Projet de Statistiques 1

Copules et modélisation de la dépendance entre marchés financiers

Taha El Ouahabi & Loic Diridollou

Proposé par Alexis Derumigny

1 Introduction : autour du théorème de Sklar

1 Soit $C_{1,2} = F_{1,2}(F_1^{-1}(u_1), F_2^{-1}(u_2))$
$\forall u_1, u_2 \in [0,1] \implies \exists (x,y) \in (S_1, S_2), u_1 = F_1(x) u_2 = F_2(y)$
Ainsi $\forall u_1, u_2$,

$$\begin{aligned} C_{1,2}(u_1,u_2) &= F_{1,2}(F_1^{-1}(F_1(x)), F_2^{-1}(F_2(y))) \\ &= F_{1,2}(x,y) \end{aligned}$$

On a bien que $C_{1,2}(F_1, F_2) = F_{1,2}$

2 [a] Soit $U_1 = F_1(X_1)$

$$\begin{aligned} \mathbb{P}(U_1 < t) &= \mathbb{P}(F_1(X_1) < t) \\ &= \mathbb{P}(X_1 < F_1^{-1}(t)) \\ &= F_1(F_1^{-1}(t)) \\ &= t \end{aligned}$$

D'où $X_1 \sim \mathcal{U}([0,1])$

[b]

$$\begin{aligned} C_{1,2}(u_1,u_2) &= \mathbb{P}(X_1 < u_1, X_2 < u_2) \\ &= \mathbb{P}(U_1 < u_1, U_2 < u_2) \end{aligned}$$

Par les propriétés de la loi uniforme, on a que $\mathbb{P}(U_1 < u_1, U_2 < u_2) \to 0$ quand $u_1, u_2 \to 0$
On applique le même raisonnement pour la limite en 1 et on a bien le résultat attendu.

[c] D'après le résultat ci-dessus, on a que $C_{1,2}$ est bien une fonction de répartition. De plus avec les propriétés de U_1 et U_2 on a que $C_{1,2}(U_1, U_2) = F_{1,2}$ est bien la fonction de répartition du couple (U_1, U_2) uniforme sur $[0,1]^2$, ce qui provient notamment du fait que la densité se renormalise forcément à 1 en somme sur l'ensemble de définition.

[d] En utilisant à la fois la question a et la question c, on a bien que les marges de $C_{1,2}$ sont uniformes, de plus par propriétés de F_1 et F_2, toutes deux fonctions de répartition, on a bien la dérivabilité en chacun des arguments.
On en déduit donc que $C_{1,2} \in \mathcal{C}$, d'où $C_{1,2}$ est une copule.

3 D'après la première question, on a bien que Γ est surjective.
On sait de plus d'après la question 2.c) que $C_{1,2}$ est la fonction de répartition du vecteur.
De cette façon, il est aisé de retrouver la première composante du vecteur qui est F_1, on retrouve de même F_2.
Il vient immédiatement la surjectivité de la fonction et le résultat attendu.

2 Estimation des copules gaussiennes

1 On a par définition :

$$\begin{aligned}
C_{1,2}(u_1,u_2) &= F_{1,2}(F_1^{-1}(u_1), F_2^{-1}(u_2)) \\
&= P(X_1 \le F_1^{-1}(u_1), X_1 \le F_2^{-1}(u_2)) \\
C_{1,2}(u_1,u_2) &= F_{1,2}(\frac{X_1-\mu_1}{\sigma_1} \le \frac{F_1^{-1}(u_1)-\mu_1}{\sigma_1}, \frac{X_2-\mu_2}{\sigma_2} \le \frac{F_2^{-1}(u_2)-\mu_2}{\sigma_2}) \\
&= \Phi_{1,2}(\frac{F_1^{-1}(u_1)-\mu_1}{\sigma_1}, \frac{F_2^{-1}(u_2)-\mu_2}{\sigma_2})
\end{aligned}$$

Il ne reste plus qu'à monter que $\frac{F_1^{-1}(u_1)-\mu_1}{\sigma_1} = \Phi_1(u_1)$.

$$\begin{aligned}
F_1(\sigma_1\Phi_1(u_1)+\mu_1) &= \mathbb{P}(X_1 \le \sigma_1\Phi_1(u_1)+\mu_1) \\
&= \mathbb{P}(\frac{F_1^{-1}(u_1)-\mu_1}{\sigma_1} \le \Phi_1(u_1)) \\
&= \Phi_1(\Phi_1^{-1}(u_1)) \\
&= u_1, \text{ d'où le résultat}
\end{aligned}$$

De la même façon on a l'égalité pour la seconde composante. Alors $C_{1,2}(u_1,u_2) = F_{1,2}(F_1^{-1}(u_1), F_2^{-1}(u_2))$ ne dépend que de ρ.

2 [a] V_{i1} et V_{i2} sont définies comme U_1 et U_2 ainsi on a l'égalité en loi $\forall i$.
De plus d'après la partie I, 2.c C_ρ est la fonction de distribution du couple (U_1, U_2), elle l'est donc aussi pour $(V_{i,1}, V_{i,2})$.

[b] Soit $F_{i1,i2}$ la fonction de répartition de (X_{i1}, X_{i2}).
On a :

$$\begin{aligned}
F_{i1,i2}(x_1,x_2) &= \mathbb{P}(\Phi_1^{-1}(V_{i1}) \le x_1, \Phi_2^{-1}(V_{i2}) \le x_2) \\
&= \mathcal{C}_\rho(\Phi_1(x_1), \Phi_2(x_2))
\end{aligned}$$

alors :

$$\begin{aligned}
F_{i1,i2}(x_1,x_2) &= \mathcal{C}_\rho(F_1(x_1), F_2(x_2)) \\
&= \mathcal{C}_{1,2}(F_1(x_1), F_2(x_2)) \\
&= F_{1,2}(x_1,x_2)
\end{aligned}$$

3 [a] $\hat{\rho} = \arg\max_\rho \sum_{i=1}^n log(c_\rho(G_1(Y_{i1})), G_2(Y_{i2}))$
Mais on a d'après 2) :

$$(G_1(Y_{i1}), G_2(Y_{i2})) = (u_1,u_2) = (\Phi(X_{i,1}), \Phi(X_{i,2}))$$

D'où on a :

$$\forall \rho, \sum_{i=1}^n log(c_\rho(G_1(Y_{i1})), G_2(Y_{i2})) = \sum_{i=1}^n log(c_\rho(\Phi(X_{i1}), \Phi(X_{i2}))$$

[b] Par définition de c_ρ, on a :

$$
\begin{aligned}
c_\rho &= \frac{\partial^2 C_\rho}{\partial x_1 \partial x_2} \\
f_{1,2}(X_1, x_2) &= \frac{\partial^2 F_{1,2}(x_1, x_2)}{\partial x_1 \partial x_2} \\
&= \frac{\partial^2 F_{1,2}(\Phi^{-1}(\Phi(x_1)), \Phi^{-1}(\Phi(x_1)))}{\partial x_1 \partial x_2} \\
&= \frac{\partial^2 C_\rho(\Phi(x_1), \Phi(x_2))}{\partial x_1 \partial x_2} \\
&= \frac{\partial}{\partial x_1} \frac{\partial C_\rho(\Phi(x_1), \Phi(x_2))}{\partial x_2} \\
&= \frac{\partial}{\partial x_1} \frac{\partial \Phi(x_2)}{\partial x_2} C_{\rho 2}(\Phi(x_1), \Phi(x_2)) \\
&= \phi(x_2) \frac{\partial}{\partial x_1} C_{\rho 2}(\Phi(x_1), \Phi(x_2)) \\
&= \phi(x_2)\phi(x_1) C_\rho(\Phi(x_1), \Phi(x_2))
\end{aligned}
$$

[c] On a d'après la question précédente et la définition de $\hat{\rho}$ Alors

$$
\begin{aligned}
\hat{\rho} &= \arg\max_\rho (\sum_{i=1}^{n} log(\phi(x_1)\phi(x_2)) + \sum_{i=1}^{n} log(f_{1,2}(x_1, x_2)) \\
&= \arg\max_\rho (\sum_{i=1}^{n} log(f_{1,2}(x_1, x_2)) + cste(\rho))
\end{aligned}
$$

Or $cste(\rho) = \sum_{i=1}^{n} log(\phi(x_1)\phi(x_2))$ qui ne dépend pas de ρ.
Alors, la maximisation de $\sum_{i=1}^{n} log(f_{1,2}(x_1, x_2))$ coïncide avec $\hat{\rho}$.
$\hat{\rho}$ est alors un EMV.

[d] $\hat{\rho}$ a alors les propriétés asymptotiques usuelles d'une MLE sous les conditions de régularités (qui sont vérifiées dans le cadre d'un vecteur gaussien) :
- converge en probabilité
- efficace asymptotiquement $\sqrt{n}(\hat{\rho} - \rho) \to \mathcal{N}(o, I^{-1}(\rho))$ donc de variance minimum pour la borne FDCR avec $I_1(\rho)$ l'information de Fischer pour le paramètre ρ.

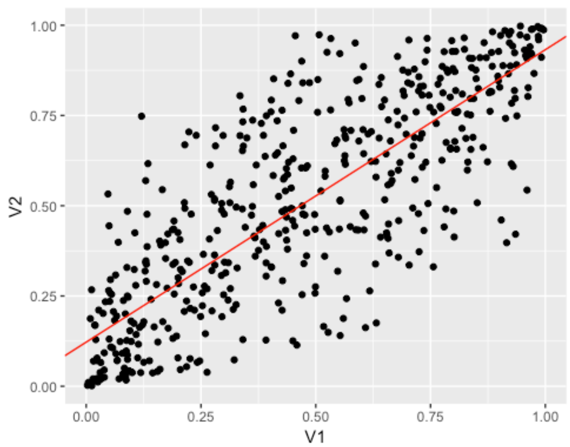
3 Application à des données simulées

```
library(ggplot2)

x <- rnorm(500, 0, 1)
z <- rnorm(500, 0, 1)

m1 <- cbind(x,z)
m2 <- scale(m1)
c1 <- chol(var(m2))
m3 <- m2 %*% solve(c1)
```

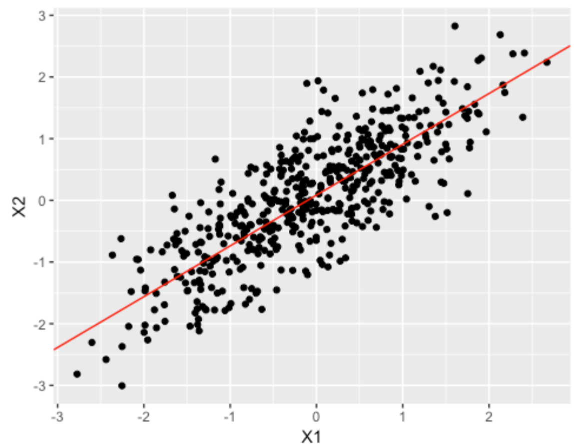
```
# la matrice de corr?lation que tu souhaites obtenir
cor1 <- matrix(c(1,0.8,0.8,1),2)

m4 <- m3 %*% chol(cor1)
m5 <- sweep( m4, 2, attr(m2, 'scaled:scale'), '*')
m5 <- sweep( m5, 2, attr(m2, 'scaled:center'), '+')

cor(m5)   # coefficient de corrÃ©lation

x1 <- m5[,1]
x2 <- m5[,2]

coef(lm(x2 ~ x1)) #pour avoir les coefficients de x2 = a + bx1

qplot(x1,x2,xlab="X1",ylab="X2") + geom_abline(intercept = 0.08506075, slope = 0.82355114,

v1 <- pnorm(x1,0,1)
v2 <- pnorm(x2,0,1)

coef(lm(v2 ~ v1)) #pour avoir les coefficients de v2 = a + b.v1

qplot(v1,v2,xlab="V1",ylab="V2") + geom_abline(intercept = 0.1221782, slope =
0.8102443, color = 'red')
# les points obtenus sont plus dispersés que pour les points (X1,X2) mais on
a quasiment la meme correlation.


qlapl <- function(x) {
#y <- 1:500
for (i in 1:length(x)){
if (x[i]<=1/2) {y <- log(2*x)}
else {y <- -log(2*(1-x))}
}
return(y)
}

y1 <- qcauchy(v1,0,1)
y2 <- qlapl(v2)

#nuage des points
qplot(y1, y2, colour = 'red')
#vu la forme du nuage de points on remarque que la variable y1 ne 'varie' pas
lorsque y2 varie, on en deduit qu'il n y a pas d'interractions entre les deux
variables, d'ou correlation nulle
lm(y2~y1)
#ceci est aussi confirme par le faible coefficient de correlation (de pearson)
qu'on obtient alors un coefficient de correlation de l'ordre de 1.645729e-05 qui
est tres faible,
```

```
#theoriquement, les variables V1 et V2 sont correles, du coup on peut rien dire
concernant la correlation entre les deux lorsque on compose V1 et V2 par les
fonctions G1 inverse et G2 inverse!.
```

2 À la vue du nuage de points, les deux variables sont non conrrélees : Y_1 ne "varie" pas lorsque Y_2 varie. Il n'y a donc pas d'interactions entre les deux variables (elles sont très faibles). Théoriquement on devrait avoir la même dépendance qu'entre X_1 et X_2.



(a) Nuage de points avec V1, V2



(b) Nuage de points avec X1, X2

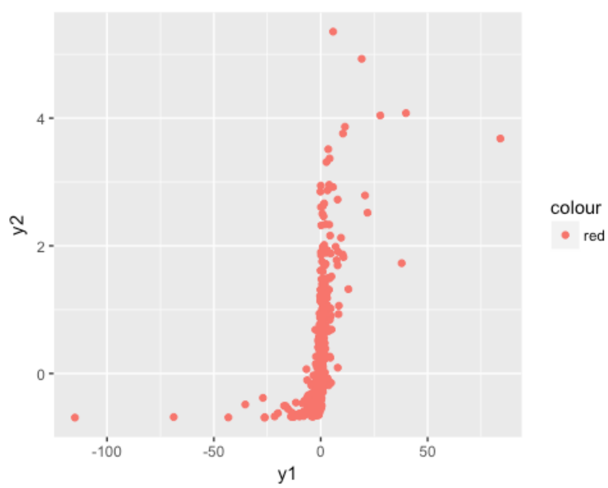


FIGURE 2 – Picture 2

4 Application à des données de marchés financiers

(a) cf. code en fin de document

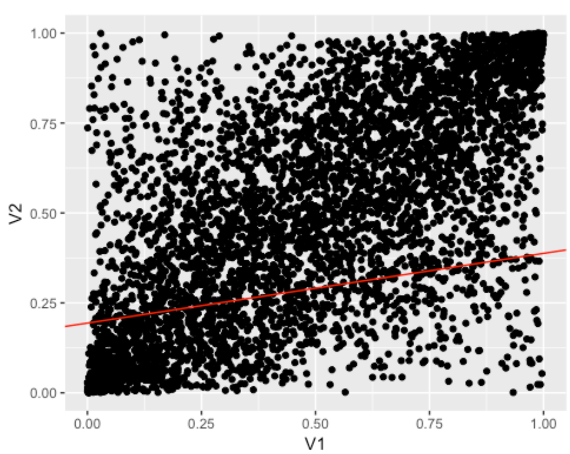
(b) À partir du nuage des points, on peut dire que les deux variables sont très faiblement corrélées au sens de Pearson.

(c) Le paramàtre obtenu ρ est 0.63, très proche de la valeur de la regression de V_2 en V_1.

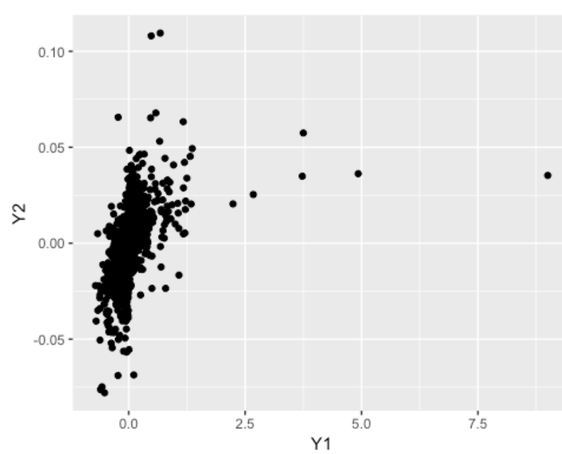
(d) Les lignes de niveaux pour la densité de la copule estimée sur l'échantillon (V_1, V_2) sont très proches en forme du contour de la densite d'un couple gaussien de corrélation $\rho = 0.63$. On peut conclure que c'est une bonne approximation de la copule gaussienne.

```
data1 = read.csv('/Users/taha/Desktop/EDU/Stat1 Projet/^DJI.csv')
data2 = read.csv('/Users/taha/Desktop/EDU/Stat1 Projet/^FCHI.csv')
#on fait un merge sur date pour que les valeurs NULL apparaissent
data = merge(data1, data2, by = 'Date')
library(dplyr)
data = data%>%
filter(Open.y != 'null')
# pour Dow jones on code la variation relative
data$var_cac <- NA
data$var_cac[1] = 0
for (i in (2:length(data$Open.x)))
{data$var_cac[i] = (data$Open.x[i]-data$Open.x[i-1])/data$Open.x[i-1]}

#la var pour CAC 40

data$var_cac1 <- NA
data$var_cac1[1] = 0
for (j in (2:length(data$Open.y)))
{data$var_cac1[j] = (as.double(data$Open.y[j])-as.double(data$Open.y[j-1]))/as.
double(data$Open.y[j-1])}


qplot(data$var_cac1, data$var_cac, xlab="Y1",ylab="Y2")

lm(data$var_cac~data$var_cac1)
# a partir du nuage des points, on peut dire que les deux variables sont très
 faiblement correles au sens de pearson


#Fonction de repartition

dis1 = ecdf(data$var_cac1)
dis2 = ecdf(data$var_cac)
V1 = dis1(data$var_cac1)
V2 = dis2(data$var_cac)

qplot(V1,V2) + geom_abline(intercept = 0.1939, slope = 0.1939, color = 'red')
lm(V1~V2)
#la correlation entre V1 et V2 (au sens de pearson) est beaucoup plus importantes
que pour Y1 et Y2

library(VineCopula)
```

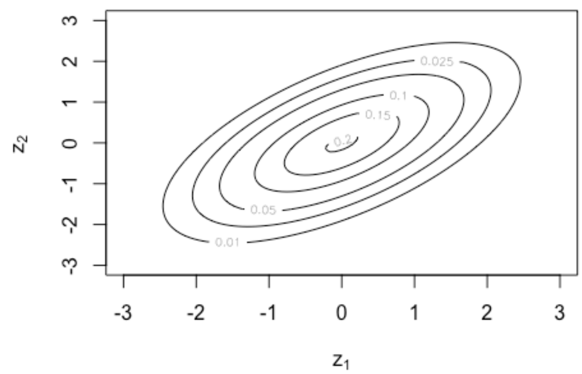
```
ro <- BiCopEst(V1, V2, family = 1, method = "mle")
summary(ro)

# pour le parametre ro = 0.63 on ecrit:
plot <- BiCop(family = 1, par = 0.63) #on simule une Copule gaussienne
plot(plot)

contour(plot)

max(V1)
max(V2)
min(V1)
min(V2)

# on une valeur de V1 ou de egale 1

for (i in 1:length(V1))
if (V1[i] == 1) print(i) # pour i = 2001 on V1 = 1

# on elimine ces observations parce que dans le cas d'une distribution normale, X a des
valeurs infinies car F(X) = 1
fus = cbind(V1,V2)
U = subset(fus, V1 !=1 & V1 != 0 &  V2 !=1 & V2 != 0)

BiCopKDE(U[,1], U[,2], type = "contour")
```



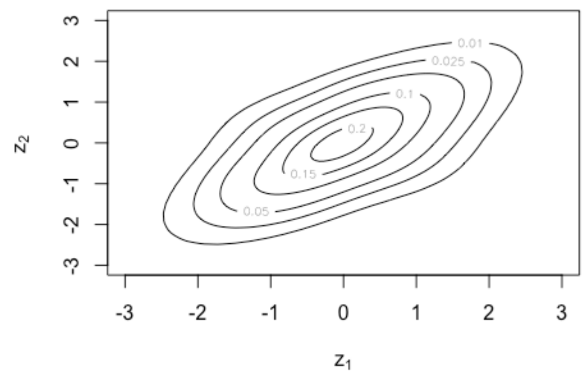
(a) Nuage de points avec V1, V2



(b) Nuage de points avec X1, X2



(a) Densité d'une copule



(b) Estimateur non paramétrique

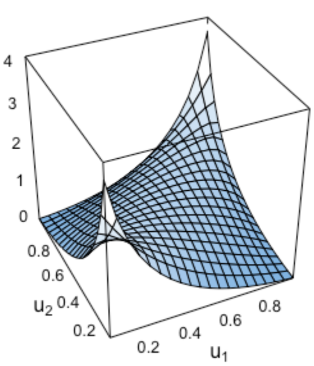


FIGURE 5