

# Serie 4

## Equazioni Non Lineari

---

©2021 - Questo testo (compresi i quesiti ed il loro svolgimento) è coperto da diritto d'autore. Non può essere sfruttato a fini commerciali o di pubblicazione editoriale. Non possono essere ricavati lavori derivati. Ogni abuso sarà punito a termine di legge dal titolare del diritto. This text is licensed to the public under the Creative Commons Attribution-NonCommercial-NoDerivs2.5 License (<http://creativecommons.org/licenses/by-nc-nd/2.5/>)

---

### 1 Metodo di bisezione

Sia data una funzione  $f$  continua, con una radice nel punto  $x = \alpha$  all'interno dell'intervallo  $I = (a, b)$ , dove  $a$  e  $b$  sono tali che  $f(a) \cdot f(b) < 0$ .

Il metodo di bisezione calcola un'approssimazione  $\tilde{x}$  dello zero  $\alpha$  tramite il seguente algoritmo: posto  $a^{(0)} = a$  e  $b^{(0)} = b$ , per  $k = 0, \dots, n_{max}$

```
x(k) = (a(k) + b(k))/2
if (f(x(k)) == 0)
     $\tilde{x} = x^{(k)}$ 
    return
elseif (f(a(k))f(x(k)) < 0)
    a(k+1) = a(k)
    b(k+1) = x(k)
else
    a(k+1) = x(k)
    b(k+1) = b(k)
end
```

Il criterio d'arresto si basa sulla semiampiezza dell'intervallo: fissata una tolleranza  $\varepsilon_1$ , il metodo si arresta quando

$$\frac{1}{2}(b^k - a^k) < \varepsilon_1. \quad (1)$$

Di conseguenza il numero massimo di iterazioni che vengono eseguite è

$$n_{max} = \left\lceil \log_2 \left( \frac{b - a}{\varepsilon_1} \right) - 1 \right\rceil, \quad (2)$$

dove  $\lceil m \rceil$  indica il più piccolo numero intero maggiore di  $m$  (arrotondamento per eccesso).

Per rendere più efficiente lo schema si controlla anche il residuo, cioè si arresta il metodo non appena  $|f(x_k)| < \varepsilon_2$  (nella pratica spesso si considera un solo valore di tolleranza,  $\varepsilon_2 = \varepsilon_1$ ).

## Esercizio 1

Vogliamo risolvere il problema della ricerca degli zeri dell'equazione non lineare  $f(x) = 0$ , dove  $f$  è definita da:

$$f(x) = x^3 - (2 + e)x^2 + (2e + 1)x + (1 - e) - \cosh(x - 1), \quad x \in [0.5, 6.5] \quad (3)$$

1. Disegnare il grafico della funzione  $f$  nell'intervallo  $[0.5, 6.5]$  ed evidenziare le radici dell'equazione. (Suggerimento: utilizzare il comando `grid on`).
2. Si dica per quali radici il metodo di bisezione è applicabile a partire dal grafico della funzione.
3. Scrivere una function Matlab<sup>®</sup> `bisez.m` che implementi il metodo di bisezione. Tale funzione
  - riceve in input gli estremi `a` e `b` dell'intervallo di ricerca, la tolleranza `toll` richiesta sul residuo e la funzione `fun`.
  - restituisce in output il vettore `xvect` delle iterate e il numero `it` delle iterazioni effettivamente eseguite.
  - si utilizzi un criterio d'arresto basato sul residuo, accoppiato ad un controllo sul massimo numero di iterate, determinato come in (2), assumendo  $\varepsilon_1 = \varepsilon_2 = \text{toll}$ .

L'intestazione della funzione sarà ad esempio:

$$[\text{xvect}, \text{it}] = \text{bisez}(\text{a}, \text{b}, \text{toll}, \text{fun})$$

( Suggerimento: definire la funzione `fun` da passare alla funzione `bisez.m` tramite i comandi `inline` o `@`).

4. Calcolare le radici di  $f$  per le quali il metodo di bisezione risulta applicabile, utilizzando la funzione `bisez.m` con una tolleranza di  $10^{-12}$ .

## 2 Metodo di Newton

Il metodo di Newton per la ricerca dello zero  $\alpha$  di un'equazione non lineare  $f(x) = 0$  è un metodo iterativo che necessita della conoscenza della derivata prima  $f'(x)$  della funzione  $f(x)$ .

Dato  $x^{(0)}$ , la formula della generica iterazione  $k$  del metodo di Newton si esprime come:

$$x^{(k+1)} = x^{(k)} - \frac{f(x^{(k)})}{f'(x^{(k)})} \quad \text{se } f'(x^{(k)}) \neq 0, \quad k \geq 0. \quad (4)$$

Nel caso in cui  $\alpha$  sia uno zero semplice di  $f$  ( $f'(\alpha) \neq 0$ ) il metodo di Newton converge quadraticamente. Se invece la molteplicità è maggiore di uno, il metodo di Newton converge linearmente.

### Metodo di Newton modificato

Detta  $m$  la molteplicità dello zero  $\alpha$ , la convergenza quadratica può essere recuperata modificando la formula generale del metodo di Newton nel modo seguente:

$$x^{(k+1)} = x^{(k)} - m \frac{f(x^{(k)})}{f'(x^{(k)})} \quad \text{se } f'(x^{(k)}) \neq 0, \quad k \geq 0. \quad (5)$$

Il metodo (5) si chiama *Newton modificato*.

### Criterio d'arresto

Per arrestare il metodo di Newton si può utilizzare il criterio seguente, detto della differenza tra iterate successive: ad ogni iterazione  $k$  si valuta se la differenza tra due iterate successive è inferiore ad una certa soglia  $\varepsilon$ :

$$|x^{(k)} - x^{(k-1)}| < \varepsilon \quad (6)$$

### Esercizio 2

Si consideri l'equazione non lineare  $f(x) = 0$  dell'Esercizio 1. Sappiamo che essa ammette tre zeri nell'intervallo indicato.

1. Dal grafico di  $f$  e di  $f'$  discutere le proprietà di convergenza del metodo di Newton per tutti gli zeri, valutando l'opportunità di applicare il metodo di Newton modificato. Noto il valore esatto della radice  $\alpha_1 = 1$  si valuti la molteplicità della radice.
2. Scrivere una funzione Matlab<sup>®</sup> `newton.m` che implementi il metodo di Newton; la funzione
  - riceve in input la guess iniziale `x0`, la tolleranza `toll`, il numero massimo di iterazioni `nmax`, la funzione `fun`, la sua derivata `dfun` e la molteplicità della radice `mol`.
  - restituisce in output il vettore `xvect` delle iterate e il numero `it` di iterazioni effettivamente eseguite.

Si utilizzi un criterio d'arresto basato sul modulo della differenza tra due iterate successive. La function `newton.m` deve comportarsi come il metodo di Newton classico nel caso in cui non venga specificata la molteplicità della radice e come il metodo di Newton modificato se viene espressamente passata in ingresso la variabile `mol` (utilizzare a tal proposito la `function nargin`).

L'istestazione della funzione sarà ad esempio la seguente:

$$[xvect, it] = \text{newton}(x0, nmax, toll, fun, dfun, mol)$$

3. Risolvere il problema della ricerca degli zeri di  $f$  tramite il metodo di Newton, utilizzando la funzione `newton.m` con tolleranza  $10^{-6}$ . Si utilizzi per il calcolo di ogni radice un opportuno valore `x0`. Nel caso della radice  $\alpha_1 = 1$  si riporti su un grafico in scala semilogaritmica l'andamento degli errori in funzione del numero di iterazioni per il metodo di Newton standard e per quello modificato.

### Esercizio 3

Si consideri l'equazione non lineare  $f(x) = 0$ , in cui  $f$  è definita come:

$$f(x) = \arctan\left(7\left(x - \frac{\pi}{2}\right)\right) + \sin\left(\left(x - \frac{\pi}{2}\right)^3\right), \quad x \in [-1, 6]. \quad (7)$$

1. Si disegni il grafico della funzione  $f$  nell'intervallo  $[-1, 6]$ .
2. Si verifichi che l'unico zero  $\alpha$  della funzione  $f$  è semplice e si utilizzi la funzione `newton.m` implementata nel laboratorio precedente per calcolarlo. Si assuma una tolleranza pari a  $10^{-10}$  e guess iniziale  $x_0 = 1.5$ . Successivamente si ripeta tale calcolo partendo da  $x_0 = 4$ . Sapendo che  $\alpha = \pi/2$ , si calcolino gli errori assoluti nei due casi e si motivino i risultati ottenuti.
3. Verificate la possibilità di utilizzare il metodo di bisezione; si utilizzi la funzione `bisez.m` implementata nel laboratorio precedente sull'intervallo  $[a, b] = [-1, 6]$  con tolleranza  $\text{toll} = (b - a)/2^{31}$ . Si riporti l'errore ottenuto.
4. Si implementi una function `biseznewton.m` che combini il metodo di bisezione e di Newton. In particolare si adotti il metodo di bisezione per l'avvicinamento allo zero e successivamente il metodo di Newton per calcolare  $\alpha$ . La funzione richiesta
  - riceve in input gli estremi dell'intervallo di ricerca  $a$  e  $b$ , la tolleranza richiesta  $\text{toll}$ , il numero massimo di iterazioni  $\text{nmax\_b}$  per le iterazioni con il metodo di bisezione, il numero massimo di iterazioni  $\text{nmax\_n}$  per le iterazioni con il metodo di Newton, la funzione `fun` e la sua derivata `dfun`.
  - restituisce in output il vettore delle iterate `xvect` e il numero `it` di iterazioni effettivamente eseguite.

L'intestazione della funzione sarà, ad esempio, la seguente:

$$[\text{xvect}, \text{it}] = \text{biseznewton}(a, b, \text{nmax\_b}, \text{nmax\_n}, \text{toll}, \text{fun}, \text{dfun})$$

5. Si calcoli utilizzando la funzione `biseznewton` lo zero  $\alpha$  di  $f$ , impiegando 5 iterazioni del metodo di bisezione. Si mantengano i valori precedentemente utilizzati per tutti gli altri parametri di input.

### 3 Metodo delle iterazioni di punto fisso

Consideriamo il problema: data una funzione  $\phi : [a, b] \rightarrow \mathbb{R}$ , trovare  $\alpha \in [a, b]$  tale che:

$$\alpha = \phi(\alpha).$$

Se un tale  $\alpha$  esiste, viene detto *punto fisso* di  $\phi$ . Se la funzione di iterazione è sufficientemente regolare e  $|\phi'(\alpha)| < 1$  la successione definita da

$$x^{(k+1)} = \phi(x^{(k)}), \quad k \geq 0 \quad (8)$$

converge ad  $\alpha$  per ogni scelta del dato iniziale  $x^{(0)}$  in un intorno opportuno di  $\alpha$ . Tale successione soddisfa la condizione di convergenza

$$\lim_{k \rightarrow \infty} \frac{x^{(k+1)} - \alpha}{x^{(k)} - \alpha} = \phi'(\alpha), \quad (9)$$

ovvero (per  $k$  grande) l'errore al passo  $(k+1)$  è uguale all'errore al passo  $k$  moltiplicato per una costante  $\phi'(\alpha)$  il cui valore assoluto è minore di 1 (ordine di convergenza uguale almeno ad 1, o convergenza almeno *lineare*).

In maniera del tutto generale si può dimostrare che se le derivate  $i$ -esime della funzione  $\phi$  valutate in  $\alpha$  si annullano per  $i = 1, \dots, p$  con  $p \geq 1$  e  $\phi^{(p+1)}(\alpha) \neq 0$ , allora il metodo di punto fisso converge con ordine  $(p+1)$ , se  $x^{(0)}$  è “sufficientemente vicino ad  $\alpha$ , e vale:

$$\lim_{k \rightarrow \infty} \frac{x^{(k+1)} - \alpha}{(x^{(k)} - \alpha)^{p+1}} = \frac{1}{(p+1)!} \phi^{(p+1)}(\alpha). \quad (10)$$

Ad esempio, se  $\phi'(\alpha) = 0$ , (ma  $\phi''(\alpha) \neq 0$ ) il metodo di punto fisso è convergente di ordine 2.

Le iterazioni di punto fisso possono essere applicate all'approssimazione degli zeri di una funzione  $f(x)$ . In generale, la funzione di iterazione  $\phi$  deve essere scelta in modo tale che

$$\phi(\alpha) = \alpha \text{ ogni volta che } f(\alpha) = 0.$$

La scelta di  $\phi$  ovviamente non è unica. Si può ricorrere infatti a manipolazioni algebriche differenti di  $f$  per ottenere delle possibili funzioni di iterazione  $\phi$ . Ad esempio, ogni funzione della forma  $\phi(x) = x + F(f(x))$  è una funzione di iterazione ammissibile, purchè  $F$  sia una funzione continua tale che  $F(0) = 0$ . Inoltre, ricordiamo che il metodo di Newton può essere riletto come un algoritmo di iterazioni di punto fisso per la funzione di iterazione

$$\phi_N(x) = x - \frac{f(x)}{f'(x)};$$

se  $f'(\alpha) \neq 0$  (ovvero  $\alpha$  è zero semplice), si ha  $\phi'(\alpha) = 0$ , ovvero il metodo è di ordine 2.

Una semplice implementazione dell'algoritmo delle iterazioni di punto fisso è la seguente:

```
function [succ, it] = ptofis(x0, phi, nmax, toll)
%
% [succ, it] = ptofis(x0, phi, nmax, toll)
% Metodo di punto fisso x = phi(x)
%
```

```

% -----Parametri di ingresso:
% x0      Punto di partenza
% phi     Funzione di punto fisso (definita inline o anonima)
% nmax    Numero massimo di iterazioni
% toll    Tolleranza sul test d'arresto
%
% -----Parametri di uscita:
% succ    Vett. contenente tutte le iterate calcolate
%         (l'ultima componente e' la soluzione)
% it      Iterazioni effettuate
err       = 1 + toll;
it        = 0;
succ      = x0;
xv        = x0;
while (it < nmax && err > toll)
    xn     = phi(xv);
    err    = abs(xn - xv);
    succ   = [succ; xn];
    it     = it + 1;
    xv     = xn;
end
fprintf(' \n Numero di Iterazioni      : %d \n', it);
fprintf(' Punto fisso calcolato      : %12.13f \n', succ(end));

```

Spesso può essere interessante visualizzare le iterate successive dell'algoritmo di punto fisso; anziché visualizzare i valori generati  $x^{(k)}$  al variare dell'indice  $k$ , si preferisce costruire un grafico formato da una linea spezzata che congiunge i punti di coordinate:

$$\left(x^{(k)}, \phi(x^{(k)})\right) \longrightarrow \left(x^{(k+1)}, x^{(k+1)}\right) \longrightarrow \left(x^{(k+1)}, \phi(x^{(k+1)})\right) \dots, k \geq 0.$$

Il grafico così ottenuto consente di distinguere immediatamente i punti fissi attrattori dai punti fissi repulsori.

## Esercizio 4

Si consideri la funzione  $f(x) = \cos^2(2x) - x^2$ .

1. Si disegni la funzione e si individuino graficamente i punti in cui  $f(x) = 0$ .
2. Si consideri la funzione di iterazione seguente, dipendente da un parametro  $A \in \mathbb{R}$  (costante)

$$\phi(x) = x + A f(x).$$

Si verifichi teoricamente per quale intervallo di valori della costante  $A$  il metodo delle iterazioni punto fisso, qui usato per la ricerca degli zeri  $\alpha$  di  $f(x)$ , può convergere allo zero  $\alpha > 0$  per una scelta opportuna del dato iniziale.

3. Si utilizzi ora la `function` `ptofis.m` con  $A = 0.1$  e  $x^{(0)} = 0.1$  per ottenere un valore dello zero  $\alpha$ , scegliendo come tolleranza  $10^{-10}$ . Si usi il valore dello zero ottenuto per verificare numericamente il range di valori ammissibili per  $A$  ottenuto teoricamente al punto precedente. Si studi la sensibilità della convergenza del metodo al variare di  $A$  e  $x^{(0)}$ .

4. Stimare l'ordine di convergenza e il fattore di convergenza del metodo di punto fisso al variare di  $A$  utilizzando la funzione `stimap.m`, verificando l'affidabilità delle stime teoriche.
5. Fornire un valore di  $A$  tale da ottenere un metodo convergente del secondo ordine.
6. Ricordando che il metodo di Newton può essere riletto come metodo di punto fisso, implementarlo utilizzando la `function` `ptofis.m`. Determinare sperimentalmente gli intervalli in cui scegliere il dato iniziale in modo che il metodo converga allo zero  $\alpha$  positivo.