

Laboratorio di Calcolo Scientifico - a.a. 2005/06

HOMEWORK

prof.ssa Rossana Vermiglio, dott. Dimitri Breda
Dipartimento di Matematica e Informatica
Università degli Studi di Udine
vermiglio,dbreda@dimi.uniud.it
<http://www.dimi.uniud.it/rossana,dbreda>

Il candidato o i candidati (max 3 persone per gruppo) devono svolgere i seguenti esercizi presentando una sintetica relazione scritta che contenga le risposte ai quesiti posti ed un'analisi critica dei risultati numerici, allegando anche i codici Matlab sviluppati ed opportunamente commentati, nonché i risultati e gli eventuali grafici richiesti. La relazione, in formato **cartaceo** e completa di codici, risultati e grafici, deve contenere **nome, cognome, numero di matricola** e **anno di corso** dei componenti il gruppo di lavoro.

DATA DI CONSEGNA: 18 novembre 2005

DATA DI RITIRO: 29 novembre 2005

1 Interpolazione di Hermite e Spline. Il polinomio cubico $\varphi(t)$ di Hermite è un polinomio che soddisfa alle seguenti condizioni di interpolazione:

$$\varphi(t_0) = y_0, \varphi(t_1) = y_1, \varphi'(t_0) = d_0, \varphi'(t_1) = d_1.$$

1.1 Determina i coefficienti $h_0(t)$, $h_1(t)$, $\tilde{h}_0(t)$, $\tilde{h}_1(t)$ che soddisfano alle condizioni:

$$\begin{aligned} h_j(t_i) &= 1 \text{ se } i = j, 0 \text{ altrimenti,} \\ h'_j(t_i) &= 0 \\ \tilde{h}_j(t_i) &= 0 \\ \tilde{h}'_j(t_i) &= 1 \text{ se } i = j, 0 \text{ altrimenti,} \end{aligned}$$

per $i, j \in \{0, 1\}$ e forniscono la seguente rappresentazione del polinomio di Hermite:

$$\varphi(t) = y_0 h_0(t) + y_1 h_1(t) + d_0 \tilde{h}_0(t) + d_1 \tilde{h}_1(t), \quad t \in [t_0, t_1].$$

1.2 Assegnati nei nodi di interpolazione t_i , $i = 0, \dots, m$, i valori y_i , d_i , $i = 0, 1, \dots, m$, scrivi un codice Matlab che calcoli i valori assunti dal polinomio interpolante di Hermite a tratti $H(t)$ in corrispondenza a valori di $t \in [t_0, t_m]$ assegnati e ne disegni il grafico.

1.3 Sia data la funzione

$$f(t) = \frac{1}{2}e^t \sin(3t) - \log(t), \quad t \in [1, 4].$$

Scelti nell'intervallo $[1, 4]$ i nodi di interpolazione equidistanti t_i , $i = 0, \dots, m$, ed i valori $y_i = f(t_i)$, $d_i = f'(t_i)$ per $i = 0, \dots, m$, calcola, per diversi valori di m , una stima dell'errore $\max_{t_0 \leq t \leq t_m} |f(t) - H(t)|$ e $\max_{t_0 \leq t \leq t_m} |f'(t) - H'(t)|$, dove $H(t)$ $t \in [t_0, t_m]$, è il polinomio di Hermite a tratti. Per i diversi valori di m , confronta le stime degli errori con quelle ottenute utilizzando la Spline cubica interpolante $S_3(t)$ del Matlab sugli stessi nodi di interpolazione e con $y_i = f(t_i)$, $i = 0, \dots, m$.

Suggerimento. Per la valutazione della derivata della spline $S_3(t)$, usa la sua pp-rappresentazione, i.e. `pp=spline(t,y)`. I coefficienti $C_{i,j}$, $j = 1, \dots, k = 4$, del polinomio $C_{i,1}(t - t_i)^3 + C_{i,2}(t - t_i)^2 + C_{i,3}(t - t_i) + C_{i,4}$ che descrivono la spline nell' i -esimo intervallo si ottengono con il comando `[t,C,m,k]=unmkpp(pp)`. Calcolati i coefficienti C_{der} del derivato, la sua pp-rappresentazione si ottiene con l'istruzione `ppder=mkpp(t,Cder)`. Per la valutazione usa il comando `ppval`.

1.4 Se si hanno a disposizione solo y_i , $i = 0, \dots, m$, i valori d_i , $i = 0, \dots, m$, si possono ottenere usando le seguenti formule:

per $i = 0, \dots, m - 1$:

$$h_i = t_{i+1} - t_i,$$

$$r_i = \frac{y_{i+1} - y_i}{h_i};$$

per $i = 1, \dots, m - 1$:

$$d_i = 0 \text{ se } r_i, r_{i-1} \text{ hanno segno opposto o sono nulli, altrimenti}$$

$$w_{i,1} = 2h_i + h_{i-1}, \quad w_{i,2} = h_i + 2h_{i-1},$$

$$d_i = \frac{\frac{w_{i,1} + w_{i,2}}{r_{i-1}} + \frac{w_{i,2}}{r_i}}{r_{i-1} + r_i};$$

$$d_0 = \frac{(2h_0 + h_1)r_0 - h_0r_1}{h_0 + h_1},$$

$d_0 = 0$ se d_0, r_0 hanno segno opposto;

$$d_m = \frac{(2h_{m-1} + h_{m-2})r_{m-1} - h_{m-1}r_{m-2}}{h_{m-1} + h_{m-2}},$$

$d_m = 0$ se d_m, r_{m-1} hanno segno opposto.

Assegnato un insieme discreto di punti P_i , $i = 0, \dots, m$, su di una curva $P(t) = (x(t), y(t))$ nel piano, costruisci l'interpolante in forma parametrica con polinomio di Hermite a tratti e con funzioni Spline e confronta graficamente i risultati. Come esempio test per il codice considera la curva, nota come *spirale di Archimede*, definita in modo parametrico

$$x(t) = t \cos(t), \quad y(t) = t \sin(t), \quad t \in [0, 6\pi].$$

scegliendo su di essa l'insieme discreto di punti individuati da $m + 1$ nodi equidistanti t_i , $i = 0, 1, \dots, m = 20$, in $[0, 6\pi]$. Con la stessa tecnica, prova a ricostruire delle curve a tua scelta selezionando opportunamente dei punti su di esse. Ad esempio prova a ricostruire alcune lettere corsive dell'alfabeto oppure il profilo della tua mano. Per ottenere i dati puoi procedere come segue:

- * riporta su lucido la curva che vuoi ricostruire;
- * fissa il lucido sullo schermo del computer;
- * scegli con il mouse alcuni punti lungo la curva con i seguenti comandi:

```
figure('position',get(0,'screensize'))
axes('position',[0 0 1 1])
[x,y]=ginput;
```

La raccolta dei punti con il comando *ginput* viene terminata premendo *invio*.

2 Minimi quadrati. Sia assegnata una serie temporale vettoriale, $\tilde{Y}(t) \in \mathbb{R}^n$, $t = 0, 1, \dots, T$. Le n componenti $Y_i(t)$, $i = 1, \dots, n$, rappresentano delle misurazioni di diverse quantità al tempo t . Per prevedere alcuni degli elementi successivi della serie temporale, i.e. $Y(t)$, $t \geq T + 1$, costruiamo il seguente modello lineare

$$Y(t) = AY(t-1), \quad t = 1, 2, \dots,$$

con $A \in \mathbb{R}^{n \times n}$. La matrice A che definisce il modello viene determinata secondo il criterio dei minimi quadrati, minimizzando rispetto a tutte le matrici A la media dell'errore, i.e.

$$\bar{E} = \frac{1}{T} \sum_{t=1}^T \|e(t)\|_2^2,$$

dove $e(t) = \tilde{Y}(t) - A\tilde{Y}(t-1)$, $t = 1, \dots, T$.

Il file *mibtel.mat* contiene la matrice di dati $YY = (YY(t, i)) \in \mathbb{R}^{101 \times 4}$. Le colonne rappresentano le serie temporali relative all'indice Mibtel (dal 2 gennaio al 28 maggio 1997)rispettivamente $YY(:, 1)$ di apertura, $YY(:, 2)$ di massimo, $YY(:, 3)$ di minimo e $YY(:, 4)$ di chiusura. Scrivi un codice Matlab che calcoli la matrice A del modello per la serie assegnata, la media dell'errore relativo risultante:

$$\bar{E}_{rel} = \frac{T\bar{E}}{\sum_{t=1}^T \|\tilde{Y}(t)\|_2^2}$$

ed i valori $Y(T+1)$, $Y(T+2)$. Rappresenta graficamente l'andamento delle varie componenti sia della serie temporale che dell'errore.

3 Interpolazione razionale. Consideriamo la seguente funzione razionale di grado n :

$$\varphi(t) = \frac{a_0 + a_1 t + \dots + a_n t^n}{1 + b_1 t + \dots + b_n t^n}$$

con $a_n \neq 0, b_n \neq 0$. Assegnati i dati $(t_i, y_i), i = 0, \dots, m = 2n$, il problema consiste nel calcolare la funzione razionale di grado n che interpola i dati.

Scrivi un codice Matlab che risolve il problema sopra descritto utilizzando i dati $y_i = f(t_i), i = 0, \dots, m$, ottenuti valutando la funzione

$$f(t) = \frac{e^t}{10t \tan(t)}, \quad t \in [0.1, 3],$$

su $m + 1$ nodi equidistanti con $m = 6$. Disegna il grafico dell'interpolante.

4 Applicazioni della SVD.

4.1 *Latent Semantic Indexing.* (M. Berry, Z. Drnac, E. Jessup: "Matrices, vector spaces and information retrieval". Siam Review, 41, 335-362 (1999)) Costruisci una matrice termini-documenti A che descrive una collezione di n documenti ognuno rappresentato da m termini, prendendo come documenti i titoli dei libri che hai consultato nella preparazione degli esami universitari e scegliendo opportunamente i termini per la loro descrizione.

La matrice A è costruita in due passi. Dapprima definisci la matrice \tilde{A} i cui elementi $\tilde{a}_{i,j}$ rappresentano il numero delle volte in cui il termine i compare nel documento j . La matrice A sarà ottenuta normalizzando in norma due i vettori colonna di \tilde{A} .

Scrivi un codice Matlab che riceve in input il vettore q di *query* relativo ad A e fornisce in output i valori

$$\cos(\theta_j) = \frac{a_j^T q}{\|q\|_2}, \quad j = 1, \dots, n,$$

dove a_j è la colonna j -esima di A . Testa il tuo codice con diversi vettori q e commenta i risultati.

Dopo aver calcolato la SVD di A mediante l'opportuna routine di Matlab, determina la matrice A_k di rango k che approssima la matrice A per diversi valori di k , analizza l'errore $\frac{\|A - A_k\|_F}{\|A\|_F}$, dove $\|\cdot\|_F$ è la norma di Frobenius, e determina la matrice di rango k che può essere scelta per approssimare la base di dati iniziale A .

Scrivi un codice Matlab che riceve in input il vettore q di *query* relativo ad A_k e fornisce in output i valori

$$\cos(\theta_j) = \frac{(A_k e_j)^T q}{\|A_k e_j\|_2 \|q\|_2}, \quad j = 1, \dots, n,$$

dove $A_k e_j$ è la colonna j -esima di A_k . Testa il tuo codice con diversi vettori q e commenta i risultati, confrontandoli con i risultati ottenuti in precedenza.

4.2 *Compressione di un'immagine.* Scrivi un codice Matlab che legge una matrice A che rappresenta un'immagine, ne calcola la SVD, le relative approssimazioni di rango inferiore A_k e calcola il fattore di compressione $\frac{(m+n)k}{m*n}$. Visualizza l'immagine originale e le varie approssimazioni e commenta i risultati.

Suggerimento. Per caricare l'immagine di un clown in una matrice X e visualizzarla puoi usare i comandi:

```
load clown
colormap(map)
image(X)
```

In alternativa (ad esempio per immagini “esterne” a tua scelta in formato .jpg) puoi usare i comandi:

```
X=imread('nomefile.jpg'); %legge l'immagine
X=rgb2gray(X);             %converte in scala di grigi
X=im2double(X);            %converte doppia precisione
imshow(X)                  %mostra
```

Consulta l'help in linea di Matlab per ulteriori informazioni.

Allegati: il file *mibtel.mat* può essere scaricato da <http://www.dimi.uniud.it/rossana,dbreda> assieme al testo dell'homework.