Gaussian Process Conditional Copulas with Applications to Financial Time Series

José Miguel Hernández-Lobato
Department of Engineering
University of Cambridge

James Robert Lloyd
Department of Engineering
University of Cambridge

Daniel Hernández-Lobato
ICTEAM - Machine Learning Group
Université catholique de Louvain

Abstract

The estimation of dependencies between multiple variables is a central problem in the analysis of financial time series. A common approach is to express these dependencies in terms of a copula function. Typically the copula function is assumed to be constant but this may be innacurate when there are covariates that could have a large influence on the dependence structure of the data. To account for this, a Bayesian framework for the estimation of conditional copulas is proposed. In this framework the parameters of a copula are non-linearly related to some arbitrary conditioning variables. We evaluate the ability of our method to predict time-varying dependencies on several equities and currencies and observe consistent performance gains compared to static copula models and other time-varying copula methods.

1 Introduction

Understanding dependencies within multivariate data is a central problem in the analysis of financial time series, underpinning common tasks such as portfolio construction and calculation of value-at-risk. Classical methods estimate these dependencies in terms of a covariance matrix (possibly time varying) which is induced from the data [4, 5, 7, 1]. However, a more general approach is to use copula functions to model dependencies [6]. Copulas have become popular since they separate the estimation of marginal distributions from the estimation of the dependence structure, which is completely determined by the copula.

The use of copulas to estimate dependencies is likely to be innacurate when the actual dependencies are strongly influenced by other covariates. For example, dependencies can vary with time or be affected by observations of other time series. Standard copula methods cannot handle such conditional dependencies. To address this limitation, we propose a probabilistic framework to estimate conditional copulas. Specifically we assume parametric copulas whose parameters are specified by unknown non-linear functions of arbitrary conditioning variables. These latent functions are approximated using Gaussian processes (GP) [17].

GPs have previously been used to model conditional copulas in [12] but this work only applies to copulas specified by a single parameter. We extend this work to accommodate copulas with multiple parameters. This is an important improvement since it allows the use of a richer set of copulas including Student t and asymmetric copulas.

We demonstrate our method by choosing the conditioning variables to be time and evaluating its ability to estimate time-varying dependencies on several currency and equity time series. Our method

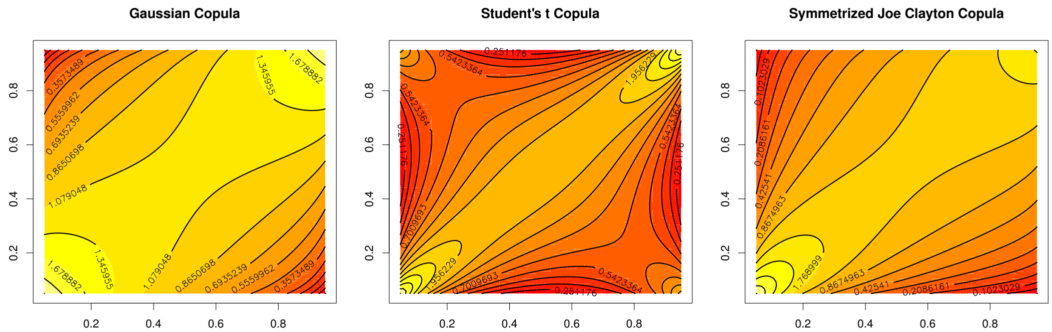

Figure 1: Left, Gaussian copula density for $\tau = 0.3$. Middle, Student's t copula density for $\tau = 0.3$ and $\nu = 1$. Right, symmetrized Joe Clayton copula density for $\tau^U = 0.1$ and $\tau^L = 0.6$. The latter copula model is asymmetric along the main diagonal of the unit square.

achieves consistently superior predictive performance compared to static copula models and other dynamic copula methods. These include models that allow their parameters to change with time, e.g. regime switching models [11] and methods proposing GARCH-style updates to copula parameters [20, 11].

The rest of the manuscript is organized as follows: Section 2 describes copulas and conditional copulas. Section 3 shows how to use GPs to construct a conditional copula model and demonstrates how to use expectation propagation (EP) for approximate inference. Section 4 discusses related work. Section 5 illustrates the performance of the proposed framework on synthetic data and several currency and equity time series. We conclude in Section 6.

2 Copulas and Conditional Copulas

Copulas provide a powerful framework for the construction of multivariate probabilistic models by separating the modeling of univariate marginal distributions from the modeling of dependencies between variables [6]. We focus on bivariate copulas since higher dimensional copulas are typically constructed using bivariate copulas as building blocks [e.g 2, 12].

Sklar's theorem [18] states that given two one-dimensional random variables, X and Y, with marginal cumulative density functions (cdfs) $F_X(X)$ and $F_Y(Y)$, we can express their joint cdf $F_{X,Y}$ as $F_{X,Y}(x, y) = C_{X,Y}[F_X(x), F_Y(y)]$, where $C_{X,Y}$ is the unique *copula* for X and Y. Since $F_X(X)$ and $F_Y(Y)$ are marginally uniformly distributed on $[0, 1]$, $C_{X,Y}$ is the cdf of a probability distribution on the unit square $[0, 1] \times [0, 1]$ with uniform marginals. Figure 1 shows plots of the copula densities for three parametric copula models: Gaussian, Student t and the symmetrized Joe Clayton (SJC) copulas.

Copula models can be learnt in a two step process [10]. First, the marginals F_X and F_Y are learnt by fitting univariate models. Second, the data are mapped to the unit square by $U = F_X(X), V = F_Y(Y)$ (i.e. a probability integral transform) and then $C_{X,Y}$ is then fit to the transformed data.

2.1 Conditional Copulas

When one has access to a covariate vector $\mathbf{Z}$, one may wish to estimate a conditional version of a copula model i.e.

$$F_{X,Y|\mathbf{Z}}(x, y|\mathbf{z}) = C_{X,Y|\mathbf{Z}} \left[F_{X|\mathbf{Z}}(x|\mathbf{z}), F_{Y|\mathbf{Z}}(y|\mathbf{z})|\mathbf{z}\right] . \tag{1}$$

Here, the same two-step estimation process can be used to estimate $F_{X,Y|\mathbf{Z}}(x, y|\mathbf{z})$. The estimation of the marginals $F_{X|\mathbf{Z}}$ and $F_{Y|\mathbf{Z}}$ can be implemented using standard methods for univariate conditional distribution estimation. However, the estimation of $C_{X,Y|\mathbf{Z}}$ is constrained to have uniform marginal distributions; this is a problem that has only been considered recently [12]. We propose a general Bayesian non-parametric framework for the estimation of conditional copulas based on GPs and an alternating expectation propagation (EP) algorithm for efficient approximate inference.

3 Gaussian Process Conditional Copulas

Let $\mathcal{D}_{\mathbf{Z}} = \{\mathbf{z}_i\}_{i=1}^n$ and $\mathcal{D}_{U,V} = \{(u_i, v_i)\}_{i=1}^n$ where (u_i, v_i) is a sample drawn from $C_{X,Y|\mathbf{z}_i}$. We assume that $C_{X,Y|\mathbf{Z}}$ is a parametric copula model $C_{\text{par}}[u, v|\theta_1(\mathbf{z}), \ldots, \theta_k(\mathbf{z})]$ specified by k parameters $\theta_1, \ldots, \theta_k$ that may be functions of the conditioning variable $\mathbf{z}$. Let $\theta_i(\mathbf{z}) = \sigma_i[f_i(\mathbf{z})]$, where f_i is an arbitrary real function and σ_i is a function that maps the real line to a set Θ_i of valid configurations for θ_i. For example, C_{par} could be a Student t copula. In this case, $k = 2$ and θ_1 and θ_2 are the correlation and the degrees of freedom in the Student t copula, $\Theta_1 = (-1, 1)$ and $\Theta_2 = (0, \infty)$. One could then choose $\sigma_1(.) = 2\Phi(.) - 1$, where Φ is the standard Gaussian cdf and $\sigma_2(.) = \exp(.)$ to satisfy the constraint sets Θ_1 and Θ_2 respectively.

Once we have specified the parametric form of C_{par} and the mapping functions $\sigma_1, \ldots, \sigma_k$, we need to learn the latent functions $f_1, \ldots, f_k$. We perform a Bayesian non-parametric analysis by placing GP priors on these functions and computing their posterior distribution given the observed data.

Let $\mathbf{f}_i = (f_i(\mathbf{z}_1), \ldots, f_i(\mathbf{z}_n))^{\text{T}}$. The prior distribution for $\mathbf{f}_i$ given $\mathcal{D}_{\mathbf{Z}}$ is $p(\mathbf{f}_i|\mathcal{D}_{\mathbf{Z}}) = \mathcal{N}(\mathbf{f}_i|\mathbf{m}_i, \mathbf{K}_i)$, where $\mathbf{m}_i = (m_i(\mathbf{z}_1), \ldots, m_i(\mathbf{z}_n))^{\text{T}}$ for some mean function $m_i(\mathbf{z})$ and $\mathbf{K}_i$ is an $n \times n$ covariance matrix generated by the squared exponential covariance function, i.e.

$$[K_i]_{jk} = \text{Cov}[f_i(\mathbf{z}_j), f_i(\mathbf{z}_k)] = \beta_i \exp\left\{-(\mathbf{z}_j - \mathbf{z}_k)^{\text{T}}\text{diag}(\boldsymbol{\lambda}_i)(\mathbf{z}_j - \mathbf{z}_k)\right\} + \gamma_i\,, \tag{2}$$

where $\boldsymbol{\lambda}_i$ is a vector of inverse length-scales and β_i, γ_i are amplitude and noise parameters. The posterior distribution for $\mathbf{f}_1, \ldots, \mathbf{f}_k$ given $\mathcal{D}_{U,V}$ and $\mathcal{D}_{\mathbf{Z}}$ is

$$p(\mathbf{f}_1, \ldots, \mathbf{f}_k|\mathcal{D}_{U,V}, \mathcal{D}_{\mathbf{Z}}) = \frac{\left[\prod_{i=1}^n c_{\text{par}}\left[u_i, v_i|\sigma_1\left[f_1(\mathbf{z}_i)\right], \ldots, \sigma_k\left[f_k(\mathbf{z}_h)\right]\right]\right]\left[\prod_{i=1}^k \mathcal{N}(\mathbf{f}_i|\mathbf{m}_i, \mathbf{K}_i)\right]}{p(\mathcal{D}_{U,V}|\mathcal{D}_{\mathbf{Z}})}\,, \tag{3}$$

where c_{par} is the density of the parametric copula model and $p(\mathcal{D}_{U,V}|\mathcal{D}_{\mathbf{Z}})$ is a normalization constant often called the *model evidence*. Given a particular value of $\mathbf{Z}$ denoted by $\mathbf{z}^\star$, we can make predictions about the conditional distribution of U and V using the standard GP prediction formula

$$\begin{aligned} p(u^\star, v^\star|\mathbf{z}^\star) = \int & c_{\text{par}}(u^\star, v^\star|\sigma_1[f_1^\star], \ldots, \sigma_k[f_k^\star]) p(\mathbf{f}^\star|\mathbf{f}_1, \ldots, \mathbf{f}_k, \mathbf{z}^\star, \mathcal{D}_{\mathbf{z}}) \\ & p(\mathbf{f}_1, \ldots, \mathbf{f}_k|\mathcal{D}_{U,V}, \mathcal{D}_{\mathbf{Z}})\, d\mathbf{f}_1, \ldots, d\mathbf{f}_k\, d\mathbf{f}^\star\,, \end{aligned} \tag{4}$$

where $\mathbf{f}^\star = (f_1^\star, \ldots, f_k^\star)^{\text{T}}$, $p(\mathbf{f}^\star|\mathbf{f}_1, \ldots, \mathbf{f}_k, \mathbf{z}^\star, \mathcal{D}_{\mathbf{z}}) = \prod_{i=1}^k p(f_i^\star|\mathbf{f}_i, \mathbf{z}^\star, \mathcal{D}_{\mathbf{z}})$, $f_i^\star = f_i(\mathbf{z}^\star)$, $p(f_i^\star|\mathbf{f}_i, \mathbf{z}^\star, \mathcal{D}_{\mathbf{z}}) = \mathcal{N}(f_i^\star|m_i(\mathbf{z}^\star) + \mathbf{k}_i^{\text{T}}\mathbf{K}_i^{-1}(\mathbf{f}_i - \mathbf{m}_i), k_i - \mathbf{k}_i^{\text{T}}\mathbf{K}_i^{-1}\mathbf{k}_i)$, $k_i = \text{Cov}[f_i(\mathbf{z}^\star), f_i(\mathbf{z}^\star)]$ and $\mathbf{k}_i = (\text{Cov}[f_i(\mathbf{z}^\star), f_i(\mathbf{z}_1)], \ldots, \text{Cov}[f_i(\mathbf{z}^\star), f_i(\mathbf{z}_n)])^{\text{T}}$. Unfortunately, (3) and (4) cannot be computed analytically, so we approximate them using expectation propagation (EP) [13].

3.1 An Alternating EP Algorithm for Approximate Bayesian Inference

The joint distribution for $\mathbf{f}_1, \ldots, \mathbf{f}_k$ and $\mathcal{D}_{U,V}$ given $\mathcal{D}_{\mathbf{Z}}$ can be written as a product of $n + k$ factors:

$$p(\mathbf{f}_1, \ldots, \mathbf{f}_k, \mathcal{D}_{U,V}|\mathcal{D}_{\mathbf{Z}}) = \left[\prod_{i=1}^n g_i(f_{1i}, \ldots, f_{ki},)\right]\left[\prod_{i=1}^k h_i(\mathbf{f}_i)\right]\,, \tag{5}$$

where $f_{ji} = f_j(\mathbf{z}_i)$, $h_i(\mathbf{f}_i) = \mathcal{N}(\mathbf{f}_i|\mathbf{m}_i, \mathbf{K}_i)$ and $g_i(f_{1i}, \ldots, f_{ki}) = c_{\text{par}}[u_i, v_i|\sigma_1[f_{1i}], \ldots, \sigma_k[f_{ki}]]$. EP approximates each factor g_i with an approximate Gaussian factor $\tilde{g}_i$ that may not integrate to one, i.e. $\tilde{g}_i(f_{1i}, \ldots, f_{ki}) = s_i \prod_{j=1}^k \exp\left\{-(f_{ji} - \tilde{m}_{ji})^2/[2\tilde{v}_{ji}]\right\}$, where $s_i > 0$, $\tilde{m}_{ji}$ and $\tilde{v}_{ji}$ are parameters to be calculated by EP. The other factors h_i already have a Gaussian form so they do not need to be approximated. Since all the $\tilde{g}_i$ and h_i are Gaussian, their product is, up to a normalization constant, a multivariate Gaussian distribution $q(\mathbf{f}_1, \ldots, \mathbf{f}_k)$ which approximates the exact posterior (3) and factorizes across $\mathbf{f}_1, \ldots, \mathbf{f}_k$. The predictive distribution (4) is approximated by first integrating $p(\mathbf{f}^\star|\mathbf{f}_1, \ldots, \mathbf{f}_k, \mathbf{z}^\star, \mathcal{D}_{\mathbf{z}})$ with respect to $q(\mathbf{f}_1, \ldots, \mathbf{f}_k)$. This results in a factorized Gaussian distribution $q^\star(\mathbf{f}^\star)$ which approximates $p(\mathbf{f}^\star|\mathcal{D}_{U,V}, \mathcal{D}_{\mathbf{Z}})$. Finally, (4) is approximated by Monte-Carlo by sampling from $q^\star$ and then averaging $c_{\text{par}}(u^\star, v^\star|\sigma_1[f_1^\star], \ldots, \sigma_k[f_k^\star])$ over the samples.

EP iteratively updates each $\tilde{g}_i$ until convergence by first computing $q^{\setminus i} \propto q/\tilde{g}_i$ and then minimizing the Kullback-Leibler divergence [3] between $g_i q^{\setminus i}$ and $\tilde{g}_i q^{\setminus i}$. This involves updating $\tilde{g}_i$ so that the first and second marginal moments of $g_i q^{\setminus i}$ and $\tilde{g}_i q^{\setminus i}$ match. However, we cannot compute the moments of $g_i q^{\setminus i}$ analytically due to the complicated form of g_i. A typical solution is to use numerical methods to compute these k-dimensional integrals. However, this typically has an exponentially large computational cost in k which is prohibitive for $k > 1$. Instead we perform an additional approximation when computing the marginal moments of f_{ji} with respect to $g_i q^{\setminus i}$. Without loss of generality, assume that we want to compute the expectation of f_{1i} with respect to $g_i q^{\setminus i}$. We make the following approximation:

$$\int f_{1i} g_i(f_{1i}, \ldots, f_{ki}) q^{\setminus i}(f_{1i}, \ldots, f_{ki})\, df_{1i}, \ldots, df_{ki} \approx \\ C \times \int f_{1i} g_i(f_{1i}, \bar{f}_{2i}, \ldots, \bar{f}_{ki}) q^{\setminus i}(f_{1i}, \bar{f}_{2i}, \ldots, \bar{f}_{ki})\, df_{1i}\,, \qquad (6)$$

where $\bar{f}_{1i}, \ldots, \bar{f}_{ki}$ are the means of $f_{1i}, \ldots, f_{ki}$ with respect to the posterior approximation q, and C is a constant that approximates the width of the integrand around its maximum in all dimensions except f_{1i}. In practice all moments are normalised by the 0-th moment so C can be ignored. The right hand side of (6) is a one-dimensional integral that can be easily computed using numerical techniques. The approximation above is similar to approximating an integral by the product of the maximum value of the integrand and an estimate of its width. However, instead of maximizing $g_i(f_{1i}, \ldots, f_{ki}) q^{\setminus i}(f_{1i}, \ldots, f_{ki})$ with respect to $f_{2i}, \ldots, f_{ki}$, we are maximizing q. This is a much easier task because q is Gaussian and its maximizer is its own mean vector. Note that q and $g_i(f_{1i}, \ldots, f_{ki}) q^{\setminus i}(f_{1i}, \ldots, f_{ki})$ should be similar since they are both approximating (5). Therefore, we expect (6) to be a good approximation. The other moments are evaluated similarly.

Since q factorizes across $\mathbf{f}_1, \ldots, \mathbf{f}_k$, our implementation of EP decouples into k EP sub-routines among which we alternate; the j-th sub-routine approximates the posterior distribution of $\mathbf{f}_j$ using as input the means of the approximate distributions generated by the other EP sub-routines. Each sub-routine finds a Gaussian approximation to a set of n one-dimensional factors; one factor per data point. In the j-th EP sub-routine, the i-th factor is given by $g_i(f_{1i}, \ldots, f_{ki})$, where each $\{f_{1i}, \ldots, f_{ki}\} \setminus \{f_{ji}\}$ is kept fixed to its current approximate posterior mean, as estimated by the other EP sub-routines. We iteratively alternate between the different sub-routines, running each one until convergence before re-running the next one. Convergence is achieved very quickly; we only run each EP sub-routine four times.

The EP sub-routines are implemented using the parallel EP update scheme described in [21]. To speed up GP related computations, we use the generalized FITC approximation [19, 14]: Each $n \times n$ covariance matrix $\mathbf{K}_i$ is approximated by $\mathbf{K}'_i = \mathbf{Q}_i + \mathrm{diag}(\mathbf{K}_i - \mathbf{Q}_i)$, where $\mathbf{Q}_i = \mathbf{K}^i_{nn_0}[\mathbf{K}^i_{n_0n_0}]^{-1}[\mathbf{K}^i_{nn_0}]^{\mathrm{T}}$, $\mathbf{K}^i_{n_0n_0}$ is the $n_0 \times n_0$ covariance matrix generated by evaluating (2) at $n_0 \ll n$ *pseudo-inputs*, and $\mathbf{K}^i_{nn_0}$ is the $n \times n_0$ matrix with the covariances between training points and pseudo-inputs. The cost of EP is $O(knn_0^2)$. Each time we call the j-th EP subroutine, we optimize the corresponding kernel hyper-parameters $\boldsymbol{\lambda}_j$, β_j and γ_j and the pseudo-inputs by maximizing the EP approximation of the model evidence [17].

4 Related Work

The model proposed here is an extension of the conditional copula model of [12]. In the case of bivariate data and a copula based on one parameter the models are identical. We have extended the approximate inference for this model to accommodate copulas with multiple parameters; previously computationally infeasible due to requiring the numerical calculation of multidimensional integrals within an inner loop of EP inference. We have also demonstrated that one can use this model to produce excellent predictive results on financial time series by conditioning the copula on time. [12] reported improved performance over benchmarks for all data sets *except* stock index data. This was due to choosing other time series variables to be the conditioning variables (an appropriate modeling method for all other data sets), rather than time.

4.1 Dynamic Copula Models

In [11] a dynamic copula model is proposed based on a two-state hidden Markov model (HMM) ($S_t \in \{0, 1\}$) that assumes that the data generating process changes between two regimes of low/high correlation. At any time t the copula density is Student t with different parameters for the two values of the hidden state S_t. Maximum likelihood estimation of the copula parameters and transition probabilities is performed using an EM algorithm [e.g. 3].

A time-varying correlation (TVC) model based on the Student t copula is described in [20, 11]. The correlation parameter[1] of a Student's t copula is assumed to satisfy $\rho_t = (1-\alpha-\beta)\rho + \alpha\varepsilon_{t-1} + \beta\rho_{t-1}$, where ε_{t-1} is the empirical correlation of the previous 10 observations and ρ, α and β satisfy $-1 \leq \rho \leq 1$, $0 \leq \alpha, \beta \leq 1$ and $\alpha + \beta \leq 1$. The number of degrees of freedom ν is assumed to be constant. The previous formula is the GARCH equation for correlation instead of variance. Estimation of ρ, α, β and ν is easily performed by maximum likelihood.

In [15] a dynamic copula based on the SJC copula (DSJCC) is introduced. In this method, the parameters τ^U and τ^L of an SJC copula are assumed to depend on time according to

$$\tau^U(t) = 0.01 + 0.98\Lambda\left[\omega_U + \alpha_U\varepsilon_{t-1} + \beta_U\tau^U(t-1)\right] , \tag{7}$$

$$\tau^L(t) = 0.01 + 0.98\Lambda\left[\omega_L + \alpha_L\varepsilon_{t-1} + \beta_L\tau^L(t-1)\right] , \tag{8}$$

where $\Lambda[\cdot]$ is the logistic function, $\varepsilon_{t-1} = \frac{1}{10}\sum_{j=1}^{10}|u_{t-j} - v_{t-j}|$, (u_t, v_t) is a copula sample at time t and the constants are used to avoid numerical instabilities. These formulae are the GARCH equation for correlations, with an additional logistic function to constrain parameter values.

We go beyond this prior work by allowing copula parameters to depend on an arbitrary conditioning variables rather than time alone. Also, the models above either assume Markov independence or GARCH-like updates to copula parameters. These assumptions have been empirically proven to be effective for the estimation of univariate variances, but the consistent performance gains of our proposed method suggest these assumptions are less applicable for the estimation of dependencies.

4.2 Other Dynamic Covariance Models

A direct extension of the GARCH equations to multiple time series, VEC, was proposed by [5]. Let $\mathbf{x}(t)$ be a multivariate time series assumed to satisfy $\mathbf{x}(t) \sim \mathcal{N}(0, \Sigma(t))$. VEC($p, q$) models the dynamics of $\Sigma(t)$ by an equation of the form

$$\text{vech}(\Sigma(t)) = c + \sum_{k=1}^{p} A_k \,\text{vech}(\mathbf{x}(t-k)\mathbf{x}(t-k)^{\mathrm{T}}) + \sum_{k=1}^{q} B_k \,\text{vech}(\Sigma(t-k)) \tag{9}$$

where vech is the operation that stacks the lower triangular part on a matrix into a column vector.

The VEC model has a very large number of parameters and hence a more commonly used model is the BEKK(p, q) model [7] which assume the following dynamics

$$\Sigma(t) = C^{\mathrm{T}}C + \sum_{k=1}^{p} A_k^{\mathrm{T}}\mathbf{x}(t-k)\mathbf{x}(t-k)^{\mathrm{T}}A_k + \sum_{k=1}^{q} B_k^{\mathrm{T}}\Sigma(t-k)B_k. \tag{10}$$

This model also has many parameters and many restricted versions of these models have been proposed to avoid over-fitting (see e.g. section 2 of [1]).

An alternative solution to over-fitting due to over-parametrisation is the Bayesian approach of [23] where Bayesian inference is performed in a dynamic BEKK$(1, 1)$ model. Other Bayesian approaches include the non-parametric generalised Wishart process [22, 8]. In these works $\Sigma(t)$ is modeled by a generalised Wishart process i.e.

$$\Sigma(t) = \sum_{i=1}^{\nu} L\mathbf{u}_i(t)\mathbf{u}_i(t)^{\mathrm{T}}L^{\mathrm{T}} \tag{11}$$

where $u_{id}(.)$ are distributed as independent GPs.

5 Experiments

We evaluate the proposed Gaussian process conditional copula models (GPCC) on a one-step-ahead prediction task with synthetic data and financial time series. We use time as the conditioning variable and consider three parametric copula families; Gaussian (GPCC-G), Student t (GPCC-T) and

[1] The parametrisation used in this paper is related by $\rho = \sin(0.5\tau\pi)$

symmetrized Joe Clayton (GPCC-SJC). The parameters of these copulas are presented in Table 1 along with the transformations used to model them. Figure 1 shows plots of the densities of these three parametric copula models.

Copula	Parameters	Transformation	Synthetic parameter function
Gaussian	correlation, τ	$0.99(2\Phi[f(t)]-1)$	$\tau(t) = 0.3 + 0.2\cos(t\pi/125)$
Student t	correlation, τ	$0.99(2\Phi[f(t)]-1)$	$\tau(t) = 0.3 + 0.2\cos(t\pi/125)$
	degrees of freedom, ν	$1 + 10^6\Phi[g(t)]$	$\nu(t) = 1 + 2(1 + \cos(t\pi/250))$
SJC	upper dependence, τ^U	$0.01 + 0.98\Phi[g(t)]$	$\tau^U(t) = 0.1 + 0.3(1 + \cos(t\pi/125))$
	lower dependence, τ^L	$0.01 + 0.98\Phi[g(t)]$	$\tau^L(t) = 0.1 + 0.3(1 + \cos(t\pi/125 + \pi/2))$

Table 1: Copula parameters, modeling formulae and parameter functions used to generate synthetic data. Φ is the standard Gaussian cumulative density function f and g are GPs.

The three variants of GPCC were compared against three dynamic copula methods and three constant copula models. The three dynamic methods include the HMM based model, TVC and DSJCC introduced in Section 4. The three constant copula models use Gaussian, Student t and SJC copulas with parameter values that do not change with time (CONST-G, CONST-T and CONST-SJC).

We perform a one-step-ahead rolling-window prediction task on bivariate time series $\{(u_t, v_t)\}$. Each model is trained on the first n_W data points and the predictive log-likelihood of the $(n_W + 1)$th data point is recorded. This is then repeated, shifting the training and test window forwards by one data point. The methods are then compared by average predictive log-likelihood; an appropriate measure of performance when estimating copula functions since they are probability distributions.

5.1 Synthetic Data

We generated three synthetic data sets of length 5001 from copula models (Gaussian, Student t, SJC) whose parameters vary as periodic functions of time, as specified in Table 1.

Table 2 reports the average predictive log-likelihood for each method on each synthetic time series. The results of the best performing method on each synthetic time series are shown in bold. The results of any other method are underlined when the differences with respect to the best performing method are not statistically significant according to a paired t test at $\alpha = 0.05$.

GPCC-T and GPCC-SJC obtain the best results in the Student t and SJC time series respectively. However, HMM is the best performing method for the Gaussian time series, This technique successfully captures the two regimes of low/high correlation corresponding to the peaks and troughs of the sinusoid that maps time t to correlation τ. The proposed methods GPCC-[G,T,SJC] are more flexible and hence less efficient than HMM in this particular problem. However, HMM performs significantly worse in the Student t and SJC time series since the different periods for the different copula parameter functions cannot be captured by a two state model.

Figure 2 shows how GPCC-T successfully tracks $\tau(t)$ and $\nu(t)$ in the Student t time series. The plots display the mean (red) and confidence bands (orange, 0.1 and 0.9 quantiles) for the predictive distribution of $\tau(t)$ and $\nu(t)$ as well as the ground truth values (blue).

Finally, Table 2 also shows that the static copula methods CONST-[G,T,SJC] are usually outperformed by all dynamic techniques GPCC-[G,T,SJC], DSJCC, TVC and HMM.

5.2 Foreign Exchange Time Series

We evaluated each method on the daily logarithmic returns of nine currency pairs shown in Table 3 (all paired with the U.S. dollar)[2]. The date range of the data is 02-01-1990 to 15-01-2013; a total of 6011 observations. We evaluated the methods on eight bivariate time series, pairing each currency pair with the Swiss franc (CHF). CHF is known to be a *safe haven* currency, meaning that investors flock to it during times of uncertainty [16]. Consequently we expect correlations between CHF and other currencies to have large variability across time in response to changes in financial conditions.

[2]All data was downloaded from `http://finance.yahoo.com/`. Data sets and source code will be made available upon publication.

We first process our data using an asymmetric AR(1)-GARCH(1,1) process with non-parametric innovations [9] to estimate the univariate marginal cdfs at all time points. We train this GARCH model on $n_W = 2016$ data points and then predict the cdf of the next data point; subsequent cdfs are predicted by shifting the training window by one data point in a rolling-window methodology. The cdf estimates are used to transform the raw logarithmic returns (x_t, y_t) into a pseudo-sample of the underlying copula (u_t, v_t) as described in Section 2. We note that any method for predicting univariate cdfs could have been used to produce pseudo-samples from the copula.

We then perform the rolling-window predictive likelihood experiment on the transformed data. The results are shown in Table 4; overall the best technique is GPCC-T, followed by GPCC-G. The dynamic copula methods GPCC-[G,T,SJC], HMM, and TVC outperform the static methods CONST-[G,T,SJC] in all the analyzed series. The dynamic method DSJCC occasionally performed poorly; worse than the static methods for 3 experiments.

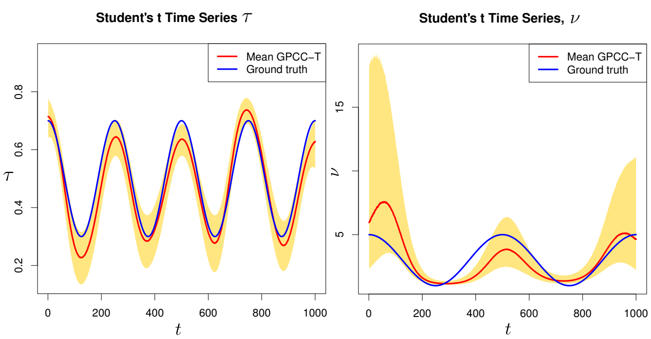

Figure 2: Predictions made by GPCC-T for $\nu(t)$ and $\tau(t)$ on the synthetic time series sampled from a Student's t copula.

Method	Gaussian	Student	SJC
GPCC-G	0.3347	0.3879	0.2513
GPCC-T	0.3397	**0.4656**	0.2610
GPCC-SJC	0.3355	0.4132	**0.2771**
HMM	**0.3555**	0.4422	0.2547
TVC	0.3277	0.4273	0.2534
DSJCC	0.3329	0.4096	0.2612
CONST-G	0.3129	0.3201	0.2339
CONST-T	0.3178	0.4218	0.2499
CONST-SJC	0.3002	0.3812	0.2502

Table 2: Avg. test log-likelihood of each method on each time series.

Code	Currency Name
CHF	Swiss Franc
AUD	Australian Dollar
CAD	Canadian Dollar
JPY	Japanese Yen
NOK	Norwegian Krone
SEK	Swedish Krona
EUR	Euro
NZD	New Zeland Dollar
GBP	British Pound

Table 3: Currencies.

Method	AUD	CAD	JPY	NOK	SEK	EUR	GBP	NZD
GPCC-G	0.1247	0.1133	0.1450	**0.2072**	0.1536	0.2424	0.3401	0.1860
GPCC-T	**0.1289**	**0.1187**	**0.1499**	0.2059	**0.1591**	0.2486	**0.3501**	**0.1882**
GPCC-SJC	0.1210	0.1095	0.1399	0.1935	0.1462	0.2342	0.3234	0.1753
HMM	0.1260	0.1119	0.1458	0.2040	0.1511	**0.2486**	0.3414	0.1818
TVC	0.1251	0.1119	0.1459	0.2011	0.1511	0.2449	0.3336	0.1823
DSJCC	0.0935	0.0750	0.1196	0.1721	0.1163	0.2188	0.3051	0.1582
CONST-G	0.1162	0.1027	0.1288	0.1962	0.1325	0.2307	0.2979	0.1663
CONST-T	0.1239	0.1091	0.1408	0.2007	0.1481	0.2426	0.3301	0.1775
CONST-SJC	0.1175	0.1046	0.1307	0.1891	0.1373	0.2268	0.2992	0.1639

Table 4: Avg. test log-likelihood of each method on the currency data.

The proposed method GPCC-T can capture changes across time in the parameters of the Student t copula. The left and middle plots in Figure 3 show predictions for $\nu(t)$ and $\tau(t)$ generated by GPCC-T. In the left plot, we observe a reduction in $\nu(t)$ at the onset of the 2008-2012 global recession indicating that the return series became more prone to outliers. The plot for $\tau(t)$ (middle) also shows large changes across time. In particular, we observe large drops in the dependence level between EUR-USD and CHF-USD during the fall of 2008 (at the onset of the global recession) and the summer of 2010 (corresponding to the worsening European sovereign debt crisis).

For comparison, we include predictions for $\tau^L(t)$ and $\tau^U(t)$ made by GPCC-SJC in the right plot of figure 3. In this case, the prediction for $\tau^U(t)$ is similar to the one made by GPCC-T for $\tau(t)$, but the prediction for $\tau^L(t)$ is much noisier and erratic. This suggests that GPCC-SJC is less robust than GPCC-T. All the copula densities in figure 1 take large values in the proximity of the points (0,0) and (1,1) i.e. positive correlation. However, the Student t copula is the only one of these three copulas which can take high values in the proximity of the points (0,1) and (1,0) i.e. negative correlation. The plot in the left of Figure 3 shows how $\nu(t)$ takes very low values at the end of the time period, increasing the robustness of GPCC-T to negatively correlated outliers.

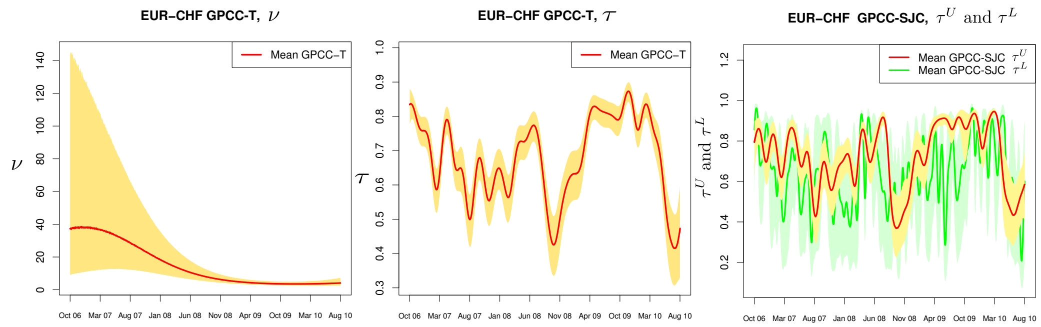

Figure 3: Left and middle, predictions made by GPCC-T for $\nu(t)$ and $\tau(t)$ on the time series EUR-CHF when trained on data from 10-10-2006 to 09-08-2010. There is a significant reduction in $\nu(t)$ at the onset of the 2008-2012 global recession. Right, predictions made by GPCC-SJC for $\tau^U(t)$ and $\tau^L(t)$ when trained on the same time-series data. The predictions for $\tau^L(t)$ are much more erratic than those for $\tau^U(t)$.

5.3 Equity Time Series

As a further comparison, we evaluated each method on the logarithmic returns of 8 equity pairs, from the same date range and processed using the same AR(1)-GARCH(1,1) model discussed previously. The equities were chosen to include pairs with both high correlation (e.g. RBS and BARC) and low correlation (e.g. AXP and BA).

The results are shown in Table 5; again the best technique is GPCC-T, followed by GPCC-G.

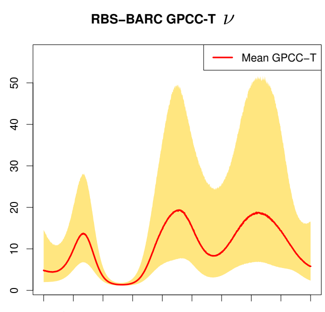

Figure 4: Prediction for $\nu(t)$ on RBS-BARC.

Method	HD HON	AXP BA	CNW CSX	ED EIX	HPQ IBM	BARC HSBC	RBS BARC	RBS HSBC
GPCC-G	0.1247	0.1133	0.1450	**0.2072**	0.1536	0.2424	0.3401	0.1860
GPCC-T	**0.1289**	**0.1187**	**0.1499**	0.2059	**0.1591**	0.2486	**0.3501**	**0.1882**
GPCC-SJC	0.1210	0.1095	0.1399	0.1935	0.1462	0.2342	0.3234	0.1753
HMM	0.1260	0.1119	0.1458	0.2040	0.1511	**0.2486**	0.3414	0.1818
TVC	0.1251	0.1119	0.1459	0.2011	0.1511	0.2449	0.3336	0.1823
DSJCC	0.0935	0.0750	0.1196	0.1721	0.1163	0.2188	0.3051	0.1582
CONST-G	0.1162	0.1027	0.1288	0.1962	0.1325	0.2307	0.2979	0.1663
CONST-T	0.1239	0.1091	0.1408	0.2007	0.1481	0.2426	0.3301	0.1775
CONST-SJC	0.1175	0.1046	0.1307	0.1891	0.1373	0.2268	0.2992	0.1639

Table 5: Average test log-likelihood for each method on each pair of stocks.

Figure 4 shows predictions for $\nu(t)$ generated by GPCC-T. We observe low values of ν during 2010 suggesting that a Gaussian copula would be a bad fit to the data. Indeed, GPCC-G performs significantly worse than GPCC-T on this equity pair.

6 Conclusions and Future Work

We have proposed an inference scheme to fit a conditional copula model to multivariate data where the copula is specified by multiple parameters. The copula parameters are modeled as unknown non-linear functions of arbitrary conditioning variables. We evaluated this framework by estimating time-varying copula parameters for bivariate financial time series. Our proposed method and inference consistently outperforms static copula models and other dynamic copula models.

In this initial investigation we have focused on bivariate copulas. Higher dimensional copulas are typically constructed using bivariate copulas as building blocks [2, 12]. Our framework could be applied to these constructions and our empirical predictive performance gains will likely transfer to this setting. Evaluating the effectiveness of this approach compared to other models of multivariate covariance would be a profitable area of empirical research.

One could also extend the analysis presented here by including additional conditioning variables as well as time. For example, including a prediction of univariate volatility as a conditioning variable would allow copula parameters to change in response to changing volatility. This would pose inference challenges as the dimension of the GP increases, but could create richer models.

References

[1] L. Bauwens, S. Laurent, and J. V. K. Rombouts. Multivariate garch models: a survey. *Journal of applied econometrics*, 21(1):79–109, 2006.

[2] T. Bedford and R. M. Cooke. Probability density decomposition for conditionally dependent random variables modeled by vines. *Annals of Mathematics and Artificial Intelligence*, 32(1-4):245–268, 2001.

[3] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 2007.

[4] T. Bollerslev. Generalized autoregressive conditional heteroskedasticity. *Journal of econometrics*, 31(3):307–327, 1986.

[5] T. Bollerslev, R. F. Engle, and J. M. Wooldridge. A capital asset pricing model with time-varying covariances. *The Journal of Political Economy*, pages 116–131, 1988.

[6] G. Elidan. Copulas and machine learning. In *Invited survey to appear in the proceedings of the Copulae in Mathematical and Quantitative Finance workshop*, 2012.

[7] R. F. Engle and K. F. Kroner. Multivariate simultaneous generalized arch. *Econometric theory*, 11(01):122–150, 1995.

[8] E. B. Fox and D. B. Dunson. Bayesian Nonparametric Covariance Regression. *Arxiv preprint arXiv:1101.2017*, 2011.

[9] J. M. Hernández-Lobato, D. Hernández-Lobato, and A. Suárez. GARCH processes with non-parametric innovations for market risk estimation. In *Artificial Neural Networks ICANN 2007*, volume 4669 of *Lecture Notes in Computer Science*, pages 718–727. Springer Berlin Heidelberg, 2007.

[10] H. Joe. Asymptotic efficiency of the two-stage estimation method for copula-based models. *J. Multivar. Anal.*, 94(2):401–419, June 2005.

[11] E. Jondeau and M. Rockinger. The copula-garch model of conditional dependencies: An international stock market application. *Journal of international money and finance*, 25(5):827–853, 2006.

[12] D. López-Paz, J. M. Hernández-Lobato, and Z. Ghahramani. Gaussian process vine copulas for multivariate dependence. In *Proceedings of the 30th International Conference on Machine Learning*, 2013.

[13] T. P. Minka. Expectation Propagation for approximate Bayesian inference. *Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence*, pages 362–369, 2001.

[14] A. Naish-Guzman and S. B. Holden. The generalized FITC approximation. In *Advances in Neural Information Processing Systems 20*, 2007.

[15] A. J. Patton. Modelling asymmetric exchange rate dependence. *International economic review*, 47(2):527–556, 2006.

[16] A. Ranaldo and P. Söderlind. Safe haven currencies. *Review of Finance*, 14(3):385–407, 2010.

[17] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 1st edition, 2006.

[18] A. Sklar. Fonctions de répartition à n dimensions et leurs marges. *Publ. Inst. Statis. Univ. Paris*, 8(1):229–231, 1959.

[19] E. Snelson and Z. Ghahramani. Sparse Gaussian processes using pseudo-inputs. *Proceedings of the 20th Conference in Advances in Neural Information Processing Systems*, pages 1257–1264, 2006.

[20] Y. K. Tse and A. K. C. Tsui. A multivariate generalized autoregressive conditional heteroscedasticity model with time-varying correlations. *Journal of Business & Economic Statistics*, 20(3):pp. 351–362, 2002.

[21] M. A. J. van Gerven, B. Cseke, F. P. de Lange, and T. Heskes. Efficient bayesian multivariate fmri analysis using a sparsifying spatio-temporal prior. *NeuroImage*, 50(1):150–161, 2010.

[22] A. G. Wilson and Z. Ghahramani. Generalised wishart processes. In Fabio Cozman and Avi Pfeffer, editors, *Proceedings of the Twenty-Seventh Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-11)*, Barcelona, Spain, 2011. AUAI Press.

[23] Y. Wu, J. M. Hernández-Lobato, and Z. Ghahramani. Dynamic Covariance Models for Multivariate Financial Time Series. *Arxiv preprint arXiv:1305.4268v1*, 2013.