Copula-based simulation of multivariate time series by Qinqin Huang, Yongyi Tang, Xiaohan Shen, and Pai Peng

1 Introduction

2 Modeling

2.1 Copulae

In this section, we basically infer the models from the paper by Aas et al. 1. #### 2.1.1 C-Vine Considering a multivariate cumulated distribution function F of n random variables $\mathbf{X} = X_1, ..., X_n$ with marginals $F_1(x_1), ..., F_n(x_n)$, Skalar's Theorem states that there exists a unique n-dimensional copula C to describte the joint distribution of these these variables. If F is absolutely continuous with strictly increasing and continuous marginal cdf F_i, the joint density function f can be written as:

$$f(x_1, x_2, ..., x_n) = c_{1:n}(F_1(x_1), F_2(x_2), ..., F_n(x_n)) \cdot \prod_{i=1}^{n} f_i(x_i).$$

which is the product of the n-dimensional copula density $c_{1:n}(\cdot)$ of C and the marginal densities $f_i(\cdot)$.

Building high-dimensional copulae is generally recognized as a challenging task. One of the most popular methods is the pair-copula construction (PCC) proposed by Aas et al. [1]1. The idea is to construct a high-dimensional copula by combining bivariate copulae. The basic principle behind PCC is that the density can be factorized as:

$$f(x_1, x_2, ..., x_n) = f_n(x_n) \cdot f(x_{n-1}|x_n) \cdot f(x_{n-2}|x_{n-1}, x_n) \cdot ... \cdot f(x_1|x_2, ..., x_n). \quad (1)$$

Each term in (1) can be decomposed into the appropriate pair-copula times a conditional marginal density, using the general formula:

$$f(x|\mathbf{v}) = c_{xv_j|\mathbf{v}_{-j}}\{F(x|\mathbf{v}_{-j}), F(v_j|\mathbf{v}_{-j})\} \cdot f_x(\mathbf{v}_{-j}).$$

Here $\mathbf{v}$ is a vector of variables, $\mathbf{v}_{-j}$ is the vector $\mathbf{v}$ with the j-th element removed.

The pair-copula construction involves marginal conditional distribution of the form $F(x|\mathbf{v})$. For every j :

$$F(x|\mathbf{v}) = \frac{\partial C_{xv_j|\mathbf{v}_{-j}}\{F(x|\mathbf{v}_{-j}), F(v_j|\mathbf{v}_{-j})\}}{\partial F(v_j|\mathbf{v}_{-j})}.$$

For the special case where v is a univariate, we have:

$$F(x|v) = \frac{\partial C_{xv}(F(x), F(v))}{\partial F(v)}.$$

We will use the function $h(x, v, \Theta)$ to represent this conditional distribution function when x and v are uniform:

$$h(x, v, \Theta) = F(x|v) = \frac{\partial C_{xv}(F(x), F(v))}{\partial F(v)}, \Theta\text{−the set of parameters of the joint distribution function.}$$

Furture, let $h^{-1}(u, v, \Theta)$ be the inverse of the h-function with respect to u, or the equivalently the inverse of the conditional distribution function.

For high-dimension distribution, there are significant number of possible pair-copular. To help organising them, Bedford and Cooke have introduced a graphical model denoted as the regular vine. Here, we concentrate on the special case of regular vines - the canonical vine (C-vine), which gives a specific way of decomposing the density. The figure below shows a C-vine with 5 variables.

The n-dimensional density corresponding to a C-vine is given by:

$$\prod_{k=1}^{n} f(x_k) \prod_{j=1} n-1 \prod_{i=1}^{n-j} c_{i,i+j|1,\ldots,j-1}\{F(x_j|x_1, ..., x_{j-1}), F(x_{i+j}|x_1, ..., x_{i+j-1})\}.$$

Fitting a canonical vine might be advantageous when a particular variable is known to be a key variable that governs interaction in the data set. In such a situation one may decide to locate this variable at the root of the canonical vine, as we have done with variable in the figure.

2.1.2 Conditional independence Assuming conditional independence may reduce the number of levels of the pair-copula decomposition, and hence simplify the construction.

In general, for any vector of variables $\mathbf{v}$, and two variables X and Y, the latter are conditionally independent given $\mathbf{V}$ if and only if:

$$c_{xy|\mathbf{v}}\{F(x|\mathbf{v}), F(y|\mathbf{v})\} = 1.$$

2.1.3 Bivariate copulae

2.1.4 Simulation from a pair-copula decomposed model In this section we show the simulation algorithm for canonical vines which follows the method discussed in Bruneau (2019) We assume for simplicity that the margins of the distribution are uniform. [^1]: For variables with other marginal distributions, we transform the data to uniform marginals before fitting the vine copula.

To sample n dependent uniform[0, 1] variables, we first sample $w_1, ..., w_n$ independent uniform on [0, 1]. Then set

$$x_1 = w_1,$$

$$x2 = F^{-1}(w_2|x_1),$$

$$x3 = F^{-1}(w_3|x_1, x_2), ...$$

$$x_n = F^{-1}(w_n|x_1, ..., x_{n-1}).$$

To determine the conditional distribution $F^{-1}(w_j|x_1, ..., x_{j-1})$, we use the h-function.

The algorithm gives the procesdure for sampling from a canonical vine. The outer loop runs over the variables to be sampled. This loop consists of two other for-loops. In the first, the ith varaible is sampled, while in the other, the conditional distribution functions needed for sampling the $(i + 1)$th variable are updated. To compute these conditional distribution functions, we repeatedly use the h-function, with previously computed conditional distribution functions, $v_{i,j} = F(x_i|x_1, ..., x_{j-1})$, as the first two arguments. The last argument of the h-function is the parameter $\Theta_{j,i}$ of the corresponding copula density $c_{j,j+i}|1, ..., j-1(\cdot, \cdot)$

The algorithm is as follows:

Algorithm 1 Simulation algorithm for a canonical vine.
Generates one sample $x_1, \ldots, x_n$ from the vine.

Sample $w_1, \ldots, w_n$ independent uniform on [0,1].
$x_1 = v_{1,1} = w_1$
for $i \leftarrow 2, \ldots, n$
    $v_{i,1} = w_i$
    for $k \leftarrow i-1, i-2, \ldots, 1$
        $v_{i,1} = h^{-1}(v_{i,1}, v_{k,k}, \Theta_{k,i-k})$
    end for
    $x_i = v_{i,1}$
    if $i == n$ **then**
        Stop
    end if
    for $j \leftarrow 1, \ldots, i-1$
        $v_{i,j+1} = h(v_{i,j}, v_{j,j}, \Theta_{j,i-j})$
    end for
end for

Figure 1: simulation algorithm

2.1.5 Estimation of the parameters In this section we describe how the parameters of the canonical vine density are estimated. To simplify the process as mentioned before, we assumme that the marginals are uniform and the the time series is stationary and independent over time. This assumption is not restrictive, as we can always transform the data to uniform marginals before fitting the vine copula.

We use the maximum likelihood method to estimate the parameters of the canonical vine. Since the actual margins are normally unknown in practice, what is being maximised is a pseudo-likelihood.

The log-likelihood is given by:

$$\sum_{j=1}^{n-1}\sum_{i=1}^{n-j}\sum_{t=1}^{T}\log c_{j,j+i|1,...,j-1}\{F(x_{j,t}|x_{1,t},...,x_{j-1,t}),F(x_{i+j,t}|x_{1,t},...,x_{j-1,t})\}.$$

For each copula in the above formula, there is at least one parameter to be determined. The algorithm for estimating the parameters is listed below in the figure. The ourter for-loop corresponds to the outer sum in the pseudo-likelihood. The inner for-loop corresponds to the sum over i. The innermost for-loop corresponds to the sum over the time series. Here, the element t of $textbfv_Pj,i$ is $v_{j,i,t} = F(x_{i+j,t}|x_{1,t},...,x_{j,t})$. $L(\mathbf{x},\mathbf{v},\Theta)$ is the log-likelihood of the chosen bivariate copula with parameters Θ and the data $\mathbf{x}$ and $\mathbf{v}$. That is,

$$L(\mathbf{x},\mathbf{v},\Theta)=\sum_{t=1}^{T}\log c(x_t,v_t,\Theta), c(u,v,\Theta)\text{is the density of the bivariate copula with parameters } \Theta.$$

2.2 Distribution of the data

2.3 CCVaR

3 Implementation

3.1 Data

3.2 Code

3.3 Results

4 Conclusion

References

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

log-likelihood $= 0$
for $i \leftarrow 1, \ldots, n$
  $\boldsymbol{v}_{0,i} = \boldsymbol{x}_i$.
end for
for $j \leftarrow 1, \ldots, n-1$
  for $i \leftarrow 1, \ldots, n-j$
    log-likelihood $=$ log-likelihood
      $+ L(\boldsymbol{v}_{j-1,1}, \boldsymbol{v}_{j-1,i+1}, \Theta_{j,i})$
  end for
  if $j == n-1$ **then**
    Stop
  end if
  for $i \leftarrow 1, \ldots, n-j$
    $\boldsymbol{v}_{j,i} = h(\boldsymbol{v}_{j-1,i+1}, \boldsymbol{v}_{j-1,1}, \Theta_{j,i})$
  end for
end for

Figure 2: estimation algorithm

[2] Catherine Bruneau, Alexis Flageollet, and Zhun Peng. (2019). VineCopula: Statistical Inference of Vine Copulas. R package version 2.0.0. https://CRAN.R-project.org/package=VineCopula