

Prima Prova in Itinere

12/04/2024 — **versione 1** —



32 pt — durata 1h 30' — MS Forms

Gli studenti aventi diritto a svolgere la **prova ridotta** del 30% secondo la L.170/2010 (indicazioni **Multichance** team) **NON** svolgono i quesiti contrassegnati con (***)

TEST — 18 pt

1 — 1 pt

Dato l'insieme dei numeri floating point $\mathbb{F}(2, t, -6, 6)$, si *stim*i il valore minimo di $t \in \mathbb{N}$ affinché l'errore assoluto $|x - fl(x)|$ tra il numero reale $x = \sqrt{\pi}$ e la sua rappresentazione in aritmetica floating point $fl(x)$ sia inferiore a 10^{-10} .

$$2^{-t} \sqrt{\pi} < 10^{-10}, t = 35$$

2 — 1 pt (***) No Multichance

Si considerino 50 sistemi lineari $A \mathbf{x}_j = \mathbf{b}_j$ per $j = 1, \dots, 50$, dove la matrice $A \in \mathbb{R}^{20 \times 20}$ è bidiagonale superiore (ovvero $a_{ij} = 0$ per $i > j$ e $i < j - 1$ per ogni $i, j = 1, \dots, 20$) e invertibile, mentre i vettori $\mathbf{b}_j \in \mathbb{R}^{20}$ rappresentano diversi termini noti. Qual è il numero di operazioni richiesto per la risoluzione di tali sistemi lineari per $j = 1, \dots, 50$ attraverso l'uso computazionalmente più efficiente di un metodo diretto?

$$50(3(20 - 1) + 1) = 2900$$

3 — 2 pt

Si consideri il sistema lineare $A \mathbf{x} = \mathbf{b}$, dove la matrice $A \in \mathbb{R}^{10 \times 10}$ è tale per cui $(A)_{i,j} = \begin{cases} (21 - i - j) & \text{per } j \geq i \\ 0 & \text{per } j < i \end{cases}$, mentre $\mathbf{x}, \mathbf{b} \in \mathbb{R}^{10}$. Quale metodo diretto è computazionalmente conveniente applicare? Sapendo che l'applicazione di tale metodo al calcolatore determina la soluzione numerica $\hat{\mathbf{x}}$ e il residuo normalizzato $\|\mathbf{b} - A\hat{\mathbf{x}}\|/\|\mathbf{b}\| = 10^{-12}$, si *stim*i l'errore relativo commesso $\|\mathbf{x} - \hat{\mathbf{x}}\|/\|\mathbf{x}\|$.

$$\text{algoritmo delle sostituzioni all'indietro, } 9.0240 \cdot 10^{-11}$$

4 — 2 pt (***) No Multichance

Si consideri il sistema lineare $A\mathbf{x} = \mathbf{b}$, con $A = \begin{bmatrix} 5 & -1 & 0 & 0 \\ -1 & 6 & -1 & 0 \\ 0 & -1 & 7 & -1 \\ 0 & 0 & -1 & 8 \end{bmatrix}$ simmet-

rica e definita positiva e $\mathbf{b} = \mathbf{1} \in \mathbb{R}^4$. Indicata con $R \in \mathbb{R}^{4 \times 4}$ la matrice triangolare superiore tale che $A = R^T R$, si riporti il valore dell'elemento $(R)_{2,3}$. Si utilizzi la matrice R per applicare il metodo della fattorizzazione di Cholesky alla soluzione del sistema lineare $A\mathbf{x} = \mathbf{b}$. Si riporti la seconda componente $(\mathbf{y})_2$ del vettore ausiliario \mathbf{y} associato alla soluzione del sistema triangolare inferiore che compare durante l'applicazione del metodo.

$(R)_{2,3} = -0.41523$, $(\mathbf{y})_2 = 0.4983$

5 — 2 pt

Si consideri il sistema lineare $A\mathbf{x} = \mathbf{b}$ con A matrice a valori reali, quadrata e invertibile. Quali delle seguenti affermazioni sono **false**?

1. È sempre possibile applicare il metodo di eliminazione di Gauss senza pivoting essendo $\det(A) \neq 0$.
2. Un metodo iterativo è convergente a \mathbf{x} per ogni scelta dell'iterata iniziale se il raggio spettrale di A è minore di 1.
3. Se A è simmetrica e possiede tutti autovalori positivi allora il metodo di Gauss-Seidel converge a \mathbf{x} per ogni scelta dell'iterata iniziale.
4. Se la matrice A è tridiagonale il metodo di Gauss-Seidel converge a \mathbf{x} per ogni scelta dell'iterata iniziale, inoltre la convergenza è più rapida di quella del metodo di Jacobi.

false (1), (2) e (4)

6 — 2 pt

Sia $A = \begin{bmatrix} 5 & -1 & 0 \\ 1 & 3 & 0 \\ 0 & 0 & 6 \end{bmatrix}$. Quali delle seguenti affermazioni sono **vere**?

1. È possibile applicare il metodo delle iterazioni QR per approssimare simultaneamente tutti gli autovalori di A .
2. Se il vettore iniziale è $\mathbf{x}^{(0)} = \mathbf{1} \in \mathbb{R}^3$, il metodo delle potenze fornisce l'autovalore approssimato $\lambda^{(2)} = 5.4336$ dopo 2 iterazioni del metodo.
3. Se il vettore iniziale è $\mathbf{x}^{(0)} = (1, 1, 0)^T$, il metodo delle potenze converge all'autovalore $\lambda_1(A)$.
4. I cerchi per riga e per colonna di Gershgorin sono coincidenti.

vere (2) e (4)

7 — 2 pt

Si utilizzi opportunamente il metodo di bisezione implementato nella funzione Matlab[®] `bisez.m` per approssimare il punto di massimo della funzione $f(x) = x \sin(x)$ nell'intervallo $I = [1.5, 3.5]$. Si riportino il valore del punto di massimo approssimato e il numero di iterazioni necessarie al metodo di bisezione a partire dall'intervallo I con una tolleranza di 10^{-4} .

2.0287, 14 iterazioni

8 — 2 pt

Si consideri la funzione $f(x) = (e^x - 1)x^2$ dotata dello zero $\alpha = 0$. Si applichi il metodo di Newton modificato implementato nella funzione Matlab[®] `newton.m`, partendo dall'iterata $x^{(0)} = 1.5$, con tolleranza $toll = 10^{-6}$ e in modo da garantire la convergenza quadratica del metodo. Si riportino i valori delle iterate $x^{(1)}$ e $x^{(2)}$ e il numero di iterazioni N_{it} effettuate.

$x^{(1)} = 0.3552$, $x^{(2)} = 0.0210$, $N_{it} = 5$

9 — 2 pt (***) No Multichance

L'applicazione del metodo di Newton ad una funzione $f(x)$ con zero $\alpha = 7$ ed iterata iniziale “sufficientemente” vicina ad α genera le iterate $\{x^{(k)}\}_{k=0,1,\dots}$ convergenti ad α , tra le quali

$$x^{(4)} = 6.675 \quad x^{(5)} = 6.800, \quad x^{(6)} = 6.900, \quad x^{(7)} = 6.980.$$

Si stimino l'ordine di convergenza p del metodo e la molteplicità m dello zero α .

$p = 1$, $m = 5$

10 — 2 pt

Per approssimare la radice quadrata $\sqrt{19}$, si può usare il metodo corrispondente al seguente algoritmo

$$x^{(k+1)} = \frac{1}{2} \left(x^{(k)} + \frac{19}{x^{(k)}} \right) \quad \text{per } k = 0, 1, \dots,$$

con $x^{(0)}$ assegnato. Il metodo converge per ogni $x^{(0)}$ “sufficientemente” vicino a $\sqrt{19}$? Se converge, con quale ordine?

si, 2

ESERCIZIO – 14 pt

Si consideri il sistema lineare $A\mathbf{x} = \mathbf{b}$, dove $A \in \mathbb{R}^{n \times n}$ è una matrice simmetrica e definita positiva e $\mathbf{x}, \mathbf{b} \in \mathbb{R}^n$ per $n \geq 1$. In particolare, poniamo $n = 10$,

$$A = \text{pentadiag}(1, -5, 8, -5, 1) \in \mathbb{R}^{10 \times 10},$$

mentre $\mathbf{b} \in \mathbb{R}^{10}$ è tale per cui $\mathbf{x} = \mathbf{1} \in \mathbb{R}^{10}$.

NOTA: Si riportino nelle risposte ai seguenti punti: tutti i comandi Matlab[®] usati, tutte le funzioni Matlab[®] implementate, le descrizioni dei procedimenti usati, le giustificazioni teoriche dei risultati e tutte le definizioni della notazione.

Punto 1) — 3 pt

Si considerino i metodi di Jacobi e Gauss–Seidel per l'approssimazione della soluzione \mathbf{x} del sistema lineare.

- Si determini se i metodi di Jacobi e Gauss–Seidel convergono a \mathbf{x} per ogni scelta dell'iterata iniziale $\mathbf{x}^{(0)} \in \mathbb{R}^{10}$.
- Si applichi il metodo di Gauss–Seidel usando la funzione Matlab[®] `gs.m` con iterata iniziale $\mathbf{x}^{(0)} = \mathbf{0} \in \mathbb{R}^{10}$ e tolleranza sul criterio d'arresto del residuo normalizzato $tol = 10^{-6}$. Si riportino il numero di iterazioni N_{it} effettuate e il valore del residuo normalizzato $\|\mathbf{b} - A\mathbf{x}^{(N_{it})}\|/\|\mathbf{b}\|$ corrispondente.
- Dopo aver risposto al punto precedente, si *stim*i l'errore relativo e si verifichi che è effettivamente superiore all'errore relativo $\|\mathbf{x} - \mathbf{x}^{(N_{it})}\|/\|\mathbf{x}\|$ commesso applicando il metodo.

Spazio per risposta lunga $\rho_{BJ} = 1.4137 > 1$, $\rho_{BGS} = 0.9726 < 1$, $N_{it} = 396$,
 $res_{norm} = 9.8498 \cdot 10^{-7}$, $err_{stim} = 1.7349 \cdot 10^{-4} > err_{rel} = 1.5982 \cdot 10^{-5}$

Punto 2) — 3 pt

Si consideri il metodo del gradiente preconditionato per l'approssimazione di \mathbf{x} , con le seguenti matrici di preconditionamento (precondizionatori):

$$P_1 = I, \quad P_2 = \text{tridiag}(-4, 9, -4), \quad P_3 = \text{tridiag}(-1, 2, -1) \in \mathbb{R}^{10 \times 10}.$$

- Per quale preconditionatore P_1, P_2 oppure $P_3 \in \mathbb{R}^{10 \times 10}$ è garantita la convergenza più rapida del metodo per ogni scelta dell'iterata iniziale $\mathbf{x}^{(0)} \in \mathbb{R}^{10}$?
- Usando il preconditionatore selezionato, si applichi il metodo usando la funzione Matlab[®] `richardson.m` con $\mathbf{x}^{(0)} = \mathbf{0} \in \mathbb{R}^{10}$ e tolleranza sul criterio d'arresto basato sul residuo normalizzato $tol = 10^{-6}$. Si riportino il numero di iterazioni N_{it} effettuate e il valore dell'errore $\|\mathbf{x}^{(N_{it})} - \mathbf{x}\|_A$ corrispondente.

Spazio per risposta lunga $P_3, K(P_3^{-1}A) = 3.8889$, $N_{it} = 23$, $err = 2.0734 \cdot 10^{-6}$

Punto 3) — 2 pt (***) No Multichance

Si consideri ora il metodo del gradiente coniugato (non preconditionato).

- In quante iterazioni il metodo converge a \mathbf{x} , in aritmetica esatta, per ogni scelta dell'iterata iniziale $\mathbf{x}^{(0)} \in \mathbb{R}^{10}$?
- Si utilizzi opportunamente la funzione Matlab[®] `pcg` per approssimare \mathbf{x} , avendo scelto $\mathbf{x}^{(0)} = \mathbf{0} \in \mathbb{R}^{10}$ e tolleranza sul criterio d'arresto del residuo normalizzato $tol = 10^{-6}$. Si riportino il numero di iterazioni N_{it} effettuate e il valore dell'errore $\|\mathbf{x}^{(N_{it})} - \mathbf{x}\|_A$ corrispondente.

Spazio per risposta lunga al più 10 it, $N_{it} = 5$, $err = 1.3035 \cdot 10^{-13}$

Punto 4) — 3 pt

Si vuole approssimare l'autovalore $\lambda_9(A)$ con il metodo delle potenze inverse con shift.

- Si determini il valore dello shift $s \in \mathbb{R}$ applicando opportunamente 3 iterazioni del metodo delle iterazioni QR alla matrice $A \in \mathbb{R}^{10 \times 10}$.
- Dopo aver determinato s , si utilizzi la funzione Matlab[®] `invpowershift.m` con tolleranza $tol = 10^{-3}$ e vettore iniziale $\mathbf{x}^{(0)} = (-1, 1, -1, 1, \dots, -1, 1)^T \in \mathbb{R}^{10}$ per approssimare $\lambda_9(A)$. Si riportino il numero di iterazioni effettuate N_{it} , l'autovalore approssimato $\lambda^{(N_{it})}$, e l'autovettore di norma unitaria corrispondente $\mathbf{y}^{(N_{it})}$.

Spazio per risposta lunga $s = 0.5414; 3, 0.5057, (0.1891, 0.3705, 0.4355, \dots, -0.1891)^T$

Punto 5) — 3 pt (*) No Multichance**

Si consideri una funzione $\Phi : \mathbb{R}^n \rightarrow \mathbb{R}$, dotata del punto di minimo $\alpha \in \mathbb{R}^n$ e del gradiente $\nabla\Phi(\mathbf{y}) \in \mathbb{R}^n$ per $\mathbf{y} \in \mathbb{R}^n$. L'approssimazione di $\alpha \in \mathbb{R}^n$ si può determinare applicando il seguente metodo iterativo, dove l'iterata $\mathbf{x}^{(k)} \approx \alpha$.

Algorithm 1: Metodo di steepest-descent

```
scegliere  $\mathbf{x}^{(0)} \in \mathbb{R}^n$ ;  
 $\gamma^{(0)} = 0.1$ ;  
for  $k = 0, 1, \dots$ , fino a che  $\|\nabla\Phi(\mathbf{x}^{(k)})\| > tol$  do  
     $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \gamma^{(k)} \nabla\Phi(\mathbf{x}^{(k)})$  ;  
     $\gamma^{(k+1)} = \frac{\left|(\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)})^T (\nabla\Phi(\mathbf{x}^{(k+1)}) - \nabla\Phi(\mathbf{x}^{(k)}))\right|}{\|\nabla\Phi(\mathbf{x}^{(k+1)}) - \nabla\Phi(\mathbf{x}^{(k)})\|^2}$ ;  
end
```

Si implementi l'algoritmo in Matlab[®] per $\Phi(\mathbf{y}) = \sin\left(\frac{1}{n}\mathbf{y}^T\mathbf{y}\right) + \frac{1}{2}\mathbf{y}^T A \mathbf{y}$, dove la matrice $A \in \mathbb{R}^{n \times n}$ è stata precedentemente assegnata; inoltre, il gradiente della funzione è $\nabla\Phi(\mathbf{y}) = \frac{2}{n} \cos\left(\frac{1}{n}\mathbf{y}^T\mathbf{y}\right) \mathbf{y} + A\mathbf{y}$. Si pongano $n = 10$, $\mathbf{x}^{(0)} = \mathbf{1} \in \mathbb{R}^{10}$ e la tolleranza $tol = 10^{-3}$. Si riportino il numero di iterazioni N_{it} eseguite e le approssimazioni $x_1^{(1)} = \left(\mathbf{x}^{(1)}\right)_1$, $x_1^{(2)} = \left(\mathbf{x}^{(2)}\right)_1$ e $x_1^{(N_{it})} = \left(\mathbf{x}^{(N_{it})}\right)_1$ ottenute.

Spazio per risposta lunga $N_{it} = 44, 0.5892, 0.5633, 0.0001$