risks

MDPI

Article

Grouped Normal Variance Mixtures

Erik Hintz *, Marius Hofert and Christiane Lemieux

Department of Statistics and Actuarial Science, University of Waterloo, 200 University Avenue West, Waterloo, ON N2L 3G1, Canada; marius.hofert@uwaterloo.ca (M.H.); clemieux@uwaterloo.ca (C.L.)
* Correspondence: erik.hintz@uwaterloo.ca

Received: 29 July 2020; Accepted: 30 September 2020; Published: 7 October 2020

Abstract: Grouped normal variance mixtures are a class of multivariate distributions that generalize classical normal variance mixtures such as the multivariate t distribution, by allowing different groups to have different (comonotone) mixing distributions. This allows one to better model risk factors where components within a group are of similar type, but where different groups have components of quite different type. This paper provides an encompassing body of algorithms to address the computational challenges when working with this class of distributions. In particular, the distribution function and copula are estimated efficiently using randomized quasi-Monte Carlo (RQMC) algorithms. We propose to estimate the log-density function, which is in general not available in closed form, using an adaptive RQMC scheme. This, in turn, gives rise to a likelihood-based fitting procedure to jointly estimate the parameters of a grouped normal mixture copula jointly. We also provide mathematical expressions and methods to compute Kendall's tau, Spearman's rho and the tail dependence coefficient λ. All algorithms presented are available in the R package nvmix (version $\geq$ 0.0.5).

Keywords: grouped normal variance mixtures; distribution functions; densities; copulas; grouped t copula; risk measures; quasi-random number sequences

1. Introduction

It is well known that for the purpose of modeling dependence in a risk management setting, the multivariate normal distribution is not flexible enough, and therefore its use can lead to a misleading assessment of risk(s). Indeed, the multivariate normal has light tails and its copula is tail-independent such that inference based on this model heavily underestimates joint extreme events. An important class of distributions that generalizes this simple model is that of *normal variance mixtures*. A random vector $\boldsymbol{X} = (X_1, \dots, X_d)$ follows a normal variance mixture, denoted by $\boldsymbol{X} \sim \mathrm{NVM}_d(\boldsymbol{\mu}, \Sigma, F_W)$, if, in distribution,

$$\boldsymbol{X} = \boldsymbol{\mu} + \sqrt{W} A \boldsymbol{Z}, \tag{1}$$

where $\boldsymbol{\mu} \in \mathbb{R}^d$ is the *location (vector)*, $\Sigma = AA^\top$ for $A \in \mathbb{R}^{d \times k}$ denotes the symmetric, positive semidefinite *scale (matrix)* and $W \sim F_W$ is a non-negative random variable independent of the random vector $\boldsymbol{Z} \sim \mathrm{N}_k(\boldsymbol{0}, I_k)$ (where $I_k \in \mathbb{R}^{k \times k}$ denotes the identity matrix); see, for example, (McNeil et al. 2015, Section 6.2) or (Hintz et al. 2020). Here, the random variable W can be thought of as a shock mixing the normal $\boldsymbol{Z}$, thus allowing $\boldsymbol{X}$ to have different tail behavior and dependence structure than the special case of a multivariate normal.

The multivariate t distribution with $\nu > 0$ degrees of freedom (dof) is also a special case of (1), for $W \sim \mathrm{IG}(\nu/2, \nu/2)$; a random variable (rv) W is said to follow an inverse-gamma distribution with shape $\alpha > 0$ and rate $\beta > 0$, notation $W \sim \mathrm{IG}(\alpha, \beta)$, if W has density $f_W(w) = \beta^{-\alpha} w^{-\alpha-1} \exp(-1/(\beta w))/\Gamma(\alpha)$ for $w > 0$ (here, $\Gamma(x) = \int_0^\infty t^{x-1} \exp(-t)\,\mathrm{d}t$ denotes the gamma

function). If $W \sim \mathrm{IG}(\nu/2, \nu/2)$, then $X_j \sim t_\nu(\mu_j, \Sigma_{jj})$, $j = 1, \dots, d$, so that all margins are univariate t with the same dof ν. The t copula, which is the implicitly derived copula from $\boldsymbol{X} \sim t_d(\nu, \mathbf{0}, P)$ for a correlation matrix P via Sklar's theorem, is a widely used copula in risk management; see, e.g., Demarta and McNeil (2005). It allows one to model pairwise dependencies, including tail dependence, flexibly via the correlation matrix P. When $P = I_d$, all k-dimensional margins of $\boldsymbol{X}$ are identically distributed. To overcome this limitation, one can allow different margins to have different dof. On a copula level, this leads to the notion of grouped t copulas of Daul et al. (2003) and generalized t copulas of Luo and Shevchenko (2010).

In this paper, we, more generally, define *grouped normal variance mixtures* via the stochastic representation

$$\boldsymbol{X} = \boldsymbol{\mu} + \mathrm{Diag}(\sqrt{\boldsymbol{W}})A\boldsymbol{Z}, \tag{2}$$

where $\boldsymbol{W} = (W_1, \dots, W_d)$ is a d-dimensional non-negative and comonotone random vector with $W_j \sim F_{W_j}$ that is independent of $\boldsymbol{Z}$. Denote by $F^{\leftarrow}_W(u) = \inf\{w \geq 0 : F_W(w) \geq u\}$ the quantile function of a random variable W. Comonotonicity of the W_j implies the stochastic representation

$$\boldsymbol{W} = (W_1, \dots, W_d) = (F^{\leftarrow}_{W_1}(U), \dots, F^{\leftarrow}_{W_d}(U)), \quad U \sim \mathrm{U}(0,1). \tag{3}$$

If a d-dimensional random vector $\boldsymbol{X}$ satisfies (2) with $\boldsymbol{W}$ given as in (3), we use the notation $\boldsymbol{X} \sim \mathrm{gNVM}(\boldsymbol{\mu}, \Sigma, F_{\boldsymbol{W}})$ where $F_{\boldsymbol{W}}(\boldsymbol{w}) = \mathbb{P}(\boldsymbol{W} \leq \boldsymbol{w})$ for $\boldsymbol{w} \in \mathbb{R}^d$ and the inequality is understood component-wise.

As mentioned above, in the case of an (ungrouped) normal variance mixture distribution from (1), the scalar random variable (rv) W can be regarded as a shock affecting all components of $\boldsymbol{X}$. In the more general setting considered in this paper where $\boldsymbol{W}$ is a vector of comonotone mixing rvs, different, perfectly dependent random variables affect different margins of $\boldsymbol{X}$. By moving from a scalar mixing rv to a comonotone random vector, one obtains non-elliptical distributions well beyond the classical multivariate t case, giving rise to flexible modeling of joint and marginal body and tail behaviors. The price to pay for this generalization is significant computational challenges: Not even the density of a grouped t distribution is available in closed form.

At first glance, the definition given in (2) does not indicate any "grouping" yet. However, Equation (3) allows one to group components of the random vector $\boldsymbol{X}$ such that all components within a group have the same mixing distribution. More precisely, let $\boldsymbol{W}$ be split into S sub-vectors, i.e., $\boldsymbol{W} = (\boldsymbol{W}_1, \dots, \boldsymbol{W}_S)$ where $\boldsymbol{W}_k$ has dimension d_k for $k = 1, \dots, S$ and $\sum_{k=1}^S d_k = d$. Now let each $\boldsymbol{W}_k$ have stochastic representation $\boldsymbol{W}_k = (F^{\leftarrow}_{W_k}(U), \dots, F^{\leftarrow}_{W_k}(U))$. Hence, all univariate margins of the subvector $\boldsymbol{W}_k$ are identically distributed. This implies that all margins of the corresponding subvector $\boldsymbol{X}_k$ are of the same type.

An example is the copula derived from $\boldsymbol{X}$ in (2) when $F_{W_k} = \mathrm{IG}(\nu_k/2, \nu_k/2)$ for $k = 1, \dots, S$; this is the aforementioned grouped t copula. Here, different margins of the copula follow (potentially) different t copulas with different dof, allowing for more flexibility in modeling pairwise dependencies. A grouped t copula with $S = d$, that is when each component has their own mixing distribution, was proposed in Venter et al. (2007) (therein called "individuated t copula") and studied in more detail in Luo and Shevchenko (2010) (therein called "t copula with multiple dof"). If $S = 1$, the classical t copula with exactly one dof parameter is recovered.

For notational convenience, derivations in this paper are often done for the case $S = d$, so that the F_{W_j} are all different; the case $S < d$, that is when grouping is present, is merely a special case where some of the F_{W_j} are identical. That being said, we chose to keep the name "grouped" to refer to this class of models so as to reflect the original motivation for this type of model, e.g., as in Daul et al. (2003), where it is used to model the components of a portfolio in which there are subgroups representing different business sectors.

Previous work on grouped t copulas and their corresponding distributions includes some algorithms for the tasks needed to handle these models, but were mostly focused on demonstrating the

superiority of this class of models over special cases such as the multivariate normal or t distribution. More precisely, in Daul et al. (2003), the grouped t copula was introduced and applied to model an internationally diversified credit portfolio of 92 risk factors split into 8 subgroups. It was demonstrated that the grouped t copula is superior to both the Gaussian and t copula in regards to modeling the tail dependence present in the data. Luo and Shevchenko (2010) also study the grouped t copula and, unlike in Daul et al. (2003), allow group sizes of 1 (corresponding to $S = d$ in our definition). They provide calibration methods to fit the copula to data and furthermore study bivariate characteristics of the grouped t copula, including symmetry properties and tail dependence.

However, to the best of our knowledge, there currently does not exist an encompassing body of work providing all algorithms and formulas required to handle these copulas and their corresponding distributions, both in terms of evaluating distributional quantities and in terms of general fitting algorithms. In particular, not even the problem of computing the distribution and density function of a grouped t copula has been addressed. Our paper fills this gap by providing a complete set of algorithms for performing the main computational tasks associated with these distributions and their associated copulas, and does so in an as automated way as possible. This is done not only for grouped t copulas, but (in many cases) for the more general grouped normal variance mixture distributions/copulas, which allow for even further flexibility in modeling the shock variables $\boldsymbol{W}$. Furthermore, we assume that the only available information about the distribution of the W_j are the marginal quantile functions in the form of a "black box", meaning that we can only evaluate these quantile functions but have no mathematical expression for them (so that neither the density, nor the distribution function of W_j are available in closed form).

Our work includes the following contributions: (i) we develop an algorithm to evaluate the distribution function of a grouped NVM model. Our method only requires the user to provide a function that evaluates the quantile function of the W_j through a black box. As such, different mixing distributions can be studied by merely providing a quantile function without having to implement an integration routine for the model at hand; (ii) as mentioned above, the density function for a grouped t distribution does not exist in closed form, neither does it for the more general grouped NVM case. We provide an adaptive algorithm to estimate this density function in a very general setting. The adaptive mechanism we propose ensures the estimation procedure is precise even for points that are far from the mean; (iii) to estimate Kendall's tau and Spearman's rho for a two-dimensional grouped NVM copula, we provide a representation as an expectation, which in turn leads to an easy-to-approximate two- or three-dimensional integral; (iv) we provide an algorithm to estimate the copula and its density associated with the grouped t copula, and fitting algorithms to estimate the parameters of a grouped NVM copula based on a dataset. While the problem of parameter estimation was already studied in Daul et al. (2003) and Luo and Shevchenko (2010), the computation of the copula density which is required for the joint estimation of all dof parameters has not been investigated in full generality for arbitrary dimensions yet, which is a gap we fill in this paper.

The four items from the list of contributions described in the previous paragraph correspond to Sections 3–6 of the paper. Section 2 includes a brief presentation of the notation used, basic properties of grouped NVM distributions and a description of randomized quasi-Monte Carlo methods that are used throughout the paper since most quantities of interest require the approximation of integrals. Section 7 provides a discussion. The proofs are given in Section 8.

All our methods are implemented in the R package `nvmix` (Hofert et al. (2020)) and all numerical results are reproducible with the demo `grouped_mixtures`.

2. Notation, Basic Properties and Tools

This section provides the necessary background, notations and tools needed throughout the remainder of this paper. The first part addresses some properties of grouped normal variance mixtures, such as mean, covariance and the relationship with elliptical distributions. The second part of this section describes randomized quasi-Monte Carlo methods, which is the type of integration method we

apply to approximate quantities that are expressed as expectations, such as the distribution function of grouped normal variance mixtures.

2.1. Grouped Normal Variance Mixtures

To simplify the notation throughout the remainder of this paper, we use the shorthand notations $F^{\leftarrow}_{\boldsymbol{W}}(U) = (F^{\leftarrow}_{W_1}(U), \dots, F^{\leftarrow}_{W_d}(U))$, $\boldsymbol{W}^{\mathrm{D}} = \mathrm{Diag}(\boldsymbol{W})$, $\boldsymbol{W}^{\mathrm{D}}(U) = \mathrm{Diag}(F^{\leftarrow}_{\boldsymbol{W}}(U))$; $\sqrt{\boldsymbol{W}}^{\mathrm{D}}$, $(1/w)^{\mathrm{D}}$ as well as $(1/\sqrt{\boldsymbol{w}})^{\mathrm{D}}$ are defined similarly.

Many properties of grouped normal variance mixtures are derived by conditioning on the d-dimensional random vector $\boldsymbol{W}$, or equivalently by conditioning on the underlying univariate uniform rv U. Indeed,

$$\boldsymbol{X} \mid \boldsymbol{W} \sim \mathrm{N}_d\left(\boldsymbol{\mu}, \sqrt{\boldsymbol{W}}^{\mathrm{D}} \Sigma \sqrt{\boldsymbol{W}}^{\mathrm{D}}\right) \quad \text{or equivalently} \quad \boldsymbol{X} \mid U \sim \mathrm{N}_d\left(\boldsymbol{\mu}, \sqrt{\boldsymbol{W}}^{\mathrm{D}}(U) \Sigma \sqrt{\boldsymbol{W}}^{\mathrm{D}}(U)\right),$$

where $\mathrm{N}_d(\boldsymbol{\mu}, \Sigma)$ denotes the d-dimensional multivariate normal distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix Σ. One can see that $\boldsymbol{W}$ "mixes" the covariance matrix of a multivariate normal and can be regarded as a shock affecting all components of $\boldsymbol{X}$.

2.1.1. Mean and Covariance

If $\mathbb{E}(\sqrt{\boldsymbol{W}})$ exists, then $\mathbb{E}(\boldsymbol{X}) = \boldsymbol{\mu}$, and if $\mathbb{E}(\boldsymbol{W}) < \infty$, then $\mathrm{cov}(\boldsymbol{X}) = \mathbb{E}(\sqrt{\boldsymbol{W}}^{\mathrm{D}} \Sigma \sqrt{\boldsymbol{W}}^{\mathrm{D}})$. Furthermore, $\mathrm{corr}(\boldsymbol{X}) = P$, where P denotes the correlation matrix corresponding to Σ. If $A = I_d$, the (uncorrelated) components of $\boldsymbol{X}$ are independent if and only if all components of $\boldsymbol{W}$ are constant with probability 1 and thus if $\boldsymbol{X}$ is multivariate normal; see (McNeil et al. 2015, Lemma 6.5). Assuming $k = d$, the matrix A is typically the Cholesky factor computed from a given Σ. Other decompositions of Σ into $AA^\top$ for some $A \in \mathbb{R}^{d \times d}$ can be obtained from the eigendecomposition or singular-value decomposition.

2.1.2. Relationship with Elliptical Distributions

It is well known that normal variance mixtures, such as the multivariate normal and t distributions, are elliptical. A d-dimensional random vector $\boldsymbol{Y}$ is said to have an elliptical distribution, denoted by $\boldsymbol{Y} \sim \mathrm{ELL}_d(\boldsymbol{\mu}, \Sigma, F_R)$, if

$$\boldsymbol{Y} \overset{\mathrm{d}}{=} \boldsymbol{\mu} + RA\boldsymbol{S}, \tag{4}$$

where $\boldsymbol{\mu} \in \mathbb{R}^d$, $AA^\top = \Sigma$ and $\boldsymbol{S} \sim \mathrm{U}(\mathbb{S}^{d-1})$ independent of a non-negative rv $R \sim F_R$. Here, $\mathbb{S}^{d-1} = \{\boldsymbol{x} \in \mathbb{R}^d : \boldsymbol{x}^\top \boldsymbol{x} = 1\}$ denotes the unit sphere in $\mathbb{R}^d$.

Let $\boldsymbol{Y} \sim \mathrm{NVM}_d(\boldsymbol{\mu}, \Sigma, F_W)$. Using the fact that $\boldsymbol{Z}/\sqrt{\boldsymbol{Z}^\top \boldsymbol{Z}} \sim \mathrm{U}(\mathbb{S}^{d-1})$ is independent of $\boldsymbol{Z}^\top \boldsymbol{Z} \sim \chi^2_d$ for $\boldsymbol{Z} \sim \mathrm{N}_d(\boldsymbol{0}, I_d)$ (see, e.g., (Devroye 1986, Chapter 5)) it is easy to see from (4) that $\boldsymbol{Y} \sim \mathrm{ELL}_d(\boldsymbol{\mu}, \Sigma, F_R)$ for $R = W\tilde{X}$ where $W \sim F_W$ is independent of $\tilde{X}^2 \sim \chi^2_d$.

If $\boldsymbol{X} \sim \mathrm{gNVM}_d(\boldsymbol{0}, I_d, F_W)$, then $\boldsymbol{X}$ is in general not elliptical, unless $F_{W_1} = \cdots = F_{W_d}$. This can be seen from (4), since the scalar radial rv R cannot be used to model comonotone shocks. Applying the same principle used to define grouped normal variance mixtures in (2), one can define *grouped elliptical distributions* via the stochastic representation

$$\boldsymbol{X} \overset{\mathrm{d}}{=} \boldsymbol{\mu} + \mathrm{Diag}(\boldsymbol{R}) A \boldsymbol{S}, \tag{5}$$

where $\boldsymbol{R} = (R_1, \dots, R_d)$ satisfies that $R_j \overset{\mathrm{d}}{=} F^{\leftarrow}_{R_j}(U) \times F_W(\tilde{U})$ for $U, \tilde{U} \overset{\text{ind.}}{\sim} \mathrm{U}(0,1)$ and $R_j \geq 0$ a.s.

If $\boldsymbol{X} \sim \mathrm{gNVM}_d(\boldsymbol{\mu}, \Sigma, F_{\boldsymbol{W}})$, we can set $F_{R_j} = F_{W_j}$ for $j = 1, \dots, d$ and F_W to be the distribution function of a χ_d random variable (the square root of a χ^2_d random variable). This shows that $\boldsymbol{X}$ is a grouped elliptical distribution in the sense of (5). In this work we focus on the more tractable class of grouped NVM distributions and do not further detail grouped elliptical distributions.

2.2. Randomized quasi-Monte Carlo Methods

Many quantities of interest in this paper, such as the distribution function of a gNVM distribution, can, after a suitable transformation, be expressed as

$$\mu = \int_{(0,1)^d} g(\boldsymbol{u})\,\mathrm{d}\boldsymbol{u}, \tag{6}$$

where $d \in \mathbb{N}$ can be large (e.g., $\mu = F(\boldsymbol{a}, \boldsymbol{b})$ for large d; see (15) in the next section) or small (e.g., $d = 3$ in (22)) and the integral cannot be computed explicitly. As a method that works flexibly for small and large dimensions, one might consider Monte Carlo (MC) estimation, that is, estimate μ by

$$\hat{\mu}_n^{\mathrm{MC}} = \frac{1}{n}\sum_{i=1}^n g(\boldsymbol{U}_i), \quad \boldsymbol{U}_1, \dots, \boldsymbol{U}_n \overset{\text{ind.}}{\sim} \mathrm{U}(0,1)^d,$$

whose asymptotic $(1-\alpha)$-confidence interval (CI) can be approximated for sufficiently large n by

$$\left[\hat{\mu}_n^{\mathrm{MC}} - z_{1-\alpha/2}\hat{\sigma}_g/\sqrt{n},\ \hat{\mu}_n^{\mathrm{MC}} + z_{1-\alpha/2}\hat{\sigma}_g/\sqrt{n}\right],$$

where $z_\alpha = \Phi^{-1}(\alpha)$ and $\hat{\sigma}_g^2 = \widehat{\mathrm{var}}(g(\boldsymbol{U})) = \sum_{i=1}^n (g(\boldsymbol{U}_i) - \hat{\mu}_n^{\mathrm{MC}})^2/(n-1)$. An often superior alternative to MC estimation are quasi-Monte Carlo (QMC) methods. Instead of averaging function values of a random sample $\boldsymbol{U}_1, \dots, \boldsymbol{U}_n$, a low discrepancy point-set $P_n = \{\boldsymbol{v}_1, \dots, \boldsymbol{v}_n\} \subset [0,1)^d$ is employed, which aims at filling the unit hypercube in a more homogeneous way. To make error estimation easily possible, one can randomize the point-set P_n in a way such that the points in the resulting point set, say $\tilde{P}_n$, are uniformly distributed over $[0,1]^d$ while keeping the low discrepancy of the point set overall, giving rise to randomized QMC (RQMC) methods. In our algorithms, we use a digitally-shifted Sobol' sequence as implemented in the function `sobol(, randomize = "digital.shift")` of the R package `qrng`; see Hofert and Lemieux (2019). We remark that generating $\tilde{P}_n$ is even slightly faster than the Mersenne Twister, which is R's default (pseudo-)random number generator.

Given B independently randomized copies of P_n, say $\tilde{P}_{n,b} = \{\boldsymbol{u}_{1,b}, \dots, \boldsymbol{u}_{n,b}\}$ for $b = 1, \dots, B$, we construct B independent RQMC estimators of the form

$$\hat{\mu}_{b,n}^{\mathrm{RQMC}} = \frac{1}{n}\sum_{i=1}^n g(\boldsymbol{u}_{i,b}), \quad b = 1, \dots, B, \tag{7}$$

which are combined into the RQMC estimator.

$$\hat{\mu}_n^{\mathrm{RQMC}} = \frac{1}{B}\sum_{b=1}^B \hat{\mu}_{b,n}^{\mathrm{RQMC}} \tag{8}$$

of μ. An approximate $(1-\alpha)$-CI for μ can be estimated as

$$\left[\hat{\mu}_n^{\mathrm{RQMC}} - z_{1-\alpha/2}\hat{\sigma}_{\hat{\mu}^{\mathrm{RQMC}}}/\sqrt{B},\ \hat{\mu}_n^{\mathrm{RQMC}} + z_{1-\alpha/2}\hat{\sigma}_{\hat{\mu}_n^{\mathrm{RQMC}}}/\sqrt{B}\right], \tag{9}$$

where $\hat{\sigma}^2_{\hat{\mu}^{\mathrm{RQMC}}} = \sum_{i=1}^B (\hat{\mu}_{b,n}^{\mathrm{RQMC}} - \hat{\mu}_n^{\mathrm{RQMC}})^2/(B-1)$ is an unbiased estimator of $\mathrm{var}(\hat{\mu}_{b,n}^{\mathrm{RQMC}})$. One can compute $\hat{\mu}_n^{\mathrm{RQMC}}$ from (8) for some initial sample size n (e.g., $n = 2^7$) and iteratively increase the sample size of each $\hat{\mu}_{b,n}^{\mathrm{RQMC}}$ in (7) until the length of the CI in (9) satisfies a pre-specified error tolerance. In our implementations, we use $B = 15$, an absolute default error tolerance $\varepsilon = 0.001$ (which can be changed by the user) and $z_{1-\alpha/2} = 3.5$ (so $\alpha \approx 0.00047$). By using $\hat{\mu}_n^{\mathrm{RQMC}}$ as approximation for the true value of μ, one can also consider relative instead of absolute errors.

Sometimes it is necessary to estimate $\log(\mu)$ rather than μ; in particular, when μ is small. For instance, if $\mu = f(\boldsymbol{x})$ where $f(\boldsymbol{x})$ is the density of some random vector evaluated at $\boldsymbol{x} \in \mathbb{R}^d$, interest may lie in $\log(f(\boldsymbol{x}))$ as this quantity is needed to compute the log-likelihood of a random sample (which then may be optimized over some parameter space). When μ is small, it is typically not a good

idea to use $\log(\mu) \approx \log(\hat{\mu}_n^{\text{RQMC}})$ directly, but rather to compute a numerically more robust estimator for $\log(\mu)$, a *proper logarithm*. Define the function LSE (for Logarithmic Sum of Exponentials) as

$$\text{LSE}(c_1, \dots, c_n) = \log\left(\sum_{i=1}^{n} \exp(c_i)\right) = c_{\max} + \log\left(\sum_{i=1}^{n} \exp(c_i - c_{\max})\right),$$

where $c_1, \dots, c_n \in \mathbb{R}$ and $c_{\max} = \max\{c_1, \dots, c_n\}$.

The sum inside the logarithm on the right side of this equation is bounded between 1 and n so that the right side of this equation is numerically more robust than the left side.

Let $c_{i,b} = \log(g(u_{i,b}))$ for $i = 1, \dots, n$ and $b = 1, \dots, B$. An estimator numerically superior to $\log(\hat{\mu}_n^{\text{RQMC}})$ (but mathematically equivalent) is given by

$$\hat{\mu}_{n,\log}^{\text{RQMC}} = -\log(B) + \text{LSE}(\hat{\mu}_{1,n,\log}^{\text{RQMC}}, \dots, \hat{\mu}_{B,n,\log}^{\text{RQMC}}), \tag{10}$$

where

$$\hat{\mu}_{b,n,\log}^{\text{RQMC}} = -\log(n) + \text{LSE}(c_{1,b}, \dots, c_{n,b}), \quad b = 1, \dots, B.$$

Using

$$\hat{\sigma}_{\hat{\mu}^{\text{RQMC}},\log} = \sqrt{\frac{1}{(B-1)} \sum_{b=1}^{B} \left(\hat{\mu}_{b,n,\log}^{\text{RQMC}} - \hat{\mu}_{n,\log}^{\text{RQMC}}\right)^2}$$

the integration error can be estimated from the length of the CI in (9) as before.

For more details about (randomized) quasi-Monte Carlo methods and their applications in the financial literature, see, e.g., Niederreiter (1992), Lemieux (2009), Glasserman (2013).

3. Distribution Function

Let $-\infty \le \boldsymbol{a} < \boldsymbol{b} \le \infty$ componentwise (entries $\pm\infty$ to be interpreted as the corresponding limits). Then $F(\boldsymbol{a}, \boldsymbol{b}) = \mathbb{P}(\boldsymbol{a} < \boldsymbol{X} \le \boldsymbol{b})$ is the probability that the random vector $\boldsymbol{X}$ falls in the hyper-rectangle spanned by the lower-left and upper-right endpoints $\boldsymbol{a}$ and $\boldsymbol{b}$, respectively. If $\boldsymbol{a} = (-\infty, \dots, -\infty)$, we recover $F(\boldsymbol{a}, \boldsymbol{x}) = F(\boldsymbol{x}) = \mathbb{P}(X_1 \le x_1, \dots, X_d \le x_d)$ which is the (cumulative) distribution function of $\boldsymbol{X}$.

Assume wlog that $\boldsymbol{\mu} = \boldsymbol{0}$, otherwise adjust $\boldsymbol{a}, \boldsymbol{b}$ accordingly. Then

$$\begin{aligned} F(\boldsymbol{a}, \boldsymbol{b}) &= \mathbb{P}(\boldsymbol{a} < \sqrt{\boldsymbol{W}}^{\text{D}} A\boldsymbol{Z} \le \boldsymbol{b}) = \mathbb{E}\Big[\mathbb{P}\big((1/\sqrt{w})^{\text{D}}(U)\, \boldsymbol{a} < A\boldsymbol{Z} \le (1/\sqrt{w})^{\text{D}}(U)\, \boldsymbol{b} \mid U\big)\Big] \\ &= \mathbb{E}\Big[\Phi_\Sigma\big((1/\sqrt{w})^{\text{D}}(U)\, \boldsymbol{a}, (1/\sqrt{w})^{\text{D}}(U)\, \boldsymbol{b}\big)\Big] = \int_0^1 \Phi_\Sigma\Big((1/\sqrt{w})^{\text{D}}(u)\, \boldsymbol{a}, (1/\sqrt{w})^{\text{D}}(u)\, \boldsymbol{b}\Big)\, \mathrm{d}u, \end{aligned} \tag{11}$$

where $\Phi_\Sigma(\boldsymbol{a}, \boldsymbol{b}) = \mathbb{P}(\boldsymbol{a} < \boldsymbol{Y} \le \boldsymbol{b})$ for $\boldsymbol{Y} \sim \text{N}_d(\boldsymbol{0}, \Sigma)$. Note that the function $\Phi_\Sigma(\boldsymbol{a}, \boldsymbol{b})$ itself is a d-dimensional integral for which no closed formula exists and is typically approximated via numerical methods; see, e.g., Genz (1992).

Comonotonicity of the W_j allowed us to write $F(\boldsymbol{a}, \boldsymbol{b})$ as a $(d+1)$-dimensional integral; had the W_j a different dependence structure, this convenience would be lost and the resulting integral in (11) could be up to $2d$-dimensional (e.g., when all W_j are independent).

3.1. Estimation

As demonstrated in Section 2.1, we need to approximate

$$F(\boldsymbol{a}, \boldsymbol{b}) = \int_0^1 \Phi_\Sigma\Big((1/\sqrt{w})^{\text{D}}(u)\, \boldsymbol{a}, (1/\sqrt{w})^{\text{D}}(u)\, \boldsymbol{b}\Big)\, \mathrm{d}u.$$

In Hintz et al. (2020), randomized quasi-Monte Carlo methods have been derived to approximate the distribution function of a normal variance mixture $\boldsymbol{X} \sim \mathrm{NVM}_d(\boldsymbol{\mu}, \Sigma, F_W)$ from (1). Grouped normal variance mixtures can be dealt with similarly, thanks to the comonotonicity of the mixing random variables in $\boldsymbol{W}$.

In order to apply RQMC to the problem of estimating $F(\boldsymbol{a}, \boldsymbol{b})$, we need to transform $F(\boldsymbol{a}, \boldsymbol{b})$ to an integral over the unit hypercube. To this end, we first address Φ_Σ. Let $C = (C_{ij})_{i,j=1}^d$ be the Cholesky factor of Σ (a lower triangular matrix such that $CC^\top = \Sigma$). We assume that Σ has full rank which implies $C_{jj} > 0$ for $j = 1, \dots, d$. Genz (1992) (see also Genz and Bretz (1999 2002 2009)) uses a series of transformations, relying on C being a lower triangular matrix, to write

$$\begin{aligned} \Phi_\Sigma(\boldsymbol{a}, \boldsymbol{b}) &= \int_{a_1}^{b_1} \cdots \int_{a_d}^{b_d} \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \exp\left(-\frac{\boldsymbol{x}^\top \Sigma^{-1} \boldsymbol{x}}{2}\right) \mathrm{d}\boldsymbol{x} \\ &= (\hat{e}_1 - \hat{d}_1) \int_0^1 (\hat{e}_2 - \hat{d}_2) \cdots \int_0^1 (\hat{e}_d - \hat{d}_d) \, \mathrm{d}u_{d-1} \dots \mathrm{d}u_1, \end{aligned} \tag{12}$$

where the $\hat{d}_i$ and $\hat{e}_i$ are recursively defined via

$$\hat{e}_1 = \Phi\left(\frac{b_1}{C_{11}}\right), \quad \hat{e}_i = \hat{e}_i(u_1, \dots, u_{i-1}) = \Phi\left(\frac{b_i - \sum_{j=1}^{i-1} C_{ij} \Phi^{-1}\left(\hat{d}_j + u_j(\hat{e}_j - \hat{d}_j)\right)}{C_{ii}}\right), \tag{13}$$

and $\hat{d}_i$ is $\hat{e}_i$ with b_i replaced by a_i for $i = 1, \dots, d$. Note that the final integral in (12) is $(d-1)$-dimensional.

Combining the representation (12) of Φ_Σ with Equation (11) yields

$$F(\boldsymbol{a}, \boldsymbol{b}) = \int_{(0,1)^d} g(\boldsymbol{u}) \, \mathrm{d}\boldsymbol{u} = \int_0^1 g_1(u_0) \int_0^1 g_2(u_0, u_1) \cdots \int_0^1 g_d(u_0, \dots, u_{d-1}) \, \mathrm{d}u_{d-1} \dots \mathrm{d}u_0, \tag{14}$$

where

$$g(\boldsymbol{u}) = \prod_{i=1}^d g_i(u_0, \dots, u_{i-1}), \quad g_i(u_0, \dots, u_{i-1}) = e_i - d_i, \quad i = 1, \dots, d, \tag{15}$$

for $\boldsymbol{u} = (u_0, u_1, \dots, u_{d-1}) \in (0,1)^d$. The e_i are recursively defined by

$$\begin{aligned} e_1 &= e_1(u_0) = \Phi\left(\frac{b_1}{C_{11}\sqrt{F_{W_1}^{\leftarrow}(u_0)}}\right), \\ e_i &= e_i(u_0, \dots, u_{i-1}) = \Phi\left(\frac{1}{C_{ii}}\left(\frac{b_i}{\sqrt{F_{W_i}^{\leftarrow}(u_0)}} - \sum_{j=1}^{i-1} C_{ij} \Phi^{-1}(d_j + u_j(e_j - d_j))\right)\right), \end{aligned} \tag{16}$$

for $i = 2, \dots, d$ and the d_i are e_i with b_i replaced by a_i for $i = 1, \dots, d$.

Summarizing, we were able to write $F(\boldsymbol{a}, \boldsymbol{b})$ as an integral over the d-dimensional unit hypercube. Our algorithm to approximate $F(\boldsymbol{a}, \boldsymbol{b})$ consists of two steps:

First, a greedy re-ordering algorithm is applied to the inputs $\boldsymbol{a}$, $\boldsymbol{b}$, Σ. It re-orders the components $1, \dots, d$ of $\boldsymbol{a}$ and $\boldsymbol{b}$ as well as the corresponding rows and columns in Σ in a way that the expected ranges of g_i in (15) are increasing with the index i for $i = 1, \dots, d$. Observe that the integration variable u_i is present in all remaining $d - i + 1$ integrals in (14) whose ranges are determined by the ranges of $g_1, \dots, g_i$; reordering the variables according to expected ranges therefore (in the vast majority of cases) reduces the overall variability of g (namely, $\mathrm{var}(g(\boldsymbol{U}))$ for $\boldsymbol{U} \sim \mathrm{U}(0,1)^d$). Reordering also makes the first variables "more important" than the last ones, thereby reducing the effective dimension of the integrand. This is particularly beneficial for quasi-Monte Carlo methods, as these methods are known to perform well in high-dimensional problems with low

effective dimension; see, e.g., Caflisch et al. (1997), Wang and Sloan (2005). For a detailed description of the method, see (Hintz et al. 2020, Algorithm 3.2) (with $a_j/\mu_{\sqrt{W}}$ replaced by $a_j/\mu_{\sqrt{W_j}}$ and similarly for b_j for $j = 1,\dots,d$ to account for the generalization); similar reordering strategies have been proposed in Gibson et al. (1994) for calculating multivariate normal and in Genz and Bretz (2002) for multivariate t probabilities.

Second, an RQMC algorithm as described in Section 2.2 is applied to approximate the integral in (14) with re-ordered $\boldsymbol{a}, \boldsymbol{b}, \Sigma$ and F_W. Instead of integrating g from (15) directly, antithetic variates are employed so that effectively, the function $\tilde{g}(\boldsymbol{u}) = (g(\boldsymbol{u}) + g(\boldsymbol{1} - \boldsymbol{u}))/2$ is integrated.

The algorithm to estimate $F(\boldsymbol{a}, \boldsymbol{b})$ just described is implemented in the function `pgnvmix()` of the R package `nvmix`.

3.2. Numerical Results

In order to assess the performance of our algorithm described in Section 3.1, we estimate the error as a function of the number of function evaluations. Three estimators are considered. First, the "Crude MC" estimator is constructed by sampling $\boldsymbol{X}_1, \dots, \boldsymbol{X}_n \overset{\text{ind.}}{\sim} \text{gNVM}(\boldsymbol{\mu}, \Sigma, F_W)$ and estimating $\mathbb{P}(\boldsymbol{X} \le \boldsymbol{x})$ by $\hat{\mu}_n^{\text{MC}} = (1/n)\sum_{i=1}^n \mathbb{1}_{\{\boldsymbol{X}_i \le \boldsymbol{x}\}}$. The second and third estimator are based on the integrand g from (15), which is integrated once using MC ("g (MC)") and once using a randomized Sobol' sequence ("g (sobol)"). In either case, variable reordering is applied first.

We perform our experiments for an inverse-gamma mixture. As motivated in the introduction, an important special case of (grouped) normal variance mixtures is obtained when the mixing distribution is inverse-gamma. In the ungrouped case when $\boldsymbol{X} \sim \text{NVM}_d(\boldsymbol{\mu}, \Sigma, F_W)$ with $W \sim \text{IG}(\nu/2, \nu/2)$, the distribution of $\boldsymbol{X}$ is multivariate t (notation $\boldsymbol{X} \sim t_d(\nu, \boldsymbol{\mu}, \Sigma)$) with density

$$f^t_{\nu,\boldsymbol{\mu},\Sigma}(\boldsymbol{x}) = \frac{\Gamma((\nu+d)/2)}{\Gamma(\nu/2)\sqrt{(\nu\pi)^d|\Sigma|}}\left(1 + \frac{D^2(\boldsymbol{x};\boldsymbol{\mu},\Sigma)}{\nu}\right)^{-\frac{\nu+d}{2}}, \quad \boldsymbol{x} \in \mathbb{R}^d. \tag{17}$$

The distribution function of $\boldsymbol{X} \sim t_d(\nu, \boldsymbol{\mu}, \Sigma)$ does not admit a closed form; estimation of the latter was discussed for instance in Genz and Bretz (2009), Hintz et al. (2020), Cao et al. (2020). The same holds for a grouped inverse-gamma mixture model. If $W_j \sim \text{IG}(\nu_j/2, \nu_j/2)$ for $j = 1,\dots,d$, the random vector $\boldsymbol{X}$ follows a grouped t distribution, denoted by $\boldsymbol{X} \sim \text{gt}_d(\nu_1,\dots,\nu_d;\boldsymbol{\mu},\Sigma)$ or by $\boldsymbol{X} \sim \text{gt}_d(\boldsymbol{\nu},\boldsymbol{\mu},\Sigma)$ for $\boldsymbol{\nu} = (\nu_1,\dots,\nu_d)$. If $1 < S < d$, denote by $d_1,\dots,d_S$ the group sizes. In this case, we use the notation $\boldsymbol{X} \sim \text{gt}_d(\nu_1,\dots,\nu_S;d_1,\dots,d_S;\boldsymbol{\mu},\Sigma)$ or $\boldsymbol{X} \sim \text{gt}_d(\boldsymbol{\nu},\boldsymbol{d},\boldsymbol{\mu},\Sigma)$ for $\boldsymbol{d} = (d_1,\dots,d_S)$. If $S = 1$, it follows that $\boldsymbol{X} \sim t_d(\nu_1,\boldsymbol{\mu},\Sigma)$.

For our numerical examples to test the performance of our procedure for estimating $F(\boldsymbol{a}, \boldsymbol{b})$, assume $\boldsymbol{X} \sim \text{gt}_d(\boldsymbol{\nu}, \boldsymbol{0}, P)$ for a correlation matrix P. We perform the experiment in $d = 5$ with $\boldsymbol{\nu} = (1.5, 2.5, \dots, 5.5)$ and in $d = 20$ with $\boldsymbol{\nu} = (1, 1.25, \dots, 5.5, 5.75)$. The following is repeated 15 times: Sample an upper limit $\boldsymbol{b} \sim \text{U}(0, 3\sqrt{d})^d$ and a correlation matrix P (sampled based on a random Wishart matrix via the function `rWishart() in` R). Then estimate $\mathbb{P}(\boldsymbol{X} \le \boldsymbol{b})$ using the three aforementioned methods using various sample sizes and estimate the error for the MC estimators based on a CLT argument and for the RQMC estimator as described in Section 2.2. Figure 1 reports the average absolute errors for each sample size over the 15 runs.

Convergence speed as measured by the regression coefficient α of $\log(\hat{\varepsilon}) = \alpha \log(n) + c$ where $\hat{\varepsilon}$ is the estimated error are displayed in the legend. As expected, the MC estimators have an overall convergence speed of $1/\sqrt{n}$; however, the crude estimator has a much larger variance than the MC estimator based on the function g. The RQMC estimator ("g (sobol)") not only shows much faster convergence speed than its MC counterparts, but also a smaller variance.

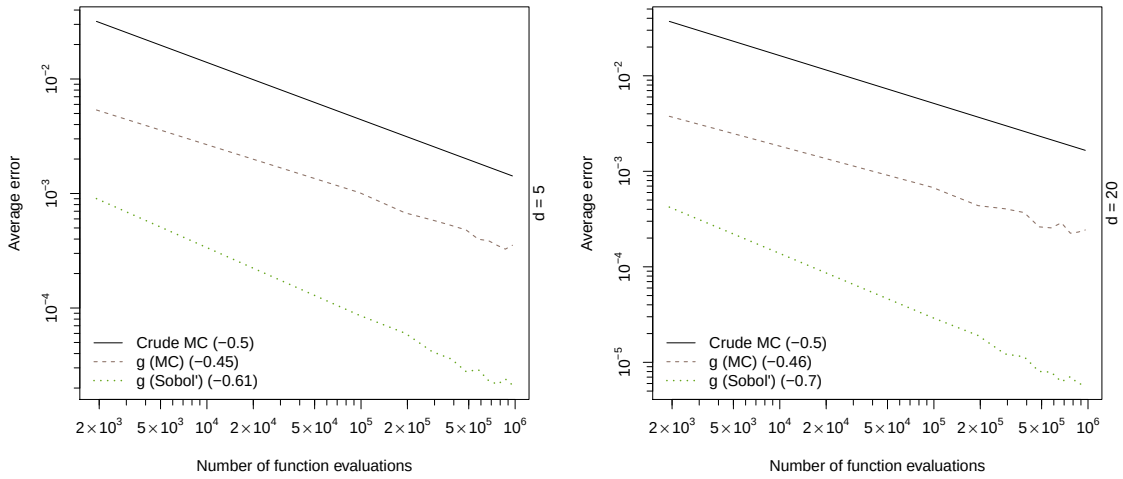

Figure 1. Estimated average errors over 15 replications. In each run and for each sample size, $\mathbb{P}(\boldsymbol{X} \leq \boldsymbol{b})$ is estimated where $\boldsymbol{X} \sim \text{gNVM}_d(\boldsymbol{0}, P, F_{\boldsymbol{W}})$ for a random correlation matrix P and a random upper limit $\boldsymbol{b}$. For $d = 5$ (**left**), $X_j \sim t_{0.5+j}$ and for $d = 20$ (**right**), $X_j \sim t_{0.75+0.25j}$ for $j = 1, \dots, d$.

4. Density Function

Let us now focus on the density of $\boldsymbol{X} \sim \text{gNVM}(\boldsymbol{\mu}, \Sigma, F_{\boldsymbol{W}})$, where we assume that Σ has full rank in order for the density to exist. As mentioned in the introduction, the density of $\boldsymbol{X}$ is typically not available in closed form, not even in the case of a grouped t distribution. The same conditioning argument used to derive (11) yields that the density of $\boldsymbol{X} \sim \text{gNVM}_d(\boldsymbol{\mu}, \Sigma, F_{\boldsymbol{W}})$ evaluated at $\boldsymbol{x} \in \mathbb{R}^d$ can be written as

$$\begin{aligned} f_{\boldsymbol{X}}(\boldsymbol{x}) &= \mathbb{E}\left(\frac{1}{\sqrt{(2\pi)^d |\sqrt{\boldsymbol{W}}^{\mathrm{D}}(U)\Sigma\sqrt{\boldsymbol{W}}^{\mathrm{D}}(U)|}} \exp\left(-\frac{D^2\left(\boldsymbol{x}; \boldsymbol{\mu}, \sqrt{\boldsymbol{W}}^{\mathrm{D}}(U)\Sigma\sqrt{\boldsymbol{W}}^{\mathrm{D}}(U)\right)}{2} \right) \right) \\ &= \int_0^1 \frac{1}{\sqrt{(2\pi)^d |\Sigma| \prod_{i=1}^d F_{W_i}^{\leftarrow}(u)}} \exp\left(-\frac{D^2\left(\boldsymbol{x}; \boldsymbol{\mu}, \sqrt{\boldsymbol{W}}^{\mathrm{D}}(u)\Sigma\sqrt{\boldsymbol{W}}^{\mathrm{D}}(u)\right)}{2} \right) \mathrm{d}u = \int_0^1 h(u)\,\mathrm{d}u, \end{aligned} \tag{18}$$

where $D^2(\boldsymbol{x}; \boldsymbol{\mu}, \Sigma) = (\boldsymbol{x} - \boldsymbol{\mu})^\top \Sigma^{-1} (\boldsymbol{x} - \boldsymbol{\mu})$ denotes the (squared) Mahalanobis distance of $\boldsymbol{x} \in \mathbb{R}^d$ from $\boldsymbol{\mu}$ with respect to Σ and the integrand $h(u)$ is defined in an obvious manner. Except for some special cases (e.g., when all W_j are inverse-gamma with the same parameters), this integral cannot be computed explicitly, so that we rely on numerical approximation thereof.

4.1. Estimation

From (18), we find that computing the density $f(\boldsymbol{x})$ of $\boldsymbol{X} \sim \text{gNVM}_d(\boldsymbol{\mu}, \Sigma, F_{\boldsymbol{W}})$ evaluated at $\boldsymbol{x} \in \mathbb{R}^d$ requires the estimation of a univariate integral. As interest often lies in the logarithmic density (or log-density) rather than the actual density (e.g., likelihood-based methods where the log-likelihood function of a random sample is optimized over some parameter space), we directly consider the problem of estimating $\log(\mu)$ for $\mu = \int_0^1 h(u)\,\mathrm{d}u$ with h given in (18).

Since μ is expressed as an integral over $(0, 1)$, RQMC methods to estimate $\log(\mu)$ from Section 2.2 can be applied directly to the problem in this form. If the log-density needs to be evaluated at several $\boldsymbol{x}_1, \dots, \boldsymbol{x}_N$, one can use the same point-sets $\tilde{P}_{n,b}$ and therefore the same realizations of the mixing random vector $\boldsymbol{W}$ for all inputs. This avoids costly evaluations of the quantile functions $F_{W_j}^{\leftarrow}$.

Estimating $\log(f(\boldsymbol{x}))$ via RQMC as just described works well for input $\boldsymbol{x}$ of moderate size, but deteriorates if $\boldsymbol{x}$ is far away from the mean. To see this, Figure 2 shows the integrand h for three

different input $\boldsymbol{x}$ and three different settings for F_W. If $\boldsymbol{x}$ is "large", most of the mass is contained in a small subdomain of $(0,1)$ containing the abscissa of the maximum of h. If an integration routine is not able to detect this peak, the density is substantially underestimated. Further complication arises as we are estimating the log-density rather than the density. Unboundedness of the natural logarithm at 0 makes estimation of $\log(\mu)$ for small μ challenging, both from a theoretical and a computational point of view due to finite machine precision.

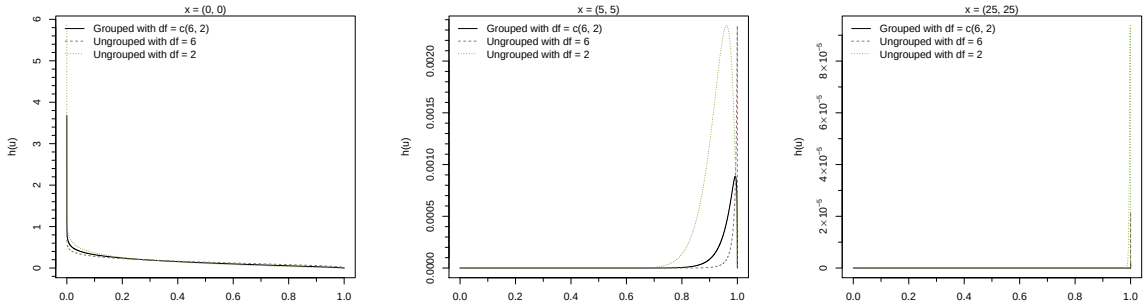

Figure 2. Integrand values h for a 2-dimensional distribution with t margins and $\boldsymbol{x} = (0,0)$ (**left**), $\boldsymbol{x} = (5,5)$ (**middle**) and $\boldsymbol{x} = (25,25)$ (**right**).

In (Hintz et al. 2020, Section 4), an adaptive RQMC algorithm is proposed to efficiently estimate the log-density of $\boldsymbol{X} \sim \mathrm{NVM}_d(\boldsymbol{\mu}, \Sigma, F_W)$. We generalize this method to the grouped case. The grouped case is more complicated because the distribution is not elliptical, hence the density does not only depend on $\boldsymbol{x}$ through $D^2(\boldsymbol{x}, \boldsymbol{\mu}, \Sigma)$. Furthermore, the height of the (unique) maximum of h in the ungrouped case can be easily computed without simulation, which helps the adaptive procedure find the relevant region; in the grouped case, the value of the maximum is usually not available. Lastly, S (as opposed to 1) quantile evaluations are needed to obtain one function value $h(u)$; from a run time perspective, evaluating these quantile functions is the most expensive part.

If $\boldsymbol{x}$ is "large", the idea is to apply RQMC only in a relevant region (u_l, u_r) with $\operatorname{argmax}_u h(u) =: u^* \in (u_l, u_r)$. More precisely, given a threshold ε_{th} with $0 < \varepsilon_{\text{th}} < h_{\max} = \max_{u \in (0,1)} h(u)$, choose u_l, u_r (l for "left" and r for "right") with $0 \le u_l \le u^* \le u_r \le 1$ so that $h(u) > \varepsilon_{\text{th}}$ if and only if $u \in (u_l, u_r)$. For instance, take

$$\varepsilon_{\text{th}} = 10^{\log(h_{\max})/\log(10) - k_{\text{th}}} \tag{19}$$

with $k_{\text{th}} = 10$ so that ε_{th} is 10 orders smaller than $h_{\max}$.

One can then apply RQMC (with a proper logarithm) in the region (u_l, u_r) (by replacing every $u_{i,b} \in (0,1)$ by $u'_{i,b} = u_l + (u_r - u_l)u_{i,b} \in (u_l, u_r)$), producing an estimate for $\log \int_{u_l}^{u_r} h(u)\mathrm{d}u$. By construction, the remaining regions do not contribute significantly to the overall integral anyway, so that a rather quick integration routine suffices here. Note that neither $h_{\max}$, nor u_l, u_r are known explicitly. However, $h_{\max}$ can be estimated from pilot-runs and u_l, u_r can be approximated using bisections.

Summarizing, we propose the following method to estimate $\log(f(\boldsymbol{x}_i))$, $i = 1, \dots, N$, for given inputs $\boldsymbol{x}_1, \dots, \boldsymbol{x}_N$ and error tolerance ε.

This algorithm is implemented in the function `dgnvmix(, log = TRUE)` in the R package `nvmix`, which by default uses a relative error tolerance.

The advantage of the proposed algorithm is that only little run time is spent on estimating "easy" integrals, thanks to the pilot run in Step 1. If $n_0 = 2^{10}$ and $B = 15$ (the current default in the `nvmix` package), this step gives 15 360 pairs $(u, F_{\boldsymbol{W}}^{\leftarrow}(u))$. These pairs give good starting values for the bisections to find u_l, u_r. Note that no additional quantile evaluations are needed to estimate the less important regions $(0, u_l)$ and $(u_r, 1)$.

4.2. Numerical Results

Luo and Shevchenko (2010) are faced with almost the same integration problem when estimating the density of a bivariate grouped t copula. They use a globally adaptive integration scheme from Piessens et al. (2012) to integrate h. While this procedure works well for a range of inputs, it deteriorates for input $\boldsymbol{x}$ with large components.

Consider first $\boldsymbol{X} \sim t_d(\nu, \boldsymbol{0}, I_d)$ and recall that the density of $\boldsymbol{X}$ is known and given by (17); this is useful to test our estimation procedure. As such, let $\boldsymbol{X} \sim t_2(\nu = 6, \boldsymbol{0}, I_2)$ and consider the problem of evaluating the density of $\boldsymbol{X}$ at $\boldsymbol{x} \in \{(0,0),(5,5),(25,25),(50,50)\}$. Some values of the corresponding integrands are shown in Figure 2. In Table 1, true and estimated (log-)density values are reported; once estimated using the R function `integrate()`, which is based on the QUADPACK package of Piessens et al. (2012) and once using `dgnvmix()`, which is based on Algorithm 1. Clearly, the `integrate()` integration routine is not capable of detecting the peak when input $\boldsymbol{x}$ is large, yielding substantially flawed estimates. The estimates obtained from `dgnvmix()`, however, are quite close to the true values even far out in the tail.

Algorithm 1: Adaptive RQMC Algorithm to Estimate $\log(f(\boldsymbol{x}_1)), \dots, \log(f(\boldsymbol{x}_n))$.

Given $\boldsymbol{x}_1, \dots, \boldsymbol{x}_N$, Σ, ε, ε_{th}, n_0, estimate $\log(f(\boldsymbol{x}_l))$, $l = 1, \dots, N$, via:

1. Compute $\hat{\mu}^{\text{RQMC}}_{\log f(\boldsymbol{x}_i), n_0}$ with sample size n_0 using the same random numbers for all input $\boldsymbol{x}_i$, $i = 1, \dots, N$. Store all uniforms with corresponding quantile evaluations $F^{\leftarrow}_{\boldsymbol{W}}$ in a list $\mathcal{L}$.
2. If all estimates $\hat{\mu}^{\text{RQMC}}_{\log f(\boldsymbol{x}_i), n_0}$, $i = 1, \dots, N$, meet the error tolerance ε, go to Step 4. Otherwise let $\boldsymbol{x}_s$, $s = 1, \dots, N'$ with $1 \le N' \le N$ be the inputs whose error estimates exceed the error tolerance.
3. For each remaining input $\boldsymbol{x}_s$, $s = 1, \dots, N'$, do:
   (a) Use all pairs $(u, F^{\leftarrow}_{\boldsymbol{W}}(u))$ in $\mathcal{L}$ to compute values of $h(u)$ and set $\hat{h}_{\max} = \max_{u \in \mathcal{L}} h(u)$. If the largest value of h is obtained for the largest (smallest) u in the list $\mathcal{L}$, set $u^* = 1$ ($u^* = 0$).
   (b) If $u^* = 1$, set $u_r = 1$ and if $u^* = 0$, set $u_l = 0$. Unless already specified, use bisections to find u_l and u_r such that $u_l < u^* < u_r$ and u_l (u_r) is the smallest (largest) u such that $h(u) > \varepsilon_{\text{th}}$ from (19) with $h_{\max}$ replaced by $\hat{h}_{\max}$. Starting intervals for the bisections can be found from the values in $\mathcal{L}$.
   (c) If $u_l > 0$, approximate $\log(\int_0^{u_l} h(u)\,\mathrm{d}u)$ using a trapezoidal rule with proper logarithm and knots $u'_1, \dots, u'_m$ where u'_i are those u's in $\mathcal{L}$ satisfying $u \le u_l$. Call the approximation $\hat{\mu}_{(0,u_l)}(\boldsymbol{x}_s)$. If $u_l = 0$, set $\hat{\mu}_{(0,u_l)} = -\infty$.
   (d) If $u_r < 1$, approximate $\log(\int_{u_r}^1 h(u)\,\mathrm{d}u)$ using a trapezoidal rule with proper logarithm and knots $u''_1, \dots, u''_p$ where u''_i are those u's in $\mathcal{L}$ satisfying $u \ge u_r$. Call the approximation $\hat{\mu}_{(u_r,1)}(\boldsymbol{x}_s)$. If $u_r = 0$, set $\hat{\mu}_{(u_r,1)}(\boldsymbol{x}_s) = -\infty$.
   (e) Estimate $\log(\int_{u_l}^{u_r} h(u)\,\mathrm{d}u)$ via RQMC. That is, compute $\hat{\mu}^{\text{RQMC}}_{\log f, n}$ from (10) where every $u_{i,b} \in (0,1)$ is replaced by $u'_{i,b} = u_l + (u_r - u_l)u_{i,b} \in (u_l, u_r)$. Increase n until the error tolerance ε is met. Then set $\hat{\mu}_{(u_l,u_r)} = \log(u_r - u_l) + \hat{\mu}^{\text{RQMC}}_{\log f, n}$ which estimates $\log(\int_{u_l}^{u_r} h(u)\,\mathrm{d}u)$.
   (f) Combine

$$\hat{\mu}^{\text{RQMC}}_{\log f(\boldsymbol{x}_s)} = \text{LSE}\left(\hat{\mu}_{(0,u_l)}(\boldsymbol{x}_s), \hat{\mu}_{(u_l,u_r)}(\boldsymbol{x}_s), \hat{\mu}_{(u_r,1)}(\boldsymbol{x}_s)\right)$$

4. Return $\hat{\mu}^{\text{RQMC}}_{\log f(\boldsymbol{x}_l)}$, $l = 1, \dots, N$.

Table 1. Estimated and true density values of a bivariate t distribution with 6 degrees of freedom. Logarithmic values are in brackets.

$\boldsymbol{x}$	**True Density**	**Estimated Density** (`integrate()`)	**Estimated Density** (`dgnvmix()`)
$(0,0)$	1.59×10^{-1} (-1.84)	1.59×10^{-1} (-1.84)	1.59×10^{-1} (-1.84)
$(5,5)$	2.10×10^{-5} (-10.77)	2.11×10^{-5} (-10.77)	2.16×10^{-5} (-10.74)
$(10,10)$	1.15×10^{-7} (-15.98)	4.47×10^{-8} (-16.92)	1.16×10^{-7} (-15.97)
$(25,25)$	8.29×10^{-11} (-23.21)	5.50×10^{-23} (-51.25)	8.00×10^{-11} (-23.25)
$(50,50)$	3.28×10^{-13} (-28.74)	3.26×10^{-76} (-173.82)	3.13×10^{-13} (-28.79)

The preceding discussion focused on the classical multivariate t setting, as the density is known in this case. Next, consider a grouped inverse-gamma mixture model and let $\boldsymbol{X} \sim \mathrm{gt}_d(\boldsymbol{\nu}, \boldsymbol{\mu}, \Sigma)$. The density $f^{\mathrm{gt}}_{\boldsymbol{\nu},\boldsymbol{\mu},P}$ of $\boldsymbol{X} \sim \mathrm{gt}_d(\boldsymbol{\nu}, \boldsymbol{\mu}, \Sigma)$ is not available in closed form, so that here we indeed need to rely on estimation of the latter. The following experiment is performed for $\boldsymbol{X} \sim \mathrm{gt}_2(\boldsymbol{\nu}, \mathbf{0}, I_2)$ with $\boldsymbol{\nu} = (3,6)$ and for $\boldsymbol{X} \sim \mathrm{gt}_{10}(\boldsymbol{\nu}, \mathbf{0}, I_{10})$ where $\boldsymbol{\nu} = (3,\dots,3,6,\dots,6)$ (corresponding to two groups of size 5 each). First, a sample from a more heavy tailed grouped t distribution of size 2500 is sampled (with degrees of freedom $\boldsymbol{\nu}' = (1,2)$ and $\boldsymbol{\nu}' = (1,\dots,1,2,\dots,2)$, respectively) and then the log-density function of $\boldsymbol{X} \sim \mathrm{gt}_d(\boldsymbol{\nu}, \mathbf{0}, I_d)$ is evaluated at the sample. The results are shown in Figure 3.

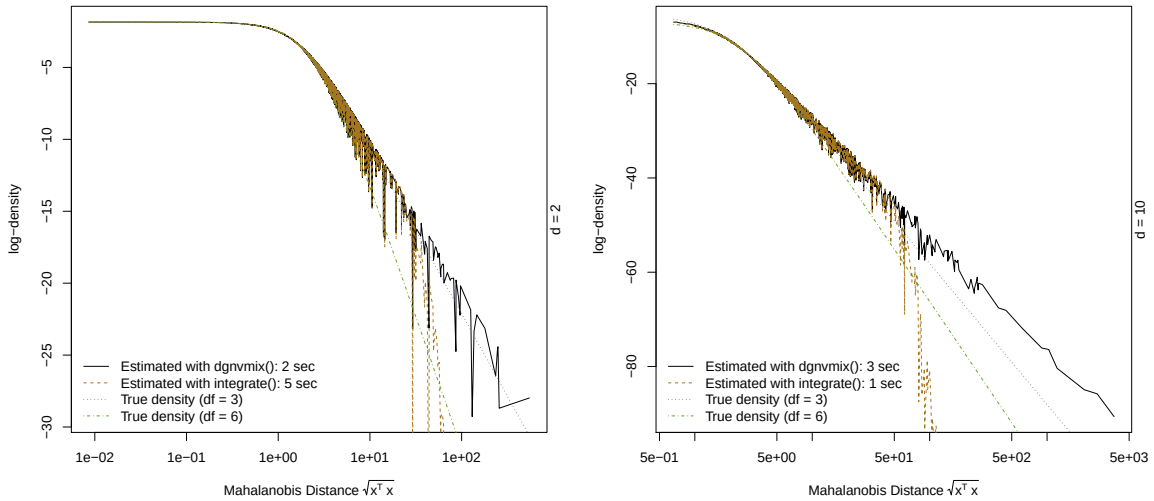

Figure 3. Estimated log-density of a grouped t distribution with $\boldsymbol{\nu} = (3,6)$ in $d = 2$ (**left**) and $\boldsymbol{\nu} = (3,\dots,3,6,\dots,6)$ in $d = 10$ (**right**). Estimation with `dgnvmix()` was carried out using a relative error tolerance of 0.01. The plot also shows the log-density function of $t_d(3, \mathbf{0}, I_d)$ and $t_d(6, \mathbf{0}, I_d)$ for comparison.

It is clear from the plots that `integrate()` again gives wrong approximations to $f(\boldsymbol{x})$ for input $\boldsymbol{x}$ far out in the tail; for small input $\boldsymbol{x}$, the results from `integrate()` and from `dgnvmix()` coincide. Furthermore, it can be seen that the density function is not monotonic in the Mahalanobis distance (as grouped normal mixtures are not elliptical anymore). The plot also includes the log-density functions of an ungrouped d-dimensional t distribution with degrees of freedom 3 and 6, respectively. The log-density function of the grouped mixture with $\boldsymbol{\nu} = (3,6)$ is not bounded by either; in fact, the grouped mixture shows heavier tails than both the t distribution with 3 and with 6 dof.

5. Kendall tau and Spearman rho

Two widely used measures of association are the rank correlation coefficients Spearman's rho ρ_S and Kendall's tau ρ_τ. For elliptical models, one can easily compute Spearman's rho as a function of the copula parameter ρ which can be useful in estimating the matrix P non-parametrically.

For grouped mixtures, however, this is not easily possible. In this section, integral representations for Spearman's rho and Kendall's tau in the general grouped NVM case are derived.

If $\boldsymbol{X} = (X_1, X_2) \sim F$ is a random vector with continuous margins F_1, F_2, then $\rho_S(X_1, X_2) = \rho(F_1(X_1), F_2(X_2))$ and $\rho_\tau(X_1, X_2) = \mathbb{P}((X_1 - Y_1)(X_2 - Y_2) > 0) - \mathbb{P}((X_1 - Y_1)(X_2 - Y_2) < 0)$, where $(Y_1, Y_2) \sim F$ independent of (X_1, X_2) and $\rho(X, Y) = \mathrm{cov}(X, Y)/\sqrt{\mathrm{var}(X)\,\mathrm{var}(Y)}$ is the linear correlation between X and Y. Both ρ_S and ρ_τ depend only on the copula of F.

If $\boldsymbol{X} \sim \mathrm{ELL}_2(\boldsymbol{\mu}, \Sigma, F_R)$ is elliptical and $\rho = \Sigma_{12}/\sqrt{\Sigma_{11}\Sigma_{22}}$, then

$$\rho_\tau(X_1, X_2) = \frac{2}{\pi}\arcsin(\rho); \tag{20}$$

see (Lindskog et al. 2003, Theorem 2). This formula holds only approximately for grouped normal variance mixtures. In Daul et al. (2003), an expression was derived for Kendall's tau of bivariate, grouped t copulas. Their result is easily extended to the more general grouped normal variance mixture case; see Section 8 for the proof.

Proposition 1. *Let* $\boldsymbol{X} \sim gNVM_2(\boldsymbol{\mu}, \Sigma, F_W)$ *and* $\rho = \Sigma_{12}/\sqrt{\Sigma_{11}\Sigma_{22}}$. *Then*

$$\rho_\tau(X_1, X_2) = \frac{2}{\pi}\mathbb{E}\left(\arcsin\left(\rho\frac{F^{\leftarrow}_{W_1}(U)F^{\leftarrow}_{W_2}(U) + F^{\leftarrow}_{W_1}(\tilde{U})F^{\leftarrow}_{W_2}(\tilde{U})}{\sqrt{(F^{\leftarrow}_{W_1}(U)^2 + F^{\leftarrow}_{W_1}(\tilde{U})^2)(F^{\leftarrow}_{W_2}(U)^2 + F^{\leftarrow}_{W_2}(\tilde{U})^2)}}\right)\right), \tag{21}$$

where $U, \tilde{U} \overset{ind.}{\sim} \mathrm{U}(0, 1)$.

Next, we address Spearman's rho ρ_S. For computing ρ_S, it is useful to study $\mathbb{P}(X_1 > 0, X_2 > 0)$. If $\boldsymbol{X} \sim \mathrm{ELL}_2(\boldsymbol{\mu}, P, F_R)$ where P is a correlation matrix with $P_{12} = \rho$ and $\mathbb{P}(\boldsymbol{X} = \boldsymbol{0}) = 0$, then

$$P(X_1 > 0, X_2 > 0) = \frac{1}{4} + \frac{\arcsin(\rho)}{2\pi},$$

see, e.g., (McNeil et al. 2015, Proposition 7.41). Using the same technique, we can show that this result also holds for grouped normal variance mixtures; see Section 8 for the proof.

Proposition 2. *Let* $\boldsymbol{X} \sim gNVM_2(\boldsymbol{\mu}, \Sigma, F_W)$ *and* $\rho = \Sigma_{12}/\sqrt{\Sigma_{11}\Sigma_{22}}$. *Then*

$$P(X_1 > 0, X_2 > 0) = \frac{1}{4} + \frac{\arcsin(\rho)}{2\pi}.$$

Remark 1. *If* $\boldsymbol{Y}$ *is a grouped elliptical distribution in the sense of* (5), *a very similar idea can be used to show that* $\mathbb{P}(Y_1 > 0, Y_2 > 0) = 1/4 + \arcsin(\rho)/(2\pi)$.

Next, we derive a new expression for Spearman's rho ρ_S for bivariate grouped normal variance mixture distributions; see Section 8 for the proof.

Proposition 3. *Let* $\boldsymbol{X} \sim gNVM_2(\boldsymbol{0}, P, F_W)$ *and* $\rho = P_{12}$. *Then*

$$\rho_S(X_1, X_2) = \frac{6}{\pi}\mathbb{E}\left(\arcsin\left(\rho\sqrt{\frac{F^{\leftarrow}_{W_1}(U)F^{\leftarrow}_{W_2}(U)}{(F^{\leftarrow}_{W_1}(U) + F^{\leftarrow}_{W_1}(\tilde{U}))(F^{\leftarrow}_{W_2}(U) + F^{\leftarrow}_{W_2}(\bar{U}))}}\right)\right), \tag{22}$$

where $U, \tilde{U}, \bar{U} \overset{ind.}{\sim} \mathrm{U}(0, 1)$.

Numerical Results

Let $\boldsymbol{X} \sim \mathrm{gNVM}_2(\boldsymbol{0}, P, F_W)$. It follows from Proposition 1 that

$$\rho_\tau(X_1, X_2) = \int_{(0,1)^2} g_\tau(\boldsymbol{u})\,\mathrm{d}\boldsymbol{u}, \quad g_\tau(\boldsymbol{u}) = \frac{2}{\pi}\arcsin\left(\rho\frac{F^{\leftarrow}_{W_1}(u_1)F^{\leftarrow}_{W_2}(u_1) + F^{\leftarrow}_{W_1}(u_2)F^{\leftarrow}_{W_2}(u_2)}{\sqrt{(F^{\leftarrow}_{W_1}(u_1)^2 + F^{\leftarrow}_{W_1}(u_2)^2)(F^{\leftarrow}_{W_2}(u_1)^2 + F^{\leftarrow}_{W_2}(u_2)^2)}}\right).$$

Similarly, Proposition 3 implies that

$$\rho_S(X_1, X_2) = \int_{(0,1)^3} g_\rho(\boldsymbol{u})\,\mathrm{d}\boldsymbol{u}, \quad g_\rho(\boldsymbol{u}) = \frac{6}{\pi}\arcsin\left(\rho\sqrt{\frac{F^{\leftarrow}_{W_1}(u_1)F^{\leftarrow}_{W_2}(u_1)}{(F^{\leftarrow}_{W_1}(u_1) + F^{\leftarrow}_{W_1}(u_2))(F^{\leftarrow}_{W_2}(u_1) + F^{\leftarrow}_{W_2}(u_3))}}\right).$$

Hence, both $\rho_\tau(X_1, X_2)$ and $\rho_S(X_1, X_2)$ can be expressed as integrals over the d-dimensional unit hypercube with $d \in \{2, 3\}$ so that RQMC methods as described in Section 2.2 can be applied directly to the problem in this form to estimate $\rho_\tau(X_1, X_2)$ and $\rho_S(X_1, X_2)$, respectively. This is implemented in the function `corgnvmix()` (with `method = "kendall"` or `method = "spearman"`) of the R package `nvmix`.

As an example, we consider three different bivariate grouped t distributions with $\boldsymbol{\nu} \in \{(1,2), (4,8), (1,5), (4,20), (1,\infty), (4,\infty)\}$ and plot estimated ρ_τ as a function of ρ in Figure 4. The elliptical case (corresponding to equal dof) is included for comparison. When the pairwise dof are close and ρ is not too close to 1, the elliptical approximation is quite satisfactory. However, when the dof are further apart there is a significant difference between the estimated ρ_τ and the elliptical approximation. This is highlighted in the plot on the right side, which displays the relative difference $(\rho_\tau^{\text{ell}} - \rho_\tau)/\rho_\tau^{\text{ell}}$. Intuitively, it makes sense that the approximation deteriorates when the dof are further apart, as the closer the dof, the "closer" is the model to being elliptical.

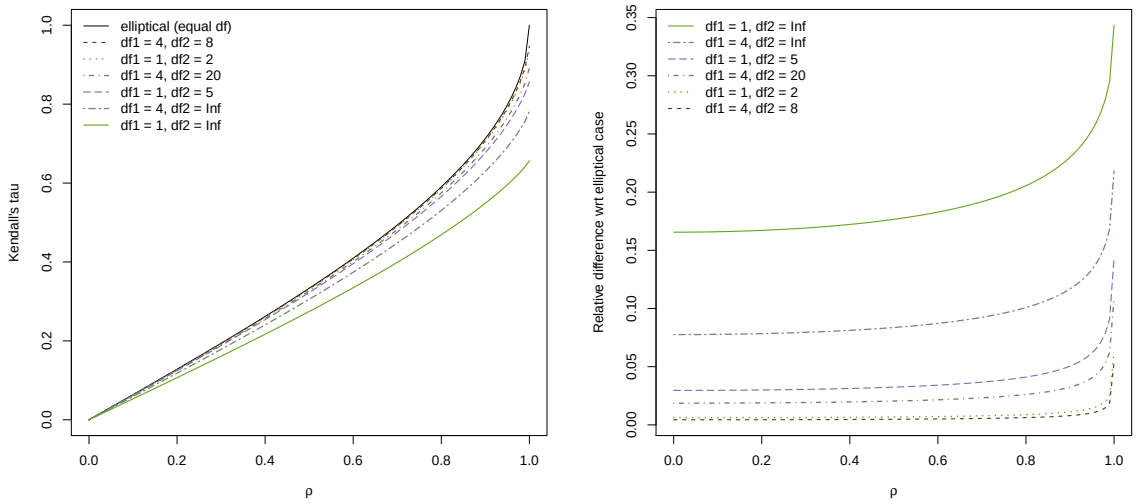

Figure 4. Kendall's tau for a bivariate grouped t distribution for various ρ estimated via `corgnvmix()` (**left**); relative difference of ρ_τ wrt the elliptical case (**right**).

6. Copula Setting

So far, the focus of this paper was on grouped normal variance mixtures. This section addresses grouped normal variance mixture copulas, i.e., the copulas derived from $\boldsymbol{X} \sim \text{gNVM}(F_W, \boldsymbol{\mu}, \Sigma)$ via Sklar's theorem. The first part addresses grouped NVM copulas in full generality and provides formulas for the copula, its density and the tail dependence coefficients. The second part details the important special case of inverse-gamma mixture copulas, that is copulas derived from a grouped t distribution, $\boldsymbol{X} \sim \text{gt}_d(\boldsymbol{\nu}, \boldsymbol{\mu}, \Sigma)$. The third part discusses estimation of the copula and its density whereas the fourth part answers the question of how copula parameters can be fitted to a dataset. The last part of this section includes numerical examples.

6.1. Grouped Normal Variance Mixture Copulas

Copulas provide a flexible tool for modeling dependent risks, as they allow one to model the margins separately from the dependence between the margins. Let $\boldsymbol{X} \sim F$ be a d-dimensional random vector with continuous margins $F_1, \dots, F_d$. Consider the random vector $\boldsymbol{U}$ given by $\boldsymbol{U} = (U_1, \dots, U_d) = (F_1(X_1), \dots, F_d(X_d))$; note that $U_j \sim \mathrm{U}(0,1)$ for $j = 1, \dots, d$. The *copula* C of F (or $\boldsymbol{X}$) is the distribution function of the margin-free $\boldsymbol{U}$, i.e.,

$$C(\boldsymbol{u}) = \mathbb{P}\left(F_1(X_1) \le u_1, \dots, F_d(X_d) \le u_d\right) = F\left(F_1^{\leftarrow}(u_1), \dots, F_d^{\leftarrow}(u_d)\right), \quad \boldsymbol{u} = (u_1, \dots, u_d) \in [0,1]^d.$$

If F is absolutely continuous and the margins $F_1, \dots, F_d$ are strictly increasing and continuous, the *copula density* is given by

$$c(\boldsymbol{u}) = \frac{\partial}{\partial u_1 \dots \partial u_d} C(u_1, \dots, u_d) = \frac{f\left(F_1^{\leftarrow}(u_1), \dots, F_d^{\leftarrow}(u_d)\right)}{\prod_{j=1}^{d} f_j\left(F_j^{\leftarrow}(u_j)\right)}, \quad \boldsymbol{u} = (u_1, \dots, u_d) \in (0,1)^d, \tag{23}$$

where f denotes the (joint) density of F and f_j is the marginal density of F_j. For more about copulas and their applications to risk management, see, e.g., Embrechts et al. (2001); Nelsen (2007).

Since copulas are invariant with respect to strictly increasing marginal transformations, we may wlog assume that $\boldsymbol{\mu} = 0$, $\Sigma = P$ is a correlation matrix and we may consider $\boldsymbol{X} \sim \mathrm{gNVM}_d(\boldsymbol{0}, P, F_{\boldsymbol{W}})$. We find using (11) that the *grouped normal variance mixture copula* is given by

$$C^{\mathrm{gNVM}}_{P,F_{\boldsymbol{W}}}(\boldsymbol{u}) = F\left(F_1^{\leftarrow}(u_1), \dots, F_d^{\leftarrow}(u_d)\right) = \int_0^1 \Phi_\Sigma\left((1/\sqrt{\boldsymbol{w}})^{\mathrm{D}}(u)\boldsymbol{x}\right) \mathrm{d}u, \quad \boldsymbol{x} = (F_1^{\leftarrow}(u_1), \dots, F_d^{\leftarrow}(u_d)), \tag{24}$$

and its density can be computed using (18) as

$$\begin{aligned} c^{\mathrm{gNVM}}_{P,F_{\boldsymbol{W}}}(\boldsymbol{u}) &= \frac{f\left(F_1^{\leftarrow}(u_1), \dots, F_d^{\leftarrow}(u_d)\right)}{\prod_{i=1}^{d} f_i\left(F_i^{\leftarrow}(u_i)\right)} \\ &= \frac{\int_0^1 \sqrt{(2\pi)^d |\Sigma| \prod_{j=1}^d F^{\leftarrow}_{W_j}(u)}^{-1} \exp\left(-\frac{D^2(\boldsymbol{x};\boldsymbol{\mu},\sqrt{\boldsymbol{W}}^{\mathrm{D}}(u)\Sigma\sqrt{\boldsymbol{W}}^{\mathrm{D}}(u))}{2}\right) \mathrm{d}u}{\prod_{j=1}^{d} f_j\left(F_j^{\leftarrow}(u_j)\right)}, \end{aligned} \tag{25}$$

where F_j and f_j denote the distribution function and density function of $X_j \sim \mathrm{NVM}_1(0, 1, F_{W_i})$ for $j = 1, \dots, d$; directly considering $\log(c^{\mathrm{gNVM}}_{P,F_{\boldsymbol{W}}}(\boldsymbol{u}))$ also makes (25) more robust to compute.

In the remainder of this subsection, some useful properties of gNVM copulas are derived. In particular, we study symmetry properties, rank correlation and tail dependence coefficients.

6.1.1. Radial Symmetry and Exchangeability

A d-dimensional random vector $\boldsymbol{X}$ is radially symmetric about $\boldsymbol{\mu} \in \mathbb{R}^d$ if $\boldsymbol{X} - \boldsymbol{\mu} \overset{\mathrm{d}}{=} \boldsymbol{\mu} - \boldsymbol{X}$. It is evident from (2) that $\boldsymbol{X} \sim \mathrm{gNVM}_d(\boldsymbol{\mu}, \Sigma, F_{\boldsymbol{W}})$ is radially symmetric about its location vector $\boldsymbol{\mu}$. In layman's terms this implies that jointly large values of $\boldsymbol{X}$ are as likely as jointly small values of $\boldsymbol{X}$. Radial symmetry also implies that $c^{\mathrm{gNVM}}_{P,F_{\boldsymbol{W}}}(\boldsymbol{u}) = c^{\mathrm{gNVM}}_{F_{\boldsymbol{W}}}(\boldsymbol{1} - \boldsymbol{u})$.

If $(X_{\Pi(1)}, \dots, X_{\Pi(d)}) \overset{\mathrm{d}}{=} (X_1, \dots, X_d)$ for all permutations Π of $\{1, \dots, d\}$, the random vector $\boldsymbol{X}$ is called exchangeable. The same definition applies to copulas. If $\boldsymbol{X} \sim \mathrm{gNVM}_d(\boldsymbol{0}, I_d, F_{\boldsymbol{W}})$, then $\boldsymbol{X}$ is in general not exchangeable unless $F_{W_1} = \cdots = F_{W_d}$ in which case $\boldsymbol{X} \sim \mathrm{NVM}_d(\boldsymbol{0}, P, F_{W_1})$. The lack of exchangeability implies that $c^{\mathrm{gNVM}}_{I_d,F_{\boldsymbol{W}}}(u_1, \dots, u_d) \neq c^{\mathrm{gNVM}}_{I_d,F_{\boldsymbol{W}}}(u_{\Pi(1)}, \dots, u_{\Pi(d)})$, in general.

6.1.2. Tail Dependence Coefficients

Consider a bivariate $C^{\mathrm{gNVM}}_{P,F_W}$ copula. Such copula is radially symmetric, hence the lower and upper tail dependence coefficients are equal, i.e., $\lambda_l = \lambda_u =: \lambda \in [0,1]$, where

$$\lambda_l = \lim_{q\to 0^+} \mathbb{P}(U_2 \le q \mid U_1 \le q) = \lim_{q\to 0^+} \frac{C(q,q)}{q},$$

for $(U_1, U_2) \sim C^{\mathrm{gNVM}}_{P,F_W}$. In the case where only the quantile functions $F^{\leftarrow}_{W_j}$ are available, no simple expression for λ is available. In Luo and Shevchenko (2010), λ is derived for grouped t copulas, as will be discussed in Section 6.2. Following the arguments used in their proof, the following lemma provides a new expression for λ in the more general normal variance mixture case.

Proposition 4. *The tail dependence coefficient λ for a bivariate C^{gNVM}_{P,F_W} with $\rho = P_{12}$ satisfies*

$$\lambda = \lim_{q\to 0^+} I(q,1,2) + I(q,2,1),$$

where for $i, j \in \{1,2\}$,

$$I(q,i,j) = \int_0^1 \frac{\phi\left(F_i^{\leftarrow}(q)/\sqrt{F^{\leftarrow}_{W_i}(u)}\right)}{\sqrt{F^{\leftarrow}_{W_i}(u)}\, f_i(F_i^{\leftarrow}(q))} \Phi\left(\frac{F_j^{\leftarrow}(q)/\sqrt{F^{\leftarrow}_{W_j}(u)} - \rho F_i^{\leftarrow}(q)/\sqrt{F^{\leftarrow}_{W_i}(u)}}{\sqrt{1-\rho^2}}\right) \mathrm{d}u.$$

6.2. Inverse-Gamma Mixtures

If $\boldsymbol{X} \sim t_d(\nu, \mathbf{0}, P)$ for a positive definite correlation matrix P, the copula of $\boldsymbol{X}$ extracted via Sklar's theorem is the well known t copula, denoted by $C^t_{\nu,P}$. This copula is given by

$$C^{\mathrm{t}}_{\nu,P}(\boldsymbol{u}) = \int_{-\infty}^{t_\nu^{-1}(u_1)} \cdots \int_{-\infty}^{t_\nu^{-1}(u_d)} \frac{\Gamma((\nu+d)/2)}{\Gamma(\nu/2)\sqrt{(\nu\pi)^d |P|}} \left(1 + \frac{\boldsymbol{x}^\top P^{-1} \boldsymbol{x}}{\nu}\right)^{-\frac{\nu+d}{2}} \mathrm{d}\boldsymbol{x}, \quad \boldsymbol{u} = (u_1, \ldots, u_d) \in [0,1]^d, \tag{26}$$

where t_ν and t_ν^{-1} denote the distribution function and quantile function of a univariate standard t distribution. Note that (26) is merely the distribution function of $\boldsymbol{X} \sim t_d(\nu, \mathbf{0}, P)$ evaluated at the quantiles $t_\nu^{-1}(u_1), \ldots, t_\nu^{-1}(u_d)$. The copula density $c^{\mathrm{t}}_{\nu,P}(\boldsymbol{u})$ is

$$c^{\mathrm{t}}_{\nu,P}(\boldsymbol{u}) = f^{\mathrm{t}}_{\nu,0,P}(t_\nu^{-1}(u_1), \ldots, t_\nu^{-1}(u_d)) \left(\prod_{j=1}^d f_{t_\nu}(t_\nu^{-1}(u_j))\right)^{-1}, \quad \boldsymbol{u} \in [0,1]^d.$$

The (upper and lower) tail dependence coefficient λ of the bivariate $C^{\mathrm{t}}_{\nu,P}$ with $\rho = P_{12}$ is well known to be

$$\lambda = 2t_{\nu+1}\left(-\sqrt{(\nu+1)(1-\rho)/(1+\rho)}\right);$$

see (Demarta and McNeil 2005, Propositon 1). The multivariate t distribution being elliptical implies the formula $\rho_\tau = 2\arcsin(\rho)/\pi$ for Kendall's tau.

A closed formula for Spearman's rho is not available, but our Proposition 3 implies that

$$\rho_{\mathrm{S}} = \frac{6}{\pi}\mathbb{E}\left(\arcsin\left(\rho W \sqrt{(W+\tilde{W})(W+\bar{W})}^{-1}\right)\right), \quad W, \tilde{W}, \bar{W} \overset{\text{ind.}}{\sim} \mathrm{IG}(\nu/2, \nu/2).$$

The same formula was given in (McNeil et al. 2015, Proposition 7.44).

Next, consider a grouped inverse-gamma mixture model. If $\boldsymbol{X} \sim \mathrm{gt}_d(\boldsymbol{\nu}, \mathbf{0}, P)$, the copula of $\boldsymbol{X}$ is the grouped t copula, denoted by $C^{\mathrm{gt}}_{\boldsymbol{\nu},P}$. From (24),

$$C^{\mathrm{gt}}_{\boldsymbol{\nu},P}(\boldsymbol{u}) = \int_0^1 \Phi_P\left((1/\sqrt{W})^{\mathrm{D}}(u)\boldsymbol{x}\right)\,\mathrm{d}u, \quad \boldsymbol{x} = (t^{-1}_{\nu_1}(u_1), \dots, t^{-1}_{\nu_d}(u_d)),$$

and the copula density follows from (25) as

$$c^{\mathrm{gt}}_{\boldsymbol{\nu},P}(\boldsymbol{u}) = f^{\mathrm{gt}}_{\boldsymbol{\nu},\mathbf{0},P}\left(t^{-1}_{\nu_1}(u_1), \dots, t^{-1}_{\nu_d}(u_d)\right)\left(\prod_{j=1}^d f_{t_{\nu_j}}\left(t^{-1}_{\nu_j}(u_j)\right)\right)^{-1}.$$

The (lower and upper) tail dependence coefficient λ of $C^{\mathrm{gt}}_{\nu_1,\nu_2,P}$ is given by

$$\begin{aligned} \lambda &= \Omega(\rho,\nu_1,\nu_2) + \Omega(\rho,\nu_2,\nu_1), \\ \Omega(\rho,\nu_1,\nu_2) &= \int_0^\infty f_{\chi^2_{\nu_1+1}}(t)\Phi\left(-(B_{\nu_1,\nu_2}t^{\nu_1/(2\nu_2)} - \rho\sqrt{t})(1-\rho^2)^{-1/2}\right)\,\mathrm{d}t, \\ B_{\nu_1,\nu_2} &= \left(\frac{2^{\nu_2/2}\Gamma((1+\nu_2)/2)}{2^{\nu_1/2}\Gamma((1+\nu_1)/2)}\right)^{1/\nu_2}; \end{aligned} \tag{27}$$

see (Luo and Shevchenko 2010, Equation (26)). Here, $f_{\chi^2_\nu}$ denotes the density of a χ^2_ν distribution.

Finally, consider rank correlation coefficients for grouped t copulas. No closed formula for either Kendall's tau or Spearman's rho exists in the grouped t case. An exact integral representation of ρ_τ for $C^{\mathrm{gt}}_{\nu_1,\nu_2,P}$ follows from Proposition 1. No substantial simplification of (21) therein can be achieved by considering the special case when $W_j \sim \mathrm{IG}(\nu_j/2, \nu_j/2)$. In order to compute ρ_τ, one can either numerically integrate (21) (as will be discussed in the next subsection) or use the approximation $\rho_\tau \approx \frac{2}{\pi}\arcsin(\rho)$ which was shown to be a "very accurate" approximation in Daul et al. (2003).

For Spearman's rho, no closed formula can be derived either, not even in the ungrouped t copula case, so that the integral in (22) in Proposition 3 needs be computed numerically, as will be discussed in the next subsection.

The discussion in this section highlights that moving from a scalar mixing rv W (as in the classical t case) to comonotone mixing rvs $W_1, \dots, W_S$ (as in the grouped t case) introduces challenges from a computational point of view. While in the classical t setting, the density, Kendall's tau and the tail dependence coefficient are available in closed form, all of these quantities need to be estimated in the more general grouped setting. Efficient estimation of these important quantities is discussed in the next subsection.

6.3. Estimation of the Copula and Its Density

Consider a d-dimensional normal variance mixture copula $C^{\mathrm{gNVM}}_{P,F_W}$. From (24), it follows that

$$C^{\mathrm{gNVM}}_{P,F_W}(\boldsymbol{u}) = F_{\boldsymbol{X}}(F_1^{\leftarrow}(u_1), \dots, F_d^{\leftarrow}(u_d)), \quad \boldsymbol{u} = (u_1, \dots, u_d) \in [0,1]^d,$$

where $F_{\boldsymbol{X}}$ is the distribution function of $\boldsymbol{X} \sim \mathrm{gNVM}(\mathbf{0}, P, F_{\boldsymbol{W}})$ and F_j is the distribution function of $\mathrm{NVM}_1(0, 1, F_{W_j})$ for $j = 1, \dots, d$. If the margins are known (as in the case of an inverse-gamma mixture), evaluating the copula is no harder than evaluating the distribution function of $\boldsymbol{X}$ so that the methods described in Section 3.1 can be applied.

When the mixing rvs W_j are only known through their quantile functions in the form of a "black box", one needs to estimate the marginal quantiles F_j of F first. Note that

$$F_j(x) = \mathbb{P}(X_j \le x) = \int_{(0,1)} \Phi\left(x/\sqrt{F^{\leftarrow}_{W_j}(u)}\right)\,\mathrm{d}u, \quad x \in \mathbb{R}, \tag{28}$$

which can be estimated using RQMC. The quantile $F_j^{\leftarrow}(u_j)$ can then be estimated by numerically solving $F_j(x) = u$ for x, for instance using a bisection algorithm or Newton's method.

The general form of gNVM copula densities was given in (25). Again, if the margins are known, the only unknown quantity is the joint density $f_{\boldsymbol{X}}$ which can be estimated using the adaptive RQMC procedure proposed in Section 4.1. If the margins are not available, $F_j^{\leftarrow}$ can be estimated as discussed above. The marginal densities f_j can be estimated using an adaptive RQMC algorithm similar to the one developed in Section 4.1; see also (Hintz et al. 2020, Section 4).

Remark 2. *Estimating the copula density is the most challenging problem discussed in this paper if we assume that $F_{\boldsymbol{W}}$ is only known via its marginal quantile functions. Evaluating the copula density $c_{P,F_{\boldsymbol{W}}}^{gNVM}$ at one $\boldsymbol{u} \in [0,1]^d$ requires estimation of:*

- *the marginal quantiles $F_j^{\leftarrow}(u_j)$, which involves estimation of F_j and then numerical root finding, for each $j = 1, \dots, d$,*
- *the marginal densities evaluated at the quantiles $f_j(F_j^{\leftarrow}(u_j))$ for $j = 1, \dots, d$. This involves estimation of the density of a univariate normal variance mixture,*
- *the joint density evaluated at the quantiles $f(F_1^{\leftarrow}(u_1), \dots, F_d^{\leftarrow}(u_d))$, which is another one dimensional integration problem.*

It follows from Remark 2 that, while estimation of $c_{P,F_{\boldsymbol{W}}}^{\mathrm{gNVM}}$ is theoretically possible with the methods proposed in this paper, the problem becomes computationally intractable for large dimensions d. If the margins are known, however, our proposed methods are efficient and accurate, as demonstrated in next subsection, where we focus on the important case of a grouped t model. Our methods to estimate the copula and the density of $C_{\boldsymbol{\nu},\boldsymbol{k},P}^{\mathrm{gt}}$ are implemented in the functions `pgStudentcopula()` and `dgStudentcopula()` in the R package `nvmix`.

6.4. Fitting Copula Parameters to a Dataset

In this subsection, we discuss estimation methods for grouped normal variance mixture copulas. Let $\boldsymbol{X}_1, \dots, \boldsymbol{X}_n$ be independent and distributed according to some distribution with $C_{P,F_{\boldsymbol{W}}}^{\mathrm{gNVM}}$ as underlying copula, with $\boldsymbol{X}_i = (X_{i,1}, \dots, X_{i,d})$ and group sizes $d_1, \dots, d_S$ with $\sum_{j=1}^S d_j = d$. Furthermore, let $\boldsymbol{\nu}_k$ be (a vector of) parameters of the kth mixing distribution for $k = 1, \dots, S$; for instance, in the grouped t case, $\boldsymbol{\nu}_k = \nu_k$ is the degrees of freedom for group k. Finally, denote by $\boldsymbol{\nu} = (\boldsymbol{\nu}_1, \dots, \boldsymbol{\nu}_S)$ the vector consisting of all mixing parameters. Note that we assume that the group structure is given. We are interested in estimating the parameter vector $\boldsymbol{\nu}$ and the matrix P of the underlying copula $C_{P,F_{\boldsymbol{W}}}^{\mathrm{gNVM}}$.

In Daul et al. (2003), this problem was discussed for the grouped t copula where $d_k \geq 2$ for $k = 1, \dots, S$. In this case, all subgroups are t copulas and Daul et al. (2003) suggest estimating the dof $\nu_1, \dots, \nu_S$ separately in each subgroup. Computationally, this is rather simple as the density of the ungrouped t copula is known analytically. Luo and Shevchenko (2010) consider the grouped t copula with $S = d$, so $d_k = 1$ for $k = 1, \dots, d$. Since any univariate margin of a copula is uniformly distributed, separate estimation is not feasible. As such, Luo and Shevchenko (2010) suggest estimating $\nu_1, \dots, \nu_S$ jointly by maximizing the copula-likelihood of the grouped mixture. In both references, the matrix P is estimated by estimating pairwise Kendall's tau and using the approximate identity $\rho_\tau(X_i, X_j) \approx 2 \arcsin(\rho_{i,j})/\pi$ for $i \neq j$. Although we have shown in Section 5 that in some cases, this approximation could be too crude, our assessment is that in the context of the fitting examples considered in the present section, this approximation is sufficiently accurate. Luo and Shevchenko (2010) also consider joint estimation of $(P, \boldsymbol{\nu})$ by maximizing the corresponding copula likelihood simultaneously over all $d + d(d-1)/2$ parameters. Their numerical results in $d = 2$ suggest that this does not lead to a significant improvement. In large dimensions $d > 2$,

the optimization problem becomes intractable, however, so that the first non-parametric approach for estimating P is likely to be preferred.

We combine the two estimation methods, applied to the general case of a grouped normal variance mixture, in Algorithm 2.

Algorithm 2: Estimation of the Copula Parameters $\boldsymbol{\nu}$ and $\boldsymbol{P}$ of $\boldsymbol{C}^{\text{gNVM}}_{\boldsymbol{P},\boldsymbol{F_W}}$.

Given iid $\boldsymbol{X}_1, \dots, \boldsymbol{X}_n$, estimate $\boldsymbol{\nu}$ and P of the underlying C^{gNVM}_{P,F_W} as follows:

1. *Estimation of P.* Estimate Kendall's tau $\rho_\tau(X_i, X_j)$ for each pair $1 \le i < j \le d$. Use the approximate identity $\rho_\tau(X_i, X_j) \approx 2\arcsin(\rho_{i,j})/\pi$ to find the estimates $\rho_{i,j}$. Then combine the estimates $\rho_{i,j}$ into a correlation matrix $\hat{P}$, which may have to be modified to ensure positive definiteness.
2. *Transformation to pseudo-observations.* If necessary, transform the data $\boldsymbol{X}_1, \dots, \boldsymbol{X}_n$ to pseudo-observations $\boldsymbol{U}_1, \dots, \boldsymbol{U}_n$ from the underlying copula, for instance, by setting $U_{i,j} = R_{i,j}/(n+1)$ where $R_{i,j}$ is the rank of $X_{i,j}$ among $X_{1,j}, \dots, X_{n,j}$.
3. *Initial parameters.* Maximize the copula log-likelihood for each subgroup k with $d_k \ge 2$ over their respective parameters separately. That is, if $\boldsymbol{U}_i^{(k)} = (U_{i,d_{k-1}+1}, \dots, U_{i,d_{k-1}+d_k})$ (where $d_0 = 0$) denotes the sub-vector of $\boldsymbol{U}_i$ belonging to group k, and if $\hat{P}^{(k)}$ is defined accordingly, solve the following optimization problems:

$$\begin{aligned} \hat{\boldsymbol{\nu}}_0^{(k)} &= \operatorname{argmax} l(\boldsymbol{\nu}^{(k)}; \boldsymbol{U}_1^{(k)}, \dots, \boldsymbol{U}_n^{(k)}), \\ l(\boldsymbol{\nu}^{(k)}; \boldsymbol{U}_1^{(k)}, \dots, \boldsymbol{U}_n^{(k)}) &= \sum_{i=1}^n \log c^{\text{gNVM}}_{\hat{P}^{(k)},F_W}(\boldsymbol{U}_i^{(k)}), \quad \forall k : d_k \ge 2. \end{aligned} \tag{29}$$

   For "groups" with $d_k = 1$, choose the initial estimate $\hat{\boldsymbol{\nu}}_0^{(k)}$ from prior/expert experience or as a hard-coded value.
4. *Joint estimation.* With initial estimates $\hat{\boldsymbol{\nu}}_0^{(k)}$, $k = 1, \dots, S$ at hand, optimize the full copula likelihood to estimate $\boldsymbol{\nu}$; that is,

$$\hat{\boldsymbol{\nu}} = \operatorname{argmax} l(\boldsymbol{\nu}; \boldsymbol{U}_1, \dots, \boldsymbol{U}_n), \quad l(\boldsymbol{\nu}; \boldsymbol{U}_1, \dots, \boldsymbol{U}_n) = \sum_{i=1}^n \log c^{\text{gNVM}}_{\hat{P},F_W}(\boldsymbol{U}_i). \tag{30}$$

The method proposed in Daul et al. (2003) returns the initial estimates obtained in Step 3. A potential drawback of this approach is that it fails to consider the dependence between the groups correctly. Indeed, the dependence between a component in group k_1 and a component in group k_2 (e.g., measured by Kendall's tau or by the tail-dependence coefficient) is determined by both $\boldsymbol{\nu}^{(k_1)}$ and $\boldsymbol{\nu}^{(k_2)}$. As such, these parameters should be estimated jointly.

Note that the copula density is not available in closed form, not even in the grouped t case, so that each call of the likelihood function in (30) requires the approximation of n integrals. This poses numerical challenges, as the estimated likelihood function is typically "bumpy", having many local maxima due to estimation errors.

If F_W is only known via its marginal quantile functions, as is the general theme of this paper, the optimization problem in (29) and in (30) become intractable (unless d and n are small) due to the numerical challenges involved in the estimation of the copula density; see also Remark 2. We leave the problem of fitting grouped normal variance mixture copulas in full generality (where the distribution of the mixing random variables W_j is only specified via marginal quantile functions in the form of a "black box") for future research. Instead, we focus on the important case of a grouped t copula. Here, the quantile functions $F_j^{\leftarrow}$ (of X_j) and the densities f_j are known for $j = 1, \dots, d$, since the margins are all t distributed. This substantially simplifies the underlying numerical procedure. Our method is implemented in the function `fitgStudentcopula()` of the R package `nvmix`. The numerical

optimizations in Steps 3 and 4 are passed to the R optimizer `optim()` and the copula density is estimated as in Section 6.3.

Example 1. *Consider a 6-dimensional grouped t copula, with three groups of size 2 each and degrees of freedom* $1, 4$ *and* 7*, respectively. We perform the following experiment: We sample a correlation matrix P using the R function* `rWishart()`*. Then, for each sample size* $n \in \{250, 500, \dots, 1750, 2000\}$*, we repeat sampling* $\boldsymbol{X}_1, \dots, \boldsymbol{X}_n$ *15 times, and in each case, estimate the degrees of freedom once using the method in Daul et al. (2003) (i.e., by estimating the dof in each group separately) and once using our method from the previous section. The true matrix P is used in the fitting, so that the focus is really on estimating the dof. The results are displayed in Figure 5. The estimates on the left are obtained for each group separately; on the right, the dof were estimated jointly by maximizing the full copula likelihood (with initial estimates obtained as in the left figure). Clearly, the jointly estimated parameters are much closer to their true values (which are known in this simulation study and indicated by horizontal lines), and it can be confirmed that the variance decreases with increasing sample size n.*

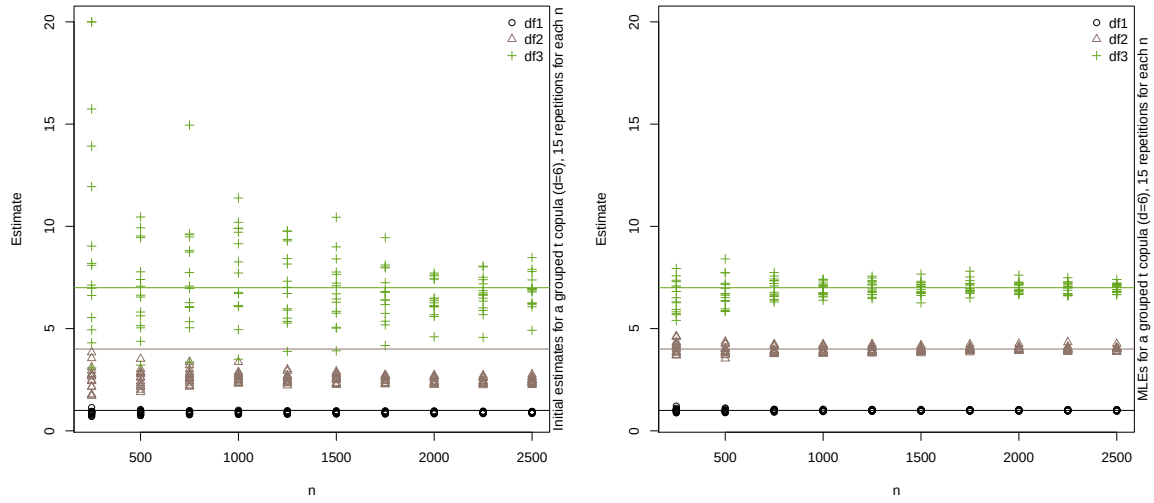

Figure 5. Estimated dof for a 6-dimensional grouped t copula with 3 groups of size 2 each. (**left**) Estimates are obtained by fitting the t copulas of each group separately; (**right**) the joint copula-likelihood of the grouped t copula is maximized.

Example 2. *Let us now consider the negative logarithmic returns of the constituents of the Dow Jones 30 index from 1 January 2014 to 31 December 2015 (*$n = 503$ *data points obtained from the R package* `qrmdata` *of Hofert and Hornik (2016)) and, after deGARCHing, fit a grouped t copula to the standardized residuals. We choose the natural groupings induced by the industry sectors of the 30 constituents and merge groups of size 1 so that 9 groups are left. Figure 6 displays the estimates obtained for various specifications of* `maxit`*, the maximum number of iterations for the underlying optimizer (note that the current default of* `optim()` *is as low as* `maxit = 500`*). The points for* `maxit = 0` *correspond to the initial estimates found from separately fitting t copulas to the groups. The initial estimates differ significantly from the maximum likelihood estimates (MLEs) obtained from the joint estimation of the dof. Note also that the MLEs change with increasing* `maxit` *argument, even though they do not change drastically anymore if 1500 or more iterations are used. Note that the initial parameters result in a much more heavy tailed model than the MLEs. Figure 6 also displays the estimated log-likelihood of the parameters found by the fitting procedure. The six lines correspond to the estimated log-likelihood using six different seeds. It can be seen that estimating the dof jointly (as opposed to group-wise) yields a substantially larger log-likelihood, whereas increasing the parameter* `maxit` *(beyond a necessary minimum) only gives a minor improvement.*

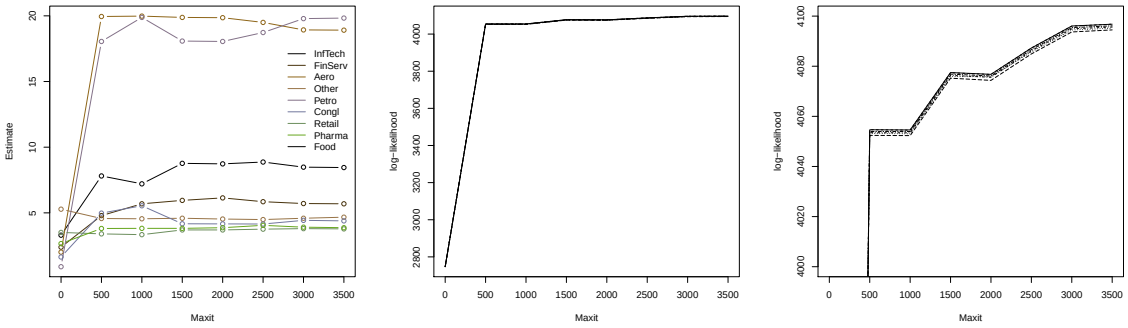

Figure 6. Estimated dof for a grouped t copula fitted to the standardized residuals obtained from the Dow Jones 30 dataset from 1 January 2014 to 31 December 2015 after deGARCHing (**left**); estimated log-likelihood of the estimated dof on the **middle** and **right** plot.

In order to examine the impact of the different estimates on the underlying copula in terms of its tail behavior, Figure 7 displays the probability $C(u, \dots, u)$ estimated using methods from Section 6.3 as a function of u; in a risk management context, $C(u, \dots, u)$ is the probability of a jointly large loss, hence a rare event. An absolute error tolerance of 10^{-7} was used to estimate the copula. The figure also includes the corresponding probability for the ungrouped t copula, for which the dof were estimated to be 6.3. Figure 7 indicates that the initial estimates yield the most heavy tailed model. This seems reasonable since all initial estimates for the dof range between 0.9 and 5.3 (with average 2.8). The models obtained from the MLEs exhibit the smallest tail probability, indicating that these are the least heavy tailed models considered here. This is in line with Figure 6, which shows that the dof are substantially larger than the initial estimates. The ungrouped t copula is more heavy tailed than the fitted grouped one (with MLEs) but less heavy tailed than the fitted grouped one with initial estimates.

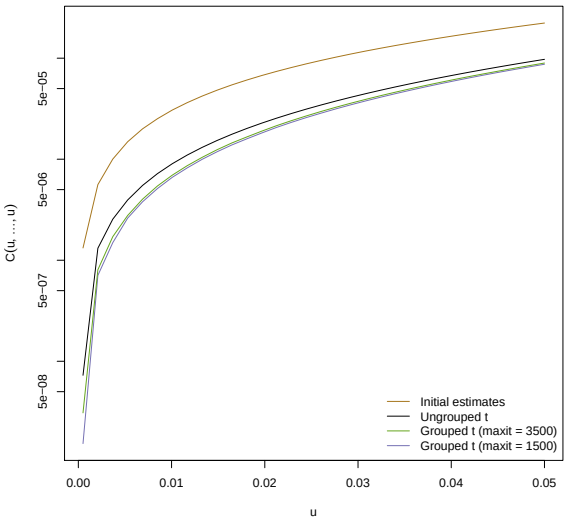

Figure 7. Estimated shortfall probability $C(u, \dots, u)$ for grouped and ungrouped t copulas fitted to the standardized residuals obtained from the Dow Jones 30 dataset from 1 January 2014 to 31 December 2015 after deGARCHing.

This example demonstrates that it is generally advisable to estimate the dof jointly when grouped modeling is of interest, rather than group-wise as suggested in Daul et al. (2003). Indeed, in this particular example, the initial estimates give a model that substantially overestimates the risk of jointly large losses. As can be seen from Figure 6, optimizing an estimated log-likelihood function is not at all trivial, in particular when many parameters are involved. Indeed, the underlying optimizer never

detected convergence, which is why the user needs to carefully assess which specification of `maxit` to use. We plan on exploring more elaborate optimization procedures which perform better in large dimensions for this problem in the future.

Example 3. *In this example, we consider the problem of mean-variance (MV) portfolio optimization in the classical Markowitz (1952) setting. Consider d assets, and denote by $\boldsymbol{\mu}_t$ and Σ_t the expected return vector on the risky assets in excess of the risk free rate and the variance-covariance (VCV) matrix of asset returns in the portfolio at time t, respectively. We assume that an investor chooses the weights $\boldsymbol{x}_t$ of the portfolio to maximize the quadratic utility function $U(\boldsymbol{x}_t) = \boldsymbol{x}_t^\top \boldsymbol{\mu}_t - \frac{\gamma}{2}\boldsymbol{x}_t^\top \Sigma_t \boldsymbol{x}_t$, where in what follows we assume the risk-aversion parameter $\gamma = 1$. When there are no shortselling (or other) constraints, one finds the optimal $\boldsymbol{x}_t$ as $\boldsymbol{x}_t = \Sigma_t^{-1}\boldsymbol{\mu}_t$. As in Low et al. (2016), we consider relative portfolio weights, which are thus given by*

$$\boldsymbol{w}_t = \frac{\Sigma_t^{-1}\boldsymbol{\mu}_t}{|\mathbf{1}^\top \Sigma_t^{-1}\boldsymbol{\mu}_t|}.$$

As such, the investor needs to estimate $\boldsymbol{\mu}_t$ and Σ_t. If we assume no shortselling, i.e., $x_{t,j} \geq 0$ for $j = 1, \dots, d$, the optimization problem can be solved numerically, for instance using the R package `quadprog` of Turlach et al. (2019).

Assume we have return data for the d assets stored in vectors $\boldsymbol{y}_t$, $t = 1, \dots, T$, and a sampling window $0 < M < T$. We perform an experiment similar to Low et al. (2016) and compare a historical approach with a model-based approach to estimate $\boldsymbol{\mu}_t$ and Σ_t. The main steps are as follows:

1. *In each period $t = M+1, \dots, T$, estimate $\boldsymbol{\mu}_t$ and Σ_t using the M previous return data $\boldsymbol{y}_i$, $i = t-M, \dots, t-1$.*
2. *Compute the optimal portfolio weights $\boldsymbol{w}_t$ and the out-of-sample return $r_t = \boldsymbol{w}_t^\top \boldsymbol{y}_t$.*

In the historical approach, $\boldsymbol{\mu}_t$ and Σ_t in the first step are merely computed as the sample mean vector and sample VCV matrix of the past return data. Our model-based approach is a simplification of the approach used in Low et al. (2016). In particular, to estimate $\boldsymbol{\mu}_t$ and Σ_t in the first step, the following is done in each time period:

1a. *Fit marginal $ARMA(1,1) - GARCH(1,1)$ models with standardized t innovations to $\boldsymbol{y}_i$, $i = t-M, \dots, t-1$.*
1b. *Extract the standardized residuals and fit a grouped t copula to the pseudo-observations thereof.*
1c. *Sample n vectors from the fitted copula, transform the margins by applying the quantile function of the respective standardized t distribution and based on these n d-dimensional residuals, sample from the fitted $ARMA(1,1) - GARCH(1,1)$ giving a total of n simulated return vectors, say $\boldsymbol{y}_i'$, $i = 1, \dots, n$.*
1d. *Estimate $\boldsymbol{\mu}_t$ and Σ_t from $\boldsymbol{y}_i'$, $i = 1, \dots, n$.*

The historical and model-based approaches each produce $T - M$ out-of-sample returns from which we can estimate the certainty-equivalent return (CER) and the Sharpe-ratio (SR) as

$$\widehat{CER} = \hat{\mu}_r - \frac{1}{2}\hat{\sigma}_r^2 \quad \text{and} \quad \widehat{SR} = \frac{\hat{\mu}_r}{\hat{\sigma}_r},$$

where $\hat{\mu}_r$ and $\hat{\sigma}_r$ denote the sample mean and sample standard deviation of the $T - M$ out-of-sample returns; see also Tu and Zhou (2011). Note that larger, positive values of the SR and CER indicate better portfolio performance.

We consider logarithmic returns of the constituents of the Dow Jones 30 index from 1 January 2013 to 31 December 2014 ($n = 503$ data points obtained from the R package `qrmdata` of Hofert and Hornik (2016)), a sampling window of $M = 250$ days, $n = 10^4$ samples to estimate $\boldsymbol{\mu}_t$ and Σ_t in the model-based approach, a risk-free interest rate of zero and no transaction costs. We report (in percent) the point estimates $\hat{\mu}_r$, $\widehat{CER}$ and $\widehat{SR}$ for the historical approach and for the model-based approach based on an ungrouped and grouped t copula in Table 2 assuming no shortselling. To limit the run time for this illustrative example, the degrees of freedom for the grouped and ungrouped t copula are estimated once and held fixed throughout all time periods $t = M+1, \dots, T$. We see that the point estimates for the grouped model exceed the point estimates for the ungrouped model.

Table 2. Estimated returns, certainty-equivalent returns (CERs) and Sharpe-ratios (SRs) (all in percentage points) of a mean-variance (MV) investor under three investment rules assuming no short-sales: Model-based with grouped and ungrouped t copula and historical.

	$\hat{\mu}_r$	$\widehat{\text{CER}}$	$\widehat{\text{SR}}$
historical	0.045	0.041	5.356
ungrouped t	0.075	0.071	8.678
grouped t	0.083	0.079	9.474

7. Discussion and Conclusions

We introduced the class of grouped normal variance mixtures and provided efficient algorithms to work with this class of distributions: Estimating the distribution function and log-density function, estimating the copula and its density, estimating Spearman's rho and Kendall's tau and estimating the parameters of a grouped NVM copula given a dataset. Most algorithms (and functions in the package `nvmix`) merely require one to provide the quantile function(s) of the mixing distributions. Due to their importance in practice, algorithms presented in this paper (and their implementation in the R package `nvmix`) are widely applicable in practice.

We saw that the distribution function (and hence, the copula) of grouped NVM distributions can be efficiently estimated even in high dimensions using RQMC algorithms. The density function of grouped NVM distributions is in general not available in closed form, not even for the grouped t distribution, so one relies on its estimation. Our proposed adaptive algorithm is capable of estimating the log-density even in high dimensions accurately and efficiently. Fitting grouped normal variance mixture copulas, such as the grouped t copula, to data is an important yet challenging task due to lack of a tractable density function. Thanks to our adaptive procedure for estimating the density, the parameters can be estimated jointly in the special case of a grouped t copula. As was demonstrated in the previous section, it is indeed advisable to estimate the dof jointly, as otherwise one might severely over- or underestimate the joint tails.

A computational challenge that we plan to further investigate is the optimization of the estimated log-likelihood function, which is currently slow and lacks a reliable convergence criterion that can be used for automation. Another avenue for future research is to study how one can, for a given multivariate dataset, assign the components to homogeneous groups.

8. Proofs

Proof of Proposition 1. This is an immediate application of a proposition proven in (Daul et al. 2003, p. 6). □

Proof of Proposition 2. Assume that $\Sigma = P$ is a correlation matrix with $P_{12} = \rho$ (otherwise standardize the margins). We can write

$$X_1 = \sqrt{W_1} Z_1, \quad X_2 = \sqrt{W_2} \left(\rho Z_1 + \sqrt{1-\rho^2} Z_2 \right),$$

where $\boldsymbol{Z} = (Z_1, Z_2) \sim \mathrm{N}_2(\boldsymbol{0}, I_2)$ and $W_j = F^{\leftarrow}_{W_j}(U)$ for $j = 1, 2$ and $U \sim \mathrm{U}(0,1)$. Let $\boldsymbol{S} = (S_1, S_2) \sim \mathrm{U}(\mathbb{S}^{2-1})$ be uniformly distributed on the unit circle. Such random vector $\boldsymbol{S}$ can be expressed via the stochastic representation

$$S_1 = \cos(\Theta), \quad S_2 = \sin(\Theta), \quad \Theta \sim \mathrm{U}([-\pi, \pi)).$$

Note that $\boldsymbol{Z}/\sqrt{\boldsymbol{Z}^\top \boldsymbol{Z}} \sim \mathrm{U}(\mathbb{S}^{2-1})$ is independent of $\boldsymbol{Z}^\top \boldsymbol{Z} \sim \chi^2_2$. Let $W_3^2 \sim \chi^2_2$ be independent of $\boldsymbol{S}$, W_1, W_2 and set $W_j' = \sqrt{W_j} W_3$ for $j = 1, 2$. Then

$$X_1 = W_1' S_1 = W_1' \cos(\Theta), \quad X_2 = W_2' \left(\rho S_1 + \sqrt{1-\rho^2} S_2 \right) = W_2' \left(\rho \cos(\Theta) + \sqrt{1-\rho^2} \sin(\Theta) \right).$$

Finally, setting $\phi = \arcsin \rho$ and noting that $\mathbb{P}(W_j' > 0) = 1$ for $j = 1, 2$ allows us to compute $\mathbb{P}(X_1 > 0, X_2 > 0)$ via

$$\begin{aligned}
&\mathbb{P}(X_1 > 0, X_2 > 0) = \mathbb{P}\left(W_1' \cos(\Theta) > 0, W_2' \left(\rho \cos(\Theta) + \sqrt{1-\rho^2} \sin(\Theta) \right) > 0 \right) \\
&= \mathbb{P}\left(\cos(\Theta) > 0,\ \sin(\phi)\cos(\Theta) + \cos(\phi)\sin(\Theta) > 0 \right) = \mathbb{P}\left(\cos(\Theta) > 0,\ \sin(\Theta + \phi) > 0 \right) \\
&= \mathbb{P}\left(\Theta \in (-\pi/2, \pi/2) \cap (-\phi, \pi - \phi) \right) = \frac{\pi/2 + \phi}{2\pi}.
\end{aligned}$$

Substituting $\phi = \arcsin(\rho)$ gives the result. □

Proof of Proposition 3. We follow (McNeil et al. 2015, Proposition 7.44), where a similar result was shown for $\boldsymbol{X} \sim \mathrm{NVM}_2(\boldsymbol{\mu}, \Sigma, F_W)$. Since $\boldsymbol{X}$ has continuous margins, we can write

$$\rho_S(X_1, X_2) = 6\mathbb{P}((X_1 - \tilde{X}_1)(X_2 - \overline{X}_2) > 0) - 3, \tag{31}$$

where $\tilde{X}_1$ and $\overline{X}_2$ are random variables with $\tilde{X}_1 \overset{\mathrm{d}}{=} X_1$ and $\overline{X}_2 \overset{\mathrm{d}}{=} X_2$ and where (X_1, X_2), $\tilde{X}_1$ and $\overline{X}_2$ are all independent; see (McNeil et al. 2015, Proposition 7.34).

Let $W_j = F_{W_j}^{\leftarrow}(U)$ for $j = 1, 2$, $\tilde{W}_1 = F_{W_1}^{\leftarrow}(\tilde{U})$ and $\overline{W}_2 = F_{W_2}^{\leftarrow}(\overline{U})$ for $U, \tilde{U}, \overline{U} \overset{\text{ind.}}{\sim} \mathrm{U}(0,1)$. Furthermore, let $\tilde{Z}, \overline{Z} \overset{\text{ind.}}{\sim} \mathrm{N}(0,1)$, independent of $\boldsymbol{Z} \sim \mathrm{N}_2(0, P)$. Finally, set

$$X_1 = \sqrt{W_1} Z_1, \quad X_2 = \sqrt{W_2} Z_2, \quad \tilde{X} = \sqrt{\tilde{W}_1} \tilde{Z}, \quad \overline{X} = \sqrt{\overline{W}_2} \overline{Z}.$$

and define the random vector $\boldsymbol{Y} = (Y_1, Y_2)$ via

$$Y_1 = X_1 - \tilde{X} = \sqrt{W_1} Z_1 - \sqrt{\tilde{W}_1} \tilde{Z}, \quad Y_2 = X_2 - \overline{X} = \sqrt{W_2} Z_2 - \sqrt{\overline{W}_2} \overline{Z}.$$

From (31) it follows that we need to compute $\mathbb{P}(Y_1 Y_2 > 0)$. Conditional on $U, \tilde{U}, \overline{U}$, we find

$$\boldsymbol{Y} \mid U, \tilde{U}, \overline{U} \sim \mathrm{N}_2\left(0, \begin{pmatrix} W_1 + \tilde{W}_1 & \sqrt{W_1 W_2}\rho \\ \sqrt{W_1 W_2}\rho & W_2 + \overline{W}_2 \end{pmatrix} \right).$$

Since the multivariate normal distribution is a special case of a gNVM distribution (obtained with $W_1 = \cdots = W_d = c$ for some $c > 0$), Proposition 2 can be applied to compute

$$\begin{aligned}
\rho_S(X_1, X_2) &= 6\mathbb{P}\left((X_1 - \tilde{X}_1)(X_2 - \overline{X}_2) > 0 \right) - 3 = 3\left(\mathbb{E}\left(2\mathbb{P}(Y_1 Y_2 > 0 \mid U, \tilde{U}, \overline{U}) \right) - 1 \right) \\
&= 3\left(\mathbb{E}\left(4\mathbb{P}(Y_1 > 0, Y_2 > 0 \mid U, \tilde{U}, \overline{U}) \right) - 1 \right) \\
&= \frac{6}{\pi} \mathbb{E}\left(\arcsin\left(\rho \sqrt{\frac{F_{W_1}^{\leftarrow}(U) F_{W_2}^{\leftarrow}(U)}{(F_{W_1}^{\leftarrow}(U) + F_{W_1}^{\leftarrow}(\tilde{U}))(F_{W_2}^{\leftarrow}(U) + F_{W_2}^{\leftarrow}(\bar{U}))}} \right) \right).
\end{aligned}$$

□

Proof of Proposition 4. For $u, q \in (0,1)$, let $x_j(u,q) = F_j^{\leftarrow}(q) / \sqrt{F_{W_j}^{\leftarrow}(u)}$ for $j = 1, 2$. Writing C instead of $C_{P,F_W}^{\mathrm{gNVM}}$ and Φ_ρ for the distribution function of $\boldsymbol{Z} \sim \mathrm{N}_2(\boldsymbol{0}, P)$, we get $C(q,q) = \int_0^1 \Phi_\rho(x_1(u,q), x_2(u,q))\, \mathrm{d}u$, which implies that

$$\lambda = \lim_{q \to 0^+} \frac{1}{q} C(q,q) = \lim_{q \to 0^+} \frac{\partial}{\partial q} \int_0^1 \Phi_\rho\left(x_1(u,q), x_2(u,q) \right) \mathrm{d}u.$$

Taking the derivative with respect to q gives

$$\frac{\partial}{\partial q}\int_0^1 \Phi_\rho\left(x_1(u,q), x_2(u,q)\right)\,\mathrm{d}u = \int_0^1 \frac{\partial \Phi_\rho}{\partial x_1}\frac{\partial x_1}{\partial q}\,\mathrm{d}u + \int_0^1 \frac{\partial \Phi_\rho}{\partial x_2}\frac{\partial x_2}{\partial q}\,\mathrm{d}u. \tag{32}$$

We find $\frac{\partial x_i}{\partial q} = \left(\sqrt{F^{\leftarrow}_{W_i}(u)}\, f_i\left(F_i^{\leftarrow}(q)\right)\right)^{-1}$ for $i = 1, 2$. Furthermore,

$$\frac{\partial}{\partial x_1}\Phi_\rho(x_1, x_2) = \frac{\partial}{\partial x_1}\mathbb{P}(Z_1 \le x_1, Z_2 \le x_2) = \phi(x_1)\mathbb{P}(Z_2 \le x_2 \mid Z_1 = x_1) = \phi(x_1)\Phi\left(\frac{x_2 - \rho x_1}{\sqrt{1-\rho^2}}\right);$$

the derivative $\frac{\partial}{\partial x_2}\Phi_\rho(x_1, x_2)$ can be found analogously by swapping the roles of Z_1 and Z_2. Plugging the derivatives into (32) and taking the limit gives the result in the statement. □

Author Contributions: Methodology, E.H., M.H., C.L.; software, E.H., M.H., C.L.; validation, E.H., M.H., C.L.; formal analysis, E.H., M.H., C.L.; data curation, E.H., M.H.; writing–original draft preparation, E.H.; writing–review and editing, E.H., M.H., C.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by NSERC Discovery Grant RGPIN-2020-04897 and NSERC Discovery Grant RGPIN-238959.

Acknowledgments: We thank five anonymous reviewers for their comments and suggestions, which helped us improve this paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

Caflisch, Russel, William Morokoff, and Art Owen. 1997. *Valuation of Mortgage Backed Securities Using Brownian Bridges to Reduce Effective Dimension*. Los Angeles: Department of Mathematics, University of California.

Cao, Jian, Marc Genton, David Keyes, and George Turkiyyah. 2020. Exploiting low rank covariance structures for computing high-dimensional normal and student-*t* probabilities. *arXiv*, arXiv:2003.11183.

Daul, Stéphane, Enrico De Giorgi, Filip Lindskog, and Alexander McNeil. 2003. The Grouped t-Copula with an Application to Credit Risk. SSRN 1358956. Available online: https://ssrn.com/abstract=1358956 (accessed on 21 August 2020).

Demarta, Stefano, and Alexander McNeil. 2005. The t copula and related copulas. *International Statistical Review* 73: 111–29. [CrossRef]

Devroye, Luc. 1986. *Non-Uniform Random Variate Generation*. New York: Springer.

Embrechts, Paul, Filip Lindskog, and Alexander McNeil. 2001. *Modelling Dependence with Copulas*. Rapport Technique. Zurich: Département de Mathématiques, Institut Fédéral de Technologie de Zurich.

Genz, Alan 1992. Numerical computation of multivariate normal probabilities. *Journal of Computational and Graphical Statistics* 1: 141–49.

Genz, A., and Frank Bretz. 1999. Numerical computation of multivariate *t*-probabilities with application to power calculation of multiple contrasts. *Journal of Statistical Computation and Simulation* 63: 103–17. [CrossRef]

Genz, Alan, and Frank Bretz. 2002. Comparison of methods for the computation of multivariate *t* probabilities. *Journal of Computational and Graphical Statistics* 11: 950–71. [CrossRef]

Genz, Alan, and Frank Bretz. 2009. *Computation of Multivariate Normal and t Probabilities*. Berlin: Springer, vol. 195.

Gibson, Garvin Jarvis, Chris Glasbey, and David Elston. 1994. Monte Carlo evaluation of multivariate normal integrals and sensitivity to variate ordering. In *Advances in Numerical Methods and Applications*. River Edge: World Scientific Publishing, pp. 120–26.

Glasserman, Paul 2013. *Monte Carlo Methods in Financial Engineering*. Berlin: Springer, vol. 53.

Hintz, Erik, Marius Hofert, and Christiane Lemieux. 2020. Normal Variance Mixtures: Distribution, Density and Parameter Estimation. *arXiv*, arXiv:1911.03017.

Hofert, Marius, Erik Hintz, and Christiane Lemieux. 2020. nvmix: Multivariate Normal Variance Mixtures. R Package Version 0.0.5. Available online: https://cran.r-project.org/package=nvmix (accessed on 21 August 2020).

Hofert, Marius, Kurt Hornik, and Alexander McNeil. 2016. qrmdata: Data Sets for Quantitative Risk Management Practice. R Package Version 2016-01-03-1. Available online: https://cran.r-project.org/package=qrmdata (accessed on 21 August 2020).

Hofert, Marius, and Christiane Lemieux. 2019. qrng: (Randomized) Quasi-Random Number Generators. R Package Version 0.0.7. Available online: https://cran.r-project.org/package=qrng (accessed on 21 August 2020).

Lemieux, Christiane. 2009. *Monte Carlo and Quasi-Monte Carlo Sampling*. Berlin: Springer.

Lindskog, Filip, Alexander Mcneil, and Uwe Schmock. 2003. Kendall's tau for elliptical distributions. In *Credit Risk*. Berlin: Springer, pp. 149–56.

Low, Rand, Robert Faff, and Kjersti Aas. 2016. Enhancing mean–variance portfolio selection by modeling distributional asymmetries. *Journal of Economics and Business* 85: 49–72. [CrossRef]

Luo, Xiaolin, and Pavel Shevchenko. 2010. The t copula with multiple parameters of degrees of freedom: Bivariate characteristics and application to risk management. *Quantitative Finance* 10: 1039–54. [CrossRef]

Markowitz, Harry. 1952. Portfolio selection. *Journal of Finance* 7: 77–91.

McNeil, Alexander, Rüdiger Frey, and Paul Embrechts. 2015. *Quantitative Risk Management: Concepts, Techniques and Tools*. Princeton: Princeton University Press.

Nelsen, Roger. 2007. *An Introduction to Copulas*. Berlin: Springer.

Niederreiter, Harald. 1992. *Random Number Generation and quasi-Monte Carlo Methods*. Philadelphia: SIAM, vol. 63.

Piessens, Robert, Elise de Doncker-Kapenga, Christoph Überhuber, and David Kahaner. 2012. *Quadpack: A Subroutine Package for Automatic Integration*. Berlin: Springer, vol. 1.

Tu, Jun, and Guofu Zhou. 2011. Markowitz meets talmud: A combination of sophisticated and naive diversification strategies. *Journal of Financial Economics* 99: 204–15. [CrossRef]

Turlach, Berwin, Andreas Weingessel, and Cleve Moler. 2019. quadprog: Functions to Solve Quadratic Programming Problems. R Package Version 1.5.8. Available online: https://cran.r-project.org/package=quadprog (accessed on 21 August 2020).

Venter, Gary, Jack Barnett, Rodney Kreps, and John Major. 2007. Multivariate copulas for financial modeling. *Variance* 1: 103–19.

Wang, Xiaoqun, and Ian. Sloan. 2005. Why are high-dimensional finance problems often of low effective dimension? *SIAM Journal on Scientific Computing* 27: 159–83. [CrossRef]