Homework 4

Due on March 21, 2025

Instructions:

- Install `pdflatex`, `R`, and `RStudio` on your computer.
- Please edit the `HW4_First_Last.Rnw` file in `Rstudio` and compile with `knitr` instead of `Sweave`. Go to the menu `RStudio|Preferences...|Sweave` choose the `knitr` option, i.e., `Weave Rnw files using knitr?` You may have to install `knitr` and other necessary packages.
- Replace "First" and "Last" in the file-name with your first and last names, respectively. Complete your assignment by modifying and/or adding necessary R-code in the text below.
- You should submit both a **PDF file** and a **ZIP file** containing the data and the `HW4_First_Last.Rnw` file required to produce the PDF. The file should be named: "`HW4_First_Last.zip`" and it should contain a single folder named "`First_Last`" with all the necessary data files, the `HW4_First_Last.Rnw` and `HW4_First_Last.pdf` file, which was obtained from compiling `HW4_First_Last.Rnw` with `knitr` and LaTeX.
- The GSI grader will annotate the PDF file you submit. However, they will also check if the code you provide in the ZIP file compiles correctly. If the file fails to compile due to errors other than missing packages, there will be an automatic 10% deduction to your score.

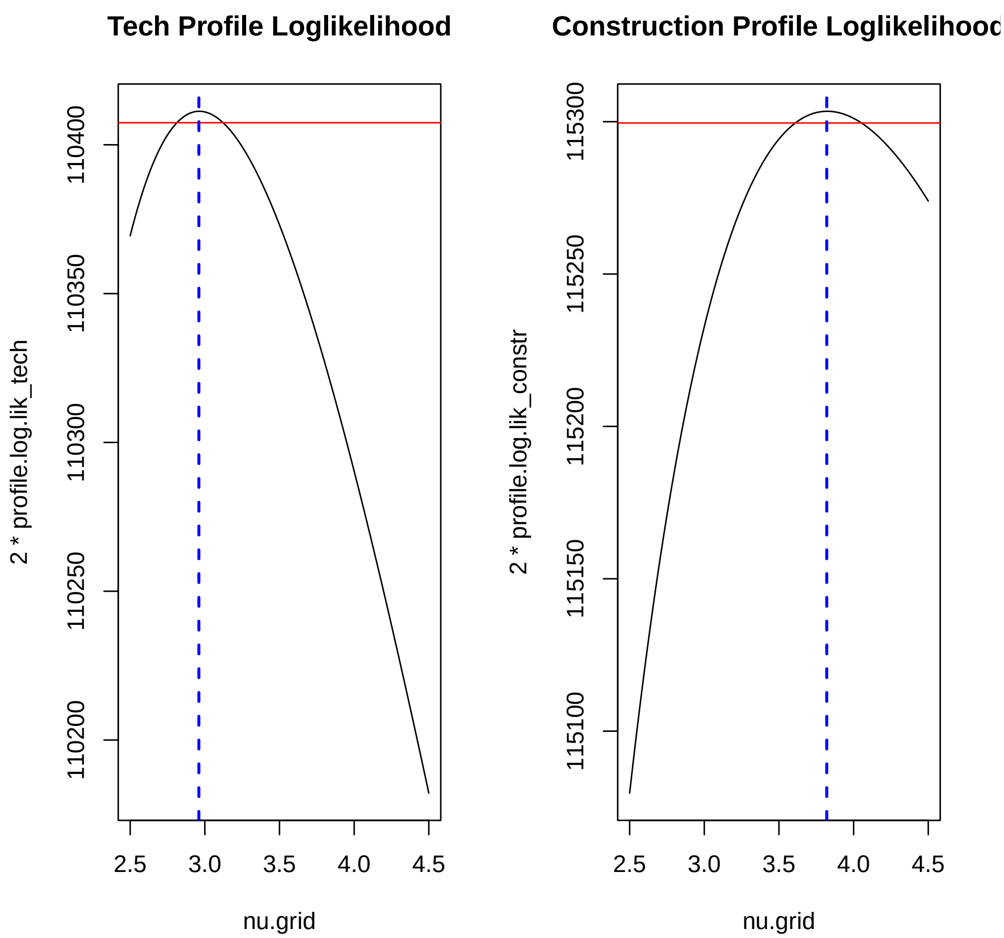
Note: For your convenience a number of R-functions are included in the .Rnw file and not shown in the compiled PDF. **It is your responsibility to fix any bugs in the included functions so that your results are correct.**

Problems:

1. Consider the R-code mimicking the code in Example 7.4 (page 168) in the text.

```r
library(MASS);
library(mnormt);
data = read.csv("stock_returns.csv",header = T)
dates = as.Date(data$date);
idx = which((dates > as.Date("2000-01-01")))

tech = c(2,3,8,9);
constr = c(4,5,6,7)
dat_tech = data[idx,tech]
dat_constr = data[idx,constr]

nu.grid = seq(2.5, 4.5, 0.01);
n = length(nu.grid);
profile.log.lik_tech = c();
profile.log.lik_constr = c();
```

```
for (nu in nu.grid){
 fit_tech = cov.trob(dat_tech,nu=nu)
 fit_constr = cov.trob(dat_constr,nu=nu)

 profile.log.lik_tech= c(profile.log.lik_tech,
   sum(log(dmt(dat_tech,mean=fit_tech$center,S=fit_tech$cov,df=nu))))

 profile.log.lik_constr= c(profile.log.lik_constr,
   sum(log(dmt(dat_constr,mean=fit_constr$center,S=fit_constr$cov,df=nu))))
}

par(mfrow=c(1,2))
plot(nu.grid,2*profile.log.lik_tech,type="l",main=
      "Tech Profile Loglikelihood")
alpha = 0.05; q.alpha = qchisq(1-alpha,1);
abline(h=2*max(profile.log.lik_tech) - q.alpha,col="red")
z1_tech = (2*profile.log.lik_tech > 2*max(profile.log.lik_tech) - q.alpha)
max_profile_loglik_tech = max(2 * profile.log.lik_tech[which(z1_tech)])
max_nu_tech = nu.grid[which(2 * profile.log.lik_tech == max_profile_loglik_tech)][1]
abline(v=max_nu_tech, col="blue", lwd=2, lty=2)


plot(nu.grid,2*profile.log.lik_constr,type="l",main=
      "Construction Profile Loglikelihood")
alpha = 0.05; q.alpha = qchisq(1-alpha,1);
abline(h=2*max(profile.log.lik_constr) - q.alpha,col="red")
z1_constr = (2*profile.log.lik_constr > 2*max(profile.log.lik_constr) - q.alpha)
max_profile_loglik_constr = max(2 * profile.log.lik_constr[which(z1_constr)])
max_nu_constr = nu.grid[which(2 * profile.log.lik_constr == max_profile_loglik_constr)][1]
abline(v=max_nu_constr, col="blue", lwd=2, lty=2)
```




```
CI_lower_tech = min(nu.grid[which(z1_tech)])
CI_upper_tech = max(nu.grid[which(z1_tech)])
cat('CI of Technology Sector: \n[', CI_lower_tech, ', ', CI_upper_tech, ']')

## CI of Technology Sector:
## [ 2.82 ,  3.12 ]

CI_lower_constr = min(nu.grid[which(z1_constr)])
CI_upper_constr = max(nu.grid[which(z1_constr)])
cat('CI of Construction Sector: \n[', CI_lower_constr, ', ', CI_upper_constr, ']')

## CI of Construction Sector:
## [ 3.61 ,  4.05 ]
```

(a) Modify the above code to fit two separate multivariate t-distribution models to the set of tech-sector stocks (”AAPL”, ”AMD”, ”INTC”, ”IBM”) and the construction sector stocks

(”AA”, ”CAT”, ”F”, ”MMM”). Provide 95% confidence intervals for the parameters nu.

Are the joint distributions for these two sectors significantly different? Comment.
From the plot, the optimal ν for the tech sector is notably lower than for the construction sector. And because the CIs do not overlap, these two distributions are significantly different.

Tech sector stocks (e.g., AAPL, AMD, INTC, IBM) are more volatile and experience heavier tails, whereas construction sector stocks (e.g., AA, CAT, F, MMM) exhibit more stable returns.

(b) Repeat the analysis in part **(a)** for each decade, i.e., $1980-1989$, $1990-1999$, $2000-2009$ and $2010-2019$ by judiciously changing the grid of ν parameter values. Plot the resulting 95% confidence intervals for ν along with the MLE $\hat{\nu}_{ML}$ as a function of the decade period. Do so for each of the two sectors and comment.

The tech sector exhibits more variability in ν, especially during the 2000-2009 period when market conditions were more turbulent.

The construction sector remains more stable, with only a slight temporary increase in heavy-tailed behavior during 2000-2009.

Confidence intervals for the two sectors do not always overlap, particularly during the 2000-2009 period, reinforcing the idea that these two sectors experience different levels of volatility and risk exposure.

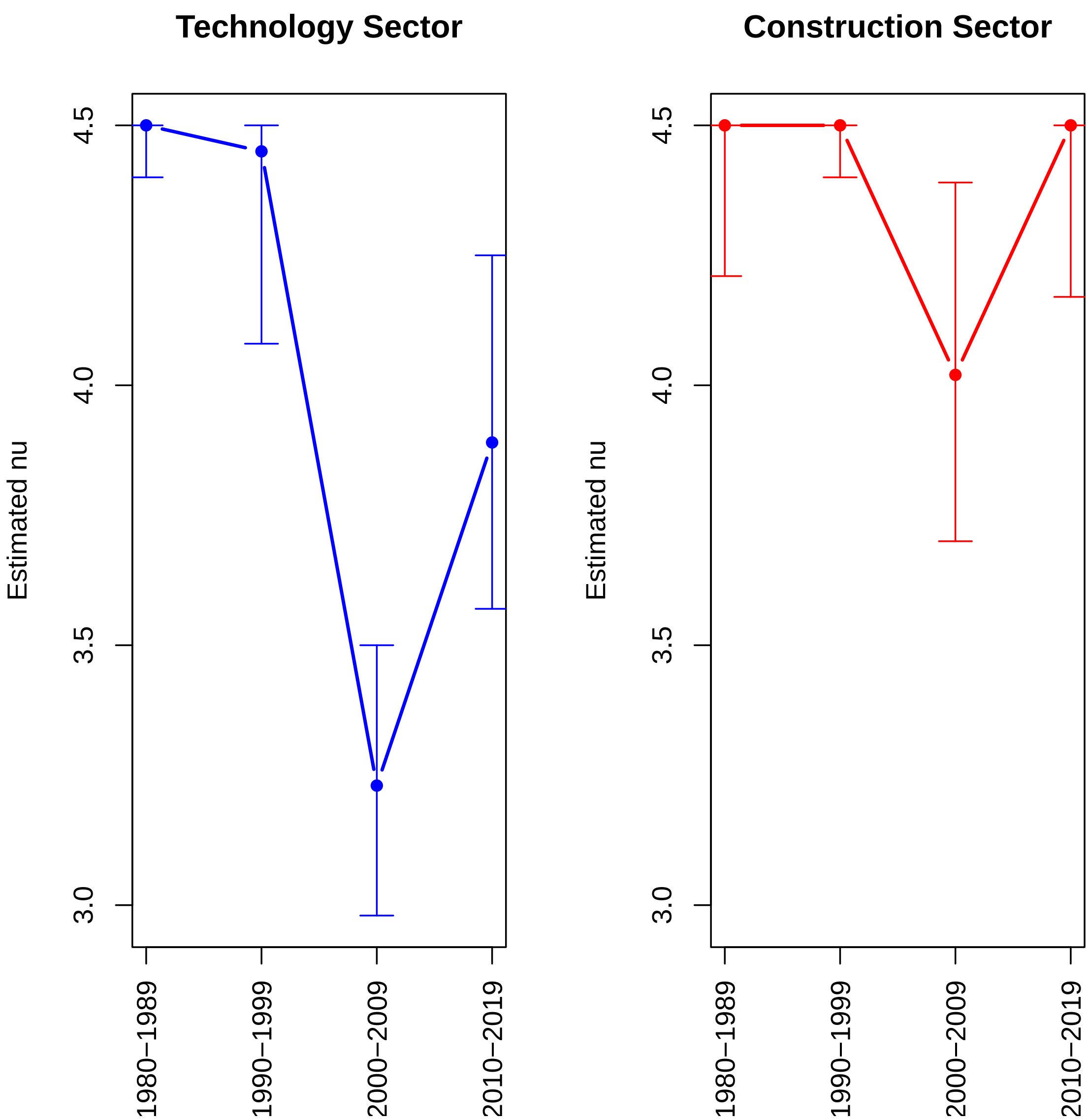
```
decades <- list(
  "1980-1989" = c(as.Date("1980-01-01"), as.Date("1989-12-31")),
  "1990-1999" = c(as.Date("1990-01-01"), as.Date("1999-12-31")),
  "2000-2009" = c(as.Date("2000-01-01"), as.Date("2009-12-31")),
  "2010-2019" = c(as.Date("2010-01-01"), as.Date("2019-12-31"))
)

results <- data.frame(Decade=character(), Sector=character(), MLE=numeric(), CI_lower=nume

for (decade in names(decades)) {
  idx <- which(dates >= decades[[decade]][1] & dates <= decades[[decade]][2])
  dat_tech <- data[idx, tech]
  dat_constr <- data[idx, constr]

  profile.log.lik_tech <- c()
  profile.log.lik_constr <- c()

  for (nu in nu.grid) {
    fit_tech = cov.trob(dat_tech, nu=nu)
    fit_constr = cov.trob(dat_constr, nu=nu)

    profile.log.lik_tech <- c(profile.log.lik_tech,sum(log(dmt(dat_tech, mean=fit_tech$cen
    profile.log.lik_constr <- c(profile.log.lik_constr,sum(log(dmt(dat_constr, mean=fit_co
```

```
  }

  # Compute MLE and Confidence Intervals for Tech Sector
  MLE_tech <- nu.grid[which.max(profile.log.lik_tech)]
  threshold_tech <- max(profile.log.lik_tech) - qchisq(0.95, df=1) / 2
  CI_bounds_tech <- nu.grid[profile.log.lik_tech >= threshold_tech]
  CI_lower_tech <- min(CI_bounds_tech)
  CI_upper_tech <- max(CI_bounds_tech)
  results <- rbind(results, data.frame(Decade=decade, Sector="Tech", MLE=MLE_tech, CI_lowe

  MLE_constr <- nu.grid[which.max(profile.log.lik_constr)]
  threshold_constr <- max(profile.log.lik_constr) - qchisq(0.95, df=1) / 2
  CI_bounds_constr <- nu.grid[profile.log.lik_constr >= threshold_constr]
  CI_lower_constr <- min(CI_bounds_constr)
  CI_upper_constr <- max(CI_bounds_constr)
  results <- rbind(results, data.frame(Decade=decade, Sector="Construction", MLE=MLE_const
}

library(knitr)
kable(results)
```

Decade	Sector	MLE	CI_lower	CI_upper
1980-1989	Tech	4.50	4.40	4.50
1980-1989	Construction	4.50	4.21	4.50
1990-1999	Tech	4.45	4.08	4.50
1990-1999	Construction	4.50	4.40	4.50
2000-2009	Tech	3.23	2.98	3.50
2000-2009	Construction	4.02	3.70	4.39
2010-2019	Tech	3.89	3.57	4.25
2010-2019	Construction	4.50	4.17	4.50

```
par(mfrow=c(1,2))
results$DecadeNum <- as.numeric(factor(results$Decade, levels=unique(results$Decade)))

# -------- (1) Plot Tech Sector with Confidence Intervals --------
tech_results <- subset(results, Sector == "Tech")

plot(tech_results$DecadeNum, tech_results$MLE, type="b", col="blue", pch=16, lwd=2,
     ylim=range(results$CI_lower, results$CI_upper),
     xlab=NA, ylab="Estimated nu", main="Technology Sector",
     xaxt="n")

arrows(tech_results$DecadeNum, tech_results$CI_lower,
       tech_results$DecadeNum, tech_results$CI_upper,
       angle=90, code=3, length=0.1, col="blue")
```

```r
axis(1, at=tech_results$DecadeNum, labels=tech_results$Decade, las=2)

# -------- (2) Plot Construction Sector with Confidence Intervals --------
constr_results <- subset(results, Sector == "Construction")

plot(constr_results$DecadeNum, constr_results$MLE, type="b", col="red", pch=16, lwd=2,
     ylim=range(results$CI_lower, results$CI_upper),
     xlab=NA, ylab="Estimated nu", main="Construction Sector",
     xaxt="n")

arrows(constr_results$DecadeNum, constr_results$CI_lower,
       constr_results$DecadeNum, constr_results$CI_upper,
       angle=90, code=3, length=0.1, col="red")

axis(1, at=constr_results$DecadeNum, labels=constr_results$Decade, las=2)
```

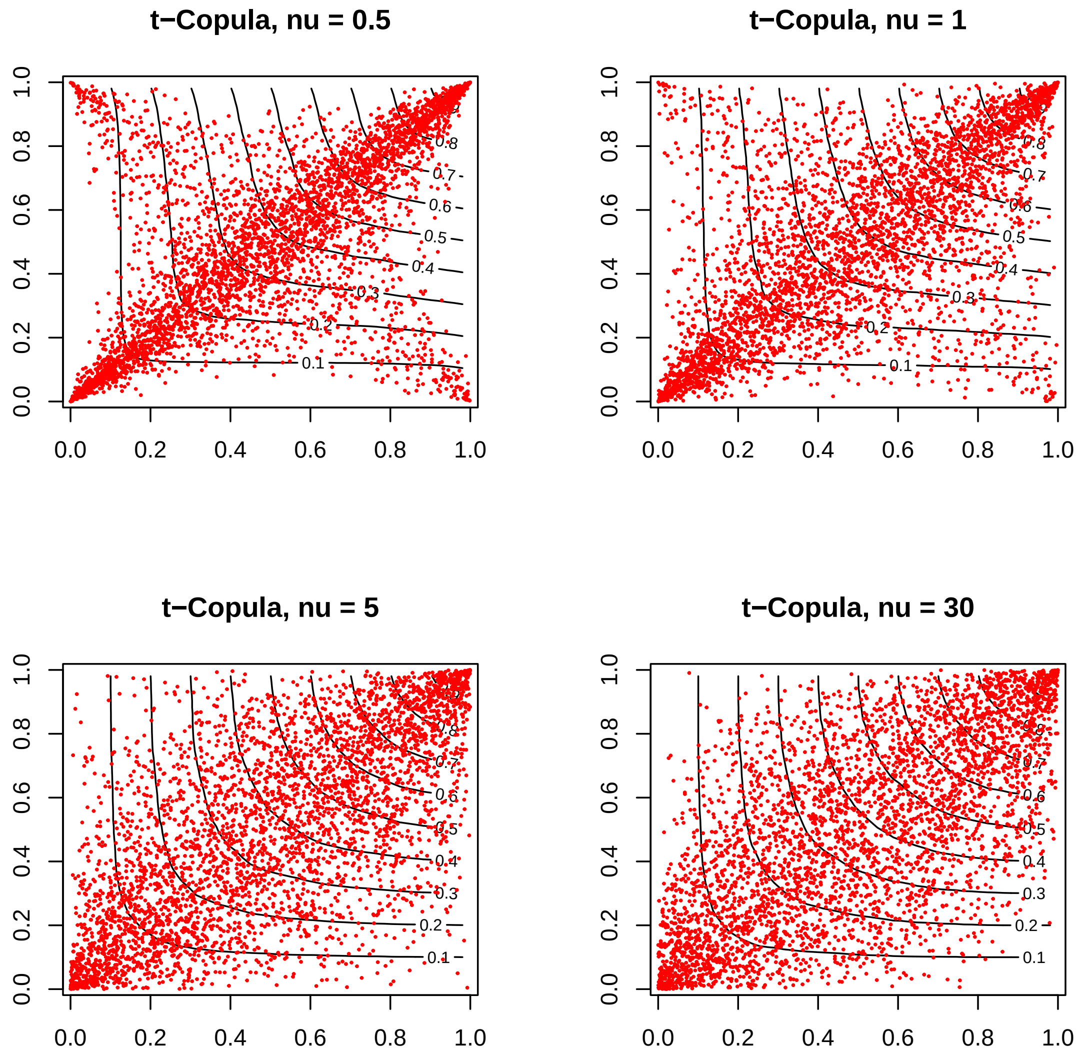


Note: If you wish you may also consider shorter time-periods like consecutive 5−year periods starting from 1980.

2. The goal of this exercise is to practice simulation from copula models, qualitative copula calibration via Kendall's τ, as well as tail-dependence inference.

   (a) The following function simulates from the Gaussian copula. Write a similar function that simulates from the t-copula, given a correlation matrix Σ and the parameter ν. Validate and illustrate the output of your simulation (for several values of ν) using the function `emp.copula` (as in part **(b)** below).

```
sim_Gauss_copula <- function(n,Sig){
   require("MASS");
   Z = MASS::mvrnorm(n = n, mu = Sig[,1]*0, Sigma=Sig);
   se = sqrt( diag(Sig) )
   se.inv = se;
   se.inv[se>0] = 1/se[se>0];
   Z = Z%*% diag(se.inv)
   return(pnorm(Z))
}

sim_T_copula <- function(n,Sig, nu){
   require("MASS");
   Z = MASS::mvrnorm(n = n, mu = Sig[,1]*0, Sigma=Sig);
   se = sqrt( diag(Sig) )
   se.inv = se;
   se.inv[se>0] = 1/se[se>0];
   Z = Z%*% diag(se.inv)

   y = rchisq(n, df=nu) / nu
   T = Z / sqrt(y)
   U = pt(T, df=nu)
   return(U)
}

rho = 0.7
Sig = matrix(c(1, rho, rho, 1), nrow=2, ncol=2)
par(mfrow=c(2,2))

nu_values = c(0.5, 1, 5, 30)

for (nu in nu_values) {
  U = sim_T_copula(n=4000, Sig=Sig, nu=nu)
  emp.copula(data=U, n.pts=50, plot.points=T,
             main=paste("t-Copula, nu =", nu))
}
```



(b) Load the data from Problem 1 of the daily log-returns of `IBM` and `CAT`. Recall the fact $\rho = \sin(\pi\rho_\tau/2)$, where ρ_τ is Kendall's τ and ρ is the correlation parameter in the matrix

$$\Sigma = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix},$$

for both the Gaussian and t-distribution copula. (See Result 8.1 in the textbook.)

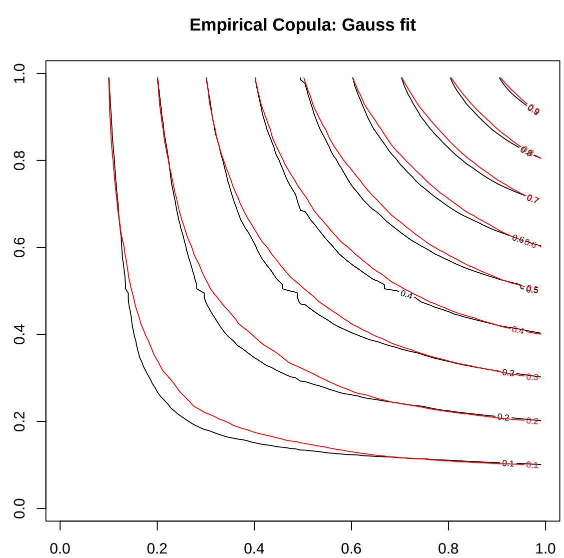
Modify and add to the following code to produce comparative empirical copula plots for the Gaussian **as well as** the t-distribution copula calibrated to the data. Do so for several different values of ν until you obtain a qualitatively good fit matching the empirical copula contours. Discuss the results of your analysis, i.e., which copula seems to be calibrated best to the data.

The Gaussian copula is inadequate for modeling IBM and CAT returns due to its lack of tail dependence.

The t-copula provides a significantly better fit, and among different values of ν, the best calibration is achieved when $\nu = 5$.

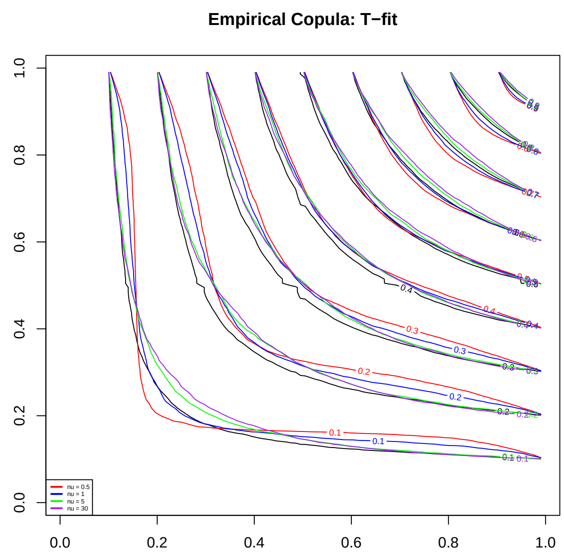
Although the result still deviates from the empirical copula, perhaps due to modeling limitations such as the assumption of constant dependence structure, finite sample size effects, or the presence of asymmetric tail dependence that the t-copula may not fully capture, $\nu = 5$ provides the best result among the values I examined.

```
rho.tau = cor(data$AAPL,data$CAT,method="kendall")
rho = sin(rho.tau*pi/2)
Sig = matrix(c(1,rho,rho,1),2,2)

par(mfrow=c(1,1))
c.emp = emp.copula(data = cbind(data$IBM,data$CAT),
               main = "Empirical Copula: Gauss fit");

U.Gauss = sim_Gauss_copula(n=10000,Sig=Sig)
c.Gauss = emp.copula(data = U.Gauss,add.lines = TRUE,col = "red")
```




```
nu_values = c(0.5, 1, 5, 30)
colors = c("red", "blue", "green", "purple")

par(mfrow=c(1,1))
c.emp = emp.copula(data = cbind(data$IBM,data$CAT),
```

```
         main = "Empirical Copula: T-fit")
for (i in 1:length(nu_values)) {
    nu = nu_values[i]
    U.t = sim_T_copula(n = 10000, Sig = Sig, nu = nu)
    emp.copula(data = U.t, add.lines = TRUE, col = colors[i],
               main = paste("t-Copula Fit, nu =", nu))
}

legend("bottomleft", legend = paste("nu =", nu_values),
       col = colors, lty = 1, lwd = 2, cex = 0.4)
```



(c) Tail-dependence. Using the qualitatively calibrated t-distribution copula model, produce an estimate of the lower tail dependence coefficient for the `IBM` and `CAT` stocks and compare it with the empirical lower tail dependence coefficient. That is, if C_t and C_{emp} are the calibrated t-copula and the empirical copula of the data, respectively, plot on the same plot

$$\lambda_{t,L}(\alpha) := \frac{C_t(\alpha,\alpha)}{\alpha} \quad \text{and} \quad \lambda_{emp,L}(\alpha) := \frac{C_{\text{emp}}(\alpha,\alpha)}{\alpha},$$

for $\alpha =$ `seq(0.001,to=0.1,by=0.001)`. You can evaluate C_t either using Monte Carlo (by simulating from the t-copula) or numerically using the R-package copula. You can use the function `lambda` defined (but not displayed) above. Do not use `emp.copula`!

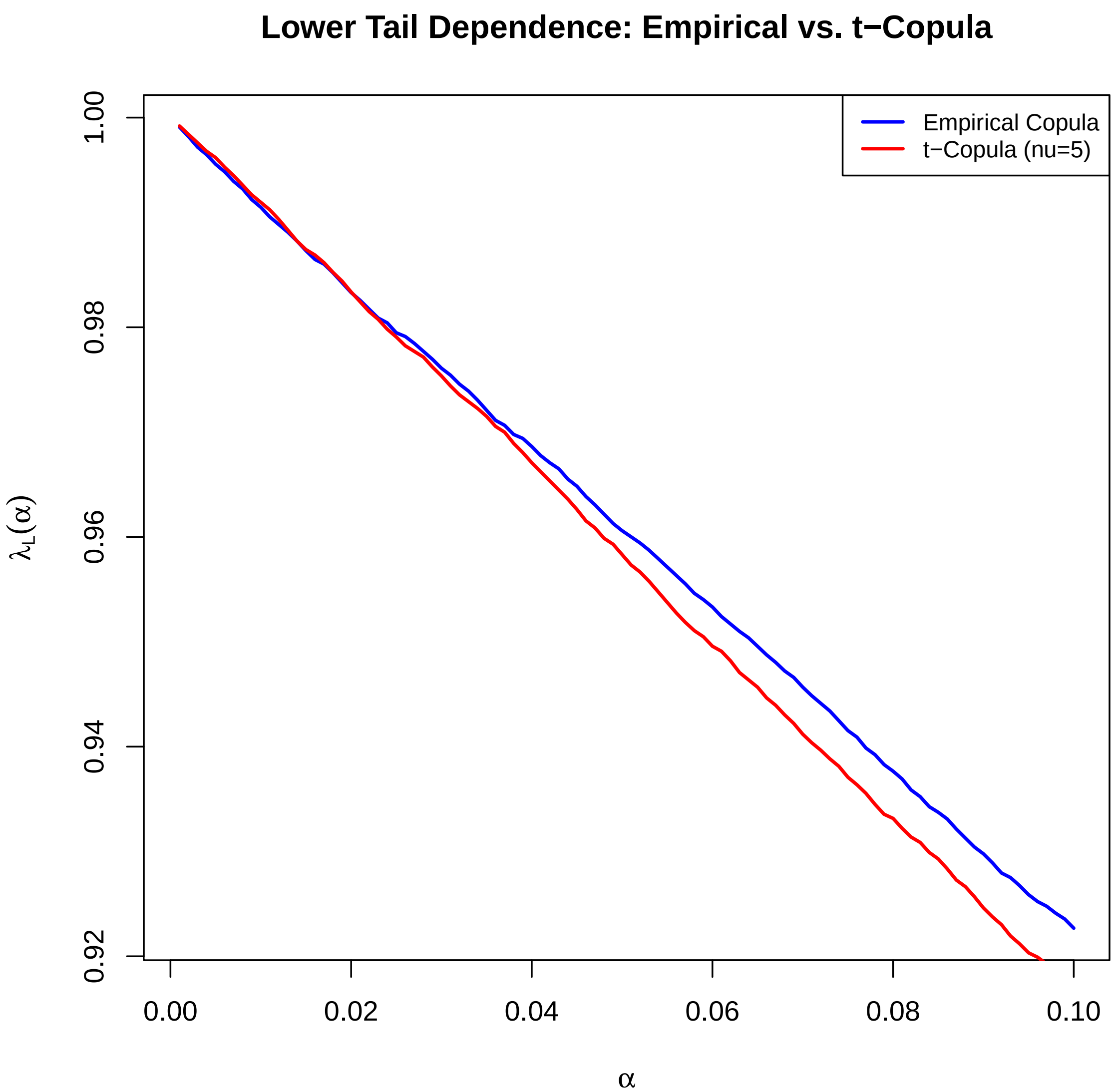
Discuss: What is the probability that the daily loss of `IBM` exceeds VaR_α(`IBM`), given that the daily loss of `CAT` exceeds VaR_α(`CAT`), for different "small" values of α?

For very small values of α (close to 0.01 or lower), both the empirical and t-copula estimates converge near 1.

This implies a strong tail dependence, meaning that when CAT experiences an extreme loss, IBM is also very likely to experience an extreme loss. For example,

when $\alpha = 0.02$, the probability of IBM Loss Exceeding VaR Given CAT Loss Exceeds VaR is approximately 0.98.

```
# Set alpha values for tail dependence estimation
alpha_seq = seq(0.001, 0.1, by=0.001)

# Compute the empirical tail dependence lambda_emp,L(alpha)
lambda_empirical = lambda(data = cbind(data$IBM, data$CAT), p = alpha_seq, upper=FALSE)

# Compute the tail dependence for the t-copula
best_nu = 5  # Assume best-fit nu is chosen from prior calibration
U_t = sim_T_copula(n = 10000, Sig = Sig, nu = best_nu)

# Estimate lower tail dependence coefficient for the t-copula
lambda_t_copula = lambda(data = U_t, p = alpha_seq, upper=FALSE)

# Plot results
par(mfrow=c(1,1))
plot(alpha_seq, lambda_empirical, type="l", col="blue", lwd=2,
     xlab=expression(alpha), ylab=expression(lambda[L](alpha)),
     main="Lower Tail Dependence: Empirical vs. t-Copula")
lines(alpha_seq, lambda_t_copula, col="red", lwd=2)
legend("topright", legend = c("Empirical Copula", "t-Copula (nu=5)"), col = c("blue", "rec
```



3. **(a)** Solve *(Exercise 1, page 213)* Kendall's τ rank correlation between X and Y is 0.55. Both X and Y are positive. What is Kendall's τ between X and $1/Y$? What is Kendall's τ between $1/X$ and $1/Y$? Explain!

   If $Y_i > Y_j$, then $\frac{1}{Y_i} < \frac{1}{Y_j}$, if Y is positive.

$$\Rightarrow \operatorname{sign}(Y_i - Y_j) = -\operatorname{sign}\left(\frac{1}{Y_i} - \frac{1}{Y_j}\right)$$

$$\Rightarrow \rho_\tau(X, 1/Y) = -\rho_\tau(X, Y)$$

Hence,

$$\rho_\tau(X, 1/Y) = -0.55$$

Similarly,

$$\rho_\tau(1/X, 1/Y) = 0.55$$

(b) Solve *(Exercise 10, page 214)* Suppose that $Y = (Y_1, \ldots, Y_d)^\top$ has a Gaussian copula, but not necessarily Gaussian marginals. Show that if $\rho_\tau(Y_i, Y_j) = 0$, for all $1 \leq i \neq j \leq d$ that $Y_1, \ldots, Y_d$ are independent.

Suppose that $Y = (Y_1, \ldots, Y_d)^\top$ follows a Gaussian copula, but the marginal distributions are not necessarily Gaussian. Given that the Kendall's τ rank correlation between Y_i and Y_j is zero for all $1 \leq i \neq j \leq d$, we will show that $Y_1, \ldots, Y_d$ are independent.

Step 1: Gaussian Copula Definition

A Gaussian copula describes the dependence structure of a random vector where the joint distribution is linked to a multivariate normal distribution. If $Y = (Y_1, ..., Y_d)^\top$ follows a Gaussian copula, then its corresponding uniform variables are:

$$U_i = F_i(Y_i), \quad i = 1, \ldots, d$$

where F_i is the cumulative distribution function (CDF) of Y_i. Transforming to standard normal variables:

$$Z_i = \Phi^{-1}(U_i), \quad i = 1, ..., d$$

ensures that $Z = (Z_1, ..., Z_d)^\top$ follows a multivariate normal distribution with mean 0 and correlation matrix R, where ρ_{ij} is the Pearson correlation coefficient.

Step 2: Kendall's Tau and Correlation in Gaussian Copula

For a Gaussian copula, the relationship between Kendall's τ and Pearson's correlation ρ_{ij} is given by:

$$\rho_\tau(Y_i, Y_j) = \frac{2}{\pi} \arcsin(\rho_{ij}).$$

Since we are given that $\rho_\tau(Y_i, Y_j) = 0$, we solve:

$$\frac{2}{\pi} \arcsin(\rho_{ij}) = 0.$$

This implies:

$$\arcsin(\rho_{ij}) = 0 \quad \Rightarrow \quad \rho_{ij} = 0.$$

Thus, $Z_1, ..., Z_d$ follow a multivariate normal distribution with a diagonal correlation matrix, meaning they are mutually independent.

Step 3: Independence of $Y_1, ..., Y_d$

Since the normal-transformed variables Z_i are independent, their corresponding uniform transformations:

$$U_i = \Phi(Z_i), \quad i = 1, ..., d$$

are also independent. Since the random variables Y_i are obtained from:

$$Y_i = F_i^{-1}(U_i),$$

and F_i^{-1} is an increasing function, the independence of U_i implies that Y_i are also independent. Since Y follows a Gaussian copula, and the Kendall's τ values between all pairs of variables are zero, we conclude that $Y_1, \dots, Y_d$ are independent.

$$\boxed{\rho_\tau(Y_i, Y_j) = 0 \quad \Rightarrow \quad Y_1, \dots, Y_d \text{ are independent.}}$$

4. Let $X = (X_1, \dots, X_d)^\top$ be a random vector with continuous marginal distribution functions $F_i,\ i = 1, \dots, d$. Suppose that C is its copula.

(a) Suppose that $X(1), \dots, X(n)$ are independent realizations of the random vector X and define the random vector of component-wise maxima

$$M(n) := \left(\max_{i=1,\dots,n} X_1(i),\ \max_{i=1,\dots,n} X_2(i),\ \dots,\ \max_{i=1,\dots,n} X_d(i) \right)^\top$$

Show that the copula of $M(n)$ is given by

$$C^n(u_1^{1/n}, u_2^{1/n}, \dots, u_d^{1/n}).$$

Hint: Without loss of generality, you can assume that the distribution functions $F_1, \dots, F_d$ are standard uniform.

Let $X = (X_1, \dots, X_d)^\top$ be a random vector with continuous marginal distribution functions F_i, and let C be its copula.

Step 1: Understanding the Copula Definition

By definition, the copula C of X satisfies:

$$P(X_1 \leq F_1^{-1}(u_1), \dots, X_d \leq F_d^{-1}(u_d)) = C(u_1, \dots, u_d).$$

Given that $X(1), \dots, X(n)$ are independent realizations of X, we define the random vector of component-wise maxima:

$$M(n) = \left(\max_{i=1,\dots,n} X_1(i), \max_{i=1,\dots,n} X_2(i), \dots, \max_{i=1,\dots,n} X_d(i) \right)^\top.$$

Our goal is to derive the copula of $M(n)$.

Step 2: Finding the CDF of $M(n)$

For the component-wise maximum random vector $M(n)$, we have:

$$P(M_1(n) \leq x_1, \ldots, M_d(n) \leq x_d).$$

Since $M_1(n) = \max_{i=1,\ldots,n} X_1(i)$, we can compute:

$$P(M_1(n) \leq x_1) = P(\max_{i=1,\ldots,n} X_1(i) \leq x_1).$$

Using the independence assumption:

$$P(\max_{i=1,\ldots,n} X_1(i) \leq x_1) = P(X_1(1) \leq x_1)^n = F_1^n(x_1).$$

Similarly, for all components:

$$P(M_1(n) \leq x_1, \ldots, M_d(n) \leq x_d) = \left[P(X_1 \leq x_1, \ldots, X_d \leq x_d)\right]^n.$$

Using the copula representation:

$$P(X_1 \leq x_1, \ldots, X_d \leq x_d) = C(F_1(x_1), \ldots, F_d(x_d)),$$

we obtain:

$$P(M_1(n) \leq x_1, \ldots, M_d(n) \leq x_d) = \left[C(F_1(x_1), \ldots, F_d(x_d))\right]^n.$$

Step 3: Expressing the Copula of $M(n)$

The copula C_n of $M(n)$ is defined as:

$$C_n(u_1, \ldots, u_d) = P(M_1(n) \leq F_1^{-1}(u_1), \ldots, M_d(n) \leq F_d^{-1}(u_d)).$$

Using our previous result:

$$C_n(u_1, \ldots, u_d) = \left[C(F_1(F_1^{-1}(u_1)), \ldots, F_d(F_d^{-1}(u_d)))\right]^n.$$

Since $F_i(F_i^{-1}(u_i)) = u_i$, we get:

$$C_n(u_1, \ldots, u_d) = C^n(u_1, \ldots, u_d).$$

However, since each maximum follows the transformed distribution:

$$F_{M_i(n)}(x) = F_i^n(x),$$

this implies that:

$$u_i = F_{M_i(n)}(x) = [F_i(x)]^n.$$

Taking the n-th root,

$$F_i(x) = u_i^{1/n}.$$

Substituting this into our previous result:

$$C_n(u_1, \ldots, u_d) = C^n(u_1^{1/n}, u_2^{1/n}, \ldots, u_d^{1/n}).$$

Conclusion

Thus, we have proven that the copula of $M(n)$ is:

$$C^n(u_1^{1/n}, u_2^{1/n}, \ldots, u_d^{1/n}).$$

(b) Consider the following bivariate copula

$$C(u_1, u_2) := \exp\left\{-2\int_0^1 \max\{x\log(1/u_1), (1-x)\log(1/u_2)\}\sigma(dx)\right\}$$

where σ is such a probability distribution on $[0,1]$ that $\int_0^1 x\sigma(dx) = \int_0^1 (1-x)\sigma(dx) = 1/2$. Prove that

$$C^t(u_1^{1/t}, u_2^{1/t}) = C(u_1, u_2), \quad \text{for all } t > 0 \text{ and } u_1, u_2 \in [0,1].$$

In view of part **(a)**, what can you say about this copula?

From part (a), we established that the copula of the component-wise maxima of n independent realizations of a random vector X is given by $C^n\left(u_1^{1/n}, u_2^{1/n}, \ldots, u_d^{1/n}\right)$. This result is fundamental in extreme value theory, as it describes how copulas transform when taking maxima over increasing sample sizes.

Now, in part (b), we are given a specific copula:

$$C(u_1, u_2) = \exp\left\{-2\int_0^1 \max\{x\log(1/u_1), (1-x)\log(1/u_2)\}\sigma(dx)\right\},$$

and we proved that it satisfies the functional equation $C^t(u_1^{1/t}, u_2^{1/t}) = C(u_1, u_2)$ for all $t > 0$ and $u_1, u_2 \in [0,1]$.

This implies that the given copula is max-stable, meaning that it remains unchanged under component-wise maxima over any sample size t. Since max-stable copulas describe the dependence structure of multivariate extremes, this copula belongs to the class of max-stable copulas. Such copulas arise as the limiting copulas of block maxima and are widely used in extreme value theory to model joint tail dependence in applications such as finance, insurance, and environmental statistics.

Answer

We need to prove that:

$$C^t\left(u_1^{1/t}, u_2^{1/t}\right) = C(u_1, u_2).$$

By definition of the copula function:

$$C(u_1, u_2) := \exp\left\{-2\int_0^1 \max\{x\log(1/u_1), (1-x)\log(1/u_2)\}\sigma(dx)\right\}.$$

Applying the transformation to C^t:

$$C^t\left(u_1^{1/t}, u_2^{1/t}\right) = \left(\exp\left\{-2\int_0^1 \max\left\{x\log\left(\frac{1}{u_1^{1/t}}\right), (1-x)\log\left(\frac{1}{u_2^{1/t}}\right)\right\}\sigma(dx)\right\}\right)^t.$$

Since $\log(1/u^{1/t}) = \frac{1}{t}\log(1/u)$, we factor out $\frac{1}{t}$:

$$= \left(\exp\left\{-\frac{2}{t}\int_0^1 \max\left\{x\log(1/u_1), (1-x)\log(1/u_2)\right\}\sigma(dx)\right\}\right)^t.$$

Now, moving the exponent t inside:

$$= \exp\left\{-2\int_0^1 \max\{x\log(1/u_1), (1-x)\log(1/u_2)\}\sigma(dx)\right\}.$$

Since this is exactly the definition of $C(u_1, u_2)$, we conclude:

$$C^t\left(u_1^{1/t}, u_2^{1/t}\right) = C(u_1, u_2).$$

5. Let $\varphi : (0,1] \to [0,\infty)$ be a strictly decreasing and twice coninuously differentiable convex function such that $\lim_{t\to 1}\varphi(t) = 0$ and $\lim_{t\downarrow 0}\varphi(t) = \infty$.

   (a) Show that
   $$C(u_1, u_2) := \varphi^{-1}(\varphi(u_1) + \varphi(u_2)), \quad (u_1, u_2) \in [0,1]^2$$
   is a valid cumulative distribution function.

   Hint: You should verify that C satisfies the conditions for being a valid CDF with uniform marginals. The hardest part is to verify that $\Delta C := C(u_1, u_2) - C(u_1, v_2) - C(v_1, u_2) + C(v_1, v_2) \geq 0$, for every choice of $(u_1, u_2) \leq (v_1, v_2)$ in $[0,1]^2$, where the last inequality is taken component-wise. You can verify that by showing that $\partial^2_{u_1,u_2} C(u_1, u_2) \geq 0$, for all $(u_1, u_2) \in [0,1]^2$.

Proof of Copula Validity

Step 1: Uniform Marginals

We verify that $C(u,1) = u$ and $C(1,v) = v$.

$$\text{Since } \varphi(1) = 0, \text{ we have } C(u,1) = \varphi^{-1}(\varphi(u) + \varphi(1)) = \varphi^{-1}(\varphi(u)) = u.$$

Similarly,

$$C(1,v) = \varphi^{-1}(\varphi(v)) = v.$$

Step 2: Monotonicity

$$\frac{\partial C}{\partial u} = (\varphi^{-1})'\big(\varphi(u_1) + \varphi(u_2)\big) \cdot \varphi'(u_1).$$

Since φ is strictly decreasing, it follows that $\varphi' < 0$. Also, since φ^{-1} is decreasing (to be proven), we have $(\varphi^{-1})' < 0$. Therefore:

$$\frac{\partial C}{\partial u} \geq 0 \text{ Similarly, } \frac{\partial C}{\partial v} \geq 0$$

Step 3: 2-Increasing Property

We verify that:

$$\frac{\partial^2 C}{\partial u \partial v} \geq 0.$$

Computing the second derivative,

$$\frac{\partial^2 C}{\partial u_1 \partial u_2} = (\varphi^{-1})''\big(\varphi(u_1) + \varphi(u_2)\big) \cdot \varphi'(u_1)\varphi'(u_2).$$

Since $\varphi' < 0$ and $(\varphi^{-1})'' \geq 0$ (due to convexity), we conclude:

$$\frac{\partial^2 C}{\partial u_1 \partial u_2} \geq 0.$$

Step 4: Proof that φ^{-1} is Decreasing

We need to show that if φ is strictly decreasing, then φ^{-1} is also decreasing.

Define $\varphi : X \to Y$, where φ is strictly decreasing. Suppose $y_1, y_2 \in Y$ with $y_1 < y_2$. Then there exist $x_1, x_2 \in X$ such that:

$$x_1 = \varphi^{-1}(y_1), \quad x_2 = \varphi^{-1}(y_2) \quad \Rightarrow \quad \varphi(x_1) = y_1, \quad \varphi(x_2) = y_2.$$

If $\varphi^{-1}(y_1) \leq \varphi^{-1}(y_2)$, then $x_1 \leq x_2$, which implies (since φ is strictly decreasing):

$$\varphi(x_1) \geq \varphi(x_2) \quad \Rightarrow \quad y_1 \geq y_2$$

which is a contradiction. Hence, φ^{-1} is also strictly decreasing.

Step 5: Convexity of φ^{-1}

By basic calculus, we have the formula for the second derivative of an inverse function:

$$(\varphi^{-1})'' = -\frac{\varphi''}{(\varphi')^3}.$$

Since φ is convex, $\varphi'' \geq 0$ and $\varphi' \leq 0$, which implies: $(\varphi^{-1})'' \geq 0$

Thus, φ^{-1} is convex, completing the proof.

(b) Suppose that (X_1, X_2) have a bivariate copula C. Determine the copula C_Y of $(Y_1, Y_2) := (-X_1, X_2)$ and the copula C_Z of $(Z_1, Z_2) := (-X_1, -X_2)$.

Hint: Suppose, without loss of generality, that X_1 and X_2 have uniform marginal distributions. The copula of (Y_1, Y_2) and $(\widetilde{Y}_1, \widetilde{Y}_2) := (1 - X_1, X_2)$ are the same, but notice that $\widetilde{Y}_1$ and $\widetilde{Y}_2$ have Uniform$(0, 1)$ distributions. Use C to express

$$C_Y(u_1, u_2) = \mathbb{P}(\widetilde{Y}_1 \leq u_1, \widetilde{Y}_2 \leq u_2) = \mathbb{P}(1 - X_1 \leq u_1, X_2 \leq u_2).$$

Copula C_Y for $(Y_1, Y_2) = (-X_1, X_2)$

We aim to determine the copula $C_Y(u_1, u_2)$, which corresponds to the transformation:

$$C_Y(u_1, u_2) = P(X_1 \geq 1 - u_1, X_2 \leq u_2).$$

Using probability properties:

$$P(X_1 \geq 1 - u_1, X_2 \leq u_2) = 1 - P(X_1 \leq 1 - u_1, X_2 \leq u_2).$$

From the definition of the copula C, we substitute:

$$C_Y(u_1, u_2) = 1 - C(1 - u_1, u_2).$$

Copula C_Z for $(Z_1, Z_2) = (-X_1, -X_2)$

Now, we analyze:

$$C_Z(u_1, u_2) = P(-X_1 \leq u_1, -X_2 \leq u_2).$$

Rewriting the inequality:

$$C_Z(u_1, u_2) = P(X_1 \geq 1 - u_1, X_2 \geq 1 - u_2).$$

Using probability properties:

$$P(X_1 \geq 1-u_1, X_2 \geq 1-u_2) = 1-P(X_1 \leq 1-u_1)-P(X_2 \leq 1-u_2)+P(X_1 \leq 1-u_1, X_2 \leq 1-u_2).$$

Substituting the copula definition:

$$C_Z(u_1, u_2) = 1 - (1 - u_1) - (1 - u_2) + C(1 - u_1, 1 - u_2).$$

Simplifying:

$$C_Z(u_1, u_2) = u_1 + u_2 - 1 + C(1 - u_1, 1 - u_2).$$