Estimation of extreme quantiles from heavy-tailed distributions with neural networks

Michaël Allouche
michael.allouche@polytechnique.edu
CMAP, CNRS, Ecole Polytechnique, Institut Polytechnique de Paris
Route de Saclay, 91128 Palaiseau Cedex, France

Stéphane Girard
stephane.girard@inria.fr
Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK
38000 Grenoble, France

Emmanuel Gobet
emmanuel.gobet@polytechnique.edu
CMAP, CNRS, Ecole Polytechnique, Institut Polytechnique de Paris
Route de Saclay, 91128 Palaiseau Cedex, France

June 24, 2022

Abstract

We propose new parametrizations for neural networks in order to estimate extreme quantiles in both non-conditional and conditional heavy-tailed settings. All proposed neural network estimators feature a bias correction based on an extension of the usual second-order condition to an arbitrary order. The convergence rate of the uniform error between extreme log-quantiles and their neural network approximation is established. The finite sample performances of the non-conditional neural network estimator are compared to other bias-reduced extreme-value competitors on simulated data. Finally, the conditional neural network estimators are implemented to investigate the behaviour of extreme rainfalls as functions of their geographical location in the southern part of France.

Keywords: Extreme-value theory, heavy-tailed distribution, quantile estimation, conditional quantile estimation, neural networks
MSC: 62G32 , 68T07, G2G08, 62G32

1 Introduction

Nowadays, estimation of extreme events is a major concern due to climate change. According to the last Intergovernmental panel on climate change [54], on a global scale, it is very likely that the extreme daily precipitation events will increase by about 7% per 1°C of global warming and become more frequent. Furthermore, the global surface temperature is also very likely to rise on average over the years 2081-2100 by a range of 1°C-5.7°C depending on the greenhouse gas emission scenario; leading to a climb of heat waves, global mean sea levels and natural disasters. Potential consequences for the human being might be both direct (increased mortality, population displacements, hunger,...), and indirect (increased costs of raw materials, food, insurance premiums,...).

In this context, numerical simulation of unfavorable extreme (but plausible) scenarios with generative models [3, 5, 62] is a major tool to study the occurrence and the size of such risks. These

models aim at learning the distribution of a random variable, whose resulting support is usually restricted to the one observed in the training dataset (of size n). In this work, we rather focus on the prediction of the variable of interest beyond the largest observed value. Such events are associated with tail probabilities α_n larger than $1-1/n$ and are therefore referred to as extreme quantiles, see for instance [8, 20] who studied extreme bank losses, [55] in the context of flood risk assessment and [17] for an application to oceanographic data. We also refer to the books [4, 15, 19] for a general overview of the theoretical background on extreme quantile estimation.

We focus on heavy-tailed distributions which have been revealed useful to describe the tail structure of actuarial and financial data (see [19, Page 9] and more recently [53, Page 1]), as well as extreme events when studying climatic risk (see [10, Section 1.2] and [47, Section 7]). One of the most famous estimators in such a context is the Weissman estimator [61] described in Section 2 thereafter. Basing on its asymptotic representation [15, Theorem 4.3.8], bias-reduced estimators have been introduced thanks to a prior estimation of additional parameters driving the first dominant bias component, see [38].

Our first main contribution is to show that all first J bias terms have a simple neural network (NN) representation, where J is linked to the complexity (depth and width) of the network. Based on this result, we derive a NN extreme quantile estimator which features an automatic estimation and removal of all J first bias terms. Second, two extensions of this NN estimator are introduced to tackle the conditional case, *i.e.* when the extreme quantiles depend on a multi-dimensional covariate. Up to our knowledge, this is the first attempt at reducing the estimation bias in conditional extreme quantiles.

This paper is organized as follows. An extrapolation principle for estimating extreme quantiles in the non-conditional heavy-tailed case is introduced in Section 2 with an emphasis on bias corrected estimators. The construction of the NN estimator is presented in Section 3 which ends up with our first theoretical result (Theorem 2) on the associated approximation error. Similarly, an extrapolation principle for estimating conditional extreme quantiles is proposed in Section 4 and two conditional NN estimators are derived in Section 5 with their associated approximation properties (Theorem 3 and Theorem 4). The finite sample properties of the NN estimators are first illustrated on simulated data in the unconditional case (Section 6) where they are compared to extreme-value competitors. Second, the conditional neural network estimators are tested on real data (Section 7) which consist in daily rainfall measurements between the years 1958 and 2000 among 524 stations in the southern part of France, see Figure 1. Proofs and algorithms are postponed to the Appendix.

2 Extrapolation principle for estimating extreme quantiles

Let $X_1, \ldots, X_n$ be an i.i.d sample from an unknown cumulative distribution function (c.d.f) F. The associated order statistics are denoted by $X_{1,n} \leq \cdots \leq X_{n,n}$. We are interested in the estimation of the quantile function defined by $q(\cdot) := F^{\leftarrow}(\cdot) = \inf\{x \in \mathbb{R} : F(x) \geq \cdot\}$, at the extreme level $1-\alpha_n$ *i.e.* such that $n\alpha_n \to 0$ as $n \to \infty$. This latter condition entails that $q(1-\alpha_n)$ is almost surely asymptotically larger than the sample maxima.

Heavy-tailed distributions. Focusing on distributions in the Maximum domain of attraction of Fréchet, it is known from [15, Theorem 1.2.1] and [15, Proposition B.1.9.9] that the tail quantile function $U(t) := q(1-1/t)$ defined for all $t > 1$, is regularly-varying with index $\gamma > 0$ (this property is denoted by $U \in \mathcal{RV}_\gamma$ in the sequel) *i.e.*

$$U(t) = t^\gamma L(t), \tag{1}$$

where γ is the so-called tail-index and $L \in \mathcal{RV}_0$ is a slowly-varying function at infinity *i.e.* L is positive and, for all $z > 0$,

$$\lim_{t\to\infty} \frac{L(tz)}{L(t)} = 1. \tag{2}$$

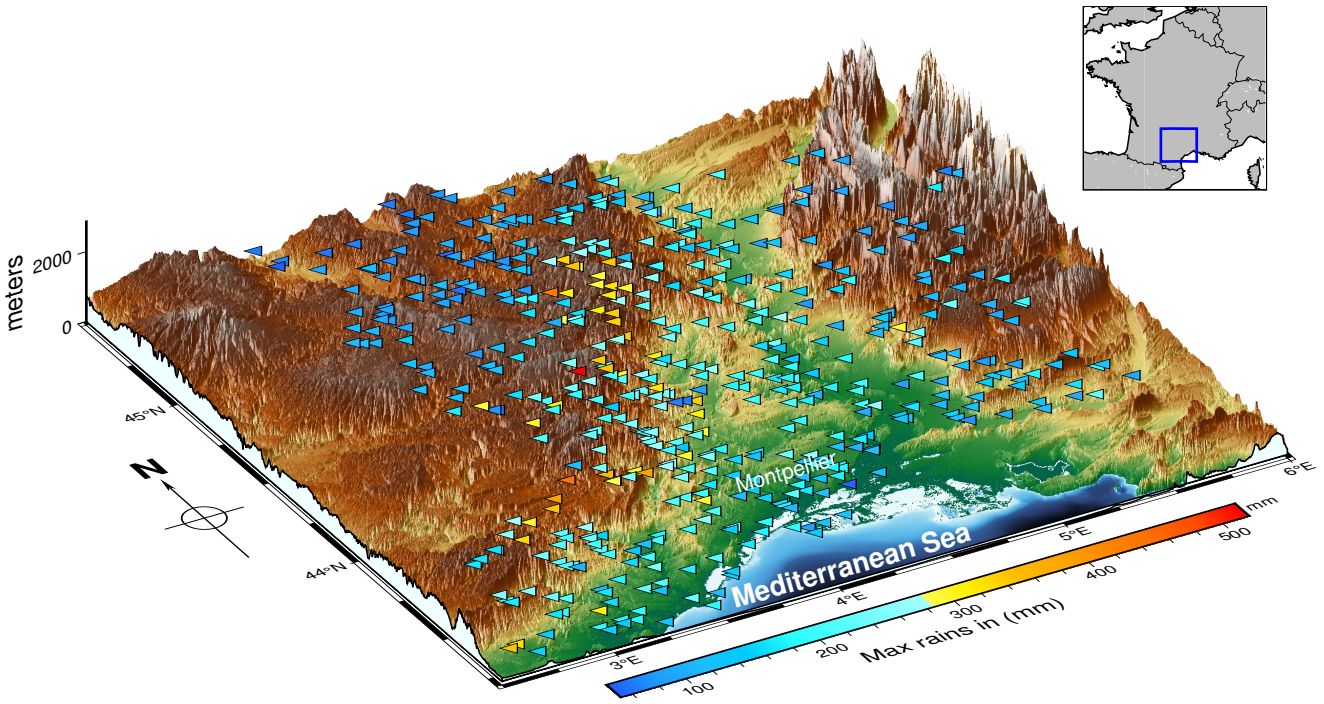


Figure 1: Historical (1958-2000) daily rainfall maxima in millimeters per station in the Cévennes-Vivarais region of France.

The index γ tunes the tail heaviness of F: the larger the index, the heavier the right tail. Examples include the (generalized) Pareto, Burr, Fisher, Inverse gamma and Student distributions, see Table 1 for the associated tail indices.

Distribution (parameters)	Density function	γ	ρ_2
Generalized Pareto ($\xi > 0$)	$(1+\xi t)^{-1-1/\xi}$, $t > 0$	ξ	$-\xi$
Burr ($\zeta, \theta > 0$)	$\zeta\theta t^{\zeta-1}\left(1+t^{\zeta}\right)^{-\theta-1}$, $t > 0$	$1/(\zeta\theta)$	$-1/\theta$
Fisher ($\nu_1, \nu_2 > 0$)	$\frac{(\nu_1/\nu_2)^{\nu_1/2}}{B(\nu_1/2,\nu_2/2)} t^{\nu_1/2-1}\left(1+\frac{\nu_1}{\nu_2}t\right)^{-(\nu_1+\nu_2)/2}$, $t > 0$	$2/\nu_2$	$-2/\nu_2$
Inverse Gamma ($\zeta > 0$)	$\frac{1}{\Gamma(\zeta)} t^{-\zeta-1}\exp(-1/t)$, $t > 0$	$1/\zeta$	$-1/\zeta$
Student ($\nu > 0$)	$\frac{1}{\sqrt{\nu}B\left(\nu/2,1/2\right)}\left(1+\frac{t^2}{\nu}\right)^{-\frac{\nu+1}{2}}$	$1/\nu$	$-2/\nu$

Table 1: Examples of heavy-tailed distributions satisfying the second-order condition (7) with the associated values of γ and ρ_2. Here, $\Gamma(\cdot)$ and $B(\cdot,\cdot)$ denote respectively the Gamma and Beta functions.

The idea underpinning the estimation is to take advantage of (1) to establish a link between the extreme quantile of interest $q(1-\alpha_n) = U(1/\alpha_n)$ and an intermediate one $q(1-\delta_n) = U(1/\delta_n)$ where δ_n is interpreted as an anchor level such that $k := \lfloor n\delta_n \rfloor \to \infty$ as $n \to \infty$. To this end, introduce the log-spacing function defined as

$$(x_1, x_2) \in \mathbb{R}_+^2 \mapsto f(x_1, x_2) = \log U(\exp(x_1 + x_2)) - \log U(\exp(x_2)) = \gamma x_1 + \varphi(x_1, x_2), \tag{3}$$

with

$$\varphi(x_1, x_2) := \log\left(\frac{L(\exp(x_1 + x_2))}{L(\exp(x_2))}\right). \tag{4}$$

It immediately follows that

$$q(1-\alpha_n) = q(1-\delta_n)\,(\delta_n/\alpha_n)^{\gamma} \exp\Big(\varphi\big(\log(\delta_n/\alpha_n), \log(1/\delta_n)\big)\Big).$$

An estimator of the extreme quantile can then be obtained using a two-step approach. First, a parametric model $\tilde{\varphi}_\theta$ is introduced for φ yielding a parametric approximation of $q(1-\alpha_n)$ with parameter $\phi = (\gamma, \theta)$:

$$\tilde{q}_\phi(1-\alpha_n; 1-\delta_n) = q(1-\delta_n)\,(\delta_n/\alpha_n)^{\gamma} \exp\Big(\tilde{\varphi}_\theta\big(\log(\delta_n/\alpha_n), \log(1/\delta_n)\big)\Big). \tag{5}$$

Second, for a given level δ_n, estimate both $q(1-\delta_n)$ by the associated order statistic $X_{n-k+1,n}$ and ϕ by a dedicated estimator to get:

$$\hat{q}_{\hat{\phi}}(1-\alpha_n; 1-\delta_n) = X_{n-k+1,n}\,(\delta_n/\alpha_n)^{\hat{\gamma}} \exp\Big(\tilde{\varphi}_{\hat{\theta}}\big(\log(\delta_n/\alpha_n), \log(1/\delta_n)\big)\Big),$$

where $\hat{\phi} = (\hat{\gamma}, \hat{\theta})$. See Figure 2 (left panel) for an illustration of (3) associated with a Burr distribution and its pointwise estimation based on order statistics.

Weissman estimator. In this setting, the simplest method consists in choosing $\tilde{\varphi}_\theta = 0$ in (5), so that $\phi = \gamma$, to get the so-called Weissman estimator [61]:

$$\hat{q}^{\mathtt{W}}_{\hat{\phi}}(1-\alpha_n; 1-\delta_n) = X_{n-k+1,n}\,(\delta_n/\alpha_n)^{\hat{\gamma}^{\mathtt{H}}(k)}, \tag{6}$$

where $\hat{\gamma}^{\mathtt{H}}(\cdot)$ is the Hill estimator [44]. This approach relies on the approximation of the slowly-varying function L in (4) by a constant, which can be not precise enough in practice.

Bias corrected estimators. The above expression (4) can be evaluated using the well-known second-order condition of the tail quantile function which states that there exist $\gamma > 0, \rho_2 \leq 0$ and a function A_2 positive or negative with $A_2(t) \to 0$ as $t \to \infty$ such that for all $z \geq 1$ [37, Equation (13)]

$$\log U(zt) - \log U(t) = \gamma \log z + A_2(t) \int_1^z z_2^{\rho_2 - 1}\,\mathrm{d}z_2 + o(A_2(t)), \qquad \text{as } t \to \infty. \tag{7}$$

Moreover, $|A_2|$ is regularly-varying with index ρ_2. This second-order parameter drives the bias of most extreme quantile estimators: the larger ρ_2 is, the larger the asymptotic bias. Assumption (7) is standard in extreme-value theory, since it controls the rate of convergence in (2). Examples of commonly used continuous distributions satisfying (7) can be found in [4, Section 3.3] and [15, Section 2.3], along with thorough discussions on the interpretation and the rationale behind this second-order condition. For instance, the (generalized) Pareto, Burr, Fréchet, Student, Fisher and Inverse-Gamma distributions all satisfy this condition, see Table 1.

Ignoring the $o(\cdot)$ term in (7) and assuming

$$A_2(t) = \gamma\beta_2 t^{\rho_2}, \tag{8}$$

with $\rho_2 < 0$ and $\beta_2 \neq 0$, give rise to the parametric model defined for every $x_1, x_2 \geq 0$ by

$$\tilde{\varphi}^{\mathtt{CW}}_\theta(x_1, x_2) = \gamma\beta_2 \exp(\rho_2 x_2)[\exp(\rho_2 x_1) - 1]/\rho_2, \tag{9}$$

with $\theta = (\beta_2, \rho_2)$. Replacing in (5) yields the quantile approximation

$$\tilde{q}^{\mathtt{CW}}_\phi(1-\alpha_n; 1-\delta_n) = q(1-\delta_n)\left(\frac{\delta_n}{\alpha_n}\right)^{\gamma} \exp\left(\gamma\beta_2\left(\frac{1}{\delta_n}\right)^{\rho_2} \frac{(\delta_n/\alpha_n)^{\rho_2} - 1}{\rho_2}\right), \tag{10}$$

and the associated Corrected Weissman estimator introduced in [38],

$$\hat{q}^{\mathtt{CW}}_{\hat{\phi}}(1-\alpha_n; 1-\delta_n) = X_{n-k+1,n}\left(\frac{\delta_n}{\alpha_n}\right)^{\hat{\gamma}} \exp\left(\hat{\gamma}\hat{\beta}_2\left(\frac{1}{\delta_n}\right)^{\hat{\rho}_2} \frac{(\delta_n/\alpha_n)^{\hat{\rho}_2} - 1}{\hat{\rho}_2}\right). \tag{11}$$

The quality of (11) hinges on a reliable estimation of γ and θ, see Paragraph 6.2 for details. In the following, we propose an extension of (10) to an higher order approximation and an estimation of the associated parameter θ by a neural network.

3 A neural network estimator of extreme quantiles

In a neural network (NN) setting (see for instance [39] for a general perspective), our purpose is to build an approximation of the log-spacing function (3) by taking advantage of higher order conditions on $U(\cdot)$ unlike classical bias-reduced estimators which are based on the second-order condition (7). Here, we focus on the class of one-hidden layer feedforward NN under the form

$$x \in \mathbb{R} \mapsto \sum_{i=1}^{d} \nu_i^{(1)} \sigma^{\mathrm{e}} \left(\nu_i^{(2)} x + \nu_i^{(3)} \right) \in \mathbb{R}, \tag{12}$$

with parameters $\{(\nu_i^{(1)}, \nu_i^{(2)}, \nu_i^{(3)}),\, i = 1, \ldots, d\} \in \Theta \subset \mathbb{R}^{3d}$ where d is the number of neurons in the hidden layer and with eLU (exponential linear unit) activation functions:

$$\sigma^{\mathrm{e}}(x) := \begin{cases} \exp(x) - 1 & , \quad x < 0 \\ x, & , \quad x \geq 0. \end{cases} \tag{13}$$

Let us first observe that $\tilde{\varphi}_\theta^{\mathtt{CW}}$ in (9) can be rewritten using two eLU functions as

$$\tilde{\varphi}_\theta^{\mathtt{CW}}(x_1, x_2) = \frac{\gamma \beta_2}{\rho_2} \left(\sigma^{\mathrm{e}}(\beta_2(x_1 + x_2)) - \sigma^{\mathrm{e}}(\beta_2 x_2) \right). \tag{14}$$

In order to build higher order approximations of $\varphi(x_1, x_2)$ using more than two activation functions, we consider a J-th order condition, introduced in [60] for all $J \geq 2$, on the tail quantile function. Assume there exist $\gamma > 0$ and, for all $j = 2, \ldots, J$, $\rho_j \leq 0$ as well as positive or negative functions A_j such that $A_j(t) \to 0$ as $t \to \infty$, $|A_j| \in \mathcal{RV}_{\rho_j}$, such that

$$\log U(tz) - \log U(t) = \gamma \log y + \sum_{j=2}^{J} \prod_{\ell=2}^{j} A_\ell(t) R_j(z) + o\left(\prod_{j=2}^{J} A_j(t) \right), \tag{15}$$

as $t \to \infty$ for all $z > 0$, where:

$$R_j(z) = \int_1^z z_2^{\rho_2 - 1} \int_1^{z_2} z_3^{\rho_3 - 1} \cdots \int_1^{z_{j-1}} z_j^{\rho_j - 1} \, \mathrm{d}z_j \ldots \mathrm{d}z_3 \, \mathrm{d}z_2. \tag{16}$$

Clearly, when $J = 2$, we recover the second-order condition (7). Moreover, $J = 3$ and $J = 4$ yield back respectively the third-order [22, 51] and fourth-order conditions [33]. In the following, we let $\bar{\rho}_J = \rho_2 + \cdots + \rho_J$. The next Proposition presents how, starting from the J-th order condition, a NN approximation of $\varphi(x_1, x_2)$ can be built using $J(J-1)$ eLU functions.

Proposition 1. *Assume the J-th order condition* (15) *holds for some* $J \geq 2$ *with*

$$A_j(t) = c_j t^{\rho_j}, \tag{17}$$

where $c_j \neq 0$ *and* $\rho_j < 0$ *for* $j = 2, \ldots, J$. *Let* $\tilde{\varphi}_\theta^{\mathtt{NN}_J}$ *be the function defined for* $x_1 > 0$ *and* $x_2 > 0$ *by*

$$\tilde{\varphi}_\theta^{\mathtt{NN}_J}(x_1, x_2) := \sum_{i=1}^{J(J-1)/2} w_i^{(1)} \left(\sigma^{e} \left(w_i^{(2)} x_1 + w_i^{(3)} x_2 \right) - \sigma^{e}(w_i^{(4)} x_2) \right), \tag{18}$$

for some $\theta = \left\{ (w_i^{(1)}, w_i^{(2)}, w_i^{(3)}, w_i^{(4)}), i = 1, \ldots, J(J-1)/2 \right\} \in \Theta := (\mathbb{R} \times \mathbb{R}_-^3)^{J(J-1)/2}$. *Then, for all* $\varepsilon > 0$, *there exists* $x_\varepsilon > 0$ *such that*

$$\varphi(x_1, x_2) = \tilde{\varphi}_\theta^{\mathtt{NN}_J}(x_1, x_2) + \Delta(\exp(x_1), \exp(x_2)) \prod_{j=2}^{J} A_j(\exp(x_2)),$$

with $|\Delta(\exp(x_1), \exp(x_2))| \leq \varepsilon \exp(x_1(\bar{\rho}_J + \varepsilon))$ *for all* $x_1, x_2 \geq x_\varepsilon$.

Hence, there exists a one-hidden layer eLU neural network approximation $\tilde{\varphi}_{\tilde{\theta}}^{\mathtt{NN}_J}$ of φ with the same representation as in (18), parameterized by some unknown $\tilde{\theta} \in \Theta$, and with a controlled error. Note that such a result is not a direct consequence of the Universal approximation Theorem [12] which ensures that a continuous function can be uniformly approximated on a compact set with arbitrary precision by a one hidden layer NN. Indeed, φ does not have a compact support and, moreover, the extrapolation framework makes necessary to control the approximation of $\varphi(x_1, x_2)$ when both x_1 and x_2 tend to infinity. Recall that the parametric model (18) encompasses (14) as a particular case when $J = 2$.

Second, for all $\tilde{\phi} = (\tilde{w}_0, \tilde{\theta}) \in \Phi := \mathbb{R}_+ \times \Theta$, consider the NN approximation of the log-spacing function

$$\tilde{f}_{\tilde{\phi}}^{\mathtt{NN}_J}(x_1, x_2) = \tilde{w}_0 x_1 + \tilde{\varphi}_{\tilde{\theta}}^{\mathtt{NN}_J}(x_1, x_2), \tag{19}$$

and, combining (5) with (19), the NN approximation of the extreme quantile is defined as

$$\tilde{q}_{\tilde{\phi}}^{\mathtt{NN}_J}(1-\alpha_n; 1-\delta_n) := q(1-\delta_n) \exp\left(\tilde{f}_{\tilde{\phi}}^{\mathtt{NN}_J}(\log(\delta_n/\alpha_n), \log(1/\delta_n))\right). \tag{20}$$

The approximating NN includes $d = J(J-1)$ neurons and CJ^2 parameters, where $C > 0$ is a constant independent of J. As a consequence of Proposition 1, we have the following convergence result on the NN approximation of the extreme quantile:

Theorem 2. *Assume the J-th order condition* (15) *holds together with (17) for some $J \geq 2$. Then, the one hidden-layer feedforward NN approximation* (20) *of the extreme quantile $q(1-\alpha_n)$ is such that*

$$\alpha_n^{\bar{\rho}_J} \inf_{\tilde{\phi} \in \Phi} \left| \log q(1-\alpha_n) - \log \tilde{q}_{\tilde{\phi}}^{\mathtt{NN}_J}(1-\alpha_n; 1-\delta_n) \right| \leq \bar{c}_J, \tag{21}$$

where $\bar{c}_J = c_2 \times \cdots \times c_J$, as $\alpha_n \to 0$ and $\delta_n/\alpha_n \to \infty$ when $n \to \infty$.

In view of (21), the error between the extreme log-quantile and its NN approximation is driven by $\bar{\rho}_J = \rho_2 + \cdots + \rho_J$. As expected, requesting higher regularity in the extreme-value model (through the J-th order condition) yields a smaller approximation error thanks to an increasing width of the proposed NN. We thus are in a position to defining the NN extreme quantile estimator

$$\hat{q}_{\hat{\phi}}^{\mathtt{NN}_J}(1-\alpha_n; 1-\delta_n) := X_{n-k+1,n} \exp\left(\tilde{f}_{\hat{\phi}}^{\mathtt{NN}_J}\left(\log\left(\delta_n/\alpha_n\right), \log\left(1/\delta_n\right)\right)\right),$$

where the estimated parameters $\hat{\phi} \in \Phi$ are computed thanks to the optimization process described in Paragraph 6.1.

4 Extrapolation principle for conditional extreme quantiles

Suppose now that X is a random variable associated with an explanatory random vector $Y \in \Pi \subset \mathbb{R}^{d_y}, d_y \geq 1$. Denoting by $F(\cdot \mid y)$ the conditional c.d.f of X given $\{Y = y\}$ for some $y \in \Pi$, the conditional quantile function is defined by $q(\cdot \mid y) = \inf\{x \in \mathbb{R} : F(x \mid y) \geq \cdot\}$ and the conditional tail quantile function is defined for all $t \geq 1$ by $U(t \mid y) := q(1 - 1/t \mid y)$. The usual unconditional extrapolation principles can be extended to this new setting basing on maximum domain attraction assumptions [13, 29]. More specifically, when the conditional distribution of X given $\{Y = y\}$ is assumed to be heavy-tailed, which is our framework in the sequel, the Weissman estimator (6) can be adapted as follows:

$$\hat{q}_{\hat{\phi}}^{\mathtt{W}}(1-\alpha_n; 1-\delta_n \mid y) = \hat{q}(1-\delta_n \mid y) \, (\delta_n/\alpha_n)^{\hat{\gamma}(y)},$$

see [14, 26]. The above conditional Weissman estimator relies on two quantities: $\hat{q}(1-\delta_n \mid y)$ which is an estimator of the intermediate conditional quantile $q(1-\delta_n \mid y)$ and $\hat{\gamma}(y)$, an estimator of the conditional tail-index $\gamma(y)$.

Estimation of intermediate conditional quantiles. Among the numerous methods dedicated to the estimation of conditional quantiles, two main lines of works can be identified. On the first hand, direct methods characterize the conditional quantile of level $\alpha \in (0,1)$ as the solution of an optimization problem:

$$q(1-\alpha \mid y) = \underset{\tau \in \mathbb{R}}{\arg\min}\, \mathbb{E}\left[\check{\rho}_{1-\alpha}\left(X-\tau\right) \mid Y = y\right],$$

where $v \in \mathbb{R} \mapsto \check{\rho}_{1-\alpha}(v) := v\left(1-\alpha-\mathbb{1}_{(-\infty,0]}\{v\}\right)$ is the so-called check-function. Estimators of the conditional quantile are then obtained by replacing the conditional expectation by some non-parametric estimator and solving the associated optimization problem, see among others [43, 46] for spline based methods and [64] for kernel smoothing techniques. On the other hand, the indirect method consists in first estimating the conditional c.d.f $F(\cdot \mid y)$, and then compute the estimated quantile via numerical inversion. Nonparametric estimators of $F(\cdot \mid y)$ include for instance kernel estimators [56] and nearest neighbor estimators [6].

Estimation of the conditional tail-index. Moving windows and nearest neighbors approaches have been developed in a fixed design setting [25, 26]. Kernel methods are proposed in [14, 30, 31, 34] to tackle the random design case. Finally, these methods have been adapted to the situation where the covariate is a random field or infinite dimensional, see respectively [52] and [27, 28].

In the next section, we show how to combine an indirect method to estimate the intermediate quantile (and more precisely, the nearest neighbor estimator, see Section 7) with a NN to estimate conditional extrapolation schemes following the ideas of Section 3. We also refer to [21, Section 3.5] for the approximation of the nearest neighbors distribution using the Hellinger distance and to [23] for the investigation of their asymptotic properties. Other indirect estimators of conditional extreme quantiles using nearest neighbor techniques are investigated in [26, 29] while direct estimators of conditional extreme quantiles are proposed in [58, 59].

5 Neural network estimators of conditional extreme quantiles

We present two approaches to estimate conditional extreme quantiles by a NN. The first one is the conditional extension of the model presented in Section 3. The second one takes advantage of a location-dispersion model assumption to get rid of the covariate in the extrapolation step.

5.1 Estimation with Conditional Extrapolation Neural Networks (CENN)

Similarly to (1), the conditional tail quantile function $U(\cdot \mid y)$ is assumed to be regularly-varying with a conditional tail-index $\gamma(y) > 0$ *i.e.* $U(t \mid y) = t^{\gamma(y)} L(t \mid y)$, where $L(\cdot \mid y)$ is a positive conditional slowly-varying function such that $\forall z > 0$,

$$\lim_{t \to \infty} \frac{L(tz \mid y)}{L(t \mid y)} = 1.$$

Similarly to (3), the conditional log-spacings function is defined as

$$\begin{aligned}(x_1, x_2, y) \in \mathbb{R}_+^2 \times \mathbb{R}^{d_y} \mapsto f(x_1, x_2 \mid y) &= \log U(\exp(x_1+x_2) \mid y) - \log U(\exp(x_2) \mid y) \\ &= \gamma(y) x_1 + \varphi(x_1, x_2 \mid y),\end{aligned}$$

with

$$\varphi(x_1, x_2 \mid y) := \log\left(\frac{L(\exp(x_1+x_2) \mid y)}{L(\exp(x_2) \mid y)}\right),$$

and it follows that

$$q(1-\alpha_n; 1-\delta_n \mid y) = q(1-\delta_n \mid y)\left(\delta_n/\alpha_n\right)^{\gamma(y)} \exp\Big(\varphi\big(\log\left(\delta_n/\alpha_n\right), \log\left(1/\delta_n\right) \mid y\big)\Big). \tag{22}$$

The same methodology as in Section 2 is applied here, where the conditional extension of the J-th order condition (15) can be written as

$$\log U(tz \mid y) - \log U(t \mid y) = \gamma(y) \log z + \sum_{j=2}^{J} \prod_{\ell=2}^{j} A_\ell(t \mid y) R_j(z \mid y) + o\left(\prod_{j=2}^{J} A_j(t \mid y) \right), \tag{23}$$

as $t \to \infty$ for all $z > 0$, with

$$R_j(z \mid y) = \int_1^z z_2^{\rho_2(y)-1} \int_1^{z_2} z_3^{\rho_3(y)-1} \cdots \int_1^{z_{j-1}} z_j^{\rho_j(y)-1} \, \mathrm{d}z_j \ldots \mathrm{d}z_3 \, \mathrm{d}z_2,$$

and

$$A_j(t \mid y) = c_j(y) t^{\rho_j(y)}, \tag{24}$$

for all $j = 2, \ldots, J$. Therefore, since all the parameters $\{(w_i^{(1)}, w_i^{(2)}, w_i^{(3)}, w_i^{(4)}), i = 1, \ldots, J(J-1)/2\}$ and γ depend now of the covariate, the idea is to replace in (18) and (19) each parameter by an appropriate NN in order to approximate the conditional quantity. Hence, we consider

$$\tilde{f}_{\tilde{\phi}}^{\mathtt{NN}_J}(x_1, x_2 \mid y) = \tilde{\varphi}_{\tilde{\theta}}^{\mathtt{NN}_J}(x_1, x_2 \mid y) + \tilde{w}_{\tilde{\theta}^{(0)}}^{\mathtt{NN}}(y) x_1,$$

with

$$\tilde{\varphi}_{\tilde{\theta}}^{\mathtt{NN}_J}(x_1, x_2 \mid y) := \sum_{i=1}^{J(J-1)/2} \tilde{w}_{\tilde{\theta}_i^{(1)}}^{\mathtt{NN}}(y) \left(\sigma^{\mathrm{e}} \left(\tilde{w}_{\tilde{\theta}_i^{(2)}}^{\mathtt{NN}}(y) x_1 + \tilde{w}_{\tilde{\theta}_i^{(3)}}^{\mathtt{NN}}(y) x_2 \right) - \sigma^{\mathrm{e}} \left(\tilde{w}_{\tilde{\theta}_i^{(4)}}^{\mathtt{NN}}(y) x_2 \right) \right),$$

where, for all $j \in \{1, \ldots, 4\}$ and $i \in \{1, \ldots, J(J-1)/2\}$, $\tilde{w}_{\tilde{\theta}^{(0)}}^{\mathtt{NN}}$ and $\tilde{w}_{\tilde{\theta}_i^{(j)}}^{\mathtt{NN}}$ are $2J(J-1)+1$ deep ReLU NNs with respectively $d^{(0)}$ and $d^{(j)}$ neurons in each of the $p^{(0)}$ and $p^{(j)}$ hidden layers. Recall that the ReLU activation function is defined by $x \in \mathbb{R} \mapsto \sigma^{\mathrm{R}}(x) = \max(x, 0)$. Unlike (12), we apply $-\sigma^{\mathrm{R}}(\cdot)$ in the output layer of $\tilde{w}_{\tilde{\theta}_i^{(2)}}^{\mathtt{NN}}, \tilde{w}_{\tilde{\theta}_i^{(3)}}^{\mathtt{NN}}, \tilde{w}_{\tilde{\theta}_i^{(4)}}^{\mathtt{NN}}$ in order to force a negative output. Thus, taking advantage of (22), we can build an approximation of the conditional extreme quantile $q(1-\alpha_n \mid y)$:

$$\tilde{q}_{\tilde{\phi}}^{\mathtt{NN}_J}(1-\alpha_n; 1-\delta_n \mid y) = q(1-\delta_n \mid y) \exp\left(\tilde{f}_{\tilde{\phi}}^{\mathtt{NN}_J}(\log(\delta_n/\alpha_n), \log(1/\delta_n) \mid y) \right). \tag{25}$$

The approximating NN includes $J(J-1)/2 \left(\sum_{j=1}^{4} p^{(j)} d^{(j)} + 2 \right) + d^{(0)}$ neurons and $C\, J^2 \sum_{j=1}^{4} p^{(j)} (d^{(j)})^2$ parameters, where $C > 0$ is a constant independent of $J, d^{(j)}, p^{(j)}$ for all $j = 1, \ldots, 4$. As a consequence of Proposition 1, we thus have the following convergence result on the NN approximation of the extreme quantile:

Theorem 3. *Assume the J-th order condition (23) holds together with (24) for some $J \geq 2$. Additionally, suppose all functions $w_i^{(1)}(\cdot), w_i^{()}(\cdot), w_i^{(3)}(\cdot), w_i^{(4)}(\cdot)), i = 1, \ldots, J(J-1)/2$, and $\gamma(\cdot)$ are continuous on the compact set $\Pi \subset \mathbb{R}^{d_y}$. Let $\bar{\rho}_{\mathtt{sup}} = \sup_{y \in \Pi} \bar{\rho}_J(y)$. Then, for all $y \in \Pi$, there exists a conditional deep feedforward NN approximation (25) of the conditional extreme quantile $q(1-\alpha_n \mid y)$ including $2J(J-1)+1$ sub-networks built for all $j \in \{0, \ldots, 4\}$ with fixed $d^{(j)} = 2d_y + 10$ number of neurons in each of the hidden layers of depths*

$$\begin{aligned} p_n^{(0)} = p_n^{(2)} &> c\, \alpha_n^{\bar{\rho}_{\mathtt{sup}}/2} \left(\log(\delta_n/\alpha_n) \right)^{1/2}, \\ p_n^{(1)} &> c\, \alpha_n^{\bar{\rho}_{\mathtt{sup}}/2}, \\ p_n^{(3)} = p_n^{(4)} &> c\, \alpha_n^{\bar{\rho}_{\mathtt{sup}}/2} \left(\log(1/\delta_n) \right)^{1/2}, \end{aligned}$$

where $c > 0$ is an arbitrary constant, $\alpha_n \to 0$ and $\delta_n/\alpha_n \to \infty$ as $n \to \infty$ such that

$$\alpha_n^{\bar{\rho}_{\mathtt{sup}}} \inf_{\tilde{\phi} \in \Phi} \sup_{y \in \Pi} \left| \log q(1-\alpha_n \mid y) - \log \tilde{q}_{\tilde{\phi}}^{\mathtt{NN}_J}(1-\alpha_n; 1-\delta_n \mid y) \right| = \mathcal{O}(1).$$

In this general conditional setting, a minimum depth (of magnitude $\simeq \alpha_n^{\bar{\rho}_{\text{sup}}/2}$) is required for the CENN to approximate the extreme quantile with a given error (of order $\simeq \alpha_n^{-\bar{\rho}_{\text{sup}}}$) while, in the previous situation, a one layer NN was sufficient. We are in a position to define the conditional NN extrapolation quantile estimator

$$\hat{q}_{\hat{\phi}}^{\texttt{NN}_J}(1-\alpha_n; 1-\delta_n \mid y) := \hat{q}(1-\delta_n \mid y) \exp\left(\tilde{f}_{\hat{\phi}}^{\texttt{NN}_J}\left(\log\left(\delta_n/\alpha_n\right), \log\left(1/\delta_n\right) \mid y\right)\right).$$

where $\hat{q}(1-\delta_n \mid y)$ is an estimator of the intermediate conditional quantile.

5.2 Estimation with Location-Dispersion Neural Networks (LDNN)

The location-dispersion regression model introduced in [57] assumes that

$$X = a(Y) + b(Y)Z, \tag{26}$$

where $a : \Pi \to \mathbb{R}$ and $b : \Pi \to \mathbb{R}^+$ are defined respectively as the regression and the dispersion functions while $Z \in \mathbb{R}$ is a real random variable. Denoting by $q_Z(\cdot)$ and $U_Z(\cdot)$ respectively the quantile and tail quantile functions of Z, it follows from (26) that

$$U(t \mid y) = a(y) + b(y)U_Z(t), \tag{27}$$

or equivalently $q(1-\alpha_n \mid y) = a(y) + b(y)q_Z(1-\alpha_n)$ and, therefore, considering three levels of quantiles $0 < \alpha_n < \delta_n < \tau_n < 1$ yields

$$\frac{q(1-\alpha_n \mid y) - q(1-\delta_n \mid y)}{q(1-\delta_n \mid y) - q(1-\tau_n \mid y)} = \frac{\frac{q_Z(1-\alpha_n)}{q_Z(1-\delta_n)} - 1}{1 - \frac{q_Z(1-\tau_n)}{q_Z(1-\delta_n)}} = \frac{\frac{U_Z(1/\alpha_n)}{U_Z(1/\delta_n)} - 1}{1 - \frac{U_Z(1/\tau_n)}{U_Z(1/\delta_n)}} = \frac{\exp\left(f_Z(\log(\delta_n/\alpha_n), \log(1/\delta_n))\right) - 1}{1 - \exp\left(f_Z(\log(\delta_n/\tau_n), \log(1/\delta_n))\right)},$$

where f_Z is defined similarly to (3) by

$$(x_1, x_2) \in \mathbb{R}_+^2 \mapsto f_Z(x_1, x_2) = \log U_Z(\exp(x_1 + x_2)) - \log U_Z(\exp(x_2)).$$

Let us stress that the above quantity does not depend on the covariate. Introducing

$$(x_1, x_2, x_3) \in \mathbb{R}_+^3 \mapsto g(x_1, x_2, x_3) = \frac{\exp(f_Z(x_1, x_2)) - 1}{1 - \exp(f_Z(x_3, x_2))},$$

one has

$$\frac{q(1-\alpha_n \mid y) - q(1-\delta_n \mid y)}{q(1-\delta_n \mid y) - q(1-\tau_n \mid y)} = g\big(\log(\delta_n/\alpha_n), \log(1/\delta_n), \log(\delta_n/\tau_n)\big),$$

and rearranging the terms yields

$$q(1-\alpha_n \mid y) = q(1-\delta_n \mid y)\left(1 + \left(1 - \frac{q(1-\tau_n \mid y)}{q(1-\delta_n \mid y)}\right) g(\log(\delta_n/\alpha_n), \log(1/\delta_n), \log(\delta_n/\tau_n))\right). \tag{28}$$

One can take advantage of (28) to estimate approximate conditional extreme quantiles from the location-dispersion regression model when Z is assumed to be heavy-tailed. We thus let $U_Z(t) = t^\gamma L_Z(t)$ with $\gamma > 0$ and $L_Z \in \mathcal{RV}_0$. It straightforwardly follows from (27) that $U(\cdot \mid y) \in \mathcal{RV}_\gamma$ meaning that X given $Y = y$ is heavy-tailed with tail-index independent of the covariate [1]. In other words, X inherits its tail behavior from Z and thus does not depend on the covariate y. Let us also note that, from (27), the regular variation property yields $U(t \mid y)/U_Z(t) \to b(y)$ as $t \to \infty$. The location-dispersion regression model (26) can thus be interpreted as a particular case of the proportional tails model [18]. It is a convenient way to model heteroscedastic extremes, see [16, 24, 32] for alternative solutions.

Following the methodology introduced in the unconditional case (see Section 3), it is possible to build an approximation of the conditional extreme quantile $q(1-\alpha_n \mid y)$ using two intermediate conditional quantiles $q(1-\delta_n \mid y)$ and $q(1-\tau_n \mid y)$:

$$\tilde{q}_{\tilde{\phi}}^{\mathtt{NN}_J}(1-\alpha_n; 1-\delta_n, 1-\tau_n \mid y) = q(1-\delta_n \mid y)\left(1+\left(1-\frac{q(1-\tau_n \mid y)}{q(1-\delta_n \mid y)}\right)\tilde{g}_{\tilde{\phi}}^{\mathtt{NN}_J}\big(\log(\delta_n/\alpha_n), \log(1/\delta_n), \log(\delta_n/\tau_n)\big)\right),$$

with

$$\tilde{g}_{\tilde{\phi}}^{\mathtt{NN}_J}\big(\log(\delta_n/\alpha_n), \log(1/\delta_n), \log(\delta_n/\tau_n)\big) = \frac{\exp\left(\tilde{f}_{\tilde{\phi}}^{\mathtt{NN}_J}(\log(\delta_n/\alpha_n),\, \log(1/\delta_n))\right) - 1}{1-\exp\left(\tilde{f}_{\tilde{\phi}}^{\mathtt{NN}_J}(\log(\delta_n/\tau_n),\, \log(1/\delta_n))\right)},$$

and where $\tilde{f}_{\tilde{\phi}}^{\mathtt{NN}_J}$ is defined in (18) and (19). The approximating NN includes $d = J(J-1)/2$ neurons and $C\,J^2$ parameters, where $C>0$ is a constant independent of J. We can thus extend the result of Theorem 2 in the conditional framework.

Theorem 4. *Assume* (26) *and conditions of Theorem 2 hold for U_Z. Suppose $a(\cdot)$ and $b(\cdot)$ are continuous functions on Π and that $b(\cdot)$ is lower bounded by a positive constant. Then, for all $y \in \Pi$, there exists a one hidden-layer feedforward neural network approximation $\tilde{q}_{\tilde{\phi}}^{\mathtt{NN}_J}(1-\alpha_n; 1-\delta_n, 1-\tau_n \mid y)$ of the conditional extreme quantile $q(1-\alpha_n \mid y)$ such that*

$$\inf_{\tilde{\phi}\in\Phi}\sup_{y\in\Pi}\left|\log q(1-\alpha_n \mid y) - \log \tilde{q}_{\tilde{\phi}}^{\mathtt{NN}_J}(1-\alpha_n; 1-\delta_n, 1-\tau_n \mid y)\right| = \mathcal{O}(\alpha_n^{-\bar{\rho}_J}) + \mathcal{O}(\tau_n^{-\bar{\rho}_J-\gamma}\delta_n^{\gamma}) \qquad (29)$$

with $\alpha_n \to 0$, $\delta_n/\tau_n \to 0$ and $\delta_n/\alpha_n \to \infty$ as $n \to \infty$.

It is then possible to tune the value of the additional sequence δ_n to balance both error terms in (29):

Corollary 5. *Assume the assumptions of Theorem 4 hold.*

- *If $\gamma + \bar{\rho}_J > 0$, then letting $\delta_n = \alpha_n^{-\bar{\rho}_J/\gamma}\tau_n^{1+\bar{\rho}_J/\gamma}$ yields*

$$\alpha_n^{\bar{\rho}_J}\inf_{\tilde{\phi}\in\Phi}\sup_{y\in\Pi}\left|\log q(1-\alpha_n \mid y) - \log \tilde{q}_{\tilde{\phi}}^{\mathtt{NN}_J}(1-\alpha_n; 1-\delta_n, 1-\tau_n \mid y)\right| = \mathcal{O}(1).$$

- *If $\gamma + \bar{\rho}_J \leq 0$, then letting $\delta_n = \xi_n\alpha_n$ and $\tau_n = \xi_n^2\alpha_n$ with $\xi_n \to \infty$ arbitrarily slowly as $n \to \infty$ yields*

$$\alpha_n^{\bar{\rho}_J}\inf_{\tilde{\phi}\in\Phi}\sup_{y\in\Pi}\left|\log q(1-\alpha_n \mid y) - \log \tilde{q}_{\tilde{\phi}}^{\mathtt{NN}_J}(1-\alpha_n; 1-\delta_n, 1-\tau_n \mid y)\right| = \mathcal{O}(\xi_n^{-2\bar{\rho}_J-\gamma}).$$

Up to the ξ_n term, one can recover the convergence rate $\alpha_n^{\bar{\rho}_J}$ of the unconditional case, see Theorem 2. The conditional NN extreme quantile estimator is defined as

$$\hat{q}_{\hat{\phi}}^{\mathtt{NN}_J}(1-\alpha_n; 1-\delta_n, 1-\tau_n \mid y) = \hat{q}(1-\delta_n \mid y)\left(1+\left(1-\frac{\hat{q}(1-\tau_n \mid y)}{\hat{q}(1-\delta_n \mid y)}\right)\tilde{g}_{\hat{\phi}}^{\mathtt{NN}_J}\big(\log(\delta_n/\alpha_n), \log(1/\delta_n), \log(\delta_n/\tau_n)\big)\right),$$

where again the intermediate conditional quantiles $q(1-\delta_n \mid y)$ and $q(1-\tau_n \mid y)$ can be estimated using the nearest neighbor estimator, see Section 7 for an illustration on real data.

6 Validation on simulated data (unconditional case)

The finite sample behviour of the (unconditional) extreme quantile NN estimator is illustrated on simulated data. To this end, we first describe both the model implementation and the model selection technique. Then, we briefly present some other bias-reduced estimators taken from the extreme-value literature. Next, we list the heavy-tailed distributions as well as the performance criteria used to compare all considered estimators.

6.1 Implementation of the NN estimator of extreme quantiles

Let us describe the implementation of the NN estimator of (unconditional) extreme quantiles introduced in Section 3. The NN approximation $\widetilde{f}^{\mathtt{NN}_J}_{\tilde{\phi}}$ of the log-spacing function is fitted to the data by minimizing some distance between two estimations of the $N = (n-1)(n-2)/2$ log-spacings:

$$\hat{\phi} = \underset{\tilde{\phi}\in\Phi}{\arg\min} \frac{1}{N} \sum_{k=2}^{n-1} \sum_{i=1}^{k-1} \left| \hat{S}_{i,k} - \widetilde{f}^{\mathtt{NN}_J}_{\tilde{\phi}}(\log(k/i), \log(n/k)) \right|^s, \ s \in \{1,2\}, \tag{30}$$

where, for $i = 1, \ldots, k-1$ and $k = 2, \ldots, n-1$, $\hat{S}_{i,k} := \log\left(X_{n-i+1,n}\right) - \log\left(X_{n-k+1,n}\right)$ is the empirical estimate of $\log q(1-i/n) - \log q(1-k/n)$.

All numerical experiments have been conducted on the Cholesky computing cluster from Ecole Polytechnique `http://meso-ipp.gitlab.labos.polytechnique.fr/user_doc`. It is composed by 4 nodes, where each one includes 2 CPU Intel Xeon Gold 6230 @ 2.1GHz, 20 cores and 4 Nvidia Tesla v100 graphics card. All the code was implemented in Python 3.8.2 and using the library PyTorch 1.7.1. We used the optimizer Adam [45] with default parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$ for all tests performed during $M = 500$ iterations. Additionally, the ranges of the neural network hyperparameters explored to find the best model are reported in Table 2. See Figure 2 (right panel) for an illustration of the NN estimation of the log-spacing function associated with a Burr distribution.

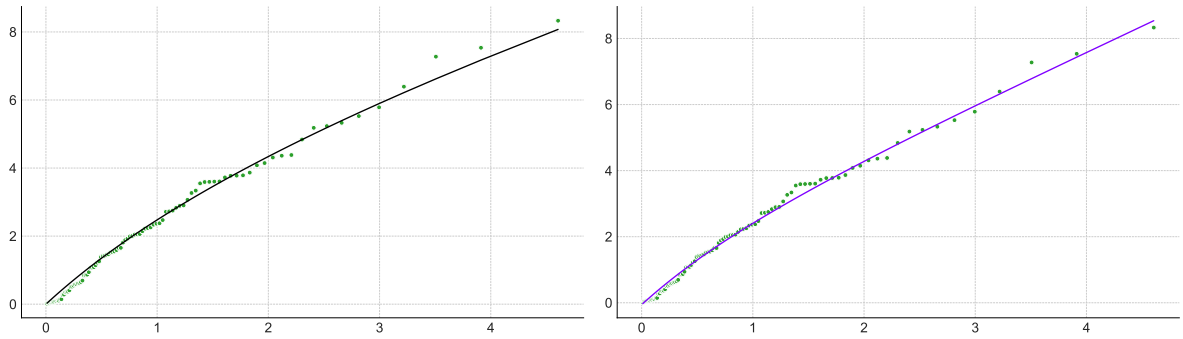


Figure 2: Log-spacing function associated with a Burr distribution ($\gamma = 1, \rho = -1/4$). Black curve: theoretical function $x_1 \mapsto f(x_1, \log(n/k))$, green dots: empirical pointwise estimation $(\log(k/i), \log X_{n-i+1,n} - \log X_{n-k+1,n})$, purple curve: NN estimation $x_1 \mapsto \widetilde{f}^{\mathtt{NN}_J}_{\hat{\phi}}(x_1, \log(n/k))$ with $i = 1, \ldots, k-1$, $k = 100$ and $n = 500$.

Setting	J	batch size	loss function
Non-conditional	$\{2,3,4,5\}$	$\{256, 512, 1024\}$	$s \in \{1,2\}$
Conditional	$\{2,3,4,5\}$	$\{256, 512, 1024\}]$	$s = 1$

Table 2: Hyperparameters ranges used for tuning NNs across the experiments.

Model selection Algorithm 1 selects the parameters $\hat{\phi} = \hat{\phi}_{m^\star}(\mathcal{A}^\star)$ associated with the best architecture $\mathcal{A}^\star$ in Table 2 and iteration $m^\star \in \{1, \ldots, M\}$ corresponding to the smallest mediane absolute deviation

$$\mathtt{MAD}\left(\left\{ \hat{q}^{\mathtt{NN}_J}_{\hat{\phi}_m}\left(1-\alpha_n; 1-\frac{k}{n}\right), \ k \in \{k_1, \ldots, k_2\} \right\} \right),$$

where, for any finite set $\mathcal{E} \subset \mathbb{R}$, the mediane absolute deviation is defined as

$$\mathtt{MAD}(\mathcal{E}) = \underset{\mathfrak{e}\in\mathcal{E}}{\text{median}} \left| \mathfrak{e} - \text{median}(\mathcal{E}) \right|. \tag{31}$$

In all the experiments, we used $k_1 = [3n/100]$ and $k_2 = [3n/4]$.

6.2 Competitors

Seven bias reduced extreme quantile estimators are considered. They can be sorted in two main families. First, one can plug a bias-reduced estimator of the tail-index γ in the Weissman estimator (6). As an example, basing on the second order condition (7) and (8), the Corrected-Hill estimator is proposed in [7]:

$$\hat{\gamma}^{\mathtt{CH}}(k) = \hat{\gamma}^{\mathtt{H}}(k)\left(1 - \frac{\hat{\beta}_2}{1-\hat{\rho}_2}\left(\frac{n}{k}\right)^{\hat{\rho}_2}\right),$$

where $\hat{\rho}_2$ and $\hat{\beta}_2$ are the respective estimators of the parameters ρ_2 and β_2. Similarly, [36] and [35] introduced a tuning parameter $p \geq 0$ in $\hat{\gamma}^{\mathtt{CH}}$ to get respectively the reduced-bias mean-of-order-p and partially reduced-biais estimators denoted by $\hat{\gamma}^{\mathtt{CH}_p}$ and $\hat{\gamma}^{\mathtt{PRB}_p}$. To select p in $\hat{\gamma}^{\mathtt{CH}_p}$ and $\hat{\gamma}^{\mathtt{PRB}_p}$, one can either follow a path stability criterion [36, Algorithm 4.2] or plug an "optimal" deterministic value [36, Page 1739], denoted by $p^\star$. In this latter case, the corresponding estimators are denoted by $\hat{\gamma}^{\mathtt{CH}_{p^\star}}$ and $\hat{\gamma}^{\mathtt{PRB}_{p^\star}}$.

Second, one may reduce simultaneously the extrapolation bias and the bias coming from the estimation of the tail-index. This idea is implemented in the Corrected Weissman estimator (CW), see (11), discussed in Section 2. More recently, a Refined Weissman (RW) estimator has been proposed in [2] featuring an adapted choice of the intermediate sequence $k^{\mathtt{H}}$ in the Hill estimator

$$k^{\mathtt{H}} = k\left(\frac{-\hat{\rho}_2}{1-\hat{\rho}_2}\frac{\log\left(k/(n\alpha_n)\right)}{1-(k/(n\alpha_n))^{\hat{\rho}_2}}\right)^{1/\hat{\rho}_2},$$

different from the intermediate sequence k used in the intermediate quantile estimator.

Replacing the Hill estimator in (6) by $\hat{\gamma}^{\mathtt{CH}}(k), \hat{\gamma}^{\mathtt{CH}_p}(k), \hat{\gamma}^{\mathtt{PRB}_p}(k), \hat{\gamma}^{\mathtt{CH}_{p^\star}}(k), \hat{\gamma}^{\mathtt{PRB}_{p^\star}}(k), \hat{\gamma}^{\mathtt{H}}(k^{\mathtt{H}})$ leads respectively to the estimators $\hat{q}^{\mathtt{CH}}_{\hat{\phi}}, \hat{q}^{\mathtt{CH}_p}_{\hat{\phi}}, \hat{q}^{\mathtt{PRB}_p}_{\hat{\phi}}, \hat{q}^{\mathtt{CH}_{p^\star}}_{\hat{\phi}}, \hat{q}^{\mathtt{PRB}_{p^\star}}_{\hat{\phi}}, \hat{q}^{\mathtt{RW}}_{\hat{\phi}}$. See [2] for a detailed account on these bias-reduced extreme quantile estimators.

6.3 Experimental design

The comparative study is achieved on six heavy-tailed distributions. The first five distributions: Burr, Fréchet, Fisher, generalized Pareto distribution (GPD), Inverse Gamma, and Student belong to the Hall-Welsh class [40, 41] which assumes that there exist $c_1 > 0, c_2 \neq 0$ such that

$$U(t) = c_1 t^\gamma \left(1 + c_2 t^{\rho_2} + o(t^{\rho_2})\right).$$

These five distributions satisfy the second-order condition (7) with (8), see Table 1 for their definitions and associated values of γ and ρ_2. The sixth distribution, denoted by NHW(γ, ρ_2), is defined for all $\gamma \geq \exp(-2)/2$ and $\rho_2 < 0$ by its tail quantile function $U(t) = t^\gamma \exp(A_2(t)/\rho_2)$ where $A_2(t) = \rho_2 t^{\rho_2}\log(t)/2$, $t \geq 1$, is the auxiliary function associated with the second-order condition (7). It thus appears that the NHW distribution does not belong to the Hall-Welsh class and does not verify (8) either.

Based on the simulation study of [2], we focus on the following settings corresponding to the most challenging situations for extreme-value estimators: large values of γ and/or large values of ρ_2:

- Burr distribution $\gamma \in \{1/8, 1/4, 1/2\}$ and $\rho_2 = -1/8$,
- NHW distribution $\gamma = 1$ and $\rho_2 \in \{-1/8, -1/4, -1/2, -1, -2\}$.
- Fisher distribution with $\nu_1 = 1$ and $\nu_2 \in \{2, 16\}$ leading to $(\gamma, \rho_2) \in \{1/8, 1\} \times \{-1/8, -1\}$,
- GPD with $\gamma = 1/8$ leading to $\rho_2 = -1/8$,
- Inverse Gamma distribution with $\zeta = 1$ leading to $\gamma = 1$ and $\rho_2 = -1$,
- Student distribution with $\nu = 1$ yielding $\gamma = 1$ and $\rho_2 = -2$.

Note that the case $\gamma = 1$ in the Fréchet and GPD distributions coincides respectively with the Inverse Gamma and Burr distributions (see Table 1). For each of these 21 considered configurations, $R = 500$ replicated data sets of size $n = 500$ are simulated and the associated extreme quantile of order $1-\alpha_n = 1 - 1/(2n)$ is estimated using the NN estimator, the Weissman estimator and the seven bias-reduced estimators described in the above paragraph.

The performance of the extreme quantile estimators is assessed using the Relative median-squared error (RMedSE):

$$\text{RMedSE}\left(\hat{q}_{\hat{\phi}}, \frac{1}{2n}\right) = \underset{r\in\{1,\dots,R\}}{\text{median}} \left[\left(\frac{\hat{q}_{\hat{\phi}}^{(r)}(1-\frac{1}{2n}; 1-\frac{k^\star(r)}{n})}{q(1-\frac{1}{2n})} - 1 \right)^2 \right], \tag{32}$$

where $\hat{q}_{\hat{\phi}}^{(r)}(1-\frac{1}{2n}; 1-\frac{k^\star(r)}{n})$ denotes an estimator of $q(1-\frac{1}{2n})$ (either the NN estimator or some of its competitors) computed with the anchor index $k^\star(r)$ selected using [2, Algorithm 1] with initial points $a^{(0)} = [3n/100]$ and $c^{(0)} = [3n/4]$ on the rth replication, $r \in \{1, \dots, R\}$.

6.4 Results

The RMedSE results are provided in Table 3 for all considered distributions. It appears that the NN approach is an efficient tool for estimating extreme quantiles in difficult heavy-tailed situations where other estimators almost all fail. The NN estimator indeed provides the best results in 12 out of 21 times. As a comparison, RW, CW, W and $\text{PRB}_{p^\star}$ estimators provide the best results respectively only in 3, 3, 2 and 1 out of 21 times. Moreover, Figure 3 illustrates that the NN estimator features a nice stability in terms of bias and RMedSE for a wide range of k values on selected situations from Table 3. This phenomenon may be highly appreciated even when our estimator is not ranked first on the RMedSE criteria basis, see for instance the top pannel of Figure 3.

As a conclusion, even though the NN method is numerically more expensive than its competitors, it provides a very effective estimator for all heavy-tailed situations. Additionally, note that in the above simulations, the NNs are built with only 2 to 20 neurons (recall that $J \in \{2, \dots, 5\}$), which remains acceptable from the computational cost point of view.

7 Illustration on rainfall data (conditional case)

The conditional NN estimators are tested on daily rainfall observations from 1958 to 2000 in the Cévennes-Vivarais region (southern part of France), see Figure 1. The region covers 256×283 km^2 where the Rhône River flows between two major mountainous massifs: the Massif Central and the Alps, respectively in the western and eastern sides. The northwestern quarter is a quite homogeneous high plateau, whereas the southern part is a large river plain bordered by the Mediterranean Sea. This region is historically very sensitive to extreme precipitations and flash floods [50]. More precisely, the rainfall distribution exhibits different statistical properties, depending on both the time scale (whether hourly or daily) and the spatial scale (whether in a flat region closer to the sea, or further in the mountains) [9]. Although daily rainfall maxima used to be modeled with a Gumbel distribution (exponential tails) [47, Section 7.2.2], better fits with heavy-tailed distributions are now preferred in order to tackle the underestimation of the extreme rainfall levels with light-tailed distributions [9, 48].

The dataset is provided by the French meteorological service Météo-France and includes both the $n_D = 15,706$ daily rainfall measurements in millimeters and the location of $n_S = 524$ stations, leading to a dataset of size $n = n_D \times n_S$. We observe that the daily rainfalls are the highest over the eastern slope of the Massif Central (Cévennes mountains range), which is a known phenomena in this region [50]. In this context, the variable of interest X is the one-dimensional daily rainfall and the covariate Y is the three-dimensional geographical location (longitude, latitude and altitude).

	NN	W	RW	CW	CH	CH_p	PRB_p	$\text{CH}_{p^\star}$	$\text{PRB}_{p^\star}$
Burr ($\gamma = 1/8$)									
$\rho = -1/8$	0.0392	-	**0.0364**	-	0.5375	0.2713	0.3745	0.3578	0.1203
Burr ($\gamma = 1/4$)									
$\rho = -1/8$	0.1567	-	**0.1421**	-	-	-	-	-	0.6357
Burr ($\gamma = 1/2$)									
$\rho = -1/8$	**0.2847**	-	0.4298	-	-	-	-	-	-
Burr ($\gamma = 1$)									
$\rho = -1/8$	**0.3133**	-	0.8625	-	-	-	-	-	-
$\rho = -1/4$	**0.1962**	-	0.5423	-	-	-	-	-	0.6617
$\rho = -1/2$	0.2142	-	0.3291	-	0.0949	0.1021	0.1488	**0.0874**	0.1185
$\rho = -1$	0.1877	-	0.2438	**0.1289**	0.4120	0.3737	0.3761	0.3658	0.4261
$\rho = -2$	**0.1432**	0.2065	0.1488	0.2115	0.3394	0.3384	0.2893	0.2933	0.3058
NHW ($\gamma = 1/8$)									
$\rho = -1/8$	**0.0275**	-	0.0340	0.0699	0.2442	0.2194	0.3285	0.2202	0.3157
NHW ($\gamma = 1/4$)									
$\rho = -1/8$	**0.0570**	-	0.0816	0.1482	0.3290	3209	0.3890	0.3212	0.3935
NHW ($\gamma = 1/2$)									
$\rho = -1/8$	**0.1168**	-	0.1794	0.3683	0.5309	0.5284	0.5697	0.5155	0.5586
NHW ($\gamma = 1$)									
$\rho = -1/8$	**0.2709**	-	0.3885	0.5644	0.7789	0.7016	0.7379	0.7891	0.8039
$\rho = -1/4$	**0.2163**	-	0.3095	0.4888	0.6851	0.6920	0.6825	0.6897	0.7252
$\rho = -1/2$	**0.1615**	0.5927	0.2217	0.2481	0.4589	0.4939	0.4803	0.4595	0.4727
$\rho = -1$	0.1596	**0.0679**	0.1557	0.1549	0.2340	0.2582	0.444	0.2302	0.2353
$\rho = -2$	0.1082	**0.0738**	0.1302	0.1576	0.2112	0.1953	0.2080	0.1865	0.1879
Fisher ($\rho = -\gamma$)									
$\gamma = 1/8$	**0.0506**	-	0.0765	-	-	-	-	-	0.6126
$\gamma = 1$	0.1792	-	0.2871	**0.0882**	0.2722	0.2736	0.2323	0.2378	0.3409
GPD ($\rho = -\gamma$)									
$\gamma = 1/8$	0.0391	-	**0.0364**	-	0.5375	0.2534	0.3266	0.3578	0.1203
Inverse Gamma ($\rho = -\gamma$)									
$\gamma = 1$	0.1863	0.9259	0.1731	**0.1269**	0.2163	0.2317	0.2232	0.2030	0.2181
Student ($\rho = -2\gamma$)									
$\gamma = 1$	**0.1515**	0.5781	0.1961	0.3565	0.5654	0.5024	0.5273	0.5157	0.5439

Table 3: RMedSE associated with nine estimators of the extreme quantile $q(\alpha_n = 1/(2n))$ on six heavy-tailed distributions. The best result is emphasized in bold. RMedSEs larger than 1 are not reported.

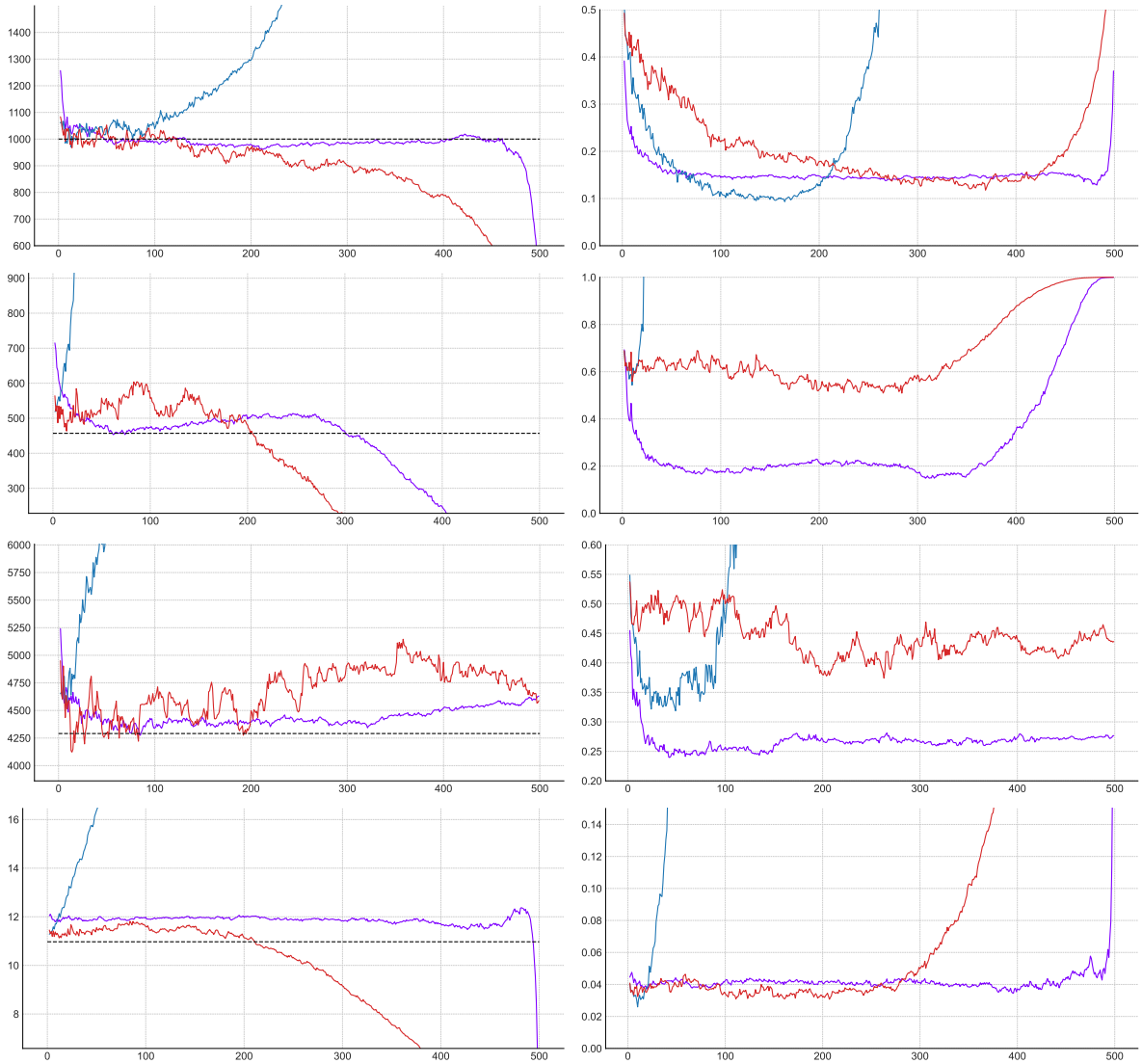


Figure 3: Illustration on simulated data sets of size $n = 500$ from a Burr distribution with $\gamma = 1$ and $\rho \in \{-2, -1/4\}$, a NHW distribution with $\gamma = 1$ and $\rho = -1/8$, and a GPD distribution with $\gamma = 1/8$ (from top to bottom). Median of the estimators (left panel) of the extreme quantile (black dashed line) at level $1 - \alpha_n = 1 - 1/(2n)$ and RMedSE (right panel), as functions of $k \in \{2, \ldots, n-1\}$, computed on $R = 500$ replications, associated with W (blue), RW (red) and NN (purple) estimators.

7.1 Data processing

As mentioned in Section 4, the conditional intermediate quantiles can no longer be estimated globally by order statistics. Therefore, the idea is to consider a small neighborhood around the geographical location of interest y and estimate $q(\cdot \mid y)$ locally by order statistics. To define this neighborhood, we fixed the number of neighbors n_K and apply the nearest neighbors estimator on the covariate Y to cluster all the stations using the Mahalanobis distance $D(Y_t, Y_{t'}) := \sqrt{(Y_t - Y_{t'})^\top \Sigma^{-1} (Y_t - Y_{t'})}$ for all $(t, t') \in \{1, \dots, n_S\}^2$, where Σ^{-1} is the inverse of the corresponding covariance matrix. Next, we merge all the historical values of the $n_K - 1$ closest stations of each station $t \in \{1, \dots, n_S\}$, leading to $n_o = n_D \times n_K$ observations which are assumed to be i.i.d within each neighborhood. We denote by by $X^{(1,n_o)}(Y_t) \leq \cdots \leq X^{(n_o,n_o)}(Y_t)$ the order statistics associated with a given station $t = 1, \dots, n_S$. In addition, we introduce $n_h \in \{1, \dots, n_o - 1\}$ and focus on the highest unique historical rainfalls $\left(X^{(n_o-i+1,n_o)}(Y_t), i = 1, \dots, n_h\right)$, for each station $t = 1, \dots, n_S$. The estimation of the conditional extreme quantile is investigated at level $1 - \alpha_n = 1 - 1/n_o$ by storing in a test set all maximum order statistics $X^{(n_o,n_o)}(Y_t)$ for further comparison; and keeping the remaining $(n_h - 1)$ ones in a train set for computation of the estimates.

Tail-index estimation Before moving to the implementation of the conditional extrapolation neural networks, it is necessary to check whether the data are heavy-tailed. Additionally, we verify that the tail-index γ is independent from the covariate Y in all the considered n_K-neighborhoods as assumed in the location-dispersion model of Section 5.2. To this end, we first fix $n_h = 100$ and, for $n_K \in \{10, 15, \dots, 50\}$, compute the Hill estimator $\hat{\gamma}_t^{\mathtt{H}}(k^\star)$, where $k^\star$ is selected by [2, Algorithm 1], for each station $t = 1, \dots, n_S$ (see Figure 4a for illustration on one station). Based on a graphical diagnosis (Figure 4b), we select $n_K = 45$ which highlights the lowest spread and skewness of the Hill estimates. The distribution of the estimated tail indices obtained with $n_h = 100$ and $n_K = 45$ (Figure 4c) has a small standard-deviation (0.031) around its mean (0.189), which confirms the hypothesis of a constant tail-index in the Cévennes-Vivarais region. Second, we validate the choice of $n_h = 100$ graphically (Figure 4d) leading to a small standard-deviation (0.029) of the slopes associated with the quantile-quantile plots around their mean (0.195).

In the next two paragraphs, we propose an estimation of the conditional extreme quantile $q(1 - 1/n_o \mid Y_t)$ at each station $t = \{1, \dots, n_S\}$ based on the two methodologies discussed in Section 5.1 and 5.2. Even if the assumption on γ is imposed only in the location-dispersion model, we keep the same dataset built with $n_h = 100$ and $n_K = 45$ for both approaches.

7.2 Conditional Extrapolation Neural Network (CENN)

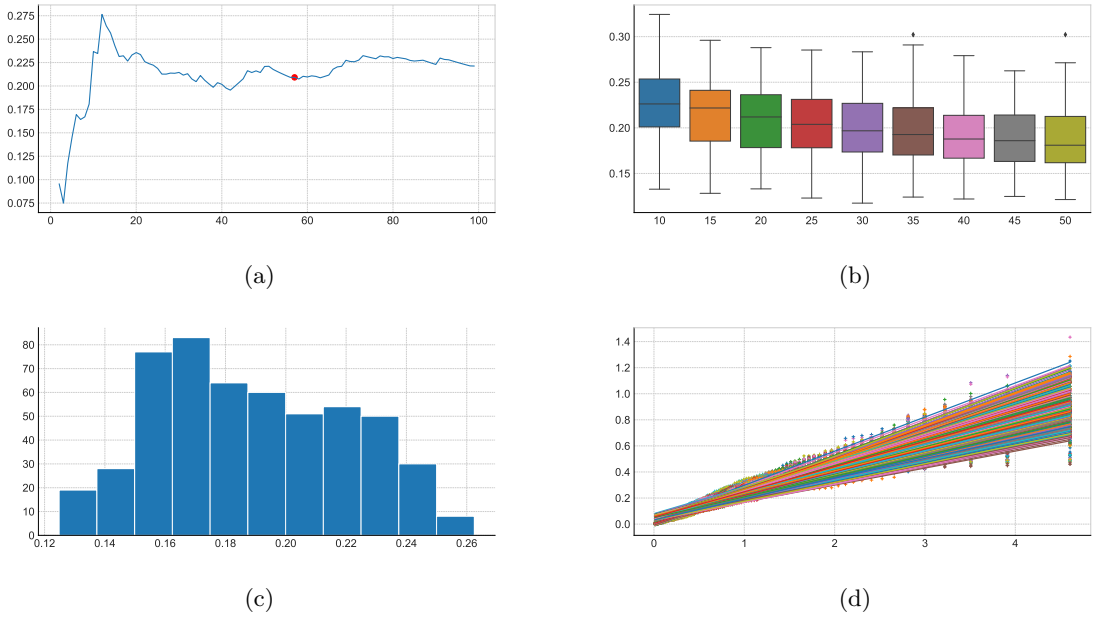
Let us describe the implementation of the NN estimator of the (conditional) extreme quantile introduced in Section 4. Starting from the real data processed in the previous paragraph, first normalize the covariate Y between 0 and 1 for a training stability purposes [49]. Second, compute within each neighborhood $t = 1, \dots, n_S$ the $N = n_S n_h (n_h - 1)/2$ empirical estimates

$$\hat{S}^{(i,k)}(Y_t) := \log\left(X^{(n_o-i+1,n_o)}(Y_t)\right) - \log\left(X^{(n_o-k+1,n_o)}(Y_t)\right),$$

of the conditional log-spacings $\log q(1 - i/n_o \mid Y_t) - \log q(1 - k/n_o \mid Y_t)$, for $i = 1, \dots, k-1$ and $k = 2, \dots, n_h$. Next, build both the test set containing the $N^{\text{test}} = n_S(n_h - 1)$ empirical estimates $\hat{S}^{(1,k)}(Y_t)$ of the log-spacings, and the train set containing the remaining $N^{\text{train}} = n_S(n_h - 1)(n_h - 2)/2$ ones. Thus, the conditional NN approximation $\widetilde{f}_{\tilde{\phi}}^{\mathtt{NN}_J}$ of the conditional log-spacing function is fitted to the training data by minimizing the L_1 distance between two estimations of the log-spacings

$$\hat{\phi} = \underset{\tilde{\phi} \in \Phi}{\arg\min} \frac{1}{N^{\text{train}}} \sum_{t=1}^{n_S} \sum_{k=3}^{n_h} \sum_{i=2}^{k-1} \left| \hat{S}^{(i,k)}(Y_t) - \widetilde{f}_{\tilde{\phi}}^{\mathtt{NN}_J}(\log(k/i), \log(n_o/k), Y_t) \right|. \tag{33}$$

Model selection Algorithm 2 selects the parameters $\hat{\phi} = \hat{\phi}_{m^\star}(\mathcal{A}^\star)$ associated with the best architecture $\mathcal{A}^\star$ in Table 2 and iteration $m^\star \in \{1, \dots, M\}$ corresponding to the smallest mediane MAD



(a) (b) (c) (d)

Figure 4: Illustrations on real data. Top left: Example of Hill estimator as a function of $k = 2, \ldots, n_h - 1$, within the neighborhood of a given station with $n_h = 100$ and $n_K = 45$. The selected $k^\star$ is depicted by the red circle. Top right: Box-plots of estimated $\hat{\gamma}^{\texttt{H}}$'s as functions of n_K with $n_h = 100$. Bottom left: Histogram of estimated $\hat{\gamma}^{\texttt{H}}$'s for all stations $t = 1, \ldots, n_S$ with $n_K = 45$ and $n_h = 100$. Bottom right: quantile-quantile plot $\log(n_h/i) \mapsto \log(X^{(n_o-i+1,n_o)}(Y_t)) - \log(X^{(n_o-n_h+1,n_o)}(Y_t))$ for all $t = 1, \ldots, n_S$, $i = 1, \ldots, n_h - 1$ with $n_h = 100$ and $n_K = 45$.

over all stations:

$$\underset{t\in\{1,\ldots,n_S\}}{\text{median}}\ \texttt{MAD}\left(\left\{\hat{q}^{\texttt{NN}_J}_{\hat{\phi}_m}\left(1-\frac{1}{n_o};1-\frac{k}{n_o}\,|\,Y_t\right),\ k\in\{k_1,\ldots,k_2\}\right\}\right),$$

see (31) for the definition of the MAD operator. In all experiments we used $k_1 = [3n_h/100]$ and $k_2 = [3n_h/4]$. The performance criteria (32) is adapted to the conditional case as

$$\text{RMedSE}\left(\hat{q}^{\texttt{NN}_J}_{\hat{\phi}_{m^\star}}, t, \frac{1}{n_o}\right) = \underset{t\in\{1,\ldots,n_S\}}{\text{median}}\left(\frac{\hat{q}^{\texttt{NN}_J}_{\hat{\phi}_{m^\star}}(1-\frac{1}{n_o};1-\frac{k^\star(t)}{n_o}\,|\,Y_t)}{X^{(n_o,n_o)}(Y_t)}-1\right)^2, \tag{34}$$

where $\hat{q}^{\texttt{NN}_J}_{\hat{\phi}_{m^\star}}(1-1/n_o;1-k^\star(t)/n_o\,|\,Y_t)$ denotes the NN estimation computed on the selected anchor index $k^\star(t) \in \{k_1, \ldots, k_2\}$ using [2, Algorithm 1] with initial points $a^{(0)} = k_1$ and $c^{(0)} = k_2$, at the t-th station, $t \in \{1, \ldots, n_S\}$.

7.3 Location-Dispersion Neural Network (LDNN)

Let us describe the implementation of the Location-Dispersion NN estimator of the conditional extreme quantile introduced in Section 5.2. Starting from the real data processed in Section 7.1, first compute the $N = n_S n_h(n_h-1)(n_h-2)/6$ empirical estimates

$$\hat{G}^{(i,j,k)}(Y_t) := \frac{X^{(n_o-i+1,n_o)}(Y_t) - X^{(n_o-k+1,n_o)}(Y_t)}{X^{(n_o-k+1,n_o)}(Y_t) - X^{(n_o-j+1,n_o)}(Y_t)}.$$

of the modified conditional spacings

$$(i,k,j,t) \mapsto \frac{q(1-i/n_o\,|\,Y_t) - q(1-k/n_o\,|\,Y_t)}{q(1-k/n_o\,|\,Y_t) - q(1-j/n_o\,|\,Y_t)},$$

within each neighborhood $t = 1, \ldots, n_S$ for all $i = 1, \ldots, k-1$, $k = 2, \ldots, j-1$ and $j = 3, \ldots, n_h$. Next, perform a similar train-test splitting as the one in Section 5.1 , resulting in $N^{\text{train}} = n_S(n_h - 1)(n_h - 2)(n_h - 3)/6$ and $N^{\text{test}} = n_S(n_h - 1)(n_h - 2)/2$. Thus, the Location-Dispersion NN approximation $\tilde{g}_{\tilde{\phi}}^{\texttt{NN}_J}$ of g is fitted to the training data by minimizing the L_1 distance between two estimations of the spacings

$$\hat{\phi} = \underset{\tilde{\phi} \in \Phi}{\arg\min} \frac{1}{N^{\text{train}}} \sum_{t=1}^{n_S} \sum_{j=4}^{n_h} \sum_{k=3}^{j-1} \sum_{i=2}^{k-1} \left| \hat{G}^{(i,j,k)}(Y_t) - \tilde{g}_{\tilde{\phi}}^{\texttt{NN}_J}(\log(k/i), \log(n_o/k), \log(k/j)) \right|, \tag{35}$$

where

$$\tilde{g}_{\tilde{\phi}}^{\texttt{NN}_J}(\log(k/i), \log(n_o/k), \log(k/j)) = \frac{\exp\left(\tilde{f}_{\tilde{\phi}_1}^{\texttt{NN}_J}(\log(k/i), \log(n_o/k))\right) - 1}{1 - \exp\left(\tilde{f}_{\tilde{\phi}_2}^{\texttt{NN}_J}(\log(k/j), \log(n_o/k))\right)} \tag{36}$$

is the Location-Dispersion NN approximation with $\tilde{\phi} = \{\tilde{\phi}_1, \tilde{\phi}_2\}$. For a larger flexibility, we built two NN in (36) with a similar architecture but with a different initialization of weights $\{\tilde{\phi}_1, \tilde{\phi}_2\}$. During the training, it may happen that $\hat{G}^{\texttt{NN}_J,(i,j,k)}(\tilde{\phi})$ is not defined if $\tilde{f}_{\tilde{\phi}}^{\texttt{NN}_J}(\log(k/j), \log(n_o/k)) = 0$ in (36) for some pair (k, j). In this case, we do not take into account the gradient associated with these inputs in the optimization part.

Model selection Similarly to the previous case, Algorithm 3 selects the parameters $\hat{\phi} := \hat{\phi}_{m^\star}(\mathcal{A}^\star)$ associated with the best architecture $\mathcal{A}^\star$ in Table 2 and iteration $m^\star \in \{1, \ldots, M\}$ corresponding to the median MAD:

$$\underset{t \in \{1,\ldots,n_S\}}{\text{median}} \texttt{MAD}\left(\left\{\hat{q}_{\hat{\phi}_m}^{\texttt{NN}_J}\left(1 - \frac{1}{n_o}; 1 - \frac{k}{n_o}; 1 - \frac{j}{n_o} \mid Y_t\right),\ k \in \{k_1, \ldots, k_2\}, j' \in \{j_1, \ldots, j_2\}, k < j\right\}\right),$$

with $k_1 = [3n_h/100]$, $k_2 = [3n_h/4]$, $j_1 = [n_h/2]$ and $j_2 = n_h$. Moreover, in order to select the two anchors points $k^\star$ and $j^\star$, we introduce Algorithm 4 with $k_{\texttt{U}} = [3n_h/100]$, $k_{\texttt{D}} = [3n_h/4]$, $j_{\texttt{L}} = [4n_h/100]$ and $j_{\texttt{R}} = n_h$, which is a 2-dimensional extension of [2, Algorithm 1]. The performance criteria considered now is similar to (34):

$$\text{RMedSE}\left(\hat{q}_{\hat{\phi}}^{\texttt{NN}_J}, t, \frac{1}{n_o}\right) = \underset{t \in \{1,\ldots,n_S\}}{\text{median}} \left(\frac{\hat{q}_{\hat{\phi}}^{\texttt{NN}_J}(1 - \frac{1}{n_o}; 1 - \frac{k^\star(t)}{n_o}; 1 - \frac{j^\star(t)}{n_o} \mid Y_t)}{X^{(n_o,n_o)}(Y_t)} - 1\right)^2.$$

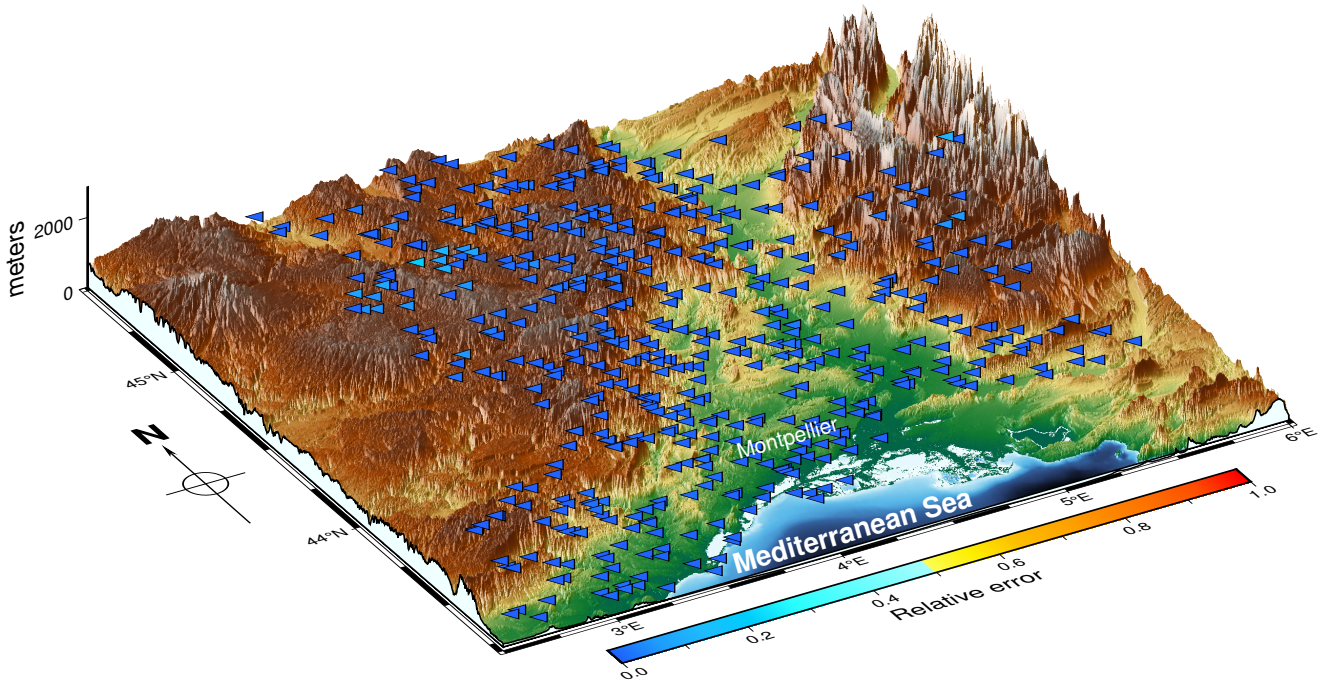
7.4 Results

The selected hyperparameters of both CENN and LDNN models are respectively $\{J = 5, \text{batch size} = 512\}$ and $\{J = 2, \text{batch size} = 1,024\}$. While the former has an heavy parametrization (2,050 parameters estimated from 2,541,924 data), the latter requires a large training dataset (10 parameters estimated from 82,188,876 data). In the following, we first study the conditional extrapolation performance in the tails of both neural networks at observed stations (local extrapolation). Then, we show an application to spatial interpolation through the CENN model (global extrapolation).

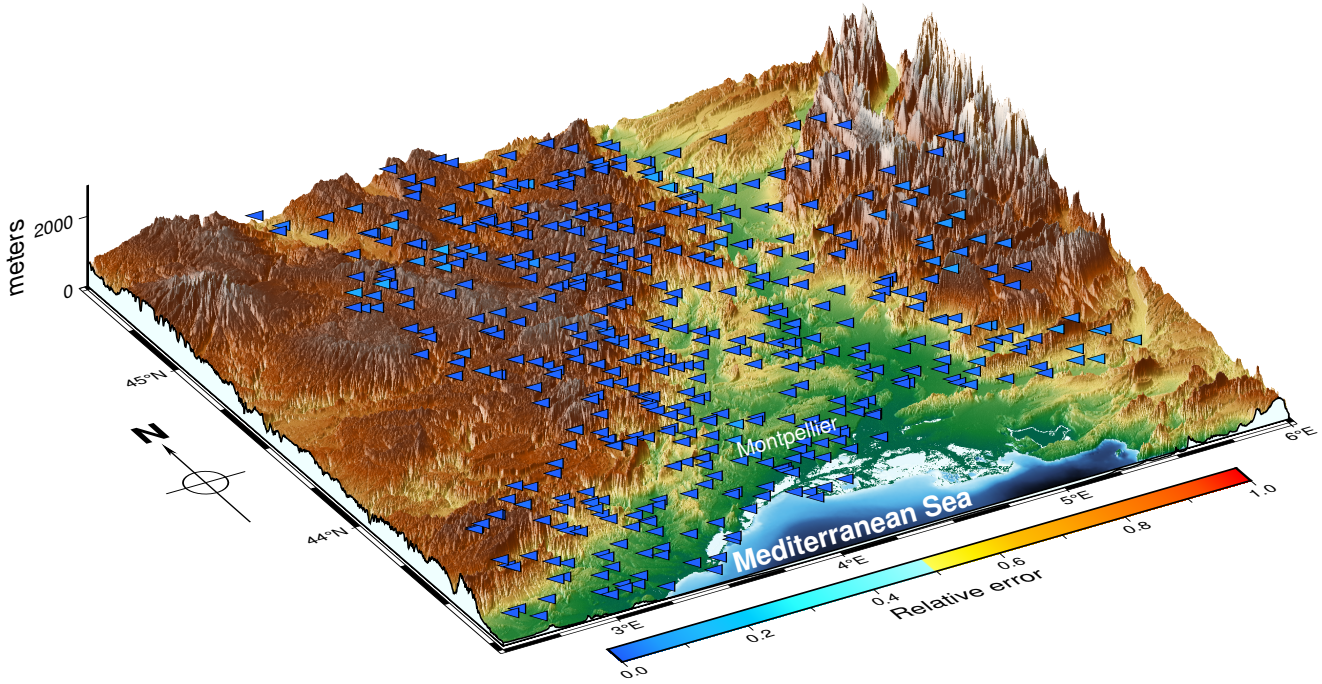
Local extrapolation Figure 5 displays the squared relative error

$$t \mapsto \left(\frac{\hat{q}_{\hat{\phi}}^{\texttt{NN}_J}(1 - 1/n_o; \cdot \mid Y_t)}{X^{(n_o,n_o)}(Y_t)} - 1\right)^2, \tag{37}$$

between two estimates of the conditional extreme quantile of order $1 - 1/n_o$ for each station $t = \{1, \ldots, n_S\}$ according to CENN and LDNN models. It appears that both models are very efficient for estimating conditional extreme quantiles. Additionally, the good results of the LDNN model confirm the assumption of a tail-index independent of the covariate.



(a) CENN (RMedSE=0.0047)



(b) LDNN (RMedSE=0.0022)

Figure 5: Estimation of the conditional extreme quantile at order $1 - \alpha_n = 1 - 1/n_o$ at each station. Relative error (37) with respect to both the CENN (top) and the LDNN (bottom) models.

Global extrapolation We extend the previous analysis to the estimation of conditional extreme quantiles at all pixels in Figure 1, thus including ungauged locations. While this can be achieved with the two models, we limit ourselves to presenting the results associated the CENN method, see Figure 6. The idea is to consider all $11,598,961$ pixels in the high resolution map of Figure 1, and provide an estimation of the conditional extreme quantile of order $1-\alpha_n = 1-1/n_o$, at locations not two far from a raingauge station. More specifically, the estimation is perfomed at points y of the covariate such that

$$D(y, Y_{t^\star}) \leq \kappa \sigma_{\texttt{MAD}}(t^\star)/\sqrt{n_K}, \tag{38}$$

with $t^\star = \arg\min_{t\in\{1,\ldots,n_S\}} D(y, Y_t)$, $\sigma_{\texttt{MAD}}(t^\star) = \texttt{MAD}(\{D_{t^\star}^{n_K-k+1,n_K},\ k \in \{1,\ldots,n_K-1\}\})$ and where $D_{t^\star}^{n_K-k+1,n_K}$ is the Mahalanobis distance between $Y_{t^\star}$ and its k-th nearest neighbor. In practice, we use $\kappa = 8$, leading to the extrapolation at $6,636,817$ points. Observe that the largest daily precipitations occur in the Cévennes mountains range, which is in line with both Figure 1 and the literature [50].

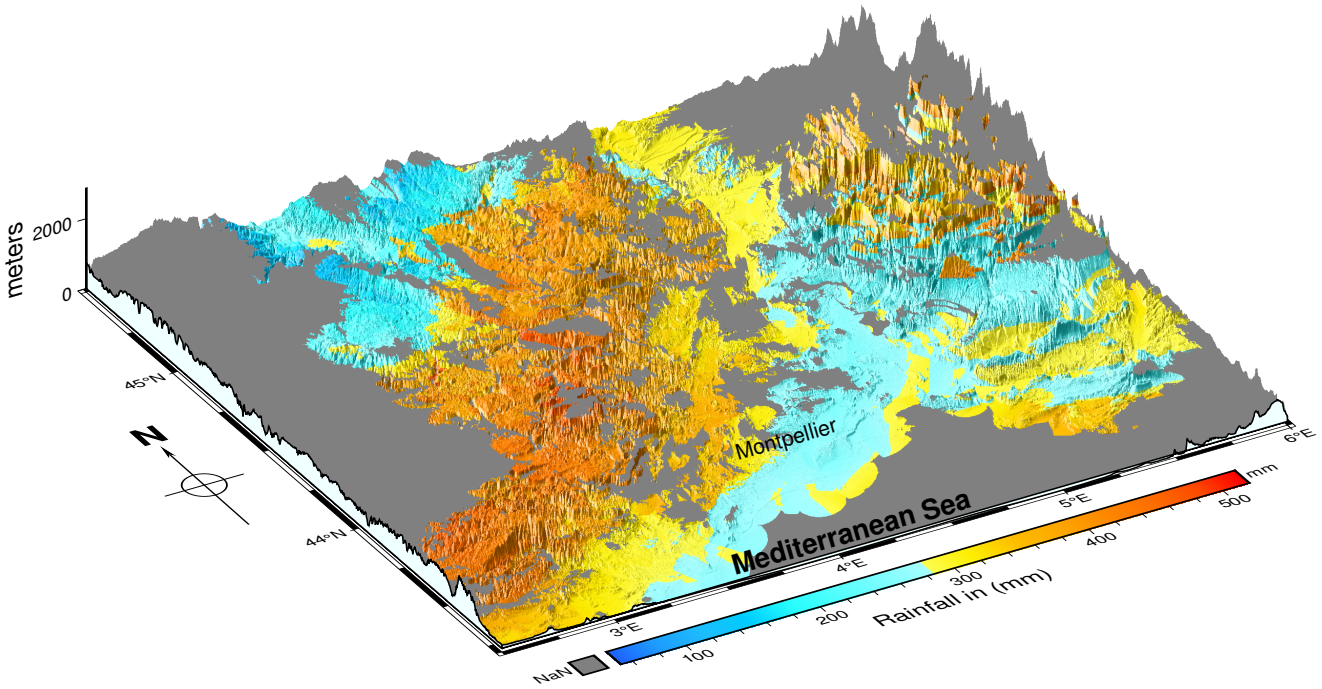


Figure 6: Estimation at a finer scale by the CENN quantile estimator at order $1-\alpha_n = 1-1/n_o$. The gray region is corresponds to Not a Number (NaN) results obtained when condition (38) is not satisfied.

8 Conclusion

We have introduced, up to our knowledge, the first neural network approach dedicated to extreme quantile estimation in both non-conditional and conditional settings. In particular, two neural networks of conditional extreme quantiles are proposed in order to tackle the cases where the tail-index depends or is independent of the covariate. From the theoretical point of view, the uniform convergence rates of the approximations underpinning the estimators are established within an extreme-value framework. From the practical point of view, our estimators have been tested both on simulated and real data; showing in the former case that the non-conditional NN estimator outperforms most of usual estimators in challenging heavy-tailed situations. In the rainfall data application, we have shown that both conditional NN estimators reproduce properly the tails at raingauge stations, and are moreover able to perform spatial interpolation to estimate extreme rainfalls at ungauged locations.

Our further work will consist in extending our NN approximations of extreme quantiles to other risk measures such as expected shortfall, or expectiles and then implementing the associated estimators. To complete the current theoretical analysis which ensures accurate approximation in the univariate case, our further work will be dedicated to investigate (in the non-conditional case) multivariate extreme quantile estimation basing on recent characterizations through optimal transport [42].

Acknowledgments This action benefited from the support of the Chair Stress Test, Risk Management and Financial Steering, led by the French Ecole polytechnique and its foundation and sponsored by BNP Paribas. This work has been partially supported by MIAI @ Grenoble Alpes (ANR-19-P3IA-0003).

A Algorithms

A.1 Model selection

We noticed that the proposed model selection techniques may be misleading during the first iterations when the NN is not well trained yet, leading to the same output for all anchor points k. Therefore, we decided to restrict the search of $m^\star$ after 10 iterations (step 3 of Algorithm 1, step 7 of Algorithm 2 and Algorithm 3).

Algorithm 1: Model selection (non-conditional case)

Input: *approximation order:* J,
extrapolation quantile level: $\alpha_n \in (0,1)$,
initial left point: $k_1 \in \{2,\dots,n-2\}$,
initial right point: $k_2 \in \{3,\dots,n-1\}$

Output: *selected parameters:* $\hat{\phi}_{m^\star}(\mathcal{A}^\star)$

for *all architecture* $\mathcal{A}$ *in Table 2* **do**
  for $m = 1 : M$ **do**
    Optimize (30) to get $\hat{\phi}_m(\mathcal{A})$
    $Z_m(\mathcal{A}) \leftarrow \texttt{MAD}\left(\left\{\hat{q}^{\texttt{NN}_J}_{\hat{\phi}_m(\mathcal{A})}\left(1-\alpha_n; 1-\frac{k}{n}\right),\ k \in \{k_1,\dots,k_2\}\right\}\right)$

$(m^\star, \mathcal{A}^\star) \leftarrow \underset{(m\in\{10,\dots,M\},\,\mathcal{A})}{\arg\min} Z_m(\mathcal{A})$

Algorithm 2: Model selection (CENN)

Input: *order condition:* J,
extrapolation quantile level: $\alpha_n \in [0,1]$,
initial left point: $k_1 \in \{2,\dots,n_h-1\}$,
initial right point: $k_2 \in \{3,\dots,n_h\}$

Output: *selected parameters:* $\hat{\phi}_{m^\star}(\mathcal{A}^\star)$

for *all architecture* $\mathcal{A}$ *in Table 2* **do**
  for $m = 1 : M$ **do**
    Optimize (33) to get $\hat{\phi}_m(\mathcal{A})$
    for $t = 1 : n_S$ **do**
      $Z_m^{(t)}(\mathcal{A}) \leftarrow \texttt{MAD}\left(\left\{\hat{q}^{\texttt{NN}_J}_{\hat{\phi}_m(\mathcal{A})}\left(1-\alpha_n; 1-\frac{k}{n_o} \mid Y_t\right),\ k \in \{k_1,\dots,k_2\}\right\}\right)$
    $\bar{Z}_m(\mathcal{A}) \leftarrow \underset{t\in\{1,\dots,n_S\}}{\text{median}}\left\{Z_m^{(t)}(\mathcal{A})\right\}$

$(m^\star, \mathcal{A}^\star) \leftarrow \underset{(m\in\{10,\dots,M\},\,\mathcal{A})}{\arg\min} \bar{Z}_m(\mathcal{A})$

Algorithm 3: Model selection (LDNN)

Input: *order condition:* J,
extrapolation quantile level: $\alpha_n \in [0,1]$,
initial left point: $k_1 \in \{2,\dots,n_h-3\}$,
initial right point: $k_2 \in \{3,\dots,n_h-2\}$
initial upper point: $j_1 \in \{4,\dots,n_h-1\}$,
initial down point: $j_2 \in \{5,\dots,n_h\}$

Output: *selected parameters:* $\hat{\phi}_{m^\star}(\mathcal{A}^\star)$

for *all architecture* $\mathcal{A}$ *in Table 2* **do**
  for $m = 1 : M$ **do**
    Optimize (35) to get $\hat{\phi}_m(\mathcal{A}^\star)$
    for $t = 1 : n_S$ **do**
      $$Z_m^{(t)}(\mathcal{A}) \leftarrow \texttt{MAD}\left(\left\{\hat{q}^{\texttt{NN}_J}_{\hat{\phi}_m(\mathcal{A})}\left(1-\alpha_n; 1-\tfrac{k}{n_o}; 1-\tfrac{j}{n_o} \mid Y_t\right), \; k \in \{k_1,\dots,k_2\}, j' \in \{j_1,\dots,j_2\}, k < j\right\}\right)$$
    $$\bar{Z}_m(\mathcal{A}) \leftarrow \underset{t\in\{1,\dots,n_S\}}{\text{median}} \left\{Z_m^{(t)}(\mathcal{A})\right\}$$
$$(m^\star, \mathcal{A}^\star) \leftarrow \underset{(m\in\{10,\dots,M\},\, \mathcal{A})}{\arg\min} \bar{Z}_m(\mathcal{A})$$

A.2 Selection of the sample fractions

Algorithm 4: Selection of k and j using random forests 2D

Input: *triangular matrix:* $\mathcal{Z} = \{Z_{k,j}\}_{(k,j)\in\{2,\dots,j-1\}\times\{3,\dots,n-1\}}$,
number of trees: $T \in \mathbb{N} \setminus \{0\}$,
initial top point: $k_{\mathrm{T}}^{(0)} \in \{2,\dots,n-3\}$,
initial down point: $k_{\mathrm{D}}^{(0)} \in \{3,\dots,n-2\}$,
initial left point: $j_{\mathrm{L}}^{(0)} \in \{4,\dots,n-1\}$,
initial right point: $j_{\mathrm{R}}^{(0)} \in \{5,\dots,n\}$

Output: *selected points:* $k^\star$, $j^\star$

for $t = 1 : T$ **do**

$j_{\mathrm{L}}^{(t)} \sim \texttt{randint}(j_{\mathrm{L}}^{(0)}, j_{\mathrm{R}}^{(0)} - 1)$

$j_{\mathrm{R}}^{(t)} \sim \texttt{randint}(j_{\mathrm{L}}^{(t)} + 1, j_{\mathrm{R}}^{(0)})$

$k_{\mathrm{D}}^{(t)} \sim \texttt{randint}(k_{\mathrm{T}}^{(0)} + 1, k_{\mathrm{D}}^{(0)} \vee j_{\mathrm{L}}^{(t)})$

$k_{\mathrm{T}}^{(t)} \sim \texttt{randint}(k_{\mathrm{T}}^{(0)}, k_{\mathrm{D}}^{(t)} - 1)$

$k^{(t)}, j^{(t)} \leftarrow \mathrm{Tree2D}(\mathcal{Z}, j_{\mathrm{L}}^{(t)}, j_{\mathrm{R}}^{(t)}, k_{\mathrm{T}}^{(t)}, k_{\mathrm{D}}^{(t)})$

$k^\star \leftarrow \mathrm{median}(k^{(1)},\dots,k^{(T)}), \quad j^\star \leftarrow \mathrm{median}(j^{(1)},\dots,j^{(T)})$

Algorithm 5: Tree2D

Input: *triangular matrix:* $\mathcal{Z} = \{Z_{k,j}\}_{(k,j)\in\{2,\dots,j-1\}\times\{3,\dots,n\}}$,
initial top point: $k_{\mathrm{T}}^{(0)} \in \{2,\dots,n-3\}$,
initial down point: $k_{\mathrm{D}}^{(0)} \in \{3,\dots,n-2\}$,
initial left point: $j_{\mathrm{L}}^{(0)} \in \{4,\dots,n-1\}$,
initial right point: $j_{\mathrm{R}}^{(0)} \in \{5,\dots,n\}$

Output: *selected points:* k, j

$k_{\mathrm{M}} \leftarrow \left[\frac{k_{\mathrm{T}}+k_{\mathrm{D}}}{2}\right], \quad j_{\mathrm{M}} \leftarrow \left[\frac{j_{\mathrm{L}}+j_{\mathrm{R}}}{2}\right]$

while $(k_{\mathrm{M}} - k_{\mathrm{T}}) > 1$ *or* $(j_{\mathrm{M}} - j_{\mathrm{L}}) > 1$ **do**

$V_{\mathrm{TL}} \leftarrow \mathrm{EmpiricalVariance2D}(\mathcal{Z}, k_{\mathrm{M}}, k_{\mathrm{T}}, j_{\mathrm{M}}, j_{\mathrm{L}})$
$V_{\mathrm{TR}} \leftarrow \mathrm{EmpiricalVariance2D}(\mathcal{Z}, k_{\mathrm{M}}, k_{\mathrm{T}}, j_{\mathrm{R}}, j_{\mathrm{M}})$
$V_{\mathrm{DL}} \leftarrow \mathrm{EmpiricalVariance2D}(\mathcal{Z}, k_{\mathrm{D}}, k_{\mathrm{M}}, j_{\mathrm{M}}, j_{\mathrm{L}})$
$V_{\mathrm{DR}} \leftarrow \mathrm{EmpiricalVariance2D}(\mathcal{Z}, k_{\mathrm{D}}, k_{\mathrm{M}}, j_{\mathrm{R}}, j_{\mathrm{M}})$
if $\min(V_{\mathrm{TL}}, V_{\mathrm{TR}}, V_{\mathrm{DL}}, V_{\mathrm{DR}}) = V_{\mathrm{TL}}$ **then**
  $k_{\mathrm{D}} \leftarrow k_M, \quad j_R \leftarrow j_{\mathrm{M}}$
else if $\min(V_{\mathrm{TL}}, V_{\mathrm{TR}}, V_{\mathrm{DL}}, V_{\mathrm{DR}}) = V_{\mathrm{TR}}$ **then**
  $k_{\mathrm{D}} \leftarrow k_{\mathrm{M}}, \quad j_{\mathrm{L}} \leftarrow j_{\mathrm{M}}$
else if $\min(V_{\mathrm{TL}}, V_{\mathrm{TR}}, V_{\mathrm{DL}}, V_{\mathrm{DR}}) = V_{\mathrm{DL}}$ **then**
  $k_{\mathrm{T}} \leftarrow k_{\mathrm{M}}, \quad j_{\mathrm{R}} \leftarrow j_{\mathrm{M}}$
else
  $k_{\mathrm{U}} \leftarrow k_{\mathrm{M}}, \quad j_{\mathrm{L}} \leftarrow j_{\mathrm{M}}$
$k_{\mathrm{M}} \leftarrow \left[\frac{k_{\mathrm{T}}+k_{\mathrm{D}}}{2}\right], \quad j_{\mathrm{M}} \leftarrow \left[\frac{j_{\mathrm{L}}+j_{\mathrm{R}}}{2}\right]$

Algorithm 6: EmpiricalVariance2D

Input: *triangular matrix:* $\mathcal{Z} = \{Z_{k,j}\}_{(k,j)\in\{2,\dots,j-1\}\times\{3,\dots,n\}}$,
initial points: $(k_a, k_b, j_a, j_b) \in \mathbb{N}^4$,

Output: *empirical variance:* $\hat{\sigma}^2$

Compute

$$\bar{Z} \leftarrow \frac{1}{(k_a - k_b) + (j_a - j_b) + 2} \sum_{k=k_b}^{k_a} \sum_{j=j_b}^{j_a} Z_{k,j}.$$

Compute

$$\hat{\sigma}^2 \leftarrow \frac{1}{(k_a - k_b) + (j_a - j_b) + 2} \sum_{k=k_b}^{k_a} \sum_{j=j_b}^{j_a} \left(Z_{k,j} - \bar{Z}\right)^2,$$

B Proofs

Lemma 6. *Let $K_t(s) := (s^t - 1)/t$ be defined for all $s \geq 1$ and $t < 0$. Then, for all $z \geq 1$ and $\rho_j < 0$ for $j \geq 2$, one can equivalently express* (16) *as:*

$$R_j(z) = \sum_{\ell=2}^{j} a_{\ell,j} K_{\bar{\rho}_\ell}(z),$$

where $\bar{\rho}_\ell = \rho_2 + \cdots + \rho_\ell$, and for some coefficients $a_{\ell,j} \in \mathbb{R}$.

Proof. For all $s \geq 1$, $p < 0$ and $q < 0$, one has

$$\int_1^s z^p K_q(z)\,\mathrm{d}z = \frac{1}{q}\left(K_{p+q+1}(s) - K_{p+1}(s)\right). \tag{39}$$

Replacing in (16) yields for all $j \geq 2$ and $y \geq 1$,

$$R_j(z) = \int_1^z z_2^{\rho_2 - 1} \int_1^{z_2} z_3^{\rho_3 - 1} \cdots \int_1^{z_{j-2}} z_{j-1}^{\rho_{j-1} - 1} K_{\rho_j}(z_{j-1})\,\mathrm{d}z_{j-1} \ldots \mathrm{d}z_3\,\mathrm{d}z_2,$$

with

$$\int_1^{z_{j-1}} z_j^{\rho_j - 1}\,\mathrm{d}z_j = \frac{z_{j-1}^{\rho_j} - 1}{\rho_j} = K_{\rho_j}(z_{j-1}).$$

Assume $\rho_j < 0$ for all $j \geq 2$, then from (39), one can show by recursion that,

$$\begin{aligned}
R_j(z) &= \frac{1}{\rho_j}\left(\ldots\left(\frac{1}{\bar{\rho}_j - \bar{\rho}_3}\left(\frac{K_{\bar{\rho}_j}(z) - R_2(z)}{\bar{\rho}_j - \bar{\rho}_2} - R_3(z)\right) - \ldots\right) - R_{j-1}(z)\right), \text{ for } j \geq 4,\\
R_2(z) &= K_{\bar{\rho}_2}(z) = \frac{z^{\rho_2} - 1}{\rho_2},\\
R_3(z) &= \frac{1}{\rho_3}\Big(K_{\bar{\rho}_3}(z) - K_{\bar{\rho}_2}(z)\Big),
\end{aligned}$$

which concludes the proof. □

Proof of Proposition 1. Combining the J-th order condition (15) and [22, Theorem 2.1], we get that, for every $\varepsilon > 0$, there exists $t_0 > 0$ such that, for all $t \geq t_0$ and $tz \geq t_0$,

$$\log U(tz) - \log U(t) = \gamma \log z + \sum_{j=2}^{J} \prod_{\ell=2}^{j} A_\ell(t) R_j(z) + \Delta(z,t) \prod_{j=2}^{J} A_j(t),$$

with

$$\begin{aligned}
|\Delta(z,t)| &:= \left|\frac{1}{A_J(t)}\left(\ldots\left(\frac{1}{A_3(t)}\left(\frac{\log U(tz) - \log U(t) - \gamma \log z}{A_2(t)} - R_2(z)\right) - R_3(z)\right) - \ldots\right) - R_J(z)\right|\\
&\leq \varepsilon z^{\bar{\rho}_J + \varepsilon}.
\end{aligned} \tag{40}$$

Thus, (1) yields

$$\log\left(\frac{L(tz)}{L(t)}\right) = \sum_{j=2}^{J} \prod_{\ell=2}^{j} A_\ell(t) R_j(z) + \Delta(z,t) \prod_{j=2}^{J} A_j(t),$$

or equivalently, considering $t = \exp(x_2)$, $z = \exp(x_1)$ and taking account of (4):

$$\begin{aligned}\varphi(x_1, x_2) &= \log\left(\frac{L(\exp(x_1 + x_2))}{L(\exp(x_2))}\right) \\ &= \sum_{j=2}^{J}\prod_{\ell=2}^{j} A_\ell(\exp(x_2))R_j(\exp(x_1)) + \Delta(\exp(x_1), \exp(x_2))\prod_{j=2}^{J} A_j(\exp(x_2)). \qquad (41)\end{aligned}$$

Using assumption (17) and replacing in (41), it follows:

$$\varphi(x_1, x_2) = \sum_{j=2}^{J}\prod_{\ell=2}^{j} c_\ell \exp(\rho_\ell x_2)R_j(\exp(x_1)) + \Delta(\exp(x_1), \exp(x_2))\prod_{j=2}^{J} A_j(\exp(x_2)),$$

and thus, letting $\bar{\rho}_j = \rho_2 + \cdots + \rho_j$ and $\bar{c}_j = c_2 \times \ldots \times c_j$, one has

$$\varphi(x_1, x_2) = \sum_{j=2}^{J} \bar{c}_j \exp(\bar{\rho}_j x_2)R_j(\exp(x_1)) + \Delta(\exp(x_1), \exp(x_2))\prod_{j=2}^{J} A_j(\exp(x_2)).$$

Introduce

$$\tilde{\varphi}_\theta^{\mathtt{NN}_J}(x_1, x_2) = \sum_{j=2}^{J} \bar{c}_j \exp(\bar{\rho}_j x_2)R_j(\exp(x_1)),$$

so that

$$\varphi(x_1, x_2) = \tilde{\varphi}_\theta^{\mathtt{NN}_J}(x_1, x_2) + \Delta(\exp(x_1), \exp(x_2))\prod_{j=2}^{J} A_j(\exp(x_2)).$$

Taking account of Lemma 6, we have

$$\begin{aligned}\tilde{\varphi}_\theta^{\mathtt{NN}_J}(x_1, x_2) &= \sum_{j=2}^{J} \bar{c}_j \exp(\bar{\rho}_j x_2)\sum_{\ell=2}^{j} a_{\ell,j} K_{\bar{\rho}_\ell}(\exp(x_1)), \\ &= \sum_{j=2}^{J}\sum_{\ell=2}^{j} \frac{\bar{c}_j a_{\ell,j}}{\bar{\rho}_\ell}\left(\exp(\bar{\rho}_\ell x_1 + \bar{\rho}_j x_2) - \exp(\bar{\rho}_j x_2)\right).\end{aligned}$$

Re-indexing, we get

$$\tilde{\varphi}_\theta^{\mathtt{NN}_J}(x_1, x_2) = \sum_{i=1}^{J(J-1)/2} w_i^{(1)}\left(\exp\left(w_i^{(2)} x_1 + w_i^{(3)} x_2\right) - \exp(w_i^{(4)} x_2)\right), \qquad (42)$$

with, $w_i^{(1)} \in \mathbb{R}$, $w_i^{(2)} < 0$, $w_i^{(3)} < 0$, $w_i^{(4)} < 0$ for all $i = 1, \ldots, J(J-1)/2$. Replacing (13) in (42) yields the expression (18) of $\tilde{\varphi}_\theta^{\mathtt{NN}_J}$ in terms of eLU functions. The result is proved. □

Proof of Theorem 2. Let (ε_n) be a sequence in $(0, -\bar{\rho}_J)$. We have, in view of the triangle inequality:

$$\begin{aligned}\inf_{\tilde{\phi}\in\Phi}\left|\log q(1-\alpha_n) - \log \tilde{q}_{\tilde{\phi}}^{\mathtt{NN}_J}(1-\alpha_n; 1-\delta_n)\right| &= \inf_{\tilde{\phi}\in\Phi}\left|f(\log(\delta_n/\alpha_n), \log(1/\delta_n)) - \tilde{f}_{\tilde{\phi}}^{\mathtt{NN}_J}(\log(\delta_n/\alpha_n), \log(1/\delta_n))\right| \\ &\leq \inf_{\tilde{w}_0\in\mathbb{R}_+}|\gamma - \tilde{w}_0|\log(\delta_n/\alpha_n) \\ &+ \inf_{\tilde{\theta}\in\Theta}\left|\varphi(\log(\delta_n/\alpha_n), \log(1/\delta_n)) - \tilde{\varphi}_{\tilde{\theta}}^{\mathtt{NN}_J}(\log(\delta_n/\alpha_n), \log(1/\delta_n))\right| \\ &\leq \left|\varphi(\log(\delta_n/\alpha_n), \log(1/\delta_n)) - \tilde{\varphi}_\theta^{\mathtt{NN}_J}(\log(\delta_n/\alpha_n), \log(1/\delta_n))\right| \\ &\leq |\Delta(\delta_n/\alpha_n, 1/\delta_n)|\left|\prod_{j=2}^{J} A_j(1/\delta_n)\right|,\end{aligned}$$

from Proposition 1. Moreover, since $\delta_n/\alpha_n \to \infty$ and $1/\delta_n \to \infty$, for n large enough,

$$|\Delta(\delta_n/\alpha_n, 1/\delta_n)| \leq \varepsilon_n \left(\frac{\delta_n}{\alpha_n}\right)^{\bar{\rho}_J+\varepsilon_n}, \tag{43}$$

while, under assumption (17),

$$\left|\prod_{j=2}^{J} A_j(1/\delta_n)\right| = |\bar{c}_J|\, \delta_n^{-\bar{\rho}_J},$$

where $\bar{c}_J = c_2 \times \cdots \times c_J$. As a conclusion,

$$\alpha_n^{\bar{\rho}_J} \inf_{\tilde{\phi}\in\Phi} \left|\log q(1-\alpha_n) - \log \tilde{q}_{\tilde{\phi}}^{\mathtt{NN}_J}(1-\alpha_n; 1-\delta_n)\right| \leq \bar{c}_J \varepsilon_n \left(\frac{\delta_n}{\alpha_n}\right)^{\varepsilon_n},$$

and letting $\varepsilon_n = \exp(-\mathcal{W}(\log(\delta_n/\alpha_n))) \to 0$ as $n \to \infty$, where $\mathcal{W}$ is the Lambert-W function [11], yields $\log(1/\varepsilon_n)/\varepsilon_n = \log(\delta_n/\alpha_n)$, the result is proved. □

Proof of Theorem 3. Introducing

$$\begin{aligned}
\varphi_n(y) &= \varphi\left(\log(\delta_n/\alpha_n), \log(1/\delta_n) \mid y\right), \\
\tilde{\varphi}_{n,\tilde{\theta}}^{\mathtt{NN}_J}(y) &= \tilde{\varphi}_{\tilde{\theta}}^{\mathtt{NN}_J}\left(\log(\delta_n/\alpha_n), \log(1/\delta_n) \mid y\right), \\
V_n(y) &= \Delta(\delta_n/\alpha_n, 1/\delta_n \mid y) \prod_{j=2}^{J} A_j(1/\delta_n \mid y), \\
H_{n,i}(y) &= \sigma^{\mathrm{e}}\left(w_i^{(2)}(y)\log(\delta_n/\alpha_n) + w_i^{(3)}(y)\log(1/\delta_n)\right) - \sigma^{\mathrm{e}}\left(w_i^{(4)}(y)\log(1/\delta_n)\right), \qquad (44) \\
\tilde{H}_{n,\tilde{\theta}_i^{(2-4)}}^{\mathtt{NN}}(y) &= \sigma^{\mathrm{e}}\left(\tilde{w}_{\tilde{\theta}_i^{(2)}}^{\mathtt{NN}}(y)\log(\delta_n/\alpha_n) + \tilde{w}_{\tilde{\theta}_i^{(3)}}^{\mathtt{NN}}(y)\log(1/\delta_n)\right) - \sigma^{\mathrm{e}}\left(\tilde{w}_{\tilde{\theta}_i^{(4)}}^{\mathtt{NN}}(y)\log(1/\delta_n)\right), \qquad (45)
\end{aligned}$$

for all $i = 1, \ldots, J(J-1)/2$ and where $\Delta(\delta_n/\alpha_n, 1/\delta_n|y)$ is defined similarly to (40) in the unconditional case. Remark that all functions $w^{(0)}(\cdot) = \gamma(\cdot)$ and $w_i^{(j)}(\cdot)$, $i = 1, \ldots, J(J-1)/2$, $j = 1, \ldots, 4$ are assumed to be continuous on Π w.r.t. the covariate y. It is known from [63, Theorem 2] that there exists a deep ReLU neural network that can uniformly approximate any of these continuous functions on a compact set with an error

$$\epsilon(p_n^{(j)}) := \max_{i=1,\ldots,J(J-1)/2} \inf_{\tilde{\theta}_i^{(j)}} \sup_{y\in\Pi} \left|w_i^{(j)}(y) - \tilde{w}_{\tilde{\theta}_i^{(j)}}^{\mathtt{NN}}(y)\right| = O((p_n^{(j)})^{-2}) \tag{46}$$

requiring $2d_y + 10$ neurons in each of the $p_n^{(j)}$ hidden layers, $j \in \{0, \ldots, 4\}$. This error is optimal with respect to the depth [63, Theorem 1(a)]. We have, in view of the triangle inequality:

$$\begin{aligned}
\inf_{\tilde{\phi}\in\Phi} \sup_{y\in\Pi} \left|\log q(1-\alpha_n \mid y) - \log \tilde{q}_{\tilde{\phi}}^{\mathtt{NN}_J}(1-\alpha_n; 1-\delta_n \mid y)\right| &\leq \inf_{\tilde{\theta}^{(0)}} \sup_{y\in\Pi} \left|\gamma(y) - \tilde{w}_{\tilde{\theta}^{(0)}}^{\mathtt{NN}}(y)\right| \log(\delta_n/\alpha_n) \\
&\quad + \inf_{\tilde{\theta}} \sup_{y\in\Pi} \left|\varphi_n(y) - \tilde{\varphi}_{n,\tilde{\theta}}^{\mathtt{NN}_J}(y)\right|.
\end{aligned}$$

The first term can easily be controlled thanks to (46):

$$\inf_{\tilde{\theta}^{(0)}} \sup_{y\in\Pi} \left|\gamma(y) - \tilde{w}_{\tilde{\theta}^{(0)}}^{\mathtt{NN}}(y)\right| \log(\delta_n/\alpha_n) \leq \epsilon(p_n^{(0)}) \log(\delta_n/\alpha_n).$$

Next, rearranging and applying the triangle inequality entail

$$\begin{aligned}
\inf_{\tilde{\theta}} \sup_{y\in\Pi} \left|\varphi_n(y) - \tilde{\varphi}^{\mathtt{NN}_J}_{n,\tilde{\theta}}(y)\right| &= \inf_{\tilde{\theta}} \sup_{y\in\Pi} \left| \sum_{i=1}^{J(J-1)/2} w_i^{(1)}(y) H_{n,i}(y) - \sum_{i=1}^{J(J-1)/2} \tilde{w}^{\mathtt{NN}}_{\tilde{\theta}_i^{(1)}}(y) \tilde{H}^{\mathtt{NN}}_{n,\tilde{\theta}_i^{(2-4)}}(y) + V_n(y) \right| \\
&= \inf_{\tilde{\theta}} \sup_{y\in\Pi} \Bigg| \sum_{i=1}^{J(J-1)/2} \left(w_i^{(1)}(y) - \tilde{w}^{\mathtt{NN}}_{\tilde{\theta}_i^{(1)}}\right) H_{n,i}(y) \\
&\qquad + \sum_{i=1}^{J(J-1)/2} \left(H_{n,i}(y) - \tilde{H}^{\mathtt{NN}}_{n,\tilde{\theta}_i^{(2-4)}}(y)\right) \tilde{w}^{\mathtt{NN}}_{\tilde{\theta}_i^{(1)}}(y) + V_n(y) \Bigg| \\
&\leq \sum_{i=1}^{J(J-1)/2} \inf_{\tilde{\theta}_i^{(1)}} \sup_{y\in\Pi} \left|w_i^{(1)}(y) - \tilde{w}^{\mathtt{NN}}_{\tilde{\theta}_i^{(1)}}\right| \sup_{y\in\Pi} |H_{n,i}(y)| \qquad (47) \\
&+ \sum_{i=1}^{J(J-1)/2} \inf_{\tilde{\theta}_i^{(2-4)}} \sup_{y\in\Pi} \left|H_{n,i}(y) - \tilde{H}^{\mathtt{NN}}_{n,\tilde{\theta}_i^{(2-4)}}(y)\right| \inf_{\tilde{\theta}^{(1)}} \sup_{y\in\Pi} \left|\tilde{w}^{\mathtt{NN}}_{\tilde{\theta}_i^{(1)}}(y)\right| \qquad (48) \\
&+ \sup_{y\in\Pi} |V_n(y)| \,. \qquad (49)
\end{aligned}$$

The three terms (47), (48) and (49) are considered separately. First, note that

$$\sup_{y\in\Pi} |H_{n,i}(y)| \leq 1, \qquad (50)$$

since $w_i^{(j)}(y) \leq 0$ for all $i = 1, \ldots, J(J-1)/2$, $j = 1, 2, 3$ and $y \in \Pi$ in (44). Combining (46) and (50) yields

$$(47) \leq \epsilon(p_n^{(1)}) \frac{J(J-1)}{2}.$$

Next, focusing on (48), and taking account of $\left(\tilde{w}^{\mathtt{NN}}_{\tilde{\theta}_i^{(2)}}(\cdot), \tilde{w}^{\mathtt{NN}}_{\tilde{\theta}_i^{(3)}}(\cdot), \tilde{w}^{\mathtt{NN}}_{\tilde{\theta}_i^{(4)}}(\cdot)\right) \in \mathbb{R}^3_-$ by construction, one has for all $i = 1, \ldots, J(J-1)/2$,

$$\begin{aligned}
&\inf_{\tilde{\theta}_i^{(2-4)}} \sup_{y\in\Pi} \left|H_{n,i}(y) - \tilde{H}^{\mathtt{NN}}_{n,\tilde{\theta}_i^{(2-4)}}(y)\right| \\
&\leq \inf_{\tilde{\theta}_i^{(2-3)}} \sup_{y\in\Pi} \left|\exp\left(w_i^{(2)}(y)\log(\delta_n/\alpha_n) + w_i^{(3)}(y)\log(1/\delta_n)\right) - \exp\left(\tilde{w}^{\mathtt{NN}}_{\tilde{\theta}_i^{(2)}}(y)\log(\delta_n/\alpha_n) + \tilde{w}^{\mathtt{NN}}_{\tilde{\theta}_i^{(3)}}(y)\log(1/\delta_n)\right)\right| \\
&+ \inf_{\tilde{\theta}_i^{(4)}} \sup_{y\in\Pi} \left|\exp\left(w_i^{(4)}(y)\log(1/\delta_n)\right) - \exp\left(\tilde{w}^{\mathtt{NN}}_{\tilde{\theta}_i^{(4)}}(y)\log(1/\delta_n)\right)\right| \\
&\leq \inf_{\tilde{\theta}_i^{(2-3)}} \sup_{y\in\Pi} \left|1 - \exp\left(\log(\delta_n/\alpha_n)\left(\tilde{w}^{\mathtt{NN}}_{\tilde{\theta}_i^{(2)}}(y) - w_i^{(2)}(y)\right) + \log(1/\delta_n)\left(\tilde{w}^{\mathtt{NN}}_{\tilde{\theta}_i^{(3)}}(y) - w_i^{(3)}(y)\right)\right)\right| \qquad (51) \\
&+ \inf_{\tilde{\theta}_i^{(4)}} \sup_{y\in\Pi} \left|1 - \exp\left(\log(1/\delta_n)\left(\tilde{w}^{\mathtt{NN}}_{\tilde{\theta}_i^{(4)}}(y) - w_i^{(4)}(y)\right)\right)\right|. \qquad (52)
\end{aligned}$$

Let us first consider

$$h_{n,\tilde{\theta}_i^{(2-3)}}(y) = \log(\delta_n/\alpha_n)\left(\tilde{w}^{\mathtt{NN}}_{\tilde{\theta}_i^{(2)}}(y) - w_i^{(2)}(y)\right) + \log(1/\delta_n)\left(\tilde{w}^{\mathtt{NN}}_{\tilde{\theta}_i^{(3)}}(y) - w_i^{(3)}(y)\right),$$

and remark that $\log(\delta_n/\alpha_n)\epsilon(p_n^{(2)}) \to 0$ and $\log(1/\delta_n)\epsilon(p_n^{(3)}) \to 0$ as $n \to \infty$ imply

$$\sup_{y\in\Pi} \left|h_{n,\tilde{\theta}_i^{(2-3)}}(y)\right| \leq \log(\delta_n/\alpha_n)\epsilon(p_n^{(2)}) + \log(1/\delta_n)\epsilon(p_n^{(3)}) \leq \log 2$$

for n large enough. Since $|1-\exp(u)| \leq 2\,|u|$ for any $|u| \leq \log 2$, it follows

$$(51) = \inf_{\tilde{\theta}_i^{(2-3)}} \sup_{y\in\Pi} \left|1 - \exp\left(h_{n,\tilde{\theta}_i^{(2-3)}}(y)\right)\right| \leq 2\left(\log(\delta_n/\alpha_n)\epsilon(p_n^{(2)}) + \log(1/\delta_n)\epsilon(p_n^{(3)})\right).$$

Second, applying the same method to control (52) yields

$$\inf_{\tilde{\theta}_i^{(2-4)}} \sup_{y\in\Pi} \left|H_{n,i}(y) - \tilde{H}^{\texttt{NN}}_{n,\tilde{\theta}_i^{(2-4)}}(y)\right| \leq 2\left(\log(\delta_n/\alpha_n)\epsilon(p_n^{(2)}) + \log(1/\delta_n)\left(\epsilon(p_n^{(3)}) + \epsilon(p_n^{(4)})\right)\right),$$

since $\log(1/\delta_n)\epsilon(p_n^{(4)}) \to 0$ as $n \to \infty$. Moreover, in view of (46) we have

$$\inf_{\tilde{\theta}_i^{(1)}} \sup_{y\in\Pi} \left|\tilde{w}^{\texttt{NN}}_{\tilde{\theta}_i^{(1)}}(y)\right| \leq \sup_{y\in\Pi} \left|w_i^{(1)}(y)\right| + \epsilon(p^{(1)}) \leq C_1 + \epsilon(p_n^{(1)}),$$

where $C_1 \geq 0$ is a constant since $w_i^{(1)}(\cdot)$ is by assumption a continuous function on a compact set. Together with (51) and (52), this entails

$$(48) \leq J(J-1)\left(\log(\delta_n/\alpha_n)\epsilon(p_n^{(2)}) + \log(1/\delta_n)\left(\epsilon(p_n^{(3)}) + \epsilon(p_n^{(4)})\right)\right)\left(C_1 + \epsilon(p_n^{(1)})\right).$$

Finally, under assumption (24), the last term (49) can be rewritten using

$$V_n(y) = \Delta\left(\delta_n/\alpha_n, 1/\delta_n \mid y\right)\delta_n^{-\bar{\rho}_J(y)}\bar{c}_J(y),$$

with $\bar{c}_J(y) = \prod_{j=2}^J c_j(y)$ and $\bar{\rho}_J(y) = \sum_{j=2}^J \rho_j(y)$. Taking advantage of (43) yields

$$\sup_{y\in\Pi} \left|\Delta\left(\delta_n/\alpha_n, 1/\delta_n \mid y\right)\right| \left(\delta_n/\alpha_n\right)^{-\bar{\rho}_J(y)} \leq \varepsilon_n \left(\delta_n/\alpha_n\right)^{\varepsilon_n}, \tag{53}$$

where $\varepsilon_n = \exp(-\mathcal{W}(\log(\delta_n/\alpha_n)))$ is defined in Proposition 1. Therefore,

$$\begin{aligned}
\sup_{y\in\Pi} |V_n(y)| &\leq \sup_{y\in\Pi} \left|\Delta\left(\delta_n/\alpha_n, 1/\delta_n \mid y\right)\right| \delta_n^{-\bar{\rho}_J(y)} \sup_{y\in\Pi} \bar{c}_J(y) \\
&\leq \sup_{y\in\Pi} \left|\Delta\left(\delta_n/\alpha_n, 1/\delta_n \mid y\right)\right| \left(\frac{\delta_n}{\alpha_n}\right)^{-\bar{\rho}_J(y)} \sup_{y\in\Pi} \alpha_n^{-\bar{\rho}_J(y)} \sup_{y\in\Pi} \bar{c}_J(y),
\end{aligned}$$

and combining with (53), it yields

$$\sup_{y\in\Pi} |V_n(y)| \leq c_{\sup}\, \varepsilon_n \left(\delta_n/\alpha_n\right)^{\varepsilon_n} \alpha_n^{-\bar{\rho}_{\sup}},$$

where $\bar{c}_{\sup} := \sup_{y\in\Pi} \bar{c}_J(y)$ and $\bar{\rho}_{\sup} := \sup_{y\in\Pi} \bar{\rho}_J(y)$. All in all, one has

$$\begin{aligned}
&\inf_{\tilde{\theta}} \sup_{y\in\Pi} \left|\varphi(y) - \tilde{\varphi}^{\texttt{NN}_J}_{\tilde{\theta}}(y)\right| \\
&\quad = \mathcal{O}\left(\epsilon(p_n^{(1)})\right) + \mathcal{O}\left(\log(\delta_n/\alpha_n)\epsilon\left(p_n^{(2)}\right)\right) + \mathcal{O}\left(\log(1/\delta_n)\left(\epsilon\left(p_n^{(3)}\right) + \epsilon\left(p_n^{(4)}\right)\right)\right) + \mathcal{O}\left(\alpha_n^{-\bar{\rho}_{\sup}}\right),
\end{aligned}$$

leading to

$$\begin{aligned}
&\alpha_n^{\bar{\rho}_{\sup}} \inf_{\tilde{\phi}\in\Phi} \sup_{y\in\Pi} \left|\log q(1-\alpha_n \mid y) - \log \tilde{q}^{\texttt{NN}_J}_{\tilde{\phi}}(1-\alpha_n; 1-\delta_n \mid y)\right| \\
&\quad = \mathcal{O}\left(\alpha_n^{\bar{\rho}_{\sup}} \log(\delta_n/\alpha_n)\left(\epsilon\left(p_n^{(0)}\right) + \epsilon\left(p_n^{(2)}\right)\right)\right) + \mathcal{O}\left(\alpha_n^{\bar{\rho}_{\sup}} \log(1/\delta_n)\left(\epsilon(p_n^{(3)}) + \epsilon(p_n^{(4)})\right)\right) \\
&\quad + \mathcal{O}\left(\alpha_n^{\bar{\rho}_{\sup}} \epsilon(p_n^{(1)})\right) + \mathcal{O}(1) \\
&\quad = \mathcal{O}\left(\alpha_n^{\bar{\rho}_{\sup}} \log(\delta_n/\alpha_n)\left(\left(p_n^{(0)}\right)^{-2} + \left(p_n^{(2)}\right)^{-2}\right)\right) + \mathcal{O}\left(\alpha_n^{\bar{\rho}_{\sup}} \log(1/\delta_n)\left(\left(p_n^{(3)}\right)^{-2} + \left(p_n^{(4)}\right)^{-2}\right)\right) \\
&\quad + \mathcal{O}\left(\alpha_n^{\bar{\rho}_{\sup}}\left(p_n^{(1)}\right)^{-2}\right) + \mathcal{O}(1),
\end{aligned}$$

and the result follows. □

Proof of Theorem 4. Introducing

$$\begin{aligned}
g_n &= g(\log(\delta_n/\alpha_n), \log(1/\delta_n), \log(\delta_n/\tau_n)),\\
\tilde{g}_n(\tilde{\phi}) &= \tilde{g}_{\tilde{\phi}}^{\mathtt{NN}_J}(\log(\delta_n/\alpha_n), \log(1/\delta_n), \log(\delta_n/\tau_n)),\\
f_{\delta_n,\cdot} &= f_Z(\log(\delta_n/\cdot), \log(1/\delta_n)),\\
\tilde{f}_{\delta_n,\cdot}(\tilde{\phi}) &= \tilde{f}_{\tilde{\phi}}^{\mathtt{NN}_J}(\log(\delta_n/\cdot), \log(1/\delta_n)),\\
\omega_n(\tilde{\phi}) &= \frac{\tilde{g}_n(\tilde{\phi})}{g_n} - 1,\\
\lambda_n(y) &= \frac{q(1-\tau_n \mid y)}{q(1-\delta_n \mid y)},
\end{aligned}$$

we have

$$\begin{aligned}
&\inf_{\tilde{\phi}\in\Phi} \sup_{y\in\Pi} \left|\log q(1-\alpha_n \mid y) - \log \tilde{q}_{\tilde{\phi}}^{\mathtt{NN}_J}(1-\alpha_n; 1-\delta_n, 1-\tau_n \mid y)\right|\\
&= \inf_{\tilde{\phi}\in\Phi} \sup_{y\in\Pi} \left|\log(1+(1-\lambda_n(y))g_n) - \log(1+(1-\lambda_n(y))\tilde{g}_n(\tilde{\phi}))\right|\\
&= \inf_{\tilde{\phi}\in\Phi} \sup_{y\in\Pi} \left|\log\left(\frac{1+(1-\lambda_n(y))(1+\omega_n(\tilde{\phi}))g_n}{1+(1-\lambda_n(y))g_n}\right)\right|\\
&= \inf_{\tilde{\phi}\in\Phi} \sup_{y\in\Pi} \left|\log\left(1+\frac{(1-\lambda_n(y))\omega_n(\tilde{\phi})g_n}{1+(1-\lambda_n(y))g_n}\right)\right|\\
&= \inf_{\tilde{\phi}\in\Phi} \sup_{y\in\Pi} \left|\log\left(1+\frac{(1-\lambda_n(y))\omega_n(\tilde{\phi})}{(1/g_n)+(1-\lambda_n(y))}\right)\right|\\
&=: \inf_{\tilde{\phi}\in\Phi} \sup_{y\in\Pi} \left|\log\left(1+\Lambda_n(\tilde{\phi} \mid y)\right)\right|. \qquad (54)
\end{aligned}$$

Besides, remark that

$$\inf_{\tilde{\phi}\in\Phi} \sup_{y\in\Pi} \left|\Lambda_n(\tilde{\phi} \mid y)\right| = \inf_{\tilde{\phi}\in\Phi} \left|\omega_n(\tilde{\phi})\right| \frac{1-\inf_{y\in\Pi}\lambda_n(y)}{(1/g_n)+(1-\inf_{y\in\Pi}\lambda_n(y))}$$

and $\inf_{y\in\Pi}\lambda_n(y) \to 0$ as $n\to\infty$ since $b(\cdot)$ is lower bounded on Π. Since $\delta_n/\tau_n \to 0$ and $\delta_n/\alpha_n \to \infty$, we get $g_n \to \infty$ so that

$$\inf_{\tilde{\phi}\in\Phi} \sup_{y\in\Pi} \left|\Lambda_n(\tilde{\phi} \mid y)\right| \sim \inf_{\tilde{\phi}\in\Phi} \left|\omega_n(\tilde{\phi})\right|,$$

as $n\to\infty$. Let us now consider

$$d_{\delta_n,\cdot}(\tilde{\phi}) = \frac{\exp(\tilde{f}_{\delta_n,\cdot}(\tilde{\phi})) - 1}{\exp(f_{\delta_n,\cdot}) - 1},$$

so that $\omega_n(\tilde{\phi}) = d_{\delta_n,\alpha_n}(\tilde{\phi})/d_{\delta_n,\tau_n}(\tilde{\phi}) - 1$. Let us then remark that $|\exp(u)-1| \le 2\,|u|$ for any $|u| \le \log 2$ implies

$$\begin{aligned}
\inf_{\tilde{\phi}\in\Phi} \left|\exp(\tilde{f}_{\delta_n,\cdot}(\tilde{\phi}) - f_{\delta_n,\cdot}) - 1\right| &\le \inf_{\substack{\tilde{\phi}\in\Phi\\ |\tilde{f}_{\delta_n,\cdot}(\tilde{\phi}) - f_{\delta_n,\cdot}| \le \log 2}} \left|\exp(\tilde{f}_{\delta_n,\cdot}(\tilde{\phi}) - f_{\delta_n,\cdot}) - 1\right|\\
&\le 2 \inf_{\substack{\tilde{\phi}\in\Phi\\ |\tilde{f}_{\delta_n,\cdot}(\tilde{\phi}) - f_{\delta_n,\cdot}| \le \log 2}} \left|\tilde{f}_{\delta_n,\cdot}(\tilde{\phi}) - f_{\delta_n,\cdot}\right|\\
&=: \eta_{\delta_n,\cdot}
\end{aligned}$$

with $\eta_{\delta_n,\cdot} \to 0$ as $n \to \infty$ from Theorem 2. As a consequence, one has

$$\inf_{\tilde{\phi}\in\Phi} \left| d_{\delta_n,\cdot}(\tilde{\phi}) - 1 \right| = \inf_{\tilde{\phi}\in\Phi} \left| \frac{\exp(f_{\delta_n,\cdot}) \left(\exp\left(\tilde{f}_{\delta_n,\cdot}(\tilde{\phi}) - f_{\delta_n,\cdot} \right) - 1 \right)}{\exp(f_{\delta_n,\cdot}) - 1} \right| \leq \frac{\eta_{\delta_n,\cdot}}{1 - \exp(-f_{\delta_n,\cdot})}.$$

Now, $f_{\delta_n,\alpha_n} \to \infty$ and $f_{\delta_n,\tau_n} \to -\infty$ since $\delta_n/\alpha_n \to \infty$ and $\delta_n/\tau_n \to 0$ as $n \to \infty$ which entails that

$$\inf_{\tilde{\phi}\in\Phi} \left| d_{\delta_n,\alpha_n}(\tilde{\phi}) - 1 \right| = \mathcal{O}(\eta_{\delta_n,\alpha_n}),$$
$$\inf_{\tilde{\phi}\in\Phi} \left| d_{\delta_n,\tau_n}(\tilde{\phi}) - 1 \right| = \mathcal{O}\left(\eta_{\delta_n,\tau_n} \exp(f_{\delta_n,\tau_n}) \right).$$

Besides, applying twice the triangle inequality yields

$$\left| \frac{d_{\delta_n,\alpha_n}(\tilde{\phi})}{d_{\delta_n,\tau_n}(\tilde{\phi})} - 1 \right| \leq \frac{|d_{\delta_n,\alpha_n}(\tilde{\phi}) - 1|}{1 - |d_{\delta_n,\tau_n}(\tilde{\phi}) - 1|} + \frac{|d_{\delta_n,\tau_n}(\tilde{\phi}) - 1|}{1 - |d_{\delta_n,\tau_n}(\tilde{\phi}) - 1|}$$

and therefore

$$\begin{aligned} \inf_{\tilde{\phi}\in\Phi} \left| \omega_n(\tilde{\phi}) \right| &= \mathcal{O}(\eta_{\delta_n,\alpha_n}) + \mathcal{O}\left(\eta_{\delta_n,\tau_n} \exp(f_{\delta_n,\tau_n}) \right) \\ &= \mathcal{O}(\alpha_n^{-\bar{\rho}_J}) + \mathcal{O}(\tau_n^{-\bar{\rho}_J-\gamma} \delta_n^{\gamma} L_Z(1/\tau_n)/L_Z(1/\delta_n)), \end{aligned}$$

from Theorem 2. Since $\bar{\rho}_2 < 0$, one can show using Karamata's representation [15, Equation (B.1.9)] that the slowly-varying function L_Z tends to a constant at infinity, so that

$$\inf_{\tilde{\phi}\in\Phi} \left| \omega_n(\tilde{\phi}) \right| = \mathcal{O}(\alpha_n^{-\bar{\rho}_J}) + \mathcal{O}(\tau_n^{-\bar{\rho}_J-\gamma} \delta_n^{\gamma}),$$

which, in turn, implies that

$$\inf_{\tilde{\phi}\in\Phi} \sup_{y\in\Pi} \left| \Lambda_n(\tilde{\phi} \,|\, y) \right| \to 0$$

as $n \to \infty$. All in all, and taking account of $|\log(1+u)| \leq 2\,|u|$ for any $|u| \leq 1/2$, one has in view of (54):

$$\begin{aligned} &\inf_{\tilde{\phi}\in\Phi} \sup_{y\in\Pi} \left| \log q(1-\alpha_n \,|\, y) - \log \tilde{q}_{\tilde{\phi}}^{\mathtt{NN}_J}(1-\alpha_n; 1-\delta_n, 1-\tau_n \,|\, y) \right| \\ &\quad \leq 2 \inf_{\tilde{\phi}\in\Phi} \sup_{y\in\Pi} \left| \Lambda_n(\tilde{\phi}, y) \right| \\ &\quad \leq 3 \inf_{\tilde{\phi}\in\Phi} \left| \omega_n(\tilde{\phi}) \right| \\ &\quad = \mathcal{O}(\alpha_n^{-\bar{\rho}_J}) + \mathcal{O}(\tau_n^{-\bar{\rho}_J-\gamma} \delta_n^{\gamma}), \end{aligned}$$

which proves the result. □

Proof of Corollary 5. (i) If $\gamma + \bar{\rho}_J > 0$, balancing the two terms in (29) yields

$$\delta_n = \alpha_n^{-\bar{\rho}_J/\gamma} \tau_n^{1+\bar{\rho}_J/\gamma}.$$

One can then check that:

$$\delta_n/\tau_n = (\alpha_n/\tau_n)^{-\bar{\rho}_J/\gamma} \to 0,$$
$$\delta_n/\alpha_n = (\tau_n/\alpha_n)^{1+\bar{\rho}_J/\gamma} \to \infty,$$

since $\alpha_n/\tau_n \to 0$ as $n \to \infty$.
(ii) If $\gamma + \bar{\rho}_J \leq 0$, then, necessarily $\alpha_n^{-\bar{\rho}_J} = o(\tau_n^{-\bar{\rho}_J-\gamma} \delta_n^{\gamma})$. Therefore, letting $\delta_n = \xi_n \alpha_n$ and $\tau_n = \xi_n^2 \alpha_n$ with $\xi_n \to \infty$ as $n \to \infty$ proves the result. □

References

[1] A. A. Ahmad, E. H. Deme, A. Diop, S. Girard, and A. Usseglio-Carleve. Estimation of extreme quantiles from heavy-tailed distributions in a location-dispersion regression model. *Electron. J. Stat.*, 14(2):4421–4456, 2020.

[2] M. Allouche, J. El Methni, and S. Girard. A refined Weissman estimator for extreme quantiles. `https://hal.inria.fr/hal-03266676`, 2021.

[3] M. Allouche, S. Girard, and E. Gobet. EV-GAN: Simulation of extreme events with ReLU neural networks. *J. Mach. Learn. Res.*, 23(150):1–39, 2022.

[4] J. Beirlant, Y. Goegebeur, J. Segers, and J. Teugels. *Statistics of Extremes: Theory and Applications*. Wiley, 2004.

[5] Siddharth Bhatia, Arjit Jain, and Bryan Hooi. ExGAN: Adversarial generation of extreme samples. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 6750–6758, 2021.

[6] G. Biau and L. Devroye. *Lectures on the nearest neighbor method*. Springer, 2015.

[7] F. Caeiro, M.I. Gomes, and D. Pestana. Direct reduction of bias of the classical Hill estimator. *Revstat Stat J*, 3(2):113–136, 2005.

[8] J. Cai, J. Einmahl, L. de Haan, and C. Zhou. Estimation of the marginal expected shortfall: the mean when a related variable is extreme. *J. R. Stat. Soc. B*, 77:417–442, 2015.

[9] D. Ceresetti, G. Molinié, and J-D. Creutin. Scaling properties of heavy rainfall at short duration: A regional analysis. *Water Resour. Res.*, 46(9), 2010.

[10] S. Coles, J. Heffernan, and J. Tawn. Dependence measures for extreme value analyses. *Extremes*, 2(4):339–365, 1999.

[11] R. Corless, G. Gonnet, D. Hare, D. Jeffrey, and D. Knuth. On the Lambert W function. *Adv. Comput. Math.*, 5(1):329–359, 1996.

[12] G. Cybenko. Approximation by superpositions of a sigmoidal function. *Math. Control Signals Systems*, 2(4):303–314, 1989.

[13] A. Daouia, L. Gardes, and S. Girard. On kernel smoothing for extremal quantile regression. *Bernoulli*, 19:2557–2589, 2013.

[14] A. Daouia, L. Gardes, S. Girard, and A. Lekina. Kernel estimators of extreme level curves. *Test*, 20(14):311–333, 2011.

[15] L. de Haan and A. Ferreira. *Extreme value theory*. Springer Series in Operations Research and Financial Engineering. Springer, New York, 2006.

[16] L. de Haan, A. K. Tank, and C. Neves. On tail trend detection: modeling relative risk. *Extremes*, 18:141–178, 2015.

[17] C. de Valk. Approximation and estimation of very small probabilities of multivariate extreme events. *Extremes*, 19:687–717, 2016.

[18] J. H. J. Einmahl, L. de Haan, and C. Zhou. Statistics of heteroscedastic extremes. *J. R. Stat. Soc. B*, 78:31–51, 2016.

[19] P. Embrechts, C. Klüppelberg, and T. Mikosch. *Modelling Extremal Events for Insurance and Finance*. Springer, 1997.

[20] P. Embrechts and G. Puccetti. Aggregating risk across matrix structured loss data: the case of operational risk. *J. Oper. Risk*, 3:29–44, 2007.

[21] M. Falk, J. Hüsler, and R.-D. Reiss. *Laws of small numbers: extremes and rare events*. Birkhäuser/Springer Basel AG, Basel, 2011.

[22] I. Fraga Alves, L. de Haan, and T. Lin. Third order extended regular variation. *Publications de l'Institut Mathématique*, 80(94):109–120, 2006.

[23] A. K. Gangopadhyay. A note on the asymptotic behavior of conditional extremes. *Statist. Probab. Lett.*, 25(2):163–170, 1995.

[24] L. Gardes. A general estimator for the extreme value index: applications to conditional and heteroscedastic extremes. *Extremes*, 18(3):479–510, 2015.

[25] L. Gardes and S. Girard. A moving window approach for nonparametric estimation of the conditional tail index. *J. Multivariate Anal.*, 99(10):2368–2388, 2008.

[26] L. Gardes and S. Girard. Conditional extremes from heavy-tailed distributions: an application to the estimation of extreme rainfall return levels. *Extremes*, 13(2):177–204, 2010.

[27] L. Gardes and S. Girard. Functional kernel estimators of large conditional quantiles. *Electron. J. Stat.*, 6:1715–1744, 2012.

[28] L. Gardes, S. Girard, and A. Lekina. Functional nonparametric estimation of conditional extreme quantiles. *J. Multivariate Anal.*, 101:419–433, 2010.

[29] L. Gardes, A. Guillou, and C. Roman. Estimation of extreme conditional quantiles under a general tail-first-order condition. *Ann. Inst. Statist. Math.*, 72(4):915–943, 2020.

[30] L. Gardes, A. Guillou, and A. Schorgen. Estimating the conditional tail index by integrating a kernel conditional quantile estimator. *J. Stat. Plan. Inference*, 142(6):1586–1598, 2012.

[31] L. Gardes and G. Stupfler. Estimation of the conditional tail index using a smoothed local Hill estimator. *Extremes*, 17(1):45–75, 2014.

[32] S. Girard, G. Stupfler, and A. Usseglio-Carleve. Extreme conditional expectile estimation in heavy-tailed heteroscedastic regression models. *Ann. Statist.*, 49(6):3358–3382, 2021.

[33] Y. Goegebeur and T. de Wet. Estimation of the third-order parameter in extreme value statistics. *Test*, 21(2):330–354, 2012.

[34] Y. Goegebeur, A. Guillou, and G. Stupfler. Uniform asymptotic properties of a nonparametric regression estimator of conditional tails. *Ann. Inst. H. Poincare Probab. Statist.*, 51(3):1190–1213, 2015.

[35] M. I. Gomes, M. F. Brilhante, F. Caeiro, and D. Pestana. A new partially reduced-bias mean-of-order p class of extreme value index estimators. *Comput. Statist. Data Anal.*, 82:223–237, 2015.

[36] M. I. Gomes, M. F. Brilhante, and D. Pestana. New reduced-bias estimators of a positive extreme value index. *Comm. Statist. Simulation Comput.*, 45(3):833–862, 2016.

[37] M.I. Gomes, L. de Haan, and L. Peng. Semi-parametric estimation of the second order parameter in statistics of extremes. *Extremes*, 5(4):387–414, 2002.

[38] M.I. Gomes and D. Pestana. A sturdy reduced-bias extreme quantile (VaR) estimator. *J. Amer. Statist. Assoc.*, 102(477):280–292, 2007.

[39] I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. Adaptive Computation and Machine Learning. MIT Press, Cambridge, MA, 2016.

[40] P. Hall. On some simple estimates of an exponent of regular variation. *J. Roy. Statist. Soc. Ser. B*, 44(1):37–42, 1982.

[41] P. Hall and A. H. Welsh. Adaptive estimates of parameters of regular variation. *Ann. Statist.*, 13(1):331–341, 1985.

[42] M. Hallin. Measure transportation and statistical decision theory. *Annu. Rev. Stat. Appl.*, 9(1):401–424, 2022.

[43] X. He and P. Ng. Quantile splines with several covariates. *J. Statist. Plann. Inference*, 75(2):343–352, 1999.

[44] B. M. Hill. A simple general approach to inference about the tail of a distribution. *Ann. Statist.*, 3(5):1163–1174, 1975.

[45] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 2014.

[46] R. Koenker, P. Ng, and S. Portnoy. Quantile smoothing splines. *Biometrika*, 81(4):673–680, 1994.

[47] N. Kottegoda and R. Rosso. *Statistics, probability and reliability for civil and environmental engineers*. Mc-Graw-Hill Publishing Company, 2008.

[48] D. Koutsoyiannis. On the appropriateness of the gumbel distribution for modelling extreme rainfall. In *ESF Exploratory*, pages 24–25, 2003.

[49] Y. A. LeCun, L. Bottou, G. B. Orr, and K-R. Müller. Efficient backprop. In *Neural networks: Tricks of the trade*, pages 9–48. Springer, 2012.

[50] G. Molinié, D. Ceresetti, S. Anquetin, J-D. Creutin, and B Boudevillain. Rainfall regime of a mountainous mediterranean region: Statistical analysis at short time steps. *J. Appl. Meteorol. Climatol.*, 51(3):429–448, 2012.

[51] C. Neves. From extended regular variation to regular variation with application in extreme value statistics. *J. Math. Anal. Appl.*, 355(1):216–230, 2009.

[52] S. Ould Abdi, S. Dabo-Niang, A. Diop, and A. Ould Abdi. Consistency of a nonparametric conditional quantile estimator for random fields. *Math. Methods Stat.*, 19(1):1–21, 2010.

[53] S. Resnick. *Heavy-Tail Phenomena: Probabilistic and Statistical Modeling*. Springer, 2007.

[54] P.R. Shukla, J. Skea, R. Slade, A. Al Khourdajie, R. van Diemen, D. McCollum, M. Pathak, S. Some, P. Vyas, R. Fradera, M. Belkacemi, A. Hasija, G. Lisboa, S. Luz, and J. Malley (eds.). *Climate Change 2022: Mitigation of Climate Change. Contribution of Working Group III to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change*. Cambridge University Press. Cambridge, UK and New York, NY, 2022.

[55] H.M.G.M. Steenbergen, B.L. Lassing, A.C.W.M. Vrouwenvelder, and P.H. Waarts. Reliability analysis of flood defence systems. *Heron*, 49:51–73, 2004.

[56] C. Stone. Consistent nonparametric regression. *Ann. Statist.*, 5:595–620, 1977.

[57] I. Van Keilegom and L. Wang. Semiparametric modeling and estimation of heteroscedasticity in regression analysis of cross-sectional data. *Electron. J. Stat.*, 4:133–160, 2010.

[58] H. J. Wang and D. Li. Estimation of extreme conditional quantiles through power transformation. *J. Amer. Statist. Assoc.*, 108:1062–1074, 2013.

[59] H. J. Wang, D. Li, and X. He. Estimation of high conditional quantiles for heavy-tailed distributions. *J. Amer. Statist. Assoc.*, 107:1453–1464, 2012.

[60] X.Q. Wang and S.H. Cheng. General regular variation of the n-th order and 2nd order Edgeworth expansions of the extreme value distribution. II. *Acta Math. Sin. (Engl. Ser.)*, 22(1):27–40, 2006.

[61] I. Weissman. Estimation of parameters and large quantiles based on the k largest observations. *J. Amer. Statist. Assoc.*, 73(364):812–815, 1978.

[62] M. Wiese, R. Knobloch, R. Korn, and P. Kretschmer. Quant GANs: deep generation of financial time series. *Quant. Finance*, 20(9):1419–1440, 2020.

[63] D. Yarotsky. Optimal approximation of continuous functions by very deep ReLU networks. In *Proceedings Mach. Learn. Res.*, pages 639–649, 2018.

[64] K. Yu and M. Jones. Local linear quantile regression. *J. Amer. Statist. Assoc.*, 93(441):228–237, 1998.