



**HAL**  
open science

## Variable selection with Multi-Layer Group Lasso

Guillemette Marot, Quentin Grimonprez, Samuel Blanck, Alain Celisse

► **To cite this version:**

Guillemette Marot, Quentin Grimonprez, Samuel Blanck, Alain Celisse. Variable selection with Multi-Layer Group Lasso. useR! 2022, Jun 2022, Virtual, United States. hal-03942579

**HAL Id: hal-03942579**

**<https://inria.hal.science/hal-03942579v1>**

Submitted on 17 Jan 2023

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Variable selection with Multi-Layer Group Lasso

**Guillemette Marot**

Univ. Lille, CHU Lille METRICS & Inria MODAL

21 June 2022

Joint work with Quentin Grimonprez, Samuel Blanck and Alain Celisse

# Plan

- 1 Introduction
- 2 MLGL : multi-layer group lasso
- 3 Application and R usage
- 4 Conclusions - Discussion

# Introduction

Context :

Regression analysis in high dimension ( $n \ll p$ )

Notations :

- $y \in \mathbb{R}^n$  response variable
- $X \in \mathcal{M}_{n,p}(\mathbb{R})$  matrix containing the values of  $p$  explanatory variables for  $n$  individuals
- $\beta \in \mathbb{R}^p$  containing  $k$  non-zero elements.

Use of penalized regression techniques to deal with high dimension.

$$\hat{\beta}(\lambda) = \arg \min_{\beta \in \mathbb{R}^p} \{C(\beta) + \lambda \text{pen}(\beta)\}$$

with for example  $C(\beta) = \|y - X\beta\|_2^2 = \sum_{i=1}^n (y_i - \sum_{j=1}^p x_{ij}\beta_j)^2$

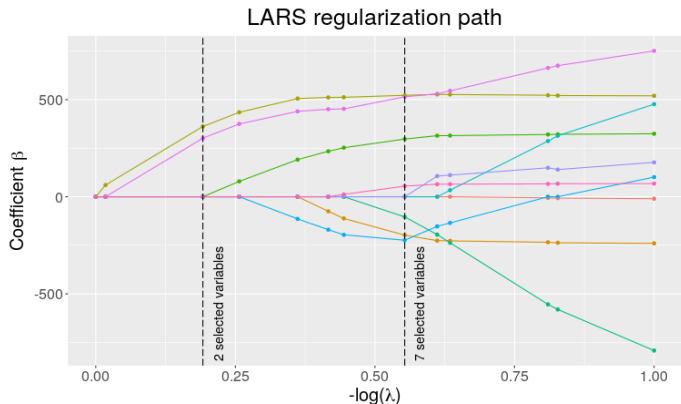
Focus on variable selection (not only prediction) to ease interpretability.

# Lasso (Tibshirani, 1996 ; Efron et al., 2004)

Lasso example in the high dimension context :

$$\hat{\beta}(\lambda) = \arg \min_{\beta \in \mathbb{R}^p} \{C(\beta) + \lambda \text{pen}(\beta)\}$$

with  $\text{pen}(\beta) = \|\beta\|_1 = \sum_{j=1}^p |\beta_j|$



# Redundancy

High dimension induces linear dependence between vectors corresponding to variables, thus inducing problems associated to redundancy or correlation (e.g. instability).

The performance of classical Lasso-based approaches strongly deteriorates as the redundancy strengthens (when  $p$  grows for a given  $n$ ).

Grouping variables in a first step and then selecting groups mitigates the instability default, but this usually requires either the calibration of additional parameters or knowledge of groups.

Objective of Multi-Layer Group-Lasso (MLGL) :

Select groups of correlated variables in high dimension, without knowing groups a priori.

# Plan

- 1 Introduction
- 2 MLGL : multi-layer group lasso**
- 3 Application and R usage
- 4 Conclusions - Discussion

# Multi-layer group lasso

## MLGL procedure :

- building a hierarchy (HCA : hierarchical cluster analysis)
- computing the paths of groups selected by group-Lasso with respect to  $\lambda$
- performing hierarchical multiple testing (HMT) to remove false positives for each  $\lambda$
- tuning  $\lambda$  to select the final groups of influential variables.

## Originality :

Exploitation of hierarchical structure of HCA and weighing in Group-lasso in order to reduce the complexity induced by the flexibility related to the possibility to **choose groups from different levels of HCA**.



## Multi-Layer Group-Lasso

Given  $\mathcal{G}_*$  the union of all the partitions at the different levels of the hierarchy  $\mathcal{G}_s$  of  $p$  variables in  $s$  groups ( $1 \leq s \leq p$ ), we define  $X^{\mathcal{G}_*} = \underbrace{[X, \dots, X]}_{p \text{ times}}$ . MLGL estimator is

$$\hat{\beta}_\lambda^{\mathcal{G}_*} = \operatorname{argmin}_{\beta \in \mathbb{R}^{p^2}} \left\{ \frac{1}{2} \|y - X^{\mathcal{G}_*} \beta\|_2^2 + \lambda \sum_{s=1}^p \rho_s \sum_{g=1}^s w_g^s \|\beta_{G_g^s}\|_2 \right\} \quad (1)$$

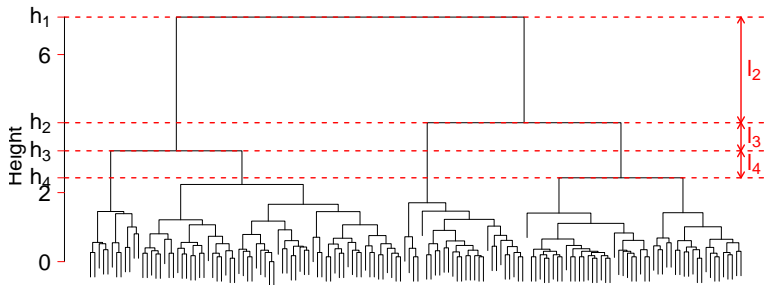
with  $\lambda \geq 0$  regularisation parameter,  $w_{g^s}$  the weight associated to group  $G_g^s$  and  $\rho_s$  the weight associated to the quality of partition  $\mathcal{G}_s$ .

# Multi-Layer Group-Lasso

Weight associated to the quality defined by the highest jump rule :

$$\rho_s = \frac{1}{\sqrt{l_s}}$$

**Cluster Dendrogram**



# Multi-Layer Group-Lasso

- reformulation of the problem such that a given group only appears once, by keeping the smallest weight
- use of classical algorithms of Group-Lasso (e.g. Yang and Zou, 2015)
- multiple testing procedure to control both multiple testing and redundancy (including a hierarchical tests procedure)
- choice of the regularisation parameter which maximises the number of rejections

# Plan

- 1 Introduction
- 2 MLGL : multi-layer group lasso
- 3 Application and R usage**
- 4 Conclusions - Discussion

## Application on the *gasoline* dataset (package *pls*)

Near infrared (NIR) spectra measured using diffuse reflectance from intervals. Search of wavelengths which enable to predict octane number.

- 60 observations
- 401 wavelengths

```
R> hc <- bootstrapHclust(scaleGasNIR, frac = 1, method =  
"average", B = 50)
```

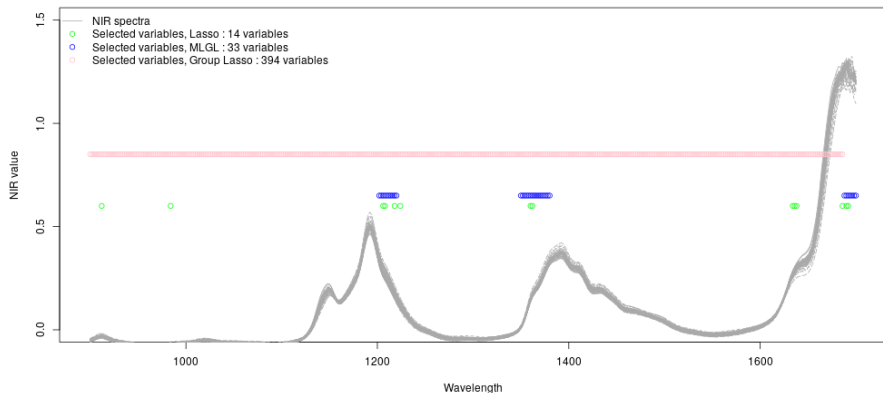
```
R> groupWeight <- computeGroupSizeWeight(hc, sizeMax = 100)
```

```
R> res <- fullProcess(scaleGasNIR, octane, hc = hc,  
+ fractionSampleMLGL = 0.5, weightSizeGroup = groupWeight)
```

`summary` and `plot` give the main results and useful graphs to interpret the quality of the regularisation parameter choice.

# Application on the *gasoline* dataset (package pls)

## Comparison with Lasso and Group-Lasso



# Plan

- 1 Introduction
- 2 MLGL : multi-layer group lasso
- 3 Application and R usage
- 4 Conclusions - Discussion**

## Conclusions - Discussion

MLGL selects groups of correlated variables in high dimension, by combining hierarchical clustering and Group-Lasso.

MLGL allows different levels of the hierarchy to be selected.

The optimal value of regularisation is chosen with a hierarchical multiple testing procedure which gives a low number of rejections. Perspectives would be to improve this step.

MLGL R package is available on CRAN, and its vignette in post-processing stage for publication in Journal of Statistical Software.