EJERCICIO COMPUTACIONAL Nº 2. ELIMINACIÓN GAUSSIANA: ACONDICIONAMIENTO Y COSTE OPERATIVO

Ángel Durán

Departamento de Matemática Aplicada
Universidad de Valladolid

25 de febrero de 2011

Contenidos

Definición y ejemplos

- Se dice que una matriz cuadrada invertible A está mal acondicionada si existe una matriz B de manera que cambios pequeños en los elementos de A o B generan cambios grandes en $X = A^{-1}B$. Se dice que el sistema $Ax = B$ está mal acondicionado si A está mal acondicionada.
- Se suele producir mal acondicionamiento cuando la matriz del sistema es casi singular.
- Cuando un sistema está mal acondicionado, los métodos numéricos para su resolución suelen perder eficacia.

Algunos ejemplos

Ejemplo 1: Matrices de Hilbert

$$H_n = (h_{ij})_{i,j=1}^{n}, h_{ij} = 1/(i+j-1), i,j = 1, \dots, n$$

. Por ejemplo

$$H(5) = \begin{pmatrix} 1 & 1/2 & 1/3 & 1/4 & 1/5 \\ 1/2 & 1/3 & 1/4 & 1/5 & 1/6 \\ 1/3 & 1/4 & 1/5 & 1/6 & 1/7 \\ 1/4 & 1/5 & 1/6 & 1/7 & 1/8 \\ 1/5 & 1/6 & 1/7 & 1/8 & 1/9 \end{pmatrix}$$

Algunos ejemplos

Para el término independiente $b = (1, 0, 0, 0, 0)^T$ y utilizando aritmética exacta, la solución del sistema $H(5)x = b$ es $x = (25, -300, 1050, -1400, 630)^T$. Operando con siete cifras significativas, la matriz resultante puede considerarse una perturbación de $H(5)$,

$$\begin{pmatrix} 1{,}0000 & 0{,}5000 & 0{,}3333 & 0{,}2500 & 0{,}2000 \\ 0{,}5000000 & 0{,}3333333 & 0{,}2500000 & 0{,}2000000 & 0{,}1666667 \\ 0{,}3333333 & 0{,}2500000 & 0{,}2000000 & 0{,}1666667 & 0{,}1428571 \\ 0{,}2500000 & 0{,}2000000 & 0{,}1666667 & 0{,}1428571 & 0{,}1250000 \\ 0{,}2000000 & 0{,}1666667 & 0{,}1428571 & 0{,}1250000 & 0{,}1111111 \end{pmatrix}$$

y la solución del sistema es
$x = (28{,}02304, -348{,}5887, 1239{,}781, -1666{,}785, 753{,}5564)^T$.

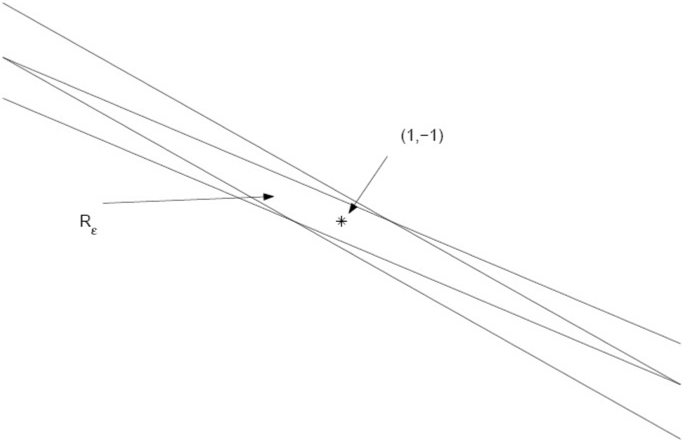
Algunos ejemplos

Ejemplo 2: sistemas casi paralelos

$$
\begin{array}{rcl}
0{,}835x + 0{,}667y = 0{,}168 & | & 0{,}835x + 0{,}667y = 0{,}167 \\
0{,}333x + 0{,}266y = 0{,}067 & | & 0{,}333x + 0{,}266y = 0{,}068 \\
(x, y) = (1, -1) & | & (x, y) = (934, -1169)
\end{array}
$$

$$
\begin{aligned}
R_\epsilon \quad = \quad & \{(x, y)/|0{,}835x + 0{,}667y - 0{,}168| < \epsilon, \\
& |0{,}333x + 0{,}266y - 0{,}067| < \epsilon\}
\end{aligned}
$$

Algunos ejemplos



Número de condición

$$\kappa(A) = ||A^{-1}||||A||.$$

- Observemos que el número de condición depende de la norma matricial elegida.
- $\kappa(A) \geq 1$ en cualquier norma y para cualquier matriz.
- $\kappa(A) >> 1 \Rightarrow$ mal acondicionamiento.

Estimación del acondicionamiento

- Perturbaciones en b: Si $Ax = b, A(x + \delta x) = b + \delta b$

$$\frac{||\delta x||}{||x||} \leq \kappa(A) \frac{||\delta b||}{||b||}$$

- Perturbaciones en A: Si A es no singular, $Ax = b, (A + \delta A)(x + \delta x) = b$ y $\frac{||\delta A||}{||A||} \leq \frac{1}{\kappa(A)}$, entonces

$$\frac{||\delta x||}{||x||} \leq \kappa(A) \frac{\frac{||\delta A||}{||A||}}{1 - \kappa(A)\frac{||\delta A||}{||A||}}.$$

Estimación del acondicionamiento

- Si A es no singular, $Ax = b$, $(A + \delta A)(x + \delta x) = b + \delta b$ y $\frac{||\delta A||}{||A||} \leq \frac{1}{\kappa(A)}$, entonces

$$\frac{||\delta x||}{||x||} \leq \kappa(A) \frac{\frac{||\delta A||}{||A||} + \frac{||\delta b||}{||b||}}{1 - \kappa(A)\frac{||\delta A||}{||A||}}.$$

Ejemplos

$$A = \begin{pmatrix} 1 & 1 \\ 2 & 2 + 10^{-3} \end{pmatrix} \Rightarrow A^{-1} = \begin{pmatrix} 1 + 2 \times 10^3 & -10^3 \\ -2 \times 10^3 & 10^3 \end{pmatrix}$$

$$||A||_\infty = 4{,}001, \, ||A^{-1}||_\infty = 3001$$

$$\Rightarrow \kappa(A)_\infty = 12{,}007 \times 10^3.$$

Ejemplos

Matrices de Hilbert

$$H_n = (h_{ij})_{i,j=1}^{n}, h_{ij} = 1/(i+j-1), i,j = 1,\ldots,n$$

n	$\kappa_2(H_n)$
1	1
2	19.2815
3	524.0568
4	1.5514E+04
5	4.7661E+05

Algoritmo de eliminación con pivotaje

```
function [p,x]=egcp(A,b) // Eliminación gaussiana con pivotaje
// n: tamaño del sistema
// A, b: elementos del sistema
n = length(b);
p = 1 : n;
// Pivotaje, factorización y sustitución progresiva
for k = 1 : n − 1
   maxi = 0; ind = 0;
   for i = k : n
      elem = abs(A(p(i), k));
      if maxi < elem then
         maxi = elem; ind = i;      end
   end
   ind2 = p(k); p(k) = p(ind); p(ind) = ind2;
```

Algoritmo de eliminación con pivotaje

```
    for i = k + 1 : n
      A(p(i), k) = A(p(i), k)/A(p(k), k);
      A(p(i), k + 1 : n) = A(p(i), k + 1 : n)...
      -A(p(i), k) * A(p(k), k + 1 : n);
      b(p(i)) = b(p(i)) - A(p(i), k) * b(p(k));
    end
  end
  // Sustitución regresiva
  b(p(n)) = b(p(n))/A(p(n), n)
  for k = n - 1 : -1 : 1
    b(p(k)) = b(p(k)) - A(p(k), k + 1 : n) * b(p(k + 1) : p(n));
    b(p(k)) = b(p(k))/A(p(k), k);
  end
  x = b(p);
  endfunction
```

Algoritmo de eliminación con pivotaje

$-- >$exec('F:\ telecos \ angel \ clases \ fmii \ egcp.sci',-1)
$-- > A = [-2 \ \ -3 \ \ 14; 2 \ \ 2 \ \ -3; 4 \ \ 2 \ \ -2]; b = [4 \ \ 5 \ \ 10]';$
$-- > [p, x] = egcp(A, b)$
$x =$
2.
2.
1.
$p =$
3. 1. 2.

Algunos comandos

Algunos comandos de SCILAB

$x = A\backslash b$: Solución del sistema $Ax = b$ si A es cuadrada no singular

$x = A\backslash b$: Solución en el sentido mínimos cuadrados si A es cuadrada singular o mal acondicionada

$x = A\backslash b$: Solución en el sentido mínimos cuadrados si A no es cuadrada.

$$||Ax - b||_2 = min_y ||Ay - b||_2$$

det(A)	determinante de la matriz cuadrada A
rank(A)	rango de la matriz
cond(A)	Número de condición de la matriz A en norma 2: $\lambda_{max} / \lambda_{min}$ (Si es muy grande, la matriz está mal condicionada)
rcond(A)	Inverso del número de condición de la matriz A en norma 1: (Si es próximo a 1, la matriz está bien condicionada; si es próximo a cero está mal condicionada)
inv(A)	inversa de la matriz A
lu(A)	factorización LU de la matriz A
chol(A)	factorización de Cholesky de la matriz A
qr(A)	factorizacion QR de la matriz A

Influencia del acondicionamiento

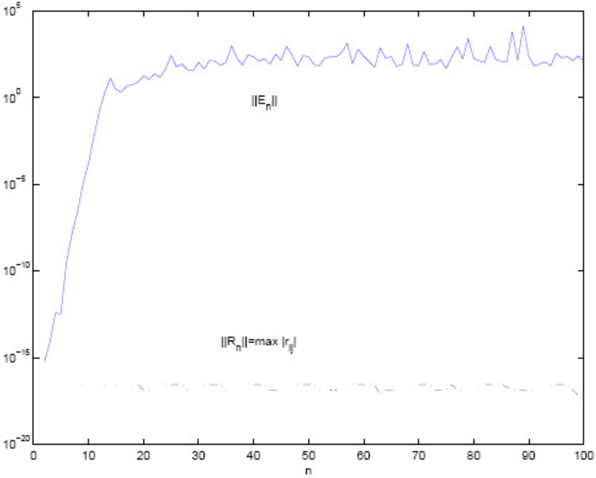
Matrices de Hilbert

$$H_n x_n = b_n,$$
$$x_n = ones(n, 1);$$

$$--> [P_n, L_n, U_n] = lu(H_n)$$
$$--> L_n y = P_n b_n$$
$$--> U_n \tilde{x}_n = y_n$$

$$\begin{aligned} E_n &= \frac{||x_n - \tilde{x}_n||_2}{||x_n||_2} \\ R_n &= L_n U_n - P_n H_n = (r_{ij})_{i,j=1}^{n} \end{aligned}$$

Influencia del acondicionamiento



Algoritmo tridiagonal

```
function b=egt(l,d,s,b)
//Eliminación para sistemas tridiagonales
//Diagonales l,d,s
//Término independiente b
// Sin pivotaje
n = length(d);
//Factorización y sustitución progresiva
for k = 1 : n - 1
   l(k) = l(k)/d(k);
   d(k + 1) = d(k + 1) - l(k) * s(k);
   b(k + 1) = b(k + 1) - l(k) * b(k);
end
```

Algoritmo tridiagonal

```
//Sustitucion regresiva
b(n) = b(n)/d(n);
for k = n-1 : -1 : 1
   b(k) = b(k) - s(k) * b(k+1);
   b(k) = b(k)/d(k);
end
endfunction
```

Algoritmo tridiagonal

$-- >$exec('F:\ telecos \ angel \ clases \ fmii \ egt.sci',-1)
$-- > l = [-1 \quad -1 \quad -1]; d = [2 \;\; 2 \;\; 2 \;\; 2]; s = l; b = [1 \;\; 0 \;\; 0 \;\; 1];$
$-- > b = egt(l, d, s, b)$
$b =$
$\quad 1. \quad 1. \quad 1. \quad 1.$

Comparación de algoritmos

$$-- > A = -2 * diag(ones(N)) + diag(ones(N-1), 1)...$$
$$+diag(ones(N-1), -1);$$
$$-- > b = [-1 \quad zeros(1, N-2) \quad -1];$$

Comparación de algoritmos