Elaborato “Calcolo Numerico”

Studente: Daniil Ryzhkov
Matricola: 6217473
Consegna: 06/09/2020

Esercizio 1
Verificare che, per *h* sufficientemente piccolo,

$$\frac{f(x-h)-2f(x)+f(x+h)}{h^2}=f''(x)+o(h^2)$$

Per il teorema di Taylor abbiamo:

$$f(x+h)=f(x)+hf'(x)+\frac{h^2}{2}f''(x)+\frac{h^3}{6}f'''(x)+o(h^4)$$

$$f(x-h)=f(x)-hf'(x)+\frac{h^2}{2}f''(x)-\frac{h^3}{6}f'''(x)+o(h^4)$$

allora

$$\frac{f(x-h)-2f(x)+f(x+h)}{h^2}=\frac{f(x)-hf'(x)+\frac{h^2}{2}f''(x)-\frac{h^3}{6}f'''(x)}{h^2}-\frac{2f(x)}{h^2}+\frac{f(x)+hf'(x)+\frac{h^2}{2}f''(x)+\frac{h^3}{6}f'''(x)}{h^2}+o(h^2)=\frac{h^2f''(x)}{h^2}+o(h^2)=f''(x)+o(h^2)$$

Esercizio 2
Eseguire il seguente script Matlab, e spiegare cosa calcola.

```
u = 1; while 1, if 1+u==1, break, end, u = u/2; end, u
```

È praticamente un modo di calcolare l’epsilon della macchina che può essere usato per misurare l’errore di round-off, ed è definito come il numero più piccolo ε, t.c. rappresentazione floating-point del valore $1+\varepsilon$ è maggiore di 1.

Esercizio 3
Eseguire il seguente script Matlab:

```
a = 1e20; b = 100; a-a+b
a = 1e20; b = 100; a+b-a
```

Spiegare i risultati ottenuti.

Nel secondo caso abbiamo ottenuto un risultato errato. Per il problema di round-off a+b ha dato il risultato pari ad “a”. E quindi a+b-a risulta pari a zero.

Esercizi 4-5
Vedi le cartelle risp. “esercizio4” e “esercizio5”.

Esercizio 6

	1 Method	2 Tolerance	3 Result	4 Iterations
1	'bisect'	'1e-03'	'7.3925781250000000e-01'	9
2	'newton'	'1e-03'	'7.3908513338528403e-01'	4
3	'secant'	'1e-03'	'7.3908511212746386e-01'	4
4	'cord'	'1e-03'	'7.3907813088002572e-01'	4
5	'bisect'	'1e-06'	'7.3908519744873047e-01'	19
6	'newton'	'1e-06'	'7.3908513321516067e-01'	5
7	'secant'	'1e-06'	'7.3908513321500124e-01'	5
8	'cord'	'1e-06'	'7.3908511564437829e-01'	6
9	'bisect'	'1e-09'	'7.3908513318747282e-01'	29
10	'newton'	'1e-09'	'7.3908513321516067e-01'	5
11	'secant'	'1e-09'	'7.3908513321516067e-01'	6
12	'cord'	'1e-09'	'7.3908513317107094e-01'	8
13	'bisect'	'1e-12'	'7.3908513321566716e-01'	39
14	'newton'	'1e-12'	'7.3908513321516067e-01'	6
15	'secant'	'1e-12'	'7.3908513321516067e-01'	6
16	'cord'	'1e-12'	'7.3908513321515512e-01'	11

Dai resultati ottenuti si vede, che il metodo di Newton richiede meno iterazioni rispetto agli altri metodi e dà anche i risultati migliori. Il metodo di bisezione invece richiede molto di più iterazioni e genera i risultati meno precisi.

Esercizio 7

	1 Method	2 Tolerance	3 Result	4 Iterations
1	'newton'	'1e-03'	'1.9940029619560963e-03'	16
2	'newton mod.'	'1e-03'	'2.7607675821411041e-04'	8
3	'aitken'	'1e-03'	'-1.5710278843430792e+00'	3
4	'newton'	'1e-06'	'1.3492222093811503e-06'	34
5	'newton mod.'	'1e-06'	'3.7870612995219788e-07'	14
6	'aitken'	'1e-06'	'-1.5707963609165836e+00'	4
7	'newton'	'1e-09'	'1.3694055305480025e-09'	51
8	'newton mod.'	'1e-09'	'1.7316238223695715e-10'	21
9	'aitken'	'1e-09'	'-1.5707963267948974e+00'	5
10	'newton'	'1e-12'	'1.3898907785952522e-12'	68
11	'newton mod.'	'1e-12'	'2.3753413201228688e-13'	27
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Il metodo di Newton modificato dà il risultato circa due volte più velocemente rispetto al metodo di Newton standard. Il metodo con accelerazione di Aitken ha dato invece un risultato sbagliato in questo caso. Però spostando il punto di partenza un po' verso una radice si ottiene il risultato corretto e più velocemente rispetto agli altri due metodi.

Esercizi 8-9

Vedi le cartelle risp. “esercizio8” e “esercizio9”.

Esercizio 10

```
Command Window
>> main
   3.9472e-15

   1.1960e-14

   1.9745e-12

   7.8863e-11

   2.4564e-08

   1.2470e-06

   1.3292e-04

    0.0154

    1.9956

  148.8848
```

Il risultato di queste operazioni è un vettore con gli errori tra il vettore X di riferimento e il vettore X ottenuto dopo i calcoli eseguiti per ciascun passo dell'algoritmo. Ad aumentare la variabile k della funzione `linsis` la matrice A tende ad una matrice singolare, perciò il risultato diventa poco preciso ed alla fine addirittura errato.

Esercizi 11-13
Vedi le cartelle risp. "esercizio11"-"esercizio13".

Esercizio 14
Nella prima espressione `A\b` abbiamo usato il cosiddetto operatore "backslash" del MATLAB, che praticamente risolve un sistema di equazioni lineari del tipo

$$Ax = b.$$

Formalmente si può esprimere x come

$$x = A^{-1}b.$$

Nel nostro esempio però la matrice A non è quadrata, perciò non possiamo invertirla. L'operatore backslash invece è un operatore più avanzato rispetto alla semplice inversione. Infatti usa gli algoritmi diversi (quindi non necessariamente l'inversione) a secondo della struttura della matrice A.
Per quanto riguarda la seconda espressione, dobbiamo riferirci al metodo dei minimi quadrati, cui soluzione a matrice si può scrivere come

$$(A^TA)^{-1}A^Tb$$

Che è infatti la soluzione di un sistema lineare del tipo

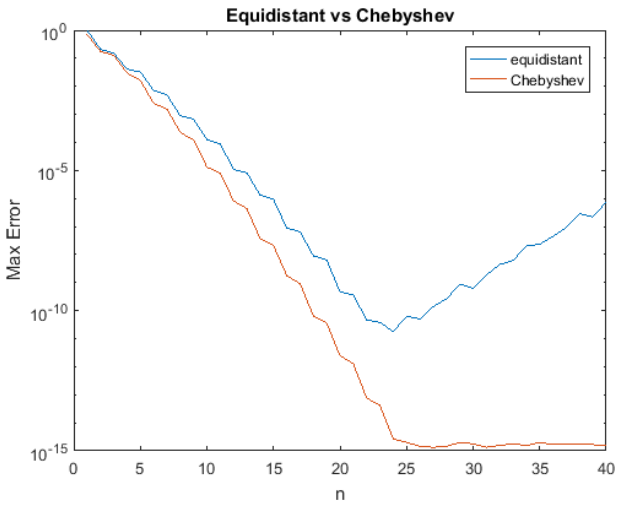
$$A^TAx = A^Tb$$

Perciò, applicando l'operatore backslash di MATLAB si può risolvere questo sistema come

```
(A'*A)\(A'*b)
```

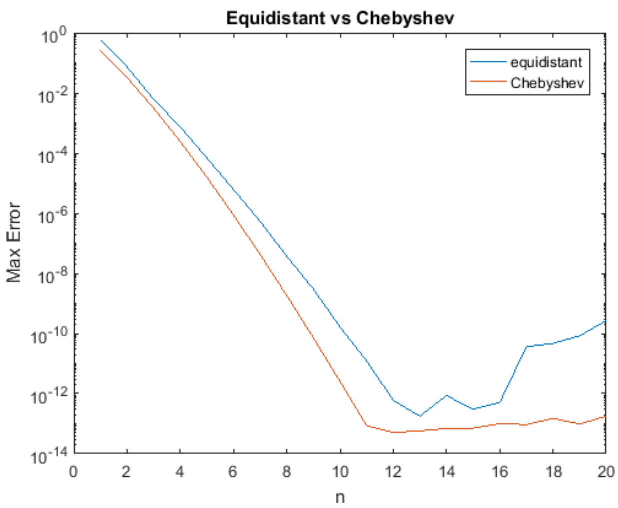
Dal punto di vista di algebra lineare sono due soluzioni identiche, però in pratica la seconda espressione ci dà un risultato errato. L'operatore nativo backslash di MATLAB risulta numericamente più stabile rispetto all'espressione seconda, e quindi dà un risultato più preciso.

Esercizio 15



Come si vede dal grafico, selezionando i nodi usando l'algoritmo di Chebyshev, l'interpolazione con polinomi di Lagrange risulta più precisa. Più che altro, usando i nodi equidistanti, dopo un certo punto, l'errore massimale comincia ad aumentare per il cosiddetto fenomeno di Runge. Con i nodi di Chebyshev invece le conseguenze di questo fenomeno non sono così gravi.

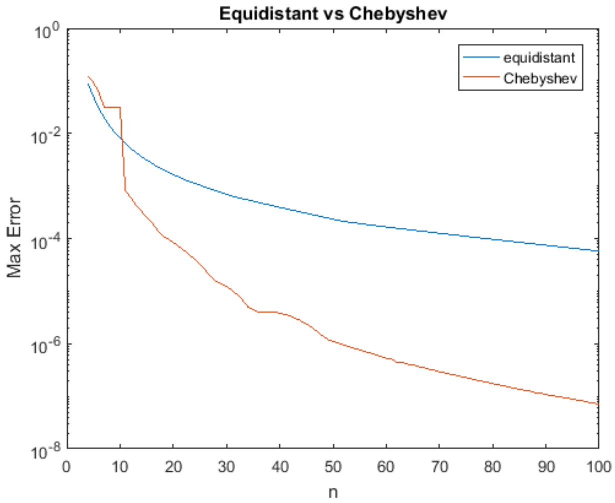
Esercizio 16



Con l'interpolazione di Hermite si ottiene la stessa precisione con meno nodi usati. Però dopo circa 10 nodi non cresce più, mentre con l'interpolazione di Lagrange si poteva ottenere un risultato anche più preciso, aggiungendo più nodi. Come sempre i nodi di Chebyshev danno un risultato più preciso.

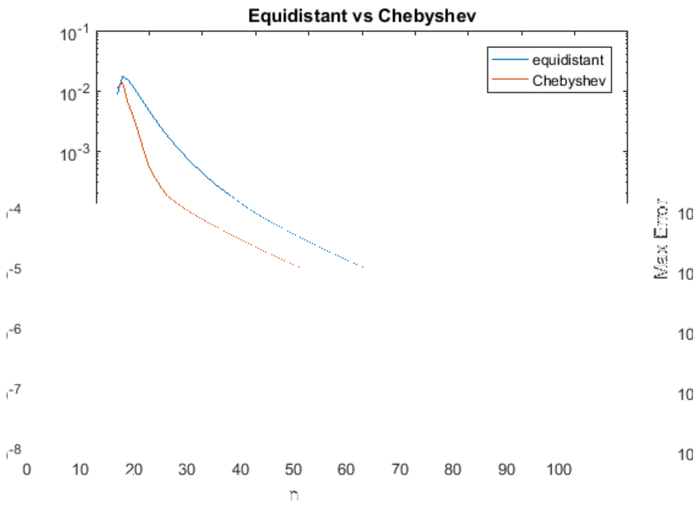
Esercizio 17
Vedi la cartella "esercizio17".

Esercizio 18



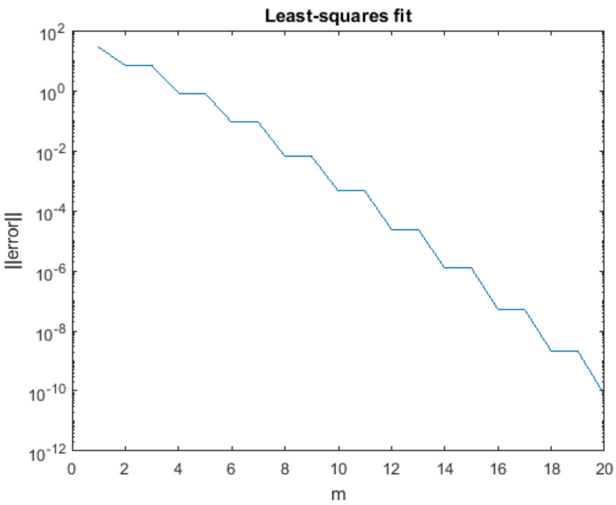
Usando l’interpolazione di spline cubica naturale il risultato ottenuto non è così preciso, come nel caso di interpolazione di Lagrange o Hermite. Però si elimina completamente il fenomeno di Runge. Anche qui i nodi di Chebyshev danno un risultato più preciso.

Esercizio 19



Usando l’interpolazione di spline ‘not-a-knot’ di MATLAB il risultato ottenuto è quasi uguale a quello di spline cubica naturale dell’esercizio precedente nel caso dei nodi di Chebyshev e un po’ meglio nel caso dei nodi equidistanti.

Esercizio 20



Il metodo dei minimi quadrati risulta poco efficiente per i polinomi di grado piccolo rispetto agli altri metodi, come ad es. interpolazione di Hermite. Però ad aumentare il grado di polinomio, la precisione si aumenta abbastanza velocemente. Comunque si deve prendere in considerazione il fatto, che il dataset da noi generato aveva varianza piccola e non soffriva dall'eteroschedasticità. Nel caso contrario il metodo dei minimi quadrati sarebbe poco efficiente.