A PENALIZED LEAST SQUARES ESTIMATOR FOR EXTREME-VALUE MIXTURE MODELS

Anas Mourahib, Anna Kiriliouk, Johan Segers

LIDAM Discussion Paper ISBA
2025 / 15

UCLouvain

ISBA | LIDAM
Louvain Institute of Data Analysis and Modeling in economics and statistics

ISBA

Voie du Roman Pays 20 - L1.04.01

B-1348 Louvain-la-Neuve

Email : lidam-library@uclouvain.be

https://uclouvain.be/en/research-institutes/lidam/isba/publication

arXiv:2506.15272v1 [stat.ME] 18 Jun 2025

A penalized least squares estimator for extreme-value mixture models

Anas Mourahib* Anna Kiriliouk† Johan Segers‡

June 19, 2025

Abstract

Estimating the parameters of max-stable parametric models poses significant challenges, particularly when some parameters lie on the boundary of the parameter space. This situation arises when a subset of variables exhibits extreme values simultaneously, while the remaining variables do not—a phenomenon referred to as an extreme direction in the literature. In this paper, we propose a novel estimator for the parameters of a general parametric mixture model, incorporating a penalization approach based on a pseudo-norm. This penalization plays a crucial role in accurately identifying parameters at the boundary of the parameter space. Additionally, our estimator comes with a data-driven algorithm to detect groups of variables corresponding to extreme directions. We assess the performance of our estimator in terms of both parameter estimation and the identification of extreme directions through extensive simulation studies. Finally, we apply our methods to data on river discharges and financial portfolio losses.

1 Introduction

Consider a random vector $\boldsymbol{X} = (X_1, \ldots, X_d)$ of risk factors and suppose the focus is on rare events. In this regime, neither parametric nor nonparametric methods are satisfactory: parametric models rely on distributional assumptions that can be dominated by the bulk of the data, and thus tend to underestimate extreme outcomes, while nonparametric techniques demand large sample sizes—often unavailable when studying extremes. Extreme value theory offers a rigorous framework for extrapolating beyond observed levels and has found applications in finance, hydrology, weather forecasting, and other fields.

Univariate extreme value theory, that is, when $d = 1$, has been extensively studied [8, 1, 7] and modeling rare events in this case is done via a finite-dimensional parametric

*UCLouvain, LIDAM/ISBA, anas.mourahib@uclouvain.be. Corresponding author. Supported by the *Fonds de la Recherche Scientifique – FNRS* (grant number T.0203.21).

†UNamur, NaXys, anna.kiriliouk@unamur.be.

‡KU Leuven, Department of Mathematics, jjjsegers@kuleuven.be, and UCLouvain, LIDAM/ISBA.

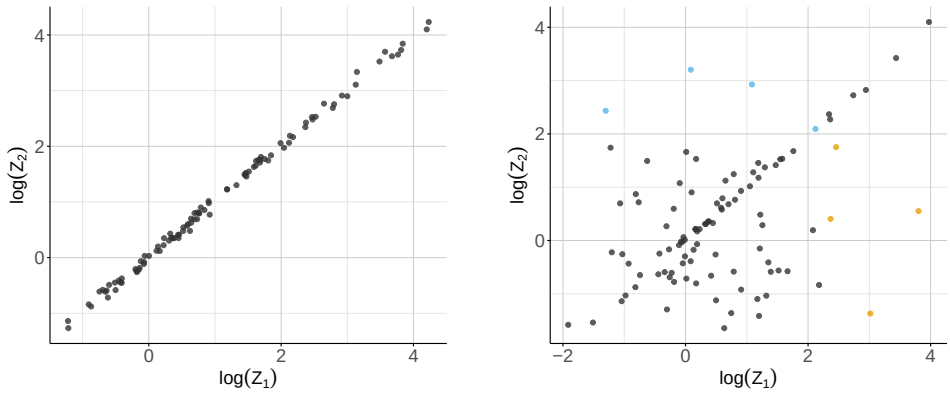

Figure 1: Log-transformed sample of size $n = 100$ from a max-stable Hüsler–Reiss model (left) and a max-stable mixture Hüsler–Reiss model (right).

family of distributions. Multivariate modeling of $d \geqslant 2$ risk factors is a more challenging topic, since risks may exhibit some sort of dependence. The goal is not only to model rare events of each risk factor independently but to capture the extremal dependence between these risk factors. Two main approaches appear in this context: the block-maxima approach and the threshold exceedances approach. In the first approach, standardized componentwise maxima of $\boldsymbol{X}$ are asymptotically modeled using a max-stable random vector $\boldsymbol{Z} = (Z_1, \ldots, Z_d)$. In the second approach, asymptotically, observations where at least one risk factor is extreme are modeled using a multivariate (generalized) Pareto random vector $\boldsymbol{Y} = (Y_1, \ldots, Y_d)$ [30].

For both approaches, many parametric families have been proposed; popular examples include the Hüsler–Reiss model and the logistic model [35, 19]. Estimation methods for such models are numerous, for example, through censored likelihood [28], weighted least squares [12], or based on neural networks [25, 29].

A drawback of most parametric models is that they force all components of $\boldsymbol{X}$ to be extreme simultaneously. Figure 1 (left) shows a log-transformed simulation from a max-stable Hüsler–Reiss model. Observations are close to the diagonal line $x = y$, suggesting that both components are large simultaneously. However, in practice, it may happen that only certain subsets $J \subset \{1, \ldots, d\}$ of variables are jointly extreme while the others are not. Such a subset J is called an *extreme direction* [27, 32].

Both the max-linear model [15, 11] and the asymmetric logistic model [35] can accommodate multiple extreme directions. However, the first model has a singular distribution (not discrete nor absolutely continuous), thus making statistical inference complicated. The second one is too specific in the sense that it cannot represent a large variety of max-stable distributions.

To capture multiple extreme directions, Mourahib et al. [27] introduce the *mixture model* that extends the classic max-linear model and can represent *any* max-stable distribution. It is identified via two sets of parameters:

- a $(d \times r)$ coefficient matrix A with entries in $[0, 1]$, whose columns correspond to distinct extreme directions of $\boldsymbol{X}$. Zero entries in A indicate which variables are not contained in a given extreme direction [27, Proposition 4.5];
- a second set of parameters that governs the dependence structure *within* each extreme direction.

A simulation example of this model is given in Figure 1 (right), where we can see that this model, in fact, captures not only the extreme direction $\{1, 2\}$, but also $\{1\}$ and $\{2\}$.

A natural question is how to estimate the model parameters. The weighted least-squares estimator of Einmahl et al. [12] requires the true parameters to be in the interior of the parameter space. In contrast, the mixture model's reliance on zero entries in A pushes parameters to the boundary of the space, violating this condition.

We propose an approach for the estimation of the parameters of max-stable distributions possessing any collection of extreme directions, using a penalized version of the least squares estimator proposed in [12]. While the focus in Mourahib et al. [27] is on the multivariate generalized Pareto density, the estimator studied in this work is based on the stable tail dependence function, since estimating the mixture model parameters from the former is computationally burdensome in high or even moderate dimensions because one would need to censor components that are below the threshold.

The proposed penalized least squares estimator is capable of identifying extreme directions *and* estimating the mixture model parameters simultaneously. This differentiates our approach from current literature which focuses solely on the identification of extreme directions [17, 32, 26].

Outline of the paper. The paper is structured as follows. Section 2 provides essential background on extreme value theory, focusing in particular on the notions of extreme directions [32], the mixture model [27] and the least-squares estimator [12]. In Section 3, we introduce a penalized estimator for the mixture model and subsequently develop a data-driven algorithm to identify the extreme directions present in a given sample. As our estimator involves tuning parameters, we demonstrate that their selection effectively reduces to the choice of a single penalization parameter, which we determine via cross-validation. Finally, we illustrate the practical relevance of our approach by applying the algorithm, first to river discharge data across stations from the Danube basin, and second to financial industry portfolios constructed from stocks listed on the NYSE, AMEX, and NASDAQ. Finally, we compare our results with those obtained by applying algorithms from the literature, in particular the DAMEX algorithm introduced by Goix et al. [17].

Notation. Let d be a positive integer and write $D = \{1, \ldots, d\}$. Throughout, bold symbols will refer to multivariate quantities. Let $\mathbf{0}$ and $\mathbf{1}$ denote vectors of zeroes and ones, respectively, of a dimension clear from the context. Let $\boldsymbol{x} = (x_1, \ldots, x_d)$ denote a d-dimensional vector and, for a non-empty set $J \subseteq \{1, \ldots, d\}$, write $\boldsymbol{x}_J = (x_j)_{j \in J}$. Mathematical operations on vectors such as addition, multiplication and comparison are considered component-wise. For two vectors $\boldsymbol{a}$, $\boldsymbol{b}$, the set $[\boldsymbol{a}, \boldsymbol{b}]$ will denote the Cartesian product $\prod_{j=1}^{d} [a_j, b_j]$ and so forth for $(\boldsymbol{a}, \boldsymbol{b})$, $[\boldsymbol{a}, \boldsymbol{b})$, $(\boldsymbol{a}, \boldsymbol{b}]$. For two integers d_1, $d_2 > 0$

and a set $\mathbb{S} \subset \mathbb{R}^+ = [0, \infty)$, let $\mathcal{M}_{d_1, d_2}(\mathbb{S})$ denote the set of $d_1 \times d_2$ matrices with entries in $\mathbb{S}$. The arrow $\rightsquigarrow$ denotes convergence in distribution (weak convergence). For two vectors $\mathbf{x}$ and $\mathbf{y}$ in $\mathbb{R}^d$, define the *lexicographic order* $\leqslant_{\text{lex}}$ as

$$\mathbf{x} \leqslant_{\text{lex}} \mathbf{y} \quad \text{if and only if} \quad x_j < y_j \text{ for the first index } j \in D \text{ for which } x_j \neq y_j. \tag{1.1}$$

The lexicographic order is a total order in $\mathbb{R}^d$, meaning that either $\boldsymbol{x} \leqslant_{\text{lex}} \boldsymbol{y}$ or $\boldsymbol{y} \leqslant_{\text{lex}} \boldsymbol{x}$. For a set S, let $S^n = S \times \cdots \times S$ (n times). Finally, let $\mathbb{E} := [0, \infty)^d \setminus \{\mathbf{0}\}$.

2 Background

2.1 Multivariate extreme value distributions

For $i = 1, \ldots, n$, let $\boldsymbol{X}_i = (X_{i1}, \ldots, X_{id})$ be i.i.d. copies of $\boldsymbol{X}$, a random vector on $\mathbb{R}^d$ with joint cumulative distribution function (cdf) F and margins $F_1, \ldots, F_d$. We say that F is in the (max-)domain of attraction of a *max-stable* distribution G if there exist sequences $\boldsymbol{a}_n \in (0, \infty)^d$ and $\boldsymbol{b}_n \in \mathbb{R}^d$ such that

$$\frac{\max_{i=1,\ldots,n} \boldsymbol{X}_i - \boldsymbol{b}_n}{\boldsymbol{a}_n} \rightsquigarrow G, \qquad n \to \infty. \tag{2.1}$$

The margins of G, denoted $G_1, \ldots, G_d$, are univariate extreme-value distributions themselves, and any max-stable distribution G admits the representation

$$G(\boldsymbol{x}) = \exp\left\{-\ell\left(-\ln G_1(x_1), \ldots, -\ln G_d(x_d)\right)\right\},$$

where $\ell : [0, \infty)^d \to [0, \infty)$ is the *stable tail dependence function (stdf)*. Then the distribution of $\boldsymbol{X}^* = (X_1^*, \ldots, X_d^*)$ with $X_j^* = 1/\left\{1 - F_j(X_j)\right\}$ for $j \in D$ is in the domain of attraction of $G^*(\boldsymbol{x}) = \exp\left\{-\ell\left(1/x_1, \ldots, 1/x_d\right)\right\}$, a max-stable distribution with unit-Fréchet margins.

The stable tail dependence function can be retrieved via

$$\ell(\boldsymbol{x}) = \lim_{t \downarrow 0} t^{-1} \mathbb{P}\left[F_1(X_1) \geqslant 1 - tx_1 \text{ or } \ldots \text{ or } F_d(X_d) \geqslant 1 - tx_d\right]. \tag{2.2}$$

The function ℓ is convex and homogeneous in the sense that for $c \geqslant 0$ and $\boldsymbol{x} \in [0, \infty)^d$, we have $\ell(c\boldsymbol{x}) = c\ell(\boldsymbol{x})$. Moreover, unit-Fréchet margins of G^* imply that $\ell(0, \ldots, 0, x_j, 0, \ldots, 0) = x_j$ for $x_j \geqslant 0$ and $j \in D$. Finally, ℓ can be bounded by

$$\max(x_1, \ldots, x_d) \leqslant \ell(x) \leqslant x_1 + \cdots + x_d, \tag{2.3}$$

where the lower bound corresponds to perfect tail dependence and the upper bound to asymptotic independence.

Suppose that $\boldsymbol{Z} = (Z_1, \ldots, Z_d)$ follows a max-stable distribution with unit-Fréchet margins. For non-empty $J \subseteq D$, let ℓ_J be the stdf associated with $\boldsymbol{Z}_J$, that is,

$$\ell_J(\boldsymbol{x}_J) = \ell\left(\tilde{\boldsymbol{x}}\right), \qquad \boldsymbol{x}_J \in [0, \infty)^{|J|},$$

with $\tilde{\boldsymbol{x}} = \sum_{j\in J} x_j \boldsymbol{e}_j$ and $\boldsymbol{e}_j$ the j-th canonical unit vector in $\mathbb{R}^d$. The *extremal coefficients* [31] are defined via $\zeta_J := \ell(\sum_{j\in J} \boldsymbol{e}_j)$, for set $J \subseteq D$ with $|J| \geqslant 2$. From (2.3), we get that $1 \leqslant \zeta_J \leqslant |J|$. The value ζ_J can be interpreted as the effective number of tail independent variables among $\boldsymbol{Z}_J$.

The *exponent measure* μ of G^* is the unique Borel measure concentrated on $\mathbb{E} = [0,\infty)^d \setminus \{\boldsymbol{0}\}$ such that

$$\mu\left(\mathbb{E} \setminus [\boldsymbol{0}, 1/\boldsymbol{x}]\right) = \ell(\boldsymbol{x}), \qquad \boldsymbol{x} \in [\boldsymbol{0}, \boldsymbol{\infty})^d. \tag{2.4}$$

The measure μ is finite on Borel sets that are bounded away from the origin. Moreover, $\mu\left(\mathbb{E} \setminus [\boldsymbol{0}, \boldsymbol{y}]\right) = \infty$ as soon as $y_j = 0$ for some $j \in D$.

2.2 Extreme directions and the mixture model

For non-empty $J \subseteq D$, define $\mathbb{E}_J := \{\boldsymbol{x} \in \mathbb{E} : \ x_j > 0 \text{ iff } j \in J\}$. Then $\{\mathbb{E}_J : \ \varnothing \neq J \subseteq D\}$ forms a partition of $\mathbb{E}$. Let $\boldsymbol{X}^*$ be again a d-dimensional random vector in the domain of attraction of a max-stable distribution with unit-Fréchet margins and exponent measure μ. The following definition was introduced in Simpson et al. [32] in order to identify *extreme directions* of $\boldsymbol{X}^*$. Intuitively, an extreme direction is a group of components of $\boldsymbol{X}^*$ that may take on large values simultaneously while the other components are of smaller order.

Definition 2.1. The set J is said to be an extreme direction of $\boldsymbol{X}^*$ or μ if $\mu(\mathbb{E}_J) > 0$.

Note that μ has no mass outside $\bigcup_{J\in\mathcal{R}(\mu)} \mathbb{E}_J$, where $\mathcal{R}(\mu)$ is the collection of extreme directions of μ. Further, even if a certain set $J \subseteq D$ is not an extreme direction of $\boldsymbol{X}^*$, it is still possible that J is a superset or a subset of an extreme direction.

Many standard max-stable models are only appropriate for data exhibiting a single extreme direction $J = D$. Notable exceptions are the asymmetric logistic [34] and the max-linear [36, 11] models. [27] introduce the *mixture model*, a smoothed version of the max-linear model, which allows any collection of non-empty subsets of D that together covers D, so that no element of D is left out, to be the extreme directions of μ. The mixture model is key to modeling data with non-standard dependence structures as it covers *all* max-stable distributions.

Definition 2.2 (Mourahib et al. [27])**.** Let r be a positive integer and write $R = \{1, \ldots, r\}$. Let $\boldsymbol{Z}_{\bullet 1}, \ldots, \boldsymbol{Z}_{\bullet r}$ be independent random vectors in $\mathbb{R}^d$. For all $s \in R$, suppose that $\boldsymbol{Z}_{\bullet s} = (Z_{1s}, \ldots, Z_{ds})$ are max-stable random vectors with unit-Fréchet margins, stdf $\ell^{(s)}$ and a single extreme direction D. Let $A = (a_{js})_{j\in D; s\in R}$ be a $(d \times r)$ coefficient matrix with $a_{js} \in [0,1]$ for all $j \in D$ and $s \in R$, unit row sums $\sum_{s=1}^r a_{js} = 1$ for all $j \in D$ and positive column sums $\sum_{j=1}^d a_{js} > 0$ for all $s \in R$. The d-variate random vector

$$\boldsymbol{M} = (M_1, \ldots, M_d) = \left(\max_{s\in R}\{a_{1s} Z_{1s}\}, \ldots, \max_{s\in R}\{a_{ds} Z_{ds}\}\right), \tag{2.5}$$

will be called the $(d \times r)$ *mixture model* with coefficient matrix A and factors $\boldsymbol{Z}_{\bullet 1}, \ldots, \boldsymbol{Z}_{\bullet r}$.

Remark 2.3. Note that two different coefficient matrices, containing the same columns but in different orders, will lead to the same mixture model $\boldsymbol{M}$ if the distributions of $\boldsymbol{Z}_{\bullet 1}, \ldots, \boldsymbol{Z}_{\bullet r}$ are identical. To ensure identifiability, we will always assume that the columns of A are decreasingly ordered according to the lexicographic order defined in equation (1.1).

In (2.5), for a given $j \in D$, only those $s \in R$ that satisfy $a_{js} > 0$ matter for the value of the maximum. Let

$$J_s := \{j \in D : \ a_{js} > 0\}, \tag{2.6}$$

denote the s-th *signature* of the coefficient matrix A.

Theorem 2.4 (Stephenson [33], Mourahib et al. [27])**.** *The random vector $\boldsymbol{M}$ in* (2.5) *follows a max-stable distribution $G_{\boldsymbol{M}}$ with unit-Fréchet margins and stdf*

$$\ell(\boldsymbol{x}) = \sum_{s \in R} \ell_{J_s}^{(s)} \left((a_{js} x_j)_{j \in J_s} \right), \qquad \boldsymbol{x} \in [0, \infty)^d. \tag{2.7}$$

Zero entries of the coefficient matrix A allow some groups of components of the random vector $\boldsymbol{M}$ to constitute an extreme direction. Mourahib et al. [27, Proposition 4.5] show that the set of extreme directions of the mixture model is given by the signatures of its coefficient matrix A. The model is general in the sense that any max-stable distribution with unit-Fréchet margins is the cdf of a mixture model whose coefficient matrix has mutually different signatures [27, Theorem 4.8].

In the rest of the paper, we will assume that an extreme direction only corresponds to a single column of the coefficient matrix A, i.e., we will assume that the signatures of A are mutually different.

Example 2.5 (Mixture logistic model)**.** For $s \in R$, suppose that the stdf associated with $\boldsymbol{Z}_{J_s,s}$ is

$$\ell_{J_s}^{(s)}(x_1, \ldots, x_{|J_s|}) = \left(x_1^{1/\alpha_s} + \cdots + x_{|J_s|}^{1/\alpha_s} \right)^{\alpha_s}, \tag{2.8}$$

for $\boldsymbol{x} \in [0, \infty)^{|J_s|}$, where $0 < \alpha_s < 1$. The stdf of the *mixture logistic model* $\boldsymbol{M}$ is

$$\ell(\boldsymbol{x}) = \sum_{s \in R} \left\{ \sum_{j \in J_s} (a_{js} x_j)^{1/\alpha_s} \right\}^{\alpha_s}, \qquad \boldsymbol{x} \in [0, \infty)^d. \tag{2.9}$$

Example 2.6 (Mixture Hüsler–Reiss model)**.** Let $s \in R$. For $|J_s| > 1$, suppose that $\ell_{J_s}^{(s)}$ is the Hüsler–Reiss stdf associated with the $(|J_s| \times |J_s|)$ variogram matrix[1] $\Gamma^{(s)}$; see, e.g., Huser and Davison [18]. That is,

$$\ell_{J_s}^{(s)}(\boldsymbol{x}) = \sum_{j \in J_s} x_j \, \Phi_{|J_s|-1} \left(\boldsymbol{\eta}^{j,(s)}(\boldsymbol{x}) \, ; \Sigma^{j,(s)} \right), \qquad \boldsymbol{x} \in [0, \infty)^{J_s},$$

[1] A matrix $\Gamma \in \mathbb{R}^{d \times d}$ is a variogram matrix if it is symmetric, has zero diagonal, and satisfies $\boldsymbol{v}^\top \Gamma \boldsymbol{v} < 0$ for all $\boldsymbol{0} \neq \boldsymbol{v}^\top \perp \boldsymbol{1}$. To each positive definite covariance matrix $\Sigma \in \mathbb{R}^{d \times d}$ is associated a variogram matrix $\Gamma \in \mathbb{R}^{d \times d}$ via $\Gamma_{st} = \Sigma_{ss} + \Sigma_{tt} - 2\Sigma_{st}$ for $s, t \in D$.

where for $j \in J_s$, $\boldsymbol{\eta}^{j,(s)}(\boldsymbol{x}) := \left\{\ln(x_j/x_t) + \Gamma_{jj}^{(s)}/2\right\}_{t\in J_s; t\neq j}$ where by convention, $0/0$ is set to ∞, and where the $(|J_s \setminus \{j\}| \times |J_s \setminus \{j\}|)$ matrix $\Sigma^{j,(s)}$ is defined by

$$\Sigma^{j,(s)} := \frac{1}{2}\left\{\Gamma_{jt}^{(s)} + \Gamma_{jt'}^{(s)} - \Gamma_{tt'}^{(s)}\right\}_{t,t'\in J_s\setminus\{j\}},$$

and finally, where $\Phi_m(\,\cdot\,;\Sigma)$ is the cdf of the centered m-variate normal distribution with covariance matrix Σ. The stdf of the mixture model $\boldsymbol{M}$ is

$$\ell(\boldsymbol{x}) = \sum_{s\in R}\left\{\sum_{j\in J_s} a_{js}x_j\,\Phi_{|J_s|-1}\left(\boldsymbol{\eta}^{j,(s)}\left(\left(a_{js}x_j\right)_{j\in J_s}\right);\Sigma^{j,(s)}\right)\right\}, \qquad \boldsymbol{x}\in[0,\infty)^d. \quad (2.10)$$

2.3 Least squares estimator of tail dependence

Let again $\boldsymbol{X}_1,\ldots,\boldsymbol{X}_n$ denote i.i.d. random vectors in the domain of attraction of a max-stable distribution with stdf ℓ. We will first recall how to estimate ℓ non-parametrically. Let $k = k_n \in (0,n]$ be an intermediate sequence such that

$$k\to\infty \quad \text{and} \quad k/n\to 0, \quad \text{as } n\to\infty. \quad (2.11)$$

The k determines the fraction of the data that is considered as extreme and that will be used for inference. We will refer to the quantity k/n as the *tail fraction*. A straightforward non-parametric estimator of the stdf is

$$\hat{\ell}_{n,k}(\boldsymbol{x}) = \frac{1}{k}\sum_{i=1}^{n}\mathbb{1}\left\{R_{i1}^n > n+1/2-kx_1 \text{ or } \ldots \text{ or } R_{id}^n > n+1/2-kx_d\right\}, \quad (2.12)$$

where R_{ij}^n denotes the rank of X_{ij} among $X_{1j},\ldots,X_{nj}$ for $j=1,\ldots,d$. Note that (2.12) is a minor modification of the standard empirical stdf [10]. Alternatives are the bias-corrected estimator proposed in Beirlant et al. [2] or a smooth estimator based on the empirical beta copula [24].

Suppose now that ℓ belongs to a parametric family $\{\ell(\,\cdot\,;\boldsymbol{\theta}) : \boldsymbol{\theta}\in\Theta\}$, with $\Theta\subset\mathbb{R}^p$ for some integer $p\geqslant 1$. Furthermore, assume the existence of a unique parameter $\boldsymbol{\theta}_0\in\Theta$ such that $\ell(\boldsymbol{x}) = \ell(\boldsymbol{x};\boldsymbol{\theta}_0)$ for all $\boldsymbol{x}\in[0,\infty)^d$. Einmahl et al. [12] propose a continuous updating weighted least-squares estimator for $\boldsymbol{\theta}_0$, which, in its simplest form, reduces to the ordinary least-squares estimator

$$\hat{\boldsymbol{\theta}}_{n,k} := \operatorname*{arg\,min}_{\boldsymbol{\theta}\in\Theta}\sum_{m=1}^{q}\left(\hat{\ell}_{n,k}(\boldsymbol{c}_m) - \ell(\boldsymbol{c}_m;\boldsymbol{\theta})\right)^2. \quad (2.13)$$

Here, $\boldsymbol{c}_1,\ldots,\boldsymbol{c}_q\in[0,\infty)^d$, with $\boldsymbol{c}_m = (c_{m1},\ldots,c_{md})$ for $m=1,\ldots,q$ are the q points in which ℓ and $\hat{\ell}_{n,k}$ are evaluated. The estimator (2.13) is easy to compute and shows good performance for many well-known max-stable models [12, Section 3].

3 Estimation of the mixture model

In Section 3.1, we will start by introducing our estimation procedure for the simplified setting where the number of extreme directions, r, is assumed to be known. The case where r is unknown will be discussed further in Section 3.2.

3.1 Known number of extreme directions

As in Definition 2.2, let $\boldsymbol{M}$ be the $d \times r$ mixture model with coefficient matrix A and factors $\boldsymbol{Z}_{\bullet 1}, \ldots, \boldsymbol{Z}_{\bullet r}$ with stdfs $\ell^{(1)}, \ldots, \ell^{(r)}$. Recall that $R = \{1, \ldots, r\}$. From now on, suppose that the stdfs $\ell^{(1)}, \ldots, \ell^{(r)}$ belong to a common parametric family $\{\ell(\,\cdot\,; \boldsymbol{\theta}_Z) : \boldsymbol{\theta}_Z \in \Theta_Z\}$, where $\Theta_Z \subset \mathbb{R}^v$ for some integer $v \geqslant 1$, and that for each $s \in R$, there exists a unique parameter $\boldsymbol{\theta}_Z^{(s)} \in \Theta_Z$ such that $\ell^{(s)} = \ell(\,\cdot\,; \boldsymbol{\theta}_Z^{(s)})$. In what follows, for simplicity, we assume that the dependence parameters across the columns are identical, meaning that all the $\boldsymbol{\theta}_Z^{(s)}$ are equal for $s \in R$. However, all the results remain valid in the general case. Let $\ell(\cdot\,; \boldsymbol{\theta})$ be the stdf associated with $\boldsymbol{M}$ as in Theorem 2.4. More specifically, let $\boldsymbol{\theta} = (A, \boldsymbol{\theta}_Z)$, where we recall $A \in \Theta_A := \mathcal{M}_{d,r}([0,1])$ and $\boldsymbol{\theta}_Z = \boldsymbol{\theta}_Z^{(1)} \in \Theta_Z \subset \mathbb{R}^v$. Note that A can equivalently be represented as a vector $\boldsymbol{\theta}_A \in [0,1]^{d \times r}$ by stacking its rows sequentially. For simplicity of notation, we will use A and $\boldsymbol{\theta}_A$ interchangeably throughout the paper.

The *preliminary non-standardized penalized least-squares (PNPLS) estimator* is defined as

$$\hat{\boldsymbol{\theta}}_{\mathrm{pr}} = \left(\hat{A}_{\mathrm{pr}}\,, \hat{\boldsymbol{\theta}}_{Z,\mathrm{pr}}\right) = \underset{\boldsymbol{\theta}=(A,\boldsymbol{\theta}_Z)\in(\Theta_A,\Theta_Z)}{\arg\min} \left\{ \sum_{m=1}^{q} \left(\hat{\ell}_{n,k}(\boldsymbol{c}_m) - \ell(\boldsymbol{c}_m; \boldsymbol{\theta})\right)^2 + \lambda \mathcal{P}(A) \right\}, \quad (3.1)$$

where $\lambda > 0$ is called the *penalization parameter* and $\mathcal{P}(A) = \sum_{j \in D} \left(\sum_{s \in R} a_{js}^p\right)^{1/p}$, for some *penalization exponent* $p \in (0,1)$.

For $\lambda = 0$, the PNPLS estimator is equal to the ordinary least-squares estimator defined in (2.13). The penalization term $\lambda P(A)$ in (3.1) is needed to find zero entries in the coefficient matrix A. We restrict our attention to $p \leqslant 1$. Geometrically, using a penalization $\mathcal{P}(A)$ with $p \leqslant 1$ promotes sparsity because its level sets are non-round and tend to intersect the loss contours at sparse solutions—similar to the well-known Lasso case when $p = 1$. This is in contrast to the case of $p > 1$, as illustrated in Figure 2. In Remark 3.3, we present some additional arguments for the choice $p \leqslant 1$ related to the minimization algorithm.

The minimization problem in (3.1) is non-convex and is not guaranteed to give a global minimum. The parameter (vector) $\boldsymbol{\theta}_Z$, corresponding to the latent factors $\boldsymbol{Z}_{\bullet 1}, \ldots, \boldsymbol{Z}_{\bullet r}$, may be particularly difficult to estimate. A straightforward way to improve the PNPLS estimator is as follows. After obtaining $\hat{\boldsymbol{\theta}}_{\mathrm{pr}}$, we follow-up with a two-step estimation procedure, where each step minimizes (3.1) with respect to either $\boldsymbol{\theta}_Z \in \Theta_Z$ or $A \in \Theta_A$. More precisely,

1. let $\hat{A}_{\mathrm{new}}$ be the partial solution of (3.1) w.r.t. A, where $\boldsymbol{\theta}_Z$ is fixed to $\hat{\boldsymbol{\theta}}_{Z,\mathrm{pr}}$;

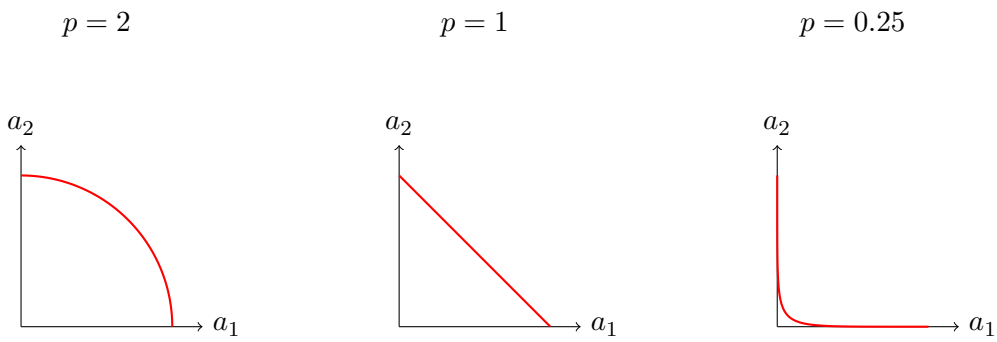

Figure 2: Bivariate contours of $(\Sigma_s a_s^p)^{1/p} = 1$ for given values of p.

2. write $\hat{A}_{\text{new}} = (\hat{a}_{11,\text{new}}, \ldots, \hat{a}_{d1,\text{new}}, \ldots, \hat{a}_{1r,\text{new}}, \ldots, \hat{a}_{dr,\text{new}})$ and standardize to ensure unit row sums,

$$\hat{a}_{js} = \hat{a}_{js,\text{new}} \Big/ \sum_{l=1,\ldots,r} \hat{a}_{jl,\text{new}}, \qquad j \in D, s \in R. \tag{3.2}$$

   This yields $\hat{A} \in \mathcal{M}_{d,r}([0,1])$ with unit row sums;

3. let $\hat{\boldsymbol{\theta}}_Z$ be the partial solution of (3.1) w.r.t. $\boldsymbol{\theta}_Z$, where A is fixed to $\hat{A}_{\text{new}}$.

The resulting estimate $\hat{\boldsymbol{\theta}} = (\hat{A}, \hat{\boldsymbol{\theta}}_Z)$ will be called the *penalized least-squares (PLS) estimator*.

The two steps of the estimation procedure can also be iteratively alternated until a predefined stopping criterion is met. However, as we will demonstrate in the simulation studies in Section 4, performing these steps only once is typically sufficient to achieve satisfactory results in practice. The full estimation procedure is explained in Algorithm 1.

Remark 3.1. Since A has unit row sums by definition, one can argue that the minimization problem (3.1) can be reduced to a lower-dimensional formulation where $\Theta_A = [0,1]^{d\times(r-1)}$ by eliminating the column $\boldsymbol{a}_{\cdot r}$ and imposing the constraint $a_{j1} + \cdots + a_{jr-1} \leqslant 1$, for each $j \in D$. However, such an approach penalizes only the first $r-1$ columns of the matrix A, leading to an asymmetric treatment of the parameters.

Remark 3.2. Theorem 2.4 states that the extreme directions of a mixture model correspond to the set of signatures of its coefficient matrix A. For $s \in R$, the sets $\hat{J}_s = \{j \in D : \hat{a}_{js} > 0\}$ thus explicitly provide estimates of the extreme directions associated with the data.

Minimization algorithm and choice of p. To solve the minimization problem (3.1), we use the bounded limited-memory modification of the BFGS quasi-Newton method (`L-BFGS-B` algorithm in `R`), see Byrd et al. [6]. This iterative method has the update step

$$\boldsymbol{\theta}_{m+1} = \boldsymbol{\theta}_m - \alpha_m H_m^{-1} \nabla L(\boldsymbol{\theta}_m), \qquad m \geqslant 0, \tag{3.4}$$

Algorithm 1 Estimation of the parameters of a mixture model for known r

Input: integers d, $r \geqslant 1$, points $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_q \in [0, \infty)^d$, sample $\mathcal{S} = \boldsymbol{X}_1, \ldots, \boldsymbol{X}_n \in [0, \infty)^d$, sample size $n \in \mathbb{N}$, integer $k \leqslant n$, reals $0 < p \leqslant 1$ and $\lambda > 0$.

Output: penalized least-squares estimator $\hat{\boldsymbol{\theta}}$.

1: Compute the empirical stdf estimates $\left(\hat{\ell}_{n,k}(\boldsymbol{c}_m)\right)_{m=1,\ldots,q}$ based on the sample $\mathcal{S}$.

2: **Preliminary non-standardized penalized least-squares estimator**: Compute the parameter estimate $\hat{\boldsymbol{\theta}}_{\mathrm{pr}} = \left(\hat{A}_{\mathrm{pr}}, \hat{\boldsymbol{\theta}}_{Z,\mathrm{pr}}\right) \in \Theta_A \times \Theta_z$ as the solution of (3.1) based on empirical stdf estimates calculated in Step 1.

3: Set $\hat{\boldsymbol{\theta}}_{\mathrm{old}} := (\hat{A}_{\mathrm{old}}, \hat{\boldsymbol{\theta}}_{Z,\mathrm{old}})$ to $\hat{\boldsymbol{\theta}}_{\mathrm{pr}}$.

4: Initialize stopping criterion to **FALSE**.

5: **while** stopping criterion is **FALSE do**

6: **Step 1. Update the coefficient matrix**: Compute $\hat{A}_{\mathrm{new}}$ as the solution of (3.1) with $\boldsymbol{\theta}_Z$ fixed at $\boldsymbol{\theta}_{Z,\mathrm{old}}$ and using the starting value $\hat{A}_{\mathrm{old}}$.

7: **Step 2. Standardization**: Apply the transformation (3.2) to $\hat{\boldsymbol{\theta}}_{A,\mathrm{new}}$ to ensure unit row sums.

8: **Step 3. Update the dependence parameter**: Compute $\hat{\boldsymbol{\theta}}_{Z,\mathrm{new}}$ as the solution of (3.1) with A fixed at $\hat{A}_{\mathrm{new}}$ and using the starting value $\hat{\boldsymbol{\theta}}_{Z,\mathrm{old}}$.

9: Check the stopping criterion; see the discussion on stopping criteria immediately following the algorithm. If verified, set stopping criterion to **TRUE**. Else, set $\hat{\boldsymbol{\theta}}_{\mathrm{old}}$ to $\hat{\boldsymbol{\theta}}_{\mathrm{new}} = (\hat{A}_{\mathrm{new}}, \hat{\boldsymbol{\theta}}_{Z,\mathrm{new}})$.

10: **end while**

11: **Standardized penalized least-squares estimator:** Define

$$\hat{\boldsymbol{\theta}} := \left(\hat{A}, \hat{\boldsymbol{\theta}}_Z\right) = \left(\hat{A}_{\mathrm{new}}, \hat{\boldsymbol{\theta}}_{Z,\mathrm{new}}\right). \tag{3.3}$$

12: **return** $(\hat{\boldsymbol{\theta}})$.

where L is the loss function in (3.1), $\alpha_m > 0$ is the step-size, H_m is a $(d \times r + v) \times (d \times r + v)$ approximate curvature positive definite matrix capturing second-order information based on the gradients observed during optimization, and finally, $\nabla L(\boldsymbol{\theta}_m)$ is the gradient of L evaluated at $\boldsymbol{\theta}_m$. Here, at each step, the gradient is approximated using finite differences method.

Remark 3.3. An additional motivation for our choice of penalization exponent $p \in (0, 1]$ is the fact that the update direction in (3.4) is opposite to the gradient. It follows that for a fixed parameter a_{js}, the penalization term in (3.1) contributes to the gradient proportionally to

$$T(p) = a_{js}^{p-1} \left(\sum_{t \in R} a_{jt}^p \right)^{1/p-1}. \tag{3.5}$$

Recall that $p < 1$ and thus $p - 1 < 0$. Consequently, if the true value of a_{js} is zero, then as the corresponding coefficient approaches zero at some step of the minimization

algorithm L-BFGS-B, the term $T(p)$ in Equation (3.5) diverges to $+\infty$. As a result, the gradient also diverges to $+\infty$, and since the update direction in (3.4) is opposite to the gradient, this leads to a solution where $\hat{a}_{js} = 0$.

Stopping criterion. The stopping criterion in Step 9 of Algorithm 1 plays a crucial role in ensuring convergence while preventing unnecessary computations. Here, we adopt a criterion based on parameter improvement, terminating the iterations when the change between successive parameter estimates becomes sufficiently small:

$$\|\hat{\boldsymbol{\theta}}_{\text{new}} - \hat{\boldsymbol{\theta}}_{\text{old}}\|_{d\times r+v} < \epsilon,$$

where ϵ is a predefined small threshold, $\hat{\boldsymbol{\theta}}_{\text{new}}$ and $\hat{\boldsymbol{\theta}}_{\text{old}}$ are as defined in Algorithm 1, and $\|\cdot\|_{d\times r+v}$ is the Euclidean norm. Additionally, to prevent infinite loops, we impose a hard limit on the number of iterations. Specifically, the algorithm is forced to stop if it reaches a predefined upper bound max.

Choice of tuning parameters. There are three tuning parameters for the estimation procedure in Algorithm 1: the tail fraction k/n, the penalization exponent p and the penalization parameter λ. First, the choice of the tail fraction is a classical trade-off between bias and variance; a smaller k/n or larger k/n introduces more variance or more bias, respectively, for the empirical stdf in (2.12). Second, for $0 < p_1 \leqslant p_2 < 1$, we have $T(p_1)/T(p_2)$ explodes to ∞ when a_{js} is closer to zero, with T defined in (3.5). The same reasoning after (3.5) shows that a smaller penalization exponent yields stronger penalization. Different choices of p will be illustrated in Section 4. Finally, the penalization parameter λ will be chosen using a K-fold cross validation procedure as explained in Algorithm 2.

3.2 Identifying the extreme directions

In practice, the number of extreme directions r is unknown. Using the same assumptions and notation as in Section 3.1, we now estimate $\boldsymbol{\theta}$ without taking any prior knowledge of the number of columns, r, of A into account. Our proposed method is iterative: at each step $t \geqslant 1$, we fix the number of columns to t and compute the standardized penalized least-squares estimator $\hat{\boldsymbol{\theta}} = \left(\hat{A}, \hat{\boldsymbol{\theta}}_Z\right) \in \Theta_A \times \Theta_Z$ using Algorithm 1. We then iterate to $t+1$ as long as all sub-vectors $(a_{1s}, \ldots, a_{ds})$ of $\boldsymbol{\theta}_A = (a_{11}, \ldots, a_{d1}, \ldots, a_{1t}, \ldots, a_{dt})$ remain non-null for all $s = 1, \ldots, t$. The method is presented in Algorithm 3.

The *DAMEX* algorithm introduced by Goix et al. [17, Algorithm 1] can also be used to identify extreme directions. It relies on three key parameters: a threshold beyond which data are considered extreme, analogous to the tail fraction k/n used in Algorithm 1; a tolerance parameter that heuristically defines the region associated with an extreme direction; and a mass threshold, which determines whether a subset $J \subset V$ with sufficiently large mass is considered an extreme direction. As explained in Goix et al. [17], in a supervised setting, these parameters are selected via cross-validation. As we will detail in the simulation study in Section 4, this is also the case for Algorithm 1:

Algorithm 2 K-fold cross validation score for a penalization parameter λ

Input: integers d, $r \geqslant 1$, points $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_q \in [0, \infty)^d$, sample $\mathcal{S} = \boldsymbol{X}_1, \ldots, \boldsymbol{X}_n \in [0, \infty)^d$, sample size $n \in \mathbb{N}$, integer $k \leqslant n$, reals $0 < p \leqslant 1$ and $\lambda > 0$.

Output: cross validation score associated to the penalization parameter λ.

Split the sample $\mathcal{S}$ into K (approximately) equal-sized folds $\mathcal{S}_1, \ldots, \mathcal{S}_K$.
for each fold $i \in \{1, \ldots, K\}$ **do**
  Validation set: set $\mathcal{S}_i$ as the validation set.
  Training set: set $\mathcal{S} \setminus \mathcal{S}_i$ (all other folds) as the training set.
  Let $\hat{\boldsymbol{\theta}}^{(-i)}$ be the output of Algorithm 1 with parameters d, r, points $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_q$, sample $\mathcal{S} \setminus \mathcal{S}_i$, sample size $\lfloor n(K-1)/k \rfloor$, integer $\lfloor k(K-1)/K \rfloor$, reals p and λ.
  Calculate the empirical stdf estimates $\left(\hat{\ell}^{(i)}_{\lfloor n/K \rfloor, \lfloor k/K \rfloor}(\boldsymbol{c}_m)\right)_{m=1,\ldots,q}$ using the the validation set.
  Evaluate the trained model on the validation set using the i-th score

$$\mathrm{CV}^{(-i)}(\lambda) = \sum_{m=1}^{q} \left(\hat{\ell}^{(i)}_{\lfloor n/K \rfloor, \lfloor k/K \rfloor}(\boldsymbol{c}_m) - \ell\left(\boldsymbol{c}_m; \hat{\boldsymbol{\theta}}^{(-i)}\right) \right)^2.$$

end for
Compute the average cross validation score along all folds via

$$\mathrm{CV}(\lambda) = \frac{1}{K} \sum_{i=1}^{K} \mathrm{CV}^{(-i)}(\lambda).$$

return $\mathrm{CV}(\lambda)$.

the choice of the tail fraction involves a trade-off between bias and variance. In contrast, the choice of the penalization exponent p is less critical, provided that the penalization parameter λ is appropriately tuned—this being achieved, as outlined in Algorithm 2, through K-fold cross validation.

Our procedure is tailored to max-stable mixture models, since Mourahib et al. [27, Theorem 4.8] show that every max-stable distribution with unit-Fréchet margins can in fact be represented as a max-stable mixture model. Of course, in practice, we limit ourselves to a limited number of parametric models for the factors $\boldsymbol{Z}_{\bullet 1}, \ldots, \boldsymbol{Z}_{\bullet r}$.

As in Section 3.1, at each step $t \geqslant 1$ of Algorithm 3, the factors $\boldsymbol{Z}_{\bullet 1}, \ldots, \boldsymbol{Z}_{\bullet t}$ are assumed to follow a parametric model ensuring that the only extreme direction of each factor is D. This assumption could be seen as a limitation of Algorithm 3, as it requires selecting and imposing a specific model for the dependence structure of the factors. However, for the purpose of identifying extreme directions, the choice of a parametric model for the factors is less critical, since the extreme directions of a mixture model $\boldsymbol{M}$ depend only on the matrix A and not on the specific parametric model imposed on the factors; see Theorem 2.4. This observation will also be validated through simulations in Section 4.

Algorithm 3 Estimation of the extreme directions associated with a data sample

Input: integer $d \geqslant 1$, points $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_q \in [0, \infty)^d$, sample $\mathcal{S} = \boldsymbol{X}_1, \ldots, \boldsymbol{X}_n \in [0, \infty)^d$, sample size $n \in \mathbb{N}$, integer $k \leqslant n$, reals $0 < p \leqslant 1$ and $\lambda > 0$.

Output: extreme directions associated with the sample $\mathcal{S}$.

1: Set $t = 1$.
2: Let $\hat{\boldsymbol{\theta}}_t = \left(\hat{A}, \hat{\theta}_Z\right)$ be the output of Algorithm 1 with parameters d, t, points $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_q$, sample $\mathcal{S}$, sample size n, integer $k \leqslant n$, reals p and λ.
3: Compute the signatures $\hat{J}_s = \{j \in D : \hat{a}_{js} > 0\}$ for $s = 1, \ldots, t$ and let $\hat{\mathcal{J}} = \{\hat{J}_1, \ldots, \hat{J}_t\}$.
4: **while** $\varnothing \notin \hat{\mathcal{J}}$ **do**
5: Iterate $t \leftarrow t + 1$.
6: Repeat Steps 2 and 3.
7: **end while**
8: Remove $\{\varnothing\}$ from $\hat{\mathcal{J}}$, that is, update $\hat{\mathcal{J}} \leftarrow \hat{\mathcal{J}} \setminus \{\varnothing\}$.
9: **return** $\hat{\mathcal{J}}$.

4 Simulation study

In this section, we conduct a simulation study to evaluate the estimation procedure introduced in Section 3. We begin by outlining the setup used throughout the simulations in Section 4.1. First we consider the setting where the number of extreme directions is known, as described in Algorithm 1 and detailed in Section 4.2. We then turn to the more general case in which the number of extreme directions is unknown, corresponding to Algorithm 3, and discussed in Section 4.3.

4.1 Preliminaries

Coefficient matrix. Throughout the simulation study, we fix the dimension $d = 4$ and the coefficient matrix

$$A = \begin{pmatrix} 1/3 & 1/3 & 1/3 & 0 \\ 1/2 & 0 & 0 & 1/2 \\ 0 & 3/4 & 1/4 & 0 \\ 0 & 1/2 & 0 & 1/2 \end{pmatrix}. \tag{4.1}$$

Domain of attraction. We simulate samples $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$ of size n in the domain of attraction of a mixture model $\boldsymbol{M}$ as defined in (2.5). To do so, we simply perturb $\boldsymbol{M}$ by adding a lighter-tailed noise. More precisely,

$$\boldsymbol{X}_i = \boldsymbol{M}_i + \boldsymbol{N}_i, \qquad i = 1, \ldots, n, \tag{4.2}$$

where for $i = 1, \ldots, n$, $\boldsymbol{M}_i \in \mathbb{R}^d$ follows a mixture model and $\boldsymbol{N}_i \in \mathbb{R}^d$ has independent centered Gaussian margins with variance $\sigma^2 > 0$. Algorithm 4 outlines the procedure for simulating from a mixture model, relying on the generation of the max-stable random

vectors $\boldsymbol{Z}_{\bullet 1}, \ldots, \boldsymbol{Z}_{\bullet r}$. This is feasible for most popular models; see Dombry et al. [9, Algorithm 1] or the `R` package `mev` [3]. Furthermore, code for Algorithm 4 is freely accessible for both the mixture logistic model and the mixture Hüsler–Reiss model.[2]

Starting points in the minimization algorithm. To calculate the PNPLS estimator using the `L-BFGS-B` algorithm, we need to provide good starting values for $\boldsymbol{\theta} = (A, \boldsymbol{\theta}_Z)$. Suppose that the number of extreme directions is fixed to t, and therefore $A \in \mathcal{M}_{d,t}([0,1])$. Consider a d-variate sample $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$ and write $\boldsymbol{X}_i = (X_{i1}, \ldots, X_{id})$, for $i = 1, \ldots, n$. We determine a starting value for A by applying the k-means clustering algorithm with t clusters to the unit-Pareto transformed data,

$$\left(\frac{n}{n+1-R_{i1}}, \ldots, \frac{n}{n+1-R_{id}} \right), \qquad i = 1, \ldots, n,$$

where for $i = 1, \ldots, n$ and $j = 1, \ldots, d$, R_{ij} is the rank of X_{ij} among $X_{1j}, \ldots, X_{nj}$. We retain only the top 10% of observations with the highest sum of transformed values.

Starting values for the dependence parameters $\boldsymbol{\theta}_Z$ will be assigned randomly, depending on the selected model. As in Section 3, recall that for simplicity, we suppose that the dependence parameters across the columns of A are the same.

Grid points $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_q$. Both the parametric and the empirical stdf are evaluated in $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_q$ during the minimization (3.1). The choice of the number q of points involves a trade-off: selecting a sufficiently large q ensures reliable estimation, while keeping q reasonably small reduces computational expense. In our case, using the `R` package `tailDepFun` [21], we select points that can be generated in a 4-dimensional space:

- **for the mixture logistic fit:** using values $\{1/4, 1/3, 1/2, 3/4, 1\}$ where each point satisfies the condition of having either two or three non-zero components. This gives $q = 650$ points;
- **for the mixture Hüsler–Reiss fit:** using values $\{1/6, 1/8, 1/4, 1/3, 2/3, 3/4, 1\}$, where each point satisfies the condition of having only two non-zero components. This gives $q = 384$ points.

Performance assessment. We simulate $N = 100$ samples as in (4.2) and for each sample, we obtain the PLS estimator $\hat{\boldsymbol{\theta}}^{(l)} = (\hat{A}^{(l)}, \hat{\boldsymbol{\theta}}_Z^{(l)})$, $l = 1, \ldots, N$, using Algorithm 1. We focus on the mixture logistic and the mixture Hüsler–Reiss models (see Section 2.2). Therefore, $\boldsymbol{\theta}_Z \in \mathbb{R}$ for the mixture logistic model and $\boldsymbol{\theta}_Z \in \mathbb{R}^{d(d-1)/2}$ for the mixture Hüsler–Reiss model.

First, we start by assessing the performance of our estimator in terms of the identification of extreme directions. To this end, we introduce a score, denoted "ED-S", which

[2] `https://github.com/AnasMourahib/Penalized_least-squares_estimator/blob/main/Functions/Simulation_mixture_model.R`.

represents the proportion of correctly identified extreme directions,

$$\text{ED-S} = \frac{1}{N}\sum_{l=1}^{N} \mathrm{D}(\hat{\mathcal{J}}_l, \mathcal{J}), \qquad \mathrm{D}(\hat{\mathcal{J}}_l, \mathcal{J}) = 1 - \frac{|\hat{\mathcal{J}}_l \cap \mathcal{J}|}{|\hat{\mathcal{J}}_l \cup \mathcal{J}|} = \frac{|\hat{\mathcal{J}}_l \triangle \mathcal{J}|}{|\hat{\mathcal{J}}_l \cup \mathcal{J}|} \tag{4.3}$$

where $\mathcal{J}$ and $\hat{\mathcal{J}}_l$ denote the signatures of A and $\hat{A}^{(l)}$, respectively, as defined in (2.6).

Second, we assess the performance of our estimator by measuring its proximity to the true parameter $\boldsymbol{\theta} = (A, \boldsymbol{\theta}_Z)$ of the mixture model. In the following, for $l = 1, \ldots, N$, we reorder the columns of $\hat{A}^{(l)}$ so as to minimize the difference $\operatorname{diff}\left(\hat{A}^{(l)}, A\right)$, as defined in (4.5), but without the denominator term. For simplicity, we continue to denote the resulting matrix by $\hat{A}^{(l)}$. To construct a robust performance metric and ensure that the marginal components of our estimator are equally scaled, we mitigate the disproportionate influence of their magnitudes. This is achieved by normalizing each coefficient of $\hat{\boldsymbol{\theta}}$ using the maximum value computed over the N estimates corresponding to that component. More precisely, let $\hat{m}_A = (\hat{m}_{A,js})_{j\in D,\, s\in R}$ denote the $(d \times r)$ matrix where the (j,s)-th entry represents the maximum value calculated based on the coefficients $\hat{a}_{js}^{(l)}$ obtained from the matrices $\hat{A}^{(l)}$, for $l = 1, \ldots, N$. Similarly, recall from Section 3 the dimension v of the dependence parameter vectors $\hat{\boldsymbol{\theta}}_Z$. Let $\hat{\boldsymbol{m}}_Z = (\hat{m}_{Z,t})_{t=1,\ldots,v}$ denote the v-dimensional vector where the t-th entry represents the maximum value calculated based on the coefficients $\hat{\theta}_t^{(l)}$ obtained from the vectors $\hat{\boldsymbol{\theta}}_Z^{(l)}$, for $l = 1, \ldots, N$. We define a standardized version of the mean-squared error as

$$\text{SMSE} = \frac{1}{N}\sum_{l=1}^{N}\left[\operatorname{diff}_{m_A}\left(\hat{A}^{(l)}, A\right) + \operatorname{diff}_{m_Z}\left(\hat{\boldsymbol{\theta}}_Z^{(l)}, \boldsymbol{\theta}_Z\right)\right], \qquad \text{where} \tag{4.4}$$

$$\operatorname{diff}_{m_A}\left(\hat{A}^{(l)}, A\right) = \sum_{j\in D}\sum_{s\in R} \frac{\left(\hat{a}_{js}^{(l)} - a_{js}\right)^2}{\hat{m}_{A,js}^2} \quad \text{and} \quad \operatorname{diff}_{m_Z}\left(\hat{\boldsymbol{\theta}}_Z^{(l)}, \boldsymbol{\theta}_Z\right) = \sum_{t=1}^{v} \frac{\left(\hat{\theta}_{Z,t}^{(l)} - \theta_{Z,t}\right)^2}{\hat{m}_{Z,t}^2}. \tag{4.5}$$

Note that the maximum of a coefficient across N estimates is zero if and only if all estimates of that coefficient are equal to zero. This occurs only for the null coefficients of A. In such cases, by convention, we define $0/0 = 0$.

4.2 Known number of columns

We consider the mixture logistic model in Example 2.5 and the mixture Hüsler–Reiss model in Example 2.6, both with coefficient matrix A in (4.1) and

1. dependence parameter $\alpha := \alpha_1 = \ldots = \alpha_4 = 0.25$ for the mixture logistic model,
2. variogram matrix $\Gamma := \Gamma^{(1)} = \ldots = \Gamma^{(4)} \in \mathcal{M}_{4,4}([0,\infty))$, with all off-diagonal elements equal to 1 for the mixture Hüsler–Reiss model.

Note that for the mixture Hüsler–Reiss model, we do not suppose that all the off-diagonal elements of the variogram matrix are equal in the estimation procedure, meaning that $\Theta_Z \subset (0,\infty)^6$ in (3.1).

In this section, we suppose that the number of extreme directions, $r = 4$, is known. We generate $N = 100$ samples of size $n \in \{1\,000,\ 2\,000,\ 3\,000\}$ in the domain of attraction of the mixture logistic model and of the mixture Hüsler–Reiss model as in (4.2), both with a noise variance set to $\sigma^2 = 9$.

Effect of the penalization exponent. We start by studying the effect of the penalization exponent p used in the minimization problem (3.1) on the scores ED-S in (4.3) and SMSE in (4.4). For the tail fraction, we set $k/n = n^c/n$ with $c = \{0.4, 0.5, 0.6\}$. With sample sizes $n \in \{1\,000,\ 2\,000,\ 3\,000\}$, this corresponds approximately to $k/n \in \{0.01, 0.02, 0.05\}$. Recall that we choose the penalization parameter λ in (3.1) based on a 10-fold cross validation (Algorithm 2) over a well chosen grid. We use an adaptive penalization parameter grid, meaning that for each value of the penalization exponent p, we have a distinct grid of penalization parameters λ.

Figure 3 displays the scores ED-S and SMSE as functions of the penalization exponent p. Results are shown for the mixture logistic fit. Qualitatively similar results arise under the mixture Hüsler–Reiss fit. The effect of p is not important as long as the penalization parameter λ in (3.1) is well chosen. In fact, for $0 < p_1 < p_2 < 1$ and a parameter a of $\boldsymbol{\theta}_A$, the term T in (3.5), which governs the opposite direction step in the minimization algorithm, satisfies $T(p_1)/T(p_2) \to \infty$, whenever the parameter $a \to 0$. It follows that the penalization in (3.1) is stronger with smaller penalization exponent. This explains the use of an adaptive grid of penalization parameters, where the values of the grid associated with p_1 are smaller than the ones used for p_2.

Effect of the tail fraction. We study the effect of the tail fraction k/n used in (3.1) on the scores ED-S in (4.3) and SMSE in (4.4). Figure 3 shows that the effect of the penalization exponent p is not important as long as the penalization parameter λ is well chosen using 10-fold cross validation. Therefore, we set $p = 0.4$ and perform a 10-fold cross validation to choose λ, where for each value p, we use an adapted grid.

Figure 4 illustrates the behavior of the scores ED-S and SMSE as functions of the tail fraction k/n for a mixture logistic and a mixture Hüsler–Reiss fit, respectively.

For the mixture logistic model, we observe that both scores tend to improve near the middle of the tail fraction grid. This pattern reflects a bias–variance trade-off: a smaller tail fraction k/n results in higher variance but lower bias, as fewer points are treated as extreme observations. In contrast, a larger k/n leads to lower variance but increased bias, since more points—possibly not truly extreme—are included in the estimation. We also observe that as the sample size increases, the optimal choice of the tail fraction k/n—that is, the optimal proportion of points considered as extreme—tends to decrease. This is intuitive: with a larger sample size, a smaller proportion of extreme observations is sufficient to achieve accurate estimation.

The score ED-S for the mixture Hüsler–Reiss fit behaves similarly to that of the mixture logistic fit. In contrast, the score SMSE does not reflect the bias-variance trade-off effect observed for the mixture logistic fit. We can also see that this score is higher for the mixture Hüsler–Reiss model. This makes sense, since the dependence coefficient

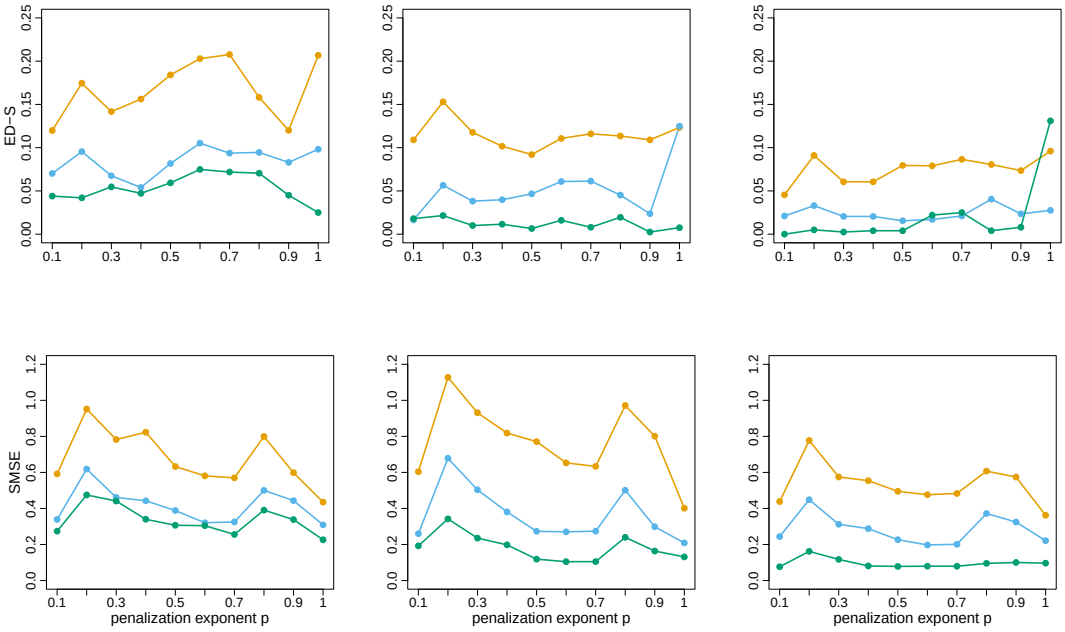

Figure 3: Scores ED-S (top) and SMSE (bottom) defined in (4.3) and (4.4), respectively, as functions of the penalization exponent p, for the mixture logistic model perturbed as in (4.2) and sample sizes $n = 1\,000$ (left), $n = 2\,000$ (center), and $n = 3\,000$ (right). Curves correspond to varying tail fractions, with approximately $k/n = 0.01$ (orange), $k/n = 0.02$ (blue), and $k/n = 0.05$ (green).

Γ for the mixture Hüsler–Reiss model is of dimension $d = 6$ (in our example), while the dependence coefficient for the mixture logistic model is of dimension $d = 1$. This makes the estimation of the dependence parameter for the mixture Hüsler–Reiss model more complicated.

We now discuss the estimation of the variogram matrix Γ of the mixture Hüsler–Reiss model in Point 2. Figure 5 shows the boxplots of the upper off-diagonal entries of Γ, except for the entry Γ_{23}; since no signature of the matrix A in (4.1) contains the pair $(2, 3)$, the coefficient Γ_{23} is not identifiable. More precisely, Γ_{23} does not contribute to the stdf in (2.7) associated to the corresponding mixture model.

4.3 Unknown number of columns

We now suppose that the number of extreme directions is unknown and identify the extreme directions using Algorithm 3. Following Figure 3, the choice of the penalization parameter p is not important as long as the penalization parameter λ is well chosen using K-fold cross validation as explained in Algorithm 2. Therefore, we simply choose $p = 0.4$ and use a 10-fold cross validation procedure to choose λ from the grid associated with

$p = 0.4$. Following Figure 4, we choose $k/n \in \{0.04, 0.06, 0.08\}$. We simulate samples $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$ from the mixture logistic model in Point 1 and samples $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n$ from the mixture Hüsler–Reiss model in Point 2, both perturbed as in (4.2). We apply Algorithm 3 with the following two settings:

1. **Correct model specification.** Fit a mixture logistic model to $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$ and a mixture Hüsler–Reiss model to $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n$;
2. **Misspecified model.** Fit a mixture logistic model to $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n$ and a mixture Hüsler–Reiss model to $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$.

We investigate the impact of the sample size n on the score ED-S defined in (4.3). The results for the mixture logistic fit are shown in Figure 6. Qualitatively similar results arise under the mixture Hüsler–Reiss fit. Both the fit under the correct model specification and the fit under the misspecified model yield satisfactory performance even for relatively small sample sizes, with good results observed from $n = 500$ onward.

5 Applications

5.1 River discharge data

We illustrate our method using river discharge data from 31 stations along the Danube River. The dataset consists of daily river discharge measurements from the summer months between 1960 and 2010. To mitigate temporal dependence, the data were declustered by Engelke et al. [13], resulting in approximately seven to ten observations per year and yielding $n = 428$ observations in total. The declustered dataset is available in the `GraphicalExtremes` package in `R`; see Engelke et al. [13].

To reduce the dimensionality and thereby lower the computational cost of identifying extreme directions, we apply a spatial clustering technique. Specifically, we partition the stations into K clusters using the Partitioning Around Medoids (PAM) algorithm [20, 4], adapted to threshold exceedances as in Kiriliouk and Naveau [23]. As a pseudo-distance between two stations s and t, we use

$$\hat{d}_{st} = \frac{1 - \hat{\chi}_{st}}{2 \cdot (3 - \hat{\chi}_{st})}, \qquad \hat{\chi}_{st} := 2 - \hat{\ell}_{n,k,st}(1,1), \tag{5.1}$$

where k/n represents the tail fraction and $\hat{\ell}_{n,k,st}$ the pairwise empirical stdf in (2.12) for the margin $\{s, t\}$.

On the one hand, the number K of clusters balances model simplicity (fewer clusters) and intra-cluster homogeneity. Various methods exist for selecting an appropriate value of K, such as minimizing within-cluster inertia [20]. On the other hand, as explained in Section 4, the choice of the tail fraction k/n is a trade-off between bias and variance. Here, we simply choose $K = 5$ and $q = 0.05$ which gives spatially coherent results as displayed in Figure 7. The cluster centers are given then by the five stations $\{7, 18, 24, 27, 30\}$.

Consider the $d = K = 5$ stations of interest to be the cluster centers. The data $\boldsymbol{X}_i = (X_{i1}, \ldots, X_{id})$ representing the time series of river discharge over the five stations are assumed to be independent and identically distributed for $i = 1, \ldots, n$ and we will identify their associated extreme directions using Algorithm 3.

Choice of tuning parameters. Before estimating the extreme directions associated with the five central stations, we first address the choice of the tuning parameters: the tail fraction k/n, the penalization exponent p, the penalization coefficient λ and the grid points $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_q \in [0, \infty)^5$. As noted in the simulation study, the choice of the penalization exponent p has limited impact provided that the penalization parameter λ is appropriately selected. We simply choose $p = 0.4$, although, in terms of results, other values of the penalization exponent yield similar findings. Guided by the results in Section 4 and given the sample size, $n = 428$, we set the tail fraction to $k/n = 0.1$. The penalization parameter λ is selected via 5-fold cross-validation procedure as described in Algorithm 2 over a grid of values. This grid is constructed in a greedy iterative way such that the optimal value λ^* lies roughly in the middle of the range: we start with a grid $\{0.1, 0.2, \ldots, 10\}$, ranging from 0.1 to 10 in steps of 0.1, we then find the optimal value with respect to the 5-fold cross-validation, and construct a new grid centered around this value. We repeat this process until results are stable. This yields a grid of values, $\{0.01, 0.02, \ldots, 1\}$, ranging from 0.01 to 1 in steps of 0.01. Note that given the limited sample size, $n = 428$, we perform 5-fold cross-validation, instead of 10-fold cross-validation, as in Section 4, where the sample size was larger. Finally, we select points that can be generated in a 5-dimensional space using values $\{1/4,\ 1/3,\ 1/2,\ 3/4,\ 1\}$ where each point satisfies the condition of having either two or three non-zero components.

Results. At each step of Algorithm 3, we fit a mixture logistic model to the five-dimensional river discharge dataset. While in the simulation study, the initial dependence parameter α of the mixture logistic model was randomly initialized within the interval $[0, 1]$, here we set a fixed initial value $\alpha = 0.5$. The results are summarized in Table 1 (left). The algorithm identifies five extreme directions, which appear spatially consistent with the stations indicated on the map in Figure 7.

As indicated in Table 1 (left), each extreme direction is associated with a weight, see [27, Proposition 5.1]. Specifically, for a mixture logistic model with column coefficients $(\boldsymbol{a}_{\bullet J} : J \in \mathcal{J})$, where $\mathcal{J}$ is the set of extreme directions (equivalently, the signature of the model) and a dependence parameter $\alpha \in [0, 1]$, the scalar

$$w_{\tilde{J}} = \frac{\left(\sum_{j \in \tilde{J}} a_{j\tilde{J}}^{1/\alpha}\right)^{\alpha}}{\sum_{J \in \mathcal{J}} \left(\sum_{j \in J} a_{jJ}^{1/\alpha}\right)^{\alpha}} \tag{5.2}$$

can be interpreted as the weight corresponding to the extreme direction $\tilde{J} \in \mathcal{J}$.

Results make sense given the hydrological nature of the dataset. The standard extreme direction $\{7, 18, 24, 27, 30\}$ can be explained by the fact that extreme river discharges are often caused by large-scale precipitation during the summer months; see Böhm and Wetzel [5]. The other non-standard extreme directions can be divided into two clusters. The first cluster corresponds to the extreme directions $\{24\}$, $\{24, 27\}$, and $\{7, 24\}$, with stations 7, 24, and 27 located along Bavarian rivers. Stations 24 and 27 are geographically close and often experience the same weather systems. However, the presence of the extreme direction $\{7, 24\}$ is surprising, since the two stations receive water from different sources;

see [14, Figure 9]. This might explain why the extreme direction $\{7, 24\}$ is detected with the lowest weight. The second cluster corresponds to the extreme direction $\{7, 18, 30\}$, with stations 7, 18, and 30 located along Alpine tributaries (southern Germany and western Austria). This can be explained by the fact that these three stations receive water from the same region, flowing from south to north; see [14, Figure 9].

Comparison to other algorithms from literature. The *DAMEX* algorithm introduced in Goix et al. [17, Algorithm 1] identifies only the extreme direction $\{7, 18, 24, 27, 30\}$, even when varying its tuning parameters. This extreme direction was also identified by our algorithm with the largest weight (see Table 1 (left)). Our extreme direction identification method thus appears capable of detecting sparser extreme-direction structures. However, it is computationally more expensive than the DAMEX algorithm.

Performance assessment. To assess the fit of the mixture logistic model to the Danube dataset, we compare the empirical extremal correlation in (5.1) with the estimated extremal correlation for each pair of stations. For a mixture logistic model with estimated matrix $\hat{A} \in \mathcal{M}_{d,r}([0,1])$ and dependence coefficient $\hat{\alpha}$, the extremal correlation between two margins s and t is

$$\hat{\chi}_{st,\,\mathrm{est}} = \sum_{k=1}^{r} \left(\hat{a}_{sk}^{1/\hat{\alpha}} + \hat{a}_{tk}^{1/\hat{\alpha}} \right)^{\hat{\alpha}}. \tag{5.3}$$

Figure 8 (left) compares the empirical and fitted extremal correlations. Here we use $p = 0.4$ and $k/n = 0.1$, noting that other choices of tuning parameters yield qualitatively similar results. The agreement is strong, as most points lie close to the line $x = y$. Note that the evaluation process is entirely separate from the model fitting.

5.2 Financial industry portfolios

We consider the value-weighted returns for $d = 5$ industry portfolios, downloaded from the Kenneth French Data Library[3]: "1 = Consumer Goods and Services", "2 = Manufacturing, Energy, and Utilities", "3 = Technology Equipment", "4 = Healthcare", "5 = Other". The individual stocks that make up the five industry portfolios are taken from all listed firms on the NYSE, AMEX, and NASDAQ. Data is available between July 1926 and March 2025, leading to a sample of size $n = 51\,922$.

Data pre-processing. We convert the standard returns to log-returns and multiply by (-1) since we are interested in extreme *losses*. Let $\boldsymbol{X}_t = (X_{t1}, \ldots, X_{td})$ denote the negative log-returns of the five portfolios at time $t = 1, \ldots, n$. In order to obtain a time series without temporal dependence, we fit a GARCH$(1,1)$ model marginally. More precisely, for each portfolio $j = 1, \ldots, 5$,

$$X_{tj} = \mu_{tj} + \sigma_{tj}\varepsilon_{tj}, \qquad t = 1, \ldots, n, \tag{5.4}$$

[3]See `https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/Data_Library/det_5_ind_port.html` for more details about the data and the definition of each portfolio.

where σ_t^2 is the conditional variance (volatility), and $\varepsilon_{jt} \overset{\text{iid}}{\sim} \mathcal{N}(0,1)$. For more details about the GARCH model and its application to financial data, see for example Francq and Zakoian [16]. We continue with the fitted residuals

$$\hat{\varepsilon}_{tj} = \frac{X_{tj} - \hat{\mu}_{tj}}{\hat{\sigma}_{tj}}, \qquad t = 1, \ldots, n.$$

We apply the Ljung–Box test for serial correlation to each of the $d = 5$ marginal residual series $\hat{\varepsilon}_{1j}, \ldots, \hat{\varepsilon}_{nj}$, using lags 1 through 10. In every case, the test fails to reject the null hypothesis of no autocorrelation, suggesting that the residuals are serially uncorrelated.

Choice of tuning parameters. Similarly to Section 5.1, we set the penalization exponent to $p = 0.4$. Other values of the penalization exponent yielded similar findings. The sample size ($n = 51\,922$) of the industry portfolios dataset is much larger than the one of the Danube dataset ($n = 428$) in Section 5.1. Therefore, in this section, we set the tail fraction k/n to a smaller value $k/n = 0.05$. The penalization parameter λ is selected via the 10-fold cross-validation procedure described in Algorithm 2 over a grid of values. As explained in Section 5.1, this grid is chosen in a greedy iterative way until results stabilize. This yields a grid of values, $\{0.001, 0.002, \ldots, 0.02\}$, ranging from 0.001 to 0.02 in steps of 0.001. Finally, we select points that can be generated in a 5-dimensional space using values $\{1/4,\ 1/3,\ 1/2,\ 3/4,\ 1\}$ where each point satisfies the condition of having either two or three non-zero components.

Results. As in Section 5.1, at each step of Algorithm 3 we fit a mixture logistic model (Example 2.5) to the five industry portfolio and set the initial value for the dependence parameter to $\alpha = 0.5$. The results are summarized in Table 1 (right). Although three extreme directions are identified, the extreme direction $\{1, 2, 3, 4, 5\}$ including all the five industry portfolios has the biggest weight. Most extreme loss events therefore affect all sectors simultaneously, indicating strong systemic dependence in crisis periods.

Comparison to other algorithms from literature. The *DAMEX* algorithm [17, Algorithm 1] identifies only the extreme direction $\{1, 2, 3, 4, 5\}$, even when varying its tuning parameters. This extreme direction was also identified by our algorithm with the largest weight, see Table 1 on the right. Again, this emphasizes that our extreme direction identification method is capable of detecting sparser extreme-direction structures.

Performance assessment. To assess the fit of the mixture logistic model to the industry portfolios dataset, we compute for each pair (s, t), both the empirical extremal correlation in (5.1) and the fitted extremal correlation in (5.3). Figure 8 (right) compares the empirical and fitted extremal correlations. We use $p = 0.4$ and $k/n = 0.05$ to compute the fitted and the empirical extremal correlations. Other choices of tuning parameters yielded qualitatively similar results. The agreement is strong, as most points lie close to the line $x = y$.

Extreme directions	Weights
$\{7, 18, 24, 27, 30\}$	0.49
$\{7, 18, 30\}$	0.17
$\{24\}$	0.15
$\{24, 27\}$	0.14
$\{7, 24\}$	0.06

Extreme directions	Weights
$\{1, 2, 3, 4, 5\}$	0.91
$\{4\}$	0.05
$\{1, 2, 4, 5\}$	0.04

Table 1: Extreme directions identified by Algorithm 3 for the Danube river discharge data at five stations, discussed in Section 5.1 (left), and the five industry portfolios, discussed in Section 5.2 (right), along with their associated weights as in (5.2).

6 Conclusion

Given a vector of d risk factors, extreme directions refer to groups of factors $J \subset \{1, \ldots, d\}$ that are large simultaneously while those outside this group are not. Mourahib et al. [27] propose a construction method of multivariate generalized Pareto distributions, or equivalently max-stable distributions, with multiple extreme directions. Extreme directions are a consequence of parameters of the mixture model being on the border of the parameter space [27, Proposition 4.5]. Parameter estimation is therefore a challenging topic. We proposed a penalized least-squares estimator (3.1) based on the estimator from [12]. The penalization term in this estimator is based on a "quasi-norm" L_p with $p \leqslant 1$. We motivated the choice of $p \leqslant 1$ both geometrically and heuristically. Next, we use this estimator to propose a data-driven algorithm for extreme-directions identification. Using a simulation study, we show that although the algorithm relies on a parametric assumption for the mixture model (e.g., mixture Hüsler–Reiss graphical model or mixture logistic model), the specific choice of the model is not crucial. In addition, the algorithm involves three tuning parameters; again through simulations, we demonstrate that the tuning essentially reduces to selecting two parameters: one that controls the proportion of data considered as extreme observations and the other that controls the penalization strength, chosen using K-fold cross-validation. We illustrated our algorithm on two real datasets, determining the sets of variables that form extreme directions. We compare our Algorithm with the *DAMEX* algorithm in Goix et al. [17] and show that our algorithm appears capable of detecting non-standard extreme directions.

We obtain good simulation results for our penalized least-squares estimator proposed in (3.1). However, no results on consistency or asymptotic normality are provided. Two main difficulties prevent us from establishing asymptotic theoretical results:

1. when the penalization exponent of the penalization term is such that $p \leqslant 1$, our PLS estimator is the solution of a non-convex minimization problem;

2. some of the true parameters lie on the boundary of the parameter space. Under certain conditions, this is treated in Kiriliouk [22] for the unpenalized version of our estimator, i.e., the weighted least-squares estimator in Einmahl et al. [12]. However, these conditions seem difficult to verify for the parametric mixture models.

In the simplest case where the true parameters are in the interior of the parameter space and no penalization is needed, asymptotic results are proved in Einmahl et al. [12] under certain conditions. However, as pointed out in the same paper, these conditions are either difficult to verify, for instance with the Hüsler–Reiss parametric model, or are not verified at all, for instance with the max-linear model.

Acknowledgment

The authors are grateful to Sebatian Engelke for a helpful discussion about the interpretation of the analysis of the river discharge data in Section 5.1.

A Appendix

Algorithm 4 Simulation of a mixture model

Input: integers $d, r \geqslant 1$, matrix $A = (a_{jk})_{j \leqslant d, k \leqslant r} \in \mathcal{M}_{d,r}([0,1])$ and max-stable random vectors $\boldsymbol{Z}_{\bullet 1}, \ldots, \boldsymbol{Z}_{\bullet r} \in \mathbb{R}^d$ as in Definition 2.2. ▷ For simplicity, we consider the vector $\boldsymbol{Z}_{\cdot k} = (Z_{1k}, \ldots, Z_{dk}) \in \mathbb{R}^d$ for each $k = 1, \ldots, r$. However, as discussed after Definition 2.2, only the sub-vector $\boldsymbol{Z}_{J_k,k}$ is relevant.

Output: mixture logistic model $\boldsymbol{M} \in \mathbb{R}^d$ in (2.5) with coefficient matrix A and factors $\boldsymbol{Z}_{\bullet 1}, \ldots, \boldsymbol{Z}_{\bullet r}$.

```
Initialize a vector M = (0, ..., 0) ∈ R^d
Initialize a list Z̃ = {Z̃_1, ..., Z̃_r}, where each Z̃_k ∈ R^d is initialized as a zero vector.
for each k ∈ {1, ..., r} do
    Define J_k as in (2.6), the k-th signature of the coefficient matrix A.
    Set Z̃_{k,j}, the j-th component of Z̃_k, to a_{jk} · Z_{jk}, for every j ∈ J_k.
end for
Compute M_j = max{Z̃_{j1}, ..., Z̃_{jr}}, for each j ∈ {1, ..., d}.
return M
```

References

[1] Jan Beirlant, Yuri Goegebeur, Johan Segers, and Jozef L Teugels. *Statistics of extremes: theory and applications*, volume 558. John Wiley & Sons, 2004.

[2] Jan Beirlant, Mikael Escobar-Bach, Yuri Goegebeur, and Armelle Guillou. Bias-corrected estimation of stable tail dependence function. *Journal of Multivariate Analysis*, 143:453–466, 2016.

[3] Leo Belzile, Jennifer L Wadsworth, Paul J Northrop, Scott D Grimshaw, Jin Zhang, Michael A Stephens, and Art B Owen. Package 'mev'. Technical report, 2024.

[4] Elsa Bernard, Philippe Naveau, Mathieu Vrac, and Olivier Mestre. Clustering of maxima: Spatial dependencies among heavy rainfall in France. *Journal of climate*, 26(20):7929–7937, 2013.

[5] Oliver Böhm and K-F Wetzel. Flood history of the Danube tributaries Lech and Isar in the Alpine foreland of Germany. *Hydrological Sciences Journal*, 51(5):784–798, 2006.

[6] Richard H. Byrd, Peihuang Lu, Jorge Nocedal, and Ciyou Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208, 1995.

[7] Stuart Coles. *An introduction to statistical modeling of extreme values*, volume 208. Springer, 2001.

[8] Laurens De Haan and Ana Ferreira. *Extreme value theory: an introduction*, volume 3. Springer, 2006.

[9] Clément Dombry, Sebastian Engelke, and Marco Oesting. Exact simulation of max-stable processes. *Biometrika*, 103(2):303–317, 2016.

[10] Holger Drees and Xin Huang. Best attainable rates of convergence for estimators of the stable tail dependence function. *Journal of Multivariate Analysis*, 64(1):25–46, 1998.

[11] John HJ Einmahl, Andrea Krajina, and Johan Segers. An M-estimator for tail dependence in arbitrary dimensions. *The Annals of Statistics*, 40(3):1764–1793, 2012.

[12] John HJ Einmahl, Anna Kiriliouk, and Johan Segers. A continuous updating weighted least squares estimator of tail dependence in high dimensions. *Extremes*, 21:205–233, 2018.

[13] Sebastian Engelke, Adrien S Hitz, Nicola Gnecco, and Manuel Hentschel. Package 'graphicalExtremes'. 2024.

[14] Sebastian Engelke, Nicola Gnecco, and Frank Röttger. Extremes of structural causal models. *arXiv preprint arXiv:2503.06536*, 2025.

[15] Anne-Laure Fougères, Cécile Mercadier, and John P Nolan. Dense classes of multivariate extreme value distributions. *Journal of Multivariate Analysis*, 116: 109–129, 2013.

[16] Christian Francq and Jean-Michel Zakoian. *GARCH models: structure, statistical inference and financial applications.* John Wiley & Sons, 2019.

[17] Nicolas Goix, Anne Sabourin, and Stéphan Clémençon. Sparse representation of multivariate extremes with applications to anomaly detection. *Journal of Multivariate Analysis*, 161:12–31, 2017.

[18] Raphaël Huser and Anthony C Davison. Composite likelihood estimation for the Brown–Resnick process. *Biometrika*, 100(2):511–518, 2013.

[19] Jürg Hüsler and Rolf-Dieter Reiss. Maxima of normal random vectors: Between independence and complete dependence. *Statistics and Probability Letters*, 7(4): 283–286, 1989.

[20] Leonard Kaufman and Peter J. Rousseeuw. *Finding groups in data: an introduction to cluster analysis*. John Wiley & Sons, 2009.

[21] Anna Kiriliouk. Vignette for the tailDepFun package. Technical report, 2016.

[22] Anna Kiriliouk. Hypothesis testing for tail dependence parameters on the boundary of the parameter space. *Econometrics and Statistics*, 16:121–135, 2020.

[23] Anna Kiriliouk and Philippe Naveau. Climate extreme event attribution using multivariate peaks-over-thresholds modeling and counterfactual theory. *The Annals of Applied Statistics*, 2020.

[24] Anna Kiriliouk, Johan Segers, and Laleh Tafakori. An estimator of the stable tail dependence function based on the empirical beta copula. *Extremes*, 21:581–600, 2018.

[25] Amanda Lenzi, Julie Bessac, Johann Rudi, and Michael L Stein. Neural networks for parameter estimation in intractable models. *Computational Statistics & Data Analysis*, 185:107762, 2023.

[26] Nicolas Meyer and Olivier Wintenberger. Multivariate sparse clustering for extremes. *Journal of the American Statistical Association*, 119(forthcoming):1–23, 2023.

[27] Anas Mourahib, Anna Kiriliouk, and Johan Segers. Multivariate generalized Pareto distributions along extreme directions. *Extremes*, pages 1–34, 2024.

[28] Simone A Padoan, Mathieu Ribatet, and Scott A Sisson. Likelihood-based inference for max-stable processes. *Journal of the American Statistical Association*, 105(489): 263–277, 2010.

[29] Jordan Richards, Matthew Sainsbury-Dale, Andrew Zammit-Mangion, and Raphaël Huser. Neural bayes estimators for censored inference with peaks-over-threshold models. *Journal of Machine Learning Research*, 25(390):1–49, 2024.

[30] Holger Rootzén and Nader Tajvidi. Multivariate generalized Pareto distributions. *Bernoulli*, 12(5):917–930, 2006.

[31] Martin Schlather and Jonathan Tawn. Inequalities for the extremal coefficients of multivariate extreme value distributions. *Extremes*, 5:87–102, 2002.

[32] Emma S Simpson, Jennifer L Wadsworth, and Jonathan A Tawn. Determining the dependence structure of multivariate extremes. *Biometrika*, 107(3):513–532, 2020.

[33] Alec Stephenson. Simulating multivariate extreme value distributions of logistic type. *Extremes*, 6(1):49–59, 2003.

[34] Jonathan A. Tawn. Bivariate extreme value theory: models and estimation. *Biometrika*, 75(3):397–415, 1988.

[35] Jonathan A Tawn. Modelling multivariate extreme value distributions. *Biometrika*, 77(2):245–253, 1990.

[36] Yizao Wang and Stilian A Stoev. Conditional sampling for spectrally discrete max-stable random fields. *Advances in Applied Probability*, 43(2):461–483, 2011.

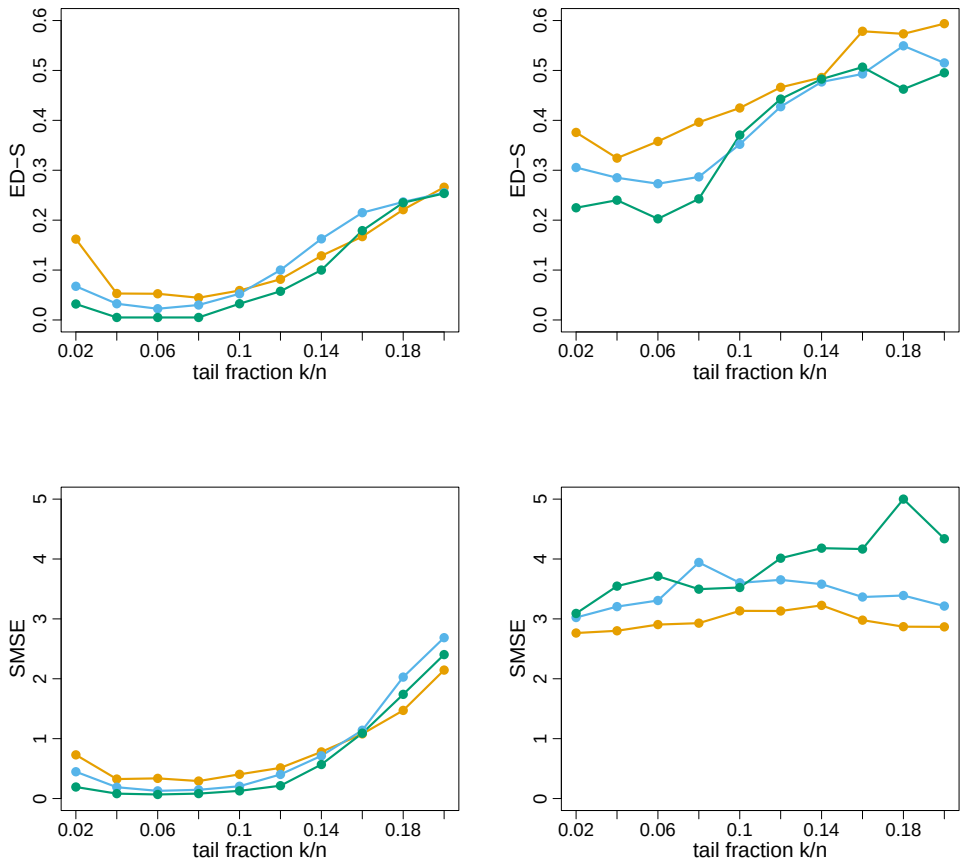

Figure 4: Scores ED-S (top) and SMSE (bottom) defined in (4.3) and (4.4), respectively, as functions of the tail fraction k/n, for the mixture logistic model (left) and the mixture Hüsler–Reiss model (right), both perturbed as in (4.2), with penalization exponent $p = 0.4$. Curves correspond to $n = 1\,000$ (orange), $n = 2\,000$ (blue) and $n = 3\,000$ (green).

.

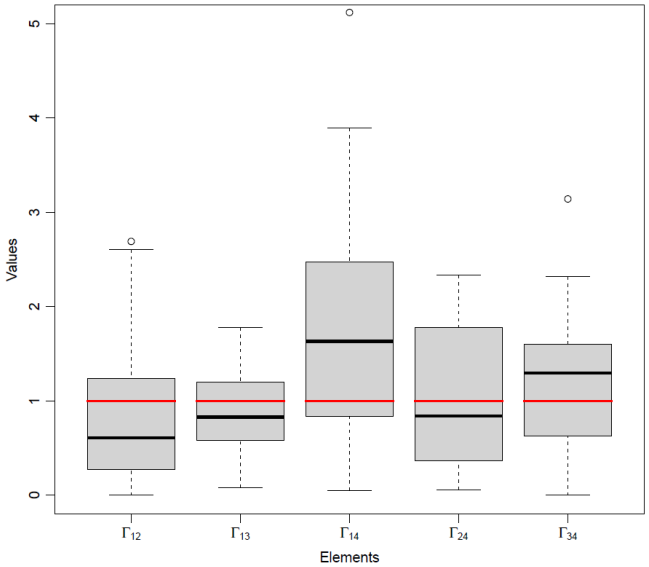

Figure 5: Boxplots of the estimated coefficients Γ_{ij} from the variogram matrix Γ for all pairs (i, j) included in some signature J of A. The true value 1 for each coefficient is presented with a horizontal red line. The tail fraction k/n is set to 0.06, the penalization exponent to $p = 0.4$ and the sample size to $n = 1\,000$.

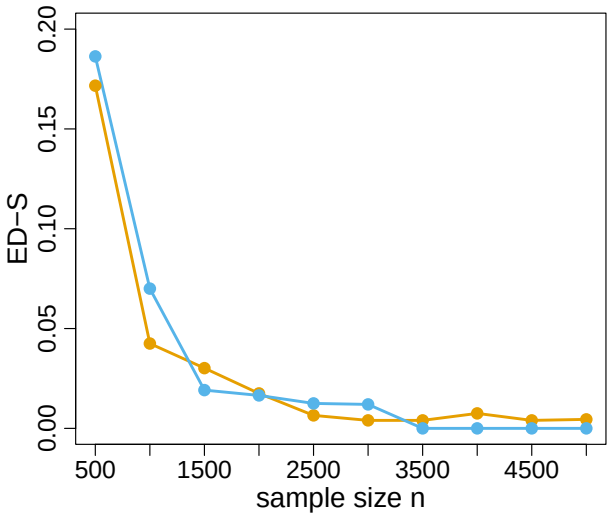

Figure 6: The score ED-S defined in (4.3), for simulations from the mixture logistic model (blue) and the mixture Hüsler–Reiss model (orange), both perturbed as in (4.2). In both cases, a mixture logistic model is fitted using the estimation procedure in Algorithm 3. The penalization exponent is fixed to $p = 0.4$ and the tail fraction to $k/n = 0.08$.

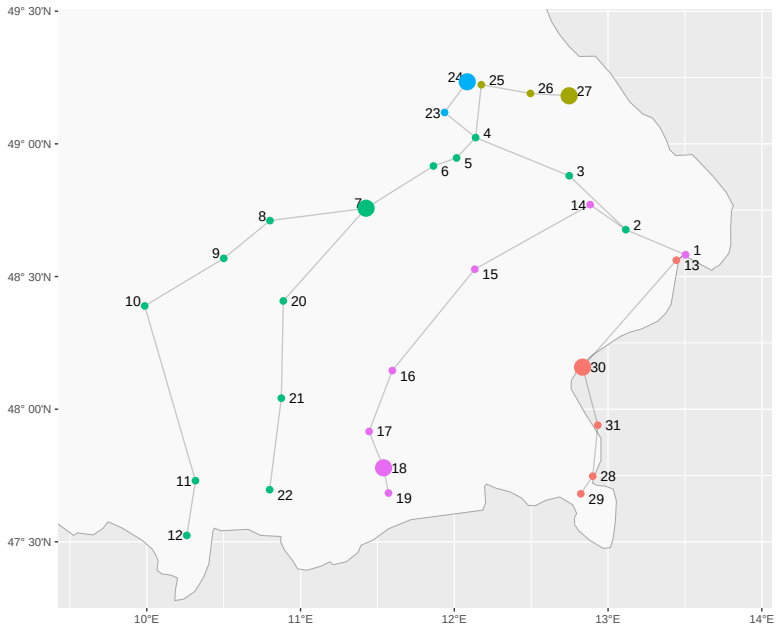

Figure 7: 31 stations from the Danube dataset clustered into $K = 5$ groups using an adapted version of the PAM algorithm with the pseudo-distance $\hat{d}$ in (5.1) with tail fraction $k/n = 0.1$. Stations with the same color belong to the same cluster, and within each cluster, the station represented by a larger-sized shape indicates the cluster center.

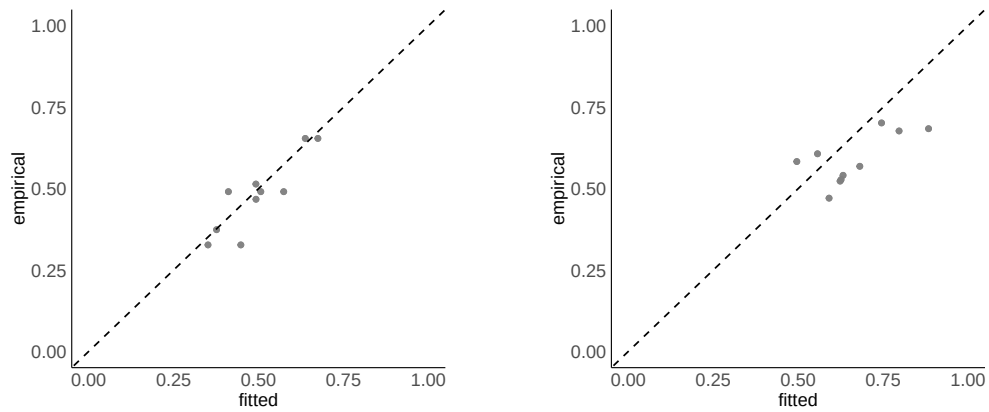

Figure 8: Empirical (5.1) versus fitted (5.3) extremal correlations for the results from the Danube dataset in Section 5.1 (left) and industry portfolios in Section 5.2 (right).