I° Esercitazione – Secondo semestre
Analisi Numerica – Calcolo Numerico
A.A. 2014– 2015

Esercizio 1. Costruire la formula di quadratura interpolatoria secondo Newton-Cotes con $n = 4$. Utilizzare tale formula per la valutazione dell'integrale

$$I = \int_0^1 e^{-x^2}\,dx$$

Calcolare l'errore effettivo.

Esercizio 2. Nella seguente tabella sono riportati i coefficienti e i resti delle formule di Newton-Cotes per $n = 1, \cdots, 7$.

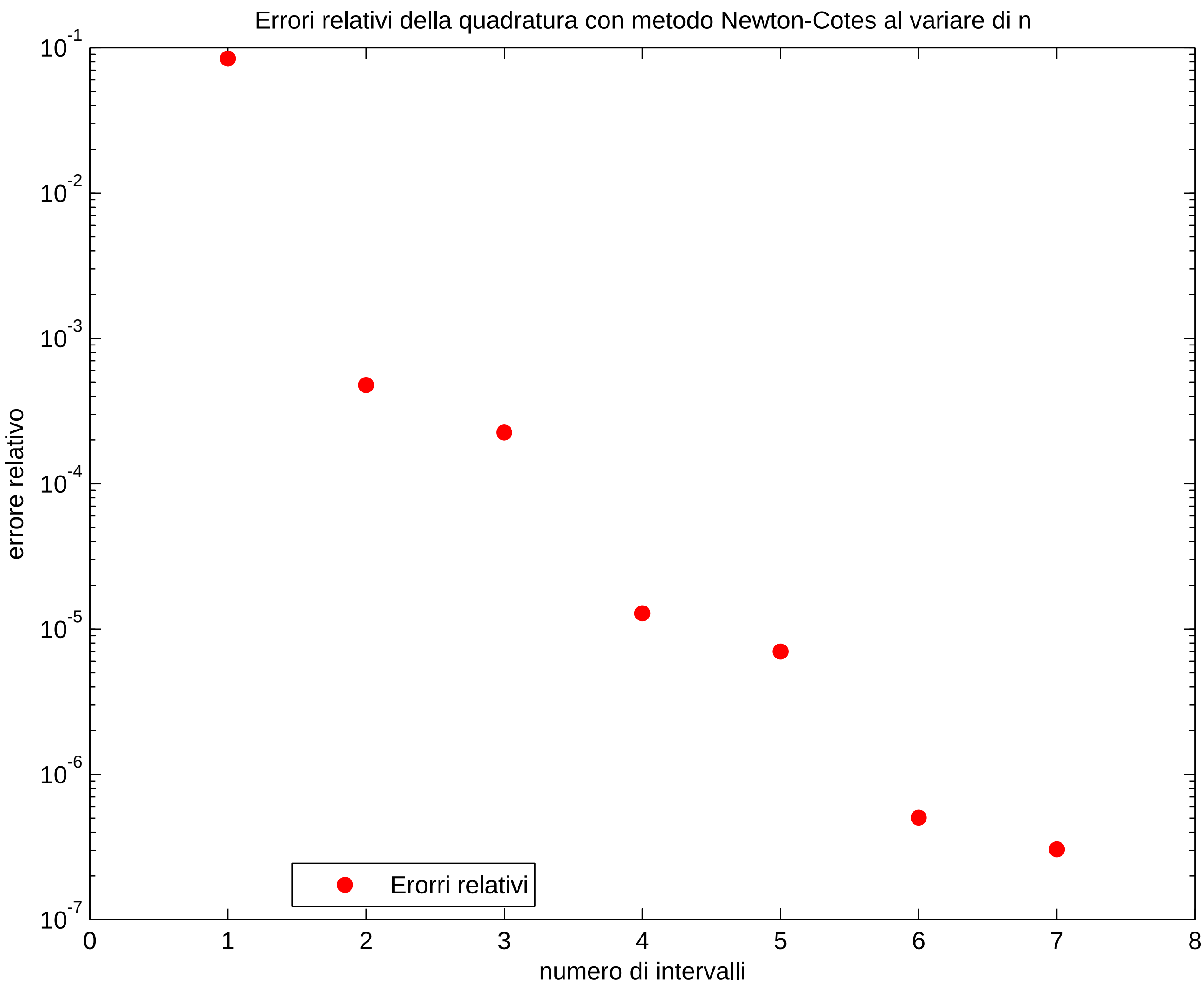
n	α_0	α_1	α_2	α_3	*resto*
1	$\frac{1}{2}$				$-\frac{1}{12}h^3 f''(\xi)$
2	$\frac{1}{3}$	$\frac{4}{3}$			$-\frac{1}{90}h^5 f^{(4)}(\xi)$
3	$\frac{3}{8}$	$\frac{9}{8}$			$-\frac{3}{80}h^5 f^{(4)}(\xi)$
4	$\frac{14}{45}$	$\frac{64}{45}$	$\frac{24}{45}$		$-\frac{8}{945}h^7 f^{(6)}(\xi)$
5	$\frac{95}{288}$	$\frac{375}{288}$	$\frac{250}{288}$		$-\frac{275}{12096}h^7 f^{(6)}(\xi)$
6	$\frac{41}{140}$	$\frac{216}{140}$	$\frac{27}{140}$	$\frac{272}{140}$	$-\frac{9}{1400}h^9 f^{(8)}(\xi)$
7	$\frac{5257}{17280}$	$\frac{25039}{17280}$	$\frac{9261}{17280}$	$\frac{20923}{17280}$	$-\frac{8183}{518400}h^9 f^{(8)}(\xi)$

Utilizzando la tabella precedente rappresentare in un grafico l'errore relativo nel calcolo delle formule di Newton-Cotes per l'approssimazione dell'integrale dell'esercizio precedente.

Esercizio 3. Rappresentare in un grafico gli errori relativi nel calcolo di $\int_0^1 \sin x\,dx$ con le formule dei trapezi e di Cavalieri-Simpson iterate (al crescere del numero di sottointervalli della decomposizione).

```
% Esercizio 2 - Esercitazione 7
format long
% matrice contenente le costanti di newton-cotes per n= 1,...7
costNC=[ 1/2 1/2 0 0 0 0 0 0;
    1/3 4/3 1/3 0 0 0 0 0;
    3/8 9/8 9/8 3/8  0 0 0 0;
    14/45 64/45 24/45 64/45 14/45 0 0 0;
    95/288 375/288 250/288 250/288 375/288 95/288 0 0;
    41/140 216/140 27/140 272/140 27/140 216/140 41/140 0;
    5257/17280 25039/17280 9261/17280 20923/17280 20923/17280 9261/17280 25039/17280 5257/17280
    ];
% costruiamo la quadratura interpolatoria secondo Newton-Cotes con n che
% varia da 1 a 7 dell'integrale di exp(-x^2) tra 0 e 1
quadr= zeros(7,8);
for i = 1:7
    h=1/i; % passo = b-a/n
    for j = 1:8
        % valutiamo la funzione in j-1 perche viene preso il valore di
        % x_ii=a+ii*h con ii=0,...,n , ed in matlab i vettori iniziano da
        % indice 1
        quadr(i,j)=costNC(i,j)*(exp(-((j-1)*h)^2));
    end
    % otteniamo il valore della quadratura sommando i valori ottenuti dalla
    % prodotto delle costanti e delle f(a+ih)
    val_quadr(i,:) = h*sum(quadr(i,:));
end
% valore esatto
x = linspace(0,1,1000);
% creiamo la funzione
funz = @(x) exp(-x.^2);
val_esa = integral(funz, 0, 1);
% errore relativo
err_rel= abs(val_esa-val_quadr)./val_esa;
plot(1:7,err_rel)
format short
```

Grafico Esercizio 2 - Esercitazione 7



```
% Esercizio 3 - Esercitazione 7
val_trap=zeros(50,1);
val_cavsim=zeros(50,1);
for i = 1:50   % numero intervalli decomposizione
   % TRAPEZI
   x_t=linspace(0,1,i+1); % i+1 punti della decomposizione con i intervalli.
   for j = 1:i % per ogni sottointervallo effettuo il calcolo
      passo_sot_trap = x_t(j+1)-x_t(j);  % passo sottointervallo trapezi
      % somma ogni quadratura di ogni intervallo della decomposizione
      val_trap(i)=val_trap(i) + ((passo_sot_trap/2)*(sin(x_t(j))+sin(x_t(j+1))));
   end
   % CAVALIERI-SIMPSON
      x_cv=linspace(0,1,2*i+1); % 2*i+1 punti della decomposizione con i intervalli.
      % il punto medio a+b/2 viene calcolato da matlab tramite linspace.
   for j = 1:2:2*i % l'indice si incrementa di 2 ogni iterata per prendere 3 punti per volta
      % con un solo punto in comune.
      h_sot_cavsim = (x_cv(j+2)-x_cv(j))/2;  % h=(b-a/n) sottointervallo cavalieri-simpson
      % somma ogni quadratura di ogni intervallo della decomposizione
      val_cavsim(i)=val_cavsim(i) + ((h_sot_cavsim/3)*(sin(x_cv(j)) + 4*sin(x_cv(j+1)) + sin(x_cv(j+2))));
   end
end
%valore esatto dell'integrale da 0 a 1 di sin x
xx=linspace(0,1,1000);
funz = @(xx) sin(xx);
val_esa= ones(50,1).*integral(funz, 0, 1);
%errore relativo della quadratura con la formula dei trapezi
err_rel_trap= abs(val_esa-val_trap)./val_esa;
%errore relativo della quadratura con la formula di cavalieri-simpson
err_rel_cavsim= abs(val_esa-val_cavsim)./val_esa;
plot(1:50,err_rel_trap,1:50,err_rel_cavsim)
```