

UNIVERSIDADE FEDERAL DE SÃO PAULO
ESCOLA PAULISTA DE POLÍTICA, ECONOMIA E NEGÓCIOS
DEPARTAMENTO DE CIÊNCIAS ATUARIAIS
GRADUAÇÃO EM CIÊNCIAS ATUARIAIS

FELIPE EDUARDO SOARES DE ALMEIDA

ROTEIRO PRÁTICO DE APLICAÇÃO DE CÓPULA PARA O
CÁLCULO DE DEPENDÊNCIA UTILIZANDO A LINGUAGEM R

Osasco

2021

FELIPE EDUARDO SOARES DE ALMEIDA

**ROTEIRO PRÁTICO DE APLICAÇÃO DE CÓPULA PARA O
CÁLCULO DE DEPENDÊNCIA UTILIZANDO A LINGUAGEM R**

Trabalho de conclusão de curso apresentado à Escola Paulista de Política, Economia e Negócios da **Universidade Federal de São Paulo** como parte dos requisitos para obtenção do título de bacharel em Ciências Atuariais.

Orientador: Prof. Dr. Luiz Jurandir Simões de Araújo

Osasco

2021

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional ou eletrônico, para fins de estudo e pesquisa, desde que citada a fonte.

Ficha catalográfica elaborada pela Biblioteca Unifesp Osasco
e Departamento de Tecnologia da Informação Unifesp Osasco,
com os dados fornecidos pelo(a) autor(a)

A447r ALMEIDA, Felipe Eduardo Soares de
Roteiro prático de aplicação de cópula para o cálculo de
dependência utilizando a linguagem R / Felipe Eduardo Soares
de Almeida. - 2021.
47 f.

Trabalho de conclusão de curso (Ciências Atuariais) -
Universidade Federal de São Paulo - Escola Paulista de Política,
Economia e Negócios, Osasco, 2021.

Orientador: Prof. Dr. Luiz Jurandir Simões de Araújo.

1. Cópulas. 2. Ajuste de distribuição. 3. Ajuste de cópula. I.
Araújo, Prof. Dr. Luiz Jurandir Simões de , II. TCC -
Unifesp/EPPEN. III. Título.

CDD: 368.01

ALMEIDA, Felipe E.S. **Roteiro prático de aplicação de cópula para o cálculo de dependência utilizando a linguagem R.** Trabalho de conclusão de curso apresentado à Escola Paulista de Política, Economia e Negócios da Universidade Federal de São Paulo como parte dos requisitos para obtenção do título de bacharel em Ciências Atuariais.

Aprovado em: __ / __ / ____

Banca Examinadora

Prof. Dr. Luiz Jurandir Simões de Araújo

Instituição: Universidade Federal de São Paulo

Julgamento _____ Assinatura _____

Prof. Dr. Wanderlei Lima de Paulo

Instituição: Universidade de São Paulo

Julgamento _____ Assinatura _____

Dedicatória

Dedico este trabalho aos meus pais, Eduardo e Solange, que lutaram para que eu tivesse boas condições de estudo, ao Prof. Dr. Antonio Cordeiro Filho e meus amigos Natanael e Ana, que me incentivaram a ingressar neste curso na UNIFESP e ao Prof. Dr. Luiz Jurandir Simões de Araújo, orientador deste trabalho.

Agradecimentos

Em primeiro lugar agradeço a Deus e Meishu-Sama pela força e pela saúde física e mental que me possibilitaram realizar este trabalho.

Agradeço aos meus pais, Eduardo e Solange, e aos meus tios, Fábio e Maria, por todo suporte que me deram durante minha jornada na Unifesp.

Agradeço à minha parceira, Thayná, por todo suporte quando tive momentos difíceis conciliando trabalho e estudos.

Agradeço meus amigos Gabriel, Igor, Isabella, Maria Luiza, Petrotielli e Thayná, e a todos os amigos que a Unifesp me deu – meus dias foram mais divertidos convosco.

Finalmente, agradeço todo o corpo docente da Unifesp, porém citando nominalmente os professores Antônio Cordeiro, Arthur Weintraub, Celso Yokomiso, Danilo Braun, Edimilson Lucas, Joelson Sampaio, Luiz Maluf, Márcio Catapani, Raphael Garcia, Ricardo Ikeda, Roberto Cazzari e o Prof. Dr. Luiz Jurandir Simões de Araújo, que aceitou me orientar no mais importante projeto do curso.

“Our science is a drop, our ignorance a sea”

(William James)

Resumo

As cópulas, por definição de Nelsen (2006), são funções de distribuição multivariadas cujas margens unidimensionais são uniformes no intervalo (0, 1). Isto é, as cópulas ligam duas ou mais distribuições marginais univariadas, uniformes no intervalo (0, 1), a uma distribuição multivariada. Assim, torna-se possível separar uma distribuição multivariada em suas distribuições marginais e uma cópula, representando a estrutura de dependência entre as variáveis analisadas.

O presente trabalho tem como objetivo encontrar, utilizando a linguagem R, as cópulas que melhor representam a estrutura de dependência entre duas variáveis aleatórias. Para este trabalho usaremos como exemplo ações e units negociadas na B³, bolsa de valores oficial do Brasil. Para tanto, extrairemos informações diárias de três índices informados pela B³: IFNC B3 (Índice BM&FBOVESPA Financeiro), INDX B3 (Índice do Setor Industrial) e ICON B3 (Índice de Consumo). Então, com os devidos procedimentos matemáticos, encontraremos as distribuições marginais e as cópulas bidimensionais cruzando todos os índices.

Palavras-chave: Cópulas; Ajuste de distribuição; Ajuste de cópula.

Abstract

Copulas, as defined by Nelsen (2006), are multivariate distribution functions whose one-dimensional margins are uniform in the interval (0, 1). I.e., copulas connect two or more univariate marginal distributions, uniform in the interval (0, 1), to a multivariate distribution. Thus, it becomes possible to separate a multivariate distribution into its marginal distributions and a copula, representing the dependency structure between the analysed variables.

This paper aims to find, using R language, the copulas that best represent the dependency structure between two random variables. For this work, we will use as an example shares and units traded on B³, the official stock exchange in Brazil. To do so, we will extract daily data from three indexes reported by B³: IFNC B3 (BM&FBOVESPA Financial Index), INDX B3 (Industry Sector Index) and ICON B3 (Consumption Index). Then, with the proper mathematical procedures, we will find the marginal distributions and the two-dimensional copulas across all indexes.

Keywords: Copulas; Distribution fitting; Copula fitting.

SUMÁRIO

1	Introdução.....	8
2	Tema do Estudo.....	9
3	Cópuas.....	10
3.1	Definição.....	10
3.2	Teorema de Sklar (1959).....	10
3.3	Limites de Fréchet-Hoeffding.....	11
3.4	Cópuas de Sobrevivência (Survival Copulas).....	11
3.5	Rotação.....	12
3.6	Famílias de Cópuas.....	12
3.6.1	Cópula Normal ou Gaussiana.....	12
3.6.2	Cópula t de Student.....	12
3.6.3	Cópula Gumbel.....	13
3.6.4	Cópula Clayton.....	14
3.6.5	Cópula Frank.....	15
3.6.6	Cópula Ali-Mikhail-Haq.....	16
3.6.7	Cópula BB1.....	16
3.6.8	Cópula BB7.....	17
3.6.9	Cópula Tawn.....	18
3.6.10	Cópuas de Valores Extremos.....	18
4	Roteiro prático para o ajuste de cópuas utilizando o R.....	19
4.1	Fluxograma.....	19
4.2	Extração e análise dos dados.....	19
4.3	Ajuste da Distribuição de Probabilidade.....	24
4.4	Ajuste de Cópuas.....	30
4.4.1	Pseudo-observações.....	30
4.4.2	Cópuas a partir da distribuição empírica.....	30
4.4.3	Cópuas a partir da distribuição teórica.....	31
5	Resultados.....	32
6	Conclusão.....	35
	Referências Bibliográficas.....	36
	Apêndice.....	37

1 Introdução

O tema escolhido para este trabalho é da área de probabilidade/estatística, e aplicável em diversas áreas, uma vez que os resultados servem de insumos para análises e cálculos diversos. Mais especificamente, esse trabalho aborda a aplicação de técnicas de ajuste de distribuições de probabilidade e de ajuste para cópulas.

As cópulas, de acordo com Nelsen (2006), são “funções que acoplam funções de distribuição multivariada a suas funções de distribuição marginal unidimensionais” ou “funções de distribuição multivariadas cujas margens unidimensionais são uniformes no intervalo $(0, 1)$ ”. Ou seja, uma cópula de n dimensões terá n distribuições marginais, sendo que cada uma delas é uniforme no intervalo $(0,1)$.

Uma definição mais pedagógica é feita por Haugh (2016), que afirma que cópulas são funções que nos permitem separar as distribuições marginais da estrutura de dependência de uma determinada distribuição multivariada. Ou seja, como a cópula é uma função que liga distribuições marginais univariadas para formar uma distribuição conjunta.

De acordo com Dall’Aglío et al. (1991), o uso de cópulas no campo da probabilidade teve início com os trabalhos de Fréchet (1951 apud DALL’AGLIO) e do próprio Dall’Aglío (1959 apud DALL’AGLIO), mas os primeiros passos neste campo foram dados pela Escola Romana de estatística. Fréchet e Dall’Aglío fizeram estudos de funções bivariadas e trivariadas, com determinadas marginais univariadas. Sklar (1959 apud DALL’AGLIO) solucionou o problema das marginais univariadas com a criação das cópulas. Pulando para o século atual, o uso das cópulas é mais disseminado e diverso. Elas são uma ferramenta extremamente interessante por permitir o estudo de medidas de dependência sem limitações comuns a outros métodos, como Pearson, Spearman e Kendall, e por ser o ponto de partida para a construção de famílias de distribuições multivariadas. Dentre vários usos, destacam-se para este trabalho dois usos no mercado financeiro (muito por conta do interesse do mercado financeiro em temas como movimento browniano e martingale): modelagem de risco sistêmico (Low, 2017) e precificação de opções (Cherubini et. al, 2004).

No entanto, utiliza-se muito a cópula gaussiana no mercado financeiro, o que já se provou perigoso em 2008, quando um modelo utilizando cópulas gaussianas para precificar Obrigações de Dívidas Colateralizadas (CDO), implementado no começo do século XXI, se provou uma “receita para o desastre” na crise do *subprime*. Hoje há estudos que utilizam cópulas Arquimedianas para fazer essa precificação, como pode ser visto em Hofert e Scherer (2008) e Gaiduchevici (2014).

As distribuições marginais das cópulas, antes da transformação para o intervalo (0,1) são dadas ou devem ser encontradas a partir da distribuição empírica dos dados via ajuste de distribuição (mais conhecido como *fit* de distribuição).

Existem diversas técnicas para o ajuste de distribuições de probabilidade. O indivíduo pode verificar estatísticas descritivas como média, moda, mediana, assimetria e curtose e, junto com um gráfico de densidade, inferir os tipos ou famílias de distribuições que mais se ajustam aos dados. Essa forma é, no entanto, imprecisa e só funcionará – e de forma grosseira – quando nos depararmos com dados que seguem alguma distribuição padrão. Utilizando dados reais, essa tarefa torna-se praticamente impossível. No entanto, hoje, esse trabalho pode ser facilitado com o uso de softwares.

Com isso, é possível considerar que o método da máxima verossimilhança é o mais utilizado hoje, disponível nos principais softwares e linguagens de programação como R, Python, Julia, SAS, SPSS etc. Os procedimentos para ajuste de cópulas são igualmente facilitados pela tecnologia.

Neste trabalho utilizaremos índices da B³ como exemplo. Os índices utilizados são os chamados Índices de Retorno Total que, além das variações nos preços dos ativos, também refletem o impacto da distribuição de proventos por parte das companhias emissoras.

2 Tema do Estudo

Para este trabalho, o tema escolhido foi o *fit* de cópulas. Os dados dos índices da B³ foram selecionados como exemplo uma vez que o mercado financeiro faz parte da seara das ciências atuariais. As cópulas são muito utilizadas no mercado financeiro em gestão de risco, gestão de portfólios, otimização de portfólios e precificação de derivativos, além do uso em análises atuariais.

No início dos anos 2000 houve um aumento nos estudos do uso de cópulas para precificação de opções, como visto nos estudos de Cherubini e Luciano (2002) e Goorbergh, Genest e Werker (2004). Esses autores publicaram sobre a precificação de opções sobre mais de um ativo subjacente, sendo que o trabalho de Cherubini e Luciano levou a publicação de um livro em 2004, onde descrevem o uso de cópulas como “a nova ferramenta mais significativa para lidar de forma flexível com o comovimento entre mercados, fatores de risco e outras variáveis relevantes” e afirmam que os novos estudos tornaram métodos amplamente utilizados, como Black-Scholes, “obsoletos”. No entanto, apesar de ser um tema pertinente ao atuário, não é um tema comumente ensinado em cursos de graduação no Brasil e há pouco material sobre esse tema na língua portuguesa.

Os ajustes de distribuições fazem parte da vida do atuário e, portanto, serão utilizados como ferramenta para a procura das cópulas.

3 Cópulas

Apresentaremos a definição e propriedades básicas das cópulas. Neste trabalho utilizaremos apenas cópulas bivariadas, ou seja, bidimensionais, uma vez que o tempo de execução dos códigos para o caso d-dimensional quando $d > 2$ é demasiado elevado. As cópulas são apresentadas em formato d-dimensional, e então as cópulas para os casos bivariados, os mais simples, podem ser encontradas fazendo a substituição $d = 2$.

A seguir, após apresentar definições e propriedades, apresentaremos algumas cópulas.

3.1 Definição

Por definição, $C : [0,1]^d \rightarrow [0,1]$ é uma cópula d-dimensional se C é a função distribuição acumulada conjunta de um vetor aleatório d-dimensional num cubo unitário $[0,1]^d$ com distribuições marginais uniformes.

$C : [0,1]^d \rightarrow [0,1]$ é uma cópula d-dimensional se:

- i. $C(u_1, \dots, u_{i-1}, 0, u_{i+1}, \dots, u_d) = 0$ (A cópula é igual a zero se qualquer um dos argumentos é zero);
- ii. $C(1, \dots, 1, u, 1, \dots, 1) = u$ (A cópula é igual a u se todos os outros elementos são iguais a 1) e;
- iii. C é d-crescente, ou seja, todo $B = \times_{i=1}^d [x_i, y_i] \subseteq [0,1]^d$ possui C-volume maior ou igual a 0, com C-volume definido por

$$\sum_{z \in \times_{i=1}^d \{x_i, y_i\}} (-1)^{\#\{k | z_k = x_k\}} c_z.$$

3.2 Teorema de Sklar (1959)

Em sua publicação “*Fonctions de répartition à n dimensions et leurs marges*”, publicada em 1959, Sklar provou o teorema que leva seu nome: o Teorema de Sklar. O teorema, aqui apresentado na forma bivariada para facilitar a compreensão, considera o seguinte: Seja H uma função de distribuição bidimensional com funções de distribuição marginal F e G . Então existe uma cópula $C : [0, 1]^2 \rightarrow [0, 1]$ tal que

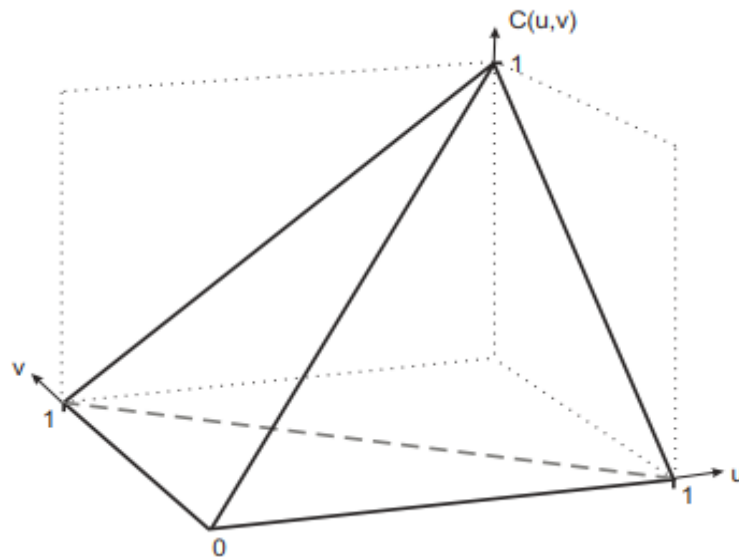
$$H(x, y) = C(F(x), G(y)), \forall (x, y) \in [-\infty, \infty]^2$$

Se $F(x)$ e $G(y)$ são contínuas, então C é única. Caso contrário, C é unicamente determinada em $\text{Im}(F) \times \text{Im}(G)$, que é o produto cartesiano entre as imagens das distribuições marginais. A recíproca também é verdadeira.

3.3 Limites de Fréchet-Hoeffding

Os limites de Fréchet-Hoeffding foram nomeados em homenagem ao matemático francês Fréchet e ao estatístico finlandês Hoeffding. De acordo com esses limites, cada cópula deve estar contida dentro da pirâmide destacada na Figura 1:

Figura 1 – Limites de Fréchet-Hoeffding



Fonte: Thorsten Schmidt, 2007

A superfície dada pela parte de baixo e a parte de trás da pirâmide é o limite inferior, que pode ser descrito como $C(u, v) = \max \{u + v - 1, 0\}$, enquanto o limite superior, a parte da frente da pirâmide, pode ser descrito como $C(u, v) = \min (u, v)$.

Então, temos para qualquer cópula $C: [0, 1]^2 \rightarrow [0, 1]$ com $(u, v) \in [0, 1]^2$ que $\max \{u + v - 1, 0\} \leq C(u, v) \leq \min (u, v)$. A definição para o caso multivariado: $\max (1 - d + \sum_{i=1}^d u_i) \leq C(u_1, u_2, \dots, u_d) \leq \min (u_1, u_2, \dots, u_d)$. As provas podem ser encontradas em Haugh (2016).

3.4 Cópulas de Sobrevivência (Survival Copulas)

A cópula de sobrevivência (relacionada à função de sobrevivência) associada a uma cópula C pode ser descrita como:

$$\hat{C}(u, v) = 1 - u - v + C(u, v) = \hat{C}(1 - u, 1 - v)$$

3.5 Rotação

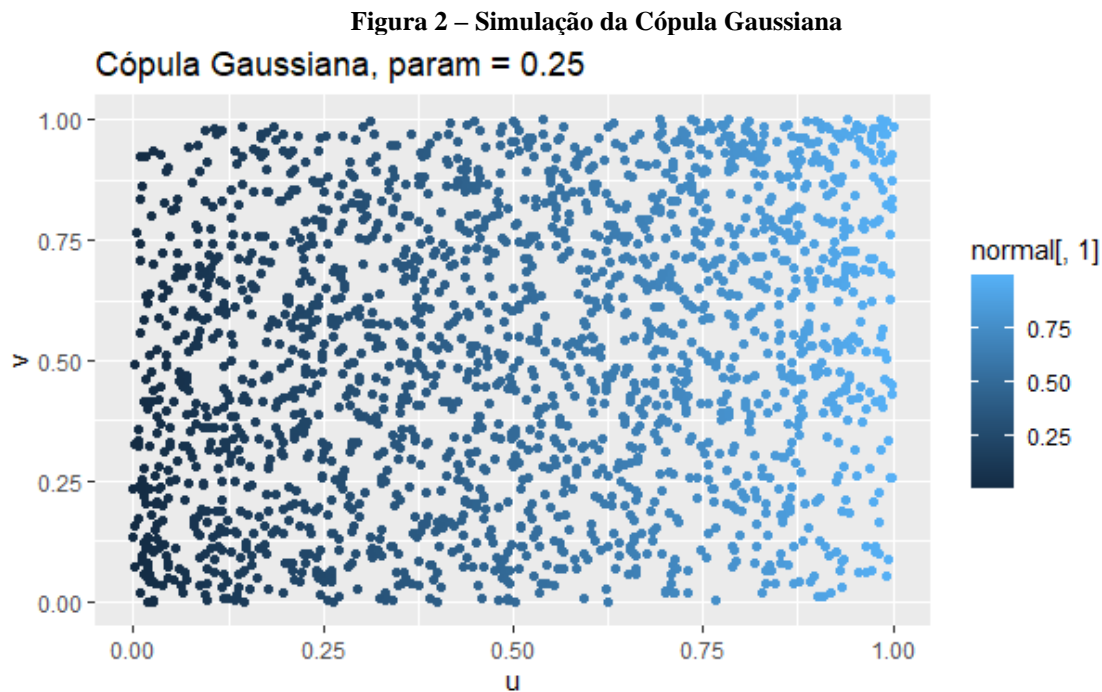
Para cópulas bivariadas é possível transformar uma ou duas variáveis u, v em $1 - u$ ou $1 - v$, rotacionando a cópula em 90° , 180° ou 270° .

3.6 Famílias de Cópulas

3.6.1 Cópula Normal ou Gaussiana

Seja R uma matriz positiva definida e simétrica, com $diag(R) = (1, 1, \dots, 1)^T$ e Φ_R uma distribuição normal multivariada padrão com matriz de correlação R . A cópula é definida por:

$C_R^{Ga}(u) = \Phi_R(\Phi^{-1}(u_1), \dots, \Phi^{-1}(u_d))$, com Φ^{-1} sendo a inversa de Φ , a distribuição normal padrão univariada. Para essa cópula, se as distribuições marginais seguem uma distribuição normal padrão, então é gerada a distribuição normal multivariada. Apresentamos na Figura 2 uma simulação da Cópula Gaussiana:

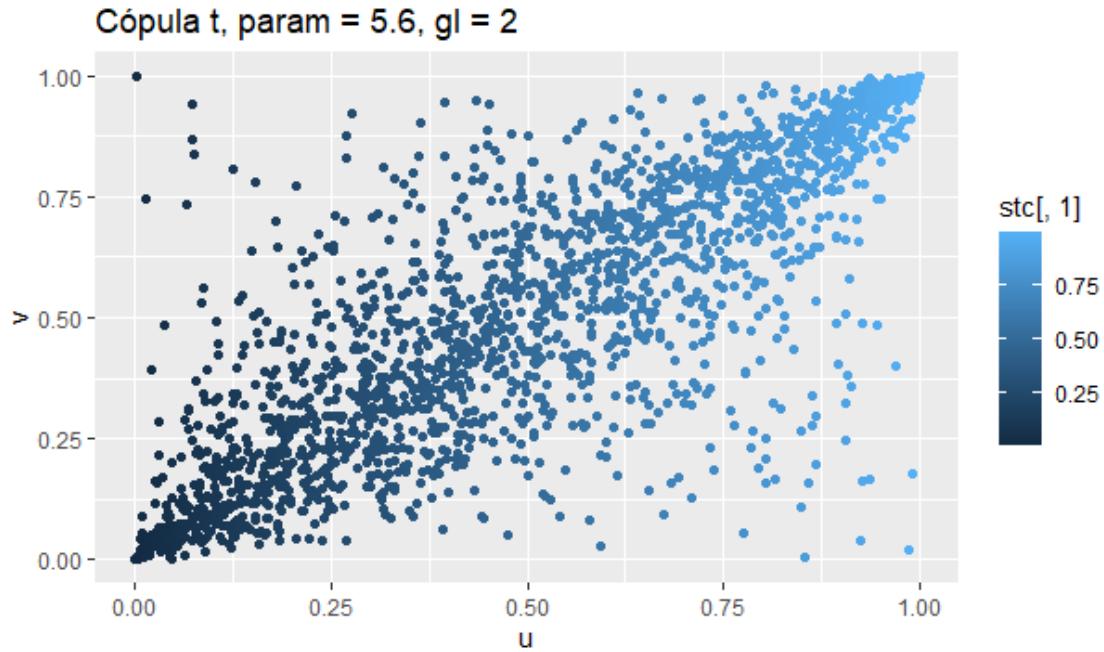


3.6.2 Cópula t de Student

Seja R uma matriz positiva definida e simétrica, com $diag(R) = (1, 1, \dots, 1)^T$ e $t_{R, \nu}$ uma distribuição t de Student multivariada padrão com matriz de correlação R e ν graus de liberdade. A cópula é definida por:

$T_{R,v}(u_1, u_2, \dots, u_d) = t_{R,v}(t_v^{-1}(u_1), t_v^{-1}(u_2), \dots, t_v^{-1}(u_d))$, com t_v^{-1} sendo a inversa da distribuição t de Student padrão univariada. Apresentamos na Figura 3 uma simulação da Cópula t de Student:

Figura 3 – Simulação da Cópula t de Student



Fonte: Elaboração própria

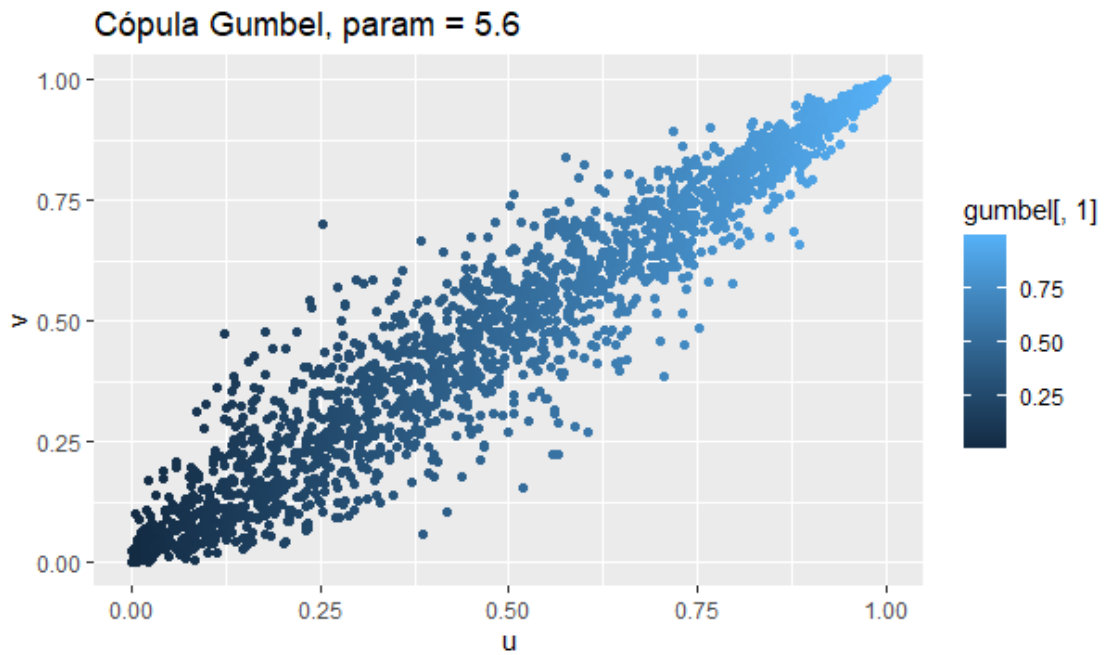
3.6.3 Cópula Gumbel

É também conhecida como cópula de Gumbel-Hougaard. Essa cópula é assimétrica e demonstra maior dependência na cauda positiva. A cópula é definida por:

$$C(u_1, u_2, \dots, u_d) = \exp\left\{-\left[\sum_{i=1}^d (-\ln u_i)^\alpha\right]^{\frac{1}{\alpha}}\right\}, \text{ com } \alpha > 1$$

Apresentamos na Figura 4 uma simulação da Cópula Gumbel:

Figura 4 – Simulação da Cópula Gumbel



Fonte: Elaboração própria

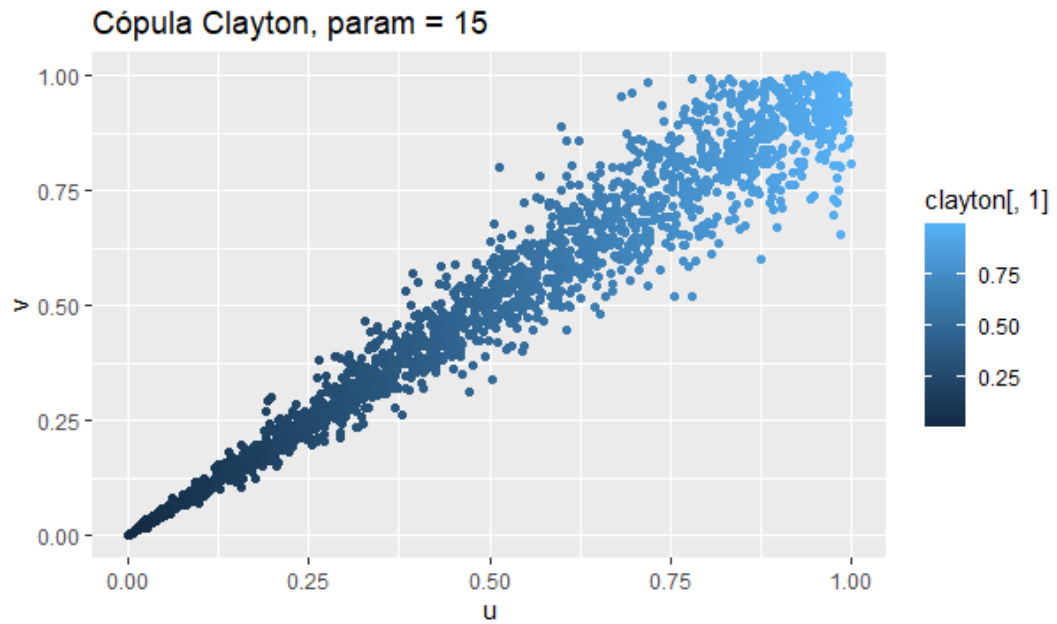
3.6.4 Cópula Clayton

É também assimétrica, como Gumbel, e demonstra maior dependência na cauda negativa. A cópula é definida por:

$$C(u_1, u_2, \dots, u_d) = \left[\sum_{i=1}^d u_i^{-\alpha} - n + 1 \right]^{-\frac{1}{\alpha}}, \text{ com } \alpha > 0$$

Apresentamos na Figura 5 uma simulação da Cópula Clayton:

Figura 5 – Simulação da Cópula Clayton



Fonte: Elaboração própria

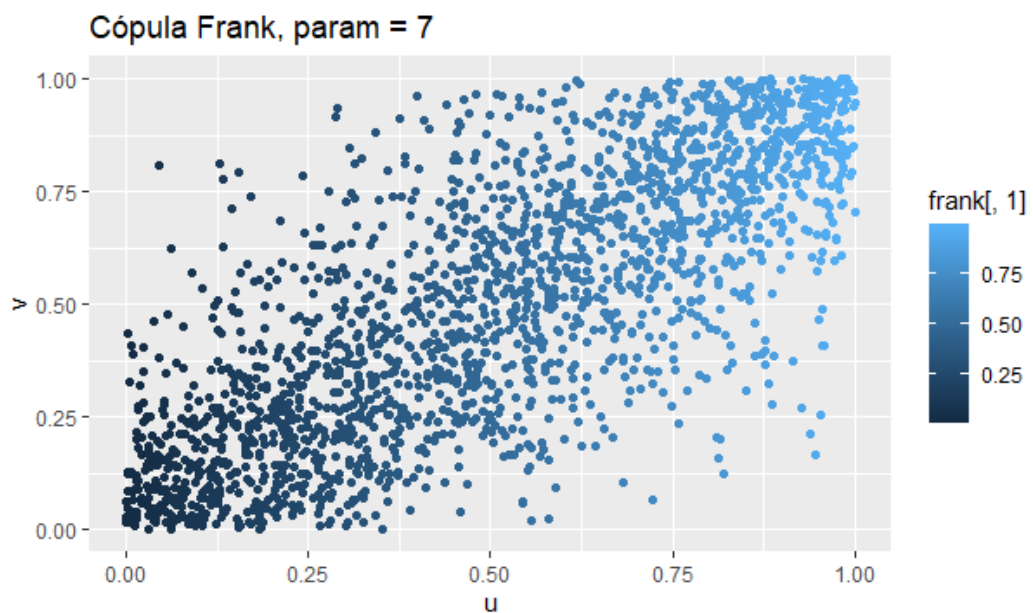
3.6.5 Cópula Frank

É uma cópula simétrica apenas no caso bidimensional. A cópula é definida por:

$$C(u_1, u_2, \dots, u_d) = -\frac{1}{\alpha} \ln \left\{ 1 + \frac{\prod_{i=1}^d (e^{-\alpha u_i} - 1)}{(e^{-\alpha} - 1)^{d-1}} \right\}, \text{ com } \alpha > 0 \text{ quando } d \geq 3. \text{ Apresentamos na}$$

Figura 6 uma simulação da Cópula Frank:

Figura 6 – Simulação da Cópula Frank



Fonte: Elaboração própria

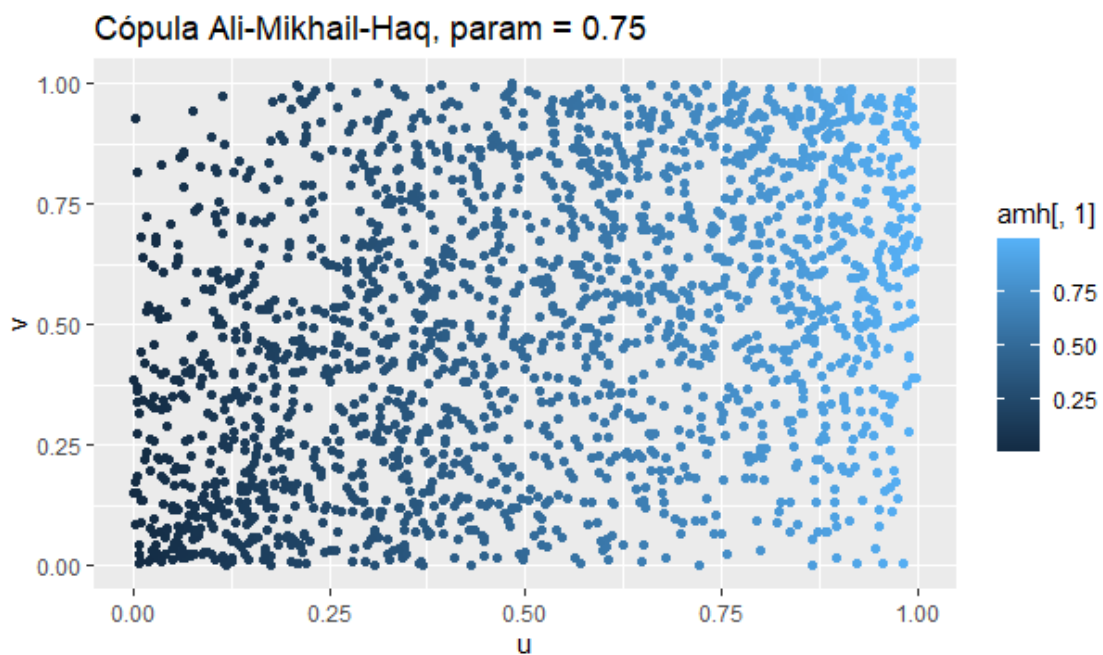
3.6.6 Cópula Ali-Mikhail-Haq

É nomeada em homenagem aos matemáticos Ali, Mikhail e Haq. Sua construção é baseada no conceito de *odds for survival*. A cópula é definida por:

$$C(u_1, u_2, \dots, u_d) = \frac{\prod_{i=1}^d u_i}{1 - \alpha \prod_{i=1}^d (1 - u_i)}, \text{ com } -1 \leq \alpha \leq 1.$$

Apresentamos na Figura 7 uma simulação da Cópula Ali-Mikhail-Haq:

Figura 7 – Simulação da Cópula Ali-Mikhail-Haq



Fonte: Elaboração própria

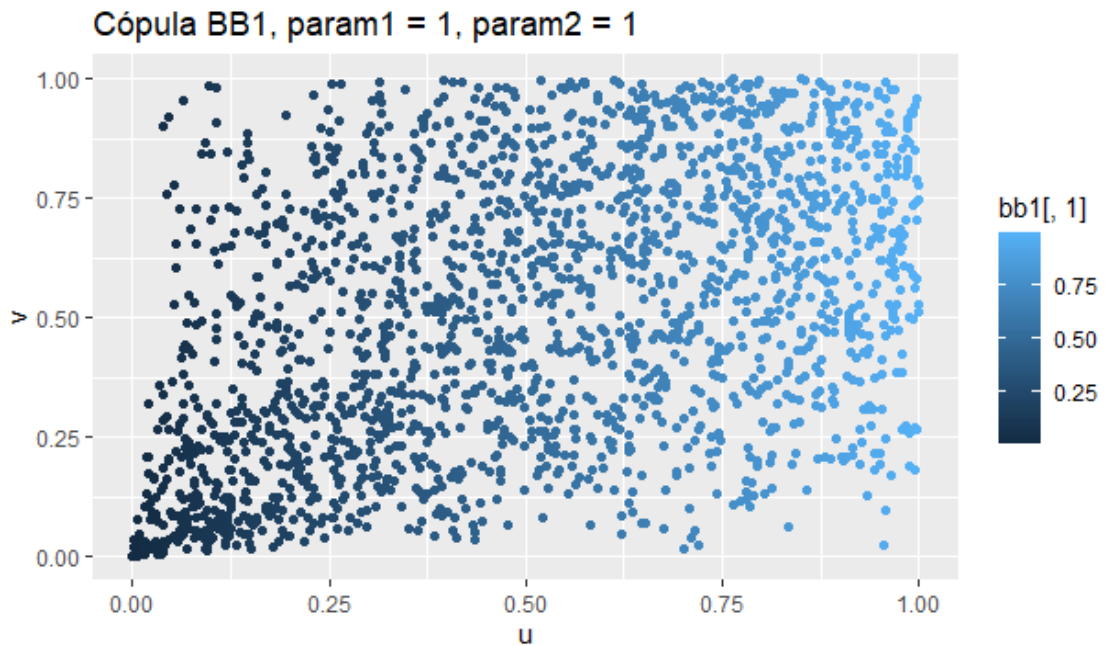
3.6.7 Cópula BB1

É uma cópula bivariable e bivariable definida por Joe (1997). É definida por:

$$C(u, v; \alpha, \delta) = \left\{ 1 + [(u^{-\alpha} - 1)^\delta + (v^{-\alpha} - 1)^\delta]^{\frac{1}{\delta}} \right\}^{-\frac{1}{\delta}}.$$

Apresentamos na Figura 8 uma simulação da Cópula BB1:

Figura 8 – Simulação da Cópula BB1



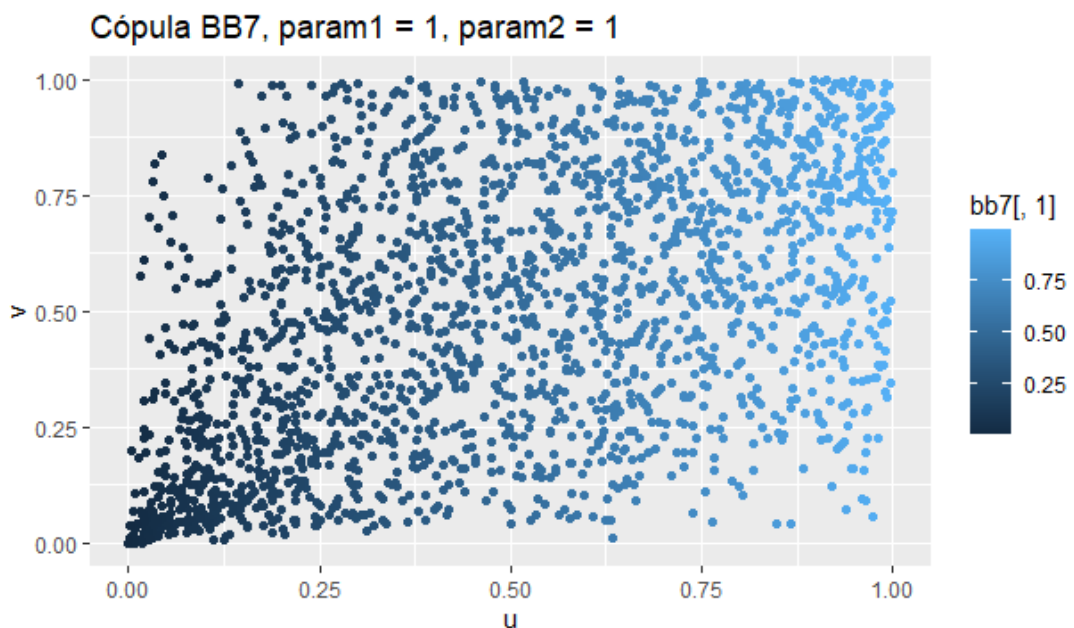
Fonte: Elaboração própria

3.6.8 Cópula BB7

É uma cópula biparamétrica e bivariada definida por Joe (1997). É definida por:

$C(u, v; \alpha, \delta) = 1 - (1 - [(1 - \bar{u}^\alpha)^{-\delta} + (1 - \bar{v}^\alpha)^{-\delta} - 1]^{-\frac{1}{\delta}})^{\frac{1}{\alpha}}$. Apresentamos na Figura 9 uma simulação da Cópula BB7:

Figura 9 – Simulação da Cópula BB7



Fonte: Elaboração própria

3.6.9 Cópula Tawn

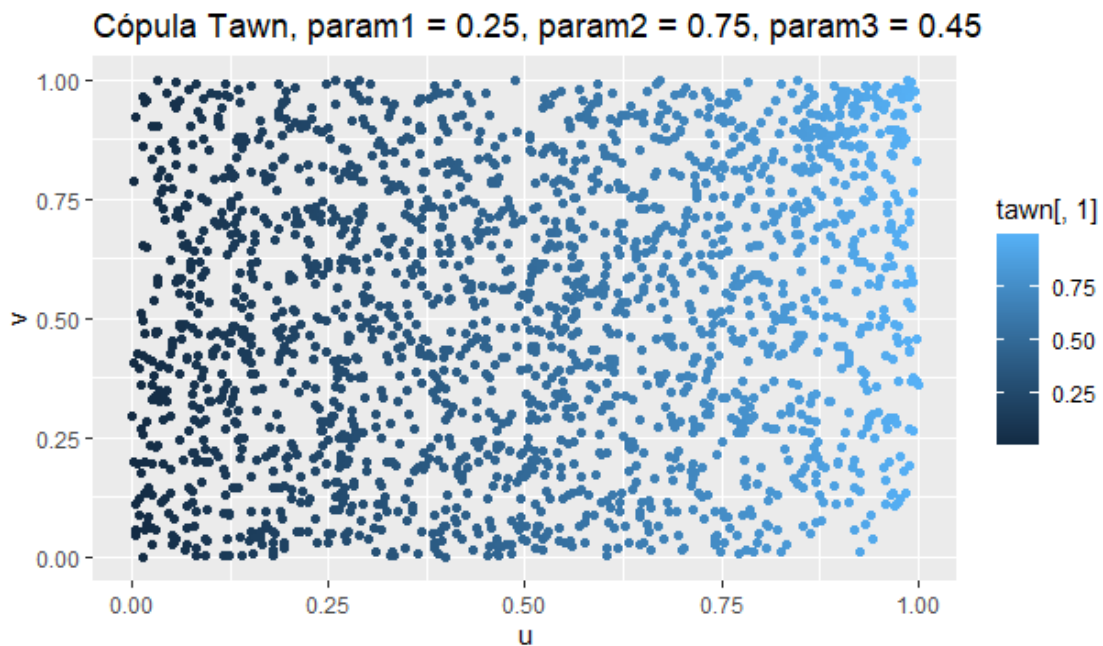
É uma cópula definida por Tawn (1988). É uma extensão da cópula Gumbel, porém com três parâmetros: α , δ_1 e δ_2 . É definida por:

$$C^{T\alpha}(u, v; \alpha, \delta_1, \delta_2) = (1 - \delta_1)t - (1 - \delta_2)(1 - t) + [(\delta_1(1 - t))^\alpha + (\delta_2 t)^\alpha]^{\frac{1}{\alpha}},$$

onde $t = \frac{\log(v)}{\log(uv)}$, $0 \leq \delta_1, \delta_2 \leq 1$ e $\alpha \in [1, \infty_+)$.

Se $\delta_1 = \delta_2$, então obtemos a cópula de Gumbel. É importante dizer que os parâmetros δ_1, δ_2 podem ser considerados como parâmetros de assimetria. Assim, podemos dividir a cópula de Tawn em dois tipos: “Tipo 1”, com $\delta_1 = 1$ (assimetria negativa) e “Tipo 2”, com $\delta_2 = 1$ (assimetria positiva). Apresentamos na Figura 10 uma simulação da Cópula Tawn:

Figura 10 – Simulação da Cópula Tawn



Fonte: Elaboração própria

3.6.10 Cópulas de Valores Extremos

Uma cópula C_* é uma cópula de valor extremo se existe uma cópula C tal que $C_*(u, v) = \lim_{n \rightarrow \infty} C^n\left(u^{\frac{1}{n}}, v^{\frac{1}{n}}\right)$ e $C(u, v) = \exp\left\{\ln(uv) A\left(\frac{\ln(v)}{\ln(uv)}\right)\right\}$.

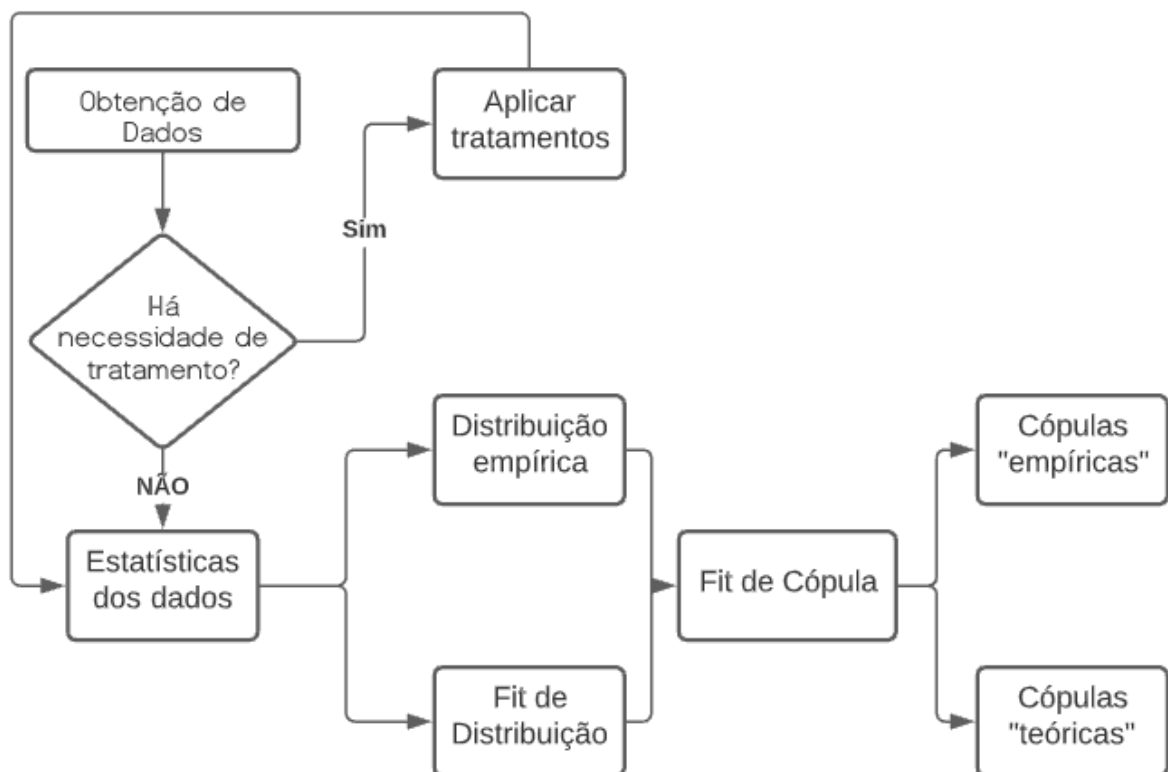
4 Roteiro prático para o ajuste de cópulas utilizando o R

Apresentaremos um guia prático, utilizando o R, para obter cópulas a partir de distribuições de probabilidade empíricas e teóricas. Para este trabalho foi utilizada a versão 4.1.0 do R. O código completo pode ser encontrado no apêndice deste trabalho.

4.1 Fluxograma

Apresentamos na Figura 11 um fluxograma que sumariza os procedimentos para encontrar as cópulas.

Figura 11 – Fluxograma simples



Fonte: Elaboração própria

4.2 Extração e análise dos dados

Dados de qualquer fonte podem ser utilizados. Utilizaremos neste trabalho dados dos índices IFNC B3 (Índice BM&FBOVESPA Financeiro), INDX B3 (Índice do Setor Industrial) e ICON B3 (Índice de Consumo). Os dados históricos, do período de 01/01/2018 a 31/12/2020, foram extraídos com auxílio do site br.investing.com. O trecho de códigos abaixo mostra como importar os dados obtidos para o R, fazer o tratamento necessário (log-retornos) e as estatísticas descritivas.

```

### Importação de Dados ###
file <- file.choose() # Abre janela de seleção de arquivos
data <- read.csv2(file) # Lê o arquivo
IFNC <- data$IFNC
INDX <- data$INDX
ICON <- data$ICON

### Tratamento ###
# Log-Retornos
LR_IFNC <- diff(log(IFNC))
LR_INDX <- diff(log(INDX))
LR_ICON <- diff(log(ICON))

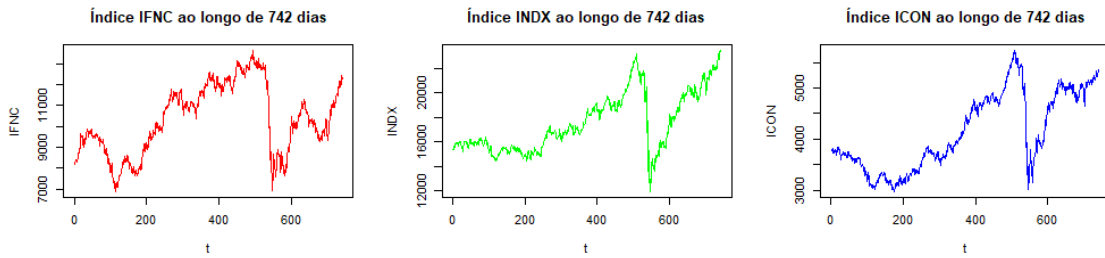
### Estatísticas Descritivas das Séries ###
# 1. Média
mean(LR_IFNC);mean(LR_INDX);mean(LR_ICON)
# 2. Mediana
median(LR_IFNC);median(LR_INDX);median(LR_ICON)
# 3. Variância
var(LR_IFNC);var(LR_INDX);var(LR_ICON)
# 4. Desvio-padrão
sd(LR_IFNC);sd(LR_INDX);sd(LR_ICON)
# 5. Range (min, max)
range(LR_IFNC);range(LR_INDX);range(LR_ICON)
# 6. Assimetria
skewness(LR_IFNC);skewness(LR_INDX);mean(LR_ICON)
# 7. Curtose
kurtosis(LR_IFNC);kurtosis(LR_INDX);mean(LR_ICON)

### Estatísticas de Correlação ###
# 1. Pearson
cor.test(LR_IFNC,LR_INDX,method="pearson")
cor.test(LR_IFNC,LR_ICON,method="pearson")
cor.test(LR_INDX,LR_ICON,method="pearson")
# 2. Spearman
cor.test(LR_IFNC,LR_INDX,method="spearman")
cor.test(LR_IFNC,LR_ICON,method="spearman")
cor.test(LR_INDX,LR_ICON,method="spearman")
# 3. Kendall
cor.test(LR_IFNC,LR_INDX,method="kendall")
cor.test(LR_IFNC,LR_ICON,method="kendall")
cor.test(LR_INDX,LR_ICON,method="kendall")

```

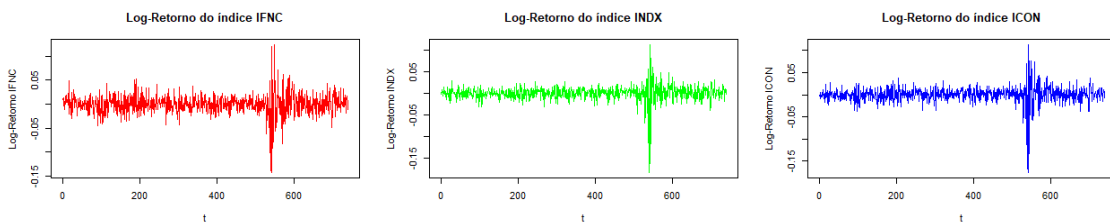
A Figura 12 ajuda a visualizar os dados utilizados antes do tratamento. Em seguida, a Figura 13 apresenta a série de log-retornos.

Figura 12 – Séries dos valores de fechamento dos índices IFNC, INDX e ICON ao longo de 742 dias



Fonte: br.investing.com, elaboração própria.

Figura 13 – Séries de log-retornos dos índices IFNC, INDX e ICON



Fonte: Elaboração própria.

Geramos a série de log-retornos para fazer uma melhor análise das dependências entre os log-retornos dos índices. Observe que, aproximadamente, o índice IFNC varia entre 6.000 e 13.000, o índice INDX varia entre 12.000 e 23.000 e o índice ICON varia entre 3.000 e 5.800. Portanto, a série de log-retornos, entre outros benefícios para este estudo, nos fornece uma métrica mais comparável entre os índices.

Tabela 1 – Estatísticas descritivas das séries de log-retornos

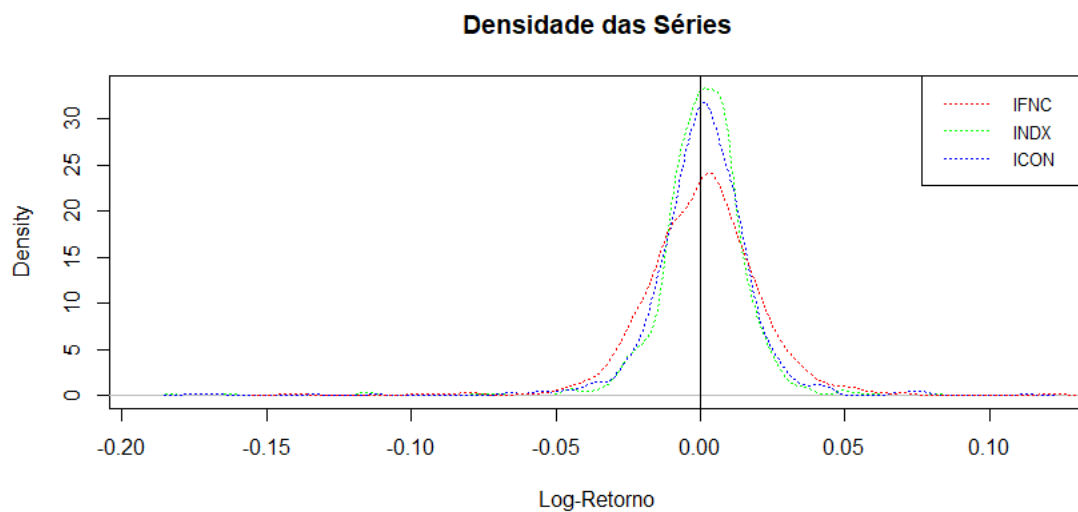
	IFNC	INDX	ICON
Média	0,0005470303	0,0005651729	0,0004584086
Erro-padrão	0,0007981651	0,000664105	0,0007223812
Mediana	0,001217943	0,001302766	0,001100682
Variância	0,000472067	0,0003268073	0,0003866794
Desvio-Padrão	0,0217271	0,01807781	0,01966417
Assimetria	-0,537865	-2,786595	-2,111543
Curtose	11,22839	32,68679	24,67568
Mín.	-0,142505	-0,1833826	-0,1761739
Máx.	0,123577	0,1125522	0,1127713
Observações	741	741	741

Fonte: Elaboração própria

Os valores das estatísticas descritivas dispostos na Tabela 1 destacam que as séries de log-retornos possuem assimetria negativa o que representa uma distribuição com uma cauda mais longa à esquerda. Assim, a mediana será maior que a média da distribuição, o que é confirmado pelos valores acima.

Os valores de curtose para as séries mostram que as distribuições são leptocúrticas e, portanto, apresentam caudas mais pesadas. Para ilustrar a distribuição dos dados e facilitar a visualização das estatísticas descritas na Tabela 1, preparamos gráficos com a densidade de cada série e com uma linha vertical em 0, para mostrar a assimetria e outros comportamentos da distribuição em relação a uma distribuição normal padrão.

Figura 14 – Densidade das séries de log-retornos do IFNC, INDX e ICON



Fonte: Elaboração própria.

Tabela 2 – Correlação entre as séries de log-retornos

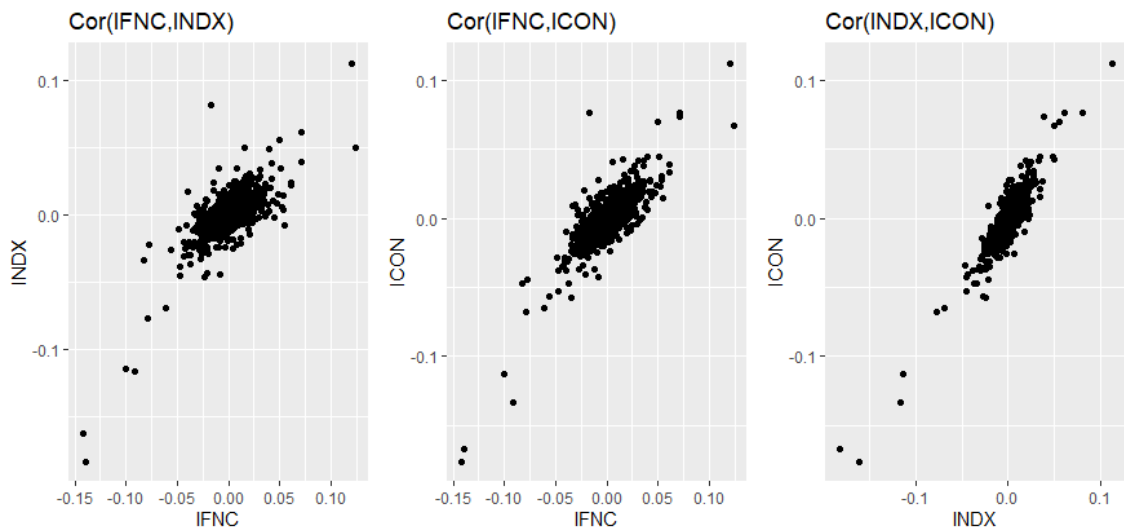
		Pearson (ρ)		
		Cor(IFNC, INDX)	Cor(IFNC, ICON)	Cor(INDX, ICON)
Estimativa		0,7353053	0,8118358	0,9110643
p-valor		< 2,2e-16	< 2,2e-16	< 2,2e-16

		Spearman (ρ)		
		Cor(IFNC, INDX)	Cor(IFNC, ICON)	Cor(INDX, ICON)
Estimativa		0,6440034	0,7233619	0,8120151
p-valor		< 2,2e-16	< 2,2e-16	< 2,2e-16

		Kendall (ρ)		
		Cor(IFNC, INDX)	Cor(IFNC, ICON)	Cor(INDX, ICON)
Estimativa		0,4672357	0,5436408	0,6319656
p-valor		< 2,2e-16	< 2,2e-16	< 2,2e-16

Fonte: Elaboração própria

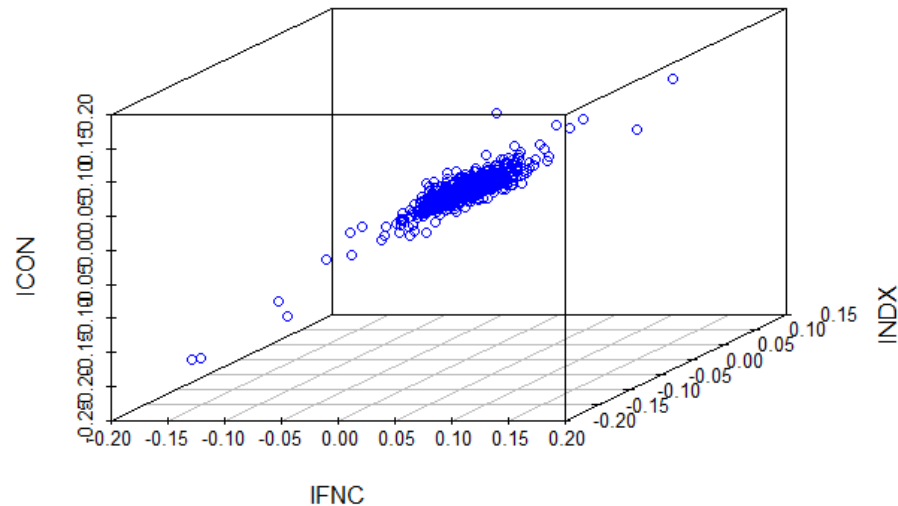
Como demonstra a Tabela 2, as séries possuem alta correlação e o valor estimado pelo teste de correlação é estatisticamente significativo, tanto para testes paramétricos, como é o caso de Pearson, quanto para os testes não-paramétricos. Para ilustrar essa correlação, a Figura 15 apresenta graficamente os valores da Tabela 2.

Figura 15 – Gráficos de dispersão

Fonte: Elaboração própria.

Adicionalmente, a Figura 15 demonstra a correlação entre as três séries de log-retornos.

Figura 15 – Gráfico de dispersão utilizando os três índices



Fonte: Elaboração própria.

4.3 Ajuste da Distribuição de Probabilidade

Com o exposto no tópico anterior é possível procurar as distribuições de probabilidade que mais se ajustam ao conjunto de dados utilizando o método da máxima verossimilhança, log-verossimilhança ou pseudo-verossimilhança, quando aplicável.

Apesar das evidências de não-normalidade, para começar testamos a distribuição normal e, logo em seguida, a distribuição t de Student. Considerando a assimetria negativa, o valor da curtose e a existência de valores negativos na distribuição, testamos os dados para as distribuições generalizadas hiperbólicas (GHD), como visto em trabalhos como Socgnia e Wilcox (2014).

As distribuições generalizadas hiperbólicas, além da distribuição t de Student, incluem as distribuições hiperbólica, normal-inversa de Gauss e variância-gama, que foram as três distribuições escolhidas para serem testadas nos dados.

Após o fit, é necessário realizar testes de *godness-of-fit* com os parâmetros encontrados para as distribuições testadas, já que é necessário encontrar a distribuição com melhor ajuste. O teste de *goodness-of-fit* escolhido para o exemplo foi o de Anderson-Darling, uma vez que dá mais peso às caudas das distribuições.

Os códigos abaixo demonstram como os ajustes podem ser feitos no R. Primeiro utilizamos as funções da biblioteca `fitdistrplus`, que permite um ajuste de distribuição de probabilidade com certa facilidade utilizando o método da máxima verossimilhança.

Para as distribuições generalizadas hiperbólicas foram utilizadas funções nativas das bibliotecas que contém essas distribuições. Essas funções permitem calcular os valores iniciais para o ajuste de distribuição, como requerido pelo método da máxima verossimilhança.

```

### Ajuste da distribuição de probabilidade ###
# Normal
fit_IFNC <- fitdist(data=LR_IFNC,distr="norm",method="mle")
fit_INDX <- fitdist(data=LR_INDX,distr="norm",method="mle")
fit_ICON <- fitdist(data=LR_ICON,distr="norm",method="mle")

# t Scaled
fit_IFNC <- fitdist(data=LR_IFNC,distr="t.scaled",method="mle",
start=list(df=5,mean=mean(LR_IFNC),sd=sd(LR_IFNC)))
fit_INDX <- fitdist(data=LR_INDX,distr="t.scaled",method="mle",
start=list(df=5,mean=mean(LR_INDX),sd=sd(LR_INDX)))
fit_ICON <- fitdist(data=LR_ICON,distr="t.scaled",method="mle",
start=list(df=5,mean=mean(LR_INDX),sd=sd(LR_INDX)))

## Distribuições Generalizadas Hiperbólicas ##
# VALORES INICIAIS #
# Hiperbólica
IFNC.hyperb.start <- hyperbFitStart(LR_IFNC,startMethodSL="Nelder-
Mead",startMethodMoM="Nelder-Mead")
INDX.hyperb.start <- hyperbFitStart(LR_INDX,startMethodSL="Nelder-
Mead",startMethodMoM="Nelder-Mead")
ICON.hyperb.start <- hyperbFitStart(LR_ICON,startMethodSL="Nelder-
Mead",startMethodMoM="Nelder-Mead")

# NIG
IFNC.nig.start <- nigFitStart(LR_IFNC,startMethodMoM="Nelder-Mead")
INDX.nig.start <- nigFitStart(LR_INDX,startMethodMoM="Nelder-Mead")
ICON.nig.start <- nigFitStart(LR_ICON,startMethodMoM="Nelder-Mead")

# Var-Gama
IFNC.vg.start <- vgFitStart(LR_IFNC,startMethodSL="Nelder-
Mead",startMethodMoM="Nelder-Mead")
IFNC.vg.start <- vgFitStart(LR_INDX,startMethodSL="Nelder-
Mead",startMethodMoM="Nelder-Mead")
IFNC.vg.start <- vgFitStart(LR_ICON,startMethodSL="Nelder-
Mead",startMethodMoM="Nelder-Mead")

# Ajuste de Distribuição #
# IFNC
hyperb.IFNC <-
hyperbFit(LR_IFNC,paramStart=c(IFNC.hyperb.start[1],IFNC.hyperb.start[
2],IFNC.hyperb.start[3],IFNC.hyperb.start[4]))
nig.IFNC <-
nigFit(LR_IFNC,paramStart=c(IFNC.nig.start[1],IFNC.nig.start[2],IFNC.n
ig.start[3],IFNC.nig.start[4]))
vg.IFNC <-
vgFit(LR_IFNC,paramStart=c(IFNC.vg.start[1],IFNC.vg.start[2],IFNC.vg.s
tart[3],IFNC.vg.start[4]))

# INDX
hyperb.INDX <-
hyperbFit(LR_INDX,paramStart=c(INDX.hyperb.start[1],INDX.hyperb.start[
2],INDX.hyperb.start[3],INDX.hyperb.start[4]))
nig.INDX <-
nigFit(LR_INDX,paramStart=c(INDX.nig.start[1],INDX.nig.start[2],INDX.n
ig.start[3],INDX.nig.start[4]))
vg.INDX <-
vgFit(LR_INDX,paramStart=c(INDX.vg.start[1],INDX.vg.start[2],INDX.vg.s
tart[3],INDX.vg.start[4]))

```

```

### Goodness of Fit ###
#Normal
ad.test(LR_IFNC, "pnorm", mean=fit.normal.IFNC$estimate[1], sd=fit.normal
.IFNC$estimate[2])
ad.test(LR_INDX, "pnorm", mean=fit.normal.INDX$estimate[1], sd=fit.normal
.INDX$estimate[2])
ad.test(LR_ICON, "pnorm", mean=fit.normal.ICON$estimate[1], sd=fit.normal
.ICON$estimate[2])

#t
ad.test(LR_IFNC, "t.scaled", df=fit.t.IFNC$estimate[1],
mean=fit.t.IFNC$estimate[2], sd=fit.t.IFNC$estimate[3])
ad.test(LR_INDX, "t.scaled", df=fit.t.INDX$estimate[1],
mean=fit.t.INDX$estimate[2], sd=fit.t.INDX$estimate[3])
ad.test(LR_ICON, "t.scaled", df=fit.t.ICON$estimate[1],
mean=fit.t.ICON$estimate[2], sd=fit.t.ICON$estimate[3])

# GHD
# IFNC
ad.test(LR_IFNC, "hyperb", mu=hyperb.IFNC$param[1], delta=hyperb.IFNC$par
am[2], alpha=hyperb.IFNC$param[3], beta=hyperb.IFNC$param[4])
ad.test(LR_IFNC, "nig", mu=nig.IFNC$param[1], delta=nig.IFNC$param[2], alp
ha=nig.IFNC$param[3], beta=nig.IFNC$param[4])
ad.test(LR_IFNC, "vg", param=c(vg.IFNC$param[1], vg.IFNC$param[2], vg.IFNC
$param[3], vg.IFNC$param[4]))

# INDX
ad.test(LR_INDX, "hyperb", mu=hyperb.INDX$param[1], delta=hyperb.INDX$par
am[2], alpha=hyperb.INDX$param[3], beta=hyperb.INDX$param[4])
ad.test(LR_INDX, "nig", mu=nig.INDX$param[1], delta=nig.INDX$param[2], alp
ha=nig.INDX$param[3], beta=nig.INDX$param[4])
ad.test(LR_INDX, "vg", param=c(vg.INDX$param[1], vg.INDX$param[2], vg.INDX
$param[3], vg.INDX$param[4]))

# ICON
ad.test(LR_ICON, "hyperb", mu=hyperb.ICON$param[1], delta=hyperb.ICON$par
am[2], alpha=hyperb.ICON$param[3], beta=hyperb.ICON$param[4])
ad.test(LR_ICON, "nig", mu=nig.ICON$param[1], delta=nig.ICON$param[2], alp
ha=nig.ICON$param[3], beta=nig.ICON$param[4])
ad.test(LR_ICON, "vg", param=c(vg.ICON$param[1], vg.ICON$param[2], vg.ICON
$param[3], vg.ICON$param[4]))

```

Os resultados dos ajustes de distribuições de probabilidade são demonstrados na Tabela 3. Como demonstra a tabela, a distribuição t de Student se ajusta melhor aos dados dos três índices. O p-valor demonstra que os resultados são significativos, ou seja, não podemos rejeitar a hipótese de que os dados seguem uma distribuição t de Student “scaled” com os parâmetros testados.

Tabela 3 – Ajuste das distribuições de probabilidade.

Índice	Distribuição	Anderson-Darling	p-valor
IFNC	Normal	80,885	0,0001015
INDX	Normal	25,124	0,0008097
ICON	Normal	21,664	0,0008097
IFNC	t	0,30208	0,9366
INDX	t	0,67835	0,5771
ICON	t	0,37742	0,8708
IFNC	Hiperbólica	0.57758	0,6694
INDX	Hiperbólica	1,5529	0.1643
ICON	Hiperbólica	11.561	0,2848
IFNC	NIG	0.47221	0,7752
INDX	NIG	1,0283	0,3427
ICON	NIG	0,64517	0,6063
IFNC	Var-Gama	21.333	0,0008097
INDX	Var-Gama	Inf	0,0008097
ICON	Var-Gama	5,4775	0,0017010

Fonte: Elaboração própria.

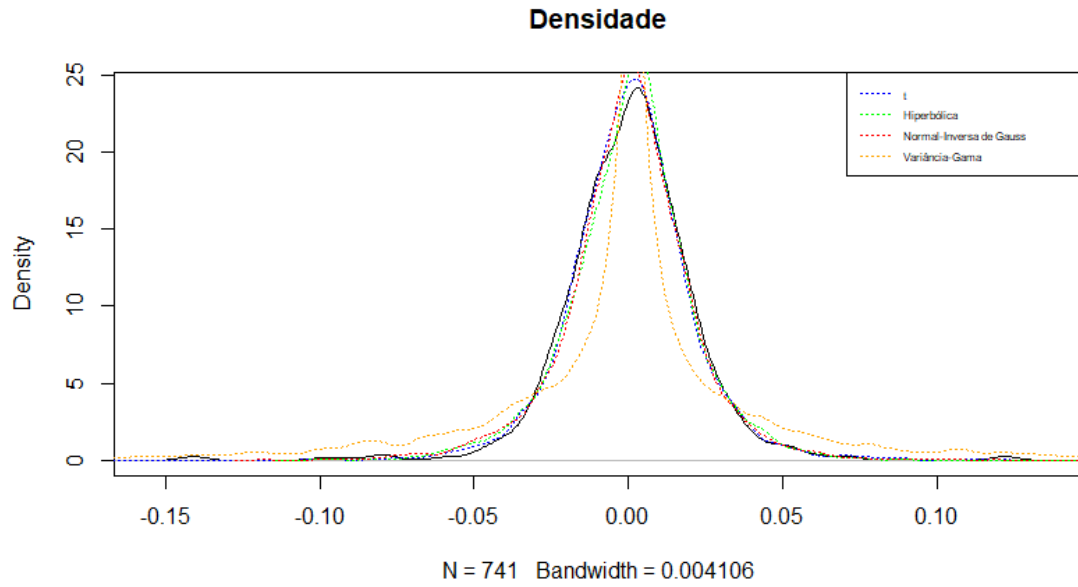
A Tabela 4 demonstra os parâmetros encontrados no processo, bem como os valores de AIC e BIC.

Tabela 4 – Parâmetros das distribuições

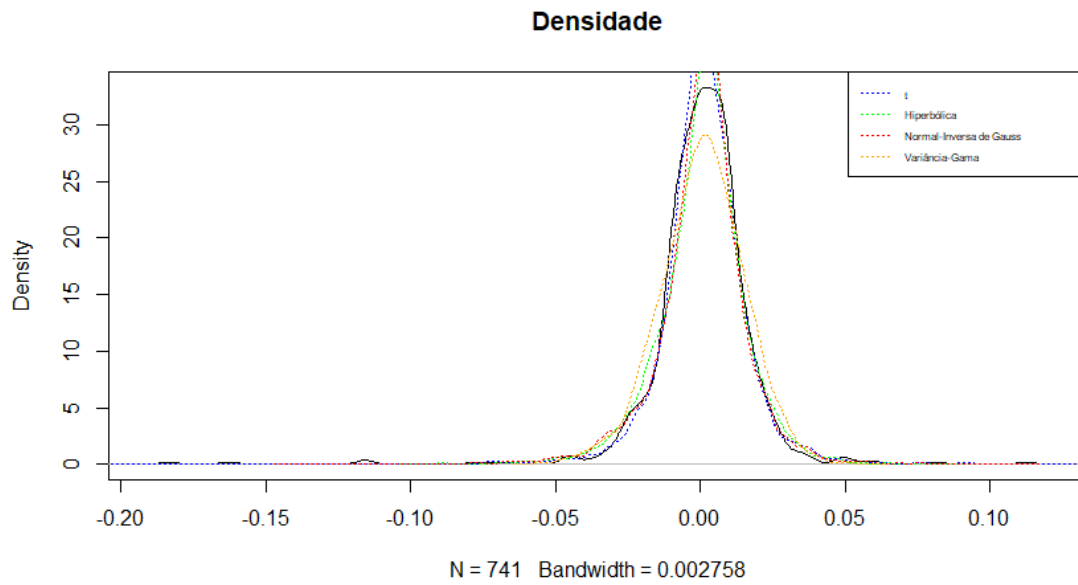
Índice	Distribuição	Graus de Liberdade	Desvio-Padrão	Média	AIC	BIC
IFNC		3,849699	0,000847	0,014785	-3.731,132	-3.717,308
INDX	<i>t</i>	2,990575	0,001404	0,009571	-4.253,837	-4.240,013
ICON		2,890473	0,001239	0,010678	-4.072,616	-4.058,792

Fonte: Elaboração própria.

Nas Figuras 16, 17 e 18 são demonstradas graficamente as distribuições ajustadas para cada índice, com exceção da distribuição normal, uma vez que sua inadequação já pode ser visualizada no capítulo 4.2 na Tabela 1 e na Figura 14. Através dos gráficos também é possível visualizar as distribuições empíricas dos dados.

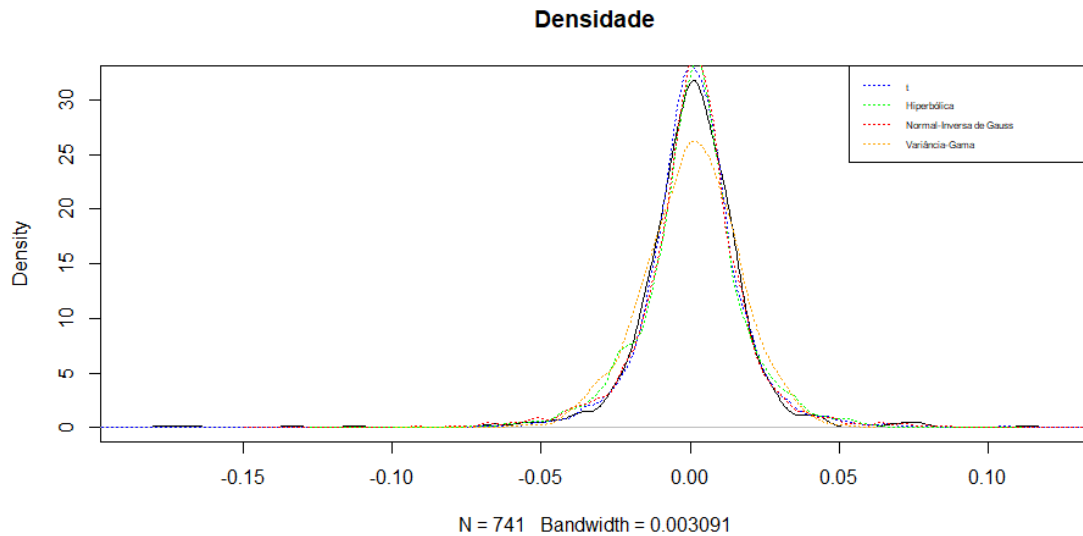
Figura 16 – Funções de densidade de probabilidade ajustadas aos log-retornos do IFNC

Fonte: Elaboração própria

Figura 17 – Funções de densidade de probabilidade ajustadas aos log-retornos do INDX

Fonte: Elaboração própria

Figura 18 – Funções de densidade de probabilidade ajustadas aos log-retornos do ICON



Fonte: Elaboração própria

4.4 Ajuste de Cópulas

Após obter as distribuições empíricas e teóricas nos capítulos anteriores, então podemos obter as cópulas. Serão feitos dois procedimentos para obter cópulas: a partir dos dados empíricos e a partir dos dados teóricos. Apenas cópulas bivariadas serão encontradas, uma vez que o tempo de execução e a “demanda computacional” crescem muito para cópulas d -dimensionais quando $d > 2$. No entanto a função `fitcopula` encontrada na biblioteca `copula` do R permite realizar o fit de cópulas de 2 ou mais dimensões facilmente.

4.4.1 Pseudo-observações

Uma vez que as distribuições marginais de uma cópula precisam ser uniformes em $(0,1)$, é preciso transformar as distribuições marginais em uniformes em $(0,1)$ por meio de pseudo-observações. As pseudo-observações são calculadas da seguinte forma: são feitas estatísticas de ordem e o rank dos dados é dividido pelo número de observações adicionado de uma unidade. Em caso de empate nos dados, o rank é feito pela média das posições dos elementos. Por exemplo: se há um vetor $(1, 2, 2, 3, 4)$, o rank é igual a $1, 2.5, 2.5, 4$ e 5 .

4.4.2 Cópulas a partir da distribuição empírica

As cópulas podem ser encontradas a partir de distribuições empíricas com facilidade. Uma vez que as distribuições marginais já foram encontradas, basta transformá-las em

uniformes em (0,1) e então procurar as cópulas. O código no R apresentado abaixo demonstra como esse procedimento foi feito para o exemplo.

```
### Pseudo-observações ###

# 1. Empíricas
epobs <- pobs(cbind(LR_IFNC, LR_INDX, LR_ICON))

# Selecionar cópulas ###
# A partir das distribuições empíricas
cop_1 <- BiCopSelect(u1=epobs[,1],u2=epobs[,2],selectioncrit="logLik")

# Cópula selecionada
# IFNC x INDX
emp_1 <- BiCopSelect(epobs[,1],epobs[,2],selectioncrit="logLik")
# IFNC x ICON
emp_2 <- BiCopSelect(epobs[,1],epobs[,3],selectioncrit="logLik")
# INDX x ICON
emp_3 <- BiCopSelect(epobs[,2],epobs[,3],selectioncrit="logLik")

# Goodness of Fit
gof_emp_1 <-
gofCopula(x=cbind(epobs[,1],epobs[,2]),VC2copula::BB1Copula(param=c(em
p_1$par,emp_1$par2)))
gof_emp_2 <-
gofCopula(x=cbind(epobs[,1],epobs[,3]),VC2copula::surBB7Copula(param=c
(emp_2$par,emp_2$par2)))
gof_emp_3 <-
gofCopula(x=cbind(epobs[,2],epobs[,3]),VC2copula::BB7Copula(param=c(em
p_3$par,emp_3$par2)))
```

Geramos as pseudo-observações com auxílio da função `pobs` da biblioteca `copula`. Então selecionamos as cópulas com auxílio da função `BiCopSelect`, disponível na biblioteca `VineCopula`, utilizando log-verossimilhança. Por fim, realizamos o teste de *goodness-of-fit* das cópulas selecionadas. O teste realizado é o de Cramér-von Mises, apenas.

Os resultados estão apresentados no capítulo 5, “Resultados”.

4.4.3 Cópulas a partir da distribuição teórica

Para as distribuições teóricas o mesmo procedimento do capítulo 4.4.2 pode ser adotado. No entanto, é possível simular valores para obter n observações, e as distribuições empíricas são limitadas às suas próprias observações. O código no R apresentado abaixo demonstra como esse procedimento foi feito para o exemplo.

```

### Pseudo-observações ###

# 2. Teóricas
set.seed(123)
pobs <- pobs(cbind(rt.scaled(n=5000,df=fit.t.IFNC$estimate[1],
                           mean=fit.t.IFNC$estimate[2],
                           sd=fit.t.IFNC$estimate[3]),
                 rt.scaled(n=5000,df=fit.t.INDX$estimate[1],
                           mean=fit.t.INDX$estimate[2],
                           sd=fit.t.INDX$estimate[3]),
                 rt.scaled(n=5000,df=fit.t.ICON$estimate[1],
                           mean=fit.t.ICON$estimate[2],
                           sd=fit.t.ICON$estimate[3])))

pobs_IFNC <- pobs[,1]
pobs_INDX <- pobs[,2]
pobs_ICON <- pobs[,3]

# Cópula selecionada
# IFNC x INDX
teo_1 <- BiCopSelect(pobs_IFNC,pobs_INDX,selectioncrit="logLik")
# IFNC x ICON
teo_2 <- BiCopSelect(pobs_IFNC,pobs_ICON,selectioncrit="logLik")
# INDX x ICON
teo_3 <- BiCopSelect(pobs_INDX,pobs_ICON,selectioncrit="logLik")

# Goodness of Fit
gof_teo_1 <-
gofCopula(x=cbind(pobs[,1],pobs[,2]),frankCopula(param=teo_1$par))
gof_teo_2 <-
gofCopula(x=cbind(pobs[,1],pobs[,3]),VC2copula::r90TawnT2Copula(param=
c(teo_2$par,teo_2$par2)))
gof_teo_3 <-
gofCopula(x=cbind(pobs[,2],pobs[,3]),VC2copula::tawnT2Copula(param=c(
teo_3$par,teo_3$par2)))

```

Geramos 5000 valores de cada distribuição teórica e, então, geramos as pseudo-observações. Então selecionamos as cópulas utilizando log-verossimilhança e, por fim, realizamos os testes de goodness-of-fit das cópulas selecionadas. O teste realizado é o de Cramér-von Mises, apenas.

Os resultados estão apresentados no capítulo 5, “Resultados”.

5 Resultados

Apresentamos os resultados obtidos no trabalho, resumidos, na Tabela 5:

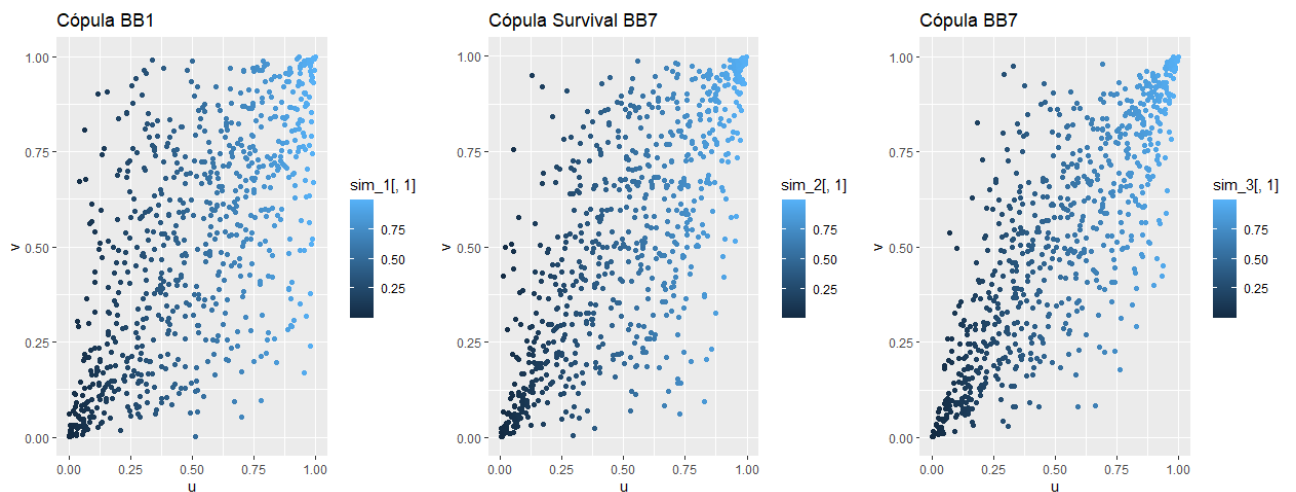
Tabela 5 – Parâmetro das cópulas e p-valor obtido nos testes de goodness-of-fit

(u, v)	Distribuição teórica	Cópula (distribuições empíricas)	Parâmetros	p-valor	Cópula (distribuições teóricas)	Parâmetros	p-valor
(IFNC, INDX)	t	BB1	$\alpha = 0,7907$ $\delta = 1,3567$	0,8529	Frank	$\alpha = 0,051468$	0,5000
(IFNC, ICON)	t	Survival BB7	$\alpha = 2,3596$ $\delta = 1,0503$	0,4804	Tawn Tipo 2 Rot. 90°	$\alpha = -1,0298$ $\delta_1 = 0,0635$ $\delta_2 = 1,0000$	1,0000
(INDX, ICON)	t	BB7	$\alpha = 2,3726$ $\delta = 2,3437$	0,1667	Tawn Tipo 2	$\alpha = 1,0752$ $\delta_1 = 0,0139$ $\delta_2 = 1,0000$	1,0000

Fonte: Elaboração própria.

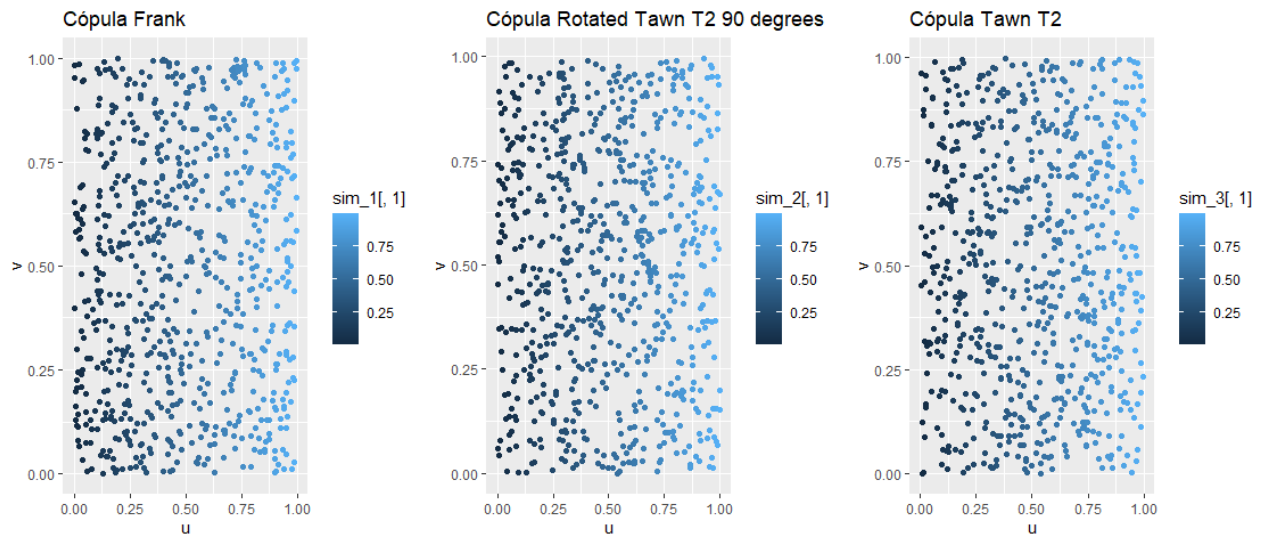
O teste de *goodness-of-fit* performado nos capítulos 4.4.2 e 4.4.3 retornou valores significativos (p-valor), o que significa que não podemos rejeitar a hipótese de que a estrutura de dependência entre os dados estudados pode ser representada pelas cópulas testadas. Apresentamos nas Figuras 19 e 20, respectivamente, as estruturas de dependências obtidas a partir dos dados empíricos e teóricos.

Figura 19 – Cópulas BB1, Survival BB7 e BB7.



Fonte: Elaboração própria

Figura 20 – Cópulas BB1, Survival BB7 e BB7.



Fonte: Elaboração própria

Como visto na Tabela 5, os resultados apontam uma maior precisão quando as distribuições teóricas são utilizadas. É importante ressaltar, no entanto, que testamos apenas poucos pontos no universo de distribuições disponíveis, e eventualmente há distribuições marginais que podem ser melhores que as encontradas para as séries de log-retornos utilizadas neste trabalho, e gerar melhores modelos. Um p-valor significativo é um bom indicativo para testes de *goodness-of-fit*, porém não deve ser a única coisa levada em consideração nas análises – a subjetividade é importante. Também, quanto melhor as especificações do computador do usuário, mais iterações podem ser feitas no momento dos ajustes, gerando parâmetros precisos e testes de *goodness-of-fit* cada vez mais fidedignos.

As três cópulas obtidas através das distribuições marginais empíricas apresentaram maior concentração nas caudas, como mostra a Figura 19. É possível notar, nos três casos, uma maior concentração de probabilidades na cauda esquerda, assim como nas distribuições marginais empíricas. A dispersão apresentada lembra levemente a que pode ser vista com a cópula de Gumbel, mas a cópula de Gumbel não define bem suas caudas. Assim, apesar de um menor p-valor (muito pelo número de observações disponível), a estrutura de dependência apresenta maior concordância com as distribuições e com as correlações.

Na Figura 20, já com as distribuições teóricas, é possível notar que há pouca concentração nas caudas e uma maior dispersão dos pontos, mostrando menor correlação. Parte desse comportamento já é esperado da cópula de Frank, já que ela não possui caudas bem definidas no seu gráfico de dispersão e geralmente apresenta um formato de nuvem linear ao longo do gráfico. As outras duas cópulas apresentaram comportamento similar – estamos

falando das mesmas distribuições marginais, porém com parâmetros diferentes. Deve-se explicar que uma cópula *t* de Student não foi ajustada pois utilizamos uma distribuição de *t* conhecida como “*t* scaled”, que é uma distribuição *t* que utiliza a média e o desvio-padrão como parâmetros além dos graus de liberdade. Além disso, vale lembrar que os testes de *goodness-of-fit* são objetivos, não subjetivos, então medem um ajuste, mas não o explicam.

6 Conclusão

Este trabalho teve como objetivo apresentar na prática o ajuste de distribuições de probabilidade e o ajuste de cópulas, utilizando dados reais. Assim, o maior objetivo foi demonstrar o uso dessas ferramentas que podem ser aplicadas no mercado financeiro, como no exemplo escolhido para ilustrar o roteiro, mas há diversas aplicações possíveis com já conhecidos dados de hidrologia, meteorologia etc.

É importante salientar que, no entanto, as aplicações sugeridas no trabalho, já descritas várias vezes em outras publicações sobre o assunto mundo afora, como a precificação de opções, têm caráter mais acadêmico apesar de apresentar maior precisão, uma vez que muito tempo seria necessário para precificar opções considerando que há mais de 40.000 companhias listadas nas bolsas de valores ao redor do mundo, uma infinidade de tipos de ações e units negociadas, e as negociações de opções ocorrem de forma diária. Então, encontrar diariamente as distribuições de probabilidade de todos os ativos subjacentes, depois encontrar as cópulas de *n*-dimensões, confirmar que essa cópula é razoável via testes de *goodness-of-fit* e, por fim, calcular o valor de uma opção é algo demasiado laborioso e demorado, o que não combina com o dinamismo do mercado atualmente. Apesar da forte evidência de não-normalidade dos dados, a fórmula de Black-Scholes permanece como uma boa opção pela rapidez com que cálculos podem ser feitos.

Espera-se que o leitor tenha compreendido os procedimentos efetuados de forma básica neste trabalho, e as limitações inerentes aos dados obtidos “do mundo real”. Todos os códigos utilizados podem ser encontrados no apêndice deste trabalho para consulta.

Por fim, ressaltamos que esse conhecimento é importante para todos os atuários, financistas e outros profissionais do mercado financeiro e de seguros, uma vez que a tendência é de aumento das aplicações das cópulas no mercado e de forma mais ágil. Há muitas dependências entre variáveis aleatórias, e tratá-las como se houvesse independência em todos os contextos apenas para viabilizar estimativas pode agredir a precisão de cálculos atuariais. Portanto, incorporar na formação dos atuários modernos o conceito e as aplicações de cópulas, além de outras formas de estimar dependências, é um avanço necessário.

Referências Bibliográficas

- AGUILERA, R; FERREIRA, L.; KOLEV, N. **Dependence Analysis via Copulas**, 2015. 185 slides. Disponível em: <<https://www.ime.usp.br/~jmsinger/MAE5705/Kolev2015.pdf>>. Acesso em 10 jun. 2021.
- CHERUBINI, U.; LUCIANO, E.; VECCHIATO, W. **Copula Methods in Finance**. Chichester: Wiley, 2004. p. 1-293.
- DALL'AGLIO, G.; KOTZ, S.; SALINETTI, G. **Advances in Probability Distributions with Given Marginals: Beyond the Copulas**. Roma: Springer, 1991. p. 13.
- LOW, R.K.Y. **Vine copulas: modelling systemic risk and enhancing higher-moment portfolio optimisation**. Accounting & Finance, Melbourne, v. 58, p. 423-463, 11 mai. 2017. Disponível em: <https://doi.org/10.1111/acfi.12274>. Acesso em: 10 jul. 2021.
- GAIDUCHEVICI, Gabriel. **Post-crisis CDO Valuation with Archimedean Copulas**. Procedia Economics and Finance, Bucareste, v. 15, n. 2014, p. 1, 8 dez. 2014. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2212567114004419>. Acesso em: 9 jul. 2021
- HAUGH, Martin. **An introduction to copulas**. 29 jan. 2016, 28 mai. 2016. 21 p. Notas de Aula.
- HOFERT, Marius; SCHERER, Matias. **CDO pricing with nested Archimedean copulas**. Quantitative Finance, Abingdon-on-Thames, ano 2011, v. 11, p. 775, 9 jun. 2010. Disponível em: <https://www.tandfonline.com/doi/abs/10.1080/14697680903508479>. Acesso em: 15 jul. 2021.
- JOE, Harry. **Multivariate Models and Dependence Concepts**. 1. ed. Londres: Chapman & Hall, 1997. p. 1 – 399.
- NELSEN, Roger B. **An Introduction to Copulas**. 2. ed. Nova Iorque: Springer, 2006. p. 1-157.
- SCHMIDT, Thorsten. **Coping With Copulas**. 2007. Disponível em: https://www.researchgate.net/publication/228876267_Coping_with_copulas. Acesso em: 15 jul. 2021.
- SOCGNIA, Virginie K.; WILCOX, Diane. **A Comparison of Generalized Hyperbolic Distribution Models for Equity Returns**. Journal of Applied Mathematics, Londres, v. 2014, p. 1-15, 25 jun. 2014. Disponível em: <<https://www.hindawi.com/journals/jam/2014/263465/>>. Acesso em: 20 jun. 2021.
- TAWN, J. A. **Bivariate extreme value theory: models and estimation**. Biometrika, [s. l.], v. 75, ed. 3, p. 397-415, 1988.
- VAN DEN GOORBERGH, R.W.J.; GENEST, C.; WERKER, B.J.M. **MULTIVARIATE OPTION PRICING USING DYNAMIC COPULA MODELS**. SSRN, Tilburg, p. 1-21,

2003. Disponível em: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=556959. Acesso em: 15 jul. 2021.

VELASCO, L. L. H. **Ajuste Bayesiano para Cópulas Bivariadas**. Dissertação (Mestrado em Estatística) – Instituto de Matemática, Estatística e Computação Científica, Universidade Estadual de Campinas. Campinas, p. 8. 2016.

Apêndice

```
library(copula)
```

```
library(ggplot2)
```

```
library(grid)
```

```
library(VineCopula)
```

```
library(scatterplot3d)
```

```
library(plotrix)
```

```
library(metRology)
```

```
library(goftest)
```

```
library(GeneralizedHyperbolic)
```

```
library(VarianceGamma)
```

```
library(VC2copula)
```

```
library(gofCopula)
```

```
library(fitdistrplus)
```

```
# Extrair índices financeiros de 2018 a 2020
```

```
# Importar
```

```
file <- file.choose()
```

```
data <- read.csv2(file)
```

```
# Indices
```

```
IFNC <- data$IFNC
```

```
INDX <- data$INDX
```

```
ICON <- data$ICON
```

```
# Plot dos Índices
```

```
par(mfrow=c(1,3))
```

```
# IFNC
```

```
plot(IFNC,type="l",xlab="t",ylab="IFNC",main="Índice IFNC ao longo de 742 dias", col = "red")
```

```
# INDX
```

```
plot(INDX,type="l",xlab="t",ylab="INDX",main="Índice INDX ao longo de 742 dias", col = "green")
```

```
# ICON
```

```
plot(ICON,type="l",xlab="t",ylab="ICON",main="Índice ICON ao longo de 742 dias", col = "blue")
```

```
# Log-Retornos
```

```

LR_IFNC <- diff(log(IFNC))
LR_INDX <- diff(log(INDX))
LR_ICON <- diff(log(ICON))

# Plot dos Log-Retornos
par(mfrow=c(1,3))
# IFNC
plot(LR_IFNC,type="l",xlab="t",ylab="Log-Retorno IFNC",main="Log-Retorno do índice IFNC", col = "red")
# INDX
plot(LR_INDX,type="l",xlab="t",ylab="Log-Retorno INDX",main="Log-Retorno do índice INDX", col =
"green")
# ICON
plot(LR_ICON,type="l",xlab="t",ylab="Log-Retorno ICON",main="Log-Retorno do índice ICON", col = "blue")

# Plot das Densidades
plot(density(LR_INDX),xlab="Log-Retorno",main="Densidade das Séries",col="green",lty="dotted")
lines(density(LR_ICON),col="blue",lty="dotted")
lines(density(LR_IFNC),col="red",lty="dotted")
abline(v=0)
legend("topright", legend=c("IFNC", "INDX", "ICON"),
      col=c("red", "green", "blue"), lty="dotted", cex=0.8)

# Correlação
# Pearson
cor.test(LR_IFNC,LR_INDX,method="pearson")
cor.test(LR_IFNC,LR_ICON,method="pearson")
cor.test(LR_INDX,LR_ICON,method="pearson")

# Spearman
cor.test(LR_IFNC,LR_INDX,method="spearman")
cor.test(LR_IFNC,LR_ICON,method="spearman")
cor.test(LR_INDX,LR_ICON,method="spearman")

# Kendall
cor.test(LR_IFNC,LR_INDX,method="kendall")
cor.test(LR_IFNC,LR_ICON,method="kendall")
cor.test(LR_INDX,LR_ICON,method="kendall")

# Scatterplot
# Plot samples

```



```

p1 <- qplot(LR_IFNC, LR_INDX,
            main="Cor(IFNC,INDX)",
            xlab = "IFNC", ylab = "INDX")
p2 <- qplot(LR_IFNC, LR_ICON,
            main="Cor(IFNC,ICON)",
            xlab = "IFNC", ylab = "ICON")
p3 <- qplot(LR_INDX, LR_ICON,
            main="Cor(INDX,ICON)",
            xlab = "INDX", ylab = "ICON")

pushViewport(viewport(layout = grid.layout(1, 3)))
print(p1, vp = viewport(layout.pos.row = 1, layout.pos.col = 1))
print(p2, vp = viewport(layout.pos.row = 1, layout.pos.col = 2))
print(p3, vp = viewport(layout.pos.row = 1, layout.pos.col = 3))

scatterplot3d(x=LR_IFNC, y=LR_INDX, z=LR_ICON, color="blue",
              xlab="IFNC",ylab="INDX",zlab="ICON")

# Fit de Distribuição
fit.normal.IFNC <- fitdist(data=LR_IFNC, distr="norm", method="mle")
fit.normal.INDX <- fitdist(data=LR_INDX, distr="norm", method="mle")
fit.normal.ICON <- fitdist(data=LR_ICON, dist="norm", method="mle")

plot(fit.normal.IFNC,breaks=100)
plot(fit.normal.INDX,breaks=100)
plot(fit.normal.ICON,breaks=100)

fit.t.IFNC <- fitdist(data=LR_IFNC, distr="t.scaled",
                     method="mle",
                     start=list(df=5,mean=mean(LR_IFNC),sd=sd(LR_IFNC)))
fit.t.INDX <- fitdist(data=LR_INDX, distr="t.scaled",
                     method="mle",
                     start=list(df=5,mean=mean(LR_INDX),sd=sd(LR_INDX)))
fit.t.ICON <- fitdist(data=LR_ICON, dist="t.scaled",
                     method="mle",
                     start=list(df=5,mean=mean(LR_ICON),sd=sd(LR_ICON)))

plot(fit.t.IFNC,breaks=100)
plot(fit.t.INDX,breaks=100)

```

```

plot(fit.t.ICON,breaks=100)

# Goodness of Fit
#Normal
ad.test(LR_IFNC,"pnorm",
        mean=fit.normal.IFNC$estimate[1],
        sd=fit.normal.IFNC$estimate[2])
ad.test(LR_INDX,"pnorm",
        mean=fit.normal.INDX$estimate[1],
        sd=fit.normal.INDX$estimate[2])
ad.test(LR_ICON,"pnorm",
        mean=fit.normal.ICON$estimate[1],
        sd=fit.normal.ICON$estimate[2])

#t
ad.test(LR_IFNC,"t.scaled",
        df=fit.t.IFNC$estimate[1],
        mean=fit.t.IFNC$estimate[2],
        sd=fit.t.IFNC$estimate[3])
ad.test(LR_INDX,"t.scaled",
        df=fit.t.INDX$estimate[1],
        mean=fit.t.INDX$estimate[2],
        sd=fit.t.INDX$estimate[3])
ad.test(LR_ICON,"t.scaled",
        df=fit.t.ICON$estimate[1],
        mean=fit.t.ICON$estimate[2],
        sd=fit.t.ICON$estimate[3])

# GHD - Starting Values
# Hiperbólica
IFNC.hyperb.start <- hyperbFitStart(LR_IFNC,startMethodSL="Nelder-Mead",startMethodMoM="Nelder-
Mead")
INDX.hyperb.start <- hyperbFitStart(LR_INDX,startMethodSL="Nelder-Mead",startMethodMoM="Nelder-
Mead")
ICON.hyperb.start <- hyperbFitStart(LR_ICON,startMethodSL="Nelder-Mead",startMethodMoM="Nelder-
Mead")

# Normal-Inversa de Gauss
IFNC.nig.start <- nigFitStart(LR_IFNC,startMethodMoM="Nelder-Mead")
INDX.nig.start <- nigFitStart(LR_INDX,startMethodMoM="Nelder-Mead")

```

```

ICON.nig.start <- nigFitStart(LR_ICON,startMethodMoM="Nelder-Mead")

# Variância-Gama
IFNC.vg.start <- vgFitStart(LR_IFNC,startMethodSL="Nelder-Mead",startMethodMoM="Nelder-Mead")
IFNC.vg.start <- vgFitStart(LR_INDX,startMethodSL="Nelder-Mead",startMethodMoM="Nelder-Mead")
IFNC.vg.start <- vgFitStart(LR_ICON,startMethodSL="Nelder-Mead",startMethodMoM="Nelder-Mead")

# Para Hyperb e NIG
# 1 = mu
# 2 = delta
# 3 = alpha
# 4 = beta

# Para VG
# 1 = vgC
# 2 = lsigma
# 3 = theta
# 4 = lnu

# Fit
# IFNC
hyperb.IFNC <- hyperbFit(LR_IFNC,paramStart=c(IFNC.hyperb.start[1],IFNC.hyperb.start[2],
      IFNC.hyperb.start[3],IFNC.hyperb.start[4]))
nig.IFNC <- nigFit(LR_IFNC,paramStart=c(IFNC.nig.start[1],IFNC.nig.start[2],
      IFNC.nig.start[3],IFNC.nig.start[4]))
vg.IFNC <- vgFit(LR_IFNC,paramStart=c(IFNC.vg.start[1],IFNC.vg.start[2],
      IFNC.vg.start[3],IFNC.vg.start[4]))

# INDX
hyperb.INDX <- hyperbFit(LR_INDX,paramStart=c(INDX.hyperb.start[1],INDX.hyperb.start[2],
      INDX.hyperb.start[3],INDX.hyperb.start[4]))
nig.INDX <- nigFit(LR_INDX,paramStart=c(INDX.nig.start[1],INDX.nig.start[2],
      INDX.nig.start[3],INDX.nig.start[4]))
vg.INDX <- vgFit(LR_INDX,paramStart=c(INDX.vg.start[1],INDX.vg.start[2],
      INDX.vg.start[3],INDX.vg.start[4]))

# ICON
hyperb.ICON <- hyperbFit(LR_ICON,paramStart=c(ICON.hyperb.start[1],ICON.hyperb.start[2],
      ICON.hyperb.start[3],ICON.hyperb.start[4]))
nig.ICON <- nigFit(LR_ICON,paramStart=c(ICON.nig.start[1],ICON.nig.start[2],

```

```

        ICON.nig.start[3],ICON.nig.start[4]))
vg.ICON <- vgFit(LR_ICON,paramStart=c(ICON.vg.start[1],ICON.vg.start[2],
        ICON.vg.start[3],ICON.vg.start[4]))

# Teste Anderson-Darling
# IFNC
ad.test(LR_IFNC,"hyperb",mu=hyperb.IFNC$param[1],
        delta=hyperb.IFNC$param[2],
        alpha=hyperb.IFNC$param[3],
        beta=hyperb.IFNC$param[4])

ad.test(LR_IFNC,"nig",mu=nig.IFNC$param[1],
        delta=nig.IFNC$param[2],
        alpha=nig.IFNC$param[3],
        beta=nig.IFNC$param[4])

ad.test(LR_IFNC,"vg",param=c(vg.IFNC$param[1],vg.IFNC$param[2],
        vg.IFNC$param[3],vg.IFNC$param[4]))

# INDX
ad.test(LR_INDX,"hyperb",mu=hyperb.INDX$param[1],
        delta=hyperb.INDX$param[2],
        alpha=hyperb.INDX$param[3],
        beta=hyperb.INDX$param[4])

ad.test(LR_INDX,"nig",mu=nig.INDX$param[1],
        delta=nig.INDX$param[2],
        alpha=nig.INDX$param[3],
        beta=nig.INDX$param[4])

ad.test(LR_INDX,"vg",param=c(vg.INDX$param[1],vg.INDX$param[2],
        vg.INDX$param[3],vg.INDX$param[4]))

# ICON
ad.test(LR_ICON,"hyperb",mu=hyperb.ICON$param[1],
        delta=hyperb.ICON$param[2],
        alpha=hyperb.ICON$param[3],
        beta=hyperb.ICON$param[4])

```

```

ad.test(LR_ICON,"nig",mu=nig.ICON$param[1],
        delta=nig.ICON$param[2],
        alpha=nig.ICON$param[3],
        beta=nig.ICON$param[4])

ad.test(LR_ICON,"vg",param=c(vg.ICON$param[1],vg.ICON$param[2],
                             vg.ICON$param[3],vg.ICON$param[4]))

# Kolmogorov-Smirnov

# Simulações para plot
# IFNC
plot(density(LR_IFNC),lwd=1,main="Densidade")
lines(density(rt.scaled(5000,df=fit.t.IFNC$estimate[1],
                      mean= fit.t.IFNC$estimate[2],
                      sd=fit.t.IFNC$estimate[3])),
      lty="dotted",col="blue")
lines(density(rhyperb(5000,param=c(hyperb.IFNC$param[1],hyperb.IFNC$param[2],
                                   hyperb.IFNC$param[3],hyperb.IFNC$param[4])),
      lty="dotted",col="green")
lines(density(rnig(5000,param=c(nig.IFNC$param[1],nig.IFNC$param[2],
                                nig.IFNC$param[3],nig.IFNC$param[4])),
      lty="dotted",col="red")
lines(density(rvg(5000,param=c(vg.IFNC$param[1],vg.IFNC$param[2],
                               vg.IFNC$param[3],vg.IFNC$param[4])),
      lty="dotted",col="orange")
legend("topright", legend=c("t", "Hiperbólica",
                            "Normal-Inversa de Gauss", "Variância-Gama"),
      col=c("blue", "green", "red", "orange"), lty="dotted", cex=0.5)

# INDX
plot(density(LR_INDX),lwd=1,main="Densidade")
lines(density(rt.scaled(5000,df=fit.t.INDX$estimate[1],
                      mean= fit.t.INDX$estimate[2],
                      sd=fit.t.INDX$estimate[3])),
      lty="dotted",col="blue")
lines(density(rhyperb(5000,param=c(hyperb.INDX$param[1],hyperb.INDX$param[2],
                                   hyperb.INDX$param[3],hyperb.INDX$param[4])),
      lty="dotted",col="green")

```

```

lines(density(rnig(5000,param=c(nig.INDX$param[1],nig.INDX$param[2],
                             nig.INDX$param[3],nig.INDX$param[4])),
        lty="dotted",col="red")
lines(density(rvg(5000,param=c(vg.INDX$param[1],vg.INDX$param[2],
                             vg.INDX$param[3],vg.INDX$param[4])),
        lty="dotted",col="orange")
legend("topright", legend=c("t", "Hiperbólica",
                             "Normal-Inversa de Gauss", "Variância-Gama"),
        col=c("blue", "green", "red", "orange"), lty="dotted", cex=0.5)

# ICON
plot(density(LR_ICON),lwd=1,main="Densidade")
lines(density(rt.scaled(5000,df=fit.t.ICON$estimate[1],
                      mean= fit.t.ICON$estimate[2],
                      sd=fit.t.ICON$estimate[3])),
        lty="dotted",col="blue")
lines(density(rhyperb(5000,param=c(hyperb.ICON$param[1],hyperb.ICON$param[2],
                                 hyperb.ICON$param[3],hyperb.ICON$param[4])),
        lty="dotted",col="green")
lines(density(rnig(5000,param=c(nig.ICON$param[1],nig.ICON$param[2],
                              nig.ICON$param[3],nig.ICON$param[4])),
        lty="dotted",col="red")
lines(density(rvg(5000,param=c(vg.ICON$param[1],vg.ICON$param[2],
                              vg.ICON$param[3],vg.ICON$param[4])),
        lty="dotted",col="orange")
legend("topright", legend=c("t", "Hiperbólica",
                             "Normal-Inversa de Gauss", "Variância-Gama"),
        col=c("blue", "green", "red", "orange"), lty="dotted", cex=0.5)

# Encontrar as cópulas
# Pseudo-Observações Empíricas
epobs <- pobs(cbind(LR_IFNC, LR_INDX, LR_ICON))

# Cópula selecionada
# IFNC x INDX
emp_1 <- BiCopSelect(epobs[,1],epobs[,2],selectioncrit="logLik")
# IFNC x ICON
emp_2 <- BiCopSelect(epobs[,1],epobs[,3],selectioncrit="logLik")
# INDX x ICON

```

```

emp_3 <- BiCopSelect(epobs[,2],epobs[,3],selectioncrit="logLik")

# Simulações para plot
sim_1 <- rCopula(n=length(LR_IFNC),BB1Copula(param=c(emp_1$par,emp_1$par2)))
sim_2 <- rCopula(n=length(LR_IFNC),surBB7Copula(param=c(emp_2$par,emp_2$par2)))
sim_3 <- rCopula(n=length(LR_IFNC),BB7Copula(param=c(emp_3$par,emp_3$par2)))

# Plot
p1 <- qplot(sim_1[,1], sim_1[,2], colour = sim_1[,1],
            main="Cópula BB1",
            xlab = "u", ylab = "v")
p2 <- qplot(sim_2[,1], sim_2[,2], colour = sim_2[,1],
            main="Cópula Survival BB7",
            xlab = "u", ylab = "v")
p3 <- qplot(sim_3[,1], sim_3[,2], colour = sim_3[,1],
            main="Cópula BB7",
            xlab = "u", ylab = "v")

pushViewport(viewport(layout = grid.layout(1, 3)))
print(p1, vp = viewport(layout.pos.row = 1, layout.pos.col = 1))
print(p2, vp = viewport(layout.pos.row = 1, layout.pos.col = 2))
print(p3, vp = viewport(layout.pos.row = 1, layout.pos.col = 3))

# Pseudo-Observações com a distribuição teórica
set.seed(123)
pobs <- pobs(cbind(rt.scaled(n=5000,df=fit.t.IFNC$estimate[1],
                           mean=fit.t.IFNC$estimate[2],
                           sd=fit.t.IFNC$estimate[3]),
                rt.scaled(n=5000,df=fit.t.INDX$estimate[1],
                           mean=fit.t.INDX$estimate[2],
                           sd=fit.t.INDX$estimate[3]),
                rt.scaled(n=5000,df=fit.t.ICON$estimate[1],
                           mean=fit.t.ICON$estimate[2],
                           sd=fit.t.ICON$estimate[3])))

pobs_IFNC <- pobs[,1]
pobs_INDX <- pobs[,2]
pobs_ICON <- pobs[,3]

# Cópula selecionada

```

```

# IFNC x INDX
teo_1 <- BiCopSelect(pobs_IFNC,pobs_INDX,selectioncrit="logLik")
# IFNC x ICON
teo_2 <- BiCopSelect(pobs_IFNC,pobs_ICON,selectioncrit="logLik")
# INDX x ICON
teo_3 <- BiCopSelect(pobs_INDX,pobs_ICON,selectioncrit="logLik")

# Simulações para plot
sim_1 <- rCopula(n=20000,frankCopula(param=teo_1$par))
sim_2 <- rCopula(n=2000,r90TawnT2Copula(param=c(teo_2$par,teo_2$par2)))
sim_3 <- rCopula(n=2000,tawnT2Copula(param=c(teo_3$par,teo_3$par2)))

# Plot
p1 <- qqplot(sim_1[,1], sim_1[,2], colour = sim_1[,1],
             main="Cópula Frank",
             xlab = "u", ylab = "v")
p2 <- qqplot(sim_2[,1], sim_2[,2], colour = sim_2[,1],
             main="Cópula Rotated Tawn T2 90 degrees",
             xlab = "u", ylab = "v")
p3 <- qqplot(sim_3[,1], sim_3[,2], colour = sim_3[,1],
             main="Cópula Tawn T2",
             xlab = "u", ylab = "v")

pushViewport(viewport(layout = grid.layout(1, 3)))
print(p1, vp = viewport(layout.pos.row = 1, layout.pos.col = 1))
print(p2, vp = viewport(layout.pos.row = 1, layout.pos.col = 2))
print(p3, vp = viewport(layout.pos.row = 1, layout.pos.col = 3))

# Goodness of Fit
gof_emp_1 <- gofCopula(x=cbind(epobs[,1],epobs[,2]),VC2copula::BB1Copula(param=c(emp_1$par,emp_1$par2)))
gof_emp_2 <- gofCopula(x=cbind(epobs[,1],epobs[,3]),VC2copula::surBB7Copula(param=c(emp_2$par,emp_2$par2)))
gof_emp_3 <- gofCopula(x=cbind(epobs[,2],epobs[,3]),VC2copula::BB7Copula(param=c(emp_3$par,emp_3$par2)))
gof_teo_1 <- gofCopula(x=cbind(pobs[,1],pobs[,2]),frankCopula(param=teo_1$par))
gof_teo_2 <- gofCopula(x=cbind(pobs[,1],pobs[,3]),VC2copula::r90TawnT2Copula(param=c(teo_2$par,teo_2$par2)))

```


gof_teo_3

<-

```
gofCopula(x=cbind(pobs[,2],pobs[,3]),VC2copula::tawnT2Copula(param=c(teo_3$par,teo_3$par2)))
```