M5 Copulas

General Insurance Modelling : Actuarial Modelling III [1]

Professor Benjamin Avanzi

03 May 2024 11:51

[1] References: Unit 3 of CS2

1. Dependence and multivariate modelling
   - Introduction to Dependence
   - Multivariate Normal Distributions
   - Measures of dependence
   - Limits of correlation

Motivation

How does dependence arise?

- Events affecting more than one variable (e.g., storm on building, car and business interruption covers)
  - $\rightarrow$ what is the impact of the recent El Niña floods in QLD and NSW?
- Underlying economic factors affecting more than one risk area (e.g., inflation, unemployment)
  - $\rightarrow$ what is the impact of the RBA cash rate on property insurance, workers compensation, claims inflation, income protection, mortgage insurance?
- Clustering/concentration of risks (e.g., household, geographical area)

Reasons for modelling dependence:

- Pricing:
  - get adequate risk loadings
    (note that dependence affects quantiles, not the mean)
- Solvency assessment:
  - bottom up: for given risks, get capital requirements
    (get quantiles of aggregate quantities)
- Capital allocation:
  - top down: for given capital, allocate portions to risk
    (for profitability assessment)
- Portfolio structure: (or strategic asset allocation)
  - how do assets and liabilities move together?

Examples

- World Trade Centre (9/11) causing losses to Property, Life, Workers' Compensation, Aviation insurers
- Dot.com market collapse and GFC causing losses to the stock market and to insurers of financial institutions and Directors & Officers (D&O) writers
- Asbestos affecting many past liability years at once
- Australian 2019-2020 Bushfires causing losses to Property, Life, credit, etc ...
- Covid-19 impacting financial markets, travel insurance, health, credit, D&O, business interruption covers, construction price inflation, etc ...
- El Niña and associated floods impacting property insurance in certain geographical areas, construction prices inflation, etc ...

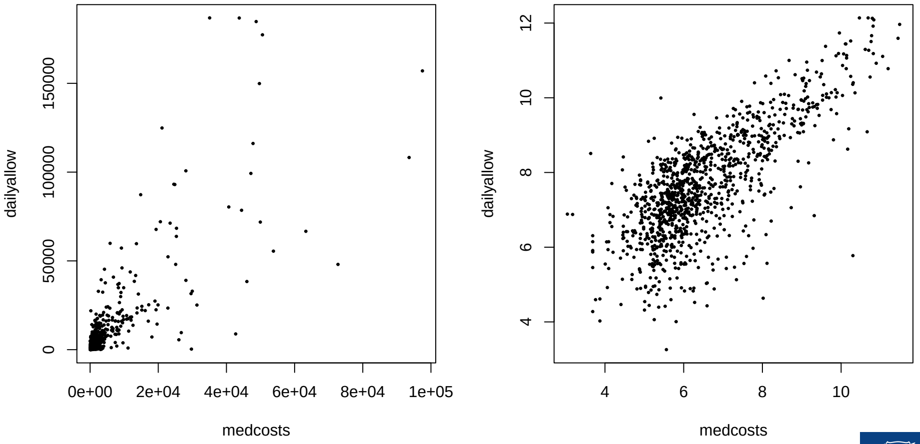
Example of real actuarial data (Avanzi, Cassar, and Wong 2011)

- Data were provided by the SUVA (Swiss workers compensation insurer)
- Random sample of 5% of accident claims in construction sector with accident year 1999 (developped as of 2003)
- 1089 of those are common (!)
- Two types of claims: 2249 medical cost claims, et 1099 daily allowance claims

```
SUVA <- read_excel("SUVA.xls")
# filtering and logging the common claims
SUVAcom <- log(SUVA[SUVA$medcosts > 0 & SUVA$dailyallow > 0,
  ])
```

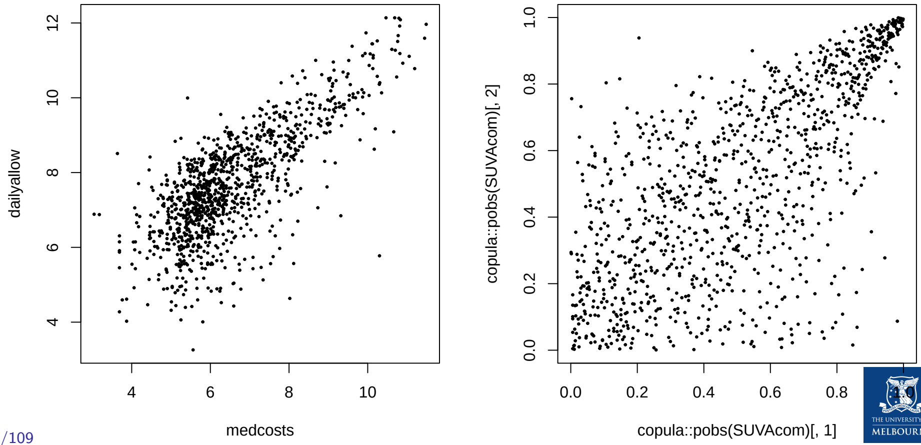
Scatterplot of those 1089 common claims (LHS) amd their log (RHS):

```
par(mfrow = c(1, 2), pty = "s")
plot(exp(SUVAcom), pch = 20, cex = 0.5)
plot(SUVAcom, pch = 20, cex = 0.5)
```



Scatterplot of the log of the 1089 common claims (LHS) and their empirical copula (ranks) (RHS):

```
par(mfrow = c(1, 2), pty = "s")
plot(SUVAcom, pch = 20, cex = 0.5)
plot(copula::pobs(SUVAcom)[, 1], copula::pobs(SUVAcom)[, 2],
  pch = 20, cex = 0.5)
```



1. Dependence and multivariate modelling
   - Introduction to Dependence
   - Multivariate Normal Distributions
   - Measures of dependence
   - Limits of correlation

The multivariate Normal distribution

$\mathbf{Z} = (Z_1, ...Z_n)' \sim MN(\mathbf{0}, \Sigma)$ if its joint p.d.f. is

$$f(z_1, ..., z_n) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} \exp\left\{-\frac{1}{2}\mathbf{z}'\Sigma^{-1}\mathbf{z}\right\}.$$

- standard Normal marginals, i.e. $Z_i \sim N(0, 1)$
- positive definite correlation matrix:

$$\Sigma = \begin{pmatrix} 1 & \rho_{12} & \cdots & \rho_{1n} \\ \rho_{21} & 1 & \cdots & \rho_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{n1} & \rho_{n2} & \cdots & 1 \end{pmatrix}$$

where $\rho_{ij} = \rho_{ji}$ is the correlation between Z_i and Z_j.

- if $\rho_{ij} = 0$ for all $i \neq j$, then we have the standard MN.

Properties

If $\mathbf{Z} \sim MN(\mathbf{0}, \Sigma)$, then with appropriate dimensions $\mathbf{A}$ and $\mathbf{C}$, the vector

$$\mathbf{X} = \mathbf{AZ} + \mathbf{C}$$

has a multivariate Normal distribution with mean

$$\mathrm{E}(\mathbf{X}) = \mathbf{C}$$

and covariance

$$\mathrm{Cov}(\mathbf{X}) = \mathbf{A}\Sigma\mathbf{A}'.$$

✠ Cholesky's decomposition

We can construct a lower triangular matrix

$$B = \begin{pmatrix} b_{11} & 0 & \cdots & 0 \\ b_{21} & b_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{nn} \end{pmatrix}$$

such that $\Sigma = \mathbf{B}\mathbf{B}^\top$.

The matrix $\mathbf{B}$ can be determined using Cholesky's decomposition algorithm (standard function in most software—see `chol()` in R).

This will be useful later on for the simulation of multivariate Gaussian random variables.

1. Dependence and multivariate modelling
   - Introduction to Dependence
   - Multivariate Normal Distributions
   - Measures of dependence
   - Limits of correlation

Pearson's correlation measure

Pearson's correlation coefficient is defined by

$$\rho\left(Z_i, Z_j\right) = \rho_{ij} = \frac{\text{Cov}\left(Z_i, Z_j\right)}{\sqrt{\text{Var}\left(Z_i\right)\text{Var}\left(Z_j\right)}}.$$

Note:

- This measures the degree of linear relationship between Z_i and Z_j.
- In general, it does not reveal all the information on the dependence structure of random couples.

Kendall's tau

Kendall's tau correlation coefficient is defined by

$$\begin{aligned}\tau\left(Z_i, Z_j\right) &= \tau_{ij} \\ &= P\left[\left(Z_i - Z_i'\right)\left(Z_j - Z_j'\right) > 0\right] - P\left[\left(Z_i - Z_i'\right)\left(Z_j - Z_j'\right) < 0\right]\end{aligned}$$

where (Z_i, Z_j) and $\left(Z_i', Z_j'\right)$ are two independent realisations.

Note:

- The first term is called the probability of concordance; the latter, probability of discordance.
- Its value is also between -1 and 1.
- It can be shown to equal: $\tau\left(Z_i, Z_j\right) = 4E\left[F\left(Z_i, Z_j\right)\right] - 1$.
- Concordance and discordance only depends on ranks, and this indicator is hence less affected by the marginal distributions of Z_i and Z_j than Pearson's correlation.

(Spearman's) rank correlation

Spearman's rank correlation coefficient is defined by

$$r\left(Z_i, Z_j\right) = r_{ij} = \rho\left(F_i\left(Z_i\right), F_j\left(Z_j\right)\right)$$

where F_i and F_j are the respective marginal distributions.

Note:

- It is indeed the Pearson's correlation but applied to the transformed variables $F_i\left(Z_i\right)$ and $F_j\left(Z_j\right)$.
- Its value is also between -1 and 1.
- It is directly formulated on ranks, and hence is less affected by the marginal distributions of Z_i and Z_j than Pearson's correlation.

Example: the case of multivariate Normal

- Pearson's correlation:

$$\rho_{ij}$$

- Kendall's tau:

$$\tau_{ij} = \frac{2}{\pi} \arcsin\left(\rho_{ij}\right)$$

- Spearman's rank correlation:

$$r_{ij} = \frac{6}{\pi} \arcsin\left(\frac{\rho_{ij}}{2}\right)$$

Example: SUVA data

```
cor(SUVAcom, method = "pearson")  # default
```

```
##            medcosts dailyallow
## medcosts   1.0000000  0.7489701
## dailyallow 0.7489701  1.0000000
```

```
cor(SUVAcom, method = "kendall")
```

```
##            medcosts dailyallow
## medcosts   1.0000000  0.5154526
## dailyallow 0.5154526  1.0000000
```

```
cor(SUVAcom, method = "spearman")
```

```
##            medcosts dailyallow
## medcosts   1.0000000  0.6899156
## dailyallow 0.6899156  1.0000000
```

Repeating those on the original claims (before log transformation):

```
cor(exp(SUVAcom), method = "pearson")  # default
```

```
##             medcosts dailyallow
## medcosts   1.0000000  0.8015752
## dailyallow 0.8015752  1.0000000
```

```
cor(exp(SUVAcom), method = "kendall")
```

```
##             medcosts dailyallow
## medcosts   1.0000000  0.5154526
## dailyallow 0.5154526  1.0000000
```

```
cor(exp(SUVAcom), method = "spearman")
```

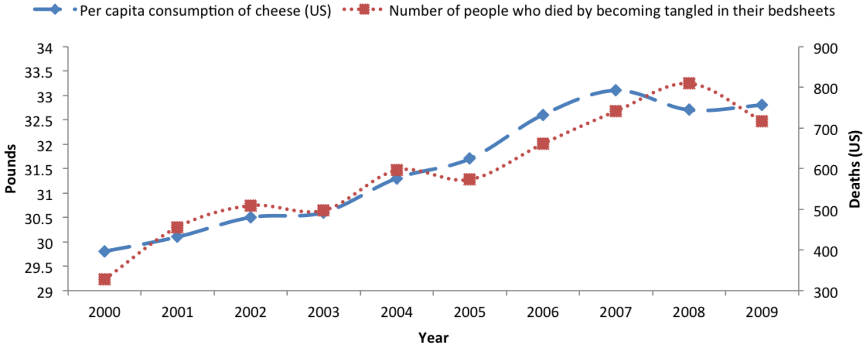
```
##             medcosts dailyallow
## medcosts   1.0000000  0.6899156
## dailyallow 0.6899156  1.0000000
```

We see that Kendall's τ and Spearman's r are unchanged. This is because the log transformation does not affect ranks in the data. The more extreme nature of the data, however, leads to a higher Pearson's correlation coefficient.

1. Dependence and multivariate modelling
   - Introduction to Dependence
   - Multivariate Normal Distributions
   - Measures of dependence
   - Limits of correlation

Correlation = dependence?

Correlation between the consumption of cheese and deaths by becoming tangled in bedsheets (in the US, see Vigen 2015):



Correlation = 0.95!!

Common fallacies

Fallacy 1: *a small correlation $\rho(X_1, X_2)$ implies that X_1 and X_2 are close to being independent*

- **wrong!**
- Independence implies zero correlation BUT
  - A correlation of zero does not always mean independence.
- See example 1 below.

Fallacy 2: *marginal distributions and their correlation matrix uniquely determine the joint distribution.*

- This is true only for elliptical families (including multivariate normal), **but wrong in general**!
- See example 2 below.

Example 1

Company's two risks X_1 and X_2

- Let $Z \sim N(0,1)$ and $\Pr(U=-1) = 1/2 = \Pr(U=1)$
- U stands for an economic stress generator, *independent* of Z
- Consider:

$$X_1 = Z \sim N(0,1)$$

and

$$X_2 = UZ \sim N(0,1).$$

Now $\text{Cov}(X_1, X_2) = E(X_1X_2) = E(UZ^2) = E(U)E(Z^2) = 0$ hence $\rho(X_1, X_2) = 0$. However, X_1 and X_2 are *strongly dependent*, with 50% probability co-monotone and 50% counter-monotone.

This example can be made more realistic

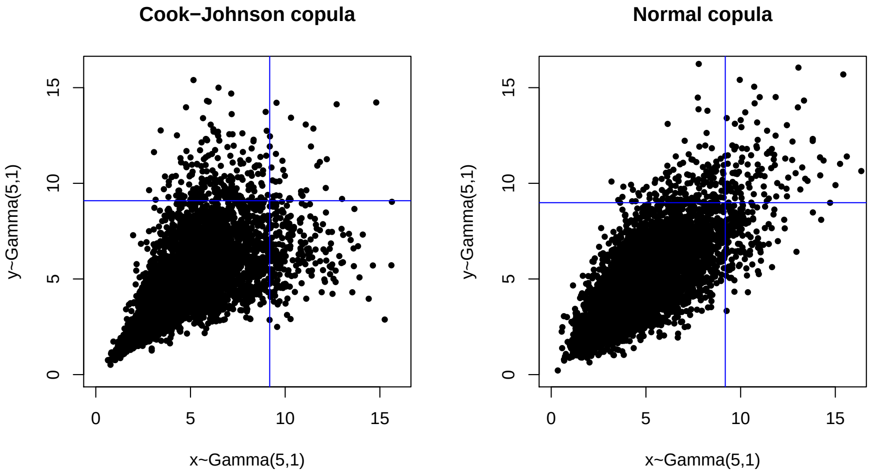
Example 2

Marginals and correlations—not enough to completely determine joint distribution

Consider the following example:

- Marginals: Gamma $(5, 1)$
- Correlation: $\rho = 0.75$
- Different dependence structures: Normal copula vs Cook-Johnson copula
- More generally, check the `Copulatheque`

Example 2 illustration: Normal vs Cook-Johnson copulas



2 Copula theory

- What is a copula?
- Density associated with a copula
- Survival copulas
- Invariance property

Sklar's representation theorem

The copula couples, links, or connects the joint distribution to its marginals.

Sklar (1959): There exists a copula function C such that

$$F\left(x_1, x_2, ..., x_n\right) = C\left(F_1\left(x_1\right), F_2\left(x_2\right), ..., F_n\left(x_n\right)\right)$$

where F_k is the marginal df of X_k, $k = 1, 2, ..., n$. Equivalently,

$$\Pr\left(X_1 \leq x_1, ..., X_n \leq x_n\right) = C\left(\Pr\left[X_1 \leq x_1\right], ..., \Pr\left[X_n \leq x_n\right]\right).$$

In the case of independence,

$$F\left(x_1, x_2, ..., x_n\right) = F_1\left(x_1\right) \cdot F_2\left(x_2\right) \cdots F_n\left(x_n\right)$$

so that

$$C(u_1, u_2, \ldots, u_n) = u_1 \cdot u_2 \cdots u_n.$$

Under certain conditions, the copula

$$C\left(u_1, ..., u_n\right) = F\left(F_1^{-1}\left(u_1\right), ..., F_n^{-1}\left(u_n\right)\right), \quad 0 \leq u_1, u_2, \ldots, u_n \leq 1,$$

is unique, where F_k^{-1} denote the quantile functions.

Note:

- This is one way of constructing copulas.
- These are called implicit copulas.
- Elliptical copulas are a prominent example (e.g., Gaussian copula)

Example

Let

$$F(x,y) = \begin{cases} \dfrac{(x+1)(e^y-1)}{x+2e^y-1} & (x,y) \in [-1,1] \times [0,\infty] \\ 1-e^{-y} & (x,y) \in (1,\infty] \times [0,\infty] \\ 0 & \text{elsewhere} \end{cases}$$

Hence

$$\begin{aligned} F(x,\infty) = F(x) &= \frac{x+1}{2} (\equiv u), \quad x \in [-1,1] \\ F^{-1}(u) &= 2u-1 = x \\ F(1,y) = G(y) &= 1-e^{-y} (\equiv v), \quad y \geq 0 \\ G^{-1}(v) &= -\ln(1-v) = y \end{aligned}$$

Finally,

$$
\begin{aligned}
C(u,v) &= \frac{(2u-1+1)[(1-v)^{-1}-1]}{2u-1+2(1-v)^{-1}-1} = \frac{2u(1-1+v)}{(2u-2)(1-v)+2} \\
&= \frac{2uv}{2u-2uv-2+2v+2} = \frac{uv}{u+v-uv} \\
&= uv \times \frac{1}{u+v-uv}
\end{aligned}
$$

Note:

- Independence copula is $C(u,v) = uv$, here "tweaked" by a function of u and v
- The copula captures the dependence structure, while separating the effects of the marginals (which are behind probabilities u and v).
- Other copulas generally contain parameter(s) to fine-tune the (strength of) dependence.

When is C a valid copula?

For $n = 2$, C is a function mapping $[0, 1]^2$ to $[0, 1]$ that is non-decreasing and right continuous, and:

1. $\lim_{u_k \to 0} C(u_1, u_2) = 0$ for $k = 1, 2$;
2. $\lim_{u_1 \to 1} C(u_1, u_2) = u_2$ and $\lim_{u_2 \to 1} C(u_1, u_2) = u_1$; and
3. C satisfies the inequality $C(v_1, v_2) - C(u_1, v_2) - C(v_1, u_2) + C(u_1, u_2) \geq 0$ for any $u_1 \leq v_1, u_2 \leq v_2$.

Corresponding heuristics are:

1. If the event on one variable is impossible, then the joint probability is impossible.
2. If the event on one variable is certain, then the joint probability boils down to the marginal of the other one.
3. There cannot be negative probabilities.

✠ Multivariate distribution function

We will now generalise this to $n \geq 2$ by first recalling the properties of a multivariate df.

A function $F : R^n \to [0,1]$ is a multivariate d.f. if it satisfies:

0. right-continuous;
1. $\lim_{x_i \to -\infty} F(x_1, ..., x_n) = 0$ for $i = 1, 2, ..., n$;
2. $\lim_{x_i \to \infty, \forall i} F(x_1, ..., x_n) = 1$; and
3. rectangle inequality holds: for all $(a_1, ..., a_n)$ and $(b_1, ..., b_n)$ with $a_i \leq b_i$ for $i = 1, ..., n$, we have

$$\sum_{i_1=1}^{2} \cdots \sum_{i_n=1}^{2} (-1)^{i_1+\cdots+i_n} F(x_{1i_1}, ..., x_{ni_n}) \geq 0,$$

where $x_{i1} = a_i$ and $x_{i2} = b_i$.

Multivariate copula

A copula $C : [0,1]^n \to [0,1]$ is a multivariate distribution function whose univariate marginals are Uniform $(0,1)$.

Properties of a multivariate copula:

1. $C(u_1, ..., u_{k-1}, 0, u_{k+1}, ..., u_n) = 0$
2. $C(1, ..., 1, u_k, 1, ..., 1) = u_k$
3. the rectangle inequality leads us to

$$P(a_1 \leq U_1 \leq b_1, ..., a_n \leq U_n \leq b_n) = \sum_{i_1=1}^{2} \cdots \sum_{i_n=1}^{2} (-1)^{i_1+\cdots+i_n} C(u_{1i_1}, ..., u_{ni_n}) \geq 0$$

for all $u_i \in [0,1]$, $(a_1, ..., a_n)$ and $(b_1, ..., b_n)$ with $a_i \leq b_i$, and $u_{i1} = a_i$ and $u_{i2} = b_i$.

Heuristics are the same as before.

2 Copula theory

Density associated with a copula

For continuous marginals with respective pdf $f_1, ... f_n$, the joint pdf of $\mathbf{X}$ can be written as

$$f(x_1, ..., x_n) = f_1(x_1) \cdots f_n(x_n) \times c(F_1(x_1), ..., F_n(x_n))$$

where the copula density c is given by

$$c(u_1, ..., u_n) = \frac{\partial^n C(u_1, ..., u_n)}{\partial u_1 \partial u_2 \cdots \partial u_n}.$$

Note:

- The copula c distorts the independence case to induce the actual dependence structure.
- If independent, $c(u_1, ..., u_n) = 1$.

Example

Let X and Y be two random variables and the copula C of X and Y is

$$C(u, v) = \frac{uv}{u + v - uv}.$$

Derive its associated density c.

2 Copula theory

THE UNIVERSITY OF MELBOURNE

Survival copulas

What if we want to work with (model) survival functions

$$\overline{F}_i(x_i) = 1 - F_i(x_i) \, (= S_i(x_i))$$

rather than distribution functions?

$\longrightarrow$ We can couple $\overline{F}_i$'s with the **survival copulas** $\overline{C}$.

In the bivariate case, this yields

$$\overline{F}(x_1, x_2) = \Pr[X_1 > x_1, X_2 > x_2] = \overline{C}\left(\overline{F}_1(x_1), \overline{F}_2(x_2)\right),$$

where

$$\overline{C}(1-u, 1-v) = 1 - u - v + C(u, v).$$

This is because

$$\begin{aligned} \Pr[X_1 > x_1, X_2 > x_2] &= 1 - \Pr[X_1 \le x_1] - \Pr[X_2 \le x_2] \\ &\quad + \Pr[X_1 \le x_1, X_2 \le x_2]. \end{aligned}$$

2 Copula theory

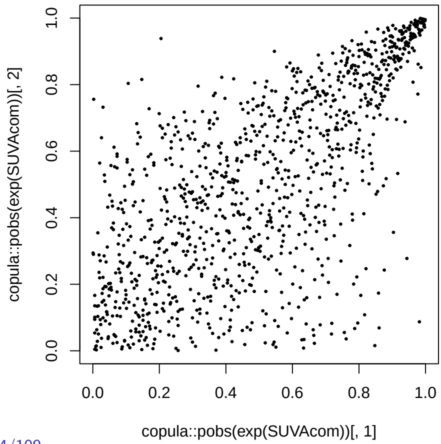
- What is a copula?
- Density associated with a copula
- Survival copulas
- **Invariance property**

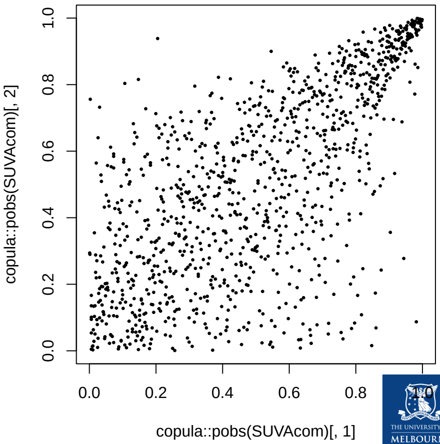
Invariance property

- Suppose random vector $\mathbf{X}$ has copula C and suppose $T_1, \ldots, T_n$ are non-decreasing continuous functions of $X_1, \ldots, X_n$, respectively.
- The random vector defined by $(T_1(X_1), \ldots, T_n(X_n))$ has the same copula C.
  Proof: see Theorem 2.4.3 (p. 25) of Nelsen (1999)
- The usefulness of this property can be illustrated in many ways. If you have a copula describing joint distribution of insurance losses of various types, and you decide the quantity of interest is a transformation (e.g. logarithm) of these losses, then the multivariate distribution structure does not change.
- The copula is then also invariant to inflation.
- Only the marginal distributions change.

Empirical copula of the 1089 common claims (LHS) and of their log (RHS):

```
par(mfrow = c(1, 2), pty = "s")
plot(copula::pobs(exp(SUVAcom))[, 1], copula::pobs(exp(SUVAcom))[,
  2], pch = 20, cex = 0.5)
plot(copula::pobs(SUVAcom)[, 1], copula::pobs(SUVAcom)[, 2],
  pch = 20, cex = 0.5)
```





3 Main bivariate copulas

- The Fréchet bounds
- The Normal (aka Gaussian) copula
- Archimedean copulas
- The Clayton copula
- The Frank copula
- The Gumbel(-Hougard) copula
- Copula models in R

The Fréchet bounds

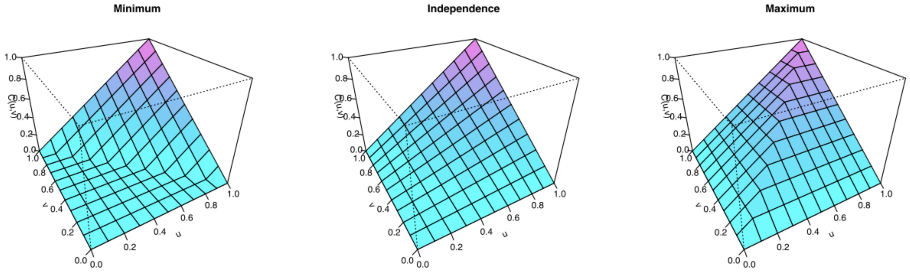
Define the Fréchet bounds as:

- Fréchet lower bound: $L_F(u_1, \ldots, u_n) = \max\left(\sum_{k=1}^{n} u_k - (n-1), 0\right)$
- Fréchet upper bound: $U_F(u_1, \ldots, u_n) = \min(u_1, \ldots, u_n)$

Any copula function satisfies the following bounds:

$$L_F(u_1, \ldots, u_n) \leq C(u_1, \ldots, u_n) \leq U_F(u_1, \ldots, u_n).$$

The Fréchet upper bound satisfies the definition of a copula, but the Fréchet lower bound does not for dimensions $n \geq 3$.



Source: Wikipedia (2020)

3 Main bivariate copulas

The Normal (aka Gaussian) copula

Recall that $\mathbf{Z} = (Z_1, ...Z_n)' \sim MN(\mathbf{0}, \Sigma)$ if its joint p.d.f. is

$$f(z_1, ..., z_n) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} \exp\left[-\frac{1}{2}\mathbf{z}'\Sigma^{-1}\mathbf{z}\right]$$

where $\mathbf{z} = (z_1, ...z_n)$.

Now define its joint distribution by

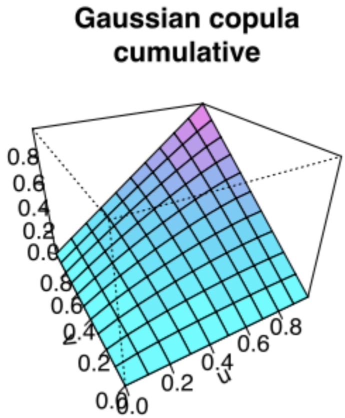
$$\Phi_\Sigma(z_1, ..., z_n) = \int_{-\infty}^{z_n}\int_{-\infty}^{z_{n-1}}\cdots\int_{-\infty}^{z_1} \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} \exp\left[-\frac{1}{2}\mathbf{z}'\Sigma^{-1}\mathbf{z}\right] dz_1 \cdots dz_n$$

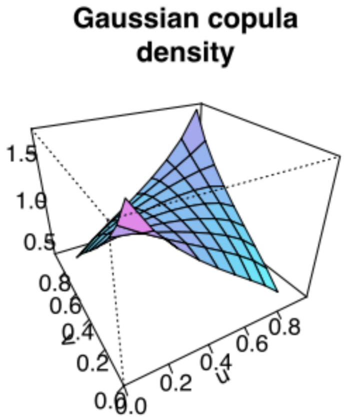
Let $\Phi(\cdot)$ denote the standard normal cumulative distribution function. The copula defined by

$$C(u_1, ..., u_n) = \Phi_\Sigma\left(\Phi^{-1}(u_1), ..., \Phi^{-1}(u_n)\right)$$

is called the Normal (or Gaussian) copula.

Illustration of the Gaussian copula:





Here $\rho = 0.4$. Source: Wikipedia (2020)

The t copula

A r vector $\mathbf{Z} = (Z_1, ...Z_n)' \sim MT\left(\mathbf{0}, \Sigma; \upsilon\right)$ if its joint p.d.f. is

$$f\left(z_1, ..., z_n\right) = \frac{\Gamma\left(\frac{\upsilon+1}{2}\right)}{\sqrt{\left(\upsilon\pi\right)^n |\Sigma| \Gamma\left(\frac{\upsilon}{2}\right)}} \left(1 + \frac{1}{\upsilon}\mathbf{z}'\Sigma^{-1}\mathbf{z}\right)^{-(\upsilon+n)/2}.$$

Now define its joint distribution by

$$T_{\Sigma,\upsilon}\left(z_1, ..., z_n\right) = \int_{-\infty}^{z_n}\int_{-\infty}^{z_{n-1}} \cdots \int_{-\infty}^{z_1} \frac{\Gamma\left(\frac{\upsilon+1}{2}\right)}{\sqrt{\left(\upsilon\pi\right)^n |\Sigma| \Gamma\left(\frac{\upsilon}{2}\right)}} \left(1 + \frac{1}{\upsilon}\mathbf{z}'\Sigma^{-1}\mathbf{z}\right)^{-(\upsilon+n)}$$

Let $t_\upsilon(\cdot)$ denote the cumulative distribution function of a standard univariate t distribution. The copula defined by

$$C\left(u_1, ..., u_n\right) = T_{\Sigma,\upsilon}\left(t_\upsilon^{-1}\left(u_1\right), ..., t_\upsilon^{-1}\left(u_n\right)\right)$$

is called the t copula.

Archimedean copulas

C is Archimedean if it has the form

$$C(u_1, ..., u_n) = \psi^{-1}(\psi(u_1) + \cdots + \psi(u_n))$$

for all $0 \leq u_1, ..., u_n \leq 1$ and for some function ψ (called the generator) satisfying:

1. $\psi(1) = 0$;
2. ψ is decreasing; and
3. ψ is convex.

3 Main bivariate copulas

THE UNIVERSITY OF MELBOURNE

The Clayton copula

The Clayton copula is defined by

$$C\left(u_1, \ldots, u_n\right) = \left(\sum_{k=1}^{n} u_k^{-\theta} - n + 1\right)^{-1/\theta}, \quad \theta \in (0, \infty)$$

It is of Archimedean type with:

- $\psi\left(t\right) = \frac{1}{\theta}(t^{-\theta} - 1)$
- $\psi^{-1}\left(s\right) = \left(1 + \theta s\right)^{-1/\theta}$

Note the correspondance with Kendall's τ (for bivariate case):

$$\theta = \frac{2\tau}{1-\tau} \quad \Longleftrightarrow \quad \tau = \frac{\theta}{2+\theta}$$

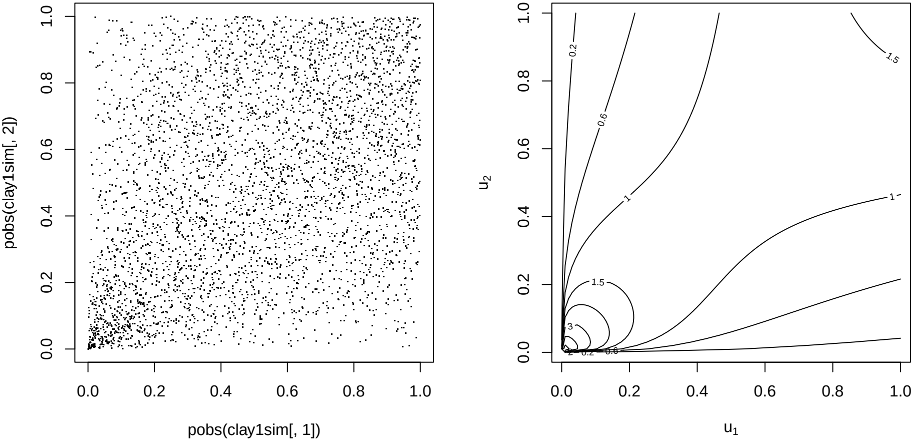
This copula is asymmetric with positive dependence in the left tail.

The Clayton copula for $n = 2$

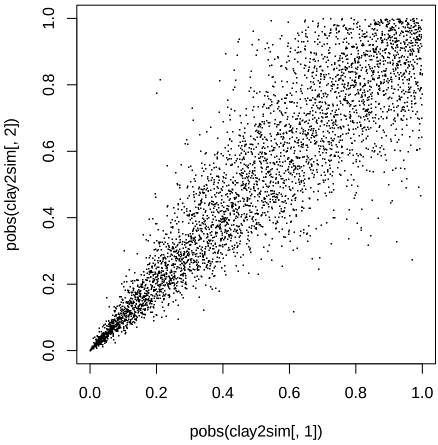
We have

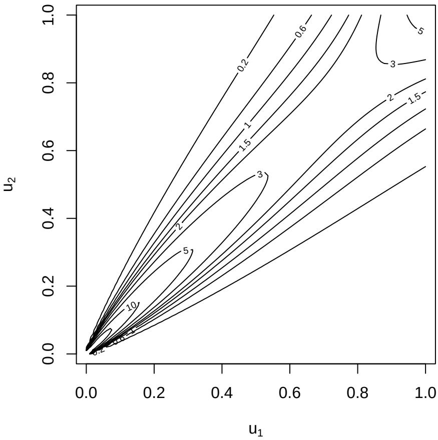
$$C(u_1, u_2) = \left(u_1^{-\theta} + u_2^{-\theta} - 1\right)^{-1/\theta}$$

With parameter $\tau = 0.25$



With parameter $\tau = 0.75$





3 Main bivariate copulas

The Frank copula

The Frank copula is defined by

$$C\left(u_1, \ldots, u_n\right) = \frac{1}{\theta}\log\left(1 + \frac{\prod_{i=1}^{n}\left(e^{\theta u_i} - 1\right)}{e^{\theta} - 1}\right), \quad \theta \in \mathbb{R}\backslash\{0\}$$

It is of Archimedean type with:

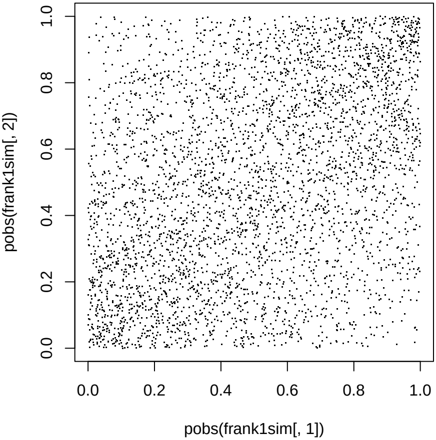
- $\psi\left(t\right) = -\log\left(\frac{e^{-\theta t} - 1}{e^{-\theta} - 1}\right)$
- $\psi^{-1}\left(s\right) = -\frac{1}{\theta}\log\left(1 + e^{-s}\left(e^{-\theta} - 1\right)\right)$

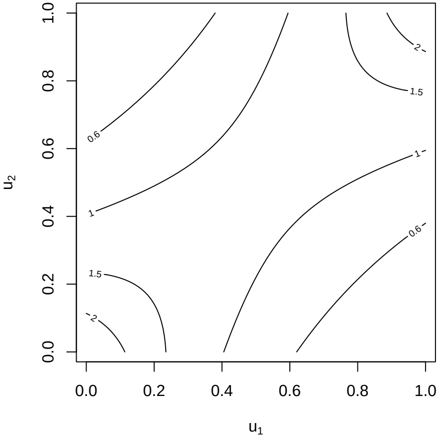
Note the correspondance with Kendall's τ (for bivariate case):

$$\tau = 1 - \frac{4}{\theta} + \frac{4}{\theta^2}\int_0^{\theta}\frac{t}{e^t - 1}dt.$$

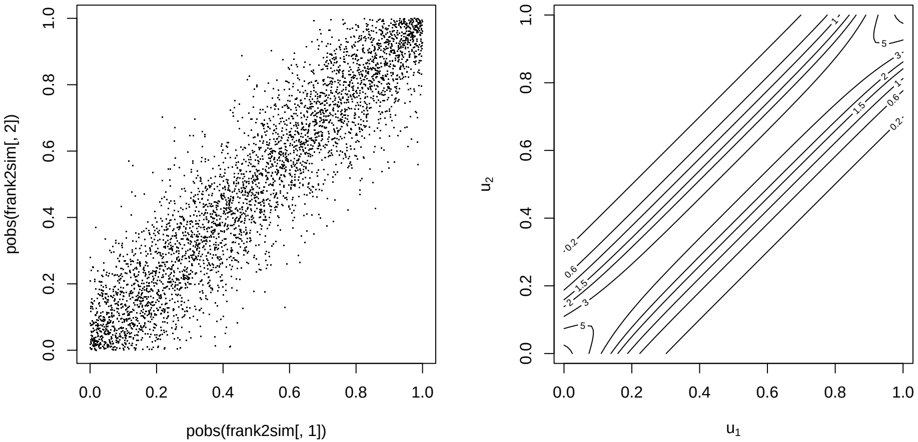
This copula is symmetric.

With parameter $\tau = 0.25$





With parameter $\tau = 0.75$



The Gumbel(-Hougard) copula

The Gumbel copula is defined by

$$C\left(u_1, \ldots, u_n\right) = \exp\left[-\left(\sum_{i=1}^{n}\left(-\log u_i\right)^{\theta}\right)^{1/\theta}\right] \quad \theta \in [1, \infty)$$

It is of Archimedean type with:

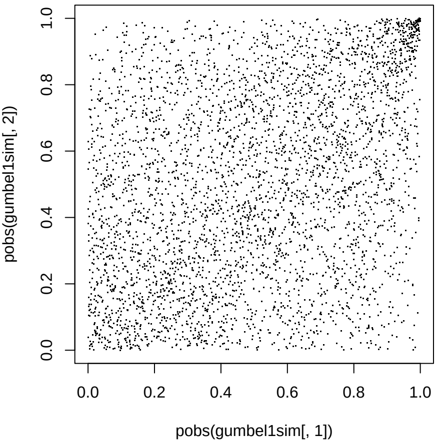
- $\psi\left(t\right) = \left(-\log t\right)^{\theta}$
- $\psi^{-1}\left(s\right) = \exp\left\{-t^{1/\theta}\right\}$

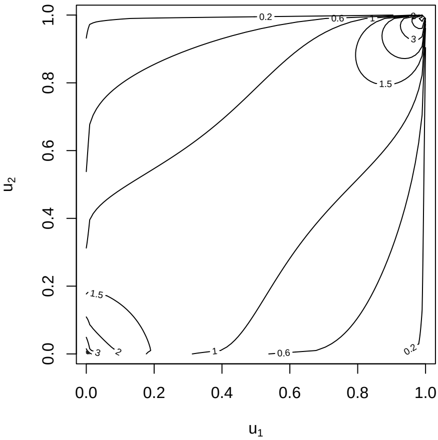
Note correspondance with Kendall's τ (for bivariate case):

$$\theta = \frac{1}{1-\tau} \quad \Longleftrightarrow \quad \tau = \frac{\theta - 1}{\theta}$$

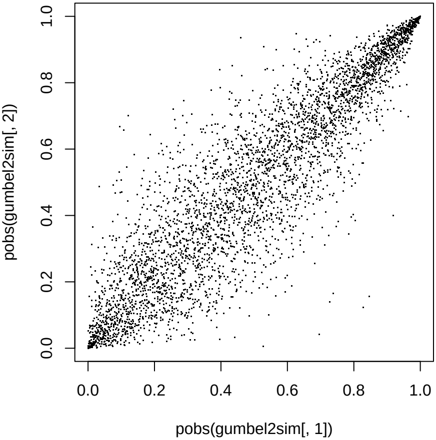
This copula is asymmetric with greater dependence in the right tail, which makes it often a good candidate for large claims with a common underlying cause.

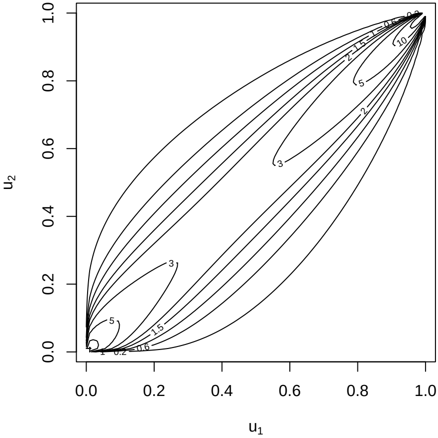
With parameter $\tau = 0.25$





With parameter $\tau = 0.75$





3 Main bivariate copulas

The `VineCopula` package

- The `VineCopula` package caters for many of the basic copula modelling requirements.
- Vine copulas (Kurowicka and Joe 2011) allow for the construction of multivariate copulas with flexible dependence structures; they are outside the scope of this Module.
- The package, however, has a series of modelling functions specifically designed for bivariate copula modelling via the "BiCop-family".

- BiCop: Creates a bivariate copula by specifying the family and parameters (or Kendall's tau). Returns an object of class BiCop. The class has the following methods:
  - print, summary: a brief or comprehensive overview of the bivariate copula, respectively.
  - plot, contour: surface/perspective and contour plots of the copula density. Possibly coupled with standard normal margins (default for contour).
- For most functions, you can provide an object of class BiCop instead of specifying family, par and par2 manually.

Bivariate copulas in `VineCopula`

The following bivariate copulas are available in the `VineCopula` package within the `bicop` family:

Copula family	`family`	`par`	`par2`
Gaussian	1	(-1, 1)	-
Student t	2	(-1, 1)	(2,Inf)
(Survival) Clayton	3, 13	(0, Inf)	-
Rotated Clayton (90° and 270°)	23, 33	(-Inf, 0)	-
(Survival) Gumbel	4, 14	[1, Inf)	-
Rotated Gumbel (90° and 270°)	24, 34	(-Inf, -1]	-
Frank	5	R {0}	-
(Survival) Joe	6, 16	(1, Inf)	-
Rotated Joe (90° and 270°)	26, 36	(-Inf, -1)	-

Copula family	`family`	`par`	`par2`
(Survival) Clayton-Gumbel (BB1)	7, 17	(0, Inf)	[1, Inf)
Rotated Clayton-Gumbel (90° and 270°)	27, 37	(-Inf, 0)	(-Inf, -1]
(Survival) Joe-Gumbel (BB6)	8, 18	[1 ,Inf)	[1, Inf)
Rotated Joe-Gumbel (90° and 270°)	28, 38	(-Inf, -1]	(-Inf, -1]
(Survival) Joe-Clayton (BB7)	9, 19	[1, Inf)	(0, Inf)
Rotated Joe-Clayton (90° and 270°)	29, 39	(-Inf, -1]	(-Inf, 0)
(Survival) Joe-Frank (BB8)	10, 20	[1, Inf)	(0, 1]
Rotated Joe-Frank (90° and 270°)	30, 40	(-Inf, -1]	[-1, 0)
(Survival) Tawn type 1	104, 114	[1, Inf)	[0, 1]
Rotated Tawn type 1 (90° and 270°)	124, 134	(-Inf, -1]	[0, 1]
(Survival) Tawn type 2	204, 214	[1, Inf)	[0, 1]
Rotated Tawn type 2 (90° and 270°)	224, 234	(-Inf, -1]	[0, 1]

All of these copulas are illustrated in the `copulatheque`.

Example of Gumbel copula:

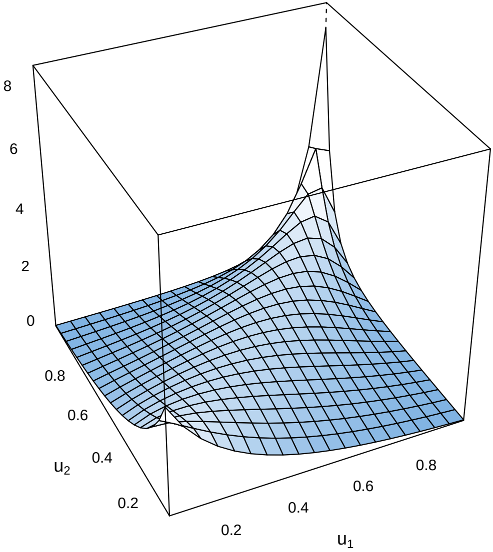
```
cop <- VineCopula::BiCop(4, 2)
print(cop)
```

```
## Bivariate copula: Gumbel (par = 2, tau = 0.5)
```

```
summary(cop)
```

```
## Family
## ------
## No:    4
## Name:  Gumbel
##
## Parameter(s)
## ------------
## par:  2
##
## Dependence measures
## -------------------
## Kendall's tau:    0.5
## Upper TD:         0.59
## Lower TD:         0
```

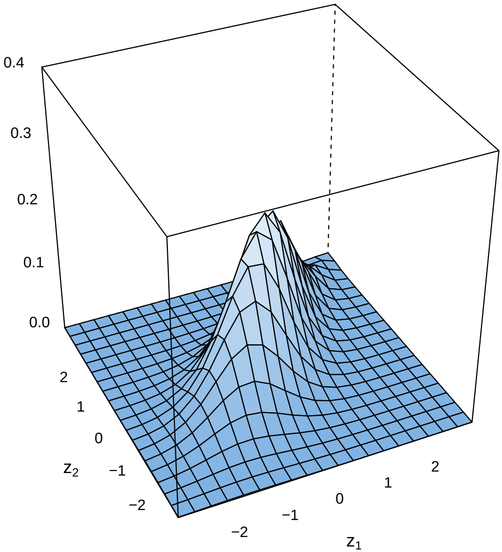
```r
plot(cop)
```



Note this is for uniform margins.

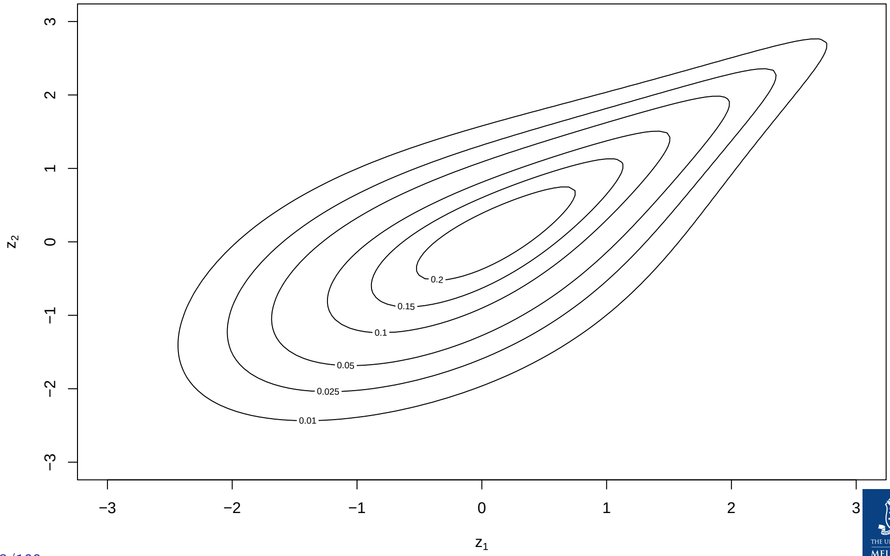
Now with a standard normal margin:

```
plot(cop, type = "surface", margins = "norm")
```



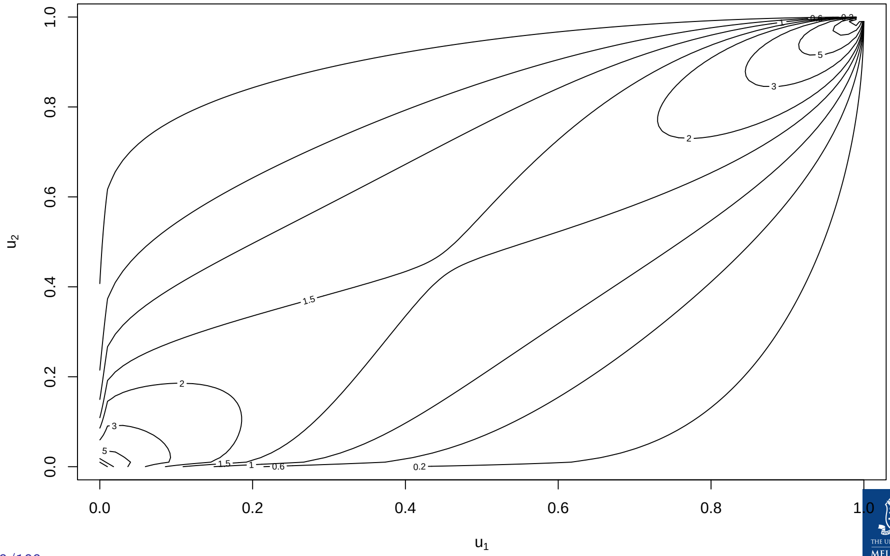
Contour plots are done with normal margins as standard:

```
plot(cop, type = "contour")
```



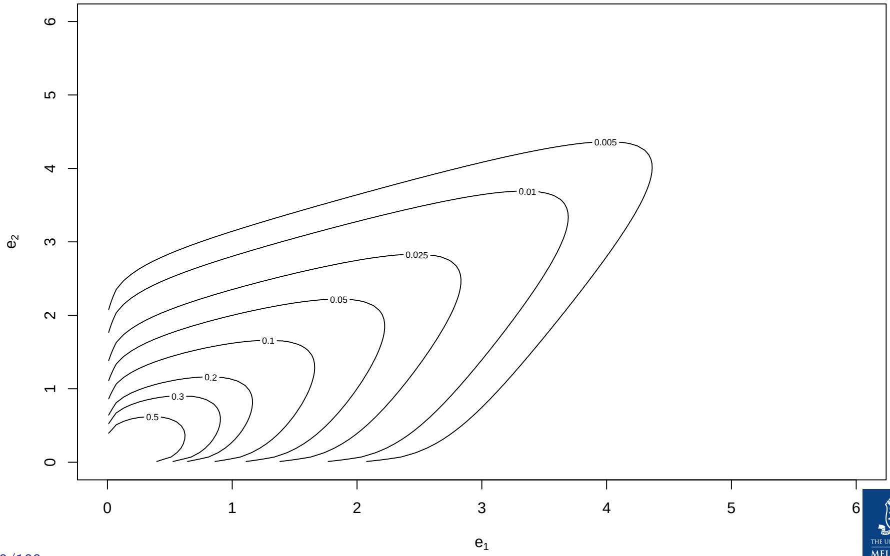
But uniform margins are still possible:

```
plot(cop, type = "contour", margins = "unif")
```



And so are exponential margins in both cases:

```
plot(cop, type = "contour", margins = "exp")
```



Conversion between dependence measures and parameters (for a given family):

- `BiCopPar2Tau`: computes the theoretical Kendall's tau value of a bivariate copula for given parameter values.
- `BiCopTau2Par`: computes the parameter of a (one parameter) bivariate copula for a given value of Kendall's tau.

Example of conversion for Clayton:

```
tau <- BiCopPar2Tau(3, 2.5)
tau
```

```
## [1] 0.5555556
```

```
theta <- 2 * tau/(1 - tau)
theta
```

```
## [1] 2.5
```

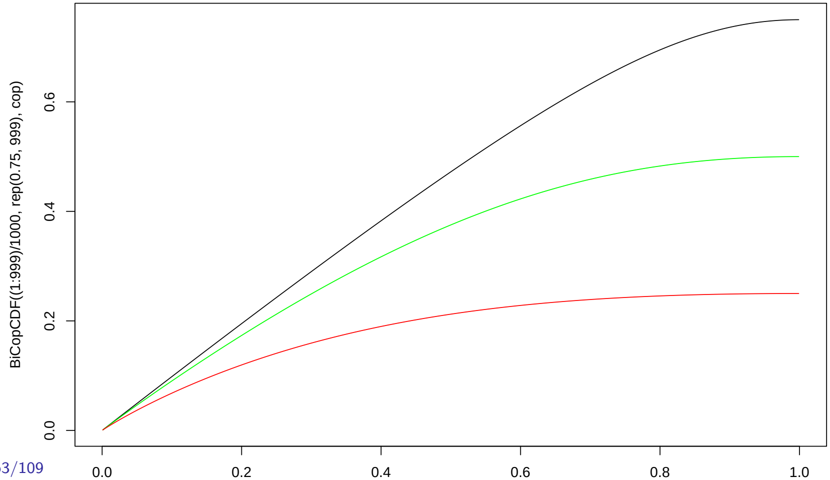
```
BiCopTau2Par(3, tau)
```

```
## [1] 2.5
```

Evaluate functions related to a bivariate copula:

- `BiCopPDF`/`BiCopCDF`: evaluates the pdf/cdf of a given parametric bivariate copula.
- `BiCopDeriv`: evaluates the derivative of a given parametric bivariate copula density with respect to its parameter(s) or one of its arguments.

```
plot((1:999)/1000, BiCopCDF((1:999)/1000, rep(0.75, 999), cop),
  type = "l", xlab = "")  #u_2 = 0.75
lines((1:999)/1000, BiCopCDF((1:999)/1000, rep(0.5, 999), cop),
  type = "l", col = "green")  #u_2 = 0.5
lines((1:999)/1000, BiCopCDF((1:999)/1000, rep(0.25, 999), cop),
  type = "l", col = "red")  #u_2 = 0.25
```



THE UNIVERSITY OF MELBOURNE

4 ✠ Simulation from bivariate copulas

- ✠ Reminder: simulation of a univariate random variable
- ✠ Overarching strategy
- ✠ Preliminary: the conditional distribution function
- ✠ The conditional distribution method
- ✠ Specific algorithms
- ✠ Example of simulation in R

✠ Reminder: simulation of a univariate random variable

- Remember that if $X \in \mathbb{R}$ has distribution function F then

$$F(X) \sim \text{uniform}(0, 1).$$

- This forms the basis of most simulation techniques, as a pseudo-uniform $u \in (0, 1)$ can then be mapped into a pseudo-random $x \in \mathbb{R}$ with df F by applying

$$x = F^{-1}(u).$$

✠ Overarching strategy

- We will introduce the general **conditional distribution method**.
- The overarching idea is (for the bivariate case):
  - simulate two independent **uniform** random variable u and t;
  - "tweak" t into a $v \in [0, 1]$ so that it has the right dependence structure (w.r.t. u) with the help of the copula;
  - map u and v into marginal x and y using their distribution function.
- However, there are some specific, more efficient algorithms that are available for certain types of copulas (see, e.g. Nelsen 1999).
- In R, the function `BiCopSim` will simulate from a given parametric bivariate copula.

✠ Preliminary: the conditional distribution function

For the "tweak", we will need the conditional distribution function for V given $U = u$, which is denoted by $c_u(v)$:

$$\begin{aligned} c_u(v) &= \Pr[V \leq v | U = u] \\ &= \lim_{\Delta u \to 0} \frac{C(u + \Delta u, v) - C(u, v)}{\Delta u} \\ &= \frac{\partial C(u, v)}{\partial u}. \end{aligned}$$

In particular, we will need its inverse.

✠ Example

For the copula

$$C(u,v) = \frac{uv}{u+v-uv}$$

we have

$$\begin{aligned} c_u(v) &= \frac{v(u+v-uv)-uv(1-v)}{(u+v-uv)^2} = \left(\frac{v}{u+v-uv}\right)^2 \equiv t \\ c_u^{-1}(t) &= \frac{\sqrt{t}u}{1-\sqrt{t}(1-u)} \equiv v \end{aligned}$$

✠ In R

For copulas of the `BiCop` family:

- `BiCopHfunc`: evaluates the conditional distribution function $c_u(v)$ (aka h-function) of a given parametric bivariate copula.
- `BiCopHinv`: evaluates the inverse conditional distribution function $c_u^{-1}(v)$ (aka inverse h-function) of a given parametric bivariate copula.
- `BiCopHfuncDeriv`: evaluates the derivative of a given conditional parametric bivariate copula (h-function) with respect to its parameter(s) or one of its arguments.

4 ✠ Simulation from bivariate copulas

✠ The conditional distribution method

Goal: generate a pair of pseudo-random variables (X, Y) with d.f.'s F and G, respectively, with dependence structure described by the copula C.

Algorithm

1. Generate two independent uniform $(0, 1)$ pseudo-random variables u and t;
2. Set $v = c_u^{-1}(t)$;
3. Map (u, v) into (x, y):

$$\begin{aligned} x &= F^{-1}(u); \\ y &= G^{-1}(v). \end{aligned}$$

✠ Example

Let X and Y be exponential with mean 1 and standard Normal, respectively. Furthermore, the copula describing their dependence is such as in the previous example:

$$C(u, v) = \frac{uv}{u + v - uv}$$

Furthermore, you are given the following pseudo-random (independent) uniforms:

$$0.3726791, 0.6189313, 0.75949099, 0.01801882$$

Simulate two pairs of outcomes for (X, Y).

Use of the conditional distribution method yields

1. We can use the uniforms given in the question such that

$$
\begin{aligned}
(u_1, t_1) &= (0.3726791, 0.6189313) \\
(u_2, t_2) &= (0.75949099, 0.01801882)
\end{aligned}
$$

1. Set $v_i = \frac{u_i\sqrt{t_i}}{1-(1-u_i)\sqrt{t_i}}$ for $i = 1, 2$:

$$
\begin{aligned}
v_1 &= 0.5788953 \\
v_2 &= 0.1053509
\end{aligned}
$$

1. Mapping (u_i, v_i) into (x_i, y_i) using

$$x_i = F^{-1}(u_i) = -\ln(1-u_i) \text{ and } y_i = \Phi^{-1}(v_i) \text{ we have}$$

$$
\begin{aligned}
(x_1, y_1) &= (0.466297, 0.199068) \\
(x_2, y_2) &= (1.424998, -1.251638)
\end{aligned}
$$

4 ✠ Simulation from bivariate copulas

- ✠ Reminder: simulation of a univariate random variable
- ✠ Overarching strategy
- ✠ Preliminary: the conditional distribution function
- ✠ The conditional distribution method
- ✠ Specific algorithms
- ✠ Example of simulation in R

✠ Specific algorithms

The following algorithms are provided for illustration purposes. They are not assessable.

✠ Simulation from a Normal copula

Let C be a Normal copula. The following algorithm generates $(x_1, ..., x_n)$ from a random vector $(X_1, \cdots, X_n)$ with marginal distribution functions $F_{X_1}(\cdot), \cdots, F_{X_n}(\cdot)$, and copula C,
i.e. $Pr(X_1 \leq x_1, \cdots, X_n \leq x_n) = C(F_{X_1}(x_1), \cdots, F_{X_n}(x_n))$:

The following algorithm generates $(x_1, ..., x_n)$ from the Normal copula:

1. construct the lower triangular matrix $\mathbf{B}$ so that the correlation matrix $\Sigma = \mathbf{B}\mathbf{B}^\top$ using Cholesky's.
2. generate a column vector of independent standard Normal rv's $\mathbf{Z} = (z_1, ..., z_n)$.
3. take the matrix product of $\mathbf{B}$ and $\mathbf{Z}$, i.e. $\mathbf{Y} = \mathbf{B}\mathbf{Z}$.
4. set $u_k = \Phi(u_k)$ for $k = 1, 2, ..., n$.
5. set $x_k = F_{X_k}^{-1}(u_k)$ for $k = 1, 2, ..., n$.
6. $(x_1, ..., x_n)$ is the desired vector with marginals F_{X_k} for $k = 1, ..., n$ and Normal copula C.

✠ Simulation from a Clayton copula

Let C be the Clayton copula. The following algorithm generates $(x_1, ..., x_n)$ from a random vector $(X_1, \cdots, X_n)$ with marginal distribution functions $F_{X_1}(\cdot), \cdots, F_{X_n}(\cdot)$, and copula C:

1. generate a column vector of independent $Exp(1)$ rv's $\mathbf{Y} = (y_1, ..., y_n)$.
2. generate z from a Gamma$(1/\theta, 1)$ distribution.
3. set $u_k = (1 + y_k/z)^{-1/\theta}$ for $k = 1, 2, ..., n$.
4. set $x_k = F_{X_k}^{-1}(u_k)$ for $k = 1, 2, ..., n$.
5. $(x_1, ..., x_n)$ is the desired vector with marginals F_{X_k} for $k = 1, ..., n$ and Clayton copula C.

4 ✠ Simulation from bivariate copulas

✠ Example of simulation in R

We simulate 4000 pairs (u_1, u_2) from the Gumbel copula (with parameter 2) defined above:

```
Simul.u <- BiCopSim(4000, cop)
head(Simul.u, 15)
```

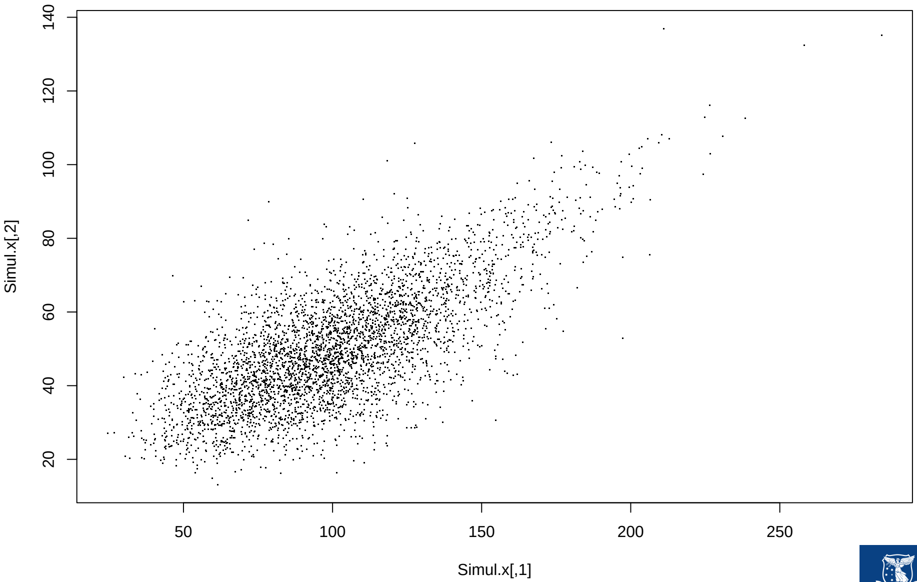
```
##              [,1]       [,2]
##  [1,] 0.63022622 0.84891783
##  [2,] 0.62018178 0.69224302
##  [3,] 0.26803698 0.59279981
##  [4,] 0.46027188 0.08479914
##  [5,] 0.68812025 0.62576048
##  [6,] 0.53865294 0.45607924
##  [7,] 0.18272442 0.64626354
##  [8,] 0.59442051 0.89604623
##  [9,] 0.52365068 0.28709672
## [10,] 0.26790776 0.05741456
## [11,] 0.78487978 0.72546290
## [12,] 0.01873784 0.02228465
## [13,] 0.65186382 0.72115161
## [14,] 0.35020402 0.50497495
```

We then need to map them into the correct margins, say two gammas of shape parameter 10 and mean 100 and 50, respectively:

```
Simul.x <- cbind(qgamma(Simul.u[, 1], 10, 0.1), qgamma(Simul.u[,
  2], 10, 0.2))
head(Simul.x, 15)
```

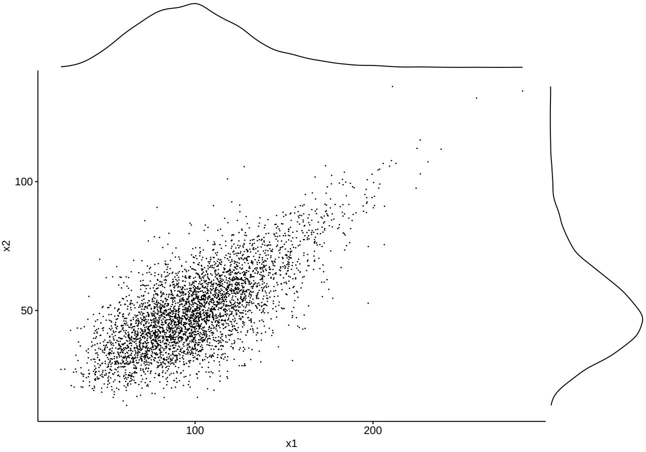
```
##               [,1]     [,2]
##  [1,] 107.36532 66.15562
##  [2,] 106.48693 56.55387
##  [3,]  78.75366 52.07494
##  [4,]  93.62748 30.06336
##  [5,] 112.70600 53.48664
##  [6,]  99.72722 46.65359
##  [7,]  71.28066 54.39720
##  [8,] 104.28606 70.58732
##  [9,]  98.53799 40.14967
## [10,]  78.74307 27.84311
## [11,] 123.25378 58.23779
## [12,]  45.68633 23.51403
## [13,] 109.30176 58.01212
## [14,]  85.24451 48.53715
## [15,]  96.82896 56.70388
```

```
plot(Simul.x, pch = ".")
```



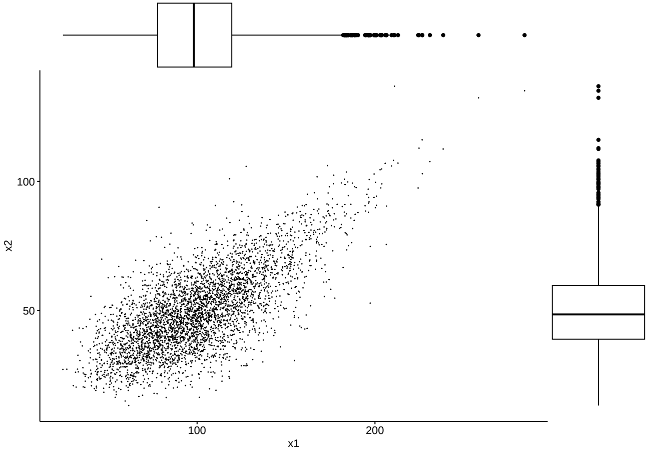
If we use the packages ggplot2, ggpubr and ggExtra one can superimpose density plots:

```
data <- tibble(x1 = Simul.x[, 1], x2 = Simul.x[, 2])
sp <- ggscatter(data, x = "x1", y = "x2", size = 0.05)
ggMarginal(sp, type = "density")
```



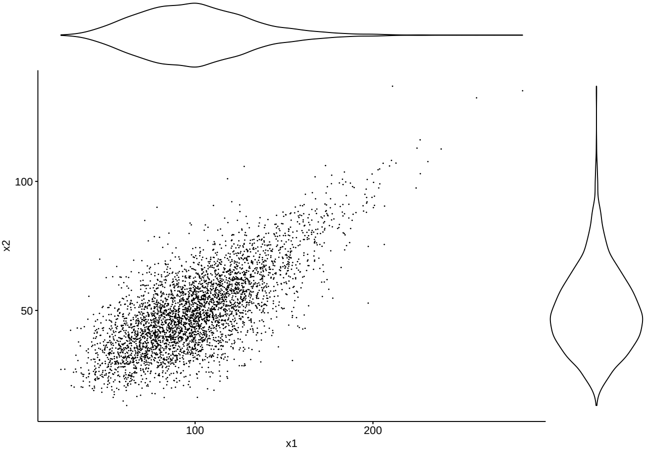
... or boxplots:

```
sp2 <- ggscatter(data, x = "x1", y = "x2", size = 0.05)
ggMarginal(sp2, type = "boxplot")
```



... or violin plots:

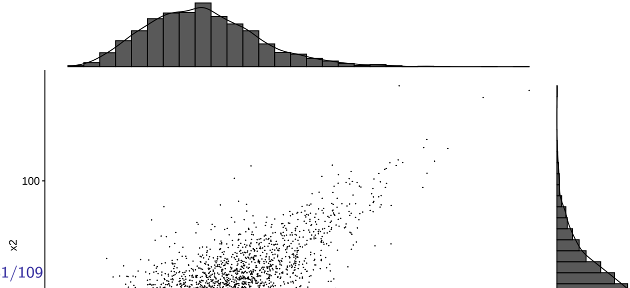
```
sp3 <- ggscatter(data, x = "x1", y = "x2", size = 0.05)
ggMarginal(sp3, type = "violin")
```



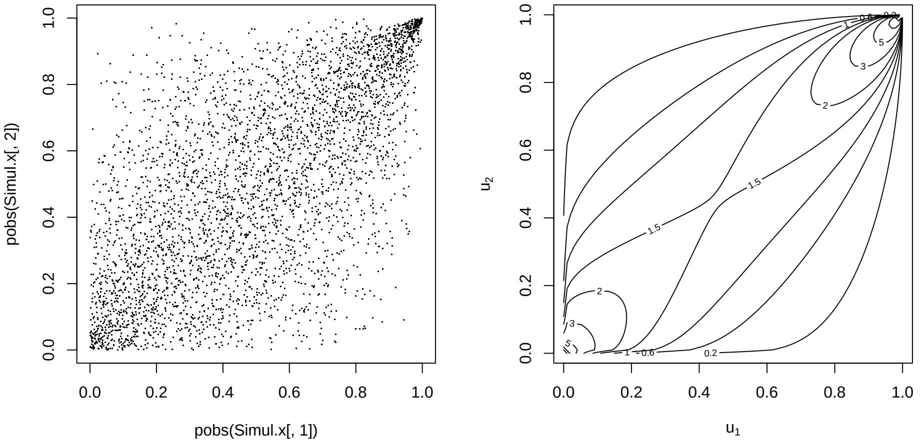
... or densigram plots:

```
sp4 <- ggscatter(data, x = "x1", y = "x2", size = 0.05)
ggMarginal(sp4, type = "densigram")
```

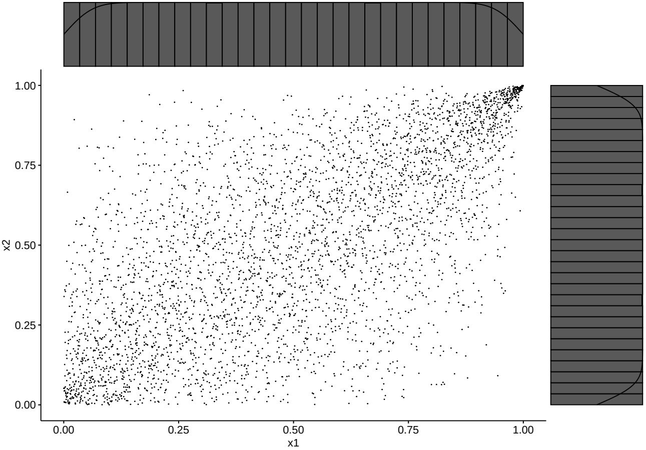
```
## Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2
## i Please use `after_stat(density)` instead.
## i The deprecated feature was likely used in the ggExtra package.
##   Please report the issue at
##   <https://github.com/daattali/ggExtra/issues>.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning
## was generated.
```



```
par(mfrow = c(1, 2), pty = "s")
plot(pobs(Simul.x[, 1]), pobs(Simul.x[, 2]), pch = ".")
plot(cop, type = "contour", margins = "unif")
```



```
data2 <- tibble(x1 = pobs(Simul.x[, 1]), x2 = pobs(Simul.x[,
  2]))
sp3 <- ggscatter(data2, x = "x1", y = "x2", size = 0.05)
ggMarginal(sp3, type = "densigram")
```



Ranks have uniform margins as expected.

5 ✠ Fitting bivariate copulas

- ✠ Using R
- ✠ Case study with `VineCopula`: SUVA data
- ✠ Case study with censoring: ISO data

✠ Using R

The `VineCopula` package offers many functions for fitting copulas:

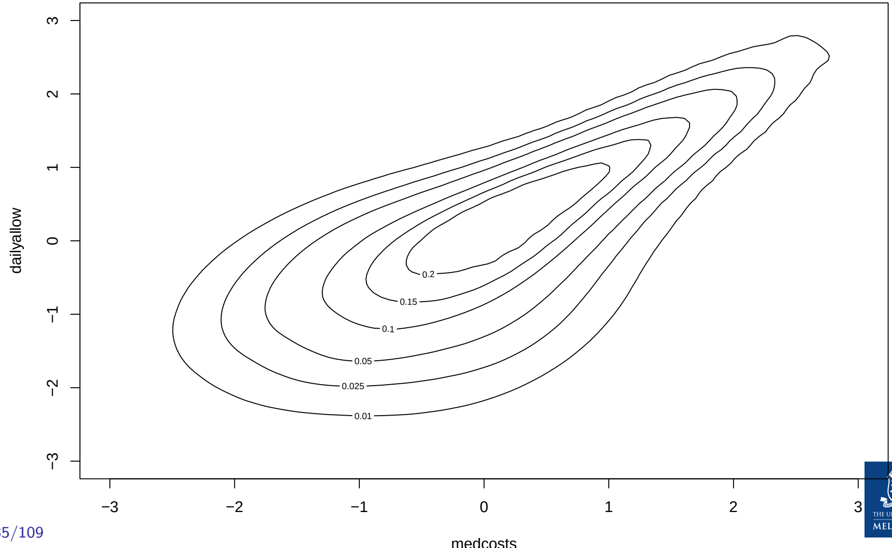
- `BiCopKDE`: A kernel density estimate of the copula density is visualised.
- `BiCopSelect`: Estimates the parameters of a bivariate copula for a set of families and selects the best fitting model (using either AIC or BIC). Returns an object of class `BiCop`.
- `BiCopEst`: Estimates parameters of a bivariate copula with a prespecified family. Returns an object of class `BiCop`. Estimation can be done by
  - maximum likelihood (method = "mle") or
  - inversion of the empirical Kendall's tau (method = "itau", only available for one-parameter families).
- `BiCopGofTest`: Goodness-of-Fit tests for bivariate copulas.

5 ✠ Fitting bivariate copulas

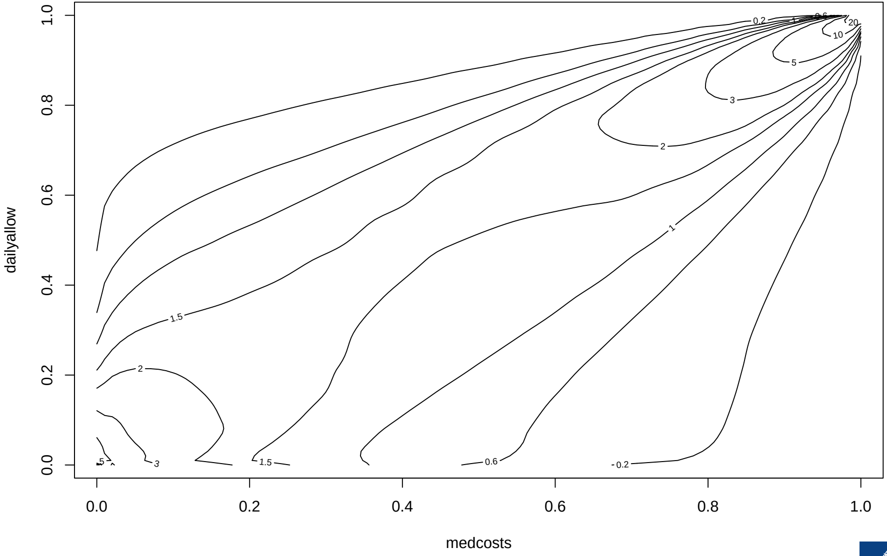
- ✠ Using R
- ✠ Case study with `VineCopula`: SUVA data
- ✠ Case study with censoring: ISO data

✠ Kernel density

```
BiCopKDE(pobs(SUVAcom[, 1]), pobs(SUVAcom[, 2]))
```



```
BiCopKDE(pobs(SUVAcom[, 1]), pobs(SUVAcom[, 2]), margins = "unif")
```



✠ Fitting I

```
SUVAselect <- BiCopSelect(pobs(SUVAcom[, 1]), pobs(SUVAcom[,
  2]), selectioncrit = "BIC")
summary(SUVAselect)
```

```
## Family
## ------
## No:     10
## Name:   BB8
##
## Parameter(s)
## ------------
## par:  3.08
## par2: 0.99
## Dependence measures
## -------------------
## Kendall's tau:    0.52 (empirical = 0.52, p value < 0.01)
## Upper TD:         0
## Lower TD:         0
##
```

✠ Fitting II

```
## Fit statistics
## --------------
## logLik:  488.99
## AIC:    -973.98
## BIC:    -964
```

For comparison:

```
SUVAsurvClayton <- BiCopEst(pobs(SUVAcom[, 1]), pobs(SUVAcom[,
  2]), family = 13)
summary(SUVAsurvClayton)
```

✠ Fitting III

```
## Family
## ------
## No:    13
## Name:  Survival Clayton
##
## Parameter(s)
## ------------
## par:  2.07
##
## Dependence measures
## -------------------
## Kendall's tau:    0.51 (empirical = 0.52, p value < 0.01)
## Upper TD:         0.72
## Lower TD:         0
##
## Fit statistics
## --------------
## logLik:  478.14
## AIC:    -954.28
```

✠ Fitting IV

```
## BIC:     -949.29
```

✠ Further GOF

White's test:

```
BiCopGofTest(pobs(SUVAcom[, 1]), pobs(SUVAcom[, 2]), SUVAselect)
```

```
## Error in BiCopGofTest(pobs(SUVAcom[, 1]), pobs(SUVAcom[, 2]), SUVAselect
```

```
BiCopGofTest(pobs(SUVAcom[, 1]), pobs(SUVAcom[, 2]), SUVAsurvClayton)
```

```
## $statistic
##           [,1]
## [1,] 1.343252
##
## $p.value
## [1] 0.19
```

We cannot perform the test on the BB8 copula (R informs us that `The goodness-of-fit test based on White's information matrix equality is not implemented for the BB copulas.`), but also cannot reject the null on the Survival Clayton.

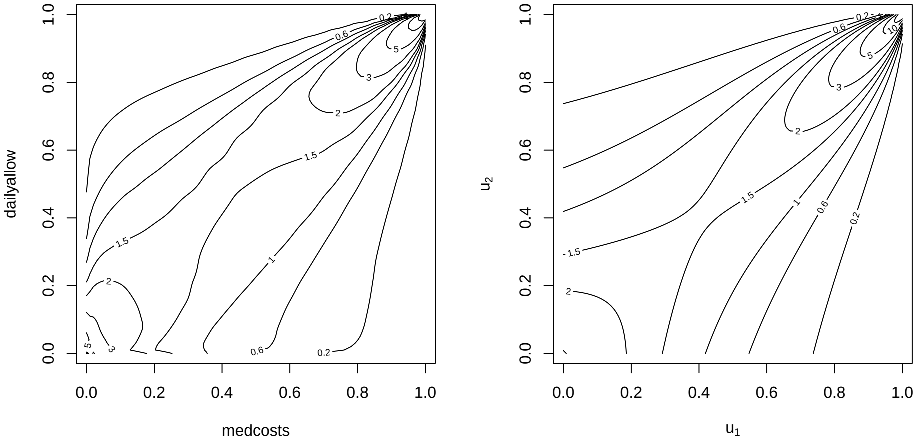
```
BiCopGofTest(pobs(SUVAcom[, 1]), pobs(SUVAcom[, 2]), SUVAselect,
  method = "kendall")
```

```
## $p.value.CvM
## [1] 0.15
##
## $p.value.KS
## [1] 0.22
##
## $statistic.CvM
## [1] 0.08458458
##
## $statistic.KS
## [1] 0.738938
```

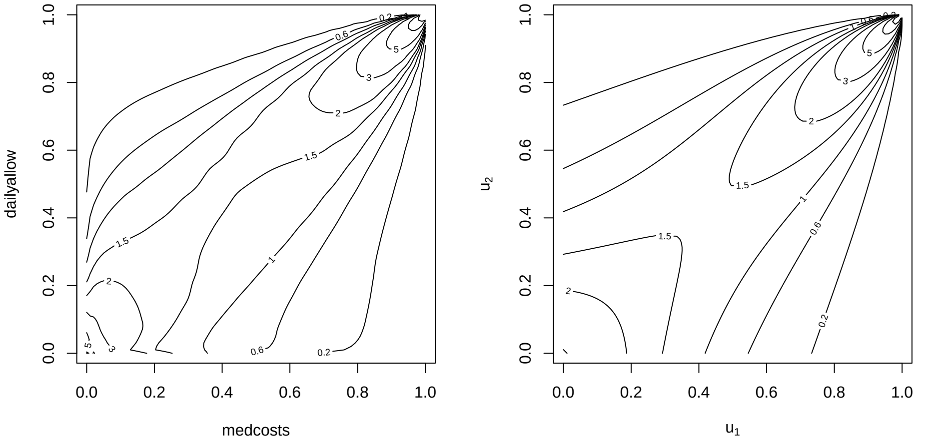
```
BiCopGofTest(pobs(SUVAcom[, 1]), pobs(SUVAcom[, 2]), SUVAsurvClayton,
  method = "kendall")
```

```
## $p.value.CvM
## [1] 0.32
##
## $p.value.KS
## [1] 0.27
##
## $statistic.CvM
## [1] 0.08144334
##
## $statistic.KS
## [1] 0.7731786
```

```
par(mfrow = c(1, 2), pty = "s")
BiCopKDE(pobs(SUVAcom[, 1]), pobs(SUVAcom[, 2]), margins = "unif")
plot(SUVAselect, type = "contour", margins = "unif")
```



```
par(mfrow = c(1, 2), pty = "s")
BiCopKDE(pobs(SUVAcom[, 1]), pobs(SUVAcom[, 2]), margins = "unif")
plot(SUVAsurvClayton, type = "contour", margins = "unif")
```



5 ✠ Fitting bivariate copulas

- ✠ Using R
- ✠ Case study with `VineCopula`: SUVA data
- ✠ Case study with censoring: ISO data

✠ Insurance company losses and expenses

- Data consists of 1,500 general liability claims.
- Provided by the Insurance Services Office, Inc.
- X_1 is the loss, or amount of claims paid.
- X_2 are the ALAE, or Allocated Loss Adjustment Expenses.
- Policy contains policy limits, and hence, censoring.
- δ is the indicator for censoring so that the observed data consists of

$$(x_{1i}, x_{2i}, \delta_i) \text{ for } i = 1, 2, ..., 1500.$$

We will fit this data mostly "by hand" for transparency (and since we need to allow for censoring). R codes are provided separately.

```
Loss.ALAE <- read_csv("LossData-FV.csv")
```

```
## Rows: 1500 Columns: 4
## -- Column specification ------------------------------------------------
## Delimiter: ","
## dbl (4): LOSS, ALAE, LIMIT, CENSOR
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this
```

```
as_tibble(Loss.ALAE)
```

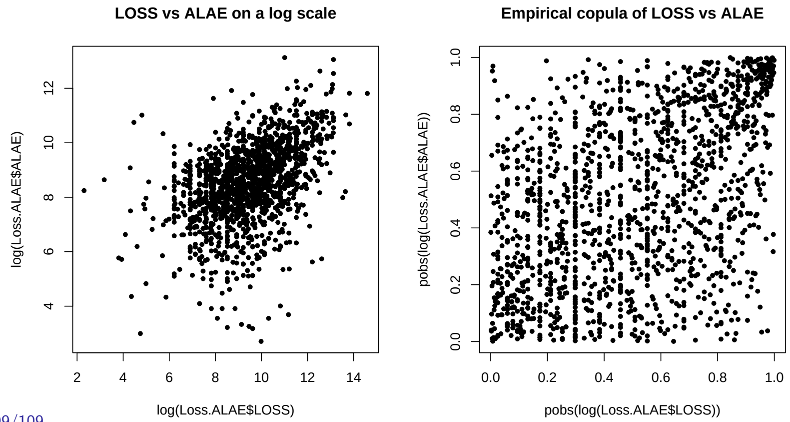
LOSS	ALAE	LIMIT	CENSOR
10	3806	500000	0
24	5658	1000000	0
45	321	1000000	0
51	305	500000	0
60	758	500000	0
74	8768	2000000	0
75	1805	500000	0
78	78	500000	0
87	46534	500000	0
100	489	300000	0

✠ Summary statistics of data

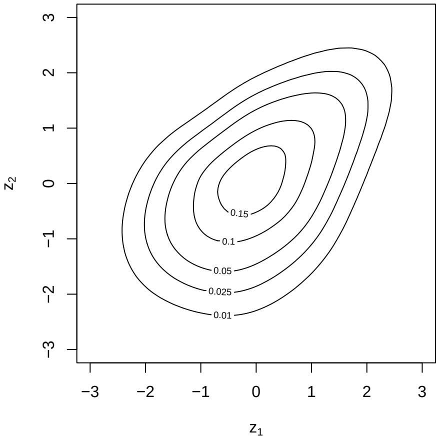
	Loss	ALAE	Policy Limit	Loss (Uncensored)	Loss (Censored)
Number	1,500	1,500	1,352	1,466	34
Mean	41,208	12,588	559,098	37,110	217,491
Median	12,000	5,471	500,000	11,048	100,000
Std Deviation	102,748	28,146	418,649	92,513	258,205
Minimum	10	15	5,000	10	5,000
Maximum	2,173,595	501,863	7,500,000	2,173,595	1,000,000
25th quantile	4,000	2,333	300,000	3,750	50,000
75th quantile	35,000	12,577	1,000,000	32,000	300,000

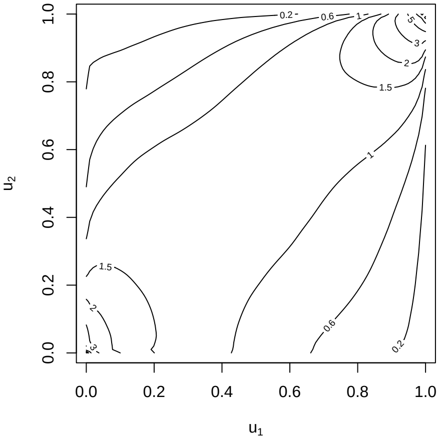
✠ loss vs ALAE

```
par(mfrow = c(1, 2), pty = "s")
plot(log(Loss.ALAE$LOSS), log(Loss.ALAE$ALAE), main = "LOSS vs ALAE on a lo
  pch = 20)
plot(pobs(log(Loss.ALAE$LOSS)), pobs(log(Loss.ALAE$ALAE)), main = "Empirica
  pch = 20)
```



```
par(mfrow = c(1, 2), pty = "s")
BiCopKDE(pobs(log(Loss.ALAE$LOSS)), pobs(log(Loss.ALAE$ALAE)))
BiCopKDE(pobs(log(Loss.ALAE$LOSS)), pobs(log(Loss.ALAE$ALAE)),
  margins = "unif")
```





✠ Maximum likelihood estimation

- Case 1: loss variable is not censored, i.e. $\delta = 0$.

$$f(x_1, x_2) = f_1(x_1) f_2(x_2) C_{12}(F_1(x_1), F_2(x_2))$$

where $C_{12}(u_1, u_2) = \frac{\partial C(u_1, u_2)}{\partial u_1 \partial u_2}$. - Case 2: loss variable is censored, i.e. $\delta = 1$.

$$\begin{aligned} \frac{\partial}{\partial x_2} P(X_1 > x_1, X_2 \leq x_2) &= \frac{\partial}{\partial x_2}[F_2(x_2) - F(x_1, x_2)] \\ &= f_2(x_2) - \frac{\partial}{\partial x_2} F(x_1, x_2) \\ &= f_2(x_2)[1 - C_2(F_1(x_1), F_2(x_2))] \end{aligned}$$

where $C_i(u_1, u_2) = \frac{\partial C(u_1, u_2)}{\partial u_i}$.

✠ Choice of marginals and copulas

- Pareto marginals: $F_k(x_k) = 1 - \left(\frac{\lambda_k}{\lambda_k + x_k}\right)^{\theta_k}$ for $k = 1, 2.$ and $x > 0$.
- For the copulas, several candidates were used:

Copula	$C(u_1, u_2)$	$C_2(u_1, u_2) = \frac{\partial C(u_1, u_2)}{\partial u_2}$	$C_{12}(u_1, u_2) = \frac{\partial^2 C(u_1, u_2)}{\partial u_1 \partial u_2}$
Independence	$u_1 \times u_2$	u_1	1
Clayton	$\left(u_1^{-\alpha} + u_2^{-\alpha} - 1\right)^{-1/\alpha}$	$(C/u_2)^{\alpha+1}$	$(\alpha+1)\, C^{\alpha} \cdot (C/u_1 u_2)^{\alpha+1}$
Gumbel-Hougaard	$\exp\left[-\left((-\log u_1)^{\alpha} + (-\log u_2)^{\alpha}\right)^{1/\alpha}\right]$	$\left(\frac{\log u_2}{\log C}\right)^{\alpha-1} \frac{C}{u_2}$	$\frac{1}{C} C_1 C_2 \left[1 + (\alpha-1)/(-\log C)\right]$
Frank	$\frac{1}{\alpha} \log\left(1 + \frac{(e^{\alpha u_1} - 1)(e^{\alpha u_2} - 1)}{e^{\alpha} - 1}\right)$	$\frac{e^{\alpha u_1}(e^{\alpha u_2} - 1)}{(e^{\alpha} - 1) + (e^{\alpha u_1} - 1)(e^{\alpha u_2} - 1)}$	$\frac{\alpha(e^{\alpha} - 1)\, e^{\alpha(u_1+u_2)}}{\left[(e^{\alpha} - 1) + (e^{\alpha u_1} - 1)(e^{\alpha u_2} - 1)\right]^2}$

✠ Parameter estimates

		Independence		Clayton		Gumbel-Hougaard		Frank	
	Parameter	Estimate	s.e.	Estimate	s.e	Estimate	s.e.	Estimate	s.e.
Loss (X_1)	λ_1	14 552	1 404	14 000	2 033	14 001	1 292	14 323	1 359
	θ_1	1.139	0.067	1.143	0.093	1.120	0.062	1.106	0.064
ALAE (X_2)	λ_2	15 210	1 661	16 059	2 603	14 122	1 409	16 306	1 762
	θ_2	2.231	0.178	2.315	0.261	2.108	0.151	2.274	0.181
Dependence	α	na	na	1.563	0.047	1.454	0.034	-3.162	0.175
Loglik		-31 950.81		-32 777.89		-31 748.81		-31 778.45	
AIC		42.61		43.71		42.34		42.38	

✠ AIC criterion

- Akaike Information Criterion (AIC)
- In the absence of a better way to choosing/selecting a copula model, one may use the AIC criterion defined by

$$\mathrm{AIC} = \left(-2\ell + 2m\right)/n$$

  where ℓ is the value of maximised log-likelihood, m is the number of parameters estimated, and n is the sample size.
- Lower AIC generally is preferred.

✠ Summary

To find the distribution of the sum of dependent random variables with copulas (one approach):

1. Fit marginals independently
2. Describe/fit dependence with a copula (roughly)
   - Get a sense of data (scatterplots, dependence measures)
   - Choose candidate copulas
   - For each candidate, estimate parameters via MLE
   - Choose a copula based on nll(highest) or AIC(lowest)
3. If focusing on the sum, one might do simulations to look at the distributions of aggregates, and how they compare with the original data.

6 Coefficients of tail dependence

- Motivation
- Coefficient of lower tail dependence
- Coefficient of upper tail dependence

Motivation

- In insurance and investment applications it is the large outcomes (losses) that particularly tend to occur together, whereas small claims tend to be fairly independent
- This is one of the reasons why tails (especially right tails) tend to be fatter in financial applications.
- A good understanding of tail behaviour is hence very important.
- It is possible to derive tail properties due to dependence from a copula model.
- The indicator we are considering here is the **coefficient of tail dependence**.
- Tail dependence can take values between 0 (no dependence) and 1 (full dependence).
- `VineCopula::BiCopPar2TailDep` computes the theoretical tail dependence coefficients for copulas of the `BiCop` family.

6 Coefficients of tail dependence

- Motivation
- Coefficient of lower tail dependence
- Coefficient of upper tail dependence

Coefficient of lower tail dependence

The coefficient of **lower tail dependence** is defined as

$$\lambda_L = \lim_{u\to 0^+} \Pr\left[X_1 \le F_{X_1}^{-1}(u) \middle| X_2 \le F_{X_2}^{-1}(u)\right] = \lim_{u\to 0^+} \frac{C(u,u)}{u}.$$

Examples (note the extensive use of de l'Hospital rule):

$$\begin{aligned}
\lambda_L^{\text{ind}} &= \lim_{u\to 0^+} \frac{u \cdot u}{u} = \lim_{u\to 0^+} u = 0 \\
\lambda_L^{\text{Clayton}} &= \lim_{u\to 0^+} \frac{\left(2u^{-\theta}-1\right)^{-1/\theta}}{u} = \lim_{u\to 0^+} \frac{\left(2-u^{\theta}\right)^{-1/\theta} u}{u} \\
&= \lim_{u\to 0^+} \left(2-u^{\theta}\right)^{-1/\theta} = 2^{-1/\theta} = \left(\frac{1}{2}\right)^{\frac{1}{\theta}}
\end{aligned}$$

The lower tail of the Clayton copula is **comprehensive** in that it allows for tail coefficients of 0 (as $\theta \to 0$) to 1 (as $\theta \to \infty$).

6 Coefficients of tail dependence

- Motivation
- Coefficient of lower tail dependence
- Coefficient of upper tail dependence

Coefficient of upper tail dependence

The coefficient of **upper tail dependence** is defined similarly but using the survival copula, which yields

$$\begin{aligned} \lambda_U &= \lim_{u\to1^-} \Pr\left[X_1 \geq F_{X_1}^{-1}(u) \,\middle|\, X_2 \geq F_{X_2}^{-1}(u)\right] \\ &= \lim_{u\to1^-} \frac{\overline{C}(1-u,1-u)}{1-u} = \lim_{u\to0^+} \frac{\overline{C}(u,u)}{u}. \end{aligned}$$

Note $\overline{C}(u,u) = 2u - 1 + C(1-u, 1-u)$. Examples:

$$\begin{aligned} \lambda_U^{\text{ind}} &= \lim_{u\to1^-} \frac{1-2u+u^2}{1-u} = \lim_{u\to1^-} 1-u = 0 \\ \lambda_U^{\text{Frank}} &= \lim_{u\to1^-} \frac{1-2u+1/\theta\left[\log(e^{2\theta u}-2e^{\theta u}+e^{\theta}) - \log(e^{\theta}-1)\right]}{1-u} \\ &= \lim_{u\to1^-} \frac{-2+1/\theta\cdot(2\theta e^{2\theta u}-2\theta e^{\theta u})/(e^{2\theta u}-2e^{\theta u}+e^{\theta})}{-1} \\ &= 0 \end{aligned}$$

References

Avanzi, Benjamin, Luke C. Cassar, and Bernard Wong. 2011. "Modelling Dependence in Insurance Claims Processes with Lévy Copulas." *ASTIN Bulletin* 41 (2): 575–609.

Kurowicka, D., and H. Joe. 2011. *Dependence Modeling Vine Copula Handbook*.

Nelsen, R. B. 1999. *An Introduction to Copulas*. Springer.

Sklar, A. 1959. "Fonctions de Répartition à *n* Dimensions Et Leurs Marges." *Publications de l'Institut de Statistique de l'Université de Paris* 8: 229–31.

Vigen, Tyler. 2015. "Spurious Correlations (Last Accessed on 18 March 2015 on `http://www.tylervigen.com`)."

Wikipedia. 2020. "Copula: Probability Theory." https://www.wikiwand.com/en/Copula_(probability_theory).