

MATLAB - SESIÓN I

1. Formatear la salida de números.

```
>> format rat;300/450
```

- `format shortE, longE, shortEng, longEng, shortG, longG` y los estilos `compact` y `loose`

2. Comandos útiles para resolver la ecuación matricial $AX = B$.

a) Basado en la descomposición LU . `X=linsolve(A,B')` `[X,r]=linsolve(A,B')`

b) Basado en el algoritmo de Gauss-Jordan. `X=A\B'`

c) Basado en el cálculo de la inversa de A . `X=inv(A)*B`

EJERCICIOS:

Resolver las ecuaciones dadas en las hojas II, III y IV.

3. Matrices mal condicionadas [LAY, pág. 114].

En las aplicaciones, a menudo los datos se obtienen de forma experimental por lo que los errores son inevitables pero, *parece razonable suponer* que si los datos medidos para resolver un problema se desvían poco de los verdaderos datos, las soluciones aproximadas se desviarán poco de las verdaderas soluciones.

a) Comprobar que la suposición anterior es cierta con los dos sistemas “*muy parecidos*” siguientes:

$$\begin{array}{rcl} 3,01x + 6y = 7 & & 3x + 6,01y = 7 \\ 5x - 2y = 2 & & 5x - 2y = 2 \end{array}$$

b) Comprobar que la suposición anterior falla estrepitosamente en los casos siguientes:

$$\begin{array}{rcl} 3,01x + 6y = 7 & & 3x + 6,01y = 7 \\ 2x + 4y = 2 & & 2x + 4y = 2 \end{array}$$

El comportamiento patológico anterior se debe a que las matrices, si bien son regulares, son en realidad “*casi singulares*”. El comando `cond(A)` mide lo “*cerca*” que está una matriz de ser singular.

EJERCICIO:

Construir parejas de sistemas con comportamiento “*patológico*”, esto es, aquellos en los que sus matrices de coeficientes son “*cercanas*” y sus soluciones están “*alejadas*”.

4. Cálculo numérico y cálculo simbólico. Poniendo a Matlab en evidencia: teclear y sacar consecuencias.

```
>> format longG;A=[1 1/2 1/3;1/2 1/3 1/4;1/3 1/4 1/5]
```

```
>> A*inv(A)
```

¡ Resulta que $A \times A^{-1} \neq I$! – Teclear ahora:

```
>> A=sym([1 1/2 1/3;1/2 1/3 1/4;1/3 1/4 1/5])
```

```
>> A*inv(A)
```

5. Matlab y la forma escalonada reducida.

a) El comando `rref`.

b) Si A es regular, ¿qué hace el comando `rref([A eye(size(A))])`?

EJERCICIO:

Resolver el problema 2 a) de la hoja III.

6. Uso de Matlab para hallar la descomposición en matrices elementales de una matriz regular A .

```
>> A=[1 2 2;1 0 2;3 1 -1]
```

```
>> E1=[1 0 0;-1 1 0;-3 0 1];A1=E1*A
```

```
>> E2=[1 0 0;0 -1/2 0;0 0 1];A2=E2*A1
```

```
>> E3=[1 -2 0;0 1 0;0 5 1];A3=E3*A2
```

```
>> E4=[1 0 0;0 1 0;0 0 -1/7];A4=E4*A3
```

```
>> E5=[1 0 -2;0 1 0;0 0 1];A5=E5*A4
```

```
>> inv(E1)*inv(E2)*inv(E3)*inv(sym(E4))*inv(E5)
```

```
# Terminar el apartado b) del problema 2 de la hoja III.
```