

Práctica 1. Variables conjuntamente Gaussianas. Procesos Aleatorios

PRIMERA PARTE. CARACTERIZACIÓN DE VARIABLES CONJUNTAMENTES GAUSSIANAS.

A) EN EL FICHERO GAUSSDIM8 TIENE 1000 REALIZACIONES DE UNA VARIABLE VECTORIAL GAUSSIANA $X=(X_1, \dots, X_8)$ DE DIMENSIÓN 8.

Sabiendo esto, tendremos que acceder al contenido almacenado en el fichero *GaussDim8*. Para ello, hago uso de una función específica de MATLAB que carga temporalmente los datos del fichero en una constante, **fopen()**, del inglés, *file open*.

```
ficheroGaussDim8 = fopen("GaussDim8");
```

En <https://es.mathworks.com/help/matlab/ref/fopen.html> puedes la referencia de la función, pero para nuestra práctica, solo tendremos que saber que, la única variable que requiere la función para hacer su función es el nombre del archivo que vamos a abrir.

En nuestro caso, hemos asignado los datos que albergaba el fichero *GaussDim8* a la variable *ficheroGaussDim8*, que posteriormente interpretaremos con la función **fread()**, del inglés, *file read*.

```
varVectGauss = fread(ficheroGaussDim8, [8,1000], 'single');
```

Nuevamente, en <https://es.mathworks.com/help/matlab/ref/fread.html> encuentras la referencia de la función, pero aun así vamos a comprender que hace la línea de código que hemos presentado.

Le hemos dado a la función *fread()* tres parámetros:

1. *ficheroGaussDim8*, variable que contiene todos los datos del fichero *GaussDim8* que previamente hemos abierto con la función *fopen()*.
2. *[8,1000]*, hemos indicado a la función que tome los datos que hay en el fichero y los coloque en una matriz de 8 x 1000.
3. *single*, es un parámetro que indica la precisión necesaria para poder interpretar esos datos. La precisión determina el número de decimales que tendremos en las operaciones. Más información sobre la precisión en <https://es.mathworks.com/help/symbolic/increase-precision-of-numeric-calculations.html> y <https://es.mathworks.com/help/matlab/ref/single.html>.

En nuestro caso, hemos asignado la matriz interpretada a la variable *varVectGauss*.

B) DETERMINE EL VECTOR DE MEDIAS m Y LA MATRIZ DE COVARIANZA K DE DICHA VARIABLE VECTORIAL GAUSSIANA A PARTIR DE LAS 1000 REALIZACIONES

Este apartado nos pide dos cosas muy sencillas, calcular la media de cada variable aleatoria y la matriz de covarianza.

Cada persona resolverá este apartado a su manera. Yo he tratado de resolverlo de tal modo, que da igual la matriz Gaussiana que usemos, siempre se calcule estos dos parámetros estadísticos.

¿Cómo hacemos esto? Pues tendremos que recurrir a la función **size()** para determinar el tamaño de la matriz *varVectGauss*.

```
matrixSize = size(varVectGauss);
```

La función *size()* nos devuelve una matriz fila con dos elementos, el primero consiste en el número de filas y el segundo en el número de columnas que tiene la matriz que le hayamos dado.

CÁLCULO DE LA MEDIA

Con el tamaño de *varVectGauss*, inicializo una matriz columna de *matrixSize(1,1)* filas, al que he llamado *var_avg*, del inglés, *variable average*. Esta matriz será una de ceros.

```
var_avg = zeros(matrixSize(1,1), 1);
```

Ahora, tenemos que hacer la media de cada variable aleatoria, es decir, de cada fila de nuestro *varVectGauss*.

Para ello recurrimos a los bucles **for**, y como estamos trabajando con una matriz, necesitaremos dos bucles. Uno para las filas y uno para las columnas.

```
for cov_row_index = 1:matrixSize(1,1)
    for cov_col_index = 1:matrixSize(1,2)
        var_avg(cov_row_index, 1) = var_avg(cov_row_index, 1) +
        (varVectGauss(cov_row_index, cov_col_index) / matrixSize(1,2));
    end
end
```

var_avg % llamo la variable para que en el Live Editor de MATLAB se vea la matriz de medias en pantalla.

```
var_avg = 8×1
-0.0140
-0.0448
-0.0559
-0.0737
-0.0213
-0.0128
0.0290
0.0486
```

El código en sí hace una media sencilla, la que siempre hemos hecho, *sumar todos los valores y dividir entre el número de valores que he sumado*. Es lo que estamos haciendo dentro de los bucles for. Estoy sumando cada dato y lo divido entre el número de datos que he sumado, valor que ya conozco gracias a que sabemos que nuestra matriz *varVectGauss* tiene 8 filas y 1000 columnas, lo que significa que la media será la sumatoria de los valores que tiene cada fila en esas 1000 columnas, entre 1000. O mejor dicho, la sumatoria de cada dato entre mil.

CÁLCULO DE LA MATRIZ DE COVARIANZA

Nuevamente, he inicializado una matriz de ceros, esta vez, cuadrada, pues sabemos que la matriz de covarianza es cuadrada.

```
cov_matrix = zeros(matrixSize(1,1), matrixSize(1,1));
```

Usando bucles **for**, implemento la fórmula de la covarianza:

```
for row_index = 1:matrixSize(1,1)
    for cov_col_index = 1:matrixSize(1,1)
        for avg_col_index = 1:matrixSize(1,2)
            cov_matrix(row_index, cov_col_index) = cov_matrix(row_index,
cov_col_index) + (varVectGauss(row_index, avg_col_index) -
var_avg(row_index, 1))*(varVectGauss(cov_col_index, avg_col_index) -
var_avg(cov_col_index, 1)) / matrixSize(1,2);
        end
    end
end

cov_matrix
```

```
cov_matrix = 8x8
    1.0423    0.7061    0.5186    0.3020    0.2314    0.2363    0.1809    0.1107
    0.7061    1.4537    1.0468    0.7075    0.4922    0.4272    0.3569    0.2479
    0.5186    1.0468    1.7699    1.2535    0.8164    0.6247    0.4872    0.3807
    0.3020    0.7075    1.2535    1.9451    1.2868    0.9175    0.6791    0.5356
    0.2314    0.4922    0.8164    1.2868    1.9103    1.3433    1.0047    0.7106
    0.2363    0.4272    0.6247    0.9175    1.3433    1.8733    1.3579    0.9429
    0.1809    0.3569    0.4872    0.6791    1.0047    1.3579    1.9648    1.3810
    0.1107    0.2479    0.3807    0.5356    0.7106    0.9429    1.3810    1.9697
```

Habrás observado que hemos recurrido al uso de tres bucles **for**. Los primeros dos bucles sirven para recorrer la matriz de covarianza *cov_matrix*, el tercero nos ayuda a construir la sumatoria necesaria para calcular el valor de cada elemento (celda) de dicha matriz.

Este tercer bucle itera a través de todas las muestras (o columnas de datos), calculando el producto de las desviaciones para cada muestra y acumulándolo en la celda correspondiente *cov_matrix(row_index, cov_col_index)*.

Hemos resuelto hasta ahora lo que nos ha pedido la práctica, pero el cálculo de la matriz de covarianza es mucho más fácil y eficiente de hacer usando la siguiente expresión, que cumple perfectamente la finalidad:

```
cov_matrix_2 = (varVectGauss-var_avg)*(varVectGauss - var_avg).' /
matrixSize(1,2)
```

```
cov_matrix_2 = 8x8
    1.0423    0.7061    0.5186    0.3020    0.2314    0.2363    0.1809    0.1107
    0.7061    1.4537    1.0468    0.7075    0.4922    0.4272    0.3569    0.2479
    0.5186    1.0468    1.7699    1.2535    0.8164    0.6247    0.4872    0.3807
    0.3020    0.7075    1.2535    1.9451    1.2868    0.9175    0.6791    0.5356
    0.2314    0.4922    0.8164    1.2868    1.9103    1.3433    1.0047    0.7106
    0.2363    0.4272    0.6247    0.9175    1.3433    1.8733    1.3579    0.9429
    0.1809    0.3569    0.4872    0.6791    1.0047    1.3579    1.9648    1.3810
    0.1107    0.2479    0.3807    0.5356    0.7106    0.9429    1.3810    1.9697
```

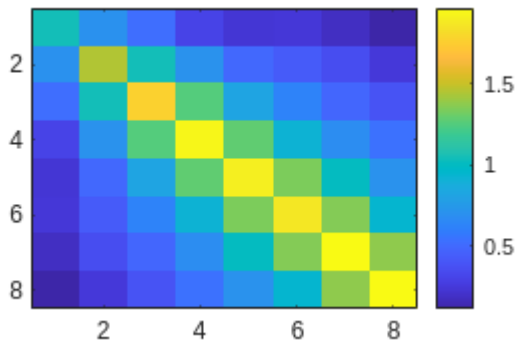
Entendamos el código. Cada fila de *varVectGauss* contiene datos de cada variable aleatoria, así que le restamos a cada fila la matriz de medias *var_avg*, es decir, que a todos los valores de X1 se le resta la media de X1, y así sucesivamente.

El operador * indica que estamos haciendo un producto de matrices y .' hace la transpuesta de la matriz que afecta, entonces, realmente es el producto de las matrices que obtenemos de restar a *varVectGauss* la matriz de medias *var_avg*, de la primera por su transpuesta.

Pero, con esta operativa, habremos calculado la matriz de dispersión. Para obtener nuestra matriz de covarianza, simplemente tenemos que dividir la matriz resultante entre el número de datos que contiene cada variable aleatoria.

Antes de continuar, representemos gráficamente la matriz de covarianza en una escala de colores:

```
figure;  
imagesc(cov_matrix);  
colorbar;
```



C) ¿SON INDEPENDIENTES ENTRE SÍ LAS 8 VARIABLES?

Fijándonos en la matriz de covarianza, observamos que no. Es más, no solo decimos que son dependientes entre sí, sino que están correladas, puesto que no hemos obtenido una matriz diagonal.

#) OBTENGA LA EXPRESIÓN DE LA PDF DE LA VARIABLE X5 Y LA EXPRESIÓN DE LA PDF CONJUNTA DE LAS VARIABLES X3 Y X4.

PDF DE LA VARIABLE X5

Sabemos que estamos trabajando con variables gaussianas, por lo tanto, la expresión de la pdf será de la forma:

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

donde μ y σ^2 denota, respectivamente, la media y la varianza.

Obtengamos los parámetros de la media y la varianza para la variable X5.

```
var_avg(5,1) % nos devuelve la media de la variable X5
```

```
ans =  
-0.0213
```

```
cov_matrix(5,5) % nos devuelve la varianza de X5
```

ans =
1.9103

Materializamos los datos en la expresión y habremos obtenido la PDF de la variable X5:

$$f(x) = \frac{1}{\sqrt{2\pi(1,9108)}} e^{-\frac{(x-(0.0017))^2}{2(1,9108)}}$$

PDF CONJUNTA DE LAS VARIABLES X3 Y X4

Para la PDF conjunta de las variables X3 y X4 necesitamos conocer la matriz de medias y la matriz de covarianza.

La matriz de medias será un vector de tamaño 2x1, conformado por la media de X3 y X4:

```
u = [var_avg(3,1);var_avg(4,1)]
```

```
u = 2x1  
-0.0559  
-0.0737
```

La matriz de covarianzas será un vector de tamaño 2x2, conformado por los siguientes elementos:

```
C = [cov_matrix(3,3) cov_matrix(3,4); cov_matrix(4,3) cov_matrix(4,4)]
```

```
C = 2x2  
1.7699    1.2535  
1.2535    1.9451
```

Calculamos el determinante de nuestra matriz de covarianza y su inversa:

```
det(C)  
inv(C)
```

Y basándonos en la siguiente expresión, completamos los datos.

$$f(x) = \frac{1}{\sqrt{2\pi|C|}} e^{-\frac{(x-\mu)^T C^{-1} (x-\mu)}{2}}$$

Recuerda que $x = \begin{bmatrix} x_3 \\ x_4 \end{bmatrix}$.

D) SI NO SON INDEPENDIENTES ENTRE SÍ, ENCUENTRE UNA TRANSFORMACIÓN LINEAL A DE LA VARIABLE VECTORIAL X QUE DE LUGAR A OTRA VARIABLE ALEATORIA VECTORIAL GAUSSIANA Y = (Y1, ..., Y8) CUYAS VARIABLES SEAN INDEPENDIENTES ENTRE SÍ. PARA ELLO UTILICE LA FUNCIÓN eig DE MATLAB. DICHA FUNCIÓN, AL APLICARLA SOBRE UNA MATRIZ SIMÉTRICA K, NOS DEVUELVE UNA MATRIZ B QUE CONTIENE LOS AUTOVECTORES DE K, Y UNA MATRIZ DIAGONAL

D QUE CONTIENE LOS AUTOVALORES DE K [B,D]=eig(K), VERIFICÁNDOSE QUE B.*K*B=D. DETERMINE EL VECTOR DE MEDIAS Y LA MATRIZ DE COVARIANZA DE LA NUEVA VARIABLE VECTORIAL GAUSSIANA Y.

Hemos visto que nuestra Variable Aleatoria Gaussiana contiene componentes dependientes entre sí (recordemos que la matriz de covarianza no es diagonal).

Usando Álgebra, puedo transformar mi V. A. Gaussiana con componentes no independientes en otra V. A. Gaussiana con componentes independientes, a través de la siguiente transformación lineal:

$$Y = A * X$$

La matriz A es aquella que diagonaliza nuestra matriz de covarianzas "cov_matrix". Para encontrarla, vamos a hacer uso de la función **eig()** que viene en MATLAB.

```
[Eigenvectors,Eigenvalues] = eig(cov_matrix);
```

Esta funcion calcula y nos devuelve los autovectores (Eigenvectors) y los autovalores (Eigenvalues). Cumpliéndose qué:

```
Eigenvectors.'*cov_matrix*Eigenvectors == Eigenvalues
```

```
ans = 8x8 logical array
    0    0    0    0    0    0    0    0
    0    0    0    0    0    0    0    0
    0    0    0    0    0    0    0    0
    0    0    0    0    0    0    0    0
    0    0    0    0    0    0    0    0
    0    0    0    0    0    0    0    0
    0    0    0    0    0    0    0    0
    0    0    0    0    0    0    0    0
```

Sabiendo que $A = \text{Eigenvectors}'$, con los autovectores puedo sacar mi nueva variable vectorial gaussiana Y:

```
varVectGaussY = Eigenvectors'*varVectGauss
```

```
varVectGaussY = 8x1000
    0.1401    0.4916    0.8295   -0.9435    0.3343   -0.3018   -1.6086    0.5763 ...
    0.8761    0.8840    0.5600    0.2050    0.2513   -0.8258    0.0512   -1.2483
    0.8192    0.2691   -0.0693    0.9969    0.0734    1.2818    1.0177    0.4320
   -0.7877   -0.1220   -1.3139   -0.4272   -0.4158    0.3757    0.0651    0.3933
   -1.0531    0.9675    0.2263    1.1503   -0.1110    0.0735   -0.5148    0.6052
   -0.9660   -0.2861    0.2597   -0.3877    1.2277   -1.5992    0.1123    1.3464
   -0.3449    0.0665    2.2610   -0.8161   -1.1209    1.5565    3.2278    1.2071
    1.0198    1.4884    1.1622    0.1071    2.3860    1.6798    5.6426   -2.0359
```

Ahora, calculemos el vector medias y la matriz de covarianza. En esencia, duplicaré y adaptaré el código que ya tenemos. Para la media tengo:

```
var_avg_y = zeros(matrixSize(1,1), 1);

for cov_row_index = 1:matrixSize(1,1)
    for cov_col_index = 1:matrixSize(1,2)
        var_avg_y(cov_row_index, 1) = var_avg_y(cov_row_index, 1) +
            (varVectGaussY(cov_row_index, cov_col_index) / matrixSize(1,2));
```

```
end
end
```

var_avg_y % llamo la variable para que en el Live Editor de MATLAB se vea la matriz de medias en pantalla.

```
var_avg_y = 8x1
    0.0185
    0.0038
   -0.0081
    0.0102
    0.0011
    0.0358
    0.1022
   -0.0483
```

Y para la covarianza:

```
cov_matrix_y = zeros(matrixSize(1,1), matrixSize(1,1));

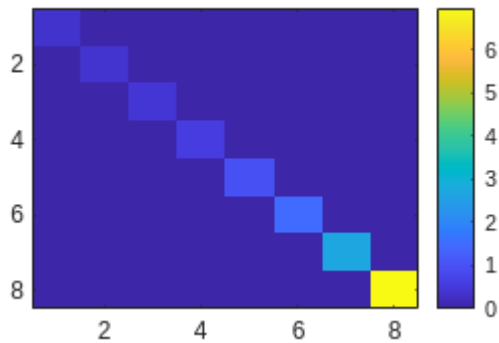
for row_index = 1:matrixSize(1,1)
    for cov_col_index = 1:matrixSize(1,1)
        for avg_col_index = 1:matrixSize(1,2)
            cov_matrix_y(row_index, cov_col_index) = cov_matrix_y(row_index,
cov_col_index) + (varVectGaussY(row_index, avg_col_index) -
var_avg_y(row_index, 1))*(varVectGaussY(cov_col_index, avg_col_index) -
var_avg_y(cov_col_index, 1)) / matrixSize(1,2);
        end
    end
end

cov_matrix_y
```

```
cov_matrix_y = 8x8
    0.3590    -0.0000     0.0000     0.0000     0.0000    -0.0000    -0.0000    -0.0000
   -0.0000     0.3956     0.0000    -0.0000     0.0000     0.0000     0.0000    -0.0000
    0.0000     0.0000     0.4537    -0.0000     0.0000     0.0000     0.0000     0.0000
    0.0000    -0.0000    -0.0000     0.5838    -0.0000    -0.0000    -0.0000    -0.0000
    0.0000     0.0000     0.0000    -0.0000     0.9782     0.0000     0.0000     0.0000
   -0.0000     0.0000     0.0000    -0.0000     0.0000     1.4877    -0.0000    -0.0000
   -0.0000     0.0000     0.0000    -0.0000     0.0000    -0.0000     2.6999    -0.0000
   -0.0000    -0.0000     0.0000    -0.0000     0.0000    -0.0000    -0.0000     6.9713
```

Veamos el resultado gráfico:

```
figure;
imagesc(cov_matrix_y);
colorbar;
```



Hemos conseguido nuestro objetivo. Hemos logrado calcular una Variable Aleatoria Gaussiana con componentes independientes.

Pero... ¿Habrá un proceso más sencillo y directo? ¡Recurrámos al Álgebra!

Habiendo aplicado la transformación lineal a nuestra variable aleatoria gaussiana X para obtener Y , consecuentemente he relacionado de la siguiente forma la media y la matriz de covarianza de Y .

Puedo calcular mi matriz de medias usando la siguiente relación:

$$m_y = A * m_x$$

Siendo m_y la matriz de medias de la V. A. Gaussiana Y y m_x la de X , y la relación $A = \text{Eigenvectors}'$:

```
var_avg_y_2 = Eigenvectors'*var_avg
```

```
var_avg_y_2 = 8x1
0.0185
0.0038
-0.0081
0.0102
0.0011
0.0358
0.1022
-0.0483
```

Y además, puedo calcular mi matriz de covarianza y usando la siguiente relación:

$$C_y = A * C_x * A'$$

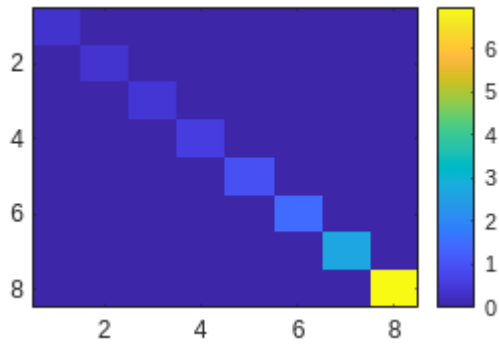
```
cov_matrix_y_2 = Eigenvectors'* cov_matrix *Eigenvectors
```

```
cov_matrix_y_2 = 8x8
0.3590 -0.0000 0.0000 -0.0000 0.0000 0.0000 0.0000 -0.0000
-0.0000 0.3956 0.0000 -0.0000 -0.0000 0.0000 0.0000 0.0000
0.0000 0.0000 0.4537 -0.0000 0.0000 0.0000 -0.0000 0.0000
-0.0000 -0.0000 -0.0000 0.5838 -0.0000 -0.0000 -0.0000 0.0000
0.0000 -0.0000 -0.0000 -0.0000 0.9782 0.0000 0.0000 0.0000
-0.0000 0.0000 0.0000 0.0000 0.0000 1.4877 -0.0000 -0.0000
0.0000 0.0000 -0.0000 -0.0000 0.0000 -0.0000 2.6999 -0.0000
0.0000 0.0000 0.0000 0.0000 0.0000 -0.0000 -0.0000 6.9713
```

Comparando los resultados, observamos que obtenemos los mismos:

```
figure;
```

```
imagesc(cov_matrix_y_2);  
colorbar;
```



Concluimos así la primera sección de la práctica 1.

```
% Redactado por Alexandru Theodor Muntenaş.
```