

Calcolo Numerico : Laboratorio

Lezione

15 aprile 2026

Questa dispensa è il risultato del contributo di varie persone che hanno proposto, elaborato e raffinato i vari esercizi.

1 Metodi di Jacobi e Gauss–Seidel

1.1 Matrici di test

Inserire in Matlab le seguenti matrici.

$$\begin{aligned} A1 &= \begin{bmatrix} 3 & 0 & 4 \\ 7 & 4 & 5 \\ -1 & -2 & 2 \end{bmatrix}, & A2 &= \begin{bmatrix} -3 & 3 & -6 \\ -4 & 7 & -8 \\ 5 & 7 & -9 \end{bmatrix}, \\ A3 &= \begin{bmatrix} 4 & 1 & 1 \\ 2 & -9 & 0 \\ 0 & -8 & -6 \end{bmatrix}, & A4 &= \begin{bmatrix} 7 & 6 & 9 \\ 4 & 5 & -4 \\ -7 & -3 & 8 \end{bmatrix}. \end{aligned}$$

Vogliamo verificare che

- Su $A1$, il metodo di Jacobi non converge ma quello di Gauss–Seidel sì;
- Su $A2$, Jacobi converge (piano) ma Gauss–Seidel no;
- Su $A3$, convergono entrambi e Gauss–Seidel è più veloce;
- Su $A4$, convergono entrambi (piano) e Jacobi è più veloce.

1.2 Jacobi

La formula che definisce il metodo di Jacobi è

$$x_i^{(new)} = \frac{1}{A_{ii}} \left(b_i - \sum_{j=1}^{i-1} A_{ij} x_j^{(old)} - \sum_{j=i+1}^n A_{ij} x_j^{(old)} \right), \quad i = 1, \dots, n.$$

Esercizio 1. Scriviamo insieme una **function** `xnew = jacobi_step(A, b, xold)` che esegue una iterazione del metodo di Jacobi partendo dal vettore x_0 .

Scriviamo poi insieme una **function** `x = jacobi(A, b, x0, m)` che esegue m passi del metodo, scrivendo dopo ogni passo il valore del residuo $\|Ax - b\|_2$, e testiamola sulle quattro matrici di esempio viste sopra.

1.3 Gauss–Seidel

La formula che definisce il metodo di Gauss–Seidel è invece

$$x_i^{(new)} = \frac{1}{A_{ii}} \left(b_i - \sum_{j=1}^{i-1} A_{ij} x_j^{(new)} - \sum_{j=i+1}^n A_{ij} x_j^{(old)} \right), \quad i = 1, \dots, n.$$

Notiamo che l'unica cosa che cambia rispetto a Jacobi è il (new) in rosso.

Esercizio 2. Scrivere una **function** `xnew = gaussseidel_step(A, b, xold)` che esegue un passo del metodo di Gauss–Seidel, e una **function** `x = gaussseidel(A, b, x0, m)` analoga a quella dell'esercizio precedente.

Testare il metodo sulle quattro matrici di test. Quante iterazioni servono per ognuna delle matrici perché il residuo scenda sotto $\varepsilon = 10^{-8}$?

Esercizio 3. Scrivere una **function** `x = gaussseidel_step(A, b, x)` che lavora utilizzando una sola variabile di tipo vettore `x` anziché due (`xold` e `xnew`), sovrascrivendo a ogni passo $i = 1, 2, \dots, n$ l'elemento $x_i^{(k)}$ con $x_i^{(k+1)}$.

Esercizio 4. Testare i due metodi sulla matrice

$$A5 = \begin{bmatrix} 3 & 0 & 4 \\ 7 & 4 & 3 \\ -1 & 1 & 0 \end{bmatrix}.$$

```
>> jacobi(A5, ones(3,1), 5)
warning: in jacobi.m near line 13, column 14:
warning: division by zero
ans = Inf
warning: division by zero
ans = NaN
warning: division by zero
ans = NaN
warning: division by zero
ans = NaN
warning: division by zero
ans = NaN
ans =
      NaN
      NaN
      NaN
```

Come mai i metodi non funzionano?

1.4 Un problema strutturato

Esercizio 5. Siano dati $\alpha \in \mathbb{R}$ e una dimensione $n \geq 3$. Consideriamo la matrice $A \in \mathbb{R}^{n \times n}$ che ha come elementi

$$A_{ij} = \begin{cases} 2 + \alpha & \text{se } i = j, \\ -1 & \text{se } |j - i| = 1 \\ 0 & \text{altrimenti.} \end{cases} \quad (1)$$

Per esempio, per $n = 4$ abbiamo

$$A = \begin{bmatrix} 2 + \alpha & -1 & 0 & 0 \\ -1 & 2 + \alpha & -1 & 0 \\ 0 & -1 & 2 + \alpha & -1 \\ 0 & 0 & -1 & 2 + \alpha \end{bmatrix}.$$

1. Determinare delle formule che permettono di calcolare gli elementi di $x^{(k+1)}$ a partire da quelli di $x^{(k)}$ quando si applica il metodo di Gauss-Seidel alla matrice A nella (1).
2. Scrivere una **function** `xnew = gs_step_strutturato(alpha, b, xold)` che esegue un passo del metodo di Gauss-Seidel per risolvere il problema $Ax = b$ sulla matrice A definita nella (1). La funzione deve usare le formule trovate al punto precedente, e richiedere tempo $O(n)$.
3. Per quali valori di α siete in grado di provare teoricamente la convergenza del metodo?
4. In uno *script*, fissate un valore di α , generate casualmente vettori b e x_0 di lunghezza $n = 10$, e richiamate più volte la funzione del punto precedente per eseguire diverse iterazioni del metodo e controllarne ad occhio la convergenza. Riuscite a determinare sperimentalmente per quali valori di α il metodo converge? Per esempio, provate diversi valori di α compresi tra -1 e 1 .
5. Nello stesso script, calcolate anche il raggio spettrale della matrice di iterazione $H = M^{-1}N$ usando opportune funzioni Matlab (e facendosi aiutare dalle funzioni `tril` e `triu`, di cui potete vedere la documentazione con comandi come `>> help tril`). I risultati sperimentali confermano il teorema visto a lezione che lega la convergenza del metodo al raggio spettrale di H ?

1.5 Esercizi facoltativi

Esercizio 6 (facoltativo). Scrivere due funzioni `x = jacobi_opt(A, b, x0, epsilon)` e `x = gaussseidel_opt(A, b, x0, epsilon)` che eseguono un numero variabile di iterazioni dei due metodi, fermandosi quando il residuo `norm(A*x-b)` scende sotto

ε . Ricordarsi che è buona norma inserire anche un numero massimo di iterazioni, ad esempio `max_iter=100` in modo che la procedura si arresti nel caso non ci sia convergenza o la convergenza sia troppo lenta.

Esercizio 7 (facoltativo). Cosa succede se applichiamo il metodo di Jacobi alla matrice

$$A = \begin{bmatrix} 3 & 1 & 1 \\ -2 & 3 & 1 \\ 2 & 2 & -1 \end{bmatrix},$$

con vettore $b = (5, 2, 3)$, a partire dal vettore

$$x_0 = [3.338379432298626, -1.590892516099053, 4.580399083988357]'$$

Cosa succede invece se prendiamo un vettore di partenza casuale? Sapete spiegare questo comportamento?

Esercizio 8 (facoltativo). Cosa succede se applichiamo i metodi di Jacobi o Gauss Seidel ad una matrice singolare?

Esercizio 9 (facoltativo). Come si può rimediare all'inconveniente che si presenta nell'esercizio 4? Diverse risposte sono possibili...