Modelaje de dependencia con cópulas bivariadas en R
Nivel: Intermedio

Rodrigo Meneses

3 de octubre de 2019

Introducción

Las *cópulas bivariadas* son funciones que ligan una función de distribución bivariada con un par de correspondientes funciones de distribución marginales.

Dicha cópula contiene *información acerca de la dependencia* existente entre las variables aleatorias que tienen como funciones de distribución a las marginales.

Según el teorema de Sklar (1959), una cópula que ligue dicha función de distribución bivariada con sus marginales siempre existe y más aún, se dan las condiciones para que dicha cópula sea única, en el caso menos general, cuando las marginales son ambas continuas.

La utilidad principal de las cópulas radica en el modelaje de la dependencia, conocer la cópula y las marginales provee la capacidad de entender la estructura de dependencia que subyace entre las variables aleatorias (los procesos que interactúan y que nos interesa comprender) y conseguir una descripción más avanzada del fenómeno o sucesos de estudio que presentan aleatoriedad (no existe previamente una función que determine completamente la relación entre procesos).

Véase [1] para mayores referencias.

Ventajas y desventajas de modelar dependencia con cópulas

Desventajas

En general, el modelaje de la dependencia mediante cópulas implica un esfuerzo grande y puede no llevar a un resultado satisfactorio para el analista, por el teorema de Sklar, la cópula siempre existe y existen condiciones de unicidad, sin embargo, para conocer de forma correcta y 'útil' a la estructura de dependencia, es necesario el conocimiento de las marginales, es decir, conocer las funciones de distribución de forma precisa. En ese punto el analista puede sentirse frustrado en tanto que no sea capaz de ajustar una distribución conocida a los datos con los que cuenta y perder sus ilusiones en un modelo incompleto de dependencia. Claro está que a su disposición se encuentran una gran cantidad de métodos para transformar los datos y poder ajustarlos a distribuciones conocidas, pero, no siempre será posible.

Aunado a lo anterior, es imprescindible el conocimiento de un catálogo de familias de cópulas que le permita conocer los detalles de las mismas, vea por ejemplo [2] o [3]. Pese a ello, existen familias de cópulas de reciente creación para las cuales la información teórica puede no ser fácilmente accesible, aun así, será posible explotarlas computacionalmente, pero se insiste en el valor que tiene la teoría para que la experiencia en el uso de cópulas sea mejor.

Ventajas

Principalmente, la capacidad de crear un modelo de dependencia de alto nivel para el cual se puede visualizar la región tridimensional que lo compone, se pueden simular datos a través de la cópula, y se puede acoplar a modelos compuestos.

Especialmente, el analista cuyo interés sea el 'pronóstico', encontrará en el modelaje de dependencia con cópulas un fuerte argumento matemático para sostener sus estimaciones.

De forma indirecta, el analista que observa el fenómeno de estudio tendrá una sensación de mayor certeza y comodidad acerca de su conocimiento respecto al fenómeno en sí, a sus interacciones y a su naturaleza misma.

Modelando dependencia con cópulas bivariadas en R

En lo subsecuente se asumirá que por cópula se entiende cópula bivariada.

Recursos

Actualmente R cuenta con dos paqueterías fundamentales en el modelaje de dependencia con cópulas, las paqueterías:

- VineCopula: debida a **Thomas Nagler** y coautores como **Ulf Schepsmeier**, **Jakob Stoeber**, **Eike Christian Brechmann** (así es, el autor de [3]), entre otros.
- copula: debida a **Martin Maechler** y coautores como **Marius Hofert**, **Ivan Kojadinovic** y **Jun Yan**.

Aunque existen otras paqueterías relacionadas, estas serán las principales.

Instalar paqueterías

Para tener siempre listas las paqueterías que se utilizarán, abra un script y coloque el siguiente header.

```
## ---- librerias------------------------------------------------------
list.of.packages <- c("kableExtra","lubridate","ggplot2","gridExtra","dplyr",
                      "VineCopula","copula")
new.packages <- list.of.packages[!(list.of.packages %in% installed.packages()[,"Package"])]
if(length(new.packages)) install.packages(new.packages)

require(kableExtra)
require(lubridate)
require(ggplot2)
require(gridExtra)
require(dplyr)
require(VineCopula)
require(copula)
```

Con ello instalará las librerías si no se encuentran ya instaladas y las activará.

Caso de estudio: IPC (índice ^MXX) VS Coca-Cola FEMSA (acción KOF)

Se descubrirá la relación del IPC contra la acción KOF de FEMSA entre las fechas 19/09/2018 - 19/09/2019 a través de su rendimiento, **los datos provienen de Yahoo Finance**.

Se le invita a meditar un poco con base en su experiencia respecto a cómo debería ser esta relación, a continuación, se realiza la importación de los datos tal cual han sido descargados, así como una limpieza de ambos con intención de contar con lo necesario en un formato útil.

Se tomará el precio de cierre para calcular los rendimientos de ambos activos.

```r
# Si no encuentra en la carpeta donde está la información,
# utilice el siguiente set

# setwd(choose.dir(caption = "Directorio donde se encuntra la
#                                 información")) # Windows

# setwd(tk_choose.dir(caption = "Directorio donde se encuntra la
#                                 información")) # Linux

ipc<-read.csv("./^MXX.csv")
kof<-read.csv("./kof.csv")

## Limpieza de datos

ipc<-ipc[,c("Date","Close")]
kof<-kof[,c("Date","Close")]

ipc<-rename(ipc,"Fecha"="Date","Cierre"="Close")
kof<-rename(kof,"Fecha"="Date","Cierre"="Close")


kof$Fecha<-gsub("/","-",kof$Fecha)

valores.fecha<-strsplit(kof$Fecha,"-")

nuevas.fechas<-c()
for(i in 1:length(valores.fecha)){
  nuevas.fechas[i]<-paste0(valores.fecha[[i]][3],"-",valores.fecha[[i]][2],"-",
                           valores.fecha[[i]][1])
}

kof$Fecha<-nuevas.fechas

kof$Cierre<-sub(",","",kof$Cierre)

kof$Cierre<-as.numeric(kof$Cierre)
ipc$Cierre<-as.numeric(ipc$Cierre)

#### Ordenando valores del IPC respecto a fecha

ipc<-ipc[order(ipc$Fecha,decreasing = TRUE),]
kof<-kof[order(kof$Fecha,decreasing = TRUE),]

# Corrigiendo valores extremos

v.e.ipc<-boxplot(ipc$Cierre,plot=FALSE)$out
posiciones<-which(ipc$Cierre %in% v.e.ipc)
if(length(posiciones)!=0){ipc<-ipc[-posiciones,]}

v.e.kof<-boxplot(kof$Cierre,plot=FALSE)$out
posiciones<-which(kof$Cierre %in% v.e.kof)
if(length(posiciones)!=0){kof<-kof[-posiciones,]}
```

```r
# En muchas ocasiones, las fechas de las observaciones difieren
# por tema de los mercados, en ese caso, se reúnen ambas bases en
# una sola, ligándolas a través de la más pequeña que en el caso
# actual es la que contiene información del IPC

activos<-left_join(ipc,kof,by="Fecha")
colnames(activos)<-c("Fecha","IPC","KOF")

# Reunir las bases permitirá que los cálculos por enfrentar
# más adelante sean posibles.

# Se calculan los rendimientos logarítmicos para ambos activos a continuación

rendimiento.ipc<-c()
for(i in 1:(nrow(activos)-1)){
  rendimiento.ipc[i]<-log(activos$IPC[i+1]/activos$IPC[i])
}

rendimiento.kof<-c()
for(i in 1:(nrow(activos)-1)){
  rendimiento.kof[i]<-log(activos$KOF[i+1]/activos$KOF[i])
}

rendimientos<-data.frame(IPC=rendimiento.ipc,KOF=rendimiento.kof)
rendimientos<-rendimientos[complete.cases(rendimientos$KOF),]
```

Se observan un par de datos para ambos activos.

```r
info<-head(kof)
row.names(info)<-NULL
kable(head(kof), format = "latex",
      caption = "Muestra de las observaciones del precio de cierre KOF",
      booktabs=TRUE, longtable=TRUE) %>%
  kable_styling(latex_options = "HOLD_position")
```

Tabla 1: Muestra de las observaciones del precio de cierre KOF

	Fecha	Cierre
254	2019-10-02	59.65
253	2019-10-01	59.62
252	2019-09-30	60.62
251	2019-09-27	60.55
250	2019-09-26	60.90
249	2019-09-25	62.06

```r
info<-head(ipc)
row.names(info)<-NULL
kable(info, format = "latex",
```

```
      caption = "Muestra de las observaciones del valor de cierre del IPC",
      booktabs=TRUE, longtable=TRUE) %>%
  kable_styling(latex_options = "HOLD_position")
```

Tabla 2: Muestra de las observaciones del valor de cierre del IPC

Fecha	Cierre
2019-09-19	43017.46
2019-09-18	43070.34
2019-09-17	43448.94
2019-09-13	42841.46
2019-09-12	42670.41
2019-09-11	42749.17

Para comprender parte de cómo interactúan los datos, es importante generar un gráfico de dispersión y un índice de correlación.

```
correlacion<-cor(rendimientos[,"KOF"],rendimientos[,"IPC"],method="pearson")
names(correlacion)<-"Correlacion"

kable(correlacion,col.names = NULL, format = "latex",
      caption = "Índice de correlación de Pearson",booktabs=TRUE,
      longtable=TRUE) %>%
  kable_styling(c("striped","bordered"),latex_options = "HOLD_position")
```

Tabla 3: Índice de correlación de Pearson

Correlacion	0.6530816

```
pairs(rendimientos[,c("KOF","IPC")])
```

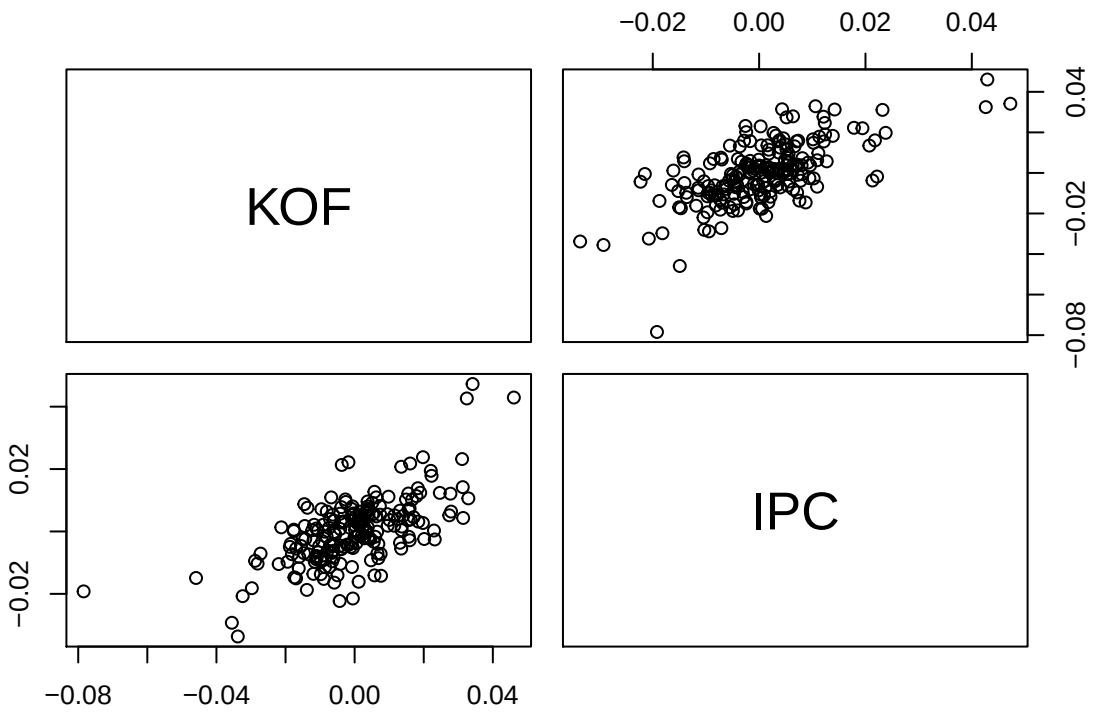

Figura 1: Gráfica de correlación

Se puede observar una correlación positiva entre las variables según el índice de Pearson, lo cual económicamente está sustentado (se le invita a investigar el por qué), se procederá a analizar el comportamiento individual de las variables a través de los rendimientos.

Se observan las frecuencias de los rendimientos.

```
  p1<-ggplot()+geom_histogram(aes(scale(rendimientos$IPC)))+
  theme_bw()+labs(x="Rendimientos IPC",y="Frecuencia")+stat_bin(bins = 30)
  p2<-ggplot()+geom_histogram(aes(scale(rendimientos$KOF)))+
  theme_bw()+labs(x="Rendimientos KOF",y="Frecuencia")+stat_bin(bins = 30)

grid.arrange(p1,p2,nrow = 2,ncol = 1)
```

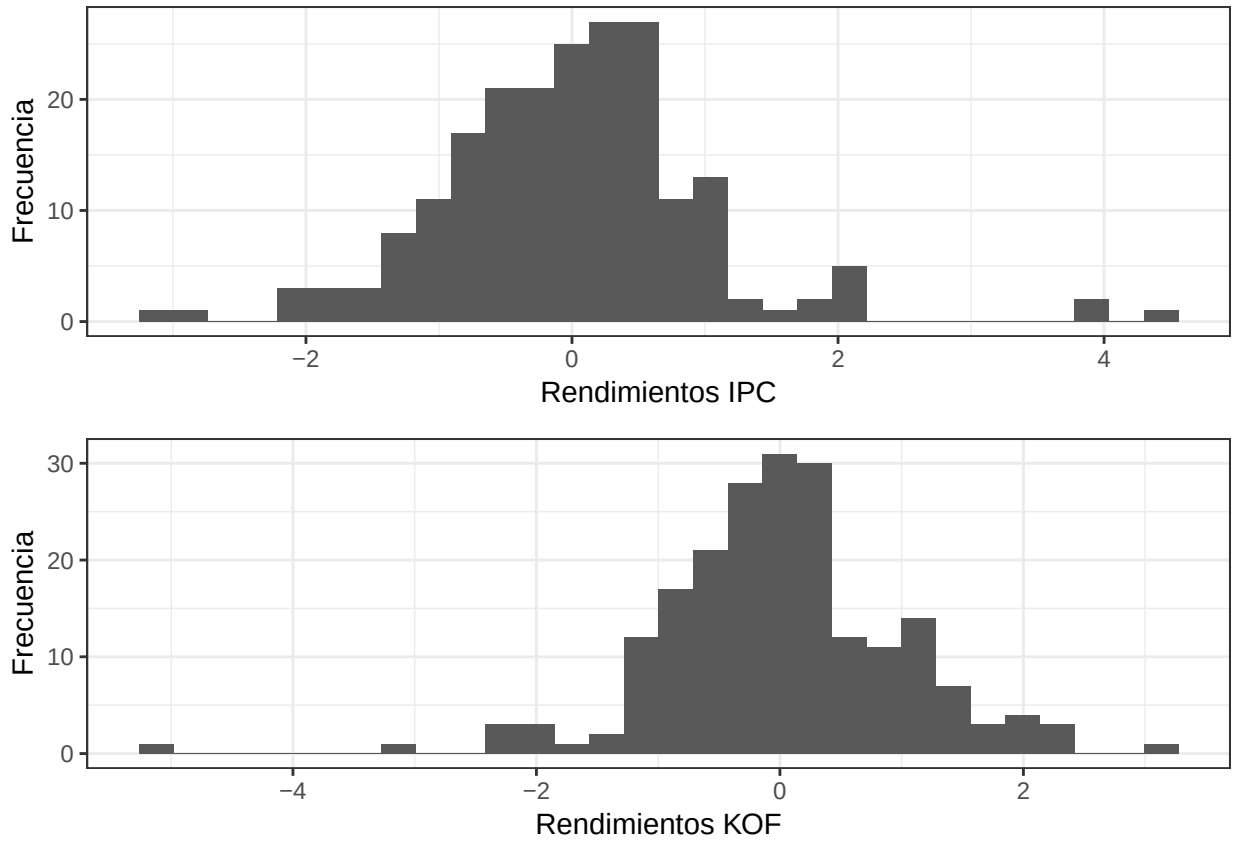

Figura 2: Frecuencias

Los rendimientos del KOF y el IPC es la información para la cual se desea modelar dependencia a través de cópulas y, por lo tanto, las funciones de densidad para los rendimientos de cada activo corresponden a las marginales necesarias para la estructuración del modelo de dependencia conforme al teorema de Sklar. Como se mencionó en un inicio, la complejidad del modelo de dependencia radica en el conocimiento de las marginales, dicho conocimiento se expresa propiamente en que existan funciones de densidad explícitas, en otro caso, el modelo podría convertirse en una tarea bastante compleja.

Dado que no es finalidad del presente texto describir métodos para ajustar distribuciones conocidas, por comodidad se asumirá sin prueba que ambos rendimientos tienen una distribución normal, **es muy importante** que el lector comprenda que este supuesto sin prueba es inválido y lo conducirá a conclusiones posiblemente erróneas, lo mejor es realizar un correcto uso de pruebas de hipótesis para que los datos puedan ser ajustados a distribuciones conocidas *siempre que sea posible*, los datos podrían ser más similares a una distribución beta, por ejemplo.

Ajustando las marginales

Se obtiene a continuación los parámetros necesarios para dos distribuciones normales correspondientes al KOF y al IPC.

```
# Parámetros para las marginales
```

```
## IPC

u_ipc<-round(mean(rendimientos$IPC),3)
sigma_ipc<-round(sd(rendimientos$IPC),3)

## KOF

u_kof<-round(mean(rendimientos$KOF),3)
sigma_kof<-round(sd(rendimientos$KOF),3)
```

Se procede a graficar mediante histogramas, se tienen dos distribuciones $N(0, 0.011)$ para el IPC y $N(0, 0.015)$ para el KOF.

```
hist(as.numeric(rendimiento.kof),breaks = 80,main = NULL,freq = F,
     density = 30,col="blue",xlab = "Rendimiento del KOF",ylab = "Densidad")
lines(seq(-1,1,0.001),dnorm(seq(-1,1,0.001),u_kof,sigma_kof),col="red",lwd=2)
legend('topright',"N(0,0.015)",col = "red",lwd=2)
```

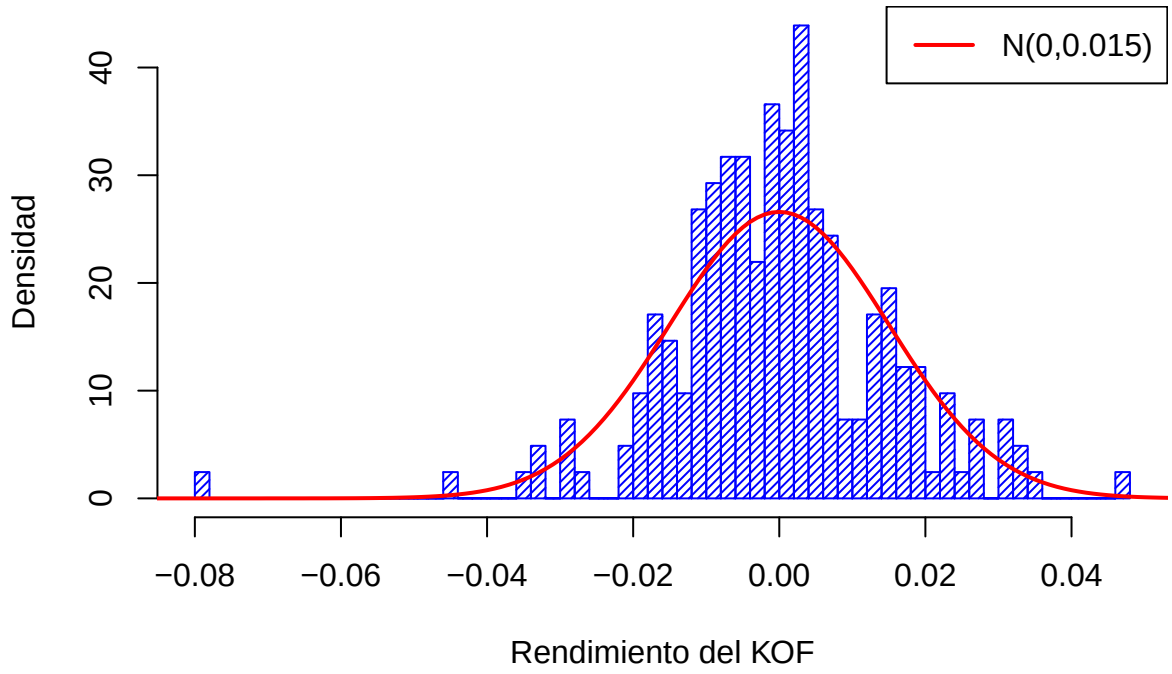

Figura 3: Densidades de probabilidad (rendimientos del KOF)

```
hist(as.numeric(rendimiento.ipc),breaks = 80,main = NULL,freq = F,
     density = 30,col="blue",xlab = "Rendimiento del IPC",ylab = "Densidad")
lines(seq(-1,1,0.001),dnorm(seq(-1,1,0.001),u_ipc,sigma_ipc),col="red",lwd=2)
legend('topright',"N(0,0.011)",col = "red",lwd=2)
```

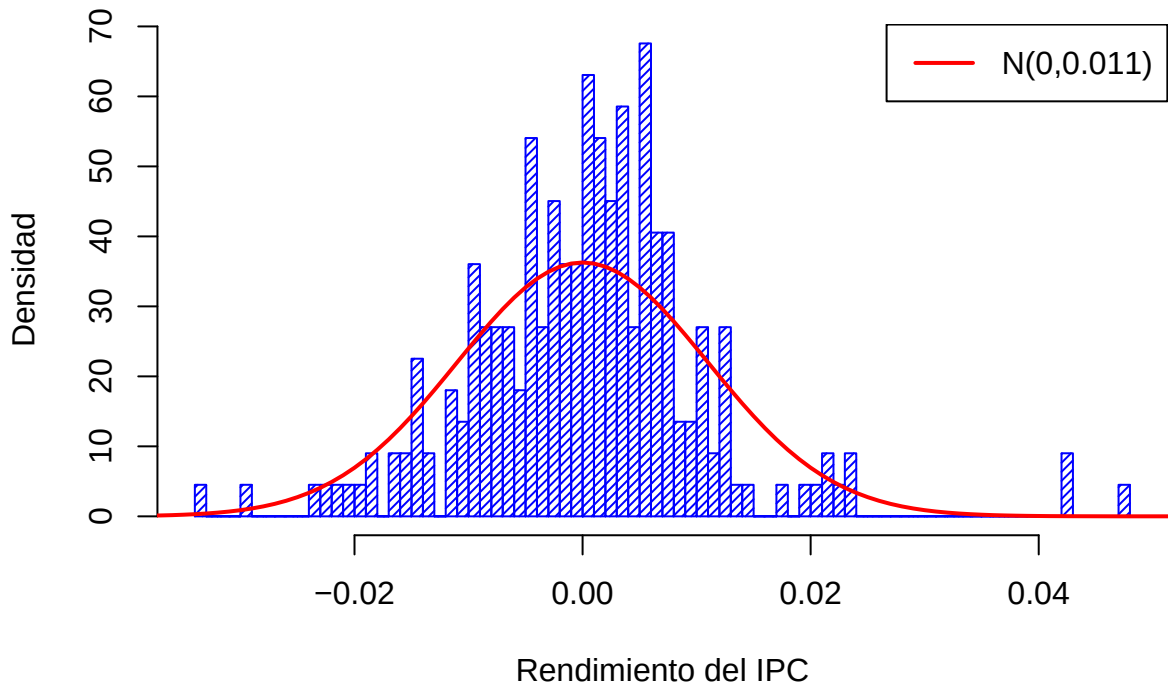

Figura 4: Densidades de probabilidad (rendimientos del IPC)

Como se puede apreciar, las distribuciones normales propuestas (las líneas en rojo) recolectan relativamente bien la información de la volatilidad aunque la información en las colas no tanto, además fallan en un punto; dicho punto es la curtosis pues no captan bien dicha información en ninguno de los dos casos, lo cual puede ser un problema en tanto que los valores centrales están subestimados y valores cuya importancia puede ser nula están recibiendo cierta porción de la densidad de los valores subestimados. Lo anterior conduce a resultados que pueden diferir de la realidad, no necesariamente erróneos, pues eso depende del analista y hasta qué punto puede otorgarse el derecho de ser impreciso.

Selección de la cópula

La forma óptima de seleccionar la cópula es a través del AIC y BIC, BIC resolverá cuál es la mejor cópula respecto a todas las que se encuentran disponibles y AIC resolverá la que mejor se ajuste respecto a los datos con los que se cuenta.

Dado que se trata con datos no continuos y que la cópula que se obtendrá en el paso anterior debe ser única, se tendrá que hacer uso de funciones cuasi-inversas para que la cópula esté bien definida y en ese sentido:

prevalezca la unicidad.

Las funciones cuasi-inversas están definidas sobre el intervalo $[0, 1]$ y serán construidas a partir de los datos con los que se cuenta para llegar a funciones marginales uniformes y poder aplicar el Teorema de Sklar de tal modo que la cópula definida sea única en todo el conjunto de datos, véase la definición 2.8 y el corolario 2.9 en [1].

Los datos transformados a través de estas funciones cuasi-inversas reciben el nombre de pseudo-observaciones y su importancia es vital en tanto que permiten la unicidad de la cópula seleccionada para el conjunto de datos marginales, de eludir este paso, la cópula seleccionada podría estar mal definida para ciertos subconjuntos de datos.

Para realizar lo anterior, se hará uso de dos funciones: `pobs` y `BiCopSelect`, ambas de la paquetería `VineCopula`, la primera permitirá convertir los datos de los rendimientos en pseudo-observaciones y la segunda permitirá seleccionar la mejor cópula según AIC y BIC.

```
# Pseudo-observaciones

u<-pobs(as.matrix(rendimientos))[,1]
v<-pobs(as.matrix(rendimientos))[,2]


# Selección de la cópula

s.copula<-BiCopSelect(u,v,familyset = NA)
s.copula
```

```
## Bivariate copula: BB7 (par = 1.72, par2 = 0.79, tau = 0.44)
```

```
s.copula<-BB7Copula(param = c(1.72,0.79))
```

```
## Se fija una semilla
set.seed(123)

## Se sentencian las pseudo-observaciones

pseudos<-pobs(as.matrix(rendimientos))

## El modelo estará basado en una cópula Joe-Clayton (BB7)

modelo<-fitCopula(s.copula,pseudos,method="mpl")
```

La función `fitCopula` permite ajustar la cópula seleccionada, en este caso la cópula Joe-Clayton BB7 (véase [4] para más detalles sobre la misma) cuyos parámetros son $theta = 1.7234$ y $delta = 0.7942$, dicho ajuste se realiza con las pseudo-observaciones por lo tanto, es importante colocar el argumento `method="mpl"` (del inglés *Maximum pseudo-likelihood estimator*), el cual implica que el estimador paramétrico de la cópula para el ajuste estará basado en pseudo-observaciones y no en información real.

A continuación, se muestra la información completa de la cópula.

```
modelo
```

```
## Call: fitCopula(copula, data = data, method = "mpl")
## Fit based on "maximum pseudo-likelihood via BiCopEst" and 205 2-dimensional observations.
```

```
## Copula: BB7Copula
##  theta  delta
## 1.7234 0.7942
## The maximized loglikelihood is 57.68
```

Finalmente, se puede observar un gráfico tridimensional compuesto por las pseudo-observaciones y la densidad resultante de aplicar las cuasi-inversas en la cópula.

```
persp(s.copula,dCopula)
```

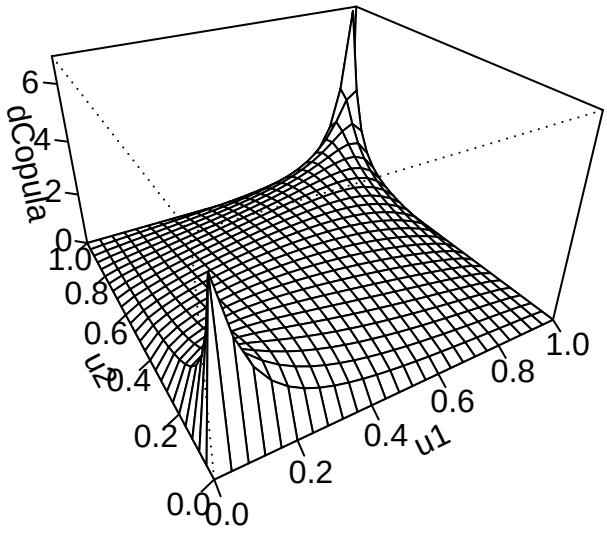

Figura 5: Región tridimensional de la cópula y pseudo-observaciones

Es observable en dicha región la existencia de correlación positiva entre el IPC y la acción KOF, lo cual tiene un gran sustento dado que la *sanidad* del IPC tiene una gran relación con el crecimiento de FEMSA y KOF es una de sus unidades de negocio.

El modelo de dependencia

Ya que se han descrito las marginales y se ha seleccionado la cópula, lo restante es ensamblar ambos elementos en un modelo de dependencia, es decir, obtener la función de distribución conjunta; el producto prometido por el teorema de Sklar.

Para lo anterior, se hará uso de la función `mvdc`, la cual hace el cálculo de la función de distribución conjunta a través de la cópula y las marginales.

```
joint.dist<-mvdc(s.copula,margins = c("norm","norm"),
paramMargins=list(list(mean=u_kof,sd=sigma_kof),
                  list(mean=u_ipc,sd=sigma_ipc)))
```

Dependiendo de las marginales que se hayan construído, los argumentos `margins` y `paramMargins` varían, es aceptable además, agregar funciones de distribución definidas por el usuario siempre que existan sus versiones "d","q","p". Por ejemplo, si ha generado una función de distribución llamada `eta`, entonces debe definir `deta`,`peta`,`qeta`, así como para `norm` existen `dnorm`,`qnorm` y `pnorm`. Por otro lado, los parámetros de las distribuciones que seleccione deben ser colocados a través de listas con `paramMargins`.

Con dicha función de distribución conjunta (`joint.dist`) podemos generar datos simulados a través de la función `rMvdc`.

```
## Generamos tantas simulaciones como pseudo-observaciones
## con la distribución conjunta joint.dist

simulaciones<-rMvdc(length(u),joint.dist)
```

Y las simulaciones pueden ser comparadas contra las observaciones de los rendimientos.

```
plot(rendimientos,col="blue")
points(simulaciones[,1],simulaciones[,2],col="red")
```

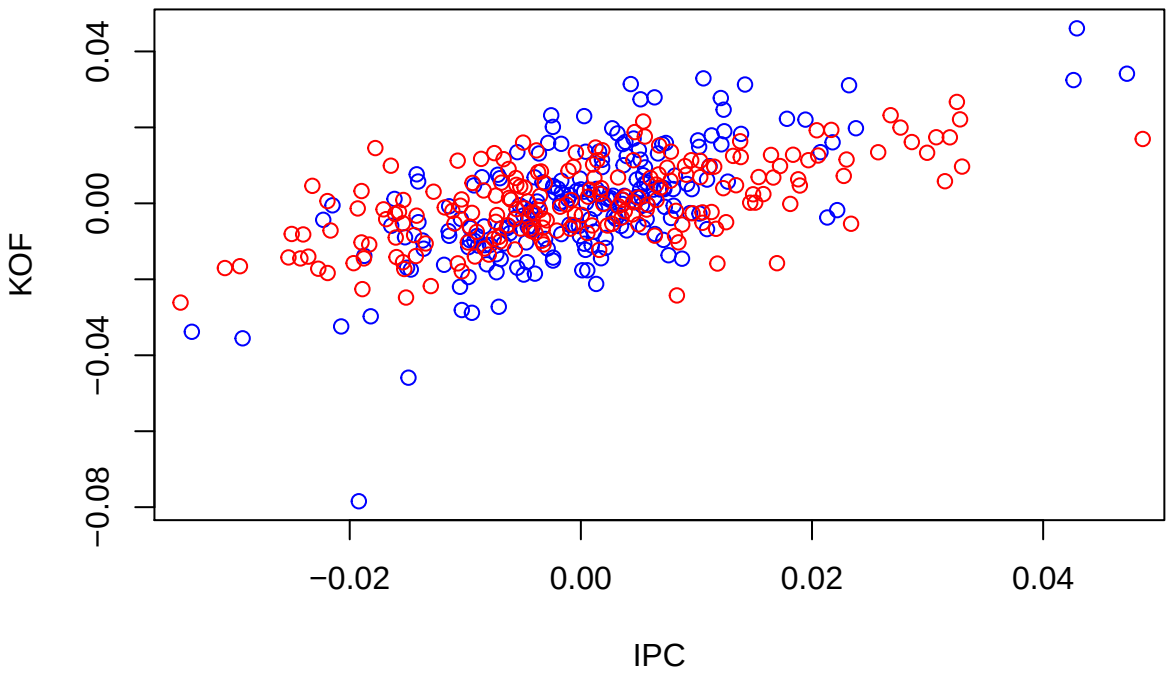

Figura 6: Comparación rendimientos reales contra simulados

Llegar a la función que produce `mvdc` es el paso final, pues ya se cuenta con el modelo de dependencia, la función `rMvdc` será útil para simular información a partir del modelo de dependencia.

Palabras finales

Las cópulas son excelentes instrumentos para el modelaje de la dependencia, pero puede ser confusa su utilidad en situaciones reales, en general, la información que proveen es utilizada para reforzar modelos más 'útiles' como las series de tiempo, pues nos dan un conocimiento mayor sobre la estructura de dependencia real del fenómeno que se estudia, pero en sí mismas no son una solución para la mayoría de los retos, puede ver múltiples aplicaciones en las **referencias** para comprender cómo es que las cópulas intervienen en los problemas reales.

Otro aspecto a considerar es la vigencia, pues la información de la cual se alimentan tanto las marginales como la cópula en sí es estática, por lo tanto, con el transcurrir del tiempo el modelo de dependencia generado podría dejar de reflejar correctamente la realidad, por ello es importante considerar el reajuste de la cópula con el tiempo.

Referencias

[1] Arturo Erdely Ruiz. *Cópulas y dependencia de variables aleatorias: Una introducción.* Universidad Autónoma Metropolitana, Cuajimalpa, México, 2009.

[2] Hans Manner. *Estimation and Model Selection of Copulas with an Application to Exchange Rates.* Universiteit Maastricht, 2007.

[3] Eike Christian Brechmann. *Truncated and simplified regular vines and their applications.* Technische Universitat Munchen, 2010.

[4] Feng Lia; Yanfei Kangb. *Improving forecasting performance using covariate-dependent copula models.* Beihang University, 2018.