An introduction to convex methods for life science
Introduction

Math et Sciences du Vivant – Université Paris-Saclay / Paris-Sud

Autumn semester 2017

http://julien.cremeriefamily.info

Outline

Intervenants

Julien Chiquet

Researcher in Statistics,
AgroParistech/INRA

julien.chiquet@inra.fr

Estelle Kuhn

Researcher in Statistics
INRA Jouy en Josas

estelle.kuhn@inra.fr

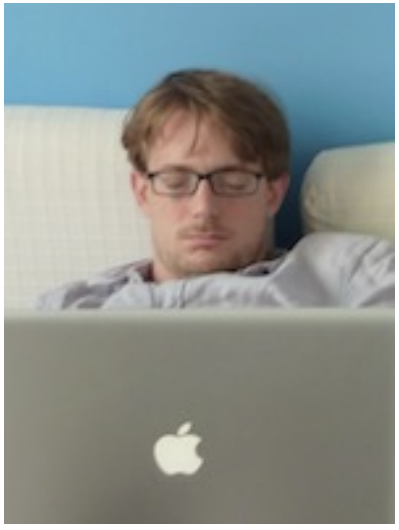
Sylvain Faure

IR CNRS
Orsay

sylvain.faure@math.u-psud.fr

Intervenants

Julien Chiquet



Researcher in Statistics,
AgroParistech/INRA

julien.chiquet@inra.fr

Estelle Kuhn

Researcher in Statistics
INRA Jouy en Josas

estelle.kuhn@inra.fr

Sylvain Faure

IR CNRS
Orsay

sylvain.faure@math.u-psud.fr

Agenda

1. Convex Optimization, me and myself
   - 1.1 Convex Optimization 1 (28/09)
     - Motivation and classes of problem
   - 1.2 Convex Optimization 2 (05/10)
     - Smooth convex problems: Gradient methods, Netwon method
   - 1.3 Convex Optimization 3 (12/10)
     - Non smooth convex problems: subgradient methods, proximal methods
2. Stochastic Optimization: Estelle Kuhn, 4 courses
   - Stochastic gradient descent
   - Expectation-Maximization algorithm and variants
   - Monte-Carlo Markov Chains
3. Numerical simulation Sylvain Faure, 3 praticals

Mark: written test for part 1 and 2 + practical for part 3

Agenda

1. Convex Optimization, me and myself
   - 1.1 Convex Optimization 1 (28/09)
     - Motivation and classes of problem
   - 1.2 Convex Optimization 2 (05/10)
     - Smooth convex problems: Gradient methods, Netwon method
   - 1.3 Convex Optimization 3 (12/10)
     - Non smooth convex problems: subgradient methods, proximal methods
2. Stochastic Optimization: Estelle Kuhn, 4 courses
   - Stochastic gradient descent
   - Expectation-Maximization algorithm and variants
   - Monte-Carlo Markov Chains
3. Numerical simulation Sylvain Faure, 3 praticals

Mark: written test for part 1 and 2 + practical for part 3

Agenda

1. Convex Optimization, me and myself
   - 1.1 Convex Optimization 1 (28/09)
     - Motivation and classes of problem
   - 1.2 Convex Optimization 2 (05/10)
     - Smooth convex problems: Gradient methods, Netwon method
   - 1.3 Convex Optimization 3 (12/10)
     - Non smooth convex problems: subgradient methods, proximal methods
2. Stochastic Optimization: Estelle Kuhn, 4 courses
   - Stochastic gradient descent
   - Expectation-Maximization algorithm and variants
   - Monte-Carlo Markov Chains
3. Numerical simulation Sylvain Faure, 3 praticals

Mark: written test for part 1 and 2 + practical for part 3

Agenda

1. Convex Optimization, me and myself
   - 1.1 Convex Optimization 1 (28/09)
     - Motivation and classes of problem
   - 1.2 Convex Optimization 2 (05/10)
     - Smooth convex problems: Gradient methods, Netwon method
   - 1.3 Convex Optimization 3 (12/10)
     - Non smooth convex problems: subgradient methods, proximal methods
2. Stochastic Optimization: Estelle Kuhn, 4 courses
   - Stochastic gradient descent
   - Expectation-Maximization algorithm and variants
   - Monte-Carlo Markov Chains
3. Numerical simulation Sylvain Faure, 3 praticals

Mark: written test for part 1 and 2 + practical for part 3

Outline

Convex optimisation: background

1. Basics in **Mathematical Analysis**
2. Basics in **Algebra**
3. Basics in **Matrix Calculus**
4. Basics in **probability and statistics**

Classical References for convex optimization

- Convex Optimization,
  Stephen Boyd and Lieve Lieven Vandenberghe
  https://web.stanford.edu/~boyd/cvxbook/
- Introductory Lectures on Convex Optimization,
  Y. Nesterov
  http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.693.855&rep=rep1&type=pdf

Courses online and people I steal from

1. Lieve Vandenberghe
   `http://www.seas.ucla.edu/~vandenbe/ee236b/ee236b.html`
2. Francis Bach
   `http://www.di.ens.fr/~fbach/orsay2017.html`
3. Alexandre d'Aspremont
   `http://www.di.ens.fr/~aspremon/MathSVM2.html`
4. Ryan Tibshirani
   `http://www.stat.cmu.edu/~ryantibs/convexopt/`
5. Stéphane Mottelet
   `http://www.utc.fr/~mottelet/polytex/cours.pdf`

Outline

Why optimization in Statistics?

Statistics deeply relies on Optimization

At some point performing estimation require solving

$$P: \qquad \arg\min_{\boldsymbol{\theta} \in \mathcal{C}} \ell(\mathbf{data}; \boldsymbol{\theta}),$$

where ℓ is a loss function and $\mathcal{C}$ some set of interest.

"Old standard" paradigm

Large sample size n, small number of parameter p

- for simple models, P can be solved analytically
- for complex models, non linear optimization solver are used

Why Optimization in life science?

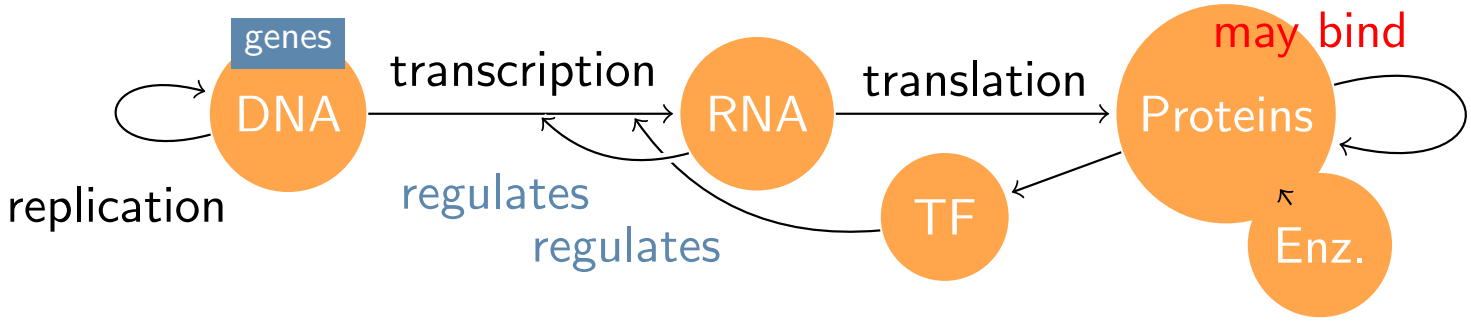
"New standard" paradigm

- Data are gathered massively but n grows more slowly than p.
- This affects the way we do statistics, ...
- ... and thus the way we use optimization.

$\rightsquigarrow$ We review some questions and statistical tasks in genomics that underlies optimization problems like P but require a new point of view.

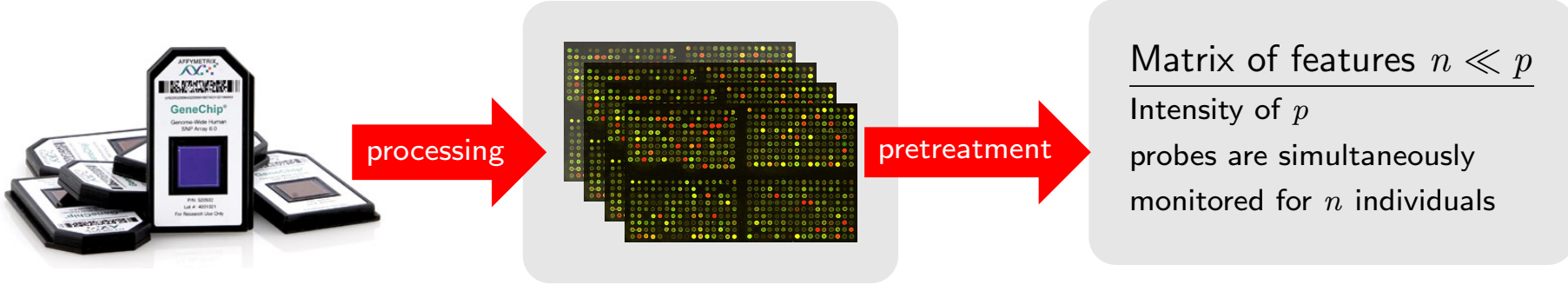
Genomics: an archetype for complex data

Goal: understanding the genomic processes that rule the cell



How: by monitoring many features at once in the cell

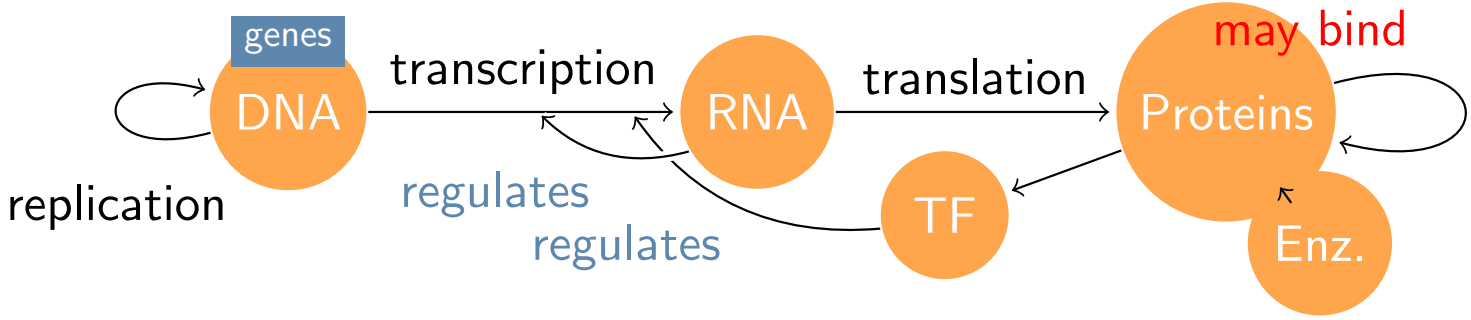
microarrays (hybridization)



Applications: medicine, agronomy, ecology, phylogeny . . .

Genomics: an archetype for complex data

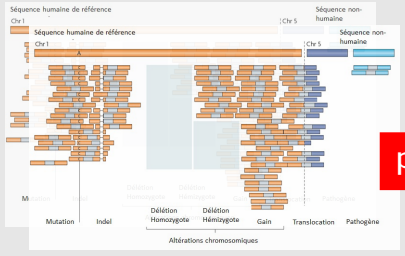
Goal: understanding the genomic processes that rule the cell



How: by monitoring many features at once in the cell

sequencing technology

assembling



pretreatment

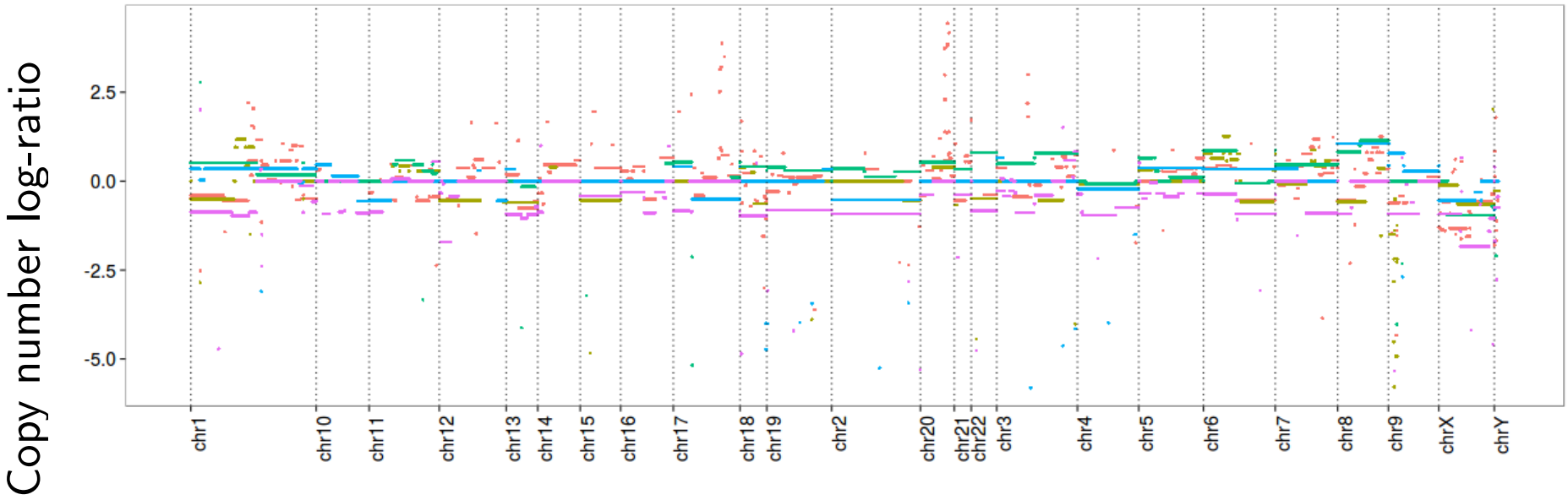
Matrix of features $n \lll p$

Counts are extracted from small repeated sequences and monitored for n individuals

Applications: medicine, agronomy, ecology, phylogeny ...

Chromosomal copy number changes, genome

aCGH Agilent 44K Human array for 5 cell lines



position along the genome ($\approx$ 44,000 sites)

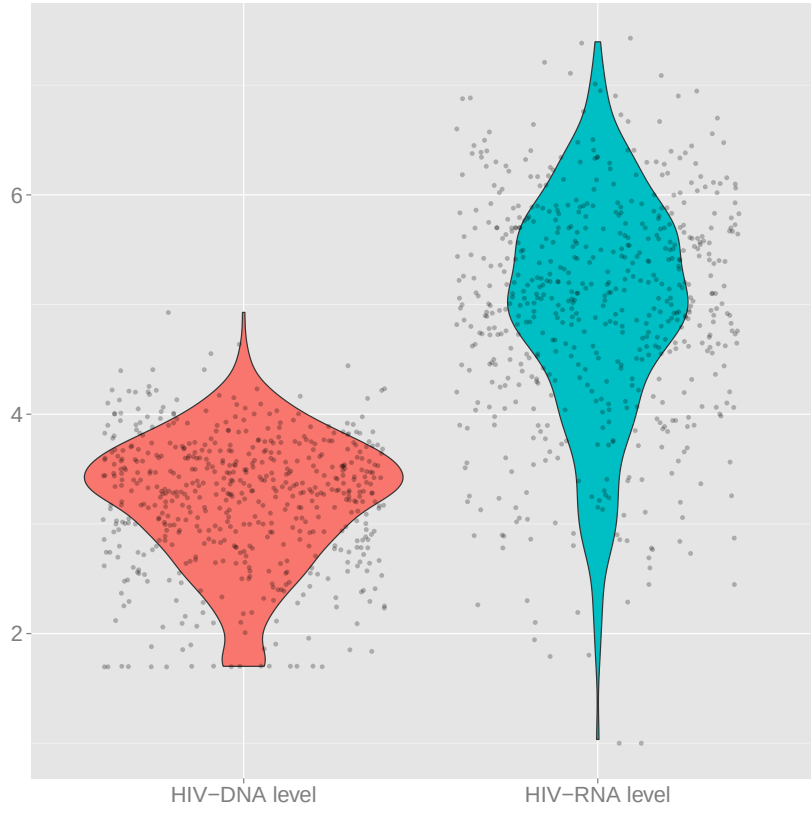
Figure: One color per breast cancer cell line

Goal: detecting genetic aberration

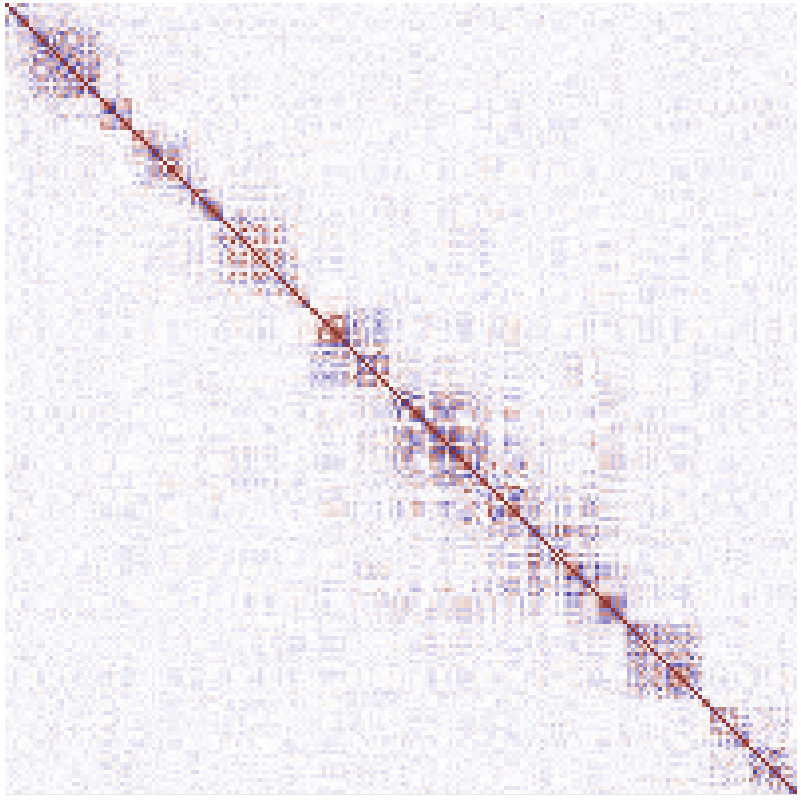
Task: segmentation/denoising; clustering

SNP-genotyping in Human with AIDS, genome

Illumina HapMap300 array of hundreds of individuals, millions of SNP



Indicators of disease progression



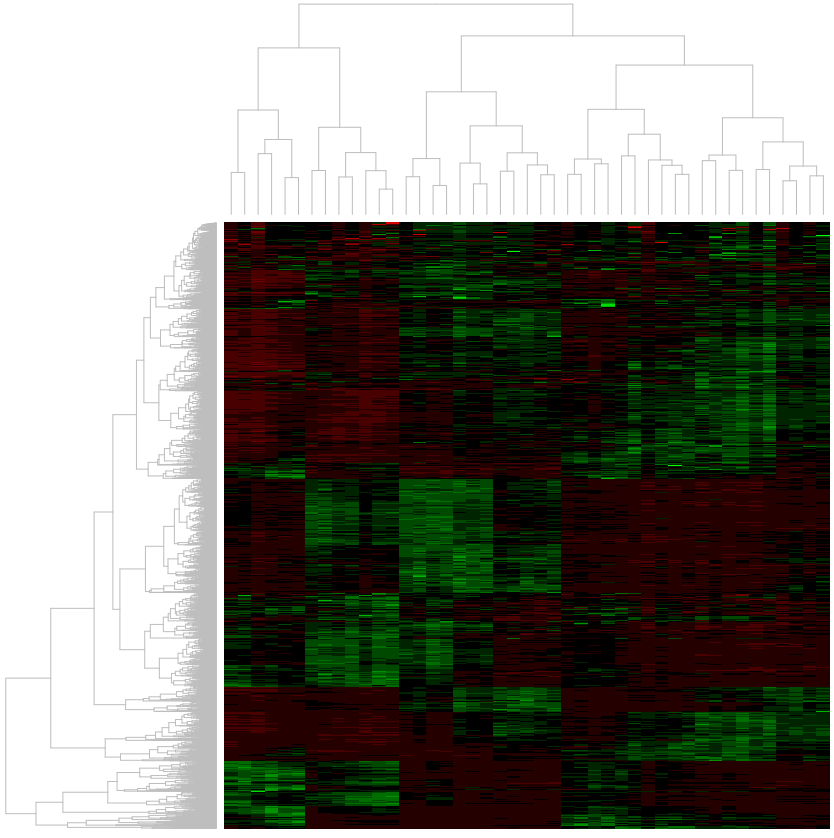
Correlation pattern between SNPs

Goal in GWAS: : find loci associated with the disease

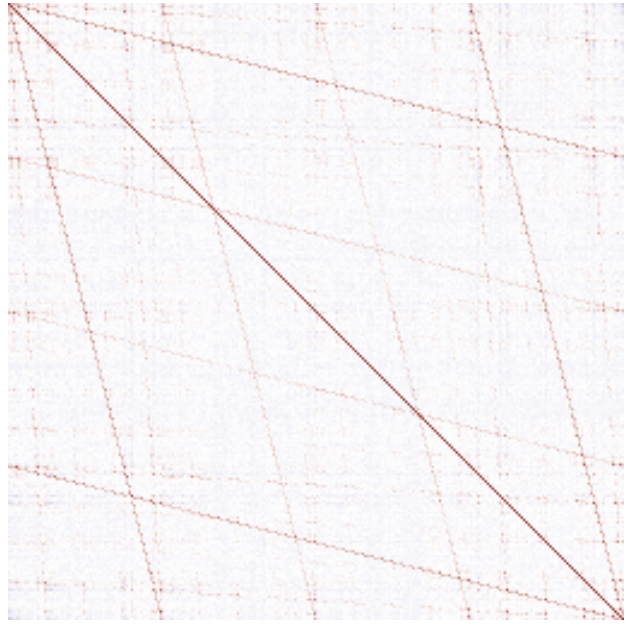
Task: feature selection, prediction of multiple outputs

Gene expression of *P. falciparum* (malaria), transcriptome

Affymetrix GeneChip array, thousands of genes, tens of conditions



Clustering of gene expression
(row: genes; columns: conditions)



Correlation pattern between 4-size counts in gene promotor regions

Many possible goals. . .

Task: clustering, multivariate analysis, prediction, feature selection

Data characteristics

Most striking features of genomics data

1. More variables than individuals
   The $n \leq p$ paradigm, or *"high-dimensional setting"*
2. Highly structured data
   Because the underlying system is well organized: there is hope!

"Secondary" features

3. Databases may be large
   - but can mostly be loaded into RAM of a usual workstation if smartly encoded and preprocessed (thanks to bioinformatics)
4. Multiple sources of heterogeneity
   - heterogeneity between samples, technologies, data-type...

Statistical learning for genomics

Goals remain the same

- supervised learning
  prediction, classification, ...
- unsupervised learning
  clustering, feature extraction, ...

Traditional methods are not tailored for them

1. Computational issues
2. Statistical issues
3. Modeling/interpretability issues

Basically because of high-dimensional feature spaces

Computational issues

Nesterov's classification (2012)

class of problem	dimension (# features p)	conceivable operations	computational cost	memory requirement	example in omics	expected task
small	$10^0 \sim 10^2$	All	$p^3 \sim p^4$	10^3 (Kb)	–	–
medium	$10^3 \sim 10^4$	A^{-1}	$p^2 \sim p^3$	10^6 (Mb)	transcriptomics	network inference
large	$10^5 \sim 10^7$	Ax	$p \sim p^2$	10^9 (Gb)	association studies	variable selection
huge	$10^8 \sim 10^{12}$	$x + y$	$\log(p) \sim p$	10^{12} (Tb)	metagenomics	clustering

Table: Typical matrix algebra operations with their computational cost and memory requirement for various problem scales

Comments

- Some purely algorithmic methods are out of reach in certain settings
  e.g., agglomerative clustering in $\mathcal{O}(n^3)$ *or* $\mathcal{O}(n^2)$
- Methods minimizing a criterion should adapt to the dimension

medium 2nd-order methods (accurate in few iterations)
large 1st-order methods (more iterations required)

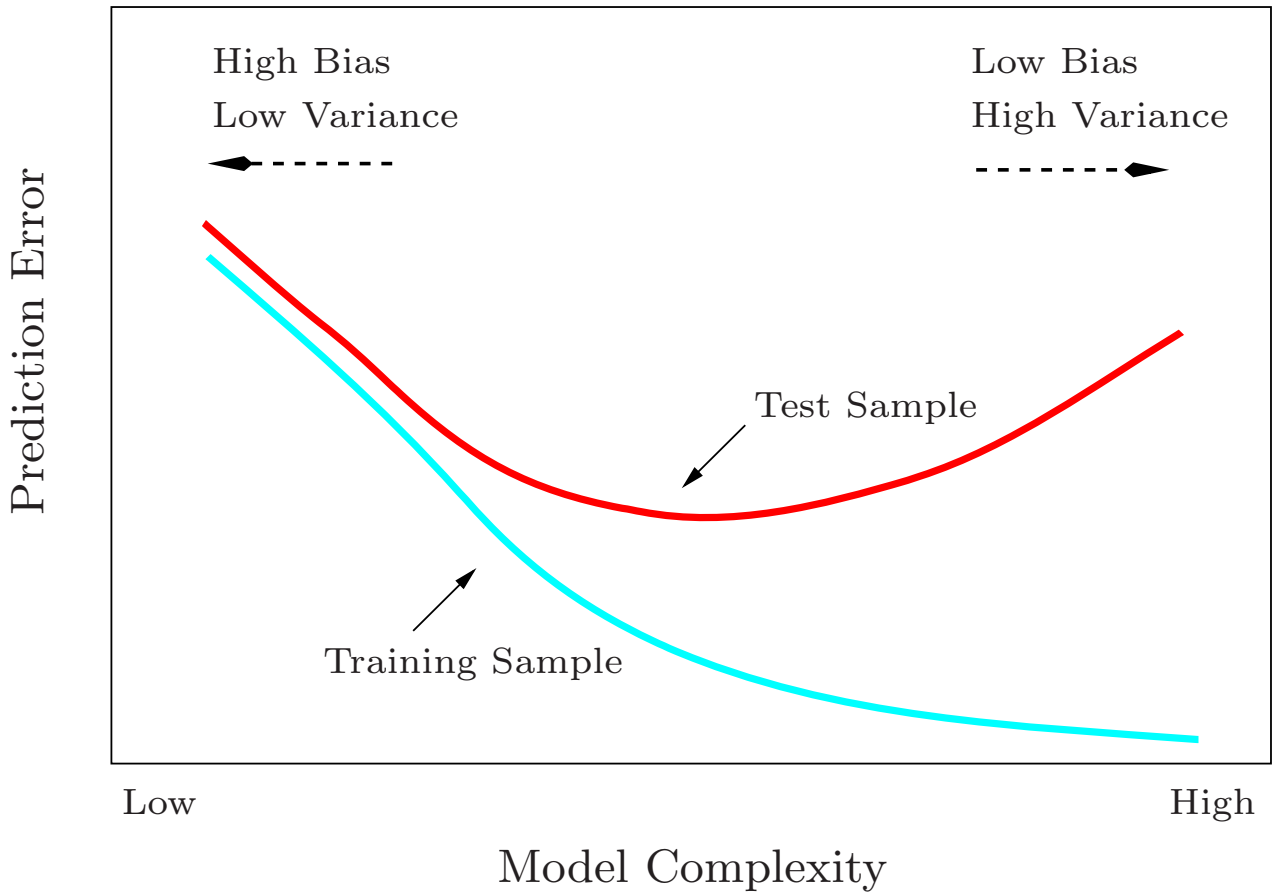
$\rightarrow$ tradeoff between speed and accuracy

Statistical issues, e.g., overfitting (I)

Model complexity in high-dimensional spaces

A well-known phenomenon affecting too complex models

- low bias and large variance $\Rightarrow$ poor capability for generalization
- worse in HD spaces (separating noise from signal is challenging)



Statistical issues: overfitting (II)

Model complexity in high-dimensional spaces

Consider a data set $\{(x_i, y_i)\}_{i=1,\dots,n}$ with

- $x_i \sim \mathcal{U}([-\pi, \pi])$
- $y_i = \sin(2x_i) + \mathcal{N}(0, \sigma^2)$, with σ chosen so that $R^2 \approx 0.8$.

Polynomial linear regression

We use p to control the dimension of the feature space, i.e., the model complexity.

$$y_i = \theta_0 + \sum_{j=1}^{p} x_i^j \theta_j + \varepsilon_i, \qquad \varepsilon_i \sim \mathcal{N}(0, \sigma^2).$$

We vary

- the model complexity $p \in \{1, \dots, 50\}$,
- the size of the training set $n \in \{10, 50, 200\}$.

Statistical issues: overfitting (III)

Model complexity in high-dimensional spaces

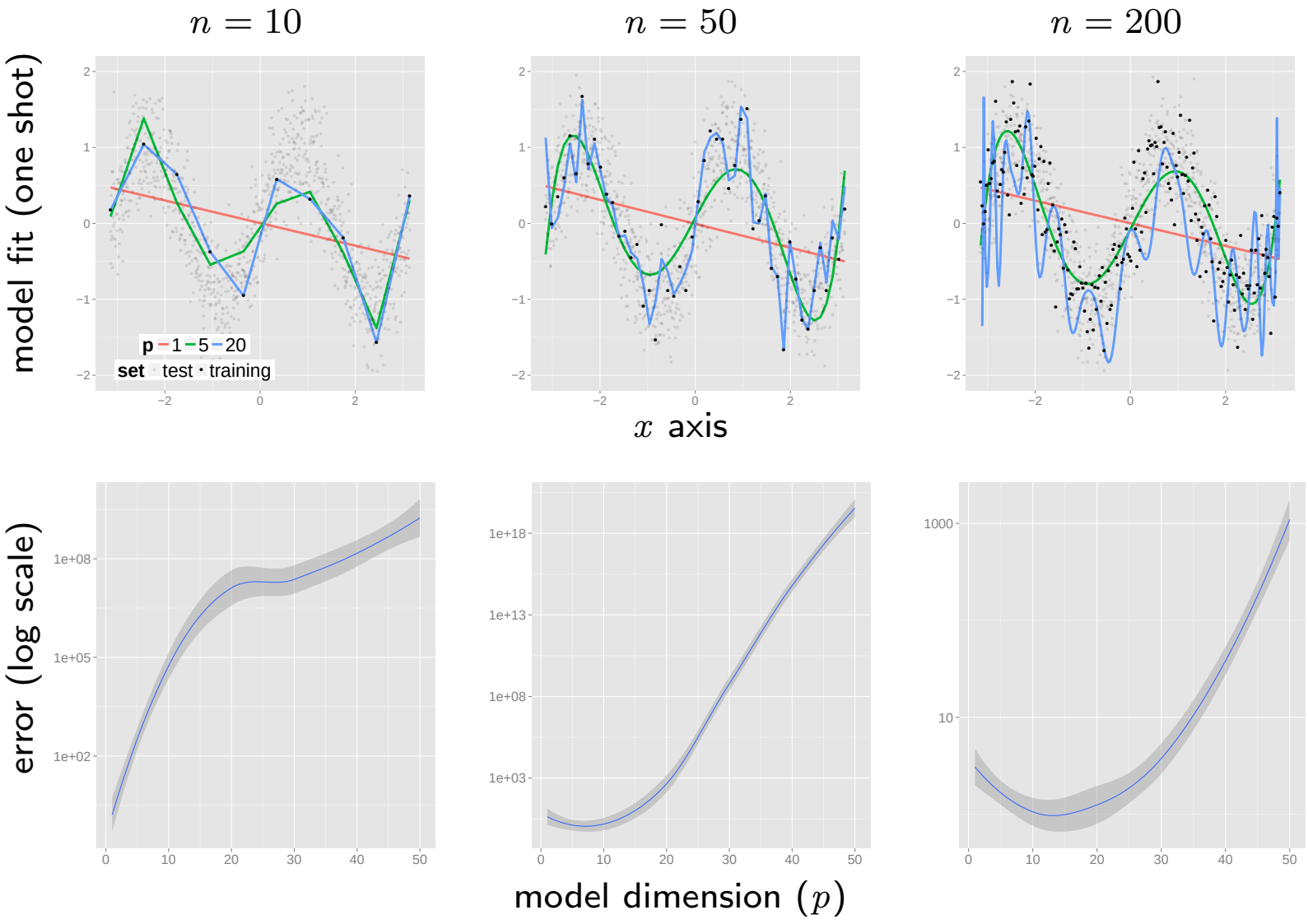
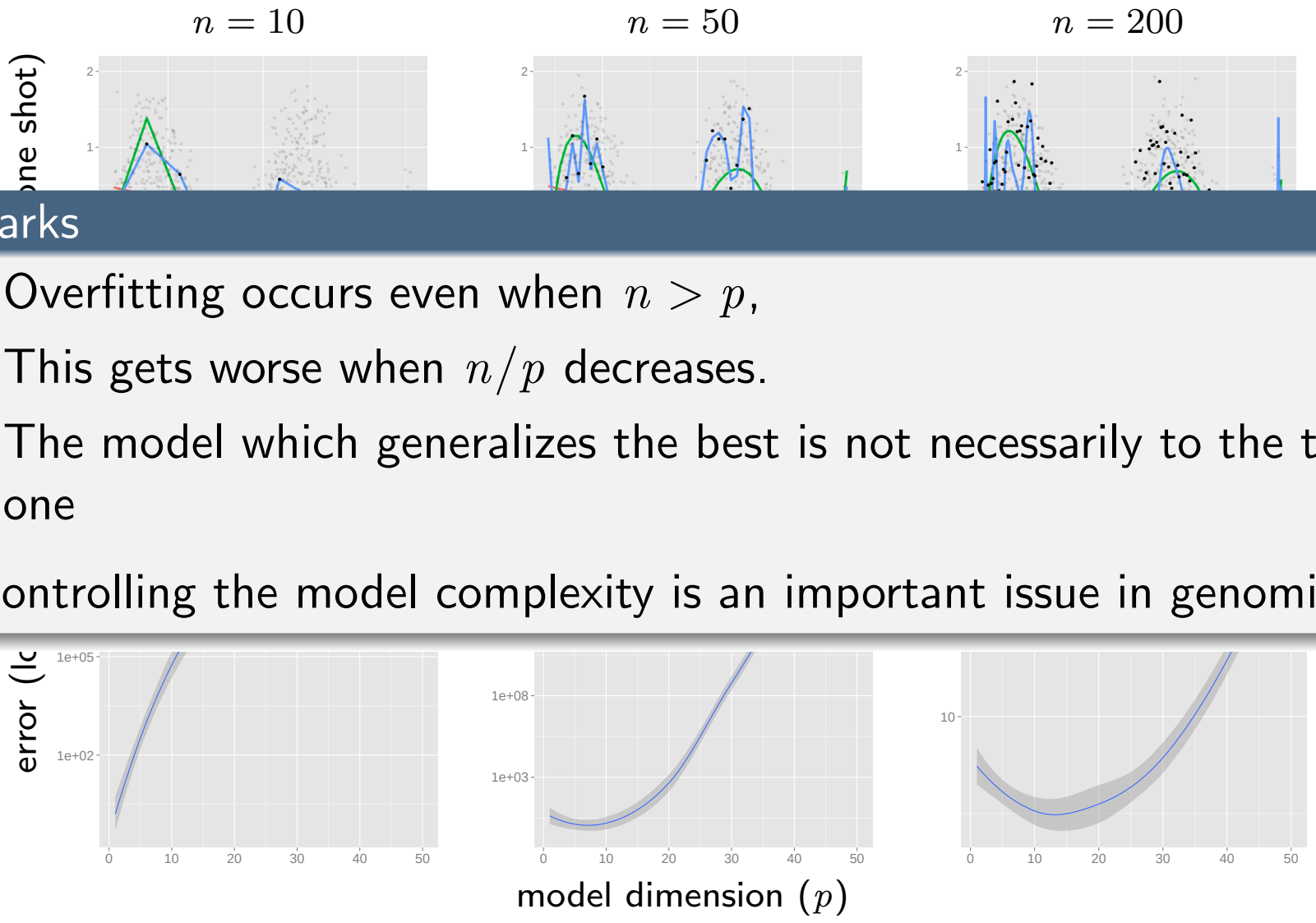


Figure: Overfitting is especially at play with high-dimensional data.

Statistical issues: overfitting (III)

Model complexity in high-dimensional spaces



Remarks

- Overfitting occurs even when $n > p$,
- This gets worse when n/p decreases.
- The model which generalizes the best is not necessarily to the true one

$\rightsquigarrow$ Controlling the model complexity is an important issue in genomics.

Figure: Overfitting is especially at play with high-dimensional data.

General strategy

Revisit "traditional" statistical methods under the light of optimization

1. statistical problem $\leftrightarrow$ optimization problem

   outcome $\mathbf{y}$, predictors $\mathbf{X}$, set fo parameters $\boldsymbol{\theta}$ (vector, matrix)

   - Linear model fitted by OLS

$$\hat{\boldsymbol{\theta}} = \arg\min \|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|_2^2 .$$

   - Principal Component Analysis

$$\left\{\hat{\boldsymbol{\theta}}, \hat{\mathbf{T}}\right\} = \arg\min \|\mathbf{X} - \mathbf{T}\boldsymbol{\theta}'\|_F^2, \quad \text{s.t} \quad \boldsymbol{\theta}'\boldsymbol{\theta} = \mathbf{I}.$$

   - Hierarchical clustering

$$\hat{\boldsymbol{\theta}} = \arg\min \|\mathbf{X} - \boldsymbol{\theta}\|_F^2, \quad \text{s.t} \sum_{i>j} \mathbf{1}_{\{\boldsymbol{\theta}_j \neq \boldsymbol{\theta}_i\}} < c.$$

   - More generally,

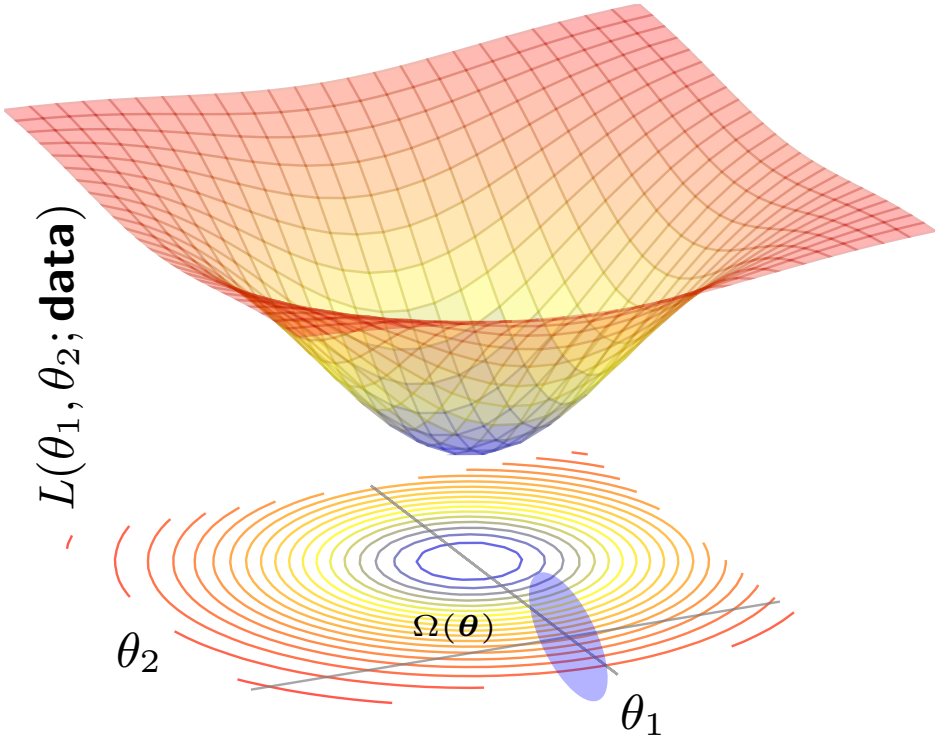
$$\underset{\boldsymbol{\theta}}{\text{minimize}}\, L(\boldsymbol{\theta}; \textbf{data}) \quad \text{s.t.} \quad \Omega(\boldsymbol{\theta}) \leq c.$$

2. modification of the original problem/regularization

General strategy

Revisit "traditional" statistical methods under the light of optimization

1. statistical problem $\leftrightarrow$ optimization problem
2. modification of the original problem/regularization



modify Ω and/or L to

- control the computational cost
- control the model complexity
- account for prior knowledge

looking for

- $\rightsquigarrow$ $\uparrow$ performance and interpretability
- $\rightsquigarrow$ trade-off between speed and accuracy

Example: Supervised Learning (1)

Regularized empirical risk

Let $\{(\mathbf{x}_i, y_i), i = 1, \ldots, n\}$ be some i.i.d. observations where we would like to predict $y_i \in \mathbb{R}$ from p regressors $\mathbf{x}_i \in \mathbb{R}^p$ with a linear function of the predictors $\boldsymbol{\theta}^\intercal \phi(\mathbf{x}_i)$

Solve an optimization problem

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta} \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^{n} \ell\left(y_i, \boldsymbol{\theta}^\intercal \phi(\mathbf{x}_i)\right) + \lambda \Omega(\boldsymbol{\theta}),$$

where

- ℓ is some loss function
- λ controls the model complexity
- equivalent if everything is convex by Lagrangian duality

Example: Supervised Learning (1)

Regularized empirical risk

Let $\{(\mathbf{x}_i, y_i), i = 1, \ldots, n\}$ be some i.i.d. observations where we would like to predict $y_i \in \mathbb{R}$ from p regressors $\mathbf{x}_i \in \mathbb{R}^p$ with a linear function of the predictors $\boldsymbol{\theta}^\intercal \phi(\mathbf{x}_i)$

Solve an optimization problem

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta} \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^{n} \ell\left(y_i, \boldsymbol{\theta}^\intercal \phi(\mathbf{x}_i)\right), \quad \text{s.t.} \quad \Omega(\boldsymbol{\theta}) \leq c.$$

where

- ℓ is some loss function
- λ controls the model complexity
- equivalent if everything is convex by Lagrangian duality

Example: Supervised Learning (2)

Some classical loss functions in classification

For $y \in \{-1, 1\}$, consider an estimate $\hat{y} = \hat{\boldsymbol{\theta}}^\mathsf{T} \phi(\mathbf{x})$.
A natural loss is the binary loss

$$\ell(y, \hat{y}) = \mathbf{1}_{\{y\hat{y} \leq 0\}}.$$

Some convex surrogates

- the quadratic loss: $\ell(y, \hat{y}) = (y - \hat{y})^2$,
- the hinge loss: $\ell(y, \hat{y}) = \max\{0, 1 - y\hat{y}\}$,
- the logistic loss: $\ell(y, \hat{y}) = \log(1 + \exp(-y\hat{y}))$.

Example: Supervised Learning (2)

Some classical loss functions in classification

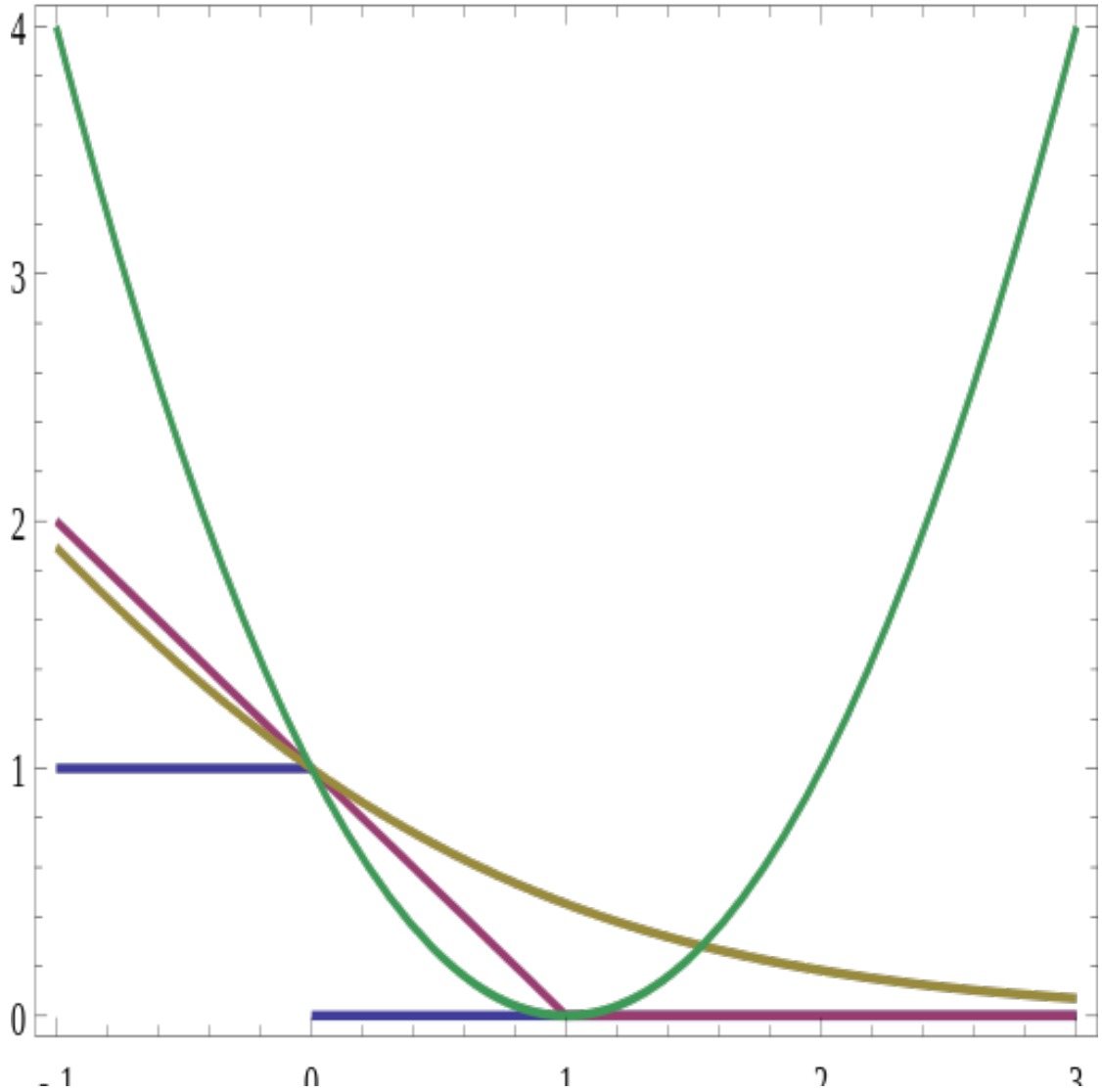


Figure: Usual loss functions in classification

Example: Supervised Learning (3)

Some classical regularizers

One usually controls the complexity of a model by controling the number of parameters, for instance in model selection, the BIC is defined by

$$\text{BIC} = -2\text{loglik}\left(\hat{\boldsymbol{\theta}}; \{(y_i, \mathbf{x}_i; i = 1, \ldots, n\}\right) + \log(n)\|\hat{\boldsymbol{\theta}}\|_0.$$

Other convex regularizers, used as surrogates

- the ℓ_2 – or Euclidean– norm, $\|\boldsymbol{\theta}\|^2$
  $\rightsquigarrow$ Controls the size of the parameters (ridge regression)

- the ℓ_1 norm, $\|\boldsymbol{\theta}\|_1$
  $\rightsquigarrow$ is the smallest convex surrogate to $\|\cdot\|_0$ (Lasso)

Central idea: convexity

Large-scale data

- suggest simple models
- fitting procedure with low complexity

When is a problem "easy"?

- classical view: opposes Linear to Nonlinear problems
- modern view: the correct distinction is Convex to Nonconvex

Other advantage of convexity

- a local optimum is a global optimum
- convexifying a problem introduce regularization, and maybe more interpretability

Main objective of this course

This course

Humbly introduces basic algorithms to solve unconstraint (non-smooth) convex problems in order to compute $\hat{\boldsymbol{\theta}}$ for problem of the form.

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta} \in \mathbb{R}^p}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} \ell\left(y, \boldsymbol{\theta}^\intercal \phi(\mathbf{x}_i)\right) + \lambda \Omega(\boldsymbol{\theta}).$$

Complementary approaches

1. Christophe Giraud's course
   How to take advantage of convexity in statistics to derive properties of regularized estimators

2. Francis Bach's course
   How to take advantage of convexity in optimization to derive properties of regularized estimators