Marginal Tail-Adaptive Normalizing Flows

Mike Laszkiewicz [1 2] **Johannes Lederer** [1] **Asja Fischer** [2]

Abstract

Learning the tail behavior of a distribution is a notoriously difficult problem. By definition, the number of samples from the tail is small, and deep generative models, such as normalizing flows, tend to concentrate on learning the body of the distribution. In this paper, we focus on improving the ability of normalizing flows to correctly capture the tail behavior and, thus, form more accurate models. We prove that the marginal tailedness of an autoregressive flow can be controlled via the tailedness of the marginals of its base distribution. This theoretical insight leads us to a novel type of flows based on flexible base distributions and data-driven linear layers. An empirical analysis shows that the proposed method improves on the accuracy—especially on the tails of the distribution—and is able to generate heavy-tailed data. We demonstrate its application on a weather and climate example, in which capturing the tail behavior is essential.

1. Introduction

Heavy-tailed distributions are known to occur in various applications in biology, finance, climate, and many other fields. Quantities with a heavy-tailed distribution are, for example, the length of protein sequences in genomes (Koonin et al., 2006), returns of stocks (Gabaix et al., 2003), or the occurence and impacts of weather and climate events (Katz, 2002). In these applications, heavy-tailed events are often the most substantial samples and hence, ignoring them—thinking of underestimating a maximum flood level or the loss of a financial crisis—would yield to crucial model failures. From a theoretical point of view, heavy-tailed distributions emerge from several circumstances, including the limiting distribution in the generalized central limit theorem, of a multiplicative process, or as the limit of an extremal process (Nair et al., 2013). Further, many distributions that arise from a functional relationship, such as the ratio of two standard normal distributed random variables, are heavy-tailed, highlighting their importance for models that incorporate known physical relationships among quantities. Given the frequency of occurrence and their potential impact, developing generative models that allow to learn heavy-tailed distributions are an urgent task to solve. We approach this task by providing important theoretical groundings regarding the expressiveness of Normalizing Flows (NFs) for heavy-tailed data.

Normalizing Flows (Rippel & Adams, 2013; Tabak & Turner, 2013; Dinh et al., 2015; Rezende & Mohamed, 2015) are a popular class of deep generative models. Despite their success in learning tractable distributions where both sampling and density evaluation can be efficient and exact, their ability to model heavy-tailed distributions is known to be limited. Jaini et al. (2020) identified the problem that autoregressive affine NFs are unable to map a light-tailed distribution to a heavy-tailed distribution. They propose to solve this issue by replacing the Gaussian base distribution by a multivariate t-distribution with one learnable degree of freedom, leading to a model referred to as Tail-Adaptive Flows (TAF).

Contributions In this paper, we extend the work of Jaini et al. (2020) in multiple ways: First, while TAF allows to model distributions with a heavy-tailed Euclidean norm, we show that modeling multivariate distributions, where some of the marginals are heavy- and some are light-tailed, still poses a problem. More precisely, we identify the problem that an autoregressive affine NF using a base distribution with solely heavy-tailed marginals (such as TAF) is only able to provide a target distribution with just heavy-tailed marginals as well. Consequently, such a NF is not capable of learning distributions with mixed marginal tail behavior. Second, to solve the problem, we derive a theoretical result that states conditions under which the marginal tailedness of the base distribution is preserved. Third, we turn these theoretical insights into a novel modification of autoregressive NFs, which allows to model the marginal tail behavior. Since the proposed model preserves marginal tailedness, we call it *marginal tail-adaptive flow* (mTAF). The proposed

[*]Equal contribution [1]Faculty of Mathematics, Ruhr University, Bochum, Germany [2]Center of Computer Science, Bochum, Germany. Correspondence to: Mike Laszkiewicz <Mike.Laszkiewicz@rub.de>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

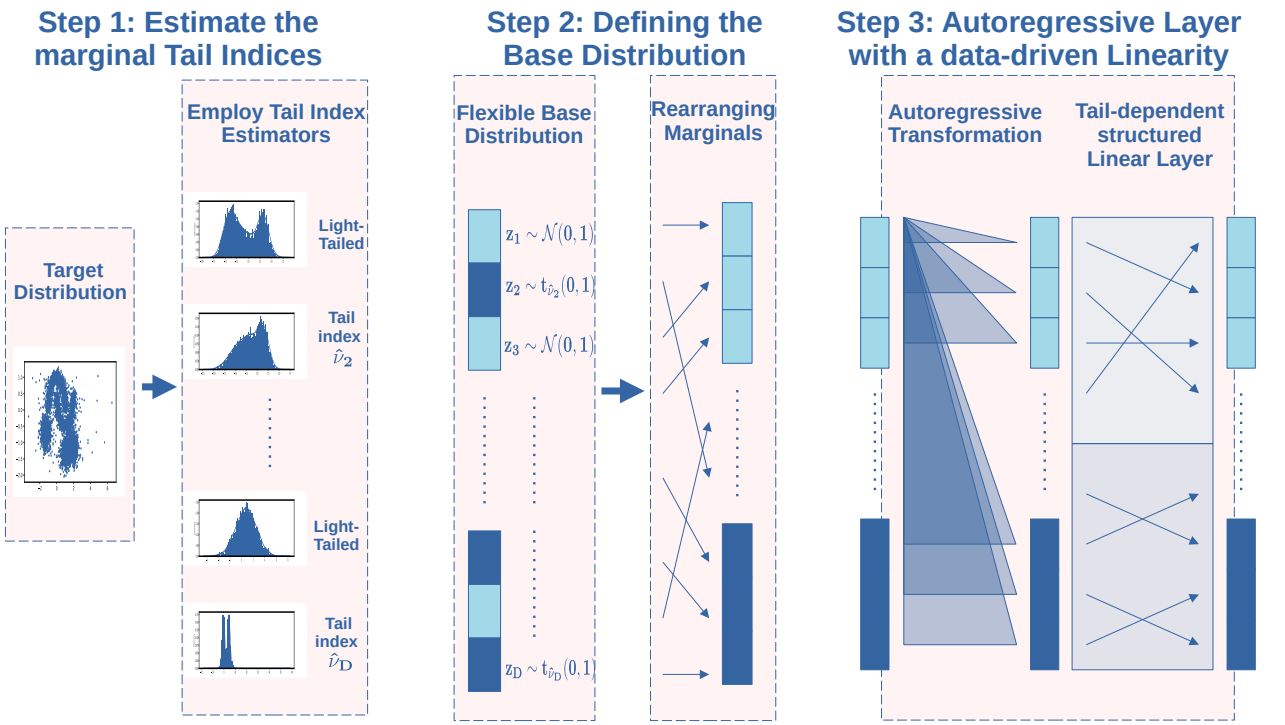


Figure 1. An Overview of mTAF. In a first step, we apply estimators from extreme value theory to classify the marginals as heavy- or light-tailed. This classification defines a flexible base distribution consisting of marginal Gaussians and marginal t-distributions with flexible degree of freedom, as illustrated by Step 2 of this figure. Further, we rearrange the marginals such that the first d_l marginals are light-tailed, whereas the remaining marginals are heavy-tailed. mTAF is then constructed using several flow-layers as visualized in Step 3: we employ a triangular mapping, followed by a 2-group permutation scheme, which can be generalized to general 2-group linearities (Section 3.3). At the end, we restore the original ordering using the inverse of the permutation employed in Step 2. Using Theorem 3.4, we prove that mTAFs are marginally tail-adaptive (Corollary 3.5).

method combines estimators from extreme value theory, a flexible base distribution, and a novel data-driven type of linearities as illustrated in Figure 1. Furthermore, we notice that the autoregressive layers in Neural Spline Flows (NSFs (Durkan et al., 2019)), which are a SOTA architecture, are linear in their tails and, therefore, that we can apply all theory derived by Jaini et al. (2020) and presented in this paper for neural spline layers. We introduce a simple modification on the LU-layers that ensures that NSF preserve the full marginal tailedness structure of the base distribution. Lastly, we present a generalized and more flexible version of mTAF, which we call *generalized Tail-Adaptive Flow* (gTAF). Our theory is backed up by an experimental analysis demonstrating the superior performance of the proposed methods in learning the tails, especially when it comes to generating synthetic tail samples. Finally, we apply mTAF and gTAF on a climate example to generate heavy-tailed weather data.

Notational Conventions In the following, we will denote random variables by bold letters, such as $\mathbf{x}$, and its realisations by non-bold letters, x. We use this notation for multivariate and for univariate random variables. Further, we denote the jth component of $\mathbf{x}$ by $\mathbf{x}_j$, and $\mathbf{x}_{\leq j}$ or $\mathbf{x}_{<j}$ are the first j or $j-1$ components of $\mathbf{x}$, respectively. We denote the random variable representing the base distribution by $\mathbf{z}$ and the random variable representing the target distribution by $\mathbf{x}$. Further, for notational convenience, we denote the probability density functions (PDFs) of $\mathbf{x}$ and $\mathbf{z}$ by p and q. Finally, we assume that both random variables $\mathbf{x}$ and $\mathbf{z}$ have continuous and positive density on $\mathbb{R}^D$, i.e $p(x),\ q(z) > 0$ for all $x, z \in \mathbb{R}^D$, where D is the dimensionality of $\mathbf{x}$ and $\mathbf{z}$.

2. Background

In this section, we give a brief introduction to heavy-tailed distributions and present needed background knowledge about normalizing flows.

2.1. Heavy-tailed Distributions

Heavy-tailed distributions are distributions that have heavier tails (i.e. decay slower) than the exponential distribution. Loosely speaking, slowly decaying tails allow to model dis-

tributions that generate samples, which differ by a large magnitude from the rest of the samples. For a univariate random variable $\mathbf{x}$ we define heavy-tailedness via its moment-generating function[1]:

Definition 2.1 (Heavy-Tailed Random Variables)**.** Consider a random variable $\mathbf{x} \in \mathbb{R}$ with PDF p. We say that $\mathbf{x}$ is heavy-tailed if and only if

$$\forall \lambda > 0 \ : \ \mathbb{E}_{\mathbf{x}}\big[e^{\lambda \mathbf{x}}\big] = \infty \ .$$

The function $m_p(\lambda) := \mathbb{E}_{\mathbf{x}}[\exp(\lambda \mathbf{x})]$ is known as the moment-generating function of $\mathbf{x}$. Random variables that are not heavy-tailed are said to be light-tailed.

Note that this definition is, strictly speaking, merely a definition for heavy right tails. We say a random variable $\mathbf{x} \in \mathbb{R}$ has heavy left tails if $-\mathbf{x}$ has heavy tails according to Definition 2.1. For simplicity of derivations and w.l.o.g., we proceed with this definition but the derived results can analogously be applied to left tails.

We can assess the degree of tailedness of a distribution. While there are many equivalent notions of the so called tail index, the most straight-forward definition is via the existence of moments:

Definition 2.2 (Tail Index)**.** A random variable $\mathbf{x} \in \mathbb{R}$ with PDF p is said to have tail index[2] α if it holds that

$$\mathbb{E}_{\mathbf{x}}[|\mathbf{x}|^{\beta}] \begin{cases} < \infty \ , & \text{if } \beta < \alpha \ , \\ = \infty \ , & \text{if } \beta > \alpha \ . \end{cases}$$

Since the tail index is tightly related to the decay rate of the PDF, it enables us to assess the degree of heavy-tailedness of a random variable. Therefore, estimation of the tail index became an important objective in extreme value theory and statistical risk assessment (see e.g. Embrechts et al., 2013). Since the existence of the moment does not depend on the "body" of $\mathbf{x}$ but only on the tails of $\mathbf{x}$ (see Proposition A.1 in Section A.1 in the Appendix), estimating the tail index by fitting a full parametric model to all data e.g. via likelihood maximization leads to a biased estimator. Instead, semi-parametric estimators have been developed, which aim to fit a distribution only on the tails. Popular methods for tail estimation include the Hill estimator (Hill, 1975), the moment estimator (Dekkers et al., 1989), and kernel-based estimators (Csorgo et al., 1985). In Section C.1 of the Appendix, we discuss these tail estimators and review some practical issues with these.

An example of a heavy–tailed distribution is the standardized t-distribution, which has parameter $\nu > 0$ referred to as the degree of freedom and a density function given by

$$p(x) := \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sqrt{\nu\pi}\,\Gamma\left(\frac{\nu}{2}\right)}\left(1 + \frac{x^2}{\nu}\right)^{-\frac{\nu+1}{2}} \ , \quad x \in \mathbb{R} \ ,$$

where Γ is the Gamma function. It is known that the t-distribution has tail index ν (see e.g. Kirkby et al. (2019) for a detailed reference).

In the multivariate setting, there exist various definitions of heavy-tailedness. For instance Resnick (2004) make use of a definition based on multivariate regular variation. Jaini et al. (2020) define a multivariate random variable $\mathbf{x}$ to be heavy-tailed if the ℓ_2-norm is heavy-tailed, a property which we refer to as ℓ_2-heavy-tailed, and which is formally defined as follows:

Definition 2.3 (ℓ_2-Heavy-Tailed)**.** Let $\mathbf{x} \in \mathbb{R}^D$ be a multivariate random variable. Then, we call $\mathbf{x}$ ℓ_2-heavy-tailed if $\|\mathbf{x}\|$ is univariately heavy-tailed according to Definition 2.1, where $\|\cdot\|$ denotes the ℓ_2-norm. Otherwise, we call $\mathbf{x}$ ℓ_2-light-tailed.

2.2. Normalizing Flows

The fundamental idea behind NFs is based on the change-of-variables formula for probability density functions (PDFs) given in the following theorem.

Theorem 2.4 (Change-of-Variables)**.** *Consider random variables $\mathbf{x}, \mathbf{z} \in \mathbb{R}^D$ and a diffeomorphic map $T : \mathbb{R}^D \to \mathbb{R}^D$ such that $\mathbf{x} = T(\mathbf{z})$. Then, it holds that the PDF of $\mathbf{x}$ satisfies*

$$p(x) = q\big(T^{-1}(x)\big)\big|\det J_{T^{-1}}(x)\big| \quad \forall x \in \mathbb{R}^D \ , \qquad (1)$$

where $J_{T^{-1}}(x)$ is the Jacobian of T^{-1} evaluated at $x \in \mathbb{R}^D$.

This formula allows us to evaluate the possibly intractable PDF of $\mathbf{x}$ if we can evaluate both, the PDF of $\mathbf{z}$ and $T^{-1}(x)$, and efficiently calculate the Jacobian-determinant $\det J_{T^{-1}}(x)$. As T maps $\mathbf{z}$ to $\mathbf{x}$, we denote the distribution of $\mathbf{z}$ and $\mathbf{x}$ as the *base* and the *target distribution*, respectively.

To model the PDF of $\mathbf{x}$ using NFs, it is common to set the base distribution to a standard normal distribution (i.e., $\mathbf{z} \sim \mathcal{N}(0, I)$) and to employ likelihood maximization to learn a parameterized transformation

$$T_\theta := T_\theta^{(L)} \circ \cdots \circ T_\theta^{(1)},$$

which, yet, remains tractable and diffeomorphic. Masked autoregressive flows (MAFs (Papamakarios et al., 2017)) are

[1] One can readily show that this definition is equivalent to the definition, which compares the tails of $\mathbf{x}$ to the tails of an exponential distribution. See Section 1 in Nair et al. (2013).

[2] Notice that the notion of a tail index is only valid for regularly varying random variables, which are a subclass of heavy-tailed random variables. For the purpose of this work, it is sufficient to consider regularly varying random variables. More details can be found in Nair et al. (2013).

one popular architecture[3], which employ transformations $T = (T_1, \dots, T_D)^\top$ of the form

$$T_j(z) := \mu_j(z_{<j}) + \exp(\sigma_j(z_{<j}))z_j \quad \text{for } j \in \{1, \dots, D\} \ , \tag{2}$$

where μ_j and σ_j are neural networks, which obtain the first $j-1$ components of z as input and output a scalar. Composing several transformations of the form (2), we obtain the MAF. The autoregressive form in (2) allows us to efficiently evaluate the Jacobian-Determinant due to the triangular form of $J_T(x)$, which is why autoregressive NFs are also referred to as triangular flows. By shuffling the ordering of the components, i.e. applying a permutation after each autoregressive transformation, we are able to form more diverse causal dependencies, leading to more expressive models. It has been shown that replacing the permutations by more general invertible layers can further improve the estimation performance (Oliva et al., 2018). In summary MAFs consist of multiple consecutive layers $T_\theta^{(l)} \circ P^{(l)}$, where $P^{(l)} \in \mathbb{R}^{D\times D}$ is some linear layer. Other examples for triangular flows include NAF (Huang et al., 2018), and NSF (Durkan et al., 2019), where the latter substitutes the affine transformation (2) by a rational of two splines. Further types of NFs include invertible ResNets (Jacobsen et al., 2018; Behrmann et al., 2019; Chen et al., 2019), continuous flows (Chen et al., 2018; Grathwohl et al., 2019), and many more (Kobyzev et al., 2020).

Tail-Adaptive Flows. Jaini et al. (2020) investigated the ability of triangular flows to learn heavy-tailed distributions. The authors have shown that if a triangular affine flow transforms a ℓ_2-light-tailed distribution, such as the multivariate Gaussian distribution, to a ℓ_2-heavy-tailed target distribution, then T_θ cannot be Lipschitz continuous. And more explicitly, it holds the following.

Theorem 2.5. *(Jaini et al., 2020) Let* $\mathbf{z}$ *be a* ℓ_2*-light-tailed random variable and* T *be an affine triangular flow such that* $T_j(z_{\leq j}) = \mu_j(z_{<j}) + \sigma_j(z_{<j})z_j$ *for all* j*. If* σ_j *is bounded above and* μ_j *is Lipschitz for all* j*, then the transformed variable* $\mathbf{x}$ *is also* ℓ_2*-light-tailed.*

Furthermore, the authors prove that any triangular mapping from an elliptical distribution to a heavier-tailed elliptical distribution must have an unbounded Jacobian-determinant. These results illuminate that learning a heavy-tailed distribution using NFs leads to non-Lipschitz transformations and unbounded Jacobians, which inevitably affects training robustness (Behrmann et al., 2021). Motivated by these result, Jaini et al. (2020) propose *Tail-Adaptive Flows* (TAF), which replace the Gaussian base distribution by a multivariate t-distribution with one learnable degree of freedom.

[3]Stricly speaking, (2) shows the transformations of an IAF (Kingma et al., 2016). However, MAF and IAF are theoretically equivalent, the differences lie only in their architectures.

3. Learning the correct marginal Tail Behavior with mTAF

In this section, we present a simple extension to triangular affine flows that allows to model distributions with a flexible tail behavior. We start by presenting our theoretical results in Section 3.1. Motivated by these results, we propose *marginally Tail-Adaptive Flow* (mTAF) in Section 3.2, which we apply to NSFs in Section 3.3. Lastly, we present a more flexible relaxation of mTAF, which we call *generalized Tail-Adaptive Flow* (gTAF) in Section 3.4.

3.1. The Necessity of a flexible Base Distribution

In this work, we investigate the tailedness of NFs more thoroughly through the lense of marginal tailedness, i.e. we consider the univariate tailedness of the marginal distributions of $\mathbf{x}_j$.Therefore, we introduce the following definitions:

Definition 3.1 (j-heavy-tailed, mixed-tailed, fully heavy–tailed, equal Tail Behavior)**.** We call a random variable $\mathbf{x} \in \mathbb{R}^D$ j-heavy-tailed if its jth marginal $\mathbf{x}_j$ is heavy-tailed according to Definition 2.1. Otherwise, we call $\mathbf{x}$ j-light-tailed. $\mathbf{x}$ is said to be mixed-tailed if there exists j_1, j_2 such that $\mathbf{x}$ is j_1-heavy-tailed and j_2-light-tailed. Further, we say that $\mathbf{x}$ is fully heavy-tailed if $\mathbf{x}$ is j-heavy-tailed for all $j \in \{1, \dots, D\}$. We define two random variables $\mathbf{x}$ and $\mathbf{z}$ to have equal tail behavior if it holds for all j that

$$\mathbf{x} \text{ is } j\text{–heavy–tailed} \quad \Leftrightarrow \quad \mathbf{z} \text{ is } j\text{–heavy–tailed} \ .$$

We found the following relation to Definition 2.3.

Proposition 3.2 (j-heavy-tailedness induces ℓ_2-heavy-tailedness)**.** *Assume that* $\mathbf{x}$ *is* j*-heavy-tailed for any* j*. Then,* $\mathbf{x}$ *is also* ℓ_2*-heavy-tailed.*

The proof can be found in Section A.1 in the Appendix. The proposition shows that j-heavy-tailedness is a more specific notion of multivariate heavy-tailedness than ℓ_2-heavy-tailedness, which allows a narrow inspection of the tail behavior. More precisely, the new notion allows us to differentiate between fully heavy-tailed random variables and mixed-tailed random variables, which are both ℓ_2-heavy-tailed. The first result states that, under mild technical conditions, fully heavy-tailedness of the base distribution is preserved by triangular affine maps.

Proposition 3.3 (Triangular affine Maps preserve fully heavy-tailedness)**.** *Let* $\mathbf{z}$ *be a fully heavy-tailed random variable that satisfies Assumption A.6*[4] *and let* T *be a a triangular affine map, that is,* $T_j(z_j, z_{<j}) = \mu_j(z_{<j}) + \sigma_j(z_{<j})z_j$ *with* $\sigma_j > 0$*. Then, it holds that* $T(\mathbf{z})$ *is also fully heavy-tailed.*

We provide a formal proof in Section A.2 of the Appendix. Assumption A.6 is a mild condition on the decay rate of

[4]This Assumption can be found in Section A.2 in the Appendix.

the copula density of $\mathbf{z}$. In Section A.3 of the Appendix, we explain this condition in more detail and give various examples.

It is clear that permuting the marginals does not change the heavy-tailedness. Hence, by iterative application of Proposition 3.3, we deduce that affine triangular flows that employ permutation layers and a fully heavy-tailed base distribution are unable to model mixed tailed distributions. Implicitly, Proposition 3.3 states that a Lipschitz normalizing flow as proposed by Jaini et al. (2020) is not able to model mixed-tailed distributions. The following theorem provides sufficient conditions under which a flow is able to model mixed-tailed distributions, which guides us towards a marginally tail-adaptive flow architecture.

Theorem 3.4 (Learning the correct Tail Behavior)**.** *Consider a random-variable $\mathbf{z}$ that is j-light-tailed for $j \in \{1, \dots, d_l\}$ for some $d_l < D$ and j-heavy-tailed for $j \in \{d_l + 1, \dots, D\}$. Then, under the same conditions as in Theorem 2.5 and Proposition 3.3, it holds that $\mathbf{z}$ and $T(\mathbf{z})$ have the same tail behavior.*

Proof. Since the result combines Theorem 2.5 and Proposition 3.3 in an evident fashion, we just quickly present a sketch of the proof. First, let us consider $j \leq d_l$. Then it holds for the moment-generating function of $\mathbf{x}_j$ that

$$\begin{aligned} m_{\mathbf{x}_j}(\lambda) &= \int_{\mathbb{R}^D} e^{\lambda T_j(z_{\leq j})} q(z) dz \\ &= \int_{\mathbb{R}^j} e^{\lambda T_j(z_{\leq j})} p_{\leq j}(z_{\leq j}) dz_{\leq j} \ , \end{aligned}$$

which has been shown to be bounded for some $\lambda > 0$ (see the proof of Theorem 2.5 in Jaini et al. (2020)). Therefore, $\mathbf{x}$ is j-light-tailed for all $j \leq d_l$. In the case $j > d_l$, we notice[5] that the proof for heavy-tailedness of $T_j(\mathbf{z}_{\leq j})$ involves just the heavy-tailedness of $\mathbf{z}_j$ and not of any other component of $\mathbf{z}_{<j}$. Hence, if $\mathbf{z}_j$ is heavy-tailed, then $\mathbf{x}_j = T_j(\mathbf{z}_{\leq j})$ is also heavy-tailed, regardless of $\mathbf{z}_{<j}$. Therefore, $\mathbf{x}$ is j-heavy-tailed for all $j > d_l$, which completes the proof. Note that in general we cannot deduce the latter conclusion for light-tailed marginals, i.e. if $\mathbf{z}_j$ is light-tailed, this does not mean that $\mathbf{x}_j$ is also light-tailed. This is only the case, if all $\mathbf{z}_{<j}$ are light-tailed as well. □

3.2. Marginally Tail-Adaptive Flow (mTAF)

Our main result, Theorem 3.4, prompts that if we maintain an ordering of the marginals such that the first marginals are light-tailed and the following are heavy-tailed in each flow step, we retain the marginal tail behavior of the base distribution in the estimated target distribution. This finding motivates the novel NF proposed in this paper. The proposed approach combines research findings from extreme value theory (Embrechts et al., 2013; Nair et al., 2013), recent findings about normalizing flows (Jaini et al., 2020; Alexanderson & Henter, 2020; Laszkiewicz et al., 2021), and the results presented herein. The proposed mTAFs consists of three steps that are depicted in Figure 1 and described in the following:

Step 1: Estimating the marginal tail indices and defining the marginal distributions. For each marginal, i.e. for the marginal distribution q_j of each $\mathbf{x}_j$, $j = 1, \dots, D$, we assess heavy- or light-tailedness using tail estimators. If the marginal is predicted to be light-tailed, we set the corresponding marginal base distribution to be standard normal distributed $\mathbf{z}_j \sim \mathcal{N}(0, 1)$. Otherwise we set the marginal to the standardized t-distribution with the estimated degree of freedom, i.e. $\mathbf{z}_j \sim t_{\hat{\nu}_j}$, where $\hat{\nu}_j$ is the Hill double-bootstrap estimator (Danielsson et al., 2001; Qi, 2008). We present all the details about the tail-assessment scheme in Section C.1.

Step 2: Defining the base distribution. We construct the base distribution as the mean-field approximation of the marginals, i.e. $\mathbf{z}$ has the density $q(z) := \prod_{j=1}^{D} q_j(z_j)$ with marginal densities q_j defined in step 1. Further, to satisfy the assumptions of Theorem 3.4, we need to permute the marginals such that it holds $\mathbf{z}_j \sim \mathcal{N}(0, 1)$ for $j \leq d_l$ and $\mathbf{z}_j \sim t_{\hat{\nu}_j}$ for $j > d_l$. We apply the same permutation to restructure our data according to the base components. To account for tail index estimation errors and for more flexible learning, one can make the tail indices (i.e. the degrees of freedom of each t-distribution) learnable. That is, we initialize the degree of freedom of the jth marginal with $\hat{\nu}_j$ but adapt the parameter together with the network parameters throughout training.

Step 3: A data-driven permutation scheme. Recall, that vanilla autoregressive flows employ a permutation step after each transformation to enhance the mixing of variables. However, purely random permutations might lead to a violation on the ordering of marginals, which is necessary to ensure Theorem 3.4. Therefore, we permute only within the set of heavy-tailed marginals and within the set of light-tailed marginals, to ensure the validity of Theorem 3.4. Within these groups one can choose any permutation scheme. We generalize this result for LU-layers in Section 3.3.

Without loss of generality, we assume that the first d_l components of $\mathbf{z}$ are light-tailed and the remaining $D - d_l$ components are heavy-tailed[6]. Then, the training objective is to optimize for flow parameters $\hat{\theta}$ and degrees of freedom

[5] For details, we refer to the proof of Proposition 3.3 in the Appendix.

[6] Otherwise we permute the marginals as described in Step 2.

$\hat{\nu} = [\hat{\nu}_{d_l+1}, \dots, \hat{\nu}_D]$ to maximize the log-likelihood

$$L(\hat{\theta}, \hat{\nu}; X) = \sum_{j=1}^{N} \Bigg\{ \sum_{i=1}^{d_l} \log \pi\Big(T_{\hat{\theta}}^{-1}\big(x^{(j)}\big)_i\Big) + \sum_{i=d_l+1}^{D} \log t_{\hat{\nu}_i}\Big(T_{\hat{\theta}}^{-1}\big(x^{(j)}\big)_i\Big) - \log\det J_{T_{\hat{\theta}}}\big(x^{(j)}\big) \Bigg\}$$

where $X := (x^{(1)}, \dots x^{(N)})$ is the data, and π and $t_{\hat{\nu}}$ are the PDF of the standard normal distribution and the standard t-distribution with $\hat{\nu}$ degrees of freedom, respectively.

When applying our theoretical results presented in the previous section to the proposed mTAF, we can show that it fulfills the desired tail-preserving property, as formalized by the following corollary:

Corollary 3.5 (Marginally tail-adaptive)**.** *Under the same assumptions as in Theorem 2.5 and in Proposition 3.3, mTAFs are marginally tail-adaptive, that is,* $\mathbf{z}$ *and* $\mathbf{x} = T(\mathbf{z})$ *have the same tail behavior.*

3.3. Marginal Tail-Adaptive Neural Spline Flows

Recent findings on NFs lead to significant improvements of their performance, such as employing LU-Layers instead of permutations (Kingma & Dhariwal, 2018; Oliva et al., 2018) and more expressive autoregressive layers. One of the current SOTA NFs, which combine both improvements, are NSFs (Durkan et al., 2019). In this section we apply the theory from the previous sections to NSFs with a modified version of the LU-Layers, while retaining their computational benefits.

First, let us provide sufficient conditions under which linear layers preserve the marginal tail behavior:[7]

Theorem 3.6. *Let* $\mathbf{z}$ *be a random variable that is* j*-light-tailed for* $j \in \{1, \dots, d_l\}$ *and* j*-heavy-tailed for* $j \in \{d_l + 1, \dots, D\}$*. Further, consider a block-diagonal invertible matrix*

$$W = \begin{pmatrix} A & 0 \\ B & C \end{pmatrix} \tag{3}$$

with $A \in \mathbb{R}^{d_l \times d_l}$, $B \in \mathbb{R}^{(D-d_l)\times d_l}$, $Cin\mathbb{R}^{(D-d_l)\times(D-d_l)}$ *and* 0 *is a zero matrix of size* $d \times (D - d_l)$*. Then, it follows that* $W\mathbf{z}$ *and* $\mathbf{z}$ *have equal tail behavior.*

As a special case of Lemma B.3, we can invert a block-diagonal matrix

$$\begin{pmatrix} A & 0 \\ B & C \end{pmatrix}^{-1} = \begin{pmatrix} A^{-1} & 0 \\ -C^{-1}BA^{-1} & C^{-1} \end{pmatrix} . \tag{4}$$

Further, the log-determinant of W is given by

$$\log\det(W) = \log\det(A) + \log\det(C) . \tag{5}$$

[7]The proof is presented in Section B.1.

Therefore, as long as log-determinant computation and inversion of A and C are efficient, we can efficiently employ a block-diagional matrix as a flow layer. Luckily, both is given if we parameterize A and C using the LU-decomposition, whereas B can be an arbitrary matrix.

Moreover, NSFs make use of monotonic rational-quadratic splines to define the autoregressive layers. These splines are defined within some interval $[-b, b]$ and are linearily extended outside this interval. Hence, the autoregressive NSF layers are in fact affine linear in their tails, which in turn means that we can apply all the theory from the previous sections on NSFs as well (compare with Lemma A.2).

In summary, NSF with linear layers according to the block-form (3) preserve the marginal tailedness of the base distribution. We want to highlight that even though each autoregressive NSF layer is linear outside $[-b, b]$, this does not mean that the whole flow is linear outside $[-b, b]$. This is because the (modified) LU-layers in between can map a component in and out of $[-b, b]$, leading to a non-trivial mapping outside that interval.

3.4. Generalized Tail-Adaptive Flows

Even though being theoretically founded, we introduce a more flexible relaxation of mTAF, which we call generalized TAF (gTAF): We drop the structural restrictions on the linearities and set the base distribution to a mean-field approximation of t-distributions with different trainable degrees of freedom. Therefore, gTAF is a compromise between the theoretically stronger mTAF and TAF. Note that since $t_\nu \xrightarrow{\nu\to\infty} \mathcal{N}(0, 1)$ we are able to approximately model heavy- as well as light-tailed marginals. Further, since LU-layers are trainable as well, gTAF is able to approximate mTAF arbitrary well by learning a structure as in (3).

Theorem 3.7. *Let* $T : \mathbb{R}^D \to \mathbb{R}^D$ *be almost surely continuous. Further, let* $z = (z_1, \dots, z_D)$ *be the mTAF base distribution with* d_l *Gaussian marginals and let* z_ν *be the gTAF base distribution with marginals*

$$z_{\nu,j} \sim \begin{cases} t_\nu(0, 1) & \text{for } j \le d_l \\ z_j & \text{for } j > d_l \end{cases} .$$

Then, it holds that $T(z_\nu) \xrightarrow{\nu\to\infty}_D T(z)$*, where* $\to_D$ *denotes convergence in distribution.*

Hence, when fixing the flow T, gTAF converges to the mTAF solution as the light-tailed degrees of freedom tend to ∞. We provide a proof in Section A.4.

4. Experimental Analysis

To investigate the benefits of the derived methods, we perform an empirical analysis on synthetic data (Section 4.1)

Table 1. Average test loss, Area under log-log plot, and tVaR (lower is better for each metric) in the setting $\nu = 2$ and $d_h \in \{1, 4\}$. The copula model serves as an oracle baseline.

d_h	1					4				
	L	Area_l	Area_h	tVaR_l	tVaR_h	L	Area_l	Area_h	tVaR_l	tVaR_h
vanilla	10.25	0.25	4.19	0.60	29.56	8.98	0.25	4.30	0.54	28.55
TAF	10.15	0.37	3.55	0.78	4.21	8.69	0.42	4.05	0.89	4.36
gTAF	10.12	0.55	3.24	1.16	2.67	8.57	0.50	3.38	0.98	5.55
mTAF	10.11	0.26	2.22	0.59	2.94	8.55	0.25	2.60	0.57	6.74
copula	9.75	0.20	1.23	0.45	2.22	9.75	0.19	1.43	0.46	3.49

to understand the behavior and benefits of mTAF and gTAF in comparison to other flows models. In Section 4.2, we demonstrate how we can exhibit the heavy-tailed behavior of the model to sample new extremes in weather and climate example. We provide a PyTorch implementation and the code for all experiments, which can be accessed through our public git repository[8].

4.1. Synthetic Experiments

In this series of experiments, we compare the performance of 4 different NSFs: vanilla flow (i.e. $\mathbf{z} \sim \mathcal{N}(0, I)$), TAF (i.e. $\mathbf{z} \sim t_{\tilde{\nu}}(0, I)$), gTAF, and mTAF with fixed degrees of freedom. The data is generated by a 8-dimensional Gaussian copula with $d_h \in \{1, 4\}$ heavy-tailed marginals with tail index $\nu = 2$. Details about the data generation can be found in Section C.2. As an oracle baseline, we fit a Gaussian copula to the data. To measure the overall fit of the model, we track the negative log-likelihood loss L. Since it is well-known that a good likelihood fit is not equivalent to high sampling quality (Theis et al., 2015), we take a closer on the samples in the tails of the distribution by considering the following three metrics.[9]

1. **Tail Value at Risk** (tVaR_α), also known as *expected shortfall*, which is the expectation of the quantile function given that we consider a quantile level larger than α. By calculating the absolute difference between tVaR_α based on the data distribution and based on synthetic samples generated by the flow, we obtain the tVaR-difference.

2. **Area under log-log plot** Area, which can be interpreted as a reweighted version of tVaR that puts more weight on the extremes.

3. **Synthetic Tail Estimates** are the marginal tail estimators based on synthetic data generated by the flow. Ideally, we expect the NF to generate samples according to the true tail-index.

We measure tVaR-differences componentwise and average over all heavy-tailed and light-tailed components to obtain tVaR_h and tVaR_l, respectively. The same applies for Area.

We fit each model 25 times to 3 different synthetic distributions and summarize the numeric results in Table 1[10]. It is no surprise that vanilla performs well for light-tailed components but suffers on capturing the tails of the heavy-tailed distributions. In the setting $d_h = 1$, we observe the same behavior for TAF, which could be attributed to having only one joint tail parameter ν to model the seven light-tailed and one heavy-tailed marginal. mTAF does not always perform best but it manages to find a good balance between the fit on the light-tailed, as well as heavy-tailed components. This is demonstrated more clearly when considering the tail estimation indices of the marginals (Figure 2). In this figure, we summarize the estimated marginal tail behavior of the learned model in a confusion matrix. While most of the generated marginals are classified as light-tailed for vanilla and TAF, mTAF is able to recover the marginal tail behavior almost perfectly. We make similar observations in the other settings and when using MAFs instead of NSFs (Section C.3). We extend our analysis by providing a 50-dimensional example in Section C.3.

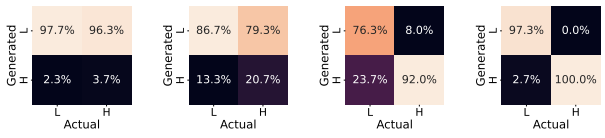


Figure 2. Marginal tail estimation based on synthetic flow samples for $d_h = 4$, $\nu = 2$ of vanilla, TAF, gTAF, and mTAF (from left to right). We classify marginals whose tail estimator is less than 10 as heavy-tailed, otherwise it is classified as light-tailed.

[8] https://github.com/MikeLasz/marginalTailAdaptiveFlow

[9] All metrics are formally defined in Section C.3.

[10] Standard deviations for one synthetic distribution are shown in Table 3.

4.2. Modeling Climate Data

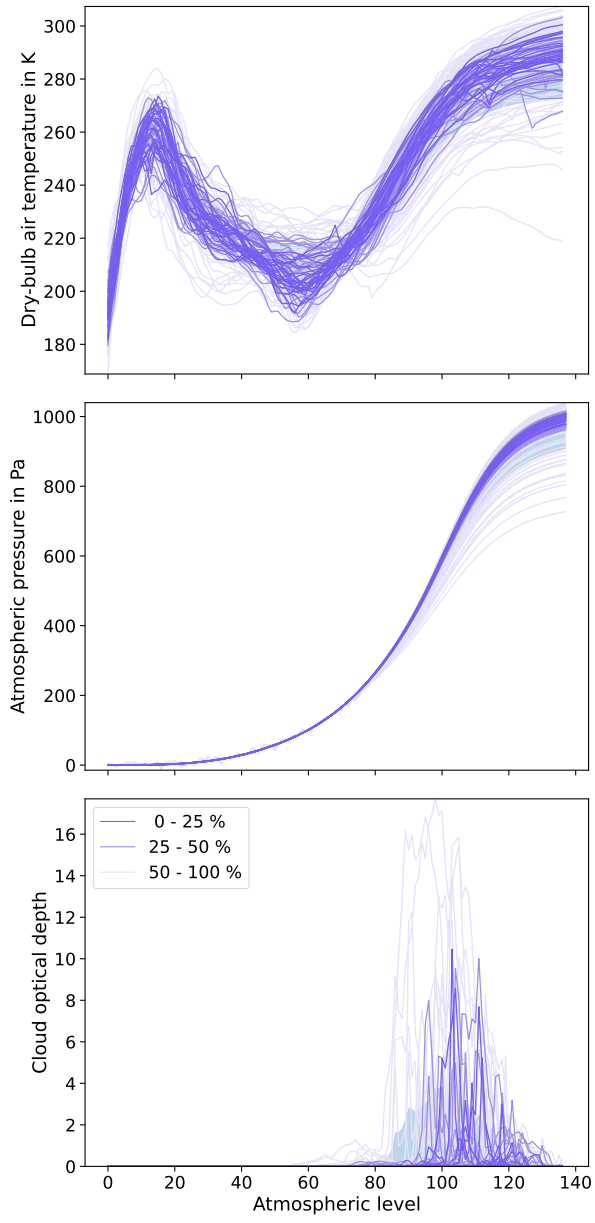


Figure 3. Synthetic flow samples using mTAF, where we clipped the lower-values of the cloud-optical depth at 0. The corresponding negative log-likelihood is -2121.46. The profiles are ordered using band depth statistics Pintado & Romo (2009) and the shaded areas represent standard deviations.

We demonstrate the benefits of the proposed methods on an example, in which tail modeling is crucial: we apply mTAF to generate new data following the distribution of the EUMETSAT Numerical Weather Prediction Satellite Application Facility (NWP-SAF) dataset (Eresmaa & McNally, 2014). The data set consists of 25 000 measurements of 3 meteorological quantities, measured on different atmospheric levels. Considering each measurement in each atmospheric level, we obtain a 412-dimensional dataset, which we visualize in Figure 9 in the Appendix. We train a vanilla, TAF, gTAF, and mTAF model using NSFs to fit the weather data. All architectural details are listed in Section C.4.

Qualitatively, all generated profile lines appear reasonable, see Figure 3 and Section C.4. From a quantitative perspective, we observe that mTAF achieves the smallest negative log-likelihood loss, as displayed in Table 2. We extend the quantitative analysis by investigating 1-dimensional random projections of the data and flow samples[11]. To do so, we follow the same procedure as Meyer et al. (2021), i.e. we generate random weights $w_1, \dots, w_{100} \sim U([0,1]^{412})$ and generate 100 random 1-dimensional data sets $X^{(j)}_{\text{flow}} := \{\langle w_j, T(z_i)\rangle : \ z_i \sim \boldsymbol{z}\}$ and $X^{(j)}_{\text{data}} := \{\langle w_j, x_i\rangle : \ x_i \in X\}$. For each j, we can compare statistics such as the means, standard deviations, and quantiles of $X^{(j)}_{\text{flow}}$ and $X^{(j)}_{\text{data}}$, which we visualize in Figure 4. While all methods are able to fit the mean very well, only mTAF generates data that obeys the same standard deviation and extreme quantiles.

More details and further comparisons are provided in Section C.4.

5. Limitations and Extensions

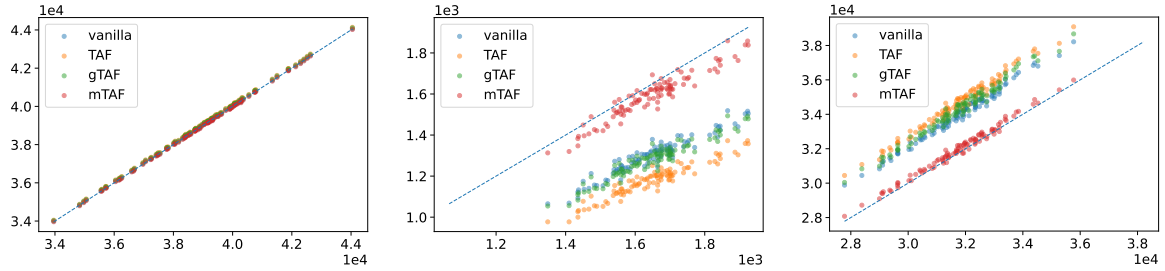
Our theoretical findings from Section 3 provide a solid foundation on learning heavy-tailed generative models. Nonetheless, there exist limitations that are worth being addressed in future works.

Asymmetric Tail Behavior is a widespread property of real distributions in which only one of the tails (lower or upper tail) is heavy-tailed whereas the other is light-tailed. For instance the cloud optical depth in the climate example (Section 4.2) has just heavy upper tails since cloud-optical depth cannot drop below 0. One way to potentially solve this issue would be to allow for even more flexible base distributions using composite models (see for instance Abu Bakar et al. (2015) and the references therein). Recently, COMET Flows (McDonald et al., 2022) employed those composite models to learn the lower tail, the upper tail, and the body of each marginal separately to improve on the performance of a variant of copula flows (Wiese et al.,

[11] Note that in contrast to the synthetic experiments, this data has much more dimensions. Quantities such as the Area under log-log plot depend on the rare tail-samples, and hence, its estimation requires many samples. This requirement is further emphasized in the high-dimensional setting, which is why we resort to these 1-dimensional projections.

Table 2. Average test loss on the NWP-SAF dataset. We average over 25 trials per model and show the standard deviations in brackets.

	vanilla	TAF	gTAF	mTAF
L	$-2101.91(\pm 9.44)$	$-2110.56(\pm 7.87)$	$-2113.48(\pm 7.93)$	$-2121.38(\pm 10.91)$



(a) Mean. (b) Standard Deviation (c) 1%-quantile

Figure 4. Statistics from 1-dimensional random projections of the data and flow samples. Each point represents the statistic $(S(X_{\text{data}(j)}), S(X_{\text{flow}(j)}))$, which are calculated using the random weight $w_j \sim U([0,1]^{412})$. In the ideal case, all points lie on the dotted diagonal line.

2019).

Tail Dependencies are a popular property investigated in areas such as financial risk analysis, and which quantify the dependency of two marginals given that a tail event occurred. More precisely, we define the tail dependency between the marginals $\mathbf{x}_i$, $\mathbf{x}_j$ as

$$\lambda_{i,j} = \lim_{u \to 1^-} \mathbb{P}\big(x_j > F_j^{-1}(u) | x_i > F_i^{-1}(u)\big) \ ,$$

where F_j^{-1} and F_i^{-1} are the quantile functions of $\mathbf{x}_j$ and $\mathbf{x}_i$, respectively. There is a large body of theoretical works revolving around copulae to model tail dependencies (Joe, 2014). Hence, a possible extension might be to replace the mean field assumption in the base distribution by a copula distribution, i.e. by dependent marginals with PDF

$$q(z) = c\big(F_1(z_1), \dots, F_D(z_D)\big) \prod_{j=1}^{D} q_j(z_j) \ ,$$

where F_j are the CDFs and q_j are the marginal PDFs of $\mathbf{x}_j$ for each j.

6. Conclusion

In this work, we deepen the mathematical understanding of the tail behavior of autoregressive normalizing flows. We note that the distribution we want to model may have heavy- as well as light-tailed marginals, and we prove that standard normalizing flows are not able to learn such distributions. Our developed theory shows how the marginal tail behavior of the target distribution of the flow relates to the tail behavior of its base distribution. Based on these theoretical findings we propose a new algorithm, which we refer to as mTAF. In particular, we initialize a base distribution based on statistical tail estimates of the target, and employ structured linearities that guarantee the correct tail behavior of the target distribution. Importantly, we extend our theory to Neural Spline Flows and provide a modification that casts them marginally tail-adaptive. Lastly, as a trade-off between the theoretically founded mTAF and the less restricted linearities in standard normalizing flows, we present a more flexible relaxation of mTAF called gTAF, which is able to converge to mTAF as a special case. An in-depth empirical analysis with heavy- and light-tailed marginals shows that mTAF and gTAF are superior in terms of estimation performance, especially when it comes to learning a heavy tail of a distribution. In contrast to standard normalizing flows, only gTAF and mTAF are able to reliably generate samples that follow the desired tail behavior.

In summary, we believe that the theoretical soundness and the ability to faithfully generate extreme samples is a major strength of gTAF and mTAF, which are both crucial properties in various applications such as finance, climate and related areas.

Acknowledgements

This work was supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC- 2092 CASA – 390781972. We also thank the anonymous reviewers for their careful reading and their useful suggestions.

References

Abu Bakar, S., Hamzah, N., Maghsoudi, M., and Nadarajah, S. Modeling loss data using composite models. *Insurance: Mathematics and Economics*, 61:146–154, 2015.

Alexanderson, S. and Henter, G. E. Robust model training and generalisation with studentising flows. In *ICML Workshop on Invertible Neural Networks, Normalizing Flows, and Explicit Likelihood Models*, 2020.

Behrmann, J., Grathwohl, W., Chen, R. T., Duvenaud, D., and Jacobsen, J.-H. Invertible residual networks. In *International Conference on Machine Learning*, pp. 573–582. PMLR, 2019.

Behrmann, J., Vicol, P., Wang, K.-C., Grosse, R., and Jacobsen, J.-H. Understanding and mitigating exploding inverses in invertible neural networks. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130, pp. 1792–1800, Virtual Event, 2021.

Chen, R. T., Rubanova, Y., Bettencourt, J., and Duvenaud, D. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, volume 31, Montreal, Canada, 2018.

Chen, R. T. Q., Behrmann, J., Duvenaud, D. K., and Jacobsen, J.-H. Residual flows for invertible generative modeling. In *Advances in Neural Information Processing Systems*, volume 32, pp. 9913–9923, Vancouver, BC, Canada, 2019.

Csorgo, S., Deheuvels, P., and Mason, D. Kernel Estimates of the Tail Index of a Distribution. *The Annals of Statistics*, 13(3):1050 – 1077, 1985.

Danielsson, J., de Haan, L., Peng, L., and de Vries, C. G. Using a bootstrap method to choose the sample fraction in tail index estimation. *Journal of Multivariate Analysis*, 76(2):226–248, 2001.

Dekkers, A. L. M., Einmahl, J. H. J., and Haan, L. D. A moment estimator for the index of an extreme-value distribution. *The Annals of Statistics*, 17(4):1833–1855, 1989.

Dinh, L., Krueger, D., and Bengio, Y. NICE: non-linear independent components estimation. In *3rd International Conference on Learning Representations, ICLR Workshop Track Proceedings*, San Diego, CA, USA, 2015.

Draisma, G., de Haan, L., Peng, L., and Pereira, T. T. A bootstrap-based method to achieve optimality in estimating the extreme-value index. *Extremes*, 2(4):367–404, 1999.

Durkan, C., Bekasov, A., Murray, I., and Papamakarios, G. Neural spline flows. *Advances in Neural Information Processing Systems*, 32:7511–7522, 2019.

Embrechts, P., Klüppelberg, C., and Mikosch, T. *Modelling extremal events: for insurance and finance*, volume 33. Springer Science & Business Media, 2013.

Eresmaa, R. and McNally, A. Diverse profile datasets from the ecmwf 137-level short-range forecasts. 10 2014. doi: 10.13140/2.1.4476.8963.

Gabaix, X., Gopikrishnan, P., Plerou, V., and Stanley, H. E. A theory of power-law distributions in financial market fluctuations. *Nature*, 423(6937):267–270, 2003.

Gallier, J. Schur complements and applications. 05 2011. doi: 10.1007/978-1-4419-9961-0_16.

Grathwohl, W., Chen, R. T. Q., Bettencourt, J., Sutskever, I., and Duvenaud, D. Ffjord: Free-form continuous dynamics for scalable reversible generative models. In *International Conference on Learning Representations*, New Orleans, LA, USA, 2019.

Groeneboom, P., Lopuhaä, H. P., and de Wolf, P.-P. Kernel-type estimators for the extreme value index. *The Annals of Statistics*, 31(6):1956 – 1995, 2003.

Hill, B. M. A simple general approach to inference about the tail of a distribution. *The annals of statistics*, pp. 1163–1174, 1975.

Hofert, M., Kojadinovic, I., Maechler, M., and Yan, J. *Elements of Copula Modeling with R*. Springer Use R! Series, 2018.

(https://math.stackexchange.com/users/491644/maxim), M. Asymptotics of inverse of normal cdf. Mathematics Stack Exchange. URL:https://math.stackexchange.com/q/2966269 (version: 2018-10-22).

Huang, C.-W., Krueger, D., Lacoste, A., and Courville, A. Neural autoregressive flows. In *International Conference on Machine Learning*, pp. 2078–2087. PMLR, 2018.

Jacobsen, J.-H., Smeulders, A., and Oyallon, E. i-revnet: Deep invertible networks. In *International Conference on Learning Representations*, Vancouver, Canada, 2018.

Jaini, P., Kobyzev, I., Yu, Y., and Brubaker, M. Tails of Lipschitz triangular flows. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pp. 4673–4681, Virtual Event, 2020.

Joe, H. *Dependence modeling with copulas*. CRC press, 2014.

Katz, R. W. Do weather or climate variables and their impacts have heavy-tailed distributions? In *Climate Variations and Forecasting (Joint with the 16th Conference Probability and Statistics and the 13th Symposium on Global Change and Climate Variations)*, 2002.

Kingma, D. P. and Dhariwal, P. Glow: Generative flow with invertible 1x1 convolutions. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Kingma, D. P., Salimans, T., Jozefowicz, R., Chen, X., Sutskever, I., and Welling, M. Improved variational inference with inverse autoregressive flow. *Advances in neural information processing systems*, 29:4743–4751, 2016.

Kirkby, J. L., Nguyen, D. H., and Nguyen, D. Moments of student's t-distribution: A unified approach. *Computation Theory eJournal*, 2019.

Kobyzev, I., Prince, S., and Brubaker, M. Normalizing flows: An introduction and review of current methods. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*. IEEE, 2020.

Koonin, E., Wolf, Y., and Karev, G. *Power Laws, Scale-Free Networks and Genome Biology*. 01 2006.

Laszkiewicz, M., Lederer, J., and Fischer, A. Copula-based normalizing flows. In *ICML Workshop on Invertible Neural Networks, Normalizing Flows, and Explicit Likelihood Models*, 2021.

Liu, J., Lipschutz, S., and Spiegel, M. *Schaum's Outline of Mathematical Handbook of Formulas and Tables, 4th Edition*. McGraw-Hill, 2012.

McDonald, A., Tan, P.-N., and Luo, L. Comet flows: Towards generative modeling of multivariate extremes and tail dependence. *arXiv preprint arXiv:2205.01224*, 2022.

McNeil, A. J., Frey, R., and Embrechts, P. *Quantitative risk management: concepts, techniques and tools-revised edition*. Princeton university press, 2015.

Meyer, D., Nagler, T., and Hogan, R. J. Copula-based synthetic data generation for machine learning emulators in weather and climate: application to a simple radiation model. *Geoscientific Model Development Discussions*, pp. 1–21, 2021.

Nair, J., Wierman, A., and Zwart, B. The fundamentals of heavy-tails: Properties, emergence, and identification. In *Proceedings of the ACM SIGMETRICS/International Conference on Measurement and Modeling of Computer Systems*, pp. 387–388, New York, NY, USA, 2013.

Oliva, J., Dubey, A., Zaheer, M., Poczos, B., Salakhutdinov, R., Xing, E., and Schneider, J. Transformation autoregressive networks. In *International Conference on Machine Learning*, pp. 3898–3907. PMLR, 2018.

Papamakarios, G., Pavlakou, T., and Murray, I. Masked autoregressive flow for density estimation. In *Advances in Neural Information Processing Systems*, volume 30, pp. 2338–2347, Long Beach, CA, USA, 2017.

Patki, N., Wedge, R., and Veeramachaneni, K. The synthetic data vault. In *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 399–410, Oct 2016. doi: 10.1109/DSAA.2016.49.

Pintado, S. and Romo, J. On the concept of depth for functional data. *Journal of the American Statistical Association*, 104, 06 2009. doi: 10.1198/jasa.2009.0108.

Qi, Y. Bootstrap and empirical likelihood methods in extremes. *Extremes*, 11:81–97, 03 2008.

Resnick, S. On the foundations of multivariate heavy-tail analysis. *Journal of Applied Probability*, 41:191–212, 2004.

Rezende, D. and Mohamed, S. Variational inference with normalizing flows. In *International conference on machine learning*, pp. 1530–1538. PMLR, 2015.

Rippel, O. and Adams, R. P. High-dimensional probability estimation with deep density models. *CoRR*, abs/1302.5125, 2013.

Tabak, E. G. and Turner, C. V. A family of nonparametric density estimation algorithms. *Communications on Pure and Applied Mathematics*, 66(2):145–164, 2013.

Theis, L., Oord, A. v. d., and Bethge, M. A note on the evaluation of generative models. *arXiv preprint arXiv:1511.01844*, 2015.

Van der Vaart, A. W. *Asymptotic statistics*, volume 3. Cambridge university press, 2000.

Voitalov, I., van der Hoorn, P., van der Hofstad, R., and Krioukov, D. Scale-free networks well done. *Phys. Rev. Research*, 1:033034, Oct 2019.

Wiese, M., Knobloch, R., and Korn, R. Copula & marginal flows: Disentangling the marginal from its joint. *arXiv preprint arXiv:1907.03361*, 2019.

A. Theory and Proofs

In this Section, we derive and prove our theoretical results. We start by presenting some preliminary results in Section A.1, which help us providing the proof of our main result in Section A.2. In Section A.3, we illuminate the technical Assumption A.6 and provide examples and intuitions.

A.1. Preliminary theoretical Results

Proposition A.1. *Let* $\mathbf{x} \in \mathbb{R}$ *be a random variable. Then, it holds that* $\mathbf{x}$ *has tail index less or equal than* α *iff. for any* $\beta > \alpha$ *and any* $C > 0$ *it is*

$$\int_{|x|>C} |x|^\beta p(x)dx = \infty \ .$$

Proof. Let us first assume that $\mathbf{x}$ has tail index at most α. Then, according to Definition 2.2, we know that for any $\beta > \alpha$ and $C > 0$

$$\begin{aligned}\infty = \mathbb{E}_{\mathbf{x}}[|\mathbf{x}|^\beta] = \int_{\mathbb{R}} |x|^\beta p(x)dx &= \int_{|x|\leq C} |x|^\beta p(x)dx + \int_{|x|>C} |x|^\beta p(x)dx \\ &\leq C^\beta \int_{|x|\leq C} p(x)dx + \int_{|x|>C} |x|^\beta p(x)dx \ .\end{aligned}$$

Since we assume p to be continuous, we can bound p on the compact interval $[-C, C]$, and hence, the first above integral must be bounded. Therefore, it is

$$\int_{|x|>C} |x|^\beta p(x)dx = \infty \ .$$

To prove the back-direction, let us consider a $\beta > \alpha$ and $C > 0$. Very similarly to the forward-proof, we can now see that

$$\begin{aligned}\infty &= \int_{|x|>C} |x|^\beta p(x)dx \\ &= C^\beta \int_{|x|\leq C} p(x)dx + \int_{|x|>C} |x|^\beta p(x)dx \\ &= \mathbb{E}_{\mathbf{x}}\left[|\mathbf{x}|^\beta\right] \ ,\end{aligned}$$

where the second equality follows from the finiteness of the integral. Since this follows for any $\beta > \alpha$, $\mathbf{x}$ must have tail index α or less.

□

This simple result demonstrates that the tail index, as indicated by the name, depends on the tail of the distribution, i.e. on the PDFs behavior for large values $|x| > C$. This fact motivates why maximum likelihood estimations of the tail index, which depend on the whole distribution are biased. For more elaborate details, we refer to Section 9 in Nair et al. (2013).

In a similar fashion to the previous result, the next technical lemma states that unboundedness of the moment-generating function is due to the unboundedness of the integrand for tail events, i.e. for $z > z^*$. This little lemma turns out to be useful in the proof of Proposition 3.3.

Lemma A.2. *Let* $\mathbf{z} \in \mathbb{R}$ *be heavy-tailed. Then it holds for any* $z^* \in \mathbb{R}$ *and* $\lambda > 0$ *that*

$$\int_{z>z^*} e^{\lambda z} p_{\mathbf{z}}(z)dz = \infty \ .$$

Proof. Since $\mathbf{z} \in \mathbb{R}$ is heavy-tailed, we know that for all $\lambda > 0$

$$\begin{aligned}\infty = m_{\mathbf{z}}(\lambda) &= \int_{\mathbb{R}} e^{\lambda z} p_{\mathbf{z}}(z) dz \\ &= \int_{z \leq z^*} e^{\lambda z} p_{\mathbf{z}}(z) dz + \int_{z > z^*} e^{\lambda z} p_{\mathbf{z}}(z) dz \\ &\leq F_{\mathbf{z}}(z^*) e^{\lambda z^*} + \int_{z > z^*} e^{\lambda z} p_{\mathbf{z}}(z) dz \ ,\end{aligned}$$

where $F_{\mathbf{z}}$ is the CDF of $\mathbf{z}$. The last inequality follows from the fact that $\exp(\lambda z)$ is monotonic increasing in z. Since the first summand is bounded, it follows that the second summand must be unbounded. This completes the proof. □

Recall that in Proposition 3.2 we state that j-heavy-tailedness induces ℓ_2-heavy-tailedness. In the following, we provide a formal proof of this result.

Proof of Proposition 3.2. For this proof, we employ the equivalent definition of heavy-tailedness of $\mathbf{x}_j$ via the decay rate of its distribution function (see e.g. Lemma 1.1. in Nair et al. (2013)), i.e.

$$\limsup_{x_j \to \infty} \frac{1 - F_j(x_j)}{e^{-\lambda x_j}} = \infty \quad \text{for all } \lambda > 0 \ , \tag{6}$$

where F_j is the CDF of $\mathbf{x}_j$. Since $x_j \leq \|x\|$ for all $x \in \mathbb{R}^D$, we can conclude that $F_j(a) \geq F_{\|x\|}(a)$ for $a \in \mathbb{R}$. Therefore,

$$\frac{1 - F_{\|\mathbf{x}\|}(a)}{e^{-\lambda a}} \geq \frac{1 - F_{x_j}(a)}{e^{-\lambda a}} \to \infty \quad \text{for } a \to \infty \ .$$

According to the equivalent definition in (6), $\|\mathbf{x}\|$ is heavy-tailed, which proves that $\mathbf{x}$ is ℓ_2-heavy-tailed. □

The following is a well-known implication of the change of variables formula and the integration rule by substitution, which we are going to apply in the subsequent proofs.

Lemma A.3 (Substitution in the Moment-Generating Function)**.** *Let T be a diffeomorphism such that $T(\mathbf{z}) = \mathbf{x}$ for some random variables $\mathbf{x}, \mathbf{z} \in \mathbb{R}^D$. Then, we can rewrite*

$$\int_{\mathbb{R}^D} e^{\lambda x} p(x) dx = \int_{\mathbb{R}^D} e^{\lambda T(z)} q(z) dz \ .$$

For completeness, we give a brief proof of this result.

Proof. Using the change of variables formula, see (1), we can write

$$\int_{\mathbb{R}^D} e^{\lambda x} p(x) dx = \int_{\mathbb{R}^D} e^{\lambda x} q\big(T^{-1}(x)\big) \big|\det J_{T^{-1}(x)}\big| dx \ .$$

Now, we can rewrite $\exp(\lambda x) = \exp(\lambda T\big(T^{-1}(x)\big)$ and substitute $z = T^{-1}$. Integration by substitution completes the proof. □

Next, we present how we can use copulae to reformulate a multivariate PDF. [Copula] A copula is a multivariate distribution with cumulative distribution function (CDF) $C : [0,1]^D \to [0,1]$ that has standard uniform marginals, i.e. the marginals C_j of C satisfy $C_j \sim U[0,1]$.

Theorem A.4 (Sklar's Theorem)**.** *Taken from Hofert et al. (2018).*

1. *For any D-dimensional CDF F with marginal CDFs $F_1, \ldots, F_D$, there exists a copula C such that*

$$F(z) = C\big(F_1(z_1), \ldots, F_D(z_D)\big) \tag{7}$$

for all $z \in \mathbb{R}^D$. The copula is uniquely defined on $\mathcal{U} := \prod_{j=1}^{D} \mathrm{Im}(F_j)$, where $\mathrm{Im}(F_j)$ is the image of F_j. For all $u \in \mathcal{U}$ it is given by

$$C(u) = F\big(F_1^{\leftarrow}(u_1), \dots, F_D^{\leftarrow}(u_D)\big) \ ,$$

where $F_j^{\leftarrow}$ are the right-inverses of F_j.

2. *Conversely, given any D-dimensional copula C and marginal CDFs $F_1, \dots F_D$, a function F as defined in (7) is a D-dimensional CDF with marginals $F_1, \dots, F_D$.*

Therefore, if F is absolutely continuous, we can differentiate (7) to obtain the PDF of $\mathbf{z}$

$$q(z) = c\big(F_1(z_1), \dots, F_D(z_D)\big) \prod_{j=1}^{D} q_j(z_j) \ ,$$

where c denotes the PDF of the copula C.

Lastly, we present the following asymptotic behavior of the inverse CDF of a standard Gaussian distribution, which we use in Section A.3 to explain Assumption A.6.

Lemma A.5 (Asymptotic Behavior[12] of $\Phi^{-1}(1-y)$)**.** *Denote by Φ the CDF of a standard Gaussian distribution. Then, it holds for the inverse of Φ that*

$$\Phi^{-1}(1-y) \sim \sqrt{-2\log(y)} \quad \textit{for } y \to 0 \ .$$

Proof. First, we note that

$$\Phi(x) = \frac{1}{2} + \frac{1}{2}\,\mathrm{erf}\left(\frac{x}{\sqrt{2}}\right) \sim 1 - \frac{1}{x\sqrt{2}} e^{-x^2/2} \ ,$$

which is a well-known asymptotic (Liu et al., 2012). Here, erf denotes the error-function. Rearranging terms gives

$$\log\big(1-\Phi(x)\big) \sim -\log\Big(x\sqrt{2\pi}\Big) - \frac{x^2}{2} \sim -\frac{x^2}{2} \quad \text{as } x \to \infty \ .$$

Finally, we can invert the above asymptotic equation to obtain

$$\Phi^{-1}(y) = \sqrt{-2\log(1-y)} \quad \text{for } y \to 1$$

or equivalently

$$\Phi^{-1}(1-y) = \sqrt{-2\log(y)} \quad \text{for } y \to 0 \ .$$

□

A.2. Proof of the Main Result

The proof of Proposition 3.3 relies on lower-bounding the moment-generating function of each marginal $\mathbf{x}_j$. In order to derive such a bound of a multivariate integral, we rewrite the joint distributions $q_{\leq j}$ using their copula densities:

$$q_{\leq j}(z_{\leq j}) = c_j\big(F_1(z_1), \dots, F_j(z_j)\big) \prod_{i<j} q_i(z_i)$$

for any $j \in \{1, \dots, D\}$ and for corresponding copula density c_j. Our proof relies on the following technical condition on the decay rate of the copula densities.

Assumption A.6 (Bounding the Marginal Decay Rate of the Copula Densities)**.** For all $j \in \{1, \dots, D\}$ and $\lambda > 0$ there exists a compact set $\mathbb{S} \subset \mathbb{R}^{j-1}$ with positive (Lebesgue-)mass, a constant $z_j^* > 0$, a scaling constant $s > 0$, and a function $f(z_{<j}) < \lambda\sigma(z_{<j})$ for $z_{<j} \in \mathbb{S}$ such that

$$c_j\big(F_1(z_1), \dots, F_j(z_j)\big) \geq s e^{-f(z_{<j})z_j} \quad \text{for } z_j > z_j^* \text{ and } z_{<j} \in \mathbb{S} \ , \tag{8}$$

where c_j is the copula density of $q_{\leq j}$

[12] The idea of the proof is due to (https://math.stackexchange.com/users/491644/maxim)

This assumption sets a bound on the decay rate of the copula density with respect to z_j. We clarify this assumption in Section A.3 with additional examples.

Now, we set all preliminaries to prove Proposition 3.3.

Proof of Proposition 3.3. We start by considering the case $j = 1$. In this case it is $\mathbf{x}_1 = \mu + \sigma \mathbf{z}_1$ and therefore

$$\begin{aligned} m_{\mathbf{x}_1}(\lambda) &= \int_{\mathbb{R}} e^{\lambda x_1} p_1(x_1) dx_1 \\ &= \int_{\mathbb{R}} e^{\lambda(\mu_1 + \sigma_1 z_1)} q_1(z_1) dz_1 \quad \text{(Lemma A.3)} \\ &= e^{\lambda \mu_1} \int_{\mathbb{R}} e^{\lambda \sigma_1 z_1} q_1(z_1) dz_1 \ . \end{aligned}$$

Defining $\lambda' := \lambda \sigma_1 > 0$, we can see that the last integral is unbounded due to the heavy-tailedness of $\mathbf{z}_1$, see Definition 2.1. Therefore, $m_{\mathbf{x}_1}(\lambda) = \infty$ for all $\lambda > 0$, which proves the heavy-tailedness of $\mathbf{x}_1$.
Next, we consider the case $j > 1$. Again, we examine the moment-generating function of $\mathbf{x}_j$. Define the jth canonical basis vector $v_j := (0, \dots, 0, 1, 0, \dots, 0)^\top$. Then,[13]

$$\begin{aligned} m_{\mathbf{x}_j}(\lambda) = m_{v_j^\top \mathbf{x}} &= \int_{\mathbb{R}^D} e^{\lambda v_j^\top x} p(x) dx \quad \text{(LOTUS)} \\ &= \int_{\mathbb{R}^D} e^{\lambda T_j(z_j, z_{<j})} q(z) dz \quad \text{(Lemma A.3)} \\ &= \int_{\mathbb{R}^j} e^{\lambda \mu(z_{<j}) + \lambda \sigma(z_{<j}) z_j} q_{\leq j}(z_{\leq j}) dz_{\leq j} \\ &= \int_{\mathbb{R}^{j-1}} e^{\lambda \mu(z_{<j})} \int_{\mathbb{R}} e^{\lambda \sigma(z_{<j}) z_j} q_{\leq j}(z_{\leq j}) dz_j dz_{<j} \ . \end{aligned} \tag{9}$$

Using Sklar's Theorem (Theorem A.4), we can write any joint PDF as the product of marginals and a copula density c_j such that

$$q_{\leq j}(z_{\leq j}) = c_j\big(F_1(z_1), \dots, F_j(z_j)\big) \prod_{i<j} q_i(z_i) \ . \tag{10}$$

We plug (10) into (9) to obtain

$$\begin{aligned} m_{\mathbf{x}_j}(\lambda) &= \int_{\mathbb{R}^{j-1}} e^{\lambda \mu(z_{<j})} q_{<j}(z_{<j}) \int_{\mathbb{R}} e^{\lambda \sigma(z_{<j}) z_j} c_j\big(F_1(z_1), \dots, F_j(z_j)\big) q_j(z_j) dz_j dz_{<j} \\ &\geq \int_{\mathbb{S}} e^{\lambda \mu(z_{<j})} q_{<j}(z_{<j}) \int_{z_j > z_j^*} e^{\lambda \sigma(z_{<j}) z_j} c_j\big(F_1(z_1), \dots, F_j(z_j)\big) q_j(z_j) dz_j dz_{<j} \ , \end{aligned} \tag{11}$$

since all quantities within the integral are positive. Using Assumption A.6, we can bound the inner integral of the above equation, which we denote by $A(z_{<j})$, and get

$$\begin{aligned} A(z_{<j}) &\geq s \int_{z_j > z_j^*} e^{(\lambda \sigma(z_{<j}) - f(z_{<j})) z_j} q_j(z_j) dz_j \\ &= s \int_{z_j > z_j^*} e^{\lambda' z_j} q_j(z_j) dz_j \quad (\text{define } \lambda' := \lambda - \sigma(z_{<j}) - f(z_{<j}) \ ) \\ &= \infty \quad \text{for all } z_{<j} \in \mathbb{S} \ , \end{aligned}$$

due to the heavy-tailedness of $\mathbf{z}_j$ and Lemma A.2. Since $\mathbb{S}$ is compact, μ and q are both continuous, and q is positive, we deduce that $\exp(\lambda \mu(z_{<j})) q_{<j}(z_{<j})$ is lower-bounded (by a constant larger than 0) in $\mathbb{S}$. Therefore, employing (11) and using that $\mathbb{S}$ has positive mass, we can lower-bound the moment-generating function by ∞, which proves the heavy-tailedness of $\mathbf{x}_j$. In summary, $\mathbf{x}$ is j-heavy-tailed for all $j \in \{1, \dots, D\}$. □

[13] Note that for the sake of clarity, we leave out the index j in μ_j and σ_j.

A.3. Notes on Assumption A.6

Assumption 1 might look troublesome at first sight, but we will illustrate in this section that the condition is indeed very reasonable. We will show how to verify it in simple examples, and we will introduce a simpler, more intuitive sufficient condition for it.

First of all, let us present a restricted but more intuitive version of Assumption A.6.

Assumption A.7 (Simplification of Assumption A.6)**.** For all $j \in \{1, \dots, D\}$ and $\lambda > 0$ it holds for $\mathbb{S} := [a, b]^{j-1}$ that there exist constants z_j^* and $s > 0$ such that

$$c\big(F_1(z_1), \dots, F_j(z_j)\big) \geq s e^{-(\lambda_\sigma - \varepsilon) z_j} \quad \text{for } z_j > z_j^* \text{ and } z_{<j} \in \mathbb{S} \ , \tag{12}$$

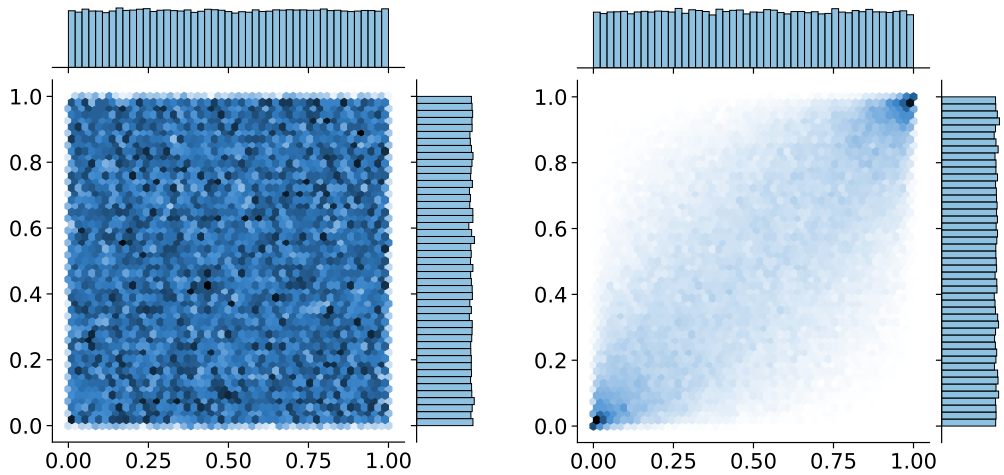
where c_j is the copula density of $q_{\leq j}$, λ_σ is a lower bound of $\lambda\sigma(z_{<j})$, $\varepsilon > 0$ is small such that $\lambda_\sigma - \varepsilon > 0$.

Let us summarize the simplifications that we make in Assumption A.7. First of all, we restricted $\mathbb{S}$ to be a closed cube $[a, b]^{j-1}$, which is obviously a specific instant of a compact set with positive mass. Further, we assumed σ to be continuous, and thus, $\lambda\sigma(z_{<j})$ must be lower-bounded in $\mathbb{S}$. This allows us to replace the function f by the constant $\lambda_\sigma - \varepsilon$ for arbitrary small $\varepsilon > 0$.

After giving this simplified sufficient condition, we provide some intuition by presenting some examples where Assumption A.7 holds true. [Independent Variables] Consider a random variable $\mathbf{z}$ with independent components, i.e. $q(z) = \prod_{j=1}^{D} q_j(z_j)$. Then, the associated copula is the independence copula (Figure 5), which is a uniform random distribution on $[0, 1]^D$. Therefore it is $c\big(F_1(z_1), \dots, F_D(z_D)\big) = 1$ for all $z \in \mathbb{R}^D$ and Assumption A.7 follows immediately since $s \exp(-(\lambda_\sigma - \varepsilon) z_j) \to 0$ for $z_j \to \infty$. [Bounded Copula Density] Consider a lower-bounded copula density, i.e. there exists a lower bound $a > 0$ such that

$$c(u_1, \dots, u_D) \geq a \quad \text{for all } u \in [0, 1]^D \ .$$

Again, the validity of Assumption A.7 in this setting is clear. Furthermore, this assumption is obviously not limited to



(a) Independence copula: $c(z_1, z_2) = 1$. (b) Gaussian copula with correlation $\rho_{12} = 0.7$.

Figure 5. Two 2-dimensional Copula densities.

bounded copula densities but also holds for copula densities that converge to 0 but whose decay rate in z_j is lower-bounded by (12). To visualize the intuition, consider the 2-dimensional copula density of a Gaussian copula in Figure 5. Imagine fixing $\mathbb{S}$ such that $F_1(z_1) \in [0.5, 0.75]$, which is compact for continuous F_1. Then (12) bounds the decay rate within the "tube" $[0.5, 0.75]$ if we consider $z_2 \to \infty$, i.e. if $F_2(z_2) \to 1$. Next, we show how we can formally prove the assumption for

Gaussian copulae. Recall, the Gaussian copula with correlation matrix $R \in \mathbb{R}^{D\times D}$ has density function

$$c(u) = \frac{1}{\sqrt{\det R}} \exp\left(-\frac{1}{2}\big(\Phi^{-1}(u_1), \dots \Phi^{-1}(u_D)\big)\big(R^{-1} - I\big)\big(\Phi^{-1}(u_1), \dots \Phi^{-1}(u_D)\big)^\top \right) \ , \tag{13}$$

where $I \in \mathbb{R}^{D\times D}$ is the identity matrix, Φ^{-1} is the inverse CDF of the univariate standard Gaussian distribution, and $u \in [0,1]^D$. In the following, we consider Assumption A.7 for $j = D$.

Note that $\mathbb{S}$ is assumed to be a compact set, therefore $F_j(z_j)$ are all upper and lower-bounded by some value for $z_{<j} \in \mathbb{S}$. This makes all polynomials of them also bounded. Hence, we can find constants a', b', c' such that we can lower-bound the term within the exponential in (13) by

$$-\frac{1}{2}\big(a'\Phi^{-1}\big(F_D(z_D)\big)^2 + b'\Phi^{-1}\big(F_D(z_D)\big) + c' \ .$$

Plugging the above into (13) gives

$$\begin{aligned} c\big(F_1(z_1), \dots, F_D(z_D)\big) &\geq \frac{1}{\sqrt{\det R}} \exp\left(-\frac{1}{2}\big(a'\Phi^{-1}\big(F_D(z_D)\big)^2 + b'\Phi^{-1}\big(F_D(z_D)\big) + c'\right) \\ &\propto \exp\left(-a\Phi^{-1}\big(F_D(z_D)\big)^2 + b\Phi^{-1}\big(F_D(z_D)\big)\right) \quad \text{(for some } a, b\text{)} \\ &\geq \exp\left(-|a|\Phi^{-1}\big(F_D(z_D)\big)^2 + |b|\Phi^{-1}\big(F_D(z_D)\big)\right) \\ &\geq \exp\left(-|a|\Phi^{-1}\big(F_D(z_D)\big)^2\right) \ , \end{aligned} \tag{14}$$

where the last line applies if $\Phi^{-1}\big(F_D(z_D)\big) \geq 0$, which is satisfied if z^* is large enough[14].

Next, we use the asymptotic relation from Lemma A.5

$$\Phi^{-1}\big(F_D(z_D)\big) \sim \sqrt{-2\log\big(1 - F_D(z_D)\big)} \ .$$

Hence, for each $\varepsilon > 0$ there exists a z^* large enough such that

$$\left| \frac{\Phi^{-1}\big(F_D(z_D)\big)}{\sqrt{-2\log\big(1 - F_D(z_D)\big)}} - 1 \right| < \varepsilon \ ,$$

which can be rearranged to

$$\Phi^{-1}\big(F_D(z_D)\big) < \sqrt{-2\log\big(1 - F_D(z_D)\big)}(1+\varepsilon) \ .$$

Plugging the above into (14), we obtain

$$\begin{aligned} c\big(F_1(z_1), \dots, F_D(z_D)\big) &\geq \exp\left(2|a|(1+\varepsilon)^2 \log\big(1 - F_D(z_D)\big)\right) \\ &= \big(1 - F_D(z_D)\big)^{2\tilde{a}} \ , \end{aligned} \tag{15}$$

where we define $\tilde{a} := 2|a|(1+\varepsilon)^2$. Hence, we are left to lower-bound (15), which we can do for a range of heavy-tailed marginal distributions such as:

1. **Pareto distribution:** The Pareto distribution with shape parameter α has CDF

$$F(z) = 1 - \frac{1}{z}^{\alpha} \ .$$

[14] for instance if z^* is larger than the median of $\mathbf{z}_D$

Therefore,

$$\begin{aligned}\big(1 - F_D(z_D)\big)^{2\tilde{a}} &= \frac{1}{z_D}^{2|a|\alpha} \\ &= \exp\big(-2\tilde{a}\alpha \log(z_D)\big) \\ &\geq \exp\big(-2\tilde{a}\alpha z_D\big)\end{aligned}$$

for $z_D \geq e$.

2. **Scale invariant distributions:** Following the same argument as above, each distribution with CDF

$$F(z) = 1 - bz^{-\alpha} \quad \text{for } z > z^* \tag{16}$$

for constants $b, \alpha, z^* > 0$ satisfies the bound

$$\big(1 - F_D(z_D)\big)^{2\tilde{a}} \geq b \exp\big(-2\tilde{a}\alpha z_D\big) \ .$$

Each distribution with CDF as in (16) is a scale-invariant distribution (see e.g. Theorem 2.1 in Nair et al. (2013)).

3. **Exponentially decaying distributions:** Every distribution that satisfies the bound

$$F(z) \leq 1 - \exp(z)^{-\alpha}$$

for $z > z^*$ and $\alpha > 0$. In this case, we can again show that Assumption A.6 is valid:

$$\big(1 - F_D(z_D)\big)^{2\tilde{a}} \geq \exp(-2\tilde{a}\alpha z_D) \ .$$

Lastly, we want to emphasize that Corollary 3.5 is derived by an iterative application of Theorem 3.4. Therefore, Assumption A.6 must hold for all "flow steps", i.e. if $T = T^{(L)} \circ \cdots \circ T^{(1)}$, we need to ensure validity of Assumption A.6 for $\mathbf{z}^{(0)} := \mathbf{z}$, $\mathbf{z}^{(1)} := T^{(1)}(\mathbf{z})$, $\mathbf{z}^{(2)} := T^{(2)} \circ T^{(1)}(\mathbf{z})$, $\ldots$, $\mathbf{z}^{(L-1)} := T^{(L-1)} \circ \cdots \circ T^{(1)}(\mathbf{z})$. In Example A.3, we show that this assumption holds true for $\mathbf{z}^{(0)}$ since we define our base distribution under the mean-field assumption. Furthermore, we conjecture that if we apply a Lipschitz-continuous diffeomorphism on a random variable with bounded copula density, then the transformed random variable must also have a bounded copula density. Hence, Assumption A.6 would be valid for all "flow steps" (see Example A.3) when starting with a mean-field base distribution $q(z) = \prod_{j=1}^{D} q_j(z_j)$. However, this conjecture needs to be studied in further research.

A.4. Proof of Theorem 3.7

Proof. We proceed in two steps: First, we show that $z_\nu \xrightarrow{\nu\to\infty}_D z$ by applying the *Cramér-Wold Theorem*. Second, we use the *Continuous-Mapping Theorem* to finish the proof. All used theorems can be found, for instance, in Van der Vaart (2000).

Step 1. By the Cramér-Wold Theorem, it is sufficient to show that for every $v \in \mathbb{R}^D$ it is

$$v^\top z_\nu = \sum_{j=1}^{D} v_j z_{\nu,j} \xrightarrow{\nu\to\infty}_D \sum_{j=1}^{D} v_j z_j = v^\top z \ .$$

We prove this convergence via the convergence of the characteristic function. Since $z_{\nu,j}$ are independent for all $j \in \{1, \ldots, D\}$, it is

$$\begin{aligned}\varphi_{v^\top z_\nu}(t) &= \varphi_{z_{\nu,1}}(v_1 t) \cdots \varphi_{z_{\nu,d_l}}(v_{d_l} t) \varphi_{z_{d_l+1}}(v_{d_l+1} t) \cdots \varphi_{z_D}(v_D t) \\ &\to \varphi_{z_1}(v_1 t) \cdots \varphi_{z_{d_l}}(v_{d_l} t) \varphi_{z_{d_l+1}}(v_{d_l+1} t) \cdots \varphi_{z_D}(v_D t) \\ &= \varphi_{v^\top z}(t) \quad \forall t \in \mathbb{R} \ ,\end{aligned}$$

where the convergence follows from the convergence $t_\nu \xrightarrow{\nu\to\infty}_D \mathcal{N}(0,1)$ and *Levy's Continuity Theorem*. The last equality follows since the marginals are independent. Using Levy's Continuity Theorem once more, we conclude that

$$v^\top z_\nu \xrightarrow{\nu\to\infty}_D v^\top z \ .$$

Hence, we can apply Cramér-Wold to obtain the convergence $z_\nu \xrightarrow{\nu\to\infty}_D z$.

Step 2. We apply the Continuous Mapping Theorem to obtain the convergence result. □

B. Data-Driven LU-Layers

One of the major reason for recent improvements of NFs is due to the generalization of the permutation layers to more general invertible linear layers (Oliva et al., 2018). One type of these general invertible linear layers are based on the LU-decomposition, which were essential building blocks for many SOTA NFs, such as in Kingma & Dhariwal (2018); Durkan et al. (2019). Therefore, it is important to address whether we can generalize our theory from Section 3.2 of the main paper to the more expressive LU-Layers. In this section we provide supplementary materials to Section 3.3 to show that we are indeed able to generalize to LU-Layers, while retaining their computational benefits. First, in Section B.1 we derive sufficient conditions under which linear layers preserve the marginal tail behaviour. In Section B.2 we present an efficient way to implement a tail-preserving LU-type invertible layer.

B.1. Marginal Tail Behavior under Linear Transformations

In this section, we assume without loss of generality that the components of $\mathbf{z}$ are ordered such that $\mathbf{z}$ is j-light-tailed for $j \in \{1, \dots, d_l\}$ and j-heavy-tailed for $j \in \{d_l + 1, \dots, D\}$. Our goal is to find conditions for a matrix W under which $\mathbf{z}$ and $W\mathbf{z}$ have equal tail-behavior.

Theorem B.1. *Let $\mathbf{z}$ be a random variable that is j-light-tailed for $j \in \{1, \dots, d_l\}$ and j-heavy-tailed for $j \in \{d_l + 1, \dots, D\}$. Further, consider a diagonal invertible block-matrix*

$$W = \begin{pmatrix} A & 0 \\ B & C \end{pmatrix} \ , \tag{17}$$

with $A \in \mathbb{R}^{d_l \times d_l}$, $B \in \mathbb{R}^{(D-d_l)\times d_l}$, $C \in \mathbb{R}^{(D-d_l)\times(D-d_l)}$ and 0 is a zero matrix of size $d_l \times (D - d_l)$. Then, it follows that $W\mathbf{z}$ and $\mathbf{z}$ have equal tail behavior.

Before heading into the proof, we introduce the following useful lemma:

Lemma B.2. *Let $0 < \lambda^*$ be a scalar such that $\mathbb{E}_{\mathbf{z}}[\exp(\lambda^* \mathbf{z})] < \infty$ for some univariate random variable $\mathbf{z}$. Then, it holds that $\mathbb{E}_{\mathbf{z}}[\exp(\lambda \mathbf{z})] < \infty$ for all $0 < \lambda \leq \lambda^*$.*

Proof. Consider $0 < \lambda \leq \lambda^*$. Then, we can split the expectation

$$\begin{aligned} \mathbb{E}_{\mathbf{z}}\left[e^{\lambda \mathbf{z}}\right] &= \int_{(-\infty,0]} q(z) e^{\lambda z} dz + \int_{(0,\infty)} q(z) e^{\lambda z} dz \\ &\leq \int_{(-\infty,0]} q(z) e^{0} dz + \int_{(0,\infty)} q(z) e^{\lambda^* z} dz \ . \end{aligned}$$

The first integral is upper-bounded by 1, and the second integral can be upper-bounded by the integral over the same integrand on $\mathbb{R}$, which is bounded by definition of per assumption. Hence, $\mathbb{E}_{\mathbf{z}}[\exp(\lambda \mathbf{z})] < \infty$ for all $0 < \lambda \leq \lambda^*$. □

Now we are ready to prove Theorem B.1.

Proof. The idea of the proof is to show that a linear combination of k (non-degenerate) random variables $\mathbf{z}_1, \dots, \mathbf{z}_k$ is light-tailed if and only if all $\mathbf{z}_1, \dots, \mathbf{z}_k$ are light-tailed. We do this via algebraic induction, i.e. we show that

1. $a\mathbf{z}_j$ is light-tailed iff. $\mathbf{z}_j$ is light-tailed for some scalar $a \in \mathbb{R}$;
2. $\mathbf{z}_1 + \mathbf{z}_2$ is light-tailed iff. $\mathbf{z}_1$ and $\mathbf{z}_2$ are both light-tailed.

For an arbitrary linear combination random variables, we can iterate through 1. and 2. to prove that the linear combination is light-tailed if and only if each component is light-tailed.

1. Let us assume that $\mathbf{z}_j$ is light-tailed. Therefore it exists some $\lambda^* \in \mathbb{R}$ such that $\mathbb{E}_{\mathbf{z}_j}[\exp(\lambda^* \mathbf{z}_j)] < \infty$. Further, for the moment-generating function of $a\mathbf{z}_j$ it is

$$\mathbb{E}_{a\mathbf{z}_j}\left[e^{\lambda \mathbf{x}}\right] = \int_{\mathbb{R}} p_{\mathbf{z}_j}(z_j) e^{\lambda a z_j} dx < \infty$$

for $\lambda := \lambda^*/a$, where the first equality follows from the LOTUS. The other direction of the equivalence follows analogously.
2. Consider $\mathbf{z}_1$ and $\mathbf{z}_2$ with joint PDF $q(z_1, z_2)$. Since both random variables are light-tailed, there exist λ_1 and λ_2 such that their moment-generating function is bounded. Then, it is for $\lambda := 0.5 \min\{\lambda_1, \lambda_2\}$

$$\begin{aligned}
\mathbb{E}_{\mathbf{z}_1+\mathbf{z}_2}\left[e^{\lambda z}\right] &= \int_{\mathbb{R}} \int_{\mathbb{R}} e^{\lambda(z_1+z_2)} q(z_1, z_2) dz_1 dz_2 \quad \text{(LOTUS)} \\
&= \int_{\mathbb{R}} \int_{-\infty}^{0} e^{\lambda(z_1+z_2)} q(z_1, z_2) dz_1 dz_2 + \int_{\mathbb{R}} \int_{0}^{\infty} e^{\lambda(z_1+z_2)} q(z_1, z_2) dz_1 dz_2 \\
&\leq \int_{\mathbb{R}} e^{\lambda z_2} q(z_2) dz_2 + \int_{\mathbb{R}} \Bigg(\int_{0}^{\min\{0,z_2\}} e^{\lambda(z_1+z_2)} q(z_1, z_2) dz_1 \\
&\qquad + \int_{\max\{0,z_2\}}^{\infty} e^{\lambda(z_1+z_2)} q(z_1, z_2) dz_1 \Bigg) dz_2 \ ,
\end{aligned}$$

where the last line follows by replacing z_2 in the first integral by 0, which is an upper bound for z_2. Using the definition of λ and Lemma B.2, we see that the first integral is bounded by some constant c_1. Using the monotonicity of the integrands once more, we get

$$\begin{aligned}
\mathbb{E}_{\mathbf{z}_1+\mathbf{z}_2}\left[e^{\lambda z}\right] &\leq c_1 + \int_{\mathbb{R}} \left(\int_{0}^{\min\{0,z_2\}} e^{2\lambda z_2} q(z_1, z_2) dz_1 + \int_{\max\{0,z_2\}}^{\infty} e^{2\lambda z_1} q(z_1, z_2) dz_1 \right) dz_2 \\
&\leq c_1 + \int_{\mathbb{R}} e^{2\lambda z_2} q(z_2) dz_2 + \int e^{2\lambda z_1} q(z_1) dz_1 \ ,
\end{aligned}$$

which follows by upper-bounding the integral from 0 to $\min\{0, z_2\}$ and the integral from $\max\{0, z_2\}$ by the integrals over $\mathbb{R}$, respectively. Using Lemma B.2 and the definition of λ once more, we can find constants c_2 and c_3 that bound the remaining integrals. Hence, we conclude that $\mathbb{E}_{\mathbf{z}_1+\mathbf{z}_2} < \infty$, which proves the backward direction of 2. To prove the forward direction, we show that if $\mathbf{z}_1$ and $\mathbf{z}_2$ are not both light-tailed (i.e at least of them is heavy-tailed), than $\mathbf{z}_1 + \mathbf{z}_2$ is not light-tailed. Without loss of generality, we assume $\mathbf{z}_2$ to be heavy-tailed. Then, we can write

$$\begin{aligned}
\mathbb{E}_{\mathbf{z}_1+\mathbf{z}_2}\left[e^{\lambda z}\right] &= \int_{\mathbb{R}} \int_{\mathbb{R}} e^{\lambda(z_1+z_2)} q(z_1, z_2) dz_1 dz_2 \\
&= \int_{\mathbb{R}} \left(\int_{-\infty}^{0} e^{\lambda(z_1+z_2)} q(z_1, z_2) dz_1 + \int_{0}^{\infty} e^{\lambda(z_1+z_2)} q(z_1, z_2) dz_1 \right) dz_2 \\
&\geq \int_{\mathbb{R}} \left(\int_{-\infty}^{0} e^{\lambda(z_1+z_2)} q(z_1, z_2) dz_1 + \int_{0}^{\infty} e^{\lambda(z_2)} q(z_1, z_2) dz_1 \right) dz_2 \ . \qquad (18)
\end{aligned}$$

Note that we can lower-bound the last inner integral by

$$\int_{0}^{\infty} e^{\lambda z_2} q(z_1, z_2) dz_1 = e^{\lambda z_2} q(z_2) - e^{\lambda z_2} \int_{-\infty}^{0} q(z_1, z_2) dz_1 \geq -e^{\lambda z_2} \int_{-\infty}^{0} q(z_1, z_2) dz_1 \ .$$

Plugging this bound into (18) gives us

$$\mathbb{E}_{\mathbf{z}_1+\mathbf{z}_2}\left[e^{\lambda z}\right] \geq \int_{\mathbb{R}} e^{\lambda z_2} q(z_2) dz_2 = \infty \ \ \forall \lambda > 0 \ ,$$

which is due to the heavy-tailedness of $\mathbf{z}_2$. Therefore, $\mathbf{z}_1 + \mathbf{z}_2$ must also be heavy-tailed. This proves the equivalence in 2. Finally, we note that due to the block-triangular form of A the first d_l components of $\mathbf{x} := A\mathbf{z}$, i.e. $\mathbf{x}_1, \ldots, \mathbf{x}_{d_l}$ are linear

combinations of light-tailed components $\mathbf{z}_1, \dots, \mathbf{z}_{d_l}$, which implies the light-tailedness of $\mathbf{x}_1, \dots, \mathbf{x}_{d_l}$. The remaining $D - d_l$ components of $\mathbf{x}$ are linear combinations containing at least one heavy-tailed component $\mathbf{z}_j$ with $j \in \{d_l, \dots, D\}$ and are therefore again heavy-tailed[15]. This completes the proof. □

B.2. Implementation of Data-Driven LU-Layers

In the previous section, we have seen that in order to retain the tail behavior, the block-matrix form given in (17) is sufficient. In this section, we give more details on an efficient parameterization leading to fast inversion and log-determinant computations. It is well-known that under mild conditions the inversion of block-matrices is efficiently solvable.

Lemma B.3 (Inversion of Block-Matrices)**.** *Consider invertible square matrices $A \in \mathbb{R}^{d_l \times d_l}$, $D \in \mathbb{R}^{d_h \times d_h}$ and arbitrary matrices $B \in \mathbb{R}^{d_l \times d_h}$, $C \in \mathbb{R}^{d_h \times d_l}$ for some $d_l, d_h \in \mathbb{N}$. Then it holds that*

$$\begin{pmatrix} A & B \\ C & D \end{pmatrix}^{-1} = \begin{pmatrix} (A - BD^{-1}C)^{-1} & 0 \\ -D^{-1}C(A - BD^{-1}C)^{-1} & D^{-1} \end{pmatrix} \begin{pmatrix} I & -BD^{-1} \\ 0 & I \end{pmatrix} .$$

As a special case, it is

$$\begin{pmatrix} A & 0 \\ B & C \end{pmatrix}^{-1} = \begin{pmatrix} A^{-1} & 0 \\ -C^{-1}BA^{-1} & C^{-1} \end{pmatrix} \tag{19}$$

for invertible square matrices $A \in \mathbb{R}^{d_l \times d_l}$, $C \in \mathbb{R}^{d_h \times d_h}$ and arbitrary matrix $B \in \mathbb{R}^{d_h \times d_l}$.

Lemma B.3 is a standard result that can be found in many linear algebra text books (see e.g. Gallier (2011)).

Furthermore, we can compute the determinant of a diagonal block-matrix W as defined in (17) as

$$\det(W) = \det(A)\det(C) \ . \tag{20}$$

Now, let us summarize the expensive computations in (19) and (20) that both need to be efficient in NFs. The computations involve inversions of A and C, only forward computations of B, and the computation of $\det(A)$ and $\det(C)$. Hence, we propose to parameterize A and C using a LU-decomposition for both matrices, which leads to efficient inverse[16] and log-determinant computations. We do not make any restrictions on B and parameterize it by a standard unconstrained linear layer. We provide a PyTorch implementation of this modified *tail-preserving LU-layer*, which can be accessed through our public git repository.

C. Algorithms and computational Details

C.1. Tail Estimation

Many heavy-tailed distributions can be characterized by their tail index, which include the set of regularly varying distributions,[17] such as the t-distribution, the Pareto distribution, and many more. However, as already shown in Section 2.1, the tail index does not depend on the body of the distribution, and hence, non-tail samples must typically be discarded for tail index estimation. Although a variety of estimators for the tail index exist, such as the Hill estimator (Hill, 1975), the moment estimator (Dekkers et al., 1989), and kernel-based estimators (Csorgo et al., 1985), none of them is considered to be as superior in all settings. A major issue of all mentioned estimators is that they are based on a threshold defining the tail, i.e. the user needs to input statements of the form "the k largest samples are considered to be tail events". Even though there exist some strategies to find k, there is none working robustly in all settings. In fact, one can construct simple counter examples for all estimators that lead to failures of tail estimation. This is due to undesired properties of the estimators, such as the lack of translation invariance of the Hills estimator (while the tail index clearly is location invariant). We refer to Section 9 in Nair et al. (2013) for a detailed text book treatment of tail index estimation. In summary, robust tail estimation is still considered as an unsolved problem, which forces practitioners to consider multiple estimators to make a well-founded decision. Furthermore, we note that the Hills estimator can only be applied for regularly varying distributions, which

[15] Note that this argument assumes that D has no empty rows, which is implicitly assume due to the invertibility of A. Compare with Equation ref(Algosection).

[16] which can be guaranteed by restricting the diagonal elements of the upper diagonal matrix in the LU-decomposition.

[17] see Section 2 in (Nair et al., 2013) for further details

Algorithm 1 Marginal Tail Estimation

```
tail.est ← [ ]
for j in {1, ..., D} do
  marginal ← Data_val[:, j]
  moments ← moments_est(|marginal|) {0 if |marginal| is estimated to be light-tailed}
  kernel ← kernel_est(|marginal|) {0 if |marginal| is estimated to be light-tailed}
  if moments==kernel==0 then
    tail_est.append(0) {light-tailed if moments and kernel estimate a light-tailed marginal}
  else
    hill ← hills_est(|marginal|)
    if hill> 10 then
      tail_est.append(0) {light-tailed if hills estimator predicts high tail index}
    else
      tail_est.append(hill)
    end if
  end if
end for
tail_est
```

excludes the application of the Hills estimator to classify light-tailed distributions. In contrast, the moments and the kernel estimator can both be applied to identify heavy-tailed marginals and to assess a tail index.

To implement the tail assessment scheme, see Step 1 of the proposed method in Section 3.2, we found that Algorithm 1 works fine in classifying the correct tail behavior and giving a decent initialization for the tail indices. That is, we use the moments double-bootstrap estimator (Draisma et al., 1999) and the kernel-type double-bootstrap estimator (Groeneboom et al., 2003) to assess heavy-tailedness of the data distribution. If both estimators predict a light-tailed distribution, we set the corresponding marginal base distribution to be standard normal distributed, otherwise we set it to a the standardized t-distribution, i.e. $\mathbf{z}_j \sim t_{\hat{\nu}_j}$, where $\hat{\nu}_j$ is the Hill double-bootstrap estimator (Danielsson et al., 2001; Qi, 2008). We reused the code by Voitalov et al. (2019), which implements all tail estimation procedures[18] from our Algorithm. Notice that we clip the tail index by 10, i.e. the algorithm classifies marginals with a tail index larger than 10 as light-tailed, which prevents a too restrictive set of allowed permutations, see Step 3 in Section 3.2. To illustrate this argument, consider the following simple example. Assume that we estimate all except of one marginal to be heavy-tailed. Then, the first component of the flow is never allowed to permute with other components, since they are classified as heavy-tailed. Hence, the mixing of the first component would be severely restricted. Further, since large tail indices indicate a less heavy-tailed distribution, it is reasonable to clip the tail index at some threshold.

C.2. Synthetic Data Generation

The generation of the synthetic distribution consists of 3 steps: 1. Generating the marginal distributions, 2. Defining a copula distribution, 3. Combining the marginal and the copula to obtain a multivariate joint distribution.

Generating the marginal distributions. The first two marginals are defined to be Gaussians. The following marginals are a 2-mixtures of Gaussians and a mixtures of three Gaussians. The last $d_h \in \{1, 4\}$ components are a mixture of two t-distributions and the remaining marginals are again mixtures of two Gaussians. All mixtures have equal weight for each mixture component and all means and standard-deviations are randomized. Means are constructed by uniformly sampling from $[-4, 4]$, whereas standard-deviations are sampled from $[1, 2]$.

Defining a copula distribution. Recall, a Gaussian copula (13) is parameterized by a correlation matrix R. To generate R, we randomly sample 16 different pairs $(i, j) \in \{1, \dots, 8\}^2$ with $i \neq j$ and set the corresponding entry of the correlation matrix $R_{i,j} := 0.25$. The diagonals of R are set to 1.

[18] including the hyperparameter selection (Danielsson et al., 2001; Qi, 2008; Draisma et al., 1999; Groeneboom et al., 2003)

Obtaining a joint distribution. Lastly, we combine the marginals with the Gaussian copula using Sklar's Theorem A.4. This gives us a multivariate distribution with specified and complex marginals with a dependency structure given by the copula, see Joe (2014) for more details on the induced dependencies.

To construct the training, test, and validation sets $15.000,\ 75\,000$, and $10\,000$ samples from this distribution are sampled, respectively.

In the setting $D = 50$, we apply a similar procedure but with $d_h = 10$ heavy-tailed components and with the first 40 marginals being 2-mixtures of Gaussians, the remaining 10 marginals being 2-mixtures t-distributions. We define the dependency-structure by randomly selecting 200 pairs $(i, j) \in \{1, \dots, 50\}^2$ with $i \neq j$ and set $R_{i,j} = 0.25$ again. Training, validation, and test sets consists of $50\,000,\ 10\,000$, and $75\,000$ samples, respectively.

C.3. Synthetic Experiments

In all synthetic experiments, we used a NSF with 5 layers and corresponding LU-linearities. mTAF employs the modified LU-linearities from Section 3.3. In the NSF layers, we used conditioner ResNets with 2 hidden layers with 30 or 200 hidden neurons in the case $D = 8$ and $D = 50$, respectively and ReLU activations. Further, we used NSF layers with 3 bins and set the tail-bound to 2. We optimized the network using Adam with $5\,000$ or $20\,000$ train steps in the case $D = 8$ and $D = 50$, respectively, with a learning rate of $1e-5$ and a weight decay of $1e-6$. To fit the Gaussian copula baseline, we use the default settings of the *copulas* (Patki et al., 2016) library.

To assess the sample quality on the tail of the distribution, we consider 3 metrics.

1. **Tail Value at Risk** for some level α, which is defined by

$$\mathrm{tVaR}_\alpha := \mathrm{tVaR}_\alpha(F) := \frac{1}{1-\alpha}\int_\alpha^1 F^{-1}(u)du$$

   for some CDF F. tVaR_α is a well-known metric and is widely used in finance (McNeil et al., 2015). We plug in the marginal empirical CDFs $\hat{F}_{\text{data}}$ and $\hat{F}_{\text{flow}}$, i.e. the empirical CDFs based on the data and on synthetic samples, respectively, and calculate the absolute difference between these quantities. The resulting metric is the marginal tVaR-difference for the level α. We set $\alpha = 0.95$ in all our experiments.

2. **Area under log-log plot** is defined by

$$\text{Area} := \sum_{i=1}^{n}\left|\log \bar{F}^{-1}_{\text{data}}\left(\frac{i}{n}\right) - \log \bar{F}^{-1}_{\text{flow}}\left(\frac{i}{n}\right)\right| \log\frac{i+1}{i}\ ,$$

   where $\bar{F}^{-1}_{\text{data}}$, $\bar{F}^{-1}_{\text{flow}}$ denote the inverse empirical complementary CDFs given by the test data and the flow samples, respectively.

3. **Synthetic Tail Estimates**, where the tail-assessment is similar to the on described in Algorithm 1. We can then assess the similarity between the tail estimators based on the data and the tail estimators based on synthetically generated flow samples.

Setting D=8 In our experimental study, we generated 3 synthetic distributions per setting as explained in Section C.2 and fit each model 25 times to each synthetic distribution. While it is reasonable to compare the averaged metrics over all 75 runs, investigating the standard deviation over all runs might be misleading since the metrics could be centered around different values for each synthetic distribution. For this reason, it is more insightful to compare standard deviations over runs where the target distribution is fixed, which we present in Table 3. In Table 4 we present the numeric results of the synthetic experiments for a larger tail index, i.e. a less extreme setting. In this setting, we observe that gTAF tends to perform slightly worse for light-tailed components, while achieving good results for heavy-tailed components. Note that in this case, the performance of mTAF degrades, which might be attributed to the less flexible structure of its linearities. When replacing the NSF-layers by MAF-layers, we observe that the MAF fails to converge for a vanilla base distribution. Again, mTAF strikes a balance between fitting heavy- and light-tailed marginals but the overall performance is better in the case of NSF-layers—especially for the heavy-tailed marginals. We conjecture that this is due to the linearity of the

Table 3. Standard deviations corresponding to the experiments shown in Table 1, i.e. in the setting $\nu = 2$ and $d_h \in \{1, 4\}$, for one target distribution.

d_h	1					4				
	L	Area_l	Area_h	tVaR_l	tVaR_h	L	Area_l	Area_h	tVaR_l	tVaR_h
vanilla	0.02	0.02	0.12	0.16	3.74	0.03	0.03	0.14	0.13	1.75
TAF	0.00	0.03	0.20	0.12	2.38	0.01	0.06	0.12	0.22	1.15
gTAF	0.01	0.06	0.26	0.17	1.30	0.01	0.06	0.14	0.20	0.64
mTAF	0.01	0.03	0.21	0.10	1.41	0.01	0.03	0.20	0.18	1.32

Table 4. Average test loss, Area under log-log plot, and tVaR (lower is better for each metric) in the setting $\nu = 3$ and $d_h \in \{1, 4\}$. The copula model serves as an oracle baseline.

d_h	1					4				
	L	Area_l	Area_h	tVaR_l	tVaR_h	L	Area_l	Area_h	tVaR_l	tVaR_h
vanilla	10.82	0.25	2.77	0.55	12.06	10.55	0.23	2.64	0.58	9.98
TAF	10.79	0.36	2.78	0.79	1.49	10.46	0.38	2.95	0.93	2.36
gTAF	10.77	0.55	1.11	1.27	2.82	10.38	0.50	1.13	1.00	2.56
mTAF	10.76	0.33	2.03	1.05	7.36	10.38	0.34	1.54	1.09	5.02
copula	9.76	0.20	0.79	0.45	1.82	9.76	0.19	0.91	0.46	1.62

tails of each NSF-layer[19], which leads to a better generalization for those low-sample regions. Therefore, we continue the following experiments using the NSF architecture. Furthermore, we investigate the tail indices of the generated samples by constructing confusion matrices similar to those in Figure 2, which we present in Figure 6. We observe a similar behavior, that is, while vanilla and TAF produce mainly marginals with light-tailed marginals, gTAF is able to produce much better samples with heavy-tailed marginals. Again, mTAF produces marginals whose marginal tail behavior almost perfectly fits the true tail behavior. For further visual inspection of the generated marginals, we consider QQ-plots of the heavy-tailed components in Figure 7 and Figure 8. In both cases, vanilla and TAF—and sometimes in gTAF—we observe humps in the tails, which surrogate a bad sampling performance in their tails. mTAF does not have these, which is in accordance to our findings derived from Figure 2 and 6.

Setting D=50 Table 6 compares the quantitative metrics for our synthetic experiments in the case $D = 50$. In this case, we generate 3 synthetic distributions as explained in Section C.2 and fit each model 5 times. We observe that mTAF clearly outperforms vanilla and TAF in terms of Area, while the flexibility in gTAF allows it to perform almost on par with the oracle copula baseline. Considering the metrics that account for the tail fit of the heavy-tailed components, we surprisingly observe that gTAF even outperforms the oracle copula model. However, considering the light-tailed components, gTAF performs a bit worse, which is not surprising since gTAF models each marginal distribution by a t-distribution.

C.4. Climate Data

In this section, we provide more details on the employed architectures of mTAF on the NWP-SAF dataset, which we visualize in Figure 9. Furthermore, we present a more in-depth discussion about the results.

We consider each quantity (i.e. dry-bulb air temperature, atmospheric pressure, and cloud optical depth) at each atmospheric level as one component, leading to a 412-dimensional dataset. Recall that mTAF requires a classification into light- and heavy-tailed marginals that lead to a reordering of the initial marginals, which we do as follows. This dataset—similar to other time-series-like datasets—gives us a natural autoregressive ordering, which we make use of in our permutation step. This leads to the initial permutation (compare with Step 2 in Figure 1 and Section 3.2), in which the first components

[19]This does not mean that the whole flow is linear in its tails since tail samples can be linearly mapped in and out of the bins, leading a non-linear mapping.

Table 5. Average test loss, Area under log-log plot, and tVaR (lower is better for each metric) in the setting $\nu = 2$ and $d_h \in \{1, 4\}$ when using a MAF.

d_h	1					4				
	L	Area_l	Area_h	tVaR_l	tVaR_h	L	Area_l	Area_h	tVaR_l	tVaR_h
vanilla	>1e6	0.82	4.88	2.76	8.67	>1e6	0.95	5.97	3.78	9.93
TAF	10.63	1.09	4.68	3.88	8.00	9.97	1.12	5.90	4.28	9.42
gTAF	10.60	1.26	3.78	4.71	2.91	9.84	1.32	4.89	5.29	4.22
mTAF	10.55	0.77	4.00	2.51	3.08	9.81	0.48	5.01	1.32	4.49

Table 6. Average test loss, Area under log-log plot, and tVaR (lower is better for each metric) in the setting $D = 50,\ \nu \in \{2, 3\}$, and $d_h = 10$. The copula model serves as an oracle baseline.

ν	2					3				
	L	Area_l	Area_h	tVaR_l	tVaR_h	L	Area_l	Area_h	tVaR_l	tVaR_h
vanilla	58.59	0.29	3.38	0.45	24.61	62.59	0.30	1.71	0.43	6.38
TAF	58.12	0.67	3.02	1.20	3.19	62.47	0.57	2.02	0.96	2.15
gTAF	58.05	0.94	0.58	1.23	1.16	62.42	0.61	0.51	0.96	0.99
mTAF	58.17	0.39	2.30	0.84	4.20	62.60	0.30	1.32	0.46	2.86
copula	57.23	0.21	1.04	0.42	2.50	56.73	0.20	0.66	0.41	1.33

are given by light-tailed components of the dry-bulb air temperature, followed by the reversed atmospheric pressure and the cloud-optical depth. However, in contrast to our synthetic experiments, where the selection of the heavy-tailed components was more or less trivial, this task is more complicated in a time series with highly dependent features. What we found works best in practice, is to deliberately choose a large set of heavy-tailed components according to Table 7, while making the degree of freedom learnable. Furthermore, we implemented all NFs—vanilla, TAF, gTAF, and mTAF—using 5 autoregressive NSF layers with LU-linearities and their modified versions from Section 3.3. The conditioner networks in the NSF-layers have 2 hidden layers with 100 hidden neurons in each layer, we set the tail-bounds to 2.5, and each spline uses 3 bins. We apply Batch-Norm after each NSF-layer. We optimize for $20\,000$ steps using the Adam optimizer with a learning rate of 1e-4 and a learning rate of 0.01 for the tail indices and scheduled the rates using cosine annealing.

We plot synthetic samples from the remaining flows in Figure 10.

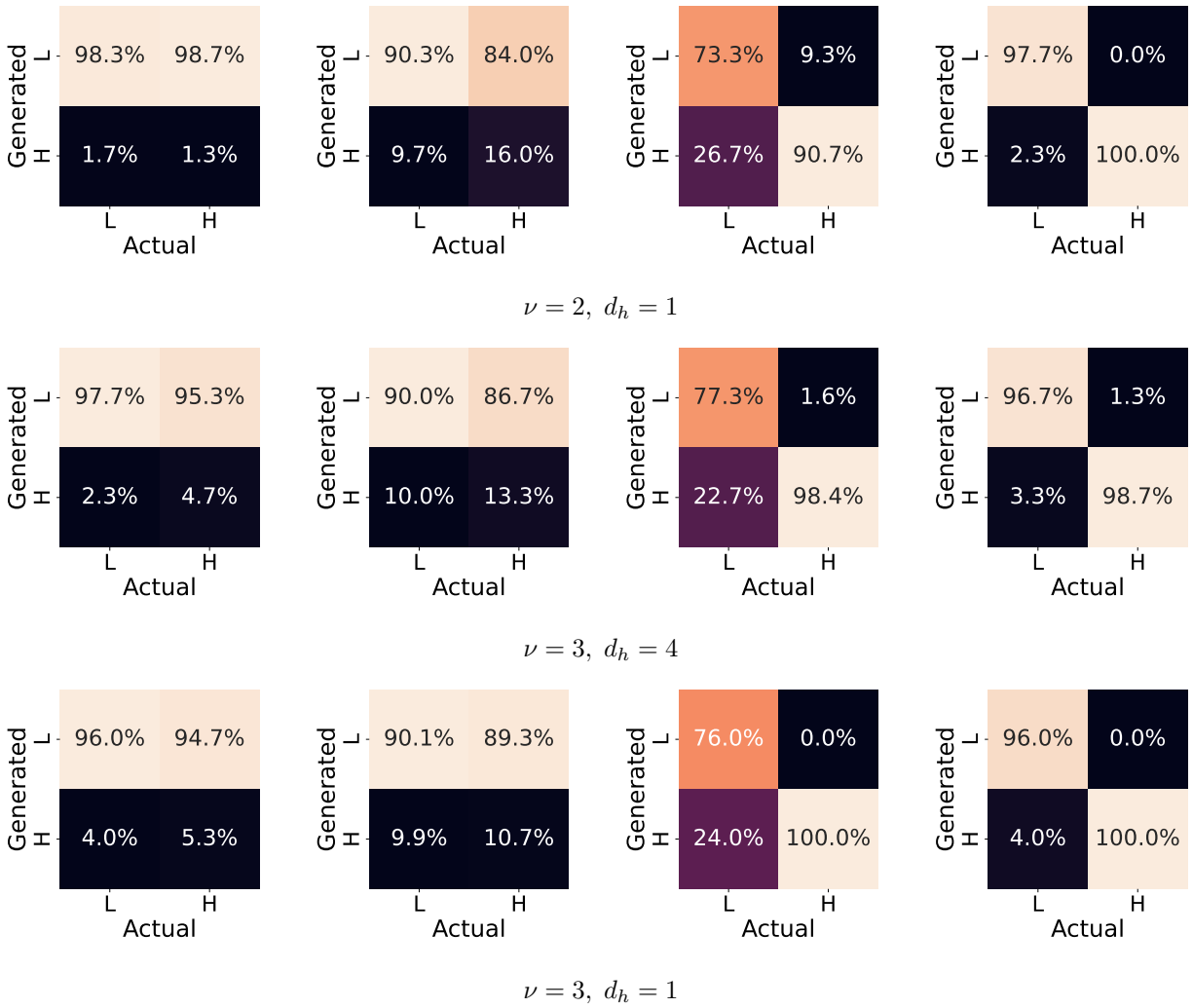


Figure 6. Marginal tail estimation based on synthetic flow samples of vanilla, TAF, gTAF, and mTAF (from left to right) for varying tail index ν and number of heavy-tailed components d_h. We classify marginals whose tail estimator is less than 10 as heavy-tailed, otherwise it is classified as light-tailed.

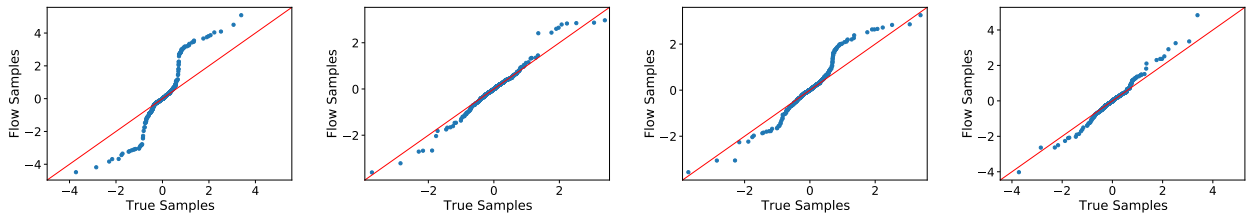


Figure 7. QQ-plots for the 8th heavy-tailed marginal in the setting $\nu = 2$ and $d_h = 1$. The QQ-plots correspond to samples generated by vanilla, TAF, gTAF, and mTAF, respectively.

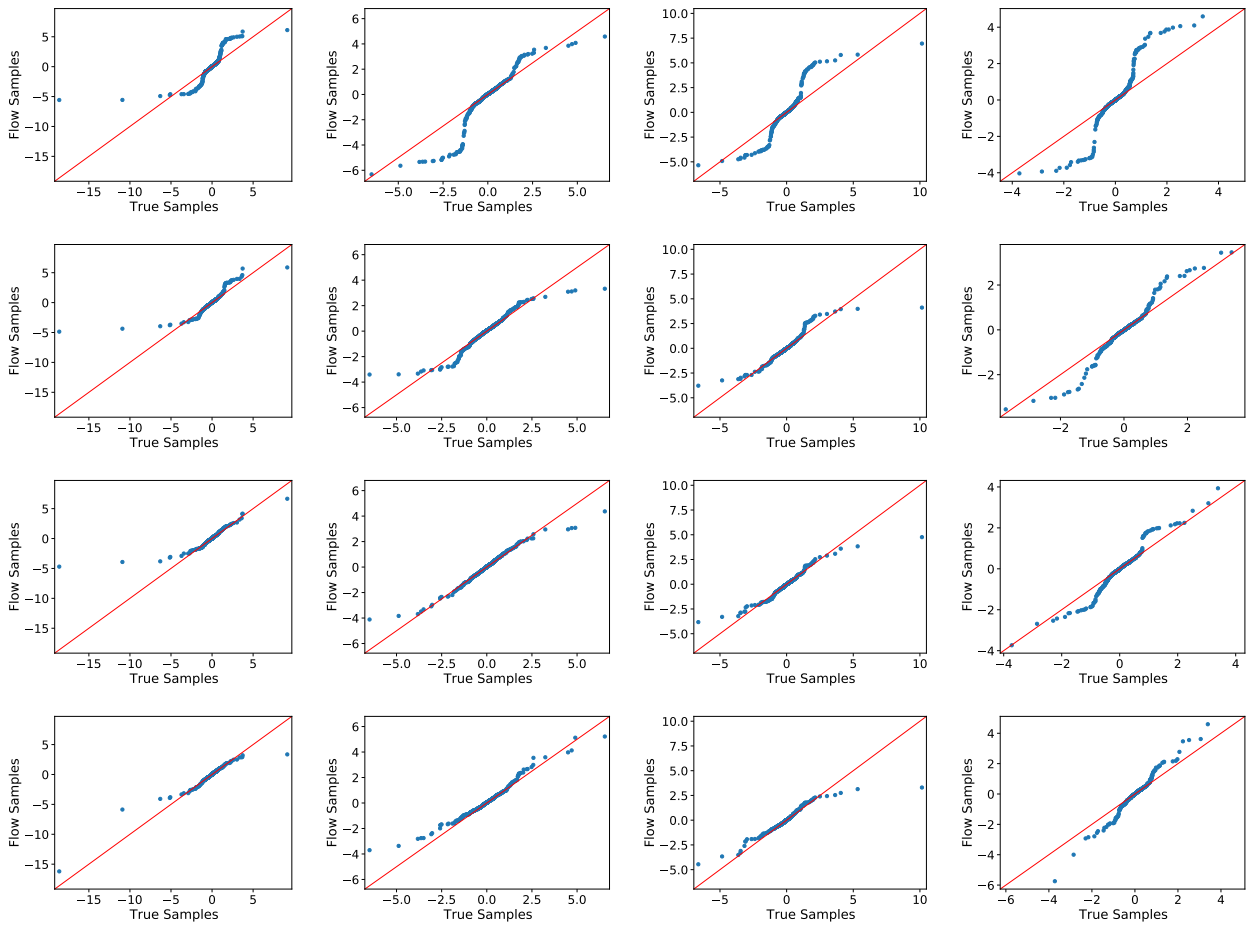


Figure 8. QQ-plots for the last 4 heavy-tailed marginals in the setting $\nu = 2$ and $d_h = 4$. The rows of QQ-plots correspond to samples generated by vanilla, TAF, gTAF, and mTAF, respectively. Each marginal in depicted in one column.

Table 7. Components (i.e. measurements at a specific atmospheric level), which we manually select as heavy-tailed based on Figure 9.

MEASUREMENT	LIGHT-TAILED	HEAVY-TAILED
DRY-BULB AIR TEMPERATURE IN K	1 - 79	80 - 137
ATMOSPHERIC PRESSURE IN HPA	1 - 99	100 - 137
CLOUD OPTICAL DEPTH	1 - 57	58 - 137

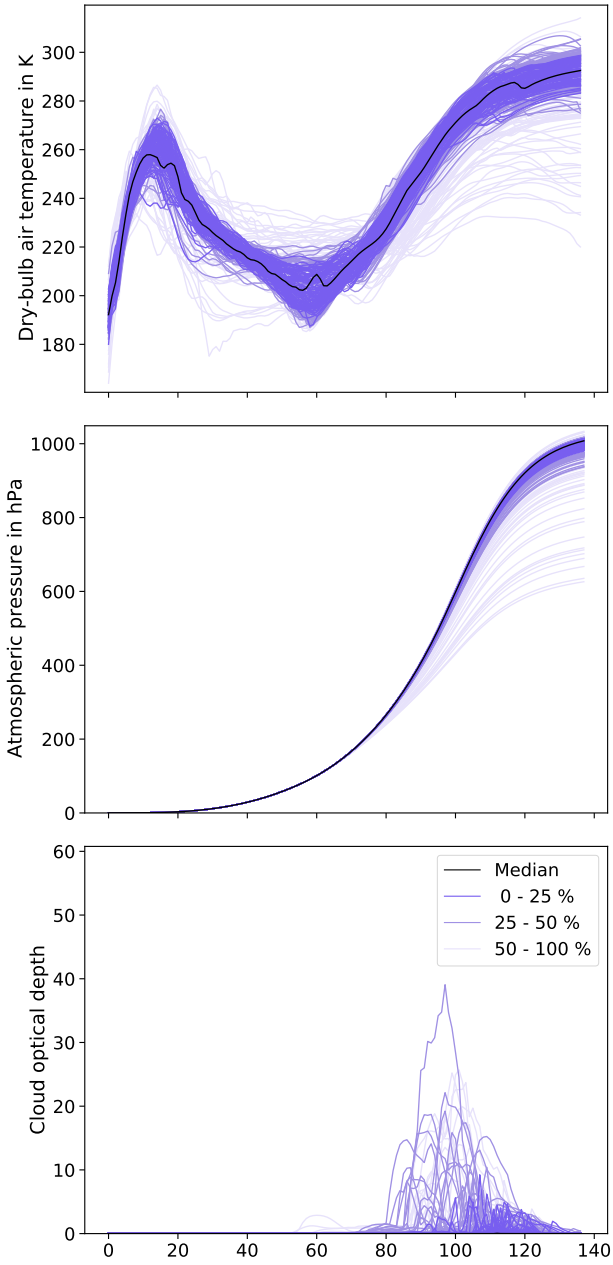


Figure 9. Real profiles from the NWP-SAF dataset. We used the implementation by Meyer et al. (2021) to generate the figure. The profiles are ordered using band depth statistics (Pintado & Romo, 2009).

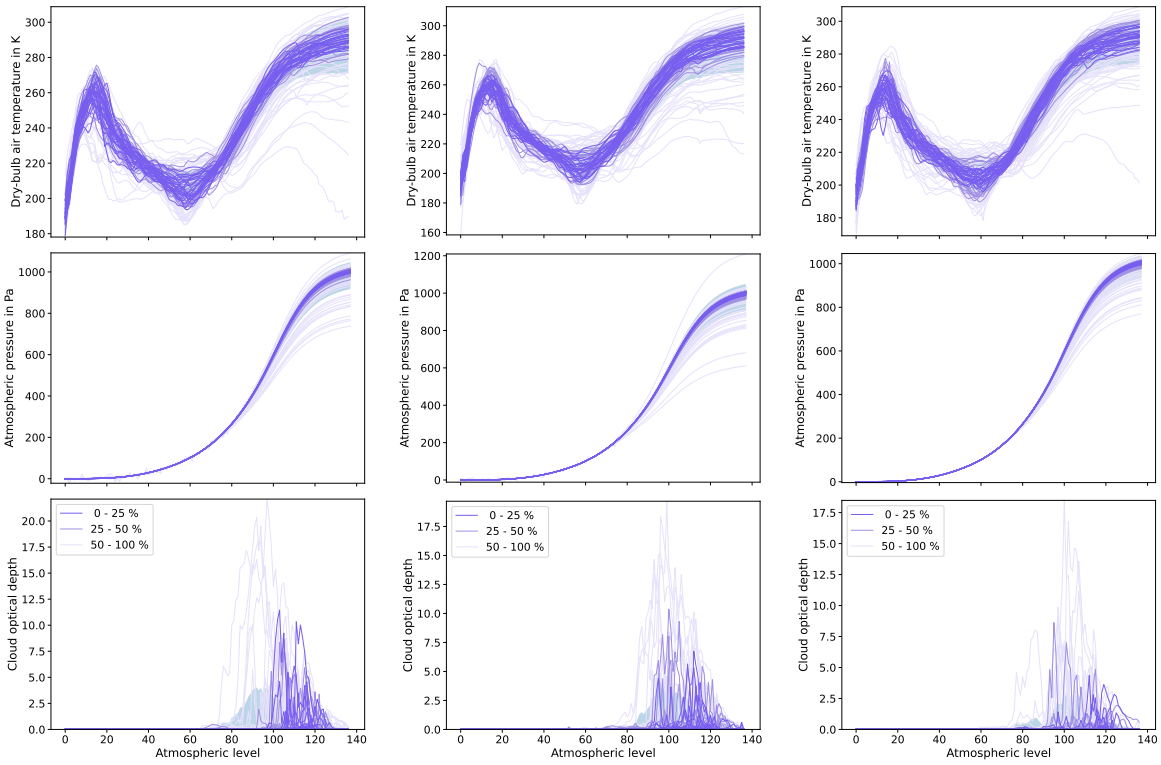


Figure 10. Synthetic flow samples using vanilla, TAF, and gTAF (from left to right), where we clipped the lower-values of the cloud-optical depth at 0. The corresponding negative log-likelihoods are $-2094.35,\ -2117.48,\ -2121.65$, respectively. We used the implementation by Meyer et al. (2021) to generate the figure. The profiles are ordered using band depth statistics (Pintado & Romo, 2009) and the shaded areas represent standard deviations.