Calcolo Numerico, III Esercitazione
CdL Informatica

Esercizi svolti a lezione
1) Scrivere una function `cholesky` che realizzi la fattorizzazione di Cholesky

$$A = \mathcal{L}\mathcal{L}^T.$$

per una matrice $A \in \mathbb{R}^{n \times n}$ simmetrica e definita positiva. La function deve restituire un messaggio di errore qualora l'algoritmo non sia applicabile (ossia qualora A non sia definita positiva). Scrivere uno script in cui si genera una matrice $A \in \mathbb{R}^{n \times n}$ simmetrica strettamente diagonale dominante a righe, si genera il vettore $b \in \mathbb{R}^n$ tale che $b = A * \texttt{ones}(\texttt{n}, \texttt{1})$ e si risolve il sistema $Ax = b$ usando la function `cholesky`. Confrontare il risultato ottenuto con quello fornito dalle function `chol` e `lu` di Matlab.

2) Scrivere una function `myQR` che realizzi la fattorizzazione

$$A = QR.$$

ove Q è ortogonale definita tramite le trasformazioni di Householder e R è triangolare superiore.

Scrivere uno script in cui la function `myQR` viene utilizzata per risolvere un sistema lineare con A generata in modo pseudo casuale.

3) Scrivere una function `jacobi` che, data una matrice $A \in \mathbb{R}^{n \times n}$ ed un vettore colonna $b \in \mathbb{R}^n$, implementi il metodo iterativo di Jacobi per risolvere il sistema lineare $Ax = b$.
Scrivere uno script che

1. costruisce la matrice tridiagonale A di dimensione 20 (con diagonale di tutti elementi -3 e sotto/sopra diagonali di tutti elementi 1);
2. il vettore b di dimensione 20 di tutti 1;
3. calcola la soluzione del sistema lineare $Ax = b$ con l'operatore $\backslash$ di Matlab;
4. calcola 10 iterazioni del metodo di Jacobi utilizzando la function del punto precedente;
5. calcola il massimo dei 10 errori relativi.

4) Scrivere una function `gaussSeidel` che, data una matrice $A \in \mathbb{R}^{n \times n}$ ed un vettore colonna $b \in \mathbb{R}^n$, implementi il metodo iterativo di Gauss-Seidel per risolvere il sistema lineare $Ax = b$.
Scrivere uno script che

1. costruisce la matrice tridiagonale A di dimensione 20 (con diagonale di tutti elementi -3 e sotto/sopra diagonali di tutti elementi 1);
2. il vettore b di dimensione 20 di tutti 1;

3. calcola la soluzione del sistema lineare $Ax = b$ con l'operatore $\backslash$ di Matlab;
4. calcola 10 iterazioni del metodo di Gauss-Seidel utilizzando la function del punto precedente;
5. calcola il massimo dei 10 errori relativi.

Esercizi suggeriti per casa

1) Scrivere una function che, date 2 matrici M (invertibile) ed N di ordine n, verifichi la convergenza di un metodo iterativo per la risoluzione di $Ax = b$ dove $A = M - N$ (si utilizzi il comando `eig()` di Matlab).
Scrivere inoltre una function che, date 2 matrici M (invertibile), N di ordine n ed un vettore b di dimensione $1 \times n$, implementi (vettorialmente) il metodo iterativo per la risoluzione di $Ax = b$ dove $A = M - N$, dopo averne verificato la convergenza utilizzando la function del punto precedente.
Scrivere infine uno script che

1. costruisce la matrice tridiagonale A di dimensione 20 (con diagonale di tutti elementi -3 e sotto/sopra diagonali di tutti elementi 1);
2. il vettore b di dimensione 20 di tutti 1;
3. calcola la soluzione del sistema lineare $Ax = b$ con l'operatore \ di Matlab;
4. costruisce uno splitting di $A = M - N$ (a vostra scelta);
5. verifica la convergenza del metodo iterativo su di esso basato ed, in caso affermativo, calcola 10 iterazioni del metodo iterativo.

2) Scrivere una function che, dati in ingresso una matrice tridiagonale $A \in \mathbb{R}^{n \times n}$ ed un vettore $b \in \mathbb{R}^n$, restituisce la soluzione del sistema lineare $Ax = b$ mediante il metodo di Gauss, utilizzando il seguente algoritmo semplificato valido per matrici tridiagonali:

Dati: $A \in \mathbb{R}^{n \times n}$ tridiagonale, $b \in \mathbb{R}^n$

1. Poni $u_1 = A_{1,1}$, e $d_1 = b_1$
2. Ripeti per $i = 2, 3, \ldots, n$
   2.1 Poni $m_i = A_{i,i-1}/u_{i-1}$
   2.2 Poni $u_i = A_{i,i} - m_i A_{i-1,i}$ e $d_i = b_i - m_i d_{i-1}$
3. Poni $x_n = d_n/u_n$
4. Ripeti per $i = n-1, n-2, \ldots, 1$
   4.1 Poni $x_i = (d_i - A_{i,i+1} x_{i+1})/u_i$
5. Fine.

Utilizzare la function definita per risolvere il sistema lineare generato nell'esercizio 3 svolto a lezione. Confrontare i risultati ottenuti con quelli forniti dall'operatore `backslash`.