===

Práctica 6. Autovalores, autovectores y SVD

Table of Contents

Valores y vectores propios de una matriz

Valores propios - MATLAB & Simulink - MathWorks ES

Ejercicio 1. Descomposición de Schur.

- Calcular la descomposición (real) de Schur de MATLAB para la siguiente matriz A y usarla para calcular sus autovalores sin usar `eig()` o `eigs()`. (Puede ser útil programar una función que calcule autovalores de una matriz 2×2.)

$$A = \begin{pmatrix} 2 & -2 & 4 & 0 \\ -2 & -3 & -2 & -1 \\ -4 & 1 & -3 & 2 \\ 3 & 2 & 4 & 1 \end{pmatrix}$$

```
vaps_schur_mat = 4×4
    0.0000   -7.5339    3.8202    1.6741
    0.1327    0.0000    2.8118   -1.3855
         0         0   -0.0000   -3.0000
         0         0         0   -3.0000

vaps_schur = 4×1 complex
       4.11302936154101e-15 -     0.999999999999991i
       4.11302936154101e-15 +     0.999999999999991i
      -1.0999524923722e-15 +                     0i
                         -3 +                     0i
```

- Usar también la función `eig()` o `eigs()` de MATLAB para calcular los autovalores y autovectores de A.

```
V = 4×4 complex
     0.58977 +          0i      0.58977 +          0i ···
    -0.29488 - 3.8049e-17i     -0.29488 + 3.8049e-17i
    -0.44233 +    0.14744i     -0.44233 -    0.14744i
     0.58977 + 8.3037e-17i      0.58977 - 8.3037e-17i
D = 4×4 complex
    4.113e-15 +         1i            0 +          0i ···
            0 +         0i    4.113e-15 -          1i
            0 +         0i            0 +          0i
            0 +         0i            0 +          0i
```

Ejercicio 2. Método de las potencias para el valor propio de módulo mayor.

Escribir una función que calcule el autovalor dominante y un autovector asociado por el método de las potencias.

`[vap,vep,iteraciones,residuo] = potencias(A, x, kmax, tol)`.

vap: autovalor dominante

vep: autovector dominante

$iteraciones$: número de iteraciones hechas

$residuo$: norma infinito del residuo (es decir, de `A*vep-vap*vep`)

A: matriz

x: vector inicial

$kmax$: máximo número de iteraciones

tol: tolerancia deseada (para la norma infinito del residuo)

- Usar la función para calcular por el método de las potencias, con cuatro cifras significativas, el valor propio dominante de la matriz $A = \begin{pmatrix} 1 & 0 & -1 \\ 1 & 2 & 1 \\ 2 & 2 & 3 \end{pmatrix}$, empezando con $x^{(0)} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$.

```
vap =
     3
vep = 3×1
     -0.49999
      0.49999
            1
iteraciones =
    10
residuo =
    2.395e-05
```

Ejercicio 3. Método de las potencias inversas para el autovalor de módulo menor.

Crear una nueva función `potencias_inv()` que calcule el autovalor de módulo estrictamente más pequeñode una matriz A. Evitar calcular A^{-1}explicitamente: se puede calcular el siguiente vector (antes de normalizar) usando `A\x` y se puede calcular el residuo a partir de A y la inversa de la aproximación al autovalor.

- Usarla para calcular, con cuatro cifras significativas, el autovalor de módulo estrictamente más pequeño de la matriz: $A = \begin{pmatrix} 1 & 2 & -2 & 4 \\ 2 & 12 & 3 & 5 \\ 3 & 13 & 0 & 7 \\ 2 & 11 & 2 & 2 \end{pmatrix}$, empezando con $x^{(0)} = \begin{pmatrix} 1 \\ -1 \\ -1 \\ -1 \end{pmatrix}$.

```
vap =
     0.012234
vep = 4×1
            1
     -0.22566
      0.25013
    -0.009047
iteraciones =
     2
residuo =
   2.9509e-05
```

Ejercicio 4. Cociente de Rayleigh aplicado a matrices simétricas.

Crear nuevas funciones `potenciasRay()` y `potencias_invRay()` a partir de las anteriores `potencias()` y `potencias_inv()` que calculen el autovalor a partir del cociente de Rayleigh. Para ello es más normalizar los vectores respecto a la norma 2 en vez de respecto a la norma ∞.

- Calcular con tolerancia $0.5 \cdot 10^{-4}$ los autovalores de módulo máximo y con tolerancia $0.5 \cdot 10^{-8}$ los autovalores de módulo mínimo, así como los autovectores correspondientes de la siguiente matriz simétrica con ambas funciones:

$$A = \begin{pmatrix} 0 & 12 & 16 & -15 \\ 12 & 388 & 309 & 185 \\ 16 & 309 & 312 & 80 \\ -15 & 185 & 80 & -600 \end{pmatrix}$$

Utilizar los siguientes vectores iniciales $x_{a\max}^{(0)} = (0,\ 1,\ 1,\ 1)^t$, $x_{a\min}^{(0)} = (-1,\ -1,\ 1,\ 1)^t$.

```
vapmax =
       690.34
vepmax = 4×1
     0.033042
            1
      0.85964
      0.19629
iteraciones =
   198
residuo =
   5.4089e-05

vapmaxRay =
       690.34
vepmaxRay = 4×1
     0.024776
      0.74983
      0.64458
```

```
      0.14718
iteracionesRay =
   195
residuoRay =
   5.0685e-05

vapmin =
     -0.72329
vepmin = 4×1
            1
     0.069681
     -0.11518
    -0.018895
iteraciones =
     6
residuo =
   7.8541e-10

vapminRay =
     -0.72329
vepminRay = 4×1
     -0.99089
    -0.069046
      0.11413
     0.018723
iteracionesRay =
     6
residuoRay =
   7.7168e-10
```

Descomposición en valores singulares (SVD). Matriz pseudoinversa.

Documentación de MATLAB, valores singulares.

Ejercicio 5. Matriz pseudoinversa.

Considerar el siguiente sistema lineal sobredeterminado $A\mathbf{x} = \mathbf{b}$. Obtener la factorización SVD de la matriz A y utilizarla para los siguientes apartados.

$$\left\{\begin{array}{rcrcrcrcr} 6x_1 & + & 2x_2 & - & x_3 & - & x_4 & = & 0 \\ 2x_1 & + & 4x_2 & + & x_3 & & & = & 7 \\ -x_1 & + & x_2 & + & 4x_3 & - & x_4 & = & -1 \\ -x_1 & & & - & x_3 & + & 3x_4 & = & 2 \\ 3x_1 & & & + & x_3 & & & = & -2 \end{array}\right. .$$

(a) Obtener los valores singulares de A. Compararlos con los obtenidos por la definición de valores singulares.

```
U = 5×5
  -0.808663318192405   0.108759875979472   0.437792076517924   0.153564835846641 ···
   0.062033929042216  -0.844688552485145   0.441135535365021   0.186229311005836
  -0.360896362715506  -0.490999030252554  -0.544407569902205  -0.526735990409657
   0.255494897640060   0.092746898959185   0.562247247914377  -0.756790431311867
  -0.383003397003534   0.158083850225090   0.035156325493804  -0.302578790392918
S = 5×4
   7.758235971482002                   0                   0                   0
                   0   5.069506229441524                   0                   0
                   0                   0   3.390306425183840                   0
                   0                   0                   0   2.369747569639747
```

```
                  0                   0                   0                   0
V = 4×4
  -0.868940986505888   0.440366749989493   0.219241405886203  -0.054331329993912
  -0.223000085237284  -0.720431798076028   0.618150238287114   0.221673790085089
  -0.364607772314956  -0.519693186478284  -0.538533501451110  -0.554041363176415
   0.249546982193462   0.130284848515394   0.528966120527045  -0.800591659499413

valores_singulares_SVD = 4×1
   7.758235971482002
   5.069506229441524
   3.390306425183840
   2.369747569639747

valores_singulares_def = 4×1
   2.369747569639746
   3.390306425183841
   5.069506229441526
   7.758235971482000
```

(b) Comparar la matriz pseudoinversa obtenidaa partir de la descomposición SVD con la que da la función `pinv()` de MATLAB.

```
pseudoinv = 4×5
   0.124809710760356  -0.056065219133082  -0.025358544988382   0.033150388590658 ···
   0.101975002003045   0.216108484897044  -0.068383943594263   0.011197019469594
  -0.078589455973079  -0.029935501962984   0.276920919798093   0.066110487941671
  -0.006790321288358  -0.013800977485778   0.068784552519830   0.353998077077157

pinvA = 4×5
   0.124809710760356  -0.056065219133082  -0.025358544988382   0.033150388590658 ···
   0.101975002003045   0.216108484897044  -0.068383943594263   0.011197019469594
  -0.078589455973079  -0.029935501962984   0.276920919798093   0.066110487941671
  -0.006790321288358  -0.013800977485778   0.068784552519830   0.353998077077157
```

(c) A partir de la pseudoinversa, obtener la solución de residuo mínimo del sistema $A\mathbf{x} = \mathbf{b}$.

```
x = 4×1
  -0.432477365595706
   1.670238762919638
  -0.488151991026360
   0.343702427690089
```

Funciones internas

`Documento preparado por I. Parada, 20 de marzo de 2024`