Esame di Calcolo Numerico — 13 febbraio 2023

Corso di Laurea in Ingegneria Chimica
Tempo a disposizione: 2 ore. È consentito consultare appunti e testi (cartacei).

Esercizio 1 (15 punti) Consideriamo una griglia $x_1, \dots, x_{N+1}$ di punti ottenuti suddividendo in N sottointervalli uguali l'intervallo $[0, \pi/2]$. Vogliamo trovare la funzione della forma $\phi(x) = \alpha_1 + \alpha_2 x^2 + \alpha_3 x^4$ (cioè un polinomio di grado 4 contenente solo i termini corrispondenti a potenze pari della x) che meglio approssima la funzione $f(x) = \cos x$ sulla griglia data, nel senso dei minimi quadrati, cioè, che realizza

$$\min_{\alpha \in \mathbb{R}^3} \sum_{i=1}^{N+1} \Big(\phi(x_i) - f(x_i)\Big)^2. \tag{1}$$

1. Quali sono la matrice X e il vettore $\mathbf{y}$ che permettono di scrivere il problema di approssimazione (1) come $\min_\alpha \|X\alpha - \mathbf{y}\|_2^2$? Qual è il sistema lineare da risolvere per calcolare il vettore $\alpha = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} \in \mathbb{R}^3$ dei coefficienti della soluzione con il metodo delle equazioni normali?

2. Scrivere una `function [alpha, E] = approssimazione(N)` che:
   - prende in input il numero N;
   - restituisce un errore se $N < 2$;
   - al suo interno, calcola la matrice X, il vettore $\mathbf{y}$, e risolve il sistema lineare del metodo delle equazioni normali utilizzando l'operatore backslash \ di Matlab;
   - restituisce il vettore α e il valore dell'errore $E = \|X\alpha - \mathbf{y}\|_2$ raggiunto.

   Riportare sul foglio il codice Matlab della funzione.

3. Eseguire la funzione con $N = 2$; quali sono i valori di α ed E restituiti da Matlab? Qual è invece il valore di E ottenuto eseguendo la funzione con $N = 3$ e $N = 4$?

4. Per quale motivazione teorica il valore di E ottenuto nel primo caso è molto minore di quelli ottenuti negli altri due?

Esercizio 2 (15 punti) Vogliamo risolvere il problema ai valori iniziali

$$y'(t) = \frac{2}{t} y(t), \quad t \in [1, 2], \quad y(1) = 1 \tag{2}$$

utilizzando il metodo implicito

$$y_{n+1} = y_n + h\left(2f(t_n, y_n) - f(t_{n+1}, y_{n+1})\right). \tag{3}$$

1. Scrivere la forma assunta dalla relazione (3) per il problema (2), e risolverla nella varabile y_{n+1} in modo da trovare un'espressione esplicita che permetta di calcolare y_{n+1} ad ogni passo.

2. Scrivere una `function [t, Y] = metodo(N)` che, dato in input il numero di intervalli N, risolve il problema (2) con il metodo (3), usando la formula esplicita trovata al punto precedente. La funzione restituisce il vettore con gli estremi $t = [t_0, t_1, \dots, t_N]$ degli N intervalli (uguali) di discretizzazione, e il vettore delle approssimazioni corrispondenti $Y = [y_0, y_1, \dots, y_N]$. Riportare sul foglio il codice Matlab della funzione.

3. Per $N \in \{20, 40, 80\}$, calcolare l'errore globale di discretizzazione $E_N = \max_{n=1,2,\dots,N} |y(t_n) - y_n|$ tra la soluzione numerica calcolata e quella esatta `y = t.^2`. Cosa indicano i risultati ottenuti sull'ordine di convergenza del metodo?

4. (*) Qual è la funzione di stabilità $R(q)$ del metodo (3)?

Soluzioni

Esercizio 1 (15 punti)

1. Si ha

$$X = \begin{bmatrix} 1 & x_1^2 & x_1^2 \\ 1 & x_2^2 & x_2^2 \\ \vdots & \vdots & \vdots \\ 1 & x_{N+1}^2 & x_{N+1}^2 \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} \cos(x_1) \\ \cos(x_2) \\ \vdots \\ \cos(x_{N+1}) \end{bmatrix}.$$

   Difatti, espandendo il prodotto $X\left[\begin{smallmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{smallmatrix}\right] - \mathbf{y}$ si ottiene nella riga i-esima la differenza $\phi(x_i) - \cos(x_i)$, per ogni $i = 1, 2, \ldots, N+1$.

   Il metodo delle equazioni normali corrisponde a risolvere il sistema $(X^T X)\alpha = (X^T \mathbf{y})$.

2. Una soluzione possibile è la seguente.

```
function [alpha, E] = approssimazione(N);
if N < 2
    error("Numero di punti insufficienti");
end

x = linspace(0, pi/2, N+1);
X = zeros(N+1, 3);
for i = 1:N+1
    X(i,1) = 1;
    X(i,2) = x(i)^2;
    X(i,3) = x(i)^4;
end

y = zeros(N+1,1);
for i = 1:N+1
    y(i) = cos(x(i));
end

alpha = (X'*X) \ (X'*y);
E = norm(X*alpha - y);
```

3. Si ottengono i seguenti risultati.

```
>> [alpha, E] = approssimazione(2)
alpha =
    1.0000
   -0.4980
    0.0376
E =
   2.6595e-15
>> [alpha, E] = approssimazione(3); E
E =
   5.0556e-04
>> [alpha, E] = approssimazione(4); E
E =
   8.6315e-04
```

4. Nel caso $N = 2$, la matrice X ottenuta è quadrata, ed è possibile verificare anche che è invertibile (per esempio calcolandone $\det(X) \approx 2.8166$). Pertanto il sistema $X\alpha = \mathbf{y}$ ha soluzione unica, e (in

aritmetica esatta) si ha $E = 0$. Il motivo per cui non viene restituito esattamente zero è la presenza degli errori dovuti all'aritmetica di macchina.

Nei casi $N = 3, N = 4$ si ha un sistema lineare sovradeterminato che non ammette soluzione, pertanto anche in aritmetica esatta il valore di E che si ottiene è diverso da zero.

Esercizio 2 (15 punti)

1. Si ha
$$y_{n+1} = y_n + h\left(2\frac{2}{t_n}y_n - \frac{2}{t_{n+1}}y_{n+1}\right)$$
da cui risolvendo otteniamo
$$\left(1 + \frac{2h}{t_{n+1}}\right)y_{n+1} = \left(1 + \frac{4h}{t_n}\right)y_n$$
e quindi
$$y_{n+1} = \frac{1 + \frac{4h}{t_n}}{1 + \frac{2h}{t_{n+1}}}y_n.$$

2. Una possibile soluzione è la seguente.

```
function [t, Y] = metodo(N)
h = 1 / N;
t = 1:h:2;

Y = zeros(1, N+1);
Y(1) = 1;

for n = 1:N
    Y(n+1) = Y(n) * (1 + 4*h / t(n)) / (1 + 2*h / t(n+1));
end
```

3. Si ha

```
>> [t, Y] = metodo(20);
>> E20 = max(abs(Y - t.^2))
E20 =
    0.2609
>> [t, Y] = metodo(40);
>> E40 = max(abs(Y - t.^2))
E40 =
    0.1395
>> [t, Y] = metodo(80);
>> E80 = max(abs(Y - t.^2))
E80 =
    0.0723
>> E20 / E40, E40 / E80
ans =
    1.8696
ans =
    1.9302
```

L'errore si riduce di circa un fattore 2 ad ogni passo, e questo indica un metodo con ordine di convergenza 1.

4. Per calcolare la funzione di stabilità del metodo, dobbiamo applicarlo al problema test $y' = \lambda y$; in questo modo otteniamo
$$y_{n+1} = y_n + h\left(2\lambda y_n - \lambda y_{n+1}\right),$$

da cui

$$(1+h\lambda)y_{n+1} = (1+2h\lambda)y_n,$$

$$y_{n+1} = \frac{1+2h\lambda}{1+h\lambda} y_n.$$

La funzione di stabilità quindi si ottiene sostituendo $q = h\lambda$ e risulta

$$R(q) = \frac{1+2q}{1+q}.$$