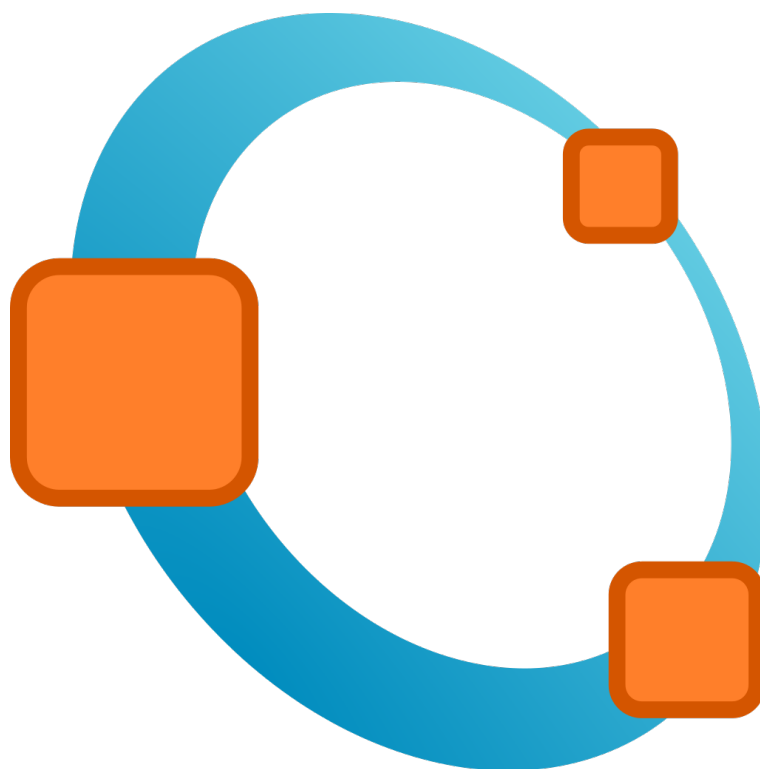


---

# Actividad de Evaluación Continua 3, Octave:Opción A

---

TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN



Autor: Alexander Sebastian Kalis  
Profesores: Dr. Lucas Castro Martínez, Dr. Isaac Seoane Pujol  
Curso: 1o, Ingeniería de Organización Industrial  
UDIMA

## 1. Introducción

En este documento se pretende mostrar un pequeño resumen del desarrollo de la actividad en Octave. El alumno está familiarizado con la estructuración y programación en varios lenguajes funcionales y orientados a objetos, sin embargo nunca ha realizado programación en MatLab/Octave.

## 2. Desarrollo

Se ha escogido la opción A a desarrollar: Crear un algoritmo que permita resolver un sistema de ecuaciones como el que se muestra a continuación y dibujar la progresión de cada una de las 3 variables de salida X1, X2 y X3 para todos los posibles valores del parámetro V. Los valores de los parámetros d son los 8 dígitos de vuestro DNI y el parámetro d es el valor medio de las 8 cifras de dicho DNI.

Los parámetros V y R varían de la siguiente manera: V tomará 51 valores equiespaciados entre -10 y 10. R es 1 valor aleatorio entre 0 y 10.

$$\begin{aligned}d_1X + d_2Y + d_3Z &= d \\d_4X + d_5Y + d_6Z &= V \\d_7X + d_8Y + dZ &= R\end{aligned}$$

Como se ha observado en la sesión de videoconferencia de Octave, se propondrá la resolución del problema utilizando la función de la inversa de una matriz.

En la primera parte se plantean las variables y constantes que utilizará el programa:

```
dv = [3 1 0 8 3 5 2 0];           #El último dígito es 0 ya que el NIE tiene sólo 7 dígitos
d = mean(dv);                     #Media del DNI
V = linspace(-10,10,51);          #Vector con los valores de V
R = rand*10;                       #Numero aleatorio de 0 a 10
A = [dv(1:3); dv(4:6); dv(7:8) d]; #Matriz A
invA = inv(A);                    #Inversa de A
XYZ = zeros(51,3);                #Declaracion de vector vacio XYZ 51x3
```

Seguidamente, mediante un bucle “for”, se crearán varios vectores que corresponderán a los resultados para cada valor de V, es decir, 51 vectores distintos:

```
for i=1:51                         #Bucle for va iterando entre los 51 valores de V
    B = [d; V(i); R];              #Matriz B
    XYZ(i,:) = invA*B;             #Extraccion de los valores
end
```

Se utiliza la función plot() para obtener la gráfica de los resultados obtenidos. Adicionalmente se pone título y leyenda a los valores representados:

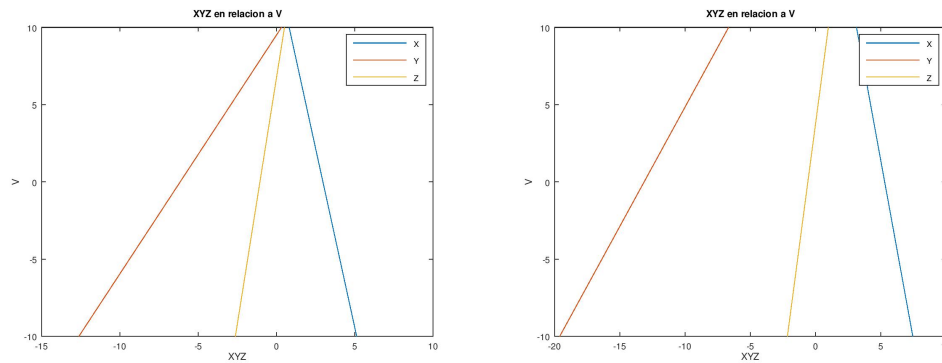
```
##Gráfica de XYZ en relación a V
plot(XYZ,V);
title("XYZ en relación a V");
legend('X', 'Y', 'Z');
xlabel("XYZ");
ylabel("V");
```

Por último se guardan los resultados en un fichero .csv el cual nos permitirá importar estos valores posteriormente, si fueran necesarios:

```
save kalis_sebastian_alexander_octave.csv XYZ V #Se guardan los datos en un fichero CSV
```

### 3. Resultados gráficos

A continuación se muestra un par ejemplos de resultados obtenidos con nuestro script de Octave:



### 4. Conclusión

Viniendo de programación funcional y secuencial principalmente, me ha costado un poco “cambiar el chip” a la forma de pensar en matrices para utilizar el lenguaje de MatLab/GNU Octave. Realmente es un lenguaje muy potente y espero con ansias poder utilizarlo en las asignaturas relacionadas con matemáticas, ya que también es divertido ver las representaciones gráficas de lo que uno computa.