Technische Universität München

Zentrum Mathematik

Truncated and simplified regular vines and their applications

Diplomarbeit

von

Eike Christian Brechmann

Themenstellerin:	Prof. Claudia Czado, Ph.D.
Betreuerin:	Prof. Claudia Czado, Ph.D.
Abgabetermin:	28. Oktober 2010

Hiermit erkläre ich, dass ich die Diplomarbeit selbstständig angefertigt und nur die angegebenen Quellen verwendet habe.

Garching, den 28. Oktober 2010

Acknowledgments

First and foremost, I would like to express my sincere gratitude to Prof. Claudia Czado, Ph.D., for her excellent supervision and valuable support in the last months. I am very thankful for all the guidance and help that I received from her.

Furthermore, I want to thank Dr. Kjersti Aas and Prof. Arnoldo Frigessi for the fruitful discussions that helped me to develop a deep understanding of the subject.

Finally, I also want to thank my parents and my girlfriend Susanne for their support and encouragement during my years of study.

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

Introduced by Bedford and Cooke (2001, 2002) and discussed in detail in Kurowicka and Cooke (2006) vines are a flexible class of high-dimensional dependency models which use only bivariate copulas as building blocks. Recently, Aas et al. (2009) considered the special cases of canonical vines (C-vines) and D-vines to derive multivariate copulas using pair copula decompositions. In an application to financial time series data they constructed and analyzed a dependency model with appropriate pair copulas such as the bivariate t, Clayton and Gumbel copulas. However, in high dimensions C- and D-vines impose a rather restrictive dependency structure on the data. The more general class of regular vines (R-vines) is less restrictive in this regard and a convenient graphical model to describe pair copula constructions. Inference of R-vines using graph theoretic algorithms has been developed in Dißmann (2010).

The flexibility though comes along with an exponentially increasing complexity in high dimensions (see Morales-Nápoles et al. (2010) for the number of possible R-vines). In order to counteract this problem and the issue of restricted time or resource availability when constructing models, we propose different approaches to facilitate the construction of regular vines using iterative Vuong tests (Vuong 1989) and Akaike and Bayesian information criteria (Akaike (1973), Schwarz (1978)). In particular, we consider two types of facilitated models: *simplification* means that we replace pair copulas in higher order R-vine trees by Gaussian copulas, while *truncation* refers to the greatest possible "simplification" which implies replacing higher order tree copulas with independence copulas, i.e., if it is possible to truncate an R-vine, this means a significant reduction of computational complexity both for maximum likelihood estimation of parameters and for simulating from the model, e.g., to compute the Value-at-Risk of a portfolio which requires a large number of simulations.

As a special case we consider the joint simplification of C-vines with a multivariate Gaussian copula to capture the dependency remaining in the model after modeling a certain number of trees with pair copulas as introduced by Valdesogo (2009). We extend his results to the more general case of R-vines with "C-vine-like" structures. The fit of the multivariate Gaussian copula is assessed using copula goodness-of-fit tests as discussed in Berg (2009) and Genest et al. (2009). In this context, we also consider the problem of joint truncation where we use multivariate independence tests to determine whether variables of higher order trees are jointly independent.
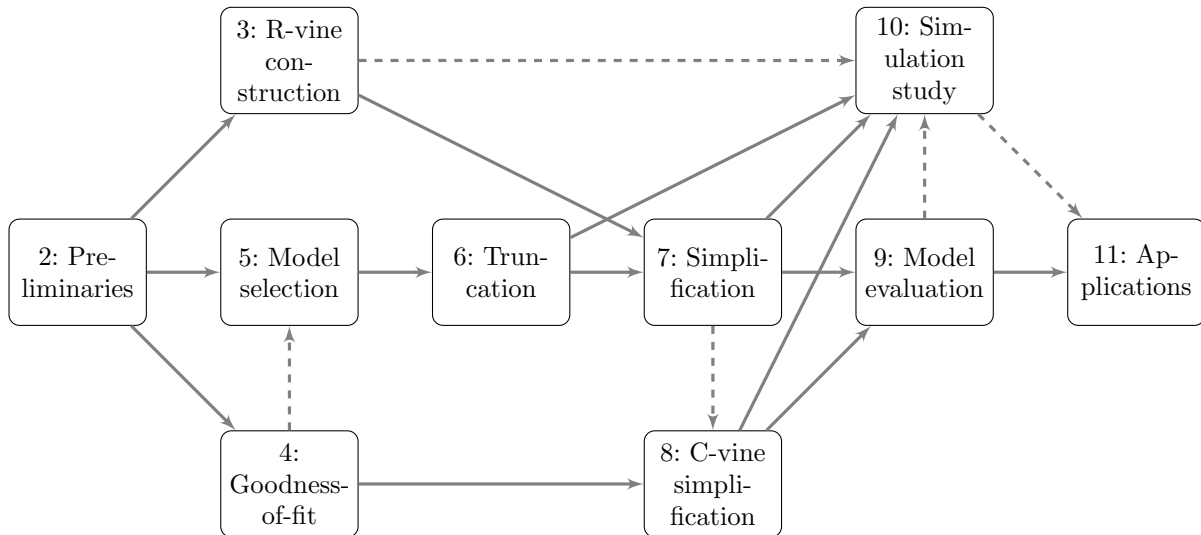
Figure 1.1: Paths through the thesis.

Due to the high complexity, our models have to be constructed heuristically using a sequential procedure. Appropriate construction principles of R-, C- and D-vines are discussed as well as selection criteria of the best-fitting pair copula for given bivariate data, such as the AIC, Vuong tests and bivariate goodness-of-fit tests as proposed by Genest and Rémillard (2008). Bivariate copulas are chosen from a range of ten one- and two-parameter bivariate copula families which include, e.g., rotated Clayton and Gumbel copulas to model possible asymmetric negative dependencies. These methods as well as the truncation and simplification procedures are validated by extensive simulation studies using adequate model selection and evaluation criteria.

Based on simplified C-vines, Heinen and Valdesogo (2009) constructed an extended version of the Capital Asset Pricing Model that can account for non-linearity and non-normality in the dependence structure of assets. We construct such a model for 46 stocks of the Euro Stoxx 50 as well as five leading national stock indices and evaluate it in detail, since it imposes some rather restrictive independence assumptions in order to fit a factor model to a C-vine structure. Moreover, we develop an alternative model based on the more flexible structure of R-vines, the so-called regular vine market sector model, and compare the models.

We also analyze a 19-dimensional financial data set from Norway with respect to dependencies among different asset classes and asymmetric (tail) behavior of returns. Truncation and simplification results allow for insightful statements regarding the type and quality of dependencies. Furthermore, we investigate dependencies between exchange rates to the US Dollar and complement results of Schepsmeier (2010) and Dißmann (2010).

The thesis is organized as follows, where connections between the chapters and possible paths through the thesis are shown in Figure 1.1. In Chapter 2, we recapitulate basic results from the theory of copulas and dependence measures as well as graph theoretical fundamentals as basis of R-vines. Moreover, linear and non-linear time series models are considered which are needed for transforming observed data to be suitable for working

with copulas. Subsequently, the construction of R-vines, in particular of R-vine trees, is discussed in Chapter 3 and different dependence measures are proposed which are required for the construction. In particular, we discuss the construction of D-vines which turns out to be an extremely challenging problem. In Chapter 4, copula goodness-of-fit as well as independence tests are described, which will be used for joint simplification and truncation of C-vines as well as for bivariate copula selection. The latter issue is discussed in Chapter 5 after introducing the common model selection criteria AIC and BIC as well as the Vuong test. In Chapters 6 and 7, we first turn to the problem of R-vine truncation and then consider simplification, where appropriate model specification procedures are motivated by those for truncation. Joint simplification of C-vines is subsequently discussed in Chapter 8. Using appropriate model evaluation criteria, which are proposed in Chapter 9, the simplification and truncation procedures are extensively evaluated in different simulations studies in Chapter 10. Finally, we apply all aspects of model specification, which are discussed in this thesis, to three different data sets in Chapter 11. The models we obtain are critically evaluated and economical interpretations are given. The thesis closes with a summary of the main points and an outlook for future research in Chapter 12.

Chapter 2

Preliminaries

In this chapter we present some of the basic concepts that are used throughout the thesis. In particular, we introduce copulas and discuss the important classes of elliptical and Archimedean copulas. Then we present a broad range of bivariate copula families and their relationships to the common dependence measures Kendall's τ and tail dependence. Bivariate copulas are the building blocks of flexible multivariate copula models, so-called pair copula constructions. Since the number of possible pair copula constructions is quite large, they need to be classified using regular vines which are based on graph theoretical concepts for modeling dependence. Finally, we consider time series models which will be used to analyze and pre-process data in our applications in Chapter 11.

2.1 Copulas

In the following, we mainly follow Nelsen (2006), while an illustrative introduction to the topic is given in Genest and Favre (2007) and further information can be found, e.g., in Joe (1997).

According to Nelsen (2006), *copulas* can briefly be described as "functions that join or 'couple' multivariate distribution functions to their one-dimensional marginal distribution functions". This description is motivated by the important result of Sklar (1959) and explains the name "copula". It also highlights the main feature of copulas: they allow to model dependency among random variables separately of their margins. We will now formally discuss the concept of copulas and their properties. In doing so we first shortly discuss bivariate copulas by way of illustration and then turn to general multivariate copulas.

Definition 2.1 (Bivariate opula.) *A two-dimensional copula is a function $C : [0,1]^2 \to [0,1]$ with the following properties:*

(i) For every $u_1, u_2 \in [0,1]$,
$$C(u_1, 0) = 0 = C(0, u_2)$$

and

$$C(u_1, 1) = u_1 \ \text{and} \ C(1, u_2) = u_2.$$

(ii) C is 2-increasing, i.e., for every $u_{11}, u_{21}, u_{12}, u_{22} \in [0,1]$ such that $u_{11} \leq u_{21}$ and $u_{12} \leq u_{22}$,

$$C(u_{21}, u_{22}) - C(u_{21}, u_{12}) - C(u_{11}, u_{22}) + C(u_{11}, u_{12}) \geq 0.$$

Hence, a two-dimensional copula is a bivariate distribution function with uniform margins, i.e., $C(u_1, u_2) = P(U_1 \leq u_1, U_2 \leq u_2)$ for uniform random variables U_1 and U_2. In contrast, the *joint survival function* $\bar{C}$ is given by

$$\bar{C}(u_1, u_2) = P(U_1 > u_1, U_2 > u_2) = 1 - u_1 - u_2 + C(u_1, u_2). \tag{2.1}$$

The simplest example of a copula is given by the bivariate independence copula $\Pi(u_1, u_2) = u_1 u_2$. Property (i) of Definition 2.2 is obviously satisfied, while property (ii) can easily be checked:

$$\Pi(u_{21}, u_{22}) - \Pi(u_{21}, u_{12}) - \Pi(u_{11}, u_{22}) + \Pi(u_{11}, u_{12}) = u_{21} u_{22} - u_{21} u_{12} - u_{11} u_{22} + u_{11} u_{12}$$
$$= (u_{21} - u_{11})(u_{22} - u_{12}) \geq 0,$$

for all $u_{11}, u_{21}, u_{12}, u_{22} \in [0,1]$ such that $u_{11} \leq u_{21}$ and $u_{12} \leq u_{22}$.

Definition 2.2 can be generalized to the multivariate case. Property (i) directly translates to arbitrary dimension d, but the second characteristic property of copulas has to be extended.

Definition 2.2 (Copula.) *A d-dimensional copula is a function $C : [0,1]^d \to [0,1]$ with the following properties:*

(i) *For every $\boldsymbol{u} = (u_1, ..., u_d)' \in [0,1]^d$,*

$$C(\boldsymbol{u}) = 0 \text{ if at least one coordinate of } \boldsymbol{u} \text{ is 0,}$$

and for all $j = 1, ..., d$,
$$C(1, ..., 1, u_j, 1, ..., 1) = u_j.$$

(ii) *C is d-increasing (see Nelsen (2006)).*

As in the bivariate case, d-dimensional copulas are thus multivariate distribution functions with uniform margins. An important property of them is stated in the following theorem.

Theorem 2.3 (Fréchet-Hoeffding bounds.) *Let C be a d-dimensional copula. Then for every $\boldsymbol{u} \in [0,1]^d$,*
$$W^d(\boldsymbol{u}) \leq C(\boldsymbol{u}) \leq M^d(\boldsymbol{u}), \tag{2.2}$$
where $W^d(\boldsymbol{u}) := \max(u_1 + ... + u_d - d + 1, 0)$ and $M^d(\boldsymbol{u}) := \min(u_1, ..., u_d)$.

It can be shown that the lower bound M^d is a copula, while the upper bound W^d is a copula only for $d = 2$. We will exploit this in Section 3.1.5 and denote the bivariate Fréchet-Hoeffding bounds by $W(u_1, u_2) := W^2(u_1, u_2) = \max(u_1 + u_2 - 1, 0)$ and $M(u_1, u_2) := M^2(u_1, u_2) = \min(u_1, u_2)$.

The most fundamental theorem, which constitutes the important role of copulas for describing dependence in statistics, is the theorem of Sklar (1959). It establishes the link between multivariate distribution functions and their univariate margins. We only state it for the continuous case which is relevant for us.

Theorem 2.4 (Sklar.) *Let F be a d-dimensional distribution function with continuous margins $F_1, ..., F_d$. Then there exists a unique copula C such that for all $\boldsymbol{x} = (x_1, ..., x_d)' \in (\mathbb{R} \cup \{-\infty, \infty\})^d$,*

$$F(\boldsymbol{x}) = C(F_1(x_1), ..., F_d(x_d)). \tag{2.3}$$

Conversely, if C is a copula and $F_1, ..., F_d$ are distribution functions, then the function F defined by (2.3) is a joint distribution function with margins $F_1, ..., F_d$.

Statement (2.3) directly yields a construction method for copulas, the so-called *inversion method*. Given a multivariate distribution function F with invertible margins $F_1, ..., F_d$, we easily obtain a copula by

$$C(\boldsymbol{u}) = F(F_1^{-1}(u_1), ..., F_d^{-1}(u_d)). \tag{2.4}$$

Examples of copula families that are constructed in this way are the Gaussian and t copulas (see (2.7) and (2.9)).

In terms of random variables and their distribution functions Sklar's Theorem 2.4 states that, if $X_i \sim F_i$, $i = 1, ..., d$, and $\mathbf{X} = (X_1, ..., X_d) \sim F$, where $F_1, ..., F_d$ are continuous, then there exists a *unique* copula such that (2.3) holds. Hence C will be called *the copula of* $\mathbf{X}$ and describes the dependence between $X_1, ..., X_d$.

Furthermore, if all necessary derivatives exist, the *copula density* c can be derived by partially differentiating and applying the chain rule as

$$f(\boldsymbol{x}) = \frac{\partial^d C(F_1(x_1), ..., F_d(x_d))}{\partial x_1 ... \partial x_d} = \frac{\partial^d C(F_1(x_1), ..., F_d(x_d))}{\partial F_1(x_1) ... \partial F(x_d)} f_1(x_1) ... f_d(x_d) \tag{2.5}$$

$$\Leftrightarrow \quad c(F_1(x_1), ..., F_d(x_d)) := \frac{\partial^d C(F_1(x_1), ..., F_d(x_d))}{\partial F_1(x_1) ... \partial F_d(x_d)} = \frac{f(\boldsymbol{x})}{f_1(x_1) ... f_d(x_d)},$$

where $f_1, ..., f_d$ and f denote the density functions corresponding to $F_1, ..., F_d$ and F, respectively.

If C is the d-dimensional independence copula $\Pi^d(\boldsymbol{u}) = u_1 ... u_d$, then

$$\pi^d(F_1(x_1), ..., F_d(x_d)) := \frac{\partial^d \Pi^d(F_1(x_1), ..., F_d(x_d))}{\partial F_1(x_1) ... \partial F_d(x_d)} = \frac{\partial^d [F_1(x_1) ... F_d(x_d)]}{\partial F_1(x_1) ... \partial F_d(x_d)} = 1,$$

and hence Equation (2.5) reduces to the well-known factorization of densities if random variables $X_1, ..., X_d$ are independent:

$$f(\boldsymbol{x}) = f_1(x_1) ... f_d(x_d). \tag{2.6}$$

This motivates the following convenient description of multivariate independence based on copulas. Furthermore, the Fréchet-Hoeffding bounds characterize perfect negative (for $d = 2$) and positive dependence.

Theorem 2.5 *For $d \geq 2$, let $X_1, ..., X_d$ be continuous random variables with copula C. Then*

(i) *$X_1, ..., X_d$ are independent iff $C = \Pi^d$,*

(ii) *each of the random variables $X_1, ..., X_d$ is almost surely a strictly increasing function of any of the others iff $C = M^d$, and*

(iii) *X_1 and X_2 are almost surely strictly increasing functions of each other iff $C = W^2$.*

Another important property of copulas is stated in the following theorem:

Theorem 2.6 *Let $X_1, ..., X_d$ be continuous random variables with copula C. Then C is invariant under strictly increasing transformations of $X_1, ..., X_d$.*

In the following, we will discuss two important classes of copulas: elliptical and Archimedean copulas which will be fundamental in our further analyses. However note that there are also copulas which are neither elliptical nor Archimedean.

2.1.1 Elliptical copulas

Elliptical copulas are copulas generated by elliptical distributions using the inversion method (2.4) as discussed above (see, e.g., Owen and Rabinovitch (1983)).

Definition 2.7 (Elliptical distribution.) *The d-dimensional random vector $\mathbf{X}$ has an elliptical distributions iff the density function $f_{\mathbf{X}}(\boldsymbol{x})$ has the representation*

$$f_{\mathbf{X}}(\boldsymbol{x}) = c_d |\Sigma|^{-\frac{1}{2}} g((\boldsymbol{x} - \boldsymbol{\mu})' \Sigma^{-1}(\boldsymbol{x} - \boldsymbol{\mu})),$$

with some constant $c_d \in \mathbb{R}$, mean vector $\boldsymbol{\mu} \in \mathbb{R}^d$, $\Sigma \in \mathbb{R}^{d \times d}$ symmetric positive definite, and some function g which is independent of d.

The most famous example of an elliptical distribution is the multivariate normal distribution with $c_d = (2\pi)^{-d/2}$ and $g(s) = \exp(-\frac{1}{2}s) \ \forall s > 0$. Using the inversion method (2.4), this yields the multivariate Gaussian copula as

$$C(\boldsymbol{u}) = \Phi_R \left(\Phi^{-1}(u_1), ..., \Phi^{-1}(u_d) \right), \tag{2.7}$$

where Φ^{-1} denotes the inverse of the standard normal cumulative distribution function (cdf) Φ and Φ_R the multivariate standard normal cdf with symmetric positive definite correlation matrix $R \in [-1, 1]^d$. The density is then given by

$$c(\boldsymbol{u}) = |R|^{-\frac{1}{2}} \exp(\frac{1}{2}\boldsymbol{x}'(I_d - R^{-1})\boldsymbol{x}), \tag{2.8}$$

where $\boldsymbol{x} = (x_1, ..., x_d)' \in \mathbb{R}^d$ with $x_i = \Phi^{-1}(u_i)$, $i = 1, ..., d$.

Another widely used elliptical copula is the multivariate t copula (cp. Demarta and McNeil (2005)) which is derived from the multivariate t distribution with constant $c_d = (\pi d)^{-d/2} \Gamma(\frac{\nu+d}{2}) / \Gamma(\frac{\nu}{2})$ and $g(s) = (1 + \frac{s}{\nu})^{-(\nu+d)/2} \; \forall s > 0$. It is hence defined as

$$C(\boldsymbol{u}) = t_{R,\nu} \left(t_\nu^{-1}(u_1), ..., t_\nu^{-1}(u_d) \right), \tag{2.9}$$

where $t_{R,\nu}$ denotes the cdf of the multivariate standard t distribution with correlation matrix $R \in [-1,1]^d$ and $\nu > 0$ degrees of freedom, while t_ν^{-1} is the inverse of the cdf t_ν of the univariate standard t distribution with ν degrees of freedom. The density of the bivariate t copula can be found in Section 2.1.3.

2.1.2 Archimedean copulas

Archimedean copulas are particularly easy to construct and have many convenient properties such as symmetry and associativity (see Nelsen (2006)).

Theorem 2.8 (Archimedean copula.) *Let $\varphi : [0,1] \to [0,\infty]$ be a continuous strictly decreasing function such that $\varphi(0) = \infty$ and $\varphi(1) = 0$ and let φ^{-1} denote the inverse of φ such that it is completely monotonic[1]. Then*

$$C(\boldsymbol{u}) = \varphi^{-1}(\varphi(u_1) + ... + \varphi(u_d))$$

is a copula.

The copula C in Theorem 2.8 is called a d-dimensional Archimedean copula with *generator* φ. In the bivariate case, the assumptions of complete monotonicity and $\varphi(0) = \infty$ are not necessary, when the *pseudo-inverse* $\varphi^{[-1]}$ of a convex generator φ is considered instead of φ^{-1}. The pseudo-inverse is defined as

$$\varphi^{[-1]}(t) = \begin{cases} \varphi^{-1}(t), & 0 \leq t \leq \varphi(0), \\ 0, & \varphi(0) \leq t \leq \infty. \end{cases}$$

If however $\varphi(0) = \infty$, φ is called *strict* and $\varphi^{[-1]} = \varphi^{-1}$. E.g., $W = W^2$ with $\varphi(t) = 1 - t$, $t \in [0,1]$ is Archimedean and $\Pi = \Pi^2$ with $\varphi(t) = -\log t$ is strictly Archimedean, while $M = M^2$ is not an Archimedean copula.

Theorem 2.8 yields an easy method to construct multivariate copulas of arbitrary dimension. However, since most commonly used generators ϕ depend on one or at most two parameters (cp. Section 2.1.3), there are only one or two parameters to model the dependency of d random variables, which is quite restrictive. Elliptical copulas such as the multivariate Gaussian and t have correlation parameters for each pair of variables, i.e., $d(d-1)/2$ correlation parameters. Hence, we will concentrate on elliptical copulas for dimensions $d > 2$, while Archimedean copulas are very appealing in the bivariate case (see the following Section 2.1.3).

---

[1] A function $g(t)$ is *completely monotonic* on an interval J if it is continuous there and if it satisfies $(-1)^k \frac{d^k}{dt^k} g(t) \geq 0$ for all t in the interior of J and $k \in \mathbb{N}_0$.

2.1.3 Bivariate copula families

Here, we describe various bivariate copula families which will be important as building blocks of our models in the following (see Section 2.1.4) and which we will concentrate on in our simulation studies (Section 5.4.4, Chapter 10) and applications (Chapter 11). Extensive overviews of bivariate copulas, also called *pair copulas*, can be found, e.g., in Table 4.1 of Nelsen (2006) and in Chapter 5 of Joe (1997). References to both sources are given in brackets, where the first number refers to Nelsen (2006) and the second to Joe (1997). Further properties such as Kendall's τ's and tail dependence parameters (see Section 2.2) will be given in Table 2.1, scatter and contour plots for standard normal margins in Appendix A.

Gaussian copula [2.3.6/B1]

According to the multivariate version defined in (2.7), the bivariate Gaussian copula with correlation parameter $\rho \in (-1, 1)$ is defined as

$$C(u_1, u_2) = \Phi_\rho \left(\Phi^{-1}(u_1), \Phi^{-1}(u_2) \right),$$

where Φ_ρ denotes the bivariate standard normal cdf with correlation ρ. The corresponding density is then given by

$$c(u_1, u_2) = \frac{1}{\sqrt{1 - \rho^2}} \exp \left[-\frac{\rho^2(x_1^2 + x_2^2) - 2\rho x_1 x_2}{2(1 - \rho^2)} \right],$$

where $x_1 = \Phi^{-1}(u_1)$ and $x_2 = \Phi^{-1}(u_2)$.

As special case of the multivariate Gaussian copula, the bivariate Gaussian copula of course also belongs to the class of elliptical copulas (see Section 2.1.1). It is *reflection symmetric*, i.e., if a random vector (U_1, U_2) follows a bivariate Gaussian copula, then $(1 - U_1, 1 - U_2)$ is distributed as the same bivariate Gaussian copula. Moreover, $C = W$ for $\rho \to -1$, $C = \Pi$ for $\rho = 0$ and $C = M$ for $\rho \to 1$.

t copula

The bivariate t copula (cp. (2.9)) is a two-parametric elliptical copula with copula distribution function
$$C(u_1, u_2) = t_{\rho, \nu} \left(t_\nu^{-1}(u_1), t_\nu^{-1}(u_2) \right),$$

where $t_{\rho, \nu}$ denotes the cdf of the bivariate standard t distribution with correlation parameter $\rho \in (-1, 1)$ and $\nu > 0$ degrees of freedom. This yields the following density:

$$c(u_1, u_2) = \frac{\Gamma(\frac{\nu+2}{2})/\Gamma(\frac{\nu}{2})}{\nu \pi dt_\nu(x_1) dt_\nu(x_2) \sqrt{1 - \rho^2}} \left[1 + \frac{x_1^2 + x_2^2 - 2\rho x_1 x_2}{\nu(1 - \rho^2)} \right]^{-\frac{\nu+2}{2}},$$

where $x_i = t_\nu^{-1}(u_i)$, $i = 1, 2$, and dt_ν is the probability density of the univariate standard t distribution with ν degrees of freedom.

Like the bivariate Gaussian copula, it is a special case of the corresponding general multivariate copula (2.9), it is reflection symmetric and its limiting cases are $C = W$ for

$\rho \to -1$, $C = \Pi$ for $\rho = 0$ and $C = M$ for $\rho \to 1$. As the t distribution tends to the normal distribution when the degrees of freedom increase, the t copula also tends to the Gaussian copula with increasing degrees of freedom, i.e., $C_\nu^t \to C^N$ for $\nu \to \infty$, where C_ν^t denotes a t copula with ν degrees of freedom and C^N a Gaussian copula with the same correlation parameter $\rho \in (-1, 1)$.

Clayton copula [4.2.1/B4]

The Clayton copula (or "Kimeldorf and Sampson copula" as in Joe (1997)) is an Archimedean copula (see Section 2.1.2) with generator $\varphi(t) = \frac{1}{\theta}(t^{-\theta} - 1)$ and therefore given by

$$C(u_1, u_2) = \left(u_1^{-\theta} + u_2^{-\theta} - 1\right)^{-\frac{1}{\theta}} \tag{2.10}$$

for $\theta > 0$. The extension to parameters $\theta \in [-1, 0)$ and negative dependence is not considered here, since the generator is not *strict* in this particular case. The density of the Clayton copula can be obtained as

$$c(u_1, u_2) = (1 + \theta)(u_1 u_2)^{-1-\theta} \left(u_1^{-\theta} + u_2^{-\theta} - 1\right)^{-\frac{1}{\theta} - 2}. \tag{2.11}$$

The limiting cases are $C = \Pi$ for $\theta \to 0$ and $C = M$ for $\theta \to \infty$.

Rotated Clayton copula

Since the Clayton copula as defined in (2.10) can only capture positive dependence (cp. Table 2.1), we also consider a rotated version, where "rotated" refers to a rotation of $90°$ in contrast to $180°$ which is often considered in the literature. Hence, we say that a random vector $(U_1, U_2) \in [0, 1]^2$ is distributed as a rotated Clayton copula with parameter $\theta < 0$ iff $(U_1, 1 - U_2)$ follows a Clayton copula with parameter $-\theta$. Then the copula distribution function is readily derived as

$$C(u_1, u_2) = u_1 - C_{(-\theta)}^C(u_1, 1 - u_2),$$

(cp. Theorem 2.4.4 in Nelsen (2006)) and the density as

$$c(u_1, u_2) = c_{(-\theta)}^C(u_1, 1 - u_2),$$

where $C_{(-\theta)}^C$ and $c_{(-\theta)}^C$ denote the Clayton copula distribution and density functions with parameter $-\theta$ as given in (2.10) and (2.11), respectively. Corresponding to the Clayton copula but with negative dependence, the limiting cases are $C = \Pi$ for $\theta \to 0$ and $C = W$ for $\theta \to -\infty$.

Gumbel copula [4.2.4/B6]

The Gumbel copula is also an Archimedean copula. Its generator is $\varphi(t) = (-\log t)^\theta$ which yields

$$C(u_1, u_2) = \exp\left[-\left((-\log u_1)^\theta + (-\log u_2)^\theta\right)^{\frac{1}{\theta}}\right], \tag{2.12}$$

where $\theta \geq 1$. The corresponding density is given by

$$c(u_1, u_2) = \frac{C(u_1, u_2)}{u_1 u_2} \frac{(\log u_1 \cdot \log u_2)^{\theta-1}}{((-\log u_1)^{\theta} + (-\log u_2)^{\theta})^{2-\frac{1}{\theta}}}$$
$$\times \left[((-\log u_1)^{\theta} + (-\log u_2)^{\theta})^{\frac{1}{\theta}} + \theta - 1 \right]. \tag{2.13}$$

Similar to the Clayton copula, the limiting cases are $C = \Pi$ for $\theta = 1$ and $C = M$ for $\theta \to \infty$.

Rotated Gumbel copula

As for the Clayton copula, we also consider the rotated Gumbel copula, where a random vector $(U_1, U_2) \in [0, 1]^2$ is distributed as a rotated Gumbel copula with parameter $\theta \leq -1$ iff $(U_1, 1 - U_2)$ follows a Gumbel copula with parameter $-\theta$. Denoting the Gumbel copula distribution and density functions with parameter $-\theta$ as given in (2.12) and (2.13) by $C_{(-\theta)}^G$ and $c_{(-\theta)}^G$, respectively, this yields the following distribution function:

$$C(u_1, u_2) = u_1 - C_{(-\theta)}^G(u_1, 1 - u_2),$$

and density:

$$c(u_1, u_2) = c_{(-\theta)}^G(u_1, 1 - u_2).$$

The independence copula Π is obtained for $\theta = -1$, while $C = W$ for $\theta \to -\infty$ as for the rotated Clayton copula.

Frank copula [4.2.5/B3]

An example of an Archimedean copula for negative and positive dependence is the Frank copula with generator $\varphi(t) = -\log\left[\frac{e^{-\theta t}-1}{e^{-\theta}-1}\right]$ and corresponding distribution function

$$C(u_1, u_2) = -\frac{1}{\theta} \log\left[1 + \frac{(e^{-\theta u_1} - 1)(e^{-\theta u_2} - 1)}{e^{-\theta} - 1}\right],$$

where $\theta \in \mathbb{R} \setminus \{0\}$. The density is

$$c(u_1, u_2) = \theta(e^{-\theta} - 1)\frac{e^{-\theta(u_1+u_2)}}{\left[e^{-\theta} - 1 + (e^{-\theta u_1} - 1)(e^{-\theta u_2} - 1)\right]^2}.$$

The limiting cases are similar to the Gaussian and t copulas: $C = W$ for $\theta \to -\infty$, $C = \Pi$ for $\theta \to 0$ and $C = M$ for $\theta \to \infty$. Furthermore, it is also reflection symmetric.

Joe copula [4.2.6/B5]

The Joe copula is another Archimedean copula with generator $\varphi(t) = -\log\left[1 - (1-t)^{\theta}\right]$ and therefore given by

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

with $\theta > 1$ and the following density:

$$c(u_1, u_2) = \left[(1 - u_1)^\theta + (1 - u_2)^\theta - (1 - u_1)^\theta (1 - u_2)^\theta \right]^{\frac{1}{\theta} - 2} (1 - u_1)^{\theta - 1} (1 - u_2)^{\theta - 1}$$
$$\times \left[\theta - 1 + (1 - u_1)^\theta + (1 - u_2)^\theta - (1 - u_1)^\theta (1 - u_2)^\theta \right].$$

As for the Gumbel copula, $C = \Pi$ for $\theta \to 1$ and $C = M$ for $\theta \to \infty$.

Clayton-Gumbel copula [4.5.3/BB1]

The Clayton-Gumbel copula is a two-parametric Archimedean copula and can be regarded as a generalization of the one-parametric Clayton and Gumbel families. To avoid confusion, it will simply be called BB1 copula in the following. Its generator is $\varphi(t) = \left(t^{-\theta} - 1 \right)^\delta$ which yields

$$C(u_1, u_2) = \left[1 + \left[(u_1^{-\theta} - 1)^\delta + (u_2^{-\theta} - 1)^\delta \right]^{\frac{1}{\delta}} \right]^{-\frac{1}{\theta}},$$

where $\theta > 0$ and $\delta \geq 1$. The corresponding copula density and other properties can be found in Schepsmeier (2010). Similar to Clayton and Gumbel copulas, we obtain the independence copula Π for $\theta \to 0$ and $\delta = 1$ as well as $C = W$ for $\theta \to \infty$ and $\delta \to \infty$. More interestingly, the Clayton copula is a subfamily when $\delta = 1$ and the Gumbel copula is the limiting case of $\theta \to 0$.

Joe-Clayton copula [-/BB7]

Similar to the BB1 copula, the Joe-Clayton copula is a two-parametric generalization of the corresponding one-parametric copula families. In the following, we will also simply refer to it as BB7 copula. The generator of this Archimedean copula is defined as $\varphi(t) = \left[1 - (1 - t)^\theta \right]^{-\delta} - 1$, giving the copula distribution function

$$C(u_1, u_2) = 1 - \left[1 - \left[(1 - (1 - u_1)^\theta)^{-\delta} + (1 - (1 - u_2)^\theta)^{-\delta} \right]^{-\frac{1}{\delta}} \right]^{\frac{1}{\theta}}$$

for $\theta \geq 1$ and $\delta > 0$. As above, the density can be found in Schepsmeier (2010). The limiting cases are $C = \Pi$ for $\theta = 1$ and $\delta \to 0$ as well as $C = W$ for $\theta \to \infty$ and $\delta \to \infty$. Furthermore, the Clayton copula is again a subfamily for $\theta = 1$, while the Joe copula is obtained for $\delta \to 0$.

2.1.4 Pair copula constructions

The following discussion of pair copula constructions is based on Aas et al. (2009) who use a cascade of bivariate copulas to model multivariate data. In contrast to the multivariate copulas we encountered so far (elliptical and Archimedean, which both impose some constraints on the respective models), such constructions are a simple and flexible way to specify multivariate dependence models.

The idea of a *pair copula construction* can be best explained in a small-dimensional example.

Example 1 (Pair copula construction in 3 dimensions.) *Consider the three-dimensional random vector $\mathbf{X} = (X_1, X_2, X_3)$ with joint density function f and univariate densities f_1, f_2 and f_3. Using the definition of conditional densities we know that*

$$f(x_1, x_2, x_3) = f_3(x_3)f(x_2|x_3)f(x_1|x_2, x_3). \tag{2.14}$$

From Sklar's Theorem 2.4 and Equation (2.5), we further know that

$$f(x_1, x_2, x_3) = c_{123}(F_1(x_1), F_2(x_2), F_3(x_3))f_1(x_1)f_2(x_2)f_3(x_3), \tag{2.15}$$

where c_{123} is the density of a three-dimensional copula. In the bivariate case, this yields

$$f(x_2, x_3) = c_{23}(F_2(x_2), F_3(x_3))f_2(x_2)f_3(x_3)$$

for a bivariate copula density c_{23}. Hence, it follows that

$$f(x_2|x_3) = \frac{f(x_2, x_3)}{f_3(x_3)} = c_{23}(F_2(x_2), F_3(x_3))f_2(x_2). \tag{2.16}$$

Similarly, we can decompose

$$f(x_1|x_2, x_3) = \frac{f(x_1, x_3|x_2)}{f(x_3|x_2)} = c_{13|2}(F(x_1|x_2), F(x_3|x_2))f(x_1|x_2), \tag{2.17}$$

where $c_{12|3}$ is an appropriate pair copula for the transformed variables $F(x_1|x_2)$ and $F(x_3|x_2)$. Decomposing $f(x_1|x_2)$ as in (2.16) gives

$$f(x_1|x_2, x_3) = c_{13|2}(F(x_1|x_2), F(x_3|x_2))c_{12}(F_1(x_1), F_2(x_2))f_1(x_1).$$

Combining all decompositions and plugging them into Equation (2.14) yields

$$f(x_1, x_2, x_3) = c_{12}(F_1(x_1), F_2(x_2))c_{23}(F_2(x_2), F_3(x_3))c_{13|2}(F(x_1|x_2), F(x_3|x_2))$$
$$\times f_1(x_1)f_2(x_2)f_3(x_3).$$

Getting back to Equation (2.15), we see that we have constructed a trivariate copula density using only bivariate copulas as building blocks:

$$c_{123}(F_1(x_1), F_2(x_2), F_3(x_3)) = c_{12}(F_1(x_1), F_2(x_2))c_{23}(F_2(x_2), F_3(x_3))$$
$$\times c_{13|2}(F(x_1|x_2), F(x_3|x_2)).$$

Note however that this decomposition is not unique. In Equation (2.14) the variables can be permuted in $3! = 6$ ways. Appropriate pair copulas can be found, e.g., in Section 2.1.3.

Moreover, note that we assume that the pair copula $c_{13|2}$ in Equation (2.17) is independent of the conditioning variable X_2, i.e.,

$$c_{13|2}(F(x_1|x_2), F(x_3|x_2); x_2) \equiv c_{13|2}(F(x_1|x_2), F(x_3|x_2)).$$

This assumption is necessary in order to construct flexible models. Hobæk Haff et al. (2010) call this the simplified pair copula construction (which is contrary to our notation of simplification (see Section 7.1)) and show that it is a good approximation to the correct decomposition.

In general, we can decompose a d-dimensional random vector $\mathbf{X} = (X_1, ..., X_d)$ with joint density f into

$$f(\boldsymbol{x}) = f_d(x_d)f(x_{d-1}|x_d)f(x_{d-2}|x_{d-1}, x_d)...f(x_1|x_2, ..., x_d). \qquad (2.18)$$

Iterating the arguments in Example 1, we can decompose each term in (2.18) into marginal densities and adequate bivariate copulas using the general formula

$$f(x|\boldsymbol{v}) = c_{xv_j|\boldsymbol{v}_{-j}}(F(x|\boldsymbol{v}_{-j}), F(v_j|\boldsymbol{v}_{-j}))f(x|\boldsymbol{v}_{-j}),$$

where $\boldsymbol{v}$ is an m-dimensional vector, v_j is an arbitrary component of $\boldsymbol{v}$ and $\boldsymbol{v}_{-j}$ denotes the $(m-1)$-dimensional vector $\boldsymbol{v}$ excluding v_j. Hence, under appropriate regularity conditions this yields a multivariate copula density expressed as the product of bivariate copulas.

The pair copulas are applied to transformed variables, which are marginal conditional distributions of the form $F(x|\boldsymbol{v})$. According to Joe (1996) these can be obtained for every j as

$$F(x|\boldsymbol{v}) = \frac{\partial C_{xv_j|\boldsymbol{v}_{-j}}(F(x|\boldsymbol{v}_{-j}), F(v_j|\boldsymbol{v}_{-j}))}{\partial F(v_j|\boldsymbol{v}_{-j})}, \qquad (2.19)$$

where $C_{xv_j|\boldsymbol{v}_{-j}}$ is a bivariate copula distribution function.

We have seen that there is no unique pair copula construction of a d-dimensional random vector. Moreover, the number of possible decompositions increases significantly with increasing dimension d: there are, e.g., already 240 different constructions for a five-dimensional density (Aas et al. 2009)! Hence, we need a way to describe such models appropriately. This will be established using so-called *regular vines* in Section 2.4. But prior to that, we will discuss some more basic concepts such as dependence measures and graph theory which will be needed for this classification of pair copula constructions.

2.2 Dependence measures

As we have seen in the previous section, *bivariate* dependence is even fundamental for explaining dependence among large numbers of variables. Thus, the question arises how this dependence can be measured appropriately.

Classically, *Pearson's product-moment correlation coefficient* between two random variables X_1 and X_2, which is given by

$$corr(X_1, X_2) = \frac{\mathrm{Cov}(X_1, X_2)}{\sqrt{\mathrm{Var}(X_1)}\sqrt{\mathrm{Var}(X_2)}}, \qquad (2.20)$$

has been widely used in statistics. However, *corr* is a measure of linear dependence only and not invariant under non-linear strictly increasing transformations. Furthermore, it is not defined for distributions with non-finite second moments (e.g., the Cauchy distribution) and the possible values of (2.20) depend on the marginal distributions of X_1 and X_2 (Kurowicka and Cooke 2006). Hence, we will concentrate on so-called "measures of association" which do not exhibit these disadvantages. As before, we mainly follow Nelsen (2006), where the proofs of the following theorems can be found.

2.2.1 Measures of association

While "correlation" usually refers to linear dependence, "association" refers to any possible type of dependence between two random variables X_1 and X_2. The idea is therefore to investigate, whether "large" values of one variable are "associated" with "large" values of the other and similarly for "small" values. This naturally leads to the consideration of *concordance*: according to Nelsen (2006), two pairs of observations (x_{i1}, x_{i2}) and (x_{j1}, x_{j2}) from the continuous random vector (X_1, X_2) are called *concordant* if $x_{i1} < x_{j1}$ and $x_{i2} < x_{j2}$, or if $x_{i1} > x_{j1}$ and $x_{i2} > x_{j2}$, or equivalently if $(x_{i1} - x_{j1})(x_{i2} - x_{j2}) > 0$. Similarly, the pairs are *discordant* if $(x_{i1} - x_{j1})(x_{i2} - x_{j2}) < 0$. The case $(x_{i1} - x_{j1})(x_{i2} - x_{j2}) = 0$ cannot occur, when X_1 and X_2 are continuous.

Based on these notations, *Kendall's τ* is defined as the probability of concordance minus the probability of discordance of two random variables X_1 and X_2, i.e.,

$$\tau(X_1, X_2) = P((X_{11} - X_{21})(X_{12} - X_{22}) > 0) - P((X_{11} - X_{21})(X_{12} - X_{22}) < 0), \quad (2.21)$$

where (X_{11}, X_{12}) and (X_{21}, X_{22}) are independent and identically distributed copies of (X_1, X_2).

An important relationship between concordance and copulas is given in the following theorem:

Theorem 2.9 *Let X_1 and X_2 be continuous random variables with copula C. Then*

$$\tau(X_1, X_2) = 4 \int_{[0,1]^2} C(u_1, u_2) dC(u_1, u_2) - 1.$$

In particular for Archimedean copulas, this yields a convenient expression of Kendall's τ in terms of the generator φ:

$$\tau(X_1, X_2) = 1 + 4 \int_0^1 \frac{\varphi(t)}{\varphi'(t)} dt.$$

An alternative measure of association is given by *Spearman's ρ*. Empirically it is defined as the correlation of the pairs of ranks (cp. (3.1)). The population version is however also defined in terms of concordance. For this let (X_{11}, X_{12}), (X_{21}, X_{22}) and (X_{31}, X_{32}) independent and identically distributed copies of (X_1, X_2). Then Spearman's ρ is defined to be proportional to the probability of concordance minus the probability of discordance of the two vectors (X_{11}, X_{12}) and (X_{21}, X_{32}), i.e.,

$$\rho(X_1, X_2) = 3 \left[P((X_{11} - X_{21})(X_{12} - X_{32}) > 0) - P((X_{11} - X_{21})(X_{12} - X_{32}) < 0) \right]. \tag{2.22}$$

Note that the copula of (X_{11}, X_{12}) is C, while (X_{21}, X_{32}) are independent and thus their copula is Π. As a result the corresponding version of Theorem 2.9 for Spearman's ρ is stated as follows:

Theorem 2.10 *Let X_1 and X_2 be continuous random variables with copula C. Then*

$$\rho(X_1, X_2) = 12 \int_{[0,1]^2} C(u_1, u_2) du_1 du_2 - 3 = 12 \int_{[0,1]^2} u_1 u_2 dC(u_1, u_2) - 3.$$

Since both Kendall's τ and Spearman's ρ can be stated in terms of the copula of two random variables (Theorems 2.9 and 2.10), it follows from Theorem 2.6 that both measures are invariant under strictly increasing transformations and independent of the marginal distributions of X_1 and X_2 – in contrast to Pearson's product-moment correlation coefficient as noted above. Moreover, they are always well defined and it can be shown that $\tau(X_1, X_2), \rho(X_1, X_2) = 1$ if X_2 is almost surely an increasing function of X_1, and $\tau(X_1, X_2), \rho(X_1, X_2) = -1$ if X_2 is almost surely an decreasing function of X_1 (cp. Theorem 2.5).

2.2.2 Tail dependence

While Kendall's τ and Spearman's ρ measure dependence on the whole space $[0, 1]^2$, tail dependence takes an alternative approach. We are again interested in describing whether "large" ("small") values of one variable appear with "large" ("small") values of the other, but now concentrate on the lower-left and upper-right quadrants of $[0, 1]^2$. Hence we define the *lower tail dependence parameter* λ^{lower} of random variables X_1 and X_2 as

$$\lambda^{lower} = \lim_{t \to 0^+} P(X_2 \le F_2^{-1}(t)|X_1 \le F_1^{-1}(t)), \tag{2.23}$$

and similarly the *upper tail dependence parameter* λ^{upper} as

$$\lambda^{upper} = \lim_{t \to 1^-} P(X_2 > F_2^{-1}(t)|X_1 > F_1^{-1}(t)), \tag{2.24}$$

if the limits exist. The following theorem shows that the tail dependence parameters can also be stated in terms of the copula C of X_1 and X_2.

Theorem 2.11 *Let X_1 and X_2 be continuous random variables with copula C. If the limits (2.23) and (2.24) exit, then*

$$\lambda^{lower} = \lim_{t \to 0^+} \frac{C(t,t)}{t} \tag{2.25}$$

and

$$\lambda^{upper} = \lim_{t \to 1^-} \frac{\bar{C}(t,t)}{1 - t} = \lim_{t \to 1^-} \frac{1 - 2t + C(t,t)}{1 - t}, \tag{2.26}$$

where $\bar{C}$ is the joint survival function (2.1).

E.g., the lower tail dependence parameter of the independence copula Π is easily obtained as

$$\lambda^{lower} = \lim_{t \to 0^+} \frac{\Pi(t,t)}{t} = \lim_{t \to 0^+} \frac{t^2}{t} = \lim_{t \to 0^+} t = 0,$$

while the upper Fréchet-Hoeffding bound M (cp. Theorem 2.3) exhibits perfect lower tail dependence:

$$\lambda^{lower} = \lim_{t \to 0^+} \frac{M(t,t)}{t} = \lim_{t \to 0^+} \frac{t}{t} = \lim_{t \to 0^+} 1 = 1.$$

The tail dependence parameters of the bivariate copula families discussed in Section 2.1.3 except for the rotated Clayton and Gumbel copulas can be found in the literature, but those of the latter families have to be calculated.

(i) Lower tail dependence parameter of the rotated Clayton copula:

$$\lambda^{lower} = \lim_{t \to 0^+} \frac{C(t,t)}{t} = \lim_{t \to 0^+} \frac{t - C^C_{(-\theta)}(t, 1-t)}{t}$$

$$= 1 - \lim_{t \to 0^+} \frac{\left(t^\theta + (1-t)^\theta - 1\right)^{\frac{1}{\theta}}}{t} = 1 - \lim_{t \to 0^+} \left(\frac{t^\theta + (1-t)^\theta - 1}{t^\theta}\right)^{\frac{1}{\theta}}$$

$$= 1 - \lim_{t \to 0^+} \left(1 + \left(\frac{1-t}{t}\right)^\theta - t^{-\theta}\right)^{\frac{1}{\theta}} \overset{\theta \leq 0}{=} 1 - 1 = 0,$$

where $C^C_{(-\theta)}$ denotes the Clayton copula distribution function with parameter $-\theta$ as given in (2.10).

(ii) Upper tail dependence parameter of the rotated Clayton copula:

$$\lambda^{upper} = \lim_{t \to 1^-} \frac{1 - 2t + C(t,t)}{1-t} = \lim_{t \to 1^-} \frac{1 - 2t + t - C^C_{(-\theta)}(t, 1-t)}{1-t}$$

$$= 1 - \lim_{t \to 1^-} \frac{\left(t^\theta + (1-t)^\theta - 1\right)^{\frac{1}{\theta}}}{1-t} = 1 - \lim_{t \to 1^-} \left(\frac{t^\theta + (1-t)^\theta - 1}{(1-t)^\theta}\right)^{\frac{1}{\theta}}$$

$$= 1 - \lim_{t \to 0^+} \left(\left(\frac{t}{1-t}\right)^\theta + 1 - (1-t)^{-\theta}\right)^{\frac{1}{\theta}} \overset{\theta \leq 0}{=} 1 - 1 = 0.$$

(iii) Lower tail dependence parameter of the rotated Gumbel copula:

$$\lambda^{lower} = \lim_{t \to 0^+} \frac{C(t,t)}{t} = \lim_{t \to 0^+} \frac{t - C^G_{(-\theta)}(t, 1-t)}{t}$$

$$= 1 - \lim_{t \to 0^+} \frac{\exp\left[-\left((-\log t)^{-\theta} + (-\log(1-t))^{-\theta}\right)^{-\frac{1}{\theta}}\right]}{t}$$

$$= 1 - \lim_{t \to 0^+} \exp\underbrace{\left[-\left((-\log t)^{-\theta} + (-\log(1-t))^{-\theta}\right)^{-\frac{1}{\theta}} - \log(t)\right]}_{\longrightarrow 0 \text{ as } t \to 0^+}$$

$$= 1 - 1 = 0,$$

where $C^G_{(-\theta)}$ denotes the Gumbel copula distribution function with parameter $-\theta$ as given in (2.12).

(iv) Upper tail dependence parameter of the rotated Gumbel copula: similarly to (iii), it holds that $\lambda^{upper} = 0$.

Kendall's τ's and tail dependence parameters of the bivariate copula families discussed in Section 2.1.3 are now shown in Table 2.1. By inversion, copula parameters can be expressed in terms of Kendall's τ and/or tail dependence parameters (cp. Section 4.1.2).

Copula family	Kendall's τ	Lower tail dependence	Upper tail dependence
Gaussian	$\frac{2}{\pi}\arcsin(\rho)$	0	0
t	$\frac{2}{\pi}\arcsin(\rho)$	$2t_{\nu+1}\left(-\sqrt{\nu+1}\sqrt{\frac{1-\rho}{1+\rho}}\right)$	
Clayton	$\frac{\theta}{\theta+1}$	$2^{-1/\theta}$	0
Rotated Clayton	$-\frac{\theta}{\theta+1}$	0	0
Gumbel	$1-\frac{1}{\theta}$	0	$2-2^{1/\theta}$
Rotated Gumbel	$\frac{1}{\theta}-1$	0	0
Frank[a]	$1-\frac{4}{\theta}+4\frac{D_1(\theta)}{\theta}$	0	0
Joe[b]	$1+\left[\frac{-2+2\gamma+2\log 2+\Psi(\frac{1}{\theta})+\Psi(\frac{1}{2}\frac{2+\theta}{\theta})+\theta}{\theta-2}\right]$	0	$2-2^{1/\theta}$
BB1	$1-\frac{2}{\delta(\theta+2)}$	$2^{-1/(\theta\delta)}$	$2-2^{1/\delta}$
BB7[c]	$1-\frac{2}{\delta(2-\theta)}+\frac{4}{\theta^2\delta}B\left(\frac{2-2\theta}{\theta}+1,\delta+2\right)$	$2^{-1/\delta}$	$2-2^{1/\theta}$

Table 2.1: Kendall's τ's and tail dependence parameters of the bivariate copula families discussed in Section 2.1.3.

[a] $D_1(\theta)=\int_0^\theta \frac{c/\theta}{\exp(x)-1}dx$ (Debey function)
[b] $\gamma=\lim_{n\to\infty}(\sum_{i=1}^n \frac{1}{i}-\log n)\approx 0.57721$ (Euler's constant), $\Psi(x)=\frac{d}{dx}\log(\Gamma(x))$ (Digamma function)
[c] $B(x,y)=\int_0^1 t^{x+1}(t-1)^{y-1}dt$ (Beta function)

2.3   Graph theory

This section gives a brief introduction to graph theory which will be needed in Section 2.4 to classify pair copula constructions. The exposition here mainly follows Diestel (2006) and is meant as a reference for the following parts of the thesis. Obviously, we begin our collection of definitions and theorems with the description of a *graph*. Examples for the notations presented here are shown in Figures 2.1 and 2.2.

Definition 2.12 (Graph, node, edge, degree.) *A graph is a pair $G = (N, E)$ of sets such that $E \subseteq \{\{x, y\} : x, y \in N\}$. The elements of E are called edges of the graph G, the elements of N are its nodes. The number of neighbors of a node $v \in N$ is the degree of v, denoted by $d(v)$.*

The graph defined above is usually referred to as *undirected*, since the order of nodes corresponding to an edge is arbitrary. In a *directed* graph, it holds that $E \subseteq \{(x, y) : x, y \in N\}$. If there is a function $w : E \to \mathbb{R}$, then G is called *weighted* and denoted by $G = (N, E, w)$, i.e., weights are assigned to each edge. Moreover, if $E = \{\{x, y\} : x, y \in N\}$ in Definition 2.12, then G is called *complete*.

A *subgraph* of a graph $G = (N, E)$ is a graph $G' = (N', E')$ with $N' \subseteq N$ and $E' \subseteq E$. Important examples of graphs are *paths* and *cycles* which often occur as subgraphs of interest.

Definition 2.13 (Path, cycle.) *A path is a graph $P = (N, E)$ with $N = \{v_0, v_1, ..., v_k\}$ and $E = \{\{v_0, v_1\}, \{v_1, v_2\}, ..., \{v_{k-1}, v_k\}\}$. A cycle is a path with $v_0 = v_k$.*

A graph G is called *connected* if any two of its nodes are linked by a path in G. Furthermore, a path in G containing every node of G is called a *Hamiltonian path*, such a cycle is a *Hamiltonian cycle*.

The most important class of graphs that will be considered in the following are *trees*, which are connected and do not contain cycles. They can be characterized by the following theorem, where $G \pm e$ denotes a graph with removed/additional edge e.

Theorem 2.14 (Characterization of trees.) *The following statements are equivalent for a graph $T = (N, E)$:*

(i) T is a tree.

(ii) Any two nodes of T are linked by a unique path in T.

(iii) T is minimally connected, i.e., T is connected but $T - e$ is disconnected for every edge $e \in E$.

(iv) T is maximally acyclic, i.e., T contains no cycle but $T + \{x, y\}$ does for any two non-adjacent nodes $x, y \in N$.

A *spanning tree* of a graph $G = (N, E)$ is a subgraph $T = (N_T, E_T)$, which is a tree with $N_T = N$. Moreover, a tree, which has a node v_0 with $d(v_0) = |N| - 1$, will be called a *star* and v_0 the *root node*. Obviously, in stars it holds that $d(v) = 1 \ \forall v \in N \setminus \{v_0\}$.
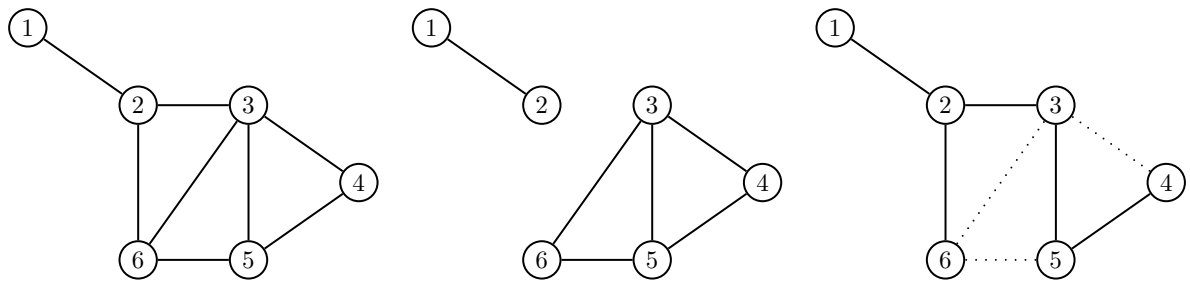
Figure 2.1: Left panel: *connected* graph $G = (N, E)$ with *nodes* $N = \{1, 2, 3, 4, 5, 6\}$ and *edges* $E = \{\{1, 2\}, \{2, 3\}, \{2, 6\}, \{3, 4\}, \{3, 5\}, \{3, 6\}, \{4, 5\}, \{5, 6\}\}$. E.g., the *degree* of node 3 is $d(3) = 4$, a *cycle* is given by $2 - 3 - 5 - 6 - 2$ and a *(Hamiltonian) path* by $1 - 2 - 3 - 4 - 5 - 6$. Middle panel: *disconnected* graph with components on nodes $N_1 = \{1, 2\}$ and $N_2 = \{3, 4, 5, 6\}$. Right panel: *spanning tree* of the graph in the left panel.
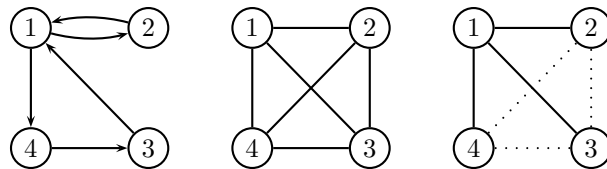
Figure 2.2: Left panel: *directed* graph. Middle panel: *complete* graph on four nodes. Right panel: *(spanning) star* with root node 1 in a complete graph.

2.4 Regular vines

As seen in Section 2.1.4, we need a way to classify different pair copula constructions. This has been established by Bedford and Cooke (2001, 2002) who introduced a graphical model called *regular vines*. Briefly, a regular vine can be described as a nested set of trees, where the edges of tree i are the nodes of tree $i+1$, and where two edges in tree i are joined by an edge in tree $i + 1$ only if they share a common node. In order to build a statistical model, edges will correspond to pair copulas which then constitute a decomposed density as discussed in Section 2.1.4.

Firstly, we formally define regular vines (Kurowicka and Cooke 2006):

Definition 2.15 (Regular vine.) $\mathcal{V}$ *is a regular vine on d elements if*

(i) $\mathcal{V} = (T_1, ..., T_{d-1})$.

(ii) $T_1 = (N_1, E_1)$ *is a tree with nodes* $N_1 = \{1, ..., d\}$. *For* $i = 2, ..., n - 1$, $T_i = (N_i, E_i)$ *is a tree with nodes* $N_i = E_{i-1}$.

(iii) *For* $i = 2, ..., n - 1$, *if* $\{a, b\} \in E_i$, *where* $a = \{a_1, a_2\}$ *and* $b = \{b_1, b_2\}$, *then exactly one of the* a_j*'s equals one of the* b_j*'s.*

Property (iii) in Definition 2.15 is usually referred to as *proximity condition*, since it expresses the fact that two nodes are adjacent in tree T_i only if the corresponding edges in tree T_{i-1} are adjacent, i.e., share a common node. An example of an R-vine on seven nodes is shown in Figure 2.3 (cp. Dißmann (2010)).

The number of possible regular vines on d nodes is however still very large ($\binom{d}{2} \times (d-2)! \times 2^{\binom{d-2}{2}}$ as shown in Morales-Nápoles et al. (2010)). Therefore, two special cases of regular vines have recently attracted special attention: *canonical vines* and *D-vines*[2] (see Aas et al. (2009)). These impose additional restrictions and hence limit the number of different models. It can be shown that there are $d!/2$ different canonical vines and D-vines on d nodes, respectively. They are described in the following definition (cp. Section 2.3 and Kurowicka and Cooke (2006)).

Definition 2.16 (Canonical vine, D-vine.) *A regular vine is called a*

(i) *canonical vine if each tree T_i, $i = 1, ..., d-1$, is a star, i.e., if each tree T_i has a unique node of degree $d-i$, the root node.*

(ii) *D-vine if T_1 is a path, i.e., if each node in T_1 has a degree of at most 2.*

Note that the first tree T_1 of a D-vine uniquely determines all higher order trees $T_2, ..., T_{d-1}$ due to the proximity condition (iii) in Definition 2.15. Examples of canonical and D-vines on 5 nodes are shown in Figures 2.4 and 2.5. In the following, regular and canonical vines will usually be denoted as *R-vines* and *C-vines*, respectively.

It was shown in Bedford and Cooke (2002) and Kurowicka and Cooke (2006) that the edges of an R-vine can be uniquely identified by two nodes, called conditioned nodes and a set of conditioning nodes. Czado (2010) proposed to identify the edges in tree T_i by $jk|D$ where $j < k$ and D is the conditioning set. Here the conditioned nodes $\{j, k\}$ are ordered to make the order of the arguments of the bivariate copulas unique which will be identified by the edges. If $D = \varnothing$, the corresponding edge is denoted by jk.

The notation of an edge e in T_i will depend on the two edges in T_{i-1}, which by the proximity condition (cp. Definition 2.15) have a common node in T_{i-1}. Denote these edges by $a = j(a), k(a)|D(a)$ and $b = j(b), k(b)|D(b)$ with $V(a) := \{j(a), k(a), D(a)\}$ and $V(b) := \{j(b), k(b), D(b)\}$, respectively. The nodes a and b in tree T_i are therefore joined by edge $e = j(e), k(e)|D(e)$, where

$$j(e) := \min\{i : i \in (V(a) \cup V(b)) \setminus D(e)\},$$
$$k(e) := \max\{i : i \in (V(a) \cup V(b)) \setminus D(e)\},$$
$$D(e) := V(a) \cap V(b).$$

Note however that this unique order of the conditioned nodes is not necessary but made out of convenience. Inference is usually based on unordered conditioned sets (see Dißmann (2010)).

Now we can build up a statistical model on an R-vine with node set $\mathcal{N} := \{N_1, ..., N_{d-1}\}$ and edge set $\mathcal{E} := \{E_1, ..., E_{d-1}\}$ by associating each edge $e = j(e), k(e)|D(e)$ in E_i

---

[2]According to Kurowicka and Cooke (2006), D-vines were originally called "drawable" vines, while the name "canonical" vines is due to the fact that sampling from such vines is most natural.
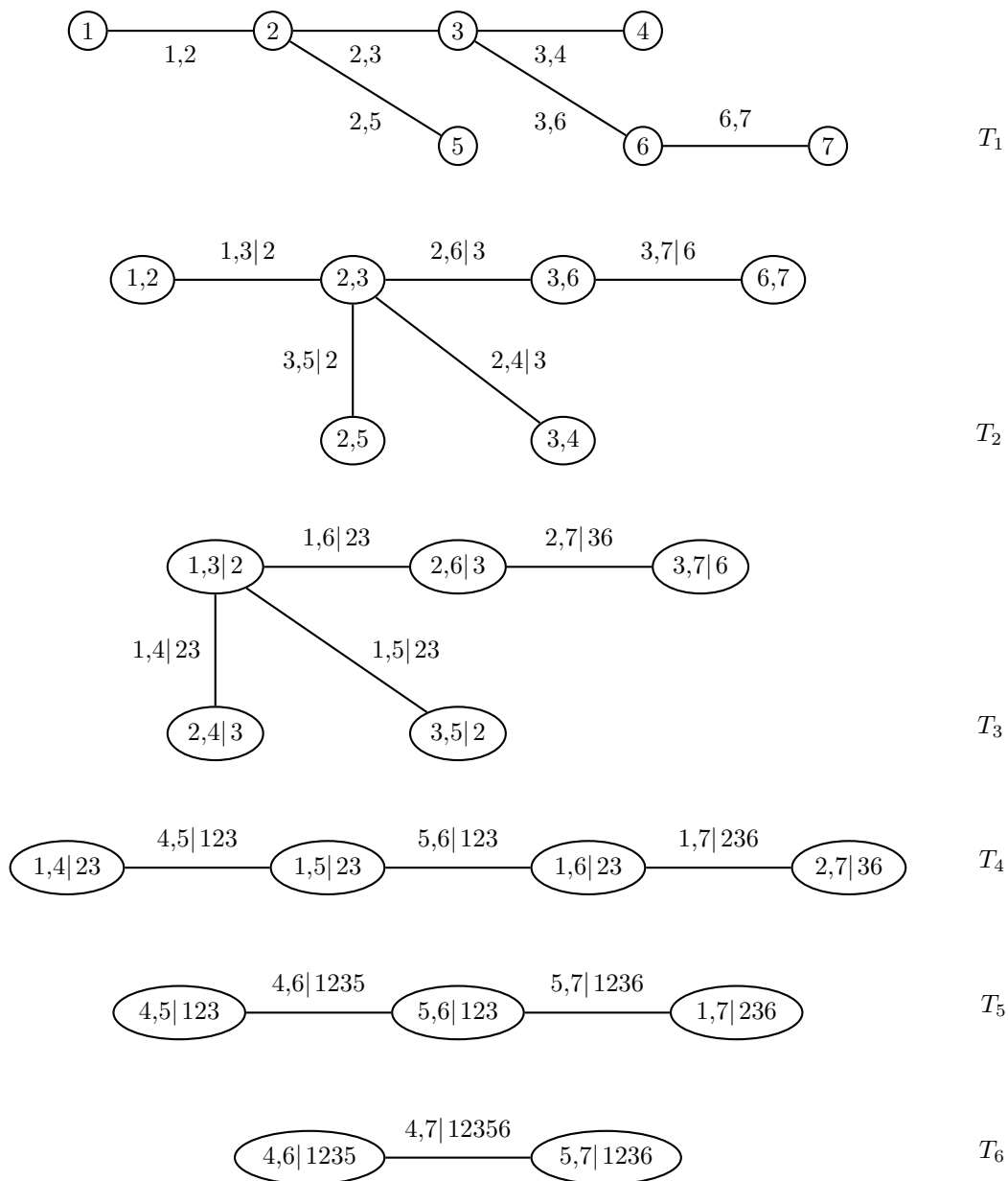
Figure 2.3: Example of a seven-dimensional R-vine.
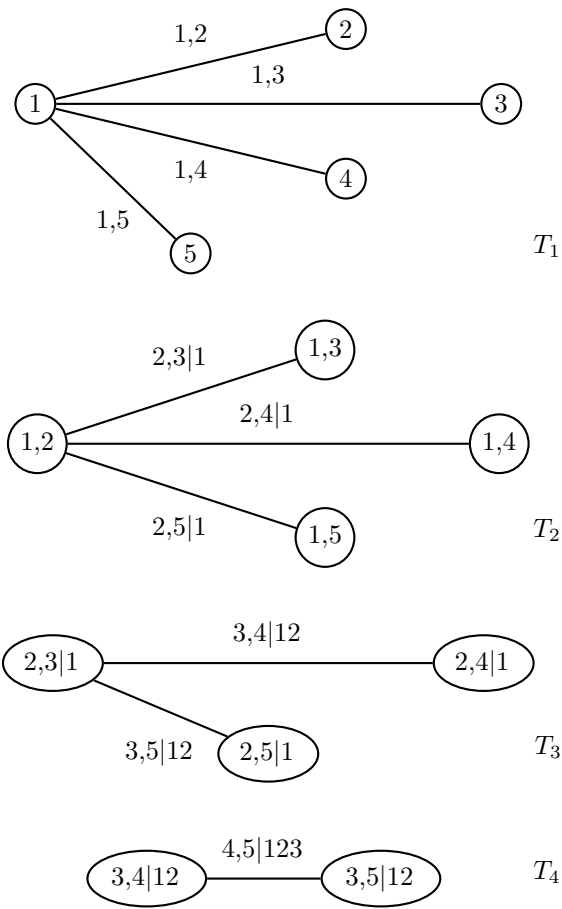
Figure 2.4: Example of a five-dimensional C-vine.
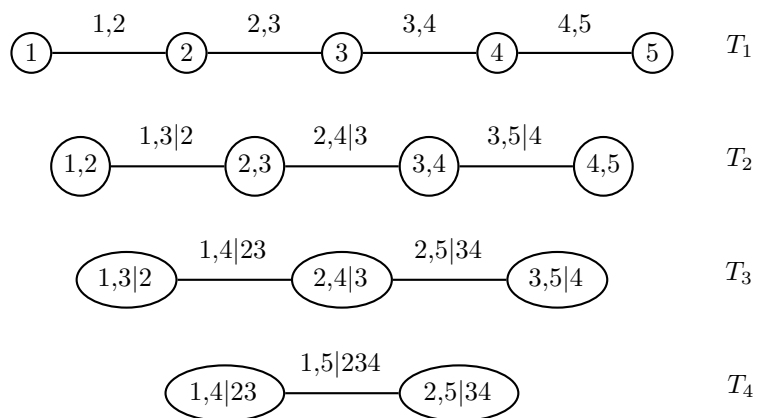
Figure 2.5: Example of a five-dimensional D-vine.

with a bivariate copula density $c_{j(e),k(e)|D(e)}$. Let $\mathbf{X}_{D(e)}$ be the sub random vector of $\mathbf{X} = (X_1, ..., X_d)$ indicated by the indices contained in $D(e)$. Then an *R-vine distribution* is defined as the distribution of the random vector $\mathbf{X}$ with marginal densities f_r, $r = 1, ..., d$, and the conditional densities of $(X_{j(e)}, X_{k(e)})$ given the variables $\mathbf{X}_{D(e)}$ specified as $c_{j(e),k(e)|D(e)}$ for the R-vine with node set $\mathcal{N}$ and edge set $\mathcal{E}$. Kurowicka and Cooke (2006) prove that the joint density of $\mathbf{X}$ is uniquely determined and given by

$$f(\boldsymbol{x}) = \prod_{r=1}^{d} f_r(x_r) \times \prod_{i=1}^{d-1} \prod_{e \in E_i} c_{j(e),k(e)|D(e)}(F(x_{j(e)}|\boldsymbol{x}_{D(e)}), F(x_{k(e)}|\boldsymbol{x}_{D(e)})), \qquad (2.27)$$

where $\boldsymbol{x}_{D(e)}$ denotes the subvector of $\boldsymbol{x}$ indicated by the indices contained in $D(e)$. The joint density in (2.27) is called an *R-vine density*.

The specific features of C- and D-vines are that C-vines have a conditioned set which only depends on the tree level, i.e., $D(e) \equiv D_i \; \forall e \in E_i$, while in D-vines the conditioning sets of edges $e = (a, b)$ are always those nodes which lie between the nodes a and b in the first tree T_1. In particular, a *C-vine density* can be written as

$$f(\boldsymbol{x}) = \prod_{r=1}^{d} f_r(x_r) \times \prod_{i=1}^{d-1} \prod_{j=1}^{d-i} c_{i,i+j|1:(i-1)} = \prod_{r=1}^{d} f_r(x_r) \times \prod_{i=1}^{d-1} \prod_{j=i+1}^{d} c_{i,j|1:(i-1)}, \qquad (2.28)$$

where $c_{jk|i_1:i_m} := c_{jk|i_1,...,i_m}(F(x_j|x_{i_1}, ..., x_{i_m}), F(x_k|x_{i_1}, ..., x_{i_m}))$. Accordingly, a *D-vine density* is obtained as

$$f(\boldsymbol{x}) = \prod_{r=1}^{d} f_r(x_r) \times \prod_{i=1}^{d-1} \prod_{j=1}^{d-i} c_{j,j+i|(j+1):(j+i-1)}. \qquad (2.29)$$

Statistical inference for C- and D-vines has been discussed in Aas et al. (2009). The general case of R-vines was recently explored by Dißmann (2010) and will be the fundamental basis of this thesis. Amongst others, algorithms for likelihood computation and simulation are given as well as a convenient representation of an R-vine in terms of a matrix, a so-called *R-vine matrix*. As we will use these matrices to display results of our applications in Chapter 11, we present this concept in the following section.

2.4.1   Regular vine matrices

R-vine matrices (RVM's) have recently been introduced by Dißmann (2010) based on work of Kurowicka (2009). In order to formally define them, we need the following notations. For this let $M = (m_{i,j})_{i,j=1,...,d} \in \{1, ..., d\}^{d \times d}$ be a lower triangular matrix.

(i) We denote the set of the non-zero entries in the i-th column of M by

$$L_M(i) = \{m_{i,i}, ..., m_{d,i}\}.$$

(ii) Further we define the following two sets

$$B_M(i) = \{(m_{i,i}, D) : k = i + 1, ..., d, \; D = \{m_{k,i}, ..., m_{d,i}\}\},$$
$$\widetilde{B}_M(i) = \{(m_{k,i}, D) : k = i + 1, ..., d, \; D = \{m_{i,i}\} \cup \{m_{k+1,i}, ..., m_{d,i}\}\}.$$

Using these notations we can now define an RVM (see Definition 1.14 in Dißmann (2010)).

Definition 2.17 (R-vine matrix.) *Let $M \in \{1, ..., d\}^{d \times d}$ be a lower triangular matrix. $M = (m_{i,j})_{i,j=1,...,d}$ is called an R-vine matrix if it satisfies the following conditions:*

(i) $L_M(i) \subset L_M(j)$ for $1 \leq j < i \leq d$,

(ii) $m_{i,i} \notin L_M(i+1)$ for $i = 1, ..., d-1$, and

(iii) for $i = 1, ..., d-1$ and for all $k = i+1, ..., d-1$,

$$
(m_{k,i}, \{m_{k+1,i}, ..., m_{d,i}\}) \in B_M(i+1) \cup ... \cup B_M(d-1)
$$
$$
\cup \widetilde{B}_M(i+1) \cup ... \cup \widetilde{B}_M(d-1). \tag{2.30}
$$

It can be shown that conditions (i) and (ii) follow from condition (iii), but they facilitate understanding of the definition. Condition (i) states that all entries of a column have to be contained in all columns on the left of this column. Together with the second condition, which ensures that there is a new entry on the diagonal in each column, this means that the variables are added to the RVM sequentially from the right to the left. The third property (2.30) is however rather tedious to check for a given matrix, but it is the critical condition of Definition 2.17 and corresponds to the proximity condition in Definition 2.15. Instead of going too much into the details here (cf. Dißmann (2010)), we will now consider how to "read" an RVM, i.e., how to construct an R-vine from a given RVM. This can be done as follows. Note however that RVM's are not unique, but there are 2^{d-1} different RVM's which correspond to the same R-vine (see Theorem 3.20 in Dißmann (2010)).

(i) The nodes of T_1 are given by $1, ..., d$.

(ii) The edges of T_1, and hence the nodes of T_2, are given by

$$
\{\{m_{i,i}, m_{d,i}\} : i = 1, ..., d-1\},
$$

i.e., by the diagonal element and the last element of columns $i = 1, ..., d-1$.

(iii) The edges of T_2 (and nodes of T_3) are given by

$$
\{\{m_{i,i}, m_{d-1,i} | m_{d,i}\} : i = 1, ..., d-2\}, \tag{2.31}
$$

i.e., by the diagonal element and the second last element conditioned on the last element of columns $i = 1, ..., d-2$.

(iv) In general, the edges of T_j are given by

$$
\{\{m_{i,i}, m_{d-j+1,i} | m_{d-j+2,i}, ..., m_{d,i}\} : i = 1, ..., d-2\},
$$

i.e., by the diagonal element and the element in row $d - j + 1$ conditioned on the last elements of columns $i = 1, ..., d-j$
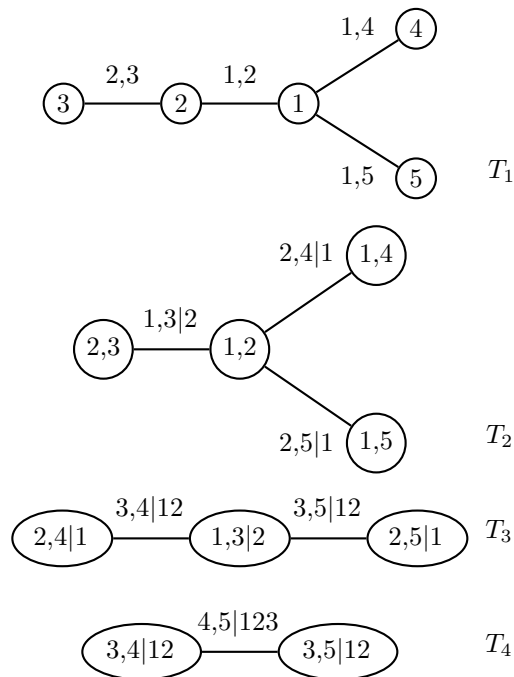
Figure 2.6: Five-dimensional R-vine corresponding to the RVM given in (2.32).

An illustrative example will be given in the following. General algorithms to transform R-vines into RVM's and vice versa can be found in Dißmann (2010).

Example 2 (Five-dimensional RVM.) *Let us consider the five-dimensional shown in Figure 2.6. A corresponding RVM is given by*

$$
M = \begin{pmatrix}
4 & & & & \\
5 & 3 & & & \\
3 & 5 & 5 & & \\
2 & 1 & 2 & 2 & \\
1 & 2 & 1 & 1 & 1
\end{pmatrix},
\tag{2.32}
$$

where zero entries in the upper triangle are omitted for simplicity.

It is obvious that conditions (i) and (ii) of Definition 2.17 are satisfied: the variables are considered in the order $1 - 2 - 5 - 3 - 4$ and each column is contained in all columns on the left of it. Condition (2.30) can also be checked with some more effort.

As described above, the R-vine corresponding to M and given in Figure 2.6 can then be constructed as follows:

(i) Edges of tree T_2 and nodes of tree T_2: $\{m_{1,1}, m_{5,1}\} = \{1, 4\}$, $\{m_{2,2}, m_{5,2}\} = \{2, 3\}$, $\{m_{3,3}, m_{5,3}\} = \{1, 5\}$ and $\{m_{4,4}, m_{5,4}\} = \{1, 2\}$.

(ii) *According to (2.31), the edges of tree T_2 (and nodes of tree T_3) can then be obtained by*

$$\{m_{1,1}, m_{4,1}|m_{5,1}\} = \{2,4|1\} \left\{ \begin{array}{ccccc} \boxed{4} \\ 5 & 3 \\ 3 & 5 & 5 \\ \boxed{2} & 1 & 2 & 2 \\ \hline \boxed{1} & 2 & 1 & 1 & 1 \end{array} \right.$$

and similarly for the second and third columns: $\{m_{2,2}, m_{4,2}|m_{5,2}\} = \{1,3|2\}$ and $\{m_{3,3}, m_{4,3}|m_{5,3}\} = \{2,5|1\}$.

(iii) *As in the previous step, we obtain the edges of tree T_3 and the nodes of tree T_4 as*

$$\{m_{1,1}, m_{3,1}|m_{4,1}, m_{5,1}\} = \{3,4|1,2\} \left\{ \begin{array}{ccccc} \boxed{4} \\ 5 & 3 \\ \boxed{3} & 5 & 5 \\ \hline \boxed{2} & 1 & 2 & 2 \\ \boxed{1} & 2 & 1 & 1 & 1 \end{array} \right.$$

and $\{m_{2,2}, m_{3,2}|m_{4,2}, m_{5,2}\} = \{3,5|1,2\}$.

(iv) *Finally, the edge of tree T_4 is given by $\{m_{1,1}, m_{2,1}|m_{3,1}, m_{4,1}, m_{5,1}\} = \{4,5|1,2,3\}$.*

C- and D-vines as defined in Defintion 2.16 can be represented by particularly well-structured RVM's.

(i) C-vine:

$$M = \begin{pmatrix} d & & & & & \\ d-1 & d-1 & & & & \\ d-2 & d-2 & d-2 & & & \\ \vdots & \vdots & \vdots & \ddots & & \\ 2 & 2 & 2 & \cdots & 2 & \\ 1 & 1 & 1 & \cdots & 1 & 1 \end{pmatrix}$$

This RVM corresponds to a C-vine which is defined by its root nodes: 1 in the first tree T_1, 2 in the second tree T_2, and so on. This order can easily be obtained by reading the first column bottom-up. The characteristic property of C-vines that the copulas of each tree have a common conditioning set is easily validated, since the entries within each row are the same.

(ii) D-vine:

$$
M = \begin{pmatrix}
d & & & & & \\
1 & d-1 & & & & \\
\vdots & & \ddots & & \ddots & \\
d-3 & \cdots & & 1 & 3 & \\
d-2 & d-3 & \cdots & & 1 & 2 \\
d-1 & d-2 & d-3 & \cdots & & 1 & 1
\end{pmatrix}
$$

The first R-vine tree constructed according to this RVM is obviously given by the pairs 1–2, 2–3,..., $(d-1)$–d which is the first tree of a D-vine. As D-vines are uniquely determined by their first tree, this RVM defines a D-vine. The characteristic feature of D-vines, that conditioning sets consist of those nodes lying in between in the first tree, is satisfied by this RVM, since the entries within each row decrease by one per column from left to right.

Chosen copula types and parameters corresponding to an R-vine specification with pair copulas can easily be denoted in matrix form as well. Since the diagonal entry in each column uniquely defines one element of the conditioned sets of the edges corresponding to this column, the corresponding pair copula types and parameters can be stored in the corresponding off-diagonal entry, i.e., copula type and parameter(s) corresponding to $\{m_{i,i}, m_{k,i} | m_{k-1,i}, ..., m_{d,i}\}$, $k < i$, are stored in entry $m_{k,i}$. This is illustrated in the following example.

Example 3 (R-vine copula type and parameter matrices.) *We consider again the R-vine of Example 2 defined by the RVM given in (2.32). Further, let us consider the following R-vine copula type and parameter matrices T and P_1, respectively:*

$$
T = \begin{pmatrix}
N & & & \\
\Pi & G & & \\
C & N & \Pi & \\
N & C & N & F
\end{pmatrix}, \qquad
P_1 = \begin{pmatrix}
0.20 & & & \\
0.00 & 1.75 & & \\
0.67 & 0.30 & 0.00 & \\
0.85 & 2.70 & 0.60 & 1.35
\end{pmatrix},
$$

where Π denotes an independence copula, N a Gaussian, C a Clayton, G a Gumbel, and F a Frank. Then we can identify, e.g., the copula type and parameter of the edge $1,3|2$ as described above:

$$
M = \begin{pmatrix}
4 & & & & \\
5 & \mathbf{3} & & & \\
3 & 5 & 5 & & \\
2 & \boxed{\mathbf{1}} & 2 & 2 & \\
1 & \mathbf{2} & 1 & 1 & 1
\end{pmatrix},
T = \begin{pmatrix}
N & & & \\
\Pi & G & & \\
C & \boxed{N} & \Pi & \\
N & C & N & F
\end{pmatrix},
P_1 = \begin{pmatrix}
0.20 & & & \\
0.00 & 1.75 & & \\
0.67 & \boxed{0.30} & 0.00 & \\
0.85 & 2.70 & 0.60 & 1.35
\end{pmatrix},
$$

i.e., the copula of the edge $1, 3|2$ *is a Gaussian copula with parameter* 0.30. *Other copula types and parameters are identified similarly.*

If a copula belongs to a two-parametric family such as the t, BB1 and BB7 copulas (cp. Section 2.1.3), we need to specify a second copula parameter matrix P_2 for the respective second parameter, e.g., the degrees of freedom of a t copula.

2.5 Time series analysis

After having concentrated on the copula part of multivariate copula distributions, we will now turn to the modeling of the margins. In particular, we consider time series as they are usually found in financial applications. One of the purposes of time series analysis is to remove serial dependence among observations in order to obtain i.i.d. data. This data can then be used as input for copula models. Here we will introduce some basic concepts and two fundamental models, namely the linear ARMA- and the non-linear GARCH-model. The discussion of the basics and the ARMA-model is based on the standard work by Brockwell and Davis (1991).

2.5.1 Basic definitions

A *time series* is a set of realizations of a stochastic process, i.e., a family of random variables $(X_t)_{t \in T}$ on a probability space $(\Omega, \mathcal{F}, P)$, where T is called the *time domain* and in the following we consider $T = \mathbb{Z}$. The characteristic feature of a time series is hence its chronological order which has to be accounted for in an adequate analysis and model building.

Like covariance matrices for finite numbers of random variables, the autocovariance function summarizes important information about the dependency of infinite collections of random variables:

Definition 2.18 (Autocovariance function.) *If* $(X_t)_{t \in \mathbb{Z}}$ *is a stochastic process with* $E(X_t^2) < \infty$ $\forall t \in \mathbb{Z}$, *then the autocovariance function (ACF)* γ_X *of* $(X_t)_{t \in \mathbb{Z}}$ *is defined as*

$$\gamma_X(r, s) = Cov(X_r, X_s), \ r, s \in \mathbb{Z}. \tag{2.33}$$

Most of the theory of time series analysis is built on *stationary* processes, which can be defined as follows:

Definition 2.19 (Stationarity.) *The time series* $(X_t)_{t \in \mathbb{Z}}$ *is called stationary if*

(i) $E(X_t^2) < \infty$ $\forall t \in \mathbb{Z}$,

(ii) $E(X_t) = m \in \mathbb{R}$ $\forall t \in \mathbb{Z}$, *and*

(iii) $\gamma_X(r, s) = \gamma_X(r + t, s + t)$ $\forall r, s, t \in \mathbb{Z}$
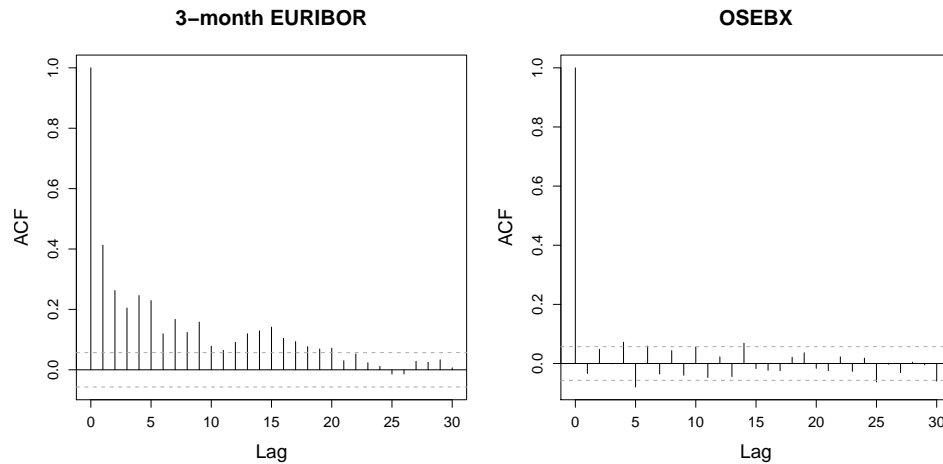
Figure 2.7: Empirical ACF's of the log returns of the 3-month Euro Interbank Offered Rate (EURIBOR) and the Oslo Stock Exchange main index (OSEBX) with lags h up to 30 and 95%-confidence intervals (dashed lines). The observed time period is from 3/25/2003 to 3/26/2008 (see Section 11.2).

Definition (2.19) is often referred to as *weak stationarity*. We will however concentrate on this property, since the concept of *strict stationarity* is too "strict" in many circumstances. Hence, when we talk about stationarity, we mean weak stationarity.

Obviously, the ACF (2.33) of a stationary process only depends on $|r - s|$, since $\gamma_X(r, s) = \gamma_X(r - s, 0)$. Therefore we can redefine the ACF of a stationary process as

$$\gamma_X(h) := \gamma_X(h, 0) = \text{Cov}(X_{t+|h|}, X_t) \ \forall t, h \in \mathbb{Z}.$$

In practice, the ACF is then estimated by its empirical analogue

$$\hat{\gamma}(h) := \frac{1}{n} \sum_{j=1}^{n-h} (x_{j+h} - \bar{x})(x_j - \bar{x}), \ |h| < n,$$

where $\bar{x}$ is the sample mean $\bar{x} = \frac{1}{n} \sum_{j=1}^{n} x_j$. Examples are shown in Figure 2.7 which displays the empirical ACF's of the log returns of the 3-month Euro Interbank Offered Rate (EURIBOR) and the Oslo Stock Exchange main index (OSEBX) with *lags h* up to 30 and confidence intervals at the 95% level (cp. Section 11.2).

An important example of a stationary process is the so-called *white noise*:

Definition 2.20 (White noise.) *A stationary process $(Z_t)_{t\in\mathbb{Z}}$ with $E(Z_t) = 0 \ \forall t \in \mathbb{Z}$ and ACF*

$$\gamma_Z(h) = \begin{cases} \sigma_Z^2 & \text{for } h = 0, \\ 0 & \text{for } h \neq 0, \end{cases}$$

where $\sigma_Z^2 > 0$, is called white noise and denoted as $(Z_t)_{t\in\mathbb{Z}} \sim WN(0, \sigma_Z^2)$.

2.5.2 The ARMA-model

Autoregressive moving average (ARMA) models are the fundamental class of linear time series models and very popular in practice. A short introduction to such models is given in the following. Detailed theoretical results can be found in Brockwell and Davis (1991).

Definition 2.21 (ARMA(p,q) process.) *The process $(X_t)_{t\in\mathbb{Z}}$ is called an ARMA(p,q) process if $(X_t)_{t\in\mathbb{Z}}$ is stationary and if for every t,*

$$X_t - \sum_{j=1}^{p} \phi_j X_{t-j} = Z_t + \sum_{j=1}^{q} \theta_j Z_{t-j}, \tag{2.34}$$

where $p, q \in \mathbb{N}_0$, $\phi_1, ..., \phi_p, \theta_1, ..., \theta_q \in \mathbb{R}$ and $(Z_t)_{t\in\mathbb{Z}} \sim WN(0, \sigma_Z^2)$. $(X_t)_{t\in\mathbb{Z}}$ is called an ARMA(p,q) process with mean μ if $(X_t - \mu)_{t\in\mathbb{Z}}$ is an ARMA(p,q) process.

Introducing the *Backward shift operator B* as $B^j(X_t) = X_{t-j}$ for all $j \geq 1$ and $t \in \mathbb{Z}$ for a process $(X_t)_{t\in\mathbb{Z}}$, we can rewrite (2.34) as

$$\Phi(B)X_t = \Theta(B)Z_t \ \forall t \in \mathbb{Z}, \tag{2.35}$$

where Φ and Θ are the so-called *AR-* and *MA-polynomials* defined as

$$\Phi(z) = 1 - \phi_1 z - ... - \phi_p z^p, \text{ and}$$
$$\Theta(z) = 1 + \theta_1 z + ... + \theta_q z^q.$$

Then a fundamental result on the existence of a unique solution to the ARMA-equation (2.35) is given in the following theorem.

Theorem 2.22 *If $\Phi(z) \neq 0$ for all $z \in \mathbb{C}$ such that $|z| = 1$, then the ARMA-equation (2.35) has the unique stationary solution*

$$X_t = \sum_{j=-\infty}^{\infty} \psi_j Z_{t-j} \ \forall t \in \mathbb{Z},$$

where the coefficients $(\psi_j)_{j\in\mathbb{Z}}$ are determined by

$$\sum_{j=-\infty}^{\infty} \psi_j z^j = \Theta(z)\Phi(z)^{-1} \text{ for } \frac{1}{r} < |z| < r,$$

with an $r > 1$.

2.5.3 The GARCH-model

ARMA-models as considered in the previous section are able to remove trends and seasonality in observations. However, they are based on a white noise process with constant variance σ_Z^2. Especially, in finance, observations exhibit changing volatility behavior which contradicts the assumptions of an ARMA-model with i.i.d. noise (see Figure 2.8 which
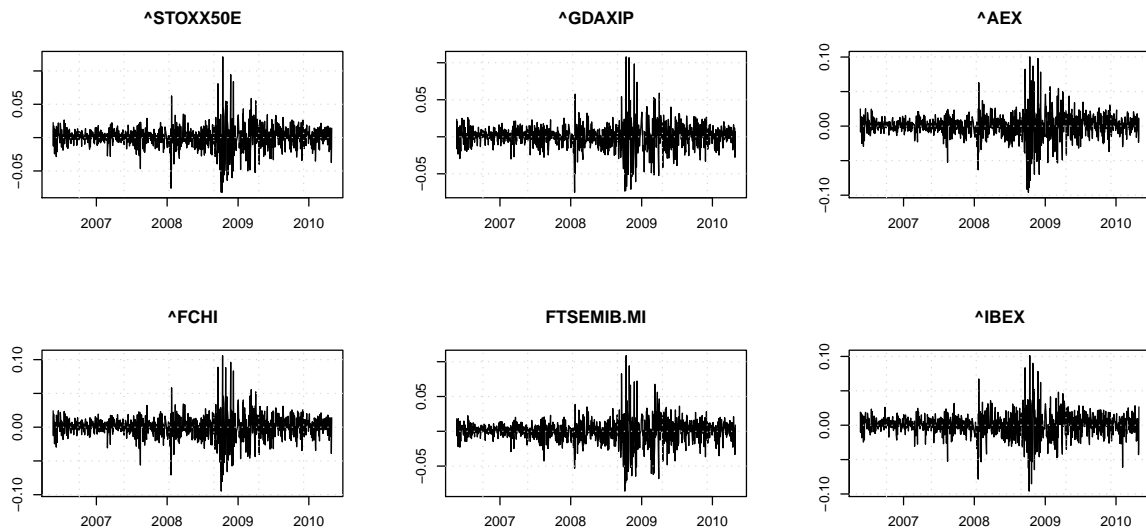
Figure 2.8: Log returns of the Euro Stoxx 50 and the leading national stock indices of Germany, the Netherlands, France, Italy and Spain over a period of four years (see Section 11.3).

shows the log returns of the Euro Stoxx 50 and the leading national stock indices of five countries over a period of four years, cp. Section 11.3). Such changing volatilities usually occur in clusters of higher or lower volatility, i.e., the variance of the time series shows a conditional behavior and we cannot assume that the noise is i.i.d..

Therefore we will now consider the *generalized autoregressive conditional heteroscedastic (GARCH)* model which was introduced by Bollerslev (1986) and can capture such non-linear behavior. Its definition is given in the following definition.

Definition 2.23 (GARCH(p,q) process.) *The process $(\varepsilon_t)_{t \in \mathbb{Z}}$ is called a GARCH(p,q) process if for every t,*

$$\varepsilon_t = \sigma_t Z_t \text{ and } \sigma_t^2 = \omega + \sum_{j=1}^q \alpha_j \varepsilon_{t-j}^2 + \sum_{j=1}^p \beta_j \sigma_{t-j}^2,$$

where $p \in \mathbb{N}_0$, $q \in \mathbb{N}$, $\omega > 0$, $\alpha_1, ..., \alpha_p, \beta_1, ..., \beta_q \geq 0$ and $(Z_t)_{t \in \mathbb{Z}}$ is an i.i.d. sequence with zero mean and unit variance independent of $\{\varepsilon_{t-k}, k \geq 1\}$ for all t.

In our applications we are particularly interested in simple but often very useful GARCH(1,1)-models. It can be shown that the GARCH(1,1) process is stationary if $\alpha_1 + \beta_1 < 1$. Furthermore we can combine the ARMA-model from Definition 2.21 with a GARCH-model for the noise as introduced above and obtain ARMA-GARCH-models. In particular, the ARMA(1,1)-GARCH(1,1)-model with mean μ is defined as follows

$$\begin{aligned} X_t &= \mu + \phi_1 X_{t-1} + \varepsilon_t + \theta_1 \varepsilon_{t-1}, \\ \sigma_t^2 &= \omega + \alpha_1 \varepsilon_{t-1}^2 + \beta_1 \sigma_{t-1}^2, \end{aligned} \tag{2.36}$$

where $\varepsilon_t = \sigma_t Z_t$ and $(Z_t)_{t\in\mathbb{Z}}$ is defined as in Definition 2.23. The standardized residuals of this model are then given by

$$\hat{Z}_t = \frac{1}{\hat{\sigma}_t}\left(X_t - \hat{\mu} - \hat{\phi}_1 X_{t-1} - \hat{\theta}_1\hat{\sigma}_{t-1}\hat{Z}_{t-1}\right),\tag{2.37}$$

where $\hat{\mu}$, $\hat{\phi}_1$, $\hat{\theta}_1$ and $\hat{\sigma}_t$ are the estimates of μ, ϕ_1, θ_1 and σ_t.

The simplest choice of an error distribution for $(Z_t)_{t\in\mathbb{Z}}$ is certainly the standard normal distribution, i.e., $Z_t \sim N(0,1)$. However, financial data often not only exhibits volatility clustering but also negative skewness and/or heavy tails. Therefore, we will also consider skewed and heavy-tailed distribution such as

(i) the Student-t distribution (Bollerslev 1987) (cp. Section 2.1.1);

(ii) the skewed Student-t distribution (Hansen 1994) defined through its density

$$f(x) = \frac{2}{\gamma + \frac{1}{\gamma}}\frac{\Gamma(\frac{\nu+1}{2})}{\Gamma(\frac{\nu}{2})\sqrt{\pi\nu\rho}}\left[1 + \frac{(x-\mu)^2}{\nu\rho}\left(\frac{1}{\gamma^2}\mathbf{1}_{[0,\infty)}(x-\mu) + \gamma^2\mathbf{1}_{(-\infty,0)}(x-\mu)\right)\right]^{\frac{\nu+1}{2}},$$

with location $\mu \in \mathbb{R}$, degrees of freedom $\nu > 0$ and skew parameter $\gamma > 0$ (cp. Fernandez and Steel (1998));

(iii) the Normal Inverse Gaussian (NIG) distribution (Andersson 2001) with density

$$f(x) = \frac{\alpha\delta K_1\left(\alpha\sqrt{\delta^2 + (x-\mu)^2}\right)}{\pi\sqrt{\delta^2 + (x-\mu)^2}}\exp\left[\delta\sqrt{\alpha^2 - \beta^2} + \beta(x-\mu)\right],$$

where K_1 is the modified Bessel function of third order and index 1. The parameters satisfy $0 \le |\beta| \le \alpha$ (asymmetry and tail heavyness), $\mu \in \mathbb{R}$ (location) and $\delta > 0$ (scale).

Since Definition 2.23 requires $E(Z_t) = 0$ and $Var(Z_t) = 1$, we use standardized versions of these distributions.

2.5.4 The Ljung-Box test

Having specified a time series model to given observations $(X_t)_{t=1,\dots,n}$, one is usually interested in the goodness-of-fit, i.e., one investigates whether the chosen model is appropriate. In terms of time series analysis, this corresponds to checking whether the residuals of the fitted model still exhibit serial dependence. The test by Ljung and Box (1978) addresses this issue by examining the hypotheses

$$H_0 : (\hat{Z}_t)_{t=1,\dots,n} \text{ is white noise} \quad \text{against} \quad H_1 : \text{ not } H_0,\tag{2.38}$$

where $(\hat{Z}_t)_{t=1,\dots,n}$ denote the standardized residuals of the model which are given in (2.37) for the ARMA(1,1)-GARCH(1,1)-model and similarly for general ARMA-GARCH-models.

The test statistic of the Ljung-Box test is constructed by considering the sample autocorrelation of $(\hat{Z}_t)_{t=1,...,n}$ which is defined as

$$\hat{\rho}_h = \frac{\sum_{t=h+1}^{n} \hat{Z}_t \hat{Z}_{t-h}}{\sum_{t=1}^{n} \hat{Z}_t^2},$$

for lags $h = 1, ..., n - 1$. Ljung and Box (1978) developed a test which jointly considers the autocorrelations of the first m lags, where $1 \leq m \leq n - 1$. The corresponding test statistic is given by

$$\hat{Q}(\hat{\rho}) = n(n + 2) \sum_{h=1}^{m} \frac{\hat{\rho}_h^2}{n - h}.$$

Under the null hypothesis (2.38), $\hat{Q}(\hat{\rho})$ asymptotically follows a χ^2 distribution with $m - p$ degrees of freedom, where p is the number of parameters in the chosen model. E.g., the ARMA(1,1)-GARCH(1,1)-model (2.36) uses 6 parameters if the error distribution is normal and 7 parameters if a Student-t distribution is chosen instead, which needs an additional shape parameter.

Chapter 3

Construction of regular vines

Before selecting any copulas or fitting any copula parameters, we have to determine which pairs of (transformed) variables are being modeled with copulas at all, i.e., we have to construct the trees of the R-vine for our observed data. As the number of possible R-vines is growing exponentially with dimension d (cp. Section 2.4), one cannot simply fit all possible R-vines and choose the best one. We therefore rely on the use of heuristic methods to determine the structure of the trees in order to obtain a good final model.

These heuristic methods typically proceed sequentially, i.e., one tree is determined each step. Considering the structure of pair copula constructions, this is the only feasible approach because for the construction of tree T_i we need transformed observations using those copulas that are specified in tree T_{i-1} (cp. (2.19)).

By this sequential method we also hope to capture the most important dependencies in the first trees so that the R-vine can possibly be simplified, or even truncated, after having specified a certain (hopefully small) number of trees (see Chapters 6, 7 and 8). In order to do this we have to define what we mean when referring to "important" dependencies. This amounts to assigning weights to each pair of variables. We will investigate different choices in the first part of this chapter. Subsequently we consider the general case of constructing R-vines before turning to the special cases of C- and D-vines. Although the latter cases both lead to the same number of possible trees (see Section 2.4), the complexity of the tree construction methods is considerably different: constructing the first tree of a D-vine results in an NP-equivalent problem, while a complete C-vine can be constructed in $\mathcal{O}(d^3)$ operations.

3.1 Weights

In this section we propose different methods to compute weights for pairs of variables. Which method is chosen mainly depends on the purpose of the data analysis. E.g., in finance one is often interested in capturing as much tail dependence as possible in the first trees.

In the following we will always work with the transformed variables $U_j = F_j(X_j)$, $j = 1, 2$, where F_1 and F_2 are the cdf's of X_1 and X_2, respectively, or, if F_1 and F_2 are unknown, their empirical versions $U_{ij} = \frac{R_{ij}}{n+1} = \frac{n}{n+1}\hat{F}_j(X_{ij})$, $i = 1, ..., n$, $j = 1, 2$, where

R_{ij} is the rank of X_{ij} and $\hat{F}_j$ the empirical cdf of $X_{1j}, ..., X_{nj}$, $j = 1, 2$, since we are not interested in any measure that is dependent on the choice of margins anyway. This issue will be discussed in more detail in Chapter 4. Furthermore, n always denotes the number of observations, in contrast to the number of variables which is denoted by d.

3.1.1 Kendall's τ and Spearman's ρ

A very natural choice of weights are the empirical versions of dependence measures such as Kendall's τ and Spearman's ρ (cp. Kurowicka and Cooke (2006)):

$$
\hat{\tau}_n(X_1, X_2) = \frac{c_n - d_n}{\sqrt{c_n + d_n + e_n^{(1)}}\sqrt{c_n + d_n + e_n^{(2)}}},
$$
$$
\hat{\rho}_n(X_1, X_2) = \frac{\sum_{i=1}^n \left(R_{i1} - \bar{R}_1\right)\left(R_{i2} - \bar{R}_2\right)}{\sqrt{\sum_{i=1}^n \left(R_{i1} - \bar{R}_1\right)^2}\sqrt{\sum_{i=1}^n \left(R_{i2} - \bar{R}_2\right)^2}},
$$

(3.1)

where c_n denotes the number of concordant pairs, d_n the number of discordant pairs and $e_n^{(1)}$ ($e_n^{(2)}$) the number of tied pairs with $x_{i1} = x_{j1}$ ($x_{i2} = x_{j2}$). Furthermore $\bar{R}_1 = \frac{1}{n}\sum_{i=1}^n R_{i1}$ and $\bar{R}_2 = \frac{1}{n}\sum_{i=1}^n R_{i2}$. If there are no ties, $\hat{\tau}_n$ simplifies to $\hat{\tau}_n(X_1, X_2) = \frac{c_n - d_n}{c_n + d_n} = (c_n - d_n)/\binom{n}{2}$.

These measures adequately summarize the joint behavior of two random variables in a single value (see Section 2.2.1). However, they cannot account for (asymmetric) behavior in the tails.

3.1.2 Tail dependence

The fundamental concept of measuring the joint tail behavior of two random variables is certainly the lower and upper tail dependence (see Section 2.2.2). Based on the definition, the most natural approach for estimating lower and upper tail dependence is substituting the theoretical copula C by its empirical version C_n:

$$
C_n(u_1, u_2) = \frac{1}{n}\sum_{i=1}^n \mathbf{1}_{\{U_{i1} \leq u_1, U_{i2} \leq u_2\}},
$$

where $u_1, u_2 \in [0, 1]$ and $\mathbf{1}$ denotes the indicator function. It will be discussed in more detail in Section 4.2.1 and defined for general dimension d in (4.5). We first investigate an estimate of the lower tail dependence (2.25) and obtain

$$
\begin{aligned}
\hat{\lambda}_{n,k}^{lower} &= \frac{C_n\left(\frac{k}{n}, \frac{k}{n}\right)}{\frac{k}{n}} = \frac{1}{k}\sum_{i=1}^n \mathbf{1}_{\{U_{i1} \leq \frac{k}{n}, U_{i2} \leq \frac{k}{n}\}} \\
&= \frac{1}{k}\sum_{i=1}^n \mathbf{1}_{\{R_{i1} \leq \frac{n+1}{n}k, R_{i2} \leq \frac{n+1}{n}k\}},
\end{aligned}
$$

(3.2)

where $k \in \mathbb{N}$ has to be determined as a function of n with $k = k(n) \xrightarrow{n\to\infty} \infty$ and $\frac{k}{n} \xrightarrow{n\to\infty} 0$. This estimator is similar to the classical estimator of Huang (1992) and to

	τ	N	true	t	true	C	true	G	true	F	true
lower	0.25	0.14 (.057)	0	0.26 (.070)	0.20	0.38 (.080)	0.35	0.10 (.050)	0	0.08 (.046)	0
	0.5	0.36 (.074)	0	0.46 (.076)	0.40	0.70 (.066)	0.70	0.24 (.068)	0	0.15 (.059)	0
	0.75	0.65 (.066)	0	0.71 (.065)	0.68	0.88 (.045)	0.89	0.51 (.074)	0	0.31 (.072)	0
upper	0.25	0.14 (.059)	0	0.25 (.073)	0.20	0.05 (.038)	0	0.33 (.078)	0.59	0.07 (.045)	0
	0.5	0.35 (.074)	0	0.46 (.078)	0.40	0.09 (.050)	0	0.59 (.076)	0.70	0.15 (.061)	0
	0.75	0.65 (.067)	0	0.71 (.066)	0.68	0.18 (.064)	0	0.81 (.054)	0.84	0.31 (.073)	0

Table 3.1: Estimated lower and upper tail dependence $\hat{\lambda}_{n,k}^{lower}$ and $\hat{\lambda}_{n,k}^{upper}$ using $k = \lfloor \sqrt{n} \rfloor$ for $R = 1000$ repetitions and three choices of Kendall's τ to determine the copula parameters (cp. Table 2.1). Standard errors are displayed in brackets, true values in the respective columns (also see Table 2.1).

estimator $\hat{\lambda}_{n,k}^{(1)}$ in Dobrić and Schmid (2005). The latter authors prove that if C has continuous partial derivatives and $k \approx \sqrt{n}$ asymptotically, then $\hat{\lambda}_{n,k}^{lower}$ is weakly consistent and asymptotically unbiased. The choice of $k \approx \sqrt{n}$ is called the "square root of n rule" and we will use it in the following, to be more precise we will use $k = \lfloor \sqrt{n} \rfloor$.

Similarly to the lower tail dependence estimator in (3.2) we can define an estimator of the upper tail dependence parameter (2.26) based on an empirical version of the joint survival function (2.1)

$$\bar{C}_n(u_1, u_2) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{\{U_{i1}>1-u_1, U_{i2}>1-u_2\}}.$$

This directly yields the following estimate

$$\hat{\lambda}_{n,k}^{upper} = \frac{1}{k} \sum_{i=1}^{n} \mathbf{1}_{\{R_{i1}>n+1-\frac{n+1}{n}k, R_{i2}>n+1-\frac{n+1}{n}k\}}. \tag{3.3}$$

To evaluate the performance of these two estimators, we simulated $n = 1000$ observations, which we believe is a reasonable sample size in many applications, from five bivariate copula families (Gaussian (N), t with four degrees of freedom, Clayton (C), Gumbel (G), Frank (F)) with three different choices of Kendall's τ which determine the respective copula parameters and computed the lower and upper tail dependence estimates $\hat{\lambda}_{n,k}^{lower}$ and $\hat{\lambda}_{n,k}^{upper}$ as given in (3.2) and (3.3), respectively. We repeated this $R = 1000$ times. The results are shown in Table 3.1.

The symmetric tail dependence of the observations generated by the t copula and the asymmetric one of the observations from the Clayton and Gumbel copulas are well
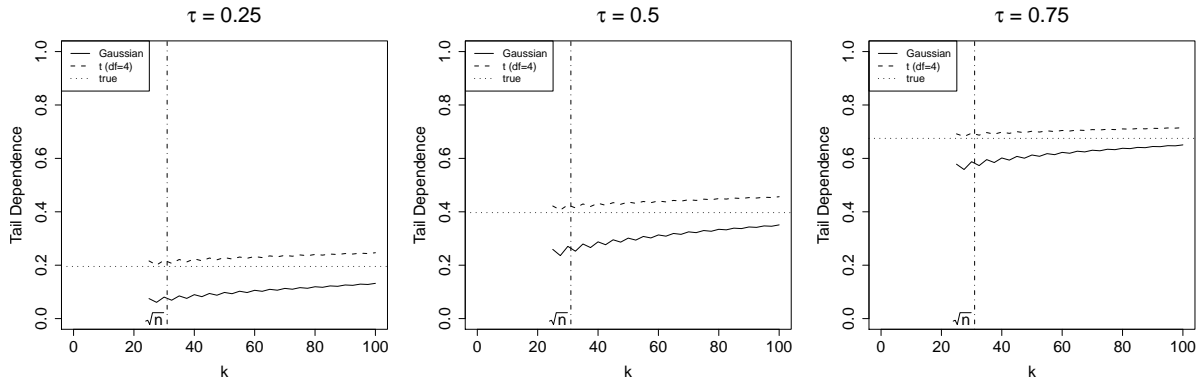
Figure 3.1: Tail dependence of Gaussian and t copulas estimated as in Klüppelberg et al. (2007) with three different choices of Kendall's τ which determine the copula parameters (cp. Table 2.1).

estimated. The estimates of the respective opposed tails for the two latter copulas are also essentially adequate (more accurate for the Clayton than the Gumbel copula) as well as the estimates for the Frank copula observations. However, the estimates for the observations from the Gaussian copula are surprisingly close to the estimates based on the data from the t copula. This is a serious drawback of these estimators. Moreover, in all cases with theoretically zero tail dependence, the estimates, especially for the observations from the Gaussian copula and less for those from the Frank copula, get worse as Kendall's τ increases.

To investigate these problems further, we consider an alternative tail dependence estimator, namely the estimator proposed by Klüppelberg et al. (2007). This estimator is constructed for elliptical distributions and therefore should be particularly useful for the comparison of observations from Gaussian and t copulas. For a definition of the estimator we refer to the paper (since elliptical distributions are symmetric and hence lower and upper tail dependence coincide, we only need one estimator). Here we report the results of a Monte Carlo study with the same setting as before, but only for the Gaussian and the t copula (the consideration of Archimedean copulas such as the Clayton or Gumbel is of course not sensible). Furthermore, we considered the parameter k, which determines the number of observations used in the estimation, for values between 25 and 100 (= 10% of all observations). The results are displayed in Figure 3.1. Our usual choice of $k = \lfloor \sqrt{n} \rfloor = 31$ is marked. It seems that it is a good choice in this setting as well.

Apparently, the estimator of Klüppelberg et al. (2007) gives a good estimate for observations generated by a t copula. However, even if it is designed especially for elliptical distributions, it does not reflect the zero tail dependence induced by the Gaussian copula either. As this estimator also exhibits this problem and it is not universally applicable, it is no reasonable alternative to our estimators $\hat{\lambda}_{n,k}^{lower}$ and $\hat{\lambda}_{n,k}^{upper}$.

Dobrić and Schmid (2005) confirm our results. In a Monte Carlo study they show that $\hat{\lambda}_{n,k}^{lower}$ is biased in finite samples. But the authors also state that there are currently no unbiased estimators available for omnibus use. Therefore we will use $\hat{\lambda}_{n,k}^{lower}$ and $\hat{\lambda}_{n,k}^{upper}$ because of their intuitively appealing form, but have to keep in mind their potential bias

and the problems with distinguishing clearly between observations from a Gaussian and a t copula with significantly different theoretical tail behavior.

3.1.3   Exceedance dependence

As the estimation of tail dependence is rather difficult and prone to bias due to the small proportion of observation which can be used for estimation (e.g., for $n = 1000$ and the "square root of n rule", only 31 observations are used!), we propose an alternative method to determine the tail behavior of pairs of random variables.

Ang and Chen (2002) use the measure of *exceedance correlation* with certain thresholds δ_1 and δ_2

$$corr(X_1, X_2 | X_1 \leq \delta_1, X_2 \leq \delta_2). \tag{3.4}$$

This measure is however not independent of the margins of X_1 and X_2, since it is based on Pearson's product-moment correlation coefficient (2.20). Therefore we slightly modify definition (3.4) and use the common dependence measures Kendall's τ and Spearman's ρ (cp. Section 2.2.1) instead of the Pearson correlation. Furthermore we define a lower and an upper version of exceedance dependence to consider both types of joint tail behavior. We thus obtain the *lower* and *upper exceedance Kendall's* τ, respectively, which are defined as follows:

$$\tau^{lower}(U_1, U_2 | U_1 \leq \delta_1, U_2 \leq \delta_2),$$
$$\tau^{upper}(U_1, U_2 | U_1 > \delta_1, U_2 > \delta_2).$$

The definition is stated in terms of the pseudo-variables U_1 and U_2 so that we can choose δ_1 and δ_2 in $[0, 1]$. (Otherwise we obtain the same results if we consider the thresholds $F_1(\delta_1)$ and $F_2(\delta_2)$ instead.) The *lower* and *upper exceedance Spearman's* ρ are given in the same way.

Similarly to Theorem 2.9 (cp. Theorems 5.1.1 and 5.1.3 in Nelsen (2006)) theoretical expressions of lower and upper exceedance Kendall's τ for continuous random variables with copula C can be obtained by

$$\tau^{lower}(U_1, U_2 | U_1 \leq \delta_1, U_2 \leq \delta_2) = \frac{4}{C(\delta_1, \delta_2)^2} \int_0^{\delta_2} \int_0^{\delta_1} C(u_1, u_2) dC(u_1, u_2) - 1,$$

$$\tau^{upper}(U_1, U_2 | U_1 > \delta_1, U_2 > \delta_2) = \frac{4}{\bar{C}(\delta_1, \delta_2)^2} \int_{\delta_2}^1 \int_{\delta_1}^1 \bar{C}(u_1, u_2) \, dC(u_1, u_2) - 1 \tag{3.5}$$

$$= \frac{4}{[1 - u_1 - u_2 + C(\delta_1, \delta_2)]^2} \int_{\delta_2}^1 \int_{\delta_1}^1 [1 - u_1 - u_2 + C(u_1, u_2)] \, dC(u_1, u_2) - 1,$$

where $\bar{C}$ is the corresponding joint survival function as defined in (2.1). Similar expressions can also be derived for the lower and upper exceedance Spearman's ρ. In most cases, explicit solutions of the integrals in (3.5) are though hard to obtain. An exception is the lower exceedance Kendall's τ of the Clayton copula with parameter θ (cp. Section 2.1.3):

$$\tau^{lower}(U_1, U_2 | U_1 \leq \delta_1, U_2 \leq \delta_2) = \frac{4}{(\delta_1^{-\theta} + \delta_2^{-\theta} - 1)^{-\frac{2}{\theta}}} \frac{(1 + \theta)(\delta_1^{-\theta} + \delta_2^{-\theta} - 1)^{-\frac{2}{\theta}}}{2(2 + \theta)} - 1$$

$$= \frac{4(1 + \theta)}{2(2 + \theta)} - 1 = \frac{2}{2 + \theta} = \tau(U_1, U_2).$$

	lower exceedance dependence					
τ	N	t	C	G	F	J
0.1	0.023	0.162	0.100	0.014	0.005	0.001
0.2	0.053	0.189	0.200	0.033	0.013	0.003
0.3	0.091	0.223	0.300	0.058	0.026	0.006
0.4	0.141	0.266	0.400	0.092	0.043	0.010
0.5	0.206	0.321	0.500	0.139	0.069	0.018
0.6	0.290	0.392	0.600	0.204	0.110	0.033
0.7	0.400	0.483	0.700	0.298	0.177	0.063
0.8	0.547	0.604	0.800	0.438	0.302	0.132
0.9	0.743	0.769	0.900	0.656	0.558	0.326

	upper exceedance dependence					
τ	N	t	C	G	F	J
0.1	0.023	0.162	0.001	0.138	0.005	0.191
0.2	0.053	0.189	0.002	0.211	0.013	0.282
0.3	0.091	0.223	0.005	0.273	0.026	0.361
0.4	0.141	0.266	0.010	0.337	0.043	0.439
0.5	0.206	0.321	0.019	0.407	0.069	0.523
0.6	0.290	0.392	0.035	0.487	0.110	0.612
0.7	0.400	0.483	0.066	0.582	0.177	0.702
0.8	0.547	0.604	0.136	0.695	0.302	0.800
0.9	0.743	0.769	0.342	0.832	0.558	0.896

Table 3.2: Theoretical lower and upper exceedance Kendall's τ's (τ^{lower} and τ^{upper}) with thresholds $\delta_1 = \delta_2 = 0.2$ for lower and $\delta_1 = \delta_2 = 0.8$ for upper dependence, respectively. Copula parameters are chosen according to Kendall's τ's between 0.1 and 0.9 (cp. Table 2.1).

This result ($\tau^{lower} = \tau$) is however an exception as shown in Table 3.2 which displays theoretical lower and upper exceedance Kendall's τ's (mainly obtained by numerical integration) for some common bivariate copula families (Gaussian (N), t with four degrees of freedom, Clayton (C), Gumbel (G), Frank (F), Joe (J)) and $\delta_1 = \delta_2 = 0.2$ for lower and $\delta_1 = \delta_2 = 0.8$ for upper dependence, respectively. Parameters of the copulas are chosen according to eight different choices of Kendall's τ.

Obviously, exceedance Kendall's τ is able to discriminate between pairs of random variables that exhibit strong joint tail behavior and those that do not. E.g., the Clayton copula with lower tail dependence has high lower exceedance dependence, while the Gumbel and the Joe copula have high upper exceedance dependence. The symmetric tail dependence of t copulas is also clearly reflected in the corresponding exceedance dependence. However, there are again some problems in discriminating Gaussian and t copulas, but, as we will see in the following, in contrast to tail dependence, the theoretical and empirical values agree when working with exceedance dependence.
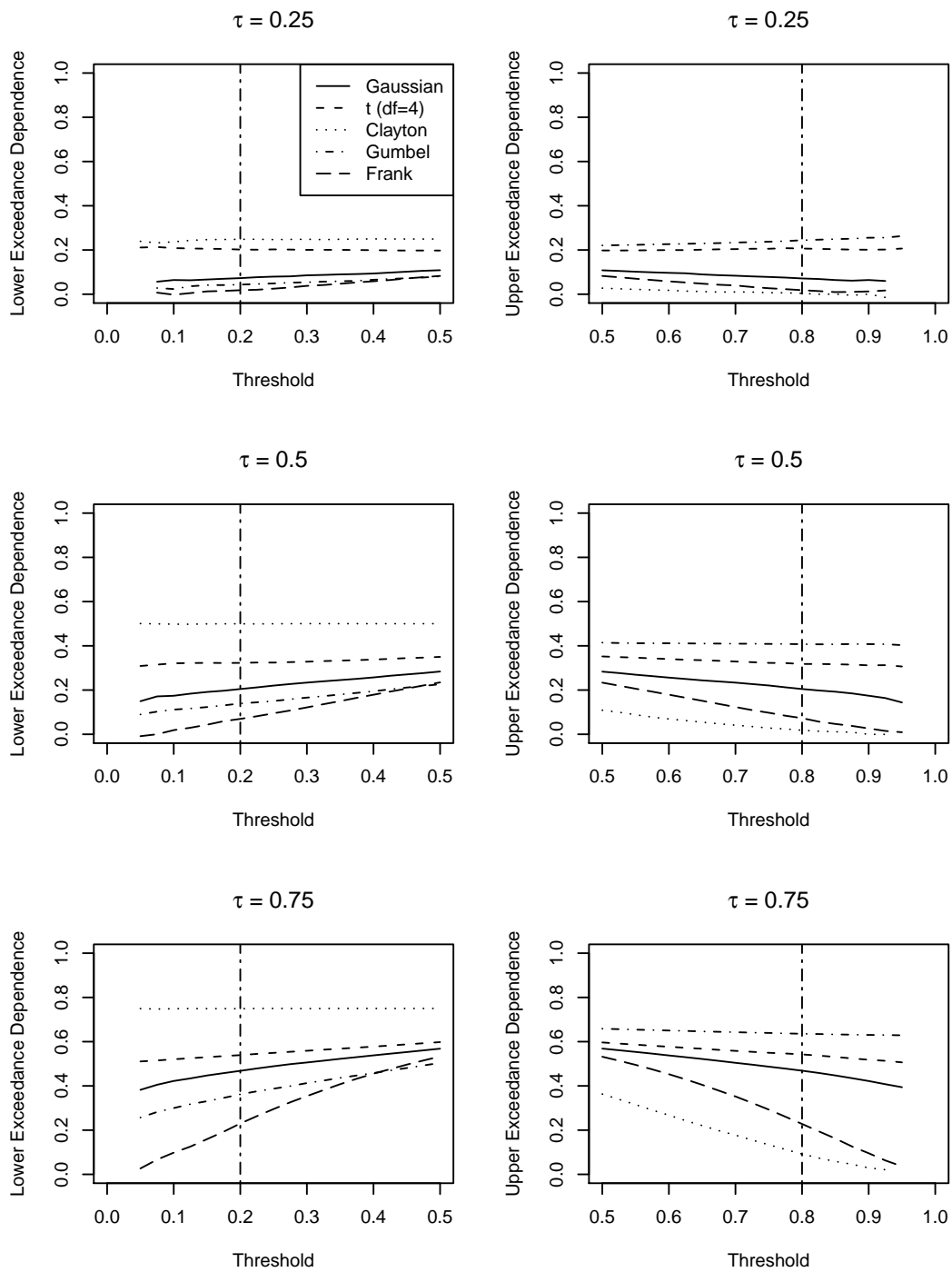
Figure 3.2: Estimated lower and upper exceedance Kendall's τ $\hat{\tau}_n^{lower}$ (left column) and $\hat{\tau}_n^{upper}$ (right column), respectively, for different thresholds and with three different choices of Kendall's τ to determine the copula parameters (cp. Table 2.1; $\tau = 0.25$ in the top row, $\tau = 0.5$ in the middle row, $\tau = 0.75$ in the bottom row). The thresholds of $\delta_1 = \delta_2 = 0.2$ for lower and $\delta_1 = \delta_2 = 0.8$ for upper dependence are indicated by vertical lines.

τ	Gaussian	t (df = 4)	Clayton	Gumbel	Frank
0.25	7.44%	8.00%	9.37%	6.68%	7.22%
0.5	11.40%	11.78%	14.29%	10.27%	10.95%
0.75	15.63%	15.83%	17.82%	14.75%	15.30%

Table 3.3: Theoretical number of observations used in the estimation of lower exceedance dependence with threshold $\delta_1 = \delta_2 = 0.2$ and with copula parameters given according to three choices of Kendall's τ (cp. Table 2.1).

Estimators are easily obtained by using the empirical versions $\hat{\tau}_n$ and $\hat{\rho}_n$ of Kendall's τ and Spearman's ρ, respectively (see (3.1)). Note that these exceedance measures critically depend on the thresholds δ_1 and δ_2, because they determine how many observations are used in the estimation. In our applications we always choose $\delta_1 = \delta_2$, since other choices require some specific a priori knowledge about the data which we do not have in general.

To investigate the empirical versions of exceedance dependence, we performed a Monte Carlo study as before: we simulated $n = 1000$ observations from five bivariate copula families with three different choices of Kendall's τ and computed the lower and upper exceedance Kendall's τ. The results of $R = 1000$ simulations are shown in Figure 3.2.

Obviously, the empirical exceedance Kendall's τ's agree with the theoretical values at $\delta_1 = \delta_2 = 0.2$ for lower and $\delta_1 = \delta_2 = 0.8$ for upper dependence (cp. Table 3.2). Moreover, these thresholds of 20% for the lower tail and 80% for the upper tail seem to be a good compromise between distinguishable results and a reasonable number of observations that are used in the estimation. These (theoretical) numbers of observations used in the estimation of the lower exceedance dependence with threshold 20% are displayed in the Table 3.3.

3.1.4 Tail cumulation

This dependence measure is graphically motivated. As before we concentrate on those variable pairs that show a strong comovement in the tails, i.e., in the lower-left and upper-right quadrants of $[0,1]^2$. When we consider scatter plots of the data, we are therefore mainly interested in the bottom left and right upper corner of the plot. Hence, we can determine dependence by simply comparing the observed data with data from independent observations. If two random variables U_1 and U_2 are independent and uniformly on $[0,1]$ distributed, we can easily compute $c_l = c_l(\alpha)$ such that $P(U_1 \le c_l, U_2 \le c_l) = \alpha \in [0,1]$, i.e., the boundaries of a box in the bottom left corner of the scatter plot which contains, e.g., $\alpha = 10\%$ of all observations.

$$\alpha = P(U_1 \le c_l, U_2 \le c_l) = P(U_1 \le c_l)P(U_2 \le c_l) = c_l^2 \quad \Leftrightarrow \quad c_l = \sqrt{\alpha},$$

and similarly $c_u = c_u(\alpha) = 1 - \sqrt{\alpha}$ so that $P(U_1 > c_u, U_2 > c_u) = \alpha$. If variables are dependent in the tails, we expect that more than $\alpha \times 100\%$ of the observations lie within the boxes determined by c_l and c_u, i.e., the observations are cumulated in this box. As an illustration consider Figure 3.3 which shows how dependence in the tails is measured.
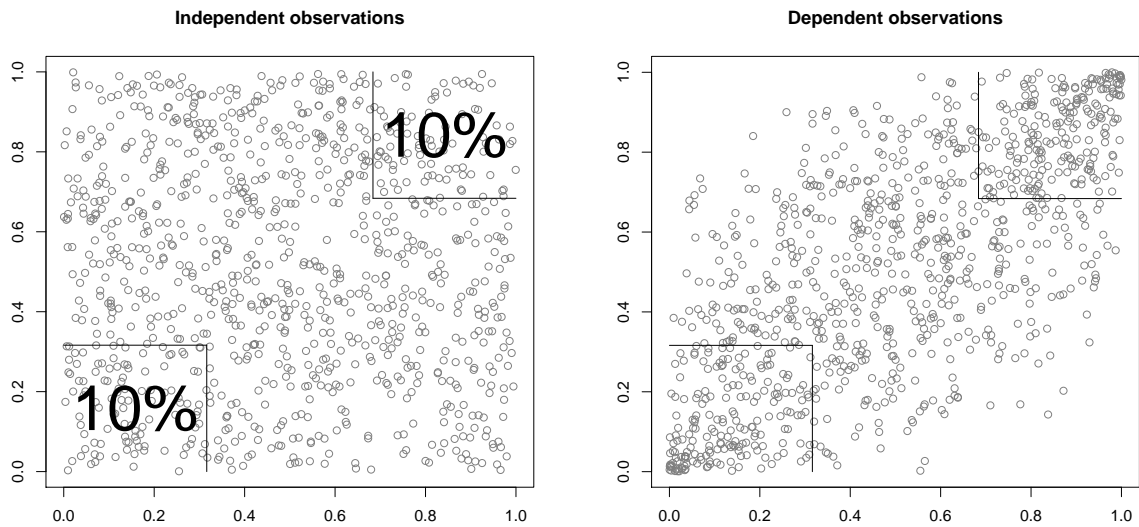
Figure 3.3: Illustration of tail cumulation.

Based on these considerations we define *lower* and *upper tail cumulation* as a graphically motivated extension of tail dependence which we considered in Section 3.1.2:

$$\hat{\gamma}_n^{lower} = \frac{\#\{\text{observations in } [0, c_l(\alpha)]^2\}}{n} - \alpha$$

$$= \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{\{U_{i1} \leq c_l(\alpha), U_{i2} \leq c_l(\alpha)\}} - \alpha,$$

$$\hat{\gamma}_n^{upper} = \frac{\#\{\text{observations in } [c_u(\alpha), 1]^2\}}{n} - \alpha$$

$$= \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{\{U_{i1} > c_u(\alpha), U_{i2} > c_u(\alpha)\}} - \alpha.$$

Positive values of $\hat{\gamma}_n^{lower}$ and $\hat{\gamma}_n^{upper}$ indicate that the random variables exhibit a certain comovement in their tails. The maximum sensible value of α is 0.25 because this means that we divide the scatter plot into four equally sized squares which, under independence, contain 25% of all observations each.

To assess the measure of tail cumulation we performed a Monte Carlo study with the same setting as for the measures of tail and exceedance dependence (see Figure 3.4). Apparently, tail cumulation is not able to clearly distinguish between variable pairs with or without strong joint tail behavior, except for the asymmetric tail dependence induced by the observations from the Clayton copula and, partly, those from the Gumbel copula for $\alpha \in [0.05, 0.15]$. However the observations generated from the t copula can barely be distinguished from those of the Gaussian copula. These results can also be inferred from the theoretical copula distribution functions $C(c, c) = P(U_1 \leq c, U_2 \leq c)$ as displayed in Figure 3.5. As a result, the measure of tail cumulation should be used carefully. It might be useful in situations when one is particularly interested in asymmetric tail dependence,
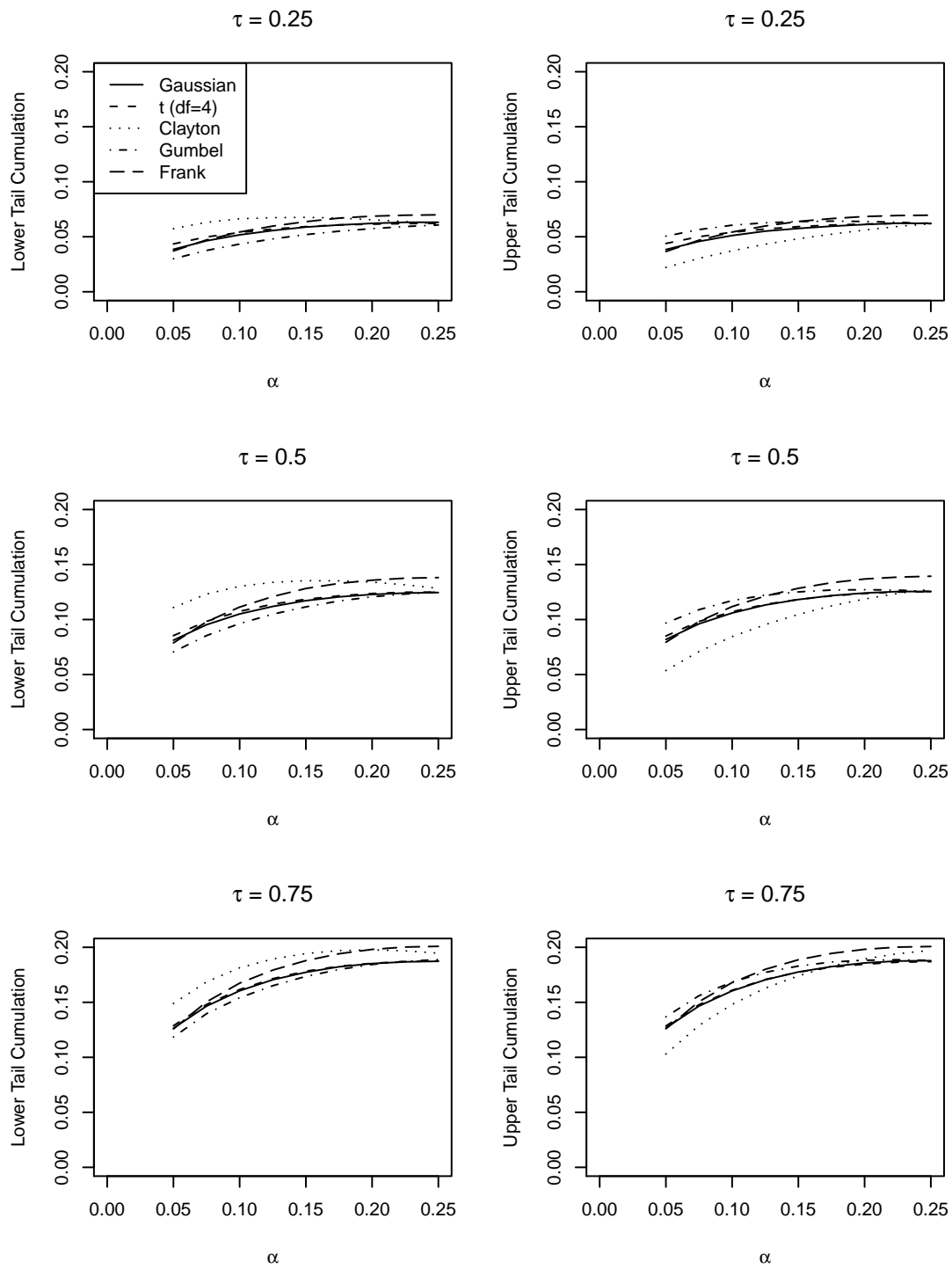
Figure 3.4: Estimated lower and upper tail cumulation $\hat{\gamma}_n^{lower}$ (left column) and $\hat{\gamma}_n^{upper}$ (right column), respectively, for different α's and with three different choices of Kendall's τ to determine the copula parameters (cp. Table 2.1; $\tau = 0.25$ in the top row, $\tau = 0.5$ in the middle row, $\tau = 0.75$ in the bottom row).
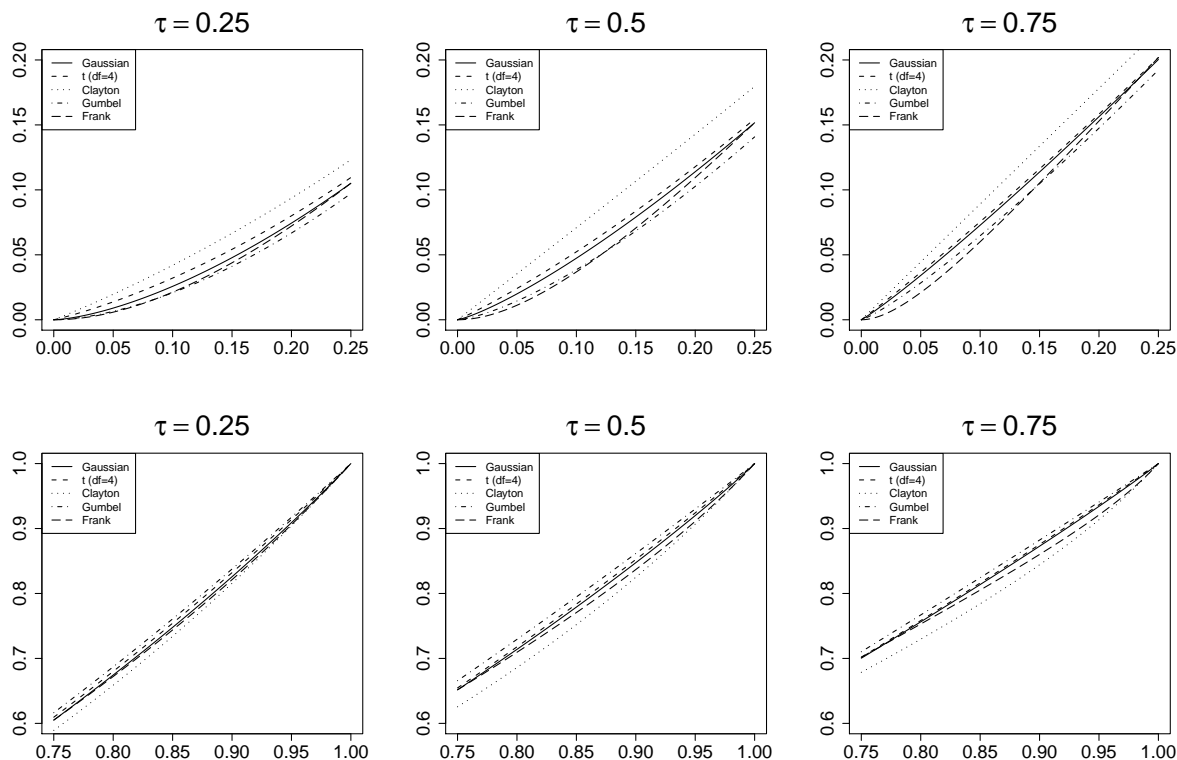
Figure 3.5: Theoretical copula distribution functions $C(c, c) = P(U_1 \leq c, U_2 \leq c)$ for five different families and three choices of Kendall's τ which determine the copula parameters (cp. Table 2.1; $\tau = 0.25$ in the left column, $\tau = 0.5$ in the middle column, $\tau = 0.75$ in the right column).

especially in lower tail dependence (as induced by the Clayton copula) which is often important in financial application. Furthermore it represents an alternative to Kendall's τ which may be insightful in certain applications due to the clear graphical interpretation. As for all measures discussed here, but for tail cumulation in particular, a high number of available observations is desirable to obtain good estimates.

3.1.5 Fréchet dependence

For every copula we know that the Fréchet-Hoeffding bounds hold (see Theorem 2.3). In the bivariate case both bounds $W(u_1, u_2) = \max(u_1 + u_2 - 1, 0)$ and $M(u_1, u_2) = \min(u_1, u_2)$ are copulas themselves. As the Fréchet-Hoeffding bounds of course also apply to the independence copula $\Pi(u_1, u_2) = u_1 u_2$, it is a reasonable idea to define the finite mixture

$$C_{\alpha,\beta}(u_1, u_2) = \alpha W(u_1, u_2) + (1 - \alpha - \beta)\Pi(u_1, u_2) + \beta M(u_1, u_2), \qquad (3.6)$$

where $\alpha, \beta \geq 0$ and $\alpha + \beta \leq 1$ (Nelsen 2006), i.e., $C_{\alpha,\beta}$ is a convex combination of W, Π and M. Moreover, $C_{\alpha,\beta}$ is again a copula and was originally considered by Fréchet (1958). It defines the most natural comprehensive copula family, where a family is called
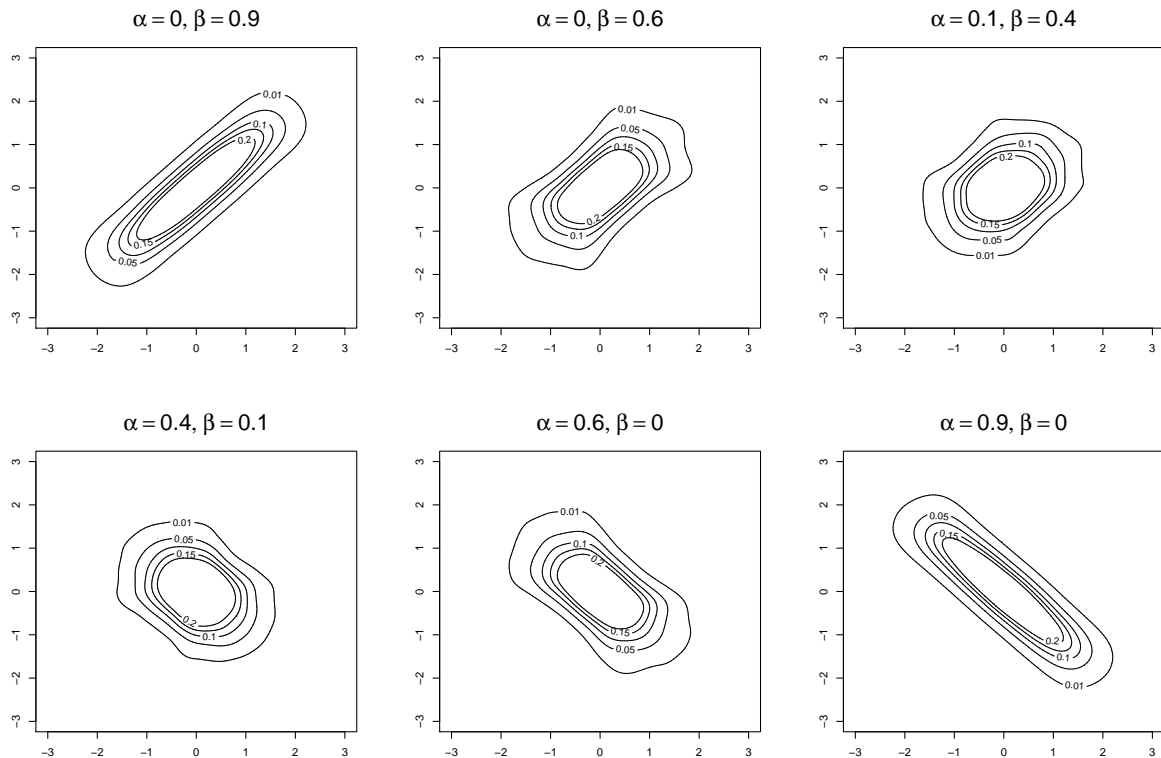
Figure 3.6: Empirical contour plots (based on 10,000 samples) of the copula $C_{\alpha,\beta}$ for different choices of α and β.

comprehensive when it includes W, Π and M. Contour plots for standard normal margins and different choices of α and β are shown in Figure 3.6. The lower and upper tail dependence of $C_{\alpha,\beta}$ is α.

Equation (3.6) motivates our definition of *negative* and *positive Fréchet dependence* α_F and β_F, respectively, as the solution of the following optimization problem which is based on a least squares approach and can be formulated as a quadratic program (see Appendix 3.3.1 of this chapter).

$$\min_{\alpha,\beta} \sum_{i=1}^{n} \Big[\alpha W(u_{i1}, u_{i2}) + (1 - \alpha - \beta)\Pi(u_{i1}, u_{i2})$$
$$+ \beta M(u_{i1}, u_{i2}) - C_n(u_{i1}, u_{i2}) \Big]^2 \tag{3.7}$$

subject to $\alpha + \beta \leq 1$, $\alpha \geq 0$, $\beta \geq 0$.

Obviously α_F is a measure of negative dependence as it reflects the "amount" of W (the copula representing perfect negative dependence, cp. Theorem 2.5) that is incorporated in the unknown copula C. Similarly, β_F measures the positive dependence.

Note that, amongst others, the Gaussian, t and Frank copula families are comprehensive, while the (rotated) Clayton and Gumbel families are not (cp. the special (limiting)

τ	Gaussian	t (df = 4)	Clayton	Gumbel	Frank
0.25	0.33 (.071)	0.33 (.069)	0.32 (.072)	0.32 (.066)	0.33 (.073)
0.5	0.62 (.072)	0.62 (.070)	0.62 (.080)	0.61 (.067)	0.63 (.071)
0.75	0.86 (.064)	0.85 (.067)	0.86 (.069)	0.85 (.063)	0.86 (.063)

Table 3.4: Estimated positive Fréchet dependence $\hat{\beta}_F$ for $R = 1000$ repetitions and three choices of Kendall's τ which determine the copula parameters (cp. Table 2.1). Standard errors are displayed in brackets.

cases of the bivariate copula families as described in Section 2.1.3). Nevertheless, even if data comes from the latter families, Fréchet dependence is an intuitive dependence measure. Table 3.4 displays the results of a Monte Carlo study with $R = 1000$ repetitions and $n = 1000$ observations from five different copula families with three choices of Kendall's τ to determine the copula parameters. Only the estimates of the positive Fréchet dependence $\hat{\beta}_F$ are displayed as we only consider positive Kendall's τ's (the estimates of the negative Fréchet dependence $\hat{\alpha}_F$ were 0 except for some very small values).

Even if the (rotated) Clayton and Gumbel families are not comprehensive and not symmetric in contrast to the quite restrictive definition of $C_{\alpha,\beta}$ in (3.6) which only allows to capture a limited amount of dependence properties (cp. Figure 3.6), the table shows that Fréchet dependence is consistently estimated across copula families. However, its values do not permit any statement about tail behavior (which is clear from the definition of $C_{\alpha,\beta}$), but Fréchet dependence is still an alternative to Kendall's τ as its interpretation is different and very intuitive. Since its values are higher than those of Kendall's τ, high dependence is more pronounced which may be interesting in certain applications.

3.1.6 Hu dependence

The derivation of Fréchet dependence is quite appealing, but the definition of the copula family $C_{\alpha,\beta}$ in (3.6) is rather restrictive with regard to dependence properties such as tail dependence. Pursuing the idea of determining dependence measures by comparing characteristics of the observed data to those of a theoretical copula family, we define an alternative mixed copula family by

$$C_\omega(u_1, u_2) = \omega_1 C_{\theta_1}^{SG}(u_1, u_2) + (1 - \omega_1 - \omega_2)C_\rho^N(u_1, u_2) + \omega_2 C_{\theta_2}^G(u_1, u_2), \qquad (3.8)$$

where C_ρ^N denotes a Gaussian copula with parameter ρ, $C_{\theta_2}^G$ a Gumbel copula with parameter θ_2 and $C_{\theta_1}^{SG}$ a Gumbel survival copula with parameter θ_1, which is defined as

$$C_{\theta_1}^{SG}(u_1, u_2) = u_1 + u_2 - 1 + C_{\theta_1}^G(1 - u_1, 1 - u_2), \qquad (3.9)$$

and exhibits lower tail dependence $\lambda^{lower} = 2 - 2^{1/\theta_1}$ but no upper tail dependence (cp. Table 2.1). Moreover, $\omega_1, \omega_2 \geq 0$ and $\omega_1 + \omega_2 \leq 1$, i.e., $C_\omega(u_1, u_2)$ is a convex combination of $C_{\theta_1}^{SG}$, C_ρ^N and $C_{\theta_2}^G$. This copula family has been considered by Hu (2006) to model the dependence structure of financial markets which usually show asymmetric tail dependencies. Contour plots for standard normal margins and different choices of ω_1 and ω_2 as well
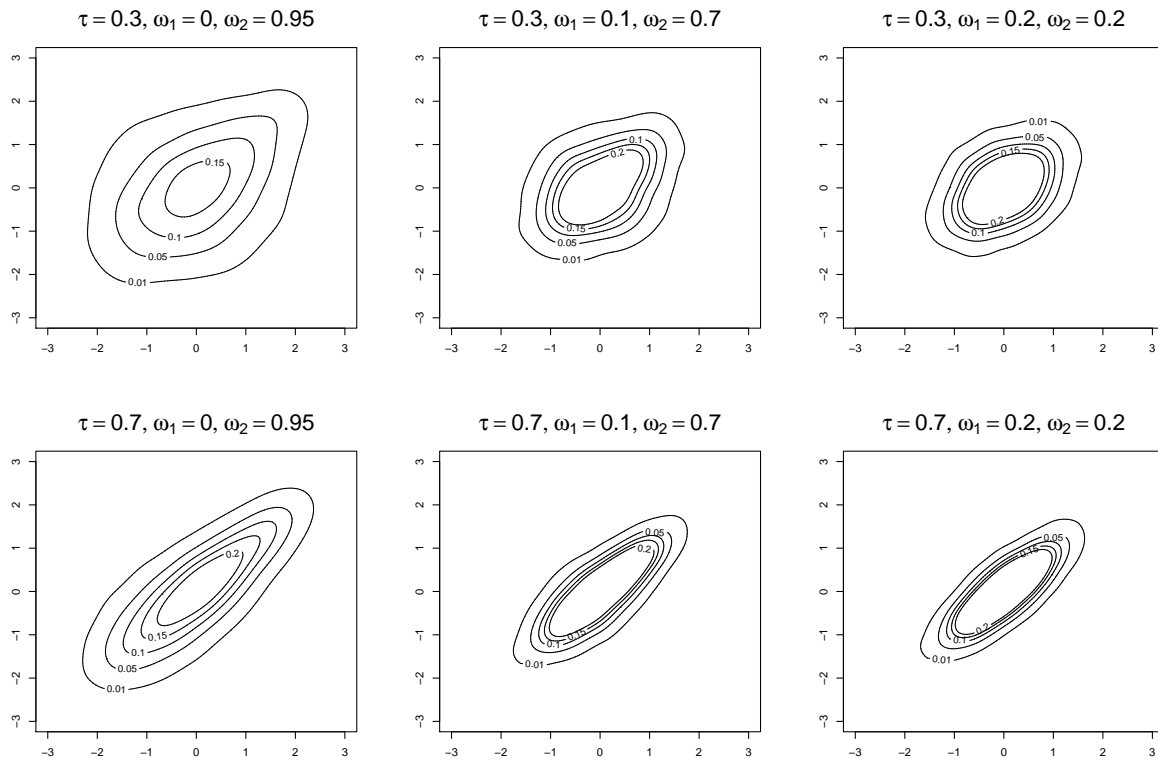
Figure 3.7: Empirical contour plots (based on 10,000 samples) of the copula C_ω for different choices of ω_1 and ω_2 as well as Kendall's τ to determine the copula parameters (cp. Table 2.1; $\tau = 0.3$ in the top row and $\tau = 0.7$ in the bottom row).

as Kendall's τ to determine the copula parameters θ_1, θ_2 and ρ according to Table 2.1 are shown in Figure 3.7 and illustrate the possible dependence properties of C_ω. (Note that copula parameters according to different choices of Kendall's τ are of course also possible but not sensible in our opinion.)

The idea is now to find the weights ω_1 and ω_2 under the assumption that our observations come from the underlying copula C_ω. Then w_1 reflects how close the unknown copula C is to the Gumbel survival copula, and w_2 the corresponding closeness to the Gumbel copula, where these copulas are chosen as proxies for lower and upper tail dependence. What remains, i.e., $1 - \omega_1 - \omega_2$, expresses the "normality" of the data. However, in contrast to the family $C_{\alpha,\beta}$ as defined in (3.6), C_ω also requires the estimation of the dependence parameters θ_1, θ_2 and ρ. To keep the computational effort rather low, we choose these parameters according to the empirical Kendall's τ and the well-known inversion formulas (cp. Table 2.1), where for the Gumbel survival copula the same inversion formula as for the Gumbel copula holds, i.e., $\hat{\theta}_1 = \hat{\theta}_2$ (also cp. Section 4.1.2). Then we define the *lower* and *upper Hu dependence* $\hat{\omega}_1$ and $\hat{\omega}_2$, respectively, as the solution of the following

τ	Hu dep.	Gaussian	t (df = 4)	Clayton	Gumbel	Frank
0.25	$\hat{\omega}_1$	0.083 (.12)	0.504 (.13)	0.932 (.04)	0.037 (.05)	0.108 (.13)
	$1 - \hat{\omega}_1 - \hat{\omega}_2$	0.836	0.000	0.000	0.052	0.787
	$\hat{\omega}_2$	0.081 (.11)	0.496 (.13)	0.066 (.04)	0.911 (.10)	0.105 (.13)
0.5	$\hat{\omega}_1$	0.050 (.07)	0.489 (.08)	0.930 (.04)	0.033 (.04)	0.267 (.11)
	$1 - \hat{\omega}_1 - \hat{\omega}_2$	0.898	0.010	0.000	0.028	0.472
	$\hat{\omega}_2$	0.052 (.08)	0.501 (.08)	0.070 (.04)	0.939 (.06)	0.261 (.11)
0.75	$\hat{\omega}_1$	0.028 (.04)	0.458 (.09)	1.000 (.00)	0.023 (.03)	0.487 (.05)
	$1 - \hat{\omega}_1 - \hat{\omega}_2$	0.946	0.084	0.000	0.022	0.025
	$\hat{\omega}_2$	0.026 (.04)	0.458 (.08)	0.000 (.00)	0.955 (.05)	0.488 (.05)

Table 3.5: Estimated lower and upper Hu dependence $\hat{\omega}_1$ and $\hat{\omega}_2$ for $R = 1000$ repetitions and three choices of Kendall's τ to determine the copula parameters (cp. Table 2.1). Standard errors are displayed in brackets.

constraint optimization problem:

$$\max_{\omega_1, \omega_2} \sum_{i=1}^{n} \log \left[\omega_1 c_{\hat{\theta}_1}^{SG}(u_{i1}, u_{i2}) + (1 - \omega_1 - \omega_2) c_{\hat{\rho}}^{N}(u_{i1}, u_{i2}) + \omega_2 c_{\hat{\theta}_2}^{G}(u_{i1}, u_{i2}) \right]$$

$$\text{subject to } \omega_1 + \omega_2 \leq 1, \ \omega_1 \geq 0, \ \omega_2 \geq 0,$$

i.e., we choose $\hat{\omega}_1$ and $\hat{\omega}_2$ so that the log likelihood of the copula density corresponding to C_ω is maximized with respect to ω_1 and ω_2. The problem can be solved using an adaptive barrier method for non-linear objective functions as described in Section 14.4 of Lange (1999), where the linear inequality constraint can be written similar to (3.13). Results of a Monte Carlo study with $R = 1000$ repetitions and $n = 1000$ observations from five different copula families with three choices of Kendall's τ to determine the copula parameters are shown in Table 3.5.

Evidently, Hu dependence of Gaussian, Clayton and Gumbel copulas is well estimated with increasing accuracy when dependence increases. Moreover, the estimates for the t copula are sensible, since it exhibits lower and upper tail dependence and hence is best approximated by a 50:50 mixture of a Gumbel and a Gumbel survival copula. The symmetry of the Frank copula is also reflected in the estimates, but these increase with higher dependency which does not correspond to the zero tail dependence of the Frank copula. The reason is that observations from Frank copulas with high dependence do not resemble the more scattered observations of Gaussian copulas (cp. Section 5.4), i.e., $\hat{\omega}_1 + \hat{\omega}_2$ can also be seen as a measure of non-normality of the observations.

An alternative measure for detecting asymmetric tail dependence could be based on the family

$$\widetilde{C}_\omega(u_1, u_2) = \omega C_{\theta_1}^{SG}(u_1, u_2) + (1 - \omega) C_{\theta_2}^{G}(u_1, u_2),$$

which corresponds to family C_ω as defined in (3.8) with $\omega_1 + \omega_2 = 1$. However, the single parameter ω only allows a statement regarding symmetric or asymmetric tail dependence but not regarding the size of tail dependence. If $\omega_1 + \omega_2 < 1$ in family C_ω, then the tail dependence of the observed data is not as strong as implied by the Gumbel (survival)

Types of tail dependence
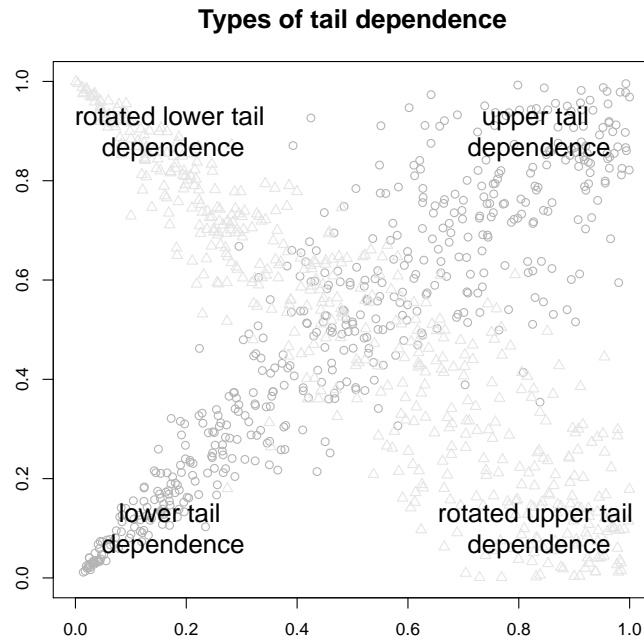
Figure 3.8: Four types of tail dependence with data generated from Clayton copulas (rotated and non-rotated) by way of illustration.

copulas, since the Gaussian copula does not exhibit tail dependence. Hence, we concentrate on the copula family C_ω which is more flexible than $\widetilde{C}_\omega$.

3.1.7 Rotated measures

Clearly tail dependence, exceedance dependence and tail cumulation, which investigate the tail behavior of variable pairs, only give sensible values for positively dependent variables. Strong joint tail behavior of negatively dependent variables, e.g., observations generated by rotated Clayton or Gumbel copulas (see Section 2.1.3), is not considered. However, the measures discussed here can easily be extended to capture these dependencies as well. We will refer to these measures as *rotated measures* correspondingly to the notation of the rotated copulas. This implies that *rotated upper tail dependence*, even if it may appear counterintuitive, is the potential comovement of variables which can be observed in the *bottom right* corner of the corresponding scatter plots, while *rotated lower tail dependence* is related to the *upper left* corner (see Figure 3.8). With this at hand, we can define the rotated versions of the measures discussed above.

- *Rotated tail dependence:* We define the rotated lower tail dependence estimator as

$$\hat{\lambda}_{n,k}^{rotated-lower} = \frac{1}{k} \sum_{i=1}^{n} \mathbf{1}_{\{R_{i1} \le \frac{n+1}{n}k, R_{i2} > n+1 - \frac{n+1}{n}k\}},$$

and the estimator of rotated upper tail dependence accordingly.

- *Rotated exceedance dependence:* The definition of an estimator for rotated lower exceedance Kendall's τ is also straightforward:

$$\hat{\tau}_n^{rotated-lower}(U_1, U_2 | U_1 \leq \theta_1, U_2 > \theta_2),$$

and similarly for Spearman's ρ and the corresponding upper versions.

- *Rotated tail cumulation:* In essentially the same way we define rotated lower tail cumulation as

$$\hat{\gamma}_n^{rotated-lower} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{\{U_{i1} \leq c_l(\alpha), U_{i2} > c_u(\alpha)\}} - \alpha,$$

and $\hat{\gamma}_n^{rotated-upper}$ accordingly.

Moreover, Hu dependence can also be extended to consider negative joint tail behavior as well. Instead of the Gumbel and the Gumbel survival copulas in Equation (3.8), we can choose the rotated Gumbel copula and an appropriately defined rotated Gumbel survival copula, i.e., the Gumbel copula rotated by 270°.

Now, the question arises how to incorporate this additional information. Rotated and non-rotated measures cannot be summarized easily, since the rotation is not unique but can be performed both clockwise and counterclockwise. Furthermore, if observations are positively dependent, there may be only a few observations in the right bottom corner and upper left corner which may lead to misleading large estimates such as -1 or 1. It is therefore unclear how to incorporate the information of rotated measures appropriately, i.e., they should be considered separately to gain additional information.

3.1.8 Implementation issues

The measures defined above allow various possible choices for the weight of a variable pair. For exceedance dependence the threshold can be chosen between 0% and 50% for lower dependence and between 50% and 100% for upper dependence, respectively, while tail cumulation permits choices of α between 0% and 25%. Sensible values can be inferred from Figures 3.2 and 3.4. Note that also the choice of k for tail dependence estimation is arbitrary as long as $k = k(n) \xrightarrow{n \to \infty} \infty$ and $\frac{k}{n} \xrightarrow{n \to \infty} 0$. Nevertheless, we think the the choice of $k = \lfloor \sqrt{n} \rfloor$ is reasonable due to its theoretic justification in Dobrić and Schmid (2005).

Measures based on Kendall's τ and Spearman's ρ should be taken as absolute values as only the absolute dependence is of interest in most cases and weights have to be positive. However, positive and negative values might also be weighted differently depending on the specific application.

Moreover, for those measures with different values for lower and upper dependence (i.e., all apart from Kendall's τ, Spearman's ρ and Fréchet dependence), one can choose, e.g.,

- either lower or upper dependence,

- the maximum of both values,

	number of values	tuning parameter	negative dep.	asymmetric dep.
Kendall's τ/Spearman's ρ	1	-	✓	-
Tail dependence	2	✓	-	✓
Exceedance dependence	2	✓	-	✓
Tail cumulation	2	✓	-	✓
Fréchet dependence	2	-	✓	-
Hu dependence	2	-	-	✓

Table 3.6: Overview of the weight measures discussed in Sections 3.1.1 to 3.1.6. As noted in Section 3.1.7, all measures can be extended to capture negative dependence as well.

- the difference of both values to pronounce asymmetric behavior, or

- a weighted sum of both values (e.g., $50 : 50$).

Additionally, rotated versions of these measures are available and can be considered as weights, too.

In the end, the choice always depends on the particular application and the preferences of the user. If tail behavior is considered, exceedance dependence is probably the best choice, since theoretical and empirical values usually agree – quite the contrary to tail dependence. The different weight measures are summarized in Table 3.6. Also note that the computation of the measures discussed here (except for Hu dependence) is faster than or equally fast as the computation of Kendall's τ's or Spearman's ρ's. This is even true for the quadratic optimization problem in the computation of Fréchet dependence, which can be solved explicitly using linear algebra (see Section 3.3.1). Hu dependence, on the other hand, does not lead to a quadratic optimization problem and therefore is computationally more complicated.

3.2   Construction methods

To construct an R-vine we are now free to choose one of the measures discussed in Section 3.1 (or any other suitable weight). The setting for the construction of the first tree T_1 is always a complete undirected graph (see Section 2.3; undirected because all dependence measures are symmetric) with the d variables as nodes and the weights w_{ij} attached to the corresponding edges e_{ij} which represent the corresponding value of the chosen measure for the variables X_i and X_j. Subsequent trees then have to be constructed under restrictions imposed from all previous trees – especially when constructing D-vines, the first tree determines all remaining trees. In the following we act under the presumption that we want to maximize the sum of weights in each tree, i.e., we want to capture as much "dependency" as possible, where "dependency" refers to the respective weights. Furthermore, we assume that $w_{ij} \in [0, 1]$ which is true for the (absolute values of the) measures defined in Section 3.1.
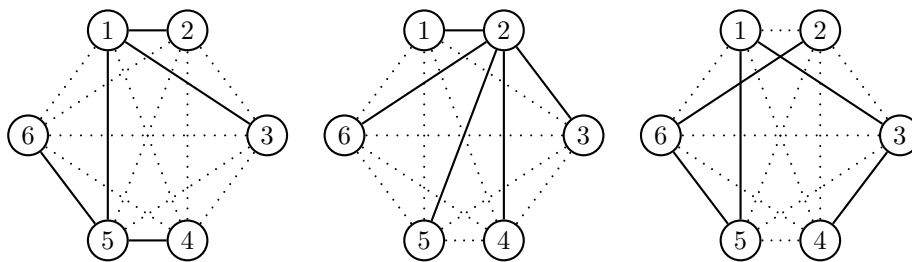
Figure 3.9: Construction of the first tree of an R-vine (left panel), a C-vine with root node 2 (middle panel) and a D-vine with path $2 - 6 - 5 - 1 - 3 - 4$ (right panel).

We start considering general R-vines and then turn to the special sub-problems of C- and D-vine construction, which correspond to finding a spanning star and a Hamiltonian path in complete graphs, respectively (see Figure 3.9), where the construction of C- and in particular R-vines has already been considered in Dißmann (2010).

3.2.1 Regular vines

As mentioned in the introduction to this chapter, we rely on a sequential method when constructing an R-vine. This means that we construct one tree per step and start with the first tree T_1. To construct this first tree, we have to find a tree that visits all nodes and maximizes the sum of the edge weights (cp. the left panel in Figure 3.9). Such a tree is called a *maximum spanning tree* (cp. Section 2.3). Usually in the literature minimization rather than maximization problems are considered and therefore we redefine our edge weights to $w'_{ij} = 1 - w_{ij}$ for weights $w_{ij} \in [0, 1]$, since

$$\min \sum_{\substack{\text{edges } e_{ij} \text{ in} \\ \text{spanning tree}}} (1 - w_{ij}) \iff \max \sum_{\substack{\text{edges } e_{ij} \text{ in} \\ \text{spanning tree}}} w_{ij}, \qquad (3.10)$$

which is due to the fact that spanning trees always have the same number of edges, namely $d - 1$ for a graph on d nodes.

Hence we want to find a *minimum spanning tree* in terms of the edge weights w'_{ij}. For this problem, there are several exact algorithms available. Here we choose the well-known algorithm of Prim (1957) which is presented in Algorithm 1 (cp., e.g., Papadimitriou and Steiglitz (1998)).

Further trees are constructed similarly. First, we construct a graph with the edges of the previous tree as nodes and edges according to the proximity condition in Definition 2.15. In general, this is not a tree (except for D-vines). Therefore we have to find a maximum spanning tree in terms of weights calculated for the respective transformed observations (2.19) and can apply Algorithm 1.

Even for simple implementations, Prim's Algorithm 1 has a time complexity of only $\mathcal{O}(|N|^2)$ (Papadimitriou and Steiglitz 1998), where N is the set of nodes. As we have to apply the algorithm at most $d - 2$ times (the graph obtained from T_{d-2} is always a tree)

Algorithm 1 Minimum spanning tree (Prim).

Input: Connected graph $G = (N, E, w)$.

1: Choose starting point $v^* \in N$. Define $\widetilde{N} := \{v^*\}$ and $\widetilde{E} = \varnothing$.

2: **while** $\widetilde{N} \neq N$ **do**

3:    Choose an edge $e = \{u, v\}$ with minimal weight w such that $u \in \tilde{N}$ and v is not.

4:    Set $\widetilde{N} := \widetilde{N} \cup \{v\}$ and $\widetilde{E} := \widetilde{E} \cup \{e\}$.

5: **end while**

Output: Minimum spanning tree $\widetilde{G} = (\widetilde{N}, \widetilde{E})$.

Algorithm 2 Root node search.

Input: Complete graph $G = (N, E, w)$ with $|N| = m$ and weight matrix $w = (w_{ij})$.

1: **for** $i = 1$ to m **do**

2:    Compute $\tilde{w}_i = \sum_{j=1}^{m} w_{ij}$.

3: **end for**

Output: Root node $v^* = \text{argmax}\{\tilde{w}_i, \ i = 1, ..., m\}$.

---

and the number of nodes is bounded by the number of variables d, the time complexity of R-vine construction with given weights is $\mathcal{O}(d^3)$.

3.2.2 Canonical vines

C-vine trees T_i are characterized by a unique root node of degree $d - i$ (see Definition 2.16). A special feature of C-vines is thus that the graph constructed from the previous tree is always complete. Hence we have to determine the root node of each tree by finding a *spanning star* (cp. Section 2.3), which maximizes the edge weights, in a complete graph (cp. the middle panel of Figure 3.9). This can be established by Algorithm 2 which is straightforward to implement. As before the first tree T_1 takes the observed data as input variables, while the subsequent trees are based on the transformed observations obtained from the previous trees (cp. (2.19)). Weights have to be calculated in each step.

The time complexity of a root node search using Algorithm 2 is obviously $\mathcal{O}(|N|^2)$ (one loop over the rows of the weight matrix and one sum over the columns). Following the same arguments as for general R-vine construction, we therefore have a time complexity of $\mathcal{O}(d^3)$ for the construction of a C-vine with given weights.

3.2.3 D-vines

In a D-vine the first tree T_1 uniquely determines all remaining trees. We are therefore only interested in the construction of this first tree and do not have to worry about all subsequent trees. However, it turns out that this construction is much harder than in the previous cases.

The problem of constructing the first D-vine tree is in fact the problem of finding a path (see Section 2.3), since in a D-vine each node in T_1 has by definition a degree of at most 2 (see Definition 2.16). We have to find the longest (in terms of the edge weights $w_{ij}, \ i, j = 1, ..., d, \ i \neq j$) sequence of the variables in which each variable occurs only
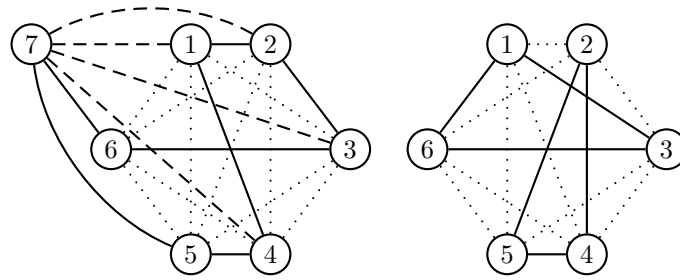
Figure 3.10: Left panel: complete graph with dummy node 7 and Hamiltonian cycle, which yields the Hamiltonian path $5 - 4 - 1 - 2 - 3 - 6$ (left panel). Right panel: two disjoint cycles ("subtours").

once, i.e., the *longest Hamiltonian path* (cp. Section 2.3; e.g., $2 - 6 - 5 - 1 - 3 - 4$ as in the right panel of Figure 3.9). Usually, in the literature such problems are considered as minimization rather than maximization problems. Therefore we proceed as before and redefine our edge weights as $w'_{ij} = 1 - w_{ij}$, which is justified in (3.10), where we now consider the edges on a Hamiltonian path instead of a spanning tree. (Here, the same reasoning as before holds, since Hamiltonian paths always have the same number of edges.)

We are then looking for the shortest Hamiltonian path in terms of the edge weights w'_{ij} and can transform the problem into a *Traveling Salesman Problem (TSP)*, i.e., the search for the shortest cycle ("tour") that visits each node exactly once (the *shortest Hamiltonian cycle*), by adding a "dummy node" $d + 1$ with weight zero on all edges to other nodes ($w'_{i,d+1} = 0 \; \forall i = 1, ..., d$) as described in Garfinkel (1986). When deleting this dummy node from a tour, we obtain the shortest Hamiltonian path without changing the length of the solution (cp. left panel of Figure 3.10).

We can formulate this extended problem as the following optimization problem (cp., e.g., Garfinkel (1986)):

$$
\min_X \sum_{i=1}^{d+1} \sum_{j=1}^{d+1} w'_{ij} x_{ij}
$$

$$
\text{subject to } \sum_{i=1}^{d+1} x_{ij} = 1, \; i = 1, ..., d + 1,
$$

$$
\sum_{j=1}^{d+1} x_{ij} = 1, \; j = 1, ..., d + 1,
$$

$$
x_{ij} \in \{0, 1\}.
$$

(3.11)

The first two constraints ensure that each node occurs in the cycle only once. Additionally so-called *subtour elimination constraints* are necessary so that we do not obtain a solution with disjoint cycles such as $1 - 3 - 6 - 1$ and $2 - 4 - 5 - 2$ in the right panel of Figure 3.10. The solution matrix $X = (x_{ij})_{i,j=1,...,d+1}$ represents a tour and can be read as follows:
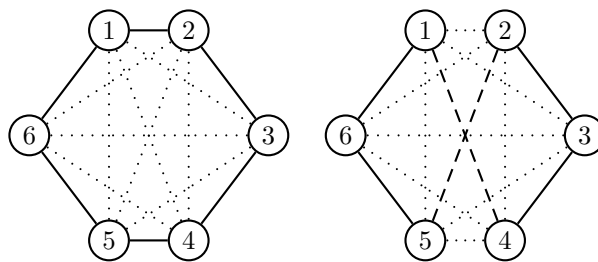
Figure 3.11: A 2-Opt step: complete graph with Hamiltonian cycle (left panel) and flipped edges (right panel).

$x_{ij} = 1$ if and only if edge e_{ij} is included in the cycle, i.e., variables X_i and X_j are directly connected.

The TSP formulated in (3.11) is *symmetric* ($w'_{ij} = w'_{ji}$) and *non-metric*, since, in contrast to the classical TSP where the weights are Euclidean distances between cities, the triangle inequality ($w'_{ij} \leq w'_{ik} + w'_{kj}$ for $k \neq i, j$) does not hold for the edge weights (the dependency between variables X_i and X_j does not allow such a statement regarding the dependencies between X_i and X_k, and X_k and X_j, respectively). Unfortunately, solving the TSP is an NP-equivalent problem (cp., e.g., Reinelt (1994)), i.e., no fast solution is known. Just computing all possible Hamiltonian paths and then choosing the best one is not feasible as there are $\frac{d!}{2}$ different D-vines in d dimensions. E.g., for $d = 10$ there are already $1,814,400$ possible D-vines!

As a result, we rely on heuristics to find a solution. There is a wide range of algorithms in the literature (see, e.g., Reinelt (1994)), and broadly, there are a two different types: constructive heuristics such as the Nearest-Neighbor algorithm (a greedy algorithm) and improvement heuristics such as the k-Opt heuristic (see below). However, because our problem (3.11) is non-metric, no statement about the approximation quality of the heuristics is possible.

In the following we will use the intuitively appealing Chained-Lin-Kernighan heuristic which is one of the best heuristics currently available (Applegate et al. 2006). This heuristic is based on the work by Lin and Kernighan (1973) who proposed a variable k-Opt heuristic in order to efficiently obtain near-optimal tours. Relaxing the TSP challenge by searching for tours, which are perhaps not optimal, increases the efficiency and is reasonable for D-vine construction, since the construction approach itself by maximizing the sum of weights is an approximation. Moreover, Lin and Kernighan (1973) state that "the procedure produces optimum solutions for all problems tested [...] up to 110 cities" and "the probability of obtaining optimum solutions in a single trial is close to 1 for small-to-medium problems, say up to the 42-city problem" which is certainly also true for the number of variables considered in D-vine construction so far.

A k-Opt heuristic iteratively exchanges ("flips") k edges of a given tour to search for a possible improvement (see Figure 3.11). Instead of choosing a fixed k (typically $k = 2$ or 3; k much larger than 3 is impractical), Lin and Kernighan (1973) described an efficient algorithm to tentatively perform a (possibly very large) number of flips. If this leads to an

improvement of the given tour, the flips are performed. Otherwise they are discarded and not considered again in any sequence. This procedure is then repeated for each possible starting point. As described above this is in fact an improvement method, but random tours can also be used for initialization of the algorithm. Other choices for the initial tour are considered, e.g., in Applegate et al. (2006) who also state that for smaller problems (which applies to our D-vine construction problem) "random tours perform nearly as well as any other choice".

This algorithm however means iteratively restarting from scratch on new tours. Martin et al. (1991) therefore proposed a chained version of the algorithm – the Chained-Lin-Kernighan heuristic – which slightly perturbs ("kicks") a previously optimized tour to leave local optima. We perform the kick if it improves the tour, otherwise we discard it and try another kick. This approach can improve the performance significantly and conveniently solve problems of dimension up to 200 (Applegate et al. 2006).

For the construction of the first D-vine tree we use the Chained-Lin-Kernighan implementation of the *Concorde* TSP code which has been applied for the record-breaking solutions of very large problems in recent years (cp. Applegate et al. (2006)).

3.3   Appendix: Quadratic programming

A quadratic program is a specific optimization problem with quadratic objective function, which is to be optimized, and linear constraints. It can be formulated as follows (cp. Nocedal and Wright (2006)):

$$\min_{\boldsymbol{x}} \frac{1}{2}\boldsymbol{x}'G\boldsymbol{x} + \boldsymbol{x}'\boldsymbol{d}$$
$$\text{subject to } A'\boldsymbol{x} \geq \boldsymbol{b}, \tag{3.12}$$

where $G \in \mathbb{R}^{n \times n}$ symmetric and $\boldsymbol{d} \in \mathbb{R}^n$ determine the objective function in terms of $\boldsymbol{x} \in \mathbb{R}^n$, while $A \in \mathbb{R}^{n \times m}$ and $\boldsymbol{b} \in \mathbb{R}^m$ define the constraints. Maximization problems can be considered as well. If $G = 0$, then we have a linear program.

Such quadratic programs can always be solved in a finite number of iterations (or are infeasible), but the efficiency depends of course on the structure of G and the number of constraints. If G is positive definite, then (3.12) is called *strictly convex* and is rather easy to solve (Nocedal and Wright 2006), e.g., using the efficient dual algorithm by Goldfarb and Idnani (1983). If G is however indefinite, the problem can be much more challenging.

3.3.1   Fréchet dependence

We now want to write the optimization problem (3.7), which we need to solve to determine the negative and positive Fréchet dependence, as a quadratic optimization problem (3.12) so that we can apply a suitable algorithm. First, we start with the constraints which can easily be transformed in the appropriate form.

$$\left.\begin{array}{r} \alpha + \beta \leq 1, \\ \alpha \geq 0, \\ \beta \geq 0 \end{array}\right\} \Leftrightarrow A' \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \geq \boldsymbol{b} \text{ with } \boldsymbol{b} = \begin{pmatrix} -1 \\ 0 \\ 0 \end{pmatrix} \text{ and } A = \begin{pmatrix} -1 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix} \tag{3.13}$$

Next, we turn to the formulation of the objective function, which is more complicated. The objective function is given in (3.7) as

$$g(\alpha, \beta) := \sum_{i=1}^{n} [\alpha W(u_{i1}, u_{i2}) + (1 - \alpha - \beta) \Pi(u_{i1}, u_{i2}) + \beta M(u_{i1}, u_{i2}) - C_n(u_{i1}, u_{i2})]^2,$$

and we want to write it as $g(\alpha, \beta) = \frac{1}{2} \boldsymbol{x}' G \boldsymbol{x} + \boldsymbol{x}' \boldsymbol{d} + c$ with suitably chosen G, $\boldsymbol{d}$ and c, where $\boldsymbol{x} = (\alpha, \beta)'$ and c is a constant independent of $\boldsymbol{x}$ that can be ignored in the optimization. In the following we will write $W_i := W(u_{i1}, u_{i2})$, $\Pi_i := \Pi(u_{i1}, u_{i2})$, $M_i := M(u_{i1}, u_{i2})$ and $C_i := C_n(u_{i1}, u_{i2})$ to simplify notation.

The first step consists of expanding the quadratic formula and arranging all terms with respect to α^2, α, β^2, β and the mixed term $\alpha\beta$. We then obtain

$$\begin{aligned} g(\alpha, \beta) = \sum_{i=1}^{n} \Big[&\alpha^2 \left(W_i^2 + \Pi_i^2 - 2W_i\Pi_i \right) + 2\alpha\beta \left(\Pi_i^2 - W_i\Pi_i + W_iM_i - \Pi_iM_i \right) \\ &+ \beta^2 \left(\Pi_i^2 + M_i^2 - 2\Pi_iM_i \right) + 2\alpha \left(-\Pi_i^2 + W_i\Pi_i - W_iC_i + \Pi_iC_i \right) \\ &+ 2\beta \left(-\Pi_i^2 + \Pi_iM_i - M_iC_i + \Pi_iC_i \right) + C_i^2 - 2\Pi_iC_i \Big]. \end{aligned}$$

Hence we define $G = (g_{ij})_{i,j=1,2}$, $\boldsymbol{d} = (d_1, d_2)'$ and c as

$$g_{11} = 2 \sum_{i=1}^{n} \left(\Pi_i - W_i \right)^2,$$

$$g_{22} = 2 \sum_{i=1}^{n} \left(\Pi_i - M_i \right)^2,$$

$$g_{12} = g_{21} = 2 \sum_{i=1}^{n} \left(\Pi_i - W_i \right) \left(\Pi_i - M_i \right),$$

$$d_1 = 2 \sum_{i=1}^{n} \left(\Pi_i - W_i \right) \left(C_i - \Pi_i \right),$$

$$d_2 = 2 \sum_{i=1}^{n} \left(\Pi_i - M_i \right) \left(C_i - \Pi_i \right),$$

$$c = \sum_{i=1}^{n} \left(C_i^2 - 2\Pi_iC_i \right).$$

Then we can write $g(\alpha, \beta) = \frac{1}{2} \boldsymbol{x}' G \boldsymbol{x} + \boldsymbol{x}' \boldsymbol{d} + c$ and solve the minimization problem given in (3.7) with respect to $\boldsymbol{x} = (\alpha, \beta)'$ using quadratic programming techniques. In order to apply efficient algorithms such as the one by Goldfarb and Idnani (1983) we have to show that G is positive definite. According to Sylvester's criterion, this is the case if and only if all of the leading principal minors are positive. Obviously, $g_{11} > 0$, since $W_i \neq \Pi_i$ for for copula data $u_{i1}, u_{i2} \in (0, 1)$. Moreover, $|G| = g_{11}g_{22} - g_{12}^2 > 0$ follows by the Cauchy-Schwarz inequality:

$$\left[\sum_{i=1}^{n} \left(\Pi_i - W_i \right)^2 \right] \left[\sum_{i=1}^{n} \left(\Pi_i - M_i \right)^2 \right] \geq \left[\sum_{i=1}^{n} \left(\Pi_i - W_i \right) \left(\Pi_i - M_i \right) \right]^2,$$

where the inequality is strict, since $(W_i - \Pi_i)_{i=1,\dots,n}$ and $(\Pi_i - M_i)_{i=1,\dots,n}$ are not linearly dependent ($W_i \neq M_i$ for copula data $u_{i1}, u_{i2} \in (0, 1)$). Thus, G is positive definite and we can apply efficient algorithms to solve (3.7).

Chapter 4

Goodness-of-fit tests

Recently, attention is being increasingly paid to goodness-of-fit tests of copulas. When estimating copula parameters one assumes that the unknown copula C belongs to a specific parametric copula family $\mathcal{C} = \{C_{\boldsymbol{\theta}} : \boldsymbol{\theta} \in \Theta\}$, where Θ is an open subset of $\mathbb{R}^p$ for some integer $p \geq 1$. Goodness-of-fit tests allow to assess whether C actually belongs to this chosen copula family or not, i.e., we want to test

$$H_0 : C \in \mathcal{C} \qquad \text{against} \qquad H_1 : C \notin \mathcal{C}. \tag{4.1}$$

There are however only few general guidelines of how to conduct this test. Here we will concentrate on a testing procedure based on the empirical copula process as suggested by Fermanian (2005), Quessy (2005) and Genest and Rémillard (2008) and which has turned out to be one of the most powerful goodness-of-fit tests in simulation studies (Berg (2009), Genest et al. (2009)). In the special case of testing the multivariate Gaussian copula against heavy tails, this test though performs rather weakly and hence we present an alternative test based on the Rosenblatt transformation which performs very well in this particular case (Berg 2009).

Before presenting these tests, we briefly review univariate goodness-of-fit tests based on the Kolmogorov-Smirnov and Cramér-von Mises statistics, since we will reduce the multivariate problem of copula goodness-of-fit testing to a univariate problem which is the fundamental approach taken in most of the copula goodness-of-fit literature. Further, we consider the most common estimation methods of copula parameters because testing (4.1) involves the estimation of $\boldsymbol{\theta}$ and the asymptotic behavior of our test statistics relies on the chosen parameter estimation method. Then we describe the above mentioned goodness-of-fit tests and show how to calculate p-values for the test based on the empirical copula process using the only recently proposed multiplier approach by Kojadinovic et al. (2010) and Kojadinovic and Yan (2010a). Finally, we also discuss multivariate, and in particular bivariate, tests of independence, where the general multivariate tests are also based on the empirical copula process as proposed by Genest and Rémillard (2004) and Schmid and Schmidt (2007).

4.1 Preliminaries

4.1.1 Univariate goodness-of-fit tests

In this section we mainly follow Chapter 14 of Lehmann and Romano (2004). Suppose that we want to test whether a set of observations $x_1, ..., x_n$ is an i.i.d. sample of a specific distribution with cdf F_0. By the Glivenko-Cantelli theorem we know that the empirical cdf $\hat{F}_n$ uniformly tends to F almost surely, where F is the cdf of the unknown underlying distribution. A common approach to testing the null hypothesis $H_0 : F = F_0$ is therefore to consider some kind of distance d between $\hat{F}_n$ and F_0 leading to a test statistic $d(\hat{F}_n, F_0)$. Popular choices are the Kolmogorov-Smirnov and the Cramér-von Mises statistics.

The *Kolmogorov-Smirnov statistic* is deduced directly from the Glivenko-Cantelli theorem and hence defined as

$$D_n = \sqrt{n} \sup_{x \in \mathbb{R}} |\hat{F}_n(x) - F_0(x)|. \tag{4.2}$$

An alternative test which is often more powerful (see Durbin and Knott (1972) and Stephens (1974)) is based on the *Cramér-von Mises statistic*

$$W_n = n \int_{-\infty}^{\infty} \left[\hat{F}_n(x) - F_0(x) \right]^2 dF_0(x). \tag{4.3}$$

If D_n or W_n are too large, the null hypothesis that the data comes from F_0 can be rejected. Critical values are tabulated.

4.1.2 Estimation of copula parameters

Before we turn to the problem of copula goodness-of-fit testing, we consider different parameter estimation methods as we have to estimate $\boldsymbol{\theta}$ for testing (4.1). The chosen estimation method decisively influences the asymptotic behavior of the goodness-of-fit test statistic.

Parameter estimation methods for copulas have already been discussed extensively (see, e.g., Genest et al. (1995), Joe (1997) and Genest and Favre (2007)). The main difference of these methods is whether parametric assumptions about the unknown margins are made or not.

(i) *Inversion of Kendall's τ or Spearman's ρ:* For many common parametric bivariate copula families closed-form expressions of Kendall's τ or Spearman's ρ in terms of the unknown parameter(s) are available (cp. Table 2.1). These can be inverted to obtain a simple estimate, which is consistent under suitable regularity conditions (Genest and Favre 2007). For elliptical copulas in higher dimensions pairwise Kendall's τ's or Spearman's ρ's can be used to efficiently estimate the parameters of the correlation matrix.

(ii) *Maximum likelihood (ML):* When the margins are known to belong to a specific parametric family, i.e., $F_j \in \mathcal{F}_j = \{F_{\gamma_j} : \gamma_j \in \Gamma_j\}$, $j = 1, ..., d$, we can simply

proceed by using standard maximum likelihood estimation and maximize

$$\ell^{ML}(\boldsymbol{\theta}, \boldsymbol{\gamma}) = \sum_{i=1}^{n} \log \left[c_{\boldsymbol{\theta}}(F_{\gamma_1}(X_{i1}), ..., F_{\gamma_d}(X_{id})) \right],$$

where $\boldsymbol{\gamma} = (\gamma_1, ..., \gamma_d)'$ and $\mathbf{X}_i = (X_{i1}, ..., X_{id})$, $i = 1, ..., n$, are samples of the random vector $\mathbf{X} = (X_1, ..., X_d)$.

Since closed form estimators are not available in general, numerical techniques are needed. However, as the number of parameters increases with dimension, numerical maximization becomes increasingly difficult in higher dimensions.

(iii) *Inference functions from margins (IFM):* Joe (1997) therefore suggests a computationally more convenient method which proceeds by first separately estimating the margins and then plugging these estimates into the log likelihood

$$\ell^{IFM}(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log \left[c_{\boldsymbol{\theta}}(F_{\hat{\gamma}_1}(X_{i1}), ..., F_{\hat{\gamma}_d}(X_{id})) \right],$$

which is subsequently maximized. Although computationally more convenient, this two-stage procedure lacks efficiency in some cases (Joe 2005). Moreover, Kim et al. (2007) show that IFM and ML estimation are not robust against extreme misspecification of the margins.

(iv) *Maximum pseudo likelihood (MPL):* Since margins are practically always unknown in practice, the most natural estimation method is therefore the nonparametric pseudo likelihood maximization (see Genest et al. (1995)). The idea is to replace the unknown distribution functions F_j's by their empirical versions

$$\hat{F}_j(t) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{\{X_{ij} \leq t\}}.$$

Then we can define the so-called *pseudo-observations*

$$U_{ij} = \frac{R_{ij}}{n+1} = \frac{n}{n+1} \hat{F}_j(X_{ij}), \tag{4.4}$$

for all $i = 1, ..., n$ and $j = 1, ..., d$, where R_{ij} is the rank of X_{ij}. The scaling factor $\frac{n}{n+1}$ is used to avoid numerical problems in the boundaries of $[0, 1]^d$. Using these pseudo-observations we define the pseudo likelihood as

$$\ell^{MPL}(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log \left[c_{\boldsymbol{\theta}}(U_{i1}, ..., U_{id}) \right]$$

and maximize it with respect to $\boldsymbol{\theta}$. The asymptotic normality of the resulting estimate $\hat{\boldsymbol{\theta}}$ was shown by Genest et al. (1995), but this method is not asymptotically semi-parametric efficient (Genest and Werker 2002). However, Kim et al. (2007) show that an estimate obtained by MPL maximization is better than IFM and ML estimates in most practical situations and due to its nonparametric structure robust against misspecification of the margins.

4.2 Copula goodness-of-fit tests

According to Genest et al. (2009) copula goodness-of-fit tests in the literature can be divided into three groups:

(i) Procedures for testing specific parametric copula families such as the Normal or Clayton families.

(ii) General tests for any copula family but which involve some kind of parameter tuning or other strategic choices of smoothing parameter, weight function or kernel.

(iii) So-called *blanket tests* which are applicable to all copula structures and do not involve any preliminary strategic choices as in (ii).

Here we will concentrate on the last group and, in particular, on two blanket tests based on the empirical copula process and on Rosenblatt's transformation, since we are interested in general procedures without any limitations in its use.

4.2.1 Blanket test based on the empirical copula process

As we have seen in Section 4.1.2 estimation of copula parameters is sensitive to the choice of margins. In copula goodness-of-fit testing we are interested in the copula alone and therefore do not want to make any assumptions with respect to the marginal distributions. Hence, it is most sensible to base goodness-of-fit tests on the ranks which are the maximally invariant statistic under continuous, strictly-increasing transformations of copula components (Nelsen 2006). This implies working with the pseudo-samples $\mathbf{U}_1, ..., \mathbf{U}_n$ as defined in (4.4). These can be considered as samples from the underlying copula C. In contrast to the sample $\mathbf{X}_1, ..., \mathbf{X}_n$ they are however no longer independent because the rank transformation induces dependence. This dependence of pseudo-observations has to be taken into account, otherwise testing procedures lack power and do not hold their nominal levels (Genest et al. 2009).

The information contained in pseudo-observations $\mathbf{U}_1, ..., \mathbf{U}_n$ is naturally summarized in the corresponding *empirical copula*

$$C_n(u_1, ..., u_d) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{\{U_{i1} \leq u_1, ..., U_{id} \leq u_d\}}, \tag{4.5}$$

where $u_1, ..., u_d \in [0, 1]$ and where the bivariate version was already considered in Section 3.1.2. The empirical copula is the empirical distribution of the observed sample and was introduced by Deheuvels (1979). It is however only asymptotically a copula. Nevertheless, it is certainly the "most objective benchmark for testing (4.1)" (Genest et al. 2009) because it is entirely non-parametric. As we have seen in Section 4.1.1, natural univariate goodness-of-fit tests are based on a distance between between the empirical and the hypothesized distribution function. Therefore it is intuitively appealing to base a copula goodness-of-fit test on a distance between the empirical copula and an estimate $C_{\boldsymbol{\theta}_n}$ of the copula C under the null hypothesis, where $\boldsymbol{\theta}_n$ is an estimator of $\boldsymbol{\theta}$ based on the pseudo-observations.

Large-scale Monte Carlo studies of various copula goodness-of-fit procedures in Berg (2009) and Genest et al. (2009) show that the test based on the Cramér-von Mises test statistic (cp. (4.3))

$$S_n = n \int_{[0,1]^d} [C_n(\boldsymbol{u}) - C_{\boldsymbol{\theta}_n}(\boldsymbol{u})]^2 \, dC_n(\boldsymbol{u}) = \sum_{i=1}^n [C_n(\boldsymbol{U}_i) - C_{\boldsymbol{\theta}_n}(\boldsymbol{U}_i)]^2 \qquad (4.6)$$

performs very well (and in particular better than the corresponding Kolmogorov-Smirnov-based test statistic). It is among the most powerful procedures and has well known asymptotics.

As defined in (4.6) the test can be performed in arbitrary dimensions. However, the computational complexity increases quickly. Genest et al. (2009) therefore study bivariate copulas exclusively, while Berg (2009) and Kojadinovic and Yan (2010a) also consider dimensions $d = 4, 8$ (except for the Student copula "due to the extreme computational load") and $d = 3, 4$, respectively. Both use MPL parameter estimation (see (iv) in Section 4.1.2) in the bivariate case, but Berg (2009) limits himself by using inversion of pairwise Kendall's τ's (see (i) in Section 4.1.2) for the correlation matrix of the multivariate Gaussian copula and inversion of the average of all pairwise Kendall's τ's for Archimedean copulas in dimension $d > 2$ to reduce the computational complexity.

Before turning to an alternative goodness-of-fit test, which is based on the Rosenblatt transformation and which performs better than the test described above in certain circumstances, we turn to the computation of p-values for test (4.6). This issue is computationally quite demanding and hence we discuss an alternative to the parametric bootstrap, the so-called *multiplier* approach.

4.2.2   Computation of p-values

The limiting distribution of S_n as defined in (4.6) is unknown in practice and depends on the hypothesized copula family and its unknown parameter. P-values therefore have to be calculated using a parametric bootstrap procedure (see Algorithm 3) as described in Genest et al. (2009) under the assumption that the analytical expression of $C_{\boldsymbol{\theta}}$ is known which is the case for those copulas we are working with, otherwise a more complicated double bootstrap procedure is necessary. Its validity was established under appropriate regularity conditions by Genest and Rémillard (2008). The authors also show that estimators of $\boldsymbol{\theta}$ based on MPL or inversion of Kendall's τ as described in in Section 4.1.2 are usually adequate to be used for testing (4.1).

However, this approach comes along with a high computational cost because each iteration step requires generation of random numbers and parameter estimation. Therefore Kojadinovic et al. (2010) and Kojadinovic and Yan (2010a) propose a computationally faster approach which is based on multiplier central limit theorems.

We follow here the exposition of Kojadinovic and Yan (2010a). For this let $\boldsymbol{\theta}_n$ be the MPL estimator of $\boldsymbol{\theta}$ which is a sensible choice in arbitrary dimension as seen in Section 4.1.2. Further define $\boldsymbol{\Theta}_n = \sqrt{n}(\boldsymbol{\theta}_n - \boldsymbol{\theta})$ and $\mathbf{V}_i = (F_1(X_{i1}), ..., F_d(X_{id})), \; i = 1, ..., n$ (in contrast to the pseudo-observations which are computed using the empirical distribution functions $\hat{F}_j$ as in (4.4)). From the work of Genest et al. (1995) we have the following

---

Algorithm 3 Computation of p-values for test (4.6) (Parametric bootstrap).

1: Compute C_n as defined in (4.5) and estimate $\boldsymbol{\theta}$ with $\boldsymbol{\theta}_n = \mathcal{T}_n(\mathbf{U}_1, ..., \mathbf{U}_n)$.

2: Choose N large (typically $N = 1000$).

3: **for** $k = 1$ to N **do**

4:    Generate a random sample $\mathbf{Y}^*_{1,k}, ..., \mathbf{Y}^*_{n,k}$ from distribution $C_{\boldsymbol{\theta}_n}$ and compute their associated rank vectors $\mathbf{R}^*_{1,k}, ..., \mathbf{R}^*_{n,k}$.

5:    Compute $\mathbf{U}^*_{i,k} = \mathbf{R}^*_{i,k}/(n+1) \;\forall i = 1, ..., n$ and let

$$C^*_{n,k}(\boldsymbol{u}) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{\{\mathbf{U}^*_{i,k} \leq \boldsymbol{u}\}}, \; \boldsymbol{u} \in [0,1]^d.$$

6:    Estimate $\boldsymbol{\theta}$ by $\boldsymbol{\theta}^*_{n,k} = \mathcal{T}_n(\mathbf{U}^*_{1,k}, ..., \mathbf{U}^*_{n,k})$.

7:    Compute

$$S^*_{n,k} = \sum_{i=1}^{n} \left[C^*_{n,k}(\mathbf{U}^*_{i,k}) - C_{\boldsymbol{\theta}^*_{n,k}}(\mathbf{U}^*_{i,k}) \right]^2.$$

8: **end for**

Output: An approximate p-value is then given by $\frac{1}{N} \sum_{k=1}^{N} \mathbf{1}_{\{S^*_{n,k} > S_n\}}$.

---

asymptotic representation under regularity conditions similar to those for ML estimation:

$$\boldsymbol{\Theta}_n = \frac{1}{n} \sum_{i=1}^{n} \mathbf{J}_{\boldsymbol{\theta}}(\mathbf{V}_i) + o_P(1), \tag{4.7}$$

where $X_n = o_p(1)$ for a set of random variables $(X_n)_{n \in \mathbb{N}}$ means $\lim_{n \to \infty} P(|X_n| > \varepsilon) = 0$ $\forall \varepsilon > 0$. Moreover, $\mathbf{J}_{\boldsymbol{\theta}} : [0,1]^d \to \mathbb{R}$ is a suitable score function and $\mathbf{J}_{\boldsymbol{\theta}}(\mathbf{V}_i), \; i = 1, ..., n$, are i.i.d. with expectation 0 and finite covariance.

Now suppose that the unknown copula C belongs to the family $\mathcal{C}$ as in (4.1) and that all members of $\mathcal{C}$ have partial derivatives with respect to each component v_j. Further let $\boldsymbol{\Theta}$ denote the weak limit of $\boldsymbol{\Theta}_n$ and $\dot{C}_{\boldsymbol{\theta}} = \partial C_{\boldsymbol{\theta}}/\partial \boldsymbol{\theta}$. Then by Quessy (2005) the goodness-of-fit process $\sqrt{n}(C_n - C_{\boldsymbol{\theta}_n})$, which is the basis of the test statistic defined in (4.6), converges weakly in $\ell^\infty([0,1]^d)$, the space of bounded, real-valued function on $[0,1]^d$, to the tight centered Gaussian process

$$\mathbb{W}_{\boldsymbol{\theta}}(\boldsymbol{v}) = \mathbb{C}_{\theta}(\boldsymbol{v}) - \dot{C}_{\boldsymbol{\theta}}{}'(\boldsymbol{v})\boldsymbol{\Theta}, \; \boldsymbol{v} \in [0,1]^d, \tag{4.8}$$

where $\mathbb{C}_{\boldsymbol{\theta}}(\boldsymbol{v}) = \alpha_{\boldsymbol{\theta}}(\boldsymbol{v}) - \sum_{j=1}^{d} C_{\boldsymbol{\theta}}^{(j)}(\boldsymbol{v})\alpha_{\boldsymbol{\theta}}(1, ..., 1, v_j, 1, ..., 1)$ with $\alpha_{\boldsymbol{\theta}}$ being a tight centered Gaussian process on $[0,1]^d$ and $C_{\boldsymbol{\theta}}^{(j)}$ the partial derivatives of $C_{\boldsymbol{\theta}}$ with respect to v_j, $j = 1, .., n$.

Using results from Rémillard and Scaillet (2009), Kojadinovic et al. (2010) show how to simulate $\mathbb{C}_{\boldsymbol{\theta}}$ using the asymptotic representation (4.7). As the random vectors $\mathbf{V}_1, ..., \mathbf{V}_n$ cannot be observed directly ($F_j, \; j = 1, ..., n$, are unknown), we have to replace them by the pseudo-observations $\mathbf{U}_1, ..., \mathbf{U}_n$. When proceeding as in Genest et al. (1995) for the

---

Algorithm 4 Computation of p-values for test (4.6) (Multiplier approach).

1: Compute C_n as defined in (4.5) and estimate $\boldsymbol{\theta}$ with $\boldsymbol{\theta}_n = \mathcal{T}_n(\mathbf{U}_1, ..., \mathbf{U}_n)$.
2: Choose N large (typically $N = 1000$).
3: **for** $k = 1$ to N **do**
4:    Generate i.i.d. random variables $Z_{1,k}, ..., Z_{n,k}$ with expectation 0 and variance 1.
5:    Compute

$$S_{n,k}^* = \int_{[0,1]^d} \left[\mathbb{C}_{n,k}(\boldsymbol{u}) - \dot{C}_{\boldsymbol{\theta}_n}{}'(\boldsymbol{u})\hat{\boldsymbol{\Theta}}_{n,k} \right]^2 dC_n(\boldsymbol{u})$$

$$= \frac{1}{n} \sum_{i=1}^{n} \left[\mathbb{C}_{n,k}(\mathbf{U}_i) - \dot{C}_{\boldsymbol{\theta}_n}{}'(\mathbf{U}_i)\hat{\boldsymbol{\Theta}}_{n,k} \right]^2 .$$

6: **end for**
Output: An approximate p-value is then given by $\frac{1}{N} \sum_{k=1}^{N} \mathbf{1}_{\{S_{n,k}^* > S_n\}}$.

---

asymptotic variance of the MPL estimator, we obtain

$$\hat{\mathbf{J}}_{\boldsymbol{\theta}}(\mathbf{U}_i) = \Sigma_n^{-1} \left[\frac{\dot{c}_{\boldsymbol{\theta}_n}(\mathbf{U_i})}{c_{\boldsymbol{\theta}_n}(\mathbf{U_i})} - \frac{1}{n} \sum_{j=1}^{d} \sum_{k=1}^{n} \mathbf{1}_{\{U_{ij} \leq U_{kj}\}} \frac{c_{\boldsymbol{\theta}_n}^{(j)}(\mathbf{U_k})\dot{c}_{\boldsymbol{\theta}_n}(\mathbf{U_k})}{c_{\boldsymbol{\theta}_n}(\mathbf{U_k})^2} \right] ,$$

where Σ_n is the sample covariance matrix of $\dot{c}_{\boldsymbol{\theta}_n}(\mathbf{U_i})/c_{\boldsymbol{\theta}_n}(\mathbf{U_i})$, $i = 1, ..., n$, $(\dot{c}_{\boldsymbol{\theta}_n} = \partial c/\partial \boldsymbol{\theta}_n)$ and $c_{\boldsymbol{\theta}_n}^{(j)} = \partial c_{\boldsymbol{\theta}_n}(\boldsymbol{u})/\partial u_j$, $j = 1, ..., d$, and can define

$$\hat{\boldsymbol{\Theta}}_{n,k} = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} Z_{i,k} \hat{\mathbf{J}}_{\boldsymbol{\theta}}(\mathbf{U}_i),$$

with $Z_{i,k}$, $i = 1, ..., n$, $k = 1, ..., N$, i.i.d. random variables with expectation 0 and variance 1. Under suitable regularity conditions, it follows for a large integer N that

$$\left(\sqrt{n}(C_n - C_{\boldsymbol{\theta}_n}), \mathbb{C}_{n,1} - \dot{C}_{\boldsymbol{\theta}_n}{}'\hat{\boldsymbol{\Theta}}_{n,1}, ..., \mathbb{C}_{n,N} - \dot{C}_{\boldsymbol{\theta}_n}{}'\hat{\boldsymbol{\Theta}}_{n,N} \right)$$

converges weakly to $(\mathbb{W}_{\boldsymbol{\theta}}, \mathbb{W}_{\boldsymbol{\theta}}^{(1)}, ..., \mathbb{W}_{\boldsymbol{\theta}}^{(N)})$ in $\ell^\infty([0,1]^d)^{\otimes(N+1)}$, where $\mathbb{W}_{\boldsymbol{\theta}}^{(1)}, ..., \mathbb{W}_{\boldsymbol{\theta}}^{(N)}$ are independent copies of the process $\mathbb{W}_{\boldsymbol{\theta}}$ as defined in (4.8), while $\mathbb{C}_{n,k}$, $k = 1, ..., N$, are constructed as approximations to $\sqrt{n}(C_n - C_{\boldsymbol{\theta}})$, where $\boldsymbol{\theta}$ is the true parameter, and hence the $\mathbb{C}_{n,k}$'s need to be corrected, since $\boldsymbol{\theta}$ is unknown and estimated by $\boldsymbol{\theta}_n$ (see Kojadinovic and Yan (2010a) for more details and the precise definition of $\mathbb{C}_{n,k}$, $k = 1, ..., N$). This asymptotic result evidently motivates the name "multiplier approach" and directly leads to the procedure to compute the p-value of test (4.6) which is presented in Algorithm 4.

Compared to Algorithm 3 the procedure defined in Algorithm 4 is computationally more efficient because line 5 only consists of simple arithmetic operations. A simulation study confirmed this and showed that the p-value computation using Algorithm 4 is much faster than the bootstrap procedure, especially when using MPL estimation as presented above (Kojadinovic and Yan 2010a). Furthermore, a sample size of $n = 300$ was determined as being sufficient for an omnibus use of the approach.

4.2.3 Blanket test based on Rosenblatt's transformation

The test discussed in Section 4.2.1 has a serious drawback: Berg (2009) found that this test performs rather poorly when testing the multivariate Gaussian copula against heavy tails, while it works quite well for testing against Archimedean alternatives such as observations from Clayton or Gumbel copulas. Testing the Gaussian copula against different alternatives, especially against heavy tailed alternatives as they often occur in financial applications, will however be quite important in the following (see Section 8.1). Therefore we also consider the goodness-of-fit test proposed by Breymann et al. (2003) which performs particularly well in this special case (Berg 2009).

In order to introduce this alternative test, we have to consider *Rosenblatt's transformation*. Given a set of dependent variables with known distribution, the Rosenblatt transformation transforms them into independent uniform variables. More formally, the transformation proceeds as follows: let $\mathbf{X} = (X_1, ..., X_d)$ be a random vector with marginal distribution functions $F_1, ..., F_d$ and conditional distributions $F_{i|1:(i-1)}(X_i|X_1, ..., X_{i-1}) = P(X_i \leq x_i | X_1 \leq x_1, ..., X_{i-1} \leq x_{i-1})$ for $i = 2, ..., d$. Then

$$
\begin{aligned}
V_1 &= F_1(X_1), \\
V_2 &= F_{2|1}(X_2|X_1), \\
&\vdots \\
V_d &= F_{d|1:(d-1)}(X_d|X_1, ..., X_{d-1}),
\end{aligned}
\tag{4.9}
$$

are i.i.d. $U(0,1)$. Note however that the F_i's and the conditional distributions are usually unknown in practice and therefore have to be estimated. But this means that $\mathbf{V} = (V_1, ..., V_d)$ is not exactly i.i.d., but only close to it. Hence, tests based on the Rosenblatt transformation (4.9) also require bootstrapping to determine p-values. Moreover, (4.9) is not invariant under permutations of the variables, since there are $d!$ permutations of the d-dimensional data. Though, Berg (2009) states that "as long as the permutation is decided randomly, the results will not be influenced in any particular direction".

Breymann et al. (2003) use this Rosenblatt transformation (4.9) to construct a copula goodness-of-fit test. By applying (4.9) to the pseudo-observations $\mathbf{U_i} = (U_{i1}, ..., U_{id})$, $i = 1, ..., n$, (4.4) under the assumptions of a hypothesized copula $C_{\boldsymbol{\theta}_n}$, we obtain samples $\mathbf{V_i}$ from the independence copula Π and can define:

$$
\chi_i = \sum_{j=1}^{d} \left[\Phi^{-1}(V_{ij}) \right]^2,
\tag{4.10}
$$

which approximately follows a χ_d^2 distribution. Hence, we consider $G(u) = P(F(\chi) \leq u)$, $u \in [0,1]$, where F is the cdf of the χ_d^2 distribution and χ is the theoretical value of (4.10). Under the null hypothesis $G(u) = u$, and the corresponding empirical version is given by

$$
G_n(u) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{\{F(\chi_i^2) \leq u\}}.
$$

The corresponding Cramér-von Mises statistic (4.3) can then be constructed as

$$
\begin{aligned}
T_n &= n \int_0^1 [G_n(u) - G(u)]^2 \, dG(u) \\
&= \frac{n}{3} + \frac{n}{n+1} \sum_{i=1}^n G_n \left(\frac{i}{n+1} \right)^2 - \frac{n}{(n+1)^2} \sum_{i=1}^n (2i+1) G_n \left(\frac{i}{n+1} \right),
\end{aligned}
\tag{4.11}
$$

as shown in Berg (2009). P-values can be computed using a parametric bootstrap approach as in Section 4.2.2, which is however computationally much more efficient than the test based on the empirical copula process (4.6). Moreover, Berg (2009) shows that this test performs very well for testing the multivariate Gaussian copula against heavy tails and that the permutation of the Rosenblatt transformation (4.9) adds no variance in this case. However, the power for testing against other alternatives such as Archimedean copulas is lower than for the test discussed in Section 4.2.1, but it is improving with increasing sample size and increasing dependency.

4.3 Tests of independence

Apart from goodness-of-fit testing for specific parametric copulas it is of substantial interest to test for independence because this obviously simplifies model construction, estimation and simulation a lot. Independence tests can also be considered as copula goodness-of-fit tests where the null hypothesis is that the unknown copula C is the independence copula Π, since random variables $X_1, ..., X_d$ are independent if and only if the copula which characterizes the joint behavior of $X_1, ..., X_d$ is the independence copula (cp. Theorem 2.5), i.e., we can formulate the test of independence as

$$
H_0 : C = \Pi^d \qquad \text{against} \qquad H_1 : C \neq \Pi^d.
\tag{4.12}
$$

Therefore we present two different multivariate testing procedures which are based on the empirical copula as defined in (4.5) and which we already used to construct the test in (4.6). However, we start with the bivariate case because there is a simple test available which is based on Kendall's τ.

4.3.1 Bivariate test based on Kendall's τ

As seen in Section 2.2.1 Kendall's τ is a concordance-based measure for dependence between two random variables. If random variables X_1 and X_2 are independent, then $\tau(X_1, X_2) = 0$. This motivates the following test of bivariate independence that is based on the empirical version of Kendall's τ as defined in (3.1).

According to Genest and Favre (2007) a test of independence can now be constructed using the fact that $\hat{\tau}_n$ is asymptotically normal with zero mean and variance $2(2n + 5)/(9n(n-1))$ under H_0 (Kendall 1962). Hence we can reject H_0 at the approximate level α if

$$
\sqrt{\frac{9n(n-1)}{2(2n+5)}} \, |\hat{\tau}_n(X_1, X_2)| > \Phi^{-1} \left(1 - \frac{\alpha}{2} \right).
\tag{4.13}
$$

A similar test is also available for the empirical version of Spearman's ρ.

4.3.2 Multivariate test based on the empirical copula process

The formulation of the test in (4.12) directly leads to the idea to base a test on the process

$$\mathbb{C}_n^0(\boldsymbol{u}) = \sqrt{n}\left[C_n(\boldsymbol{u}) - \prod_{j=1}^{d} u_j\right], \ \boldsymbol{u} \in [0,1]^d, \tag{4.14}$$

which is a special case of the general copula goodness-of-fit process $\sqrt{n}(C_n - C_{\boldsymbol{\theta}_n})$ using the empirical copula as defined in (4.5) and choosing $C_{\boldsymbol{\theta}_n} = \Pi^d$.

This test was initially suggested by Deheuvels (1981) and studied in Genest and Rémillard (2004). The idea is to use the so-called *Möbius-transform* in order to decompose the process (4.14) into $2^d - d - 1$ subprocesses $\mathbb{G}_{A,n} = \mathcal{M}_A(\mathbb{C}_n^0)$, where $A \subseteq \{1,...,d\}$ with $|A| > 1$, and which are defined as

$$\mathbb{G}_{A,n}(\boldsymbol{u}) = \sum_{B \subset A}(-1)^{|A \setminus B|}\mathbb{C}_n^0(\boldsymbol{u}^B)\prod_{j \in A \setminus B} u_j$$

$$= \frac{1}{\sqrt{n}}\sum_{i=1}^{n}\prod_{j \in A}(\mathbf{1}_{\{R_{ij} \leq (n+1)u_j\}} - u_j),$$

where $\boldsymbol{u}^B = (u_1^B,...,u_d^B) \in [0,1]^d$ is given for $j = 1,...,d$ by

$$u_j^B = \begin{cases} u_j & \text{if } j \in B, \\ 1 & \text{if } j \notin B. \end{cases}$$

Under H_0 the $\mathbb{G}_{A,n}$'s converge jointly to a continuous centered Gaussian process $\mathbb{G}_A$ with the essential property that the random variables $X_1,...,X_n$ are independent if and only if $\mathbb{G}_A(\boldsymbol{u}) = 0 \ \forall \boldsymbol{u} \in [0,1]^d$ and all A as defined above. This amounts to the consideration of $2^d - d - 1$ test statistics instead of only one based on (4.14). As before we concentrate on the respective Cramér-von Mises statistic which was introduced in Section 4.1.1 and obtain

$$S_{A,n}^0 = n\int_{[0,1]^d}[\mathbb{G}_{A,n}(\boldsymbol{u})]^2\,d\boldsymbol{u}, \tag{4.15}$$

with $A \subseteq \{1,...,d\}$, $|A| > 1$. Genest and Rémillard (2004) show how to combine these $2^d - d - 1$ test statistics to obtain a combined p-value using *Fisher's approach* and how to conduct the test in practice (see Algorithm 5). In a simulation study the authors also find that this procedure leads to a powerful test.

Alternatively, we can base the test directly on $\mathbb{C}_n^0$ as defined in (4.14) and consider the Cramér-von Mises statistic

$$S_n^0 = n\int_{[0,1]^d}[\mathbb{C}_n^0(\boldsymbol{u})]^2\,d\boldsymbol{u}. \tag{4.16}$$

The validity of this test, as well as the validity of the test defined in (4.15), were shown in Genest et al. (2007) and p-values can be computed using Algorithm 3, which can be simplified significantly, since $C_{\boldsymbol{\theta}_n} \equiv \Pi^d$ and hence no estimation of the copula parameter(s)

Algorithm 5 Computation of p-values for test (4.15).

1: Choose N large (typically $N = 1000$).
2: **for** $k = 1$ to N **do**
3:     Generate random variables $U_{1,k}, ..., U_{n,k}$ from a uniform distribution on $[0,1]^d$.
4:     Compute the test statistics $S^0_{A,n,k}$, $A \subseteq \{1, ..., d\}$, $|A| > 1$ as defined in (4.15) from the sample $U_{1,k}, ..., U_{n,k}$.
5: **end for**
6: Set

$$\hat{F}_{A,n}(t) = \frac{1}{N} \sum_{k=1}^{N} \mathbf{1}_{\{S^0_{A,n,k} \leq t\}}.$$

7: **for** $k = 1$ to N **do**
8:     Compute

$$W_{n,k} = -2 \sum_{A \subseteq \{1,...,d\}, |A| > 1} \log \left[1 - \hat{F}_{A,n}(S^0_{A,n,k}) \right].$$

9: **end for**
10: Compute $W_n = -2 \sum_{A \subseteq \{1,...,d\}, |A| > 1} \log \left[1 - \hat{F}_{A,n}(S^0_{A,n}) \right]$.

Output: An approximate p-value is then given by $\frac{1}{N} \sum_{k=1}^{N} \mathbf{1}_{\{W_{n,k} > W_n\}}$.

---

is necessary. In our applications, we will use this approach for testing (4.12), because it is computationally faster (we do not have to consider $2^d - d - 1$ different test statistics which is not feasible in large dimensions d).

An alternative multivariate test, which is rather simple and easy to implement, is discussed in the following section.

4.3.3 Multivariate test based on Spearman's ρ

Quessy (2009) studied the theoretical efficiency of tests based on multivariate extensions of Spearman's ρ (2.22) as proposed by Schmid and Schmidt (2007). These considerations are motivated by the fact that $\rho(X_1, X_2) = 0$ when two random variables X_1 and X_2 are independent and by a bivariate test similar to the one based on Kendall's τ and discussed in Section 4.3.1, i.e., the null hypothesis of independence between X_1 and X_2 can be rejected if $\hat{\rho}_n(X_1, X_2)$ as defined in (3.1) is too large.

According to Schmid and Schmidt (2007) a simple multivariate extension of Spearman's ρ is given by

$$\rho_d = \frac{2}{d(d-1)} \sum_{j<k} \rho_2(C_{jk}), \tag{4.17}$$

where C_{jk} is the copula of the random variables X_j and X_k, $j, k = 1, ..., d$, $j \neq k$, and as in Theorem 2.10

$$\rho_2(f) = 12 \int_{[0,1]^2} f(u_1, u_2) du_1 du_2 - 3.$$

The dependence measure ρ_d as defined in (4.17) is then simply the mean of Spearman's ρ's between all possible variable pairs and reduces to the well-known bivariate form if

$d = 2$. An estimator of ρ_d is easily obtained by plugging in the bivariate empirical copulas as defined in (4.5), i.e.,

$$\hat{\rho}_d = \frac{2}{d(d-1)} \sum_{j<k} \rho_2(C_{n,jk}),$$

where $C_{n,jk}$ is the empirical copula of the pseudo-samples $(U_{1j}, ..., U_{nj})$ and $(U_{1k}, ..., U_{nk})$.

This motivates the construction of the following test statistic which is based on the empirical copula process for independence (4.14):

$$\begin{aligned}
T_n^0 &= \frac{2}{d(d-1)} \sum_{j<k} \sqrt{n} \left[\rho_2(C_{n,jk}) - \rho_2(\Pi) \right] \\
&= \frac{6}{d(d-1)} \frac{1}{\sqrt{n}} \sum_{j<k} \sum_{i=1}^{n} \left(4U_{ij}U_{ik} - 1 \right),
\end{aligned} \tag{4.18}$$

where Π denotes the bivariate independence copula as usual and U_{ij}, $i = 1, ..., n$, $j = 1, ..., d$, the pseudo-observations as defined in (4.4). The second equality has been shown by Quessy (2009) who recommends this test statistic for multivariate independence testing, since it exhibits a very good asymptotic efficiency in contrast to alternative tests and locally yields more powerful results than the test considered in the previous section. Since this to some extent contradicts the performance results of Genest et al. (2007), we will investigate the performance of both tests for our purposes in Chapter 10. P-values for testing the independence hypothesis (4.12) can easily be obtained by a bootstrap procedure.

4.4 Overview

Table 4.1 gives an overview of the tests discussed in this chapter, their purpose in the following and their software implementation.

	test	purpose	software impl.
copula goodness-of-fit	based on empirical copula process	pair copula selection, (test for multivariate normality)	Kojadinovic and Yan (2010b)
	based on Rosenblatt's transformation	test for multivariate normality	Berg (2009)
bivariate independence	based on Kendall's τ	pre-test for independence in pair copula selection	*own*
multivariate independence	based on empirical copula process	(test for multivariate independence)	Kojadinovic and Yan (2010b)
	based on Spearman's ρ	test for multivariate independence	*own*

Table 4.1: Copula goodness-of-fit and independence tests with purpose (in brackets if subordinated) and software implementation.

Chapter 5

Model selection

When having constructed different models based on the same data, the question arises which model is "better" in some sense. In this chapter we investigate this question in detail and show different methods to approach the problem.

First, we define the fundamental Kullback-Leibler information criterion (KLIC) which is the basis of many model selection methods. Then, we turn to two simple measures which allow an easy model comparison, namely the popular Akaike and Bayesian information criteria (AIC and BIC). Subsequently, we discuss the Vuong test, which directly compares two given models and assigns a significance level to its decision. The similarities to AIC and BIC when correcting for model complexity, i.e., the number of parameters, are also pointed out.

Finally, we consider the question of copula selection for bivariate data. This problem arises in R-vine specification when having decided which variable pairs are being modeled, i.e., after having constructed a tree as described in Chapter 3. Hence, we need efficient and reliable criteria to decide which pair copula reproduces the dependence characteristics most accurately. First, we therefore analyze the problem by considering KLIC's between bivariate copula families and the limiting cases of the BB1 and BB7 copulas. Then, from a theoretical point of view, our selection method of choice will be goodness-of-fit testing as discussed in Section 4.2, in particular the goodness-of-fit test based on the empirical copula process (Section 4.2.1). However, even in the bivariate case, this is computationally quite demanding due to the intricate p-value computation. Therefore, we develop alternative copula selection criteria and evaluate them in a simulation study.

5.1   The Kullback-Leibler information criterion

The *Kullback-Leibler information criterion (KLIC)* (Kullback and Leibler 1951) is based on the same principle as the univariate goodness-of-fit tests discussed in Section 4.1.1: it measures the distance between the true but unknown distribution and a specified, approximating model with estimate $\hat{\boldsymbol{\theta}}$ of the pseudo true value $\boldsymbol{\theta}$ (note that this is not the parameter of the true distribution). We follow here the exposition of Vuong (1989)

who defines the KLIC as

$$KLIC(h_0, f, \hat{\boldsymbol{\theta}}) := \int h_0(x) \log \left[\frac{h_0(x)}{f(x|\hat{\boldsymbol{\theta}})} \right] dx = E_0[\log h_0(X)] - E_0 \left[\log f(X|\hat{\boldsymbol{\theta}}) \right], \quad (5.1)$$

where X is a random variable following the true distribution with density $h_0(\cdot)$ and E_0 denotes the expectation with respect to this true distribution which is approximated by $f(\cdot|\hat{\boldsymbol{\theta}})$.

If we are given a collection of models, it is now natural to choose the model which minimizes the KLIC. However, since the true model $h_0(\cdot)$ is in general unknown, we usually settle for choosing the model which maximizes $E_0[\log f(X|\hat{\boldsymbol{\theta}})]$ with respect to $\hat{\boldsymbol{\theta}}$ reflecting the classical maximum likelihood principle.

5.2   Akaike and Bayesian information criteria

By consideration of a risk function in terms of the KLIC (5.1) and an extension of the maximum likelihood principle, Akaike (1973) developed a convenient model selection criterion. Given observations $x_i, \ i = 1, ..., n$, the *Akaike information criterion (AIC)* is defined as

$$AIC := -2 \sum_{i=1}^{n} \log f(x_i|\hat{\boldsymbol{\theta}}) + 2k, \quad (5.2)$$

where $\hat{\boldsymbol{\theta}}$ denotes the maximum likelihood estimate of $\boldsymbol{\theta}$ and k is the number of parameters $\boldsymbol{\theta} = (\theta_1, ..., \theta_k)'$ in the model which penalizes the log likelihood in order to avoid overfitting. E.g., the AIC of an R-vine copula model is given by

$$\begin{aligned}
AIC &= -2 \sum_{i=1}^{n} \log \prod_{\ell=1}^{d-1} \prod_{e \in E_\ell} c_{j(e),k(e)|D(e)}(F(x_{i,j(e)}|\boldsymbol{x}_{i,D(e)}), F(x_{i,k(e)}|\boldsymbol{x}_{i,D(e)})) + 2k \\
&= -2 \sum_{i=1}^{n} \sum_{\ell=1}^{d-1} \sum_{e \in E_\ell} \log \left[c_{j(e),k(e)|D(e)}(F(x_{i,j(e)}|\boldsymbol{x}_{i,D(e)}), F(x_{i,k(e)}|\boldsymbol{x}_{i,D(e)})) \right] + 2k,
\end{aligned} \quad (5.3)$$

with the density of an R-vine given in (2.27) and k denoting the number of parameters in the model.

An alternative penalty term was proposed by Schwarz (1978) in a Bayesian setting. Nevertheless, the *Bayesian information criterion (BIC)*, which is defined as

$$BIC := -2 \sum_{i=1}^{n} \log f(x_i|\hat{\boldsymbol{\theta}}) + \log(n)k, \quad (5.4)$$

is mostly used in a non-Bayesian context. Its penalty term is stronger than in the AIC and therefore leads to more parsimonious models.

Model selection using the AIC or BIC can now proceed by choosing the model which minimizes the respective criterion. Often AIC and BIC are only used when comparing nested models, where *nestedness* of models means that we have a "full" model that is

specified in terms of certain parameters, while the "reduced" model is a special case of this "full" model and parametrized in a subset of these parameters. However, Ripley (2008, pp. 34, 35) states that AIC comparisons are also feasible for non-nested models but at the expense of an increased variability of the estimated AIC difference for pairs of non-nested models ($\mathcal{O}_p(\sqrt{n})$ instead of $\mathcal{O}_p(1)$, where $X_n = \mathcal{O}_p(g(n))$ for a set of random variables $(X_n)_{n \in \mathbb{N}}$ means given $\varepsilon > 0$ there is a constant B such that $P(|X_n/g(n)| > B) < \varepsilon \; \forall n$).

5.3 The Vuong test

An alternative method for directly comparing two models is the test proposed by Vuong (1989) which exhibits similarities to the information criteria considered above. As it is a statistical test, it also assigns a significance level to its decision (or does not make a decision at all at the pre-specified level).

Suppose we are given two competing models $f_1(\cdot | \hat{\boldsymbol{\theta}}_1)$ and $f_2(\cdot | \hat{\boldsymbol{\theta}}_2)$ with maximum likelihood estimates $\hat{\boldsymbol{\theta}}_1$ and $\hat{\boldsymbol{\theta}}_2$, respectively, to consistently estimate the pseudo true value. Then the Vuong test investigates the null hypothesis that

$$
\begin{aligned}
& KLIC(h_0, f_1, \hat{\boldsymbol{\theta}}_1) = KLIC(h_0, f_2, \hat{\boldsymbol{\theta}}_2) \\
\Leftrightarrow \quad & E_0 \left[\log f_1(X | \hat{\boldsymbol{\theta}}_1) \right] = E_0 \left[\log f_2(X | \hat{\boldsymbol{\theta}}_2) \right],
\end{aligned}
\tag{5.5}
$$

i.e., that one cannot distinguish between both models. Obviously, when $E_0[\log f_1(X | \hat{\boldsymbol{\theta}}_1)] > E_0[\log f_2(X | \hat{\boldsymbol{\theta}}_2)]$, model 1 is better, and vice versa. However, the question arises whether one of the models is *significantly* better than the other. Thus we have to consider asymptotic results to construct an appropriate test statistic.

For observations x_i, $i = 1, ..., n$, define $m_i := \log \left[\frac{f_1(x_i | \hat{\boldsymbol{\theta}}_1)}{f_2(x_i | \hat{\boldsymbol{\theta}}_2)} \right]$, $i = 1, ..., n$. Under the true distribution h_0, $\boldsymbol{m} = (m_1, ..., m_n)'$ is a random vector with mean $\boldsymbol{\mu_0^m} = (\mu_1^m, ..., \mu_n^m)'$. Since $\boldsymbol{\mu_0^m}$ is unknown (h_0 is unknown), we cannot directly test whether both models are equally good in terms of their approximation of the true distribution. However, using the law of large numbers Vuong (1989) shows under reasonable assumptions that

$$
\frac{1}{n} LR_n(\hat{\boldsymbol{\theta}}_1, \hat{\boldsymbol{\theta}}_2) := \frac{1}{n} \sum_{i=1}^{n} m_i \xrightarrow[n \to \infty]{a.s.} \boldsymbol{\mu_0^m}.
$$

Defining the sample variance of LR_n as

$$
\hat{\omega}^2 := \frac{1}{n} \sum_{i=1}^{n} (m_i - \bar{m})^2,
$$

Vuong (1989) further obtains the asymptotical distribution of LR_n:

$$
\nu := \frac{LR_n(\hat{\boldsymbol{\theta}}_1, \hat{\boldsymbol{\theta}}_2)}{\sqrt{n \hat{\omega}^2}} \xrightarrow[n \to \infty]{\mathcal{D}} N(0, 1).
\tag{5.6}
$$

This readily yields an asymptotic test for model selection among nested as well as non-nested models:

$$
H_0 : \boldsymbol{\mu_0^m} = 0 \quad \text{against} \quad H_1 : \boldsymbol{\mu_0^m} \neq 0,
\tag{5.7}
$$

and reject H_0 at level α if $|\nu| > \Phi^{-1}\left(1 - \frac{\alpha}{2}\right)$. In particular, if $\nu > \Phi^{-1}\left(1 - \frac{\alpha}{2}\right)$, we prefer model 1 to model 2, since the test indicates that the KLIC with regard to model 1 is significantly smaller than the KLIC of model 2 (cp. (5.5)). Similarly, we choose model 2 if $\nu < -\Phi^{-1}\left(1 - \frac{\alpha}{2}\right)$.

5.3.1 Correction factors in the Vuong test

The test defined in (5.7) does however not take into account the possibly different number of parameters of both models. Hence the test is called *unadjusted* and Vuong (1989) gives the definition of an adjusted statistic

$$\widetilde{LR}_n\left(\hat{\boldsymbol{\theta}}_1, \hat{\boldsymbol{\theta}}_2\right) := LR_n\left(\hat{\boldsymbol{\theta}}_1, \hat{\boldsymbol{\theta}}_2\right) - K_n\left(f_1, f_2\right),$$

where $K_n\left(f_1, f_2\right)$ is the correction factor which depends on the characteristics of the competing models 1 and 2 and is assumed to satisfy

$$n^{-1/2} K_n\left(f_1, f_2\right) = o_p(1),$$

where $X_n = o_p(1)$ for a set of random variables $(X_n)_{n \in \mathbb{N}}$ means $\lim_{n \to \infty} P(|X_n| > \varepsilon) = 0$ $\forall \varepsilon > 0$. Then the asymptotic result (5.6) also holds for $\widetilde{LR}_n$ and we can redefine test (5.7) in terms of $\widetilde{LR}_n$.

Corresponding to the definitions of AIC (5.2) and BIC (5.4), we consider the following two corrections suggested by Vuong (1989):

- *Akaike correction:* $K_n^A\left(f_1, f_2\right) = k_1 - k_2$,

- *Schwarz correction:* $K_n^S\left(f_1, f_2\right) = \left(\frac{k_1}{2}\right)\log(n) - \left(\frac{k_2}{2}\right)\log(n)$,

where k_1 and k_2 denote the number of parameters of models 1 and 2, respectively. The Schwarz correction again leads to more parsimonious models. Note however that these choices were made out of convenience and other choices might be more appropriate depending on the specific setting, since there is a wide range of possible correction factors!

Moreover, the Vuong test, as well as the AIC and BIC, only allows for a model selection among a set of possible models. If this model set is chosen badly, the selected "best" model can still be far off from the true model. The set of possible models therefore has to be carefully chosen.

5.4 Bivariate copula selection

Now, we turn to the problem of finding an appropriate copula for given bivariate data. When specifying R-vines, these are the (pseudo-)observations as input for tree T_1 and transformed observations (2.19) for all subsequent trees. Here, we will present and evaluate different methods to select an adequate copula, where we exclusively study formal tests, since graphical tests such as K-plots (see, e.g., Genest and Favre (2007)) are impractical in high dimensions as one has to check these plots for $d(d-1)/2$ different variable pairs.

Before considering different criteria, we study the problem of bivariate copula selection in more detail: for certain parameter choices copula families are "close" to each other.

We will quantify the closeness by the KLIC as described in Section 5.1. In particular, we investigate the limiting cases of the BB1 and BB7 copulas (see Section 2.1.3) in a simulation study.

5.4.1 Kullback-Leibler information criteria of bivariate copula families

As noted above, the main problem of pair copula selection is that some families are "close" to other families for certain parameter choices (cp. the limiting cases in Section 2.1.3). This "closeness" can be measured by the KLIC (5.1) between the families, since the copula densities are known, i.e., the KLIC between the true copula C_0 and the alternative copula C_1 is given by

$$KLIC(C_0, C_1) := \int_{[0,1]^2} c_0(u_1, u_2) \log \left[\frac{c_0(u_1, u_2)}{c_1(u_1, u_2)} \right] du_1 du_2 \qquad (5.8)$$

where c_0 and c_1 denote the copula densities corresponding to the copulas C_0 and C_1, respectively, where both copulas might depend on parameters.

Based on three choices of Kendall's τ to compute the copula parameters (cp. Table 2.1), we computed the KLIC's (5.8) for the parametric copula families presented in Section 2.1.3 by numerical integration (see Tables 5.1, 5.2 and 5.3). For the t copula we consider two different numbers of degrees of freedom, and for the BB1 and BB7 copulas we choose the parameters according to three different values of upper tail dependence depending on Kendall's τ. Furthermore, KLIC's with respect to the Clayton and Gumbel survival copulas are computed, where the latter is given in (3.9) and the first is defined accordingly as

$$C_\theta^{SC}(u_1, u_2) = u_1 + u_2 - 1 + C_\theta^C(1 - u_1, 1 - u_2),$$

for a Clayton copula C_θ^C with parameter θ.

The KLIC's of pair copula families shown in Tables 5.1, 5.2 and 5.3 underline the theoretical background (cp. Section 2.1.3 and the scatter and contour plots in Appendix A) as well as empirical observations in Section 3.1. The main findings are:

(i) KLIC's between copula families are increasing with increasing dependence, i.e., differences become more distinct.

(ii) As expected the Gaussian copula is close to the t copula, especially for higher degrees of freedom.

(iii) The Clayton copula which exhibits lower tail dependence can clearly be distinguished from the Gumbel and Joe copulas with upper tail dependence. The latter copulas themselves are similar to each other.

(iv) The KLIC's of the Clayton and Gumbel copulas and their respective survival versions are obviously the same with respect to symmetric copulas because survival copulas are the mirror images of the original copulas. The Clayton survival copula with upper tail dependence is quite close to the Gumbel and Joe copulas, while the Gumbel survival copula with lower tail dependence is similar to the Clayton copula.

TRUE		ALTERNATIVE													
	N	t (df=4)	t (df=8)	C	C (surv.)	G	G (surv.)	F	J	BB1 $\lambda^U=0.1$	BB1 $\lambda^U=0.2$	BB1 $\lambda^U=0.3$	BB7 $\lambda^U=0.1$	BB7 $\lambda^U=0.2$	BB7 $\lambda^U=0.3$
N	-	0.023	0.006	0.031	0.031	0.014	0.014	0.007	0.046	0.013	0.006	0.011	0.017	0.010	0.010
t (df=4)	0.029	-	0.006	0.042	0.042	0.020	0.020	0.032	0.046	0.020	0.011	0.017	0.023	0.014	0.012
t (df=8)	0.008	0.005	-	0.030	0.030	0.011	0.011	0.014	0.040	0.010	0.002	0.008	0.014	0.005	0.004
C	0.029	0.034	0.026	-	0.106	0.065	0.005	0.038	0.124	0.005	0.021	0.055	0.003	0.014	0.035
C (surv.)	0.029	0.034	0.026	0.106	-	0.005	0.065	0.038	0.003	0.061	0.029	0.007	0.070	0.043	0.020
G	0.015	0.018	0.011	0.072	0.005	-	0.040	0.024	0.009	0.035	0.012	0.000	0.042	0.022	0.006
G (surv.)	0.015	0.018	0.011	0.005	0.072	0.040	-	0.024	0.087	0.002	0.009	0.032	0.002	0.005	0.018
F	0.007	0.029	0.012	0.038	0.038	0.020	0.020	-	0.052	0.023	0.015	0.018	0.028	0.021	0.021
J	0.043	0.040	0.036	0.126	0.003	0.009	0.080	0.053	-	0.075	0.039	0.012	0.084	0.053	0.026
BB1 $\lambda^U=0.1$	0.013	0.017	0.009	0.006	0.067	0.036	0.002	0.025	0.081	-	0.006	0.028	0.000	0.002	0.014
BB1 $\lambda^U=0.2$	0.006	0.009	0.002	0.025	0.033	0.013	0.010	0.018	0.043	0.006	-	0.009	0.009	0.002	0.002
BB1 $\lambda^U=0.3$	0.012	0.014	0.008	0.062	0.008	0.000	0.033	0.021	0.013	0.029	0.008	-	0.035	0.016	0.004
BB7 $\lambda^U=0.1$	0.017	0.020	0.013	0.004	0.076	0.042	0.002	0.029	0.090	0.000	0.009	0.034	-	0.004	0.018
BB7 $\lambda^U=0.2$	0.010	0.010	0.005	0.017	0.048	0.022	0.006	0.025	0.059	0.002	0.001	0.017	0.004	-	0.005
BB7 $\lambda^U=0.3$	0.010	0.008	0.004	0.041	0.022	0.007	0.020	0.026	0.029	0.014	0.002	0.004	0.018	0.005	-

Table 5.1: Theoretical KLIC's (5.8) for the copula families presented in Section 2.1.3 with parameter values chosen according to Kendall's $\tau = 0.25$ (cp. Table 2.1). Degrees of freedom of the t copula are chosen as four and eight. Upper tail dependence of the BB1 and BB7 copulas as 0.1, 0.2 and 0.3 (corresponding to lower tail dependence of 0.25, 0.11 and 0.00 for the BB1 and 0.30, 0.22 and 0.11 for the BB7 copula, respectively).

TRUE \ ALTERNATIVE	N	t (df=4)	t (df=8)	C	C (surv.)	G	G (surv.)	F	J	BB1 $\lambda^U=0.25$	BB1 $\lambda^U=0.4$	BB1 $\lambda^U=0.55$	BB7 $\lambda^U=0.25$	BB7 $\lambda^U=0.4$	BB7 $\lambda^U=0.55$
N	-	0.025	0.007	0.142	0.142	0.038	0.038	0.031	0.158	0.059	0.021	0.024	0.091	0.059	0.033
t (df=4)	0.037	-	0.006	0.126	0.126	0.032	0.032	0.056	0.135	0.046	0.013	0.018	0.072	0.039	0.014
t (df=8)	0.010	0.006	-	0.126	0.126	0.028	0.028	0.037	0.139	0.044	0.009	0.013	0.073	0.041	0.016
C	0.118	0.103	0.102	-	0.452	0.226	0.027	0.122	0.468	0.020	0.066	0.179	0.011	0.036	0.112
C (surv.)	0.118	0.103	0.102	0.452	-	0.027	0.226	0.122	0.001	0.275	0.159	0.048	0.344	0.258	0.133
G	0.038	0.030	0.027	0.269	0.032	-	0.106	0.061	0.039	0.136	0.059	0.003	0.185	0.124	0.047
G (surv.)	0.038	0.030	0.027	0.032	0.269	0.106	-	0.061	0.284	0.004	0.012	0.074	0.012	0.010	0.037
F	0.037	0.058	0.038	0.198	0.198	0.074	0.074	-	0.209	0.116	0.072	0.063	0.159	0.131	0.104
J	0.135	0.113	0.115	0.476	0.001	0.035	0.243	0.136	-	0.293	0.174	0.057	0.364	0.275	0.146
BB1 $\lambda^U=0.25$	0.051	0.037	0.037	0.023	0.284	0.118	0.004	0.083	0.300	-	0.014	0.085	0.003	0.003	0.037
BB1 $\lambda^U=0.4$	0.020	0.009	0.008	0.075	0.171	0.053	0.012	0.063	0.185	0.015	-	0.031	0.032	0.011	0.007
BB1 $\lambda^U=0.55$	0.023	0.016	0.013	0.211	0.055	0.003	0.074	0.056	0.065	0.096	0.034	-	0.137	0.085	0.025
BB7 $\lambda^U=0.25$	0.074	0.057	0.058	0.013	0.333	0.152	0.011	0.105	0.349	0.003	0.029	0.114	-	0.008	0.056
BB7 $\lambda^U=0.4$	0.048	0.029	0.032	0.040	0.241	0.098	0.009	0.096	0.256	0.003	0.010	0.068	0.008	-	0.021
BB7 $\lambda^U=0.55$	0.029	0.010	0.013	0.111	0.125	0.037	0.032	0.087	0.137	0.033	0.007	0.021	0.053	0.021	-

Table 5.2: Theoretical KLIC's (5.8) for the copula families presented in Section 2.1.3 with parameter values chosen according to Kendall's $\tau = 0.5$ (cp. Table 2.1). Degrees of freedom of the t copula are chosen as four and eight. Upper tail dependence of the BB1 and BB7 copulas as 0.25, 0.4 and 0.55 (corresponding to lower tail dependence of 0.63, 0.52 and 0.08 for the BB1 and 0.68, 0.65 and 0.55 for the BB7 copula, respectively).

TRUE	N	t (df=4)	t (df=8)	C	C (surv.)	G	G (surv.)	F	J	BB1 λ^U=0.4	BB1 λ^U=0.6	BB1 λ^U=0.8	BB7 λ^U=0.4	BB7 λ^U=0.6	BB7 λ^U=0.8
N	-	0.026	0.008	0.353	0.353	0.069	0.069	0.088	0.347	0.204	0.090	0.053	0.349	0.316	0.135
t (df=4)	0.048	-	0.007	0.300	0.300	0.054	0.054	0.116	0.295	0.162	0.060	0.039	0.285	0.243	0.073
t (df=8)	0.012	0.006	-	0.316	0.316	0.053	0.053	0.096	0.311	0.173	0.065	0.037	0.306	0.268	0.093
C	0.372	0.255	0.276	-	1.320	0.585	0.094	0.252	1.280	0.028	0.113	0.530	0.013	0.057	0.355
C (surv.)	0.372	0.255	0.276	1.320	-	0.094	0.585	0.252	0.000	0.997	0.673	0.121	1.351	1.292	0.655
G	0.075	0.052	0.052	0.650	0.101	-	0.209	0.127	0.101	0.435	0.241	0.002	0.650	0.596	0.228
G (surv.)	0.075	0.052	0.052	0.101	0.650	0.209	-	0.127	0.634	0.031	0.008	0.177	0.092	0.079	0.102
F	0.178	0.152	0.138	0.644	0.644	0.234	0.234	-	0.619	0.482	0.333	0.219	0.710	0.732	0.546
J	0.385	0.259	0.282	1.342	0.000	0.097	0.596	0.256	-	1.015	0.686	0.125	1.374	1.315	0.668
BB1 λ^U=0.4	0.183	0.128	0.138	0.028	0.891	0.349	0.027	0.202	0.868	-	0.028	0.308	0.014	0.016	0.161
BB1 λ^U=0.6	0.078	0.046	0.051	0.103	0.575	0.183	0.007	0.164	0.563	0.026	-	0.154	0.078	0.053	0.057
BB1 λ^U=0.8	0.056	0.036	0.035	0.571	0.127	0.002	0.171	0.124	0.127	0.371	0.195	-	0.566	0.513	0.181
BB7 λ^U=0.4	0.286	0.206	0.223	0.015	1.089	0.471	0.074	0.271	1.061	0.014	0.073	0.424	-	0.015	0.241
BB7 λ^U=0.6	0.213	0.152	0.167	0.048	0.844	0.352	0.060	0.282	0.827	0.016	0.044	0.313	0.013	-	0.134
BB7 λ^U=0.8	0.097	0.048	0.064	0.222	0.348	0.115	0.068	0.253	0.346	0.108	0.044	0.097	0.177	0.118	-

ALTERNATIVE

Table 5.3: Theoretical KLIC's (5.8) for the copula families presented in Section 2.1.3 with parameter values chosen according to Kendall's $\tau = 0.75$ (cp. Table 2.1). Degrees of freedom of the t copula are chosen as four and eight. Upper tail dependence of the BB1 and BB7 copulas as 0.4, 0.6 and 0.8 (corresponding to lower tail dependence of 0.87, 0.84 and 0.17 for the BB1 and 0.90, 0.90 and 0.87 for the BB7 copula, respectively).
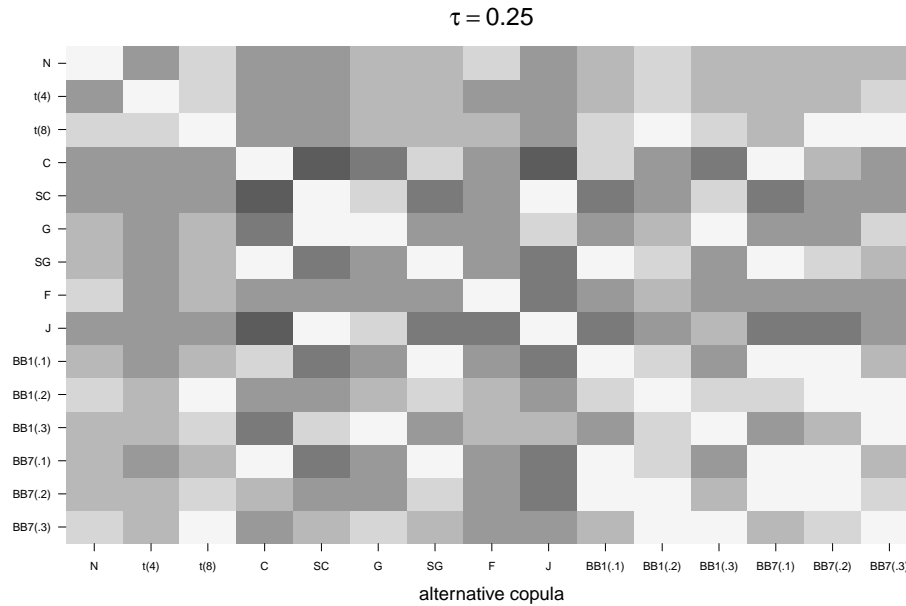
Figure 5.1: Illustration of Table 5.1. Light colors indicate small values, dark colors larger values (same scale as in Figures 5.2 and 5.3). Rows correspond to the respective true model. Degrees of freedom of the t copula family and upper tail dependence parameters of the BB1 and BB7 copula families are denoted in brackets.

(v) For small Kendall's τ, the Frank copula is close to the Gaussian copula, but this similarity reduces in higher dependence, where the Frank copula is even closer to the t copula which corresponds to the empirical results of Hu dependence in Section 3.1.6 and shows that the Frank copula is increasingly non-normal with increasing Kendall's τ.

(vi) The limiting cases of the BB1 and BB7 copulas are well reproduced by the KLIC's: Gumbel and Clayton copulas can hardly be distinguished from BB1 and BB7 copulas in the respective cases. The Joe copula is however less close to the BB7 copula than the Gumbel copula. This is due to the specific parameter choices, since, e.g., Kendall's $\tau = 0.75$ and upper tail dependence $\lambda^{upper} = 0.8$ correspond to parameters $\theta = 3.80$ and $\delta = 4.81$ which are clearly not the limiting cases leading to the Joe copula. These limiting cases are investigated in more detail below.

(vii) Due to their flexibility, the BB1 and BB7 copulas are quite close to many copula families, in particular to the t copula. It is therefore a delicate issue of copula selection criteria to distinguish the true copula from a BB1 or BB7 copula with appropriate parameters.

Figures 5.1 to 5.3 illustrate the values in Tables 5.1 to 5.3. Light colors indicate small values, dark colors larger values.
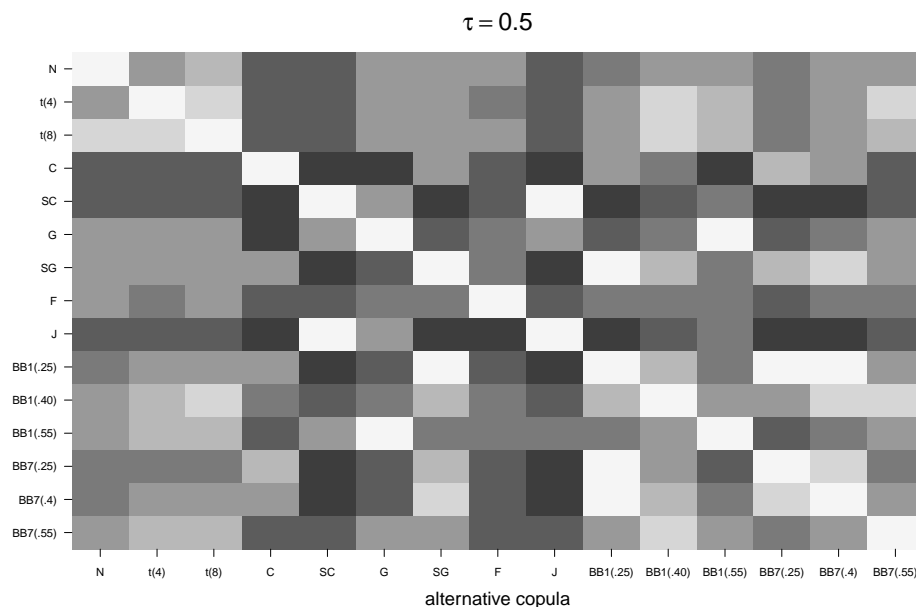
Figure 5.2: Illustration of Table 5.2. Light colors indicate small values, dark colors larger values (same scale as in Figures 5.1 and 5.3). Rows correspond to the respective true model. Degrees of freedom of the t copula family and upper tail dependence parameters of the BB1 and BB7 copula families are denoted in brackets.
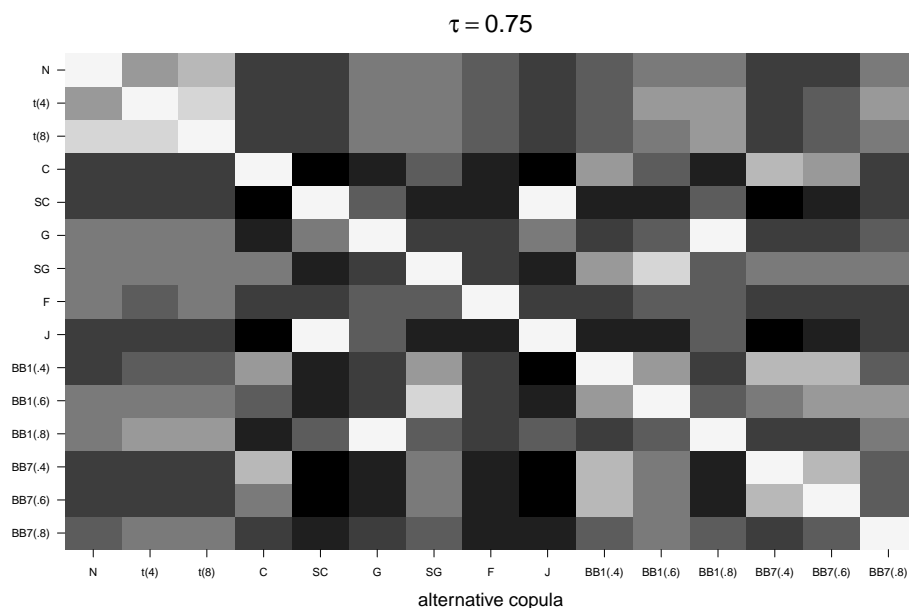

Figure 5.3: Illustration of Table 5.3. Light colors indicate small values, dark colors larger values (same scale as in Figures 5.1 and 5.2). Rows correspond to the respective true model. Degrees of freedom of the t copula family and upper tail dependence parameters of the BB1 and BB7 copula families are denoted in brackets.

5.4.2 Limiting cases of the BB1 and BB7 copulas

As seen by the theoretical KLIC's of bivariate copula families, the BB1 and BB7 copulas are close to some other families. This is due to their flexible two-parameter structure and their limiting cases (cp. Section 2.1.3): the Clayton copula is a subfamily of the BB1 copula with $\delta = 1$ and of the BB7 copula with $\theta = 1$. Furthermore, the limiting case of the BB1 copula when $\theta \to 0$ is the Gumbel copula, while the Joe copula is obtained from the BB7 copula for $\delta \to 0$.

Hence, we are interested in the thresholds of parameter values, where the BB1 and BB7 copulas become statistically indistinguishable from their limiting cases. To investigate this issue, we used Vuong tests without correction (this is not a question of numbers of parameters) as described in Section 5.3. According to the Vuong test with significance level 5%, the null hypothesis that two models are indistinguishable cannot be rejected if the test statistic in absolute terms is smaller than the 97.5%-quantile of the standard normal distribution. We exploit this by repeatedly simulating from the BB1 and BB7 copulas for different parameter choices close to the respective parameter limit and for three choices of Kendall's τ (0.25, 0.5, 0.75) to determine the other parameter and then performing a Vuong test between the true copula and the respective limiting copula. E.g., in Table 5.4 (left panel) we simulated $n = 1000$ observations from a BB1 copula with $\delta = 1.05, ..., 1.3$ and Kendall's $\tau \in \{0.25, 0.5, 0.75\}$. Then a Vuong test is performed for the given simulated data between the BB1 copula and a Clayton copula with parameter chosen according to the given Kendall's τ (cp. Table 2.1)). The percentage of non-rejections (of the null hypothesis that the respective two copulas are indistinguishable) for $R = 1000$ repetitions is displayed in the last column. All results are shown in Tables 5.4 and 5.5.

Note that results for the BB7 copula with Kendall's $\tau = 0.75$ and small parameters δ close to the limit 0 have not been computed due to numerical problems in the simulation for this extreme parameter choice (cp. Table C.29 in Schepsmeier (2010)). The overall results can be summarized as follows:

(i) The closeness to the limiting copulas increases with increasing dependence.

(ii) The limiting one-parameter copula families exhibit either lower or upper tail dependence. The tail dependence parameter of the BB1/BB7 copula corresponding to the respective zero tail dependence parameter of the one-parameter copula family approaches 0 for values close to the limiting case.

(iii) The respective non-zero tail dependence parameters closely agree for parameter close to the respective limit. The difference decreases with increasing Kendall's τ.

In all four studies, the null hypothesis that both copula models are indistinguishable cannot be rejected in more than 50% of the repetitions when the respective parameter is accurate to 0.1 to the limiting case. Even if Tables 5.4 (right table) and 5.5 (right table) indicate that we could even choose 0.15 or 0.2 in these cases, we think that an accuracy of 0.1 to the limiting case is certainly reasonable for all limiting cases due to the unknown simulation error and the limited number of different cases considered. We will exploit this in the simulation study in Section 5.4.4 and in Chapters 10 and 11.

Left table

τ	BB1 copula θ	δ	λ^{lower}	λ^{upper}	Clayton copula θ	λ^{lower}	Vuong test non-rejection
0.25	0.54	1.05	0.29	0.06	0.67	0.35	88.2%
	0.42	1.10	0.23	0.12	0.67	0.35	51.7%
	0.32	1.15	0.15	0.17	0.67	0.35	18.7%
	0.22	1.20	0.07	0.22	0.67	0.35	4.4%
	0.13	1.25	0.02	0.26	0.67	0.35	0.3%
	0.05	1.30	0.00	0.30	0.67	0.35	0.1%
0.5	1.81	1.05	0.69	0.06	2.00	0.71	91.3%
	1.64	1.10	0.68	0.12	2.00	0.71	74.2%
	1.48	1.15	0.67	0.17	2.00	0.71	42.9%
	1.33	1.20	0.65	0.22	2.00	0.71	21.1%
	1.20	1.25	0.63	0.26	2.00	0.71	7.7%
	1.08	1.30	0.61	0.30	2.00	0.71	2.6%
0.75	5.62	1.05	0.89	0.06	6.00	0.89	95.4%
	5.27	1.10	0.89	0.12	6.00	0.89	89.2%
	4.96	1.15	0.89	0.17	6.00	0.89	75.3%
	4.67	1.20	0.88	0.22	6.00	0.89	59.5%
	4.40	1.25	0.88	0.26	6.00	0.89	45.9%
	4.15	1.30	0.88	0.30	6.00	0.89	29.9%

Right table

τ	BB1 copula θ	δ	λ^{lower}	λ^{upper}	Gumbel copula θ	λ^{upper}	Vuong test non-rejection
0.25	0.05	1.30	0.00	0.30	1.33	0.32	93.2%
	0.10	1.27	0.00	0.27	1.33	0.32	85.0%
	0.15	1.24	0.02	0.25	1.33	0.32	70.1%
	0.20	1.21	0.06	0.23	1.33	0.32	53.4%
	0.25	1.19	0.10	0.21	1.33	0.32	35.0%
	0.30	1.16	0.14	0.18	1.33	0.32	20.7%
0.5	0.05	1.95	0.00	0.57	2.00	0.59	92.8%
	0.10	1.90	0.03	0.56	2.00	0.59	84.4%
	0.15	1.86	0.08	0.55	2.00	0.59	71.9%
	0.20	1.82	0.15	0.54	2.00	0.59	60.8%
	0.25	1.78	0.21	0.52	2.00	0.59	46.7%
	0.30	1.74	0.26	0.51	2.00	0.59	32.7%
0.75	0.05	3.90	0.03	0.81	4.00	0.81	92.3%
	0.10	3.81	0.16	0.80	4.00	0.81	85.1%
	0.15	3.72	0.29	0.80	4.00	0.81	73.4%
	0.20	3.64	0.39	0.79	4.00	0.81	63.0%
	0.25	3.56	0.46	0.78	4.00	0.81	48.2%
	0.30	3.48	0.51	0.78	4.00	0.81	34.3%

Table 5.4: Results of $R = 1000$ simulations from a BB1 copula of $n = 1000$ observations each. In the left table, parameter δ is chosen close to the limiting case 1 and parameter θ is determined according to δ and Kendall's $\tau \in \{0.25, 0.5, 0.75\}$ (cp. Table 2.1), while in the right table, parameter θ is chosen close to the limiting case 0 and parameter δ is determined according to θ and Kendall's $\tau \in \{0.25, 0.5, 0.75\}$. The respective last columns report the percentage of non-rejections of Vuong tests between the BB1 copula with given parameters and a Clayton or Gumbel copula, respectively, with parameter chosen according to Kendall's τ.

Left table:

τ	BB7 copula				Clayton copula		Vuong test
	θ	δ	λ^{lower}	λ^{upper}	θ	λ^{lower}	non-rejection
0.25	1.05	0.61	0.32	0.06	0.67	0.35	91.8%
	1.10	0.56	0.29	0.12	0.67	0.35	72.6%
	1.15	0.50	0.25	0.17	0.67	0.35	40.4%
	1.20	0.45	0.21	0.22	0.67	0.35	19.4%
	1.25	0.39	0.17	0.26	0.67	0.35	6.9%
	1.30	0.33	0.12	0.30	0.67	0.35	1.0%
0.5	1.05	1.96	0.70	0.06	2.00	0.71	96.2%
	1.10	1.93	0.70	0.12	2.00	0.71	88.3%
	1.15	1.89	0.69	0.17	2.00	0.71	72.5%
	1.20	1.85	0.69	0.22	2.00	0.71	53.7%
	1.25	1.80	0.68	0.26	2.00	0.71	37.2%
	1.30	1.76	0.67	0.30	2.00	0.71	20.6%
0.75	1.05	6.07	0.89	0.06	6.00	0.89	97.0%
	1.10	6.14	0.89	0.12	6.00	0.89	96.4%
	1.15	6.21	0.89	0.17	6.00	0.89	91.6%
	1.20	6.27	0.90	0.22	6.00	0.89	83.2%
	1.25	6.33	0.90	0.26	6.00	0.89	75.6%
	1.30	6.38	0.90	0.30	6.00	0.89	64.6%

Right table:

τ	BB7 copula				Joe copula		Vuong test
	θ	δ	λ^{lower}	λ^{upper}	θ	λ^{upper}	non-rejection
0.25	1.55	0.05	0.00	0.44	1.59	0.45	93.7%
	1.51	0.10	0.00	0.42	1.59	0.45	79.1%
	1.46	0.15	0.01	0.39	1.59	0.45	63.4%
	1.42	0.20	0.03	0.37	1.59	0.45	38.6%
	1.37	0.25	0.06	0.34	1.59	0.45	19.8%
	1.33	0.30	0.10	0.32	1.59	0.45	8.8%
0.5	2.81	0.05	0.00	0.72	2.85	0.72	94.1%
	2.76	0.10	0.00	0.72	2.85	0.72	88.9%
	2.72	0.15	0.01	0.71	2.85	0.72	78.0%
	2.68	0.20	0.03	0.70	2.85	0.72	68.7%
	2.63	0.25	0.06	0.70	2.85	0.72	46.8%
	2.59	0.30	0.10	0.69	2.85	0.72	35.8%
0.75	6.74	0.05	0.00	0.89	6.70	0.89	-
	6.69	0.10	0.00	0.89	6.70	0.89	-
	6.65	0.15	0.01	0.89	6.70	0.89	-
	6.61	0.20	0.03	0.89	6.70	0.89	-
	6.56	0.25	0.06	0.89	6.70	0.89	-
	6.52	0.30	0.10	0.89	6.70	0.89	-

Table 5.5: Results of $R = 1000$ simulations from a BB7 copula of $n = 1000$ observations each. In the left table, parameter θ is chosen close to the limiting case 1 and parameter δ is determined according to θ and Kendall's $\tau \in \{0.25, 0.5, 0.75\}$ (cp. Table 2.1), while in the right table, parameter δ is chosen close to the limiting case 1 and parameter θ is determined according to δ and Kendall's $\tau \in \{0.25, 0.5\}$. Results for $\tau = 0.75$ have not been computed in this case due to numerical inaccuracies in the simulation of the BB7 copula for this extreme parameter choice (cp. Table C.29 in Schepsmeier (2010)). The respective last columns report the percentage of non-rejections of Vuong tests between the BB7 copula with given parameters and a Clayton or Joe copula, respectively, with parameter chosen according to Kendall's τ.

5.4.3   Selection criteria

After having analyzed the problem of bivariate copula selection from a theoretical point of view, we now consider different selection criteria and critically evaluate them in a simulation study.

(i) *Goodness-of-fit testing:*

From a theoretical point of view, the method of choice for copula selection is goodness-of-fit testing because it not only allows to choose among different families of copulas, but also attaches a significance level to this choice. As described in Section 4.2, the most natural choice of a copula goodness-of-fit test is the blanket test defined in (4.6). This test can be performed for several copula families and then the copula with the highest p-value is chosen, but only if this p-value is larger than the pre-specified significance level α, because it means that the null hypothesis that the unknown copula belongs to the respective family cannot be rejected at level α. If however the maximum p-value is smaller than α, no copula can be selected based on goodness-of-fit tests and we have to refer to another method (see below for three possible choices).

This selection criterion can also be interpreted as a two-step approach: first, determine which pair copulas are eligible at all and, second, among these copulas select the copula with the smallest test statistic of the chosen goodness-of-fit test. E.g., the Cramér-von Mises test statistic of the blanket test (4.6) can be regarded as a distance measure between the respective hypothesized copula family and the unknown copula represented by the empirical copula (also cp. Section 4.1.1).

Note that other goodness-of-fit tests than the blanket test based on the empirical copula could be used as well. We however believe that our choice is adequate, since it is the most objective and most natural approach and has shown good performance results for bivariate data in the studies by Genest et al. (2009) and Berg (2009).

(ii) *Copula selection based on data characteristics:*

If one is not willing to conduct goodness-of-fit tests because they are computationally too demanding (p-value computation of the test based on the empirical copula process is demanding even if the multiplier approach (Section 4.2.2) is used) or the goodness-of-fit tests rejected each copula family, we can take the following simple approach to copula selection which is based on empirical characteristics of the observations.

For this, calculate the empirical Kendall's τ as well as the empirical upper and lower exeedance Kendall's τ's as described in Section 3.1.3. Next fit bivariate copulas for all families that are considered. Then calculate theoretical Kendall's τ and exceedance dependence based on the estimated coefficients and choose the family which minimizes

$$\frac{1}{2}\left[\hat{\tau}_n^{lower} - \tau_{\hat{\boldsymbol{\theta}}}^{lower}\right]^2 + \left[\hat{\tau}_n - \tau_{\hat{\boldsymbol{\theta}}}\right]^2 + \frac{1}{2}\left[\hat{\tau}_n^{upper} - \tau_{\hat{\boldsymbol{\theta}}}^{upper}\right]^2,$$

where $\hat{\boldsymbol{\theta}}$ denotes the estimated parameter(s) of the respective copula family, and $\tau_{\hat{\boldsymbol{\theta}}}^{lower}$, $\tau_{\hat{\boldsymbol{\theta}}}^{upper}$ and $\tau_{\hat{\boldsymbol{\theta}}}$ are the theoretical exceedance dependence and Kendall's τ with respect to $\hat{\boldsymbol{\theta}}$ and the corresponding copula family, where we usually choose the thresholds $\delta_1 = \delta_2 = 0.2$ for lower and $\delta_1 = \delta_2 = 0.8$ for upper dependence. The weights of $1/2$ are chosen in order to obtain a balanced criterion which does not overemphasizes tail behavior. Different weightings are of course possible as well.

The idea is that we not only capture the dependence in the center by Kendall's τ, but also the exceedance dependence, i.e., the joint tail behavior, which is characteristic for copula families as investigated in Section 3.1.3. Different weightings could pronounce an adequate modeling of, e.g., lower tail behavior more strongly. A copula selection based on only one of the measures (e.g., Kendall's τ) is of course also feasible but less accurate and should only be used in exceptional cases, since it contains only few information about the data. Note that, in contrast to goodness-of-fit tests, this approach cannot report some kind of confidence level with which a family is chosen, i.e., we do not know whether the chosen family is a good choice in absolute terms but only in relative terms compared to the other families considered.

(iii) *Information criteria:*

An alternative copula selection approach is based on the AIC as defined in (5.2) and given for a specific copula with density c by

$$AIC = -2 \sum_{i=1}^{n} \log c(u_{i1}, u_{i2}|\hat{\boldsymbol{\theta}}) + 2k,$$

where $\hat{\boldsymbol{\theta}} = (\hat{\theta}_1, ..., \hat{\theta}_k)'$ and either $k = 1$ or $k = 2$ for the bivariate copula families discussed in Section 2.1.3, e.g., $k = 1$ and $\hat{\theta} = \hat{\rho}$ for the bivariate Gaussian copula.

The use of the AIC for model selection is proposed, e.g., in Section 10.3 in Joe (1997). Manner (2007) investigates its performance in bivariate copula selection and shows that it performs quite well in identifying the correct copula family, especially when sample sizes and dependence increase. For weak dependence the performance of the AIC as selection criterion is unsatisfactory. Therefore the AIC is an ad-hoc approach to copula selection and we also have to keep in mind the increased uncertainty in AIC comparisons due to the fact that copula families are not nested in general (cp. Section 5.2). Given this increased uncertainty, it may not be an alternative to copula goodness-of-fit testing when enough time and resources are available, but it is certainly an alternative to copula selection based on exceedance dependence and Kendall's τ and we will examine its performance in detail in the simulation study below (Section 5.4.4). Of course, the BIC can be used as well. Its use obviously penalizes two-parameter copula families compared to one-parameter families, since $\log(n) > 2$ for $n \geq 8$.

Grønneberg and Hjort (2008) also investigate the use of the AIC in copula model selection and find that when using maximum pseudo likelihood estimation (cp. Section 4.1.2) the derivation of the AIC formula is incorrect. Therefore they extend the formula and develop the so-called *copula information criterion (CIC)*. As we use

MPL estimation as well, it would be sensible to use the CIC instead of the AIC, but there are some issues that prevent the practical use of the CIC: Grønneberg and Hjort (2008) state that the CIC often does not exist, e.g., for the Gumbel and Joe families and moreover it lacks the advantage of AIC and BIC that it is easy to compute. As a result, we will not consider the CIC in copula selection.

(iv) *Scoring method using Vuong tests:*

Since the AIC may possibly be only an ad-hoc approach to accurate copula selection as discussed above, we consider an alternative copula selection method based on Vuong tests which we discussed in Section 5.3 (cp. Belgorodski (2010))

In contrast to copula goodness-of-fit tests, Vuong tests cannot test the hypothesis (4.1) directly. Therefore we lower our sights and consider the hypotheses

$$H_0 : C \in \mathcal{C} \qquad \text{against} \qquad H_1 : C \in \widetilde{\mathcal{C}}, \tag{5.9}$$

where $\widetilde{\mathcal{C}}$ is a parametric copula family other than $\mathcal{C}$. As it would be quite inappropriate to perform this test only for one single alternative hypothesis, we consider different families $\widetilde{\mathcal{C}}$ which are adequate for the given data characteristics (e.g., if observations are negatively dependent, non-rotated Clayton and Gumbel as well as Joe, BB1 and BB7 families are no possibilities). These tests can be performed using Vuong tests based on the fitted bivariate copulas that are considered. Hence, we obtain a series of tests which tell us whether a specific model, say $\mathcal{A}$, is better, equally good or worse than other models.

The final copula selection is then made by the following scoring method: each time model $\mathcal{A}$ is preferred to another model the score of model $\mathcal{A}$ is increased by 1. If model $\mathcal{A}$ cannot be significantly distinguished from another model, the score is left unchanged. If however, another model is preferred to model $\mathcal{A}$, we subtract 1 from the score of model $\mathcal{A}$. This procedure is repeated for each model under consideration and we choose the model with the highest score. Unfortunately, two models can have the same score. Then no copula selection based on the scoring approach is possible and we have to use another method.

5.4.4 Simulation study

Having different copula selection methods at hand, we evaluate the performance of the methods in a Monte Carlo study. In each of the $R = 1000$ repetitions, we simulated $n \in \{500, 1000\}$ observations from six different pair copula families (Gaussian (N), t with four degrees of freedom, Clayton (C), Gumbel (G), Frank (F), Joe (J)) for three choices of Kendall's $\tau \in \{0.25, 0.5, 0.75\}$ to determine the copula parameters (cp. Table 2.1) and then performed the three selection methods discussed above as well as copula selection using goodness-of-fit testing. The latter is mainly considered for comparison only and the set of copulas to choose from is restricted to the five copula families N, t, G, C and F as denoted above, while the other three methods also consider the Joe, BB1 and BB7 copulas. The performance of the blanket goodness-of-fit test based on the empirical copula

True	τ	Kendall's τ and exceedance dependence (see 5.4.3 (ii))								AIC (see 5.4.3 (iii))							
		N	t	C	G	F	J	BB1	BB7	N	t	C	G	F	J	BB1	BB7
N	0.25	**0.216**	0.029	0.123	0.122	0.332	0.018	0.047	0.082	**0.787**	0.031	0.007	0.022	0.086	0.000	0.047	0.020
	0.5	**0.570**	0.036	0.014	0.084	0.134	0.000	0.143	0.019	**0.907**	0.072	0.000	0.000	0.004	0.000	0.017	0.000
	0.75	**0.729**	0.030	0.000	0.037	0.004	0.000	0.195	0.005	**0.922**	0.075	0.000	0.000	0.000	0.000	0.003	0.000
t (df=4)	0.25	0.070	**0.424**	0.091	0.094	0.019	0.025	0.103	0.174	0.000	**0.902**	0.002	0.010	0.003	0.000	0.018	0.065
	0.5	0.104	**0.314**	0.009	0.064	0.002	0.000	0.333	0.174	0.000	**0.840**	0.000	0.000	0.000	0.000	0.128	0.032
	0.75	0.130	**0.299**	0.000	0.023	0.000	0.000	0.475	0.073	0.001	**0.815**	0.000	0.000	0.000	0.000	0.184	0.000
C	0.25	0.049	0.136	**0.713**	0.018	0.072	0.000	0.002	0.010	0.001	0.000	**0.970**	0.000	0.000	0.000	0.007	0.022
	0.5	0.006	0.027	**0.925**	0.000	0.001	0.000	0.007	0.034	0.000	0.000	**0.948**	0.000	0.000	0.000	0.020	0.032
	0.75	0.000	0.000	**0.884**	0.000	0.000	0.000	0.048	0.068	0.000	0.000	**0.898**	0.000	0.000	0.000	0.053	0.049
G	0.25	0.054	0.150	0.004	**0.225**	0.062	0.270	0.014	0.221	0.011	0.020	0.000	**0.753**	0.005	0.087	0.030	0.094
	0.5	0.028	0.082	0.000	**0.461**	0.007	0.163	0.024	0.235	0.000	0.003	0.000	**0.841**	0.000	0.000	0.065	0.091
	0.75	0.012	0.022	0.000	**0.699**	0.000	0.012	0.046	0.209	0.000	0.004	0.000	**0.912**	0.000	0.000	0.071	0.013
F	0.25	0.135	0.046	0.097	0.081	**0.576**	0.007	0.033	0.025	0.080	0.013	0.002	0.010	**0.892**	0.000	0.003	0.000
	0.5	0.129	0.019	0.003	0.019	**0.826**	0.000	0.004	0.000	0.004	0.002	0.000	0.000	**0.993**	0.000	0.001	0.000
	0.75	0.037	0.000	0.000	0.003	**0.960**	0.000	0.000	0.000	0.000	0.000	0.000	0.000	**1.000**	0.000	0.000	0.000
J	0.25	0.008	0.107	0.000	0.207	0.024	**0.642**	0.000	0.012	0.000	0.000	0.000	0.077	0.000	**0.904**	0.000	0.019
	0.5	0.000	0.004	0.000	0.157	0.000	**0.780**	0.000	0.059	0.000	0.000	0.000	0.001	0.000	**0.928**	0.000	0.071
	0.75	0.000	0.000	0.000	0.007	0.000	**0.809**	0.000	0.184	0.000	0.000	0.000	0.000	0.000	**0.988**	0.000	0.012

True	τ	Scoring (Vuong test with Akaike correction) (see 5.4.3 (iv))								Goodness-of-fit (see 5.4.3 (i))							
		N	t	C	G	F	J	BB1	BB7	N	t	C	G	F	J	BB1	BB7
N	0.25	**0.565**	0.020	0.000	0.037	0.008	0.000	0.036	0.000	**0.611**	0.130	0.009	0.089	0.145	0.000	0.036	0.000
	0.5	**0.636**	0.019	0.000	0.000	0.001	0.000	0.093	0.000	**0.758**	0.147	0.000	0.024	0.053	0.000	0.093	0.000
	0.75	**0.672**	0.023	0.000	0.000	0.000	0.000	0.018	0.000	**0.806**	0.156	0.000	0.016	0.003	0.000	0.018	0.000
t (df=4)	0.25	0.001	**0.524**	0.000	0.040	0.001	0.000	0.047	0.033	0.227	**0.464**	0.010	0.221	0.046	0.000	0.047	0.000
	0.5	0.000	**0.218**	0.000	0.000	0.000	0.000	0.195	0.003	0.410	**0.447**	0.000	0.103	0.008	0.000	0.195	0.000
	0.75	0.000	**0.117**	0.000	0.000	0.000	0.000	0.186	0.000	**0.586**	0.301	0.000	0.063	0.000	0.000	0.186	0.000
C	0.25	0.008	0.000	**0.836**	0.000	0.001	0.000	0.001	0.008	0.013	0.036	**0.911**	0.000	0.006	0.000	0.001	0.000
	0.5	0.000	0.000	**0.935**	0.000	0.000	0.000	0.000	0.000	0.000	0.001	**0.947**	0.000	0.000	0.000	0.000	0.000
	0.75	0.000	0.000	**0.825**	0.000	0.000	0.000	0.000	0.000	0.000	0.000	**0.961**	0.000	0.000	0.000	0.000	0.000
G	0.25	0.008	0.001	0.000	**0.607**	0.001	0.000	0.014	0.010	0.051	0.046	0.000	**0.857**	0.024	0.000	0.014	0.010
	0.5	0.000	0.000	0.000	**0.530**	0.000	0.000	0.049	0.023	0.025	0.015	0.000	**0.929**	0.005	0.000	0.049	0.023
	0.75	0.000	0.000	0.000	**0.686**	0.000	0.000	0.035	0.000	0.010	0.001	0.000	**0.960**	0.000	0.000	0.035	0.000
F	0.25	0.102	0.036	0.000	0.039	**0.422**	0.000	0.020	0.000	0.067	0.034	0.001	0.012	**0.846**	0.000	0.020	0.000
	0.5	0.011	0.021	0.000	0.000	**0.822**	0.000	0.000	0.000	0.000	0.009	0.000	0.000	**0.946**	0.000	0.000	0.000
	0.75	0.000	0.000	0.000	0.001	**0.999**	0.000	0.000	0.000	0.000	0.000	0.000	0.000	**0.968**	0.000	0.000	0.000
J	0.25	0.000	0.000	0.000	0.102	0.000	**0.312**	0.000	0.000	0.000	0.000	0.000	0.520	0.000	0.000	0.000	0.000
	0.5	0.000	0.000	0.000	0.005	0.000	**0.815**	0.000	0.032	0.000	0.000	0.000	0.035	0.000	0.000	0.000	0.000
	0.75	0.000	0.001	0.000	0.001	0.000	**0.652**	0.000	0.000	0.000	0.001	0.000	0.002	0.000	0.000	0.000	0.000

Table 5.6: Percentages of selected copula families of $R = 1000$ repetitions with $n = 500$ simulated observations each. Parameters of the true copulas (as indicated in the left column) are chosen according to three choices of Kendall's τ (cp. Table 2.1).

Kendall's τ and exceedance dependence (see 5.4.3 (ii))

True	τ	N	t	C	G	F	J	BB1	BB7
N	0.25	**0.329**	0.023	0.090	0.080	0.316	0.000	0.124	0.038
	0.5	**0.760**	0.033	0.003	0.035	0.057	0.000	0.111	0.001
	0.75	**0.826**	0.039	0.000	0.009	0.000	0.000	0.126	0.000
t (df=4)	0.25	0.029	**0.476**	0.064	0.060	0.002	0.001	0.165	0.203
	0.5	0.055	**0.321**	0.001	0.024	0.000	0.000	0.446	0.153
	0.75	0.083	**0.290**	0.000	0.001	0.000	0.000	0.609	0.017
C	0.25	0.035	0.100	**0.844**	0.003	0.016	0.000	0.000	0.002
	0.5	0.000	0.002	**0.980**	0.000	0.000	0.000	0.003	0.015
	0.75	0.000	0.000	**0.935**	0.000	0.000	0.000	0.024	0.041
G	0.25	0.044	0.132	0.000	**0.352**	0.017	0.219	0.002	0.234
	0.5	0.008	0.047	0.000	**0.638**	0.000	0.085	0.012	0.210
	0.75	0.003	0.008	0.000	**0.831**	0.000	0.002	0.019	0.137
F	0.25	0.183	0.051	0.043	0.051	**0.640**	0.000	0.026	0.006
	0.5	0.086	0.004	0.000	0.001	**0.909**	0.000	0.000	0.000
	0.75	0.002	0.000	0.000	0.000	**0.998**	0.000	0.000	0.000
J	0.25	0.005	0.047	0.000	0.223	0.000	**0.718**	0.000	0.007
	0.5	0.000	0.001	0.000	0.089	0.000	**0.876**	0.000	0.034
	0.75	0.000	0.000	0.000	0.001	0.000	**0.836**	0.000	0.163

AIC (see 5.4.3 (iii))

True	τ	N	t	C	G	F	J	BB1	BB7
N	0.25	**0.892**	0.048	0.000	0.005	0.026	0.000	0.023	0.006
	0.5	**0.928**	0.072	0.000	0.000	0.000	0.000	0.000	0.000
	0.75	**0.924**	0.076	0.000	0.000	0.000	0.000	0.000	0.000
t (df=4)	0.25	0.000	**0.986**	0.000	0.000	0.000	0.000	0.002	0.012
	0.5	0.000	**0.948**	0.000	0.000	0.000	0.000	0.048	0.004
	0.75	0.000	**0.921**	0.000	0.000	0.000	0.000	0.079	0.000
C	0.25	0.000	0.000	**0.996**	0.000	0.000	0.000	0.000	0.004
	0.5	0.000	0.000	**0.978**	0.000	0.000	0.000	0.003	0.019
	0.75	0.000	0.000	**0.927**	0.000	0.000	0.000	0.027	0.046
G	0.25	0.002	0.005	0.000	**0.824**	0.001	0.037	0.013	0.118
	0.5	0.000	0.000	0.000	**0.930**	0.000	0.000	0.031	0.039
	0.75	0.000	0.000	0.000	**0.961**	0.000	0.000	0.038	0.001
F	0.25	0.038	0.006	0.000	0.000	**0.955**	0.000	0.001	0.000
	0.5	0.000	0.000	0.000	0.000	**1.000**	0.000	0.000	0.000
	0.75	0.000	0.000	0.000	0.000	**1.000**	0.000	0.000	0.000
J	0.25	0.000	0.000	0.000	0.015	0.000	**0.973**	0.000	0.012
	0.5	0.000	0.000	0.000	0.000	0.000	**0.955**	0.000	0.045
	0.75	0.000	0.000	0.000	0.000	0.000	**0.998**	0.000	0.002

Scoring (Vuong test with Akaike correction) (see 5.4.3 (iv))

True	τ	N	t	C	G	F	J	BB1	BB7
N	0.25	**0.625**	0.016	0.000	0.005	0.000	0.000	0.096	0.000
	0.5	**0.767**	0.029	0.000	0.000	0.000	0.000	0.038	0.000
	0.75	**0.874**	0.034	0.000	0.000	0.000	0.000	0.000	0.000
t (df=4)	0.25	0.000	**0.628**	0.000	0.003	0.000	0.000	0.010	0.024
	0.5	0.000	**0.239**	0.000	0.000	0.000	0.000	0.108	0.001
	0.75	0.000	**0.172**	0.000	0.000	0.000	0.000	0.022	0.000
C	0.25	0.000	0.000	**0.970**	0.000	0.000	0.000	0.000	0.003
	0.5	0.000	0.000	**0.978**	0.000	0.000	0.000	0.000	0.000
	0.75	0.000	0.000	**0.909**	0.000	0.000	0.000	0.000	0.000
G	0.25	0.001	0.002	0.000	**0.565**	0.000	0.000	0.009	0.019
	0.5	0.000	0.000	0.000	**0.442**	0.000	0.000	0.020	0.023
	0.75	0.000	0.000	0.000	**0.877**	0.000	0.000	0.001	0.000
F	0.25	0.051	0.034	0.000	0.006	**0.512**	0.000	0.022	0.000
	0.5	0.000	0.001	0.000	0.000	**0.965**	0.000	0.000	0.000
	0.75	0.000	0.000	0.000	0.000	**1.000**	0.000	0.000	0.000
J	0.25	0.000	0.000	0.000	0.004	0.000	**0.538**	0.000	0.000
	0.5	0.000	0.000	0.000	0.000	0.000	**0.948**	0.000	0.005
	0.75	0.000	0.000	0.000	0.001	0.000	**0.776**	0.000	0.001

Goodness-of-fit (see 5.4.3 (i))

True	τ	N	t	C	G	F
N	0.25	**0.770**	0.093	0.000	0.026	0.085
	0.5	**0.864**	0.094	0.000	0.003	0.006
	0.75	**0.864**	0.098	0.000	0.000	0.000
t (df=4)	0.25	0.249	**0.508**	0.001	0.117	0.010
	0.5	0.420	**0.443**	0.000	0.021	0.000
	0.75	0.590	**0.276**	0.000	0.000	0.012
C	0.25	0.001	0.004	**0.943**	0.000	0.000
	0.5	0.000	0.000	**0.945**	0.000	0.000
	0.75	0.000	0.000	**0.954**	0.000	0.000
G	0.25	0.022	0.021	0.000	**0.924**	0.005
	0.5	0.004	0.001	0.000	**0.967**	0.000
	0.75	0.001	0.000	0.000	**0.964**	0.000
F	0.25	0.030	0.014	0.000	0.000	**0.915**
	0.5	0.000	0.000	0.000	0.000	**0.964**
	0.75	0.000	0.000	0.000	0.000	**0.968**
J	0.25	0.000	0.000	0.000	0.165	0.000
	0.5	0.000	0.000	0.000	0.000	0.000
	0.75	0.000	0.000	0.000	0.000	0.000

Table 5.7: Percentages of selected copula families of $R = 1000$ repetitions with $n = 1000$ simulated observations each. Parameters of the true copulas (as indicated in the left column) are chosen according to three choices of Kendall's τ (cp. Table 2.1).
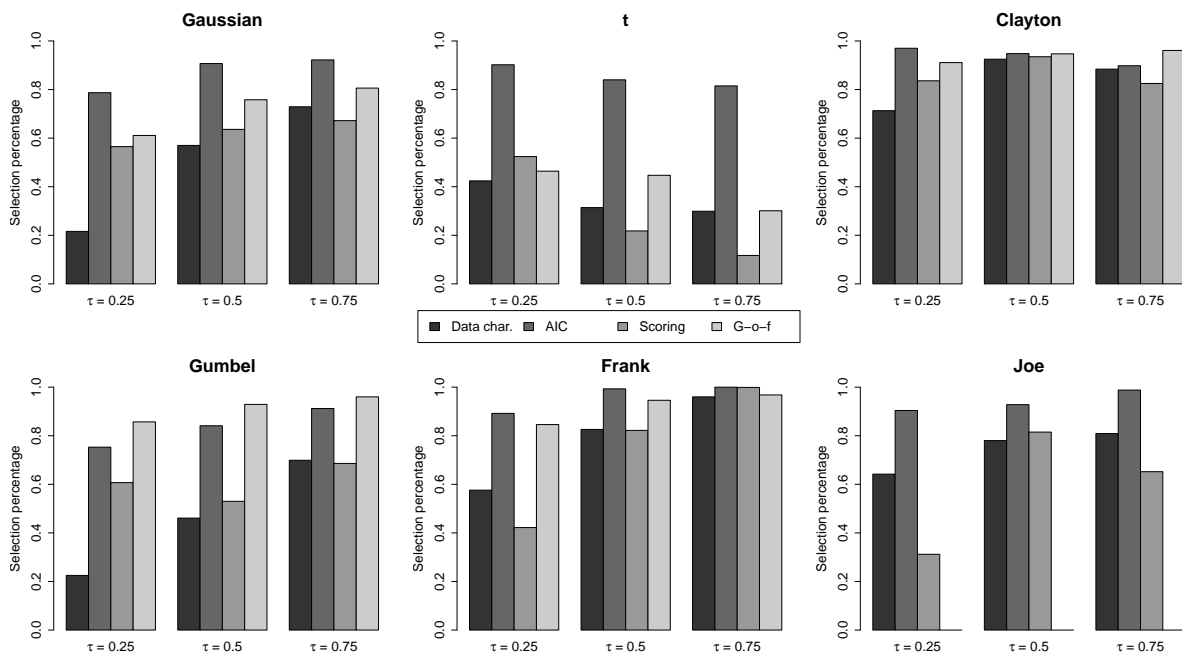
Figure 5.4: Illustration of Table 5.6: percentages of correctly selected copula families of $R = 1000$ repetitions with $n = \mathbf{500}$ simulated observations each.
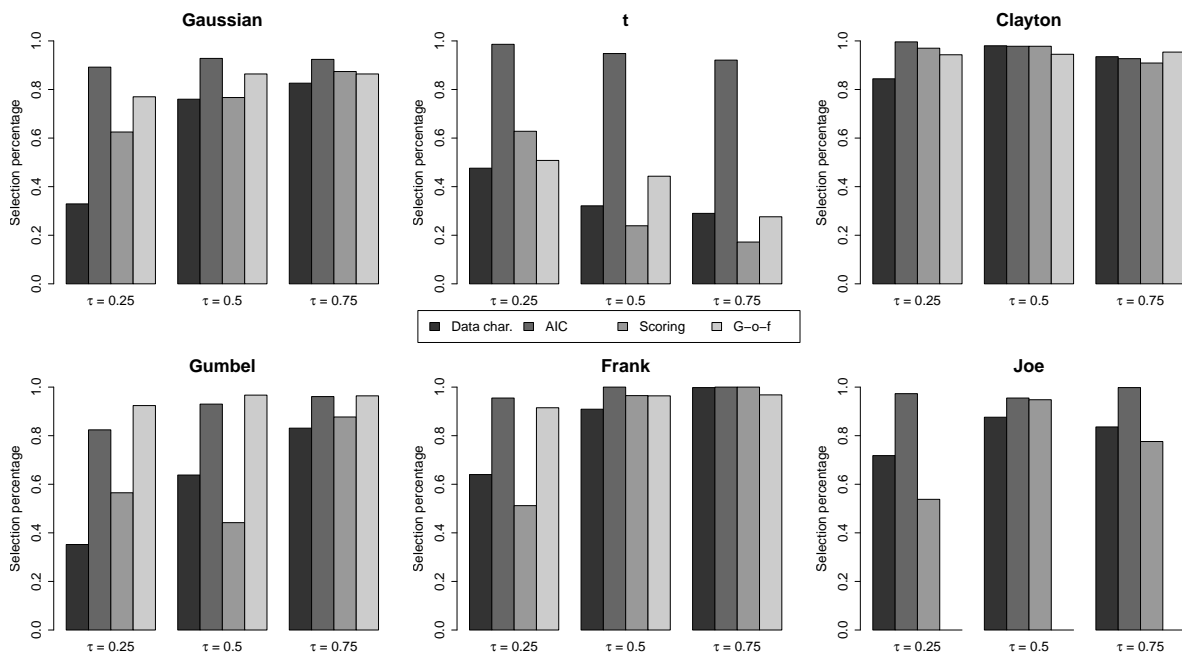
Figure 5.5: Illustration of Table 5.7: percentages of correctly selected copula families of $R = 1000$ repetitions with $n = \mathbf{1000}$ simulated observations each.

process has already been discussed in the detailed studies by Genest et al. (2009) and Berg (2009).

The results of our simulation study are shown in Tables 5.6 and 5.7 and illustrated in Figures 5.4 and 5.5. t copulas with more than 30 degrees of freedom are not considered further, since they are too close to the Gaussian. Similarly, BB1 copulas with $\theta \leq 0.1$ or $\delta \leq 1.1$ are too close to Clayton and Gumbel copulas, respectively, while BB7 copulas are too close to Clayton and Joe copulas if $\theta \leq 1.1$ and $\delta \leq 0.1$, respectively (cp. Sections 2.1.3 and, in particular, 5.4.2). Possible differences of the sum of percentages to 100% result from the fact that the scoring method cannot select among families with equal score and that goodness-of-fit testing can show that no family is significant (e.g., in the Joe copula case quite often no copula is selected in accordance to the fact that Joe copulas are not considered in goodness-of-fit testing). The results can be summarized as follows:

(i) The accuracy of all selection procedures increases with increasing Kendall's τ, i.e., with increasing dependency, and with increasing number of observations, unless observations come from a t copula or, to some extent, from a Clayton copula.

(ii) Observations simulated from t and Clayton copulas are repeatedly selected as BB1 and BB7 copulas, while in some cases Gaussian observations are confounded with the Frank copula as well as Gumbel observations with the Joe copula and vice versa. These confusions are due to similar copula characteristics (see Table 2.1) which is underlined by the KLIC's between bivariate copula families as discussed above.

(iii) The correct identification of observations from the t copula is the hardest task in most cases. Even the goodness-of-fit tests fail rather often. This has also been shown in the numerical studies by Berg (2009) and corresponds to the KLIC's which are rather small compared to alternative copula families (see Tables 5.1, 5.2 and 5.3).

(iv) Copula selection based on the AIC has the highest accuracy in the majority of cases. Mostly it is even superior to the blanket goodness-of-fit test, although this test considers less alternatives!

Even if the different copula families are not nested and thus AIC comparisons are subject to an increased uncertainty, the AIC turns out to be the best criterion for copula selection. Moreover, selection based on data characteristics (empirical Kendall's τ and exceedance dependence) performs surprisingly well and similarly good to the scoring method based on the Vuong test whose performance is not satisfying given that it is based on statistical tests (also cp. the results in Belgorodski (2010) who considers slightly different alternatives but obtains similar results). The scoring approach apparently requires some adjustments to be an adequate selection method. One idea might be to combine goodness-of-fit testing and the scoring approach in a two-step procedure similar to the one discussed above: use goodness-of-fit tests to determine which copulas are eligible and then select among these copulas using the scoring approach.

To sum it up, copula selection using the AIC is not just an ad-hoc approach which is computationally more efficient than goodness-of-fit testing, but also a reliable and mostly superior method for omnibus use. Moreover our study confirmed the results of Manner (2007) that the performance of selection using the AIC improves considerably when dependence and sample sizes increase.
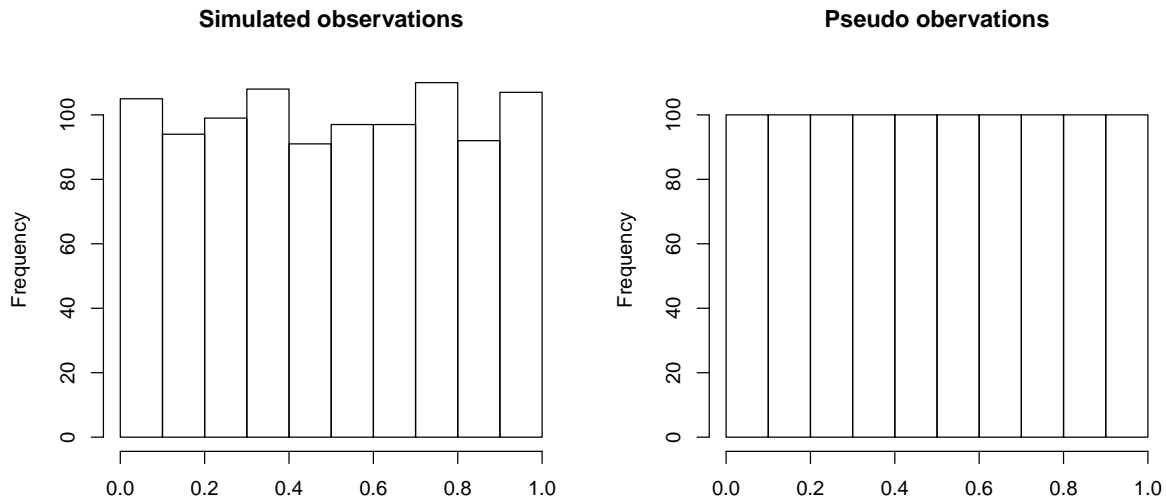
Figure 5.6: Histograms of $n = 1000$ observations simulated from a Gaussian copula with $\rho = 0.5$ and corresponding pseudo-observations.

Remark: Goodness-of-fit tests for simulated data

The p-value computation of the goodness-of-fit test based on the empirical copula process, which we discussed in Section 4.2.2 and which we use here, relies on the assumption that the data is rank-based. This is the case when working with pseudo-observations as defined in (4.4). However, when we simulate copula data, it exhibits different properties, although its margins are also approximately uniformly distributed on $[0, 1]$. Figure 5.6 illustrates the problem: the histogram of $n = 1000$ observations simulated from a Gaussian copula with $\rho = 0.5$ is rather "bumpy", while the histogram of the corresponding pseudo-observations is even which reflects the marginal uniformity. Of course, by the law of large numbers, the simulated copula data will behave like their corresponding pseudo-observations, but for finite sample sizes there is a simulation error induced by the intrinsic randomness of simulation. Hence, simulated copula data **cannot** be used directly for p-value computations. Otherwise, the results are wrong and misleading (p-values equal to 0 even for the true underlying copula). Therefore, we calculated pseudo-observations of the simulated data in the simulation study above.

Chapter 6

Truncation of regular vines

After having discussed various issues of model building in the previous chapters, we are now ready to establish procedures that facilitate the construction of R-vine models which is essential in high dimensional applications and under time or resource restrictions. In a first step, we therefore consider the issue of *truncation* of R-vines which corresponds to setting copulas in higher order trees to the independence copula, i.e., assuming independence after a certain R-vine tree. The notation of truncation will be discussed and defined in the first part of this chapter. Subsequently, we develop two truncation procedures based on the information criteria and the Vuong test which we described in Chapter 5. These procedures then also motivate the *simplification* procedures, which are discussed in Chapter 7, and where simplification refers to a more general way of facilitating R-vine construction, of which truncation is an important special case.

6.1 Motivation

The idea of R-vine *truncation* is to simplify all higher order trees after a certain tree K, where K is called the *truncation level*, by setting all pair copula terms which involve a conditioning set of size larger or equal to K to independence copulas. This means that we simplify the model to the greatest possible extent and facilitate maximum likelihood estimation and simulation for larger dimensions.

Note that this definition of truncation of R-vines differs from the notation of "truncation" in Valdesogo (2009) which corresponds to our definition of simplification (see Section 7.1). We however believe that our distinction of simplification and truncation (where the latter is not explicitly discussed in Valdesogo (2009)) is less ambiguous and clearly highlights the different purposes.

Before we move on to formal definitions of the points discussed above, we present a simple example.

Example 4 (Truncation.) *Let us consider a D-vine on five variables (cp. Figure 2.5), i.e., we have to specify ten pair copulas within four trees (see Figure 6.1).*

Now assume that we have sequentially specified the first two trees T_1 and T_2 with appropriate pair copulas and then detect that we can truncate our D-vine after tree $K = 2$, since all important dependencies were already captured in the first trees and we can neglect

93

$$
\begin{array}{llll}
T_1 : c_{12} & c_{23} & c_{34} & c_{45} \\
T_2 : & c_{13|2} & c_{24|3} & c_{35|4} \\
T_3 : & c_{14|23} & c_{25|34} \\
T_4 : & c_{15|234} &
\end{array}
$$

Figure 6.1: Pair copula terms of a five-dimensional D-vine (cp. Figure 6.1).

$$
\begin{array}{llll}
T_1 : c_{12} & c_{23} & c_{34} & c_{45} \\
T_2 : & c_{13|2} & c_{24|3} & c_{35|4} \\
\hline
T_3 : & \pi_{14|23} & \pi_{25|34} \\
T_4 : & \pi_{15|234} &
\end{array}
$$

Figure 6.2: Pair copula terms of a five-dimensional D-vine truncated after the second tree T_2, where $\pi_{ij|D}$ denote independence copulas.

the remaining dependencies. Then all copulas in trees T_3 and T_4 can be set to independence copulas $\pi_{ij|D}$ as illustrated in Figure 6.2 (of course, $\pi_{ij|D} = 1$, but we will always write $\pi_{ij|D}$ in the following in order to clarify that we are dealing with copulas).

Note that for time-ordered nodes $1, ..., 5$ truncation corresponds to a second order Markov process, since X_{i+3} is independent of X_i given X_{i+2} and X_{i+1} ($i = 1, 2$; cp. Section 6.1.1).

These considerations motivate the following notations. An R-vine which has been truncated after tree T_K will be called a *pairwisely truncated R-vine at level K*. The additional notation "pairwisely" refers to the fact that we simplify *pair* copulas in contrast to a *joint* multivariate copula which we will consider later on for C-vines in Chapter 8.

Obviously, truncation facilitates maximum likelihood estimation in larger dimensions, since independence copulas have no parameters at all and thus, using the R-vine density given in (2.27), the *parametrized density of a truncated R-vine at level K* is given for $\boldsymbol{u} \in [0, 1]^d$ by

$$
c_T^K(\boldsymbol{u}|\boldsymbol{\theta}_T(K)) = \prod_{i=1}^{K} \prod_{e \in E_i} c_{j(e),k(e)|D(e)}, \tag{6.1}
$$

where the arguments of the copulas have been omitted for simplicity (cp. densities in Section 2.4) and $\boldsymbol{\theta}_T(K)$ is the parameter set of c_T^K, i.e.,

$$
\boldsymbol{\theta}_T(K) = \{\boldsymbol{\theta}_{j(e),k(e)|D(e)} : \ e \in E_i, \ i = 1, ..., K\},
$$

with $\boldsymbol{\theta}_{j(e),k(e)|D(e)}$ denoting the parameter(s) of the copula $c_{j(e),k(e)|D(e)}$. As stated above, truncation then evidently simplifies models to the greatest possible extent, especially when $K \ll d - 1$.

In the following we want to determine if an R-vine can be truncated and detect the appropriate truncation level. Before we turn to the discussion of appropriate procedures, we briefly consider truncated D-vines and their relationship to Markov processes.

"SMALL" MODEL				"FULL" MODEL			
$T_1 : c_{12}$	c_{23}	c_{34}	c_{45}	$T_1 : c_{12}$	c_{23}	c_{34}	c_{45}
$T_2 :$	$c_{13\|2}$	$c_{24\|3}$	$c_{35\|4}$	$T_2 :$	$c_{13\|2}$	$c_{24\|3}$	$c_{35\|4}$
$T_3 :$	$\pi_{14\|23}$	$\pi_{25\|34}$		$T_3 :$	$c_{14\|23}$	$c_{25\|34}$	
$T_4 :$	$\pi_{15\|234}$			$T_4 :$	$\pi_{15\|234}$		

Figure 6.3: Pair copula terms of five-dimensional D-vines truncated after the second tree T_2 ("small" model) and after the third tree T_3 ("full" model), respectively, where $\pi_{ij|D}$ denote independence copulas.

6.1.1   Truncated D-vines as Markov processes

In Example 4 we saw that a pairwisely truncated D-vine at level $K = 2$ with five time ordered nodes corresponds to a second order Markov process on time points 1 to 5. This result obviously also holds in general. According to Equations (2.29) and (6.1), the parametrized density of a pairwisely truncated D-vine at level K can be written as

$$c_T^K(\boldsymbol{u}|\boldsymbol{\theta}_T(K)) = \prod_{j=1}^{K}\prod_{i=1}^{d-j} c_{i,(i+j)|(i+1):(i+j-1)}.$$

This means that X_{i+K+1} is independent of X_i given $X_{i+K}, ..., X_{i+1}$ for $i = 1, ..., d-K-1$, i.e., a pairwisely truncated D-vine at level K with time ordered nodes corresponds to a K-th order Markov process on time points 1 to d, since

$$P(X_{i+K+1}|X_{i+K}, ..., X_{i+1}, X_i, ..., X_1) = P(X_{i+K+1}|X_{i+K}, ..., X_{i+1}),$$

for all $i = 1, ..., d-K-1$.

6.2   Iterative pairwise truncation based on the Vuong test

The first truncation procedure we consider is based on the Vuong test described in Section 5.3. To give an understanding of our iterative approach, we first consider an illustrative example.

Example 5 (Stepwise truncation based on the Vuong test.) *Let us consider again the five-dimensional D-vine of Example 4 and assume again that we have already appropriately specified the pair copulas of the first two trees T_1 and T_2. We now want to determine whether the D-vine can be truncated at this level.*

Our truncation procedure directly exploits the concept that usually most dependencies are captured in the first trees. The idea is to specify the next tree, in this case T_3, with appropriate pair copulas and then compare the "full" model on three trees to the "small" model on two trees, where higher order trees are truncated (cp. Figure 6.3).

Note that the "full" model is not exactly "full" as tree T_4 has also been modeled as independent. Hence this is not an exact model comparison between a truncated and a fully

---

Algorithm 6 Truncation of R-vines based on the Vuong test.

Input: (Pseudo-)observations of d variables, significance level α.

  1: **for** $j = 0, ..., d - 2$ **do**

  2:     Specify model $\mathcal{M}_T(j+1)$ by constructing tree T_{j+1} and specifying appropriate pair copulas (refer to Section 7.4 for more details on this issue).

  3:     Perform a Vuong test for comparing models $\mathcal{M}_T(j)$ (model 1) and $\mathcal{M}_T(j+1)$ (model 2), i.e., determine test statistic ν as in (5.6).

  4:     **if** $\nu > -\Phi^{-1}\left(1 - \frac{\alpha}{2}\right)$ **then**

  5:        Truncate the R-vine at level $K = j$, i.e., exit the loop with model $\mathcal{M}_T(j)$.

  6:     **end if**

  7: **end for**

Output: Pairwisely truncated R-vine at level K, or fully specified R-vine, if no truncation is possible.

---

specified D-vine, but only an approximation to the truth. However, under the assumption that most dependencies are captured in the first trees, this should be a reasonable approximation. In other words, the pair copula terms in tree T_3 quantify the possible gain if we fit this additional tree. If this marginal gain is too "small", we decide that we have already captured the most important dependencies in the first trees and truncate at level $K = 2$.

This model comparison can now be established by the Vuong test with or without one of the corrections proposed in Section 5.3.1. If the "full" model is preferred, it is apparently still too early to truncate the D-vine, i.e., the gain of the additional tree is still too large. If, on the other hand, the "small" model is preferred, we can truncate the D-vine at truncation level $K = 2$. However, the question remains what happens if the null hypothesis that one cannot distinguish between the models is not rejected. In this case, we also truncate at level $K = 2$, since according to the Vuong test both models are equally good and we are interested in the simplest model possible.

Truncation in five dimensions may seem rather silly, but as soon as we move on to higher dimensions, things look quite different. Hence, we now state the simplification procedure at level j in general form. For this, let $\mathcal{M}_T(K)$ be the statistical model for an i.i.d. sample $\boldsymbol{u} = (\boldsymbol{u}_1', ..., \boldsymbol{u}_n')'$ with density $c_T^K(\boldsymbol{u}|\boldsymbol{\theta}_T(K))$ as given in (6.1), where $\boldsymbol{u}_i = (u_{i1}, ..., u_{id})'$, $i = 1, ..., n$. This corresponds to modeling the dependence in the first K trees and then truncating. $\mathcal{M}_T(0)$ therefore denotes the statistical model where all variables are jointly independent.

Using this notation we can formulate Algorithm 6. For $j = 0$ a pre-test is performed, which tests whether the R-vine can be truncated after tree "0", i.e., whether the variables are jointly independent. α is the significance level used in all tests.

Note that line 2 is stated rather informally. A more detailed description is given in Section 7.4 which brings together all the methods discussed in the chapters up to that point to establish a general framework for R-vine model building.

In the next section, we refine the truncation procedure discussed above and exploit the fact that $\mathcal{M}_T(j)$ and $\mathcal{M}_T(j + 1)$ are nested models.

Algorithm 7 Truncation of R-vines using information criteria.

Input: (Pseudo-)observations of d variables.

1: **for** $j = 0, ..., d - 2$ **do**
2:    Specify model $\mathcal{M}_T(j+1)$ by constructing tree T_{j+1} and specifying appropriate pair copulas (refer to Section 7.4 for more details on this issue).
3:    Compute the AIC (5.3) for models $\mathcal{M}_T(j)$ (model 1) and $\mathcal{M}_T(j+1)$ (model 2).
4:    **if** $AIC_1 < AIC_2$ **then**
5:       Truncate the R-vine at level $K = j$, i.e., exit the loop with model $\mathcal{M}_T(j)$.
6:    **end if**
7: **end for**
Output: Pairwisely truncated R-vine at level K, or fully specified R-vine, if no truncation is possible.

6.3   Iterative pairwise truncation using information criteria

In this section we propose an alternative sequential truncation procedure which is simpler than the one discussed in the previous section. Primarily the determination whether truncation is possible or not is very similar to the procedure based on the Vuong test. Given already specified trees $T_1, ..., T_j$, we want to compare the "small" model $\mathcal{M}_T(j)$ on j trees to the "full" model $\mathcal{M}_T(j+1)$ on $j+1$ trees. Higher order trees are truncated in both models (cp. Example 5). In contrast to the procedure based on the Vuong test, we then exploit that $\mathcal{M}_T(j+1)$ and $\mathcal{M}_T(j)$ are obviously nested, since $\boldsymbol{\theta}_T(j) \subseteq \boldsymbol{\theta}_T(j+1)$. Thus we can simply compare AIC's or BIC's of both models and chose the model with the respective smaller criterion without having to worry about an increased variability as discussed in Section 5.2. If the "small" model $\mathcal{M}_T(j)$ is preferred, we can hence truncate the R-vine.

Similar to Algorithm 6 we now state the truncation procedure using information criteria in general form in Algorithm 7. For reasons of readability the algorithm is formulated using the AIC. Of course, the BIC can be used instead.

The AIC/BIC comparison is of course slightly faster than performing a Vuong test. Nevertheless, this approach also motivates a corresponding simplification procedure, which crucially takes advantage of the AIC/BIC and some additional assumptions to increase the computational efficiency. The issue of simplification will be discussed in the following Chapter 7.

Chapter 7

Simplification of regular vines

Since truncation as discussed in Chapter 6 may be a rather strong assumption in many cases, we will now concentrate on an alternative way to "simplify" the construction of R-vines. Hence, we begin by defining what exactly we mean when dealing with *simplification* in the context of R-vines and show that truncation is a special case of this concept. Motivated by the R-vine truncation procedures discussed on Sections 6.2 and 6.3, we then develop procedures for the determination of appropriate simplification levels.

Finally, we discuss some implementation issues to see how these simplification procedures can be applied to practical problems and state an algorithm for general R-vine specification which proceeds hierarchically and which incorporates all aspects of model building discussed up to this point. The algorithms will then be studied in the following Chapters 10 and 11.

7.1   Simplification versus truncation

In contrast to R-vine truncation, the idea of R-vine *simplification* is to replace all pair copulas in higher order trees by bivariate Gaussian copulas (see Section 2.1.3), i.e., after a certain tree K, where K is called the *simplification level*, all pair copula terms with conditioning set of size larger or equal to K are replaced by Gaussian copulas. Obviously, truncation is then a special case of simplification, since it corresponds to bivariate Gaussian copulas with parameter 0. There are several reasons why we have chosen Gaussian copulas instead of any other copula family:

  (i) Gaussian copulas are symmetric. The use of an asymmetric copula family such as the Clayton or Gumbel with lower and upper tail dependence, respectively, is not sensible, since it imposes a rather restrictive structure on all pair copulas in higher order trees. Of course, we could consider families with parameters to specify different tail dependence such as the BB1 and BB7 families, but these families are computationally more demanding to specify and therefore contradict the idea of simplification.

  (ii) Gaussian copulas are easy to specify. Compared to other symmetric copula families such as the t or Frank, Gaussian copulas are rather easy to specify and to interpret in

$$
\begin{array}{lcccc}
T_1: & c_{12} & c_{23} & c_{34} & c_{45} \\
T_2: & c_{13|2} & c_{24|3} & c_{35|4} \\
\hline
T_3: & c^{\rho}_{14|23} & c^{\rho}_{25|34} \\
T_4: & c^{\rho}_{15|234}
\end{array}
$$

Figure 7.1: Pair copula terms of a five-dimensional D-vine simplified after the second tree T_2, where $c^{\rho}_{ij|D}$ denote Gaussian pair copulas.

terms of the correlation parameter. Moreover, when considering t copulas, despite the increased computational effort for estimating the degrees of freedom (which again contradicts the idea of simplification), one may frequently obtain large degrees of freedom so that one ends up with an almost-Gaussian copula anyway. (Usually we consider t copulas with more than 30 degrees of freedom as Gaussian copulas.)

(iii) Simulation from large numbers of Gaussian copulas is fast. After having specified a model, one often wants to simulate observations, e.g., to obtain Value-at-risk estimates. This is much easier if the model consists of a considerable number of bivariate Gaussian copulas.

As noted in Section 6.1, our notations of simplification and truncation should not be mixed up with what Valdesogo (2009) denotes as "truncation", since this corresponds to our definition of simplification. Moreover, our notation of simplification is not to be confounded with "simplified pair copula constructions" as considered in Hobæk Haff et al. (2010). As mentioned in Section 2.1.4, such simplified pair copula constructions refer to the fact that inference from pair copula constructions relies on the simplifying assumption that all the pair copulas do not directly depend on the conditioning variables, but only through the two conditional distribution functions as arguments.

As for truncation, we will first consider a simple example before we move on to formal definitions of the points discussed above.

Example 6 (Simplification.) *We consider the same D-vine on five variables as in Example 4 (cp. Figure 2.5) which amounts to investigating ten pair copulas within four trees. Let us assume that the first two trees T_1 and T_2 have already been sequentially specified with appropriate pair copulas and we now detect that we can specify the remaining trees T_3 and T_4 with Gaussian copulas, i.e., simple Gaussian copulas are sufficient to capture the remaining dependencies, while the "more complicated" dependencies are modeled in the first two trees. Thus we can simplify our D-vine after tree $K = 2$ and obtain the model illustrated in Figure 7.1.*

Similarly to a pairwisely truncated R-vine, an R-vine which has been simplified with Gaussian pair copulas after tree T_K will be called a *pairwisely simplified K level R-vine*. Obviously, simplification also facilitates maximum likelihood estimation in larger dimensions, since Gaussian copulas have only one parameter, which is easy to estimate. Hence, considering the R-vine density given in (2.27), the *parametrized density of a simplified K*

level R-vine is given by

$$c_S^K(\boldsymbol{u}|\boldsymbol{\theta}_S(K)) = \left[\prod_{i=1}^{K}\prod_{e\in E_i} c_{j(e),k(e)|D(e)}\right] \times \left[\prod_{i=K+1}^{d-1}\prod_{e\in E_i} c_{j(e),k(e)|D(e)}^{\rho}\right], \qquad (7.1)$$

where $\boldsymbol{u} \in [0,1]^d$ and the arguments of the copulas have again been omitted for simplicity. Further, $c_{j(e),k(e)|D(e)}^{\rho}$ denote Gaussian pair copulas with correlation parameter $\rho_{j(e),k(e)|D(e)}$ and $\boldsymbol{\theta}_S(K)$ is the parameter set of c_S^K, i.e.,

$$\begin{aligned}\boldsymbol{\theta}_S(K) = &\{\boldsymbol{\theta}_{j(e),k(e)|D(e)} :\ e \in E_i,\ i = 1, ..., K\} \\ &\cup \{\rho_{j(e),k(e)|D(e)} :\ e \in E_i,\ i = K+1, ..., d-1\},\end{aligned} \qquad (7.2)$$

with $\boldsymbol{\theta}_{j(e),k(e)|D(e)}$ denoting the parameter(s) of the copula $c_{j(e),k(e)|D(e)}$. Here we do not denote Gaussian pair copulas by c^N as before in order to avoid confusion with the simplified/truncated density c^K.

Note that in the special case of C-vines we will see that the second part of Equation (7.1) collapses to the density of a multivariate Gaussian copula (see Section 8.1).

Since truncation allows the greatest possible simplification, we usually consider it first. Either by running a truncation procedure for the full R-vine first or by a sequential approach which considers truncation before simplification with Gaussian pair copulas in each step – this will be referred to as *hierarchical* in the following and discussed in Section 7.4, where we combine the procedures proposed in this and the previous chapters to obtain a hierarchical R-vine specification algorithm for omnibus use. Firstly, we concentrate on the first option, i.e., we want to determine whether an R-vine can be simplified irrespectively of possible truncation and, if yes, find an adequate simplification level. Therefore we develop appropriate procedures which are motivated by the respective truncation methods discussed in Sections 6.2 and 6.3

7.2   Iterative pairwise simplification based on the Vuong test

As before, our first simplification procedure is based on the Vuong test which we discussed in Section 5.3. The idea of the stepwise procedure is very similar to the one of the corresponding truncation procedure described in Section 6.2 and illustrated in the following example.

Example 7 (Stepwise simplification based on the Vuong test.) *As in Examples 5 and 6, we consider again a five-dimensional D-vine with adequately specified pair copulas in the first two trees T_1 and T_2. We are now interested in determining whether the D-vine can be simplified at this level, i.e., modeled with Gaussian pair copulas for trees T_3 and T_4.*

The approach here proceeds on the same lines as for truncation (cp. Section 6.2): we exploit the concept that usually most dependencies are captured in the first trees and only specify the next tree T_3 with appropriate pair copulas (not necessarily Gaussian). Then we

<div align="center">

"SMALL" MODEL   "FULL" MODEL

</div>

$T_1: c_{12}$	c_{23}		c_{34}	c_{45}	$T_1: c_{12}$	c_{23}	c_{34}	c_{45}
$T_2:$	$c_{13\|2}$	$c_{24\|3}$		$c_{35\|4}$	$T_2:$	$c_{13\|2}$	$c_{24\|3}$	$c_{35\|4}$
$T_3:$		$c^\rho_{14\|23}$	$c^\rho_{25\|34}$		$T_3:$		$c_{14\|23}$	$c_{25\|34}$
$T_4:$			$c^\rho_{15\|234}$		$T_4:$			$\tilde{c}^\rho_{15\|234}$

Figure 7.2: Pair copula terms of five-dimensional D-vines simplified after the second tree T_2 ("small" model) and after the third tree T_3 ("full" model), respectively, where $c^\rho_{ij\|D}$ and $\tilde{c}^\rho_{ij\|D}$ denote Gaussian pair copulas.

compare the "full" model on three trees to the "small" model on two trees, where higher order trees are modeled with Gaussian pair copulas (cp. Figure 7.2).

It is important to note that, in general, for the correlation parameters of the Gaussian pair copulas in the fourth tree T_4 of the "small" model, $c^\rho_{15\|234}$, and the "full" model, $\tilde{c}^\rho_{15\|234}$, it holds that $\rho_{15\|234} \neq \tilde{\rho}_{15\|234}$, since the specification of tree T_3 is different in both models and hence the transformed observations based on tree T_3 are different, too.

As for truncation, we now compare both models with the Vuong test and simplify the D-vine at level $K = 2$ if the "full" model is not preferred, i.e., if the additional gain of a fully specified third tree in contrast to a simplified third tree with Gaussian copulas is not too large.

In order to state the simplification procedure at level j in general form, we need some additional notations. Corresponding to $\mathcal{M}_T(K)$, we define $\mathcal{M}_S(K)$ as the statistical model for an i.i.d. sample $\boldsymbol{u} = (\boldsymbol{u}_1', ..., \boldsymbol{u}_n')'$ with density $c_S^K(\boldsymbol{u}|\boldsymbol{\theta}_S(K))$ as given in (7.1), where $\boldsymbol{u}_i = (u_{i1}, ..., u_{id})'$, $i = 1, ..., n$, i.e., $\mathcal{M}_S(K)$ is the model with fully modeled dependence in the first K trees and then simplified higher order trees. Accordingly, $\mathcal{M}_S(0)$ denotes the statistical model where all variables are jointly normally distributed. Note however that for two models $\mathcal{M}_S(j)$ and $\mathcal{M}_S(j + 1)$ in general $\rho_{j(e),k(e)|D(e)} \neq \tilde{\rho}_{j(e),k(e)|D(e)}$ for all $e \in E_i$, $i \geq j + 1$, $\rho_{j(e),k(e)|D(e)} \in \boldsymbol{\theta}_S(j)$ and $\tilde{\rho}_{j(e),k(e)|D(e)} \in \boldsymbol{\theta}_S(j + 1)$ if the trees T_i, $i \geq j + 1$, are constructed in exactly the same way, which does not necessarily hold, since the transformed observations from trees T_{j+1} in both models are different due to the different copula choices in T_{j+1}, i.e., trees may contain different edges and thus different pair copula terms.

Similarly to Algorithm 6 the simplification procedure based on the Vuong test can then be formulated in Algorithm 8 with a pre-test for $j = 0$, i.e., a test whether the variables are jointly normal and thus all pair copulas can be set to bivariate Gaussian copulas corresponding to a multivariate Gaussian copula for all variables (cp. Section 8.1.1). A more detailed description of the pair copula specification in lines 2 and 3 of Algorithm 8 is given in Section 7.4. The significance level is chosen as α in all tests.

Algorithm 8 Simplification of R-vines based on the Vuong test.

Input: (Pseudo-)observations of d variables, significance level α.

1: **for** $j = 0, ..., d - 2$ **do**
2: Specify model $\mathcal{M}_S(j)$ by constructing higher order trees $T_{j+1}, ..., T_{d-1}$ with bivariate Gaussian copulas.
3: Specify model $\mathcal{M}_S(j + 1)$ by additionally constructing tree T_{j+1} and specifying appropriate pair copulas (refer to Section 7.4 for more details on this issue), and by constructing higher order trees $T_{j+2}, ..., T_{d-1}$ with bivariate Gaussian copulas.
4: Perform a Vuong test for models $\mathcal{M}_S(j)$ (model 1) and $\mathcal{M}_S(j + 1)$ (model 2), i.e., determine test statistic ν as in (5.6).
5: **if** $\nu > -\Phi^{-1}\left(1 - \frac{\alpha}{2}\right)$ **then**
6:   Simplify the R-vine at level $K = j$, i.e., exit the loop with model $\mathcal{M}_S(j)$.
7: **end if**
8: **end for**

Output: Pairwisely simplified K level R-vine, or fully specified R-vine, if no simplification is possible.

7.3 Iterative pairwise simplification using information criteria

Motivated by the truncation approach in Section 6.3, we propose an alternative sequential simplification procedure. Using information criteria such as the AIC and BIC, which we described in Section 5.2, we obtain a simple procedure which however relies on some additional approximations compared to the approach based on the Vuong test. This procedure can therefore be regarded as a "quick and dirty" approach to the simplification issue for applications in very high dimensions and/or under strict time and resource restrictions.

In contrast to the case of truncation, models are not nested anymore when testing for potential simplification ($\boldsymbol{\theta}_S(j) \nsubseteq \boldsymbol{\theta}_S(j + 1)$), which means that we have to deal with an increased variability in AIC comparisons (see Section 5.2). The following considerations show that it might however be suitable to use information criteria for model selection.

The main assumption that motivates the consideration of simplification and truncation is, as already stressed before, that most dependency is captured in the first trees while higher order trees only account for small remaining dependencies. Hence we can assume that, if we consider models $\mathcal{M}_S(j)$ and $\mathcal{M}_S(j + 1)$, then the bivariate Gaussian copulas of the simplified trees have very similar parameters, i.e., we can ignore trees $T_{j+2}, ..., T_{d-1}$ in a likelihood ratio-based approach such as the Vuong test or the difference of AIC's or BIC's, since

$$AIC_{\mathcal{M}_S(j)} - AIC_{\mathcal{M}_S(j+1)}$$

$$= \left[-2\sum_{i=1}^{n} \log c_S^j(\boldsymbol{u}_i|\boldsymbol{\theta}_S(j)) + 2k_j\right] - \left[-2\sum_{i=1}^{n} \log c_S^{j+1}(\boldsymbol{u}_i|\boldsymbol{\theta}_S(j + 1)) + 2k_{j+1}\right]$$

$$= -2\sum_{i=1}^{n} \log \left[\frac{c_S^j(\boldsymbol{u}_i|\boldsymbol{\theta}_S(j))}{c_S^{j+1}(\boldsymbol{u}_i|\boldsymbol{\theta}_S(j + 1))}\right] + 2(k_j - k_{j+1})$$

$$
= -2\sum_{i=1}^{n} \log \left[\frac{\left[\prod_{e\in E_{j+1}} c_{j(e),k(e)|D(e)}^{\rho}\right] \times \left[\prod_{\ell=j+2}^{d-1} \prod_{e\in E_\ell} c_{j(e),k(e)|D(e)}^{\rho}\right]}{\left[\prod_{e\in E_{j+1}} c_{j(e),k(e)|D(e)}\right] \times \left[\prod_{\ell=j+2}^{d-1} \prod_{e\in \tilde{E}_\ell} \tilde{c}_{j(e),k(e)|D(e)}^{\rho}\right]} \right] + 2(k_j - k_{j+1})
$$

$$
\approx -2\sum_{i=1}^{n} \log \left[\frac{\prod_{e\in E_{j+1}} c_{j(e),k(e)|D(e)}^{\rho}}{\prod_{e\in E_{j+1}} c_{j(e),k(e)|D(e)}} \right] + 2(k_j - k_{j+1}),
$$

where the density of a simplified j level R-vine, c_S^j, is given in (7.1) and k_j and k_{j+1} denote the number of parameters of models $\mathcal{M}_S(j)$ and $\mathcal{M}_S(j+1)$, respectively. A similar derivation holds for the Vuong test and the BIC. Note that in the case of truncation, the last approximate equality is exact and the numerator of the last fraction reduces to 1, i.e.,

$$
\begin{aligned}
AIC_{\mathcal{M}_T(j)} - AIC_{\mathcal{M}_T(j+1)} &= -2\sum_{i=1}^{n} \log \left[\frac{\prod_{e\in E_{j+1}} \pi_{j(e),k(e)|D(e)}}{\prod_{e\in E_{j+1}} c_{j(e),k(e)|D(e)}} \right] + 2(k_j - k_{j+1}) \\
&= 2\sum_{i=1}^{n} \log \left[\prod_{e\in E_{j+1}} c_{j(e),k(e)|D(e)} \right] + 2(k_j - k_{j+1}).
\end{aligned}
\tag{7.3}
$$

Model $\mathcal{M}_S(j+1)$ with the same Gaussian copulas as in $\mathcal{M}_S(j)$ for trees $T_{j+2},...,T_{d-1}$ will be denoted by $\widehat{\mathcal{M}}_S(j+1)$ as it is only an approximation to model $\mathcal{M}_S(j+1)$, i.e., we assume that T_i, $i \geq j+1$, are constructed in exactly the same way in both models $\mathcal{M}_S(j)$ and $\widehat{\mathcal{M}}_S(j+1)$ and $\rho_{j(e),k(e)|D(e)} = \tilde{\rho}_{j(e),k(e)|D(e)}$ for all $e \in E_i$, $i \geq j+1$, $\rho_{j(e),k(e)|D(e)} \in \boldsymbol{\theta}_S(j)$ and $\tilde{\rho}_{j(e),k(e)|D(e)} \in \widehat{\boldsymbol{\theta}}_S(j+1)$, where $\widehat{\boldsymbol{\theta}}_S(j+1)$ denotes the parameter set of model $\widehat{\mathcal{M}}_S(j+1)$.

Furthermore, we know that the likelihood of the first j trees is equal, so that the comparison of likelihoods reduces to comparing the likelihoods of tree T_{j+1} in models $\mathcal{M}_S(j)$ and $\mathcal{M}_S(j+1)$ as shown above. In contrast to the procedure based on the Vuong test, primarily we do not even have to specify the bivariate Gaussian copulas of trees $T_{j+2},...,T_{d-1}$, unless we decide to simplify the R-vine. Hence, from a heuristical point of view we have achieved "as much nestedness as possible" and obtained a very simple model comparison based on the likelihood of only one tree, namely T_{j+1}. The use of the AIC or BIC for this comparison is an ad-hoc approach which involves an increased variability regarding the model selection accuracy. Nevertheless it is often done by practitioners and at least incorporates a correction for the number of parameters used (which is not the case in direct likelihood ratio comparisons). Thus, we will thoroughly investigate the performance of this procedure in our simulation studies in Chapter 10.

Before stating the corresponding algorithm, we briefly get back to Examples 6 and 7 for an illustration of the approach.

Example 8 (Stepwise simplification using AIC/BIC.) *Suppose that we have already specified the first two trees of the five-dimensional D-vine of Example 6 and now want to determine whether simplification is feasible. As described above we specify trees T_3 and T_4 of the "small" model $\mathcal{M}_S(2)$ with Gaussian pair copulas and assume that these are also a good approximation of tree T_4 in the "full" model $\widehat{\mathcal{M}}_S(3)$ (cp. Figure 7.3).*

In contrast to the setting of the approach based on the Vuong test, we now assume that the correlation parameter of the Gaussian pair copula in the fourth tree T_4 of both

"SMALL" MODEL   "FULL" MODEL

$T_1: c_{12} \qquad c_{23} \qquad c_{34} \qquad c_{45}$ $\qquad$ $T_1: c_{12} \qquad c_{23} \qquad c_{34} \qquad c_{45}$

$T_2: \qquad c_{13|2} \qquad c_{24|3} \qquad c_{35|4}$ $\qquad$ $T_2: \qquad c_{13|2} \qquad c_{24|3} \qquad c_{35|4}$

$T_3: \qquad c^{\rho}_{14|23} \qquad c^{\rho}_{25|34}$ $\qquad$ $T_3: \qquad c_{14|23} \qquad c_{25|34}$

$T_4: \qquad c^{\rho}_{15|234}$ $\qquad$ $T_4: \qquad c^{\rho}_{15|234}$

Figure 7.3: Pair copula terms of five-dimensional D-vines simplified after the second tree T_2 ("small" model) and approximately simplified after the third tree T_3 ("full" model), respectively, where $c^{\rho}_{ij|D}$ denote Gaussian pair copulas.

models, $\rho_{15|234}$, is the same. Next, we calculate the likelihoods of T_3 in both models and determine the number of parameters used in this tree. A straightforward comparison of AIC's or BIC's then gives the desired result.

Similar to Algorithms 7 and 8 the simplification procedure using information criteria is stated in Algorithm 9. Instead of the AIC, the BIC can of course be used as well and will lead to more parsimonious models.

As stated above, the main computational advantage lies in the fact that in line 2 of Algorithm 9 we only have to specify one additional tree (in excess of tree T_{j+1}) in contrast to $[(d-1)-j] + [(d-1)-(j+1)] = 2(d-j) - 3$ additional trees in lines 2 and 3 of Algorithm 8 for the simplification procedure based on the Vuong test. Furthermore, the AIC/BIC comparison is somewhat faster than performing a Vuong test.

Algorithm 9 Simplification of R-vines using information criteria.

Input: (Pseudo-)observations of d variables.

1: **for** $j = 0, ..., d - 2$ **do**
2:      Specify model $\mathcal{M}_S(j)$ by constructing tree T_{j+1} with bivariate Gaussian copulas (the remaining trees can be ignored primarily).
3:      Specify model $\widehat{\mathcal{M}}_S(j+1)$ by constructing tree T_{j+1} with appropriate pair copulas.
4:      Compute the AIC (5.3) for models $\mathcal{M}_S(j)$ (model 1) and $\widehat{\mathcal{M}}_S(j+1)$ (model 2) based on tree T_{j+1}.
5:      **if** $AIC_1 < AIC_2$ **then**
6:          Simplify the R-vine at level $K = j$ by specifying the remaining Gaussian pair copulas for trees $T_{j+2}, ..., T_{d-1}$, i.e., exit the loop with model $\mathcal{M}_S(j)$.
7:      **end if**
8: **end for**

Output: Pairwisely simplified K level R-vine, or fully specified R-vine, if no simplification is possible.

7.4 Implementation

With the truncation and simplification methods discussed in Sections 6.2, 6.3, 7.2 and 7.3 at hand, we can now formulate a model specification procedure for omnibus use. The sequential procedure consists of three fundamental tasks in each step.

(i) *Tree construction:* Given (pseudo-)observations for tree T_1 or transformed observations (2.19) for higher order trees T_j, $j \geq 2$, we want to construct an appropriate R-vine tree according to the already mentioned paradigm that we try to capture as much dependency as possible in each tree so that hopefully most of it is explained in the first trees and higher order trees can be simplified or even truncated. Suitable construction heuristics were discussed in Chapter 3 and are available for R-, C- and D-vines.

(ii) *Copula selection:* Given a tree structure, we have to specify adequate pair copulas. In Section 5.4 we have discussed and evaluated appropriate selection methods for bivariate copulas. Note that we usually test for bivariate independence first (using the test based on Kendall's τ described in Section 4.3.1) to obtain the simplest possible model which is crucial in high dimensions. In order to obtain simple models, this independence test is also performed if we want to specify remaining trees with Gaussian copulas in one of our simplification procedures (cp. lines 2 and 3 in Algorithm 8, and lines 2 and 6 in Algorithm 9). If a t copula has more than 30 degrees of freedom, we do not consider it further as it is too close to the Gaussian. Moreover, if $\theta \leq 0.1$ or $\delta \leq 1.1$, the BB1 copula is too close to a Clayton and a Gumbel copula, respectively. Similarly, the BB7 copula is too close to a Clayton and a Joe copula if $\theta \leq 1.1$ and $\delta \leq 0.1$, respectively (cp. Sections 2.1.3 and, in particular, 5.4.2). After having selected and fitted appropriate copulas, we generate transformed observations according to (2.19).

(iii) *Truncation and simplification:* Apply one (or an adequate combination) of the procedures discussed in Sections 6.2, 6.3, 7.2 and 7.3 to determine whether the R-vine can be simplified or even truncated. We usually investigate the issue of truncation first, because it allows for the greatest possible simplification. A pre-test is also performed before the first tree is constructed to determine whether the data can possibly be modeled completely Gaussian or independent.

The parts of Algorithms 6 (line 2), 7 (line 2), 8 (lines 2 and 3) and 9 (lines 2, 3 and 6), which were stated in rather general terms, obviously refer to tasks (i) and (ii) of the list above. If the copula family is given (usually Gaussian), of course no copula selection but only fitting has to be performed. Nevertheless, we also test for bivariate independence which is a (limiting) special case of all copula families (e.g., $\rho = 0$ in a bivariate Gaussian copula) so that our model is as simple as possible.

7.4.1 Hierarchical algorithm

We will now bring together all concepts discussed so far and state a general algorithm for R-vine specification using stepwise truncation and simplification procedures. It is however
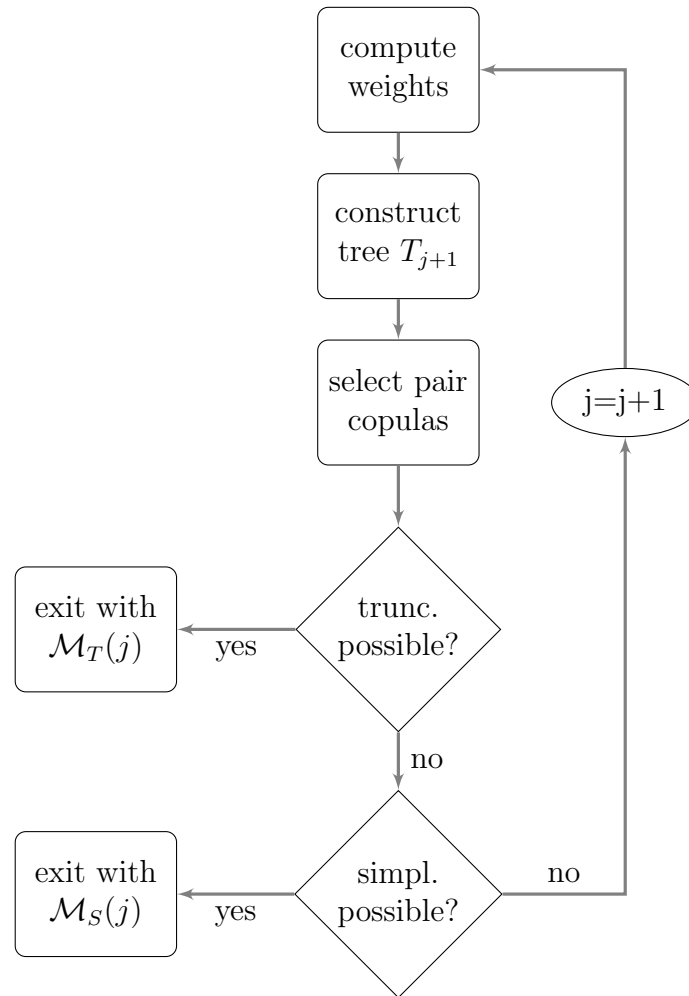
Figure 7.4: Flow chart of the hierarchical Algorithm 10.

important to note that the algorithm we propose proceeds sequentially – tree by tree – and only the likelihood of each tree separately is maximized each step in contrast to a full maximum likelihood estimation, which is not feasible especially in higher dimensions and can only be performed after having sequentially specified a complete R-vine. However, the Vuong test as well as the AIC and BIC require a full maximum likelihood estimation, i.e., we make another approximation at this point. Empirical applications showed that parameter estimates and value of the likelihood obtained by sequential estimation do not change to a great extent when performing a full maximum likelihood estimation (see Appendix B and Dißmann (2010)), so that we believe that this additional approximation is reasonable.

Algorithm 10 now combines all methods and concepts discussed so far in the way that we stepwisely test for truncation first and then for simplification if truncation is not possible after tree T_j (also cp. the corresponding flow chart in Figure 7.4). Therefore, we call this procedure *hierarchical*.

Again for $j = 0$ a pre-test is included and the transformed observations are simply the

---

Algorithm 10 Hierarchical specification of R-vines.

Input: (Pseudo-)observations of d variables, significance level α.

1: **for** $j = 0, ..., d - 2$ **do**
2:    Compute pairwise weights for the transformed observations from the previous step (using one of the methods described in Section 3.1) and use them to construct tree T_{j+1} with the appropriate method from Section 3.2.
3:    Select copulas as described in Section 5.4, estimate parameters using maximum likelihood estimation (cp. Section 4.1.2) and compute transformed observations according to (2.19).
4:    **if** Simplification based on Vuong test (cp. Section 7.2) **then**
5:       Perform a Vuong test for models $\mathcal{M}_T(j)$ (model 1) and $\mathcal{M}_T(j + 1)$ (model 2), i.e., determine test statistic ν as in (5.6).
6:       **if** $\nu > -\Phi^{-1}\left(1 - \frac{\alpha}{2}\right)$ **then**
7:          Truncate the R-vine at level $K = j$, i.e., exit the loop with model $\mathcal{M}_T(j)$.
8:       **else**
9:          Specify model $\mathcal{M}_S(j)$: sequentially construct higher order trees $T_{j+1}, ..., T_{d-1}$ with bivariate Gaussian copulas.
10:          Specify model $\mathcal{M}_S(j+1)$: sequentially construct higher order trees $T_{j+2}, ..., T_{d-1}$ with bivariate Gaussian copulas.
11:          Perform a Vuong test for models $\mathcal{M}_S(j)$ (model 1) and $\mathcal{M}_S(j + 1)$ (model 2), i.e., determine test statistic ν as in (5.6).
12:          **if** $\nu > -\Phi^{-1}\left(1 - \frac{\alpha}{2}\right)$ **then**
13:             Simplify the R-vine at level $K = j$, i.e., exit the loop with model $\mathcal{M}_S(j)$.
14:          **end if**
15:       **end if**
16:    **else if** Simplification based on AIC (cp. Section 7.3) **then**
17:       Compute the AIC (5.3) for tree T_{j+1} , i.e., for model $\mathcal{M}_T(j + 1)$.
18:       **if** $0 < AIC$ **then**
19:          Truncate the R-vine at level $K = j$, i.e., exit the loop with model $\mathcal{M}_T(j)$.
20:       **else**
21:          Specify (reduced) model $\mathcal{M}_S(j)$: construct tree T_{j+1} with bivariate Gaussian copulas (the remaining trees can be ignored primarily).
22:          Compute the AIC (5.3) for model $\mathcal{M}_S(j)$ (model 1) based on tree T_{j+1}. (AIC of $\widehat{\mathcal{M}}_S(j + 1)$ (model 2) based on tree T_{j+1} is the same as the AIC of $\mathcal{M}_T(j+1)$.)
23:          **if** $AIC_1 < AIC_2$ **then**
24:             Specify (full) model $\mathcal{M}_S(j)$: sequentially construct higher order trees $T_{j+2}, ..., T_{d-1}$ with bivariate Gaussian copulas.
25:             Simplify the R-vine at level $K = j$, i.e., exit the loop with model $\mathcal{M}_S(j)$.
26:          **end if**
27:       **end if**
28:    **end if**
29: **end for**

Output: Pairwisely simplified K level R-vine, pairwisely truncated R-vine at level K, or fully specified R-vine if neither simplification nor truncation are possible (or required).

---

original observations. When using the Vuong test, a correction factor can be chosen as described in Section 5.3.1 (or it can be performed without correction). And for reasons of readability Algorithm 10 is formulated using the AIC, but the BIC can of course be used instead.

In lines 17 to 19 we use the fact that the models $\mathcal{M}_T(j)$ and $\mathcal{M}_T(j + 1)$ are nested and therefore the AIC comparison reduces to the comparison of tree T_{j+1}, where the AIC of $\mathcal{M}_T(j)$ is obviously 0 for this tree (cp. (7.3)). Also note that the Vuong test in lines 5 to 7 can be performed using only the information of tree T_{j+1}, since the Vuong test is likelihood ratio-based and hence the likelihoods of the first trees $T_1, ..., T_j$ cancel each other out as discussed above. This facilitates the test significantly.

Chapter 8

Joint simplification

C-vines constitute a special structure when considering simplification. Besides pairwise simplification, they allow for another very appealing simplification approach based on multivariate copulas. As we will see, unfortunately this does not work for D-vines, and only for some special R-vines. Detailed theoretical results are stated in the last part of this chapter. However, this method is rather impractical anyway as it requires computationally quite demanding multivariate copula goodness-of-fit tests (cp. Section 4.2).

8.1   Joint simplification of canonical vines

In Section 7.1, we have seen how to write the parametrized density of a simplified K level R-vine. If we consider the special case of a C-vine (cp. Section 2.4), this density is given by

$$c_S^K(\boldsymbol{u}|\boldsymbol{\theta}_S(K)) = \left[\prod_{i=1}^{K}\prod_{j=i+1}^{d} c_{i,j|1:(i-1)}\right] \times \left[\prod_{i=K+1}^{d-1}\prod_{j=i+1}^{d} c_{i,j|1:(i-1)}^{\rho}\right], \tag{8.1}$$

where $\boldsymbol{u} \in [0,1]^d$, $c_{i,j|1:(i-1)}^{\rho}$ denote bivariate Gaussian copulas and $\boldsymbol{\theta}_S(K)$ is the parameter set of the density.

The special structure of the conditioning set of a C-vine now allows to rewrite the second part of the above equation. Valdesogo (2009) showed in Theorem 2.3.1 that in a C-vine all pair copulas with a conditioning set larger than K (i.e., the second part of Equation (8.1)) can be modeled jointly involving a $(d-K)$-dimensional copula. We will establish this result in more detail in Section 8.2 and can then rewrite Equation (8.1) to

$$c_J^K(\boldsymbol{u}|\boldsymbol{\theta}_J(K)) = \left[\prod_{i=1}^{K}\prod_{j=i+1}^{d} c_{i,j|1:(i-1)}\right] \times c_{(K+1):d|1:K}^{\rho}, \tag{8.2}$$

where $c_{(K+1):d|1:K}^{\rho}$ denotes a $(d-K)$-dimensional Gaussian copula (cp. (2.7)) and where we write c_J^K instead of c_S^K to highlight the different simplification method. The parameter set $\boldsymbol{\theta}_J(K)$ is defined similar to (7.2) as

$$\boldsymbol{\theta}_J(K) = \{\boldsymbol{\theta}_{i,j|1:(i-1)} : \ j = i+1, ..., d, \ i = 1, ..., K\}$$
$$\cup \{\rho_{ij|1:K} : \ i, j = K+1, ..., d, \ i \neq j\},$$

where $\boldsymbol{\theta}_{i,j|1:(i-1)}$ are the parameters of the pair copulas $c_{i,j|1:(i-1)}$, while $\rho_{ij|1:K}$ denote the entries of the correlation matrix of the multivariate Gaussian copula $c^{\rho}_{(K+1):d|1:K}$.

Similarly to $\mathcal{M}_S(K)$ we also define $\mathcal{M}_J(K)$ as the statistical model corresponding to the density in Equation (8.2) and we speak of a *jointly simplified K level C-vine* – in contrast to a stepwisely simplified C-vine corresponding to (8.1). In the case of a multivariate independence copula instead ($\rho_{ij|1:K} \equiv 0 \ \forall i, j = K+1, ..., d, \ i \neq j$), we speak of a *jointly truncated C-vine at level K* and denote the density by $c^K_T(\boldsymbol{u}|\boldsymbol{\theta}_S(K))$ as before. Following similar arguments as in Section 7.1, we will not consider other multivariate copulas such as the multivariate t copula or multivariate Archimedean copulas. While the first is more difficult to estimate and hence to validate in a goodness-of-fit test, multivariate Archimedean copulas are too restrictive in their structure (the multivariate t copula is also restrictive in the sense that it has only one joint degrees of freedom parameter). The multivariate Gaussian copula however is intuitively appealing, easy to estimate and well studied, but we have to keep in mind its drawbacks such as symmetry and zero tail dependence for all variables pairs.

The corresponding hierarchical model building procedure is then straightforward to obtain and described in Algorithm 11 using one of the multivariate independence tests described in Sections 4.3.2 and 4.3.3 as well as one of the two multivariate copula goodness-of-fit tests of Section 4.2 with significance level α. The procedure combines truncation and simplification by always considering truncation first to obtain the simplest possible model in each step, i.e., it is hierarchical. Of course, Algorithm 11 can also be split into two procedures: one which considers truncation only and the other one concentrating on simplification as we did in Chapters 6 and 7.

Line 11 in Algorithm 11 is stated in rather general terms and refers to the tasks (i) and (ii) in Section 7.4 (cp. Algoritm 10).

The following example explains the procedures of lines 2 and 6 in more detail.

Example 9 (Joint simplification of C-vines.) *We consider a five-dimensional C-vine similar to the D-vine discussed in Examples 4 to 8 (also cp. Figure 2.4). This means we have to specify four trees with ten pair copulas in total. Again assume that we have already appropriately specified the first two trees T_1 and T_2 and now want to investigate whether we can truncate or simplify at level $K = 2$, i.e., if one of the two models illustrated in Figure 8.1 is appropriate.*

As we start with truncation first, we take the transformed observations from tree T_2 and test whether they are jointly independent using one of the tests described in Sections 4.3.2 and 4.3.3. If the p-value of the chosen test is larger than the pre-specfied significance

TRUNCATED MODEL				SIMPLIFIED MODEL			
$T_1:$ c_{12}	c_{13}	c_{14}	c_{15}	$T_1:$ c_{12}	c_{13}	c_{14}	c_{15}
$T_2:$	$c_{23|1}$	$c_{24|1}$	$c_{25|1}$	$T_2:$	$c_{23|1}$	$c_{24|1}$	$c_{25|1}$
$T_{3/4}:$		$\pi_{345|12}$		$T_{3/4}:$		$c^{\rho}_{345|12}$	

Figure 8.1: Pair and multivariate copula terms of five-dimensional C-vines jointly truncated and jointly simplified after the second tree T_2, respectively, where $\pi_{345|12}$ denotes a three-dimensional independence copula and $c^{\rho}_{345|12}$ a three-dimensional Gaussian copula.

Algorithm 11 Joint simplification of C-vines.

Input: (Pseudo-)observations of d variables, significance level α.

 1: **for** $j = 0, ..., d - 2$ **do**
 2:     Perform a multivariate independence test for model $\mathcal{M}_T(j)$, i.e., for the transformed observations from tree T_j, e.g., using the test statistics defined in (4.16) or in (4.18).
 3:     **if** p-value $> \alpha$ **then**
 4:         Truncate the C-vine at level $K = j$, i.e., exit the loop with model $\mathcal{M}_T(j)$.
 5:     **else**
 6:         Perform a multivariate copula goodness-of-fit test for model $\mathcal{M}_J(j)$, i.e., test if the transformed observations from tree T_j can be appropriately modeled with a $(d-j)$-dimensional Gaussian copula, e.g., using the test statistics defined in (4.6) or in (4.11).
 7:         **if** p-value $> \alpha$ **then**
 8:             Simplify the C-vine at level $K = j$, i.e., exit the loop with model $\mathcal{M}_J(j)$.
 9:         **end if**
10:     **end if**
11:     Construct tree T_{j+1} and specify appropriate pair copulas (refer to Section 7.4 for more details on this issue).
12: **end for**

Output: Jointly simplified K level C-vine, jointly truncated C-vine at level K, or fully specified C-vine, if neither simplification nor truncation are possible.

---

level α, then the null hypothesis that the transformed observations are jointly independent cannot be rejected and hence we truncate the C-vine at level $K = 2$.

If no truncation has been detected (or we are not interested in truncation), we examine if at least simplification is possible. Therefore we take the transformed observations from tree T_2 and test whether they follow a multivariate (here: trivariate) Gaussian copula $c_{345|12}^{\rho}$ using one of the tests described in Section 4.2. Again, if the p-value is larger than α, we simplify the C-vine at level $K = 2$, otherwise we continue with the next tree.

In contrast to the procedures discussed in Sections 6.2, 6.3, 7.2 and 7.3, this procedure is an exact procedure without any approximations or further assumptions. In each step j it also does not require the specification of tree T_{j+1} (or any other higher order trees) in advance. For $j = 0$ a pre-test whether the variables are jointly independent or follow a d-dimensional Gaussian copula is performed.

The use of a multivariate goodness-of-fit test in line 6 of Algorithm 11 is not trivial though: on the one hand, even if the multiplier approach described in Section 4.2.2 is used, multivariate goodness-of-fit testing based on the empirical copula process is computationally extremely demanding in high dimensions. On the other hand, the test based on Rosenblatt's transformation (Section 4.2.3) has a lower power in most cases except for testing against heavy tails. Its performance however increases with increasing sample size and increasing dependency (Berg 2009). Moreover, multivariate goodness-of-fit testing has not been extensively studied in the literature so far. As noted earlier, Genest et al. (2009) study only bivariate copulas, while Berg (2009) and Kojadinovic and Yan (2010a) consider dimensions $d = 4, 8$ and $d = 3, 4$, respectively. The multivariate independence

test based on the empirical copula process proposed by Genest and Rémillard (2004) was also considered only for dimensions up to $d = 5$. Such dimensions are still rather small for extensive practical applications (cp. the 52-dimensional application in Section 11.3).

A possible cure to these issues might be to use one (or an appropriate combination) of the procedures discussed in Sections 7.2 and 7.3 as an approximation to the exact joint simplification of C-vines and, if the C-vine is simplified, reconstruct the multivariate Gaussian copula from the Gaussian pair copulas. (Regarding truncation, it is clear that stepwisely truncated C-vines are equivalent to jointly truncated C-vines.) We will discuss this in the following section and investigate it in detail in the simulation studies in Chapter 10. There, we will also examine whether the goodness-of-fit test based on the Rosenblatt transformation can generally be used for our purposes, since it is computationally more efficient than the test based on the empirical copula process.

8.1.1 Computing the correlation matrix from partial correlations

If the chosen simplification method (either based on the Vuong test or on AIC/BIC) determines a significant simplification of the higher order trees $T_{K+1}, ..., T_{d-1}$ with bivariate Gaussian copulas in a C-vine as in (8.1), we want to compute the correlation matrix of the multivariate Gaussian copula which corresponds to this simplification as in (8.2).

In the case of normally distributed random variables, we know that the following two statements are equivalent (see, e.g., Lanzendörfer (2009, p. 20) using a C-vine instead of a D-vine):

(i) $X_1, ..., X_d \sim N_d(0, R)$ with correlation matrix $R \in [-1, 1]^d$

(ii) $X_i \sim N(0, 1) \; \forall i = 1, ..., d$, with joint density

$$f(\boldsymbol{x}) = \prod_{k=1}^{d} \varphi(x_k) \prod_{j=1}^{d-1} \prod_{i=1}^{d-j} c_{j,j+i|1:(j-1)}^{\rho}(u_{j|1:(j-1)}, u_{j+i|1:(j-1)}|\rho_{j,j+i|1:(j-1)}), \qquad (8.3)$$

where $\varphi(\cdot)$ denotes the density of the standard normal distribution, $u_{j|1:(j-1)} := F(x_j|x_1, ..., x_{j-1})$, $u_{j+i|1:(j-1)} := F(x_{j+i}|x_1, ..., x_{j-1})$, and $c_{j,j+i|1:(j-1)}^{\rho}(\cdot, \cdot|\rho)$ is the density of a bivariate Gaussian copula with parameter $\rho = \rho_{j,j+i|1:(j-1)}$ which denotes the partial correlation between X_j and X_{j+i} given $X_1, ..., X_{j-1}$. Note that (8.3) is the density of a C-vine (cp. (2.28)) with standard normal margins.

With statement (i) it follows that

$$f(\boldsymbol{x}) = (2\pi)^{-\frac{d}{2}}|R|^{-\frac{1}{2}} \exp(-\frac{1}{2}\boldsymbol{x}'R^{-1}\boldsymbol{x})$$

$$= \left[\prod_{k=1}^{d} \varphi(x_k)\right]|R|^{-\frac{1}{2}} \exp(\frac{1}{2}\boldsymbol{x}'(I_d - R^{-1})\boldsymbol{x}), \qquad (8.4)$$

where $\boldsymbol{x} = (x_1, ..., x_d)'$ and the second part of Equation (8.4) is the density of a multivariate Gaussian copula (cp. (2.8)). With Equation (8.3) from statement 2 above, this yields that

for normally distributed random variables we can write the density of a multivariate Gaussian copula as a C-vine density with bivariate Gaussian copulas:

$$|R|^{-\frac{1}{2}} \exp(\frac{1}{2}\boldsymbol{x}'(I_d - R^{-1})\boldsymbol{x}) = \prod_{j=1}^{d-1}\prod_{i=1}^{d-j} c_{j,j+i|1:(j-1)}^{\rho}(u_{j|1:(j-1)}, u_{j+i|1:(j-1)}|\rho_{j,j+i|1:(j-1)}).$$

However, usually we cannot assume that our data comes from a normal distribution. Therefore we assume that we already have copula data, i.e., data which comes from uniformly on [0,1] distributed random variables $U_1, ..., U_d$ (otherwise we can transform our data by the empirical distribution function to obtain pseudo-observations as defined in (4.4)). Then $Z_i = \Phi^{-1}(U_i)$, $i = 1, ..., d$, are known to be normally distributed. Moreover, we know that copulas are invariant under strictly increasing transformations of random variables (cp. Theorem 2.6). Thus the joint distribution of $Z_1, ..., Z_d$ is determined by the same multivariate copula as the joint distribution of $U_1, ..., U_d$ and vice versa.

We now exploit this fact to determine the correlation matrix of the multivariate Gaussian copula which we use to simplify our C-vine on observed data coming from the random variables $U_1, ..., U_d$. Assume in the following that our C-vine has been determined to be simplified at level K and hence to be adequately modeled with bivariate Gaussian copulas for trees $T_{K+1}, ..., T_{d-1}$. Partial correlations have been estimated for all pair copulas. Then $V_{K+1} := F(U_{K+1}|U_1, ..., U_K), ..., V_d := F(U_d|U_1, ..., U_K)$ are again uniformly distributed on [0,1], where $F(\cdot|U_1, ..., U_K)$ is determined by the pair copula construction in $T_1, ..., T_K$ (cp. (2.19)). Using the fact above and that $Z_i = \Phi^{-1}(V_i)$, $i = K + 1, ..., d$, are normally distributed, we can conclude that the joint distribution of $Z_{K+1}, ..., Z_d$ is described by the bivariate Gaussian pair copula construction given by the trees $T_{K+1}, ..., T_{d-1}$. As shown earlier this is equivalent to a multivariate Gaussian copula, and using the fact above again, it follows that the joint distribution of $V_{K+1}, ..., V_d$ is determined by the same multivariate Gaussian copula.

Note that this one-to-one correspondence only works because $V_{K+1}, ..., V_d$ are uniquely determined by the first K trees. In the case of a general R-vine this is not the case (except for $K = 0$, see Lanzendörfer (2009) for the D-vine case) because we cannot choose such transformed variables so that we obtain a valid density. E.g., in D-vines, there are $2d - 4$ different transformed variables in the second tree instead of $d - 1$ when we consider a C-vine! The problem is that there is no common conditioning set in each tree (cp. Section 8.1.2). There is however a special case where R-vines can be jointly simplified as well (see below).

For C-vines, now the question remains how we can compute the correlation matrix of this multivariate Gaussian copula, when we know the partial correlations of the bivariate Gaussian copulas. This can be done using the following recursive formula (see, e.g., p. 290 in Yule and Kendall (1965)):

$$\rho_{ij|D} = \frac{\rho_{ij|D_{-v}} - \rho_{iv|D_{-v}}\ \rho_{jv|D_{-v}}}{\sqrt{1 - \rho_{iv|D_{-v}}^2}\ \sqrt{1 - \rho_{iv|D_{-v}}^2}}, \tag{8.5}$$

where D is a set of indices with $v \in D$, $D_{-v} := D \setminus \{v\}$ and, if $D = \varnothing$, $\rho_{ij|\varnothing} := \rho_{ij}$. If $|D| = k$, this equation expresses the k-th order partial correlation of two random variables

---

Algorithm 12 Computation of unconditional from conditional correlations in a C-vine.

Input: Dimension $d > 2$, partial correlations $\rho_{j,j+i|1:(j-1)}$ adapted to a C-vine.
 1: Create empty matrix $R = (R_{i,j})_{i,j=1,...,d}$.
 2: $diag(R) := (1,...,1)'$
 3: **for** $i = 2$ to d **do**
 4:     $R_{1,i} = \rho_{1,i}$
 5: **end for**
 6: **for** $k = d - 2$ to 1 **do**
 7:     **for** $j = k + 1$ to $d - 1$ **do**
 8:       **for** $i = j + 1$ to d **do**
 9:            $\rho_{ij|1:(k-1)} = \rho_{ik|1:(k-1)}\rho_{jk|1:(k-1)} + \rho_{ij|1:k}\sqrt{1 - \rho_{ik|1:(k-1)}^2}\sqrt{1 - \rho_{jk|1:(k-1)}^2}$
10:         **if** $k = 1$ **then**
11:             $R_{i,j} = \rho_{ij|1:(k-1)} = \rho_{ij}$
12:         **end if**
13:       **end for**
14:     **end for**
15: **end for**
Output: Correlation matrix R.

---

as a function of three $(k-1)$-th order partial correlations. Solving Equation (8.5) for $\rho_{ij|D_{-v}}$ gives

$$\rho_{ij|D_{-v}} = \rho_{iv|D_{-v}}\ \rho_{jv|D_{-v}} + \rho_{ij|D}\sqrt{1 - \rho_{iv|D_{-v}}^2}\ \sqrt{1 - \rho_{iv|D_{-v}}^2},$$

which, together with the fact that conditional and partial correlations are equal for normally distributed random variables (Yule and Kendall 1965), yields Algorithm 12 to compute the unconditional correlations from partial correlations in a C-vine. The algorithm is stated without reference to the simplification level K. However, it is easy to see that we can apply Algorithm 12 directly to $V_{K+1},...,V_d$ which are taken from the stepwisely simplified K level C-vine.

The procedure of Algorithm 12 can be summarized as follows: first, we store all available unconditional correlations. Then we proceed by reducing the order of the conditional correlations by 1 in each step of the first loop. The second and third loops run over all $(k-1)$-th order conditional correlations. When conditional correlations of order 0, i.e., unconditional correlations, are obtained, they are stored.

When implementing Algorithm 12, we can start with the matrix R having the conditional correlations as entries. Then line 10 simplifies to

$$R_{i,j} = \rho_{ik|1:(k-1)}\rho_{jk|1:(k-1)} + R_{i,j}\sqrt{1 - \rho_{ik|1:(k-1)}^2}\sqrt{1 - \rho_{jk|1:(k-1)}^2}.$$

Note that the output of the algorithm is a lower triangular matrix. As correlations matrices are, of course, symmetric, the upper triangle of the matrix is readily given.

We also adapted Algorithm 12 to the structure of RVM's (cp. Section 2.4.1), since we consider C-vines as a special case of R-vines. Because an RVM is not unique for a given C-vine (Dißmann 2010), the algorithmic formulation is more tricky but similar to

$$c_{12}(F(x_1), F(x_2)) \qquad c_{23}(F(x_2), F(x_3)) \qquad c_{34}(F(x_3), F(x_4)) \qquad c_{45}(F(x_4), F(x_5))$$

$$c_{13|2}(F(x_1|x_2), F(x_3|x_2)) \quad c_{24|3}(F(x_2|x_3), F(x_4|x_3)) \quad c_{35|4}(F(x_3|x_4), F(x_5|x_4))$$

$$c_{14|23}(F(x_1|x_2, x_3), F(x_4|x_2, x_3)) \quad c_{25|34}(F(x_2|x_3, x_4), F(x_5|x_3, x_4))$$

$$c_{15|234}(F(x_1|x_2, x_3, x_4), F(x_5|x_2, x_3, x_4))$$

Figure 8.2: Pair copulas with arguments of a five-dimensional D-vine.

Algorithm 12. The main difference is that we have to attach labels to each conditional correlation so that we can identify the necessary lower order conditional correlations in line 9 of Algorithm 12.

To sum it up, if our chosen specification procedure now determines that we can *stepwise* simplify our C-vine with Gaussian pair copulas for all remaining trees, it is equivalent to state that the C-vine is *jointly* simplified by a multivariate Gaussian copula and the correlation matrix of this copula can then be computed by Algorithm 12 using the estimated conditional correlation parameters of the Gaussian pair copulas.

8.1.2   Joint simplification of regular vines

Despite its obvious computational drawbacks, the idea of simplifying an R-vine by a multivariate copula is intuitively very appealing. However, we will consider a small example to see why this is not feasible in general. In the example we consider a D-vine which can be regarded as the exact contrary to C-vines, while general R-vines are models between these boundary cases.

Example 10 (Infeasibility of joint simplification of D-vines.) *Once again we consider the five-dimensional D-vine of Examples 4 to 8. For reasons of readability, we have always omitted the arguments of the copulas so far. As these are fundamental in the understanding of joint simplification, we state the ten copulas with their respective arguments in Figure 8.2.*

Now suppose that we want to jointly simplify the D-vine at level $K = 1$, i.e., we want to specify trees T_2, T_3 and T_4 with a multivariate Gaussian copula. The input arguments of tree T_2, and therefore the input arguments of a possible multivariate copula, are $F(x_1|x_2)$, $F(x_3|x_2)$, $F(x_2|x_3)$, $F(x_4|x_3)$, $F(x_3|x_4)$ and $F(x_5|x_4)$. It is not obvious, and in fact not feasible, to combine these transformed observations so that we get a valid probability density similar to (8.2) (apart from multivariate independence). In particular, we have crosswise relationships such as $F(x_2|x_3)$ and $F(x_3|x_2)$ which complicate the situation. The same problem occurs for the input arguments of tree T_3, while it does not make sense to simplify tree T_4. As will be shown in Section 8.2, there are no such crossing overs in C-vines.

Of course, for general R-vines the above is also true. However, there is one very special case where joint simplification of higher order trees is possible: if an R-vine reduces to a C-vine structure in higher order trees (cp. Figure 8.3 for the simplest possible example which
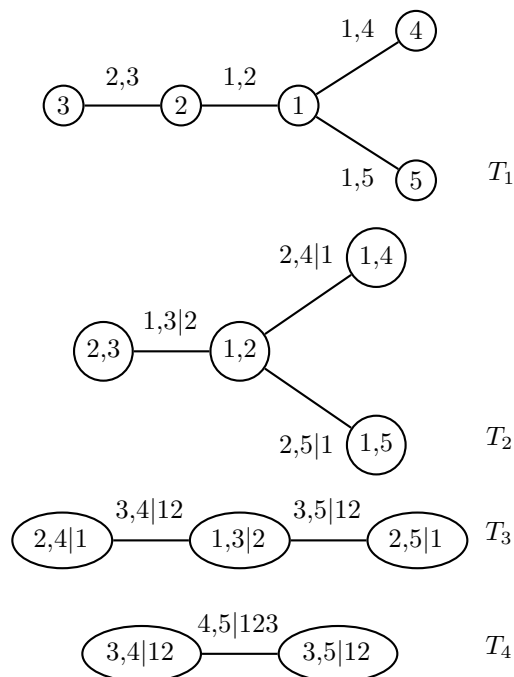
Figure 8.3: Example of an R-vine which can be jointly simplified at level $K = 2$ with a trivariate copula.

is the five-dimensional R-vine considered in Example 2), then this sub-C-vine can of course be simplified with a multivariate copula, since the arguments have the same conditioning set. This result will be established as a general theorem in Section 8.2. However, due to the computational issues of joint simplification and since this is a rather special case, we will not investigate this possibility further in the following chapters. But if one is interested in jointly simplifying an R-vine after some higher order tree, one should look for first trees that are "star-shaped" so that they reduce to C-vine structures in higher order trees.

8.2   Theoretical results

8.2.1   Canonical vines

Jointly simplified C-vines were first applied in Heinen and Valdesogo (2009) and constructed more generally in Chapter 2 of Valdesogo (2009). In this section, we present the results of the latter author which we already used in Section 8.1. The work of Heinen and Valdesogo (2009) will be reviewed and critically evaluated in Chapter 11.

To illustrate the idea of Valdesogo (2009), we begin with a five-dimensional example (cp. Example 9).

Example 11 (Joint simplification of C-vines.) *Let* $\mathbf{X} = (X_1, ..., X_5)$ *be a five-dimensional random vector with joint density function* $f(x_1, ..., x_5)$*. Using the conditional*

$$\begin{array}{c|cccc} T_1 : & c_{12} & c_{13} & c_{14} & c_{15} \\ \hline T_{2/3/4} : & & c_{2345|1} & & \end{array}$$

Figure 8.4: Pair and multivariate copula terms of a five-dimensional C-vine jointly simplified after the first tree T_1.

version of Sklar's Theorem 2.4 we know that

$$f(x_2, ..., x_5|x_1) = \left[\prod_{k=2}^{5} f(x_k|x_1)\right] c_{2:5|1}(F(x_2|x_1), ..., F(x_5|x_1))$$

$$\Leftrightarrow \quad f(x_1, ..., x_5) = f_1(x_1) \left[\prod_{k=2}^{5} f(x_k|x_1)\right] c_{2:5|1}(F(x_2|x_1), ..., F(x_5|x_1)), \quad (8.6)$$

where $c_{2:5|1}$ is a four-dimensional copula. As in the pair copula construction in Section 2.1.4, we also know that

$$f(x_j|x_1) = f_j(x_j)c_{1j}(u_1, u_j) \quad (8.7)$$

for all $j = 2, ..., 5$ and pair copulas c_{1j}, where $u_k := F(x_k) \ \forall k = 1, ..., 5$. Substituting (8.7) into (8.6), we obtain

$$f(x_1, ..., x_5) = \left[\prod_{k=1}^{5} f_k(x_k)\right] \left[\prod_{j=2}^{5} c_{1j}(u_1, u_j)\right] c_{2:5|1}(F(x_2|x_1), ..., F(x_5|x_1)). \quad (8.8)$$

Of course, we also know according to Sklar (1959) that

$$f(x_1, ..., x_5) = \left[\prod_{k=1}^{5} f_k(x_k)\right] c_{1:5}(u_1, ..., u_5),$$

where $c_{1:5}$ is a five-dimensional copula, and hence the copula density corresponding to Equation (8.8) is

$$c_{1:5}(u_1, ..., u_5) = \left[\prod_{j=2}^{5} c_{1j}(u_1, u_j)\right] c_{2:5|1}(F(x_2|x_1), ..., F(x_5|x_1)),$$

i.e., we expressed a five-dimensional copula as the product of the pair copulas of a first C-vine tree with root node 1 and a four-dimensional copula. Using the notation of copula terms as in previous examples, this corresponds to the jointly simplified $K = 1$ level C-vine shown in Figure 8.4.

Figure 8.5 illustrates this C-vine. The dashed lines in graph $\widetilde{T}_2$ indicate that this is not a tree anymore, since trees are a convenient method to model bivariate dependence, but they are not appropriate to display multivariate dependency relationships. Joint simplification can be more appropriately displayed by complete graphs (here: complete graph on four nodes; cp. Section 2.3), since the multivariate copula corresponding to the simplification directly links all (conditioned) variables. The indices of this copula are denoted in the middle of graph $\widetilde{T}_2$.
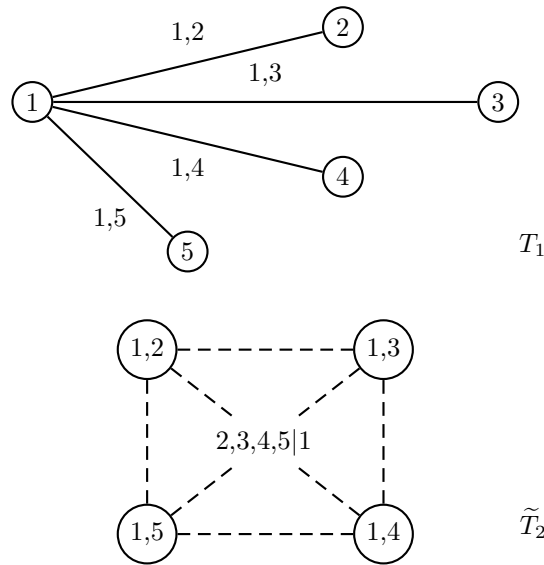
Figure 8.5: Jointly simplified C-vine at level $K = 1$.

From the derivation, it is clear why this joint simplification does not work for R-vines in general. In order to be able to jointly simplify an R-vine, we also need a common conditioning set (here: variable X_1) which is only given for C-vines and special cases of R-vines (cp. Figure 8.3 and Section 8.2.2), but never for D-vines (except in tree T_{d-1}, but there is nothing left to be simplified as there is only one pair copula to be specified).

Valdesogo (2009) generalizes this result and shows how to decompose d-dimensional copulas into the product of the pair copulas of the first K C-vine trees and a $(d-K)$-dimensional copula.

Theorem 8.1 (Joint simplification of C-vines.) *Let* $\mathbf{X} = (X_1, ..., X_d)$ *be a d-dimensional random vector with joint copula density function $c_{1:d}(u_1, ..., u_d)$, where $u_i := F(x_i)$ $\forall i = 1, ..., d$. Further let $K \in \{0, ..., n-2\}$, then the copula density can be written as*

$$c_{1:d}(u_1, ..., u_d) = \left[\prod_{i=1}^{K} \prod_{j=i+1}^{d} c_{i,j|1:(i-1)}(F(x_i|x_1, ..., x_{i-1}), F(x_j|x_1, ..., x_{i-1})) \right]$$
$$\times c_{(K+1):d|1:K}(F(x_{K+1}|x_1, ..., x_K), ..., F(x_d|x_1, ..., x_K)).$$

The proof can be found in Valdesogo (2009). It uses the conditional Sklar's Theorem in higher dimension and crucially relies on the C-vine structure with common conditioning sets. Any other permutation of the variables is obviously also possible. The order $1...d$ is chosen only for illustrative reasons.

8.2.2 Regular vines

As we have seen in Section 8.1.2, under certain conditions it is possible to simplify R-vines as well. We state these conditions in the following theorem.

Theorem 8.2 (Joint simplification of R-vines.) *Let* $\mathbf{X} = (X_1, ..., X_d)$ *be a d-dimensional random vector with joint copula density function* $c_{1:d}(u_1, ..., u_d)$*, where* $u_i := F(x_i)$ $\forall i = 1, ..., d$*. Further let* $K \in \{0, ..., n-2\}$ *and* $D \subset \{1, ..., d\}$ *with* $|D| = K$*. If*

$$D(e) \equiv D \ \forall e \in E_{K+1},$$

then the copula density can be written as

$$c_{1:d}(u_1, ..., u_d) = \left[\prod_{i=1}^{K} \prod_{e \in E_i} c_{j(e),k(e)|D(e)}(F(x_{j(e)}|\boldsymbol{x}_{D(e)}), F(x_{k(e)}|\boldsymbol{x}_{D(e)})) \right]$$
$$\times c_{D^c|D}(F(x_{v_{K+1}}|\boldsymbol{x}_D), ..., F(x_{v_d}|\boldsymbol{x}_D)),$$

where $D^c = \{1, ..., d\} \setminus D$ *and* $D^c := \{v_{K+1}, ..., v_d\}$*.*

Theorem 8.1 is then a special case of Theorem 8.2 with additional restrictions regarding the conditioning set, namely

$$\forall i = 1, ..., K : \ D(e) = \{1, ..., i-1\} \ \forall e \in E_i.$$

The proof of Theorem 8.2 follows along the same lines as the proof of Theorem 8.1 (Valdesogo 2009) and is stated in the following. Subsequently an informal explanation and an example are presented.

Proof of Theorem 8.2: Set $D := \{v_1, ..., v_K\}$ similar to the definition of D^c. Using the definition of conditional densities we know that

$$f(x_1, ..., x_d) = f(\boldsymbol{x}_D)f(\boldsymbol{x}_{D^c}|\boldsymbol{x}_D). \tag{8.9}$$

By assumption $D(e) \equiv D \ \forall e \in E_{K+1}$. According to the R-vine construction principles (cp. Section 2.4) it must therefore hold that

$$\forall i = 1, ..., K : \ D(e) \subset D \ \forall e \in E_i. \tag{8.10}$$

Then the first part of Equation (8.9) can be decomposed using the R-vine density as given in (2.27):

$$f(\boldsymbol{x}_D) = \left[\prod_{j=1}^{K} f_{v_j}(x_{v_j}) \right] \left[\prod_{i=1}^{K-1} \prod_{e \in E_i^K} c_{j(e),k(e)|D(e)}(F(x_{j(e)}|\boldsymbol{x}_{D(e)}), F(x_{k(e)}|\boldsymbol{x}_{D(e)})) \right], \tag{8.11}$$

where E_i^K denotes the respective edge sets of this R-vine on K variables. Applying Sklar's Theorem 2.4 for conditional distributions we get for the second part of Equation (8.9):

$$f(\boldsymbol{x}_{D^c}|\boldsymbol{x}_D) = \left[\prod_{j=K+1}^{d} f(x_{v_j}|\boldsymbol{x}_D) \right] c_{D^c|D}(F(x_{v_{K+1}}|\boldsymbol{x}_D), ..., F(x_{v_d}|\boldsymbol{x}_D)). \tag{8.12}$$

Now let $v_j \in D^c$. We claim that $\forall i = 1, ..., K \; \exists w_i^{(j)} : \; v_j, w_i^{(j)} | D_{ij} \in E_i$, where D_{ij} is the corresponding conditioning set. To prove this statement, it suffices to show that $\exists w_i^{(1)} : v_j, w_1^{(j)} \in E_1$. Then the existence of $w_i^{(j)}$, $i = 2, ..., K$, follows by R-vine construction principles.

To show this, we assume that the statement is not true. Since trees are connected by definition, $\exists \tilde{w} \in \{1, ..., d\} \setminus D = D^c : \; e_j := v_j, \tilde{w} \in E_1$. Using one more time that trees are connected we further know that

$$\left(\exists a \in \{1, ..., d\} : e_a := a, v_j \in E_1\right) \vee \left(\exists b \in \{1, ..., d\} : e_b := b, \tilde{w} \in E_1\right).$$

We consider both cases separately.

(i) $\exists a \in \{1, ..., d\} : e_a := a, v_j \in E_1$: By R-vine construction principles, this yields that $e_j, e_a \in V_{i+1}$ are joined by the edge $a, \tilde{w} | D_a \in E_2$, where

$$v_j = D_a := \{a, v_j\} \cap \{v_j, \tilde{w}\}.$$

But $v_j \in D^c$ by assumption and therefore $D_a \not\subseteq D$ in contradiction to (8.10).

(ii) $\exists b \in \{1, ..., d\} : e_b := b, \tilde{w} \in E_1$: Similarly to (i), we get $\tilde{w} = D_b := \{b, \tilde{w}\} \cap \{v_j, \tilde{w}\}$, and hence $D_b \not\subseteq D$ in contradiction to (8.10).

This proves the above claim, which basically states that in each tree, each variable, which is not in the common conditioning set D, is linked by a pair copula to a variable of this set D. We can exploit this fact by considering the following R-vine decomposition for $j \in \{K+1, ..., d\}$:

$$f(\boldsymbol{x}_D, x_{v_j}) = \left[\prod_{k=1}^{K} f_{v_k}(x_{v_k})\right] f_{v_j}(x_{v_j}) \left[\prod_{i=1}^{K} \prod_{e \in E_i^{(j,K)}} c_{j(e),k(e)|D(e)}(F(x_{j(e)}|\boldsymbol{x}_{D(e)}), F(x_{k(e)}|\boldsymbol{x}_{D(e)}))\right]$$

$$= \left[\prod_{k=1}^{K} f_{v_k}(x_{v_k})\right] \left[\prod_{i=1}^{K-1} \prod_{e \in E_i^{K}} c_{j(e),k(e)|D(e)}(F(x_{j(e)}|\boldsymbol{x}_{D(e)}), F(x_{k(e)}|\boldsymbol{x}_{D(e)}))\right]$$

$$\times f_{v_j}(x_{v_j}) \left[\prod_{i=1}^{K} c_{v_j,w_i^{(j)}|D_{ij}}(F(x_{v_j}|\boldsymbol{x}_{D_{ij}}), F(x_{w_i^{(j)}}|\boldsymbol{x}_{D_{ij}}))\right], \tag{8.13}$$

where $E_i^{(j,K)}$ are the corresponding edge sets with $E_i^K \subseteq E_i^{(j,K)}$ and which depend on K and j.

Combining Equations (8.11) and (8.13), we obtain

$$f(x_{v_j}|\boldsymbol{x}_D) = \frac{f(\boldsymbol{x}_D, x_{v_j})}{f(\boldsymbol{x}_D)} = f_{v_j}(x_{v_j}) \left[\prod_{i=1}^{K} c_{v_j,w_i^{(j)}|D_{ij}}(F(x_{v_j}|\boldsymbol{x}_{D_{ij}}), F(x_{w_i^{(j)}}|\boldsymbol{x}_{D_{ij}}))\right], \tag{8.14}$$

for all $j = K+1, ..., d$, which we can directly plug into Equation (8.12):

$$f(\boldsymbol{x}_{D^c}|\boldsymbol{x}_D) = \left[\prod_{j=K+1}^{d} f_{v_j}(x_{v_j}) \prod_{i=1}^{K} c_{v_j,w_i^{(j)}|D_{ij}}(F(x_{v_j}|\boldsymbol{x}_{D_{ij}}), F(x_{w_i^{(j)}}|\boldsymbol{x}_{D_{ij}}))\right]$$
$$\times c_{D^c|D}(F(x_{v_{K+1}}|\boldsymbol{x}_D), ..., F(x_{v_d}|\boldsymbol{x}_D)). \tag{8.15}$$

Taking Equations (8.11) and (8.15) and plugging them into Equation (8.9), then yields the joint density as

$$
\begin{aligned}
f(x_1, ..., x_d) &= \left[\prod_{j=1}^{d} f_{v_j}(x_{v_j})\right] \left[\prod_{i=1}^{K-1} \prod_{e \in E_i^K} c_{j(e),k(e)|D(e)}(F(x_{j(e)}|\boldsymbol{x}_{D(e)}), F(x_{k(e)}|\boldsymbol{x}_{D(e)}))\right] \\
&\quad \times \left[\prod_{i=1}^{K} \prod_{j=K+1}^{d} c_{v_j,w_i^{(j)}|D_{ij}}(F(x_{v_j}|\boldsymbol{x}_{D_{ij}}), F(x_{w_i^{(j)}}|\boldsymbol{x}_{D_{ij}}))\right] \\
&\quad \times c_{D^c|D}(F(x_{v_{K+1}}|\boldsymbol{x}_D), ..., F(x_{v_d}|\boldsymbol{x}_D)) \\
&= \left[\prod_{k=1}^{d} f_k(x_k)\right] \left[\prod_{i=1}^{K} \prod_{e \in E_i} c_{j(e),k(e)|D(e)}(F(x_{j(e)}|\boldsymbol{x}_{D(e)}), F(x_{k(e)}|\boldsymbol{x}_{D(e)}))\right] \\
&\quad \times c_{D^c|D}(F(x_{v_{K+1}}|\boldsymbol{x}_D), ..., F(x_{v_d}|\boldsymbol{x}_D)),
\end{aligned}
$$

where the last equality follows from the statement of the auxiliary claim above. This concludes the proof of Theorem 8.2.   □

The idea of the proof can be summarized as follows: We want to show that R-vines with C-vine structure in higher order trees can be jointly simplified. To do so, we start by decomposing the joint density of $\mathbf{X} = (X_D, X_{D^c})$ into the density of the variables of the common conditioning set D and the density of the remaining conditioned variables in Equation (8.9). The variables of the common conditioning set can be modeled as an R-vine because their dependencies are "self-contained" by (8.10). By an application of the conditional Sklar's Theorem we further obtain a $(d - K)$-dimensional copula and univariate densities conditioned on D in Equation (8.11). By an auxiliary claim, these conditional densities can be decomposed into the unconditional densities and pair copulas linking the variables to the R-vine structure of the variables in D in each tree (Equation (8.14), cp. Example 12). These pair copulas then complement the R-vine structure of the variables in D so that we finally obtain an R-vine on K trees and a $(d - K)$-dimensional copula.

Example 12 (Joint simplification of R-vines.) *In this example, we consider the R-vine given in Figure 8.3 and formally link it to Theorem 8.2. Figure 8.6 shows the same R-vine which has now been simplified at level $K = 2$. As in Figure 8.5, the dashed lines in graph $\widetilde{T}_3$ indicate that this is not a tree anymore but a complete graph on three nodes to illustrate that the conditioned variables are jointly linked by the trivariate copula whose indices are denoted in the middle of the graph $\widetilde{T}_3$.*

Obviously, the common conditioning set is $D = \{1, 2\}$. Simplification at level $K = 1$ is not possible, since, e.g., $D(\{1, 3|2\}) = 2 \neq 1 = D(\{2, 4|1\})$. However, note that $D(\{1, 3|2\}) \subset D$ and $D(\{2, 4|1\}) \subset D$. The modeling of the variables in D as an R-vine is not very exciting in this example, because a simple bivariate copula is of course an R-vine.

What is more interesting here, is the statement of the auxiliary claim in the proof: in each tree we see that the pair copulas are linked to the set D, e.g., $\underline{1}, 5$ in tree 1 or $\underline{2}, 4|1$ in
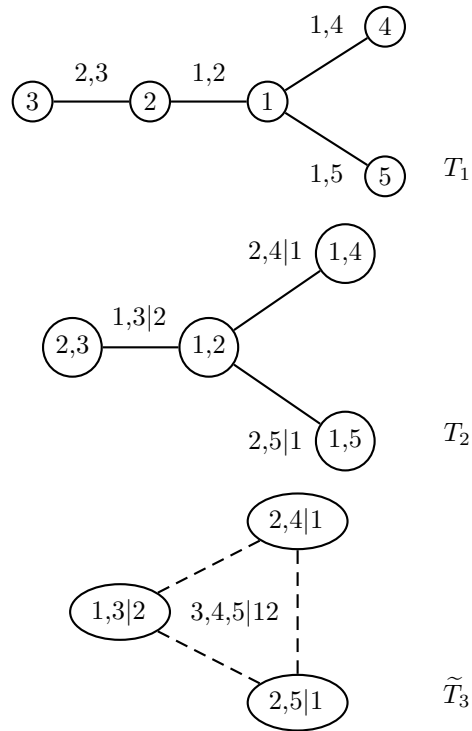
Figure 8.6: Jointly simplified R-vine at level $K = 2$ (cp. Figure 8.3).

tree 2. These pair copulas make up the trees around D, i.e., Theorem 8.2 can be thought of as a C-vine structure built around a "root set" which itself is modeled as an arbitrary R-vine.

Chapter 9

Model evaluation

Having fully specified an R-vine (e.g., using Algorithm 10), one is usually interested in evaluating the model with respect to different criteria, especially such that are important for the intended use of the model. If alternative models are also proposed, one can easily use one of the model selection criteria discussed in Chapter 5 for a relative comparison. In this chapter, we will concentrate on the problem that we are given only one model (since model specification is costly in terms of time and resources) and now want to assess whether it is reliable and adequate for future use, i.e., evaluate it in absolute terms. Model evaluation therefore proceeds by comparing characteristics of the observed data, which was used for model specification, with simulated observations from the specified R-vine model (cp. Dißmann (2010) for details on simulation from R-vines).

With regard to our fundamental issues of truncation and simplification, we consider a model $\mathcal{M}_T(K)$, $\mathcal{M}_S(K)$ or $\mathcal{M}_J(K)$ with density given in (6.1), (7.1) or (8.2), respectively, obtained from the respective chosen truncation or simplification procedure (see Chapters 6, 7 and 8) and now want to figure out whether the model has been reasonably simplified or truncated, i.e., the possible loss compared to a fully specified model should be small. If there are problems in the model specification, especially in higher order trees that have been simplified or truncated, we hope to detect them in the model evaluation. Hence, we assume in the following that we examine a model $\mathcal{M}(K)$ which has been stepwisely or jointly simplified or even truncated at a certain level K. Simulations from this model are denoted by $(\boldsymbol{u}_1^K, ..., \boldsymbol{u}_n^K)$, where $\boldsymbol{u}_i^K = (u_{i1}^K, ..., u_{id}^K)'$, $i = 1, ..., m$, with $m \geq n$ depending on the chosen criterion. Note that simulations are of course particularly easy to obtain for truncated models.

Model evaluation criteria can be divided into two groups: on the one hand, there is graphical model evaluation which is usually considered in two dimension, most notably using scatter plots. On the other hand, there is a wide range of summarizing criteria such as the comparison of dependence measures. Summarizing model evaluation is also more suitable in larger dimensions, since graphical methods then often become rather impractical. Nevertheless, we will first discuss graphical evaluation criteria, since they often motivate or directly lead to summarizing measures which are appropriate for higher dimensions. Finally, we consider the Value-at-Risk of a portfolio which is of particular interest in financial applications and can be used to test the forecasting accuracy of a model.

9.1 Graphical model evaluation

As noted above, graphical model evaluation concentrates on bivariate comparisons, since these are easier to work with and to interpret. The most natural method are bivariate scatter plots, while contour plots of variables with standard normal margins are often an interesting alternative. Depending on the specific application, special areas of bivariate plots should get attention, e.g., the bottom left and right upper corners to assess tail behavior when modeling financial risk. The comparison of empirical copula distribution functions and copula Q-Q plots are other insightful diagnostic tools.

9.1.1 Bivariate scatter plots

As stated above, the most natural graphical evaluation criterion is the comparison of bivariate scatter plots of $\{(u_{ir}^K, u_{is}^K), \ i = 1, ..., m\}$ to the observed ones $\{(u_{ir}, u_{is}), \ i = 1, ..., n\}$ for some $r, s \in \{1, ..., d\}$ (cp. Panel A of Figure 9.1). Often $m = n$ is used for direct comparability. However choices of m larger than n are also possible to get a better impression of the underlying distribution.

Special consideration should by given to those pairs which are not modeled unconditionally in the corresponding parametrized density (e.g., variables 1 and 3 in a D-vine with pair copula terms c_{12}, c_{23} and $c_{13|2}$) and, in particular, to those which are affected by simplification or truncation, since obviously conditional normality or independence are not equivalent to the respective unconditional property. For jointly simplified C-vines, this corresponds to the consideration of scatter plots for $r, s \in \{K + 1, ..., d\}$.

9.1.2 Contour plots

Often it might be easier to interpret plots of variables with standard normal margins. Therefore we propose to consider the bivariate scatter plots of $\{(z_{ir}^K, z_{is}^K), \ i = 1, ..., m\}$ versus $\{(z_{ir}, z_{is}), \ i = 1, ..., n\}$ for some $r, s \in \{1, ..., d\}$, where $z_{ir}^K := \Phi^{-1}(u_{ir}^K)$ and $z_{ir} := \Phi^{-1}(u_{ir})$. Interpretation becomes even easier, when contour plots are used rather than simple scatter plots (cp. Panels B and C of Figure 9.1). To compute the contour lines, it might in particular be sensible to choose $m > n$ to obtain more accurate results. Again, one should especially focus on variable pairs affected by simplification or truncation.

9.1.3 Empirical copula distribution functions

If one is particularly interested in an accurate modeling of the joint tail behavior of variables, it might be interesting to consider the empirical copula distribution functions in the tails, i.e., compute $C_n(\alpha, \alpha)$ (lower tail) and $C_n(1 - \alpha, 1 - \alpha)$ (upper tail) for $\alpha \in [0, 0.1]$ based on the empirical copula defined in (4.5) (also see Figure 3.5). Similarly compute the empirical copula based on the simulated observations and evaluate it in the lower and upper tails. The comparison of both observed and simulated empirical copulas gives an indication whether comovement in the tails is reproduced appropriately by model $\mathcal{M}(K)$. An example is shown in Panel D of Figure 9.1.
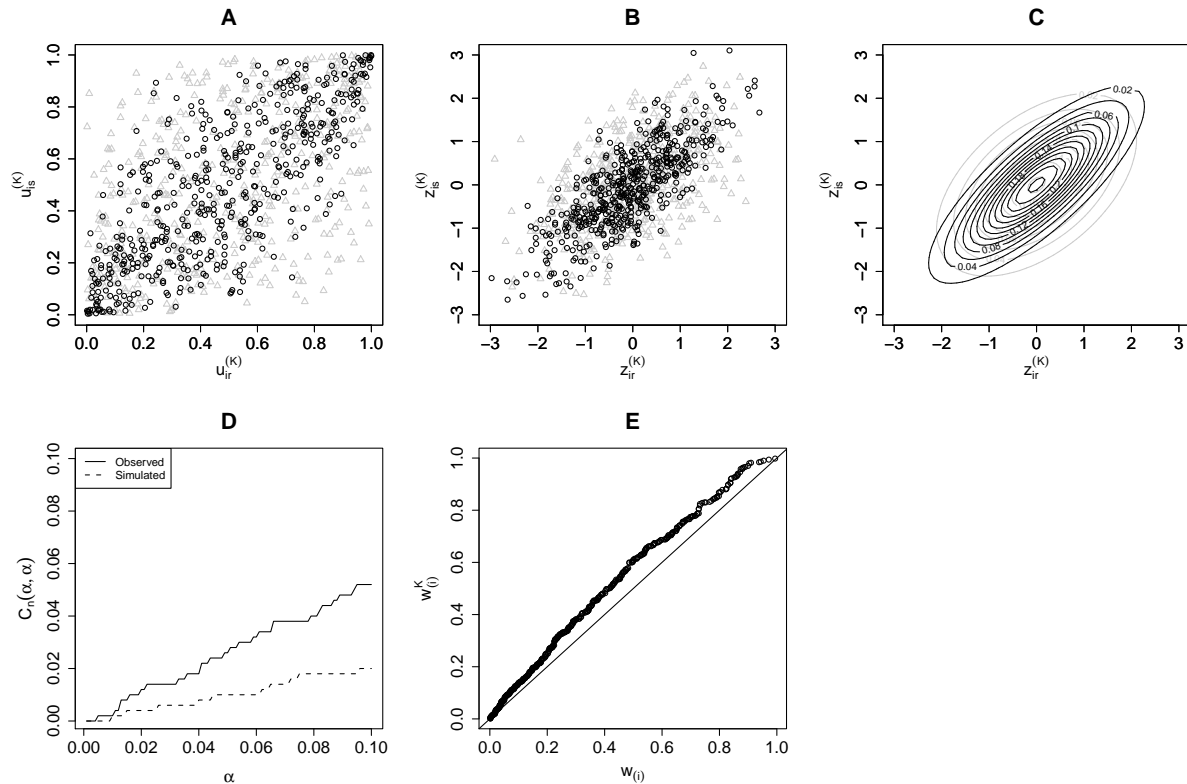
Figure 9.1: Graphical model evaluation criteria: superimposed scatter plots of copula data (Panel A) and data transformed with standard normal margins (Panel B), the corresponding contour plot (Panel C), a plot of the empirical copula distribution functions (Panel D) and the copula Q-Q plot of the data (Panel E). The observations are obtained from Gaussian copulas with parameter 0.8 (representing the "observed" data; black circles/lines) and 0.5 (representing the "simulated" data from model $\mathcal{M}(K)$; gray triangles/lines), respectively.

9.1.4 Copula Q-Q plots

An alternative diagnostic tool are Q-Q plots as proposed in Wilk and Gnanadesikan (1968). Such plots allow to compare two empirical cdf's, in our case the empirical cdf's of the observed and simulated data, where the observed data comes from the unknown true distribution and we want to investigate if the distribution induced by $\mathcal{M}(K)$ is close to this true distribution. Hence we compute

$$w_i^K = \frac{1}{n}\sum_{j=1}^{n}\mathbf{1}_{\{u_{jr}^K \leq u_{ir}^K, u_{js}^K \leq u_{is}^K\}}, \text{ and } w_i = \frac{1}{n}\sum_{j=1}^{n}\mathbf{1}_{\{u_{jr} \leq u_{ir}, u_{js} \leq u_{is}\}},$$

for some $r, s \in \{1, ..., d\}$. Then plot the order statistics $\{(w_{(i)}, w_{(i)}^K), \ i = 1, ..., n\}$, where necessarily $m = n$. If both cdf's are close to each other, the plot is similar to the line $y = x$. An example is shown in Panel E of Figure 9.1.

9.2 Summarizing evaluation criteria

While graphical diagnostic tools as discussed in the previous section are usually restricted to the bivariate case, summarizing criteria are, by their name, suitable to summarize information in arbitrary dimension. However, they are mostly based on bivariate measures that are aggregated in some sense. This approach corresponds to R-vine modeling where variables are also considered pairwisely.

When we discussed the appropriate choice of edge weights for the construction of R-vine trees in Section 3.1, we already introduced measures that detect different types of dependence in bivariate data. Obviously, it is important that the specified R-vine model $\mathcal{M}(K)$ can reproduce such dependencies and hence we also consider these measures in the context of model evaluation.

Instead of examining each of the $d(d-1)/2$ bivariate scatter or Q-Q plots separately, we can compute a summary measure for both the observed and the simulated observations and then consider the (absolute) differences. Recalling the weights discussed in Section 3.1, we can compute, e.g., Kendall's τ, lower and/or upper tail or exceedance dependence as well as tail cumulation which all have a graphical motivation. Note that for those pairs which are modeled explicitly (unconditionally) in the R-vine, we can even calculate theoretical values for some of the measures above based on the respective parameter estimates (cp. Table 2.1). Again, special attention should be paid to those pairs of variables that are influenced by simplification or truncation.

9.2.1 Portfolio evaluation of copula data

In certain circumstances, it might not only be too tedious to examine bivariate graphical diagnostics but also to consider each summarizing criterion individually. Then the information has to be aggregated in some way to obtain an overall performance measure. Such a measure can also be directly used for model comparison if more than one model is available.

The most obvious way of aggregation is certainly to consider the sum or the mean of the differences in the empirical bivariate summary measures based on observed versus simulated observations. However, we do not always want to rely on bivariate data characteristics and therefore we need alternative quantities. The most commonly used one is given by the mean of the copula data over its d components,

$$S_i^K := \frac{1}{d} \sum_{r=1}^{d} u_{ir}^K, \text{ and } S_i := \frac{1}{d} \sum_{r=1}^{d} u_{ir}, \tag{9.1}$$

for all $i = 1, ..., n$. The appropriateness of model $\mathcal{M}(K)$ can then be assessed by comparing histograms and empirical quantiles based on $\{S_i^K, \ i = 1, ..., n\}$ and $\{S_i, \ i = 1, ..., n\}$. Moreover, in financial applications we can easily extend the defintion of S_i^K and S_i in (9.1) to incorporate different portfolio weights:

$$\widetilde{S}_i^K := \sum_{r=1}^{d} w_r u_{ir}^K, \text{ and } \widetilde{S}_i := \sum_{r=1}^{d} w_r u_{ir}, \tag{9.2}$$

where $w_r \geq 0 \ \forall r = 1, ..., d$, and $\sum_{r=1}^{d} w_r = 1$. Note that the weights are of course the same for observed and simulated observations. The definitions in (9.1) then correspond to a portfolio with equal weights $w_r = \frac{1}{d} \ \forall r = 1, ..., d$. These portfolios of copula data are however not to be confounded with financial portfolios, where the marginal models also have to be taken into account in the portfolio evaluation (see Section 11.3.6).

9.2.2 Simplification based on data characteristics

The summary measures discussed in the previous sections cannot only be used for model evaluation, but they also directly motivate the construction of alternative simplification procedures based on data characteristics in contrast to the procedures based on statistical methods discussed in Chapters 6, 7 and 8. As in the pairwise simplification procedures based on the Vuong test and on AIC/BIC, we have to proceed iteratively and investigate whether additionally specified R-vine trees significantly increase the model fit. In particular, we work in the same setting as the methods based on the Vuong test as described in Section 7.2 because we need completely specified models when simulating from simplified R-vines, i.e., we cannot simply omit trees $T_{j+2}, ..., T_{d-1}$ when we simulate to determine whether simplification is possible in step j. This is not the case in the AIC/BIC-based procedure (cp. Section 7.3), where only tree T_{j+1} is considered in each step.

As we proceed iteratively, we always specify one additional tree T_{j+1} in each step j. In a hierarchical approach, we then investigate whether the marginal "gain" of the extra tree is "large enough" compared to a truncated tree. If not, truncate, but otherwise specify models $\mathcal{M}_S(j)$ and $\mathcal{M}_S(j+1)$ as described in Section 7.2 and compare again whether the additional "gain" of the extra tree is "large enough". Here, "gain" refers to an appropriate summarizing criterion, e.g., a specific quantile of the weighted sum of copula data as defined in (9.2), where the number of simulated observations should be chosen sufficiently large to ensure that the influence of the simulation error is negligible. In the following, assume that we are interested in large values of the chosen summarizing criterion, i.e., the criterion increases when the fit becomes more accurate. Finally, a threshold has to be defined in order to determine what "large enough" means. Such a threshold can also depend on step j, e.g., if the threshold is lower in higher order trees, the R-vine is more likely to be simplified.

However, this approach involves simulation from the R-vine specified up to a certain tree and is therefore computationally more demanding than the use of Vuong tests or AIC/BIC. Nevertheless, such a procedure is an interesting alternative due to its motivation based on data characteristics. Algorithm 13 summarizes the procedure using the same notation as in Algorithm 10. $\hat{S}$ denotes the chosen summarizing criterion and M_j is the threshold in step j. Alternatively tests such as $\hat{S}_2 / \hat{S}_1 \leq 1 + \varepsilon_j$ with relative thresholds ε_j are, of course, also possible.

Algorithm 13 Simplification of R-vines based on data characteristics.

Input: (Pseudo-)observations of d variables, summarizing criterion $\hat{S}$, thresholds M_j.

1: **for** $j = 0, ..., d - 2$ **do**
2:     Compute pairwise weights for the transformed observations from the previous step (using one of the methods described in Section 3.1) and use them to construct tree T_{j+1} with the appropriate method from Section 3.2.
3:     Select copulas as described in Section 5.4, estimate parameters using maximum likelihood estimation (cp. Section 4.1.2) and compute transformed observations according to (2.19).
4:     Compute $\hat{S}$ for models $\mathcal{M}_T(j)$ (model 1) and $\mathcal{M}_T(j + 1)$ (model 2).
5:     **if** $\hat{S}_1 + M_j \geq \hat{S}_2$ **then**
6:         Truncate the R-vine at level $K = j$, i.e., exit the loop with model $\mathcal{M}_T(j)$.
7:     **else**
8:         Specify model $\mathcal{M}_S(j)$: sequentially construct higher order trees $T_{j+1}, ..., T_{d-1}$ with bivariate Gaussian copulas.
9:         Specify model $\mathcal{M}_S(j + 1)$: sequentially construct higher order trees $T_{j+2}, ..., T_{d-1}$ with bivariate Gaussian copulas.
10:       Compute $\hat{S}$ for models $\mathcal{M}_S(j)$ (model 1) and $\mathcal{M}_S(j + 1)$ (model 2).
11:       **if** $\hat{S}_1 + M_j \geq \hat{S}_2$ **then**
12:         Simplify the R-vine at level $K = j$, i.e., exit the loop with model $\mathcal{M}_S(j)$.
13:       **end if**
14:     **end if**
15: **end for**

Output: Pairwisely simplified K level R-vine, pairwisely truncated R-vine at level K, or fully specified R-vine, if neither simplification nor truncation are possible.

9.3 Value-at-Risk

In finance the *Value-at-Risk (VaR)* is the most popular risk measure and often used to backtest models for market risk management. According to McNeil et al. (2005) for a *confidence level* α the $(1 - \alpha)$-*VaR* of a portfolio of assets with return R is defined as the smallest number ℓ such that the probability of the portfolio loss exceeding ℓ is not larger than α, or, equivalently, as minus the largest value s such that the probability of the portfolio return dropping below s is not larger than α, i.e.,

$$\text{VaR}(1 - \alpha) = \inf\{\ell : P(-R > \ell) \leq \alpha\} = -\sup\{s : P(R < s) \leq \alpha\}. \quad (9.3)$$

If the (unknown) return distribution is continuous, this definition corresponds to minus the α-quantile of the return distribution.

In applications we are interested in evaluating a time series of ex-ante VaR forecasts, $\text{VaR}_t(1 - \alpha)$, $t = 1, ..., n$, corresponding to a time series of portfolio returns, R_t, $t = 1, ..., n$, where the forecasts are obtained from the model under consideration (see Section 11.3.6 for more details regarding VaR forecasting from R-vine models). In order to do so, we

define the hit sequence I_t, $t = 1, ..., n$, of ex-post loss exceedances as

$$I_t = \begin{cases} 1, & \text{if } R_t < -\text{VaR}_t(1 - \alpha) \\ 0, & \text{else} \end{cases}.$$

Christoffersen (1998) then investigates the null hypothesis that the exceedances $(I_t)_{t=1,...,n}$ are i.i.d. Bernoulli distributed with probability α according to the definition of the VaR in (9.3) and refers to this as the test of correct *conditional coverage*.

The test is split into two sub-tests: since the proportion of exceedances should approximately equal the confidence level α of the VaR forecasts, the hypothesis of correct *unconditional coverage*,

$$H_0 : E(I_t) = \alpha \qquad \text{against} \qquad H_1 : E(I_t) \neq \alpha, \tag{9.4}$$

is investigated. This test implicitly assumes independent exceedances, which are tested explicitly by considering $(I_t)_{t=1,...,n}$ as a first order Markov chain with transition probability matrix

$$P = \begin{pmatrix} 1 - \pi_{01} & \pi_{01} \\ 1 - \pi_{11} & \pi_{11} \end{pmatrix},$$

where $\pi_{ij} = P(I_t = j | I_{t-1} = i)$. As exceedances should occur independently without clustering, we then examine the hypothesis of *independence*,

$$H_0 : \pi_{01} = \pi_{11} \qquad \text{against} \qquad H_1 : \pi_{01} \neq \pi_{11}. \tag{9.5}$$

Combining tests (9.4) and (9.5) then gives the joint test of coverage and independence to investigate the hypothesis of conditional coverage,

$$H_0 : \pi_{01} = \pi_{11} = \alpha \qquad \text{against} \qquad H_1 : \text{ not } H_0. \tag{9.6}$$

For the unconditional coverage test we have to consider the likelihood function of a sample of n i.i.d. Bernoulli distributed random variables with unknown probability π_1, which is

$$L(\pi_1) = \pi_1^{n_1}(1 - \pi_1)^{n - n_1},$$

where n_1 is the number of ones in the sample. The maximum likelihood estimate of π_1 is $\hat{\pi}_1 = n_1/n$ and hence the likelihood ratio test statistic of the unconditional coverage test is given by

$$LR_{uc} = 2(\log L(\hat{\pi}_1) - \log L(\alpha)). \tag{9.7}$$

If n_{ij} denotes the number of observations with value i followed by j, the likelihood under the alternative hypothesis of the independence test is

$$L(\pi_{01}, \pi_{11}) = \pi_{01}^{n_{01}}(1 - \pi_{01})^{n_0 - n_{01}} \pi_{11}^{n_{11}}(1 - \pi_{11})^{n_1 - n_{11}},$$

with maximum likelihood estimates $\hat{\pi}_{01} = n_{01}/n_0$ and $\hat{\pi}_{11} = n_{11}/n_1$. Then the likelihood ratio test statistic of the independence test is given by

$$LR_{ind} = 2(\log L(\hat{\pi}_{01}, \hat{\pi}_{11}) - \log L(\hat{\pi}_1)). \tag{9.8}$$

Combining both test statistics (9.7) and (9.8) we obtain the likelihood ratio test statistic of the conditional coverage test as

$$LR_{cc} = LR_{uc} + LR_{ind} = 2(\log L(\hat{\pi}_{01}, \hat{\pi}_{11}) - \log L(\alpha)).$$

Christoffersen (1998) shows that the test statistics LR_{uc} and LR_{ind} are both asymptotically χ^2 with 1 degree of freedom and hence the test statistic of the conditional coverage is also asymptotically χ^2 but with 2 degrees of freedom.

To sum it up, we can evaluate a model with respect to the accuracy of its VaR forecasts by checking whether the proportion of VaR exceedances approximately equals the VaR confidence level and whether exceedances do not occur in clusters. This leads to the likelihood ratio test of conditional coverage for which we know the asymptotic distribution to obtain approximate p-values.

Chapter 10

Simulation study

In Chapters 6, 7 and 8 we developed R-vine simplification and truncation procedures which rely on a couple of assumptions and approximations. In particular, the methods we developed in Sections 6.2, 6.3, 7.2 and 7.3 are heuristics to ensure that the computational complexity is limited to a reasonable amount. Hence, we will investigate in different settings whether these procedures are adequate to determine simplification and truncation levels. Special attention is paid to the simplification procedure based on AIC/BIC, since it relies on more approximations than the one based on the Vuong test. For the latter procedure, we consider the Akaike and Schwarz corrections as well as no correction at all (cp. Section 5.3.1). In the following, let *V.Akaike* and *V.Schwarz* denote model selection using the Vuong test with the Akaike and Schwarz corrections, respectively.

Another important issue discussed in this chapter is the joint simplification of C-vines as presented in Section 8.1. Besides investigating whether joint simplification works correctly, we compare the resulting correlation matrix of the multivariate Gaussian copula when using the "correct" procedure to the correlation matrix computed from Gaussian pair copulas when using one of our stepwise simplification methods.

Each time we also specify the non-truncated and non-simplified full model as benchmark model in order to investigate the loss involved by simplification or truncation and to distinguish shortcomings of the procedures from randomness due to simulation and from problems in the model reconstruction due to finite sample sizes and heuristic construction methods (cp. Chapter 3). Estimates are obtained by sequential estimation, since up to eight different models (one for each procedure under consideration) have to be specified in each simulation and hence the computational effort is quite high. The procedures are then evaluated with regard to a range of criteria.

(i) *Simplification/truncation level:* First of all, we are of course interested in the ability of a procedure to identify the correct simplification or truncation level K_0. As the procedure might simplify or truncate too early or too late at level $K \in \{0, ..., d-1\}$, we use a scoring method: each time K_0 is identified correctly, the score is increased by 1; if however $K \neq K_0$ but K is close to K_0, the score is increased by a value smaller than 1, which depends on the specific scenario (if $d = 100$ and $K_0 = 20$, $K = 25$ might be close, while $K = 25$ is not quite good if $d = 30$ and $K_0 = 20$).

(ii) *Mean KLIC:* We also compute empirical KLIC's (5.1) to compare the models ob-

tained by the different procedures (note that we know the true model that we simulate from) and then take the mean over all simulations:

$$\overline{KLIC}(h_0) := \frac{1}{R} \sum_{r=1}^{R} \widehat{KLIC}(h_0, f_r, \hat{\boldsymbol{\theta}}_r),$$

where R is the number of repeated simulations and h_0 the density of the true vine model, while the density of the approximating vine model in the r-th repetition is given by f_r with estimated parameter $\hat{\boldsymbol{\theta}}_r$ (cp. the general R-vine density as given by (2.27) and the truncated and simplified R-vine densities as defined in (6.1) and in (7.1)). The empirical KLIC for $r = 1, ..., R$ is defined as

$$\widehat{KLIC}(h_0, f_r, \hat{\boldsymbol{\theta}}_r) := \frac{1}{n} \sum_{i=1}^{n} \log \left[h_0(\boldsymbol{u}_i^{0(r)}) \right] - \frac{1}{n} \sum_{i=1}^{n} \log \left[f_r(\boldsymbol{u}_i^{0(r)} | \hat{\boldsymbol{\theta}}_r) \right],$$

where n is the sample size and $\boldsymbol{u}_i^{0(r)}$ the i-th simulated observation from the true vine model with density h_0.

(iii) *Vuong tests:* Furthermore, we perform Vuong tests with and without correction (see Section 5.3) between the true model and the respective fitted model to examine whether the model is at least close to the true model if K_0 is not correctly identified. Thus, we count how often the simplified/truncated model cannot be distinguished from (or is superior to) the true model at significance level $\alpha = 0.05$, i.e., how often the null hypothesis that both models are equivalent cannot be rejected or the simplified/truncated model is better than the true model.

(iv) *τ-matrix and quartile differences:* In consideration of the model evaluation criteria discussed in Chapter 9, we simulate data from all models, $\boldsymbol{u}_i^{K(r)}$, $i = 1, ..., n$, $r = 1, ..., R$, and compute two different summarizing criteria (cp. Section 9.2): the sum of absolute differences between empirical Kendall's τ's and between empirical quartiles of the equally weighted copula data portfolios $S_i^{K(r)}$ and $S_i^{0(r)}$, $i = 1, ..., n$, $r = 1, ..., R$, as defined in (9.1), where $S_i^{0(r)}$ denotes the portfolio value with respect to the simulated data from the true model. The mean results of all repetitions are reported:

$$\overline{\tau}^{0,K} := \frac{1}{R} \sum_{r=1}^{R} \sum_{\{s,t\} \in \binom{\{1,...,d\}}{2}} |\hat{\tau}_n((u_{is}^{0(r)})_i, (u_{it}^{0(r)})_i) - \hat{\tau}_n((u_{is}^{K(r)})_i, (u_{it}^{K(r)})_i)|,$$

$$\overline{q}^{0,K} := \frac{1}{R} \sum_{r=1}^{R} \sum_{p \in \{0, 0.25, 0.5, 0.75, 1\}} |\hat{F}_{S^{0(r)}}^{-1}(p) - \hat{F}_{S^{K(r)}}^{-1}(p)|,$$

(10.1)

where $\binom{\{1,...,d\}}{2} := \{\{s,t\} : s \neq t, \; s,t \in \{1, ..., d\}\}$ is the set of unordered pairs in $\{1, ..., d\}$, $(u_{is}^{0(r)})_i := (u_{is}^{0(r)})_{i=1,...,n}$, $(u_{is}^{K(r)})_i := (u_{is}^{K(r)})_{i=1,...,n}$, and $\hat{F}_{S^{0(r)}}$ and $\hat{F}_{S^{K(r)}}$, $r = 1, ..., R$, are the empirical cdf's of $S_i^{K(r)}$ and $S_i^{0(r)}$, $i = 1, ..., n$, respectively.

Table 10.1 summarizes the evaluation criteria. Simulation results are shown at the end of each section.

evaluation criterion	range	target value
truncation/simplification level	$[0, 100]$	100
mean KLIC	$\mathbb{R}$	$-\infty$
Vuong tests	$[0, 100]$	100
τ-matrix differences	$\mathbb{R}_{\geq 0}$	0
quartile differences	$\mathbb{R}_{\geq 0}$	0

Table 10.1: Evaluation criteria for vine models in simulation studies.

10.1 Regular vine in seven dimensions

This first study is based on the seven-dimensional R-vine which was given as an example in Section 2.4 (Figure 2.3). As a reminder it is shown again on a smaller scale in Figure 10.1 with the pair copulas according to the third truncation scenario below. Using this R-vine structure, we investigate first the truncation procedures given in Algorithms 6 and 7 and then simplification as described in Algorithms 8 and 9. For each problem, we developed four different scenarios to examine the performance of the respective procedures. Within each scenario we simulated $n \in \{500, 1000\}$ observations from the respective specification. Trees are constructed using Kendall's τ as weights (cp. Chapter 3). Copulas are selected using the AIC (cp. Section 5.4). In consideration of the dimension being $d = 7$, a score of 0.5 is assigned if

$$|K - K_0| = 1.$$

10.1.1 Truncation

In each of the following four scenarios, the truncation level is $K_0 = 2$, i.e., trees $T_3, ..., T_6$ are specified with independence copulas. Hence, the scenarios only differ in the first two trees T_1 and T_2. Copula parameters are chosen with respect to Kendall's τ (cp. Table 2.1).

- **Scenario 1:** Clayton copulas in T_1 and T_2 with Kendall's τ's 0.6 (T_1) and 0.3 (T_2).

- **Scenario 2:** t copulas in T_1 and T_2 with Kendall's τ's 0.6 (T_1) and 0.3 (T_2) and degrees of freedom 3 (T_1) and 7 (T_2).

- **Scenario 3:** Mixed copulas (t, Clayton, Gumbel) in T_1 and T_2 with Kendall's τ's 0.6 (T_1) and 0.3 (T_2) and degrees of freedom 3 (T_1) and 7 (T_2). t copulas are chosen for the "central" pairs $2, 3$ and $3, 6$ as well as $2, 6|3$. For all other pairs, Clayton or Gumbel copulas are chosen (cp. Figure 10.1).

- **Scenario 4:** t copulas in T_1 and T_2 with mixed Kendall's τ's between 0.6 and 0.8 in T_1 and between 0.2 and 0.4 in T_2. Degrees of freedom are chosen between 3 and 5 in T_1 and between 7 and 9 in T_2. The first two trees of this scenario are shown in Figure 10.2.
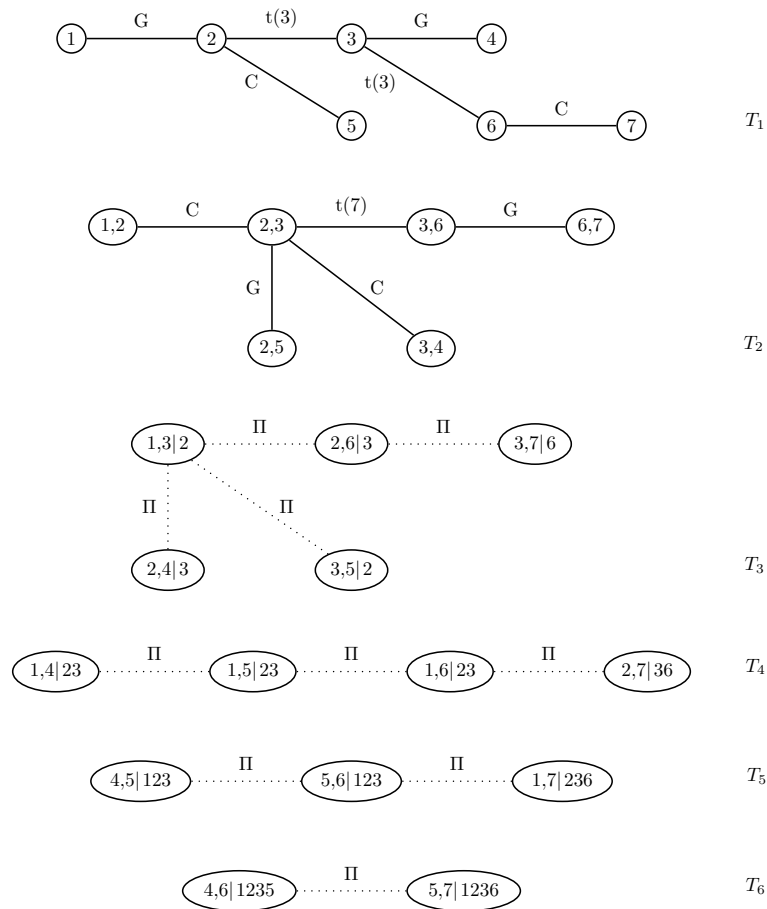
Figure 10.1: Seven-dimensional R-vine structure for simulation studies with copulas according to the third truncation scenario. C denotes the Clayton, G the Gumbel and Π the independence copula. Degrees of freedom of the t copulas are denoted in brackets.
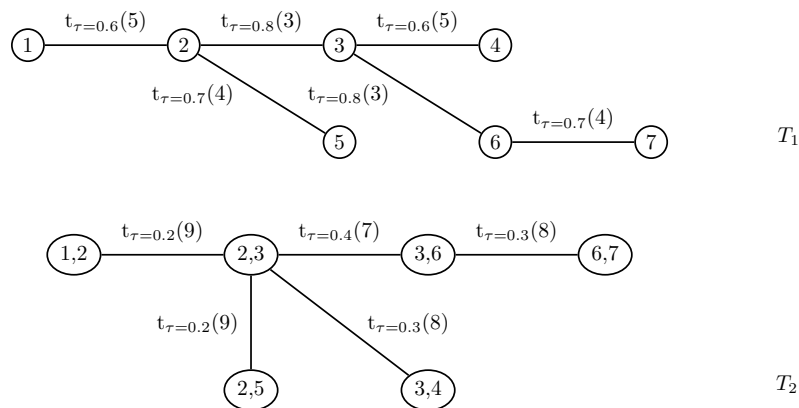

Figure 10.2: First two trees of the seven-dimensional R-vine structure for simulation studies with copulas according to the fourth truncation scenario. The indices give Kendall's τ's to determine the parameters of the t copulas, degrees of freedom are denoted in brackets.

Results for $R = 100$ repetitions are shown in Table 10.2 and can be summarized as follows:

(i) Truncation based on the Vuong test always identifies the truncation level more accurately than the AIC/BIC based procedures. Using the parsimonious Schwarz correction and the BIC work best within the respective class of procedures.

(ii) There is a trade-off between increased truncation level identification accuracy and small KLIC. Closeness to the true model determined by Vuong tests confirms this (there is a close relationship between KLIC's and Vuong tests as discussed in Section 5.3).

(iii) The truncated models exhibit slightly increased differences of Kendall's τ's and quartiles of S_i^K and S_i^0, but they are evidently still quite close to the fully specified model.

(iv) The effect of an increasing number of observations is heterogeneous. In terms of the differences of Kendall's τ's and quartiles of S_i^K and S_i^0, the performance always increases, which is partly due to larger sample sizes.

(v) Models with t copulas in the first trees are well reconstructed as indicated by the mean KLIC's and the results of Vuong tests in scenarios 2 and 4. This is probably due to the use of Kendall's τ as weight measure for R-vine construction, which cannot account for asymmetric tail dependence such as in Clayton or Gumbel copulas in scenarios 1 and 3.

(vi) All procedures work best in scenario 4. This suggests that model identification is facilitated when different dependencies are present in the first trees and hence important dependencies are easier to detect.

(vii) The method based on AIC/BIC is certainly an alternative to the one using Vuong tests, since truncation levels are almost equally well identified in scenarios 1, 2 and 3, and if the truncation level is not correctly identified, the chosen model is at least close to the full model.

10.1.2 Simplification

Again we examine four different scenarios with simplification level $K_0 = 2$. But now trees $T_3, ..., T_6$ are specified with Gaussian copulas with decreasing correlations: 0.25 in T_3, 0.20 in T_4, 0.15 in T_5 and 0.10 in T_6, corresponding to Kendall's τ's smaller than 0.16. The scenarios only differ in the specification of the first two trees T_1 and T_2 which are chosen as in Section 10.1.1.

- **Scenario 1:** Clayton copulas in T_1 and T_2 with Kendall's τ's 0.6 (T_1) and 0.3 (T_2).

- **Scenario 2:** t copulas in T_1 and T_2 with Kendall's τ's 0.6 (T_1) and 0.3 (T_2) and degrees of freedom 3 (T_1) and 7 (T_2).

- **Scenario 3:** Mixed copulas (t, Clayton, Gumbel) in T_1 and T_2 with Kendall's τ's 0.6 (T_1) and 0.3 (T_2) and degrees of freedom 3 (T_1) and 7 (T_2) (cp. Figure 10.1).

- **Scenario 4:** t copulas in T_1 and T_2 with mixed Kendall's τ's between 0.6 and 0.8 in T_1 and between 0.2 and 0.4 in T_2. Degrees of freedom are chosen between 3 and 5 in T_1 and between 7 and 9 in T_2 (cp. Figure 10.2).

Results for $R = 100$ repetitions are shown in Table 10.3. The main findings are:

(i) Again the procedure based on the Vuong test performs better than the AIC/BIC based approach. This is due to the additional approximations made in the latter simplification method. The more parsimonious selection criteria are also superior to the respective alternative criteria.

(ii) The trade-off between increased accuracy of simplification level identification and small mean KLIC's is most pronounced in scenarios 1 and 2. In scenario 3, the specified models are not quite close to the true model.

(iii) As before, the differences of Kendall's τ's and quartiles of S_i^K and S_i^0 are slightly increased for the simplified models, which are however still quite accurate compared to the non-simplified full model.

(iv) General performance increases with increasing sample sizes only in scenario 2 and 3, but always for the differences of Kendall's τ's and quartiles of S_i^K and S_i^0.

(v) Scenarios 2 and 4 with t copulas in the first trees again give the best results for all models. Especially, in scenario 4, the results are quite good. Even the AIC/BIC based procedure works well here.

(vi) Although the method based on AIC/BIC performs rather poorly with respect to the identification of the simplification level in scenarios 1, 2 and 3, it can be seen as a valid alternative to the procedure based on the Vuong test, since models are again close to the full model, if the simplification level is not identified correctly. Moreover, in scenario 4, the method performs fairly well.

scenario	n	eval. crit.	full	TRUNCATION PROCEDURE				
				Vuong	V.Akaike	V.Schwarz	AIC	BIC
1	500	truncation level	-	22.5	23.5	26.0	19.0	21.0
		mean KLIC	11.09	15.15	16.35	17.67	12.90	13.02
		Vuong test	70	64	63	61	67	67
		V. test (Akaike)	52	51	50	48	51	51
		V. test (Schwarz)	29	29	29	29	29	29
		τ-matrix diff.	0.495	0.515	0.522	0.475	0.472	0.482
		quartile diff.	0.078	0.080	0.079	0.074	0.078	0.079
	1000	truncation level	-	15.5	15.5	17.5	13.0	15.0
		mean KLIC	28.79	37.43	37.43	40.74	35.77	36.72
		Vuong test	32	31	31	31	32	31
		V. test (Akaike)	25	24	24	24	25	24
		V. test (Schwarz)	17	17	17	17	17	17
		τ-matrix diff.	0.327	0.343	0.345	0.360	0.353	0.356
		quartile diff.	0.058	0.054	0.058	0.056	0.059	0.051
2	500	truncation level	-	32.5	33.5	35.0	30.0	30.5
		mean KLIC	-1.38	2.40	3.13	4.81	0.66	0.73
		Vuong test	98	92	91	88	95	95
		V. test (Akaike)	92	87	86	83	89	89
		V. test (Schwarz)	67	66	65	64	66	66
		τ-matrix diff.	0.452	0.489	0.476	0.475	0.464	0.488
		quartile diff.	0.067	0.065	0.067	0.062	0.067	0.068
	1000	truncation level	-	31.0	31.5	33.0	26.0	30.5
		mean KLIC	10.29	11.94	12.03	16.45	11.51	11.88
		Vuong test	80	79	79	76	79	79
		V. test (Akaike)	58	54	54	53	56	55
		V. test (Schwarz)	32	32	32	32	32	32
		τ-matrix diff.	0.326	0.312	0.324	0.319	0.313	0.300
		quartile diff.	0.044	0.047	0.047	0.044	0.046	0.045
3	500	truncation level	-	42.0	42.5	43.5	34.5	38.5
		mean KLIC	63.40	65.11	65.21	65.66	64.57	64.80
		Vuong test	26	25	25	25	26	25
		V. test (Akaike)	11	10	10	10	10	10
		V. test (Schwarz)	7	7	7	7	7	7
		τ-matrix diff.	0.463	0.484	0.493	0.472	0.480	0.484
		quartile diff.	0.078	0.072	0.078	0.073	0.073	0.076
	1000	truncation level	-	48.5	48.5	49.0	39.5	43.5
		mean KLIC	110.56	111.45	111.45	112.17	110.66	110.99
		Vuong test	16	15	15	15	16	15
		V. test (Akaike)	14	14	14	14	14	14
		V. test (Schwarz)	9	9	9	9	9	9
		τ-matrix diff.	0.342	0.342	0.359	0.361	0.378	0.358
		quartile diff.	0.054	0.057	0.051	0.054	0.057	0.054
4	500	truncation level	-	61.5	74.0	95.5	49.5	53.5
		mean KLIC	-0.74	2.56	5.43	13.28	0.75	1.27
		Vuong test	99	96	91	74	99	97
		V. test (Akaike)	99	98	96	85	99	98
		V. test (Schwarz)	99	99	99	96	99	99
		τ-matrix diff.	0.397	0.389	0.419	0.417	0.418	0.377
		quartile diff.	0.071	0.069	0.069	0.072	0.072	0.074
	1000	truncation level	-	49.5	50.5	68.5	42.5	47.5
		mean KLIC	7.39	8.89	9.11	16.74	7.97	8.64
		Vuong test	85	79	78	63	82	80
		V. test (Akaike)	80	77	77	64	78	77
		V. test (Schwarz)	67	65	65	58	67	66
		τ-matrix diff.	0.271	0.260	0.280	0.286	0.281	0.267
		quartile diff.	0.050	0.046	0.046	0.046	0.042	0.044

Table 10.2: Simulation results of the four stepwise truncation scenarios described in Section 10.1.1. Each column corresponds to a different truncation procedure.

scenario	n	eval. crit.	full	Vuong	V.Akaike	V.Schwarz	AIC	BIC
					SIMPLIFICATION PROCEDURE			
1	500	simplification level	-	41.5	42.0	45.5	0.5	1.0
		mean KLIC	7.91	17.96	18.62	22.00	8.12	8.73
		Vuong test	91	70	68	62	90	89
		V. test (Akaike)	87	59	57	52	85	85
		V. test (Schwarz)	68	46	44	41	67	67
		τ-matrix diff.	0.431	0.442	0.429	0.433	0.462	0.440
		quartile diff.	0.076	0.076	0.071	0.075	0.080	0.075
	1000	simplification level	-	23.5	23.0	28.5	2.0	4.5
		mean KLIC	33.04	43.27	43.10	46.31	33.48	34.07
		Vuong test	28	20	20	16	28	27
		V. test (Akaike)	22	15	15	12	22	21
		V. test (Schwarz)	11	8	8	7	10	10
		τ-matrix diff.	0.330	0.309	0.345	0.316	0.318	0.325
		quartile diff.	0.054	0.055	0.055	0.056	0.056	0.052
2	500	simplification level	-	64.0	68.0	72.5	3.5	13.5
		mean KLIC	-2.72	10.81	14.67	29.16	-2.53	-0.26
		Vuong test	97	79	72	51	97	93
		V. test (Akaike)	97	81	75	54	97	94
		V. test (Schwarz)	93	81	79	64	93	91
		τ-matrix diff.	0.444	0.445	0.415	0.456	0.428	0.434
		quartile diff.	0.069	0.072	0.064	0.070	0.070	0.071
	1000	simplification level	-	41.5	52.0	68.5	8.5	16.0
		mean KLIC	4.03	11.08	16.62	26.29	4.59	6.75
		Vuong test	87	78	67	55	86	84
		V. test (Akaike)	82	72	64	54	81	79
		V. test (Schwarz)	68	62	59	53	68	67
		τ-matrix diff.	0.304	0.308	0.303	0.304	0.326	0.281
		quartile diff.	0.048	0.048	0.049	0.045	0.045	0.047
3	500	simplification level	-	64.5	66.0	70.5	5.5	11.5
		mean KLIC	87.05	97.59	97.91	103.34	87.24	87.83
		Vuong test	7	7	7	7	7	7
		V. test (Akaike)	7	7	7	7	7	7
		V. test (Schwarz)	8	8	8	7	8	8
		tau-matrix diff.	0.461	0.470	0.473	0.457	0.463	0.456
		quartile diff.	0.074	0.077	0.076	0.076	0.078	0.073
	1000	simplification level	-	52.5	56.0	62.0	15.5	17.0
		mean KLIC	117.03	123.74	124.69	128.59	117.42	117.70
		Vuong test	16	16	16	16	16	16
		V. test (Akaike)	16	15	15	15	16	15
		V. test (Schwarz)	15	15	15	15	15	15
		τ-matrix diff.	0.331	0.345	0.335	0.336	0.354	0.329
		quartile diff.	0.051	0.053	0.053	0.054	0.052	0.055
4	500	simplification level	-	93.5	96.0	81.5	50.5	56.0
		mean KLIC	-1.93	3.20	4.10	11.16	-1.53	-1.22
		Vuong test	99	93	93	75	97	97
		V. test (Akaike)	100	100	100	99	100	100
		V.Schwarz	100	100	100	100	100	100
		τ-matrix diff.	0.370	0.372	0.401	0.409	0.378	0.357
		quartile diff.	0.071	0.068	0.067	0.066	0.068	0.067
	1000	simplification level	-	70.0	78.5	95.5	46.0	50.5
		mean KLIC	4.90	7.71	9.78	15.04	5.00	5.24
		Vuong test	90	87	81	66	90	90
		V. test (Akaike)	100	99	98	91	100	100
		V.Schwarz	100	100	100	100	100	100
		τ-matrix diff.	0.292	0.275	0.271	0.260	0.285	0.259
		quartile diff.	0.048	0.050	0.053	0.048	0.053	0.048

Table 10.3: Simulation results of the four stepwise simplification scenarios described in Section 10.1.2. Each column corresponds to a different simplification procedure.

10.2 Joint truncation and simplification of canonical vines

This simulation study is devoted to the investigation whether joint truncation and simplification of C-vines work well. In particular, we examine whether stepwise procedures (either based on the Vuong test or on AIC/BIC) are capable of accurately reproducing the correlation matrix of the simplifying multivariate Gaussian copula (cp. Section 8.1.1). Moreover, in the simplification scenarios, we compare the results of joint C-vine simplification based on the two different goodness-of-fit tests presented in Section 4.2 (based on the empirical copula process or on Rosenblatt's transformation), while the performance of the multivariate independence tests based on the empirical copula process and on Spearman's ρ (see Section 4.3) is compared in the truncation scenarios. To do this, we examine four scenarios of differently specified truncated and simplified ten-dimensional C-vines, respectively. In each scenario the truncation/simplification level is $K_0 = 2$ and a score of 0.5 is assigned if

$$|K - K_0| = 1,$$

as in Section 10.1.

For each scenario we simulate $n \in \{500, 1000\}$ observations. Trees are constructed with Kendall's τ as weights. Bivariate copulas are chosen according to the AIC.

10.2.1 Truncation

As noted above, we are primarily interested in the performance of the joint truncation procedures using the multivariate independence tests based on the empirical copula process and on Spearman's ρ as discussed in Sections 4.3.2 and 4.3.3, respectively. Their results are compared to the results of the pairwise truncation methods in order to obtain a complete view of adequate procedures.

The four scenarios under consideration are chosen similar to the scenarios in Section 10.1.1 with the difference that we consider a ten-dimensional C-vine now. Given that the truncation level is 2, all remaining trees $T_3, ..., T_9$ are specified with independence copulas, while the first two trees T_1 and T_2 are specified as follows with parameters chosen according to Table 2.1:

- **Scenario 1:** Clayton copulas in T_1 and T_2 with Kendall's τ's 0.6 (T_1) and 0.3 (T_2).

- **Scenario 2:** t copulas in T_1 and T_2 with Kendall's τ's 0.6 (T_1) and 0.3 (T_2) and degrees of freedom 3 (T_1) and 7 (T_2).

- **Scenario 3:** Mixed copulas (t, Clayton, Gumbel) in T_1 and T_2 with Kendall's τ's 0.6 (T_1) and 0.3 (T_2) and degrees of freedom 3 (T_1) and 7 (T_2).[1]

---

[1]Pairs with t copulas: 1,4; 1,7; 1,10; 2,5|1; 2,8|1. Pairs with Clayton copulas: 1,3; 1,6; 1,9; 2,4|1; 2,7|1; 2,10|1. Pairs with Gumbel copulas: 1,2; 1,5; 1,8; 2,3|1; 2,6|1; 2,9|1.

- **Scenario 4:** t copulas in T_1 and T_2 with mixed Kendall's τ's between 0.6 and 0.8 in T_1 and between 0.2 and 0.4 in T_2. Degrees of freedom are chosen between 3 and 5 in T_1 and between 7 and 9 in T_2.[2]

In Tables 10.4 and 10.5, the results for $R = 100$ repetitions are shown. They can be summarized as follows:

(i) As before, there is a trade-off between accuracy of truncation level identification and KLIC's: the better the identification accuracy the higher the KLIC's. However, given the fully specified model as benchmark model, all truncated models are still rather good approximations. Note that negative KLIC's indicate that the respective model provides a better fit than the actual true model which is possible due to the simulation error.

(ii) Accuracy in terms of τ-matrix and quartile differences increases with increasing number of observations. This is not necessarily true for the other criteria. Especially the results in scenario 4 get worse.

(iii) The parsimonious procedures based on the Schwarz correction and on the BIC identify the truncation level more accurately than alternative procedures. Altogether, the performance of the methods based on the AIC/BIC is inferior to the methods based on the Vuong test and on independence tests, which both show similar results.

(iv) Both procedures based on independence tests perform equally well, where the test based on the Spearman's ρ is slightly more accurate. Hence, both methods are adequate to identify jointly truncated C-vine models.

10.2.2 Simplification

After having discussed joint C-vine truncation we now turn to the issue of joint simplification. In doing so, we consider another evaluation criterion in addition to the criteria considered before: in each step of a simplification scenario we also compute the correlation matrices obtained from the respective stepwise procedure and compare it to the correlation matrices determined by the goodness-of-fit tests based on the empirical copula test and on the Rosenblatt transformation. Note that for the stepwise simplification procedures no bivariate independence tests are performed in order to reconstruct the correlation matrix most accurately which also takes into account small (conditional) correlations.

The four scenarios are again very similar to the scenarios in Section 10.1, in particular to the simplification scenarios in Section 10.1.2. Since $K_0 = 2$, higher order trees $T_3, ..., T_9$ are specified with Gaussian copulas with decreasing correlations: 0.25 in T_3, 0.20 in T_4 and T_5, 0.15 in T_6 and T_7, and 0.10 in T_8 and T_9, corresponding to Kendall's τ's smaller than 0.16. As in Section 10.2.1, the first two trees are chosen as follows (copula parameters are chosen according to Table 2.1):

---

[2]Degrees of freedom: tree $T_1 : (\nu_{1,2}, ..., \nu_{1,10}) = (3, 4, 5, 3, 4, 5, 3, 4, 5)$, tree $T_2 : (\nu_{2,3|1}, ..., \nu_{2,10|1}) = (9, 8, 7, 9, 8, 7, 9, 8)$. Kendall's τ's: tree $T_1 : (\tau_{1,2}, ..., \tau_{1,10}) = (0.8, 0.7, 0.6, 0.8, 0.7, 0.6, 0.8, 0.7, 0.6)$, tree $T_2 : (\tau_{2,3|1}, ..., \tau_{2,10|1}) = (0.4, 0.3, 0.2, 0.4, 0.3, 0.2, 0.4, 0.3)$.

- **Scenario 1:** Clayton copulas in T_1 and T_2 with Kendall's τ's 0.6 (T_1) and 0.3 (T_2).

- **Scenario 2:** t copulas in T_1 and T_2 with Kendall's τ's 0.6 (T_1) and 0.3 (T_2) and degrees of freedom 3 (T_1) and 7 (T_2).

- **Scenario 3:** Mixed copulas (t, Clayton, Gumbel) in T_1 and T_2 with Kendall's τ's 0.6 (T_1) and 0.3 (T_2) and degrees of freedom 3 (T_1) and 7 (T_2).

- **Scenario 4:** t copulas in T_1 and T_2 with mixed Kendall's τ's between 0.6 and 0.8 in T_1 and between 0.2 and 0.4 in T_2. Degrees of freedom are chosen between 3 and 5 in T_1 and between 7 and 9 in T_2.

Results for $R = 100$ repetitions are displayed in Tables 10.6 and 10.7, where "mean abs. R diff. 1/2" denotes the mean absolute difference of the correlation matrices obtained from the respective stepwise procedure and the joint procedure using the goodness-of-fit test based on the empirical copula test (1) or on the Rosenblatt transformation (2). The results can be summarized as follows:

(i) Joint simplification using the goodness-of-fit test based on the Rosenblatt transformation is always superior to the one based on the empirical copula process even if we expected this only for scenarios 2 and 4 according to Berg (2009) who found that the test based on Rosenblatt's transformation is suitable for testing the Gaussian copula against heavy tails induced by the t copula. Especially in scenarios 2 and 4, the test based on the empirical copula process fails completely as shown in particular by the mean KLIC's. This corresponds to the results of Berg (2009).

(ii) Even if joint simplification requires less assumptions and approximations in order to determine simplification levels, the performance of the stepwise methods based on the Vuong test is stronger. This is probably due to the fact that we also fit an additional tree T_{j+1} in step j and hence have additional information in order to decide whether or not the C-vine can be simplified. Moreover, copula goodness-of-fit testing in high dimensions has not yet been investigated very thoroughly in the literature.

(iii) The procedures based on AIC/BIC again perform rather poorly (cp. Section 10.1.2). The performance increases with increasing number of observations and the identified models are usually at least close to the true model, since they are simplified very late or not at all.

(iv) The correlation matrices are well approximated by the procedures based on the Vuong test in most cases. The accuracy increases with increasing number of observations. Since the models obtained by the test based on the empirical copula process are not quite good, the approximation is obviously also rather bad in most cases.

scenario	n	eval. crit.	full	TRUNCATION PROCEDURE							
				joint (emp. copula)	joint (Spearman's ρ)	Vuong	V.Akaike	V.Schwarz	AIC	BIC
1	500	truncation level	-	98.0	96.0	97.0	100.0	100.0	56.5	82.5
		mean KLIC	-11.10	-7.22	-7.51	-7.54	-7.07	-7.07	-10.65	-8.74
		Vuong test	100	100	100	100	100	100	100	100
		V. test (Akaike)	100	100	100	99	99	99	100	99
		V. test (Schwarz)	99	99	99	99	99	99	99	99
		τ-matrix diff.	1.997	1.982	2.184	2.081	2.162	2.224	2.185	2.237
		quartile diff.	0.106	0.103	0.112	0.114	0.110	0.116	0.117	0.117
	1000	truncation level	-	96.5	97.5	97.0	99.0	100.0	51.0	88.0
		mean KLIC	-9.26	-4.69	-4.69	-5.18	-4.79	-4.52	-8.83	-5.99
		Vuong test	100	100	100	100	100	100	100	100
		V. test (Akaike)	100	100	100	100	100	100	100	100
		V. test (Schwarz)	100	100	100	100	100	100	100	100
		τ-matrix diff.	1.356	1.469	1.480	1.482	1.530	1.451	1.462	1.437
		quartile diff.	0.082	0.082	0.078	0.083	0.084	0.080	0.089	0.085
2	500	truncation level	-	93.0	98.0	98.0	99.5	100.0	52.5	81.5
		mean KLIC	-18.66	-14.61	-14.34	-14.45	-14.25	-14.14	-18.08	-16.00
		Vuong test	100	100	100	100	100	100	100	100
		V. test (Akaike)	100	100	100	100	100	100	100	100
		V. test (Schwarz)	100	100	100	100	100	100	100	100
		τ-matrix diff.	1.985	2.008	1.990	1.976	1.968	1.972	2.012	1.979
		quartile diff.	0.105	0.099	0.100	0.097	0.105	0.102	0.098	0.106
	1000	truncation level	-	93.5	96.0	97.0	99.5	100.0	56.5	85.5
		mean KLIC	-15.32	-11.11	-10.93	-11.34	-10.86	-10.72	-14.77	-12.31
		Vuong test	99	99	99	99	99	99	99	99
		V. test (Akaike)	99	99	99	99	99	99	99	99
		V. test (Schwarz)	99	99	99	99	99	99	99	99
		τ-matrix diff.	1.405	1.369	1.431	1.434	1.368	1.478	1.400	1.421
		quartile diff.	0.065	0.064	0.064	0.065	0.063	0.064	0.067	0.065

Table 10.4: Simulation results of the joint truncation scenarios 1 and 2 described in Section 10.2.1. Each column corresponds to a different truncation procedure.

scenario	n	eval. crit.	full	TRUNCATION PROCEDURE							
				joint (emp. copula)	joint (Spearman's ρ)	Vuong	V.Akaike	V.Schwarz	AIC	BIC
3	500	truncation level	-	93.5	96.0	91.5	96.0	97.5	52.0	75.0
		mean KLIC	11.85	16.65	17.70	15.77	16.58	17.38	12.31	14.05
		Vuong test	92	92	92	92	92	92	92	92
		V. test (Akaike)	92	92	92	92	92	92	92	92
		V. test (Schwarz)	92	92	92	92	92	92	92	92
		τ-matrix diff.	1.867	1.942	2.155	2.057	2.007	1.993	2.015	1.936
		quartile diff.	0.105	0.112	0.105	0.116	0.113	0.108	0.102	0.106
	1000	truncation level	-	96.5	98.5	96.0	98.5	99.5	51.0	87.5
		mean KLIC	-8.74	-4.24	-3.97	-4.71	-4.19	-3.94	-8.21	-5.32
		Vuong test	99	99	99	99	99	99	99	99
		V. test (Akaike)	99	99	99	99	99	99	99	99
		V. test (Schwarz)	98	98	98	98	98	98	98	98
		τ-matrix diff.	1.441	1.449	1.374	1.412	1.379	1.433	1.412	1.384
		quartile diff.	0.082	0.074	0.076	0.077	0.076	0.083	0.079	0.076
4	500	truncation level	-	94.5	95.0	97.0	99.5	100.0	60.0	82.0
		mean KLIC	21.14	25.31	25.32	25.03	25.48	25.60	21.96	23.68
		Vuong test	70	49	49	51	49	48	66	58
		V. test (Akaike)	90	88	87	87	87	87	90	89
		V. test (Schwarz)	100	100	100	100	100	100	100	100
		τ-matrix diff.	1.722	1.655	1.717	1.684	1.746	1.786	1.553	1.636
		quartile diff.	0.095	0.096	0.101	0.104	0.090	0.101	0.097	0.098
	1000	truncation level	-	96.5	97.0	94.5	99.0	100.0	56.0	83.5
		mean KLIC	54.67	59.51	59.70	58.73	59.69	59.93	55.43	57.65
		Vuong test	8	3	3	3	3	3	8	4
		V. test (Akaike)	25	16	16	17	16	15	23	19
		V. test (Schwarz)	60	62	62	63	63	62	61	63
		τ-matrix diff.	1.308	1.179	1.235	1.212	1.164	1.211	1.162	1.255
		quartile diff.	0.067	0.073	0.069	0.067	0.069	0.068	0.068	0.073

Table 10.5: Simulation results of the joint truncation scenarios 3 and 4 described in Section 10.2.1. Each column corresponds to a different truncation procedure.

scenario	n	evaluation criterion	full	SIMPLIFICATION PROCEDURE						
				joint (emp. copula)	joint (Rosenblatt)	Vuong	V.Akaike	V.Schwarz	AIC	BIC
1	500	simplification level	-	43.0	55.0	67.5	71.0	68.0	0.0	2.0
		mean KLIC	7.46	37.23	101.02	35.06	37.77	45.99	7.60	8.31
		Vuong test	94	40	4	47	42	36	94	93
		V. test (Akaike)	90	31	3	46	42	35	90	89
		V. test (Schwarz)	81	20	1	43	40	32	81	81
		τ-matrix diff.	0.929	0.988	0.987	0.936	0.980	0.896	0.986	0.904
		quartile diff.	0.107	0.105	0.109	0.107	0.112	0.109	0.108	0.103
		mean abs. R diff. 1	-	-	-	0.022	0.022	0.022	0.028	0.027
		mean abs. R diff. 2	-	-	-	0.027	0.027	0.027	0.035	0.034
	1000	simplification level	-	9.0	43.5	51.0	58.5	66.0	0.5	2.5
		mean KLIC	46.24	65.43	133.25	84.35	88.86	97.75	46.52	47.86
		Vuong test	29	7	1	6	3	3	29	28
		V. test (Akaike)	20	3	1	4	2	2	18	16
		V. test (Schwarz)	9	0	1	2	1	1	9	9
		τ-matrix diff.	0.803	0.781	0.761	0.807	0.693	0.730	0.807	0.693
		quartile diff.	0.084	0.082	0.083	0.079	0.081	0.082	0.083	0.082
		mean abs. R diff. 1	-	-	-	0.013	0.013	0.013	0.018	0.017
		mean abs. R diff. 2	-	-	-	0.017	0.017	0.017	0.024	0.022
2	500	simplification level	-	5.0	63.0	74.0	74.5	73.0	5.0	22.0
		mean KLIC	-20.60	411.17	17.04	8.37	17.08	28.76	-20.02	-15.88
		Vuong test	99	1	73	81	74	68	99	98
		V. test (Akaike)	98	1	73	84	78	73	98	97
		V. test (Schwarz)	96	5	70	83	80	77	96	96
		τ-matrix diff.	0.968	1.297	0.989	0.925	0.934	0.908	0.945	0.922
		quartile diff.	0.098	0.110	0.092	0.102	0.093	0.090	0.090	0.093
		mean abs. R diff. 1	-	-	-	0.038	0.038	0.038	0.041	0.041
		mean abs. R diff. 2	-	-	-	0.023	0.024	0.023	0.030	0.029
	1000	simplification level	-	11.5	87.5	90.0	90.0	92.5	18.0	41.5
		mean KLIC	-24.14	688.05	12.44	-11.83	-10.26	-5.98	-22.20	-18.69
		Vuong test	93	1	78	90	90	88	93	93
		V. test (Akaike)	92	1	76	92	91	91	92	92
		V. test (Schwarz)	92	1	76	92	92	92	92	92
		τ-matrix diff.	0.624	0.944	0.704	0.667	0.608	0.660	0.674	0.599
		quartile diff.	0.064	0.069	0.072	0.073	0.072	0.072	0.068	0.069
		mean abs. R diff. 1	-	-	-	0.029	0.029	0.029	0.029	0.029
		mean abs. R diff. 2	-	-	-	0.016	0.016	0.016	0.019	0.017

Table 10.6: Simulation results of the joint simplification scenarios 1 and 2 described in Section 10.2. Each column corresponds to a different joint simplification procedure.

scenario	n	evaluation criterion	full	joint (emp. copula)	joint (Rosenblatt)	Vuong	V.Akaike	V.Schwarz	AIC	BIC
								SIMPLIFICATION PROCEDURE		
3	500	simplification level	-	3.0	58.5	86.0	87.5	88.0	5.0	21.5
		mean KLIC	15.06	1300.62	144.56	35.38	37.37	39.74	15.71	18.03
		Vuong test	85	0	13	74	73	72	85	85
		V. test (Akaike)	86	0	13	74	73	72	86	86
		V. test (Schwarz)	89	0	12	76	74	72	89	89
		τ-matrix diff.	0.962	1.782	1.061	0.971	0.965	0.983	0.937	0.952
		quartile diff.	0.105	0.117	0.112	0.111	0.115	0.106	0.112	0.111
		mean abs. R diff. 1	-	-	0.000	0.060	0.060	0.060	0.057	0.059
		mean abs. R diff. 2	-	-	-	0.030	0.030	0.030	0.034	0.033
	1000	simplification level	-	51.5	84.0	95.5	96.0	97.0	36.0	57.0
		mean KLIC	30.45	675.15	138.17	41.95	45.04	46.95	34.27	36.79
		Vuong test	87	15	57	87	87	87	87	87
		V. test (Akaike)	87	15	57	87	87	87	87	87
		V. test (Schwarz)	87	14	54	87	87	87	87	87
		τ-matrix diff.	0.641	0.996	0.747	0.599	0.624	0.654	0.623	0.611
		quartile diff.	0.078	0.072	0.073	0.071	0.076	0.082	0.075	0.073
		mean abs. R diff. 1	-	-	-	0.031	0.031	0.031	0.031	0.031
		mean abs. R diff. 2	-	-	-	0.020	0.020	0.020	0.020	0.020
4	500	simplification level	-	5.0	63.0	74.0	74.5	73.0	5.0	22.0
		mean KLIC	-20.60	411.17	17.04	8.37	17.08	28.76	-20.02	-15.88
		Vuong test	99	1	73	81	74	68	99	98
		V. test (Akaike)	98	1	73	84	78	73	98	97
		V. test (Schwarz)	96	5	70	83	80	77	96	96
		τ-matrix diff.	0.968	1.297	0.989	0.925	0.934	0.908	0.945	0.922
		quartile diff.	0.098	0.110	0.092	0.102	0.093	0.090	0.090	0.093
		mean abs. R diff. 1	-	-	-	0.038	0.038	0.038	0.041	0.041
		mean abs. R diff. 2	-	-	-	0.023	0.024	0.023	0.030	0.029
	1000	simplification level	-	11.5	87.5	90.0	90.0	92.5	18.0	41.5
		mean KLIC	-24.14	688.05	12.44	-11.83	-10.26	-5.98	-22.20	-18.69
		Vuong test	93	1	78	90	88	88	93	93
		V. test (Akaike)	92	1	76	92	91	91	92	92
		V. test (Schwarz)	92	1	76	92	92	92	92	92
		τ-matrix diff.	0.624	0.944	0.704	0.667	0.608	0.660	0.674	0.599
		quartile diff.	0.064	0.069	0.072	0.073	0.072	0.072	0.068	0.069
		mean abs. R diff. 1	-	-	-	0.029	0.029	0.029	0.029	0.029
		mean abs. R diff. 2	-	-	-	0.016	0.016	0.016	0.019	0.017

Table 10.7: Simulation results of the joint simplification scenarios 3 and 4 described in Section 10.2. Each column corresponds to a different joint simplification procedure.

10.3   Large scale simulation

In order to examine the performance of our truncation and simplification methods in larger dimensions, we consider a 19-dimensional example which is based on the R-vine models specified for the financial data set from Norway considered in Section 11.2. In particular, we investigate truncation by simulating from the pairwisely truncated R-vine at level 4 $\mathcal{M}_T(4)$, while simplification is considered using the pairwisely simplified 2 level R-vine $\mathcal{M}_S(2)$. The corresponding RVM's can be found in Appendix B.2, where the specification of $\mathcal{M}_T(4)$ is obtained from the corresponding matrices of the full R-vine model by simply replacing all entries of rows after the truncation level $K_0 = 4$ with zeros.

From the models we simulated $n \in \{500, 1000\}$ observations each and then applied our different procedures. The R-vine trees are constructed using Kendall's τ as weights and bivariate copulas are selected using the AIC. Evaluation criteria as in Section 10.1 are reported in Table 10.8, where we assign a score of 0.5 if

$$|K - K_0| = 1,$$

since the truncation/simplification level K_0 is either 2 or 4.

The main findings of $R = 100$ repetitions for both the truncation and the simplification scenarios are:

(i) Again the procedures based on the Vuong test are superior to those based on AIC/BIC. While the parsimonious BIC based method here performs better than the respective AIC based procedure, in particular in smaller sample sizes of $n = 500$, using the Vuong test with Schwarz correction gives weaker results compared to the Vuong test with and without Akaike correction.

(ii) The results improve with increasing sample size: truncation/simplification level identification accuracy, mean KLIC's as well as τ-matrix and quartile differences indicate that the models are more accurate for larger sample sizes.

(iii) While the performance of the AIC/BIC based method is fairly well for truncation, it performs rather poorly with respect to simplification which is probably due to the additional simplifying assumptions made here. This is underlined by increased τ-matrix differences, while these are smaller for truncated models, since the AIC/BIC based procedures tend to truncate too late and therefore model additional trees which increase the model accuracy.

(iv) The mean KLIC's and the Vuong tests indicate that all model are quite good compared to the true model. This is however due to the simulation error and the fact that the specified models can adjust to this error, since they are re-estimated in each repetition, while the true model is fixed in advance.

			TRUNCATION AND SIMPLIFICATION PROCEDURES					
scenario	n	evaluation criterion	full	Vuong	V.Akaike	V.Schwarz	AIC	BIC
truncation	500	truncation level	0.0	86.0	81.5	41.0	46.0	72.0
		mean KLIC	-39.24	-19.57	-14.67	10.53	-25.68	-21.24
		Vuong test	100	100	99	72	100	100
		V. test (Akaike)	100	100	100	98	100	100
		V. test (Schwarz)	100	100	100	100	100	100
		τ-matrix diff.	5.135	5.370	5.465	5.680	5.263	5.394
		quartile diff.	0.071	0.073	0.073	0.070	0.072	0.072
	1000	truncation level	0.0	86.0	92.5	91.5	45.5	77.5
		mean KLIC	-45.18	-21.97	-20.81	-16.03	-27.83	-23.18
		Vuong test	100	100	100	98	100	100
		V. test (Akaike)	100	100	100	98	100	100
		V. test (Schwarz)	100	100	100	100	100	100
		τ-matrix diff.	3.532	3.761	3.814	3.827	3.751	3.731
		quartile diff.	0.059	0.058	0.060	0.062	0.054	0.059
simplification	500	simplification level	0.0	89.0	84.0	65.5	5.5	12.0
		mean KLIC	-36.15	-24.71	-23.22	-14.64	-32.35	-31.37
		Vuong test	100	100	100	99	100	100
		V. test (Akaike)	100	100	100	100	100	100
		V. test (Schwarz)	100	100	100	100	100	100
		τ-matrix diff.	5.172	5.074	5.180	5.169	5.291	5.239
		quartile diff.	0.071	0.066	0.067	0.071	0.065	0.064
	1000	simplification level	0.0	94.0	94.5	84.0	4.0	12.5
		mean KLIC	-46.98	-36.61	-35.92	-25.93	-42.65	-41.33
		Vuong test	100	99	99	99	100	100
		V. test (Akaike)	100	100	100	100	100	100
		V. test (Schwarz)	100	100	100	100	100	100
		τ-matrix diff.	3.569	3.530	3.535	3.529	3.591	3.598
		quartile diff.	0.063	0.062	0.063	0.063	0.065	0.065

Table 10.8: Simulation results of the large scale truncation and simplification scenarios described in Section 10.3. Each column corresponds to a different pairwise truncation and simplification procedure, respectively.

10.4 Summary

Considering the above simulation studies, we arrive at the conclusion that...

(i) the procedure based on the Vuong test with or without correction for the number of parameters should be used in most cases and identifies the correlation matrix of a jointly simplifying Gaussian copula quite accurately,

(ii) the procedure based on AIC/BIC can be regarded as a "quick and dirty" alternative, which often fails to identify the correct simplification or truncation level, but is fast and usually at least close to the true model,

(iii) more parsimonious models can be selected using the methods based on the Vuong test with Schwarz correction or on the BIC, i.e., under strict time and/or resource constraints these procedures should be used as they tend to truncate/simplify models earlier,

(iv) both multivariate independence tests under consideration for joint truncation perform well, where the test based on Spearman's ρ is slightly more accurate in identifying the true truncation level and also computationally faster, and

(v) using the goodness-of-fit test based on Rosenblatt's transformation for joint C-vine simplification is superior to the test based on the empirical copula process.

Reasons for the rather poor results of the AIC/BIC based procedure with respect to identification of truncation and simplification are, one the one hand, the additional assumptions and approximations, which are made when applying the procedure, and, on the other hand, the different selection threshold: as the Vuong test attaches a significance level to its decision, we can also choose the "small" model when the "full" model is slightly, but not significantly, better $(-\Phi^{-1}(1 - \frac{\alpha}{2}) < \nu < 0)$. If however the AIC or BIC are used for selection, we only select the "small" model if its respective criterion is smaller, i.e., this corresponds to a Vuong test which neglects the asymptotics and chooses a model according to the test statistic being positive or negative.

Remark: Computation time

The term "quick and dirty" for the procedure based on AIC/BIC is in fact warranted. In a 52-dimensional example (cp. the data set considered in Section 11.3), truncation at the same level was identified between 40 and 44% faster than using the procedure based on the Vuong test. Regarding simplification, the AIC/BIC based method simplified at a much higher level than the alternative procedure ("dirty"), but nevertheless it was 70 to 80% faster ("quick").

Chapter 11

Applications

After having described and analyzed different simplification and truncation procedures, we will now investigate the benefit of these procedures in three different applications. We begin with a nine-dimensional data set of exchange rates to the US Dollar which has already been examined in Schepsmeier (2010) and Dißmann (2010). In this application, we are particularly interested in joint C-vine truncation, since the observations of this data set have already been shown to be adequately modeled with a C-vine and the dependencies in higher order trees to be rather small.

In a second application, we analyze a 19-dimensional data set of Norwegian and international financial variables. Here, simplification as well as truncation are important in order to minimize the computational demand of specifying an appropriate R-vine. Hence we will investigate the different results of simplification, truncation and hierarchical specification.

Thirdly, we review the work of Heinen and Valdesogo (2009) and investigate their assumptions. In order to construct an extended Capital Asset Pricing Model, they simplify a C-vine at a specific level and make crucial independence assumptions. We will check these assumptions in an application to 46 stocks of the Euro Stoxx 50 and the respective national leading stock indices. Moreover, we develop an alternative model based on the more flexible structure of R-vines, the so-called *regular vine market sector model*, and compare all models to R- and C-vine specifications of the data which do not require any restricting assumptions.

In all models, we work with transformed residuals of marginal time series models. Pair copulas will be selected using AIC, since it turned out to be the best bivariate copula selection criterion (see Section 5.4). Furthermore, it is computationally faster than goodness-of-fit testing. The models are sequentially fitted first before performing full maximum likelihood estimation using the sequential estimates as starting values.

11.1   Exchange rates

Schepsmeier (2010) analyzed a data set of nine exchange rates to the US Dollar with 1007 observations from 7/22/2005 to 7/17/2009 (see Table 11.1 and Figure 11.1). ARMA(1,1)-GARCH(1,1)-models were fitted to each time series and standardized residuals transformed by their respective empirical cdf's in order to obtain pseudo-observations that are
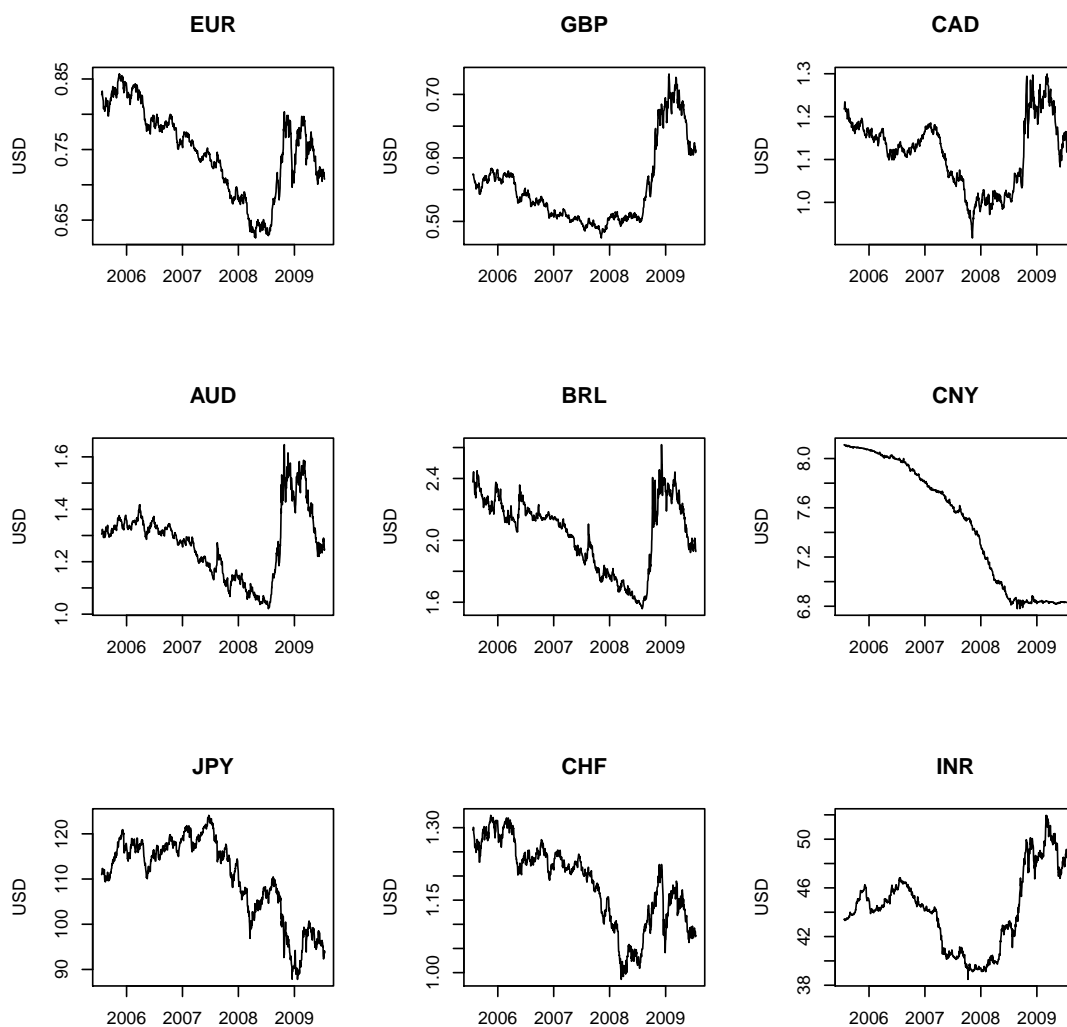
Figure 11.1: Time series of exchange rates to the US Dollar.

ID	code	currency
V1	EUR	Euro
V2	GBP	British Pound
V3	CAD	Canadian Dollar
V4	AUD	Australian Dollar
V5	BRL	Brazilian Real
V6	CNY	Chinese Yuan
V7	JPY	Japanese Yen
V8	CHF	Swiss Franc
V9	INR	Indian Rupee

Table 11.1: Exchange rates to the US Dollar.

approximately uniformly distributed. For more details on the transformations and on the data set itself see Schepsmeier (2010).

Dißmann (2010) also analyzed this data set in order to investigate whether an R-vine specification instead of a C-vine as in Schepsmeier (2010) improves the model. Using the Vuong test, he concluded that an R-vine structure is not significantly superior to a C-vine. Moreover, he found that an R-vine with t copulas for every pair is inferior to mixed copula modeling where the bivariate copulas were determined using goodness-of-fit tests.

We take these results as starting point for a further analysis. Since a C-vine is appropriate to model the observations, we are interested in determining whether this C-vine can be jointly simplified or even truncated. Hence, we apply the procedures discussed in Chapters 6, 7 and 8 and compare the results in Table 11.2, where the first part corresponds to modeling without testing for bivariate independence when choosing pair copula families ("complete models"; cp. Section 7.4), which is done mainly for illustration of the benefit of simplification and truncation. The second part of Table 11.2 then shows the results for models obtained using such a bivariate independence test ("reduced models"). As weights for tree construction we use Kendall's τ. Log likelihoods, number of parameters as well as AIC and BIC of all models are reported. Furthermore, we performed Vuong tests between the respective full model and the simplified/truncated models, since we are interested in the null hypothesis

$$H_0 : \text{full model and simplified/truncated model are equivalent.} \qquad (11.1)$$

If both models are equivalently good (or the simplified/truncated model is even preferred to the full model), we can conclude that the simplified/truncated model is an adequate C-vine specification. Hence, Table 11.2 shows the test statistics of the Vuong tests with Akaike and Schwarz correction as well as without correction for the number of parameters (the full model is chosen as "model 1" and hence small values indicate that the simplified/truncated model is close to the full model or even better, if negative). For the reduced models truncated at level 6, a Vuong test cannot be performed, since they are equal to the full model, i.e., no real truncation is determined because the reduced full model has independence copulas in its last trees T_7 and T_8 (also cp. Schepsmeier (2010)). This corresponds to the results of the complete models: models truncated at level 6 are significantly superior to the full model with fully specified trees T_7 and T_8.

Note that we examine all procedures discussed in the previous chapters, but the results of the AIC/BIC based methods have to be considered carefully, since it turned out that these procedures are less accurate than the methods based on the Vuong test (cp. Section 10.2). In contrast to Dißmann (2010) we use the AIC to select appropriate pair copulas, so that our resulting full C-vine is slightly different to his model.

For comparison, models with t copulas for every pair are also fitted, where a t copula is set to a Gaussian copula, if the degrees of freedom are estimated as larger than 30. Especially the results from the parsimonious criterion using the Schwarz correction for the number of parameters indicate that these models are no alternatives to appropriately simplified and truncated models with accurately specified first trees. Models with only t copulas simply require too many parameters and cannot account for asymmetric dependence.

proc. type	procedure	trunc./ simpl. level	log likelihood	number of param.	AIC	BIC	Vuong stat.	V. stat. (Akaike corr.)	V. stat. (Schwarz corr.)
COMPLETE (no bivariate indep. tests)	full model	-	2243.73	49	-4389.47	-4148.65	-	-	-
trunc.	t copulas	-	2240.73	59	-4363.46	-4073.49	0.35*	1.50*	4.34
	joint (emp. cop.)	4	2212.40	38	-4348.80	-4162.04	3.65	2.37	-0.78*
	joint (Spearman)	2	1987.79	24	-3927.59	-3809.64	9.14	8.25	6.05
	Vuong	6	2243.43	46	-4394.86	-4168.79	0.39*	-3.46*	-12.90*
	V.Akaike	5	2232.42	43	-4378.84	-4167.51	2.40	1.13*	-2.00*
	V.Schwarz	3	2181.97	32	-4299.93	-4142.66	5.12	3.71	0.25*
	AIC/BIC	6	2243.43	46	-4394.86	-4168.79	0.39*	-3.46*	-12.90*
simpl.	joint (emp. cop.)	0	2106.89	36	-4141.77	-3964.84	5.14	4.65	3.45
	joint (Rosenblatt)	2	2207.44	45	-4324.88	-4103.71	4.19	3.73	2.59
	Vuong	1	2204.50	42	-4324.99	-4118.57	3.36	2.76	1.29*
	V.Akaike	1	2204.50	42	-4324.99	-4118.57	3.36	2.76	1.29*
	V.Schwarz	1	2204.50	42	-4324.99	-4118.57	3.36	2.76	1.29*
	AIC/BIC	-	2243.73	49	-4389.47	-4148.65	-	-	-
REDUCED (bivariate indep. tests)	full model	-	2220.51	34	-4373.02	-4205.92	-	-	-
trunc.	t copulas	-	2220.21	40	-4360.42	-4163.83	0.04*	0.80*	2.66
	joint (emp. cop.)	4	2199.87	31	-4337.74	-4185.38	3.22	2.75	1.60*
	joint (Spearman)	2	1985.03	23	-3924.06	-3811.03	8.96	8.54	7.51
	Vuong	6	2220.51	34	-4373.02	-4205.92	-	-	-
	V.Akaike	4	2199.87	31	-4337.74	-4185.38	3.22	2.75	1.60*
	V.Schwarz	4	2199.87	31	-4337.74	-4185.38	3.22	2.75	1.60*
	AIC/BIC	6	2220.51	34	-4373.02	-4205.92	-	-	-
simpl.	joint (emp. cop.)	0	2106.89	36	-4141.77	-3964.84	4.39	4.47	4.66
	joint (Rosenblatt)	2	2204.35	44	-4320.70	-4104.45	2.36	3.82	7.41
	Vuong	1	2197.43	30	-4334.86	-4187.42	2.51	2.07	1.00*
	V.Akaike	1	2197.43	30	-4334.86	-4187.42	2.51	2.07	1.00*
	V.Schwarz	1	2197.43	30	-4334.86	-4187.42	2.51	2.07	1.00*
	AIC/BIC	6	2220.51	34	-4373.02	-4205.92	-	-	-

Table 11.2: C-vine specifications of the exchange rates data set obtained from different procedures (full maximum likelihood estimation). Test statistics indicated by "*" imply that the null hypothesis (11.1) cannot be rejected at the 5% level or that the simplified/truncated model is chosen over the full model.

order	code	currency
1st	EUR	Euro
2nd	AUD	Australian Dollar
3rd	CHF	Swiss Franc
4th	BRL	Brazilian Real
5th	INR	Indian Rupee
6th	CAD	Canadian Dollar
7th	GBP	British Pound
$8^{th}/9^{th}$	CNY/JPY	Chinese Yuan/Japanese Yen

Table 11.3: Order of the conditioning set of the full C-vine model (reduced model selection setting) for the exchange rates data set.

Apparently, simplification is already feasible after the first tree T_1 as consistently determined by the simplification methods based on the Vuong test and confirmed by Vuong tests, i.e., the first tree captures most asymmetric (tail) dependencies and only symmetric dependencies without significant strong joint tail behavior remain in the higher order trees $T_2, ..., T_8$. Alternatively, the goodness-of-fit test based on the Rosenblatt transformation determines a simplification level of 2, which confirms the above results, but also models the second tree T_2 in order to get a more accurate model. The rather weak results with respect to AIC/BIC and Vuong tests with correction are due to the fact that joint simplification with a Gaussian copula involves the estimation of all entries of the correlation matrix, even if they are small. Hence, more parameters are estimated compared to models with independence copulas for some variable pairs, but compared to the full model with completely specified trees still somewhat less parameters are estimated. The consideration of the log likelihood therefore shows that the jointly simplified 2 level C-vine is a quite good model. However, the goodness-of-fit test based on the empirical copula process determines an entirely inadequate model: all criteria highlight that a simplification at level 0 is inappropriate, i.e., the data is clearly not jointly normally distributed and the procedure completely fails here.

The truncation results give a different perspective: a truncation level of 3 or 4 is determined which corresponds to the fact that conditioned on two or three variables, respectively, all significant dependencies are explained. A truncation level of 6 is not adequate, since there is apparently no dependence left in the last two trees as already mentioned above. On the other hand, truncation after tree T_2 as determined by the truncation procedure based on Spearman's ρ is also inappropriate as indicated by AIC and BIC as well as the Vuong test statistics. This definitely results in too parsimonious models here.

The order of the conditioning set of the C-vine is identified by the full model in the reduced model selection setting as indicated in Table 11.3, where the order of the last two variables is arbitrary (cp. the corresponding RVM in Appendix B.1). Note that the truncation procedures based on the Vuong test with Akaike and Schwarz correction identify the last three variables in a slightly different order: the seventh variable in the conditioning set is the Japanese Yen instead of the British Pound.

	full	$\mathcal{M}_T(4)$	$\mathcal{M}_S(1)$	$\mathcal{M}_J(2)$
τ-matrix diff.	0.02029	0.02194	0.02063	0.02057
quartile diff.	0.08779	0.09218	0.09582	0.08704

Table 11.4: Mean absolute difference of the empirical Kendall's τ's of all variable pairs ("τ-matrix diff.") and sum of absolute differences between empirical quartiles of S_i^K and S_i ("quartile diff.") for C-vine models of the exchange rates data set.

11.1.1   Model evaluation

Given the results discussed above, we are now interested if the simplified/truncated models adequately reproduce the data characteristics or if they are too simple. Therefore, we apply model evaluation criteria as discussed in Chapter 9. In particular, we will concentrate on the truncated C-vine at level 4, $\mathcal{M}_T(4)$, and the pairwisely simplified 1 level C-vine, $\mathcal{M}_S(1)$, as well as the jointly simplified 2 level C-vine, $\mathcal{M}_J(2)$, since these models turned out to be the most adequate models in terms of model selection criteria. We only consider the models in the reduced model specification setting, since we are interested in the simplest models possible. Chosen copula types and parameter estimates of the models are given in Appendix B.1.

In a first step we investigate the dependencies of the first C-vine tree, i.e., the relationships of the Euro exchange rate to all other exchange rates in terms of their transformed residuals. According to Section 9.1, we consider scatter plots with standard normal margins, empirical copula distributions as well as copula Q-Q plots for the three simplified/truncated models as well as for the fully specified model. The results are shown in Figure 11.2 for the truncated model $\mathcal{M}_T(4)$ as well as in Appendix B.1 for the other three models.

Judging from the analytical plots, all models accurately reproduce the dependencies of the Euro exchange rates to all other exchange rates, since there are no systematic deviations. Moreover, existing deviations are partly due to the simulation error.

For a further analysis and comparison of the models, we consider summarizing model evaluation criteria (see Section 9.2), namely the mean absolute difference of the empirical Kendall's τ's of all variable pairs as well as the sum of absolute differences between empirical quartiles of S_i^K and S_i as defined in (9.1). The mean results of $R = 100$ repeated simulations are shown in Table 11.4 (cp. (10.1)).

According to both criteria, the jointly simplified model $\mathcal{M}_J(2)$ is closer to the true model than the pairwisely simplified model $\mathcal{M}_S(1)$. Moreover, the first model is also quite close to the fully specified model. The truncated model $\mathcal{M}_T(4)$, on the other hand, exhibits slightly weaker results which is certainly due to the neglected dependencies in the higher order trees. Nevertheless, given the values of the evaluation criteria for the full model, all three simplified/truncated models are quite good approximations to the full model. This underlines the results from the model selection criteria as shown in Table 11.2. Both simplified models provide an adequate fit of the exchange rates data set, where the jointly simplified level 2 C-vine model $\mathcal{M}_J(2)$ is naturally slightly more accurate, since one more tree is fully specified.
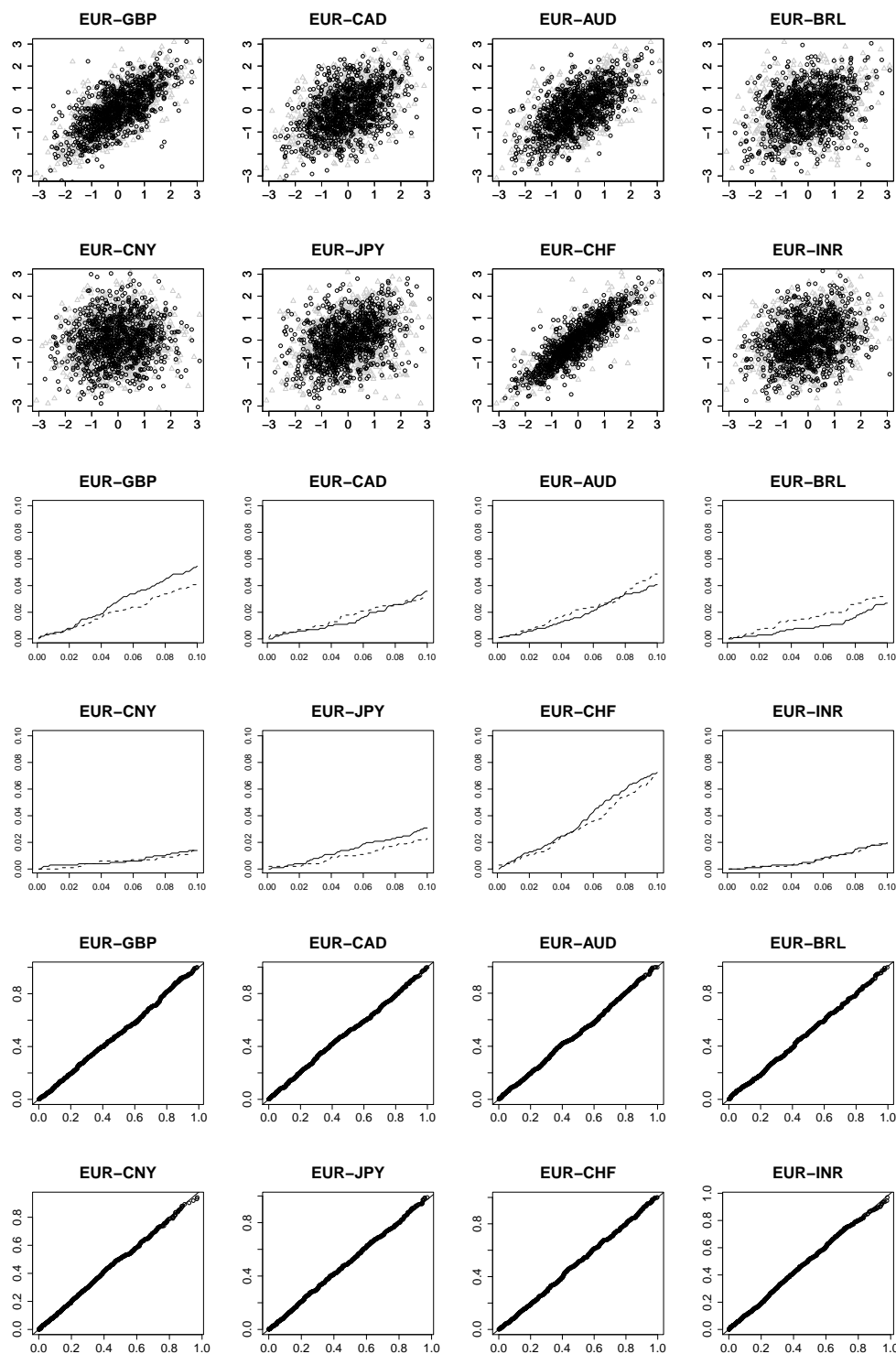
Figure 11.2: Truncated model $\mathcal{M}_T(4)$ for the exchange rates data set: scatter plots with standard normal margins, empirical copula distribution functions (lower tail) and copula Q-Q plots for observed (gray triangles, solid lines) versus simulated data (black circles, dashed lines).

11.1.2 Conclusion

Based on the truncation results we can state that by structurally conditioning on the Euro, Australian Dollar and Swiss Franc exchange rates to the US Dollar all significant dependencies among transformed residuals of exchange rates can be explained. These results are refined by simplification: the relationships to the Euro exchange rate are the most important to construct accurate models for explaining interdependencies among exchange rates. All other relationships conditioned on the Euro exchange rates can be modeled simply by Gaussian copulas. However, to get somewhat more accurate results, the relationships to the Australian Dollar exchange rate (conditioned on the Euro exchange rate) should also be taken into account separately. For a more detailed economical interpretation we refer to Schepsmeier (2010).

11.2 Financial data from Norway

The data set considered in this application consists of 19 Norwegian and international financial variables, whose descriptions can be found in Table 11.5. The observed time period is from 3/25/2003 to 3/26/2008 resulting in 1107 daily observations. The corresponding time series are shown in Figure 11.3 and descriptive statistics of the log returns in Table 11.6.

ID	variable name	description
V1	FINX	Norwegian Financial Index
V2	USDNOK	USD-NOK exchange rate
V3	EURNOK	EURO-NOK exchange rate
V4	Yen	YEN-NOK exchange rate
V5	GBP	GBP-NOK exchange rate
V7	NIBOR3M	3-month Norwegian Inter Bank Offered Rate
V8	NIBOR5Y	Norwegian 5-year Swap Rate
V9	EUR3M	3-month Euro Interbank Offered Rate
V10	EUR5Y	5-year German Government Rate
V11	USD3M	3-month US Libor Rate
V12	USD5Y	5-year US Government Rate
V13	NoOblig	Norwegian bond index (BRIX)
V14	IntOblig	Citigroup World Government Bond Index (WGBI)
V15	Anleggsoblig	Norwegian 6-year Swap Rate
V16	Pengemarked	ST2X - Government Bond Index (fix modified duration of 0.5 years)
V17	IntAksjer	Morgan Stanley World Index (MSCI)
V18	NoAksjer	OSEBX - Oslo Stock Exchange main index
V19	Eiendom	Oslo Stock Exchange Real Estate Index
V20	Hedgefond	S&P Hedge Fund Index

Table 11.5: Financial data set from Norway.
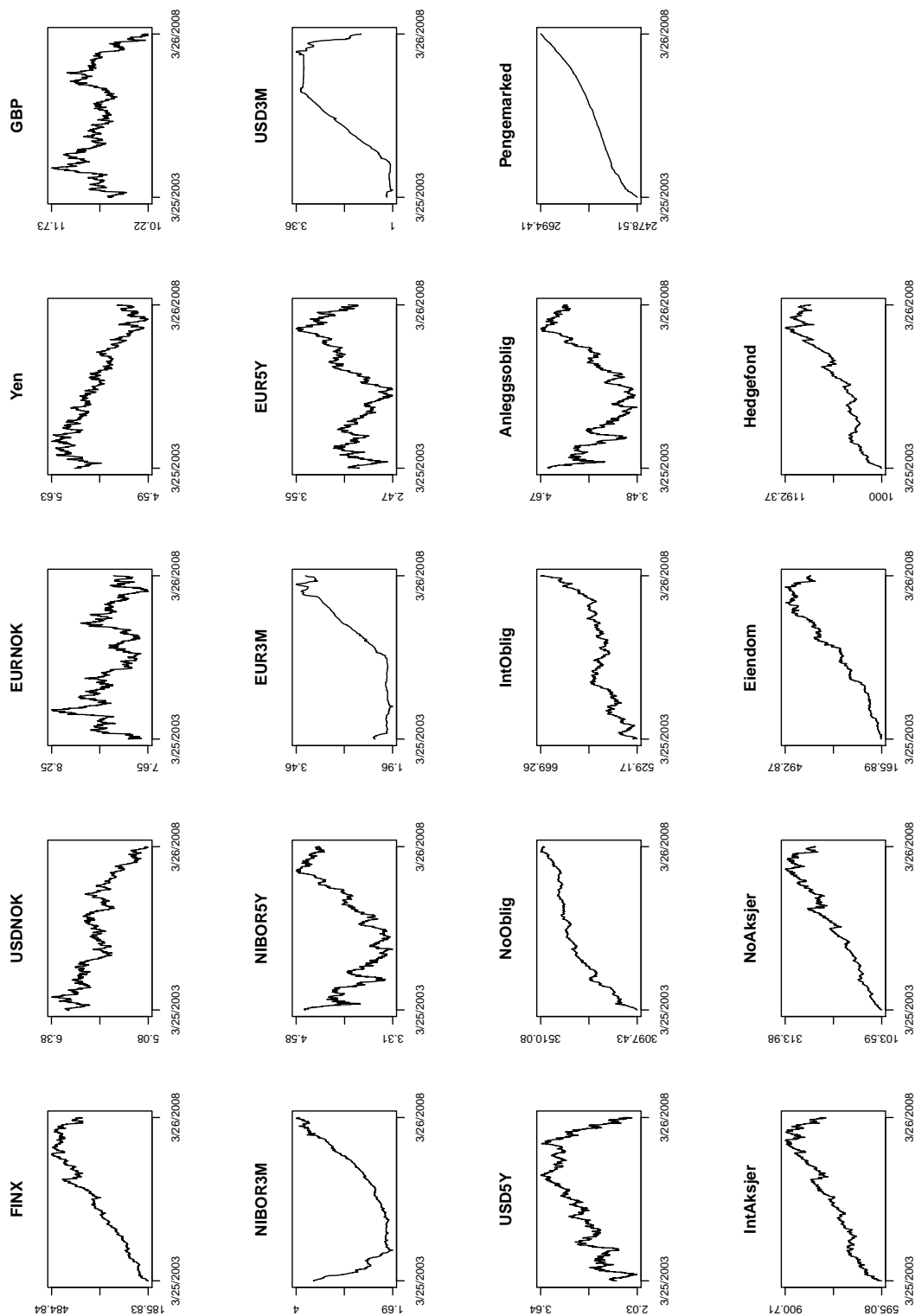
Figure 11.3: Time series of the financial data set from Norway.

	V1	V2	V3	V4	V5	V7	V8	V9	V10	V11
mean	0.001	-0.000	0.000	-0.000	-0.000	0.001	-0.000	0.001	0.000	0.001
std. dev.	0.012	0.007	0.004	0.008	0.005	0.010	0.011	0.004	0.013	0.008
skewness	-0.039	0.183	0.280	0.405	-0.028	-1.568	0.033	0.820	0.324	-2.464
kurotsis	1.709	0.614	1.277	1.364	0.896	18.618	6.891	21.818	1.902	64.391

	V12	V13	V14	V15	V16	V17	V18	V19	V20
mean	-0.000	0.000	0.000	-0.000	0.000	0.000	0.001	0.001	0.000
std. dev.	0.020	0.001	0.005	0.011	0.000	0.007	0.013	0.012	0.003
skewness	-0.013	0.433	-0.029	-0.032	1.984	-0.360	-0.413	0.602	-1.087
kurtosis	4.638	7.089	0.839	6.569	11.151	2.038	2.494	4.727	3.901

Table 11.6: Descriptive statistics of the variables in the financial data set from Norway.

Before analyzing the dependence in the data set, we have to select appropriate models for the univariate margins in order to obtain i.i.d. copula data by transforming standardized residuals by their respective empirical cdf's. Hence, we have to find adequate time series models as presented in Section 2.5. To do this, we arranged the variables into groups with similar characteristics (exchange rates, stock indices, short term interest rates,...) and then fitted similar models to each class of variables, where we preliminarily decided to use the flexible standardized NIG distribution for the errors if possible. The results are shown in Table 11.7 alongside with p-values of Kolmogorov-Smirnov tests (4.2) for the respective distributions of the errors.

While we were able to select the standardized NIG distribution for the errors of 17 time series, we selected standardized Student-t and skewed Student-t for the hedge fund and real estate indices (V19, V20), respectively (cp. Section 2.5.3). As expected, the stock indices (FINX (V1), IntAksjer (V17), NoAksjer (V18)) exhibit a slightly positive mean corresponding to positive expected returns of stocks. These indices as well as the exchange rates can be modeled by GARCH(1,1)-models. Bond indices, on the other hand, require an additional autoregressive term (between 0.010 and 0.081) and no significant GARCH terms have been found for the ST2X Government Bond Index (V16). Similarly, the short term interest rates are modeled by ARMA(1,1)-models only, which exhibit rather high coefficients. Note, that the Kolmogorov-Smirnov tests are clearly rejected for the variables NIBOR3M (V7) and USD3M (V11). Nevertheless, we will use this model, since within the given range of examined models, we did not find a better model and hence we chose the same model as for EUR3M (V9). In contrast to these short term rates, long term rates can be adequately modeled by AR(1)-GARCH(1,1)-models.

Note that we use a non-parametric two step estimation approach here, since the marginal time series are preliminarily estimated and the standardized residuals then transformed non-parametrically to approximately uniform data before the vine copula model is specified. This is in contrast to the Bayesian joint estimation approach taken by Hofmann and Czado (2010). Given the dimension of the data set, the two step approach is however reasonable.

	model	error distr.	$\hat{\mu}$	$\hat{\phi}_1$	$\hat{\theta}_1$	$\hat{\omega}$	$\hat{\alpha}_1$	$\hat{\beta}_1$	skew	shape	KS
V1	GARCH(1,1)	Std. NIG	0.001	-	-	0.000	0.145	0.749	0.046	3.311	0.724
V17	GARCH(1,1)	Std. NIG	0.001	-	-	0.000	0.069	0.915	-0.216	3.614	0.300
V18	GARCH(1,1)	Std. NIG	0.002	-	-	0.000	0.108	0.847	-0.329	4.108	0.680
V2	GARCH(1,1)	Std. NIG	-0.000	-	-	0.000	0.018	0.000	0.142	5.000	0.989
V3	GARCH(1,1)	Std. NIG	-0.000	-	-	0.000	0.044	0.908	0.076	2.376	0.914
V4	GARCH(1,1)	Std. NIG	-0.000	-	-	0.000	0.064	0.917	0.269	3.680	0.921
V5	GARCH(1,1)	Std. NIG	-0.000	-	-	0.000	0.029	0.954	0.006	5.007	0.855
V7	ARMA(1,1)	Std. NIG	-0.001	-0.981	0.998	-	-	-	-0.056	0.449	0.001
V9	ARMA(1,1)	Std. NIG	0.001	-0.517	0.806	-	-	-	0.161	0.077	0.380
V11	ARMA(1,1)	Std. NIG	0.001	-0.745	0.923	-	-	-	0.040	0.019	0.000
V8	AR(1)-GARCH(1,1)	Std. NIG	0.000	0.060	-	0.000	0.055	0.936	-0.022	1.632	0.204
V10	AR(1)-GARCH(1,1)	Std. NIG	0.000	0.010	-	0.000	0.037	0.961	0.062	5.376	0.643
V12	AR(1)-GARCH(1,1)	Std. NIG	0.000	-0.066	-	0.000	0.048	0.954	-0.103	2.954	0.885
V15	AR(1)-GARCH(1,1)	Std. NIG	0.000	0.051	-	0.000	0.054	0.935	-0.019	1.831	0.139
V13	AR(1)-GARCH(1,1)	Std. NIG	0.000	0.081	-	0.000	0.045	0.935	-0.013	1.000	0.504
V14	AR(1)-GARCH(1,1)	Std. NIG	0.000	0.010	-	0.000	0.020	0.975	0.072	3.047	0.355
V16	AR(1)	Std. NIG	0.000	0.079	-	-	-	-	0.465	0.423	0.810
V19	ARMA(1,1)-GARCH(1,1)	Std. skew-t	0.001	0.478	-0.534	0.000	0.199	0.455	1.064	2.729	0.070
V20	GARCH(1,1)	Std. t	0.000	-	-	0.000	0.117	0.864	1.000	6.579	0.384

Table 11.7: Chosen time series models for the financial data set from Norway. "skew" and "shape" indicate the skewness and shape parameters of the respective error distribution. "KS" stands for the p-value of the respective Kolmogorov-Smirnov test.

11.2.1 Simplification and truncation

In order to investigate whether simplification or even truncation of the R-vine specification corresponding to this data set are possible, we apply our different procedures and compare them in Table 11.8. For tree construction we consider Kendall's τ and exceedance Kendall's τ with $\delta_1 = \delta_2 = 0.2$ for lower and $\delta_1 = \delta_2 = 0.8$ for upper dependence (maximum of lower and upper dependence; see Section 3.1) as weights, since we are particularly interested in joint tail behavior. As before, we report log likelihoods, number of parameters, AIC, BIC as well as the test statistics of Vuong tests with respect to the null hypothesis (11.1) in order to investigate whether the simplified/truncated model is an adequate R-vine specification. When selecting pair copulas, bivariate independence tests are performed.

We also specify models with t copulas for every pair as in Section 11.1. However, the results show that these models are rather not an alternative to the respective full models with mixed copulas, since the Vuong tests with Schwarz correction for the number of parameters reject these models which are less parsimonious than the full models. Although t copulas cannot account for asymmetric dependence, the log likelihood of the model using exceedance dependence as weights is higher than the corresponding full model. This is however achieved at the expense of an increased number of parameters and hence lower AIC and BIC values as for the model constructed using Kendall's τ als weights.

The results of hierarchical model specification as presented in Algorithm 10 can be deduced from Table 11.8 by comparing simplification and truncation levels of a particular procedure. The procedures based on the Vuong test always detected simplification before truncation, while the AIC/BIC based procedures for models built with Kendall's τ as weights detected the same simplification and truncation levels, i.e., truncation is determined by hierarchical specification.

We first discuss the results of the models which were constructed with Kendall's τ as weights. As seen in the simulation studies in Chapter 10, the procedures based on the Vuong test are more reliable to detect simplification and truncation than those based on AIC/BIC. Hence, we concentrate on the results of the first mentioned methods. However, note that the AIC/BIC based methods here identify the same truncation level as the procedures based on the Vuong test with and without Akaike correction (no additional assumptions are made here), while the simplification results do not indicate a significant improvement in facilitating model building (simplification at level 6, which is also determined as truncation level!). Moreover, the parsimonious Vuong test using the Schwarz correction indicates that the obtained R-vines are equivalent to the full model even if we simplify at level 2! This is however not true for truncation. Apparently, there are still considerable dependencies after tree T_6 (and of course after the truncation level 4, which is determined by the procedure based on the Vuong test with parsimonious Schwarz correction). Nevertheless, this can be seen as an approximation to the full model when looking for rather parsimonious models. As indicated by the procedure based on the Vuong test with Schwarz correction, a truncation level of 4 might even be an alternative, since the BIC and Vuong test statistics are similar to the truncated R-vine at level 6.

R-vine construction using exceedance dependence as weights allows an alternative perspective on the data: all models include more parameters and are truncated later than the respective models built with Kendall's τ as weights, i.e., less overall dependence has

weights	proc. type	procedure	trunc./ simpl. level	log likeli-hood	number of param.	AIC	BIC	Vuong stat.	V. stat. (Akaike corr.)	V. stat. (Schwarz corr.)
Kendall's τ		full model	-	6390.75	92	-12597.50	-12130.22	-	-	-
		t copulas	-	6378.33	104	-12548.65	-12020.42	0.82*	1.61*	3.61
	trunc.	Vuong	6	6274.47	77	-12394.93	-12003.83	7.25	6.31	3.94
		V.Akaike	6	6274.47	77	-12394.93	-12003.83	7.25	6.31	3.94
		V.Schwarz	4	6234.05	68	-12332.10	-11986.72	7.97	6.75	3.65
		AIC	6	6274.47	77	-12394.93	-12003.83	7.25	6.31	3.94
		BIC	6	6274.47	77	-12394.93	-12003.83	7.25	6.31	3.94
	simpl.	Vuong	2	6350.09	84	-12532.17	-12105.52	3.19	2.57	0.97*
		V.Akaike	2	6350.09	84	-12532.17	-12105.52	3.19	2.57	0.97*
		V.Schwarz	2	6350.09	84	-12532.17	-12105.52	3.19	2.57	0.97*
		AIC	6	6373.80	88	-12571.60	-12124.63	2.46	1.88*	0.41*
		BIC	6	6373.80	88	-12571.60	-12124.63	2.46	1.88*	0.41*
exceedance Kendall's τ (max.)		full model	-	6397.87	113	-12569.74	-12061.27	-	-	-
		t copulas	-	6412.22	129	-12566.43	-11911.22	-0.75*	0.09*	2.21
	trunc.	Vuong	13	6041.28	109	-11864.56	-11310.93	11.67	11.54	11.21
		V.Akaike	13	6041.28	109	-11864.56	-11310.93	11.67	11.54	11.21
		V.Schwarz	9	5891.65	97	-11589.30	-11096.62	13.22	12.81	11.74
		AIC	15	6046.79	111	-11871.59	-11307.80	11.51	11.44	11.28
		BIC	13	6041.28	109	-11864.56	-11310.93	11.67	11.54	11.21
	simpl.	Vuong	1	6237.48	93	-12288.96	-11816.60	6.38	5.59	3.57
		V.Akaike	1	6237.48	93	-12288.96	-11816.60	6.38	5.59	3.57
		V.Schwarz	1	6237.48	93	-12288.96	-11816.60	6.38	5.59	3.57
		AIC	13	6373.80	112	-12523.59	-11954.72	3.47	3.32	2.95
		BIC	9	6366.37	112	-12508.76	-11939.89	3.52	3.41	3.13

Table 11.8: R-vine specifications of the Norwegian financial data set obtained from different procedures (full maximum likelihood estimation) and tree construction principles (exceedance Kendall's τ uses $\delta_1 = \delta_2 = 0.2$ for lower and $\delta_1 = \delta_2 = 0.8$ for upper dependence). Test statistics indicated by "*" imply that the null hypothesis (11.1) cannot be rejected at the 5% level.

been captured in the first trees. This corresponds to the different construction principles: while exceedance dependence captures joint tail behavior and does not allow any statement regarding independence of variable pairs, Kendall's τ measures the general dependence which is used in the bivariate independence test that we use (see Section 4.3.1). Hence, the models constructed using exceedance dependence as weights are less parsimonious, but possibly capture asymmetric dependence more accurately, which we will investigate below. Furthermore note that the simplification procedures based on the Vuong test indicate that simplification is already feasible at level 1, since the model construction principle probably captured more asymmetric dependence, which cannot be modeled by Gaussian copulas, in the first tree. However, AIC/BIC and Vuong tests indicate that these models do not provide a good fit. Hence, we will concentrate on the models constructed using Kendall's τ in the following (without explicit reference to the construction principle anymore).

The chosen copula types and estimated parameters in the full R-vine model as well as in the pairwisely simplified 2 level R-vine model are shown in Appendix B.2, where the copula types and estimated parameters of the truncated models can also be deduced.

Figure 11.4 displays the first trees T_1 of the R-vine models constructed using Kendall's τ (left panel) and exceedance Kendall's τ (right panel) as weights, respectively. They are the same for all model specifications, since the R-vines are simplified at level 1 at the earliest. As already discussed above, tree construction using exceedance dependence as weights of course captures less dependence in terms of Kendall's τ which is underlined by the trees in Figure 11.4. Abbreviations as in Table 11.5 are used.

11.2.2   Model evaluation

As in Section 11.1 we now turn to the evaluation of the models considered above. In doing so, we examine the pairwisely simplified 2 level R-vine $\mathcal{M}_S(2)$ and, in particular, the pairwisely truncated R-vines at level 4 $\mathcal{M}_T(4)$ and at level 6 $\mathcal{M}_T(6)$, since the model selection criteria indicated that these models are possibly too parsimonious, i.e., truncated too early (cp. Table 11.8). Therefore, we consider a variable pair which is directly affected by truncation after tree T_4 and by simplification after T_2: the transformed variable pair NIBOR3M-EUR5Y (conditioned on previous trees) exhibits an induced Kendall's τ of -0.1 in tree T_6 of the full R-vine model, i.e., it is explicitly modeled in $\mathcal{M}_T(6)$ but truncated in $\mathcal{M}_T(4)$ and simplified in $\mathcal{M}_S(2)$. Hence, we simulated 1107 observations from each model and compare scatter plots with standard normal margins, empirical copula distributions as well as copula Q-Q plots with respect to the observed data (cp. Section 9.1) in Figure 11.5.

While scatter plots and empirical copula distributions of the simulated data do not show any systematic deviations from the observed data characteristics, the copula Q-Q plots give a more detailed impression: on the one hand, $\mathcal{M}_T(6)$, $\mathcal{M}_S(2)$ and the full model, which model the variable NIBOR3M-EUR5Y explicitly (or at least in a simplified way), quite accurately reproduce the observed data characteristics. On the other hand, the Q-Q plot of $\mathcal{M}_T(4)$, which truncates this variable pair, shows a systematic deviation from the line $y = x$ and hence indicates that this model provides a slightly inferior fit. It is however still rather close to the observed data.

Since it is not sensible to consider all $\frac{19 \times 18}{2} = 171$ variable pairs, we now consider
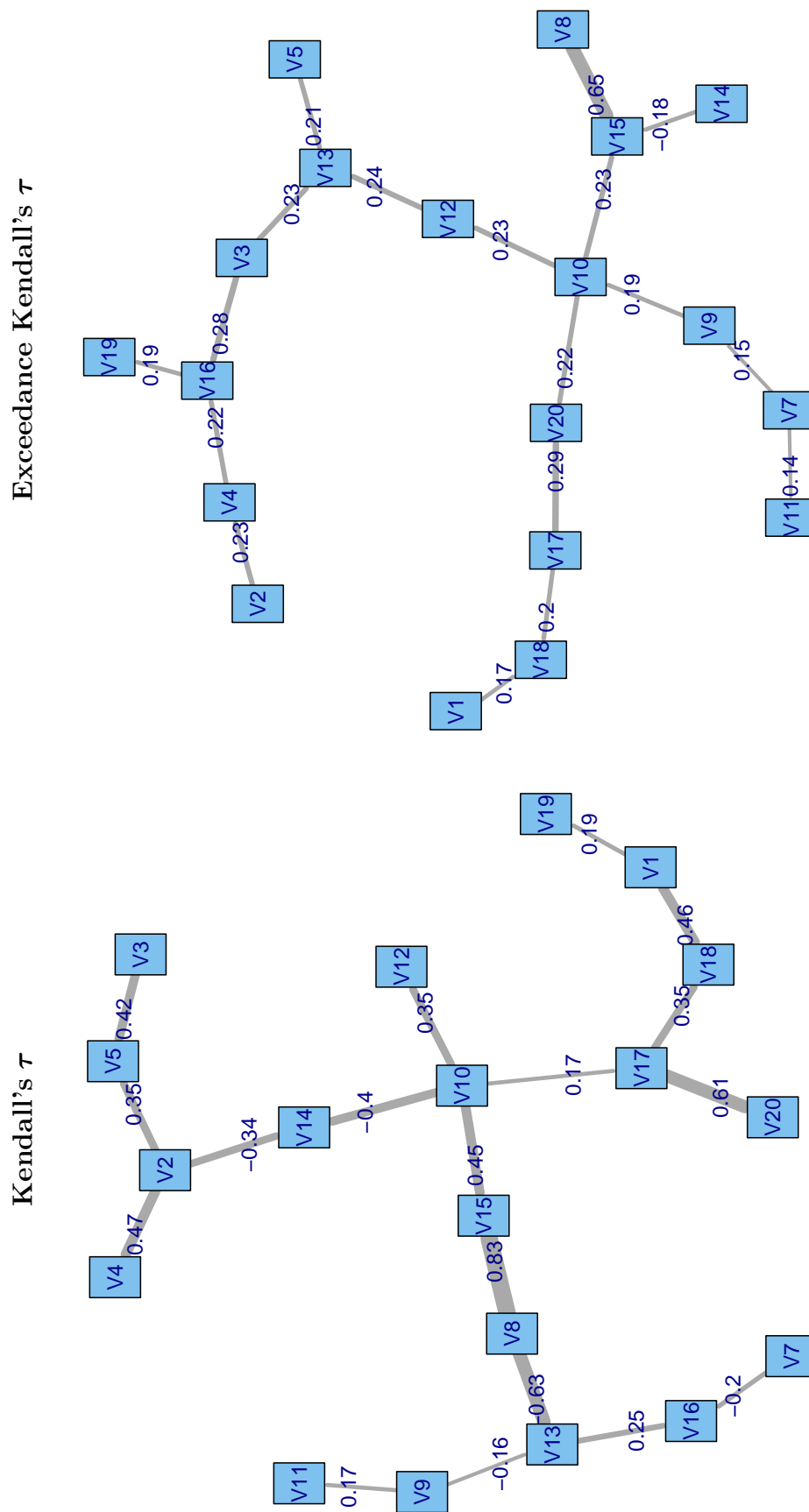
Figure 11.4: First trees T_1 of the R-vine models constructed using Kendall's τ (left panel) and exceedance Kendall's τ (right panel) as weights, respectively, for the financial data set from Norway. The edge labels indicate empirical Kendall's τ's between the respective variables.
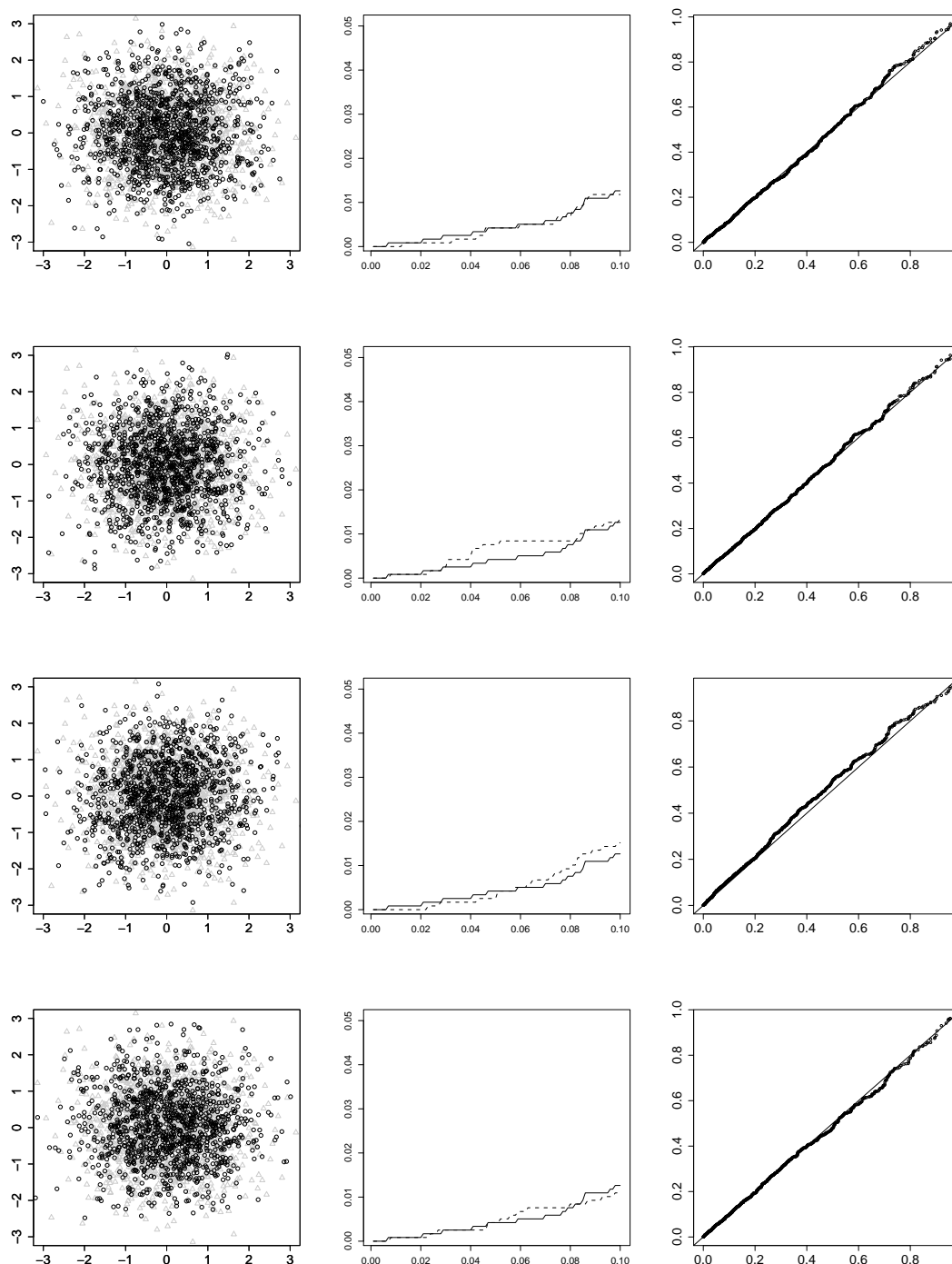
Figure 11.5: Scatter plots with standard normal margins (left column), empirical copula distributions (middle column) and copula Q-Q plots (right column) of the variable pair NIBOR3M-EUR5Y for observed and simulated data from the full R-vine model (first row) as well as $\mathcal{M}_T(6)$ (second row), $\mathcal{M}_T(4)$ (third row) and $\mathcal{M}_S(2)$ (fourth row).

	Kendall's τ				exceedance Kendall's τ	
	full	$\mathcal{M}_T(6)$	$\mathcal{M}_T(4)$	$\mathcal{M}_S(2)$	full	$\mathcal{M}_S(1)$
τ-matrix diff.	0.04056	0.04610	0.04863	0.03983	0.04369	0.04055
quartile diff.	0.05335	0.05587	0.05383	0.05499	0.05189	0.05288
τ^{lower}-matrix diff.	0.23283	0.23219	0.22834	0.23413	0.22766	0.23038
τ^{upper}-matrix diff.	0.23197	0.23018	0.23132	0.23461	0.22739	0.22717

Table 11.9: Mean absolute differences of the empirical Kendall's τ's of all variable pairs (''τ-matrix diff.'') as well as of the empirical upper and lower exceedance Kendall's τ's (''τ^{lower}-matrix diff.'' and ''τ^{upper}-matrix diff. '', respectively) and sums of absolute differences between empirical quartiles of S_i^K and S_i (''quartile diff.'') for R-vine models of the financial data set from Norway, which were constructed using Kendall's τ or exceedance Kendall's τ as weights.

summarizing criteria to evaluate the models (cp. Section 9.2). As in Section 11.1.1, we examine the mean absolute difference of the empirical Kendall's τ's of all variable pairs as well as the sum of absolute differences between empirical quartiles of S_i^K and S_i which are defined in (9.1) and correspond to equally weighted portfolios of all variables (cp. (10.1)). Furthermore, we consider the mean absolute differences of the empirical upper and lower exceedance Kendall's τ's of all variable pairs in order to investigate whether the models are able to reproduce the joint tail behavior of the data. We compare the results to the full and simplified 1 level R-vine models constructed using exceedance dependence as weights. The mean results of $R = 100$ repeated simulations are shown in Table 11.9.

Given that the models constructed using exceedance dependence as weights use more parameters, it is not surprising that they perform quite well in terms of all criteria. However, also the full model and $\mathcal{M}_S(2)$, which were constructed using Kendall's τ as weights, exhibit good results with respect to the empirical Kendall's τ's and quartiles of S_i^K, but, as expected, the simplified model is weaker in reproducing the asymmetric tail behavior as measured by the lower and upper exceedance dependence. The truncated models $\mathcal{M}_T(6)$ and $\mathcal{M}_T(4)$ show some problems with respect to empirical Kendall's τ's, but are superior to the simplified model when modeling asymmetric dependence, since they model more trees explicitly. Altogether, corresponding to their construction principle, the models obtained by construction with exceedance Kendall's τ as weights are best for reproducing the asymmetric dependence of the data. However, this is apparently a rather complicated issue as the deviations are substantial for all models. This result underlines the problem of adequately estimating joint tail behavior of variables, since only a limited number of observations can be used (cp. Section 3.1). In terms of the other criteria, the models can be seen as adequate specifications for the data.

11.2.3 Interpretation

Model selection and evaluation criteria show that an adequate R-vine model for the financial copula data set from Norway can be truncated at level 6 or even 4 depending on the desired level of parsimony (of course at the expense of accuracy), i.e., the most

important dependencies in the transformed residuals are captured in the first four to six trees. Moreover, simplification at level 2 is also appropriate indicating that all important asymmetric dependencies are already captured in the first two trees.

In economical terms, this has an apparent interpretation: the first R-vine tree in the left panel of Figure 11.4 shows that there are clusters of economically similar variables. In the upper part of the tree, the dependencies among the exchange rates are modeled. The link to the other variables is established by the USD-NOK exchange rate (V2) and the WGBI (V14) which are internationally important variables. Then the relationships of interest rates and (government) bonds are modeled, where the 5-year German Government Rate (V10) and the BRIX (V13) play central roles due to their role for the European and Norwegian market, respectively. The 5-year German Government Rate also establishes the link to the bottom part of the tree which models the relationships among stock indices and related markets, where the MSCI World (V17) obviously is fundamental for explaining the dependencies in the residuals. Hence, the simplification and truncation results mean that the most important (asymmetric) dependencies are mainly found within these three groups of variables and modeling a few links is sufficient to establish a good model, e.g., it is not necessary to explicitly model the dependency between the YEN-NOK exchange rate (V4) and the Norwegian 5-year Swap Rate (V8).

In terms of asymmetric dependencies, model construction using exceedance dependence as weights allows a different perspective (see the right panel in Figure 11.4): while there is also a cluster of stock indices (bottom left part of the tree) and the 5-year German Government Rate (V10) plays a central role again, the short term interest rates (V7, V9, V11) are now also clearly clustered in contrast to the exchange rates which are intermixed with long term interest rates and (government) bonds, where the BRIX and ST2X indices play central roles. Hence, if one is interested in modeling asymmetric dependencies among variables more explicitly, one can choose the less parsimonious models obtained by this alternative construction principle.

11.3 Extended CAPM

In this section we want to review and evaluate the work of Heinen and Valdesogo (2009). Using a simplified C-vine structure, they construct an extended Capital Asset Pricing Model (CAPM) with sectorial effects, which can capture non-linear and non-Gaussian behavior of stock returns. They however make crucial (independence) assumptions in order to specify their model. We therefore construct a similar model for the log returns of 46 stocks of the Euro Stoxx 50 as well as five national leading stock indices: the German DAX, the French CAC 40, the Dutch AEX, the Spanish IBEX 35 and the Italian FTSE MIB. Based on this data, we will investigate the assumptions of Heinen and Valdesogo (2009) and compare appropriate models including R- and C-vine specifications as well as a newly proposed dependency model for financial returns based on R-vines, the so-called *regular vine market sector model*.

11.3.1 Canonical vine market sector model

Heinen and Valdesogo (2009) developed an extension of the CAPM (Sharpe 1964) which can capture non-linear and non-Gaussian behavior of the cross-section of asset returns as well as model their dependencies to the market and the respective sector return. Their model is based on a simplified C-vine structure and therefore called *canonical vine autoregressive model*, where "autoregressive" refers to the fact that the chosen bivariate copulas are possibly time-varying. It is constructed based on the classical CAPM which is the most famous and also simplest factor model for the cross-section of asset returns: the classical CAPM assumes that at time t the individual asset returns $r_{i,t}$, the market return $r_{M,t}$ and the idiosyncratic error terms $\varepsilon_{i,t}$, which are independent of $\varepsilon_{i,t-1}$ and of $\varepsilon_{j,t}$ $\forall j \neq i$, are jointly normally distributed and follow the linear relationship

$$r_{i,t} = \beta_i r_{M,t} + \varepsilon_{i,t},$$

where β_i is usually called the *sensitivity* of asset i.

Heinen and Valdesogo (2009) loosen these assumptions of normality and linearity by using a variety of GARCH-models for the marginal time series of stock returns and by modeling the dependence between assets and the market with bivariate copulas for the transformed residuals (note that this is again a two step estimation approach in contrast to the Bayesian joint estimation approach of Hofmann and Czado (2010) and justified by the dimension of most financial data sets). The remaining (idiosyncratic) dependence is captured with a multivariate Gaussian copula. This obviously corresponds to a jointly simplified 1 level C-vine, where the root node of the first tree is chosen as the market. Heinen and Valdesogo (2009) call this the *market model*.

As the market model can only capture a limited amount of dependencies in the residuals, Heinen and Valdesogo (2009) extend this model by adding additional factors, namely a set of sectors, where each asset belongs to one sector (e.g., utilities or financial services). Then each asset is assumed to depend on the market and on its sector. In terms of a C-vine, this model induces independence assumptions: conditionally on the market, sectorial returns are assumed to be independent and asset returns independent of sector returns other than their own. The remaining dependence of asset returns conditioned on the market and on the respective sectors is again modeled by a multivariate Gaussian copula. The model is then called the *market sector model* and, if p is the number of sectors, it corresponds to a jointly simplified $p+1$ level C-vine, where the first root node is the market and the root nodes 2 to $p+1$ are the sectors. In the following, we will refer to it as *canonical vine market sector (CVMS) model* in order to highlight the underlying model structure. It is illustrated in the following example (cp. Heinen and Valdesogo (2009)).

Example 13 (CVMS model.) *Let r_1^A, r_2^A, r_1^B and r_2^B be the returns of stocks belonging to sectors A and B, respectively. Further, let r_A and r_B be the returns of sectors A and B as well as r_M the market return. According to the CVMS model, the following independence assumptions hold:*

(i) r_A is independent of r_B conditioned on r_M, and

(ii) r_1^A and r_2^A are independent of r_B conditioned on r_M, while r_1^B and r_2^B are independent of r_A conditioned on r_M.
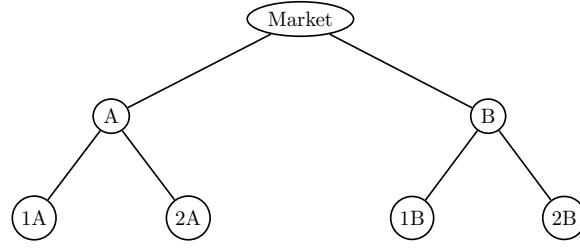
Figure 11.6: Dependence structure of the CVMS model with two sectors A and B.

Then the joint density of the returns r_1^A, r_2^A, r_1^B, r_2^B, r_A, r_B and r_M is given by

$$
\begin{aligned}
f(r_1^A, r_2^A, r_1^B, r_2^B, r_A, r_B, r_M) &= f(r_1^A)f(r_2^A)f(r_1^B)f(r_2^B)f(r_A)f(r_B)f(r_M) \\
&\times c_{1A,M}\left(F(r_1^A), F(r_M)\right) c_{2A,M}\left(F(r_2^A), F(r_M)\right) \\
&\times c_{1B,M}\left(F(r_1^B), F(r_M)\right) c_{2B,M}\left(F(r_2^B), F(r_M)\right) \\
&\times c_{A,M}\left(F(r_A), F(r_M)\right) c_{B,M}\left(F(r_B), F(r_M)\right) \\
&\times c_{1A,A|M}\left(F(r_1^A|r_M), F(r_A|r_M)\right) c_{2A,A|M}\left(F(r_2^A|r_M), F(r_A|r_M)\right) \\
&\times c_{1B,B|M}\left(F(r_1^B|r_M), F(r_B|r_M)\right) c_{2B,B|M}\left(F(r_2^B|r_M), F(r_B|r_M)\right) \\
&\times c_{1A,1B,2A,2B|M,A,B}^{\rho}\left(F(r_1^A|r_M, r_A), ..., F(r_2^B|r_M, r_B)\right),
\end{aligned}
$$

(11.2)

where $c_{1A,1B,2A,2B|M,A,B}^{\rho}$ denotes a four-dimensional Gaussian copula and Figure 11.6 illustrates the dependence structure of the model.

 We can rewrite this density as the density of a jointly simplified 3 level C-vine with the market as the first root node and the sectors as second and third root nodes, where the order is arbitrary due to the independence assumptions. Now, let sector A be the second and B the third root node. The copula terms of the first C-vine tree T_1 are

$$
c_{1A,M}\left(F(r_1^A), F(r_M)\right), c_{2A,M}\left(F(r_2^A), F(r_M)\right), c_{1B,M}\left(F(r_1^B), F(r_M)\right),
$$
$$
c_{2B,M}\left(F(r_2^B), F(r_M)\right), c_{A,M}\left(F(r_A), F(r_M)\right), c_{B,M}\left(F(r_B), F(r_M)\right)
$$

as in lines 2 to 4 in (11.2). Since A is the second root node, the pair copulas of the second tree T_2 are given by

$$
c_{1A,A|M}\left(F(r_1^A|r_M), F(r_A|r_M)\right), c_{2A,A|M}\left(F(r_2^A|r_M), F(r_A|r_M)\right),
$$
$$
\underbrace{c_{1B,A|M}\left(F(r_1^B|r_M), F(r_A|r_M)\right)}_{\overset{(ii)}{=}1}, \underbrace{c_{2B,A|M}\left(F(r_2^B|r_M), F(r_A|r_M)\right)}_{\overset{(ii)}{=}1},
$$

according to independence assumption (ii) (the density of the independence copula is of course 1) and

$$
c_{A,B|M}\left(F(r_A|r_M), F(r_B|r_M)\right) \overset{(i)}{=} 1
$$

because of the first independence assumption (cp. line 5 in (11.2)). Further, the copulas of the third tree T_3 are

$$
\underbrace{c_{1A,B|M,A}\left(F(r_1^A|r_M, r_A), F(r_B|r_M, r_A)\right)}_{\overset{(ii)}{=}1}, \underbrace{c_{2A,B|M,A}\left(F(r_2^A|r_M, r_A), F(r_B|r_M, r_A)\right)}_{\overset{(ii)}{=}1},
$$
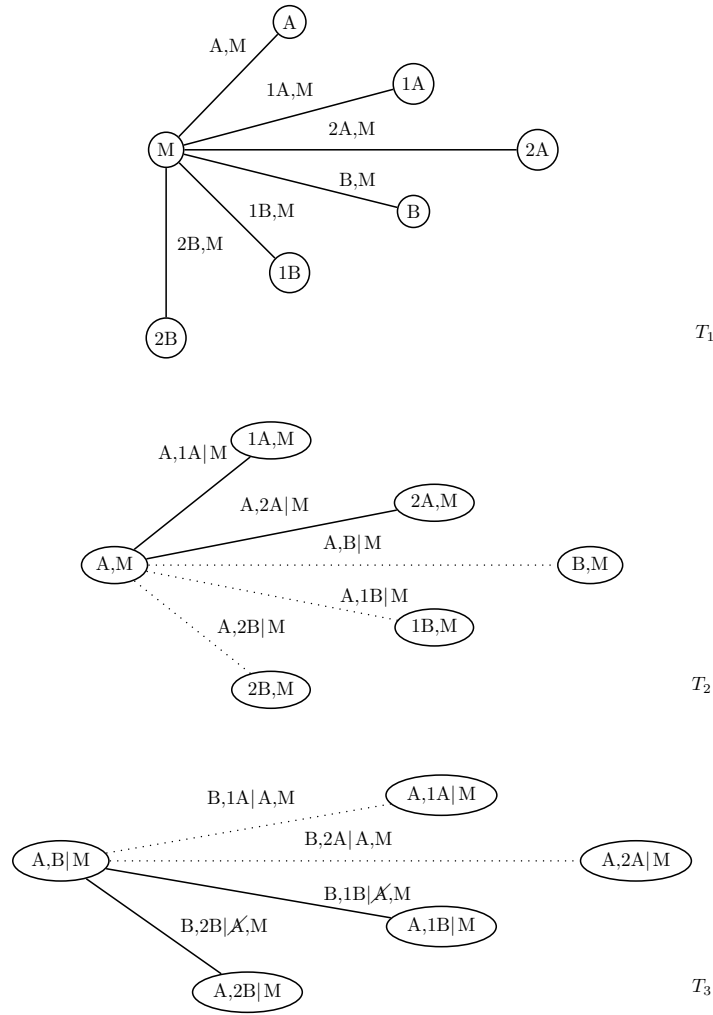
Figure 11.7: First three trees of the CVMS model in Example 13.

as well as

$$c_{1B,B|M,A}\Big(\underbrace{F(r_1^B|r_M,r_A)}_{\overset{(ii)}{=}F(r_1^B|r_M)},\underbrace{F(r_B|r_M,r_A)}_{\overset{(i)}{=}F(r_B|r_M)}\Big) = c_{1B,B|M}\left(F(r_1^B|r_M),F(r_B|r_M)\right),$$

and similarly for $c_{2B,B|M,A}$ using both independence assumptions (cp. line 6 in (11.2)).

Finally, the input arguments of the simplifying four-dimensional Gaussian copula $c_{1A,1B,2A,2B|M,A,B}^{\rho}$ as in line 7 of (11.2) are given by $F(r_i^A|r_M,r_A,r_B) = F(r_i^A|r_M,r_A)$ and $F(r_i^B|r_M,r_A,r_B) = F(r_i^B|r_M,r_B)$ for $i = 1,2$ due to independence assumption (ii).

Combining all equations, we then get the copulas in (11.2), i.e., Equation (11.2) is the density of a jointly simplified 3 level C-vine under independence assumptions (i) and (ii) and where the order of the second and third root node is obviously arbitrary. The first three trees of the C-vine are shown in Figure 11.7. Dotted lines illustrate the independence assumptions, i.e., independence copulas are chosen for the copulas corresponding to dotted edges.

indices	Germany	France	Netherlands	Spain	Italy
^STOXX50E	ALV.DE	ACA.PA	AGN.AS	BBVA.MC	ENEL.MI
^GDAXIP	BAS.DE	AI.PA	INGA.AS	IBE.MC	ENI.MI
^FCHI	BAYN.DE	ALO.PA	PHIA.AS	REP.MC	G.MI
^AEX	DAI.DE	BN.PA	UNA.AS	SAN.MC	ISP.MI
^IBEX	DB1.DE	BNP.PA		TEF.MC	TIT.MI
FTSEMIB.MI	DBK.DE	CA.PA			UCG.MI
	DTE.DE	CS.PA			
	EOAN.DE	DG.PA			
	MUV2.DE	FP.PA			
	RWE.DE	FTE.PA			
	SAP.DE	GLE.PA			
	SIE.DE	GSZ.PA			
		MC.PA			
		OR.PA			
		SAN.PA			
		SGO.PA			
		SU.PA			
		UL.PA			
		VIV.PA			

Table 11.10: Ticker symbols of analyzed indices and stocks.

Obviously, the CVMS model makes some crucial independence assumptions in order to fit the factor model to a C-vine structure. We will therefore construct such a model and investigate these assumptions as well as compare the results to R- and C-vines specified by our simplification procedures. In doing so, we will not consider consider dynamical terms in order to limit the model complexity to a reasonable amount.

11.3.2 Euro Stoxx 50 and national stock indices

In order to critically evaluate the CVMS model discussed in the previous section, we now consider the log returns of 46 stocks of the Euro Stoxx 50 (market variable) according to its composition as of 2/8/2010 and adjusted for dividends and splits. As sectorial variables we furthermore consider the five national leading stock indices corresponding to the home countries of the chosen stocks: the German DAX, the French CAC 40, the Dutch AEX, the Spanish IBEX 35 and the Italian FTSE MIB, where all indices are taken as price indices (instead of performance indices). Four stocks of the Euro Stoxx 50 are not considered, since they are the only ones from their country (Nokia, Anheuser-Busch InBev, Cement Roadstone Holding) or listed in multiple indices (ArcelorMittal). These four stocks correspond to 6.7% of the total index weight. All stocks and indices considered in our analyses are shown in Table 11.10. Time series of the indices are shown in Figure 11.8 for the time period from 5/22/2006 to 4/29/2010.

As in the previous Sections 11.1 and 11.2 and according to Heinen and Valdesogo
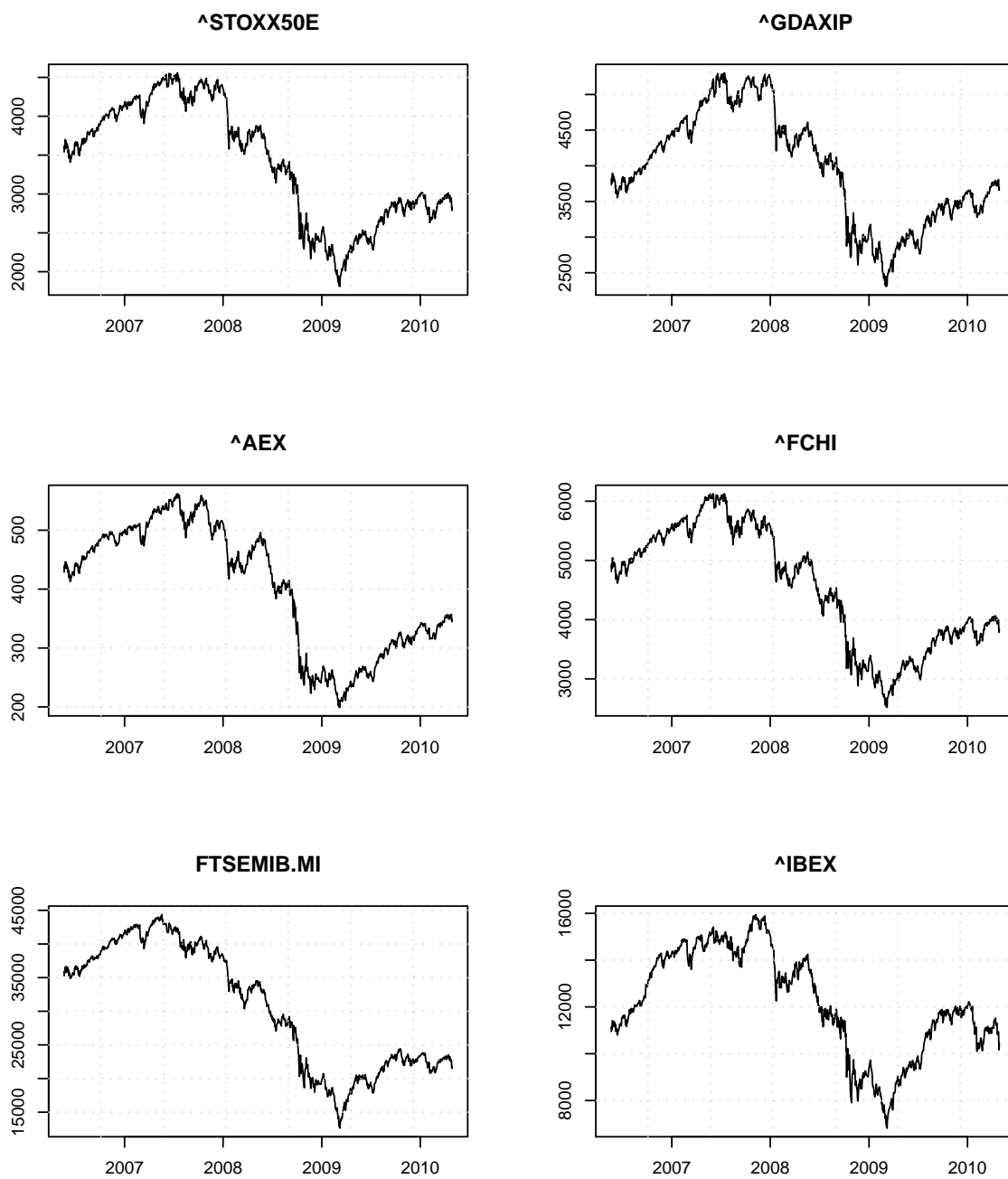
Figure 11.8: Time series of the Euro Stoxx 50 and the leading national stock indices of Germany, the Netherlands, France, Italy and Spain.
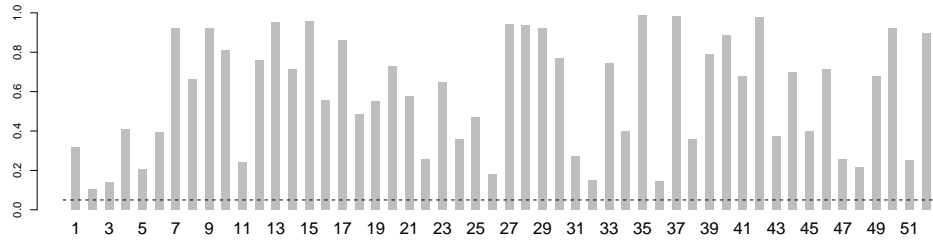
Figure 11.9: P-values of the Kolmogorov-Smirnov tests for the residuals of the time series models presented in Table 11.11. The dashed line indicates the 5% level. The numbers correspond to the rows in Table 11.11.
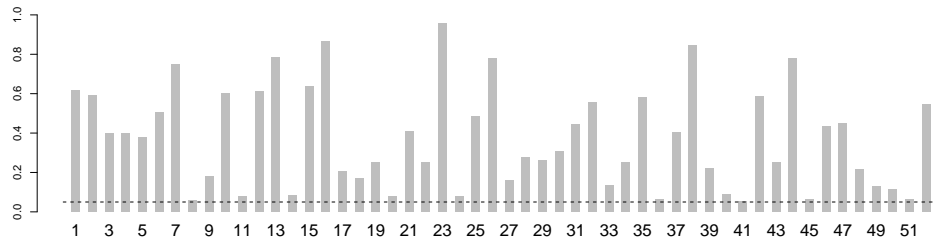
Figure 11.10: P-values of the Ljung-Box tests with lag 30 for the residuals of the time series models presented in Table 11.11. The dashed line indicates the 5% level. The numbers correspond to the rows in Table 11.11.

(2009), appropriate models for the univariate time series have to be found to obtain i.i.d. copula data which can then be used for analyzing the dependence. Here, time series models (see Section 2.5) are selected according to a stepwise procedure:

  (i) We fit ARMA(1,1)-GARCH(1,1)-, AR(1)-GARCH(1,1)-, MA(1)-GARCH(1,1)- and GARCH(1,1)-models with mean and Student-t error distribution to the univariate time series and perform Kolmogorov-Smirnov goodness-of-fit tests (see Section 4.1.1) for the standardized residuals. The model with the highest p-value is chosen if this value is at least larger than 5%.

 (ii) If the degrees of freedom of the Student-t error distribution are larger than ten, we choose a standard normal distribution instead (if the p-value of the corresponding Kolmogorov-Smirnov test is larger than 5%).

(iii) Since the Kolmogorov-Smirnov test sometimes lacks power, we also perform Ljung-Box tests (see Section 2.5.4) with lag 30 for all residuals. If a p-value is smaller than 5%, we stepwisely increase the corresponding ARMA(p,q) terms, so that the model remains rather parsimonious, until both the Ljung-Box test and the respective Kolmogorov-Smirnov test for the residuals have p-values larger than 5%.

The resulting marginal time series models are shown in Table 11.11, while Figures 11.9 and 11.10 display the p-values of the Kolmogorov-Smirnov and Ljung-Box tests, respectively.

	model	$\hat{\mu}$	$\hat{\phi}_1$	$\hat{\theta}_1$	$\hat{\omega}$	$\hat{\alpha}_1$	$\hat{\beta}_1$	$\hat{\nu}$	KS	LB
^STOXX50E	GARCH(1,1)	0.001	-	-	0.000	0.104	0.891	8.586	0.317	0.615
^GDAXIP	GARCH(1,1)	0.000	-	-	0.000	0.087	0.906	8.665	0.103	0.589
^AEX	GARCH(1,1)	0.001	-	-	0.000	0.115	0.880	-[a]	0.142	0.399
^FCHI	GARCH(1,1)	0.001	-	-	0.000	0.096	0.897	9.489	0.409	0.400
FTSEMIB.MI	GARCH(1,1)	0.000	-	-	0.000	0.104	0.897	9.426	0.206	0.376
^IBEX	GARCH(1,1)	0.000	-	-	0.000	0.105	0.890	7.212	0.393	0.503
ACA.PA	GARCH(1,1)	-0.000	-	-	0.000	0.085	0.915	6.611	0.921	0.749
AGN.AS	GARCH(1,1)	0.000	-	-	0.000	0.129	0.875	7.401	0.662	0.180
AI.PA	MA(1)-GARCH(1,1)	0.001	-	-0.114	0.000	0.094	0.873	5.932	0.923	0.060
ALO.PA	AR(1)-GARCH(1,1)	0.001	-0.065	-	0.000	0.058	0.935	6.732	0.810	0.602
ALV.DE	ARMA(5,0)-GARCH(1,1)[b]	0.001	0.012	-	0.000	0.092	0.903	6.538	0.240	0.078
BAS.DE	ARMA(1,1)-GARCH(1,1)	0.002	-0.684	0.668	0.000	0.097	0.900	5.518	0.760	0.612
BAYN.DE	ARMA(1,1)-GARCH(1,1)	0.000	0.522	-0.570	0.000	0.106	0.849	6.591	0.951	0.787
BBVA.MC	GARCH(1,1)	0.000	-	-	0.000	0.115	0.878	7.160	0.711	0.085
BN.PA	ARMA(1,1)-GARCH(1,1)	0.000	0.626	-0.699	0.000	0.089	0.893	8.381	0.958	0.640
BNP.PA	GARCH(1,1)	0.000	-	-	0.000	0.089	0.909	7.189	0.554	0.864
CA.PA	GARCH(1,1)	0.000	-	-	0.000	0.069	0.926	5.680	0.860	0.208
CS.PA	MA(1)-GARCH(1,1)	0.000	-	0.019	0.000	0.106	0.890	9.658	0.486	0.171
DAI.DE	AR(1)-GARCH(1,1)	0.001	0.011	-	0.000	0.102	0.896	5.319	0.552	0.251
DB1.DE	GARCH(1,1)	0.001	-	-	0.000	0.086	0.908	7.695	0.730	0.079
DBK.DE	AR(1)-GARCH(1,1)	0.000	0.028	-	0.000	0.116	0.891	6.442	0.576	0.408
DG.PA	GARCH(1,1)	0.001	-	-	0.000	0.094	0.902	8.534	0.256	0.254
DTE.DE	GARCH(1,1)	0.000	-	-	0.000	0.075	0.918	5.197	0.647	0.956
ENEL.MI	MA(1)-GARCH(1,1)	0.001	-	-0.073	0.000	0.124	0.870	6.325	0.356	0.078
ENI.MI	ARMA(1,1)-GARCH(1,1)	0.001	-0.580	0.526	0.000	0.112	0.874	8.328	0.470	0.484
EOAN.DE	GARCH(1,1)	0.001	-	-	0.000	0.100	0.881	4.809	0.179	0.781
FP.PA	AR(1)-GARCH(1,1)	0.000	-0.001	-	0.000	0.089	0.893	8.904	0.944	0.161
FTE.PA	MA(1)-GARCH(1,1)	0.000	-	-0.021	0.000	0.037	0.954	5.802	0.936	0.274
G.MI	GARCH(1,1)	0.000	-	-	0.000	0.102	0.883	5.635	0.922	0.263
GLE.PA	ARMA(1,1)-GARCH(1,1)	-0.000	-0.416	0.498	0.000	0.117	0.888	7.323	0.770	0.305
GSZ.PA	AR(1)-GARCH(1,1)	0.001	-0.032	-	0.000	0.090	0.894	8.604	0.272	0.443
IBE.MC	ARMA(1,1)-GARCH(1,1)	0.000	-0.217	0.232	0.000	0.182	0.792	5.047	0.152	0.558
INGA.AS	GARCH(1,1)	0.000	-	-	0.000	0.166	0.846	5.952	0.746	0.137
ISP.MI	ARMA(1,1)-GARCH(1,1)	-0.000	-0.133	0.166	0.000	0.103	0.893	6.325	0.398	0.250
MC.PA	GARCH(1,1)	0.001	-	-	0.000	0.060	0.933	7.057	0.990	0.581
MUV2.DE	ARMA(1,1)-GARCH(1,1)	0.001	-0.782	0.744	0.000	0.116	0.868	5.877	0.145	0.062
OR.PA	MA(1)-GARCH(1,1)	0.001	-	-0.087	0.000	0.082	0.893	8.200	0.983	0.405
PHIA.AS	MA(1)-GARCH(1,1)	0.001	-	-0.035	0.000	0.072	0.924	7.058	0.358	0.846
REP.MC	ARMA(1,1)-GARCH(1,1)	0.000	0.494	-0.502	0.000	0.111	0.871	5.994	0.792	0.223
RWE.DE	GARCH(1,1)	0.001	-	-	0.000	0.067	0.906	5.728	0.886	0.087
SAN.MC	MA(1)-GARCH(1,1)	0.001	-	-0.017	0.000	0.124	0.873	8.252	0.677	0.052
SAN.PA	GARCH(1,1)	0.000	-	-	0.000	0.049	0.946	5.068	0.975	0.586
SGO.PA	ARMA(1,1)-GARCH(1,1)	0.001	-0.693	0.639	0.000	0.094	0.907	7.899	0.375	0.779
SIE.DE	GARCH(1,1)	0.001	-	-	0.000	0.058	0.936	5.161	0.700	0.063
SU.PA	ARMA(3,0)-GARCH(1,1)[c]	0.001	-0.052	-	0.000	0.073	0.920	7.556	0.396	0.435
TEF.MC	AR(1)-GARCH(1,1)	0.001	-0.006	-	0.000	0.094	0.882	7.507	0.714	0.451
TIT.MI	ARMA(1,1)-GARCH(1,1)	-0.000	0.820	-0.867	0.000	0.111	0.869	8.506	0.256	0.217
UCG.MI	AR(1)-GARCH(1,1)	0.000	-0.023	-	0.000	0.109	0.899	8.399	0.218	0.128
UL.PA	GARCH(1,1)	0.001	-	-	0.000	0.099	0.889	8.806	0.677	0.115
UNA.AS	GARCH(1,1)	0.001	-	-	0.000	0.083	0.902	5.791	0.922	0.064
VIV.PA	GARCH(1,1)	0.001	-	-	0.000	0.091	0.902	-[a]	0.250	0.544
SAP.DE	MA(1)-GARCH(1,1)	0.000	-	0.010	0.000	0.027	0.961	3.970	0.898	0.251

Table 11.11: Marginal time series models for the log returns of the considered six indices and 46 stocks.

[a]Normal error distribution.

[b]$\hat{\phi}_2 = -0.000$, $\hat{\phi}_3 = -0.036$, $\hat{\phi}_3 = 0.010$ and $\hat{\phi}_4 = -0.040$.

[c]$\hat{\phi}_2 = -0.017$ and $\hat{\phi}_3 = -0.092$.
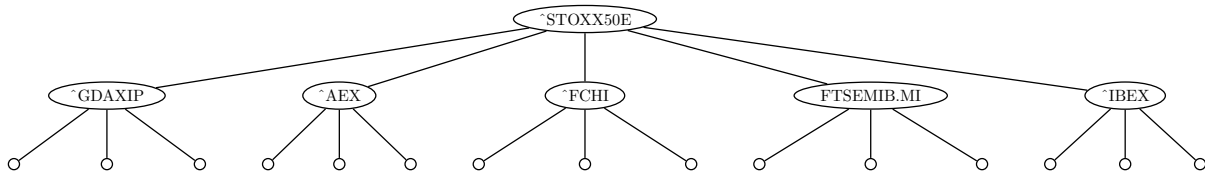
Figure 11.11: Dependence structure of the CVMS model for the Euro Stoxx 50 data.

11.3.3 Validity of independence assumptions

The CVMS model considered in Section 11.3.1 crucially relies on independence assumptions to construct a jointly simplified C-vine structure. In order to specify a CVMS model for the Euro Stoxx 50 data described above, these assumptions can be explicitly stated as:

(i) the returns of the national indices conditioned on the market return, i.e., on the return of the Euro Stoxx 50, are independent and

(ii) conditionally on the market return, the individual returns are independent of returns of national indices other than their own, e.g., returns of German stocks are independent of the French CAC 40 conditioned on the Euro Stoxx 50.

Figure 11.11 illustrates the dependence structure of the Euro Stoxx 50 data. This tree should however not be confounded with R- or C-vine trees.

We now want to investigate whether the independence assumptions of the CVMS model are appropriate for the Euro Stoxx 50 data. Here, the CVMS model corresponds to a jointly simplified 6 level C-vine, where ˆSTOXX50E is the first root node. As the order of the root nodes 2 to 6 in the conditioning set of the C-vine is arbitrary (cp. Example 13), we choose the order according to the number of stocks under consideration (see Table 11.10), i.e., we choose ˆFCHI as second, ˆGDAXIP as third, FTSEMIB.MI as fourth, ˆIBEX as fifth and finally ˆAEX as sixth root node. The stepwise construction of the CVMS model for the transformed residuals of the Euro Stoxx 50 data is described in the following.

(i) Appropriate (according to the AIC) pair copulas are fitted for the first tree T_1 with root node ˆSTOXX50E, i.e., for all 51 variable pairs involving ˆSTOXX50E, to model the dependence on the market.

(ii) Second tree T_2 with root node ˆFCHI: We specify appropriate pair copulas for all 19 returns of French stocks with respect to ˆFCHI conditionally on ˆSTOXX50E. According to the CVMS model, for all other stock returns and indices the respective copulas with respect to ˆFCHI are set to independence copulas due to independence assumptions (i) (for the indices) and (ii) (for the stocks). These assumptions are investigated in the first column of Table 11.12 which displays p-values of the bivariate independence test based on Kendall's τ (cp. Section 4.3.1) for variable pairs involving ˆFCHI.

	^FCHI	^GDAXIP	FTSEMIB.MI	^IBEX	^AEX
^FCHI	-	*0.26**	*0.91**	0.00	0.00
^GDAXIP	*0.26**	-	0.00	0.00	*0.86**
FTSEMIB.MI	*0.91**	0.00	-	*0.32**	*0.05**
^IBEX	0.00	0.00	*0.32**	-	0.00
^AEX	0.00	*0.86**	*0.05**	0.00	-
ACA.PA	-	0.01	0.00	*0.38**	*0.74**
AI.PA	-	0.00	*0.94**	0.00	0.00
ALO.PA	-	0.00	*0.07**	*0.82**	0.00
BN.PA	-	*0.05**	0.02	*0.64**	*0.20**
BNP.PA	-	0.00	0.01	*0.56**	0.00
CA.PA	-	0.02	0.04	*0.68**	*0.36**
CS.PA	-	*0.62**	0.00	*0.33**	*0.29**
DG.PA	-	*0.55**	*0.15**	0.01	0.01
FP.PA	-	*0.69**	*0.34**	0.00	0.00
FTE.PA	-	0.01	0.04	0.01	0.00
GLE.PA	-	0.00	0.00	*0.88**	*0.13**
GSZ.PA	-	*0.37**	*0.13**	*0.70**	*0.47**
MC.PA	-	*0.09**	*0.26**	*0.73**	0.00
OR.PA	-	*0.12**	0.03	*0.60**	*0.09**
SAN.PA	-	*0.10**	*0.05**	0.00	0.03
SGO.PA	-	*0.12**	*0.79**	*0.55**	0.01
SU.PA	-	0.00	*0.27**	*0.60**	0.00
UL.PA	-	*0.73**	0.02	0.00	0.00
VIV.PA	-	*0.32**	*0.72**	*0.06**	*0.20**
ALV.DE	0.01	-	0.01	*0.32**	*0.14**
BAS.DE	*0.90**	-	*0.32**	0.00	0.01
BAYN.DE	*0.69**	-	0.01	0.00	*0.05**
DAI.DE	*0.08**	-	*0.47**	*0.93**	0.04
DB1.DE	*0.07**	-	*0.07**	*0.90**	*0.37**
DBK.DE	0.00	-	0.00	*0.88**	*0.29**
DTE.DE	0.00	-	*0.17**	0.01	0.00
EOAN.DE	*0.63**	-	0.02	*0.24**	0.02
MUV2.DE	0.01	-	*0.44**	*0.07**	*0.50**
RWE.DE	*0.27**	-	*0.70**	*0.86**	*0.30**
SAP.DE	*0.06**	-	*0.50**	*0.09**	*0.19**
SIE.DE	*0.13**	-	0.01	0.00	*0.07**
ENEL.MI	*0.12**	*0.08**	-	*0.74**	*0.93**
ENI.MI	0.00	*0.08**	-	0.00	0.00
G.MI	0.02	0.02	-	*0.52**	*0.24**
ISP.MI	0.00	0.02	-	*0.31**	*0.05**
TIT.MI	0.02	*0.20**	-	0.02	*0.67**
UCG.MI	0.00	0.00	-	*0.86**	0.01
BBVA.MC	0.00	0.00	0.01	-	*0.12**
IBE.MC	0.00	0.00	0.00	-	0.03
REP.MC	*0.73**	0.00	*0.51**	-	*0.09**
SAN.MC	0.00	0.00	*0.15**	-	0.00
TEF.MC	0.00	0.00	0.00	-	0.00
AGN.AS	*0.42**	*0.07**	0.00	*0.24**	-
INGA.AS	0.04	0.01	0.00	*0.26**	-
PHIA.AS	0.00	*0.05**	*0.43**	0.01	-
UNA.AS	*0.25**	*0.18**	*0.17**	0.03	-

Table 11.12: P-values of bivariate independence tests based on Kendall's τ (cp. Section 4.3.1) for the Euro Stoxx 50 data **conditioned on ^STOXX50E**, e.g., the entry in row 6, column 2 corresponds to the independence test between $F(r_{ACA.PA}|r_{STOXX50E})$ and $F(r_{GDAXIP}|r_{STOXX50E})$. P-values indicated by "*" imply that the independence hypothesis cannot be rejected at the 5% level in accordance with the assumptions of the CVMS model.

	^FCHI	^GDAXIP	FTSEMIB.MI	^IBEX	^AEX
French	-	42%	47%	32%	58%
German	33%	-	42%	33%	33%
Italian	83%	50%	-	33%	33%
Spanish	80%	100%	60%	-	60%
Dutch	50%	25%	50%	50%	-
total	56%	50%	47%	34%	48%

Table 11.13: Percentages of rejection of the independence hypothesis for stocks from each country versus indices of countries other than their own conditioned on ^STOXX50E (see Table 11.12).

(iii) Third tree T_3 with root node ^GDAXIP: As in the previous step, we choose appropriate pair copulas for all 12 returns of German stocks with respect to ^GDAXIP conditionally on ^STOXX50E (no conditioning on ^FCHI because of the independence assumptions; cp. Example 13). All other copulas are set to independence copulas even if the bivariate independence test rejects the null hypothesis of independence, where the corresponding p-values are shown in the second column of Table 11.12.

(iv) Trees T_4, T_5 and T_6 are specified similarly. P-values of the corresponding bivariate independence tests are shown in columns 3, 4 and 5 of Table 11.12.

(v) A 46-dimensional Gaussian copula is fitted to the transformed variables obtained from tree T_6. (The parameters of the correlation matrix are estimated by the inversion of Kendall's τ (see Table 2.1) due to the computational complexity of estimating the entries of the 46-dimensional matrix.) We perform the copula goodness-of-fit test based on Rosenblatt's transformation (see Section 4.2.3), since this test is computationally very efficient and performed quite well in the simulation study in Section 10.2.1.

Table 11.13 summarizes the results of the bivariate independence tests to examine the independence assumptions. It shows the percentages of rejected independence hypotheses for stocks from each country (e.g., the null hypothesis of independence of German stocks and ^FCHI is rejected for 33% of the German stocks), where the CVMS model assumes 0%.

Apparently, the independence assumptions do not hold in general. In particular, conditionally on the market, i.e., the Euro Stoxx 50, there is significant dependence of Italian and Spanish stocks on the French CAC 40 as well as dependence of all Spanish stocks on the German DAX, which contradicts independence assumption (ii). In total, the assumption is wrong for about 50% of all variable pairs under consideration. Moreover, the null hypothesis of independence is pairwisely rejected for the Spanish IBEX 35, the Dutch AEX and the CAC 40 as well as for the Italian FTSE MIB and the IBEX 35 with respect to the DAX in contradiction to independence assumption (i).

As indicated by the problematic independence assumptions, there are important dependencies which are neglected in the first trees. The CVMS model now aims at capturing all remaining dependencies with a multivariate Gaussian copula. This however means that the transformed variables obtained from tree T_6 are jointly normal which is a rather strict assumption given that the data might exhibit, e.g., asymmetric dependence. Hence, we performed the copula goodness-of-fit test based on Rosenblatt's transformation which clearly rejected the null hypothesis of normality with a p-value of 0, i.e., a multivariate Gaussian copula is not appropriate which again contradicts the assumptions of the CVMS model.

11.3.4 Regular vine market sector model

As seen in the previous section, the assumptions of the CVMS model are not appropriate for the given data set. Therefore, we construct an alternative model which exploits the more general structure of R-vines. Clearly, we expect that there are strong relationships between the returns of a stock and the sector it belongs to, here the respective national stock index. In contrast to the CVMS model, we thus model these dependencies first as well as the dependencies of the sectors to the market to take into account the joint driver of dependencies among sectors. If all remaining dependencies are then captured by Gaussian pair copulas in higher order trees (pairwise simplification at level 1), we speak of the *regular vine sector (RVS) model*. If however the dependencies to the market, here the Euro Stoxx 50, are also modeled conditionally on the respective sectors in the second R-vine tree before setting all pair copulas of higher order trees to bivariate Gaussian copulas (pairwise simplification at level 2), we call the model a *regular vine market sector (RVMS) model*. As in the CVMS model, we furthermore assume that sectors are independent conditioned on the market in the RVMS model. Moreover, note that the construction of higher order trees is not uniquely determined by the first or second tree in a RVS or RVMS model, respectively. We solve this problem by simply modeling the highest dependencies in each tree as we did before (see Section 3.2) and where "dependency" refers to a measure as discussed in Section 3.1. Examples of both models are given in the following.

Example 14 (RVS and RVMS models.) *Similar to Example 13 we consider returns* r_1^A, r_2^A, r_1^B, r_2^B, r_1^C *and* r_2^C *of stocks belonging to sectors A, B and C with returns* r_A, r_B *and* r_C, *respectively. Furthermore,* r_M *denotes the return of the market.*

Then the first R-vine tree in the RVS as well as the RVMS model is specified with appropriate pair copulas as

$$c_{1A,A}\left(F(r_1^A), F(r_A)\right) c_{2A,A}\left(F(r_2^A), F(r_A)\right) c_{1B,B}\left(F(r_1^B), F(r_B)\right)$$
$$\times c_{2B,B}\left(F(r_2^B), F(r_B)\right) c_{1C,C}\left(F(r_1^C), F(r_C)\right) c_{2C,C}\left(F(r_2^C), F(r_C)\right)$$
$$\times c_{A,M}\left(F(r_A), F(r_M)\right) c_{B,M}\left(F(r_B), F(r_M)\right) c_{C,M}\left(F(r_C), F(r_C)\right).$$

In the RVS model, all pair copulas of higher order trees are then set to bivariate Gaussian copulas. The RVMS model, on the other hand, also models the second R-vine tree under the independence assumption that sectors are independent conditionally on the market, i.e.,

$$c_{A,B|M}\left(F(r_A|r_M), F(r_B|r_M)\right) = 1,$$
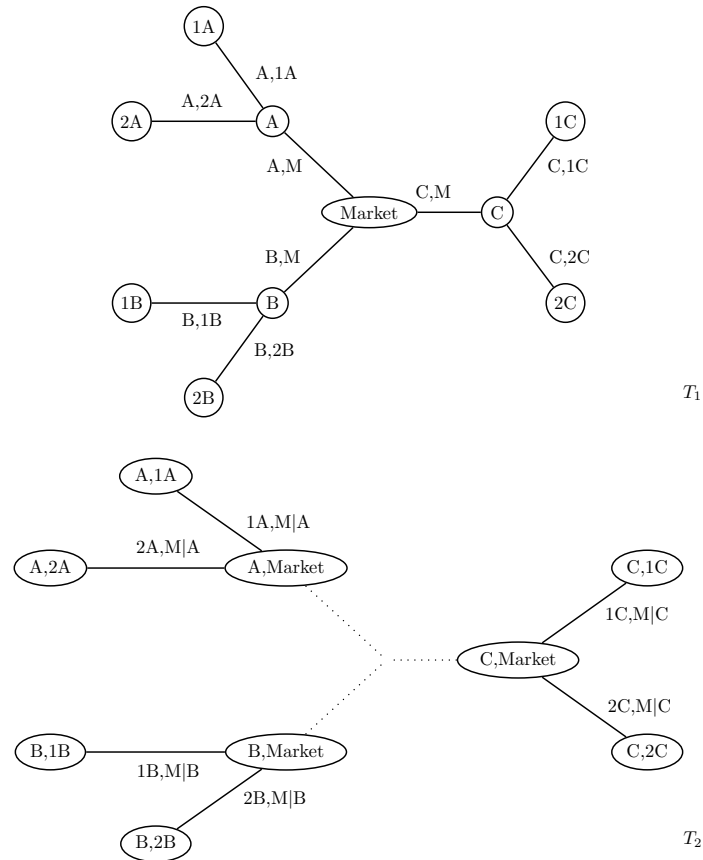
Figure 11.12: First and second tree of the RVMS model in Example 14.

and similarly for A and C as well as for B and C. Hence, the pair copulas of the second tree of the RVMS model are

$$
\begin{aligned}
& c_{1A,M|A}\left(F(r_1^A|r_A), F(r_M|r_A)\right) c_{2A,M|A}\left(F(r_2^A|r_A), F(r_M|r_A)\right) \\
& \times c_{1B,M|B}\left(F(r_1^B|r_B), F(r_M|r_B)\right) c_{2B,M|B}\left(F(r_2^B|r_B), F(r_M|r_B)\right) \\
& \times c_{1C,M|C}\left(F(r_1^C|r_C), F(r_M|r_C)\right) c_{2C,M|C}\left(F(r_2^C|r_C), F(r_M|r_C)\right).
\end{aligned}
$$

Note that without the above independence assumptions, we would have to choose a tree structure for the sector variables, since the pair copulas $c_{A,B|M}$, $c_{A,C|M}$ and $c_{B,C|M}$ cannot all be included in an R-vine at once. Due to the independence assumptions, the model is independent of this choice. All higher order trees of the RVMS model are then specified with Gaussian pair copulas.

Figure 11.12 shows the first tree of the RVS and the RVMS models as well as the second tree of the RVMS model. The dotted lines in the second tree illustrate the independence assumption for sectors conditionally on the market.

We now specify an RVS and an RVMS model for the Euro Stoxx 50 data. The first tree of both models is shown in the left panel of Figure 11.13, while the second tree of the RVMS model is illustrated in the right panel of Figure 11.13. To facilitate understanding

Figure 11.13: The first tree of the RVS and RVMS models (left panel) as well as an illustration of the second tree of the RVMS model (right panel) for the Euro Stoxx 50 data (cp. Example 14 and, in particular, Figure 11.12).

order	CVMS model	full C-vine model
1st	^STOXX50E	^STOXX50E
2nd	^FCHI	GLE.PA
3rd	^GDAXIP	^FCHI
4th	FTSEMIB.MI	^GDAXIP
5th	^IBEX	^IBEX
6th	^AEX	INGA.AS
7th	–	FTSEMIB.MI
$\vdots$	$\vdots$	$\vdots$
13th	–	^AEX
$\vdots$	$\vdots$	$\vdots$

Table 11.14: Order of the conditioning sets of the CVMS model and a full C-vine model for the Euro Stoxx 50 data.

of the model we omitted the edges between the sectors conditioned on the market which are modeled as independent anyway. Thus the second "tree" as displayed here is not a tree, since it is not connected (see Section 2.3).

As in the CVMS model, we investigate these independence assumptions. In the first five rows of Table 11.12, the p-values of bivariate independence tests between the national indices conditionally on the Euro Stoxx 50 are displayed (same independence assumptions regarding the sectors as in the CVMS model). Apparently, the assumption of pairwisely independent sectors is not true here (the hypothesis is rejected for 50% of the pairs). However, in contrast to the CVMS model, we can circumvent the problem, since not all bivariate copulas for two sectors conditioned on the market can be included in the model so that it remains a valid R-vine (*tree* structure!). If we choose the pair copulas ^AEX–^GDAXIP, ^GDAXIP–^FCHI, ^FCHI–FTSEMIB.MI and FTSEMIB.MI–^IBEX (which happen to be a D-vine structure with first tree ^AEX–^GDAXIP–^FCHI–FTSEMIB.MI–^IBEX), we obtain a valid density and the independence assumptions are satisfied. Remaining dependencies are captured by Gaussian pair copulas in higher order trees.

11.3.5   Comparison to regular and canonical vine specifications

In the following, we will now compare the CVMS model as well as the RVS and the RVMS models to adequate R- and C-vine models for the Euro Stoxx data. With regard to the latter, we are particularly interested in the order of the conditioning set identified by our model construction methods (see Chapter 3) and the joint simplification level determined by the procedures discussed in Chapters 7 and 8 and evaluated in Section 10.2.2. In R-vine models we investigate how well the dependence of the Euro Stoxx 50 data is modeled in comparison to the RVS, the RVMS and the C-vine models, i.e., we want to determine the loss when imposing more restrictive structures than a general R-vine structure.

As mentioned above, the first issue is the order of the conditioning set of appropriate C-vines for the Euro Stoxx 50 data. Hence, we specify a full C-vine model without simplification or truncation. The order of root nodes is given in Table 11.14 alongside the
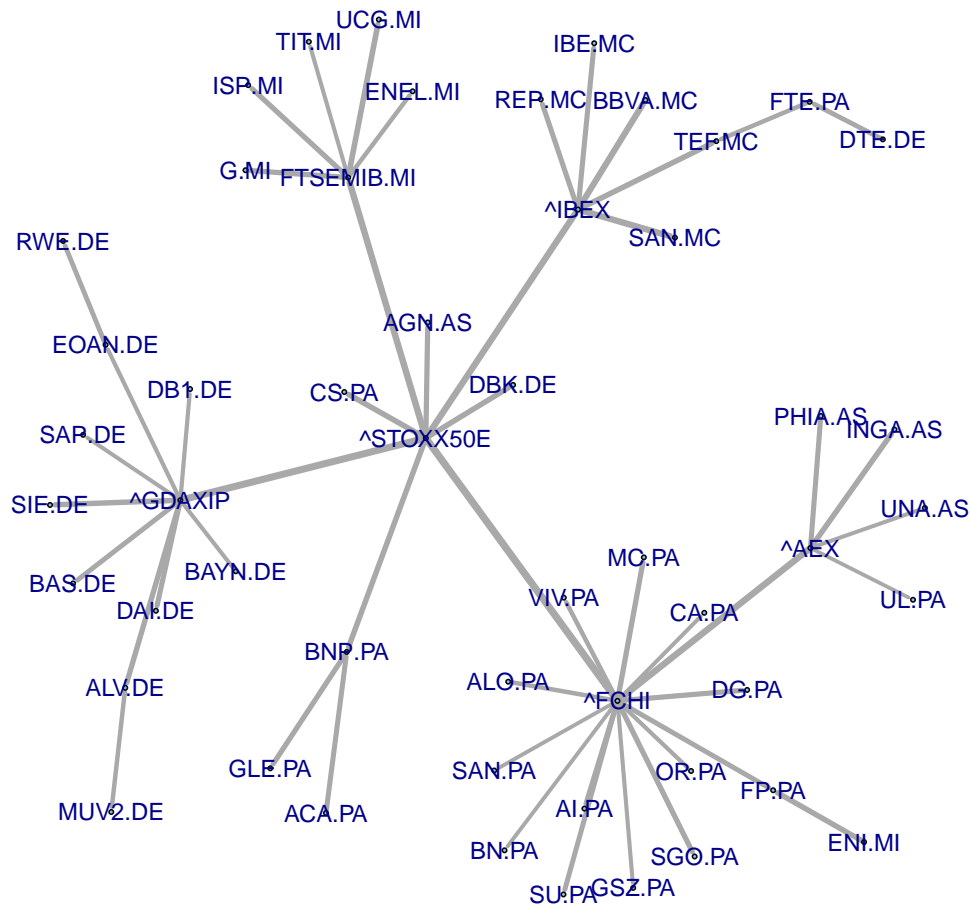
Figure 11.14: First tree of the full R-vine model for the Euro Stoxx 50 data.

corresponding order induced by our CVMS model, where the order of the variables 2 to 6 could also be chosen differently and no root nodes after tree T_6 are determined due to the joint simplification, since the modeling with the multivariate Gaussian copula does not require the specification of additional tree structures.

In terms of the order of the conditioning set, the CVMS model is apparently a good approximation, since the order identified by the full C-vine model is similar except for the stocks of the financial companies Société Générale (GLE.PA) and ING Groep (INGA.AS) which exhibit strong dependencies to other stock returns. Moreover, the AEX plays only a minor role due to its small size.

In the next step, we compare this "dependency ranking" to the dependencies determined by a full R-vine model for the Euro Stoxx 50 data. Using Kendall's τ as weights for the tree construction, we obtain the first R-vine tree as shown in Figure 11.14.

Clearly, the Euro Stoxx 50 as well as the national leading stock indices are identified as main dependency drivers, where the Euro Stoxx 50 can be identified as some kind of "root node" of the R-vine. Moreover, the national indices are directly connected to this root node, except for the Dutch AEX which is, probably due to its small size, linked to the French CAC 40. Compared to the first tree of the RVS and RVMS models (cp. Figure

11.13), there are also some connections and even clusters between stocks that deviate from the assumptions of these models. These connections will be discussed in more detail below. All in all, the dependence structure of this full R-vine confirms the assumptions of the CVMS model and, in particular, of the RVS and RVMS models.

As both the CVMS model and the RVS and RVMS models are simplified at a pre-specified level, we now investigate simplified R- and C-vine models. Moreover, we use our simplification procedures to determine the appropriate simplification level of a restricted R-vine model, where the first tree corresponds to the first tree of the RVS and RVMS models. Furthermore, we determine truncation levels for all models in order to evaluate where most of the dependencies are captured. The resulting models are shown in Table 11.15 and compared using AIC/BIC as well as Vuong tests with respect to the most general full R-vine model. Estimates are obtained by sequential estimation, since full maximum likelihood estimation is computationally extremely demanding for these 52-dimensional models and, as mentioned earlier and shown in Table 11.17 below as well as in Appendix B for the models discussed in the previous two sections, sequential estimates are usually rather close to those obtained by full maximum likelihood estimation, i.e., sufficient for basic model comparisons. Trees are constructed with Kendall's τ as weights.

As simplification/truncation procedures we concentrate on those based on the Vuong test, since they turned out to be the best-performing methods (see Chapter 10). We specify models using the Vuong test without correction and with Schwarz correction in order to obtain a rather parsimonious model on the one hand and a less parsimonious but nevertheless adequately simplified/truncated model on the other hand. The results of joint truncation and simplification are not reported, since a truncation level of 36 and a simplification level of 42 do not indicate a significant facilitation in model specification, where the levels are identified using the computationally faster tests based on Spearman's ρ and the Rosenblatt transformation, respectively (cp. Chapter 4). Finally, we also consider an alternative specification of the CVMS model: instead of jointly simplifying at level 6, we *pairwisely* simplify at this level and also perform bivariate independence tests for the pairs of transformed variables in each tree, i.e., in this way we obtain a much more parsimonious model. The resulting model is denoted as *CVMS-P* to indicate the pairwise simplification.

The main results of the different models shown in Table 11.15 can be summarized as follows.

(i) The full R-vine with pre-specified first tree and the full C-vine are rather close to the unrestricted full R-vine model. Considering the Vuong test without correction for the number of parameters, the models are equivalent. However, the C-vine model requires more parameters and therefore the null hypothesis of equivalent models is rejected if the Vuong test is corrected for the number of parameters.

(ii) The simplification levels of R-vines as identified by the procedures based on the Vuong test confirm the assumptions of the RVS and RVMS models. Note however that the trees in the more general R-vines may look differently and therefore model stronger dependencies than under the assumptions of the RVS and RVMS models.

(iii) Though simplification of C-vine does not confirm the assumptions of the CVMS

proc. type	procedure	trunc./ simpl. level	log likeli-hood	number of param.	AIC	BIC	Vuong stat.	V. stat. (Akaike corr.)	V. stat. (Schwarz corr.)
unrestricted R-vine	full model	-	30777.84	596	-60363.67	-57447.66	-	-	-
trunc.	Vuong	28	30410.01	548	-59724.02	-57042.86	11.07	9.62	6.09
	V.Schwarz	8	29060.04	341	-57438.08	-55769.69	16.82	14.32	8.21
simpl.	Vuong	4	30473.00	543	-59860.00	-57203.29	7.95	6.57	3.19
	V.Schwarz	2	30350.24	515	-59670.47	-57150.76	8.22	6.66	2.85
R-vine with T_1 as in RV(M)S	full model	-	30709.30	616	-60186.50	-57172.68	1.25*	1.61*	2.50
trunc.	Vuong	25	30392.70	563	-59659.50	-56904.93	6.33	5.78	4.46
	V.Schwarz	9	28904.00	370	-57068.00	-55257.70	16.22	14.27	9.48
simpl.	Vuong	3	30390.40	560	-59660.80	-56920.92	6.79	6.16	4.62
	V.Schwarz	2	30325.57	548	-59555.14	-56873.97	7.34	6.56	4.66
unrestricted C-vine	full model	-	30661.14	685	-59952.29	-56600.83	1.33*	2.34	4.81
trunc.	Vuong	35	30626.10	677	-59898.20	-56585.89	1.71*	2.62	4.86
	V.Schwarz	21	30201.80	601	-59201.60	-56261.12	6.02	6.08	6.20
simpl.	Vuong	2	30061.37	567	-58988.73	-56214.61	5.65	5.42	4.86
	V.Schwarz	1	30000.99	548	-58905.97	-56224.80	6.00	5.63	4.73
(market-)sector models	CVMS	6^a	30309.46	1212	-58194.93	-52265.05	4.75	10.99	26.27
	CVMS-P	6^a	29861.63	495	-58733.26	-56311.40	8.93	7.94	5.54
	RVS	1^a	29964.20	506	-58916.39	-56440.71	7.50	6.67	4.64
	RVMS	2^a	30126.28	535	-59182.55	-56564.99	6.52	5.91	4.41

Table 11.15: R- and C-vine specifications obtained from different procedures as well as (market-)sector models for the Euro Stoxx 50 data (sequential estimation). Test statistics indicated by "*" imply that the null hypothesis that full R-vine model and alternative model are equivalent cannot be rejected at the 5% level.

---

[a]Simplified by construction.

models, which assume a simplification level of 6. According to AIC/BIC and the Vuong tests with regard to the full R-vine model, the CVMS models are inferior to the pairwisely truncated C-vines as specified using the procedures based on the Vuong test.

(iv) The CVMS model of Heinen and Valdesogo (2009) requires a lot of parameters, since a 46-dimensional correlation matrix has to be fitted for the remaining dependency after C-vine tree 6. The pairwisely simplified CVMS model, on the other hand, is much more parsimonious and superior in terms of AIC/BIC as well as Vuong tests with correction for the number of parameters.

(v) As noted above, the CVMS model of Heinen and Valdesogo (2009) is specified without pairwise independence tests and hence the likelihood is higher than for the pairwisely simplified model. It is also higher than the likelihood of the RVS and RVMS models. This is however only due to the large number of parameters and therefore AIC/BIC as well as Vuong tests with correction for the number of parameters show that the RVMS model and even the RVS model are superior to the CVMS model of Heinen and Valdesogo (2009) and also slightly superior to the CVMS-P model, since they impose less restrictive independence assumptions which may not be satisfied.

(vi) Truncation of R- or C-vines for the Euro Stoxx 50 is apparently not sensible as long as we also test for bivariate independence for each pair of transformed variables. The identified truncation levels are either rather high, i.e., do not indicate a significant facilitation for inference, or rather low but do not perform well in terms of AIC/BIC and Vuong tests.

11.3.6   Value-at-Risk backtesting

As the RVMS model apparently fits the data rather well even in comparison to the fully specified R-vine model, we now examine both models more closely by checking their VaR forecasting accuracy as discussed in Section 9.3. For this we consider a testing period from 4/30/2010 to 9/16/2010 with 100 daily observations and simulate the VaR of the portfolio of the 46 Euro Stoxx 50 stocks in our data set with weights according to the index composition as of 2/8/2010. In order to do so, we proceed iteratively:

 (i) We specify ARMA(1,1)-GARCH(1,1)-models for the marginal time series with a moving window of 985 observations (corresponding to the above analysis).

 (ii) Then we fit a fully specified R-vine and the RVMS model to the transformed standardized residuals.

(iii) Using the estimated copula parameters we simulate 100,000 samples from both models. The samples are then transformed back by the cdf's of the standardized residuals and the portfolio value is computed as the weighted sum of the one-step-ahead forecasts obtained by plugging residuals and estimated ARMA-GARCH parameters into Equation (2.36).

(iv) Finally, we compute the 90%-VaR of the portfolio.

Note that we preliminarily decide to fit ARMA(1,1)-GARCH(1,1)-models to all marginal time series in contrast to the more detailed analysis before. This is done in order to limit the complexity of the approach and also approximately accurate according to Table 11.11. Moreover, we only use sequential estimation because of the high computational effort of fitting 100 R-vine models for the full model and the RVMS model, respectively. A confidence level of 10% is chosen in accordance to the rather short testing period.

In our simulations, the fully specified R-vine and the RVMS model both produced the same hit sequence with 13 VaR exceedances. At the 5% significance level the null hypothesis of unconditional coverage (9.4) with $\alpha = 0.1$ clearly cannot be rejected with a p-value of 0.337, while the independence hypothesis (9.5) is marginally rejected with a p-value of 0.047. The combined test of conditional coverage (9.6) however gives a p-value of 0.088, so that we can conclude that both models produce sufficiently accurate VaR forecasts. In particular, the RVMS model performs just as well as the fully specified R-vine model, but at the same time VaR forecasts are computed approximately 40% faster, i.e., the RVMS model is evidently more efficient.

11.3.7   Conclusion and interpretation

The CVMS model is not adequate for the data analyzed here, since the independence assumptions are not entirely satisfied and the model is clearly inferior to alternative models. In particular, the less restrictive RVMS model and even the simpler RVS model provide a good fit to the data and correspond to the results of more general R-vine specifications. The problems in the fit of the CVMS model may be due to the specific structure of the data (using national indices as sectors). We however think that the RVMS model is more flexible, less restrictive and even easier to specify, since the fit of a high-dimensional Gaussian copula with many correlation parameters is computationally demanding if we want to use, e.g., maximum likelihood estimation instead of the inversion of Kendall's τ or Spearman's ρ, and since it leads to over-specified models with too many parameters.

The dependence structure of the full R-vine model, which is similar to that of the RVS and RVMS models, corresponds very well to the economical intuition. The tree construction method, which is based on Kendall's τ to capture the most important dependencies in the transformed residuals (cp. Chapter 3), identifies the national indices as clusters of dependent stocks, where all indices themselves are dependent on the Euro Stoxx 50 except for the Dutch AEX, which is closely related to the French CAC 40 due to its small size (cp. Figure 11.14). Moreover, there are some deviations from the first tree as assumed by the RVS and RVMS models. These form interesting "sub-clusters" of similarly behaving stock returns:

(i) Financial companies are linked directly to the Euro Stoxx 50: AEGON (AGN.AS), AXA (CS.PA), Deutsche Bank (DBK.DE) and BNP Parisbas (BNP.PA) with Société Générale (GLE.PA) and Crédit Agricole (ACA.PA).

(ii) German insurance companies: Munich Re (MUV2.DE) more strongly depends on Allianz (ALV.DE) than on the German DAX.

model	tree	Π	N	t	C	G	F	J	BB1	BB7	RC	RG
R-vine	1	0	0	50	0	0	0	0	1	0	0	0
	2	2	2	28	3	0	12	0	0	0	1	2
	3	23	3	17	1	1	4	0	0	0	0	0
	4	24	4	10	1	2	5	0	0	0	0	2
	5	31	3	3	0	0	6	0	0	0	2	2
RVMS	1	0	0	48	0	0	1	0	2	0	0	0
	2	26	2	14	0	1	5	0	0	0	2	0
	3	0	49	0	0	0	0	0	0	0	0	0
	4	16	32	0	0	0	0	0	0	0	0	0
	5	22	25	0	0	0	0	0	0	0	0	0

Table 11.16: Copula types of the first five trees of the full R-vine and RVMS models. Notation of copula families: Π = independence, N = Gaussian, C = Clayton, G = Gumbel, F = Frank, J = Joe, RC = rotated Clayton, RG = rotated Gumbel.

(iii) German energy companies: RWE (RWE.DE) also more strongly depends on E.ON (EOAN.DE) than on the German DAX.

(iv) Oil companies: Eni (ENI.MI) is closely related to Total (FP.PA).

(v) Unibail-Rodamco (UL.PA), which is a French-Dutch company, is more strongly dependent on the Dutch AEX than on the French CAC 40.

(vi) Telecommunications companies are closely linked and more dependent on each other than on the respective national indices: Telefónica (TEF.MC), France Télécom (FTE.PA) and Deutsche Telekom (DTE.DE).

Due to this clustered first tree, the simplification results indicate that the more complex dependencies among residuals are all covered within these clusters, since simplification at levels 2, 3 or 4 seems to be appropriate and gives more parsimonious models compared to full model specifications. The main dependency drivers are identified by the C-vine as the Euro Stoxx 50 (as expected), Société Générale, the CAC 40, the DAX, the IBEX, ING Groep and the FTSE MIB. The central role of the financial companies Société Générale and ING Groep corresponds to the fact that 28.2% of the Euro Stoxx 50 are composed of financial companies (including insurance companies). Moreover, the importance of the CAC 40 and the DAX is due to the high number of French and German stocks represented in the Euro Stoxx 50 (weights of 35.3% and 28.1%, respectively). Given theses clearly identified dependencies, C-vine models can also be regarded as alternatives to R-vine specifications but require more parameters, since dependencies are less flexibly identified and modeled.

In contrast to simplification, truncation does not lead to a significant gain in the model specifications. Apparently the dependencies among the stocks are too high to be neglected after having modeled the most important relationships.

The copula types of the first five trees of the full R-vine model and the RVMS model are shown in Table 11.16. Dependencies to the national indices and to the Euro Stoxx

model	log lik. (seq. est.)	log lik. (full MLE)	number of param.	BIC	Vuong stat.	V. stat. (Schwarz)
R-vine	30777.84	30880.57	596	-57653.12	-	-
RVMS	30126.28	30168.06	535	-56648.56	7.40	5.22

Table 11.17: Log likelihoods of the full R-vine and RVMS models obtained by sequential ("seq. est.") and full maximum likelihood estimation ("full MLE") as well as BIC's and test statistics of Vuong tests between both models (with and without Schwarz correction).

50 are mainly captured by heavy tailed t copulas, while higher order trees are modeled with different copulas for different types of dependencies. Due to the pre-specified first two trees in the RVMS model, there is more dependency left in the third tree and hence no variable pair is determined to be independent in contrast to the full R-vine model, where the number of chosen independence copulas is increasing after the first tree. This is also true after the third tree in the RVMS model.

Table 11.17 compares the sequentially estimated log likelihoods of the full R-vine and the RVMS model to those obtained by full maximum likelihood estimation. The sequential estimates are evidently good starting values, since the log likelihoods only change by clearly less than 1%. BIC and Vuong tests of the now fully estimated models confirm the previous results that the RVMS model provides a reasonably good fit of the data compared to the fully specified R-vine model. It also efficiently and rather accurately forecasts the VaR of the Euro Stoxx 50 data as seen above.

All in all, the RVMS model illustrates the flexible and powerful structure of R-vines to construct models for high-dimensional data. Its application to the Euro Stoxx 50 data gives insightful information about dependency structures among important European stocks.

Chapter 12

Conclusion and outlook

In this thesis, we considered different aspects of the appropriate specification of R-vine models. First, we discussed the construction of R-vine trees and proposed suitable graph theoretical algorithms and dependence measures, such as the new measures of exceedance Kendall's τ or Hu dependence, which are required for construction according to the paradigm that most dependencies are captured in the first trees, where the identification of the "best" construction method for specific circumstances is the subject of future research; we usually use Kendall's τ as dependence measure. Then, we turned to the issue of copula goodness-of-fit and independence testing – an area where much research is still being conducted in. Furthermore, we considered model selection criteria such as the AIC and BIC as well as the Vuong test. For a range of ten one- and two-parametric copula families, we computed and compared Kullback-Leibler information criteria and evaluated bivariate copula selection criteria such as the AIC, a scoring method based on the Vuong test and copula goodness-of-fit tests, where we found that the AIC is the best-performing criterion.

The main part of this thesis deals with the problem of accurately determining whether R-vines can be pairwisely truncated or simplified with Gaussian pair copulas after a certain tree. Different procedures were proposed and evaluated in extensive simulation studies, which showed that in particular the procedures based on the Vuong test perform well. However, it still has to be explored under which conditions truncated R-vines converge to the true model. Composite likelihood methods might be useful to investigate this consistency issue.

We also considered the special case of joint C-vine simplification, where remaining dependencies are captured by a multivariate Gaussian copula. Here, simplification levels can be determined using multivariate goodness-of-fit tests which are however computationally quite demanding. In a simulation study, we therefore compared this method to the procedures discussed before for general R-vines and found that these can reasonably be used instead. Moreover, we extended the results with regard to joint C-vine simplification to special cases of R-vines and proved a corresponding theorem.

After having discussed appropriate model evaluation criteria, we considered three different applications of the procedures discussed in this thesis. First, we examined a data set of nine exchange rates to the US Dollar and determined joint and pairwise truncation and simplification levels of an adequate C-vine specification. Second, a 19-dimensional fi-

nancial data set from Norway was analyzed with truncated and simplified R-vine models and in particular with respect to tail behavior. Finally, we developed a flexible R-vine model for stock market dependencies, the RVMS model, and compared it to alternative models based on a 52-dimensional data set of national stock indices and 46 stocks of the Euro Stoxx 50. Although the RVMS model showed quite good results, further research is to be done on investigating the model more closely in different applications and improve it by including time dependencies and possibly loosening independence assumptions. In particular, VaR backtesting has to be performed for longer testing periods.

To sum it up, the methods discussed in this thesis allow to efficiently construct R-vine models even in higher dimensions and under time or resource restrictions – not least because the easily computed AIC turned out to be the best performing criterion for bivariate copula selection. As such, R-vine models constitute a flexible and powerful class of high-dimensional dependency models, which is available for a wide range of applications.

Appendix A

Contour and scatter plots of bivariate copula families

In the following, contour and scatter plots of the bivariate copula families discussed in Section 2.1.3 are presented. Parameters are determined according to different choices of Kendall's τ using the relationships shown in Table 2.1, where only the Gaussian, t and Frank as well as the rotated Clayton and Gumbel copulas exhibit negative dependence. The latter two copulas are shown in the respective figures of the Clayton and Gumbel copulas for negative dependence.

The degrees of freedom of the t copula are chosen as four and eight. For the BB1 and BB7 copulas, we choose different upper tail dependence values so that the two copula parameters are uniquely determined.
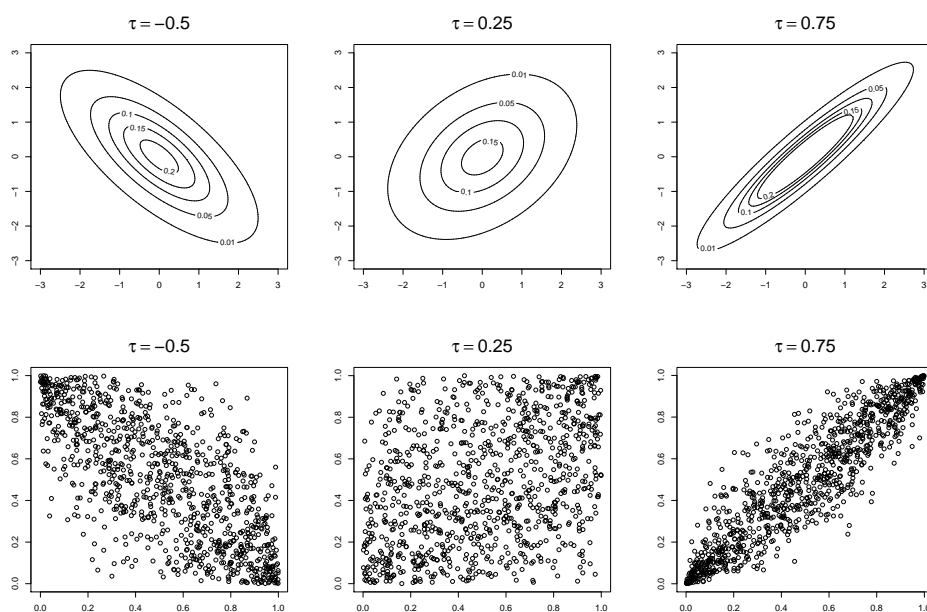
Figure A.1: Contour and scatter plots of the Gaussian copula for three choices of Kendall's τ.
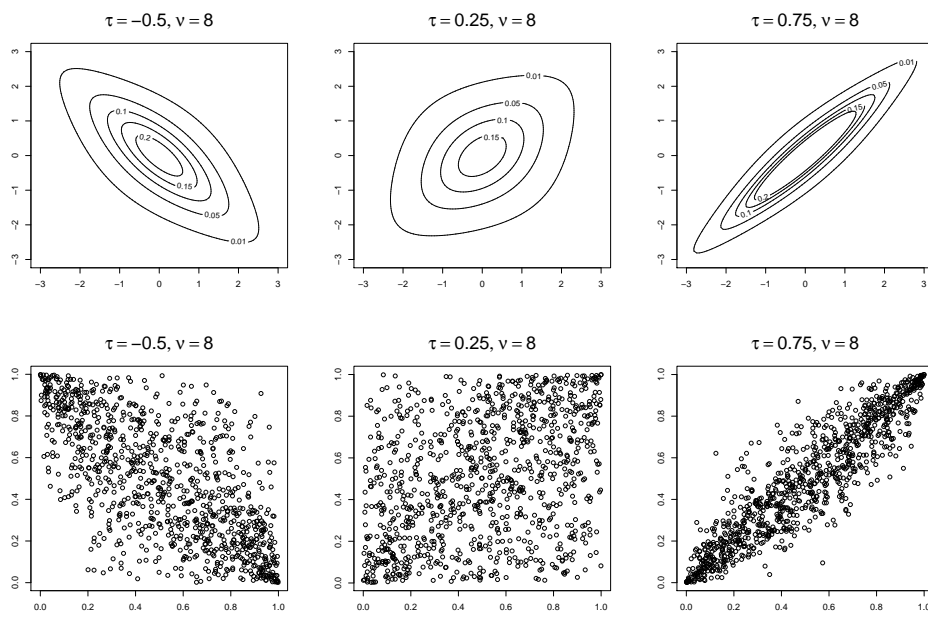
Figure A.2: Contour and scatter plots of the t copula for three choices of Kendall's τ and eight degrees of freedom.
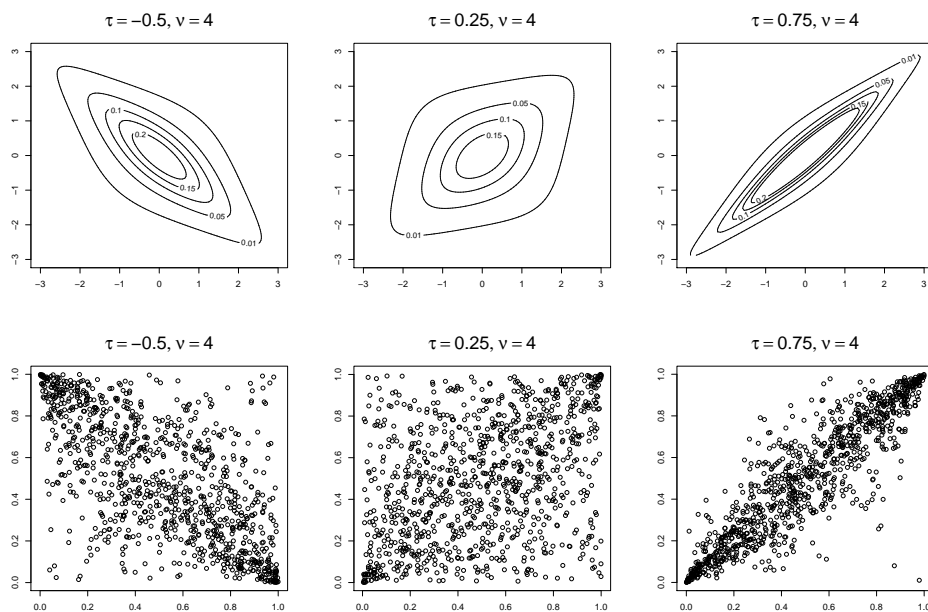
Figure A.3: Contour and scatter plots of the t copula for three choices of Kendall's τ and four degrees of freedom.
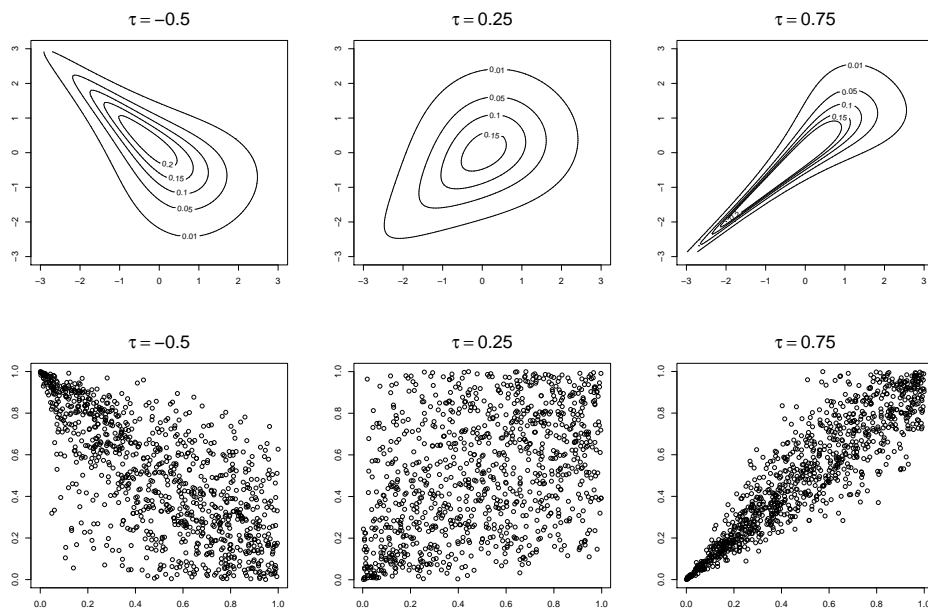
Figure A.4: Contour and scatter plots of the Clayton and rotated Clayton copulas for three choices of Kendall's τ (left panel: rotated Clayton, middle and right panel: Clayton).
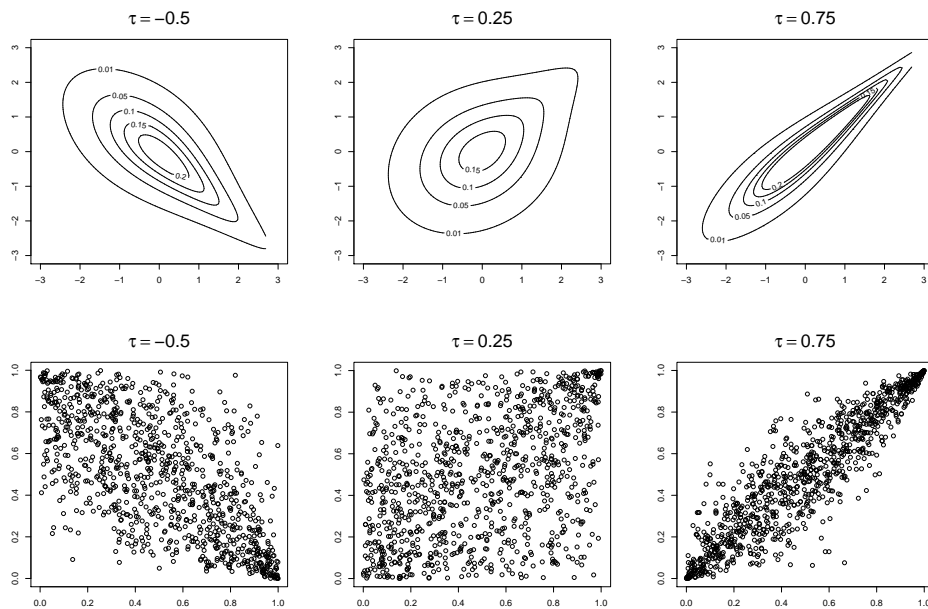
Figure A.5: Contour and scatter plots of the Gumbel and rotated Gumbel copulas for three choices of Kendall's τ (left panel: rotated Gumbel, middle and right panel: Gumbel).
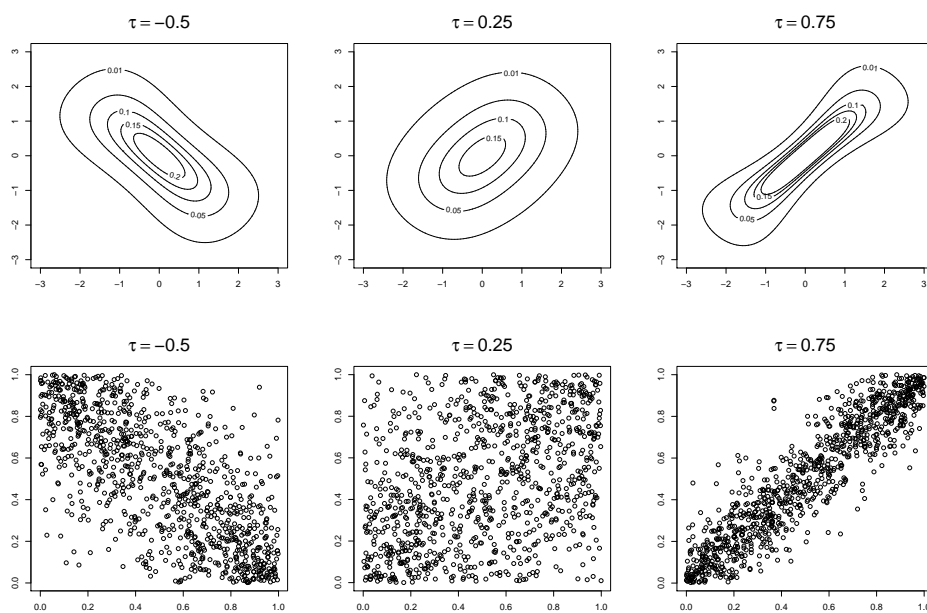
Figure A.6: Contour and scatter plots of the Frank copula for three choices of Kendall's τ.
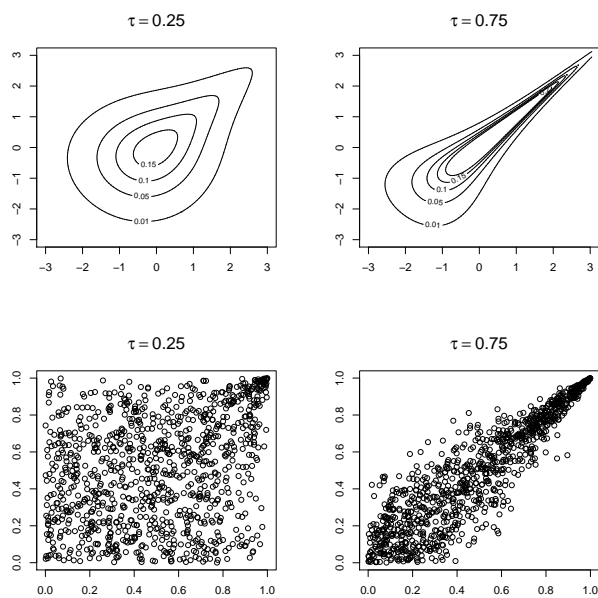
Figure A.7: Contour and scatter plots of the Joe copula for two choices of Kendall's τ.
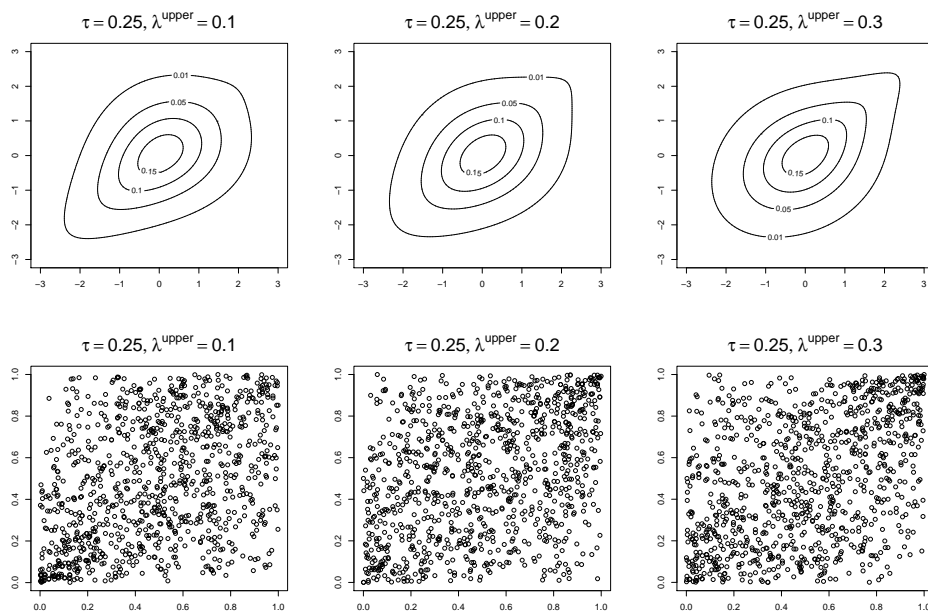
Figure A.8: Contour and scatter plots of the BB1 copula for Kendall's $\tau = 0.25$ and different choices of upper tail dependence.
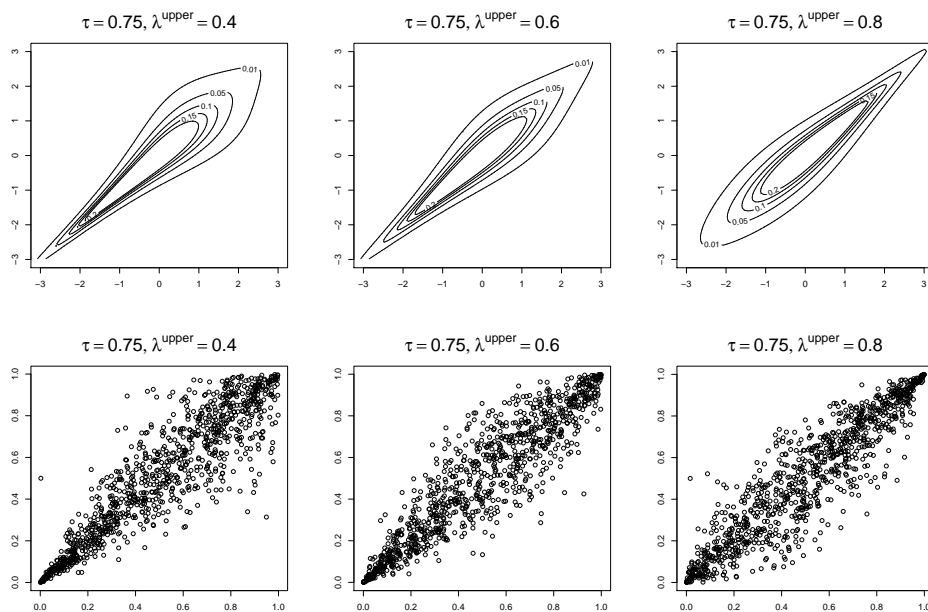
Figure A.9: Contour and scatter plots of the BB1 copula for Kendall's $\tau = 0.75$ and different choices of upper tail dependence.
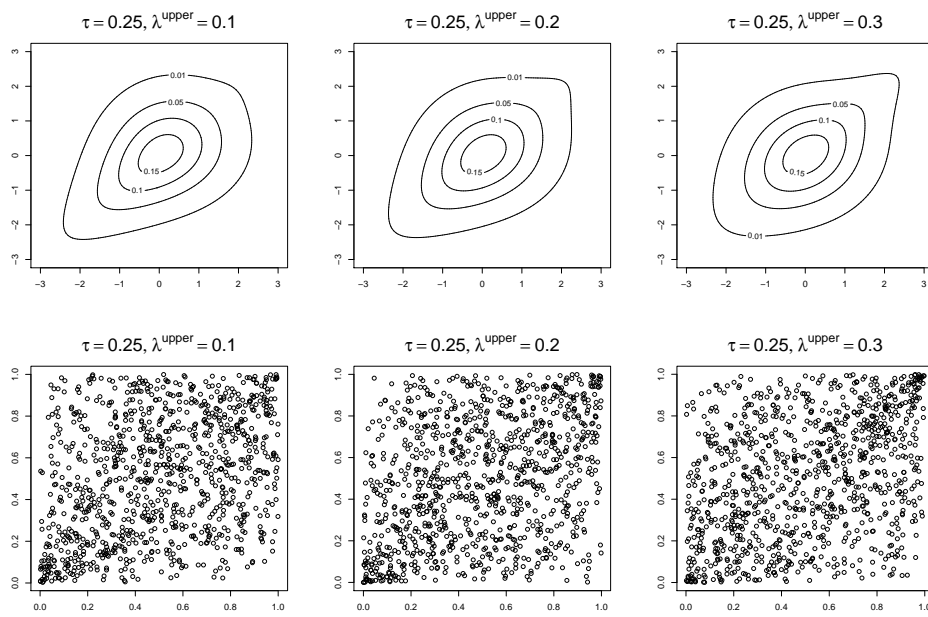
Figure A.10: Contour and scatter plots of the BB7 copula for Kendall's $\tau = 0.25$ and different choices of upper tail dependence.
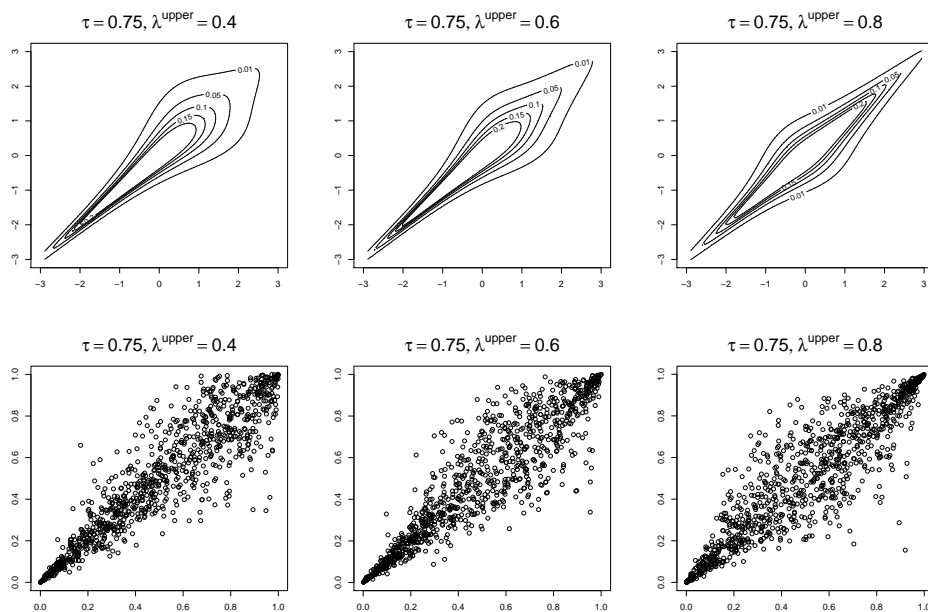
Figure A.11: Contour and scatter plots of the BB7 copula for Kendall's $\tau = 0.75$ and different choices of upper tail dependence.

Appendix B

Additional information on applications

B.1 Exchange rates

Here, R-vine copula type and parameter matrices of the C-vine models discussed in Section 11.1 (reduced model selection setting with bivariate independence tests) are presented. In particular, we consider the fully specified C-vine model, the pairwisely simplified 1 level C-vine model $\mathcal{M}_S(1)$ as well as the jointly simplified 2 level R-vine model $\mathcal{M}_J(2)$. The matrices of the truncated model $\mathcal{M}_T(4)$ are not shown, since they can be obtained from the copula type and parameter matrices of the full model by replacing all entries of rows after the truncation level $K = 4$ with zeros.

In the following, M denotes the RVM of a model (cp. 2.4.1; entries as in Table 11.1), T the copula type matrix and P_1 and P_2 the parameter matrices, where P_1 shows the estimated parameters of all one-parameter copula families as well as the correlation parameters of the t copula and the parameters θ of the BB1 and BB7 copulas, while P_2 shows the degrees of freedom of the t copula as well as the parameters δ of the BB1 and BB7 copulas. The copula types in T are denoted as follows:

0	1	2	3	4	5
independence	Gaussian	t	Clayton	Gumbel	Frank

6	7	8	13	14
Joe	BB1	BB7	rotated Clayton	rotated Gumbel

Furthermore, Kendall's τ matrices τ_M are shown, i.e., matrices which display the Kendall's τ's corresponding to the respective copula type and parameters (cp. Table 2.1). However, these matrices should not be compared directly, since the models are built differently!

The order of the conditioning sets of the C-vines can easily be obtained by reading the first column of a copula type matrix bottom-up. E.g., the first column of the copula type matrix of the full C-vine model corresponds to the order given in Table 11.3.

Correlation parameters of the multivariate Gaussian copula in the jointly simplified model $\mathcal{M}_J(2)$ have been transformed to partial correlations similarly to Algorithm 12 but

	proc. type	procedure	trunc./ simpl. level	log likelihood (full MLE)	log likelihood (seq. est.)
COMPLETE (no bivariate indep. tests)		full model	-	2243.73	2237.62
		t copulas	-	2240.73	2237.05
	trunc.	Vuong	6	2243.43	2237.29
		V.Schwarz	3	2181.97	2176.36
		AIC/BIC	6	2243.43	2237.29
	simpl.	Vuong	1	2204.50	2202.36
		V.Schwarz	1	2204.50	2202.36
		AIC/BIC	-	2243.73	2237.62
REDUCED (bivariate indep. tests)		full model	-	2220.51	2215.35
		t copulas	-	2220.21	2216.72
	trunc.	Vuong	6	2220.51	2215.35
		V.Schwarz	4	2199.87	2194.86
		AIC/BIC	6	2220.51	2215.35
	simpl.	Vuong	1	2197.43	2195.24
		V.Schwarz	1	2197.43	2195.24
		AIC/BIC	6	2220.51	2215.35

Table B.1: Comparison of log likelihoods computed via sequential estimation ("seq. est.") and full maximum likelihood estimation ("full MLE") for C-vine specifications of the exchange rates data set obtained from different procedures.

the other way round and according to C-vine construction principles, i.e., after having determined the correlation matrix at level 2 we also constructed C-vine trees $T_3, ..., T_8$ as discussed in Section 3.2 with Kendall's τ as weights and then set the pair copulas to bivariate Gaussian and computed the corresponding partial correlation parameters from the correlation matrix of the multivariate Gaussian copula.

Before now stating the copula type and parameter matrices, we briefly consider the sequential estimates which are directly obtained from C-vine construction as starting values for full maximum likelihood estimation. Table B.1 compares the estimated log likelihoods of C-vine specifications and shows that sequential estimates are evidently quite good starting values as also discussed in Dißmann (2010).

After the copula type and parameter matrices, graphical model evaluation criteria of the simplified/truncated C-vine models for the exchange rates data set are presented as discussed in Section 11.1.1. Here, scatter plots with standard normal margins, empirical copula distributions as well as copula Q-Q plots for the pairwisely simplified 1 level C-vine $\mathcal{M}_S(1)$ and the jointly simplified 2 level C-vine $\mathcal{M}_J(2)$ as well as for the fully specified model are shown.

Full C-vine model

$$
M = \begin{pmatrix}
6 \\
7 & 2 \\
2 & 7 & 3 \\
3 & 3 & 7 & 7 \\
9 & 9 & 9 & 9 & 5 \\
5 & 5 & 5 & 5 & 9 & 8 \\
8 & 8 & 8 & 8 & 8 & 9 & 4 \\
4 & 4 & 4 & 4 & 4 & 4 & 9 & 9 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1
\end{pmatrix}, \quad
T = \begin{pmatrix}
0 \\
0 & 0 \\
0 & 1 & 5 \\
5 & 0 & 0 & 0 \\
0 & 0 & 4 & 5 & 1 \\
0 & 0 & 5 & 2 & 5 & 1 \\
0 & 7 & 2 & 14 & 2 & 14 & 5 \\
3 & 2 & 7 & 2 & 2 & 2 & 2 & 1
\end{pmatrix}
$$

$$
P_1 = \begin{pmatrix}
0.00 \\
0.00 & 0.00 \\
0.00 & 0.08 & -0.76 \\
0.85 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 1.11 & -0.95 & 0.08 \\
0.00 & 0.00 & -0.78 & 0.49 & -1.61 & -0.12 \\
0.00 & 0.17 & 0.34 & -1.10 & 0.37 & -1.13 & 1.15 \\
0.17 & 0.73 & 0.27 & 0.36 & 0.32 & 0.88 & 0.64 & 0.26
\end{pmatrix}
$$

$$
P_2 = \begin{pmatrix}
0.00 \\
0.00 & 0.00 \\
0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 11.57 & 0.00 & 0.00 \\
0.00 & 1.10 & 12.69 & 0.00 & 14.04 & 0.00 & 0.00 \\
0.00 & 9.27 & 1.26 & 5.90 & 6.83 & 4.61 & 15.39 & 0.00
\end{pmatrix}
$$

$$
\tau_M = \begin{pmatrix}
0.00 \\
0.00 & 0.00 \\
0.00 & 0.05 & -0.08 \\
0.09 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.10 & -0.10 & 0.05 \\
0.00 & 0.00 & -0.09 & 0.33 & -0.17 & -0.08 \\
0.00 & 0.16 & 0.22 & -0.09 & 0.24 & -0.12 & 0.13 \\
0.08 & 0.52 & 0.30 & 0.24 & 0.20 & 0.69 & 0.44 & 0.17
\end{pmatrix}
$$

Pairwisely simplified level 1 C-vine model $\mathcal{M}_S(1)$

$$
M = \begin{pmatrix}
6 & & & & & & & & \\
7 & 2 & & & & & & & \\
2 & 7 & 3 & & & & & & \\
3 & 3 & 7 & 7 & & & & & \\
9 & 9 & 9 & 9 & 5 & & & & \\
5 & 5 & 5 & 5 & 9 & 8 & & & \\
8 & 8 & 8 & 8 & 8 & 9 & 4 & & \\
4 & 4 & 4 & 4 & 4 & 4 & 9 & 9 & \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1
\end{pmatrix}, \quad
T = \begin{pmatrix}
0 & & & & & & & \\
0 & 0 & & & & & & \\
0 & 1 & 1 & & & & & \\
1 & 0 & 0 & 0 & & & & \\
0 & 0 & 1 & 1 & 1 & & & \\
0 & 0 & 1 & 1 & 1 & 1 & & \\
0 & 1 & 1 & 1 & 1 & 1 & 1 & \\
3 & 2 & 7 & 2 & 2 & 2 & 2 & 1
\end{pmatrix}
$$

$$
P_1 = \begin{pmatrix}
0.00 & & & & & & & \\
0.00 & 0.00 & & & & & & \\
0.00 & 0.09 & -0.12 & & & & & \\
0.12 & 0.00 & 0.00 & 0.00 & & & & \\
0.00 & 0.00 & 0.15 & -0.14 & 0.08 & & & \\
0.00 & 0.00 & -0.10 & 0.48 & -0.24 & -0.12 & & \\
0.00 & 0.26 & 0.35 & -0.21 & 0.38 & -0.25 & 0.18 & \\
0.17 & 0.72 & 0.26 & 0.36 & 0.31 & 0.89 & 0.64 & 0.26
\end{pmatrix}
$$

$$
P_2 = \begin{pmatrix}
0.00 & & & & & & & \\
0.00 & 0.00 & & & & & & \\
0.00 & 0.00 & 0.00 & & & & & \\
0.00 & 0.00 & 0.00 & 0.00 & & & & \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & & & \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & & \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & \\
0.00 & 8.78 & 1.26 & 5.76 & 7.01 & 4.71 & 13.73 & 0.00
\end{pmatrix}
$$

$$
\tau_M = \begin{pmatrix}
0.00 & & & & & & & \\
0.00 & 0.00 & & & & & & \\
0.00 & 0.06 & -0.08 & & & & & \\
0.08 & 0.00 & 0.00 & 0.00 & & & & \\
0.00 & 0.00 & 0.10 & -0.09 & 0.05 & & & \\
0.00 & 0.00 & -0.07 & 0.32 & -0.15 & -0.07 & & \\
0.00 & 0.17 & 0.23 & -0.14 & 0.25 & -0.16 & 0.11 & \\
0.08 & 0.52 & 0.30 & 0.23 & 0.20 & 0.69 & 0.44 & 0.17
\end{pmatrix}
$$

Jointly simplified level 2 C-vine model $\mathcal{M}_J(2)$

$$M = \begin{pmatrix} 5 \\ 6 & 2 \\ 2 & 6 & 6 \\ 9 & 9 & 9 & 3 \\ 3 & 3 & 3 & 9 & 7 \\ 7 & 7 & 7 & 7 & 9 & 8 \\ 8 & 8 & 8 & 8 & 8 & 9 & 4 \\ 4 & 4 & 4 & 4 & 4 & 4 & 9 & 9 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}, \quad T = \begin{pmatrix} 1 \\ 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 7 & 0 & 2 & 14 & 14 & 5 \\ 2 & 2 & 3 & 7 & 2 & 2 & 2 & 1 \end{pmatrix}$$

$$P_1 = \begin{pmatrix} 0.01 \\ -0.07 & -0.00 \\ 0.08 & 0.05 & 0.12 \\ 0.14 & 0.07 & 0.01 & 0.06 \\ -0.13 & -0.03 & 0.00 & -0.14 & 0.00 \\ -0.23 & -0.00 & 0.04 & -0.10 & 0.48 & -0.11 \\ 0.37 & 0.17 & 0.00 & 0.35 & -1.11 & -1.14 & 1.14 \\ 0.32 & 0.73 & 0.18 & 0.25 & 0.36 & 0.89 & 0.64 & 0.26 \end{pmatrix}$$

$$P_2 = \begin{pmatrix} 0.00 \\ 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 14.32 & 1.10 & 0.00 & 13.03 & 0.00 & 0.00 & 0.00 \\ 7.54 & 9.24 & 0.00 & 1.27 & 5.75 & 4.56 & 15.72 & 0.00 \end{pmatrix}$$

$$\tau_M = \begin{pmatrix} 0.01 \\ -0.04 & -0.00 \\ 0.05 & 0.03 & 0.08 \\ 0.09 & 0.04 & 0.01 & 0.04 \\ -0.09 & -0.02 & 0.00 & -0.09 & 0.00 \\ -0.15 & -0.00 & 0.02 & -0.07 & 0.32 & -0.07 \\ 0.24 & 0.16 & 0.00 & 0.23 & -0.10 & -0.12 & 0.13 \\ 0.21 & 0.52 & 0.08 & 0.30 & 0.23 & 0.69 & 0.44 & 0.17 \end{pmatrix}$$
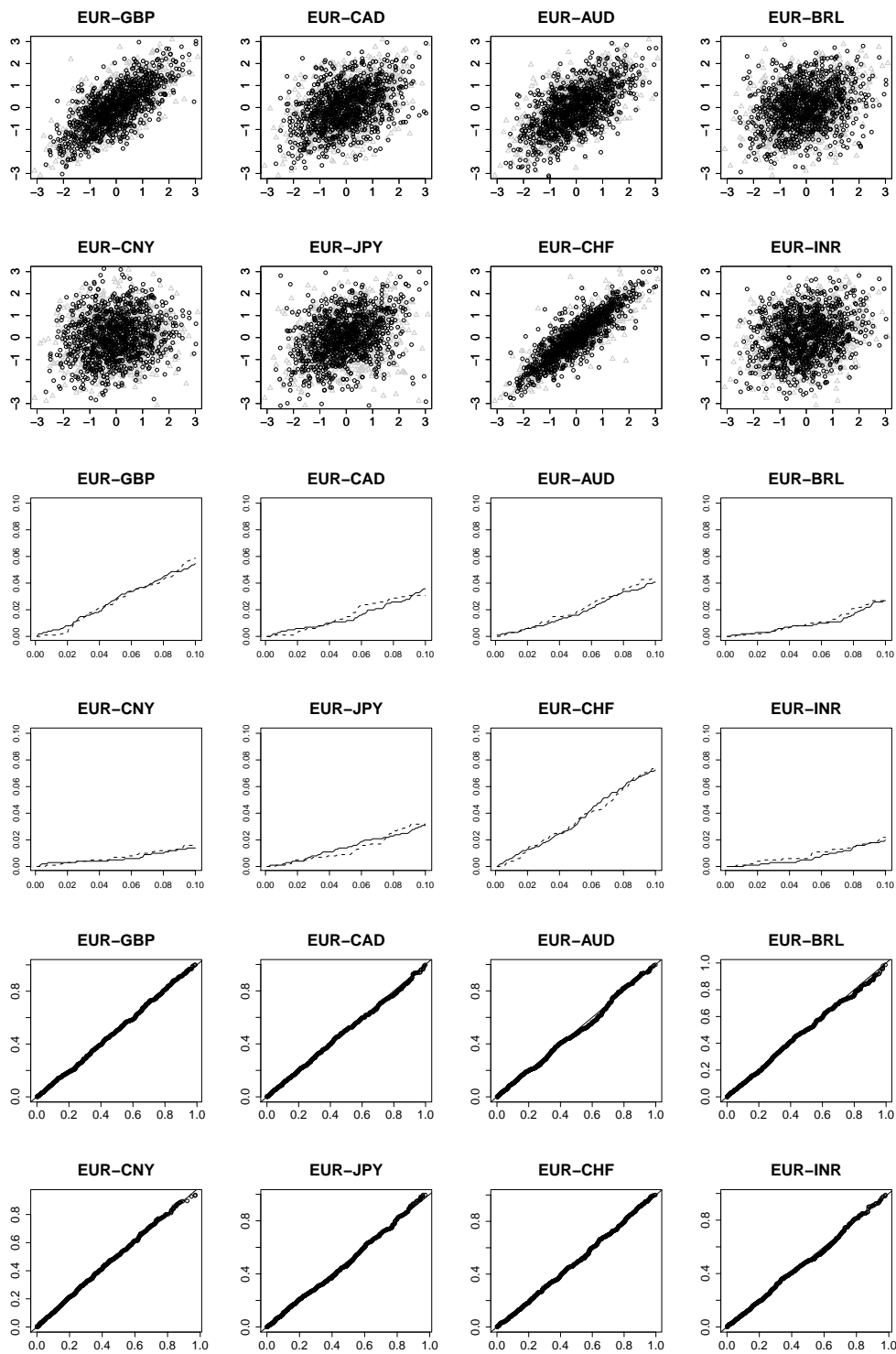
Figure B.1: **Full** model for the exchange rates data set: scatter plots with standard normal margins, empirical copula distribution functions (lower tail) and copula Q-Q plots for observed (gray triangles, solid lines) versus simulated data (black circles, dashed lines).
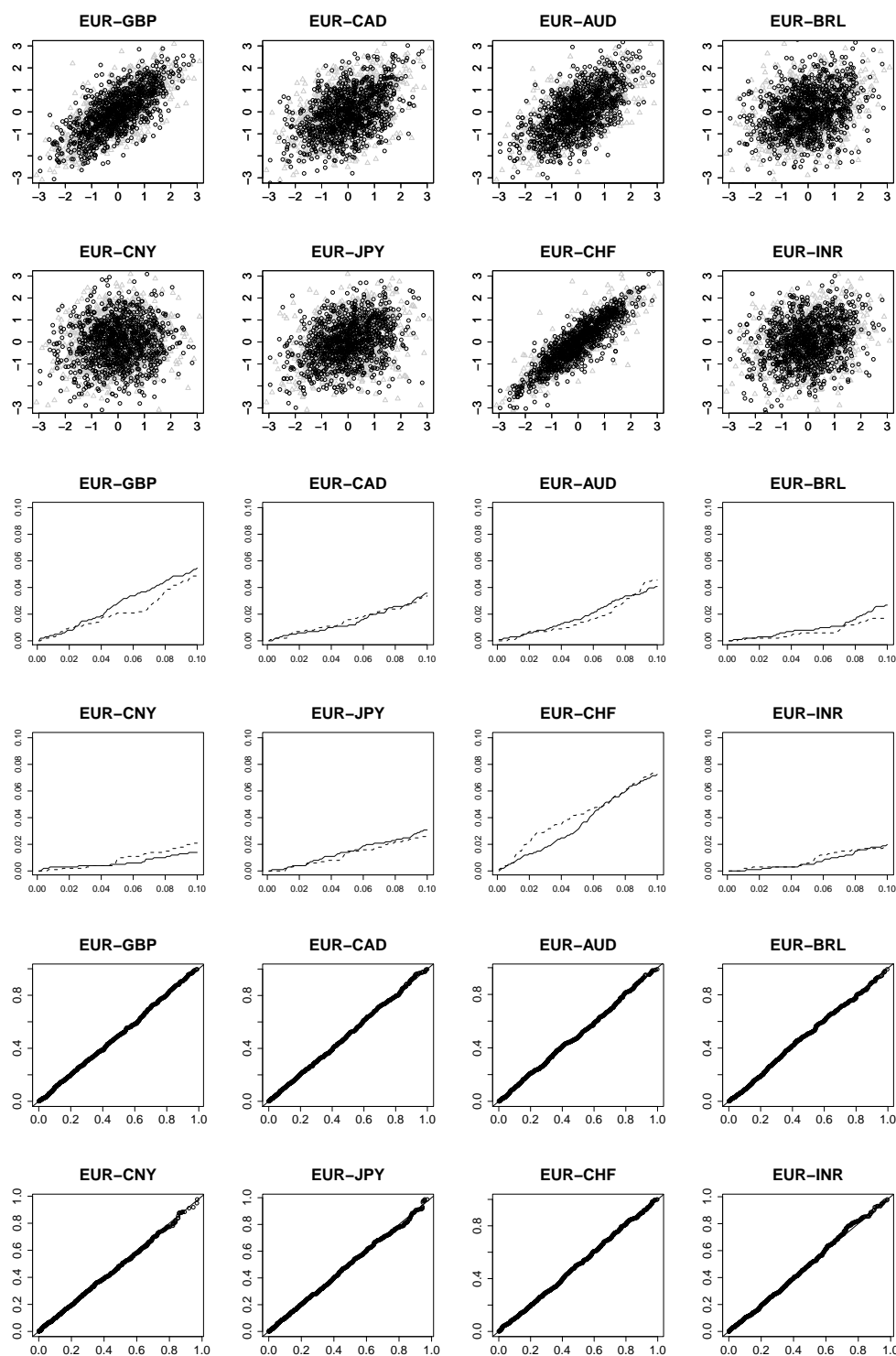
Figure B.2: **Pairwisely simplified 1 level** model $\mathcal{M}_S(1)$ for the exchange rates data set: scatter plots with standard normal margins, empirical copula distribution functions (lower tail) and copula Q-Q plots for observed (gray triangles, solid lines) versus simulated data (black circles, dashed lines).
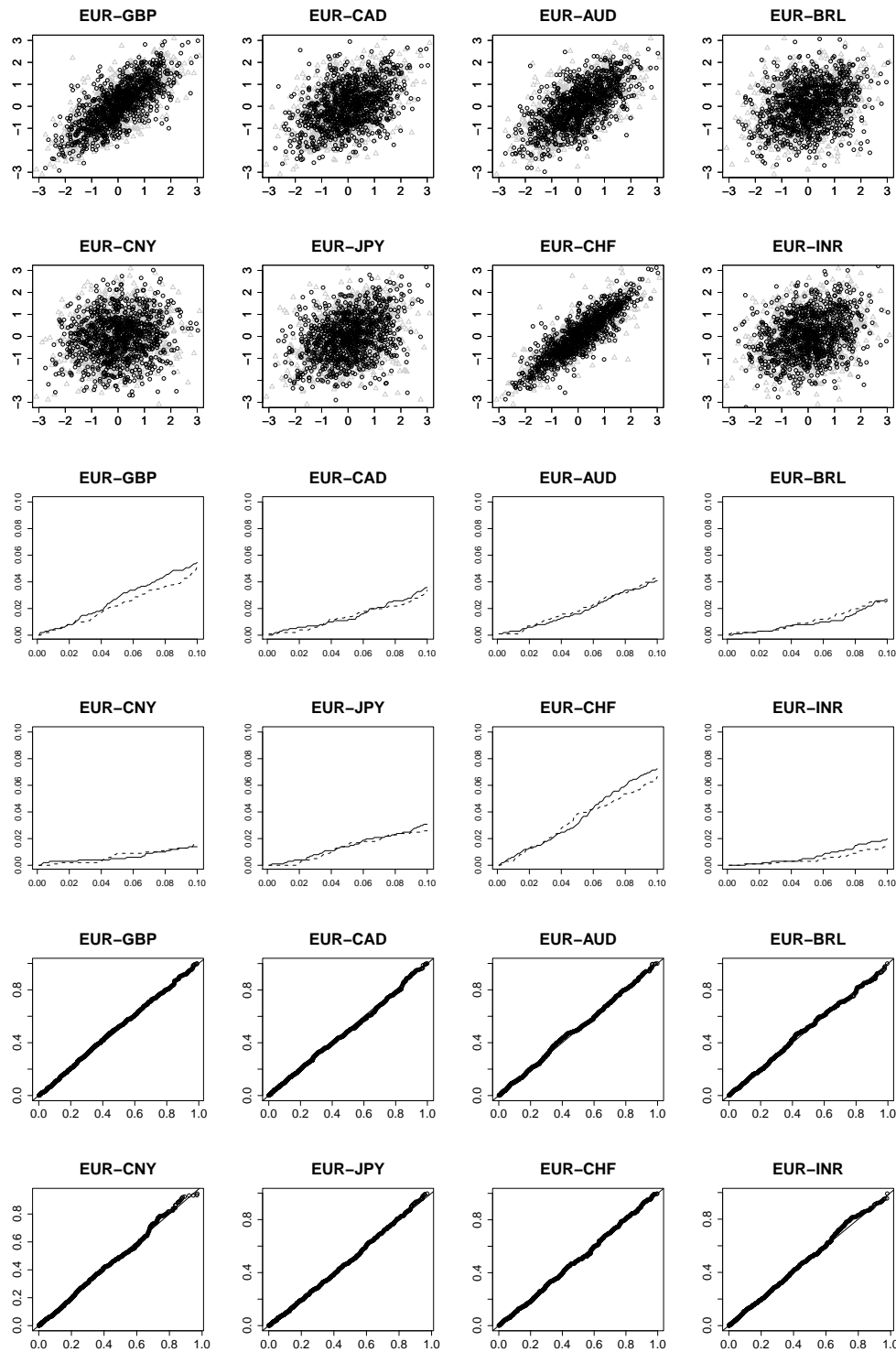
Figure B.3: **Jointly simplified 2 level** model $\mathcal{M}_J(2)$ for the exchange rates data set: scatter plots with standard normal margins, empirical copula distribution functions (lower tail) and copula Q-Q plots for observed (gray triangles, solid lines) versus simulated data (black circles, dashed lines).

B.2   Financial data from Norway

As in the previous section, we report R-vine copula type and parameter matrices of the models discussed in Section 11.2 and built using Kendall's τ as weights. We consider the fully specified R-vine model as well as the pairwisely simplified 2 level R-vine model $\mathcal{M}_S(2)$, while matrices of the truncated models can be obtained by replacing all entries of rows after the truncation level K in the matrices of the full model with zeros. All notations are the same as above. Note that the entries of the RVM's range from 1 to 19. Since there is no variable V6, V7 corresponds to entry 6, V8 to 7, and so on.

The comparison of the log likelihoods obtained by sequential and by full maximum likelihood estimation (see Table B.2) again shows that the sequential estimates are good starting values for full maximum likelihood estimation if R-vines are built using Kendall's τ as weights. If however exceedance Kendall's τ is used, the difference between estimated log likelihoods is larger. This is probably due to the different focus of model building, i.e., the focus on tail behavior.

weights	proc. type	procedure	trunc./ simpl. level	log likelihood (full MLE)	log likelihood (seq. est.)
Kendall's τ		full model	-	6390.75	6387.39
		t copulas	-	6378.33	6374.92
	trunc.	Vuong	6	6274.47	6272.77
		V.Schwarz	4	6234.05	6232.62
		AIC/BIC	6	6274.47	6272.77
	simpl.	Vuong	2	6350.09	6348.76
		V.Schwarz	2	6350.09	6348.76
		AIC/BIC	6	6373.80	6371.74
exceedance Kendall's τ (max.)		full model	-	6397.87	6359.46
		t copulas	-	6412.22	6388.59
	trunc.	Vuong	13	6041.28	6014.28
		V.Schwarz	9	5891.65	5866.84
		AIC/BIC	15	6046.79	6019.59
	simpl.	Vuong	1	6237.48	6220.44
		V.Schwarz	1	6237.48	6220.44
		AIC/BIC	13	6373.80	6338.26

Table B.2: Comparison of log likelihoods computed via sequential estimation ("seq. est.") and full maximum likelihood estimation ("full MLE") for R-vine specifications of the Norwegian financial data set obtained from different procedures.

Full R-vine model

$$
M = \begin{pmatrix}
18 \\
10 & 3 \\
6 & 10 & 5 \\
3 & 6 & 10 & 4 \\
5 & 1 & 6 & 10 & 10 \\
4 & 19 & 1 & 6 & 1 & 1 \\
2 & 17 & 19 & 1 & 2 & 6 & 2 \\
13 & 16 & 17 & 19 & 13 & 2 & 6 & 13 \\
8 & 8 & 16 & 17 & 19 & 13 & 19 & 6 & 19 \\
15 & 15 & 8 & 16 & 17 & 8 & 17 & 19 & 6 & 17 \\
12 & 12 & 15 & 8 & 16 & 15 & 16 & 17 & 8 & 6 & 16 \\
7 & 7 & 12 & 15 & 11 & 12 & 8 & 16 & 15 & 8 & 6 & 11 \\
14 & 14 & 7 & 12 & 9 & 7 & 15 & 8 & 12 & 15 & 8 & 6 & 9 \\
11 & 11 & 14 & 7 & 14 & 14 & 12 & 15 & 7 & 12 & 15 & 8 & 6 & 14 \\
9 & 9 & 11 & 14 & 7 & 11 & 7 & 12 & 14 & 7 & 12 & 15 & 8 & 6 & 7 \\
19 & 13 & 9 & 11 & 6 & 9 & 14 & 7 & 11 & 14 & 7 & 12 & 15 & 8 & 6 & 8 \\
16 & 4 & 13 & 9 & 15 & 19 & 11 & 14 & 9 & 11 & 14 & 7 & 12 & 15 & 8 & 6 & 6 \\
17 & 2 & 4 & 13 & 12 & 16 & 9 & 11 & 17 & 9 & 11 & 14 & 7 & 12 & 15 & 15 & 12 & 15 \\
1 & 5 & 2 & 2 & 8 & 17 & 13 & 9 & 16 & 16 & 9 & 9 & 14 & 7 & 12 & 12 & 15 & 12 & 12
\end{pmatrix}
$$

$$
T = \begin{pmatrix}
5 \\
0 & 0 \\
0 & 0 & 2 \\
0 & 0 & 0 & 0 \\
0 & 4 & 0 & 0 & 0 \\
0 & 1 & 2 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 2 & 13 \\
0 & 0 & 0 & 0 & 0 & 0 & 14 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 1 \\
0 & 0 & 14 & 0 & 0 & 0 & 2 & 1 & 5 & 0 & 0 & 0 & 2 & 0 \\
1 & 2 & 0 & 2 & 0 & 0 & 1 & 0 & 0 & 4 & 0 & 0 & 2 & 0 & 0 \\
0 & 4 & 5 & 2 & 5 & 0 & 0 & 0 & 14 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 4 & 2 & 2 & 5 & 1 & 2 & 2 & 3 & 1 & 2 & 0 & 0 & 0 & 5 & 2 & 5 \\
1 & 7 & 1 & 2 & 2 & 1 & 2 & 2 & 7 & 7 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 8
\end{pmatrix}
$$

$$
P_1 =
\begin{bmatrix}
0.37 & 0.00 & 0.08 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & -0.16 & 0.00 & 0.00 & 0.00 & 1.28 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.07 & 0.00 & 0.00 & -0.06 & -0.82 \\
0.00 & 0.00 & 0.00 & 0.00 & -0.09 & 0.00 & -0.21 & 0.27 & -0.11 & 0.00 & 0.00 & 0.00 & 0.08 & 0.00 & 0.57 & -0.24 & -0.33 \\
0.00 & 1.04 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & -0.07 & 0.00 & 0.00 & 0.06 & 0.00 & -1.07 & 0.00 & 0.00 & 0.00 & 1.05 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & -0.10 & 0.06 & 0.56 & 1.04 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & -1.05 & 0.00 & 0.00 & 0.00 & -0.15 & 0.00 & -1.06 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.07 & 0.26 & 0.52 & -0.19 & -0.81 & 0.08 & -0.27 & -0.18 & 0.22 & 0.15 & 0.11 & 0.57 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.14 & 1.16 & 0.20 & 0.53 & -1.05 & 0.08 & -0.51 & -0.58 & 0.98 & 0.47 & 0.25 & 0.53 & 0.64 & 0.96 & -0.84 & -0.24 & -0.33 \\
0.29 & 0.23 & 0.53 & 0.67 & 0.28 & 0.66 & -0.51 & -0.58 & 0.98 & 0.47 & 0.25 & 0.53 & 0.64 & 0.96 & -0.84 & 8.02 & 0.43 \\
\end{bmatrix}
$$

$$
P_2 =
\begin{bmatrix}
0.00 & 0.00 & 12.01 & 0.00 & 0.00 & 0.00 & 14.53 & 63.35 & 0.00 & 0.00 & 0.00 & 0.00 & 13.45 & 0.00 & 0.00 & 16.42 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 29.04 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 9.40 & 0.00 & 0.00 & 12.71 & 8.02 \\
0.00 & 16.84 & 0.00 & 12.40 & 0.00 & 13.56 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 5.01 & 0.00 & 0.43 \\
0.00 & 0.00 & 26.48 & 10.26 & 8.64 & 21.52 & 9.48 & 0.00 & 0.00 & 10.46 & 6.25 & 7.13 & 3.04 & & & & \\
1.53 & 0.00 & 9.13 & 6.92 & 12.01 & 6.85 & 1.61 & 1.22 & 12.30 & & & & & & & & \\
\end{bmatrix}
$$

$$\tau_M = \begin{pmatrix}
0.04 \\
0.00 & 0.00 \\
0.00 & 0.00 & 0.05 \\
0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.04 & 0.00 & 0.00 & 0.00 \\
0.00 & -0.05 & 0.06 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & -0.06 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & -0.13 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.04 & 0.00 & 0.00 & 0.17 & -0.05 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & -0.06 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.05 & 0.00 & -0.10 \\
0.00 & 0.00 & -0.05 & 0.00 & 0.00 & 0.00 & -0.06 & 0.04 & 0.06 & 0.04 & 0.00 & 0.00 & 0.05 & 0.00 \\
0.04 & 0.17 & 0.00 & -0.12 & 0.00 & 0.00 & -0.10 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.05 & 0.00 & 0.00 \\
0.00 & 0.12 & 0.06 & 0.08 & -0.09 & 0.00 & 0.00 & 0.00 & -0.05 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.09 & 0.14 & 0.13 & 0.36 & -0.12 & 0.05 & -0.17 & -0.12 & 0.10 & 0.10 & 0.07 & 0.00 & 0.00 & 0.00 & 0.06 & -0.04 & -0.09 \\
0.19 & 0.42 & 0.36 & 0.47 & 0.18 & 0.46 & -0.34 & -0.40 & 0.58 & 0.34 & 0.16 & 0.36 & 0.44 & 0.82 & -0.63 & -0.15 & -0.21 & 0.27
\end{pmatrix}$$

Pairwisely simplified level 2 R-vine model $\mathcal{M}_S(2)$

$$
M = \begin{pmatrix}
3 \\
18 & 5 \\
10 & 18 & 4 \\
6 & 10 & 18 & 2 \\
8 & 6 & 10 & 18 & 10 \\
1 & 8 & 6 & 10 & 18 & 6 \\
19 & 1 & 8 & 6 & 1 & 18 & 8 \\
17 & 19 & 1 & 8 & 19 & 1 & 18 & 13 \\
16 & 17 & 19 & 1 & 17 & 19 & 1 & 18 & 15 \\
15 & 16 & 17 & 19 & 16 & 17 & 19 & 1 & 18 & 12 \\
12 & 15 & 16 & 17 & 13 & 16 & 17 & 19 & 1 & 18 & 7 \\
7 & 12 & 15 & 16 & 11 & 13 & 16 & 17 & 19 & 1 & 18 & 14 \\
14 & 7 & 12 & 15 & 9 & 11 & 13 & 16 & 17 & 19 & 1 & 18 & 11 \\
11 & 14 & 7 & 12 & 14 & 9 & 11 & 15 & 16 & 17 & 19 & 1 & 18 & 9 \\
9 & 11 & 14 & 7 & 7 & 14 & 9 & 12 & 11 & 16 & 17 & 19 & 1 & 18 & 18 \\
13 & 9 & 11 & 14 & 6 & 7 & 14 & 7 & 9 & 11 & 16 & 17 & 19 & 1 & 19 & 1 \\
4 & 13 & 9 & 11 & 15 & 8 & 7 & 14 & 14 & 9 & 11 & 16 & 17 & 19 & 16 & 19 & 17 \\
2 & 4 & 13 & 9 & 12 & 12 & 15 & 11 & 7 & 14 & 9 & 11 & 16 & 17 & 17 & 16 & 19 & 19 \\
5 & 2 & 2 & 13 & 8 & 15 & 12 & 9 & 12 & 7 & 14 & 9 & 9 & 16 & 1 & 17 & 16 & 16 & 16
\end{pmatrix}
$$

$$
T = \begin{pmatrix}
0 \\
0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\
1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
4 & 2 & 2 & 2 & 5 & 5 & 2 & 2 & 5 & 0 & 0 & 0 & 2 & 1 & 1 & 1 & 3 \\
7 & 1 & 2 & 2 & 2 & 2 & 2 & 2 & 8 & 2 & 2 & 2 & 2 & 2 & 1 & 1 & 7 & 7
\end{pmatrix}
$$

$$P_1 =$$

$$
\begin{pmatrix}
0.00 \\
0.00 & 0.00 \\
0.00 & 0.08 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.05 \\
0.07 & 0.09 & 0.00 & 0.00 & 0.00 & 0.00 \\
-0.07 & 0.00 & 0.00 & -0.21 & 0.06 & -0.05 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & -0.10 & -0.10 & 0.00 & 0.00 & 0.26 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.08 & 0.00 & 0.00 & 0.00 & 0.00 & 0.07 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.08 & 0.00 & 0.00 \\
-0.07 & -0.07 & 0.00 & -0.10 & 0.00 & -0.15 & 0.06 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & -0.19 & -0.14 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.08 & 0.00 & 0.00 \\
0.26 & 0.08 & 0.12 & 0.00 & -0.13 & 0.00 & 0.00 & 0.00 & 0.59 & 0.00 & 0.00 & 0.05 & 0.00 & -0.13 & 0.07 \\
0.22 & 0.18 & 0.53 & -0.25 & -1.05 & -0.82 & -0.06 & -0.17 & 0.08 & 0.00 & 0.00 & 0.00 & 0.12 & 0.15 & 0.00 & 0.08 & 0.22 \\
1.16 & 0.53 & 0.68 & -0.51 & 0.27 & -0.33 & -0.24 & -0.59 & 1.28 & -0.84 & 0.96 & 0.64 & 0.53 & 0.26 & 0.30 & 0.66 & 0.46 & 0.96
\end{pmatrix}
$$

$$P_2 =$$

$$
\begin{pmatrix}
0.00 \\
0.00 & 0.00 \\
0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 8.56 & 18.47 & 0.00 & 0.00 & 15.98 & 8.93 & 0.00 & 0.00 & 0.00 & 9.59 & 0.00 & 0.00 & 0.00 \\
0.00 & 17.32 & 9.49 & 10.43 & 6.24 & 8.15 & 12.05 & 7.24 & 0.42 & 4.99 & 3.03 & 7.02 & 6.38 & 13.00 & 0.00 & 1.23 & 0.00 \\
1.54 & 0.00 & & & & & & & & & & & & & & & 1.62
\end{pmatrix}
$$

$$
\tau_M =
\begin{pmatrix}
0.00 \\
0.00 & 0.00 \\
0.00 & 0.05 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.03 \\
0.05 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
-0.04 & 0.06 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & -0.13 & 0.04 & -0.03 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & -0.06 & -0.06 & 0.00 & 0.00 & 0.17 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.05 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & -0.10 & 0.04 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
-0.05 & 0.00 & 0.00 & -0.07 & 0.00 & 0.00 & 0.00 & 0.00 & 0.04 & 0.00 & 0.00 & 0.05 & 0.00 \\
0.00 & -0.04 & 0.00 & -0.09 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.03 & 0.00 & 0.00 \\
0.17 & 0.00 & -0.12 & 0.00 & 0.00 & 0.00 & 0.00 & 0.04 & 0.05 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.04 \\
0.14 & 0.05 & 0.08 & 0.00 & -0.08 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & -0.08 & 0.00 & 0.00 \\
0.14 & 0.12 & 0.36 & -0.16 & -0.12 & -0.09 & -0.04 & -0.11 & 0.07 & 0.00 & 0.00 & 0.00 & 0.07 & 0.10 & 0.09 & 0.05 & 0.10 \\
0.41 & 0.36 & 0.47 & -0.34 & 0.18 & -0.21 & -0.15 & -0.40 & 0.27 & -0.63 & 0.82 & 0.44 & 0.36 & 0.17 & 0.19 & 0.46 & 0.34 & 0.58
\end{pmatrix}
$$

Bibliography

Aas, K., C. Czado, A. Frigessi, and H. Bakken (2009). Pair-copula constructions of multiple dependence. *Insurance: Mathematics and Economics 44*(2), 182–198.

Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. In B. N. Petrov and F. Csaki (Eds.), *Proceedings of the Second International Symposium on Information Theory Budapest*, Akademiai Kiado, pp. 267–281.

Andersson, J. (2001). On the normal inverse Gaussian stochastic volatility model. *Journal of Business and Economic Statistics 19*(1), 44–54.

Ang, A. and J. Chen (2002). Asymmetric correlations of equity portfolios. *Journal of Financial Economics 63*, 443–494.

Applegate, D., R. Bixby, V. Chvátal, and W. Cook (2006). *The Traveling Salesman Problem: A Computational Study*. Princeton: Princeton University Press.

Bedford, T. and R. M. Cooke (2001). Probability density decomposition for conditionally dependent random variables modeled by vines. *Annals of Mathematics and Artificial intelligence 32*, 245–268.

Bedford, T. and R. M. Cooke (2002). Vines - a new graphical model for dependent random variables. *Annals of Statistics 30*, 1031–1068.

Belgorodski, N. (2010). Selecting pair-copula families for regular vines with application to the multivariate analysis of european stock market indices. Master's thesis, Technische Universität München.

Berg, D. (2009). Copula goodness-of-fit testing: an overview and power comparison. *The European Journal of Finance 15*(7), 675–701.

Bollerslev, T. (1986). Generalized autogressive conditional heteroskedasticity. *Journal of Econometrics 31*, 307–327.

Bollerslev, T. (1987). A conditionally heteroskedastic time series model for speculative prices and rates of return. *The Review of Economics and Statistics 69*(3), 542–547.

Breymann, W., A. Dias, and P. Embrechts (2003). Dependence structures for multivariate high-frequency data in finance. *Quantitative Finance 1*, 1–14.

Brockwell, P. J. and R. A. Davis (1991). *Time Series: Theory and Methods* (2nd ed.). New York: Springer.

Christoffersen, P. F. (1998). Evaluating interval forecasts. *International Economic Review 39*, 841–862.

Czado, C. (2010). Pair-copula constructions of multivariate copulas. In P. Jaworski, F. Durante, W. Härdle, and T. Rychlik (Eds.), *Copula Theory and Its Applications.* Berlin: Springer.

Deheuvels, P. (1979). La fonction de dépendance empirique et ses propriétés. Un test non paramétrique d'indépendance. *Bulletin de la Classe des Sciences, V. Série, Académie Royale de Belgique 65*, 274–292.

Deheuvels, P. (1981). An asymptotic decomposition for multivariate distribution-free tests of independence. *Journal of Multivariate Analysis 11*(1), 102–113.

Demarta, S. and A. J. McNeil (2005). The t copula and related copulas. *International Statistical Review 73*(1), 111–129.

Diestel, R. (2006). *Graph Theory* (3rd ed.). Berlin: Springer.

Dißmann, J. (2010). Statistical inference for regular vines and application. Master's thesis, Technische Universität München.

Dobrić, J. and F. Schmid (2005). Nonparametric estimation of the lower tail dependence λ_l in bivariate copulas. *Journal of Applied Statistics 32*(4), 387–407.

Durbin, J. and M. Knott (1972). Components of Cramér-von Mises statistics. Part I. *Journal of the Royal Statistical Society B 34*(2), 290–307.

Fermanian, J.-D. (2005). Goodness-of-fit tests for copulas. *Journal of Multivariate Analysis 95*, 119–152.

Fernandez, C. and M. F. Steel (1998). On Bayesian modeling of fat tails and skewness. *Journal of the American Statistical Association 93*, 359–371.

Fréchet, M. R. (1958). Remarques au sujet de la note précédente. *Comptes Rendus de l'Académie des Sciences, Series I, Mathematics 246*, 2719–2720.

Garfinkel, R. S. (1986). Motivation and modeling. In E. Lawler, A. R. J. Lenstra, and D. Shmoys (Eds.), *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*, Chapter 2, pp. 17–36. Chichester: John Wiley.

Genest, C. and A.-C. Favre (2007). Everything you always wanted to know about copula modeling but were afraid to ask. *Journal of Hydrologic Engineering 12*, 347–368.

Genest, C., K. Ghoudi, and L.-P. Rivest (1995). A semiparametric estimation procedure of dependence parameters in multivariate families of distributions. *Biometrika 82*, 543–552.

Genest, C., J.-F. Quessy, and B. Rémillard (2007). Asymptotic local efficiency of Cramér-von Mises tests for multivariate independence. *The Annals of Statistics 35*, 166–191.

Genest, C. and B. Rémillard (2004). Tests for independence and randomness based on the empirical copula process. *Test 13*(2), 335–369.

Genest, C. and B. Rémillard (2008). Validity of the parametric bootstrap for goodness-of-fit testing in semiparametric models. *Annales de l'Institut Henri Poincaré: Probabilités et Statistiques 44*, 1096–1127.

Genest, C., B. Rémillard, and D. Beaudoin (2009). Goodness-of-fit tests for copulas: a review and a power study. *Insurance: Mathematics and Economics 44*, 199–213.

Genest, C. and B. J. M. Werker (2002). Conditions for the asymptotic semiparametric efficiency of an omnibus estimator of dependence parameters in copula models. In C. M. Cuadras, J. Fortiana, and J. A. Rodríguez Lallena (Eds.), *Distributions with Given Marginals and Statistical Modelling*, pp. 103–112. Dordrecht: Kluwer.

Goldfarb, D. and A. U. Idnani (1983). A numerically stable dual method for solving strictly convex quadratic programs. *Mathematical Programming 27*(1), 1–33.

Grønneberg, S. and N. L. Hjort (2008). The copula information criterion. Statistical Research Report No. 7, Department of Mathematics, University of Oslo.

Hansen, B. E. (1994). Autoregressive conditional density estimation. *International Economic Review 35*, 705–730.

Heinen, A. and A. Valdesogo (2009). Asymmetric CAPM dependence for large dimensions: the canonical vine autoregressive model. CORE discussion papers 2009069, Université catholique de Louvain, Center for Operations Research and Econometrics (CORE).

Hobæk Haff, I., K. Aas, and A. Frigessi (2010). On the simplified pair-copula construction - simply useful or too simplistic? *Journal of Multivariate Analysis 101*(5), 1296–1310.

Hofmann, M. and C. Czado (2010). Assessing the VaR of a portfolio using D-vine copula based multivariate GARCH models. *Journal of Statistical Software*, submitted.

Hu, L. (2006). Dependence patterns across financial markets: a mixed copula approach. *Applied Financial Economics 16*(10), 717–729.

Huang, X. (1992). *Statistics of bivariate extreme values*. Ph. D. thesis, Erasmus University Rotterdam, Tinbergen Institute. Research Series 22.

Joe, H. (1996). Families of m-variate distributions with given margins and $m(m-1)/2$ bivariate dependence parameters. In L. Rüschendorf, B. Schweizer, and M. D. Taylor (Eds.), *Distributions with fixed marginals and related topics*, pp. 120–141. Hayward: Institute of Mathematical Statistics.

Joe, H. (1997). *Multivariate Models and Dependence Concepts*. London: Chapman & Hall.

Joe, H. (2005). Asymptotic efficiency of the two-stage estimation method for copula-based models. *Journal of Multivariate Analysis 94*, 401–419.

Kendall, M. G. (1962). *Rank Correlation Methods* (3rd ed.). London: Griffin.

Kim, G., M. J. Silvapulle, and P. Silvapulle (2007). Comparison of semiparametric and parametric methods for estimating copulas. *Computational Statistics & Data Analysis 51*(6), 2836–2850.

Klüppelberg, C., G. Kuhn, and L. Peng (2007). Estimating the tail dependence function of an elliptical distribution. *Bernoulli 13*(1), 229–251.

Kojadinovic, I. and J. Yan (2010a). A goodness-of-fit test for multivariate multiparameter copulas based on multiplier central limit theorems. *Statistics and Computing*, in press.

Kojadinovic, I. and J. Yan (2010b). Modeling multivariate distributions with continuous margins using the copula R package. *Journal of Statistical Software 34*(9), 1–20.

Kojadinovic, I., J. Yan, and M. Holmes (2010). Fast large-sample goodness-of-fit test for copulas. *Statistica Sinica*, in press.

Kullback, S. and R. A. Leibler (1951). On information and sufficiency. *The Annals of Mathematical Statistics 22*(1), 79–86.

Kurowicka, D. (2009). Vine truncations. 3rd Vine Copula Workshop, Oslo.

Kurowicka, D. and R. M. Cooke (2006). *Uncertainty Analysis with High Dimensional Dependence Modelling.* Chichester: John Wiley.

Lange, K. (1999). *Numerical Analysis for Statisticians.* Berlin: Springer.

Lanzendörfer, J. N. (2009). Joint estimation of parameters in multivariate normal regression with correlated errors using pair-copula constructions and an application to finance. Master's thesis, Technische Universität München.

Lehmann, E. L. and J. P. Romano (2004). *Testing Statistical Hypotheses* (3rd ed.). Berlin: Springer.

Lin, S. and B. W. Kernighan (1973). An effective heuristic algorithm for the Traveling-Salesman Problem. *Operations Research 21*(2), 498–516.

Ljung, G. M. and G. E. P. Box (1978). On a measure of a lack of fit in time series models. *Biometrika 65*(1), 297–303.

Manner, H. (2007). Estimation and model selection of copulas with an application to exchange rates. METEOR research memorandum 07/056, Maastricht University.

Martin, O., S. W. Otto, and E. W. Felten (1991). Large-step Markov chains for the Traveling Salesman Problem. *Complex Systems 5*, 299–326.

McNeil, A. J., R. Frey, and P. Embrechts (2005). *Quantitative Risk Management: Concepts Techniques and Tools.* Princeton: Princeton University Press.

Morales-Nápoles, O., R. M. Cooke, and D. Kurowicka (2010). About the number of vines and regular vines on n nodes. *Discrete Applied Mathematics*, submitted.

Nelsen, R. B. (2006). *An Introduction to Copulas* (2nd ed.). Berlin: Springer.

Nocedal, J. and S. J. Wright (2006). *Numerical Optimization* (3rd ed.). New York: Springer.

Owen, J. and R. Rabinovitch (1983). On the class of elliptical distributions and their applications to the theory of portfolio choice. *Journal of Finance 38*, 745–752.

Papadimitriou, C. H. and K. Steiglitz (1998). *Combinatorial Optimization: Algorithms and Complexity.* Mineola, New York: Dover Publications, Inc.

Prim, R. C. (1957). Shortest connection networks and some generalizations. *Bell System Technical Journal 36*, 1389–1401.

Quessy, J.-F. (2005). *Méthodologie et application des copules: tests d'adéquation, tests d'indépendance, et bornes pour la valeur-à-risque*. Ph. D. thesis, Université Laval, Québec, Canada.

Quessy, J.-F. (2009). Theoretical efficiency comparisons of independence tests based on multivariate versions of Spearman's rho. *Metrika 70*, 315–338.

Reinelt, G. (1994). *The Traveling Salesman: Computational Solutions for TSP Applications*. Berlin: Springer.

Rémillard, B. and O. Scaillet (2009). Testing for equality between two copulas. *Journal of Multivariate Analysis 100*(3), 377–386.

Ripley, B. D. (2008). *Pattern Recognition and Neural Networks*. Cambridge: Cambridge University Press.

Schepsmeier, U. (2010). Maximum likelihood estimation of C-vine pair-copula constructions based on bivariate copulas from different families. Master's thesis, Technische Universität München.

Schmid, F. and R. Schmidt (2007). Multivariate extensions of Spearman's rho and related statistics. *Statistics and Probability Letters 77*, 407–416.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics 6*(2), 461–464.

Sharpe, W. (1964). Capital asset prices: a theory of market equilibrium under conditions of risk. *Journal of Finance 19*(3), 425–442.

Sklar, A. (1959). Fonctions de répartition à n dimensions et leurs marges. *Publications de l'Institut de Statistique de L'Université de Paris 8*, 229–231.

Stephens, M. A. (1974). EDF statistics for goodness of fit and some comparisons. *Journal of the American Statistical Association 69*(347), 730–737.

Valdesogo, A. (2009). *Multivariate volatility models using copulas*. Ph. D. thesis, Center for Operations Research and Econometrics, Université Catholique de Louvain, Belgium.

Vuong, Q. H. (1989). Ratio tests for model selection and non-nested hypotheses. *Econometrica 57*(2), 307–333.

Wilk, M. B. and R. Gnanadesikan (1968). Probability plotting methods for the analysis of data. *Biometrika 55*(1), 1–17.

Yule, G. U. and M. G. Kendall (1965). *An Introduction to the Theory of Statistics* (14th ed.). London: Charles Griffin and Company.