

MÉTODOS PARA LA DETERMINACIÓN DE AUTOVALORES

La determinación de los autovalores consiste en la determinación de las raíces de su polinomio característico, si la matriz A es de orden n , entonces el polinomio característico también tiene grado n , para ello es necesario determinar explícitamente el polinomio, lo cual representa un procedimiento demasiado grande respecto al número de operaciones cuando el orden de la matriz es grande.

Una estrategia más simple consiste en determinar el polinomio característico de una matriz semejante que tenga una forma simple como una matriz tridiagonal o de la forma de Hessenberg, también es posible usar el procedimiento iterativo de la descomposición QR usando las matrices de Householder, por otro lado también se puede hallar directamente los autovalores pero en forma aproximada a estas técnicas se les denominan métodos iterativos.

- 1. LA DESCOMPOSICIÓN QR:** Este método busca encontrar una sucesión de matrices $(A_n)_{n \in \mathbb{N}}$ que converja a una matriz triangular en cuya diagonal aparecen los autovalores de la matriz A .

Tomamos inicialmente $A_0 = A$

Si tenemos hallada la matriz A_n se realiza la factorización $A_n = Q_n R_n$ mediante matrices de Householder y $A_{n+1} = R_n Q_n$

Es decir

$$\begin{aligned} A_0 &= A = Q_0 R_0 \\ A_1 &= R_0 Q_0 = Q_0^* A Q_0 = Q_1 R_1 \\ A_2 &= R_1 Q_1 = Q_1^* Q_0^* A Q_0 Q_1 = Q_2 R_2 \\ &\vdots \\ A_{n+1} &= R_n Q_n = Q_n^* \dots Q_1^* Q_0^* A Q_0 Q_1 \dots Q_n = Q_{n+1} R_{n+1} \\ A_{n+1} &= R_n Q_n = (Q_0 Q_1 \dots Q_n)^* A (Q_0 Q_1 \dots Q_n) \end{aligned}$$

Al ser A_n semejante a A todas las matrices de la sucesión tienen los mismos autovalores.

La sucesión $(A_n)_{n \in \mathbb{N}}$ converge a una matriz triangular o aproximadamente triangular.

Si se verifica que, $\lambda_1, \lambda_2, \dots, \lambda_n$ son los autovalores de A , y $|\lambda_1| > |\lambda_2| > \dots > |\lambda_n|$ el algoritmo converge dado que

$$A_k = \begin{bmatrix} \lambda_1 & u \\ 0 & \hat{A}_k \end{bmatrix}$$

Aplicando el algoritmo a \hat{A}_k el proceso continua hacia una matriz triangular.

En el caso de que existan autovalores de igual modulo la sucesión converge a una matriz triangular por bloques, en la que cada bloque contiene autovalores de igual modulo.

TEOREMA: Sea A una matriz real de orden n , existe una ortogonal Q de orden n , tal que

$$Q^t A Q = R = \begin{bmatrix} R_{11} & R_{12} & \cdots & R_{1k} \\ 0 & R_{11} & \cdots & R_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & R_{kk} \end{bmatrix}$$

Donde cada R_{ii} es una matriz escalar de orden 2, cuyos autovalores son complejos conjugados.

LA DESCOMPOSICIÓN QR PARA LA DETERMINACIÓN DE AUTOVALORES

Tomaremos la sucesión de matrices $(A_n)_{n \in \mathbb{N}}$ que converja a una matriz triangular en cuya diagonal aparecen los autovalores de la matriz A .

Tomamos inicialmente $A_0 = A$, hallaremos la descomposición QR de $A_n = Q_n R_n$ mediante matrices de Householder y $A_{n+1} = R_n Q_n$

$$\begin{aligned} A_0 &= A = Q_0 R_0 \\ A_1 &= R_0 Q_0 \\ A_2 &= R_1 Q_1 \\ &\vdots \\ A_{n+1} &= R_n Q_n \end{aligned}$$

EJEMPLO: Hallar los autovalores de la matriz simétrica

$A =$

```
20  14  3  23  19
14  26  7  22  20
3   7  5   7   7
23  22  7  31  24
19  20  7  24  24
```

Usaremos el programa **autoQR**

```
function M_triangular=autoQR(A)
Ai=A;n=length(A);e=1;
while e>0.1
[Q,R,QR]=DesQR(Ai);
A0=Ai;
Ai=R*Q;
e=norm(Ai-A0,1);
end
autovalores=diag(Ai)
M_triangular=Ai;
```

```
>> M_triangular=autoQR(A)
```

Ai =

87.0502	11.7636	2.5336	0.0972	-0.0109
11.7636	12.2687	0.7581	-0.3155	-0.0005
2.5336	0.7581	3.6865	-0.3672	-0.0064
0.0972	-0.3155	-0.3672	2.9863	0.0162
-0.0109	-0.0005	-0.0064	0.0162	0.0083

Ai =

88.9111	1.4336	0.1034	0.0032	0.0000
1.4336	10.5190	0.1476	-0.0912	-0.0000
0.1034	0.1476	3.6519	-0.2861	0.0000
0.0032	-0.0912	-0.2861	2.9097	-0.0000
0.0000	-0.0000	0.0000	-0.0000	0.0082

Ai =

88.9371	0.1694	0.0042	0.0001	-0.0000
0.1694	10.4970	0.0553	-0.0250	0.0000
0.0042	0.0553	3.6887	-0.2235	-0.0000
0.0001	-0.0250	-0.2235	2.8690	0.0000
-0.0000	0.0000	-0.0000	0.0000	0.0082

Ai =

88.9375	0.0200	0.0002	0.0000	0.0000
0.0200	10.4971	0.0204	-0.0068	-0.0000
0.0002	0.0204	3.7122	-0.1723	0.0000
0.0000	-0.0068	-0.1723	2.8450	-0.0000
0.0000	-0.0000	0.0000	-0.0000	0.0082

Ai =

88.9375	0.0024	0.0000	0.0000	0.0000
0.0024	10.4971	0.0074	-0.0018	-0.0000
0.0000	0.0074	3.7263	-0.1314	-0.0000
0.0000	-0.0018	-0.1314	2.8309	0.0000
-0.0000	0.0000	-0.0000	0.0000	0.0082

autovalores =

```
88.9375
10.4971
3.7263
2.8309
0.0082
```

M_triangular =

```
88.9375  0.0024  0.0000  0.0000  0.0000
0.0024  10.4971  0.0074 -0.0018 -0.0000
0.0000  0.0074  3.7263 -0.1314 -0.0000
0.0000 -0.0018 -0.1314  2.8309  0.0000
-0.0000 0.0000 -0.0000  0.0000  0.0082
```

Verificamos el resultado usando la instrucción **eig**

```
>> eig(A)
```

ans =

```
0.0082
2.8120
3.7451
10.4971
88.9375
```

2. MÉTODO DE LA POTENCIA

Este método consiste en determinar el autovalor de mayor valor absoluto de una matriz y su correspondiente autovector.

Dada una matriz A real diagonalizable, y sean $\lambda_1, \lambda_2, \dots, \lambda_n$ los autovalores asociados a los autovectores v_1, v_2, \dots, v_n tal que:

$$|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|$$

Si los autovectores son linealmente independientes, para cualquier sucesión definida por

$$y_{k+1} = Ay_k \Leftrightarrow y_{k+1} = A^k y_0$$

donde y_0 es el vector inicial

entonces

$$\lim_{k \rightarrow \infty} \frac{(y_{k+1})_r}{(y_k)_r} = \lambda_1$$

y si $k \rightarrow \infty$ $y_k \rightarrow v_1$

Prueba:

Siendo $\mathcal{B} = \{v_1, v_2, \dots, v_n\}$ una base para el vector y_0

$$y_0 = \alpha_1 v_1 + \alpha_2 v_2 + \dots + \alpha_n v_n$$

$$\begin{aligned} y_{k+1} &= A^k y_0 = A^k (\alpha_1 v_1 + \alpha_2 v_2 + \dots + \alpha_n v_n) = \alpha_1 A^k v_1 + \alpha_2 A^k v_2 + \dots + \alpha_n A^k v_n \\ &= \lambda_1^k \alpha_1 v_1 + \lambda_2^k \alpha_2 v_2 + \dots + \lambda_n^k \alpha_n v_n = \alpha_1 \lambda_1^k \left[v_1 + \sum_{i=2}^n \left(\frac{\lambda_i}{\lambda_1} \right)^k \frac{\alpha_i}{\alpha_1} v_i \right] \end{aligned}$$

Dado que $|\lambda_1| > |\lambda_i|$ se tiene que

$$\lim_{k \rightarrow \infty} \left(\frac{\lambda_i}{\lambda_1} \right)^k = 0 \quad \forall i = 2, 3, \dots, n$$

Por tanto

$$\lim_{k \rightarrow \infty} \frac{(y_{k+1})_r}{(y_k)_r} = \frac{\alpha_1 \lambda_1^{k+1} \left(v_1 + \sum_{i=2}^n \left(\frac{\lambda_i}{\lambda_1} \right)^{k+1} \frac{\alpha_i}{\alpha_1} v_i \right)_r}{\alpha_1 \lambda_1^k \left(v_1 + \sum_{i=2}^n \left(\frac{\lambda_i}{\lambda_1} \right)^k \frac{\alpha_i}{\alpha_1} v_i \right)_r} = \lambda_1$$

Algoritmo:

$$z_{k+1} = A y_k$$

$$y_{k+1} = \frac{1}{\alpha_{k+1}} z_{k+1} \quad \text{donde} \quad \alpha_{k+1} = \max_{1 \leq r \leq n} \{|(z_{k+1})_r|\} = \|z_{k+1}\|_\infty$$

$$\begin{cases} z_1 = A y_0 \\ y_1 = \frac{1}{\alpha_1} z_1 = \frac{1}{\alpha_1} A y_0 \\ z_2 = A y_1 = \frac{1}{\alpha_1} A^2 y_0 \\ y_2 = \frac{1}{\alpha_2} z_2 = \frac{1}{\alpha_1 \alpha_2} A^2 y_0 \\ z_3 = A y_2 = \frac{1}{\alpha_1 \alpha_2} A^3 y_0 \\ y_3 = \frac{1}{\alpha_3} z_3 = \frac{1}{\alpha_1 \alpha_2 \alpha_3} A^3 y_0 \\ \vdots \\ z_k = A y_{k-1} = \frac{1}{\alpha_1 \alpha_2 \dots \alpha_{k-1}} A^k y_0 \\ y_k = \frac{1}{\alpha_k} z_k = \frac{1}{\alpha_1 \alpha_2 \dots \alpha_k} A^k y_0 \end{cases}$$

Así

$$\lim_{k \rightarrow \infty} \frac{(z_{k+1})_r}{(y_k)_r} = \lim_{k \rightarrow \infty} \frac{\left(\frac{1}{\alpha_1 \alpha_2 \cdots \alpha_k} A^{k+1} y_0\right)_r}{\left(\frac{1}{\alpha_1 \alpha_2 \cdots \alpha_k} A^k y_0\right)_r} = \lim_{k \rightarrow \infty} \frac{(A^{k+1} y_0)_r}{(A^k y_0)_r} = \lambda_1$$

Para obtener una precisión e consideraremos el error relativo

$$\frac{|\lambda_1^{k+1} - \lambda_1^k|}{|\lambda_1^{k+1}|} < e$$

Si $|\lambda_1| > 1$ la sucesión (z_n) converge a la dirección determinada por el autovector, pero sus normas van creciendo, con ello las coordenadas de los vectores de la sucesión se vuelven infinitas, por tanto la sucesión diverge. Por otro lado si $|\lambda_1| < 1$ la sucesión (z_n) converge al vector nulo.

Debido a que solo importa determinar la dirección, podemos dividir o multiplicar en cada etapa al vector z_k por una constante sin cambiar su dirección, a este proceso se le denomina escalado, así los vectores escalados

$$w_k \text{ y } w_k = \frac{z_k}{\|z_k\|_\infty}$$

Por ser muy próximos tendrán la coordenada de mayor modulo en la misma posición y además valdrán 1, así la sucesión no será divergente y tampoco convergerá al vector nulo

EJEMPLO: Usando el método de la potencia determinar el autovalor de mayor valor absoluto de la matriz

$$A = \begin{bmatrix} 3 & 0 & 1 \\ 2 & 2 & 2 \\ 4 & 2 & 5 \end{bmatrix}$$

Con una precisión de 10^{-1}

Tomando $y_0 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$ tenemos

$$\begin{cases} z_1 = Ay_0 = \begin{bmatrix} 4 \\ 6 \\ 11 \end{bmatrix} & \alpha_1 = 11 \\ y_1 = \frac{1}{\alpha_1} z_1 = \begin{bmatrix} 0.3636 \\ 0.5455 \\ 1 \end{bmatrix} \\ z_2 = Ay_1 = \begin{bmatrix} 2.0909 \\ 3.8182 \\ 7.5455 \end{bmatrix} \end{cases}$$

Calculando el error relativo

$$\lambda_1^{(1)} = \frac{(z_2)_r}{(y_1)_r} = \begin{bmatrix} 2.0909/0.3636 \\ 3.8182/0.5455 \\ 7.5455/1 \end{bmatrix} = \begin{bmatrix} 5.7500 \\ 7 \\ 7.5455 \end{bmatrix}$$

siendo

$$\lambda_1^{(2)} = \frac{(z_3)_r}{(y_2)_r}$$

Hallaremos y_2 , z_3

$$\begin{cases} y_2 = \frac{1}{\alpha_2} z_2 = \begin{bmatrix} 0.2771 \\ 0.5060 \\ 1 \end{bmatrix} \\ z_3 = Ay_2 = \begin{bmatrix} 1.8313 \\ 3.5663 \\ 7.1205 \end{bmatrix} \end{cases}$$

Luego

$$\lambda_1^{(2)} = \frac{(z_3)_r}{(y_2)_r} = \begin{bmatrix} 6.6087 \\ 7.0276 \\ 7.1205 \end{bmatrix}$$

así

$$E = \frac{|\lambda_1^{(2)} - \lambda_1^{(1)}|_r}{|\lambda_1^{(2)}|_r} = \begin{bmatrix} 0.1299 \\ 0.0068 \\ 0.0597 \end{bmatrix}$$

$$\|E\|_\infty = 0.1299$$

$$\begin{cases} y_3 = \frac{1}{\alpha_3} z_3 = \begin{bmatrix} 0.2572 \\ 0.5008 \\ 1 \end{bmatrix} \\ z_4 = Ay_3 = \begin{bmatrix} 1.7716 \\ 3.5161 \\ 7.0305 \end{bmatrix} \end{cases}$$

$$\lambda_1^{(3)} = \frac{(z_4)_r}{(y_3)_r} = \begin{bmatrix} 6.8882 \\ 7.0203 \\ 7.0305 \end{bmatrix}$$

$$E = \frac{|\lambda_1^{(3)} - \lambda_1^{(2)}|_r}{|\lambda_1^{(3)}|_r} = \begin{bmatrix} 0.0406 \\ 0.0039 \\ 0.0128 \end{bmatrix}$$

$$\|E\|_\infty = 0.0406$$

Por tanto

$$\lambda_1 = 7.0305 \quad \text{y} \quad v_1 = y_3 = \begin{bmatrix} 0.2572 \\ 0.5008 \\ 1 \end{bmatrix}$$

Usando un código creado en Matlab obtenemos

tipo =

inf

$|a(1)| = 11.0000$ error =
 $z(1) = 4.0000 \ 6.0000 \ 11.0000$
 $y(0) = 1.0000 \ 1.0000 \ 1.0000$

$|a(2)| = 7.5455$ error =
 $z(2) = 2.0909 \ 3.8182 \ 7.5455$
 $y(1) = 0.3636 \ 0.5455 \ 1.0000$

$|a(3)| = 7.1205$ error = 0.1299
 $z(3) = 1.8313 \ 3.5663 \ 7.1205$
 $y(2) = 0.2771 \ 0.5060 \ 1.0000$

$|a(4)| = 7.0305$ error = 0.0406
 $z(4) = 1.7716 \ 3.5161 \ 7.0305$
 $y(3) = 0.2572 \ 0.5008 \ 1.0000$

autoValor =

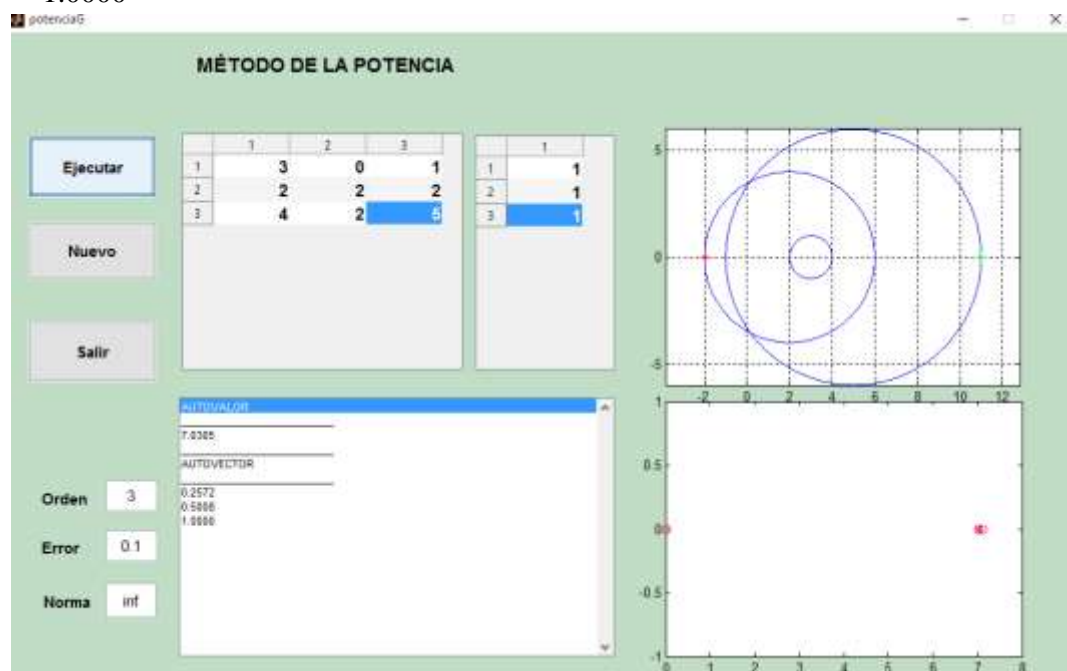
7.0305

autoVetor =

0.2572

0.5008

1.0000



Usando instrucciones del Matlab

```
>> A=[3 0 1;2 2 2;4 2 5]
```

A =

```
3    0    1
2    2    2
4    2    5
```

```
>> [V,D]=eig(A)
```

V =

```
-0.2182 -0.6667 -0.3333
-0.4364  0.3333 -0.6667
-0.8729  0.6667  0.6667
```

D =

```
7.0000    0    0
0  2.0000    0
0    0  1.0000
```

Luego como el vector hallado de forma aproximada es unitario, tenemos denotando por w al autovector asociado

```
>> w=[-0.2182;-0.4364;-0.8729]
```

w =

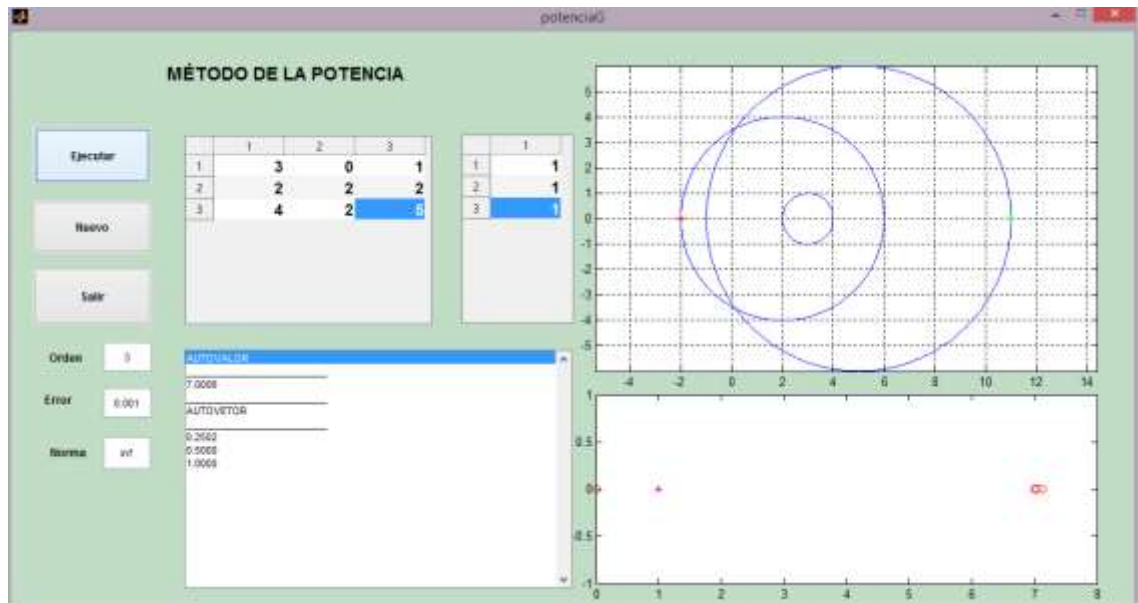
```
-0.2182
-0.4364
-0.8729
```

```
>> w/norm(w,inf)
```

ans =

```
-0.2500
-0.4999
-1.0000
```

Tomando una precisión de 10^{-3}



Obtenemos el autovector

0.2502

0.5000

1.0000

EJEMPLO: Usando el método de la potencia determinar el autovalor de mayor valor absoluto de la matriz

$$A = \begin{bmatrix} 4 & 1 & 1 \\ 0 & 2 & 1 \\ -2 & 0 & 9 \end{bmatrix}$$

Con una precisión de 10^{-1}

Tomando $y_0 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$ tenemos

tipo =

inf

$|a(1)| = 7.0000$ error =

$z(1) = 6.0000 \ 3.0000 \ 7.0000$

$y(0) = 1.0000 \ 1.0000 \ 1.0000$

$|a(2)| = 7.2857$ error =

$z(2) = 4.8571 \ 1.8571 \ 7.2857$

$y(1) = 0.8571 \ 0.4286 \ 1.0000$ $|a(3)| = 7.6667$ error = 0.2684

$z(3) = 3.9216 \ 1.5098 \ 7.6667$

$y(2) = 0.6667 \ 0.2549 \ 1.0000$

$|a(4)| = 7.9770$ error = 0.1632

z(4)= 3.2430 1.3939 7.9770
y(3)= 0.5115 0.1969 1.0000

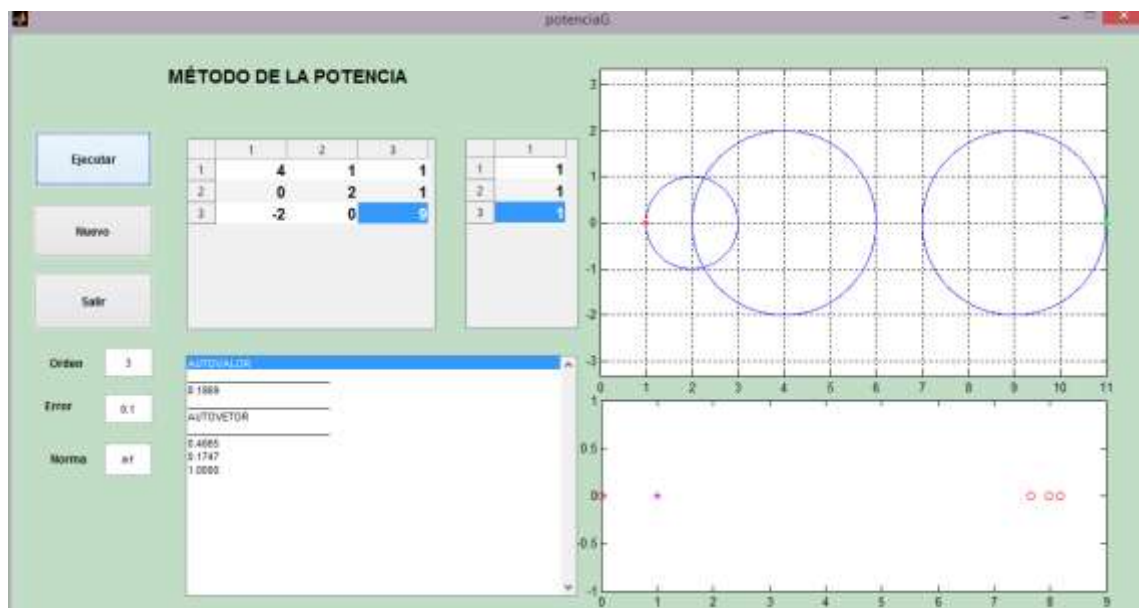
|a(5)|= 8.1869 error = 0.0835
z(5)= 2.8009 1.3495 8.1869
y(4)= 0.4065 0.1747 1.0000

autoValor =

8.1869

autoVetor =

0.4065
0.1747
1.0000



Usando instrucciones del Matlab

```
>> A=[4 1 1;0 2 1;-2 0 9]
```

A =

```
4 1 1
0 2 1
-2 0 9
```

```
[V,D]=eig(A)
```

V =

```

0.2465 -0.8981 -0.3826
0.1477 -0.1562  0.9176
0.9578 -0.4112 -0.1075

```

D =

```

8.4853    0    0
  0  4.6318    0
  0    0  1.8828

```

Luego como el vector hallado de forma aproximada es unitario, tenemos denotando por w al autovector asociado

```
>> w=[0.2465;0.1477;0.9578]
```

w =

```

0.2465
0.1477
0.9578

```

```
>> w/norm(w,inf)
```

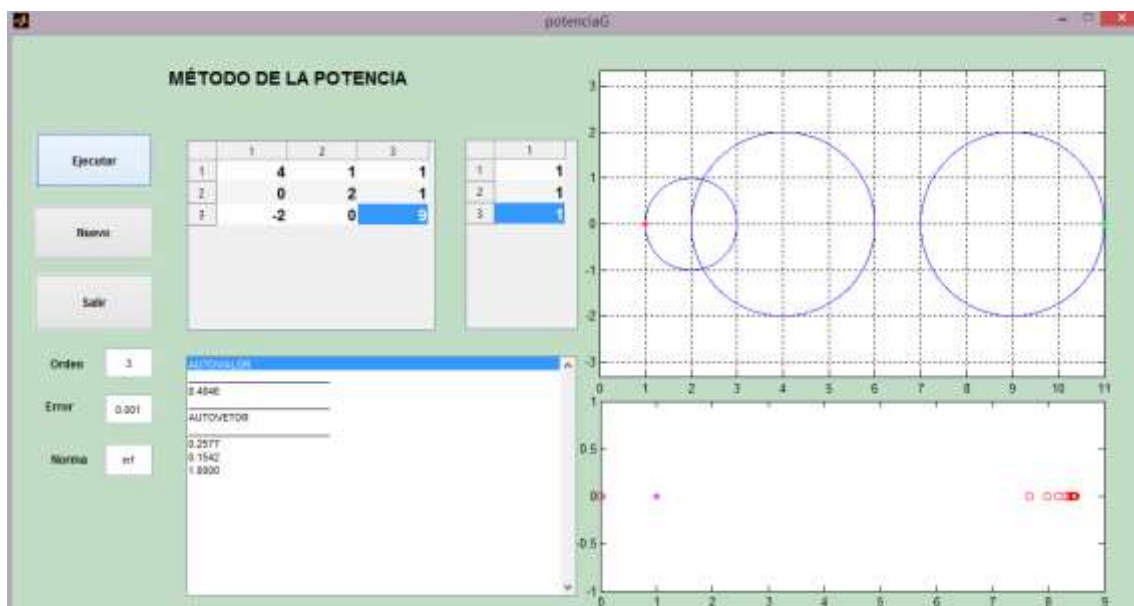
ans =

```

0.2574
0.1542
1.0000

```

Tomando una precisión de 10^{-3}



Obtenemos el autovector

0.2502
0.5000
1.0000

3. MÉTODO DE LA POTENCIA INVERSA

Dada una matriz A este método es usado para determinar el autovalor de menor valor absoluto y su correspondiente autovector. El método puede no funcionar si la matriz no posee autovectores linealmente independiente, más aun el autovalor de menor valor tiene que estar bien separado de los demás en forma similar al método de la potencia tenemos

$$|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_{n-1}| > |\lambda_n|$$

Los autovalores $\mu_1, \mu_2, \dots, \mu_n$ de la matriz A^{-1} son los inversos de los autovalores de A es decir:

$$\mu_i = \frac{1}{\lambda_i} \quad \text{para } i = 1, 2, \dots, n$$

Así

$$\frac{1}{|\lambda_n|} > \frac{1}{|\lambda_{n-1}|} \geq \dots \geq \frac{1}{|\lambda_1|} \Leftrightarrow |\mu_n| > |\mu_{n-1}| \geq \dots \geq |\mu_1|$$

Es decir al autovalor de menor valor absoluto de A le corresponde el de mayor valor absoluto de A^{-1} , por ello aplicando el método de la potencia simple a la matriz A^{-1} se obtendrá el autovalor de menor valor absoluto.

$$\begin{cases} z_{k+1} = A^{-1}y_k \\ y_k = \frac{1}{\alpha_k} z_k \end{cases}$$

y por tanto

$$\lambda_n = \frac{(z_{k+1})_r}{(y_k)_r}$$

En la práctica no es necesario hallar A^{-1} dado que

$$z_{k+1} = A^{-1}y_k \Leftrightarrow Az_{k+1} = y_k$$

Dado que para hallar z_{k+1} involucra resolver un sistema de ecuaciones donde solo cambia el lado derecho, para las iteraciones usaremos la descomposición LU de A

EJEMPLO: Usando el método de la potencia inversa determinar el autovalor de menor valor absoluto de la matriz

$$A = \begin{bmatrix} 2 & 1 & 0 \\ 2 & 5 & 3 \\ 0 & 1 & 6 \end{bmatrix}$$

Con una precisión de 10^{-1}

$$A = LU \quad L = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0.25 & 1 \end{bmatrix} ; \quad U = \begin{bmatrix} 2 & 1 & 0 \\ 0 & 4 & 3 \\ 0 & 0 & 5.25 \end{bmatrix}$$

Tomando $y_0 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$ tenemos

$$\left\{ \begin{array}{l} z_1 = A^{-1}y_0 \Rightarrow Az_1 = y_0 \text{ usando la descomposición LU} \\ \quad z_1 = \begin{bmatrix} 0.5715 \\ -0.1429 \\ 0.1905 \end{bmatrix} \quad \alpha_1 = 0.5715 \\ \quad y_1 = \frac{1}{\alpha_1} z_1 = \begin{bmatrix} 1 \\ -0.2500 \\ 0.3333 \end{bmatrix} \\ z_2 = A^{-1}y_1 \Rightarrow Az_2 = y_1 \text{ usando la descomposición LU} \\ \quad z_2 = \begin{bmatrix} 0.7024 \\ -0.4048 \\ 0.1230 \end{bmatrix} \quad \alpha_2 = 0.7024 \end{array} \right.$$

Calculando el error relativo

$$(\lambda_1^{-1})^{(1)} = \frac{(z_2)_r}{(y_1)_r} = \begin{bmatrix} 0.7024 \\ 1.6192 \\ 0.3690 \end{bmatrix}$$

siendo

$$(\lambda_1^{-1})^{(2)} = \frac{(z_3)_r}{(y_2)_r}$$

Hallaremos y_2 , z_3

$$\left\{ \begin{array}{l} y_2 = \frac{1}{\alpha_2} z_2 = \begin{bmatrix} 1 \\ -0.5763 \\ 0.1751 \end{bmatrix} \\ z_3 = A^{-1}y_2 \Rightarrow Az_3 = y_2 \\ z_3 = \begin{bmatrix} 0.7377 \\ -0.4754 \\ 0.1084 \end{bmatrix} \quad \alpha_3 = 0.7377 \end{array} \right.$$

Luego

$$(\lambda_1^{-1})^{(2)} = \frac{(z_3)_r}{(y_2)_r} = \begin{bmatrix} 0.7377 \\ 0.8249 \\ 0.6192 \end{bmatrix}$$

así

$$\begin{cases} y_3 = \frac{1}{\alpha_3} z_3 = \begin{bmatrix} 1 \\ -0.6444 \\ 0.1469 \end{bmatrix} \\ z_4 = A^{-1} y_3 \Rightarrow Az_4 = y_3 \\ z_4 = \begin{bmatrix} 0.7454 \\ -0.4908 \\ 0.1063 \end{bmatrix} \alpha_4 = 0.7454 \end{cases}$$

$$(\lambda_1^{-1})^{(3)} = \frac{(z_4)_r}{(y_3)_r} = \begin{bmatrix} 0.7454 \\ 0.7617 \\ 0.7235 \end{bmatrix}$$

$$\begin{cases} y_4 = \frac{1}{\alpha_4} z_4 = \begin{bmatrix} 1 \\ -0.6584 \\ 0.1426 \end{bmatrix} \\ z_5 = A^{-1} y_4 \Rightarrow Az_5 = y_4 \\ z_5 = \begin{bmatrix} 0.7471 \\ -0.4942 \\ 0.1061 \end{bmatrix} \alpha_5 = 0.7471 \end{cases}$$

$$(\lambda_1^{-1})^{(4)} = \frac{(z_5)_r}{(y_4)_r} = \begin{bmatrix} 0.7471 \\ 0.7506 \\ 0.7443 \end{bmatrix}$$

$$E = \frac{|(\lambda_1^{-1})^{(4)} - (\lambda_1^{-1})^{(3)}|_r}{|(\lambda_1^{-1})^{(4)}|_r} = \begin{bmatrix} 0.0023 \\ 0.0148 \\ 0.0279 \end{bmatrix}$$

$$\|E\|_\infty = 0.0279$$

Por tanto

$$\lambda_1^{-1} = 0.7471 \Rightarrow \lambda_1 = 1.3385$$

Utilizando un código obtenemos

```
>> A=[2 1 0;2 5 3;0 1 6]
```

```
A =
```

```
2    1    0
2    5    3
0    1    6
```

```
>> y0=[1;1;1]
```

y0 =

1
1
1

>> e=0.01,tip0=inf

e =

0.0100

tipo =

Inf

>> [autoValor,autoVetor]=potenciaINVGERSCH(A,y0,e,tip0)

|a(0)| = 1.0000 error =

z(0)= 0.5714 -0.1429 0.1905

y(0)= 1.0000 1.0000 1.0000

|a(1)| = 0.7024 error = 0.1864

z(1)= 0.7024 -0.4048 0.1230

y(1)= 1.0000 -0.2500 0.3333

|a(2)| = 0.7377 error = 0.0479

z(2)= 0.7377 -0.4754 0.1084

y(2)= 1.0000 -0.5763 0.1751

|a(3)| = 0.7454 error = 0.0104

z(3)= 0.7454 -0.4908 0.1063

y(3)= 1.0000 -0.6444 0.1470

|a(4)| = 0.7471 error = 0.0023

z(4)= 0.7471 -0.4942 0.1061

y(4)= 1.0000 -0.6585 0.1426

autoValor =

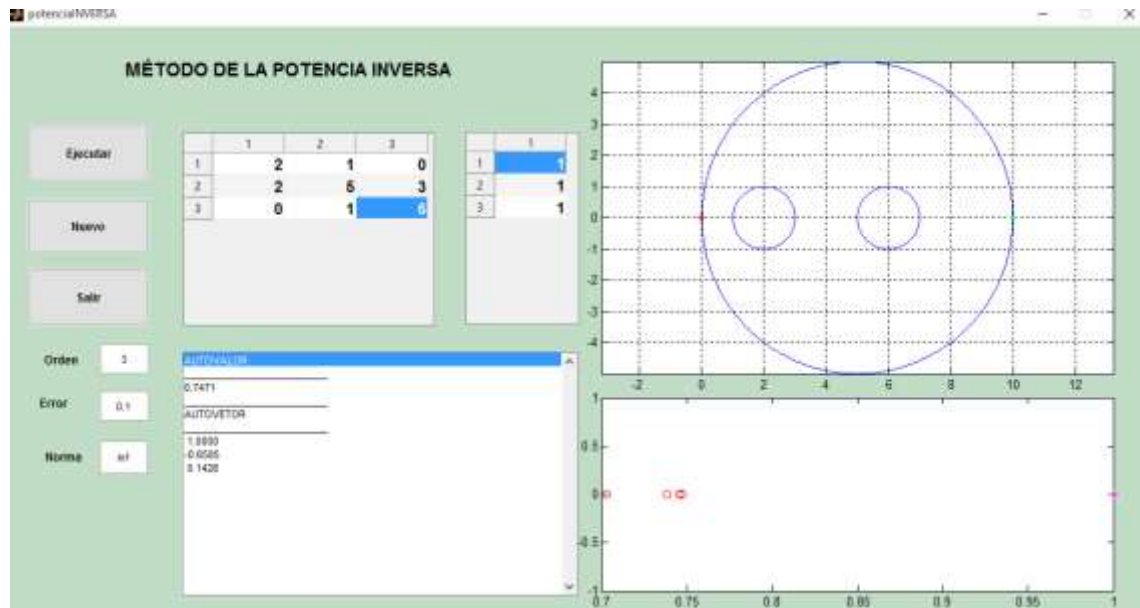
1.3385

autoVetor =

1.0000

-0.6585

0.1426



Utilizando instrucciones del Matlab

```
>> [V,D]=eig(A)
```

V =

```
0.8279  0.3659  0.1493
-0.5484  0.8116  0.8130
0.1176 -0.4555  0.5628
```

D =

```
1.3375    0    0
    0  4.2181    0
    0    0  7.4444
```

4. EL DESPLAZAMIENTO O CAMBIO DE ORIGEN

Si los autovalores de una matriz A son λ_i , para $1 \leq i \leq n$, los valores propios de la matriz $A - \sigma I$ son

$$\lambda_i - \sigma, \quad 1 \leq i \leq n$$

Los autovalores de A y $A - \sigma I$ son idénticos

Considerando que una matriz A tal que todos sus autovalores tienen modulo diferente, si suponemos que σ es una aproximación de un determinado autovalor λ_k , siendo la aproximación buena se verifica que

$$|\lambda_k - \sigma| < |\lambda_i - \sigma| \quad i = 1, 2, \dots, n \quad i \neq k$$

Denominaremos desplazamiento al valor σ , los desplazamientos se utilizan para acelerar la convergencia

5. EL COCIENTE DE RAYLEIGH

El cociente de Rayleigh de un vector x es el escalar

$$r(x) = \frac{x^t A x}{x^t x}$$

Dado un autovalor de una matriz A la mejor estimación del correspondiente autovalor λ de halla mediante la técnica de minimos cuadrados

$$x\lambda \cong Ax$$

Luego

$$x^t x \lambda \cong x^t A x$$

La solución es

$$\lambda = \frac{x^t A x}{x^t x}$$

El cociente de Rayleigh puede utilizarse como desplazamiento para acelerar la convergencia de los métodos iterativos

6. MÉTODO DE DEFLACIÓN

Calcular un autovalor y autovector y obtener los otros por “deflación” si por ejemplo x es un autovector asociado a λ_1 el autovalor dominante de A , siendo H su matriz de Householder tal que $Hx = \alpha e_1$, la transformación de semejanza que determinan transforma A de la siguiente forma

$$A_1 = HAH^{-1} = \begin{bmatrix} \lambda_1 & b^t \\ 0 & A_2 \end{bmatrix}$$

Donde A_2 es de orden $n-1$ cuyos autovalores son $\lambda_1, \lambda_2, \dots, \lambda_n$ los mismos de A , a excepción de λ_1

Luego se sigue un proceso análogo con A_2 para hallar λ_2

Luego de calcular λ_2 si y_2 es el autovector propio de A_2 asociado a λ_2

$$x_2 = H^{-1} \begin{bmatrix} \alpha \\ y_2 \end{bmatrix}$$

Donde $\alpha = \frac{b^t y_2}{\lambda_2 - \lambda_1}$