Machine learning for precision medicine

Jean-Philippe Vert

Data Science Summer School, Ecole Polytechnique, June 26, 2018
1 body = $10^{14}$ human cells (and 100x more non-human cells)
1 cell = $6 \times 10^9$ ACGT coding for 20,000+ genes
The sequencing revolution
The sequencing revolution

Cost per Genome

Moore's Law

NIH National Human Genome Research Institute

genome.gov/sequencingcosts
Sequencing is a swiss army knife for "omics"

(Frese et al., 2013)
Cancer

In normal cells, cell division eventually stops.

Cells with damaged DNA

Cancer cell division

A cancer cell (1900)
A cancer cell (1960)
A cancer cell (2010)
All cancers are different

All happy families are alike; each unhappy family is unhappy in its own way.
- Leon Tolstoy, Anna Karenina.
Precision medicine

Personalized Cancer Therapy

1. Molecular Profiling
   - Markers predictive of drug sensitivity/resistance
   - Markers predictive of adverse events

2. Prognostic Markers
   - ABC1
   - AD1
   - BC1
   - CD1

https://pct.mdanderson.org
Precision medicine

Personalized Cancer Therapy

1. Molecular Profiling
2. Prognostic Markers
   - Markers predictive of drug sensitivity/resistance
   - Markers predictive of adverse events

https://pct.mdanderson.org
Machine learning
Machine learning
Many promising applications in health (mostly images) (Mobadersany et al., 2018)

Also: high-content screening, digital pathology, radiomics, skin diagnosis, EHR, ...
Omics data are more challenging

- Gene expression

- Somatic mutations

\[ n = 10^2 \sim 10^4 \text{ (patients)} \]
\[ p = 10^4 \sim 10^7 \text{ (genes, mutations, copy number, ...)} \]

Data of various nature (continuous, discrete, structured, ...)  
Data of variable quality (technical/batch variations, noise, ...)
Consequence: limited accuracy

Breast cancer prognosis competition, $n = 2000$ (Bilal et al., 2013)

- C: 16 standard clinical data (age, tumor size, ...)
- M: 80k molecular features (gene expression, DNA copy number)
- P: incorporate prior knowledge
Consequence: unstable biomarker selection

Gene expression profiling predicts clinical outcome of breast cancer

Laura J. van ’t Veer*, Hengyue Dai†, Marc J. van de Vijver*, Yudong He‡, Augustinus A. M. Hart*, Mao Mao‡, Hans L. Peterse†, Karin van der Kooy†, Matthew J. Marton‡, Anke T. Witteveen‡, George J. Schreiber‡, Ron M. Kerkhoven‡, Chris Roberts‡, Peter S. Linsley‡, René Bernards§ & Stephen H. Friend‡

* Divisions of Diagnostic Oncology, Radiotherapy and Molecular Carcinogenesis and Center for Biomedical Genetics, The Netherlands Cancer Institute, 121 Plesmanlaan, 1066 CX Amsterdam, The Netherlands
‡ Rosetta Inpharmatics, 12040 115th Avenue NE, Kirkland, Washington 98034.


3 genes in common

van ’t Veer et al. (2002); Wang et al. (2005)
Some research directions

- Regularize and incorporate prior knowledge
- Find a better representation
Outline

1. Regularize
2. Change representation
Outline

1. Regularize
2. Change representation
Typical problem

- $n$ samples (patients), $p$ features (genes)
- $X \in \mathbb{R}^{n \times p}$ gene expression profile of each patient
- $Y \in \mathcal{Y}^n$ survival information of each patient
- Fit a linear model for a sample $x \in \mathbb{R}^p$:
  \[
  f(x) = \beta^\top x = \sum_{i=1}^{p} \beta_i x_i
  \]

- Standard methods (least squares or logistic regression) won’t work because $n < p$
In high dimension, estimate $\beta$ by solving

$$\min_{\beta \in \mathbb{R}^p} R(Y, X\beta) + \lambda J(\beta),$$

where

- $R(Y, X\beta)$ is an empirical risk to measures the fit to the training data
- $J(\beta)$ is a penalty to control the complexity of the model
- $\lambda > 0$ is a regularization parameter
Standard empirical risk

\[
\min_{\beta \in \mathbb{R}^p} R(Y, X\beta) + \lambda J(\beta)
\]

\[
R(Y, X\beta) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, \beta^\top x_i)
\]

where

- **MSE**: \( \ell(y, t) = (t - y)^2 \) for regression \((y \in \mathbb{R})\)
- **Logistic loss**: \( \ell(y, t) = \ln(1 + e^{-ty}) \) for classification \((y \in \{-1, 1\})\)
- **Hinge loss**: \( \ell(y, t) = \max(0, 1 - ty) \) for classification \((y \in \{-1, 1\})\)
Standard regularizations

\[
\min_{\beta \in \mathbb{R}^p} R(Y, X\beta) + \lambda J(\beta)
\]

where

- **Lasso**: \( J(\beta) = \|\beta\|_1 \) for gene selection.
- **Ridge**: \( J(\beta) = \|\beta\|_2^2 \) to address \( n \gg m \).
- **Elastic net**: \( J(\beta) = \alpha \|\beta\|_2^2 + (1 - \alpha)\|\beta\|_1 \)

Estimation returned by lasso (left) vs. ridge (right) Tibshirani (1996).
Which regularization is the best?

- **Feature selection** (lasso, t-tests, ...) is popular, it leads to a limited set of genes that form a molecular signatures
- **Ridge** is less interpretable but often leads to better performance... e.g., breast cancer prognosis \( (n = 286) \):

![Graph showing the influence of signature size on breast cancer prognosis performance.](image)

- **Signature size**
- **AUC**

The graph illustrates the performance of a regularised LR classifier using a signature of varying size trained on the Wang expression dataset to predict relapse within 5 years. The genes in the signatures are selected either randomly, or by decreasing significance according to a t-test. The performance is estimated by 5-fold cross-validation, averaged over 10 repeats. In this example, it is better to keep all genes to train the classifier.
Adding prior knowledge: network-based regularizations

\[ \mathcal{G} = (\mathcal{V}, \mathcal{E}) \] a graph of genes (PPI, metabolic, signaling, regulatory network...)

Prior knowledge:
- \( \beta \) should be "smooth" on the graph?
- Selected genes should be connected?
Examples of network-based regularizations

\[ J_G(\beta) = \sum_{i \sim j} (\beta_i - \beta_j)^2 \]  
(Rapaport et al., 2007)

\[ J_G(\beta) = a\|\beta\|_1 + (1 - a) \sum_{i \sim j} (\beta_i - \beta_j)^2 \]  
(Li and Li, 2008)

\[ J_G(\beta) = \sup_{\alpha \in \mathbb{R}^p : \forall i \sim j \alpha_i^2 + \alpha_j^2 \leq 1} \alpha^\top \beta \]  
(Jacob et al., 2009)

\[ J_G(\beta) = a\|\beta\|_1 + (1 - a) \sum_{i \sim j} |\beta_i - \beta_j| \]  
(Hoefling, 2010)
Gene selection with the graph lasso

\[ J_G(\beta) = \sup_{\alpha \in \mathbb{R}^p : \forall i \sim j, \|\alpha_i^2 + \alpha_j^2\| \leq 1} \alpha^T \beta \]

Jacob et al. (2009)
BC prognosis: Lasso signature (accuracy 0.61)

Jacob et al. (2009)
BC prognosis: Graph Lasso signature (accuracy 0.64)

Jacob et al. (2009)
Smoothness regularization and Fourier transform

- "Connected genes have similar weights" (Rapaport et al., 2007; Li and Li, 2008)

\[ J_G(\beta) = \sum_{i \sim j} (\beta_i - \beta_j)^2 \]

- No feature selection
- Reinterpretation in the Fourier domain (Rapaport et al., 2007):

\[ \sum_{i \sim j} (\beta_i - \beta_j)^2 = \sum_{i=1}^{p} \lambda_i \hat{\beta}_i^2 \]

where
- \( \hat{\beta}_i \) is the \( i \)-th Fourier coefficient of \( \beta \)
- \( \lambda_i \) is the \( i \)-th frequency

"\( \beta \) has little energy at high frequency" and is therefore smooth on the graph
"Connected genes have similar weights" (Rapaport et al., 2007; Li and Li, 2008)

\[ J_G(\beta) = \sum_{i \sim j} (\beta_i - \beta_j)^2 \]

- No feature selection
- Reinterpretation in the Fourier domain (Rapaport et al., 2007):

\[ \sum_{i \sim j} (\beta_i - \beta_j)^2 = \sum_{i=1}^{p} \lambda_i \hat{\beta}_i^2 \]

where
- \( \hat{\beta}_i \) is the \( i \)-th Fourier coefficient of \( \beta \)
- \( \lambda_i \) is the \( i \)-th frequency

"\( \beta \) has little energy at high frequency" and is therefore smooth on the graph
Eigenvectors $U$ of the graph Laplacian matrix form the Fourier basis:

$$\hat{\beta} = U^T \beta$$

Eigenvalues $\Lambda = (0 = \lambda_1 \leq \ldots \leq \lambda_p)$ represent the "frequencies" of the Fourier basis.
Graph Fourier transform $\hat{\beta}$?

- Eigenvectors $U$ of the graph Laplacian matrix form the Fourier basis:
  \[ \hat{\beta} = U^T \beta \]

- Eigenvalues $\Lambda = (0 = \lambda_1 \leq \ldots \leq \lambda_p)$ represent the "frequencies" of the Fourier basis
Graph Fourier transform \( \hat{\beta} \)?

- Eigenvectors \( U \) of the graph Laplacian matrix form the Fourier basis:

\[
\hat{\beta} = U^T \beta
\]

- Eigenvalues \( \Lambda = (0 = \lambda_1 \leq \ldots \leq \lambda_p) \) represent the "frequencies" of the Fourier basis.
Graph Fourier transform $\hat{\beta}$?

- Eigenvectors $U$ of the graph Laplacian matrix form the Fourier basis:
  $$\hat{\beta} = U^T \beta$$

- Eigenvalues $\Lambda = (0 = \lambda_1 \leq \ldots \leq \lambda_p)$ represent the "frequencies" of the Fourier basis.

![Graph Laplacian matrix with eigenvalues and eigenvectors](image)
Graph Fourier transform $\hat{\beta}$?

- Eigenvectors $U$ of the graph Laplacian matrix form the Fourier basis:
  \[ \hat{\beta} = U^T \beta \]

- Eigenvalues $\Lambda = (0 = \lambda_1 \leq \ldots \leq \lambda_p)$ represent the "frequencies" of the Fourier basis.

Lambda = 2.2

\[
\begin{array}{|c|c|c|c|c|}
\hline
2 & 4 & 6 & 8 &
\hline
\end{array}
\]

\[
\begin{array}{|c|c|c|c|c|}
\hline
-0.4 & -0.2 & 0.0 & 0.2 & 0.4
\hline
\end{array}
\]
Graph Fourier transform $\hat{\beta}$?

- Eigenvectors $U$ of the graph Laplacian matrix form the Fourier basis:
  \[ \hat{\beta} = U^T \beta \]

- Eigenvalues $\Lambda = (0 = \lambda_1 \leq \ldots \leq \lambda_p)$ represent the "frequencies" of the Fourier basis.
Graph Fourier transform $\hat{\beta}$?

- Eigenvectors $U$ of the graph Laplacian matrix form the Fourier basis:
  
  $$\hat{\beta} = U^T \beta$$

- Eigenvalues $\Lambda = (0 = \lambda_1 \leq \ldots \leq \lambda_p)$ represent the "frequencies" of the Fourier basis.

![Graph Laplacian Eigenvectors and Eigenvalues](image)

Lambda = 3.6
Graph Fourier transform $\hat{\beta}$?

- Eigenvectors $U$ of the graph Laplacian matrix form the Fourier basis:

$$\hat{\beta} = U^T \beta$$

- Eigenvalues $\Lambda = (0 = \lambda_1 \leq \ldots \leq \lambda_p)$ represent the "frequencies" of the Fourier basis

![Graph with labeled lambda and Lambda values](image-url)
Graph Fourier transform $\hat{\beta}$?

- Eigenvectors $U$ of the graph Laplacian matrix form the Fourier basis:
  \[ \hat{\beta} = U^T \beta \]

- Eigenvalues $\Lambda = (0 = \lambda_1 \leq \ldots \leq \lambda_p)$ represent the "frequencies" of the Fourier basis

![Graph Laplacian Matrix Diagram]

Lambda = 6.3

Lambda = 3.9
Rapaport et al. (2007) extends

\[ \sum_{i \sim j} (\beta_i - \beta_j)^2 = \sum_{i=1}^{p} \lambda_i \hat{\beta}_i^2 \]

to

\[ \sum_{i=1}^{p} \phi(\lambda_i) \hat{\beta}_i^2 \]

for \( \phi: \mathbb{R}^+ \rightarrow \mathbb{R}^+ \) non-decreasing.

Example: \( \phi(\lambda) = \exp(-\gamma \lambda) \) linked to the diffusion kernel on the graph.
1. Regularize

2. Change representation
The regularization / representation duality

\[
\min_{\beta \in \mathbb{R}^p} R(Y, X\beta) + \lambda \beta^T L\beta \\
= \min_{\beta \in \mathbb{R}^p} R(Y, XL^{-1/2}L^{1/2}\beta) + \lambda \beta^T L^{1/2}L^{1/2}\beta \\
= \min_{\gamma \in \mathbb{R}^p} R(Y, XL^{-1/2}\gamma) + \lambda \gamma^T \gamma \\
= \min_{\gamma \in \mathbb{R}^p} R(Y, \psi(X)\gamma) + \lambda \gamma^T \gamma
\]

where

\[
\psi(X) = XL^{-1/2}
\]

- This amounts to standard ridge regularization, after data smoothing on the graph
- Useful in practical implementation
- The duality between regularization/representation is not always so easy, it is worth investigating both directions
Large-scale efforts to collect somatic mutations

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

<table>
<thead>
<tr>
<th>Cancer type</th>
<th>Patients</th>
<th>Genes</th>
</tr>
</thead>
<tbody>
<tr>
<td>LUAD (Lung adenocarcinoma)</td>
<td>430</td>
<td>20 596</td>
</tr>
<tr>
<td>SKCM (Skin cutaneous melanoma)</td>
<td>307</td>
<td>17 463</td>
</tr>
<tr>
<td>GBM (Glioblastoma multiforme)</td>
<td>265</td>
<td>14 750</td>
</tr>
<tr>
<td>BRCA (Breast invasive carcinoma)</td>
<td>945</td>
<td>16 806</td>
</tr>
<tr>
<td>KIRC (Kidney renal clear cell carcinoma)</td>
<td>411</td>
<td>10 609</td>
</tr>
<tr>
<td>HNSC (Head and Neck squamous cell carcinoma)</td>
<td>388</td>
<td>17 022</td>
</tr>
<tr>
<td>LUSC (Lung squamous cell carcinoma)</td>
<td>169</td>
<td>13 590</td>
</tr>
<tr>
<td>OV (Ovarian serous cystadenocarcinoma)</td>
<td>363</td>
<td>10 195</td>
</tr>
</tbody>
</table>
Patient stratification (unsupervised) from raw mutation profiles

✓ Non-Negative matrix factorisation (NMF)

✓ Desired behaviour:

✓ Observed behaviour:

Patients share very few mutated genes!
Survival prediction from raw mutation profiles

- Each patient is a **binary vector**: each gene is mutated (1) or not (0)
- Silent mutations are removed
- Survival model estimated with sparse survival SVM
- Results on 5-fold cross-validation repeated 4 times
Approach: change representation?

Can we replace

\[ x \in \{0, 1\}^p \] with \( p \) very large, very sparse

by a representation with more information shared between samples

\[ \Phi(x) \in \mathcal{H} \]

that would allow better supervised and unsupervised classification?
Network-based stratification of tumor mutations

Matan Hofree¹, John P Shen², Hannah Carter², Andrew Gross³ & Trey Ideker¹–³

¹Department of Computer Science and Engineering, University of California, San Diego, La Jolla, California, USA. ²Department of Medicine, University of California, San Diego, La Jolla, California, USA. ³Department of Bioengineering, University of California, San Diego, La Jolla, California, USA. Correspondence should be addressed to T.I. (tideker@ucsd.edu).

RECEIVED 14 FEBRUARY; ACCEPTED 12 AUGUST; PUBLISHED ONLINE 15 SEPTEMBER 2013; DOI:10.1038/NMETH.2651

Network smoothing:
- Gene
- Gene-gene interaction
- Patient genotype 1
- Patient genotype 2
- Co-occurrence of genotype 1 and 2

Network-based stratification:
- Affinity matrix
- Post-smoothing matrix
- Network influence constraint


d
Network-based stratification

NPG © 2013 Nature America, Inc. All rights reserved.
NetNorm Overview (Le Morvan et al., 2017)

Take

\[ \mathcal{H} = \left\{ x \in \{0, 1\}^p : \sum_{i=1}^{p} x_i = K \right\} \]

and use a gene network to transform \( x \) to \( \phi(x) \in \mathcal{H} \) by adding/removing mutations.
1. **Add** mutations for patients with **few** (less than $K$) mutations

2. **Remove** mutations for patients for **many** (more than $K$) mutations

In practice, $K$ is a free parameter optimized on the training set, typically a few 100's.
Results: unsupervised classification

Log-rank statistic

Number of subtypes

Survival probability

Time (months)
Results: survival prediction

Use Pathway Commons as gene network.
NSQN = Network Smoothing / Quantile Normalization (Hofree et al., 2013)
Learning from genomic data is **challenging**

**Regularization** is needed in high dimension

A good **representation** is worth a thousand learning algorithms

Subtle **interplay** between biology and math/CS

**Impact** on the final quality/performance of the model

Recent trend: **learn** the representation
1/!

PROMOTION DU PROGRAMME

• Rencontre avec les directeurs, directeurs scientifiques et directeurs des études de l’ENS, Mines, ENSCP, ESPCI pour organiser la promotion du programme auprès des étudiants.

• Ecole des Mines : présentation d’ITI devant le conseil de l’enseignement ; 8 représentants d’élèves ; relance

• ENSCP : mailing de présentation d’ITI aux 3A via la direction des études.

• ENS : mailing ; présentation directe auprès des étudiants (2 élèves présents) ; diffusion des plaquettes et du syllabus ; relance

• ESPCI : mailing aux 3A + présentation d’ITI / rencontre avec les étudiants en présence de la directrice de la scolarité et de la directrice des relations entreprises ; 8 élèves présents

• Contacts en cours avec les Ponts et l’ENS Lyon (en attente de réponse)

• Rencontre à l’ANRT avec le délégué général et la chef du service CIFRE : accord de communication sur le programme ITI via le site internet de l’ANRT (rubrique “zoom sur”)

• Promotion aux Rencontres Universités-Entreprises (RUE)

• Echange avec Stéphane Mallat et réflexion sur la pertinence du programme tronc commun

• Rencontre avec le responsable des relations internationales de la National Taiwan University à l’ESPCI : présentation d’ITI

• Contact en cours pour visite d’entreprise

• Article dans les Echos

• Création du logo PSL/ITI et du Diplôme Supérieur de Recherche et d’Innovation de PSL/ITI (DSRI), dépôt INPI en cours.

2/!

MISE EN ŒUVRE OPERATIONNELLE

DU PROGRAMME


NetNorM and NSQN benefit from biological information in the gene network.

Comparison with 10 randomly permuted networks:
Selected genes represent "true" or "proxy" mutations

Genes selected in at least 50% of the cross-validated sparse SVM model
Proxy mutations encode both total number of mutations and local mutational burden.
Adding good old clinical factors

Combination by averaging predictions
QN after network smoothing

- Patient A - NS
- Patient B - NS
- Patient A - NSQN
- Patient B - NSQN

Mutation values vs. Sorted genes
Another representation

\[ \Phi_{i,j}(x) = \begin{cases} 
1 & \text{if } x_i \leq x_j, \\
0 & \text{otherwise.} 
\end{cases} \]
Geometry of the embedding

For any two permutations $\sigma, \sigma' \in S_n$:

- Inner product
  $$\Phi(\sigma)^\top \Phi(\sigma') = \sum_{1 \leq i \neq j \leq n} 1_{\sigma(i) < \sigma(j)} 1_{\sigma'(i) < \sigma'(j)} = n_c(\sigma, \sigma')$$

  $n_c = \text{number of concordant pairs}$

- Distance
  $$\| \Phi(\sigma) - \Phi(\sigma') \|^2 = \sum_{1 \leq i, j \leq n} (1_{\sigma(i) < \sigma(j)} - 1_{\sigma'(i) < \sigma'(j)})^2 = 2n_d(\sigma, \sigma')$$

  $n_d = \text{number of discordant pairs}$
The Kendall kernel is

\[ K_{\tau}(\sigma, \sigma') = n_c(\sigma, \sigma') \]

The Mallows kernel is

\[ \forall \lambda \geq 0 \quad K_{M}^{\lambda}(\sigma, \sigma') = e^{-\lambda n_d(\sigma, \sigma')} \]

Theorem (Jiao and Vert, 2015, 2017)

The Kendall and Mallows kernels are positive definite.

Theorem (Knight, 1966)

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

*Kernel trick useful with few samples in large dimensions*
Kondor and Barbarosa (2010) proposed the diffusion kernel on the Cayley graph of the symmetric group generated by adjacent transpositions.

Computationally intensive ($O(n^{2n})$)

Mallows kernel is written as

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

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

It can be computed in $O(n \log n)$
Average performance on 10 microarray classification problems (Jiao and Vert, 2017).