CN #34,35 / 2 maggio 2013 / A11

---

* COSTO *

<u>def</u> (costo) il costo del calcolo del valore di una funzione è <u>il numero di op elem effettuate</u>.

<u>Oss</u>: def ragionevole <u>se</u>...

<u>es</u>: $x, y \in \mathbb{R}^n$; $x^T y$ costa... n prodotti + + $(n-1)$ somme ... costo (asintotico) $\sim 2n$.

① Costo EGPP : $\sim \frac{2}{3} n^3$

② Costo SI, SA : $\sim n^2$

③ Costo calcolo "buono" di fatt QR : $\sim \frac{4}{3} n^3$

$\Rightarrow$ costo soluz sist (fatt + SA + SI) :

$\sim \frac{2}{3} n^3$ (EGPP) / $\sim \frac{4}{3} n^3$ (QR)

```
//
// Esempio con matrice di Vandermonde. La matrice ha un numero di
// condizionamento rapidamente crescente con la dimensione del sistema.
//
// Il sistema è risolto in due diversi modi: utilizzando EGPP ed utilizzando
// qr.
//
// Osservazione: la misura relativa dei residui N_RES (paragonabile nei due casi)
// indica che il sistema è stato risolto "bene" dal calcolatore. L'errore
// relativo, rapidamente crescente con la dimensione, è dovuto al pessimo
// condizionamento del sistema.
//
clear;
//
Percorso = "PERCORSO files seguenti";  // <------ *** MODIFICARE ***
exec(Percorso + "EGPP.sci");
exec(Percorso + "SA.sci");
exec(Percorso + "SI.sci");
//
// variabili utili
//
DIM = [];
COND_A = [];
ERR_EG = [];
N_RES_EG = [];
ERR_QR = [];
N_RES_QR = [];
SUP_EG = [];
SUP_QR = [];
//
// ciclo sulla dimensione del sistema
//
for N = 5:2:25,
DIM($+1) = N;
//
// generazione della matrice
//
colonna = linspace(0,1,N)';
A = zeros(N,N);
for k=1:N,
    A(:,k) = colonna.^(k-1);
end;
//
// soluzione esatta e termine noto corrispondente
//
x_esatta = ones(N,1)/sqrt(2);
b = A*x_esatta;
//
// numero di condizionamento della matrice
//
COND_A($+1) = cond(A);
//
// soluzione del sistema con EGPP, SA e SI
//
[S,D,P,info_EGPP] = EGPP(A);
[c_EG,info_SA] = SA(S,P*b,"disabilita");
[x_EG,info_SI] = SI(D,c_EG,"disabilita");
//
// soluzione con qr, moltiplicazione e SI
//
[U,T] = qr(A);
c_QR = U' * b;
[x_QR,info_SI] = SI(T,c_QR,"disabilita");
//
// misura relativa degli errori
//
ERR_EG($+1) = norm(x_EG - x_esatta)/norm(x_esatta);
ERR_QR($+1) = norm(x_QR - x_esatta)/norm(x_esatta);
//
// misura relativa dei residui
```

```
//
N_RES_EG($+1) = norm(A*x_EG - b)/norm(b);
N_RES_QR($+1) = norm(A*x_QR - b)/norm(b);
//
// uso del Teorema di condizionamento
//
SUP_EG($+1) = COND_A($)*N_RES_EG($);
SUP_QR($+1) = COND_A($)*N_RES_QR($);
end;
//
// grafici
//
scf(0);clf(0);
plot(DIM,log10(COND_A),"gs",...
     DIM,log10(SUP_EG),"rs",DIM,log10(ERR_EG),"r+",DIM,log10(N_RES_EG),"rx",...
     DIM,log10(SUP_QR),"bo",DIM,log10(ERR_QR),"b+",DIM,log10(N_RES_QR),"bx");
xlabel("dimensione");
ylabel("logaritmo in base 10 di...");
legend("COND_A","SUP_EG","ERR_EG","N_RES_EG","SUP_QR","ERR_QR","N_RES_QR",2);
title("confronto EGPP , qr");
```

* METODI ITERATIVI PER SISTEMI LINEARI *

dati: $H \in \mathbb{R}^{n \times n}$, $c \in \mathbb{R}^n$, $\gamma \in \mathbb{R}^n$

- $x_0 = \gamma$
- per $k = 1, 2, \dots$ ripeti: $x_k = H x_{k-1} + c$

uscita: quando un opportuno criterio di arresto è verificato, x_k.

Oss: E' il m it ad un pto def della funzione $h: \mathbb{R}^n \to \mathbb{R}^n$ t.c. $h(x) = Hx + c$. Questa f è CONTINUA e q.di: SE la succes generata dal m. it è convergente a $v \in \mathbb{R}^n$, v è PUNTO UNITO di h, ovvero: $v = Hv + c$.

Il m it def da H e c approssima le soluzioni del sistema: $(I-H)x = c$

- Se si vuole appros la soluz del sistema $Ax = b$, occorre che i sist $Ax = b$ e $(I-H)x = c$ siano EQUIVALENTI. In part: $\det(I-H) \neq 0$.

• Per approx la soluz, occorre trovare succes CONVERGENTI ovvero, note H e c, valori iniziali γ che garantiscono la convergenza.

<u>Es.</u> $A = \boxed{\begin{matrix} 1/2 & 0 \\ 0 & -1 \end{matrix}}$, $b = 0$;

$H = I - A = \boxed{\begin{matrix} 1/2 & 0 \\ 0 & 2 \end{matrix}}$, $c = b$

I sistemi $Ax = b$ e $(I-H)x = c$ sono <u>equiv</u>!

• Scelto $\gamma = \begin{pmatrix} \gamma_1 \\ \gamma_2 \end{pmatrix} \in \mathbb{R}^2$, si ha:

$$x_0 = \gamma, \quad x_1 = Hx_0 + c = H\gamma, \quad x_2 = Hx_1 + c = H^2\gamma$$

$$\dots \quad x_k = H^k \gamma = \boxed{\begin{matrix} \frac{1}{2^k} & 0 \\ 0 & 2^k \end{matrix}} \gamma = \begin{bmatrix} \gamma_1/2^k \\ 2^k \gamma_2 \end{bmatrix}$$

• $\boxed{\text{la succes } x_k \text{ è convergente} \iff \gamma_2 = 0}$

e, in tal caso, il lim è la soluz...

→ la succes converge solo per γ particolari!

$\underline{\text{Es}}$: $A = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$, $b = 0$

- $A = 2I + \underbrace{J}_{\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}}$, $(2I + J)x = b \sim 2x = -Jx + b$

$\sim x = -\frac{1}{2}Jx + \frac{1}{2}b \Rightarrow \boxed{H = \begin{pmatrix} 0 & -1/2 \\ -1/2 & 0 \end{pmatrix}, \; c = \frac{1}{2}b}$

- H è diagonalizz.: $H = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} -1/2 & 0 \\ 0 & 1/2 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}^{-1}$

- Scelto $\gamma = \begin{pmatrix} \gamma_1 \\ \gamma_2 \end{pmatrix} \in \mathbb{R}^2$, è

$$x_k = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} -\frac{1}{2^k} & 0 \\ 0 & \frac{1}{2^k} \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}^{-1} \gamma$$

e $\boxed{\forall \gamma_1, \gamma_2 \text{ si ha } \lim_{k \to \infty} x_k = 0.}$

ovvero: la success converge per OGNI γ!