

# Nonparametric Copula Density Estimation Using a Petrov–Galerkin Projection

Dana Uhlig and Roman Unger

**Abstract** Nonparametrical copula density estimation is a meaningful tool for analyzing the dependence structure of a random vector from given samples. Usually kernel estimators or penalized maximum likelihood estimators are considered. We propose solving the Volterra integral equation

$$\int_0^{u_1} \cdots \int_0^{u_d} c(s_1, \dots, s_d) ds_1 \cdots ds_d = C(u_1, \dots, u_d)$$

to find the copula density  $c(u_1, \dots, u_d) = \frac{\partial^d C}{\partial u_1 \cdots \partial u_d}$  of the given copula  $C$ . In the statistical framework, the copula  $C$  is not available and we replace it by the empirical copula of the pseudo samples, which converges to the unobservable copula  $C$  for large samples. Hence, we can treat the copula density estimation from given samples as an inverse problem and consider the instability of the inverse operator, which has an important impact if the input data of the operator equation are noisy. The well-known curse of high dimensions usually results in huge nonsparse linear equations after discretizing the operator equation. We present a Petrov–Galerkin projection for the numerical computation of the linear integral equation. A special choice of test and ansatz functions leads to a very special structure of the linear equations, such that we are able to estimate the copula density also in higher dimensions.

## 1 Copula Density Estimation as an Inverse Problem

A copula is a multivariate distribution function of a  $d$ -dimensional random vector with uniformly distributed margins. Sklar's theorem ensures that any joint multivariate distribution  $F$  of a  $d$ -dimensional vector  $\mathbf{X} = (X_1, \dots, X_d)^T$  with margins  $F_j$

---

D. Uhlig (✉) · R. Unger

Department of Mathematics, Technische Universität Chemnitz, 09107 Chemnitz, Germany  
e-mail: dana.uhlig@mathematik.tu-chemnitz.de

R. Unger

e-mail: roman.unger@mathematik.tu-chemnitz.de

© The Author(s) 2015

K. Glau et al. (eds.), *Innovations in Quantitative Risk Management*,  
Springer Proceedings in Mathematics & Statistics 99,  
DOI 10.1007/978-3-319-09114-3\_25

423

( $j = 1, \dots, d$ ) can be expressed as

$$F(x_1, \dots, x_d) = C(F_1(x_1), \dots, F_d(x_d)) \quad \forall \mathbf{x} = (x_1, \dots, x_d)^T \in \mathbb{R}^d$$

where the copula is unique on  $range(F_1) \times \dots \times range(F_d)$ , that is for continuous margins  $F_1, \dots, F_d$  the copula  $C$  is unique on the whole domain. Consequently, the copula contains the complete dependence structure of the random vector  $\mathbf{X}$ . For a detailed introduction to copulas and their properties see, for example, [8, 9, Chap. 5] or [10]. In risk management, knowledge of the dependence is of paramount importance.

If the copula is sufficiently smooth, the copula density

$$c(u_1, \dots, u_d) = \frac{\partial^d C}{\partial u_1 \dots \partial u_d} \tag{1}$$

exists and then the density gives us the dependence structure in a more convenient way, because usually the graphs of the copulas look very similar and there are only small differences in the slope. For this reason the reconstruction of the copula density is a vibrant field of research in finance and many other scientific fields. Particularly in practical tasks, the dependence structure of more than two random variables is of special interest as the dimension  $d$  is large. In the nonparametric statistical estimation, usually kernel estimators are used, but they have often problems with the boundary bias. There are also spline- or wavelet-based approximation methods, but most of them are only discussed in the two-dimensional case. Likewise, in [12], the authors discuss a penalized nonparametrical maximum likelihood method in the two-dimensional case. A detailed survey of literature about nonparametrical copula density estimation can be found in [6]. However, most of the nonparametrical methods are faced with the curse of dimensionality such that the numerical computations are only for sufficiently low dimensions possible. Actually, many authors discuss only the two-dimensional case in non-parametrical copula density estimation.

In this paper we develop an alternative approach based on the theory of inverse problems. The copula density (1) exists only for absolutely continuous copulas. Obviously, the copula is not observable for a sample  $\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_T$  in the statistical framework, but we can approximate it with the empirical copula

$$\hat{C}(\mathbf{u}) = \frac{1}{T} \sum_{j=1}^T \mathbf{1}_{\{\hat{U}_j \leq \mathbf{u}\}} = \frac{1}{T} \sum_{j=1}^T \prod_{k=1}^d \mathbf{1}_{\{\hat{U}_{kj} \leq u_k\}} \tag{2}$$

of the margin transformed pseudo samples  $\hat{\mathbf{U}}_1, \hat{\mathbf{U}}_2, \dots, \hat{\mathbf{U}}_T$  with  $\hat{U}_{kj} = \hat{F}_k(X_{kj})$  where

$$\hat{F}_k(x) = \frac{1}{T} \sum_{j=1}^T \mathbf{1}_{\{X_{kj} \leq x\}}$$

denotes the empirical margins. It is well-known that the empirical copula uniformly converges to the copula (see [2])

$$\max_{\mathbf{u} \in [0, 1]^d} |C(\mathbf{u}) - \hat{C}(\mathbf{u})| = \mathcal{O} \left( \frac{(\log \log T)^{\frac{1}{2}}}{T^{\frac{1}{2}}} \right) \text{ a.s. for } T \rightarrow \infty \tag{3}$$

Therefore, we treat the empirical copula as a noisy representation of the unobservable copula  $C^\delta = \hat{C}$ . The estimation problem of the density is faced with differentiating the empirical copula, which is obviously not smooth. However, for each density it yields the integral equation

$$\int_0^{u_1} \cdots \int_0^{u_d} c(s_1, \dots, s_d) ds_1 \cdots ds_d = C(u_1, \dots, u_d) \quad \forall \mathbf{u} = (u_1, \dots, u_d)^T \in \Omega = [0, 1]^d \tag{4}$$

which can be seen as a weak formulation of Eq. (1). In the following, we therefore consider the linear Volterra integral operator  $A \in \mathcal{L}(L^1(\Omega), L^2(\Omega))$  and solve the linear operator equation

$$Ac = C \tag{5}$$

to find the copula density  $c$ . In the following, we assume attainability which means  $C \in \mathcal{R}(A)$ , hence we only consider copulas  $C \in L^2(\Omega)$  which have a solution  $c \in L^1(\Omega)$

The injective Volterra integral operator is well-studied in the inverse problem literature. Even in the one-dimensional case, this is an ill-posed operator resulting from the noncontinuity of the inverse  $A^{-1}$ , which is the differential operator. Hence, solving Eq. (1) leads to numerical instabilities if the right-hand side of (5) has only a small data error. Because the solution is sensitive to small data errors, regularization methods to overcome the instability are discussed in the inverse problem literature. For a detailed introduction to regularization see, for example, [4, 13].

In Sect. 2 we discuss a discretization of the integral equation (4) and in Sect. 3, we illustrate the numerical instability if we use the empirical copula instead of the exact one and discuss regularization methods for the discretized problem.

The basics to the numerical implementation of the problem and especially the details of the Kronecker multiplication are presented in the authors working paper [14] and a discussion that the Petrov–Galerkin projection is not a simple counting algorithm is done in [15]. This paper gives an summary of the proposed method for effective computation of the right-hand side for larger dimensions and discusses in more detail the analytical aspects of the inverse problem and reasons for the existence of the Kronecker structure.

## 2 Numerical Approximation

We discuss the numerical computation of the copula density  $c \in X = L^1(\Omega)$  from a given copula  $C \in Y = L^2(\Omega)$ , which is in principle a numerical differentiation and in higher dimensions, a very hard problem (see [1]). Moreover, in practical applications, the measured data  $C^\delta$  have some noise  $\delta$  with  $\|C - C^\delta\|_Y \leq \delta$  and very often the function is not smooth enough that is  $C^\delta \notin C^1(\Omega)$  even  $C \in C^1(\Omega)$ , which leads to numerical instabilities making a usual numerical differentiation impossible.

For the sake of convenience, we write

$$\int_{\mathbf{0}}^{\mathbf{u}} c(\mathbf{s}) d\mathbf{s} = C(\mathbf{u}) \quad \forall \mathbf{u} = (u_1, \dots, u_d)^T \in \Omega = [0, 1]^d$$

for Eq. (4) as a short form. We propose applying a Petrov–Galerkin projection (see [5]) for some discretization size  $h$  and consider the finite dimensional approximation

$$c_h(s) = \sum_{j=1}^N c_j \phi_j(s), \tag{6}$$

where  $\Phi = \{\phi_1, \phi_2, \dots, \phi_N\}$  is a basis of the ansatz space  $V_h$ . The vector of coefficients  $\mathbf{c} = (c_1, \dots, c_N)^T \in \mathbb{R}^N$  is chosen such that

$$\int_{\Omega} \int_{\mathbf{0}}^{\mathbf{u}} c_h(\mathbf{s}) d\mathbf{s} \psi(\mathbf{u}) d\mathbf{u} = \int_{\Omega} C(\mathbf{u}) \psi(\mathbf{u}) d\mathbf{u} \quad \forall \psi \in \tilde{V}_h. \tag{7}$$

It is sufficient to fulfill Eq. (7) for  $N$  linear independent test functions  $\psi_i \in \tilde{V}_h$ . This yields the system of linear equations

$$K \mathbf{c} = \mathbf{C} \tag{8}$$

with right-hand side

$$C_i = \int_{\Omega} C(\mathbf{u}) \psi_i(\mathbf{u}) d\mathbf{u}, \quad i = 1, \dots, N \tag{9}$$

and the  $N \times N$  matrix  $K$  with

$$K_{ij} = \int_{\Omega} \int_{\mathbf{0}}^{\mathbf{u}} \phi_j(\mathbf{s}) d\mathbf{s} \psi_i(\mathbf{u}) d\mathbf{u}.$$

If the exact copula is replaced by the empirical copula, we obtain a noisy representation  $\mathbf{C}^\delta$  with

$$C_i^\delta = \int_{\Omega} \hat{C}(\mathbf{u}) \psi_i(\mathbf{u}) d\mathbf{u}, \quad i = 1, \dots, N \tag{10}$$

of the exact right-hand side  $\mathbf{C}$ . A typical phenomenon of ill-posed inverse problems is that the numerically computed solution based on noisy data (10) will be high oscillating without choosing a proper regularization. This problem is not caused by the numerical approximation, but rather by the discontinuity of the inverse operator. In Section 3 this will be illustrated. Figure 3 shows the reconstructed density of the Student copula for exact data (9), whereas Fig. 5 shows it for different noise levels.

In principle, we can choose arbitrary ansatz functions  $\phi_j \in V_h$  and test functions  $\psi_i \in \tilde{V}_h$ . However, having the curse of high dimensions in mind, we choose very simple ansatz functions such that the matrix  $K$  gets a very special structure allowing us to solve (8) and compute the approximated copula density also for higher dimensional copulas. Obviously, the approximated density (6) is not smooth and in order to obtain a smoother approximated copula  $C_h$  with

$$C_h(\mathbf{u}) = \int_{\mathbf{0}}^{\mathbf{u}} c_h(\mathbf{s}) d\mathbf{s}$$

we choose the test functions as integrated ansatz functions, such that the approximated copula

$$C_h(\mathbf{u}) = \sum_{j=1}^N c_j \psi_j(\mathbf{u})$$

is smoother than the approximated density.

We discretize the domain  $\Omega$  by splitting each one-dimensional interval  $[0, 1]$  in  $n$  equal subintervals of length  $h = \frac{1}{n}$ . Hence, we obtain  $N = n^d$  equal-sized hypercubes and call these elements  $e_1, \dots, e_N$ . We number the elements in a specific order, illustrated in Fig. 1 such that if we look at the  $(d + 1)$ -dimensional problem, the first  $n^d$  elements of the new problem have the same number and location as the elements of the  $d$ -dimensional problem.

We set  $N = n^d$  and choose the ansatz functions

$$\phi_j(\mathbf{u}) = \begin{cases} 1 & \mathbf{u} \in e_j \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

and the test functions  $\psi_i$  as the integrated ansatz functions

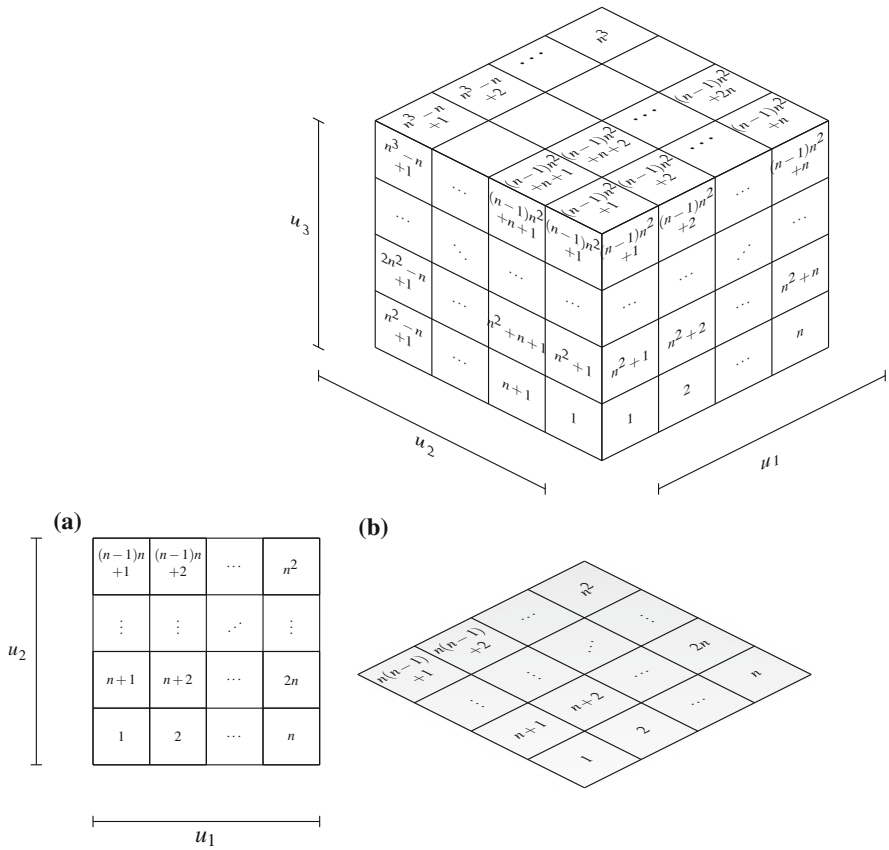
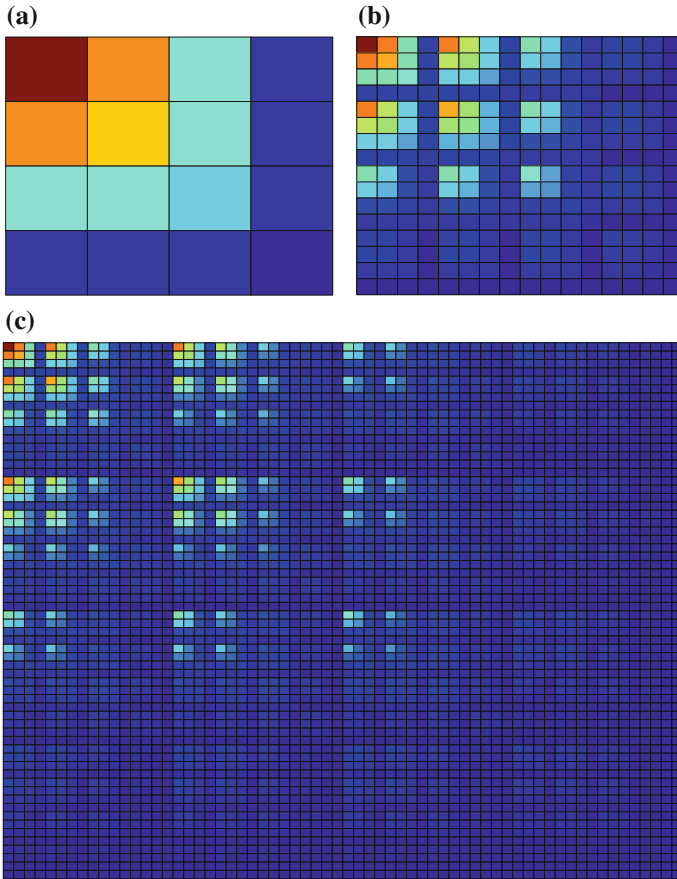


Fig. 1 Discretization of the domain  $\Omega = [0, 1]^d$ . **a**  $d = 2$ . **b**  $d = 3$

$$\psi_i(\mathbf{u}) = \int_{\mathbf{0}}^{\mathbf{u}} \phi_i(\mathbf{s}) d\mathbf{s}. \tag{12}$$

In contrast to finite element discretizations, the system matrix  $K$  is not sparse and the system size  $N = n^d$  grows exponentially with the dimension  $d$ . A straightforward assembling and solving of the linear system (8) becomes impossible for usual discretizations  $n$ . Even in the three-dimensional case, the matrix storage of the system matrix for  $n = 80$  needs approximately one terabyte, even when exploiting symmetry, and computing times for assembling and solving such systems will become enormous.

The choices (11) and (12) yield a structure of the  $N \times N$  system matrix  $K$ , illustrated in Fig. 2, allowing us to solve (8) also for  $d > 2$ . The matrixplot shows that the  $n \times n$  system matrix of the one-dimensional case is equivalent to the upper left  $n \times n$  corners of the two- and three-dimensional matrices. Moreover, the other parts of the system matrices are scaled replications of the one-dimensional  $n \times n$



**Fig. 2** Matrixplots of the system matrix  $K$  for  $n = 4$  and different dimensions  $d$ . **a** System matrix for  $d = 1$ . **b** System matrix for  $d = 2$ . **c** System matrix for  $d = 3$

system matrix. This effect is based by a Kronecker factorization of the  $d$ -dimensional system matrix into  $d$  one-dimensional matrices of the one-dimensional problem.

One important reason for this structure is that the chosen ansatz functions decomposed into a product of one-dimensional ansatz functions. In order to illustrate this, we consider the lowest corner  $\mathbf{b}^i$  of the  $i$ th element and define the one-dimensional function

$$\phi_i^k = \mathbf{1}_{\{[b_k^i, b_k^i+h]\}}$$

This yields

$$\phi_i(\mathbf{u}) = \prod_{k=1}^d \phi_i^k(u_k) \tag{13}$$

as well as

$$\psi_i(\mathbf{u}) = \prod_{k=1}^d \psi_i^k(u_k) \tag{14}$$

with the one-dimensional test functions

$$\psi_i^k(u) = \int_0^u \phi_i^k(s) ds.$$

We only formulate the main result allowing us to compute solutions of (8) also for higher dimensions  $d$ . Details and proofs can be found in the working paper [14].

**Theorem 1** *The system matrix for the  $(d + 1)$ -dimensional case can be extracted from the one and  $d$ -dimensional system matrices.*

$${}^{(d+1)}K = {}^{(1)}K \otimes {}^{(d)}K$$

**Corollary 1** *The system matrix  ${}^{(d)}K$  is the  $d$ -fold Kronecker product of the  $n \times n$  matrix  ${}^{(1)}K$*

$${}^{(d)}K = {}^{(1)}K \otimes {}^{(1)}K \otimes \dots \otimes {}^{(1)}K \tag{15}$$

*and the inverse system matrix of the  $d$ -dimensional problem is the  $d$ -times Kronecker product of the one-dimensional inverse system matrix*

$${}^{(d)}K^{-1} = {}^{(1)}K^{-1} \otimes {}^{(1)}K^{-1} \otimes \dots \otimes {}^{(1)}K^{-1} .$$

Following Corollary 1 we only have to assemble the one-dimensional system matrix  ${}^{(1)}K$  of dimension  $n \times n$ , compute its inverse  ${}^{(1)}K^{-1}$  and have to perform the Kronecker factorization for computing the solution  $\mathbf{c} = {}^{(d)}K^{-1}\mathbf{C}$  of (8). Details of the algorithm and an effective Kronecker multiplication are written in [14]. Using effective parallelization methods, the running time can be accelerated. Actually, the computation of the right-hand side (9) is the crucial part and much more expensive than solving the linear system, because we have to evaluate  $N = n^d$  different  $d$ -dimensional integrals over the whole domain  $\Omega$ . Note that for our special choice of ansatz functions (6) we have

$$C_i = \int_{\Omega} \mathbf{C}(\mathbf{u}) \psi_i(\mathbf{u}) d\mathbf{u} = \sum_{l=i}^N \mathbf{1}_{\{b^l \geq b^i\}} \int_{e_l} \mathbf{C}(\mathbf{u}) \psi_i(\mathbf{u}) d\mathbf{u} , \tag{16}$$



which also reduces the numerical effort. In higher dimensions, the number of elements  $e_i$  with zero values grows, such that using Eq. (16) instead of (9) improves the running times.

In the most practical relevant case, where the components of the right-hand side (10) are evaluated over the empirical copula (2), the numerical effort can be radically reduced, because the  $d$ -dimensional integral

$$C_i^\delta = \int_{\Omega} \hat{C}(\mathbf{u}) \psi_i(\mathbf{u}) d\mathbf{u} = \frac{1}{T} \sum_{j=1}^T \int_{\Omega} \prod_{k=1}^d \mathbf{1}_{\{\hat{U}_{kj} \leq u_k\}} \psi_i^k(u_k) d\mathbf{u} = \frac{1}{T} \sum_{j=1}^T \prod_{k=1}^d I_{ij}^k \quad (17)$$

degenerates in a product of  $d$  one-dimensional integrals

$$I_{ij}^k = \int_0^1 \mathbf{1}_{\{\hat{U}_{kj} \leq s\}} \psi_i^k(s) ds = \begin{cases} h(1 - b_k^i) - \frac{1}{2}h^2, & \hat{U}_{kj} < b_k^i \\ h(1 - b_k^i) - \frac{1}{2}h^2 - \frac{1}{2}(\hat{U}_{kj} - b_k^i)^2, & b_k^i \leq \hat{U}_{kj} \leq b_k^i + h \\ h(1 - \hat{U}_{kj}), & \hat{U}_{kj} > b_k^i + h \end{cases}$$

using Eqs. (13) and (14). In this case, the numerical effort is of order  $\mathcal{O}(NTd)$  which is an extreme improvement to  $\mathcal{O}\left(N3^d T + \frac{N^2+N}{2}3^d\right)$ , if the  $d$ -dimensional integrals (10) are numerically computed by a usual  $3^d$ -points Gauss formula. We want to point out that the computation of the right-hand side (10) for the empirical copula based on formula (17) is still possible for  $d = 9$ , whereas the computational effort for computing (16) for an arbitrary given copula  $C$  is exorbitant, even if the discretization size  $n$  is moderately chosen. The numerical effort is illustrated in Table 1.

Note that contrary to what might be expected, the vector  $\mathbf{c} = (c_1, \dots, c_N)^T$  does not count the number of samples in the elements, even though the approximated solution  $c_h$  is a piecewise constant function on the elements and the Petrov–Galerkin projection is not simple counting (for more details see [15]).

**Table 1** Computing times using (16) for the independence copula

$d$	$n$	$N$	$s_{rhs}$	$t_{rhs}$ (s)	$t_{rhs}$ using (16)	$t_{solve}$ (s)	$\ \mathbf{c} - \mathbf{c}_h\ _{L^1(\Omega)}$
2	30	900	1	0.2	<1 s	0.0005	2.5e – 10
2	60	3,600	1	2.2	<1 s	0.003	4.9e – 9
2	100	10,000	3	6.7	3 s	0.01	2.8e – 8
3	30	27,000	10	60.7	18 s	0.01	4.8e – 7
3	60	216,000	30	1,440	379 s	0.13	3.4e – 5
3	100	1,000,000	30	32,163	8,031 s	1.04	7.1e – 4
4	30	810,000	30	72,989	10,876 s	0.29	1.2e – 3
5	30	24,300,000	30		≈112 days		
6	30	729,000,000	30		≈270 years		

### 2.1 Examples

In order to illustrate the computing times and approximation quality, we use the independent copula

$$C(\mathbf{u}) = \prod_{k=1}^d u_k$$

which has the exact solution  $c(u) = 1$ . Please note that for this example, we used the exact copula as right-hand side without generating samples. So, there is no data noise and hence  $\delta = 0$ , which allows us to separate the approximation error and the ill-posedness resulting from the uncontinuity of the inverse operator  $C^{-1}$ .

Many authors (see, for example, [11]) look at the integrated square error, which is the squared  $L^2$ -norm of the difference between the copula density and its approximation. For the independent copula, the integrated square error can easily be computed

$$ISE(c, c_h) = \|c - c_h\|_{L^2(\Omega)}^2 = \frac{1}{N} \left\| \mathbf{c} - (1, 1, \dots, 1)^T \right\|_{l^2}^2 .$$

Actually, this error measure is unsuitable, because the natural space for densities is  $L^1$  instead of  $L^2$  (see [3]) and so we measure the difference in the  $L^1$ -norm, which also can be easily computed for the independence copula

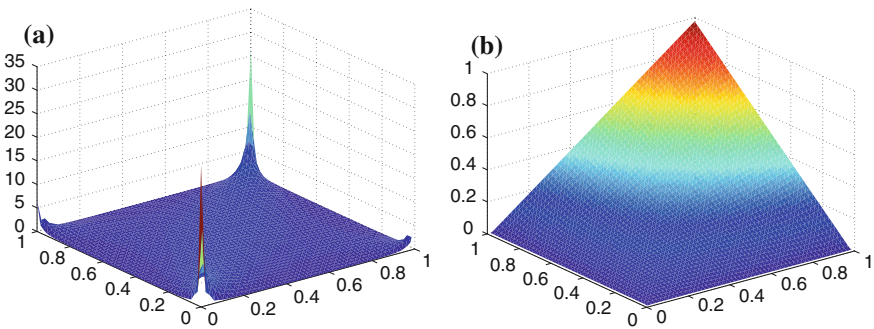
$$\|c - c_h\|_{L^1(\Omega)} = \int_{\Omega} |c(\mathbf{u}) - c_h(\mathbf{u})| \, d\mathbf{u} = \frac{1}{N} \left\| \mathbf{c} - (1, 1, \dots, 1)^T \right\|_{l^1} .$$

In Table 1, we give the following quantities for different discretization steps  $n$  in dimension 1 and dimension  $d$ : the system size  $N = n^d$ , the computing times  $t_{rhs}$  for assembling the right-hand side,  $t_{solve}$  for solving the system,  $s_{rhs}$  as the number of computing slaves and the  $L^1$ -approximation errors. For the computation of the right-hand side, a parallel OpenMPI implementation was used with  $s_{rhs}$  computing slaves. For solving the system with the Kronecker factorization, a sequential C++ implementation is used. The exact computation of an ordinary right-hand side without using the product structure gets still impossible for  $d \geq 5$  and the times are estimated computing times. In summary, the example of the independence copula shows that for exact data of the right-hand side, the approximation error is suitable but grows with decreasing discretization size  $h = \frac{1}{n}$ . We want to point out that this is typical phenomenon of inverse problems, called “regularization by discretization”.

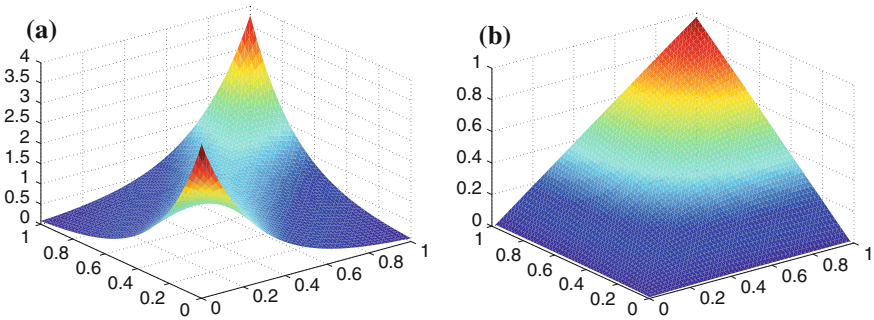
If we consider the more practical relevant case, that the empirical copula, generated by  $T$  independent samples of the independence copula, is used, we are faced with data noise  $\delta > 0$  and ill-posedness. Table 2 shows that the computation based on (17) is still possible for  $d \approx 10$ . However, the approximation error increase with the dimension  $d$ , which is a direct consequence of the ill-posedness, because the

**Table 2** Computing times using (17) for  $T = 100,000$  samples

$d$	$n$	$N$	$s_{rhs}$	$t_{rhs}$	$t_{solve}$ (s)	$\ c - c_h\ _{L^1(\Omega)}$
2	30	900	1	1.3 s	0.0002	$8.89e - 2$
2	60	3,600	1	5.3 s	0.002	$1.76e - 1$
2	100	10,000	3	5 s	0.014	$2.96e - 1$
3	30	27,000	10	7.2 s	0.013	$5.17e - 1$
3	60	216,000	30	19 s	0.11	$1.45e + 0$
3	100	1,000,000	30	86 s	0.95	$2.71e + 0$
4	30	810,000	30	97 s	0.25	$2.72e + 0$
5	30	24,300,000	30	3,607 s	8.49	$1.82e + 2$
6	10	1,000,000	30	197 s	0.14	$3.50e + 0$
7	10	10,000,000	30	2,371 s	1.68	$3.54e + 1$
8	10	100,000,000	30	26,329 s	18.2	$7.28e + 3$
9	10	1,000,000,000	30	303,239 s	253	$9.63e + 5$
10	10	10,000,000,000	30	$\approx 40$ days	2,025	



**Fig. 3** Student copula,  $\rho = 0.5$ ,  $\nu = 1$ ,  $n = 50$ , **a** reconstructed density  $c$ , **b** copula  $C$



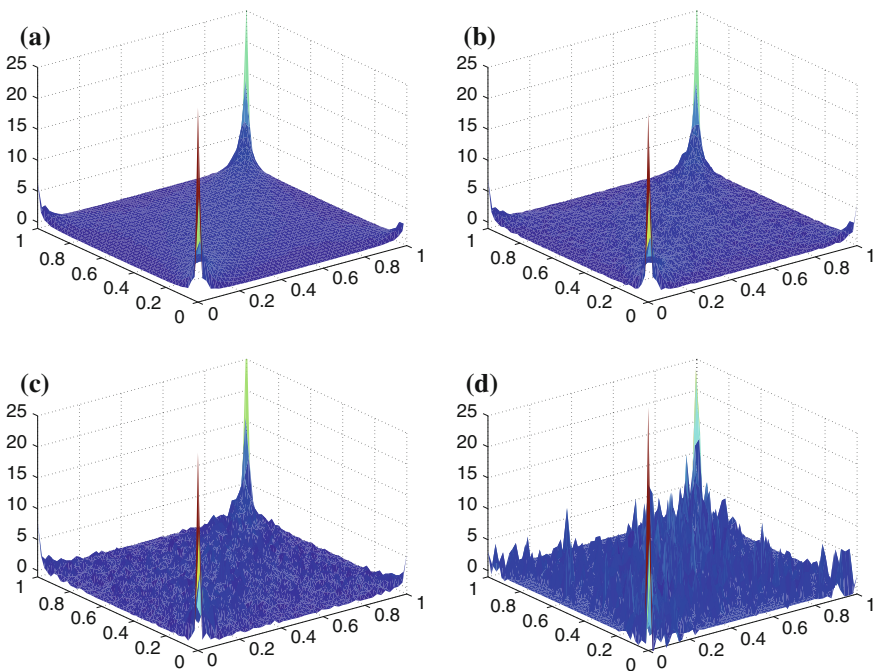
**Fig. 4** Frank copula,  $\theta = 4$ ,  $n = 50$ , **a** reconstructed density  $c$ , **b** copula  $C$

condition number of the system matrix  $K$  is the condition number of the one-dimensional system matrix  ${}^{(1)}K$  to the power of  $d$ .

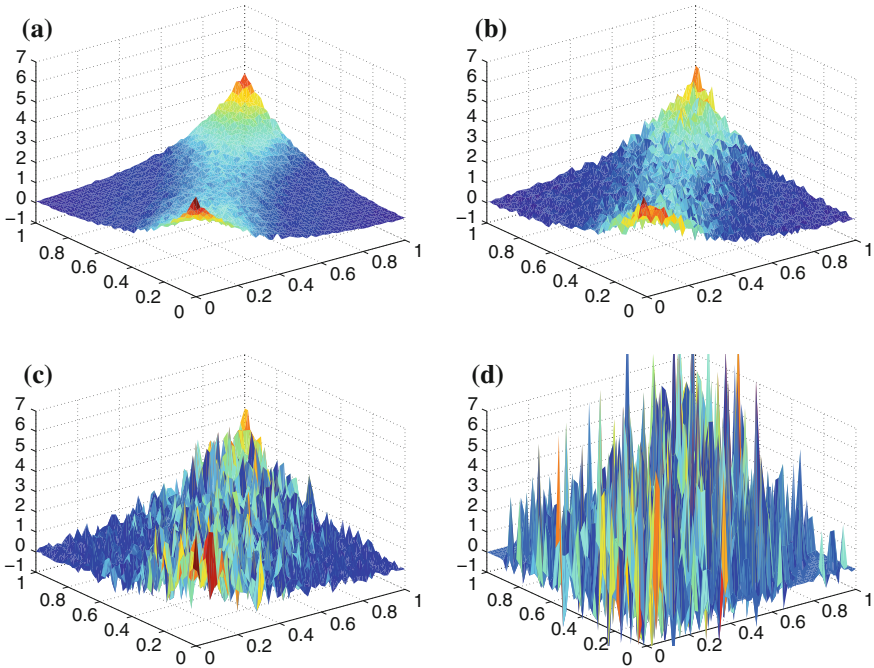
Naturally, our proposed method works not only for the rather simple independence copula, it also works quite well for all typical copula families. The approximation error for noise free right-hand sides can be neglected. Figures 3 and 4 show the reconstructed densities for the Student and Frank copula, using exact data for the right-hand side. However, ill-posedness is expected when empirical copulas are used. In [14], numerical results for other copula families, like the Gaussian, Gumbel, or Clayton copula, can also be found. However, ill-posedness is expected when empirical copulas are used and we are faced with data noise, which we discuss in the next section.

### 3 Ill-Posedness and Regularization

Note that in real problems, the copula  $C$  is not known and we only have noisy data (10) instead of (9). In order to illustrate the expected numerical instabilities, we have simulated  $T$  samples for each two-dimensional copula and present the nonparametric reconstructed densities using the Petrov–Galerkin projection with grid size  $n = 50$ .



**Fig. 5** Student copula density,  $\rho = 0.5$ ,  $\nu = 1$ ,  $n = 50$ . **a**  $T = 1,000,000$ . **b**  $T = 100,000$ . **c**  $T = 10,000$ . **d**  $T = 1,000$



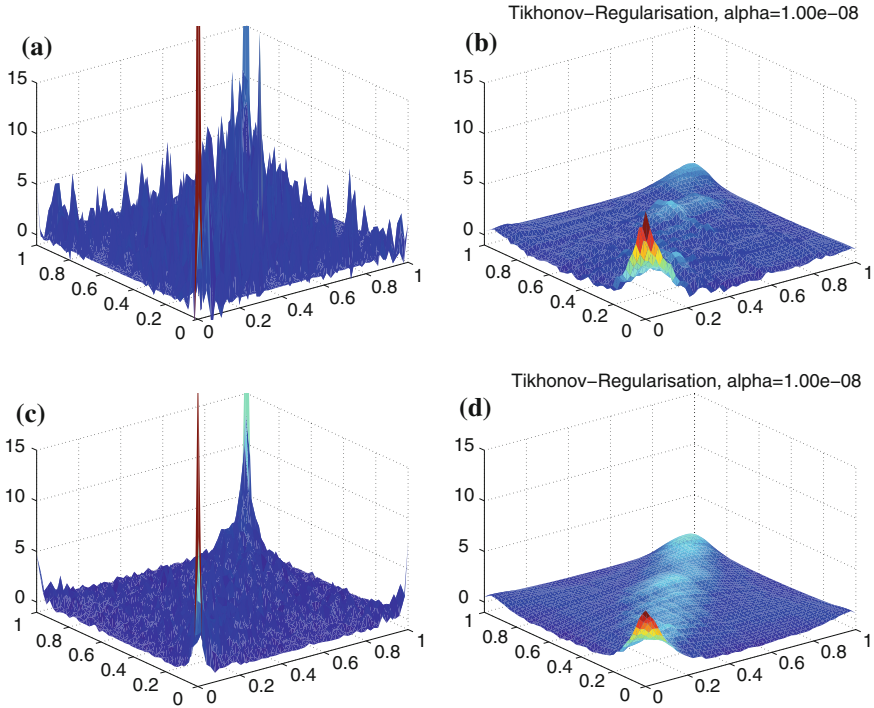
**Fig. 6** Frank copula density,  $\theta = 4$ ,  $n = 50$ . **a**  $T = 1,000,000$ . **b**  $T = 100,000$ . **c**  $T = 10,000$ . **d**  $T = 1,000$

A typical problem of ill-posed inverse problems is, that the numerical instability decreases if the grid size  $n$  decreases, which can also be seen in Table 1. Therefore, we fix the grid size  $n = 50$  and look at the influence of sample size  $T$ .

Because of (3), the data noise  $\delta$  increases if  $T$  decreases. Figures 5 and 6 show the expected ill-posedness appearing for decreasing sample size  $T$ . Of course, this instabilities also occur for the other copula families, but we restrict our illustration here to these two examples. More examples can be found in [14].

To overcome the ill-posedness, an appropriate regularization for the discretized problem (8) is required. Figures 7 and 8 show the reconstructed copula densities for  $T = 1,000$  and  $T = 10,000$  samples using the well-known Tikhonov regularization. There is no regularization, if the regularization parameter  $\alpha = 0$  is chosen. The left-hand side of the figures shows the unregularized solutions. The choice of the regularization parameter  $\alpha = 10^{-8}$  is very naive and arbitrary and serves only as demonstration how the instability can be handled. A better parameter choice should improve the reconstructed densities. It is further work to discuss an appropriate parameter choice rule for Tikhonov regularization as well as other regularization methods.

In order to avoid the complete assembling of the system matrix  $K$  leading to high-dimensional systems for  $d > 2$ , we are interested in regularization methods



**Fig. 7** Regularized Student copula density,  $\rho = 0.5, \nu = 1, n = 50$ . **a**  $\alpha = 0, T = 1,000$  samples. **b**  $\alpha = 10^{-8}, T = 1,000$  samples. **c**  $\alpha = 0, T = 10,000$  samples. **d**  $\alpha = 10^{-8}, T = 10,000$  samples

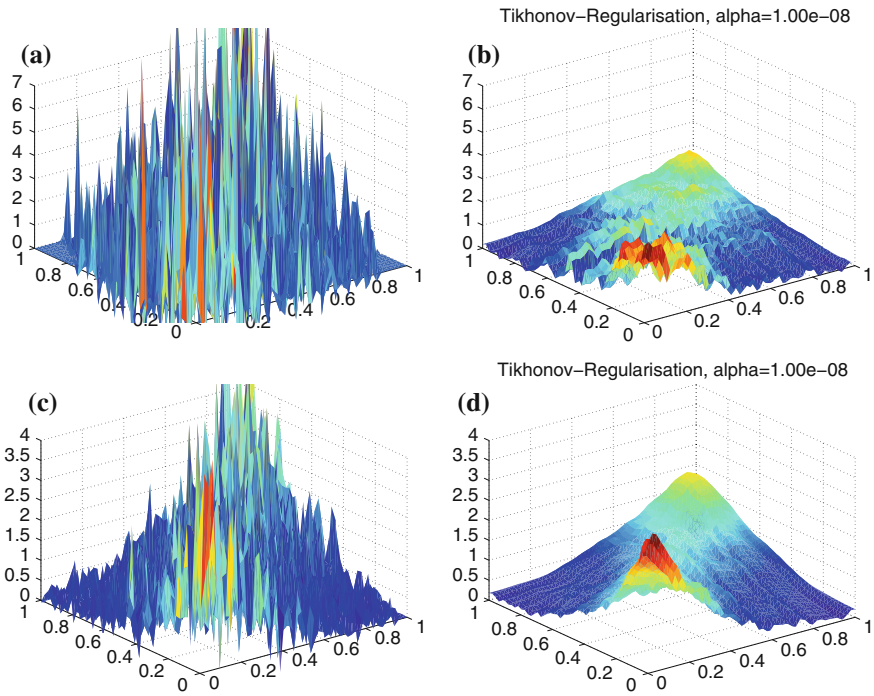
using the special structure (15). In particular, all regularization methods based on the singular value or eigenvalue decomposition of  $K$  can be easily handled because the eigenvalue decomposition of the one-dimensional matrix<sup>(1)</sup>  $K = V \Lambda V^T$  leads to the eigenvalue decomposition of the system matrix

$$K = (V \otimes \dots \otimes V) (\Lambda \otimes \dots \otimes \Lambda) (V^T \otimes \dots \otimes V^T) .$$

A typical property of Tikhonov regularization is that true peaks in the density will be smoothed. This effect appears in particular for the Student copula density. Hence, the reconstruction quality should be improved, if other regularization methods are used. In the inverse problem theory, it is well-known that Tikhonov regularization accompanies  $L^2$ -norm penalization of the regularized solutions. Therefore,  $L^1$  penalties or total variation penalties (see [7]) seem more suitable.

Furthermore, the approximated copula

$$C_h(\mathbf{u}) = \int_{\mathbf{0}}^{\mathbf{u}} c_h(\mathbf{s}) d\mathbf{s} = \sum_{j=1}^N c_j \psi_j(\mathbf{u})$$



**Fig. 8** Regularized Frank copula density,  $\theta = 4, n = 50$ . **a**  $\alpha = 0, T = 1,000$  samples. **b**  $\alpha = 10^{-8}, T = 1,000$  samples. **c**  $\alpha = 0, T = 10,000$  samples. **d**  $\alpha = 10^{-8}, T = 10,000$  samples

should yield the typical properties of copulas. For example, the requirement

$$C_h(1, \dots, 1) \stackrel{!}{=} 1$$

yields the condition  $\sum_{j=1}^N c_j = 1$  and the requirements

$$C_h(1, \dots, 1, u_k, 1, \dots, 1) \stackrel{!}{=} u_k \quad k = 1, \dots, d$$

lead to additional conditions on the vector  $\mathbf{c}$ , which all together can be used to build problem specific regularization methods.

**Open Access** This chapter is distributed under the terms of the Creative Commons Attribution Noncommercial License, which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.



## References

1. Anderssen, R.S., Hegland, M.: For numerical differentiation, dimensionality can be a blessing!. *Math. Comput.* **68**(227), 1121–1141 (1999)
2. Deheuvels, P.: Non parametric tests of independence. In: Raoult J.P. (ed.) *Statistique non Paramétrique Asymptotique*. Lecture Notes in Mathematics, vol. 821, pp. 95–107. Springer, Berlin Heidelberg (1980). doi:[10.1007/BFb0097426](https://doi.org/10.1007/BFb0097426)
3. Devroye, L., Györfi, L.: *Nonparametric Density Estimation: the L1 View*. Wiley Series in Probability and Mathematical Statistics. Wiley, New York (1985)
4. Engl, H., Hanke, M., Neubauer, A.: *Regularization of Inverse Problems*. Mathematics and Its Applications. Springer, New York (1996)
5. Grossmann, C., Roos, H., Stynes, M.: *Numerical Treatment of Partial Differential Equations*. Universitext. Springer, Berlin (2007)
6. Kauermann, G., Schellhase, C., Ruppert, D.: Flexible copula density estimation with penalized hierarchical b-splines. *Scand. J. Stat.* **40**(4), 685–705 (2013)
7. Koenker, R., Mizera, I.: Density estimation by total variation regularization. *Adv. Stat. Model. Inference* pp. 613–634 (2006)
8. Mai, J., Scherer, M.: *Simulating Copulas: Stoch. Models. Sampling Algorithms and Applications*. Series in Quantitative Finance. Imperial College Press, London (2012)
9. McNeil, A., Frey, R., Embrechts, P.: *Quantitative Risk Management: Concepts. Techniques and Tools*. Princeton Series in Finance. Princeton University Press, USA (2010)
10. Nelsen, R.B.: *An Introduction to Copulas* (Springer Series in Statistics). Springer, New York (2006)
11. Qu, L., Qian, Y., Xie, H.: Copula density estimation by total variation penalized likelihood. *Commun. Stat.—Simul. Comput.* **38**(9), 1891–1908 (2009). doi:[10.1080/03610910903168587](https://doi.org/10.1080/03610910903168587)
12. Qu, L., Yin, W.: Copula density estimation by total variation penalized likelihood with linear equality constraints. *Computat. Stat. Data Anal.* **56**(2), 384–398 (2012). doi: <http://dx.doi.org/10.1016/j.csda.2011.07.016>
13. Schuster, T., Kaltenbacher, B., Hofmann, B., Kazimierski, K.: *Regularization Methods in Banach Spaces*. Radon Series on Computational and Applied Mathematics. Walter De Gruyter In., Berlin (2012)
14. Uhlig, D., Unger, R.: A Petrov-Galerkin projection for copula density estimation. Technical report, TU Chemnitz, Department of Mathematics (2013). <http://www.tu-chemnitz.de/mathematik/preprint/2013/PREPRINT.php?year=2013&num=07>
15. Uhlig, D., Unger, R.: The Petrov-Galerkin projection for copula density estimation isn't counting. Technical report, TU Chemnitz, Department of Mathematics (2014). <http://www.tu-chemnitz.de/mathematik/preprint/2014/PREPRINT.php?year=2014&num=03>