D-vine quantile regression with discrete variables

Niklas Schallhorn Daniel Kraus* Thomas Nagler Claudia Czado

Zentrum Mathematik, Technische Universität München

May 24, 2017

Abstract

Quantile regression, the prediction of conditional quantiles, finds applications in various fields. Often, some or all of the variables are discrete. The authors propose two new quantile regression approaches to handle such mixed discrete-continuous data. Both of them generalize the continuous D-vine quantile regression, where the dependence between the response and the covariates is modeled by a parametric D-vine. D-vine quantile regression provides very flexible models, that enable accurate and fast predictions. Moreover, it automatically takes care of major issues of classical quantile regression, such as quantile crossing and interactions between the covariates. The first approach keeps the parametric estimation of the D-vines, but modifies the formulas to account for the discreteness. The second approach estimates the D-vine using continuous convolution to make the discrete variables continuous and then estimates the D-vine nonparametrically. A simulation study is presented examining for which scenarios the discrete-continuous D-vine quantile regression can provide superior prediction abilities. Lastly, the functionality of the two introduced methods is demonstrated by a real-world example predicting the number of bike rentals.

Keywords: quantile regression; discrete variables; continuous convolution; nonparametric; vine copulas

1 Introduction

Quantile regression, the estimation of quantiles of a response random variable conditioned covariates, has gained importance in various fields since its first appearance in Koenker and Bassett (1978). Kraus and Czado (2017) propose a new method of quantile regression, where the dependence between the response and the covariates is modeled by a parametric D-vine as introduced in Aas et al. (2009). The D-vine is estimated by sequentially adding variables to the model until none of the remaining variables provides additional information. The D-vine approach remedies various shortcoming of classical quantile regression. The models are flexible and parsimonious, they prevent quantile crossing, and interactions between covariates are automatically taken into account. Kraus and Czado (2017) show that the D-vine quantile regression is a competitive approach that often shows superior prediction quality.
However, the model proposed by Kraus and Czado (2017) requires that the marginal distributions of the response and all of the covariates are continuous. Genest and Nešlehová (2007) give an overview of the difficulties of copula models with discrete variables. Implications are, for instance, that the copula is no longer uniquely defined and that the dependence between variables is not captured by the copula alone, but also involves the discrete marginal distributions. Taking these implications into account, Panagiotelis et al. (2012) present an algorithm to fit vine copula models to purely discrete data. Onken and Panzeri (2016) modify this

*Corresponding author: daniel.kraus@tum.de

algorithm to allow for cases where only some of the variables are discrete. Quantile regression methods that can handle discrete data are, e.g., linear quantile regression (Koenker and Bassett, 1978), additive quantile regression (Koenker, 2011; Fenske et al., 2012), and kernel quantile regression (Li et al., 2013).

In this paper, we modify the continuous D-vine quantile regression from Kraus and Czado (2017) in two ways. The first extends the formulas of the parametric model of Kraus and Czado (2017) such that it can handle mixed discrete-continuous data. In contrast to the purely continuous setting, the conditional quantiles cannot be expressed in closed form but can be calculated by numerically inverting the conditional distribution function. The second approach replaces the parametric estimation of pair-copulas by a nonparametric kernel density estimator. Discrete variables are handled by adding a small amount of noise which makes them continuous. As shown by Nagler (2017), the resulting estimator is still a valid estimator of the discrete-continuous conditional quantile function. Thereby, the simplicity of the continuous D-vine quantile regression is preserved in the discrete-continuous setting when using a nonparametric estimation approach.

The remainder of the paper is organized as follows. Section 2 introduces the continuous D-vine quantiles regression including the necessary concepts of D-vine copulas, while Section 3 presents the two approaches described above to handle discrete data. A simulation study that compares the two discussed methods to several competitor methods is shown in Section 4. Section 5 applies the proposed methods to a real-world example of bike rentals. Finally, Section 6 draws conclusions and gives an outlook to areas of further research.

2 Parametric D-vine quantile regression for continuous variables

This section is a summary of what is explained in more detail in Sections 2 and 3 of Kraus and Czado (2017). We are interested in the conditional quantiles q_α at some quantile level α of a continuous response Y given a continuous covariate vector $\mathbf{X} = (X_1, \dots, X_d)'$ taking on values $\mathbf{x} = (x_1, \dots, x_d)'$. The conditional quantile is defined as the inverse of the conditional distribution function of Y given $\mathbf{X}$, i.e.

$$q_\alpha(x_1, \dots, x_d) := F^{-1}_{Y|X_1,\dots,X_d}(\alpha|x_1, \dots, x_d). \tag{2.1}$$

Using Sklar's Theorem (Sklar, 1959) the right hand-side can be expressed in terms of the marginal distributions F_Y of Y and F_j of X_j and the copula between Y and $\mathbf{X}$ as

$$F^{-1}_{Y|X_1,\dots,X_d}(\alpha|x_1, \dots, x_d) = F^{-1}_Y\left(C^{-1}_{V|U_1,\dots,U_d}(\alpha|u_1, \dots, u_d)\right), \tag{2.2}$$

where $V = F_Y(Y)$, $U_j = F_j(X_j)$ are the uniformly distributed probability integral transforms of Y and $\mathbf{X}$, and $u_j = F_j(x_j)$ are their realizations. Further, $C_{V,U_1,\dots,U_d}$ is the distribution function of $(V, U_1, \dots, U_d)'$ and called the copula associated with the joint distribution of Y and $\mathbf{X}$ (for an introduction to copulas, see, Joe, 1997; Nelsen, 2007) and $C_{V|U_1,\dots,U_d}$ is the associated conditional distribution function of V given $(U_1, \dots, U_d)'$. This representation facilitates flexible modeling of q_α plugging in suitable estimators for the marginal distributions and the copula. Kraus and Czado (2017) propose to use kernel estimators for the marginals and a simplified D-vine copula for modeling $C_{V,\mathbf{U}}$. A simplified D-vine copula is a special case of regular vine copulas (see Aas et al., 2009; Bedford and Cooke, 2002). It constructs a d-dimensional copula density using the product of conditional and unconditional bivariate copulas:

$$c(u_1, \dots, u_d) = \prod_{i=1}^{d-1} \prod_{j=i+1}^{d} c_{ij;i+1,\dots,j-1}\big(C_{i|i+1,\dots,j-1}\left(u_i|u_{i+1}, \dots, u_{j-1}\right), C_{j|i+1,\dots,j-1}\left(u_j|u_{i+1}, \dots, u_{j-1}\right)\big). \tag{2.3}$$

Note that due to the simplifying assumption the pair-copulas $c_{ij;i+1,\dots,j-1}$ do not depend on the conditioning values $u_{i+1},\dots,u_{j-1}$ (see Hobæk Haff et al., 2010; Stöber et al., 2013; Killiches et al., 2017, for a more detailed discussion of the simplifying assumption). The arguments $C_{i|D}\left(u_i|\mathbf{u}_D\right)$ of the pair-copulas $c_{ij;D}$ can be derived recursively (Joe, 1997) using that for $l \in D$ and $D_{-l} := D \setminus \{l\}$ it holds that

$$C_{i|D}\left(u_i|\mathbf{u}_D\right) = h_{i|l;D_{-l}}\left(C_{i|D_{-l}}\left(u_i|\mathbf{x}_{D_{-l}}\right)|C_{l|D_{-l}}\left(u_l|\mathbf{u}_{D_{-l}}\right)\right), \tag{2.4}$$

where $h_{i|j;D}(u|v) = \partial C_{ij;D}(u,v)/\partial v$ is called the *h-function* associated with the pair-copula $C_{ij;D}$.

D-vine copulas inherit the great modeling flexibility attributed to vine copulas: every pair-copula can be modeled with a different copula family and parameter. The main reason why Kraus and Czado (2017) use a D-vine copula model in Equation (2.2) is that the inverse of the conditional distribution of the first variable V given the covariates $\mathbf{U}$ can be expressed analytically as a recursion over h-functions and their inverses. For example, in three dimensions the recursion in Equation (2.4) can be used to express the conditional distribution of V given $(U_1, U_2)'$ as

$$C_{V|U_1,U_2}(v|u_1,u_2) = h_{V|U_2;U_1}(h_{V|U_1}(v|u_1)|h_{U_2|U_1}(u_2|u_1)),$$

and therefore the conditional quantile as

$$C^{-1}_{V|U_1,U_2}(\alpha|u_1,u_2) = h^{-1}_{V|U_1}\left(h^{-1}_{V|U_2;U_1}\left(\alpha|h_{U_2|U_1}(u_2|u_1)\right)|u_1\right).$$

The order of the covariates in the D-vine can be chosen arbitrarily. Kraus and Czado (2017) proposed a fitting algorithm that sequentially adds the covariate to the model that improve the model fit the most. The model fit is measured in terms of the conditional log-likelihood for V given $\mathbf{u}$. More precisely, given copula data v_i and $\mathbf{u}_i$ and a fitted D-vine copula density $\hat{c}_{V,\mathbf{U}}$ the conditional log-likelihood (cll) is defined by $\sum_{i=1}^{n} \log \hat{c}_{V|\mathbf{U}}(v_i|\mathbf{u}_i)$. Here, $c_{V|\mathbf{U}}$ is the density associated with $C_{V|\mathbf{U}}$. Variables that do not improve the model fit are omitted, thus accomplishing an automatic forward covariate selection. Depending on the desired degree of parsimony, instead of the cll one can also use an AIC- or BIC-corrected version of the cll, penalizing the number of parameters in the model (cf. Kraus and Czado, 2017).

In a simulation study as well as real data applications, Kraus and Czado (2017) demonstrate the superiority of D-vine quantile regression over competitor methods in many settings. However, two important issues of D-vine quantile regression were left as open research problems: the need for nonparametric pair-copulas to avoid misspecifications as described in Dette et al. (2014), as well as the inability to handle data containing discrete variables. In the following section two new approaches are presented that generalize D-vine quantile regression to account for discrete data: one is parametric (Section 3.1) and the other is nonparametric (Section 3.2).

3 D-vine quantile regression for mixed discrete and continuous variables

Let now some or all of the variables, the response variable Y and the predictors X_j, be discrete. For $j \in \{1,\dots,d\}$, let $y \in \text{ran}(F_Y^{-1})$, $x_j \in \text{ran}(F_j^{-1})$ be observed values on the original scale and $v := F_Y(y)$, $u_j := F_j(x_j)$ the associated values on the original scale. Here, $\text{ran}(G^{-1})$ is the set of all possible values of a random variable with distribution G. As before, we express the conditional quantile of Y given $\mathbf{X} = \mathbf{x}$ in the following way:

$$q_\alpha(x_1,\dots,x_d) = F_Y^{-1}(C^{-1}_{V|U_1,\dots,U_d}(\alpha|u_1,\dots,u_d)).$$

To compute $C^{-1}_{V|U_1,\dots,U_d}(\alpha|u_1,\dots,u_d)$, we need to model the joint distribution of V and $U_1,\dots,U_d$. Similar to the continuous case, this joint distribution is modeled using a D-vine that has V fixed as the first node, which enables us to compute the conditional quantiles in an easy fashion.

3.1 Parametric modeling

Let $u, v, u_1, u_2, v_1, v_2 \in [0,1]$, $u_1 > u_2, v_1 > v_2$. As analogues of the h-functions $h_{i|j;D}(u,v) = \partial C_{i,j;D}(u,v)/\partial v$ for continuous conditioning variables, we define for discrete conditioning variables with $i, j \notin D$,

$$\tilde{h}_{i|j;D}(u|v_1,v_2) := \frac{C_{i,j;D}(u,v_1) - C_{i,j;D}(u,v_2)}{v_1 - v_2}, \tag{3.5a}$$

$$\tilde{h}_{j|i;D}(v|u_1,u_2) := \frac{C_{i,j;D}(u_1,v) - C_{i,j;D}(u_2,v)}{u_1 - u_2}. \tag{3.5b}$$

For a discrete random variable X, $x \in \mathrm{ran}(F_X^{-1})$ and $u = F_X(x)$, we denote by $u^- := F_X(x^-) := \lim_{a \nearrow x} F_X(a)$ the PIT-value of the next smaller value attained by X.
The following expressions for the conditional distribution function and the conditional density are derived in Stöber (2013). More detailed derivations including explicit expressions for all cases in 2 and 3 dimensions can be found in Chapter 3 of Schallhorn (2017).
If the joint distribution of $(V, \mathbf{U})$ is modeled by a D-vine with order $V - U_{l_1} - \ldots - U_{l_d}$, with $(l_1,\dots,l_d)'$ being a permutation of $(1,\dots,d)$, then the conditional distribution function $C_{V|\mathbf{U}}(v|\mathbf{u})$ can be computed iteratively by

$$C_{V|\mathbf{U}}(v|\mathbf{u}) = \begin{cases} h_{V|U_{l_d};\mathbf{U}_{-l_d}}\big(C_{V|\mathbf{U}_{-l_d}}(v|\mathbf{u}_{-l_d})\big|C_{U_{l_d}|\mathbf{U}_{-l_d}}(u_{l_d}|\mathbf{u}_{-l_d})\big), & X_{l_d} \text{ continuous}, \quad (3.6a) \\ \tilde{h}_{V|U_{l_d};\mathbf{U}_{-l_d}}\big(C_{V|\mathbf{U}_{-l_d}}(v|\mathbf{u}_{-l_d})\big|C_{U_{l_d}|\mathbf{U}_{-l_d}}(u_{l_d}|\mathbf{u}_{-l_d}), & \\ \qquad C_{U_{l_d}|\mathbf{U}_{-l_d}}(u_{l_d}^-|\mathbf{u}_{-l_d})\big), & X_{l_d} \text{ discrete}. \quad (3.6b) \end{cases}$$

For example, when X_1 is discrete we have

$$\begin{aligned} C_{V|U_1}(v|u_1) &= \frac{Pr(V \leqslant v, U_1 = u_1)}{Pr(U_1 = u_1)} = \frac{Pr(V \leqslant v, U_1 \leqslant u_1) - Pr(V \leqslant v, U_1 \leqslant u_1^-)}{Pr(U_1 \leqslant u_1) - Pr(U_1 \leqslant u_1^-)} \\ &= \frac{F_{V,U_1}(v,u_1) - F_{V,U_1}(v,u_1^-)}{u_1 - u_1^-} = \frac{C_{V,U_1}(F_V(v),F_{U_1}(u_1)) - C_{V,U_1}(F_V(v),F_{U_1}(u_1^-))}{u_1 - u_1^-} \\ &= \frac{C_{V,U_1}(v,u_1) - C_{V,U_1}(v,u_1^-)}{u_1 - u_1^-} = \tilde{h}_{V|U_1}(v|u_1,u_1^-). \end{aligned}$$

For the estimation of the D-vine, we need to estimate pair-copulas in a discrete-continuous setting. If $C_{1,2}$ is the pair-copula of U_1 and U_2, then their joint density is given by

$$f_{U_1,U_2}(u_1,u_2;C_{1,2}) = \begin{cases} c_{1,2}(u_1,u_2), & X_1, X_2 \text{ both continuous}, \quad (3.7a) \\ h_{2|1}(u_2|u_1) - h_{2|1}(u_2^-|u_1), & X_1 \text{ continuous}, X_2 \text{ discrete}, \quad (3.7b) \\ h_{1|2}(u_1|u_2) - h_{1|2}(u_1^-|u_2), & X_1 \text{ discrete}, X_2 \text{ continuous}, \quad (3.7c) \\ C_{1,2}(u_1,u_2) - C_{1,2}(u_1^-,u_2) - & \\ C_{1,2}(u_1,u_2^-) + C_{1,2}(u_1^-,u_2^-), & X_1, X_2 \text{ both discrete}. \quad (3.7d) \end{cases}$$

For given data $\mathbf{x}_j = \left(x_j^{(1)}, \dots, x_j^{(n)}\right)$, estimated marginal distribution functions $\hat{F}_j$ and $\hat{\mathbf{u}}_j := \hat{F}_j(\mathbf{x}_j)$ for $j = 1, 2$, the copula $C_{1,2}$ is estimated parametrically by minimizing the AIC

$$\mathrm{AIC}(C_{1,2}; \hat{\mathbf{u}}_1, \hat{\mathbf{u}}_2) := -2 \sum_{i=1}^{n} \log f_{U_1,U_2}\left(\hat{u}_1^{(i)}, \hat{u}_2^{(i)}; C_{1,2}\right) + 2|\boldsymbol{\theta}(C_{1,2})| \tag{3.8}$$

over all available pair-copula families and their parameters, where $|\boldsymbol{\theta}(C_{1,2})|$ is the number of parameters of the pair-copula $C_{1,2}$.
For independent observations $\mathbf{y} = \left(y^{(1)}, \dots, y^{(n)}\right)$, $\mathbf{x}_j = \left(x_j^{(1)}, \dots, x_j^{(n)}\right)$, $j = 1, \dots, d$ of the variables Y and $X_1, \dots, X_d$, we set $\hat{\mathbf{v}} := \hat{F}_Y(\mathbf{y})$ and $\hat{\mathbf{u}}_j := \hat{F}_j(\mathbf{x}_j)$ using the estimated marginal distribution functions. To fit the D-vine, the same estimation process of sequentially adding variables to the D-vine as for the continuous case is used. However, the conditional density $f_{V|\mathbf{U}}$ in the conditional log-likelihood (cll) is computed recursively as follows. If both Y and X_{l_d} are continuous, then it holds

$$\begin{aligned} f_{V|\mathbf{U}}\left(\hat{v}^{(i)}\middle|\hat{\mathbf{u}}^{(i)}\right) = \\ c_{V,U_{l_d};\mathbf{U}_{-l_d}}\left(F_{V|\mathbf{U}_{-l_d}}(\hat{v}^{(i)}|\hat{\mathbf{u}}_{-l_d}^{(i)}), F_{U_{l_d}|\mathbf{U}_{-l_d}}(\hat{u}_{l_d}^{(i)}|\hat{\mathbf{u}}_{-l_d}^{(i)})\right) \cdot f_{V|\mathbf{U}_{-l_d}}\left(\hat{v}^{(i)}\middle|\hat{\mathbf{u}}_{-l_d}^{(i)}\right). \end{aligned} \tag{3.9a}$$

If Y is continuous and X_{l_d} is discrete, then the following is fulfilled:

$$\begin{aligned} f_{V|\mathbf{U}}\left(\hat{v}^{(i)}\middle|\hat{\mathbf{u}}^{(i)}\right) =& \Big[h_{U_{l_d}|V;\mathbf{U}_{-l_d}}\left(F_{U_{l_d}|\mathbf{U}_{-l_d}}(\hat{u}_{l_d}^{(i)}|\hat{\mathbf{u}}_{-l_d}^{(i)})\middle|F_{V|\mathbf{U}_{-l_d}}(\hat{v}^{(i)}|\hat{\mathbf{u}}_{-l_d}^{(i)})\right) \\ & - h_{U_{l_d}|V;\mathbf{U}_{-l_d}}\left(F_{U_{l_d}|\mathbf{U}_{-l_d}}(\hat{u}_{l_d-}^{(i)}|\hat{\mathbf{u}}_{-l_d}^{(i)})\middle|F_{V|\mathbf{U}_{-l_d}}(\hat{v}^{(i)}|\hat{\mathbf{u}}_{-l_d}^{(i)})\right)\Big] \\ & \cdot \frac{f_{V|\mathbf{U}_{-l_d}}\left(\hat{v}^{(i)}\middle|\hat{\mathbf{u}}_{-l_d}^{(i)}\right)}{F_{U_{l_d}|\mathbf{U}_{-l_d}}(\hat{u}_{l_d}^{(i)}|\hat{\mathbf{u}}_{-l_d}^{(i)}) - F_{U_{l_d}|\mathbf{U}_{-l_d}}(\hat{u}_{l_d-}^{(i)}|\hat{\mathbf{u}}_{-l_d}^{(i)})}. \end{aligned} \tag{3.9b}$$

If Y is discrete and X_{l_d} is continuous, we have

$$\begin{aligned} f_{V|\mathbf{U}}\left(\hat{v}^{(i)}\middle|\hat{\mathbf{u}}^{(i)}\right) =& \, h_{V|U_{l_d};\mathbf{U}_{-l_d}}\left(F_{V|\mathbf{U}_{-l_d}}(\hat{v}^{(i)}|\hat{\mathbf{u}}_{-l_d}^{(i)})\middle|F_{U_{l_d}|\mathbf{U}_{-l_d}}(\hat{u}_{l_d}^{(i)}|\hat{\mathbf{u}}_{-l_d}^{(i)})\right) \\ & - h_{V|U_{l_d};\mathbf{U}_{-l_d}}\left(F_{V|\mathbf{U}_{-l_d}}(\hat{v}_{-}^{(i)}|\hat{\mathbf{u}}_{-l_d}^{(i)})\middle|F_{U_{l_d}|\mathbf{U}_{-l_d}}(\hat{u}_{l_d}^{(i)}|\hat{\mathbf{u}}_{-l_d}^{(i)})\right). \end{aligned} \tag{3.9c}$$

If both Y and X_{l_d} are discrete, it holds

$$\begin{aligned} f_{V|\mathbf{U}}\left(\hat{v}^{(i)}\middle|\hat{\mathbf{u}}^{(i)}\right) = \\ \tilde{h}_{V|U_{l_d};\mathbf{U}_{-l_d}}\left(F_{V|\mathbf{U}_{-l_d}}(\hat{v}^{(i)}|\hat{\mathbf{u}}_{-l_d}^{(i)})\middle|F_{U_{l_d}|\mathbf{U}_{-l_d}}(\hat{u}_{l_d}^{(i)}|\hat{\mathbf{u}}_{-l_d}^{(i)}), F_{U_{l_d}|\mathbf{U}_{-l_d}}(\hat{u}_{l_d-}^{(i)}|\hat{\mathbf{u}}_{-l_d}^{(i)})\right) \\ - \tilde{h}_{V|U_{l_d};\mathbf{U}_{-l_d}}\left(F_{V|\mathbf{U}_{-l_d}}(\hat{v}_{-}^{(i)}|\hat{\mathbf{u}}_{-l_d}^{(i)})\middle|F_{U_{l_d}|\mathbf{U}_{-l_d}}(\hat{u}_{l_d}^{(i)}|\hat{\mathbf{u}}_{-l_d}^{(i)}), F_{U_{l_d}|\mathbf{U}_{-l_d}}(\hat{u}_{l_d-}^{(i)}|\hat{\mathbf{u}}_{-l_d}^{(i)})\right). \end{aligned} \tag{3.9d}$$

Here, we set $\hat{v}_{-}^{(i)} := \left(\hat{v}^{(i)}\right)^{-}$ and $\hat{u}_{l_d-}^{(i)} := \left(\hat{u}_{l_d}^{(i)}\right)^{-}$.
Once the D-vine is specified, then $C^{-1}_{V|U_1,\dots,U_d}(\alpha|u_1, \dots, u_d)$ can be obtained by numerically inverting $C_{V|\mathbf{U}}(\cdot|\mathbf{u}) := C_{V|U_1,\dots,U_d}(\cdot|u_1, \dots, u_d)$, i.e.

$$C^{-1}_{V|\mathbf{U}}(\alpha|\mathbf{u}) = \underset{\substack{q\in[0,1]\\ C_{V|\mathbf{U}}(q|\mathbf{u})\geqslant\alpha}}{\operatorname{argmin}} \Big(C_{V|\mathbf{U}}(q|\mathbf{u}) - \alpha\Big), \tag{3.10}$$

opposed to the continuous case, where $C^{-1}_{V|\mathbf{U}}(\alpha|\mathbf{u})$ is expressed in terms of nested inverse h-functions. This modification is made since the $\tilde{\text{h}}$-functions defined in Equation (3.5) would need to be inverted numerically. Hence, it is more stable to directly numerically invert the

conditional distribution function composed of nested h- and $\widetilde{\text{h}}$-functions instead of computing the conditional quantile function using several numerically inverted $\widetilde{\text{h}}$-functions.
To ensure that non-influential variables are excluded and to reflect preference for parsimonious models, we use the AIC-corrected conditional log-likelihood as defined for the continuous case (Kraus and Czado, 2017), i.e. $\text{cll}_{\text{AIC}} = -2\,\text{cll} + 2|\boldsymbol{\theta}|$, where $|\boldsymbol{\theta}|$ is the number of parameters of the fitted D-vine. Since the pair-copulas in the estimation of the D-vine are determined parametrically, we call this method **parametric D-vine quantile regression (PDVQR)**.

3.2 Nonparametric modeling

Vine copula models can also be estimated nonparametrically. For the case where all variables are continuous, Nagler et al. (2017) surveyed existing methods for estimation of the vine copula density. It is straightforward to use these methods as nonparametric D-vine quantile regression estimators by following the construction of Kraus and Czado (2017): Given an estimate $\widehat{c}_{V,U_1,\dots,U_d}$ of the joint density of $(V, U_1, \dots, U_d)$, we can derive an estimate of the conditional distribution function $C_{V|U_1,\dots,U_d}$ as

$$\widehat{C}_{V|U_1,\dots,U_d}(v|u_1,\dots,u_d) = \int_0^v \widehat{c}_{V|U_1,\dots,U_d}(s|u_1,\dots,u_d)ds.$$

A nonparametric estimator of the conditional quantile function q_a is then defined by invoking (2.2). In the continuous case, $C^{-1}_{V|U_1,\dots,U_d}$ can even be derived in (almost) closed form, only involving h-functions and their inverses (see, Kraus and Czado, 2017).
This construction is straightforward as long as all variables are continuous, but none of the methods in Nagler et al. (2017) are applicable when some of the variables are discrete. Furthermore, the arguments in Section 3.1 do not apply since they are specific to maximum likelihood inference of a finite-dimensional parameter. Our solution to this problem is based on *continuous convolution*. The idea is to make all discrete variables continuous by adding a small amount of noise. Nagler (2017) showed that this still leads to valid estimators of conditional quantile functions if the noise distribution belongs to a certain class. We shall make this more precise in the following paragraphs.
For $\mathcal{D} \subseteq \{1,\dots,d\}$, suppose that Y and X_j, $j \in \{1,\dots,d\}\backslash\mathcal{D}$, are continuous variables, whereas $X_j, j \in \mathcal{D}$, are discrete. Let further $E_j, j \in \mathcal{D}$, be *iid* random variables independent of $(Y, X_1, \dots, X_d)$ with density η satisfying the following constraint: for some $0 < \gamma_1 \leqslant \gamma_2 < 1$, $\eta(x) = 1$ for $x \in [-\gamma_1, \gamma_1]$ and $\eta(x) = 0$ for $x \in \mathbb{R}\backslash(-\gamma_2, \gamma_2)$. An example of such a density is $\eta(x) = \mathbb{1}(|x| < 0.5)$, i.e., the E_j's are uniformly distributed on $(-0.5, 0.5)$.
The continuous convolution of $(X_1, \dots, X_d)$ is defined as $(\tilde{X}_1, \dots, \tilde{X}_d)$, where $\tilde{X}_j = X_j + E_j$ for all $j \in \mathcal{D}$, and $\tilde{X}_j = X_j$ for all $j \in \{1,\dots,d\}\backslash\mathcal{D}$. Then Proposition 5 of Nagler (2017) shows that for all $\alpha \in [0,1]$, $(x_1,\dots,x_d) \in \bigtimes_{j=1}^d \text{ran}(X_j)$,

$$F^{-1}_{Y|X_1,\dots,X_d}(\alpha|x_1,\dots,x_d) = F^{-1}_{Y|\tilde{X}_1,\dots,\tilde{X}_d}(\alpha|x_1,\dots,x_d). \tag{3.11}$$

The right hand side of (3.11) is the conditional quantile function of continuous variables only. It can thus be estimated by using any of the nonparametric methods in Nagler et al. (2017). And since (3.11) is an equality, this also yields an estimator of the left hand side, the conditional quantile function we are actually interested in. The case where Y is discrete can be handled similarly. However, a correction term has to be added to the right hand side of (3.11) (see, Nagler, 2017, Proposition 5). Nagler (2017) stressed that this approach is only valid for nonparametric estimation and, thus, must not be used with parametric models.
In the context of density estimation, Nagler and Czado (2016) showed that nonparametric estimators based on simplified vine copulas have an appealing property: they do not suffer from the curse of dimensionality. More specifically, convergence rates are equivalent to those

of a two-dimensional problem, no matter how large d actually is. Since the D-vine quantile regression estimator is derived from the estimated density, similar findings can be expected in our setting. The exact asymptotic behavior can be established by arguments similar to those in Nagler and Czado (2016), but is beyond the scope of this article.
In the simulations and application we will use the vine copula density estimator that performed best in Nagler et al. (2017). It estimates the pair-copula densities by a local likelihood approach proposed by Geenens et al. (2017). For the noise density η, we choose the uniform density, $\eta(x) = \mathbb{1}(|x| < 0.5)$. We call this method **nonparametric D-vine quantile regression (NPDVQR)**.

4 Simulation study

We will compare the two methods presented in this paper to three other commonly used methods for quantile regression. We start by a brief summary of the competitor methods, followed by a description of the simulation setup and results.

4.1 Competitor methods

Linear quantile regression (LQR) Introduced in Koenker and Bassett (1978), it is assumed that the conditional quantiles linearly depend on the conditioning values, i.e.

$$\hat{q}_\alpha(x_1, \dots, x_d) = \hat{\beta}_0 + \sum_{j=1}^{d} \hat{\beta}_j x_j.$$

The estimates for the regression coefficients $\hat{\beta}_j$ are obtained as the solution of the minimization problem

$$\min_{\beta \in \mathbb{R}^d} \left(\alpha \sum_{i=1}^{n} \left(y^{(i)} - \beta_0 - \sum_{j=1}^{d} \beta_j x_j^{(i)} \right)^+ + (1-\alpha) \sum_{i=1}^{n} \left(\beta_0 + \sum_{j=1}^{d} \beta_j x_j^{(i)} - y^{(i)} \right)^+ \right).$$

This method has various shortcomings described in Bernard and Czado (2015), for instance the estimates are not necessarily monotonically increasing in α. LQR can be performed using the function `rq` of the package `quantreg` (Koenker, 2015).

Boosted additive quantile regression (BAQR) To relax the linear assumption as above, Koenker (2005) proposes to use additive models for quantile regression, i.e.

$$\hat{q}_\alpha(x_1, \dots, x_d) = \hat{\beta}_0 + \sum_{j=1}^{J} \left(\sum_{k=1}^{K_j - 1} \hat{\beta}_j^k I_j^k(x_j) \right) + \sum_{j=J+1}^{d} g_j(x_j),$$

where the discrete variables $X_1, \dots, X_J$ are estimated by ordinary least squares using a dummy coding with K_j denoting the number of values attained by X_j, and g_j denotes a smooth function based on B-splines. Fenske et al. (2012) use a boosting technique to estimate the model parameters, minimizing a given loss function including penalizing terms by stepwise updating the estimator along the steepest gradient of the loss function. The algorithm is implemented in the function `gamboost` of the package `mboost` (Hothorn et al., 2016).

nonparametric quantile regression (NPQR) As introduced in Li et al. (2013), the conditional quantiles are obtained via numerical inversion of the conditional distribution function, i.e.

$$\hat{q}_\alpha(x_1, \dots, x_d) = \operatorname*{argmin}_{q \in \mathbb{R}} \left| \widehat{F}_{Y|X_1,\dots,X_d}(q|x_1, \dots, x_d) - \alpha \right|.$$

The estimate $\widehat{F}_{Y|X_1,\ldots,X_{d-1}}$ is obtained nonparametrically using a kernel estimator with an automatic data-driven bandwidth selector. NPQR can be performed using the function `npqreg` of the package `np` (Hayfield and Racine, 2008).
The three methods can handle continuous and discrete predictors. However, if Y is discrete then the estimated quantiles are not necessarily values actually attained by Y, so the obtained conditional quantiles have to be rounded to the closest value attained by Y.

4.2 Setup

We compare the five methods in different settings. For each setting and each replication $r = 1, \ldots, R = 100$, we simulate a training dataset $(y_{r,i}^{train}, \mathbf{x}_{r,i}^{train})_{i=1,\ldots,n_{train}}$ from the joint distribution of $(Y, \mathbf{X})$ and an evaluation dataset $(\mathbf{x}_{r,i}^{eval})_{i=1,\ldots,n_{eval}}$, $n_{eval} = 1000$, from the distribution of $\mathbf{X}$. For each method m and $\alpha \in (0,1)$, we compute the estimate of the conditional quantile function $\hat{q}_{m,\alpha}(\cdot)$ based on the training dataset. The evaluation dataset is used to estimate the root average squared error. We take the mean over all replications, giving us the out-of-sample mean root average squared error ($\text{MRASE}_{m,\alpha}$) of method m,

$$\text{MRASE}_{m,\alpha} := \frac{1}{R}\sum_{r=1}^{R}\sqrt{\frac{1}{n_{eval}}\sum_{i=1}^{n_{eval}}\left(\hat{q}_{m,\alpha}\left(\mathbf{x}_{r,i}^{eval}\right) - q_\alpha\left(\mathbf{x}_{r,i}^{eval}\right)\right)^2}, \tag{4.12}$$

where $q_\alpha(\cdot)$ denotes the true conditional quantile function.
The data is generated as follows. Using the package `copula` (Hofert et al., 2016), we simulate $(\mathbf{u}_1, \ldots, \mathbf{u}_d)$ from a d-dimensional Clayton copula with parameter $\theta = 1$, corresponding to an unconditional pairwise Kendall's τ of $1/3$, and sample size n_{train}, i.e. we have $\mathbf{u}_j \in [0,1]^{n_{train}}$. We consider $d \in \{3, 5\}$ and $n_{train} \in \{250, 1000\}$. The first two variables are discretized by applying the quantile function of the binomial distribution $F^{-1}(\cdot; N, 1/2)$ with parameters $N \in \{2, 8\}$ and $p = 1/2$. So if the j-th variable shall be discretized, we set

$$\mathbf{x}_j = F^{-1}(\mathbf{u}_j; N, 1/2).$$

The remaining continuous variables are transformed using the quantile function of the standard normal distribution Φ^{-1}, i.e. if the j-th variable shall be continuous, we set

$$\mathbf{x}_j = \Phi^{-1}(\mathbf{u}_j).$$

We then compute

$$\mathbf{y} = g(\mathbf{x}_1, \ldots, \mathbf{x}_d) + \boldsymbol{\varepsilon},$$

with some function $g : \mathbb{R}^d \to \mathbb{R}$ and $\boldsymbol{\varepsilon} \in \mathbb{R}^{n_{train}}$ that consists of $\varepsilon_i = \sqrt{\frac{\text{Var}(g(\mathbf{X});\theta,N)}{\text{SNR}}}\, Z_i$ with i.i.d. $Z_i \sim \mathcal{N}(0,1)$. Here, SNR denotes the signal-to-noise ratio, for which we consider $\text{SNR} \in \{0.5, 2\}$, and $\text{Var}(g(\mathbf{X}); \theta, N)$ is the variance of $g(\mathbf{X})$ depending on θ and N.

4.3 Results

Table 1, Table 2 and Table 3 present the results for the considered model specifications. Additional results for other specifications can be found in Schallhorn (2017). For each model specification and each α, the MRASEs marked in **bold** are the smallest MRASE or those which are not significantly larger than the smallest MRASE. The significance is measured by a t-test, for which we choose a significance level of 5%.
For **dimension 3 and g linear** as shown in Table 1, LQR has, as expected, the best prediction quality in almost every setting, while PDVQR clearly performs better for $\alpha = 0.01$ in the cases with both large errors (SNR $= 0.5$) and a small sample size ($n_{train} = 250$). For a larger sample size and smaller errors however, LQR also performs better in the tails.

SNR	n_{train}	N	α	PDVQR	NPDVQR	LQR	BAQR	NPQR
0.5	250	2	0.01	**1.27**	1.88	1.62	1.82	1.73
			0.5	0.71	1.36	**0.53**	0.71	1.05
		8	0.01	**1.44**	1.98	1.84	2.18	2.12
			0.5	0.82	1.02	**0.58**	0.78	1.40
	1000	2	0.01	1.08	1.01	**0.87**	1.48	1.29
			0.5	0.41	0.49	**0.28**	0.34	0.62
		8	0.01	1.12	1.20	**0.90**	1.86	1.62
			0.5	0.56	0.56	**0.31**	0.46	0.90
2	250	2	0.01	**0.82**	**0.84**	**0.81**	1.01	1.05
			0.5	0.43	0.48	**0.27**	0.37	0.65
		8	0.01	**0.95**	1.03	**0.92**	1.47	1.40
			0.5	0.55	0.58	**0.29**	0.41	0.93
	1000	2	0.01	0.79	0.62	**0.44**	0.81	0.77
			0.5	0.29	0.36	**0.14**	0.17	0.38
		8	0.01	0.76	0.69	**0.45**	1.36	0.99
			0.5	0.40	0.35	**0.15**	0.20	0.57

Table 1: MRASE for $d = 3$, linear $g(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3) = 2\boldsymbol{x}_1 - 3\boldsymbol{x}_3$ for all considered methods.

BAQR performs relatively bad in the tails, particularly when the errors are large, but it provides reasonable predictions for the conditional median. The performances of NPDVQR and NPQR are relatively bad, as to be expected in this linear case. However, NPDVQR and NPQR outperform BAQR for $\alpha = 0.01$ and NPDVQR is better than PDVQR in the setting $N = 8$ and $n = 1000$.

For **dimension 3 and** g **non-linear** as shown in Table 2, NPQR is the best method when the errors are small (SNR = 2). In the case of large errors however, it provides bad results for $\alpha = 0.01$, where the D-vine methods show the best prediction ability. Between PDVQR and NPDVQR, PDVQR performs better for the highly discrete cases with $N = 2$, while NPDVQR performs better for the more continuous cases with $N = 8$. The non-linear g-function implies a non-monotonic relationship between the response variable and the predictors X_2 and X_3.

SNR	n_{train}	N	α	PDVQR	NPDVQR	LQR	BAQR	NPQR
0.5	250	2	0.01	**3.22**	**3.12**	4.01	6.25	**3.26**
			0.5	1.98	**1.81**	2.04	**1.85**	**1.85**
		8	0.01	6.52	**3.96**	8.04	9.63	5.08
			0.5	3.97	**2.60**	5.24	5.62	3.00
	1000	2	0.01	**2.20**	2.43	2.66	5.92	2.43
			0.5	1.57	1.61	1.69	1.46	**1.23**
		8	0.01	5.10	**2.91**	6.93	9.51	3.72
			0.5	3.70	2.14	5.05	5.46	**2.03**
2	250	2	0.01	2.20	2.68	2.62	2.19	**1.99**
			0.5	1.55	1.65	1.71	1.38	**1.25**
		8	0.01	6.58	3.62	9.67	5.41	**3.28**
			0.5	3.77	2.31	5.08	5.36	**2.15**
	1000	2	0.01	**1.55**	2.37	2.02	1.92	**1.55**
			0.5	1.28	1.46	1.59	1.19	**0.84**
		8	0.01	6.14	2.79	9.48	5.34	**2.41**
			0.5	3.55	1.99	5.03	5.28	**1.43**

Table 2: MRASE for $d = 3$, non-linear $g(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3) = \boldsymbol{x}_1 - 2(\boldsymbol{x}_2 - 3)^2 + 4\sqrt{|\boldsymbol{x}_3|}$ for all considered methods.

As Dette et al. (2014) show, none of the popular parametric pair-copula families can model a non-monotonic dependency, disadvantaging PDVQR. Since the non-monotonicity is stronger for $N = 8$, PDVQR shows worse predictions in the tails in these cases. LQR performs worse than PDVQR, NPDVQR and NPQR for almost all model specifications. Again, BAQR performs considerably worse than the other methods (especially in the tails and for large errors) suggesting that it might be prone to over-fitting.

For **dimension 5 and g non-linear** as shown in Table 3, NPDVQR shows superior predictions for almost all specifications. The second best quantile prediction method for this non-linear example is NPQR. BAQR performs well for the conditional median in some cases, while it shows very large estimation errors for $\alpha = 0.01$ and SNR $= 0.5$. This might be due to the interaction term in the g-function, since we do not include interaction terms in the BAQR model. There are also non-monotonic dependencies in the data, explaining why PDVQR does not provide very accurate predictions. PDVQR still shows better predictions than BAQR for $\alpha = 0.01$ and a similar prediction quality for $\alpha = 0.5$. Again, LQR performs worse than PDVQR, NPDVQR and NPQR for all model specifications.

SNR	n_{train}	N	α	PDVQR	NPDVQR	LQR	BAQR	NPQR
0.5	250	2	0.01	6.05	**4.86**	8.04	11.11	5.52
			0.5	3.71	**3.18**	5.29	3.48	4.16
		8	0.01	**6.21**	**6.22**	7.86	11.00	**6.11**
			0.5	**4.19**	4.49	5.44	**4.19**	4.66
	1000	2	0.01	5.38	**3.74**	6.18	10.51	4.43
			0.5	2.84	**2.58**	4.88	2.99	**2.65**
		8	0.01	6.02	**4.54**	6.28	10.48	4.87
			0.5	3.46	**3.19**	5.05	3.85	**3.11**
2	250	2	0.01	5.21	**3.71**	6.14	5.95	4.32
			0.5	3.12	**2.70**	4.94	**2.69**	3.15
		8	0.01	5.60	**4.60**	6.39	6.49	4.91
			0.5	3.75	**3.33**	5.12	**3.27**	3.61
	1000	2	0.01	4.72	**3.15**	5.62	5.57	3.49
			0.5	2.36	2.37	4.85	2.52	**2.12**
		8	0.01	5.35	**3.61**	5.92	6.26	3.95
			0.5	3.14	**2.65**	5.01	3.15	**2.57**

Table 3: MRASE for $d = 5$, non-linear $g(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \boldsymbol{x}_4, \boldsymbol{x}_5) = 3\sqrt{\boldsymbol{x}_1} - \boldsymbol{x}_3^2 + (\boldsymbol{x}_4+1)^3 - \boldsymbol{x}_2 \cdot \boldsymbol{x}_5$ for all considered methods.

In conclusion, the D-vine quantile regression methods provide much better predictions than LQR in the cases with a non-linear g. Further, NPQR performed best among all methods in the non-linear scenario for $d = 3$, but NPDVQR shows better results when d is increased. This may be due to the fact that the convergence rate of NPDVQR is constant in d (cf. Nagler and Czado, 2016), while NPQR suffers from the curse of dimensionality. BAQR appears to work better when the signal-to-noise ratio is high. For the scenarios with lower signal-to-noise ratio errors, BAQR shows very large estimation errors (particularly in the tails) and is inferior to the D-vine quantile regressions methods. For a linear g, LQR outperforms all other methods as the assumption of linearity is fulfilled. Interestingly, the D-vine quantile regression delivers better results in the tails in settings with a small sample size and large errors. Thus, the D-vine quantile regression shows its merits particularly in the difficult cases, i.e., when the signal is hard to detect and extreme quantiles are the target.

We also want to briefly discuss the **run-time** for the model fitting and prediction of the conditional quantiles. Computation times over all 100 repetitions with $\alpha \in \{0.01, 0.5\}$, are shown in Table 4. PDVQR clearly has the longest run-time. BAQR shows the second and third longest run-time for $n_{train} = 250$ and $n_{train} = 1{,}000$, while NPQR shows the third

and second longest run-time for $n_{train} = 250$ and $n_{train} = 1{,}000$. NPDVQR is the fastest among the more sophisticated methods. Due to it's simplicity LQR can be computed almost instantly and is several orders of magnitude much faster than the other methods.
PDVQR is much slower than NPDVQR since the estimation of the pair-copulas takes more time. This is because parameters have to be estimated for several pair-copula families before the best fitting model can be selected; NPDVQR only estimates one nonparametric model. Another factor is that the likelihood of each pair-copula in PDVQR is more complex than in the continuous case (see Equation (3.7)), involving differences of the copula distribution function (which is demanding for some families).

n_{train}	PDVQR	NPDVQR	LQR	BAQR	NPQR
250	162.79	11.86	0.02	38.17	35.25
1,000	466.53	22.75	0.03	80.57	409.81

Table 4: Run-times in seconds of the different methods for $d = 3$, non-linear g, SNR $= 0.5$ and $N = 2$ (recorded on an 8-way Opteron with 16 cores, each with 2.0 GHz and 16 GB of memory).

5 Application

Thanks to the methods described in this paper, the application of D-vine quantile regression is no longer restricted to continuous data sets. We investigate the bike sharing data set from the UCI machine learning repository (Lichman, 2013), first analyzed in Fanaee-T and Gama (2013). It contains information on rental counts from the bicycle sharing system *Capital Bikeshare* offered in Washington, D.C., together with weather and seasonal information. As a response for the quantile regression we choose the daily count of bike rentals, observed in the years 2011-2012 (731 observations). They are displayed in the left panel of Figure 1.

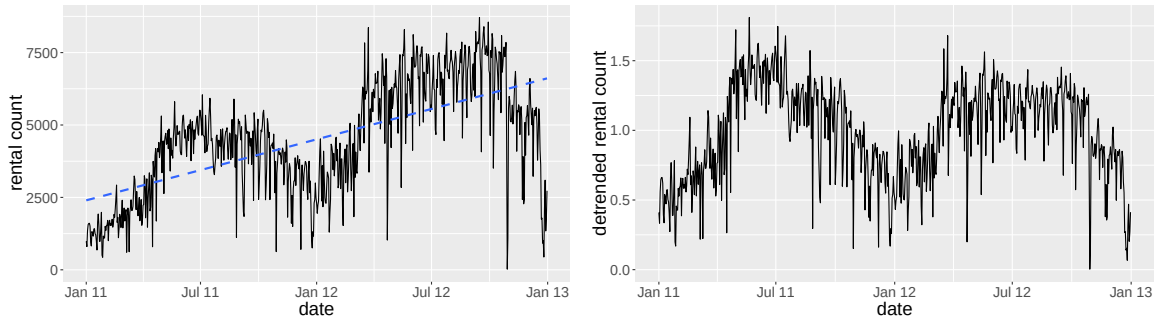

Figure 1: Observed (left) and detrended (right) bike rental counts in the years 2011-2012.

There is an obvious seasonal pattern and a linear trend reflecting a growth of the bike share system (visualized by the dashed line which is the least square linear line). While the seasonal pattern will be handled by the covariates, we cannot account for the linear trend. Therefore we remove the linear trend by dividing each observation by the least squares estimate of the linear trend. We use the division rather than the subtraction of the trend since the trend is a measure for the overall members of the bike sharing community and we are interested in the proportion of members renting bikes. The resulting detrended response is plotted in the right panel of Figure 1.
For each day we have continuous covariates *temperature* (apparent temperature in Celsius), *wind speed* (in mph) and *humidity* (relative in %). Additionally, there is the discrete variable *weather situation* giving information about the overall weather with values 1 (clear to partly cloudy), 2 (misty and cloudy) and 3 (rain, snow, thunderstorm). Further, we have information

about the *season* (spring, summer, fall and winter), *month* and *weekday* of the observed day and an indicator whether the day is a *working day*.
We applied all quantile regression methods discussed in this paper to the bike sharing data set for the quantile levels 0.1, 0.5 and 0.9 and use 10-fold cross-validation to evaluate their out-of-sample performance. Table 5 displays the corresponding averaged cross-validated tick-losses (see e.g. Komunjer, 2013), given by $\frac{1}{731}\sum_{i=1}^{731}\rho_\alpha(y^{(i)}-\hat{q}_\alpha^{(i)})$, where $\rho_\alpha(y)=y(\alpha-\mathbb{1}(y<0))$ denotes the check function, $y^{(i)}$ is the i-th observation of the response and $\hat{q}_\alpha^{(i)}$ is the α-quantile prediction. As before, the smallest losses and those which are not significantly larger than the smallest losses are printed in bold. Again, a Student's t test at 5% level was used to test whether larger values are significantly larger than the smallest value in a row.

α	PDVQR	NPDVQR	LQR	BAQR	NPQR
0.1	**0.039**	**0.035**	0.041	**0.035**	0.090
0.5	0.082	**0.069**	0.078	**0.064**	0.250
0.9	0.042	**0.032**	0.036	**0.032**	0.295

Table 5: Averaged in-sample tick-losses of the different quantile regression methods applied to the bike sharing data.

NPDVQR and BAQR produce the best results, significantly beating LQR and NPQR. Between the two new D-vine copula based quantile regression methods introduced in this paper, the nonparametric one significantly outperforms the parametric one for $\alpha=0.5$ and $\alpha=0.9$. The reason is that most of the covariates enter the models in a non-monotone fashion, as we will see. The ranking of the covariates by the nonparametric sequential selection algorithm is: temperature — humidity — wind speed — month — weather situation — weekday — working day — season.
In Figure 2 the influence of each of the covariates in the nonparametric D-vine quantile

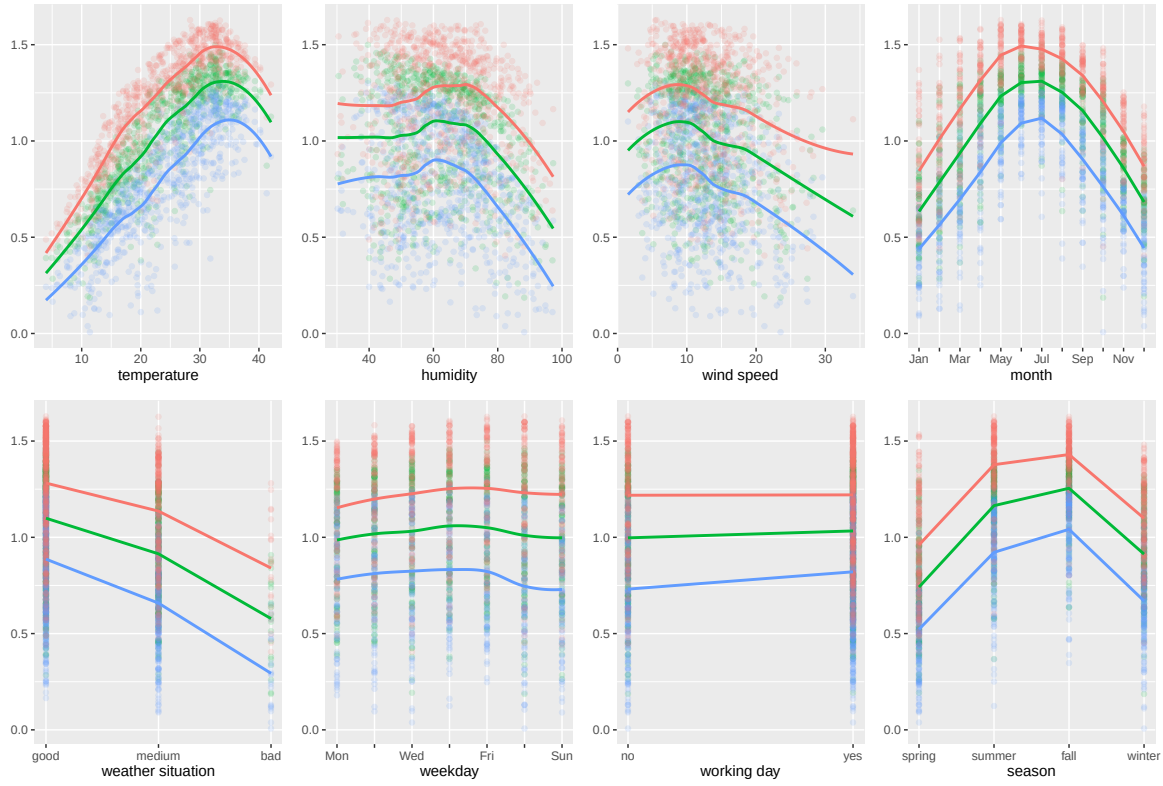

Figure 2: Influence of the different covariates on the response bike rentals using NPDVQR.

regression model is visualized. To be precise, for a covariate X_j we calculate for all quantile levels α of interest $\hat{q}_\alpha^{(i)} = \hat{F}^{-1}_{Y|\mathbf{X}}(\alpha|\mathbf{X} = \mathbf{x}^{(i)})$, $i = 1, \ldots, 731$, plot it against x_{ij} and add a smooth curve through the point cloud (fitted by `loess`). Figure 2 shows this for the quantile levels 0.1 (lower line), 0.5 (middle line) and 0.9 (upper line).
Higher temperatures generally go along with more bike rentals, until it gets too warm. For temperatures higher than 32 degrees Celsius, each additional degree causes a decline in bike rentals. Similar observations can be made for humidity. Bike rentals increase up to a relative humidity of around 60% and decrease afterwards. Wind speed also has a strong influence with fewer bike rentals on windy days. It is not surprising that the warm summer months encourage many citizens to rent bikes while in the cold winter rentals decrease on average by approximately 60%. The inclination to borrow bikes seems to grow during the week. On the weekend however, especially the 10% quantile drops considerably, which may be explained by many people leaving the city to visit their families or doing leisure activities on weekends. This is also supported by the influence of variable working day, with a few more rentals on working days. The variables weather situation and season support the thesis that more people tend to rent bicycles when the weather is good.
To investigate the differences between predictions of the various methods, we shall look more closely at the temperature variable. Figure 3 shows the effect of temperature on the predicted bike rentals using NPDVQR, PDVQR, LQR, BAQR and NPQR (from left to right).

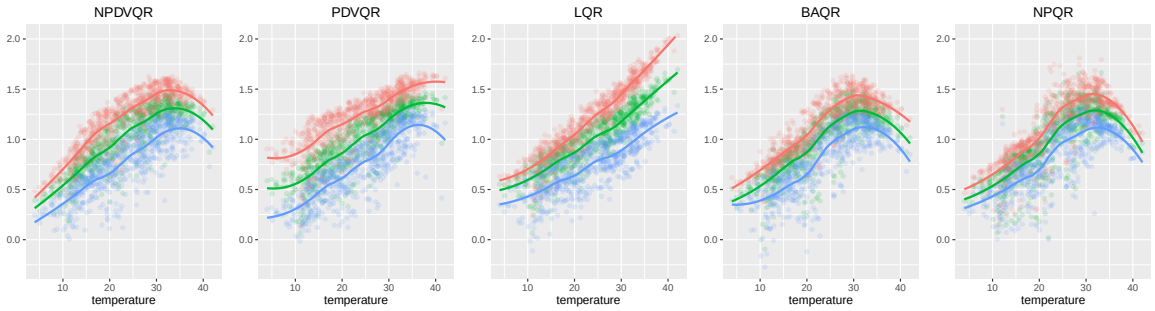

Figure 3: Influence of temperature on bike rentals for different quantile regression methods.

We see that the parametric D-vine as well as linear quantile regression are not really able to model the decline in rentals for very hot temperatures.
Apart from assessing the influence of covariates on the response, quantile regression can also be used to predict quantiles of the response in different scenarios. Suppose we know tomorrow is going to be a warm August Saturday with medium humidity and low wind-speed. Then, using our nonparametric D-vine copula based quantile regression model, we would predict a median of 8872 bikes to be rented with 10%- and 90% quantiles 7431 and 10485, respectively. In contrast, for a cold December Monday with heavy snow and high wind-speed the three predicted quantiles would be 22, 674 and 1152. As an operator of such a bike sharing system we could thus adapt our supply of rental bikes to the predicted demand.

6 Conclusion and outlook

Two new methods to predict conditional quantiles in a mixed discrete-continuous setting are proposed. They are based on a D-vine copula model that is estimated either parametrically or nonparametrically. The simulation study shows that the non-parametric D-vine quantile regression provides fast and accurate predictions for non-linear relationships between the quantile and the covariates. The parametric approach is often less accurate. This is due to the fact that non-linear relationships imply non-monotonic effects of some covariates on the response, which cannot be adequately modeled by most of the popular parametric pair-

copula families. This shortfall could be overcome by using parametric families that allow for non-monotonic dependence patterns. Developing such models will be a promising path for future research.

Acknowledgment

The third and fourth authors are supported by the German Research Foundation (DFG grants CZ 86/5-1 and CZ 86/4-1). Numerical calculations were performed on a Linux cluster supported by DFG grant INST 95/919-1 FUGG.

References

Aas, K., Czado, C., Frigessi, A., and Bakken, H. (2009). Pair-copula constructions of multiple dependence. *Insurance: Mathematics and Economics*, 44(2):182–198.

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

Bernard, C. and Czado, C. (2015). Conditional quantiles and tail dependence. *Journal of Multivariate Analysis*, 138(C):104–126.

Dette, H., Hecke, R. V., and Volgushev, S. (2014). Some comments on copula-based regression. *Journal of the American Statistical Association*, 109(507):1319–1324.

Fanaee-T, H. and Gama, J. (2013). Event labeling combining ensemble detectors and background knowledge. *Progress in Artificial Intelligence*, pages 1–15.

Fenske, N., Kneib, T., and Hothorn, T. (2012). Identifying risk factors for severe childhood malnutrition by boosting additive quantile regression. *Journal of the American Statistical Association*, 106(494):494–510.

Geenens, G., Charpentier, A., and Paindaveine, D. (2017). Probit transformation for nonparametric kernel estimation of the copula density. *Bernoulli*, 23(3):1848–1873.

Genest, C. and Nešlehová, J. (2007). A primer on copulas for count data. *ASTIN Bulletin*, 37(2):475–515.

Hayfield, T. and Racine, J. S. (2008). Nonparametric econometrics: The np package. *Journal of Statistical Software*, 27(5).

Hobæk Haff, I., Aas, K., and Frigessi, A. (2010). On the simplified pair-copula construction — simply useful or too simplistic? *Journal of Multivariate Analysis*, 101(5):1296–1310.

Hofert, M., Kojadinovic, I., Maechler, M., and Yan, J. (2016). *copula: Multivariate dependence with copulas*. R package version 0.999-15.

Hothorn, T., Buehlmann, P., Kneib, T., Schmid, M., and Hofner, B. (2016). *mboost: Model-based boosting*. R package version 2.7-0.

Joe, H. (1997). *Multivariate models and multivariate dependence concepts*. CRC Press.

Killiches, M., Kraus, D., and Czado, C. (2017). Examination and visualisation of the simplifying assumption for vine copulas in three dimensions. *Australian & New Zealand Journal of Statistics*, 59(1):95–117.

Koenker, R. (2005). *Quantile regression*. Cambridge University Press, Cambridge, Cambridgeshire, United Kingdom.

Koenker, R. (2011). Additive models for quantite regression: Model selection and confidence bandaids. *Brazilian Journal of Probability and Statistics*, 25(3):239–262.

Koenker, R. (2015). *quantreg: Quantile regression*. R package version 5.19.

Koenker, R. and Bassett, G. (1978). Regression quantiles. *Econometrica*, 46(1):33–50.

Komunjer, I. (2013). Quantile prediction. In *Handbook of Economic Forecasting*, pages 767–785. Elsevier.

Kraus, D. and Czado, C. (2017). D-vine copula based quantile regression. *Computational Statistics and Data Analysis*, 110C:1–18.

Li, Q., Lin, J., and Racine, J. S. (2013). Optimal bandwidth selection for nonparametric conditional distribution and quantile functions. *Journal of Business & Economic Statistics*, 31(1):57–65.

Lichman, M. (2013). UCI machine learning repository.

Nagler, T. (2017). A generic approach to nonparametric function estimation with mixed data. *arXiv preprint, arXiv:1704.07457*.

Nagler, T. and Czado, C. (2016). Evading the curse of dimensionality in nonparametric density estimation with simplified vine copulas. *Journal of Multivariate Analysis*, 151:69–89.

Nagler, T., Schellhase, C., and Czado, C. (2017). Nonparametric estimation of simplified vine copula models: comparison of methods. *arXiv preprint, arXiv:1701.00845*.

Nelsen, R. B. (2007). *An introduction to copulas*. Springer Science & Business Media.

Onken, A. and Panzeri, S. (2016). Mixed vine copulas as joint models of spike counts and local field potentials. 30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain. URL: https://papers.nips.cc/paper/6069-mixed-vine-copulas-as-joint-models-of-spike-counts-and-local-field-potentials.pdf.

Panagiotelis, A., Czado, C., and Joe, H. (2012). Pair copula constructions for multivariate discrete data. *Journal of the American Statistical Association*, 107(499):1063–1072.

Schallhorn, N. (2017). D-vine quantile regression for mixed discrete and continuous data with applications to bank stress testing. Master's thesis, Technische Universität München, Germany.

Sklar, A. (1959). Fonctions dé repartition á n dimensions et leurs marges. *Publications de l'Instutut de Statistique de l'Université de Paris*, 8:229–231.

Stöber, J. (2013). *Regular vine copulas with the simplifying assumption, time-variation, and mixed discrete and continuous margins*. Dissertation, Technische Universität München, München.

Stöber, J., Joe, H., and Czado, C. (2013). Simplified pair copula constructions—limitations and extensions. *Journal of Multivariate Analysis*, 119:101–118.