UMEÅ UNIVERSITY

Johans Thesis

Subtitle of Thesis

Johan Herbert[†]

Examination Date: April 11, 2025

Supervisors	Kevin Kamm	Umeå University
	Victor Jonsson	Vermiculus Financial Technology
Examiner	Christian Ewald	Umeå University

Master thesis, 30 ECTS
M.Sc in Industrial Engineering and Management, 300 ECTS
Spring term 2025

[†]johe6227@student.umu.se

Abstract

This is the english abstract.

Sammanfattning

Det är svenska sammanfattning.

Contents

1 Introduction

This section introduces the company and its activities Section 1.1, the topic of clearing, and the need for risk measures is explained together with its relation to dependency modeling Section 1.2. An introduction to copulas is given **??**. Finally, it motivates this project and specifies its purpose Section 1.5.

1.1 About Vermiculus Financial Technology

Vermiculus Financial Technology is a software company that builds systems for financial transactions. Its three main areas of operations are trading systems, clearing systems, and Central Securities Depository (CSD) systems. Trading systems match buy and sell orders in the market and find the price at which trades should be executed. Clearing systems reduce counterparty risk in financial transactions by acting as the central hub through which all transactions flow. CSD systems keep track of who owns which stocks on an exchange.

This project will be related to the clearing section of Vermiculus activities.

1.2 Background

Clearing systems are in place to minimize counterparty risk in financial transactions[1]. Counterparty risk is the risk of having the other party in a transaction not fulfilling its obligations in a deal and hence defaulting on its obligations[2]. The clearing house acts as a middleman in all transactions, selling to all buyers and buying to all sellers[3]. Hence, each party only faces the clearing house as their counterparty, removing the counterparty risk, this has been nicely illustrated[4]. In the right part of Figure 1 we can see the clearing members marked B, usually banks, that only face the clearing house marked CCP. In this system, as long as the clearing house does not go bankrupt, trades can go through even if a clearing member defaults on their commitment. The clearinghouse does not offer this risk removal for free; rather, it requires each party to post collateral covering the costs if a clearing member defaults. The clearinghouse makes money from fees and transaction costs, making it worthwhile.

The alternative to centrally cleared trading is that each market participant trades directly with each other. In this case, both sides of a transaction are exposed to counterparty risk

[1] See https://www.riksbank.se/en-gb/financial-stability/the-financial-system/the-financial-infrastructure/systems-in-the-financial-infrastructure/. Last Accessed: 2025-01-29

[2] See https://www.occ.treas.gov/topics/supervision-and-examination/capital-markets/financial-markets/counterparty-risk/index-counterparty-risk.html. Last Accessed: 2025-01-29

[3] See https://www.investopedia.com/terms/c/clearinghouse.asp

[4] See https://analystprep.com/study-notes/frm/part-1/financial-markets-and-products/central-clearing/. Last Accessed: 2025-01-29

from when making a trade until it has been settled. The risk in this scenario is that the buyer does not have enough money to pay or that the seller does not have the asset it has agreed to sell. This is nicely illustrated[4] in the left part of Figure 1. We can see that each clearing member, marked B, faces each other if having the buy and sell side of the same position. In this system, if one of the clearing members defaults, it can impact the other clearing members, who will not get their money. This can cause contagion so if one clearing member goes bankrupt, others may follow suit.

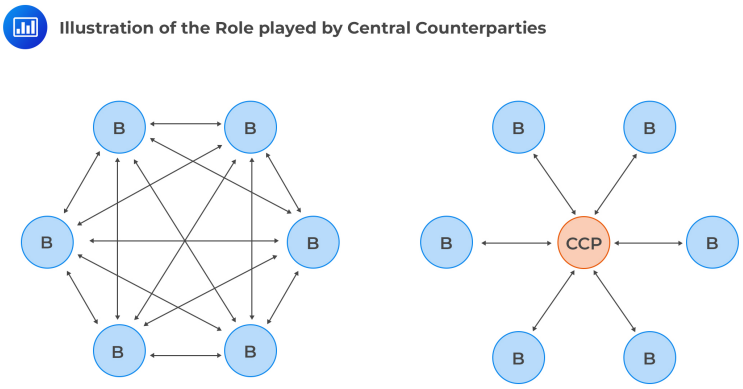


Figure 1: Bilateral and central counterparty clearing compared. The left picture shows a system of parties that are clearing transactions bilaterally. The right picture shows a system where central counterparty clearing is used. This illustration was found at AnalystPrep[4].

To determine how much collateral is required from each clearing member the risk of its portfolio must be measured. Historically this has been done using a framework called Standard Portfolio Analysis of Risk (SPAN) that was developed by Chicago Mercantile Exchange[5]. The SPAN framework has several scenarios based on changes in price, volatility, and time to maturity for derivatives. These scenarios are then used to calculate what change in value they would have for some security. Since the global financial crisis in 2008, more focus has been put on risk management. In later years a shift towards using Value at Risk (VaR) has taken place[6]. VaR is a measure defined as the maximum expected loss in portfolio value over a time horizon for some level of confidence[7]. There are several ways of calculating value at risk VaR such as historical, parametric, and Monte Carlo (MC) simulated among others [8].

Historical value at risk uses historical returns from the different assets in it to calculate the return of the portfolio at each time step. These portfolio returns are then ordered and the percentile corresponding to the confidence level is calculated giving the VaR. This method has advantages and disadvantages. The advantage is that the dependence between assets is

[5]See https://www.cmegroup.com/solutions/risk-management/performance-bonds-margins/span-methodology-overview.html#how-it-works. Last Accessed: 2025-01-29
[6]See https://www.fia.org/marketvoice/articles/navigating-new-era-derivatives-clearing
[7]See https://www.risk.net/definition/value-at-risk-var. Last Accessed: 2025-01-29
[8]See https://corporatefinanceinstitute.com/resources/career-map/sell-side/risk-management/value-at-risk-var/

completely realistic as it comes from the true observations on the market[8]. A disadvantage is however that this model can be insufficient because the observations from the data are limited. This might sound strange but when observations falling in the tails of a distribution is of course rare. This means that for VaR, which focuses on the lower tail, the amount of data available might not be enough to precisely determine the VaR.

Parametric VaR assumes that the returns for an asset are generated from some distribution. These returns are then used for calculating the described characteristics of the distribution. Usually, the Gaussian distribution is used in which case the mean vector and covariance matrix are estimated. This distribution is then used to calculate the VaR so that the probability of ending up below the limit aligns with the confidence[8]. This method has the advantage that it is simple. It has a disadvantage in that it requires the data to follow some known distribution which might not always be a realistic assumption.

MC VaR utilizes simulated random numbers to generate artificial return scenarios from which the VaR can be calculated[8]. MC simulations have a major advantage over the other methods mentioned in that they can be used for calculating the risks of different types of financial instruments. This is done by simulating plausible scenarios for the underlying assets of a portfolio.

The returns need to be simulated from distributions with dependence that reflect how different assets move in relation to one another. There are multiple methods to do this, such as using covariance matrices or copulas. Covariance matrices can be used with different correlation measures such as Pearson, Spearman, and Kendal. Spearman and Kendal are both rank correlation measures, Alexander (2008, pp. 256-258). In addition, there are also different types of copulas to use, meaning that there are many alternative methods to choose from. This leads us to the purpose of this project, but first, copulas need to be defined.

Remove about correlation measures

example correlation insufficient (Nonlinearity, ellipticity)

1.3 literature review

1.4 historical overview

on how it links to mathematical finance (credit risk and basket options), what did neural copula do?

1.5 Purpose

The purpose of this project is to investigate different methods of modeling dependence and to provide clarity about which method to use for what type of data. While doing this, it is reasonable to expect that shortcomings and risks of different methods will be discovered. In that case, potential shortcomings and risks will be reported to provide an understanding of when different methods can be used.

contributio what is

Problem, what is goal when use copula and neural copula and correlation

Specify more

1.6 Assignment description

This thesis will investigate how different methods of modeling dependence between different assets perform for different types of dependency structures. It will also investigate how machine learning can be used as a substitute or complement to better model dependence.

We will first fit a Neural Network (NN) as a copula function that captures the dependence between different random variables. The idea is that the copula function can be used for generating new random numbers, with a similar dependency structure to what is observed in the market.

The following questions will be studied in this project:

Aim of the thesis.

RQ1 Is the copula invariant to the marginal distributions?
RQ2 How should the neural copula function be trained to obtain consistently reliable results?
RQ3 Can a neural copula be used to better model the dependence between assets than other copulas?

Change these

1.7 Limitations

To define the scope of this project some limitations will be set initially. In some areas, the constraints are best defined after or during the literature study when more knowledge is obtained. The limitations

1. A selection of traditional measures to compare the Neural Copula (NC) to will be made.
2. The number of portfolios will be limited by only considering portfolios with two assets. This will both serve for interpretability and limit the number of portfolios to evaluate.
3. The data used will be artificially simulated to control the dependence between the assets in the portfolios and to be able to evaluate each method under consistent conditions.
4. The main focus of this report is to implement and evaluate the NC. When this is done, other measures will be considered.

These are the initial limitations set for the project. More limitations might be added and the ones defined here might be adjusted.

change these

outline of the rest of thesis

2 Theory

2.1 Mathematical finance

In mathematical finance, one is typically interested in the returns of financial assets, over discrete time steps, rather than their price Danielsson (2011, p. 2). Let P_{t_i} denote the price at time increment t_i, where t_i is usually in daily time increments, but can be any unit of time.

2.1.1 Returns

This section will introduce different types of returns, simple and log, as well as their different properties. These properties are for example return aggregation over time, portfolio return calculations, generation of new prices, and the bounds for returns. The reason for introducing different types of returns is to show the connections between financial returns and statistical distributions that in turn connect to the theory of copulas.

Firstly we should introduce what financial returns on assets are. The return of a financial asset is the relative price change over a given time interval, often expressed as a percentage Danielsson (2011, p. 2). One type of return is simple returns, defined in Definition 2.1.

Definition 2.1. Simple-Returns Danielsson (2011, p. 3) A *simple return* is the percentage change in price, over time period t_i, indicated by R_{t_i}:

$$R_{t_i} = \frac{P_{t_i} - P_{t_{i-1}}}{P_{t_{i-1}}}.$$

To aggregate several simple (daily) returns over some time (week) to the return over the whole period (week), one has to construct several change factors that are multiplied together before subtracting one. This aggregation process is described in Danielsson (2011, p. 3), where the aggregated return over n periods can be calculated as

$$R_{t_i}(n) = (1 + R_{t_i})(1 + R_{t_{i-1}})(1 + R_{t_{i-2}}) \dots (1 + R_{t_{i-n+1}}) - 1 = \frac{P_{t_i}}{P_{t_{i-n}}} - 1.$$

In many situations, it is necessary to calculate the return of an entire portfolio of assets from its underlying asset returns. For simple returns, this is simply done by calculating a weighted average of the individual asset returns. Danielsson (2011, p. 3) describes the calculation of the portfolio return $R_{t_i,\text{port}}$ for a portfolio with K different assets as

$$R_{t,\text{port}} = \sum_{k=1}^{K} w_k R_{t,k}.$$

Many times, it can be useful to simulate new stock prices by using randomly generated returns. For simple returns, new prices can be calculated as

$$P_{t_i} = P_{t_{i-1}}(1 + R_{t_i}).$$

The space in which realizations of returns can be observed differs for different types of returns. In Example 2.2 the range for simple returns is derived.

Example 2.2. To see the bounds for simple returns, we investigate what happens as the stock price moves to zero, corresponding to bankruptcy, and when the stock price moves to infinity. We denote the simple return when the price goes to zero and infinity by R_t^- and R_t^+ respectively

$$R_{t_i}^- = \lim_{P_{t_i} \to 0} \frac{P_{t_i} - P_{t_{i-1}}}{P_{t_{i-1}}} = \frac{0 - P_{t_{i-1}}}{P_{t_{i-1}}} = -1$$
$$R_{t_i}^+ = \lim_{P_{t_i} \to \infty} \frac{P_{t_i} - P_{t_{i-1}}}{P_{t_{i-1}}} = \frac{\infty - P_{t_{i-1}}}{P_{t_{i-1}}} = \infty.$$

Hence, we conclude that $R_t \in (R_t^-, R_t^+) = [-1, \infty)$.

Continuously compounded returns or so-called logarithmic returns are often used for financial modeling given their desirable properties. A desirable property is that the returns are *symmetric*, so positive and negative returns of the same magnitude cancel each other out Danielsson (2011, p. 4). Log returns are defined, as done by Danielsson (2011, p. 3), in Definition 2.3.

Definition 2.3. Log-Returns
The logarithm of gross return or *log-returns*, indicated by Y_{t_i}:

$$Y_{t_i} = \log(1 + R_{t_i}) = \log\left(\frac{P_{t_i}}{P_{t_{i-1}}}\right) = \log(P_{t_i}) - \log(P_{t_{i-1}})$$

Log-returns have the advantage that the multiperiod, n period, returns are just the sum of one period returns Danielsson (2011, p. 3), that is

$$Y_{t_i}(n) = Y_{t_i} + Y_{t_{i-1}} + \cdots + Y_{t_{i-n+1}}.$$

As illustrated by Danielsson (2011, p. 4), the portfolio return is more complicated to com-

pute for log returns because the relation is not simply a weighted sum

$$Y_{t_i,\text{port}} = \log\left(\frac{P_{t_i,\text{port}}}{P_{t_{i-1},\text{port}}}\right) \neq \sum_{k=1}^{K} w_k \log\left(\frac{P_{t_i,k}}{P_{t_{i-1},k}}\right), \text{where}$$
$$P_{t_i,\text{port}} = \sum_{k=1}^{K} w_k P_{t_i,k}.$$

A weighted average is approximately, but not quite correctly, describing the portfolio returns in terms of the individual returns as described by Danielsson (2011, p. 3),

$$Y_{t_i,\text{port}} \approx \sum_{k=1}^{K} w_k R_{t_i,k}.$$

The correct relation between portfolio log returns and its sub-components log returns is given by first converting the log returns to prices, then calculating the portfolio returns before calculating the log return from the portfolio return

$$Y_{t_i,\text{port}} = \log\left(\frac{\sum_{k=1}^{K} w_k P_{t_i,k}}{\sum_{k=1}^{K} w_k P_{t_{i-1},k}}\right), \text{where } P_{t_i,k} = P_{t_i-1,k} e^{Y_{t_i,k}}.$$

In the above expression, the nominator represents the new portfolio value while the denominator represents the initial portfolio value. $K \in \mathbb{N}$ is the number of assets in the portfolio and w_k is the weight of the total portfolio in asset k. Note that the individual asset prices are updated, using the log return, by themselves before weighing them together as a portfolio value.

In many applications, one wants to simulate returns to produce artificial stock price developments. Log returns from stock prices usually seem to be generated from some sort of bell-shaped probability distribution. This makes it convenient becausecause one can use samples from some probability distribution as the log returns when simulating prices for an asset. As an example, the famous Black-Scholes model assumes that stock returns are log normally distributed. That is, the log returns are normally distributed.

To obtain the price P_{t_i} at the end of time period t_i using a simulated log return Y_{t_i} for period t_i, one can calculate

$$P_{t_i} = P_{t_{i-1}} e^{Y_{t_i}}.$$

The range in which log returns can be observed is, unlike those in the simple return case, unbounded both positively and negatively.

Example 2.4. To see this we investigate what happens to the returns when the stock price moves to zero, corresponding to bankruptcy, and when the stock price moves to infinity. We

denote the log return when the price goes to zero and infinity by Y_t^- and Y_t^+ respectively.

$$Y_{t_i}^- = \lim_{P_{t_i} \to 0} \log\left(\frac{P_{t_i}}{P_{t_{i-1}}}\right) = \log(0) = -\infty$$
$$Y_{t_i}^+ = \lim_{P_{t_i} \to \infty} \log\left(\frac{P_{t_i}}{P_{t_{i-1}}}\right) = \log(\infty) = \infty$$

Hence, we conclude that $Y_{t_i} \in (Y_{t_i}^-, Y_{t_i}^+) = (-\infty, \infty)$.

To better understand how the different types of returns work, Example 2.5 shows that simple returns are not symmetrical whereas log returns are.

Example 2.5. **Symmetry of returns** (own example, on the same lines as Danielsson (2011, p. 4) but not quite) Consider an example of a stock having initial price $P_0 = 100$, price after one day $P_1 = 200$, and price at day two $P_2 = 100$. Let's examine what these price changes do to simple and log returns respectively.

Simple returns
The return on the first day is

$$R_1 = \frac{P_1 - P_0}{P_0} = \frac{200 - 100}{100} = 1 = 100\%.$$

The return on the second day is

$$R_2 = \frac{P_2 - P_1}{P_1} = \frac{100 - 200}{200} = -\frac{1}{2} = -50\%.$$

Log returns
The return on the first day is

$$Y_1 = \log\left(\frac{P_1}{P_0}\right) = \log\left(\frac{200}{100}\right) = \log(2) \approx 69\%.$$

The return on the second day is

$$Y_2 = \log\left(\frac{P_2}{P_1}\right) = \log\left(\frac{100}{200}\right) = \log\left(\frac{1}{2}\right) \approx -69\%.$$

We can see that the log returns are symmetrical so that monetary gains and losses of equal magnitude have log returns of equal magnitude. This is in contrast to simple returns where gains and losses are not symmetrical.

To write about here

Argue for stocks being modeled using statistical distributions. Perhaps using GBM or a figure showing some stock returns.

Talk about how many applications use dependence between assets and why it is important. Also why it ties together with distributions.

How correlation is used, and when it breaks(using example)

- Value at risk
- Basket options

Talk about the use of Monte-Carlo methods for financial applications.

2.1.2 Monte Carlo methods

I would say

Monte Carlo methods is a blanket term for computational methods that utilize random numbers to model uncertain events. Given an assumption about the underlying distribution of a random process. The random process is simulated multiple times and used as the input in a deterministic function, the result of which is averaged.

Many times in different scientific fields, relationships between different variables can be described using deterministic relationships between input variables and output variables. In some situations, these deterministic relationships have random input parameters. In these cases, Monte Carlo methods can be useful.

Monte Carlo simulation is a method for simulating events that, due to some source of randomness, are uncertain. In Monte Carlo methods, a statistical distribution is identified for each source of randomness. The method is then to sample random numbers from these distributions to use as input in the deterministic relationship. This gives the outcome for multiple possible scenarios. The final step is to perform some statistical analysis of the generated output values from the functions. This can, for example, be used to calculate the mean, standard deviation, or percentile Raychaudhuri (2008, pp. 91-92).

Connect to mathematical finance.

2.1.3 Geometric Brownian motion

Geometric Brownian Motion (GBM) is a stochastic process that is used to model the evolution of stock prices. The GBM is defined by the stochastic differential equation (SDE)

$$dS_t = \mu S_t dt + \sigma S_t dW_t$$
$$S_0 = s_0,$$

where S_t is the stock price at time t, μ is the drift, σ is the volatility, and W_t is a Wiener process BJORK (2009, p. 67) . The GBM can be solved using the Itô formula, which gives the solution

Version Of Book

$$S_t = s_0 \exp\left(\left(\mu - \frac{\sigma^2}{2}\right)t + \sigma W_t\right),$$

BJORK (2009, p. 69).

2.1.4 Euler-Maruyama Scheme

To approximately simulate Stochastic Differential Equation (SDE)s the Euler-Maruyama scheme can be used. The Euler-Maruyama scheme is a numerical method for solving SDEs.

For an SDE of the form

$$dX_t = a(X_t)dt + b(X_t)dW_t,$$

the Euler-Maruyama scheme is defined as follows. Let $\hat{X}$ denote the approximate solution to the SDE and let X denote the exact solution. The Euler-Maruyama scheme is defined by the following recursive formula

$$\hat{X}(t_{i+1}) = \hat{X}(t_i) + a(\hat{X}(t_i))[t_{i+1} - t_i] + b(\hat{X}(t_i))\sqrt{t_{i+1} - t_i}Z_{i+1},$$

where Z_{i+1} is a standard normal random variable. The time step $t_{i+1} - t_i$ is the time increment, and a and b are the drift and diffusion functions respectively GLASSERMAN (2004, pp. 339-340).

We can use the Euler-Maruyama scheme to simulate a stock trajectory with the GBM by setting $a(S_t) = \mu S_t$ and $b(S_t) = \sigma S_t$. The resulting scheme is

$$\hat{S}(t_{i+1}) = \hat{S}(t_i) + \mu\hat{S}(t_i)[t_{i+1} - t_i] + \sigma\hat{S}(t_i)\sqrt{t_{i+1} - t_i}Z_{i+1}.$$

To simulate a stock trajectory, the time increment $t_{i+1} - t_i$ is set to a small value, and the process is iterated for a large number of steps.

We can simulate a pair of dependent stock trajectroies by simulating a system of GBMs

$$dS_t^d = \mu S_t^d dt + \sigma S_t^d dW_t^d,\ d = 1, 2,$$

where W_t^1 and W_t^2 are standard one dimensional Wiener processes with correlation ρ GLASSERMAN (2004, p. 104). One can also simulate a system of GBMs with a dependence different than correlation. This can be done by using a copula to generate the dependent Weiner processes.

Need reference for this?

2.2 Probability theory

As mentioned above, stock returns are often modeled using some statistical distribution. Statistical distributions are fundamental to the theory of copulas and therefore we need to define the terminology around distributions more formally. Throughout the upcoming sections, illustrations will be made to explain the ideas visually. Unless otherwise stated the Gaussian distribution, with mean 0 and standard deviation 1, will be used for these illustrations.

First, we need to define what statistical distributions are, beginning with Cumulative Distribution Function (CDF) in one dimension defined in Definition 2.6.

Maybe write in terms of random variable?

Definition 2.6. CDF one dimension Nelsen (2006, p. 17)
A *distribution function* or CDF is a function F with domain $\bar{R} = [-\infty, \infty]$ such that

1. F is nondecreasing;
2. $F(-\infty) = 0$ and $F(\infty) = 1$.

One can think of the CDF at each point on its domain $\bar{R}$ as the probability of a realization being below that point. Tied to the CDF is the Probability Density Function (PDF), which is defined in definition 2.7.

Definition 2.7. PDF one dimension Devore and Berk (2012, pp. 160-161)
Let X be a continuous random variable. Then a *probability distribution* or PDF of X is a function $f(x)$ such that for any two numbers a and b with $a \leq b$,

$$P(a \leq X \leq b) = \int_a^b f(x)dx.$$

Remark 2.8. Note that for a function to be a valid PDF, the following conditions must be satisfied.

$$f(x) \geq 0 \text{ for all } x;$$
$$\int_{-\infty}^{\infty} f(x)dx = 1.$$

These are implied by Definition 2.7 but are explicitly stated here for clarity.

Figure 2 illustrates both a PDF (left) and CDF (right) for a random variable X, being standard normally distributed. In the left figure, we can see a histogram of simulated data generated from the standard normal distribution as well as the theoretical normal distribution. The PDF shows the likelihood of a realization of the X to end up at each point in the domain

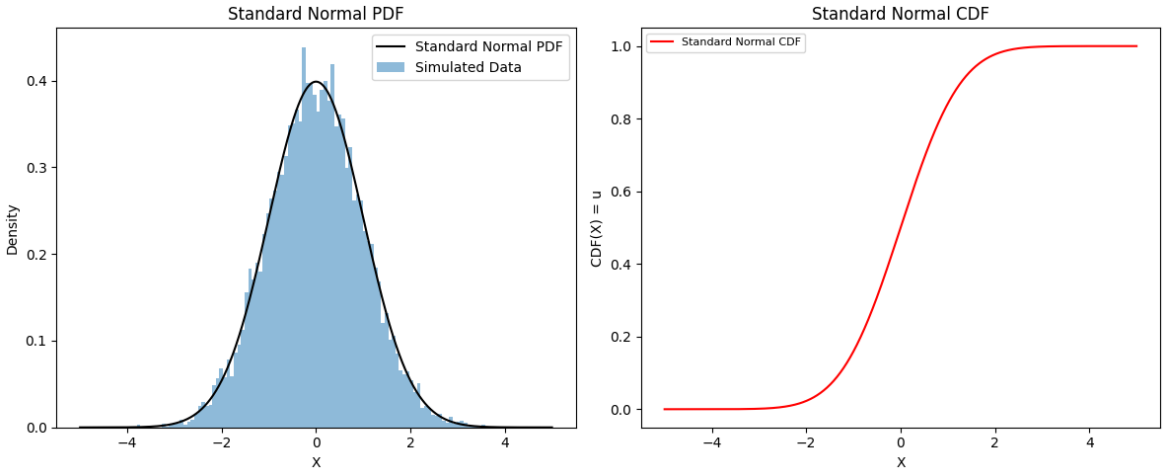


Figure 2: Illustration of a PDF (left) and CDF (right) for a random variable X, being standard normally distributed, in one dimension.

of X. The right picture shows the CDF corresponding to the PDF, its function value in each point is defined as the probability of ending up below that point on the domain of X.

When dealing with multiple assets the notion of a CDF generalizes to a multivariate CDF. To define a multivariate CDF formally in two dimensions we will need to define the H-volume of a function in two dimensions and what the meaning of a function being 2-increasing is. This is done in the same manner as in Nelsen (2006, p. 8) in Definition 2.9 and Definition 2.10.

Definition 2.9. H-Volume Nelsen (2006, p. 8)
Let u and v be nonempty subsets of $\bar{R} = [-\infty, \infty]$, and let H be a two-place real function such that Dom $H = u \times v$. Let $B = [u_1, v_2] \times [u_1, v_2]$ be a rectangle all of whose vertices are in Dom H. Then the *H-volume* of B is given by

$$V_H(B) = H(u_2, v_2) - H(u_2, v_1) - H(u_1, v_2) + H(u_1, v_1).$$

Definition 2.10. 2-increasing Nelsen (2006, p. 8)
A 2-place real function H is *2-increasing* if its H-volume $V_H(B) \geq 0$, for all rectangles B whose vertices lie in Dom H.

The notion of a 2-increasing function is illustrated in Figure 3. In the figure, the left picture shows a function that is 2-increasing meaning that it is increasing in both directions. The right picture illustrates when a function is not 2-increasing and how this will be captured by the H-volume of the function. This property is needed to define a CDF in several dimensions, as done by Nelsen (2006, p. 17), in Definition 2.11.

Definition 2.11. Joint CDF
A *joint distribution function* or joint CDF is a function F with domain $\bar{R}^2 = [-\infty, \infty]^2$ such

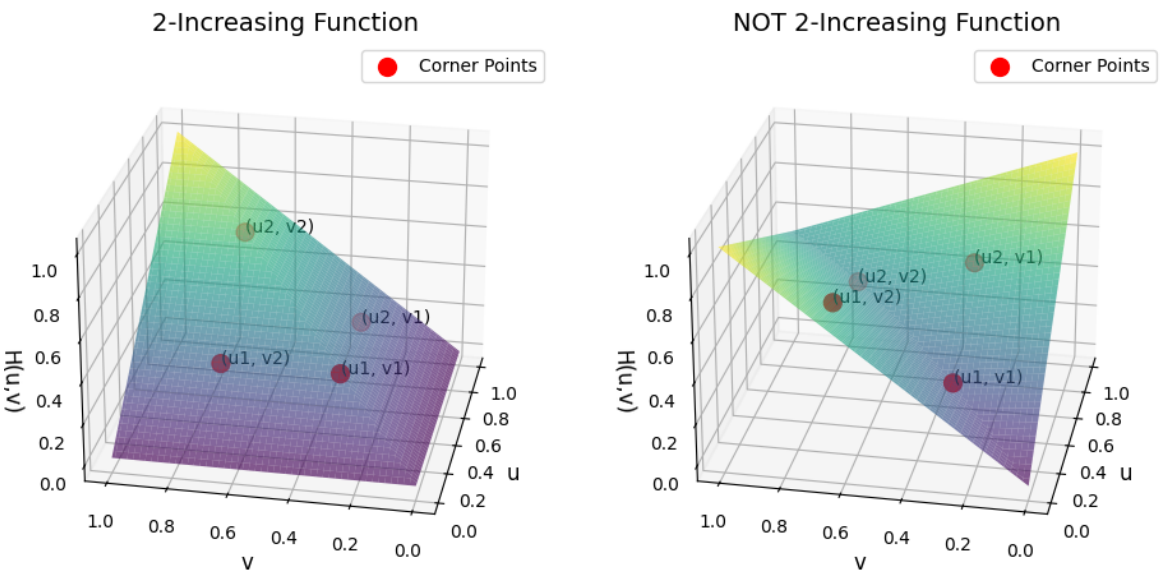


Figure 3: Illustration of what it means for a function to be, and not to be, 2-increasing using the definition of the H-volume.

that

1. F is 2-increasing;
2. $F(x_1, -\infty) = F(-\infty, x_2) = 0$, and $F(\infty, \infty) = 1$.

As in the univariate case the function value of the joint CDF at any point in $\text{Dom}X \times \text{Dom}Y$ $\bar{R}^2$ is the probability of being below that point. In the two-dimensional setting, it is the probability of a point ending up below the point in both dimensions simultaneously.

We can also define a PDF in two dimensions as done by Devore and Berk (2012, p. 235) in Definition 2.12.

Definition 2.12. Joint PDF
Let X and Y be continuous random variables. Then $f(x, y)$ is the *joint* PDF for X and Y if for any two-dimensional set A

$$P[(X, Y) \in A] = \iint_A f(x, y)dxdy.$$

In particular, if A is the two-dimensional rectangle $\{(x, y) : a \leq x \leq b, c \leq y \ \leq d\}$, then

$$P[(X, Y) \in A] = P(a \leq X \leq b, c \leq Y \leq d) = \int_a^b \int_c^d f(x, y)dxdy.$$

Remark 2.13. As seen in the univariate case, the same applies to the bivariate case. To be a

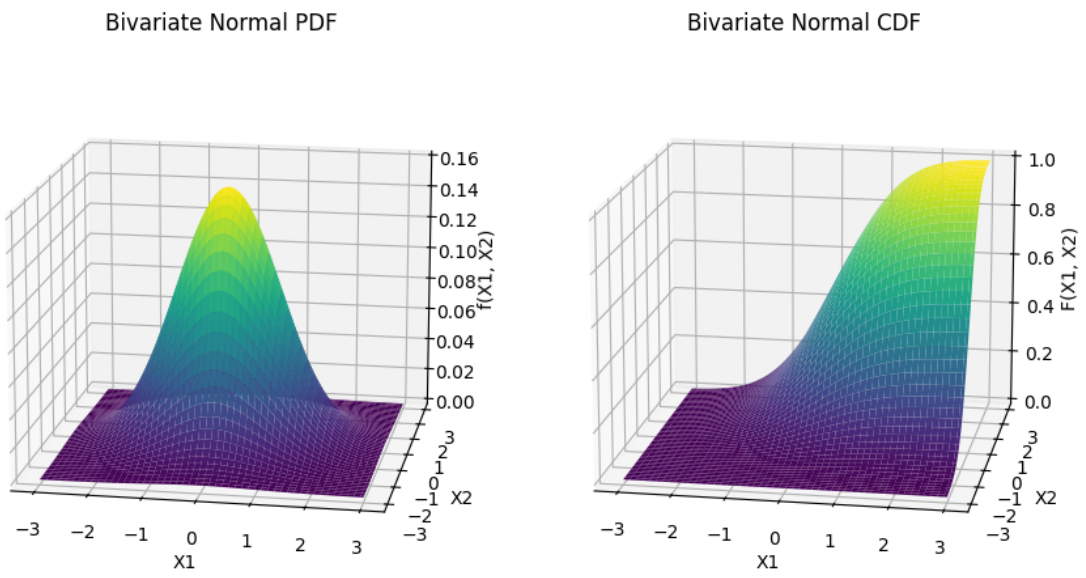


Figure 4: Illustration of a PDF (left) and CDF (right) for a two dimensional random variable $X = (X_1, X_2)$, being standard normally distributed.

candidate to be a joint PDF $f(x, y)$ must satisfy

$$f(x, y) \geq 0, \text{ and}$$
$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x, y)dxdy = 1.$$

Analogously to the univariate setting, the joint PDF and CDF are illustrated to give a visual understanding. In Figure 4 the left picture shows a joint PDF for the random variable $X = (X_1, X_2)$, being standard normally distributed with zero correlation. In the right picture, the joint CDF is displayed.

The Probability Integral Transform (PIT) refers to transforming a continuous random variable to a uniformly distributed random variable. This maps the domain of a continuous one-dimensional random variable through its CDF to the $[0, 1]$ space, which may in the sequel be referred to as *probability space*. This transformation will be central when defining copulas and is introduced in Theorem 2.14 as done in Danielsson (2011, p. 27).

Theorem 2.14. ***Probability integral transform***
Let X be a continuous random variable with distribution function F. Define a new random variable $U = F(X)$, then $U \sim \text{unif}(0, 1)$.

The probability integral transform is illustrated in Figure 5a. We can think of the PIT as a method of mapping all observed data points on Dom X to probability space. In the figure, this can be seen as the dotted lines representing data points being mapped.

Deeply connected to the PIT is the inverse transform method, which is a method of generating random numbers from a given distribution. The method is based on the PIT and works as defined by Glasserman (2004, p. 54).

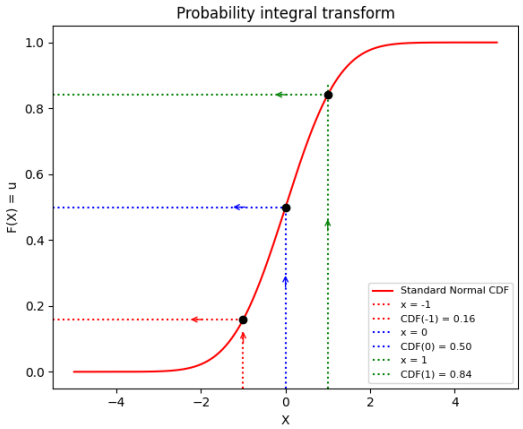
Definition 2.15. Inverse transform method

The *inverse transform method* is a method of generating random numbers from a given distribution.

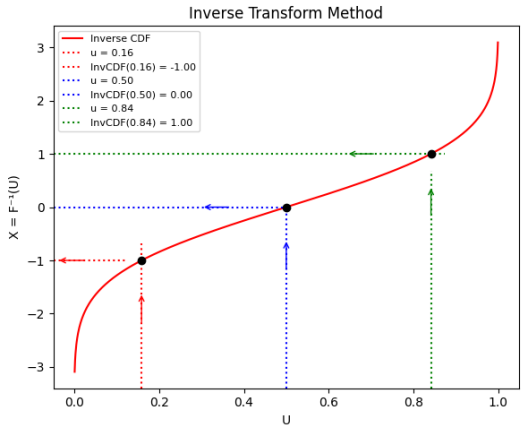
1. Generate random numbers $U \sim \text{Unif}(0, 1)$;
2. Insert the random numbers in the inverse CDF of the desired distribution such that $X = F^{-1}(U)$.

The random variable X will then be distributed according to the desired distribution. That is $P(X \leq x) = F(x)$ for all x.

The inverse transform method is illustrated in Figure 5b and can be seen as performing the PIT in reverse. In the figure the red line is the inverse CDF from which the sample is desired to be. The dotted lines represent the random numbers generated from the uniform distribution. The random numbers are then inserted into the inverse CDF to obtain the desired sample.



(a) Illustration of how the probability integral transform is used to transform realizations of a random variable X having distribution function F into a uniformly distributed random variable $U = F(X)$.



(b) Illustration of the inverse transform method and how it is used to generate random numbers from a given distribution given sampled points from a uniform distribution.

Figure 5: Probability Integral Transform and Inverse Transform Method.

Another important notion is that of a marginal distribution of a joint distribution, which can be thought of as the distribution if only considering one of the dimensions that the joint distribution is made up of. The marginal CDF and PDF are defined in Definition 2.16 and Definition 2.17 respectively, in the same way as in Evans and Rosenthal (2004, p. 81) and Wasserman (2010, p. 34) respectively.

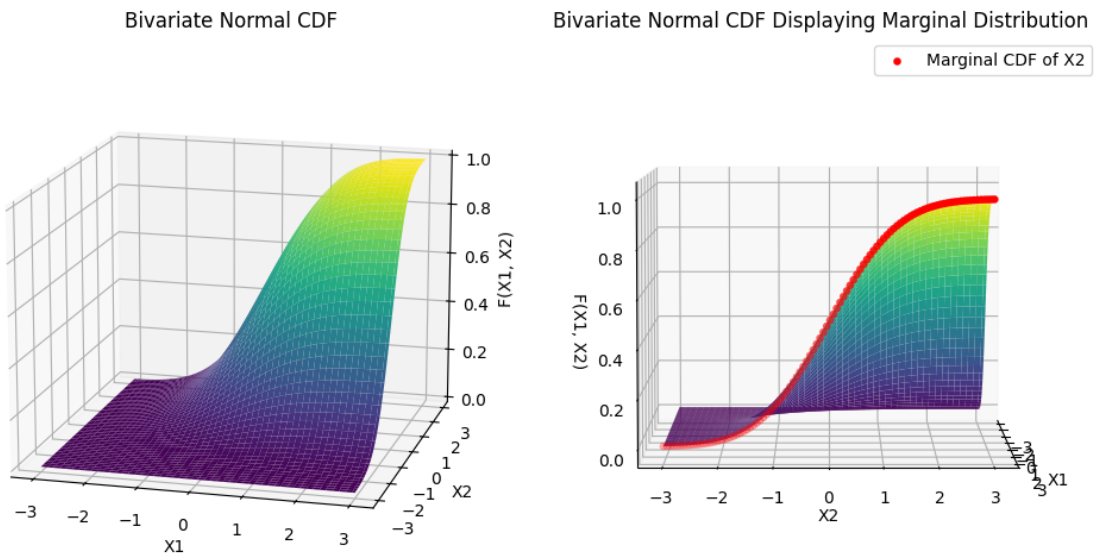


Figure 6: Illustration of what the marginal CDF looks like for a bivariate normal distribution.

Definition 2.16. Marginal CDF Let $X$and Y be two random variables having joint CDF $F_{X,Y}$, then the CDF F_X of X can be obtained from $F_{X,Y}$ because

$$F_X(x) = P(X \leq x) = P(X \leq x, Y \leq \infty) = \lim_{y \to \infty} F_{X,Y}(x, y).$$

F_X is called the *marginal distribution function* or marginal CDF of X. The marginal distribution F_Y can be obtained similarly.

Definition 2.17. Marginal PDF For a continuous random variable, with domain Dom$X \times$ Dom$Y$ $f_{X,Y}$, having joint PDF $f_{X,Y}$, the *marginal density function* or marginal PDF of X is given by

$$f_X(x) = \int_{-\infty}^{\infty} f_{X,Y}(x, y)dy.$$

The marginal density f_Y can be obtained similarly.

In Figure 6 the notion of a marginal distribution is visualized. In the left figure, a joint CDF is shown to connect to the prior explanation of a CDF. If the distribution is rotated to show it straight from one direction, as shown in the right figure, we can see the marginal distribution. In this case, it is the marginal distribution of X_2 that is displayed in red. This figure is of course simplified as it would not be plausible to view the limit when $X_1 \to \infty$.

To conclude this subsection, we will show how the PIT is utilized to transform data to the probability space and explain how it sets the stage for copulas. Section 2.2 illustrates how the PIT is used to transform two-dimensional data into the two-dimensional probability space, by transforming through each of the marginal distributions separately. This is done for

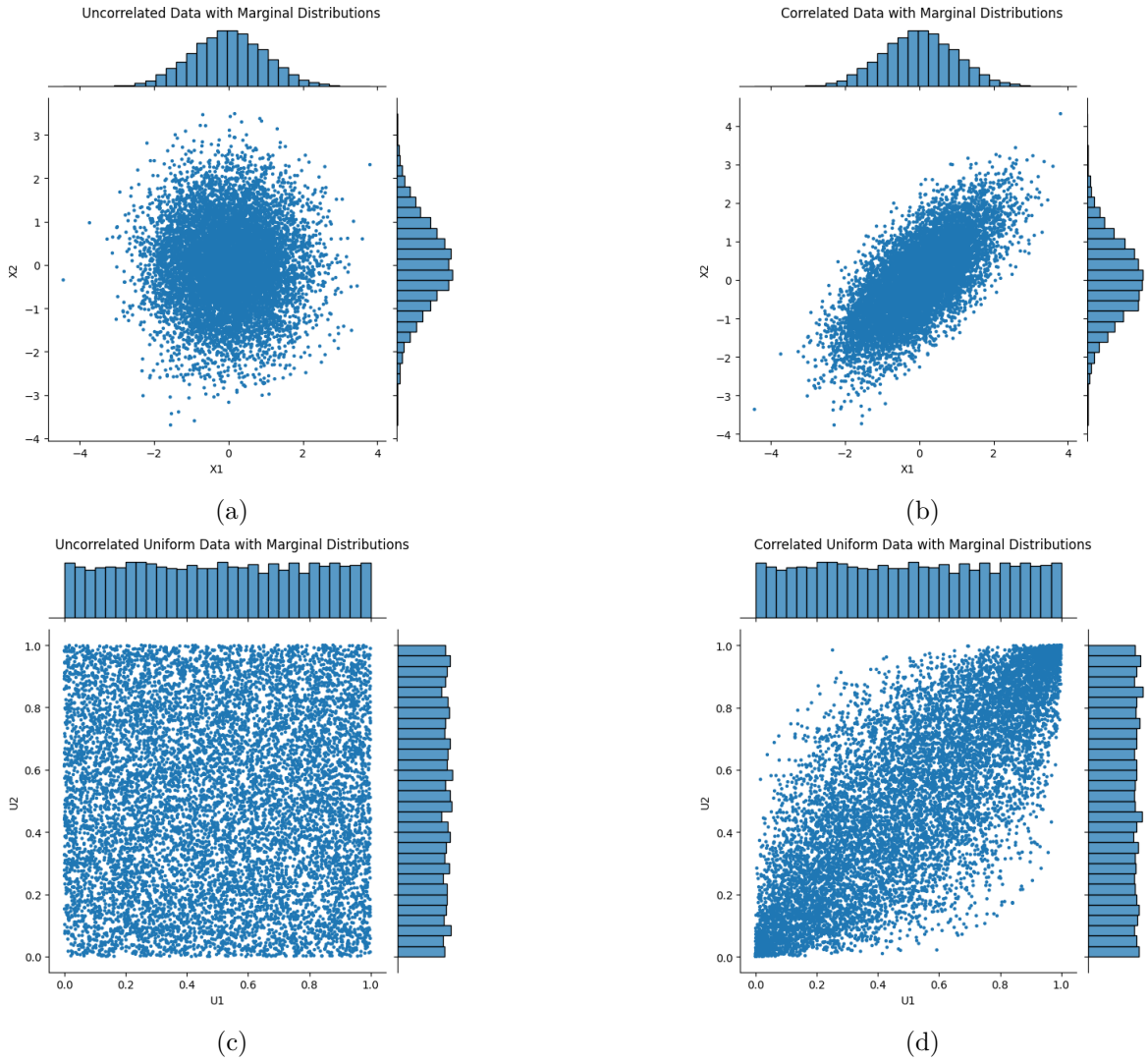


Figure 7: Illustration of how the PIT is used to transform two-dimensional data into the $[0, 1]$-domain by considering the marginal distribution of each dimension separately.

standard normally distributed data with and without correlation in (b) and (a) respectively, to highlight the differences in the distribution of points in the probability space when dependence is present and not in (d) and (c) respectively.

Before introducing copulas in the next section we can simply describe the setting for copulas as CDF of the data after having transformed it into the probability space using the PIT. Relating to Section 2.2 we the copula is the CDF of the data in (c) or (d).

Maximum likelihood Another important concept which will be used when fitting the different copulas is the Maximum Likelihood Estimation (MLE). To illustrate what it is, we first need to define the likelihood function as done by Wasserman (2010, p. 122)

Definition 2.18. Likelihood function The likelihood function for a sample of observations

$X_1, \ldots, X_n$ being IID with PDF $f(x;\theta)$ is defined by

$$\mathcal{L}_n(\theta) = \prod_{i=1}^{n} f(X_i;\theta).$$

Usually, the likelihood function becomes very small when the sample size increases. This is because the likelihood is often a value smaller than one, and a product of such values often goes to zero. Therefore, it is common to instead of using the likelihood function use the log likelihood function defined by

$$l_n(\theta) = \log(\mathcal{L}_n(\theta)).$$

We can see that the logarithm makes the product sum into a regular sum such that

$$l_n(\theta) = \log\left(\prod_{i=1}^{n} f(X_i;\theta)\right) = \sum_{i=1}^{n} \log(f(X_i;\theta)).$$

We can now define the MLE as done by Wasserman (2010, p. 122)

Definition 2.19. The *maximum likelihood estimator*, denoted by $\hat{\theta}_n$, is the value of θ that maximizes $\mathcal{L}_n(\theta)$.

Maximizing the likelihood function is equivalent to maximizing the log likelihood function, meaning that the parameter estimate is the same Wasserman (2010, p. 123).

2.3 Copula Theory

Write about why and when copulas are useful, Maybe show the solution to the problem described in the example earlier.

Now that we have a fundamental understanding of some probability theory we can introduce copulas. To do so we will need to define what it means for a function to be grounded, in Definition 2.20, and what it means to have margins, in Definition 2.21, Nelsen (2006, p. 9).

Definition 2.20. Grounded
Consider a function on the domain $S_1 \times S_2$ where S_1 and S_2 have the smallest elements a_1 and a_2 respectively. A function H from $S_1 \times S_2$ into $\mathbf{R}$ is *grounded* if $H(x, a_2) = H(a_1, y) = 0$ for all $(x, y) \in S_1 \times S_2$.

Definition 2.21. Margins
Consider a function on the domain $S_1 \times S_2$ where S_1 and S_2 have the largest elements b_1 and

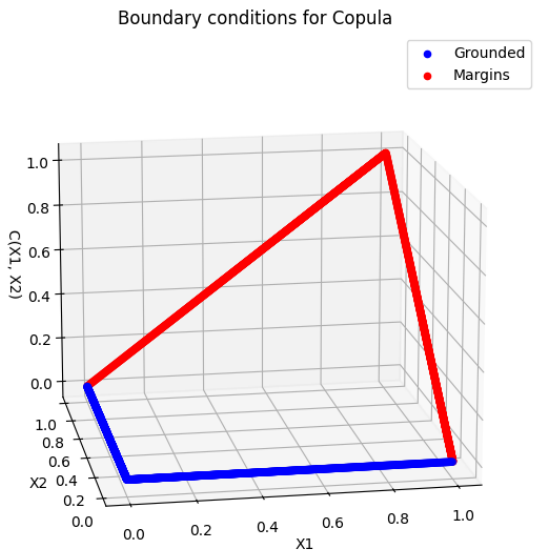


Figure 8: Illustration showing what it means for a function to be grounded and to have margins.

b_2 respectively. A function H from $S_1 \times S_2$ into $\mathbf{R}$ has *margins* F and G given by

$$\begin{aligned} \text{Dom}F = S_1, \text{ and } F(x) = H(x, b_2) \text{ for all } x \in S_1 \\ \text{Dom}G = S_2, \text{ and } G(x) = H(b_1, y) \text{ for all } y \in S_2. \end{aligned}$$

Figure 8 illustrates what Definition 2.20 and Definition 2.21 means when S_1 and S_2 are both the probability space. The blue points show what it means to be grounded and the red points show what it means to have margins.

Now we can formally define what a copula is. This is done as in Nelsen (2006, p. 10) in Definition 2.22.

Definition 2.22. Copula in 2 dimensions
Equivalently, a copula is a function C from $\mathrm{I}^2 = [0,1]^2$ to $\mathrm{I} = [0,1]$ with the following properties:

1. For every $u, v \in \mathrm{I}$,

$$\begin{aligned} C(u, 0) = C(0, v) = 0, \\ C(u, 1) = u, \text{ and } C(1, v) = v; \end{aligned}$$

2. For every $u_1, u_2, v_1, v_2 \in \mathrm{I}$ such that $u_1 \leq u_2$ and $v_1 \leq v_2$,

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

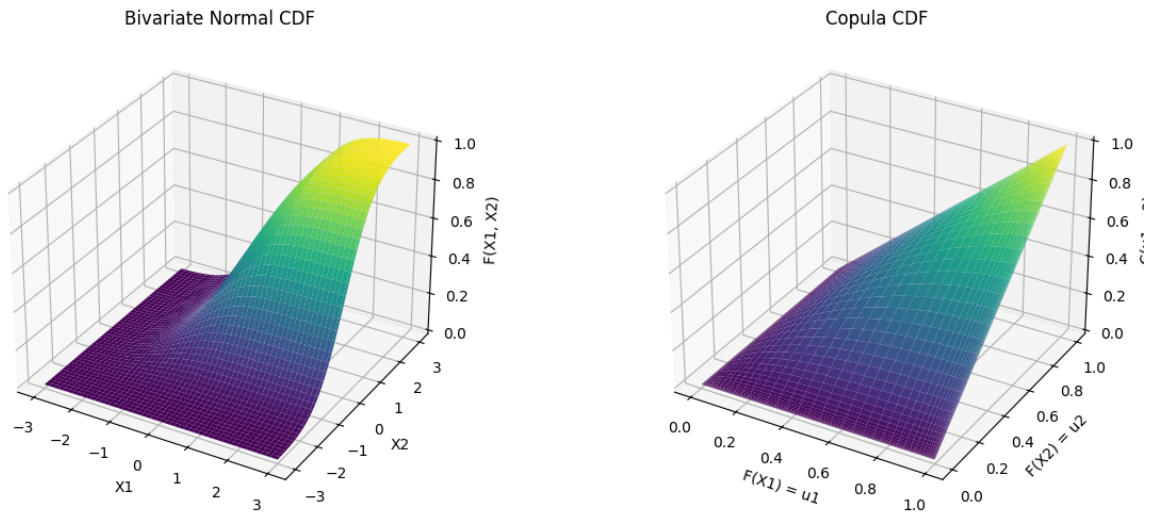


Figure 9: Illustration showing the correspondence between a bivariate normal CDF and its corresponding copula function.

So, to describe what a copula is in simpler terms, it is a CDF capturing the dependence between points on the probability space, obtained by performing the PIT on each marginal distribution.

A fundamental theorem connected to copulas is described in Theorem 2.23, as formulated by Nelsen (2006, p. 18).

Theorem 2.23. ***Theorem: Sklar's theorem***
Let H be a joint distribution function with margins F and G then there exists a copula C such that for all $x, y \in \bar{\mathbf{R}} = [-\infty, \infty]$,

$$H(x, y) = C(F(x), G(y)). \tag{1}$$

If F and G are continuous, then C is unique; otherwise, C is uniquely determined on $\text{Ran}(F) \times \text{Ran}(G)$. Conversely, if C is a copula and F and G are distribution functions, then the function H is a joint distribution function with margins F and G.

Figure 9 illustrates the correspondence, given in Equation (1), between a bivariate normal CDF and its corresponding copula function. The copula has the same function value as the CDF in each point where the mapping of points in $X_1 \times X_2$ to $[0, 1] \times [0, 1]$ is done by the PIT. The left picture shows the joint CDF on the $\bar{R}^2$ domain. The right picture shows the corresponding copula function which lives in probability space. Hence, the difference between the joint CDF and the copula is that the copula only exists on the unit square.

fix Picture so that it isn't cut

Sometimes it can be more convenient to write the Equation (1) theorem in terms of the marginal probabilities u_i rather than the marginal distributions x_i. It can be done by utilizing

that $F_i(x_i) = u_i$ and $F_i^{-1}(u_i) = x_i$ so that

$$H(F_1^{-1}(u_1), F_2^{-1}(u_2)) = H(x_1, x_2) = C(F_1(x_1), F_2(x_2)) = C(u_1, u_2).$$

Another important result for copulas is that they are translation invariant, the meaning of this is presented in Theorem 2.24, as done by Nelsen (2006, p. 25).

Theorem 2.24. ***Theorem: Translation Invariance***
Let X and Y be continuous random variables with copula $C_{X,Y}$. If α and β are strictly increasing on $Ran(X)$ and $Ran(Y)$, respectively, then $C_{\alpha(X),\beta(Y)} = C_{X,Y}$. Thus $C_{X,Y}$ is invariant under strictly increasing transformations of X and Y.

From the requirements for a function to be a copula given in the definition of a copula in Definition 2.22 one can obtain bounds for how a copula can look. These bounds are defined as in Theorem 2.25.

Theorem 2.25. ***Fréchet-Hoeffding bounds*** *(Schmidt (2006, p. 7) or Nelsen (2006, p. 11)) Consider a copula $C(\mathbf{u}) = C(u_1, u_2)$. Then the C is bounded by*

$$\begin{aligned} &L(u_1, u_2) \leq C(u_1, u_2) \leq U(u_1, u_2), \text{ where} \\ &L(u_1, u_2) = (u_1 + u_2 - 1)^+ \\ &U(u_1, u_2) = \min(u_1, u_2). \end{aligned}$$

L and U are referred to as the Fréchet-Hoeffding lower and upper bounds respectively.

Another important copula is the *independence copula* or product copula corresponding to when the two marginal distributions are independent. The independence copula is defined by (Brigo and Mercurio (2006, p. 712) or Schmidt (2006, p. 7)) as

$$\Pi(u_1, u_2) = u_1 u_2.$$

Figure 10 illustrates the upper (left) and lower (middle) bounds for copulas. We can think of it as if these bounds define a space where all copulas must be confined. One such copula is the independence copula (right) corresponding to the case when random variables are independent.

We have now discussed what copula functions are and arrived at that copulas are CDFs on the unit square. When fitting copulas to data using MLE it is often useful to have the copula PDF. Deriving the copula PDF is not as straightforward as one might initially think. Therefore we will derive the copula PDF in the general case. Let $C(u_1, u_2)$ be the cpula CDF you want to derive the PDF for. The copula PDF is defined as the second derivative of the

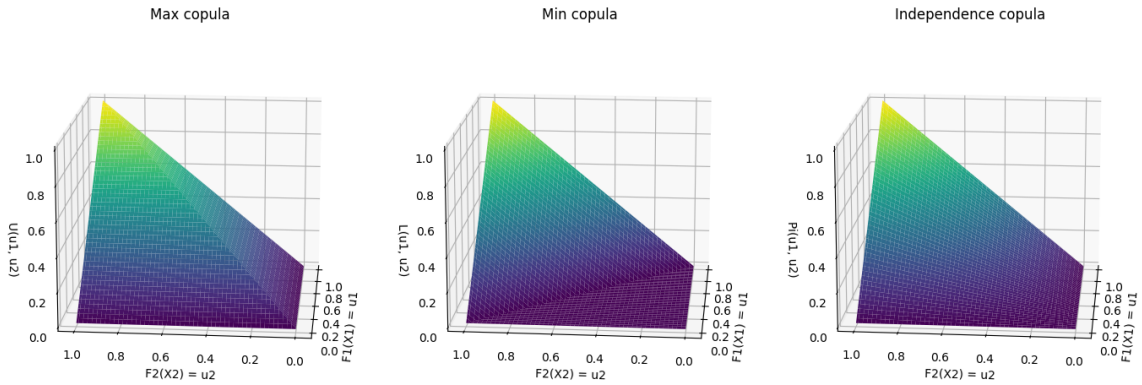


Figure 10: Illustration of the Fréchet-Hoeffding upper U (left), and lower L (middle) bounds, as well as the independence copula Π (right).

copula CDF with respect to u_1 and u_2.

$$c(u_1, u_2) = \frac{\partial^2 C(u_1, u_2))}{\partial u_1 \partial u_2} = \frac{\partial^2 F(F_1^{-1}(u_1), F_2^{-1}(u_2)))}{\partial u_1 \partial u_2} = f(F_1^{-1}(u_1), F_2^{-1}(u_2)) \frac{d}{du_1} F_1^{-1}(u_1) \frac{d}{du_2} F_2^{-1}(u_2),$$

where it can be shown that

$$\frac{d}{du_i} F_i^{-1}(u_i) = \frac{1}{f_i(F_i^{-1}(u_i))}.$$

In the above equations, F_i is the i:th marginal distribution function, CDF, and f_i is the marginal density function PDF, F is the joint distribution function.

Hence the copula PDF can be written as

$$c(u_1, u_2) = f(F_1^{-1}(u_1), F_2^{-1}(u_2)) \frac{1}{f_1(F_1^{-1}(u_1))} \frac{1}{f_2(F_2^{-1}(u_2))}.$$

It can also be useful to write the copula in terms of data points that in the return space as this is typically where the data is observed

$$c(x_1, x_2) = f(x_1, x_2) \frac{1}{f_1(x_1)} \frac{1}{f_2(x_2)},$$

where $x_i = F_i^{-1}(u_i)$ and f_i is the marginal PDF of the i:th marginal distribution.

2.3.1 Usefulness of copulas

The following examples are to illustrate how correlation can fail to capture the true dependence between random variables. The first example shows how correlation fails to capture the dependence between two random variables that are dependent but uncorrelated. The second

example shows how correlation underestimates the true dependence between two random variables that are dependent but not normally distributed. This is to illustrate how copulas can be a better alternative to correlation when measuring dependence between random variables.

Example 2.26. **Dependence is not always captured by correlation** Danielsson (2011, p. 21)
Let X be a random variable such that, $X \sim N(0,1)$. Obviously, X and X^2 are dependent. The covariance, $\text{Cov}(X, X^2)$ is

$$\begin{aligned}\text{Cov}(X, X^2) &= E\left[(X - E[X])(X^2 - E\left[X^2\right])\right] \\ &= E\left[(X)(X^2 - 1)\right] \\ &= E\left[X^3\right] - E[X] \\ &= 0 - 0 = 0.\end{aligned}$$

Since $\text{Cov}(X, X^2) = \rho_{X,X^2}\sigma_X\sigma_{X^2} = \rho_{X,X^2}$ this shows that $\rho_{X,X^2} = 0$. Hence, X and X^2 are dependent but uncorrelated. This shows how correlation falls short as a method of capturing non-linear dependence.

Using a copula instead we get

$$\begin{aligned}F_{X^2}(X^2) &= P(X^2 \leq x^2) \\ &= P(\sqrt{X^2} \leq \sqrt{x^2}) \\ &= P(|X| \leq |x|) \\ &= P(X \leq x) = U2 \neq U1.\end{aligned}$$

Since the absolute value signs information is lost and hence

$$P(U_1 \leq u1, U_2 \leq u2) = P(U_1 \leq u1, U_1 \leq u2) = \min(u_1, u_2).$$

Example 2.27. Let X be a random variable such that, $X \sim N(0,1)$. Obviously, X and e^X

are dependent. The covariance, $\text{Cov}(X, e^X)$ is

$$\begin{aligned}
\text{Cov}(X, e^X) &= E\left[(X - E[X])(e^X - E\left[e^X\right])\right] \\
&= E\left[X(e^X - E\left[e^X\right])\right] \\
&= E\left[Xe^X\right] - E\left[XE\left[e^X\right]\right] \\
&= E\left[Xe^X\right] \\
&= \int_{-\infty}^{\infty} xe^x \frac{1}{\sqrt{2\pi}} e^{\frac{-x^2}{2}} dx \\
&= \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} xe^{x-\frac{x^2}{2}} dx \\
&= \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} xe^{-\frac{1}{2}(x-1)^2+\frac{1}{2}} dx \\
&= \frac{e^{\frac{1}{2}}}{\sqrt{2\pi}} \int_{-\infty}^{\infty} xe^{-\frac{1}{2}(x-1)^2} dx \\
&\text{Let } u = x - 1 \text{ then } x = u + 1 \\
&= \frac{e^{\frac{1}{2}}}{\sqrt{2\pi}} \int_{-\infty}^{\infty} (u+1)e^{\frac{-u^2}{2}} du \\
&= \frac{e^{\frac{1}{2}}}{\sqrt{2\pi}} \left(\int_{-\infty}^{\infty} ue^{\frac{-u^2}{2}} du + \int_{-\infty}^{\infty} e^{\frac{-u^2}{2}} du\right) \\
&= \frac{e^{\frac{1}{2}}}{\sqrt{2\pi}} \sqrt{2\pi} \\
&= e^{\frac{1}{2}}.
\end{aligned}$$

If dividing with the standard deviations of X and e^X we get the correlation. To do this we need to calculate the variances of X and e^X. We know that $\text{Var}(X) = 1$ but need to calculate the variance of e^X. The variance of e^X is calculated as

$$\begin{aligned}
\text{Var}(e^X) &= E[e^{2X}] - E[e^X]^2 \\
&= e^2 - e.
\end{aligned}$$

The correlation is then calculated as

$$\begin{aligned}
\text{corr}(X, e^X) &= \frac{\text{cov}(x, e^X)}{\sqrt{\text{var}(x)}\sqrt{\text{var}(e^X)}} \\
&= \frac{e^{\frac{1}{2}}}{1\sqrt{e^2 - e}} \\
&= \sqrt{\frac{1}{e-1}} \approx 0.76.
\end{aligned}$$

If instead using the copula to calculate the correlation we can see that the correlation is

not capturing the true dependence between X and e^X. To see this we can use the PIT to transform X and e^X into uniform random variables. If we let $F_X(X) = U_1$ be the normal data transformed to uniform random variables and $F_{e^X}(e^X) = U_2$ be the log-normal data transformed to uniform random variables we can see that

$$\begin{aligned} F_{e^X}(e^X) &= P(e^X \leq e^x) \\ &= P(\ln(e^X) \leq \ln(e^x)) \\ &= P(X \leq x) = U2 = U1. \end{aligned}$$

Since $U_1 = U_2$ we have

$$P(U_1 \leq u1, U_2 \leq u2) = P(U_1 \leq u1, U_1 \leq u2) = \min(u_1, u_2).$$

This is the upper copula given by the upper Fréchet-Hoeffding bound, which is the copula corresponding to the correlation being one. Hence, correlation underestimates the true dependence between the random variables. Showing how copulas can capture dependence when correlation fails.

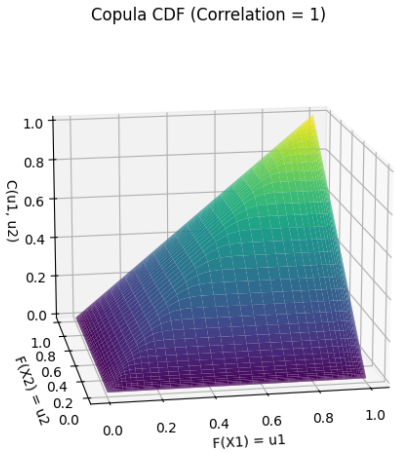
Would be nice to tie this to something in finance. Such as the relationship between stocks returns and derivative prices.
This shows that estimating the correlation in a setting when the different marginals are not both normal can be misleading. This is a problem in finance where one often uses correlation to estimate the dependence between different assets. This is especially true when using correlation to estimate the dependence between stocks and derivatives.

Up to now we have not yet provided an example of the usefulness of copulas. The next section will show an example of how copulas are useful in practice.

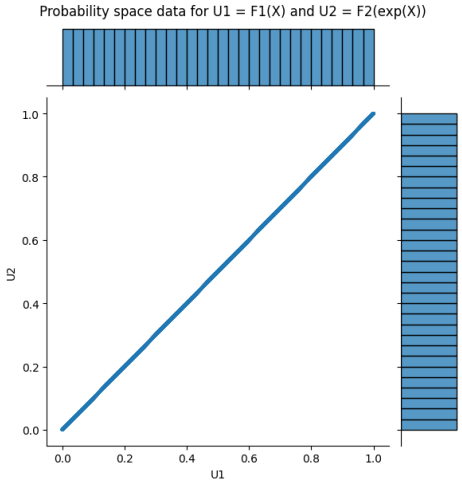
2.3.2 Usecase for copula in practice

If there are several sources of randomness present in a system where monte carlo methods are used, potential dependence between the sources of randomness has to be taken into account. Copulas are a powerful tool for modeling the dependence structure between random variables. They allow for separation of the dependence and the marginal distributions. By using copulas, we can generate samples from multivariate distributions with specified marginals and a desired dependence structure by generating dependent marginally distributed uniform random numbers. These are then transformed to the desired marginal distributions using the inverse of the marginal CDFs by the inverse transform method Definition 2.15.

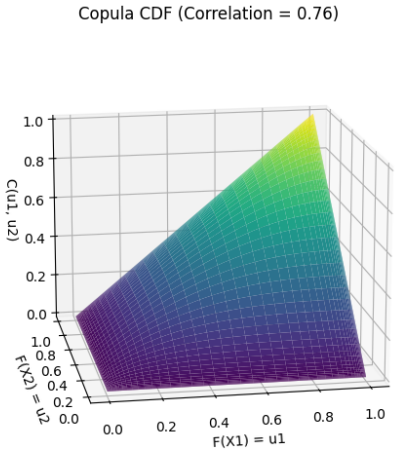
Let $U = (U_1, U_2)$ be a point specified by two marginally uniformly distributed random variables U_1 and U_2 sampled from a copula giving them some dependence. Then random numbers



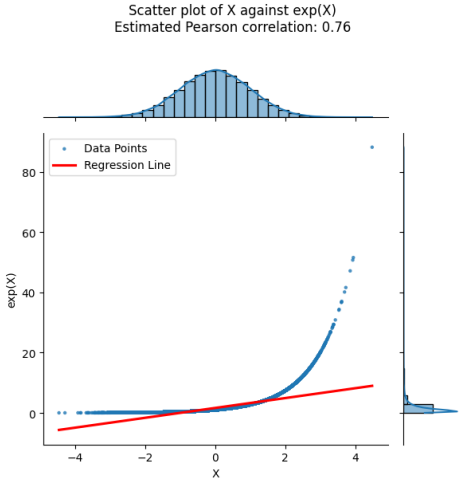
(a)



(b)



(c)



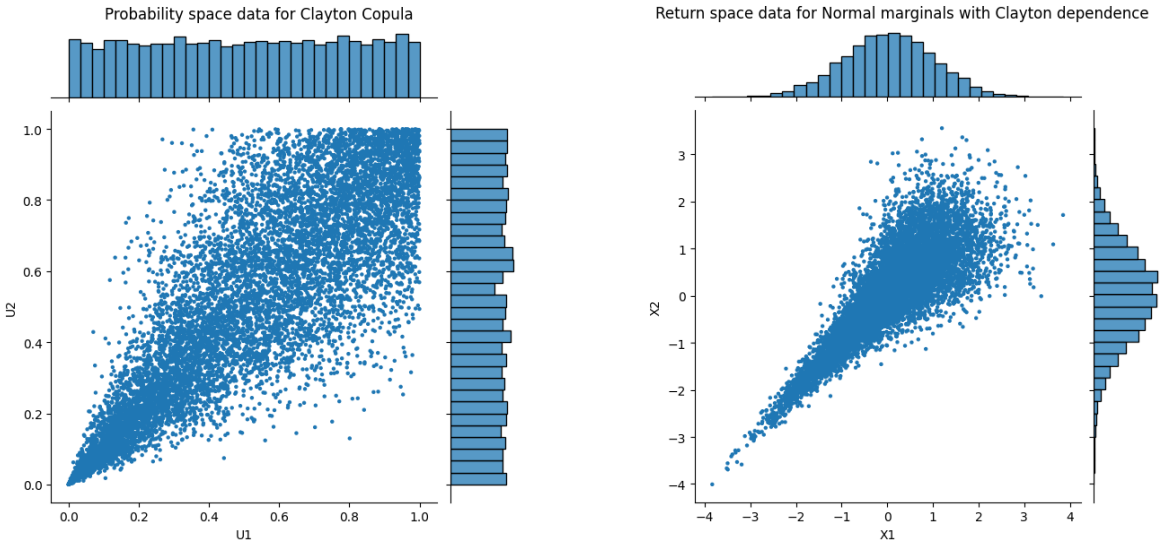
(d)

Figure 11: .

$X = (X_1, X_2)$ with the dependence specified by the copula and desired marginal distributions can be generated by transforming the uniform random numbers using the Inverse Transform Method (ITM). This is done by using the inverse of the marginal CDFs as follows

$$X_1 = F_1^{-1}(U_1);$$
$$X_2 = F_2^{-1}(U_2).$$

The usefulness of copulas in sampling is visualised in Figure 12. The left figure shows a sample from a Clayton copula with $\alpha = 4$. The right figure shows the same sample after applying the ITM using a standard normal distribution for each marginal. Note that the ITM can be applied to each of the dimensions of the copula data thanks to the fact that copulas by definition have uniform marginals.



(a) Sample from a Clayton copula with $\alpha = 4$.

(b) Sample from Clayton copula after applying the inverse transform method using a standard normal distribution.

Figure 12: Illustration of sampling from a clayton copula with standard normal marginal distributions.

In the above example we have seen the usefulness of copulas in sampling. The example does not go into the nontrivial task of sampling from a copula. Therefore the next section introduces conditional sampling which is a method of sampling from a copula.

2.3.3 Conditional sampling from copulas

todo

2.4 Other copulas

This section introduces some of the most commonly used copulas. Some copulas are defined through known statistical distributions such as the Gaussian copula and the Student's t copula. Another type of copulas is the family of Archimedean copulas, of which the Clayton copula is an example.

2.4.1 Gaussian Gopula

A commonly used copula is the Gaussian copula which is derived from the centered multivariate normal distribution.

Let $\mathbf{\Phi}_\Sigma(x_1, x_2)$ be the bivariate normal distribution CDF with correlation matrix Σ. Let $\Phi(x)$ be the univariate normal CDF. Then the Gaussian copula is defined, as in Cherubini et al. (2004, p. 112) by

$$C_\Sigma(u_1, u_2) = \mathbf{\Phi}_\Sigma(\Phi^{-1}(u_1), \Phi^{-1}(u_2)).$$

In the above expression Σ is the correlation matrix in two dimensions

$$\Sigma = \begin{bmatrix} 1 & \rho_{1,2} \\ \rho_{1,2} & 1 \end{bmatrix},$$

so in the sequel when ρ is used it refers to $\rho_{1,2}$ in the correlation matrix.

The copula function, which is a CDF can be expressed on integral form as done by Cherubini et al. (2004, p. 112)

$$C_\Sigma(u_1, u_2) = \int_{-\infty}^{\Phi^{-1}(u_1)} \int_{-\infty}^{\Phi^{-1}(u_2)} \frac{1}{2\pi\sqrt{1-\rho^2}} \exp\left\{ -\frac{s^2 - 2\rho st + t^2}{2(1-\rho^2)} \right\} dsdt.$$

The PDF of the Gaussian copula function can be shown to be

$$c_\Sigma(u_1, u_2) = \frac{1}{\sqrt{1-\rho^2}} \exp\left\{ \frac{-2\rho^2\Phi^{-1}(u_1)^2 + 2\rho\Phi^{-1}(u_1)\Phi^{-1}(u_2) - 2\rho^2\Phi^{-1}(u_2)^2}{2(1-\rho^2)} \right\},$$

as stated in Alexander (2008, p.267).

Fitting todo

Sampling todo

2.4.2 Students t Copula

Cherubini et al. (2004, p. 116) The Student's t copula can be defined similarly to the Gaussian copula above. We denote the Student's t copula by

$$C_{\nu,\Sigma}(u_1, u_2) = \boldsymbol{t}(t_\nu^{-1}(u_1), t_\nu^{-1}(u_1)),$$

as done by Alexander (2008, p. 268), where $\nu > 0$ is the degrees of freedom, $\boldsymbol{t}$ is the multivariate Student's t distribution, t is the univariate Student's t distribution, and Σ is the correlation matrix matrix.

The Student's t copula CDF is

$$C^t_{\nu,\Sigma}(u_1, u_2) = \int_{-\infty}^{t_\nu^{-1(u_1)}} \int_{-\infty}^{t_\nu^{-1(u_2)}} \frac{1}{2\pi\sqrt{1-\rho^2}} \left\{1 + \frac{s^2 - 2\rho st + t^2}{\nu(1-\rho^2)}\right\} dsdt,$$

as defined by Cherubini et al. (2004, p. 116). The corresponding copula CDF is, by the same logic as for the Gaussian case, the integrand in the expression above times the two partial derivatives of the marginal distributions. The resulting expression for the copula PDF $c_{\nu,\rho}$ is therefore

$$c_{\nu,\Sigma}(u_1, u_2) = \frac{\Gamma(\frac{\nu+2}{2})\Gamma(\frac{\nu}{2}) \prod_{j=1}^{2} \left(1 + \frac{t_\nu^{-1}(u_j)^2}{\nu}\right)^{\frac{\nu+2}{2}}}{\sqrt{\rho}\,\Gamma(\frac{\nu+1}{2})^2 \left(1 + \frac{t_\nu^{-1}(u_1)^2 + t_\nu^{-1}(u_2)^2 - 2\rho t_\nu^{-1}(u_1) t_\nu^{-1}(u_2)}{\nu(1-\rho^2)}\right)},$$

as stated by Cherubini et al. (2004, p. 117).

Say what gamma is

Fitting

todo

Sampling

todo

2.4.3 Clayton Copula

Archimedian copulas have the general expression

$$C(u_1, u_2) = \Psi^{-1}(\Psi(u_1), \Psi(u_2)),$$

for some generator function Ψ, as defined by Cherubini et al. (2004, p. 150).

The Clayton copula, which is one instance of an Archimedean copula, uses the generator function

$$\Psi(u) = u^{-\alpha} - 1,$$

and its inverse

$$\Psi^{-1}(x) = (x+1)^{\frac{-1}{\alpha}},$$

where $\alpha > 0$.

This gives the bivariate Clayton copula CDF

$$C(u_1, u_2) = (u_1^{-\alpha} + u_2^{-\alpha} - 1)^{\frac{-1}{\alpha}}.$$

The Clayton copula PDF is

$$c(u_1, u_2) = (\alpha+1)(u_1^{-\alpha} + u_2^{-\alpha} - 1)^{-2-\frac{1}{\alpha}} u_1^{-\alpha-1} u_2^{-\alpha-1},$$

as stated by Alexander (2008, p. 272).

Fitting todo

Sampling todo

2.4.4 Correlation measures

Keep this part until I decide if I will go deeper into this. (in Market risk analysis p.280 and p.256) Denote the correlation measure as $\hat{\rho}$

Pearson correlation for Pearson correlation $\hat{\rho} = \rho_p$

Spearman correlation For Spearman rho ρ_s:

$$\rho_s = 1 - \frac{6D}{n(n^2-1)}, \text{ where } D = \sum_{i=1}^{n} d_i^2.$$

$\hat{\rho} = 2\sin\left(\frac{\pi}{6}\rho_p\right)$

Kendals Tau For Kendals Tau τ:

$$\tau = \frac{N_c - N_d}{\frac{1}{2}n(n-1)}$$

where N_c and N_d are the numbers of concordant and discordant pairs respectively. $\hat{\rho} = \sin\left(\frac{\pi}{2}\tau\right)$ Maybe remove

2.5 Neural Networks

References to standard literature about NNs

ToDo

2.6 Neural Copula (Own model formulation)

Reference to paper

A NC is a method proposed to use a NN to approximate a copula function.

2.6.1 Overall procedure

The overall procedure when using a NC is to fit one NN approximating each marginal distribution. These functions and their derivatives are then used to calculate the copula function value and the copula PDF function values

transform the data in each data dimension to the $[0, 1]$ domain via the PIT. The resulting data points form data points that lie in probability space. These data points are then used for fitting a NN approximating the copula function.

2.6.2 Data

First, define the observed data as $x^{\text{obs}} \in \mathbb{R}^2$. To fit the marginal distributions, the data must be normalized to lie in the interval $\mathbb{I}^2 = [0, 1]^2$. We normalize the data x by using min max normalization defined as

$$x_{i,j} = \frac{x_{i,j}^{\text{obs}} - \min_i(x_j^{\text{obs}})}{\max_i(x_j^{\text{obs}}) - \min_i(x_j^{\text{obs}})}.$$

The normalized data x is then used to fit a marginal distribution $\hat{F}_j$ to each dimension x_j, where $j = \{1, 2\}$ if x has two dimensions.

2.6.3 Marginal model architecture

A marginal distribution is fitted for each dimension of the data using a NN having the following architecture

$$\begin{aligned}
\text{Input layer} &: \mathbf{h}_m^0 = x_j \in \Omega_j = \mathbb{I}; \\
\text{Hidden layer} &: \mathbf{h}_m^{k+1} = \tanh(\mathbf{w}_m^k \mathbf{h}_m^k + \mathbf{b}_m^k),\ k \in \{0, 1, \ldots, l_m - 1\}; \\
\text{Output layer} &: \hat{F}_j = \text{sigmoid}(\mathbf{w}_m^{l_m} \mathbf{h}_m^{l_m} + \mathbf{b}_m^{l_m}) \in [0, 1],
\end{aligned}$$

where $\hat{F}_j$ is the estimated marginal CDF, Ω is the entire domain where the data can be, l_m is the number of layers in the network, $\mathbf{w}_m^k$ are the weights in the kth layer, $\mathbf{h}_m^k$ is the input data in the kth layer, and $\mathbf{b}_m^k$ is the biases in the kth layer. Note that the weights, biases, and input data are different for each dimension in the data (maybe obvious, but for notations' sake).

2.6.4 Marginal loss function

This section defines the loss function for a marginal model $\hat{F}_j$. First, we define the datasets that will be used for calculating the loss function. Let D_{obs} be the set of observed data points x_j for the jth dimension of the data. Also, let D_u be a set of uniformly distributed data points on the domain Ω of the data. Let $\hat{f}_j$ denote the PDF corresponding to $\hat{F}_j$.

The loss function consists of four parts that are designed to ensure that the network fits the data as well as possible whilst following the requirements of a CDF. The first part of the loss function maximizes the log likelihood of the fitted CDF to the observed data. Since the network minimizes the loss during training, we need the negative log likelihood as the loss. The first part of the loss is defined as

$$L_1^m = \frac{-1}{|D_{obs}|} \sum_{i \in D_{obs}} \log(\hat{f}_j(x_i)).$$

The second part of the loss function ensures that the PDF is not negative. This corresponds to the loss function

$$L_2^m = \int_{x_{i \in \Omega}} (-\hat{f}(x))^+ dx \approx \frac{-1}{|D_u|} \sum_{i \in D_u} (-\hat{f}_j(x_i))^+.$$

The third part of the loss function ensures that the integral of $\hat{f}$ over Ω is one as stated in remark 2.8. The loss can be formulated as

$$L_3^m = \left| 1 - \int_{x \in \Omega} \hat{f}(x) dx \right| \approx \left| 1 - \frac{1}{|D_u|} \sum_{i \in D_u} \hat{f}_j(x_i) \right|.$$

The final loss ensures that the CDF begins at zero and ends at one. This is ensured by the loss

$$L_4^m = \hat{F}_j(0) + |1 - \hat{F}_j(1)|.$$

The total loss can be formulated as a linear combination of the loss components.

$$L^m = \sum_{i=1}^{4} \lambda_i L_i^m,$$

where λ_i is the weigh put on the i:th loss component.

2.6.5 Copula model architecture

Zeng and Wang (2022, pp. 9–11) The copula model $\hat{C}$ is defined as

$$\begin{aligned}
\text{Input layer} &: \mathbf{h}_c^0 = x \in \Omega^2 = \mathbb{I}^2; \\
\text{Hidden layer} &: \mathbf{h}_c^{k+1} = \tanh(\mathbf{w}_c^k \mathbf{h}_c^k + \mathbf{b}_c^k),\ k \in \{0, 1, \ldots, l_c - 1\}; \\
\text{Output layer} &: \hat{C} = \text{sigmoid}(\mathbf{w}_c^{l_c} \mathbf{h}_c^{l_c} + \mathbf{b}_c^{l_c}) \in [0, 1]\,.
\end{aligned}$$

Should i change u and x to bold since vectors?

2.6.6 Copula loss function

Before defining the copula loss function, we need to define the different data sets used to compute the loss. Additionally, we need to define the copula CDF and its corresponding copula PDF.

First, we define the datasets. Let D_{obs} be the observed data points $x \in \Omega_1\Omega_2 = \mathbb{I}^2$. Let D_u be uniformly distributed data on $\mathbb{I}^2$. Let $D_{\bar{u}}$ be uniformly distributed points on the upper boundary of the unit square $\mathbb{I}^2$. Let $D_{\underline{u}}$ be a set of points uniformly distributed on the lower boundary of the unit square $\mathbb{I}$. In Figure 13 the datasets used to train the NC are visualized. Before explaining the picture, we ephasise that the data used here is in the x-space and not in the transformed space u as one might think. The reason for this is the case is that it will be easier to write the copula PDF in terms of x than of u as explained below. The blue data points are uniformly distributed over the upper boundaries of the region where one variable is one. The grey points are uniformly distributed on the lower boundaries of the region where one of the variables is zero. The red data points are the observed data. The black data points are uniformly distributed over the interval.

The copula model itself approximates a copula function which is a CDF on the unit square $\mathbb{I}^2$. Normally the copula function is denoted in terms of its uniform marginals u consisting of u_1 and u_2. The copula function can be written in terms of the data points x consisting of x_1 and x_2.

$$\hat{C}(u) = \hat{C}(u_1, u_2) = \hat{C}(\hat{F}_1(x_1), \hat{F}_2(x_2)) = \hat{C}(x).$$

The corresponding copula PDF can be written as

2nd half unnecessary?

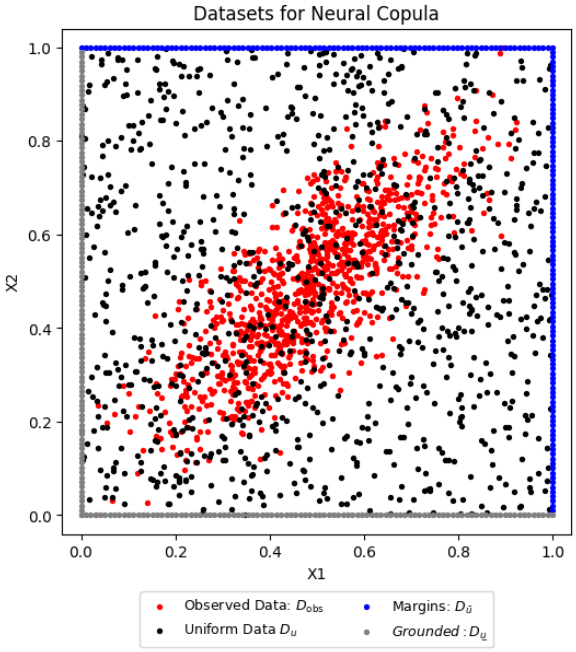


Figure 13: The different datasets used for training the NC visualized.

$$\begin{aligned}\hat{c}(x) = \hat{c}(x_1, x_2) &= \frac{\partial^2 \hat{C}(\hat{F}_1(x_1), \hat{F}_2(x_2))}{\partial \hat{F}_1(x_1) \partial \hat{F}_2(x_2)} \hat{f}_1(x_1), \hat{f}_2(x_2)) \\ &= \frac{\partial^2 \hat{C}(u_1, u_2)}{\partial u_1 \partial u_2} \hat{f}_1(\hat{F}_1^{-1}(u_1)) \hat{f}_2(\hat{F}_2^{-1}(u_2)) = \hat{c}(u).\end{aligned}$$

In the loss function, it is advantageous to have the copula CDF and PDF written in terms of x rather than of u. This is because there is no mechanism for calculating the inverse of a NN. The inverse is needed in order to write the copula PDF in terms of u. However, as shown above, the copula PDF can be written in terms of x using the marginal CDFs and PDFs, removing the need for the inverse CDFs.

Now we are ready to introduce the copula loss function, which consists of five parts. The first part, as in the marginal model, maximizes the log likelihood and is defined as

$$L_1^c = \frac{-1}{|D_{obs}|} \sum_{i \in D_{obs}} \log(\hat{c}(x_i)).$$

The second term ensures positivity of the copula PDF and is defined by

$$L_2^c = \int_{x_1 \in \Omega} \int_{x_2 \in \Omega} (-\hat{c}(x))^+ dx_1 dx_2 \approx \frac{-1}{|D_u|} \sum_{i \in D_u} (-\hat{c}(x_i))^+.$$

The third part ensures that the copula PDF integrates to one and is defined as

$$L_3^c = \left| 1 - \int_{x_1 \in \Omega} \int_{x_2 \in \Omega} \hat{c}(x_j) dx_1 dx_2 \right| \approx \left| 1 - \frac{1}{|D_{\mathrm{u}}|^2} \sum_{i \in D_u} \hat{c}(x_i) \right|.$$

The fourth term is supposed to ensure that the copula function is grounded and that it has margins. It is defined as

$$L_4^c = \sum_{i \in D_{\underline{u}}} \hat{C}(x_i) + \sum_{i \in D_{\bar{u}}} \left| \hat{C}(x_i) - \max(x_i) \right|.$$

The fifth term is supposed to ensure that the copula is 2-increasing and is defined as

$$L_5^c = \frac{1}{|D_{\text{obs}}||D_u|} \sum_{i \in D_u} \left| \hat{C}(x_i) - \sum_{j \in D_{\text{obs}}} \text{flag}(x_i, x_j) \right|$$

where flag(x, y) is defined as

$$\text{flag}(x, y) = \begin{cases} 1 \text{ if } \forall j \leq 2, y_j < x_j \\ 0 \text{ otherwise.} \end{cases}$$

So the flag function compares two points x and y and gives back one if all dimensions x is greater than y in all dimensions.

As for the marginal model, the total loss is given by a linear combination of the loss terms defined as follows

$$L^c = \sum_{i=1}^{5} \lambda_i L_i^c,$$

where λ_i is the weight assigned to each part loss term.

Fitting

Mention that in neural copula article the data is normalized to 0,1. Talk about implications in risk management.

Sampling

Mention chebyshev nodes, could be good to use splines gfor inverted marginals.

2.7 Goodness of fit measures

To use for results evaluation

3 Methodology

This section explains and motivates the method used for the experiment tested in this thesis. First, we provide a summary of the method used, as this will help to give the overall procedure without getting stuck in the details. Then, a detailed description of each different part will

be given.

1. Method overview
2. Test of marginal model (does it deviate for some different distributions)
3. Choice of method for copula
4. Portfolio testing (Actual experiment)

3.1 Method overview

Connect to research question and hypothesis

This section describes the method used for the experiment. The method is divided into three main parts.

First, a test of the marginal model is performed to see how well the marginal distributions are fitted. This is to see if the marginal distribution fitted by the marginal model works as intended. Theoretically there should not be any difference between the fitted marginal distribution and the true distribution of the data given a sufficient number of samples. This is important as the copula is invariant to strictly increasing transformations, meaning that the fitted copula should be able to replicate the joint distribution of the data regardless of the marginal distributions used. In the main experiment we will know that the log returns are normally distributed. In reality this is not always the case, and therefore we want to test how well the fitted marginal distribution works.

The second part of the method is to test how the neural copula should be trained. This is done by testing different training schemes for the NC. The goal of this part is to find a method that works universally for the neural copula meaning that the trained copula model produces a valid copula function that fits the data well. This is done by testing different training schemes and comparing the fitted copulas to one another. This is all to find a method that works well for the NC meaning that the fitted copula is valid.

illustrate workflow

The third part of the method is the main experiment of this thesis and the overall procedure is as follows. + To begin with, different portfolios with different types of dependency structures will be generated. This will be done by sampling data in probability space from different copulas and transforming it to return space using the ITM to obtain dependent normally distributed returns. This ensures that the marginal distributions of the created portfolios have normal marginal distributions, removing the need for fitting them in this experiment. This allows for an evaluation of the pure performance of the copula, without conflating it with potential errors from fitting marginal distributions. The generated normally distributed random numbers are then used as the random shocks from the Weiner process when simulating the GBM using the Euler-Maruyama scheme to replicate stock price time series. This creates a realistic setting tha for when using copulas would be suitable.

The generated price time series are then divided into two different parts. These different parts

represent the historical and the future returns of the portfolios. The historical part is used for fitting the different copulas, while the future part can be considered the true distribution of future returns. As described when introducing MC methods the key assumption is that of the statistical distribution of the data and that the distribution remains the same in the future. Hence the historical data is used to fit the copulas to the historical data, using MLE, from which random numbers, replicating the joint distribution, can be generated. If the copula adequately captures the dependence the generated data should be similar to the future data. This is the main goal of this thesis, to evaluate how well different copulas can replicate the joint distribution of the data.

To evaluate the various copulas' ability to accurately capture the dependence between random variables, it is therefore sensible to compare the generated data to the testing data. The reason for using simulated data in this experiment is to ensure that the joint distribution is constant over time as this is a key assumption when using monte carlo methods. If the data generated from the fitted copula is similar to the testing data, it shows that the dependence is appropriately modeled by the copula. If not, it shows that the copula is not well suited to model the dependence. Hence this is a good way to evaluate the copulas' performance in an isolated manner.

To emphasize the key assumption when using copulas to simulate data for Monte Carlo purposes. The assumption is that the joint distribution observed in the past continues to be the distribution from which the data is generated into the future.

Method for comparison... continue

3.2 Marginal model test

To validate that the marginal model works as intended, a test is performed to see how well the fitted marginal distribution matches the true distribution of the data. This is done by generating data from a known distribution and then fitting a marginal model to it. The fitted marginal model is then compared to the true distribution of the data by transforming the data to probability space using the PIT using both the fitted distribution and the true distribution. To evaluate how similar the points in probability space are to each other the Mean absolute error (MAE) is calculated. Additionally a QQ-plot is created to visualize how well the fitted distribution matches how well the data follows the true distribution. During this experiment it is important to keep in mind that the generated data is subject to random noise, therefore it is not necessarily the case that the true distribution is the same as the observed distribution even tough they should be the same. It is however a good benchmark to see how well the fitted distribution matches the true distribution, which would be a reasonable choice of distribution to use if the neural network was not used. The distributions used for the test are listed in Table 1 where the distribution, the parameter values, and a short description is displayed for each of the tested distributions.

Table 1: Distributions and parameters used for the marginal model test.

Distribution	Parameters	Description
Gaussian	$\mu = 0, \sigma = 1$	Standard normal distribution
Student's t	$\nu = 5$	Student's t-distribution with 5 degrees of freedom
Uniform	$a = 0, b = 1$	Uniform distribution on [0, 1]
Exponential	$\lambda = 1$	Exponential distribution with rate 1
Laplace	$\mu = 0, b = 1$	Laplace distribution with mean 0 and scale 1
Log-normal	$\mu = 0, \sigma^2 = 1$	Log-normal distribution with mean 0 and variance 1

3.3 Neural Copula training scheme test

In this section the procedure for finding a method that reliably obtains valid copula functions in training. Wether or not a copula function is valid is determined by wether or not it satisfies the conditions of being a copula defined in Definition 2.22. In practice for the neural copula this means that all but the first of the loss terms, defined in Section 2.6.6, should approach zero when training is done. Several different training schemes will be tested with the goal of finding a method that works well for the neural copula meaning that the fitted copula is valid.

Several datasets will be used for testing the neural copula fitting procedure. This is to ensure that the method works well, regardless of the data, consistently resulting in valid copulas. The method for this workflow is as follows for each dataset. First different datasets will be generated and a grid of different parameter values will be created. The grid will contain How different values for the parameters of the neural copula such as the solver used, learning rates, number of epochs, batch size, network architecture, number of data points in the additional uniform datasets. This creates a large number of different combinations of choices to test which will hopefully result in a method for finding a training method that works well universally.

For each of the combinations in the grid the an ensemble of runs will be performed to mitigate the impact of what random seeds are used for initializing the weights of the network. The ensemble will consist of 10 runs for each combination in the grid. The best run from the ensemble will be kept for each combination in the grid.

For each of the different runs in the ensemble both a several different schemes for training the neural copula will be tested. The first scheme is to train the copula on the dataset without any pretraining. The second scheme is to pretrain the copula on a dataset generated from the independence copula and then using the weights of the pretrained copula to initialize the weights of the neural copula when training on the dataset. There are three different variations of this pretraining scheme. The first is to train the copula on the independence copula using the full loss function defined in Section 2.6.6. The second is to train the copula on the independence copula using the loss function without the L1 term. The third is to train the copula on the independence copula using only the MSE as loss function. The rationale

behind pretraining this is that this should produce a network mimicking the independence copula which can be thought of as a middle ground between the upper and lower Frechet-Hoeffding bounds. This is potentially a good starting point for the weights when fitting the copula to other data.

After pretraining, the neural copula is trained on the dataset using the weights from pretraining as a starting point. The starting weights in each of the pretraining schemes are saved so that after evaluation they can be used as a starting point for training the copula on a dataset. This will be needed in the final experiment of this thesis, described in Section 3.4, where the best training scheme and initial weights will be used for the NC.

After training the best overall method over each of the datasets will be used evaluated by validating that the losses relating to the copula function constraints are almost zero. This indicates that the fitted copula is valid. Of the methods that do not violate the copula definition the method that has the best fit overall will be declared the best one and hence be used in the final experiment, described in Section 3.4.

Given generated datasets and a specified grid

For each dataset

- Fit marginal models
- **For grid alternative**
  - **For run in ensemble**
    - **Normal Training save weights**
      - Train copula on dataset
    - **Using Pretraining**

      Pretraining on independence copula save weights
      - Full loss function
      - Training without L1 term in loss function
      - Training with MSE loss function

      Posttraining on dataset starting at pretraining end weights
      - Training on dataset for each pretraining scheme
  - **Choose best run in ensemble keep weights for that run (to use if the method is used)**

Evaluate the average performance of methods over the datasets

Neural copula what ive tested (state best and tell what ive tested)

How have i changed the neural copula approach ToDo

3.4 Portfolio testing

This section details the steps in the test of the copulas on the different portfolios.

3.4.1 Data Generation

To evaluate the performance of different copulas when fitting them to data. Given that this thesis focuses on the use of copulas for modeling financial returns, we want to create test portfolios consisting of pairs of stocks. These portfolios should ideally cover a wide range of dependence structures to test the different copulas' versatility and robustness in varying conditions.

This thesis focuses on the role of the copula purely therefore, the aim is not to conflate the results from the copula's performance with that of the marginal fitting procedure. Therefore, we want each marginal distribution of the generated portfolios to be the same known distribution. The marginal distributions used should not matter, given that the copula is invariant to strictly increasing transformations as stated in Theorem 2.24. In this study, it is advantageous to remove the need for fitting the different marginal distributions, by just using the normal distribution. If having different marginal distributions the number of combinations to test during model fitting becomes large.

Add reference saying that marginal dist fitting is studied.

To generate portfolios with different dependence structures and the same marginal distributions, copulas will be used. This will be done by first sampling data from analytical copulas with different parameter values. This will result in data points in probability space that contain the pure dependence between the different variables. These data points will then be plugged into the inverse CDF of a Standard normal distribution. This performs the PIT in reverse, creating data points with standard normal marginal distributions with the dependence described by the copula.

After having generated these pairs of dependent standard, normally distributed random numbers, they are used as the random shocks when simulating the bivariate GBM using the Euler-Maruyama scheme. This results in the test portfolios replicating stock prices over time, creating a realistic setting for when using copulas is appropriate.

as defined in...

The above procedure is used with the following copulas and parameter values.

Decide on portfolios, and time horizon, and mean and volatility

The data generated is divided into a fitting and a testing part. The fitting part will be used in the model fitting and can be thought of as historically observed data. The testing part will be used in the model evaluation and can be thought of as data that will appear in the future, which we want to replicate as well as possible by fitting a copula to the historical data.

3.4.2 Model Fitting

The fitting data from each of the previously generated portfolios is used for fitting the different copulas. To do this, the data needs to be converted so that the copulas can work with it. Hence, the log returns are calculated as defined in Definition 2.3. These are then standardized and centered to have a zero mean and unit variance.

maybe not necessary to scale...

Gaussian Copula To fit the Gaussian copula, the correlation matrix of the log returns data is calculated.

Student's t copula The Student's t copula is fitted by first estimating the correlation matrix from the log returns. The degree of freedom used for the copula is then fitted using MLE. This can be done using the copula PDF by calculating the sum of log likelihoods over the data transformed to probability space using the PIT. The log likelihood is then maximized with respect to the degrees of freedom.

Clayton copula The Clayton copula is fitted by MLE by maximizing the sum of log likelihoods from the copula PDF over the transformed data with respect to the parameter α.

Neural copula The NC is fitted to the data by the method described in ... Talk about the neural copula fitting

3.4.3 Model Sampling

The main use for copulas is arguably to sample random numbers that can be used to generate dependent realizations of a pair of random variables, regardless of their marginal distributions. Hence, it should make sense to evaluate the copulas based on how well generated random numbers from a fitted copula replicate the true dependence structure of the data.

Gaussian Copula To generate samples from a Gaussian copula, one can generate data from a multivariate standard Gaussian distribution with a correlation matrix. Each dimension of the generated data can then be plugged into its marginal distributions (being standard normal) CDF. This performs the PIT on the data, ensuring that the resulting data lies in the unit square.

Student's t copula To generate data from a Student's t copula works in the same way as for the Gaussian copula. That is, one sample from a multivariate Student's t distribution with some correlation matrix, and then perform the PIT on the data by plugging each dimension of the sampled data into the marginal distribution, which is a univariate Student's t distribution with the same degrees of freedom.

Clayton copula Sampling from a Clayton copula requires the use of conditional sampling. detail this

Neural copula TBC

The sampled data points from the copulas are inserted into the inverse Gaussian distribution and then scaled to match the observed standard deviation of the fitting data. This should generate data similar to the data in the testing part if the copula adequately captures the dependence in the data.

3.4.4 Model evaluation

Evaluation

3.5 Experiment workflow

1. Data generation
   a) Generate returns
   b) Put into GBM as random shocks
   c) Calculate log returns
   d) Split into different parts (train - test)
   e) Normalize and center data
2. Model fitting (for each copula)
   a) Fit marginal on training data if necessary (only for neural)
   b) Perform PIT on training
   c) Fit copula functions on transformed training
3. Model evaluation
   Alternative 1
   a) Sample new returns based on fitted model
   b) Compare on distribution level to the testing data
   Alternative 2
   a) Compare fitted copula to empirical copula of test data

4 Results and Discussion

Results and Discussion

5 Conclusion

Conclusion

References

Alexander, Carol (2008). *Market Risk Analysis - Practical Financial Econometrics*. Vol. Vol. 2. Market Risk Analysis. John Wiley & Sons Ltd. isbn: 978-0-470-99801-4.

Bjork, Tomas (2009). *Arbitrage Theory in Continuous Time*. OUP Catalogue 9780199574742. Oxford University Press. isbn: ARRAY(0x5f5f2288). url: https://ideas.repec.org/b/oxp/obooks/9780199574742.html.

Brigo, Damiano and Fabio Mercurio (2006). *Interest Rate Models - Theory and Practice*. 2. Ed. Springer Finance. Springer. isbn: 978-3-540-22149-4.

Cherubini, Umberto., Elisa. Luciano and Walter. Vecchiato (2004). *Copula methods in finance*. eng. Wiley finance series. Hoboken, NJ: John Wiley & Sons. isbn: 0470863447.

Danielsson, Jon (Mar. 2011). *Financial risk forecasting: the theory and practice of forecasting market risk with implementation in R and Matlab*.

Devore, Jay L. and Kenneth N. Berk (2012). *Modern Mathematical Statistics with Applications*. 2nd. Springer Texts in Statistics. New York, NY: Springer. url: https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=2540618&site=ehost-live.

Evans, M.J. and J.S. Rosenthal (2004). *Probability and Statistics: The Science of Uncertainty*. W. H. Freeman. isbn: 9780716747420. url: https://books.google.se/books?id=hkWK8kFzXWIC.

Glasserman, Paul (2004). *Monte Carlo methods in financial engineering*. New York: Springer. isbn: 0387004513 9780387004518 1441918221 9781441918222. url: http://www.amazon.com/Financial-Engineering-Stochastic-Modelling-Probability/dp/0387004513/ref=pd_sim_b_68?ie=UTF8&refRID=1AN8JXSDGMEV2RPHFC2A.

Nelsen, Roger B. (2006). *An introduction to copulas*. New York: Springer. isbn: 0387286594 9780387286594. url: http://www.worldcat.org/search?qt=worldcat_org_all&q=0387286594.

Raychaudhuri, Samik (2008). "Introduction to Monte Carlo simulation". In: *2008 Winter Simulation Conference*, pp. 91–100. doi: 10.1109/WSC.2008.4736059.

Schmidt, T (2006). "Coping with Copulas". In.

Wasserman, Larry (2010). *All of statistics : a concise course in statistical inference*. New York: Springer. isbn: 9781441923226 1441923225. url: http://www.amazon.de/All-Statistics-Statistical-Inference-Springer/dp/1441923225/ref=sr_1_2?ie=UTF8&qid=1356099149&sr=8-2.

Zeng, Zhi and Ting Wang (2022). "Neural Copula: A unified framework for estimating generic high-dimensional Copula functions". In: *arXiv* 0.0. doi: 10.48550/arXiv.2205.15031.