Dottorato in Ingegneria Civile

*** Introduzione a Matlab***

Dott. F.Incensi **4 Maggio 2011**

Esercitazione 2 :

Risolvere in ambiente Matlab i seguenti esercizi.

PRIMA DI INIZIARE:

Scaricare dall'indirizzo

http://www.mathworks.com/moler/index_ncm.html

i capitoli *interp* e *least squares*.

File utilizzati del pacchetto *ncm*:

- polyinterp: interpolazione Lagrange
- piecelin: interpolazione lineare a tratti
- interpgui: confronto tra diverse strategie interpolanti

Altri files:

- polyinterpN: interpolazione Newton

1. Si calcoli il polinomio interpolante della funzione

   $$F(x)=\sin(x)$$

   definita sull'intervallo $[-\pi;\pi]$ impiegando 5,11, 21, 41 punti equispaziati, implementando entrambi i polinomi interpolanti di Lagrange e di Newton. Si valuti il comportamento dell'errore (in norma inf).

2. Si ripeta l'esercizio precedente con la funzione di Runge

   $$F(x)=1/(1+x^2).$$

   nell'intervallo [-5 ; 5] impiegando 5, 11, 21, 41 e 81 nodi equispaziati, valutando l'errore di interpolazione.

   Provare successivamente una interpolazione lineare a tratti (*piecelin*), valutando l'errore commesso.

3. Data

   $$F(x) = \sin(x^2),$$

   definita sull'intervallo [0 , 3]:

   - si costruisca l'interpolante lineare polinomiale a tratti di F usando successivamente n = 2, 4, 8, 16, 32, 64 e 128 punti equispaziati;
   - per ciascun valore di n, determinare l'errore di interpolazione e riportare i valori così ottenuti in un grafico doppio logaritmico, in funzione dell'ampiezza della triangolazione $h = (b-a)/n$

4. Realizzare lo script ***interpola.m*** che calcoli il polinomio interpolante in [a,b] di grado n di un insieme di punti $P_i=(x_i,y_i)$ con x_i a scelta dell'utente:
   - punti x_i equidistanti;
   - punti x_i definiti dagli zeri dei polinomi di Chebychev :

$$x_i = \frac{a+b}{2} - \frac{b-a}{2}\cos\left(\frac{2(i-1)+1}{2(n+1)} * \pi\right), \quad i = 1,..,n+1$$

e y_i ottenuti dalla campionamento (valutazione) nei punti x_i della funzione

y=sin(x)-2sin(2x).

Si utilizzi il metodo di Lagrange, quello Newton, l'interpolazione lineare a tratti e provare l'interpolazione a tratti con spline cubiche (built-in function ***spline***()).

Lo script infine visualizzi in uno stesso grafico la funzione test da interpolare, i punti di interpolazione e i due polinomi interpolanti.

Modificare lo script affinché consideri la funzione test da interpolare $y=1/(1+x^2)$, $x \in [-5,5]$ (funzione di Runge)

Verificare cosa succede al variare del grado n.

5. Realizzare l'esercizio 3.4 del capitolo *interp* del Moler utilizzando le funzioni dei interpolazione finora viste.
6. Realizzare l'esercizio 5.8 *a-b* del capitolo *leastsquares* del Moler (per calcolare l'approssimante polinomiale utilizzare la funzione *polyfit* di Matlab*)*.
7. La temperatura T in prossimita' del suolo varia al variare della concentrazione k dell'acido carbonico e della latitudine L. Per k=1.5 la temperatura al suolo subisce una variazione dipendente dalla temperatura secondo la seguente tabella

L	-55	-45	-35	-25	-15	-5	5	15	25	35	45	55	65
T	3.7	3.7	3.52	3.27	3.2	3.15	3.15	3.25	3.47	3.52	3.65	3.67	3.52

Si vuole costruire un modello che descriva la legge T=T(L) anche per latitudini non misurate, per esempio si vuole valutare la variazione di temperatura a Roma (L=42°).

Sperimentare nello script ***test.m*** le seguenti tecniche:

- Approssimazione nel senso dei minimi quadrati con un polinomio di grado1 e 2;
- Interpolazione polinomiale;
- Interpolazione con spline cubiche.