Università degli studi Milano Bicocca - Dipartimento di Fisica

Esperimentazioni di Fisica Computazionale

S. Franceschina

June 22, 2025

Abstract

La presente relazione contiene gli esercizi svolti durante il corso di Esperimentazioni di Fisica Computazionale.
Gli esercizi sono stati svolti in Julia e sono raccolti in una cartella git, di cui si riporta il link:
`https://github.com/bunchi3/lab_computazionale1.git`

Contents

1 Analisi dell'errore

1.1 Esercizio 1.0.1

Considera la funzione $f(x) = e^x$ nell'intervallo $x \in [0, 1]$. Scrivi un programma che calcoli la serie approssimante:

$$g_N(x) = \sum_{n=0}^{N} \frac{x^n}{n!} \,. \tag{1.1}$$

1. Verifica che l'errore assoluto $\Delta = |f(x) - g_N(x)|$ scali approssimativamente come $x^{N+1}/(N+1)!$ per $N = 1, 2, 3, 4$.

2. L'errore Δ, nell'intervallo dato di x, differisce da $x^{N+1}/(N+1)!$. Perché accade questo e per quali valori di x?

1.1.1 Soluzione

Si sono rappresentate nel grafico 1.1 le funzioni Δ e $\frac{x^{N+1}}{(N+1)!}$, con $N = 1, 2, 3, 4$, al variare di x nell'intervallo $[0, 1]$.

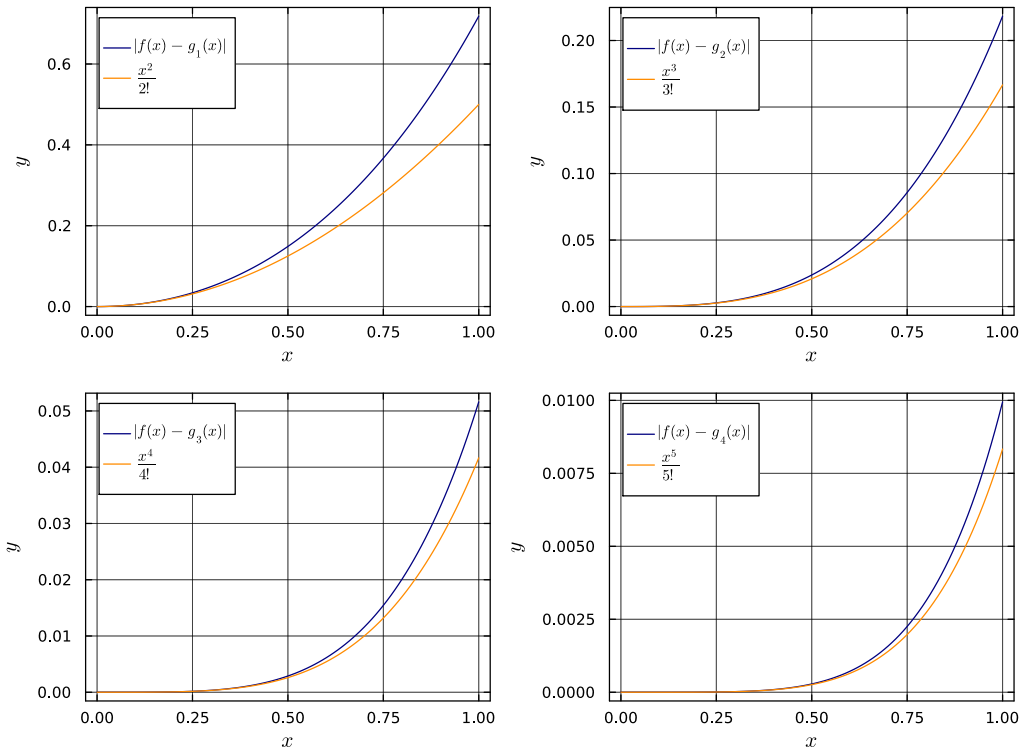


Figure 1.1: Confronto tra Δ e $\frac{x^{N+1}}{(N+1)!}$ per $N = 1, 2, 3, 4$.

Una prima considerazione è che all'aumentare di N la funzione Δ si avvicina globalmente a zero. Questo significa che la distanza tra il valore della funzione $f(x)$ presa in esame e la sua espansione di Taylor troncata all'ordine N diminuisce. In effetti ci aspettiamo che la funzione Δ sia esattamente zero nel caso in cui $N \to \infty$. Inoltre ciascuno dei grafici mostra che la funzione Δ è tanto più prossima allo zero quanto più ci si avvicina all'origine, poichè l'espansione in serie richiede $x \to 0$.
La seconda considerazione è che le funzioni Δ e $\frac{x^{N+1}}{(N+1)!}$ si avvicinano tra loro all'aumentare

di N. Il motivo è che la funzione $\frac{x^{N+1}}{(N+1)!}$ è l'ordine successivo della serie di Taylor troncata all'ordine N e può essere presa come stima per l'errore commesso nell'approssimazione di f. Non è però una stima ottima: la serie di Taylor associata a f conta infiniti termini e la sola $\frac{x^{N+1}}{(N+1)!}$ non li contiene tutti. Per questa ragione essa non coincide con Δ ma si avvicina, essendo all'aumentare di N una stima migliore dell'errore.

1.1.2 Conclusioni

L'esercizio affrontato mirava ad analizzare una delle possibili fonti di errore in contesti numerici, gli errori di approssimazione. Sono errori originati dal modo specifico con cui si risolve il problema. Nel caso proposto assumono la forma di errori di troncamento, perchè si rappresenta la funzione come una somma troncata a un certo ordine e non come una somma di infiniti termini.

1.2 Esercizio 1.2.1

Calcolare la seguente somma

$$\sum_{n=1}^{\infty} \frac{1}{n^2} = \frac{\pi^2}{6} = \lim_{N\to\infty} S(N) \quad \text{con} \quad S(N) = \sum_{n=1}^{N} \frac{1}{n^2} \tag{1.2}$$

1. Calcolarla in single precision utilizzando l'ordinamento normale, $n = 1, 2, 3, \ldots, N$.
2. Calcolarla in single precision utilizzando l'ordinamento inverso, $n = N, \ldots, 2, 1$.
3. Studiare la convergenza di entrambe le implementazioni in funzione di N tracciando il grafico di $|S(N) - \pi^2/6|$.
4. Ripetere i punti da 1 a 3 utilizzando double precision.

1.2.1 Soluzione

Procediamo prendendo in esame i due casi, single precision e double precision.

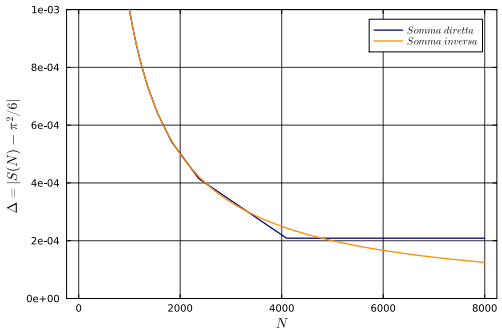


Figure 1.2: Δ in Float32.

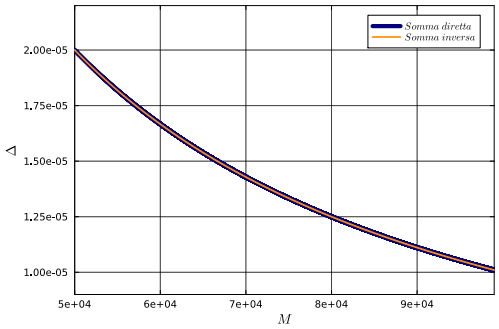


Figure 1.3: Ingrandimento di Δ in Float64

Single precision: Si riportano in figura 1.2 gli errori di troncamento della serie $\sum_{n=1}^{\infty} \frac{1}{n^2}$, laddove N è il troncamento. La somme sono state eseguite con variabili di tipo Float32 (single precision).
In figura 1.2 notiamo due comportamenti degni di nota. Il primo riguarda la curva blu, che arresta la sua discesa poco dopo $N = 4000$. Il secondo riguarda il discostarsi delle due curve a

causa dell'andamento a linea spezzata della curva blu.
La spiegazione del primo fenomeno è la seguente: nel caso della somma con ordinamento diretto, all'aumento dell'indice di somma, gli addendi sono sempre più piccoli. In particolare la somma comincia da 1, e sappiamo che dovrà raggiungere $\frac{\pi^2}{6} \simeq 1.64$, perciò per tutto il processo la somma parziale resterà nell'intervallo $[1, 2)$. In tale intervallo la distanza tra due floating point numbers è $\varepsilon_{mach} = 2^{-23}$. Osserviamo il grafico 1.2: la curva blu comincia ad essere costante a partire da $N = 4096 = 2^{12}$. L'addendo corrispondente è $\frac{1}{N^2} = 2^{-24}$. Essendo più piccolo di ε_{mach}, la distanza tra due floating point numbers in $[1, 2)$, viene considerato come 0.
La spiegazione del secondo fenomeno è simile: in questo caso alla somma viene aggiunto un addendo che è più piccolo del precedente, ma non così tanto da essere più piccolo di ε_{mach}. In questo modo alla somma viene aggiunto un valore arrotondato rispetto al suo valore vero. Il successivo addendo, pur essendo differente in teoria, a causa dell'arrotondamento risulta essere uguale al precedente. Ciò fa in modo che venga sommato sempre lo stesso numero, e così l'andamento della curva blu è lineare.
I due fenomeni non si verificano per la somma con ordinamento inverso perchè il primo numero della somma è molto piccolo. Questo fa in modo che i successivi floating point number siano molto vicini tra loro, e così sommare l'addendo successivo fa cadere la somma parziale vicina al floating point number che la approssima.

Si può mostrare matematicamente che gli errori $\Delta = |S(N) - \pi^2/6|$ scalano come $O(\frac{1}{N})$. Per verificare questa affermazione possiamo interpolare la curva degli errori con un metodo che vedremo nella sezione 2. Sulla base delle considerazioni di questa sezione, possiamo aspettarci che sia più conveniente interpolare la curva con ordinamento inverso, perchè i valori che la compongono sono meno affetti da errori di tipo numerico. Utilizzando il modello linearizzato $log(\Delta) = -clog(N)$ con $c = 1$ si ottiene per l'ordinamento inverso $c_{fit} = 1.00003$ e per quello diretto $c_{fit} = 0.986$. Come atteso il caso di ordinamento diretto restituisce un risultato peggiore di quello inverso.

Double precision: Si riporta in figura 1.3 un ingrandimento del grafico degli errori di troncamento della serie $\sum_{n=1}^{\infty} \frac{1}{n^2}$, laddove N è il troncamento. La somme sono state eseguite con variabili di tipo Float64 (double precision). Nel caso di figura 1.3 possiamo notare che le due curve sono sovrapposte, nonostante l'ingrandimento a valori di $N \in [5000, 10000]$. Per osservare un fenomeno simile a quello di figura 1.2 dovremmo raggiungere valori di $N = 2^{26}$. Infatti l'approssimazione non migliora quando vengono sommati numeri dell'ordine di ε_{mach} per double precision, cioè $\frac{1}{N^2} \simeq 2^{-52}$.

1.2.2 Conclusioni

Lo scopo dell'esercizio è stato quello di studiare un algoritmo affetto da errori di arrotondamento. Per evitare tali errori si è riformulato il problema della somma di N termini utilizzando la somma inversa. In alternativa si può aumentare la precisione della rappresentazione, fintantochè sia possibile.

1.3 Esercizio 1.4.1

(a) In statistica, definiamo la varianza di un campione di valori $x_1, \ldots, x_n$ come

$$\sigma^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \overline{x})^2, \qquad \overline{x} = \frac{1}{n} \sum_{i=1}^{n} x_i. \tag{1.3}$$

Scrivi una funzione che prenda in input un vettore x di lunghezza arbitraria e restituisca σ^2 calcolata con la formula sopra.

(b) La formula 1.3 ha lo svantaggio di scorrere due volte il vettore dei dati. Considera la formula 1.4 a un ciclo:

$$\sigma^2 = \frac{1}{n-1}\left(u - \frac{1}{n}v^2\right), u = \sum_{i=1}^{n} x_i^2, \qquad v = \sum_{i=1}^{n} x_i. \tag{1.4}$$

Prova entrambe le formule per i seguenti dataset, ciascuno dei quali ha varianza esattamente uguale a 1. Esegui i calcoli sia in single che in double precision.

$$x_1 = [1 \cdot 10^3,\ 1 + 10^3,\ 2 + 10^3] \qquad x_2 = [1 \cdot 10^6,\ 1 + 10^6,\ 2 + 10^6]$$
$$x_3 = [1 \cdot 10^7,\ 1 + 10^7,\ 2 + 10^7] \qquad x_4 = [1 \cdot 10^8,\ 1 + 10^8,\ 2 + 10^8]$$

1.3.1 Soluzione

(a) Si è testato il codice che implementa la formula 1.3 su tre vettori contenenti elementi di tipo Float64. Il primo contiene solo numeri 1, il secondo contiene numeri generati a partire da una distribuzione uniforme, il terzo numeri generati a partiere da una distribuzione normale. I valori sono riportati in tabella 1.1.

Table 1.1: Risultati del test della formula 1.3 su tre vettori di tipo Float64

Vettore	**Lunghezza**	**Varianza attesa**	**Varianza (output)**
Solo uno	10^3	0.000	0.000
Uniforme	10^8	0.083	0.083
Normale	10^8	1.000	1.000

I risultati di tabella 1.1 mostrano che l'implementazione del codice è corretta.

(b) Si è testato il codice che implementa la formula 1.4 su quattro vettori contenenti elementi di tipo Float32, Float64 e LongDouble. I risultati sono riportati in tabella 1.2.

Table 1.2: Valori di varianza. Metodo 1: equazione 1.3. Metodo 2: equazione 1.4.

Set	σ^2 **attesa**	**Float32**		**Float64**		**Long Double**	
		Metodo 1	Metodo 2	Metodo 1	Metodo 2	Metodo 1	Metodo 2
x_1	1.0	1.0	1.0	1.0	1.0	1.0	1.0
x_2	1.0	1.0	-131072.0	1.0	1.0	1.0	1.0
x_3	1.0	1.0	0.0	1.0	1.0	1.0	1.0
x_4	1.0	0.0	0.0	1.0	0.0	1.0	1.0

I risultati di tabella 1.2 mostrano che il calcolo della varianza con la formula 1.4 è più affetto dalla mancanza di precisione della rappresentazione dei numeri durante il calcolo. All'aumento della precisione di rappresentazione, il risultato della formula 1.4 si avvicina a quello atteso.
C'è da notare che anche la formula 1.3 è affetta dalla stessa problematica, ma in maniera meno evidente. Basti osservare il risultato del metodo 1 per il vettore x_4 in Float32, che è 0.0, contro l'atteso 1.0.
Parte del problema risiede nella cancellazione numerica. Infatti, per valori degli elementi del vettore molto grandi, la differenza tra il quadrato della sommma e la somma dei quadrati è

Table 1.3: Differenza $u - \frac{v^2}{n}$ per ciascun dataset.

Dataset	Single	Double	Long Double
x_1	2.0	2.0	2.0
x_2	-262144.0	2.0	2.0
x_3	0.0	2.0	2.0
x_4	0.0	0.0	2.0

molto piccola e incorre in problema di cancellazione. Per verificare questa affermazione, si riportano in tabella 1.3 i valori della differenza $u - \frac{v^2}{n}$ per la formula 1.4.

Leggendo la tabella 1.3 possiamo notare che per il dataset x_1 la differenza è esattamente 2.0, il risultato atteso. Per il dataset x_2 la differenza è molto grande, -262144.0, indice di un significativo problema di cancellazione numerica. Per il dataset x_3 e x_4 la differenza è 0.0. Evidentemente i singoli valori u e $\frac{v^2}{n}$ distano tra loro meno del floating point number più vicino, la loro differenza è quindi zero.
Per concludere, possiamo dire che la formula 1.4 è più efficiente in termini di tempo di calcolo, ma è più affetta da errori numerici rispetto alla formula 1.3.

1.3.2 Conclusioni

L'esercizio svolto è un buon esempio di come più risoluzioni dello stesso problema non siano equivalenti. Nel primo caso si è trovata una formula che portasse alla soluzione senza grandi errori, pur essendo costosa dal punto di vista computazionale. Ciò mostra che il problema non è intrinsecamente mal condizionato. Nel secondo caso la formula è meno costosa dal punto di vista computazionale ma costituita da un passaggio mal condizionato, la sottrazione. Vediamo perciò che nel caso in cui il problema affrontato non sia instabile per natura è possibile riformulare l'algoritmo che lo risolve in maniera stabile.

1.4 Esercizio 1.4.2

(a) Sia $f(x) = \frac{e^x - 1}{x}$. Trova il numero di condizionamento $\kappa_f(x)$. Qual è il massimo di $\kappa_f(x)$ nell'intervallo $-1 \leq x \leq 1$?

(b) Usa l'algoritmo "naive"

$$f(x) = \frac{e^x - 1}{x} \tag{1.5}$$

per calcolare $f(x)$ per $x = 10^{-3}, 10^{-4}, 10^{-5}, \ldots, 10^{-16}$.

(c) Crea un secondo algoritmo utilizzando i primi n termini della serie di Mc Laurin, cioè

$$p(x) = 1 + \frac{1}{2!}x + \frac{1}{3!}x^2 + \cdots + \frac{1}{(n+1)!}x^n. \tag{1.6}$$

Valutalo sugli stessi valori di x del punto (b). Per farlo devi scegliere un valore per n. Verifica la stabilità del risultato al variare di n. Avresti potuto indovinare un buon valore di n fin dall'inizio?

(d) Confronta i risultati delle due implementazioni in funzione di x. Quale algoritmo pensi sia più accurato, e perché?

1.4.1 Soluzione

(a) Il numero di condizionamento di una funzione f è definito come:

$$\kappa_f(x) = \left| \frac{xf'(x)}{f(x)} \right| .$$

Per la funzione $f(x) = \frac{e^x - 1}{x}$, calcoliamo la derivata:

$$f'(x) = \frac{e^x x - (e^x - 1)}{x^2} = \frac{e^x(x-1)+1}{x^2} .$$

Quindi il numero di condizionamento diventa:

$$\kappa_f(x) = \left| \frac{x\left(\frac{e^x(x-1)+1}{x^2}\right)}{\frac{e^x-1}{x}} \right| = \left| \frac{e^x(x-1)+1}{(e^x-1)x} \right| = \left| \frac{e^x x}{(e^x-1)} - 1 \right| .$$

La derivata di $k_f(x)$ non si annulla mai nell'intervallo $-1 \leq x \leq 1$, e $\kappa_f(0) = 0$. Dato che $\kappa_f(x)$ è monotona crescente per $x \geq 0$ e monotona decrescente per $x < 0$, il massimo si ha in $x = 1$, dove assume il valore $\kappa_f(1) = 0.58198$.

(b)-(c): Si riportano in figura 1.4 i risultati dei due algoritmi. La dicitura $f(x)$ si riferisce alla funzione calcolata con l'algoritmo "naive", mentre $p_n(x)$ si riferisce alla funzione calcolata con la serie di Mc Laurin, fino al termine n.

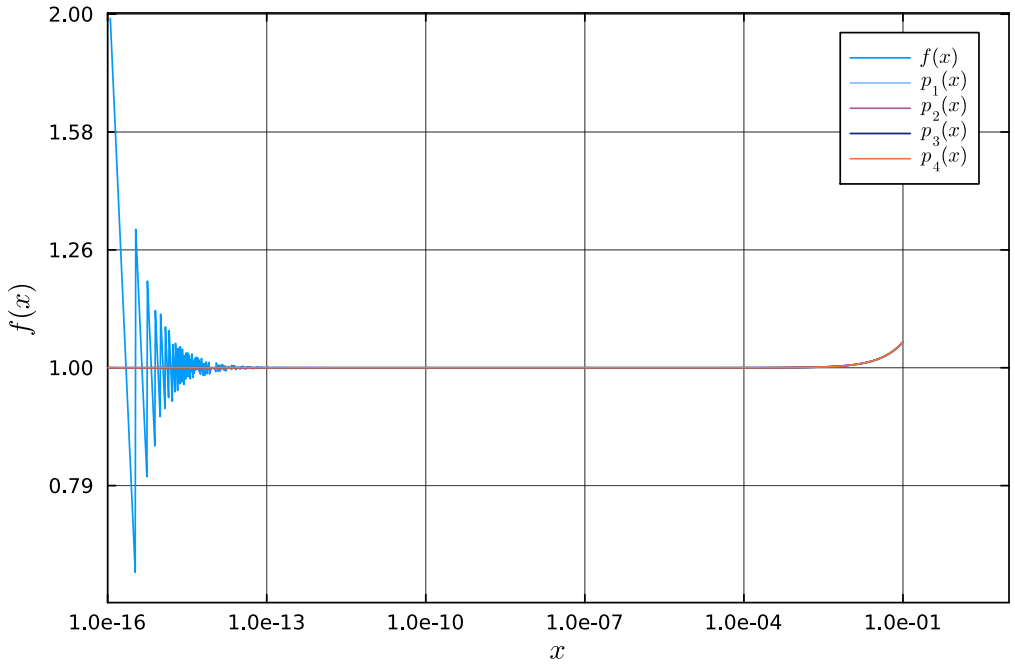


Figure 1.4: Confronto tra $f(x)$ e $p_n(x)$ al variare di x.

Per uno studio più accurato, si riportano nelle figure 1.5 e 1.6 ingrandimenti delle curve di figura 1.4.

Per quanto riguarda il comportamento della funzione $f(x)$ calcolata con l'algoritmo 1.5, possiamo notare che per valori di x piccoli la funzione assume valori non corretti, nonostante $\kappa_f(x)$ abbia un massimo non molto grande e peraltro non assunto nell'intorno di zero. La spiegazione di questo fenomeno è che l'algoritmo 1.5, nonostante non abbia $\kappa_f(x)$ complessivo

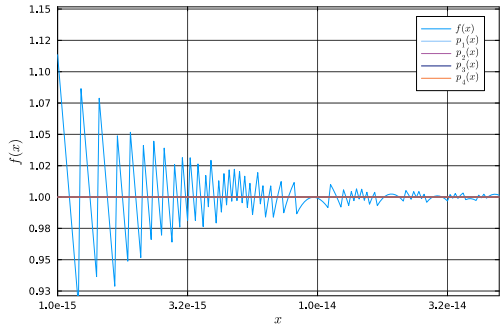


Figure 1.5: Ingrandimento. $x \in [10^{-15}, 10^{-14}]$.

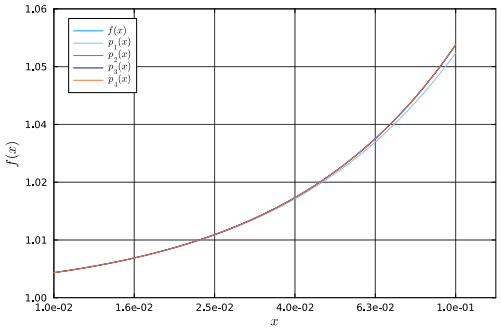


Figure 1.6: Ingrandimento. $x \in [10^{-2}, 10^{-1}]$.

molto grande, è affetto da errori di cancellazione numerica a causa della forma in cui è scritta la funzione, ma non a causa del fatto che il problema sia intrinsecamente instabile. Si può perciò intuire l'esistenza di una possibile riformulazione dell'algoritmo di calcolo che non sia affetto da cancellazione numerica. Una riformulazione possibile è l'utilizzo della serie di Mc Laurin, implementata nell'algoritmo $p_n(x)$.
L'algoritmo 1.6 non essendo affetto da cancellazione numerica restituisce valori corretti per x piccoli. Nella regione evidenziata da figura 1.6 restituisce valori errati perchè ci allontaniamo dalla regione di convergenza della serie di Mc Laurin.
Osservando la figura 1.6 si nota che per valori di x fissati l'algoritmo 1.6 è stabile, cioè le curve che descrivono i valori di $p_n(x)$ sono sempre più vicine tra loro all'aumentare di n, fino a sovrapporsi.
In linea di principio si sarebbe potuto stimare a priori un buon valore di n da utilizzare per calcolare $p_n(x)$. La serie di Mc Laurin è una stima peggiore di f quanto più ci si allontana da zero. Nel caso di studio valutiamo la distanza tra i valori di $p_n(x)$ al variare di n, fissato $x = 0.1$, punto sufficientemente lontano dall'origine. Si è rappresentato in figura 1.7 l'andamento della distanza tra i valori di $p_n(0.1)$ e $p_{n+1}(0.1)$ al variare di n. In formula:

$$\Delta_{pn} = |p_n(0.1) - p_{n+1}(0.1)| \,. \tag{1.7}$$

Ai fini di questo esercizio possiamo dire che $n = 2$ è già un buon valore perchè le distanze tra i polinomi successivi non sono apprezzabili nel range $[1, 1.06]$, caratteristico della figura 1.6. È chiaro che volendo raggiungere precisione massima occorrerà scegliere $n = 10$, dato che Δ_{pn} corrispondente è più piccolo di ε_{mach}

(d): Come confronto tra i due algoritmi si riporta in figura 1.8 il grafico dell'errore tra i due algoritmi, cioè

$$\Delta(x) = |f(x) - p_n(x)| \,. \tag{1.8}$$

La figura 1.8 mostra un'evidente differenza tra i due algoritmi nella regione di instabilità numerica dell'algoritmo 1.5, cioè per valori di x piccoli. E' interessante notare che la distanza tra i due algoritmi diminuisce all'aumentare di x, ma torna ad aumentare a partire da $x = 10^{-5}$, laddove l'algoritmo 1.6 esce dalla regione di convergenza della serie di Mc Laurin.

Conclusioni Questo esercizio evidenzia contemporaneamente le due possibili fonti di errore in contesti numerici, gli errori di arrotondamento per valori di x piccoli, gli errori di troncamento per x grandi. Nel merito dell'esercizio gli errori di arrotondamento hanno luogo a causa di un passaggio mal condizionato nel calcolo di $f(x)$; gli errori di troncamento hanno luogo a causa della necessità di interrompere la serie a N finito.

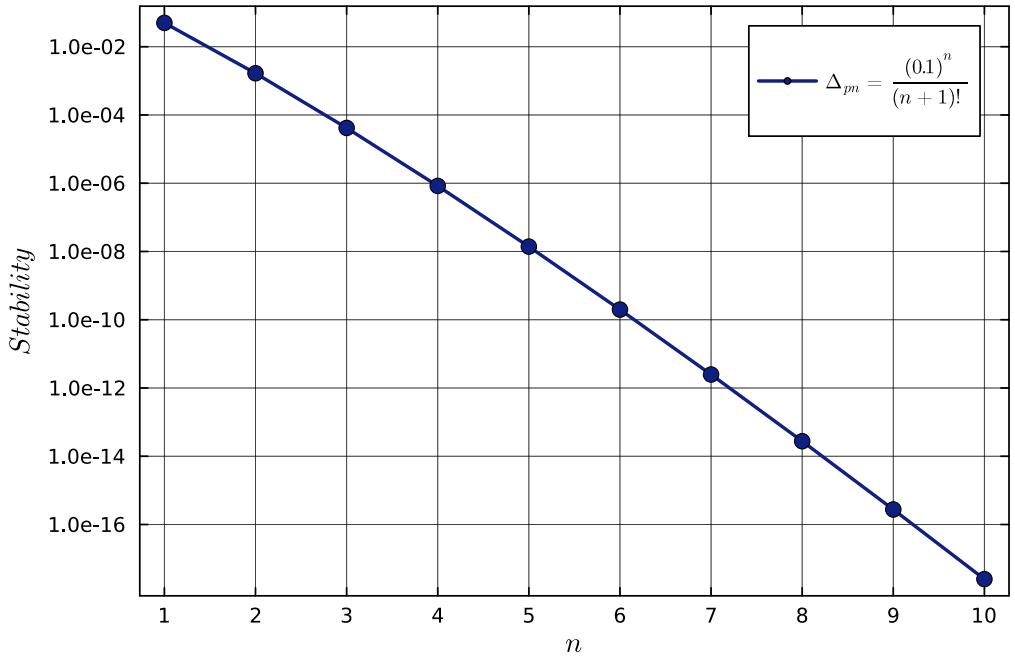


Figure 1.7: Distanza tra $p_n(0.1)$ e $p_{n+1}(0.1)$ al variare di n.

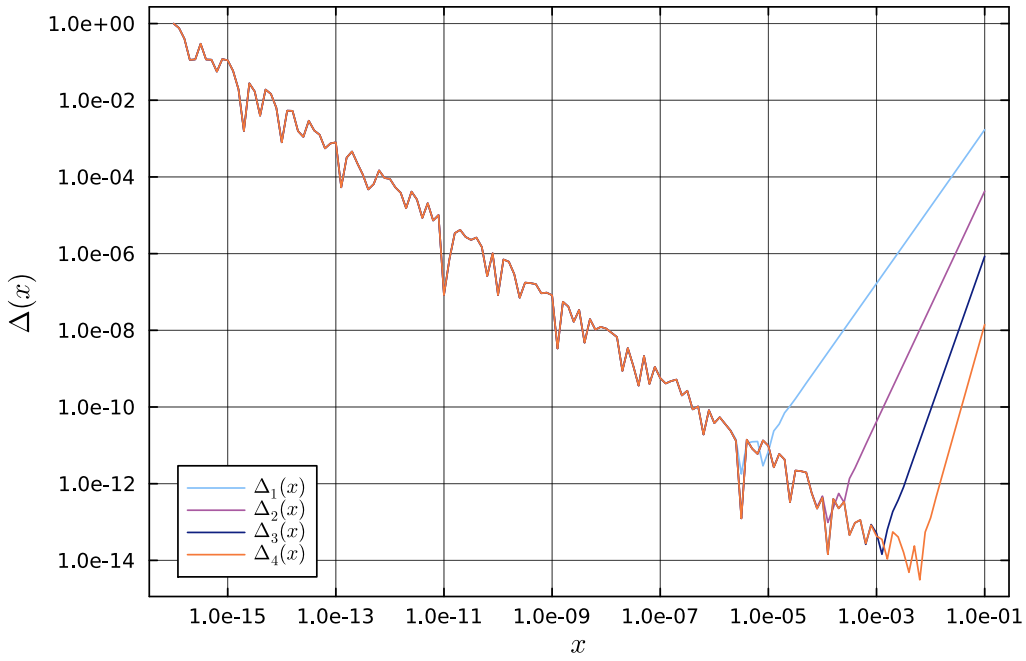


Figure 1.8: Distanza tra $f(x)$ e $p_n(x)$.

2 Sistemi lineari

2.1 Esercizi 2.1.1, 2.1.2, 2.1.3, 2.1.4

Scrivi una funzione che esegua la sostituzione in avanti su una matrice triangolare inferiore $n \times n$.
2. Testa il tuo codice sui seguenti sistemi lineari:

(a) $\begin{bmatrix} -2 & 0 & 0 \\ 1 & -1 & 0 \\ 3 & 2 & 1 \end{bmatrix} \mathbf{x} = \begin{bmatrix} -4 \\ 2 \\ 1 \end{bmatrix}$ (b) $\begin{bmatrix} 4 & 0 & 0 & 0 \\ 1 & -2 & 0 & 0 \\ -1 & 4 & 4 & 0 \\ 2 & -5 & 5 & 1 \end{bmatrix} \mathbf{x} = \begin{bmatrix} -4 \\ 1 \\ -3 \\ 5 \end{bmatrix}$

Puoi verificare la soluzione risolvendo il sistema a mano e/o calcolando $\mathbf{Lx} - \mathbf{b}$.

3. Scrivi una funzione che esegua la sostituzione all'indietro su una matrice triangolare superiore $n \times n$.

4. Testa il tuo codice sui seguenti sistemi lineari:

(a) $\begin{bmatrix} 3 & 1 & 0 \\ 0 & -1 & -2 \\ 0 & 0 & 3 \end{bmatrix} \mathbf{x} = \begin{bmatrix} 1 \\ 1 \\ 6 \end{bmatrix}$ (b) $\begin{bmatrix} 3 & 1 & 0 & 6 \\ 0 & -1 & -2 & 7 \\ 0 & 0 & 3 & 4 \\ 0 & 0 & 0 & 5 \end{bmatrix} \mathbf{x} = \begin{bmatrix} 4 \\ 1 \\ 1 \\ 5 \end{bmatrix}$

Puoi verificare la soluzione risolvendo il sistema a mano e/o calcolando $\mathbf{Ux} - \mathbf{b}$.

2.1.1 Soluzione

Per risolvere i sistemi lineari proposti, si sono implementate le funzioni di sostituzione in avanti e all'indietro. Di seguito sono riportati i risultati delle soluzioni.

Sostituzione in avanti:

- Per il sistema (a) si ha $\mathbf{x} = \begin{bmatrix} -4.0 & 2.0 & 1.0 \end{bmatrix}$.
- Per il sistema (b) si ha $\mathbf{x} = \begin{bmatrix} -4.0 & 1.0 & -3.0 & 5.0 \end{bmatrix}$.

Risolvendo i sistemi a mano, i valori sono:

- Per il sistema (a) si ha $\mathbf{x} = \begin{bmatrix} -4 & 2 & 1 \end{bmatrix}$.
- Per il sistema (b) si ha $\mathbf{x} = \begin{bmatrix} -4 & 1 & -3 & 5 \end{bmatrix}$.

Sostituzione all'indietro:

- Per il sistema (a) si ha $\mathbf{x} = \begin{bmatrix} 1.0 & 1.0 & 6.0 \end{bmatrix}$.
- Per il sistema (b) si ha $\mathbf{x} = \begin{bmatrix} 4.0 & 1.0 & 1.0 & 5.0 \end{bmatrix}$.

Risolvendo i sistemi a mano, i valori sono:

- Per il sistema (a) si ha $\mathbf{x} = \begin{bmatrix} 1 & 1 & 6 \end{bmatrix}$.
- Per il sistema (b) si ha $\mathbf{x} = \begin{bmatrix} 4 & 1 & 1 & 5 \end{bmatrix}$.

I risultati confermano la correttezza delle implementazioni delle funzioni di sostituzione in avanti e all'indietro.

2.2 Esercizi 2.3.1, 2.3.2, 2.3.4

1. Scrivi una funzione che esegua la fattorizzazione LU di una matrice $n \times n$.
2. Per ciascuna matrice, calcola la fattorizzazione LU e verifica la correttezza.

(A) $\begin{bmatrix} 2 & 3 & 4 \\ 4 & 5 & 10 \\ 4 & 8 & 2 \end{bmatrix}$ **(B)** $\begin{bmatrix} 6 & -2 & -4 & 4 \\ 3 & -3 & -6 & 1 \\ -12 & 8 & 21 & -8 \\ -6 & 0 & -10 & 7 \end{bmatrix}$ **(C)** $\begin{bmatrix} 1 & 4 & 5 & -5 \\ -1 & 0 & -1 & -5 \\ 1 & 3 & -1 & 2 \\ 1 & -1 & 5 & -1 \end{bmatrix}$

4. Calcola il determinante delle matrici dell'esercizio 2.3.2 tramite la fattorizzazione LU.

2.2.1 Soluzione

Punti 1 e 2: Si sono implementati gli algoritmi di fattorizzazione LU, e si sono testati sui tre sistemi proposti. Le scomposizioni ottenute sono le seguenti:

$$\mathbf{L}_A = \begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 2.0 & 1.0 & 0.0 \\ 2.0 & -2.0 & 1.0 \end{bmatrix} \qquad \mathbf{U}_A = \begin{bmatrix} 2.0 & 3.0 & 4.0 \\ 0.0 & -1.0 & 2.0 \\ 0.0 & 0.0 & -2.0 \end{bmatrix}$$

$$\mathbf{L}_B = \begin{bmatrix} 1.0 & 0.0 & 0.0 & 0.0 \\ 0.5 & 1.0 & 0.0 & 0.0 \\ -2.0 & -2.0 & 1.0 & 0.0 \\ -1.0 & 1.0 & -2.0 & 1.0 \end{bmatrix} \qquad \mathbf{U}_B = \begin{bmatrix} 6.0 & -2.0 & -4.0 & 4.0 \\ 0.0 & -2.0 & -4.0 & -1.0 \\ 0.0 & 0.0 & 5.0 & -2.0 \\ 0.0 & 0.0 & 0.0 & 8.0 \end{bmatrix}$$

$$\mathbf{L}_C = \begin{bmatrix} 1.0 & 0.0 & 0.0 & 0.0 \\ -1.0 & 1.0 & 0.0 & 0.0 \\ 1.0 & -0.25 & 1.0 & 0.0 \\ 1.0 & -1.25 & -1.0 & 1.0 \end{bmatrix} \qquad \mathbf{U}_C = \begin{bmatrix} 1.0 & 4.0 & 5.0 & -5.0 \\ 0.0 & 4.0 & 4.0 & -10.0 \\ 0.0 & 0.0 & -2.0 & 6.0 \\ 0.0 & 0.0 & 0.0 & 2.0 \end{bmatrix}$$

Per verificare la correttezza delle scomposizioni, si è calcolato il prodotto **LU** per ciascuna matrice e lo si è sottratto alla matrice originale. Si sono ottenute matrici di zeri, in tutti i casi, confermando la correttezza dell'algoritmo.

Punto 4: Si è calcolato il determinante delle matrici dell'esercizio 2.3.2 utilizzando la loro fattorizzazione LU. Ricordando le proprietà del determinante e che la matrice **L** è triangolare inferiore con elementi diagonali unitari, si ha che:

$$\det(\mathbf{M}) = \det(\mathbf{LU}) = \det(\mathbf{L}) \cdot \det(\mathbf{U}) = \det(\mathbf{U}) \,.$$

Per il calcolo del determinante delle matrici proposte si è eseguito il prodotto degli elementi della diagonale principale della matrice **U**. A titolo di confronto si è calcolato il determinante delle matrici con la funzione `det()` di Julia. I risultati sono riportati in tabella 2.1.

Table 2.1: Determinanti delle matrici dell'esercizio 2.

Matrice	**Determinante (LU)**	**Determinante (Julia)**
A	4.0	4.0
B	-480.0	-480.0
C	80.0	80.0

2.3 Esercizio 2.3.3

Le matrici

$$\mathbf{T}(x, y) = \begin{bmatrix} 1 & 0 & x \\ 0 & 1 & y \\ 0 & 0 & 1 \end{bmatrix}, \qquad \mathbf{R}(\theta) = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

sono usate per rappresentare traslazioni e rotazioni di punti del piano in computer grafica. Per quanto segue, sia

$$\mathbf{A} = \mathbf{T}(3, -1)\, \mathbf{R}(\pi/5)\, \mathbf{T}(-3, 1), \qquad \mathbf{z} = \begin{bmatrix} 2 \\ 2 \\ 1 \end{bmatrix}.$$

(a) Calcolare $\mathbf{b} = \mathbf{Az}$.

(b) Trovare la fattorizzazione LU di $\mathbf{A}$.

(c) Usare i fattori e le sostituzioni triangolari per risolvere $\mathbf{Ax} = \mathbf{b}$ e calcolare $\mathbf{x} - \mathbf{z}$.

2.3.1 Soluzione

(a) Si è calcolato il prodotto tra le matrici $\mathbf{A}$ e $\mathbf{z}$, ottenendo:

$$\mathbf{b} = \mathbf{Az} = \begin{bmatrix} 3.95 \\ 2.01 \\ 1.0 \end{bmatrix}.$$

(b) Si è calcolata la fattorizzazione LU della matrice $\mathbf{A}$, ottenendo:

$$\mathbf{L} = \begin{bmatrix} 1.0 & 0.0 & 0.0 \\ -0.8 & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}, \qquad \mathbf{U} = \begin{bmatrix} 0.81 & 0.59 & 1.16 \\ 0.0 & 1.24 & 2.42 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}.$$

(c) Si è risolto il sistema $\mathbf{Ax} = \mathbf{b}$ utilizzando le sostituzioni triangolari. Si ottiene, con le dovute approssimazioni:

$$\mathbf{x} = \begin{bmatrix} 2.0 \\ 2.0 \\ 1.0 \end{bmatrix}, \qquad \mathbf{x} - \mathbf{z} = \begin{bmatrix} -2.22 \cdot 10^{-16} \\ -4.44 \cdot 10^{-16} \\ 0.0 \end{bmatrix}.$$

2.4 Esercizi 2.4.1, 2.4.2

1. Scrivi un programma che esegua la decomposizione di Cholesky di una matrice $n \times n$.
2. Per ciascuna matrice, utilizza la decomposizione di Cholesky per determinare se è definita positiva.

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 4 & 5 \\ -1 & 5 & 10 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 4 & 5 \\ 1 & 5 & 10 \end{bmatrix} \qquad \mathbf{C} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 4 & 5 \\ 1 & 5 & 1 \end{bmatrix}$$

$$\mathbf{D} = \begin{bmatrix} 6 & 2 & 1 & 0 \\ 2 & 6 & 2 & 1 \\ 1 & 2 & 5 & 2 \\ 0 & 1 & 2 & 4 \end{bmatrix} \qquad \mathbf{E} = \begin{bmatrix} 4 & 1 & 2 & 7 \\ 1 & 1 & 3 & 1 \\ 2 & 3 & 5 & 3 \\ 7 & 1 & 3 & 1 \end{bmatrix}$$

2.4.1 Soluzione

Si è implementato l'algoritmo di decomposizione di Cholesky, e si sono testate le matrici proposte. L'algoritmo è giunto a conclusione per le matrici $\mathbf{A}$, $\mathbf{B}$ e $\mathbf{D}$, ma ha restituito un messaggio di errore per le matrici $\mathbf{C}$ e $\mathbf{E}$, indicando che non sono definite positive. Si riportano i risultati delle decomposizioni giunte a termine, con i dovuti arrotondamenti.

$$\mathbf{R_A} = \begin{bmatrix} 1.0 & 0.0 & -1.0 \\ 0.0 & 2.0 & 2.5 \\ 0.0 & 0.0 & 1.7 \end{bmatrix} \qquad \mathbf{R_B} = \begin{bmatrix} 1.0 & 0.0 & 1.0 \\ 0.0 & 2.0 & 2.5 \\ 0.0 & 0.0 & 1.7 \end{bmatrix} \qquad \mathbf{R_D} = \begin{bmatrix} 2.45 & 0.82 & 0.41 & 0.00 \\ 0.00 & 2.29 & 0.73 & 0.41 \\ 0.00 & 0.00 & 2.06 & 0.82 \\ 0.00 & 0.00 & 0.00 & 1.84 \end{bmatrix}$$

2.5 Esercizio 2.5.1

Consideriamo la matrice:

$$\mathbf{A} = \begin{bmatrix} -\epsilon & 1 \\ 1 & -1 \end{bmatrix}$$

Se $\varepsilon = 0$, la fattorizzazione LU senza pivoting parziale fallisce per $\mathbf{A}$. Ma se $\varepsilon \neq 0$, possiamo procedere senza pivoting, almeno in linea di principio.

(a) Costruisci $\mathbf{b} = \mathbf{Ax}$ prendendo $\epsilon = 10^{-12}$ per la matrice $\mathbf{A}$ e $\mathbf{x} = [1, 1]$ come soluzione.

(b) Fattorizza la matrice usando la decomposizione LU senza pivoting e risolvi numericamente per $\mathbf{x}$. Quanto è accurato il risultato?

(c) Verifica la fattorizzazione LU calcolando $\mathbf{A} - \mathbf{LU}$. Funziona?

(d) Ripeti per $\epsilon = 10^{-20}$. Quanto è accurato ora il risultato?

(e) Calcola il numero di condizionamento della matrice $\mathbf{A}$; puoi scegliere la norma che preferisci. Il sistema $\mathbf{Ax} = \mathbf{b}$ è mal condizionato?

(f) Calcola le matrici $\mathbf{L}$ e $\mathbf{U}$ della fattorizzazione LU di $\mathbf{A} = \mathbf{LU}$ a mano.

(g) Calcola i numeri di condizionamento di $\mathbf{L}$ e $\mathbf{U}$.

2.5.1 Soluzione

Per poter operare un confronto tra i risultati ottenuti con $\epsilon = 10^{-12}$ e $\epsilon = 10^{-20}$, si è scelto di trattare il punto (d) insieme ai punti (a), (b) e (c). Il pedice 1 è legato al caso $\epsilon = 10^{-12}$, il pedice 2 al caso $\epsilon = 10^{-20}$.

(a) Si è calcolato il prodotto tra la matrice $\mathbf{A}$ e il vettore $\mathbf{x}$, ottenendo, a meno di arrotondamenti, i vettori:

$$\mathbf{b_1} = \begin{bmatrix} 1.0 \\ 0.0 \end{bmatrix}. \qquad \mathbf{b_2} = \begin{bmatrix} 1.0 \\ 0.0 \end{bmatrix}.$$

(b) Si è calcolata la fattorizzazione LU della matrice $\mathbf{A}$, ottenendo:

$$\mathbf{L_1} = \begin{bmatrix} 1.0 & 0.0 \\ -1.0 \cdot 10^{12} & 1.0 \end{bmatrix}, \qquad \mathbf{U_1} = \begin{bmatrix} -1.0 \cdot 10^{-12} & 1.0 \\ 0.0 & 1.0 \cdot 10^{12} \end{bmatrix},$$

$$\mathbf{L_2} = \begin{bmatrix} 1.0 & 0.0 \\ -1.0 \cdot 10^{20} & 1.0 \end{bmatrix}, \qquad \mathbf{U_2} = \begin{bmatrix} -1.0 \cdot 10^{-20} & 1.0 \\ 0.0 & 1.0 \cdot 10^{20} \end{bmatrix}.$$

Si è risolto il sistema $\mathbf{Ax} = \mathbf{b}$ utilizzando le sostituzioni triangolari, ottenendo:

$$\mathbf{x_1} = \begin{bmatrix} 1.0 \\ 1.0 \end{bmatrix}, \qquad \mathbf{x_2} = \begin{bmatrix} 0.0 \\ 1.0 \end{bmatrix}.$$

(c) Si è calcolato il prodotto tra le matrici $\mathbf{L}$ e $\mathbf{U}$, in seguito lo si è sottratto ad $\mathbf{A}$ ottenendo:

$$\mathbf{A_1} - (\mathbf{LU})_\mathbf{1} = \begin{bmatrix} 0.0 & 0.0 \\ 0.0 & 0.0 \end{bmatrix}, \qquad \mathbf{A_2} - (\mathbf{LU})_\mathbf{2} = \begin{bmatrix} 0.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix}.$$

È evidente che il risultato per $\epsilon = 10^{-12}$ è accurato contrariamente al risultato ottenuto per $\epsilon = 10^{-20}$, in cui la verifica della fattorizzazione LU non porta alla matrice di zeri.

(e) Si è calcolato $\kappa(A)$ utilizzando sia la norma 1 che la norma infinito, ottenendo lo stesso risultato a causa della natura simmetrica della matrice $\mathbf{A}$. La formula generale per il numero di condizionamento della matrice $\mathbf{A}$ presa in esame è $\kappa(A) = \frac{4}{1-\epsilon}$.

$$\kappa_1(A) = \kappa_\infty(A) = 4.000000000004\,, \qquad \kappa_1(A) = \kappa_\infty(A) = 4.0\,.$$

Come si può notare, in entrambi i casi il numero di condizionamento non è molto grande, quindi il sistema $\mathbf{Ax} = \mathbf{b}$ non è mal condizionato.

(f) Si sono calcolate a mano le matrici $\mathbf{L}$ e $\mathbf{U}$ della fattorizzazione LU di $\mathbf{A}$ per ciascun valore di ϵ. I calcoli dettagliati sono disponibili nella cartella git, nel percorso `lab_computazionale1\esercizi\pen_and_paper`. Ad ogni modo le matrici ottenute sono già state riportate al punto 2.5.1.

(g) Si sono calcolati i numeri di condizionamento di $\mathbf{L}$ e $\mathbf{U}$, utilizzando sia la norma 1 che la norma infinito, ancora una volta ottenendo lo stesso numero.
Le forumle generali per il numero di condizionamento delle matrici $\mathbf{L}$ e $\mathbf{U}$ sono:

$$\kappa_1(L) = \kappa_\infty(L) = \left(1 + \frac{1}{\epsilon}\right)^2\,, \qquad \kappa_1(U) = \kappa_\infty(U) = \frac{1}{\epsilon^2}\,.$$

In tabella 2.2 sono riassunti i risultati ottenuti.

Table 2.2: Valori dei numeri di condizionamento al variare di ε

	$\varepsilon = 10^{-12}$	$\varepsilon = 10^{-20}$
L	$1.0 \cdot 10^{24}$	$1.0 \cdot 10^{40}$
U	$1.0 \cdot 10^{24}$	$1.0 \cdot 10^{40}$

2.5.2 Conclusioni

L'esercizio ha mostrato come il numero di condizionamento di una matrice sia un indicatore globale della stabilità del problema, ma non dell'algoritmo scelto. Sebbene il numero di condizionamento della matrice $\mathbf{A}$ sia relativamente basso, la risoluzione del sistema richiede il calcolo delle matrici $\mathbf{L}$ e $\mathbf{U}$. Nel caso con $\epsilon = 10^{-20}$, in cui $\kappa(A) = 4.0$ si ha $\kappa(L) = 1.0 \cdot 10^{40}$ e $\kappa(U) = 1.0 \cdot 10^{40}$. Dato che l'algoritmo utilizza le matrici $\mathbf{L}$ e $\mathbf{U}$ per risolvere il sistema è evidente che l'errore numerico aumenta a causa delle stesse.
Se ne deduce che esiste una riformulazione dell'algoritmo che permetta di evitare il calcolo di matrici mal condizionate. Nel caso in esame si può affrontare il problema con l'applicazione di row-pivoting, considerando ε molto vicino a zero.

2.6 Esercizio 2.6.1

(a) Scrivi un programma che prenda in input una matrice $\mathbf{A}$ e un vettore $\mathbf{b}$ e risolva il problema dei minimi quadrati $\text{argmin}\|\mathbf{b} - \mathbf{Ax}\|_2$ utilizzando la decomposizione di Cholesky.

(b) Per verificare il funzionamento del tuo codice, calcola la soluzione ai minimi quadrati quando

$$\mathbf{A} = \begin{bmatrix} 2 & -1 \\ 0 & 1 \\ -2 & 2 \end{bmatrix}, \qquad \mathbf{b} = \begin{bmatrix} 1 \\ -5 \\ 6 \end{bmatrix}$$

2.6.1 Soluzione

Si è implementato l'algoritmo per il metodo dei minimi quadrati e lo si è testato sulle matrici proposte, ottenendo:

$$\mathbf{x} = \begin{bmatrix} -2.0 & -1.0 \end{bmatrix}.$$

2.7 Esercizio 2.6.2

Keplero ha scoperto che il periodo orbitale τ di un pianeta dipende dalla sua distanza media R dal Sole secondo la legge $\tau = cR^{\alpha}$, dove α è un semplice numero razionale.

(a) Esegui un fit lineare ai minimi quadrati utilizzando la tabella 7.1 per determinare il valore più probabile e semplice di α.

(b) Realizza un grafico dei dati e del risultato del fit.

2.7.1 Soluzione

(a) Il metodo dei minimi quadrati sviluppato durante il corso è adatto a risolvere problemi lineari. Non è il caso del problema proposto, dato che α si trova ad esponente. Per risolvere la questione si è scelto di linearizzare il problema tramite il logaritmo naturale. In formule:

$$\ln(\tau) = \ln(c) + \alpha \ln(R). \tag{2.1}$$

Da qui in poi si è implementato il metodo dei minimi quadrati, applicando la scomposizione di Cholesky alla matrice $n \times n$ ottenuta dal prodotto della matrice di design e la sua trasposta. Per chiarezza si riporta la matrice di design trasposta:

$$\mathbf{A}^T = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \ln(57.59) & \ln(108.11) & \cdots & \ln(4499.9) \end{bmatrix}.$$

La matrice di regressione è stata costruita in modo che la prima colonna fosse composta da 1 per poter calcolare il termine noto della retta e la seconda colonna fosse composta dai logaritmi naturali delle distanze dei pianeti dal Sole.
La legge di Keplero prevista è la seguente:

$$\tau = cR^{3/2}, \qquad \text{con } \alpha = \frac{3}{2}, \quad c = \frac{2\pi}{\sqrt{G(M+m)}} \tag{2.2}$$

dove G è la costante di gravitazione universale, M è la massa del Sole, m è la massa del pianeta.
La regressione lineare che si vuole effetturare considera c costante nonostante dipenda dalla massa del pianeta. Consideriamo Giove, il pianeta più massivo di quelli presi in esame ($m \approx 1.90 \cdot 10^{27}\,\text{kg}$). Si ha:

$$\begin{aligned} c_1 &= \frac{2\pi}{\sqrt{G(M+m)}} = 5.47 \cdot 10^{-10} \frac{\text{sec}}{m^{3/2}} = 0.203 \frac{days}{Mkm^{3/2}} \\ c_2 &= \frac{2\pi}{\sqrt{GM}} = 5.47 \cdot 10^{-10} \frac{\text{sec}}{m^{3/2}} = 0.203 \frac{days}{Mkm^{3/2}} \\ \frac{c_1}{c_2} &\approx 1.0 \end{aligned}$$

Possiamo quindi affermare che la costante c è indipendente dalla massa del pianeta, entro la precisione necessaria per il nostro scopo.
Si sono quindi stimati il valore di α e della costante c, tramite la regressione, ottenendo:

$$c = 0.206 \frac{days}{Mkm^{3/2}} \qquad \alpha = 1.499$$

I risultati sono in accordo con quanto atteso.

(b) Si riporta in figura 2.1 il grafico dei dati e relativo fit.

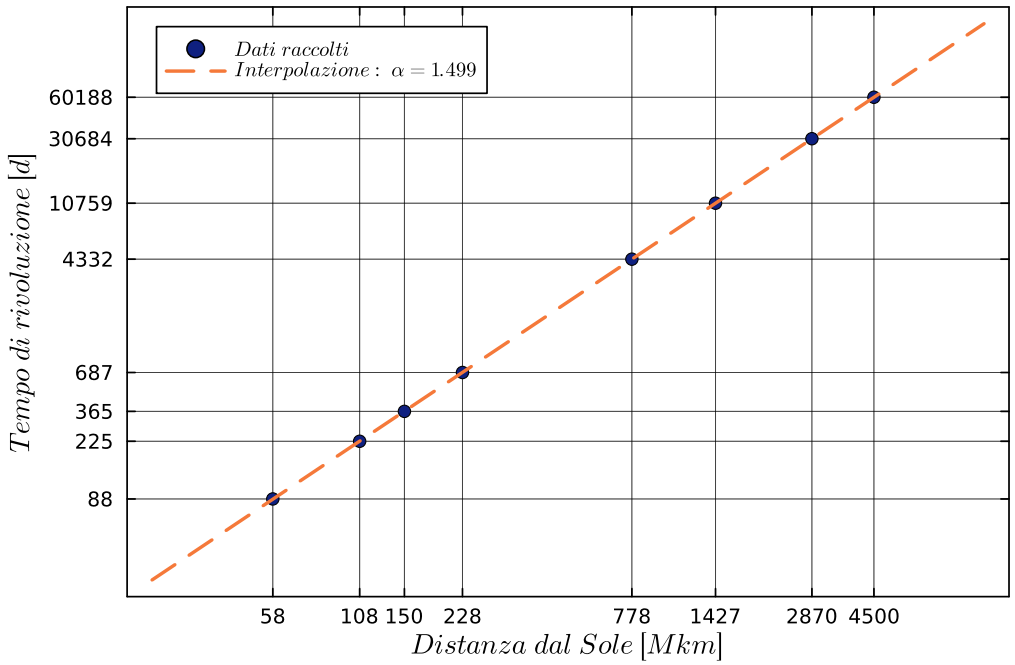


Figure 2.1: Fit lineare ai minimi quadrati dei dati dei pianeti del Sistema Solare.

2.8 Esercizio 2.6.3

In questo esercizio si vuole trovare un'approssimazione della funzione periodica $g(t) = e^{\sin(t-1)}$ su un periodo, $0 < t \leq 2\pi$. Come dati, si definiscono

$$t_i = \frac{2\pi i}{60}, \quad y_i = g(t_i), \quad i = 1, \ldots, 60.$$

(a) Trova i coefficienti del fit ai minimi quadrati

$$y(t) \approx c_1 + c_2 t + \cdots + c_7 t^6. \tag{2.3}$$

Sovrapponi un grafico dei valori dei dati come punti con una curva che mostra il fit.

(b) Trova i coefficienti del fit ai minimi quadrati

$$y \approx d_1 + d_2 \cos(t) + d_3 \sin(t) + d_4 \cos(2t) + d_5 \sin(2t). \tag{2.4}$$

A differenza del punto (a), questa funzione di fitting è essa stessa periodica. Sovrapponi un grafico dei valori dei dati come punti con una curva che mostra il fit.

2.8.1 Soluzione

Dopo avere generato i dati richiesti, si è eseguito il metodo dei minimi quadrati con le seguenti matrici di regressione:

$$\mathbf{D_{pol}} = \begin{bmatrix} 1 & t_1 & \dots & t_1^6 \\ 1 & t_2 & \dots & t_2^6 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & t_{60} & \dots & t_{60}^6 \end{bmatrix}, \qquad \mathbf{D_{Four}} = \begin{bmatrix} 1 & \cos(t_1) & \sin(t_1) & \cos(2t_1) & \sin(2t_1) \\ 1 & \cos(t_2) & \sin(t_2) & \cos(2t_2) & \sin(2t_2) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \cos(t_{60}) & \sin(t_{60}) & \cos(2t_{60}) & \sin(2t_{60}) \end{bmatrix},$$

ottenendo i coefficienti delle formule 2.3 e 2.4:

- Coefficienti del polinomio: $[0.712; -1.351; 2.225; -0.538; -0.074; 0.033; -0.003]$
- Coefficienti della funzione di Fourier: $[1.266; -0.951; 0.611; 0.113; -0.247]$

Si riportano i grafici dei dati e dei fit in figura 2.2.

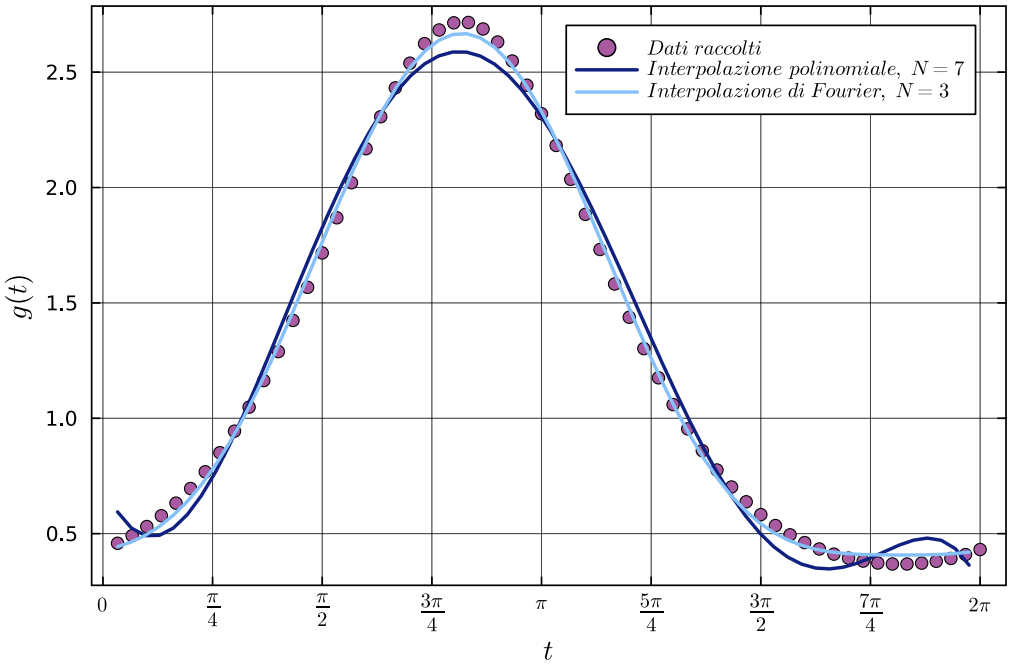


Figure 2.2: Fit polinomiale e di Fourier ai minimi quadrati della funzione $g(t) = e^{\sin(t-1)}$.

2.8.2 Conclusioni

In figura 2.2 si osserva che il fit meglio riuscito è quello di Fourier, nonostante richieda meno parametri. Il motivo è che la funzione $g(t)$ è periodica, di conseguenza il fit con modello 2.4 riesce a catturare meglio la periodicità della funzione rispetto al fit eseguito con 2.3.
In un contesto sperimentale la funzione $g(t)$ non è nota a priori, anzi lo scopo dell'esperimento è solitamente quello di risalire a tale funzione. Il fatto che il fit secondo il modello di Fourier riesca meglio è indice della natura periodica del fenomeno osservato.
Si potrebbe supporre che l'errore sulle code nel caso di interpolazione polinomiale sia di natura numerica, dato che il polinomio calcolato in maniera naive come $\sum_{k=0}^{n-1} c_k x^k$ non è garante di stabilità. Tale supposizione è falsa perchè si è calcolato lo stesso polinomio con il metodo di Horner e il risultato è lo stesso. Si rimanda a `\lab_computazionale\esercizi\2_6_3.ipynb` per i dettagli.

3 Radici di equazioni non lineari

3.1 Esercizi 3.2.1, 3.2.2

1. Scrivi un programma che implementi il metodo di bisezione e testalo con una funzione semplice, ad esempio $f(x) = x$.

2. (a) Usa il metodo di bisezione per trovare le soluzioni di $x^3 - 7x^2 + 14x - 6 = 0$ su $[0, 1]$.

   (b) Studia la convergenza del metodo di bisezione rispetto alla radice trovata in (a). (Puoi prendere come approssimazione per la radice $r = m_{n_{\max}}$, dove $n_{\max}$ è l'indice corrispondente all'ultima iterazione di bisezione.)
   Qual è il suo ordine di convergenza q? Puoi stimare la costante d'errore asintotica C?
   Suggerimento: Studia $d_n = -\log_{10} |x_n - r|$ come funzione di n.

3.1.1 Soluzione

(1) Si è implementato il metodo di bisezione testandolo con la funzione $f(x) = x$ per due intervalli: $[-1.0, 1.0]$ e $[-5.0, 1.0]$. Ci aspettiamo $x = 0$ come soluzione, nel caso di intervallo simmetrico il numero di iterazioni è pari ad 1, per intervallo asimmetrico il numero di iterazioni dovrebbe aumentare.
Il metodo si interrompe quando la distanza tra gli estremi risulta minore di $\max(|a|, |b|) \cdot 10^{-16}$, dove a e b sono gli estremi dell'intervallo. Il metodo ha prodotto i seguenti risultati:

- Intervallo $[-1.0, 1.0]$: $x = 0.0$ in 1 iterazione.
- Intervallo $[-5.0, 1.0]$: $x = 1.1 \cdot 10^{-162}$ in 539 iterazioni.

(2a) Si è implementato il metodo di bisezione per la funzione $f(x) = x^3 - 7x^2 + 14x - 6$.
Il metodo ha restituito come valore della radice $r = 0.586$ con $f(r) = 0.0$, in 50 iterazioni. In figura 3.1 si riporta il grafico della funzione e delle radici successive trovate.

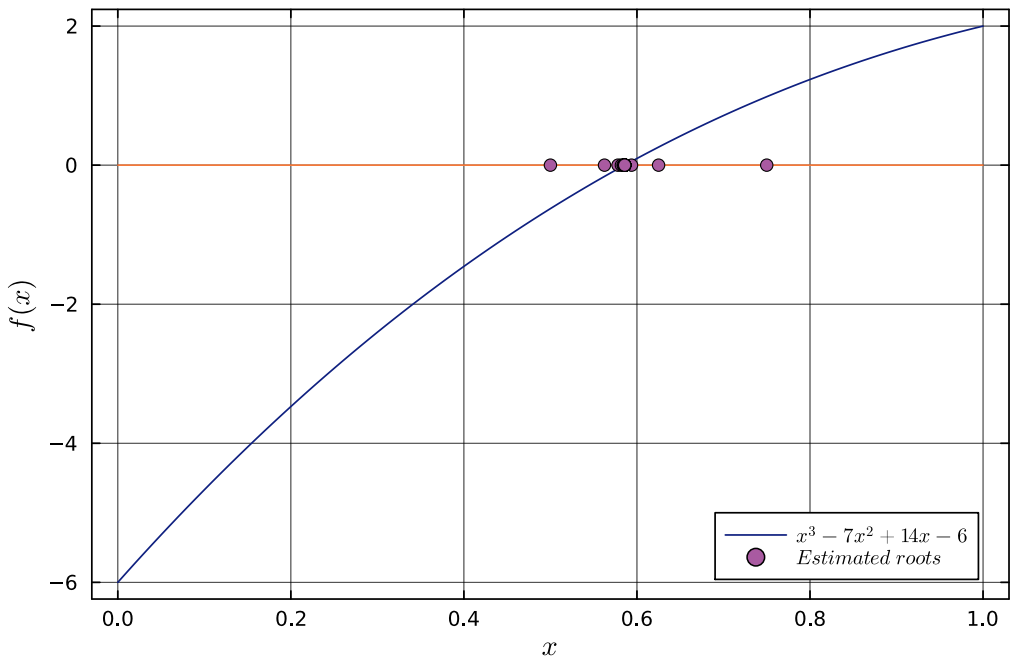


Figure 3.1: Funzione $f(x) = x^3 - 7x^2 + 14x - 6$ e radice ottenuta con metodo di bisezione.

(2b) Per studiare la convergenza del metodo di bisezione bisogna fare alcune considerazioni preliminari.
È noto che per il metodo di bisezione vale la relazione $|m_n - r| < 2^{-(n+1)}(b_0 - a_0)$, dove m_n è la stima della radice al passo n, r è la radice cercata, a_0 e b_0 sono gli estremi dell'intervallo iniziale.
Considerando l'espressione analoga per $|m_{n+1} - r|$ si ricava $\frac{|m_{n+1}-r|}{|m_n-r|} < \frac{1}{2}$. Al limite:

$$\lim_{n\to\infty} \frac{|m_{n+1} - r|}{|m_n - r|} = \frac{1}{2}, \tag{3.1}$$

da cui si ottiene che q, ordine di convergenza, è 1, mentre C, la costante di errore asintotico, vale $\frac{1}{2}$.
Per verificare i valori teorici di q e C è necessario tenere a mente le seguenti relazioni. La prima si ottiene applicando il logaritmo decimale alla 3.1:

$$d_{n+1} \approx q\, d_n - \log_{10}|C| \quad \Rightarrow \quad \frac{d_{n+1}}{d_n} \approx q - \frac{\log_{10}|C|}{d_n} \xrightarrow{n\to\infty} q \tag{3.2}$$

dove $d_n = -\log_{10}|m_n - r|$.
La seconda si ottiene iterando la relazione $|x_{k+1} - r| = C|x_k - r|$:

$$|x_{k+1} - r| = C^{k+1}|x_0 - r| \tag{3.3}$$

Stima di q e C: Per stimare q si è calcolato il rapporto tra i valori di d_n e d_{n+1} e lo si è messo in grafico in funzione di n, ottenendo la figura 3.2.

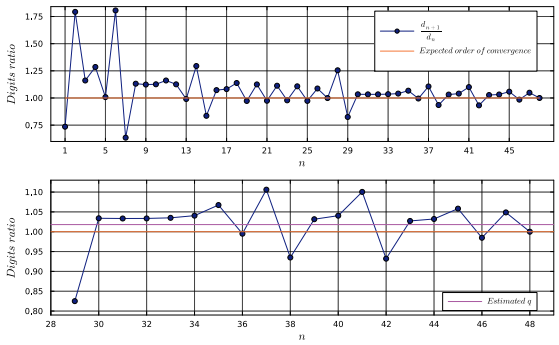


Figure 3.2: Stima di q

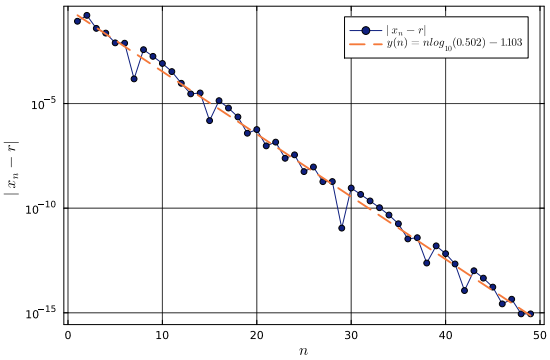


Figure 3.3: Stima di C

Il rapporto tende a 1, come atteso. Si vuole ora quantificare tale valore.
La relazione 3.2 vale per $n \to \infty$, quindi la stima di q dovrebbe essere calcolata per l'n più grande possibile. Si è deciso però di stimare il valore di q tramite l'interpolazione con funzione costante perchè il suo valore fluttua attorno a 1, ed è possibile che per n maggiori di quelli disponibili il valore cercato continui a fluttuare attorno a 1. Per l'interpolazione si sono considerati gli ultimi 20 punti della serie, riportati nell'ingrandimento di 3.2. Il fit su tali punti restituisce un valore di $q = 1.02$.

Per stimare C si è eseguito il fit con metodo dei minimi quadrati sul modello linearizzato di 3.3:

$$log|x_{k+1} - r| = (k+1)log(C) + log|x_0 - r| \quad \Rightarrow \quad y(x) = xlog(C) + A, \tag{3.4}$$

avendo considerato r come l'ultimo valore di m_n calcolato. Si riporta il grafico del fit in figura 3.3. Il fit ha restituito i seguenti valori:

$$C = 0.5, \qquad A = -0.1.$$

Il valore di C è in accordo con quanto atteso, A non è significativo.

Commento: È possibile stimare i valori di q e C in modo diverso, con un solo fit. Interpolando la relazione 3.2 con un'iperbole e ignorando il limite si riescono a ottenere i valori di q e C in un colpo solo. I risultati ottenuti hanno una precisione minore di quelli presentati, perchè tale metodo non tiene conto del carattere asintotico delle relazioni. Per questa ragione non sono riportati. Si rimanda alla cartella git `lab\computazionale1\esercizi\3_2_2.ipynb` per i dettagli.

3.2 Esercizi 3.3.1, 3.3.2

1. Scrivi un programma che implementi il metodo di Newton.
2. Per ciascuna delle seguenti funzioni, esegui i seguenti passi:
   (a) Riscrivi l'equazione nella forma standard per la ricerca degli zeri, $f(x) = 0$, e calcola $f'(x)$.
   (b) Traccia il grafico di f nell'intervallo indicato e determina quante radici sono presenti nell'intervallo.
   (c) Usa il metodo di Newton per trovare ciascuna radice.
   (d) Studia l'errore nella sequenza di Newton per determinare numericamente se la convergenza è approssimativamente quadratica per la radice trovata.

   Le funzioni da considerare sono:

   - $x^2 = e^{-x}$, su $[-2, 2]$
   - $2x = \tan x$, su $[-0.2, 1.4]$
   - $e^{x+1} = 2 + x$, su $[-2, 2]$

3.2.1 Soluzione

(1) Si è implementato il metodo di Newton, testandolo su $f(x) = e^x - 1$ con valore iniziale $x_0 = 10$. Il metodo ha restituito il valore $-6.68 \cdot 10^{-17}$ in 16 iterazioni. Valore atteso: $x = 0$.

(2a) Le funzioni in forma di ricerca degli zeri sono:

- $f_1(x) = x^2 - e^{-x}$, con $f_1'(x) = 2x + e^{-x}$, su $[-2, 2]$.
- $f_2(x) = 2x - \tan x$, con $f_2'(x) = 2 - \tan^2 x$, su $[-0.2, 1.4]$.
- $f_3(x) = e^{x+1} - 2 - x$, con $f_3'(x) = e^{x+1} - 1$, su $[-2, 2]$.

(2b) I grafici delle funzioni sono riportati in figura 3.4, da cui si evince che:

- f_1 e f_3 hanno una radice ciascuna nell'intervallo $[-2, 2]$.
- f_2 ha due radici nell'intervallo $[-0.2, 1.4]$.

(2c) L'algoritmo di Newton è stato applicato alle tre funzioni, con i seguenti risultati:

- f_1: radice $x = 0.703$, in 8 iterazioni. Punto di partenza $x_0 = -2$.
- f_2: radici $x_1 = 0.000$, $x_2 = 1.166$, entrambe in 7 iterazioni. Punti di partenza $x_{0,1} = 0.65$, $x_{0,2} = 1.4$.
- f_3: radice $x = -1.000$ in 29 iterazioni. Punto di partenza $x_0 = 2$.

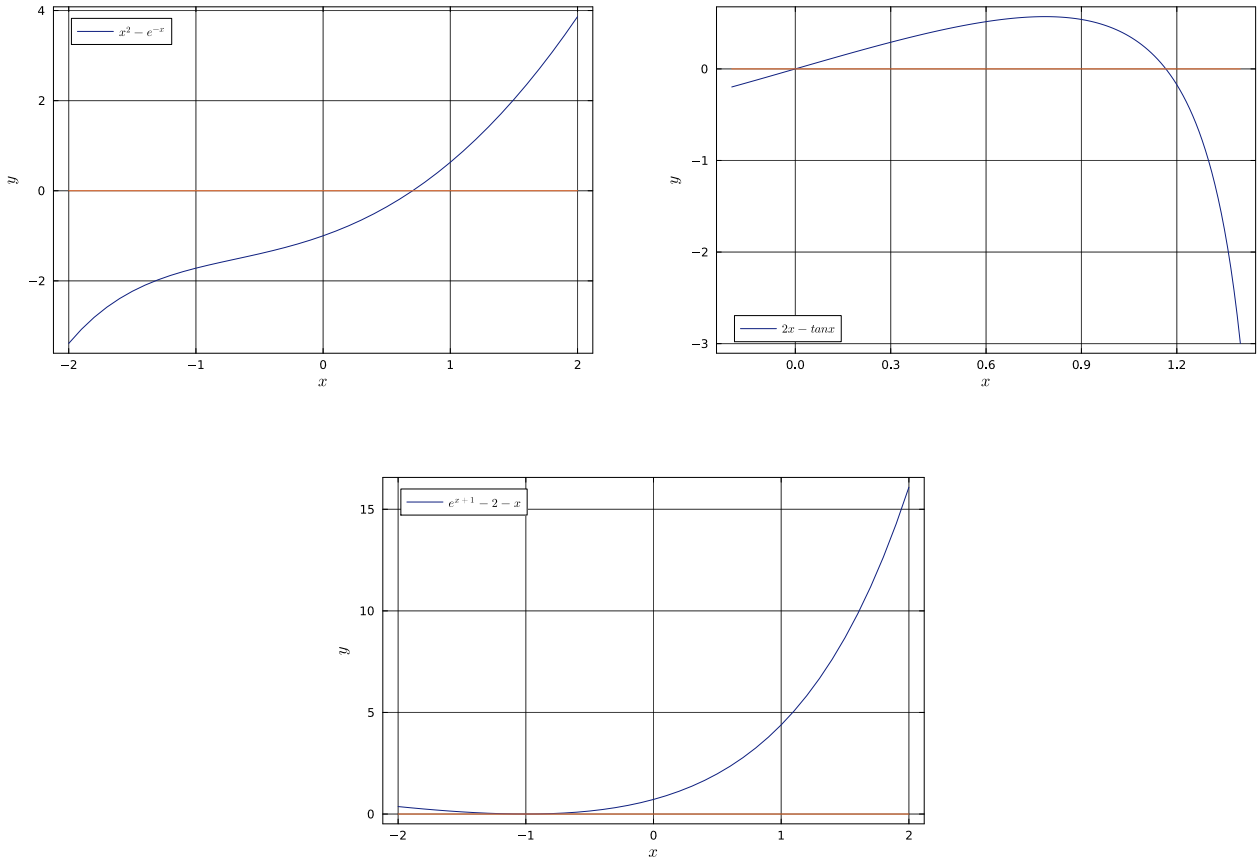


Figure 3.4: Grafici delle funzioni f_1, f_2 e f_3.

(2d) - Prima funzione Scopo dello studio dell'errore è verificare se i valori di q e C sono in accordo con quanto atteso. In particolare ci si aspetta che il metodo di Newton per questo particolare zero abbia $q = 2$ e $C = \frac{1}{2}\left|\frac{f''(r)}{f'(r)}\right| = 0.396$. Per determinare i valori di q, qui e nelle altre funzioni, si fa riferimento alla formula 3.2; per determinare C, qui e nelle altre funzioni, si fa riferimento alla relazione $\lim_{n\to\infty} \frac{|x_{n+1}-r|}{|x_n-r|^q} = \lim_{n\to\infty} \frac{|\epsilon_{n+1}|}{|\epsilon_n|^q} = C$
Si riportano i due grafici in figura 3.5.

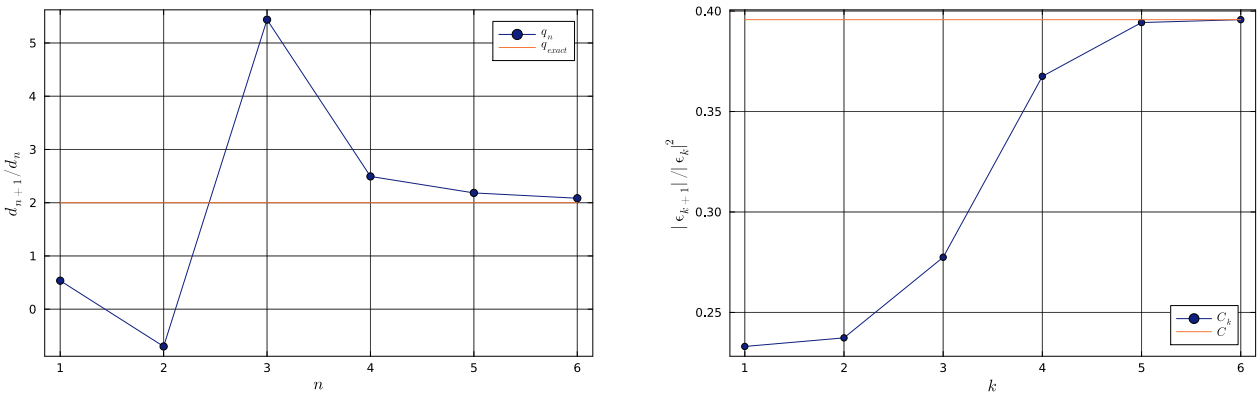


Figure 3.5: Studio del valore di q e C per f_1.

Per le stime di q e C si è scelto di considerare l'ultimo punto della successione corrispondente

a causa del numero limitato di iterazioni, qui e nelle altre funzioni. Si ottiene:

$$q = 2.084, \qquad C = 0.396.$$

Si osserva che i valori ottenuti sono in accordo con quanto atteso, il metodo di Newton converge in modo quadratico per la radice trovata.

(2d) - Seconda funzione

Prima radice, $x = 0.0$: È importante notare che la funzione f_2 in zero ha un andamento lineare perchè la sua derivata seconda è nulla. I valori di q e C non sono determinati dalle formule precedenti, perchè ottenute a partire da espansione di Taylor al secondo ordine. Per ovviare al problema si è scelto di ricavare le formule già utilizzate espandendo a ordine 3. I calcoli sono riportati nella cartella git `lab_computazionale1\esercizi\pen_and_paper`.
Di seguito sono riportati i valori attesi per q e C con le dovute correzioni:

$$q = 3, \qquad C = \frac{f'''(r)}{3f'(r)} = 0.667.$$

In figura 3.6 si riportano i grafici delle successioni q_n e C_n per la prima radice di f_2.

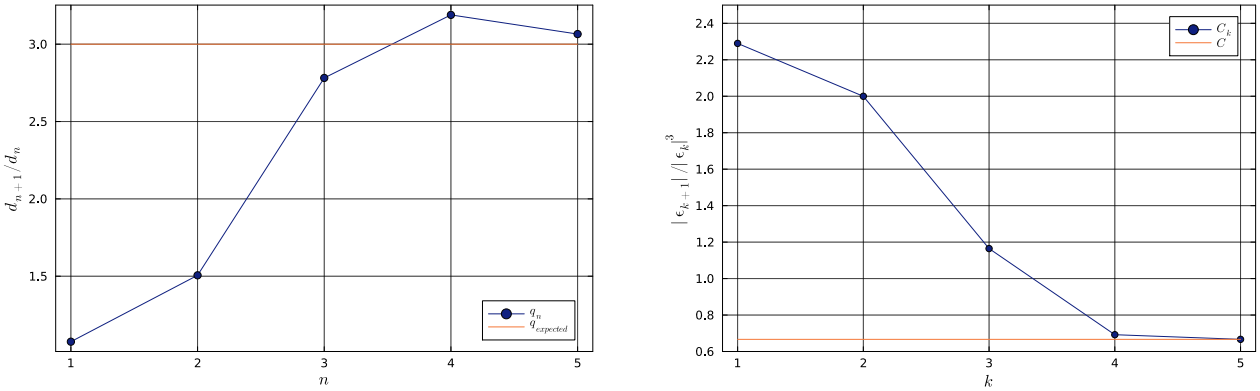


Figure 3.6: Studio del valore di q e C per f_2, prima radice.

Si ottengono i seguenti valori:

$$q = 3.065, \qquad C = 0.667.$$

Ci si può chiedere cosa succede se, non conoscendo l'andamento della funzione, si utilizzano le formule precedenti. In particolare cosa succede se si fissa $q = 2$ e si stima C come l'ultimo elemento della successione $C_n = \frac{|x_{n+1}-r|}{|x_n-r|^2}$. L'aspettativa è che C risulti nullo, dato che la derivata seconda è nulla. Il grafico corrispondente può essere trovato nella cartella git `lab_computazionale1\esercizi\3_3_2.ipynb`, conferma le aspettative.

Seconda radice, $x = 1.166$: la seconda radice di f_2 non presenta problemi, così le formule sono quelle viste inizialmente. I valori attesi sono $q = 2$ e $C = 3.383$. Si riportano i grafici delle successioni q_n e C_n in figura 3.7. Si ottengono i seguenti valori:

$$q = 1.918, \qquad C = \frac{f''(r)}{2f'(r)} = 3.382.$$

Come atteso il metodo di Newton converge in modo quadratico per la radice trovata.

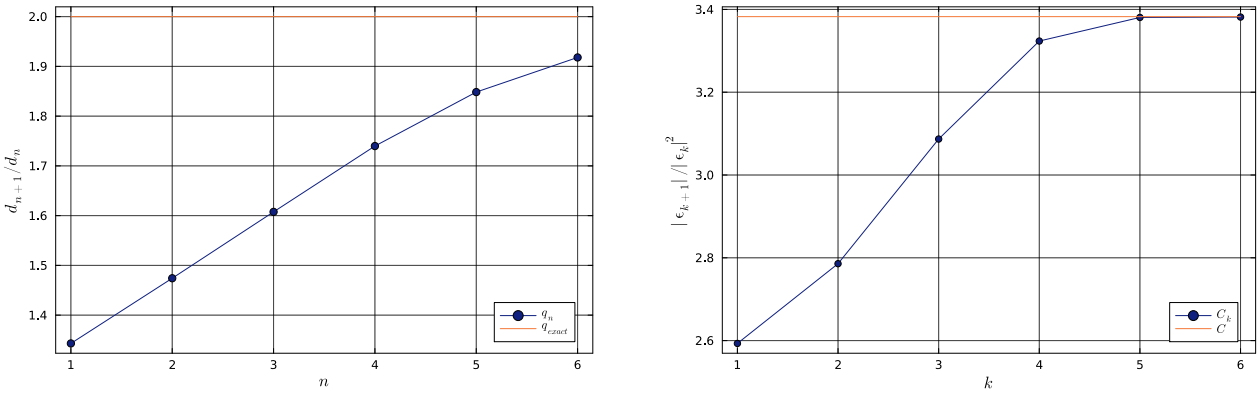


Figure 3.7: Studio del valore di q e C per f_2, seconda radice.

(2d) - Terza funzione Osservando il grafico della funzione f_3 si nota che la radice è un punto di minimo. Il numero di iterazioni molto elevato rispetto agli altri casi conferma questa supposizione.
Come è noto dalla teoria, il metodo di Newton in caso di radici multiple converge linearmente, perciò vale $\varepsilon_{n+1} = C\varepsilon_n$. La costante d'errore asintotica diviene $(1 - \frac{1}{m})$.
Il primo passo è verificare il valore di q. Si riporta in figura 3.8 il grafico della successione q_n che restituisce $q = 1.067$.

Si riportano in figura 3.8 i grafici della successione C_n, una volta senza considerare la molteplicità della radice, e una volta considerando la molteplicità della radice.

Non avendo considerato la molteplictà della radice il grafico a sinistra mostra che i valori delle stime successive di C si discostano molto da $C = \frac{f''(r)}{2f'(r)}$. Nel grafico a destra si osserva che i valori successivi di C_n si stabilizzano attorno al valore $C = 0.5$, per poi calare nuovamente a causa di errori di cancellazione numerica.
Scegliendo $C = C_{15}$ come stima del plateau si ottengono i seguenti valori:

$$C = 1 - \frac{1}{m} = 0.500, \qquad m = 2.000,$$

in accordo con quanto atteso.

Commento: Nella cartella git `lab_computazionale1\esercizi\3_3_2.ipynb` si trovano i dettagli su metodi alternativi con cui stimare la radice per la funzione f_3. In particolare, non conoscendone la molteplicità, si è provato a cambiare il problema $f(x) = 0$ in $g(x) = f(x)/f'(x) = 0$. Dalla teoria è noto che il problema riformulato in questa maniera ammette la stessa radice, con il vantaggio di essere radice semplice. Inoltre la convergenza per questo metodo è quadratica. Alla fine del notebook relativo sono mostrati i risultati ottenuti.

3.3 Esercizio 3.3.3

Traccia la funzione $f(x) = x^{-2} - \sin x$ nell'intervallo $x \in [0.5, 10]$. Per ciascun valore iniziale $x_1 = 1$, $x_1 = 2$, ..., $x_1 = 7$, applica il metodo di Newton a f e realizza una tabella che mostri x_1 e la radice trovata dal metodo. In quale caso l'iterazione converge a una radice diversa da quella più vicina al valore iniziale? Usa il grafico per spiegare perché ciò è accaduto.

3.3.1 Soluzione

Si riporta in figura 3.9 il grafico della funzione $f(x) = x^{-2} - \sin x$ nell'intervallo $[0.5, 10]$, con i punti di partenza del metodo di Newton.

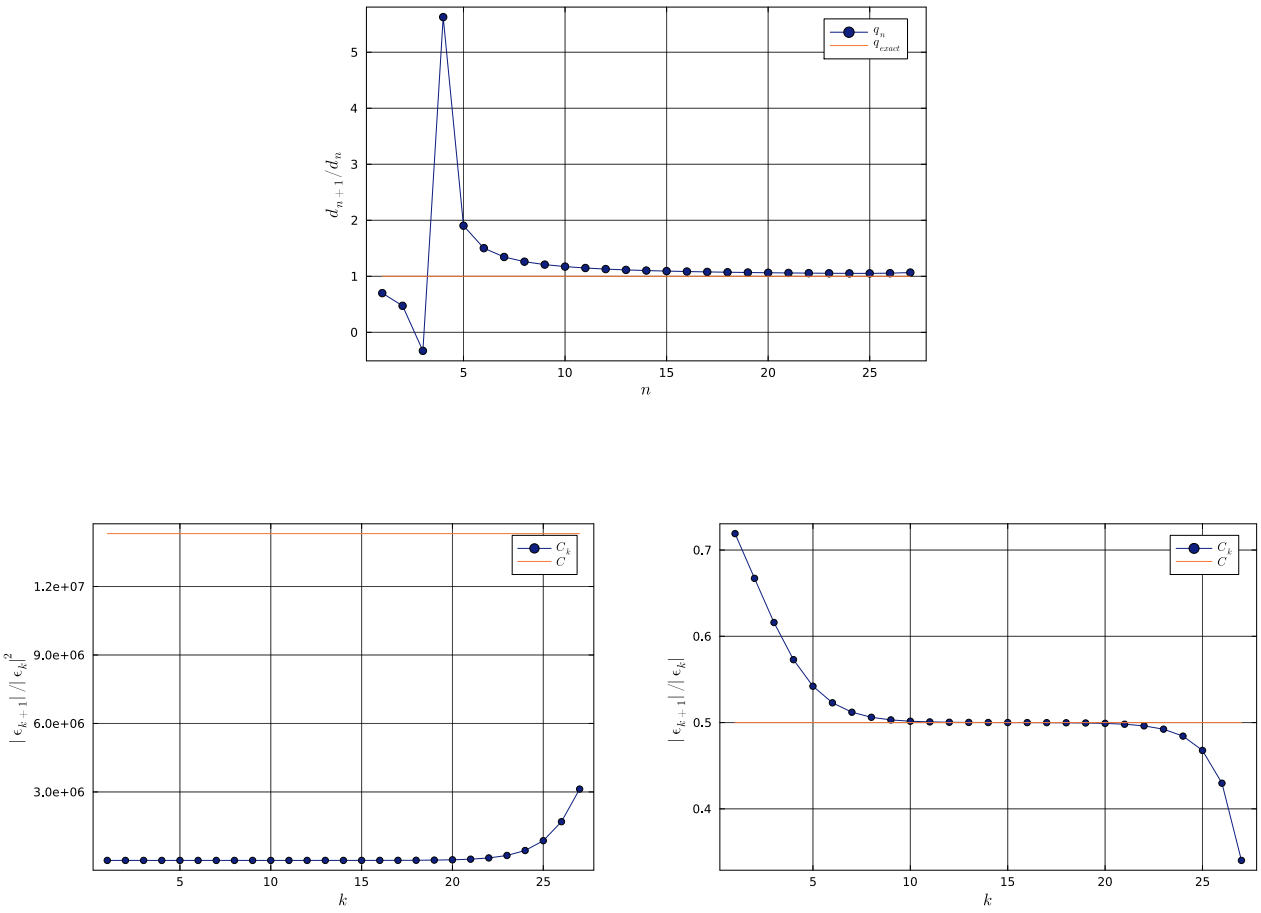


Figure 3.8: Studio del valore di q (sopra) e C (sotto) per f_3.

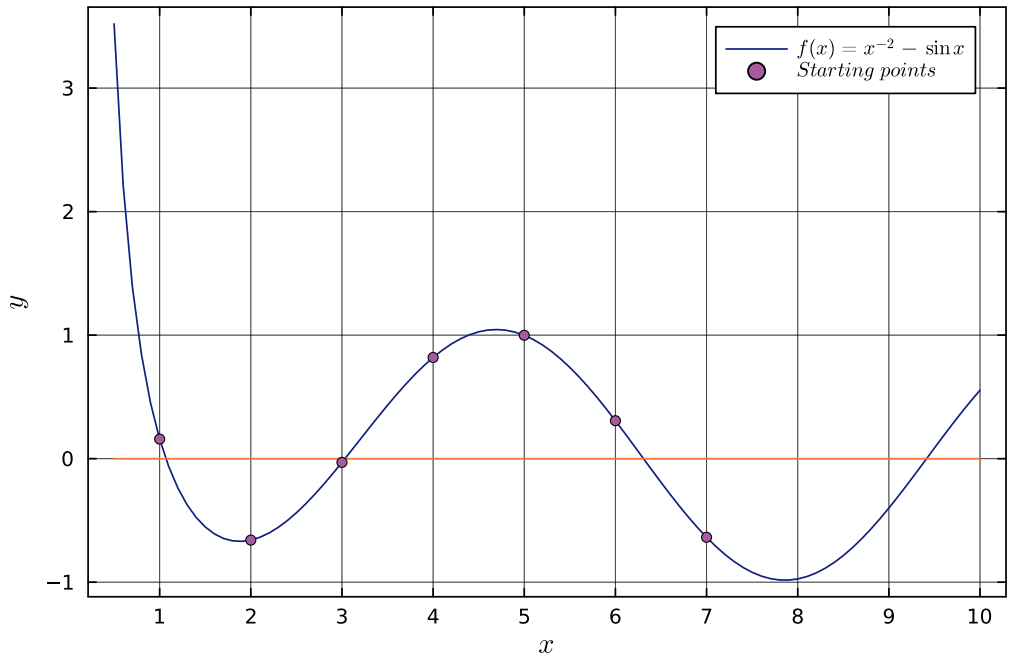


Figure 3.9: Grafico della funzione $f(x) = x^{-2} - \sin x$ nell'intervallo $[0.5, 10]$.

Applicando il metodo di Newton con i valori iniziali, non sempre si ottengono le radici più vicine al valore di partenza. Si riportano i risultati in tabella 3.1.

Table 3.1: Valori relativi ai punti $x_1, \ldots, x_7$

	$x_1 = 1$	$x_2 = 2$	$x_3 = 3$	$x_4 = 4$	$x_5 = 5$	$x_6 = 6$	$x_7 = 7$
x_0	1.0	2.0	3.0	4.0	5.0	6.0	7.0
x_{root}	1.1	6.3	3.0	3.0	9.4	6.3	6.3
$f'(x_0)$	-2.5	0.2	0.9	0.6	-0.3	-0.9	-0.8

Le due colonne evidenziate in arancione sono i punti iniziali che non convergono alla radice più vicina. Osservando il grafico della funzione è facile osservare che i punti iniziali x_2 e x_5 sono vicini a un minimo e a un massimo locali, rispettivamente. In effetti la derivata prima in questi punti è vicina a zero, come riportato in tabella. Non è esattamente zero: ciò significa che il metodo di Newton converge, ma non necessariamente alla radice più vicina. La tangente infatti interseca l'asse x lontano dalla radice desiderata. Se ne è presente un'altra nell'intorno del nuovo punto il metodo di Newton non riesce a distinguerle e converge all'una o all'altra a seconda dell'inclinazione della retta tangente.

3.4 Esercizio 3.3.4

Le proprietà più facilmente osservabili dell'orbita di un corpo celeste attorno al Sole sono il periodo τ e l'eccentricità ellittica ϵ. (Un cerchio ha $\epsilon = 0$.) Da questi parametri è possibile determinare, in ogni istante t, la *vera anomalia* $\theta(t)$. Questa è l'angolo tra la direzione del perielio (il punto dell'orbita più vicino al Sole) e la posizione attuale del corpo, visto dal fuoco principale dell'ellisse (dove si trova il Sole). Questo si ottiene tramite

$$\tan\frac{\theta}{2} = \sqrt{\frac{1+\epsilon}{1-\epsilon}}\,\tan\frac{\psi}{2} \tag{3.5}$$

dove l'*anomalia eccentrica* $\psi(t)$ soddisfa l'equazione di Keplero:

$$\psi - \epsilon\sin\psi - \frac{2\pi t}{\tau} = 0\,. \tag{3.6}$$

L'equazione 3.6 deve essere risolta numericamente per trovare $\psi(t)$, e poi la 3.5 può essere risolta analiticamente per ottenere $\theta(t)$. L'asteroide Eros ha $\tau = 1.7610$ anni e $\epsilon = 0.2230$. Utilizzando il metodo di Newton per la 3.6, realizza un grafico di $\theta(t)$ per 100 valori di t compresi tra 0 e τ, cioè per un'orbita completa.
(Nota: usa $\texttt{mod}(\theta, 2\pi)$ per riportare l'angolo tra 0 e 2π se vuoi che il risultato sia una funzione continua.)

3.5 Soluzione

Per prima cosa si è risolta l'equazione 3.6 con il metodo di Newton. Per capire meglio il procedimento si riporta l'equazione 3.6 nel grafico sinistro di 3.10, dove si osserva che la funzione ha un solo zero nell'intervallo $[0, 2\pi]$. Il metodo di Newton è stato applicato al variare del tempo t, cioè è stato applicato alla stessa funzione ma traslata in verticale, come si può vedere nella figura.

Ottenuti i valori di ψ in corrispondenza dei 100 valori di t compresi tra 0 e τ, si è calcolata la vera anomalia θ tramite la relazione 3.5. Si riporta il grafico di $\theta(t)$ in figura sinistra di 3.10.

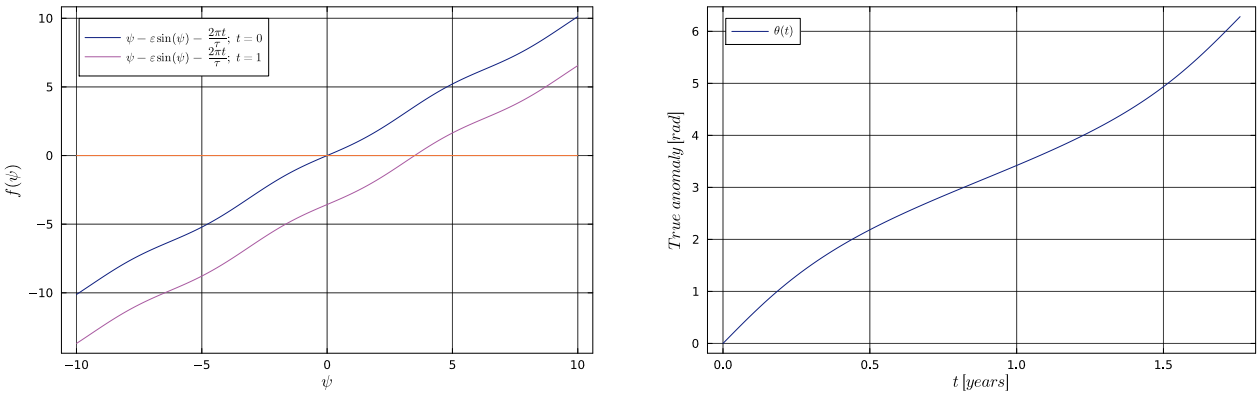


Figure 3.10: Grafico dell'equazione 3.6 (sinistra) e $\theta(t)$ (destra).

3.6 Esercizi 3.4.1, 3.4.2

1. Scrivi un programma che implementi il metodo della secante.
2. Per ciascuna delle seguenti funzioni, esegui i seguenti passi:
   (a) Riscrivi l'equazione nella forma standard per la ricerca degli zeri, $f(x) = 0$.
   (b) Traccia il grafico di f nell'intervallo indicato e determina quante radici sono presenti nell'intervallo.
   (c) Per ciascuna radice determina un intervallo che la racchiuda. Poi usa il metodo della secante, partendo dagli estremi dell'intervallo, per trovare ciascuna radice.
   (d) Per una delle radici, usa gli errori nella sequenza della secante per determinare numericamente se la convergenza è apparentemente compresa tra lineare e quadratica.
3. Le funzioni da considerare sono:
   (a) $x^2 = e^{-x}$, su $[-2, 2]$
   (b) $2x = \tan x$, su $[-0.2, 1.4]$
   (c) $e^{x+1} = 2 + x$, su $[-2, 2]$

3.6.1 Soluzione

(1) Si è implementato il metodo della secante e lo si è testato con la funzione $x^2 - 1$, ottenedo come radice $x = 1.0$ in 10 iterazioni.

(2a)-(2b) Per questi esercizi si rimanda alle sezioni 3.2.1 e 3.2.1.

(2c) L'algoritmo della secante è stato applicato alle tre funzioni, con i seguenti risultati:

- f_1: radice $x = 0.703$, in 9 iterazioni. Intervallo iniziale $[1, 2]$.
- f_2: radici $x_1 = 0.000$, $x_2 = 1.166$, in 6 e 11 iterazioni, rispettivamente. Intervalli iniziali: $[-0.1, 0.2]$ e $[0.9, 1.3]$.
- f_3: radice $x = -1.000$ in 42 iterazioni. Intervallo iniziale $[-2, 0]$.

In generale si può notare che il numero di iterazioni nel metodo della secante è maggiore rispetto a quello del metodo di Newton, nonostante gli intervalli iniziali nel caso della secante siano stati scelti in modo da essere molto vicini alla radice. Ciò è dovuto al fatto che il metodo della secante converge con ordine del rapporto aureo, mentre il metodo di Newton converge con ordine quadratico per radici semplici.
Il vantaggio del metodo della secante rispetto al metodo di Newton è che, seppure converga meno rapidamente, richiede la metà delle valutazioni di funzioni, perchè al passo successivo valuta la funzione nel nuovo punto e riutilizza la valutazione precedente.

(2d) Si è scelto di studiare l'errore della secante per la radice di f_1. Si riporta in figura 3.11 il grafico della successione $d_k = |x_k - x_{k-1}|$. Come stima di q si è scelto di considerare l'ultimo punto della successione, e si ottiene $q = 1.692$, sufficientemente in accordo con $q_{atteso} = 1.618$.

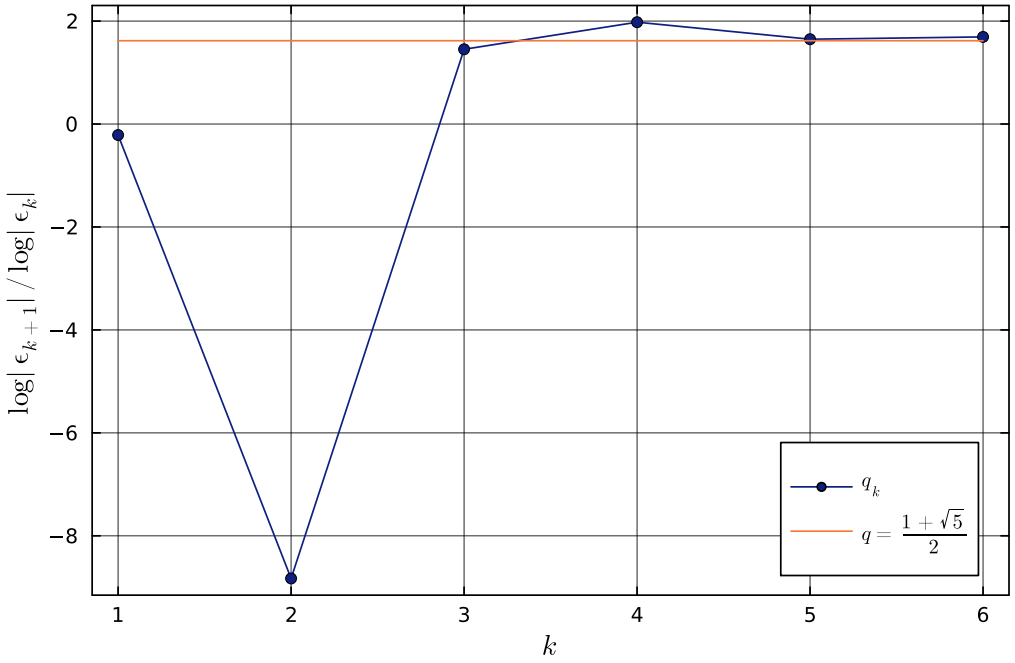


Figure 3.11: Studio del valore di q per la radice di f_1.

3.7 Esercizio 3.4.3

Utilizza un grafico per localizzare approssimativamente tutte le radici di $f(x) = x^{-2} - \sin(x)$ nell'intervallo $[0.5, 10]$. Poi trova una coppia di punti iniziali per ciascuna radice tale che il metodo della secante converga a quella radice.

3.7.1 Soluzione

Si riporta in figura 3.12 il grafico della funzione $f(x) = x^{-2} - \sin(x)$ nell'intervallo $[0.5, 10]$, da cui si evince che la funzione ha 4 radici nell'intervallo. In figura sono riportate anche le radici stimate.

Si riportano in tabella 3.2 i valori delle radici stimate, i punti iniziali scelti e il numero di iterazioni necessarie per la convergenza. Le radici trovate sono in accordo con quelle stimate nell'esercizio 3.3 con metodo di Newton.

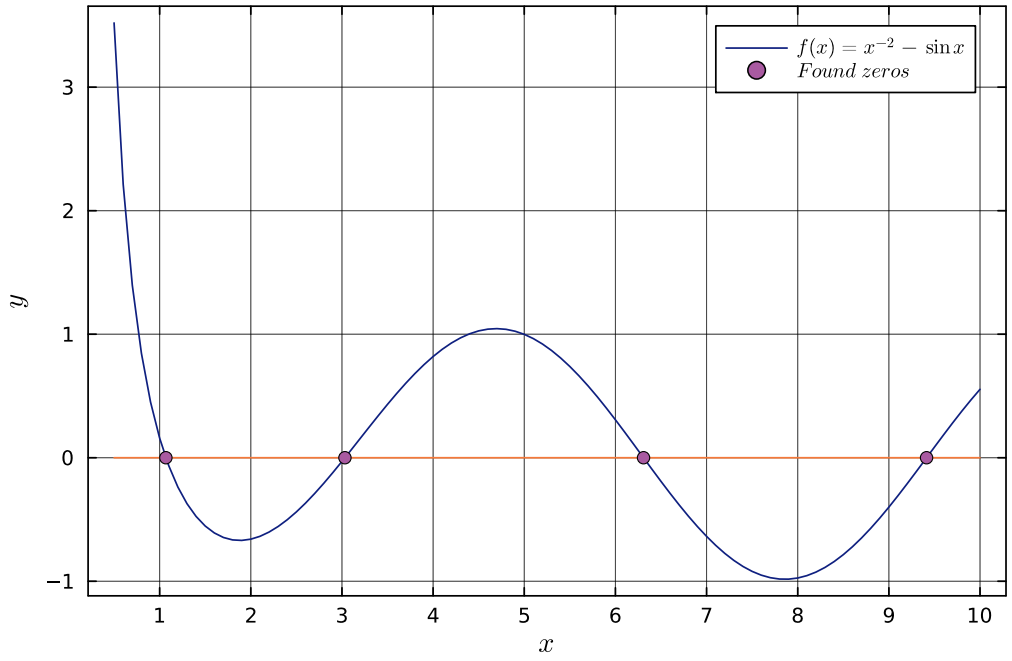


Figure 3.12: Grafico della funzione $f(x) = x^{-2} - \sin(x)$ nell'intervallo $[0.5, 10]$.

Radice	Radice stimata	Punti iniziali	Numero di iterazioni
r_1	1.068	[1.0, 2.0]	11
r_2	3.033	[2.0, 4.0]	8
r_3	6.308	[6.0, 7.0]	7
r_4	9.413	[9.0, 10.0]	7

Table 3.2: Radici stimate e punti iniziali per il metodo della secante.

4 Interpolazioni

4.1 Esercizi 4.2.1, 4.2.2

1. Scrivi un programma che implementi la formula baricentrica dell'interpolazione di Lagrange per nodi in posizioni generiche.

2. In ciascun caso, interpola la funzione data usando n nodi equispaziati nell'intervallo indicato. Rappresenta graficamente ciascuna funzione interpolante insieme alla funzione esatta.

   (a) $f(x) = \ln(x), \quad n = 2, 3, 4, \quad x \in [1, 10]$

   (b) $f(x) = \tanh(x), \quad n = 2, 3, 4, \quad x \in [-3, 2]$

   (c) $f(x) = \cosh(x), \quad n = 2, 3, 4, \quad x \in [-1, 3]$

   (d) $f(x) = |x|, \quad n = 3, 5, 7, \quad x \in [-2, 1]$

4.1.1 Soluzione

(1) Si è implementata la formula baricentrica dell'interpolazione di Lagrange per nodi in posizioni generiche, e la si è testata nell'esercizio 4.2.2.

(2) In questo esercizio, nonostante sia data la possibilità di utilizzare la formula baricentrica semplificata dai nodi equispaziati, si è scelto di utilizzare la formula baricentrica per nodi generici, come test dell'implementazione dell'esercizio precedente. In esercizi successivi sarà richiesto di implementare la formula baricentrica per nodi di Chebyshev, e in tale contesto si è sviluppata una funzione Julia che chiede la natura dei nodi e restituisce la funzione baricentrica corrispondente, anche per il caso equispaziato.
Si riportano i grafici delle funzioni interpolanti in figura 4.1.

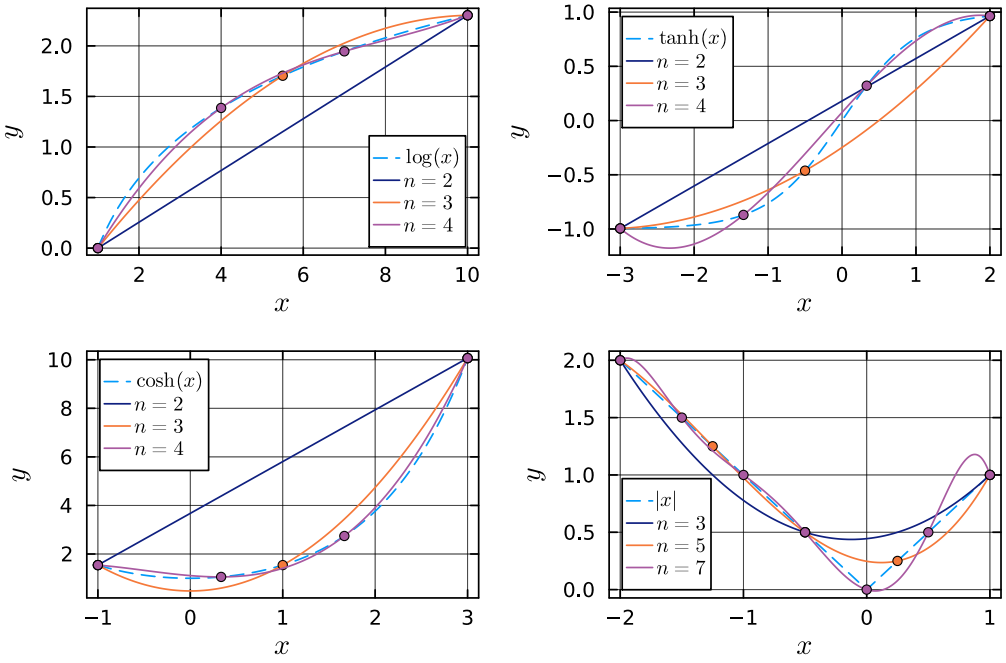


Figure 4.1: Grafici delle funzioni interpolanti per i vari casi.

Come si può notare, all'aumentare del numero di nodi l'interpolazione migliora. Il grafico della funzione $|x|$ però richiede un numero di nodi maggiore per essere interpolato con una certa accuratezza, ciò è dovuto alla non derivabilità della funzione in $x = 0$. Il problema è che si sta cercando di interpolare una funzione non derivabile con un polinomio, notoriamente derivabile in ogni punto.

4.2 Esercizi 4.4.1, 4.4.2

1. Scrivi un programma che implementi la formula baricentrica dell'interpolazione di Lagrange per il caso particolare dei nodi di Chebyshev, utilizzando i risultati analitici per i pesi. Verifica la correttezza della tua implementazione confrontandola con la routine per nodi generici.

2. (a) Per ciascun caso sotto riportato, calcola il polinomio interpolante usando n nodi di Chebyshev di seconda specie in $[-1, 1]$ per $n = 4, 8, 12, \ldots, 60$. Per ogni valore di n, calcola l'errore in norma ∞ (ossia, $||f - p||_\infty = \max_{x\in[-1,1]} |p(x) - f(x)|$ valutato su almeno 4000 valori di x). Utilizzando una scala log-lineare, rappresenta graficamente l'errore in funzione di n e determina una buona approssimazione della costante K.
   - $f(x) = 1/(25x^2 + 1)$
   - $f(x) = \tanh(5x + 2)$
   - $f(x) = \cosh(\sin x)$
   - $f(x) = \sin(\cosh x)$

(b) Realizza un grafico analogo utilizzando n punti equidistanti per l'interpolazione.

4.2.1 Soluzione

(1) Si è implementata la formula baricentrica dell'interpolazione di Lagrange per il caso particolare dei nodi di Chebyshev, e la si è testata nell'esercizio 4.4.2, confrontandola con il caso di nodi equidistanti.

(2a) Il calcolo dell'errore in norma ∞ è stato effettuato con un passo di campionamento sulle x pari a 0.0005, in modo da ottenere 4000 punti tra -1 e 1. Si riportano in figura 4.2 a sinistra i grafici dell'errore in norma ∞ in funzione di n, per i casi delle varie funzioni.

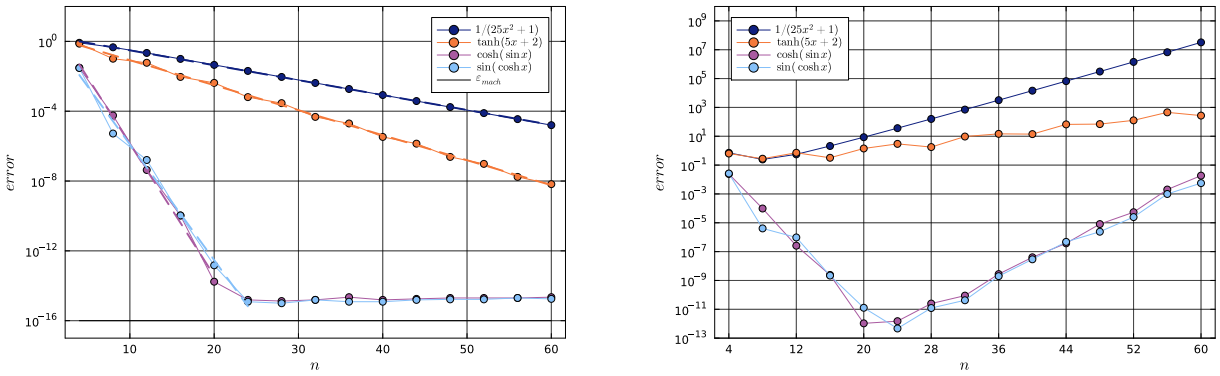


Figure 4.2: Errori per nodi di Chebyshev (sinistra) e nodi equidistanti (destra).

L'errore in norma ∞ decresce all'aumentare di n, come ci si aspetta data la scelta dei nodi di Chebyshev. La decrescita ha andamento esponenziale, fino ad arrestare la sua discesa intorno al valore di $\varepsilon_{mach} = 10^{-16}$.
La relazione che ha consentito di stimare la costante K è la seguente:

$$|p(x) - f(x)| = C \cdot K^{-n}, \qquad \log_{10} |p(x) - f(x)| = \log_{10}(C) + n(-\log_{10}(K)), \tag{4.1}$$

dove si è scelto di linearizzare i dati con il logaritmo per stimare la costante K. I fit sono stati eseguiti scartando i punti corrispondenti a errori minori di 10^{-16} e in figura 4.2 sono rappresentati dalle linee tratteggiate.
Si riportano i valori di K ottenuti per le varie funzioni:

- $f_1(x) = 1/(25x^2 + 1)$: $K = 1.217$.
- $f_2(x) = \tanh(5x + 2)$: $K = 1.391$.
- $f_3(x) = \cosh(\sin x)$: $K = 5.704$.
- $f_4(x) = \sin(\cosh x)$: $K = 4.598$.

(2b) Si è ripetuto lo stesso procedimento con i nodi equidistanti, e si riportano in figura 4.2 a destra i grafici dell'errore in norma ∞ in funzione di n, per i vari casi. Come atteso, l'errore in norma ∞ non decresce globalmente all'aumentare di n. Per le prime due funzioni l'errore è crescente fin dall'inizio, mentre per le ultime due funzioni l'errore decresce fino a un certo punto, per tornare a crescere successivamente. È un comportamento noto come *fenomeno di Runge*, ed è dovuto alla scelta dei nodi equidistanti.

4.3 Esercizio 4.4.3

I punti di Chebyshev possono essere utilizzati anche quando l'intervallo di interpolazione è $[a, b]$ invece di $[-1, 1]$, tramite un opportuno cambiamento di variabile. Traccia il polinomio interpolante di $f(z) = \cosh(\sin z)$ su $[0, 2\pi]$ utilizzando $n = 40$ nodi di Chebyshev.

4.3.1 Soluzione

Il cambio di variabile utilizzato per passare da $[-1, 1]$ a $[a, b]$ è il seguente:

$$z = \psi(x) = a + (b - a)\frac{x + 1}{2}, \tag{4.2}$$

dove x è il nodo di Chebyshev.
L'algoritmo implementato tiene conto di questo cambio di variabile. In particolare si sono calcolati i nodi di Chebyshev, li si è trasformati con 4.2 e poi si è calcolato il polinomio nei nuovi punti con i soliti pesi. Si riporta in figura 4.3 il grafico ottenuto con $n = 40$ nodi di Chebyshev, per la funzione $f(z) = \cosh(\sin z)$ su $[0, 2\pi]$.

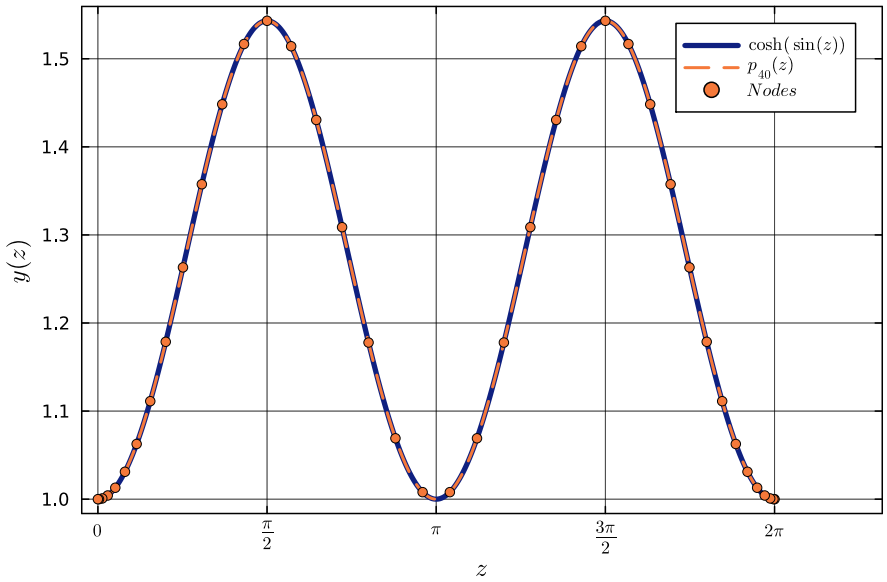


Figure 4.3: $f(z) = \cosh(\sin z)$ su $[0, 2\pi]$ con $n = 40$ nodi di Chebyshev.

4.4 Esercizio 4.4.4

Siano $x_1, \ldots, x_n$ i punti di Chebyshev standard. Questi vengono mappati nella variabile z come $z_i = \phi(x_i)$ per ogni i, dove ϕ è una trasformazione sulla retta reale. Supponiamo che $f(z)$ sia una funzione data, il cui dominio è l'intera retta reale. Allora i valori della funzione $y_i = f(z_i)$ possono essere associati ai nodi di Chebyshev x_i, portando a un polinomio interpolante $p(x)$. Questo implica a sua volta una funzione interpolante sulla retta reale, definita come

$$q(z) = p\big(\phi^{-1}(z)\big) = p(x)\,. \tag{4.3}$$

Implementa questa idea per tracciare un interpolante di $f(z) = (z^2 - 2z + 2)^{-1}$ usando $n = 30$. Il tuo grafico deve mostrare $q(z)$ valutato in 1000 punti equispaziati in $[-6, 6]$, con marcatori sui valori nodali (quelli che ricadono nella finestra $[-6, 6]$).
Suggerimento: se preferisci evitare di gestire potenziali infiniti considera l'uso dei nodi di Chebyshev di prima specie.

4.4.1 Soluzione

Il cambio di variabile utilizzato è il seguente:

$$z = \phi(x) = \frac{2x}{1 - x^2}, \tag{4.4}$$

La funzione Julia di interpolazione implementata tiene conto di questo cambio di variabile. In particolare calcola i nodi di Chebyshev del secondo tipo nell'intervallo $[-1, 1]$, denotati come x_i, e li trasforma con 4.4 per ottenere i nodi di Chebyshev z_i, ora distribuiti sull'intera retta reale. I punti per i quali passa il polinomio vengono indicati con $y_i = f(z_i)$.
La funzione restituisce il polinomio interpolante $q(z)$. Esso è una funzione Julia che utilizza la trasformazione inversa di 4.4 per portare i punti z in cui verrà calcolato il polinomio q nell'intervallo $[-1, 1]$. A questo punto avviene l'interpolazione con i nodi di Chebyshev x_i e i valori y_i, e così il polinomio restituito interpola su tutto l'asse reale avendo calcolato l'interpolazione su un intervallo finito.
Si riporta in figura 4.4 il grafico della funzione $f(z) = (z^2 - 2z + 2)^{-1}$ e della sua interpolazione con $n = 30$ nodi di Chebyshev. Per motivi di rappresentazione grafica l'intervallo è stato limitato a $[-6, 6]$, si noti infatti che il numero di nodi in grafico è pari a venti: i dieci restanti sono al di fuori dell'intervallo.

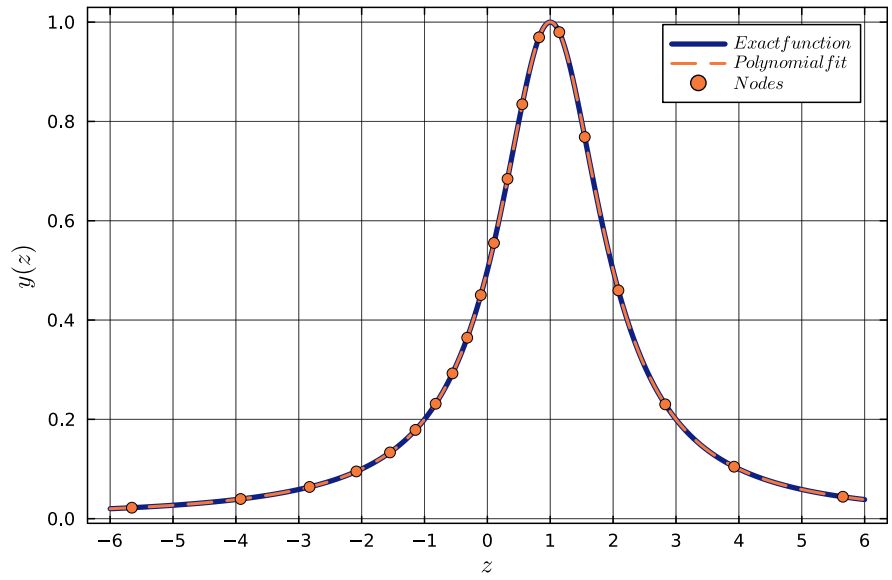


Figure 4.4: $f(z) = (z^2 - 2z + 2)^{-1}$ e relativa interpolazione con 30 nodi di Chebyshev.

4.5 Esercizio 4.6.1

Ognuna delle seguenti funzioni è 2-periodica. Scrivi una funzione che esegua l'interpolazione trigonometrica su $[-1, 1]$ e traccia la funzione insieme ai suoi interpolanti trigonometrici per $n = 3, 6, 9$. Poi, per $n = 2, 3, \ldots, 30$, calcola l'errore in norma ∞ dell'interpolante trigonometrico campionando in almeno 1000 punti, e realizza un grafico di convergenza in scala semilogaritmica.

(a) $f(x) = e^{\sin(2\pi x)}$

(b) $f(x) = \log[2 + \sin(3\pi x)]$

(c) $f(x) = \cos^{12}[\pi(x - 0.2)]$

4.5.1 Soluzione

Ai fini dell'esercizio si è sviluppata una funzione Julia che calcola l'interpolazione trigonometrica su un intervallo $[-1, 1]$ per una funzione $f(x)$, e restituisce il polinomio interpolante. Dato che le funzioni sono 2-periodiche le si è interpolate sull'intervalo $[-1, 1]$. Per poter interpolare funzioni T-periodiche bisogna inserire nell'algoritmo un cambio di variabile, proprio come per l'esercizio 4.4.3.
Si riportano nelle figure 4.5, 4.6 e 4.7 i grafici delle varie funzioni e dei loro interpolanti trigonometrici per $n = 3, 6, 9$. A fianco si riportano i grafici dell'errore in norma ∞ dell'interpolante trigonometrica al variare del numero di nodi, avendo campionato in 1000 punti l'intervallo.

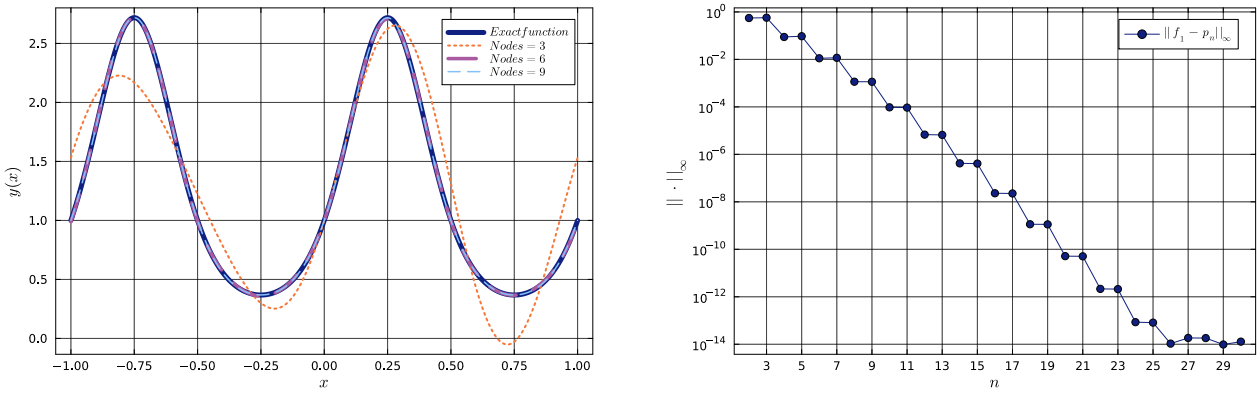


Figure 4.5: Studio delle interpolazioni trigonometriche per la funzione $f(x) = e^{\sin(2\pi x)}$.

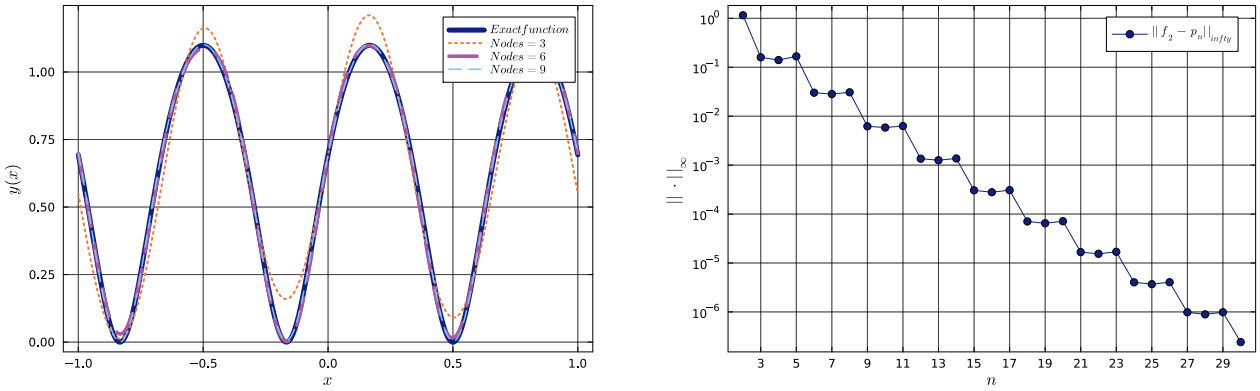


Figure 4.6: Studio delle interpolazioni trigonometriche per la funzione $f(x) = \log[2+\sin(3\pi x)]$.

Osservazioni Si osserva che l'errore in norma ∞ diminuisce all'aumentare del numero di nodi in tutti e tre i casi, confermando la convergenza dell'interpolazione trigonometrica. In generale l'errore non diminuisce mai oltre il valore di 10^{-14} per motivazioni legate alla rappresentazione numerica. È interessante notare che in tutti e tre i casi l'errore ha un andamento a gradino. Questo fenomeno mostra che l'aggiunta di nodi non sempre migliora l'accuratezza dell'interpolazione, anche se globalmente l'errore diminuisce.
Curioso è il comportamento dell'errore della terza funzione, in cui per $n = 12$ si osserva una diminuizione drastica dell'errore. Sulle ragioni di questo fatto si è avanzata una spiegazione. Osservando il grafico 4.7 si nota che all'aumento di n le zone che sono peggio approssimate sono quelle di appiattimento della funzione. Sono quelle che contribuiscono maggiormente all'errore, calcolato come il massimo della distanza tra la funzione e il polinomio interpolante, su tutto l'intervallo.

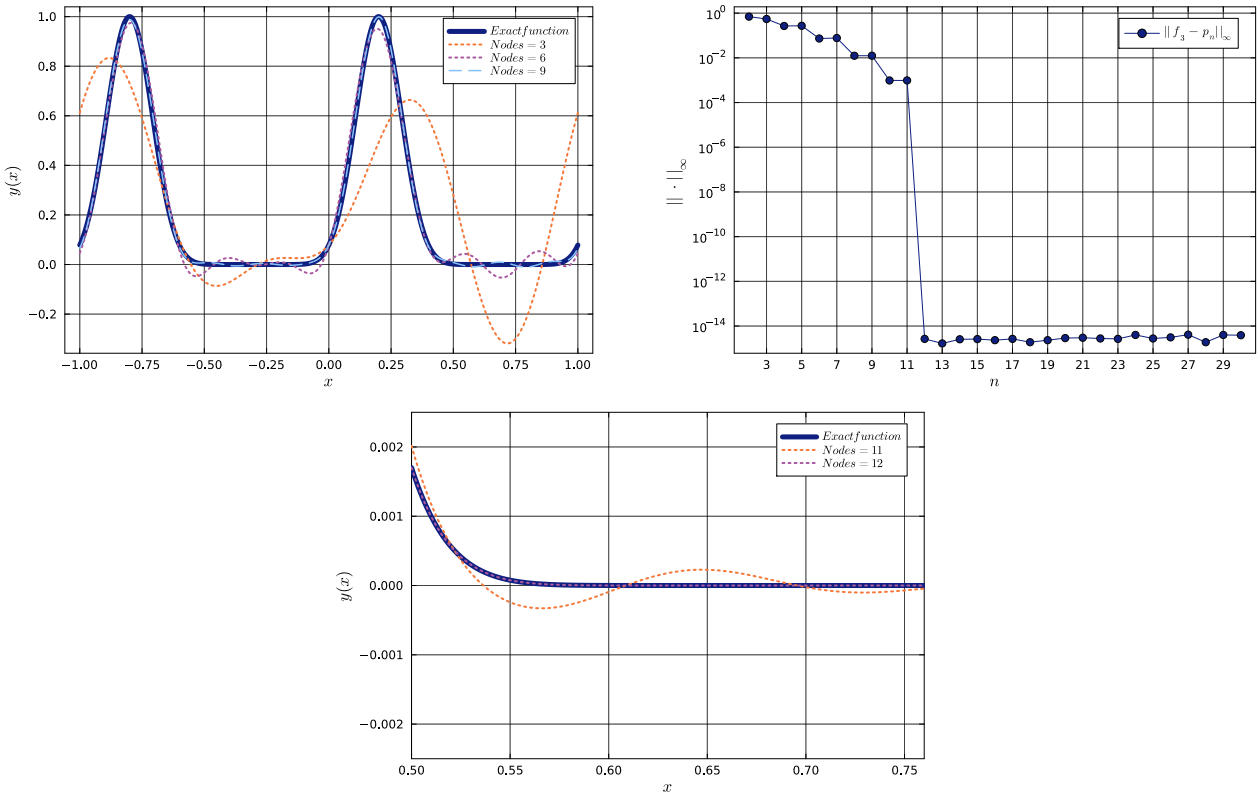


Figure 4.7: Studio delle interpolazioni trigonometriche per la funzione $f(x) = \cos^{12}[\pi(x - 0.2)]$.

La figura in basso a figura 4.7 si ha un ingrandimento della regione di plateau e si osserva che per $n = 12$ il polinomio interpolante si appiattisce definitivamente, con la conseguente riduzione dell'errore.

5 Integrazione numerica

5.1 Esercizi 5.1.1, 5.1.2

1. Scrivi un programma che implementi la regola del trapezio composta per un intervallo generico $[a, b]$.

2. Per ciascuno dei seguenti integrali, utilizza la regola del trapezio per stimare il valore dell'integrale con $m = 10 \cdot 2^k$ nodi, per $k = 1, 2, \ldots, 10$. Per ogni caso, rappresenta in scala log-log l'errore in funzione di m e verifica se la convergenza è di secondo ordine.

(a) $\displaystyle\int_0^1 x \log(1 + x)\, dx = \frac{1}{4}$ (b) $\displaystyle\int_0^1 x^2 \tan^{-1} x\, dx = \frac{\pi - 2 + 2\log 2}{12}$

(c) $\displaystyle\int_0^{\pi/2} e^x \cos x\, dx = \frac{e^{\pi/2} - 1}{2}$ (d) $\displaystyle\int_0^1 \frac{\tan^{-1}(\sqrt{2 + x^2})}{(1 + x^2)\sqrt{2 + x^2}}\, dx = \frac{5\pi^2}{96}$

(e) $\displaystyle\int_0^1 \sqrt{x} \log(x)\, dx = -\frac{4}{9}$ (f) $\displaystyle\int_0^1 \sqrt{1 - x^2}\, dx = \frac{\pi}{4}$

5.1.1 Soluzione

(1) Si è implementata la regola del trapezio composta per un intervallo generico $[a, b]$, e la si è testata nell'esercizio 5.1.2.

(2) Si è calcolato l'integrale con la regola del trapezio per tutti i casi, al variare del numero di nodi. Oltre ai grafici dell'errore, si sono rappresentate le derivate seconde delle integrande, in modo verificare la convergenza della regola del trapezio. Infatti l'errore della regola del trapezio scala al secondo ordine con il passo di campionamento, ma ha per coefficiente la derivata seconda dell'integranda, calcolata in un certo ξ appartenente all'intervallo di integrazione. In formule:

$$R_T(h) = -\frac{(b-a)^3}{12} h^2 f''(\xi)$$

Si riportano nelle figure 5.1, 5.2, 5.3, 5.4, 5.5, 5.6 gli studi dei vari casi.

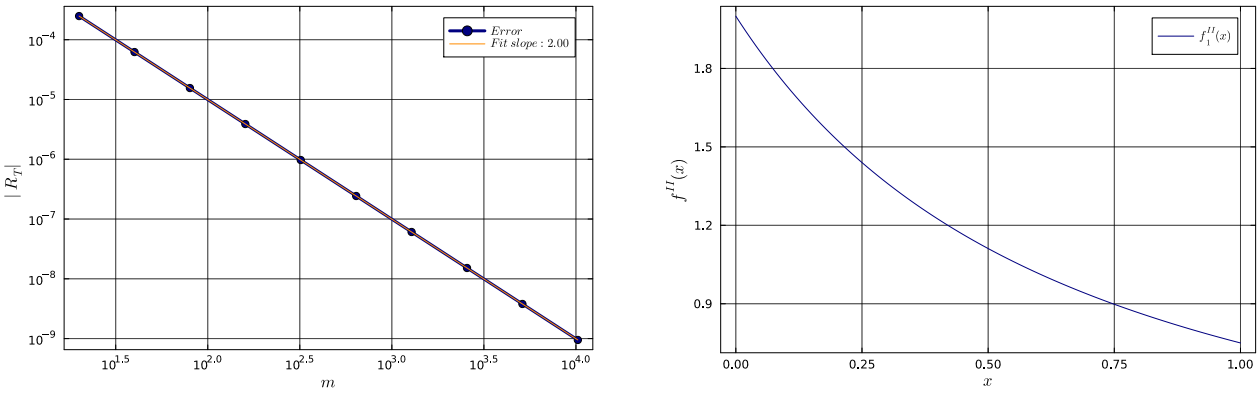


Figure 5.1: Studio della regola del trapezio per l'integrale $\int_0^1 x \log(1+x)\, dx$.

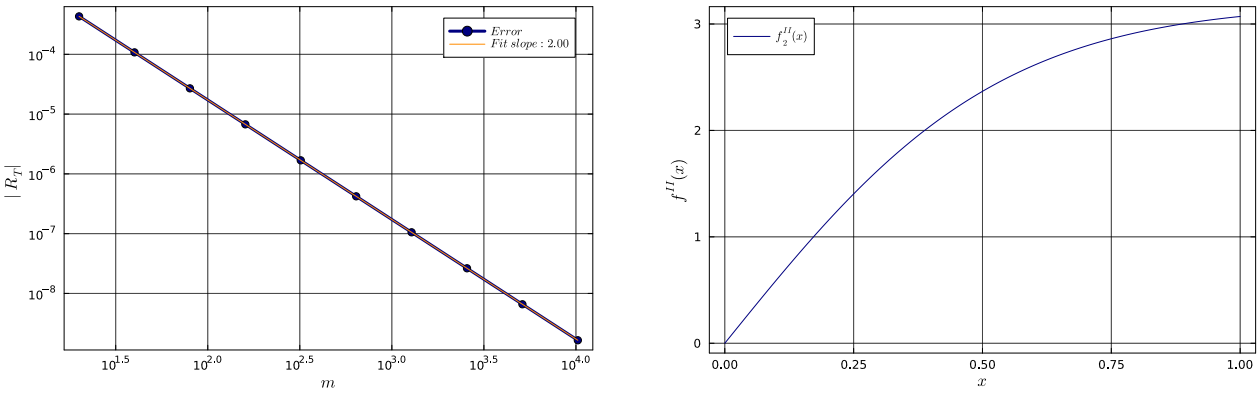


Figure 5.2: Studio della regola del trapezio per l'integrale $\int_0^1 x^2 \tan^{-1} x\, dx$.

Nei grafici viene riportata la pendenza del fit dell'errore. Nei primi quattro casi l'errore decresce con un andamento quadratico nel passo h, come atteso. Nel quinto e sesto caso l'errore decresce con un ordine inferiore. Come già accennato, questo è dovuto al fatto che l'integranda ha derivata seconda discontinua nel dominio di integrazione.

Commento: Qui e nei prossimi esercizi l'interpolazione per l'errore R_T (o R_S) è stata eseguita in funzione del passo h, anche se in grafico viene riportata rispetto al numero di nodi m. Ciò vuole dire che le pendenze si riferiscono non alla retta riportata in figura ma alla retta del fit, rappresentando l'ordine di convergenza del metodo studiato.

5.2 Esercizi 5.1.3, 5.1.4

3. Scrivi un programma che implementi la regola di Simpson composta per un intervallo $[a, b]$.

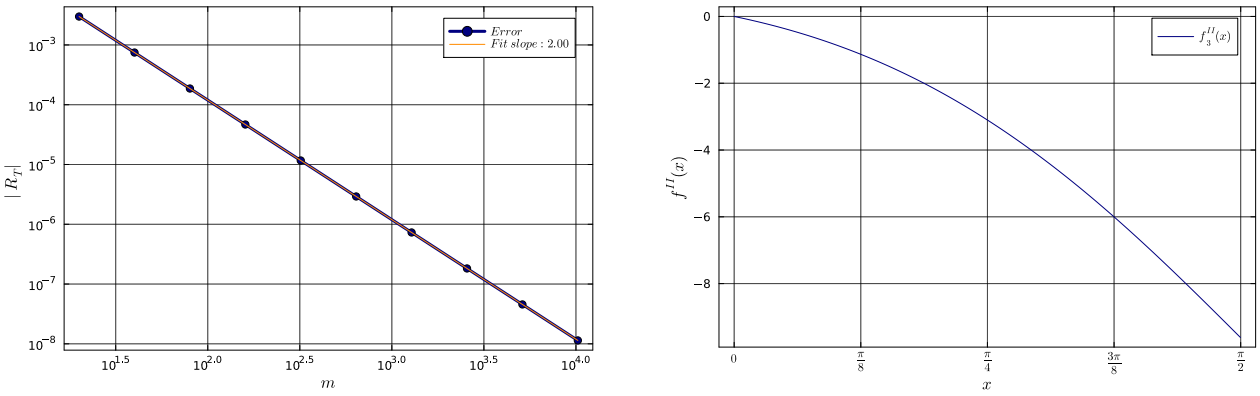


Figure 5.3: Studio della regola del trapezio per l'integrale $\int_0^{\pi/2} e^x \cos x \, dx$.

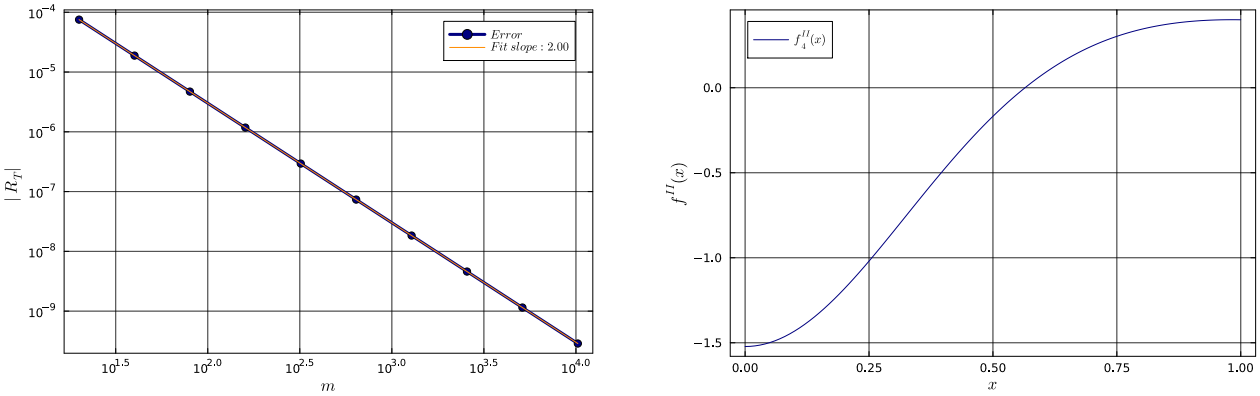


Figure 5.4: Studio della regola del trapezio per l'integrale $\int_0^1 \frac{\tan^{-1}(\sqrt{2+x^2})}{(1+x^2)\sqrt{2+x^2}} \, dx$.

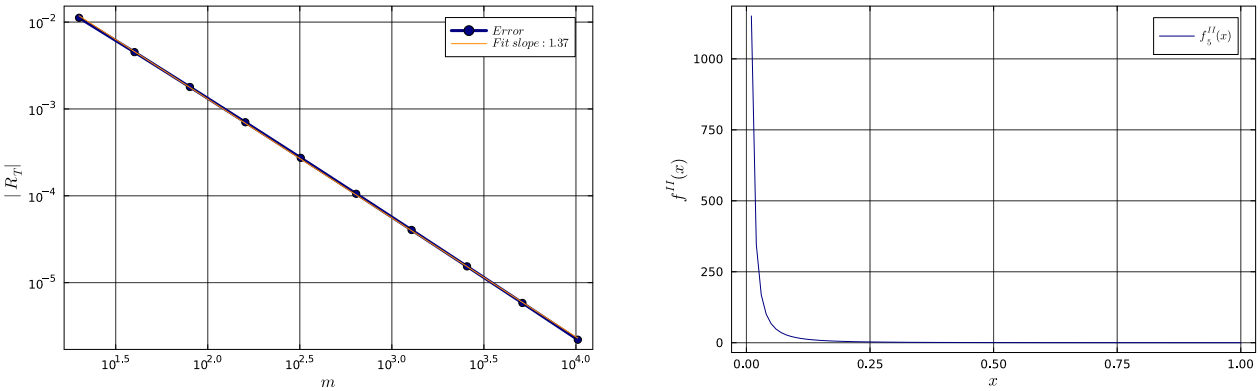


Figure 5.5: Studio della regola del trapezio per l'integrale $\int_0^1 \sqrt{x} \log(x) \, dx$.

4. Per ciascun integrale dell'esercizio 5.1.1, applica la formula di Simpson e confronta gli errori con la convergenza di ordine quattro.

5.2.1 Soluzione

(3) Si è implementata la regola di Simpson composta per un intervallo $[a, b]$, e la si è testata nell'esercizio 5.1.4.

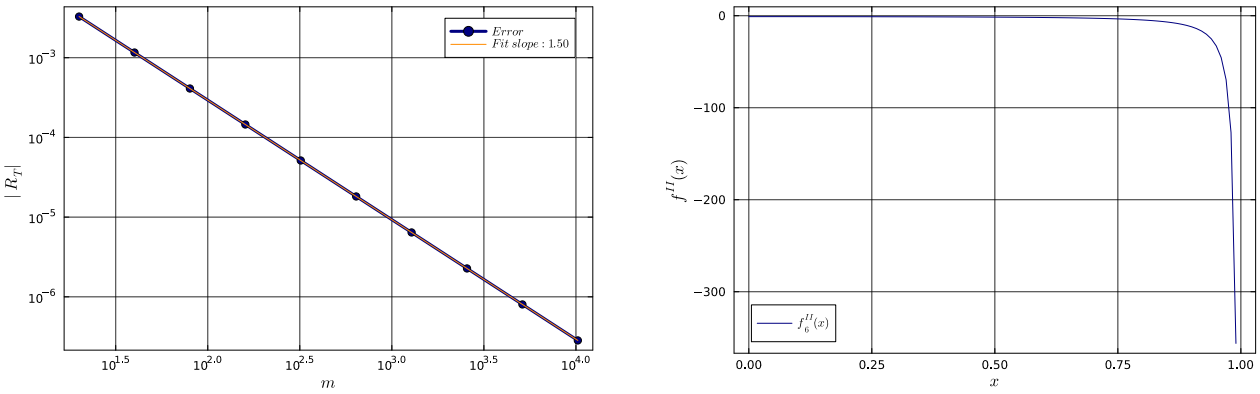


Figure 5.6: Studio della regola del trapezio per l'integrale $\int_0^1 \sqrt{1-x^2}\,dx$.

(4) Si è calcolato l'integrale con la regola di Simpson per tutti i casi, al variare del numero di nodi. Per le stesse ragioni dell'esercizio 5.1.2 si sono rappresentate, oltre ai grafici dell'errore, le derivate quarte dell'integranda. Nel caso di Simpson la formula dell'errore è:

$$R_S(h) = -\frac{(b-a)}{180}h^4 f^{(4)}(\xi), \qquad \text{con } \xi \in (a,b)$$

Si riportano nelle figure 5.7, 5.8, 5.9, 5.10, 5.11, 5.12 gli studi dei vari casi.

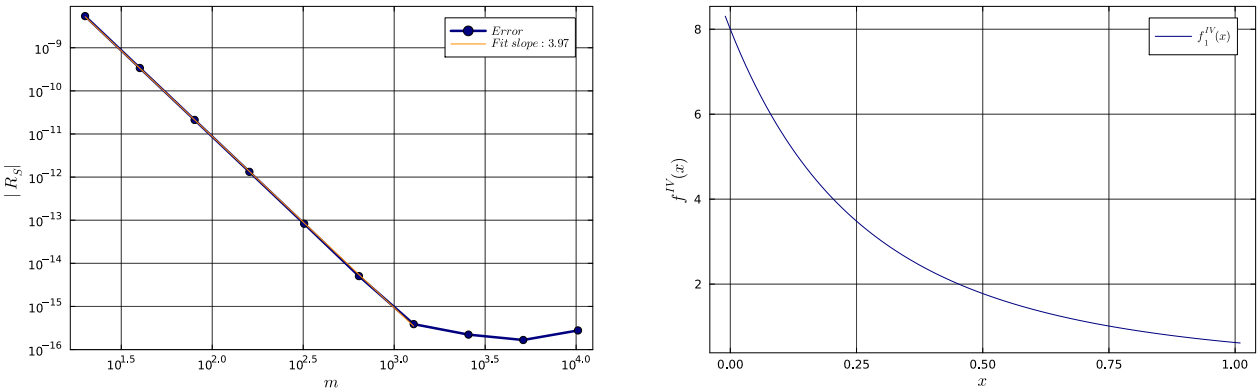


Figure 5.7: Studio della regola di Simpson per l'integrale $\int_0^1 x\log(1+x)\,dx$.

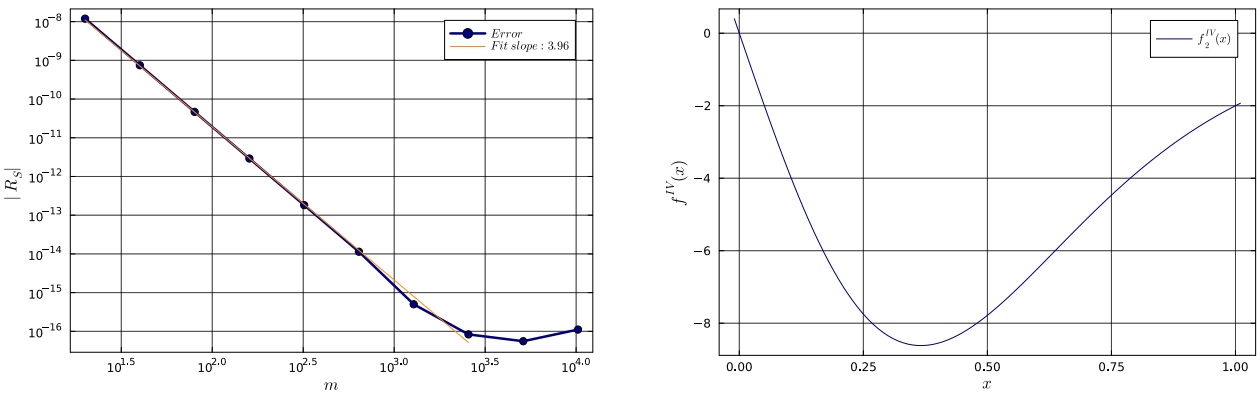


Figure 5.8: Studio della regola di Simpson per l'integrale $\int_0^1 x^2\tan^{-1}x\,dx$.

Si può osservare che l'errore decresce con un andamento quartico nei primi quattro casi, e con uno di ordine inferiore negli altri due. Nei primi quattro casi, si osserva che l'errore diminuisce

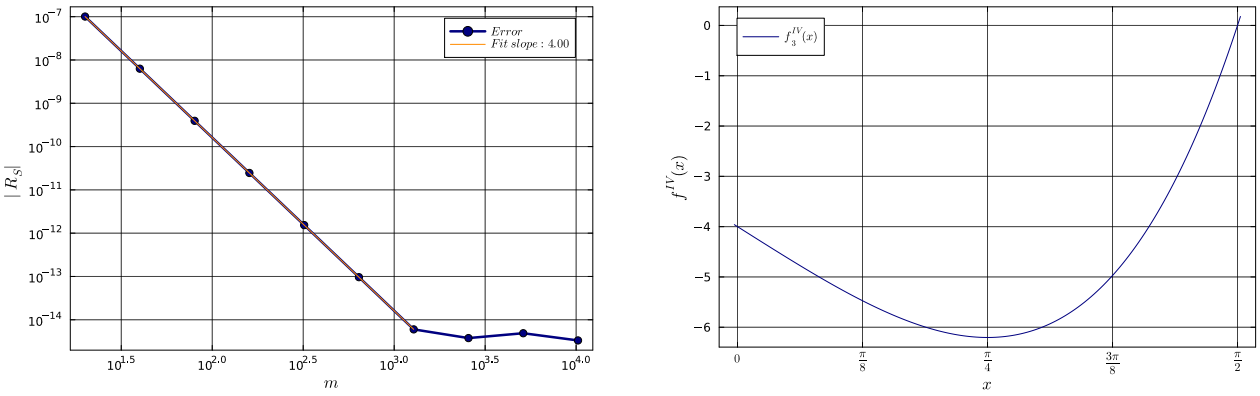


Figure 5.9: Studio della regola di Simpson per l'integrale $\int_0^{\pi/2} e^x \cos x \, dx$.

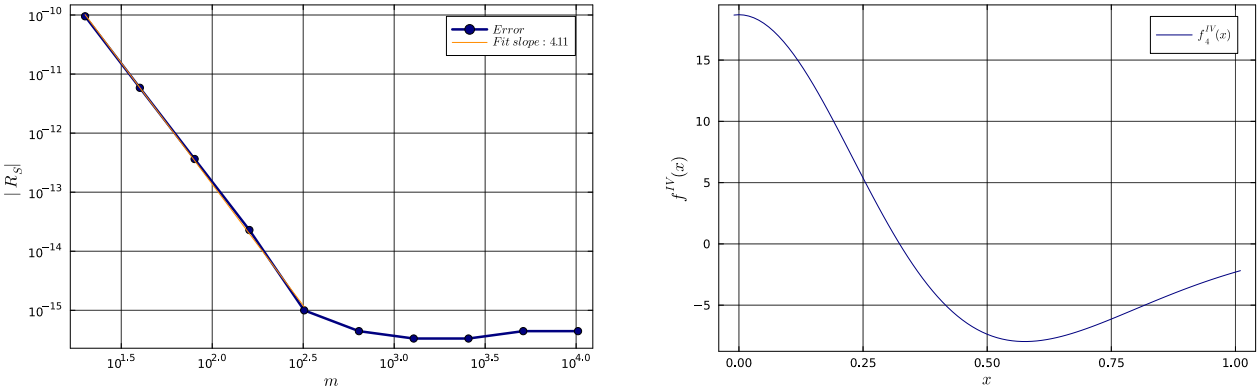


Figure 5.10: Studio della regola di Simpson per l'integrale $\int_0^1 \frac{\tan^{-1}(\sqrt{2+x^2})}{(1+x^2)\sqrt{2+x^2}} \, dx$.

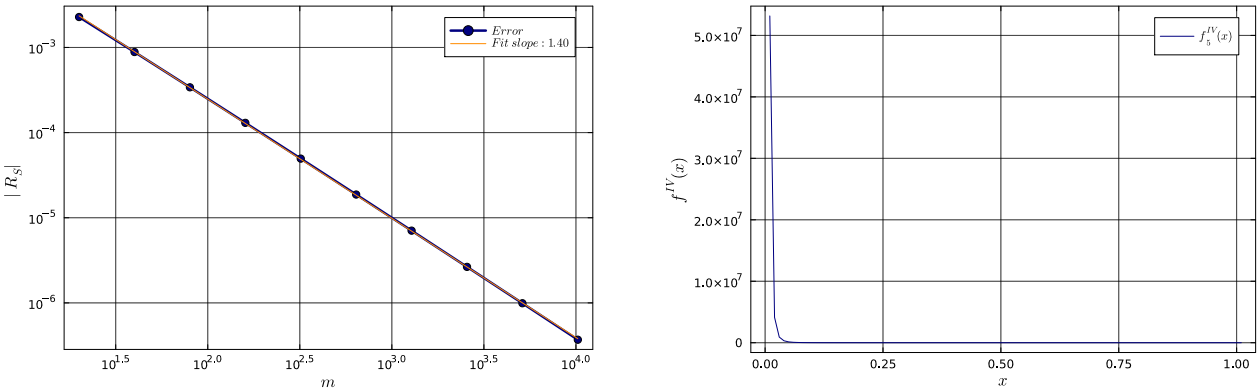


Figure 5.11: Studio della regola di Simpson per l'integrale $\int_0^1 \sqrt{x} \log(x) \, dx$.

rapidamente fino a ε_{mach}, per poi stabilizzarsi per motivazioni numeriche. Chiaramente il fit dell'errore è stato eseguito solo sui punti che si trovano al di sopra di questo valore.

5.3 Esercizio 5.3.1

1. Scrivi un programma che calcoli i nodi e i pesi per la quadratura di Gauss-Legendre nell'intervallo $[-1, 1]$. Di seguito alcuni suggerimenti per l'implementazione.
(a) Per trovare le radici x_k, con $k = 1, \ldots, n$, del polinomio di Legendre $P_n(x)$ di ordine n, usa il metodo di Newton. Come condizione iniziale per la k-esima radice, utilizza

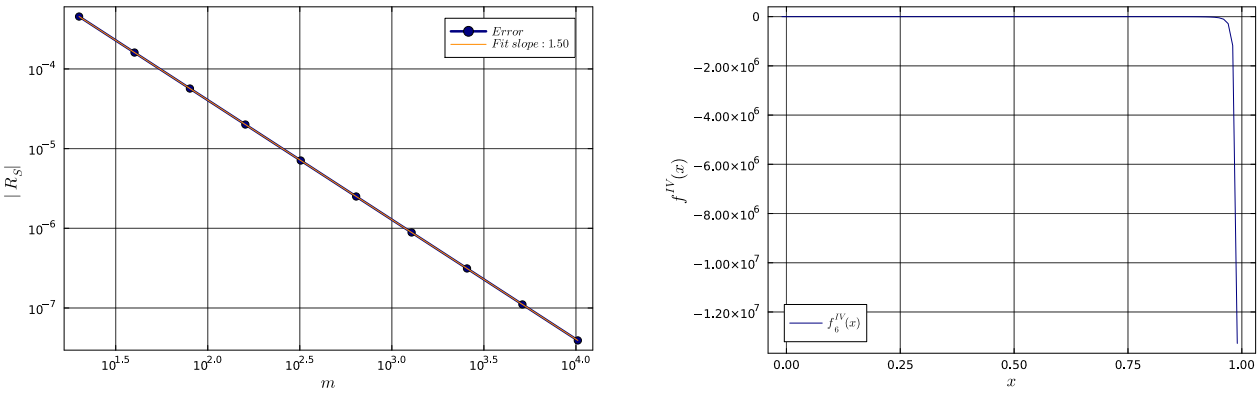


Figure 5.12: Studio della regola di Simpson per l'integrale $\int_0^1 \sqrt{1-x^2}\,dx$.

$$x_k^{(0)} = \cos(\phi_k)\,, \qquad \phi_k = \frac{4k-1}{4n+2}\pi\,. \tag{5.1}$$

(b) Per applicare il metodo di Newton è necessario valutare $P_n(x_k^{(i)})$ e la derivata $P_n'(x_k^{(i)})$, dove $x_k^{(i)}$ è la i-esima iterazione nella ricerca della k-esima radice x_k. Puoi ottenerle risolvendo la relazione di ricorrenza

$$(m+1)P_{m+1}(x) = (2m+1)xP_m(x) - mP_{m-1}(x)\,, \tag{5.2}$$

per $m = 1, \ldots, n-1$, dove $P_0(x) = 1$ e $P_1(x) = x$, con $x = x_k^{(i)}$. La derivata si calcola poi tramite la relazione

$$(x^2-1)P_n'(x) = n[xP_n(x) - P_{n-1}(x)]\,. \tag{5.3}$$

(c) Una volta raggiunta la radice x_k entro una certa tolleranza, il valore della derivata $P_n'(x_k)$ permette di ottenere il peso corrispondente tramite:

$$w_k = \frac{2}{(1-x_k^2)[P_n'(x_k)]^2}\,. \tag{5.4}$$

(d) Verifica i tuoi risultati confrontandoli con quelli tabulati.

5.3.1 Soluzione

Per comodità di implementazione la soluzione riporta i quattro passaggi nel seguente ordine: (b), (a), (c), (d).

(b) Si è sviluppato un algoritmo che calcola sia i polinomi di Legendre $P_n(x)$ che le loro derivate $P_n'(x)$ utilizzando la relazione di ricorrenza 5.2. Si riporta in figura 5.13 il grafico dei polinomi e delle loro derivate fino ad ordine $n = 4$.

(a) (c) (d) Si è implementato il metodo di Newton per trovare le radici dei polinomi di Legendre $P_n(x)$ e dei pesi associati. Una volta calcolati gli zeri di $P_n(x)$, si sono trovati i pesi associati utilizzando la formula 5.4. Si riporta in tabella 5.1 il confronto tra i nodi e pesi calcolati con i corrispettivi valori tabulati, con ordine del polinomio $n = 4$.

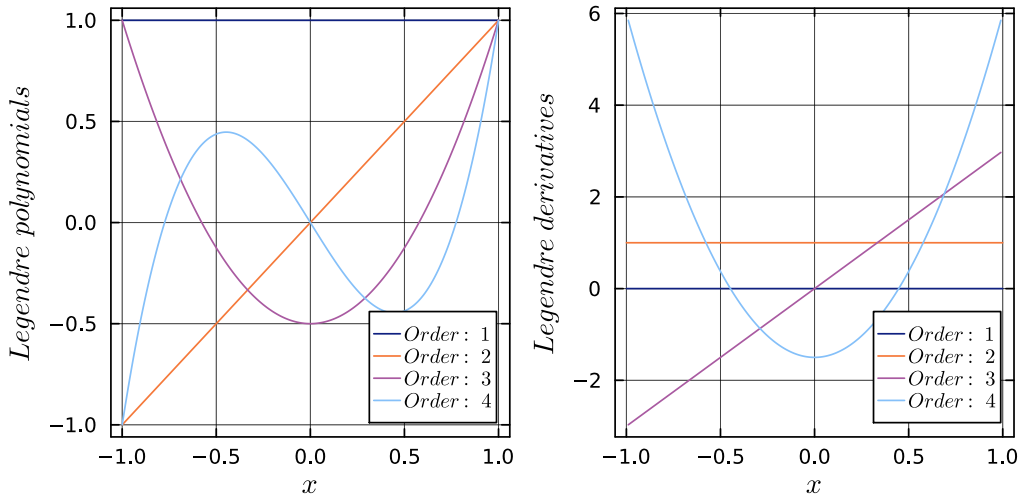


Figure 5.13: Polinomi di Legendre $P_n(x)$ e le loro derivate $P'_n(x)$ fino ad ordine $n = 4$.

Table 5.1: Confronto tra zeri e pesi del polinomio di Legendre di ordine 4

x_i	x_i^{tab}	w_i	w_i^{tab}
0.86114	0.86114	0.65215	0.65215
0.33998	0.33998	0.34785	0.34785
-0.33998	-0.33998	0.34785	0.34785
-0.86114	-0.86114	0.65215	0.65215

5.4 Esercizio 5.4.1

Per ciascun integrale, utilizza le regole di quadratura di Gauss-Legendre e Clenshaw-Curtis con $n = 4, 6, 8, \ldots, 40$. Rappresenta graficamente gli errori $|I_n - I|$ di entrambi i metodi in funzione di n su una scala semi-logaritmica. Implementa la regola di Clenshaw-Curtis solo per n pari.

(a) $\displaystyle\int_{-1}^{1} e^{-4x}\,dx = \frac{1}{2}\sinh(4)$ (b) $\displaystyle\int_{-1}^{1} e^{-9x^2}\,dx = \frac{\sqrt{\pi}}{3}\,\text{erf}(3)$

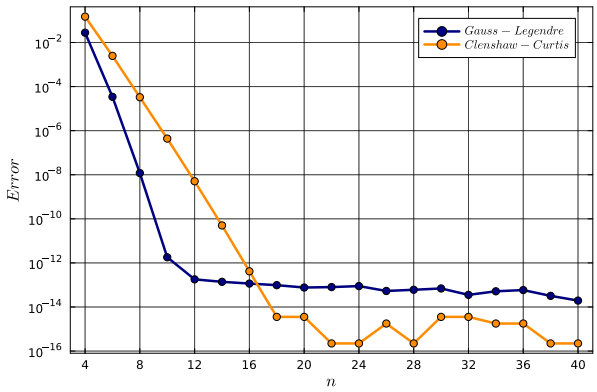
(c) $\displaystyle\int_{-1}^{1} \text{sech}(x)\,dx = 2\tan^{-1}[\sinh(1)]$ (d) $\displaystyle\int_{-1}^{1} \frac{1}{1+9x^2}\,dx = \frac{2}{3}\tan^{-1}(3)$

(e) $\displaystyle\int_{\pi/2}^{\pi} x^2 \sin 8x\,dx = -\frac{3\pi^2}{32}$

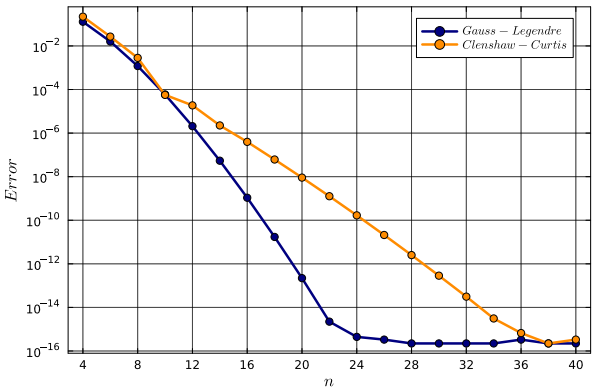
5.4.1 Soluzione

Per ciascun integrale sono state calcolate le approssimazioni utilizzando le regole di quadratura di Gauss-Legendre e Clenshaw-Curtis con $n = 4, 6, 8, \ldots, 40$.
Dato che la convergenza dei due metodi dipende dalle proprietà di olomorfia delle funzioni integrande estese al piano complesso, i grafici non sono presentati nell'ordine richiesto. Si riportano in figura 5.14, gli studi degli errori per gli integrali di funzioni olomorfe.

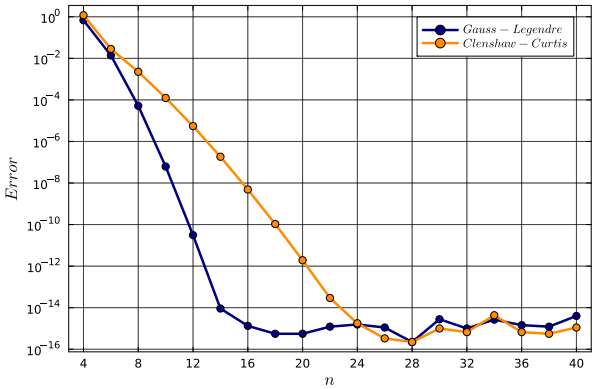
I grafici di figura 5.14 mostrano che gli errori decrescono in maniera molto rapida all'aumentare di n, raggiungendo l'ordine di grandezza di ε_{mach} per n piccoli. Questo fenomeno è legato all'olomorfia dell'integranda su tutto il piano complesso.
Il metodo di Clenshaw-Curtis integra esattamente polinomi di ordine $n-1$, il metodo di Gauss-Legendre integra esattamente polinomi di ordine $2n-1$. Tale proprietà è mostrata graficamente dal fatto che gli errori di Clenshaw sono sempre maggiori degli errori di Gauss.
Infine si può mostrare matematicamente che inizialmente gli errori del metodo di Clenshaw-Curtis coincidono con gli errori del metodo di Gauss-Legendre, proprio come si nota in figura.



(a)



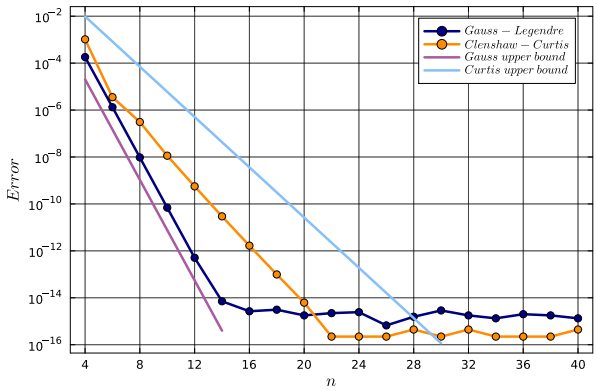
(b)



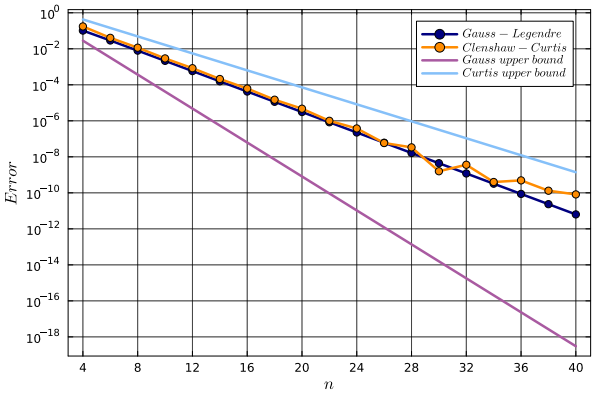
(e)

Figure 5.14: Studio degli errori per gli integrali di funzioni olomorfe.

Si riportano in figura 5.15 gli studi degli errori per gli integrali di funzioni non olomorfe.



(c)



(d)

Figure 5.15: Studio degli errori per gli integrali di funzioni non olomorfe.

Nei casi di figura 5.15 si è potuto studiare l'errore in maniera più accurata tramite le ellissi di Bernstein. Dalla teoria è noto che la più grande ellissi nel piano complesso definita da:

$$E_\rho = \left\{ z \in C : z = \frac{1}{2}\left(\rho e^{i\theta} + \rho^{-1} e^{-i\theta}\right),\ \theta \in [0, 2\pi) \right\}. \tag{5.5}$$

non contenente singolarità determina il parametro ρ delle seguenti maggiorazioni:

$$|I[f] - I_n[f]| \leq \frac{64}{15}\frac{M\rho^{1-n}}{\rho^2-1}\,, \qquad |I[f] - I_n[f]| \leq \frac{64}{15}\frac{M\rho^{-2n}}{\rho^2-1}\,. \tag{5.6}$$

La prima riguarda Clenshaw-Curtis, la seconda Gauss-Legendre.
Si è perciò calcolato il parametro ρ per i casi (c) e (d), sapendo che le singolarità delle integrande estese al piano complesso si trovano in $\frac{i\pi}{2} + ik\pi$ per $k \in \mathbb{N}$ nel primo caso, in $\pm\frac{i}{3}$ nel secondo. Tramite ρ si sono stimati i limiti superiori degli errori al variare di n. Nei grafici vengono riportate le rette, ma dato che nelle 5.6 è presente anche M, il grafico si concentra sulla pendenza e non sulla traslazione verticale. Si può notare che nel caso (c) le rette ricalcano precisamente la pendenza degli errori.
Nel caso (d) gli errori di Curtis rispettano l'andamento atteso, al contrario degli errori di Legendre. Sono necessari più punti per osservare la regione in cui gli errori scalano come $O(n^2)$, infatti i grafici dell'errore sono ancora nella regione in cui gli errori coincidono e la precisione non è scesa sotto 10^{-11}. Per questi motivi in (d) non si osserva il comportamento atteso.

5.5 Esercizio 5.4.2

Per ciascun integrale improprio, calcola le approssimazioni utilizzando la regola di quadratura double-exponential appropriata con $N = n/2$ e $n = 4, 6, 8, \ldots, 60$. Rappresenta graficamente gli errori in funzione di n su una scala semi-logaritmica.

(a) $\displaystyle\int_{-\infty}^{\infty} \frac{1}{1+x^2+x^4}\,dx = \frac{\pi}{\sqrt{3}}$ (b) $\displaystyle\int_{-\infty}^{\infty} e^{-x^2}\cos(x)\,dx = e^{-1/4}\sqrt{\pi}$

(c) $\displaystyle\int_{-\infty}^{\infty} (1+x^2)^{-2/3}\,dx = \frac{\sqrt{\pi}\,\Gamma(1/6)}{\Gamma(2/3)}$ (d) $\displaystyle\int_{0}^{\infty} \frac{1}{1+x^2}\,dx = \frac{\pi}{2}$

(e) $\displaystyle\int_{0}^{\infty} \frac{e^{-x}}{\sqrt{x}}\,dx = \sqrt{\pi}$

5.5.1 Soluzione

Si riportano nella figura 5.16 gli studi degli errori per gli integrali impropri, con la regola di quadratura double-exponential.

In figura 5.16 possiamo notare che si sono riusciti a integrare correttamente integrali definiti su intervalli non limitati, grazie al cambio di varaiabile della quadratura double-exponential.
Nell'ultimo grafico si sono rappresentati gli errori dell'integrazione con il cambio di variabile standard e con l'utilizzo del cambio di variabile apposito per funzioni con e^{-x}. Si può notare che nel secondo caso l'errore ha una convergenza molto più rapida, raggiungendo l'ordine di ε_{mach}. Infine il caso (e) è particolarmente adatto all'integrazione doppio esponenziale poichè presenta singolarità all'estremo di integrazione.

5.6 Esercizio 5.4.3

Per ciascun integrale, calcola le approssimazioni utilizzando la regola di quadratura di Gauss-Legendre con $n = 4, 6, 8, \ldots, 60$ e la regola di quadratura double-exponential appropriata con $N = n/2$. Rappresenta graficamente gli errori in funzione di n su una scala semi-logaritmica.

(a) $\displaystyle\int_{0}^{1} \sqrt{x}\log(x)\,dx = -\frac{4}{9}$ (b) $\displaystyle\int_{0}^{1} \sqrt{1-x^2}\,dx = \frac{\pi}{4}$

(c) $\displaystyle\int_{0}^{1} (\log x)^2\,dx = 2$ (d) $\displaystyle\int_{0}^{\pi/2} \log(\cos(x))\,dx = -\frac{\pi}{2}\log(2)$

(e) $\displaystyle\int_{0}^{\pi/2} \sqrt{\tan(x)}\,dx = \frac{\pi}{\sqrt{2}}$

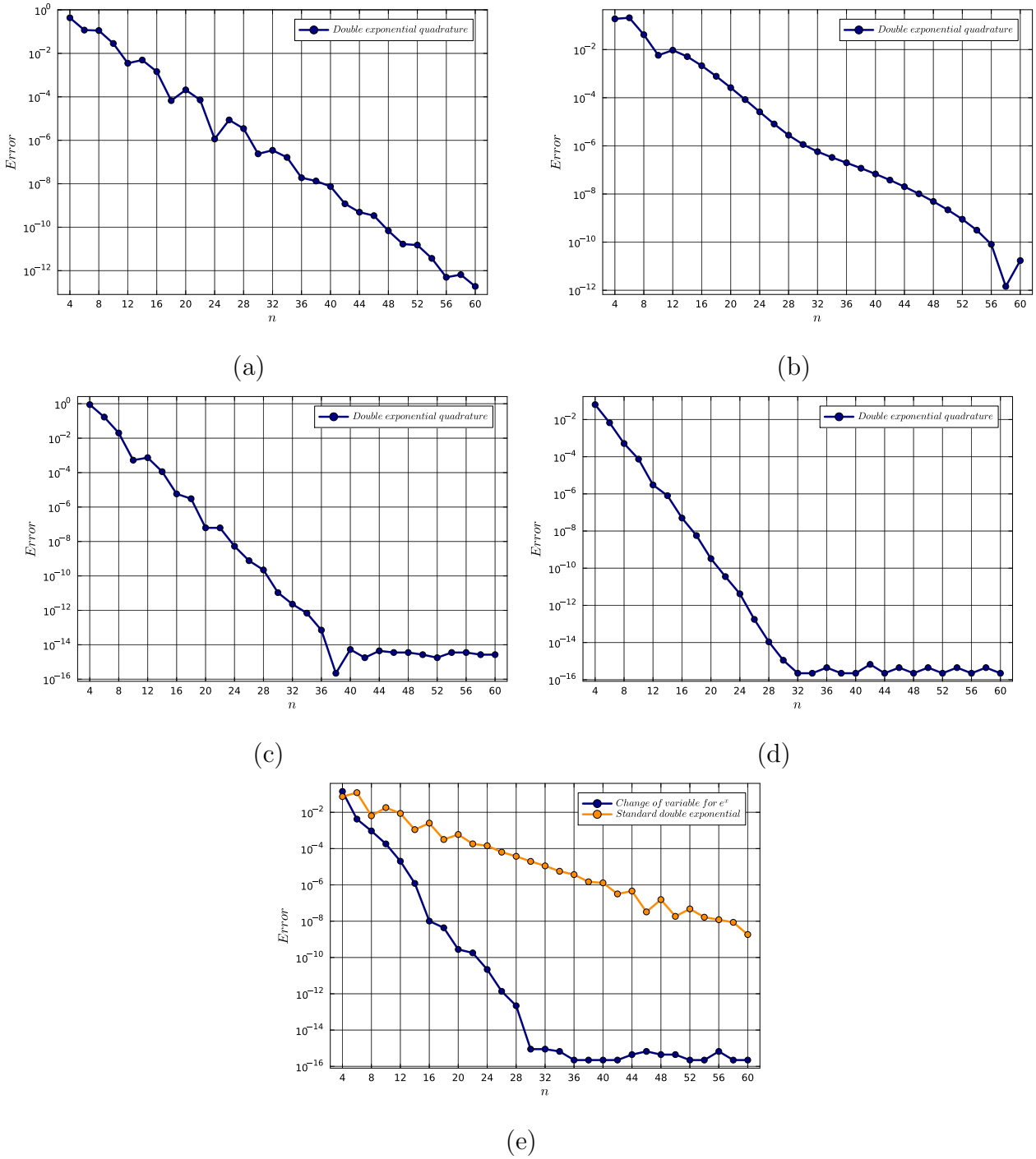


Figure 5.16: Studio degli errori per integrali con la regola double-exponential.

5.6.1 Soluzione

Si riportano in figura 5.17 gli studi degli errori per gli integrali con la regola di quadratura di Gauss-Legendre, insieme agli errori della regola di quadratura double-exponential.

Dai grafici si osserva la rapidità del metodo di quadratura double-exponential rispetto a Gauss-Legendre, infatti tutti gli integrali presentano singolarità al bordo del dominio di integrazione. Tale rapidità è garantita dal fatto che la quadratura double exponential è robusta rispetto all'integrazione su domini singolari, contrariamente all'integrazione di Gauss-Legendre. Lo svantaggio è il costo computazionale elevato a causa del cambio di variabile che richiede un numero maggiore di valutazioni di funzione rispetto all'algoritmo di Gauss-Legendre. Ricordando i risultati degli esercizi precedenti si conclude che Gauss-Legendre è un algoritmo migliore

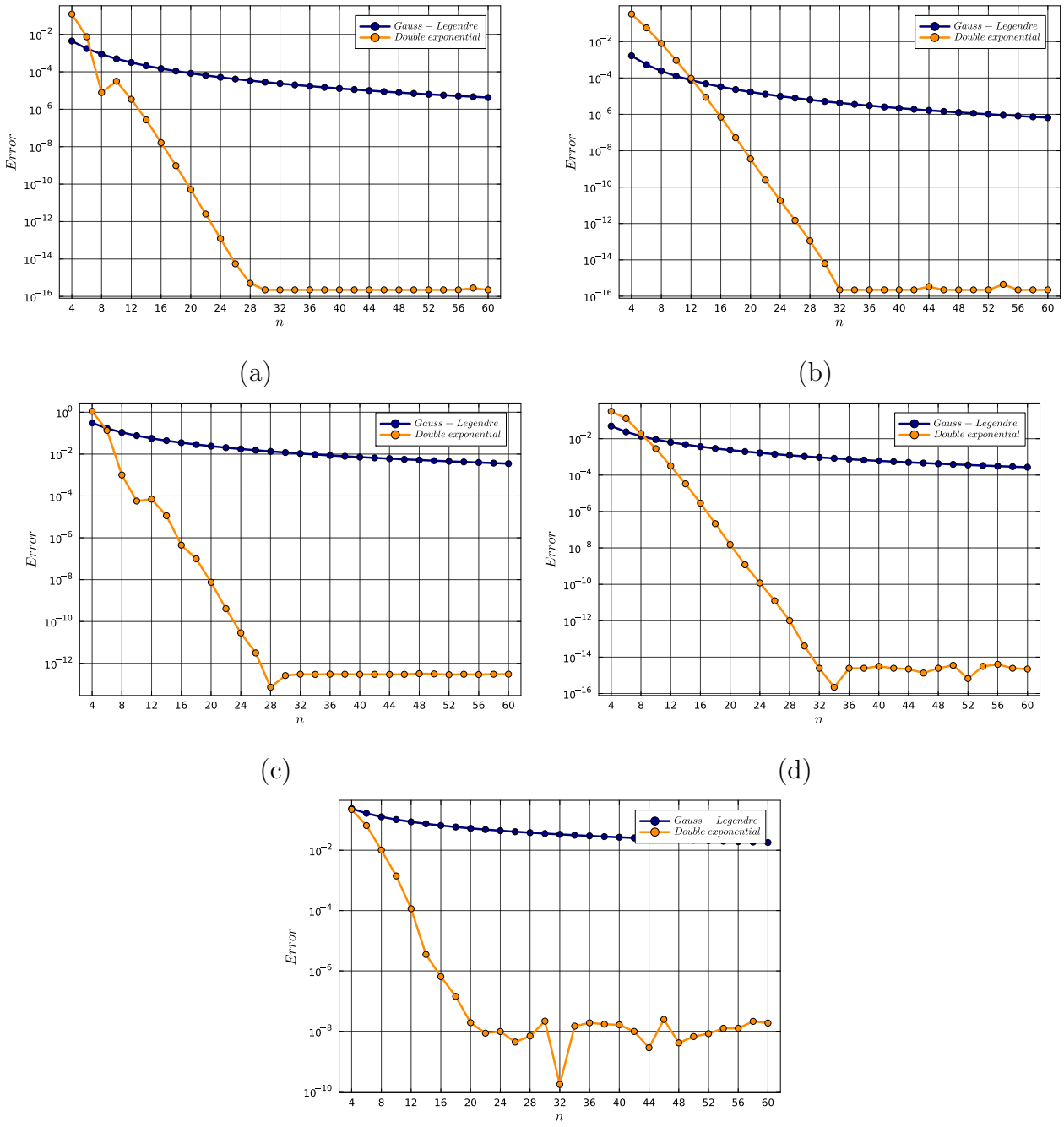


Figure 5.17: Studio degli errori di quadratura Gauss-Legendre e double-exponential.

nel caso di integrande regolari, mentre la quadratura double-exponential è migliore nel caso di singolarità nell'intervallo di integrazione. Clenshaw-Curtis, per come è stato sviluppato nel corso, converge più lentamente di Gauss e si sarebbe tentati di affermare che sia peggiore degli altri due metodi. Ciò non è vero nel momento in cui il calcolo dei pesi di Clenshaw-Curtis viene unito all'algoritmo della *fast Fourier transform*, che abbassa di molto il costo computazionale.

6 Equazioni differenziali ordinarie

6.1 Esercizi 6.2.1, 6.2.2

1. Scrivi un programma che implementi il metodo di Eulero per risolvere un *sistema* di equazioni differenziali.
2. Per ciascun problema ai valori iniziali (IVP), risolvi il problema usando il metodo di Eulero.
(i) Traccia il grafico della soluzione per $n = 320$.
(ii) Per $n = 10 \cdot 2^k$, $k = 2, 3, \ldots, 10$, calcola l'errore come $||u - \hat{u}||_\infty = \max_{0 \le i \le n} |u_i - \hat{u}(t_i)|$ e all'istante finale, $|u_n - \hat{u}(t_n)|$. Realizza un grafico di convergenza log-log, includendo una retta di riferimento per la convergenza del primo ordine.
(a) $u' = -2tu, \;\; 0 \le t \le 2, \;\; u(0) = 2; \qquad \hat{u}(t) = 2e^{-t^2}$
(b) $u' = u + t, \;\; 0 \le t \le 1, \;\; u(0) = 2; \qquad \hat{u}(t) = -1 - t + 3e^t$
(c) $(1 + t^3)uu' = t^2, \;\; 0 \le t \le 3, \;\; u(0) = 1; \qquad \hat{u}(t) = [1 + (2/3)\ln(1 + t^3)]^{1/2}$

6.1.1 Soluzione

Si è implementato il metodo di Eulero per risolvere un sistema di equazioni differenziali. La funzione che sviluppa il metodo viene testata nei prossimi esercizi.
Si riportano nelle figure 6.1, 6.2, 6.3 i grafici delle soluzioni, uniti ai grafici degli errori per i tre casi richiesti. Le soluzioni sono calcolate per $n = 320$ come richiesto, i grafici degli errori corrispondono a più soluzioni con n che varia da $10 \cdot 2^k$ con $k = 2, 3, \ldots, 10$.

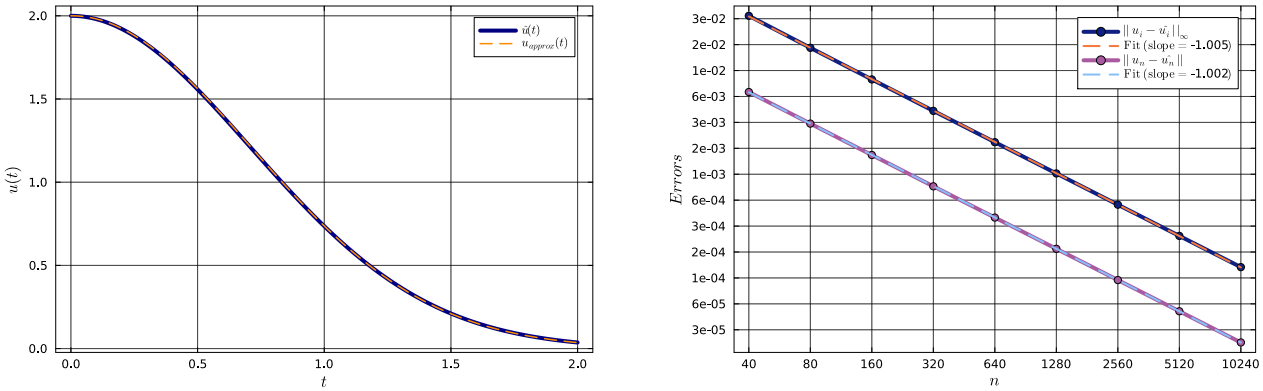


Figure 6.1: Soluzione e errore per il caso (a).

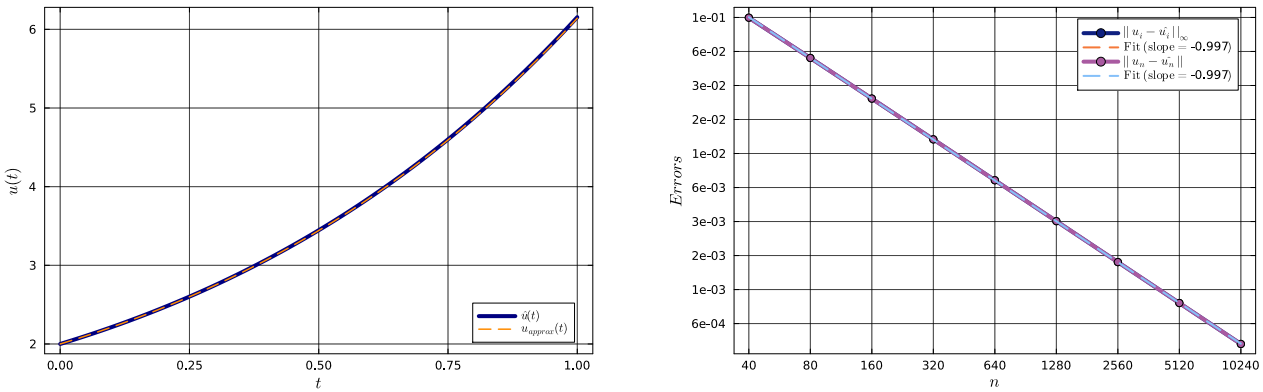


Figure 6.2: Soluzione e errore per il caso (b).

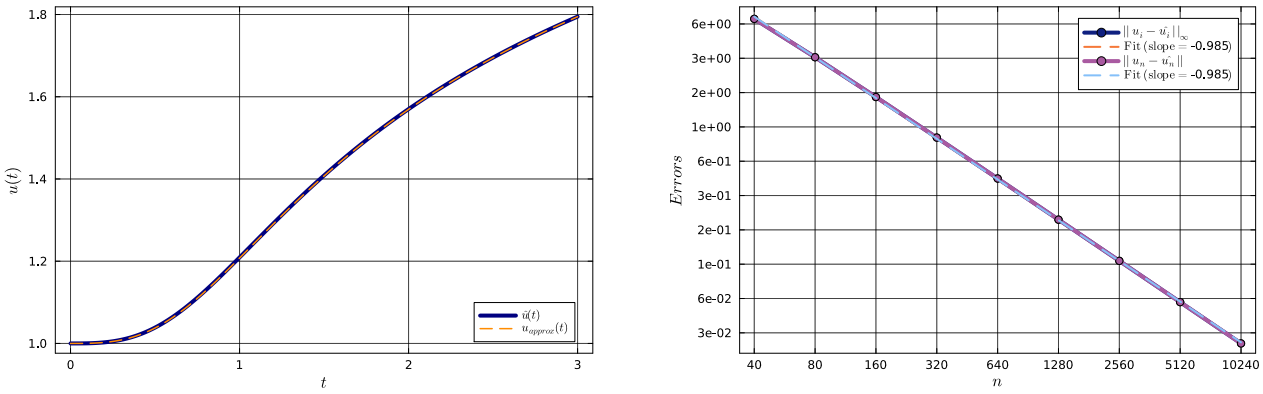


Figure 6.3: Soluzione e errore per il caso (c).

La prima osservazione che si può fare riguarda la convergenza del metodo di Eulero. Gli errori decrescono come $O(h)$, come atteso. Nei grafici infatti è stato riportato il coefficiente angolare del fit degli errori con il metodo dei minimi quadrati. In tutti e tre i casi, per ciascuna tipologia di errore, il coefficiente angolare è prossimo a -1.
La seconda osservazione riguarda la differenza tra i due tipi di errore. Si osserva che solo nel primo grafico le due rette degli errori sono distinte, mentre negli altri due casi sono sovrapposte. È lo stesso comportamento che si è studiato nella teoria, in cui perturbazioni del valore iniziale influenzano la soluzione. L'esempio riguardava le due equazioni $u' = u$ e $u' = -u$, in cui la soluzione della prima equazione amplifica gli errori, mentre la soluzione della seconda equazione li smorza. Nel nostro caso, la soluzione dell'equazione (a) smorza gli errori avanzando nel tempo, il massimo errore non si raggiunge all'istante finale e quindi le rette sono distinte. Negli altri due casi la soluzione amplifica gli errori, il massimo si raggiunge all'istante finale e le rette risultano sovrapposte.

6.2 Esercizio 6.2.3

Risolvi i seguenti problemi ai valori iniziali (IVP) con il metodo di Eulero utilizzando $n = 1000$ passi. Traccia in un unico grafico la soluzione e la sua derivata prima, e in un altro grafico rappresenta l'errore in ciascuna componente in funzione del tempo.

(a) $y'' + 9y = \sin(2t), \quad 0 < t < 2\pi, \quad y(0) = 2, \; y'(0) = 1;$
$\hat{y}(t) = \frac{1}{5}\sin(3t) + 2\cos(3t) + \frac{1}{5}\sin(2t)$

(b) $y'' - 4y = 4t, \quad 0 < t < 1.5, \quad y(0) = 2, \; y'(0) = -1;$
$\hat{y}(t) = e^{2t} + e^{-2t} - t$

(c) $y'' + 4y' + 4y = t, \quad 0 < t < 4, \quad y(0) = 1, \; y'(0) = \frac{3}{4};$
$\hat{y}(t) = (3t + \frac{5}{4})e^{-2t} + \frac{t-1}{4}$

6.2.1 Soluzione

Fissato $n = 1000$ si è risolto il problema ai valori iniziali (IVP) con il metodo di Eulero. Per prima cosa si sono riscritte le equazioni differenziali di secondo ordine come un sistema di

equazioni differenziali di primo ordine, in modo da poter utilizzare il metodo di Eulero:

$$\begin{cases} z_1' &= z_2 \\ z_2' &= -9z_1 + \sin(2t) \\ z_1(0) &= 2 \\ z_2(0) &= 1 \end{cases} \qquad \begin{cases} z_1' &= z_2 \\ z_2' &= 4z_1 + 4t \\ z_1(0) &= 2 \\ z_2(0) &= -1 \end{cases} \qquad \begin{cases} z_1' &= z_2 \\ z_2' &= -4z_1 - 4z_2 + t \\ z_1(0) &= 1 \\ z_2(0) &= \dfrac{3}{4} \end{cases}$$

Si riportano in figura 6.4, 6.5, 6.6 i grafici delle soluzioni e delle loro derivate prime, insieme ai grafici degli errori per i tre casi richiesti. Gli errori rappresentano la distanza tra la soluzione calcolata e quella esatta, in ciascuna componente del sistema.

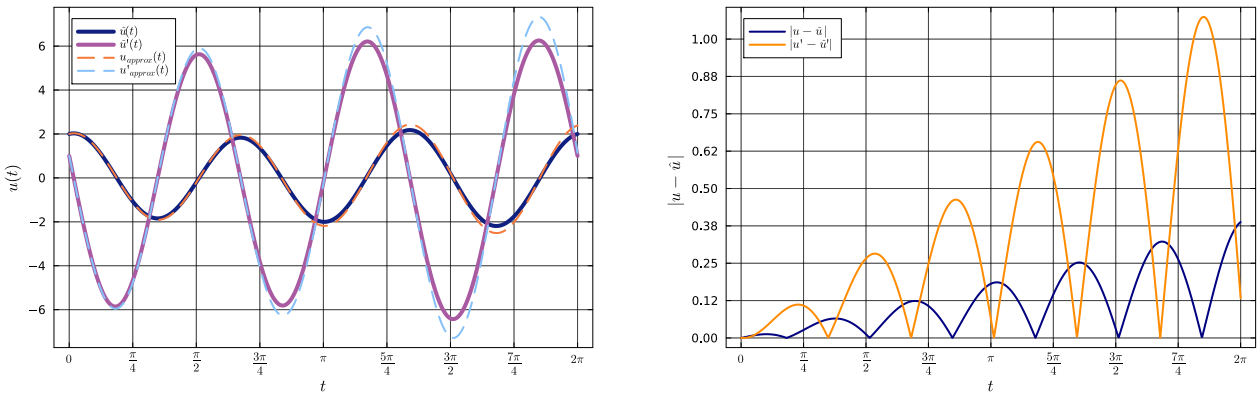


Figure 6.4: Soluzione e derivata prima per il caso (a). Errori.

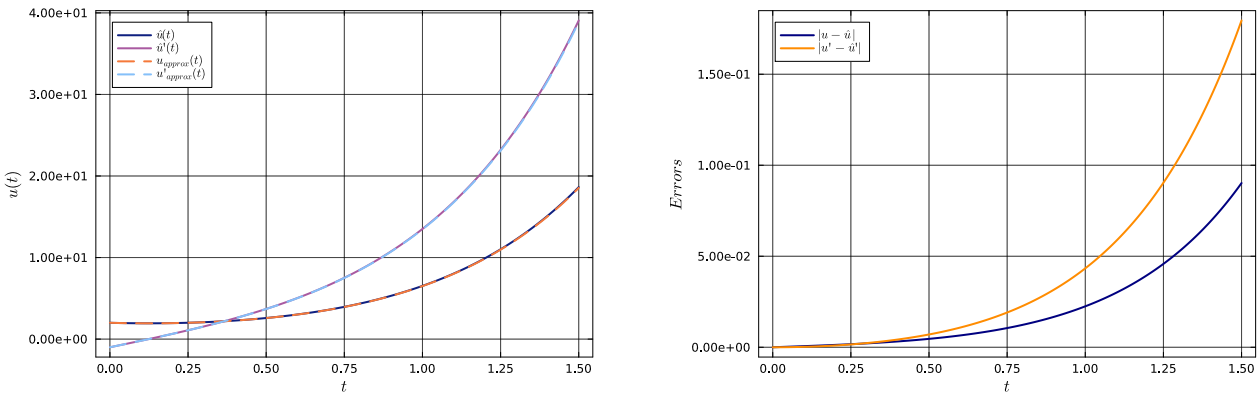


Figure 6.5: Soluzione e derivata prima per il caso (b). Errori.

Nelle figure 6.4 e 6.5 è possibile notare che gli errori aumentano insieme al tempo. Infatti nel caso (a) la soluzione esatta è una combinazione di funzioni trigonometriche, per cui le perturbazioni iniziali si amplificano nel tempo. Nel caso (b) si ha lo stesso comportamento perchè a tempi grandi domina l'esponenziale e^{2t}, amplificando le perturbazioni iniziali.
Al contrario in figura 6.6 si osservano gli errori decrescere nel tempo perchè la soluzione esponenziale decrescente smorza l'errore.

6.3 Esercizio 6.3.1

(a) Scrivi un programma che risolva un sistema di equazioni differenziali utilizzando il metodo IE2 (Eulero implicito esplicito di ordine 2).
(b) Scrivi un programma che risolva un sistema di equazioni differenziali utilizzando il metodo

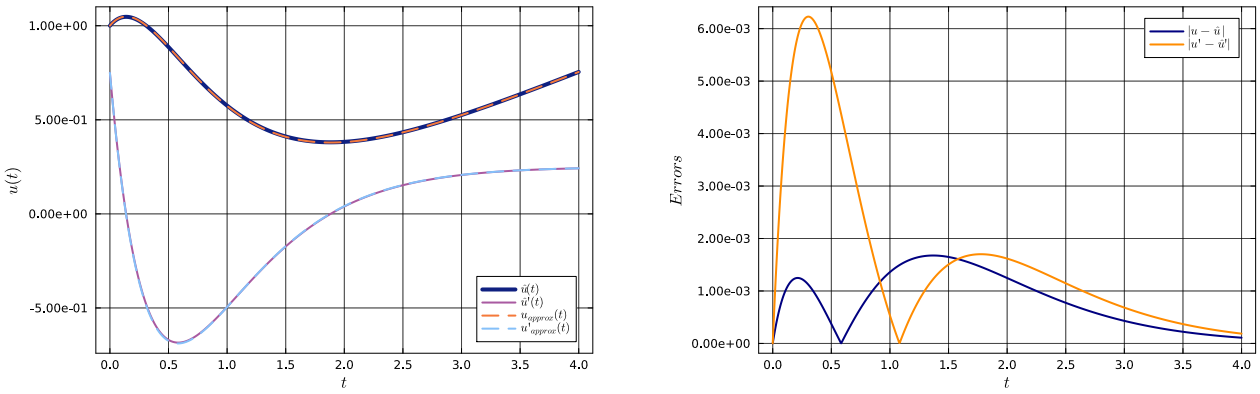


Figure 6.6: Soluzione e derivata prima per il caso (c). Errori.

RK4 (Runge-Kutta del quarto ordine).
(c) Testa la tua implementazione dei metodi IE2 e RK4 sul problema ai valori iniziali

$$u' = -2tu, \;\; 0 \leq t \leq 2, \;\; u(0) = 2; \qquad \hat{u}(t) = 2e^{-t^2}$$

Risolvi per $n = 30, 60, 90, \ldots, 300$ e rappresenta graficamente l'errore massimo $||u - \hat{u}||_\infty = \max_{0 \leq i \leq n} |u_i - \hat{u}(t_i)|$ in funzione del numero di valutazioni di funzione in un grafico log-log, insieme a una retta che mostri il tasso di convergenza atteso.

6.3.1 Soluzione

Si sono implementati i due metodi richiesti e applicati al problema ai valori iniziali (IVP) presentato. In figura 6.7 si riportano i grafici degli errori per i due metodi, calcolati per $n = 30, 60, 90, \ldots, 300$. Nel grafico sono state tracciate le rette interpolanti degli errori, ottenute

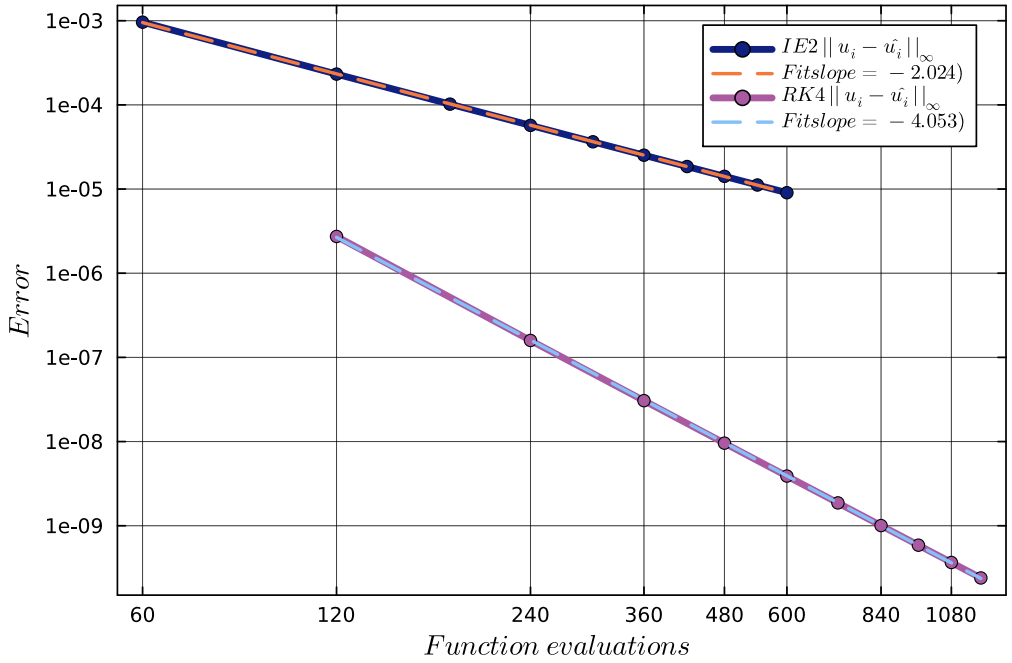


Figure 6.7: Confronto tra gli errori dei metodi IE2 e RK4.

con il metodo dei minimi quadrati. Di tali rette sono riportate sul grafico le pendenze. Il metodo IE2 ha ordine di convergenza pari a 2, il metodo RK4 ha ordine di convergenza pari a 4. Tali

valori teorici sono in accordo con le pendenze trovate.
I grafici sono stati realizzati in funzione del numero di valutazioni della funzione. Sapendo che IE2 richiede 2 valutazioni della funzione per ogni passo, e che RK4 richiede 4 valutazioni della funzione per ogni passo, si può spiegare il fatto che i punti del grafico si trovano incolonnati uno ogni due passi di IE2.
Infine, a parità di numero di valutazioni della funzione il metodo RK4 ha un errore molto più piccolo rispetto al metodo IE2, evidenziandone l'efficacia.

6.4 Esercizio 6.3.2

In ciascuno dei seguenti casi, utilizza l'integratore RK4 per risolvere il sistema di ODE per $0 \leq t \leq 10$ con le condizioni iniziali indicate. (A tal fine, devi trovare una procedura sensata per scegliere il numero di passi n da utilizzare.) Rappresenta i risultati come curve nel piano delle fasi, cioè con x e y come assi del grafico.

$$\text{(a)} \quad \begin{cases} x' = -4y + x(1 - x^2 - y^2) \\ y' = 4x + y(1 - x^2 - y^2) \end{cases} \qquad [x(0),\, y(0)] = [0.1,\, 0], \quad [0,\, 1.9]$$

$$\text{(b)} \quad \begin{cases} x' = -4y - \dfrac{1}{4}x(1 - x^2 - y^2)(4 - x^2 - y^2) \\ y' = 4x - \dfrac{1}{4}y(1 - x^2 - y^2)(4 - x^2 - y^2) \end{cases} \qquad [x(0),\, y(0)] = [0.95,\, 0], [0,\, 1.05], [-2.5,\, 0]$$

6.4.1 Soluzione

Si sono risolti i due sistemi di equazioni differenziali con il metodo RK4, per ciascuna delle condizioni iniziali indicate. Al fine di stimare un valore per il passo h, si è scelto di utilizzare il seguente algoritmo:

1. Si sceglie un numero di passi iniziale n.
2. Si calcolano le soluzioni per n e $2n$ passi, ottenendo rispettivamente u e $\tilde{u}$.
3. Si calcola l'errore come $||u - \tilde{u}||_\infty$, scegliendo per $\tilde{u}$ punti alterni.
4. Se l'errore è maggiore di una tolleranza ε, si raddoppia il passo e si torna al punto 2, altrimenti l'algoritmo si interrompe, restituisce n e $h = \frac{b-a}{n}$.

Per tutte le condizioni iniziali si è scelto come tolleranza $\varepsilon = 10^{-6}$, e come numero di punti iniziali $n = 1000$.
Da notare che nell'algoritmo presentato ad ogni passo bisogna calcolare solamente una nuova soluzione perchè si riutilizza quella precedente. Con l'algoritmo proposto si riesce a stimare un valore di h che garantisca un errore inferiore alla tolleranza, ma non è certamente paragonabile al metodo adattivo, che a ogni passaggio permette di scegliere il passo in maniera efficiente.
Si riportano in figura 6.8 e 6.9 i grafici delle soluzioni ottenute, rappresentate come curve nel piano delle fasi. Si sono rappresentate inoltre le soluzioni in tre dimensioni.

Infine si riportano in tabella 6.1 i valori di h ottenuti tramite il metodo precedentemente descritto.

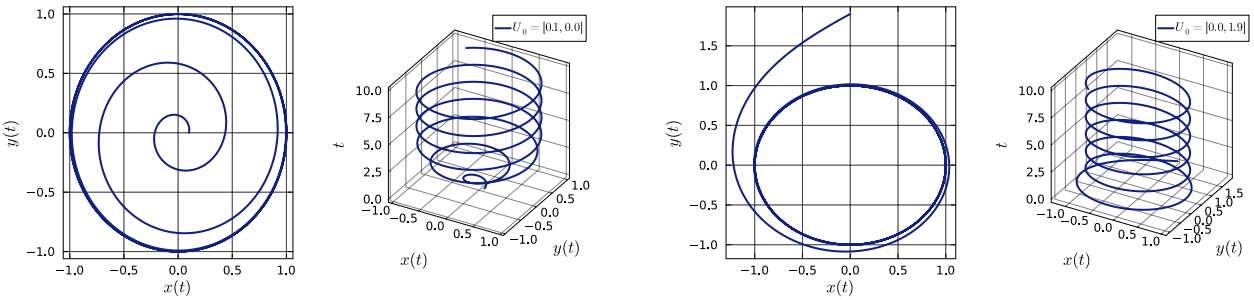


Figure 6.8: Soluzioni del sistema di ODE (a).

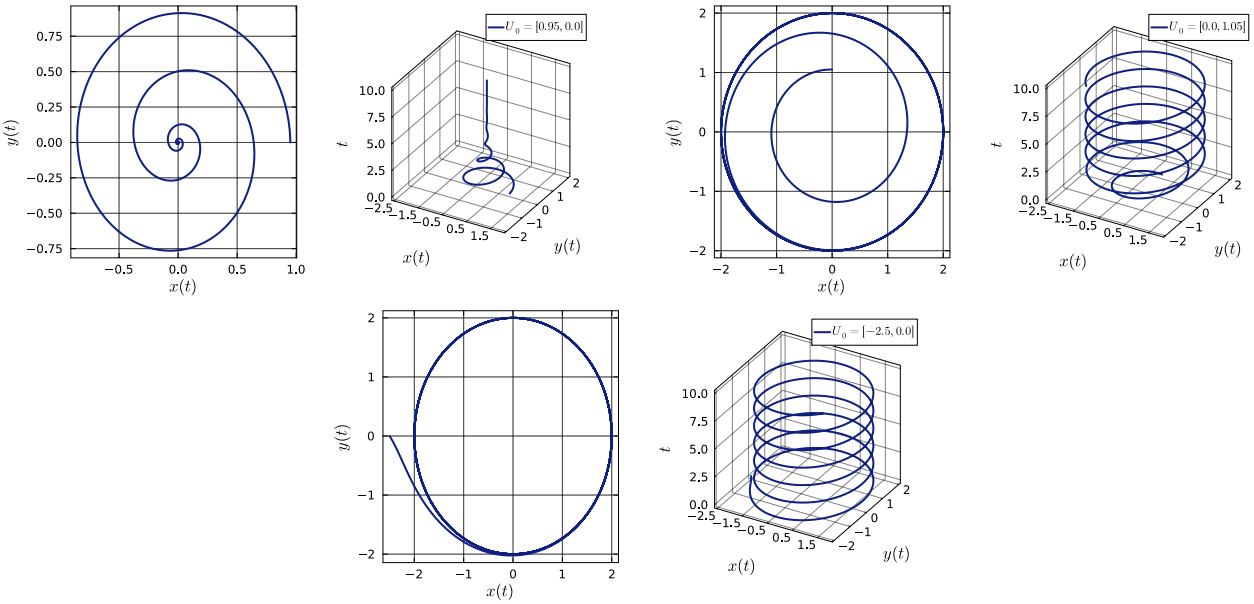


Figure 6.9: Soluzioni del sistema di ODE (b).

Table 6.1: Valori di h ottenuti per i due sistemi di ODE

Sistema	**Condizioni iniziali**	h
a	$[0.1, 0]$	0.0025
	$[0, 1.9]$	0.0025
b	$[0.95, 0]$	0.0025
	$[0, 1.05]$	0.00125
	$[-2.5, 0]$	0.00125

6.5 Esercizio 6.3.3

Una malattia endemica in una popolazione può essere modellata tracciando la frazione di popolazione suscettibile all'infezione, $v(t)$, e la frazione infetta, $w(t)$. (Il resto della popolazione si considera guarito e immune.) Un modello tipico è il *modello SIR*:

$$\frac{dv}{dt} = 0.2(1 - v) - 3vw\,, \qquad \frac{dw}{dt} = (3v - 1)w\,. \tag{6.1}$$

Partendo da $v(0) = 0.95$ e $w(0) = 0.05$, utilizza il metodo RK4 per trovare i valori stazionari a lungo termine di $v(t)$ e $w(t)$. Rappresenta graficamente entrambe le componenti della soluzione in funzione del tempo.

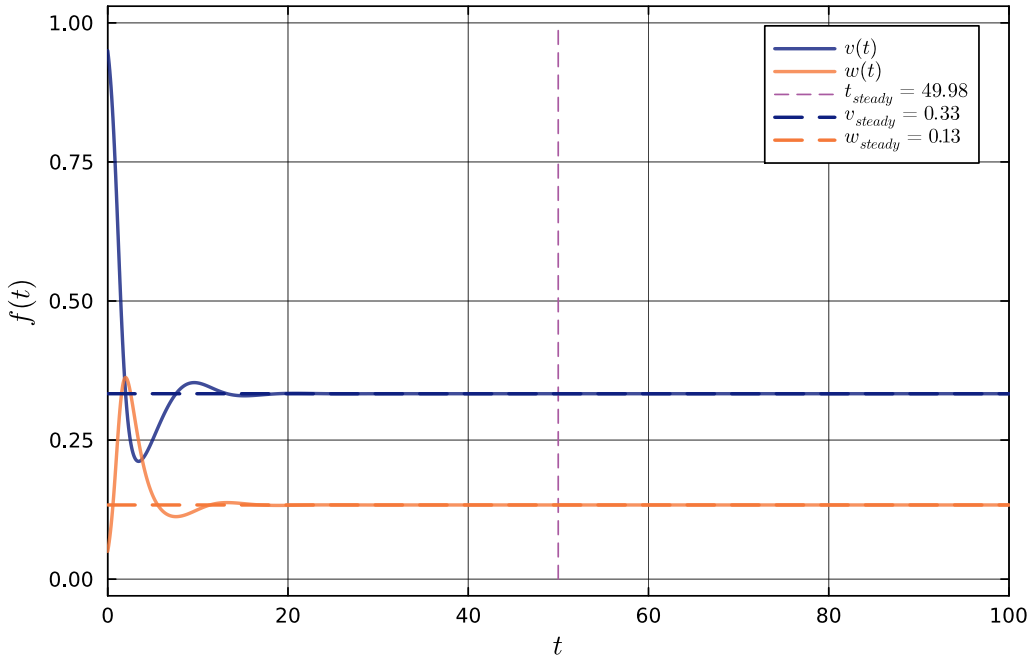


Figure 6.10: Soluzione del modello SIR.

6.5.1 Soluzione

Si è risolto il sistema di equazioni differenziali con il metodo RK4, partendo dalle condizioni iniziali $v(0) = 0.95$ e $w(0) = 0.05$. Il passo h è stato stimato allo stesso modo dell'esercizio 6.4, in particolare si è ottenuto $h = 0.015625$ con un numero di step iniziali $n = 100$. Si riporta in figura 6.10 il grafico della soluzione, in cui si osserva che le due componenti convergono a un valore stazionario.

Per valutare i valori stazionari si sono calcolate le differenze tra i valori successivi di v e w, fino a che non si è raggiunta la tolleranza di 10^{-9}. In corrispondenza di tale tolleranza si sono ottenuti due valori di tempo, uno per ciascuna componente della soluzione, e se ne è scelto il massimo, t_{steady}. È importante notare che il valore di t_{steady} potrebbe essere una sovrastima, infatti la valutazione della distanza tra due valori successivi è un metodo ingenuo perchè non tiene conto del cambiamento globale della soluzione. Per quanto riguarda il caso in esame, poco importa: se anche il metodo adottato portasse a una sovrastima di t_{steady} ciò non influenzerebbe la valutazione degli stati stazionari perchè è noto che le due componenti convergono a un valore stazionario.
I valori stazionari ottenuti sono riportati in grafico 6.10, calcolati come $v(t_{steady})$ e $w(t_{steady})$.

6.6 Esercizio 6.3.5

(a) Scrivi un programma che risolva un sistema di equazioni differenziali utilizzando il metodo adattivo BS23.

(b) Testa la tua implementazione risolvendo

$$u' = -2tu, \quad 0 \leq t \leq 2, \quad u(0) = 2; \qquad \hat{u}(t) = 2e^{-t^2}$$

Rappresenta graficamente la soluzione in funzione del tempo per $\delta = 10^{-8}$ in un grafico e il valore del passo in un altro. Determina inoltre il passo minimo e medio, escludendo il primo e l'ultimo passo.

(c) Considera $\delta = 10^{-6}, 10^{-5}, \ldots, 10^{-12}$ e studia come l'errore sulla soluzione all'istante finale, $|u_n - \hat{u}(t_n)|$, dipende dalla tolleranza in input δ.

6.6.1 Soluzione

Si è implementato il metodo adattivo BS23 per risolvere un sistema di equazioni differenziali, e si riporta in figura 6.11 il grafico della soluzione ottenuta, insieme al grafico del passo h in funzione del tempo.

Si è scelto di togliere il primo e l'ultimo passo perchè il primo è scelto dall'utente e l'ultimo

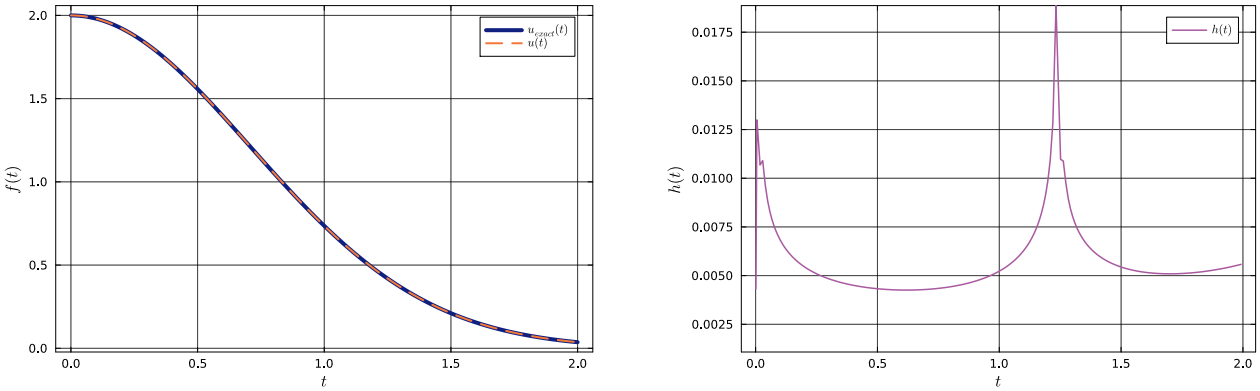


Figure 6.11: Soluzione e passo del metodo BS23.

è forzato in modo da raggiungere l'estremo b. Si ottengono un passo minimo di $h_{min} = 0.0043$ e un passo medio di $h_{med} = 0.0054$. Lo studio del passo è interessante perchè permette di capire di quanto si è guadagnato rispetto a un metodo non adattivo. Se si fosse scelto passo fisso pari a h_{med} si sarebbe persa risoluzione perchè nel metodo adattivo si può raggiungere un passo minore. Nel caso in esame il guadagno non è molto significativo, in generale però si possono ottenere integrazioni migliori se la differenza fra il passo minimo e il passo medio è molto grande.

In figura 6.12 si riporta il grafico dell'errore all'istante finale, $|u_n - \hat{u}(t_n)|$, in funzione della tolleranza δ scelta. Si osserva che l'errore cresce all'aumentare della tolleranza. Infatti la media dei passi h cresce all'aumentare della tolleranza, e la soluzione viene integrata da un numero minore di punti.

6.7 Esercizio 6.3.6

Utilizzando il metodo BS23 con tolleranza sull'errore $\delta = 10^{-8}$, risolvi il problema $y'' + (1 + y')^3 y = 0$ nell'intervallo $0 \leq t \leq 4\pi$ per le seguenti condizioni iniziali. Rappresenta graficamente $y(t)$ e $y'(t)$ in funzione di t e, separatamente, il passo h in funzione di t. Determina inoltre il passo minimo e medio, escludendo il primo e l'ultimo valore.

- $y(0) = 0.1, \quad y'(0) = 0$
- $y(0) = 0.5, \quad y'(0) = 0$
- $y(0) = 0.75, \quad y'(0) = 0$
- $y(0) = 0.95, \quad y'(0) = 0$

6.7.1 Soluzione

Il sistema associato all'equazione differenziale in esame è

$$\begin{cases} z_1' &= z_2 \\ z_2' &= -(1 + z_2)^3 z_1 \end{cases} \qquad \text{con } z_1 = y, \quad z_2 = y'$$

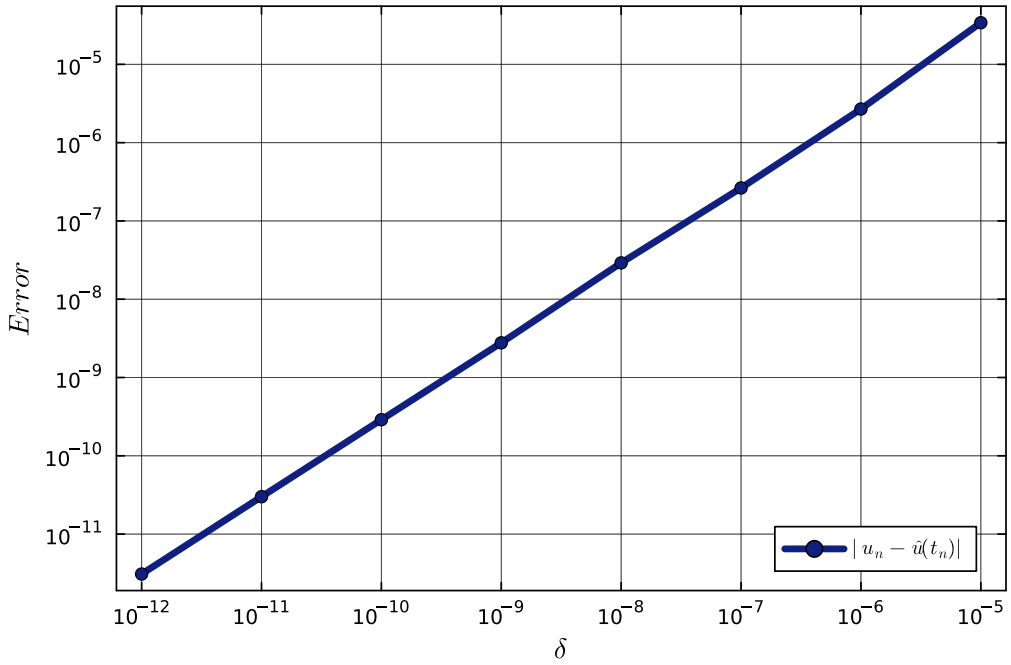


Figure 6.12: Errore all'istante finale in funzione della tolleranza δ.

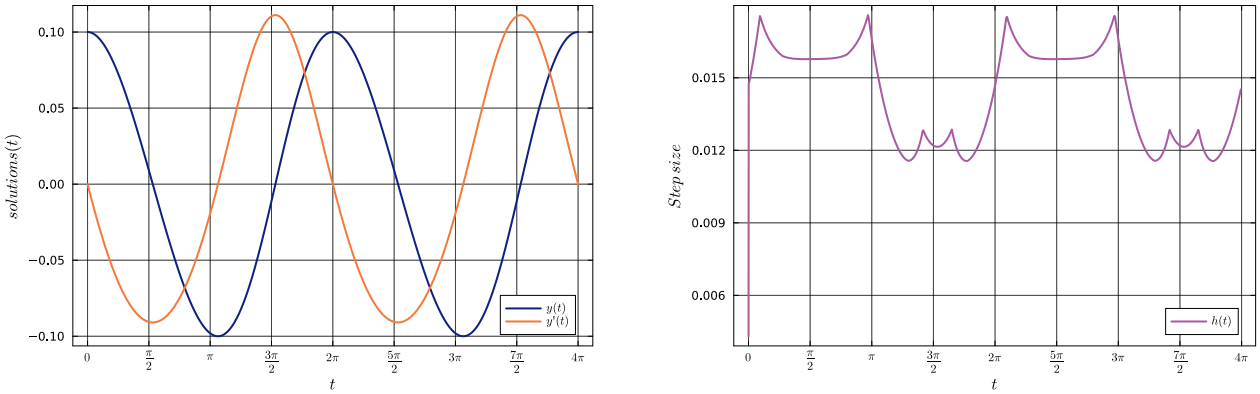


Figure 6.13: Soluzioni per il caso $y(0) = 0.1$, $y'(0) = 0$. Passo del metodo BS23.

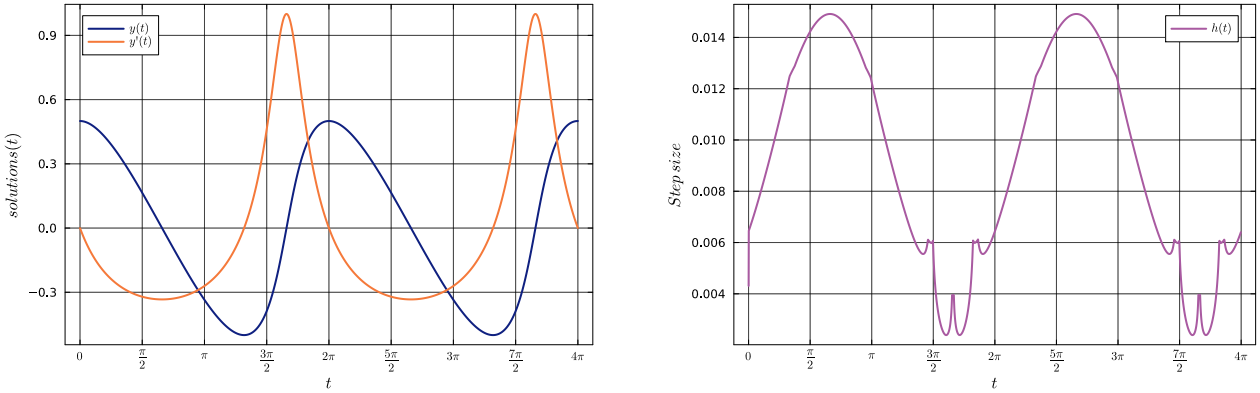


Figure 6.14: Soluzioni per il caso $y(0) = 0.5$, $y'(0) = 0$. Passo del metodo BS23.

Nelle figure 6.13, 6.14, 6.15, 6.16 sono riportati $y(t)$, $y'(t)$ e $h(t)$.

Nei vari casi si osserva che, tanto più la soluzione presenta picchi accentuati, tanto più il metodo BS23 sceglie un passo piccolo. In tabella 6.2 sono riportati i valori di passo minimo e

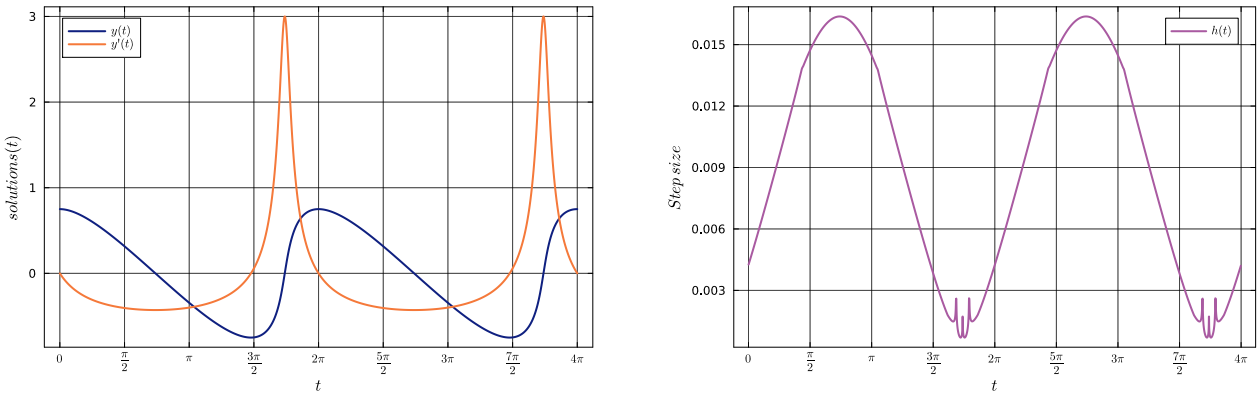


Figure 6.15: Soluzioni per il caso $y(0) = 0.75$, $y'(0) = 0$. Passo del metodo BS23.

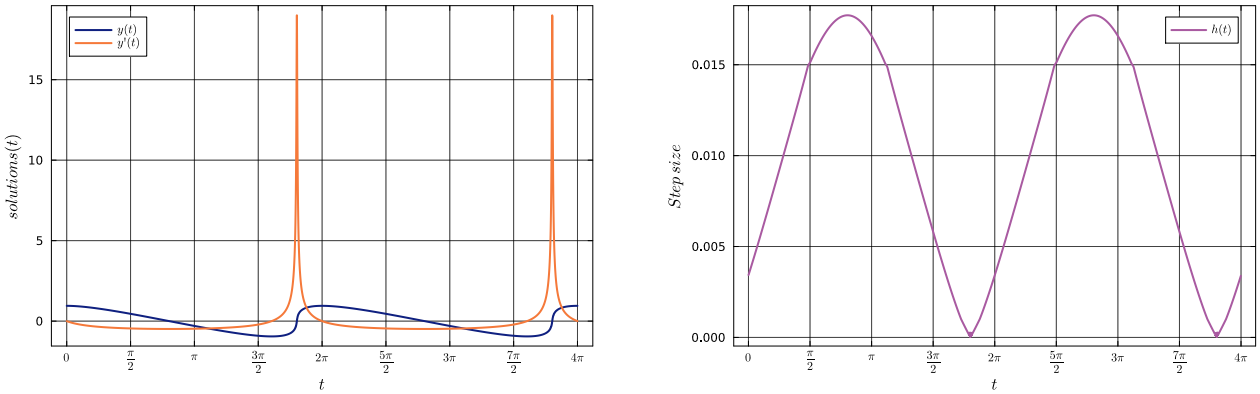


Figure 6.16: Soluzioni per il caso $y(0) = 0.95$, $y'(0) = 0$. Passo del metodo BS23.

medio. Si rimanda alla sezione 6.6 la dicussione sull'importanza dello studio del passo.

Table 6.2: Valori di passo minimo e medio per i quattro casi

Caso	h_{min}	h_{med}
a	4.3×10^{-3}	1.41×10^{-2}
b	2.4×10^{-3}	6.9×10^{-3}
c	6.97×10^{-4}	4.48×10^{-3}
d	5.56×10^{-5}	2.49×10^{-3}

La tabella 6.2 mostra che il guadagno in risoluzione rispetto a un metodo non adattivo è molto significativo negli ultimi due casi, dove le soluzioni presentano picchi molto accentuati, perchè il metodo BS23, nonostante abbia passo medio dell'ordine di 10^{-3}, raggiunge passi minimi dell'ordine di 10^{-4} e 10^{-5}.

6.8 Esercizio 6.3.7

Risolvi il problema $u' = 100u^2 - u^3$, $u(0) = 0.0002$, $0 \leq t \leq 100$ utilizzando il metodo BS23 e realizza grafici che mostrino sia la soluzione sia i passi temporali scelti per $\delta = 10^{-8}$. La soluzione effettua una rapida transizione tra due stati quasi costanti. Il passo scelto dall'algoritmo si comporta allo stesso modo in entrambi gli stati?

6.8.1 Soluzione

Si è risolto il problema ai valori iniziali (IVP) con il metodo BS23. Si riporta in figura 6.17 il grafico della soluzione ottenuta, insieme al grafico del passo h in funzione del tempo.

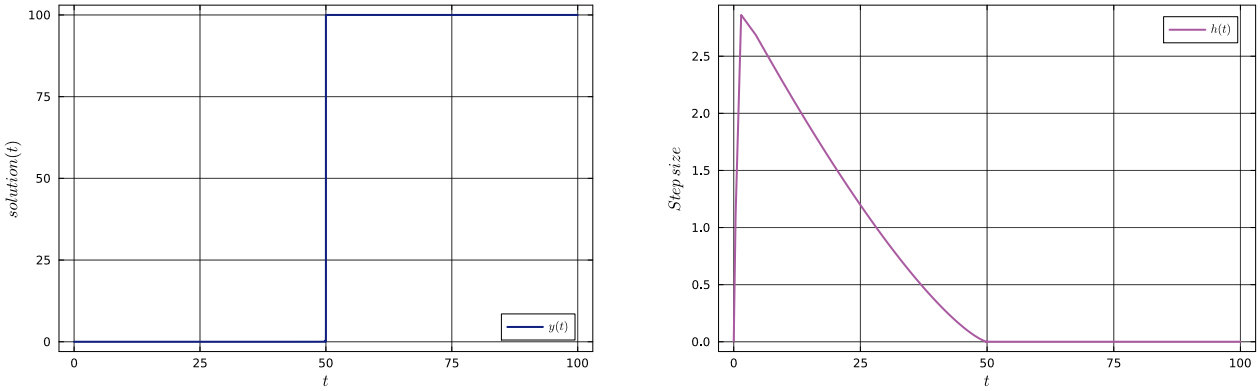


Figure 6.17: Soluzione e passo del metodo BS23.

L'integrazione avviene con successo, ma il grafico del passo mostra un comportamento peculiare. Ci si potrebbe aspettare che, data la presenza di due zone analoghe per la soluzione, il passo scelto dall'algoritmo sia simile in entrambe le zone, con una fase transiente in cui il passo diminuisce rapidamente. Questo non è il caso della figura 6.17 a destra.
Per studiare meglio il comportamento del passo, si è scelto di calcolarne il minimo e il medio, dividendo le due regioni della soluzione. La prima regione comincia da $t = 0$ e termina a $t = 51$, dove il valore finale si è scelto in modo da includere la fase transiente, e la seconda regione comincia da $t = 51$ e termina a $t = 100$. In tabella 6.3 sono riportati i valori di passo minimo e medio per le due regioni.

Giustamente si osserva il minimo nella prima regione, perchè il minimo valore di h viene

Table 6.3: Valori di passo minimo e medio per il problema stiff

Regione	h_{min}	h_{med}
Prima regione	1.6×10^{-6}	0.00845
Seconda regione	0.00025	0.00025

assunto in corrispondenza del salto della soluzione, in cui il metodo adattivo cerca passi molto piccoli per ottenere una soluzione accurata. Ciò che è particolare è che nella seconda regione il passo si stabilizzi, rimanendo costante e molto piccolo. Tale comportamento rende l'equazione presa in esame appartenente alla categoria dei cosiddetti *stiff problems*.

7 Appendice

Table 7.1: Distanza dal Sole e periodo orbitale dei pianeti del Sistema Solare

Pianeta	**Distanza dal Sole [Mkm]**	**Periodo orbitale [giorni]**
Mercurio	57.59	87.99
Venere	108.11	224.7
Terra	149.57	365.26
Marte	227.84	686.98
Giove	778.14	4332.4
Saturno	1427	10759
Urano	2870.3	30684
Nettuno	4499.9	60188