

# Relazione progetto di programmazione: calcolo degli zeri di una funzione

*Il progetto è stato realizzato da...*

*Altobel Giulia , Carapucci Rivo , Corso Nicole e Gianello Alessio*

## 1 Introduzione

Il nostro progetto tratta la risoluzione dell'equazione:

$$f(x) = 0 \tag{1}$$

con  $x \in \mathbb{R}$ .

Abbiamo implementato quattro metodi diversi per il calcolo degli zeri dell'equazione:

Metodo di bisezione;

Metodo di Newton;

Metodo delle corde;

Metodo delle secanti.

L'iterazione con l'utente risulterà essere fondamentale per il funzionamento del programma. Esso chiederà all'utente di:

1. Scegliere il metodo da usare per la ricerca;
2. Inserire:
  - Una funzione iniziale a scelta;
  - La derivata quando è necessario;
  - Gli estremi dell'intervallo di ricerca;
  - Eventuale/i nodo/i di partenza;

- La tolleranza desiderata;
- Massimo numero di iterazioni.

Una volta acquisiti gli input richiesti, il programma restituirà lo zero della funzione o una sua approssimazione o in caso contrario un messaggio di errore in relazione alla tipologia di problema incontrato. In seguito il programma permetterà all'utente di procedere con l'inserimento di un'altra funzione da risolvere con lo stesso metodo della precedente (usando la parola chiave *Yes o yes*) oppure di cambiare il metodo di risoluzione (usando la frase *cambia metodo*). Infine, per uscire dal programma, sarà sufficiente digitare la parola chiave *chiudi*. Affinché l'intervallo di ricerca sia accettato dal programma, è necessario che sia valido il seguente teorema:

**Teorema 1 (di Bolzano)** *Siano  $a$  e  $b$  due numeri reali tali che:*

- $f(a)f(b) < 0$ ;
- $f$  sia continua su  $[a, b]$

*allora esiste almeno un  $\bar{x} \in (a, b)$  tale che  $f(\bar{x}) = 0$ .*

Nel caso in cui esso non sia valido, il programma restituirà un messaggio di errore e non potrà essere utilizzato nessun metodo. Verrà chiesto quindi all'utente di inserire una nuova funzione o un nuovo intervallo.

## 2 Metodo di Bisezione

### 2.1 Procedimento:

Si procede dividendo l'intervallo in due parti uguali e calcolando il valore della funzione nel punto medio di ascissa  $m = \frac{(a+b)}{2}$ .

Se risulta  $f(m) = f(\frac{(a+b)}{2}) = 0$  allora  $m$  è la radice cercata; altrimenti tra i due intervalli  $[a, m]$  e  $(m, b]$  si sceglie quello ai cui estremi la funzione assume valori di segno opposto. Si ripete per questo nuovo intervallo la procedura di dimezzamento. Così continuando si ottiene una successione di intervalli  $[a_1, b_1], [a_2, b_2], \dots, [a_n, b_n]$  incapsulati, cioè ognuno incluso nel precedente.

Questi intervalli hanno come ampiezze:

$$b_n - a_n = \frac{(b - a)}{2^n} \quad (2)$$

per  $n = 1, 2, \dots$

I valori  $a_i$  sono valori approssimati per difetto della radice, i valori di  $b_i$

sono invece i valori della radice approssimati per eccesso. Gli  $a_n$  formano una successione crescente limitata ed i  $b_n$  formano una successione decrescente limitata. Le due successioni ammettono lo stesso limite che è la radice dell'equazione esaminata.

## 2.2 Criterio di arresto ed errore:

L'algoritmo viene arrestato quando  $f(m)$  è abbastanza vicino a 0 e/o si raggiunge il numero massimo di iterazioni consentite. Dunque come stima della soluzione  $\alpha$  alla fine avremo un certo  $m$ .

## 2.3 Analisi:

### *Costo computazionale:*

Per quanto riguarda il costo di questo algoritmo in termini di operazioni, considerando il fatto che viene eseguita una sola valutazione di funzione ad ogni iterazione, si può affermare che ha un costo di *andamento grossomodo logaritmico* all'allargarsi dell'intervallo di confidenza.

La velocità di convergenza del metodo di bisezione cresce linearmente, anche se solo localmente (cioè non vale per ogni intervallo).

Il costo in termini di memoria è anch'esso molto contenuto, si utilizzano infatti soltanto poche variabili.

### *Vantaggi:*

- Vale la seguente relazione:

$$\lim_{n \rightarrow \infty} |e_n| = 0 \quad (3)$$

quindi la convergenza del metodo è globale (converge sempre); è quindi un metodo molto robusto.

- Ha il notevole pregio di essere stabile in ogni occasione e quindi di garantire sempre la buona riuscita dell'operazione.
- Dal punto di vista del costo computazionale, poichè ad ogni iterazione si esegue una sola valutazione di funzione, il costo computazionale risulta essere basso.

### *Svantaggi:*

- L'efficienza del metodo di bisezione è scarsa e sono richieste ipotesi particolarmente restrittive;

- Se richiediamo

$$|e_n| \leq \textit{tolleranza} \quad (4)$$

otteniamo la seguente condizione per n:

$$n \geq \log_2\left(\frac{b-a}{\textit{toller}}\right) - 1. \quad (5)$$

Essendo  $\log_2 10 \approx 3.32$  servono in media più di tre bisezioni per migliorare di una cifra significativa l'accuratezza della radice, quindi la convergenza è lenta;

- Inoltre la riduzione dell'errore a ogni passaggio non è monotona, cioè non è detto che:

$$|e_{n+1}| < |e_n| \quad (6)$$

per ogni  $n = 1, 2, \dots$

- Il metodo funziona solo se la funzione ammette un solo zero e non può calcolare zeri complessi.

## 3 Metodo di Newton o delle tangenti

### 3.1 Procedimento:

Il metodo consiste nel sostituire alla curva  $y=f(x)$  la tangente alla curva stessa, partendo da un qualsiasi punto: considerando l'equazione del fascio di rette:

$$y - f(x_0) = m(x - x_0) \quad (7)$$

posto  $y=0$  e  $m=f'(x_0)$ ,  $x_1$  sarà:

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} \quad (8)$$

Generalizzando all'iterazione k:  $x_{k-1} = x_k - (f(x_k))/f'(x_k)$ , che è la relazione di ricorrenza del metodo (che permette di determinare successive approssimazioni della radice dell'equazione  $y = f(x) = 0$ ).

### 3.2 Criterio di arresto ed errore:

L'algoritmo viene arrestato quando viene raggiunto il massimo numero di iterazioni e/o quando l'errore risulta minore della tolleranza prestabilita. L'errore è dato dalla seguente relazione:

$$e_n = |x_1 - x| \quad (9)$$

Dove con  $x$  si intende l'iterazione precedente e con  $x_1$  si intende l'iterazione corrente.

### 3.3 Analisi:

#### *Costo computazionale:*

Il metodo di Newton ha un costo leggermente più alto del precedente: richiede infatti due valutazioni di funzione ad ogni iterazione. In questo caso il costo massimo della determinazione dello zero non è determinabile a priori, e dipende dalla scelta del numero massimo di iterazioni effettuata dall'utente. Il costo in termini di memoria, come nel caso del metodo di bisezione è anch'esso molto contenuto e si utilizzano soltanto poche variabili.

#### *Vantaggi:*

Con le ipotesi poste, si dimostra che la successione delle  $x_n$  converge alla radice piuttosto rapidamente. La convergenza è quadratica, cioè il numero di cifre significative approssimativamente raddoppia ad ogni iterazione.

#### *Svantaggi:*

- Il metodo è applicabile solo a funzioni derivabili con derivata prima diversa da zero nei punti della successione;
- La convergenza non è garantita, in particolare quando  $f'(x)$  varia notevolmente in prossimità dello zero. Inoltre, il metodo assume che  $f'(x)$  sia disponibile direttamente per un dato  $x$ . Nei casi in cui questo non si verifichi e risultasse necessario calcolare la derivata attraverso una differenza finita, è consigliabile usare il metodo delle secanti;
- Il numero di iterazioni dipende principalmente dalla posizione del punto iniziale rispetto allo zero. In base alla funzione analizzata e al valore di  $x_0$  assunto come prima iterazione il metodo può convergere o meno. Il metodo risulta meno solido rispetto al metodo di bisezione.

## 4 Metodo delle corde

### 4.1 Procedimento:

Il metodo consiste nel costruire una successione di punti con il seguente criterio: assegnato un punto iniziale  $x_0$ , per ogni  $n \geq 0$ , il punto  $x_{n+1}$

sia lo zero della retta passante per il punto  $(x_n, f(x_n))$  e di coefficiente angolare  $m = \frac{f(a)-f(x_n)}{a-x_n}$  ovvero quello della retta passante per i punti  $(x_n, f(x_n))$  e  $(a, f(a))$ .

Iterando il procedimento del calcolo dell'intersezione delle varie rette con l'asse delle ascisse, si ottiene la relazione di ricorrenza:

$$x_{n+1} = a - f(a) \frac{(x_n - a)}{f(x_n) - f(a)} \quad (10)$$

## 4.2 Criterio di arresto ed errore:

L'algoritmo viene arrestato quando viene raggiunto il massimo numero di iterazioni e/o quando l'errore risulta minore della tolleranza prestabilita. L'errore è dato dalla seguente relazione:

$$e_n = |x_1 - x| \quad (11)$$

Dove con  $x$  si intende l'iterazione corrente e con  $x_1$  si intende l'iterazione successiva.

## 4.3 Analisi:

### *Costo computazionale:*

Questo metodo ad ogni ciclo iterativo svolge soltanto una valutazione di funzione, il costo per iterata è quindi molto basso.

Bisogna inoltre notare come la convergenza del metodo sia molto legata al tipo di funzione e alla scelta del punto di partenza;

se questo non è abbastanza vicino alla radice oppure la funzione non è abbastanza regolare, il metodo può convergere molto lentamente.

### *Vantaggi:*

Il metodo presenta un basso costo computazionale per iterazione (si valuta solo una volta  $f'(x)$  e una sola volta per iterazione  $f(x)$ ), ma si può dimostrare che l'ordine di convergenza è solo lineare.

### *Svantaggi:*

- Il metodo potrebbe convergere molto lentamente (vedi *costo computazionale 4.3*)

- Il metodo delle corde converge linearmente se, detta  $\alpha$  la soluzione corretta, vale:

$$0 < \frac{f'(\alpha)}{m} < 2 \quad (12)$$

In altri termini,  $m$  e  $f'(\alpha)$  devono avere lo stesso segno e l'intervallo  $[a, b]$  deve soddisfare la condizione:

$$b - a < 2 \frac{f(b) - f(a)}{f'(\alpha)} \quad (13)$$

Negli altri casi il metodo potrebbe non convergere affatto.

## 5 Metodo delle secanti

### 5.1 Procedimento

Il metodo consiste nel costruire una successione di punti con il seguente criterio: assegnati due punti iniziali  $x_0$  e  $x_1$ , per ogni  $n \geq 1$ , il punto  $x_{n+1}$  sia lo zero della retta passante per i punti  $(x_{n-1}, f(x_{n-1}))$ ,  $(x_n, f(x_n))$ . Si ottiene:

$$x_{n+1} = x_n - \frac{(x_n - x_{n-1})}{(f(x_n) - f(x_{n-1}))} f(x_n) \quad (14)$$

Rispetto al metodo delle corde, quello delle secanti richiede un punto iniziale in più e ad ogni passo il calcolo del rapporto che compare nella formula.

### 5.2 Criterio di arresto ed errore:

L'algoritmo viene arrestato quando viene raggiunto il massimo numero di iterazioni e/o quando l'errore risulta minore della tolleranza prestabilita. L'errore è dato dalla seguente relazione:

$$e_n = |x_1 - x_0| \quad (15)$$

Dove con  $x_0$  si intende l'iterazione corrente e con  $x_1$  si intende l'iterazione successiva.

### 5.3 Analisi:

*Costo computazionale:*

Questo metodo ad ogni ciclo iterativo svolge soltanto una valutazione di funzione, il costo per iterata è quindi veramente molto basso. Inoltre la

convergenza è locale, cioè dipende dalla scelta dei punti iniziali  $x_0$  e  $x_1$ ; il guadagno è però una maggiore velocità di convergenza, che risulta superlineare.

*Vantaggi:*

- Si può usare quando non si conosce la derivata di  $f(x)$  o quando  $f(x)$  è nota per punti;
- Ad ogni passo richiede una sola valutazione di funzione, quindi il costo computazionale per iterata è veramente molto basso.

*Svantaggi:*

- Servono 2 approssimazioni iniziali  $x_0$  e  $x_1$ ;
- La scelta di  $x_0$  e  $x_1$  deve essere "accurata".

## 6 Bibliografia

Di seguito vengono citate le fonti utilizzate nella nostra relazione:

- Dispensa-ricercaZeri (fornita dal professore);
- Le seguenti pagine di Wikipedia:
  - Metodo della bisezione;
  - Metodo delle tangenti;
  - Metodo delle corde;
  - Metodo delle secanti.
- Appunti dal corso di Calcolo numerico I.