Sistemi di equazioni non lineari - Il metodo di Newton

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

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
```

Valutazione ciclica della matrice Jacobiana

Per ridurre il costo computazionale si può mantenere fissa la matrice jacobiana per un certo numero di passi $p \geq 2$.

In questo caso si risolvono p sistemi lineari con la stessa matrice quindi può essere conveniente calcolare la fattorizzazione LU di A.

Esercizio

Escrivere il metodo di Newton per sistemi di equazioni non lineari con valutazione della matrice jacobiana ogni p iterazioni. Il numero p deve essere dato in input.

Metodo di Broyden

$\mathbf{x}^{(0)}$ assegnato, $Q^{(0)}$ assegnato, $\mathbf{F}^{(0)} = \mathbf{F}(\mathbf{x}^{(0)})$

Per $k \geq 0$

sia $\mathbf{s}^{(k)}$ soluzione del sistema lineare $Q^{(k)}\mathbf{s}^{(k)} = -\mathbf{F}(\mathbf{x}^{(k)})$

$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \mathbf{s}^{(k)}$

Se $\|\mathbf{s}^{(k)}\| < toll$ STOP

$\mathbf{F}^{(k+1)} = \mathbf{F}(\mathbf{x}^{(k+1)})$

$$Q^{(k+1)} = Q^{(k)} + \frac{\mathbf{F}^{(k+1)} - \mathbf{F}^{(k)} - Q^{(k)}\mathbf{s}^{(k)}}{\|\mathbf{s}^{(k)}\|^2}(\mathbf{s}^{(k)})^T$$

Ma

$$\mathbf{F}^{(k+1)} - \mathbf{F}^{(k)} - Q^{(k)}\mathbf{s}^{(k)} = \mathbf{F}^{(k+1)}$$

```
function [x,nit,flag]=broyden(F,Q,x)
nmax=1000;
toll=1.e-12;
flag=1;
Fx=feval(F,x);
for nit=1:nmax
    s=-Q\Fx;
    x=x+s;
    inc=norm(s);
    if abs(inc)<toll, flag=0; return, end
    Fx=feval(F,x);
    Q=Q+Fx*s'/(s'*s);
end
```

Metodo di Broyden (aggiornando l'inversa)

$\mathbf{x}^{(0)}$ assegnato, $\mathbf{F}^{(0)} = \mathbf{F}(\mathbf{x}^{(0)})$, $H^{(0)}$ assegnato $(= [J(\mathbf{F})(\mathbf{x}^{(0)})]^{-1})$

Per $k \geq 0$

$$
\begin{aligned}
&\mathbf{s}^{(k)} = -H^{(k)}\mathbf{F}^{(k)} \\
&\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \mathbf{s}^{(k)} \\
&\text{Se } \|\mathbf{s}^{(k)}\| < \mathit{toll} \text{ STOP} \\
&\mathbf{F}^{(k+1)} = \mathbf{F}(\mathbf{x}^{(k+1)}) \\
&\mathbf{z}^{(k+1)} = H^{(k)}\mathbf{F}^{(k+1)} \\
&\beta_k = (\mathbf{s}^{(k)})^T\mathbf{s}^{(k)} + (\mathbf{s}^{(k)})^T\mathbf{z}^{(k+1)} \\
&H^{(k+1)} = H^{(k)} - \beta_k^{-1}\mathbf{z}^{(k+1)}(\mathbf{s}^{(k)})^T H^{(k)}
\end{aligned}
$$

```
function [x,nit,flag]=broyden(F,H,x)
nmax=1000;
toll=1.e-12;
flag=1;
Fx=feval(F,x);
for nit=1:nmax
    z=-H*Fx;
    x=x+s;
    inc=norm(s);
    if abs(inc)<toll, flag=0; return, end
    Fx=feval(F,x);
    z=H*Fx;
    beta=s'*s+s'*z;
    H=H-1/beta*z*s'*H;
end
```

Esempio

Risolvere i seguenti sistemi di equazioni non lineari

$$
\begin{aligned}
x_1^2 - 10x_1 + x_2^2 + 8 = 0 \\
x_1x_2^2 + x_1 - 10x_2 + 8 = 0
\end{aligned}
$$

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

$$
\begin{aligned}
& 3x_1 - \cos(x_2x_3) - \tfrac{1}{2} = 0 \\
& x_1^2 - 81(x_2 + 0.1)^2 + \sin(x_3) + 1.06 = 0 \\
& e^{-x_1x_2} + 20x_3 + \tfrac{10\pi - 3}{3} = 0
\end{aligned}
$$