Tesine del Corso di Calcolo Numerico (CLS Elettronica)

Prof.ssa Laura Pezza   a.a. 2010-11.

I gruppo

1. Scrivere un sottoprogramma per il calcolo di A^{-1} mediante la fattorizzazione LU e testarlo per varie matrici A (vedi libro L. Gori, M.L. Lo Cascio, F. Pitolli), tra cui le matrici di Hilbert e di Vandermonde. Confrontare i risultati con quelli ottenuti attraverso il comando   >> **inv**(A) (*).

Utilizzando il sottoprogramma realizzato, scrivere un programma per il calcolo del *numero di condizionamento* di una matrice e testarlo per le suddette matrici. Confrontare i risultati con quelli ottenuti attraverso il comando   >> **cond**(A) (*).

2. Scrivere tre sottoprogrammi per la soluzione di un sistema lineare tridiagonale con la fattorizzazione LU, con il metodo di Jacobi e con il metodo di Gauss-Seidel e testarli per vari sistemi lineari (vedi libro "Esercizi di Calcolo Numerico", L.Gori, M.L. Lo Cascio, F. Pitolli). Utilizzare i sottoprogrammi per risolvere il *sistema lineare*

$$\begin{cases} 2x_1 - x_2 \qquad = 1 \\ -x_{i-1} + 2x_i - x_{i+1} = 0, \quad \text{i=2,...,n-1} \\ \qquad -x_{n-1} + 2x_n = 1 \end{cases}$$

per $n = 20$. Confrontare l'accuratezza dei risultati ed il costo computazionale. Verificare che il metodo di Gauss-Seidel ha una velocita' di convergenza circa doppia rispetto al metodo di Jacobi.

3. In riferimento al sistema lineare dell'esercizio precedente con $n = 10$, si calcoli $\rho(G_{GS})$ con il metodo delle potenze. Si utilizzi tale valore per calcolare il parametro ottimo ω_0 del metodo del rilassamento e si utilizzi tale metodo per risolvere il suddetto sistema.
(si veda il libro di Laura Gori "Calcolo Numerico")

(*) La fattorizzazione LU puo' essere sviluppata o tramite l'algoritmo di eliminazione di Gauss o con l'algoritmo di Banachiewitz-Doolittle (rif. libro di Laura Gori "Calcolo Numerico").

4. Data l'*equazione non lineare*

$$f(x) = e^{-x} - 2cos(x) = 0,$$

approssimare la radice negativa ξ con il metodo delle bisezioni, con il metodo delle secanti ad estremi variabili ed infine con quello di Newton. Arrestare i tre processi iterativi quando l'errore e_k risulta in modulo minore di $0.5{*}10^{-4}$. Confrontare le velocita' di convergenza ed il costo computazionale dei tre metodi.

5. Data l'*equazione non lineare* dipendente da parametro reale α

$$e^{-x} - \alpha cos(x) = 0,$$

per $\alpha = 2$ approssimare l'unica radice negativa, per $\alpha = -1$ approssimare la radice minore positiva, utilizzando in entrambi i casi il metodo di Newton, e nel I caso anche il metodo delle approssimazioni successive generate dalla funzione di iterazione $\varphi(x) = -log(\alpha cos(x))$. Arrestare entrambi i processi iterativi quando l'errore e_k risulta in modulo minore di $0.5 * 10^{-4}$. Confrontare le velocita' di convergenza dei due metodi per entrambi i valori di α.

II gruppo

1. Si considerino i valori di $f(x) = sin(x)$ nei nodi di $[0, 1]$: $x_i = ih$, $i = 0, ..., 4$, $h = 0.25$. A partire da esssi si calcolino: $p_4(x)$, polinomio interpolatore, $P_4(x)$, polinomio ai minimi quadrati, $T_k(x)$, $k = 1, 2, 3$, sviluppo di Taylor in $x_0 = 0$ all'ordine k. Si diano stime numeriche dell'errore globale in $x = 0.3$ fornite dai 5 polinomi e le si confrontino con i relativi errori effettivi. Si confrontino inoltre, su tutto $[0, 1]$, i grafici dei 5 polinomi costruiti e della funzione $f(x)$.

2A. Preparare un programma che, a partire dai valori $\{x_i\}_{i=0}^n$ ed $\{f_i\}_{i=0}^n$, costruisca $P_n(x)$, *polinomio interpolatore alle differenze divise* di grado n.

Utilizzare il programma per costruire la tavola alle differenze divise della funzione

$$f(x) = \begin{cases} xsinx & \text{per } x \leq 0 \\ sin^2 x & \text{per } x > 0 \end{cases}$$

nei nodi $\{-0.4, -0.3, 0.1, 0.15, 0.25\}$. Costruire il polinomio interpolatore di terzo grado $P_3(x)$ utilizzando i primi quattro nodi e calcolarlo in $x^* = 0$ e $x^* = 0.2$.

Stimare l'errore commesso e confrontarlo con il valore dell'errore effettivo $f(x^*) - P_3(x^*)$.

2B. Preparare un programma che, a partire dai valori $\{x_i\}_{i=0}^n$ ed $\{f_i\}_{i=0}^n$, costruisca $P_n(x)$, *polinomio interpolatore di Lagrange* di grado n.

Utilizzare il programma per costruire il polinomio interpolatore di terzo grado $P_3(x)$ relativo alla funzione dell'esercizio precedente nei nodi $\{-0.4, -0.3, 0.1, 0.15\}$. Stimare l'errore commesso in $x^* = 0$ e $x^* = 0.2$ e confrontarlo con il valore dell'errore effettivo $f(x^*) - P_3(x^*)$.

3. Verificare il *fenomeno di Runge* per la funzione $f(x) = \frac{1}{1+x^2}$.

Scegliere vari intervalli e verificare per quali di essi si ha la convergenza dei polinomi interpolatori. Verificare la convergenza delle spline cubiche naturali, dopo averle costruite secondo l'algoritmo riportato nel libro di Laura Gori "Calcolo Numerico".

4. Preparare un programma che calcoli l'integrale definito di una funzione con il *Metodo di Romberg.*

Utilizzarlo per calcolare $log(1.5)$ con 6 cifre decimali esatte, approssimando l'integrale

$$\int_0^{0.5} \frac{1}{1+x} dx,$$

senza far uso delle derivate della funzione integranda.

5. Risolvere un *problema differenziale* del primo ordine (vedi libro L.Gori, M.L. Lo Cascio, F. Pitolli) con il metodo di Eulero ed il metodo di Heun e confrontare i risultati. Studiare il comportamento della soluzione approssimata al diminuire del passo.

6. La crescita di una popolazione puo' essere studiata, in un modello semplificato, attraverso la seguente equazione differenziale detta "equazione logistica":

$$\begin{array}{l} \frac{dy}{dt} = (a - y(t))y(t) \quad t \in (0, 10) \\ y_0 = 0.1 \end{array}$$

Si utilizzi il metodo di Runge-Kutta del 4 ordine per trovare, per ciascuno dei seguenti valori del parametro a: $a = 1$, $a = 1.4$ e $a = 1.6$, la corrispondente soluzione numerica.