Variable clustering for Husler-Reiss graphical models

CAPEL Alexandre

2025-05-12

Contents

We want to use the graphical model tools for extreme value theory to build a new way of clustering variable, as done for the graphical models for Gaussian vector (Touw et al. 2024).

Let $V = \{1, \dots, d\}$.

1 Introduction

For a multivariate random variable, it can be useful to know the dependence structure between the components. Particularly, we can summarise the conditional dependence structure with a graph $\mathcal{G} = (V, E)$ with $E \subset V \times V$ as below :

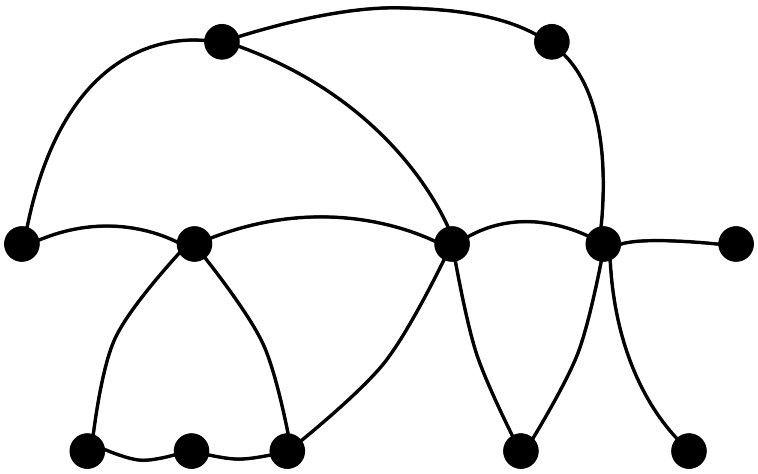

Figure 1: Exemple of a graph

For classical conditional independence (in term of density factorisation), we call such variables graphical models.

Construction of a graph

Let X a graphical model according to the graph $\mathcal{G} = (V, E)$.

Then, by definition, there is the relation :

$$(i, j) \in E \iff X_i \perp\!\!\!\perp X_j \mid X_{V \setminus \{i,j\}}$$

It is the pairwise Markov property.

The graphoid structure of such a relation gives us the equivalence between parwise Markov property and the global Markov property :

$$A \perp_{\mathcal{G}} B \mid C \Longrightarrow X_A \perp\!\!\!\perp X_B \mid X_C$$

where $\perp_{\mathcal{G}}$ is the separation relation between sets of nodes.

We would like to cluster the variable using the graphical model structure to be able to get an interpretation of the clusters and then reduce the dimension of the graph. In that sense, we would have the nodes as the clusters and the edge thanks to the global Markov properties relationship existing between these.

However for general graph, it is not easy. Indeed even with three clusters, we can have this type of situation :

and each time, we have the fact that :

$$X_A \perp\!\!\!\perp X_B \mid X_C$$

So we want to :

- get a **unique** decomposition using the graphical model structure.
- link this decomposition to **a way of clustering the variables.**

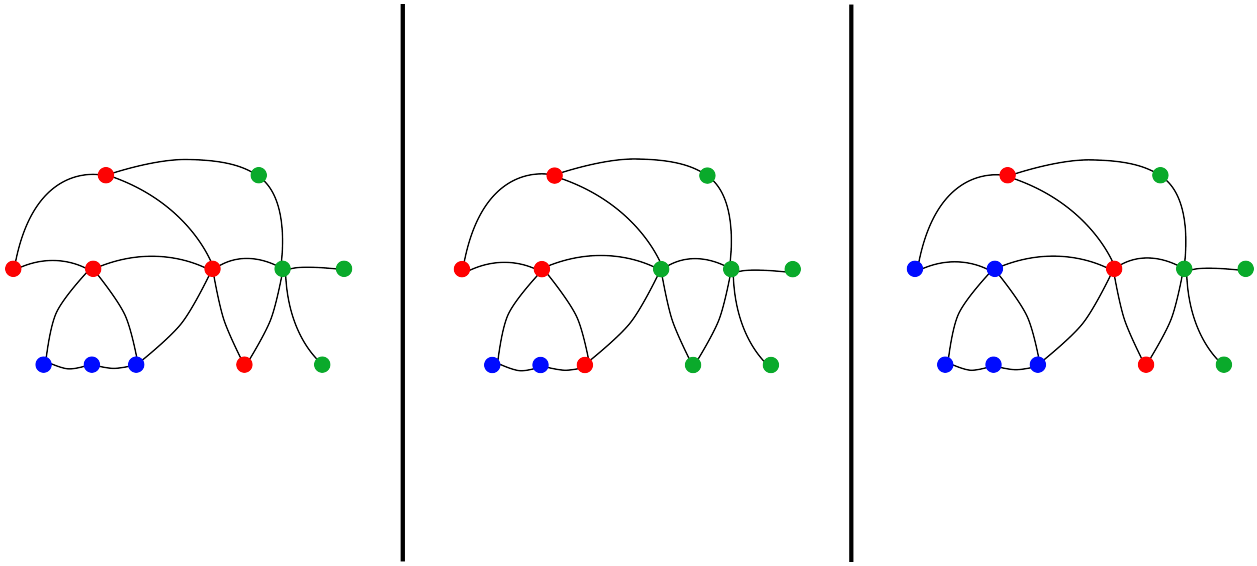

Figure 2: 3 differents way of clustering where A is in blue, B in green and C in red.

2 Husler-Reiss graphical model

In this section, we will present quickly the Husler-Reiss distribution in the MGPD case, and a present the notion of conditional independence in this context, together with the characterisation of this conditional independence for Husler-Reiss graphical models.

2.1 Definition of a Husler-Reiss model

Now we consider a MGPD random vector Y indexed by V.

The Husler-Reiss model is a MGPD parameterized by a symmetric conditionally negative definite matrix Γ with $\mathrm{diag}(\Gamma) = 0$.

One knows that every MGPD is defined by an exponent measure Λ giving a correspondence between MGEV and MGPD (see (Rootzén and Tajvidi 2006)). For the case of the Husler-Reiss distribution, the exponent measure is absolutely continuous with respect to the Lebesgue measure on the cone $\mathcal{E} = \mathbb{R}_+^d \setminus \{0\}$ and its derivative is given by (Engelke et al. 2012) for any $k \in V$:

$$\lambda(y) = y_k^{-2} \prod_{i \neq k} y_i^{-1} \phi(\tilde{y}_{\setminus k}; \Sigma^{(k)}), \quad y \in \mathcal{E}$$

where $\phi(., \Sigma)$ is the density function of a gaussian vector centered with covariance matrix Σ, $\tilde{y}_{\setminus k} = (\log(y_i/y_k) + \Gamma_{ik}/2)_{i \in V}$ and :

$$\Sigma_{ij}^{(k)} = \frac{1}{2}(\Gamma_{ik} + \Gamma_{kj} - \Gamma_{ij}) \quad \text{for } i, j \neq k$$

which is obviously definite positive.

2.2 Characterisation of a HRGM

For extreme value theory, the notion of conditional independence is very complicated to defined. First, for max-stable distribution with continuous positive density, the notion of conditional independence is equivalent to the global independence of the variables (Papastathopoulos reference). Moreover, for the MGPD case,

the random vector is not even defined in a product space which make impossible the use of conidtional independence.

Hopefully, (Engelke and Hitz 2020) build a new notion of conditional independence, adapted to MGPD distribution and then permit us to make graphical model with this type of distribution.

Let A, B and C a parition of V. Then for MGPD random vector Y indexed by V, we say that Y_A is conditionally independent of Y_B given Y_C if :

$$Y_A^k \perp\!\!\!\perp Y_B^k \mid Y_C^k, \qquad \forall k \in V.$$

where Y^k is defined as the vector Y conditionally to the event $\{Y_k > 1\}$.

We note then $Y_A \perp\!\!\!\perp_e Y_B \mid Y_C$.

Moreover, in the same article, they give a first characterisation of the extremal conditional independence for Husler-Reiss models :

Proposition. For a Husler-Reiss graphical model (HRGM) Y with variogram Γ, then for all $i, j \in V$ and $k \in V$ we have :

$$Y_i \perp_e Y_j \mid Y_{\setminus\{i,j\}} \Leftrightarrow \begin{cases} \Theta_{ij}^{(k)} = 0, & \text{if } i, j \neq k, \\ \sum \Theta_{lj}^{(k)} = 0, & \text{if } i = k, j \neq k, \\ \sum \Theta_{il}^{(k)} = 0, & \text{if } i \neq k, j = k \end{cases}$$

where $\Theta^{(k)}$ is the precision matrix of $\Sigma^{(k)}$ (i.e $\Theta^{(k)} = (\Sigma^{(k)})^{-1}$).

In (Hentschel, Engelke, and Segers 2023), they build an extended precision matrix Θ which summarize all the information we need for the conditional independence relationship for the extremal graphical models, in that sense :

$$Y_i \perp\!\!\!\perp_e Y_j \mid Y_{V\setminus\{i,j\}} \quad \Longleftrightarrow \quad \Theta_{ij} = 0.$$

This matrix can be obtain by using some applications (which are bijections) that garanties a form of unicity of the spectral representation.

Therefore, let's consider the following applications :

$$\sigma : \Gamma \mapsto \Pi_d(-\frac{1}{2}\Gamma)\Pi_d, \qquad \theta : \Gamma \mapsto \sigma(\Gamma)^+$$

where the matrix A^+ is the general inverse of A, and Π_d the orthogonal projection matrix in the space $< \mathbb{1} >^\perp$.

They show in (Hentschel, Engelke, and Segers 2023) that the above applications are homeomorphisms between the set of the strictly conditionally negative definite variogram matrix $\mathcal{D}_d$ ad the set of symmetric positive semi-definite matrix with kernel equal to $< \mathbb{1} >$, denoted by $\mathcal{P}_d^1$. More they show that :

$$\sigma^{-1}(\Sigma) = \gamma(\Sigma), \qquad \theta^{-1}(\Theta) = \gamma(\Theta^+),$$

where $\gamma(\Sigma) = \mathbb{1}\text{diag}(\Sigma)^T + \text{diag}(\Sigma)\mathbb{1}^T - 2\Sigma$.

3 Clusterpath for GGM

In this section, we will present the matrix structure we will use for the Husler-Reiss graphical model. More, we will present an algorithm to estimate this graphical structure.

3.1 Gaussian Graphical Model

In (Touw et al. 2024), they build a graphical model that we can use for clustering, in the case of Gaussian graphical model (GGM).

For the GGM, there exists an easy characterisation of the conditional independence which is similar to HRGM for the extreme. For a Gaussian graphical model X with covariance matrix $\tilde{\Sigma}$, we have :

$$X_i \perp\!\!\!\perp_e X_j \mid X_{V\setminus\{i,j\}} \quad \Longleftrightarrow \quad \tilde{\Theta}_{ij} = 0,$$

where $\tilde{\Theta} = \tilde{\Sigma}^{-1}$, the precision matrix.

Let assume that the variable X can be grouped in K clusters $\{C_1, \ldots, C_K\}$ with $p_k = |C_k|$.

The goal was to encouraging clustering of the graph by forcing the precision matrix to have a K blocks structure as follows :

$$\tilde{\Theta} = \begin{pmatrix} (a_1 - r_{11})I_{p_1} & 0 & \ldots & 0 \\ 0 & (a_2 - r_{22})I_{p_2} & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & (a_K - r_{KK})I_{p_K} \end{pmatrix} + \begin{pmatrix} r_{11}\mathbb{1}\mathbb{1}^t & r_{12}\mathbb{1}\mathbb{1}^t & \ldots & r_{1K}\mathbb{1}\mathbb{1}^t \\ r_{21}\mathbb{1}\mathbb{1}^t & r_{22}\mathbb{1}\mathbb{1}^t & \ldots & r_{2K}\mathbb{1}\mathbb{1}^t \\ \vdots & \vdots & \ddots & \vdots \\ r_{K1}\mathbb{1}\mathbb{1}^t & r_{K2}\mathbb{1}\mathbb{1}^t & \ldots & r_{KK}\mathbb{1}\mathbb{1}^t \end{pmatrix},$$

where A is a $K \times K$ diagonal matrix, R a $K \times K$ symmetric matrix, I_p the $p \times p$identity matrix.

We can then get this type of "graph factorisation" which is unique due to precision matrix structure :

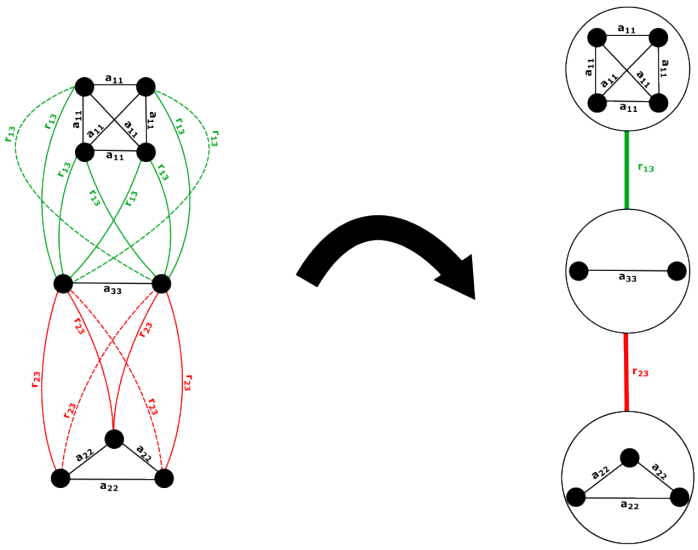

Figure 3: Graph factorisation thanks to the precision matrix structure

Thus, with this factorisation, we build three clusters with an interpretation of conditional independence between them.

For the estimation of the precision matrix, they use the following likelihood :

$$L(\Theta) = -\log(|\Theta|) + \text{tr}(\Sigma\Theta),$$

where $\log(|\cdot|)$ is the logarithm of the determinant and $\text{tr}(\cdot)$ the trace.

To get the maximum likelihood, they use a convex penalty to get the unknown block structure of the precision matrix. Thus, we got this optimisation program :

$$\hat{\Theta} = \arg\min_{\Theta} \Big[-\log(|\Theta|) + \text{tr}(\overline{\Sigma}\Theta) + \lambda\mathcal{P}(\Theta)\Big], \qquad s.t. \Theta^t = \Theta, \quad \Theta > 0$$

where $\overline{\Sigma}$ is an estimation of the covariance matrix $\tilde{\Sigma}$.

From now, as we have for $i, j \in C_k$ that $\theta_{\cdot i} = \theta_{\cdot j}$, we will note θ_{C_k} the vector of the precision matrix of the cluster C_k.

3.2 Clusterpath algorithm

In order to find the groups in precision matrix $\tilde{\Theta}$, we will use the clusterpath algorithm from (Hocking et al., n.d.).

For these convex optimisation programs, we impose to the penalty function to be of the form :

$$\mathcal{P}(\Theta) = \sum_{i<j} w_{ij} D(\theta_{\cdot i}, \theta_{\cdot j}),$$

where w_{ij} are some positive weights, and D a distance in $\mathbb{R}^d$.

The distance D

We can use a lot of distance :

- with the l^p norm for $p \in [1, \infty]$.
- in particular l^1, l^2 and l^∞.
- in (Touw et al. 2024) they use another distance defined as :

$$D(\theta_{\cdot i}, \theta_{\cdot j}) = \sqrt{(\theta_{ii} - \theta_{jj})^2 + \sum_{k \neq i,j} (\theta_{ik} - \theta_{jk})^2}$$

  which "can be interpreted as a group lasso penalty".

Choice of w_{ij}

The choice of w_{ij} is also free, even if they present one which seems better (or nearer from the data) using :

$$w_{ij} = \exp(-\chi||\theta_{\cdot i} - \theta_{\cdot j}||^2)$$

where $||.||$ is the l^2 norm.

Clusterpath algorithm

The algorithm is a gradient descent algorithm, adding conditions to detect clusters and fuse variables.

Algorithm 1 : Clusterpath

Input: initial guess Θ, initial estimation $\overline{\Sigma}$, initial clusters, weight w_{ij}, regularisation λ
G <- `gradient(.)`
while $||G|| > \varepsilon$ **do**
  Θ <- `step_grad(.)`
  Θ, clusters <- `detect_cluster(.)`
  G <- `gradient(.)`
end while
return Θ, clusters

The `gradient` function depends on all the parameters, `step_grad(.)` is just the step part of a gradient descent algorithm : we update the estimation by :

$$\hat{\Theta}_{k+1} \leftarrow \hat{\Theta}_k - h \times \nabla L(\Theta)$$

For the `detect_cluster(.)`, the clusters merged if the distance between two groups C_1 and C_2 is under a small threshold. Then, the coefficient of the new cluster C is computed by the weighted mean of the two other one :

$$\theta_C = \frac{|C_1|\theta_{C_1} + |C_2|\theta_{C_2}}{|C_1| + |C_2|}.$$

We can also try to fuse clusters if the cost function decreases if merging.

4 Clusterpath adaptated for HRGM

Now we want to adapt the previous method to the Husler-Reiss graphical models.

4.1 Maximum likelihood for graphical model

For the estimation of the precision matrix for HRGM, (Hentschel, Engelke, and Segers 2023) shows that find the Θ is equivalent to minimise :

$$L(\Theta) = -\log(|\Theta|_+) - \frac{1}{2} tr(\overline{\Gamma}\Theta),$$

where $\overline{\Gamma}$ is an estimation of the variogram matrix Γ and $|\cdot|_+$ the generalised determinant.

For the next, we will first use the l^2 norm penalty and we will try to minimize :

$$L_{\mathcal{P}}(\Theta, \lambda) = L(\Theta) + \lambda \mathcal{P}(\Theta)$$

with $\lambda > 0$ and $\Theta \in \mathcal{P}_d^1$.

4.2 Adaptation of the expression

As $\Theta \in \mathcal{P}_d^1$, there are supplementary conditions on the matrix : the rows must sum to one !

It follows that :

$$a_k = r_{kk} - \sum_{j=1}^{K} p_j r_{kj}, \qquad \forall k \in \{1, \dots, K\}$$

More, we can rewrite the matrix Θ as follow (Touw et al. 2024) :

$$\Theta = \begin{pmatrix} (a_1 - r_{11})I_{p_1} & 0 & \dots & 0 \\ 0 & (a_2 - r_{22})I_{p_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & (a_K - r_{KK})I_{p_K} \end{pmatrix} + URU^t,$$

where U is a $d \times K$ matrix such that $u_{ij} = 1$ if $i \in C_j$ and 0 otherwise.

Then we can deduce the computation of the trace of $\overline{\Gamma}\Theta$:

$$tr(\overline{\Gamma}\Theta) = \text{tr}(\overline{\Gamma}URU^t) + \sum_{k=1}^{K}(a_k - r_{kk})\text{tr}(\overline{\Gamma}_{C_k})$$

with $\overline{\Gamma}_{C_k}$ the $p_k \times p_k$ matrix computed from $\overline{\Gamma}$ with the indices in C_k. But as $\overline{\Gamma}$ get only zero in its diagonal, we finally obtain :

$$tr(\overline{\Gamma}\Theta) = \text{tr}(\overline{\Gamma}URU^t)$$

Adaptation of the distance

Let's take square l^2 norm penalty for $\mathcal{P}$.

For i, j in the same cluster C_k we have :

$$||\theta_{.i} - \theta_{.j}||^2 = 2(a_k - r_{kk})^2$$

Here, we wish that for two variables in the same cluster, this distance is equal to zero.

The distance D from (Touw et al. 2024) is built for this reason. But in our case we can upgrade the distance : indeed, our matrix Θ have an additionally constraint that rows sum to 0. Thus, we can remove the $(\theta_{ii} - \theta_{jj})^2$ term as the latter is obviously equal to zero if we are in the same cluster.

Therefore, we will consider the following squared distance for the next :

$$D^2(\theta_{.i}, \theta_{.j}) = \sum_{t \neq i,j} (\theta_{it} - \theta_{jt})^2$$

Now, it is time to write the penalty formula with the R matrix. We have :

- for i, j in the same cluster $D^2(\theta_{.i}, \theta_{.j}) = 0$ (it is the goal of this distance).
- for i, j in respectively the clusters C_k and C_l we have :

$$\begin{aligned} D^2(\theta_{.i}, \theta_{.j}) &= \sum_{t \neq i,j} (\theta_{it} - \theta_{jt})^2 \\ &= \sum_{q \neq k,l} p_m(r_{kq} - r_{lq})^2 + (p_k - 1)(r_{kk} - r_{lk})^2 + (p_l - 1)(r_{ll} - r_{lk})^2 \\ &= \tilde{D}^2(r_{.k}, r_{.l}). \end{aligned}$$

Then, by grouping all the terms in $\mathcal{P}$, we get :

$$\mathcal{P}(R) = \sum_{l>k} W_{kl} \tilde{D}^2(r_{\cdot k}, r_{\cdot l})$$

with $W_{lk} = \sum_{i \in C_k} \sum_{j \in C_l} w_{ij}$.

Our likelihood can now be expressed as a function of the R-matrix.

4.3 Computation of the derivative

We can see that the penalized negative log-likelihood can be decomposed as a sum of three element :

$$L_{\mathcal{P}}(R, \lambda) = L_{log}(R) + L_{trace}(R) + \lambda L_{pen}(R)$$

So we just need to compute separately all the derivative for each coefficient (in the upper triangular part, by symmetry).

4.3.1 Gradient of $L_{log}(R)$

Let's denote $E = \{(i,j) \in [\![1,d]\!]^2, i < j\}$ and $F = \{(k,l) \in [\![1,K]\!]^2, k \leq l\}$ and we consider $\mathbb{R}^E$ and $\mathbb{R}^F$ the set of real-valued vector indexed by E anf F.

One can show that there exists a function $f : \mathbb{R}^F \to \mathbb{R}^E$ such that :

$$f(R) = U(\Theta)$$

where U is the application defined in (Hentschel, Engelke, and Segers 2023) which maps a matrix to a vector in $\mathbb{R}^E$ containing the entries in the upper triangular part of the matrix.

Then we can defined also the function $g : \mathbb{R}^E \to \mathbb{R}$ as :

$$g(U(\Theta)) = -\log(|\Theta|_+)$$

which is well defined by symmetry of Θ.

Thus we can see our sub-loglikelihood as :

$$L_{log}(R) = g \circ f(R)$$

And then use the chain rule formula to get the partial derivative for $n \in F$:

$$\frac{\partial L_{log}(R)}{\partial r_n} = \sum_{m \in E} \frac{\partial g}{\partial \theta_m}(f(R)) \times \frac{\partial f_m}{\partial r_n}(R)$$

In the appendix of (Hentschel, Engelke, and Segers 2023), they show that :

$$\frac{\partial g(U(\Theta))}{\partial \theta_m} = \gamma(\Theta^+)_m, \qquad \text{for } m \in E$$

Now we just have to compute the derivative of f. Let $m = (i,j) \in E$. We can stress first that :

$$f_m(R) = r_{kl}, \quad \text{if } i \in C_k,\ j \in C_l, \text{ or } i \in C_l,\ j \in C_k,$$

Then we can deduce the expression below :

$$\frac{\partial f}{\partial r_{kl}} = \begin{cases} 1 & \text{if } i \in C_k,\ j \in C_l,\ \text{or } i \in C_l,\ j \in C_k, \\ 0 & \text{otherwise} \end{cases}$$

We can now associate these expressions to deduce for $k < l$ that :

$$\begin{aligned} \frac{\partial L_{log}(R)}{\partial r_{kl}} &= \sum_{i<j} \gamma(\Theta^+)_{ij} \Big[\mathbb{1}_{i \in C_k} \mathbb{1}_{j \in C_l} + \mathbb{1}_{i \in C_l} \mathbb{1}_{j \in C_k} \Big] \\ &= \sum_{i \in C_k} \sum_{j \in C_l} \gamma(\Theta^+)_{ij} \\ &= u_k^t \gamma(\Theta^+) u_l, \end{aligned}$$

with u_k the k-th column of the matrix of clusters U and because $\gamma(\Theta^+)_{ii} = 0$.

Moreover if $k = l$, we get :

$$\begin{aligned} \frac{\partial L_{log}(R)}{\partial r_{kk}} &= \sum_{\substack{i,j \in C_k \\ i<j}} \gamma(\Theta^+)_{ij} \\ &= \frac{1}{2} u_k^t \gamma(\Theta^+) u_k \end{aligned}$$

that ends the calculation.

Warning. When $p_k = 1$, we have $\frac{\partial L_{log}(R)}{\partial r_{kk}} = 0$.

4.3.2 Gradient of $L_{trace}(R)$

We recall that :

$$L_{trace}(R) = -\frac{1}{2} \text{tr}(\overline{\Gamma} U R U^t)$$

As we have for all symmetric matrix A :

$$\frac{\partial tr(AB)}{\partial b_{ij}} = \begin{cases} 2a_{ij} & \text{if } i \neq j, \\ a_{ii} & \text{otherwise.} \end{cases}$$

Then we can deduce that for $(k, l) \in F$:

$$\frac{\partial L_{trace}(R)}{\partial r_{kl}} = \begin{cases} -(U^t \overline{\Gamma} U)_{kl} & \text{if } k = l, \\ -\frac{1}{2}(U^t \overline{\Gamma} U)_{kk} & \text{otherwise.} \end{cases}$$

Warning. Like the previous section, when $p_k = 1$, we have $\frac{\partial L_{trace}(R)}{\partial r_{kk}} = 0$.

4.3.3 Gradient of $L_{pen}(R)$

We recall that :

$$L_{pen}(R) = \sum_{l'>k'} W_{k'l'} \tilde{D}^2(r_{\cdot k'}, r_{\cdot l'})$$

So we just need to compute the derivative of $\tilde{D}^2$ for each coefficient r_{kl}.

Let $(k, l) \in F$.

Thus, for $k < l$, we have :

$$\frac{\partial D^2(r_{\cdot k'}, r_{\cdot l'})}{\partial r_{kl}} = \begin{cases} 2p_k(r_{kl} - r_{kk'}) & \text{if } k \neq k', l = l' \\ 2p_l(r_{kl} - r_{ll'}) & \text{if } k = k', l \neq l' \\ 2(p_k - 1)(r_{kl} - r_{kk}) + 2(p_l - 1)(r_{kl} - r_{ll}) & \text{if } k = k', l = l' \\ 0 & \text{otherwise.} \end{cases}$$

and for $k' < l'$:

$$\frac{\partial D^2(r_{\cdot k'}, r_{\cdot l'})}{\partial r_{kk}} = \begin{cases} 2(p_k - 1)(r_{kk} - r_{kl'}) & \text{if } k = k' \\ 2(p_k - 1)(r_{kk} - r_{k'k}) & \text{if } k = l' \\ 0 & \text{otherwise.} \end{cases}$$

We can deduce the derivatives :

$$\nabla L_{pen}(R) = \sum_{k'<l'} W_{k'l'} \nabla \tilde{D}^2_{k'l'}(R)$$

where $\tilde{D}^2_{k'l'}(R) = D^2(r_{\cdot k'}, r_{\cdot l'})$.

4.4 Choice of a step for descent

In the gradient descent algorithm, we compute :

$$R^{(t+1)} \leftarrow R^{(t)} - h\nabla_R L(R), \quad \text{with } h > 0$$

where h is called the step size.

However, in that case, we need to verify that after the step, $f(R^{(t+1)})$ is still a symmetric positive matrix.

4.4.1 Research of relationship between R and $f(R) = \Theta$

In this section, we want to find condition on R to make the Θ matrix positive.

First, let's define the $K \times K$ alternative matrix of clusters :

$$\tilde{R} = RP - T$$

where $P = \text{diag}((p_k)_{k=1,\dots,K})$ and $T = \text{diag}((Rp)_{k=1,\dots,K})$.

Proposition. Let R and $\Theta = f(R)$. Assume that :

- $\forall k \in \{1, \dots, K\}, a_k - r_{kk} \geq 0$.
- the alternative matrix of clusters $\tilde{R}$ is positive.

Then Θ is positive.

Proof. A characterization of the positiveness of a matrix is that all its eigen values are positive. We want to get the expression of the latters with the coefficient of R, so we need to find the roots of the characteristic polynomial.

$$
\begin{aligned}
P_\Theta(X) =& \det(\Theta - XI_d) \\
=& \begin{vmatrix}
A_1 - XI_{p_1} & r_{12}\mathbb{1}\mathbb{1}^t & \cdots & \cdots & r_{1K}\mathbb{1}\mathbb{1}^t \\
r_{21}\mathbb{1}\mathbb{1}^t & A_2 - XI_{p_2} & \ddots & & \vdots \\
\vdots & \ddots & \ddots & \ddots & \vdots \\
\vdots & & \ddots & A_{K-1} - XI_{p_{K-1}} & r_{KK-1}\mathbb{1}\mathbb{1}^t \\
r_{K1}\mathbb{1}\mathbb{1}^t & \cdots & \cdots & r_{KK-1}\mathbb{1}\mathbb{1}^t & A_K - XI_{p_K}
\end{vmatrix}
\end{aligned}
$$

with $A_k = \begin{pmatrix} a_k & r_{kk} & \cdots & r_{kk} \\ r_{kk} & a_k & \ddots & \vdots \\ \vdots & \ddots & \ddots & r_{kk} \\ r_{kk} & \cdots & r_{kk} & a_{kk} \end{pmatrix}$.

First, we can focus our calculation on the p_1 first lines. If we subtract the $p_1 - 1$ first lines by the $p_1 - th$, we obtain :

$$
\begin{aligned}
P_\Theta(X) =& \\
& \begin{vmatrix}
a_1 - r_{11} - X & 0 & \cdots & 0 & -(a_1 - r_{11} - X) & 0 & \cdots & 0 \\
0 & \ddots & \ddots & \vdots & \vdots & \vdots & & \vdots \\
\vdots & \ddots & \ddots & 0 & \vdots & \vdots & & \vdots \\
0 & \cdots & 0 & a_1 - r_{11} - X & -(a_1 - r_{11} - X) & 0 & \cdots & 0 \\
r_{11} & \cdots & & r_{11} & a_1 - X & r_{12}\mathbb{1}^t & \cdots & r_{1K}\mathbb{1}^t \\
 & & r_{21}\mathbb{1}\mathbb{1}^t & & & A_2 - XI_{p_2} & \ddots & \vdots \\
 & & \vdots & & & \ddots & \ddots & \vdots \\
 & & \vdots & & & & \ddots & A_{K-1} - XI_{p_{K-1}} \quad r_{KK-1}\mathbb{1}\mathbb{1}^t \\
 & & r_{K1}\mathbb{1}\mathbb{1}^t & & & \cdots & \cdots & r_{KK-1}\mathbb{1}\mathbb{1}^t \quad A_K - XI_{p_K}
\end{vmatrix} \\
=& (a_1 - r_{11} - X)^{p_1 - 1} \begin{vmatrix}
1 & 0 & \cdots & 0 & -1 & 0 & \cdots & 0 \\
0 & \ddots & \ddots & \vdots & \vdots & \vdots & & \vdots \\
\vdots & \ddots & \ddots & 0 & \vdots & \vdots & & \vdots \\
0 & \cdots & 0 & 1 & -1 & 0 & \cdots & 0 \\
r_{11} & \cdots & r_{11} & a_1 - X & r_{12}\mathbb{1}^t & \cdots & & r_{1K}\mathbb{1}^t \\
 & r_{21}\mathbb{1}\mathbb{1}^t & & & A_2 - XI_{p_2} & \ddots & & \vdots \\
 & \vdots & & & \ddots & \ddots & \ddots & \vdots \\
 & \vdots & & & & \ddots & A_{K-1} - XI_{p_{K-1}} & r_{KK-1}\mathbb{1}\mathbb{1}^t \\
 & r_{K1}\mathbb{1}\mathbb{1}^t & & & \cdots & \cdots & r_{KK-1}\mathbb{1}\mathbb{1}^t & A_K - XI_{p_K}
\end{vmatrix}
\end{aligned}
$$

Thus we can reproduce these operations in each p_k blocks. Finally, by summing the one column with the minus one, we get :

$$P_\Theta(X) = \prod_{k=1}^{K} (a_k - r_{kk} - X)^{p_k - 1} \det(\tilde{R})$$

Obviously, if the condition of the proposition are true, then the root of this polynomial are all positive.

□

4.4.2 Gridline search for the optimal step size

With the previous proposition, we can have an upper bound for the step size h. Indeed, let's note δ_{kl} the gradient along r_{kl}. Then, if the previous step $R^{(t)}$ verifies the conditions of the proposition, the optimal step size h^* should verify :

$$\begin{aligned}
&\forall k \in \{1, \dots, K\},\ a_k^{(t+1)} - r_{kk}^{(t+1)} \geq 0 \\
\iff &\forall k \in \{1, \dots, K\},\ \sum_{l=1}^{K} p_l r_{lk}^{(t+1)} \leq 0 \\
\iff &\forall k \in \{1, \dots, K\},\ \sum_{l=1}^{K} p_l (r_{lk}^{(t)} - h^* \delta_{lk}) \leq 0 \\
\iff &\forall k \in \{1, \dots, K\},\ \sum_{l=1}^{K} p_l r_{lk}^{(t)} \leq h^* \sum_{l=1}^{K} p_l \delta_{lk}
\end{aligned}$$

Therefore, we obtain the cases :

- if $\sum_{l=1}^{K} p_l \delta_{lk} > 0$ then $h^* \geq \sum_{l=1}^{K} p_l r_{lk}^{(t)} / \sum_{l=1}^{K} p_l \delta_{lk}$ which is a negative number as $R^{(t)}$ follows the relations. So, as $h^* > 0$, there is no constraint in this case.
- if $\sum_{l=1}^{K} p_l \delta_{lk} < 0$ then $h^* \leq \sum_{l=1}^{K} p_l r_{lk}^{(t)} / \sum_{l=1}^{K} p_l \delta_{lk}$.
- if $\sum_{l=1}^{K} p_l \delta_{lk} = 0$, no constraint as well except non negative.

Thus, we will set our grid line in the segment $(0, h_{max}]$ where :

$$h_{max} = \min_{k=1,\dots,K} \left| \frac{\sum_{l=1}^{K} p_l r_{lk}^{(t)}}{\sum_{l=1}^{K} p_l \delta_{lk}} \right|$$

and we will choose the value h^* which minimizes the objective function provided that the alternative matrix of clusters $\tilde{R}^{(t+1)}$ is positive.

4.5 Fusing approximation

Let suppose that we can write $\Theta = f(R)$ and let's note ε_f the fusing threshold between two clusters. Assume that the two clusters C_k and C_l are mergeable that is to say :

$$\tilde{D}^2(r_{r\cdot}, r_{l\cdot}) \leq \varepsilon_f$$

In that case, we note the fused cluster $C = C_k \cup C_l$ and the new matrix R^* which is indexed by $(\{1, \dots, K\} \setminus \{k, l\}) \cup C$ is defined as :

$$r^*_{pm} = \begin{cases} \frac{p_k r_{km} + p_l r_{lm}}{p_k + p_l} & \text{if } m \neq p = C \\ \frac{p_k r_{kp} + p_l r_{lp}}{p_k + p_l} & \text{if } p \neq m = C \\ r_{kl} & \text{if } m = p = C \\ r_{pm} & \text{otherwise} \end{cases}$$

Proposition In the setting as above, we have :

- for small enough ε_f, $f(R^*)$ is valid.
- for fixed λ and weights w, we have :

$$|L_{\mathcal{P}}(R, \lambda) - L_{\mathcal{P}}(R^*, \lambda)| \underset{\varepsilon_f \to 0}{\longrightarrow} 0$$

Proof. For the first point, let's consider the metric space $\mathcal{S}^1_d$, the space of symmetric matrix with kernel equal to $\mathbf{span}(\{\mathbb{1}\})$, equipped with the topology of $(\mathcal{M}_d(\mathbb{R}), ||\cdot||_F)$, where $||\cdot||_F$ is the Frobenius norm.

Now, we focus on the function :

$$\begin{aligned} g : \mathcal{S}^1_d &\longrightarrow \mathbb{R}^{d-1} \\ \Theta &\longmapsto \lambda(\Theta) = (\lambda_i(\Theta))_{i=1,\dots,d-1} \end{aligned}$$

We know that g is continuous then $g^{-1}((\mathbb{R}^*_+)^{d-1})$ is an open set of $\mathcal{S}^1_d$. Therefore, it exists $\varepsilon^* > 0$ such that :

$$B(\Theta, \varepsilon^*) \cap \mathcal{S}^1_d \subset g^{-1}((\mathbb{R}^*_+)^{d-1})$$

Thus, we know that if ε_f is small, then $\Theta^* = f(R^*) \in \mathcal{S}^1_d$ and Θ are closed for the Frobenius norm, so we finally have $\Theta^* \in \mathcal{P}^1_d$.

For the second point, we need to show first that :

$$\tilde{D}^2(r_{r\cdot}, r_{l\cdot}) \leq \varepsilon_f^2 \Longrightarrow ||f(R) - f(R^*)||_F \leq \sqrt{(2 + ||p||^2) p_C} \varepsilon_f$$

First, we have :

$$(\Theta - \Theta^*)_{ij} = \begin{cases} a_{k'} - a^*_{k'} & \text{if } i = j \in C_{k'} \\ r_{k'k'} - r^*_{k'k'} & \text{if } i, j \in C_{k'}, i \neq j \\ r_{k'l'} - r^*_{k'l'} & \text{if } i \in C_{k'}, j \in C_{l'} \end{cases}$$

We choose to keep the same indices for R and R^* and $r^*_{kl'} = r^*_{Cl'}$ and the same for all the other cases (it does not change the results).

Case $k' \notin \{k, l\}$

We have :

$$
\begin{aligned}
a^*_{k'} &= r^*_{k'k'} - \sum_{l' \neq k,l} p_{l'} r^*_{k'l'} - p^*_C r_{Ck'} \\
&= r_{k'k'} - \sum_{l' \neq k,l} p_{l'} r_{k'l'} - (p_k + p_l) \frac{p_k r_{k'k} + p_l r_{k'l}}{p_k + p_l} \\
&= r_{k'k'} - \sum_{l'=1}^{K} p_{l'} r_{k'l'} \\
&= a_{k'}
\end{aligned}
$$

and therfore $(\Theta - \Theta^*)_{ii} = 0$ for $i \in C_{k'}$.

Case $k' \in \{k, l\}$

Let's focus ourselves in the case when $k' = k$ (the other is exactly the same). We have :

$$
a^*_k = r^*_{CC} - \sum_{l' \neq k,l} p_{l'} r^*_{Cl'} - p^*_C r_{CC} = r_{kl} - \sum_{l' \neq k,l} p_{l'} r^*_{Cl'} - (p_k + p_l) r_{kl}
$$

Thus :

$$
(a_k - a^*_k)^2 = \Big[\sum_{l' \neq k,l} p_{l'} (r_{kl'} - r^*_{Cl'}) + r_{kk} - r_{kl} - p_k r_{kk} - p_l r_{kl} - (p_k + p_l) r_{kl} \Big]^2 = \Big[\sum_{l' \neq k,l} p_{l'} (r_{kl'} - r^*_{Cl'}) + (p_k - 1)(r_{kk} - r_{kl}) \Big]^2
$$

By using Lipschtizian property of linear application in finite dimension we finaly get :

$$
(a_k - a^*_k)^2 \leq ||p||^2 \varepsilon_f^2
$$

We get the same inequality if $i \in C_l$.

Now $i \neq j$.

Case $i, j \in C_{k'}$

First, we can notice that this case can happen for a k' only if $p_{k'} > 1$. We have :

$$
(\Theta - \Theta^*)_{ij} = \begin{cases} r_{kk} - r_{kl} & \text{if } k' = k, \\ r_{ll} - r_{kl} & \text{if } k' = l, \\ 0 & \text{otherwise.} \end{cases}
$$

Case $i \in C_{k'}, j \in C_{l'}$

$$
(\Theta - \Theta^*)_{ij} = \begin{cases} r_{kl'} - r^*_{kl'} & \text{if } k' = k, l' \notin \{k, l\} \\ r_{ll'} - r^*_{ll'} & \text{if } k' = l, l' \notin \{k, l\} \\ 0 & \text{otherwise.} \end{cases}
$$

(Sure, the role of k' and l' is symmetric due to the symmetry of the matrices).

Now we need to see the structure of the difference between the mamtrices. We can summarize this with the following figure :

The black stars in the upper figure is zero. Thus we have :

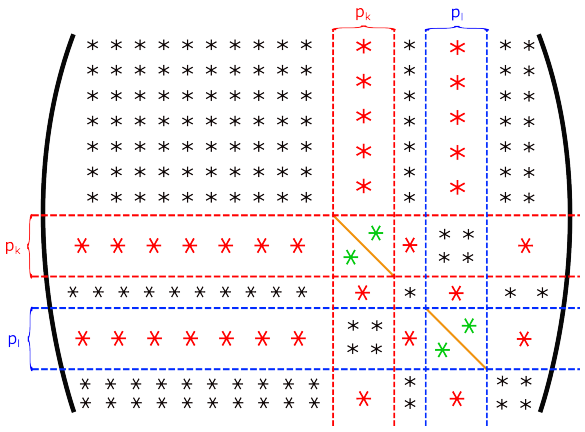

Figure 4: Coefficient repartition in the difference of the matrices.

- by combining red stars and green stars that :

$$\sum_{j\neq i}(\Theta_{ij}-\Theta^*_{ij})^2=\sum_{l'\neq k,l}p_{l'}(r_{kl'}-r^*_{Cl'})^2+(p_k-1)(r_{kk}-r_{kl})^2\leq\varepsilon_f^2$$

if $i\in C_k$.

- by combining red stars and green stars again that :

$$\sum_{j\neq i}(\Theta_{ij}-\Theta^*_{ij})^2=\sum_{l'\neq k,l}p_{l'}(r_{ll'}-r^*_{Cl'})^2+(p_l-1)(r_{ll}-r_{kl})^2\leq\varepsilon_f^2$$

if $i\in C_l$.

Thus, we get :

- in black, the zeros.
- in orange, $p_C(=p_k+p_l)$ majorations by $||p||^2\varepsilon_f^2$.
- for the red zone (without orange), we have $2p_k$ majorations by ε_f^2.
- for the blue zone (without orange), we have $2p_l$ majorations by ε_f^2.

Therefore, we finally obtain :

$$\begin{aligned}||\Theta-\Theta^*||_F^2&\leq 2p_C\varepsilon_f^2+p_C||p||^2\varepsilon_f^2\\&\leq\left[2+||p||^2\right]p_C\varepsilon_f^2\end{aligned}$$

Note that we will obtain the same expression if $p_{\max}=1$ because there will be zero green majoration.

Now, we know that in $[a,\infty)$ the logarithm is $\frac{1}{a}$-Lispschizian, so we can deduce by the lemma on generalised determinant that:

$$|\log(|f(R)|_+)-\log(|f(R)|_+)|\leq\frac{L_m}{a}||f(R)-f(R^*)||_F$$

Also, we can find that :

$$
\begin{aligned}
|\frac{1}{2}\mathrm{tr}(\bar{\Gamma}\Theta) - \frac{1}{2}\mathrm{tr}(\bar{\Gamma}\Theta^*)| &= |\frac{1}{2}\sum_{i,j}\bar{\gamma}_{ij}(\theta_{ij} - \theta^*_{ij})| \\
&\leq \frac{1}{2}\sum_{i,j}|\bar{\gamma}_{ij}||\theta_{ij} - \theta^*_{ij}| \\
&\leq \sum_{i<j}|\bar{\gamma}_{ij}|||\Theta - \Theta^*||_F
\end{aligned}
$$

That means :

$$
|\frac{1}{2}\mathrm{tr}(\bar{\Gamma}f(R)) - \frac{1}{2}\mathrm{tr}(\bar{\Gamma}f(R^*))| \leq k_{\bar{\Gamma}}||f(R) - f(R^*)||_F
$$

where $k_{\bar{\Gamma}} = \sum_{i<j}|\bar{\gamma}_{ij}|$.

And finally, as we have :

$$
|D^2(\theta_{.i}, \theta_{.j}) - D^2(\theta^*_{.i}, \theta^*_{.j})| \leq D^2(\theta_{.i}, \theta^*_{.i}) + D^2(\theta_{.j}, \theta^*_{.j})
$$

And that :

$$
D^2(\theta_{.i}, \theta^*_{.i}) \leq ||\Theta - \Theta^*||^2_F
$$

We can deduce that :

$$
|\mathcal{P}(\Theta) - \mathcal{P}(\tilde{\Theta}^*)| \leq k_W||\Theta - \Theta^*||^2_F
$$

where $k_W = 2\sum_{i<j}w_{ij}$

Thus, we can fuse all the inequality to get :

$$
\begin{aligned}
|L_{\mathcal{P}}(R, \lambda) - L_{\mathcal{P}}(R^*, \lambda)| &\leq (\frac{L_m}{a} + k_{\bar{\Gamma}} + \lambda k_W||f(R) - f(R^*)||_F)||f(R) - f(R^*)||_F \\
&\leq \sqrt{(2 + ||p||^2)p_C}(\frac{L_m}{a} + k_{\bar{\Gamma}} + \lambda\sqrt{(2 + ||p||^2)p_C}\varepsilon_f k_W)\varepsilon_f
\end{aligned}
$$

And so $|L_{\mathcal{P}}(R, \lambda) - L_{\mathcal{P}}(R^*, \lambda)| \to 0$ when $\varepsilon_f \to 0$.

□

I think we can optimize the constant.

4.6 Condition on the one cluster fusing

One problem can appear during the optimization procedure : the non fusion of the clusters in one element when $\lambda \to \infty$.

Indeed, let's take W the weigth matrix. From this matrix, we can build a graph $\mathcal{G}$ on V were the edges are defined :

$$
(i, j) \in E \iff w_{ij} \neq 0
$$

Now if we assume that this graph is not connected (and there exists $M > 1$ connected components, denoted by E_m), then we can rewrite the objective function as :

$$L_{\mathcal{P}}(R, \lambda) = L_{\log}(R) + L_{trace}(R) + \lambda \sum_{m=1}^{M} \Big(\sum_{i,j \in E_m} w_{ij} d^2(\theta_{i\cdot}, \theta_{j\cdot}) \Big)$$

In that equation, we see that we will not have garantee that the clusters E_m fuse between them because for large λ we will normally get the E_m clusters and then the penalty will not appear when we get the M clusters.

4.7 Hierachical clustering

We would like to justify the use of a dendrogram to represents the clusterpath of the variables along the value of λ.

First let's recall the problem and see what we can say. We would like to get a solution :

$$Q_\lambda : \arg\min_{\Theta} \{L(\Theta) + \lambda \mathcal{P}(\Theta)\}, \qquad \Theta \in \mathcal{P}_d^1.$$

First, (Hentschel, Engelke, and Segers 2023) show that the function L is convex on $\mathcal{P}_d^1$. As $\mathcal{P}$ is only a linear combination (with positive coefficient) of distances, the penalised likelihood is also convex on $\mathcal{P}_d^1$.

Moreover, we have also :

Proposition. The set $\mathcal{P}_d^1$ is convex.

Proof. By definition a matrix A belongs to $\mathcal{P}_d^1$ if it is symmetric and :

$$x^t A x \geq 0, \qquad \text{with equality iff } x \in < \mathbb{1} > .$$

Let be A and B two matrices in $\mathcal{P}_d^1$. Let $\mu \in (0, 1)$ then we have :

- $\mu A + (1 - \mu) B$ is symmetric.
- let $x \in \mathbb{R}^d$, then :
$$x^t(\mu A + (1-\mu)B)x = \mu \underbrace{(x^t A x)}_{\geq 0} + (1-\mu) \underbrace{x^t B x}_{\geq 0} \geq 0$$

More $x^t(\mu A + (1-\mu)B)x = 0$ if and only if $x^t A x = 0$ and $x^t B x = 0$ (because all is positive). As A and B are in $\mathcal{P}_d^1$ theses two equality occur if and only if $x \in < \mathbb{1} >$. Thus, $\mu A + (1-\mu)B \in \mathcal{P}_d^1$.

□

Thus the problem Q_λ is a convex problem and admits a unique minimizer.

Now, let's study in details a specific model.

Proposition. Let $\lambda > 0$ and the problem Q_λ. Assume that it exists $\lambda^* > 0$ such that the solution of the corresponding problem verifies $\mathcal{P}(\Theta) = 0$. Then for all $\lambda \geq \lambda^*$, the solution of Q_λ is the same as Q_{λ^*}.

Proof. First let's note that for $\lambda' \leq \lambda$ with Θ and Θ' the respective solutions of Q_λ and $Q_{\lambda'}$, and we have :

$$\mathcal{P}(\Theta') \geq \mathcal{P}(\Theta)$$

Now if we take $\lambda' = \lambda^*$, we have :

$$\mathcal{P}(\Theta) = 0$$

And thus for all $\lambda^* \leq \lambda$ the problem Q_λ is equivalent to :

$$\arg \min_{\Theta, \mathcal{P}(\Theta)=0} L(\Theta)$$

But as the set $\{\Theta \in \mathcal{P}_1^d, \mathcal{P}(\Theta) = 0\}$ is convex, the solution is unique again and then does not change from λ^*.

□

From now, we will suppose that the assumptions of the last proposition are verified. Then, we can define a the following function :

$$\begin{aligned} \Theta : [0, \lambda^*] &\to \mathcal{P}_d^1 \\ \lambda &\mapsto \Theta(\lambda) \end{aligned}$$

where $\Theta(\lambda)$ is the solution of the problem Q_λ (well defined by the convexity of the problem).

We would like to show that it is a path between two elements in $\mathcal{P}_d^1$: the first one is the classical estimator of the precision matrix $\Theta(0) = \hat{\Theta}$ and the second one is a matrix such that we will note $\Theta^* = \Theta(\lambda^*)$ for convenience. Thus, the main goal here is to prove the continuity of the application of solution.

Lemma. Let be $\{\Theta(\lambda)\}_{\lambda \in [0,\lambda^*]}$, the collection of all the solution of the problem Q_λ for all $\lambda \geq 0$. The set $\{||\Theta(\lambda)||_F \mid \lambda \in [0, \lambda^*]\}$ is bounded.

Proof. First, let's stress that the application L is coercive on $\mathcal{P}_d^1$, that means that :

$$\lim_{||\Theta||_F \to +\infty} L(\Theta) = +\infty$$

Then, as we have for all $\lambda \leq \lambda^*$:

$$L(\Theta(\lambda)) + \lambda \mathcal{P}(\Theta(\lambda)) \leq L(\Theta^*) + \lambda^* \mathcal{P}(\Theta^*)$$

Therefore, as the image of the collection by the application is bounded, we must have that the set $\{||\Theta(\lambda)||_F \mid \lambda \in [0, \lambda^*]\}$ is bounded otherwise it will bring a contradiction with the coercivity of the appplication L.

□

Proposition. The application $\lambda \mapsto \Theta(\lambda)$ is a path between $\hat{\Theta}$ and Θ^*.

Proof. Let be $(\lambda_n)_{n \in \mathbb{N}}$ a sequence of $[0, \lambda^*]$ which converges to a limit λ'. Then, we can build a new sequence $(\Theta_n)_{n \in \mathbb{N}}$ in $\mathcal{P}_d^1$, induced by the latter, and defined by :

$$\Theta_n = \Theta(\lambda_n)$$

We would like to show that the sequence $(\Theta_n)_{n \in \mathbb{N}}$ converges to the matrix $\Theta' = \Theta(\lambda')$.

First, from the lemma, we have that the sequence $(\Theta_n)_{n \in \mathbb{N}}$ is bounded. Moreover, let's take a convergent subsequence $(\Theta_{\phi(n)})_{n \in \mathbb{N}}$ and let's note its limit $\tilde{\Theta}$. By definition, we have that for all $\Theta \in \mathcal{P}_d^1$:

$$L(\Theta_{\phi(n)}) + \lambda_{\phi(n)} \mathcal{P}(\Theta_{\phi(n)}) \leq L(\Theta) + \lambda_{\phi(n)} \mathcal{P}(\Theta)$$

Therefore, if we take the limit :

$$L(\tilde{\Theta}) + \lambda' \mathcal{P}(\tilde{\Theta}) \leq L(\Theta) + \lambda' \mathcal{P}(\Theta)$$

But, this means that $\tilde{\Theta}$ is a solution of the problem $Q_{\lambda'}$ and by unicity we have $\tilde{\Theta} = \Theta'$.

So, as the sequence $(\Theta_n)_{n\in\mathbb{N}}$ is bounded and admits only one subsequential limit, the sequence converge to this limit which is $\Theta(\lambda')$. So the application $\lambda \mapsto \Theta(\lambda)$ is continuous.

□

Now, we would like to show the crucial point of our algorithm. Let's note :

$$d_{ij}(\Theta) = D(\theta_{.i}, \theta_{.j}),$$

for all $i, j \in V$.

Theorem. Let be the problem Q_λ and let's assume that it exists $\lambda^* > 0$ such that the solution of the corresponding problem verifies $\mathcal{P}(\Theta) = 0$. Thus, we have for all $\lambda < \lambda'$:

$$d_{ij}(\Theta(\lambda)) < \varepsilon_f \implies \quad d_{ij}(\Theta(\lambda')) < \varepsilon_f$$

This means that if we fuse only two variable for some penalization, we will fuse the two same variables for stronger penalization.

Proof. Let's assume that it exists a $lambda > 0$ such that :

$$d_{ij}(\Theta(\lambda)) < \varepsilon_f$$

Then, we can find a positive number $\eta > 0$ such that :

$$d_{ij}(\Theta(\lambda)) \leq \varepsilon_f - \eta$$

As the Θ function is defined on a compact space ($[0, \lambda^*]$) it follows from the Heine Theorem that $\lambda \mapsto d_{ij}(\Theta(\lambda))$ is also uniformly continuous, that means we can find δ_η such that :

$$|\lambda_1 - \lambda_2| \leq \delta_\eta \Rightarrow |d_{ij}(\Theta(\lambda_1)) - d_{ij}(\Theta(\lambda_2))| < \frac{\eta}{2}$$

Now, let's take $\lambda' > 0$ such that $|\lambda - \lambda'| \leq \delta_\eta$, then :

$$d_{ij}(\Theta(\lambda')) \leq \frac{\eta}{2} + d_{ij}(\Theta(\lambda)) \leq \frac{\eta}{2} + \varepsilon_f - \eta \leq \varepsilon_f - \frac{\eta}{2}$$

□

5 Simulation study

Now, we want to check if the algorithm is working on basic simulation of Husler-Reiss data. For the simulation, we will use the function `rmpareto()` from the `R`-package `graphicalExtremes` developed in (Engelke and Hitz 2020).

5.1 First simulation

We will first try our algorithm on easy example. We will assume that the Husler-Reiss model gets only two clusters of size 4 and 3. We will fix the following values :

$$R = \begin{pmatrix} 0.5 & -2 \\ -2 & 1 \end{pmatrix}$$

and the clusters are $C_1 = \{1, 2, 3, 4\}$ and $C_2 = \{5, 6, 7\}$.

We can also verify that this matrix follows the condition of the previous section.

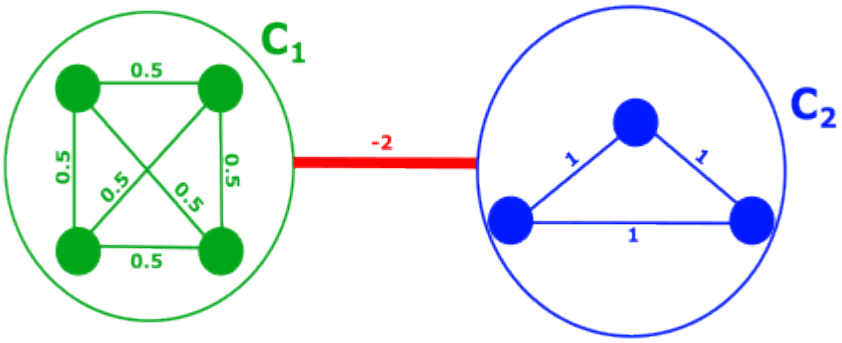

Figure 5: Graph factorisation for the first simulation.

We can then deduce the precision matrix Θ of our model :

$$\Theta = \begin{pmatrix} 4.5 & 0.5 & 0.5 & 0.5 & -2 & -2 & -2 \\ 0.5 & 4.5 & 0.5 & 0.5 & -2 & -2 & -2 \\ 0.5 & 0.5 & 4.5 & 0.5 & -2 & -2 & -2 \\ 0.5 & 0.5 & 0.5 & 4.5 & -2 & -2 & -2 \\ -2 & -2 & -2 & -2 & 6 & 1 & 1 \\ -2 & -2 & -2 & -2 & 1 & 6 & 1 \\ -2 & -2 & -2 & -2 & 1 & 1 & 6 \end{pmatrix}$$

with the variogram of our Husler-Reiss graphical model (using `Theta2Gamma()` function) and also the extremal coefficient matrix χ :

$$\Gamma = \begin{pmatrix} 0 & 0.5 & 0.5 & 0.5 & 0.36 & 0.36 & 0.36 \\ 0.5 & 0 & 0.5 & 0.5 & 0.36 & 0.36 & 0.36 \\ 0.5 & 0.5 & 0 & 0.5 & 0.36 & 0.36 & 0.36 \\ 0.5 & 0.5 & 0.5 & 0 & 0.36 & 0.36 & 0.36 \\ 0.36 & 0.36 & 0.36 & 0.36 & 0 & 0.4 & 0.4 \\ 0.36 & 0.36 & 0.36 & 0.36 & 0.4 & 0 & 0.4 \\ 0.36 & 0.36 & 0.36 & 0.36 & 0.4 & 0.4 & 0 \end{pmatrix}, \chi = \begin{pmatrix} 1 & 0.72 & 0.72 & 0.72 & 0.76 & 0.76 & 0.76 \\ 0.72 & 1 & 0.72 & 0.72 & 0.76 & 0.76 & 0.76 \\ 0.72 & 0.72 & 1 & 0.72 & 0.76 & 0.76 & 0.76 \\ 0.72 & 0.72 & 0.72 & 1 & 0.76 & 0.76 & 0.76 \\ 0.76 & 0.76 & 0.76 & 0.76 & 1 & 0.75 & 0.75 \\ 0.76 & 0.76 & 0.76 & 0.76 & 0.75 & 1 & 0.75 \\ 0.76 & 0.76 & 0.76 & 0.76 & 0.75 & 0.75 & 1 \end{pmatrix}$$

Now, we have the parameters to build simulations and try to cluster the variable according to the underlying structure.

Setup

- we will simulate $n = 2000$ variables following Husler-Reiss multivariate Pareto distribution, with the upper variogram.
- we will estimate the variogram using the empirical extremal variogram estimator (using the function `emp_vario()`). The empirical precision matrix $\overline{\Theta}$ is deduced from this previous estimation.
- we will use the exponential weights defined by :

$$w_{ij} = \exp(-\chi D^2(\overline{\theta}_{i,\cdot}, \overline{\theta}_{j,\cdot}))$$

  where the tune parameter χ is equal to one.
- the merging threshold in `merge_clusters()` is set at $\varepsilon = 10^{-1}$.

Moreover, the optimisation will begin with no assumption in the clusters, and so there is as much clusters as variables.

Finally, we will choose the λ parameter using a grid and taking the one which produce the smallest penalised negative log-likelihood.

For the results, we get :

- we obtained the right clusters C_1 and C_2 when we finish we two clusters.
- we estimated the $\hat{R}$ matrix as :

$$\hat{R} = \begin{pmatrix} 0.6901078 & 0.6467034 \\ 0.6467034 & 4.1433788 \end{pmatrix}$$

- a negative log-likelihood which takes the value -3.91 whereas it takes the value -3.92 for non-penalised optimization.

As we saw previously, there is the graph of the weight matrix W for this simulation :

We immediately notice that the graph seems to be numerically unconnected, and this can be felt on the optimization with large value of λ : indeed, for $\lambda = 10^7$ we still find the two right clusters which did not merge.

For better understanding, there are several graph which describe the evolution of the number of clusters and the correspondence between the true clsuter and the estimated one :

One last thing we can show is the hierarchical clustering graph, that is original goal of this method :

With replications.

There is the results for 500 replications of the previous experiment :

6 Application on fligth delay data

7 Appendix

7.1 Frobenius norm

Definition/Proposition. Let $\mathcal{M}_d(\mathbb{R})$ the space of $d \times d$ matrices. We can define the following bi-linear application :

$$[A|B] = \mathrm{tr}(AB^t)$$

which is a scalar product. The corresponding norm is called Frobenius norm and can be expressed as :

$$||A||_F = \sqrt{tr(AA^t)} = \sqrt{\sum_{i=1}^{d}\sum_{j=1}^{d} a_{ij}^2}$$

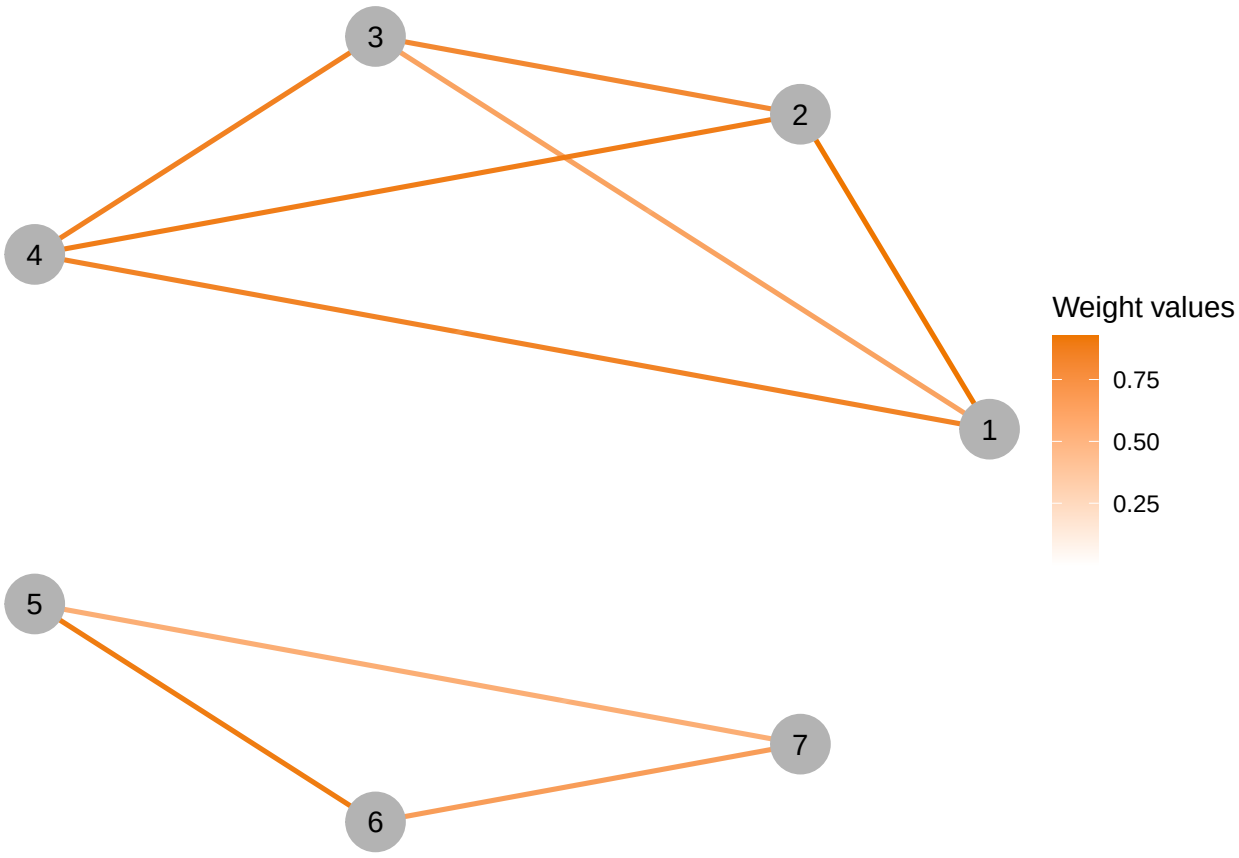

Figure 6: Graph according to the adjacency matrix of weights W.

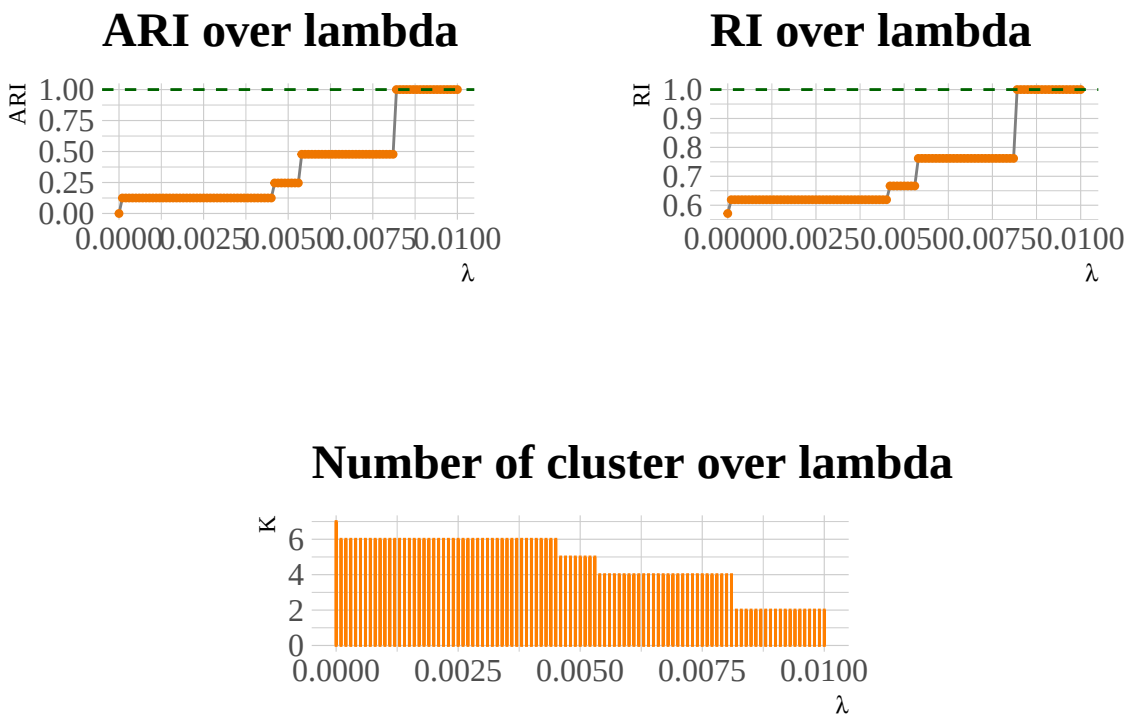

Figure 7: Summary of the optimization results with the λ.

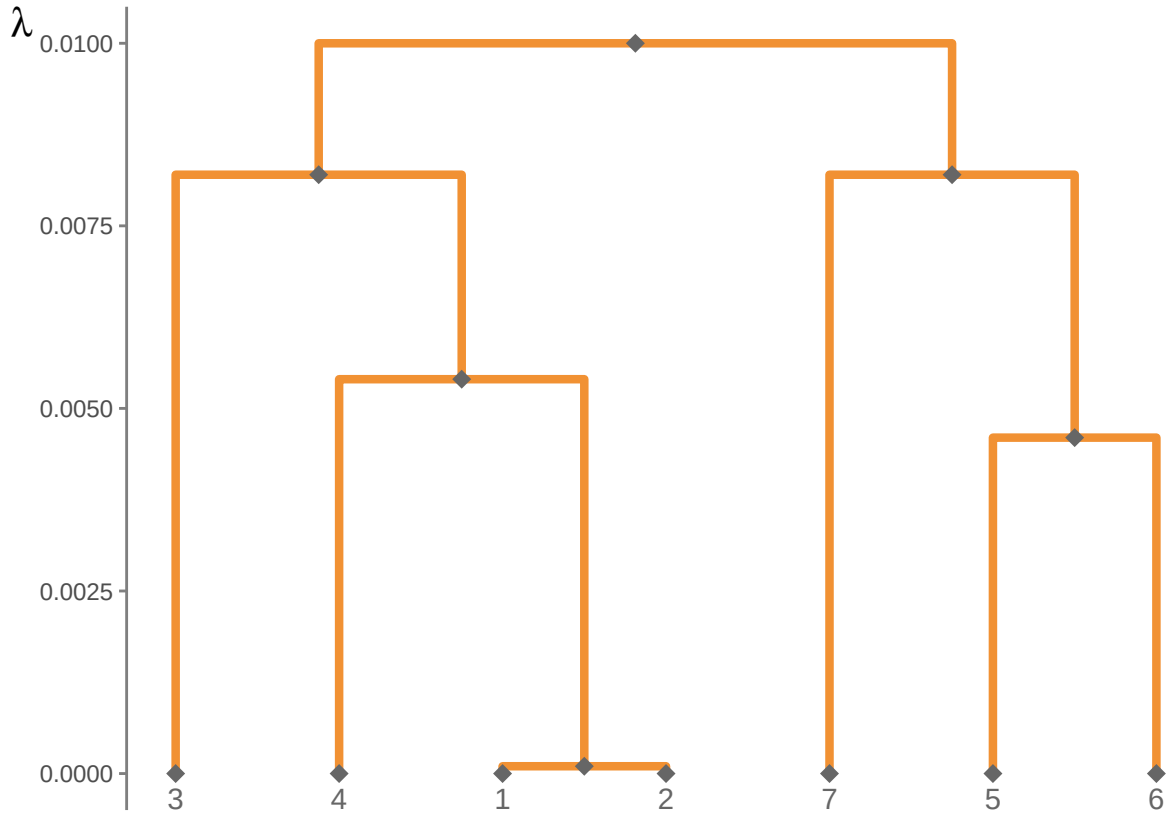

Figure 8: Hierarchical clustering of the variable according to λ.

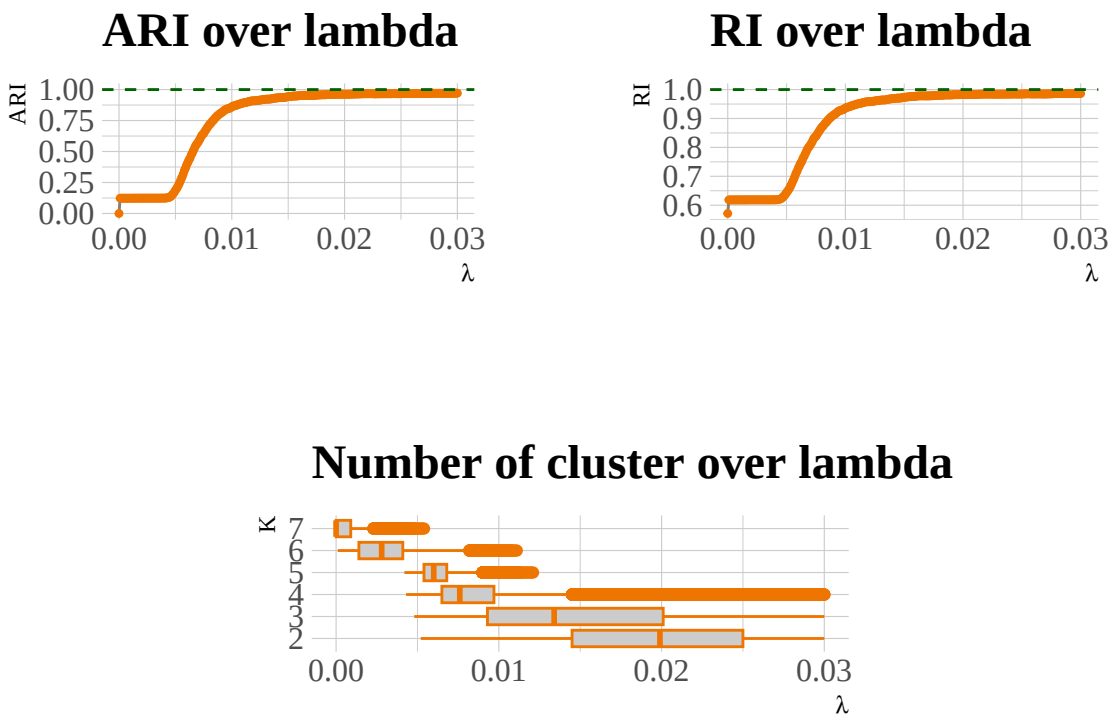

Figure 9: Summary of the optimization results with the λ.

This norm gets some interesting properties as :

- $[A|I_d] = \mathrm{tr}(A)$
- for a projection Π in a vector subspace of dimension p, then :

$$||\Pi||_F^2 = p$$

- we have a "Pythagore theorem" : if Π is a projection, then for all $A \in \mathcal{M}_d(\mathbb{R})$ we have :

$$||A||_F^2 = ||A\Pi||_F^2 + ||A(I_d - \Pi)||_F^2$$

In this paper, we focus ourself on the space of symmetric positive matrices $\mathcal{P}_d$ equipped with this scalar product.

7.2 Continuity and semi-algebraicity of eigen value function

For a symmetric matrix M of size d, we denote by $\lambda_1(M) \geq \lambda_2(M) \geq \cdots \geq \lambda_d(M)$ its eigen values.

Lemma. For each $i \in V$, we have :

$$\begin{aligned} \lambda_i : \mathcal{S}_d &\longrightarrow \mathbb{R} \\ M &\longmapsto \lambda_i(M) \end{aligned}$$

is semi-algebraic.

Proof. Indeed, if we consider the function of all the eigen value $\lambda : \mathcal{S}_d \to \mathbb{R}^d$, we have the following graph NO TO CORRECT :

$$\Gamma_\lambda = \{(\Theta, \lambda) \in \mathcal{S}_d \times \mathbb{R}^d \mid \lambda(\Theta) = \lambda\} = \{(\Theta, \lambda) \in \mathcal{S}_d \times \mathbb{R}^d \ \mid \ \forall i \in V \setminus \{d\}, \lambda_i \geq \lambda_{i+1}, \prod_{i=1}^{d} P_\Theta(\lambda_i) = 0\}$$

But we can also see P_Θ as a polynomial $P \in \mathbb{R}[X_1, \ldots, X_{\frac{d(d+1)}{2}}, X]$ where $P(\theta_1, \ldots, \theta_{\frac{d(d+1)}{2}}, X) = P_\Theta(X)$. Then we have that :

$$\Gamma_\lambda = \{x \in \mathbb{R}^{\frac{d(d+1)}{2}+d} \mid \ P(x) = 0, \ Q_k(x) \geq 0\}$$

where $P(\theta_1, \ldots, \theta_{\frac{d(d+1)}{2}}, Y_1, \ldots, Y_d) = \prod_{i=1}^{d} P_\Theta(Y_i)$ and $Q_k(\theta_1, \ldots, \theta_{\frac{d(d+1)}{2}}, Y) = Y_{k+1} - Y_k$.

We can deduce that λ is semi-algebraic. As the projection preserve the property, all the eigen functions are semi-algebraic.

□

We will continue by showing the continuity of the easier one.

Proposition. Let be :

$$\begin{aligned} \lambda_{\max} : \mathcal{P}_d &\longrightarrow \mathbb{R} \\ M &\longmapsto \lambda_{\max}(M) \end{aligned}$$

the function of the maximal eigen value. Then we have :

$$|\lambda_{\max}(M) - \lambda_{\max}(M')| \leq ||M - M'||_F$$

that means $\lambda_{\max}$ is 1-Lipschitzian.

Proof. It is know that the maximal eigen value is also the spectral distance of a symmetric positive matrix :

$$\lambda_{\max}(M) = ||M||_2 = \sup_{x \in \mathbb{R}^d} \frac{||Mx||}{||x||}$$

Thus we have :

$$|\lambda_{\max}(M) - \lambda_{\max}(M')| \leq \lambda_{\max}(M - M') = ||M - M'||_2$$

But it is know too that :

$$||M - M'||_2 \leq ||M - M'||_F$$

which is ending the proof.

□

For the other eigen values, we will use another characterization of them using the Min-max theorem.

Theorem. Let $A \in \mathcal{P}_d$ and let's denote by $\lambda_1, \geq \cdots \geq \lambda_d$ its eigen values. Then we have :

$$\lambda_k = \min_{\dim V = k-1} \max\{||Ax|| : x \perp V, ||x|| = 1\}$$

We can stress that :

- in the case where $k = 1$, it is the just the formula of the spectral norm.
- we can rewrite the expression as :

$$\lambda_{d-1} = \min_{\dim V = k-1} ||A\Pi_{V^\perp}||_2$$

Thanks to this theorem we are able to show the next proposition :

Proposition. Let be for $k \in V$:

$$\begin{aligned} \lambda_k : \mathcal{P}_d &\longrightarrow \mathbb{R} \\ M &\longmapsto \lambda_k(M) \end{aligned}$$

the function of the k-th eigen value. Then we have :

$$|\lambda_k(M) - \lambda_k(M')| \leq ||M - M'||_F$$

that means λ_k is 1-Lipschitzian for all $k \in V$.

Proof. Let $M \in \mathcal{P}_d$. From the Min-max theorem, we have $fork \in V$:

$$\lambda_k(M) = \min_{\dim V = k-1} ||M\Pi_{V^\perp}||_2$$

Put V_M the vector subspace of dimension $k-1$ such that :

$$\lambda_k(M) = ||M\Pi_{V_M^\perp}||_2$$

Thus we have :

$$\lambda_k(M') - \lambda_k(M) \leq ||M'\Pi_{V_M^\perp}||_2 - ||M\Pi_{V_M^\perp}||_2 \leq ||(M'-M)\Pi_{V_M^\perp}||_2$$

But we also have :

$$||(M'-M)\Pi_{V_M^\perp}||_2 \leq ||(M'-M)\Pi_{V_M^\perp}||_F \leq ||M'-M||_F$$

So we finally get :

$$\lambda_k(M') - \lambda_k(M) \leq ||M'-M||_F$$

We can have the same inequality by switching the role of M and M' and therefore :

$$|\lambda_k(M) - \lambda_k(M')| \leq ||M-M'||_F$$

□

7.3 Some Lipschitz results

Let's make a tour in the proof of "Lipschitzian results" for some function.

Proposition. Let E a metric space. Let be $f, g : E \to \mathbb{R}$ two functions of $\mathcal{L}^\infty(E, \mathbb{R})$, the space of bounded functions. If f and g are respectively K and L Lispchitz, then fg is $L||f||_\infty + K||g||_\infty$.

Proof. Let $x, y \in E$ then we have :

$$|f(x) - f(y)| \leq K d_E(x,y) \qquad \text{and} \qquad |g(x) - g(y)| \leq L d_E(x,y)$$

So, we can deduce that :

$$\begin{aligned} |f(x)g(x) - f(y)g(y)| &= |g(y)(f(x)-f(y)) + f(x)(g(x)-g(y))| \\ &\leq |g(y)||f(x)-f(y)| + |f(x)||g(x)-g(y)| \\ &\leq ||g||_\infty K d_E(x,y) + ||f||_\infty L d_E(x,y) \end{aligned}$$

□

Lemma. Let $a \in \mathbb{R}$. Let's define the sequence $(u_n)_{n\in\mathbb{N}}$ by :

$$\begin{cases} u_1 = 1 \\ u_{n+1} = a u_n + a^n \end{cases}$$

Then $\forall n \in \mathbb{N}, u_n = n a^{n-1}$.

Proof. By induction.

□

With these results, we can say :

Lemma. The generalised determinant application $\Theta \mapsto |\Theta|_+$ is Lipschitzian on $\bar{B}(\Theta, 1)$.

Proof. Let $\tilde{\Theta} \in \bar{B}(\Theta, 1)$.

$$|\lambda_k(\tilde{\Theta}) - \lambda_k(\Theta)| \leq 1$$

And thus :

$$\lambda_k(\tilde{\Theta}) \leq \lambda_k(\Theta) + 1 \leq \lambda_{\max}(\Theta) + 1$$

So, $||\lambda_k||_\infty \leq \lambda_{\max}(\Theta) + 1 = k_\Theta$, that is to say λ_k is bounded. Moreover, λ_k are 1-Lipschitzian.

Thus, we have by the lemma $\lambda_1 \times \lambda_2$ is $2k_\Theta$-Lipschitzian. We can apply the lemma by induction, and if $\Pi_{k=1}^m \lambda_k$ is L_m-Lipschitzian and is bounded by k_Θ^m then we get that $\Pi_{k=1}^{m+1} \lambda_k$ is L_{m+1}-Lipschitzian where $L_{m+1} = L_m k_\Theta + k_\Theta^m$.

We get the following expression for the Lipschitz constant :

$$\begin{cases} L_1 & = 1 \\ L_{m+1} & = L_m k_\Theta + k_\Theta^m \end{cases}$$

By the lemma, we have the general expression :

$$L_m = m k_\Theta^{m-1}$$

Finally, we have by induction that :

- the generalised determinant is Lipschitzian.
- the Lipschitz constant is L_d.

□

References

Engelke, Sebastian, and Adrien S. Hitz. 2020. "Graphical Models for Extremes." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 82 (4): 871–932. https://doi.org/10.1111/rssb.12355.

Engelke, Sebastian, Alexander Malinowski, Zakhar Kabluchko, and Martin Schlather. 2012. "Estimation of Huesler-Reiss Distributions and Brown-Resnick Processes." arXiv. https://doi.org/10.48550/arXiv.1207.6886.

Hentschel, Manuel, Sebastian Engelke, and Johan Segers. 2023. "Statistical Inference for Hüsler-Reiss Graphical Models Through Matrix Completions." arXiv. https://doi.org/10.48550/arXiv.2210.14292.

Hocking, Toby Dylan, Armand Joulin, Francis Bach, and Jean-Philippe Vert. n.d. "Clusterpath: An Algorithm for Clustering Using Convex Fusion Penalties."

Rootzén, Holger, and Nader Tajvidi. 2006. "Multivariate Generalized Pareto Distributions." *Bernoulli* 12 (5): 917–30. https://doi.org/10.3150/bj/1161614952.

Touw, D. J. W., A. Alfons, P. J. F. Groenen, and I. Wilms. 2024. "Clusterpath Gaussian Graphical Modeling." arXiv. https://doi.org/10.48550/arXiv.2407.00644.