Quadratura in Matlab

Alvise Sommariva

Università degli Studi di Padova
Dipartimento di Matematica Pura e Applicata

24 maggio 2023

Sia $f \in C([a,b])$, con $[a,b]$ limitato e supponiamo di voler approssimare

$$I(f) = \int_a^b f(x)dx.$$

- *La formula dei trapezi approssima $I(f)$ con*

$$S_T(f) = \frac{b-a}{2}(f(a) + f(b))$$

  ed è tale che $I(p_1) = S_T(p_1)$ per ogni polinomio p_1 di grado al piú 1.

- *La formula di Cavalieri-Simpson approssima $I(f)$ con*

$$S_{CS}(f) = \frac{b-a}{6}\left(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b)\right)$$

  ed è tale che $I(p_3) = S_T(p_3)$ per ogni polinomio p_3 di grado al piú 3.

Esempio (1)

Si approssimi mediante la regola dei trapezi e Cavalieri-Simpson

$$I = \int_0^1 \frac{1}{1+x} dx \approx 0.6931471805599453.$$

Nota. (Calcolo dell'integrale esatto)

Posto $t = 1 + x$, *abbiamo* $dt = dx$, *e quindi, per sostituzione e il teorema fondamentale del calcolo integrale:*

$$\begin{aligned} I &= \int_0^1 \frac{1}{1+x} dx = \int_1^2 \frac{1}{t} dt \\ &= \log_e(2) - \log_e(1) = \log_e(2) \\ &\approx 0.6931471805599453. \end{aligned} \tag{1}$$

Definiamo demo_quadratura1 come segue.

```
function demo_quadratura1

% demo per paragonare la regola dei trapezi e di Cavalieri Simpson.

% integranda
f=@(x) 1./(1+x);

% intervallo di integrazione
a=0; b=1;

% risultato esatto
sol=log(2);

% regola dei trapezi
S_T=((b-a)/2)*(f(a)+f(b));

% regola di Cavalieri-Simpson
c=(a+b)/2; % punto medio
S_CS=((b-a)/6)*(f(a)+4*f(c)+f(b));

% display risultati ed errori assoluti
fprintf('\n \t Integrale esatto: %1.15e',sol);
fprintf('\n \t Regola dei Trap.: %1.15e',S_T);
fprintf('\n \t Regola di Ca.Si.: %1.15e',S_CS);
fprintf('\n ')
fprintf('\n \t Err.Ass.Trap. : %1.15e',abs(sol-S_T));
fprintf('\n \t Err.Ass.Ca.Si.: %1.15e',abs(sol-S_CS));
fprintf('\n \n')
```

Lanciando tale codice da command-window

```
>> demo_quadratura1

   Integrale esatto: 6.931471805599453e-01
   Regola dei Trap.: 7.500000000000000e-01
   Regola di Ca.Si.: 6.944444444444443e-01

   Err.Ass.Trap. : 5.685281944005471e-02
   Err.Ass.Ca.Si.: 1.297263884499023e-03

>>
```

Commento

Nonostante le poche valutazioni della funzione, cioé 2 nel caso della formula dei trapezi e 3 in quella di Cavalieri-Simpson, abbiamo ottenuto una approssimazione, seppur imprecisa, dell'integrale richiesto.

1. *Nelle ipotesi della sezione precedente, la quantità*

$$I(f) = \int_a^b f(x)dx$$

è approssimabile mediante la formula dei trapezi composta

$$S_N^{(T)}(f) = \frac{h}{2} f(x_1) + hf(x_2) + \ldots + hf(x_N) + \frac{h}{2} f(x_{N+1}) \qquad (2)$$

ove

- $x_k = a + (k-1) \cdot h$, *con* $k = 1, \ldots, N+1$,
- $h = \frac{b-a}{N}$.

2. *Si vede facilmente che tale formula è del tipo* $\sum_{i=1}^{N+1} w_i f(x_i)$ *con*

- $w_1 = w_{N+1} = \frac{h}{2}$,
- $w_2 = \ldots = w_N = h$.

ove

- *le ascisse* x_k, *con* $k = 1, \ldots, N+1$ *sono dette nodi*,
- *i valori* w_k, *con* $k = 1, \ldots, N+1$ *sono detti pesi*.

Tale formula si ottiene suddividendo $[a, b]$ in N subintervalli aventi la stessa ampiezza, applicando in ognuno di loro la regola dei trapezi.

La funzione `trapezi_composta` appena esposta calcola i nodi e i pesi della omonima formula composta.

```
function [x,w]=trapezi_composta(N,a,b)

% Formula dei trapezi composta.

% input:
% N: numero di subintervalli
% a,b: estremi di integrazione
%
% output:
% x: nodi di integrazione (vettore colonna)
% w: pesi di integrazione (vettore colonna)

h=(b-a)/N;                  % passo di integrazione
x=a:h:b; x=x';              % nodi di integrazione
w=ones(N+1,1);              % pesi di integrazione
w(1)=0.5; w(N+1)=0.5;
w=w*h;
```

Approssimiamo $I(f) = \int_a^b f(x)dx$ mediante la formula di Cavalieri-Simpson composta

$$\boxed{S_N^{(CS)}(f) = \frac{h}{3} f(x_1) + \frac{h}{3} f(x_{2N+1}) + \frac{4h}{3} \sum_{s=1}^{N} f(x_{2s}) + \frac{2h}{3} \sum_{s=1}^{N-1} f(x_{2s+1})}$$

in cui $x_k = a + (k-1)h$, con $k = 1, \ldots, 2N+1$ e $h = \frac{b-a}{2N}$.

Si vede facilmente che tale formula è del tipo $\sum_{i=1}^{2N+1} w_i f(x_i)$ dove

- $w_1 = w_{2N+1} = \frac{h}{3}$,
- $w_2 = w_4 = \ldots = w_{2N} = \frac{4h}{3}$,
- $w_3 = w_5 = \ldots = w_{2N-1} = \frac{2h}{3}$,

con $h = \frac{b-a}{2N}$.

Nota.

Si osservi che spesso si scrive pure

$$S_N^{(CS)}(f) = \frac{h^*}{6} f(x_1) + \frac{h^*}{6} f(x_{2N+1}) + \frac{4h^*}{3} \sum_{s=1}^{N} f(x_{2s}) + \frac{2h^*}{6} \sum_{s=1}^{N-1} f(x_{2s+1})$$

ove $x_k = a + (k-1)h^*/2$, con $k = 1, \ldots, 2N+1$, in cui $h^* = \frac{b-a}{N} = 2h$.

La funzione `cavalieri_simpson_composta` sotto esposta calcola i nodi e i pesi della omonima formula.

```
function [x,w]=cavalieri_simpson_composta(N,a,b)

% Formula dei trapezi composta.
% input:
% N: numero di subintervalli
% a,b: estremi di integrazione
%
% output:
% x: nodi di integrazione (vettore colonna)
% w: pesi di integrazione (vettore colonna)

% ---- calcolo dei nodi ----
h=(b-a)/(2*N);      % passo "h" della formula
x=a:h:b; x=x';      % nodi

% ---- calcolo dei pesi ----
w=zeros(2*N+1); % inizializzazione
w(1)=h/3;       % primo peso
w(2*N+1)=h/3;   % ultimo peso
% pesi con indici pari intermedi
ind_pari=2:2:2*N;
w(ind_pari)=4*h/3;
% pesi con indici dispari intermedi
ind_disp=3:2:2*N-1;
w(ind_disp)=2*h/3;
```

Commento

1. *Osserviamo che fissato il numero di subintervalli N*
   - *la formula dei trapezi composta ha $N+1$ punti;*
   - *la formula di Cavalieri-Simpson composta ha $2N+1$ punti;*

   Modifichiamo demo_quadratura1 in demo_quadratura2, così da testare tali formule.

2. *Affinchè il numero di valutazioni sia uguale,*
   - *utilizziamo $N = 10$ intervalli per la formula dei trapezi composta,*
   - *utilizziamo $N = 5$ intervalli per Cavalieri-Simpson composta,*

   così che entrambe eseguano 11 valutazioni dell'integranda.

```
function demo_quadratura2

% demo per paragonare la regola dei trapezi e di Cavalieri Simpson.

% integranda
f=@(x) 1./(1+x);

% intervallo di integrazione
a=0; b=1;

% risultato esatto
sol=log(2);

% formula dei trapezi composta
N=10; % scegliere numero pari
[x,w]=trapezi_composta(N,a,b);
S_T=sum(w.*f(x));

% formula di Cavalieri-Simpson composta
N=N/2;
[x,w]=cavalieri_simpson_composta(N,a,b);
S_CS=sum(w.*f(x));

% display risultati ed errori assoluti
fprintf('\n \t Integrale esatto: %1.15e',sol);
fprintf('\n \t Regola dei Trap.: %1.15e',S_T);
fprintf('\n \t Regola di Ca.Si.: %1.15e \n',S_CS);
fprintf('\n \t Err.Ass.Trap. : %1.15e',abs(sol-S_T));
fprintf('\n \t Err.Ass.Ca.Si.: %1.15e \n \n',abs(sol-S_CS));
```

Digitando `demo_quadratura2` nella command-window

```
>> demo_quadratura2

   Integrale esatto: 6.931471805599453e-01
   Regola dei Trap.: 6.937714031754280e-01
   Regola di Ca.Si.: 6.931502306889304e-01

   Err.Ass.Trap.  : 6.242226154826724e-04
   Err.Ass.Ca.Si.: 3.050128985160327e-06

>>
```

Commento

A parità di nodi, la formula composta di Cavalieri-Simpson ha fornito un risultato migliore di quella dei trapezi composta.

Esercizio (1)

Si scriva una function esercizio_quadratura *che*

1. *Definisca la funzione* $f(x) = \exp(-x^2)$, *e si ponga* `a=0`, `b=1`.
2. *Dopo aver confrontato l'help della funzione* `erf`, *calcoli il valore*

$$\int_0^1 \exp(-x^2)dx$$

   utilizzando `erf` *e lo assegni alla variabile* `sol`.

3. *Per* $k = 1, \ldots, 8$
   - *Calcoli il valore ottenuto dalla formula dei trapezi composta in* $N = 2^k$ *subintervalli e lo salvi in* `S_T(k)`.
   - *Calcoli il valore ottenuto dalla formula di Cavalieri-Simpson composta in* $N = 2^k$ *subintervalli e lo salvi in* `S_CS(k)`.
   - *Assegni a* `E_T(k)` *il valore* `abs(sol-S_T(k))` *e a* `E_CS(k)` *il valore* `abs(sol-S_CS(k))`.
4. *In una sola figura disegni in scala semilogaritmica*
   - *il grafico delle coppie* $(2^k, E\_T(k))$, $k = 1, \ldots, 8$ *(con linea rossa, di spessore 2),*
   - *il grafico delle coppie* $(2^k, E\_CS(k))$, $k = 1, \ldots, 8$ *(con linea blu, di spessore 2).*
5. *Nella stessa figura, usi come titolo*

`Esercizio quadratura`

   e legenda con stringhe `Trapezi`, `Cavalieri-Simpson`.