Bayesian approaches of mixture copulas with applications

Thesis submitted in accordance with the requirements of the University of Liverpool for the degree of Doctor in Philosophy

by

Yujian Liu

Principal Supervisor: Dr. Dejun Xie

January 2024

Acknowledgements

Years of PhD study will soon come to an end. Academic life is full of challenges, but also encouraging. Although the daily academic norms are full of lengthy and painful repetitions of labs and writing, I enjoy every moment that a small spark comes up in my mind and some tiny progress is made. This journey is in no terms easy. I therefore feel very grateful to those who accompany me, guide me, and give me any help.

I want to express my sincere thanks to my principal supervisor Dr.Dejun Xie, who guided me from life to academics in countless days and nights. He is not only my supervisor but also a dear friend, a knowledgeable life mentor.

I feel very thankful to my dearest parents, who give me support financially and mentally throughout my journey of studies, without whom my pursuit of academics would be impossible.

Last but not least, I want to thank my girlfriend Noya, who gives me love, companionship and always has faith in me. Her encouragement and love are indispensable.

Abstract

Copula theory has become one of the most important ideologies and methodologies for modeling the dependence among random variables. Rather than using point performance metrics such as Pearson linear correlation, copula functions enable us to construct the multivariate distributions among the concerned random variables by starting from the corresponding marginal distributions. Hence, it gives us a full description of the dependence mode.

The most frequently used copula models are parametric copulas such as Gaussian, Clayton, and Gumbel copulas. However, in many practical scenarios, these copulas often fail to fully describe the dependence as real data often contain complex patterns with multi-modals. In addition, classic copulas are mostly studied in their bivariate form, leaving the application of copulas into higher dimensional data non-trivial. This thesis intends to approach the above-mentioned problems by utilizing Bayesian samplers into mixture copulas. In particular, we study the problems of estimating, selecting, and simulating mixture components of copulas by using Bayesian approaches. Families of multivariate elliptical and skew-elliptical copulas are given special attention as they can be naturally extended to higher dimensions.

For applications, we apply our proposed approaches to study the dependence among financial markets. Meanwhile, we extend the application of our Bayesian mixture copulas to improve the oversampling methods for imbalance learning problems in the field of data science.

The thesis mainly consists of four major parts. In the first part, we applied the Bayesian sparse finite mixture model to the copula mixture modeling, which enables us to estimate and select the correct finite mixture copulas simultaneously without having to repeatedly estimate various forms of models and compare their AICs or BICs.

The second part focused on the construction of infinite mixture t copulas using the Dirichlet process prior. Although we are concentrated on the t copulas due to their usefulness in financial applications. This approach can be extended to more general copulas. The approaches further advance the previously proposed finite mixture Bayesian approaches

despite being more complicated in terms of modeling.

The third part further extends previous parts to construct the non-parametric Bayesian copula mixture models for serially correlated data. In particular, we discuss the modeling of the hidden Markov models (HMM) with multivariate emission distributions. We use copula theories to decompose the construction of multivariate emission distributions into univariate marginal distributions and a dependence structure. Meanwhile, many real-life applications of HMM have an unknown number of states, which need to be manually specified by analysts if the classic HMM method is used. Introducing the hierarchical Dirichlet process into the Copula-HMM model enables us to infer the number of unknown states from the dataset automatically. We thoroughly introduce the inference method of this non-parametric Bayesian copula-HMM model therein.

The final part is about the introduction and study of the evaluation metrics of imbalance learning problems as well as applying the mixture copulas approach to solving the data imbalance. One major obstacle of applying the copulas approach to imbalanced datasets is the high dimensional features of many tasks. On the other hand, data science applications often include features that are discrete-valued, while most of the copulas literature only deals with continuous random vectors. Therefore, we develop the MCMC approaches for estimating the mixed valued copulas (i.e., the copula contains both continuous and discrete valued variables) and apply them to estimate the dataset and perform the oversampling. The Bayesian approach would be useful in these tasks as the real applications often involve high dimensional large dataset, whereas the classic MLE approaches struggle in this case due to the exponential complexity in evaluating the discrete dimensions. The approaches are applied to the simulated dataset to prove its validity in the paper. Meanwhile, the real oversampling task is performed using mixture copulas, and the results are compared with the classic random oversampling and the SMOTE approaches.

Declaration

I hereby declare that the work presented in this PhD thesis, is my own original work and has been carried out under the primary guidance of Dr. Dejun Xie. Any ideas, data, images, or text resulting from the work of others (whether published or unpublished) are fully identified with appropriate citations and included in the bibliography.

No portion of this work has been submitted in support of any other degree or qualification at this or any other institution. All research activities associated with this thesis have been conducted ethically and responsibly.

Yujian Liu
Oct.2023

List of Relevant Publications

Peer-reviewed Journal Publications

1. Liu, Y., Li, Y., & Xie, D. (2024). Implications of imbalanced datasets for empirical ROC-AUC estimation in binary classification tasks. Journal of Statistical Computation and Simulation, 94(1), 183-203

2. Liu, Y., Xie, D., & Yu, S. (2023). Bayesian Mixture Copula Estimation and Selection with Applications. Analytics, 2(2), 530-545.

3. Liu, Y., Xie, D., Edwards, D. A., & Yu, S. (2023). Mixture copulas with discrete margins and their application to imbalanced data. Journal of the Korean Statistical Society, 52(4), 878-900

4. Liu, Y., Xie, D., Li, Y., & Yu, S. (2023). Nonparametric Bayesian modeling on infinite mixture Student t copulas. Communications in Statistics-Simulation and Computation.

Manuscript submitted to the Journal

1. Liu, Y., Xie, D., & Yu, S. Copula hidden Markov model with unknown number of states. Computational Statistics (Under Review).

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Introduction

1.1.1 Background and motivation

The development of the concept of the copula, which can be dated back to the early work of Sklar (1959, 1973), has received tremendous attention in the past few decades, especially in the area of finance, where modeling the correlation across an asset and estimating the risk of a portfolio are essential (Cherubini et al., 2004; Jaworski et al., 2010; McNeil et al., 2015). The power of this tool arises from Sklar's theorem in (Sklar, 1959, 1973), which states that for an arbitrary multivariate distribution, there exists a copula that decomposes the high dimensional distributions into a link function and univariate margins. This enables us to construct the high dimensional structure using a "bottom-up" approach from the univariate margins instead of directly working with the multivariate part (Smith, 2011). Instead of using the summary statistics such as Pearson correlations, which can only capture the degree of bivariate linear dependence, modeling the dependence through copulas gives a full description of the dependence structure among the random variables.

When applying the copula methods to real life, such as modeling the dependence between global trading markets, it is often insufficient to rely on a single parametric copula family because of the data complexity and heterogeneity. A possible remedy is to use the mixture modeling, McLachlan et al. (2019) give a recent comprehensive review of this classic statistical topic. In terms of the copula functions, we are therefore motivated to write a copula function as the mixture of others.

In most of the research papers regarding the topics of finite mixture models, it is common

to assume that the mixed distributions come from the same parametric family. This is also mentioned in McLachlan et al. (2019); Gelman et al. (2013). However, in the literature on copula methods, mixture copulas consisting of several parametric copula families are also common. Hu (2006) is one of a few pioneering works of the mixture copula application in the field of finance. The author used mixed Normal-Gumbel-Survival Gumbel copula with empirical marginal distributions to model the stock dependence between FTSE 100 of the U.K., Nikkei 225 of Japan, S&P 500 of the U.S.A. and Hang Seng of H.K. S.A.R., quasi-likelihood was used for parameters estimation, and Chi-square test was applied for the goodness of fit. Arakelian and Karlis (2014) uses the expectation maximization (EM) approach to estimate the mixture copula with two components and use them to detect the changing dependence between financial markets, the combination of Gaussian, Clayton, and Gumbel Copula is mainly considered there, and the model selection criteria is log-likelihood. Vrac et al. (2012) combined the dynamic clustering with the gradient ascent to solve the mixture copula, Frank copula family with nonparametric margins are used in their geographical application, the best model is selected by minimizing the approximate weight of evidence (AWE), the asymptotic convergence of their methods are obtained. More recently, Liu et al. (2022) proposed to construct the semi-parametric conditional mixture copulas to assess the global currency market, their best models are selected by comparing the Bayes information criterion (BIC), and asymptotic consistency is obtained therein.

On the other hand, the vast majority of the mixture copula literature handles the model estimation and selection problem by employing the classic MLE-based framework (Hu, 2006; Arakelian and Karlis, 2014; Cai and Wang, 2014; Roy and Parui, 2014; Liu et al., 2019). The exploration of Bayesian approaches in the field of mixture copula remains relatively few. This is also true when we extend our discussion range to the whole field of copula modeling as pointed out by the survey of Smith (2011), where they comprehensively discussed the existing Bayesian copula approaches.

We outline some studies of the Bayesian methods in copulas modeling as follows. In terms of the Bayesian inference of copula parameters, some typical studies include Pitt et al. (2006), where the Gaussian regression copula was estimated by Markov Chain Monte Carlo (MCMC). Almeida and Czado (2012) studied the estimation problems of time-varying copula models using Bayesian computational approaches. Smith et al. (2012) constructed the skew-t copulas using the Bayesian framework. Smith and Klein (2021) estimated the regression copula using Hamiltonian MCMC and the method of variational inference, Deng et al. (2023) provided scalable variational Bayes approaches for skew-t copulas.

In terms of Bayesian copula selections, Some previous works include Huard et al. (2006), where the author treats the copula parameters as a nuisance and selects the copula with the highest posterior probability. Their method is free from estimating the copula parameters. Silva and Lopes (2008) proposed to select the model using deviance information criteria (DIC), expected Bayes information criteria (EBIC), and expected Akaike information criteria (EAIC). They also pointed out the importance of the joint estimation of copula parameters using the Bayesian approach from the perspective of considering parameter dependence. Their work can be viewed as the Bayesian version of the popular frequentist AIC (BIC) approaches.

Nevertheless, as mentioned previously, the construction of mixture copulas under the setting of Bayesian frameworks lacks thorough consideration in the literature, and the previous discussed Bayesian methods of copula estimation and selection are not directly suitable for the mixture copula models. Moreover, the applications of mixture copula models are relatively limited in the literature. The main topics of this thesis are therefore to discuss mixture copula estimation and selection problems using Bayesian frameworks and we also intend to extend the relevant mixture copula applications in the field of finance and data science.

There is a long history of debate regarding whether the Bayesian approach or the frequentist approach is the best in terms of statistical modeling. There is no answer yet. We study primarily the Bayesian methods here, mainly due to its advantage in terms of automatic model selection, estimation and inference while simultaneously sampling mechanism, especially when empowered by the MCMC scheme. Meanwhile, the Bayesian viewpoints are less considered in the copula field when compared with the classic approach. However, as the famous sentence by George E.P.Box stated (Box et al., 2011), *All models are wrong, but some are useful.* We do not try to compare the general mindset of the statistical approach as all methodologies have their pros and cons, but use them as we find them effective in solving our problems.

1.1.2 Overall structure of the thesis

The structure of the rest of the thesis is organized as follows. The remainder of the first chapter is mainly concerned with the general introduction of the basic definitions and methodologies used in later chapters, including some fundamental definitions and theorems regarding the copula theory, different types of parametric copulas, and the methodology of Bayesian

computation.

The second chapter proposes the Bayesian method that would enable us to select and estimate the correct finite mixture copulas simultaneously. This is done by first overfitting the model and then conducting the Bayesian estimations. The MCMC algorithm as well as the EM algorithm, are proposed for implementing the Bayesian method. We verify the correctness of our approach by numerical simulations and the real data analysis is done by studying the dependencies among three major financial markets. The methodology is particularly useful when applied to the mixture copula with heterogeneous components.

The third chapter proposes an infinite mixture Student t copula model using a nonparametric Bayesian approach. We establish a corresponding MCMC sampler for this model. In contrast to the normal mixture model, our proposed model is more suitable for data exhibiting tail dependence, which is frequently encountered in financial risk management. We evaluated the proposed algorithm through theoretical simulations and real data analysis. Parameter estimation results from the simulations demonstrate that our approach is competitive when compared to the standard maximum likelihood estimation method. The analysis of real financial data supports the validity of our approach and highlights the importance of applying a t copula in the presence of heavy tails.

In the fourth chapter, we develop the copula Bayesian infinite hidden Markov model (copula-iHMM) by extending the theory of the third chapter to the correlated data. This is in particular very useful in terms of modeling the financial data as they are time-correlated instead of the i.i.d case discussed in the previous chapter.

In the fifth chapter, we introduce the basics of imbalance learning and study the statistical properties of empirical ROC-AUC, which is the most commonly used metric in measuring the performance of the classifiers when applied to the imbalanced learning task. We demonstrate both analytically and empirically that the empirical AUC estimation could be highly volatile in many circumstances when applied to an imbalanced dataset. To be more specific, we have proved that under some frequently encountered circumstances, variances of the empirical AUC estimator increase with the imbalanced level of the dataset. Hence, under the imbalanced setting, variances could be high. Furthermore, we conduct several simulations and experiments to solidify our findings. Therefore, when the empirical ROC-AUC is used to summarize the classifier's performance, especially when the dataset presents a high-class imbalance, we must include the information on the variance to make our finding reliable.

In the sixth chapter, we apply the Bayesian approach to estimate the mixture copula with discrete margins, we further apply our models to solve the class imbalanced problems

by oversampling the mixture copulas. The methodology makes it possible to learn and sample from the data set with the discrete and continuous features existing simultaneously. On the other hand, the discreetness of factors in a data set is not naturally considered for the classic SMOTE algorithm of imbalance learning, and classic random oversampling is simply performed by generating the already existing points, which do not give any new information to the classifiers and is easy to overfit. Copula methods enable us to generate new points with the correlation structure memorized by learning from the training set. Hence, the overfitting problems are reduced. Experiments with synthetic and real data are done in the chapter following the introduction of the methodology. The outcomes show the validity of the approach when compared with the benchmark methods.

Finally, the seventh chapter is devoted to the conclusion remarks and possible future research directions.

1.2 Copula functions

Copula functions are widely applied in the modeling of multidimensional random vectors. For a d-dimensional application, copula functions are defined over the d-hypercube $[0,1]^d$, and they satisfy the definition of probability distributions. Following the definition in Nelsen (2006); McNeil et al. (2015), the copula function $C(\cdot) : [0,1]^d \to [0,1]$ is a d-variate distribution function that satisfies the following conditions.

1. The copula function is non-decreasing in every dimension. In particular, for any dimension k and $C(u_1, u_2, \ldots, u_k, \ldots, u_d) \geq C(u_1, u_2, \ldots, v_k, \ldots, u_d)$ if $u_k \geq v_k$.

2. $C(u_1, u_2, \ldots, u_d) = 0$ if one or more arguments among d dimensions equal to zero. If all arguments except u_k equal to 1, we have $C(1, 1, \ldots, u_k, \ldots, 1) = u_k$.

3. Define the subtraction operator

$$\Delta_{v_k}^{u_k} C(u_1, u_2, \ldots, u_d) = C(u_1, u_2, \ldots, u_k, \ldots, u_d) - C(u_1, u_2, \ldots, v_k, \ldots, u_d).$$

   For Cartesian product $\otimes_{i=1}^{d}[v_i, u_i] \subset [0,1]^d$, we have

$$\Delta_{v_d}^{u_d} \ldots \Delta_{v_2}^{u_2} \Delta_{v_1}^{u_1} C(\cdot) \geq 0.$$

   In summary, as stated by McNeil et al. (2015), a copula function is a multivariate

distribution function with uniform margins.

Sklar's theorem (Sklar, 1959) enables the application of copula functions into applied modelings. It states that any multivariate random vector $(X_1, X_2, \ldots, X_d)$ with distribution $F(x_1, x_2, \ldots, x_d)$ can be constructed from its marginal distributions $F_1, F_2, \ldots, F_d$ and copula function C. Particularly, for any d-dimensional distribution, there exists a copula $C : [0,1]^d \to [0,1]$ such that

$$F(x_1, x_2, \ldots, x_d) = C(F_1(x_1), F_2(x_2), \ldots, F_d(x_d)). \tag{1.1}$$

The copula function is unique if the x_j are all continuous. If some of the x_j are discrete, the copula is unique in the range of the marginal distributions. Sklar's Theorem allows us to form a multivariate distribution by linking the underlying univariate marginal distributions with a copula function. The copula therefore gives us the full description of the relation between variables, which is more informative than single correlation statistics.

For an absolutely continuous distribution $F(\cdot)$, the relationship between the distribution and copula densities can be obtained by differentiating both sides of (1.1) such that

$$\frac{\partial^d F(x_1, x_2, \ldots, x_d)}{\partial x_1 \partial x_2 \cdots \partial x_d} = f(x_1, x_2, \ldots, x_d) = c(F_1(x_1), F_2(x_2), ..., F_d(x_d)) \prod_{i=1}^{d} f_i(y_i),$$

where $c(\cdot)$ is the density of the copula $C(\cdot)$.

1.3 Parametric copula families

1.3.1 Elliptical copulas

The elliptical copulas are one of the most common choices for modeling the dependence structures among variables, especially in high dimensional settings (Smith and Loaiza-Maya, 2022). From the Sklar theorem, copulas are of the form

$$C(u_1, u_2, \ldots, u_d) = F(F_1^{-1}(u_1), F_2^{-1}(u_2), \ldots, F_d^{-1}(u_d)). \tag{1.2}$$

Since the elliptical distribution is closed under the marginalization, we can therefore get the corresponding parametric copula implicitly defined by (1.2). For example, by inverting the marginal of the standard multivariate normal distribution, we obtain the normal copula,

which is

$$C_{\boldsymbol{P}}(u_1, u_2, \ldots, u_d) = \int_{-\infty}^{\phi^{-1}(u_d)} \cdots \int_{-\infty}^{\phi^{-1}(u_1)} \left((2\pi)^d |\boldsymbol{P}|\right)^{-1/2} \exp(-\frac{1}{2}\boldsymbol{x}'\boldsymbol{P}^{-1}\boldsymbol{x}) d\boldsymbol{x}, \tag{1.3}$$

where $\boldsymbol{P}$ is the positive definite correlation matrix, and $\boldsymbol{x} = (\phi^{-1}(u_1), \phi^{-1}(u_2), \ldots, \phi^{-1}(u_d))'$ with $\phi(\cdot)$ being the quantile function. On the other hand, taking the same action to the multivariate t distribution yields the t copula,

$$C_{v,\boldsymbol{P}}(u_1, u_2, \ldots, u_d) = \int_{-\infty}^{t_v^{-1}(u_d)} \cdots \int_{-\infty}^{t_v^{-1}(u_1)} \frac{\Gamma(\frac{1}{2}(v+d))/\Gamma(\frac{v}{2})}{\sqrt{(\pi v)^d |\boldsymbol{P}|}} (1 + \frac{\boldsymbol{y}'\boldsymbol{P}^{-1}\boldsymbol{y}}{v}) d\boldsymbol{y}, \tag{1.4}$$

$t_v^{-1}(u_1)$ is the quantile function of the univariate standard t distribution with v degree of freedom and $\boldsymbol{y} = (t_v^{-1}(u_1), t_v^{-1}(u_2), t_v^{-1}(u_3), \ldots, t_v^{-1}(u_d))'$, $\boldsymbol{P}$ is a correlation matrix. The respective copula density $c(\cdot)$ can be obtained due to the differentiation

$$f(F_1^{-1}(u_1), F_2^{-1}(u_2), \ldots, F_d^{-1}(u_d)) = c(u_1, u_2, \ldots, u_d) \prod_{j=1}^{d} f_j(F_j^{-1}(u_j)). \tag{1.5}$$

One potential advantage of using the t copula (1.4) over the normal copula is its ability to model the tail dependence. That is, we wish to measure the degree of dependence on the upper tail ρ_u and on the lower tail ρ_l,

$$\rho_l = \lim_{u \to 0^+} \mathcal{P}(X_2 \leq F_2^{-1}(u) \mid X_1 \leq F_1^{-1}(u)) = \lim_{u \to 0^+} \frac{C(u,u)}{u}$$
$$\rho_u = \lim_{u \to 1^-} \mathcal{P}(X_2 > F_2^{-1}(u) \mid X_1 > F_1^{-1}(u)) = \lim_{u \to 1^-} \frac{1 - 2u + C(u,u)}{u}.$$

For the two dimensional copula $d = 2$, we have $\rho_l = \rho_u = 0$ for the normal copula but for the t copula with v degree of freedom we have

$$\rho_l = \rho_u = 2F_{v+1;t}(-\sqrt{(v+1)(1-c)/(1+c)}),$$

where $F_{v+1;t}(\cdot)$ is the t distribution function with v degree of freedom. One criticism of the elliptical copula families is their symmetric property $c(\boldsymbol{u}) = c(1 - \boldsymbol{u})$, which might be unrealistic for modeling the asymmetrical correlation that often occurs in the financial market (Ang and Chen, 2002). Therefore, many authors have proposed the skewed elliptical copula. Smith et al. (2012) proposed the skew t copula and the estimation of the parameters is

performed by MCMC. Wu et al. (2014) uses a nonparametric Bayesian approach to construct infinite mixture skew normal copula. Wei et al. (2019) explored some theoretical properties of the skew-normal copula. Alternatively, Archimedean families of copulas are another solution for the issue.

1.3.2 Skew-normal distribution and copula

We consider variables from the skew-normal distribution (Azzalini, 1985; Azzalini and Valle, 1996). Suppose that we have two normal variables X_0 and X_j, X_0 is standard normal. Then we define the corresponding *skew-normal* random variable Y_j *via*

$$Y_j = \delta_j |X_0| + \sqrt{1-\delta_j^2} X_j, \tag{1.6}$$

where $\delta_j \in (-1, 1)$ is a given *skewness parameter*. We denote this as $Y_j \sim \text{SkewNormal}(\lambda_j)$, where

$$\lambda_j = \frac{\delta_j}{\sqrt{1-\delta_j^2}}. \tag{1.7}$$

The resulting distribution for such a variable is given by (Azzalini and Valle, 1996, eq. 1.1)

$$f_j(y_j) = \sqrt{\frac{2}{\pi}} e^{-y_j^2/2} \Phi(\lambda_j y_j). \tag{1.8a}$$

Note that when $\delta_j = 0$, all skew-normal results reduce to the normal case. $\Phi(u)$ is the cumulative density function of the standard normal.

We now extend this result to d dimensions by considering the following $d+1$-multivariate normal random vector:

$$\begin{pmatrix} X_0 \\ \mathbf{X} \end{pmatrix} \sim N_{d+1} \left(\mathbf{0}, \begin{pmatrix} 1 & \mathbf{0}^T \\ \mathbf{0} & \boldsymbol{P} \end{pmatrix} \right),$$

where $\mathbf{X}$ is the random vector and $\boldsymbol{P}$ is its correlation matrix. Jointly, the density of the d-multivariate skew normal is written as Azzalini (1985); Azzalini and Valle (1996)

$$f(\mathbf{y}) = 2(2\pi)^{-d/2} |\boldsymbol{P}_\delta|^{-1/2} \exp\left(-\frac{1}{2} \mathbf{y}^T \boldsymbol{P}_\delta^{-1} \mathbf{y} \right) \Phi(\boldsymbol{\alpha}^T \mathbf{y}), \tag{1.8b}$$

where

$$
\begin{aligned}
\mathbf{y}^T &= (y_1, y_2, \ldots, y_d) \\
\boldsymbol{\delta}^T &= (\delta_1, \delta_2, ..., \delta_d), \\
\boldsymbol{\lambda}^T &= (\lambda_1, \lambda_2, ..., \lambda_d), \\
\Lambda &= (I_d - \mathrm{diag}(\boldsymbol{\delta})^2)^{1/2}, \\
\boldsymbol{P}_\delta &= \Lambda(\boldsymbol{\lambda}\boldsymbol{\lambda}^T + \boldsymbol{P})\Lambda, \\
\boldsymbol{\alpha} &= \Lambda^{-1}\boldsymbol{P}^{-1}\boldsymbol{\lambda}(\boldsymbol{\lambda}^T\boldsymbol{P}^{-1}\boldsymbol{\lambda} + 1)^{-1/2}.
\end{aligned}
$$

When $\boldsymbol{\delta} = \mathbf{0}$ the skew-normal results degenerate to the standard joint Gaussian distribution. Hence, we are able to represent more complex, especially asymmetrical data distributions using the skewed family.

Therefore, rewriting our results to obtain the multivariate skew-normal copula as in (Wu et al., 2014; Wei et al., 2019), we have

$$
c_{\mathrm{SN}}(u_1, u_2, \ldots, u_d) = \frac{f\left(F_1^{-1}(u_1), F_2^{-1}(u_2) \ldots, F_d^{-1}(u_d)\right)}{\prod_{j=1}^{d} f_j\left(F_j^{-1}(u_j)\right)}, \tag{1.9}
$$

where the subscript "SN" refers to "skew-normal". Here the forms of f are in (1.8), and we may use the integral of (1.8a) to obtain F_j^{-1} (numerically).

1.3.3 Archimedean copulas

Archimedean copulas have been widely researched and applied in the field of credit risk modeling (McNeil et al., 2015). They can be constructed by satisfying the following linear additive property, that is,

$$
\varphi^{-1}(C(u_1, u_2, \ldots, u_d)) = \sum_{i=1}^{d} \varphi^{-1}(u_i), \tag{1.10}
$$

where $\varphi(\cdot) : [0, +\infty) \to [0, 1]$ is usually called the Archimedean copula generator satisfying convexity, continuity, and completely monotonicity with $\varphi(0) = 1$ and $\lim_{t\to\infty} \varphi(t) = 0$. The generator with such properties can be derived from the Laplace transform of the positive

random variable X with its distribution function having $F_X(0) = 0$,

$$\varphi(t) = \int_0^\infty \exp(-tx) dF_X(x).$$

Taking different forms of $\varphi(t)$ yields different Archimedean families of copulas, a comprehensive table can be found in the textbooks (Nelsen, 2006, Table 4.1). We give several copulas that we will use with their generators in Table 1.1, and the corresponding distribution function is $C(u_1, \ldots, u_d) = \varphi(\sum_{i=1}^d \varphi^{-1}(u_i))$.

Table 1.1: Some classic Archimedean family of copulas.

Copula type	$\varphi(t)$	θ range
Frank	$\theta^{-1} \ln\left(\frac{1}{1+(\exp(-\theta-t)-\exp(-t))}\right)$	$\mathbb{R} \setminus \{0\}$
Gumbel	$\exp\left(-t^{\theta^{-1}}\right)$	$[1, +\infty)$
Clayton†	$(1+\theta t)_+^{-\theta^{-1}}$	$(0, +\infty)$

† We denote $(\cdot)_+ := \max(\cdot, 0)$

One noticeable property of the Archimedean copula is its exchangeability. That is, for any permutation $\sigma(i)$ of $\{1, 2, \ldots, d\}$, we have $C(u_1, u_2, \ldots, u_d) = C(u_{\sigma(1)}, u_{\sigma(2)}, \ldots, u_{\sigma(d)})$. This characteristic would be attractive for some applications, such as portfolio default modeling in the credit market. However, for the more general purpose, it might be undesirable when we have the copula dimension $d \geq 3$ since this implies that the connection between variables is assumed to be homogeneous. Some improvement has been made on this problem, including nonexchangeable copulas named asymmetric Archimedean copulas (Genest et al., 1998; McNeil et al., 2015)

$$C^\gamma(u_1, u_2, \ldots, u_d) = \left(\prod_{j=1}^d u_j^{\gamma_i}\right) C(u_1^{\gamma_1}, u_2^{\gamma_2}, \ldots, u_d^{\gamma_d}).$$

Otherwise, some amendment in (1.10) yields so-called nonexchangeable nested Archimedean copula (Joe, 1997).

Different Archimedean copulas can model different kinds of dependence, Figure.1.1 gives us a plot of 4 different types of copulas. In particular, normal and Frank copulas are symmetric in the sense of $c(1-u_1, 1-u_2) = c(u_1, u_2)$ with 0 tail dependence but the Frank copula was, in addition, proved to be radial symmetric (Frank, 1979). Gumbel copula is able

to depict the extreme upper tail dependence with $p_u = -2^{\theta_{\text{Gumbel}}^{-1}} + 2$ but $p_l = 0$. Oppositely, the Clayton copula is able to describe the extreme lower tail dependence with $p_l = 2^{-\theta_{\text{Clayton}}^{-1}}$ for $\theta_{\text{Clayton}} > 0$ but $\rho_u = 0$.

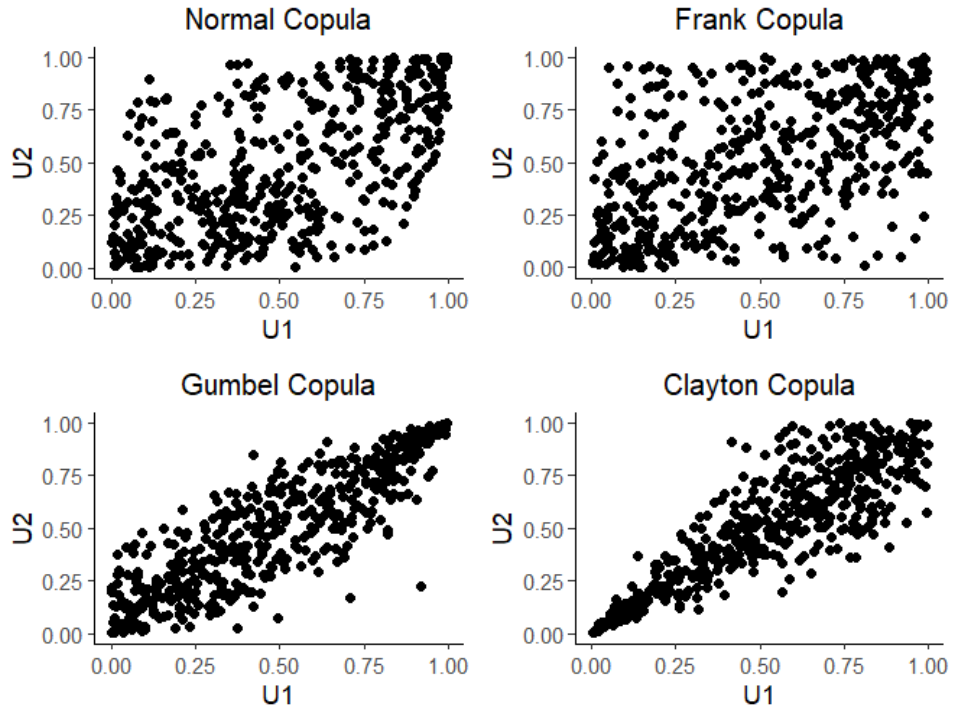

Figure 1.1: A plot of different families of Copula with 2000 points, $\theta = 0.6, 5, 3, 3$ for Normal, Clayton, Frank, Gumbel Copula respectively.

1.3.4 Mixture copulas

For complex data structures in many real-life applications, a single parametric copula might be insufficient to capture all important features when performing analysis. It is therefore motivated to introduce finite mixture copulas,

$$C_{\text{mix}} = \sum_{i=1}^{K} w_i C^{(i)}, \quad \sum_{i=1}^{K} w_i = 1, \ w_i \geq 0 \ \forall i = 1, 2, \ldots, K. \tag{1.11}$$

Where $C^{(i)}$ refers to a single copula component and K is usually a predefined hyperparameter. In classic finite mixture models' discussion, $C^{(i)}$ are from the same parametric family

(McLachlan et al., 2019). However, mixture models with heterogeneous components are also common in the copula literature, see (Hu, 2006; Arakelian and Karlis, 2014) for examples. It is straightforward to check (1.11) to satisfy the definition of the copula function.

1.4 MCMC methods of Bayesian computation

In this section, we introduce the basics of Bayesian statistics and its computational methods. Markov Chain Monte Carlo (MCMC) is given specific attention. This will be the primary estimation and inference method we use to study our copula functions.

1.4.1 Bayesian statistics and motivation of MCMC

In the realm of Bayesian statistics, we treat the parameters $\boldsymbol{\theta} \in \Theta$ to be estimated with uncertainty. That is, instead of directly applying the data to optimize the likelihood of models, we first utilize some *prior* knowledge of the models proposed and describe them as the probability distributions $p(\boldsymbol{\theta})$. We hence apply our data $\mathcal{D} = \{X_1, X_2, \ldots, X_n\}$ and models to infer the *posterior* probability distribution adjusted from our *prior*. Mathematically speaking, the mindset of this process can be expressed as the classic Bayesian formula.

$$p(\boldsymbol{\theta} \mid \mathcal{D}) = \frac{p(\mathcal{D} \mid \boldsymbol{\theta})p(\boldsymbol{\theta})}{\int p(\mathcal{D} \mid \boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}}. \tag{1.12}$$

For the most frequently used examples, such as the posterior distributions from the i.i.d Gaussian distribution. Analytical forms of the posterior mean μ and covariance matrix Σ can be obtained by using conjugate priors, see Bishop and Nasrabadi (2006) for detailed computation. On the other hand, the analytical forms of the denominator for (1.12) are often unavailable for more complex distributions. The information known to us is merely the numerator. That is,

$$p(\boldsymbol{\theta} \mid \mathcal{D}) \propto p(\mathcal{D} \mid \boldsymbol{\theta})p(\boldsymbol{\theta}). \tag{1.13}$$

We need to come up with some computational approaches such that the posterior distribution of (1.12) can be evaluated. One straightforward idea is to evaluate the denominator using

numerical integration. such that

$$\hat{p}(\boldsymbol{\theta} \mid \mathcal{D}) = \frac{p(\mathcal{D} \mid \boldsymbol{\theta})p(\boldsymbol{\theta})}{\sum_i p(\mathcal{D} \mid \boldsymbol{\theta}_i)p(\boldsymbol{\theta}_i)\boldsymbol{\Delta\theta}_i}. \tag{1.14}$$

Unfortunately, the usual complexity of such approaches is $O(N^d)$ where N is the scale of precision, and d is the dimension of parameter space. The exponential term in the complexity indicates that these kinds of methods suffer from the curse of dimension (Köppen, 2000) when the dimension d becomes large. The tasks would very soon become computationally prohibitive as $d \to \infty$.

To overcome the dependence between the dimension of integration and the complexity of numerical approximation. Monte Carlo methods are often sought. That is, we can approximate the posterior distribution as

$$\tilde{p}(\boldsymbol{\theta} \mid \mathcal{D}) = \frac{p(\mathcal{D} \mid \boldsymbol{\theta})p(\boldsymbol{\theta})}{\frac{1}{M}\sum_{i=1}^{M} p(\mathcal{D} \mid \boldsymbol{\theta}_i)}. \tag{1.15}$$

Where $\{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_i, \ldots, \boldsymbol{\theta}_M\}$ is a finite collection of M points sampled from the prior $p(\boldsymbol{\theta})$. The method needs M points for one evaluation, independent from the dimensionality d, and the prior is usually some well-studied distributions. However, the prior and the resultant posterior are usually far distant for tasks where the data are informative. This means the exploration of parameter spaces using the prior $p(\boldsymbol{\theta})$ is often very slow and inefficient. More sophisticated approaches are necessary in this regard.

In the following, we introduce Markov Chain Monte Carlo (MCMC) simulation approaches. They are types of approaches where we simulate data from Markov chains, and we expect the limiting distributions of Markov chains to be our posteriors. By proposing a reasonable transition kernel, we expect samples from the chain are much well guided than the random samples from priors.

1.4.2 Basic of MCMC methods

A stochastic process $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots)$ is a Markov chain if for any set $\mathcal{B}$ with non-zero measure, we have

$$P(\boldsymbol{\theta}_n \in \mathcal{B} \mid \boldsymbol{\theta}_{n-1}, \boldsymbol{\theta}_{n-2}, \ldots, \boldsymbol{\theta}_1) = P(\boldsymbol{\theta}_n \in \mathcal{B} \mid \boldsymbol{\theta}_{n-1}).$$

That is, the distribution of the next sample is only dependent on the previous one. In this thesis, we only consider homogenous Markov chains such that the *transition kernel*

$P(\boldsymbol{\theta}_n \in \mathcal{B} \mid \boldsymbol{\theta}_{n-1})$ takes the same form for all $n = 2, 3, 4, \ldots$.

There might exist an invariant distribution for the Markov chain. That is, for any set with non-zero measure, we have an invariant distribution $\Pi(\boldsymbol{\theta})$ and

$$\int P(\boldsymbol{\theta}_n \in \mathcal{B} \mid \boldsymbol{\theta}_{n-1}) d\Pi(\boldsymbol{\theta}_{n-1}) = \Pi(\boldsymbol{\theta}_n \in \mathcal{B}).$$

In this case, the chain will remain in the invariant distribution once the previous state is sampled from it.

We want to sample points $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_n, \ldots$ from the Markov chain so that when they reach stationary, they are sampled from the posterior distribution $p(\boldsymbol{\theta} \mid \mathcal{D})$. More formally, by proposing the appropriate transition kernel $P(\boldsymbol{\theta}_n \in \mathcal{B} \mid \boldsymbol{\theta}_{n-1})$, the outcome we hope to reach is therefore

$$N^{-1} \sum_{j=1}^{N} g(\boldsymbol{\theta}_j) \to \int g(\boldsymbol{\theta}) p(\boldsymbol{\theta} \mid \mathcal{D}) d\boldsymbol{\theta} \quad N \to \infty \tag{1.16}$$

almost surely for any $g(\cdot)$ that is $L^1(\Theta)$ integrable with respect to the posterior measure and the initial guess $\boldsymbol{\theta}_1$ can be arbitrary (in the almost surely sense) within the support of parameter distribution.

This is the result of Tierney (1994), which states that if a Markov chain has the invariant distribution π and it is aperiodic, π-irreducible as well as satisfying some forms of recurrent conditions, it has the unique stationary distribution π such that $P(\boldsymbol{\theta}_n \in \mathcal{B} \mid \boldsymbol{\theta}_1) \to \pi(\boldsymbol{\theta} \in \mathcal{B})$ as $n \to \infty$ for $\boldsymbol{\theta}_1 \in \Theta$ almost surely and (1.16) is satisfied. See Geweke et al. (2011) for extended discussion.

Therefore, as long as we are able to construct such chains with the invariant distribution to be the posterior $p(\boldsymbol{\theta} \mid D)$. We would be able to reach the stationary distribution $p(\boldsymbol{\theta} \mid D)$ when we sample the chain sufficiently long.

The convergence speed of chains needs to be assessed case-by-case in the MCMC applications as there are currently no comprehensive results available. The most straightforward way is by inspecting the trace plot to see if it is unstable. Some numerical diagnostics are also available such as Gelman-Rubin diagnostic (Gelman and Rubin, 1992), Geweke diagnostic (Geweke, 1991). When the stationary is detected, we often apply (1.16) with truncation. That is $(N - M - 1)^{-1} \sum_{j=M+1}^{N} g(\boldsymbol{\theta}_j)$ is usually considered to discard the first M non-stationary points called burn-in stage.

1.4.3 Metroplis-Hastings algorithm and its variants

The Metropolis-Hasting algorithm (M-H) and its variant are the most popular MCMC approaches. In order to sample from the posterior distribution $p(\boldsymbol{\theta} \mid \mathcal{D})$, it is necessary to propose the Markov chain such that the invariant distribution is the posterior as we argued in the last section. This is realized by first sampling a new point $\boldsymbol{\theta}^*$ from a *proposal distribution* $q(\cdot)$, and then calculating the acceptance rate α to see if we should accept $\boldsymbol{\theta}^*$ so that $\boldsymbol{\theta}_t = \boldsymbol{\theta}^*$ in the chain or just keep last point so $\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1}$. The steps are the following.

1. **Initialize** starting state $\boldsymbol{\theta}_0$.
2. **For** $t = 1$ to N **do**
   (a) Propose a new state $\boldsymbol{\theta}^*$ from the proposal distribution $q(\boldsymbol{\theta}^*|\boldsymbol{\theta}_{t-1})$.
   (b) Compute acceptance probability:

$$\alpha(\boldsymbol{\theta}^*, \boldsymbol{\theta}_{t-1}) = \min\left(1, \frac{p(\boldsymbol{\theta}^*|\mathcal{D})}{p(\boldsymbol{\theta}_{t-1}|\mathcal{D})} \times \frac{q(\boldsymbol{\theta}_{t-1}|\boldsymbol{\theta}^*)}{q(\boldsymbol{\theta}^*|\boldsymbol{\theta}_{t-1})}\right).$$

   (c) Draw $u \sim \text{Uniform}(0, 1)$.
   (d) **If** $u < \alpha(\boldsymbol{\theta}^*, \boldsymbol{\theta}_{t-1})$ **then**
      i. Accept the proposed state: $\boldsymbol{\theta}_t = \boldsymbol{\theta}^*$.
   (e) **Else**
      i. Reject the proposed state: $\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1}$.
3. **Output**: $\{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_N\}$.

In this regard, we can see that the density of the transition kernel is of the mixture form such that

$$k(\boldsymbol{\theta}_t \mid \boldsymbol{\theta}_{t-1}) = \alpha(\boldsymbol{\theta}_t, \boldsymbol{\theta}_{t-1}) q(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{t-1}) + \big(1 - \int \alpha(\boldsymbol{\theta}^*, \boldsymbol{\theta}_{t-1}) q(\boldsymbol{\theta}^*|\boldsymbol{\theta}_{t-1}) d\boldsymbol{\theta}^*\big) \delta_{\boldsymbol{\theta}_{t-1}}(\boldsymbol{\theta}_t). \quad (1.17)$$

The M-H transition kernel (1.17) can be verified to satisfy the reversible condition

$$p(\boldsymbol{\theta}^* \mid \mathcal{D}) k(\boldsymbol{\theta} \mid \boldsymbol{\theta}^*) = p(\boldsymbol{\theta} \mid \mathcal{D}) k(\boldsymbol{\theta}^* \mid \boldsymbol{\theta}). \quad (1.18)$$

A sufficient (but not necessary) condition to have the posterior distribution $p(\boldsymbol{\theta} \mid \mathcal{D})$ being the invariant distribution of the Markov chain. This is because (1.18) implies

$$p(\boldsymbol{\theta}^* \mid \mathcal{D}) = \int p(\boldsymbol{\theta} \mid \mathcal{D}) k(\boldsymbol{\theta}^* \mid \boldsymbol{\theta}) d\boldsymbol{\theta}$$

by integrating both side with respect to $\boldsymbol{\theta}$. Furthermore, for the kernel of the M-H (1.17),

$$\begin{aligned}
&p(\boldsymbol{\theta}^* \mid \mathcal{D}) k(\boldsymbol{\theta} \mid \boldsymbol{\theta}^*) = \alpha(\boldsymbol{\theta}, \boldsymbol{\theta}^*) q(\boldsymbol{\theta}|\boldsymbol{\theta}^*) p(\boldsymbol{\theta}^* \mid \mathcal{D}) + \big(1 - \int \alpha(\boldsymbol{\theta}, \boldsymbol{\theta}^*) q(\boldsymbol{\theta}|\boldsymbol{\theta}^*) d\boldsymbol{\theta}\big) \delta_{\boldsymbol{\theta}^*}(\boldsymbol{\theta}) p(\boldsymbol{\theta}^* \mid \mathcal{D}) \\
&= \min\left(q(\boldsymbol{\theta}|\boldsymbol{\theta}^*) p(\boldsymbol{\theta}^* \mid \mathcal{D}), p(\boldsymbol{\theta}|\mathcal{D}) \times q(\boldsymbol{\theta}^*|\boldsymbol{\theta})\right) + \big(1 - \int \alpha(\boldsymbol{\theta}^*, \boldsymbol{\theta}) q(\boldsymbol{\theta}^*|\boldsymbol{\theta}) d\boldsymbol{\theta}\big) \delta_{\boldsymbol{\theta}}(\boldsymbol{\theta}^*) p(\boldsymbol{\theta} \mid \mathcal{D}) \\
&= p(\boldsymbol{\theta}|\mathcal{D}) q(\boldsymbol{\theta}^* \mid \boldsymbol{\theta}) \min\left(\frac{p(\boldsymbol{\theta}^*|\mathcal{D})}{p(\boldsymbol{\theta}|\mathcal{D})} \times \frac{q(\boldsymbol{\theta}|\boldsymbol{\theta}^*)}{q(\boldsymbol{\theta}^*|\boldsymbol{\theta})}, 1\right) + \big(1 - \int \alpha(\boldsymbol{\theta}^*, \boldsymbol{\theta}) q(\boldsymbol{\theta}^*|\boldsymbol{\theta}) d\boldsymbol{\theta}\big) \delta_{\boldsymbol{\theta}}(\boldsymbol{\theta}^*) p(\boldsymbol{\theta} \mid \mathcal{D}) \\
&= \alpha(\boldsymbol{\theta}^*, \boldsymbol{\theta}) q(\boldsymbol{\theta}^*|\boldsymbol{\theta}) p(\boldsymbol{\theta} \mid \mathcal{D}) + \big(1 - \int \alpha(\boldsymbol{\theta}^*, \boldsymbol{\theta}) q(\boldsymbol{\theta}^*|\boldsymbol{\theta}) d\boldsymbol{\theta}\big) \delta_{\boldsymbol{\theta}}(\boldsymbol{\theta}^*) p(\boldsymbol{\theta} \mid \mathcal{D}) = p(\boldsymbol{\theta} \mid \mathcal{D}) k(\boldsymbol{\theta}^* \mid \boldsymbol{\theta}).
\end{aligned}$$

The second part of the addition are equivalent when we interchange the position of $\boldsymbol{\theta}$ and $\boldsymbol{\theta}^*$ because it is 0 unless $\boldsymbol{\theta} = \boldsymbol{\theta}^*$.

1.4.4 Block-wise M-H sampling

The M-H algorithm is sometimes easier to implement in blocks. This often occurs when we want to take some latent variables into consideration so that our modeling becomes clearer. Mathematically, suppose our goal is to sample the distribution of $\boldsymbol{\theta}$, it might be more straightforward if we can add latent vectors $\boldsymbol{z}, \boldsymbol{k}$ such that

$$p(\boldsymbol{\theta} \mid \mathcal{D}) = \int p(\boldsymbol{\theta}, \boldsymbol{z}, \boldsymbol{k} \mid \mathcal{D}) d\boldsymbol{z} d\boldsymbol{k}$$

and $p(\boldsymbol{\theta} \mid \boldsymbol{z}, \boldsymbol{k}, \mathcal{D})$ is easier to sample. We can, in this case, use the block-wise sampling strategy to construct the Markov chain for $p(\boldsymbol{\theta}, \boldsymbol{z}, \boldsymbol{k} \mid \mathcal{D})$ and the target $\boldsymbol{\theta}$ is the marginal of the stationary distribution which can be collected by simply ignoring the latent dimensions of samples.

Define the partition of $\boldsymbol{\theta}$ into B blocks: $\boldsymbol{\theta} = (\boldsymbol{\theta}^1, \boldsymbol{\theta}^2, \ldots, \boldsymbol{\theta}^B)$. We give the generalized algorithm of block-wise sampling as follows.

1. **Initialize** starting state $\boldsymbol{\theta}_0$ inside the relevant domain.
2. **For** $t = 1$ to N **do**

(a) **For** $b = 1$ to B **do**

i. Propose a new block state $\boldsymbol{\theta}^{b*}$ from the proposal distribution $q_b(\boldsymbol{\theta}^{b*}|\boldsymbol{\theta}^b_{t-1}, \boldsymbol{\theta}^{<b}_t, \boldsymbol{\theta}^{>b}_{t-1})$, where $\boldsymbol{\theta}^{<b}_t$ represents the blocks updated before block b at iteration t, and $\boldsymbol{\theta}^{>b}_{t-1}$ represents the blocks not yet updated at iteration t.

ii. Compute block acceptance probability:

$$\alpha_b(\boldsymbol{\theta}^{b*}, \boldsymbol{\theta}^b_{t-1}) = \min\left(1, \frac{p(\boldsymbol{\theta}^{b*}, \boldsymbol{\theta}^{<b}_t, \boldsymbol{\theta}^{>b}_{t-1}|\mathcal{D})}{p(\boldsymbol{\theta}^b_{t-1}, \boldsymbol{\theta}^{<b}_t, \boldsymbol{\theta}^{>b}_{t-1}|\mathcal{D})} \times \frac{q_b(\boldsymbol{\theta}^b_{t-1}|\boldsymbol{\theta}^{b*}, \boldsymbol{\theta}^{<b}_t, \boldsymbol{\theta}^{>b}_{t-1})}{q_b(\boldsymbol{\theta}^{b*}|\boldsymbol{\theta}^b_{t-1}, \boldsymbol{\theta}^{<b}_t, \boldsymbol{\theta}^{>b}_{t-1})}\right).$$

iii. Draw $u^b \sim \text{Uniform}(0, 1)$.

iv. **If** $u^b < \alpha_b(\boldsymbol{\theta}^{b*}, \boldsymbol{\theta}^b_{t-1})$ **then**

A. Accept the proposed block state: $\boldsymbol{\theta}^b_t = \boldsymbol{\theta}^{b*}$.

v. **Else**

A. Reject the proposed block state: $\boldsymbol{\theta}^b_t = \boldsymbol{\theta}^b_{t-1}$.

3. **Output**: $\{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_N\}$.

For some blocks, the analytical distribution of $p(\boldsymbol{\theta}^{b*} \mid \boldsymbol{\theta}^{<b}_t, \boldsymbol{\theta}^{>b}_{t-1}\mathcal{D})$ might be available. In this case, we will usually let $q_b(\boldsymbol{\theta}^{b*}|\boldsymbol{\theta}^b_{t-1}, \boldsymbol{\theta}^{<b}_t, \boldsymbol{\theta}^{>b}_{t-1}) := p(\boldsymbol{\theta}^{b*} \mid \boldsymbol{\theta}^{<b}_t, \boldsymbol{\theta}^{>b}_{t-1}\mathcal{D})$ called Gibbs sampler and the acceptance rate of this step is 1.

This is because by substituting the proposal distribution, we get:

$$\alpha_b(\boldsymbol{\theta}^{b*}, \boldsymbol{\theta}^b_{t-1}) = \min\left(1, \frac{p(\boldsymbol{\theta}^{b*}, \boldsymbol{\theta}^{<b}_t, \boldsymbol{\theta}^{>b}_{t-1}|\mathcal{D})}{p(\boldsymbol{\theta}^b_{t-1}, \boldsymbol{\theta}^{<b}_t, \boldsymbol{\theta}^{>b}_{t-1}|\mathcal{D})} \times \frac{p(\boldsymbol{\theta}^b_{t-1} \mid \boldsymbol{\theta}^{<b}_t, \boldsymbol{\theta}^{>b}_{t-1}, \mathcal{D})}{p(\boldsymbol{\theta}^{b*} \mid \boldsymbol{\theta}^{<b}_t, \boldsymbol{\theta}^{>b}_{t-1}, \mathcal{D})}\right)$$

Now, the factor involving the ratios of the joint distributions simplifies to the ratio of the conditional distributions for block b:

$$\frac{p(\boldsymbol{\theta}^{b*}, \boldsymbol{\theta}^{<b}_t, \boldsymbol{\theta}^{>b}_{t-1}|\mathcal{D})}{p(\boldsymbol{\theta}^b_{t-1}, \boldsymbol{\theta}^{<b}_t, \boldsymbol{\theta}^{>b}_{t-1}|\mathcal{D})} = \frac{p(\boldsymbol{\theta}^{b*} \mid \boldsymbol{\theta}^{<b}_t, \boldsymbol{\theta}^{>b}_{t-1}, \mathcal{D})}{p(\boldsymbol{\theta}^b_{t-1} \mid \boldsymbol{\theta}^{<b}_t, \boldsymbol{\theta}^{>b}_{t-1}, \mathcal{D})}$$

Hence, the acceptance rate α^b is:

$$\alpha_b(\boldsymbol{\theta}^{b*}, \boldsymbol{\theta}^b_{t-1}) = 1$$

every proposed sample is accepted for the Gibbs sampler.

Chapter 2

Finite mixture copula estimation and selection using Bayesian methods

2.1 Introduction

In this chapter, we develop a Bayesian MCMC and EM approach of estimating and selecting the mixture copula models. The general framework of the methodology is called sparse finite mixture model, first developed and proofed theoretically by Rousseau and Mengersen (2011) and later applied in several works including (Gelman et al., 2013; Van Havre et al., 2015; Malsiner-Walli et al., 2016; Frühwirth-Schnatter and Malsiner-Walli, 2019). In contrast to employing the classical methodologies to select mixture copula models such as repeatedly calculating AIC (Wagenmakers and Farrell, 2004), BIC (Kuha, 2004), Bayes factor (Kass and Raftery, 1995), and DIC (Spiegelhalter et al., 2014) for different forms of models and comparing their values, Bayesian modeling of sparse finite mixture estimates and selects its components simultaneously. This is particularly convenient in the case of mixture model inference and selection as the number of candidate best models is sometimes combinatorial (e.g. mixture models with heterogeneous components).

We therefore propose to estimate and select the copula mixture by the framework of the Bayesian sparse finite mixture models in this chapter, which has rarely been done elsewhere in the literature. Following our introduction, the rest of the chapter is organized as follows. In section 2.2, we introduce the general framework of the Bayesian sparse copula mixture models, including how we overfit and construct the valid priors of the copula mixture first and then proceed to estimate the parameters while simultaneously determining the correct

number of components. These are followed by outlining the implementation algorithms of MCMC and EM, which are constructed based on the model

$$C_{\text{mix}}(u_1, u_2) = w_1 C_{\text{fr}}(u_1, u_2) + w_2 C_{\text{No}}(u_1, u_2) + w_3 C_{\text{Cl}}(u_1, u_2) + w_4 C_{\text{Gu}}(u_1, u_2), \tag{2.1}$$

where $C_{\text{fr}}, C_{\text{No}}, C_{\text{Cl}}, C_{\text{Gu}}$ is the Frank Copula, Normal Copula, Clayton Copula and Gumbel Copula respectively. As we can see from Figure 1.1, the mixture of them would be able to describe various dependence patterns such as extreme upper tail, extreme lower tail, or spherical symmetric. The connection of our approach with the penalized likelihood method from Wang (2008); Cai and Wang (2014) has also been made through the EM approach in this section. In the following numerical simulations, we test the validity of our algorithms using the model (2.1). Furthermore, we also work with 3-dimensional mixture of Gaussian copulas in Section 2.3.3 to further demonstrate the validity of our approach in multi-dimensional situations. In the section on real data analysis, we again apply the model (2.1) along with our Bayesian sparse mixture methods to study the dependence among major stock markets in Shanghai, Hong Kong, and New York.

2.2 Estimation and selection

The general starting point is to construct the model by writing out

$$C_{mix}(\boldsymbol{u}) = \sum_{j=1}^{K} w_j C_j(\boldsymbol{u}; \boldsymbol{\theta}_j), \tag{2.2}$$

with the knowledge that a true model has the form

$$C^0(\boldsymbol{u}) = \sum_{j=1}^{K'} w_j C_j(\boldsymbol{u}; \boldsymbol{\theta}_j^0), \text{for} \quad K' \leq K.$$

where $C_j(\cdot), C_k(\cdot)$ for any $j, k \leq K$ can either from the same parametric family or not, although for the former case, one needs to take extra measures for the label switching problems (Gelman et al., 2013, Section 22.3). We then proceed to directly estimate (2.2) by the Bayesian approach of finite mixture models but with regularized Dirichlet weighting priors. Rousseau and Mengersen (2011) showed that by applying this approach to the standard finite mixture distribution, it would clear out the redundant components asymptotically as long

as the regularity conditions are met. In particular, they showed for $\boldsymbol{w} = (w_1, w_2, \ldots, w_K) \sim$ Dirichlet$(\alpha_1, \alpha_2, \ldots, \alpha_K)$ with $\max_{j=1,2,\ldots,K} \alpha_j < \dim(\boldsymbol{\theta_j})/2$ plus some other constraints, the posterior estimation of weights has the property $\sum_{j=K'+1}^{K} E[w_j|D] = O_P(1/\sqrt{n})$. This result has shown us the extra stability of the Bayesian estimation due to its shrinkage property compared with the maximum likelihood approach (MLE) since the MLE of an over-fitted model only guarantees the convergence to an unidentifiable set with the limiting distribution $C_\infty(\cdot) = C^0(\cdot)$ in the domain as $n \to \infty$ (Feng and McCulloch, 1996). However, the asymptotic results do not guarantee sparsity. This would cause a failure to identify the correct number of components if, for example, $i, j, k \leq K$, $w_i C_i(u) + w_j C_j(u) = w_k C_k(u)$ is achievable in the model setting.

On the other hand, Cai and Wang (2014) approached the mixed copula estimation and selection problem using penalized MLE approach. In terms of its nature, this approach is quite similar to Bayesian estimation. However, the authors only applied penalties to the weighting parameters, whereas the Bayesian counterpart typically applies penalties to all parameters especially when the informative priors are used. The connection between these two approaches is established using the EM approach of the posterior, as outlined at the end of Section 2.2.2, where we compare the maximization form of the posterior mode and the penalized MLE.

2.2.1 Markov chain Monte Carlo

We show the MCMC sampling algorithm of model (2.1), but the general framework and methodology of the inference remain the same for all forms of (2.2). In addition, it is straightforward to extend the work to the high dimensional implicit copulas introduced in Smith (2023), including some skew elliptical copulas by minor modifications. Hence, our approaches remain valid in high-dimensional settings.

For the d-variate data set of sample size n, $D_n = \{\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_n\}$ where $\boldsymbol{X}_i \in R^d$ represents an i.i.d sampling from some multivariate distribution. In most cases, we do not possess any additional information regarding the marginal distributions, it would be necessary to estimate them together or treat them as nuisance parameters using the nonparametric approach. That is, we either specify the marginal parametric model so that the likelihood for the i.i.d data is

$$p(D_n \mid \boldsymbol{\alpha}, \boldsymbol{\theta}_{mix}) = \prod_{i=1}^{n} c_{mix}(F_1(X_{i1}; \boldsymbol{\alpha_1}), F_2(X_{i2}; \boldsymbol{\alpha_2}), \ldots, F_d(X_{id}; \boldsymbol{\alpha_d}); \boldsymbol{\theta}_{mix}) \prod_{j=1}^{d} f_j(X_{ij}; \boldsymbol{\alpha_1}).$$

Or, to avoid misspecification of the marginal models, we use the semiparametric approach. The pseudo-likelihood for the i.i.d data is

$$p(D_n|\boldsymbol{\theta}_{mix}) \propto \prod_{i=1}^{n} c_{mix}(\hat{F}_{n1}(X_{i1}), \hat{F}_{n2}(X_{i2}), \ldots, \hat{F}_{nd}(X_{id}); \boldsymbol{\theta}_{mix})$$

where we have

$$\hat{F}_{nj}(x) = \frac{1}{n+1} \sum_{i=1}^{n} \mathbb{I}(X_{ij} \leq x).$$

Other alternatives of the margins $\hat{F}_n = (\hat{F}_{n1}, \hat{F}_{n2}, \ldots, \hat{F}_{nd})$ such as kernel density estimations are also available (Patton, 2012). Therefore, only $\boldsymbol{\theta}_{mix}$ is estimated here. In this paper, we focus on the discussion of semiparametric cases.

We specifying the prior of $\boldsymbol{w}$ and $\boldsymbol{\theta}_{mix}$ with

$$\begin{aligned}
\pi(\boldsymbol{w}) &\sim \mathbf{Dirichilet}(\alpha_1, \alpha_2, \ldots, \alpha_K) \\
\pi(\boldsymbol{\theta}_{mix}) &\sim N_d(\mathbf{0}, I_d)
\end{aligned}$$

Note that for any copula parameters which do not have the range $(-\infty, +\infty)$, when convenient, we transfer them from the original parameter space to $\mathbb{R}$ so that $\theta = \phi(\theta_0) \in (-\infty, +\infty)$. Hence, we will be able to unify the prior to be normal. In case of the model (2.1), denote $\boldsymbol{\theta}_{mix} \in (-\infty, +\infty)^4$, the original parameters can be obtained by

$$\begin{aligned}
\theta^{\text{ori}}_{\text{clayton}} &= \exp(\theta_{\text{mixclayton}}) \\
\theta^{\text{ori}}_{\text{gumbel}} &= \exp(\theta_{\text{mixgumbel}}) + 1 \\
\theta^{\text{ori}}_{\text{normal}} &= \frac{1 - \exp(-\theta_{\text{mixnormal}})}{1 + \exp(-\theta_{\text{mixnormal}})}. \\
\theta^{\text{ori}}_{\text{frank}} &= \theta_{\text{mixfrank}}
\end{aligned} \tag{2.3}$$

where θ^{ori} refer to the parameters in the classical copula settings and $\theta_{\text{mix...}}$ refer to the parameters after the transformation $\phi(\cdot)$. We augment our data to $(\boldsymbol{X}_i, Z_i)$, where Z_i denotes the cluster of the point i, so that $p(\boldsymbol{X}_i \mid Z_i = k, \boldsymbol{\theta}_{mix}) \propto c_k(\hat{F}_{n1}(X_{i1}), \hat{F}_{n2}(X_{i2}), \ldots, \hat{F}_{nd}(X_{id}); \boldsymbol{\theta}_k)$. The Metropolis–Hasting algorithm of sampling the posterior $p(\boldsymbol{\theta}_{mix}, \boldsymbol{w} \mid D_n)$ follows as:

1. Setting initial values $\boldsymbol{\theta}^{(0)}_{mix}, \boldsymbol{w}^{(0)}$.

2. Denote the current round to be t, iteratively updating $Z_i^{(t)}$ such that $p(Z_i^{(t)} \mid \boldsymbol{Z}^{(t)}_{\setminus i}, \boldsymbol{D_n}, \boldsymbol{w})$

$\propto p(\boldsymbol{X}_i \mid \boldsymbol{Z}^{(t)}, \boldsymbol{w}) p(Z_i^{(t)} \mid \boldsymbol{w})$ for $i = 1, 2, \ldots, n$ using Gibbs procedure; this can be sampled from the multinomial distribution with $p_k = \frac{w_k c_k(\hat{F}_n(\boldsymbol{X}_i)|\theta_k)}{\sum_j w_j c_j(\hat{F}_n(\boldsymbol{X}_i)|\theta_i)}$ with $k = 1, 2, \ldots, K$.

3. For all $i = 1, 2, \ldots, K$, we propose $f(\theta_i^* \mid \theta_i^{t-1}) \sim N_d(\boldsymbol{\theta}_i^{t-1}, \hat{\Sigma})$ where $\hat{\Sigma}$ is updated every 50 iterations from the sample variance of previously accepted points. We accept the $\theta_i^* = \theta_i^t$ with the acceptance rate

$$a_i = \frac{\prod_{j=1}^{n_i} c_i(X_{ij}; \theta_i^*) \pi(\theta_i^*) f(\theta_i^{t-1} \mid \theta_i^*)}{\prod_{j=1}^{n_i} c_i(X_{ij}; \theta_i^{t-1}) \pi(\theta_i^{t-1}) f(\theta_i^* \mid \theta_i^{t-1})}.$$

4. Update $\boldsymbol{w}^{(t)} \sim \textbf{Dirichilet}(\alpha_1 + \sum_{i=1}^n \mathbb{I}(Z_i^{(t)} = 1), \ldots, \alpha_K + \sum_{i=1}^n \mathbb{I}(Z_i^{(t)} = K))$, where $\alpha_1, \alpha_2, \ldots, \alpha_K$ are set to be $1/K$ to satisfied the regularity condition of Rousseau and Mengersen (2011).

5. Repeat steps 2–4 until the stopping criteria are reached, for example, after 10,000 iterations.

The MCMC method would be sufficient for our purpose. However, by setting up the EM method for the posterior mode, we can bridge between the Bayesian methods and the penalized likelihood methods discussed in Wang (2008); Cai and Wang (2014). In addition, the EM approach could enable a truncation process during its iterations, making the final results much more straightforward to read. That is, only the components selected by the EM would have non-zero weightings after the algorithm stops.

2.2.2 EM algorithm

Start from the complete data (X_i, Z_i) where Z_i is the cluster label as previously. Therefore, we denote $Q(Z) := \log p(\boldsymbol{w}, \boldsymbol{\theta}_{mix}, Z|X)$; our goal is to work iteratively so that

$$(w^{t+1}, \theta^{t+1}) = \text{argmax}_{\theta, \boldsymbol{w}} \int Q(Z) p(Z \mid \boldsymbol{X}, \boldsymbol{\theta}_{mix}^t, \boldsymbol{w}^t) dZ = \text{argmax}_{\theta, \boldsymbol{w}} E_{p(Z|\boldsymbol{X}, \boldsymbol{\theta}_{mix}^t, \boldsymbol{w}^t)}(Q) \tag{2.4}$$

In more detail,

$$
\begin{aligned}
&Q(Z) = \log p(\boldsymbol{w}, \boldsymbol{\theta}_{mix}, Z|\boldsymbol{X}) \\
&\propto \log p(\boldsymbol{X}|\boldsymbol{w}, \boldsymbol{\theta}_{mix}, Z) + \log p(Z|\boldsymbol{w}) + \log p(\boldsymbol{w}, \boldsymbol{\theta}_{mix}) \\
&\propto \log \prod_{i=1}^{n}\prod_{j=1}^{K} c_j(F_n(\boldsymbol{X}_i);\theta_j)^{\mathbb{I}(Z_i=j)} + \log \prod_{i=1}^{n}\prod_{j=1}^{K} w_j^{\mathbb{I}(Z_i=j)} \\
&+ \log \prod_{j=1}^{K} w_j^{(\alpha_j-1)} - \sum_{j=1}^{K}\frac{1}{2}||\theta_j||^2 + C,
\end{aligned}
$$

where we have denoted the irrelevant constant to be C, and $p(\boldsymbol{w}, \boldsymbol{\theta}_{mix}) = p(\boldsymbol{w})p(\boldsymbol{\theta}_{mix})$.

Hence, we take the expectation so that the argmax of (2.4) would be equivalent as

$$
\begin{aligned}
&\text{argmax}_{\boldsymbol{w},\boldsymbol{\theta}_{mix}} \sum_{i,j} \log c_j(\hat{F}_n(\boldsymbol{X}_i);\theta_j) E(\mathbb{I}(Z_i = j)) \\
&+ \sum_{i,j} E(\mathbb{I}(Z_i = j)) \log w_j + \sum_{j=1}^{K}(\alpha_j - 1)\log w_j - \sum_{j=1}^{K}\frac{1}{2}||\theta_j||^2 \qquad (2.5) \\
&= \sum_{i,j} r_{ij}^t \log c_j(\hat{F}_n(\boldsymbol{X}_i);\theta_j) + \sum_{i,j} r_{ij}^t \log w_j - (1 - \frac{1}{K})\sum_j \log w_j - \sum_j \frac{1}{2}||\theta_j||^2,
\end{aligned}
$$

where we have taken $\alpha_j = 1/K$ to make it less informative while satisfying the regularity condition of Rousseau and Mengersen (2011) and

$$
r_{ij}^t = \frac{w_j^t c_j(y_i|\theta_j^t)}{\sum_j w_j^t c_j(y_i|\theta_j^t)}
$$

To achieve the maximum, we differentiate with respect to w_j while adding the Lagrange multiplier $\lambda(1 - \sum w_j)$, we have

$$
w_j^{t+1} = \frac{1}{N + 1 - K}\bigg(\sum_i \frac{w_j^t c_j(y_i|\theta_j^t)}{\sum_j w_j^t c_j(y_i|\theta_j^t)} - (1 - \frac{1}{K})\bigg). \qquad (2.6)
$$

Differentiate with respect to θ_j, and it can be solved numerically using quasi-Newton methods.

We note that the goal of the EM method is to find the mode of the log posterior

$$\begin{aligned}\log p(\boldsymbol{w}, \boldsymbol{\theta}_{mix}|\boldsymbol{X}) &\propto \sum_i \log \sum_j w_j c_{ij}(\hat{F}_n(\boldsymbol{X}_i); \theta_j) - (1 - \frac{1}{K}) \sum_j \log w_j - \sum_j \frac{1}{2}||\theta_j||^2 \\ &= \sum_i \log \sum_j w_j c_{ij}(\hat{F}_n(\boldsymbol{X}_i); \theta_j) - n \sum_j \Omega^{\boldsymbol{w}}_{(1-1/K)}(w_j) - n \sum_j \Omega^{\boldsymbol{\theta}_{mix}}_{(1/2)}(\theta_j),\end{aligned} \tag{2.7}$$

This form shares a similar structure as (3.2) in Wang (2008) or (3) in Cai and Wang (2014) despite the fact that they do not penalize the copula parameters. Wang (2008) proved the $\sqrt{n}$—asymptotic consistency and sparsity of their semiparametric SCAD-penalized likelihood approach. The validity of our Bayesian methods will be tested empirically in the next part. However, the theoretical demonstrations of consistency are more challenging to consider with Dirichlet distribution priors due to the singularity of $\log w_i$ at $w_i = 0$ (Fan and Li, 2001).

One shortcoming of using the EM method is the difficulty in obtaining the confidence interval of estimators. Bootstrap could be a very computationally intensive solution. On the other hand, one may consider the fisher information matrix $-E\nabla\nabla_{\boldsymbol{w},\boldsymbol{\theta}} \log p(\boldsymbol{X} \mid \hat{\boldsymbol{w}}, \hat{\boldsymbol{\theta}}_{\boldsymbol{mix}})$ as an asymptotic approximation of the precision matrix. Gelman et al. (2013) (p. 324) provide an approach to iteratively calculate the asymptotic variance matrix along with the parameter estimations.

2.3 Numerical simulations

2.3.1 Markov chain monte carlo

We perform two types of numerical simulations. Firstly, we assume that the marginal distributions of the data are perfectly known. Therefore, we focus on the estimation of the copula using the data $(U_{i1}, U_{i2}, \ldots, U_{id}) = (F_{i1}(\boldsymbol{X}_1), F_{i2}(\boldsymbol{X}_2), \ldots, F_{id}(\boldsymbol{X}_d))$ for $i = 1, 2, \ldots, n$. the dimension d is set to be 2 for our simulation purpose. Our working model is (2.1). That is,

$$C_{\text{mix}}(u_1, u_2) = w_1 C_{\text{fr}}(u_1, u_2) + w_2 C_{\text{No}}(u_1, u_2) + w_3 C_{\text{Cl}}(u_1, u_2) + w_4 C_{\text{Gu}}(u_1, u_2).$$

We sample the data from different true models, which are submodels of (2.1), and we estimate them using the MCMC method of Section 2.2.1. Secondly, we assume that the marginal distributions are unknown, we hence estimate the margins empirically using $\hat{F}_{np}(x) =$

$\frac{1}{n+1}\sum_{i=1}^{n}\mathbb{I}(X_{ip} \leq x)$. Thus, we have $(\hat{U}_{i1}, \hat{U}_{i2}, \ldots, \hat{U}_{id}) = (\hat{F}_{i1}(\boldsymbol{X}_1), \hat{F}_{i2}(\boldsymbol{X}_2), \ldots, \hat{F}_{id}(\boldsymbol{X}_d))$ for $i = 1, 2, \ldots, n$ and the copula parameters can be estimated thereafter.

We perform 3000 MCMC iterations for all models, with the first 2500 points discarded as the burning stage. The number of the sample points is $n = 400, 800, 2000$. Tables 2.1 and 2.2 display the simulation results. In general, the weighting parameters as well as the copula parameters of non-zero weighting components approach the truth with decreasing Monte Carlo standard deviation. The mean and error estimations of the copula parameters with zero weightings remain close to its priors, which might be considered as an advantage over the penalized method used in Wang (2008); Cai and Wang (2014) as they proved that the zero weighting copula parameters would end up randomly in their parameter spaces by using their penalized likelihood approach. Three major misidentification cases were found in tables, that is, $n = 400, 800$ of Frank copulas simulations in Table 2.1 and $n = 800$ of Frank copulas in the Table 2.2. All cases mentioned seem to be misidentified as normal copulas, which are understandable as the normal copula and Frank copula share very similar structures with zero tail dependence.

2.3.2 Expectation maximization

In this part, we investigate the performances of the EM algorithm introduced in Section 2.2.2. The approach is computationally demanding. Therefore, we only show the results with the sample size of $n = 200, 400, 800$ for one-component copulas. Data are generated directly from the true copula models. More specifically, for each sample size of $n = 200, 400, 800$, we generate 10 batches of data from the true distribution. Every batch is learned by the EM method, and the stopping criteria are 1000 full iterations or the absolute sum of the parameters increase less than 0.001 for an iteration. We calculate the mean and variance estimators for each sample size. Table 2.3 displays the results of the EM approach. It shows comparable outcomes with the MCMC. Although all algorithms fail to distinguish the Frank copulas from the normal ones due to their similarities, other copulas are selected with satisfactory accuracy. One clear advantage of using the EM is its convenience in introducing an exit mechanism for unlikely copulas during the training process. That is, due to the shrinkage term of the weight in (2.6), we can eliminate components when their corresponding weights fall down to non-positive during the training. By adding this procedure, we can automatically consider fewer mixture components at later stages. As we can see from Table 2.3, there are many components with deterministic 0 weightings. However, the shortcomings of the

Table 2.1: MCMC estimations of the copula with the marginal distributions fully known. The numbers inside parentheses indicate standard errors, and estimations of the true components are denoted in bold font.

True Copula(param)									
		MCMC Estimation							
	n	Clayton		Gumbel		Normal		Frank	
		w	θ	w	θ	w	θ	w	θ
Normal(0.5)	400	0.089(0.088)	1.568(1.670)	0.047(0.036)	3.801(2.750)	**0.839(0.105)**	**0.444(0.046)**	0.025(0.032)	-0.029(0.994)
	800	0.0439(0.066)	1.173(1.586)	0.007(0.011)	2.494(1.370)	**0.940(0.066)**	**0.514(0.026)**	0.009(0.014)	0.113(0.916)
	2000	0.039(0.049)	1.363(1.424)	0.028(0.034)	3.087(1.968)	**0.913(0.063)**	**0.494(0.024)**	0.020(0.038)	0.043(1.120)
Clayton(5)	400	**0.990(0.011)**	**4.914(0.246)**	0.005(0.010)	2.490(1.612)	0.003(0.005)	0.526(0.224)	0.002(0.003)	0.012(0.965)
	800	**0.992(0.009)**	**4.876(0.185)**	0.003(0.006)	2.641(1.774)	0.003(0.005)	0.488(0.207)	0.002(0.004)	0.037(0.944)
	2000	**0.996(0.003)**	**5.091(0.133)**	0.001(0.002)	2.411(1.569)	0.001(0.002)	0.568(0.205)	0.001(0.001)	-0.198(0.983)
Gumbel(2.5)	400	0.017(0.027)	1.676(1.505)	**0.957(0.038)**	**2.486(0.105)**	0.022(0.033)	0.530(0.210)	0.004(0.007)	0.080(1.002)
	800	0.002(0.004)	1.480(1.593)	**0.991(0.009)**	**2.701(0.071)**	0.004(0.007)	0.545(0.210)	0.002(0.005)	0.070(1.051)
	2000	0.006(0.008)	1.442(1.268)	**0.988(0.014)**	**2.470(0.048)**	0.005(0.011)	0.533(0.194)	0.001(0.002)	-0.091(0.968)
Frank(5)	400	0.061(0.083)	1.903(1.512)	0.030(0.041)	2.386(2.115)	0.875(0.087)	0.647(0.038)	**0.033(0.047)**	**0.416(1.037)**
	800	0.058(0.041)	3.774(2.751)	0.019(0.039)	2.324(2.043)	0.899(0.055)	0.603(0.031)	**0.024(0.031)**	**0.280(0.984)**
	2000	0.007(0.012)	1.358(1.223)	0.004(0.007)	2.255(1.394)	0.205(0.055)	0.790(0.041)	**0.784(0.058)**	**4.408(0.285)**
0.5Gumbel(2.5)+0.5Clayton(5)	400	**0.439(0.057)**	**6.079(0.761)**	**0.533(0.059)**	**2.756(0.242)**	0.024(0.036)	0.569(0.207)	0.004(0.007)	0.176(1.003)
	800	**0.567(0.034)**	**5.332(0.390)**	**0.429(0.040)**	**2.328(0.143)**	0.002(0.004)	0.514(0.210)	0.002(0.004)	0.126(0.976)
	2000	**0.509(0.034)**	**5.111(0.356)**	**0.480(0.032)**	**2.505(0.076)**	0.005(0.008)	0.523(0.200)	0.006(0.007)	0.182(1.005)
0.5Clayton(5)+0.5Normal(0.5)	400	**0.513(0.087)**	**5.150(1.054)**	0.061(0.070)	2.606(2.353)	**0.383(0.095)**	**0.554(0.080)**	0.044(0.067)	0.280(1.036)
	800	**0.573(0.041)**	**4.107(0.336)**	0.165(0.079)	1.833(0.534)	**0.191(0.144)**	**0.410(0.126)**	0.069(0.086)	0.365(1.028)
	2000	**0.456(0.035)**	**5.500(0.372)**	0.069(0.046)	2.750(0.788)	**0.473(0.035)**	**0.466(0.035)**	0.002(0.003)	-0.105(0.941)

Table 2.2: MCMC estimations of the copula with the marginal distributions estimated by empirical distribution. The numbers inside parentheses indicate standard errors, and estimations of the true components are denoted in bold font. The corresponding true marginal distribution is $N(1,1)$ and $N(0.5,1)$.

True Copula(param)		MCMC Estimation							
	n	Clayton		Gumbel		Normal		Frank	
		w	θ	w	θ	w	θ	w	θ
Normal(0.5)	400	0.003(0.005)	1.637(1.785)	0.114(0.125)	2.015(1.073)	**0.878(0.124)**	**0.590(0.038)**	0.005(0.009)	0.064(0.993)
	800	0.08(0.08)	1.543(1.321)	0.008(0.011)	2.569(1.758)	**0.886(0.102)**	**0.568(0.033)**	0.026(0.037)	0.246(1.054)
	2000	0.025(0.039)	0.845(1.039)	0.021(0.021)	1.740(0.786)	**0.952(0.048)**	**0.541(0.022)**	0.002(0.004)	-0.072(0.951)
Clayton(5)	400	**0.987(0.015)**	**4.856(0.240)**	0.006(0.014)	2.648(1.718)	0.004(0.007)	0.530(0.204)	0.002(0.004)	0.061(0.929)
	800	**0.994(0.006)**	**4.733(0.185)**	0.003(0.005)	2.957(1.793)	0.001(0.002)	0.499(0.226)	0.001(0.003)	-0.050(1.023)
	2000	**0.996(0.004)**	**5.423(0.130)**	0.002(0.004)	3.438(2.236)	0.001(0.002)	0.539(0.197)	0.001(0.001)	-0.088(0.959)
Gumbel(2.5)	400	0.009(0.018)	1.589(1.533)	**0.971(0.034)**	**2.830(0.122)**	0.018(0.031)	0.554(0.190)	0.002(0.004)	-0.104(1.088)
	800	0.005(0.007)	2.293(2.862)	**0.981(0.021)**	**2.652(0.084)**	0.012(0.019)	0.484(0.199)	0.002(0.003)	0.076(0.931)
	2000	0.004(0.008)	2.295(2.647)	**0.993(0.008)**	**2.530(0.044)**	0.001(0.001)	0.522(0.216)	0.002(0.004)	0.156(0.962)
Frank(5)	400	0.012(0.016)	2.220(2.809)	0.096(0.099)	2.333(1.019)	0.005(0.010)	0.532(0.197)	**0.887(0.094)**	**4.533(0.376)**
	800	0.147(0.044)	4.664(1.414)	0.006(0.011)	2.430(1.634)	0.836(0.048)	0.599(0.028)	**0.011(0.018)**	**0.145(1.028)**
	2000	0.005(0.007)	2.334(3.135)	0.050(0.028)	2.659(0.833)	0.016(0.024)	0.453(0.208)	**0.929(0.023)**	**5.107(0.281)**
0.5Gumbel(2.5)+0.5Clayton(5)	400	**0.551(0.085)**	**4.327(0.556)**	**0.363(0.129)**	**2.586(0.276)**	0.080(0.135)	0.603(0.206)	0.006(0.009)	0.072(1.071)
	800	**0.413(0.046)**	**5.149(0.526)**	**0.538(0.060)**	**2.645(0.167)**	0.050(0.050)	0.558(0.211)	0.003(0.007)	-0.031(1.136)
	2000	**0.531(0.030)**	**4.792(0.270)**	**0.464(0.031)**	**2.500(0.089)**	0.004(0.008)	0.508(0.197)	0.001(0.002)	0.104(1.026)
0.5Clayton(5)+0.5Normal(0.5)	400	**0.502(0.082)**	**4.871(0.992)**	0.044(0.077)	2.376(1.230)	**0.450(0.109)**	**0.488(0.077)**	0.005(0.009)	-0.122(0.960)
	800	**0.526(0.042)**	**5.321(0.461)**	0.015(0.020)	2.841(1.888)	**0.444(0.054)**	**0.485(0.048)**	0.015(0.026)	0.065(1.021)
	2000	**0.534(0.034)**	**4.725(0.325)**	0.106(0.058)	1.751(0.353)	**0.351(0.070)**	**0.512(0.060)**	0.009(0.011)	0.015(0.966)

EM approach are also very clear. It is more computationally demanding especially when we seek to obtain some estimation errors or work with high dimensional copulas. On the other hand, the EM seeks to find the posterior mode which is less favorable than the posterior mean in statistical decision theories, while the MCMC approach gives full posterior distributions, and it is well acknowledged that the performance of the EM could be affected by starting points.

2.3.3 Multi-dimensional cases

We proceed to test the effectiveness of our approach in a multi-dimensional case. As the classic Archimedean families of copulas are rarely used in multi-dimensional (i.e. the dimensions more than 3) applications due to the restriction of their parameter spaces. We apply the more commonly used Gaussian mixture copulas with 3 components to perform the estimations while the dimension of the data is set to be 3. That is, we use the MCMC sampler to estimate the model

$$c_{NormalMix} = w_\alpha c_\alpha(\boldsymbol{u}; \boldsymbol{\Sigma_\alpha}) + w_\beta c_\beta(\boldsymbol{u}; \boldsymbol{\Sigma_\beta}) + w_\gamma c_\gamma(\boldsymbol{u}; \boldsymbol{\Sigma_\gamma}). \tag{2.8}$$

A major obstacle to performing MCMC of such type is the sampling of the correlation matrices $(\Sigma_\alpha, \Sigma_\beta, \Sigma_\gamma)$. The valid sampler should generate symmetric positive definite matrices every time with every entry from 0 to 1 and 1 in their diagonal. The approach will be made clearer in the latter chapter. On the other hand, when performing the MCMC sampling with mixture copulas from the same parametric families, it should also be noticed that label-switching problems often occurred. This is because that (2.8) has 3! equivalent forms by just switching the labels, some engineering efforts should be made to mitigate the circumstances. After every round of iteration, one can post-process the model so that the component with the highest weighting always ranks first. In addition, if the weightings are too close to distinguish, further criteria such as $\det|\Sigma| + \text{trace}(\Sigma)$ should be used.

In this study, we use the data simulated from the submodels of (2.8) with parameters

$$\begin{aligned}
(\Sigma_\alpha^{12}, \Sigma_\alpha^{23}, \Sigma_\alpha^{13}) &= (0.7, 0.7, -0.6) \\
(\Sigma_\beta^{12}, \Sigma_\beta^{23}, \Sigma_\beta^{13}) &= (0.6, 0.6, 0.6) \\
(\Sigma_\gamma^{12}, \Sigma_\beta^{23}, \Sigma_\gamma^{13}) &= (-0.7, 0.7, 0.7).
\end{aligned}$$

Table 2.3: EM estimations of the copula with the marginal distributions fully known. The numbers inside parentheses indicate standard errors, and estimations of the true components are denoted in bold font. The starting value of the EM is $\boldsymbol{w} = (0.25, 0.25, 0.25, 0.25), \boldsymbol{\theta}_{mix} = (1, 1, 1, 1)$.

True Copula(param)		EM Estimations							
	n	Clayton		Gumbel		Normal		Frank	
		w	θ	w	θ	w	θ	w	θ
Normal(0.5)	200	0.020(0.060)	1.137(0.435)	0.035(0.110)	1.957(0.135)	**0.947(0.117)**	**0.509(0.030)**	0(0)	0.5(0)
	400	0.031(0.078)	1.479(1.661)	0.050(0.12)	2.031(0.351)	**0.921(0.129)**	**0.506(0.042)**	0(0)	0.5(0)
	800	0.031(0.068)	1.021(0.134)	0(0)	2(0)	**0.969(0.068)**	**0.485(0.021)**	0(0)	0.5(0)
Clayton(5)	200	**1(0)**	**4.955(0.525)**	0(0)	2(0)	0(0)	0.5(0)	0(0)	0.5(0)
	400	**0.989(0.035)**	**4.972(0.246)**	0.012(0.036)	2.628(1.985)	0(0)	0.5(0)	0(0)	0.5(0)
	800	**1(0)**	**4.988(0.162)**	0(0)	2(0)	0(0)	0.5(0)	0(0)	0.5(0)
Gumbel(2.5)	200	0(0)	1(0)	**1(0)**	**2.486(0.177)**	0(0)	0.5(0)	0(0)	0.5(0)
	400	0(0)	1(0)	**1(0)**	**2.500(0.088)**	0(0)	0.5(0)	0(0)	0.5(0)
	800	0(0)	1(0)	**0.979(0.038)**	**2.562(0.081)**	0.02(0.04)	0.513(0.058)	0(0)	0.5(0)
Frank(5)	200	0.117(0.138)	2.182(1.415)	0.155(0.268)	2.016(0.118)	0.723(0.244)	0.557(0.102)	**0.010(0.030)**	**0.498(0.007)**
	400	0.054(0.087)	1.619(1.075)	0.184(0.187)	2.283(0.383)	0.764(0.228)	0.555(0.080)	**0(0)**	**0.5(0)**
	800	0.075(0.085)	2.201(1.456)	0.060(0.103)	2.249(0.498)	0.840(0.089)	0.608(0.047)	**0.030(0.050)**	**0.517(0.029)**

And we set the true weighting of the experiments to be

$$(w_\alpha, w_\beta, w_\gamma) = (1, 0, 0) \quad (w_\alpha, w_\beta, w_\gamma) = (0, 0.7, 0.3)$$

respectively. Furthermore, the sample sizes of the simulated data are set to be

$$n = 400, 800, 2000.$$

Table 2.4 displays the results of experiments. It shows good signs of convergence to the truth while we increase the sample size. For the 2000 sample size experiments, the true components are successfully filtered out with low uncertainty, indicating the effectiveness of our proposed approach beyond the classical two-dimensional copula applications.

2.4 Real data analysis

In the real data analysis, we use financial trading data from three major indices, that is, Standard&Poors 500 (SP 500), Shanghai Composite Index (SSEC), and Hang Seng Index (HSI). Daily close prices from 09.Oct.2017 to 29.Sep.2022 were extracted, we aligned three series with the common trading days among them, other days were omitted. To ease the analysis of the dependence pattern among them, we take the log returns respectively so that $R_i = \log P_i - \log P_{i-1}, i = 1, 2, 3$. This method of transforming stock prices into log returns is a well-established technique in the field of financial analysis, as evidenced by several studies (Fergusson and Platen, 2006; Almeida and Czado, 2012; Ardia and Hoogerheide, 2014). Table 2.5 shows the Pearson and Spearman correlation among markets. SSEC and HSI display strong levels of dependence, while their connection with SP500 is relatively weak for those two markets. However, as we argued previously, the single metric of correlation does not give the full picture of the dependence. It is therefore reasonable to apply the mixture copula models for further analysis. In addition, the Ljung-Box tests to the absolute values $|R_i|$ of series indicate all series are correlated to themselves through time. Moreover, the augmented Dickey-Fuller tests show that they are covariance stationary. To apply the copula models to the autocorrelated data, we use the standard method of standardizing. That is, the

Table 2.4: MCMC estimations of the 3-dimensional mixture Gaussian copulas with the marginal distributions fully known. The numbers inside parentheses indicate standard errors, and estimations of the true components are denoted in bold font. Comp is the abbreviation for Component and the components are ordered by their weightings.

True Copula(param)								
		MCMC Estimations						
	n	Comp1		Comp2		Comp3		
		w	θ	w	θ	w	θ	
Normal(0.7,-0.7,-0.6)	400	**0.815(0.160)**	**0.691(0.031),-0.676(0.042),-0.602(0.048)**	0.173(0.152)	0.685(0.212),-0.658(0.278),-0.430(0.342)	0.019(0.016)	0.414(0.306),-0.050(0.539),-0.152(0.413)	
	800	**0.991(0.014)**	**0.680(0.017),-0.715(0.015),-0.604(0.023)**	0.007(0.012)	0.371(0.429),-0.435(0.318),-0.413(0.571)	0.002(0.003)	0.312(0.445),-0.341(0.377),-0.308(0.566)	
	2000	**0.992(0.007)**	**0.704(0.009),-0.699(0.010),-0.612(0.012)**	0.007(0.007)	0.154(0.350),-0.362(0.498),-0.336(0.252)	0.001(0.001)	-0.225(0.283),0.303(0.445),-0.230(0.239)	
0.7Normal(0.6,0.6,0.6) + 0.3Normal(-0.7,-0.7,0.7)	400	**0.609(0.097)**	**0.679(0.069),0.616(0.048),0.602(0.043)**	**0.289(0.047)**	**-0.595(0.310),-0.592(0.327),0.713(0.066)'**	0.102(0.085)	-0.224(0.387),-0.170(0.452),0.468(0.141)	
	800	**0.656(0.074)**	**0.567(0.035),0.599(0.042),0.576(0.038)**	**0.216(0.042)**	**-0.535(0.326),-0.594(0.252),0.687(0.076)**	0.128(0.054)	-0.331(0.392),-0.450(0.342),0.724(0.087)	
	2000	**0.663(0.030)**	**0.636(0.017),0.607(0.021),0.603(0.014)**	**0.310(0.020)**	**-0.677(0.030),-0.690(0.029),0.740(0.020)**	0.026(0.027)	0.190(0.320),0.138(0.319),0.257(0.193)	

autocorrelation is removed by rescaling the volatility of the GARCH(1,1) model, assume

$$\begin{aligned} R_t &= \mu + \sigma_t z_t \quad \text{i.i.d} \quad z_t \sim \mathrm{N}(0,1) \\ \sigma_t^2 &= \alpha\sigma_{t-1}^2 + \beta z_{t-1}^2 + \gamma, \end{aligned} \tag{2.9}$$

We apply the data $(Z_1, Z_2, \ldots, Z_T) = (\frac{R_1-\mu}{\sigma_1}, \frac{R_2-\mu}{\sigma_2}, \ldots, \frac{R_T-\mu}{\sigma_T})$ to copula model and use semi-parametric approach of Section 2.2.1 to learn the parameters. The MCMC samplings are done 5000 times with the last 500 times used for analyzing the parameters. Table 2.6 shows the results of the estimation with the insignificant components omitted. As we observe the strong Clayton components in the first two columns, but the Gumbel components, on the other hand, are all very weak among three markets. Due to the asymmetry nature of the Clayton copula at its left tail, which can be seen in Figure 1.1. This indicates the existence of asymmetry dependence among markets, especially at the lower left tail, and the dependence on the upper right tail is less obvious. Our finding means that the stock markets are usually more easily to have a downward comovement but much less likely to move upward together. In contrast, the dependence pattern between HSI and SP500 is more symmetrical, with dominating Normal and Frank components and a very weak Gumbel component. Given the absence of extreme left tail dependence, a portfolio consisting of HSI and SP500 indexes is less likely to experience significant losses compared to other cross-market portfolios during extreme financial conditions.

Table 2.5: Pearson and Spearman Correlation among three markets from Oct 2017 to Sep 2020

	SSEC	HSI	SP500	SSEC	HSI	SP500
	Pearson Correlation			Spearman Correlation		
SSEC	1	0.699	0.18	1	0.679	0.173
HSI	0.699	1	0.25	0.679	1	0.224
SP500	0.18	0.25	1	0.173	0.224	1

2.5 Concluding remarks

In this chapter, we study the Bayesian approach of estimating the overfitted finite mixture copula models; MCMC and EM approaches were tested to verify its validity, and applications

Table 2.6: Parameters estimation of the stocks data with mean estimator and 90% credible interval.

		SSEC-HSI	SSEC-SP500	HSI-SP500
Clayton	w	0.280(0.144,0.372)	0.685(0.508,0.814)	0
	θ	2.53(1.65,3.65)	0.168(0.069,0.247)	
Gumbel	w	0	0	0.104(0.015,0.257)
	θ			1.484(1.130,2.368)
Normal	w	0.668(0.587,0.785)	0.222(0.062,0.350)	0.528(0.350,0.672)
	θ	0.722(0.681,0.764)	0.366(0.190,0.563)	0.400(0.269,0.561)
Frank	w	0	0	0.33(0.201,0.542)
	θ			-0.534(-1.557,0.509)

in the real financial market were done.

Compared with the classic MLE approaches, the Bayesian sampling scheme presented can make the uncertainty of estimators readily available. Hence, we were able to filter the mixture components that were not significant. The existing literature also showed the theoretical merit of Bayesian estimation when compared with the MLE.

If more precise parameter estimation is needed, one could first apply our proposed approach to the model. After filtering out the insignificant component, a second round of estimation could be applied to the submodel. Two rounds of estimation would be sufficient in most cases.

Chapter 3

Nonparametric Bayesian modeling on infinite mixture Student t copulas

3.1 Introduction

In this chapter, we introduce a Bayesian method for estimating an infinite mixture t copula, which is more robust to the corresponding normal mixture model when the data exhibit extreme tail dependence. For instance, in terms of financial modeling, it is widely recognized that the returns of the market exhibit behavior with heavy tails (Borak et al., 2011; Nolan, 2014; Zi-Yi, 2017; Sun et al., 2020; Van Tran and Kukal, 2022), and the returns of financial assets usually experience large co-movements in the event of an extreme market condition (Hu, 2006; Muteba Mwamba and Angaman, 2021; Zi-Yi, 2017). Under these circumstances, using the Student t copula instead of a normal copula will be beneficial, this is because the student t copula is able to depict the extreme market dependence while the normal counterpart fails to achieve this ability (McNeil et al., 2015). Hence, we construct the infinite mixture of the t copula to improve the modeling outcome from the classic normal model.

To the best of our knowledge, few articles consider the infinite t mixture copula model. Wei and Li (2012) studied the estimation of the infinite mixture t-distribution using variational inference, and Wu et al. (2014, 2015) discussed the infinite mixture normal and skew-normal copula. Our intention here is to extend their work to the scenario of the t copula-based infinite mixture model. In terms of the MCMC techniques employed in this chapter, our approach is inspired by the slice sampling approach discussed in Walker (2007)

and Kalli et al. (2011). The sampler is of the Gibbs–Metropolis–Hasting (Gibbs-MH) type, and the general methodology of this sampling scheme can be understood from the introduction chapter and chapters on Bayesian computation in Gelman et al. (2013).

This remaining chapter is organized as follows, in section 3.2, we give an introduction to the model construction of infinite mixture t copulas. In section 3.3, we introduce the Gibbs-MH algorithm for the estimation of the model parameters, which is followed by a simulation study and a real-data analysis using the returns of the Shanghai Composite Index and Shenzhen Component Index from 2018 to 2023. The model correctly identifies the number of mixture components in the simulation study and returns sensible results when compared with the benchmark model in the real-data experiment.

3.2 Infinite mixture t copula and the Dirichlet process

We present the non-parametric Bayesian framework for estimating mixture copulas, the main advantage of which is its ability to automatically determine the number of mixture components. We would like to restate the shortcomings of the classic approach hereby.

The MLE-based approaches typically treat the number of mixture components as a hyper-parameter and must be specified in advance for the mixture copula estimation. Determining the number of components usually involves many rounds of estimations and comparisons by employing performance metrics such as the AIC, BIC, and DIC. This iterative process can be burdensome, particularly when working with data involving many clusters.

Additionally, deriving confidence intervals for the estimators of mixture copula parameters can pose significant challenges when employing MLE-based methods. On the other hand, Bayesian methodologies intrinsically consider uncertainty, thus providing a clear advantage in these scenarios.

If compared with the previous chapter, the former discussion proposed using the MCMC scheme of sparse finite mixture models to estimate while selecting mixture copulas. However, the usefulness of the approach depends on correctly specifying the upper limit of the mixture components. Therefore, setting the component number too low would lead to incorrect model specifications, while setting the number too high could make the algorithm slow. We work with an infinite mixture model here, which automatically determines the number of components without the need to specify the upper limit. In addition, sample points can only be allocated among a limited number of mixture components for the infinite model during iterations, whereas for the finite mixture counterpart, the allocation probability of every

point must be calculated on each component.

To facilitate non-parametric Bayesian modeling, we start by constructing the infinite mixture t copula as follows:

$$C_{\text{inf}}(u_1, u_2, \ldots, u_d) = \sum_{k=1}^{\infty} w_k C_{\mathbf{P_k}, v_k}(u_1, u_2, \ldots, u_d). \tag{3.1}$$

The density of which is expressed as

$$c_{\text{inf}}(u_1, u_2, \ldots, u_d) = \sum_{k=1}^{\infty} w_k c_{\mathbf{P_k}, v_k}(u_1, u_2, \ldots, u_d), \tag{3.2}$$

constrained by $\sum_k w_k = 1$ and $w_k \geq 0 \; \forall k$.

Following our discussion, introducing the infinite model and applying the non-parametric Bayesian approach would free us from the need to manually select the hyperparameter K, which is the number of mixture components in the classical finite mixture model

$$c_{\text{finite}}(u_1, u_2, \ldots, u_d) = \sum_{k=1}^{K} w_k c_{\mathbf{P}_k, v_k}(u_1, u_2, \ldots, u_d).$$

To further construct the non-parametric Bayesian approach, the prior of

$$\{(w_k, \mathbf{P}_k, v_k) \mid k = 1, 2, \ldots\}$$

for (3.2) is set to follow a Dirichlet process (DP), denoted by $G \sim DP(\alpha, G_0)$, where α is the concentration parameter of the Dirichlet process and G_0 is the base measure covering the parameter space of t copula. The main properties of the DP are as follows: For any event S, $\mathbb{E}(G(S)) = G_0(S)$ and $\text{var}(G(S)) = \frac{G_0(S)(1-G_0(S))}{1+\alpha}$. Here, G sampled from the DP can be written in the following atomic form:

$$G = \sum_{k=1}^{\infty} w_k \delta_{\mathbf{\Theta}_k},$$

where δ is the usual notation of the Dirac delta, $\mathbf{\Theta}_k = (\mathbf{P}_k, v_k)$ denotes the parameters of the k^{th} clustering (i.e., the k^{th} mixture component of the copula in our analysis).

For an arbitrary observation Y_i from a mixture model, we introduce latent variables $k_i, \forall i = 1, 2, \ldots, n$, such that the conditional distribution $Y_i | k_i, \mathbf{\Theta}_{k_i} \sim F_{\mathbf{\Theta}_{k_i}}$. The overall

sampling process of the DP prior can be constructed to follow the stick-breaking process (Ishwaran and James, 2001). Similar to the presentation in Papaspiliopoulos and Roberts (2008), this can be formulated as

$$
\begin{aligned}
&(\mathbf{P}_k, v_k) \sim G_0, \\
&V_k \sim \text{Beta}(1, \alpha), \\
&w_k = V_k \prod_{s=1}^{k-1} (1 - V_s), \\
&k = 1, 2, 3, \dots
\end{aligned}
$$

If given the complete information of each point $\mathcal{D} = \{(y_i, k_i, \mathbf{\Theta}_{k_i}) \quad \forall i = 1, 2, \dots, n\}$, we have the posterior of DP $G|\mathcal{D} \sim DP(\alpha + n, \frac{\alpha G_0 + \sum_{i=1}^{n} \delta_{\mathbf{\Theta}_{k_i}}}{\alpha + n})$ (see the chapter on non-parametric Bayesian models in Murphy (2023) for a good explanation of the details). Hence, by introducing $G \sim DP(\alpha, G_0)$ as the prior for the mixture components, our copula mixture model is expressed as

$$
c_{\text{inf}}(\cdots) = \int c_{\mathbf{\Theta}}(\cdots) dG(\mathbf{\Theta}) = \sum_{k=1}^{\infty} w_k c_{\mathbf{P_k}, v_k}(u_1, u_2, \dots, u_d),
$$

which is equivalent to saying that a copula dataset $U_i \sim C_{\text{inf}}(\cdot)$ follows

$$
\begin{aligned}
&G \sim DP(\alpha, G_0), \\
&\mathbf{\Theta}_{k_1}, \mathbf{\Theta}_{k_2}, \dots \mathbf{\Theta}_{k_n} \sim G, \\
&U_i \sim C_{\Theta_{k_i}}.
\end{aligned}
$$

Within the scope of this Bayesian framework, our primary objective is to estimate the parameters

$$
\{(w_k, \mathbf{P}_k, v_k) \mid k = 1, 2, \dots\}
$$

by using the posterior sampling techniques.

3.3 Sampling methodology

3.3.1 Gibbs-MH process

Assume the i-th copula data point is expressed as

$$\boldsymbol{u}_i = (u_{i1}, u_{i2}, \dots, u_{id}) = (F_1(y_{i1}), F_2(y_{i2}), \dots, F_d(y_{id})),$$

where the margins of the d-dimensional data set of the sample size n, $D_n = (\boldsymbol{y}_1, \boldsymbol{y}_2, \dots, \boldsymbol{y}_n)$ are unknown. Again, we estimate the margins of the data empirically using

$$u_{ij} = \frac{1}{n+1} \sum_{m=1}^{n} I\{y_{mj} \leq y_{ij}\} \quad \forall i = 1, 2, \dots, n. \tag{3.3}$$

This is the same as the previous chapter, in the spirit of inference for margins, where the margins of the copulas and their dependence structure are treated separately (see Genest et al. (1995); Joe (2005) for details).

The goal of the estimation, as stated at the end of the previous section, is to estimate the parameters $w_k, \mathbf{P}_k, v_k$ for all $k \in \{1, 2, 3, \dots\}$. To ease the inference, we first augment our data to add two extra dimensions, and the complete data are now written as $\mathcal{C} = \{(\boldsymbol{u}_i, k_i, z_i), i = 1, 2, \dots, n \mid \boldsymbol{u}_i \geq 0, k_i \in [1, +\infty), z_i > 0\}$. Moreover, the likelihood of point i, w.r.t. the model (3.2) for the complete data is

$$L_{\mathbf{P},\mathbf{v},\mathbf{w}}(\boldsymbol{u}_i, k_i, z_i) = \frac{w_{k_i}}{r_{k_i}} I(z_i \leq r_{k_i}) c_{\mathbf{P}_{\mathbf{k_i}}, v_{k_i}}(\boldsymbol{u}_i). \tag{3.4}$$

Here, $\{r_k\}_{k=1}^{\infty}$ is a deterministic positive sequence decreasing to zero with respect to k, which can be set by us. For example, $r_k = \exp(-\alpha k)$ or $\tilde{r}_k = \beta\gamma^{-\eta k}$, where $\alpha, \beta, \gamma, \eta$ are predefined hyperparameters. $\mathbf{P}, \mathbf{v}, \mathbf{w}$ are the collections of $\mathbf{P}_k, v_k$ and w_k for all k respectively.

The rationale for introducing the constant sequence $\{r_k\}_{k=1}^{\infty}$ is as follows. For the convenience of implementation, the number of candidate clusters that we could potentially allocate to each sample point should be finite. This can be realized by decreasing r_k to zero, as there exists $k \geq k^*$ such that $r_k < z_i$ in (3.4). More discussion on this point is available in Kalli et al. (2011). The validity of the augmentation (3.4) is clarified by the following proposition.

Proposition 1. *The marginal density of $\boldsymbol{u}_i$ with respect to $L_{\mathbf{P},\mathbf{v},\mathbf{w}}(\boldsymbol{u}_i, k_i, z_i)$ is* (3.2).

Proof. We interchange the integral and summation if necessary

$$\begin{aligned}\int_{\mathbb{R}^+} \sum_k \frac{w_k}{r_k} I(z_i \le r_k) c_{\mathbf{P_k}, v_k}(\boldsymbol{u}_i) dz_i &= \sum_k \int_{\mathbb{R}^+} \frac{w_k}{r_k} I(z_i \le r_k) c_{\mathbf{P_k}, v_k}(\boldsymbol{u}_i) dz_i \\ &= \sum_k w_k c_{\mathbf{P_k}, v_k}(\boldsymbol{u}_i) = (3.2).\end{aligned}$$

□

Therefore, sampling from the augmented variables $(\boldsymbol{u}_i, k_i, z_i)_{i=1}^n$ and taking the marginal value of $\boldsymbol{u}_i$ is equivalent to sampling from (3.2). To realize the posterior sampling stated at the end of Section 3.2, we sample from $p(\mathbf{P}, \mathbf{v}, \mathbf{w}, \{k_i\}_{i=1}^n, \{z_i\}_{i=1}^n \mid \mathcal{U})$. We utilize the following Gibbs-MH procedure:

Initialization: In our simulation, we assume K^0, which is the initial number of different components with non-zero weight in (3.2), to be 5. This value should be sufficient for most cases, but we can always lift this restriction to have more clusters. We set r_k in the equation (3.4) to be $r_k = (1-\kappa)\kappa^{k-1}$ for $k = 1, 2, \ldots, 10, \ldots$ according to Kalli et al. (2011), and we choose $\kappa = 0.1$ in our simulation. For $i = 1, 2, \ldots, n$, we initialize k_i to a randomly sampled integer from 1 to K^0, and we set $\mathbf{P}_i^0 = \mathrm{I}, z_i^0 \sim \mathrm{U}(0, r_i)$, and $V_i \sim \mathrm{Beta}(1, \alpha)$, where α can be determined by solving the approximation $\mathbb{E}(K \mid \alpha, n) \approx \alpha \ln(1 + \frac{n}{\alpha})$, as indicated by Antoniak (1974). Here, K refers to the number of mixture components in the data. Therefore, if $\alpha = 0.5$ and $n = 500$, the expected number of groups would be approximately two. Other methods to determine with α are also available, such as hierarchical gamma priors (Escobar and West, 1995) and empirical Bayes (McAuliffe et al., 2006). The prior of the degrees of freedom $\boldsymbol{v}$ should be set with care. It is generally not easy to estimate the degrees of freedom (Sun et al., 2020), especially when we have very few data or if the distribution is not sharp enough. Therefore, we assign a relatively strong prior to it to avoid large oscillations in the MCMC sampling. Hence, we let $v_i \sim 1 + \mathrm{Gamma}(1, 1)$ for all i as priors. These priors of v_i are efficient, especially for heavy-tailed financial data.

For data $D_n = (\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_n)$, we transform them into copula data using

$$\boldsymbol{u}_i = (u_{i1}, u_{i2}, \ldots, u_{id}) = (F_1(y_{i1}), F_2(y_{i2}), \ldots, F_d(y_{id}))$$

for $\mathcal{U} = \{\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_n\}$. Where the margins $F_1, F_2, \ldots, F_d$ must be estimated, we use (3.3).

Step 1: We use Metropolis–Hasting (M-H) steps for updating the correlation matrix. The natural choice would be to use the inverse Wishart distribution $S \sim \mathcal{IW}(\Sigma, d+2)$,

where d is the dimension of the copula model, $E(S) = \Sigma$ as per Wu et al. (2015). However, we found the mixing rate of the MCMC to be slow, especially for high dimensional data. Hence, we adopted the following approach of Danaher and Smith (2011):

For every component $k = 1, 2, \ldots, \max_{q=1,2,\ldots,n} k_q^{t-1}$ and for $\tau = 2, 3, \ldots, d$. $j = 1, \ldots, \tau - 1$. By denoting t the current iteration and $t-1$ the last iteration. We iterated the following steps.

Step 1.1: Propose $l_{\tau j}^{new} \sim \mathcal{N}(l_{\tau j}^{t-1}, 0.1^2)$, $\mathbf{P^{new}} = \text{diag}^{-\frac{1}{2}}(\Sigma)\Sigma\text{diag}^{-\frac{1}{2}}(\Sigma)$, where $\Sigma^{-1} = L^{\text{new}}(L^{\text{new}})^T$. Matrix L is a lower triangular matrix with $(L)_{\tau\tau} = 1$ and $(L)_{\tau j} = l_{\tau j}$ for $\tau > j, \tau = 1, 2, \ldots, d$. We update each $l_{\tau j}$ individually.

Step 1.2: Calculate the acceptance $a_\Sigma = \max\{1, \frac{c_k(\mathcal{U}_k|\mathbf{P^{new}}, v_k^{t-1}) p_0(l_{\tau j}^{new}) p(l_{\tau j}^{t-1}|l_{\tau j}^{new})}{c_k(\mathcal{U}_k|\mathbf{P}^{t-1}, v_k^{t-1}) p_0(l_{\tau j}^{t-1}) p(l_{\tau j}^{new}|l_{\tau j}^{t-1})}\}$, where p_0 is the prior density $\mathcal{N}(0, 0.5^2)$, $\mathcal{U}_k = \{\boldsymbol{u}_i \in \mathcal{U} \mid k_i^{t-1} = k, \ \forall i\}$ and p is the random walk proposal.

Step 1.3: Accept $\mathbf{P}_k^t = \mathbf{P}_k^{\mathbf{new}}$ with probability a_Σ; otherwise, reject it so that $\mathbf{P}_k^t = \mathbf{P}_k^{t-1}$ and $L_k^t = L_k^{t-1}$.

We repeat these steps until we have first exhausted all possible j, τ and then exhausted all k.

Step 2: We simulate

$$p(v_k^t \mid \mathbf{v}_{1,2,\ldots,k-1}^t, \mathbf{v}_{k+1,\ldots}^{t-1}, \mathbf{P}^t, \mathbf{w}^{t-1}, \{k_i^{t-1}\}_{i=1}^n, \{z_i^{t-1}\}_{i=1}^n, \mathcal{U}), k = 1, 2, \ldots, \max_{i=1,2,\ldots,n} k_i^{t-1}$$

The proposal of v_k follows the argument from the supplementary material of Frühwirth-Schnatter and Pyne (2010) by letting

$$\log(v_k^{new} - 1) \sim \text{U}\big(\log(v_k^{t-1} - 1) - \epsilon, \log(v_k^{t-1} - 1) + \epsilon\big).$$

We set $\epsilon = 0.1$, but this can be tuned. Therefore, for every component that has points assigned to it. We can perform a simulation as follows:

Step 2.1: Propose v_k so that $\log(v_k^{new} - 1) \sim \text{U}\big(\log(v_k^{t-1} - 1) - \epsilon, \log(v_k^{t-1} - 1) + \epsilon\big)$.

Step 2.2: Calculate the acceptance $a_{vk} = \max\{1, \frac{c_k(\mathcal{U}_k|\mathbf{P}^{\text{t}}, v_k^{\text{new}}) p_{\text{Gamma}}(v_{\text{new}} - 1)(v_k^{\text{new}} - 1)}{c_k(\mathcal{U}_k|\mathbf{P}^{\text{t}}, v_k^{\text{t-1}}) p_{\text{Gamma}}(v^{t-1} - 1)(v_k^{\text{t-1}} - 1)}\}$.

Step 2.3: Accept $v_k^t = v_k^{\text{new}}$ with probability a_v; otherwise, we let $v_k^t = v_k^{t-1}$.

Step 3: We have the following update for $z_i, i = 1, 2, \ldots, n$.

$$p(z_i^t \mid z_{1,2,\ldots,i-1}^t, z_{i+1,i+2,\ldots,n}^{t-1}, \mathbf{P}^t, \mathbf{v}^t, \mathbf{w}^{t-1}, \{k_i^{t-1}\}_{i=1}^n, \mathcal{U}) \sim \text{U}(0, (1-\kappa)\kappa^{k_i^{t-1}-1}).$$

Step 4: We perform the following update for $k_i, i = 1, 2, \ldots, n$

$$p(k_i^t = k \mid k_{1,2,\ldots,i-1}^t, k_{i+1,i+2,\ldots,n}^{t-1}, \mathbf{P}^t, \mathbf{v}^t, \mathbf{w}^{t-1}, \{z_i^t\}_{i=1}^n, \mathcal{U}) \propto c_k(\boldsymbol{u}_i | v_k^t, \mathbf{P}_k^t) r_k^{-1} w_k^{t-1} I(z_i^t \leq r_k).$$

After Step 4, we remove the redundant states with no assigned data point. If points assign new states, the corresponding copula parameters are sampled from the priors.

Step 5: We perform the following update for $V_k, k = 1, 2, \ldots, \max_{i=1,2,\ldots,n} k_i^t$, where $\max_{i=1,2,\ldots,n} k_i^t$ denotes the maximum number of components allocated at the current round:

$$V_k^t \sim \text{Beta}(1 + \sum_{i=1}^n I(k_i^t = k), \alpha + \sum_{i=1}^n I(k_i^t > k)).$$

We can then update w_k as follows:

$$w_k^t = V_k \prod_{s=1}^{k-1} (1 - V_s).$$

Step 6: The components should be ranked according to their weights in descending order, and relabelling should be performed accordingly. A new set of copula data $\boldsymbol{u}^*$ can be sampled from the current round of parameters $c(\cdot; \boldsymbol{v}^t, \boldsymbol{w}^t, \boldsymbol{P}^t)$. To obtain the data from the estimated multivariate distribution, we invert the copula data $\boldsymbol{u}^*$ using (3.3).

We iterate Steps 1–6 until the maximum number of iterations has been reached.

3.3.2 Sampling distributions

This part clarifies the sampling distributions used in Steps 2–4 by proving the corresponding formulations. First, we show that by proposing the log-uniform distribution in Step 2.1, the M-H acceptance probability a_v in Step 2.2 follows.

Proposition 2. *Let the prior of $v_k \sim 1 + Gamma(1, 1)$ for all k. In M-H Steps 2.1 and 2.2, the proposal distribution of v_k^{new} follows $\log(v_k^{new} - 1) \sim U\big(\log(v_k^{t-1} - 1) - \epsilon, \log(v_k^{t-1} - 1) + \epsilon\big)$. Then, the acceptance rate a_{vk} obtained in the M-H steps follows Step 2.2.*

Proof. The acceptance rate follows $a_{vk} = \max\{1, \frac{c_k(\mathcal{U}_k | P^{\text{t}}, v_k^{\text{new}}) p_{\text{Gamma}}(v_{\text{new}} - 1) p(v_k^{t-1} | v_k^{\text{new}})}{c_k(\mathcal{U}_k | P^{\text{t}}, v_k^{\text{t-1}}) p_{\text{Gamma}}(v^{t-1} - 1) p(v_k^{\text{new}} | v_k^{t-1})}\}$. Furthermore, for the point v' sampled from the above proposal, we have

$$p(v' \mid v) = p_{\text{U}}(\log(v' - 1) \mid v) | \frac{d \log(v' - 1)}{dv'} | = \frac{1}{2\epsilon(v' - 1)},$$

inside the corresponding interval. Hence, the result of Step 2.2 follows. □

The next propositions explain the sampling distributions in Steps 3–4.

Proposition 3. *The conditional sampling distribution of z_i follows*

$$p(z_i^t \mid z_{1,2,\dots,i-1}^t, z_{i+1,i+2,\dots,n}^{t-1}, \mathbf{P}^t, \mathbf{v}^t, \mathbf{w}^{t-1}, \{k_i^{t-1}\}_{i=1}^n, \mathcal{U}) \sim U(0, r_{k_i}),$$

where $r_{k_i} = (1-\kappa)\kappa^{k_i - 1}$ in Step 3.

The conditional sampling distribution of k_i is

$$p(k_i^t = k \mid k_{1,2,\dots,i-1}^t, k_{i+1,i+2,\dots,n}^{t-1}, \mathbf{P}^t, \mathbf{v}^t, \mathbf{w}^{t-1}, \{z_i^t\}_{i=1}^n, \mathcal{U}) \propto c_k(\boldsymbol{u}_i | v_k^t, \mathbf{P}_k^t) r_k^{-1} w_k^{t-1} I(z_i^t \le r_k).$$

Furthermore, this follows a discrete categorical distribution

$$k_i \sim \mathbf{Cat}(p_1, p_2, \dots, p_{k^*}),$$

where $p_i = p(k_i^t = i)$ and k^* depends on r_k.

Proof. From (3.4), we have the following:

$$\begin{aligned} p(z_i^t \mid z_{1,2,\dots,i-1}^t, z_{i+1,i+2,\dots,n}^{t-1}, \mathbf{P}^t, \mathbf{v}^t, \mathbf{w}^{t-1}, \{k_i^{t-1}\}_{i=1}^n, \mathcal{U}) &\propto p(\boldsymbol{u}_i, k_i^{t-1}, z_i^t \mid \mathbf{P}^t, \mathbf{v}^t, \mathbf{w}^{t-1}) \\ &\propto I(z_i \le r_{k_i^{t-1}}). \end{aligned}$$

This yields a uniform distribution from 0 to $r_{k_i^{t-1}}$. Similarly,

$$\begin{aligned} p(k_i^t = k \mid k_{1,2,\dots,i-1}^t, k_{i+1,i+2,\dots,n}^{t-1}, \mathbf{P}^t, \mathbf{v}^t, \mathbf{w}^{t-1}, \{z_i^t\}_{i=1}^n, \mathcal{U}) &\propto p(\boldsymbol{u}_i, k_i^t, z_i^t \mid \mathbf{P}^t, \mathbf{v}^t, \mathbf{w}^{t-1}) \\ &\propto c_k(\boldsymbol{u}_i | v_k^t, \mathbf{P}_k^t) r_k^{-1} w_k^{t-1} I(z_i^t \le r_k). \end{aligned}$$

Here, k_i^t is chosen from $[1, +\infty)$ for our infinite models. However, because r_k decreases to zero as k approach positive infinity, we take $k^* + 1 = \min\{k : r_k < z_i^t\}$. When $k > k^*$, we have $p(k_i^t = k \mid \cdots) = 0$. Hence, we complete the proof with

$$p_k = \frac{c_k(\boldsymbol{u}_i | v_k^t, \mathbf{P}_k^t) r_k^{-1} w_k^{t-1} I(z_i^t \le r_k)}{\sum c_j(\boldsymbol{u}_i | v_j^t, \mathbf{P}_j^t) r_j^{-1} w_j^{t-1} I(z_i^t \le r_j).}$$

□

Finally, we prove the sampling distribution of Step 5.

Proposition 4. *With the prior of* $V_k \sim \text{Beta}(1, \alpha)$, $w_k = V_k \prod_{s=1}^{k-1}(1 - V_s)$, *the conditional distribution* $p(V_k^t \mid \cdots)$ *follows Step 5.*

Proof. We denote $\boldsymbol{k}^t = (k_1^t, k_2^t, \ldots, k_n^t)$, $\boldsymbol{V} = (V_1, V_2, \ldots)$. Therefore,

$$
\begin{aligned}
&p(V_k \mid \mathbf{v}^t, \boldsymbol{V}_{\setminus k}, \mathbf{P}^t, \boldsymbol{k}^t, \mathcal{U}) = p(V_k \mid \boldsymbol{V}_{\setminus k}, \boldsymbol{k}^t) \propto p(\boldsymbol{k}^t \mid \boldsymbol{V}) p_0(V_k) \\
&\propto V_k^0 (1 - V_k)^{\alpha - 1} \prod_{i=1}^{n} w_{k_i} \propto V_k^0 (1 - V_k)^{\alpha - 1} \prod_{\{i: k_i \geq k\}} w_{k_i} \\
&\propto V_k^0 (1 - V_k)^{\alpha - 1} \prod_{\{i: k_i = k\}} V_k \prod_{i: k_i > k} (1 - V_k) \propto V_k^{\sum_{i=1}^n I(k_i = k)} (1 - V_k)^{\alpha + \sum_{i=1}^n I(k_i > k)}
\end{aligned}
$$

This follows the density function of $\text{Beta}(1 + \sum_{i=1}^n I(k_i^t = k), \alpha + \sum_{i=1}^n I(k_i^t > k))$. □

3.3.3 Label-switching problems

It is well known that the MCMC steps of mixture models will cause the problem of label switching (Gelman et al., 2013, Section 22.3). This is basically caused by the identifiability of the latent clustering variables, which in our cases are $k_i, \forall i = 1, 2, \ldots, n$. For a mixture model of K components, there are $K!$ equivalent ways of labeling these components. It is as if we have five different baskets, and we can therefore number these baskets in 5! equivalent ways. To mitigate the label-switching problem, we add a post-processing step after every epoch. That is, after we finish Steps 1–5, we add an extra step to relabel the clusters as per their weighting percentage w_k so that the cluster with a lower weight w_k is always ranked beneath one with a higher weight. If the label switching is not fully resolved, further criteria can be considered to distinguish between groups.

3.4 Numerical simulations

3.4.1 Data with known margins

To verify the practicability of the algorithm, we first work directly with the copula data. This corresponds to the case when we have exact knowledge of the marginal distributions. Hence, for a d dimensional point $\boldsymbol{y}_i = (y_{i1}, y_{i2}, \ldots, y_{id})$, the copula data can be obtained without errors by the transform $u_{ij} = F_j(y_{ij}), j = 1, 2, \ldots, d$. Therefore, we directly sample

points from the mixed three-dimensional t copula distribution of

$$C_{\text{mix}}(u_1, u_2, u_3) = w_1 C_{\mathbf{P_1}, v_1}(u_1, u_2, u_3) + w_2 C_{\mathbf{P_2}, v_2}(u_1, u_2, u_3) \tag{3.5}$$

with

$$\big((\mathbf{P}_1)_{12}, (\mathbf{P}_1)_{13}, (\mathbf{P}_1)_{23}\big) = (0.8, -0.6, -0.5), v_1 = 3, w_1 = 0.7$$

and

$$\big((\mathbf{P}_2)_{12}, (\mathbf{P}_2)_{13}, (\mathbf{P}_2)_{23}\big) = (-0.5, -0.7, 0.4), v_2 = 1.5, w_2 = 0.3.$$

We further simulate points from the three-component mixed t copula

$$C_{\text{mix3}}(u_1, u_2, u_3) = w_1 C_{\mathbf{P_1}, v_1}(u_1, u_2, u_3) + w_2 C_{\mathbf{P_2}, v_2}(u_1, u_2, u_3) + w_3 C_{\mathbf{P_3}, v_3}(u_1, u_2, u_3) \tag{3.6}$$

with

$$\big((\mathbf{P}_1)_{12}, (\mathbf{P}_1)_{13}, (\mathbf{P}_1)_{23}\big) = (0.6, -0.5, -0.4), v_1 = 4, w_1 = 0.7;$$
$$\big((\mathbf{P}_2)_{12}, (\mathbf{P}_2)_{13}, (\mathbf{P}_2)_{23}\big) = (-0.7, -0.8, 0.5), v_2 = 1.6, w_2 = 0.2;$$

and

$$\big((\mathbf{P}_3)_{12}, (\mathbf{P}_3)_{13}, (\mathbf{P}_3)_{23}\big) = (0.3, 0.6, 0.5), v_3 = 5, w_3 = 0.1.$$

To give a graphical sense of the mixture data, we take approaches similar to those in Burda and Prokhorov (2014). We plot the marginal densities of our simulated dependence patterns (3.6) for (u_1, u_2) and (u_2, u_3). In addition, we transformed our copula data back into distribution data using the inversion of the standard normal marginal CDF $\Phi^{-1}(u)$. Figure 3.1 displays the results. As the plot reveals, dependence patterns are mixed from different directions. This becomes clearer when we observe the multi-modal features of the transformed data in the second row of Figure 3.1, which is a good incentive for us to apply a mixture model approach. We therefore use our algorithm to estimate the parameter from the sampled data. The full posterior parameter distribution was simulated using Gibbs-MH 10,000 times. Figure 3.2 depicts the MCMC sampling trace plots of the component weightings estimated from the simulated data of setting (3.5) using 1000 points. The labels of the weightings are ranked by their values such that $w_1 \geq w_2 \geq w_3 \geq \cdots \geq \cdots$. All weightings except for those of the first two components decrease to zero after around 3000 iterations,

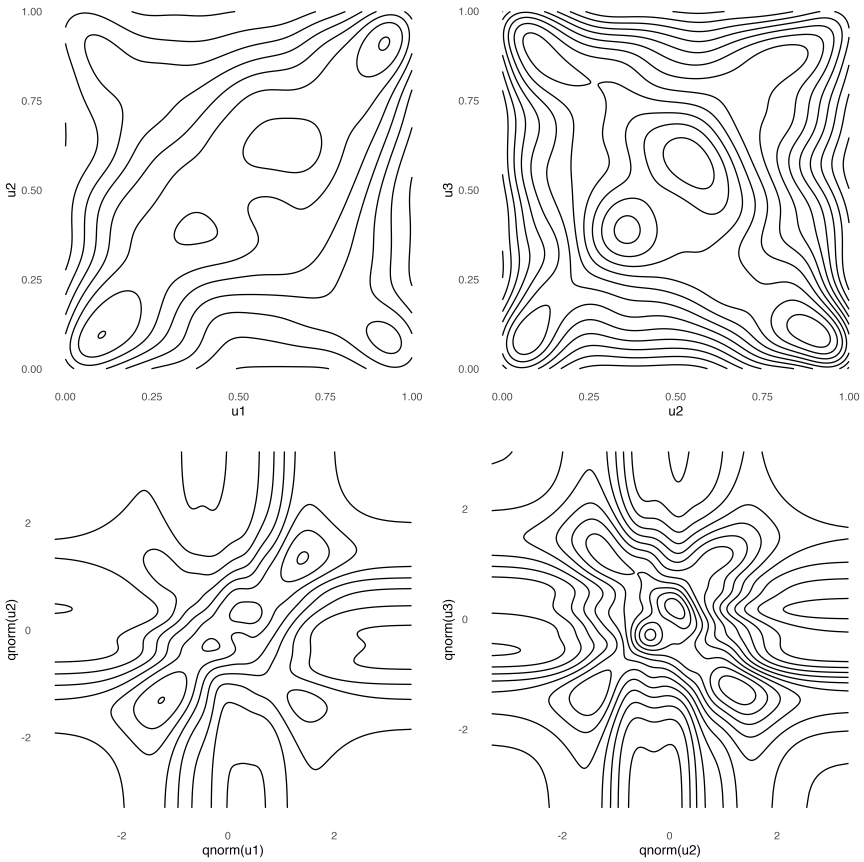

Figure 3.1: Contour plots of the marginal densities for the simulation example (3.6) (u_1, u_2) and (u_2, u_3) (above) and the transformed density plots $\big(\Phi^{-1}(u_1), \Phi^{-1}(u_2)\big)$ and $\big(\Phi^{-1}(u_2), \Phi^{-1}(u_3)\big)$, when the standard normal marginal distribution $\Phi(\cdot)$ are used.

showing the validity of the algorithm in determining the number of groups. For the 1000-point simulation of (3.5), we have the mean estimation of $(\hat{w}_1, \hat{w}_2, \hat{w}_3) = (0.72, 0.27, 0.005)$ with 95% confidence intervals of $(0.61, 0.81), (0.19, 0.39)$, and $(0, 0.03)$ respectively. Figure 3.3 provides the corresponding trace plots and density plots of the degrees of freedom for the first two mixture components, with the corresponding posterior mean $\hat{v_1} = 2.95, \hat{v_2} = 1.55$, and the 95% credible regions $\mathcal{I}_1 = (2.29, 4.29)$ and $\mathcal{I}_2 = (1.15, 2.33)$, respectively, for the dataset with 1000 observations.

The full results of the parameter estimations are reported in Table 3.1, and we compare them with the results of L-BFGS-B MLE, which is embedded in R software using `fitCopula()` in `library(Copula)` (Hofert et al., 2022). We fixed the number of mixture components to be the *true* value of two.

In Table 3.1, it is evident that Bayesian samplers yield estimations comparable to those of

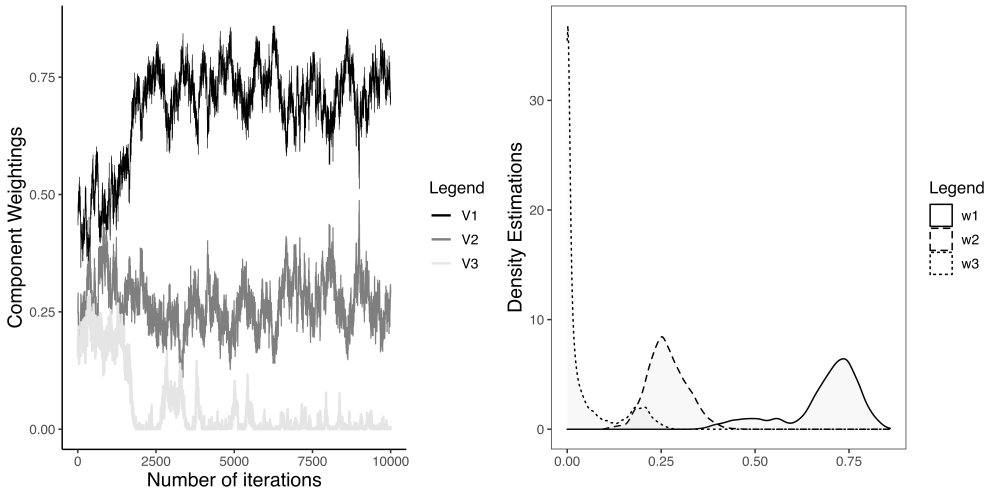

Figure 3.2: Left: Trace plot of the weighting parameters w_1, w_2, w_3. Right: Corresponding density estimation of the weighting

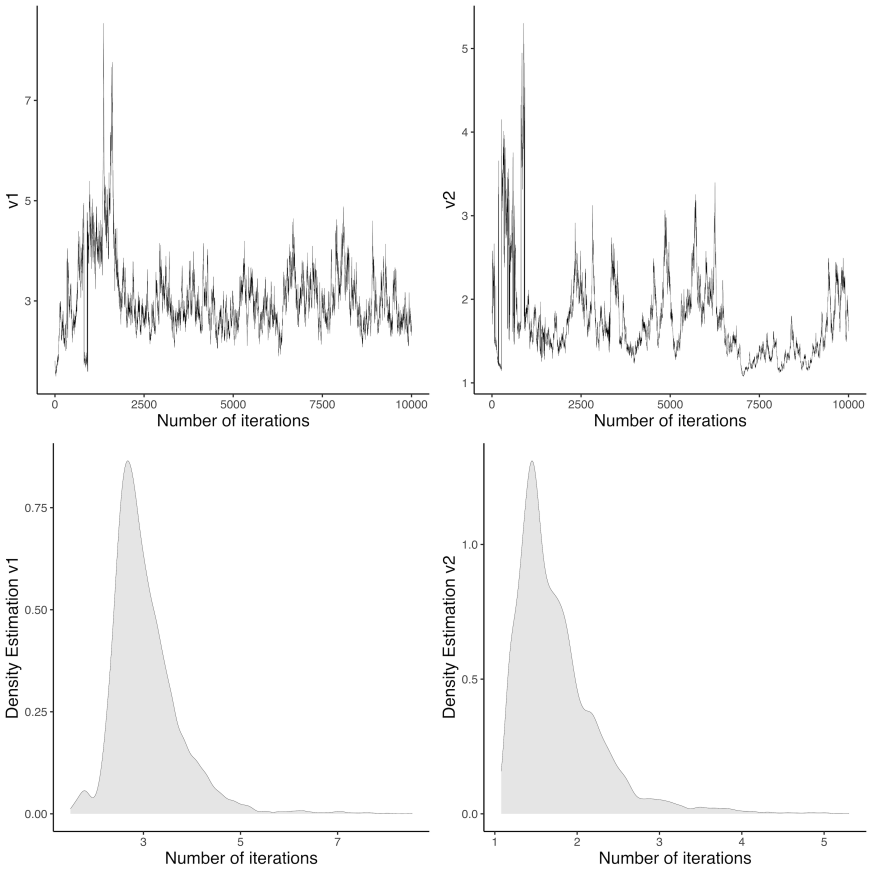

Figure 3.3: Degrees-of-freedom posterior estimations of the first two mixture components

Table 3.1: Comparison of the Gibbs-MH algorithm with L-BFGS-B MLE for model 3.5

	Gibbs-MH						L-BFGS-B MLE			
	Mean estimate			95% CI			Point estimate			Truth
N	200	500	1000	200	500	1000	200	500	1000	
$(P_1)_{12}$	0.70	0.76	0.78	$(0.57, 0.81)$	$(0.67, 0.84)$	$(0.72, 0.83)$	**0.71**	**0.78**	**0.79**	0.8
$(P_1)_{13}$	-0.54	**-0.62**	**-0.63**	$(-0.67, -0.36)$	$(-0.69, -0.54)$	$(-0.69, -0.57)$	**-0.57**	-0.63	**-0.63**	-0.6
$(P_1)_{23}$	**-0.40**	**-0.51**	**-0.53**	$(-0.56, -0.20)$	$(-0.61, -0.40)$	$(-0.62, -0.43)$	**-0.40**	-0.52	-0.54	-0.5
$(P_2)_{12}$	**-0.38**	**-0.65**	**-0.51**	$(-0.82, -0.22)$	$(-0.81, -0.46)$	$(-0.76, -0.18)$	-0.8	- 0.75	-0.56	-0.5
$(P_2)_{13}$	**-0.53**	-0.63	**-0.66**	$(-0.84, 0.03)$	$(-0.76, -0.41)$	$(-0.76, -0.52)$	**-0.53**	**-0.66**	**-0.66**	-0.7
$(P_2)_{23}$	**0.47**	**0.53**	**0.58**	$(0.12, 0.71)$	$(0.30, 0.72)$	$(0.34, 0.76)$	0.72	0.58	0.64	0.4
v_1	1.94	2.6	**2.96**	$(1.39, 2.80)$	$(1.87, 3.87)$	$(2.29, 4.19)$	**2.17**	**2.7**	2.95	3
v_2	2.72	**1.87**	**1.55**	$(1.26, 5.56)$	$(1.32, 2.84)$	$(1.15, 2.33)$	**1.2**	2.4	1.60	1.5
w_1	**0.76**	**0.75**	**0.72**	$(0.56, 0.90)$	$(0.67, 0.84)$	$(0.61, 0.81)$	0.83	0.76	0.73	0.7
w_2	**0.21**	0.21	**0.27**	$(0.09, 0.35)$	$(0.16, 0.33)$	$(0.19, 0.38)$	0.17	**0.24**	**0.27**	0.3

Starting values of $((P_1)^0_{12}, (P_1)^0_{13}, (P_1)^0_{23}, (P_2)^0_{12}, (P_2)^0_{13}, (P_2)^0_{23}, v^0_1, v^0_2, w^0_1, w^0_2) = (0, 0, 0, 0, 0, 0, 3, 3, 0.7, 0.3)$. The concentration parameter $\alpha = 0.2$. The confidence interval (CI) of the MLE is not given because it was not included in the R package. The bold numbers are the best estimate in terms of the absolute loss between the Bayesian method and the MLE method on the same dataset.

the maximum likelihood estimation (MLE) based methods implemented by `fitCopula()` in the `library(Copula)` package. This holds true for both the absolute and mean squared loss of the parameters, even when the correct number of mixture components is provided for MLE estimation.

Furthermore, we note that the MLE-based estimations are highly sensitive to the initial values in these cases. Variations in initial values can lead to substantially different final estimations. In our application, we give the correct weightings $(w_1, w_2) = (0.7, 0.3)$ as the starting value for the MLE estimations. We found that some other weighting values could lead to substantially distant results. On the other hand, the high-dimensional setting makes it challenging to obtain asymptotic variances of the parameters for the MLEs, and the efficacy of bootstrapping methods for computing confidence intervals of estimators becomes uncertain.

In contrast, Bayesian posterior sampling naturally incorporates the credible region, presenting a more dependable approach for parameter estimation.

We conducted an additional simulation using model (3.6). More specifically, we sampled $N = 2000$ and $N = 3000$ points from the three-component mixture copula. The sampler was run for 10,000 full iterations, with the initial 8000 iterations discarded as burn-in points. Similar to the previous analysis, we compare the results with those obtained using the MLE approach. Table 3.2 presents the sampler's results alongside the MLE method's results, which were obtained using the R library. Furthermore, Figure 3.4 compares the true model data with the posterior sampling distributions of $N = 3000$ data points.

Our results for this simulation case (Table 3.2 and Figure 3.4) reveal that the MLE method produces more precise estimates when the sample size is $N = 2000$. However, as the number of observations increases to $N = 3000$, the MLE's and the Bayesian sampler exhibit

Table 3.2: Comparison of the Gibbs-MH algorithm with L-BFGS-B MLE for the model 3.6

	Gibbs-MH				L-BFGS-B MLE		
	Mean estimate		95% CI		Point estimate		Truth
N	2000	3000	2000	3000	2000	3000	
$(P_1)_{12}$	0.58	**0.60**	$(0.52, 0.83)$	$(0.55, 0.64)$	**0.61**	0.61	0.6
$(P_1)_{13}$	-0.35	-0.39	$(-0.42, -0.30)$	$(-0.54, -0.45)$	**-0.42**	**-0.50**	-0.5
$(P_1)_{23}$	-0.31	**-0.39**	$(-0.40, -0.25)$	$(-0.45, -0.34)$	**-0.37**	**-0.41**	-0.4
$(P_2)_{12}$	-0.60	**-0.69**	$(-0.73, -0.38)$	$(-0.82, -0.54)$	**-0.67**	-0.73	-0.7
$(P_2)_{13}$	-0.74	**-0.79**	$(-0.81, -0.66)$	$(-0.87, -0.71)$	**-0.79**	-0.82	-0.8
$(P_2)_{23}$	0.45	**0.53**	$(0.24, 0.60)$	$(0.39, 0.69)$	**0.50**	0.57	0.5
$(P_3)_{12}$	-0.16	**0.15**	$(-0.69, 0.66)$	$(-0.14, 0.43)$	**0.25**	0.12	0.3
$(P_3)_{13}$	0.17	**0.53**	$(-0.68, 0.72)$	$(0.22, 0.77)$	**0.87**	0.68	0.6
$(P_3)_{23}$	0.13	0.38	$(-0.48, 0.73)$	$(0.12, 0.62)$	**0.64**	**0.53**	0.5
v_1	3.35	3.93	$(2.80, 4.05)$	$(3.30, 4.88)$	**3.8**	**3.98**	4
v_2	1.28	1.50	$(1.15, 2.30)$	$(1.15, 2.33)$	**1.44**	**1.62**	1.6
v_3	2	2.82	$(1.00, 3.96)$	$(1.79, 4.97)$	**4.94**	**4.2**	5
w_1	0.77	**0.73**	$(0.69, 0.83)$	$(0.69, 0.78)$	**0.74**	0.75	0.7
w_2	**0.22**	**0.18**	$(0.14, 0.23)$	$(0.17, 0.30)$	**0.18**	**0.18**	0.2
w_3	0.01	**0.08**	$(0.00, 0.03)$	$(0.05, 0.12)$	**0.07**	0.07	0.1

Starting values of

$$((P_1)_{12}^0, (P_1)_{13}^0, (P_1)_{23}^0, (P_2)_{12}^0, (P_2)_{13}^0, (P_2)_{23}^0, (P_3)_{12}^0, (P_3)_{13}^0, (P_3)_{23}^0, v_1^0, v_2^0, v_3^0, w_1^0, w_2^0, w_3^0) = (0, 0, 0, 0, 0, 0, 0, 0, 0, 3, 3, 3, 0.3, 0.4, 0.3).$$

The concentration parameter $\alpha = 0.2$. The confidence interval (CI) of the MLE is not given because it was not included in the R package. The bold numbers are the best estimate in terms of the absolute loss between the Bayesian method and the MLE method on the same dataset.

similar performances. Note that the MLE approach relies on the correct specification of the number of components, whereas the Bayesian algorithm automatically determines the number of groups. Therefore, the MLE method possesses additional information, which gives it an advantage. Furthermore, as presented in Table 3.1, the MLE-based approach is sensitive to the initial starting point and does not inherently provide confidence intervals. In contrast, the Bayesian sampler does not suffer from these issues.

3.4.2 Data with unknown margins

The previous section discusses the case when the margins of the data are known from external knowledge. We therefore focused on estimating the copulas. For most applications, margins are unknown and need to be estimated using the data. In this scenario, we use non-parametric estimation (3.3) for the marginal distributions before we perform the MCMC samplings.

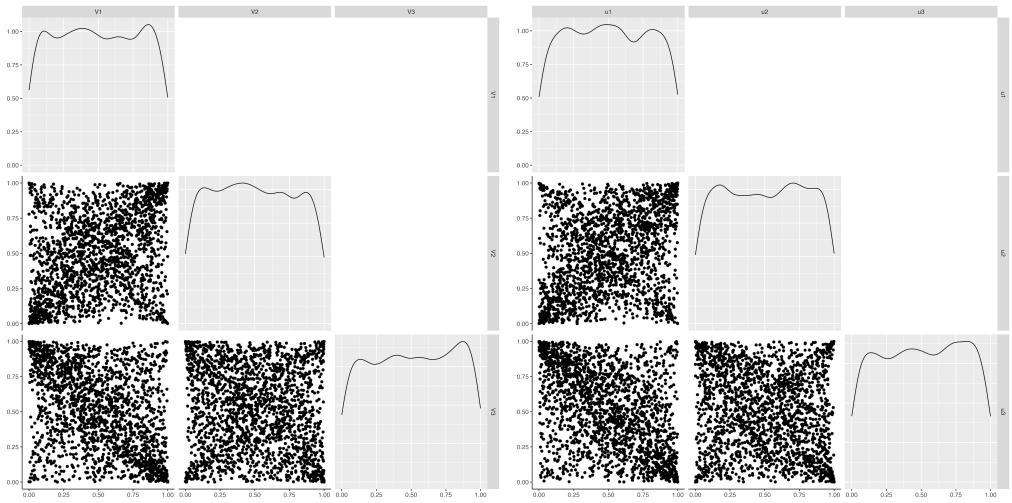

Figure 3.4: Left: Pair plots of i.i.d. observations from the true model (3.6). Right: The posterior model using 3000 observations generated by the Bayesian sampler after the burn-in stage.

In this part, to produce our synthetic data, we first sample the copula points directly using the copula function (3.5). We further set the true underlying margins of our synthetic data to be the standard normal distributions so the sampled points from the copula are inverse transformed by the quantile function of the standard normal distribution marginally. Note that in practice, the data we gather are $\big(y_{i1}, y_{i2}, \ldots, y_{id}\big) = \big({F_1}^{-1}(u_{i1}), {F_2}^{-1}(u_{i2}), \ldots, {F_d}^{-1}(u_{id})\big),\ i = 1, 2, \ldots, n,$ where the margins $F_1, F_2, \ldots, F_d$ are unknown. We therefore proceed with our general procedure by first estimating the margins empirically using (3.3) and then estimate the copula using the data $(\hat{u}_{i1}, \hat{u}_{i2}, \ldots, \hat{u}_{id}),\ i = 1, 2, \ldots, n$ obtained from (3.3). For the sake of comparison, the copula data used are the same data used to produce Table 3.1, and all aspects of the algorithms are kept the same as the previous part except for extra marginal estimations. We display the results in Table 3.3. Both the MCMC and MLE methods fail to obtain good estimations for $n = 200$, in contrast to the previous experiment (Table 3.1). In the current $n = 200$ case, the MCMC approach obtains the right signs of the correlations for the first mixture component but fails to be successful for the other parameters while the MLE method gives the same estimation for both components, even when we start the iteration with the true values of the weighting parameters. As the sample size increases, we obtain similar performances for both approaches when compared with the results in Table 3.1, indicating the validity of using empirical estimations (3.3) when margins are unknown. Again, we find the performances of the MLE approach highly dependent on the correct specifica-

tions and initial starting values, and we give good initial points because we know the true parameters. However, this can be difficult in real applications.

Table 3.3: Comparison of the Gibbs-MH algorithm with L-BFGS-B MLE for the model 3.5 with margins being the standard normal distributions. Starting values of $((P_1)^0_{12}, (P_1)^0_{13}, (P_1)^0_{23}, (P_2)^0_{12}, (P_2)^0_{13}, (P_2)^0_{23}, v_1^0, v_2^0, w_1^0, w_2^0) = (0, 0, 0, 0, 0, 0, 3, 3, 0.7, 0.3)$ were used. The concentration parameter $\alpha = 0.2$. The confidence interval (CI) of the MLE is not given because it was not included in the R package. The margins are treated as unknown when conducting the estimation, and non-parametric estimations (3.3) were used for the margins.

	Gibbs-MH						L-BFGS-B MLE			
	Mean estimate			95% CI			Point estimate			Truth
N	200	500	1000	200	500	1000	200	500	1000	
$(P_1)_{12}$	0.47	0.78	0.79	$(0.31, 0.66)$	$(0.70, 0.84)$	$(0.74, 0.83)$	0.2	0.79	0.80	0.8
$(P_1)_{13}$	-0.55	-0.64	-0.62	$(-0.68, -0.43)$	$(-0.72, -0.56)$	$(-0.68, -0.55)$	-0.2	-0.60	-0.64	-0.6
$(P_1)_{23}$	-0.23	-0.54	-0.54	$(-0.42, -0.06)$	$(-0.65, -0.42)$	$(-0.62, -0.45)$	-0.20	-0.50	-0.56	-0.5
$(P_2)_{12}$	0.08	-0.68	-0.51	$(-0.55, 0.70)$	$(-0.87, -0.30)$	$(-0.68, -0.29)$	0.2	- 0.80	-0.44	-0.5
$(P_2)_{13}$	-0.06	-0.64	-0.69	$(-0.63, 0.75)$	$(-0.77, -0.46)$	$(-0.77, -0.54)$	-0.2	-0.7	-0.64	-0.7
$(P_2)_{23}$	0.03	0.53	0.61	$(-0.62, 0.59)$	$(0.25, 0.76)$	$(0.43, 0.72)$	-0.2	0.63	0.60	0.4
v_1	1.19	3.01	3.59	$(1.07, 1.38)$	$(2.10, 4.58)$	$(2.68, 4.83)$	2.39	3	1.43	3
v_2	2.48	2.64	1.6	$(1.13, 4.75)$	$(1.54, 4.20)$	$(1.32, 2.02)$	2.39	3	1.43	1.5
w_1	0.93	0.74	0.72	$(0.58, 1.00)$	$(0.66, 0.84)$	$(0.64, 0.78)$	0.70	0.77	0.70	0.7
w_2	0.07	0.22	0.27	$(0.00, 0.41)$	$(0.13, 0.31)$	$(0.20, 0.36)$	0.30	0.23	0.30	0.3

3.5 Real data analysis

We collected the daily closing prices of the Shanghai Stock Exchange 50 Index (SSE 50) and Shenzhen Stock Exchange 100 Index (SZSE 100) from January 3, 2018, to March 17, 2023; 1263 trading days in total. We converted these daily closing prices P_t into log returns using the formula $r_{t+1} = \log(P_{t+1}/P_t)$. The summary statistics of the log returns over the aforementioned period are presented in Table 3.4.

Table 3.4: Summary statistics of the daily log return between January, 2018, and March, 2023.

	Min	Max	Skewness	Kurtosis	JB test p value	ADF test p-value
Shanghai	-0.07	0.07	-0.14	5.66	$< 2.2 \times 10^{-16}$	0.01
Shenzhen	-0.09	0.05	-0.37	4.98	$< 2.2 \times 10^{-16}$	0.01

The daily log return data exhibit the strong sign of a heavy tail, with the Jarque–Bera (JB) test significantly rejecting the null hypothesis of normality. It is generally recognized that the Shanghai and Shenzhen indexes have a strong correlation. Our analysis aims to

determine if there is any change in the dependence pattern, especially one due to the occurrences of the COVID-19 pandemic after 2020. This situation naturally fits our model as we do not know how many mixtures exist. However, although the ADF test rejects the existence of the unit root, the log return of these indexes does not form an i.i.d. series. This is illustrated in Figure 3.5, where the plot of the absolute log return of the Shanghai index clearly shows some auto-correlation, although the log return plot on the left is seemingly not correlated. Note that a Ljung–Box test with five lags was rejected for the squared log return of the Shanghai and Shenzhen indexes.

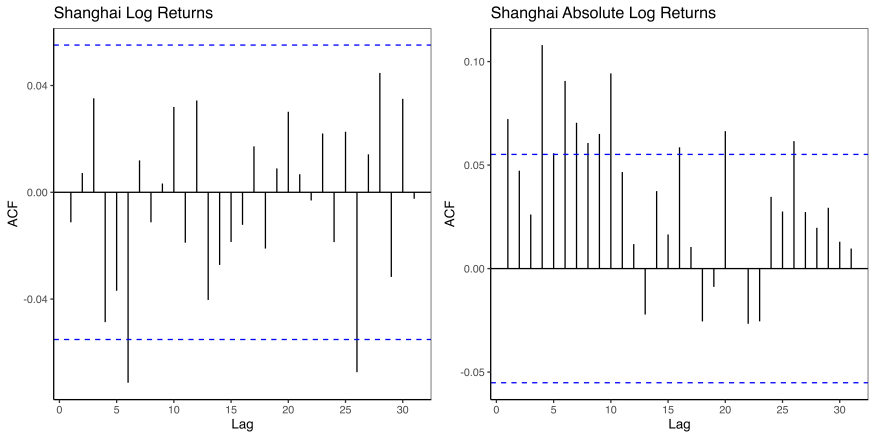

Figure 3.5: Auto-correlation of the log return (left) and the absolute log return (right) of the Shanghai composite index for Jan. 2018 to March 2023

To produce the i.i.d. series of log returns, we model each marginal series using a t-GARCH(1,1), i.e.,

$$\begin{aligned} r_t &= \sigma_t \epsilon_t \quad \epsilon_t \sim t_v \\ \sigma_t^2 &= \beta + \alpha_0 \sigma_{t-1}^2 + \alpha_1 r_{t-1}^2. \end{aligned} \tag{3.7}$$

In particular, the innovation of our model (3.7) ϵ_t is fit by the standard t-distribution, which is more consistent with the heavy tails of the market returns. After we fit the GARCH model to each marginal return series, we extract the standard innovation by $\varepsilon_i = (\frac{r_{i1}}{\sigma_{i1}}, \frac{r_{i2}}{\sigma_{i2}}, \ldots, \frac{r_{in}}{\sigma_{in}})$, $i = 1, 2$, where n refers to the total number of observations, which is $n = 1263$. The index i indicates whether the series is from Shanghai or Shenzhen. The p-values of the Ljung–Box test with five lags after scaling for ε_1^2 and ε_2^2 were 0.31 and 0.75, respectively, indicating the sign of weak or no temporal correlation. We further applied the marginal empirical distribution $u_i = \hat{F}_i(\epsilon_i) = \frac{1}{n+1} \sum_{t=1}^{n} I\{\hat{\epsilon}_t \leq \epsilon_i\}$ so that the marginal data was transformed

into the copula data to enable the model training. Figure 3.6 displays the highly positive correlated pattern with heavy extreme dependence of the transformed copula data, which supports the credibility of using t copula structures. We applied our infinite mixture model

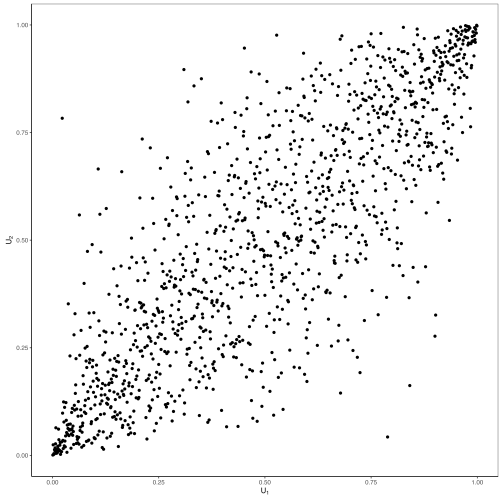

Figure 3.6: Transformed copula data from the Shanghai and Shenzhen log returns

to the transformed copula data, running 10,000 full iterations for each parameter. We considered the initial 8000 iterations to be the burn-in period, as done previously. Figure 3.7 displays the plot of the posterior predictive points (transformed back to the original scale using the inverse of the empirical functions) against the original standardized residuals. The predictive points generated by the samplers closely resemble the original residuals, indicating a good fit using our model.

Moreover, the posterior proportion of w_1 remains consistently high, with a mean of 97.63% and a standard deviation of 0.015, strongly suggesting that only one component is needed. Figure 3.8 presents the trace plot of v_1 and the correlation $(P_1)_{12}$, which represent the parameters of the first mixture component.

To further validate our approach, we again compared the estimation with the MLE approach of a single-component t copula, which was implemented in R using the library mentioned previously. The results are listed in Table 3.5. The close results of the two methods confirm that the parameter estimation of the infinite t copula mixture has comparable quality to that of the MLE estimation embedded in the standard package `library(Copula)` in R. In this case, the Bayesian approach has more flexibility in determining the number of mixture components, with a shorter confidence region regarding the degree of freedom v.

Our analysis of the results reveals that the dependence structure of the Shenzhen–

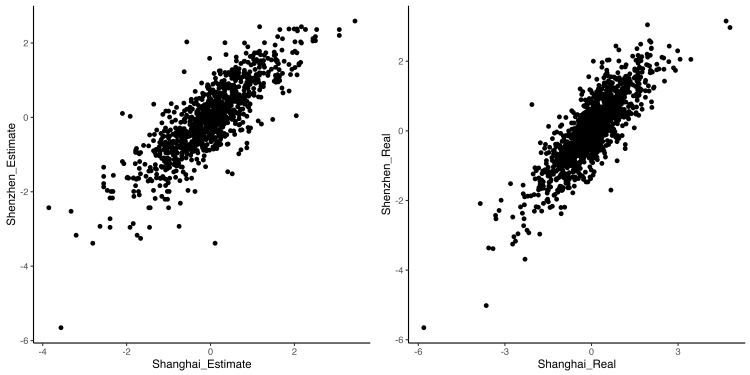

Figure 3.7: Samples from the estimated Bayesian sampler (left) and data from the real standardized residuals (right)

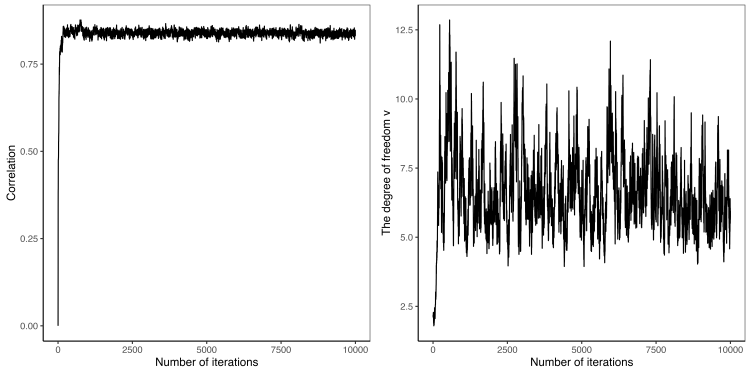

Figure 3.8: Trace plot of $(P_1)_{12}$ and v_1.

Table 3.5: Comparison of the Gibbs-MH algorithm with L-BFGS-B MLE on Shanghai-Shenzhen stock data. "CI" refers to the confidence interval.

	Gibbs-MH		L-BFGS-B MLE of a single t copula	
	Mean estimate	95% Credible Region	Point estimate	95% CI
$(P_1)_{12}$	0.84	$(0.82, 0.85)$	0.84	$(0.83, 0.86)$
v_1	6.2	$(4.6, 8.6)$	8.32	$(4.1, 12.5)$
Log-likelihood	789.5		790.3	

Shanghai stock market remained unchanged before and after the COVID-19 pandemic. The correlation remains consistently high throughout the period, with only one mode, and the tail dependence is significant, as evidenced by the degrees-of-freedom parameters. Estimating the degrees of freedom is crucial in financial applications, as it captures the co-movement of financial assets during extreme losses (or returns). A normal copula cannot grasp this

feature.

To express this more mathematically, we define the extreme tail loss as $\Lambda = \lim_{u\to 0} \mathbb{P}(X_1 \leq F_1^{-1}(u) \mid X_2 \leq F_2^{-1}(u))$, where $F_1(X_1)$ and $F_2(X_2)$ represent the marginal return distributions of individual assets. In terms of the copula, we have $\Lambda = \lim_{u\to 0} \frac{\mathbb{P}(F_1(X_1)\leq u, F_2(X_2)\leq u)}{\mathbb{P}(F_2(X_2)\leq u)} = \lim_{u\to 0} \frac{C(u,u)}{u}$. Citing results from (McNeil et al., 2015, p.249-250), we find that $\Lambda = 0$ for the normal copula and $\Lambda = 2F_{t,v+1}\big(-\sqrt{(v+1)(1-r)/(1+r)}\big)$ for the t copula with degrees of freedom v, correlation parameter r, and distribution function $F_{t,v+1}(\cdot)$ referring to the corresponding t-distribution. Consequently, according to the posterior mean estimation parameters, the extreme lower tail dependence between the SSE 50 and SZSE 100 stocks is 45%. In other words, there is a 45% probability that the SZSE 100 index will suffer a significant loss when the SSE 50 experiences an extreme loss. This type of information would be lost if a normal copula instead of a t copula were to be used. Therefore, it is advisable to model the dependence of financial assets using a t copula when compared with the normal conterpart.

3.6 Concluding remakrs

In this chapter, we have developed the infinite Student t copulas using the non-parametric Bayesian approach. More specifically, the stick-breaking process was used to construct our models, and the application in the Shanghai-Shenzhen market was performed.

The construction of t infinite mixture copulas would be useful, especially in the area of financial risk management. This is because the ability to capture the extreme dependence among financial instruments is essential in this field. Further extensions can be done to consider the mixture of skew t copulas, which can make the detection of skewness possible.

Chapter 4

Copula hidden Markov model with unknown number of states

4.1 Introduction

The hidden Markov model (HMM) (Rabiner and Juang, 1986) has been widely applied as a tool for modeling the regime switches in finance (De Angelis and Paas, 2013; Dias et al., 2015; Nguyen, 2018). Therefore, extending the mixture of copulas into the setting of hidden Markov models can ensure that the transition matrix among states can be effectively estimated. The copula-HMM structure is a convenient tool for modeling multivariate time series when cross-sectional dimensions are correlated.

Studies have been conducted on copula–HMM models. Derrode and Pieczynski (2016); Yu (2017) applied the Gaussian copula–HMM model to the field of signal processing. Ötting et al. (2021); Ötting and Karlis (2022) analyzed football games data using the HMM models with the emission distributions formed by Frank, Clayton, and AMH copulas. The number of states was selected based on the best AIC and BIC scores. Oflaz et al. (2023) used the binomial copulas along with the HMM structure for chronic disease analysis. Zimmerman et al. (2022) incorporated the inference for margins (IFM) techniques into the EM method and proposed the efficient IFM–EM approach for the copula–HMM model estimations.

However, in the classic HMM model, the number of states have to be specified as a hyperparameter, which is inconvenient for many tasks, as this information is usually not available as prior knowledge. To determine the number of states from the data, conventional approaches use the AIC or BIC to compare models with different hyperparameters. This

involves repeated estimation of the models from the data.

In contrast, we employ an infinite hidden Markov model (Beal et al., 2001; Van Gael et al., 2008; Fox et al., 2011; Maheu and Yang, 2016) to enable automatic state number determination for copula–HMM structures. This is realized by utilizing the hierarchical Dirichlet process (HDP) of Teh et al. (2006) to construct an infinite hidden Markov model. Tekumalla and Bhattacharyya (2016) mentioned the concept of HDP–HMM–copula models in their introduction but the latter implementation and application directly focused on the multivariate normal.

In this study, we introduce inference approaches for a copula-based infinite hidden Markov model. This is followed by numerical simulations for validation, and real data applications to detect different dependence modes between major stock markets.

4.1.1 Copula–iHMM model

We can write the multimodal data in the mixture form as

$$F(x_1, x_2, \ldots, x_d) = \sum_{i=1}^{K} w_i F^i(x_1, x_2, \ldots, x_d), \quad \sum_{i=1}^{K} w_i = 1, \ w_i \geq 0.$$

Where $F^i, i = 1, 2, \ldots, K$ represents the different distributions (McLachlan et al., 2019).

The density form of the aforementioned distribution (if it exists) can be expressed as

$$\begin{aligned} f(x_1, x_2, \ldots, x_d) &= \sum_{i=1}^{K} w_i f^i(x_1, x_2, \ldots, x_d) \\ &= \sum_{i=1}^{K} w_i c^i(F_1(x_1), F_2(x_2), \ldots, F_d(x_d); \boldsymbol{\Theta}_i) \prod_{j=1}^{d} f_j^i(x_j; \boldsymbol{\alpha}_j^i). \end{aligned} \tag{4.1}$$

The second equality is obtained by applying

$$f(x_1, x_2, \ldots, x_d) = c(F_1(x_1), F_2(x_2), \ldots, F_d(x_d)) \prod_{j=1}^{d} f_j(x_j).$$

to every component of the mixture $f^i, i = 1, 2, \ldots, K$, where f_j^i is the corresponding j^{th} margin.

For time-series data, utilizing the model (4.1) directly is inappropriate. Therefore, we introduce a copula-based hidden Markov model as follows.

Let the stochastic process be defined as $\{(H_t, \boldsymbol{O}_t) \mid t = 1, 2, 3, \dots\}$, where $H_t \in \{1, 2, 3, \dots, K\}$ are discrete-valued hidden states such that the Markov property satisfies

$$p(H_t = k \mid H_{t-1} = j, H_{t-2}, H_{t-3}, \dots) = p(H_t = k \mid H_{t-1} = j) = p_{jk}.$$

We denote the $K \times K$ matrix P as a homogeneous transition matrix such that $(P_t)_{ij} = (P)_{ij} = p_{ij},\ t = 1, 2, \dots, T$. In particular, p_{jk} does not change with time for all state transitions between H_t and H_{t-1}.

By contrast, $\boldsymbol{O}_t = (x_1^t, x_2^t, \dots, x_d^t)$ follows multivariate emission distributions such that their densities are

$$\boldsymbol{O}_t \mid H_t = k \ \sim\ c^k\big(F_1^k(x_1^t), F_2^k(x_2^t), \dots, F_d^k(x_d^t); \boldsymbol{\Theta_k}\big) \prod_{j=1}^{d} f_j^k(x_j^t; \boldsymbol{\alpha}_j^k). \tag{4.2}$$

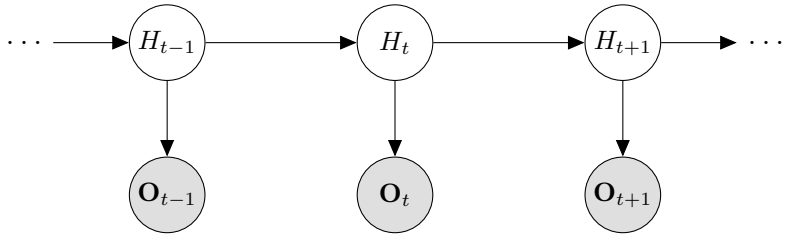

Figure 4.1: Hidden Markov model as a directed graphical model. The hidden states are denoted H_t and the observed emission states are $\mathbf{O}_t$.

Figure 4.1 shows a graphical representation of the HMM structure. We represent our observed distributions using the copulas (4.2) and Sklar's theorem.

Furthermore, we aimed to simultaneously infer the number of states K, copula parameters $\boldsymbol{\Theta}_k$, marginal parameters $\boldsymbol{\alpha}_j^k$, and hidden states $H_1, H_2, \dots, H_T$ considering T-length observable data

$$\mathbf{O}_{1:T} = (\mathbf{O}_1, \mathbf{O}_2, \dots, \mathbf{O}_T).$$

We realized this by using the hierarchical Dirichlet process (Teh et al., 2006), a non-parametric Bayesian framework, to extend the hidden Markov model to the infinite hidden Markov model; this ensures that the number of states can be estimated alongside other parameters of interest.

4.2 Priors of Dirichlet processes and Hierarchical Dirichlet processes in mixture modeling

4.2.1 Dirichlet priors

For the self contained purpose, the Dirichlet process is denoted in this chapter by $DP(\alpha, B_0)$, where α is the positive concentration parameter and B_0 is the base probability measure. Sample $B_1 \sim DP(\alpha, B_0)$ is an atomic probability measure such that, for any finite segmentation of the sample space Ω such that $\cup_{i=1}^{n} E_i = \Omega$ and $E_i \cap E_j = \emptyset \ \forall i, j$, we have

$$\Big(B_1(E_1), B_1(E_2), \ldots, B_1(E_n)\Big) \sim \textbf{Dirichlet}(\alpha B_0(E_1), \alpha B_0(E_2), \ldots, \alpha B_0(E_n)).$$

Dirichlet$(\cdot)$ denotes Dirichlet probability distribution.

Therefore, for finite mixture distributions within the parametric family of densities, $\{f(\cdot;\theta) \mid \theta \in \Theta\}$. We set the DP as a prior such that the mixture is as follows:

$$\begin{aligned} B_1 &\sim DP(\alpha, B_0) \\ f_{mix} &= \int_{\theta \in \Theta} f(\cdot \mid \theta) dB_1. \end{aligned}$$

Let us revisit the stick-breaking representation of Dirichlet priors of Sethuraman (1994), the generation process for B_1 is

$$\begin{aligned} &\theta_i \sim B_0, v_i \sim \text{Beta}(1, \alpha) \\ &w_1 = v_1, w_i = v_i \prod_{j=1}^{i-1} (1 - v_j) \\ &i = 1, 2, 3, \ldots. \end{aligned}$$

where $\text{Beta}(1, \alpha)$ is the beta distribution with the parameters 1 and α. The stick-breaking process is now denoted as $\{w_i\}_{i=1}^{\infty} \sim \textbf{SBP}(\alpha)$.

Furthermore, the latent group label for every point i is now denoted by $k_i \in \{1, 2, 3, \ldots\}$.

The generation process of samples $O_i, i = 1, 2, \ldots, T$ can be represented as

$$
\begin{aligned}
&k_1, k_2, k_3, \ldots, k_T \sim \textbf{Multinomial}(w_1, w_2, \ldots) \\
&\theta_j \sim B_0,\ j = 1, 2, \ldots, K \\
&O_i \sim f^{k_i}(\cdot \mid \theta_{k_i}),\ i = 1, 2, \ldots T.
\end{aligned}
$$

4.2.2 Hierarchical Dirichlet priors

The hierarchical Dirichlet (Teh et al., 2006; Gelman et al., 2013) adds a different layer to the introduced Dirichlet process. For Dirichlet processes with concentration parameters α, β, we obtain:

$$
\begin{aligned}
&B_1 \sim DP(\alpha, B_0), \\
&B_j \sim DP(\beta, B_1), j = 1, 2, 3, \ldots
\end{aligned}
$$

Therefore, the sample from the first layer of DP was used as the base measure for later layers. One significant feature of this structure is that the later layers share the same parameter, $\theta_i \sim B_0$ as the top layer. This enables a dependent inference among levels. Van Gael (2012) provides the stick-breaking construction of the hierarchical Dirichlet process as

$$
\begin{aligned}
\{w_i\}_{i=1}^{\infty} \sim &\textbf{SBP}(\alpha) \\
\big(p_{j1}, p_{j2}, \ldots, p_{jk}, \sum_{m \geq k+1} p_{jm}\big) \sim &\textbf{Dirichlet}\big(\beta w_1, \beta w_2, \ldots \beta w_k, \beta(\sum_{m \geq k+1} w_k)\big), \forall k \\
\theta_j \sim &B_0 \\
j = &1,2,3,\ldots.
\end{aligned}
$$

The second line of the above representation is denoted by $\textbf{SBP}(\beta, \{w_i\}_{i=1}^{\infty})$ for simplicity.

The infinite hidden Markov model can be naturally constructed from the representation because we can treat $\boldsymbol{p}_j = (p_{j1}, p_{j2}, p_{j3}, \ldots)^T, j = 1, 2, 3, \ldots$ as the transition vector from state j to any other state. We employ the notations in Section 4.1.1. The IHMM in the

setting of the mixture copulas can be expressed as

$$
\begin{aligned}
&\{w_i\}_{i=1}^{\infty} \sim \mathbf{SBP}(\alpha),\\
&\boldsymbol{p}_j \sim \mathbf{SBP}(\beta, \{w_i\}_{i=1}^{\infty}), (\boldsymbol{\Theta}_j, \boldsymbol{\alpha}^j) \sim B_0 \ j = 1, 2, 3, \dots\\
&H_t = s_2 \mid H_{t-1} = s_1 \sim p_{s_1 s_2}, \ H_1 = 1, \ t = 1, 2, \dots, T\\
&\boldsymbol{O}_t \mid H_t = k \ \sim \ c^k\big(F_1^k(x_1^t), F_2^k(x_2^t), \dots, F_d^k(x_d^t); \boldsymbol{\Theta}_k\big) \prod_{l=1}^{d} f_l^k(x_l^t; \boldsymbol{\alpha}_l^k),
\end{aligned}
$$

where $\boldsymbol{\alpha}^j = (\boldsymbol{\alpha}_1^j, \boldsymbol{\alpha}_2^j, \dots, \boldsymbol{\alpha}_d^j)^T$.

From the current construction, $E(p_{ij}) = E[E(p_{ij} \mid \{w_i\}_{i=1}^{\infty})] = E[w_j]$. This enables fast switching between states (Dufays, 2016), which is disadvantageous for some applications. Fox et al. (2011) proposed the sticky infinite hidden Markov model to mitigate the issue. This is realized by adding a sticky probability to the current state j such that

$$
\begin{aligned}
&\big(p_{j1}, p_{j2}, \dots, p_{jk}, \sum_{m \geq k+1} p_{jm}\big) \sim\\
&\mathbf{Dirichlet}\big(\beta w_1/(\beta+\kappa), \beta w_2/(\beta+\kappa), \dots, (\beta w_j + \kappa)/(\beta+\kappa), \dots, \beta(\sum_{m \geq k+1} w_k)/(\beta+\kappa)\big).
\end{aligned}
$$

Therefore, the current state becomes more persistent for $\kappa > 0$, and the original iHMM recovers when $\kappa = 0$. For financial analysis, retaining the original iHMM construction might help to ensure that sudden changes in market conditions can be detected (Dufays, 2016)

4.3 Bayesian parameters estimation

In this section, we present a posterior Bayesian inference methodology for the copula–iHMM.

Considering sequential data $\mathbf{O}_{1:T} = (\mathbf{O}_1, \mathbf{O}_2, \dots, \mathbf{O}_T)$, our aim is to sample from the posterior

$$
p\bigg((\boldsymbol{\Theta_i})_{i=1}^{\infty}(\boldsymbol{\alpha}^i)_{i=1}^{\infty}; (w_i)_{i=1}^{\infty}; (H_i)_{i=1}^{T}, P, \alpha, \beta \mid O_{1:T}\bigg).
$$

A major difficulty in performing posterior inference of the iHMM is its infinite structure. In particular, the transition from one state to an infinite state must be considered. This problem can be overcome using the slice sampler proposed in Walker (2007); Van Gael et al. (2008).

Particularly, we augment the posterior distribution with another set of ancillary variables

$(u_t)_{t=1}^T$ such that

$$p\big(H_t = s_2, u_t \mid H_{t-1} = s_1, P\big) = \mathbb{I}(0 < u_t < p_{s_1 s_2}).$$

where $\mathbb{I}(\cdot)$ denotes an indicator function.

The augmentation maintains the marginal transition probability as

$$\int p\big(H_t = s_2, u_t' \mid H_{t-1} = s_1, P\big) du_t' = p_{s_1 s_2}.$$

Meanwhile, as

$$p\big(H_t = s_2 \mid u_t, H_{t-1} = s_1, P\big) \propto p\big(H_t = s_2, u_t \mid H_{t-1} = s_1, P\big) = \mathbb{I}(0 < u_t < p_{s_1 s_2}).$$

This prevents us from considering the states s_i, s_j with $p_{s_i s_j} \leq u_t$ at the point O_t.

Therefore, after augmentation, the posterior distribution of interest becomes

$$p\bigg((\boldsymbol{\Theta_i})_{i=1}^{\infty}, (\boldsymbol{\alpha}^i)_{i=1}^{\infty}, (w_i)_{i=1}^{\infty}, (H_i)_{i=1}^{T}, (u_i)_{i=1}^{T}, P, \alpha, \beta \mid O_{1:T}\bigg).$$

Let $\mathcal{J} = \{(\boldsymbol{\Theta_i})_{i=1}^{\infty}, (\boldsymbol{\alpha}^i)_{i=1}^{\infty}, (w_i)_{i=1}^{\infty}, (H_i)_{i=1}^{T}, (u_i)_{i=1}^{T}, P, \alpha, \beta\}$ be the joint parameters of interest.

We used Gibbs within Metropolis–Hasting approaches to perform posterior sampling. The steps can be broken down as follows after we set up the initialization:

1. Sample $p\big((\boldsymbol{\Theta_i})_{i=1}^K, (\boldsymbol{\alpha}^i)_{i=1}^K \mid \mathcal{J}_{\setminus (\boldsymbol{\Theta_i})_{i=1}^K, (\boldsymbol{\alpha}^i)_{i=1}^K}, O_{1:T}\big)$ by first sampling $p\big((\boldsymbol{\alpha}^i)_{i=1}^K \mid \mathcal{J}_{\setminus (\boldsymbol{\alpha}^i)_{i=1}^K}, O_{1:T}\big)$ and then followed by $p\big((\boldsymbol{\Theta_i})_{i=1}^K \mid \mathcal{J}_{\setminus (\boldsymbol{\Theta_i})_{i=1}^K}, O_{1:T}\big)$, where K is the largest number of states by considering the current group label.

2. Sample $p(P \mid \mathcal{J}_{\setminus P}, O_{1:T})$, which consists of sampling $\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots \boldsymbol{p}_K$ from the corresponding posterior Dirichlet distributions.

3. Sample $p\big((w_i)_{i=1}^K \mid \mathcal{J}_{\setminus (w_i)_{i=1}^K}, O_{1:T}\big)$ and the concentration parameters $p\big(\alpha, \beta \mid \mathcal{J}_{\setminus \alpha, \beta}, O_{1:T}\big)$. These samplers can be considered together, and their posterior distributions are related to the Pólya urn representation of the hierarchical Dirichlet process (see Teh et al. (2006); Van Gael (2012); Maheu and Yang (2016)).

4. Sample $p\big((u_i)_{i=1}^T \mid \mathcal{J}_{\setminus (u_i)_{i=1}^T}, O_{1:T}\big)$. This step should be looped until

$$\min_{t=1,2,\ldots,T}(u_t) > \max_{s=1,2,\ldots,K}(p_{sK+1})$$

is satisfied. This is to guarantee that we have considered all possible non-vanished states. For every loop, K should be increased to ensure that $K \leftarrow K + 1$ and the corresponding parameters $(\boldsymbol{p}_K, \boldsymbol{\Theta}_K, \boldsymbol{\alpha}^K, w_K)$ of the newly considered states should be sampled from priors.

5. Sample $p\big((H_i)_{i=1}^T \mid \mathcal{J}_{\setminus(H_i)_{i=1}^T}, O_{1:T}\big)$. Applying the forward–backward techniques of the classic HMM becomes possible after the slice sampler of step 4.

6. Remove the empty states after step 5. Label-switching problems can be adjusted at this step if required. For example, ranking the clusters according to their conditional likelihood usually distinguishes them well. This ranking criterion can be further refined to consider the transition matrix if necessary.

7. Return to step 1 and begin a new iteration. The algorithm is stopped when the maximum number of iterations is reached.

For application in this study, we provide more details for the use of a t-copula with normal margins; however, the framework also applies to other copulas.

Sample $(\boldsymbol{\Theta_i})_{i=1}^K$ and $(\boldsymbol{\alpha}^i)_{i=1}^K$ The sampling of distribution parameters consists of sampling from the copula parameters $(\boldsymbol{\Theta_i})_{i=1}^K$ and from the marginal distributions $(\boldsymbol{\alpha}^i)_{i=1}^K$. Important techniques of inference for margins (IFM) (Genest et al., 1995; Joe and Xu, 1996) in the copula theory can be utilized for estimation. Particularly, the marginal parameters can be estimated separately from the copula structure. For normal margins, with parameters $\boldsymbol{\alpha}_j^i = (\mu_j^i, \sigma_j^i)$ $\forall i = 1, 2, \ldots, K$ and $j = 1, 2 \ldots, d$, we propose parameters according to the conjugate posterior, considering the current round of state labels. For the parameters of the i^{th} group, the particular group data $G_i = \{O_t \mid H_t = i\}$.

We set the prior of any (μ_j^i, σ_j^i) with $i = 1, 2, \ldots, K$ and $j = 1, 2, \ldots, d$ as

$$p_0(\mu_j^i, \tau_j^i) = \text{NIG}(\mu_j^i, \tau_j^i | \mu_0, \lambda_0, \alpha_0, \beta_0),$$

where $\tau_j^i = 1/{\sigma_j^i}^2$, $\mu_0 = 0$, $\lambda_0 = 1$, $\alpha_0 = 0.5$, $\beta_0 = 0.005$.

Set the cardinality of set $|G_i| = n$ and let $\bar{O}$ be the sample mean of the data in the set. The updated parameters for the posterior normal inverse gamma distribution are

$$\mu' = \frac{\lambda_0 \mu_0 + n\bar{O}}{\lambda_0 + n}$$

$$\lambda' = \lambda_0 + n$$

$$\alpha' = \alpha_0 + \frac{n}{2}$$

$$\beta' = \beta_0 + \frac{1}{2}\sum_{j=1}^{n}(O_{i_j} - \bar{O})^2 + \frac{n\lambda_0}{2(\lambda_0 + n)}(\bar{O} - \mu_0)^2$$

where points $O_{i_1}, O_{i_2}, ..., O_{i_n}$ are the data from G_i. Therefore, we sample from the conjugate posterior

$$(\mu_j^{i*}, \sigma_j^{i*}) \sim p^*(\mu_j^i, \tau_j^i | G_i) = \text{NIG}(\mu_j^i, \tau_j^i | \mu', \lambda', \alpha', \beta').$$

This is posterior, considering only the margins. We apply the Metropolis–Hasting steps to accept the proposed $(\mu_j^{i*}, \sigma_j^{i*})$. Therefore,

$$\alpha_{(\mu_j^{i*}, \sigma_j^{i*})} = \frac{\prod_{t \in i_1, i_2, \ldots, i_n} c^i(F_1^i(x_1^t), \ldots, F_j^i(x_j^t; \mu_j^{i*}, \sigma_j^{i*}), \ldots, F_d^i(x_d^t)) f_j^i(x_j^t; (\mu_j^{i*}, \sigma_j^{i*})) p_0(\mu_j^{i*}, \sigma_j^{i*}) p^*(\mu_j^i, \tau_j^i | G_i)}{\prod_{t \in i_1, i_2, \ldots, i_n} c^i(F_1^i(x_1^t), \ldots, F_j^i(x_j^t; \mu_j^i, \sigma_j^i), \ldots, F_d^i(x_d^t)) f_j^i(x_j; (\mu_j^{i*}, \sigma_j^{i*})) p_0(\mu_j^i, \sigma_j^i) p^*(\mu_j^{i*}, \tau_j^{i*} | G_i)}.$$

For the t-copula densities, we must sample the $d \times d$ positive definite correlation matrix R_i and degree-of-freedom parameters ν_i for any group $i = 1, 2, \ldots, K$. The sampling of R_i is obtained from Danaher and Smith (2011), which is the same as the previous chapter.

In particular, we sample one element at a time from the lower triangular matrix L_i such that $(L_i)_{jj} = 1, \ \forall j = 1, 2, \ldots d$. Thereafter, $\Sigma_i^{-1} = L_i L_i^T$ and $R_i = \text{diag}(\Sigma)^{-\frac{1}{2}}(\Sigma)\text{diag}(\Sigma)^{-\frac{1}{2}}$. For row $j = 1, 2, \ldots, d$, the following procedure is looped from column $k = 1, \ldots, j-1$:

(i) Sample from the random walk proposal to update the lower triangular matrix of the current round $(L_i)_{jk}^* \sim \text{N}((L_i)_{jk}, 0.05^2)$.

(ii) Calculate new correlation such that

$$\Sigma_i^{*-1} = {L_i}^* {L_i}^{*T}, R_i^* = \text{diag}(\Sigma)^{-\frac{1}{2}}(\Sigma)\text{diag}(\Sigma)^{-\frac{1}{2}}$$

(iii) The proposal matrix is accepted as per the probability

$$\alpha_{R^*} = \frac{\prod_{t \in i_1, i_2, \ldots, i_n} c^i\big(F_1^i(x_1^t), \ldots, F_d^i(x_d^t); R^*\big) p_0((L_i)_{jk}^*) p^*((L_i)_{jk} | (L_i)_{jk}^*)}{\prod_{t \in i_1, i_2, \ldots, i_n} c^i\big(F_1^i(x_1^t), \ldots, F_d^i(x_d^t); R\big) p_0((L_i)_{jk}^*) p^*((L_i)_{jk}^* | (L_i)_{jk})}.$$

where the prior $p_0 \sim \text{N}(0, 0.5^2)$ and the proposal $p^*((L_i)_{jk}^* | (L_i)_{jk}) \sim \text{N}((L_i)_{jk}, 0.05^2)$.

To sample the degrees of freedom ν_i, we follow the approach of Frühwirth-Schnatter and Pyne (2010) and propose the following:

$$\log(\nu_i^* - 1) \sim \mathrm{U}(\log(\nu_i - 1) - \epsilon, \log(\nu_i - 1) + \epsilon).$$

The prior distribution of ν_i is set to $p_0(\nu_i - 1) \sim \mathrm{Gamma}(1, 1)$. The acceptance rate is

$$\alpha_{\nu^*} = \frac{\prod_{t \in i_1, i_2, \ldots, i_n} c^i\big(F_1^i(x_1^t), \ldots, F_d^i(x_d^t); \nu^*\big) p_0(\nu_i^* - 1)(\nu_i^* - 1)}{\prod_{t \in i_1, i_2, \ldots, i_n} c^i\big(F_1^i(x_1^t), \ldots, F_d^i(x_d^t); \nu\big) p_0(\nu_i - 1)(\nu_i - 1)}.$$

Sampling $p(P \mid \mathcal{J}_{\setminus P}, O_{1:T})$ To sample $\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots \boldsymbol{p}_K$, recall the posterior distributions of the Dirichlet distribution such that for any $i = 1, 2, \ldots, K$

$$(\boldsymbol{p}_{ij})_{j=1}^K, \sum_{j=K+1}^{\infty} (\boldsymbol{p}_{ij}) \mid \mathcal{J}_{\setminus P} \sim \mathbf{Dirichlet}(\beta_1', \beta_2', \ldots, \beta_K', \beta \sum_{j=K+1}^{\infty} w_j)$$
$$\beta_\tau' = \beta w_\tau + \sum_{t=1}^{T-1} \mathbb{I}(H_t = i, H_{t+1} = \tau),\ \tau = 1, 2, \ldots, K.$$

Sampling $p\big((w_i)_{i=1}^K \mid \mathcal{J}_{\setminus (w_i)_{i=1}^K}, O_{1:T}\big)$ The inference of the posterior distribution of $(w_i)_{i=1}^K$ is related to the Pólya urn representation of the hierarchical Dirichlet process hidden Markov model(Teh et al., 2006; Van Gael, 2012; Maheu and Yang, 2016).

Following Teh et al. (2006), we obtain the following posterior sampler:

$$(w_i)_{i=1}^K, \sum_{j=K+1}^{\infty} w_j \sim \mathbf{Dirichlet}(m_{\cdot 1 \cdot}, m_{\cdot 2 \cdot}, \ldots, m_{\cdot K \cdot}, \alpha).$$

In the Pólya urn representation, $(m_{\cdot 1 \cdot}, m_{\cdot 2 \cdot}, \ldots, m_{\cdot K \cdot})$ are the states of the samples drawn from the top-level $DP(\alpha)$.

To simulate $(m_{\cdot 1 \cdot}, m_{\cdot 2 \cdot}, \ldots, m_{\cdot K \cdot})$ in each iteration, we sequentially draw

$$m_{i,j,t} \sim \mathrm{Bin}\big(1, \beta w_j / (t - 1 + \beta w_j)\big)$$
$$t = 1, 2, \ldots, \sum_{\tau=1}^{T-1} \mathbb{I}(H_\tau = i, H_{\tau+1} = j).$$

We denote $\mathrm{Bin}(n, p)$ as the binomial distribution of n trails with probability p. Therefore,

$m_{\cdot j\cdot} = \sum_i \sum_t m_{ijt}$.

Sampling $p\big(\alpha, \beta \mid \mathcal{J}_{\backslash\alpha,\beta}, O_{1:T}\big)$ The sampling of hyperparameters follows directly from Teh et al. (2006); Fox et al. (2011); Maheu and Yang (2016); Dufays (2016).

Proceeding from the last step, set $m_{\cdots} = \sum_{ijt} m_{ijt}$. The sampling of α consists of

$$\tilde{\alpha} \sim \text{Bin}\big(1, m_{\cdots}/(m_{\cdots} + \alpha)\big),\ \tilde{\tilde{\alpha}} \sim \text{Beta}(\alpha + 1, m_{\cdots})$$
$$\alpha \sim \text{Gamma}(\eta_0 + \sum_{j=1}^{K} \mathbb{I}(m_{\cdot j\cdot} > 0) - \tilde{\alpha}, \gamma_0 - \log \tilde{\tilde{\alpha}}).$$

where η_0, γ_0 are hyperparameters. We set them to 1 and 5 for the later simulations and real-data experiments, respectively.

The sampling of β consists of

$$\tilde{\beta}_i \sim \text{Bin}\bigg(1, \frac{\sum_j \sum_{t=1}^{T-1} \mathbb{I}(H_t = i, H_{t+1} = j)}{\sum_j \sum_{t=1}^{T-1} \mathbb{I}(H_t = i, H_{t+1} = j) + \beta}\bigg),\ i = 1, 2, \ldots, K$$
$$\tilde{\tilde{\beta}}_i \sim \text{Beta}\bigg(\beta + 1, \sum_j \sum_{t=1}^{T-1} \mathbb{I}(H_t = i, H_{t+1} = j)\bigg),\ i = 1, 2, \ldots, K$$
$$\beta \sim \text{Gamma}(\eta_1 + m_{\cdots} - \sum_i \tilde{\beta}_i, \gamma_1 - \sum_i \log \tilde{\tilde{\beta}}_i).$$

We set the hyperparameters $\eta_1 = 1, \gamma_1 = 5$.

Sampling $p\big((u_i)_{i=1}^T \mid \mathcal{J}_{\backslash(u_i)_{i=1}^T}, O_{1:T}\big)$ The ancillary parameters should be sampled according to

$$p\big(u_1 \mid \mathcal{J}_{\backslash(u_i)_{i=1}^T}, O_{1:T}\big) \sim \text{U}(0, w_{H_1})$$
$$p\big(u_t \mid \mathcal{J}_{\backslash(u_i)_{i=1}^T}, O_{1:T}\big) \sim \text{U}(0, p_{H_{t-1}H_t}),\ t = 2, 3, \ldots, T$$

This procedure is repeated if $\min_{t=1,2,\ldots,T}(u_t) < \max_{s=1,2,\ldots K}(p_{sK+1})$ and if new states arise, and all the corresponding parameters of the copulas and margins should be sampled from the priors for this new state. Furthermore, the new state $K + 1$ has an initial transition matrix

$$\big(p_{j1}, p_{j2}, \ldots, p_{jK+1}, \sum_{m \geq K+2} p_{jm}\big) \sim \textbf{Dirichlet}\big(\beta w_1, \beta w_2, \ldots \beta w_{K+1}, \beta(\sum_{m \geq K+2} w_m)\big)$$
$$j = K + 1$$

The top level $DP(\alpha)$ corresponds to

$$w_{K+1} = (1 - \sum_{i=1}^{K} w_i) v_{K+1}, v_{K+1} \sim \text{Beta}(1, \alpha).$$

The transition probability from other existing states to state K is set as

$$p_{iK+1} = (1 - \sum_{j=1}^{K} p_{ij}) V_{K+1},\ V_{K+1} \sim \text{Beta}\big(\beta w_{K+1}, \beta(1 - \sum_{i=1}^{K+1} w_i)\big)$$
$$i = 1, 2, \ldots, K.$$

Finally, we update the largest number of states in the current iteration K by adding one.

Sampling $p\big((H_i)_{i=1}^T \mid \mathcal{J}_{\setminus (H_i)_{i=1}^T}, O_{1:T}\big)$ To sample the hidden state $(H_i)_{i=1}^T$, we employed the forward–backward trick of the classical hidden Markov model.

Working backwardly,

$$\begin{aligned} &p(H_t = i \mid H_{t+1:T}, O_{1:T}, u_{1:T}) \propto \\ &p(H_t \mid O_{1:t}, u_{1:t}) p(H_{t+1}, u_{t+1} \mid H_t) = \mathbb{I}(u_{t+1} < p_{H_t H_{t+1}}) p(H_t \mid O_{1:t}, u_{1:t}) \end{aligned}$$

The backward iteration from $t = T, T-1, T-2, \ldots$ enables us to sample the hidden state H_t if we know

$$p(H_t \mid O_{1:t}, u_{1:t}), t = 1, 2, 3, \ldots, T$$

.

These values are computed by working forward. In particular, for $t = 1, 2, 3, \ldots, T$

$$\begin{aligned} p(H_t \mid O_{1:t}, u_{1:t}) &\propto p(H_t, O_t, u_t \mid O_{1:t-1}, u_{1:t-1}) \\ &\propto p(O_t \mid H_t, u_t, O_{1:t-1}, u_{1:t-1}) \sum_S p(H_t, u_t, H_{t-1} = S \mid u_{1:t-1}, O_{1:t-1}) \\ &\propto p(O_t \mid H_t) \sum_S \mathbb{I}(0 < u_t < p_{sH_t}) p(H_{t-1} = s \mid O_{1:t-1}, u_{1:t-1}). \end{aligned}$$

Here,

$$p(H_1 \mid O_1, u_1) \propto p(O_1 \mid H_1) p(H_1 \mid u_1) \propto \mathbb{I}(0 < u_1 < w_{H_1}) p(O_1 \mid H_1).$$

Thereafter, follow Steps 6–7.

4.4 Simulation studies

We conducted simulation studies for two and three hidden Markov model states. On both occasions, we simulated the data points for three different sets of parameters with sequence lengths $T = 1000$ for 2-state models and $T = 2000$ for 3-state models. We used our proposed MCMC sampler with 10000 iterations; the first 8000 points were discarded as burn-in. The last 2000 iterations were used as the posterior means and 95% credible interval calculations. Specifically, we assumed that for each $H_t = k, k = 1, 2, 3$. The synthetic data were obtained from

$$\boldsymbol{O} \mid H_t = k \;\sim\; c_t^k\big(F_1^k(x_1), F_2^k(x_2), \ldots, F_d^k(x_d); R^k, \nu^k\big) \prod_{j=1}^{d} f_j^k(x_j; \mu_j^k, \sigma_j^k).$$

where $c_t(\cdot; R, \nu)$ is the t-copula with the correlation matrix R and degrees of freedom ν. Further, we assumed that the data dimensions $d = 3$. We assumed that the marginal distributions follow $f_j(\mu_j, \sigma_j) \sim \mathrm{N}(\mu_j, \sigma_j)$. To generate the synthetic HMM, we first sampled $H_t = k \mid H_{t-1} = j \sim p_{jk}$. This step generates the copula data

$$(\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_T) \mid H_{1:T} \sim \prod_{i=1}^{T} c_t^{H_i}(\cdot, R^{H_i}, \nu^{H_i}).$$

Thereafter, we inverted the copula data in every dimension using their marginal distributions, $[(\boldsymbol{O}_i)_j \mid H_i = k] = {F_j}^{-1}((\boldsymbol{u}_i)_j; \mu_j^k, \sigma_j^k)$., where ${F_j}^{-1}$ is the inverse normal distribution of group k.

In our simulation studies, we used

$$f^1 = (f_1^1, f_2^1, f_3^1) \sim N(\boldsymbol{\mu}^1 = (0.1, 0.1, 0.1)^T, \Sigma^1 = 0.1^2 I_3),$$

and $f_j^1, j = 1, 2, 3$ as marginal distributions of state one. In addition,

$$f^2 = (f_1^2, f_2^2, f_3^2) \sim N(\boldsymbol{\mu}^2 = (-0.1, -0.1, -0.1)^T, \Sigma^2 = 0.1^2 I_3)$$

are marginal distributions of state 2. Here, I_d is the d-dimensional identity matrix. For

three-state simulations, we set

$$f^1 \sim N(\boldsymbol{\mu}^1 = (0.1, 0.1, 0.1)^T, \Sigma^1 = 0.1^2 I_3)$$
$$f^2 \sim N(\boldsymbol{\mu}^2 = (-0.1, -0.1, -0.1)^T, \Sigma^2 = 0.1^2 I_3)$$
$$f^3 \sim N(\boldsymbol{\mu}^3 = (0, 0, 0)^T, \Sigma^3 = 0.1^2 I_3).$$

Furthermore, the transition matrix for the 2-state and 3-state HMM is set as

$$P = \begin{bmatrix} 0.7 & 0.3 \\ 0.4 & 0.6 \end{bmatrix}$$

and

$$P' = \begin{bmatrix} 0.6 & 0.3 & 0.1 \\ 0.5 & 0.4 & 0.1 \\ 0.7 & 0.1 & 0.2 \end{bmatrix}.$$

Table 4.1: Simulation results of the 2-state copula infinite hidden Markov models with samples $n = 1000$.

Params	Simulation 1 ($T = 1000$)			Simulation 2 ($T = 1000$)			Simulation 3 ($T = 1000$)		
	Mean 1	Credible interval 1	True value 1	Mean 2	Credible interval 2	True value 2	Mean 3	Credible interval 3	True value 3
$(R^1)_{12}$	0.671	$(0.607, 0.724)$	0.7	0.751	$(0.704, 0.793)$	0.8	0.888	$(0.869, 0.907)$	0.9
$(R^1)_{13}$	0.487	$(0.389, 0.571)$	0.6	0.651	$(0.589, 0.701)$	0.7	0.796	$(0.761, 0.830)$	0.8
$(R^1)_{23}$	0.454	$(0.356, 0.545)$	0.5	0.542	$(0.472, 0.610)$	0.6	0.700	$(0.649, 0.742)$	0.7
$(R^2)_{12}$	-0.651	$(-0.703, -0.586)$	-0.7	-0.784	$(-0.825, -0.741)$	-0.8	-0.914	$(-0.930, -0.894)$	-0.9
$(R^2)_{13}$	-0.555	$(-0.618, -0.475)$	-0.6	-0.694	$(-0.750, -0.630)$	-0.7	-0.780	$(-0.812, -0.739)$	-0.8
$(R^2)_{23}$	0.472	$(0.392, 0.543)$	0.5	0.634	$(0.568, 0.691)$	0.6	0.697	$(0.640, 0.741)$	0.7
ν_1	3.000	$(2.370, 3.815)$	3	3.120	$(2.54, 3.91)$	3	3.310	$(2.574, 4.439)$	4
ν_2	3.951	$(2.888, 5.123)$	3	3.800	$(2.85, 5.20)$	4	4.471	$(3.277, 6.109)$	5
p_{11}	0.643	$(0.594, 0.690)$	0.7	0.711	$(0.673, 0.750)$	0.7	0.704	$(0.663, 0.744)$	0.7
p_{12}	0.357	$(0.310, 0.406)$	0.3	0.289	$(0.250, 0.327)$	0.3	0.296	$(0.256, 0.337)$	0.3
p_{21}	0.355	$(0.313, 0.401)$	0.4	0.414	$(0.364.0.467)$	0.4	0.427	$(0.376, 0.478)$	0.4
p_{22}	0.644	$(0.598, 0.686)$	0.6	0.585	$(0.533, 0.635)$	0.6	0.573	$(0.522, 0.624)$	0.6

We report the number of active states K estimated using the copula-iHMM model for the last 2000 MCMC iterations in Figure 4.2. The state numbers are correctly estimated for every case because the models choose the correct specifications under posterior estimations.

Tables 4.1 and 4.2 report the parameter estimation results with the corresponding MCMC 95% credible intervals. For 2-state models, the estimators of the degrees of freedom ν are less precise when compared with the correlation and transition matrices. This is expected because estimations of ν require numerical inversions and the densities are less sensitive to the parameter. However, the true values of ν were all covered by the 95% credible intervals of the MCMC and the components with larger true degrees of freedom obtained larger mean

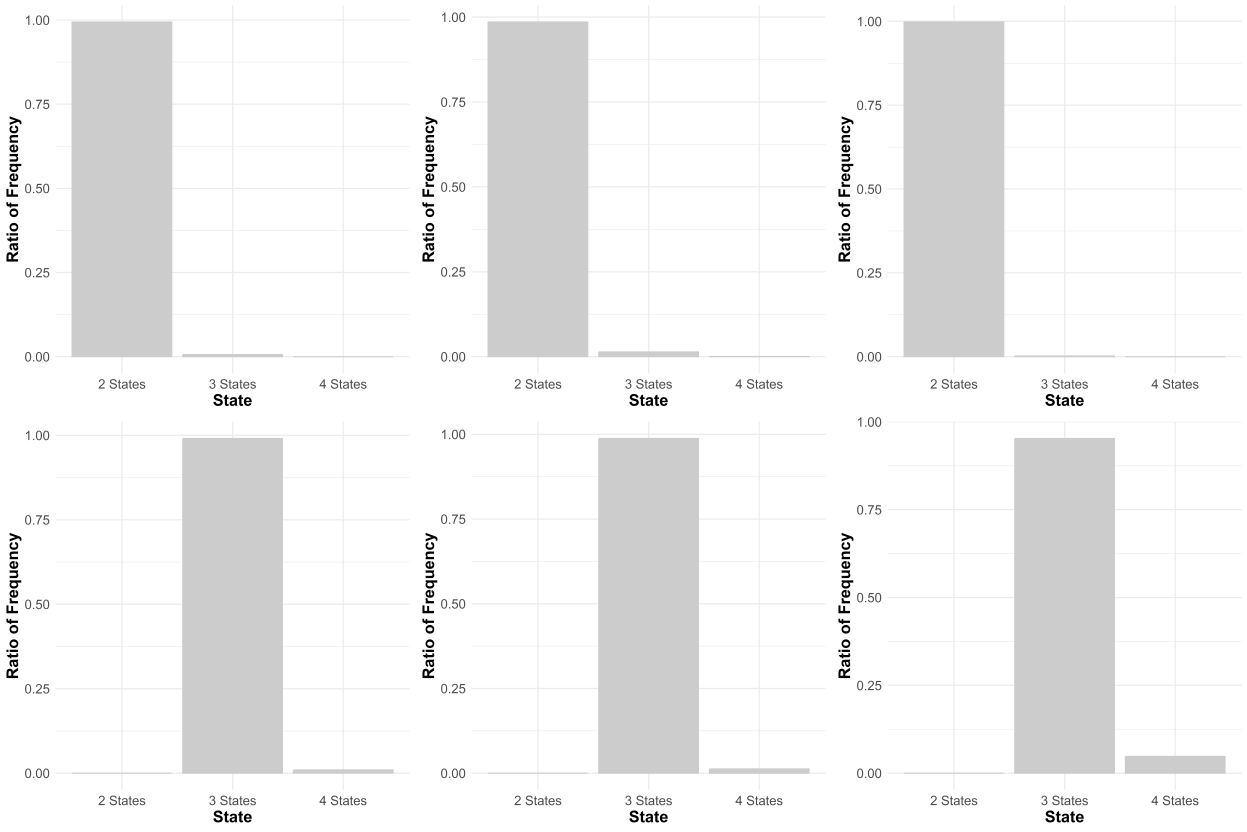

Figure 4.2: Estimated active states K for the copula-iHMM 2-state models (above) and 3-state models (below) using the last 2000 iterations.

Table 4.2: Simulation results of 3-state copula infinite hidden Markov models with samples $n = 2000$.

Params	Simulation 1 ($T = 2000$)			Simulation 2 ($T = 2000$)			Simulation 3 ($T = 2000$)		
	Mean 1	Credible interval 1	True value 1	Mean 2	Credible interval 2	True value 2	Mean 3	Credible interval 3	True value 3
$(R^1)_{12}$	0.719	$(0.68, 0.76)$	0.7	0.57	$(0.495, 0.635)$	0.6	0.779	$(0.748, 0.808)$	0.8
$(R^1)_{13}$	0.608	$(0.56, 0.65)$	0.6	0.53	$(0.446, 0.609)$	0.5	0.702	$(0.663, 0.735)$	0.7
$(R^1)_{23}$	0.515	$(0.46, 0.57)$	0.5	0.41	$(0.318, 0.490)$	0.4	0.614	$(0.559, 0.662)$	0.6
$(R^2)_{12}$	-0.673	$(-0.747, -0.584)$	-0.7	-0.58	$(-0.662, -0.471)$	-0.6	-0.775	$(-0.818, -0.723)$	-0.8
$(R^2)_{13}$	-0.616	$(-0.677, -0.543)$	-0.6	-0.48	$(-0.580, -0.375)$	-0.5	-0.624	$(-0.692, -0.559)$	-0.7
$(R^2)_{23}$	0.506	$(0.426, 0.580)$	0.5	0.33	$(0.330, 0.505)$	0.4	0.524	$(0.452, 0.594)$	0.6
$(R^3)_{12}$	-0.653	$(-0.794, -0.499)$	-0.7	-0.39	$(-0.597, -0.048)$	-0.6	-0.764	$(-0.852, -0.631)$	-0.8
$(R^3)_{13}$	0.65	$(0.519, 0.750)$	0.7	0.48	$(0.316, 615)$	0.6	0.799	$(0.721, 0.857)$	0.8
$(R^3)_{23}$	-0.665	$(-0.771, -0.529)$	-0.7	-0.4	$(-0.599, 0.002)$	-0.6	-0.722	$(-0.809, -0.550)$	-0.8
ν_1	3.931	$(3.161, 4.986)$	5	4.16	$(3.013, 5.920)$	4	5.68	$(4.288, 7.755)$	6
ν_2	3.87	$(2.879, 5.555)$	4	2.79	$(2.265, 3.353)$	3	3.656	$(2.676, 5.020)$	5
ν_3	2.863	$(2.080, 4.028)$	3	2.06	$(1.51, 2.89)$	3	2.751	$(2.015, 4.268)$	4
p_{11}	0.626	$(0.571, 0.680)$	0.6	0.59	$(0.487, 0.661)$	0.6	0.585	$(0.536, 0.632)$	0.6
p_{12}	0.292	$(0.248, 0.337)$	0.3	0.32	$(0.255, 0.388)$	0.3	0.318	$(0.278, 0.360)$	0.3
p_{13}	0.082	$(0.051, 0.117)$	0.1	0.10	$(0.057, 0.154)$	0.1	0.097	$(0.070, 0.127)$	0.1
p_{21}	0.498	$(0.426, 0.574)$	0.5	0.45	$(0.371, 0.532)$	0.5	0.482	$(0.421, 0.544)$	0.5
p_{22}	0.406	$(0.336, 0.470)$	0.4	0.45	$(0.381, 0.522)$	0.4	0.435	$(0.381, 0.488)$	0.4
p_{23}	0.096	$(0.060, 0.138)$	0.1	0.10	$(0.055, 0.149)$	0.1	0.082	$(0.048, 0.123)$	0.1
p_{31}	0.728	$(0.600, 0.850)$	0.7	0.65	$(0.491, 0.796)$	0.7	0.64	$(0.529, 0.749)$	0.7
p_{32}	0.059	$(0.000, 0.144)$	0.1	0.01	$(0, 0.076)$	0.1	0.116	$(0.043, 0.199)$	0.1
p_{33}	0.212	$(0.122, 0.312)$	0.2	0.33	$(0.201, 0.500)$	0.2	0.243	$(0.157, 0.331)$	0.2

estimations for every simulation.

For 3-state models, the performance of the sampler was maintained at similar levels. However, large errors were observed. For example, in Simulation 2 in Table 4.2, the true values of ν_3 and p_{32} are not covered by the credible region of the MCMC sampler. This is because State 3 has fewer samples than States 1 and 2, which makes the estimations worse. However, the estimation for States 1 and 2 in simulation 2 maintained a good level.

Overall, the estimation results are reasonably good, with a small bias with respect to the truth, and do not having significantly wide MCMC credible intervals for most parameters. This demonstrates the effectiveness of the proposed approach in estimating the number of mixture components and their corresponding parameters of interest when setting the copula-iHMM model.

4.5 Real data analysis

We employed our proposed copula-iHMM model to study the bilateral relations between the Shanghai Composite Index (SSECI) and the Hang Seng Index (HSI), the Standard & Poor's 500 Index (SPX), and the Financial Times Stock Exchange 100 index (FTSE), four important global financial indices. The daily closing prices of the four series from January 2018 to July 2023 were taken. We computed the corresponding log returns using $\log(r_t/r_{t-1})$.

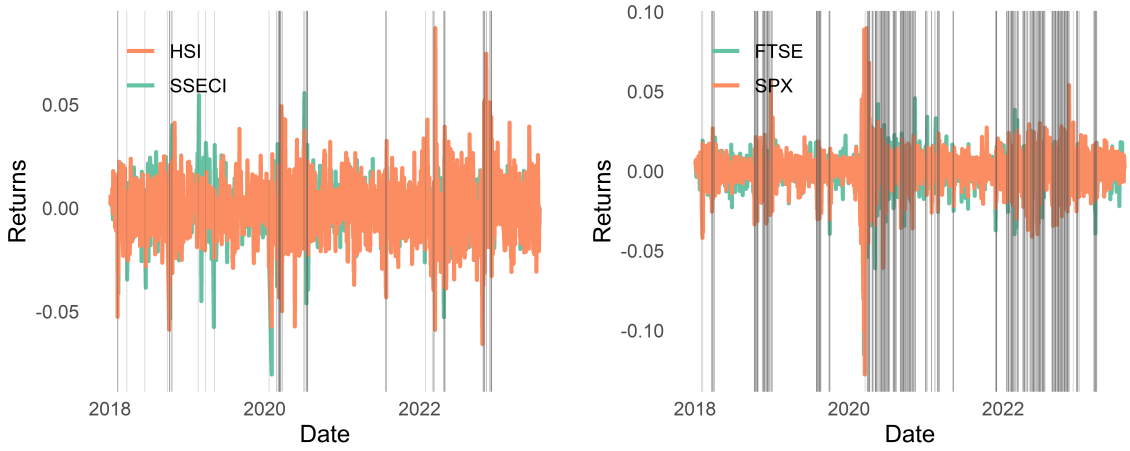

Figure 4.3: Returns of SSECI-HSI and FTSE-SPX from January 2018 to July 2023 using daily closed prices. The shaded regions are the times when the series are under the state $H_t = 2$

The returns from the common trading days between SSECI and HSI, SPX and FTSE were then paired respectively into a 2-dimensional time series.

Rather than using the classic approach in the copula literature by first employing the GARCH model to extract the changing volatility, and then proceeding with the copula estimation of the standardized data (Arakelian and Karlis, 2014), we directly applied our infinite hidden Markov model to the series. This approach is beneficial because the hidden Markov is directly applied to the time series. Meanwhile, the transition matrix can provide a more detailed dynamic structure of the series compared to ordinary finite mixture copulas $c = w_1c_1 + w_2c_2, \ldots, w_nc_n, w_i \geq 0, \sum w_i = 1$.

The parametric families we chose to fit the data were t-copulas along with normal margins, which were the same as the parametric models used in the simulation studies. All parameters of the margins and copulas were unknown, and we let our MCMC sampler determine their corresponding posterior distributions.

The MCMC algorithm was run for 10000 iterations, with the first 8000 iterations discarded as burn-in and the last 2000 iterations used as estimators of the posterior distributions, consistent with the simulation.

Table 4.3 presents our estimation results for the copula parameters and transition matrices. We omitted reporting the marginal distributions, as they are almost identical between states with approximately zero means.

Table 4.3: Estimation results of SSECI-HSI and SP500-FTSE100 daily returns from January 2018 to July 2023 using copula infinite hidden Markov models.

Estimated Parameters	SSECI-HSI		SPX-FTSE	
	Mean estimate	Credible interval	Mean estimate	Credible interval
$(R^1)_{12}$	0.61	$(0.567, 0.647)$	0.49	$(0.421, 0.558)$
$(R^2)_{12}$	0.68	$(0.566, 0.772)$	0.42	$(0.331, 0.507)$
$(R^3)_{12}$			0.60	$(0.400, 0.773)$
ν_1	11.5	$(8.00, 16.67)$	7.47	$(4.45, 11.76)$
ν_2	6.23	$(3.29, 9.73)$	6.74	$(4.17, 10.04)$
ν_3			2.93	$(2.05, 5.22)$
p_{11}	0.97	$(0.951, 0.985)$	0.97	$(0.953, 0.981)$
p_{12}	0.03	$(0.014, 0.049)$	0.03	$(0.018, 0.046)$
p_{13}			0.001	$(0, 0.003)$
p_{21}	0.69	$(0.480, 0.850)$	0.06	$(0.034, 0.100)$
p_{22}	0.31	$(0.144, 0.520)$	0.94	$(0.897, 0.965)$
p_{23}			0.002	$(0, 0.009)$
p_{31}			0.005	$(0, 0.052)$
p_{32}			0.04	$(0, 0.123)$
p_{33}			0.96	$(0.861, 0.997)$

For the SSECI-HSI estimation, the estimated number of states K has the probability

$$p(K = 2) = 0.9995,\ p(K = 3) = 0.0005.$$

This implies that we are strongly in favor of the two-state HMM. State 1 was estimated to be more persistent, with a 3% transition probability to another state. This corresponds to the usual market conditions. However, State 2 can easily return to State 1 with a probability of approximately 69%, and State 2 has a higher correlation with lower degrees of freedom. This could correspond to oscillating market conditions when the co-movement of market prices is considerably more extreme than usual. However, this was observed less often. We estimated the posterior mode of the state labels for each trading day; 5% of the days were determined to follow State 2, and the other days were estimated to follow State 1.

For results for SPX-FTSE, the number of states K has a posterior distribution

$$p(K = 2) = 0.9965,\ p(K = 3) = 0.0035.$$

This implies that we are also in favor of the two-state HMM. Among the trading days, 71.4% of the days are determined to follow State 1, 28.6% of the days have the posterior mode in State 2.

This differs from our observations in the patterns of SSECI-HSI. SPX-FTSE data seem to be more sticky, indicating that provided the series has entered a certain state, it is prone to remain in this state over time. Meanwhile, the parameters of the first two states appear to be close to each other. By comparing this case with the short-term market oscillation case in the SSECI-HSI pattern, State 2 on the SPX-FTSE data might indicate another type of market dependence.

Figure 4.3 shows the log returns for the series. The gray regions represent the times when the series enters a different state other than the most common state. The oscillating states of HSI-SSECI occasionally arise as time progresses. In contrast, the switching from State 1 to State 2 in the SPX-FTSE pattern was more persistent.

4.6 Concluding remarks

This chapter is an advancement of the previous chapter to consider the data with correlation. This would be useful when we want to directly apply our copula method to the time series data. For example, the time-correlated stock returns. Our proposed copula-iHMM model, constructed on the basis of the hierarchical Dirichlet process (HDP), would be able to automatically infer the number of hidden states. Therefore, it would be particularly suitable when applied to dynamically changing data since the tuning of the number of states is avoided each time. In financial risk management, the computation of value-at-risk and other risk metrics can benefit from this framework compared with the usual finite mixture copula approach (Hu, 2006). This is because HMM provides a transition matrix from state to state. This detailed transition structure is unavailable in the finite mixture model. One can also extend this copula-iHMM framework to other state-of-the-art copulas, such as vine copulas and factor copulas. This enriches the applicable scenarios of the copula-iHMM framework.

Chapter 5

Introduction to imbalance learning and the empricial ROC-AUC metrics

5.1 Introduction

Imbalanced learning problems are classification tasks in which the proportion of instances differs significantly among classes. Imbalanced datasets are common in many areas of applied sciences, such as medical image diagnostics (Mena and Gonzalez, 2006; Rezaei et al., 2020), financial fraud detection (Bhattacharyya et al., 2011), or cybersecurity (Cieslak et al., 2006).

Learning from an imbalanced dataset poses certain problems. Models tend to capture a lot more information from the *majority* class but often fail to identify the *minority* class. Therefore, when used for prediction, the ability to predict the minority classes can be weak.

Several methods exist to tackle this problem from different angles, including sampling techniques such as random oversampling, random undersampling, and the synthetic minority oversampling technique (SMOTE). Another remedy is cost-sensitive approaches, which are realized by giving different weights to different classes. For a more detailed discussion of the nature of the imbalanced problems and some relevant methodologies, readers are referred to (He and Garcia, 2009).

Measuring the performance of a classifier on imbalanced data requires a different approach from handling the class imbalance problem. The receiver operating characteristics (ROC) curve is commonly used to assess the capabilities of binary classifiers (Streiner and Cairney, 2007). It is a plot of false-positive versus true-positive rates in the space of $\mathbb{R}^2$. Statistical techniques are often required to compare two ROC curves or decide if a certain ROC curve

is above an ideal standard. Table 4.2 of (Pepe et al., 2003, p. 80) provides a useful list of the statistical indexes for describing the ROC curve, including the area under the curve (AUC), single ROC point (McNeil and Hanley, 1984), partial AUC (pAUC) (McClish, 1989), symmetry point, and Kolmogorov-Smirnov measure. Among those indexes, the area under ROC curves (ROC-AUC) is by far the most popular metric for the performance of a classifier in many scientific fields (He and Garcia, 2009).

To estimate the ROC-AUC from the data, the empirical AUC estimator is often used. This chapter will first introduce the empirical ROC-AUC metrics followed by studying the variances of this estimator when it is used in an imbalanced learning task. We first proved that under many common situations, variances of the empirical AUC increase with the imbalanced level of the dataset, and we further use simulations and real data analysis to demonstrate that, if classifiers are applied to an imbalanced dataset, there is a risk that the empirical AUC estimation will have high variances. Therefore, extra attention is required when using the empirical AUC to assess the models' performance on highly imbalanced data. The importance of the findings lies in their implication on the experimental design and model evaluations under the imbalanced setting. One must either carefully design their experimental methodologies to mitigate the issue mentioned or fully report variances of metrics when performing the evaluation procedures.

The structure of the chapter is as follows. The next section introduces notations used. This is followed by the main analytical results of the study in Section 5.3. We proceed to verify our findings using simulations in Section 5.4, and the real data analysis is performed in Section 5.6 to manifest the implications for learning tasks.

5.2 Classification

For the binary classification task with features, $X = (x_1, x_2, x_3, \ldots, x_p)$, and the class label $Y = \{0, 1\}$. The usual framework of evaluating a classifier's performance, followed by Krzanowski and Hand (2009), is to define a real-valued continuous score function $S(X) : \mathbb{R}^p \to \mathbb{R}$. The aim of defining such a function is to have sufficient separation between the positive (class 1) and negative (class 0) classes. Without loss of generality, we expect a high score for a positive instance and a low score for a negative instance.

An example of the score function is $S(x) = P(Y = 1 \mid X = x)$. This case presents the probability of an instance belonging to class 1 given its features. If X is regarded as a $\mathbb{R}^p$-valued random vector from the probability space (Ω, Σ, P), $S(X)$ becomes a real-valued

random variable. Hence, we can consider the score distributions of two classes; that is, for the absolute continuous $S(X)$, $(S(X) \mid Y = 1) \sim f$ and $(S(X) \mid Y = 0) \sim g$ are densities of the positive and negative population, and the corresponding distribution functions are denoted as F, G.

To make a classification decision, a threshold $t \in \mathbb{R}$ is needed. Instances with scores larger than t are classified as positive, and vice versa. Thus, given a threshold t, the false-positive rate (FPR) becomes $fp(t) = 1 - G(t)$, and the true-positive rate (TPR) becomes $tp(t) = 1 - F(t)$. We further let p_P and p_N represent the prior proportion of the positive and negative classes. Therefore, the accuracy of the prediction is $tp(t) \times p_P + (1 - fp(t)) \times p_N$ or simply $tp(t) \times p_P + (1 - fp(t)) \times (1 - p_p)$ since $p_N + p_p = 1$.

5.2.1 The area under a receiver operating characteristic curve

The ROC curve can be constructed from the scores and the threshold. If we vary the decision threshold t from $-\infty$ to $+\infty$ and plot the corresponding $(fp(t), tp(t))$, the ROC curve is obtained (Bradley, 1997). Assume we have the continuous densities of the positive and negative scores. Since $fp(t) = 1 - G(t)$ and $tp(t) = 1 - F(t)$, smiliar as (Krzanowski and Hand, 2009, p. 23), we arrive at

$$tp(t) = 1 - F(G^{-1}(1 - fp(t))) \tag{5.1}$$

For $fp(t) \in (0, 1)$. This fully characterizes the ROC curve using F and G, which must start from $(0, 0)$ and end with $(1, 1)$.

An important convention in the ROC analysis is to consider only the ROC curves that are fully above the chance diagonal $y = x$. This is because, if we obtained an ROC curve with some portions below the chance diagonal, we could flip the decision of our classifier in those portions to make the ROC curve above the chance diagonal. Another usual consideration is to study only cases in which the ROC curve is concave. If a continuous ROC curve has a non-concave portion, we can construct a new randomized classifier with better prediction power than the preadjusted one and obtain the concave ROC curve (Krzanowski and Hand, 2009, Section 8.3). The most straightforward approach to compare two ROC curves is to see if one ROC curve is over the other. The curve that lies above corresponds to the better classifier. However, this approach is not always applicable as, in most cases, the ROC curves cross each other. In this situation, the area under the ROC curve can be computed for comparison. If two classifiers $\mathcal{C}_1$ and $\mathcal{C}_2$ have $\text{AUC}_1 > \text{AUC}_2$, we usually consider $\mathcal{C}_1$ to have

better overall performance. Let S_P and S_N represent the random variables that follow the score distributions of positive and negative populations. Given Equation (5.1), the AUC can be interpreted in terms of the probability as follows:

$$\begin{aligned}
\text{AUC} &= \int_0^1 tp(fp)\ d(fp) \\
&= -\int_{-\infty}^{\infty} tp(t)\frac{d\big(fp(t)\big)}{dt}dt \\
&= \int_{-\infty}^{\infty} \big(1 - F(t)\big)g(t)dt \\
&= \int_{-\infty}^{\infty} \mathrm{P}\big(S_P > S_N \mid S_N = t\big)\mathrm{P}(S_N = t)dt \\
&= \mathrm{P}\big(S_P > S_N\big)
\end{aligned} \tag{5.2}$$

Therefore, the AUC can be understood as the probability of a randomly drawn positive sample's score being higher than a randomly drawn negative sample's score. A widely adopted approach of estimating the AUC from the testing samples is calculating the area under the empirical ROC curve, which is drawn from $tp(t) = 1 - \hat{F}(\hat{G}^{-1}(1 - fp(t)))$ where $\hat{F}$ and $\hat{G}$ are the corresponding empirical distributions. The empirical AUC estimator is equivalent to Mann-Whitney U statistics, as first pointed out by Bamber (1975). We have

$$\hat{A} = \frac{1}{n_N n_P}\sum_{i,j}[I(S_{P_j} > S_{N_i}) + \frac{1}{2}I(S_{P_j} = S_{N_i})], \tag{5.3}$$

where n_N, n_P refers to the numbers of positive and negative testing samples, S_{P_j}, S_{N_i} are the instances drawn from the corresponding random variables, and $I(\cdot)$ is the 0-1 indicator. For the continuous scores, we obtain the unbiasedness from

$$E(\hat{A}) = E(\frac{1}{n_N n_P}\sum_{i,j}[I(S_{P_j} > S_{N_i})]) = \mathrm{P}\big(S_P > S_N\big),$$

which coincides with (5.2). Furthermore, the variance of $\hat{A}$ is also important and requires scrutiny. The remainder of this chapter discusses this variance in the class imbalance setting.

Although the ROC-AUC has become a popular measure for assessing ROC curves, some scholars note that it has some deficiencies when used to compare performance between classifiers. For example, McNeil and Hanley (1984) find that two ROC curves might cross each other for a similar AUC with differing properties. Moreover, even if the curves are not

crossed, we may only be interested in certain regions of ROC and thus only need to summarize some portions of the area. They propose comparing the ROC using a single point of the ROC curve when the FPR or TPR is fixed. McNeil and Hanley (1984) emphasized that the ROC-AUC would give equal weight to every point of the FPR, but, in some areas of application, such as a clinic, only certain intervals of the FPR are of real concern. Therefore, they propose using partial AUC (pAUC) instead of the full AUC. pAUC is defined as $pAUC = \int_{c_1}^{c_2} y(x)dx$ where $[c_1, c_2]$ are the concerned interval of the FPR. Hand (2009) notes that the use of the AUC would automatically change the misclassification cost for different classifiers. Therefore, it is inconsistent, and they propose a new measure to evaluate the ROC: H-measure.

Despite the potential flaws of the AUC discussed above, it remains one of the most widely used summary statistics of the ROC curve. The following sections analyze the variance of the empirical AUC estimator (5.3).

5.3 Variance of the empirical AUC estimator

Variances of the empirical AUC estimator play an essential role in model evaluations and experimental design. Model evaluations are often concerned about whether a model's true AUC is above the required level or whether one classifier's AUC is significantly higher than another. This yields a hypothesis test requiring variances. In experimental design, variances influence determination of the sample size required to achieve specific statistical power in experiments. A related study is Hanley and McNeil (1982), which explores methods of calculating the sample size required to distinguish the AUC from continuous and discrete rating data. Obuchowski and McCLISH (1997) provide detailed methods of calculating different sample sizes for different indexes of ROC, which include AUC, pAUC, and single-point sensitivity. However, this formula is under the assumption of binormal scores. Further, Blume (2009) explores the bounds of maximum and minimum samples required for a specific test level with less assumption. Janes and Pepe (2006) comes up with an optimal positive-negative ratio to ensure the empirical AUC, pAUC estimator has minimal asymptotic variance.

Hanley and McNeil (1982) introduce a formula for estimating the variance of the empirical AUC estimator (5.3),

$$s^2(\hat{A}) = \frac{1}{n_N n_P}(A[1-A] + [n_p - 1][Q_1 - A^2] + [n_N - 1][Q_2 - A^2]), \tag{5.4}$$

where $\hat{A}$ and A are denoted as the empirical estimator and the true AUC of the classifier. We define Q_1 as the probability that the scores of two randomly selected positive samples are larger than that of another randomly selected negative sample, i.e.,

$$Q_1 = \mathrm{P}(S_{P_1}, S_{P_2} > S_N).$$

Meanwhile, Q_2 is defined similarly as the probability that the scores of two randomly selected negative samples are less than that of a randomly selected positive sample, that is,

$$Q_2 = \mathrm{P}(S_{N_1}, S_{N_2} < S_P).$$

As Q_1 and Q_2 are difficult to compute, they also suggest an approximation by the true underlying AUC

$$Q_1 = \frac{A}{2-A}, \quad Q_2 = \frac{2A^2}{(1+A)}. \tag{5.5}$$

The approximation of Q_1 and Q_2 becomes exact when the positive and negative scores follow the exponential distribution. The numerical experiments performed by Hanley and McNeil (1982) suggest that it is a good approximation in most cases.

Assuming the positive class to be the minority, let us denote x as the number of positive samples in the dataset and M as the total number of samples. We use $p = \frac{x}{M}$ to denote the proportion of the minority. Then,

$$v(x) = \frac{1}{x(M-x)}[A(1-A) + (x-1)(\frac{A}{2-A} - A^2) + (M-x-1)(\frac{2A^2}{(1+A)} - A^2)] \tag{5.6}$$

is a decreasing function with respect to x–that is, if we fix the sample size M, the true AUC A and use (5.5) as the approximation of Q_1 and Q_2. The approximated variance (5.6) of the empirical AUC will increase as the dataset becomes increasingly imbalanced. This can be summarized as the following Proposition:

Proposition 5 (Li (2020))**.** *Let M be the fix sample size, $x \in (1, M/2) \cap Z^+$ be the number of minority class, Q_1 and Q_2 be defined as (5.5), $0.5 \leq A < 1$. Then, $v(x)$ is a decreasing function with respect to x.*

Proof. See Appendix. □

Although Proposition 5 is a result under the special case, that is, we compute Q_1 and Q_2 under the circumstances of exponential scores, it gives us a good approximation of how

variances could change as the data become increasingly imbalanced.

If we work under the exact definition of Q_1 and Q_2, monotonicity still holds when we pose some restriction to the score distributions.

Proposition 6. *Let minority and majority score distributions have densities $S_P \sim f(x)$ and $S_N \sim g(x)$, which belong to a location family. That is , $f(x) = h(x - \mu_P)$ and $g(x) = h(x - \mu_N)$, and $h(x)$ is a symmetry density around 0. Then, (5.4) with $n_P = x, n_N = M - x$ is a decreasing function with respect to x. Otherwise, if there exists a monotone increasing function $T(x)$ such that $T(S_P) \sim h(x - \mu_P)$ and $T(S_N) \sim h(x - \mu_N)$, the result still holds.*

Proof. From the proof of Proposition 5, it suffices to prove that $Q_1 - Q_2 \leq 0$. Hence, we complete the proof by combining the result from Proposition 5.

As the following, let scores for the positive and negative populations be continuously distributed with CDF $F(x) = H(x - \mu_P)$ and $G(x) = H(x - \mu_N)$, respectively. We denote S_{P_1}, S_{P_2} to be a sample drawn independently from the score distribution of the positive population and S_N to be a sample drawn independently from the score distribution of the negative population. Then, we have

$$Q_1 = \mathrm{P}(S_{P_1}, S_{P_2} > S_N) = \mathrm{P}(T(S_{P_1}), T(S_{P_2}) > T(S_N)).$$

By independence,

$$\begin{aligned} Q_1 &= \int_R P(T(S_{P_1}) > x, T(S_{P_2}) > x \mid T(S_N) = x) P(T(S_N) = x) dx \\ &= \int_R P(T(S_{P_1}) > x \mid T(S_N) = x) P(T(S_{P_2}) > x \mid T(S_N) = x) P(T(S_N) = x) dx \\ &= \int_R P(T(S_{P_1}) > x) P(T(S_{P_2}) > x) P(T(S_N) = x) dx. \end{aligned}$$

Therefore,

$$\begin{aligned} Q_1 &= \int_R [1 - H(x - \mu_P)]^2 h(x - \mu_N) dx \\ &= \int_R [1 - H(z + \mu_N - \mu_P)]^2 h(z) dz \\ &= \int_R H^2(\mu_P - \mu_N - z) h(z) dz = \int_R H^2(\mu_P - \mu_N + z) h(z) dz, \end{aligned}$$

where we use the symmetry property $h(-z) = h(z)$ to change the variable at the last equality.

Similarly, let S_{N_1}, S_{N_2} be samples drawn independently from the negative population.

$$\begin{aligned} Q_2 = P(S_P > S_{N_1}, S_{N_2}) &= \int_R G(x)^2 f(x) dx \\ &= \int_R H(x - \mu_N)^2 h(x - \mu_P) dx = \int_R H^2(\mu_P - \mu_N + z) h(z) dz = Q_1. \end{aligned}$$

Therefore, $Q_1 - Q_2 = 0$. Hence, we complete the proof.

□

The Proposition 6 contains many commonly considered score distributions, such as the binormal scores with the same variance $S_P \sim N(\mu_P, \sigma^2)$, $S_N \sim N(\mu_N, \sigma^2)$ or the scores that can be transformed into the binormal scores.

Suppose we release the condition of Proposition 6. In that case, the monotonicity of variances with respect to the proportion of the minority p is not always guaranteed. As shown later, there might exist an optimal $p < 0.5$ such that the minimum variance is achieved for a fixed M. However, in a practical sense, this ratio is unlikely to approach near 0. Hence, under the problems of high class imbalance, the volatility of the estimation is still of great concern.

To explore the optimal proportion p, we write (5.4) as

$$s^2(\hat{A}) = \frac{1}{M^2 p(1-p)} (A[1 - A] + [Mp - 1][Q_1 - A^2] + [M(1-p) - 1][Q_2 - A^2]) \qquad (5.7)$$

By denoting $Q_1 - A^2 = \alpha^*, Q_2 - A^2 = \beta^*, A(1 - A) = \gamma^*$,

$$(5.7) = \frac{1}{M} \left(\frac{1}{Mp(1-p)} [\gamma^* - \alpha^* - \beta^*] + \frac{\alpha^*}{1-p} + \frac{\beta^*}{p} \right).$$

For a fixed M, we seek the stationary point by differentiating with respect to p and equating it to 0,

$$(\alpha^* - \beta^*)p^2 + 2(\beta^* + \frac{\gamma^* - \alpha^* - \beta^*}{M})p - (\beta^* + \frac{\gamma^* - \alpha^* - \beta^*}{M}) = 0. \tag{5.8}$$

As $\alpha^* - \beta^* = Q_1 - Q_2$, we have proofed the monotonicity of s^2 with respect to $p \in (0, 0.5)$ previously when $Q_1 - Q_2 \leq 0$, what left to be discussed is the case when $\alpha^* - \beta^* = Q_1 - Q_2 > 0$. Now, since

$$A = P(S_P > S_N) = \int_R (1 - F(y))g(y)dy = \int_R G(x)f(x)dx \tag{5.9}$$

By Jensen's inequality,

$$\begin{aligned} \alpha^* &= Q_1 - A^2 = \int_R (1 - F(y))^2 g(y)dy - (\int_R (1 - F(y))g(y)dy)^2 \geq 0 \\ \beta^* &= Q_2 - A^2 = \int_R (G(x))^2 f(x)dx - (\int_R G(x)f(x)dx)^2 \geq 0. \end{aligned} \tag{5.10}$$

Practically speaking, $\beta^* + \frac{\gamma^* - \alpha^* - \beta^*}{M} \approx \beta^*$ because $\gamma^*, \alpha^*, \beta^*$ are at the same scale, which is often much smaller than the scale of M. So, we can solve (5.8) by neglecting $\frac{\gamma^* - \alpha^* - \beta^*}{M}$.

$$p_{optim} \approx \frac{\sqrt{\beta^*}}{\sqrt{\alpha^*} + \sqrt{\beta^*}}. \tag{5.11}$$

The result (5.11) becomes exact asymptotically, but a large dataset is not required for this approximation to be sufficiently accurate.

By noticing from (5.10) that

$$\begin{aligned} \alpha^* &= E_{p(y)\sim g}(1 - F(Y))^2 - \left(E_{p(y)\sim g}(1 - F(Y))\right)^2 = Var(1 - F(S_N)) \\ \beta^* &= E_{p(x)\sim f}(G(X))^2 - \left(E_{p(x)\sim f}(G(X))\right)^2 = Var(G(S_P)). \end{aligned} \tag{5.12}$$

The expression (5.11), therefore, coincides with (Janes and Pepe, 2006, eq. 6.1). However, their derivation works from the asymptotic variance of $\hat{A}$ proposed by DeLong et al. (1988),

$$s^2(\hat{A}) = \frac{Var(G(S_P))}{n_P} + \frac{Var(1 - F(S_N))}{n_N}.$$

Our derivation, however, is based on the finite sample variances, which enables the error analysis of (5.11) when the dataset is small. Under the previously mentioned location family

case, as $Q_1 = Q_2$, we have $p_{optim} = \frac{1}{2}$.

5.4 Numerical simulation

We have shown in the last section that imbalanced levels of datasets could influence the empirical AUC estimator's variances. Additionally, the shapes of score distributions could also affect the estimator's accuracy. Therefore, in this section, we perform simulations by assuming the analytical distributions of scores. Monte Carlo simulations are performed to calculate sample statistics. This is done by first sampling the scores from the assumed distributions with the batch size $n = 200, 500$ and the replication is done for 500 times. The asymptotic theory guarantees their convergence to the actual value. Therefore, the Monte Carlo results are considered benchmarks when we assess the accuracy of approximations (5.6). In this section, we aim to determine how the optimal point (5.11) changes while the shapes of the scores change. We want to investigate the magnitude of variances in extremely imbalanced cases to see whether it is significant and examine the accuracy of the exponential approximation (5.6).

5.4.1 Normal score distribution

We begin by assuming that the score follows a normal distribution, a popular choice in the literature (Zou et al., 1997; Zou and Hall, 2000; Qin and Hotilovac, 2008). Assume that positive and negative scores are independently distributed with $S_P \sim \mathcal{N}(\mu_P, \sigma_P^2)$ and $S_N \sim \mathcal{N}(\mu_N, \sigma_N^2)$. We focus on discussing cases in which the positive class is the minority. Therefore,

$$\text{AUC} = \text{P}(S_p - S_N > 0) = \Phi\left(\frac{\mu_P - \mu_N}{\sqrt{\sigma_P^2 + \sigma_N^2}}\right)$$

where Φ represents the standard normal CDF and

$$Q_1 = \int_R (1 - \Phi(\frac{x - \mu_P}{\sigma_P}))^2 \phi_{(\mu_N, \sigma_N)}(x) dx$$
$$Q_2 = \int_R \Phi(\frac{x - \mu_N}{\sigma_N})^2 \phi_{(\mu_P, \sigma_P)}(x) dx$$

where $\phi_{(\mu,\sigma)}(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp \frac{(x-\mu)^2}{2\sigma^2}$.

The positive and negative scores share the same variance. That is, $S_P \sim \mathcal{N}(\mu_P, \sigma^2)$ and $S_N \sim \mathcal{N}(\mu_N, \sigma^2)$. Proposition 6 has proved that the $p_{optim} = 0.5$. Therefore, variances of the

empirical AUC increased with the decreasing proportion of minorities. To explore the optimal ratio (5.11), we fix the distribution of the majority as the standard normal, $S_N \sim \mathcal{N}(0,1)$, while, for the minority distribution, we set $S_P \sim \mathcal{N}(\mu_P, \sigma_P^2)$ where $\sigma_P \in (0.1, 10)$ and μ_p is adjusted according to

$$\mu_P = \sqrt{\sigma_P^2 + \sigma_N^2}\Phi^{-1}(\text{AUC}) + \mu_N \tag{5.13}$$

Therefore, we can investigate it under a certain level of actual AUC.

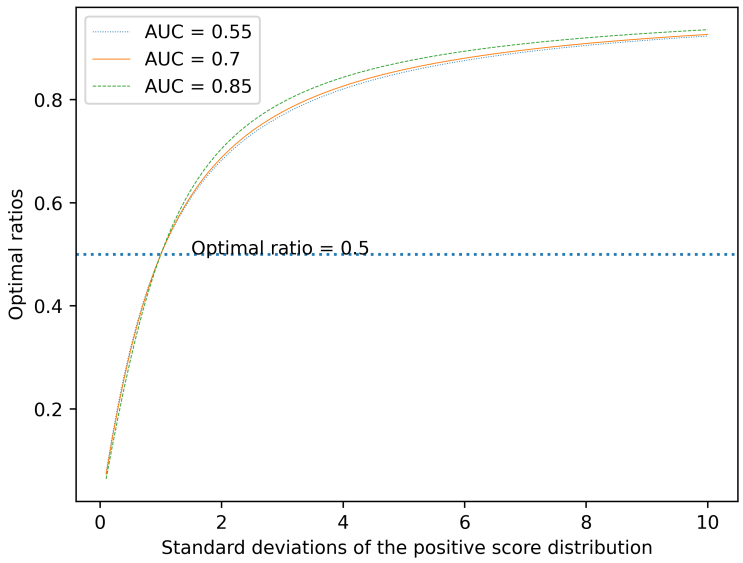

Figure 5.1: Optimal ratios p_{optim} for various standard deviations and AUCs

Fig 5.1 plots the optimal proportions of the positive class under the changing σ_p and three different AUC levels. As we are discussing the minority population, values above the horizontal line become meaningless. Notably, the actual AUC merely influences the optimal ratio under the binormal scores. The ratio is monotonically increasing with respect to standard deviations of the positive population, which is intuitively reasonable as we need more points to estimate when scores are much more uncertain. Therefore, we expect the outcome of Proposition 5 and 6 also holds for binormal scores with $\sigma_P > \sigma_N$.

Following the discussion above, we perform our simulations in three cases: $\sigma_P = \sigma_N = 1$, $\frac{\sigma_P}{2} = \sigma_N = 1$, and $2\sigma_P = \sigma_N = 1$. The negative scores are set to be the standard normal while the mean of the positive score is properly adjusted according to (5.13). In the first two cases, the empirical AUC achieves its lowest variances when the minority has the proportion $p_{optim} = 1/2$ while the last one has $p_{optim} < 1/2$.

Figure 5.2 plots the positive and negative binormal densities with $\sigma_P = \sigma_N = 1, \mu_N = 0$

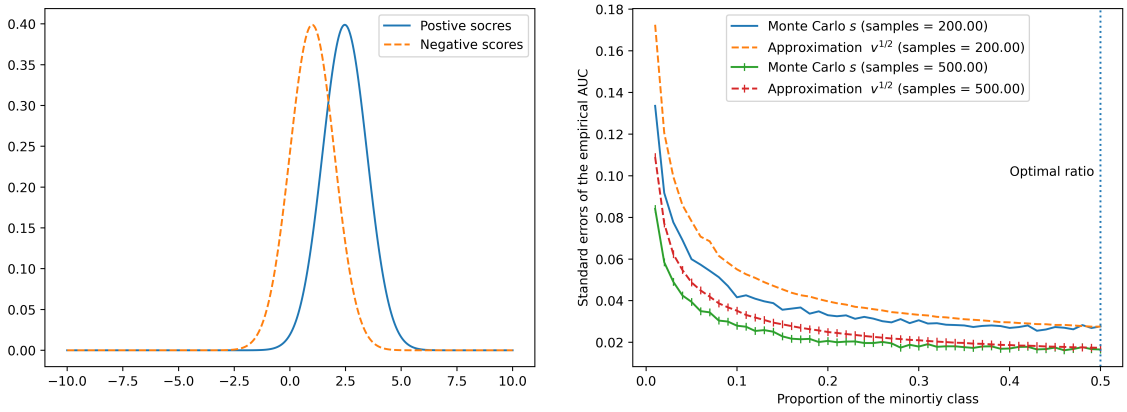

Figure 5.2: The positive and negative binormal scores with $\sigma_P = \sigma_N = 1, \mu_N = 0$, and AUC = 0.85 (left). The corresponding standard errors plot with sample size $200, 500$ by Monte Carlo and (5.6) .

and AUC= 0.85. The Monte Carlo sample variances decrease with the increasing proportion of the minority. The exponential approximation using the sample mean of $\hat{A}$ also shows its monotonicity, as we proved in the Proposition 5, and it is a conservative but close approximation to the truth. As shown in the plot, variances reach a very high level when the dataset is extremely imbalanced. To give a numerical sense of the fluctuation, we use the asymptotic interval of the empirical AUC following from (Krzanowski and Hand, 2009, Section 4.3.2),

$$\big(A - s(\hat{A})\Phi^{-1}(1 - \frac{\alpha_c}{2}), A + s(\hat{A})\Phi^{-1}(1 - \frac{\alpha_c}{2})\big), \tag{5.14}$$

where α_c is the type I error and A is the true AUC. If A is 0.85, with the sample size equaling 500, the 95% coverage of $\hat{A}$ when the proportion of the minority equal to $p = 0.05, 0.1$ is approximately $(0.77, 0.92)$ and $(0.80, 0.90)$, and the deviation is significant and beyond any negligible levels.

Fig 5.3 plots the densities and variances with $0.5\sigma_P = \sigma_N = 1$ or $\sigma_P = 2\sigma_N = 1$, respectively. The actual AUC is set to be 0.85. When $\sigma_P = 2\sigma_N$, standard errors of the empirical AUC become even larger compared with Fig 5.2. The proxy (5.6) is slightly biased downwards but still gives a reasonably close estimation. Further, the latter plot with $\sigma_P = 0.5\sigma_N$ has the minimum variance at $p < 0.5$. In this case, the proxy (5.6) gives a very conservative value, especially when the dataset is imbalanced. Nevertheless, it correctly displays the trend of standard errors when the dataset becomes highly imbalanced. That is, variances of the empirical AUC increase quickly when p approaches 0. When $\sigma_P = 2$

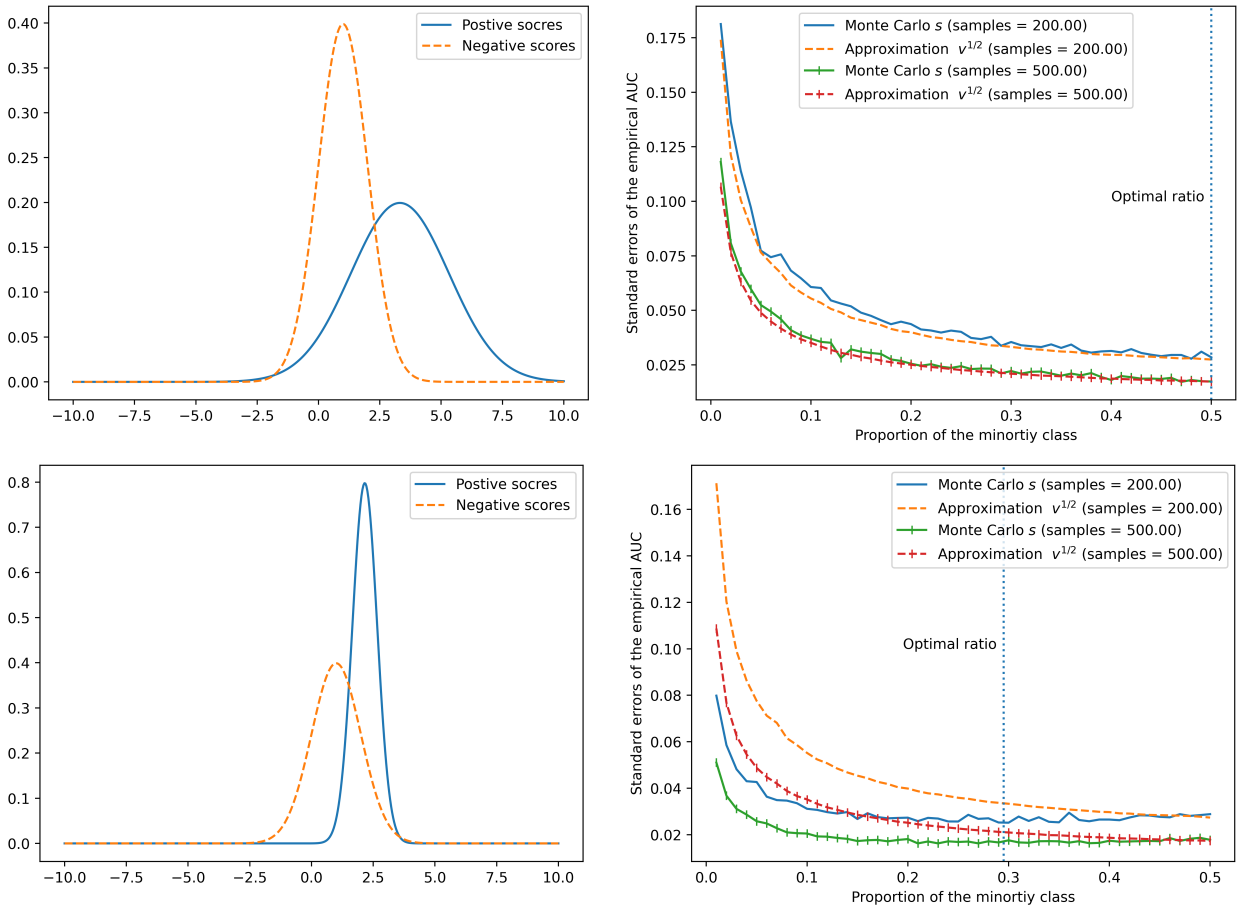

Figure 5.3: Plots of densities and standard errors with $0.5\sigma_P = \sigma_N = 1$ (above) and $2\sigma_P = \sigma_N = 1$ (below), $\mu_P = 0$ and AUC= 0.85.

and AUC 0.85, the 95% coverage of $\hat{A}$ with $p = 0.1$ and $n_P + n_N = 500$ is approximately $(0.79, 0.92)$. However, for the $\sigma_P = 0.5$ case, the interval is $(0.81, 0.89)$.

As the plots show, variances of the empirical AUC surge to a high level when $p \to 0$ in all three cases. Despite when $\sigma_P = 1/2$, $s^2(\hat{A})$ having a local minimum on $p < 1/2$, its behavior on the extremely imbalanced data remains unchanged. Moreover, standard errors of the empirical AUC would be influenced by the uncertainty of scores; higher uncertainty in the minority classes can lead to the higher standard errors of $\hat{A}$.

5.4.2 Probabilistic score distributions

In this section, we perform simulations based on the bi-beta score distributions, which restrict the range of scores to $[0, 1]$. This feature agrees with the property of probabilistic classifiers. Some popular such classifiers include logistic regression (Kleinbaum et al., 2002), naive Bayes (Zhang et al., 2009; Ren et al., 2009), and neural networks with specific structures Ruby and Yendapalli (2020).

In the following, we assume score distributions of the positive and negative populations to be $F \sim Beta(\alpha, \beta)$ and $G \sim Beta(\beta, \gamma)$ with $\gamma = 5$. The actual AUC and Q_1 , Q_2 can be calculated numerically from (5.9) and (5.10). In Fig 5.4, we vary the parameter α with different fixed β and depict the corresponding dynamics of p_{optim} (5.11), the relative variances of scores σ_P^2/σ_N^2, and the changing AUCs. Compared with what we did in the binormal simulations, controlling the AUC of bi-beta scores is demanding when we change the parameters. However, the values of optimal proportion appear to be predominantly controlled by the relative variances between the positive and the negative regardless of the AUCs. Lower relative variances of the positive correspond to lower optimal proportions of the minority with the relative variance equal to 1 being the threshold. This finding is consistent with what we observe in Fig. 5.1. Therefore, as in the previous section, we also conduct three simulations with $\sigma_P = \sigma_N$, $\sigma_P < \sigma_N$, and $\sigma_P > \sigma_N$.

Fig 5.4 plots the empirical AUC variances using the Monte Carlo and approximations (5.6) with $S_N \sim Beta(3, 5)$ and $S_P \sim Beta(5, 3), Beta(8, 3), Beta(3, 3)$. This corresponds to $\sigma_P = \sigma_N$, $\sigma_P \approx 0.8\sigma_N$, and $\sigma_P \approx 1.2\sigma_N$, respectively. The pattern we observe closely follows Fig. 5.2 and Fig. 5.3. High relative variances of positive scores can lead to high standard errors of empirical AUCs. When $\sigma_P < \sigma_N$, the optimal proportion of the minority class $p_{optim} < 0.5$. The performances of the proxy (5.6) are sensible for the top and bottom cases. When $\sigma_P < \sigma_N$, it gives the correct trend but might be too conservative, as discussed in the binormal cases.

For the top, middle, and bottom plots in Fig 5.5, the standard errors of the empirical AUC estimator and the true AUCs when $p = 0.05, 0.1$ with the sample size being 500 are $s_{top} = 0.037, 0.027, \text{AUC}_{top} = 0.86$, $s_{middle} = 0.020, 0.014, \text{AUC}_{middle} = 0.95$, $s_{bottom} = 0.059, 0.042, \text{AUC}_{bottom} = 0.69$. Even for the least variant middle case, the 95% interval when $p = 0.05$ is as wide as $(0.91, 0.99)$. Hence, enough attention must be paid to the variances of empirical AUC when the data tested are imbalanced.

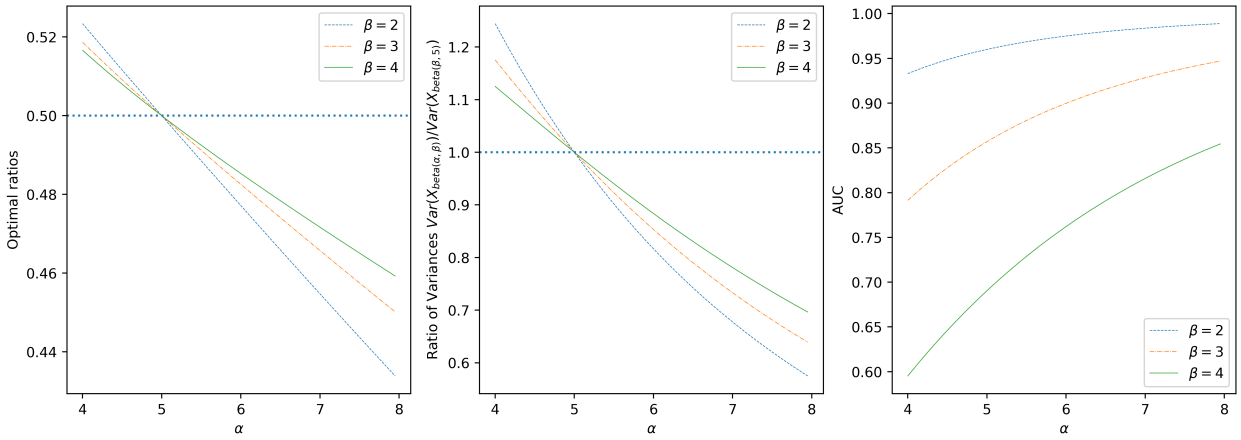

Figure 5.4: The plot of α with respect to optimal proportions of minority (left), the plot of α with respect to relative variances (middle), and the plot of α vs. AUC (right).

5.5 Computing sample sizes required

This section is devoted to computing the sample size required to correctly reject the wrong null hypothesis $H_0 : A \leq \theta_0$ in contrast to the alternative $H_1 : A > \theta_0$ when we use the empirical AUC estimation (5.3).

Several studies have discussed the problems of sample size calculation with regard to ROC curves and AUC. Some seminal works include (Hanley and McNeil, 1983), who proposed the method of comparing two correlated AUCs. However, their approach to computing the correlation is based on the assumption of normality. DeLong et al. (1988) further improved the method by removing the normality assumption.

We test whether a classifier has a true AUC value A larger than θ_0, which typically involves a statistical test with

$$H_0 : A \leq \theta_0 \quad H_1 : A > \theta_0 \tag{5.15}$$

for a fix $\theta_0 = \alpha$. As suggested by Krzanowski and Hand (2009, Chapter 4), we use the statistic

$$T = \frac{\hat{A} - \theta_0}{s_{\theta_0}(\hat{A})} \tag{5.16}$$

where $\hat{A}$ is an estimation of A and s_{θ_0} is the variance of the $\hat{A}$ when $A = \theta_0$. We estimate the variance s_{θ_0} using Equation (5.4). We aim to reject the null hypothesis H_0 when our

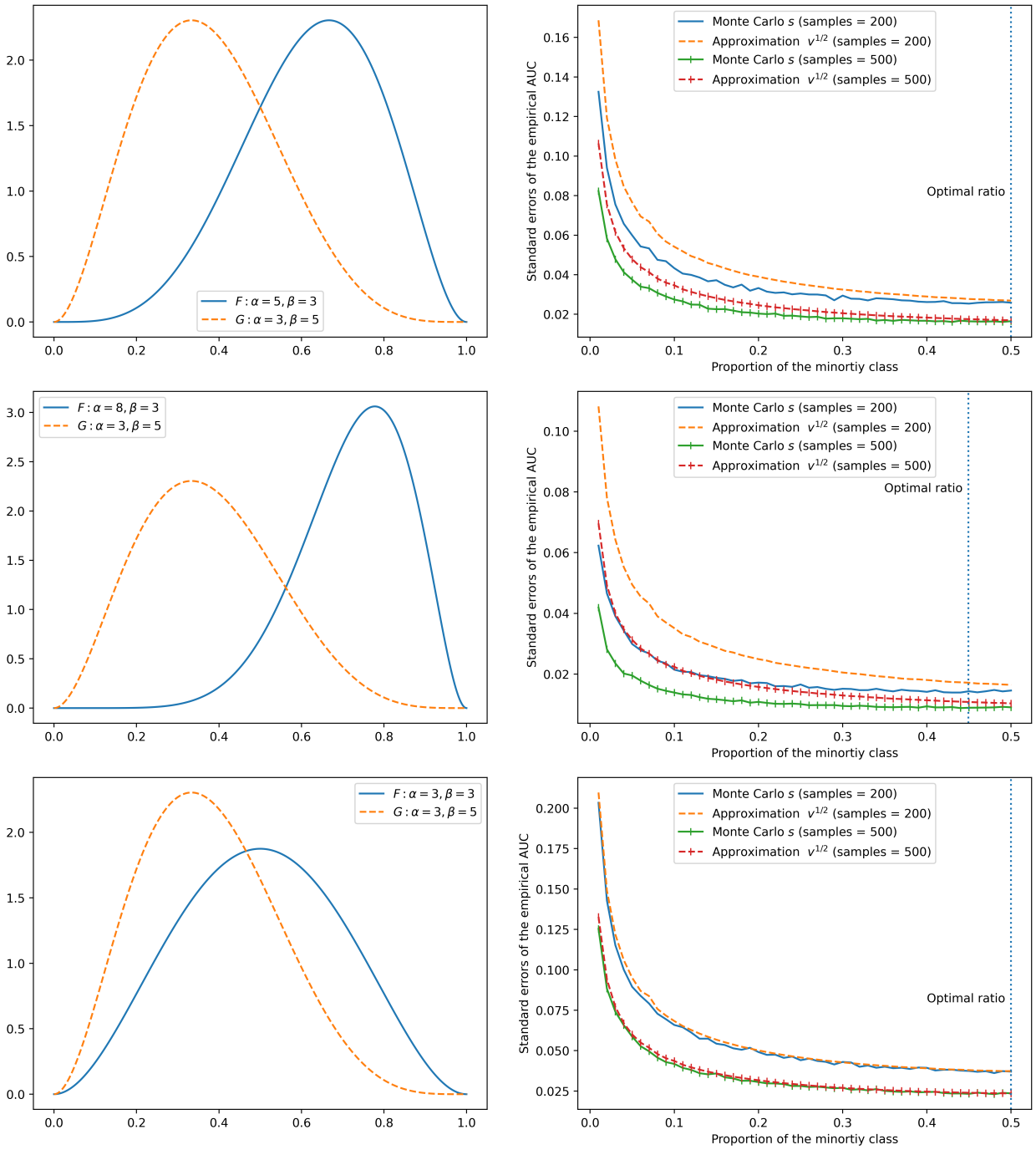

Figure 5.5: Plots of empirical AUC variances with $\sigma_P = \sigma_N$ (top), $\sigma_P < \sigma_N$ (middle), and $\sigma_P > \sigma_N$ from bi-Beta distribution with $G \sim Beta(3, 5)$

true AUC, A, deviates from θ_0. That is, by considering the imbalance levels, we want to compute the sample size needed to achieve satisfactory statistical power.

As discussed, the empirical AUC estimator has a high variance when it is applied to highly imbalanced datasets. Therefore, we aim to examine the extent to which different imbalance levels could impact the required sample sizes when the type I and II errors are given. More specifically, if our true AUC is θ_1, which is different from θ_0, we want to perform the testing (5.15) with the confidence level $1-\alpha$ and statistical power β. That is,

$$\mathrm{P}\left(\frac{\hat{A}-\theta_0}{s_{\theta_0}} > \Phi^{-1}(1-\alpha)\right) \geq \beta,$$

that is equivalent to

$$\mathrm{P}\left(\frac{\hat{A}-\theta_1}{s_{\theta_1}} > \frac{\theta_0-\theta_1+\Phi^{-1}(1-\alpha)s_{\theta_0}}{s_{\theta_1}}\right) \geq \beta. \tag{5.17}$$

If θ_1 is the true AUC, $\frac{\hat{A}-\theta_1}{s_{\theta_1}}$ is asymptotically normal. Therefore, if our imbalance level is p, to calculate the minimum sample size required for a given power β , we have to solve the following equation:

$$\frac{\theta_0-\theta_1+\Phi^{-1}(1-\alpha)s_{\theta_0}}{s_{\theta_1}} = -\Phi^{-1}(\beta).$$

Rearranging the equation,

$$\phi^{-1}(1-\alpha)s_{\theta_0} + \phi^{-1}(\beta)s_{\theta_1} - (\theta_1-\theta_0) = 0. \tag{5.18}$$

As shown in Section 5.4, the approximation $Q_1 = \frac{A}{2-A}$ and $Q_2 = \frac{2A^2}{1+A}$ gives either conservation or close results of variances. Therefore, it is a suitable approximation for the sample size calculation. We proceed to use (5.6) to compute the standard error, that is,

$$s_\theta = \sqrt{\frac{1}{n_N n_P}\left(\theta[1-\theta] + [n_p-1]\left[\frac{\theta}{2-\theta}-\theta^2\right] + [n_N-1]\left[\frac{2\theta^2}{1+\theta}-\theta^2\right]\right)}. \tag{5.19}$$

Substituting (5.19) into (5.18), we can solve Equation (5.18) numerically (e.g., the Newton-Raphson method). The following Table 5.1 lists the minimum samples required to achieve 90% and 80% power for a 95% confidence level hypothesis test with the null hypothesis being $H_0 : \theta \leq 0.8$, $H_1 : \theta > 0.8$. The true AUC θ_1 is supposed to be 0.85, 0.9, 0.95 under

the dataset with several proportions of the minority $p = 0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5$. We visualize the statistical power $\beta = 80\%$ case in Fig.5.6.

Table 5.1: Minimum samples required to achieve 90% power (left) and 80% power (right) for the 95% confidence level hypothesis test $H_0 : \theta \leq 0.8$, $H_1 : \theta > 0.8$. The true $\theta = 0.85, 0.9, 0.95$, $p = 0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5$).

p	$\theta_1 = 0.85$	$\theta_2 = 0.9$	$\theta_3 = 0.95$
0.01	22560	4962	1790
0.05	4566	1005	363
0.1	2328	512	185
0.2	1217	267	97
0.3	857	188	68
0.4	688	150	54
0.5	601	130	47

p	$\theta_1 = 0.85$	$\theta_2 = 0.9$	$\theta_3 = 0.95$
0.01	16562	3763	1433
0.05	3362	764	291
0.1	1715	289	149
0.2	898	204	78
0.3	632	143	55
0.4	508	115	44
0.5	446	100	39

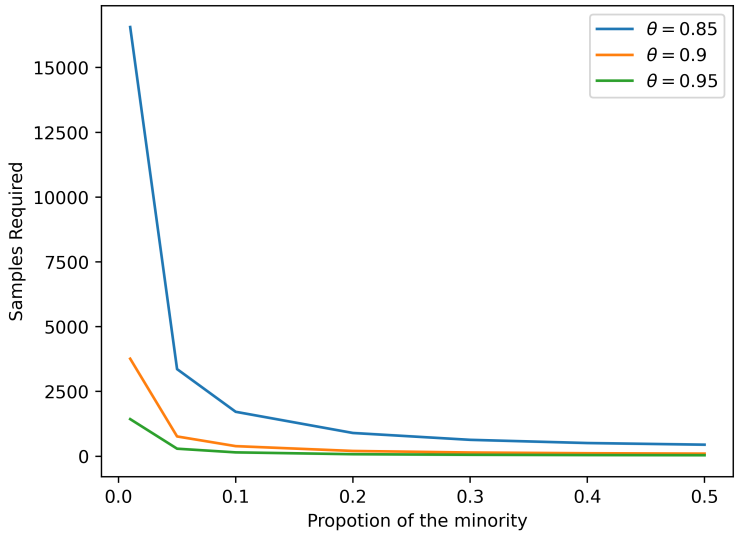

Figure 5.6: Plot of samples required to achieve 80% power for 95% one side hypothesis test

We can conclude from Table 5.1 and Fig 5.6 that samples required to maintain a reasonable statistical power become extremely large when the dataset is highly imbalanced, especially when the true AUC is close to the tested level θ_0. Influences of imbalanced levels must be considered when conducting such experiments.

5.6 Real dataset experiments

5.6.1 Student performance prediction

To illustrate how an imbalanced dataset might affect the estimation of empirical ROC-AUCs in real experiments, we first take a dataset from the Kaggle website named *on-time graduation classification* [1]. The dataset collects the grade point average (GPA) of each student for four semesters and records if they graduate on time (see Table 5.2). Specifically, each GPA column is real-valued with the range of $[0, 4]$ up to two decimal points. The label column *If Graduate* is Boolean to indicate whether a specific student had passed.

Column Name	GPA s1	GPA s2	GPA s3	GPA s4	If Graduate
Data Type	Float	Float	Float	Float	Boolean

Table 5.2: On-time graduation dataset structure

The dataset is highly imbalanced because only a few students did not graduate on time. To be specific, 92% of students in this dataset completed their degree, and 8% did not graduate on time. The total number of data points is 1687. We perform stratified tenfold cross-validation to compute the mean empirical AUC and the sample standard deviation. The variance approximation formula (5.6) is also used for comparison.

Table 5.3: Mean AUCs and corresponding standard deviations of the graduation dataset

Classifiers	Mean AUC	Standard errors	
		Tenfold	Approximation (5.6)
Logistic Regression (LR)	0.512	0.025	0.084
K-nearest Neighbors (KNN)	0.511	0.021	0.084
Naive Bayes (NB)	0.582	0.068	0.086
Random Forest (RF)	0.515	0.030	0.084
Adaptive Boosting (Adaboost)	0.522	0.031	0.085
Extreme Gradient Boosting (XGBoost)	0.601	0.095	0.086
SMOTE-LR	0.629	0.068	0.086
SMOTE-KNN	0.610	0.066	0.086
SMOTE-NB	0.621	0.071	0.086
SMOTE-RF	0.589	0.071	0.086
SMOTE-AdaBoost	0.616	0.062	0.086
SMOTE-XgBoost	0.634	0.083	0.086

Table 5.3 reports the results of our experiments. We use six popular classifiers–logistic

[1] https://www.kaggle.com/oddyvirgantara/on-time-graduation-classification

regression, K-nearest neighbors, naive Bayes, random forest, adaboost, and xgboost–together with their corresponding SMOTE versions to deal with the imbalanced dataset. For the tenfold validation, every test set has 168 samples with approximately 8% being the minority samples. As the table shows, variances of the estimation on test sets are not negligible. The standard errors even increase with the sample mean, which would discredit their mean values statistically. Specifically, we consider the confidence interval $(\text{mean} - 1.645s/\sqrt{10}, \text{mean} + 1.645s/\sqrt{10})$, where s is the tenfold standard errors. Fig 5.7 gives us the outcome for the RF, KNN, and LR models. Their confidence intervals reach or almost reach the threshold $x = 0.5$, meaning they might not be statistically better than a random guess.

Notably, the sample means $\hat{A}$ of the xgboost, SMOTE-xgboost, SMOTE-adaboost, SMOTE-NB, SMOTE-KNN, and SMOTE-LR models are all higher than 0.6. However, their high sample variances indicate that they could not be confirmed to have AUCs higher than 0.6 statistically at a 95% confidence level. Therefore, in a highly imbalanced dataset, the standard errors must be considered along with sample means of empirical AUCs to reach any correct conclusions.

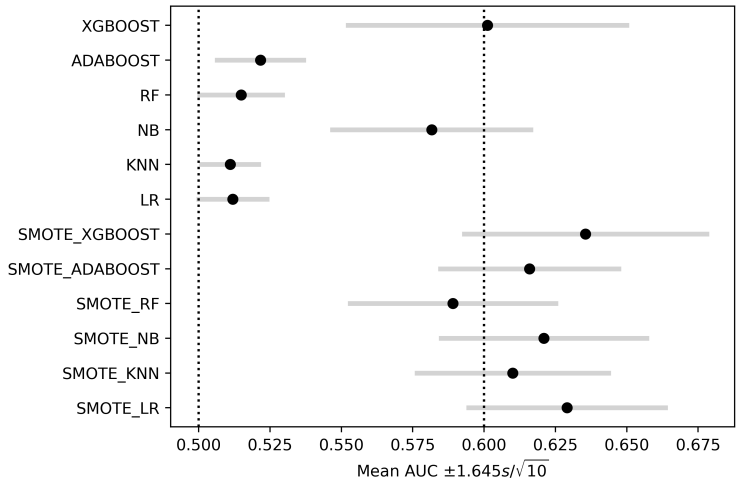

Figure 5.7: Mean AUCs of tenfold validations and the corresponding $\pm 1.645s/\sqrt{10}$ interval using the graduation data

5.6.2 Wine quality prediction

We offer two other examples of binary classification tasks using the *red wine quality* dataset [2] originally sourced from Cortez et al. (2009). The dataset contains 11 real-valued features

[2]https://archive.ics.uci.edu/ml/datasets/wine+quality

of red wine and an ordinal label column with a scale of $0-10$, which indicates the quality of the wine. The typical samples are displayed in Table.5.4.

Features	sample 1	sample 2	sample 3
Fix acidity	7.4	7.8	11.2
Volatile acidity	0.7	0.88	0.28
Citric acidity	0	0	0.56
Residual sugar	1.9	2.6	1.9
Chlorides	1.9	2.6	1.9
Free.sulfur.dioxide	11	25	17
Total.sulfur.dioxide	34	67	60
Density	0.9978	0.9968	0.9980
PH	3.5	3.2	3.16
Sulphates	0.56	0.68	0.58
Alcohol	9.4	9.8	9.8
Quality	5	5	6

Table 5.4: Typical samples of the red-wine dataset

The total number of samples in the dataset is $n = 1599$. In the first task, we classify the red wine with a quality lower than 5 to be a bad wine and that with quality higher than 5 to be not bad. For the second task, we classify great wine as wine with a quality greater than or equal to 7; otherwise, it is not great. Both tasks are highly imbalanced. For the first task, 63 samples are bad with the proportion of the minority $p = 0.039$. For the second task, 217 samples are great. Our imbalanced level is $p = 0.136$. We use stratified tenfold cross-validation with the standard z normalization applied to both assignments. We summarize the results using the same classifiers as in the previous section and compute corresponding mean AUCs and standard errors. The outcome is listed in Table.5.5, and the corresponding confidence interval is plotted in Fig 5.8.

Every test set of task 1 has 159 samples with $p = 0.039$, and every test set of task 2 has 159 samples with $p = 0.136$. The table and figure reveal that the estimation is overall more volatile for task 1 than for task 2. This observation is consistent with our expectation that working with more imbalanced datasets could result in more uncertainty of the empirical AUC estimation. In the figure, the classifier with the highest mean AUC in the task 1 is the xgboost model with a wide confidence interval $(0.67, 0.81)$. The interval of the xgboost model in task 2 decreases to $(0.87, 0.92)$, which gives us more certainty. However, even in the latter case, we cannot consider the mean estimation 0.896 alone as the variances' scale cannot be ignored. This is particularly important when deciding if one classifier is greater

Table 5.5: Mean AUCs and corresponding standard deviations of the red wine set

Classifiers	Mean AUC		Standard errors			
			Tenfold		Approximation (5.6)	
	Task1	Task2	Task 1	Task 2	Task1	Task2
Logistic Regression (LR)	0.533	0.643	0.071	0.039	0.123	0.069
K-nearest Neighbors (KNN)	0.521	0.675	0.037	0.047	0.122	0.068
Naive Bayes (NB)	0.577	0.774	0.059	0.029	0.124	0.062
Random Forest (RF)	0.506	0.757	0.023	0.05	0.121	0.064
Adaptive Boosting (Adaboost)	0.557	0.692	0.083	0.047	0.124	0.067
Extreme Gradient Boosting (XGBoost)	0.740	0.896	0.131	0.048	0.118	0.047
SMOTE-LR	0.713	0.795	0.067	0.033	0.121	0.060
SMOTE-KNN	0.667	0.802	0.106	0.058	0.124	0.060
SMOTE-NB	0.667	0.770	0.095	0.061	0.124	0.063
SMOTE-RF	0.623	0.816	0.080	0.046	0.125	0.058
SMOTE-AdaBoost	0.672	0.794	0.092	0.043	0.123	0.061
SMOTE-XgBoost	0.693	0.898	0.115	0.040	0.122	0.046

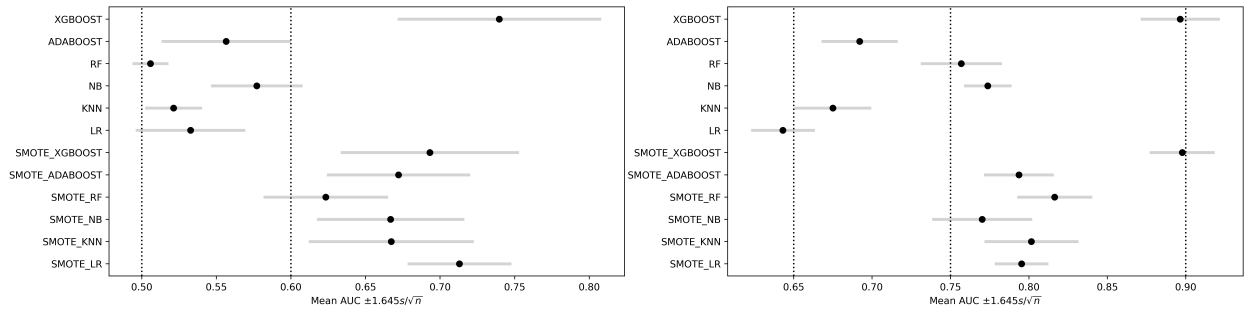

Figure 5.8: Mean AUCs of tenfold validations and the corresponding $\pm 1.645s/\sqrt{10}$ interval of task 1 (left) and task 2 (right) using the red wine data

than another or if one classifier's AUC is above certain thresholds.

By inspecting Table 5.3 and Table 5.5, we see that, even in the real data analysis, the approximation (5.6) still gives conservative or otherwise close results of variances, which is consistent with our finding in the simulation section. Therefore, if one can only go through the test set once due to the computational limit so that sample variances are not available, it is sensible to use the approximation (5.6) as a substitute for the variances estimation.

5.7 Concluding remarks

This chapter aims to make an introduction to the problem of imbalance learning, especially its most popular evaluation metrics - empirical ROC-AUC. We carefully study the statistical

properties of the empirical AUC under the setting of highly imbalanced datasets. When we encounter such scenarios, variances of the empirical AUC could be very high. Therefore, we must report the information on the deviation before we make conclusions about the performance of our models.

Chapter 6

Mixture copulas with discrete margins and their application to imbalanced data

6.1 Introduction

To mitigate the class size imbalance, many methods have been proposed to create an artificially balanced dataset with the same properties as the original data set. For instance, the random oversampling technique creates a larger balanced set by randomly generating instances from the minority class using its empirical distribution. Alternatively, random undersampling erases members of the majority class at random until the dataset is balanced. Some experiments with these two methods for different classifiers can be found in Mohammed et al. (2020). Although these two simple techniques indeed improve the accuracy of the classifier, they are not without drawbacks. By removing members of the majority class, random undersampling methods may discard useful information in the dataset. Methods that employ random oversampling are prone to overfitting (He and Garcia, 2009).

A popular and very powerful alternative that addresses the shortcomings of the random oversampling method is the *synthetic minority over-sampling technique* (SMOTE) introduced in Chawla et al. (2002). Rather than simply using the empirical distribution of the minority class, SMOTE generates new points using a nearest-neighbor approach. First, given a value of K. SMOTE randomly chooses two points: $\mathbf{x}_0$, a base point in the minority class, and $\mathbf{x}'$, which is one of the K nearest neighbors of $\mathbf{x}_0$. A new point $\mathbf{x}^*$ is added to the minority

class, which is a randomly chosen convex combination of $\mathbf{x}_0$ and $\mathbf{x}'$:

$$\mathbf{x}^* = U\mathbf{x}_0 + (1-U)\mathbf{x}', \qquad U \sim \mathcal{U}(0,1),$$

where $\mathcal{U}$ refers to the uniform distribution. SMOTE and its variants enjoy great success in a wide variety of applications; see Fernández et al. (2018) for a recent overview.

However, challenges remain for the SMOTE algorithm. The performance of the algorithm can be highly sensitive to K, whose choice for a particular application can then become somewhat arbitrary. The results sometimes have large variance. Moreover, because the assignment of $\mathbf{x}^*$ uses a uniform distribution, SMOTE may not be suited to skewed data sets (Wang and Japkowicz, 2004; He and Garcia, 2009; Fernández et al., 2018).

To avoid these issues, recently several authors have been using copula functions to implement the oversampling. These functions are powerful tools for modeling the dependence between different factors in the data. Zhu et al. (2019) oversampled an imbalanced data set using a Gaussian copula with a kernel-based marginal distribution. Xue et al. (2022) apply the copula-based oversampling methods in an imbalanced rock burst data set. In this work, the authors use both a Gaussian copula and t-copula with the marginal distribution of each factor chosen by using Kolmogorov–Smirnov (KS) statistics. Both articles show the validity of the methods for their particular data set, as well as superiority over the SMOTE for certain of the classifiers.

Though the copula-based approaches in those manuscripts show promise, they share a shortcoming with the SMOTE method: they are not well designed for data sets with categorical (discrete) marginal factors. However, in many applications (for example the credit card approval task), many of the factors are categorical: educational background, nationality, etc. Moreover, for simplicity, many of the more quantitative variables (income, age) are often placed into categorical bins for study. Simply ignoring the discrete treatment of these factors in the data may hinder the effectiveness of the final results.

We hereby consider the problem of oversampling in data sets with both continuous and discrete features. We introduce the idea of implementing the oversampling using mixture of normal and skew-normal copulas with discrete margins by Bayesian augmentation and the correlated pseudo method in Deligiannidis and Doucet (2018). Our work is an extension of the work of Pitt et al. (2006); Smith and Khaled (2012); Gunawan et al. (2019), where the former two papers introduced Bayesian augmentation approaches to estimate copulas with discrete margins. Gunawan et al. (2019) introduced the work of Deligiannidis and

Doucet (2018) into copulas literature and used the correlated pseudo method to estimate Archimedean copulas. On the other hand, in the paper of Gunawan et al. (2019), their implementations and applications mainly focused on the one-parameter Archimedean families, which might not be well suited for many complex data. We extend their approaches to the normal and skew-normal copulas of any dimensions.

Current studies of copulas with discrete margins largely use Gaussian copulas (Pitt et al., 2006; Smith and Khaled, 2012; Meyer, 2013; Jiryaie et al., 2016). Some authors have also considered cases of Archimedean copulas (Smith and Khaled, 2012; Gunawan et al., 2019; Geenens, 2020) or other classic copulas such as t copulas for the discreteness problems (Smith et al., 2012). In order to make the considerations suitable for higher dimensions as well as complex data, vine copulas are of major interest (Smith, 2011; Smith and Khaled, 2012; Panagiotelis et al., 2012; Loaiza-Maya and Smith, 2019). However, despite the usefulness of mixture models of copulas in modeling complex distribution patterns, they are less studied under the circumstances. Therefore, in this chapter, we study algorithms for estimating parameters of mixture copulas with discrete or mixed margins using Bayesian approaches. Normal and skew-normal mixture copulas are given special attention. Furthermore, we propose to use copula mixture models in the field of imbalanced learning. The integration of Bayesian sampling methods, coupled with the algorithm's capacity to incorporate discrete data features, renders the mixture copulas aptly suited for addressing the real problems in the field of data science.

The rest of this chapter is organized as follows. In section 6.2, we introduce the parametric families of mixture copulas we use throughout the discussion. This is followed by the statistical construction of copulas with discrete margins, which we shall encounter in many real-life applications. Section 6.4 is devoted to discussing the relevancy of identifiability under the setting of this study. Furthermore, section 6.5 introduces the most well-known augmentation techniques when dealing with discreet margins and we outline why this classic approach may not be efficient when complex copula models along with large datasets are applied. In section 6.6, we introduce the main learning algorithm of this study. To test the algorithm, we perform experiments on synthetic and real data in section 6.7.

6.2 Mixture copulas

We utilize mixture of normal copulas and skew-normal copulas. That is, we consider two mixture models

$$C_{\text{NormalMix}} = \sum_{i=1}^{K_1} w_i C_{\text{N}}^{(i)} \quad \text{and} \quad C_{\text{Skew}_\text{m}} = \sum_{i=1}^{K_2} w_i C_{\text{SN}}^{(i)}.$$

Where the density of the normal copula denoted as $c_{\text{N}}^{(i)}$ follows (1.3) and the density of skew normal copula $c_{\text{SN}}^{(i)}$ follows (1.9). Furthermore, we estimate them by Bayesian Markov chain Monte Carlo (MCMC) sampling to enable the model selection and parameter estimation simultaneously by specifying a large K, as we studies in the first chapter, and the redundant groups would be assigned a zero weight asymptotically (Rousseau and Mengersen, 2011).

6.3 The categorical case

We now compute the analog of the copula density in the case that all of the random variables are discrete. We denote these variables as s_j to distinguish them from the continuous case and suppose that there are d of them. The discrete variables in the classification problems of interest are typically data category identifiers, so we further assume that the s_j take on integral values. In this case, it is convenient to define the following difference operator:

$$\Delta_j C(v_1, v_2, \ldots, v_d) \equiv C(v_1, v_2, \ldots, F_j(s_j), \ldots, v_d) - C(v_1, v_2, \ldots, F_j(s_j - 1), \ldots, v_d), \quad (6.1a)$$

$$v_j \equiv F_j(s_j). \quad (6.1b)$$

With the definition in (6.1), we can find the probability mass function by taking repeated differences:

$$p(s_1, s_2, \ldots, s_d) = \Delta_1 \Delta_2 \cdots \Delta_d C\big(F_1(s_1), F_2(s_2), \ldots, F_d(s_d)\big) \equiv \Delta_{1,2,3,\ldots,d} C, \quad (6.2)$$

where $p(\cdot)$ refers to the probability mass function and we have defined the iterated operator Δ for simplicity.

We will now consider cases where the data set contains both continuous and categorical variables.

Assume we have m categorical variables and $d - m$ continuous variables. Therefore, The

distribution can be expressed as

$$F(s_1, s_2, ..., s_m, x_{m+1}, ..., x_d) = C\big(F_1(s_1), F_2(s_2), \ldots, F_m(s_m), F_{m+1}(x_{m+1}), \ldots, F_d(x_d)\big).$$

We are then computing a hybrid between a probability mass and density function. Hence, with first m dimensions to be discrete features, and let $(\mathbf{s}, \mathbf{x}) = (s_1, s_2, ..., s_m, x_{m+1}, ..., x_d)^T$. By assuming the absolutely continuous of the considered copula functions, we have :

$$f(\mathbf{s}, \mathbf{x}) = f(\mathbf{x})p(\mathbf{s} \mid \mathbf{x}) = c(\mathbf{u}) \prod_{j=m+1}^{d} f_j(x_j)\Delta_{1,2,3,\ldots,m}C(\mathbf{v} \mid \mathbf{u}). \tag{6.3}$$

Where

$$\mathbf{v} = (v_1, v_2, \ldots, v_m)^T = \big(F_1(s_1), F_2(s_2), \ldots, F_m(s_m)\big)^T \tag{6.4}$$

is the copula variables with the categorical margins, and

$$\mathbf{u} = (u_{m+1}, u_{m+2}, \ldots, u_{m+d})^T = \big(F_{m+1}(x_{m+1}), F_{m+2}(x_{m+2}), \ldots, F_{m+d}(x_{m+d})\big)^T \tag{6.5}$$

is the copula variables with continuous margins. We denote $C(\mathbf{v} \mid \mathbf{u}) = \int_{\mathbf{0}}^{\mathbf{v}} c(\mathbf{v}' \mid \mathbf{u})\mathbf{dv}'$ to be the conditional copula function given $\mathbf{u}$, $c(\mathbf{u}) = \int c(\mathbf{v}', \mathbf{u})d\mathbf{v}'$ is the marginal copula density of continuous variables.

6.4 Model identifiability

The identifiability problems are important in statistics. As this chapter studies the approaches of discrete copulas and mixture models. One may raise doubt about the model's identifiability.

First of all, as noted in the Sklar theorm, the copulas are only uniquely defined up to the range of marginal distributions. This poses identifiability issues for the copulas with discrete variables as one could use different copulas to construct the same discrete probability distribution. Faugeras (2017) gave examples regarding this problem. This would in general decrease the reliability of any conclusions drawn from a user-chosen copula in modeling procedures if variables have discreetness (Faugeras, 2017; Geenens, 2020). Some tools are available for diagnosing the identifiability of this type. Nasr and Remillard (2023) proved

that for parametric families of copulas with parameters $\theta \in \Theta$, it is identifiable whenever $C_\theta(F_1(x_1), F_2(x_2), \ldots, F_d(x_d))$ is injective with respect to $\theta \in \Theta$. In other words, $C_{\theta_1}(\cdot)$ must not equivalent to $C_{\theta_2}(\cdot)$ when $\theta_1 \neq \theta_2$ in the domain of consideration. When margins are unknown, they suggested using empirical margins to check the conditions. On the other hand, we are in favor of the point raised by the paper that as long as we are aware of the restrictions posed, it is reasonable to proceed by using one particular choice of copulas in applications. For the sampling task considered in the chapter, it is most important to reconstruct the dependency in the domain of concern. That is, abilities to reconstruct the probability distributions through copulas are the main consideration, which is guaranteed by Sklar's theorem.

The identifiability of mixture models of copulas is another potential problem. This refers to the scenario when we have two mixtures and $\sum_{i=1}^{K} p_i F_i = \sum_{j=1}^{K'} p'_j F'_j$ but left and right side are not equivalent up to the label permutation. That is, $p_i = p'_i,\ F_i = F'_i \quad \forall i$ and $K = K'$ does not hold even after any label adjustment. Seminal works regarding this issue for general finite mixture models include Teicher (1961, 1963) and Yakowitz and Spragins (1968). Yakowitz and Spragins (1968) proved that the mixture models are identifiable if and only if the corresponding class of the component-wise distributions is linearly dependent over the real number field.

The identifiability issue of this kind is difficult to address in general and usually needs to be considered case by case for different families of mixtures. Holzmann et al. (2006) proved the identifiability of elliptical mixtures. Otiniano et al. (2015) showed that the multivariate skew normal and zero mean univariate skew t mixtures are identifiable. Therefore, the identifiability of the normal mixture copulas within their own normal parametric family can be readily obtained by recalling the construction formula

$$\sum_i w_i C_i(u_1, u_2, \ldots, u_d; \boldsymbol{P}_i) = \sum_i w_i \Phi_d(\Phi^{-1}(u_1), \Phi^{-1}(u_2), \ldots, \Phi^{-1}(u_d); \boldsymbol{P}_i, \boldsymbol{\mu} = 0).$$

Where $\Phi(\cdot)$ is the distribution of standard normal, $\Phi_d(\cdot; \boldsymbol{P})$ is the zero mean multivariate normal distribution with the standardized covariance matrix $\boldsymbol{P}$. If there exist two different normal copula mixtures such that $\Sigma_i w_i C_i = \Sigma w'_i C'_i$. This means

$$\sum_i w_i \Phi_d(\Phi^{-1}(u_1), \Phi^{-1}(u_2), \ldots, \Phi^{-1}(u_d); R_i) = \sum_i w'_i \Phi_d(\Phi^{-1}(u_1), \Phi^{-1}(u_2), \ldots, \Phi^{-1}(u_d); \boldsymbol{P}'_i)$$

which contradict the identifiability of the normal mixtures.

Deeper identifiable results are not available among mixture copula literature to the best of the authors' knowledge, other works discussed and applied the mixture of copulas either claimed the identifiability is not the key issue in their assignments (Wang, 2008; Cai and Wang, 2014; Mazo and Averyanov, 2019) or totally ignored identifiability problems. Otherwise, they declared it as open questions (Arakelian and Karlis, 2014; Kosmidis and Karlis, 2016; Mazo and Averyanov, 2019). We quote the ideas from Mazo and Averyanov (2019) that although identifiability is very important in statistical theory, verifying it can be difficult and the applied statistical work often achieves satisfactory outcomes for models with identifiability issues, such as neural networks. Hence, for many cases including mixture copulas applications as above, the identifiability problem may be set aside.

Besides, in our study, we use the Bayesian paradigm of estimations, Rousseau and Mengersen (2011) showed us that as long as the parameters $\alpha_1, \alpha_2, \ldots, \alpha_{K'}$ for the Dirichlet weighting prior $\mathbf{w} \sim \mathbf{Dirichlet}(\alpha_1, \alpha_2, \ldots, \alpha_{K'})$ are small enough, along with some other regularity conditions. The overfitted finite mixtures achieve the sparsity. That is, if the samples are from the true model $\sum_{i=1}^{K} p_i f_i$, using the overfitted model $\sum_{i=1}^{K'} p_i f_i$ where $K' > K$ for Bayesian estimations would result in $\sum_{K+1}^{K'} w_i = O(1/\sqrt{n})$ asymptotically under the regularity. This outcome adds another layer of usefulness for the Bayesian.

6.5 Bayesian data augmentation approach

The equation (6.3) naturally motivates us to estimate to copula with mixed margins by Maximum Likelihood Estimation (MLE)

$$\log L(\mathbf{s}, \mathbf{x}) = \log c(\mathbf{u}) + \log \Delta_{1,2,3,\ldots,m} C(\mathbf{v} \mid \mathbf{u}) + \sum_{j=m+1}^{d} \log f_j(x_j), \tag{6.6}$$

where the notation is kept the same as (6.3), (6.4) and (6.5).

However, as suggested by Smith (2011); Smith and Khaled (2012), the calculation of m dimensional discrete features involves $O(2^m)$ evaluations of the copula function for every data point, this becomes computationally prohibited when we encounter high dimensional large data set. In addition, it is not easy to maximize the likelihood in such cases. They suggest using the Bayesian data augmentation approach for parameter learning. Let $(\mathbf{s}^l, \mathbf{x}^l) = (s_1^l, s_2^l, ..., s_m^l, x_{m+1}^l, ..., x_d^l)^T,\ l = 1, 2, \ldots n$ be the n data points and the first m features are discrete. We give an augmented variables $\boldsymbol{\nu} = (\nu_1, \nu_2, ..., \nu_m, \nu_{m+1}, ..., \nu_d)^T$ such that the joint

density is

$$\begin{aligned}\prod_{l=1}^{n} f(\mathbf{s}^l, \mathbf{x}^l, \boldsymbol{\nu}^l) &= \prod_{l=1}^{n} f(\mathbf{s}^l, \mathbf{x}^l \mid \boldsymbol{\nu}^l) c(\boldsymbol{\nu}^l) = \prod_{l=1}^{n} f(\mathbf{x}^l \mid \boldsymbol{\nu}^l) f(\mathbf{s}^l \mid \mathbf{x}^l, \boldsymbol{\nu}^l) c(\boldsymbol{\nu}^l) \\ &= \prod_{l=1}^{n} \Big(\prod_{j=1}^{m} I\big(F_j(s_j^l - 1) < \nu_j^l \ \leq F_j(s_j^l)\big) \prod_{k=m+1}^{d} \delta(F_k(x_k^l) = \nu_k^l) f_k(x_k^l) \Big) c(\boldsymbol{\nu}^l),\end{aligned} \tag{6.7}$$

n represents the total number of points available, $\delta(\cdot)$ is the Dirac delta. In this sense,

$$f(\mathbf{s}, \mathbf{x}) = \int f(\mathbf{s}, \mathbf{x}, \boldsymbol{\nu}) d\boldsymbol{\nu} = \int f(\mathbf{s}, \mathbf{x} \mid \boldsymbol{\nu}) c(\boldsymbol{\nu}) d\boldsymbol{\nu}.$$

From the above, we can naturally use the Gibbs within Metropolis-Hasting (M-H) types of sampling techniques for parameters learning and sample generations, which we summarize as **Algorithm 6.1**.

Algorithm 6.1 Bayesian data augmentation

1: Initialize $\boldsymbol{\nu}$ marginally by using $\hat{F}_{nj}(x) = \frac{1}{n+1} \sum_{i=1}^{n} I(x_{ij} \leq x)$. Initialize copula parameter $\boldsymbol{\Theta}^{(0)}$.
2: **for** $t = 1, 2, \ldots$ **do**
3: **for** $l = 1, 2, \ldots n$ **do**
4: **for** $j = 1, 2, \ldots, m$ **do**
5: Sample $p(\nu_j^l \mid \mathbf{s}, \mathbf{x}, \boldsymbol{\nu}_{\setminus j}^l, \boldsymbol{\Theta}^{t-1}) \propto p(\mathbf{s}, \mathbf{x} \mid \boldsymbol{\nu}^l, \boldsymbol{\Theta}^{t-1}) c(\nu_j^l \mid \boldsymbol{\nu}_{\setminus j}^l, \boldsymbol{\Theta}^{t-1}) =$

$$\Big(\prod_{j=1}^{m} I\big(F_j(s_j^l - 1) < \nu_j \leq F_j(s_j^l)\big) \prod_{k=m+1}^{d} I(F_k(x_k^l) = \nu_k) f_k(x_k^l) \Big) c(\cdot).$$

This can be generated by $\mathrm{u}' \sim \text{Uniform}(\hat{F}_{nj}(x_j^l - 1), \hat{F}_{nj}(x_j^l))$ and $\nu_j^l = C^{-1}(\mathrm{u}' \mid \boldsymbol{\nu}_{\setminus j}^l)$.
6: **end for**
7: **end for**
8: Sample the parameter $\boldsymbol{\Theta}^t$ by $p(\boldsymbol{\Theta}^t \mid \boldsymbol{\nu}, \mathbf{x}, \mathbf{s}) = p(\boldsymbol{\Theta}^t \mid \boldsymbol{\nu}) \propto \prod_{l=1}^{n} c(\boldsymbol{\nu}^l; \boldsymbol{\Theta}^t) \pi(\boldsymbol{\Theta}^t)$. More specifically, $\boldsymbol{\Theta}^t \sim p(\boldsymbol{\Theta} \mid \boldsymbol{\nu})$ is sampled by Metropolis-Hasting methods:
We propose the parameters by a proposal $\Theta^{\text{prop}} \sim pp(\boldsymbol{\Theta} \mid \boldsymbol{\Theta}^{(t-1)})$ where $pp(\cdot \mid \boldsymbol{\Theta}^{t-1})$ is a proposal distribution, the new proposal is accepted according to the M-H acceptance probability.
9: Implementing the oversampling by $\mathbf{u}^{\text{new}} \sim c(\mathbf{u} \mid \boldsymbol{\Theta}^t)$ and $(\mathbf{s}, \mathbf{x})_j^{\text{new}} = \hat{F}_{nj}^{-1}(\mathrm{u}_j^{\text{new}})$ for $j = 1, 2, 3, ..., d$, where $\mathbf{u}^{\text{new}} = (\mathrm{u}_1^{\text{new}}, \mathrm{u}_2^{\text{new}}, ..., \mathrm{u}_d^{\text{new}})^T$.
10: **end for**

The conditional copula needs to be computed when sampling ν_j^l for $j = 1, 2, \ldots, m,\ l =$

$1, 2, \ldots, n$. This can be derived from

$$C_{\text{Normal}}(u_i \mid u_1, u_2, ..., u_{i-1}, u_{i+1}, ..., u_d) = F(\Phi^{-1}(u_i) \mid \Phi^{-1}(u_1), \Phi^{-1}(u_2), ..., \Phi^{-1}(u_d)),$$

F can be obtained from the formula of conditional normal distribution $X_i \mid \mathbf{Y} = y$ with $X \sim N(0,1)$ and $\mathbf{Y} \sim \mathrm{N}_{d-1}(0, M_{ii})$. M_{ii} refers to the correlation matrix $\boldsymbol{P}$ with i^{th} row and i^{th} columns deleted.

On the other hand, Sampling from the skew-normal distribution requires extra parameters of skewness $\boldsymbol{\delta}^T = (\delta_1, \delta_2, ..., \delta_d)$ where we can propose each δ_i by truncated normal distribution from -1 to 1 with the mean being $\delta_i^{\text{current}}$. The conditional copula $C_{\text{SN}}(u_i \mid u_1, u_2, ..., u_{i-1}, u_{i+1}, ...u_d) = F_{\text{SN}}(F_i^{-1}(u_i) \mid F_1^{-1}(u_1), F_2^{-1}(u_2), ..., F_d^{-1}(u_d))$ is more involved, as per Azzalini (2013)[Section 5.3], the conditional distribution $F_{\text{SN}}(\cdot \mid \cdot)$ follows the extended skew normal distribution, the density of which is denoted as

$$\mathrm{ESN}_d(\boldsymbol{\mu}, \boldsymbol{P}, \boldsymbol{\alpha}, \tau) = \phi_d(\boldsymbol{x} - \boldsymbol{\mu}; \boldsymbol{P})\Phi(\tau(1 + \boldsymbol{\alpha}^T \boldsymbol{P} \boldsymbol{\alpha})^{1/2} + \boldsymbol{\alpha}^T(\boldsymbol{x} - \boldsymbol{\mu}))/\Phi(\tau).$$

Let $\mathbf{X} = (X_1, \mathbf{X_2}^T)^T$ and $\boldsymbol{P}$ and $\boldsymbol{\alpha}$ is partitioned into $\boldsymbol{P}_{11}, \boldsymbol{P}_{12}, \boldsymbol{P}_{22}, \boldsymbol{P}_{21}$ and α_1, α_2 according to $X_1, \mathbf{X}_2$. We have

$$X_1 \mid \mathbf{X_2} = \mathbf{x}_2 \sim \mathrm{ESN}_1(\varepsilon_{1\cdot 2}, \boldsymbol{P}_{11\cdot 2}, \alpha_1, \tau_{1\cdot 2}).$$

Where we follow the notation of Azzalini (2013)[p.130][p.151],

$$\begin{aligned}
\boldsymbol{P}_{11\cdot 2} &= \boldsymbol{P}_{11} - \boldsymbol{P}_{12}\boldsymbol{P}_{22}^{-1}\boldsymbol{P}_{21} \\
\varepsilon_{1\cdot 2} &= \boldsymbol{P}_{12}\boldsymbol{P}_{22}^{-1}(\mathbf{x}_2) \\
\tau_{1\cdot 2} &= \left(\frac{\alpha_2 + \boldsymbol{P}_{22}^{-1}\boldsymbol{P}_{21}\alpha_1}{\sqrt{1 + \alpha_1' \boldsymbol{P}_{11\cdot 2}\alpha_1}}\right)^T \mathbf{x}_2.
\end{aligned}$$

This method works well when the analytical forms of the conditional copulas and their corresponding inversions are available. However, for complex copula models, one often needs to obtain the inversions numerically, this task is computationally demanding and sometimes unstable when the discrete dimensions m and the sample size n become large. As for each iteration, we need to sample $O(nm)$ from conditional copulas. Gunawan et al. (2019) used the pseudo marginal method based on unbiased estimators of likelihood functions and applied it to learn the one-parameter Archimedean copulas, they showed that this largely improved the computational time compared with the augmentation method. We extend their work to

the mixture copulas of high dimensional normal and skew-normal copulas, which could be more applicable when we analyze complex high dimensional data structures.

6.6 Methodology

Consistent with what we have discussed in the Algorithm 6.1 and previous chapters, we learn the marginal cumulative distribution with its modified empirical counterpart

$$\hat{F}_{nj}(x) = \frac{1}{n+1} \sum_{i=1}^{n} I(x_{ij} \leq x). \tag{6.8}$$

Where x_{ij} is the j^{th} dimension of the i^{th} data, $i = 1, 2, \ldots, n$.

In the continous case, Bayesian learning would be directly applied by using $p(\theta \mid \mathbf{x}) \propto f(\mathbf{x} \mid \theta)\pi(\theta)$ where MCMC is often used for sampling the posterior. Oversampling the minority class in the data set is through

$$\theta^* \sim p(\theta \mid \mathbf{x})$$
$$y' \sim f(\mathbf{x} \mid \theta^*)$$

For the data set with discrete features, extra attention needs to be paid. Algorithm 6.1 uses Bayesian augmentation approach for modelling, we approach the problem here from different angles. Let $(\mathbf{s}, \mathbf{x}) = (s_1, s_2, ..., s_m, x_{m+1}, ..., x_d)^T$ be the d-dimensional data with first m features are discrete. From (6.3), if the copula distribution $C(\cdot)$ is absolutely continuous with the density $c(\cdot)$, similar as what have been presented in Gunawan et al. (2019), we can write (6.3) as

$$L(\mathbf{s}, \mathbf{x}) = \int_{\boldsymbol{F(s-1)}}^{\boldsymbol{F(s)}} c(\mathbf{v}', \mathbf{u}) d\mathbf{v}' \prod_{j=m+1}^{d} f_j(x_j). \tag{6.9a}$$

Where $\boldsymbol{F(s-1)} = \big(F_1(s_1 - 1), F_2(s_2 - 1), \ldots, F_m(s_m - 1)\big)$ and $\mathbf{u}$ follows (6.5) such that

$$\mathbf{u} = (u_{m+1}, u_{m+2}, \ldots, u_{m+d})^T = \big(F_{m+1}(x_{m+1}), F_{m+2}(x_{m+2}), \ldots, F_{m+d}(x_{m+d})\big)^T.$$

By change of variables of the integration,

$$L(\mathbf{s},\mathbf{x}) = \prod_{i=1}^{m} \big[F_i(s_i) - F_i(s_i - 1)\big] \int_{\mathbf{0}}^{\mathbf{1}} c(\mathbf{v}'' \bigodot \big(\mathbf{F}(\mathbf{s}) - \mathbf{F}(\mathbf{s}-\mathbf{1})\big) + \mathbf{F}(\mathbf{s}-\mathbf{1}), \mathbf{u}) d\mathbf{v}'' \prod_{j=m+1}^{d} f_j(x_j), \tag{6.9b}$$

where $\bigodot$ refers to the component-wise product of vectors.

More specifically,

$$\begin{aligned} &\big(\mathbf{v} \bigodot \big(\mathbf{F}(\mathbf{s}) - \mathbf{F}(\mathbf{s}-\mathbf{1})\big) + \mathbf{F}(\mathbf{s}-\mathbf{1}), \mathbf{u}\big)_j \\ &= \begin{cases} v_j\big(F_j(s_j) - F_j(s_j - 1)\big) + F_j(s_j - 1), & j = 1, 2, \ldots, m \\ u_j \quad j = m+1, m+2, \ldots, d \end{cases} \end{aligned} \tag{6.10}$$

This motive us to approximate the integral of (6.9b) by Monte Carlo

$$\begin{aligned} &L(\mathbf{s},\mathbf{x}) \approx \\ &\prod_{j=m+1}^{d} f_j(x_j) \prod_{i=1}^{m} \big[F_i(s_i) - F_i(s_i - 1)\big] \frac{1}{N'} \sum_{j=1}^{N'} c\big(\mathbf{p_j} \bigodot \big(\mathbf{F}(\mathbf{s}) - \mathbf{F}(\mathbf{s}-\mathbf{1})\big) + \mathbf{F}(\mathbf{s}-\mathbf{1}), \mathbf{u}\big) \\ &:= L(\mathbf{s},\mathbf{x},\mathbf{p}) \end{aligned} \tag{6.11}$$

Where $\mathbf{p_j} \sim \mathcal{U}_m(0,1)$ is the m-dimensional uniform distribution and N' is predefined. The equation (6.11) gives an unbiased estimation of (6.9b) numerically. Noticing that

$$p(\theta \mid \mathbf{s},\mathbf{x}) \propto L_\theta(\mathbf{s},\mathbf{x})\pi(\theta) = \pi(\theta) \int_0^1 L_\theta(\mathbf{s},\mathbf{x},\mathbf{p}) f_{\mathcal{U}_m}(\mathbf{p}) d\mathbf{p}.$$

Sampling the posterior of θ can be realized by sampling $p(\theta, \mathbf{p} \mid \mathbf{s},\mathbf{x}) \propto p(\theta \mid \mathbf{p},\mathbf{s},\mathbf{x}) f_{\mathcal{U}_m}(\mathbf{p})$ and take the marginal part, where $f_{\mathcal{U}_m}$ is denoted as the density of m-variates uniform distribution. Gibbs-M-H types algorithm can therefore be constructed.

To realize the sampling of mixture copulas, we assign the group label $k_j \in \{1, 2, 3, \ldots, K\}$, for our observations $j = 1, 2, \ldots, n$. The prior of the group weight is the Dirichlet distributions

$$\pi(\mathbf{w}) \sim \mathbf{Dirichlet}(1/K, 1/K, \ldots, 1/K).$$

Therefore, we present the pseudo marginal algorithm for mixture copula with discrete and mixed margins in **Algorithm 6.2**, which circumvents the necessity of sampling from

the conditional copulas of every dimension and every data point.

6.7 Algorithm validation

In order to validate our approach, we firstly use it to learn the synthetic data sampled from mixture copulas of our own design so that the correctness of the sampler can be empirically tested. Then, we solve classification problems involving real experimental data by oversampling from mixture copulas.

6.7.1 Synthetic data

For the first synthetic test, we simulate the data from a 3-dimensional mixture normal copula and discretize it using categorical marginal distributions. In particular, the marginal distribution is set to be **Categorical**$(a_1, a_2, \ldots, a_{10})$ where $a_1 : a_2 : a_3, \cdots : a_{10} = 1 : 2 : 3 \cdots : 10$. The sample data are transformed using $x_{ij} = F^{-1}(u_{ij})$ where $F^{-1}(\cdot)$ is the inverse of the categorical distribution. As for the copula, we use

$$c_{\mathrm{m}}(\mathbf{v}) = w_1 c_{1\mathbf{N}}(\mathbf{v}; \boldsymbol{\rho}^1 = (0.6, -0.5, -0.6)^T) + w_2 c_{2\mathbf{N}}(\mathbf{v}; \boldsymbol{\rho}^2 = (0.8, 0.7, 0.8)^T), \tag{6.13}$$

where the subscript "m" refers to "mixed" , $\mathbf{v} \in [0,1]^3$ and $\boldsymbol{\rho} = (\rho_{12}, \rho_{13}, \rho_{23})^T$ determine the corresponding correlation matrix . We generate

$$(n_1, n_2) = (200, 0), (500, 0), (1000, 0), (150, 50), (375, 125), (750, 250)$$

points from the first and second copulas $c_{1\mathbf{N}}$ and $c_{2\mathbf{N}}$ respectively and since the data points are exchangeable, this corresponding to estimate copulas with $(w_1, w_2) = (1, 0), (0.75, 0.25)$.

We set the initial number of mixture components K to be 3 and let the algorithm in the section 6.6 to decide if it is appropriate. We generate 5000 posterior points for each parameter. The commonly occurring label-switching problems are solved by ranking the group number according to their weights at the end of each iteration. We calculate the posterior mean of parameters after discarding the first 3000 points through burn-in. The experiments stated above were repeated 30 times for each sample size and the estimations for the posterior means were averaged over repetitions and the corresponding standard deviations were calculated. Table 6.1 displays the results. As we have overfitted the number of groups $K = 3$, we can see that the algorithm correctly selects the number of groups even

Algorithm 6.2 Bayesian pseudo correlated method for mixture copula with mixed margins

1: Data points are of the form $\{(\mathbf{s}_i, \mathbf{x}_i) : i = 1, 2, \ldots, n\}$, $\mathbf{u}_{ij} = \hat{F}_{nj}(\mathbf{s}_i, \mathbf{x}_i)_j$, Initialize number of clusters K, Monte Carlo precision N' as specified in (6.11), copula data with group labels $(\mathbf{u}_i^T, k_i)$, copula parameters for K copulas $\boldsymbol{\Theta}_c^{(0)}, c = 1, 2, \ldots, K$, group weightings $\mathbf{w}^{(0)}$, N' points of m dimensional (the dimension of discrete features) uniform samples $\boldsymbol{P}_i^{(0)} = \{\mathbf{p_{i1}}^{(0)}, \mathbf{p_{i2}}^{(0)}, \ldots, \mathbf{p_{iN'}}^{(0)}\}$ for every $i = 1, 2, \ldots, n$.

2: **for** $t = 1, 2, \ldots$ Max-iteration **do**

3: **for** $k = 1, 2, \ldots K$ **do**

4: Propose the k^{th} component copula parameters $\boldsymbol{\Theta}_k^{\text{prop}}$, For the skew normal copulas. These include the correlation matrix Σ_k and the skewness parameter $\boldsymbol{\delta}_k$. Sample $\boldsymbol{P}'_{\{i:k_i^{t-1}=k\}}$ with high correlation (e.g. 0.99) from last sample $\boldsymbol{P}^{t-1}_{\{i:k_i^{t-1}=k\}}$ to ensure a good convergence property (Deligiannidis and Doucet, 2018), which can be done by sampling from correlated normal and do the conversion. We accept $\boldsymbol{\Theta}_k^{\text{prop}}$ and $\boldsymbol{P}'_{\{i:k_i^{t-1}=k\}}$ with the probability

$$\alpha_{\text{acceptance}} = \min\left\{\frac{\prod_{\{l:k_l^{(t-1)}=k\}} \frac{1}{N'} \sum_{j=1}^{N'} c^{(k)}_{\boldsymbol{\Theta}^{\text{prop}}}(\mathbf{p'_{lj}} \odot (\hat{\mathbf{F}}_{\mathbf{n}}(\mathbf{s_l}) - \hat{\mathbf{F}}_{\mathbf{n}}(\mathbf{s_l - 1})) + \hat{\mathbf{F}}_{\mathbf{n}}(\mathbf{s_l - 1}), \mathbf{u_l}) \pi(\boldsymbol{\Theta}^{\text{prop}}) pp(\boldsymbol{\Theta}^{t-1} | \boldsymbol{\Theta}^{\text{prop}})}{\prod_{\{l:k_l^{(t-1)}=k\}} \frac{1}{N'} \sum_{j=1}^{N'} c^{(k)}_{\boldsymbol{\Theta}^{t-1}}(\mathbf{p^{(t-1)}_{lj}} \odot (\hat{\mathbf{F}}_{\mathbf{n}}(\mathbf{s_l}) - \hat{\mathbf{F}}_{\mathbf{n}}(\mathbf{s_l - 1})) + \hat{\mathbf{F}}_{\mathbf{n}}(\mathbf{s_l - 1}), \mathbf{u_l}) \pi(\boldsymbol{\Theta}^{\text{prop}}) pp(\boldsymbol{\Theta}^{\text{prop}} | \boldsymbol{\Theta}^{t-1})}, 1\right\}. \tag{6.12}$$

The methods of sampling of the correlation Σ and $\boldsymbol{\delta}$ has been discussed in section 6.5.

5: **end for**

6: Relocate each data points into groups by

$$p(k_j = i \mid \boldsymbol{P}^t, \mathbf{u}_j, \mathbf{x}_j, \mathbf{s}_j, \Theta^t) \propto w_i^{t-1} \frac{1}{N'} \sum_{r=1}^{N'} c^{(i)}_{\Theta^t}\big(\mathbf{p^{(t)}_{jr}} \bigodot \big(\hat{\mathbf{F}}_{\mathbf{n}}(\mathbf{s_j}) - \hat{\mathbf{F}}_{\mathbf{n}}(\mathbf{s_j - 1})\big) + \hat{\mathbf{F}}_{\mathbf{n}}(\mathbf{s_j - 1}), \mathbf{u_j}\big)$$

7: Relocate the weight

$$w_i^t \sim \textbf{Dirichlet}(1/K + \sum_{i=1}^{n} \mathrm{I}(k_i = 1), 1/K + \sum_{i=1}^{n} \mathrm{I}(k_i = 2), \ldots, 1/K + \sum_{i=1}^{n} \mathrm{I}(k_i = K))$$

8: Sampling from the copula by first choosing the group

$$k^* \sim \textbf{Categorical}(w_1^t, w_2^t, \ldots, w_k^t)$$

and $\boldsymbol{u}^* \sim c^{k*}(\cdots \mid \boldsymbol{\Theta}^t_{k^*})$.

9: $(\mathbf{s}^*, \mathbf{x}^*)_j = \hat{F}_j^{-1}(\mathrm{u}_j^*)$ for $j = 1, 2, 3, ..., d$, where $\mathbf{u}^* = (\mathrm{u}_1^*, \mathrm{u}_2^*, ..., \mathrm{u}_d^*)^T$.

10: **end for**

for relatively small sample sizes. Only insignificant amount of weightings are assigned to the empty components. With the increase of the data points, the posterior means show a good sign of convergence.

Table 6.1: Means and standard deviations of the posterior mean estimators for synthetic discrete data from normal copulas over 30 repetitions of MCMC experiments. Mean±sd are reported, $\boldsymbol{\rho}^1, \boldsymbol{\rho}^2$ are the correlations for the first and second normal copulas. The number of uniform samplings $N' = 30$.

n_1,n_2	$200, 0$	$500, 0$	$1000, 0$	$150, 50$	$375, 125$	$750, 250$
w_1	0.88 ± 0.08	0.92 ± 0.07	0.94 ± 0.05	0.68 ± 0.08	0.69 ± 0.07	0.71 ± 0.06
w_2	0.10 ± 0.06	0.07 ± 0.06	0.05 ± 0.05	0.26 ± 0.06	0.24 ± 0.04	0.24 ± 0.03
ρ^1_{12}	0.60 ± 0.06	0.61 ± 0.03	0.60 ± 0.02	0.60 ± 0.07	0.62 ± 0.04	0.61 ± 0.03
ρ^1_{13}	-0.52 ± 0.06	-0.51 ± 0.04	-0.49 ± 0.03	-0.43 ± 0.15	-0.50 ± 0.06	-0.51 ± 0.04
ρ^1_{23}	-0.61 ± 0.05	-0.61 ± 0.03	-0.60 ± 0.02	-0.53 ± 0.13	-0.61 ± 0.06	-0.60 ± 0.04
ρ^2_{12}				0.71 ± 0.12	0.76 ± 0.05	0.78 ± 0.06
ρ^2_{13}				0.45 ± 0.25	0.56 ± 0.18	0.63 ± 0.14
ρ^2_{23}				0.51 ± 0.27	0.64 ± 0.17	0.72 ± 0.17

For the skew-normal copula, the estimation of the parameters are more difficult, especially for the $\boldsymbol{\delta}$ parameters. We sample from

$$\begin{aligned} c_{\text{skew}_\text{m}}(\mathbf{v}) = w_1 c_{1\mathbf{SN}}(\mathbf{v}; \boldsymbol{\rho}^1 = (0.6, 0.6, 0.6)^T, \boldsymbol{\delta}^1 = (0.8, 0.8, 0.8)^T) \\ + w_2 c_{2\mathbf{SN}}(\mathbf{v}; \boldsymbol{\rho}^2 = (-0.8, -0.8, 0.8)^T, \boldsymbol{\delta}^2 = (-0.8, -0.8, -0.8)^T). \end{aligned} \tag{6.14}$$

Data points are converted similarly but using the categorical distribution with 30 categories, and the corresponding probability for each category is $a_1 : a_2 : \ldots a_{30} = 1 : 2 : \cdots : 30$. We report the results with two sets of data which are

$$(n_1, n_2) = (2000, 0), (1500, 1000), (3000, 1000).$$

This corresponds to $(w_1, w_2) = (1, 0), (0.6, 0.4), (0.75, 0.25)$. We estimate the parameters by setting $K = 2$. The MCMC method is implemented for 5000 iterations, with the first 2000 points discarded for the first two experiments and 3000 points discarded for the last experiment as burn-in. Due to the computational burden, the experiments are not repeated. Table 6.2 shows the results. Noticeably, the first component of the skew-normal copula when $(w_1, w_2) = (0.6, 0.4)$ is not correctly estimated, although other parts of the results are reasonably acceptable. In general, we find out through multiple experiments that the learning of mixture skew normal copulas sometimes requires much more data than the corresponding

mixture normal copulas, especially for the skewness parameters $\boldsymbol{\delta}$. On the other hand, increasing the number of uniform samples N' as introduced in (6.11) could lead to faster mixing of the MCMC sampler. However, this would lead to slower computational iterations.

Table 6.2: Posterior mean and standard deviation estimators of synthetic discrete data from skew normal copula with the form Mean±sd. The number of uniform samplings $N' = 30$.

n_1, n_2	2000, 0	1500, 1000	3000, 1000
w_1	0.98 ± 0.03	0.59 ± 0.03	0.78 ± 0.02
w_2	0.02 ± 0.03	0.41 ± 0.03	0.22 ± 0.02
$\boldsymbol{\rho}^1$	$(0.63 \pm 0.04, 0.67 \pm 0.02, 0.69 \pm 0.02)^T$	$(0.75 \pm 0.08, 0.77 \pm 0.09, 0.74 \pm 0.02)^T$	$(0.61 \pm 0.04, 0.64 \pm 0.03, 0.64 \pm 0.03)^T$
$\boldsymbol{\delta}^1$	$(0.76 \pm 0.07, 0.76 \pm 0.06, 0.67 \pm 0.06)^T$	$(0.03 \pm 0.18, -0.01 \pm 0.18, -0.3 \pm 0.24)^T$	$(0.78 \pm 0.07, 0.80 \pm 0.06, 0.77 \pm 0.05)^T$
$\boldsymbol{\rho}^2$		$(-0.72 \pm 0.10, -0.72 \pm 0.10, 0.83 \pm 0.02)^T$	$(-0.75 \pm 0.10, -0.78 \pm 0.08, 0.83 \pm 0.03)^T$
$\boldsymbol{\delta}^2$		$(-0.84 \pm 0.07, -0.68 \pm 0.08, -0.71 \pm 0.08)^T$	$(-0.82 \pm 0.06, -0.66 \pm 0.14, -0.64 \pm 0.14)^T$

6.7.2 Real experimental data

To test our approach against real data, we select 3 imbalanced datasets from KEEL (Alcalá-Fdez et al., 2009), which are *abalone9-18, car-vgood* and *kr-vs-k-zero-one_vs_draw*. The abalone data set contains eight attributes of captured abalones, which are used to predict if the abalone is an older one or a young one. Only the first measurement is categorical (so $m = 1$) with three levels; the remaining seven factors are continuous (so $d - m = 7$). The data is highly imbalanced: only 42 of the 731 total instances belong to the "older" class. The car dataset includes 1728 observations, 6 categorical features are used to predict if the car has a "very good" quality, only 65 instances out of the total samples belong to "very good" class. The last mentioned data set is a chess data set. There are 2901 observations in total with six categorical features indicating the status of the current game. We use the features to predict the outcome of games, the dataset only contains 3.6% positive instances.

We split the data using the random hold-out method. The car dataset is separated with $90\% - 10\%$ train-test set ratio. The chess dataset is divided according to $80\% - 20\%$ train-test ratio and the abalone dataset is divided into $70\% - 30\%$ train-test ratio. We use different ratios to ensure that there are enough minority samples for us to train models. In order to keep the proportion between majority and minority classes in our training and test sets, we use the stratified train test split. That is, the proportion between classes are kept the same in train and test set when we conduct the splitting. Finally, the random hold out approach is used for 5 times in each dataset. Figure 6.1 shows the scatter plot of the minority class in the abalone dataset between $(U_i, U_j) = (\hat{F}_{ni}(x_i), \hat{F}_{nj}(x_j))$ for $i, j = 1, ..., 8$, where $\hat{F}_{ni}(\cdot)$ is defined in (6.8). Since the first attribute is discrete, U_1 is sampled uniformly from $[\hat{F}_{n1}(x_1 - 1), \hat{F}_{n1}(x_1)]$ for every instance on the plot.

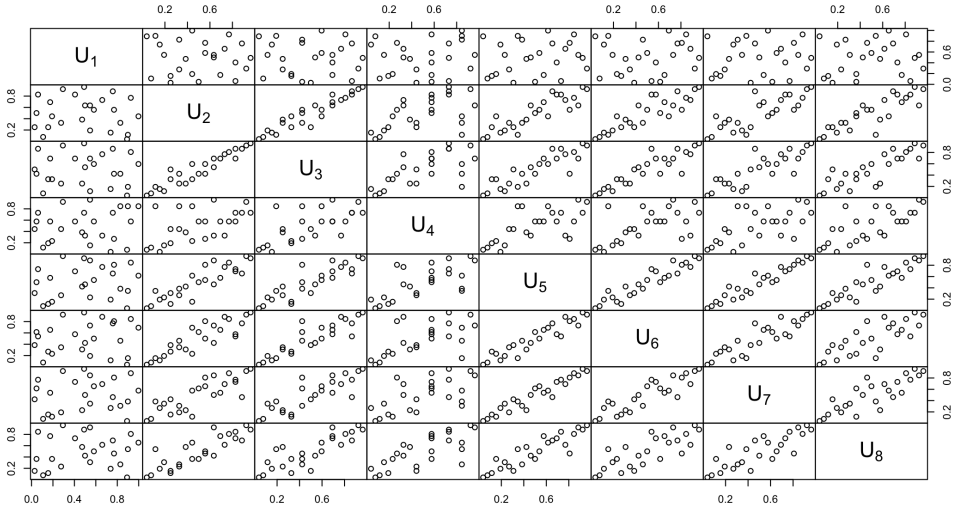

Figure 6.1: Pairs plots between the attributes for the minority class in the training set

Since our datasets are imbalanced, we oversample the minority class using the mixture copulas to balance the training set. As before, we use MCMC techniques following the Algorithm 6.2 to generate 2500 points. Sufficient samples up till the last are used to balance the set and the remaining are discarded. We use this approach with our two copula methods, random oversampling, and SMOTE.

Table 6.3: Comparison of different oversampling methods for the 3 datasets. 'Car' refers to *car-vgood*. 'Abalone' refers to *'abalone9-18'* and 'Chess' refers to *kr-vs-k-zero-one_vs_draw*. Classifiers are random forest (RF), support vector machine (SVM), and logistic regression (LR). Values shown are mean and sd estimators of ROC-AUC for 5 experiments. Bold values are best.

	Car			Abalone			Chess		
Classifier	RF	SVM	LR	RF	SVM	LR	RF	SVM	LR
Oversampling method	Mean ROC-AUC ± SD								
Normal copula	.992 ± .005	.992 ± .005	.890 ± .059	.657 ± .063	.747 ± .053	.817 ± .063	**.993 ± .003**	.981 ± .012	.967 ± .006
Skew normal copula	.992 ± .005	.992 ± .005	**.904 ± .039**	**.664 ± .047**	**.760 ± .048**	.809 ± .037	.980 ± .022	.965 ± .034	**.968 ± .006**
Random oversampling	.995 ± .005	.995 ± .004	.896 ± .056	.537 ± .030	.733 ± .053	**.857 ± .048**	.993 ± .011	.994 ± .002	.962 ± .022
SMOTE	**.997 ± .005**	**.996 ± .005**	.900 ± .056	.661 ± .050	.720 ± .049	.855 ± .069	.984 ± .012	**.994 ± .001**	.962 ± .022
Original data set	.871 ± .154	.943 ± .060	.527 ± .039	.523 ± .034	.515 ± .034	.694 ± .079	.947 ± .036	.995 ± .061	.880 ± .047

We then apply the random forest, support vector machine, logistic regression classifiers to learn the parameters from the balanced training datasets and test them in the test sets. Every experiments are repeated 5 times as we split the data 5 times using the random hold out approach and we calculate the mean and sd estimators from there; the results are shown in Table 6.3. For 9 comparisons over different classifiers and datasets. The copula methods

win 5 times. We can say that the copula oversampling methods do perform better than the random oversampling and SMOTE under many circumstances. On the other hand, all copula models perform significantly better in the statistical sense than the original unbalanced data. Therefore, the approach is promising when marginals of the data display highly correlated complex patterns, especially if the margins are mixed with continuous and discrete features which may not be handled well with the classical SMOTE or random oversampling methods.

6.8 Concluding remarks

This chapter is an application of Bayesian copula models to the problems of imbalance learning. To deal with the multi-modal correlation structure, we incorporated the mixture copula model (Arakelian and Karlis, 2014), which is useful for processing the complex real dataset. In addition, we consider the copulas with both continuous and discrete margins. This consideration, although less frequently occurred in the copula literature, is suitable when applied to daily data science tasks because the discreteness of features is common in the application such as credit card approval, financial fraud detection, and spam mail detection.

Experiments have shown that our mixture copula oversampling significantly improves the ability of classifiers in all datasets, and shows its merits in many datasets when compared with the classic oversampling methods.

We focused on two types of copulas: normal and skew-normal in this chapter. But any of the wide variety of copulas in the literature can be used with our approach, which will cause further advancement in this study of imbalanced learning and clustering. If the data set has very few points, the one-parameter Archimedean family of the copula in Genest and Rivest (1993) can be used. Moreover, various copula selection approaches such as Huard et al. (2006) may be further considered when we select the best model for the data set. These additional cases could be explored in further research.

Chapter 7

Conclusions

This thesis studies the estimation and selection of mixture copulas from the Bayesian viewpoint, Non-parametric Bayesian approach is given specific attention. Notably, the method discussed in Chapter 2 can be viewed as a finite approximation of the later chapters using non-parametric Dirichlet priors. Therefore, the later work from Chapter 2 is indeed a refinement by making the methodology more precise. For the application, we study the classic copula application among major financial markets. We hope this can reflect the latest characteristic of the dependence mode among exchanges. In addition, we study the class imbalance problem, an essential concept in data science, from a statistical perspective. More specifically, we discuss the properties of the evaluation metric ROC-AUC, which is frequently used in the field. We further merge the copula tools into the field to improve the oversampling problems.

Chapter-wisely, in Chapter 2, we discussed the method of selecting and estimating the finite mixture copula simultaneously using the Bayesian approach. This is mainly realized by utilizing the Dirichlet distribution as the weight prior, which is the finite approximation of the Dirichlet process. We first overfit the model with all potential mixture components and then estimate the parameters by Bayesian methods. The MCMC and EM methods are proposed to learn the parameters, and we have performed numerical simulations to validate the correctness. Furthermore, we apply the methodology to the financial markets to detect the asymmetry dependencies among them.

Chapter 3 constructed an infinite mixture model of a t copula by using the DP process prior along with the Gibbs-MH sampler. The model was evaluated using simulation experiments and real data analysis. The results obtained from the simulation experiments indicate the reliability of the approach when compared with the benchmark standard MLE

estimation method embedded in the copula library of R. The Bayesian estimation results of the real data analysis using the daily closing prices of the Shanghai and Shenzhen indexes from 2018 to 2023 further confirm the quality of the model estimator when compared with the results of the MLE.

Chapter 4 proposed an estimation method for a copula infinite hidden Markov model using a hierarchical Dirichlet approach. The Bayesian MCMC sampler is introduced, accompanied by simulation studies of the 2-state t-copula hidden Markov models and 3-state t-copula models. In the real data analysis section, the daily closing prices of SSECI-HSI and SPX-FTSE were used to train the model. Some characteristics of market dependence could be derived from the estimation outcomes. Introducing the copula-iHMM model could be beneficial for high-dimensional time-series modeling, particularly for cases in which dependence patterns among series must be estimated. In contrast, for the classic HMM approach, the number of states K must be specified as a hyperparameter. This might be inconvenient if K changes rapidly when new data are included or if K is considerably large. The iHMM structure is a convenient tool for determining K automatically.

Chapter 5 introduced the field of imbalance learning and the implications of estimating empirical AUCs for a highly imbalanced dataset. The theoretical results reveal that, in many practical situations, variances of the Mann-Whitney U statistics increase monotonically with respect to the imbalanced level of the data, which could result in a highly volatile empirical AUC estimator if the test datasets have medium to small sample sizes and the test sets have high-class imbalance. Numerical simulations are performed in varying cases to confirm this finding. To show the implications for experimental design, we calculate the sample size required for performing hypothesis tests with the required statistical power under imbalance scenarios. Finally, we use two real datasets to demonstrate our findings and illustrate potential problems that should receive attention when computing classifiers' AUC using the empirical estimator. For the application with small sample sizes and high-class imbalance, the information on the variances of metrics must be included to ensure the validity of the findings.

When faced with imbalanced data sets, many algorithms implement a preprocessing step to oversample the minority class in order to obtain a balanced training set. In the work of Chapter 6, we introduced the algorithm for learning the mixture copula with mixed margins and applied the approach for performing the oversampling. This enables us to oversample data with both discrete and continuous features. The classical random oversampling method replicates points from the existing distribution and hence is prone to overfitting. In contrast,

our proposed copula methods may generate new points with the correlation between margins already captured and hence are less prone to overfit. Another classical method, the SMOTE algorithm, is not naturally applicable to the discrete features. This may cause problems in cases where discrete data is an important attribute. We applied our method to both synthetic data to validate its correctness and used real-life datasets to perform the oversampling. Our copula approach has shown some merits over the benchmark methodologies. Although under some circumstances random oversampling and SMOTE still performed best, our methods were competitive as can be seen. Therefore, this new methodology can be incorporated into the oversampling toolbox for more applications.

For future work, first of all, efficient Bayesian inference approaches can be sought, for example, the variational inference and its variant (Kingma and Welling, 2013; Blei et al., 2017). This is in particular very useful when we need to apply our method to high-dimensional big data applications. Some theoretical aspects can also be studied, such as model identifiability in terms of different types of mixture copulas. Furthermore, we only apply our approach to the limited families of copulas. In elliptical and skew elliptical families of copulas, skew t copulas are very useful in terms of financial modeling, a mixture extension of which can be studied. For the high dimensional application, we can also extend our approach to the vine copulas and its variant (Joe and Kurowicka, 2011), one of the most popular research topics regarding copulas. Some extensions in terms of application can also be considered. First, more application examples in different areas of science and engineering can be tested to verify the usefulness of our proposed approaches. More specifically, there are some fundamental applications that are pending to be done. For example, in financial risk management and insurance, Value-at-Risk (VaR) and condition Value-at-Risk (CVaR) are the two most popular risk measures. The combination of our proposed methods with the VaR/CVaR risk calculation can be trailed to test the effectiveness of our proposed copula models in terms of capturing complex financial risks. New applications, such as option pricing and credit modeling, can also be sought where copula methods are often considered.

Appendix A

Proof of Propostion 5

Proposition 5. *Let M be the fix sample size, $x \in (1, M/2) \cap Z^+$ be the number of minority class, Q_1 and Q_2 be defined as (5.5), $0.5 \leq A < 1$. Then, $v(x)$ is a decreasing function with respect to x.*

Proof. Let $w(x) = \log(v(x))$,

$$\begin{aligned} w(x) - w(x-1) &= \log\big(v(x)\big) - \log\big(v(x-1)\big) \\ &= \log(x-1) + \log(M-x+1) - \log(x) - \log(M-x) \\ &\quad + \log\left[\frac{A(1-A) + (x-1)(Q_1 - A^2) + (M-x-1)(Q_2 - A^2)}{A(1-A) + (x-2)(Q_1 - A^2) + (M-x)(Q_2 - A^2)}\right]. \end{aligned}$$

Since we have

$$\begin{aligned} &\log(x-1) + \log(M-x+1) - \log(x) - \log(M-x) \\ &= \log\left(\frac{-x^2 + (M+2)x - (M+1)}{-x^2 + Mx}\right), \end{aligned}$$

Meanwhile, as $2x < M$, we have

$$\big(-x^2 + (M+2)x - (M+1)\big) - (-x^2 + Mx) = 2x - (M+1) < 0.$$

Therefore,

$$\frac{\big(-x^2 + (M+2)x - (M+1)\big)}{(-x^2 + Mx)} < 1,$$

and

$$\log(\frac{-x^2+(M+2)x-(M+1)}{-x^2+Mx}) < 0.$$

Letting $q(x) = A(1-A) + (x-1)(Q_1 - A^2) + (M-x-1)(Q_2 - A^2)$, we arrive at

$$q(x) - q(x-1) = Q_1 - Q_2.$$

Substitute (5.5) and recall that $0.5 \leq A < 1$

$$Q_1 - Q_2 = \frac{A}{2-A} - \frac{2A^2}{1+A} = \frac{A(A-1)(2A-1)}{(2-A)(1+A)} \leq 0$$

This completes the proof. □

Bibliography

Alcalá-Fdez, J., Sanchez, L., Garcia, S., del Jesus, M. J., Ventura, S., Garrell, J. M., Otero, J., Romero, C., Bacardit, J., Rivas, V. M., et al. (2009). KEEL: a software tool to assess evolutionary algorithms for data mining problems. *Soft Computing*, 13(3):307–318.

Almeida, C. and Czado, C. (2012). Efficient Bayesian inference for stochastic time-varying copula models. *Computational Statistics and Data Analysis*, 56(6):1511–1527.

Ang, A. and Chen, J. (2002). Asymmetric correlations of equity portfolios. *Journal of Financial Economics*, 63(3):443–494.

Antoniak, C. E. (1974). Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *The Annals of Statistics*, 2(6):1152–1174.

Arakelian, V. and Karlis, D. (2014). Clustering dependencies via mixtures of copulas. *Communications in Statistics-Simulation and Computation*, 43(7):1644–1661.

Ardia, D. and Hoogerheide, L. F. (2014). GARCH models for daily stock returns: Impact of estimation frequency on Value-at-Risk and Expected Shortfall forecasts. *Economics Letters*, 123(2):187–190.

Azzalini, A. (1985). A class of distributions which includes the normal ones. *Scandinavian Journal of Statistics*, 12(2):171–178.

Azzalini, A. (2013). *The skew-normal and related families*. Cambridge University Press.

Azzalini, A. and Valle, A. D. (1996). The multivariate skew-normal distribution. *Biometrika*, 83(4):715–726.

Bamber, D. (1975). The area above the ordinal dominance graph and the area below the receiver operating characteristic graph. *Journal of Mathematical Psychology*, 12(4):387–415.

Beal, M., Ghahramani, Z., and Rasmussen, C. (2001). The infinite hidden Markov model. *Advances in Neural Information Processing Systems*, 14:577–584.

Bhattacharyya, S., Jha, S., Tharakunnel, K., and Westland, J. C. (2011). Data mining for credit card fraud: A comparative study. *Decision Support Systems*, 50(3):602–613.

Bishop, C. M. and Nasrabadi, N. M. (2006). *Pattern recognition and machine learning*. Springer, New York.

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. (2017). Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877.

Blume, J. D. (2009). Bounding sample size projections for the area under a ROC curve. *Journal of Statistical Planning and Inference*, 139(3):711–721.

Borak, S., Misiorek, A., and Weron, R. (2011). Models for heavy-tailed asset returns. In *Statistical tools for finance and insurance*. Springer.

Box, G. E., Luceño, A., and del Carmen Paniagua-Quinones, M. (2011). *Statistical control by monitoring and adjustment*. John Wiley & Sons.

Bradley, A. P. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159.

Burda, M. and Prokhorov, A. (2014). Copula based factorization in Bayesian multivariate infinite mixture models. *Journal of Multivariate Analysis*, 127:200–213.

Cai, Z. and Wang, X. (2014). Selection of mixed copula model via penalized likelihood. *Journal of the American Statistical Association*, 109(506):788–801.

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357.

Cherubini, U., Luciano, E., and Vecchiato, W. (2004). *Copula methods in finance*. John Wiley & Sons.

Cieslak, D. A., Chawla, N. V., and Striegel, A. (2006). Combating imbalance in network intrusion datasets. In *IEEE Int. Conf. Granular Comput.*

Cortez, P., Cerdeira, A., Almeida, F., Matos, T., and Reis, J. (2009). Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems*, 47(4):547–553.

Danaher, P. J. and Smith, M. S. (2011). Modeling multivariate distributions using copulas: Applications in marketing. *Marketing Science*, 30(1):4–21.

De Angelis, L. and Paas, L. J. (2013). A dynamic analysis of stock markets using a hidden Markov model. *Journal of Applied Statistics*, 40(8):1682–1700.

Deligiannidis, G. and Doucet, A. (2018). The correlated pseudomarginal method. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 80(5):839–870.

DeLong, E. R., DeLong, D. M., and Clarke-Pearson, D. L. (1988). Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach. *Biometrics*, 44(3):837–845.

Deng, L., Smith, M. S., and Maneesoonthorn, W. (2023). Efficient Variational Inference for Large Skew-t Copulas with Application to Intraday Equity Returns. *arXiv preprint arXiv:2308.05564*.

Derrode, S. and Pieczynski, W. (2016). Unsupervised classification using hidden markov chain with unknown noise copulas and margins. *Signal Processing*, 128:8–17.

Dias, J. G., Vermunt, J. K., and Ramos, S. (2015). Clustering financial time series: New insights from an extended hidden Markov model. *European Journal of Operational Research*, 243(3):852–864.

Dufays, A. (2016). Infinite-state Markov-switching for dynamic volatility. *Journal of Financial Econometrics*, 14(2):418–460.

Escobar, M. D. and West, M. (1995). Bayesian density estimation and inference using mixtures. *Journal of the American Statistical Association*, 90(430):577–588.

Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*, 96(456):1348–1360.

Faugeras, O. P. (2017). Inference for copula modeling of discrete data: a cautionary tale and some facts. *Dependence Modeling*, 5(1):121–132.

Feng, Z. D. and McCulloch, C. E. (1996). Using bootstrap likelihood ratios in finite mixture models. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(3):609–617.

Fergusson, K. and Platen, E. (2006). On the Distributional Characterization of Daily Log-Returns of a World Stock Index. *Applied Mathematical Finance*, 13(01):19–38.

Fernández, A., García, S., Galar, M., Prati, R. C., Krawczyk, B., and Herrera, F. (2018). *Learning from imbalanced data sets*. Springer.

Fox, E. B., Sudderth, E. B., Jordan, M. I., and Willsky, A. S. (2011). A sticky HDP-HMM with application to speaker diarization. *The Annals of Applied Statistics*, 5(2A):1020–1056.

Frank, M. J. (1979). On the simultaneous associativity of F(x,y) and x+y-F(x,y). *Aequationes mathematicae*, 19(1):194–226.

Frühwirth-Schnatter, S. and Malsiner-Walli, G. (2019). From here to infinity: sparse finite versus Dirichlet process mixtures in model-based clustering. *Advances in Data Analysis and Classification*, 13:33–64.

Frühwirth-Schnatter, S. and Pyne, S. (2010). Bayesian inference for finite mixtures of univariate and multivariate skew-normal and skew-t distributions. *Biostatistics*, 11(2):317–336.

Geenens, G. (2020). Copula modeling for discrete random vectors. *Dependence Modeling*, 8(1):417–440.

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2013). *Bayesian Data Analysis*. CRC Press, New York.

Gelman, A. and Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statistical Science*, 7(4):457–472.

Genest, C., Ghoudi, K., and Rivest, L.-P. (1995). A semiparametric estimation procedure of dependence parameters in multivariate families of distributions. *Biometrika*, 82(3):543–552.

Genest, C., Ghoudi, K., and Rivest, L.-P. (1998). "Understanding relationships using copulas," by Edward Frees and Emiliano Valdez, January 1998. *North American Actuarial Journal*, 2(3):143–149.

Genest, C. and Rivest, L.-P. (1993). Statistical inference procedures for bivariate Archimedean copulas. *Journal of the American statistical Association*, 88(423):1034–1043.

Geweke, J. (1991). Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments. Technical report, Federal Reserve Bank of Minneapolis.

Geweke, J., Koop, G., and van Dijk, H. K. (2011). *The Oxford handbook of Bayesian econometrics*. Oxford University Press, USA.

Gunawan, D., Tran, M.-N., Suzuki, K., Dick, J., and Kohn, R. (2019). Computationally efficient Bayesian estimation of high-dimensional Archimedean copulas with discrete and mixed margins. *Statistics and Computing*, 29(5):933–946.

Hand, D. J. (2009). Measuring classifier performance: a coherent alternative to the area under the ROC curve. *Machine Learning*, 77(1):103–123.

Hanley, J. A. and McNeil, B. J. (1982). The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, 143(1):29–36.

Hanley, J. A. and McNeil, B. J. (1983). A method of comparing the areas under receiver operating characteristic curves derived from the same cases. *Radiology*, 148(3):839–843.

He, H. and Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263–1284.

Hofert, M., Kojadinovic, I., Maechler, M., and Yan, J. (2022). *copula: Multivariate Dependence with Copulas*. R package version 1.1-0.

Holzmann, H., Munk, A., and Gneiting, T. (2006). Identifiability of finite mixtures of elliptical distributions. *Scandinavian Journal of Statistics*, 33(4):753–763.

Hu, L. (2006). Dependence patterns across financial markets: a mixed copula approach. *Applied Financial Economics*, 16(10):717–729.

Huard, D., Evin, G., and Favre, A.-C. (2006). Bayesian copula selection. *Computational Statistics and Data Analysis*, 51(2):809–822.

Ishwaran, H. and James, L. F. (2001). Gibbs sampling methods for stick-breaking priors. *Journal of the American Statistical Association*, 96(453):161–173.

Janes, H. and Pepe, M. (2006). The optimal ratio of cases to controls for estimating the classification accuracy of a biomarker. *Biostatistics*, 7(3):456–468.

Jaworski, P., Durante, F., Hardle, W. K., and Rychlik, T. (2010). *Copula theory and its applications*. Springer.

Jiryaie, F., Withanage, N., Wu, B., and De Leon, A. (2016). Gaussian copula distributions for mixed data, with application in discrimination. *Journal of Statistical Computation and Simulation*, 86(9):1643–1659.

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

Joe, H. (2005). Asymptotic efficiency of the two-stage estimation method for copula-based models. *Journal of multivariate Analysis*, 94(2):401–419.

Joe, H. and Kurowicka, D. (2011). *Dependence modeling: vine copula handbook*. World Scientific, London.

Joe, H. and Xu, J. J. (1996). The estimation method of inference functions for margins for multivariate models. *Technical Report*, 166.

Kalli, M., Griffin, J. E., and Walker, S. G. (2011). Slice sampling mixture models. *Statistics and Computing*, 21(1):93–105.

Kass, R. E. and Raftery, A. E. (1995). Bayes factors. *Journal of the American Statistical Association*, 90(430):773–795.

Kingma, D. P. and Welling, M. (2013). Auto-encoding variational Bayes. *arXiv preprint arXiv:1312.6114*.

Kleinbaum, D. G., Dietz, K., Gail, M., Klein, M., and Klein, M. (2002). *Logistic regression*. Springer.

Köppen, M. (2000). The curse of dimensionality. In *5th online world conference on soft computing in industrial applications (WSC5)*.

Kosmidis, I. and Karlis, D. (2016). Model-based clustering using copulas with applications. *Statistics and Computing*, 26:1079–1099.

Krzanowski, W. J. and Hand, D. J. (2009). *ROC curves for continuous data*. CRC Press.

Kuha, J. (2004). AIC and BIC: Comparisons of assumptions and performance. *Sociological Methods and Research*, 33(2):188–229.

Li, Y. (2020). *Addressing class imbalance for logistic regression*. PhD thesis, Imperial College London, London, UK.

Liu, G., Long, W., Yang, B., and Cai, Z. (2022). Semiparametric estimation and model selection for conditional mixture copula models. *Scandinavian Journal of Statistics*, 49(1):287–330.

Liu, G., Long, W., Zhang, X., and Li, Q. (2019). Detecting financial data dependence structure by averaging mixture copulas. *Econometric Theory*, 35(4):777–815.

Loaiza-Maya, R. and Smith, M. S. (2019). Variational Bayes estimation of discrete-margined copula models with application to time series. *Journal of Computational and Graphical Statistics*, 28(3):523–539.

Maheu, J. M. and Yang, Q. (2016). An infinite hidden Markov model for short-term interest rates. *Journal of Empirical Finance*, 38:202–220.

Malsiner-Walli, G., Frühwirth-Schnatter, S., and Grün, B. (2016). Model-based clustering based on sparse finite Gaussian mixtures. *Statistics and Computing*, 26(1-2):303–324.

Mazo, G. and Averyanov, Y. (2019). Constraining kernel estimators in semiparametric copula mixture models. *Computational Statistics and Data Analysis*, 138:170–189.

McAuliffe, J. D., Blei, D. M., and Jordan, M. I. (2006). Nonparametric empirical Bayes for the Dirichlet process mixture model. *Statistics and Computing*, 16(1):5–14.

McClish, D. K. (1989). Analyzing a portion of the ROC curve. *Medical Decision Making*, 9(3):190–195.

McLachlan, G. J., Lee, S. X., and Rathnayake, S. I. (2019). Finite mixture models. *Annual Review of Statistics and Its Application*, 6:355–378.

McNeil, A. J., Frey, R., and Embrechts, P. (2015). *Quantitative risk management: concepts, techniques and tools-revised edition*. Princeton University Press.

McNeil, B. J. and Hanley, J. A. (1984). Statistical approaches to the analysis of receiver operating characteristic (ROC) curves. *Medical Decision Making*, 4(2):137–150.

Mena, L. J. and Gonzalez, J. A. (2006). Machine Learning for Imbalanced Datasets: Application in Medical Diagnostic. In *Flairs Conference*.

Meyer, C. (2013). The bivariate normal copula. *Communications in Statistics-Theory and Methods*, 42(13):2402–2422.

Mohammed, R., Rawashdeh, J., and Abdullah, M. (2020). Machine learning with oversampling and undersampling techniques: overview study and experimental results. In *2020 11th international conference on information and communication systems (ICICS)*. IEEE.

Murphy, K. P. (2023). *"Probabilistic Machine Learning: Advanced Topics"*. MIT Press.

Muteba Mwamba, J. W. and Angaman, E. S. E. F. (2021). Modeling System Risk in the South African Insurance Sector: A Dynamic Mixture Copula Approach. *International Journal of Financial Studies*, 9(2):29.

Nasr, B. R. and Remillard, B. N. (2023). Identifiability and inference for copula-based semiparametric models for random vectors with arbitrary marginal distributions. *arXiv preprint arXiv:2301.13408*.

Nelsen, R. B. (2006). *An introduction to copulas*. Springer Science and Business Media.

Nguyen, N. (2018). Hidden Markov model for stock trading. *International Journal of Financial Studies*, 6(2):36.

Nolan, J. P. (2014). Financial modeling with heavy-tailed stable distributions. *Wiley Interdisciplinary Reviews: Computational Statistics*, 6(1):45–55.

Obuchowski, N. A. and McCLISH, D. K. (1997). Sample size determination for diagnostic accuracy studies involving binormal ROC curve indices. *Statistics in Medicine*, 16(13):1529–1542.

Oflaz, Z., Yozgatligil, C., and Selcuk-Kestel, A. S. (2023). Modeling comorbidity of chronic diseases using coupled hidden Markov model with bivariate discrete copula. *Statistical Methods in Medical Research*, 32(4):829–849.

Otiniano, C., Rathie, P., and Ozelim, L. (2015). On the identifiability of finite mixture of skew-normal and skew-t distributions. *Statistics and Probability Letters*, 106:103–108.

Ötting, M. and Karlis, D. (2022). Football tracking data: a copula-based hidden markov model for classification of tactics in football. *Annals of Operations Research*, 325:1–17.

Ötting, M., Langrock, R., and Maruotti, A. (2021). A copula-based multivariate hidden Markov model for modelling momentum in football. *AStA Advances in Statistical Analysis*, 107:1–19.

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

Papaspiliopoulos, O. and Roberts, G. O. (2008). Retrospective Markov chain Monte Carlo methods for Dirichlet process hierarchical models. *Biometrika*, 95(1):169–186.

Patton, A. J. (2012). A review of copula models for economic time series. *Journal of Multivariate Analysis*, 110:4–18.

Pepe, M. S. et al. (2003). *The statistical evaluation of medical tests for classification and prediction*. Oxford University Press.

Pitt, M., Chan, D., and Kohn, R. (2006). Efficient Bayesian inference for Gaussian copula regression models. *Biometrika*, 93(3):537–554.

Qin, G. and Hotilovac, L. (2008). Comparison of non-parametric confidence intervals for the area under the ROC curve of a continuous-scale diagnostic test. *Statistical Methods in Medical Research*, 17(2):207–221.

Rabiner, L. and Juang, B. (1986). An introduction to hidden Markov models. *IEEE ASSP Magazine*, 3(1):4–16.

Ren, J., Lee, S. D., Chen, X., Kao, B., Cheng, R., and Cheung, D. (2009). Naive bayes classification of uncertain data. In *2009 Ninth IEEE International Conference on Data Mining*, pages 944–949. IEEE.

Rezaei, M., Yang, H., and Meinel, C. (2020). Recurrent generative adversarial network for learning imbalanced medical image semantic segmentation. *Multimedia Tools and Applications*, 79(21):15329–15348.

Rousseau, J. and Mengersen, K. (2011). Asymptotic behaviour of the posterior distribution in overfitted mixture models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(5):689–710.

Roy, A. and Parui, S. K. (2014). Pair-copula based mixture models and their application in clustering. *Pattern Recognition*, 47(4):1689–1697.

Ruby, U. and Yendapalli, V. (2020). Binary cross entropy with deep learning technique for image classification. *Int. J. Adv. Trends Comput. Sci. Eng*, 9(10).

Sethuraman, J. (1994). A constructive definition of Dirichlet priors. *Statistica Sinica*, pages 639–650.

Silva, R. d. S. and Lopes, H. F. (2008). Copula, marginal distributions and model selection: a Bayesian note. *Statistics and Computing*, 18(3):313–320.

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

Sklar, A. (1973). Random variables, joint distribution functions, and copulas. *Kybernetika*, 9(6):449–460.

Smith, M. S. (2011). Bayesian approaches to copula modelling. *arXiv preprint arXiv:1112.4204*.

Smith, M. S. (2023). Implicit copulas: An overview. *Econometrics and Statistics*, 28:81–104.

Smith, M. S., Gan, Q., and Kohn, R. J. (2012). Modelling dependence using skew t copulas: Bayesian inference and applications. *Journal of Applied Econometrics*, 27(3):500–522.

Smith, M. S. and Khaled, M. A. (2012). Estimation of copula models with discrete margins via Bayesian data augmentation. *Journal of the American Statistical Association*, 107(497):290–303.

Smith, M. S. and Klein, N. (2021). Bayesian inference for regression copulas. *Journal of Business and Economic Statistics*, 39(3):712–728.

Smith, M. S. and Loaiza-Maya, R. (2022). Implicit copula variational inference. *Journal of Computational and Graphical Statistics*, 32(3):769–781.

Spiegelhalter, D. J., Best, N. G., Carlin, B. P., and Linde, A. (2014). The deviance information criterion: 12 years on. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 76(3):485–493.

Streiner, D. L. and Cairney, J. (2007). What's under the ROC? An introduction to receiver operating characteristics curves. *The Canadian Journal of Psychiatry*, 52(2):121–128.

Sun, L.-H., Lee, C.-S., and Emura, T. (2020). A Bayesian inference for time series via copula-based Markov chain models. *Communications in Statistics-Simulation and Computation*, 49(11):2897–2913.

Teh, Y. W., Jordan, M. I., Beal, M. J., and Blei, D. M. (2006). Hierarchical Dirichlet Processes. *Journal of the American Statistical Association*, 101(476):1566–1581.

Teicher, H. (1961). Identifiability of mixtures. *The Annals of Mathematical statistics*, 32(1):244–248.

Teicher, H. (1963). Identifiability of finite mixtures. *The Annals of Mathematical statistics*, pages 1265–1269.

Tekumalla, L. S. and Bhattacharyya, C. (2016). Copula-HDP-HMM: Non-parametric Modeling of Temporal Multivariate Data for I/O Efficient Bulk Cache Preloading. In *Proceedings of the 2016 SIAM International Conference on Data Mining*. SIAM.

Tierney, L. (1994). Markov chains for exploring posterior distributions. *the Annals of Statistics*, 22(4):1701–1728.

Van Gael, J. (2012). *Bayesian nonparametric hidden Markov models*. PhD thesis, University of Cambridge.

Van Gael, J., Saatci, Y., Teh, Y. W., and Ghahramani, Z. (2008). Beam sampling for the infinite hidden Markov model. In *Proceedings of the 25th international conference on Machine learning*.

Van Havre, Z., White, N., Rousseau, J., and Mengersen, K. (2015). Overfitting Bayesian mixture models with an unknown number of components. *PloS one*, 10(7):e0131739.

Van Tran, Q. and Kukal, J. (2022). A novel heavy tail distribution of logarithmic returns of cryptocurrencies. *Finance Research Letters*, 47:102574.

Vrac, M., Billard, L., Diday, E., and Chédin, A. (2012). Copula analysis of mixture models. *Computational Statistics*, 27(3):427–457.

Wagenmakers, E.-J. and Farrell, S. (2004). AIC model selection using Akaike weights. *Psychonomic Bulletin and Review*, 11:192–196.

Walker, S. G. (2007). Sampling the Dirichlet mixture model with slices. *Communications in Statistics - Simulation and Computation*, 36(1):45–54.

Wang, B. X. and Japkowicz, N. (2004). Imbalanced data set learning with synthetic samples. In *Proc. IRIS machine learning workshop*.

Wang, X. (2008). *Selection of mixed copulas and finite mixture models with applications in finance*. PhD thesis, The University of North Carolina at Charlotte.

Wei, X. and Li, C. (2012). The infinite Student's t-mixture for robust modeling. *Signal Processing*, 92(1):224–234.

Wei, Z., Kim, S., Choi, B., and Kim, D. (2019). Multivariate skew normal copula for asymmetric dependence: estimation and application. *International Journal of Information Technology and Decision Making*, 18(01):365–387.

Wu, J., Wang, X., and Walker, S. G. (2014). Bayesian nonparametric inference for a multivariate copula function. *Methodology and Computing in Applied Probability*, 16(3):747–763.

Wu, J., Wang, X., and Walker, S. G. (2015). Bayesian nonparametric estimation of a copula. *Journal of Statistical Computation and Simulation*, 85(1):103–116.

Xue, Y., Li, G., Li, Z., Wang, P., Gong, H., and Kong, F. (2022). Intelligent prediction of rockburst based on Copula-MC oversampling architecture. *Bulletin of Engineering Geology and the Environment*, 81(5):1–14.

Yakowitz, S. J. and Spragins, J. D. (1968). On the identifiability of finite mixtures. *The Annals of Mathematical Statistics*, 39(1):209–214.

Yu, H. (2017). A novel semiparametric hidden Markov model for process failure mode identification. *IEEE Transactions on Automation Science and Engineering*, 15(2):506–518.

Zhang, M.-L., Peña, J. M., and Robles, V. (2009). Feature selection for multi-label naive Bayes classification. *Information Sciences*, 179(19):3218–3229.

Zhu, Q., Wang, S., Chen, Z., He, Y., and Xu, Y. (2019). A virtual sample generation method based on kernel density estimation and copula function for imbalanced classification. In *2019 IEEE 8th Data Driven Control and Learning Systems Conference (DDCLS)*. IEEE.

Zi-Yi, G. (2017). Heavy-tailed distributions and risk management of equity market tail events. *Journal of Risk and Control*, 4(1):31–41.

Zimmerman, R., Craiu, R. V., and Leos-Barajas, V. (2022). Copula Modelling of Serially Correlated Multivariate Data with Hidden Structures. *arXiv preprint arXiv:2207.04127*.

Zou, K. H. and Hall, W. (2000). Two transformation models for estimating an ROC curve derived from continuous data. *Journal of Applied Statistics*, 27(5):621–631.

Zou, K. H., Hall, W., and Shapiro, D. E. (1997). Smooth non-parametric receiver operating characteristic (ROC) curves for continuous diagnostic tests. *Statistics in Medicine*, 16(19):2143–2156.