Entrega Mètodes Numèrics 2

Christian José Soler

23 de octubre de 2016

1. Exercici 14

1.1. Bound the iteration matrices norm of B_J and B_{GS}

Sigui $A = L + D + U$. On L és la matriu triangular superior amb els elements a sobre de la diagonal d'A, D la matriu diagonal amb els elements de la diagonal d'A i U la matriu triangular inferior amb els elements de sota la diagonal d'A.

Llavors,

$$B_J = -D^{-1}(L+U)$$

Tenim que

$$-D^{-1} = \begin{pmatrix} -\frac{1}{3} & & & \\ & -\frac{1}{3} & & \\ & & \dots & \\ & & & -\frac{1}{3} \end{pmatrix}$$

$$(L+U) = \begin{pmatrix} 0 & & & & & & 1 & 0 \\ 0 & & & & & & & 1 \\ 1 & & & & & & & \\ 0 & \dots & & & & & & \\ & & \dots & & & & & \\ 0 & & & \dots & & & & \\ 1 & & & & \dots & & & \\ 0 & & & & & 1 & 0 & \end{pmatrix}$$

$$B_J = -D^{-1}(L+U) = \begin{pmatrix} 0 & & & & & & -\frac{1}{3} & 0 \\ 0 & & & & & & & -\frac{1}{3} \\ -\frac{1}{3} & & & & & & & \\ 0 & \dots & & & & & & \\ & & \dots & & & & & \\ 0 & & & \dots & & & & \\ -\frac{1}{3} & & & & \dots & & & \\ 0 & & & & & -\frac{1}{3} & 0 & \end{pmatrix}$$

Calculem la norma sub-infinit de la matriu de Jacobi donada la seva senzillesa:

$$||B_J||_\infty = \max_{1\leq i\leq n} \sum_{j=1}^{n} |a_{ij}| = \frac{2}{3}$$

S'ha provat a fer directament la norma de la matriu de Gauss-Seidel, però resulta ser massa complicada, llavors invocarem una demostració feta a classe amb l'enunciat següent, que ens donarà una cota de la norma:

"Sigui A una matriu diagonalment dominant en sentit estricte, aleshores Gauss-Seidel és convergent".

En aquesta demostració hem vist que la norma de B_{GS} es pot acotar per

$$||B_{GS}|| \leq \max_{1\leq k\leq n} \frac{r_k}{1-s_k}$$

on

$$r_k = \sum_{j>k} \left|\frac{a_{kj}}{a_{kk}}\right| \text{ i } s_k = \sum_{j<k} \left|\frac{a_{kj}}{a_{kk}}\right|$$

Per veure quin és el màxim de l'expressió $\frac{r_k}{1-s_k}$, fem casos dins la matriu A, mirant totes les possibilitats per columnes

1. $s_k = \frac{2}{3}$, $r_k = 0 \implies \frac{r_k}{1-s_k} = 0$
2. $s_k = \frac{1}{3}$, $r_k = \frac{1}{3} \implies \frac{r_k}{1-s_k} = \frac{1}{2}$
3. $s_k = \frac{1}{3}$, $r_k = 0 \implies \frac{r_k}{1-s_k} = 0$
4. $s_k = 0$, $r_k = \frac{2}{3} \implies \frac{r_k}{1-s_k} = \frac{2}{3}$

   És a dir tenim que

$$||B_{GS}|| \leq \frac{2}{3}$$

1.2. Solve the linear system using the method of Jacobi.

Tasca completada en 73 iteracions.
Els primers components de la solució del sistema són:
-0.12360659775, -0.12360639775, 0.0472141955, 0.0472143955, -0.01803298875
(Per tota la resta d'algorismes serà igual, llavors no el torno a escriure més)

1.3. Solve the linear system using the method of Gauss-Seidel.

Tasca completada en 37 iteracions.

1.4. Solve the linear system using SOR for several values of ω, trying to find a suitable value for ω. Compare with Gauss-Seidel.

S'ha fet una equipartició de l'interval (0,2) en 20 trossos i amb els ω que s'ha provat el mètode, el millor ha resultat ser $\omega = 1{,}1$ que ha acabat amb 31 iteracions, al contrari que Gauss-Seidel que ha acabat amb 37 iteracions.

2. Exercici 15

2.1. Find an expression for the vector $p = \nabla Q(x)$

Veïem la derivada de $Q(x)$ respecte un x_i per deduïr la gradient:

$$\frac{\partial Q}{\partial x_i} = \frac{1}{2}e_i^t Ax + \frac{1}{2}x^t Ae_i - b^t e_i =$$

Notem que $e_i^t Ax = (x^t Ae_i)^t$ per ser A simètrica i com els dos són una matriu 1x1, vol dir que són iguals. Podem simplificar llavors:

$$= e_i^t Ax - e_i^t b = e_i^t (Ax - b)$$

És a dir, el gradient $\nabla Q(x)$ serà

$$\nabla Q(x) = (Ax - b)^t$$

2.2. Use this method to solve the linear system of Exercise 14.

2.3. Add a relaxation factor ω multiplying the values α_k. Apply the method for several values of the relaxation factor, trying to find a good value for ω.

S'ha fet una equipartició de l'interval (0,2) en 20 trossos i amb els ω que s'ha provat el mètode, el millor ha resultat ser $\omega = 0{,}9$ que ha acabat amb 35 iteracions.

Output final del programa que ho computa tot:

```
    Jacobi algorithm
Solving system:
-0.12360659775, -0.12360639775, 0.0472141955, 0.0472143955, -0.01803298875,  .... Done in 73 iterations
    Gauss-Seidel algorithm
Solving system:
-0.12360659775, -0.12360639775, 0.0472141955, 0.0472143955, -0.01803298875,  .... Done in 37 iterations
    SOR algorithm
Solving system with best parameter = 1.1 :
-0.12360659775, -0.12360639775, 0.0472141955, 0.0472143955, -0.01803298875,  .... Done in 31 iterations
    Steepest Descent algorithm
Solving system with best parameter = 0.9 :
-0.12360659775, -0.12360639775, 0.0472141955, 0.0472143955, -0.01803298875,  .... Done in 35 iterations
```