Esercitazione 1:
Errori macchina

Richiami di Teoria

Errore assoluto e relativo:

$$\varepsilon_a = |x^\star - x| \qquad \text{e} \qquad \varepsilon_r = \frac{|x^\star - x|}{|x|}\,.$$

dove $x^\star$ è il valore "approssimato", mentre x è il valore "vero", cioè quello di riferimento.

Il numero di cifre decimali in comune tra il numero x e il suo valore approssimato $x^\star$ si ottiene calcolando

$$-log_{10}\varepsilon_r.$$

Esercizio 1

Eseguire le seguenti istruzioni MatLab e commentare i risultati ottenuti.

```
% Esempio di generazione spontanea di cifre


disp('ESEMPIO: Cifre spontanee');


format long e

% Primo esempio
a = 2.6 + 0.2
a = a + 0.2
a = a + 0.2

% Secondo esempio
b = 2.6 + 0.6
b = b + 0.6
b = b + 0.6
b = b + 0.6

% Ritorno alla visualizzazione classica
format
```

Risultati

```
ESEMPIO: Cifre spontanee
a =
    2.800000000000000e+000
a =
    3.000000000000000e+000
a =
    3.200000000000001e+000
b =
    3.200000000000000e+000
b =
    3.800000000000000e+000
b =
    4.400000000000000e+000
b =
     5
```

Esercizio 2

Calcolare l'errore assoluto, relativo, il numero di cifre in comune dei seguenti numeri:

- π (pi-greco) [pi];
- e (numero di nepero) [exp(1)] ;
- $0.123456789 \cdot 10^{-20}$ [0.123456789e-20];
- $0.123456789 \cdot 10^{+20}$ [0.123456789e+20];

dove il valore approssimato si ottiene considerando solo alcune cifre (a scelta dallo studente) che formano il numero stesso.

Risultati

Ad esempio si veda il seguente codice per il primo esempio.

```
% Esempio di calcolo dell'errore relativo


disp('ESEMPIO: Calcolo errori assoluto e relativo');


% Pi-greco e sua approssimazione
xv = pi;
xappr = 3.1415926;

% Errore assoluto, relativo e numero di cifre
err_ass = abs(xv-xappr);
err_rel = abs(xv-xappr)/abs(xv);
cf = floor(-log10(err_rel));

% Visualizzazione a video
fprintf('%15s : %16.10e\n','x_vera',xv);
fprintf('%15s : %16.10e\n','x_approx',xappr);
fprintf('%15s : %16.10e\n','err_ass',err_ass);
fprintf('%15s : %16.10e\n','err_rel',err_rel);
fprintf('%15s : %d\n\n','cifre sign.',cf);
```

Il codice precedente produce il seguente output.

```
ESEMPIO: Calcolo errori assoluto e relativo
         x_vera : 3.1415926536e+000
       x_approx : 3.1415926000e+000
        err_ass : 5.3589793048e-008
        err_rel : 1.7058160926e-008
    cifre sign. : 7
```

Esercizio 3

Calcolare la derivata della funzione $f(x) = x^3+1$ mediante la formula del rapporto incrementale $f'(x) = \frac{f(x+h)-f(x)}{h}$ per valori di h pari a $h = 10^{-i}$ con $i = 0, 1, 2, \ldots 16$ ed $x = 1$.
Riportare i risultati ottenuti in in una tabella del tipo:

h	$f'(x)$	Errore Assoluto	Errore Relativo
...	...	...	...

Riportare inoltre l'andamento dell'errore relativo in scala semi–logaritmica.
Dire per quale h il valore calcolato risulta il "più vicino" a quello teorico ($f'(x) = 3x^2$) pari a tre.
Commentare i risultati ottenuti in base al fenomeno della cancellazione numerica.

La stampa della tabella e del grafico dell'andamento dell'errore relativo possono essere ottenuti mediante le seguenti istruzioni MatLab:

```
% Tabella
% ih        : vett. col. degli indici i
% h         : vett. col. dei passi
% yp_vera   : valore esatto della derivata
% yp_approx : vett. col. dei rapp. incr.
% err_ass   : vett. col. degli errori assoluti
% err_rel   : vett. col. degli errori relativi
tabella = [ih,h,repmat(yp_vera,length(h),1),yp_approx,err_ass,err_rel];

% Stampa a video in modo formattato della tabella
fprintf('%s\n',repmat('-',1,80));
fprintf('%4s  %8s  %13s %20s %14s %14s\n',...
    'i', 'h','df esatta','Der. prima approx',...
    'Err. Ass.','Err. Rel.');
fprintf('%s\n',repmat('-',1,80));
fprintf('%4d  %8.1e  %9.6e %20.10e %14.5e %14.5e\n',tabella');
fprintf('%s\n',repmat('-',1,80));

%Disegno errore relativo
figure;
h = semilogy(ih,err_rel,'*-');
set(h,'LineWidth',2);
```

Risultati

i	h	df esatta	Der. prima approx	Err. Ass.	Err. Rel.
0	1.0e+000	3.000000e+000	7.0000000000e+000	4.00000e+000	1.33333e+000
1	1.0e-001	3.000000e+000	3.3100000000e+000	3.10000e-001	1.03333e-001
2	1.0e-002	3.000000e+000	3.0301000000e+000	3.01000e-002	1.00333e-002
3	1.0e-003	3.000000e+000	3.0030010000e+000	3.00100e-003	1.00033e-003
4	1.0e-004	3.000000e+000	3.0003000100e+000	3.00010e-004	1.00003e-004
5	1.0e-005	3.000000e+000	3.0000300001e+000	3.00001e-005	1.00000e-005
6	1.0e-006	3.000000e+000	3.0000029998e+000	2.99980e-006	9.99933e-007
7	1.0e-007	3.000000e+000	3.0000003015e+000	3.01512e-007	1.00504e-007
8	1.0e-008	3.000000e+000	2.9999999818e+000	1.82324e-008	6.07747e-009
9	1.0e-009	3.000000e+000	3.0000002482e+000	2.48221e-007	8.27404e-008
10	1.0e-010	3.000000e+000	3.0000002482e+000	2.48221e-007	8.27404e-008
11	1.0e-011	3.000000e+000	3.0000002482e+000	2.48221e-007	8.27404e-008
12	1.0e-012	3.000000e+000	3.0002667017e+000	2.66702e-004	8.89006e-005
13	1.0e-013	3.000000e+000	2.9976021665e+000	2.39783e-003	7.99278e-004
14	1.0e-014	3.000000e+000	3.0198066270e+000	1.98066e-002	6.60221e-003
15	1.0e-015	3.000000e+000	3.5527136788e+000	5.52714e-001	1.84238e-001
16	1.0e-016	3.000000e+000	0.0000000000e+000	3.00000e+000	1.00000e+000

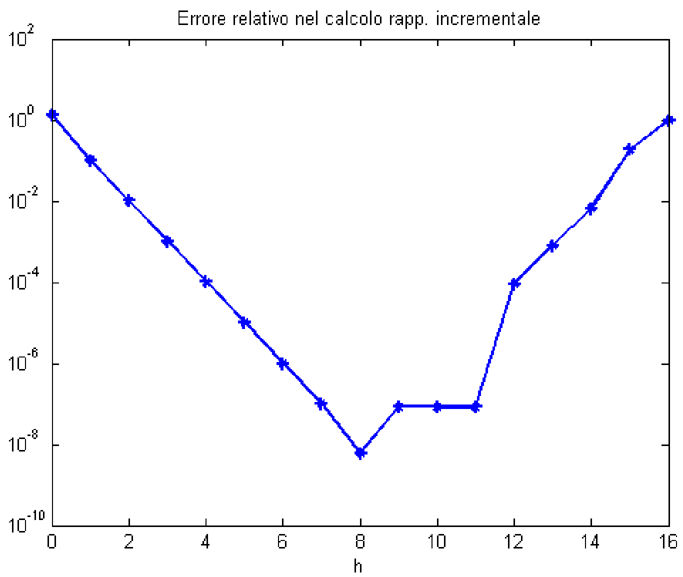

Figura 1: Risultati esercizio 3: Errore relativo.

Esercizio 4

Si calcoli il valore di questa espressione

$$y = \frac{(1+\eta)-1}{\eta}$$

al variare di η in un opportuno intervallo vicino allo zero.
Si osservi il fenomeno della cancellazione numerica rispetto al valore vero pari a 1.
Commentare i risultati ottenuti in base al fenomeno della cancellazione numerica.

La stampa di due figure in MatLab può essere effettuata mediante le seguenti istruzioni:

```
% Visualizzazione della funzione
figure;
h = plot(eta,y);
set(h,'LineWidth',2);
title('Espressione ((1+\eta)-1)/\eta');
xlabel('\eta');

% Visualizzazione dell'errore relativo
figure;
h = plot(eta,err_rel);
set(h,'LineWidth',2);
title('Errore relativo ((1+\eta)-1)/\eta');
xlabel('\eta');
```

Risultati

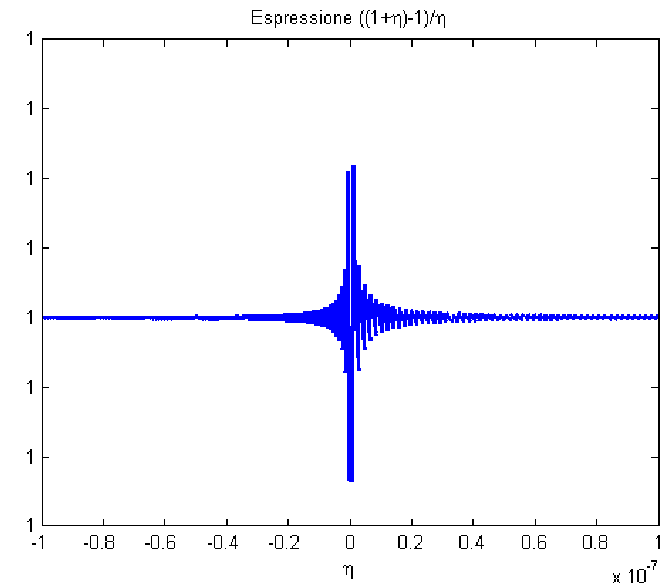

Figura 2: Risultati esercizio 4: Grafico della funzione.

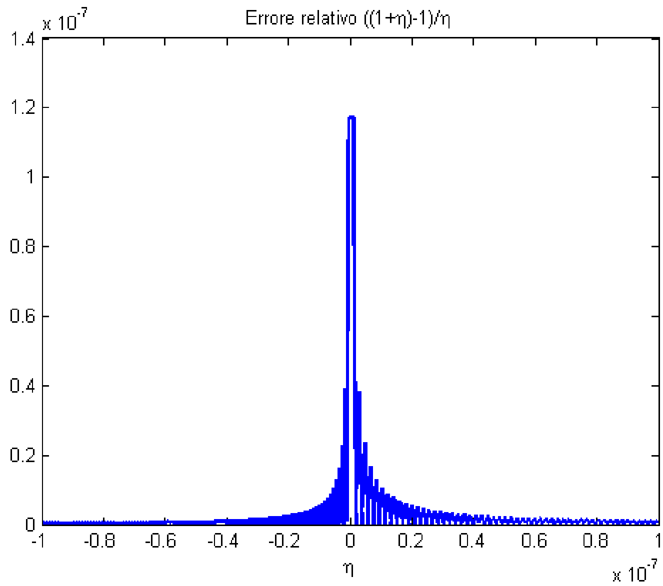

Figura 3: Risultati esercizio 4: Errore relativo.

Esercizio 5

Si osservi il fenomeno della cancellazione numerica (per x vicino a 1.0) sulle seguenti due espressioni equivalenti:

$$y = (x - 1.0)^7 \quad \text{(Formulazione numericamente stabile)}$$

$$y = x^7 - 7.0x^6 \cdots - 1.0 \quad \text{(Formulazione non numericamente stabile)}$$

Lo sviluppo di $(x - 1.0)^7$ può essere ottenuto attraverso l'utilizzo del toolbox simbolico di MatLab oppure trattato come un polinomio. Il codice seguente riporta un esempio di utilizzo del toolbox simbolico e un esempio di utilizzato dei polinomi in MatLab.

```
% Sviluppo dei coefficienti mendiante toolbox simbolico
syms x
ris = expand((x-1)^7)
ris_string = vectorize(ris)
P1 = inline(ris_string);

% numerico mediante prodotti di polinomi
% In MatLab i polinomi sono array di coefficienti memorizzati dal grado
% piu' elevato al grado meno elevato
P2 = 1;      % Polinomio costante
for i=1:7,
    P2 = conv(P2,[1 -1])      % Moltiplica P2 per (x - 1) e salvalo in P2
end

% utilizzo polyval per valutare P2 in un insieme di punti x
```

Il disegno di due funzioni su una stessa finestra può essere effettuato con il seguente codice MatLab:

```
% Disegno a video i dati
figure;
h = plot(x,y2,'b-',x,y1,'r-');
set(h,'LineWidth',2);
legend('P2','P1');
xlabel('x');
title('Valutazione di due espr. polinomiali');
axis tight
```

Risultati

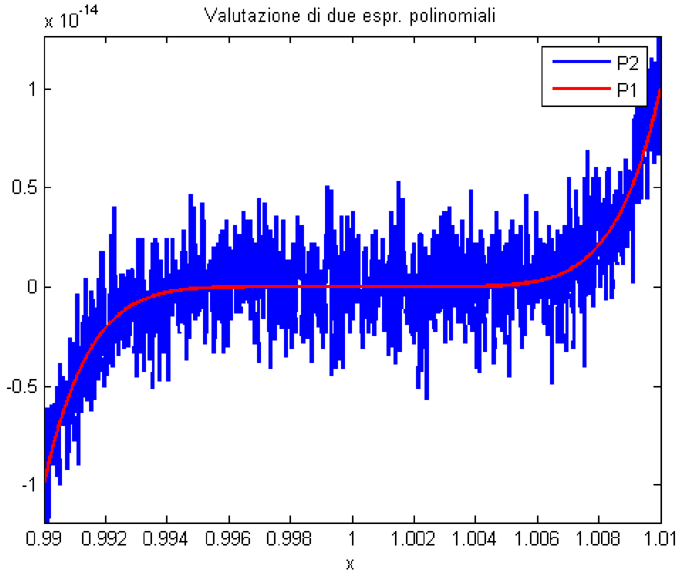

Figura 4: Risultati esercizio 5: Grafico funzioni.