Laboratorio di Programmazione e Calcolo
Canale 2
Appunti del corso

Simone Cacace e Giuseppe Visconti
Dipartimento di Matematica
Sapienza Università di Roma

Anno Accademico 2025–2026

Laboratorio – Integrazione numerica

Esercizi

E0. Confronto formule di quadratura

Calcolare
$$I = \int_0^{2\pi} xe^{-x}\cos(2x)\,\mathrm{d}x$$
usando le seguenti regole composite: rettangoli, punto medio, trapezi, Gauss 2-punti, Simpson. Il valore esatto (di riferimento) è
$$I = -0.122122604618968.$$

- Implementa tutte le regole come funzioni che restituiscono il valore approssimato dell'integrale $Q(f)$.
- Scegli una sequenza di M crescenti (ad esempio: $M = 8, 16, 32, 64, 128, 256, 512$).
- Per ogni M calcola l'ampiezza di ogni intervallo $h = (b-a)/M$, $Q(f)$ per ciascun metodo richiamando le funzioni scritte, e l'errore assoluto $|I - Q(f)|$.
- Salva su ogni file `<nome_quadratura>.txt` una tabella con colonne:

  ```
  M h Q(f) |I-Q(f)| p_n
  ```

  dove p_n è l'EOC (ordine sperimentale di convergenza) calcolato confrontando l'errore con quello del M precedente (per il primo M lasciare –).
- Genera plot log–log con `gnuplot`: ascissa M (o $1/h$) e ordinata errore $|I - Q(f)|$. Metti tutte le curve sullo stesso grafico per confronto.
- Genera un secondo plot con `gnuplot`: ascissa M (o $1/h$) e ordinata p_n.
- Commenta i risultati e verifica che l'EOC osservato si avvicini agli ordini teorici:
  - Rettangoli sinistro: ordine 1 (errore $O(h)$).
  - Punto medio: ordine 2 (errore $O(h^2)$).
  - Trapezi: ordine 2 (errore $O(h^2)$).
  - Gauss 2 punti: ordine 4 (errore $O(h^4)$).
  - Simpson: ordine 4 (errore $O(h^4)$).

Extra. Potete anche confrontare i metodi contando il numero di valutazioni di f (per esempio incrementando una variabile globale): è utile per confrontare efficienza. Nota: Gauss a 2 punti e Simpson hanno stesso ordine di convergenza. Però, Gauss a 2 punti richiede 2 valutazioni per sottointervallo; Simpson usa un valore in comune tra intervalli (ottimizza per non ricalcolare).

Esempio di script gnuplot. Salva in `plot_E0.gp`:

```
set term pngcairo size 1000,700
set output 'E0_errors.png'
set logscale xy
set xlabel 'M'
set ylabel 'errore |I-Q|'
plot 'rettangoli.txt' using 1:4 with linespoints title 'rect', \
'puntomedio.txt' using 1:4 with linespoints title 'mid', \
'trapezi.txt' using 1:4 with linespoints title 'trap', \
'gauss.txt' using 1:4 with linespoints title 'gauss2', \
'simpson.txt' using 1:4 with linespoints title 'simpson'
```

Esempio di output atteso e verifica rapida. Ci si aspetta che, raddoppiando M, l'errore per Simpson scenda di circa $2^4 = 16$ volte ($p \approx 4$) nella regione asintotica; per trap e midpoint di circa $2^2 = 4$ volte ($p \approx 2$); per rettangoli di circa $2^1 = 2$ volte ($p \approx 1$). Gauss2 composito si comporta come Simpson ($p \approx 4$) per funzioni abbastanza lisce.

E1. Stima di π.

Stimare π utilizzando

$$\pi = \int_{-1}^{1} 2\sqrt{1-x^2}\,\mathrm{d}x$$

con:

- Trapezi composito per $M = 8, 16, 32, 64, 128, 256, 512$.
- Metodo Monte Carlo: per ogni M lancia $N = M \cdot 100$ punti casuali all'interno del rettangolo $[-1, 1] \times [0, f_{\max}]$, $f_{\max} = \max_{x \in [-1,1]} f(x) = 2$, di area $A_R = 4$.
  Conta quanti punti casuali cadono sotto la semicirconferenza. Calcola la stima di π come $\hat{\pi} = 4$`(area_stimata)`.

Suggerimenti di implementazione:

1. Per il trapezio procedere come in E0 e salvare `E1_trap.txt` con colonne

   ```
   M h Q(f) e_M
   ```

   dove $e_M = |\pi - \hat{\pi}|$ è l'errore commesso con M sottointervalli.
2. Per Monte Carlo, eseguire più repliche (ad esempio: 10 repliche) per ogni N e salvare la media delle stime. Salvare `E1_mc.txt` con colonne N, media, errore assoluto medio.
3. Confronta accuratezza: plot log–log di M vs. errore per i due metodi.

Osservazione. Monte Carlo ha convergenza lenta (tipicamente $O(N^{-1/2})$ in media), ma è semplice e scala bene in dimensioni maggiori.

Esempio di output atteso e verifica rapida. Trapezi: convergenza veloce rispetto a Monte Carlo per lo stesso numero di valutazioni (in 1D). Monte Carlo mostrerà oscillazioni: la deviazione tipica scala come $1/\sqrt{N}$.

EOC – ordine sperimentale di convergenza

Sia e_n l'errore per la discretizzazione con passo h_n (ampiezza di ogni sottointervallo scegliendo n sottointervalli). Se il metodo è di ordine p, ovvero asintoticamente del tipo $e_n \approx Ch_n^p$, allora usando due step successivi si stima p come

$$p \approx \frac{\log_2(e_{n+1}/e_n)}{\log_2(h_{n+1}/h_n)}.$$

Nel caso in cui si raddoppi il numero di sottointervalli (cioè $h_{n+1} = h_n/2$), diventa

$$p_n \; = \; \frac{\log_2(e_{n+1}/e_n)}{\log_2(1/2)} \; = \; -\log(e_{n+1}/e_n).$$

Quindi, nella pratica, per ogni coppia consecutiva di M si calcola p_n con la formula

$$p_n = \log_2(e_n/e_{n-1}).$$