Simulation of Gaussian copulas

Simulation of Gaussian copulas

Observe: Consider a symmetric positive definite matrix $R \in \mathbb{R}^{d \times d}$ and its Cholesky factorization $AA^T = R$ with $A \in \mathbb{R}^{d \times d}$. If $Z_1, Z_2, \ldots, Z_d \sim N(0,1)$ are independent, then $\mu + AZ \sim N_d(\mu, R)$.

Simulation of Gaussian copulas

Observe: Consider a symmetric positive definite matrix $R \in \mathbb{R}^{d \times d}$ and its Cholesky factorization $AA^T = R$ with $A \in \mathbb{R}^{d \times d}$. If $Z_1, Z_2, \ldots, Z_d \sim N(0,1)$ are independent, then $\mu + AZ \sim N_d(\mu, R)$.

Algorithm: for the generation of a random vector $U = (U_1, U_2, \ldots, U_d)$ whose distribution function is the copula C_R^{Ga}, R positive definite with all ones on the main diagonal.

- Compute the Cholesly factorization of R: $R = AA^T$.

Simulation of Gaussian copulas

Observe: Consider a symmetric positive definite matrix $R \in \mathbb{R}^{d \times d}$ and its Cholesky factorization $AA^T = R$ with $A \in \mathbb{R}^{d \times d}$. If $Z_1, Z_2, \ldots, Z_d \sim N(0,1)$ are independent, then $\mu + AZ \sim N_d(\mu, R)$.

Algorithm: for the generation of a random vector $U = (U_1, U_2, \ldots, U_d)$ whose distribution function is the copula C_R^{Ga}, R positive definite with all ones on the main diagonal.

- Compute the Cholesly factorization of R: $R = AA^T$.
- Simulate d independent standard normally distributed r.v. $Z_1, Z_2, \ldots, Z_d \sim N(0,1)$

Simulation of Gaussian copulas

Observe: Consider a symmetric positive definite matrix $R \in \mathbb{R}^{d\times d}$ and its Cholesky factorization $AA^T = R$ with $A \in \mathbb{R}^{d\times d}$. If $Z_1, Z_2, \ldots, Z_d \sim N(0,1)$ are independent, then $\mu + AZ \sim N_d(\mu, R)$.

Algorithm: for the generation of a random vector $U = (U_1, U_2, \ldots, U_d)$ whose distribution function is the copula C_R^{Ga}, R positive definite with all ones on the main diagonal.

- Compute the Cholesly factorization of R: $R = AA^T$.
- Simulate d independent standard normally distributed r.v. $Z_1, Z_2, \ldots, Z_d \sim N(0,1)$
- Set $X := AZ$

Simulation of Gaussian copulas

Observe: Consider a symmetric positive definite matrix $R \in \mathbb{R}^{d \times d}$ and its Cholesky factorization $AA^T = R$ with $A \in \mathbb{R}^{d \times d}$. If $Z_1, Z_2, \dots, Z_d \sim N(0,1)$ are independent, then $\mu + AZ \sim N_d(\mu, R)$.

Algorithm: for the generation of a random vector $U = (U_1, U_2, \dots, U_d)$ whose distribution function is the copula C_R^{Ga}, R positive definite with all ones on the main diagonal.

- Compute the Cholesly factorization of R: $R = AA^T$.
- Simulate d independent standard normally distributed r.v. $Z_1, Z_2, \dots, Z_d \sim N(0,1)$
- Set $X := AZ$
- Set $U_k := \phi(X_k)$ for $k = 1, 2, \dots, d$, where ϕ is the standard normal distribution function.

Simulation of Gaussian copulas

Observe: Consider a symmetric positive definite matrix $R \in \mathbb{R}^{d\times d}$ and its Cholesky factorization $AA^T = R$ with $A \in \mathbb{R}^{d\times d}$. If $Z_1, Z_2, \ldots, Z_d \sim N(0,1)$ are independent, then $\mu + AZ \sim N_d(\mu, R)$.

Algorithm: for the generation of a random vector $U = (U_1, U_2, \ldots, U_d)$ whose distribution function is the copula C_R^{Ga}, R positive definite with all ones on the main diagonal.

- Compute the Cholesly factorization of R: $R = AA^T$.
- Simulate d independent standard normally distributed r.v. $Z_1, Z_2, \ldots, Z_d \sim N(0,1)$
- Set $X := AZ$
- Set $U_k := \phi(X_k)$ for $k = 1, 2, \ldots, d$, where ϕ is the standard normal distribution function.
- Output $U = (U_1, U_2, \ldots, U_d)$; U has distribution function C_R^{Ga}.

Simulation of t-copulas

Simulation of t-copulas

Algorithm: for the generation of a random vector $U = (U_1, U_2, \ldots, U_d)$ whose distribution function is the copula $C^t_{\nu,R}$, R positive definite with all ones on the main diagonal, $\nu \in \mathbb{N}$.

- Compute the Cholesly factorization of R: $R = AA^T$.

Simulation of t-copulas

Algorithm: for the generation of a random vector $U = (U_1, U_2, \dots, U_d)$ whose distribution function is the copula $C^t_{\nu,R}$, R positive definite with all ones on the main diagonal, $\nu \in \mathbb{N}$.

- Compute the Cholesly factorization of R: $R = AA^T$.
- Simulate d independent standard normally distributed r.v. $Z_1, Z_2, \dots, Z_d \sim N(0, 1)$

Simulation of t-copulas

Algorithm: for the generation of a random vector $U = (U_1, U_2, \ldots, U_d)$ whose distribution function is the copula $C^t_{\nu,R}$, R positive definite with all ones on the main diagonal, $\nu \in \mathbb{N}$.

- Compute the Cholesly factorization of R: $R = AA^T$.
- Simulate d independent standard normally distributed r.v. $Z_1, Z_2, \ldots, Z_d \sim N(0,1)$
- Simulate a r.v. $S \sim \chi^2_\nu$ independent from von $Z_1, \ldots, Z_d$.

Simulation of t-copulas

Algorithm: for the generation of a random vector $U = (U_1, U_2, \ldots, U_d)$ whose distribution function is the copula $C^t_{\nu,R}$, R positive definite with all ones on the main diagonal, $\nu \in \mathbb{N}$.

- Compute the Cholesly factorization of R: $R = AA^T$.
- Simulate d independent standard normally distributed r.v. $Z_1, Z_2, \ldots, Z_d \sim N(0,1)$
- Simulate a r.v. $S \sim \chi^2_\nu$ independent from von $Z_1, \ldots, Z_d$.
- Set $Y := AZ$

Simulation of t-copulas

Algorithm: for the generation of a random vector $U = (U_1, U_2, \ldots, U_d)$ whose distribution function is the copula $C^t_{\nu,R}$, R positive definite with all ones on the main diagonal, $\nu \in \mathbb{N}$.

- Compute the Cholesly factorization of R: $R = AA^T$.
- Simulate d independent standard normally distributed r.v. $Z_1, Z_2, \ldots, Z_d \sim N(0,1)$
- Simulate a r.v. $S \sim \chi^2_\nu$ independent from von $Z_1, \ldots, Z_d$.
- Set $Y := AZ$
- Set $X := \frac{\sqrt{\nu}}{\sqrt{S}} Y$

Simulation of t-copulas

Algorithm: for the generation of a random vector $U = (U_1, U_2, \dots, U_d)$ whose distribution function is the copula $C^t_{\nu,R}$, R positive definite with all ones on the main diagonal, $\nu \in \mathbb{N}$.

- Compute the Cholesly factorization of R: $R = AA^T$.
- Simulate d independent standard normally distributed r.v. $Z_1, Z_2, \dots, Z_d \sim N(0,1)$
- Simulate a r.v. $S \sim \chi^2_\nu$ independent from von $Z_1, \dots, Z_d$.
- Set $Y := AZ$
- Set $X := \frac{\sqrt{\nu}}{\sqrt{S}} Y$
- Set $U_k = t_\nu(X_k)$ for $k = 1, 2, \dots, d$, where t_ν is the distribution function of a standard t-distribution with ν degrees of freedom.

Simulation of t-copulas

Algorithm: for the generation of a random vector $U = (U_1, U_2, \ldots, U_d)$ whose distribution function is the copula $C^t_{\nu,R}$, R positive definite with all ones on the main diagonal, $\nu \in \mathbb{N}$.

- Compute the Cholesly factorization of R: $R = AA^T$.
- Simulate d independent standard normally distributed r.v. $Z_1, Z_2, \ldots, Z_d \sim N(0,1)$
- Simulate a r.v. $S \sim \chi^2_\nu$ independent from von $Z_1, \ldots, Z_d$.
- Set $Y := AZ$
- Set $X := \frac{\sqrt{\nu}}{\sqrt{S}} Y$
- Set $U_k = t_\nu(X_k)$ for $k = 1, 2, \ldots, d$, where t_ν is the distribution function of a standard t-distribution with ν degrees of freedom.
- Output $U = (U_1, U_2, \ldots, U_d)$; $U = (U_1, U_2, \ldots, U_d)$ has distribution function $C^t_{\nu,R}$.

Simulation of Archimedian copulas

Simulation of Archimedian copulas

A generic algorithm to generate a random vector $U = (U_1, U_2, \ldots, U_d)$ with a given Archimedian copula as cumulative distribution function

Simulation of Archimedean copulas

A generic algorithm to generate a random vector $U = (U_1, U_2, \ldots, U_d)$ with a given Archimedean copula as cumulative distribution function

Input: The dimension $d \in \mathbb{N}$, the Archimedean Copula $C(u) = \varphi^{-1}(\varphi(u_1) + \varphi(u_2) + \ldots + \varphi(u_d))$ specified in terms of its generator φ.

Simulation of Archimedean copulas

A generic algorithm to generate a random vector $U = (U_1, U_2, \ldots, U_d)$ with a given Archimedean copula as cumulative distribution function

Input: The dimension $d \in \mathbb{N}$, the Archimedean Copula $C(u) = \varphi^{-1}(\varphi(u_1) + \varphi(u_2) + \ldots + \varphi(u_d))$ specified in terms of its generator φ.

- Simulate a variable X with distribution function G, such that the Laplace-Stieltjes transform ψ of G is the inverse function of the generator φ of the input copula, $\psi = \varphi^{-1}$.

Simulation of Archimedian copulas

A generic algorithm to generate a random vector $U = (U_1, U_2, \ldots, U_d)$ with a given Archimedian copula as cumulative distribution function

Input: The dimension $d \in \mathbb{N}$, the Archimedian Copula $C(u) = \varphi^{-1}(\varphi(u_1) + \varphi(u_2) + \ldots + \varphi(u_d))$ specified in terms of its generator φ.

- Simulate a variable X with distribution function G, such that the Laplace-Stieltjes transform ψ of G is the inverse function of the generator φ of the input copula, $\psi = \varphi^{-1}$.
- Simulate d i.i.d. r.v. V_1,V_2,..., V_d uniformly distributed on $[0, 1]$.

Simulation of Archimedian copulas

A generic algorithm to generate a random vector $U = (U_1, U_2, \ldots, U_d)$ with a given Archimedian copula as cumulative distribution function

Input: The dimension $d \in \mathbb{N}$, the Archimedian Copula $C(u) = \varphi^{-1}(\varphi(u_1) + \varphi(u_2) + \ldots + \varphi(u_d))$ specified in terms of its generator φ.

- Simulate a variable X with distribution function G, such that the Laplace-Stieltjes transform ψ of G is the inverse function of the generator φ of the input copula, $\psi = \varphi^{-1}$.
- Simulate d i.i.d. r.v. V_1,V_2,..., V_d uniformly distributed on $[0, 1]$.
- Set $U = (\psi(-\ln(V_1)/X), \psi(-\ln(V_2)/X), \ldots, \psi(-\ln(V_d)/X))$. The distribution function of U is C.

Simulation of Archimedian copulas

A generic algorithm to generate a random vector $U = (U_1, U_2, \dots, U_d)$ with a given Archimedian copula as cumulative distribution function

Input: The dimension $d \in \mathbb{N}$, the Archimedian Copula $C(u) = \varphi^{-1}(\varphi(u_1) + \varphi(u_2) + \ldots + \varphi(u_d))$ specified in terms of its generator φ.

- Simulate a variable X with distribution function G, such that the Laplace-Stieltjes transform ψ of G is the inverse function of the generator φ of the input copula, $\psi = \varphi^{-1}$.
- Simulate d i.i.d. r.v. V_1,V_2,..., V_d uniformly distributed on $[0, 1]$.
- Set $U = (\psi(-\ln(V_1)/X), \psi(-\ln(V_2)/X), \dots, \psi(-\ln(V_d)/X))$. The distribution function of U is C.

Output: U

Simulation of Archimedian copulas

A generic algorithm to generate a random vector $U = (U_1, U_2, \ldots, U_d)$ with a given Archimedian copula as cumulative distribution function

Input: The dimension $d \in \mathbb{N}$, the Archimedian Copula $C(u) = \varphi^{-1}(\varphi(u_1) + \varphi(u_2) + \ldots + \varphi(u_d))$ specified in terms of its generator φ.

- Simulate a variable X with distribution function G, such that the Laplace-Stieltjes transform ψ of G is the inverse function of the generator φ of the input copula, $\psi = \varphi^{-1}$.
- Simulate d i.i.d. r.v. V_1,V_2,. . ., V_d uniformly distributed on $[0, 1]$.
- Set $U = (\psi(-\ln(V_1)/X), \psi(-\ln(V_2)/X), \ldots, \psi(-\ln(V_d)/X))$. The distribution function of U is C.

Output: U

The generator $\varphi(t) = (t^{-\theta} - 1)/\theta$, $\theta > 0$ yields the Clayton copula C_θ^{Cl}.

Simulation of Archimedian copulas

A generic algorithm to generate a random vector $U = (U_1, U_2, \ldots, U_d)$ with a given Archimedian copula as cumulative distribution function

Input: The dimension $d \in \mathbb{N}$, the Archimedian Copula $C(u) = \varphi^{-1}(\varphi(u_1) + \varphi(u_2) + \ldots + \varphi(u_d))$ specified in terms of its generator φ.

- Simulate a variable X with distribution function G, such that the Laplace-Stieltjes transform ψ of G is the inverse function of the generator φ of the input copula, $\psi = \varphi^{-1}$.
- Simulate d i.i.d. r.v. $V_1, V_2, \ldots, V_d$ uniformly distributed on $[0, 1]$.
- Set $U = (\psi(-\ln(V_1)/X), \psi(-\ln(V_2)/X), \ldots, \psi(-\ln(V_d)/X))$. The distribution function of U is C.

Output: U

The generator $\varphi(t) = (t^{-\theta} - 1)/\theta$, $\theta > 0$ yields the Clayton copula C_θ^{Cl}. Alternatively also $\tilde{\varphi}(t) = t^{-\theta} - 1$ is a generator of the Clayton copula.

Simulation of Archimedian copulas

A generic algorithm to generate a random vector $U = (U_1, U_2, \ldots, U_d)$ with a given Archimedian copula as cumulative distribution function

Input: The dimension $d \in \mathbb{N}$, the Archimedian Copula $C(u) = \varphi^{-1}(\varphi(u_1) + \varphi(u_2) + \ldots + \varphi(u_d))$ specified in terms of its generator φ.

- Simulate a variable X with distribution function G, such that the Laplace-Stieltjes transform ψ of G is the inverse function of the generator φ of the input copula, $\psi = \varphi^{-1}$.
- Simulate d i.i.d. r.v. V_1,V_2,..., V_d uniformly distributed on $[0, 1]$.
- Set $U = (\psi(-\ln(V_1)/X), \psi(-\ln(V_2)/X), \ldots, \psi(-\ln(V_d)/X))$. The distribution function of U is C.

Output: U

The generator $\varphi(t) = (t^{-\theta} - 1)/\theta$, $\theta > 0$ yields the Clayton copula C_θ^{Cl}. Alternatively also $\tilde{\varphi}(t) = t^{-\theta} - 1$ is a generator of the Clayton copula.

For $X \sim Gamma(1/\theta, 1)$ with d.f. $f_X(x) = \left(x^{1/\theta-1}e^{-x}\right)/\Gamma(1/\theta)$ we have: $E(e^{-sX}) = \int_0^\infty e^{-sx}\frac{1}{\Gamma(1/\theta)}x^{1/\theta-1}e^{-x}dx = (s+1)^{-1/\theta} = \tilde{\varphi}^{-1}(s)$.

Simulation of the Clayton copula ($\theta > 0$)

Simulation of the Clayton copula ($\theta > 0$)

A generic algorithm to generate a random vector $U = (U_1, U_2, \ldots, U_d)$ with the Clayton C_θ^{Cl} copula as distribution function.

Simulation of the Clayton copula ($\theta > 0$)

A generic algorithm to generate a random vector $U = (U_1, U_2, \ldots, U_d)$ with the Clayton C_θ^{Cl} copula as distribution function.

Input: The dimension $d \in \mathbb{N}$, the parameter $\theta > 0$.

Simulation of the Clayton copula ($\theta > 0$)

A generic algorithm to generate a random vector $U = (U_1, U_2, \ldots, U_d)$ with the Clayton C_θ^{Cl} copula as distribution function.

Input: The dimension $d \in \mathbb{N}$, the parameter $\theta > 0$.

- Simulate $X \sim Gamma(1/\theta, 1)$.

Simulation of the Clayton copula ($\theta > 0$)

A generic algorithm to generate a random vector $U = (U_1, U_2, \dots, U_d)$ with the Clayton C_θ^{Cl} copula as distribution function.

Input: The dimension $d \in \mathbb{N}$, the parameter $\theta > 0$.

- Simulate $X \sim \mathit{Gamma}(1/\theta, 1)$.
- Set $\psi(s) := (s+1)^{-\frac{1}{\theta}}$ for $s \geq 0$.

Simulation of the Clayton copula ($\theta > 0$)

A generic algorithm to generate a random vector $U = (U_1, U_2, \ldots, U_d)$ with the Clayton C_θ^{Cl} copula as distribution function.

Input: The dimension $d \in \mathbb{N}$, the parameter $\theta > 0$.

- Simulate $X \sim Gamma(1/\theta, 1)$.
- Set $\psi(s) := (s+1)^{-\frac{1}{\theta}}$ for $s \geq 0$.
- Simulate d i.i.d. r.v. V_1,V_2,..., V_d uniformly distributed on $[0, 1]$.

Simulation of the Clayton copula ($\theta > 0$)

A generic algorithm to generate a random vector $U = (U_1, U_2, \ldots, U_d)$ with the Clayton C_θ^{Cl} copula as distribution function.

Input: The dimension $d \in \mathbb{N}$, the parameter $\theta > 0$.

- Simulate $X \sim \mathit{Gamma}(1/\theta, 1)$.
- Set $\psi(s) := (s+1)^{-\frac{1}{\theta}}$ for $s \geq 0$.
- Simulate d i.i.d. r.v. V_1,V_2,..., V_d uniformly distributed on $[0, 1]$.
- The distribution function of

$$U = (\psi(-\ln(V_1)/X), \psi(-\ln(V_2)/X), \ldots, \psi(-\ln(V_d)/X))$$

is the Clayton copula C_θ^{Cl}.

Simulation of the Clayton copula ($\theta > 0$)

A generic algorithm to generate a random vector $U = (U_1, U_2, \ldots, U_d)$ with the Clayton C_θ^{Cl} copula as distribution function.

Input: The dimension $d \in \mathbb{N}$, the parameter $\theta > 0$.

- Simulate $X \sim Gamma(1/\theta, 1)$.
- Set $\psi(s) := (s+1)^{-\frac{1}{\theta}}$ for $s \geq 0$.
- Simulate d i.i.d. r.v. V_1,V_2,…, V_d uniformly distributed on $[0, 1]$.
- The distribution function of

  $$U = (\psi(-\ln(V_1)/X), \psi(-\ln(V_2)/X), \ldots, \psi(-\ln(V_d)/X))$$

  is the Clayton copula C_θ^{Cl}.

Output: U

Simulation of the Gumbel copula ($\theta \geq 1$)

Simulation of the Gumbel copula ($\theta \geq 1$)

Let X be a positive stable r.v., $X \sim St(1/\theta, 1, \gamma, 0)$ with $\gamma = (\cos(\pi/(2\theta)))^{\theta} > 0$ (and $\alpha = \frac{1}{\theta}$, $\beta = 1$, $\delta = 0$)

Simulation of the Gumbel copula ($\theta \geq 1$)

Let X be a positive stable r.v., $X \sim St(1/\theta, 1, \gamma, 0)$ with $\gamma = (\cos(\pi/(2\theta)))^\theta > 0$ (and $\alpha = \frac{1}{\theta}$, $\beta = 1$, $\delta = 0$)
The Laplace-Stieltjes transform of F_X is the generator $\varphi(t) = \exp(-t^{1/\theta})$ of the Gumbel copula C_θ^{Gu}.

Simulation of the Gumbel copula ($\theta \geq 1$)

Let X be a positive stable r.v., $X \sim St(1/\theta, 1, \gamma, 0)$ with $\gamma = (\cos(\pi/(2\theta)))^\theta > 0$ (and $\alpha = \frac{1}{\theta}$, $\beta = 1$, $\delta = 0$)
The Laplace-Stieltjes transform of F_X is the generator $\varphi(t) = \exp(-t^{1/\theta})$ of the Gumbel copula C_θ^{Gu}.

The simulation of $Z \sim ST(\alpha, \beta, 1, 0)$ is not straightforward (see Nolan 2002).

Simulation of the Gumbel copula ($\theta \geq 1$)

Let X be a positive stable r.v., $X \sim St(1/\theta, 1, \gamma, 0)$ with $\gamma = (\cos(\pi/(2\theta)))^\theta > 0$ (and $\alpha = \frac{1}{\theta}$, $\beta = 1$, $\delta = 0$)
The Laplace-Stieltjes transform of F_X is the generator $\varphi(t) = \exp(-t^{1/\theta})$ of the Gumbel copula C_θ^{Gu}.

The simulation of $Z \sim ST(\alpha, \beta, 1, 0)$ is not straightforward (see Nolan 2002).

For $\alpha \neq 1$ we get: $X = \delta + \gamma Z \sim St(\alpha, \beta, \gamma, \delta)$.

Simulation of the Gumbel copula ($\theta \geq 1$)

Let X be a positive stable r.v., $X \sim St(1/\theta, 1, \gamma, 0)$ with $\gamma = (\cos(\pi/(2\theta)))^\theta > 0$ (and $\alpha = \frac{1}{\theta}$, $\beta = 1$, $\delta = 0$)
The Laplace-Stieltjes transform of F_X is the generator $\varphi(t) = \exp(-t^{1/\theta})$ of the Gumbel copula C_θ^{Gu}.

The simulation of $Z \sim ST(\alpha, \beta, 1, 0)$ is not straightforward (see Nolan 2002).

For $\alpha \neq 1$ we get: $X = \delta + \gamma Z \sim St(\alpha, \beta, \gamma, \delta)$.

The case $\alpha = 1$ is more complicated.

Simulation of the Gumbel copula ($\theta \geq 1$)

Let X be a positive stable r.v., $X \sim St(1/\theta, 1, \gamma, 0)$ with $\gamma = (\cos(\pi/(2\theta)))^\theta > 0$ (and $\alpha = \frac{1}{\theta}$, $\beta = 1$, $\delta = 0$)
The Laplace-Stieltjes transform of F_X is the generator $\varphi(t) = \exp(-t^{1/\theta})$ of the Gumbel copula C_θ^{Gu}.

The simulation of $Z \sim ST(\alpha, \beta, 1, 0)$ is not straightforward (see Nolan 2002).

For $\alpha \neq 1$ we get: $X = \delta + \gamma Z \sim St(\alpha, \beta, \gamma, \delta)$.

The case $\alpha = 1$ is more complicated.

Alternative approach:
Let $\theta \geq 1$ and $\bar{F}(x) = 1 - F(x) = \exp(-x^{1/\theta})$ for $x \geq 0$.

Simulation of the Gumbel copula ($\theta \geq 1$)

Let X be a positive stable r.v., $X \sim St(1/\theta, 1, \gamma, 0)$ with $\gamma = (\cos(\pi/(2\theta)))^\theta > 0$ (and $\alpha = \frac{1}{\theta}$, $\beta = 1$, $\delta = 0$)
The Laplace-Stieltjes transform of F_X is the generator $\varphi(t) = \exp(-t^{1/\theta})$ of the Gumbel copula C_θ^{Gu}.

The simulation of $Z \sim ST(\alpha, \beta, 1, 0)$ is not straightforward (see Nolan 2002).

For $\alpha \neq 1$ we get: $X = \delta + \gamma Z \sim St(\alpha, \beta, \gamma, \delta)$.

The case $\alpha = 1$ is more complicated.

Alternative approach:
Let $\theta \geq 1$ and $\bar{F}(x) = 1 - F(x) = \exp(-x^{1/\theta})$ for $x \geq 0$. Let $V \sim U(0,1)$ and let S be a r.v. independent from V with density function $h(s) = (1 - 1/\theta + s/\theta)\exp(-s)$ for $s \geq 0$.

Simulation of the Gumbel copula ($\theta \geq 1$)

Let X be a positive stable r.v., $X \sim St(1/\theta, 1, \gamma, 0)$ with $\gamma = (\cos(\pi/(2\theta)))^\theta > 0$ (and $\alpha = \frac{1}{\theta}$, $\beta = 1$, $\delta = 0$)
The Laplace-Stieltjes transform of F_X is the generator $\varphi(t) = \exp(-t^{1/\theta})$ of the Gumbel copula C_θ^{Gu}.

The simulation of $Z \sim ST(\alpha, \beta, 1, 0)$ is not straightforward (see Nolan 2002).

For $\alpha \neq 1$ we get: $X = \delta + \gamma Z \sim St(\alpha, \beta, \gamma, \delta)$.

The case $\alpha = 1$ is more complicated.

Alternative approach:
Let $\theta \geq 1$ and $\bar{F}(x) = 1 - F(x) = \exp(-x^{1/\theta})$ for $x \geq 0$. Let $V \sim U(0,1)$ and let S be a r.v. independent from V with density function $h(s) = (1 - 1/\theta + s/\theta)\exp(-s)$ for $s \geq 0$.

Set $(Z_1, Z_2)^T := (VS^\theta, (1-V)S^\theta)^T$.

Simulation of the Gumbel copula ($\theta \geq 1$)

Let X be a positive stable r.v., $X \sim St(1/\theta, 1, \gamma, 0)$ with $\gamma = (\cos(\pi/(2\theta)))^\theta > 0$ (and $\alpha = \frac{1}{\theta}$, $\beta = 1$, $\delta = 0$)
The Laplace-Stieltjes transform of F_X is the generator $\varphi(t) = \exp(-t^{1/\theta})$ of the Gumbel copula C_θ^{Gu}.

The simulation of $Z \sim ST(\alpha, \beta, 1, 0)$ is not straightforward (see Nolan 2002).

For $\alpha \neq 1$ we get: $X = \delta + \gamma Z \sim St(\alpha, \beta, \gamma, \delta)$.

The case $\alpha = 1$ is more complicated.

Alternative approach:
Let $\theta \geq 1$ and $\bar{F}(x) = 1 - F(x) = \exp(-x^{1/\theta})$ for $x \geq 0$. Let $V \sim U(0,1)$ and let S be a r.v. independent from V with density function $h(s) = (1 - 1/\theta + s/\theta)\exp(-s)$ for $s \geq 0$.

Set $(Z_1, Z_2)^T := (VS^\theta, (1-V)S^\theta)^T$.
The distribution function of $(\bar{F}(Z_1), \bar{F}(Z_2))^T$ is C_θ^{Gu}. Convince yourself!

Simulation of the Gumbel copula ($\theta \geq 1$) (contd.)

Simulation of the Gumbel copula ($\theta \geq 1$) (contd.)

Algorithm to generate a random vector $U = (U_1, U_2, \ldots, U_d)$ with the Gumbel copula C_θ^{Gu} as distribution function.

Simulation of the Gumbel copula ($\theta \geq 1$) (contd.)

Algorithm to generate a random vector $U = (U_1, U_2, \ldots, U_d)$ with the Gumbel copula C_θ^{Gu} as distribution function.
Input: The dimension $d \in \mathbb{N}$, the parameter $\theta \geq 1$.

Simulation of the Gumbel copula ($\theta \geq 1$) (contd.)

Algorithm to generate a random vector $U = (U_1, U_2, \ldots, U_d)$ with the Gumbel copula C_θ^{Gu} as distribution function.
Input: The dimension $d \in \mathbb{N}$, the parameter $\theta \geq 1$.

- Simulate two i.i.d. r.v. $V_1, V_2 \sim U(0, 1)$.

Simulation of the Gumbel copula ($\theta \geq 1$) (contd.)

Algorithm to generate a random vector $U = (U_1, U_2, \ldots, U_d)$ with the Gumbel copula C_θ^{Gu} as distribution function.
Input: The dimension $d \in \mathbb{N}$, the parameter $\theta \geq 1$.

- Simulate two i.i.d. r.v. $V_1, V_2 \sim U(0,1)$.
- Simulate two independent r.v. W_1, W_2 with $W_1 \sim \Gamma(1,1)$, $W_2 \sim \Gamma(2,1)$
- Set $S := I_{V_2 \leq 1/\theta} W_1 + I_{V_2 > 1/\theta} W_2$.
- Set $(Z_1, Z_2) := (V_1 S^\theta, (1 - V_1) S^\theta)$.
- The distribution function of $U = \left(\exp(-Z_1^{1/\theta}), \exp(-Z_2^{1/\theta}) \right)^T$ is C_θ^{Gu}.

Simulation of the Gumbel copula ($\theta \geq 1$) (contd.)

Algorithm to generate a random vector $U = (U_1, U_2, \ldots, U_d)$ with the Gumbel copula C_θ^{Gu} as distribution function.
Input: The dimension $d \in \mathbb{N}$, the parameter $\theta \geq 1$.

- Simulate two i.i.d. r.v. $V_1, V_2 \sim U(0,1)$.
- Simulate two independent r.v. W_1, W_2 with $W_1 \sim \Gamma(1,1)$, $W_2 \sim \Gamma(2,1)$
- Set $S := I_{V_2 \leq 1/\theta} W_1 + I_{V_2 > 1/\theta} W_2$.
- Set $(Z_1, Z_2) := (V_1 S^\theta, (1 - V_1) S^\theta)$.
- The distribution function of $U = \left(\exp(-Z_1^{1/\theta}), \exp(-Z_2^{1/\theta})\right)^T$ is C_θ^{Gu}.

Output: U

Calibration of copulas

Calibration of copulas

Goal: Determine a copula and the marginal distributions to model a given multi-dimensional data set.

Calibration of copulas

Goal: Determine a copula and the marginal distributions to model a given multi-dimensional data set.

Input: A sample $\{X_1, X_2, \ldots, X_d\}$ of a c.d.f. F with continuous marginal distributions F_1, F_2,..., F_d.

Calibration of copulas

Goal: Determine a copula and the marginal distributions to model a given multi-dimensional data set.

Input: A sample $\{X_1, X_2, \ldots, X_d\}$ of a c.d.f. F with continuous marginal distributions F_1, F_2,..., F_d.

Output: A copula C_θ and an estimator $\hat{\theta}$ for the parameter vector θ of the copula C_θ such which $F(x) \approx C_{\hat{\theta}}(F_1(x_1), \ldots, F_d(x_d))$ holds.

Calibration of copulas

Goal: Determine a copula and the marginal distributions to model a given multi-dimensional data set.

Input: A sample $\{X_1, X_2, \ldots, X_d\}$ of a c.d.f. F with continuous marginal distributions F_1, F_2,..., F_d.

Output: A copula C_θ and an estimator $\hat{\theta}$ for the parameter vector θ of the copula C_θ such which $F(x) \approx C_{\hat{\theta}}(F_1(x_1), \ldots, F_d(x_d))$ holds.

Question 1: Which family of (known) copulas to use?

Calibration of copulas

Goal: Determine a copula and the marginal distributions to model a given multi-dimensional data set.

Input: A sample $\{X_1, X_2, \ldots, X_d\}$ of a c.d.f. F with continuous marginal distributions F_1, F_2,..., F_d.

Output: A copula C_θ and an estimator $\hat{\theta}$ for the parameter vector θ of the copula C_θ such which $F(x) \approx C_{\hat{\theta}}(F_1(x_1), \ldots, F_d(x_d))$ holds.

Question 1: Which family of (known) copulas to use?

Answer: Selection of a suitable family of copulas based on (a) the visual comparison of the graphical representations of the data set on one side and of known copulas on the other, and (b) the empirical tail dependence coefficients.

Calibration of copulas

Goal: Determine a copula and the marginal distributions to model a given multi-dimensional data set.

Input: A sample $\{X_1, X_2, \ldots, X_d\}$ of a c.d.f. F with continuous marginal distributions F_1, F_2,..., F_d.

Output: A copula C_θ and an estimator $\hat{\theta}$ for the parameter vector θ of the copula C_θ such which $F(x) \approx C_{\hat{\theta}}(F_1(x_1), \ldots, F_d(x_d))$ holds.

Question 1: Which family of (known) copulas to use?

Answer: Selection of a suitable family of copulas based on (a) the visual comparison of the graphical representations of the data set on one side and of known copulas on the other, and (b) the empirical tail dependence coefficients.

Question 2: What are the parameters of the prespecified family of copulas used for the modelling?

Parameter estimation for C_R^{Ga}, $C_{\nu,R}^t$, C_θ^{Cl} and C_θ^{Gu}

$C_R^{Ga} = \phi_R^d(\phi^{-1}(u_1), \ldots, \phi^{-1}(u_d))$

$R_{ij} = \sin(\pi(\rho_\tau)_{ij}/2)$

Parameter estimation for C_R^{Ga}, $C_{\nu,R}^t$, C_θ^{Cl} and C_θ^{Gu}

$$C_R^{Ga} = \phi_R^d(\phi^{-1}(u_1), \ldots, \phi^{-1}(u_d)) \qquad R_{ij} = \sin(\pi(\rho_\tau)_{ij}/2)$$

$$C_{\nu,R}^t = t_{\nu,R}^d(t_\nu^{-1}(u_1), \ldots, t_\nu^{-1}(u_d)) \qquad R_{ij} = \sin(\pi(\rho_\tau)_{ij}/2)$$

Parameter estimation for C_R^{Ga}, $C_{\nu,R}^t$, C_θ^{Cl} and C_θ^{Gu}

$$C_R^{Ga} = \phi_R^d(\phi^{-1}(u_1), \ldots, \phi^{-1}(u_d)) \qquad R_{ij} = \sin(\pi(\rho_\tau)_{ij}/2)$$

$$C_{\nu,R}^t = t_{\nu,R}^d(t_\nu^{-1}(u_1), \ldots, t_\nu^{-1}(u_d)) \qquad R_{ij} = \sin(\pi(\rho_\tau)_{ij}/2)$$

$$C_\theta^{Gu}(u) = \exp\left(-[(-\ln u_1)^\theta + \ldots + (-\ln u_d^\theta]^{1/\theta}\right) \qquad \theta = 1/(1-(\rho_\tau)_{ij})$$

Parameter estimation for C_R^{Ga}, $C_{\nu,R}^t$, C_θ^{Cl} and C_θ^{Gu}

$$C_R^{Ga} = \phi_R^d(\phi^{-1}(u_1), \ldots, \phi^{-1}(u_d)) \qquad R_{ij} = \sin(\pi(\rho_\tau)_{ij}/2)$$

$$C_{\nu,R}^t = t_{\nu,R}^d(t_\nu^{-1}(u_1), \ldots, t_\nu^{-1}(u_d)) \qquad R_{ij} = \sin(\pi(\rho_\tau)_{ij}/2)$$

$$C_\theta^{Gu}(u) = \exp\left(-[(-\ln u_1)^\theta + \ldots + (-\ln u_d^\theta]^{1/\theta}\right) \qquad \theta = 1/(1 - (\rho_\tau)_{ij}) \quad ,$$

$$C_\theta^{Cl}(u) = (u_1^{-\theta} + \ldots + u_d^{-\theta} - d + 1)^{-1/\theta} \qquad \theta = 2(\rho_\tau)_{ij}/(1 - (\rho_\tau)_{ij})$$

Parameter estimation for C_R^{Ga}, $C_{\nu,R}^t$, C_θ^{Cl} and C_θ^{Gu}

$$
\begin{array}{ll}
C_R^{Ga} = \phi_R^d(\phi^{-1}(u_1), \ldots, \phi^{-1}(u_d)) & R_{ij} = \sin(\pi(\rho_\tau)_{ij}/2) \\
C_{\nu,R}^t = t_{\nu,R}^d(t_\nu^{-1}(u_1), \ldots, t_\nu^{-1}(u_d)) & R_{ij} = \sin(\pi(\rho_\tau)_{ij}/2) \\
C_\theta^{Gu}(u) = \exp\left(-[(-\ln u_1)^\theta + \ldots + (-\ln u_d^\theta]^{1/\theta}\right) & \theta = 1/(1-(\rho_\tau)_{ij}) \\
C_\theta^{Cl}(u) = (u_1^{-\theta} + \ldots + u_d^{-\theta} - d + 1)^{-1/\theta} & \theta = 2(\rho_\tau)_{ij}/(1-(\rho_\tau)_{ij})
\end{array},
$$

where

$$
\begin{aligned}
(\rho_\tau)_{ij} &= \rho_\tau(X_{k,i}, X_{k,j}) \\
&= P((X_{k,i} - X_{l,i})(X_{k,j} - X_{l,j}) > 0) - P((X_{k,i} - X_{l,i})(X_{k,j} - X_{l,j}) < 0) \\
&= E(sign((X_{k,i} - X_{l,i})(X_{k,j} - X_{l,j}))).
\end{aligned}
$$

Parameter estimation for C_R^{Ga}, $C_{\nu,R}^t$, C_θ^{Cl} and C_θ^{Gu}

$$
\begin{array}{ll}
C_R^{Ga} = \phi_R^d(\phi^{-1}(u_1), \ldots, \phi^{-1}(u_d)) & R_{ij} = \sin(\pi(\rho_\tau)_{ij}/2) \\
C_{\nu,R}^t = t_{\nu,R}^d(t_\nu^{-1}(u_1), \ldots, t_\nu^{-1}(u_d)) & R_{ij} = \sin(\pi(\rho_\tau)_{ij}/2) \\
C_\theta^{Gu}(u) = \exp\left(-[(-\ln u_1)^\theta + \ldots + (-\ln u_d^\theta]^{1/\theta}\right) & \theta = 1/(1-(\rho_\tau)_{ij}) \\
C_\theta^{Cl}(u) = (u_1^{-\theta} + \ldots + u_d^{-\theta} - d + 1)^{-1/\theta} & \theta = 2(\rho_\tau)_{ij}/(1-(\rho_\tau)_{ij})
\end{array},
$$

where

$$
\begin{aligned}
(\rho_\tau)_{ij} &= \rho_\tau(X_{k,i}, X_{k,j}) \\
&= P((X_{k,i} - X_{l,i})(X_{k,j} - X_{l,j}) > 0) - P((X_{k,i} - X_{l,i})(X_{k,j} - X_{l,j}) < 0) \\
&= E(sign((X_{k,i} - X_{l,i})(X_{k,j} - X_{l,j}))).
\end{aligned}
$$

Standard empirical estimator of Kendalls Tau:
$\widehat{\rho_\tau}_{ij} = \binom{n}{2}^{-1} \sum_{1 \le k < l \le n} sign((X_{k,i} - X_{l,i})(X_{k,j} - X_{l,j}))$.

Calibration of the correlation matrix for Gaussian and t-copulas

Calibration of the correlation matrix for Gaussian and t-copulas

It may happen that $\hat{R} = (\hat{R}_{ij})$, $\hat{R}_{ij} = \sin(\pi \widehat{\rho_\tau}_{ij}/2)$, is not positive definite.

Calibration of the correlation matrix for Gaussian and t-copulas

It may happen that $\hat{R} = (\hat{R}_{ij})$, $\hat{R}_{ij} = \sin(\pi \widehat{\rho_\tau}_{ij}/2)$, is not positive definite.

Replace $\hat{R}$ by a correlation matrix R^*, selected such the "distance" between R^* and $\hat{R}$ is "small".

Calibration of the correlation matrix for Gaussian and t-copulas

It may happen that $\hat{R} = (\hat{R}_{ij})$, $\hat{R}_{ij} = \sin(\pi \widehat{\rho_\tau}_{ij}/2)$, is not positive definite.

Replace $\hat{R}$ by a correlation matrix R^*, selected such the "distance" between R^* and $\hat{R}$ is "small".

Eigenvalue approach (Rousseeuw and Molenberghs 1993)

- Compute the spectral decomposition $\hat{R} = \Gamma\Lambda\Gamma^T$ of $\hat{R}$, where Λ is a diagonal matrix, containing the eigenvalues of $\hat{R}$ on the diagonal, and Γ is an orthogonal matrix with the eigenvectors of $\hat{R}$ in its columns.

Calibration of the correlation matrix for Gaussian and t-copulas

It may happen that $\hat{R} = (\hat{R}_{ij})$, $\hat{R}_{ij} = \sin(\pi\widehat{\rho_\tau}_{ij}/2)$, is not positive definite.

Replace $\hat{R}$ by a correlation matrix R^*, selected such the "distance" between R^* and $\hat{R}$ is "small".

Eigenvalue approach (Rousseeuw and Molenberghs 1993)

- Compute the spectral decomposition $\hat{R} = \Gamma\Lambda\Gamma^T$ of $\hat{R}$, where Λ is a diagonal matrix, containing the eigenvalues of $\hat{R}$ on the diagonal, and Γ is an orthogonal matrix with the eigenvectors of $\hat{R}$ in its columns.
- Replace the negative eigenvalues in Λ by some small number $\delta > 0$ to obtain $\tilde{\Lambda}$.

Calibration of the correlation matrix for Gaussian and t-copulas

It may happen that $\hat{R} = (\hat{R}_{ij})$, $\hat{R}_{ij} = \sin(\pi \widehat{\rho_\tau}_{ij}/2)$, is not positive definite.

Replace $\hat{R}$ by a correlation matrix R^*, selected such the "distance" between R^* and $\hat{R}$ is "small".

Eigenvalue approach (Rousseeuw and Molenberghs 1993)

- Compute the spectral decomposition $\hat{R} = \Gamma\Lambda\Gamma^T$ of $\hat{R}$, where Λ is a diagonal matrix, containing the eigenvalues of $\hat{R}$ on the diagonal, and Γ is an orthogonal matrix with the eigenvectors of $\hat{R}$ in its columns.
- Replace the negative eigenvalues in Λ by some small number $\delta > 0$ to obtain $\tilde{\Lambda}$.
- Compute $\tilde{R} = \Gamma\tilde{\Lambda}\Gamma^T$. $\tilde{R}$ is symmetric and positive definite but not necessarily a correlation matrix; the diagonal elements $\hat{R}_{ii}$ might be unequal 1.

Calibration of the correlation matrix for Gaussian and t-copulas

It may happen that $\hat{R} = (\hat{R}_{ij})$, $\hat{R}_{ij} = \sin(\pi \widehat{\rho_\tau}_{ij}/2)$, is not positive definite.

Replace $\hat{R}$ by a correlation matrix R^*, selected such the "distance" between R^* and $\hat{R}$ is "small".

Eigenvalue approach (Rousseeuw and Molenberghs 1993)

- Compute the spectral decomposition $\hat{R} = \Gamma \Lambda \Gamma^T$ of $\hat{R}$, where Λ is a diagonal matrix, containing the eigenvalues of $\hat{R}$ on the diagonal, and Γ is an orthogonal matrix with the eigenvectors of $\hat{R}$ in its columns.
- Replace the negative eigenvalues in Λ by some small number $\delta > 0$ to obtain $\tilde{\Lambda}$.
- Compute $\tilde{R} = \Gamma \tilde{\Lambda} \Gamma^T$. $\tilde{R}$ is symmetric and positive definite but not necessarily a correlation matrix; the diagonal elements $\hat{R}_{ii}$ might be unequal 1.
- Set $R^* := D\tilde{R}D$ where D is a diagonal matrix with $D_{k,k} = 1/\sqrt{\tilde{R}_{k,k}}$.

Estimation of the number of the degrees of freedom ν for t-copulas

Estimation of the number of the degrees of freedom ν for t-copulas

1. Let $\hat{F}_1, \ldots, \hat{F}_d$ be the estimated marginal distributions.

Estimation of the number of the degrees of freedom ν for t-copulas

1. Let $\hat{F}_1$, ..., $\hat{F}_d$ be the estimated marginal distributions.
2. Generate a pseudo-sample of the copula

$$\hat{U}_k = (\hat{U}_{k,1}, \hat{U}_{k,2}, \ldots, \hat{U}_{k,d}) := \Big(\hat{F}_1(X_{k,1}), \ldots, \hat{F}_d(X_{k,d})\Big),$$

for $k = 1, 2, \ldots, n$ (see Genest und Rivest 1993).

Estimation of the number of the degrees of freedom ν for t-copulas

1. Let $\hat{F}_1$, ..., $\hat{F}_d$ be the estimated marginal distributions.
2. Generate a pseudo-sample of the copula

$$\hat{U}_k = (\hat{U}_{k,1}, \hat{U}_{k,2}, \dots, \hat{U}_{k,d}) := \Big(\hat{F}_1(X_{k,1}), \dots, \hat{F}_d(X_{k,d})\Big),$$

   for $k = 1, 2, \dots, n$ (see Genest und Rivest 1993).

$\hat{F}_k$ can be generated by :

- a parametric estimation method;
  $\hat{F}_k$ is assumed to be a certain parametric distribution and the parameter is estimated by a maximum likelihood (ML) approach

Estimation of the number of the degrees of freedom ν for t-copulas

1. Let $\hat{F}_1$, ..., $\hat{F}_d$ be the estimated marginal distributions.
2. Generate a pseudo-sample of the copula

$$\hat{U}_k = (\hat{U}_{k,1}, \hat{U}_{k,2}, \ldots, \hat{U}_{k,d}) := \Big(\hat{F}_1(X_{k,1}), \ldots, \hat{F}_d(X_{k,d})\Big),$$

   for $k = 1, 2, \ldots, n$ (see Genest und Rivest 1993).

$\hat{F}_k$ can be generated by :

- a parametric estimation method;
  $\hat{F}_k$ is assumed to be a certain parametric distribution and the parameter is estimated by a maximum likelihood (ML) approach
- a non-parametric estimation method;
  $\hat{F}_i$ is the empirical distribution function $\hat{F}_i(x) = \frac{1}{n+1}\sum_{t=1}^{n} I_{\{X_{t,i} \leq x\}}$, $1 \leq i \leq d$.

Estimation of the number of the degrees of freedom ν for t-copulas (contd.)

Estimation of the number of the degrees of freedom ν for t-copulas (contd.)

Maximum likelihood estimator of ν: $\nu = \arg\max_{\xi} \ln L(\xi; \hat{U}_1, \hat{U}_2, \ldots, \hat{U}_n)$

Estimation of the number of the degrees of freedom ν for t-copulas (contd.)

Maximum likelihood estimator of ν: $\nu = \arg\max_\xi \ln L(\xi; \hat{U}_1, \hat{U}_2, \ldots, \hat{U}_n)$ where

$$L(\xi; \hat{U}_1, \hat{U}_2, \ldots, \hat{U}_n) = \Pi_{k=1}^n c_{\xi,R}^t(\hat{U}_k)$$

and $c_{\xi,R}^t$ is the density of the t-copula $C_{\xi,R}^t$.

This implies

$$\ln L(\xi; \hat{U}_1, \hat{U}_2, \ldots, \hat{U}_n) =$$

$$\sum_{k=1}^{n} \ln g_{\xi,R}(t_\xi^{-1}(\hat{U}_{k,1}), \ldots, t_\xi^{-1}(\hat{U}_{k,d})) - \sum_{k=1}^{n}\sum_{j=1}^{d} \ln g_\xi(t_\xi^{-1}(\hat{U}_{k,j})),$$

Estimation of the number of the degrees of freedom ν for t-copulas (contd.)

Maximum likelihood estimator of ν: $\nu = \arg\max_{\xi} \ln L(\xi; \hat{U}_1, \hat{U}_2, \ldots, \hat{U}_n)$ where

$$L(\xi; \hat{U}_1, \hat{U}_2, \ldots, \hat{U}_n) = \Pi_{k=1}^{n} c_{\xi,R}^{t}(\hat{U}_k)$$

and $c_{\xi,R}^{t}$ is the density of the t-copula $C_{\xi,R}^{t}$.

This implies

$$\ln L(\xi; \hat{U}_1, \hat{U}_2, \ldots, \hat{U}_n) =$$

$$\sum_{k=1}^{n} \ln g_{\xi,R}(t_{\xi}^{-1}(\hat{U}_{k,1}), \ldots, t_{\xi}^{-1}(\hat{U}_{k,d})) - \sum_{k=1}^{n} \sum_{j=1}^{d} \ln g_{\xi}(t_{\xi}^{-1}(\hat{U}_{k,j})),$$

where $g_{\xi,R}$ is the cumulative density function of a d-dimensional standard t-distribution with ξ degrees of freedom and correlation matrix R, and g_{ξ} is the density function of a univariate standard t-distribution with ξ degrees of freedom.