Simulación ITAM
10 de diciembre de 2020 2020-II

Tarea 5.

No hay fecha de entrega, son problemas de práctica.Yo subiré soluciones el 1 de diciembre.

Lecturas

- Robert & Casella Capítulos 6 y 7
- Dagpunar Capítulos 5 y 8
- Efron y Gong: A leisurely look at the Boostrap, the Jackknife and Cross-Validation
- Chib y Greenberg: Understanding the Metropolis-Hastings Algorithm
- Casella & George. Explaining the Gibbs sampler.

Problemas

1. Supongan que $Y|\theta \sim \mathcal{G}(1,\theta)$ y que $\theta \sim IG(\alpha,\beta)$.
   - Encuentren la distribución posterior de θ.
   - Encuentren la media y varianza posterior de θ.
   - Encuentren la moda posterior de θ.
   - Escriban dos ecuaciones integrales que se pueden resolver para encontrar el intervalo de 95 % de colas simétricas para θ

   Solución.

   a. Queremos encontrar $\pi(\theta|y)$ conociendo la verosimilitud $\pi(y|\theta)$ y la distrbución inicial $\pi(\theta)$. Por las relaciones entre distribuciones:

   $$\begin{aligned}\pi(\theta|y) = \frac{\pi(y|\theta)\pi(\theta)}{p(y)} &\propto \pi(y|\theta)\pi(\theta)\\ &= \theta^{-1}e^{-y\theta}\theta-(\alpha+1)e^{-\beta\theta}\\ &= \theta^{-(\alpha+2)}e^{\frac{-(y+\beta)}{\theta}}\end{aligned}$$

   Entonces la posterior es $\theta|y \sim IG(\alpha+1,\beta+y)$.

b. A partir del resultado anterior, se tiene que $E(\theta|y) = \frac{1}{\alpha(\beta+y)}$ y $\text{Var}(\theta|y) = \frac{1}{\alpha^2(\beta+y)^2(\alpha-1)}$

c. Para encontrar la moda, definimos el kernel de la distribución como una función de θ y maximizamos (podemos ignorar las constantes). Entonces tomemos $h(\theta) = \theta^k e^{\frac{l}{\theta}}$, con $k = -(\alpha+2)$ y $l = -(y+\beta)$. Derivando se obtiene:

$$k\theta^{k-1}e^{\frac{l}{\theta}} + \theta^k e^{\frac{l}{\theta}}(-l/\theta^2) = 0$$

Simplificando:

$$\theta k - 2e^{l/\theta}(k\theta - l) = 0 \implies \theta* = \frac{l}{k} = \frac{y+\beta}{\alpha+2}$$

d. Las dos ecuaciones corresponden a encontrar los cuantiles $q_{.025}$ y $q_{0.975}$:

$$F_{\theta|y}(q_{0.025}) = 0.025 \text{ y } F_{\theta|y}(q_{0.975}) = 0.975$$

Obviamente estas son las ecuaciones integrales que hay que resolver para tener colas simétricas en la distribución.

□

2. Los siguientes datos corresponden a las horas adicionales de sueño de 10 pacientes tratados con un somnífero B comparado con un somnífero A:

$$1.2, 2.4, 1.3, 1.3, 0, 1, 1.8, 0.8, 4.6, 1.4$$

Lleven a cabo un análisis bayesiano de estos datos y extraigan conclusiones, asumiendo cada componente de la verosimilitud que sea:

- normal
- $t_{(3)}$
- $t_{(1)}$
- Bernoulli (de alguna manera que se les ocurra)

En este ejercicio, escriban un código para manejar cualquier integración necesaria y cálculo de probabilidades marginales posteriores.

Solución.

Como ilustración del procedimiento Bayesiano, consideremos el caso de la distribución normal. Los otros casos son análogos, y es valioso considerar y comparar varios modelos para realizar una selección de modelo. Los pasos a seguir consisten en

- identifcar los parámetros a estimar

- Seleccionar una distribución inicial para los parámetros
- Obtener la distribución posterior
- Obtener características de la distrbución posterio (media, varianza, moda, intervalos de credibilidad, etc.)

En el caso normal, contamos con una muestra aleatoria de tamaño n de una distribución normal $\mathcal{N}(\theta, \sigma^2)$ en donde θ y σ son desconocidos. Entonces $\bar{y}$ y s^2 son estimadores suficientes para (θ, σ) y se distribuyen independientemente con distruciones $\mathcal{N}(\theta, \sigma^2/n)$ y $\sigma^2/(n-1)\chi^2_{n-1}$. La función de verosimilitud es:

$$l(\theta, \sigma|y) \propto p(\bar{y}|\theta, \sigma^2)p(s^2|\sigma^2) \propto \sigma^{-n} e^{-\frac{(n-1)s^2+n(\theta-\bar{y})^2}{2\sigma^2}}$$

Entonces, dados los datos y, la distribución posterior de (θ, σ) es

$$\pi(\theta, \sigma|y) \propto \pi(\theta, \sigma)p(\bar{y}|\theta, \sigma^2)p(s^2|\sigma^2).$$

Podemos asumir que apriori θ y σ son independientes, y considerar distribuciones iniciales de referencia no informativas para cada parámetro, $\pi(\theta, \sigma) = \pi(\theta)\pi(\sigma) \propto c/\sigma$. En este caso, la distribución posterior es

$$\pi(\theta, \sigma|y) \propto \sigma^{-(n+1)} e^{-\frac{(n-1)s^2+n(\theta-\bar{y})^2}{2\sigma^2}}$$

Con los datos que tenemos, evaluamos los parámetros necesarios para estimar valores a partir de la distribución posterior:

```
x <- c(1.2, 2.4, 1.3, 1.3, 0, 1, 1.8, 0.8, 4.6, 1.4)
n <- 10
xbar <- mean(x)
s2 <- var(x)
xbar                    #moda para la media
```

```
[1] 1.58
```

```
sqrt(s2/(n+1))  #moda para sigma
```

```
[1] 0.3708576
```

A partir de esta moda para los estimadores y de la distribución, podemos calcular curvas de nivel para la distribución conjunta de (θ, σ).

□

3. Spiegelhalter et al. (1995) analiza la mortalidad del escarabajo del trigo en la siguiente tabla, usando `BUGS`.

Dosis	# muertos	# expuestos
w_i	y_i	n_i
1.6907	6	59
1.7242	13	60
1.7552	18	62
1.7842	28	56
1.8113	52	63
1.8369	53	59
1.8610	61	62
1.8839	60	60

Estos autores usaron una parametrización usual en dos parámetros de la forma $p_i \equiv P(\text{muerte}|w_i)$, pero comparan tres funciones ligas diferentes:

$$\begin{aligned} \text{logit: } p_i &= \frac{\exp(\alpha+\beta z_i)}{1+\exp(\alpha+\beta z_i)} \\ \text{probit: } p_i &= \Phi(\alpha+\beta z_i) \\ \text{complementario log-log: } p_i &= 1-\exp[-\exp(\alpha+\beta z_i)] \end{aligned}$$

en donde se usa la covariada centrada $z_i = w_i - \bar{w}$ para reducir la correlación entre la ordenada α y la pendiente β. En `OpenBUGS` el código para implementar este modelo es el que sigue:

```
model{
  for (i in 1:k){
    y[i] ~ dbin(p[i],n[i])
        logit(p[i]) <- alpha + beta*(w[i]-mean(w[])
        #       probit(p[i]) <- alpha + beta*(w[i]-mean(w[])
        #       cloglog(p[i]) <- alpha + beta*(w[i]-mean(w[])
      } #fin del loop i

      alpha ~ dnorm(0.0,1.0e-3)
      dbeta ~ dnorm(0.0,1.0e-3)
} #fin del código
```

Lo que sigue al símbolo # es un comentario, así que esta versión corresponde al modelo logit. También `dbin` denota la distribución binomial y `dnorm` denota la distribución normal, donde el segundo argumento denota la precisión, no la varianza (entonces las iniciales normales para α y β tienen precisión 0.001, que son aproximadamente iniciales planas (no informativas)). Hacer el análisis en `OpenBUGS`.

Solución.

La siguiente función que vimos en clase crea el archivo con el modelo, carga los datos y compila el modelo para su uso. Adicionalmente, ejecuta la corida con un número de observaciones y

```
library(BRugs)


Welcome to BRugs connected to OpenBUGS version 3.2.3


run.model <- function(modelo, con="modelog.txt", muestras, datos = list(), longcadena = 10000,
burnin = 0.10, vinit, nchains = 1, thin = 1) {
writeLines(modelo, con=con)
modelCheck(con)    #Envía el modelo a BUGS, para verificar sintaxis
if(length(datos)>0)         #Si hay datos disponibles,
modelData(bugsData(datos)) #BRugs los pone en un archivo y los envía a BUGS

modelCompile(nchains)         #BRugs compila el modelo

if(missing(vinit)) {
modelGenInits()            #Inicializa la cadena al azar si no hay valores iniciales
} else {
for(chain in 1:nchains) modelInits(bugsInits(vinit))
}

modelUpdate(round(longcadena*burnin,0)) #porcentaje de las simulaciones a descartarse
samplesSet(muestras)
samplesSetThin(thin)
modelUpdate(longcadena)
}

datos <- list( w = c(1.6907, 1.7242, 1.7552, 1.7842, 1.8113, 1.8369, 1.8610, 1.8839),
n = c(59, 60, 62, 56, 63, 59, 62, 60),
y = c(6, 13, 8, 28, 52, 53, 61, 60), k = 8)
```

A continuación escribimos los tres modelos y ejecutamos la función `run.model` en cada uno de ellos.

```
# Modelo LOGIT:
modelol <- "
model
{
for (i in 1:k){
y[i] ~ dbin(p[i],n[i])
logit(p[i]) <- alpha + beta*(w[i]-mean(w[]))
} #fin del loop i
alpha ~ dnorm(0.0,0.001)
beta ~ dnorm(0.0,0.001)
}
"

# Resultados Modelo LOGIT
run.model(modelol, con="modelol.txt", datos = datos, muestras = c("alpha","beta","p"), longcadena = 10000)


model is syntactically correct

data loaded

model compiled

initial values generated, model initialized

1000 updates took 0 s

monitor set for variable 'alpha'

monitor set for variable 'beta'

monitor set for variable 'p'

10000 updates took 0 s


samplesStats("*")


         mean       sd  MC_error val2.5pc   median val97.5pc start sample
alpha  0.6001 0.136700 1.565e-03  0.33630  0.59880   0.87120  1001  10000
beta  36.5900 3.063000 3.963e-02 30.85000 36.49000  42.86000  1001  10000
p[1]   0.0425 0.012590 1.606e-04  0.02187  0.04103   0.07078  1001  10000
```

```
p[2]   0.1285 0.024710 3.213e-04  0.08449  0.12690   0.18030  1001  10000
p[3]   0.3113 0.033460 4.301e-04  0.24630  0.31110   0.37900  1001  10000
p[4]   0.5650 0.032490 3.820e-04  0.50100  0.56540   0.62830  1001  10000
p[5]   0.7768 0.027440 3.086e-04  0.72150  0.77760   0.82750  1001  10000
p[6]   0.8978 0.019430 2.234e-04  0.85690  0.89900   0.93180  1001  10000
p[7]   0.9543 0.011960 1.401e-04  0.92820  0.95540   0.97400  1001  10000
p[8]   0.9793 0.006877 8.135e-05  0.96390  0.98020   0.99000  1001  10000
```

```
# Modelo PROBIT
modelop <- "
model
{
for (i in 1:k){
y[i] ~ dbin(p[i],n[i])
probit(p[i])  <- alpha + beta*(w[i]-mean(w[]))
} #fin del loop i
alpha ~ dnorm(0.0,0.001)
beta ~ dnorm(0.0,0.001)
}
"

# Resultados Modelo PROBIT
run.model(modelop, con="modelolp.txt", datos = datos, muestras = c("alpha","beta","p"), longcadena = 10000)
```

```
model is syntactically correct

data loaded

model compiled

****** Sorry something went wrong in procedure Node.Value in module GraphProbit ******

Error in handleRes(res): Internal "trap" error in OpenBUGS, or non-existent module or procedure called.
```

```
samplesStats("*")
```

```
Variable *: model must be initialized before monitors used


data frame with 0 columns and 0 rows
```

```
# Modelo de Valor Extremo (Complimentary log-log
modeloc <- "
model
{
for (i in 1:k){
y[i] ~ dbin(p[i],n[i])
cloglog(p[i]) <- alpha + beta*(w[i]-mean(w[]))
} #fin del loop i
alpha ~ dnorm(0.0,0.001)
beta ~ dnorm(0.0,0.001)
}
"
# Resultados Modelo CLOGLOG
run.model(modeloc, con="modeloc.txt", datos = datos, muestras = c("alpha","beta","p"), longcadena = 10000)
```

```
model is syntactically correct

data loaded

model compiled

initial values generated, model initialized

1000 updates took 0 s

monitor set for variable 'alpha'

monitor set for variable 'beta'

monitor set for variable 'p'

10000 updates took 0 s
```

```
samplesStats("*")
```

```
          mean       sd  MC_error val2.5pc   median val97.5pc start sample
alpha -0.12930 0.082630 5.355e-03 -0.28240 -0.13410   0.02422  1001  10000
beta  24.15000 1.854000 9.415e-02 20.66000 24.13000  28.06000  1001  10000
p[1]   0.07244 0.015560 9.085e-04  0.04496  0.07085   0.10750  1001  10000
p[2]   0.15380 0.023540 1.444e-03  0.10930  0.15070   0.20050  1001  10000
p[3]   0.29580 0.029560 1.921e-03  0.23730  0.29450   0.35360  1001  10000
p[4]   0.50540 0.030540 2.028e-03  0.44690  0.50440   0.56330  1001  10000
p[5]   0.74120 0.028660 1.655e-03  0.67870  0.74220   0.79080  1001  10000
p[6]   0.91720 0.021210 1.008e-03  0.86730  0.91980   0.95210  1001  10000
p[7]   0.98720 0.007718 3.399e-04  0.96730  0.98930   0.99690  1001  10000
p[8]   0.99920 0.001085 4.767e-05  0.99640  0.99960   1.00000  1001  10000
```

□

4. Consideren las siguientes dos distribuciones condicionales completas, analizadas en el artículo de Casella y George (1992) que les incluí como lectura:

$$f(x|y) \propto ye^{-yx}, \qquad 0 < x < B < \infty$$
$$f(y|x) \propto xe^{-xy}, \qquad 0 < y < B < \infty$$

   - Obtener un estimado de la distribución marginal de X cuando $B = 10$ usando el Gibbs sampler.
   - Ahora supongan que $B = \infty$ así que ls distribuciones condicionales completas son ahora las ordinarias distribuciones exponenciales no truncadas. Mostrar analíticamente que $f_x(t) = 1/t$ es una solución a la ecuación integral en este caso:
   $$f_x(x) = \int \left[\int f_{x|y}(x|y) f_{y|t}(y|t) dy \right] f_x(t) dt$$
   ¿El Gibbs sampler convergerá a esta solución?

5. En una prueba real, 12 lotes de mantequilla de cacahuate tienen residuos de aflatoxin en partes por mil millones de 4.94, 5.06, 4.53, 5.07, 4.99, 5.16, 4.38, 4.43, 4.93, 4.72, 4.92, y 4.96.

   - ¿Cuántas posibles muestras bootstrap hay en estos datos?
   - Usando `R` y la función `sample`, o una tabla de números aleatorios, generar 100 remuestras de los datos de la muestra. Para cada una de estas remuestras, obtener la media. Comparar la media de las medias obtenidas en las remuestras con la media de la muestra original.
   - Encontrar de las 100 remuestras, un intervalo de confianza del 95 % para la media.

Solución.

   - Hay $12^{12} = 8.9161004 \times 10^{12}$ muestras diferentes.
   - El siguiente código hace lo solicitado

```
datos <- c(4.94, 5.06, 4.53, 5.07, 4.99, 5.16, 4.38, 4.43, 4.93, 4.72, 4.92,4.96)
Muestras <-matrix(0,nrow=100,ncol=length(datos))
for(i in 1:100) Muestras[i,] <- sample(datos,size = 12,replace = T)
head(Muestras) #Ejemplo de las muestras obtenidas

     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12]
[1,] 4.38 5.06 4.92 4.96 4.43 4.92 4.92 5.07 4.72  5.06  4.93  4.93
[2,] 4.94 4.92 4.94 4.92 5.07 4.99 4.94 4.93 4.96  4.96  4.96  4.92
[3,] 5.07 4.38 5.07 4.38 4.38 5.07 5.06 4.93 4.99  4.53  4.92  5.06
[4,] 5.06 4.93 4.72 4.99 4.43 5.16 5.07 4.72 4.99  4.93  4.94  4.93
[5,] 5.16 5.16 4.43 4.38 4.53 4.38 4.93 4.99 4.99  4.99  4.53  4.43
[6,] 4.96 5.06 4.99 4.72 4.53 4.96 4.92 4.53 4.99  4.93  4.43  4.93

medias <- apply(Muestras,1,mean)
c(mean(medias),mean(datos))

[1] 4.838600 4.840833

mean(medias)-mean(datos) #Comparación de la media de las medias con la media de la muestra

[1] -0.002233333
```

- El intervalo basado en los cuantiles es:

```
quantile(medias,c(.025,.975))

    2.5%    97.5%
4.697354 4.961437
```

□

6. El número de accidentes aéreos de 1983 a 2006 fueron 23, 16, 21, 24, 34, 30, 28, 24, 26, 18, 23, 23, 36, 37, 49, 50, 51, 56, 46, 41, 54, 30, 40, 31.
   - Para la muestra de datos, calcular la media y su error estándar (a partir de la desviación estándar), así como la mediana.
   - Usando R, calcular estimados bootstraps de la media y la mediana con estimados de sus errores estándar, usando $B = 1000$ remuestras. También calcular la mediana de las medianas muestrales.
   - ¿Cómo se comparan los dos incisos anteriores?

Solución.

- El siguiente script realiza los cálculos requeridos

```
datos <- c(23, 16, 21, 24, 34, 30, 28, 24, 26, 18, 23, 23, 36, 37, 49,
50, 51, 56, 46, 41, 54, 30, 40, 31)
mean(datos) #media de los datos.

[1] 33.79167

sd(datos) #desviación estándar

[1] 12.06497

se <- sd(datos)/sqrt(length(datos)) #error estándar
se

[1] 2.462751

quantile(datos,0.5) #mediana

 50%
30.5
```

- A contnuación se calculan las muestras bootstrap para la media y la mediana:

```
B <- 1000
Muestras <-matrix(0,nrow=B,ncol=length(datos))
for(i in 1:B) Muestras[i,] <- sample(datos,size = 12,replace = T)
medias <- apply(Muestras,1,mean)
mediahat <- mean(medias)
mediahat
```

```
[1] 33.75592
```

```
se.mediahat <- sd(medias)
se.mediahat
```

```
[1] 3.311767
```

```
medianas <-apply(Muestras,1,quantile,0.5)
medianahat <- quantile(medianas,0.5)
medianahat
```

```
50%
 31
```

```
se.medianahat <- sd(medianas)
se.medianahat
```

```
[1] 5.126176
```

- Podemos ver de los resultados que la mediana y la media no coinciden, por lo que la distribución de las muestras no es simétrica (y por lo tanto no es normal. También vemos que los estimadores tienen diferente error estándar, siendo la estimación de la mediana más variable que la estimación de la media.

□

7. Supongan que una variable aleatoria y se distribuye de acuerdo a la densidad poli-Cauchy:

$$g(y) = \prod_{i=1}^{n} \frac{1}{\pi(1 + (y - a_i)^2)}$$

donde $a = (a_1, \ldots a_n)$ es un vector de parámetros. Supongan que $n = 6$ y $a = (1, 2, 2, 6, 7, 8)$.

- Escriban una función que calcule la log-densidad de y.
- Escriban una función que tome una muestra de tamaño 10,000 de la densidad de y, usando Metropolis-Hastings con función propuesta una caminata aleatoria con desviación estandar C. Investiguen el efecto de la elección de C en la tasa de aceptación, y la mezcla de la cadena en la densidad.
- Usando la muestra simulada de una "buena" elección de C, aproximar la probabilidad $P(6 < Y < 8)$.

8. Supongan que el vector (X, Y) tiene función de distribución conjunta:

$$f(x, y) = \frac{x^{a+y-1} e^{-(1+b)x} b^a}{y! \Gamma(a)}, x > 0, y = 0, 1, 2, \cdots$$

y deseamos simular de la densidad conjunta.

- Mostrar que la densidad condicional $f(x|y)$ es una Gamma e identificar los parámetros.
- Mostrar que la densidad condicional $f(y|x)$ es Poisson.
- Escriban una función para implementar el Gibbs sampler cuando las constantes son dadas con valores $a = 1$ y $b = 1$.
- Con su función, escriban 1000 ciclos del Gibss sampler y de la salida, hacer los historgramas y estimar $E(Y)$.

9. La τ de Kendall entre X y Y es 0.55. Tanto X como Y son positivas. ¿Cuál es la τ entre X y $1/Y$? ¿Cuál es la τ de $1/X$ y $1/Y$?

   Solución.

   Hay que recordar que la τ de Kendall es una estadística basada en los rangos de la variable aleatoria (expresados en términos de las concordancias y discordancias de las observaciones) y que es invariante ante transformaciones monótonas. Entonces, partiendo de que las variables son positivas, y como $1/Y$ es monótona decreciente, el valor de la τ se mantiene, pero cambia el signo. En el caso en el que se cambian ambas variables, el valor de la estadística queda el mismo. Lo podemos comprobar con una pequeña simulación

```
X <- runif(100)
Y <- X + runif(100)
cor(X,Y,method="kendal")
```

```
[1] 0.4884848
```

```
cor(X,1/Y,method="kendall")
```

```
[1] -0.4884848
```

```
cor(1/X,1/Y,method="kendall")
```

```
[1] 0.4884848
```

□

10. Mostrar que cuando $\theta \to \infty$, $C^{Fr}(u_1, u_2) \to \min\{u_1, u_2\}$, donde C^{Fr} es la cópula de Frank.

    Solución.

Cuando $\theta \to \infty$, $e^{-\theta} - 1 \approx -1$, por lo que

$$\begin{aligned} C^{Fr}(u,v) &\approx -\frac{1}{\theta}\log[1-(e^{-\theta u}-1)((e^{-\theta v}-1)] \\ &= -\frac{1}{\theta}\log[1-(e^{-\theta(u+v)}-e^{-\theta u}-e^{-\theta v}+1)] \\ &= -\frac{1}{\theta}\log[-e^{-\theta(u+v)}+e^{-\theta u}+e^{-\theta v}] \\ &= -\frac{1}{\theta}\log[e^{-\theta u}(-e^{-\theta v}+1+e^{-\theta(v-u)})] \quad \text{si } u<v \\ &= u-\frac{1}{\theta}\log[1-e^{-\theta v}+e^{-\theta(v-u)}] \\ &\to u \quad \text{cuando } \theta \to \infty \end{aligned}$$

Y el límite es simétrico, por lo que $\theta \to \infty$, $C^{Fr}(u,v) \to min\{u,v\}$.

□

11. Consideren la cópula de Clayton. Mostrar que converge a la cópula de comonotonicidad cuando $\theta \to \infty$. [Hint: usen la regla de l'Hópital considerando que la cópula de Clayton se puede escribir como $\exp\{log(u_1^{-\theta}+u_2^{-\theta}-1)/\theta\}$ para θ positivo.]

Solución.

Sea $u < v$ para $u, v \in (0,1)$. Entonces $log(u) > log(v)$ y por lo tanto $\theta(log(v) - \theta log(u) < 0$. Siguiendo el hint, notemos que

$$\begin{aligned} log\left(u^{-\theta}+v^{-\theta}-1\right) &= log\left(e^{-\theta log(u)}(1+e^{\theta(log(v)-log(u))}+e^{\theta log(u)})\right) \\ &= -\theta log(u) + log\left(1+e^{\theta(log(v)-log(u))}+e^{\theta log(u)}\right) \end{aligned}$$

Así que

$$\begin{aligned} \exp\left(\frac{1}{\theta}log\left(u^{-\theta}+v^{-\theta}-1\right)\right) &= \exp(-log(u))\exp\left(\frac{1}{\theta}log\left(1+e^{\theta(logv-logu)}+e^{\theta logu}\right)\right) \\ &= u\exp\left(\frac{-log(1+e^{\theta k}+e^{\theta m})}{\theta}\right) \end{aligned}$$

donde $k = log(u) - log(v) < 0$ y $m = log(u) < 0$ Si aplicamos l'Hôpital a este cociente, tenemos

$$lim_{\theta\to\infty}\frac{-log(1+e^{\theta k}+e^{\theta m})}{\theta} = lim_{\theta\to\infty}\frac{ke^{\theta k}+me^{\theta m}}{1+e^{\theta k}+e^{\theta m}} = 0$$

Así que $u\exp\left(\frac{-log(1+e^{\theta k}+e^{\theta m})}{\theta}\right) \to ue^0 = u$. Como el resultado es simétrico en u y en v, se tiene que

$$lim_{\theta\to\infty}C(u,v)^C = min\{u,v\}$$

□

12. Supongan que tienen dos vectores de datos $(x_1, \ldots, x_n)$ y $(y_1, \ldots, y_n)$. Entonces la cópula empírica es la función $C : [0,1] \times [0,1] \to [0,1]$ definida por

$$C(u,v) = \frac{1}{n} \sum_{j=1}^{n} I\left(\frac{r_j}{n+1} \leq u, \frac{s_j}{n+1} \leq v \right)$$

donde $(r_1, \ldots, r_n)$ y $(s_1, \ldots, s_n)$ denotan los vectores de rangos de x y y respectivamente.

Escriban una función llamada `empCopula` que tome cuatro argumentos `u, v, xVec` y `yVec`. Pueden suponer que los valores `u,v` están en $[0,1]$ y que `xVec` y `yVec` son vectores numéricos que tienen la misma longitud (no vacios).

Solución.

La función que se pide es la siguiente. Es muy simple pero no está *vectorizada*, por lo que no puedo aplicar la función `outer` para poder generar un grid y hacer una gráfica.

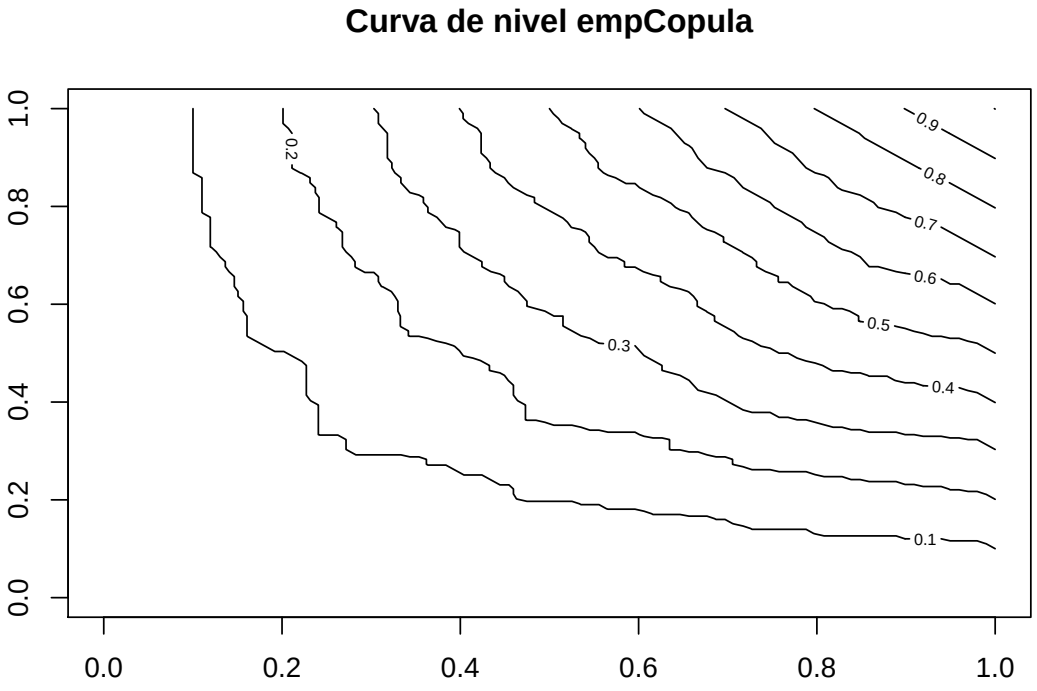
```
empCopula <- function(u,v,xVec,yVec){
#esta función calcula la cópua empírica de un par de vectores aleatorios
#que tienen la misma longitud.
n <- length(xVec)
rx <- rank(xVec)/(n+1)
ry <- rank(yVec)/(n+1)
return(mean((rx<=u) & (ry<=v)))
}
```

Para hacer la gráfica, genero manualmente el grid para la función

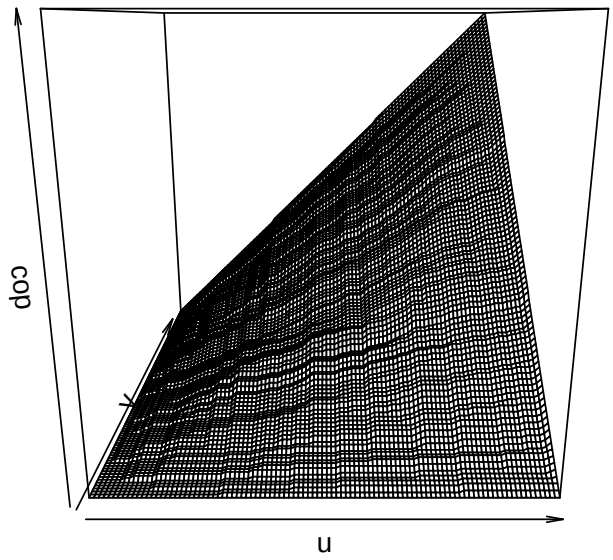
```
k <- 100
u <- v <- seq(0, 1, length = k)
xVec <- runif(200)
yVec <- rnorm(200)
cop <- matrix(numeric(),nrow=k,ncol=k)

#genera un grid para graficar:
for(uu in u)
for(vv in v)
cop[which(uu==u),which(vv==v)] <- empCopula(uu,vv,xVec,yVec)


contour(u, v, cop, main = "Curva de nivel empCopula" )
```




```
persp(u,v,cop, main= "empCopula")
```



□

13. la cópula Farlie-Gumbel-Morgenstern es $C(u,v) = uv[1+\alpha(1-u)(1-v)]$ para $|\alpha| \leq 1$. Mostrar que la densidad conjunta correspondiente $\frac{\partial^2 C(u,v)}{\partial u \partial v}$ es no negativa. Mostrar que C tiene marginales uniformes en $(0,1)$. Encontrar el coeficiente de correlación de Spearman y la tau de Kendall.

Solución.

Este ejercicio tiene tres partes:

- Para obtener la densidad conjunta, derivamos con respecto a cada una de las variables:

$$\begin{aligned} \frac{\partial C}{\partial u} &= v + \alpha v(1-v)[1-2u] \\ \frac{\partial^2 C}{\partial u \partial v} &= 1 + \alpha[1-2u][1-2v] \end{aligned}$$

Como $1-2u \leq 0$ y $1-2v \leq 0$, para $0 \leq u, v \leq 1$ entonces el producto toma el valor mínimo en $u = 1$ y $v = 0$ o $u = 0$ y $v = 1$. En ese caso, para que el producto sea no negativo basta que $\alpha \leq 1$, lo cual siempre se cumple.

- Para ver que C tiene marginales uniformes, basta con evaluar C en los siguientes valores $C(1,v)$ y $C(u,1)$. Como la función es simétrica en los dos valores, basta hacer $C(1,v) = v + \alpha v(0)(1-v) = v$. Por lo tanto, las marginales son uniformes.
- Para esta parte, tenemos que resolver las ecuaciones:

$$\tau = 4\int_0^1\int_0^1 C(u,v)dC(u,v) - 1$$

para la τ de Kendall y

$$\rho_S = 12\int_0^1\int_0^1 C(u,v)dudv - 3$$

para la ρ de Spearman.
Haciendo primero la fórmula de la τ de Kendall, tenemos que:

$$\tau = 4\int_0^1\int_0^1 (uv + \alpha uv(1-u)(1-v)(1+\alpha(1-2u)(1-2v))dudv - 1$$

Haciendo el producto, cada una de las integrales dobles es fácil de resolver todas son polinomiales y simétricas. Por ejemplo:

$$\int_0^1\int_0^1 uvdudv = 1/4$$

$$\int_0^1\int_0^1 \alpha uv(1-2u)(1-2v)dudv = \alpha/36$$

$$\int_0^1\int_0^1 \alpha uv(1-u)(1-v)dudv = \alpha/36$$

$$\int_0^1\int_0^1 \alpha^2 uv(1-u)(1-v)(1-2u)(1-2v)dudv = 0$$

Entonces $\tau = 4(1/4 + \alpha/18) - 1 = 1 + 2\alpha/9 - 1 = 2\alpha/9$. Para el segundo caso,

$$\rho_S = 12\int_0^1\int_0^1 uv(1+\alpha(1-u)(1-v))dudv - 3$$

Expandiendo los términos, nos quedan de nuevo integrales que ya hicimos, por lo que

$$\rho_S = 12(1/4 + \alpha/36) - 3 = 3 + 12\alpha/36 - 3 = \alpha/3$$

□

14. Este es un ejercicio de calibración de las cópulas utilizando correlaciones de rangos. Supongan que una muestra produce un estimado de la τ de Kendall de 0.2. ¿Qué parámetro debe usarse para

    i. la cópula normal,

    ii. la cópula de Gumbel,

    iii. la cópula de Clayton?

    Solución.

    i. La correlación en la cópula normal – y de hecho en cualquier elíptica – debe ser establecida igual a $\rho = \sin(0.1\pi) = 0.309$.

    ii. En la cópula de Gumbel se establece $\delta = (1 - 0.2)^{-1} = 1.25$

    iii. En la cópula de Clayton establecemos $\alpha = 0.4(1 - 0.2)^{-1} = 0.5$.

    □

15. Usen la función `normalCopula` del paquete `copula` para crear una cópula gaussiana bidimensional con un parámetro de 0.9. Luego creen otra cópula gaussiana con parámetro de 0.2 y describan la estructura de ambas cópulas (diferencias y semejanzas).

    Solución.

    Las siguientes gráficas ayudan a entender las diferencias entre las diferentes cópulas. Claramente la cópula con el coeficiente de correlación más alto es la que relaciona linealmente mejor a las dos variables. La dependencia lineal en la cópula gaussiana se relaciona directamente con el parámetro.

```
library(copula)
```

```
Attaching package: 'copula'
```

```
The following object is masked _by_ '.GlobalEnv':
```

```
    empCopula
```

```
normal_0.9 <- normalCopula(param = 0.9, dim = 2)
str(normal_0.9) #despliega las características de la cópula
```

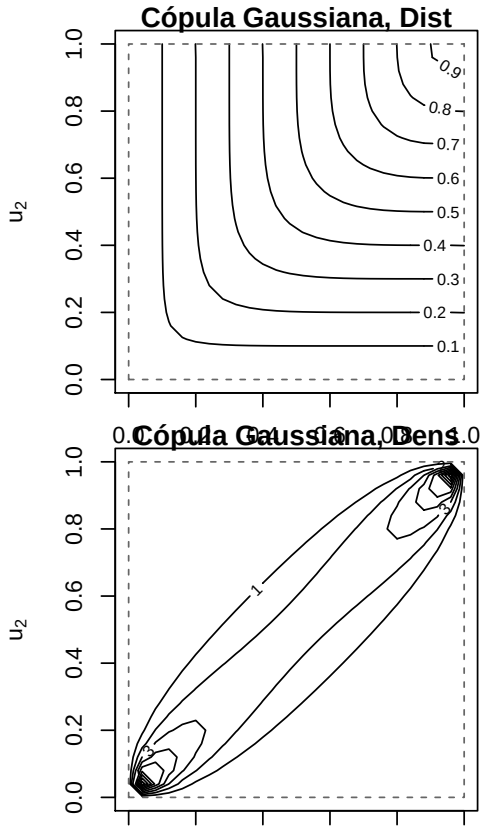
```
Formal class 'normalCopula' [package "copula"] with 8 slots
  ..@ dispstr     : chr "ex"
  ..@ getRho      :function (obj)
  ..@ parameters  : num 0.9
  ..@ param.names : chr "rho.1"
  ..@ param.lowbnd: num -1
  ..@ param.upbnd : num 1
  ..@ fullname    : chr "<deprecated slot>"
  ..@ dimension   : int 2
```

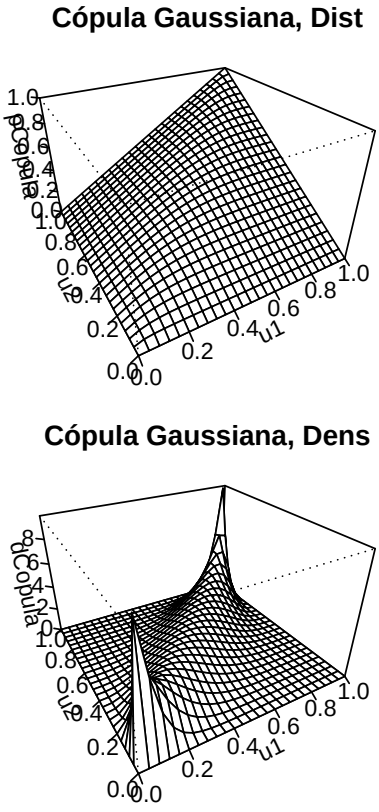
```r
normal_0.2 <- normalCopula(param = 0.2, dim = 2)
str(normal_0.2 ) #despliega las características de la cópula
```

```
Formal class 'normalCopula' [package "copula"] with 8 slots
  ..@ dispstr     : chr "ex"
  ..@ getRho      :function (obj)
  ..@ parameters  : num 0.2
  ..@ param.names : chr "rho.1"
  ..@ param.lowbnd: num -1
  ..@ param.upbnd : num 1
  ..@ fullname    : chr "<deprecated slot>"
  ..@ dimension   : int 2
```

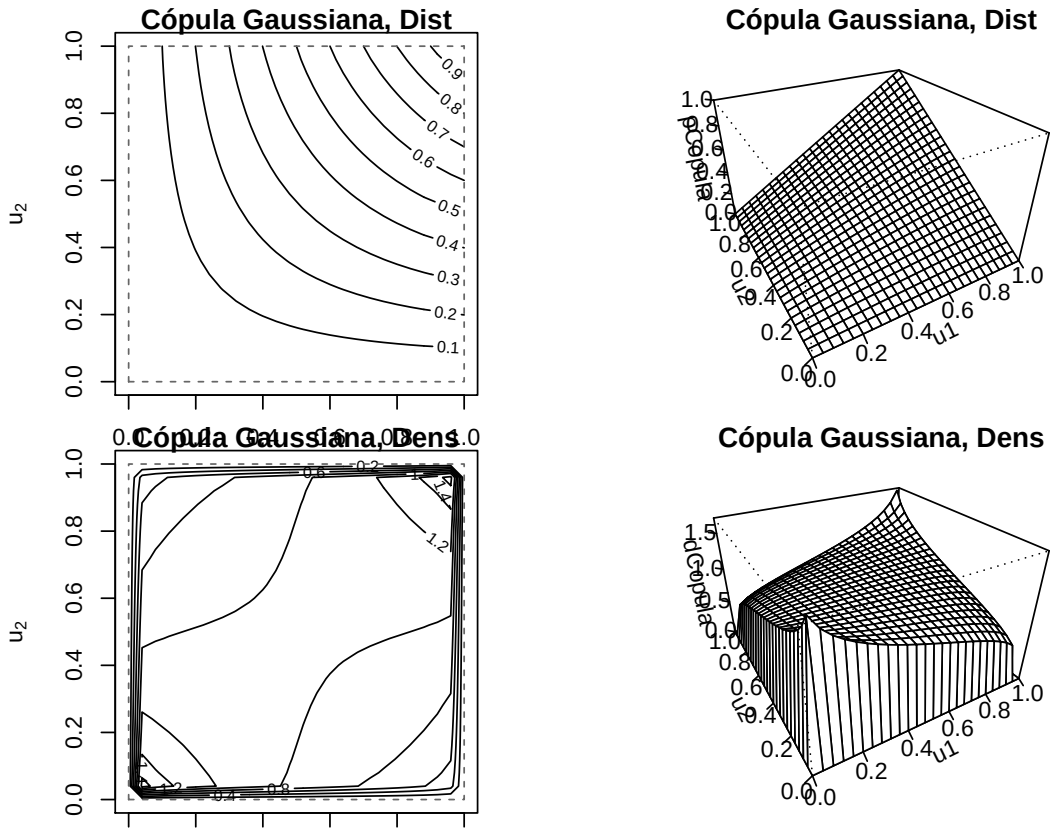
```r
par(mar=c(1,2,1,1))
par(mfrow = c(2,2), pty="s")

contour(normal_0.9, pCopula, main = "Cópula Gaussiana, Dist" )
persp(normal_0.9, pCopula, main = "Cópula Gaussiana, Dist")
contour(normal_0.9, dCopula, main = "Cópula Gaussiana, Dens" )
persp(normal_0.9, dCopula, main = "Cópula Gaussiana, Dens")
```






```r
contour(normal_0.2, pCopula, main = "Cópula Gaussiana, Dist" )
persp(normal_0.2, pCopula, main = "Cópula Gaussiana, Dist")
contour(normal_0.2, dCopula, main = "Cópula Gaussiana, Dens" )
persp(normal_0.2, dCopula, main = "Cópula Gaussiana, Dens")
```

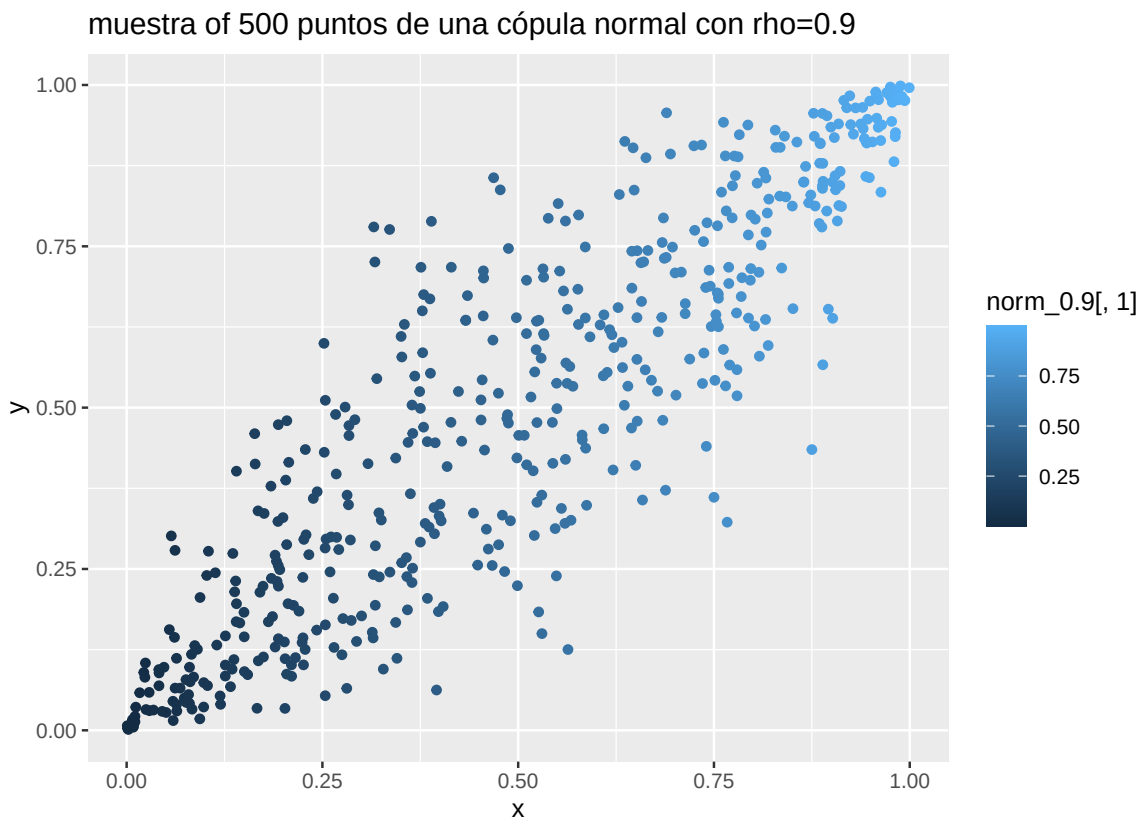


□

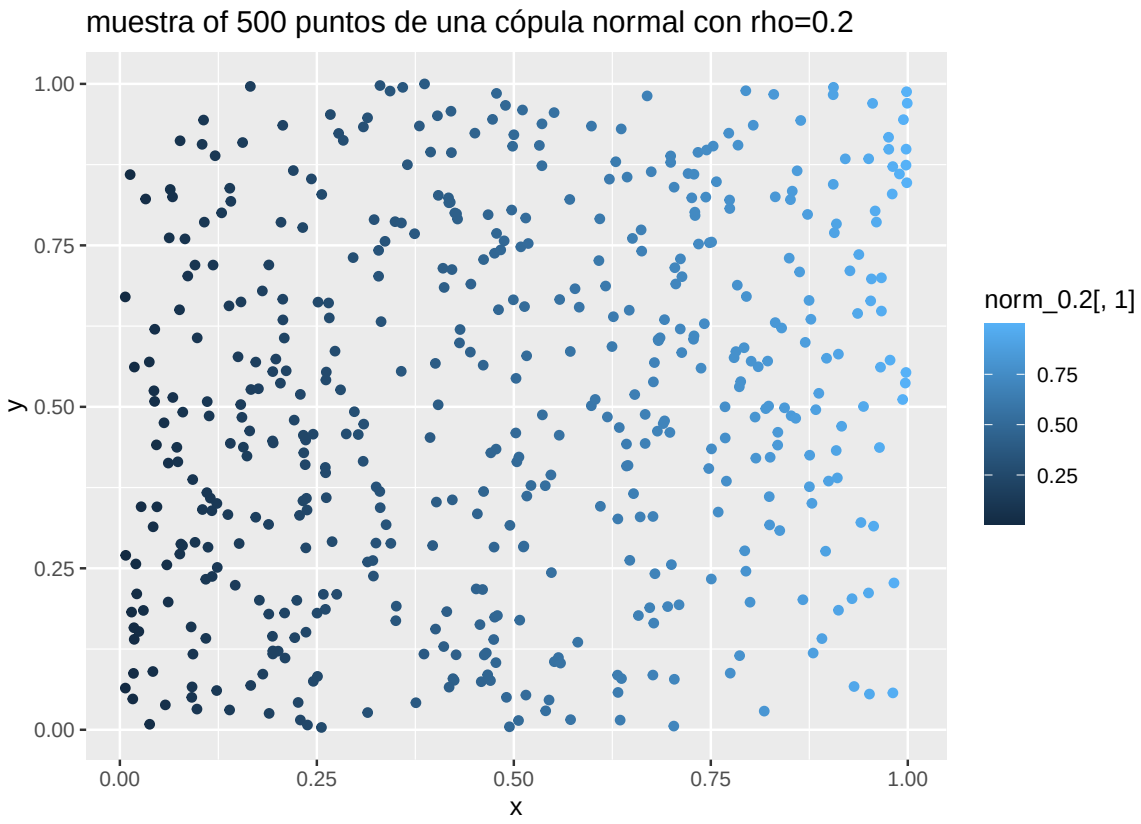
16. Usen la función `rCopula` del paquete `copula` para generar muestras de 500 puntos cuya distribución son las cópulas del ejercicio 8 anterior. Hagan una gráfica de las dos muestras. Teniendo en mente que una cópula determina la estructura de dependencia de una distribución multivariada conjunta, mirando estas gráficas, ¿pueden decir cuál de estas dos cópulas debe usarse para simular una distribución con una fuerte dependencia entre las marginales?

Solución.

```
norm_0.9 <- rCopula(500, normal_0.9) #muestra de la normal con rho=0.9
norm_0.2 <- rCopula(500, normal_0.2) #muestra de la normal con rho=0.2
library(ggplot2)
plot.norm_0.9 <- qplot(norm_0.9[,1], norm_0.9[,2], colour =     norm_0.9[,1],
                                    main="muestra of 500 puntos de una cópula normal con rho=0.9", xlab = "x", ylab = "y")
plot.norm_0.9
```




```
plot.norm_0.2 <- qplot(norm_0.2[,1], norm_0.2[,2], colour = norm_0.2[,1],
                       main="muestra of 500 puntos de una cópula normal con rho=0.2", xlab = "x", ylab = "y")
plot.norm_0.2
```



Claramente la cópula con parámetro 0.9 es la que relaciona densidades marginales con mayor dependencia.

□