Risoluzione di equazioni non lineari

Il comando `fzero`

- `x=fzero(fun,x0)` *"tries to find a zero of fun near x0"*.
- `x0` può essere un scalare o un vettore di due componenti.
- Se `x0` e un vettore di due componenti `fzero` assume che `x0` è un intervallo e che il segno di `fun(x0(1))` e diverso del segno di `fun(x0(2))`. Se non è cosi da errore.
- L'algoritmo che usa è una combinazione del metodo di bisezione, il metodo delle secanti e tecniche di interpolazione.
- `fun` è un *"function handle"*.

Esempio

Per trovare una soluzione vicina ad 1 dell'equazione $x^2 - 2 = 0$

```
>> fzero(@(x) x.^2-2,1)
ans =
    1.4142
```

oppure in un M-file scrivo la funzione $f(x) = x^2 - 2$

```
function y=esempio(x)
y=x.^2-2;
```

e nella riga di comandi

```
>> fzero(@esempio,1)
ans =
    1.4142
```

Il metodo di Newton

La seguente funzione implementa il metodo di Newton

```
function [x,nit,xv,flag]=newton(fun,dfun,x)
nmax=100;
toll=1.e-12;
flag=1;
xv=[x];
for nit=1:nmax
    inc=feval(fun,x)/feval(dfun,x);
    x=x-inc;
    xv=[xv x];
    if abs(inc)<toll, flag=0; return, end
end
```

Esercizio

Usando la funzione Newton risolvere

$$\alpha - e^{-\alpha} = 0 \quad \text{e} \quad (\alpha - e^{-\alpha})^2 = 0\,.$$

Che cosa si osserva sulla velocità di convergenza?

Stima dell'ordine di convergenza di un metodo iterativo - I

La successione $\{x^{(k)}\}_{k\geq 0}$ converge ad α con ordine p se

$$\lim_{k\to\infty} \frac{|x^{(k+1)} - \alpha|}{|x^{(k)} - \alpha|^p} = K$$

e $K > 0$.

Per k sufficientemente grande risulta essere

$$|x^{(k+1)} - \alpha| \approx K|x^{(k)} - \alpha|^p$$

ma anche

$$|x^{(k+2)} - \alpha| \approx K|x^{(k+1)} - \alpha|^p \, .$$

Stima dell'ordine di convergenza di un metodo iterativo - II

- Faccendo il rapporto

$$\frac{|x^{(k+2)} - \alpha|}{|x^{(k+1)} - \alpha|} \approx \left(\frac{|x^{(k+1)} - \alpha|}{|x^{(k)} - \alpha|} \right)^p$$

Chiamando $R_k = \frac{|x^{(k+1)} - \alpha|}{|x^{(k)} - \alpha|}$ abbiamo

$$R_{k+1} \approx R_k^p$$

- Calcolando il logaritmo

$$\log R_{k+1} \approx p \log R_k \,.$$

- Quindi

$$p \approx \frac{\log R_{k+1}}{\log R_k} \,.$$

Esercizio

- La seguente funzione stima l'ordine di convergenza di una successione convergente:

```
function p=ordine(v)
err=abs(v-v(end));
R=err(2:end)./err(1:end-1);
p=log(R(2:end-1))./log(R(1:end-2));
```

- Confrontare l'ordine di convergenza del metodo di Newton nel risolvere

$$\alpha - e^{-\alpha} = 0 \qquad \text{e} \qquad (\alpha - e^{-\alpha})^2 = 0\,.$$

Esercizio

- Scrivere una funzione di Matlab che implementi il metodo delle secanti

$$x^{(-1)},\, x^{(0)} \quad \text{assegnati}$$

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

  Fermare le iterazioni quando $|x^{(k+1)} - x^{(k)}| < 10^{-10}$.
- Scrivere una funzione di Matlab che implementi il metodo di bisezione.

Sistemi di equazioni non lineari - Il metodo di Newton

$$\mathbf{F} : \mathbf{R}^n \to \mathbf{R}^n \qquad \mathbf{F}(\alpha) = \mathbf{0}$$

Sia $J(\mathbf{F})(\mathbf{x})$ la matrice jacobiana di $\mathbf{F}$ nel punto $\mathbf{x}$

$$[J(\mathbf{F})(\mathbf{x})]_{i,j} = \frac{\partial F_i}{\partial x_j}(\mathbf{x})$$

$\mathbf{x}^{(0)}$ assegnato
Per $k \geq 0$

sia $\mathbf{z}^{(k)}$ soluzione del sistema lineare $[J(\mathbf{F})(\mathbf{x}^{(k)})]\mathbf{z}^{(k)} = \mathbf{F}(\mathbf{x}^{(k)})$
$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \mathbf{z}^{(k)}$
Se $\|\mathbf{z}^{(k)}\| < toll$ STOP

Metodo di Newton per sistemi

```
function [x,nit,flag]=newtonsistemi(F,JF,x)
nmax=100;
toll=1.e-12;
flag=1;
for nit=1:nmax
    Fx=feval(F,x);
    JFx=feval(JF,x);
    z=JFx\Fx;
    x=x-z;
    if norm(z)<toll, flag=0; return, end
end
```

Esempio

Risolvere i seguenti sistemi di equazioni non lineari

$$\begin{aligned} x_1^2 - 10x_1 + x_2^2 + 8 = 0 \\ x_1 x_2^2 + x_1 - 10x_2 + 8 = 0 \end{aligned}$$

$$\begin{aligned} x_1^2 + x_2 - 37 = 0 \\ x_1 - x_2^2 - 5 = 0 \\ x_1 + x_2 + x_3 - 3 = 0 \end{aligned}$$

$$\begin{aligned} &3x_1 - \cos(x_2 x_3) - \tfrac{1}{2} = 0 \\ &x_1^2 - 81(x_2 + 0.1)^2 + \sin(x_3) + 1.06 = 0 \\ &e^{-x_1 x_2} + 20x_3 + \tfrac{10\pi - 3}{3} = 0 \end{aligned}$$