Sistemi nonlineari.

Alvise Sommariva

Università degli Studi di Padova
Dipartimento di Matematica

30 maggio 2012

Cenni alla risoluzione numerica di sistemi nonlineari

Problema. Data una funzione $F : \Omega \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^n$ si tratta di trovare gli $x^* \in \Omega$ tali che $F(x^*) = 0$.
Esempio:

$$\left\{ \begin{array}{l} x - \frac{1}{2}\cos(y) = 0 \\ y - \frac{1}{2}\sin(x) = 0 \end{array} \right. \qquad (1)$$

Soluzione (unica!):

$$x = 0.48640515466592, \;\; y = 0.23372550195872.$$

Generalizzazione punto fisso e Teorema di Banach

Teorema. Sia (X, d) uno spazio metrico completo ed M un sottoinsieme non vuoto e chiuso di X. Se $\phi : M \rightarrow M$ è L contrattiva cioè $d(\phi(\mathbf{x}), \phi(\mathbf{y})) \leq Ld(\mathbf{x}, \mathbf{y})$ con $L < 1$ allora

1. l'equazione $\mathbf{x} = \phi(\mathbf{x})$ ha una ed una sola soluzione $\mathbf{x}^*$;
2. la successione $\mathbf{x_{k+1}} = \phi(\mathbf{x_k})$ (detta di Banach o di punto fisso) converge alla soluzione $\mathbf{x}^*$ per ogni scelta del punto iniziale $\mathbf{x_0}$;
3. per $k = 0, 1, 2, \ldots$ si ha
$$d(\mathbf{x_k}, \mathbf{x}^*) \leq L^k(1-L)^{-1}d(\mathbf{x_0}, \mathbf{x_1}), \text{ a priori}$$
$$d(\mathbf{x_{k+1}}, \mathbf{x}^*) \leq L(1-L)^{-1}d(\mathbf{x_{k+1}}, \mathbf{x_k}), \text{ a posteriori}$$
4. per $k = 0, 1, 2, \ldots$ si ha
$$d(\mathbf{x_{k+1}}, \mathbf{x}^*) \leq Ld(\mathbf{x_k}, \mathbf{x}^*)$$

Punto fisso per sistemi e Teorema di Banach

Teorema. Sia $\mathbb{R}^d$ (dotato della norma euclidea $\|x\| = \sqrt{\sum_{k=1}^{d} x_k^d}$) ed M un sottoinsieme non vuoto e chiuso di $\mathbb{R}^d$. Se $\phi : M \to M$ è L contrattiva cioè $\|\phi(\mathbf{x}) - \phi(\mathbf{y}\| \leq L\|\mathbf{x} - \mathbf{y}\|$ con $L < 1$ allora

1. l'equazione $\mathbf{x} = \phi(\mathbf{x})$ ha una ed una sola soluzione $\mathbf{x}^*$;
2. la successione $\mathbf{x_{k+1}} = \phi(\mathbf{x_k})$ (detta di Banach o di punto fisso) converge alla soluzione $\mathbf{x}^*$ per ogni scelta del punto iniziale $\mathbf{x_0}$;
3. per $k = 0, 1, 2, \ldots$ si ha
$$\|\mathbf{x_k} - \mathbf{x}^*\| \leq L^k (1 - L)^{-1} \|\mathbf{x_0} - \mathbf{x_1}\|, \text{ a priori}$$
$$\|\mathbf{x_{k+1}} - \mathbf{x}^*\| \leq L(1 - L)^{-1} \|\mathbf{x_{k+1}} - \mathbf{x_k}\|, \text{ a posteriori}$$
4. per $k = 0, 1, 2, \ldots$ si ha
$$\|\mathbf{x_{k+1}} - \mathbf{x}^*\| \leq L\|\mathbf{x_k} - \mathbf{x}^*\|$$

Esempio punto fisso per sistemi

Esempio: il sistema (1) è eqv. a risolvere il problema di punto fisso

$$\begin{cases} x = \frac{1}{2}\cos(y) \\ y = \frac{1}{2}\sin(x) \end{cases} \tag{2}$$

che si scrive come $\mathbf{v} = \phi(\mathbf{v})$ con $\mathbf{v} = (x, y)$ e

$$\phi(\mathbf{v}) = \left(\frac{1}{2}\cos(y), \frac{1}{2}\sin(x)\right).$$

Si dimostra che ϕ è L contrazione con $L = 1/2$. Per applicare il teorema di punto fisso per sistemi basta porre

$$M = [-1, 1] \times [-1, 1] \subset \mathbb{R}^2 \text{ (quadrato unitario)}$$

e osservare che

$$\phi(M) \subseteq [-1/2, 1/2] \times [-1/2, 1/2] \subset M$$

in quanto $-1 \leq \cos(x) \leq 1$, $-1 \leq \sin(x) \leq 1$.

L-contrattiva

Mostriamo che $\phi : \mathbb{R}^2 \to \mathbb{R}^2$ è contr. Dall'esp. di Taylor in $\mathbb{R}^d$

$$\phi_i(\mathbf{v}) = \phi_i(\mathbf{v_0}) + (grad(\phi_i)(\xi_\mathbf{i})) \cdot (\mathbf{v} - \mathbf{v_0}),\ \xi_i \in S,\ i = 1, 2$$

con S il segmento che congiunge $\mathbf{v}$ con $\mathbf{v_0}$ e $\phi = (\phi_1, \phi_2)$.
Nel nostro esempio quindi, detta ϕ' la matrice Jacobiana di ϕ

$$(\phi')_{j,k}(\mathbf{v}) = \begin{pmatrix} 0 & (-1/2)\sin(v_1) \\ (+1/2)\cos(v_2) & 0 \end{pmatrix}$$

Essendo $\|A\|_\infty = \max_j \sum_i |a_{i,j}|$ si ha

$$\begin{aligned} \|(\phi')(\mathbf{v})\|_\infty &= \max_j \sum_i |\left((\phi')(\mathbf{v})\right)_{i,j}| \\ &= \max(|(-1/2)\sin(v_1)|, |(+1/2)\cos(v_2)|) \leq \frac{1}{2} \end{aligned}$$

Quindi

$$\|\phi(\mathbf{v}) - \phi(\mathbf{v_0})\|_\infty \leq \|(\phi'(\xi))\|_\infty \|(\mathbf{v} - \mathbf{v_0})\|_\infty,\ \xi \in S$$

da cui $\phi : \mathbb{R}^2 \to \mathbb{R}^2$ è una L-contrazione con $L = 1/2$.

Metodo Punto fisso per sistemi, in Matlab

```
function xhist=punto_fisso(x0, maxit, tol, g)

% INPUT.
% x0      : PUNTO INIZIALE.
% maxit   : NUMERO MASSIMO DI ITERAZIONI.
% tol     : TOLLERANZA CRITERIO DI ARRESTO.
% g       : EQUAZIONE DA RISOLVERE x=g(x)

x_old=x0;
xhist=x_old';
for index=1:maxit
    x_new=feval(g,x_old);
    if norm(x_new-x_old,inf) < tol
        return
   else
      xhist=[xhist; x_new'];
      x_old=x_new;
   end
end
fprintf('\n \t [WARNING]: MAX NUM.ITER.');
```

Metodo Punto fisso per sistemi, in Matlab

```
function res=phi(v)
res(1,1)=0.5*cos(v(2));
res(2,1)=0.5*sin(v(1));
```

Scrivo nella shell di Matlab

```
>> x0=[0; 0]; maxit=1000; tol=10^(-14);
>> xhist=punto_fisso(x0, maxit, tol, 'phi');
>> % NUMERO ITERAZIONI.
>> size(xhist)
ans =
    23     2
>>
```

Esempio punto fisso per sistemi

Successione punto fisso (convergente alla soluzione di (2)):

$$x_{k+1} = \frac{1}{2}\cos(y_k), \quad y_{k+1} = \frac{1}{2}\sin(x_k)$$

x_k	y_k
0.00000000000000	0.00000000000000
0.50000000000000	0.00000000000000
0.50000000000000	0.23971276930210
0.48570310513698	0.23971276930210
0.48570310513698	0.23341513183103
0.48644107261515	0.23341513183103
...	...
0.48640515466592	0.23372550195872

Metodo Newton per sistemi

Supponiamo di dover risolvere $f(\mathbf{v}) = \mathbf{0}$ con $f : \Omega \subseteq \mathbb{R}^d \to \mathbb{R}^d$.

La successione di Newton viene descritta come

$$\begin{cases} f'(\mathbf{v}_k)) \cdot h_{k+1} = -f(\mathbf{v}_k), \\ \mathbf{v_{k+1}} := \mathbf{v_k} + h_{k+1} \end{cases} \tag{3}$$

dove f' è la matrice Jacobiana di f cioè se $\mathbf{v} = (v_1, \ldots, v_d)$ allora

$$(f'(\mathbf{v}_k)))_{i,j} = \frac{\partial f_i(\mathbf{v})}{\partial v_j}.$$

Si noti che ad ogni iterazione si deve risolvere il sistema lineare

$$f'(\mathbf{v}_k)) \cdot h_{k+1} = -f(\mathbf{v}_k).$$

Esempio Newton per sistemi

Consideriamo il precedente esempio, come $f(x) = 0$. Allora posto $\mathbf{v} = (x, y)$ si ha $f(x, y) = (f_1(x, y), f_2(x, y))$ con

$$\begin{aligned} f_1(x, y) &= x - \frac{1}{2}\cos(y) \\ f_2(x, y) &= y - \frac{1}{2}\sin(x). \end{aligned}$$

Inoltre

$$f'(x, y) = \begin{pmatrix} 1 & \frac{1}{2}\sin(y) \\ -\frac{1}{2}\cos(x) & 1 \end{pmatrix} \tag{4}$$

Metodo Newton per sistemi, in Matlab

```
function [v_hist, step_hist, residual_hist]=...
       newton_sistemi(v0,maxit,tol)
v_old=v0; v_hist=[v0'];
Fv=F(v_old);   JF=DF(v_old);
residual=norm(Fv); residual_hist=[residual];
h=-JF\Fv; v_new=v_old+h;
v_hist=[v_hist; v_new']; step_hist=[];
for index=1:maxit
    Fv=F(v_new); residual=norm(Fv,2);
    residual_hist=[residual_hist; residual];
    loc_step=norm(v_new-v_old,2);
    step_hist=[step_hist; loc_step];
    if max(loc_step,residual) < tol
        return;
    else
        v_old=v_new; JF=DF(v_old);  h=-JF\Fv;
        v_new=v_old+h;  v_hist=[v_hist; v_new'];
    end
end
fprintf('\n \t [WARNING]: MAX.NUM.ITER.')
```

Metodo Newton per sistemi, in Matlab

```
function res=F(v)
res(1,1)=0.5*cos(v(2))-v(1);
res(2,1)=0.5*sin(v(1))-v(2);
```

```
function res=DF(v)
res=[-1 -0.5*sin(v(2)); 0.5*cos(v(1)) -1];
```

Scriviamo sulla shell di Matlab

```
>> v0=[0; 0]; maxit=1000; tol=10^(-14);
>> [v_hist, step_hist, residual_hist]=newton_sistemi(v0
    ,maxit,tol);
>> % ITERAZIONI.
>> size(v_hist)
ans =
     6     2
>>
```

Esempio Newton per sistemi

Successione Newton:

$$\begin{cases} f'(\mathbf{v}_k)) \cdot h_{k+1} = -f(\mathbf{v}_k), \\ \mathbf{v_{k+1}} := \mathbf{v_k} + h_{k+1} \end{cases} \tag{5}$$

x_k	y_k
0.0000000000000000	0.0000000000000000
0.5000000000000000	0.2500000000000000
0.4864635133551991	0.2337730769877319
0.4864051551457132	0.2337255025688199
0.4864051546659213	0.2337255019587208
0.4864051546659213	0.2337255019587208

Esercizi in Matlab

Si calcoli la soluzione del sistema nonlineare

$$\begin{aligned} x_1 &= \arctan(x_1 + x_2) \cdot \sin(x_2)/10 \\ x_2 &= \cos(x_1/4) + \sin(x_2/4) \end{aligned} \tag{6}$$

con il metodo di punto fisso e di Newton. Convergono entrambi?
E' utile isolare la soluzione?