

Esercitazione 4

Istruzioni trattate: l'operatore \$, xgrid, legend, plot2d, l'operatore : .

In questa esercitazione si considera il problema di approssimare lo zero, α , della funzione f definita per $x > 0$ da:

$$f(x) = x + \log x$$

Nella prima parte si confrontano le approssimazioni ottenute realizzando i metodi ad un punto definiti dalle funzioni:

$$h_2 = e^{-x} \quad \text{e} \quad h_3(x) = \frac{x + e^{-x}}{2}$$

Nella seconda parte si confrontano queste approssimazioni con quella ottenuta realizzando il *metodo di bisezione* applicato alla funzione f .

Prima parte

La definizione che segue è una realizzazione del *metodo ad un punto* definito dalla funzione h .¹

```
function x = MetodoUnPunto(h,x0,N)
//
// x: colonna di numeri reali. Le componenti di questo vettore sono i primi
// N + 1 elementi della successione generata dal metodo definito
// da h a partire da x0.
//
x = x0;
k = 0;
while k < N,
    x($+1) = h(x($));
    k = k + 1;
end;
endfunction
```

Si osservi che la condizione che si è scelto di utilizzare per realizzare il criterio di arresto (arresta la costruzione della successione alla $N - \text{esima}$ iterazione) è certamente *calcolabile* ed *efficace* ma, quando verificata, *non* fornisce alcuna informazione sull'accuratezza dell'approssimazione trovata.

Come primo metodo per approssimare α , si considera quello definito dalla funzione h_2 . Lo studio *analitico* del metodo ha consentito di scoprire che h_2 ha *un solo punto unito* (lo zero α) separato dall'intervallo $[\frac{1}{2}, 1]$, che il metodo è *utilizzabile* ed ha *ordine di convergenza uno* ad α e che la successione generata a partire da $x_0 = \frac{1}{2}$ converge ad α (più precisamente: ciascun elemento della successione dista da α *meno* del precedente e si trova "dalla parte opposta" di α rispetto al precedente). Vediamo che informazioni si possono ottenere, invece, per via *grafica*.

Definiamo la funzione h_2 realizzazione di h_2 :

```
function y = h2(x)
//
// x,y: matrici ad elementi reali di uguale dimensione;
//
y = exp(-x);
endfunction
```

Riportiamo su uno stesso piano cartesiano i grafici, sull'intervallo $[0, 1]$, delle funzioni $y = x$ e $y = h_2(x)$:²

¹Nella definizione compare l'operatore \$. Sia v un vettore di n elementi. Le componenti di v si indicano con $v(1), \dots, v(n)$. L'ultima componente si può indicare anche con $v(\$)$. L'assegnamento $v(\$+1) = 2 * v(\$)$ ha l'effetto di creare una nuova componente di v (la $n + 1 - \text{esima}$) ed assegnarle il valore richiesto $2v(n)$.

²Più correttamente: un'approssimazione di tali grafici.

```
-->x = linspace(0,1,301)';
-->plot2d(x, [x,h2(x)]);
-->xgrid(3); xlabel('x'); legend('x','h2(x)', 'in_lower_right');
```

Si ottiene il disegno riportato a sinistra in Figura 1. Il valore 3 dell'argomento del comando `xgrid` cambia il colore con cui viene disegnata la griglia da nero (colore predefinito) a verde. L'opzione `'in_lower_right'` presente nel comando `legend` fa posizionare la legenda nell'angolo *in basso a destra* della figura.

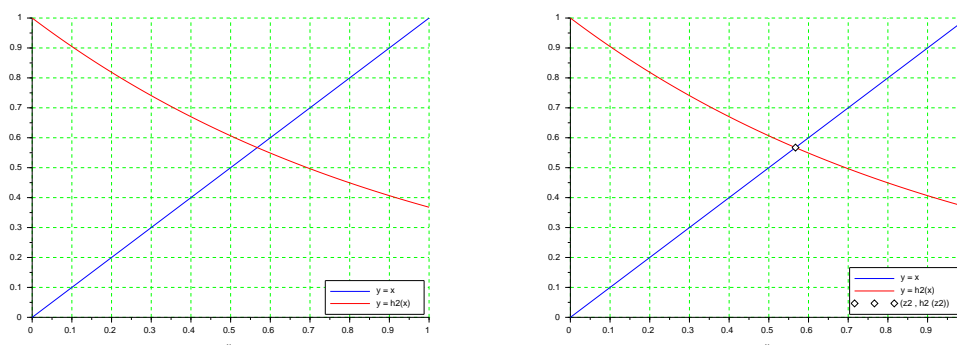


Figura 1: Studio grafico per il metodo definito da h_2 .

Dal disegno si deduce che, nell'intervallo $[0,1]$: (a) i grafici di $y = h_2(x)$ ed $y = x$ si intersecano in un solo punto, di ascissa maggiore di 0.5; (b) in tutti i punti del grafico di h_2 con ascissa maggiore o uguale a 0.5, la retta tangente ha pendenza negativa e maggiore di meno uno e (c) il punto unito è più vicino a 0.5 che a 1. Ne segue che: (a) la funzione h_2 ha un solo punto unito in $[0,1]$, separato dall'intervallo $[\frac{1}{2},1]$; (b) l'intervallo $[\frac{1}{2},1]$ verifica le ipotesi del Teorema di convergenza dunque il metodo definito da h_2 è utilizzabile per l'approssimazione e (c) la successione generata dal metodo a partire da $x_0 = \frac{1}{2}$ risulta convergente al punto unito.

Si osservi che dal disegno si possono dedurre informazioni relative al *solo* intervallo $[0,1]$. In particolare, il disegno *non fornisce informazioni sull'esistenza di altri punti uniti al di fuori di tale intervallo*. La scelta dell'intervallo da considerare per lo studio è *responsabilità dell'utilizzatore*.

Con le informazioni ottenute, analiticamente o graficamente, possiamo utilizzare la procedura `MetodoUnPunto` per ottenere un'approssimazione del punto unito α :

```
-->x0 = 1/2; N = 20;
-->z2 = MetodoUnPunto(h2,x0,N);
-->plot2d(z2($),h2(z2($)),style = -5);
-->legend('x','h3(x)', '(z2 , h2(z2))', 'in_lower_right');
```

Si ottiene, nella finestra grafica, il disegno riportato a destra in Figura 1. Il valore negativo per l'opzione `style` del comando `plot2d` fa disegnare *un simbolo* con centro nelle coordinate del punto richiesto (in questo caso il simbolo \diamond con centro in $(z2(\$),h2(z2(\$)))$). Si osserva che l'ultimo elemento della successione generata approssima il punto unito con sufficiente "accuratezza grafica."

Si considera adesso il metodo definito dalla funzione h_3 . Lo studio *analitico* del metodo ha consentito di scoprire che h_3 ha *un solo punto unito* (lo zero α) separato dall'intervallo $[\frac{1}{2},1]$, che il metodo è *utilizzabile* ed ha *ordine di convergenza uno ad α* e che la successione generata a partire da $x_0 = \frac{1}{2}$ converge ad α (più precisamente: la successione è *monotona crescente*). Anche in questo caso vediamo che informazioni si possono ottenere per via *grafica*.

Si consideri la seguente realizzazione della funzione h_3 :

```
function y = h3(x)
//
// x,y: matrici ad elementi reali di uguale dimensione;
//
y = (x + exp(-x)) / 2;
endfunction
```

Procedendo come per il metodo precedente si ottiene inizialmente il disegno riportato a sinistra nella Figura 2.

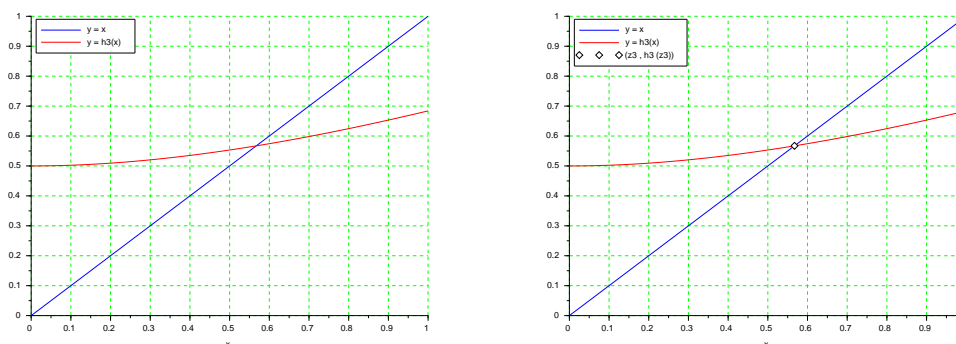


Figura 2: Studio grafico per il metodo definito da h_3 .

Dal disegno si deduce che, *nell'intervallo* $[0, 1]$: (a) i grafici di $y = h_3(x)$ ed $y = x$ si intersecano in un solo punto, di ascissa maggiore di 0.5; (b) in tutti i punti del grafico di h_3 la retta tangente ha pendenza positiva e minore di uno e (c) il punto unito è più vicino a 0.5 che a 1. Ne segue che: (a) la funzione h_3 ha un solo punto unito in $[0, 1]$, separato dall'intervallo $[\frac{1}{2}, 1]$; (b) l'intervallo $[\frac{1}{2}, 1]$ verifica le ipotesi del Teorema di convergenza dunque il metodo definito da h_3 è utilizzabile per l'approssimazione e (c) la successione generata dal metodo a partire da $x_0 = \frac{1}{2}$ risulta convergente al punto unito.

Come già osservato, dal disegno si possono dedurre informazioni relative al *solo* intervallo $[0, 1]$.

Con le informazioni ottenute, analiticamente o graficamente, possiamo utilizzare la procedura `MetodoUnPunto` per ottenere una nuova approssimazione del punto unito α e poi il disegno riportato a destra in Figura 2. Anche in questo caso si osserva che l'ultimo elemento della successione generata approssima il punto unito con sufficiente "accuratezza grafica."

Per confrontare graficamente l'andamento delle porzioni di successione determinate dai due metodi, se ne tracciano sullo stesso piano cartesiano i grafici con le istruzioni:³

```
-->clf(); plot2d([0:N]', [z2,z3], style = [2,5]);
-->xgrid(); xlabel('k'); legend('z2(k)', 'z3(k)');
```

Si ottiene il disegno riportato in Figura 3.⁴

Il disegno conferma quanto già noto dallo studio analitico riguardo al comportamento qualitativo delle successioni.

Un grafico più significativo si ottiene confrontando gli elementi delle successioni con α . Per poter effettuare il confronto è però necessario utilizzare *un'approssimazione* di α . Come già osservato, una scelta ragionevole è di approssimare α con $z3(\$)$. Si consideri allora il disegno riportato a sinistra nella Figura 4 generato dai comandi seguenti:

```
-->a = z3($);
```

³Nel secondo comando compare l'operatore `:` (colon). Se n ed m sono due numeri interi con $n < m$ allora $[n:m]$ è la *riga* di componenti $n, n + 1, \dots, m - 1, m$.

⁴Il grafico della porzione di successione è costituito dai soli *vertici* delle spezzate. I lati sono mantenuti per chiarezza grafica.

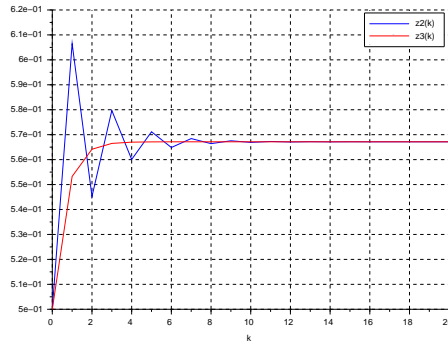


Figura 3: I grafici delle porzioni delle successioni ottenute.

```
-->clf(); plot2d([0:N]', [abs(z2 - a), abs(z3 - a)], style = [2,5]);
-->xgrid(); xlabel('k'); legend(' | z2(k) - a |', ' | z3(k) - a |');
```

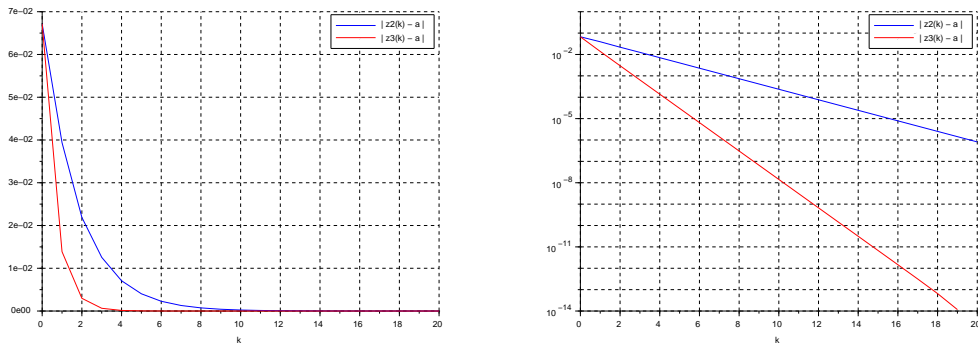


Figura 4: A sinistra: distanza da α delle successioni. A destra: distanza in scala *semilogaritmica*.

Il disegno evidenzia, almeno per k da 1 a 9, che a parità di indice k la distanza da α dell'elemento della successione generata dal metodo definito da h_3 è *minore* di quella della successione generata dal metodo definito da h_2 . Il disegno a destra nella stessa figura riporta gli stessi grafici ma in scala *semilogaritmica*⁵ e mostra che anche per k da 10 a 19 a distanza da α dell'elemento della successione generata dal metodo definito da h_3 è *minore* di quella della successione generata dal metodo definito da h_2 .

Questi disegni suggeriscono che la successione generata dal metodo definito da h_3 converga ad α *più rapidamente* di quella generata dal metodo definito da h_2 .

Seconda parte

Si considera infine il *metodo di bisezione* per ottenere un'altra approssimazione dello zero α . Si utilizza la seguente modifica della procedura che realizza il metodo di bisezione:

```
function x = Bisezione(f,a,b,N)
//
// Applica il metodo di bisezione alla funzione f a partire dall'intervallo
// [a,b]. La costruzione della successione si arresta dopo N iterazioni
// oppure quando si è trovato uno zero della funzione f.
//
// La funzione restituisce la colonna x che contiene gli elementi della successione
```

⁵Il comando utilizzato per generare il grafico è: `plot2d('nl',[0:N]', [abs(z2 - a), abs(z3 - a)], style = [2,5]);`

```

// che la procedura ha determinato.
if f(a)*f(b) > 0 then
    error('la funzione non assume valori di segno opposto agli estremi');
else
    x = (a+b)/2;
    k = 0;
    while (k < N & f(x($)) ~= 0),
        if sign(f(a)) ~= sign(f(x($))) then b = x($); else a = x($); end;
        x($+1) = (a+b)/2;
        k = k + 1;
    end;
end;
endfunction

```

La definizione seguente introduce la funzione f da utilizzare per approssimare i valori di f :

```

function y = f(x)
//
// x: matrice ad elementi numeri reali positivi.
// y: matrice di numeri reali della stessa dimensione di x.
//
y = x + log(x);
endfunction

```

Con le istruzioni:

```

-->a = 1/2; b = 1; N = 20;
-->zB = Bisezione(f,a,b,N);
-->clf(); plot2d('nl',[0:N]',[abs(z2-a),abs(z3-a),abs(zB-a)], style = [2,5,1]);
-->xgrid(3); xlabel('x'); legend('| z2(k) - a |','| z3(k) - a |','| zB(k) - a |');

```

si ottiene il grafico riportato in Figura 5. Il disegno mostra come al crescere di k la distanza da α dell'elemento della successione generata dal metodo di bisezione tende a zero con andamento *non monotono decrescente*. Ad esempio, l'elemento ottenuto con $k = 2$ è più vicino ad α del successivo. Inoltre, la successione generata dal metodo definito da h_3 appare convergere ad α *più rapidamente* anche di quella generata dal metodo di bisezione.

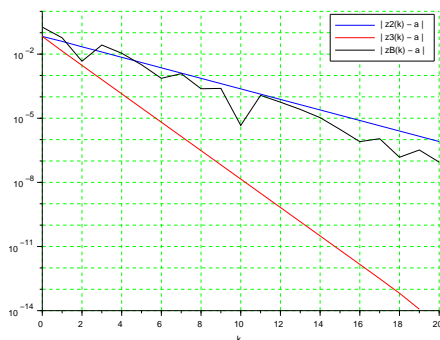


Figura 5: Confronto con la successione ottenuta con il metodo di bisezione.

Esercizi

1. Siano $t \in (0, 1)$ e $d_0 > 0$ numeri reali. Posto:

$$d_k = t^k d_0$$

utilizzare il comando `plot2d` per disegnare il grafico della successione d_k , per $k = 0, \dots, 20$, in scala *semilogaritmica* e spiegare analiticamente perché si ottiene un segmento di retta. Rivedere, alla luce di quanto ottenuto, i grafici riportati nella parte destra della Figura 4.

2. La definizione della funzione `f` data nella seconda parte *non* include una verifica della coerenza dell'argomento con quanto richiesto nel commento (ovvero: non si verifica che le componenti di `x` siano numeri reali *positivi*). Verificare che *Scilab* segnala un errore in risposta alla richiesta di calcolare `f(0)`, ma *non* segnala alcun errore in risposta alla richiesta di calcolare `f(-3)`.

In *Scilab* la *funzione predefinita* `log` corrisponde alla funzione *logaritmo complesso*, definita per ogni numero complesso $z \neq 0$ da:

$$\log(z) = \log(|z|) + i \arg(z)$$

dove $\arg(z)$ è l'*argomento principale* di z (l'unico argomento di z in $(-\pi, \pi]$).⁶ Se si vuole che *Scilab* segnali la presenza di componenti negative nell'argomento, è *necessario* includere una verifica nella definizione della funzione.

Particolare cautela occorre, in *Scilab*, quando si opera su numeri complessi con i comandi di stampa e disegno. Ad esempio, eseguire i comandi:

```
-->disp(f(-3), 'f(-3) =', 'Il valore di f in -3 è un numero complesso:');  
  
-->printf('\n Il valore di f in -3 è un numero complesso:\n\n' + ...  
        ' f(-3) =\n\n %8.7f', f(-3));  
  
-->clf(); x = linspace(-1,1,100); xN = x(1:50); xP = x(51:100);  
  
-->plot(xN,f(xN),xP,f(xP)); xgrid(); xlabel('x'); ylabel('f(x)');
```

⁶Per approfondire, vedere, ad esempio: https://en.wikipedia.org/wiki/Logarithm#Complex_logarithm.