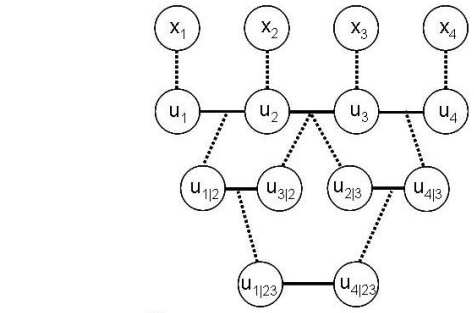
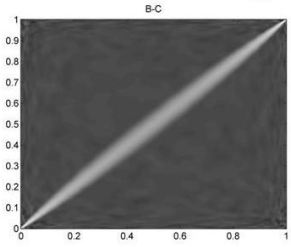
Norsk Regnesentral
NORWEGIAN COMPUTING CENTER

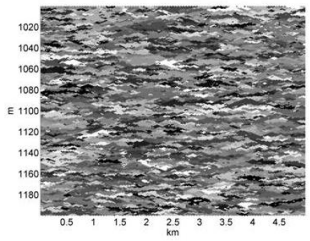
Note

D-vine Creation of Non-Gaussian Random Fields

Two-point Interactions to the Limit







Note no SAND/07/08

Authors Marita Stien
Odd Kolbjørnsen

Date 22nd September 2008

Norsk Regnesentral
NORWEGIAN COMPUTING CENTER

Norwegian Computing Center

Norsk Regnesentral (Norwegian Computing Center, NR) is a private, independent, non-profit foundation established in 1952. NR carries out contract research and development projects in the areas of information and communication technology and applied statistical modeling. The clients are a broad range of industrial, commercial and public service organizations in the national as well as the international market. Our scientific and technical capabilities are further developed in co-operation with The Research Council of Norway and key customers. The results of our projects may take the form of reports, software, prototypes, and short courses. A proof of the confidence and appreciation our clients have for us is given by the fact that most of our new contracts are signed with previous customers.

Norsk Regnesentral
Norwegian Computing Center
Postboks 114, Blindern
NO-0314 Oslo, Norway

Besøksadresse
Office address
Gaustadalléen 23
NO-0373 Oslo, Norway

Telefon · telephone
(+47) 22 85 25 00
Telefaks · telefax
(+47) 22 69 76 60

Internett · internet
www.nr.no
E-post · e-mail
nr@nr.no

Title	**D-vine Creation of Non-Gaussian Random Fields**
Authors	**Marita Stien** <marita.stien@nr.no> **Odd Kolbjørnsen** <odd.kolbjornsen@nr.no>
Date	22nd September 2008
Publication number	SAND/07/08

Abstract

We have developed a model for continuous random fields on a regular grid. We use the context of Markov mesh models to define the spatial model, and the construction of D-vines to define a transition kernel. The model is more general than the Gaussian random fields in that it can capture non-linear relation in dependencies between pairs of variables. We describe how to use this model and how it can be estimated through a non-parametric estimator. We illustrate the method in some simple examples where we calibrate our model to a training image, and generate samples from this distribution. The estimated models captures the non linear features in these examples.

Keywords	Property modeling, vines, copula, nonlinear dependency
Target group	Multipoint project group
Availability	Internal
Project	Multipoint
Project number	808002
Research field	Multipoint dependency
Number of pages	24

Contents

1 Introduction

Traditionally, geostatistical methods were developed for the purpose of prediction. Kriging was the optimal unbiased linear predictor, see Matheron (1963). Later it was acknowledged that the optimality was linked to an implicit assumption of jointly Gaussian distributions. This lead to approaches such as Disjunctive Kriging, see Matheron (1976), and indicator Kriging, see Journel (1983). These two approaches are more general than the the bias correction found in Trans-Gaussian Kriging, see e.g. Cressie (1993). In Trans-Gaussian Kriging the transform to Gaussian distribution is accounted for, the model is however still assumed Gaussian after transforming the marginal distributions. This means that spatial correlations are linear in the transformed Gaussian domain. In Disjunctive Kriging and indicator Kriging it is possible to include other types of spatial dependencies such as the mosaic model. In these models the bivariate distribution between pairs of random variables are required.

Target functions that have a nonlinear relation to the spatial parameters, e.g. flow response, and the increasing computer power have the last decades driven the use of stochastic simulations to find the optimal predictors in geostatistics. Stochastic simulation requires the full joint distribution for all random variables. Simulations of continuous random fields is frequently based on the Gaussian distribution. Point by point transformations, such as the normal score transform, see Cressie (1993), and the cloud transform, see Bashore et al. (1994), account for the non-gaussian marginal distributions. The spatial dependency is however linear in the Gaussian domain. The approach of direct sequential simulation, see Journel (1994), tries to resolve this issue by introducing non-Gaussian distributions in the sampling, but this approach frequently result in Gaussian looking distributions since the linear structure is imposed trough kriging relations.

In our approach we use a pair-copula construction to create non-linear spatial dependencies in a Markov mesh model. In particular we propose to use the pair-copula construction that is denoted D-vines. The resulting model is estimated using nonparametric estimation techniques. A pair-copula construction uses bivariate functions to describe a class of multi dimensional distributions. The pioneer work in this area is Joe (1996), Joe (1997) Bedford and Cooke (2002), and Aas et al. (2008). In this report we present how this density is constructed using pair-copulas and the marginals, giving a very compact representation of a general class of high dimensional distributions. We have used a vine decomposition called the D-vine Kurowicka and Cooke (2004), which is a symmetric decomposition that on a one dimensional grid can be formulated as a Markov random field. Sampling a D-vine requires non-linear transformations of the variables, both unconditional and conditional. The transformations are defined trough the hierarchical structure of the vine. Estimation is based on non-parametric kernel methods, both in one and two dimensions. The pair-copulas estimates the dependency between transformed variables from a given training image, and takes advantage of the hierarchial structure.

The work presented in this report has also resulted in a paper presented at Geostat 2008, Kolbjørnsen and Stien (2008). The report is organized as follows; Section 2 introduces continuous Markov random fields. In Section 3 the steps involved in a D-vine model is explained, and a detailed illustration of the simulation is given. Section 4 describes kernel estimation methods, followed by the application of these methods for D-vines in Section 5. Then two examples are displayed and discussed in Section 6, and some concluding remarks are given in Section 7.

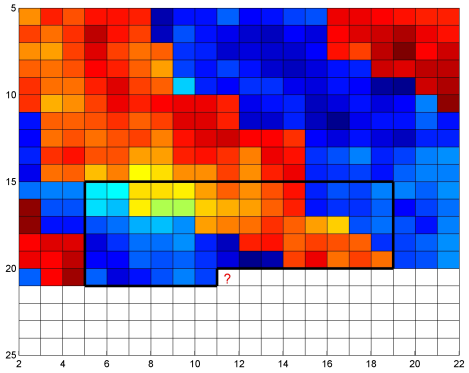


Figure 1. Unilateral path and sequential neighborhood. The grey nodes denotes all previously simulated cells in the unilateral path going top-down left-right. The sequential neighborhood is indicated by a thick black line. The conditional distribution of the random variable in the position of the question mark given cells in the past, is only dependent on the cells inside the sequential neighborhood.

2 Continuous Markov mesh model

A Markov mesh model on a regular grid is defined by first selecting a unilateral path that scans systematically through the cells, and then to model the distribution of a variable conditioned to all previously simulated variables. This gives a sequential decomposition of the distribution. Let the nodes be numerated according to the position in the systematic scan, and denote the random variable in location i by X_i. The sequential decomposition of the joint distribution is then written as,

$$f(x_1, ..., x_n) = f(x_1) \cdot f(x_2|x_1) \cdots f(x_i|x_1, \ldots, x_{i-1}) \cdots f(x_n|x_1, \ldots, x_{n-1}), \tag{1}$$

with f being a generic notation for density where the relevant variable is indicated by the argument passed to the density. Figure 1 illustrate the situation in 2D, where cell i is marked with a question mark. The distribution of X_i is defined conditioned to all cells that are previously simulated, i.e. the cells that are colored. To further constrain the dependencies, the distribution for the current cell, i.e. X_i, is assumed to only depend on those cells that are contained in a sequential neighborhood around cell i. The sequential neighborhood is denoted Γ_i, and indicated by a thick black line in the Figure 1. The decomposition of the joint distribution in the Markov mesh setting is then defined by the relation,

$$f(x_1, ..., x_n) = f(x_1) \cdot f(x_2|x_1) \cdots f(x_i|x_{\Gamma_i}) \cdots f(x_n|x_{\Gamma_n}), \tag{2}$$

where x_{Γ_i} is the set of random variables in the sequential neighborhood. The Markov mesh model is thus defined through the transition probability $f(x_i|x_{\Gamma_i})$. In our approach this is built up using pair copulas.

3 Vines and pair-copulas

A vine is a graphical model with a density made up by a number of pair-copulas and the marginal densities, and in this section we show how such a density function is constructed. Then we, more graphically, show how variables are hierachically transformed and combined in order to get a D-

vine decomposition. Simulation from a vine is illustrated using an example from the multivariate normal distribution, before a more general algorithm is presented.

3.1 Notation and pair-copula decomposition

The vine model represents the distribution of the sequential neighborhood $\{X_1, ..., X_{n-1}\}$ joint with the node we want to sample, X_n. Each of these continuous variables are associated with a marginal density function $f_i(x_i)$ and the marginal cumulative function $F_i(x_i)$. Their joint density function is $f(x_1, ..., x_n)$ with the corresponding cumulative function $F(x_1, ..., x_n)$.

Any joint density function can be factorized as which is unique up to the permutation of the variables. This joint distribution implicitly contains information about the marginal distribution of each variable and their dependency structure. Sklar's theorem decomposes a general distribution into the marginal distribution of the variables and a multivariate copula,

$$F(x_1, ..., x_n) = C\{F_1(x_1), ..., F_n(x_n)\}. \tag{3}$$

A copula is a multivariate distribution defined on the unit hyper cube with uniform variables. It gives a full measure of dependence and couples random variables such that their marginals are preserved. Using the chain rule, the density function in (3) becomes

$$f(x_1, ..., x_n) = c_{1...n}\{F_1(x_1), ..., F_n(x_n)\} \cdot f_1(x_1) \dots f_n(x_n)$$

for some n-variate copula density.

Next, we will express the factorization in (1) in terms of unconditional and conditional bivariate copula densities. We start with the second factor $f(x_2|x_1)$ which can be written

$$f(x_2|x_1) = \frac{f(x_1, x_2)}{f_1(x_1)}.$$

Replacing the numerator with

$$f(x_1, x_2) = c_{12}\{F_1(x_1), F_2(x_2)\} \cdot f_1(x_1) \cdot f_2(x_2),$$

yields the conditional probability

$$f(x_2|x_1) = c_{12}\{F_1(x_1), F_2(x_2)\} \cdot f_2(x_2). \tag{4}$$

The next factor in (1) is $f(x_3|x_2, x_1)$ and involves the conditioning nodes x_1 and x_2. We write

$$f(x_3|x_2, x_1) = \frac{f(x_3, x_2|x_1)}{f(x_2|x_1)} = c_{23|1}\{F(x_3|x_1), F(x_2|x_1)\} \cdot f(x_3|x_1), \tag{5}$$

where $f(x_3|x_1)$ is again given by a pair-copula and the marginal as in (4). Note that it could just as well have been $x_3|x_2$ and $x_1|x_2$ that was coupled, however, that is a question of the permutation of the variables. Equation (5) enforces a structure on the probability distribution. In general the first term in expression (5) should be $c_{23|1}\{F(x_3|x_1), F(x_2|x_1), F(x_1)\}$, but the dependency on $F(x_1)$ is dropped to simplify the model.

For two and higher number of conditioning variables, the conditional probabilities are recursively decomposed into (conditional) pair-copulas and marginals. We introduce the two sets x_ν which is a subset of the variables and $x_{\nu_{-j}}$ which is a set same set excluding x_j. In general terms we get the recursive expression

$$f(x_i|x_\nu) = c_{ij|\nu_{-j}}\{F(x_i|x_{\nu_{-j}}), F(x_j|x_{\nu_{-j}})\} \cdot f(x_i|x_{\nu_{-j}}).$$

Where the function $c_{ij|\nu_{-j}}\{F(x_i|x_{\nu_{-j}}), F(x_j|x_{\nu_{-j}})\}$ in general could depend on $x_{\nu_{-j}}$, but this dependency is dropped to simplify the model. Constructing the pair-copulas involve the computation of various cumulative conditional distributions $F(x_i|x_{\nu_{-j}})$ and $F(x_j|x_{\nu_{-j}})$. With the hierarchical construction of the vine, these can be computed from pair-copulas at lower levels. The

meaning of level will be clear when we introduce the vine graphically, but in mathematical terms it corresponds to the number of conditioning nodes. We let $x_{\nu_{-k}}$ be the set $x_{\nu_{-j}}$ where x_k is excluded. The conditional cumulative probabilities can then be computed by

$$\begin{aligned} F(x_i|x_{\nu_{-j}}) &= \int_{-\inf}^{x_i|x_{\nu_{-j}}} f(y|x_{\nu_{-j}})\,dy = \int_{-\inf}^{x_i|x_{\nu_{-k}}} c_{ik|\nu_{-k}}(F(y), F(x_k|x_{\nu_{-k}})) \cdot f(y)\,dy \\ &= \int_0^{F(x_i|x_{\nu_{-k}})} c_{ik|\nu_{-k}}(u, F(x_k|x_{\nu_{-k}}))\,du, \end{aligned}$$

and

$$F(x_j|x_{\nu_{-j}}) = \int_0^{F(x_j|x_{\nu_{-k}})} c_{jk|\nu_{-k}}(u, F(x_k|x_{\nu_{-k}}))\,du,$$

and we see how the $F(\cdot|\cdot)$ are computed by summing over the corresponding copula density. For notational ease we denote the various $F(\cdot|\cdot)$ in terms of $u_{\cdot|\cdot}$ for instance

$$F(x_i|x_{\nu_{-j}}) = u_{i|\nu_{-j}}.$$

The inverse $x_i = F^{-1}(y|x_{\nu_{-j}})$, is found by solving for x_i in

$$y = \int_0^{F(x_i|x_{\nu_{-k}})} c_{ik|\nu_{-k}}(u, F(x_k|x_{\nu_{-k}}))\,du.$$

We clearly see that the joint density function $f(x_1, ..., x_n)$ can been expressed as a product of pair-copulas, acting on several conditional probability distributions. The construction is iterative, and can be built using several different groupings of the variables. Next, we show one such possible permutation and hierarchical coupling called a D-vine.

3.2 D-vine decomposition

A vine is a graphical model that together with the pair-copulas specifies the dependence structure of a high-dimensional distributions. Unlike the Markov field model and the bayesian belief nets that are based on conditional independence, a vine model easily incorporates conditional dependence. The constraints of the vine models, therefore, does not lie in the independence but rather in how the variables are hierarchically paired.

Bedford and Cooke(2001b,2002) introduces a class of regular vines, which embraces a large number of possible pair-copula decompositions. The D-vine is one such decomposition (Kurowicka and Cooke,2004). Figure 2 displays the D-vine decomposition for a four dimensional distribution $f(x_1, x_2, x_3, x_4)$.

The interpretation of the graphical model is as follows; The horizontal solid lines represent the various pair-copula decompositions, i.e. they link the two transformed variables that form the corresponding copula. For instance the edge between $u_{1|2}$ and $u_{3|2}$ represents the copula $c_{13|2}(u_{1|2}, u_{3|2})$. The dotted vertical lines represent the various transformations both unconditional and conditional. By transformation we mean the computation of the u_i's and $u_{i|\nu_{-j}}$'s. Note that the graphical model is generalized for higher dimensions than four, and that the vine is generic in the sense that the order of the variables in the first row determines the vine completely.

The graphical model is an aid for constructing the joint expression $f(x_1, ..., x_n)$. The building blocks consist of the pair-copulas that correspond to the solid edges and the marginal distributions. Recall the factorization expression in (1), it can be shown that with a $\{X_1, ..., X_n\}$ permutation we get the general D-vines expression

$$\prod_{k=1}^{n} f(x_k) \prod_{j=1}^{n-1} \prod_{i=1}^{n-j} c_{i,i+j|i+1,...,i+j-1}\{F(x_i|x_{i+1}, ..., x_{i+j-1}), F(x_{i+j}|x_{i+1}, ..., x_{i+j-1})\}.$$

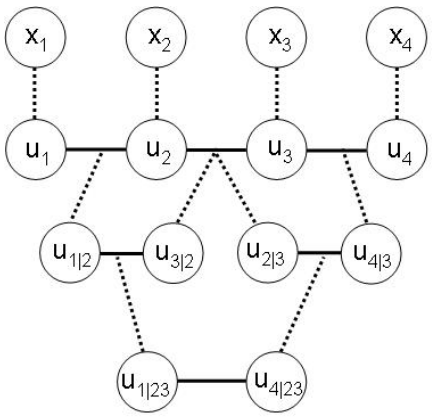


Figure 2. Illustration of a four dimensional D-vine decomposition. The solid line represents the copula between the various transformed variables and the dotted lines represent the variable transformations.

In our four dimensional example we get the following joint distributions,

$$
\begin{aligned}
f(x_1,x_2,x_3,x_4) = f_1(x_1)\cdot\\
&c_{12}\{F_1(x_1),F_2(x_2)\}\cdot f_2(x_2)\cdot\\
&c_{13|2}\{F(x_1|x_2),F(x_3|x_2)\}\cdot c_{23}\{F_2(x_2),F_3(x_3)\}\cdot f_3(x_3)\cdot\\
&c_{14|23}\{F(x_1|x_2,x_3),F(x_4|x_2,x_3)\}\cdot c_{24|3}\{F(x_2|x_3),F(x_4|x_3)\ \cdot c_{34}\{F_3(x_3),F_4(x_4)\}\cdot f_4(x_4).
\end{aligned}
$$

3.3 D-vine simulation

A joint distribution has now been established according to the D-vine decomposition. This section describes the hierarchical sampling routine for the D-vine, first with a three dimensional example followed by a generalization to higher dimensions. However, we start with an example based on a multi-normal distribution where the transformations are linear and easy to interpret and understand.

We let the variables X_1, X_2, and X_3 be gaussian distributed with mean μ and covariance structure $\mathbf{\Sigma}$, i.e.

$$
\begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix} \sim N\left(\begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{bmatrix}, \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{12} & \sigma_{22} & \sigma_{23} \\ \sigma_{13} & \sigma_{23} & \sigma_{33} \end{bmatrix} \right).
$$

First, the variables are transformed to the standard normal domain

$$
\begin{aligned}
n_1 &= \frac{-\mu_1}{\sigma_{11}} + \frac{1}{\sigma_{11}} x_1 = T(x_1) \\
n_2 &= \frac{-\mu_2}{\sigma_{22}} + \frac{1}{\sigma_{22}} x_2 = T(x_2) \\
n_3 &= \frac{-\mu_3}{\sigma_{33}} + \frac{1}{\sigma_{33}} x_3 = T(x_3),
\end{aligned}
$$

where $T(\cdot)$ represents the linear transform. These variables are sampled sequentially starting with n_1, which is straight forward drawn from the standard normal distribution.

Next, we sample $n_2|n_1$. The conditional probability can easily be computed by gauss elimination of the covariance structure, i.e. we obtain transformed variables such that n_2 and n_3 are algebraically independent from n_1. These transformed variables are given by

$$
\begin{aligned}
n_{2|1} &= n_2|n_1 = n_2 - \frac{\sigma_{12}}{\sigma_{11}} n_1 \\
n_{3|1} &= n_3|n_1 = n_3 - \frac{\sigma_{13}}{\sigma_{11}} n_1,
\end{aligned}
$$

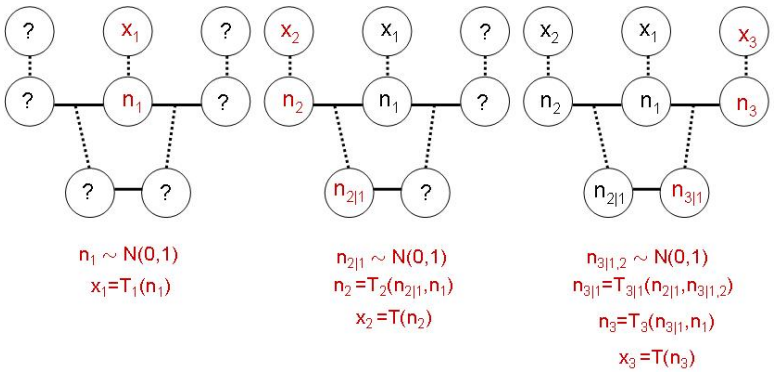


Figure 3. Visualization of the vine simulation of X_3 for a three dimensional normal distribution.

having the covariance matrix

$$\begin{bmatrix} \sigma_{22} - \frac{\sigma_{12}^2}{\sigma_{11}} & \sigma_{23} - \frac{\sigma_{13}\sigma_{12}}{\sigma_{11}} \\ \sigma_{23} - \frac{\sigma_{13}\sigma_{12}}{\sigma_{11}} & \sigma_{33} - \frac{\sigma_{13}^2}{\sigma_{11}} \end{bmatrix}.$$

The space is now reduced with one dimension, and again, we sample the first variable which now is $n_{2|1}$.

The variables n_2 and n_3 is thus expressed in terms of the linear transformations

$$\begin{aligned} n_2 &= n_{2|1} + \frac{\sigma_{12}}{\sigma_{11}} n_1 = T_2(n_1, n_{2|1}) \\ n_3 &= n_{3|1} + \frac{\sigma_{13}}{\sigma_{11}} n_1 = T_3(n_1, n_{2|1}), \end{aligned}$$

where $n_{3|1}$ is computed by yet another linear transform

$$n_{3|1} = T_{3|1}(n_{2|1}, n_{3|2,1}).$$

Now, $n_{3|2,1}$ is sampled and a full sample of x_1, x_2 and x_3 is obtained by the linear transformations.

Figure 3 visualizes the above simulation of the trivariate gaussian distribution. The transformed values of the covariance matrix are analogous to the more common term; partial correlations. For non-gaussian variables these are called conditional dependencies and are non-linear transformations. Vines is a generic way of creating non-linear transforms.

When sampling a non-gaussian variable we work in the uniform domain, i.e. the conditional dependency structure is computed from the cumulative transform of the variables. Thus, to sample a three dimensional D-vine distribution we start by sampling a uniform (0,1) variables u_1, and x_1 is computed from the inverse cumulative marginal transform, $x_1 = F_1^{-1}(u_1)$. Next, we draw a uniform(0,1) value for $u_{2|1}$, and u_2 is computed from the relation

$$u_{2|1} = \int_0^{u_2} c_{12}(u_2|u_1).$$

Finally, we draw a value for $u_{3|1,2}$ and solve for $u_{3|1}$ in

$$u_{3|1,2} = \int_0^{u_{3|1}} c_{23|1}(u_{3|1}|u_{2|1}),$$

and for u_3 in $u_{3|1} = \int_0^{u_3} c_{1|3}(u_3|u_1)$. Now, with the marginal inverse functions we easily obtain the sample $\{x_1, x_2, x_3\}$. These steps are all visualized in Figure 4.

We note that in our application of the vine model the variables $x_1, ..., x_{n-1}$ are known and only x_n needs to be sampled. This still involves the hierarchical computations of the conditional

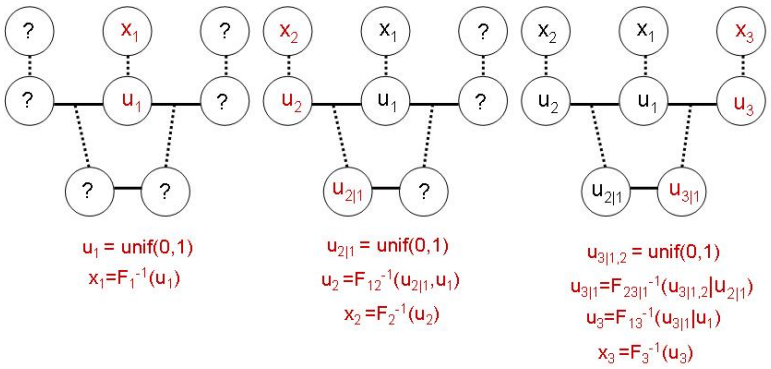


Figure 4. Visualization of the vine simulation of X_3 for a three dimensional general distribution.

variables, however, only one uniform(0,1) variable has to be drawn at the top of the hierarchy. From the illustration in Figure 4 this corresponds to the rightmost computations.

Algorithm 1 gives the procedure for sampling the variable x_n in an n dimensional distribution. First, all the variables are transformed by the cumulative densities to the uniform domain. Next, the outer loop runs over the levels from 1 to the top level $n-2$, and the inner loop computes all the transformed variables that do not involve the nth and unknown variable. Once the top level is reached, only one of the two transformed variables are known. A random number is drawn, such that the unknown variable can be computed with the inverse transform. A new loop runs from the top level to the bottom level in order to compute all the transformed variables that involve x_n. Once we get the value for u_n, the marginal inverse is applied and we get the x_n sample.

1: **for** $i \leftarrow 1, n-1$ **do**
2: Compute $u_i = F^{-1}(x_i)$
3: **end for**
4: **for** $level \leftarrow 1, n-2$ **do**
5: **for** $i \leftarrow 1, n-level-2$ **do**
6: Compute $u_{i|i+1,\ldots,i+level}$
7: Compute $u_{i+level+1|i+1,\ldots,i+level}$
8: **end for**
9: **end for**
10: Draw $u \sim unif(0,1)$
11: **for** $level \leftarrow n-2, 1$ **do**
12: $u = F^{-1}(u|u_{n-level-1|n-level,\ldots,n-1})$
13: **end for**
14: Sample $x_n = F_n^{-1}(u)$

Algorithm 1. Simulation algorithm for D-vine. Generates the n-th sample variable in an n-dimensional distribution

4 Non-parametric estimation

Estimation of the copulas can be achieved fully parametrically by assuming parametric models for both the copula and the marginals, followed by a parametric estimation procedure like the

maximum likelihood estimator. However, unlike the joint and marginal distributions which are directly observable, the copula has a hidden dependence structure. Finding a suitable parametric model is thus very often intractable. In risk management parametric copulas are useful when there is a joint tail behavior. Data for this kind of behavior is limited, and a parametric model helps prevent underestimation in these areas.

We get our data from training images with highly repetitive characteristics. Thus, the amount of data is assumed to be sufficient and a non-parametric estimation method is considered to give a good estimate of the dependency structure. This way we do not give any prior assumption on the data and the dependency structure, yielding a very general approach where training images with very different features can be used. We use a non-parametric estimation method for both that marginals and the pair-copulas, with a kernel based approach. This way we get a smooth and differentiable estimate of the distributions.

In this section we describe how the one dimensional distribution of the marginals and the two dimensional copula densities are estimated. We base our estimators on theory of kernel density estimation as presented in Silverman (1986). Both methods are based on computation of histograms. In one dimension we denote the histogram by $H_1(i)$ where i represents the bin number, and $H_2(i,j)$ in two dimensions where (i,j) represents the bin.

4.1 One-dimensional kernel smoothing

We use the one dimensional gauss kernel given by

$$K_1\left(\frac{r}{nh}\right) \propto \exp\left\{-0.5\frac{r^2}{(nh)^2}\right\},$$

where r is an integer, n is the number of histogram bins and h is a smoothing parameter. The kernel is applied to the histogram where it in each histogram bin works as a weighted sum over the neighboring bins. The smoothing parameter determines the range of the set of neighboring bins. We get a binned estimate f with the same number of bins as the input histogram. The estimate is given by

$$\hat{f}(i) = \frac{1}{nh}\sum_{r=-l}^{l} H_1(i-r)K_1\left(\frac{r}{nh}\right), \tag{6}$$

where l is an integer which truncates the gauss kernel. We set the l value approximately equal to three standard deviations.

The choice of smoothing parameter is of crucial importance in the density estimation. The mean integrated square error is a measure of discrepancy of the density estimator $\hat{f}$ from the true density f,

$$\begin{aligned} MISE &= E\left[\int\{\hat{f}(x)-f(x)\}^2\right] = \int \text{bias}(\hat{f}(x))^2\,dx + \int \text{Var}(\hat{f}(x))\,dx \\ &\approx \frac{1}{4}h^4k_1\int f''(x)^2\,dx + \frac{1}{nh}\int K_1(t)^2\,dt, \end{aligned}$$

where k_1 is a constant and is dependent of the kernel.

The two components of the mean integrated square error, bias and variance, both have contributions based on the smoothing h. The attempt to eliminate the bias requires a very small value of h, however, that results in a very large variance, and vice versa. Thus, there is a trade-off between random and systematic error and the smoothing parameter must be chosen carefully. Simple derivation give the optimal h that minimizes the expression,

$$h_{opt} \approx k_1^{-2/5}\left\{\int K(t)^2\,dt\right\}^{1/5}\left\{\int f''(x)^2\,dx\right\}^{-1/5} n^{-1/5}. \tag{7}$$

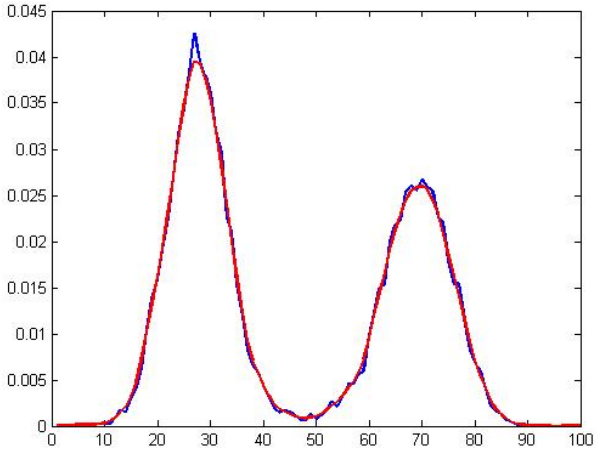


Figure 5. One dimensional kernel smoothing estimates.

We see that h slowly converge to zero as the sample size increases. Also, smaller values of h is appropriate when the data is highly fluctuating.

To assign the value of $f''(x)$ we turn to the standard normal distribution, this yields

$$h_{opt} = \frac{4}{3}^{1/5} \sigma n^{-1/5},$$

where n is the number of observations and σ is the standard deviation of the data. This parameter will work well if the data really is normally distributed, otherwise it may oversmooth. In many applications a subjective choice of smoothing parameter is satisfactory. We, however, want an automatic computation of the smoothed parameter, but since the data can have any distribution we add a tuning parameter a_1 such that $h = a_1 \cdot h_{opt}$. This way we can adjust for some of the possible discrepancy from the normal distribution. Figure 5 shows the histogram h in blue and the smoothed estimates $\hat{f}$ in red as an example.

4.2 Two-dimensional kernel smoothing

In two dimensions we get the expression

$$\hat{f}(i,j) = \sum_{r=-l}^{l} \sum_{s=-l}^{l} H(i-r, j-s) \cdot K_2 \left(\frac{r}{h_x \cdot N_2}, \frac{s}{h_y \cdot N_2} \right), \tag{8}$$

where $\hat{f}(i,j)$ is a two-dimensional matrix of the same size as H. The optimal smoothing parameter h_{opt} is given by

$$h_{opt} \approx 2 \cdot \mathbf{k_2}^{-2/6} \left\{ \int K(\mathbf{t})^2 \, d\mathbf{t} \right\}^{1/6} \left\{ \int (\nabla f(\mathbf{x}))^2 \, d\mathbf{x} \right\}^{-1/6} n^{-1/6}. \tag{9}$$

The data in the bivariate density are uniform(0,1). Figure 6(a) show the scatter plot of two such variables. We see the changes in the mass have different scales in different areas, especially in the corners and in the middle. A straight forward approach calls for boundary corrections since the domain of definition is finite, see Gijbels and Mielniczuk (1990). We avoid the problem of boundary effects by transforming the uniform variables into a standard normal domain. Figure 6(b) shows the normal quantiles for the data in Figure 6(a).

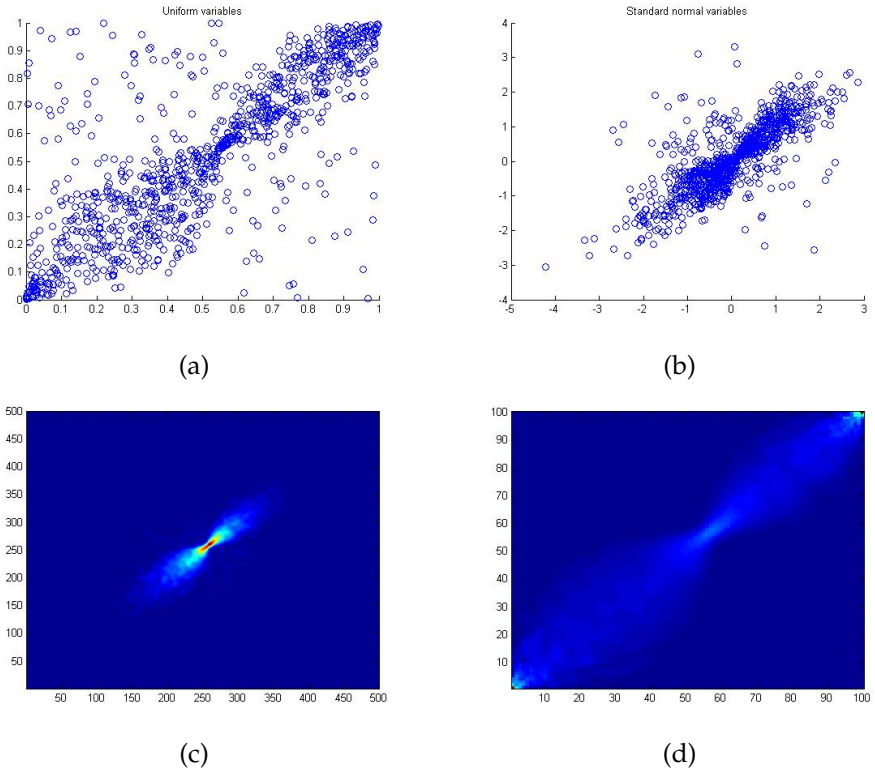


Figure 6. Two dimensional kernel smoothing estimates.

Estimating the bivariate density based on the normal transformed data is done using a 2D-gaussian kernel. This gives us a general and very simple approach, and we get the nested sum

$$\hat{f}(i,j) = \sum_{r=-l}^{l} K_1\left(\frac{r}{h \cdot N_2}\right) \cdot \sum_{s=-l}^{l} H(i-r, j-s) \cdot K_1\left(\frac{s}{h \cdot N_2}\right),$$

where h is set equal in both directions since we have the same marginal distributions. The standard deviation is 1, yielding

$$h = a_2 \cdot 0.7 \cdot n^{-1/6},$$

where a_2 is similar to a_1 in the one-dimensional case, it is an adjustment factor for multi modality.

The kernel estimated distribution $\hat{f}$ is transformed back to their respective uniform marginals. In Figures 6 (a)-(d) below we display some results for a two dimensional kernel smoothing, (a) scatter of two uniform variables (b) their standard normal quantiles (c) the kernel smoothed bivariate estimate of the standard normal data, and (d) the estimated copula, i.e. the copula of the estimated density.

5 Non-parametric estimation of the D-vine

Up to now, we have established how each marginal and pair-copula are estimated using a non-parametric approach. In this section we describe in a similar way as with the simulation procedure, how the full distribution is hierarchically estimated by sequentially transforming the variables to estimate the pair-copulas.

First of all the marginals are estimated with the one dimensional kernel method. The data is then transformed to the uniform domain. Starting at the first level, the unconditional pair-copulas

are estimated according to the given bivariate constraints, i.e. the variables that are connected by a solid line are paired. Then the uniform data are transformed according to the dotted lines to a new set of uniform data by applying (6) on the estimated pair-copulas. This process is continued until the highest level is reached and all of the involved pair-copulas are estimated. This estimation procedure is more formally written in Algorithm 2.

---

for $i \leftarrow 1, n-1$ **do**
  for $j \leftarrow 1, \#data$ **do**
    Compute $u_i^j = F^{-1}(x_i^j)$
  end for
end for
for $level \leftarrow 1, n-2$ **do**
  for $i \leftarrow 1, n-level-2$ **do**
    for $j \leftarrow 1, \#data$ **do**
      Compute $u^j_{i|i+1,\ldots,i+level}$
      Compute $u^j_{i+level+1|i+1,\ldots,i+level}$
    end for
    Estimate $c_{i,i+level+1|i+1,\ldots,i+level}(\mathbf{u}_{i|i+1,\ldots,i+level}, \mathbf{u}_{i+level+1|i+1,\ldots,i+level})$
  end for
end for

Algorithm 2. Estimation algorithm for an n-dimensional D-vine distribution.

6 Results

In this section we show two examples of usage of the methodology. The first example is generated using a D-vine random field, next the model is estimated back using both a 3D kernel estimator and the hierarchical estimation procedure from Section 5. Markov Mesh realizations are generated from both the vine model and the 3D kernel estimator, and these are compared to the true model visually based on realizations and the resulting 2D densities. The next example is a mosaic model generated on a hexagonal grid, the results are evaluated visually based on realizations and 2D scatter plots.

6.1 Example 1

As a test case we have generated a D-vine random field displayed in Figure 7. The sequential neighborhood is illustrated in Figure 8. When sampling X_3 given the past, only the values of the field directly above and to the left are taken into account.

The joint density of (X_1, X_2, X_3) is estimated non parametrically by the use of 3D kernel estimates and and the vines model described in Section 5. For comparison of the kernel estimates we display the marginal distribution for the three variables in Figure 9, and the distribution of pairs of variables on a grid ranging from -6 to 6 in each direction in Figure 10. In both figures the rows of images represent respectively the distribution of the true model, the vine model and the 3D kernel estimator. Both models yields a satisfactory result. For an application in reservoir characterization it is the resulting picture that is generated from the model that is important. Figure 11 shows the realizations for the two estimated models. Realization from the vine model to the left and from the 3D kernel model to the right. Also for this case we see that both models

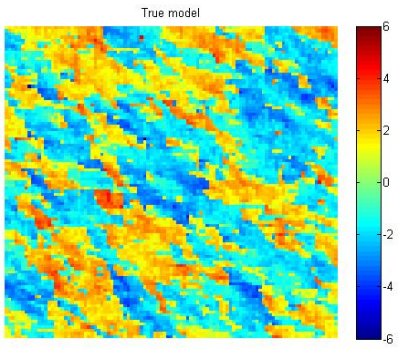


Figure 7. Training image example 1.

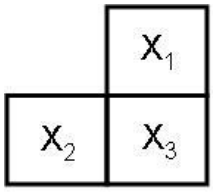


Figure 8. Neighborhood example 1. Illustration of the neighbors X_1, X_2, and the sampling node X_3.

preform equally well, but the vines model is represented in a more compact manner. The vines model use using 3 2D tables whereas the 3D smoother use one 3D table. In this case this storage for the D-vine model is just 3% of what is needed for the 3D kernel case.

6.2 Example 2

We define a mosaic random field on a hexagonal grid to test the approach. In a Mosaic random field the field value in two cells are either identical or independent. The structure of the hexagonal grid is displayed to the left in Figure 12. Note that each column in the grid is shifted $\frac{1}{2}$ cell this gives horizontal features in the grid rather than skewed as is displayed in previous example. In the grid we do a unilateral scan moving left-right, top-down. For each cell D to be simulated we draw the value according to the following rule:

$$f(x(D)|past) = p_0\delta_{x(A)} + p_0\delta_{x(B)} + (1-2p_0)N(0,1).$$

This means that with probability $(1-2p_0)$ the value of $x(D)$ is sampled from a Gaussian distribution independent of everything in sight, otherwise $x(D)$ is randomly selected to be equal to $x(A)$ or $x(B)$. In the simulation displayed to the right in Figure 12, we have used $p_0 = 0.45$. We use a training image of size 1000×1000 to estimate the model. The smoothing parameter was fit manually to get an reasonable smoothness in the copulas. In the estimation procedure we use the nodes A,B,C, and D. These are translated into the D-vine displayed in Figure 2 using the relations $x_1 = x(A)$, $x_2 = x(B)$, $x_3 = x(C)$, and $x_4 = x(D)$. The estimated copulas are displayed in Figure 13. Note that copulas in layers below the first displays non-trivial features. A simulation based on the estimated model is shown in Figure 14. The simulation is similar to the original model displayed on the right hand side of Figure 12. In Figure 15 we display scatter plots of the original mosaic model in the upper row and the estimated model in the lower row. We see that the method reproduce the mosaic nature in all 2D distributions, but the delta nature of the central

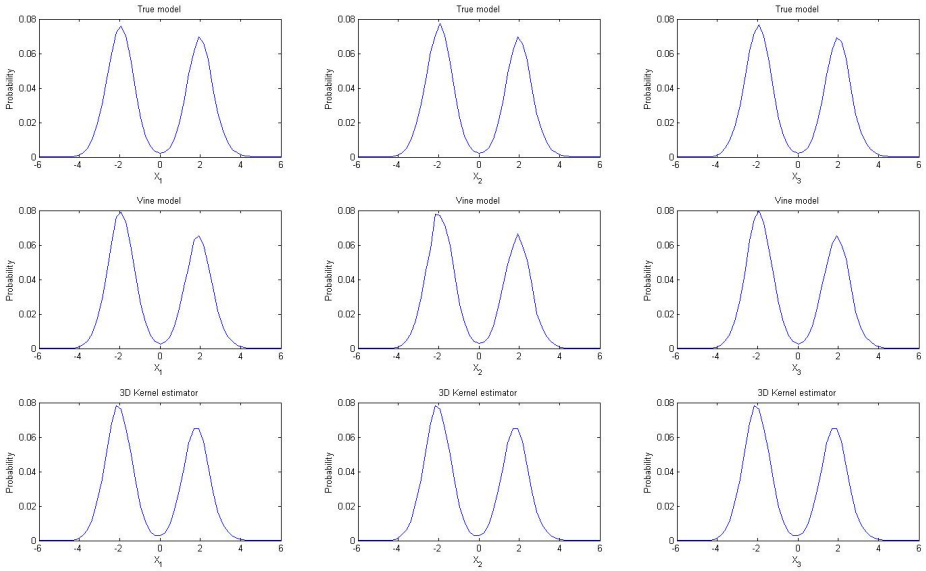


Figure 9. The marginal distribution for the variables. The rows of plots represent the true model, the vine model and the 3D kernel estimator, respectively.

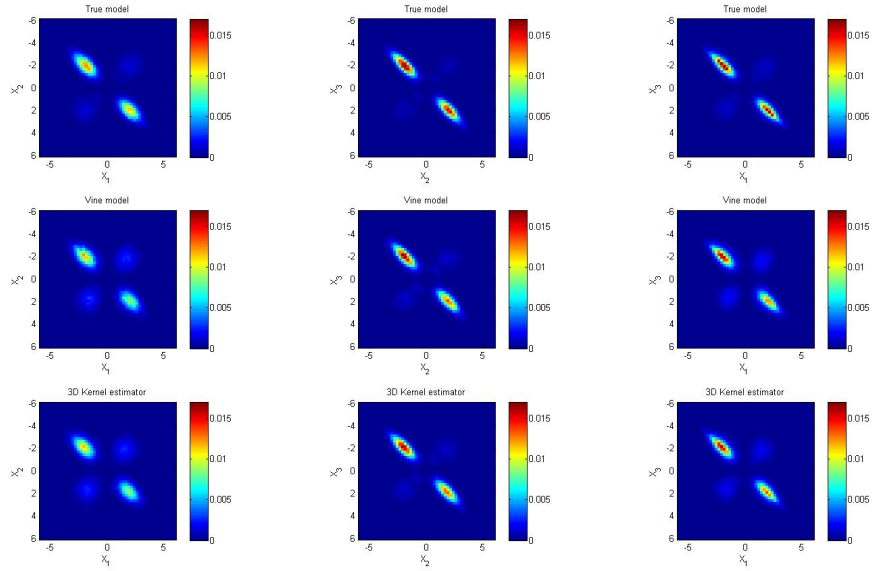
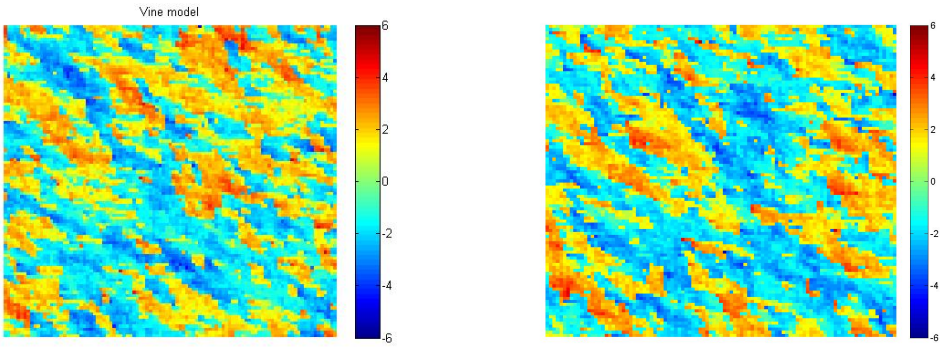


Figure 10. Pair-wise distribution of the variables. The rows of images represent the true model, the vine model and the 3D kernel estimator, respectively.



(a) (b)

Figure 11. Realization from the vine model to the left and from the 3D kernel model to the right.

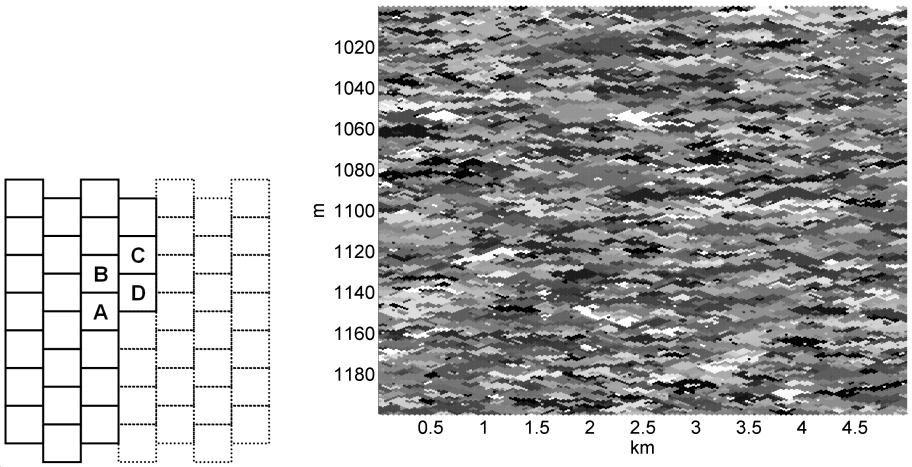


Figure 12. Input for example 2.The simulation grid, and neighborhood is displayed on the left side. Each column is shifted $\frac{1}{2}$ cell relative to to the previous column. A sample from the mosaic model is displayed on the right side.

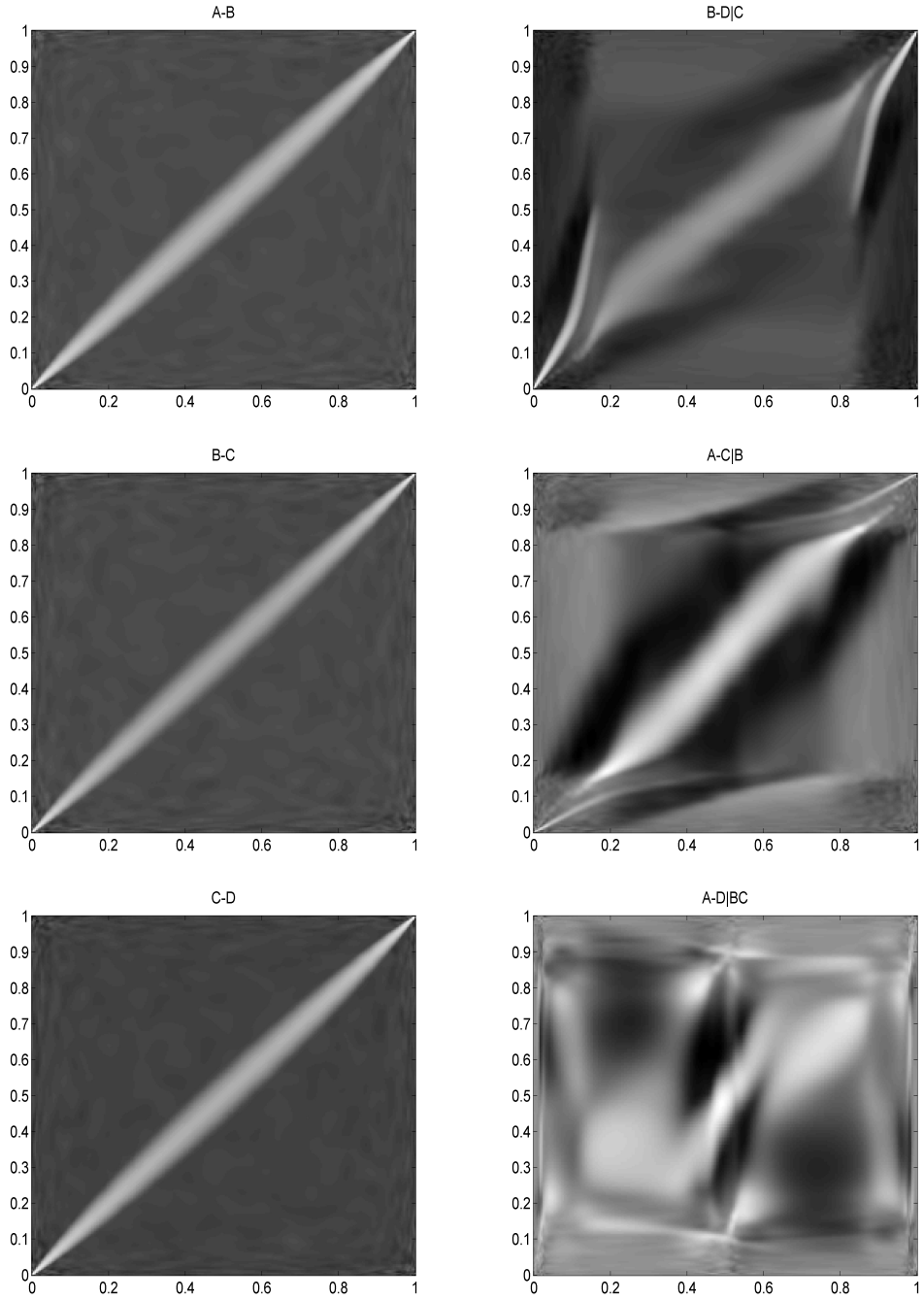


Figure 13. Estimated pair copula densities. In the left column is the copula densities from the first level. The top two in the right column is from the second level, the bottom right is from the third level. Dark color indicates low likelihood.

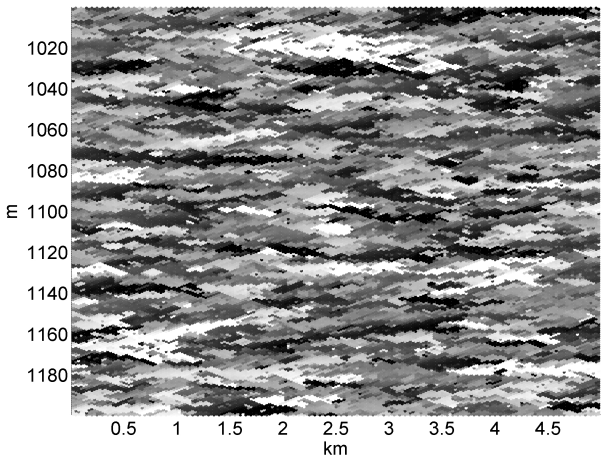


Figure 14. Simulation from a D-vine random field which approximate the mosaic model.

part is smoothed out due to the scatter plot smoothing.

7 Discussion and conclusions

We have described how D-vines can be used to create Random fields that contain non-linear features such as the mosaic model. In our examples the model reproduce non-linear features present in the training image.

There are, however, still unresolved issues. One issue is the question of ordering of explanatory variables that goes into the D-vine. A change in the ordering will generally result in a different multi-dimensional model. This calls for special considerations when the ordering is selected. It could also be that a model is better explained by using a pair-copula decomposition that is not in the class of D-vines. For this a detailed investigation of the structure in the problem is required.

The approach described is limited to sequences of bivariate interactions. It is however also possible to include quasi-multipoint interactions by regarding bivariate interactions between the target variable and a multipoint statistics, e.g. the sum of the variables in the sequential neighborhood. A further generalization of the approach is to investigate if it is possible to include higher order interactions by defining a triple-copula decomposition.

On the theoretical side a major challenge is to relate the transition probabilities in our model to the limiting distribution for the same variables. With respect to non parametric estimation it will be interesting to investigate sampling properties of the kernel estimators. In particular how the error propagates when copulas on one level is used to transform data onto the next level.

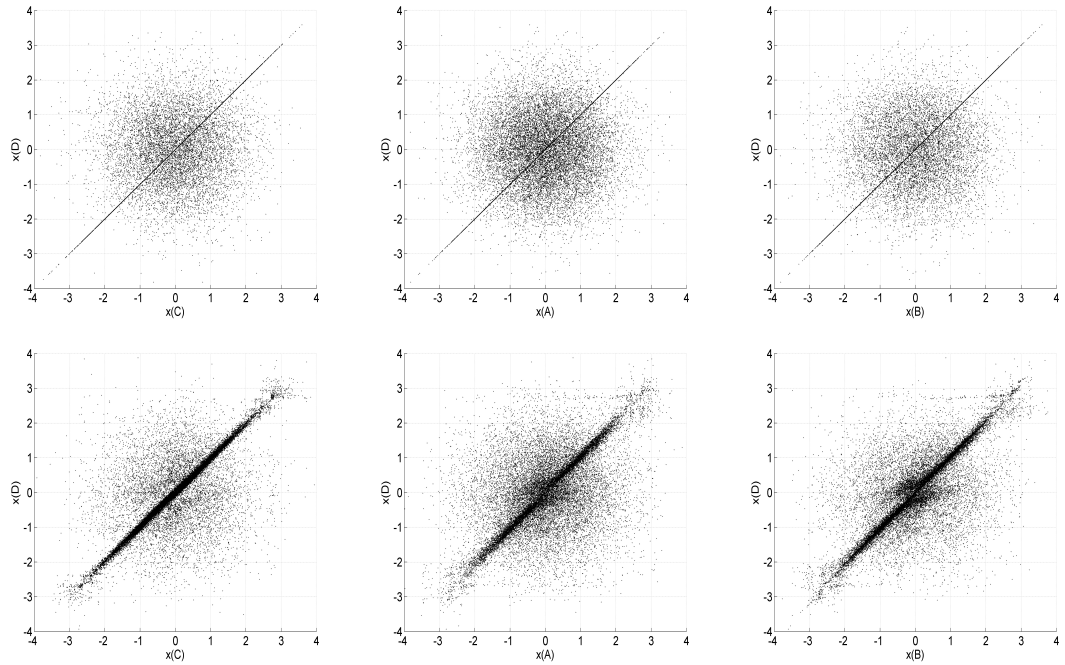


Figure 15. Comparison of scatterplots. Scatterplots for the original model is shown in the upper row, the corresponding plots for the approximate model is in the lower row.

References

Aas, K.; Czado, C.;, Frigessi, A.; and Bakken, H. *Pair-copula constructions of multiple dependence.* Insurance: Mathematics and Economics, (in press); 2008.

Bashore, W M.; Araktingi U.G.; Levy M; Schweller U.G. *Importance of a Geological Framework for Reservoir modelling and subsequent Fluid -Flow Predictions* AAPG Computer application in geology., No.3; Jeffery M. Yarus and Richard L Chambers eds.; 1994.

Bedford, T. and R. M. Cooke. *Vines - a new graphical model for dependent random variables.* Annals of Statistics 30 (4), 1031-1068; 2002.

Champagnat, F.; Idier, J.; and Goussard, Y. *Stationary Markov random fields on a finite rectangular lattice.* IEEE Transactions on: Information Theory. Vol. 44, No. 7 pp. 2901-2916, 1998.

Cressie, Noel A. C. *Statistics for Spatial Data;* Wiley: New York, 1993.

Gijbels,I. and Mielniczuk, J. *Estimating the denssity of a copula function* Communications statistics theory method, Vol. 19, No. 2, pp 445-464, 1990.

Joe, H. *Families of m-variate distributions with given margins and m(m-1)/2 bivariate dependence parameters.* In L. R¨uschendorf and B. Schweizer and M. D. Taylor (Ed.), Distributions with Fixed Marginals and Related Topics; 1996

Joe, H. *Multivariate Models and Dependence Concepts.* London: Chapman & Hall; 1997.

Journel, A.G. *Nonparametric estimation of spatial distributions.* Mathematical Geology Vol. 15, No. 3, pp 445-468, 1983.

Journel, A.G. *Modelling Uncertainty: some conceptual thoughts* In: Geostatistics for the next century (eds R. Dimitrakopoulos), Kluwer, Dordrecht, Holland,pp 30-43,1994

Kolbjørnsen, O. and Stien, M. *The D-Vine Creation of non-Gaussian Random Fields* Accepted for proceedings of Geostats 08, Chile. 2008.

Kurowicka, D. and Cooke, R. M. *Distribution - free continuous bayesian belief nets* In Fourth International Conference on Mathematical Methods in Reliability Methodology and Practice, Santa Fe, New Mexico, 2004.

Matheron, G. *Principles of geostatistics* Economic Geology Vol. 58, pp 1246-1266, 1963.

Matheron, G. *A simple substitute for conditional expectation: The disjunctive kriging* In advanced Geostatistics in the mining Industry. M. Guarascio, M. David and C Huijbregts, eds. Reidel, Dordrect, pp 221-236, 1976

Rivoirard J. (1994). *Introduction to Disjunctive Kriging and Non-linear Geostatistics;* New York: Oxford press, 1994.

Silverman B.W. *Density Estimation for Statistics and Data Analysis*.Chapman & Hall 1986.