Sparsity and dimension reduction
(Statistics and Machine Learning)

Ghislain DURIF

CNRS, IMAG, Université de Montpellier, France
ghislain.durif@umontpellier.fr
https://gdurif.perso.math.cnrs.fr/

April 4th 2019
Bordeaux Data Science

This presentation and its content are available under the license CC-BY-NC-SA 2.0 FR (except for third-party images/contents for which license is set by their respective authors).
Outline

Introduction

Sparsity and regression/classification

Dimension reduction and sparse PLS regression

Dimension reduction and probabilistic matrix factorization

To conclude
$x_{ij} = \text{recording of variable (feature) } j \text{ in individual (observation) } i$

$X_{n \times p} = \begin{bmatrix}
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
1 & \cdots & \cdots & \cdots & \cdots & \cdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\end{bmatrix}
$
Sparsity

Sparse matrix

- lots of null values

“Sparse methods”

- Analyze sparse data
- Enforce sparsity in underlying objects (e.g. parameters)
Sparse data

Hapmap data (88 CEU, 100 YRI, 46 MEX), \(~ 50 \times 10^3\) SNPs and \(~ 80\%\) of zeros

Human genome

- length: \(2.7 \times 10^9\) bp
- mutations: \(1.5 \times 10^6\) SNPs

0.5\% of non-null values (Sachidanandam et al., 2001)
Sparse data

Single-cell gene expression datasets: 40 – 90% of zeros (see examples in Durif et al., 2019)
 Sparse data

User reviews/ratings on products

$r_{ij} = \text{rating/review given by user } i \text{ to product } j$
Methods with underlying sparsity

Hypothesis of parsimony

▶ Some variables are uninformative = noise to be removed

Examples

▶ Enforce sparsity on the parameters of a statistical model
▶ Learn a sparse representation of the data
▶ Select relevant variables and remove non pertinent ones

Why?

▶ Analysis interpretation
▶ Dimension reduction
High-dimensional data

\[ X_{n \times p} = \begin{bmatrix}
1 & \ldots & \ldots & \ldots & \ldots & p \\
\vdots & \ddots & & & & \\
1 & \ldots & \ldots & \ldots & \ldots & \ldots & p
\end{bmatrix} \begin{bmatrix}
1 \\
\vdots \\
n
\end{bmatrix} \text{observations}

Variables

- High dimension: 
  \( n \) grows but \( \ll p \)
  \( \rightarrow n \sim 100/1000 \) 
  and \( p \sim 10000+ \)

- \( \neq \) large scale data: \( n \) is huge \( (\sim 10^6/10^9) \)
Issues with high dimensional data ($p \gg n$)

“The curse of high-dimensionality” (Donoho, 2000)

- **Geometry**: counter-intuitive behavior of standard metrics
  - Representation: how to visualize thousands of variables?

- **Optimization**: numerical singularities due to spurious dependencies (colinearity)

- **Computational efficiency and scalability**
Issues with high dimensional data \((p \gg n)\)

“The curse of high-dimensionality” (Donoho, 2000)

- **Geometry**: counter-intuitive behavior of standard metrics
  - Representation: how to visualize thousands of variables?

- **Optimization**: numerical singularities due to spurious dependencies (colinearity)

- **Computational efficiency and scalability**

→ Dimension reduction approaches
Dimension reduction

Statistical challenges

- Data exploration (visualization / clustering)
- Regression/Classification (prediction)

- Representation of the data in a lower dimensional subspace
- Consider sparsity: select the variables that contribute to this representation
Dimension reduction

Regression/Classification
(prediction)

$\downarrow$

Sparse PLS regression

Data exploration
(visualization / clustering)

$\downarrow$

Probabilistic PCA and matrix factorization
Outline

Introduction

Sparsity and regression/classification

Dimension reduction and sparse PLS regression

Dimension reduction and probabilistic matrix factorization

To conclude
Regression

*n* observations

- predictors/covariates $x_i \in \mathbb{R}^p$
- continuous response $y_i \in \mathbb{R}$

\[
X_{n \times p} = \begin{bmatrix}
\vdots
\end{bmatrix}
\quad \text{and} \quad
Y = \begin{bmatrix}
\vdots \\
y_i \\
\vdots
\end{bmatrix} \in \mathbb{R}^n
\]

**Purpose**

- Model $y = f(x)$ with $f : \mathbb{R}^p \rightarrow \mathbb{R}$
- Learn $f$ by using the sample $(y_i, x_i)_{i=1,...,n}$
- Predict $y \in \mathbb{R}$ with $f(x)$ for any new $x \in \mathbb{R}^p$

**Over-fitting vs ability to generalize?**
Linear model

\[ y_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ij} + \varepsilon_i \]

- \( \beta_0 = \) intercept
- \( \beta_1, \ldots, \beta_p = \) linear coefficients → parameters of the model
- \( \varepsilon_i = \) error term

Source: wikipedia.org
Linear model

\[ y_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ij} + \varepsilon_i \]

**Note:** We can “forget” about \( \beta_0 \) (add a column of 1 to \( X \) or center columns of \( X \))

Source: wikipedia.org
Linear model

\[ y_i = \sum_{j=1}^{p} \beta_j x_{ij} + \varepsilon_i \]

Least square regression

\[ \text{argmin}_{\beta_1, \ldots, \beta_p} \sum_{i=1}^{n} \left( y_i - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \]

Learn \( \beta_1, \ldots, \beta_p \)

Source: wikipedia.org
Linear model

\[ y_i = \sum_{j=1}^{p} \beta_j x_{ij} + \varepsilon_i \]

Vectorial notation

\[ y_i = \mathbf{x}_i^T \mathbf{\beta} + \varepsilon_i \]

with \( \mathbf{\beta} = (\beta_1, \ldots, \beta_p)^T \in \mathbb{R}^p \)

Matrix notation

\[
\begin{align*}
\mathbf{Y} & = \mathbf{X} \mathbf{\beta} + \mathbf{E} \\
& \quad \text{n×1} \quad \text{n×p} \quad \text{p×1} \quad \text{n×1}
\end{align*}
\]
Linear model

Matrix notation

\[ Y = X \beta + E \]

\( n \times 1 \quad n \times p \quad p \times 1 \quad n \times 1 \)

Least Squares objective

\[ \hat{\beta} = \arg\min_{\beta \in \mathbb{R}^p} \| Y - X\beta \|_2^2 \]
Penalized least squares

Solution of linear regression:

\[ \beta = (X^T X)^{-1} X^T Y \]

Issue when \( p > n \)

- **Ridge regression** \( (\lambda \geq 0) \)

\[
\text{argmin}_{\beta \in \mathbb{R}^p} \| Y - X\beta \|_2^2 + \lambda \| \beta \|_2^2
\]

\[ \beta = (X^T X + \lambda \text{Id})^{-1} X^T Y \]

Sparsity ?

- \( \ell_1 \) penalty \( \rightarrow \) **Lasso**

\[ ^1 \ell_2 \text{ norm: } \| \beta \|_2^2 = \sum_{j=1}^{p} |\beta_j|^2 \]

\[ ^2 \ell_1 \text{ norm: } \| \beta \|_1 = \sum_{j=1}^{p} |\beta_j| \]
Lasso (Tibshirani, 1996)

\[
\arg\min_{\beta_1, \ldots, \beta_p} \sum_{i=1}^{n} \left( y_i - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \quad \text{wit } \lambda \geq 0
\]
Lasso (Tibshirani, 1996)

\[
\arg\min_{\beta \in \mathbb{R}^p} \| Y - X\beta \|_2^2 + \lambda \| \beta \|_1
\]

Sparsity ?

Source: wikipedia.org
Lasso (Tibshirani, 1996)

Source: towardsdatascience.com
Sparsity in linear regression

Coefficients $\beta$ and $\hat{\beta}$ ($n = 50$, $p = 100$)

Choice of $\lambda$:

- cross-validation
- stability selection (Meinshausen and Bühlmann, 2010)
Elastic Net (Zou and Hastie, 2004)
Fused Lasso (Tibshirani et al., 2005)
Adaptive Lasso (Zou, 2006)
Group Lasso (Yuan and Lin, 2006)
Lasso and derivatives

- Elastic Net (Zou and Hastie, 2004)

\[
\underset{\beta \in \mathbb{R}^p}{\text{argmin}} \| Y - X\beta \|^2_2 + \lambda_1 \sum_{j=1}^p |\beta_j| + \lambda_2 \sum_{j=1}^p |\beta_j|^2
\]

- Fused Lasso (Tibshirani et al., 2005)

- Adaptive Lasso (Zou, 2006)

- Group Lasso (Yuan and Lin, 2006)
Lasso and derivatives

- Elastic Net (Zou and Hastie, 2004)
- Fused Lasso (Tibshirani et al., 2005)

\[
\arg\min_{\beta \in \mathbb{R}^p} \| Y - X\beta \|_2^2 + \lambda_1 \sum_{j=1}^{p} |\beta_j| + \lambda_2 \sum_{j=2}^{p} |\beta_j - \beta_{j-1}|
\]

- Adaptive Lasso (Zou, 2006)
- Group Lasso (Yuan and Lin, 2006)
Lasso and derivatives

- Elastic Net (Zou and Hastie, 2004)
- Fused Lasso (Tibshirani et al., 2005)
- Adaptive Lasso (Zou, 2006)

\[
\arg\min_{\beta \in \mathbb{R}^p} \| Y - X\beta \|^2_2 + \sum_{j=1}^{p} \lambda_j |\beta_j|
\]

- Group Lasso (Yuan and Lin, 2006)
Lasso and derivatives

- Elastic Net (Zou and Hastie, 2004)
- Fused Lasso (Tibshirani et al., 2005)
- Adaptive Lasso (Zou, 2006)
- Group Lasso (Yuan and Lin, 2006)

\[
\arg\min_{\beta \in \mathbb{R}^p} \| Y - X\beta \|_2^2 + \sum_{g=1}^{N} \lambda_g \| \beta_{G_g} \|_2
\]

where \( G_1, \ldots, G_N \) is a partition of \( \{1, \ldots, p\} \)
and \( \beta_{G_g} = (\beta_j)_{j \in G_g} \)
Classification

$n$ observations

- predictors/covariates $x_i \in \mathbb{R}^p$
- discrete response $y_i \in \{0, 1\}$

$$X_{n \times p} = \begin{bmatrix} x_{ij} \end{bmatrix}$$

and

$$Y = \begin{bmatrix} y_i \end{bmatrix} \in \{0, 1\}^n$$

Generalized linear model

$$\mathbb{E}[Y] = f(x^T \beta) \text{ with } f : \mathbb{R} \to \{0, 1\}$$
Logistic regression

Logistic model:

\[ Y_i \mid x_i \sim \mathcal{B}(\pi_i) \]
\[
\left\{ \begin{array}{l}
\Pr(Y_i = 1) = \pi_i \\
\Pr(Y_i = 0) = 1 - \pi_i 
\end{array} \right.
\]

\[ \pi_i = \mathbb{E}[Y_i \mid x_i] = \logit^{-1}(x_i^T \beta) \quad \text{with} \quad \logit(\pi) = \log \left( \frac{\pi}{1 - \pi} \right) \]

Optimization criterion (log-likelihood)

\[ \arg\min_{\beta \in \mathbb{R}^p} \sum_{i=1}^{n} \left( y_i \eta_i - \log \left( 1 + \exp(\eta_i) \right) \right) + \text{Pen}_\lambda(\beta) \]

with \( \eta_i = x_i^T \beta \) and \( \text{Pen}_\lambda(\cdot) \) a penalty function
## Libraries for penalized regression

<table>
<thead>
<tr>
<th></th>
<th>SPAMS(^1)</th>
<th>GLMNET(^2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Python</td>
<td>✓</td>
<td>×</td>
</tr>
<tr>
<td>R</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Matlab</td>
<td>✓</td>
<td>×</td>
</tr>
<tr>
<td>Lin. &amp; log. model</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Ridge</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Lasso</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Elastic Net</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Fused Lasso</td>
<td>✓</td>
<td>×</td>
</tr>
<tr>
<td>Group Lasso</td>
<td>✓</td>
<td>×</td>
</tr>
</tbody>
</table>

\(^1\) [http://spams-devel.gforge.inria.fr/](http://spams-devel.gforge.inria.fr/)
\(^2\) [https://cran.r-project.org/package=glmnet](https://cran.r-project.org/package=glmnet)
Outline

Introduction

Sparsity and regression/classification

Dimension reduction and sparse PLS regression

Dimension reduction and probabilistic matrix factorization

To conclude
Latent space projection

Projection of $X$ onto a lower dimensional space (of dim. $K$)

$$t_{ik} = \sum_{j=1}^{p} x_{ij} w_{jk}$$

$$t_k = X w_k \quad \text{with} \quad \begin{cases} t_k \in \mathbb{R}^n \\ w_j \in \mathbb{R}^p \end{cases}$$
Variable selection

\[
\begin{bmatrix}
X_{n \times p}
\end{bmatrix}
\times
\begin{bmatrix}
0 \\
0 \\
0 \\
0
\end{bmatrix}
=
\begin{bmatrix}
\end{bmatrix}
\]

- **Enforce sparsity:** only a few variables contribute to the model
- **Objective:** drop non relevant variables from the model
Dimension reduction

\( T_{n \times K} = \) representation of individuals

\[ T_{n \times K} = \text{representation of individuals} \]

Dimension \( K \ll \min(n, p) \)
(Sparse) PLS regression

- Find new components $\mathbf{T}$ that summarize information within $\mathbf{X}$ and explain a response $\mathbf{Y}$

→ **Partial Least Squares regression** (review in Wold et al., 2001)

Latent space projection

Sparse Partial Least Squares

Variable selection

(Lê Cao et al., 2008; Chun and Keleş, 2010)
Partial Least Squares (PLS) regression

\( X_c = \text{centered version of } X \) \hspace{1cm} \( Y_c = \text{centered version of } Y \)

<table>
<thead>
<tr>
<th>PCA</th>
<th>PLS(^1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Components</td>
<td>( t_k = X_c w_k \in \mathbb{R}^n )</td>
</tr>
<tr>
<td>Weights</td>
<td>( w_k \in \mathbb{R}^p )</td>
</tr>
<tr>
<td>Objective</td>
<td>( \arg\max_{w \in \mathbb{R}^p} \Var(Xw) )</td>
</tr>
<tr>
<td></td>
<td>( \arg\max_{w \in \mathbb{R}^p} w^T X_c^T X_c w )</td>
</tr>
</tbody>
</table>

Additional constraints:

\( \| w_k \|_2 = 1 \) for \( k = 1, \ldots, K \)

\( \text{orthogonality between } t_1, \ldots, t_K \)

\(^1\) with univariate response
Partial Least Squares (PLS) regression

- Original model:
  \[ Y_c = X_c \beta + E \]
  \[ n \times 1 \quad n \times p \quad p \times 1 \quad n \times 1 \]
  \( p \gg n \)

- Linear model in lower dimension space:
  \[ Y_c = T q + \tilde{E} \]
  \[ n \times 1 \quad n \times K \quad K \times 1 \quad n \times 1 \]
  \( K \ll p \)

- Plug the relation \( T = X_c W \) in the previous model

  \[ \hat{\beta} = W \hat{q} \]

  \( \rightarrow \) Estimate \( \beta \in \mathbb{R}^p \) thanks to the estimation of \( q \in \mathbb{R}^K \):
Sparse PLS

Penalized covariance maximization

\[
\arg\min_{w \in \mathbb{R}^p} \left\{ - \text{Cov}(Xw, Y) + \lambda_S \|w\|_1 \right\} \\
\|w\|_2 = 1
\]

\[\lambda_S \geq 0 \text{ penalty parameter}\]
Adaptive penalty (Durif et al., 2018)

\[
\text{argmin}_{\mathbf{w} \in \mathbb{R}^p} \left\{ -\mathbf{w}^T \mathbf{X}_c^T \mathbf{Y}_c + \sum_j \lambda_j |w_j| \right\}
\]

with \(\lambda_j \geq 0\) for \(j = 1, \ldots, p\)

→ Penalize more the less significant predictors
**Visualization**

$t_1$ vs $t_2$ (individual representation)

$n = 50, \ p = 100, \ \#\{\beta_j \neq 0\} = 20$
Visualization

$w_1 \text{ vs } w_2$ (variable representation)

$n = 50, \ p = 100, \ \#\{\beta_j \neq 0\} = 20$
Sparse PLS for classification

- (sparse) PLS for discriminant analysis
- PLS for logistic regression

Our approach: logit-SPLS\textsuperscript{1} (Durif et al., 2018)

- Combination of logistic regression and sparse PLS for classification
  \[\text{estimate } \beta \text{ in } \mathbb{E}[Y] = f(x^T \beta)\]

\textsuperscript{1}https://doi.org/10.1093/bioinformatics/btx571
https://cran.r-project.org/package=plsgenomics
Data visualization

$t_1$ vs $t_2$ (individual representation)

$n = 50, \ p = 100, \ \#\{\beta_j \neq 0\} = 20$
Data visualization

$w_1$ vs $w_2$ (variable representation)

$n = 50$, $p = 100$, $\#\{\beta_j \neq 0\} = 20$
R package plsgenomics\textsuperscript{1}

- (sparse) PLS regression
- (sparse) PLS for classification (logistic regression)

\textsuperscript{1}https://cran.r-project.org/package=plsgenomics
Single T cell expression profile

- Expression profiles (~ 20000 genes) of individual T cells
- Sampling during and after an immune response to a yellow fever vaccine shot

<table>
<thead>
<tr>
<th>Day</th>
<th># cells</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>591</td>
</tr>
<tr>
<td>136</td>
<td>572</td>
</tr>
<tr>
<td>908</td>
<td>192</td>
</tr>
</tbody>
</table>
Groups and sub-types of T cells:

i) “Effector” (TEMRA and EM)

ii) “Memory” (CM, TSCM)

Cells were manually identified based on the measure of two surface markers (CCR7 and CD45RA)

Problem: a huge proportion of unidentified T cells
Multi-group classification

Prediction of the cell type with multinomial sparse PLS

1) First round: prediction based on 22 surface markers + the expression of the 22 associated genes

2) Differential Expression Analysis based on the predicted cell types

3) Second round: prediction based on 22 surface markers + the expression of the 22 associated genes + 61 differentially expressed genes

1“One-class versus a reference” multi-group classification
https://doi.org/10.1093.bioinformatics/btx571
https://cran.r-project.org/package=plsgenomics
Cell type prediction

Multinomial SPLS on the train set

- Latent space projection
Cell type prediction

Multinomial SPLS on the full data set after prediction

- Latent space projection
Cell type prediction

Estimated cell type membership probabilities

Effector probability distribution through time

Day
- D15
- D136
- D908

p_EM
- 1.0
- 0.75
- 0.50
- 0.25
Introduction

Sparsity and regression/classification

Dimension reduction and sparse PLS regression

Dimension reduction and probabilistic matrix factorization

To conclude
Matrix factorization: $X \approx UV^T$

Individuals: $U \in \mathbb{R}^{n \times K}$

Variables: $V \in \mathbb{R}^{p \times K}$

$\left\{ \text{Low dimensional representation} \right\}$

→ Low-rank representation of $X$
Matrix factorization: \( X \approx UV^T \)

Data visualization: scatter plot
\((u_{i1}, u_{i2})_{i=1:n}\)
Approximation $X \approx UV^T$?

Sense of the approximation?
Approximation $X \approx UV^T$?

Principal Component Analysis:

- Principal components = linear projection of $X$ with maximum variance
- SVD algorithm: $\arg\min_{U \in \mathbb{R}^{n \times K}, V \in \mathbb{R}^{p \times K}} \|X - UV^T\|_F^2$

→ Least squares approximation
Sparse matrix factorization

Penalization on $\ell_1$ norm (Lasso)

\[
\arg\min_{u \in \mathbb{R}^n, v \in \mathbb{R}^p} \left\{ \| X - uv^T \|_F^2 + \lambda \sum_{j=1}^{p} |v_j| \right\}
\]

$\rightarrow$ shrink contributions of non pertinent variables to zero
Embed PCA with a **probabilistic model**

- Replace $\| \cdot \|_2$ approximation by likelihood-based approaches
- $X_{ij} \sim$ probability distribution in the exponential family

→ Factorization of $\mathbb{E}[X]$ rather than $X$
Example: Poisson-Non-negative Matrix Factorization (NMF, Lee and Seung, 1999)

- \( X_{ij} \sim \mathcal{P}(\lambda_{ij}) \) with the Poisson rate matrix \( \Lambda = [\lambda_{ij}]_{n \times p} \)
- Factorization: \( \mathbb{E}[X] = \Lambda \simeq UV^T \)
1) Gamma-Poisson hierarchical model (Cemgil, 2009)

Factors = latent variables

Marginal distribution is over-dispersed:
Var(\(X_{ij}\)) > \(\mathbb{E}[X_{ij}]\)

\(\Gamma(\beta_{k,1}, \beta_{k,2})\)

\(\Gamma(\alpha_{k,1}, \alpha_{k,2})\)

\(\mathcal{P}(\sum_k U_{ik} V_{jk})\)

\(X_{ij}\)

\(V_{jk}\)

\(U_{ik}\)

---

\(^1\) Durif et al. (2019)
2) “Zero-inflated” Gamma-Poisson factor model

\[ \Gamma(\beta_{k,1}, \beta_{k,2}) \]

\[ \beta_{k,1} \quad \beta_{k,2} \quad \Gamma(\alpha_{k,1}, \alpha_{k,2}) \]

\[ V_{jk} \quad U_{ik} \quad D_{ij} \]

\[ \mathcal{B}(\pi_j^D) \]

\[ \pi_j^D \]

- \( D_{ij} = \) drop-out event indicator

- Sparse data

---

\(^1\)Durif et al. (2019)
Sparsity on V?

- Variable $j$ contributes to factor $k$ if $V_{jk} \neq 0$

- Objective: force the $V_{jk}$’s to be null for non pertinent genes

If $V$ was a parameter $\rightarrow \ell_1$ penalty

$$\underset{u \in \mathbb{R}^n, v \in \mathbb{R}^p}{\operatorname{argmin}} \left\{ \|X - uv^T\|_F^2 + \lambda \sum_{j=1}^{p} |v_j| \right\}$$
Probabilistic variable selection

$V_{jk}$ is a random variable $\rightarrow$ necessary to use sparsity-inducing priors

Spike and slab model:

- Continuous one-group prior: shrinkage to small value near zero (ex: Bayesian Lasso with Laplace prior)

- **Two-group prior**: mixture between a Dirac and a continuous distribution, true mass at zero
  $\rightarrow$ to induce a “sparse” posterior with a mass at zero
Sparse Gamma-Poisson model

\[ V_{jk} \sim (1 - \pi_j^s) \delta_0 + \pi_j^s \Gamma(\beta_{k,1}, \beta_{k,2}) \]

- **Gamma-Dirac mixture**
- \( \pi_j^s \in [0, 1] \) probability that gene \( j \) contributes to the model
- \( S_{jk} = \) sparsity indicator
Data visualization

$u_1$ vs $u_2$ (individual representation)
Data visualization

$v_1 \text{ vs } v_2$ (variable representation)
Data visualization

Single-cell expression data

Durif et al. (2019)
**Softwares**

**SPAMS**
- Dictionary learning
- Least squares decomposition $X \approx UV^T$
- Penalized decomposition

**pCMF**
- Model-based decomposition $X \approx UV^T$
- Sparse data
- Probabilistic induced sparsity

---

2. [https://github.com/gdurif/pCMF](https://github.com/gdurif/pCMF)
Outline

Introduction

Sparsity and regression/classification

Dimension reduction and sparse PLS regression

Dimension reduction and probabilistic matrix factorization

To conclude
Sparsity and dimension reduction?

- **Data exploration** (unsupervised): visualization / clustering
- **Regression, classification** (supervised): prediction
- Combination of latent representation and variable selection
- Probabilistic PCA
- Sparse PLS regression
- **Softwares**: pCMF, plsgenomics, SPAMS
Thanks for your attention

ghislain.durif@umontpellier.fr
https://gdurif.perso.math.cnrs.fr/

- Franck Picard (CNRS, LBBE)
- Sophie Lambert-Lacroix (Université Grenoble Alpes, TIMC)
- Laurent Modolo (CNRS, LBMC)
- Jeff Mold (Karolinska Institutet, Stockholm)
- Julien Mairal (Inria)
- Michael Blum (CNRS, TIMC)
Thanks for your attention

Funding:  
ANR “Investissement d’Avenir” project ABS4NGS  
ANR project MACARON  
ERC SOLARIS

pCMF  
https://doi.org/10.1093/bioinformatics/btz177  
https://arxiv.org/abs/1710.11028  
https://gitlab.inria.fr/gdurif/pCMF

Sparse PLS (plsgenomics)  
https://doi.org/10.1093/bioinformatics/btx571  
https://cran.r-project.org/package=plsgenomics


