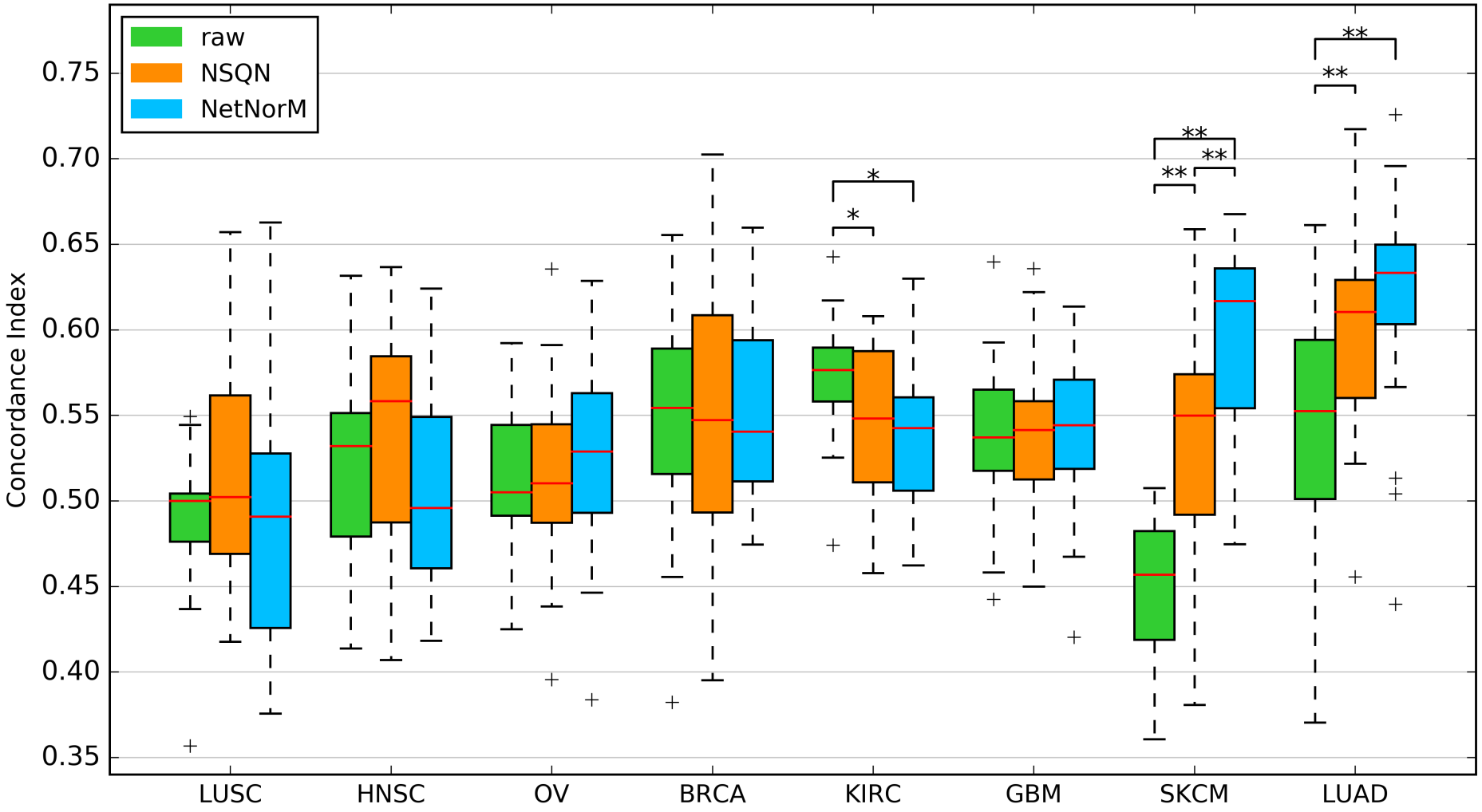
Patient stratification

NetNorM: a simplified NSQN



- Transforms a binary vector of mutation into another binary vector, with a fixed number k of mutations.
- Given a mutation profile $x \in \{0,1\}^p$ with m mutations:
  - If $m < k$, add $k - m$ "proxy" mutations: the ones with the largest number of mutated neighbors
  - If $m > k$, remove $m - k$ "unimportant" mutation: the ones with the smallest degree in the gene network
- k is the only parameter, chosen by heuristics or optimized by cross-validation

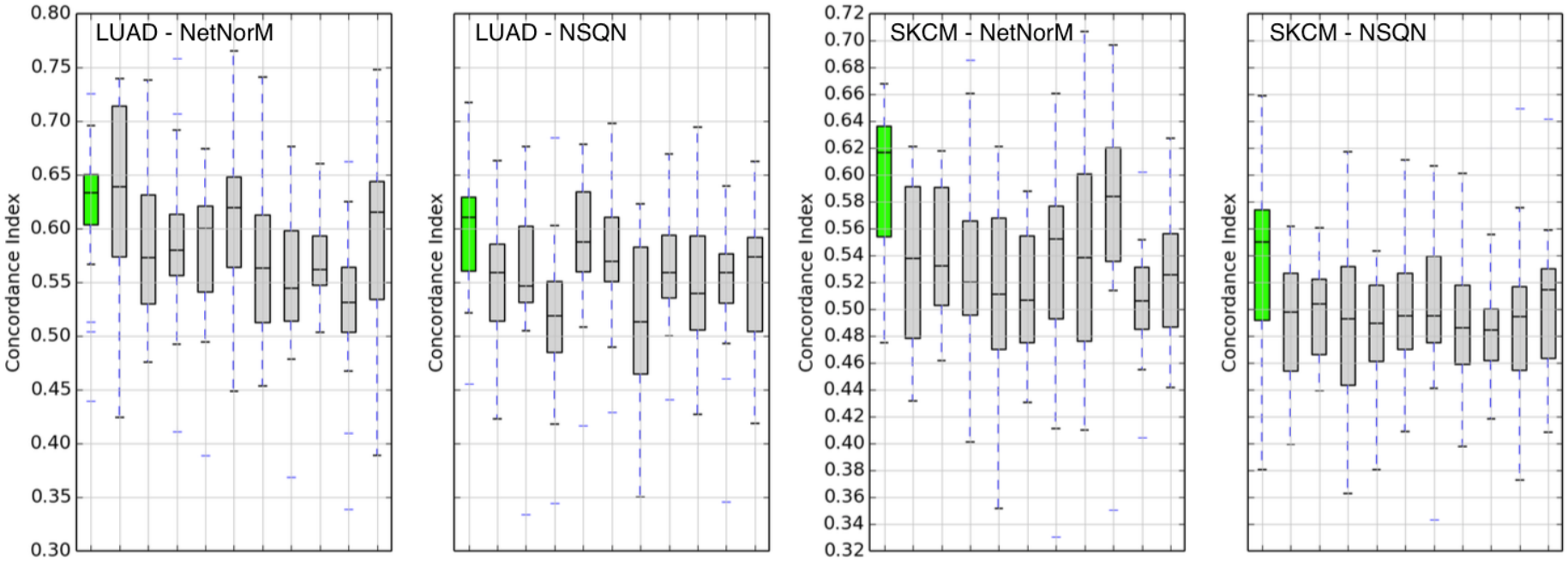
Impact of NetNorM on survival prediction



Use Pathway Commons as gene network.

NSQN and NetNorM benefit from biological information in Pathway Commons

Comparison with 20 randomly permuted networks:



P-values (Welch *t*-test):

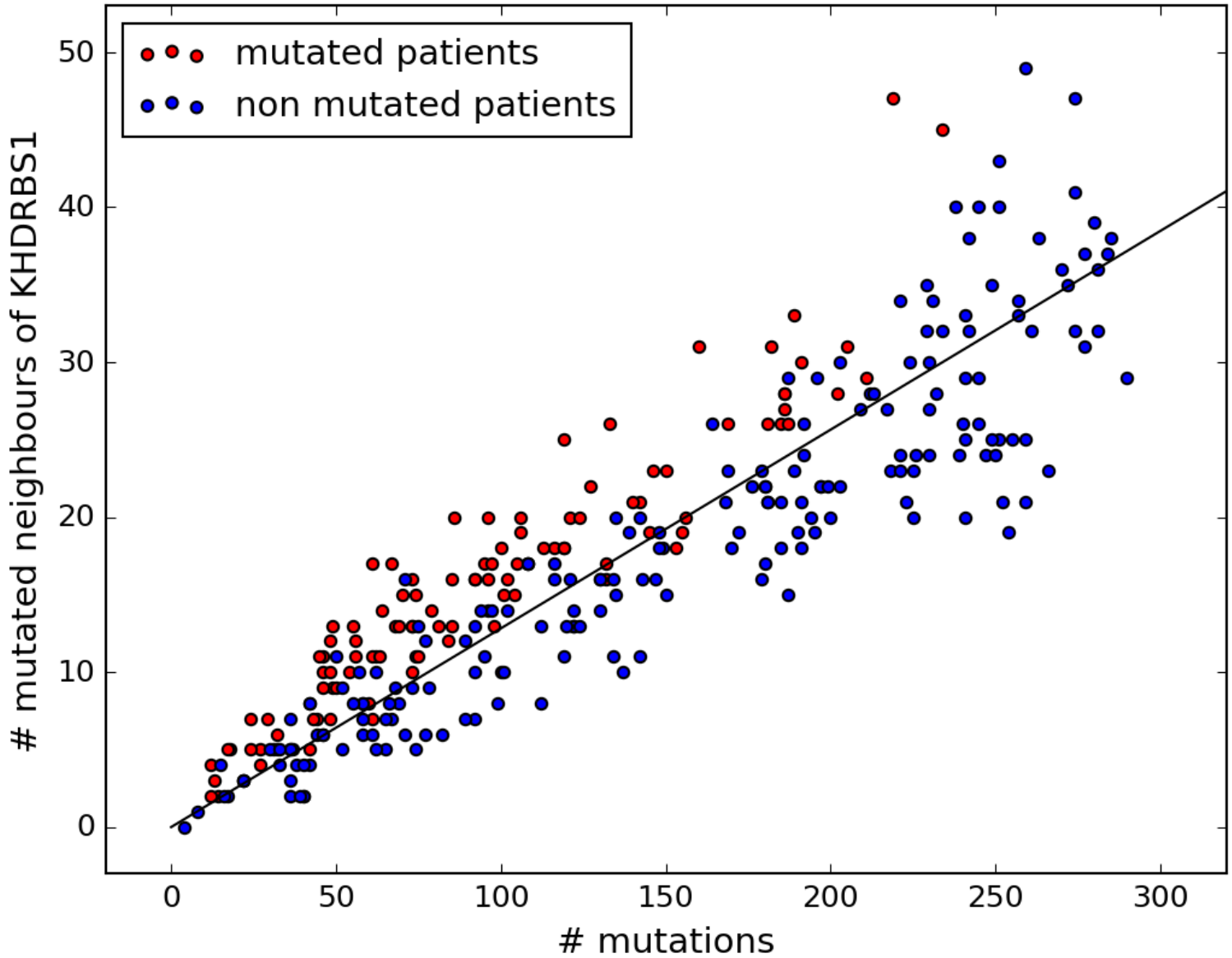
	NSQN	NetNorM
LUAD	2×10^{-3}	3.5×10^{-2}
SKCM	1.2×10^{-2}	1×10^{-4}

Genes frequently selected for survival prediction in LUAD

	freq	coef	m_{all}		$m_{<k_{med}}$		$m_{\geq k_{med}}$		Log-rank test (p-value)		Welsh t-test (p-value)	
			raw	NetNorM	raw	NetNorM	raw	NetNorM	raw	NetNorM	raw	NetNorM
TP53	19	-0.16	238	274	123	159	115	115	7.6×10^{-2}	9.4×10^{-2}	5.2×10^{-22}	1.2×10^{-13}
CRB1	18	-0.4	44	38	22	22	22	16	1.6×10^{-4}	1.4×10^{-6}	9.9×10^{-4}	6.9×10^{-2}
NOTCH4	17	-0.23	42	26	14	14	28	12	9.3×10^{-1}	3.3×10^{-2}	1.9×10^{-6}	2.6×10^{-1}
ANK2	17	0.1	90	90	33	33	57	57	1.2×10^{-2}	1.2×10^{-2}	6.3×10^{-10}	6.3×10^{-10}
RPS9	16	0.38	0	106	0	106	0	0	-	1.8×10^{-1}	-	4.2×10^{-47}
LAMA2	15	0.16	52	38	14	15	38	23	1.5×10^{-2}	2.3×10^{-2}	6.3×10^{-9}	2.6×10^{-3}
RYR2	14	0.07	165	161	70	70	95	91	1.4×10^{-2}	2.1×10^{-2}	6.7×10^{-19}	1×10^{-15}
IGF2BP2	14	-0.15	6	67	2	63	4	4	1.4×10^{-5}	3.6×10^{-3}	1×10^{-1}	6.8×10^{-7}
SMARCA5	14	-0.09	5	137	1	133	4	4	2.1×10^{-1}	5.3×10^{-3}	1.3×10^{-1}	1×10^{-27}
KHDRBS1	13	0.11	7	117	2	112	5	5	7.1×10^{-1}	9.7×10^{-1}	6.5×10^{-2}	1.3×10^{-18}
YWHAZ	13	-0.18	2	241	0	239	2	2	2.5×10^{-31}	6.1×10^{-4}	4.7×10^{-1}	4.4×10^{-37}
HRNR	13	-0.12	62	64	20	22	42	42	1.1×10^{-1}	1.1×10^{-1}	6×10^{-10}	2.9×10^{-9}
CSNK2A2	11	0.06	2	129	1	128	1	1	9×10^{-1}	8.8×10^{-1}	5.9×10^{-1}	4.2×10^{-27}
MED12L	11	0.04	27	27	8	8	19	19	5.5×10^{-2}	5.5×10^{-2}	1.7×10^{-4}	1.7×10^{-4}

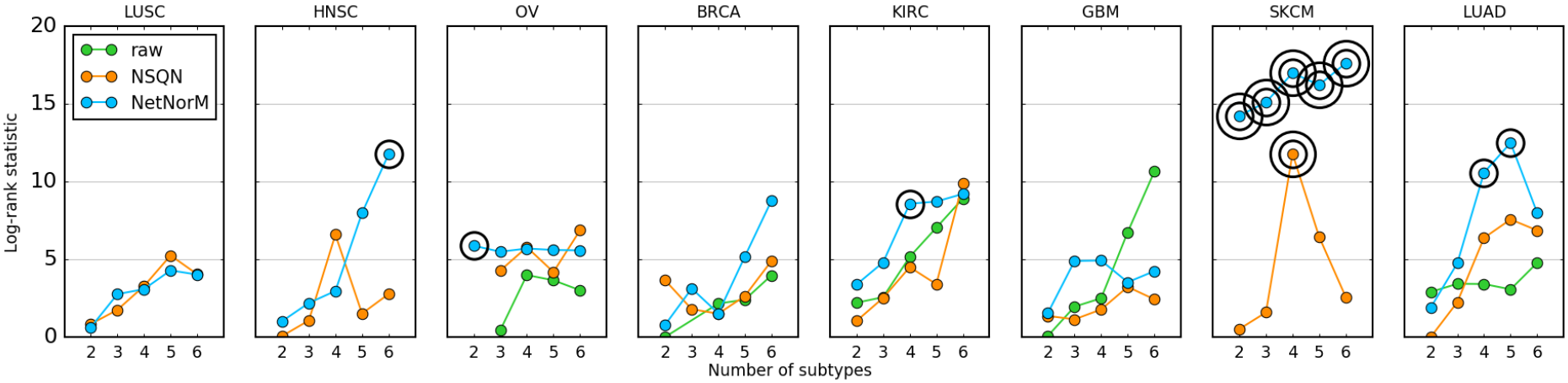
- 14 genes are selected at least 50% of the time
- 6/14 are "proxy" genes (in blue)
  - big hubs in the network
  - get mutated by NetNorm in patients with few mutations $\Longrightarrow$ they encode the mutation rate
- 8/14 are "normal" prognostic genes

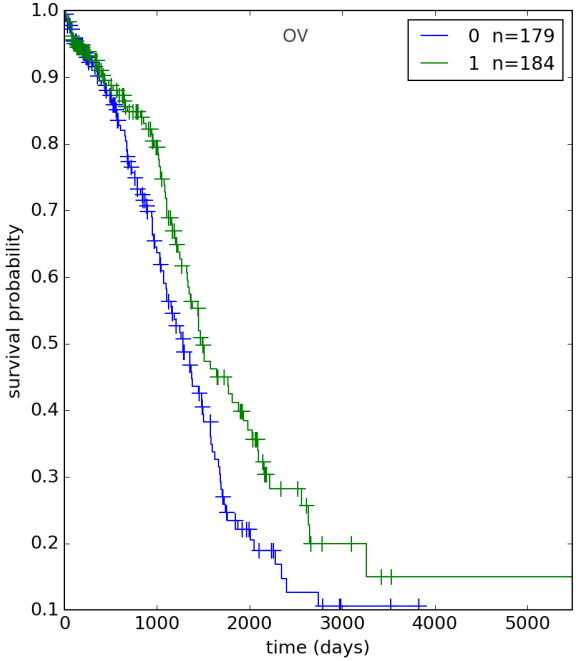
Proxy mutations encode also local mutational burden

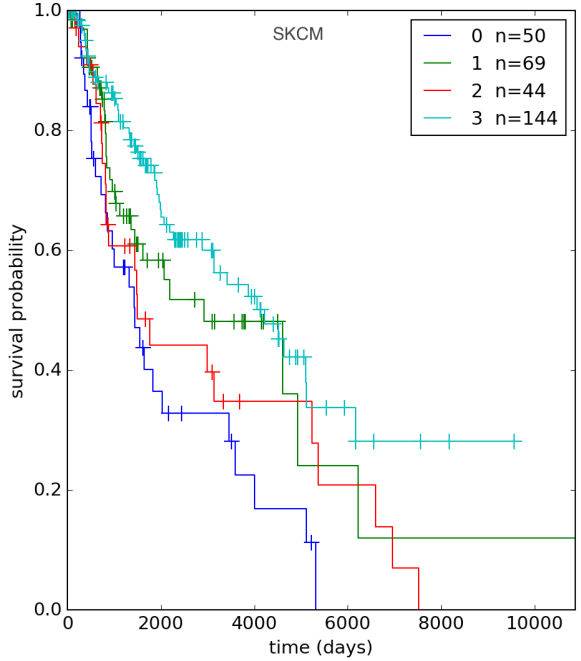


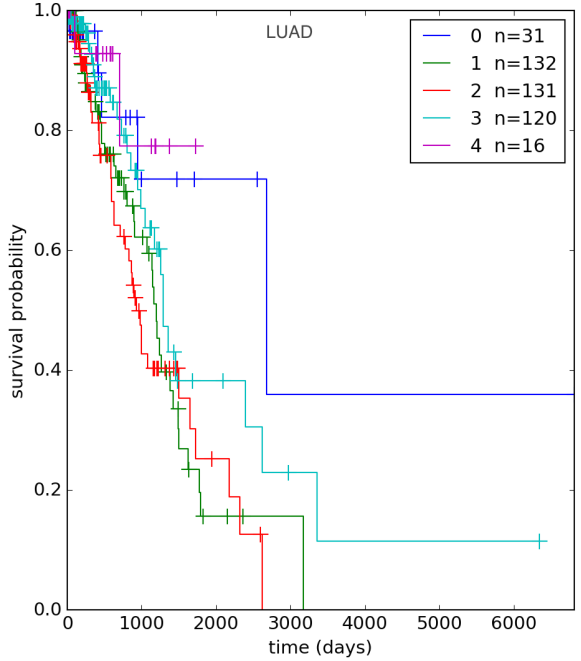
KHDRBS1: a member of the K homology domain-containing, RNA-binding, signal transduction-associated protein family

Unsupervised patient stratification









Summary

- Somatic profiles are challenging because
  - Little overlap between patients
  - Large variability in number of mutations
- Network smoothing / local averaging sometimes helps
  - but with current methods, looking at the direct neighbors is good enough
- Normalizing for total number of mutations is at least as important
  - through QN or NetNorm, for example
  - this is not for biological reasons, but for mathematical reasons
  - probably room for improvement

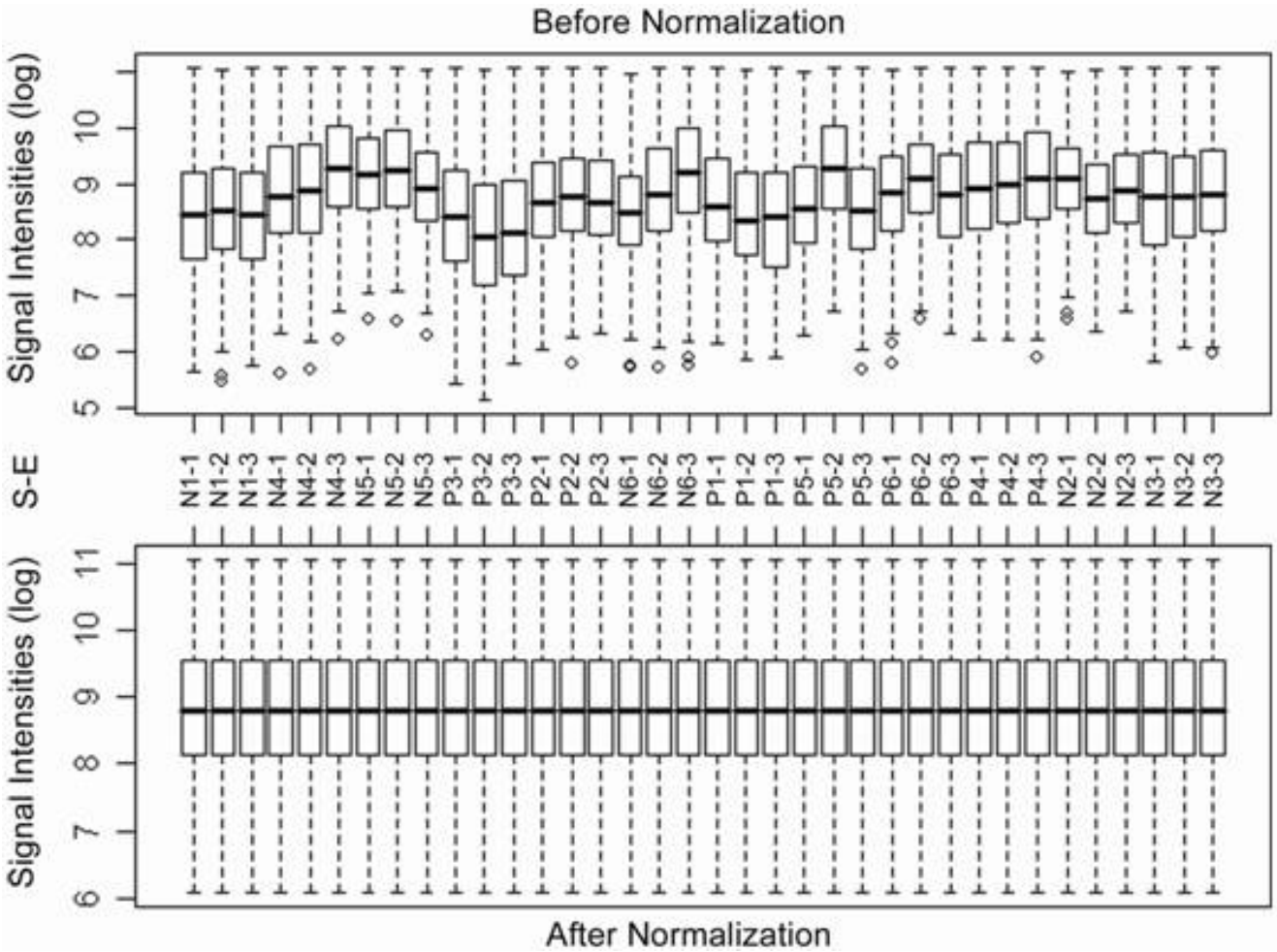
Outline

1. Patient stratification from somatic mutations using gene network
2. Supervised quantile normalization
3. Learning from rankings through pairwise comparisons
4. Conclusion

Joint work with

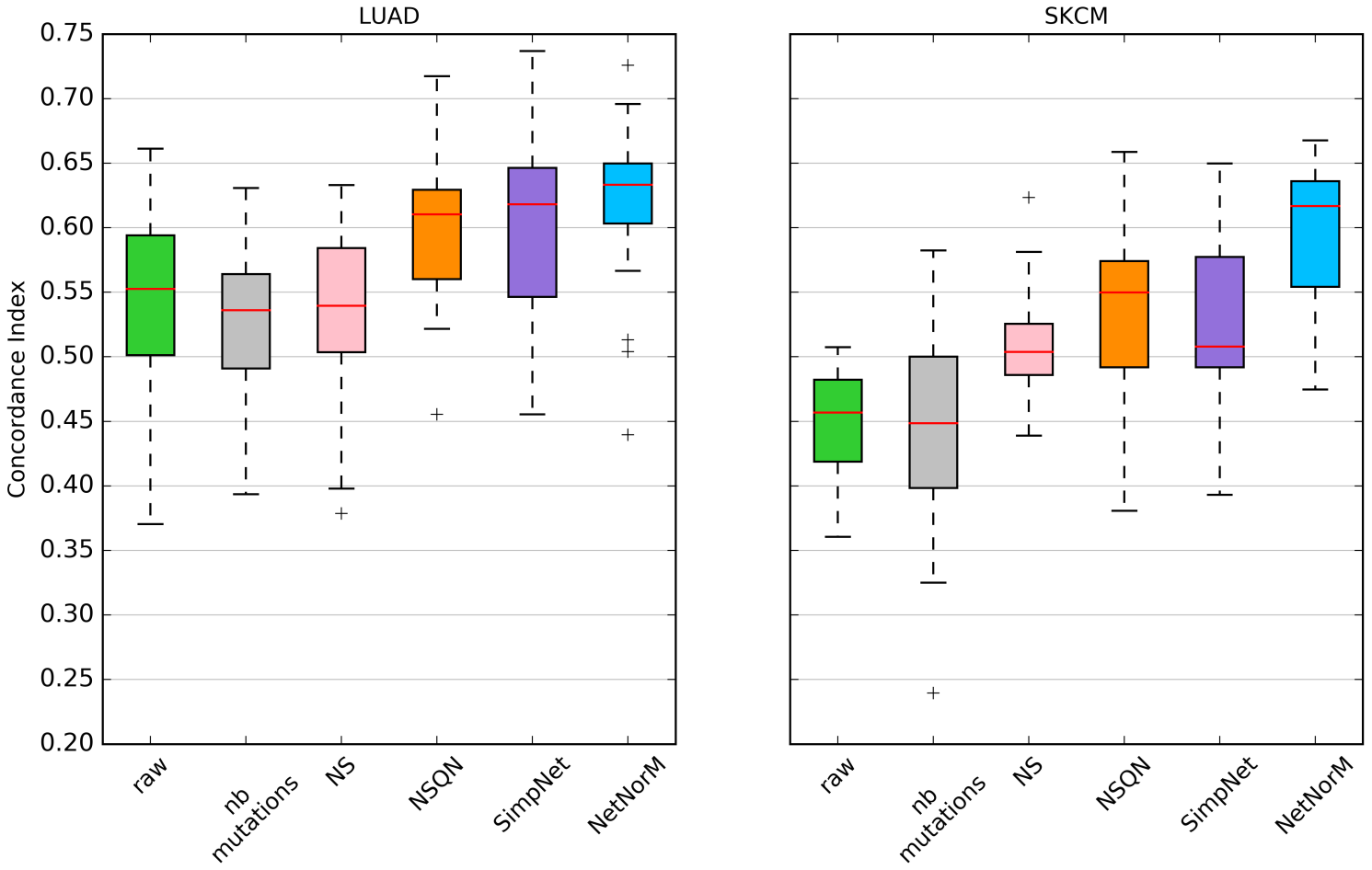
Marine Le Morvan

Standard full quantile normalization



Useful we believe the "true" signal should have the same distribution but is perturbed by "noise" (e.g., batch effect)

QN for mutations



- The difference in distribution is not due to noise
- However QN helps, and impacts the performance
- How to choose the "best" target distribution?

Learning the target distribution

- $x_1, \ldots, x_n$ a set of p-dimensional samples
- $f \in \mathbb{R}^p$ a non-decreasing target distribution (CDF)
- For $x \in \mathbb{R}^p$, let $\Phi_f(x) \in \mathbb{R}^p$ be the data after QN with target distribution f
- Standard approaches (NSQN, NetNorM, ...)
  1. Fix f arbitrarily
  2. QN all samples to get $\Phi_f(x_1), \ldots, \Phi_f(x_n)$
  3. Learn a generalized linear model (w, b) on normalized data:

$$\min_{w,b} \frac{1}{n} \sum_{i=1}^{n} \ell_i \left(w^\top \Phi_f(x_i) + b \right) + \lambda \Omega(w)$$

- SUQUAN: jointly learn f and (w, b):

$$\min_{w,b,f} \frac{1}{n} \sum_{i=1}^{n} \ell_i \left(w^\top \Phi_f(x_i) + b \right) + \lambda \Omega(w)$$

SUQAN: supervised quantile normalization

- For $x \in \mathbb{R}^p$, let $\Pi_x \in \mathbb{R}^{p \times p}$ the permutation matrix of x's entries

$$x = \begin{pmatrix} 4.5 \\ 1.2 \\ 10.1 \\ 8.9 \end{pmatrix} \quad \Pi_x = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad f = \begin{pmatrix} 0 \\ 1 \\ 3 \\ 4 \end{pmatrix}$$

- Quantile normalized x with target distribution f is:

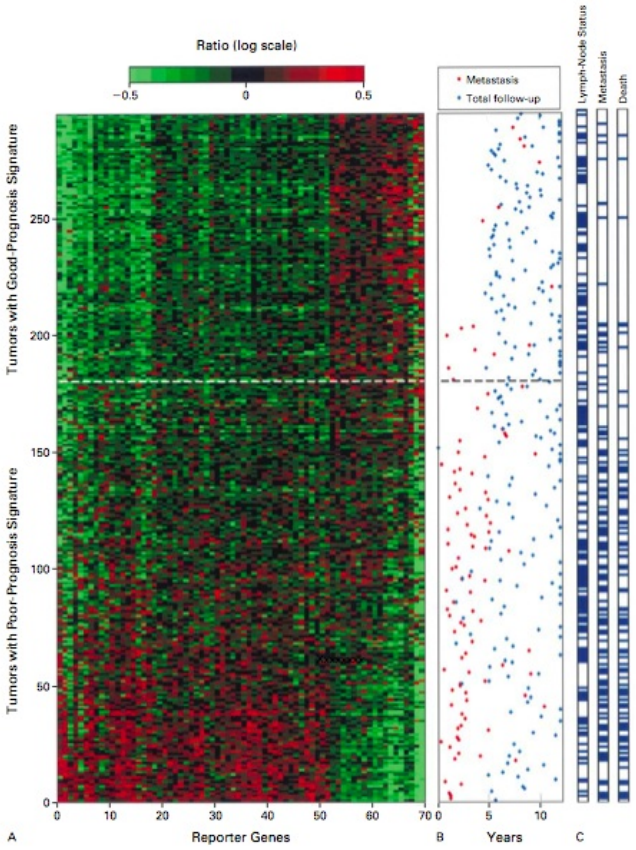
$$\Phi_f(x) = \Pi_x f$$

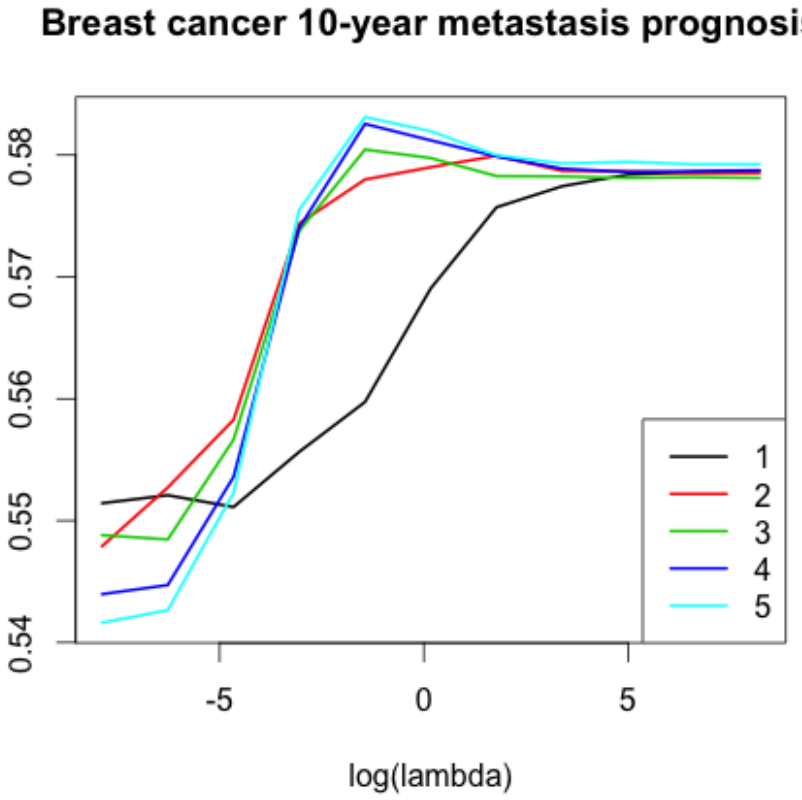
- SUQUAN solves

$$\begin{aligned} &\min_{w,b,f} \frac{1}{n} \sum_{i=1}^{n} \ell\left(w^\top \Pi_{x_i} f + b\right) + \lambda \Omega(w) \\ &= \min_{w,b,f} \frac{1}{n} \sum_{i=1}^{n} \ell\left(< w f^\top, \Pi_{x_i} > + b\right) + \lambda \Omega(w) \end{aligned} \tag{1}$$

- A particular rank-1 matrix optimization, x is replaced by Π_x
- Solved by alternatively optimizing f (isotonic GLM) and w

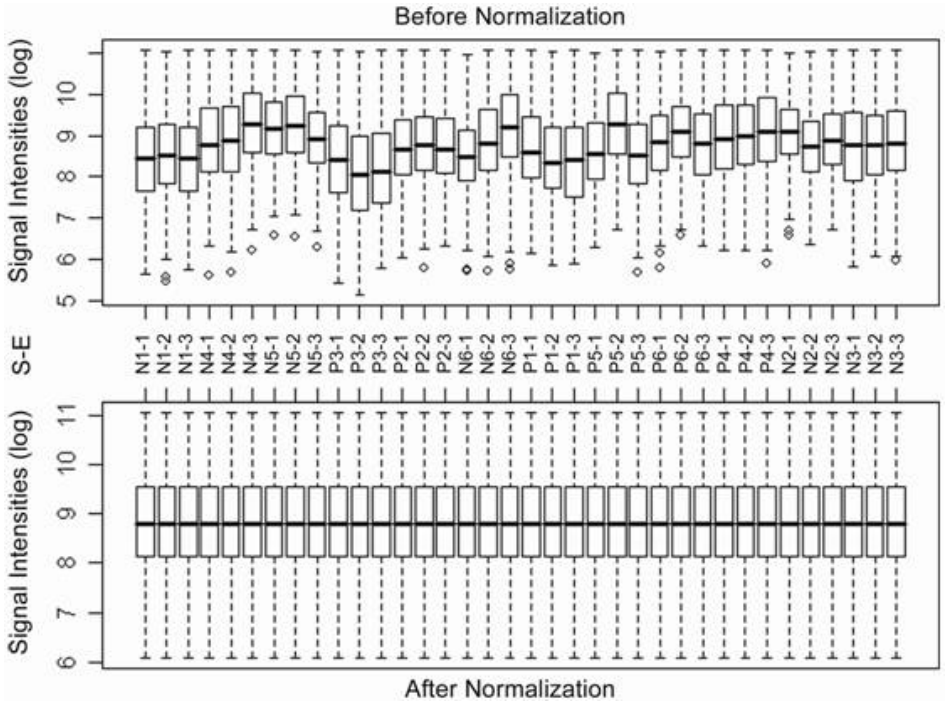
Results (preliminary)





- Breast cancer prognosis from gene expression data (survival logistic regression), TRANSBIG, $n = 198$
- Performance after $1, 2, \ldots, 5$ iterations of alternative optimization of f and (w, b)

SUQUAN summary



- The target distribution of QN can be seen as a parameter to optimize.
- SUQUAN boils down to
  - Represent each sample x by the permutation matrix Π_x that represents the ranking of its features
  - Learn a linear model over these matrices, with a rank-1 matrix of weights

Outline

1. Patient stratification from somatic mutations using gene network
2. Supervised quantile normalization
3. Learning from rankings through pairwise comparisons
4. Conclusion

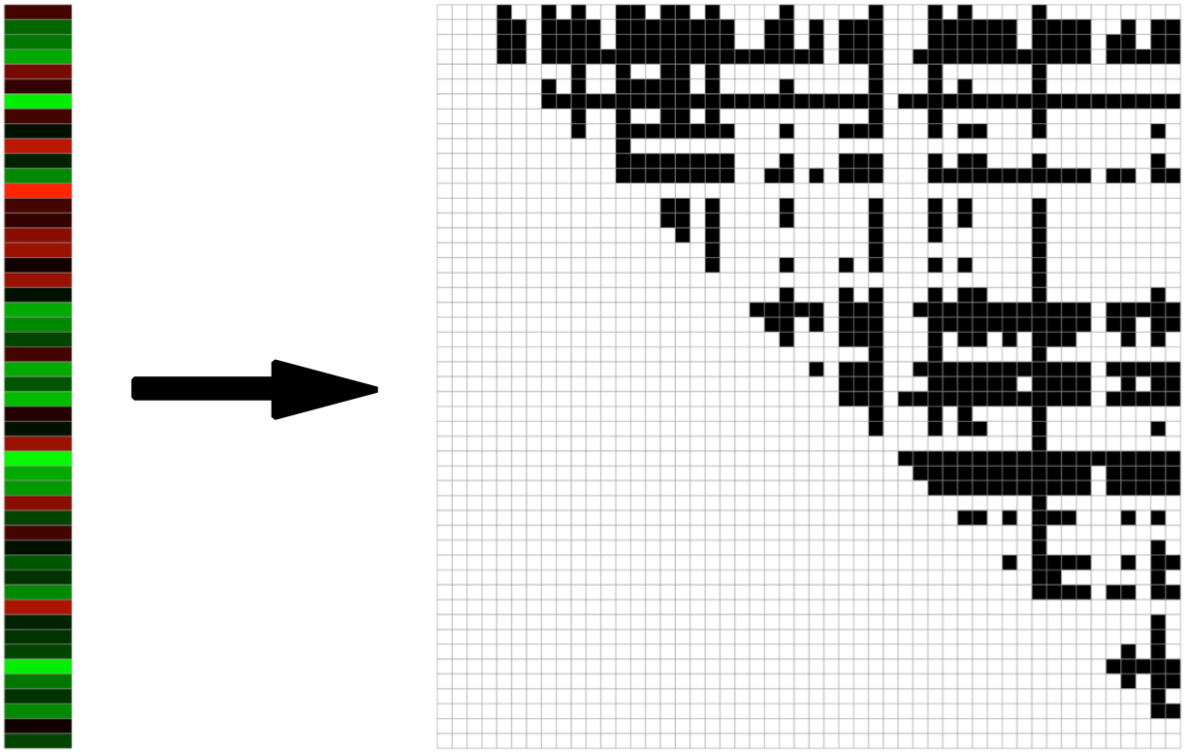
Joint work with

Yunlong Jiao

An idea: all pairwise comparisons

Replace $x \in \mathbb{R}^p$ by $\Phi(x) \in \{0,1\}^{p(p-1)/2}$:

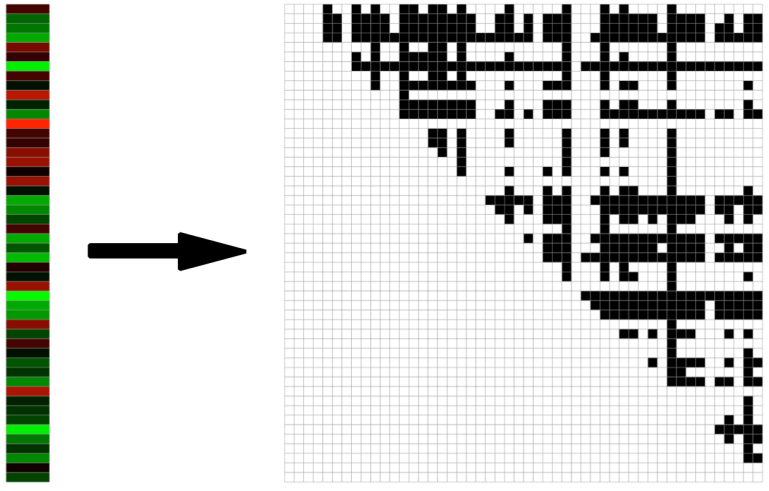
$$\Phi_{i,j}(x) = \begin{cases} 1 & \text{if } x_i \leq x_j\,, \\ 0 & \text{otherwise.} \end{cases}$$



One sample x
p features

Mapping f(x)
p(p-1)/2 bits

Remark: representation of the symmetric group

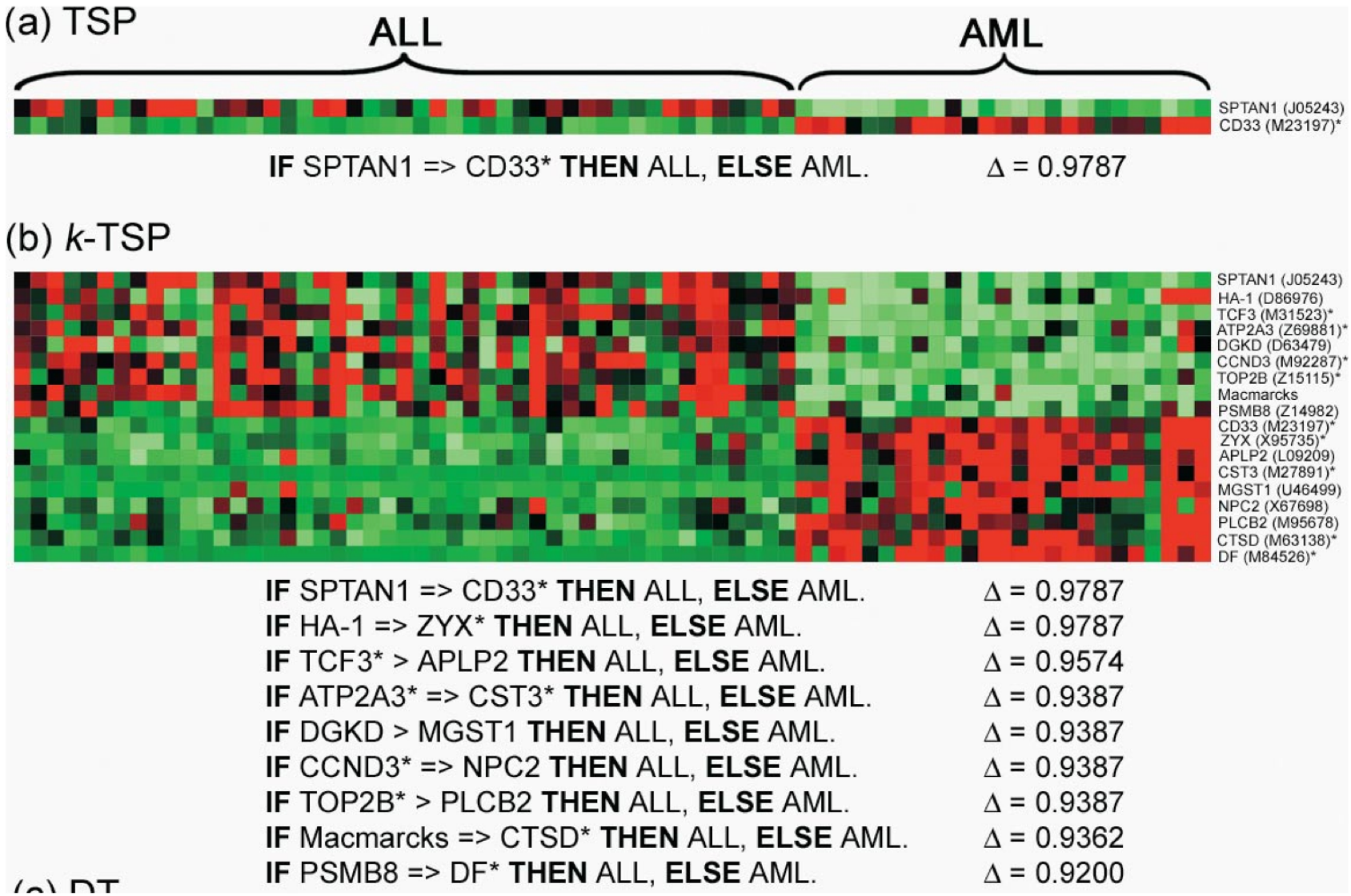


One sample x
p features

Mapping f(x)
p(p-1)/2 bits

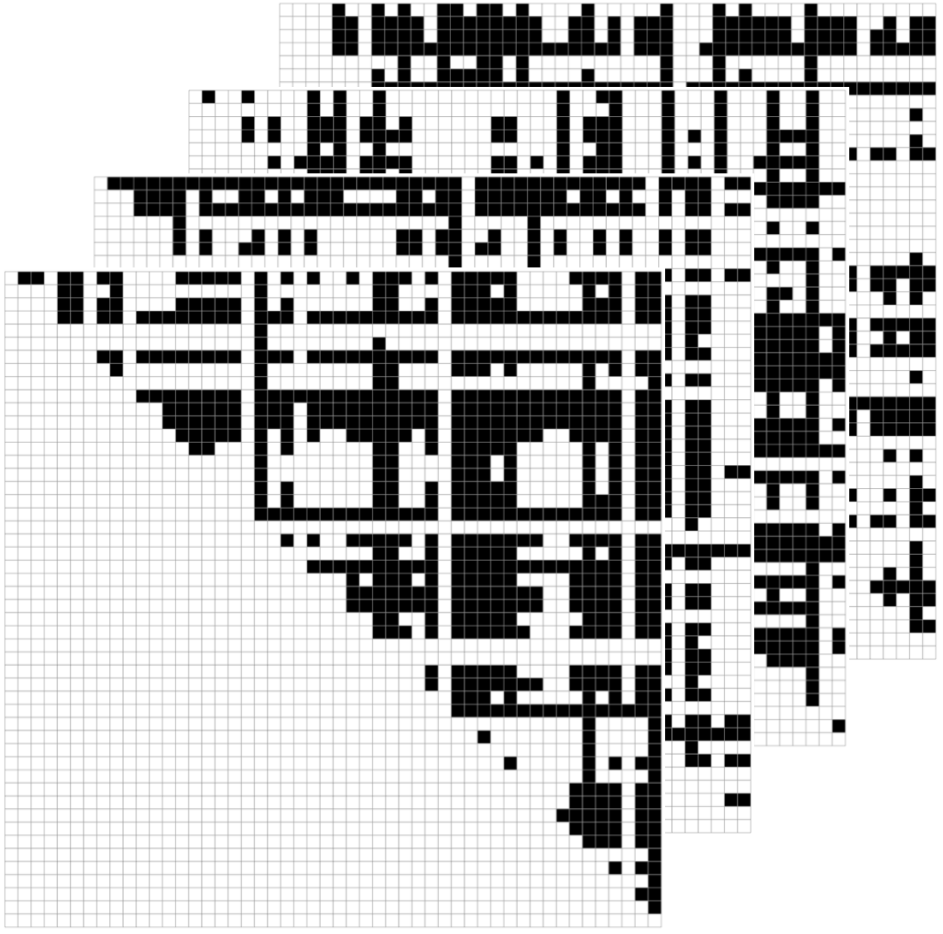
- Obviously, this representation as $O(p^2)$ bits exists for any ranking or permutation of p items
- Many other applications in learning over rankings, learning to rank, learning permutations etc...
- We are interested particularly in practical solutions when p is large

Related work: Top scoring pairs (TSP)



(Geman et al., 2004; Tan et al., 2005; Leek, 2009)

Practical challenge



- Need to store $O(p^2)$ bits per sample
- Need to train a model in $O(p^2)$ dimensions

Kernel trick

Theorem (Wahba, Schölkopf, ...)

Training a linear model over a representation $\Phi(x) \in \mathbb{R}^Q$ of the form:

$$\min_{w \in \mathbb{R}^Q} \frac{1}{n} \sum_{i=1}^{n} \ell(w^\top \Phi(x_i), y_i) + \lambda ||w||^2$$

can be done efficiently, independently of Q, if the kernel

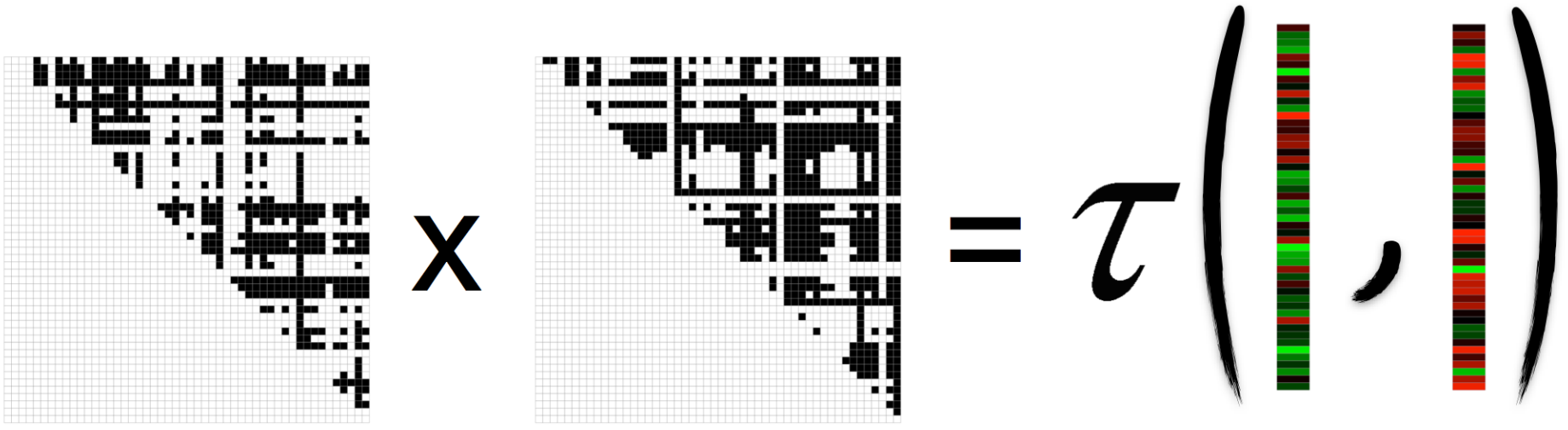
$$K(x, x') = \Phi(x)^\top \Phi(x')$$

can be computed efficiently.

Ex: ridge regression, $O(Q^3 + nQ^2)$ becomes $O(n^3 + n^2 T)$
Other: SVM, logistic regression, Cox model, survival SVM, ...

Kernel trick for us: Kendall's τ

$$\Phi(x)^\top \Phi(x') = \tau(x, x') \quad \text{(up to a scaling)}$$



O(p^2) O(p log(p))

Good news for SVM and kernel methods!

More formally

- For two permutations σ, σ' let $n_c(\sigma, \sigma')$ (resp. $n_d(\sigma, \sigma')$) the number of concordant (resp. discordant) pairs.
- The Kendall kernel (a.k.a. Kendall tau coefficient) is defined as

$$K_\tau(\sigma, \sigma') = \frac{n_c(\sigma, \sigma') - n_d(\sigma, \sigma')}{\binom{p}{2}} .$$

- The Mallows kernel is defined for any $\lambda \geq 0$ by

$$K_M^\lambda(\sigma, \sigma') = e^{-\lambda n_d(\sigma, \sigma')} .$$

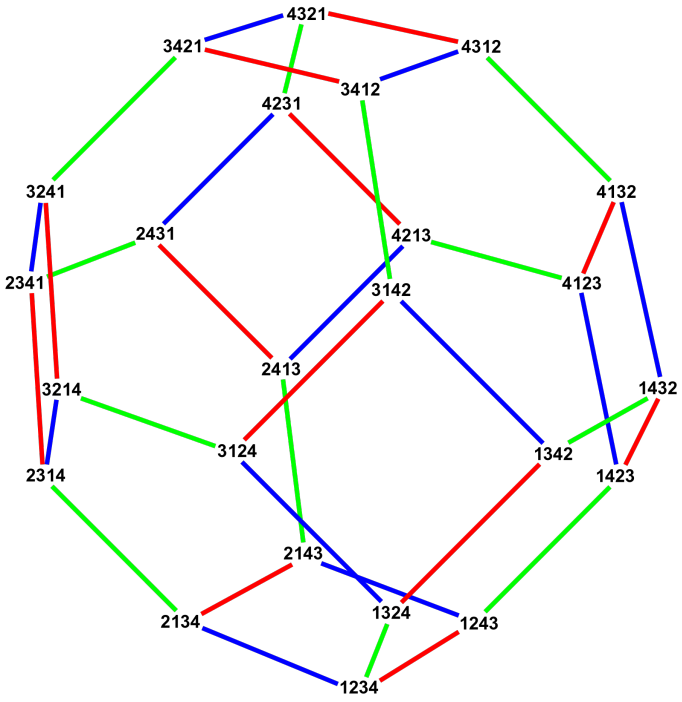
Theorem (Jiao and V., 2015)

The Kendall and Mallows kernels are positive definite.

Theorem (Knight, 1966)

These two kernels for permutations can be evaluated in $O(p \log p)$ time.

Related work



Cayley graph of $\mathbb{S}_4$

- Kondor and Barbarosa (2010) proposed the diffusion kernel on the Cayley graph of the symmetric group generated by adjacent transpositions.
- Computationally intensive ($O(p^p)$)
- Mallows kernel is written as

$$K_M^{\lambda}(\sigma, \sigma') = e^{-\lambda n_d(\sigma,\sigma')},$$

where $n_d(\sigma, \sigma')$ is the shortest path distance on the Cayley graph.

- It can be computed in $O(p \log p)$

Application: supervised classification

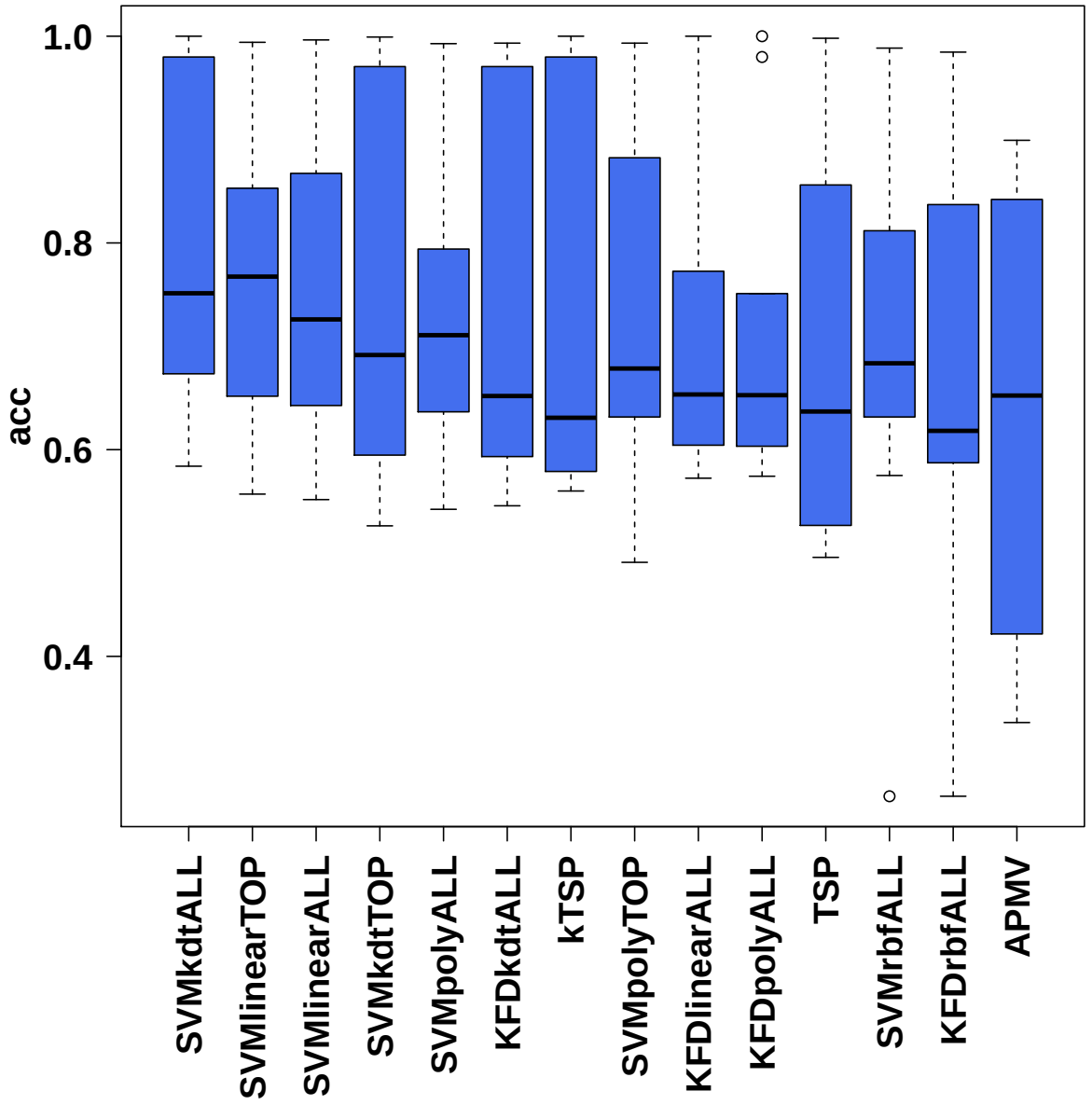
Datasets

Dataset	No. of features	No. of samples (training/test)	
		C_1	C_2
Breast Cancer 1	23624	44/7 (Non-relapse)	32/12 (Relapse)
Breast Cancer 2	22283	142 (Non-relapse)	56 (Relapse)
Breast Cancer 3	22283	71 (Poor Prognosis)	138 (Good Prognosis)
Colon Tumor	2000	40 (Tumor)	22 (Normal)
Lung Cancer 1	7129	24 (Poor Prognosis)	62 (Good Prognosis)
Lung Cancer 2	12533	16/134 (ADCA)	16/15 (MPM)
Medulloblastoma	7129	39 (Failure)	21 (Survivor)
Ovarian Cancer	15154	162 (Cancer)	91 (Normal)
Prostate Cancer 1	12600	50/9 (Normal)	52/25 (Tumor)
Prostate Cancer 2	12600	13 (Non-relapse)	8 (Relapse)

Methods

- Kernel machines Support Vector Machines (SVM) and Kernel Fisher Discriminant (KFD) with Kendall kernel, linear kernel, Gaussian RBF kernel, polynomial kernel.
- Top Scoring Pairs (TSP) classifiers [**?**].
- Hybrid scheme of SVM + TSP feature selection algorithm.

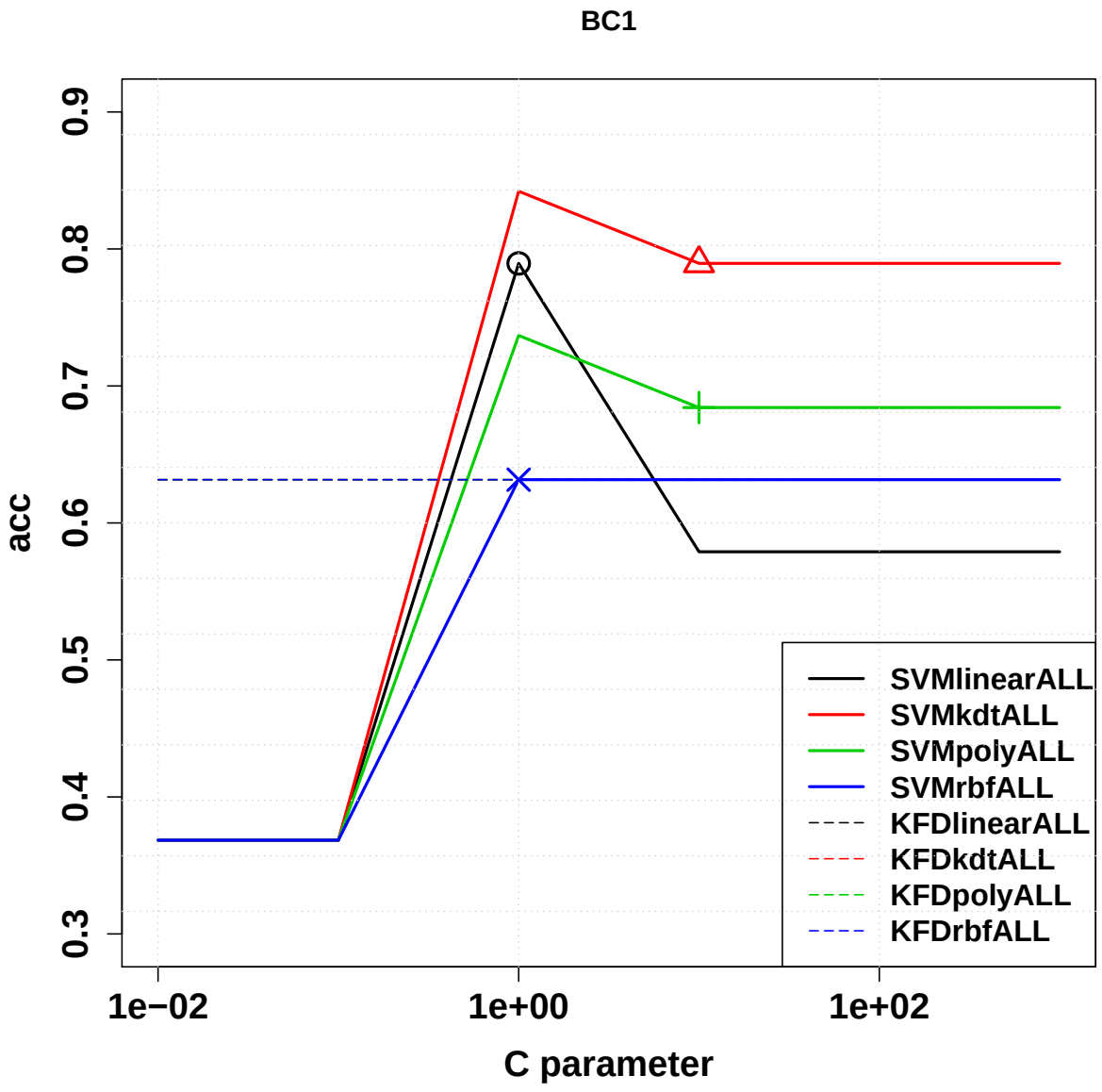
Results



Kendall kernel SVM

- **Competitive accuracy!**
- Less sensitive to regularization parameter!
- No need for feature selection!

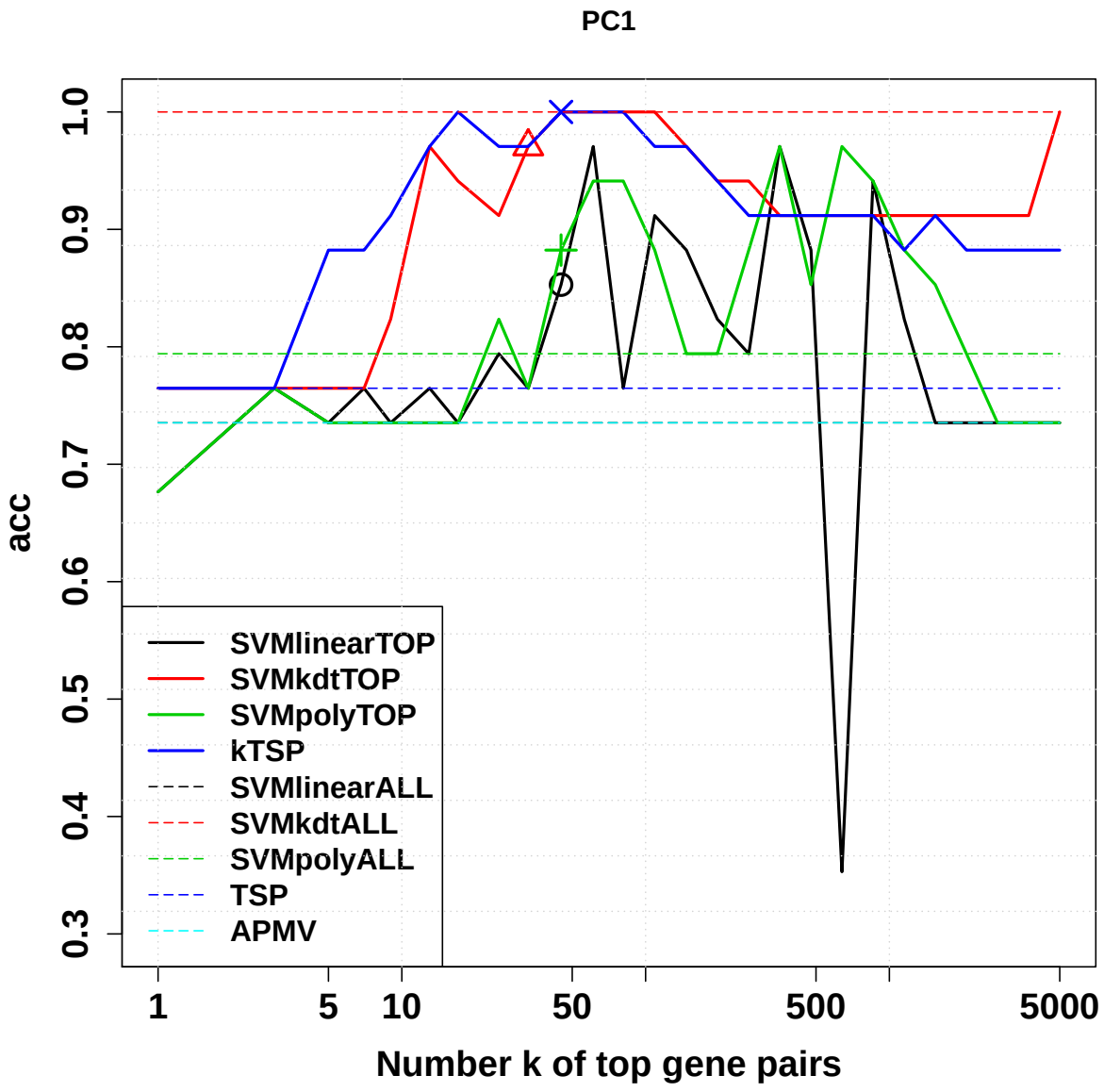
Results



Kendall kernel SVM

- Competitive accuracy!
- **Less sensitive to regularization parameter!**
- No need for feature selection!

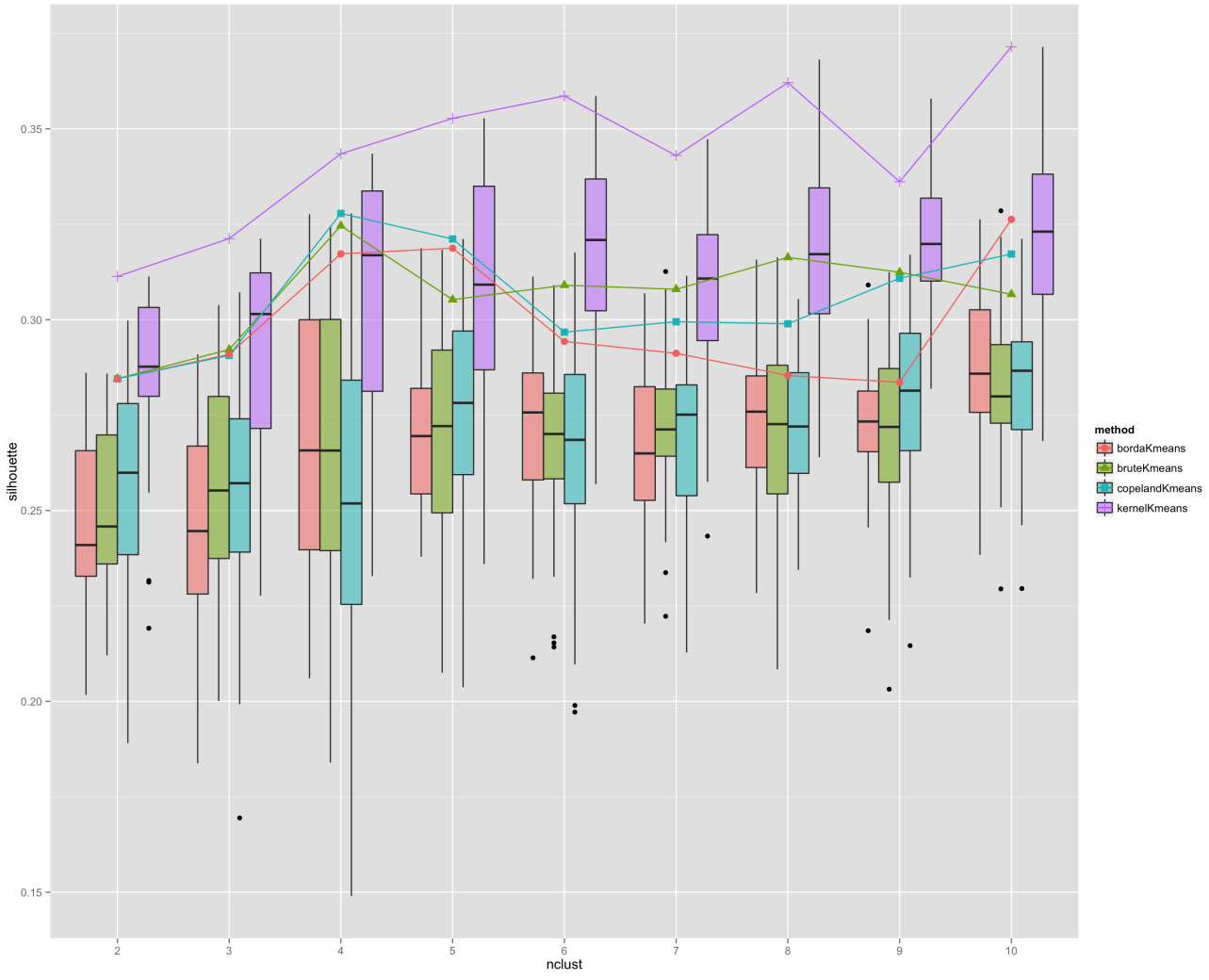
Results



Kendall kernel SVM

- Competitive accuracy!
- Less sensitive to regularization parameter!
- **No need for feature selection!**

Application: clustering



- APA data (full rankings)
- $n = 5738$, $p = 5$
- (new) Kernel k-means vs (standard) k-means in $\mathbb{S}_5$
- Show silhouette as a function of number of clusters (higher better)

Extension to partial rankings

- Two interesting types of partial rankings are interleaving partial ranking

$$x_{i_1} \succ x_{i_2} \succ \cdots \succ x_{i_k}, \quad k \leq n.$$

  and top-k partial ranking

$$x_{i_1} \succ x_{i_2} \succ \cdots \succ x_{i_k} \succ X_{\text{rest}}, \quad k \leq n.$$

- Partial rankings can be uniquely represented by a set of permutations compatible with all the observed partial orders.

Theorem

For these two particular types of partial rankings, the convolution kernel (Haussler, 1999) induced by Kendall kernel

$$K_\tau^\star(R, R') = \frac{1}{|R||R'|} \sum_{\sigma \in R} \sum_{\sigma' \in R'} K_\tau(\sigma, \sigma')$$

can be evaluated in $O(k \log k)$ time.

Extension to partial rankings

- Two interesting types of partial rankings are interleaving partial ranking

$$x_{i_1} \succ x_{i_2} \succ \cdots \succ x_{i_k}, \quad k \leq n.$$

  and top-k partial ranking

$$x_{i_1} \succ x_{i_2} \succ \cdots \succ x_{i_k} \succ X_{\text{rest}}, \quad k \leq n.$$

- Partial rankings can be uniquely represented by a set of permutations compatible with all the observed partial orders.

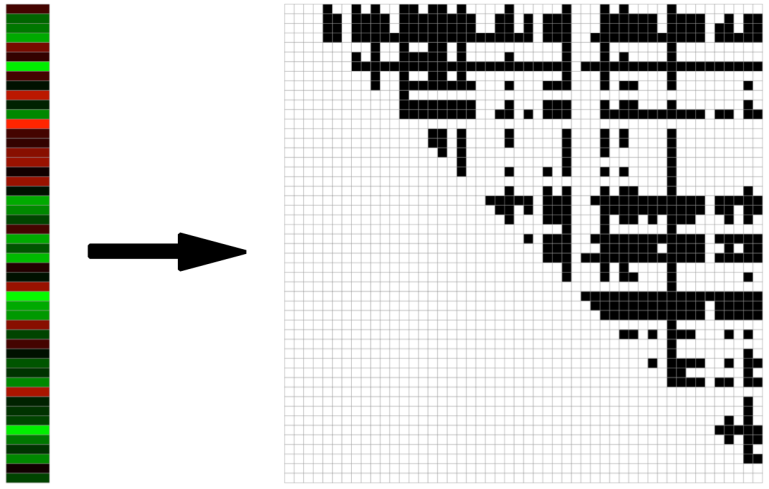
Theorem

For these two particular types of partial rankings, the convolution kernel (Haussler, 1999) induced by Kendall kernel

$$K_\tau^\star(R, R') = \frac{1}{|R||R'|} \sum_{\sigma \in R} \sum_{\sigma' \in R'} K_\tau(\sigma, \sigma')$$

can be evaluated in $O(k \log k)$ time.

Extension to smoother, continuous representations



One sample x
p features

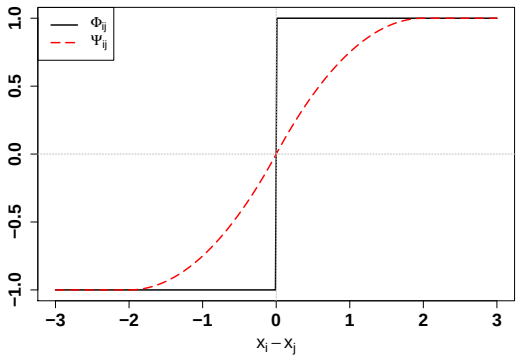
Mapping f(x)
p(p-1)/2 bits

- Instead of $\Phi : \mathbb{R}^p \to \{0,1\}^{p(p-1)/2}$, consider the continuous mapping $\Psi_a : \mathbb{R}^p \to \mathbb{R}^{p(p-1)/2}$:

$$\Psi_a(x) = \mathbb{E}\Phi(x+\epsilon) \quad \text{with} \quad \epsilon \sim (\mathcal{U}[-\frac{a}{2}, \frac{a}{2}])^n$$

- Corresponding kernel $G_a(x, x') = \Psi_a(x)^\top \Psi_a(x')$

Computation of $G(x, x')$



- $G_a(x, x')$ can be computed exactly in $O(p^2)$ by explicit computation of $\Psi_a(x)$ in $\mathbb{R}^{p(p-1)/2}$

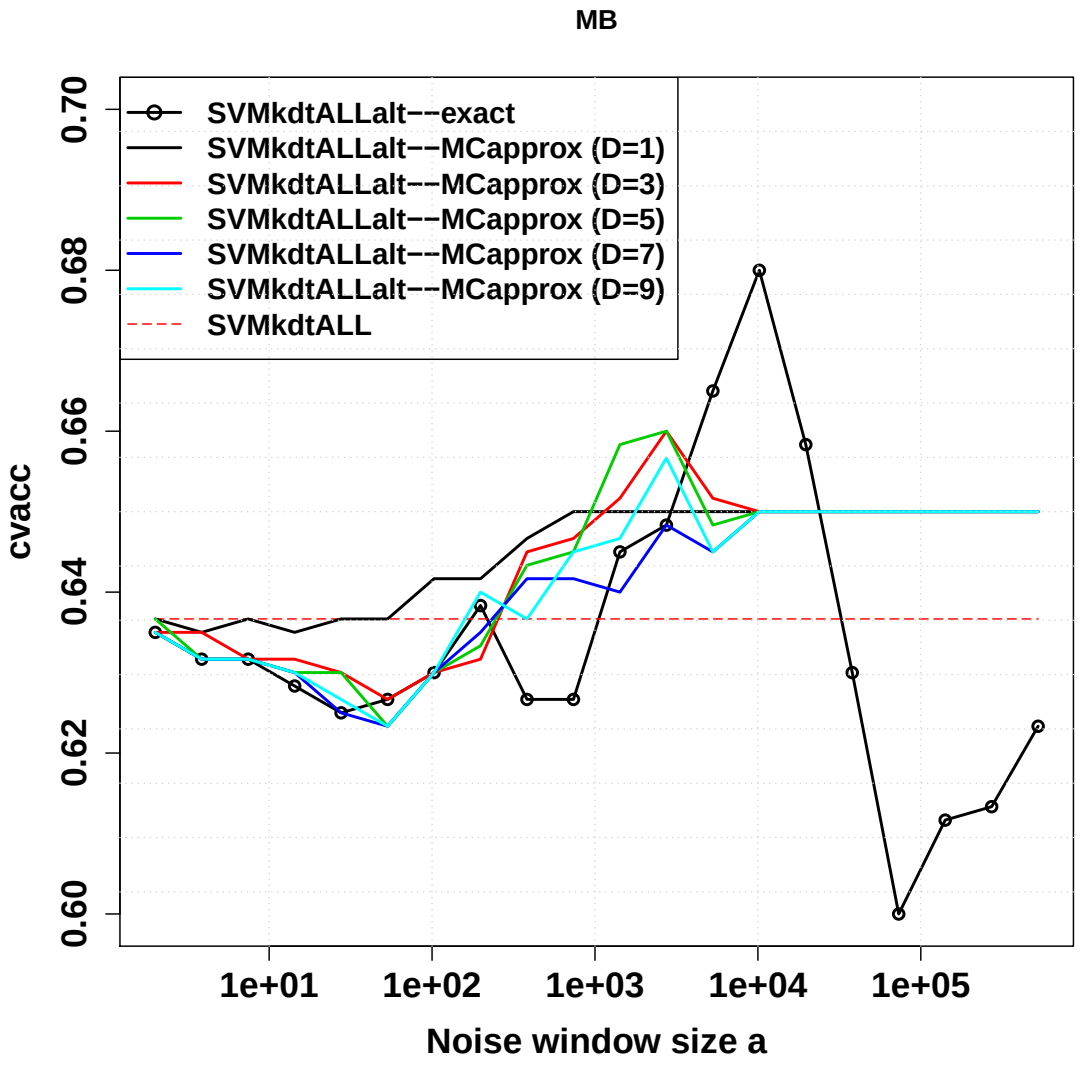
- $G_a(x, x')$ can be computed approximately in $O(D^2 p \log p)$ by Monte-Carlo approximation:

$$\tilde{G}_a(x, x') = \frac{1}{D^2} \sum_{i,j=1}^{D} K(x + \epsilon_i, x' + \epsilon'_j)$$

- Theorem: for supervised learning, Monte-Carlo approximation is better[1] than exact computation when $n = o(p^{1/3})$

[1] faster for the same accuracy

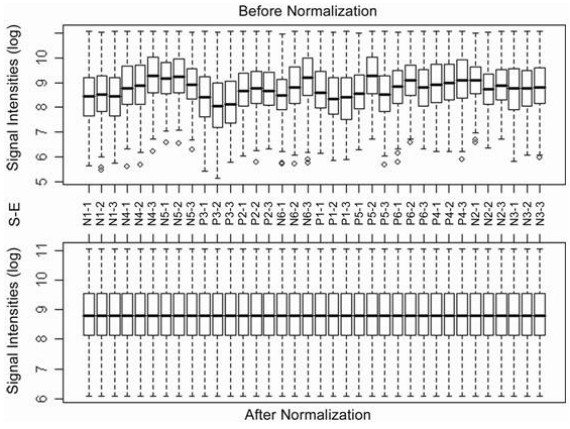
Performance of $G_a(x, x)$

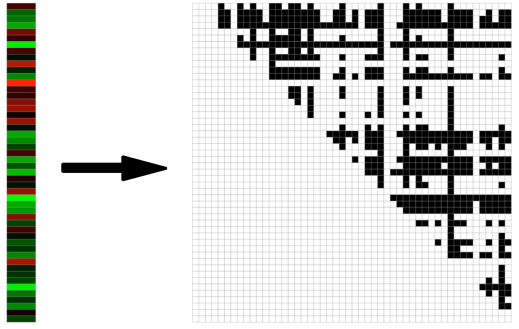


Outline

1. Patient stratification from somatic mutations using gene network
2. Supervised quantile normalization
3. Learning from rankings through pairwise comparisons
4. Conclusion

Conclusion





One sample x
p features

Mapping f(x)
p(p-1)/2 bits

- Representing omics data as permutations has some potential
  - NetNorM normalization of somatic mutation profiles
  - SUQUAN supervised quantile normalization as matrix regression
  - Kendall and Mallows kernel in $O(p \ln(p))$
- Understanding the benefits and cost of different representations remains very heuristic and sometimes counterintuitive
- Learning representation may help

Thanks

The Adolph C. and Mary Sprague Miller Institute for Basic Research in Science
University of California, Berkeley