Model selection for discrete regular vine copulas

Anastasios Panagiotelis, Claudia Czado, Harry Joe and Jakob Stöber

July, 2015

Abstract

Discrete vine copulas, introduced by Panagiotelis et al. (2012), provide a flexible modeling framework for high-dimensional data and have significant computational advantages over competing methods. A vine-based multivariate probability mass function is constructed from bivariate copula building blocks and univariate marginal distributions. However, even for a moderate number of variables, the number of alternative vine decompositions is very large and additionally there is a large set of candidate bivariate copulas that can be used as building blocks in any given decomposition. Together, these two issues ensure that it is infeasible to evaluate all possible vine copula models. In this paper we introduce two greedy algorithms for automatically selecting vine structures and component pair copula building blocks. The algorithms are tested in a simulation study that is itself driven by real world data from online retail. We show that both algorithms select vines that provide accurate estimates of the joint probabilities. Although the vine copulas selected are not exactly the same as the 'true' model in simulation studies, they are statistically indistinguishable from the true model according to the closeness test of Vuong (1989). Our algorithms outperform a Gaussian copula benchmark, especially for data with high dependence and also when predicting low probability tail events. Finally, we show that our selection algorithms outperform a Gaussian copula benchmark for data from the General Social Survey both in-sample and out-of-sample.

Keywords: model selection, count data, overfitting, tail asymmetry, tail dependence.

Dr. Anastasios Panagiotelis is a Lecturer at the Department of Econometrics and Business Statistics, Monash University. Claudia Czado is an Associate Professor at Zentrum Mathematik at the Technische Universität München. Harry Joe is a Professor at the Department of Statistics at the University of British Columbia. Dr. Jakob Stöber is a former PhD student and affiliate researcher at Technische Universität München. Anastasios Panagiotelis acknowledges the support of the Alexander von Humboldt Foundation, Claudia Czado is supported by a DFG grant and Harry Joe is supported by an NSERC Discovery grant. Address for correspondence is: Anastasios Panagiotelis, Econometrics and Business Statistics, Monash University Caulfield Campus, PO Box 197, Caulfield East, Victoria 3145, Australia. Email: anastasios.panagiotelis@monash.edu

1 Introduction

For many years, high-dimensional discrete datasets have been available in a large number of fields of statistical science including medicine and psychometrics. In recent years there has also been growth in multivariate discrete datasets in the field of marketing; for example, counts of online purchases made by an individual across different websites. There is an increasing need to find descriptive or predictive models that are able to capture the often complicated dependence found in such datasets.

A state of the art approach to modeling such data is to use discrete vine pair copula constructions, introduced by Panagiotelis et al. (2012); with applications to longitudinal data, the presentation of algorithms was made in the special boundary case of a D-vine. For the more general regular vine, the algorithm for log-likelihood is included in Joe (2014) (see Algorithm 7, Chapter 6). For a regular vine pair copula construction, a multivariate probability mass function is decomposed into bivariate copula functions and univariate marginal distributions. This approach has two major advantages. The first is that the computational complexity of computing the probability mass function grows quadratically with the dimension or number of variables, whereas for alternative approaches the computational complexity of computing the probability mass function grows exponentially with the dimension (see Nikoloulopoulos and Karlis (2008)). The second is that pair copula constructions are highly flexible since there is a large number of bivariate parametric copulas with different tail symmetry/asymmetry and tail dependence properties that can be used as building blocks in a pair copula construction. The vine copula approach can also be extended to the case where some margins are continuous and others are discrete (see Stöber et al. (2015)).

One issue left open in Panagiotelis et al. (2012) was model selection for discrete pair copula constructions. The number of alternative pair copula constructions is large and can be summarised and organised using graphical structures known as vines (Bedford and Cooke (2001)). In some cases the data suggest a specific vine structure, for example the D-vine can be used for intraday longitudinal data (see Panagiotelis et al. (2012) as well as Smith et al. (2010) for an example in the continuous case). However, in general there is a need for heuristic methods that automatically select two features of the model. The first is the vine structure which describes the way that the multivariate probability mass function is decomposed into bivariate building blocks, or edges of the vine, from conditional dependence relations. The second is to select bivariate parametric copula families to correspond to edges of the vine.

For high-dimensional data, it is infeasible to estimate all possible combinations of vine structure and bivariate parametric copula families on edges of the vine, and compare them on the basis of information criteria or other model comparison metrics. One way to tackle the problem of model selection for discrete regular vines would be to take a Bayesian approach and develop a reversible jump Markov chain Monte Carlo algorithms (Green (1995)) algorithm that searches high posterior probability regions of the model space. This approach was used by Min and Czado (2011) and Czado et al. (2013) to select bivariate copula families where the vine structure was assumed to be known. However, developing an algorithm that is also able to transition between different vine structures is a much more challenging problem. Although Gruber and Czado (2015a) and Gruber and Czado (2015b) have made progress in this area any MCMC based algorithm will be computationally slow especially for high-dimensional data.

Instead we take an alternative approach somewhat motivated by the famous maxim of George Box that ‘all models are wrong, some are useful’. Rather than try to select the somewhat artificial construct of the ‘true’ model, we aim to develop algorithms that are fast and identify a vine copula model that accurately captures nuances in the dependence structure of the data. Our contribution in this paper is to introduce two such algorithms for selecting discrete regular vines. The first adapts the sequential algorithm of Dissmann et al. (2013) for continuous vine copulas to the discrete case. The second is also a sequential algorithm, but aims to avoid overfitting by using cross-validation type ideas. Both our proposed approaches are greedy algorithms that can be contrasted to approaches found in Joe (2014) and Brechmann and Joe (2015). In Joe (2014) and Brechmann and Joe (2015) a truncated vine structure is selected under the initial assumption that pair copulas are Gaussian, and in a second step non-Gaussian copulas are assigned to some pairs where necessary to account for tail asymmetry and tail dependence relative to Gaussian. No heuristic method can be expected to be best for all data sets, and comparisons of algorithms for large multivariate discrete data sets is a topic of future research.

We will test both algorithms in a simulation study that is driven by a real dataset from the literature on online retail. We vary the strength of dependence and sample size in the simulation study to investigate how the algorithms perform for different types of data. Although, the selection algorithms do not select the 'true' model, they do choose models that are almost indistinguishable from the 'true' model in all but the most challenging setup, and over a wide range of criteria the selection algorithms we propose perform better than the benchmark of a multivariate Gaussian copula. In particular the models selected by our algorithms are able to accurately estimate the mass function at points of the domain that correspond to low probability events. To demonstrate the potential of our approach we look at a further application, the General Social Survey data and compare the performance of vines selected by our proposed algorithms to a Gaussian benchmark.

The following is a summary of the remainder of the paper. In Section 2 we provide necessary background on discrete pair copula constructions and vines including key definitions. In Section 3 we outline the two proposed selection algorithms in detail. In Section 4 we discuss the data upon which the simulation study is based. In Section 5 we describe the simulation study and present results. In Section 6 we introduce the General Social Survey dataset that has previously been modeled using a Gaussian copula and show that both algorithms lead to vine structures that improve the in-sample and out-of-sample fit compared to the Gaussian benchmark. In Section 7 we conclude and point to future avenues of research.

2 Background to discrete vine copulas

Vine copulas are an increasingly popular tool for the flexible modelling of multivariate data and comprehensive surveys of the literature may be found in Kurowicka and Joe (2011) and Czado (2010). Vine copulas have mostly been used to model continuous data, although Panagiotelis et al. (2012) and Nikoloulopoulos and Joe (2015) recently extended the idea of vine copulas to discrete data. The approach in Nikoloulopoulos and Joe (2015) involves latent variables to form factor copula models, and in this section we briefly reiterate the approach of Panagiotelis et al. (2012) which is suitable when there are no latent variables to explain the dependence in the observed variables.

2.1 Pair copula constructions for discrete data

Let $\mathbf{Y} = (Y_1, Y_2, \ldots, Y_m)$ be an m-dimensional random vector with realisation $\mathbf{y} = (y_1, y_2, \ldots, y_m)$. For clarity of exposition we assume $\mathbf{Y}$ has domain $\mathbf{Y} \in \mathbb{N}^m$ although a discrete domain with support on negative or non-integer values can be handled with no loss of generality. The probability mass function of $\mathbf{Y}$ can be decomposed as follows

$$\begin{aligned}\Pr(Y_1 = y_1, \ldots, Y_m = y_m) = \Pr(Y_1 = y_1 | Y_2 = y_2, \ldots, Y_m = y_m) \times \\ \Pr(Y_2 = y_2 | Y_3 = y_3, \ldots, Y_m = y_m) \times \cdots \times \Pr(Y_m = y_m)\,. \qquad (2.1)\end{aligned}$$

Each term on the right hand side of Equation (2.1) has the form $\Pr(Y_j = y_j | \mathbf{V} = \mathbf{v})$ where Y_j is a scalar element of $\mathbf{Y}$ and $\mathbf{V}$ is a subset of $\mathbf{Y}$. The conditioning set $\mathbf{V}$ can be broken down into a single element V_h and the remaining elements of $\mathbf{V}$ which will be denoted $\mathbf{V}_{\backslash h}$. The following expression is obtained

$$\begin{aligned}\Pr(Y_j = y_j | \mathbf{V} = \mathbf{v}) &= \frac{\Pr(Y_j = y_j, V_h = v_h | \mathbf{V}_{\backslash h} = \mathbf{v}_{\backslash h})}{\Pr(V_h = v_h | \mathbf{V}_{\backslash h} = \mathbf{v}_{\backslash h})} && (2.2)\\ &= \frac{\sum\limits_{i_j=0,1} \sum\limits_{i_h=0,1} (-1)^{i_j+i_h} \Pr(Y_j \le y_j - i_j, V_h \le v_h - i_h | \mathbf{V}_{\backslash h} = \mathbf{v}_{\backslash h})}{\Pr(V_h = v_h | \mathbf{V}_{\backslash h} = \mathbf{v}_{\backslash h})} && (2.3)\end{aligned}$$

By the theorem of Sklar (1959), the bivariate conditional probability in the numerator of (2.3) can be expressed in terms of a copula yielding

$$= \frac{\sum\limits_{i_j=0,1} \sum\limits_{i_h=0,1} (-1)^{i_j+i_h} C_{Y_j, V_h | \mathbf{V}_{\backslash h}}(F_{Y_j | \mathbf{V}_{\backslash h}}(y_j - i_j | \mathbf{v}_{\backslash h}), F_{V_h | \mathbf{V}_{\backslash h}}(v_h - i_h | \mathbf{v}_{\backslash h}))}{\Pr(V_h = v_h | \mathbf{V}_{\backslash h} = \mathbf{v}_{\backslash h})}\,, \qquad (2.4)$$

where $F_{A|B}(a|b)$ is used as generic notation for the distribution function of $\Pr(A \le a | B = b)$. The arguments of the copula functions in Equation (2.4) are evaluated using the following expression

$$F_{Y_j|V_h,\mathbf{V}_{\backslash h}}(y_j|v_h,\mathbf{v}_{\backslash h}) = \Big[C_{Y_j,V_h|\mathbf{V}_{\backslash h}}\Big(F_{Y_j|\mathbf{V}_{\backslash h}}(y_j|\mathbf{v}_{\backslash h}), F_{V_h|\mathbf{V}_{\backslash h}}(v_h|\mathbf{v}_{\backslash h})\Big) - C_{Y_j,V_h|\mathbf{V}_{\backslash h}}\Big(F_{Y_j|\mathbf{V}_{\backslash h}}(y_j|\mathbf{v}_{\backslash h}), F_{V_h|\mathbf{V}_{\backslash h}}(v_h-1|\mathbf{v}_{\backslash h})\Big)\Big]\Big/\Pr(V_h = v_h|\mathbf{V}_{\backslash h} = \mathbf{v}_{\backslash h})\,. \quad (2.5)$$

Note that the original expression $\Pr(Y_j = y_j|\mathbf{V} = \mathbf{v})$ has been decomposed so that it involves bivariate copula functions and conditional probabilities whose conditioning set $\mathbf{V}_{\backslash h}$ has been reduced by one element. The sequential application of Equation (2.4) to the terms in the right-hand side of Equation (2.1) is what allows the entire multivariate probability mass function to be expressed in terms of bivariate copula functions and univariate marginal probabilities. The approach to modelling high-dimensional multivariate discrete data suggested by Panagiotelis et al. (2012) turns this decomposition on its head. After fitting appropriate univariate marginal distributions, suitable parametric bivariate copulas are chosen to correspond to the pair copulas in the decomposition, and the end product is a flexible high-dimensional discrete distribution. For a more parsimonious model, a simplifying assumption that is implicitly made is that the functional form and parameters of pair copulas do not directly depend on the values of variables in the conditioning set; Sklar's theorem applied in (2.4) would yield $C_{Y_j,V_h|\mathbf{V}_{\backslash h}}(\cdot;\mathbf{v}_{\backslash h})$. If data showed some conditional dependence that is increasing in $\mathbf{v}_{\backslash h}$ say, our construction can accommodate this with parameters of pair copulas that depend on $\mathbf{v}_{\backslash h}$. See Panagiotelis et al. (2012) for a discussion on the restrictiveness of the simplifying assumption.

This approach to modeling multivariate discrete data is highly flexible, and comes with an additional computational advantage; the complexity of computing the likelihood only grows quadratically with respect to the dimension m and not exponentially as is generally the case for copula models for discrete data. However the approach comes at the cost of a difficult model selection issue, in addition to selecting appropriate families to use as pair-copulas building blocks there is also a choice to be made as to an appropriate decomposition. The number of ways in which a multivariate distribution can be decomposed into bivariate copula building blocks can be summarised and organised using *vines.* Vines are graphical models introduced by Bedford and Cooke (2001) and Bedford and Cooke (2002). We next briefly summarise some important concepts and direct the reader to Kurowicka and Cooke (2006) and Stöber and Czado (2012) for more detailed discussion.

2.2 Regular Vines

A vine is made up of a set of trees $\mathcal{T} = \{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_{m-1}\}$ where each tree $\mathcal{T}_s = \{\mathcal{N}_s, \mathcal{E}_s\}$ is made up of a set of nodes $\mathcal{N}_s$ and a set of edges $\mathcal{E}_s$. The nodes of the first tree simply index the variables, i.e. $\mathcal{N}_1 = \{1, 2, \ldots, m\}$, while the nodes of all subsequent trees are equal to the edges of the previous tree, i.e $\mathcal{N}_s = \mathcal{E}_{s-1}$ for $s = 2, \ldots, m-1$. To determine how edges are formed requires three definitions; the *complete union* the *conditioning set* and the *conditioned set*. Let $E_{s,g}$ be the edge g on $\mathcal{T}_s$ which connects the nodes N_{s,h_1} and N_{s,h_2}. The complete union of $E_{s,g}$, denoted $U_{E_{s,g}}$, is given by $U_{N_{s,h_1}} \cup U_{N_{s,h_2}}$ or equivalently $U_{E_{s-1,h_1}} \cup U_{E_{s-1,h_2}}$ (recall that all nodes on $\mathcal{T}_s$ are edges on $\mathcal{T}_{s-1}$). The conditioning set, denoted $D_{E_{s,g}}$, is given by $U_{N_{s,h_1}} \cap U_{N_{s,h_2}}$ or equivalently $U_{E_{s-1,h_1}} \cap U_{E_{s-1,h_2}}$. The conditioned set, denoted $C_{E_{s,g}}$, is given by the symmetric difference between the sets $U_{N_{g,h_1}}$ and $U_{N_{g,h_2}}$. In the context of pair copula constructions, each edge of a vine will correspond to a bivariate pair copula, with the copula corresponding to edge $E_{s,g}$ capturing the dependence in $C_{s,g}|D_{s,g}$ (two univariate conditional distributions).

To make this clearer we include an example of a 5-dimensional vine in Figure 1. There are four panels in the plot, each corresponding a different tree. To illustrate the concepts of complete union, conditioning set and conditioned set we focus out attention to the fourth tree in the the bottom right-hand panel of Figure 1. The nodes of $\mathcal{T}_4$ are $1, 4|2, 3$ and $4, 5|2, 3$ are equivalent to the edges on $\mathcal{T}_3$. The compete union of the edge of $\mathcal{T}_4$ is given by $\{1, 2, 3, 4\} \cup \{2, 3, 4, 5\} = \{1, 2, 3, 4, 5\}$. The conditioning set is given by $\{1, 2, 3, 4\} \cap \{2, 3, 4, 5\} = \{2, 3, 4\}$ and the conditioned set is given by $\{1, 5\}$ which is the symmetric difference between the complete union of the two nodes in $\mathcal{T}_4$. The pair copula that corresponds to the edge on $\mathcal{T}_4$ would capture the dependence between Y_1 and Y_5 given Y_2, Y_3, Y_4, where $\mathbf{Y} = (Y_1, Y_2, Y_3, Y_4, Y_5)'$ is the random vector we are trying to model.

——Figure 1 about here——

Not all vines can be used to obtain a pair copula construction, and a vine structure that is consistent with a valid pair copula construction is known as a regular vine or R-vine. The constraints required to ensure a vine is a regular vine are outlined in Dissmann et al. (2013). An important constraint is the so called *proximity condition.* This states that two nodes N_{s,h_1},N_{s,h_2} on tree $\mathcal{T}_s$ may only be connected if the equivalent edges E_{s-1,h_1},E_{s-1,h_2} on the previous tree $\mathcal{T}_{s-1}$ share a node. However even with these constraints, the number of possible R-vines is large even for moderate values of m (see Morales-Nápoles et al. (2013) for more detail).

3 Selection algorithm

A regular vine copula distribution is given by $\mathcal{V} = \{\mathcal{F}, \mathcal{T}, \mathcal{B}\}$ with the three elements of $\mathcal{V}$ corresponding to one of three modelling decisions. First $\mathcal{F} = \{F_1, F_2, \ldots, F_m\}$ is a set of univariate marginal distributions functions. An appeal of the copula approach is that completely different distributions can be used for each margin, for example, copula models could be used to combine a Negative Binomial distributed variable in one margin with a Poisson distributed variable in another margin. Also F_j can be distributions that are conditioned on some exogenous variables, and as a result copulas provide a straightforward way of extending regression style approaches such as generalised linear models to the multivariate case. The methods used to select and estimate univariate marginal models will depend on the application, in this paper we will abstract from this issue as much as possible (for example, in our simulation studies we assume the univariate marginal distribution functions are completely known). The second decision is to select a regular vine structure which is defined by the set of trees $\mathcal{T}$. The third decision is to select bivariate pair copulas $\mathcal{B}$. In our approach, we define a large set of candidate parametric bivariate copula families $\mathcal{C} = \left\{C^{\boldsymbol{\theta}^1}, \ldots, C^{\boldsymbol{\theta}^R}\right\}$, with one element of $\mathcal{C}$ chosen to correspond to each element of $\mathcal{B}$. The term $\boldsymbol{\theta}^r$ is used in the superscript to denote that the copulas are parametric copula families with parameters $\boldsymbol{\theta}^r$. Although efforts have been to model pair copulas using non-parametric techniques in the continuous case (Kauermann and Schellhase (2014), Nagler and Czado (2015)), non-parametric methods are much more difficult to implement in the discrete case since the copula is only unique on the product of the ranges of the univariate marginal distribution functions and not over the entire unit cube (see Genest and Nešlehová (2007) for more discussion).

Choices of parametric families for in $\mathcal{C}$ can be based on initial data analysis of bivariate margins to check for departures for fits of the Gaussian copula (or probit model or discretized multivariate Gaussian). Common departures are tail asymmetry or more dependence in the joint tails relative to Gaussian. Hua and Joe (2011) and Joe (2014) CHAPTER? use the concept of *tail orders* in the joint lower and upper tails to summarize tail asymmetry and tail dependence.

If C is a bivariate copula family, then other copula families can be obtained by reflection of Uniform(0,1) random variables. If $(U_1, U_2) \sim C$, then $(1 - U_1, 1 - U_2) \sim \widehat{C}$, where $\widehat{C}(u, v) = u + v - 1 + C(1 - u, 1 - v)$ is the survival copula associated with C. If C has tail asymmetry skewed to the joint upper tail, then $\widehat{C}$ has tail asymmetry skewed to the joint lower tail. If $C = \widehat{C}$, then C is said to be reflection symmetric. Also the reflections $(1-U_1, U_2) \sim C^{*1}$ and $(U_1, 1-U_2) \sim C^{*2}$ lead to associated copulas $C^{*1}(u, v) = v-C(1-u, v)$ and $C^{*2}(u, v) = u - C(u, 1 - v)$, which we call the 1-reflected and 2-reflected versions of C. If C is a family with positive dependence, then C^{*1} and C^{*2} are associated copula families with negative dependence, and typically have different tail skewness for the joint tails where one variable is larger and the other is smaller.

Some families which represent some departures from bivariate Gaussian dependence are the following: (a) t_ν copulas are reflection symmetric with upper and lower tail dependence (stronger than Gaussian); (b) bivariate Frank copulas are reflection symmetric with tail quadrant independence (weaker than Gaussian) and negative dependence can be accommodated; (c) Gumbel and survival Gumbel copulas have one joint tail with stronger dependence than Gaussian and can model positive dependence only; (d) Mardia-Takahasi-Clayton-Cook-Johnson (MTCJ) and Joe copulas and their survival versions can model positive dependence and extreme tail asymmetry where one joint tail has tail dependence and the other has tail quadrant independence. 1-reflected versions of (c) and (d) can handle negative dependence where the two joint tails, of one variable large and the other small, are quite different in behavior. Other 2-parameter copula families are given in Chapter 4 of Joe (2014).

3.1 Selection Algorithm 1

The first algorithm is an adaptation of the algorithm of Dissmann et al. (2013) to the discrete case and has the following steps. Recall that the data are $\mathbf{y}_i = (y_{i1}, y_{i2}, \ldots, y_{im})$ for $i = 1, 2, \ldots, n$.

1. If the marginal distribution functions $F_j(.)$ are known, then compute 'pseudo data' $u_{ij}^+ := F_j(y_{ij})$ and $u_{ij}^- := F_j(y_{ij} - 1)$ for $j = 1 \ldots, m$ and $i = 1, \ldots, n$, where y_{ij} is the value of the response for the j^{th} margin and the i^{th} observation. If the marginal distribution functions are not known then $u_{ij}^+ := \hat{F}_j(y_{ij})$ and $u_{ij}^- := \hat{F}_j(y_{ij} - 1)$ where $\hat{F}_j(.)$ is an estimate for the distribution function (for example if parametric models have been assumed for the margins, estimates for the distribution function can be obtained by estimating the marginal parameters by maximum likelihood).

2. For a given pair of margins $\{l_1, l_2\} \subset \{1, \ldots, m\}$

   (a) For the rth indexed parametric copula family, fit the copula $C^{\boldsymbol{\theta}^r}$ to the pseudo data for margin l_1 and l_2 by

   $$\hat{\boldsymbol{\theta}}^r = \arg\max_{\boldsymbol{\theta}^r} \ln L^r_{l_1,l_2}(\boldsymbol{\theta}^r) \tag{3.1}$$

   where,

   $$\begin{aligned} \ln L^r_{l_1,l_2}(\boldsymbol{\theta}^r) &= \sum_{i=1}^{n} \ln \Big(C^{\boldsymbol{\theta}^r}(u^+_{il_1}, u^+_{il_2}) - C^{\boldsymbol{\theta}^r}(u^+_{il_1}, u^-_{il_2}) \\ &\quad - C^{\boldsymbol{\theta}^r}(u^-_{il_1}, u^+_{il_2}) + C^{\boldsymbol{\theta}^r}(u^-_{il_1}, u^-_{il_2}) \Big) \end{aligned}$$

   (b) Compute a modified Akaike Information Criterion (AIC), that removes the effect of the margins, given by

   $$mAIC^r = -2 \ln L^r_{l_1,l_2}(\hat{\boldsymbol{\theta}}^r) - \ln L_{l_1} - \ln L_{l_2} + 2q_r \tag{3.2}$$

where q_r is the dimension of $\boldsymbol{\theta}^r$, $\ln L_{l_1} = \sum_{i=1}^{n} \ln(u_{il_1}^+ - u_{il_1}^-)$ and $\ln L_{l_2} = \sum_{i=1}^{n} \ln(u_{il_2}^+ - u_{il_2}^-)$. A smaller $mAIC$ value indicates a better parametric model.

(c) The edge weight for the pair $\{l_1, l_2\}$ is given by $mAIC^{r^*}$ where $r^* = \arg\min_r mAIC^r$.

(d) Using the weights computed in the step 4(d), find the minimum spanning tree to select the edges of $\mathcal{T}_1$. Any edge selected in $\mathcal{T}_1$ will have a corresponding pair copula given by $C^{\boldsymbol{\theta}^{r^*}}$.

(e) Compute new pseudodata $\mathbf{u}_{i,h_1|h_2}^+ := F_{h_1|h_2}(y_{ih_1}|y_{ih_2})$, $\mathbf{u}_{i,h_1|h_2}^- := F_{h_1|h_2}(y_{ih_1} - 1|y_{ih_2})$, $\mathbf{u}_{i,h_1|h_2}^+ := F_{h_2|h_1}(y_{ih_2}|y_{ih_1})$, $\mathbf{u}_{i,h_2-1|h_1}^- := F_{h_2|h_1}(u_{ih_2}|u_{ih_1})$ using Equation (2.5) and $C^{\hat{\boldsymbol{\theta}}^{r^*}}$ which is the copula chosen at step 4(c) evaluated at the estimate of the parameters obtained in step 2(a).

3. Repeat step 2 for all pairs of the new pseudodata making sure all pairs satisfy the proximity condition to select the edges of $\mathcal{T}_2$ and corresponding pair copulas. Also compute new pseudodata in a similar fashion as step 2(e).

4. Iterate to select the entire vine structure and corresponding pair copulas.

3.2 Selection Algorithm 2

Due to concern that the first algorithm may overfit, we introduce a second algorithm. This is also a sequential algorithm but is based on a cross validation style approach using log predictive scores. The algorithm involves the following steps.

1. Define K non-overlapping subsets of $\mathcal{I} = \{1, \ldots, n\}$ of roughly equal size which will be denoted $\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_K$. These will be used to divide the dataset into training samples $(\mathcal{I} \backslash \mathcal{I}_k)$ and test samples $\mathcal{I}_k$ for cross validation where $A \backslash B$ denotes the elements of set A not including the elements of set B. We will do this randomly, but every other way of dividing the data set would be valid too.

2. Compute pseudo data in the same fashion as step 1 of Selection Algorithm 1.

3. For a given pair of margins $\{l_1, l_2\} \subset \{1, \ldots, m\}$.

(a) For a given k, divide the data into a training and test sample. For the rth indexed copula family, fit the copula $C^{\boldsymbol{\theta}^r}$ to the training data for margin l_1 and l_2 by pairwise maximum likelihood. Explicitly

$$\hat{\boldsymbol{\theta}}^r = \underset{\boldsymbol{\theta}^r}{\arg\max} \ln L_{l_1,l_2}^{r(k)}\left(\boldsymbol{\theta}^r\right), \tag{3.3}$$

where

$$\begin{aligned} \ln L_{l_1,l_2}^{r(k)}\left(\boldsymbol{\theta}^r\right) &= \sum_{i \notin \mathcal{I}_k} \ln \Big(C^{\boldsymbol{\theta}^r}(u_{il_1}^+, u_{il_2}^+) - C^{\boldsymbol{\theta}^r}(u_{il_1}^+, u_{il_2}^-) \\ &\quad - C^{\boldsymbol{\theta}^r}(u_{il_1}^-, u_{il_2}^+) + C^{\boldsymbol{\theta}^r}(u_{il_1}^-, u_{il_2}^-) \Big). \end{aligned}$$

(b) Compute the log predictive score (see Gneiting and Raftery (2007)) for the k^{th} partition which is given by

$$\begin{aligned} S_k^r &= \sum_{i \in \mathcal{I}_k} \Big[\ln \Big(C^{\hat{\boldsymbol{\theta}}^r}(u_{il_1}^+, u_{il_2}^+) - C^{\hat{\boldsymbol{\theta}}^r}(u_{il_1}^+, u_{il_2}^-) \\ &\quad - C^{\hat{\boldsymbol{\theta}}^r}(u_{il_1}^-, u_{il_2}^+) + C^{\hat{\boldsymbol{\theta}}^r}(u_{il_1}^-, u_{il_2}^-) \Big) + \ln(u_{il_1}^+ - u_{il_1}^-) + \ln(u_{il_2}^+ - u_{il_2}^-) \Big]. \end{aligned}$$

(c) Repeat steps 3(a)–3(b) for all $k = 1, 2, \dots, K$ and compute $S^r = \sum_{k=1}^{K} S_k^r$ for all $r = 1, 2, \dots, R$.

(d) The edge weight for the pair $\{l_1, l_2\}$ is given by S^{r^*} where $r^* = \underset{r}{\arg\max} S^r$.

4. Using the weights computed in the step 3(d), find the maximum spanning tree to select the edges of $\mathcal{T}_1$. Any edge selected in $\mathcal{T}_1$ will have a corresponding pair copula given by $C^{\boldsymbol{\theta}^{r^*}}$.

5. Compute pseudo observations for the second tree in a similar fashion to step 2(e) of Algorithm 1.

6. Iterate through steps 2–5 to select the entire vine structure and corresponding pair copulas.

For both algorithms, the $\hat{\boldsymbol{\theta}}^{r^*}$ that are computed sequentially can be used as estimates for the entire vine copula. However, these estimates are not as efficient as evaluating full maximum likelihood estimation subsequent to selecting $\mathcal{V}$. The sequential estimates do however, provide good starting values that speed up the implementation of full maximum likelihood estimation.

4 Motivating example: comScore data

The simulation study in Section 5 will be driven by a real online retail dataset collected by comScore. ComScore are an analytics company who install passive tracking software that records the browsing and transaction activity of households who opt to use their services. Data have been collected on over two million households, although only a subsample of 100000 households is made available for academic purposes through the Wharton Retail Data Service (WRDS). Of those households, we restrict our attention to households that during 2007 visited *all* of the 6 following websites: amazon.com, apple.com, jcpenney.com, victoriassecret.com, expedia.com, orbitz.com. This results in a $m = 6$ dimensional dataset with $n = 1755$ observations which is manageable for the large scale simulation study we consider in Section 5. The response y_{ij} for household i and website j can take three values; $y_{ij} = 0$ if no purchase is made, $y_{ij} = 1$ if a single purchase is made and $y_{ij} = 2$ if two or more purchases are made. Although the full set of counts was available, the decision to censor the number of sales at 2 was made with the simulation studies in mind. By censoring the response, the domain of the data consists of $3^6 = 729$ unique points therefore it is feasible in the simulation study to compute the Kullback-Leibler divergence between different estimated models and the known data generating process.

The empirical marginal distributions of the data are summarised in Table 1. For all websites there is a high incidence of no sale being made, and for most websites the proportion of repeat sales is the lowest. The two exceptions to this are amazon.com and apple.com which in 2007 were the largest online retailers measured by total sales and total number of transactions respectively (Panagiotelis et al. (2014)). It should be noted that the majority of transactions for these two online retailers are low priced goods namely books in the case of amazon.com and 99 cent iTunes songs from apple.com. The proportion of repeat sales is lowest for the two travel websites, expedia.com and orbitz.com who are typically selling the highest priced goods such as plane tickets and hotel accommodation.

——Table 1 about here——

Since some individuals have a higher propensity to shop online it is reasonable to expect that there will be positive dependence between the margins. This is borne out by looking at the empirical pairwise Kendall's τ_b (see Agresti (2010)) which correct for ties and are summarised in Table 2. The pairs with the strongest dependence are those pairs which include amazon.com. An exception is that expedia.com shares its strongest dependence with apple.com which may be a result of these companies appealing to a similar demographic.

——Table 2 about here——

We can visualise this dependence structure when we apply Selection Algorithm 1 to the data. The set of candidate copulas $\mathcal{C}$ includes the bivariate Gaussian, MTCJ and Gumbel copulas as well as reflected versions of the MTCJ, Gumbel and Joe copulas. The copulas reflected around both axes are 'survival' copulas, while the terminology 1-reflected and 2-reflected will be used to denote copulas reflected around the first and second argument respectively. The selected vine structure is summarised in Table 3 and is depicted in Figure 2. Table 3 also indicates the parametric family of copula chosen for each pair. Asymmetric copulas are selected for some of the pairs which suggest there are nuances in the data which cannot necessarily be captured by copulas with symmetric dependence. The parameter estimates are also provided in Table 3. Since it is difficult to compare parameter values across different copula families, the Kendall's τ corresponding to each of the parameters was also computed. Note that Kendall's τ computed in this way do not correspond to Kendall's τ_b for a bivariate discrete distribution since the latter depend on the margins. Copmuting Kendall's τ for the different copulas does provide a simple way to compare their dependence on the same scale.

——Figure 2 and Table 3 about here——

5 Simulation Study

To assess the effectiveness of the two algorithms for selecting R-vines we consider the following simulation study. Data were simulated from the R-vine fitted in the previous section (summarised in Table 3) with margins given by the proportions in Table 1. Throughout the univariate marginal distributions are treated as known. The probability integral transforms of the data are then fitted using the following procedures.

1. As a naïve benchmark we consider the case where the data are independent. Since the margins are assumed to be known, this does not require any estimation.
2. As another benchmark we consider a multivariate Gaussian copula (discretized multivariate Gaussian) which we fit by maximum likelihood. The computational complexity of maximum likelihood estimation grows exponentially with the dimension although for $m = 6$, this is a feasible, albeit slow benchmark.
3. We apply Selection Algorithm 1 and estimate the parameters by maximum likelihood
4. We apply Selection Algorithm 2 with $K = 5$ and estimate the parameters by maximum likelihood. As a robustness check we also tried $K = 20$ although the differences in the final results were minimal.
5. We assume the true vine structure and pair copulas families are known and fit the parameters by maximum likelihood.

We note that the aim of both selection algorithms is not necessarily to select the 'true' model, and indeed the exact combination of R-vine structure and pair copulas is never selected by either algorithm even over 100 replications. For Gaussian vines for continuous variables, many different vines with bivariate Gaussian copulas on the edges lead to the same multivariate distribution. For discretized multivariate Gaussian for ordinal response variables, the vine pair copula construction with many different vines can be good approximations; examples are in Section 5.7.4 of Joe (2014). Hence for a multivariate distribution distribution that is not too far from discretized multivariate Gaussian, we can expect different discrete pair copula constructions to provide almost equally good fits.

Our aim with both selection algorithms is to find a vine structure and pair copula construction that is close to the true model that provides accurate estimates of the joint probability mass function. In light of this we use the following three diagnostics to evaluate our methods. The first is the Kullback-Leibler divergence given by $\sum_{\mathbf{y}\in D(\mathbf{Y})} (\ln(p_{\mathbf{y}}) - \ln(q_{\mathbf{y}}))\, p_y$, where $\mathbf{y}$ is each possible realisation in the domain of $\mathbf{Y}$, $D(\mathbf{Y})$, $p_{\mathbf{y}}$ is the probability that $\mathbf{Y} = \mathbf{y}$ implied by the true model, while $q_{\mathbf{y}}$ are the probabilities that $\mathbf{Y} = \mathbf{y}$ implied by a fitted model. Since one of the advantages of vines is that they allow for tail asymmetric dependence structures, it is more likely that vines are able to estimate the correct probabilities for low probability events that occur in the tail of the distribution. For this reason our second diagnostic for comparing the models is the contribution of one of the terms in the sum that makes up the Kullback-Leibler divergence, specifically the term corresponding to $\mathbf{y} = (2, 2, 2, 2, 2, 2)'$. In our application this would refer to a customer that makes repeat purchases at all websites. This customer could be considered as an attractive high-value customer to online retailers and it is not difficult to envision a cost function that would weight accurate forecasts of such a tail probability more heavily than the Kullback-Leibler divergence. Finally we conduct the test of Vuong (1989) at the 5% significance level for comparing two non-nested models. In all Vuong tests we conduct, one of these two models is the true model, while the other model is either a model selected by Selection Algorithm 1, a model selected by Selection Algorithm 2 or the Gaussian copula. A failure to reject the null indicates that although the selected model is misspecified, it is so close to the true model so as to be statistically indistinguishable. For these diagnostics we are unable to report results for the independent case since the independent case does not require any estimation.

Since the dependence in our data is quite weak, we conduct a further simulation study to investigate how well the selection algorithms work for data with strong dependence. In this study we scale up the Kendall's τ from Table 3 by a factor of 3, which results in a model with copula parameters summarised in Table 4. We then obtain new 'true' values of the parameters using the inverse of the bijective relationship used to obtain the Kendall's

τ in the fist place. Discrete data with a strong dependence structure are typically found in item response data for marketing as well as psychometric testing. Throughout the rest of the paper, the data simulated from the model with parameters in Table 3 will be referred to as the low dependence case, while the data simulated from the model with parameters in Table 4 will be referred to as the high dependence case. For both the high dependence and low dependence cases we consider a sample size of $n = 1755$, which is the sample size of the actual dataset, and also a sample size of $n = 17550$, which is a realistic scenario since the dataset available for academic purposes is only a small sub-sample of the full proprietary database. For all combinations of high and low dependence as well as large and small sample size, 100 replications of the simulation study were performed. The results are summarised in Table 5 with boxplots of the overall Kullback Leibler divergence and the contribution to Kullback Leibler divergence from the point $\mathbf{y} = (2, 2, 2, 2, 2, 2)'$ provided in Figure 3 and Figure 4 respectively.

——Table 5, Figure 3 and Figure 4 about here——

In all cases the true model has the lowest Kullback-Leibler divergence since any divergence is purely a result of uncertainty in the parameter estimates. In the low dependence case the Gaussian copula has a marginally lower Kullback-Leibler divergence on average compared to both selection algorithms although this is reversed for a sample size $n = 17550$. The models chosen by the selection algorithms estimate the tail probability more accurately than the Gaussian copula which lacks the flexibility to capture asymmetry in the dependence structure. The models selected by both algorithms are so close to the true model that a Vuong test at the 5% significance level is never able to distinguish between the two even for $n = 17550$. This contrasts with the Gaussian copula where the null of the Vuong test is rejected 7% of the time for $n = 1755$ and 59% of the time when $n = 17550$.

The boxplots demonstrate however that in the low dependence case it is difficult to distinguish between models. This is in not the case when we turn our attention to the high dependence case where it is much easier to pick up asymmetries in the dependence structure. In the right hand panels of Table 5 we observe that both selection algorithms perform much better than a Gaussian benchmark both for capturing tail probabilities and for the overall Kullback-Leibler divergence. In the high dependence for $n = 1755$ case there is never enough evidence to distinguish between the models selected by the two algorithms and the true model using a Vuong test. This is in contrast to the Gaussian copula where the Vuong test always results in the conclusion that the Gaussian copula is not close to the true model. The asymptotic properties of the Vuong test only become apparent when dependence is high and the sample size is $n = 17550$ and even in this highly challenging case the models selected by the selection algorithms are still statistically indistinguishable from the true model in approximately one quarter of the replications.

In summary we conclude that both selection algorithms always choose a model that accurately estimates the joint probability mass function, particularly for tail events. This is the case even when dependence is high and the Gaussian benchmark completely breaks down. Furthermore both selection algorithms provide similar results which suggests that any overfitting in Selection Algorithm 1 is not too severe. However, it should be noted that the computational demands of the two selection algorithms differ vastly. When $n = 1755$, Selection Algorithm 1 typically took roughly 3 minutes (180 seconds) on a personal computer with a 2.27 GHz CPU to select a model while Selection Algorithm 2 took over an hour. Maximum likelihood estimation took roughly 10 minutes to compute, although this was improved to about 6 minutes if the sequential estimates for a selection algorithm were used as starting values. For both algorithms the computational demands are lower than for estimation of the Gaussian copula which took over 90 minutes.

6 Application to General Social Survey

To futher demonstrate how the proposed selection algorithms lead to vine structures with good fit both in-sample and out-of-sample we consider a second application. These data were modelled using a Gaussian copula in Hoff (2007) and were originally sourced from the 1994 General Social Survey. We use data on 7 variables, namely income, parent's income (when child was 16), degree, maximum of mother and father's degree, number of children, parents' number of children and age of the respondent. All variables except age are ordered categorical variables. Since our objective is to study discrete data, we bin the age variable into four categories, 18–30, 31–45, 46–60 and 60+. In addition we reduced the number of categories for income from 21 to 5 by collapsing categories together. For the number of children, there were very few respondents with more than 5 children, so all of these observations were collapsed into a single 5+ children category. Similarly for the number of parents' children a single 9+ category was created. After removing missing observations, 464 observations remain.

Table 6 summarises the polychoric correlations between the variables. Note the presence of negative dependence which cannot be modelled simply by reversing the order of the categories for some of the variables, for example, if the categories for children, parent's children and age are reversed there will remain two negative associations. Initial data analysis was conducted and involved comparing Gaussian copulas and other bivariate copulas for each pair of variables to check for deviations from latent Gaussian distributions for bivariate margins. We found that there was often tail asymmetry relative to Gaussian. Gumbel or Survival Gumbel copulas were sometimes a better fit than Gaussian for two positively dependent variables with respectively more probability in the joint upper or lower tail and 1-reflected or 2-reflected Gumbel were sometimes a better fit than Gaussian for two negatively dependent variables.

——Table 6 about here——

To assess the simplifying assumption (Section 2.1) for tree 2 of the vine, we computed some Kendall τ_b values were computed for conditional distributions of two variables Y_i, Y_j given a third variable Y_k taking each categorical value. Because our algorithms aims to have stronger dependence in tree 1, we looked at cases where the bivariate dependence of (Y_i, Y_k) and (Y_j, Y_k) is stronger than that of (Y_i, Y_j). In these cases, the Kendall τ_b values varied a little with the category of Y_k and were overall not strong and showed no trends. Hence we proceed with the algorithms of Section 3 assuming the simplifying assumption is acceptable for approximations.

6.1 Evaluation of Algorithms

The margins were modeled by their empirical distribution functions and the following copula models were fitted to the data using the inference function for margins approach:

- A Gaussian copula benchmark (which in this context is equivalent to a multivariate ordered probit or discretized multivariate Gaussian model).
- A Vine copula selected by Selection Algorithm 1 with only bivariate Gaussian copulas used as candidate pair copulas.
- A Vine copula selected by Selection Algorithm 1 with a pair copula candidate set that included the bivariate Gaussian, MCTJ, Gumbel, Frank, Joe copulas and reflected versions thereof.
- A vine copula with the same candidate set as above but using selection algorithm 2.

The in-sample fit can be assessed using the log-likelihood, which is equivalent to doing model comparison by AIC or BIC since all models have the same number of parameters. To assess the out-of-sample forecasting properties we carry out a leave-one-out cross validation which requires the following steps for $l = 1, 2, \ldots, 464$.

1. Set the training sample to all observations excluding the l^{th} observation.
2. Using the training sample, estimate the distribution function of the margins with an empirical estimate and obtain u_{ij}^{+} and u_{ij}^{-} as defined in Section 3.
3. Where necessary run the relevant selection algorithm.
4. Estimate the copula parameters by maximising the likelihood with univariate margins fixed and the pair copulas selected in the previous step.
5. Using the fitted model as a predictive distribution, compute the log score for the l^{th} observation, which is simply the log probability mass function of the fitted model evaluated at the realisation of the l^{th} observation.

The summary measure of out-of-sample performance is the average log score over l.

6.2 Results

The results are summarised in Table 7. We observe that both algorithms select vines that outperform the multivariate Gaussian benchmark both in-sample and out-of-sample. The improvement in fit is largely due to the use of non-Gaussian bivariate pairs in the vine copula. The vine structure selected by selection algorithm 1 is summarised in Table 8 along with estimates of the copula parameters. Note that many of the selected copulas are non-Gaussian, including copulas with tail asymmetry and also copulas with permutation asymmetry. If the non-Gaussian pairs are replaced by Gaussian copulas, then the resulting model is in fact the worst performing model both in-sample and out-of-sample.

——Table 7 and Table 8 about here——

Finally, although the simulation study suggested that Selection Algorithm 2 offers little improvement over selection algorithm 1, for this application we observe that selection algorithm 2, which is based on cross validation, does lead to the best out-of-sample performance. Although the improvement is quite small, this result does suggests that selection algorithm 2 may have the greatest potential for problems that involve prediction.

7 Conclusion

We have developed two greedy algorithms that select both the structure and component bivariate pair copula building blocks of a discrete regular vine. In a simulation study we showed that both algorithms select vines that provide accurate estimates of the joint probability mass function, particularly for low probability events in the tail of the distribution. Even when the dependence between the data is high and the Gaussian benchmark breaks down, the vine copulas selected by both algorithms perform well. Since Selection Algorithm

1 is a faster algorithm we advocate its use, although the results from the application imply that Selection Algorithm 2 may be better for applications that require prediction.

For the data example in Section 6, the dependence is quite weak in trees 4–6. So we might want to compare with a truncated vine where the copulas in these three trees are replaced with conditional independence. For continuous response variables, algorithms for finding good truncated vines are given in Brechmann et al. (2012) and Brechmann and Joe (2015). It is topic of future research to use ideas from these truncated vine algorithms in combination with our selection algorithms in Section 3 for high-dimensional multivariate discrete models.

In this paper we only considered fairly low-dimensional cases since we wanted to compare both selection algorithms with a Gaussian benchmark, and show how tail asymmetric copula families are used in the pair copula construction. In analysis of other multivariate discrete data sets, with item responses from an instrument or ordinal responses from a survey, we find that the property of tail dependence or tail asymmetry relative to Gaussian is common. We believe the algorithms introduced here provide a major step forward for flexibly modelling complicated high-dimensional discrete data.

References

Agresti, A. (2010). *Analysis of ordinal categorical data (2nd edition).* John Wiley & Sons, New York.

Bedford, T. and Cooke, R. (2001). Probability density decomposition for conditionally dependent random variables. *Ann. Math. Artf. Intell.*, 32:245–268.

Bedford, T. and Cooke, R. (2002). Vines – A new graphical model for dependent random variables. *Annals of Statistics*, 30:1031–1068.

Brechmann, E., Czado, C., and Aas, K. (2012). Truncated regular vines in high dimensions with application to financial data. *Canadian Journal of Statistics*, 40:68–85.

Brechmann, E. and Joe, H. (2015). Vine copulas using fit indices. *Journal of Multivariate Analysis*, 138:53–75.

Czado, C. (2010). Pair-copula constructions of multivariate copulas. In Jaworki, P., Durante, F., Härdle, W., and W.Rychlik, editors, *Workshop on Copula Theory and its Applications*, pages 93–109. Springer.

Czado, C., Brechmann, E., and Gruber, L. (2013). Selection of vine copulas. In Jaworski, P., Durante, F., and Härdle, W., editors, *Copulae in Mathematical and quantitative finance*. Springer.

Dissmann, J., Brechmann, E., Czado, C., and Kurowicka, D. (2013). Selecting and estimating regular vine copulae and application to financial returns. *Computational Statistics and Data Analysis*, 59:52–69.

Genest, C. and Nešlehová, J. (2007). A primer on copulas for count data. *The Astin Bulletin*, 37:475–515.

Gneiting, T. and Raftery, A. (2007). Strictly proper scoring rules, prediction and estimation. *Journal of the American Statistical Association*, 102:359 – 378.

Green, P. (1995). Reversible jump Markov chian Monte Carlo computation and Bayesian model determination. *Biometrika*, 82:711 – 732.

Gruber, L. and Czado, C. (2015a). Bayesian model selection of regular vine copulas. Submitted.

Gruber, L. and Czado, C. (2015b). Sequential Bayesian model selection of regular vine copulas. *Bayesian Analysis*, To appear.

Hoff, P. (2007). Extending the rank likelihood for semiparametric copula estimation. *The Annals of Applied Statistics*, 1:265–283.

Hua, L. and Joe, H. (2011). Tail order and intermediate tail dependence of multivariate copulas. *Journal of Multivariate Analysis*, 102(10):1454–1471.

Joe, H. (2014). *Dependence Modelling with Copulas.* Chapman and Hall/CRC.

Kauermann, G. and Schellhase, C. (2014). Flexible pair-copula estimation in d-vines with penalized splines. *Statistics and Computing*, 24:1081–1100.

Kurowicka, D. and Cooke, R. (2006). *Uncertainty Analysis with High Dimensional Dependence Modelling*. Wiley Series in Probability and Statistics. Wiley, Chichester.

Kurowicka, D. and Joe, H., editors (2011). *Dependence Modeling: Vine Copula Handbook*, London. World Scientific Publishing Company.

Min, A. and Czado, C. (2011). Bayesian model selection for D-vine pair-copula constructions. *Canadian Journal of Statistics*, 39:239–258.

Morales-Nápoles, O., Cooke, R., and Kurowicka, D. (2013). About the number of vines and regular vines on n nodes. Submitted to Journal of Statistical Planning and Inference.

Nagler, T. and Czado, C. (2015). Evading the curse of dimensionality in multivariate kernel density estimation with simplified vine copulas. submitted.

Nikoloulopoulos, A. and Karlis, D. (2008). Multivariate logit copula model with an application to dental data. *Statistics in Medicine*, 27:6393–6406.

Nikoloulopoulos, A. K. and Joe, H. (2015). Factor copula models for item response data. *Psychometrika*, 80(1):126–150.

Panagiotelis, A., Czado, C., and Joe, H. (2012). Pair copula constructions for multivariate discrete data. *Journal of the American Statistical Association*, 107:1063 – 1072.

Panagiotelis, A., Smith, M., and Danaher, P. (2014). From Amazon to Apple: Modeling online retail sales, purchase incidence and visit behavior. *Journal of Business and Economic Statistics*, 32:14 – 29.

Sklar, A. (1959). Fonctions de répartition à n dimensions et leur marges. *Publ. Inst. Statist. Univ. Paris*, 8:229–231.

Smith, M., Min, A., Almeida, C., and Czado, C. (2010). Modelling longitudinal data using a pair-copula decomposition of serial dependence. *Journal of the American Statistical Association*, 105:1467–1479.

Stöber, J. and Czado, C. (2012). Sampling pair copula constructions with applications to mathematical finance. In Mai, J.-F. and Scherer, M., editors, *Simulating copulas: Stochastic Models, Sampling algorithms and applications*.

Stöber, J., Hong, H., Czado, C., and Ghosh, P. (2015). Comorbidity of chronic diseases in the elderly: Patterns identified by a copula design for mixed responses. *Computational Statistics and Data Analysis*, 88:28–39.

Vuong, Q. (1989). Likelihood ratio tests for model selection and non-nested hypotheses. *Econometrica*, 57:307–333.

	No Sale	Single Sale	Repeat Sale
	($y_{ij} = 0$)	($y_{ij} = 1$)	($y_{ij} = 2$)
amazon.com	0.611	0.175	0.214
apple.com	0.687	0.079	0.234
jcpenney.com	0.854	0.090	0.056
victoriassecret.com	0.775	0.135	0.089
expedia.com	0.893	0.080	0.027
orbitz.com	0.915	0.072	0.013

Table 1: Proportion of no sales, single sales and repeat sales for each website.

	Am.	Ap.	J.C.P.	V.S.	Exp.	Orb.
Amazon (Am.)	1	0.238	0.201	0.206	0.089	0.123
Apple (Ap)	0.147	1	0.038	0.094	0.092	0.046
J.C. Penney (J.C.P.)	0.107	0.020	1	0.072	0.110	0.019
Victoria's Secret (V.S.)	0.125	0.055	0.027	1	0.072	0.024
Expedia (Exp.)	0.038	0.047	0.048	0.028	1	0.099
Orbitz (Orb.)	0.057	0.022	0.004	0.011	0.035	1

Table 2: Polychoric correlations and Pairwise Kendall's τ (with correction for ties) between the data discussed in Section 4. The abbreviations of company names in parentheses will be used throughout. Numbers in the upper triangle are polychoric correlations and numbers in the lower table are Kendall's τ_b

	Pair Copula	Copula Family	Parameter	Kendall's τ
Tree 1	Am.,Ap.	Normal	0.2382	0.1531
	Am.,V.S.	Surv. Gumbel	1.2032	0.1689
	Am.,J.C.P.	Normal	0.2006	0.1286
	Ap., Exp.	MCTJ	0.3225	0.1388
	Am.,Orb.	Surv. Gumbel	1.1257	0.1117
Tree 2	V.S.,Ap.|Am.	Gumbel	1.0293	0.0285
	J.C.P.,V.S.|Am.	Surv. MCTJ	0.0381	0.0187
	Exp.,Am.|Ap.	Gumbel	1.0373	0.0360
	Orb., Ap.|Am.	Normal	0.0296	0.0189
Tree 3	J.C.P., Ap.|V.S.,Am.	Gumbel	1.0062	0.0062
	V.S., Exp.|Ap.,Am.	Gumbel	1.0246	0.0240
	Orb.,Exp.|Ap.,Am.	Surv. MCTJ	0.0537	0.0261
Tree 4	J.C.P.,Exp.|V.S.,Ap.,Am.	Surv. Gumbel	1.1032	0.0936
	V.S.,Orb.|Exp.,Ap.,Am.	MCTJ	0.0243	0.0120
Tree 5	J.C.P.,Orb.|V.S.,Exp.,Ap.,Am.	1-Refl. MCTJ	-0.0261	-0.0129

Table 3: Summary of the vine copula selected by applying Algorithm 1 to the data and estimating parameters by maximum likelihood. Kendall's τ are obtained by using the bijection between copula parameters and Kendall's τ and are evaluated at the maximim likelihood estimates. Note that the bijective relationship provides a valid estimate of Kendall's τ in the continuous case only, however here the primary aim is to compare parameter values with differing domains on the same scale.

	Pair Copula	Copula Family	Parameter	Kendall's τ
Tree 1	Am.,Ap.	Normal	0.6606	0.4594
	Am.,V.S.	Surv. Gumbel	2.0266	0.5066
	Am.,J.C.P.	Normal	0.5696	0.3858
	Ap., Exp.	MCTJ	1.4278	0.4165
	Am.,Orb.	Surv. Gumbel	1.5038	0.3350
Tree 2	V.S.,Ap.|Am.	Gumbel	1.0935	0.0855
	J.C.P.,V.S.|Am.	Surv. MCTJ	0.1187	0.0560
	Exp.,Am.|Ap.	Gumbel	1.1209	0.1079
	Orb., Ap.|Am.	Normal	0.0888	0.0566
Tree 3	J.C.P., Ap.|V.S.,Am.	Gumbel	1.0190	0.0186
	V.S., Exp.|Ap.,Am.	Gumbel	1.0775	0.0719
	Orb.,Exp.|Ap.,Am.	Surv. MCTJ	0.1702	0.0784
Tree 4	J.C.P.,Exp.|V.S.,Ap.,Am.	Surv. Gumbel	1.3903	0.2807
	V.S.,Orb.|Exp.,Ap.,Am.	MCTJ	0.0747	0.0360
Tree 5	J.C.P.,Orb.|V.S.,Exp.,Ap.,Am.	1-Refl. MCTJ (90)	-0.0803	-0.0386

Table 4: Summary of model parameters for high dependence case. These parameters are obtained by scaling the Kendall's τ from Table 3 up by a factor of three and then using the bijection between the Kendall's τ and copula parameters to recover values of the copula parameters. Note that the bijective relationship provides a valid estimate of Kendall's τ in the continuous case only, however here the primary aim is to compare parameter values with differing domains on the same scale.

	Low dependence			*High dependence*		
n=1755	KL div.	Tail Prob.	Vuong	KL div.	Tail Prob.	Vuong
	$(\times 10^{-2})$	$(\times 10^{-7})$		$(\times 10^{-2})$	$(\times 10^{-7})$	
Independence	3.5471	111.321	-	36.0631	1120.254	-
Gaussian	0.4766	2.901	0.07	7.092	427.3815	1.00
Sel. Alg. 1	0.6282	0.589	0.00	0.9160	1.437	0.00
Sel. Alg. 2	0.6417	0.254	0.00	0.9390	-1.563	0.00
True	0.3838	2.399	-	0.4185	-2.328	-

	Low dependence			*High dependence*		
n=17550	KL div.	Tail Prob.	Vuong	KL div.	Tail Prob.	Vuong
	$(\times 10^{-2})$	$(\times 10^{-7})$		$(\times 10^{-2})$	$(\times 10^{-7})$	
Independence	3.5471	111.321	-	36.0631	1120.254	-
Gaussian	0.0772	0.0356	0.59	7.1691	4570.019	1.00
Sel. Alg. 1	0.0732	0.0369	0.00	0.2587	175.767	0.73
Sel. Alg. 2	0.0759	0.0057	0.00	0.2622	170.619	0.75
True	0.0401	0.0942	-	0.0426	-0.2649	-

Table 5: Summary of results from simulation study where numbers are averages over 100 replications. ‘Independence’ refers to the case where we assume margins are independent, ‘Gaussian’ refers to assuming a 6-variate Gaussian copula, Sel. Alg. 1 refers to the vine copula selected by Selection Algorithm 1 with parameters estimated by maximum likelihood, Sel. Alg. 2 refers to the vine copula selected by Selection Algorithm 2 with parameters estimated by maximum likelihood, while ‘True’ refers to the case where we assume the true vine structure is known but the parameters are estimated by maximum likelihood. KL div. refers to the exact Kullback-Leibler divergence between the true model and all of the estimated models. Tail Prob. is the contribution to the Kullback-Leibler divergence of a single point of the domain, namely the point $\mathbf{Y} = (2, 2, 2, 2, 2, 2)'$. ‘Vuong’ is the proportion of the 100 replications for which the null that the assumed vine structure and the true vine structure are equally close is rejected. The results on the left hand side corresponds to the model with parameters summarised in Table 3, while results on the right hand side correspond to parameters summarised in Table 4. The top set of results correspond to n=1755, the bottom set of resutls to n=17550.

	INC	PINC	CHD	PCHD	DEG	PDEG	AGE
INC	1	0.17	0.16	-0.11	0.52	0.26	0.29
PINC	0.11	1	-0.17	-0.23	0.21	0.44	-0.10
CHD	0.13	-0.11	1	0.20	-0.12	-0.24	0.57
PCHD	-0.08	-0.16	0.15	1	-0.26	-0.34	0.09
DEG	0.38	0.15	-0.08	-0.19	1	0.46	0.03
PDEG	0.18	0.34	-0.17	-0.26	0.34	1	-0.20
AGE	0.23	-0.07	0.43	0.07	0.03	-0.15	1

Table 6: Matrix of polychoric correlations and Kendall's τ_b for the data in in Section 6. The variables are INC= income, PINC= parent's income, CHD= No. of Children, PCHD= Parents' no. of children, DEG = highest degree attained, PDEG = highest degree obtained by either parent, AGE = age. Numbers in the upper triangle are polychoric correlations and numbers in the lower table are Kendall's τ_b.

Method	In-Sample	Out-of-sample
Gaussian Copula	-4414.9	-9.635
Sel. Alg 1	-4387.9	-9.621
Sel. Alg. 2	-4390.5	-9.614
Sel. Alg 1 (Gauss. pairs)	-4430.5	-9.653
Sel. Alg 1 (3-Truncated)	-4396.0	-9.607

Table 7: Summary of results for the application in 6. The method 'Gaussian copula' refers to a multivariate Gaussian copula, while 'Sel. Alg. 1' and 'Sel. Alg. 2' refer to the vines selected by selection algorithm 1 and selection algorithm 2 respectively. The method 'Sel.Alg 1 (Gauss. pairs)' refers to the vine structure selected by Selection Algorithm 1, but with all pair copulas replaced by bivariate Gaussians, while Sel.Alg.1 (3-truncated) is the vine selected by Selection Algorithm 1 but with all pair copulas from the 4th tree onwards set to independence copulas. The measure of in-sample fit is negative log likelihood, while the measure of out-of-sample fit is the average log score from a leave-one-out cross validation.

	Pair Copula	Copula Family	Parameter	Tau
Tree 1	DEG, PDEG	Surv. Gumbel	1.442	0.307
	PDEG, PCHD	1-Refl. MCTJ	-0.596	-0.229
	PDEG, PINC	Surv. Gumbel	1.416	0.294
	DEG, INC	Normal	0.488	0.325
	INC,AGE	MCTJ	0.524	0.208
	AGE, CHD	Surv. Gumbel	1.668	0.400
Tree 2	DEG, PCHD|PDEG	Frank	-0.908	-0.100
	PCHD,PINC|PDEG	1-Refl. Joe	-1.107	-0.058
	PDEG,INC|DEG	Gumbel	1.037	0.036
	DEG,AGE|INC	2-Refl Joe	-1.178	-0.093
	INC,CHD|AGE	MCTJ	0.082	0.039
Tree 3	DEG,PINC|PCHD, PDEG	Surv. MCTJ	0.024	0.012
	PCHD,INC|PDEG, DEG	Surv. MCTJ	0.054	0.026
	PDEG, AGE|DEG,INC	2-refl. Joe	-1.310	-0.149
	DEG,CHD|INC, AGE	Frank	-1.242	-0.136
Tree 4	PINC,INC|PDEG, DEG, PCHD	Joe	1.104	0.056
	PCHD, AGE|DEG,INC,PDEG	2-refl. Joe	-1.050	-0.028
	PDEG,CHD|DEG,INC, AGE	Normal	-0.137	-0.088
Tree 5	PINC, AGE|DEG,INC,PCHD,PDEG	1-refl. Joe	-1.079	-0.044
	PCHD,CHD|DEG,PDEG,INC, AGE	Frank	0.884	0.097
Tree 6	PINC,CHD|PCHD, DEG,PDEG,INC, AGE	2-Refl. Gumbel	-1.047	-0.045

Table 8: Summary of vine selected by Selection Algorithm 1 for the application is Section 6. The variable names are abbreviated in the same way as Table 6. To compare the dependence from different copulas on the same scale the bijection with Kendall's τ is used as was the case for Tables 3 and 4

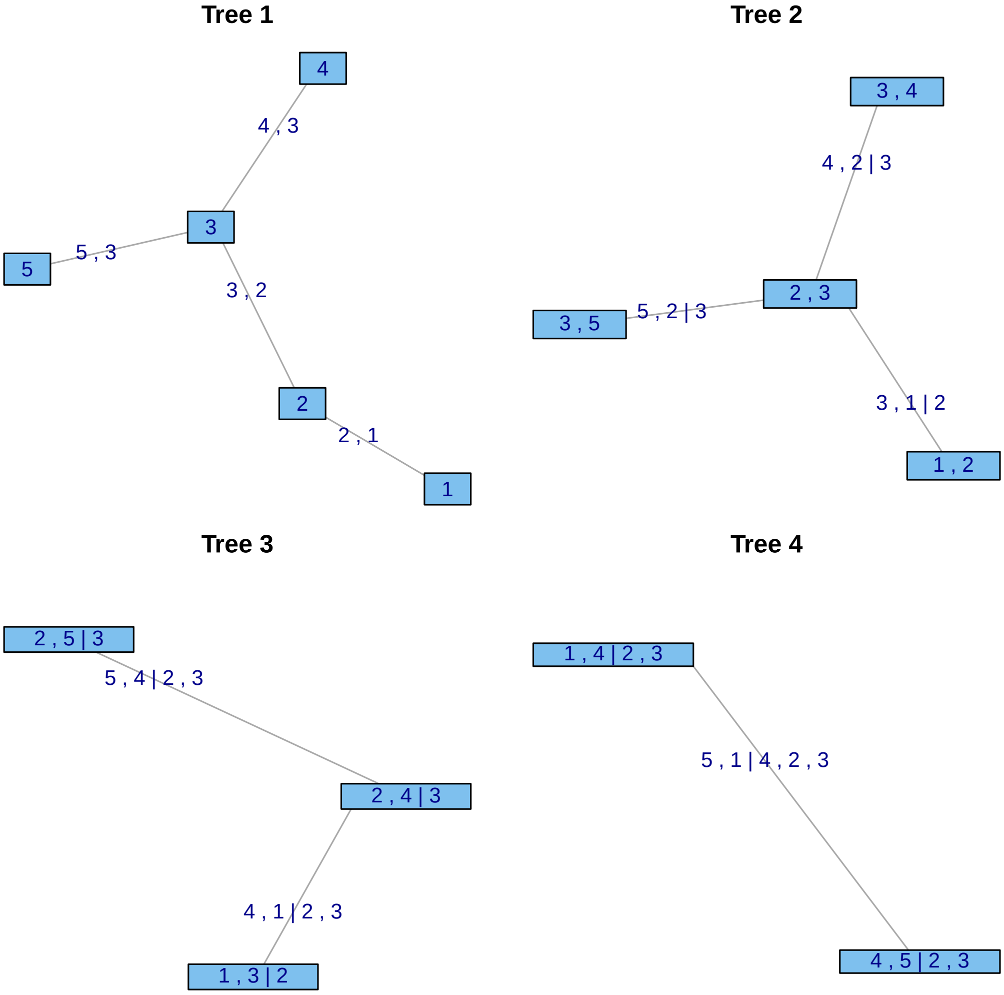

Figure 1: An example of a 5 dimensional regular vine where each panel corresponds to a different tree.

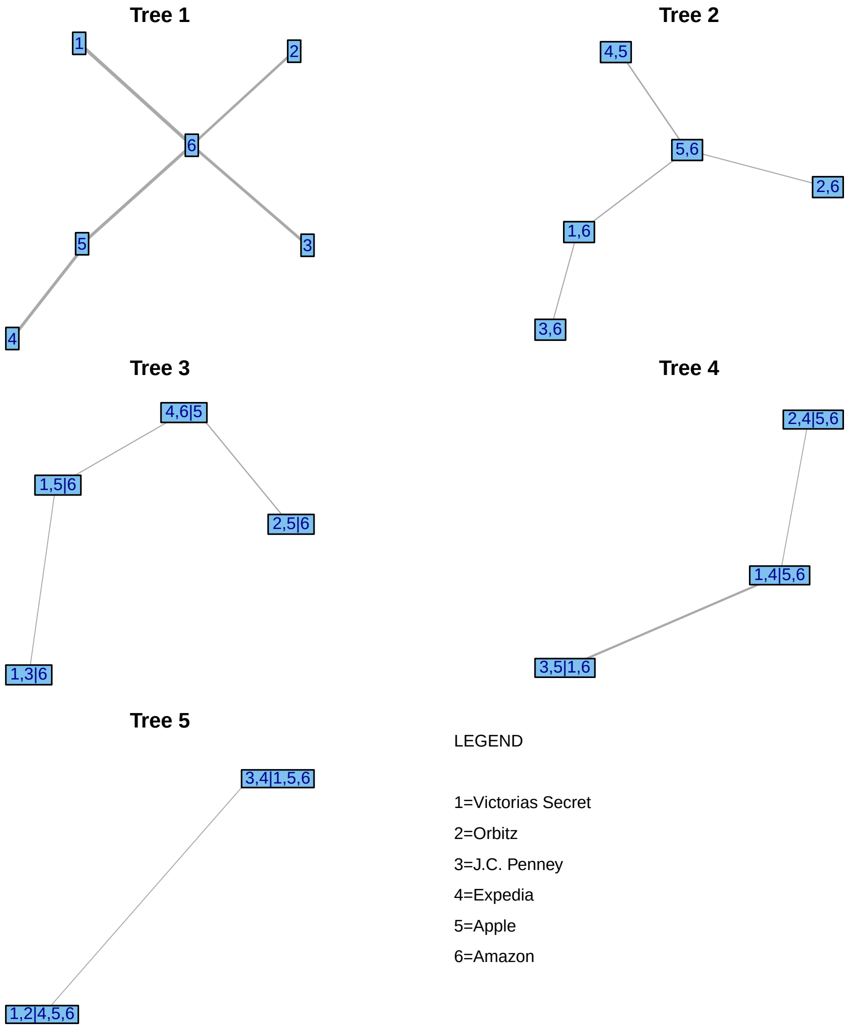

Figure 2: The vine structure selected when Selection Algorithm 1 is applied to the data described in section 4. Each panel corresponds to a different tree.

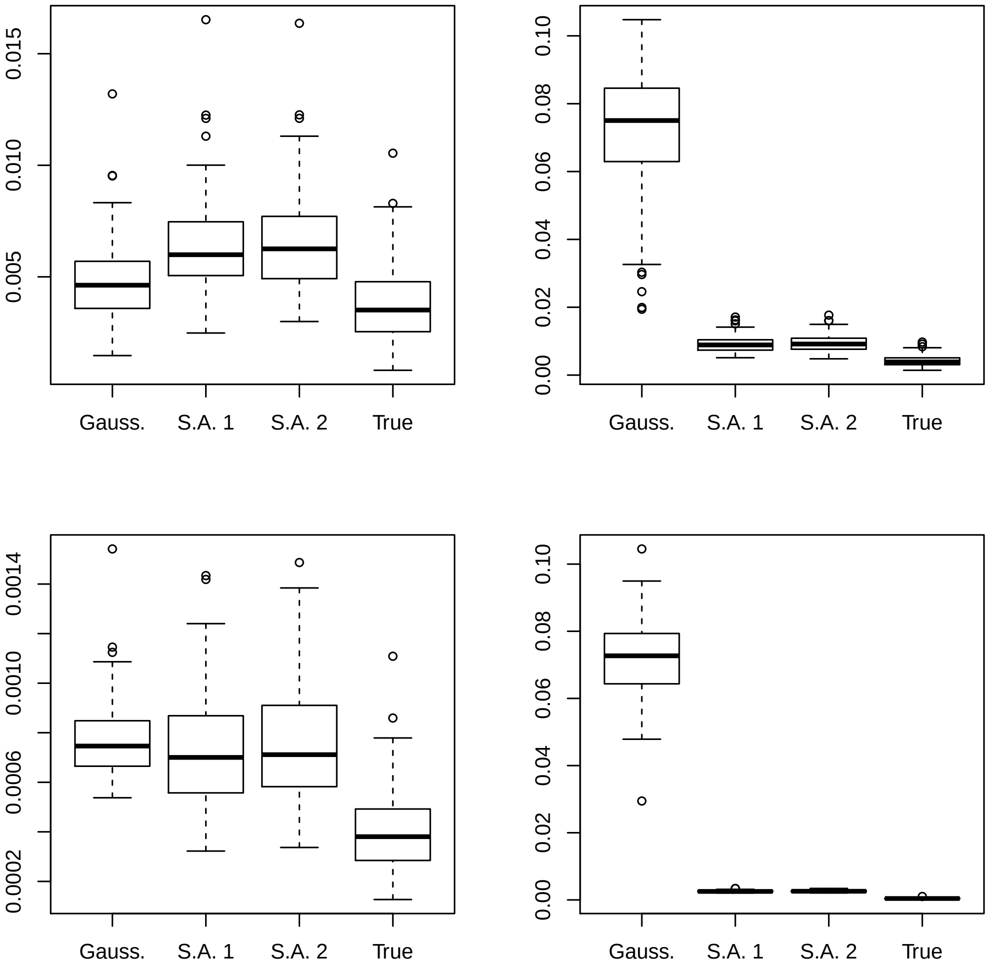

Figure 3: Boxplots of the Kullback Leibler divergence over 100 replications between estimated models and the true model. The panels on the left refer to the low-dimensional case (true parameter values are summarised in Table 3) while panels on the right refer to the high-dimensional case (true parameter values are summarised in Table 4). The top panels refer to a sample size of $n = 1755$, the bottom panels refer to a sample size of $n = 17550$. In all panels 'Gauss.' refers to estimates where a 6-dimensional Gaussian copula is assumed, 'Sel. Alg. 1' refers to estimates for the model selected by Selection Algorithm 1,'Sel. Alg. 2' refers to estimates for the model selected by Selection Algorithm 2, and 'True' refers to estimates where it is assumed the true model is known.

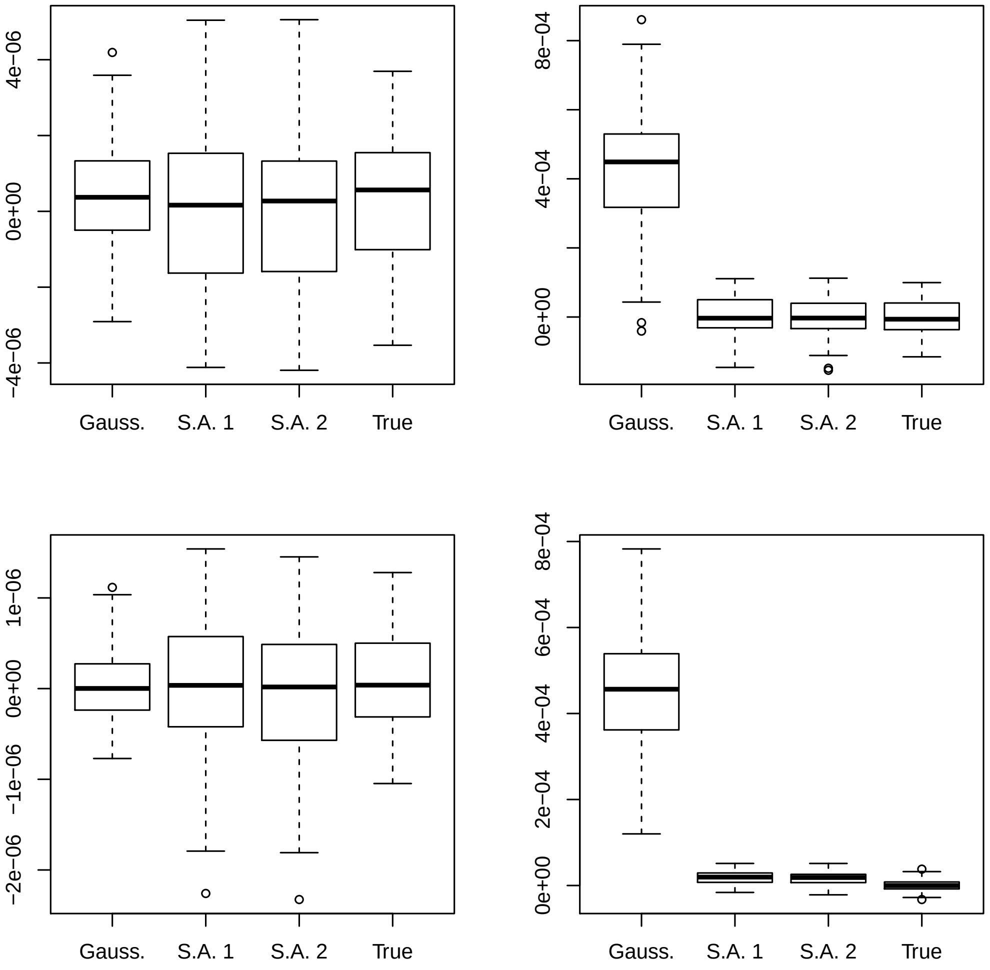

Figure 4: Boxplots of the contribution of the single point $\mathbf{y} = (2, 2, 2, 2, 2, 2)'$ of the domain to the Kullback Leibler divergence over 100 replications between estimated models and the true model. The panels on the left refer to the low-dimensional case (true parameter values are summarised in Table 3) while panels on the right refer to the high-dimensional case (true parameter values are summarised in Table 4). The top panels refer to a sample size of $n = 1755$, the bottom panels refer to a sample size of $n = 17550$. In all panels 'Gauss.' refers to estimates where a 6-dimensional Gaussian copula is assumed, 'Sel. Alg. 1' refers to estimates for the model selected by Selection Algorithm 1,'Sel. Alg. 2' refers to estimates for the model selected by Selection Algorithm 2, and 'True' refers to estimates where it is assumed the true model is known.