ANALISI MATEMATICA II

ESERCITAZIONE 2

Argomento: sistemi lineari

1. Selezionare il formato di output `format long e` e risolvere i sistemi lineari $Ax = b$ con A matrice di Hilbert di ordine $n = 5, 10, 15$ e b definito in modo tale che il corrispondente vettore soluzione x coincida con il vettore unitario. Per ciascun valore di n, calcolare l'errore relativo della soluzione in norma euclidea e il numero di condizionamento. Commentare i risultati.

2. Generare la seguente matrice
$$A = \begin{pmatrix} 11 & 2 & 3 & 4 \\ 0 & 8 & 2 & 3 \\ 0 & 0 & 4 & 0 \\ 1 & 0 & 0 & 5 \end{pmatrix}.$$
Determinare le matrici P, L e U della fattorizzazione $PA = LU$ della matrice A mediante la *function* `lu` di Matlab. Successivamente utilizzare i suddetti fattori per risolvere il sistema lineare $Ax = b$, con b definito in modo tale che la corrispondente soluzione x coincida con il vettore unitario.

3. Generare una matrice A di numeri pseudo-casuali, non singolare e di ordine 3, e determinare le matrici P, L e U della fattorizzazione $PA = LU$ della matrice A mediante la *function* `lu` di Matlab. Successivamente utilizzare i suddetti fattori per invertire la matrice A. Utilizzare la *function* `inv` di Matlab per verificare la correttezza del risultato.

4. Generare una matrice B di numeri pseudo-casuali, non singolare e di ordine 5. Verificare che la matrice $A = B^T B$ è simmetrica e definita positiva ed utilizzare la *function* `chol` di Matlab per determinare la decomposizione di Choleski $A = R^T R$.

   Successivamente utilizzare la suddetta decomposizione per calcolare la matrice inversa di A e per risolvere il sistema lineare $Ax = b$, con b definito in modo tale che la corrispondente soluzione x coincida con il vettore unitario.

5. Risolvere in modo efficiente, minimizzando il numero di operazioni aritmetiche, i seguenti sistemi
$$\begin{cases} Ax_1 = b_1 \\ Ax_2 = b_2 \\ Ax_3 = b_3 \\ Ax_4 = b_4 \end{cases}$$
aventi tutti la stessa matrice dei coefficienti A, non singolare, di ordine 6 e costituita da numeri pseudo-casuali; inoltre il termine noto b_1 è definito in modo tale che la corrispondente soluzione x coincida con il vettore unitario e $b_i = b_{i-1}/i$, $i = 2, 3, 4$.

6. Risolvere in modo efficiente il sistema
$$A^4 z = b$$
ove A è non singolare, di ordine 5 e costituita da numeri pseudo-casuali e b è definito in modo tale che la corrispondente soluzione z coincida con il vettore unitario.

7. Implementare in due *m-file* di tipo *function*, denominati `jacobi.m` e `gauss_seidel.m`, i metodi iterativi di Jacobi e di Gauss-Seidel rispettivamente, per la risoluzione di un sistema lineare $Ax = b$.

Strutturare le *function* `jacobi` e `gauss_seidel` in modo tale che, ricevendo in input la matrice dei coefficienti A, il termine noto b, un numero massimo di iterazioni $itermax$, una tolleranza relativa $toll$ ed un vettore colonna iniziale $x^{(0)}$, restituiscano in output il vettore soluzione $x = x^{(k)}$ tale che $||x^{(k)} - x^{(k-1)}||_\infty \leq toll\,||x^{(k)}||_\infty$ e il numero di iterazioni $iter$ eseguite per raggiungere la tolleranza richiesta. Costruire gli algoritmi utilizzando la formula iterativa dei suddetti metodi in forma matriciale. Infine, dopo aver selezionato il formato di output `format long e`, confrontare i suddetti metodi nella risoluzione dei sistemi $A_i x = b_i$, $i = 1, \ldots, 4$, con

$$A_1 = \begin{pmatrix} 1 & -2 & 2 \\ -1 & 1 & -1 \\ -2 & -2 & 1 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 4 & 0 & 2/5 \\ 0 & 5 & 2/5 \\ 5/2 & 2 & 1 \end{pmatrix},$$

$$A_3 = \begin{pmatrix} 2 & -1 & 1 \\ 2 & 2 & 2 \\ -1 & -1 & 2 \end{pmatrix}, \quad A_4 = \begin{pmatrix} 3 & -1 & 0 & 0 & 0 & -1 \\ -1 & 3 & -1 & 0 & -1 & 0 \\ 0 & -1 & 3 & -1 & 0 & 0 \\ 0 & 0 & -1 & 3 & -1 & 0 \\ 0 & -1 & 0 & -1 & 3 & -1 \\ -1 & 0 & 0 & 0 & -1 & 3 \end{pmatrix},$$

e b_i definito in modo tale che la corrispondente soluzione x coincida con il vettore unitario. Per ciascuno di essi richiedere la tolleranza relativa $toll = 1.0\mathrm{e}-7$, un numero massimo di iterazioni $itermax = 100$ e fornire il vettore nullo come vettore iniziale $x^{(0)}$. Infine, calcolare e stampare in ciascuna *function* il raggio spettrale della corrispondente matrice di iterazione e commentare i risultati.