Modellistica Numerica

Università di Pavia

30 gennaio 2020 — ore 10:00 — aula E9

Punti totali: 33 Durata dell'esame: 2 ore.

Problema 1. Differenze finite di ordine 3 per la derivata prima [11 punti]

Sia $f : \mathbb{R} \to \mathbb{R}$ una funzione liscia, $x \in \mathbb{R}$, $h > 0$.

(a) Mostrare che la differenza finita centrata $D_h^C f(x) = \frac{f(x+h)-f(x-h)}{2h}$ approssima $f'(x)$ con un errore quadratico $\mathcal{O}(h^2)$ per $h \to 0$.

(b) Calcolare $A, B, C, D \in \mathbb{R}$ tali che

$$D_h^\triangle f(x) := \frac{1}{h}\Big(Af(x+h) + Bf(x) + Cf(x-h) + Df(x-2h)\Big)$$

approssimi $f'(x)$ con errore $\mathcal{O}(h^3)$.

Suggerimento: scrivere $D_h^\triangle f(x)$ come somma di espansioni di Taylor di ordine opportuno e scegliere i coefficienti in modo da cancellare i termini indesiderati.

(c) Verificare la correttezza della formula ottenuta controllando che fornisca il valore esatto di $f'(0)$ per $f(x) = x^2$ e $f(x) = x^3$.

Perché sappiamo che per questi polinomi $D_h^\triangle f$ è esatta?

Soluzione: (a) Espandendo con Taylor intorno a x:

$$f(x \pm h) = f(x) \pm hf'(x) + \frac{h^2}{2}f''(x) \pm \frac{h^3}{6}f'''(\xi_\pm) \qquad \text{per } \xi_+ \in (x, x+h), \xi_- \in (x-h, x), \quad \text{quindi}$$

$$f'(x) - D_h^C f(x) = f'(x) - \frac{2hf'(x) + \frac{h^3}{6}f'''(\xi_+) + \frac{h^3}{6}f'''(\xi_-)}{2h} = \frac{h^2}{6}f'''(\xi) \qquad \xi \in (x-h, x+h).$$

(b) Usiamo l'espansione di Taylor $f(x+d) = f(x) + df'(x) + \frac{d^2}{2}f''(x) + \frac{d^3}{6}f'''(x) + \mathcal{O}(d^4)$ per $d = h, -h, -2h$:

$$\begin{aligned}
D_h^\triangle f(x) - f'(x) &= \frac{1}{h}\Big(Af(x+h) + Bf(x) + Cf(x-h) + Df(x-2h)\Big) - f'(x)\\
&= \frac{1}{h}\bigg(A\Big[f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \frac{h^3}{6}f'''(x)\Big] + Bf(x)\\
&\qquad + C\Big[f(x) - hf'(x) + \frac{h^2}{2}f''(x) - \frac{h^3}{6}f'''(x)\Big]\\
&\qquad + D\Big[f(x) - 2hf'(x) + 4\frac{h^2}{2}f''(x) - 8\frac{h^3}{6}f'''(x)\Big] + \mathcal{O}(h^4)\bigg) - f'(x)\\
&= \frac{1}{h}\underbrace{\Big(A + B + C + D\Big)}_{I}f(x) + \underbrace{\Big(A - C - 2D - 1\Big)}_{II}f'(x)\\
&\qquad + \frac{h}{2}\underbrace{\Big(A + C + 4D\Big)}_{III}f''(x) + \frac{h^2}{6}\underbrace{\Big(A - C - 8D\Big)}_{IV}f'''(x) + \frac{1}{h}\mathcal{O}(h^4).
\end{aligned}$$

Basta scegliere i coefficienti in modeo che le 4 parentesi I–II–III–IV siano uguali a 0 e abbiamo ottenuto l'errore desiderato. Sottraendo II da IV otteniamo $D = \frac{1}{6}$. Le ultime due parentesi III e IV diventano $A + C = -\frac{2}{3}$ e $A - C = \frac{4}{3}$, da cui $A = \frac{1}{3}$ e $C = -1$. Dalla I, $B = \frac{1}{2}$. Abbiamo ottenuto

$$D_h^\triangle f(x) = \frac{1}{h}\Big(\frac{1}{3}f(x+h) + \frac{1}{2}f(x) - f(x-h) + \frac{1}{6}f(x-2h)\Big) = \frac{2f(x+h) + 3f(x) - 6f(x-h) + f(x-2h)}{6h}.$$

(c) Con $f(x) = x^2$, $D_h^{\triangle} f(0) = \frac{2h^2+0-6h^2+4h^2}{6h} = 0$. Con $f(x) = x^3$, $D_h^{\triangle} f(0) = \frac{2h^3+0+6h^3-8h^3}{6h} = 0$.

Nell'espansione di Taylor di $f(x+d)$ usata al punto (b), il termine $\mathcal{O}(d^4)$ si può scrivere $\frac{d^4}{24} f^{iv}(\xi)$ per qualche $\xi \in [x-|d|, x+|d|]$. Quindi anche l'errore $D_h^{\triangle} f(x) - f'(x)$ dipende solo dalla derivata quarta di f in $[x-2h, x+h]$. Se f è un polinomio di grado ≤ 3, $f^{iv} = 0$, quindi $D_h^{\triangle} f(x) = f'(x)$.

Problema 2. Problema non lineare [11 punti]

(a) Consideriamo il problema al bordo non lineare

$$\begin{aligned} u''(x) &= F\big(u(x)\big) \qquad x \in (0,1), \\ u(0) &= \alpha, \\ u(1) &= \beta \end{aligned}$$

dove $F \in C^1(\mathbb{R})$ e $\alpha, \beta \in \mathbb{R}$ sono dati. Descrivere come si può approssimare la soluzione u usando il metodo delle differenze finite insieme al metodo di Newton. Usare i nodi equispaziati $x_j = hj$, $h = \frac{1}{n+1}$, $j = 0, \dots, n+1$. Scrivere l'espressione di vettori e matrici coinvolti.

(b) Assumiamo che il problema al bordo ammetta una soluzione u liscia. Definire e stimare l'errore di troncamento commesso dal metodo. Qual è l'ordine di convergenza del troncamento rispetto ad h?

(c) Dare una condizione sufficiente sulla funzione non lineare F che garantisca che tutti i sistemi lineari necessari per il metodo di Newton siano invertibili, indipendentemente dalla scelta di n e del vettore iniziale.

Suggerimento: sfruttare le proprietà delle matrici definite positive o negative.

Soluzione: (a) Approssimiamo u'' con la differenza finita centrata $D_h^{2C} u$. Cerchiamo $\vec{\mathbf{U}} \in \mathbb{R}^n$ tale che

$$G_j(\vec{\mathbf{U}}) := \frac{U_{j-1} - 2U_j + U_{j+1}}{h^2} - F(U_j) = 0, \quad j = 1, \dots, n, \qquad U_0 := \alpha, \quad U_{n+1} := \beta.$$

Questo definisce una funzione non lineare $\mathbf{G} : \mathbb{R}^n \to \mathbb{R}^n$, di cui stiamo cercando uno zero. Le componenti dello zero di $\vec{\mathbf{G}}$ rappresentano l'approssimazione del valore di u nei nodi: $U_j \approx u(x_j)$. Usiamo il metodo di Newton per approssimare $\vec{\mathbf{U}}$: scelto un vettore iniziale $\vec{\mathbf{U}}^0$, costruiamo la successione di vettori

$$\vec{\mathbf{U}}^{k+1} = \vec{\mathbf{U}}^k - \big(\underline{\underline{\mathbf{J}}}\vec{\mathbf{G}}(\vec{\mathbf{U}}^k)\big)^{-1} \vec{\mathbf{G}}(\vec{\mathbf{U}}^k) \qquad k = 0, 1, \dots$$

La matrice Jacobiana ($\underline{\underline{\mathbf{J}}}\vec{\mathbf{G}}_{i,j}(\vec{\mathbf{w}}) = \frac{\partial G_i(\vec{\mathbf{w}})}{\partial w_j}$) è calcolata derivando le componenti di $\vec{\mathbf{G}}$:

$$\underline{\underline{\mathbf{J}}}\vec{\mathbf{G}}(\vec{\mathbf{U}}) = \begin{pmatrix} -\frac{2}{h^2} - F'(U_1) & \frac{1}{h^2} & & & & \\ \frac{1}{h^2} & -\frac{2}{h^2} - F'(U_2) & \frac{1}{h^2} & & & \\ & \frac{1}{h^2} & -\frac{2}{h^2} - F'(U_3) & \frac{1}{h^2} & & \\ & & \ddots & \ddots & \ddots & \\ & & & \frac{1}{h^2} & -\frac{2}{h^2} - F'(U_{n-1}) & \frac{1}{h^2} \\ & & & & \frac{1}{h^2} & -\frac{2}{h^2} - F'(U_n) \end{pmatrix}.$$

(b) Possiamo definire l'errore di troncamento del metodo come $\vec{\mathbf{T}} = \vec{\mathbf{G}}(\vec{\mathbf{u}})$, dove il vettore $\vec{\mathbf{u}}$ di componenti $u_j = u(x_j)$ raccoglie i valori della soluzione esatta nei nodi. Per $j = 1, \dots, n$, abbiamo

$$T_j = \frac{u_{j-1} - 2u_j + u_{j+1}}{h^2} + F(u_j) = u''(x_j) + \frac{h^2}{12} u^{(iv)}(\xi) + F\big(u(x_j)\big) = \frac{h^2}{12} u^{(iv)}(\xi), \qquad \xi \in (x_{j-1}, x_{j+1}),$$

cioè l'errore di troncamento è quadratico in h.

(c) La matrice tridiagonale $\underline{\underline{\mathbf{A}}}$ definita da $A_{j,j} = \frac{2}{h^2}$ e $A_{j,j\pm1} = -\frac{1}{h^2}$ (cioè quella delle differenze finite per $-u'' = f$ con condizioni di Dirichlet) è simmetrica e definita positiva (sappiamo che i suoi autovalori sono positivi). Una matrice diagonale $\underline{\underline{\mathbf{D}}}$ con elementi $D_{j,j} \geq 0$ è semi-definita positiva. Quindi la somma $\underline{\underline{\mathbf{A}}} + \underline{\underline{\mathbf{D}}}$ è definita positiva, in particolare invertibile.

La matrice Jacobiana usata nel metodo di Newton si scrive $\underline{\underline{\mathbf{J}}}\vec{\mathbf{G}}(\vec{\mathbf{U}}) = -(\underline{\underline{\mathbf{A}}} + \underline{\underline{\mathbf{D}}})$, dove $\underline{\underline{\mathbf{D}}}$ è diagonale con $D_{j,j} = F'(U_j)$. Per garantire che la Jacobiana sia definita negativa, quindi invertibile, basta quindi che $\boxed{F'(t) \geq 0}$ per ogni $t \in \mathbb{R}$, cioè che F sia non decrescente.

Problema 3. Perturbazione di rango uno di un sistema lineare [11 punti]

Sia $\underline{\underline{\mathbf{M}}} \in \mathbb{R}^{n\times n}$ una matrice invertibile data. Immaginiamo di avere a disposizione una funzione Matlab `a = FastSolve(b)` che risolve il sistema lineare $\underline{\underline{\mathbf{M}}}\vec{\mathbf{a}} = \vec{\mathbf{b}}$ con complessità computazionale $O(n)$, dato un qualsiasi vettore $\vec{\mathbf{b}} \in \mathbb{R}^n$.

Vogliamo risolvere il sistema perturbato $(\underline{\underline{\mathbf{M}}} + \vec{\mathbf{u}}\vec{\mathbf{w}}^\top)\vec{\mathbf{x}} = \vec{\mathbf{y}}$ dove $\vec{\mathbf{u}}, \vec{\mathbf{w}}, \vec{\mathbf{y}} \in \mathbb{R}^n$ sono vettori dati (intesi come vettori colonna) e $^\top$ indica il trasposto. Ricordiamo che la soluzione $\vec{\mathbf{x}}$ è data dalla formula

$$\vec{\mathbf{x}} = \left(\underline{\underline{\mathbf{I}}} - \frac{(\underline{\underline{\mathbf{M}}}^{-1}\vec{\mathbf{u}})\vec{\mathbf{w}}^\top}{1 + \vec{\mathbf{w}}^\top(\underline{\underline{\mathbf{M}}}^{-1}\vec{\mathbf{u}})}\right)(\underline{\underline{\mathbf{M}}}^{-1}\vec{\mathbf{y}}) \tag{1}$$

dove $\underline{\underline{\mathbf{I}}}$ è la matrice identità. Applichiamo questa formula usando il seguente comando Matlab

```
x = (eye(n) - FastSolve(u)*w' / (1 + w'*FastSolve(u))) * FastSolve(y);
```

dove `y,u,w` sono vettori colonna di lunghezza `n`. (Ricordiamo che `eye(n)` è la matrice identità $n \times n$.)

(a) Questo codice fornisce il vettore `x` corretto ma non è soddisfacente per n grande: perché?

Qual è la complessità computazionale in n di questa funzione?

E la quantità di memoria necessaria (come potenza di n)?

Giustificare le risposte.

(b) Scrivere una breve funzione Matlab che calcoli la soluzione $\vec{\mathbf{x}}$ di $(\underline{\underline{\mathbf{M}}} + \vec{\mathbf{u}}\vec{\mathbf{w}}^\top)\vec{\mathbf{x}} = \vec{\mathbf{y}}$ con la complessità desiderata, sfruttando `FastSolve` e usando la formula (1).

Suggerimento: manipolare (1) in modo da ottenere un'espressione più utile. Usare le proprietà dei prodotti matriciali in modo furbo.

(c) Dimostrare che il vettore $\vec{\mathbf{x}}$ definito da (1) è soluzione del sistema lineare $(\underline{\underline{\mathbf{M}}} + \vec{\mathbf{u}}\vec{\mathbf{w}}^\top)\vec{\mathbf{x}} = \vec{\mathbf{y}}$.

Suggerimento: definire i vettori ausiliari $\vec{\mathbf{p}}$ e $\vec{\mathbf{q}}$ come le soluzioni dei sistemi lineari $\underline{\underline{\mathbf{M}}}\vec{\mathbf{p}} = \vec{\mathbf{y}}$ e $\underline{\underline{\mathbf{M}}}\vec{\mathbf{q}} = \vec{\mathbf{u}}$.

(d) Descrivere un esempio di situazione in cui l'uso del metodo in (1) può essere computazionalmente conveniente.

Soluzione: (a) Il comando mostrato ha due difetti: (i) assembla una matrice densa $n \times n$ e (ii) calcola il prodotto vettore-colonna-per-vettore-riga $(\underline{\underline{\mathbf{M}}}^{-1}\vec{\mathbf{u}})\vec{\mathbf{w}}^\top$ che richiede $n \times n$ moltiplicazioni. Quindi il numero di operazioni richieste e la quantità di memoria necessaria sono entrambi dell'ordine di $O(n^2)$. Il codice non è soddisfacente perché manipolando la formula (1) e usando la proprietà associativa del prodotto si può ottenere lo stesso risultato con costo $O(n)$:

$$\vec{\mathbf{x}} = \underbrace{\left(\underline{\underline{\mathbf{I}}} - \frac{(\underline{\underline{\mathbf{M}}}^{-1}\vec{\mathbf{u}})\vec{\mathbf{w}}^\top}{1 + \vec{\mathbf{w}}^\top(\underline{\underline{\mathbf{M}}}^{-1}\vec{\mathbf{u}})}\right)}_{\text{matrice}} \underbrace{(\underline{\underline{\mathbf{M}}}^{-1}\vec{\mathbf{y}})}_{\text{vettore}} = \underline{\underline{\mathbf{M}}}^{-1}\vec{\mathbf{y}} - \underbrace{(\underline{\underline{\mathbf{M}}}^{-1}\vec{\mathbf{u}})}_{\text{vettore}} \underbrace{\frac{\vec{\mathbf{w}}^\top(\underline{\underline{\mathbf{M}}}^{-1}\vec{\mathbf{y}})}{1 + \vec{\mathbf{w}}^\top(\underline{\underline{\mathbf{M}}}^{-1}\vec{\mathbf{u}})}}_{\text{scalare}}.$$

Moltiplicando $\mathbf{w}^\top$ per $\underline{\underline{\mathbf{M}}}^{-1}\vec{\mathbf{y}}$ abbiamo un prodotto vettore-riga-per-vettore-colonna (cioè il prodotto scalare tra i due vettori), che dà uno scalare con sole n moltiplicazioni.

Inoltre il codice proposto richiede tre chiamate a `FastSolve`, mentre ne bastano due.

(b) Un codice che calcola $\vec{\mathbf{x}}$ con complessità $O(n)$ è ad esempio:

```
1 p = FastSolve(y);
2 q = FastSolve(u);
3 x = p - q * ( w' * p ) / ( 1 + w' * q );
```

Le parentesi nella riga 3 sono fondamentali: se avessimo scritto `(q * w') * p` invece di `q * (w' * p)` avremmo avuto complessità subottimale $O(n^2)$. Se avessimo scritto `q * w' * p` senza specificare le parentesi, Matlab avrebbe calcolato `(q * w') * p` , poiché i prodotti sono valutati da sinistra a destra.

(c) Chiamiamo $\vec{\mathbf{p}}$ e $\vec{\mathbf{q}}$ le soluzioni dei sistemi lineari $\underline{\underline{\mathbf{M}}}\vec{\mathbf{p}} = \vec{\mathbf{y}}$ e $\underline{\underline{\mathbf{M}}}\vec{\mathbf{q}} = \vec{\mathbf{u}}$. Allora la formula (1) diventa $\vec{\mathbf{x}} = \vec{\mathbf{p}} - \vec{\mathbf{q}}\frac{\vec{\mathbf{w}}^\top\vec{\mathbf{p}}}{1+\vec{\mathbf{w}}^\top\vec{\mathbf{q}}}$. Moltiplicando questa espressione per $\underline{\underline{\mathbf{M}}} + \vec{\mathbf{u}}\vec{\mathbf{w}}^\top$ si verifica facilmente:

$$\begin{aligned}(\underline{\underline{\mathbf{M}}} + \vec{\mathbf{u}}\vec{\mathbf{w}}^\top)\vec{\mathbf{x}} &= (\underline{\underline{\mathbf{M}}} + \vec{\mathbf{u}}\vec{\mathbf{w}}^\top)\left(\vec{\mathbf{p}} - \vec{\mathbf{q}}\frac{\vec{\mathbf{w}}^\top\vec{\mathbf{p}}}{1+\vec{\mathbf{w}}^\top\vec{\mathbf{q}}}\right) = \underline{\underline{\mathbf{M}}}\vec{\mathbf{p}} + \vec{\mathbf{u}}\vec{\mathbf{w}}^\top\vec{\mathbf{p}} - \underline{\underline{\mathbf{M}}}\vec{\mathbf{q}}\frac{\vec{\mathbf{w}}^\top\vec{\mathbf{p}}}{1+\vec{\mathbf{w}}^\top\vec{\mathbf{q}}} - \vec{\mathbf{u}}\vec{\mathbf{w}}^\top\vec{\mathbf{q}}\frac{\vec{\mathbf{w}}^\top\vec{\mathbf{p}}}{1+\vec{\mathbf{w}}^\top\vec{\mathbf{q}}}\\ &= \vec{\mathbf{y}} + \vec{\mathbf{u}}(\vec{\mathbf{w}}^\top\vec{\mathbf{p}}) - \vec{\mathbf{u}}\frac{(\vec{\mathbf{w}}^\top\vec{\mathbf{p}})}{1+\vec{\mathbf{w}}^\top\vec{\mathbf{q}}} - \vec{\mathbf{u}}\frac{(\vec{\mathbf{w}}^\top\vec{\mathbf{p}})(\vec{\mathbf{w}}^\top\vec{\mathbf{q}})}{1+\vec{\mathbf{w}}^\top\vec{\mathbf{q}}} = \vec{\mathbf{y}} + \vec{\mathbf{u}}(\vec{\mathbf{w}}^\top\vec{\mathbf{p}})\frac{1+\vec{\mathbf{w}}^\top\vec{\mathbf{q}} - 1 - \vec{\mathbf{w}}^\top\vec{\mathbf{q}}}{1+\vec{\mathbf{w}}^\top\vec{\mathbf{q}}} = \vec{\mathbf{y}}.\end{aligned}$$

(Ricordiamo che $\vec{\mathbf{a}}^\top \vec{\mathbf{b}} = \vec{\mathbf{a}} \cdot \vec{\mathbf{b}}$ è il prodotto scalare tra due vettori mentre $\vec{\mathbf{a}}\vec{\mathbf{b}}^\top$ è una matrice.)

(d) La matrice del metodo delle differenze finite per un problema al bordo con condizioni periodiche è perturbazione di rango uno di una matrice tridiagonale. Il metodo (1) permette di risolvere il sistema corrispondente con complessità $O(n)$.

Un altro esempio è la soluzione di un sistema lineare al variare di un elemento della matrice.