Seconda prova di recupero relativa ai capitoli 1–3 con correzione

---

Esercizio 1 Dimostrare che, se f è una funzione sufficientemente regolare in un punto x assegnato, allora $[f(x+h) - 2f(x) + f(x-h)]/h^2 = f''(x) + O(h^2)$.

Sviluppando in serie di Taylor in x si ottiene:

$$\begin{aligned} f(x+h) &= f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \frac{h^3}{6}f'''(x) + O(h^4), \\ f(x-h) &= f(x) - hf'(x) + \frac{h^2}{2}f''(x) - \frac{h^3}{6}f'''(x) + O(h^4). \end{aligned}$$

Sommando membro a membro, e dividendo per h^2, si ottiene il risultato.

Esercizio 2 Spiegare esaurientemente in cosa consiste il problema della *cancellazione numerica*.

La somma algebrica tra due numeri $x, y \in \mathbb{R}$, non opposti tra loro, ha numero di condizionamento dato da

$$\kappa = \frac{|x| + |y|}{|x+y|}.$$

Nel caso in cui i due addendi abbiano segno discorde, è evidente come κ non sia uniformemente limitato (cosa che avviene nel caso in cui siano concordi, invece). In particolare, se x e y sono quasi opposti il numero di condizionamento può essere arbitrariamente grande. Il malcondizionamento del problema, in questo caso, si manifesta nel fenomeno della *cancellazione numerica* in cui, pur avendo due addendi con tutte le cifre rappresentate esatte, si ottiene un risultato con un numero di cifre significative inferiore, ovvero si osserva la *cancellazione* di cifre significative.

Esercizio 3 Scrivere una *function* Matlab che implementi efficientemente il metodo di Newton per la ricerca di uno zero di una funzione.

```
function [x,flag] = newton( f, f1, x0, tolx, maxit )
%
%   [x,flag] = newton( f, f1, x0, tolx [, maxit] )
%
%             Metodo di Newton per determinare una approssimazione
%             della radice di f(x)=0 con tolleranza (mista) tolx, a
%             partire da x0, entro maxit iterationi (default = 100).
```

```
%               f1 implementa f'(x) mentre in uscita flag vale -1, se
%               la tolleranza non è soddisfatta entro maxit iterate o
%               la derivata si annulla, altrimenti ritorna il numero
%               di iterazioni richieste.
%
if nargin<4, error('numero argomenti insufficienti')
elseif nargin==4, maxit = 100;
end
if tolx<eps, error('tolleranza non idonea'), end
x = x0;
flag = -1;
for i = 1:maxit
   fx  = feval( f, x );
   f1x = feval( f1, x );
   if f1x==0, break, end
   x = x - fx/f1x;
   if abs(x-x0)<=tolx*(1+abs(x0)), flag = i; break
   else, x0 = x;
   end
end
return
```

Esercizio 4 Derivare il metodo di accelerazione di Aitken, spiegando, preliminarmente, a cosa serve.

È noto che il metodo di Newton converge solo linearmente verso radici multiple: il metodo di accelerazione di Aitken serve, in questo caso, a ripristinare la convergenza quadratica. In dettaglio, detto $e_i = x^* - x_i$ l'errore al passo i del metodo di Newton, avendo denotato con x^* la radice a cui il metodo converge, si avrà, in virù della convergenza lineare,

$$e_{i+1} \approx ce_i, \qquad e_i \approx ce_{i-1},$$

essendo c la costante asintotica dell'errore. Pertanto, dividendo membro a membro le due eguaglianze approssimate, si ricava

$$e_{i+1}e_{i-1} \approx e_i^2,$$

ovvero

$$(x^* - x_{i+1})(x^* - x_{i-1}) \approx (x^* - x_i)^2.$$

Ricavando x^*, si ottiene, pertanto,

$$x^* \approx \frac{x_{i+1}x_{i-1} - x_i^2}{x_{i+1} - 2x_i + x_{x-1}} =: x_i^*.$$

È possibile, quindi dimostrare che la successione $\{x_i^*\}$ ottenuta:

1. eseguendo due iterazioni consecutive del metodo di Newton,

2. estrapolando il corrispondente valore x_i^*, e ritornando al punto 1. ripartendo da questa nuova approssimazione,

converge ora quadraticamente alla radice.

Esercizio 5 Sotto quali condizioni una matrice nonsingolare $A \in \mathbb{R}^{n\times n}$ è fattorizzabile LU?

Sia data una matrice $A \in \mathbb{R}^{n\times n}$ nonsingolare, e si denoti con A_k la sua sottomatrice principale di ordine k, ottenuta come intersezione delle prima k righe e colonne di A. Allora esiste la fattorizzazione $A = LU$, con L triangolare inferiore a diagonale unitaria, ed U triangolare superiore, se e solo se $\det(A_k) \neq 0$, $k = 1, \ldots, n$.

Esercizio 6 Definire cosa è una matrice simmetrica e definita positiva, e dimostrare che essa è fattorizzabile LU.

Una matrice $A \in \mathbb{R}^{n\times n}$ è simmetrica e definita positiva se è:

simmetrica: ovvero, $A = A^\top$;

definita positiva: ovvero, $\forall \boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{x} \neq \boldsymbol{0}$, risulta $\boldsymbol{x}^\top A\boldsymbol{x} > 0$.

Per la seconda parte del quesito, vedere i Lemmi 3.10-3.11 ed il Teorema 3.4 del libro di testo.

Esercizio 7 Definire cosa è il *numero di condizione* di una matrice e spiegarne il significato.

Data una matrice nonsingolare $A \in \mathbb{R}^{n\times n}$ ed i sistemi lineari

$$A\boldsymbol{x} = \boldsymbol{b}, \qquad (A + \Delta A)(\boldsymbol{x} + \Delta\boldsymbol{x}) = \boldsymbol{b} + \Delta\boldsymbol{b},$$

essendo il secondo una perturbazione del primo, si ottiene, utilizzando una qualunque norma su vettore e la corrispondente norma indotta su matrice,

$$\frac{\|\Delta\boldsymbol{x}\|}{\|\boldsymbol{x}\|} \lesssim \kappa(A)\left(\frac{\|\Delta\boldsymbol{b}\|}{\|\boldsymbol{b}\|} + \frac{\|\Delta A\|}{\|A\|}\right).$$

Nella precedente diseguaglianza

$$\kappa(A) = \|A\| \cdot \|A^{-1}\|$$

è il numero di condizione della matrice A che, pertanto, misura il condizionamento di un sistema lineare avente A come matrice dei coefficienti.

Esercizio 8 Scrivere una *function* Matlab che, avendo in ingresso una matrice quadrata A, restituisce una matrice ed un vettore contenenti l'informazione della fattorizazzione LU con pivoting parziale di A.

```
function [LU,p] = fattlu( A )
%
%    [LU,p] = fattlu( A )
%
%                scrive la matrice LU con l'informazione dei fattori L ed U
%                della fattorizzazione P*A = L*U.  L'informazione della
%                matrice di permutazione P è contenuta nel vettore p.
%
[m,n] = size(A);
if m~=n, error('matrice non quadrata'); end
LU = A;
p   = [1:n]';
for i = 1:n-1
    [mi,ki] = max( abs( LU(i:n,i) ) );
    if mi==0, error('matrice singolare'), end
    ki = ki+i-1;
    if ki>i
       LU([i ki],:) = LU([ki i],:);
       p([i ki])      = p([ki i]);
    end
    LU(i+1:n,i) = LU(i+1:n,i)/LU(i,i);
    LU(i+1:n,i+1:n) = LU(i+1:n,i+1:n) -LU(i+1:n,i)*LU(i,i+1:n);
end
return
```

Esercizio 9 Spiegare cosa si intende per fattorizzazione QR di una matrice $A \in \mathbb{R}^{m\times n}$, con $m > n$ ed avente rango massimo. Spiegare l'utilizzo di questa fattorizzazione per risolvere il sistema lineare $A\boldsymbol{x} = \boldsymbol{b}$ nel senso dei minimi quadrati.

Una matrice A come nel testo dell'esercizio è fattorizzabile nella forma $A = QR$, con $Q \in \mathbb{R}^{m \times m}$, $Q^\top Q = I$, e

$$R = \begin{pmatrix} \hat{R} \\ O \end{pmatrix},$$

con $\hat{R} \in \mathbb{R}^{n \times n}$ triangolare superiore e nonsingolare, e $O \in \mathbb{R}^{(m-n) \times n}$. Questa fattorizzazione può essere convenientemente utilizzata, tenendo conto del fatto che la norma euclidea di un vettore non cambia se questo è moltiplicato per una matrice ortogonale, per risolvere il sistema lineare

$$A\boldsymbol{x} = \boldsymbol{b},$$

con $\boldsymbol{x} \in \mathbb{R}^n$ e $\boldsymbol{b} \in \mathbb{R}^m$, nel senso dei minimi quadrati, ovvero determinando la (unica soluzione) $\boldsymbol{x} = \boldsymbol{x}^*$ che minimizza la norma euclidea del corrispondente vettore residuo,

$$\|\boldsymbol{r}\|_2^2 = \|A\boldsymbol{x} - \boldsymbol{b}\|_2^2.$$

Per i dettagli, vedere il libro di testo a pagina 68.
