

Practica V - Distribucion Normal Multivariante

Xose Manuel Vilan Fragueiro

27 de mayo de 2017

Contents

Introducción	1
Ejercicio [1]	1
(a) Univariante	1
(b) Bivariante	2
(c) Trivariante	10
Ejercicio [2]	13
Conclusión <i>dowjones</i>	14

Introducción

Esta práctica puede consultarse en formato **html** en xvilan.github.io/practicasADM

Se compone de dos ejercicios en los que se realiza un estudio de la distribución normal multivariante para muestras aleatorias y para el conjunto de datos de Dow Jones (paquete QRM) utilizando R.

En probabilidad y estadística, una distribución normal multivariante, también llamada distribución gaussiana multivariante, es una generalización de la distribución normal unidimensional a dimensiones superiores.

Ejercicio [1]

El primer ejercicio consiste en la construcción de representaciones gráficas de datos aleatorios pertenecientes a una distribución normal. El primer paso es definir el número de observaciones de la muestra.

```
n = 250;
```

(a) Univariante

Se crea un vector aleatorio X1, a partir de la distribución normal utilizando la función **rnorm** y se comprueban los primeros valores.

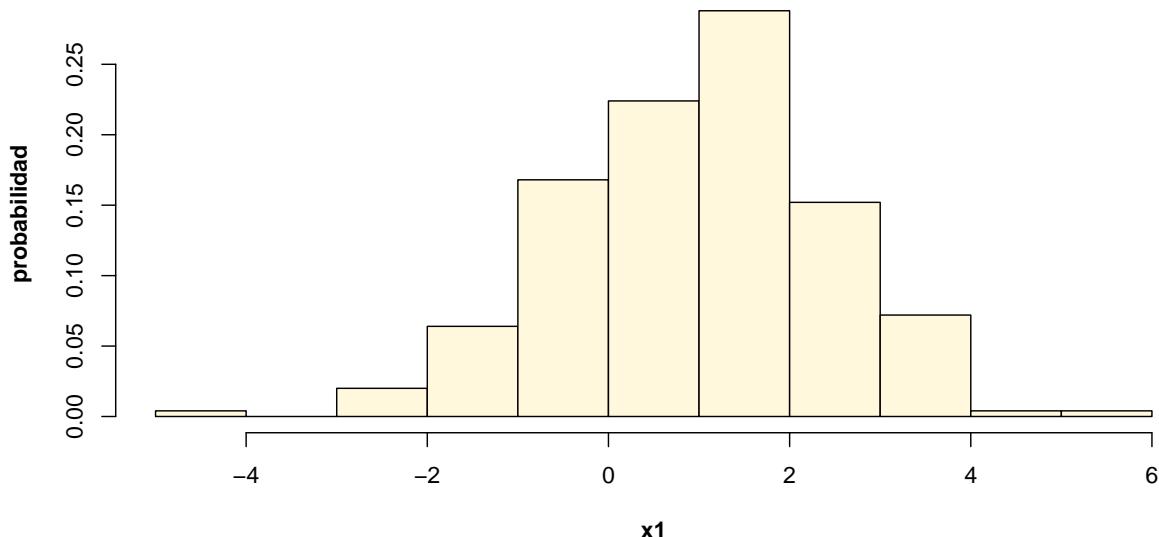
```
X1 = rnorm(n, mean = 1, sd = sqrt(2));  
head(X1);
```

```
[1] -1.1171490 -0.2079257 1.9133777 1.4973979 -0.7196596 2.7770702
```

Se realiza el histograma de frecuencias relativas para comprobar gráficamente su forma.

```
hist(X1, col = 'cornsilk', xlab = 'x1',
main = 'Simulacion N(m = 1, var = 2), n = 250',
freq = FALSE, ylab = 'probabilidad',
font.lab = 2);
```

Simulacion N(m = 1, var = 2), n = 250



Se comprueba, utilizando mediante la función `range`, que los valores están entre la media y tres veces su desviación típica, a ambos lados de la media. Es decir, que el 99,73% de los valores se encuentra en $\mu \pm 3\sigma$. Se comprueba que el centro de la distribución se encuentra en el valor 1 como se había definido en su media.

```
range(X1);
```

```
[1] -4.279741  5.633666
```

Por tanto, los valores x concuerdan con su distribución teórica $X \sim N(\mu, \sigma)$.

(b) Bivariante

Cargamos la librería `mvtnorm` de distribución normal multivariante y comprobamos en la cabecera de su descripción los parámetros que utiliza la función `rmvnorm`,

```
library(mvtnorm);
head(rmvnorm, n = 2);
```

```
1 function (n, mean = rep(0, nrow(sigma)), sigma = diag(length(mean)),
2   method = c("eigen", "svd", "chol"), pre0.9_9994 = FALSE)
```

Se crea el vector de medias y de varianzas. La función `rmvnorm` necesita un vector de medias y otro de matriz de covarianzas. Devuelve una lista con una matriz de 2×2 y una matriz de 250×2 , de la que sólo se muestra su cabecera.

```
n = 250;
(mean = c(1,2));

[1] 1 2

(sigma = matrix(c(2,1,1,4), nrow = 2));

[,1] [,2]
[1,]    2    1
[2,]    1    4

X2 = rmvnorm(n, mean = c(1,2), sigma = matrix(c(2,1,1,4), nrow = 2));
head(X2);

[,1]      [,2]
[1,] -0.53349451  1.5195985
[2,] -0.07017926  2.6760794
[3,]  0.32417973 -0.7215001
[4,]  0.30426479  7.2875621
[5,] -0.54567675  2.1902237
[6,] -0.68663660  1.8167537
```

Se pasa la lista X2 a un formato `data.frame` de R, se asignan nombres a las variables y se comprueba su estructura.

```
X2 = data.frame(X2);

colnames(X2) = paste0('x', 1:2);

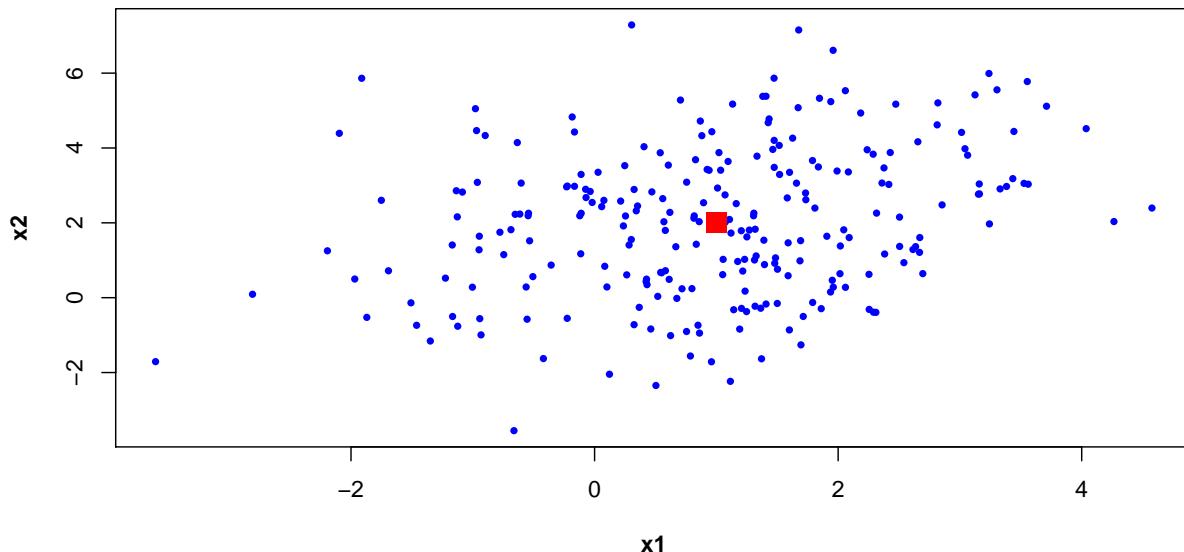
str(X2);

'data.frame': 250 obs. of 2 variables:
 $ x1: num -0.5335 -0.0702 0.3242 0.3043 -0.5457 ...
 $ x2: num 1.52 2.676 -0.722 7.288 2.19 ...
```

Se representa gráficamente la normal bivariada con la función `plot`, añadiendo al gráfico con la función `with` un punto cuadrado de color rojo que representa la media poblacional.

```
library(car)
with(X2, plot(x1,x2, font.lab = 2,
pch = 16, col = 'blue', cex = 0.7,
main = 'Simulación N2[c(1,2), matrix(c(2,1,1,4), nr = 2)], n = 250',
col.main = 'blue'));
points(1,2, pch = 15, cex = 2, col = 'red');
```

Simulación N2[c(1,2), matrix(c(2,1,1,4), nr = 2)], n = 250



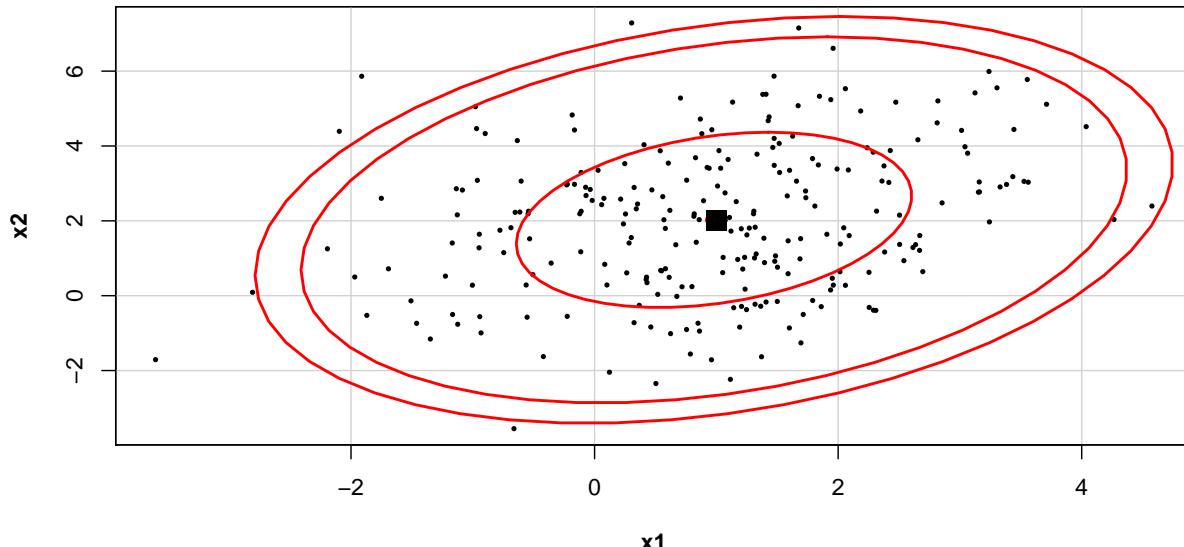
En este primer gráfico no están concentrados todos los puntos entorno a la media, sino que parecen dispersos por la región central. Teóricamente cuanto más grande sea el tamaño muestral mejor replicamos el modelo poblacional y más concentrados entorno a la media estarán los valores.

Seguidamente, se ejecuta el scatterplot y los histogramas de cada variable. Al gráfico de dispersión se añaden elipses que representan percentiles 50, 95 y 9.75 utilizando la función `dataEllipse`. Por definición, cada proyección (distribuciones marginales) deben comportarse como una normal univariante $N(1,2)$ y $N(2,2)$, debido a las propiedades de la distribución condicionada de la normal multivariante. Las marginales son las normales de los momentos correspondientes.

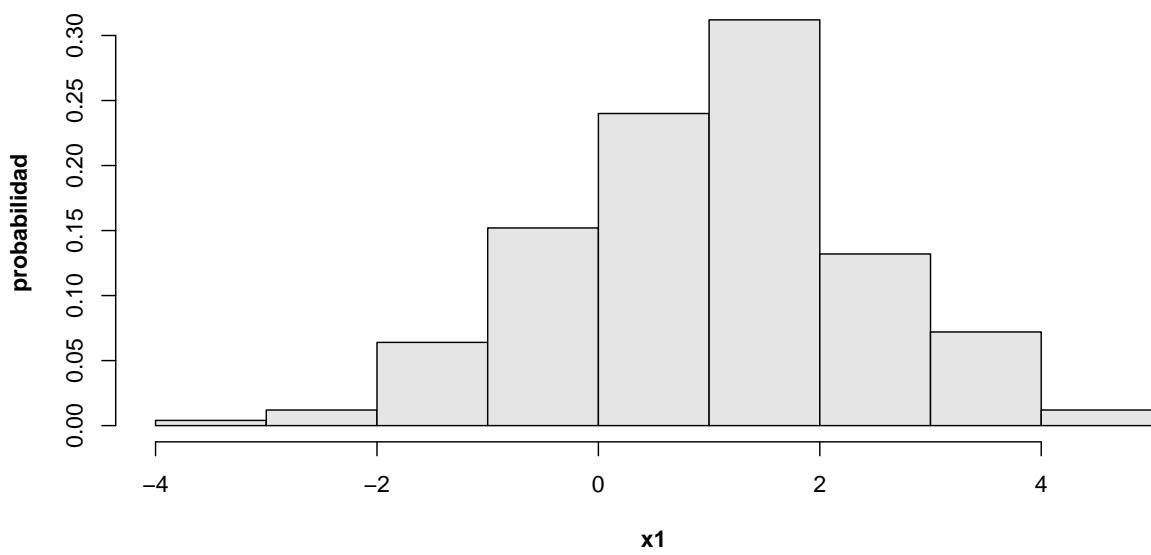
$$X1|X2 = y \sim N_1(y, 1)$$

```
library(knitr)
dataEllipse(as.matrix(X2), levels = c(0.5, 0.95, 0.975), font.lab = 2,
           pch = 16, cex = 0.5, main = 'Simulación N2[c(1,2), matrix(c(2,1,1,4), nr = 2)], n = 250');
points(1,2,pch = 15, cex = 2);
```

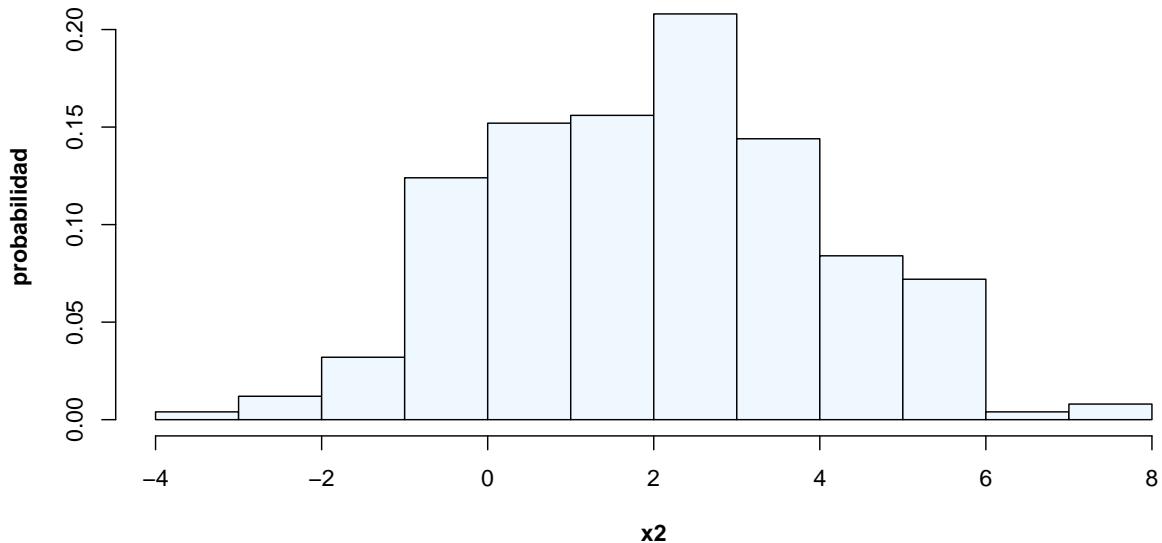
Simulación N2[c(1,2), matrix(c(2,1,1,4), nr = 2)], n = 250



```
with(X2, hist(x1, col = 'grey90', xlab = 'x1',
freq = FALSE, main = '', font.lab = 2,
ylab = 'probabilidad'));
```



```
with(X2, hist(x2, col = 'aliceblue', xlab = 'x2',
freq = FALSE, main = '', font.lab = 2,
ylab = 'probabilidad'));
```



```
kable(cbind(mean, sigma, round(cov2cor(sigma), digits = 4)));
```

mean					
1	2	1	1.0000	0.3536	
2	1	4	0.3536	1.0000	

Los histogramas son las proyecciones del grafico de dispersión sobre los ejes. El primero está centrada en 1 y el segundo está centrada en 2.

Según el gráfico del vector de medias, no está claro, lo que está sucediendo con 250 puntos respecto a la normalidad de los datos.

En la tabla final, se muestran los vectores de media y varianza poblacional, esta última contiene los valores de correlación (0.3536). Se deduce que es una correlación positiva, como también se observa en gráfico de dispersión por la dirección el semieje mayor de la ellipse.

Se vuelven a ejecutar los gráficos de este apartado, pero esta vez cambiando el tamaño de la muestral a 2000 valores procedentes de la normal.

```
library(car)

#tamaño muestral
n = 2000;

#vector de medias y varianza
mean = c(1,2);
sigma = matrix(c(2,1,1,4), nrow = 2);
X2 = data.frame (rmvnorm(n, mean = c(1,2), sigma = matrix(c(2,1,1,4), nrow = 2)));
colnames(X2) = paste0( 'x', 1:2);

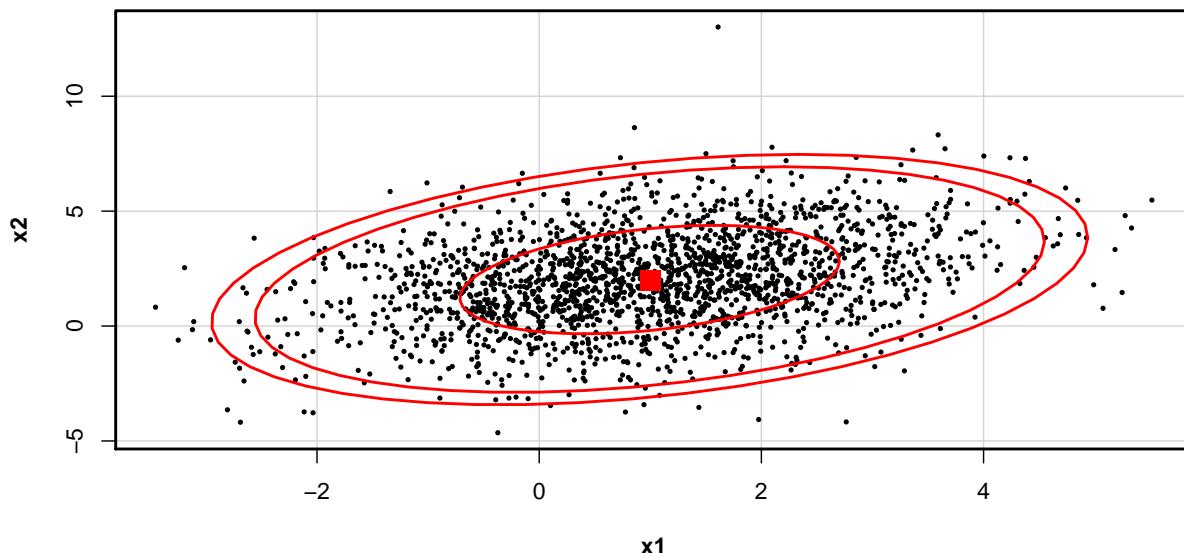
#Gráfico de dispersión con las elipses de percentiles 50, 95 y 9.75
dataEllipse(as.matrix(X2), levels = c(0.5,0.95,0.975), font.lab = 2,
```

```

pch = 16, cex = 0.5, main = 'Simulación N2[c(1,2), matrix(c(2,1,1,4), nr = 2)], n = 2000';
points(1,2,pch = 15, cex = 2, col = 'red');
box(lwd = 2);

```

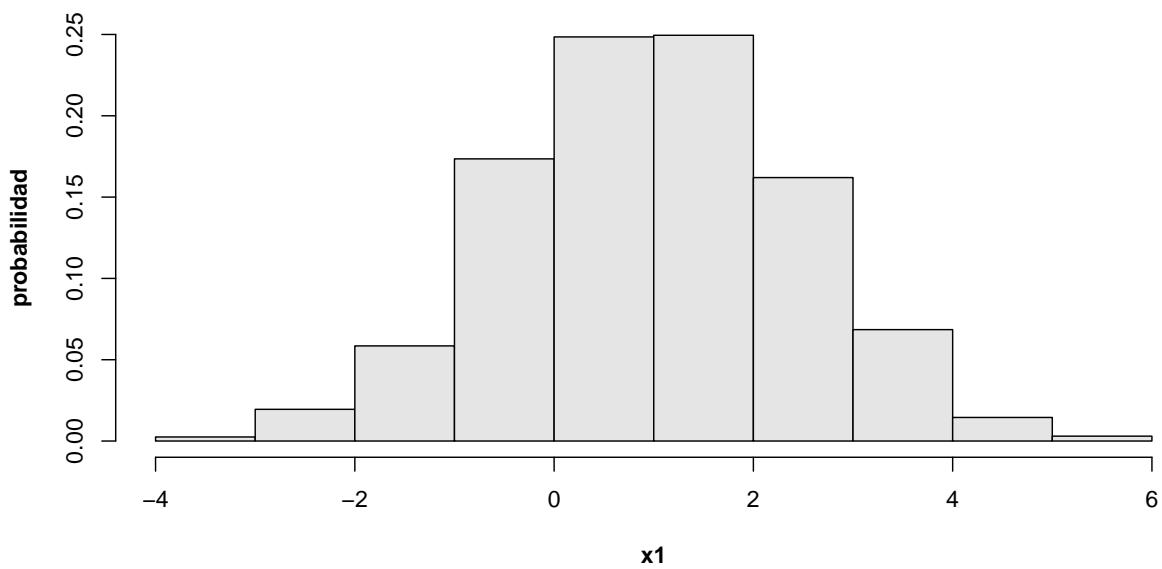
Simulación N2[c(1,2), matrix(c(2,1,1,4), nr = 2)], n = 2000



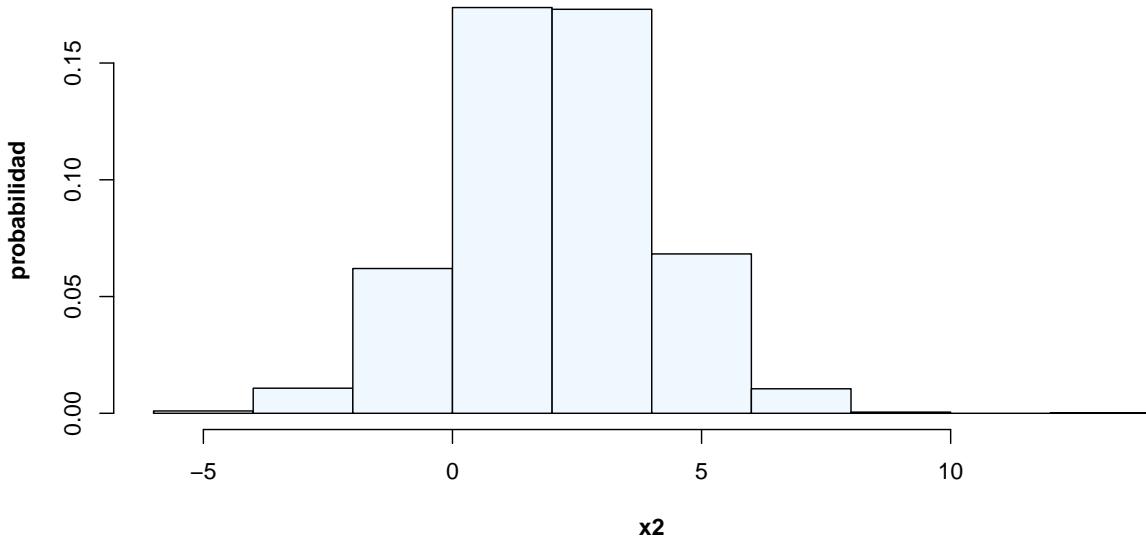
```

#Histograma de la distribución marginal x1
with(X2, hist(x1, col = 'grey90', xlab = 'x1',
freq = FALSE, main = '', font.lab = 2,
ylab = 'probabilidad'));

```



```
#Histograma de la distribución marginal x2
with(X2, hist(x2, col = 'aliceblue', xlab = 'x2',
freq = FALSE, main = '', font.lab = 2,
ylab = 'probabilidad'));
```



```
#Mismos valores de media y varianza poblacional.
```

En la simulación de tamaño muestral ($n = 2000$) se aprecia en el gráfico claramente, cómo los puntos se agrupan alrededor de la media muestral y hay simetría.

Finalmente, otra forma de comprobar las propiedades de la simetría elíptica de la distribución normal bivariante, es representar cuatro gráficos de dispersión con distinto tamaño muestral en un solo grid. En el que se pueda comprobar como a medida que aumenta el tamaño muestral, se confirman las propiedades de la simetría. Al aumentar el tamaño muestral, nos acercamos al modelo poblacional y los puntos están más próximos al vector de media poblacional.

```
mean = c(1,2);
sigma = matrix(c(2,1,1,4), nrow = 2);
kable(cbind(mean, sigma, round(cov2cor(sigma), digits = 4)));
```

mean				
1	2	1	1.0000	0.3536
2	1	4	0.3536	1.0000

```
#tamaño muestral.
n = c(250, 500, 750, 1000);
library(car)
opar = par(no.readonly = TRUE);
par(mfrow = c(2,2), oma = c(2,2,3,2),
mar = c(4.5, 4.5, 1.5, 1.5));
```

```

#Bucle para generar el gráfico de dispersión con cada tamaño muestral
for (i in 1:length(n))
{
X2 = rmvnorm(n[i], mean = c(1,2), sigma = matrix(c(2,1,1,4), nrow = 2))
X2 = data.frame(X2)
colnames(X2) = paste0('x',1:2)

with(X2, plot(x1,x2, font.lab = 2,
pch = 16, col = 'blue', cex = 0.4,
main = paste0('n = ',n[i]),
col.main = 'blue',
cex.main = 0.6));

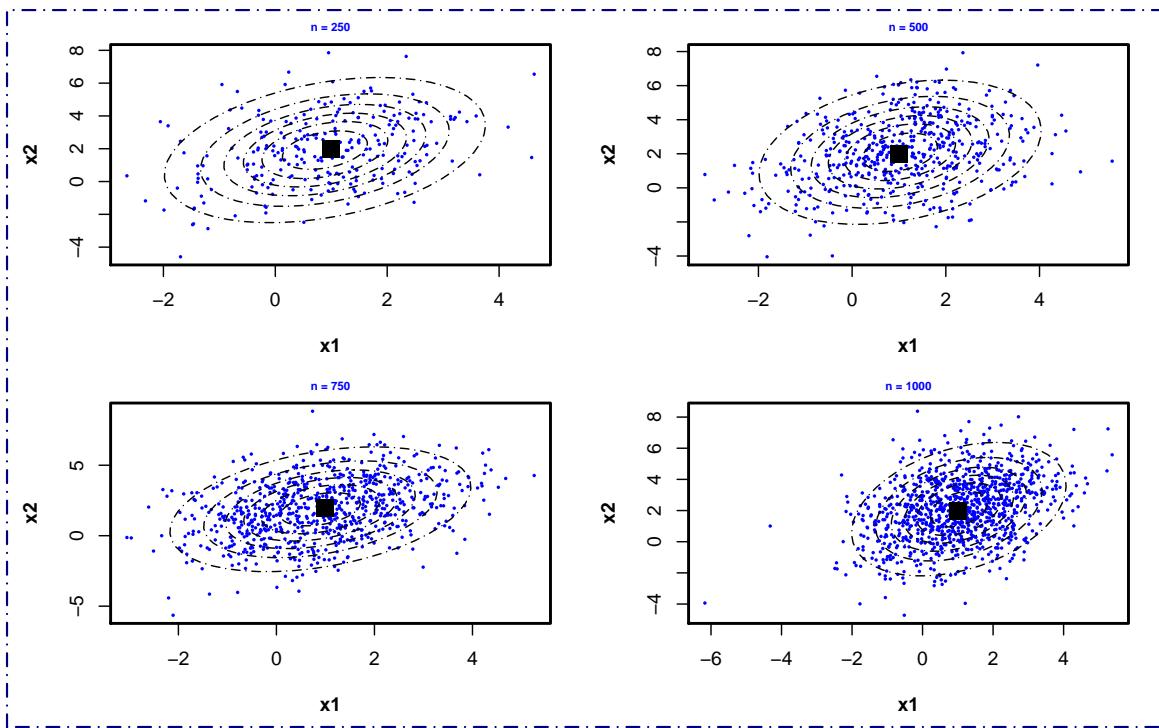
points(1,2,pch = 15, cex = 2);
box(lwd = 2);

dataEllipse(as.matrix(X2),
            plot.points = FALSE,
            levels = 0.15*1:6,
            draw = TRUE,
            add = TRUE,
            xlim = range(X2[,1]),
            ylim = range(X2[,2]),
            grid = FALSE,
            lwd = 1.,
            center.pch = FALSE,
            col = 'black',
            lty = '1373');

if (i == 1)
{
mtext('Simulación N2[c(1,2), matrix(c(2,1,1,4), nr = 2)]',
col = 'brown', font = 2, outer = TRUE, line = .5,cex = 1.2);
box('inner', lty = '1373', col = 'navy', lwd = 1.5);
}
}

```

Simulación N2[c(1,2), matrix(c(2,1,1,4), nr = 2)]



(c) Trivariante

En este caso se utilizan tres variables (Trinormal). Se repite el paso previo de construcción de la lista con vector de medias y varianza.

```
n=250;
mean = c(0,3,-1);
sigma = matrix(c(2,1,1,1,4,0,1,0,5), nrow = 3);
X3 = rmvnorm(n, mean = c(0,3,-1), sigma = matrix(c(2,1,1,1,4,0,1,0,5), nrow = 3));
X3 = data.frame(X3);

colnames(X3) = paste0('x',1:3);

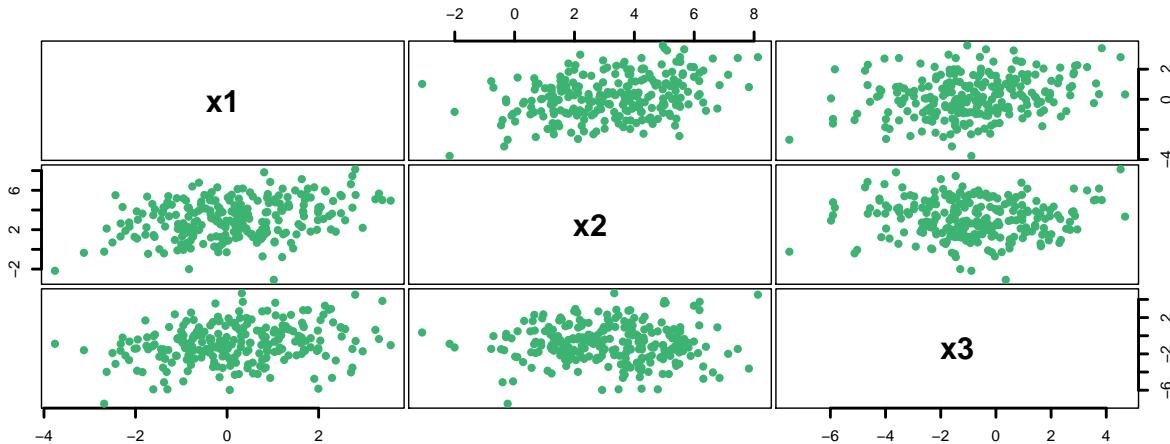
kable(head(X3));
```

	x1	x2	x3
	0.2095413	2.980909	0.9557712
	2.7391208	4.238002	-3.5142218
	3.2403017	5.090770	0.6444555
	1.6618444	3.865300	2.8102257
	-0.1539915	2.597905	-0.4979152
	1.8278252	5.260851	-0.2672691

Se genera un gráfico tipo scatterplot de forma matricial,

```
plot(X3, col = 'mediumseagreen', pch = 16, cex = 1.2,
font.labels = 2, lwd = 2, gap = .25,
main = 'Simulación N2[c(0,3,-1), matrix(c(2,1,1,1,4,0,1,0,5), nr = 3)], n = 250');
```

Simulación N2[c(0,3,-1), matrix(c(2,1,1,1,4,0,1,0,5), nr = 3)], n = 250



```
kable(cbind(mean, sigma, round(cov2cor(sigma), digits = 4)));
```

mean							
0	2	1	1	1.0000	0.3536	0.3162	
3	1	4	0	0.3536	1.0000	0.0000	
-1	1	0	5	0.3162	0.0000	1.0000	

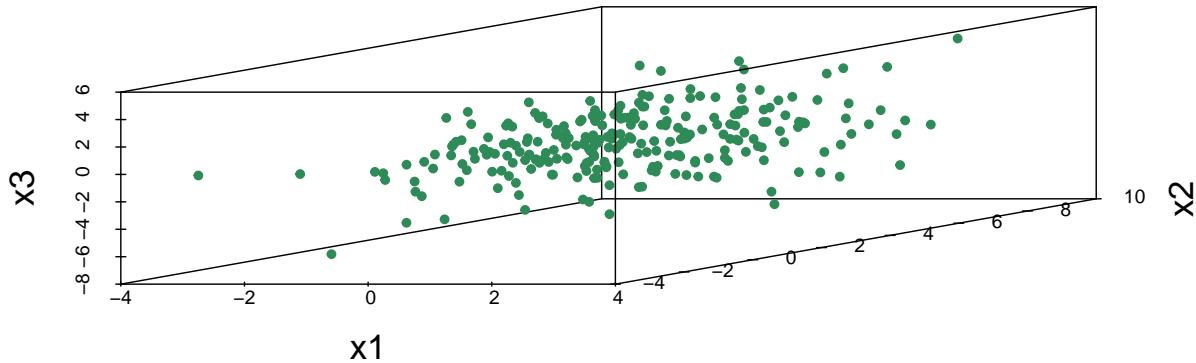
Se observa que X1 y X2 están correladas positivamente ($r = 0.3536$) por la dirección de la elipse que forman gráficamente, mientras que con X3 están incorreladas ($r = 0$). Esto también se puede comprobar en los valores de correlación de la tabla.

Como complemento, se puede generar la misma nube de puntos, pero en lugar de hacerlo de forma bidimensional, en el que se muestren dispersiones de las variables dos a dos, realizado un gráfico tridimensional.

```
library(scatterplot3d);

scatterplot3d(as.matrix(X3), pch = 19, cex = 0.8,
color = 'seagreen', main = '3D Scatterplot del array X3, n=250',
grid = F, cex.axis = 0.8, cex.lab = 1.5);
```

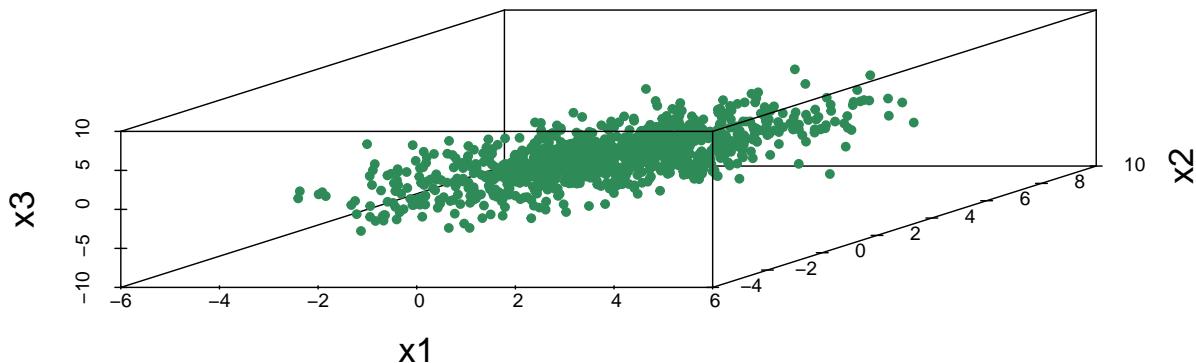
3D Scatterplot del array X3, n=250



Y para un tamaño muestral mayor,

```
#Definición del vector de tamaño 1000.  
n=1000;  
mean = c(0,3,-1);  
sigma = matrix(c(2,1,1,1,4,0,1,0,5), nrow = 3);  
X3 = data.frame(rmvnorm(n, mean = c(0,3,-1),  
sigma = matrix(c(2,1,1,1,4,0,1,0,5),  
nrow = 3))  
);  
colnames(X3) = paste0('x',1:3);  
  
#Gráfico tridimensional.  
scatterplot3d(as.matrix(X3), pch = 19, cex = 0.8,  
color = 'seagreen', main = '3D Scatterplot del array X3, n=1000',  
grid = F, cex.axis = 0.8, cex.lab = 1.5);
```

3D Scatterplot del array X3, n=1000



Ejercicio [2]

En este ejercicio se discutirá la normalidad multivariante de los retornos diarios asociados a una serie de activos correspondientes al conjunto de datos del Dow Jones del paquete QRM. El objetivo es determinar gráficamente si los datos del DJ normales.

En primer lugar, se cargan las librerías necesarias para obtener los datos.

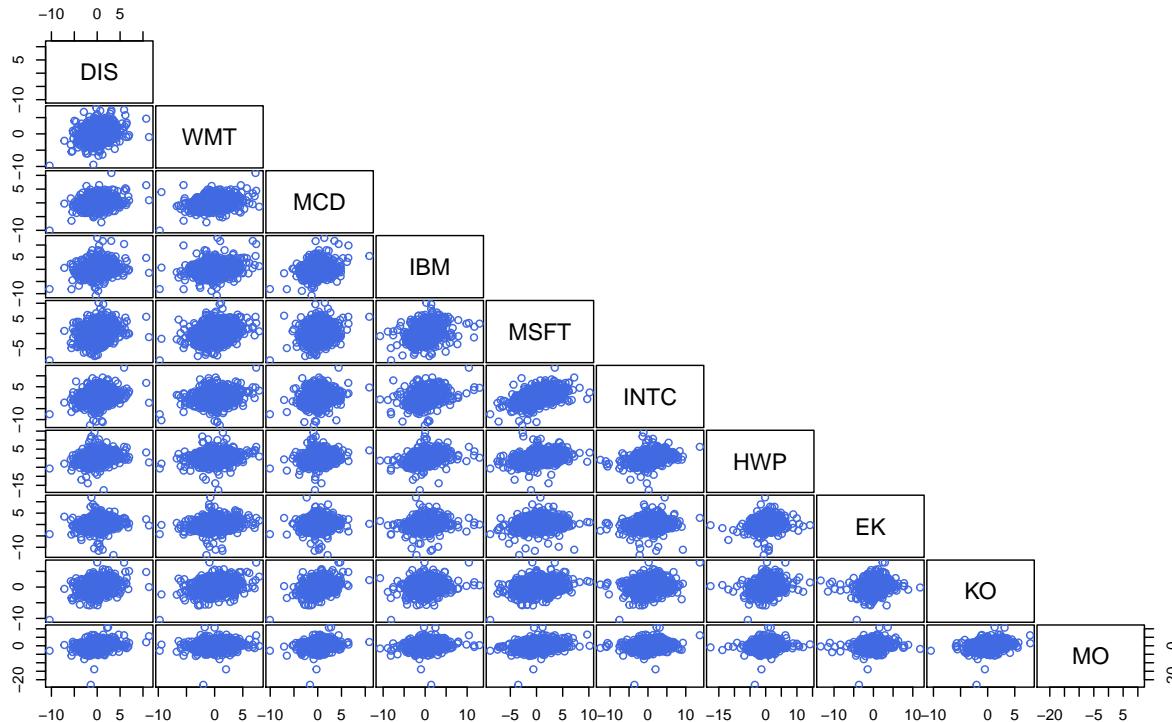
Se genera un vector con las cabeceras de los campos. Se utiliza la función “*window*” del paquete *timeseries* que permite extraer los **valores** para unas fechas dadas por el ejercicio.

```
I = c('MO', 'KO', 'EK', 'HWP', 'INTC', 'MSFT', 'IBM', 'MCD', 'WMT', 'DIS');
data.0 = window(DJ[,I], '1992-01-01', '1998-12-31');
data.0 = returns(data.0, method = 'discrete');
data.1 = 100*data.frame(data.0);
```

Se realiza un diagrama de nube de puntos. Una característica curiosa de este tipo de datos, retornos diarios, es la “asimetría de las colas” que es observable gráficamente.

```
plot(data.1[10:1], upper.panel = NULL,
main = 'Retornos diarios índice Dow Jones 1992-1998',
col = 'royalblue',
labels = colnames(data.1[10:1]),
cex.labels = 1.5,
gap = .15);
```

Retornos diarios índice Dow Jones 1992-1998

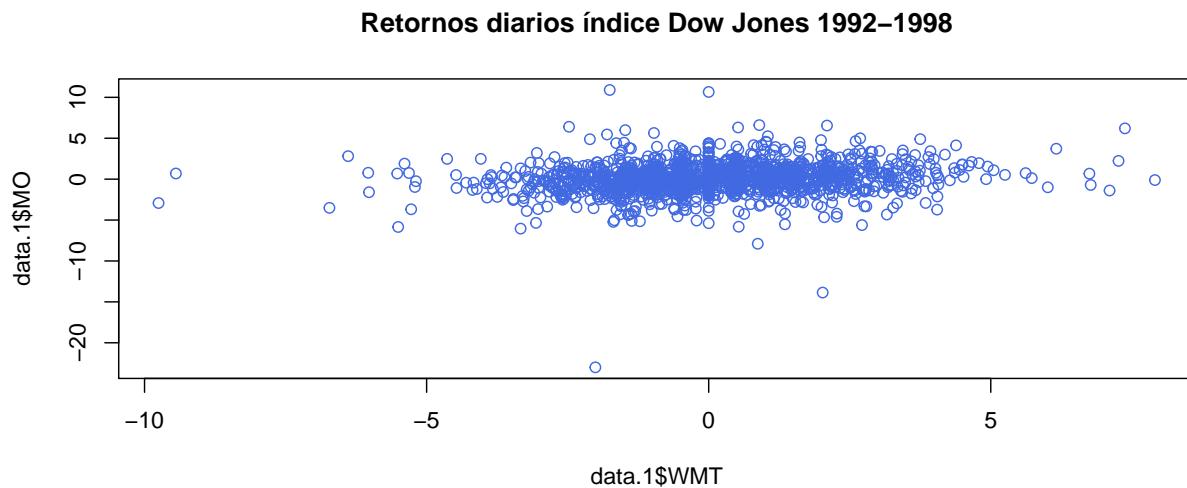


```
kable(round(sqrt(diag(cov(data.1))), digits = 4));
```

MO	1.7738
KO	1.4801
EK	1.7146
HWP	2.2456
INTC	2.3315
MSFT	2.0620
IBM	1.9161
MCD	1.5213
WMT	1.7756
DIS	1.7033

Conclusión *dowjones*

- A pesar de estar compuesto por 1.800 datos no presentan un comportamiento normal puro.
- La parte central de la matriz de gráficos se comporta de forma simétrica y parece que lo es, pero tiene datos muy lejanos al centro. Es decir, las colas de la distribución no lo son y por tanto los retornos no pueden ser normales.
- Se da lo que se denomina problema del goteo, hay puntos que “gotean” de la parte central. Por ejemplo, en el gráfico de dispersión WMT vs MO.



- No son normales por que no se comporta de la misma manera que los datos aleatorios normales del ejercicio anterior. Hay patrones que contradicen lo anterior, como por ejemplo que los puntos están dispuestos hacia afuera.
- Una manera de explicar comportamiento sería definirlo mediante una mixtura. Hay goteo, asimetría, pero una parte central en la que se podría modelar la densidad como una normal (π_i) y la parte del goteo como $(1-\pi_i)$. Podría incluirse una tercera componente para explicar la asimetría. No podemos calcularlo porque los parámetro son desconocidos.