Copula models

H. A. Mohtashami-Borzadaran N. Doodman M. Amini

Ferdowsi University of Mashhad

Webpage: https://hamb8066.github.io/IntroCopula

Outline

In this part, we are going to investigate the behaviour of different copula models.

- Elliptical copula
- Archimedean copula
- Extreme-value copula
- Frechet-Hoeffding copula
- Khoudraji copula
- Some remarks

Elliptical copula: some sub-families

Two well-known copulas induced from the family of elliptical distributions are

- Normal copula,
- T copula.

We will investigate the behavior of these two copulas.

Elliptical copula: Normal copula

Let $\underline{X} \sim N_d(\underline{0}, P)$ and P be its covariance matrix, then the normal copula associated with $\underline{X}$ is

$$C(u_1, u_2) = \Phi_P\Big(\Phi^{-1}(u_1), \Phi^{-1}(u_2)\Big),$$

Where Φ^{-1} denotes the quantile function of $N(0, 1)$.

Properties

- Covers positive & negative dependece
- No tail dependence ($\lambda_l = \lambda_u = 0$)
- Exchangeable
- Radially symmetric

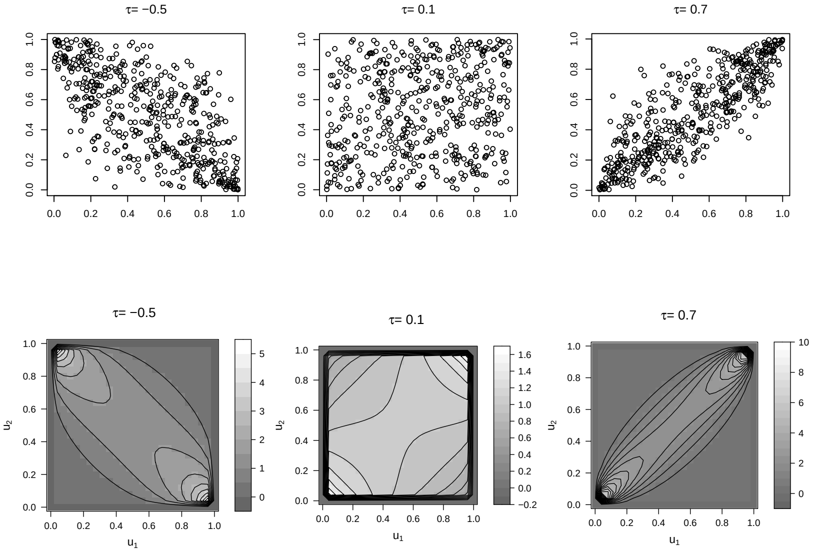
Elliptical copula: Normal copula

Let ρ be the Pearson correlation coefficient. Then for Kendall's tau and tail dependence measures we have:

$$\rho_s = \frac{6}{\pi}\arcsin(\rho/2) \qquad \text{and} \qquad \tau = \frac{2}{\pi}\arcsin(\rho)$$

```
#library(copula)
#nc1 <- normalCopula(iTau(normalCopula() ,tau=-0.5))
#nc2 <- normalCopula(iTau(normalCopula() ,tau=0.1))
#nc3 <- normalCopula(iTau(normalCopula() ,tau=0.7))
#par(mfrow=c(1,3))
#plot(rCopula(500,copula=nc1),xlab="",ylab="",main=expression(paste(tau,"= -0.5")))
#plot(rCopula(500,copula=nc2),xlab="",ylab="",main=expression(paste(tau,"= 0.1")))
#plot(rCopula(500,copula=nc3),xlab="",ylab="",main=expression(paste(tau,"= 0.7")))
#par(mfrow=c(1,1))
#
#p1=contourplot2(nc1, dCopula,main=expression(paste(tau,"= -0.5")))
#p2=contourplot2(nc2, dCopula,main=expression(paste(tau,"= 0.1")))
#p3=contourplot2(nc3, dCopula,main=expression(paste(tau,"= 0.7")))
#library(gridExtra)
#grid.arrange(p1, p2,p3, nrow=1,ncol=3)
```

Elliptical copula: Normal copula



Elliptical copula: T copula

Let $\underline{X} \sim T_{d,df}(\underline{0}, P)$ and P be its covariance matrix, then the T-copula associated with $\underline{X}$ is

$$C(u_1, u_2) = \Phi_{P,df}\Big(\Phi^{-1}(u_1), \Phi^{-1}(u_2)\Big),$$

Where Φ^{-1} denotes the quantile function of $T_{df}(0, 1)$.

Properties

- Covers positive & negative dependece
- Has lower and upper tail dependence and they are equal
- Exchangeable
- Radially symmetric

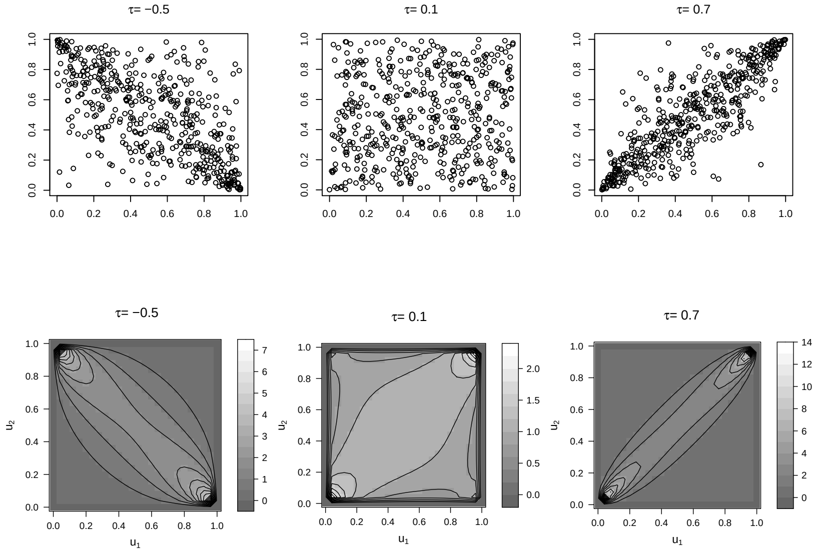
Elliptical copula: T copula

Let ρ be the Pearson correlation coefficient. Then for Kendall's tau and tail dependence measures we have:

$$\tau = \frac{2}{\pi}\arcsin(\rho) \quad \text{and} \quad \lambda_l = \lambda_u = 2t_{df+1}\Big(-\sqrt{\frac{(df+1)(1-\rho)}{1+\rho}}\Big)$$

```
#library(copula)
#nc1 <- tCopula(iTau(tCopula() ,tau=-0.5))
#nc2 <- tCopula(iTau(tCopula() ,tau=0.1))
#nc3 <- tCopula(iTau(tCopula() ,tau=0.7))
#par(mfrow=c(1,3))
#plot(rCopula(500,copula=t1),xlab="",ylab="",main=expression(paste(tau,"= -0.5")))
#plot(rCopula(500,copula=t2),xlab="",ylab="",main=expression(paste(tau,"= 0.1")))
#plot(rCopula(500,copula=t3),xlab="",ylab="",main=expression(paste(tau,"= 0.7")))
#par(mfrow=c(1,1))
#
#p1=contourplot2(t1, dCopula,main=expression(paste(tau,"= -0.5")))
#p2=contourplot2(t2, dCopula,main=expression(paste(tau,"= 0.1")))
#p3=contourplot2(t3, dCopula,main=expression(paste(tau,"= 0.7")))
#library(gridExtra)
#grid.arrange(p1, p2,p3, nrow=1,ncol=3)
```

Elliptical copula: T copula



Archimedean copula

A copula C is called Archimedean if it admits the representation

$$C(u_1, \ldots, u_d; \theta) = \psi^{[-1]}(\psi(u_1; \theta) + \ldots + \psi(u_d; \theta); \theta)$$

where $\psi : [0, 1] \times \Theta \to [0, \infty)$ is a continuous, strictly decreasing and convex function such that $\psi(1; \theta) = 0$.

Note: The Frank copula family is the only bivariate radial symmetric Archimedean family (Joe 2014 pp.65).

```
#Archimedean generator
#C(u) = psi(psi^{-1}(u_1)+...+psi^{-1}(u_d)), u in [0,1]^d.
#psi(claytonCopula(2),0.2)
#psi(claytonCopula(2),seq(0,1,0.1))
#iPsi(claytonCopula(2),seq(0.1,0.9,0.1))
#diPsi(claytonCopula(2),0.2) #first two derivatives of iPsi()
```

Archimedean copula: Important sub-models

- Frank copula
- Clayton copula
- AMH copula
- Gumbel copula (Also Extreme-value copula)
- Joe copula
- BB1 copula
- BB7 copula

Archimedean copula: Frank copula

The bivariate Frank copula is given as

$$C(u_1, u_2) = -\frac{1}{\delta}\Big(\frac{1}{1-e^{-\delta}}[(1-e^{-\delta}) - (1-e^{-\delta u_1})(1-e^{-\delta u_2})]\Big),$$

where $\delta \in [-\infty, \infty] - \{0\}$. For $\delta \to 0^+$ we have independence.

Properties

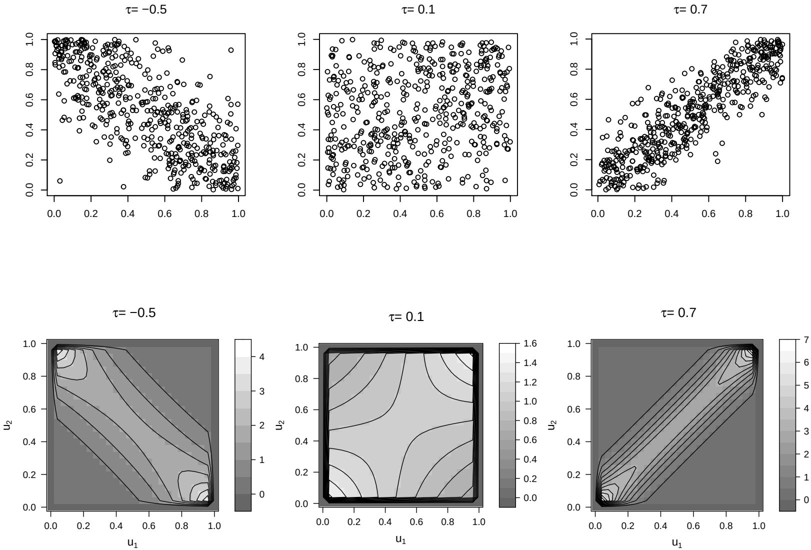
- Covers positive & negative dependece
- No tail dependence
- Exchangeable
- Radially symmetric

Archimedean copula: Frank copula

```r
library(copula)
#fk1 <- frankCopula(iTau(frankCopula() ,tau=-0.5))
#fk2 <- frankCopula(iTau(frankCopula() ,tau=0.1))
#fk3 <- frankCopula(iTau(frankCopula() ,tau=0.7))
#par(mfrow=c(1,3))
#plot(rCopula(500,copula=fk1),xlab="",ylab="",main=expression(paste(tau,"= -0.5")))
#plot(rCopula(500,copula=fk2),xlab="",ylab="",main=expression(paste(tau,"= 0.1")))
#plot(rCopula(500,copula=fk3),xlab="",ylab="",main=expression(paste(tau,"= 0.7")))
#par(mfrow=c(1,1))
```

```r
#p1=contourplot2(fk1, dCopula,main=expression(paste(tau,"= -0.5")))
#p2=contourplot2(fk2, dCopula,main=expression(paste(tau,"= 0.1")))
#p3=contourplot2(fk3, dCopula,main=expression(paste(tau,"= 0.7")))
#library(gridExtra)
#grid.arrange(p1, p2,p3, nrow=1,ncol=3)
```

Archimedean copula: Frank copula



Archimedean copula: Clayton copula

The bivariate Clayton copula is given as

$$C(u_1, u_2) = \max\left\{u_1^{-\theta} + u_2^{-\theta} - 1, 0\right\}; \qquad (u_1, u_2) \in [0,1]^2$$

where $\theta \in [-1, \infty] - \{0\}$.

Properties

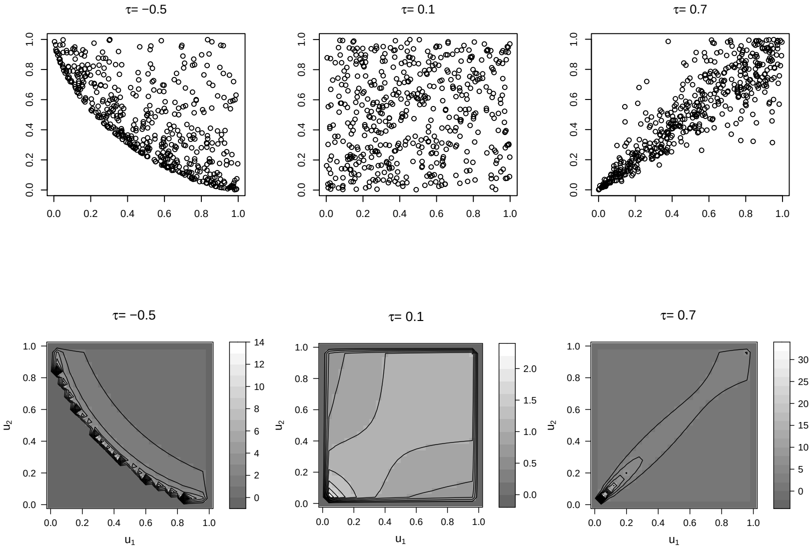
- Covers positive & negative dependece
- has only lower tail dependence when $\theta > 0$
- Exchangeable
- Not Radially symmetric

Archimedean copula: Clayton copula

```r
#library(copula)
#cl1 <- claytonCopula(iTau(claytonCopula() ,tau=-0.5))
#cl2 <- claytonCopula(iTau(claytonCopula() ,tau=0.1))
#cl3 <- claytonCopula(iTau(claytonCopula() ,tau=0.7))
#par(mfrow=c(1,3))
#plot(rCopula(500,copula=cl1),xlab="",ylab="",main=expression(paste(tau,"= -0.5")))
#plot(rCopula(500,copula=cl2),xlab="",ylab="",main=expression(paste(tau,"= 0.1")))
#plot(rCopula(500,copula=cl3),xlab="",ylab="",main=expression(paste(tau,"= 0.7")))
#par(mfrow=c(1,1))
```

```r
#p1=contourplot2(cl1, dCopula,main=expression(paste(tau,"= -0.5")))
#p2=contourplot2(cl2, dCopula,main=expression(paste(tau,"= 0.1")))
#p3=contourplot2(cl3, dCopula,main=expression(paste(tau,"= 0.7")))
#library(gridExtra)
#grid.arrange(p1, p2,p3, nrow=1,ncol=3)
```

Archimedean copula: Clayton copula



Archimedean copula: AMH copula

The bivariate Ali–Mikhail–Haq (AMH) copula is derived when

$$\psi(t) = \frac{1-\theta}{\exp(t)-\theta}; \quad \theta \in [0,1).$$

Properties

- Covers positive & negative dependece
- No tail dependence
- Exchangeable
- Not Radially symmetric

Archimedean copula: AMH copula

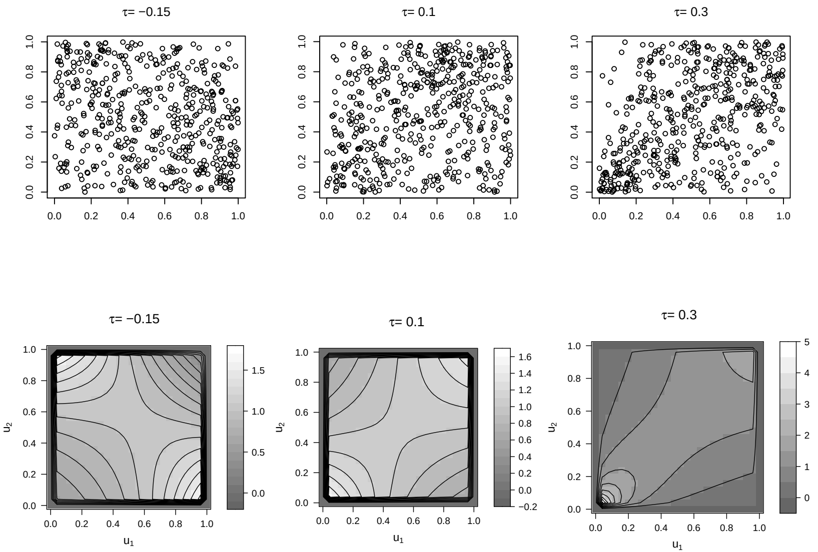
Kendall's Tau for this model is

$$\tau = 1 - \frac{\theta + (1-\theta)^2 \log(1-\theta)}{3\theta^2}; \qquad \tau \in [-0.1817, 0.3333]$$

```
#library(copula) #tau.amh ~= [-0.1817, 0.3333]
#AMH1 <- amhCopula(iTau(amhCopula() ,tau=-0.15))
#AMH2 <- amhCopula(iTau(amhCopula() ,tau=0.1))
#AMH3 <- amhCopula(iTau(amhCopula() ,tau=0.3))
#par(mfrow=c(1,3))
#plot(rCopula(500,copula=AMH1),xlab="",ylab="",main=expression(paste(tau,"= -0.15")))
#plot(rCopula(500,copula=AMH2),xlab="",ylab="",main=expression(paste(tau,"= 0.1")))
#plot(rCopula(500,copula=AMH3),xlab="",ylab="",main=expression(paste(tau,"= 0.3")))
#par(mfrow=c(1,1))
```

```
#p1=contourplot2(AMH1, dCopula,main=expression(paste(tau,"= -0.15")))
#p2=contourplot2(AMH2, dCopula,main=expression(paste(tau,"= 0.1")))
#p3=contourplot2(AMH3, dCopula,main=expression(paste(tau,"= 0.3")))
#library(gridExtra)
#grid.arrange(p1, p2,p3, nrow=1,ncol=3)
```

Archimedean copula: AMH copula



Archimedean copula: Gumbel-Hougaard copula

The bivariate Gumbel-Hougaard copula is given as

$$C(u_1, u_2) = \exp\left\{-\left((-\log u_1)^{\theta} + (-\log u_2)^{\theta}\right)^{1/\theta}\right\}; \quad (u_1, u_2) \in [0, 1]^2$$

where $\theta \in [1, \infty]$.

Properties

- Covers positive dependece
- Has only upper tail dependence
- Exchangeable
- Not Radially symmetric

Archimedean copula: Gumbel-Hougaard copula

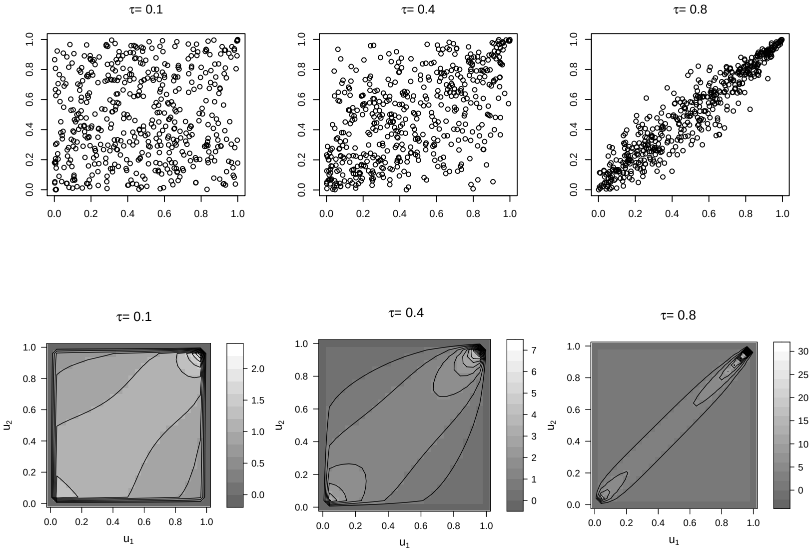
For this model:

$$\tau = \frac{\theta - 1}{\theta} \quad \text{and} \quad \lambda_l = 0, \lambda_u = 2 - 2^{1/\theta}.$$

```
#library(copula)
#GH1 <- gumbelCopula(iTau(gumbelCopula() ,tau=0.1))
#GH2 <- gumbelCopula(iTau(gumbelCopula() ,tau=0.4))
#GH3 <- gumbelCopula(iTau(gumbelCopula() ,tau=0.8))
#par(mfrow=c(1,3))
#plot(rCopula(500,copula=GH1),xlab="",ylab="",main=expression(paste(tau,"= 0.1")))
#plot(rCopula(500,copula=GH2),xlab="",ylab="",main=expression(paste(tau,"= 0.4")))
#plot(rCopula(500,copula=GH3),xlab="",ylab="",main=expression(paste(tau,"= 0.8")))
#par(mfrow=c(1,1))
```

```
#p1=contourplot2(GH1, dCopula,main=expression(paste(tau,"= 0.1")))
#p2=contourplot2(GH2, dCopula,main=expression(paste(tau,"= 0.4")))
#p3=contourplot2(GH3, dCopula,main=expression(paste(tau,"= 0.8")))
#library(gridExtra)
#grid.arrange(p1, p2,p3, nrow=1,ncol=3)
```

Archimedean copula: Gumbel-Hougaard copula



Archimedean copula: Joe copula

The bivariate Joe copula is derived when

$$\psi(t) = 1 - (1 - \exp(-t))^{1/\theta}; \quad \theta \in [1, \infty).$$

Properties

- Covers positive dependece
- Has upper tail dependence
- Exchangeable
- Not Radially symmetric

Archimedean copula: Joe copula

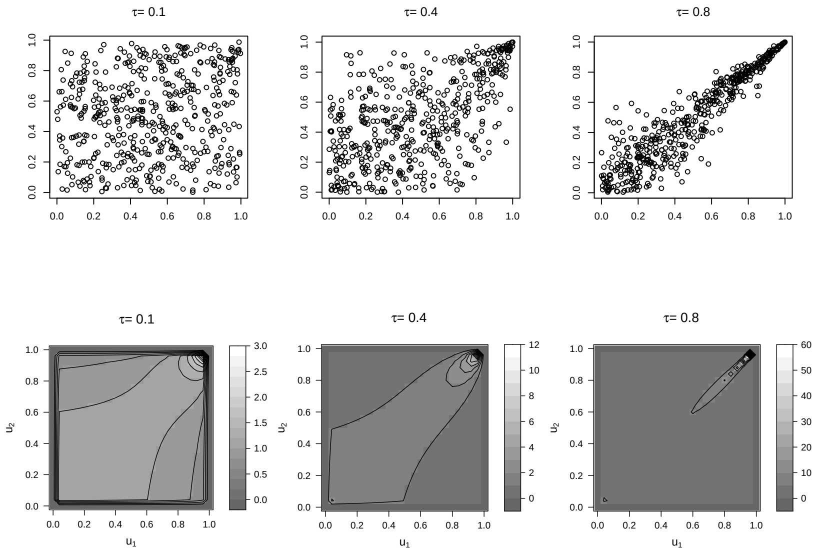
For this model

$$\lambda_l = 0 \quad \text{and} \quad \lambda_u = 2 - 2^{1/\theta}.$$

```
#(copula)
#JOE1 <- joeCopula(iTau(joeCopula() ,tau=0.1))
#JOE2 <- joeCopula(iTau(joeCopula() ,tau=0.4))
#JOE3 <- joeCopula(iTau(joeCopula() ,tau=0.8))
#par(mfrow=c(1,3))
#plot(rCopula(500,copula=JOE1),xlab="",ylab="",main=expression(paste(tau,"= 0.1")))
#plot(rCopula(500,copula=JOE2),xlab="",ylab="",main=expression(paste(tau,"= 0.4")))
#plot(rCopula(500,copula=JOE3),xlab="",ylab="",main=expression(paste(tau,"= 0.8")))
#par(mfrow=c(1,1))
```

```
#p1=contourplot2(JOE1, dCopula,main=expression(paste(tau,"= 0.1")))
#p2=contourplot2(JOE2, dCopula,main=expression(paste(tau,"= 0.4")))
#p3=contourplot2(JOE3, dCopula,main=expression(paste(tau,"= 0.8")))
#library(gridExtra)
#grid.arrange(p1, p2,p3, nrow=1,ncol=3)
```

Archimedean copula: Joe copula



Archimedean copula: BB1 copula

The BB1 copula is

$$C(u,v) = \left\{1 + [(u^{-\theta} - 1)^{\delta} + (v^{-\theta} - 1)^{\delta}]^{1/\delta}\right\}^{-1/\theta}; \qquad u, v \in (0,1),$$

where $\theta > 0$ and $\delta \geq 1$ (for more information see Czado 2019).

Properties

- Covers positive dependece
- Has lower & upper tail dependence
- Exchangeable
- Not Radially symmetric

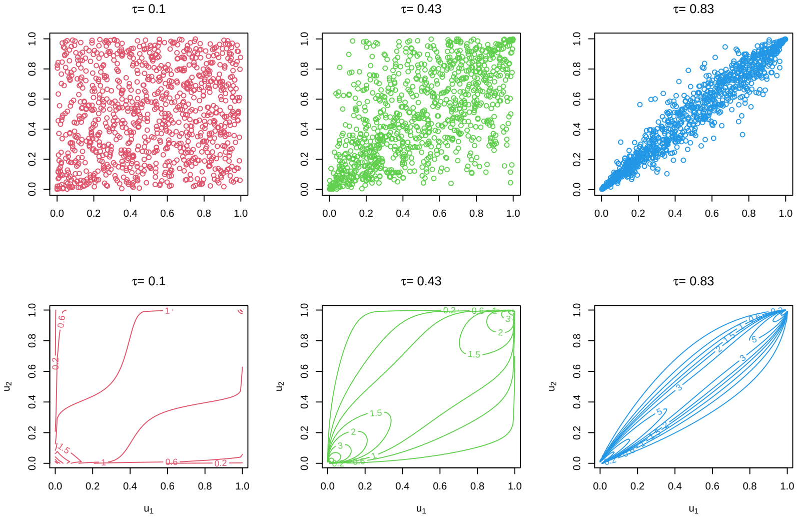
Archimedean copula: BB1 copula

For this model

$$\tau = 1 - \frac{2}{\delta(\theta + 2)}, \qquad \lambda_l = 2^{-1/(\theta\delta)} \qquad \text{and} \qquad \lambda_u = 2 - 2^{1/\delta}.$$

```
#library(VineCopula)
#bb1c<- BiCop(family = 7, par = 0.2, par2 = 1.01)
#bb2c<- BiCop(family = 7, par = 0.9, par2 = 1.21)
#bb3c<- BiCop(family = 7, par = 2, par2 = 3)
#par(mfrow=c(2,3))
#plot(BiCopSim(1000, bb1c),col=2,xlab="",ylab="",main=expression(paste(tau,"= 0.1")))
#plot(BiCopSim(1000, bb2c),col=3,xlab="",ylab="",main=expression(paste(tau,"= 0.43")))
#plot(BiCopSim(1000, bb3c),col=4,xlab="",ylab="",main=expression(paste(tau,"= 0.83")))
#contour(bb1c,margins="unif",col=2,main=expression(paste(tau,"= 0.1")))
#contour(bb2c,margins="unif",col=3,main=expression(paste(tau,"= 0.43")))
#contour(bb3c,margins="unif",col=4,main=expression(paste(tau,"= 0.83")))
#par(mfrow=c(1,1))
```

Archimedean copula: BB1 copula



Archimedean copula: BB7 copula

The BB7 copula is

$$C(u,v) = 1 - \left\{1 - \left[(1-(1-u)^{\theta})^{-\delta} + (1-(1-u)^{\theta})^{-\delta} - 1\right]^{-1/\delta}\right\}^{-1/\theta},$$

where $\theta \geq 1$ and $\delta > 0$ (for more information see Czado 2019 pp.47).

Properties

- Covers positive dependece
- Has lower & upper tail dependence
- Exchangeable
- Not Radially symmetric

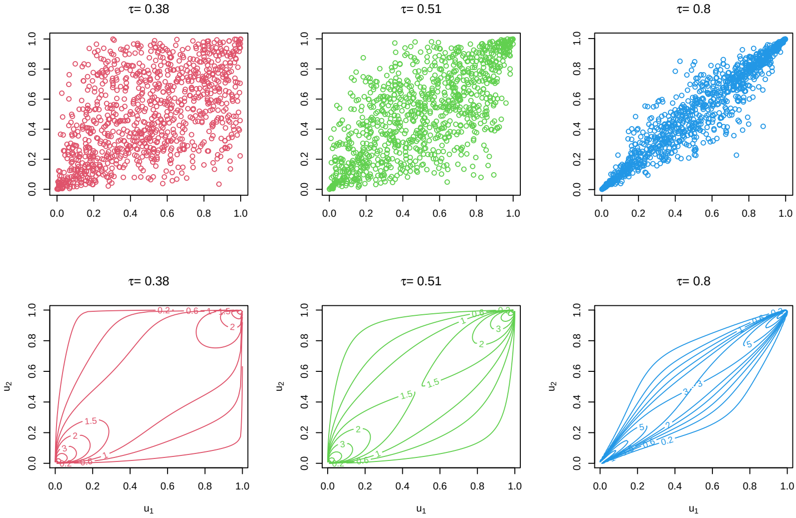
Archimedean copula: BB7 copula

For this model

$$\lambda_l = 2^{-1/\delta} \qquad \text{and} \qquad \lambda_u = 2 - 2^{1/\theta}.$$

```
#library(VineCopula)
#bb71c<- BiCop(family = 9, par = 1.2, par2 = 1.01)
#bb72c<- BiCop(family = 9, par = 1.9, par2 = 1.21)
#bb73c<- BiCop(family = 9, par = 5, par2 = 8)
#par(mfrow=c(2,3))
#plot(BiCopSim(1000, bb71c),col=2,xlab="",ylab="",main=expression(paste(tau,"= 0.38")))
#plot(BiCopSim(1000, bb72c),col=3,xlab="",ylab="",main=expression(paste(tau,"= 0.51")))
#plot(BiCopSim(1000, bb73c),col=4,xlab="",ylab="",main=expression(paste(tau,"= 0.8")))
#contour(bb71c,margins="unif",col=2,main=expression(paste(tau,"= 0.38")))
#contour(bb72c,margins="unif",col=3,main=expression(paste(tau,"= 0.51")))
#contour(bb73c,margins="unif",col=4,main=expression(paste(tau,"= 0.8")))
#par(mfrow=c(1,1))
```

Archimedean copula: BB7 copula



Extreme value copula

A copula C is called max stable if for every $m \geq 1$, it satisfies

$$C(u_1, u_2) = \left[C(u_1^{1/m}, u_2^{1/m})\right]^m.$$

A bivariate copula C is an extreme-value copula if and only if it is a max stable copula.

Pickands representation

A bivariate copula C is an extreme-value copula if and only if

$$C(u_1, u_2) = \exp\Big\{\big(\log u_1 + \log u_2\big)A\big(\frac{\log u_2}{\log u_1 + \log u_2}\big)\Big\},$$

where the function $A : [0, 1] \rightarrow [\frac{1}{2}, 1]$ is convex and satisfies $\max\{1 - t, t\} \leq A(t) \leq 1$ for all $t \in [0, 1]$. The function A is called the pickands dependence function.

Extreme value copula

```
#Extreme-Value generator
#A: [0,1] -> [1/2, 1] is convex and satisfies
#max(t,1-t) <= A(t) <= 1 for all t in [0,1].
#just for ev-copulas in "copula" package!
#"gumbelCopula", "galambosCopula", "huslerReissCopula",
#"tawnCopula", or "tevCopula".
#$A(galambosCopula(2),0.2)
#dAdu(galambosCopula(2),seq(0,1,0.1))$der1 #first derivati
#dAdu(galambosCopula(2),seq(0,1,0.1))$der2 #second derivat
```

Extreme value copula: sub-families

Note

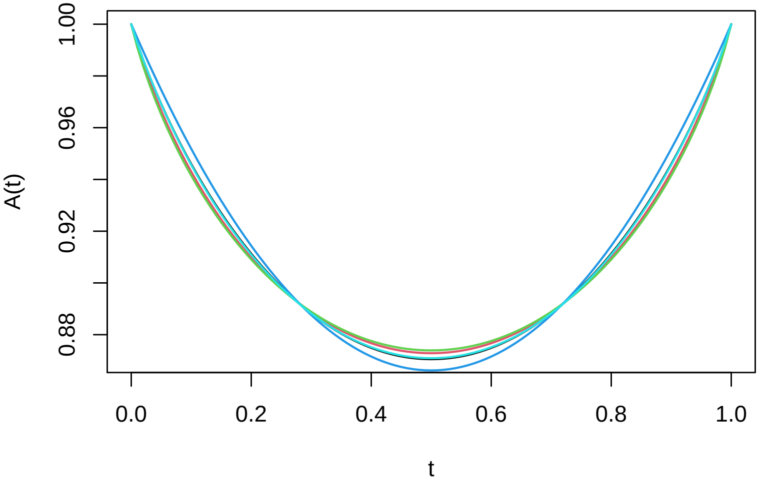
Extreme-value copulas can only model positive dependence.

- T-ev copula
- Tawn copula
- Galambos copula
- Gumbel copula (Also Archimedean copula)
- Hüsler-Reiss copula
- Marshall-Olkin copula
- BB8 copula

Extreme value copula

```
#curve(A(gumbelCopula(iTau(gumbelCopula(),tau=0.2)   ), x), 0, 1,lwd=3)
#curve(A(galambosCopula(iTau(galambosCopula(),tau=0.2)   ), x), add=TRUE, col=2,lwd=3)
#curve(A(huslerReissCopula(iTau(huslerReissCopula(),tau=0.2)   ), x), add=TRUE, col=3,lwd=3)
#curve(A(tawnCopula(iTau(tawnCopula(),tau=0.2)   ), x), add=TRUE, col=4,lwd=3)
#curve(A(tevCopula(iTau(tevCopula(),tau=0.2)   ), x), add=TRUE, col=5,lwd=3)
```

Extreme value copula



Extreme value copula: T-ev copula

The T-Extreme-value copula is
$C(u_1, u_2) = \exp\big(-l(-\log u_1, -\log u_2)\big)$, where

$$l(x_1, x_2) = x_1 t_{\nu+1}\Big(\frac{(x_1/x_2)^{1/\nu} - \rho}{\sqrt{1-\rho^2}}\sqrt{\nu+1} + x_2 t_{\nu+1}\Big(\frac{(x_2/x_1)^{1/\nu} - \rho}{\sqrt{1-\rho^2}}\sqrt{\nu+1}\Big),$$

where $\nu > 0$ and $-1 < \rho < 1$.

Properties

- Covers positive dependece
- Has upper tail dependence
- Exchangeable
- Not Radially symmetric

Extreme value copula: T-ev copula

For this model

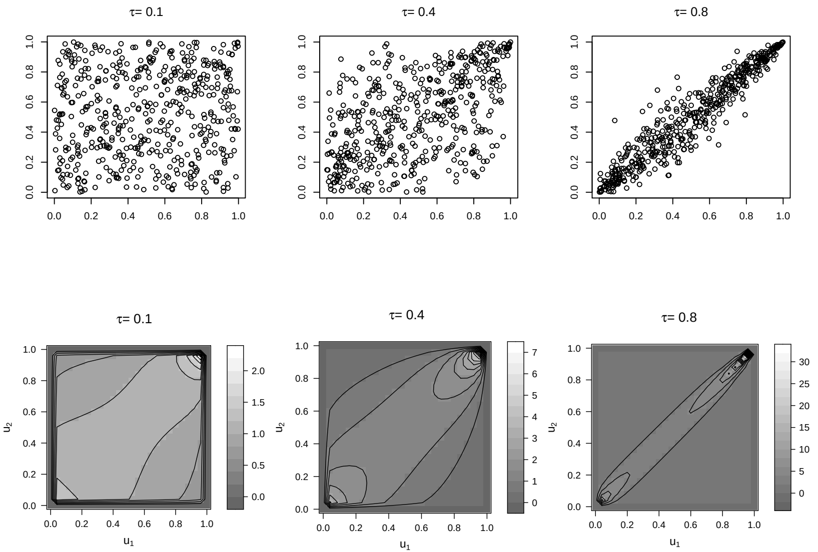
$$\lambda_l = 0 \quad \text{and} \quad \lambda_U = 2[1 - T_{\nu+1}(z_1/2)].$$

See Czado (2019) pp. 59 for more information.

```
#library(copula)
#tev1 <- tevCopula(iTau(tevCopula() ,tau=0.1))
#tev2 <- tevCopula(iTau(tevCopula() ,tau=0.4))
#tev3 <- tevCopula(iTau(tevCopula() ,tau=0.8))
#par(mfrow=c(1,3))
#plot(rCopula(500,copula=tev1),xlab="",ylab="",main=expression(paste(tau,"= 0.1")))
#plot(rCopula(500,copula=tev2),xlab="",ylab="",main=expression(paste(tau,"= 0.4")))
#plot(rCopula(500,copula=tev3),xlab="",ylab="",main=expression(paste(tau,"= 0.8")))
#par(mfrow=c(1,1))
```

```
#p1=contourplot2(tev1, dCopula,main=expression(paste(tau,"= 0.1")))
#p2=contourplot2(tev2, dCopula,main=expression(paste(tau,"= 0.4")))
#p3=contourplot2(tev3, dCopula,main=expression(paste(tau,"= 0.8")))
#library(gridExtra)
#grid.arrange(p1, p2,p3, nrow=1,ncol=3)
```

Extreme value copula: T-ev copula



Extreme value copula: Tawn copula

The Tawn copula is

$$C(u, v) = uv \exp\{-\theta \frac{\log u \log v}{\log uv}\}, \qquad \theta \in (0, 1).$$

Properties

- Covers positive dependece
- Has upper tail dependence
- Exchangeable
- Not Radially symmetric

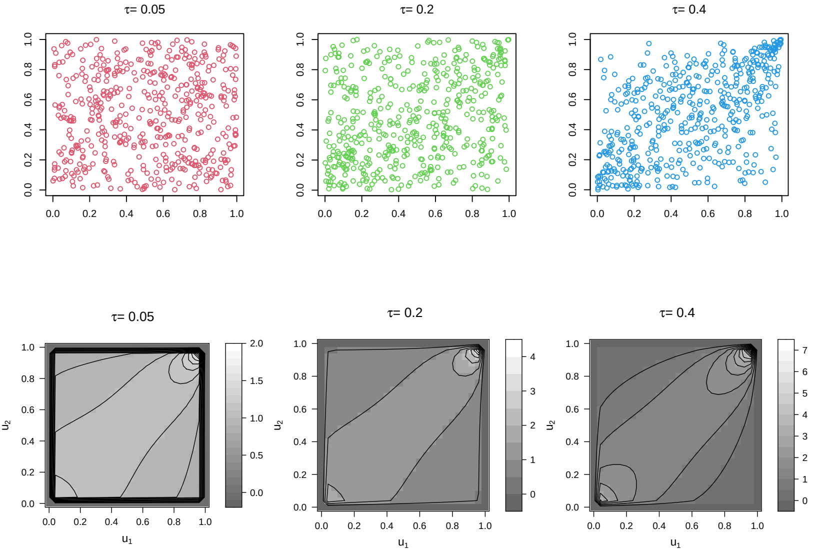
Extreme value copula: Tawn copula

For this model $\tau \in [0, 0.4183992]$.

```
#library(copula)
#tawn1 <- tawnCopula(iTau(tawnCopula() ,tau=0.05))
#tawn2 <- tawnCopula(iTau(tawnCopula() ,tau=0.2))
#tawn3 <- tawnCopula(iTau(tawnCopula() ,tau=0.4))
#par(mfrow=c(1,3))
#plot(rCopula(500,copula=tawn1),col=2,xlab="",ylab="",main=expression(paste(tau,"= 0.05")))
#plot(rCopula(500,copula=tawn2),col=3,xlab="",ylab="",main=expression(paste(tau,"= 0.2")))
#plot(rCopula(500,copula=tawn3),col=4,xlab="",ylab="",main=expression(paste(tau,"= 0.4")))
#par(mfrow=c(1,1))
```

```
#p1=contourplot2(tawn1, dCopula,main=expression(paste(tau,"= 0.05")))
#p2=contourplot2(tawn2, dCopula,main=expression(paste(tau,"= 0.2")))
#p3=contourplot2(tawn3, dCopula,main=expression(paste(tau,"= 0.4")))
#library(gridExtra)
#grid.arrange(p1, p2,p3, nrow=1,ncol=3)
```

Extreme value copula: Tawn copula



Extreme value copula: Galambos copula

The Galambos copula

$$C(u_1, u_2) = u_1 u_2 \exp\left\{ -\left((-\log u_1)^{\theta} + (-\log u_2)^{\theta}\right)^{1/\theta}\right\},$$

where $\theta \in (0, \infty)$.

Properties

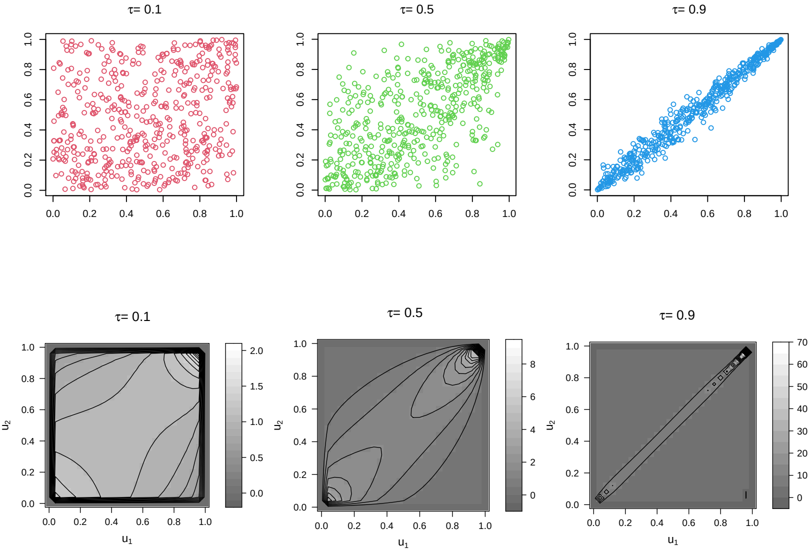
- Covers positive dependece
- Has upper tail dependence
- Exchangeable
- Radially symmetric

Extreme value copula: Galambos copula

```
#library(copula)
#gal1 <- galambosCopula(iTau(galambosCopula() ,tau=0.1))
#gal2 <- galambosCopula(iTau(galambosCopula() ,tau=0.5))
#gal3 <- galambosCopula(iTau(galambosCopula() ,tau=0.9))
#par(mfrow=c(1,3))
#plot(rCopula(500,copula=gal1),col=2,xlab="",ylab="",main=expression(paste(tau,"= 0.1")))
#plot(rCopula(500,copula=gal2),col=3,xlab="",ylab="",main=expression(paste(tau,"= 0.5")))
#plot(rCopula(500,copula=gal3),col=4,xlab="",ylab="",main=expression(paste(tau,"= 0.9")))
#par(mfrow=c(1,1))
```

```
#p1=contourplot2(gal1, dCopula,main=expression(paste(tau,"= 0.1")))
#p2=contourplot2(gal2, dCopula,main=expression(paste(tau,"= 0.5")))
#p3=contourplot2(gal3, dCopula,main=expression(paste(tau,"= 0.9")))
#library(gridExtra)
#grid.arrange(p1, p2,p3, nrow=1,ncol=3)
```

Extreme value copula: Galambos copula



Extreme value copula: Huesler-Reiss copula

The pickands function for Huesler-Reiss copula is

$$A(t) = (1-t)\Phi(z_{1-t}) + t\Phi(z_t),$$

where $z_t = (\frac{1}{\lambda} + \frac{\lambda}{2}\ln\frac{t}{1-t})$ and $\lambda \geq 0$.

Properties

- Covers positive dependece
- Has upper tail dependence
- Exchangeable
- Not Radially symmetric

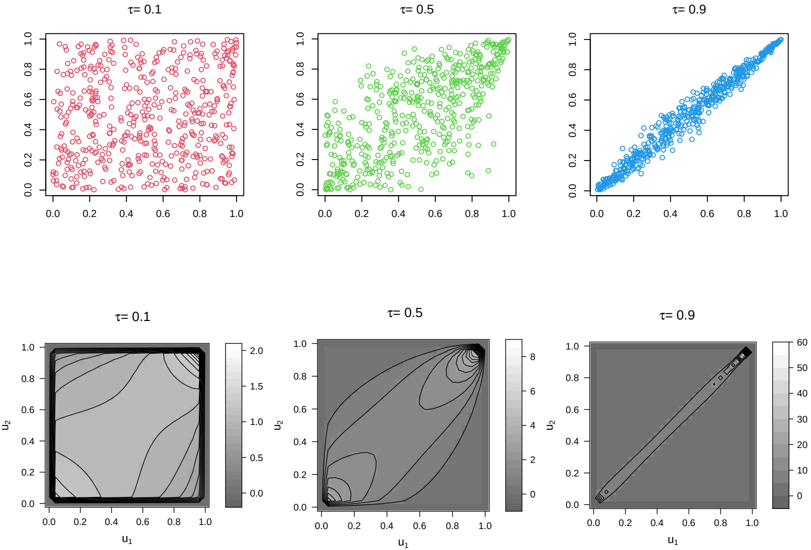
Extreme value copula: Huesler-Reiss copula

The upper tail dependence for this model is $\lambda_U = 2[1 - \Phi(\frac{1}{\lambda})]$.

```
#library(copula)
#hus1 <- huslerReissCopula(iTau(huslerReissCopula() ,tau=0.1))
#hus2 <- huslerReissCopula(iTau(huslerReissCopula() ,tau=0.5))
#hus3 <- huslerReissCopula(iTau(huslerReissCopula() ,tau=0.9))
#par(mfrow=c(1,3))
#plot(rCopula(500,copula=hus1),col=2,xlab="",ylab="",main=expression(paste(tau,"= 0.1")))
#plot(rCopula(500,copula=hus2),col=3,xlab="",ylab="",main=expression(paste(tau,"= 0.5")))
#plot(rCopula(500,copula=hus3),col=4,xlab="",ylab="",main=expression(paste(tau,"= 0.9")))
#par(mfrow=c(1,1))
```

```
#p1=contourplot2(hus1, dCopula,main=expression(paste(tau,"= 0.1")))
#p2=contourplot2(hus2, dCopula,main=expression(paste(tau,"= 0.5")))
#p3=contourplot2(hus3, dCopula,main=expression(paste(tau,"= 0.9")))
#library(gridExtra)
#grid.arrange(p1, p2,p3, nrow=1,ncol=3)
```

Extreme value copula: Huesler-Reiss copula



Extreme value copula: Marshall-Olkin copula

The Marshall-Olkin copula is

$$C(u, v) = \min\{uv^{1-\alpha}, u^{1-\beta}v\}, \quad \alpha, \beta > 0.$$

Properties

- Covers positive dependece
- Has upper tail dependence
- Non-exchangeable (when $\alpha = \beta$ exchangable)
- Not Radially symmetric

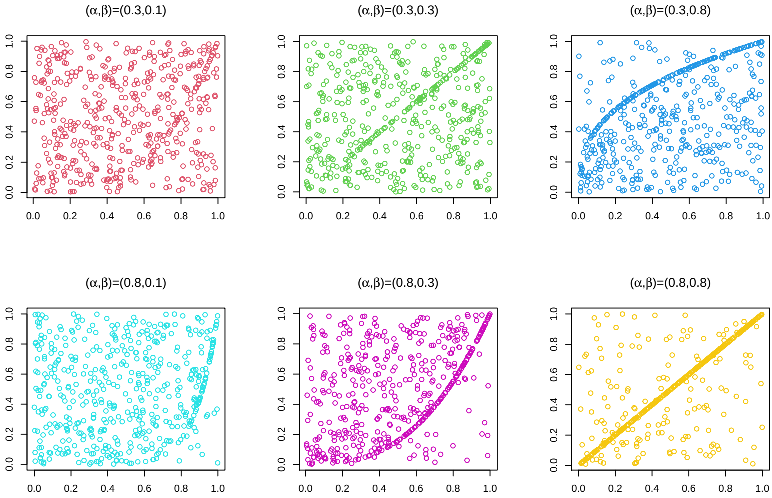
Extreme value copula: Marshall-Olkin copula

The upper tail dependence is

$$\lambda_U = \min\{\alpha, \beta\}.$$

```
#library(copula)
#mo1 <- moCopula(c(0.3,0.1))
#mo2 <- moCopula(c(0.3,0.3))
#mo3 <- moCopula(c(0.3,0.8))
#mo4 <- moCopula(c(0.8,0.1))
#mo5 <- moCopula(c(0.8,0.3))
#mo6 <- moCopula(c(0.8,0.8))
#par(mfrow=c(2,3))
#plot(rCopula(500,copula=mo1),col=2,xlab="",ylab="",main=expression(paste("(",alpha,",",beta,")=(0.3,0.1)")
#plot(rCopula(500,copula=mo2),col=3,xlab="",ylab="",main=expression(paste("(",alpha,",",beta,")=(0.3,0.3)")
#plot(rCopula(500,copula=mo3),col=4,xlab="",ylab="",main=expression(paste("(",alpha,",",beta,")=(0.3,0.8)")
#plot(rCopula(500,copula=mo4),col=5,xlab="",ylab="",main=expression(paste("(",alpha,",",beta,")=(0.8,0.1)")
#plot(rCopula(500,copula=mo5),col=6,xlab="",ylab="",main=expression(paste("(",alpha,",",beta,")=(0.8,0.3)")
#plot(rCopula(500,copula=mo6),col=7,xlab="",ylab="",main=expression(paste("(",alpha,",",beta,")=(0.8,0.8)")
#par(mfrow=c(1,1))
```

Extreme value copula: Marshall-Olkin copula



Extreme value copula: BB8 copula

The BB8 copula is

$$C(u,v) = \frac{1}{\delta}\Big(1 - \{1 - \frac{1}{\eta}[1-(1-\delta u)^{\nu}][1-(1-\delta v)^{\nu}]\}^{1/\nu}\Big),$$

where $\nu \geq 1$ and $0 < \delta \leq 1$.

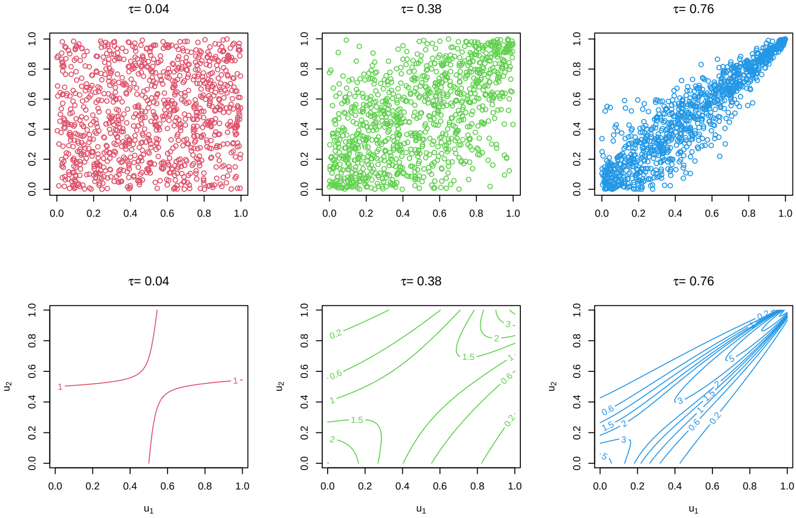
Properties

- Covers positive dependece
- No tail dependence
- Exchangeable
- Not Radially symmetric

Extreme value copula: BB8 copula

```
#library(VineCopula)
#bb81c<- BiCop(family = 10, par = 2, par2 = 0.3)
#bb82c<- BiCop(family = 10, par = 5, par2 = 0.6)
#bb83c<- BiCop(family = 10, par = 8, par2 = 0.95)
#par(mfrow=c(2,3))
#plot(BiCopSim(1000, bb81c),col=2,xlab="",ylab="",main=expression(paste(tau,"= 0.04")))
#plot(BiCopSim(1000, bb82c),col=3,xlab="",ylab="",main=expression(paste(tau,"= 0.38")))
#plot(BiCopSim(1000, bb83c),col=4,xlab="",ylab="",main=expression(paste(tau,"= 0.76")))
#contour(bb81c,margins="unif",col=2,main=expression(paste(tau,"= 0.04")))
#contour(bb82c,margins="unif",col=3,main=expression(paste(tau,"= 0.38")))
#contour(bb83c,margins="unif",col=4,main=expression(paste(tau,"= 0.76")))
#par(mfrow=c(1,1))
```

Extreme value copula: BB8 copula



The FGM copula

The Farlie-Gumbel-Morgenstern (FGM) copula is

$$C(u, v) = uv\big(1 + \theta(1 - u)(1 - v)\big),$$

where $\theta \in (-1, 1)$.

Properties

- Covers positive & negative dependece
- No tail dependence
- Exchangeable
- Radially symmetric

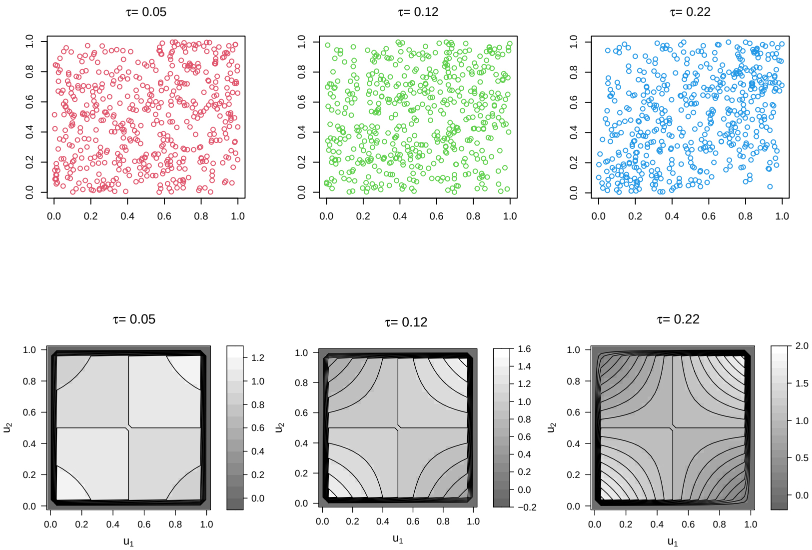
The FGM copula

For this model $\tau \in (-2/9, 2/9) \approx (-0.23, 0.23)$.

```
#library(copula)
#fgm1 <- fgmCopula(iTau(fgmCopula() ,tau=0.05))
#fgm2 <- fgmCopula(iTau(fgmCopula() ,tau=0.12))
#fgm3 <- fgmCopula(iTau(fgmCopula() ,tau=0.22))
#par(mfrow=c(1,3))
#plot(rCopula(500,copula=fgm1),col=2,xlab="",ylab="",main=expression(paste(tau,"= 0.05")))
#plot(rCopula(500,copula=fgm2),col=3,xlab="",ylab="",main=expression(paste(tau,"= 0.12")))
#plot(rCopula(500,copula=fgm3),col=4,xlab="",ylab="",main=expression(paste(tau,"= 0.22")))
#par(mfrow=c(1,1))
```

```
#p1=contourplot2(fgm1, dCopula,main=expression(paste(tau,"= 0.05")))
#p2=contourplot2(fgm2, dCopula,main=expression(paste(tau,"= 0.12")))
#p3=contourplot2(fgm3, dCopula,main=expression(paste(tau,"= 0.22")))
#library(gridExtra)
#grid.arrange(p1, p2,p3, nrow=1,ncol=3)
```

The FGM copula



Frechet-Hoeffding copulas

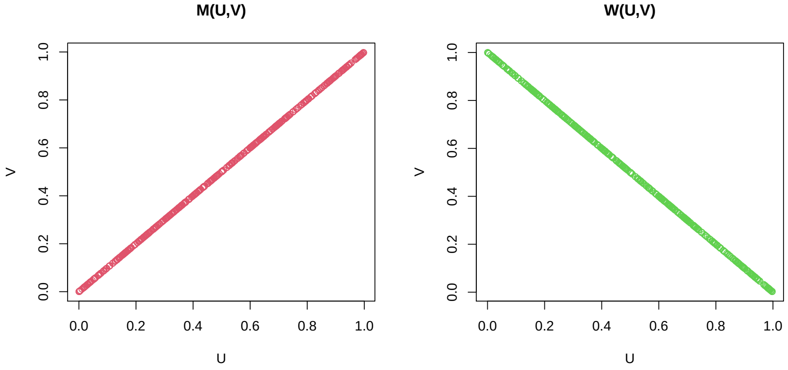
For any copula C, we know that

$$W(u,v) \leq C(u,v) \leq M(u,v),$$

where $W(u,v) = \max\{u+v-1, 0\}$ and $M(u,v) = \min(u,v)$. The copulas M and W are known as Frechet-Hoeffding copulas.

Frechet-Hoeffding copulas

```r
U <- runif(500)
par(mfrow=c(1,2))
plot(cbind(U,U),main="M(U,V)",xlab="U",ylab="V",col=2)
plot(cbind(U,1-U),main="W(U,V)",xlab="U",ylab="V",col=3)
```




```r
par(mfrow=c(1,1))
```

Khoudraji copula

The Khoudraji copula constructed from C_1 and C_2 is

$$C(u, v) = C_1(u^{1-a}, v^{1-b})C_2(u^a, v^b)$$

for some $a, b \in (0, 1)$.

Using this copula, you can construct new models that have different characteristic (like dependence structure, tail dependence, exchangeability, radial symmetry).

Khoudraji copula: An example

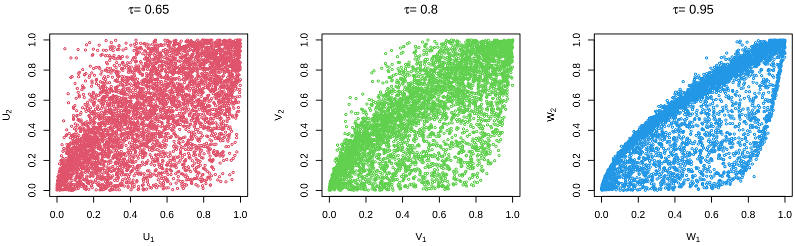
In this example we want to construct a Khoudraji copula from Gumbel and Clayton copulas with both $\tau = 0.65$.

```
th1 <- iTau(gumbelCopula(),0.65)
th2 <- iTau(claytonCopula(),0.65)
kho <- khoudrajiCopula(copula1=gumbelCopula(th1),
                       copula2=claytonCopula(th2),
                       shapes=c(0.6,0.95))
kho
```

```
## Khoudraji copula, dim. d = 2, constructed from
##  Gumbel copula
##  Clayton copula
## Dimension:  2
## Parameters:
##   c1.alpha   = 2.857143
##   c2.alpha   = 3.714286
##   shape1     = 0.600000
##   shape2     = 0.950000
```

Khoudraji copula: An example

```
n=5000
library(copula)
param<-function(tau) c(iTau(copula1(),tau),iTau(copula2(),tau))
s<-c(0.6,0.95)
copula1<-gumbelCopula
copula2<-claytonCopula
U <- rCopula(n, copula=kho)
par(mfrow=c(1,3))
plot(U, cex=0.5, xlab=quote(U[1]), ylab=quote(U[2])
     ,main=expression(paste(tau,"= 0.65")),col=2)
V <- rCopula(n, copula= setTheta(kho, value=c(param(0.8),s)))
plot(V, cex=0.5, xlab=quote(V[1]), ylab=quote(V[2])
     ,main=expression(paste(tau,"= 0.8")),col=3)
W <- rCopula(n, copula= setTheta(kho, value=c(param(0.95),s)))
plot(W, cex=0.5, xlab=quote(W[1]), ylab=quote(W[2])
     ,main=expression(paste(tau,"= 0.95")),col=4)
```



Some remarks

- Rotation of copulas
- Mixture of copulas

Some remarks: Rotation of copulas

Let $(U, V) \sim C$. Then

- 90 degree rotation of the copula C is

$$C_{1-U,V}(u, v) = v - C_{U,V}(1-u, v)$$

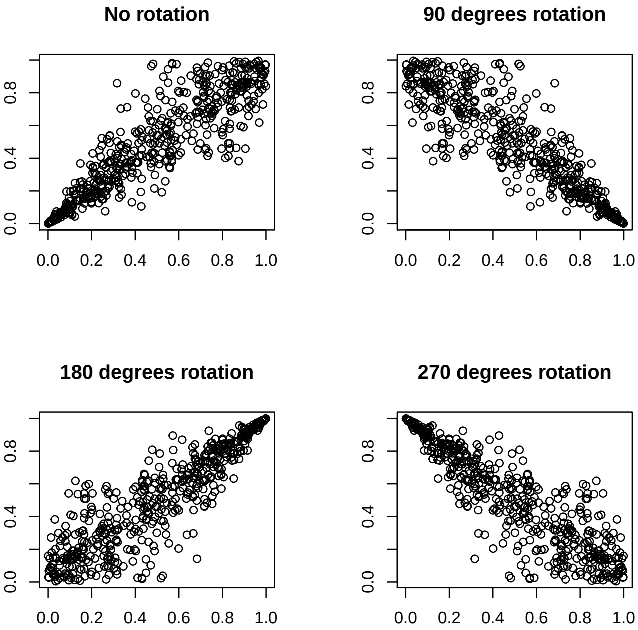
- 180 degree rotation of the copula C is

$$C_{1-U,1-V}(u, v) = u + v - 1 + C_{U,V}(1-u, 1-v)$$

- 270 degree rotation of the copula C is

$$C_{U,1-V}(u, v) = u - C_{U,V}(u, 1-v)$$

Some remarks: Rotation of clayton copula



Some remarks: Rotation of clayton copula

```
#cl3 <- claytonCopula(iTau(claytonCopula() ,tau=0.7))
#par(mfrow=c(2,2))
#U <- rCopula(500,copula=cl3)
#plot(U,xlab="",ylab="",main="No rotation")
#plot(1-U[,1],U[,2],xlab="",ylab="",main="90 degrees rotation")
#plot(1-U[,1],1-U[,2],xlab="",ylab="",main="180 degrees rotation")
#plot(U[,1],1-U[,2],xlab="",ylab="",main="270 degrees rotation")
#par(mfrow=c(2,2))
```

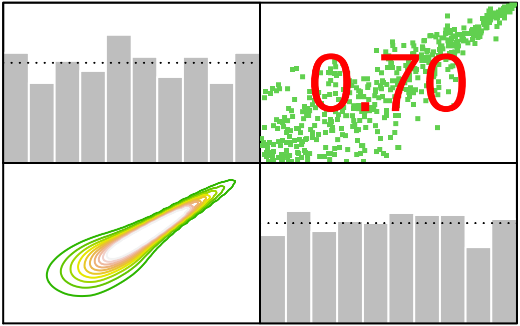
Some remarks: Rotation of copulas

If you have rotated data, you can proceed with two strategies:

1. Rotate the data using a transformation
2. Use a rotated copula model

Some remarks: Rotation of copulas, example

Let's suppose the data are illustrated as following:



Some remarks: Rotation of copulas, example

At first glance, this figure shows that the data have positive dependence structure, has upper tail dependence and is exchangeable. You might choose the joe or Gumbel copulas. However, you can also

- rotate the data 180 degrees and choose a clayton copula
- Use a 180 degrees rotated clayotn copula model

```
# First load fake data
#cl3 <- claytonCopula(iTau(claytonCopula() ,tau=0.7))
#U <- rCopula(500,copula=cl3)
#V<-cbind(1-U[,1],1-U[,2])
#
#Rotate the data 180 degrees
#V.rotated <- cbind(1-V[,1],1-V[,2])
#
#use a rotated clayoton copula
#rotated.cop <- rotCopula(claytonCopula(4),flip=c(TRUE,TRUE))
```

Some remarks: Mixture of copulas

The mixture of $C_1, \ldots, C_m$ with mixing vector weights $w = (w_1, \ldots, w_m)$ is

$$mix_w(C_1, \ldots, C_m) = \sum_{k=1}^{m} w_k C_k.$$

A nice property would be

$$\lambda_l = \sum_{k=1}^{m} \lambda_l^{(k)}$$

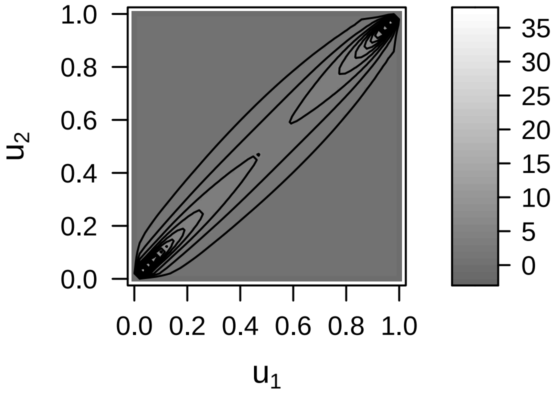
$$\lambda_u = \sum_{k=1}^{m} \lambda_u^{(k)}$$

Some remarks: Mixture of copulas, example

If we want to construct a model with lower and upper tail dependence, we can use the mixture of Gumbel and Clayton copulas.

```
#cc <- claytonCopula(iTau(claytonCopula(),tau= 0.75))
#gc <- gumbelCopula(iTau(gumbelCopula(),tau= 0.75))
#weights <- c(1/3,2/3)
#mcg <- mixCopula(list(cc,gc),w=weights)
#U <- rCopula(1000, copula=mcg)
#contourplot2(mcg, FUN=dCopula, cuts=32, n.grid=50, pretty=FALSE)
```

Some remarks: Mixture of copulas, example



Any questions?

This session is finished!