NONPARAMETRIC ESTIMATION OF COPULAS FOR TIME SERIES

Anthony YAZDANI

May 15, 2020 - GENEVA - SWITZERLAND

Abstract

In this report we will discuss a paper entitled "Nonparametric Estimation of Copulas for Time Series" published in November 2002 by Jean-David FERMANIAN, professor at ENSAE and Olivier SCAILLET, professor at the University of Geneva. A non-parametric smoothing technique is employed to estimate the bivariate copula of two stationary time series, known as Kernel Density Estimation (KDE). This report summarizes the original paper and elaborates some extensions. We will prove the link between copulas and some rank-based dependency measures, namely, Kendall's tau, Spearman's rho and Blomqvist's beta. Moreover, we will intuitively approach KDE and prove the relevance of this method through Monte Carlo simulations. After discussing these concepts, we will analyze a data set on cryptocurrencies returns making use of the R package `KernelMVA` designed according to the notions discussed here. In this last section we will also propose a new measure of dependence called the Kernel Copula Dependence.

Introduction

The scientific literature agrees on the fact that the Gaussian copula approach was a standard in the financial industry until the global financial crisis. They used this tool to link CDO's marginal default distribution to their joint default distribution (Watts 2016). The problem is that the Gaussian copula is tail independent, meaning that they largely underestimated the probability of observing individual rare events simultaneously. Furthermore, Pearson's correlation coefficient has never been suitable due to non-linear relationships between financial assets. KDE has the advantage to provide a differentiable estimation without assuming any form of dependence between margins and without losing the usual parametric rate of convergence (Scaillet and Fermanian 2002). Once the estimates of copulas are available, concepts such as positive quadrant dependence, Kernel Copula Dependence, Kendall's tau, Spearman's rho, Gini's gamma and Blomqvist's beta can be used.

To fulfill our goal, we will introduce bivariate copulas, dependency measures, KDE and the concept of time series stationarity. In a second step, we will evaluate our method through Monte Carlo simulations and will empirically introduce the package `KernelMVA`. In order to bring this illustration up to date, we will consider the Ethereum and Bitcoin couple.

I - Bivariate copulas

1 - Definition

Copulas are multivariate cumulative density functions (CDF) usually denoted $C(u_1, \ldots, u_d)$, used to describe the dependence between random variables. The distribution of each variable $u_i = F(x_i)$ for $i = 1, \ldots, d$, is uniform on the interval $[0,1]$, i.e. $C : [0,1]^d \to [0,1]$, where d is the dimension of the copula. As we are interested in the bivariate case, we can recall some properties of copulas in the following way:

$C(0,u) = C(u,0) = 0$.

$C(1,u) = C(u,1) = u$.

$C(u_2,v_2) - C(u_2,v_1) - C(u_1,v_2) + C(u_1,v_1) \geq 0$,

$\forall \{u_1, u_2, v_1, v_2\}$ that satisfies $0 \leq u_1 \leq u_2 \leq 1$ and $0 \leq v_1 \leq v_2 \leq 1$.

Moreover, copulas are invariant to any strictly monotonically increasing transformation T_i. We have $F_{T_i(X_i)}(T_i(x_i)) = F_{X_i}(x_i)$, implying that $C\left(F_{T_1(X_1)}(T_1(x_1)), F_{T_2(X_2)}(T_2(x_2))\right) = C(F_{X_1}(x_1), F_{X_2}(x_2))$.

2 - Sklar's Theorem

Let $J_{X,Y}$ be a joint distribution function with margins $F(x)$ and $G(y)$. According to Sklar's Theorem there exists a copula, such that for all (x,y) in $\mathbb{R}^2 : J_{X,Y}(x,y) = C(F(x), G(y))$. Where C is unique if all margins are continuous, otherwise C is unique on $\text{ran}(F(x)) \times \text{ran}(G(y))$. This implies that we can recover $C(u_1,u_2)$ with the joint distribution $J_{X,Y}$ and its margins $F(x), G(y)$:

$$C(u_1,u_2) = J_{X,Y}\left(F^{-1}(u_1), G^{-1}(u_2)\right). \tag{1}$$

More generally, we should consider the generalized inverse $F^{\leftarrow}$ instead of F^{-1}. However, we will deal with continuous random variables. This is why we will opt for the following writing:

$$F^{\leftarrow}(u) = \inf\{x \in \mathbb{S} : F(x) \geq u\} = F^{-1}(u), \text{ if } X \text{ is a continuous random variable.}$$

Where $\mathbb{S}$ is the support on which the CDF is defined.

3 - Fréchet–Hoeffding copula bounds

M. Fréchet and W. Hoeffding state that for any copula C we have $C^-(u_1,u_2) \leq C(u_1,u_2) \leq C^+(u_1,u_2)$, where the counter-monotonic copula C^-, the independent copula $C^\perp$ and the comonotonic copula C^+ are define as:

$$\begin{aligned} C^-(u_1,u_2) &= P(F(x) \leq u_1, 1 - F(x) \leq u_2) \\ &= P(F(x) \leq u_1, F(x) \geq 1 - u_2) = P(1 - u_2 \leq F(x) \leq u_1) = \max(0, u_1 + u_2 - 1). \end{aligned}$$

$$\begin{aligned} C^\perp(u_1,u_2) &= P(F(x) \leq u_1, G(y) \leq u_2) \\ &= P(F(x) \leq u_1) \times P(G(y) \leq u_2) = u_1 \times u_2. \end{aligned}$$

$$\begin{aligned} C^+(u_1,u_2) &= P(F(x) \leq u_1, F(x) \leq u_2) \\ &= P(F(x) \leq u_2 | F(x) \leq u_1) P(F(x) \leq u_1) = 1 \times u_1 = \min(u_i)_{i=1,2}, \text{ where } u_1 \leq u_2. \end{aligned}$$

II - Rank-based dependency measures

The Pearson correlation coefficient is used to describe linear relationships between random variables. In general, financial assets are non-linearly related to each other and therefore we will discuss alternative measures of dependency. Rank-based measures quantify the degree of dependence of two random variables on the principle of concordance and discordance making them invariant to any strictly monotonically increasing transformations. These measures can be expressed as functions of copulas. Note that equations (2) to (4) will be proven in the appendix.

1 - Kendall's tau

Let K denote the difference between the probabilities of concordance and discordance of (X_1, Y_1) and (X_2, Y_2):

$$K(C,C) = P\left[(X_1 - X_2)(Y_1 - Y_2) > 0\right] - P\left[(X_1 - X_2)(Y_1 - Y_2) < 0\right] = 4\int_0^1 \int_0^1 C(u_1, u_2) dC(u_1, u_2) - 1. \tag{2}$$

2 - Spearman's rho

Spearman's rho is defined as the Pearson correlation coefficient on ranked data:

$$\rho_S(X,Y) = \rho\left(F(x), G(y)\right) = 12\int_0^1 \int_0^1 C(u_1, u_2) du_1 du_2 - 3. \tag{3}$$

3 - Blomqvist's beta

Let's define β as the probability of concordance minus the probability of discordance of a random vector (X, Y) and there medians $(\tilde{X}, \tilde{Y})$:

$$\beta = P[(X - \tilde{X})(Y - \tilde{Y}) > 0] - P[(X - \tilde{X})(Y - \tilde{Y}) < 0] = 4C\left(\frac{1}{2}, \frac{1}{2}\right) - 1. \tag{4}$$

4 - Gini's gamma

As an extension of Kendall's tau, we can define $\gamma = K\left(C, C^+\right) + K\left(C, C^-\right)$. A concordance and discordance measure between C, the comonotonic copula C^+ and the counter-monotonic copula C^-:

$$\gamma = K\left(C, C^+\right) + K\left(C, C^-\right) = 4\int_0^1 [C(u, 1-u) + C(u, u)] du - 2. \tag{5}$$

5 - Quadrant dependence

Positive quadrant dependence (Lehmann 1966) is defined as $PQD = J_{X,Y}(x, y) \geq F(x)G(y)$ for all $(x, y) \in R^2$. We can rewrite PQD using copulas:

$$PQD = C\left(u_1, u_2\right) \geq u_1 u_2 = \frac{C(u_1, u_2)}{u_2} \geq u_1. \tag{6}$$

III – Kernel density estimation

The density function $f(x)$ is generally not known and has to be estimated. Parametric estimation is not suitable when the density has an unusual shape, particularly when the underlying distribution is multimodal. In the multivariate framework, it can be difficult to construct a density following the dependency structure suggested by the data, which is not an issue with KDE. However, KDE have a cost. This method is time-consuming, particularly with large data sets. Moreover, KDE requires an estimate of the bandwidth (h). We will get rid of that constraint using the well-known rule of thumb $h = 1.06\hat{\sigma}n^{-1/5}$ for a Gaussian Kernel (Turlach 1993). To support our choice, we will show how the KDE estimation was constructed and discuss its properties. The density estimation $\hat{f}(x)$ will be of the form $\hat{f}(x) = k/nh$. Where k is the total number of observations $x_{i=1,2,\dots,n}$ of a given data set that falls into an interval $I_{j=0,1,\dots,p}$ of length h. Evidence will be shown in the appendix.

1 - Histogram

Informally, the Gaussian KDE can be seen as changes in the way of computing k starting from the way it is computed for the histogram.

Let $k = \#\{x_i \in I_j\} = \#\{x_i \in [x_0 + jh\ ,\ x_0 + (j+1)h]\}$, where x_0 is the interval's origin and h the interval's length. In this case, the shape of the density estimate highly depends on the interval's origin and the estimation is not differentiable. Therefore, the histogram is not an optimal solution but is a first step in introducing Gaussian KDE.

For the sake of consistency, we will define k as a function of K, the kernel function:

$$
\begin{aligned}
k &= \#\left\{x_i \in [x_0 + jh, x_0 + (j+1)h]\right\} = \sum_{i=1}^{n} \mathbf{1}\left[x_i \in [x_0 + jh, x_0 + (j+1)h]\right] \\
&= \sum_{i=1}^{n} \mathbf{1}\left[j \leq \frac{x_i - x_0}{h} \leq j+1\right] \\
&= \sum_{i=1}^{n} K\left(\frac{x_i - x_0}{h}, j\right).
\end{aligned}
$$

Where $K\left(\frac{x_i - x_0}{h}, j\right) = 1 \quad \text{if} \quad j \leq \frac{x_i - x_0}{h} \leq j+1 \quad \text{and} \quad K\left(\frac{x_i - x_0}{h}, j\right) = 0 \quad \text{otherwise.}$

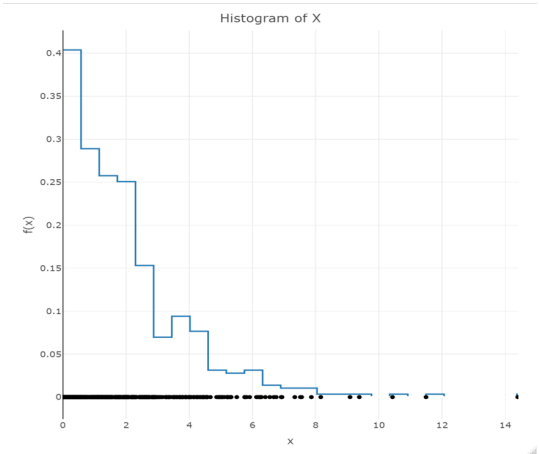

Figure 1 : Histogram of $X \sim \chi^2_2$ with 500 Observations.

2 - Parzen's windows

Parzen's windows method is very similar to the histogram. The idea of Emanuel Parzen was to consider I_j centered in x of length h. Thus, we can define k as follows:

$$k = \sum_{i=1}^{n} \mathbf{1}\left[x - 0,5h \leq x_i \leq x + 0,5h\right] = \sum_{i=1}^{n} \mathbf{1}\left[-0.5 \leq \frac{x_i - x}{h} \leq 0.5\right] = \sum_{i=1}^{n} K\left(\frac{x_i - x}{h}\right).$$

$$\text{Where } K\left(\frac{x_i - x}{h}\right) = 1 \quad \text{if} \quad \left|\frac{(x_i - x)}{h}\right| \leq \frac{1}{2} \quad \text{and} \quad K\left(\frac{x_i - x}{h}\right) = 0 \quad \text{otherwise.}$$

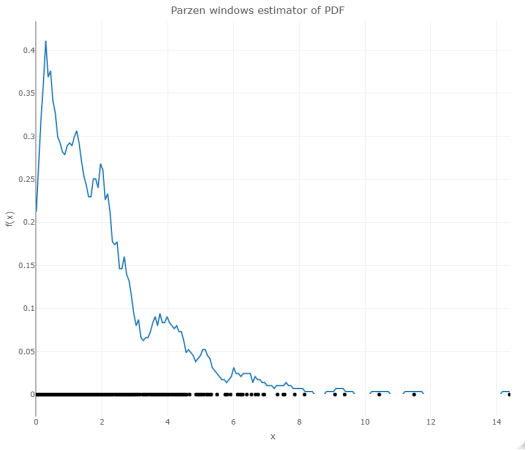

Figure 2 : Parzen's Windows PDF Estimation of $X \sim \chi^2_2$ with 500 Observations.

In fact, Parzen's estimator have many drawbacks. The density estimate is not sufficiently smooth (i.e. non-differentiable), which is a necessary condition to make use of rank-based dependency measures previously mentioned. Secondly, all the data points within an interval are equally weighted regardless their distance to x. Most relevant is that we have considered I_j centered in x of length h, which is a second step in introducing Gaussian KDE.

3 - Gaussian kernel

Gaussian kernel is a more sophisticated kernel function which allows us to get a differentiable estimation. Because of the peculiar shape of the Gaussian distribution, observations relatively far from a point x contributes significantly less than relatively close neighbors. More generally, the kernel function must be a continuous probability density function with mean zero and finite variance.

$$\int_{\mathbb{S}} K(x)dx = 1, \qquad \int_{\mathbb{S}} xK(x)dx = 0, \qquad \int_{\mathbb{S}} x^2K(x)dx < \infty, \qquad K(x) \geq 0, \qquad \forall\, x \in \mathbb{S}.$$

Where $\mathbb{S}$ is the support on which the kernel function is defined. In this case, the kernel is defined as follows:

$$K\left(\frac{x_i - x}{h}\right) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x_i - x}{h}\right)^2\right), \quad \forall\, x \in \mathbb{R}, \quad \sigma = 1, \quad \mu = 0 \text{ and } h = 1.06\hat{\sigma}n^{-1/5}. \tag{7}$$

When choosing a value for h, a compromise has to be made. Informally, the bandwidth h has to be large enough to include a sufficient number of observations within I_j and, at the same time, be small enough to satisfy $f(x) \cong k/nh$. When considering a Gaussian kernel, the rule of thumb $h = 1.06\hat{\sigma}n^{-1/5}$ is found based on minimization of the asymptotic mean squared error (Turlach 1993). One major drawback of Gaussian KDE is that the symmetry of the kernel can lead to boundary bias, which is due to weight allocation by the kernel outside the support of the distribution.

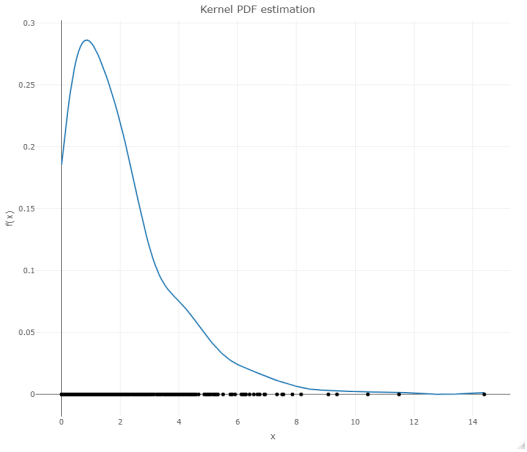

Figure 3 : Gaussian Kernel PDF Estimation of $X \sim \chi^2_2$ with 500 Observations.

4 - Kernel CDF estimation

We can compute all the tools we will use regarding Kernel CDF estimation. Note that Φ will stand for the Gaussian CDF.

Univariate CDF:

$$\begin{aligned}\hat{F}(x) &= \int_{-\infty}^{x} \hat{f}(x)dx \\ &= \int_{-\infty}^{x} \frac{1}{nh}\sum_{i=1}^{n} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x_i - x}{h}\right)^2\right) dx \\ &= \frac{1}{n}\sum_{i=1}^{n} \Phi\left(\frac{x_i - x}{h}\right), \forall x \in \mathbb{R}.\end{aligned} \tag{8}$$

Bivariate CDF:

$$\begin{aligned}\hat{J}(x,y) &= \int_{-\infty}^{x}\int_{-\infty}^{y} \hat{f}(x)\hat{g}(y)dxdy \\ &= \frac{1}{n}\sum_{i=1}^{n} \Phi\left(\frac{x_i - x}{h_x}\right)\Phi\left(\frac{y_i - y}{h_y}\right), \forall (x,y) \in \mathbb{R}^2.\end{aligned} \tag{9}$$

Using equation (1), the copula CDF estimation can be expressed as:

$$\hat{C}\left(u_1, u_2\right) = \hat{J}\left(\hat{F}^{-1}\left(u_1\right), \hat{G}^{-1}\left(u_2\right)\right), \forall \left(u_1, u_2\right) \in [0,1]^2. \tag{10}$$

IV – Weak stationarity

The process $\boldsymbol{Y}$ is said to be weakly stationary if, for any integer $s \geq 1$, and $t_1 \leq t_2 \leq \ldots \leq t_s$, and all integer k, all the joint moments of order one and two of $\{Y_{t_1}, Y_{t_2}, \ldots, Y_{t_s}\}$ exist, are finite, and equal to the corresponding joint moments of $\{Y_{t_1+k}, Y_{t_2+k}, \ldots, Y_{t_s+k}\}$ (La Vecchia 2019).

One can derive the asymptotic distribution of kernel estimators for copulas under some conditions on the kernel, on the properties of the process and on the asymptotic behaviour of the bandwidth. In fact, it can be shown that $\sqrt{n}(\hat{C} - C)$ tends to a centered Gaussian process (Scaillet and Fermanian 2002).

V – Monte Carlo simulation

This section will be dedicated to the evaluation of KDE estimation by means of Monte Carlo simulations. We will use the bias and the mean squared error (MSE) as a criterion to evaluate several kernel estimates $\hat{C}(u_1, u_2)$ in the tails and in the center of the distribution. We will consider stationary vector autoregressive processes with Gaussian innovations composed of two components (VAR_2):

$$\boldsymbol{Y}_t = \boldsymbol{c} + \boldsymbol{\phi}\boldsymbol{Y}_{t-1} + \boldsymbol{\epsilon}_t \quad \Leftrightarrow \begin{bmatrix} Y_{1,t} \\ Y_{2,t} \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} + \begin{bmatrix} \phi_{1,1} & \phi_{1,2} \\ \phi_{2,1} & \phi_{2,2} \end{bmatrix} \begin{bmatrix} Y_{1,t-1} \\ Y_{2,t-1} \end{bmatrix} + \begin{bmatrix} \epsilon_{1,t} \\ \epsilon_{2,t} \end{bmatrix}. \tag{11}$$

Where $\boldsymbol{\epsilon}_t \sim N(0, \Sigma)$.

Multiple cases will be studied here as whether or not the components are independent and how it changes with the sample size. Bias and MSE will be computed on $m = 10000$ random samples of length $n = 1000$ and $n = 10$. Note that all figures in tables are divided by 10^4.

1 - Independent components

Consider the following stationary $\text{VAR}_2^{\perp}$ model:

$$\begin{bmatrix} Y_{1,t} \\ Y_{2,t} \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 0.25 & 0 \\ 0 & 0.75 \end{bmatrix} \begin{bmatrix} Y_{1,t-1} \\ Y_{2,t-1} \end{bmatrix} + \begin{bmatrix} \epsilon_{1,t} \\ \epsilon_{2,t} \end{bmatrix}, \text{ where } \Sigma = \begin{bmatrix} 0.75 & 0 \\ 0 & 1.25 \end{bmatrix}.$$

That can be written as $Y_t \sim N(\boldsymbol{\mu}\ ,\ \Omega) = N(\begin{bmatrix} 1.\bar{3} \\ 4 \end{bmatrix}\ ,\ \begin{bmatrix} 0.8 & 0 \\ 0 & 2.86 \end{bmatrix})$ since $\boldsymbol{\mu} = (I - \boldsymbol{\phi})^{-1}\boldsymbol{c}$ and $\Omega = (I - \boldsymbol{\phi} \otimes \boldsymbol{\phi})^{-1}\Sigma$.

True values for $C(u_1, u_2)$ are easy to compute since we know that we are dealing with independent random variables i.e. $C(u_1, u_2) = u_1 u_2$.

	C(0.01,0.01)	C(0.05,0.05)	C(0.25,0.25)	C(0.5,0.5)	C(0.75,0.75)	C(0.95,0.95)	C(0.99,0.99)
True	1.00000	25.00000	625.00000	2500.00000	5625.00000	9.025e+03	9.801e+03
Bias	0.00000	0.00000	0.00003	0.00001	0.00000	0.000e+00	0.000e+00
MSE	0.00035	0.01094	0.19986	0.37172	0.19858	1.047e-02	3.800e-04

Table 1 : Bias and MSE of $\hat{C}(u_1, u_2)$ from $\text{VAR}_2^{\perp}$ Simulations when $n = 1000$.

It appears from `Table 1` that bias and MSE are satisfactory.

	C(0.01,0.01)	C(0.05,0.05)	C(0.25,0.25)	C(0.5,0.5)	C(0.75,0.75)	C(0.95,0.95)	C(0.99,0.99)
True	1.00000	25.00000	625.00000	2500.00000	5625.00000	9025.00000	9.801e+03
Bias	0.00000	0.00000	0.00014	0.00025	0.00000	0.00000	0.000e+00
MSE	0.00031	0.11402	7.58727	16.38343	7.76247	0.11452	3.100e-04

Table 2 : Bias and MSE of $\hat{C}(u_1, u_2)$ from $\text{VAR}_2^{\perp}$ Simulations when $n = 10$.

We can notice from `Table 2` that the MSE is larger than in the previous simulation, which is exactly what we would expect by imposing smaller samples.

2 - Dependent components

Consider the following stationary VAR_2^{+} model:

$$\begin{bmatrix} Y_{1,t} \\ Y_{2,t} \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 0.25 & 0.2 \\ 0.2 & 0.75 \end{bmatrix} \begin{bmatrix} Y_{1,t-1} \\ Y_{2,t-1} \end{bmatrix} + \begin{bmatrix} \epsilon_{1,t} \\ \epsilon_{2,t} \end{bmatrix}, \text{ where } \Sigma = \begin{bmatrix} 0.75 & 0.5 \\ 0.5 & 1.25 \end{bmatrix}.$$

We can find $\boldsymbol{\mu}$ in the same way as before i.e. $\boldsymbol{\mu} = (I - \boldsymbol{\phi})^{-1}\boldsymbol{c}$. However, for the covariance it is more tedious. We know that $Cov(\boldsymbol{Y}_t) = \boldsymbol{\phi} Cov(\boldsymbol{Y}_{t-1})\boldsymbol{\phi}^T + Cov(\boldsymbol{\epsilon}_t) = \boldsymbol{\phi} Cov(\boldsymbol{Y}_t)\boldsymbol{\phi} + Cov(\boldsymbol{\epsilon}_t)$, because $\boldsymbol{Y}_t$ is stationary and $\boldsymbol{\phi}$ symmetric. Hence, we can derive the following:

$$\begin{aligned} \text{Cov}\left(\boldsymbol{Y}_t\right) &= \begin{bmatrix} \phi_1 & \phi_2 \\ \phi_3 & \phi_4 \end{bmatrix} \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix} \begin{bmatrix} \phi_1 & \phi_2 \\ \phi_3 & \phi_4 \end{bmatrix} + \Sigma \\ &= \begin{bmatrix} \phi_1^2 a_1 + \phi_1\phi_2 a_3 + \phi_1\phi_3 a_2 + \phi_2\phi_3 a_4 & \phi_1\phi_2 a_1 + \phi_2^2 a_3 + \phi_1\phi_4 a_2 + \phi_2\phi_4 a_4 \\ \phi_3\phi_1 a_1 + \phi_1\phi_4 a_3 + \phi_3^2 a_2 + \phi_3\phi_4 a_4 & \phi_3\phi_2 a_1 + \phi_4\phi_2 a_3 + \phi_3\phi_4 a_2 + \phi_4^2 a_4 \end{bmatrix} + \Sigma. \end{aligned}$$

Which leads to the following linear system of equation:

$$\begin{cases} a_1 = \phi_1^2 a_1 + \phi_1\phi_2 a_3 + \phi_1\phi_3 a_2 + \phi_2\phi_3 a_4 + \sum_{1,1} \\ a_2 = \phi_1\phi_2 a_1 + \phi_2^2 a_3 + \phi_1\phi_4 a_2 + \phi_2\phi_4 a_4 + \sum_{1,2}^2 \\ a_3 = \phi_3\phi_1 a_1 + \phi_1\phi_4 a_3 + \phi_3^2 a_2 + \phi_3\phi_4 a_4 + \sum_{2,1}^2 \\ a_4 = \phi_3\phi_2 a_1 + \phi_4\phi_2 a_3 + \phi_3\phi_4 a_2 + \phi_4^2 a_4 + \sum_{2,2}^2 \end{cases} \Leftrightarrow \begin{cases} \sum_{1,1} = \left(1 - \phi_1^2\right) a_1 - \phi_1\phi_2 a_3 - \phi_1\phi_3 a_2 - \phi_2\phi_3 a_4 \\ \sum_{1,2} = -\phi_1\phi_2 a_1 - \phi_2^2 a_3 + \left(1 - \phi_1\phi_4\right) a_2 - \phi_2\phi a_4 \\ \sum_{2,1} = -\phi_3\phi a_1 + \left(1 - \phi_1\phi_4\right) a_3 - \phi_3^2 a_2 - \phi_3\phi a_4 \\ \sum_{2,2} = -\phi_3\phi_2 a_1 - \phi_4\phi_2 a_3 - \phi_3\phi_4 a_2 + \left(1 - \phi_4^2\right) a_4. \end{cases}$$

Meaning that we can easily recover $\text{Cov}(\boldsymbol{Y}_t)$ with a simple matrix multiplication.

$$\text{vec}\left(\text{Cov}\left(\boldsymbol{Y}_t\right)\right) = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} = \begin{bmatrix} 1 - \phi_1^2 & -\phi_1\phi_3 & -\phi_1\phi_2 & -\phi_2\phi_3 \\ -\phi_1\phi_2 & 1 - \phi_1\phi_4 & -\phi_2^2 & -\phi_2\phi_4 \\ -\phi_3\phi_1 & -\phi_3^2 & 1 - \phi_1\phi_4 & -\phi_3\phi_4 \\ -\phi_3\phi_1 & -\phi_3\phi_4 & -\phi_4\phi_2 & 1 - \phi_4^2 \end{bmatrix}^{-1} \begin{bmatrix} \sum_{1,1} \\ \sum_{1,2} \\ \sum_{2,1} \\ \sum_{2,2} \end{bmatrix}$$

Thus, $Y_t \sim N(\begin{bmatrix} 3.05 \\ 4 \end{bmatrix}, \begin{bmatrix} 1.13 & 1.5 \\ 1.5 & 4.01 \end{bmatrix})$.

Since we are dealing with dependent random variables, true values of $C(u_1, u_2)$ can be computed thanks to equation (1), i.e. $C(u_1, u_2) = J\left(F^{-1}(u_1), G^{-1}(u_2)\right) = \Phi_\Sigma\left(\Phi^{-1}(u_1), \Phi^{-1}(u_2)\right)$, where Σ is the covariance matrix.

	C(0.01,0.01)	C(0.05,0.05)	C(0.25,0.25)	C(0.5,0.5)	C(0.75,0.75)	C(0.95,0.95)	C(0.99,0.99)
True	27.04184	197.74823	1510.98229	3742.68798	6510.98229	9197.74823	9827.04184
Bias	0.00194	0.04309	0.61177	1.04730	0.61092	0.04410	0.00216
MSE	0.00867	0.08438	0.78869	1.28124	0.78756	0.08521	0.00892

Table 3 : Bias and MSE of $\hat{C}(u_1, u_2)$ from VAR_2^+ Simulations when $n = 1000$.

It appears from `Table 3` that bias and MSE are also satisfactory.

	C(0.01,0.01)	C(0.05,0.05)	C(0.25,0.25)	C(0.5,0.5)	C(0.75,0.75)	C(0.95,0.95)	C(0.99,0.99)
True	27.04184	197.74823	1510.98229	3742.68798	6510.98229	9197.74823	9827.04184
Bias	0.05485	1.40227	13.48800	21.83176	13.36759	1.37909	0.05460
MSE	0.05565	1.68248	19.60782	29.95329	19.52593	1.66366	0.05542

Table 4 : Bias and MSE of $\hat{C}(u_1, u_2)$ from VAR_2^+ Simulations when $n = 10$.

Similar results can be seen from `Table 4`. Bias and MSE increase when the sample size decrease.

When it comes to vector autoregressive models with Gaussian innovations, we can conclude that it is a satisfactory estimation method in terms of bias and MSE.

VI – Analysis of Ethereum and Bitcoin couple

This section will be dedicated to the application of the methods we have seen. We will use the statistical programming software R to illustrate our point. An R package has been developed to make this analysis reproducible by anyone. The purpose is to make a quick demonstration of the `KernelMVA` package, therefore, some functions will not be shown here. In the following, we will focus on the dependency structure of Bitcoin and Ethereum couple before and after COVID-19 pandemic.

1 - KernelMVA package

To install this package you can simply enter the following lines of code on your Rstudio development environment. If you are invited to do any updates, please ignore them.

```
install.packages("devtools")
require(devtools)
devtools::install_github("anthonyyazdani/KernelMVA2")
require(KernelMVA)
```

2 - Data analysis

2.a - Exploratory data analysis

We will use the built-in data set called `Bitcoin.Ethereum` which contains Bitcoin's and Ethereum's returns from 2017-08-19 to 2020-04-23. However, we will split the data set into two equal parts composed of 83 observations: the before COVID-19 sample from 2019-11-10 to 2020-01-31 and the after COVID-19 sample from 2020-02-01 to 2020-04-23.

Bitcoin_returns	Ethereum_returns
Min. :-0.058072	Min. :-0.0817449
1st Qu.:-0.015233	1st Qu.:-0.0177254
Median :-0.001653	Median :-0.0011165
Mean : 0.001380	Mean : 0.0006843
3rd Qu.: 0.010004	3rd Qu.: 0.0148280
Max. : 0.098736	Max. : 0.1536426

Table 5 : Summary of Bitcoin and Ethereum Couple Before COVID-19.

Bitcoin_returns	Ethereum_returns
Min. :-0.3950485	Min. :-0.445969
1st Qu.:-0.0178634	1st Qu.:-0.027896
Median : 0.0009702	Median : 0.002762
Mean :-0.0013381	Mean : 0.003531
3rd Qu.: 0.0205027	3rd Qu.: 0.037983
Max. : 0.1622083	Max. : 0.243369

Table 6 : Summary of Bitcoin and Ethereum Couple After COVID-19.

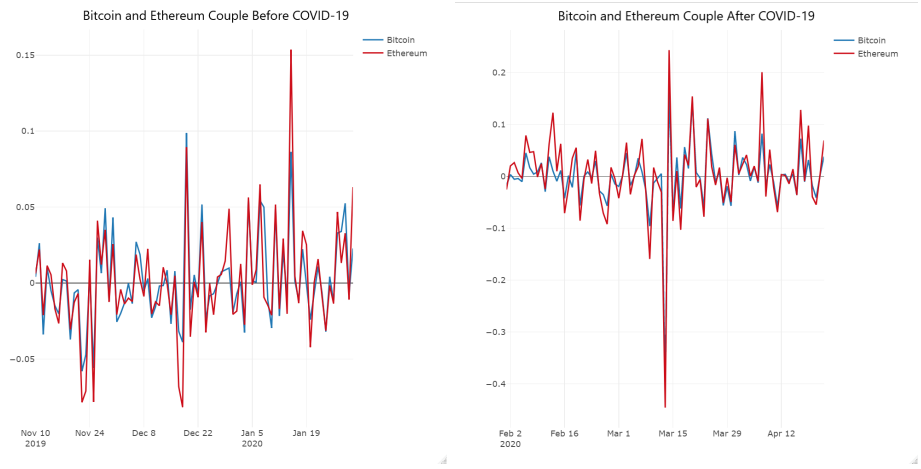

Figure 4 : Plot of Bitcoin's and Ethereum's Returns Before and After COVID-19.

We can observe from `Table 5` and `Table 6` that the volatility increased. In fact, standard deviation increased from 0.03 to 0.06 for the Bitcoin and from 0.035 to 0.08 for the Ethereum. One can therefore speculate that this is a market reaction to the COVID-19 pandemic. Moreover, we observe from `Figure 4` an extreme loss of -40% on March 13, the day Donald Trump declared the state of emergency in the United States. Since this is not the main subject of our analysis, we will now turn our attention on dependency structures.

2.b - KernelMVA functions

In this sub-section we will use the functions provided by the `KernelMVA` package. First of all, it may be interesting to see the graphical representation of the joint PDFs. For this we can use the function `bi.pdf.plot()`.

```
beforepdf <- bi.pdf.plot(beforeCOVID,100)
afterpdf <- bi.pdf.plot(afterCOVID,100)
beforepdf$Contour
afterpdf$Contour
```

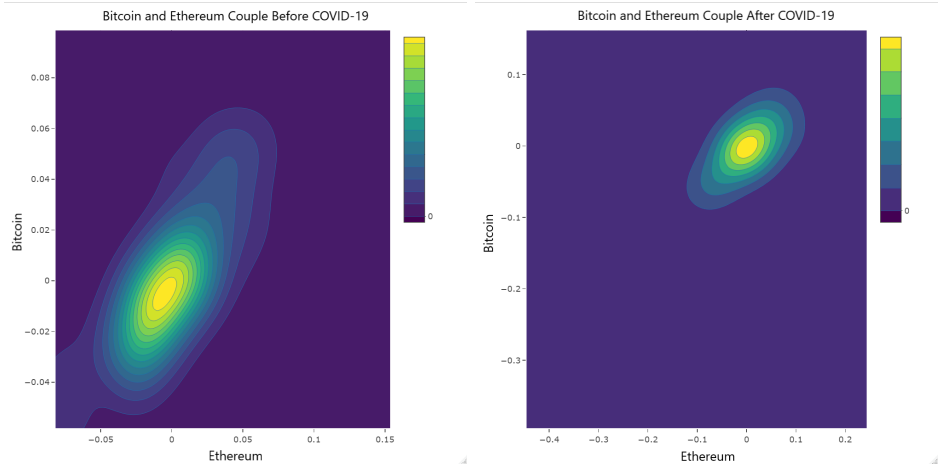

Figure 5 : PDF Plots of Bitcoin and Ethereum before and after COVID-19.

As we can see from `Figure 5`, Bitcoin and Ethereum exhibit linear dependence. However, to appropriately understand their dependence, we will look at the copula's PDFs for a visual understanding of their dependency structure and use ranked based measures. We can use the function `copula.pdf.plot()` with the argument `sqrt = TRUE` to estimate the copula on the square root scale. This argument is used in order to make the PDFs easier to interpret. To compare, we will also show how the comonotonic copula C^+ looks like when it is estimated by Gaussian KDE in a large sample ($n = 1000$) and in a sample of size $n = 83$.

```
beforeCOPpdf = copula.pdf.plot(beforeCOVID, 200, sqrt = T)
afterCOPpdf = copula.pdf.plot(afterCOVID, 200, sqrt = T)
beforeCOPpdf$Plot3D
afterCOPpdf$Plot3D
```

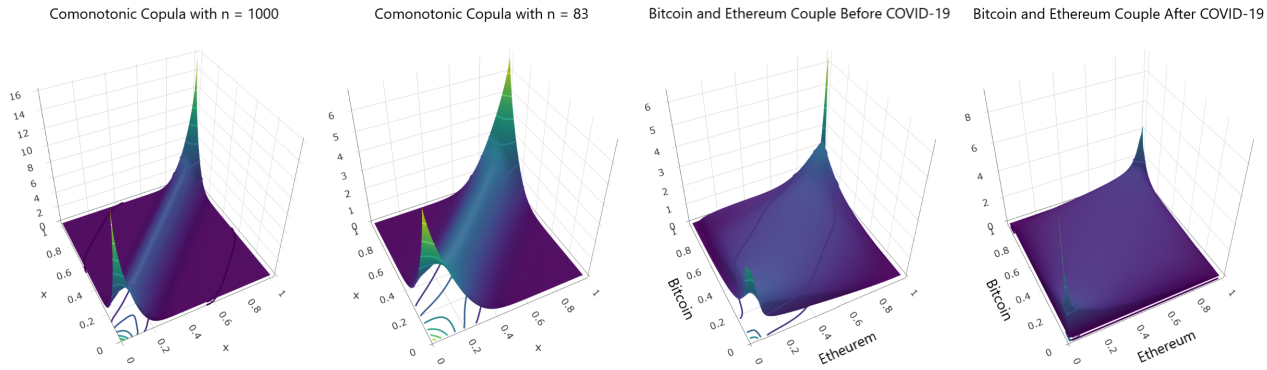

Figure 6 : PDF Plots of the comonotonic copula C^+ and the Copula before and after COVID-19.

As we can see from `Figure 6`, even if we have two comonotonic random variables, Gaussian KDE introduces variability. This is why mentioned dependency measures computed by KDE will tend to be underestimated in small sample. This effect vanishes as the sample size increase. Qualitatively, the copula PDF estimation before COVID-19 looks closer to the comonotonic copula C^+ than after COVID-19. Visual understanding is useful, but we need to quantify this dependence. We can use the functions `spearmans.rho()`, `ginis.gamma()` and `blomqvists.beta()`.

	Rho	Gamma	Beta		Rho	Gamma	Beta
Before COVID-19	0.69	0.56	0.48	Comonotonic Copula n = 1000	0.94	0.83	0.83
After COVID-19	0.60	0.47	0.38	Comonotonic Copula n = 83	0.85	0.72	0.71

Table 6 : Spearman's Rho, Gini's Gamma and Blomqvist's Beta.

We can see from `Table 6` that our intuition is confirmed. However, to get rid of underestimation in small sample, we can suggest a new measure, the Kernel Copula Dependence (KCD).

Proposition: *Let $\hat{C}_n$ denote the usual copula KDE and $\hat{C}_n^+$, $\hat{C}_n^-$, denote the estimated comonotonicity and counter-monotonicicity copula. The subscript n indicates that $\hat{C}_n$, $\hat{C}_n^+$ and $\hat{C}_n^-$ are all estimated on the same sample size n. The Kernel Copula Dependence can be expressed as follows:*

$$KCD_n(\hat{C}) = 1 - \frac{2\int_{u_1}\int_{u_2} \hat{C}_n^+ - \hat{C}_n \, du_1 du_2}{\int_{u_1}\int_{u_2} \hat{C}_n^+ - \hat{C}_n^- \, du_1 du_2} = 1 - \frac{2\theta(\hat{C}_n)}{\theta(\hat{C}_n^-)}. \tag{12}$$

Where $\theta(\hat{A}_n)$ is a dissimilarity measure between $\hat{C}_n^+$ and any copula $\hat{A}_n$.

- *We can show that KCD ranges from -1 to 1 in the following way:*

$$\begin{aligned}
\hat{C}_n^- \leq \hat{C}_n \leq \hat{C}_n^+ &\iff \theta\left(\hat{C}_n^-\right) \geq \theta\left(\hat{C}_n\right) \geq \theta\left(\hat{C}_n^+\right) \\
&\iff \theta\left(\hat{C}_n^-\right) \geq \theta\left(\hat{C}_n\right) \geq 0, \text{ where } \theta\left(\hat{C}_n^-\right) \neq 0. \\
&\iff 1 \geq \frac{\theta\left(\hat{C}_n\right)}{\theta\left(\hat{C}_n^-\right)} \geq 0 \\
&\iff -1 \leq 1 - \frac{2\times\theta\left(\hat{C}_n\right)}{\theta\left(C_n^-\right)} \leq 1 = -1 \leq KCD_n(\hat{C}) \leq 1.
\end{aligned}$$

- *We can show that $\int_{u_1}\int_{u_2} C^+ - C^- \, du_1 du_2 \neq 0$ in the following way:*

$$\begin{aligned}
\int_{u_1}\int_{u_2} C^+ - C^- \, du_1 du_2 &= \int_0^1\int_0^1 \min(u_1, u_2)\, du_1 du_2 - \int_0^1\int_0^1 \max(0, u_1 + u_2 - 1) \\
&= 2\int_{0\leq u_1\leq u_2\leq 1} u_1 du_2 du_1 - \int_{(u_1+u_2-1)\geq 0} (u_1 + u_2 - 1)\, du_1 du_2 \\
&= 2\int_0^1 \frac{u^2}{2} du - \int_0^1\int_{1-u_2}^1 (u_1 + u_2 - 1)\, du_1 du_2 \\
&= \frac{1}{3} - \frac{1}{6} = \frac{1}{6}.
\end{aligned}$$

- *$\theta(\hat{C}_n^-)$ is a scaling factor in finite sample. Moreover, $\lim_{n\to\infty}\theta(\hat{C}_n^-) = \int_{u_1}\int_{u_2} C^+ - C^- \, du_1 du_2 = \frac{1}{6}$, because of the asymptotic properties of KDE.*

Thus, if $KCD_n(\hat{C})$ approaches 1 it means that there is a strong positive dependence and, analogously, if $KCD_n(\hat{C})$ approaches -1 it suggests that there is a strong negative dependence. For this, we can use the function `KCD()`.

	KCD			KCD
Before COVID-19	0.81	|	Comonotonic Copula n = 1000	1
After COVID-19	0.70	|	Comonotonic Copula n = 83	1
		|	Counter-momonotonic Copula n = 1000	-1
		|	Counter-momonotonic Copula n = 83	-1

Table 7 : Kernel Copula Dependence.

Moreover, large negative losses are of particular interest in risk management. For this we can use a common measure called quadrant dependence, $QD = C(u_1, u_2) - u_1 u_2$ (Lehmann 1966) in the lower tail of the distribution. This quantity can be either positive or negative and therefore can be labelled PQD or NQD. Intuitively, $|QD|$ measures how far we are from $C^{\perp}$ in terms of probabilities and sign(QD) tells us in which direction. To compute this quantity, we can use the function `quadrant.dependence()`.

	QD(.01)	QD(.02)	QD(.03)	QD(.04)	QD(.05)
Before COVID-19	0.002	0.006	0.011	0.018	0.024
After COVID-19	0.008	0.014	0.017	0.020	0.023

Table 8 : Quadrant Dependence of Large Negative Losses Before and After COVID-19.

It appears from `Table 8` that Bitcoin and Ethereum exhibit PQD for $u_1 = u_2 = 0.01, \ldots, 0.05$. More specifically, Bitcoin and Ehtereum exhibit a higher PQD for extreme losses ($QD(.01)$, $QD(.02)$) after the COVID-19 pandemic.

In this section we were able to make a hypothesis about the increase of volatility. This may indeed be due to investors uncertainty in the context of the COVID-19 pandemic. Secondly, we computed three different measures of dependence, all three suggested a decline in dependency. Thirdly, we introduced Kernel Copula Dependence (KCD), which allows us to draw the same conclusions. Most relevant, we have seen that KCD have interesting properties in small samples. Last but not least, we concluded that Bitcoin and Ethereum exhibit PQD in the lower tail in both cases. More specifically, we found that after the beginning of the pandemic, the dependency for extreme losses ($QD(.01)$, $QD(.02)$) had increased.

Conclusion

In this report we recalled some tools to understand the core of this work, the kernel copula's density estimation (KDE). After having introduced KDE from a historical perspective, we used this non-parametric technique to estimate the bivariate copula of two stationary time series. We evaluated the performance of our estimator through Monte-Carlo simulations and it has been empirically proven to be relevant. Moreover, we showed how rank-based dependency measures are related to copulas, introduced a new dependence measure and an R package to make our Bitcoin and Ehtereum analysis accessible and replicable. From this analysis we were able to suggest that the COVID-19 pandemic had an influence on the dependency structure of Bitcoin and Ethereum. To conclude, we can mention a possible way of improving KDE. In fact, The Gaussian KDE was a major breakthrough in non-parametric density estimation. However, we can mention another kernel to avoid boundary bias, the so-called Beta kernel.

Appendix

Kendall's tau

Let C_1 and C_2 denote the copulas of $V = (X_1, Y_1)$ and $W = (X_2, Y_2)$ an independent copy of V. Let K denote the difference between the probabilities of concordance and discordance of (X_1, Y_1) and (X_2, Y_2), i.e.:

$K = P\left[(X_1 - X_2)(Y_1 - Y_2) > 0\right] - P\left[(X_1 - X_2)(Y_1 - Y_2) < 0\right]$.

Since $P\left[(X_1 - X_2)(Y_1 - Y_2) < 0\right] = 1 - P\left[(X_1 - X_2)(Y_1 - Y_2) > 0\right]$ we can write:

$K = 2P\left[(X_1 - X_2)(Y_1 - Y_2) > 0\right] - 1$.

And knowing that $P\left[(X_1 - X_2)(Y_1 - Y_2) > 0\right] = P\left[X_1 > X_2, Y_1 > Y_2\right] + P\left[X_1 < X_2, Y_1 < Y_2\right]$, the first term can be written as:

$$\begin{aligned} P\left[X_1 > X_2, Y_1 > Y_2\right] &= \int_R \int_R P\left[X_2 \leq x, Y_2 \leq y\right] dC_1(F(x), G(y)) \\ &= \int_R \int_R C_2(F(x), G(y)) dC_1(F(x), G(y)) \\ &= \int_0^1 \int_0^1 C_2(u_1, u_2) dC_1(u_1, u_2). \end{aligned}$$

The second term can be written as:

$$\begin{aligned} P\left[X_1 < X_2, Y_1 < Y_2\right] &= P\left[X_2 > X_1, Y_2 > Y_1\right] \\ &= \int_R \int_R P\left[X_2 > x, Y_2 > y\right] dC_1(F(x), G(y)) \\ &= \int_R \int_R \left[1 - F(x) - G(y) + C_2(F(x), G(y))\right] dC_1(F(x), G(y)) \\ &= \int_0^1 \int_0^1 \left[1 - u_1 - u_2 + C_2(u_1, u_2)\right] dC_1(u_1, u_2) \\ &= 1 - \frac{1}{2} - \frac{1}{2} + \int_0^1 \int_0^1 C_2(u_1, u_2) dC_1(u_1, u_2) \\ &= \int_0^1 \int_0^1 C_2(u_1, u_2) dC_1(u_1, u_2). \end{aligned}$$

Thus,

$$P\left[(X_1 - X_2)(Y_1 - Y_2) > 0\right] = 2 \int_0^1 \int_0^1 C_2(u_1, u_2) dC_1(u_1, u_2).$$

Since $K = 2P\left[(X_1 - X_2)(Y_1 - Y_2) > 0\right] - 1$, the following equation holds:

$$K(C, C) = 4 \int_0^1 \int_0^1 C(u_1, u_2) dC(u_1, u_2) - 1.$$

Spearman's rho

We know that Spearman's rho is defined as the Pearson correlation coefficient on ranked data.

$$\begin{aligned}
\rho_S(X,Y) &= \rho\left(F(x), F(y)\right) \\
&= \frac{\operatorname{Cov}\left(F(x), F(y)\right)}{\sqrt{\operatorname{Var}\left(F(x)\right)}\sqrt{\operatorname{Var}\left(F(y)\right)}} \\
&= \frac{\operatorname{Cov}(u_1, u_2)}{\sqrt{\operatorname{Var}(u_1)}\sqrt{\operatorname{Var}(u_2)}} \\
&= 12\operatorname{Cov}(u_1, u_2) \\
&= 12E[(u_1 - E[u_1])(u_2 - E[u_2])] = 12E\left[\left(u_1 - \frac{1}{2}\right)\left(u_2 - \frac{1}{2}\right)\right] \\
&= 12\left(E[u_1u_2] - \frac{1}{4}\right) = 12E[u_1u_2] - 3 \\
&= 12\int_0^1\int_0^1 u_1u_2c(u_1, u_2)du_1du_2 - 3 \\
&= 12\int_0^1\int_0^1 u_1u_2dC(u_1, u_2) - 3.
\end{aligned}$$

Meaning that the following equation holds:

$$\rho_S(X,Y) = 12\int_0^1\int_0^1 C(u_1, u_2)du_1du_2 - 3.$$

Blomqvist's beta

Let's define $\beta = P[(X - \tilde{X})(Y - \tilde{Y}) > 0] - P[(X - \tilde{X})(Y - \tilde{Y}) < 0]$, the probability of concordance minus the probability of discordance of a random vector and their medians:

$$\begin{aligned}
\beta &= 2P[(X - \tilde{X})(Y - \tilde{Y}) > 0] - 1 \\
&= 2\{P[X < \tilde{X}, Y < \tilde{Y}] + P[X > \tilde{X}, Y > \tilde{Y}]\} - 1 \\
&= 2\{J(\tilde{X}, \tilde{Y}) + [1 - F(\tilde{X}) - G(\tilde{Y}) + J(\tilde{X}, \tilde{Y})]\} - 1 = 4J(\tilde{X}, \tilde{Y}) - 1.
\end{aligned}$$

Meaning that the following equation holds:

$$\beta = 4C\left(\frac{1}{2}, \frac{1}{2}\right) - 1.$$

Empirical estimator of the density

Let I_j be an interval around a point x of length h. Let $p(x) = \int_{I_j} f(x)dx$ be the probability that an observation falls into I_j.

We define the probability that k of the n observations falls into I_j as:

$$P\left(X_j = k\right) = (n!)/(k!(n-k)!) \times p(x)^k (1-p(x))^{n-k},$$
$$\text{where } E\left(X_j/n\right) = p(x) \text{ and } \operatorname{Var}\left(X_j/n\right) = p(x)(1-p(x))/n.$$

As $n \to \infty$ we have that $k \to \infty$, $h \to 0$ and $k/n \to 0$. Asymptotically, $p(x) = k/n = \int_{I_j} f(x)dx = f(x)h$, meaning that the following equation holds:

$$\lim_{n\to\infty} \frac{k}{nh} = f(x).$$

References

La Vecchia, Davide. 2019. "Weak Stationarity, Course of Statistical Analysis of Time Series - Unige."

Lehmann, Erich Leo. 1966. "Some Concepts of Dependence." *The Annals of Mathematical Statistics*, 1137–53.

Scaillet, Olivier, and Jean-David Fermanian. 2002. "Nonparametric Estimation of Copulas for Time Series." *FAME Research Paper*, no. 57.

Turlach, Berwin A. 1993. "Bandwidth Selection in Kernel Density Estimation: A Review." In *CORE and Institut de Statistique*. Citeseer.

Watts, Samuel. 2016. "The Gaussian Copula and the Financial Crisis: A Recipe for Disaster or Cooking the Books?" June.