Copula density estimation by total variation penalized likelihood with linear equality constraints

Leming Qu *
Department of Mathematics, Boise State University,
lqu@boisestate.edu
and
Wotao Yin
Department of of Computational and Applied Mathematics, Rice University,
wotao.yin@rice.edu

October 9, 2009

Abstract

A copula density is the joint probability density function (PDF) of a random vector with uniform marginals. An approach to bivariate copula density estimation is introduced that is based on a maximum penalized likelihood estimation (MPLE) with a total variation (TV) penalty term. The marginal unity and symmetry constraints for copula density are enforced by linear equality constraints. The TV-MPLE subject to linear equality constraints is solved by an augmented Lagrangian and operator-splitting algorithm. It offers an order of magnitude improvement in computational efficiency over another TV-MPLE method without constraints solved by log-barrier method for second order cone program. A data-driven selection of the regularization parameter is through K-fold cross-validation (CV). Simulation and real data application show the effectiveness of the proposed approach. The MATLAB code implementing the methodology is available online.

Key Words: Copula density estimation; Total variation; Maximum penalized likelihood estimation; Augmented Lagrangian method

1 Introduction

A bivariate copula density $c(u, v)$, $[u, v] \in [0, 1]^2$ can be regarded as the joint probability density function (PDF) of a bivariate standard uniform random variable $[U, V]$. Most copulas are

*Corresponding author. Leming Qu, Department of Mathematics, Boise State University, 1910 University Dr. Boise ID, 83725-1555 USA. Tel: 1-208-426-2803; Fax: 1-208-426-1356.

exchangeable, thus implying $c(u, v)$ is symmetric. The $c(u, v)$ must satisfy the following four properties:

(P1) $c(u, v) \geq 0$, for $[u, v] \in [0, 1]^2$;

(P2) $\int_0^1 c(u, v)du = 1$, for $0 \leq v \leq 1$;

(P3) $\int_0^1 c(u, v)dv = 1$, for $0 \leq u \leq 1$;

(P4) $c(u, v) = c(v, u)$.

Note that (P2) and (P4) implies (P3), so (P3) is redundant.

A bivariate copula $C(u, v)$ defined on the unit square $[0, 1]^2$ is a bivariate cumulative distribution function (CDF) with univariate standard uniform margins:

$$C(u, v) = \int_0^u \int_0^v c(s, t)\mathrm{d}s\mathrm{d}t.$$

Sklar's Theorem (Sklar 1959) states that the joint CDF $F(x, y)$ of a bivariate random variable (X, Y) with marginal CDF $F_X(x)$ and $F_Y(y)$ can be written as $F(x, y) = C(F_X(x), F_Y(y))$, where copula C is the joint CDF of $(U, V) = (F_X(X), F_Y(Y))$. This indicates a copula connects the marginal distributions to the joint distribution and justifies the use of copulas for building bivariate distributions.

Let $(X_1, Y_1), \ldots, (X_n, Y_n)$ be a random sample from the unknown distribution F of (X, Y). We wish to estimate aspects of the joint distribution of X and Y, in particular, the copula density function $c(u, v)$.

The copula density estimation has been mostly studied in a parametric framework, whereby $c(u, v)$ is assumed to be a member of a copula family determined by a few parameters (for example, Shih and Louis(1995)). The parametric copula density estimation problem is then essentially reduced to estimate the few parameters that determine the copula. We propose here to estimate the bivariate copula density non-parametrically. For practitioners, nonparametric estimates could be used as the first step toward selecting the right parametric family.

Nonparametric estimation of copula and its density does not assume a specific parametric form for the copula and the marginals and thus provides great flexibility and generality. Nonparametric estimators of a bivariate copula density using kernels have been suggested by Gijbels and Mielniczuk (1990) and by Fermanian and Scaillet (2003). Kernel estimators have a severe drawback

as they require a very large amount of data (page 195, Malevergne and Sornette, 2006). Sancetta and Satchell (2004) employed techniques based on Bernstein polynomials. Hall and Neumeyer (2006) used a wavelet estimator to approximate the copula density. Autin et.al. (2009) dealt with the copula density estimation using wavelet methods by adaptive shrinkage procedures based on thresholding rules.

What does a copula density $c(u, v)$ look like? In one extreme, when U and V are independent of each other, $c(u, v) = 1$. When U and V are dependent, $c(u, v)$ can be smooth, have sharp boundaries, or even be unbounded. It is reasonable to assume that the total variation (TV) of $c(u, v)$, or at least its discrete version, is bounded. In practice, we often estimate and display the density in a finite grid. We propose a maximum penalized likelihood estimation (MPLE) with TV penalty method. This method is capable of capturing sharp changes in the target copula density, suffering less from edge effects when the copula density can be unbounded at boundaries in some statistically important cases.

The TV penalty based MPLE for copula density was proposed in Qu et al.(2009), where the penalty term is the TV of the log density, and the unity requirement for a density function is imposed. However, the properties (P2) (P3) and (P4) are not enforced. In fact, we are not aware of any method that explicitly imposes the properties (P1-P4). The main reason behind this is probably related to the difficulty of the induced estimation or optimization procedure. In this paper, we enforce the properties (P2), (P3), and (P4) as linear equality constraints for the discretized copula density. We solve the problem of minimizing penalized negative log likelihood with TV penalty subject to linear equality constraints by an augmented Lagrangian and operator-splitting algorithm. The effectiveness of our method is illustrated through numerical experiments.

Density estimation by TV penalized likelihood has been proposed by several groups of researchers. Koenker and Mizera (2007) used the TV of the derivative of the log density as the penalty in the univariate case and TV of the log density defined in a triogram in the bivariate case. Sardy and Tseng (2009), Mohler et al.(2009) used the TV of the density itself as the penalty. Mohler et al.(2009) presented a fast and accurate numerical method, based upon the Split Bregman L1 minimization technique (Goldstein and Osher 2009).

2 Problem Formulation

When the two marginal distributions are continuous, the copula density $c(u, v)$ is the unique bivariate density of $(U, V) = (F_X(X), F_Y(Y))$ as implied by Sklar's theorem. As copulas are not directly observable, a nonparametric copula density estimator has to be formed in two stages: obtaining the observations for (U, V) first and then estimating the copula density based on these observations.

In the first stage, the original data set (X_i, Y_i) for $i = 1, \ldots, n$ is converted to $(\hat{U}_i, \hat{V}_i) = (\hat{F}_X(X_i), \hat{F}_Y(Y_i))$, where $\hat{F}_X$ and $\hat{F}_Y$ are conventional estimators of F_X and F_Y. If models are available for the marginal distributions of X and Y but not for the joint distribution, one can use a technique such as maximum likelihood to estimate the marginal distribution functions. Otherwise, some nonparametric univariate distribution estimation methods or simply the following empirical CDFs (ECDFs) can be used:

$$\hat{F}_X(x) = \frac{1}{n}\sum_{i=1}^{n} I(X_i \le x), \quad \hat{F}_Y(y) = \frac{1}{n}\sum_{i=1}^{n} I(Y_i \le y), \tag{1}$$

where $I(\cdot)$ is the indicator function. When ECDFs are used as the marginal CDF estimators (e.g., in Autin et al.(2009)), $\{(\hat{U}_i, \hat{V}_i)\}_{i=1}^n$ is nothing but the standardized ranks.

In the second stage, we estimate the copula density $c(u, v)$ based on the observations $\{(\hat{U}_i, \hat{V}_i)\}_{i=1}^n$. Specifically, we do not assume any parametric form for $c(u, v)$ and instead, obtain an estimate of it that satisfies properties (P1-P4) and is defined on a unit rectangle grid, defined by equally dividing domain of $c(u, v)$, $[0, 1]^2$, into $N = m^2$ rectangle cells with cell size $(1/m) \times (1/m)$. A reasonable grid size for sample size $n = 1000$ is 64×64 (i.e., $m = 64$). A much finer discretization will slow down computation unnecessarily.

Let us use $i, j = 1, \ldots, m$ to index all the N cells of this grid. On each cell (i, j), $i, j = 1, \ldots, m$, let x_{ij} denote the constant estimate of $c(u, v)$ over the cell and set p_{ij} to the number of observations $\{(\hat{U}_i, \hat{V}_i)\}_{i=1}^n$ falling in this cell.

The marginal integral of $c(u, v)$ can be approximated by the Riemann sum

$$\int_0^1 c(u, v)du \approx \frac{1}{m}\sum_{i}^{m} x_{ij} = 1, \quad j = 1, \ldots, m$$

and

$$\int_0^1 c(u, v)dv \approx \frac{1}{m}\sum_{j}^{m} x_{ij} = 1, \quad i = 1, \ldots, m.$$

TV of $\boldsymbol{x}$ is defined as

$$\mathrm{TV}(\boldsymbol{x}) = \sum_{i,j=1,\ldots,m} \sqrt{(x_{i+1,j} - x_{i,j})^2 + (x_{i,j+1} - x_{i,j})^2} \approx \int\int \left\| D\left(c(u,v)\right) \right\|_2 ,$$

where we set the Nuemann boundary conditions for TV, namely, $x_{m+1,j} \equiv x_{m,j}$, $j = 1, \ldots m$, and $x_{i,m+1} \equiv x_{i,m}$, $i = 1, \ldots, m$.

In Qu et. al.(2009), by defining $z_{ij} = \log x_{ij}$, the MPLE-TV is to solve :

$$\min_{\boldsymbol{z}} \; T_\lambda(\mathbf{z}) = -\sum_{i=1}^{m}\sum_{j=1}^{m} p_{ij} z_{ij} + \lambda \mathrm{TV}(\mathbf{z}), \quad \text{s.t.} \quad \frac{1}{N}\sum_{i=1}^{m}\sum_{j=1}^{m} \exp(z_{ij}) = 1,$$

where λ is a smoothing parameter controlling the smoothness of the estimate. The above constrained minimization problem is equivalent to the following unconstrained minimization problem:

$$\min_{\boldsymbol{z}} \; T_\lambda(\mathbf{z}) = \sum_{i,j} [-p_{ij} z_{ij} + \frac{n}{N} \exp(z_{ij})] + \lambda \mathrm{TV}(\mathbf{z}).$$

Even though this unconstrained minimization formulation is attractive, it does not impose the properties (P2) and (P4).

In terms of $\mathbf{z} = \log \boldsymbol{x}$, the property (P2) requires the nonlinear constraints

$$\sum_{i=1}^{m} \exp(z_{ij}) = m, \quad j = 1, \ldots, m.$$

Nonlinear constraints are more difficult to work with than linear constraints, so it is preferable to minimize with respect to $\boldsymbol{x}$ instead of $\mathbf{z}$ if properties (P2) and (P4) are to be imposed.

Imposing the marginal unity (P2) and symmetry (P4) properties, we estimate a copula density as a $m \times m$ digital image by solving:

$$\begin{aligned} \min_{\boldsymbol{x}} \; T_\lambda(\boldsymbol{x}) &= -\sum_{i,j} p_{ij} \log x_{ij} + \lambda \mathrm{TV}(\boldsymbol{x}), \\ \text{s.t.} \quad \sum_{i=1}^{m} x_{ij} &= m, \quad j = 1, \ldots, m, \text{ and} \\ x_{ij} &= x_{ji}, \quad i, \; j = 1, \ldots, m. \end{aligned}$$

where λ is a smoothing parameter controlling the smoothness of the estimate.

The linear equality constraints in the above minimization problem can be written in the form $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ by forming the $m(m+1)/2 \times N$ matrix $\boldsymbol{A}$ and $m(m+1)/2$-vector $\boldsymbol{b}$ as follows:

$$\begin{aligned} A(i,j) &= 1, \quad i = 1, \ldots, m, \quad j = (i-1)m + 1, \ldots, im; \\ b(i) &= m, \quad i = 1, \ldots, m; \end{aligned}$$

and

$$\begin{aligned} A(m+(i-1)(i-2)/2+j,\ (j-1)m+i) &= 1, \quad i=2,\ldots,m, \quad j=1,\ldots,i-1; \\ A(m+(i-1)(i-2)/2+j,\ (i-1)m+j) &= -1, \quad i=2,\ldots,m, \quad j=1,\ldots,i-1; \\ b(i) &= 0, \quad i=m+1,\ldots,m(m+1)/2; \end{aligned}$$

and

$$A(i,j)=0, \text{ otherwise.}$$

The matrix A is very sparse.

Let $f(\boldsymbol{x}) = -\sum_{i,j} p_{ij} \log x_{ij}$, then $\nabla_{\boldsymbol{x}} f(\boldsymbol{x}) = -\boldsymbol{p}./\boldsymbol{x}$, where $\nabla_{\boldsymbol{x}}$ denotes the gradient operator with respect to $\boldsymbol{x}$ and ./ denotes element-wise division. This gradient will be used in the optimization algorithm discussed in the next section.

Our proposed copula density estimate solves:

$$\min_{\boldsymbol{x}} \ T_\lambda(\boldsymbol{x}) = f(\boldsymbol{x}) + \lambda \mathrm{TV}(\boldsymbol{x}), \text{ s.t. } \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}. \tag{2}$$

3 Augmented Lagrangian and operator–splitting algorithm

This section describes how to efficiently solve problem (2) by the augmented Lagrangian and operator–splitting techniques with modifications. The so-call augmented Lagrangian of (2) is

$$L(\boldsymbol{x},\boldsymbol{y}) = f(\boldsymbol{x}) + \lambda \mathrm{TV}(\boldsymbol{x}) + \frac{\alpha}{2}||\boldsymbol{A}\boldsymbol{x}-\boldsymbol{b}||^2 - <\boldsymbol{y}, \boldsymbol{A}\boldsymbol{x}-\boldsymbol{b}>,$$

where $\boldsymbol{y}$ contains the Lagrange multipliers. The traditional augmented Lagrangian algorithm is the iteration of

(1) $\boldsymbol{x} \leftarrow \arg\min_{\boldsymbol{x}} L(\boldsymbol{x},\boldsymbol{y})$;

(2) $\boldsymbol{y} \leftarrow \boldsymbol{y} + \alpha(\boldsymbol{b}-\boldsymbol{A}\boldsymbol{x})$,

which avoids solving the original constrained problem or requiring the penalty parameter α to increasing to infinity. However, because of the nonsmooth TV term and matrix $\boldsymbol{A}$, it is a time consuming task to complete step (1) above, i.e., minimizing $L(\boldsymbol{x},\boldsymbol{y})$ with respect to $\boldsymbol{x}$. A good way to get around the computational complexity of step (1) above is to consider the linearization,

which is related to the classical work of augmented Lagrangian and alternating direction methods (Glowinski and Tallec, 1989).

Introduce

$$g(\boldsymbol{x}) = \nabla_{\boldsymbol{x}}(f(\boldsymbol{x}) + \frac{\alpha}{2}||\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}||^2).$$

For the problem formulated in the last section, $g(\boldsymbol{x}) = -\boldsymbol{p}./\boldsymbol{x} + \alpha\boldsymbol{A}^T(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b})$. The one-step operator-splitting iteration is

$$\boldsymbol{x} \leftarrow R(\boldsymbol{x} - \beta(g(\boldsymbol{x}) - \boldsymbol{A}^T\boldsymbol{y})), \tag{3}$$

where $R(\boldsymbol{z}) = \arg\min_{\boldsymbol{x}} \frac{1}{2}||\boldsymbol{z}-\boldsymbol{x}||^2+\lambda\mathrm{TV}(\boldsymbol{x})$ ("arg min" is well-defined because the strong convexity of $\|\boldsymbol{z} - \boldsymbol{x}\|^2$ gives solution existence and uniqueness). In iteration, this step can be equivalently written as

Step 1: $\boldsymbol{x}^{k+1} \leftarrow \arg\min_{\boldsymbol{x}} \frac{1}{2}||(\boldsymbol{x}^k - \beta(g(\boldsymbol{x}^k) - \boldsymbol{A}^T\boldsymbol{y}^k)) - \boldsymbol{x}||^2 + \lambda\mathrm{TV}(\boldsymbol{x})$.

The update to the multipliers $\boldsymbol{y}$ is

Step 2: $\boldsymbol{y}^{k+1} \leftarrow \boldsymbol{y}^k + \gamma\alpha(\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^{k+1})$.

The algorithm starts with $\boldsymbol{y}^0 = \boldsymbol{0}$ and an initial $\boldsymbol{x}^0$, then iterates through steps 1 and 2 until certain convergence criteria is met.

We note that Step 1 above is different from the so-called alternating direction method (ADM, Glowinski and Tallec, 1989) or the recent algorithm TVAL3 (Li et.al., 2009). To minimize a function in the form of $a(B\boldsymbol{x}) + b(\boldsymbol{x})$ where B is a certain operator, ADM introduces an unknown vector $\boldsymbol{z}$ together with constraints $B\boldsymbol{x} = \boldsymbol{z}$ and uses an augmented Lagrangian $L(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{y})$ to relax these constraints. However, ADM has a different Step 1. In Step 1, ADM computes $x \leftarrow \arg\min_{\boldsymbol{x}} L(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{y})$ and uses the updated $\boldsymbol{x}$ to obtain $\boldsymbol{z} \leftarrow \arg\min_{\boldsymbol{z}} L(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{y})$. TVAL3 is similar to ADM as it splits $\mathrm{TV}(\boldsymbol{x}) = \sum_{ij} \|(D\boldsymbol{x})_{ij}\|_2$ into $\sum_{ij} \|\boldsymbol{z}_{ij}\|_2$ and constraints $\boldsymbol{z} = D\boldsymbol{x}$, where $\boldsymbol{z}$ is an unknown vector. In Step 1, however, TVAL3 does not exactly minimize with respect to $\boldsymbol{x}$ but updates $\boldsymbol{x}$ by one or more gradient descents. Different from ADM and TVAL3, our approach does not split $\mathrm{TV}(\boldsymbol{x})$ or exactly minimizes any term involving $f(\boldsymbol{x})$ in Step 1.

It is sometimes tricky to set appropriate penalty parameter α and step length β. One usually has a bound beforehand and tries different values in practice. An excessive large α overweighs the penalty term $(1/2)\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2$, causing a slowly convergent or even non-convergent algorithm. We found that $\alpha = 0.05$ worked well and fixed it throughout our simulations. According to Glowinski

and Tallec (1989), γ must be strictly less than $(\sqrt{5}+1)/2$ for ADM to converge, but one can simply try with different values. We set γ as 1 in our simulations.

The step length β should be smaller than $2/||\boldsymbol{J}(g(\boldsymbol{x}))||$, where $\boldsymbol{J}(g(\boldsymbol{x}))$ denotes the Jacobian of $g(\boldsymbol{x})$, to essentially guarantee that update (3) and thus Step 1 above are non-expansive. Loosely speaking, $||\boldsymbol{J}(g(\boldsymbol{x}))||$ is basically the max curvature of the graph of $f(\boldsymbol{x}) + \alpha/2||\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}||^2$ around the current $\boldsymbol{x}$. The larger the curvature, the smaller a step should be because the gradient information is accurate in a smaller perimeter. The log function has unlimited curvature but locally, it is bounded. In our numerical study, we fix β as 0.1 and it works well for all our experiments. Line search techniques can be applied to automate β choice.

The step 1 above solves a ROF/TV-L2 denoising problem, for which a few efficient algorithms exist in the literature. They include the latest graph-cut/max-flow algorithms (Chambolle 2005, Darbon and Sigelle 2006, Goldfarb and Yin 2007). We use the parametric maximum flow code for fast Total Variation minimization available from Yin (2007). The efficient ROF/TV-L2 denoising algorithm is at the core of the above augmented Lagrangian method. It is essential to make the computational intensive data adaptive choice of λ feasible.

4 Data-Driven Selection of λ

It is well known that the choice of the regularization parameter λ is one of the most important steps of the MPLE. It has to be tuned for practical applications. In this work we use the popular K-fold cross-validation (CV) method for density estimation to select the tuning parameter. Liu et al.(2009) gave a brief review of this method in its section 5.1. For completeness, we give details of K-fold CV below in the context of copula density estimation.

We randomly divide all the samples $\{(U_i, V_i)\}_{i=1}^n$ into K disjoint subsets (folds) of approximately the same size. Let S_k be the index set of the kth subset, $k = 1, ..., K$, $\hat{c}_\lambda(u, v)$ be the copula density estimate based on the entire data set, and $\hat{c}_{\lambda,-k}(u, v)$ be the the copula density estimate based on all data points except those in the kth subset.

The quality of a copula density estimator $\hat{c}_\lambda(u, v)$ is measured by $E(\text{Loss}(\hat{c}_\lambda, c))$ where $\text{Loss}(\hat{c}_\lambda, c)$ is a Loss function or distance measure between $\hat{c}_\lambda(u, v)$ and the true copula density estimator $c(u, v)$. Two commonly used distance measure between two densities are integrated squared error

$$\text{ISE}(\hat{c}_\lambda, c) = \int_0^1 \int_0^1 \left(\hat{c}_\lambda(u, v) - c(u, v)\right)^2 dudv,$$

and Kullback-Leibler distance

$$\mathrm{KLD}(\hat{c}_\lambda, c) = \int_0^1 \int_0^1 \log\left(\frac{c(u,v)}{\hat{c}_\lambda(u,v)}\right) c(u,v)dudv. \tag{4}$$

Given a data set $\{(U_i, V_i)\}_{i=1}^n$ generated from $c(u,v)$, we aim to find the λ which minimizes $\mathrm{Loss}(\hat{c}_\lambda, c)$.

Least squares CV represents a data-driven attempt at constructing $\hat{c}_\lambda(u,v)$ so as to minimize $\mathrm{ISE}(\hat{c}_\lambda, c)$. By expanding ISE, we have

$$\mathrm{ISE}(\hat{c}_\lambda, c) = \int_0^1 \int_0^1 \hat{c}_\lambda(u,v)^2 dudv - 2\int_0^1 \int_0^1 \hat{c}_\lambda(u,v)c(u,v)dudv + \int_0^1 \int_0^1 c(u,v)^2 dudv.$$

The term $\int_0^1 \int_0^1 c(u,v)^2 dudv$ does not depend on λ, so it can be dropped for the purpose of searching for λ. The term

$$\int_0^1 \int_0^1 \hat{c}_\lambda(u,v)c(u,v)dudv = E(\hat{c}_\lambda(U,V))$$

may be estimated approximately by

$$\frac{1}{K}\sum_{k=1}^K \frac{1}{|S_k|}\sum_{i\in S_k} \hat{c}_{\lambda,-k}(U_i, V_i),$$

where $|S_k|$ is the cardinality of S_k. Hence, the least squares CV score $\mathrm{LS}(\lambda)$ is defined as

$$\mathrm{LS}(\lambda) = \int_0^1 \int_0^1 \hat{c}_\lambda(u,v)^2 dudv - \frac{2}{K}\sum_{k=1}^K \frac{1}{|S_k|}\sum_{i\in S_k} \hat{c}_{\lambda,-k}(U_i, V_i).$$

Likelihood CV represents a data-driven attempt at constructing $\hat{c}_\lambda(u,v)$ so as to minimize $\mathrm{KLD}(\hat{c}_\lambda, c)$ (Hall 1987). By expanding KLD, we have

$$\mathrm{KLD}(\hat{c}_\lambda, c) = \int_0^1 \int_0^1 (\log c(u,v))\, c(u,v)dudv - \int_0^1 \int_0^1 (\log \hat{c}_\lambda(u,v))\, c(u,v)dudv.$$

The first term on the right hand side above can be dropped for the purpose of searching for λ. The term

$$\int_0^1 \int_0^1 (\log \hat{c}_\lambda(u,v))\, c(u,v)dudv = E\left(\log \hat{c}_\lambda(U,V)\right)$$

may be approximated by

$$\frac{1}{K}\sum_{k=1}^K \frac{1}{|S_k|}\sum_{i\in S_k} \log \hat{c}_{\lambda,-k}(U_i, V_i).$$

Hence, the likelihood CV score $\mathrm{KL}(\lambda)$ is defined as

$$\mathrm{KL}(\lambda) = -\frac{1}{K}\sum_{k=1}^K \frac{1}{|S_k|}\sum_{i\in S_k} \log \hat{c}_{\lambda,-k}(U_i, V_i).$$

Then, we choose $\lambda_{CV} = \arg\min \mathrm{CV}_{\lambda \in G}(\lambda)$ as the best tuning parameter, where G is a pre-specified discrete or continuous set in which λ is searched over, and CV score is either LS score or KL score. For simplicity, one usually pre-specifies G as a fine finite grid, where λ_{CV} is found by a simple grid search. For $\mathrm{CV}_{\lambda \in G}(\lambda)$ over a continuous region G, λ_{CV} may be found by some simple single variable minimization methods such as bisection method or golden section search method. One should make sure that λ_{CV} is not at the boundaries of the set G. In case λ_{CV} is located at the boundaries of the set G, one needs to enlarge the G and includes the added portion into the search.

Van der Laan et al.(2004) studied the choice of K. They established asymptotic optimality of K-fold CV, in the sense that the CV selector performs asymptotically as well (w.r.t. to the Kullback-Leibler distance to the true density) as an optimal benchmark model selector which depends on the true density. Crucial conditions of their theorem are that the size of the validation sample n/K goes to infinity, which excludes leave-one-out CV, and that the candidate density estimates are bounded away from zero and infinity. Some copula densities may not be bounded away from infinity, but it is not a concern for finite sample studies.

5 Postprocessing

A well-known drawback of TV regularized estimates is the staircase effect: the estimated values produced by TV regularization tend to cluster in patches. We observed these artifacts in our copula density estimates too. For example, the TV regularized copula density estimates displayed in Fig. 4(c), (e), (g) exhibit strong staircase effect.

To alleviate the staircase effect, we use a bilateral filter (BLF) (Tomasi and Manduchi 1998) as a postprocessing procedure. This strategy was used in Cai et al.(2009) to remove possible artifacts from noise in a frame-based image deblurring procedure. The BLF is chosen because it is edge-preserving, simple to implement and efficient at removing artifacts. We use the code of Paris and Durand (2009) provided in Chen (2007) for a BLF. The MATLAB code of Chen (2007) takes 0.02 second to produce the bilateral filtered version Fig. 4(d) from the input data Fig. 4(c).

6 Simulations

We report results from simulation studies which were designed to demonstrate the effectiveness of the MPLE with TV penalty subject to linear equality constraints for copula density estimation and the K-fold CV regularization parameter selector.

The stopping criteria of our augmented Lagrangian and operator-splitting algorithm were $||\boldsymbol{x}^{k+1}-\boldsymbol{x}^{k}||/||x^{k}|| <= 10^{-5}$ or total number of iterations reaching 20, where each iteration includes going through steps 1 and 2 once.

In the simulation, the marginal distributions F_X and F_Y were estimated by ECDFs (1). This amounts to use the standardized ranks of the sample $\{(X_i, Y_i)\}_{i=1}^n$ as estimates of $\{(U_i, V_i)\}_{i=1}^n$ (remind that $U_i = F_X(X_i)$ and $V_i = F_Y(Y_i)$). The CDF of a continuous random variable is continuous and increasing within its domain, which implies that the ranks of X_i's are the same as the ranks of U_i's, so are the ranks of Y_i's and those of V_i's. Therefore it is unnecessary to explicitly specify the F_X and F_Y in our simulation for copula density estimation. One can first generate $\{(U_i, V_i)\}_{i=1}^n$ from an underlying copula density $c(u, v)$, then use their standardized ranks as their estimates.

The setting of our simulation study is mostly the same as the one in Autin et al.(2009) as we intend to make a comparison with their wavelet thresholding estimates. We tested five parametric families of copulas: Gaussian, Student, Clayton, Frank and the Gumbel families. For each copula model, independent and identically distributed standard uniform random bivariate variables $\{(U_i, V_i)\}_{i=1}^n$ were generated from the specified copula with parameter θ using MATLAB's *copularnd*() function. That was, $\{U_i\}_{i=1}^n$ was a sample from a Uniform(0,1) distribution, and so was the $\{V_i\}_{i=1}^n$. The joint pdf of (U, V) was the specified copula density $c(u, v)$ with parameter θ. The sample sizes considered were $n = 500$ and $n = 2000$.

Various error measures were evaluated over the equally spaced grid points within $[0, 1]^2$ where the copula densities were estimated. For one data set, the quality of an estimate $\hat{c}_\lambda(u, v)$ of the true copula density $c(u, v)$ was measured by an error measure $\text{Loss}(\hat{c}_\lambda, c)$, which can be either relative errors

$$RE_q = \frac{||\hat{c}_\lambda - c||_{N,q}}{||c||_{N,q}}, \quad \text{for } q = 1,\ 2,\ \infty,$$

or the KLD (4). The sample average of an error measure $\text{Loss}(\hat{c}_\lambda, c)$ over replications of random data set approximates the population mean of the error measure $E(\text{Loss}(\hat{c}_\lambda, c))$ for the proposed

estimator $\hat{c}_\lambda(U,V)$. We replicated 100 times for each experiment setting and report the sample average, the associated standard errors (in parentheses) and boxplots of these 100 error measures in Tables 1-4 and Figs. 7-13, respectively.

The regularization parameter λ was chosen from the grid $G = \{0.01 \times 10^{2(i-1)/27}\}_{i=1}^{28}$, i.e., 28 equally spaced numbers in [0.01, 1] in a log10 scale. All the best regularization parameters were found near the central portion of this G. For the error measure $\text{Loss}(\hat{c}_\lambda, c)$, the best regularization parameter $\lambda_{Loss} = \arg\min_{\lambda\in G} \text{Loss}(\hat{c}_\lambda, c)$ and the best estimate is Loss-best=$\hat{c}_{\lambda_{Loss}}$. The CV data driven regularization parameter $\lambda_{CV} = \arg\min_{\lambda\in G} \text{CV}(\hat{c}_\lambda, c)$ and the data adaptive estimate TV-CV = $\hat{c}_{\lambda_{CV}}$. The closer the λ_{CV} is to λ_{best}, the better a TV-CV is in terms of $\text{Loss}(\hat{c}_\lambda, c)$.

For the number of folders in CV, we used $K = 10$. To see the effectiveness of the 10-fold CV regularization parameter selector, in Fig. 1, we plotted the curves of some $\text{Loss}(\hat{c}_\lambda, c)$ and $\text{CV}(\lambda)$ vs. λ respectively for a typical run of the case: Gaussian copula with $\theta = 0.5$, sample size $n = 2000$, and grid size $m = 64$. In this specific case, the λ_{RE2} which minimized $RE_2(\hat{c}_\lambda, c)$ for $\lambda \in G$ coincided with λ_{KLD} which minimized $\text{KLD}(\hat{c}_\lambda, c)$. The λ_{LS} which minimized $\text{LS}(\lambda)$ was 1 grid blow λ_{RE2}; The λ_{KL} which minimized $\text{KL}(\lambda)$ was 1 grid above λ_{KLD}. Fig. 2 shows the scatter plots of the original data $\{(U_i, V_i)\}_{i=1}^n$ and their standardized ranks $\{(\hat{U}_i, \hat{V}_i)\}_{i=1}^n$. Note the close similarity of these two plots. Fig. 3 displays the true and estimated copula densities. For comparison, we computed a 2D kernel density estimate using the kde2D program provided by Botev (2007). The RE2-best estimate catches the two peaks in the front and back corners well. Both the TV-LS and TV-KL are close to RE2-best. RE2-best-BLF, TV-LS-BLF and TV-KL-BLF are the bilateral filtered version of RE2-best, TV-LS and TV-KL, respectively and they exhibit less staircase effect.

Fig. 4 plots the true and estimated Frank copula density with $\theta = 4$ for $n = 1000$ and $m = 64$ in a typical run. We see that in the bilateral filtered version of the estimates, the staircase effect is alleviated.

To have a sense of the speed of the algorithm, for the data set used for Fig. 1, Fig. 5 plot the times in seconds needed to obtain the solution $\hat{c}_\lambda$ from the full data set for a sequence of λ and the times in seconds needed to obtain both the $\text{LS}(\lambda)$ and $\text{KL}(\lambda)$ for the 10-fold CV for a sequence of λ. For a fixed λ, 10-fold CV took 3.42 seconds on average to finish, while it took only 0.4 seconds on average to obtain $\hat{c}_\lambda$ for the full data set. This computational efficiency is an order of magnitude improvement over another TV-MPLE method (Qu et al. 2009) solved by log-barrier

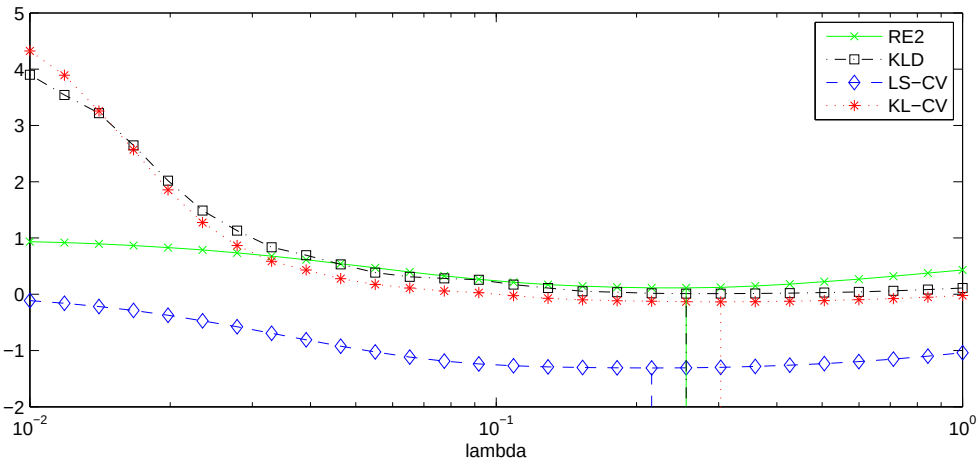

Figure 1: Loss and CV curves in a typical run of the case: Gaussian copula with $\theta = 0.5$, sample size $n = 2000$, grid size $m = 64$.

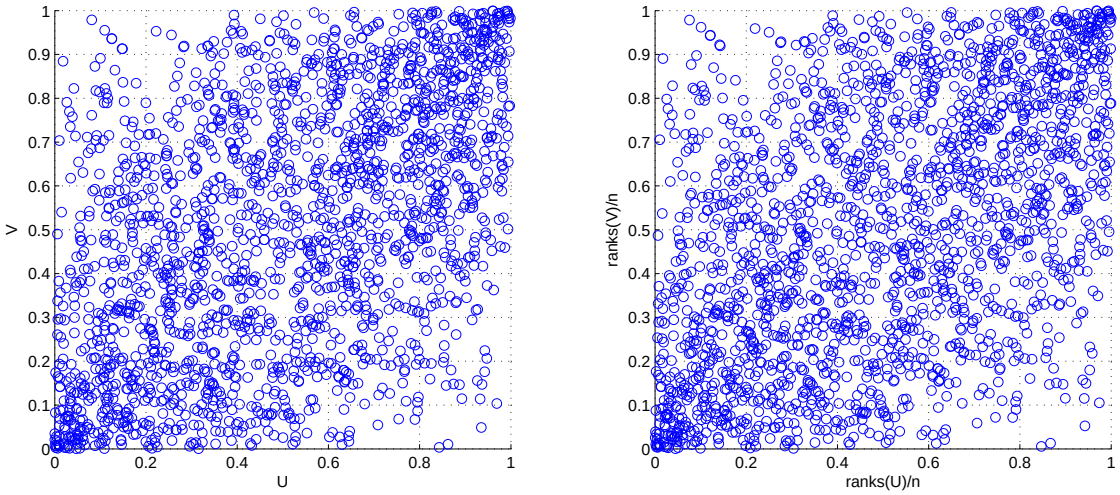

Figure 2: scatter plots of the original data $\{(U_i, V_i)\}_{i=1}^n$ and their standardized ranks $\{(\hat{U}_i, \hat{V}_i)\}_{i=1}^n$ in a typical run of the case: Gaussian copula with $\theta = 0.5$, sample size $n = 2000$, and grid size $m = 64$.

method for second order cone program (SOCP) which took 2 minutes to solve the same problem. We did not compare our proposed method in this paper with the method in Qu et al. (2009) because of the low computational efficiency of the latter.

The side-by-side boxplots in Figs. 6-13 show that both TV-LS and TV-KL are close to TV-best. In general, TV-LS is closer to RE_q-best than TV-KL; and TV-KL is closer to KL-best than

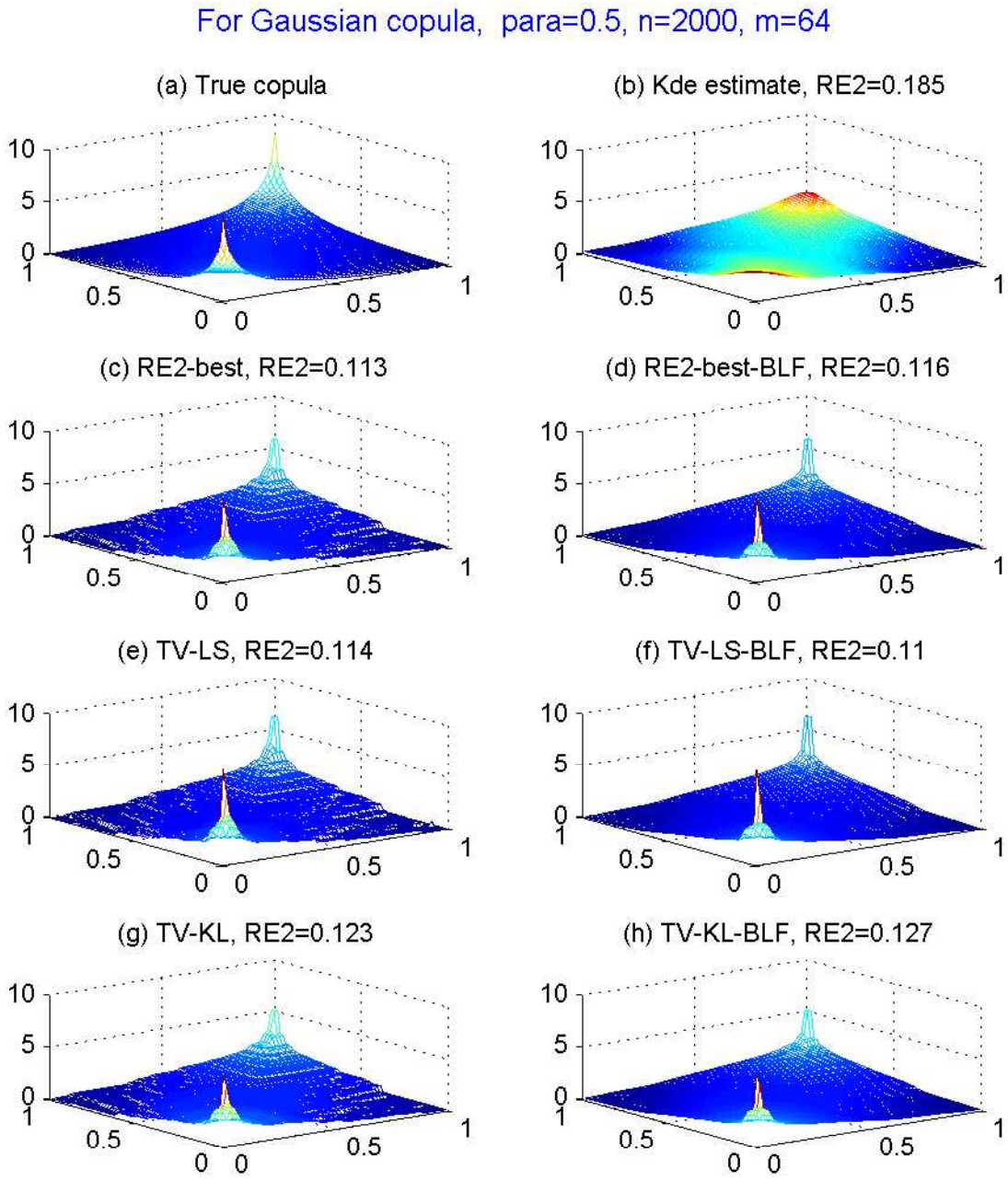

Figure 3: True and estimated copula densities in a typical run of the case: Gaussian copula with $\theta = 0.5$, sample size $n = 2000$, grid size $m = 64$.

TV-LS which is what we should expect because the goal of TV-LS is to minimize RE_2 and the goal of TV-KL is to minimize KLD.

Table 1, 3 and table 2, 4 list Monte Carlo approximations to $E(\text{Loss}(\hat{c}_\lambda, c))$ over 100 replications for respectively $n = 500,\ m = 32$ and $n = 2000,\ m = 64$. Comparing the mean RE_q of our TV-LS and TV-LS-BLF with those of WaveThresh-Local in Table A.3 and A.4 of Autin et al.(2009), we

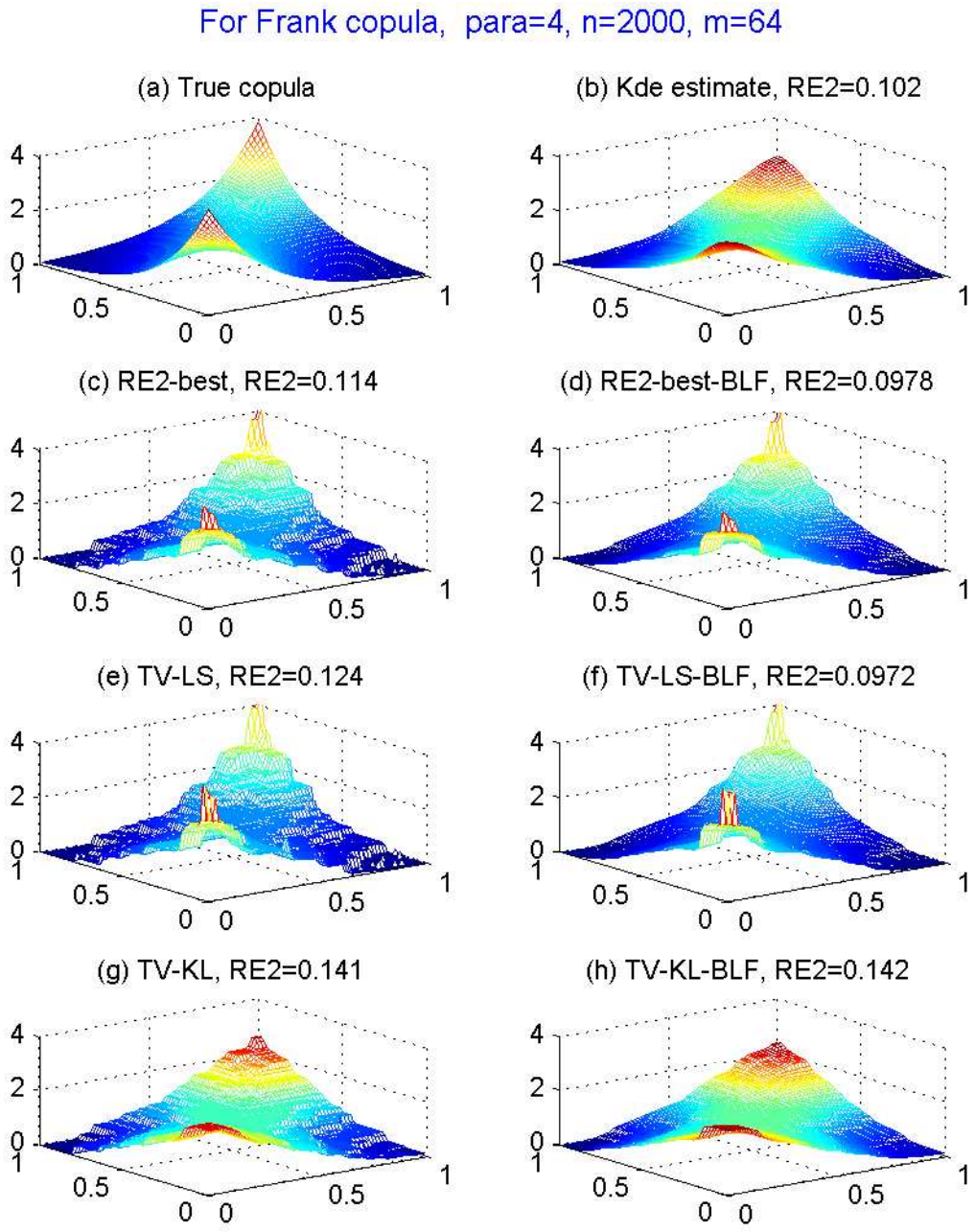

Figure 4: True and estimated copula densities in a typical run of the case: Frank copula with $\theta = 4$, sample size $n = 2000$, grid size $m = 64$.

observe that (1) the mean RE_1 of TV-LS is mostly smaller than those by WaveThresh-Local; (2) the mean RE_2 of TV-LS is all larger than those by WaveThresh-Local except for the Gumbel copula with $\theta = 8.3$; (3) the mean RE_∞ of TV-LS is all smaller than those by WaveThresh-Local except for the Frank copula with $\theta = 4.0$.

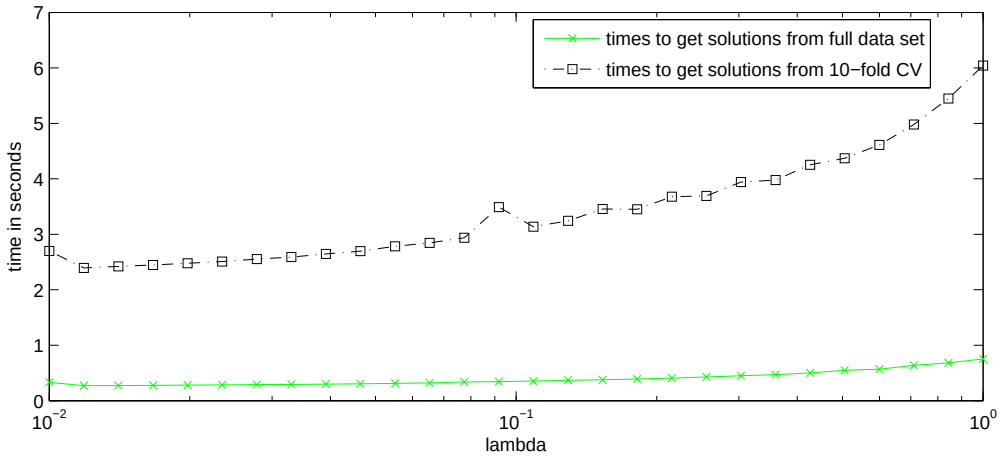

Figure 5: Times (in seconds) to get solutions for a sequence of λ in a typical run of the case: Gaussian copula with $\theta = 0.5$, sample size $n = 2000$, grid size $m = 64$.

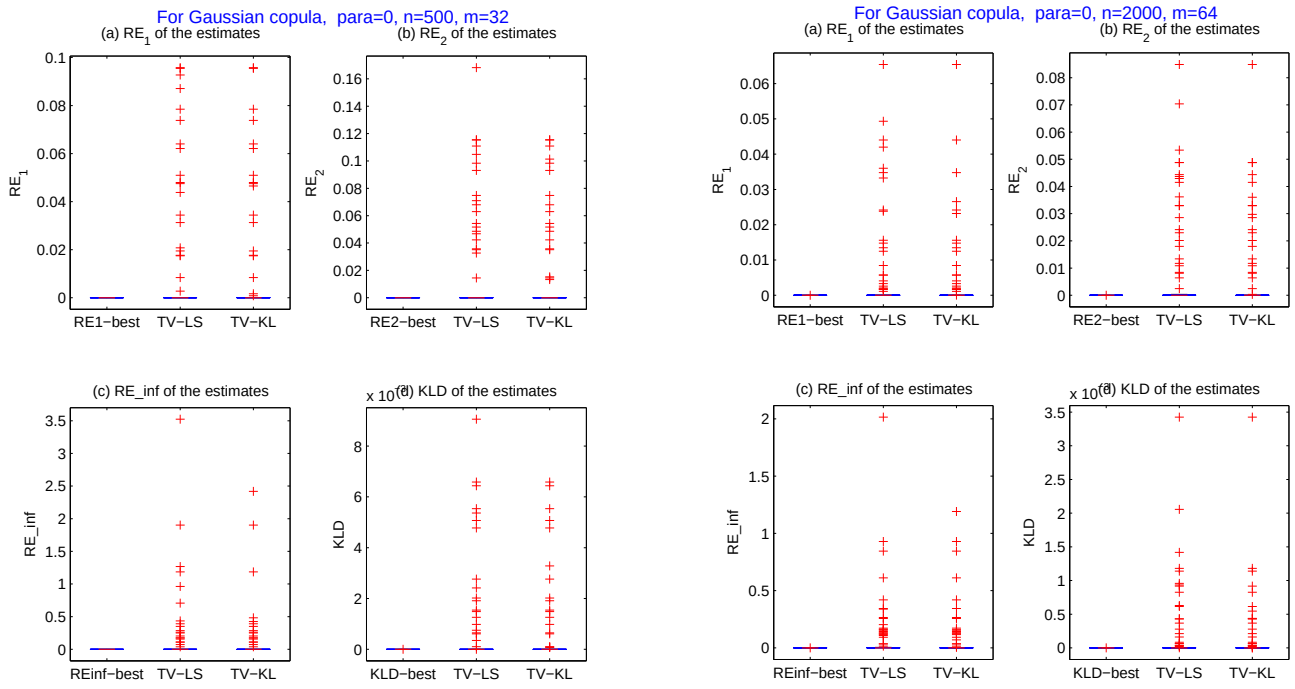

Figure 6: Boxplots of errors of different estimates for the Gaussian copula with $\theta = 0$.

7 Application to Real Data

We apply our MPLE-TV method to a subset of the Framingham Heart study data (http://www.framingham.com/heart/). We focus on the dependence structure underlying the diastolic (DBP) and the systolic (SBP) blood pressures (in mmHg) measured on 663 male subjects at their first

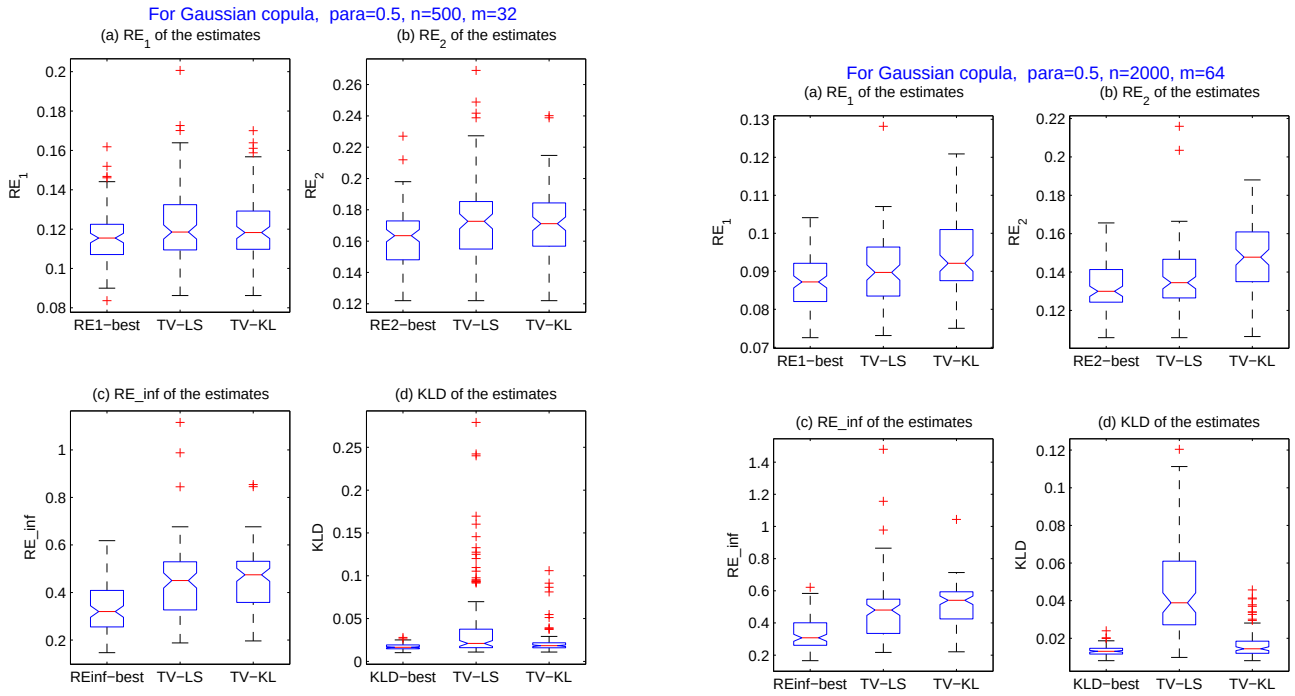

Figure 7: Boxplots of errors of different estimates for the Gaussian copula with $\theta = 0.5$.

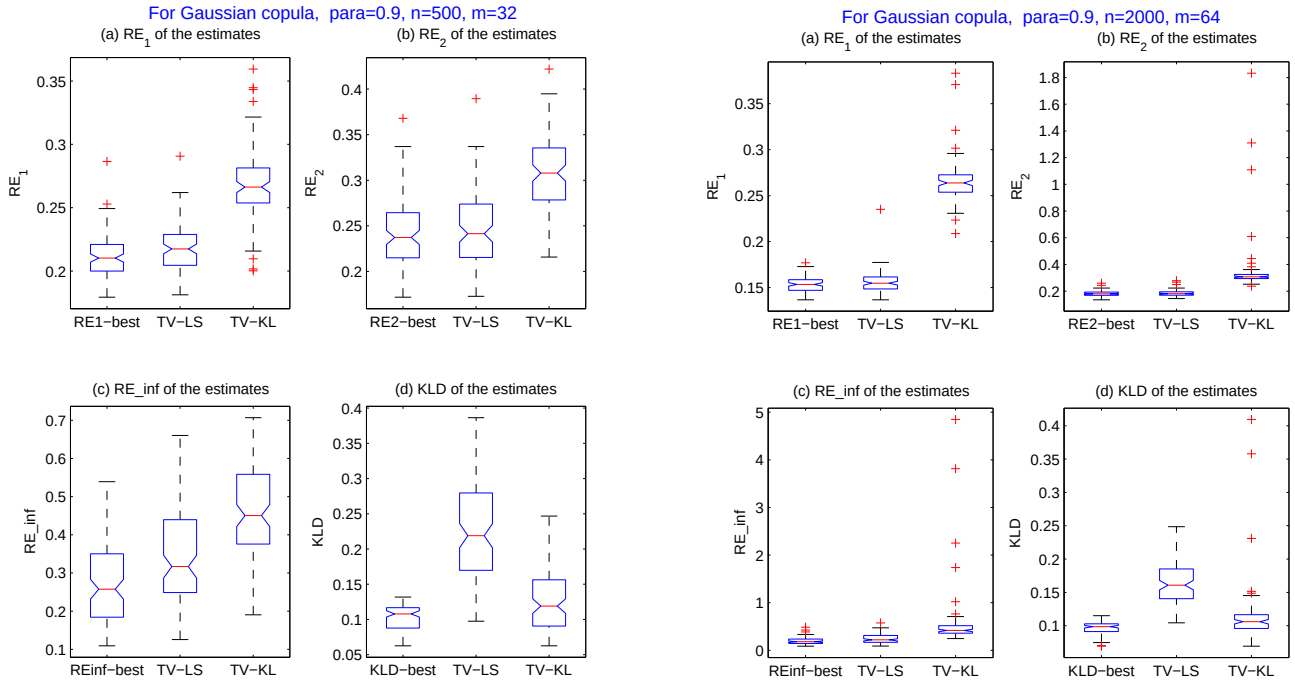

Figure 8: Boxplots of errors of different estimates for the Gaussian copula with $\theta = 0.9$. The number of replications is 100.

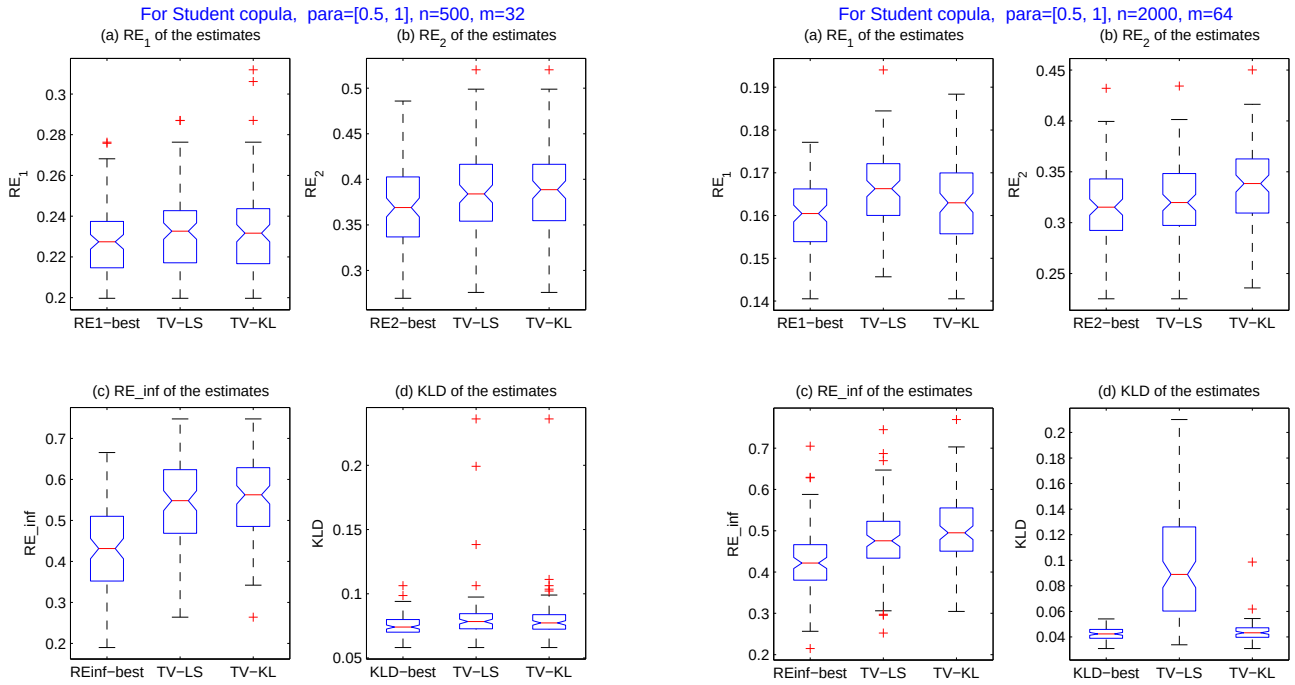

Figure 9: Boxplots of errors of different estimates for the Student copula with $\theta = (0.5, 1)$.

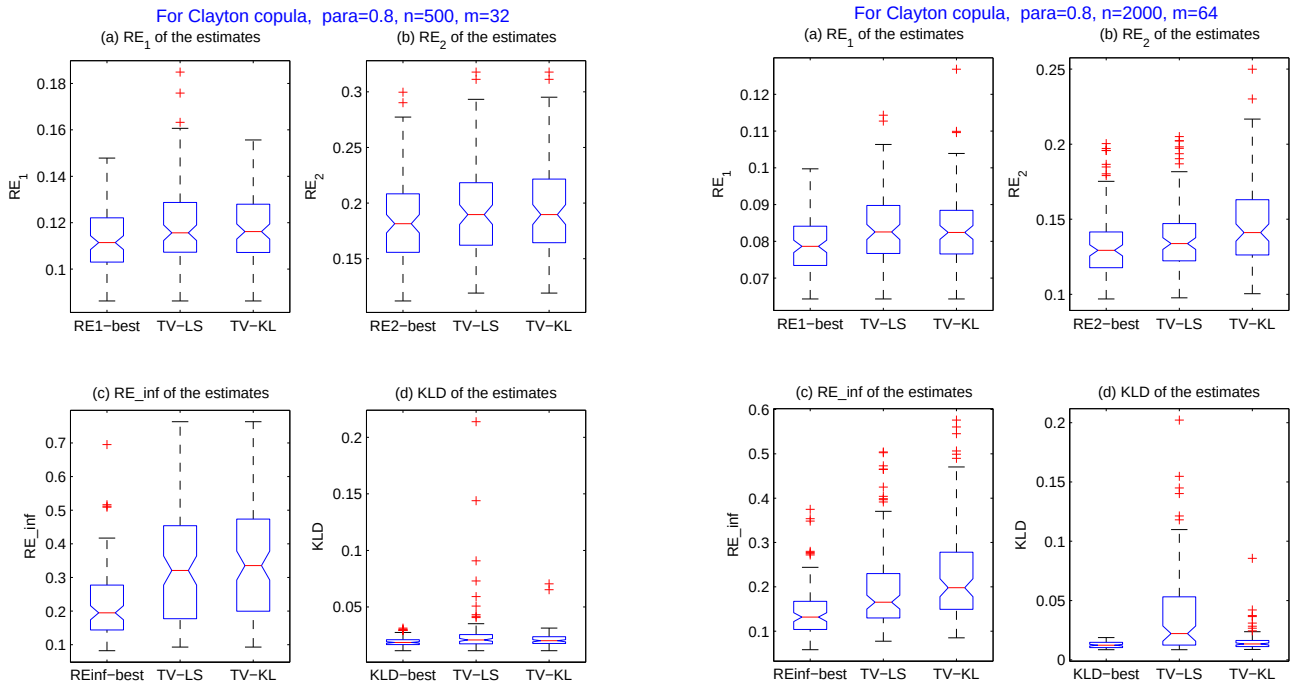

Figure 10: Boxplots of errors of different estimates for the Clayton copula with $\theta = 0.8$.

visit. The scatterplot of the log-blood pressures and the scatterplot of the standardized ranks of the log-blood pressures can be found in Fig. 14. It is evident that there is a strong positive

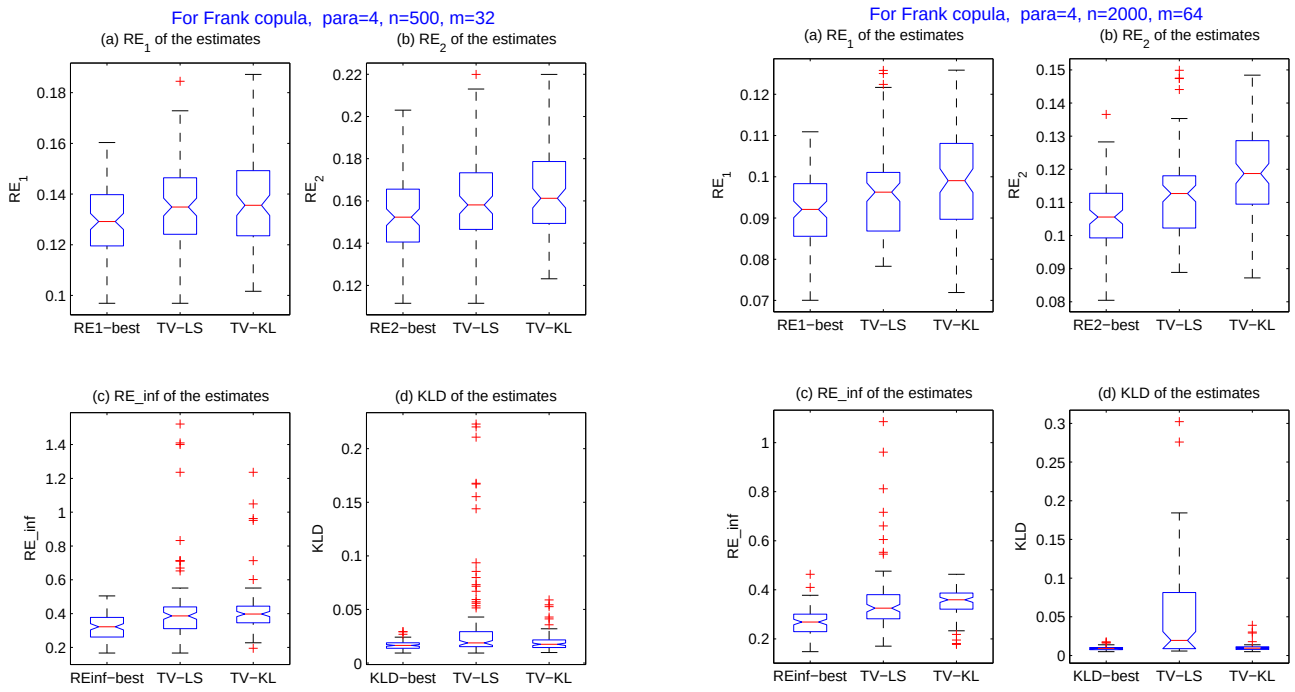

Figure 11: Boxplots of errors of different estimates for the Frank copula with $\theta = 4$.

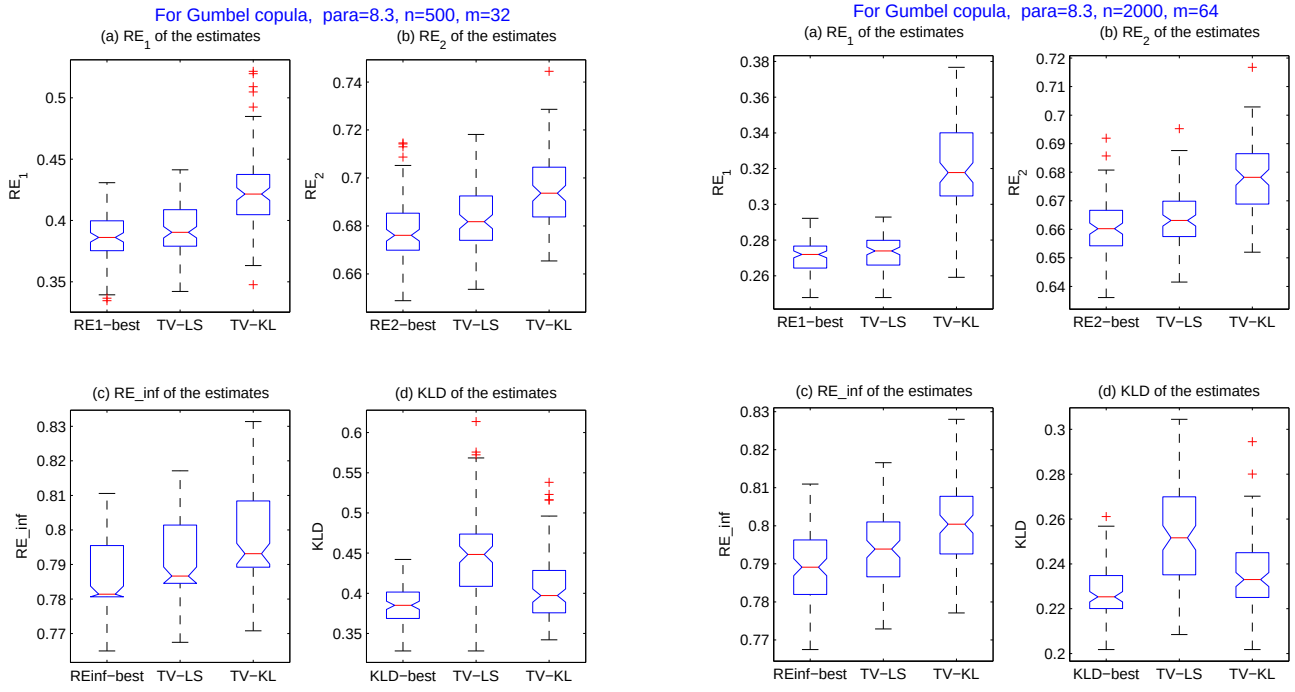

Figure 12: Boxplots of errors of different estimates for the Gumbel copula with $\theta = 8.3$.

dependence between the two responses. Lambert (2007) analyzed this data set assuming the

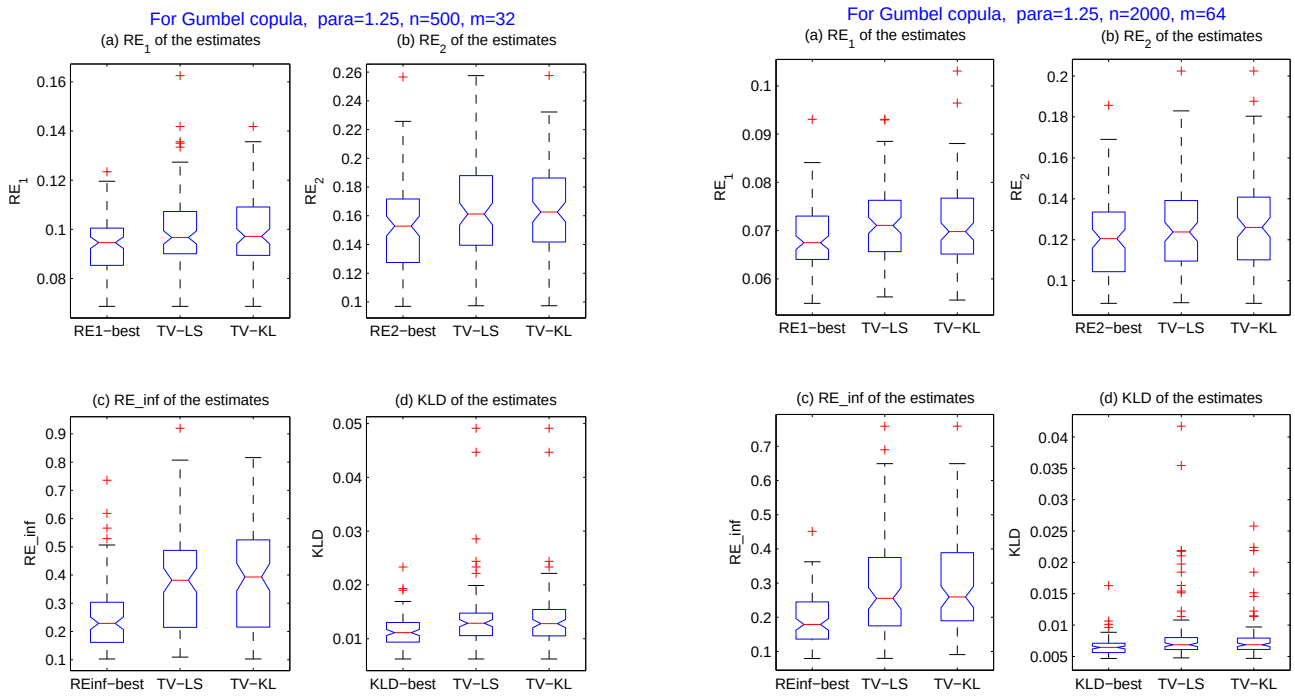

Figure 13: Boxplots of errors of different estimates for the Gumbel copula with $\theta = 1.25$.

copula density of the log-blood pressures was a sub-family of copulas named Archimedean with unknown (strict) generator. Lambert (2007) proposed a ratio approximation of the Archimedean copula generator and of its first derivative using B-splines, estimated the associated parameters using Markov chains Monte Carlo methods, and found that Gumbel copula was appropriate for this data without being fully satisfactory.

We applied our estimation procedure to this data set, and used 10-fold LS and KL CV to select the regularization parameter λ. The grid size m was set to 38. We estimated parametric copula densities by assuming Gumbel, Gaussian, Clayton and Frank copula respectively for the data as well. The parameters of the parametric estimates were estimated by the Canonical Maximum Likelihood (CML) method using MATLAB's $copulafit()$ function. We measure the distance between our nonparametric estimate $\hat{c}_\lambda$ and the parametric estimate $\hat{c}_{\hat{\theta}}$ by their relative errors

$$RE_q(\hat{\theta}) = \frac{||\hat{c}_\lambda - c_{\hat{\theta}}||_{N,q}}{||c_{\hat{\theta}}||_{N,q}}, \quad \text{for } q = 1,\ 2,\ \infty.$$

Table 5 lists these relative errors. We find that Gumbel copula is closet to our TV-LS-BLF estimate. This is in agreement with Lambert (2007)'s finding that Gumbel copula is appropriate for this data. Fig. 15 plots the TV-LS-BLF on the left panel and the Gumbel copula density on the right panel. They look similar, with some difference in the front corner.

Table 1: Monte Carlo approximations to $E(\text{Loss}(\hat{c}_\lambda, c))$ over 100 replications for $n = 500$, $m = 32$

Copula	par.	Method	RE_1	RE_2	RE_∞	KLD
Gaussian	0.00	TV-best	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)
Gaussian	0.00	TV-LS	0.010 (0.024)	0.014 (0.033)	0.126 (0.447)	0.001 (0.002)
Gaussian	0.00	TV-KL	0.008 (0.021)	0.012 (0.029)	0.090 (0.335)	0.000 (0.001)
Gaussian	0.50	TV-best	0.116 (0.013)	0.162 (0.020)	0.338 (0.103)	0.017 (0.003)
Gaussian	0.50	TV-LS	0.123 (0.018)	0.173 (0.026)	0.443 (0.156)	0.044 (0.053)
Gaussian	0.50	TV-KL	0.120 (0.016)	0.171 (0.022)	0.453 (0.127)	0.023 (0.016)
Gaussian	0.90	TV-best	0.211 (0.017)	0.241 (0.037)	0.263 (0.097)	0.104 (0.017)
Gaussian	0.90	TV-LS	0.218 (0.019)	0.247 (0.038)	0.333 (0.118)	0.225 (0.066)
Gaussian	0.90	TV-KL	0.269 (0.029)	0.308 (0.042)	0.448 (0.121)	0.127 (0.041)
Student	0.5, 1	TV-best	0.228 (0.017)	0.370 (0.049)	0.455 (0.106)	0.076 (0.009)
Student	0.5, 1	TV-LS	0.233 (0.019)	0.384 (0.052)	0.554 (0.109)	0.082 (0.022)
Student	0.5, 1	TV-KL	0.234 (0.021)	0.386 (0.051)	0.557 (0.106)	0.080 (0.019)
Clayton	0.80	TV-best	0.113 (0.013)	0.186 (0.040)	0.220 (0.110)	0.019 (0.004)
Clayton	0.80	TV-LS	0.119 (0.018)	0.195 (0.042)	0.332 (0.168)	0.027 (0.025)
Clayton	0.80	TV-KL	0.117 (0.014)	0.196 (0.042)	0.342 (0.170)	0.022 (0.008)
Frank	4.00	TV-best	0.129 (0.014)	0.154 (0.019)	0.323 (0.078)	0.017 (0.004)
Frank	4.00	TV-LS	0.135 (0.016)	0.161 (0.022)	0.425 (0.230)	0.036 (0.045)
Frank	4.00	TV-KL	0.137 (0.018)	0.163 (0.022)	0.418 (0.153)	0.020 (0.009)
Gumbel	8.30	TV-best	0.387 (0.021)	0.678 (0.013)	0.787 (0.012)	0.386 (0.025)
Gumbel	8.30	TV-LS	0.393 (0.022)	0.684 (0.014)	0.792 (0.012)	0.448 (0.053)
Gumbel	8.30	TV-KL	0.424 (0.032)	0.695 (0.016)	0.798 (0.013)	0.406 (0.042)
Gumbel	1.25	TV-best	0.093 (0.012)	0.154 (0.031)	0.253 (0.119)	0.011 (0.003)
Gumbel	1.25	TV-LS	0.100 (0.015)	0.165 (0.033)	0.387 (0.190)	0.014 (0.006)
Gumbel	1.25	TV-KL	0.100 (0.015)	0.165 (0.032)	0.393 (0.185)	0.014 (0.006)

8 Concluding Remarks

We presented a TV penalized maximum likelihood copula density estimate subject to the the constraints that the marginal distributions are standard uniforms. The linear equality constrained

Table 2: Monte Carlo approximations to $E(\text{Loss}(\hat{c}_\lambda, c))$ over 100 replications for $n = 2000$, $m = 64$

Copula	par.	Method	RE_1	RE_2	RE_∞	KLD
Gaussian	0.00	TV-best	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)
Gaussian	0.00	TV-LS	0.004 (0.012)	0.008 (0.017)	0.078 (0.250)	0.000 (0.000)
Gaussian	0.00	TV-KL	0.003 (0.010)	0.006 (0.014)	0.062 (0.190)	0.000 (0.000)
Gaussian	0.50	TV-best	0.086 (0.008)	0.128 (0.013)	0.319 (0.099)	0.013 (0.003)
Gaussian	0.50	TV-LS	0.091 (0.010)	0.134 (0.015)	0.440 (0.163)	0.051 (0.028)
Gaussian	0.50	TV-KL	0.095 (0.012)	0.150 (0.020)	0.514 (0.133)	0.014 (0.006)
Gaussian	0.90	TV-best	0.153 (0.008)	0.182 (0.020)	0.193 (0.069)	0.096 (0.010)
Gaussian	0.90	TV-LS	0.156 (0.012)	0.184 (0.023)	0.242 (0.089)	0.165 (0.031)
Gaussian	0.90	TV-KL	0.266 (0.022)	0.346 (0.201)	0.550 (0.602)	0.113 (0.044)
Student	0.5, 1	TV-best	0.160 (0.008)	0.318 (0.039)	0.424 (0.086)	0.043 (0.005)
Student	0.5, 1	TV-LS	0.166 (0.009)	0.323 (0.040)	0.478 (0.087)	0.096 (0.044)
Student	0.5, 1	TV-KL	0.164 (0.009)	0.339 (0.039)	0.503 (0.086)	0.044 (0.007)
Clayton	0.80	TV-best	0.079 (0.007)	0.134 (0.024)	0.145 (0.062)	0.013 (0.002)
Clayton	0.80	TV-LS	0.084 (0.010)	0.139 (0.025)	0.200 (0.102)	0.039 (0.038)
Clayton	0.80	TV-KL	0.083 (0.010)	0.147 (0.029)	0.233 (0.118)	0.016 (0.009)
Frank	4.00	TV-best	0.092 (0.008)	0.106 (0.010)	0.267 (0.062)	0.009 (0.002)
Frank	4.00	TV-LS	0.096 (0.010)	0.112 (0.013)	0.350 (0.143)	0.052 (0.062)
Frank	4.00	TV-KL	0.099 (0.012)	0.119 (0.013)	0.351 (0.059)	0.010 (0.005)
Gumbel	8.30	TV-best	0.271 (0.010)	0.660 (0.010)	0.790 (0.009)	0.227 (0.011)
Gumbel	8.30	TV-LS	0.273 (0.010)	0.663 (0.010)	0.794 (0.009)	0.253 (0.023)
Gumbel	8.30	TV-KL	0.320 (0.026)	0.678 (0.012)	0.801 (0.010)	0.235 (0.016)
Gumbel	1.25	TV-best	0.068 (0.007)	0.121 (0.020)	0.191 (0.073)	0.007 (0.002)
Gumbel	1.25	TV-LS	0.072 (0.007)	0.126 (0.022)	0.289 (0.151)	0.009 (0.006)
Gumbel	1.25	TV-KL	0.071 (0.008)	0.128 (0.023)	0.302 (0.151)	0.008 (0.003)

TV regularized MPLE problem is solved by an augmented Lagrangian combined with operator-splitting algorithm. A fast ROF/TV-L2 denoising solver is at the core of the method. The K-fold CV regularization parameter selector based on integrated squared error or Kullback-Leibler

Table 3: Monte Carlo approximations to $E(\text{Loss}(\hat{c}_\lambda, c))$ over 100 replications for $n = 500$, $m = 32$

Copula	par.	Method	RE_1	RE_2	RE_∞	KLD
Gaussian	0.00	TV-best-BLF	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)
Gaussian	0.00	TV-LS-BLF	0.009 (0.022)	0.014 (0.031)	0.126 (0.446)	0.001 (0.001)
Gaussian	0.00	TV-KL-BLF	0.008 (0.020)	0.011 (0.028)	0.090 (0.335)	0.000 (0.001)
Gaussian	0.50	TV-best-BLF	0.100 (0.016)	0.152 (0.022)	0.357 (0.108)	0.013 (0.005)
Gaussian	0.50	TV-LS-BLF	0.103 (0.019)	0.161 (0.027)	0.458 (0.153)	0.014 (0.008)
Gaussian	0.50	TV-KL-BLF	0.107 (0.021)	0.165 (0.027)	0.468 (0.127)	0.015 (0.006)
Gaussian	0.90	TV-best-BLF	0.195 (0.022)	0.242 (0.035)	0.270 (0.099)	0.104 (0.022)
Gaussian	0.90	TV-LS-BLF	0.194 (0.028)	0.249 (0.041)	0.338 (0.122)	0.054 (0.011)
Gaussian	0.90	TV-KL-BLF	0.284 (0.037)	0.329 (0.040)	0.457 (0.129)	0.091 (0.018)
Student	0.5, 1	TV-best-BLF	0.222 (0.021)	0.367 (0.047)	0.460 (0.110)	0.075 (0.011)
Student	0.5, 1	TV-LS-BLF	0.228 (0.026)	0.393 (0.052)	0.560 (0.114)	0.077 (0.013)
Student	0.5, 1	TV-KL-BLF	0.230 (0.027)	0.396 (0.050)	0.563 (0.112)	0.078 (0.013)
Clayton	0.80	TV-best-BLF	0.096 (0.014)	0.178 (0.041)	0.227 (0.109)	0.021 (0.004)
Clayton	0.80	TV-LS-BLF	0.100 (0.017)	0.188 (0.044)	0.336 (0.166)	0.023 (0.006)
Clayton	0.80	TV-KL-BLF	0.101 (0.015)	0.192 (0.044)	0.347 (0.169)	0.022 (0.004)
Frank	4.00	TV-best-BLF	0.116 (0.016)	0.144 (0.020)	0.332 (0.075)	0.014 (0.005)
Frank	4.00	TV-LS-BLF	0.124 (0.020)	0.154 (0.023)	0.430 (0.227)	0.017 (0.011)
Frank	4.00	TV-KL-BLF	0.130 (0.021)	0.159 (0.024)	0.425 (0.149)	0.015 (0.006)
Gumbel	8.30	TV-best-BLF	0.412 (0.027)	0.678 (0.013)	0.789 (0.012)	0.405 (0.031)
Gumbel	8.30	TV-LS-BLF	0.427 (0.034)	0.687 (0.014)	0.793 (0.013)	0.359 (0.022)
Gumbel	8.30	TV-KL-BLF	0.478 (0.046)	0.700 (0.016)	0.798 (0.013)	0.397 (0.033)
Gumbel	1.25	TV-best-BLF	0.084 (0.015)	0.148 (0.034)	0.260 (0.120)	0.006 (0.004)
Gumbel	1.25	TV-LS-BLF	0.088 (0.017)	0.157 (0.036)	0.389 (0.191)	0.006 (0.005)
Gumbel	1.25	TV-KL-BLF	0.091 (0.018)	0.160 (0.036)	0.397 (0.186)	0.007 (0.005)

distance works well.

The theoretical questions such as the consistency and convergence rate of the estimator wait to be investigated. Few work exists regarding the asymptotic properties of TV regularized estimators,

Table 4: Monte Carlo approximations to $E(\text{Loss}(\hat{c}_\lambda, c))$ over 100 replications for $n = 2000$, $m = 64$

Copula	par.	Method	RE_1	RE_2	RE_∞	KLD
Gaussian	0.00	TV-best-BLF	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)	0.000 (0.000)
Gaussian	0.00	TV-LS-BLF	0.004 (0.011)	0.007 (0.016)	0.078 (0.250)	0.000 (0.000)
Gaussian	0.00	TV-KL-BLF	0.003 (0.009)	0.006 (0.014)	0.062 (0.190)	0.000 (0.000)
Gaussian	0.50	TV-best-BLF	0.071 (0.009)	0.125 (0.015)	0.350 (0.102)	0.011 (0.003)
Gaussian	0.50	TV-LS-BLF	0.070 (0.010)	0.127 (0.017)	0.461 (0.161)	0.007 (0.002)
Gaussian	0.50	TV-KL-BLF	0.089 (0.015)	0.156 (0.020)	0.539 (0.130)	0.012 (0.003)
Gaussian	0.90	TV-best-BLF	0.181 (0.012)	0.238 (0.018)	0.217 (0.060)	0.109 (0.023)
Gaussian	0.90	TV-LS-BLF	0.178 (0.018)	0.242 (0.021)	0.256 (0.080)	0.044 (0.008)
Gaussian	0.90	TV-KL-BLF	0.313 (0.022)	0.397 (0.197)	0.554 (0.603)	0.105 (0.017)
Student	0.5, 1	TV-best-BLF	0.182 (0.009)	0.340 (0.032)	0.427 (0.089)	0.060 (0.007)
Student	0.5, 1	TV-LS-BLF	0.171 (0.015)	0.353 (0.034)	0.480 (0.090)	0.048 (0.006)
Student	0.5, 1	TV-KL-BLF	0.195 (0.014)	0.377 (0.033)	0.505 (0.089)	0.058 (0.007)
Clayton	0.80	TV-best-BLF	0.068 (0.008)	0.155 (0.025)	0.180 (0.067)	0.014 (0.003)
Clayton	0.80	TV-LS-BLF	0.069 (0.009)	0.156 (0.028)	0.223 (0.095)	0.012 (0.002)
Clayton	0.80	TV-KL-BLF	0.077 (0.012)	0.172 (0.027)	0.258 (0.106)	0.014 (0.003)
Frank	4.00	TV-best-BLF	0.079 (0.010)	0.096 (0.012)	0.279 (0.057)	0.007 (0.003)
Frank	4.00	TV-LS-BLF	0.083 (0.013)	0.102 (0.015)	0.355 (0.141)	0.008 (0.005)
Frank	4.00	TV-KL-BLF	0.096 (0.015)	0.118 (0.015)	0.360 (0.053)	0.008 (0.003)
Gumbel	8.30	TV-best-BLF	0.440 (0.013)	0.670 (0.009)	0.791 (0.010)	0.306 (0.021)
Gumbel	8.30	TV-LS-BLF	0.445 (0.016)	0.677 (0.009)	0.795 (0.010)	0.258 (0.010)
Gumbel	8.30	TV-KL-BLF	0.511 (0.032)	0.695 (0.013)	0.801 (0.010)	0.303 (0.023)
Gumbel	1.25	TV-best-BLF	0.060 (0.007)	0.128 (0.022)	0.225 (0.070)	0.005 (0.002)
Gumbel	1.25	TV-LS-BLF	0.060 (0.008)	0.132 (0.025)	0.308 (0.142)	0.004 (0.002)
Gumbel	1.25	TV-KL-BLF	0.064 (0.011)	0.138 (0.025)	0.321 (0.141)	0.005 (0.002)

even in denoising problems. The consistency theorems in 1D TV-denoising problems have recently been proved in Dumbgen and Kovac (2009). Much work is ahead to establish the consistency results for TV regularized density estimators.

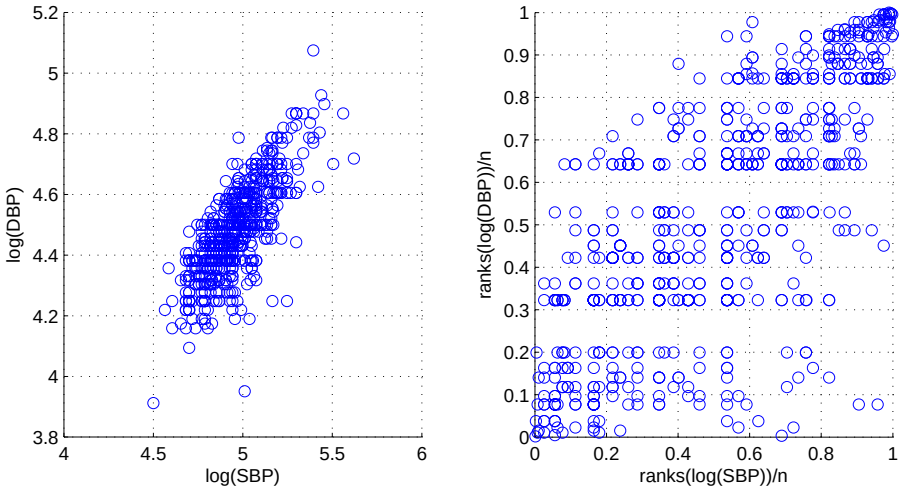

Figure 14: Left: log(DBP) vs. log(SBP); Right: standardized ranks of log(DBP) vs. those of standardized ranks of log(SBP)

Table 5: Relative error $RE_q(\hat{\theta})$ for the real data with $n = 663$, $m = 38$

MPLE-TV Estimate	Parametric Estimate	$RE_1(\hat{\theta})$	$RE_2(\hat{\theta})$	$RE_\infty(\hat{\theta})$
TV-LS-BLF	Gumbel	0.1915	0.2956	0.2632
TV-KL-BLF	Gumbel	0.2954	0.3969	0.4274
TV-LS-BLF	Gaussian	0.1803	0.3438	0.7624
TV-KL-BLF	Gaussian	0.2788	0.3898	0.8331
TV-LS-BLF	Clayton	0.3602	0.6959	0.8890
TV-KL-BLF	Clayton	0.3646	0.6837	0.9277
TV-LS-BLF	Frank	0.2054	0.3817	2.8122
TV-KL-BLF	Frank	0.3026	0.3857	1.9624

The MATLAB code implementing the method is available on the authors's website.

Acknowledgment

We thank Philippe Lambert for providing the Framingham Heart study data.

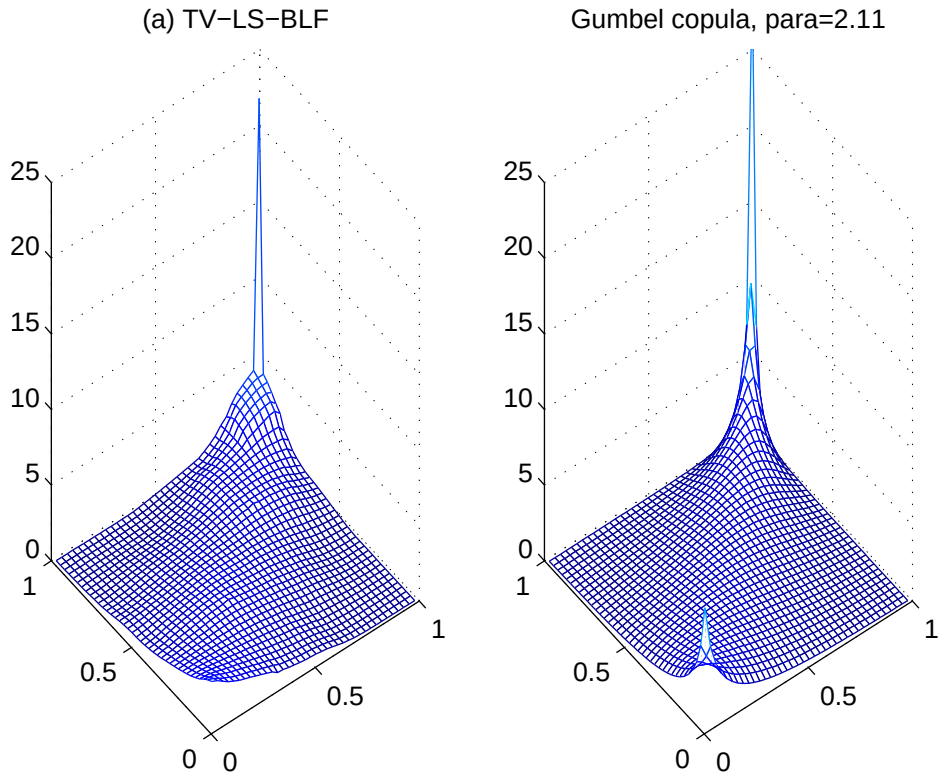

Figure 15: Left: TV based estimate, with λ chosen by 10-fold least squares (LS) cross-validation(CV), post-processed by bilateral linear filter (BLF); Right: parametric estimate assuming Gumbel copula

The second author's work was supported in part by NSF Grant DMS-07-48839, ONR Grant N00014-08-1-1101, and an Alfred P. Sloan Research Fellowship.

References

Autin, F., Pennecb, E. Le, and Tribouley, K. 2009. Thresholding methods to estimate the copula density. *Journal of Multivariate Analysis*, in press, doi:10.1016/j.jmva.2009.07.009.

Botev, Z. I. 2007. Nonparametric density estimation via diffusion mixing. *Postgraduate series* , Department of Mathematics, The University of Queensland, *http://www.maths.uq.edu.au/~botev/*

Cai, J., Osher, S. and Shen, Z. 2009. Linearized Bregman Iterations for Frame-Based Image Deblurring. *SIAM J. Imaging Sciences*, 2, 226-252.

Chambolle, A. 2005. Total variation minimization and a class of binary MRF models. *Tech. Report UMR CNRS 7641, Ecole Polytechnique.*

Chen, J. 2007. *http://people.csail.mit.edu/jiawen/#code.*

Darbon, J. and Sigelle, M. 2006. Image restoration with discrete constrained total variation, Part I: fast and exact optimization. *Journal of Mathematical Imaging and Vision*, 26 , 261-276.

Dumbgen, L. and Kovac, A. 2009. Extensions of smoothing via taut strings. *Electronic Journal of Statistics*, 3, 41-75.

Fermanian, J.D. and Scaillet, O., 2003. Nonparametric estimation of copulas for time series. *Journal of Risk*, 5(4), 2554.

Gijbels, I., Mielniczuk, J., 1990. Estimating the density of a copula function," *Communications in Statistics - Theory and Methods* 19, 445-464.

Glowinski, R. and Tallec, P., 1989. *Augmented Lagrangian and Operator-Splitting Methods*, SIAM.

Goldfarb, D. and Yin, W. , 2007. Parametric maximum flow algorithms for fast total variation minimization," *Rice University CAAM Technical Report TR07-09.*

Goldstein, T. and Osher, S., 2009. The Split Bregman method for L1 regularized problems. *SIAM J. Imaging Sciences*, 2(2), 323-343.

Hall, P., 1987). On Kullback-Leibler Loss and Density Estimation. *Ann. Statist*, 15(4), 1491-1519.

Hall, P., Neumeyer, N., 2006. Estimating a bivariate density when there are extra data on one or both components. *Biometrika*, 93, 439-450.

Koenker, R., Mizera, I., 2007. Density estimation by total variation regularization. *Aadvances in Statistical Modeling and Inference Essays in Honor of Kjell A Doksum*, V. Nair (ed.). World Scientific, 613-634.

Lambert, P., 2007. Archimedean Copula Estimation Using Bayesian Splines Smoothing Techniques. *Computational Statistics & Data Analysis*, 51(12), 6307-6320.

Li, C., Yin, W. and Zhang, Y., 2009. TVAL3: TV Minimization by Augmented Lagrangian and ALternating Direction Algorithms. *http://www.caam.rice.edu/ optimization/L1/TVAL3/*

Liu , L., Levine, M. and Zhu, Y., 2009. A Functional EM Algorithm for Mixing Density Estimation via Nonparametric Penalized Likelihood Maximization. *Journal of Computational and Graphical Statistics*, 18(2), 481-504

Malevergne, Y. and Sornette, D., 2006. *Extreme Financial Risks*, Heidelberg: Springer.

Mohler, G. O., Bertozzi, A. L. Goldstein, T. A., and Osher, S. J., 2009. Fast TV regularization for 2D Maximum Penalized Likelihood Estimation. *UCLA Computational and Applied Mathematics Reports*.

Paris, S., and Durand, F., 2009. A fast approximation of the bilateral filter using a signal processing approach. *International Journal of Computer Vision, Special Issue: Best of the European Conference on Computer Vision 2006 (ECCV'06)*, 81(1), 24-52.

Qu, L., Qian, Y. and Xie, H., 2009. Copula Density Estimation by Total Variation Penalized Likelihood. *Communications in Statistics - Simulation and Computation*, 38(9),1891-1908,

Sancetta, A., Satchell, S., 2004. The Bernstein copula and its applications to modeling and approximations of multivariate distributions. *Econometric Theory* 20, 535-562.

Sardy, S., Tseng, P., 2009. Density estimation by total variation penalized likelihood driven by the sparsity L1 information criterion. *Scandinavian Journal of Statistics*, in press.

Shih, J. H. and Louis, T. A., 1995. Inferences on the association parameter in copula models for bivariate survival data. *Biometrics*, 51, 1384-1399.

Sklar, A., 1959. Fonctions de rpartition n dimensions et leurs marges. *Publ Inst Stat Univ Paris*, 8, 229-231.

Tomasi, C. Manduchi, R., 1998. Bilateral filtering for gray and color images. *Proceedings of the 1998 IEEE International Conference on Computer Vision, Bombay, India*, pp. 839846.

Van der Laan, M. J., Dudoit, S., and Keles, S., 2004. Asymptotic optimality of likelihood based cross-validation. *Statistical Applications in Genetics and Molecular Biology*, 3, Article 4.

Yin, W., 2007. A parametric max-flow code for total variation and non-local total variation minimization. *http://www.caam.rice.edu/~wy1/ParaMaxFlow/*