

Teoria dell'Approssimazione e Applicazioni, A.A. 2012/13

ESERCITAZIONE DI LABORATORIO DEL 10 GIUGNO 2013

Prof. Stefano De Marchi

1. Illustriamo con esempi, l'approccio simmetrico dell'interpolazione di Hermite per l'interpolazione di Hermite del primo ordine (ovvero quando si conoscono i valori posizionali e i gradienti).

Consideriamo la funzione

$$f(x, y) = \frac{\tanh(9(y - x)) + 1}{\tanh(9) + 1}, \quad (1)$$

di cui possiamo facilmente calcolare le derivate parziali prime.

Usando RBF multiquadriche con parametro di forma $\epsilon = 6$, si confrontino 4 problemi

- (a) interpolazione di Lagrange su N punti equispaziati di $[0, 1]^2$;
- (b) interpolazione di Lagrange su $3N$ *clustered points* (gruppi di punti) con *separation distance* $q = h/10$, dove h è la *fill distance* di un insieme di punti equispaziati;
- (c) come nel punto precedente ma con $q = h/100$;
- (d) interpolazione di Hermite dei valori della funzione e valori delle derivate parziali prime, di N punti equispaziati come nel punto (a).

I files da usare sono:

- per il punto (a), la funzione `RBFInterpolation2D.m` sostituendo opportunamente la linea 1 (dove si definisce la RBF usata) e le linee 2-6 (dove si definisce la funzione test) con l'espressione della funzione (1) ;
- per il punto (b), si usi sempre la funzione `RBFInterpolation2D.m`, ma sostituendo le linee 8 e 9 con le seguenti (ricordando di calcolare o leggere preventivamente i `dsites`)

```
q=0.1/(sqrt(N)-1);  
grid=linspace(0,1,sqrt(N));  
shifted=linspace(q,1+q,sqrt(N)); shifted(end)=1-q;  
[xc1,yc1]=meshgrid(shifted,grid);  
[xc2,yc2]=meshgrid(grid,shifted);  
dsites=[dsites;xc1(:) yc1(:); xc2(:) yc2(:)];
```
- per (c), come al punto precedente con le opportune modifiche
- per il punto (d) usare lo script `RBFHermite_2D.m` che fa uso della funzione `DifferenceMatrix.m`.

Produrre due tabelle, che per ogni approccio, fanno vedere come cambia il RMS-error e il numero di condizionamento delle matrici d'interpolazione.

	Lagrange	clustered $q = h/10$
mesh	RMS-error — Condition number	RMS-error — Condition number

	clustered $q = h/100$	Hermite
mesh	RMS-error — Condition number	RMS-error — Condition number

Le mesh da testare sono di dimensioni: 3×3 , 5×5 , 9×9 , 17×17 e 33×33 .

Si verificherà che l'interpolazione di Hermite rispetto ai dati *clustered* con Lagrange, migliora i risultati soprattutto con la mesh 33×33 , dove si ha un miglioramento (sensibile) del numero di condizionamento e dell'errore.

2. Si consideri l'equazione di Poisson con condizioni al bordo di Dirichlet

$$\begin{aligned}\nabla^2 u(x, y) &= -\frac{5}{4}\pi^2 \sin(\pi x) \cos\left(\frac{\pi y}{2}\right), \quad (x, y) \in \Omega = [0, 1]^2 \\ u(x, y) &= \sin(\pi x) \quad (x, y) \in \Gamma_1 \\ u(x, y) &= 0 \quad (x, y) \in \Gamma_2\end{aligned}\tag{2}$$

dove $\Gamma_1 = \{(x, y) : 0 \leq x \leq 1, y = 0\}$ e $\Gamma_2 = \partial\Omega \setminus \Gamma_1$. La soluzione esatta è $u(x, y) = \sin(\pi x) \cos\left(\frac{\pi y}{2}\right)$.

- usare `KansaLaplace_2D.m` per risolvere l'equazione (2) con il metodo non-simmetrico di Kansa con multiquadriche inverse IMQ e con gaussiane G, con $\epsilon = 3$. Lo script implementa le IMQ.

Per la gaussiana, bisogna sostituire opportunamente le linee nel codice che definiscono la RBF usata e il suo laplaciano. Per la gaussiana dovremo scrivere

```
rbf = @(e,r) exp(-(e*r).^2); ep=3;
Lrbf = @(e,r) 4*e^2 exp( -(e*r).^2).*((e*r).^2-1);
```

In entrambi i casi si prendano $N = 3^2, 5^2, 9^2, 17^2$ centri interni e $M = 2^2, 4^2, 6^2, 8^2$ centri addizionali sul bordo (che si possono prendere sul bordo del quadrato oppure al di fuori del bordo del quadrato, come descritto nel codice).

I punti interni possono essere scelti come punti di Halton o equispaziati.

Produrre una tabella con il RMS-error e il numero di condizionamento della matrice di collocazione.

Nota. Serve il bi-laplaciano della funzione gaussiana. Ricordo le seguenti due formule del laplaciano e del bi-laplaciano di $\varphi(r) = \varphi(\|x\|)$

$$\begin{aligned}\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right) \varphi(\|x\|) &= \frac{d^2}{r^2} \varphi(r) + \frac{1}{r} \varphi(r) \quad \text{laplaciano} \\ \left(\frac{\partial^4}{\partial x^4} + 2\frac{\partial^2}{\partial x^2 \partial y^2} + \frac{\partial^4}{\partial y^4}\right) \varphi(\|x\|) &= \frac{d^4}{dr^4} \varphi(r) + \frac{2}{r} \frac{d^3}{dr^3} \varphi(r) - \frac{1}{r} \frac{d^2}{dr^2} \varphi(r) + \frac{1}{r^3} \frac{d}{dr} \varphi(r) \quad \text{bi-lapl.}\end{aligned}$$

- Usare `HermiteLaplace_2D.m` per risolvere l'equazione (2) con il metodo simmetrico di Fasshauer basato sulla collocazione di Hermite. Fare gli stessi esperimenti con IMQ e G.

Ricordo che il link dove trovare i files delle funzioni Matlab è:
<http://www.math.unipd.it/~demarchi/TAA2010>