Prova d'esame di Laboratorio di Calcolo I
per il corso di laurea in Matematica
20 Giugno 2013

Tema d'esame: Calcolo degli autovalori e dell'inversa di una matrice i cui corrispondenti *cerchi di Gershgorin* sono tutti disconnessi tra loro. La matrice inversa viene determinata risolvendo degli opportuni sistemi di equazioni lineari tramite il *metodo di Jacobi.*

Descrizione del metodo di calcolo

I coefficienti $\{a_{i,j}\}_{i,j=0,\ldots,n-1}$ di una matrice *simmetrica* A, cioè

$$A = \begin{pmatrix} a_{0,0} & \cdots & a_{0,n-1} \\ \vdots & \vdots & \vdots \\ a_{n-1,0} & \cdots & a_{n-1,n-1} \end{pmatrix} \quad , \tag{1}$$

sono tali che

$$|a_{i,j}| \le \frac{1}{2n} \qquad \forall\ 0 \le i < j \le n-1 \ , \tag{2}$$

mentre gli elementi diagonali sono semplicemente definiti come segue:

$$a_{i,i} = i + 1 \qquad \forall\ 0 \le i \le n-1 \ . \tag{3}$$

Sia

$$R_i = \sum_{\substack{0 \le j < n \\ j \ne i}} |a_{i,j}| \qquad \forall\ 0 \le i \le n-1 \ , \tag{4}$$

allora le formule (1)–(4) implicano che i cerchi di Gershgorin (nel piano complesso) sono tutti disgiunti tra loro, ovvero $\mathcal{B}_{R_i}(i+1) \cap \mathcal{B}_{R_j}(j+1) = \emptyset \ \ \forall\ 0 \le i < j \le n-1$. In questo contesto, il secondo teorema di Gershgorin ci garantisce che gli n autovalori $\{\lambda_i\}_{i=0}^{n-1}$ possono essere facilmente localizzati grazie alla seguente relazione di inclusione:

$$\lambda_i \in \left(i + \frac{1}{2} \, , \, i + \frac{3}{2} \right) \qquad \forall\ 0 \le i \le n-1 \ , \tag{5}$$

dove è stata utilizzata anche la disuguaglianza $\max_{0 \le i \le n-1}\{R_i\} < 1/2$, la quale segue banalmente dalle formule (2) e (4). Siccome è ben noto che le matrici simmetriche ammettono n autovalori reali che sono le soluzioni dell'equazione che determina gli "zeri" per il *polinomio caratteristico* di A, cioè

$$\mathcal{P}(\lambda) = 0 \ , \quad \text{dove} \quad \mathcal{P}(\lambda) = \det\big(A - \lambda \mathbf{I}\big) \ , \tag{6}$$

allora dalla (5) segue che

$$\mathcal{P}(i + 1/2) \cdot \mathcal{P}(i + 3/2) < 0 \qquad \forall\ 0 \le i \le n-1 \ . \tag{7}$$

In altri termini, siccome il polinomio caratteristico assume valori con segni opposti agli estremi dell'intervallo $[i + 1/2 \, , \, i + 3/2]$, allora l'$i$–esimo autovalore λ_i può essere determinato applicando il metodo di bisezione all'equazione $\mathcal{P}(\lambda) = 0$ proprio su quello stesso intervallo.

Per poter determinare i coefficienti $\{\alpha_{i,j}\}_{i,j=0,\ldots,n-1}$ della matrice *inversa* di A, cioè

(8) $$A^{-1} = \begin{pmatrix} \alpha_{0,0} & \ldots & \alpha_{0,n-1} \\ \vdots & \vdots & \vdots \\ \alpha_{n-1,0} & \ldots & \alpha_{n-1,n-1} \end{pmatrix} \ ,$$

si può procedere risolvendo n sistemi lineari che si possono scrivere, per ogni fissato $l = 0, \ldots, n-1$, nel modo seguente:

$$\sum_{j=0}^{n-1} a_{i,j}\alpha_{j,l} = \delta_{i,l} \qquad \forall\ 0 \le i \le n-1 \ ,$$

dove $(\alpha_{0,l}\,, \ldots\,, \alpha_{n-1,l})^T$ è il vettore incognito e il simbolo $\delta_{i,l}$ denota la ben nota *delta di Kronecker*, cioè

(9) $$\delta_{i,l} = \begin{cases} 1 & \text{se} \quad i = l \ ; \\ 0 & \text{se} \quad i \neq l \ . \end{cases}$$

Siamo quindi ricondotti al tipico problema lineare, cioè il seguente sistema di n equazioni:

(10) $$A\mathbf{x} = \mathbf{b}$$

dove $\mathbf{b} \in \mathbf{R}^n$ è il cosiddetto "vettore noto", mentre $\mathbf{x} \in \mathbf{R}^n$ è quello incognito. Quando la matrice A soddisfa le formule (1)–(3), allora gli elementi diagonali sono "dominanti" rispetto a quelli non diagonali e siamo nelle condizioni ideali per applicare il *metodo di Jacobi*. Tale metodo permette di risolvere il problema lineare del tipo (10) per approssimazioni successive, applicando l'algoritmo che è descritto dalle seguenti prescrizioni le quali sono opportunamente adattate al presente contesto.

(i) Si fissi una "soglia di tolleranza sull'errore relativo" $\sigma > 0$.

(ii) Siano l'approssimazione iniziale $\mathbf{x}^{\sim}$ e le matrici D e S (che rispettivamente sono diagonale, la prima, e simmetrica con elementi diagonali nulli, la seconda) tali che

$$\mathbf{x}^{\sim} = \mathbf{b} \ , \qquad D = \mathrm{diag}\big\{a_{0,0}\,, \ldots\,, a_{n-1,n-1}\big\} \ , \qquad S = D - A \ .$$

(iii) Si eseguano ripetutamente le seguenti istruzioni (iv)–(vi) mentre è verificata la condizione espressa al punto (vii).

  (iv) Si aggiorni la "vecchia approssimazione" $\mathbf{x}^{\star}$ in modo tale che $\mathbf{x}^{\star} = \mathbf{x}^{\sim}$.

  (v) Si determini il vettore "temporaneo" $\mathbf{c}$ in modo tale che $\mathbf{c} = S\mathbf{x}^{\sim}$.

  (vi) Si aggiorni la "nuova approssimazione" $\mathbf{x}^{\sim}$ in modo tale che

$$x_i^{\sim} = \frac{b_i + c_i}{D_{i,i}} \qquad \forall\ 0 \le i \le n-1 \ .$$

(vii) Se è soddisfatta la seguente condizione sull'errore relativo

$$\frac{|\mathbf{x}^{\sim} - \mathbf{x}^{\star}|}{|\mathbf{x}^{\sim}|} > \sigma \ ,$$

allora si torni al punto (iv).

Ovviamente, nella disuguaglianza espressa al punto (vii), la quale esprime la condizione di permanenza nel ciclo (associato al metodo di Jacobi), si intende che $|\cdot|$ indica la norma così definita:

$$|\mathbf{v}| = \sum_{j=0}^{n-1} |v_j| \qquad \forall\ \mathbf{v} \in \mathbf{R}^n \ .$$

Quando l'algoritmo descritto dai punti (i)–(vii) è concluso, cioè quando avviene che $|\mathbf{x}^{\sim} - \mathbf{x}^{\star}|/|\mathbf{x}^{\sim}| \le \sigma$, allora si intende che la soluzione $\mathbf{x} \simeq \mathbf{x}^{\sim}$ è determinata a meno di un errore relativo (calcolato rispetto all'approssimazione precedente) che è *non maggiore della "soglia di tolleranza"* σ.

Obiettivo (intermedio) 1:
si scriva un programma in linguaggio **C** che, dopo aver definito una matrice A che soddisfa le richieste espresse nelle formule (1)–(3), *verifica che gli autovalori di A possono essere determnati applicando il metodo di bisezione all'equazione associata al suo polinomio caratteristico.* In altri termini, tale programma deve verificare numericamente la correttezza della formula (7). *Il programma deve contenere*:

(A) le *function* che sono necessarie per il calcolo del determinante di una matrice;

(B) una *function* che restituisce il valore del polinomio caratteristico $\mathcal{P}(\lambda) = \det(A - \lambda\mathbf{I})$ e ha tre argomenti: una matrice A, il valore della variabile λ e il valore della dimensione effettiva n; tale *function*, *al suo interno* definisce una matrice temporanea, i cui coefficienti devono essere tali che essa sia uguale a $A - \lambda\mathbf{I}$; inoltre, tale *function* esegue il calcolo di $\det(A - \lambda\mathbf{I})$, tramite un'opportuna chiamata ad una delle function richieste al punto (A), e infine restituisce il valore così calcolato all'ambiente chiamante;

(C) la *main function* che deve essere strutturata in modo tale che, a sua volta, essa contenga:

- (C1) l'*input da tastiera* del valore del numero n di righe e di colonne della matrice A; *qualora non sia verificato* che $2 \le n \le NDIM$, dove $NDIM$ è la massima dimensione possibile, allora il valore di n dovrà *essere reinserito correttamente*;
- (C2) la definizione degli elementi diagonali della matrice A, in modo che sia soddisfatta l'equazione (3);
- (C3) la definizione degli elementi non diagonali della matrice A, in modo tale che i coefficienti $a_{i,j}$ (quando $i \neq j$) costituiscano una successione pseudocasuale di numeri compresi nell'intervallo $[-1/(2n), 1/(2n)]$ e, quindi, soddisfino la disequazione (2);
- (C4) la stampa sul video dei coefficienti della matrice A in modo che sia *esteticamente non disprezzabile*;
- (C5) un ciclo che deve essere ripetuto, mentre l'indice i va da 0 a $n-1$, e che effettua le seguenti operazioni (C51)–(C53):
- (C51) il calcolo del valore $\mathcal{P}(\lambda) = \det(A - \lambda\mathbf{I})$ del polinomio caratteristico quando $\lambda = i + 1/2$, grazie a un'opportuna *chiamata della function descritta al punto (B)*;
- (C52) il calcolo del valore $\mathcal{P}(\lambda) = \det(A - \lambda\mathbf{I})$ del polinomio caratteristico quando $\lambda = i + 3/2$, grazie a un'opportuna *chiamata della function descritta al punto (A)*;
- (C53) la stampa sul video dei valori richiesti ai punti (C51)–(C52), in modo che essa sia *esteticamente non disprezzabile* e che renda evidente che tali valori hanno segni opposti.

Alcuni consigli

È sicuramente comodo (e *prudente*) utilizzare delle *functions* o parti di programma, che sono incluse in altri programmi precedentemente scritti dagli studenti stessi o dal docente (e reperibili in rete).
È sicuramente opportuno stabilire il valore *costante* di $NDIM$ per mezzo di una direttiva `#define`.
La scrittura del programma è enormemente facilitata se si ricorre al seguente "trucco": quando si deve decidere quanta parte della memoria deve essere destinata ad ospitare i vari elementi dei vettori e delle matrici, si effettuino dei "sovradimensionamenti" in modo tale da allocare sempre, rispettivamente, $NDIM$ e $NDIM \times NDIM$ celle di memoria. Ciò nonostante, le istruzioni che permettono il calcolo dei valori dei componenti dei vettori e delle matrici verranno effettuate tenendo conto che la *dimensione effettiva* dello spazio vettoriale che stiamo considerando è $n \leq NDIM$.
Al fine di poter effettuare dei controlli di correttezza del programma, può essere utile riprodurre esattamente gli stessi risultati ottenuti dal programma scritto dal docente (si ricorda che tali risultati sono messi a disposizione in rete). A questo scopo, si raccomanda di *non* usare la *function* `srand` e di utilizzare opportunamente la *function* `rand` una sola volta all'interno di ciascuna iterazione del doppio ciclo annidato, il quale viene ripetuto in modo da definire gli elementi sovradiagonali di A, mentre l'indice i va da 0 a $n-1$ e l'indice j va da $i+1$ a $n-1$.

Obiettivo (intermedio) 2:

si integri il programma richiesto dall'obiettivo 1, in modo tale da calcolare tutti gli autovalori della matrice A, utilizzando il metodo di bisezione. A tal fine si proceda come segue:

(A) si scriva una *function* che ha cinque argomenti: gli estremi a e b di un intervallo, la matrice A, il valore della dimensione effettiva n e, infine, un *puntatore a una function g*, la quale, a sua volta, dipenderà da altri tre argomenti, che sono una matrice $NDIM \times NDIM$–dimensionale di numeri di tipo `double`, una variabile di tipo `double` e un'altra di tipo `int`; *(alla fine della chiamata)* tale *function* deve restituire il valore approssimato della soluzione dell'equazione $g(x) = 0$, quando il secondo argomento x della *function g* può variare nell'intervallo (a, b); *al suo interno*, tale *function* deve determinare la soluzione iterando *il metodo di bisezione mentre l'errore relativo* $|(b-a)/(2x)|$ *è superiore o uguale a* 10^{-15};

(B) all'interno della *main function*, si scriva un ciclo che deve essere ripetuto, mentre l'indice i va da 0 a $n-1$, e che effettua le seguenti operazioni (B1)–(B2):
   - (B1) ricerca della soluzione dell'equazione $\mathcal{P}(\lambda) = \det(A - \lambda \mathbf{I}) = 0$ nell'intervallo $(i + 1/2, i+3/2)$, grazie a *un'opportuna chiamata della function descritta al punto (A)*; inoltre, *la suddetta chiamata della function descritta al punto (A)* deve essere tale che, tra gli argomenti, viene passato anche l'indirizzo della *function* descritta al punto (B) dell'obiettivo 1;
   - (B2) la stampa su video del valore calcolato dell'i–esimo autovalore λ_i.

Obiettivo (intermedio) 3:

si integri il programma richiesto dall'obiettivo 2, in modo tale da determinare la soluzione $\mathbf{x}$ del sistema lineare (10), quando $\mathbf{b}$ è il primo versore della base fondamentale, cioè nel caso $\mathbf{b} = (1, 0, \dots, 0)$. A tal fine si proceda come segue:

(A) si scriva una *function* che ha cinque argomenti: una matrice A , un vettore noto $\mathbf{b}$, il valore della dimensione effettiva n , il valore della soglia σ sull'errore relativo e un altro vettore $\mathbf{x}$; tale *function* restituisce (attraverso il quinto argomento) la soluzione $\mathbf{x}$ del sistema lineare $A\mathbf{x} = \mathbf{b}$; all'interno di tale *function*, la soluzione deve essere (approssimativamente) determinata traducendo in linguaggio **C** il *metodo di Jacobi* così come è stato riassunto dalle prescrizioni (ii)–(vii) alla fine dell'introduzione, che descrive il metodo di calcolo.

(B) si includa nel programma una *function* che ha quattro argomenti: una matrice A , un vettore $\mathbf{v}$, il valore della dimensione effettiva n e un vettore $\mathbf{w}$; tale *function* restituisce (attraverso il quarto argomento) il vettore risultato del seguente prodotto matrice per vettore $\mathbf{w} = A\mathbf{v}$;

(C) all'interno della *main function*, si definisca un vettore $\mathbf{e}$ in modo tale che sia $\mathbf{e} = (1, 0, \dots, 0)$; successivamente, con *un'opportuna chiamata della function descritta al punto (A)* si calcoli la soluzione numerica del sistema $A\mathbf{x} = \mathbf{e}$, in modo tale che l'errore relativo sia non superiore a 10^{-15}; inoltre, si stampino sul video le componenti del vettore $\mathbf{x}$;

(D) all'interno della *main function*, si calcoli il vettore $\mathbf{c} = A\mathbf{x} - \mathbf{e}$; tale calcolo deve essere effettuato anche grazie a un'opportuna *chiamata della function descritta al punto (B)*;

(E) all'interno della *main function*, si effettui una stampa sul video del valore di $\sum_{j=0}^{n-1} |c_j|$ *in formato esponenziale*; ovviamente, la determinazione del vettore incognito $\mathbf{x}$ *deve considerarsi ben riuscita* se $\sum_{j=0}^{n-1} |c_j|$ è circa dell'ordine di grandezza dell'*errore di macchina.*

Obiettivo (finale) 4:

si integri il programma richiesto dall'obiettivo 3, in modo tale da determinare gli elementi della matrice inversa A^{-1}, per poi controllare che i suoi autovalori sono effettivamente λ_0^{-1} , ... , λ_{n-1}^{-1} . A tal fine si proceda come segue:

(A) all'interno della *main function*, si scriva un ciclo che deve essere ripetuto, mentre l'indice j va da 0 a $n - 1$, e che effettua le seguenti operazioni (A1)–(A3):

   (A1) si ridefinisca il vettore $\mathbf{e}$, in modo tale che sia uguale al j–esimo versore della base fondamentale, cioè $\mathbf{e} = (\delta_{0,j}\ , \dots ,\ \delta_{n-1,j})$ (con il simbolo $\delta_{i,j}$ definito dalla (9));

   (A2) per mezzo di un'opportuna *chiamata della function descritta al punto (A) dell'obiettivo 3*, si ricalcoli il vettore $\mathbf{x}$ in modo che esso sia uguale alla soluzione numerica del sistema $A\mathbf{x} = \mathbf{e}$;

   (A3) si ponga la j–esima colonna della matrice inversa A^{-1} uguale a $\mathbf{x}$, cioè, in riferimento alla formula (8), si deve far sì che $\alpha_{i,j} = x_i\ \forall\ i = 0, \dots, n-1$;

(B) all'interno della *main function*, per ogni valore dell'indice i va da 0 a $n-1$ e per mezzo di un'opportuna *chiamata della function descritta al punto (B) dell'obiettivo 1*, si stampi il valore di $\det(A^{-1} - \lambda_i^{-1}\mathbf{I})$ *in formato esponenziale*; ovviamente, la determinazione della matrice inversa A^{-1} *deve considerarsi ben riuscita* se tutti i suddetti valori di $\det(A^{-1} - \lambda_i^{-1}\mathbf{I})$ sono circa dell'ordine di grandezza dell'*errore di macchina.*