

Metodi Matematici per l'Ingegneria

(A.A. 2019-2020)

Metodi Numerici
Equazioni non lineari

Docente Domenico Vitulano

Email: domenico.vitulano@sbai.uniroma1.it

Ufficio: Via A. Scarpa,

Pal. B, I piano, Stanza n. 11

Tel. 06 49766555

Ricevimento: consultare la pagina web dedicata al corso

Testi consigliati:

Calcolo Numerico, L. Gori, Ed. Kappa, 2006

Esercizi di Calcolo Numerico, L. Gori-M.L. Lo Cascio, F. Pitolli, Ed. Kappa, 2007

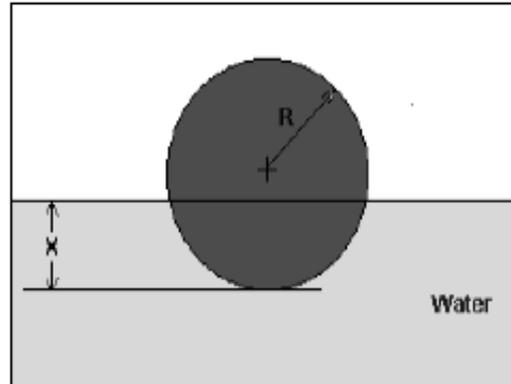
Il materiale didattico è disponibile sul sito

<https://www.sbai.uniroma1.it/vitulano-domenico/analisi-numerica/2019-2020>

Equazioni non lineari

Problema 1

Si vuole determinare la parte sommersa di una boa sferica di raggio $R = 0.055m$ e densità $\rho_B = 0.6Kg/m^3$



Indicando con F_ρ la spinta idrostatica, con M_B la massa della boa sferica, g l'accelerazione di gravità e $\rho_w (= 1Kg/m^3)$ la densità dell'acqua, si ha

$$M_B g = F_\rho$$

e quindi

$$\frac{4}{3}\pi R^3 \rho_B g = \pi x^2 \left(R - \frac{x}{3} \right) \rho_w g$$

da cui

$$x^3 \rho_w - 3x^2 R \rho_w + 4R^3 \rho_B = 0.$$

Per calcolare il valore di x è necessario risolvere un' **equazione non lineare**

L'equazione da risolvere è la seguente:

$$x^3 - 0.165x^2 + 3.993 \cdot 10^{-4} = 0.$$

Problema 2

La **crescita di una popolazione** può essere modellata, su un periodo di tempo piccolo, assumendo che la popolazione abbia un tasso di crescita proporzionale al numero di individui presenti in ciascun istante. Se $N(t)$ indica il numero di individui al tempo t e λ è il fattore di crescita della popolazione, allora $N(t)$ soddisfa l'equazione differenziale

$$\frac{dN(t)}{dt} = \lambda N(t),$$

la cui soluzione è $N(t) = N_0 e^{\lambda t}$, dove N_0 indica la popolazione all'istante di tempo iniziale, cioè $N_0 = N(t_0)$.

Questo modello è valido solo quando la popolazione è isolata e non c'è immigrazione dall'esterno. **Se si suppone che ci sia una immigrazione** a un tasso costante ν , il modello differenziale diventa

$$\frac{dN(t)}{dt} = \lambda N(t) + \nu$$

la cui soluzione è $N(t) = N_0 e^{\lambda t} + \frac{\nu}{\lambda} (e^{\lambda t} - 1)$.

Supponendo che la popolazione iniziale sia di 1000'000 di individui, che la comunità cresca di 435'000 immigrati il primo anno e che 1'564'000 individui siano presenti alla fine del primo anno, determinare il tasso di crescita λ della popolazione.

E' necessario risolvere l'**equazione non lineare**

$$N|_{t=1 \text{ anno}} = N_0 e^\lambda + \frac{\nu}{\lambda} (e^\lambda - 1)$$

nell'incognita λ , con $N|_{t=1 \text{ anno}} = 1'564'000$, $N_0 = 1'000'000$, $\nu = 435'000$.

$$\Rightarrow f(\lambda) = e^\lambda + \frac{0.435}{\lambda} (e^\lambda - 1) - 1.564 = 0$$

Equazioni non lineari

Un' **equazione non lineare** è un'equazione del tipo

$$f(x) = 0$$

Le **soluzioni** ξ dell'equazione, cioè quei valori tali che

$$f(\xi) = 0,$$

vengono chiamati **radici** dell'equazione non lineare o **zeri** della funzione f .

Ci limiteremo al caso di **radici reali**.

Separazione delle radici

Prima di utilizzare un metodo numerico bisogna sapere:

- **quante** sono le radici (reali);
- **dove** si trovano approssimativamente;
- se ci sono delle **simmetrie**.

Per rispondere a queste domande si può ricorrere alla **tabulazione** o al **grafico** della funzione f .

Separazione grafica delle radici

Calcolo tasso di crescita (problema 2)

$$f(\lambda) = e^\lambda + \frac{0.435}{\lambda}(e^\lambda - 1) - 1.564 = 0$$

1) Studio (classico) della funzione

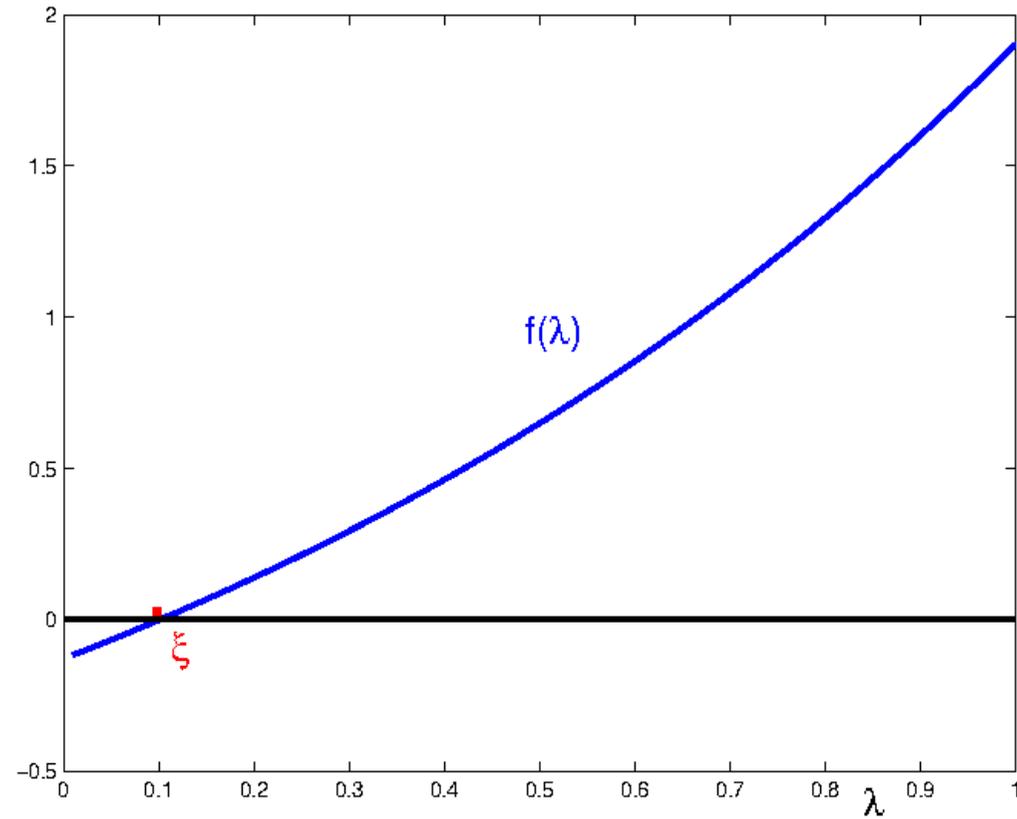
2) Separazione grafica: si traccia il **grafico della funzione** e si individuano gli intervalli in cui la funzione **interseca l'asse delle ascisse**.

$$f(\lambda) = e^\lambda + \frac{0.435}{\lambda}(e^\lambda - 1) - 1.564 = 0$$

In questo caso, la funzione f risulta definita e continua in $\mathbf{R} - \{0\}$. Inoltre, da uno studio preliminare di f nel semiasse positivo, si ha

- $\lim_{\lambda \rightarrow 0} f(\lambda) < 0$
- $\lim_{\lambda \rightarrow +\infty} f(\lambda) = +\infty$
- $f'(\lambda) = e^\lambda \left(1 + 0.435 \frac{\lambda-1}{\lambda^2}\right) + \frac{0.435}{\lambda^2} > 0$, cioè la funzione è **monotona crescente**

e quindi si può concludere che f ha un unico zero ξ nel semiasse positivo.



Osservando il grafico tracciato è possibile individuare un intorno di ξ ;
per esempio

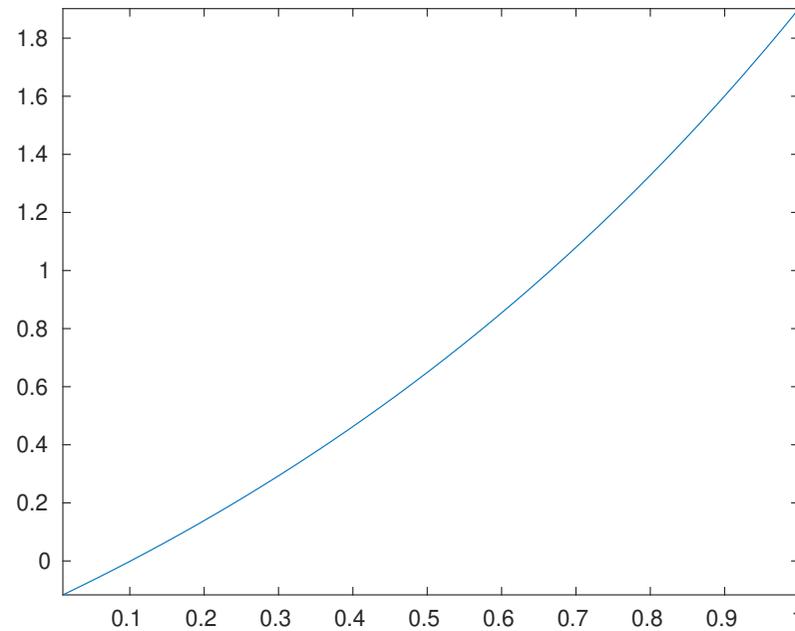
Intervallo di separazione: $I = [a, b] = [0.05, 0.15]$

$$\Rightarrow f(a) = -0.0667, f(b) = 0.0672 \Rightarrow f(a)f(b) < 0$$

Matlab

Grafico in Matlab:

```
fplot( @(lambda) exp(lambda) +  
+0.435./lambda.* (exp(lambda) - 1) - 1.564, [0.01, 1.00])
```



```
clear
```

```
lambda = [0.01:0.01:1.00];
```

```
f = [exp(lambda) + 0.435./lambda .* (exp(lambda)-1) - 1.564];
```

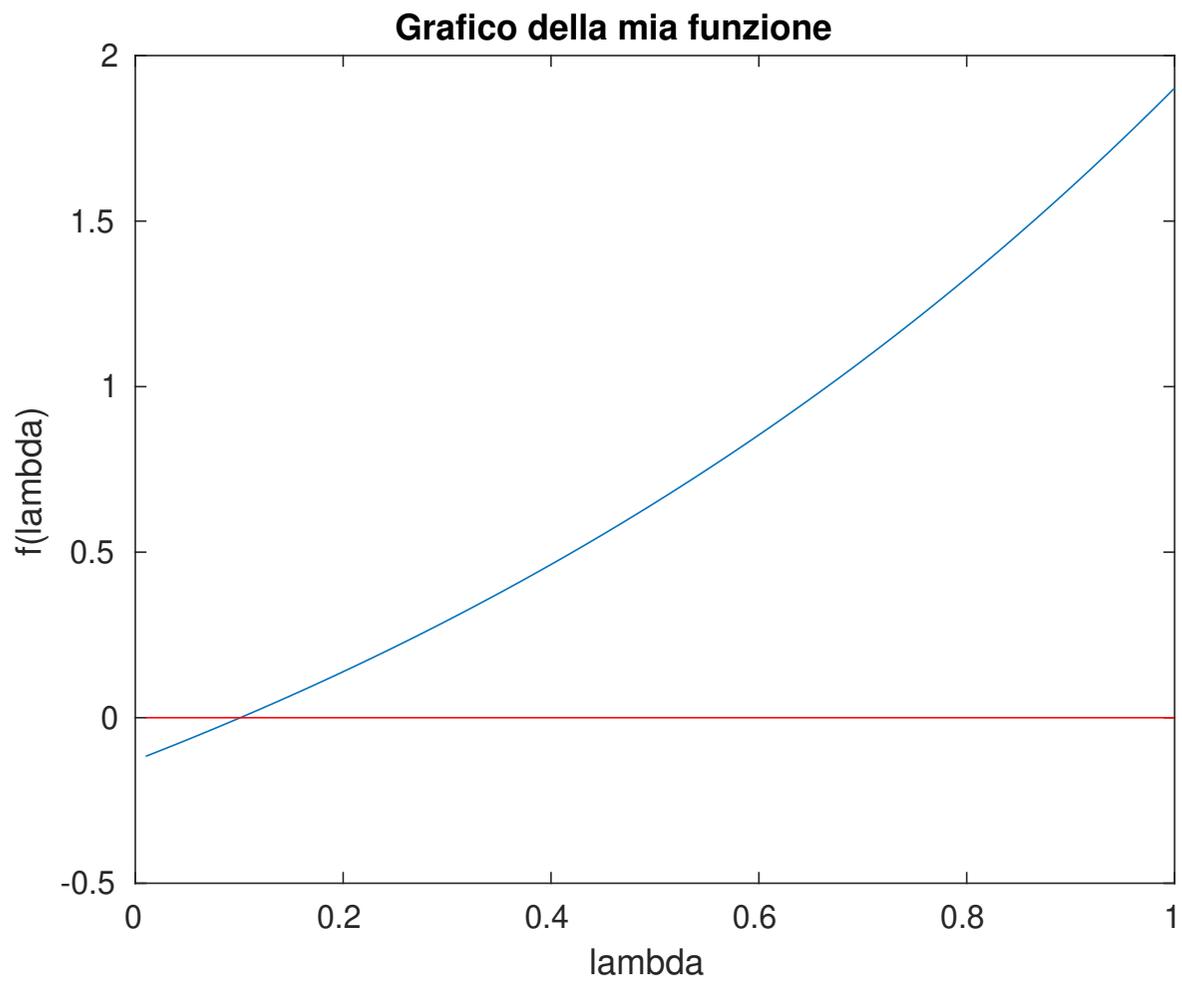
```
figure, plot(lambda,f)
```

```
hold on
```

```
plot(lambda,zeros(1,length(f)),'r')
```

```
title('Grafico della mia funzione')
```

```
ylabel('f(lambda)'), xlabel('lambda')
```



Problema 2: separazione delle radici - tabulazione

$$f(\lambda) = e^\lambda + \frac{0.435}{\lambda}(e^\lambda - 1) - 1.564 = 0$$

Si valuta la funzione in corrispondenza di valori equidistanti della variabile λ in un certo intervallo e si osserva il segno dei valori ottenuti:

λ	$f(\lambda)$
0.10	-0.00133558829528
0.12	0.02567293855461
0.14	0.05319595959218
0.16	0.08124355150079
0.18	0.10982599066618
0.20	0.13895375715854

Intervallo di separazione: $I = [a, b] = [0.10, 0.12]$

$$\Rightarrow f(a) = -0.0013, f(b) = 0.0257 \Rightarrow f(a)f(b) < 0$$

Matlab

Nota: In Matlab la tabella precedente si otterrebbe con

```
>> lambda = (0.10:0.02:0.20);
```

```
>> f = [exp(lambda) + 0.435./lambda .* (exp(lambda)-1) - 1.564]
```

```
f =
```

```
-0.0013    0.0257    0.0532    0.0812    0.1098    0.1390
```

MA:

```
>> format long
```

```
>> lambda = (0.10:0.02:0.20);
```

```
>> f = [exp(lambda) + 0.435./lambda .* (exp(lambda)-1) - 1.564]
```

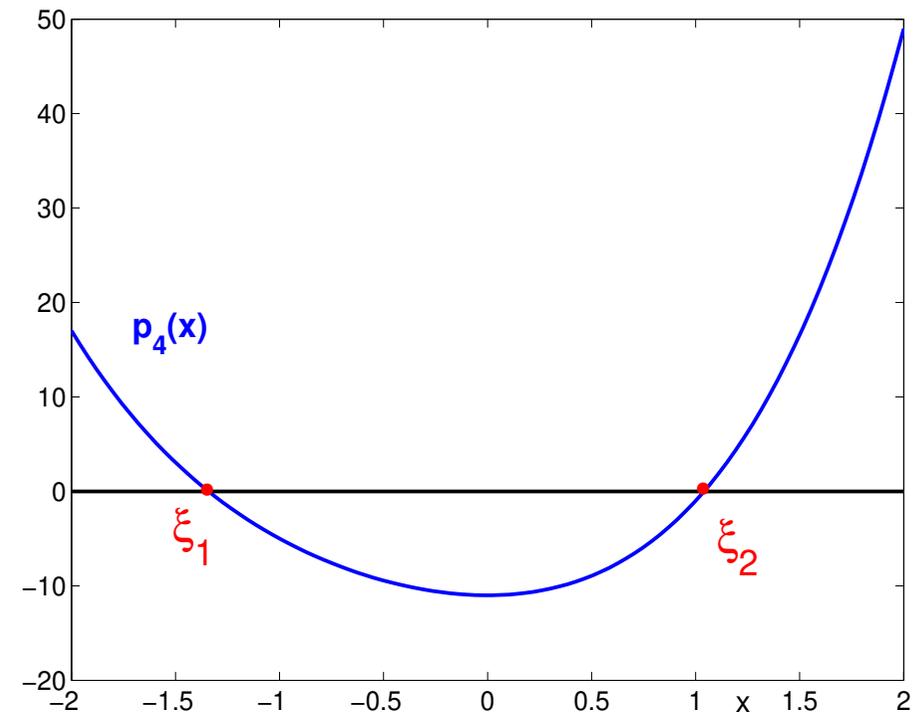
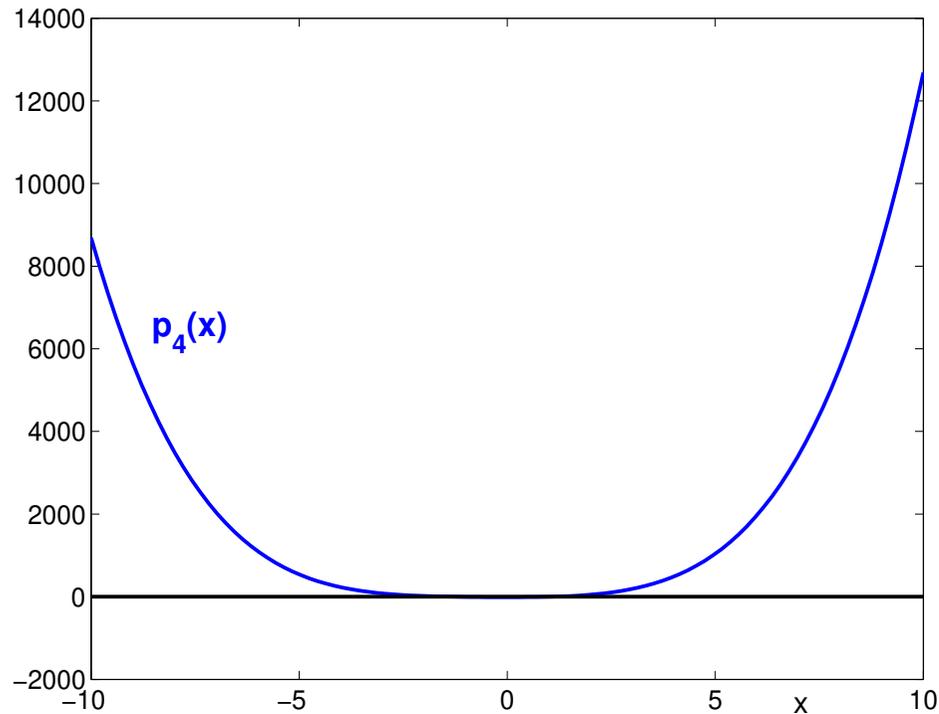
```
f =
```

```
-0.001335588295285    0.025672938554613    0.053195959592184
```

```
0.081243551500795    0.109825990666184    0.138953757158539
```

Separazione delle radici: esempio 1

Equazioni polinomiali: $p_4(x) = x^4 + 2x^3 + 7x^2 - 11 = 0$



Restringendo l'intervallo di osservazione si identificano meglio le due radici reali

Delle **4 radici** di $p_4(x)$ **due** sono **reali**, $\xi_1 \in [-1.5, -1]$ e $\xi_2 \in [0.75, 1.25]$, mentre **due** sono **complesse coniugate**.

Separazione delle radici: esempio 2

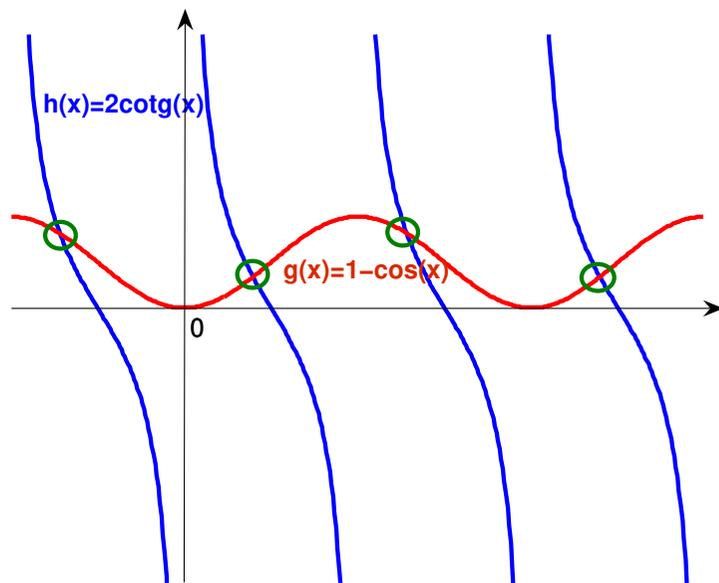
Equazione trascendente: $f(x) = \operatorname{tg}x(1 - \cos x) - 2 = 0$

L'equazione $f(x) = 0$ si può riscrivere come $1 - \cos x = 2\cot x$.

In questo modo è possibile risolvere il problema equivalente

$$h(x) = g(x)$$

corrispondente a determinare i punti di intersezione delle funzioni $h(x) = 2\cot x$ e $g(x) = 1 - \cos x$



Esistono **infinite soluzioni**
 $\xi_k \in (k\pi, (2k + 1)\frac{\pi}{2}), \quad k \in \mathbf{Z}$

Metodo di bisezione (o metodo dicotomico)

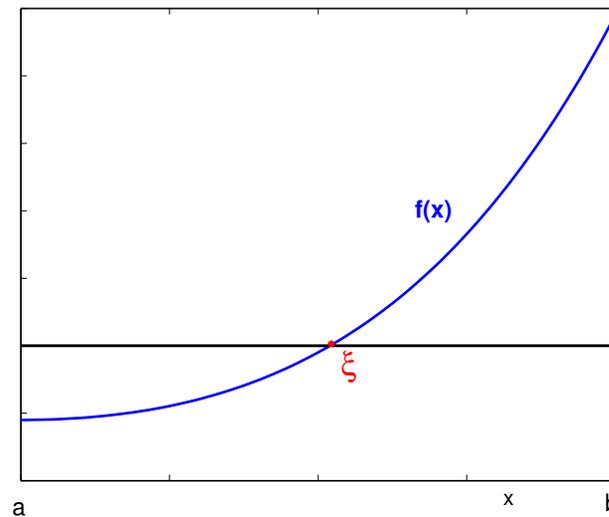
Il **metodo di bisezione** è un metodo molto **semplice**:

una volta individuato un intervallo di separazione in cui si trova **una sola** radice, permette di costruire una **succe-**
sione $\{x_k\}$ di **approssimazioni** di ξ .

Ipotesi di applicabilità :

- è stato **separato** un intervallo $I = [a, b]$ in cui c'è un'**unica radice** ξ ;
- la funzione f è **continua** in I : $f \in C^0[a, b]$;

- $f(a)f(b) < 0$.



Metodo di bisezione: algoritmo

Si genera una **successione** di approssimazioni $\{x_k\}$ con $x_k \in [a_{k-1}, b_{k-1}]$ e $\xi \in [a_{k-1}, b_{k-1}]$.

Algoritmo:

$$a_0 = a, \quad b_0 = b$$

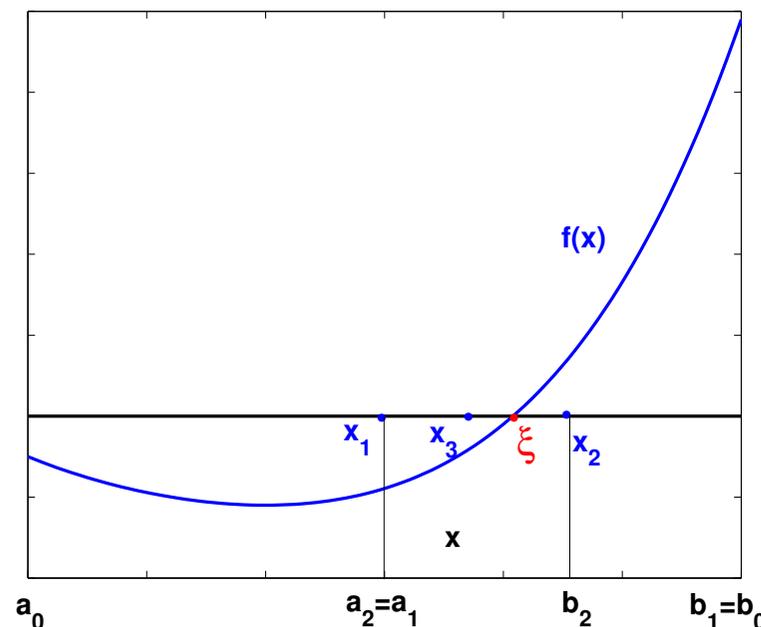
per $k = 1, 2, 3, \dots$

$$x_k = \frac{a_{k-1} + b_{k-1}}{2} \quad (\text{punto medio di } [a_{k-1}, b_{k-1}])$$

se $f(x_k) = 0$, allora stop

se $f(a_{k-1})f(x_k) < 0$, allora $[a_k, b_k] = [a_{k-1}, x_k]$

se $f(x_k)f(b_{k-1}) < 0$, allora $[a_k, b_k] = [x_k, b_{k-1}]$



Convergenza del metodo di bisezione

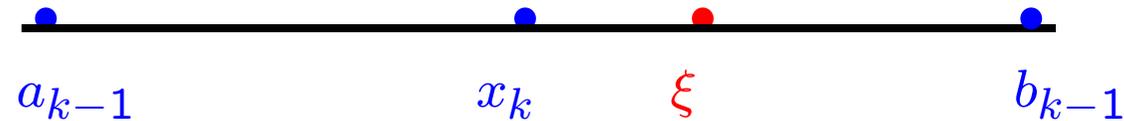
Errore di troncamento: $e_k = \xi - x_k$

è l'errore che si commette approssimando la radice ξ con il **k-esimo** elemento della successione costruita usando l'algoritmo descritto precedentemente.

Il procedimento iterativo converge alla radice ξ se la successione $\{x_k\}$ converge a ξ per $k \rightarrow +\infty$

Convergenza: $\lim_{k \rightarrow \infty} x_k = \xi \Leftrightarrow \lim_{k \rightarrow \infty} |e_k| = 0$

Per il **metodo di bisezione** si ha



Alla *k*-esima iterazione $\xi \in [x_k, b_{k-1}]$ oppure $\xi \in [a_{k-1}, x_k]$.

Ne segue che $|e_k| < \frac{b_{k-1} - a_{k-1}}{2}$.

Inoltre, considerando che l'ampiezza dell'intervallo $[a_{k-1}, b_{k-1}]$ è pari alla metà dell'ampiezza dell'intervallo $[a_{k-2}, b_{k-2}]$ costruito all'iterazione precedente, si ha

$$|e_k| < \frac{b_{k-1} - a_{k-1}}{2} = \frac{b_{k-2} - a_{k-2}}{2^2} = \dots = \frac{b - a}{2^k}$$

e quindi

$$0 \leq \lim_{k \rightarrow \infty} |e_k| < \lim_{k \rightarrow \infty} \frac{b - a}{2^k} = 0$$

Ordine di convergenza

Sia $\{x_k\}$ una **successione di approssimazioni convergente** a ξ . La successione ha **ordine di convergenza** p e **fattore di convergenza** C , se esistono due reali $p \geq 1$ e $C > 0$ tali che

$$\lim_{k \rightarrow \infty} \frac{|e_{k+1}|}{|e_k|^p} = C$$

Nota. La convergenza si dice **lineare** se $p = 1$,
quadratica se $p = 2$.

Metodo di bisezione: ordine di convergenza

Per $k \rightarrow \infty$ si ha $\frac{|e_{k+1}|}{|e_k|} \simeq \frac{\frac{b-a}{2^{k+1}}}{\frac{b-a}{2^k}} = \frac{2^k}{2^{k+1}} = \frac{1}{2}$.

⇒ **Ordine di convergenza: 1 (lineare)**

Fattore di convergenza: $\frac{1}{2}$

La convergenza è **lenta**, in quanto ad ogni passo l'**errore** viene **dimezzato**, cioè ad ogni passo si guadagna una **cifra binaria**

⇒ poiché $2^{-4} < 10^{-1} < 2^{-3}$, per guadagnare una **cifra decimale** servono **3-4 iterazioni**.

Metodo di bisezione: criteri di arresto

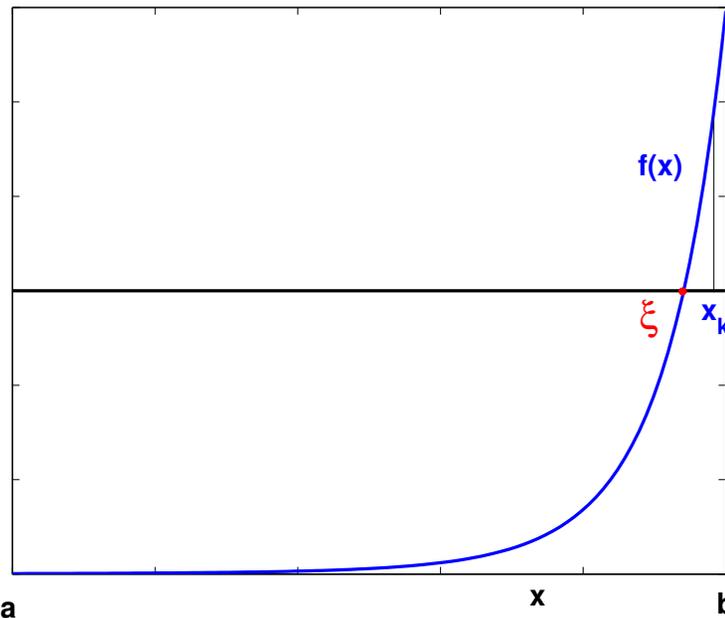
Nella pratica, a causa degli **errori di arrotondamento** e degli **errori di troncamento** non si verifica **mai** che $f(x_k) = 0$. Quando si arrestano le iterazioni?

Criteri di arresto a posteriori $\left\{ \begin{array}{l} |e_k| \simeq |x_k - x_{k-1}| < \varepsilon \\ |f(x_k)| < \varepsilon \end{array} \right.$

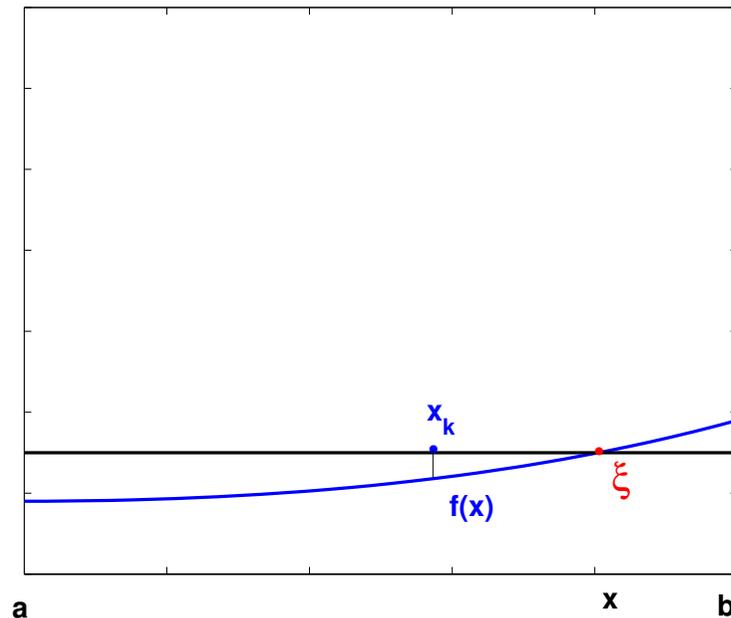
Criteri di arresto a posteriori: esempi

$$|e_k| \simeq |x_k - x_{k-1}| < \epsilon$$

$$\text{oppure } |f(x_k)| < \epsilon$$



$f(x_k)$ è "*grande*" anche se
 x_k è "*vicino*" a ξ



$f(x_k)$ è "*piccolo*" anche
se x_k è "*lontano*" da ξ

Criterio di arresto a priori:

Usando l'espressione dell'errore di troncamento, è possibile dare una **stima a priori** del numero di iterazioni K necessario per ottenere un **errore minore** di ε . Infatti, risulta

$$|e_k| < \frac{b-a}{2^k} < \varepsilon \quad \Rightarrow \quad K > \frac{\log(b-a) - \log(\varepsilon)}{\log 2}$$

Oss.: Poichè K deve essere un numero intero positivo, può essere posto uguale all'intero più grande e più vicino alla quantità a secondo membro dell'ultima disequazione.

Soluzione Problema 3

Si vuole usare il metodo di bisezione per calcolare la radice dell'equazione

$$f(\lambda) = e^\lambda + \frac{0.435}{\lambda}(e^\lambda - 1) - 1.564 = 0 \quad \text{in} \quad I = [a, b] = [0.05, 0.15]$$

k	a_{k-1}	b_{k-1}	x_k	$ x_k - x_{k-1} $	$ f(x_k) $
1	0.0500000000000000	0.1500000000000000	0.1000000000000000	10.0000000000000000	10.0000000000000000
2	0.1000000000000000	0.1500000000000000	0.1250000000000000	0.0250000000000000	0.03250506973938
3	0.1000000000000000	0.1250000000000000	0.1125000000000000	0.0125000000000000	0.01548498364220
4	0.1000000000000000	0.1125000000000000	0.1062500000000000	0.0062500000000000	0.00704990930651
5	0.1000000000000000	0.1062500000000000	0.1031250000000000	0.0031250000000000	0.00285098217571
6	0.1000000000000000	0.1031250000000000	0.1015625000000000	0.0015625000000000	0.00075615469668
7	0.1000000000000000	0.1015625000000000	0.1007812500000000	0.0007812500000000	0.00029010206821
8	0.1007812500000000	0.1015625000000000	0.1011718750000000	0.0003906250000000	0.00023292996053
9	0.1007812500000000	0.1011718750000000	0.1009765625000000	0.0001953125000000	0.00002861013771
10	0.1009765625000000	0.1011718750000000	0.1010742187500000	0.0000976562500000	0.00010215388987
11	0.1009765625000000	0.1010742187500000	0.1010253906250000	0.0000488281250000	0.00003677037077

Dalla tabella è possibile osservare che se si sceglie $\varepsilon = 0.5 \cdot 10^{-4}$ e si usa come criterio di arresto $|f(x_k)| < \varepsilon$, il procedimento iterativo si interrompe quando $k = 9$ ($|f(x_k)| = 0.00002861013771$).

Scegliendo come criterio di arresto $|x_k - x_{k-1}| < \varepsilon$, il procedimento iterativo si arresta per $k = 11$ ($|x_k - x_{k-1}| = 0.00004882812500$).

E' possibile osservare che per $k = 11$ è soddisfatta anche la condizione $|f(x_k)| < \varepsilon$.

Usando, invece, il criterio di arresto a priori, si ha

$$K > \log_2(0.10) - \log_2(0.5 \cdot 10^{-4}) \approx 10.9658$$

cioè $K \geq 11$,

Esercizio 1

Esempio 3.2.1 (*Libro*) Stabilire quante (e quali) radici ammette l'equazione

$$f(x) = \log(x + 1) + \sqrt{x + 2} - 1 = 0.$$

Soluzione La funzione è definita e continua nell'intervallo $(-1, +\infty)$.
Inoltre

$$\lim_{x \rightarrow -1} f(x) = -\infty < 0, \quad \lim_{x \rightarrow +\infty} f(x) = +\infty > 0.$$

Pertanto, lo zero (l'intervallo di zeri) esiste.

Poichè $f'(x) = \frac{1}{x+1} + \frac{1}{2\sqrt{x+2}} < 0$, la funzione è monotona nel suo dominio di definizione e quindi ha un unico zero in $(-1, +\infty)$.

Infine, si osserva che $\forall x \geq 0 \quad f(x) > 0$, quindi lo zero di f sicuramente appartiene all'intervallo $(-1, 0]$.

Poichè $f(-1/2) < 0$, si può scegliere $I = [a, b] = [-1/2, 0]$ come **intervallo di separazione**.

Applicando il **metodo di bisezione** si ha

$$x_1 = \frac{a_0 + b_0}{2} = \frac{-1/2}{2} = -\frac{1}{4}$$

$$f(a_0) = -0.468$$

$$f(b_0) = +0.414 \Rightarrow \begin{matrix} a_1 = a_0 \\ b_1 = x_1 \end{matrix}$$

$$f(x_1) = +0.035$$

$$x_2 = \frac{a_1 + b_1}{2} = \frac{-1/2 - 1/4}{2} = -\frac{3}{8}$$

$$f(a_1) = -0.468$$

$$f(b_1) = +0.035 \Rightarrow \begin{matrix} a_2 = x_2 \\ b_2 = b_1 \end{matrix}$$

$$f(x_2) = -0.195$$

.....

k	x_k	f(x_k)	 ξ - x_k
1	-0.2500000	0.0351936	0.0203336
2	-0.3750000	-0.1952488	0.1046664
3	-0.3125000	-0.0756553	0.0421664
4	-0.2812500	-0.0192306	0.0109164
5	-0.2656250	0.0082212	0.0047086
6	-0.2734375	-0.0054435	0.0031039
7	-0.2695313	0.0014040	0.0008023
8	-0.2714844	-0.0020160	0.0011508
9	-0.2705078	-0.0003050	0.0001742

Dopo 9 iterazioni, l'approssimazione prodotta produce un errore dell'ordine di 10^{-3}

Il numero di iterazioni K necessarie affinché la soluzione prodotta sicuramente abbia una certa precisione ε può essere stimato prima di eseguire le iterazioni nel modo seguente:

$$K \geq \log_2(b - a) - \log_2(\varepsilon)$$

con $\varepsilon = 0.5 \cdot 10^{-3}$ da cui

$$K > \log_2(0.5) - \log_2(0.5 \cdot 10^{-3}) \approx 9.9658$$

cioè $K \geq 10$.

Il valore di K stimato assicura che l'errore tra due approssimazioni successive sia inferiore alla tolleranza fissata. Ovviamente, poichè K deriva da una stima superiore dell'errore di troncamento, può accadere che il metodo raggiunga la precisione richiesta con un numero di iterazioni **inferiore** al valore K stimato usando la **stima a priori**.

Esercizio 2

Si consideri

$$f(x; \lambda) = e^{-x} - 2x - \lambda,$$

con $\lambda \in \mathbf{R}$.

- Determinare per quali valori del parametro λ la funzione $f(x)$ ammette un unico zero nell'intervallo $[0, 1]$.
- Posto $\lambda = -1$, dare una stima del numero di iterazioni necessarie per avere un' approssimazione dello zero ξ con almeno tre decimali esatti usando il metodo di bisezione.

Soluzione

$f(x; \lambda) \in C^\infty(\mathbf{R})$. Inoltre, $f'(x; \lambda) < 0$, $\forall x \in \mathbf{R}$. Pertanto, se lo zero esiste, allora è unico.

Poichè $f(x; \lambda) = e^{-x} - 2x - \lambda$:

$$f(0)f(1) < 0 \iff (1 - \lambda)(1/e - 2 - \lambda) < 0$$

segue che $f(x; \lambda)$ ammette uno zero per valori di $\lambda \in I = \left(\frac{1}{e} - 2, 1\right)$.

Si osserva che $-1 \in I$ e che le condizioni di applicabilità del metodo di bisezione sono verificate; dunque, applicando la condizione di arresto a priori si ha

$$K > \log_2(1 - 0) - \log_2(0.5 \cdot 10^{-3}) \approx 10.9658.$$

Pertanto, $K \geq 11$.

Metodi di linearizzazione

Si approssima la funzione $f(x)$ in un intorno I di ξ con la sua **tangente** o con la sua **secante**, calcolate tramite un opportuno **sviluppo in serie di Taylor**.

- **Metodo di Newton-Raphson o metodo delle tangenti**
- **Metodo delle secanti**

Metodo di Newton-Raphson

Approssimazione iniziale: x_0

Prima iterazione:

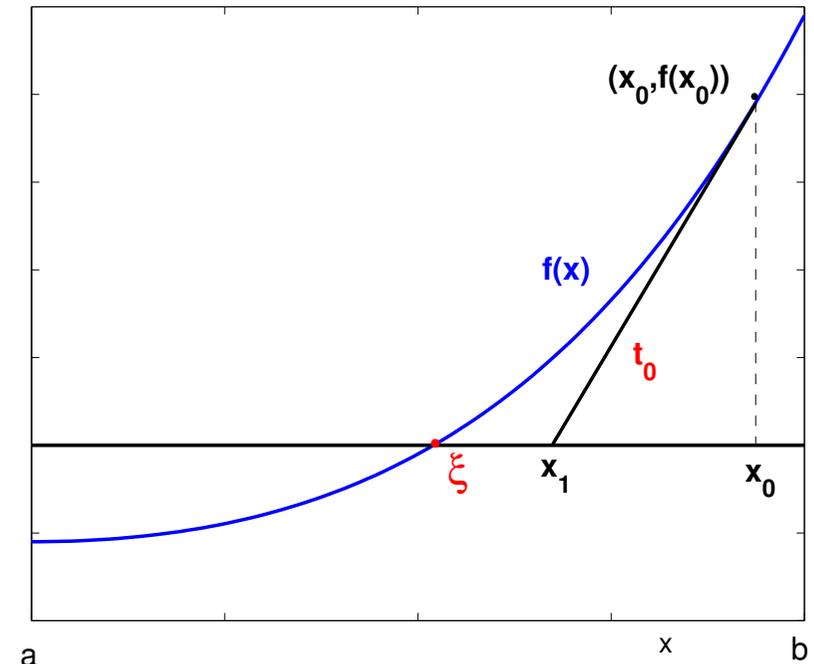
t_0 è la **retta tangente** a $f(x)$
nel punto $(x_0, f(x_0))$:

$$t_0 \rightarrow y = f(x_0) + f'(x_0)(x - x_0)$$

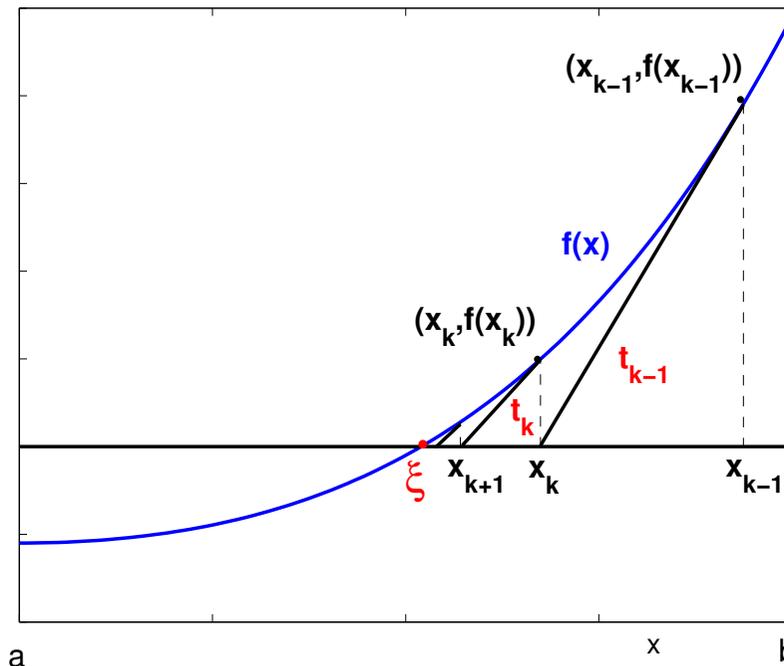
Nuova approssimazione x_1 :

intersezione tra t_0 e $y = 0$

$$\Rightarrow f(x_0) + f'(x_0)(x_1 - x_0) = 0 \rightarrow x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$



Metodo di Newton-Raphson: algoritmo



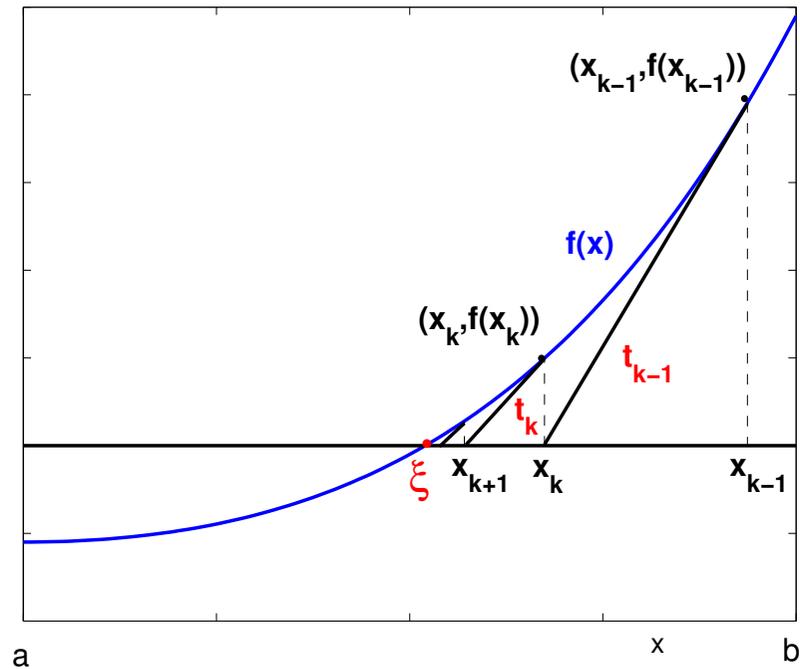
Ad ogni **iterazione** $k = 1, 2, \dots$ la **nuova approssimazione** x_k è data dall'**intersezione** tra la **retta** t_{k-1} , **tangente** a $f(x)$ nel punto $(x_{k-1}, f(x_{k-1}))$, e la retta $y = 0$:

$$t_{k-1} \rightarrow y = f(x_{k-1}) + f'(x_{k-1})(x - x_{k-1})$$

e quindi $f(x_{k-1}) + f'(x_{k-1})(x_k - x_{k-1}) = 0$



Metodo di Newton-Raphson: algoritmo



Algoritmo:

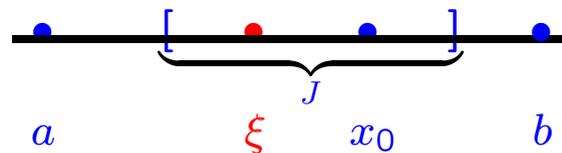
$$\begin{cases} x_0 & \text{dato} \\ x_k = x_{k-1} - \frac{f(x_{k-1})}{f'(x_{k-1})}, & k = 1, 2, \dots \end{cases}$$

Metodo di Newton-Raphson: convergenza

$$\begin{cases} x_0 & \text{dato} \\ x_k = x_{k-1} - \frac{f(x_{k-1})}{f'(x_{k-1})}, & k = 1, 2, \dots \end{cases}$$

Ipotesi di applicabilità :

- è stato **separato** un intervallo $I = [a, b]$ in cui c'è un'**unica radice** ξ ;
 - f, f', f'' sono **continue** in I : $f \in C^2[a, b]$;
 - $f'(x) \neq 0$ per $x \in [a, b]$
- \Rightarrow esiste un **intorno** $J \subseteq I$ di ξ tale che, se $x_0 \in J$, la **successione delle approssimazioni** $\{x_k\}$ **converge** a ξ .



Oss: il teorema garantisce solo l'esistenza di J

Metodo di Newton-Raphson: ordine di convergenza

Si vuole determinare l'ordine di convergenza del metodo di Newton-Raphson

Ordine di convergenza p: $\lim_{k \rightarrow \infty} \frac{|e_{k+1}|}{|e_k|^p} = C$

L'errore di troncamento alla *k+1-esima* iterazione si può scrivere come segue

$$e_{k+1} = \xi - x_{k+1} = \left(\xi - \frac{f(\xi)}{f'(\xi)} \right) - \left(x_k - \frac{f(x_k)}{f'(x_k)} \right) = (\xi - x_k) - \left(\frac{f(\xi)}{f'(\xi)} - \frac{f(x_k)}{f'(x_k)} \right)$$

Il valore di $f(x_k)$ si può stimare considerando i primi tre termini dello **sviluppo in serie di Taylor** attorno alla radice ξ , cioè

$$f(x_k) = f(\xi) + f'(\xi) \underbrace{(x_k - \xi)}_{-e_k} + \frac{1}{2} f''(\xi) (x_k - \xi)^2 + \dots$$

mentre, supponendo che x_k sia molto vicino a ξ , si può assumere che $f'(x_k) \simeq f'(\xi)$

Sostituendo i valori di $f(x_k)$ e $f'(x_k)$ così ottenuti nell'espressione di e_{k+1} si ha

$$|e_{k+1}| \simeq \left| e_k - \frac{f(\xi) - f(\xi) + f'(\xi)e_k - \frac{1}{2}f''(\xi)e_k^2}{f'(\xi)} \right| = \left| \frac{\frac{1}{2}f''(\xi)e_k^2}{f'(\xi)} \right|$$

da cui risulta

$$\lim_{k \rightarrow \infty} \frac{|e_{k+1}|}{|e_k|^2} = \frac{1}{2} \left| \frac{f''(\xi)}{f'(\xi)} \right| \Rightarrow \boxed{p \geq 2}$$

e quindi, se $f(x) \in C^3[a, b]$ la convergenza è almeno **quadratica**

Efficienza computazionale

Per valutare l'**efficienza** di un metodo iterativo bisogna tener conto sia dell'**ordine di convergenza** che del **costo computazionale**, cioè della quantità di calcoli richiesta ad ogni passo.

Efficienza computazionale: $E = p^{1/r}$

p : **ordine** di convergenza del metodo

r : numero di **valutazioni funzionali** (calcolo di funzioni o derivate) richieste ad ogni passo

Metodo di bisezione: $E = 1$

(ad ogni passo si richiede una sola valutazione funzionale, $f(x_k)$, e quindi $r = 1$)

Metodo di Newton: $E = 2^{1/2}$

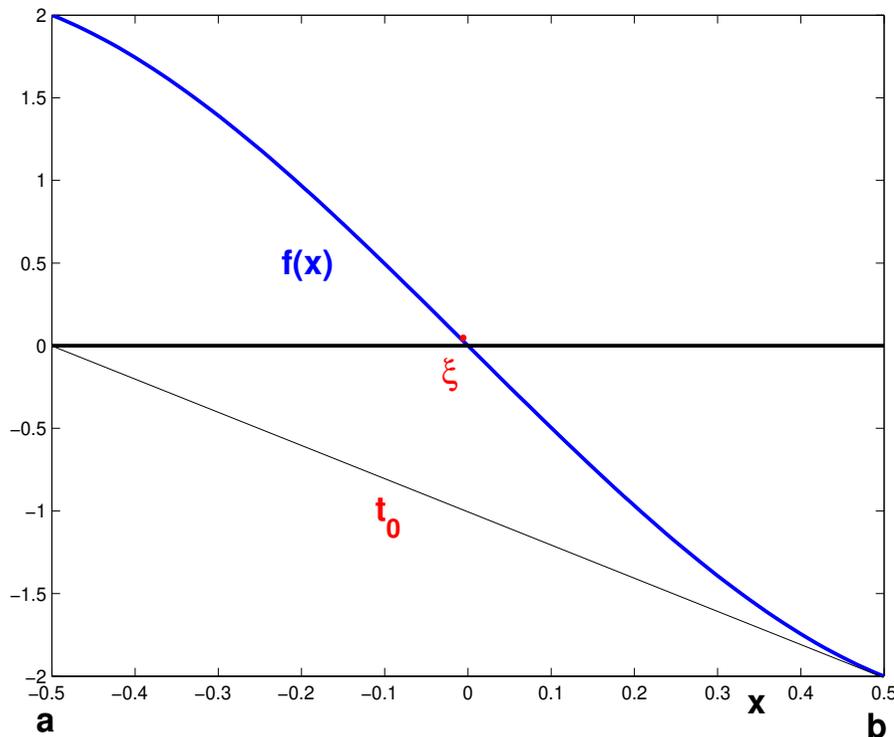
(ad ogni passo si richiedono due valutazioni funzionali, $f(x_k)$ e $f'(x_k)$, e quindi $r = 2$)

Metodo di Newton-Raphson: esempio

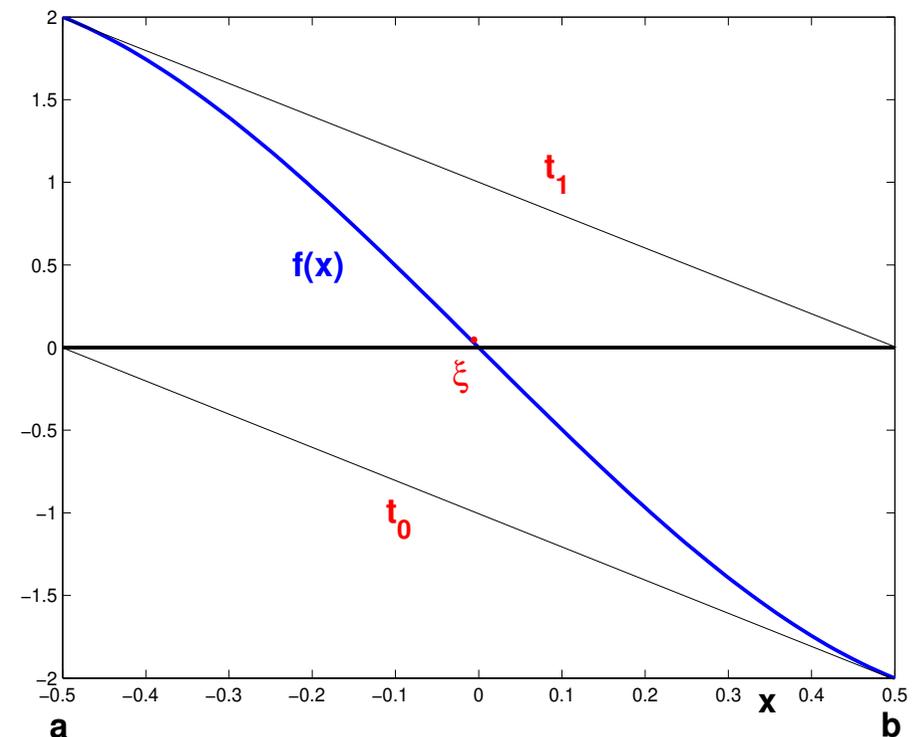
Approssimare la radice $\xi = 0$ dell'equazione

$$f(x) = 4x^3 - 5x = 0$$

con il **metodo di Newton-Raphson** nell'intervallo $I = [-0.5, 0.5]$ e scegliendo come approssimazione iniziale una volta $x_0 = 0.5$ e una volta $x_0 = 0.4$.



$$x_{2k} = 0.5$$



$$x_{2k+1} = -0.5$$

Si verifica facilmente che f verifica le condizioni di applicabilità del **metodo di Newton-Raphson**.

Infatti, $f \in C^2(I)$, $f'(x) = 12x^2 - 5 = 0 \iff x = \pm \frac{\sqrt{5}}{\sqrt{12}} \notin I$.

Scegliendo $x_0 = 0.5$ si ha una *situazione di stallo* e il metodo non converge. Infatti, l'algoritmo di Newton per f è il seguente

$$x_k = x_{k-1} - \frac{4x_{k-1}^3 - 5x_{k-1}}{12x_{k-1}^2 - 5}$$

e quindi

$$x_1 = x_0 - \frac{4x_0^3 - 5x_0}{12x_0^2 - 5} = 0.5 - \frac{-2}{-2} = -0.5$$

$$x_2 = x_1 - \frac{4x_1^3 - 5x_1}{12x_1^2 - 5} = -0.5 - \frac{3}{-2} = 0.5 \quad ???$$

Al contrario, scegliendo $x_0 = 0.4$, il metodo converge

k	x_k	$ x_k - x_{k-1} $	$ f(x_k) $
1	-0.16623376623377	0.56623376623377	0.81279423831355
2	0.00787190837207	0.17410567460584	0.03935759066802
3	-0.00000078059303	0.00787268896510	0.00000390296513
4	0.00000000000000	0.00000078059303	0.00000000000000

Infatti, il teorema di convergenza del metodo di Newton-Raphson assicura la convergenza per ogni scelta dell'approssimazione iniziale in un opportuno intorno J del punto ξ . Scegliendo come approssimazioni iniziali punti che non appartengono all'intorno J , la convergenza non può essere garantita.

Oss: $f''(x) = 24x = 0 \iff x = 0 \in I$.

Estremo di Fourier

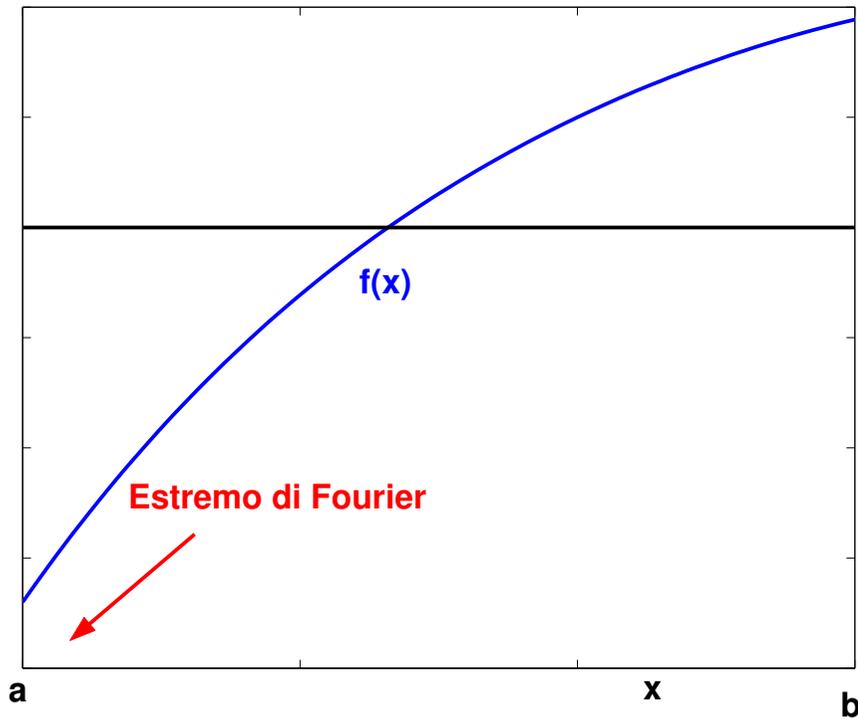
Se $f(x)$ ha **concavità fissa** in un intervallo I , è possibile stabilire un criterio di scelta dell'approssimazione iniziale che garantisce la **convergenza** del metodo.

Estremo di Fourier:

Data una funzione f **continua** e **convessa** in $I = [a, b]$ con $f(a)f(b) < 0$, si dice **estremo di Fourier** di I l'estremo verso cui f rivolge la convessità .

Se **esiste** f'' , allora l'estremo di Fourier è a o b a seconda che $f(a)f''(a) > 0$ oppure $f(b)f''(b) > 0$.

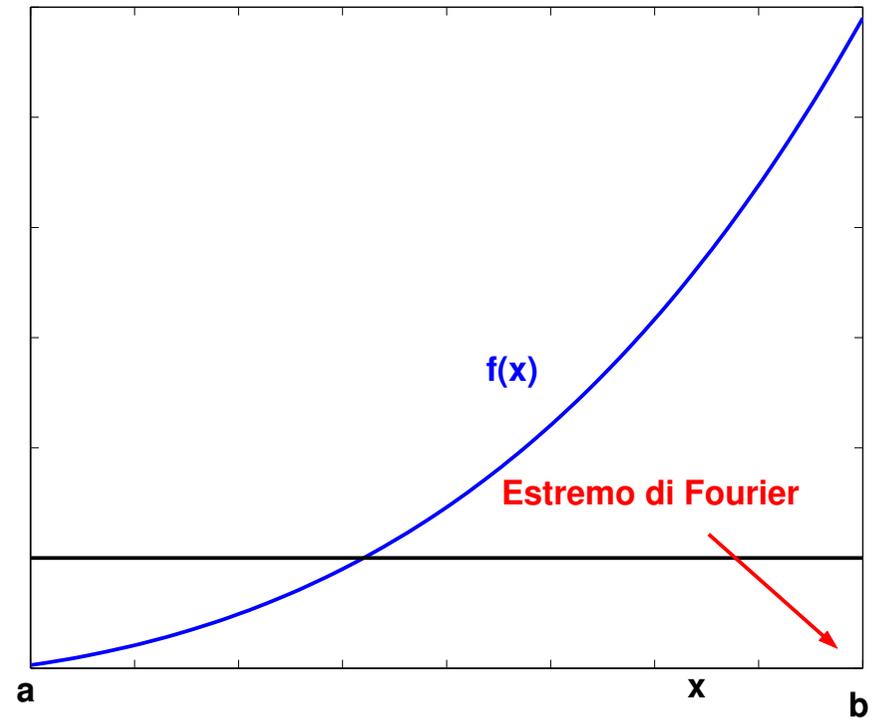
Estremo di Fourier: esempi



$$f''(x) < 0 \text{ per } x \in [a, b]$$

$$\begin{cases} f(a)f''(a) > 0 \\ f(b)f''(b) < 0 \end{cases}$$

$\Rightarrow a$ è **estremo di Fourier**



$$f''(x) > 0 \text{ per } x \in [a, b]$$

$$\begin{cases} f(a)f''(a) < 0 \\ f(b)f''(b) > 0 \end{cases}$$

$\Rightarrow b$ è **estremo di Fourier**

Metodo di Newton-Raphson: convergenza

Ipotesi di applicabilità :

- $f(a)f(b) < 0$
- f, f', f'' sono **continue** in I : $f \in C^2[a, b]$;
- $f'(x) \neq 0$ per $x \in [a, b]$;
- $f''(x) \neq 0$ per $x \in [a, b]$ e x_0 è l'**estremo di Fourier** di $[a, b]$.

⇒

- 1) esiste un'**unica radice** $\xi \in [a, b]$;
- 2) la **successione delle approssimazioni**

$$\left\{ x_k = x_{k-1} - \frac{f(x_{k-1})}{f'(x_{k-1})} \right\} \quad k = 1, 2, \dots$$

è **monotona** e **converge** a ξ ;

- 3) se $f \in C^3[a, b]$, la convergenza è **quadratica**.

Nell'esempio precedente, relativo alla funzione $f(x) = 4x^3 - 5x$, non è soddisfatta l'ipotesi relativa alla derivata seconda. Al contrario, per la funzione $f(x) = x^3 - 10x^2 + 5$, considerando l'intervallo $I = [0.6, 0.8]$, si ha

$$f''(x) = 6x - 20 = 0 \iff x = \frac{10}{3} \notin I$$

Le ipotesi del teorema sono verificate e, poichè

$$f''(0.6) = \frac{18}{5} - 20 < 0 \quad f''(0.8) = \frac{24}{5} - 20 < 0,$$

mentre

$$f(0.6) > 0 \quad f(0.8) < 0,$$

l'estremo di Fourier è l'estremo $b = 0.8$, che quindi può essere scelto come approssimazione iniziale del metodo di Newton-Raphson, cioè $x_0 = 0.8$.

Esercizio: eseguire le iterazioni e verificare la convergenza monotona.

Problema 2: Soluzione con Metodo di NR

$$f(\lambda) = e^\lambda + \frac{0.435}{\lambda}(e^\lambda - 1) - 1.564 = 0 \quad \lambda \in I = [0.05, 0.15]$$

- Nell'intervallo I è stato separato un **unico zero**
- f, f', f'' sono **continue** in I
- $f'(\lambda) = e^\lambda - \frac{0.435}{\lambda^2}(e^\lambda - 1) + \frac{0.435}{\lambda}e^\lambda \neq 0$ per $\lambda \in I$

\Rightarrow Lo zero può essere approssimato con il **metodo delle tangenti**

Inoltre $f''(\lambda) = e^\lambda + \frac{0.87}{\lambda^3}(e^\lambda - 1) - \frac{0.87}{\lambda^2}e^\lambda + \frac{0.435}{\lambda}e^\lambda > 0$ in I

\Rightarrow esiste l'**estremo di Fourier** di I :

$$f(0.15)f''(0.15) > 0 \Rightarrow x_0 = 0.15$$

k	x_k	$ x_k - x_{k-1} $	$ f(x_k) $
0	0.1500000000000000	10.0000000000000000	0.06715354664030
1	0.10211384134812	0.04788615865188	0.00149497988981
2	0.10099851665312	0.00111532469500	0.00000078594305
3	0.10099792968591	0.00000058696721	0.000000000000022
4	0.10099792968575	0.000000000000016	0.000000000000000
5	0.10099792968575	0.000000000000000	0.000000000000000

Ripetere scegliendo $x_0 = 0.05$

Metodi di linearizzazione

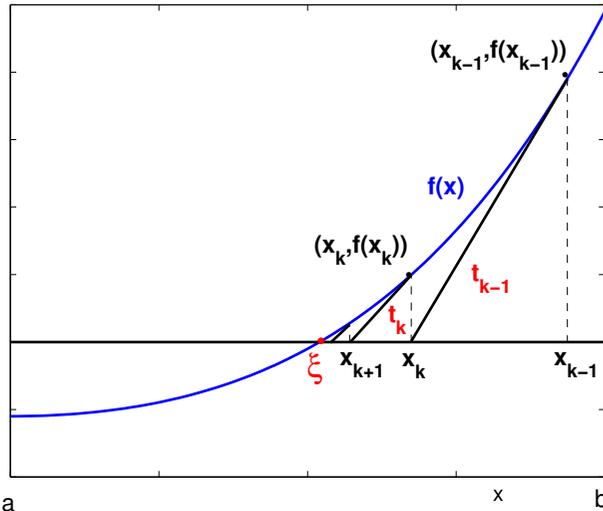
Il metodo di Newton-Raphson richiede la conoscenza della $f'(x)$ che non sempre è di facile valutazione.

Metodi alternativi sono, per esempio,

- il metodo della *tangente fissa*: ad ogni iterazione $f'(x_n)$ è approssimato con $f'(x_0)$
- il metodo delle *secanti con estremi variabili*: $f'(x_n)$ è approssimato con il rapporto incrementale
- il metodo delle *secanti con estremo fisso*: $f'(x_n)$ è approssimato con il rapporto incrementale in cui un punto rimane fisso ad ogni iterazione

Metodi di linearizzazione

Metodo di Newton-Raphson

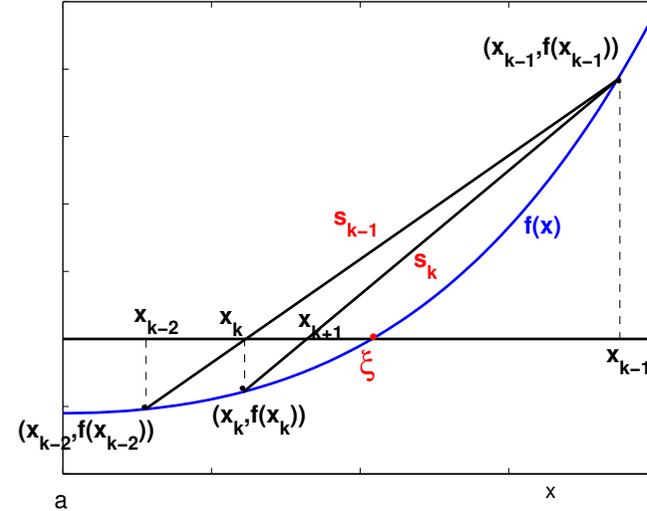


Ad ogni **iterazione** $k = 1, 2, \dots$ la **nuova approssimazione** x_k è data dall'**intersezione** tra la **retta** t_{k-1} , **tangente** a $f(x)$ nel punto $(x_{k-1}, f(x_{k-1}))$, e $y = 0$.

Algoritmo:

$$\begin{cases} x_0 \text{ dato} \\ x_k = x_{k-1} - \frac{f(x_{k-1})}{f'(x_{k-1})}, \quad k = 1, 2, \dots \end{cases}$$

Metodo delle secanti con estremi variabili



Ad ogni **iterazione** $k = 2, 3, \dots$ la **nuova approssimazione** x_k è data dalla **intersezione** tra la **retta** s_{k-1} , **secante** $f(x)$ nei punti $(x_{k-2}, f(x_{k-2}))$ e $(x_{k-1}, f(x_{k-1}))$, e $y = 0$.

Algoritmo:

$$\begin{cases} x_0, x_1 \text{ dati} \\ x_k = x_{k-1} - f(x_{k-1}) \frac{x_{k-1} - x_{k-2}}{f(x_{k-1}) - f(x_{k-2})}, \quad k \geq 2 \end{cases}$$

Metodo delle secanti

$$\begin{cases} x_0, x_1 & \text{dati} \\ x_k = x_{k-1} - f(x_{k-1}) \frac{x_{k-1} - x_{k-2}}{f(x_{k-1}) - f(x_{k-2})}, & k = 2, \dots \end{cases}$$

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 funzionale

Svantaggi:

- servono **due approssimazioni iniziali** x_0 e x_1
- la scelta di x_0 e x_1 deve essere **”accurata”**

Convergenza del metodo delle secanti

Se

- è stato **separato** un intervallo $I = [a, b]$ **simmetrico** intorno alla **radice** ξ ,
- f, f', f'' sono **continue** in I : $f \in C^2[a, b]$,
- $f'(x) \neq 0$ per $x \in [a, b]$,

\Rightarrow esiste un **intorno** $J \subseteq I$ di ξ tale che, se $x_0, x_1 \in J$, la **successione delle approssimazioni** $\{x_k\}$ **converge** a ξ con convergenza **superlineare**, cioè $1 < p < 2$.

Se $f''(x) \neq 0$ in I , l'**ordine di convergenza** è $p = \frac{1+\sqrt{5}}{2}$
 $\Rightarrow E = p \simeq 1.62$

Convergenza del metodo delle secanti

Se $f''(x) \neq 0$ in I , allora $e_{k+1} \approx C_N^{\frac{p}{p+1}} e_k^p$

con $p = \frac{1+\sqrt{5}}{2}$ e

C_N la costante asintotica del metodo di Newton ($C_N = \frac{f''(\xi)}{2f'(\xi)}$).

Dim:

. . .

Esercizio

Data l'equazione non lineare

$$f(x) = x^3 + \alpha - \cos x = 0:$$

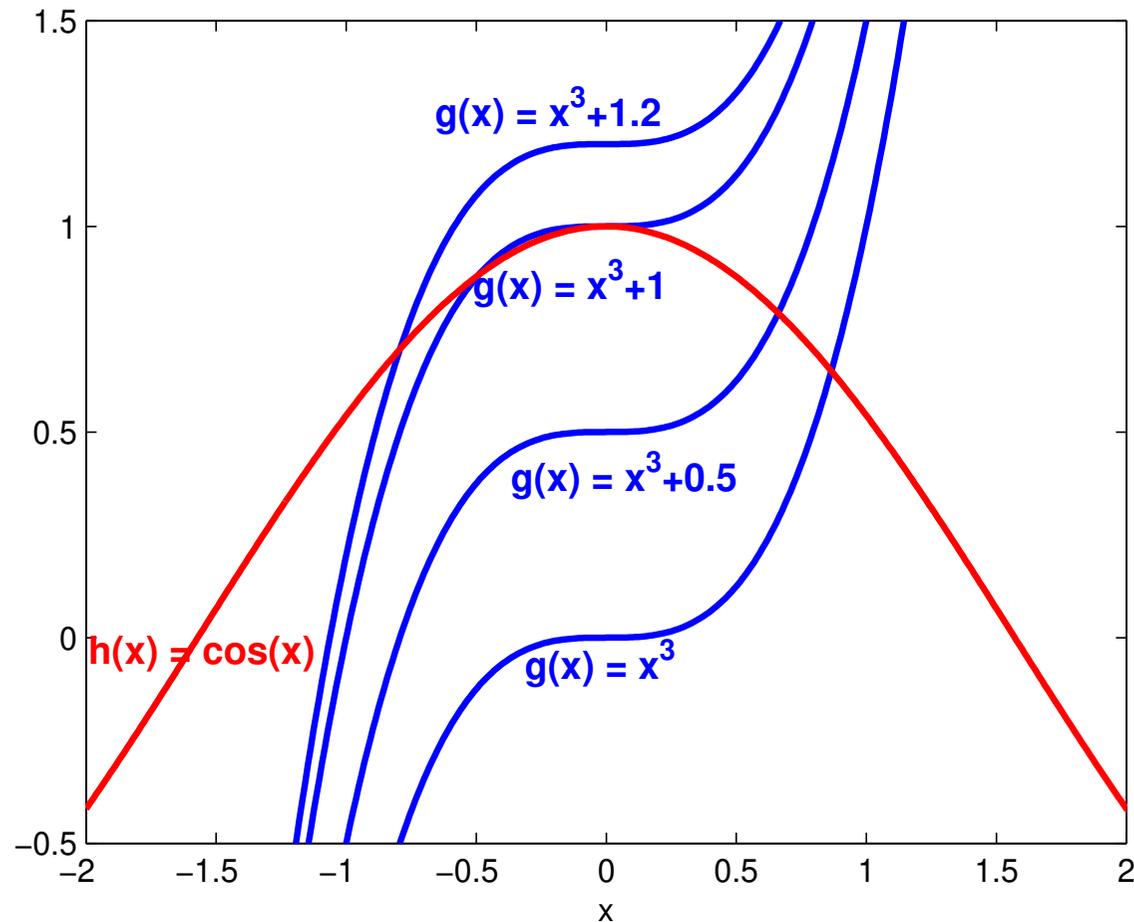
- 1) individuare per quali valori di $\alpha \in \mathbf{R}$, l'equazione non ammette radici positive
- 2) per $\alpha = \frac{1}{3}$ separare la radice più piccola $\tilde{\xi}$
- 3) fornire una stima a priori del numero di iterazioni necessarie per approssimare $\tilde{\xi}$ con un errore inferiore a $\epsilon = 10^{-6}$ tramite il **metodo di bisezione**;
- 4) quante iterazioni sono necessarie per approssimare con la stessa tolleranza ϵ la radice $\tilde{\xi}$ tramite il **metodo di Newton** e il **metodo delle secanti**?

Traccia della soluzione

1) Tracciando **qualitativamente** i grafici delle funzioni

$$y = g(x) = x^3 + \alpha \quad y = h(x) = \cos x,$$

si deduce che se $\alpha \geq 1$, la funzione $f(x)$ non ha zeri positivi.



2) L'intervallo $[0, 1]$ contiene l'**unica** radice positiva dell'equazione:

$$f(x) = x^3 + 1/3 - \cos x = 0$$

Infatti

$$f(0) = -2/3 < 0, f(1) = 4/3 - \cos(1) > 0 \quad \Rightarrow \quad f(0)f(1) < 0$$

e inoltre

$$f'(x) = 3x^2 + \sin(x) > 0 \quad \text{per } x \in (0, 1]$$

3) Nell'intervallo $[0, 1]$ sono verificate le **ipotesi di applicabilità** del metodo di bisezioni.

Quindi il numero di iterazioni K per cui $|e_K| \leq \epsilon$ si ricava dalla relazione

$$|e_K| = |\xi - x_K| \leq \frac{b-a}{2^K} \leq \epsilon$$
$$\Rightarrow K > \frac{\log(b-a) - \log \epsilon}{\log 2}.$$

In questo caso $a = 0$, $b = 1$, $\epsilon = 10^{-6}$

$$\Rightarrow K > \frac{\log(1) - \log 10^{-6}}{\log 2} \approx 19.9320 \Rightarrow K \geq 20.$$

4) Nel caso dei metodi di Newton e delle secanti si possono verificare le **ipotesi di applicabilità** nell'intervallo $[0, 1]$:

- nell'intervallo I è stato separato un **unico zero**
- f, f', f'' sono **continue** in I
- $f'(x) = 3x^2 + \sin(x) > 0$ per $x \in J \subset I$, $J = [\delta, 1]$ con $0 < \delta \ll 1$
- $f''(x) = 6x + \cos(x) > 0$ per $x \in I$

\Rightarrow l'estremo $b = 1$ è l'**estremo di Fourier** dell'intervallo J .

Il numero di iterazioni si può calcolare eseguendo le iterate.

In alternativa, l'approssimazione iniziale si può scegliere come la soluzione stimata eseguendo un'iterazione del metodo di bisezione, ovvero $x_0 = \frac{1}{2}$. (**OSS**: non è garantita la convergenza!)

k	x_k (bisez.)	$ x_k - x_{k-1} $	x_k (Newton)	$ x_k - x_{k-1} $	x_k (secanti)	$ x_k - x_{k-1} $
1	0.500000000		1.		0., 1.	
2	0.750000000	0.25e+0	0.793560583	0.21e+0	0.456715571	0.54e+0
3	0.625000000	0.12e+0	0.742925006	0.51e-1	0.658587384	0.20e+0
4	0.687500000	0.62e-1	0.739971453	0.29e-2	0.775393863	0.12e+0
5	0.718750000	0.31e-1	0.739961685	0.98e-5	0.736625691	0.39e-1
6	0.734375000	0.16e-1	0.739961685	0.11e-9	0.739832822	0.32e-2
7	0.742187500	0.78e-2			0.739962167	0.13e-3
8	0.738281250	0.39e-2			0.739961685	0.48e-6
9	0.740234375	0.19e-2				
10	0.739257812	0.97e-3				
11	0.739746097	0.49e-3				
12	0.739990234	0.24e-3				
13	0.739868164	0.12e-3				
14	0.739929199	0.61e-4				
15	0.739959717	0.30e-4				
16	0.739974976	0.15e-4				
17	0.739967346	0.76e-5				
18	0.739963531	0.38e-5				
19	0.739961624	0.19e-5				
20	0.739962578	0.95e-6				

IDEAL AND NONIDEAL GAS LAWS (CHEMICAL/BIO ENGINEERING)

Background: La legge dei gas perfetti è :

$$pV = nRT$$

dove: p è la pressione, V è il volume, n è il numero di moli, R la costante dei gas universale e T è la temperatura.

Ma: Accurata solo in un range limitato di p, T

Un'equazione alternativa é quella di Van der Waals:

$$\left(p + \frac{a}{v^2}\right) (v - b) = RT$$

dove: $v = V/n$ è il volume molare, a, b sono costanti empiriche dipendenti dal gas

Problema: Per la progettazione di appositi contenitori occorre una stima accurata di v per biossido di carbonio e ossigeno per diverse combinazioni di temperatura e pressione (confrontare i risultati delle due leggi.)

Si hanno i seguenti dati:

$$\begin{array}{l} R = 0.082054 \text{ L atm/(mol K)} \\ \left. \begin{array}{l} a = 3.592 \\ b = 0.04267 \end{array} \right\} \text{ carbon dioxide} \\ \left. \begin{array}{l} a = 1.360 \\ b = 0.03183 \end{array} \right\} \text{ oxygen} \end{array}$$

La progettazione richiede di esaminare i valori di pressione 1, 10, 100 atm in combinazione con le temperature 300, 500, 700 K.

Soluzione:

Usando la legge dei gas perfetti con $n = 1$, $p = 1 \text{ atm}$ and $T = 300 \text{ K}$

$$v = \frac{V}{n} = \frac{RT}{p} = 0.082054 \frac{\text{L atm}}{\text{mol K}} \frac{300 \text{ K}}{1 \text{ atm}} = 24.6162 \text{ L/mol}$$

Questi calcoli sono ripetuti per tutte le temperature e le pressioni richieste.

Temperature, K	Pressure, atm	Molal Volume (Ideal Gas Law), L/mol
300	1	24.6162
	10	2.4616
	100	0.2462
500	1	41.0270
	10	4.1027
	100	0.4103
700	1	57.4378
	10	5.7438
	100	0.5744

Ma per le soluzioni con l'eq. di van der Waals:

$$f(v) = \left(p + \frac{a}{v^2} \right) (v - b) - RT$$

bisogna utilizzare i **metodi numerici**.

È facile calcolare la **derivata prima**:

$$f'(v) = p - \frac{a}{v^2} + \frac{2ab}{v^3}$$

per cui posso usare il **metodo di Newton-Raphson**:

$$v_{i+1} = v_i - \frac{f(v_i)}{f'(v_i)}$$

```
clear
```

```
close all
```

```
R = 0.082054;    a1 = 3.592;    b1 = 0.04267;
```

```
p = [1 10 100];    T = [300 500 700];
```

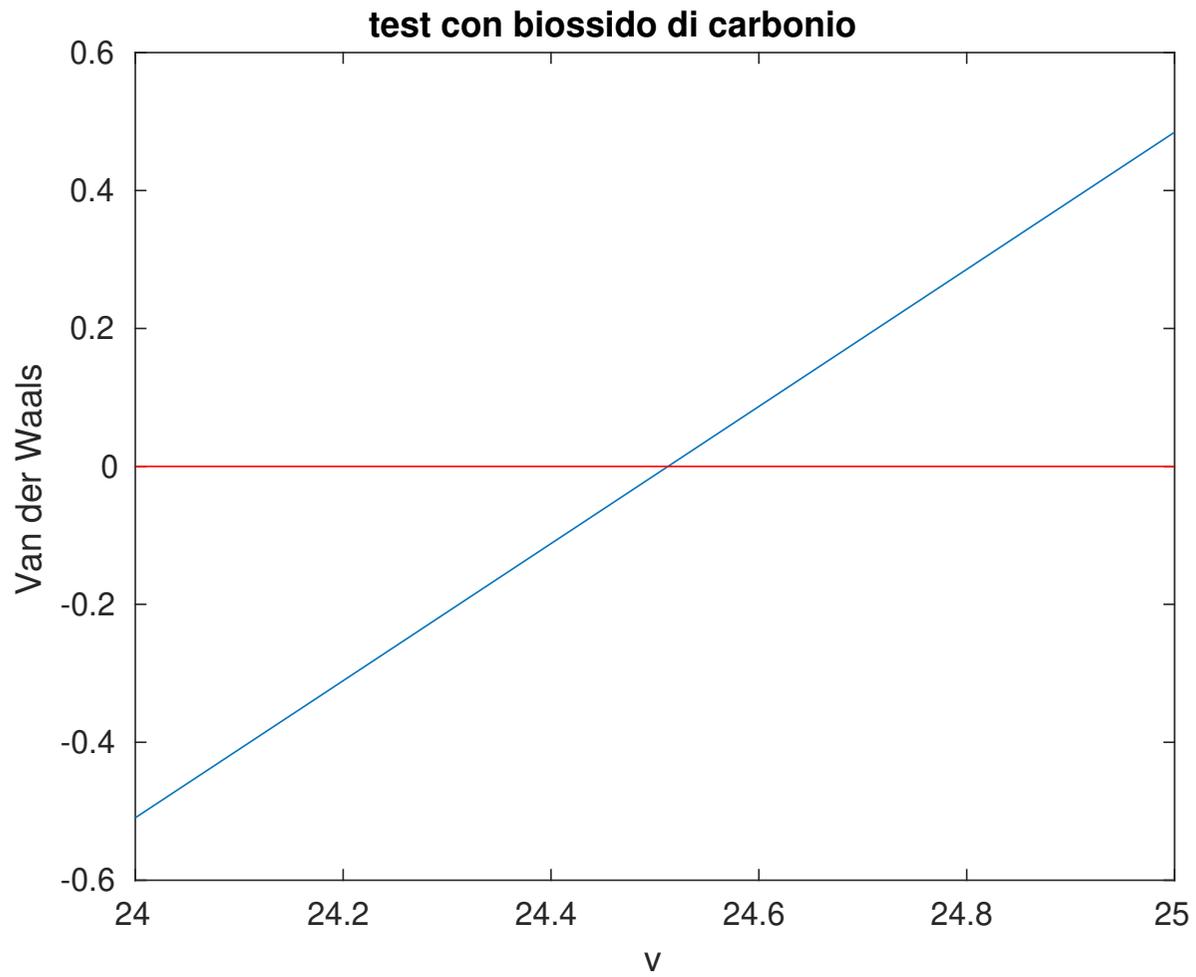
```
v = [24 : 0.0001 : 25];
```

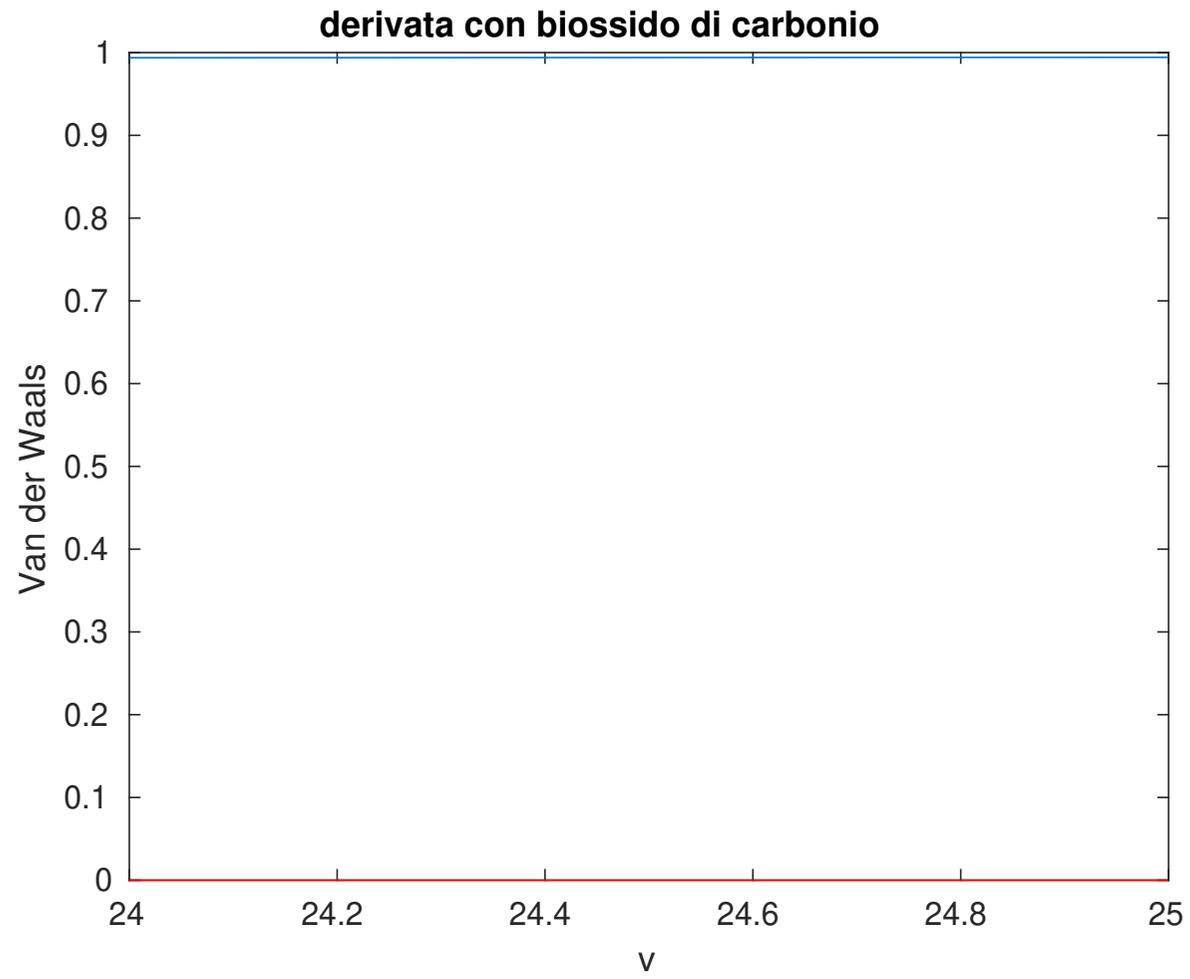
```
f = (p(1) + a1./v.^2). * (v - b1) - R * T(1);
```

```
figure, plot(v,f), hold on,
```

```
plot(v,zeros(1,length(v)),'r')
```

```
xlabel('v'), ylabel('Van der Waals'), title('test con biossido di carbo-  
nio')
```





Es.: Con punto iniziale 24.6162, il volume molare di biossido di carbonio a 300K e 1atm è 24.5126L/mol (dopo solo due iterazioni con un **errore approssimato** ($\epsilon_a = (v_k - v_{k-1})/v_k < 0.001\%$). In generale:

Temperature, K	Pressure, atm	Molal Volume (Ideal Gas Law), L/mol	Molal Volume (van der Waals) Carbon Dioxide, L/mol	Molal Volume (van der Waals) Oxygen, L/mol
300	1	24.6162	24.5126	24.5928
	10	2.4616	2.3545	2.4384
	100	0.2462	0.0795	0.2264
500	1	41.0270	40.9821	41.0259
	10	4.1027	4.0578	4.1016
	100	0.4103	0.3663	0.4116
700	1	57.4378	57.4179	57.4460
	10	5.7438	5.7242	5.7521
	100	0.5744	0.5575	0.5842

Alcune considerazioni:

- 1) La stima dei valori con l'eq. di van der Waals (piú precisa) differisce da quella ottenuta con l'eq. dei gas perfetti: strategica per una corretta progettazione !!!
- 2) La conoscenza della derivata prima ci ha condotto al metodo di Newton-Raphson altrimenti avremmo potuto scegliere altri metodi
- 3) Il metodo di NR però sarebbe da preferire nel caso di una progettazione con calcolo automatico di vari gas a diverse condizioni, rispetto al metodo di bisezione in quanto **piú efficiente** (ad es. 1s per tante stime!)
- 4) Si potrebbe scegliere l'estremo di Fourier?