

Misure e gra! ci: Python al pos to della carta millimetrata



OBIE TIVO Disegnare un gra! co sperimentale che mostra le barre di errore sui dati, da allegare in una relazione di laboratorio.

LAFISICA COINVOLTA

Consideriamo l'esperimento del pendolo semplice, nel quale un oggetto è appeso a un'estremità di un filo che abbia l'altra estremità fissata. Il filo ha una massa trascurabile rispetto all'oggetto, che è libero di oscillare avanti e indietro.

Le grandezze fisiche rilevanti per l'esperimento sono:

la lunghezza l del filo;

il periodo di oscillazione T , cioè la durata completa di un'oscillazione.

Misuriamo con un cronometro il periodo del pendolo per diversi valori della lunghezza del filo, misurata con un metro.

Nella tabella sotto riportiamo alcune misure effettuate durante l'esperimento. Ogni riga rappresenta una misura del periodo per una determinata lunghezza del filo e ciascuna misura è accompagnata dalla sua incertezza.

$l(m)$	Incertezza $l(m)$	$T(s)$	Incertezza $T(s)$
0,30	0,02	1,1	0,2
0,50	0,02	1,4	0,2
0,70	0,02	1,6	0,1
0,90	0,02	1,9	0,1
1,10	0,02	2,1	0,1
1,30	0,02	2,2	0,1
1,50	0,02	2,4	0,1

Ogni misura di periodo è stata ottenuta facendo una media su dieci misure di periodo, quindi la corrispondente incertezza è lo scarto quadratico medio delle dieci misurazioni.

L'incertezza sulla lunghezza del pendolo dovrebbe essere pari alla sensibilità dello strumento di misura (0,5 mm per un normale metro), ma in questo caso è posta a 0,02 m per rendere ben visibili le barre di errore nel grafico.

IL PROGRAMMA IN PYTHON ASSO DOPO ASSO

Il codice del programma (file misure_gra! co.py) comincia con l'importazione dei moduli: Matplotlib e Numpy (righe 1 e 2, necessari rispettivamente alla creazione del grafico e alla gestione degli array che conterranno le misure delle grandezze fisiche in gioco).

```
1 import matplotlib.pyplot as plt
2 import numpy as np
```

La sezione di codice successiva definisce quattro array.

`misure_x` (riga 6) contiene le misure della grandezza che riporteremo sull'asse x del grafico. In questo caso si tratta della lunghezza l del pendolo in metri, i cui valori sono trascritti nel codice a partire dalla prima colonna della tabella precedente.

errori_x (riga 7) contiene le incertezze delle misure della grandezza che rappresenteremo sull'asse x del grafico, quindi della lunghezza l . I valori inseriti nel codice sono presi dalla seconda colonna della tabella.

misure_y (riga 8) contiene le misure della grandezza che rappresenteremo sull'asse y del grafico. In questo caso si tratta del periodo T del pendolo, misurato in secondi. I valori sono presi dalla terza colonna della tabella.

errori_y (riga 9) contiene le incertezze delle misure della grandezza che riporteremo sull'asse y del grafico, quindi del periodo T (quarta colonna della tabella).

```
4 #=====
5 #MISURE
6 misure_x = np.array([0.30,0.50,0.70,0.90,1.10,1.30,1.50])
7 errori_x = np.array([0.02,0.02,0.02,0.02,0.02,0.02,0.02])
8 misure_y = np.array([1.1,1.4,1.6,1.9,2.1,2.2,2.4])
9 errori_y = np.array([0.2,0.2,0.1,0.1,0.1,0.1,0.1])
10 #=====
```

Suggerimento Nel linguaggio Python il carattere separatore per le cifre decimali di un numero è il punto (.), mentre la virgola (,) fa parte della sintassi del linguaggio ed è utilizzata, per esempio, per separare i diversi elementi di un array.

Ora che abbiamo gli array contenenti le misure, procediamo alla creazione del grafico con il codice sottostante.

```
12 #=====
13 #GRAFICO
14 fig, ax = plt.subplots(figsize=(8,8))
15 ax.set_title("Misure del periodo T di un pendolo semplice al variare della lunghezza l")
16 ax.errorbar(misure_x, misure_y, xerr=errori_x, yerr=errori_y, ecolor="black", elinewidth=0.5,
17             capsize=3, marker="o", linestyle=" ")
18 ax.set_xlabel("l(m)")
19 ax.set_ylabel("T(s)")
20 ax.grid("both")
21 plt.show()
22 #=====
```

La riga `%c` crea la figura e gli assi, come descritto nella scheda introduttiva a Python. Ogni grafico sperimentale deve essere corredato da un titolo descrittivo, inserito nella riga `%c`.

Il codice delle righe `%e %c` si tratta di una sola istruzione spezzata su due righe per essere più leggibile) crea i punti del grafico e le loro barre di errore. Alla funzione `ax.errorbar` sono forniti i seguenti parametri:

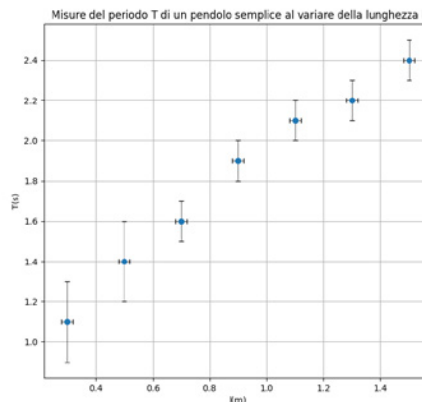
- `misure_x` `misure_y` che contengono i valori delle grandezze misurate;
- `xerr=errori_x` `yerr=errori_y` che assegnano i valori delle incertezze alle barre di errore del grafico;
- `ecolor="black"` che assegna il colore nero alle barre di errore;
- `elinewidth=0.5` che determina lo spessore delle barre di errore;
- `capsize=3` che definisce la larghezza dei trattini terminali posti sulle barre di errore;
- `marker="o"` che indica di rappresentare ogni misura con un piccolo cerchio centrato sul punto corrispondente nel piano cartesiano x-y;
- `linestyle=" "` che ordina di non tracciare linee di congiunzione tra i diversi punti.

Le righe `%e %c` creano le etichette sui due assi del grafico, all'interno delle quali abbiamo riportato le grandezze misurate con le loro unità di misura.

La riga `ax.grid("both")` disegna una griglia, utile a migliorare la leggibilità del grafico. Inoltre, la riga `plt.show()` crea il grafico secondo tutte le direttive date nel codice tra la riga `%e` e la riga `ax.grid("both")`.

I RISULTATI

L'esecuzione del programma produce la figura seguente.



Osserviamo che ogni punto ha come ascissa e ordinata rispettivamente i valori di lunghezza e periodo riportati nella tabella iniziale. Inoltre, ogni punto è dotato di due barre di errore che rappresentano le incertezze sulla lunghezza (direzione x) e sul periodo (direzione y).

I RISULTATI

Possiamo modificare il programma per renderlo adatto a esperimenti diversi da quello del pendolo semplice. L'unico requisito è che le grandezze fisiche in gioco siano due, analogamente alla lunghezza e al periodo. Per adeguare il programma a queste nuove situazioni dobbiamo:

- 1. inserire le misure delle due grandezze coinvolte e le loro incertezze negli array definiti tra riga 9 e riga 10;
- 2. adeguare la descrizione presente nel titolo del grafico (riga 11);
- 3. inserire i nomi delle grandezze misurate e le loro unità di misura nelle etichette degli assi (righe 12 e 13).

A questo punto il programma è pronto per creare il nostro nuovo grafico sperimentale.

Potremo utilizzare il programma anche in ambiti diversi dalla fisica, per esempio per tracciare un grafico che descrive la crescita di una pianta, lo sviluppo di una coltura batterica oppure per visualizzare lo storico della nostra media dei voti rispetto allo storico del numero di ore dedicate allo studio.

ESERCIZIO

Facendoti aiutare dall'insegnante, realizza un esperimento di fisica nel quale individui due grandezze fisiche che sei in grado di misurare. Puoi usare metri, righelli, cronometri, ma ricorda che anche il tuo smartphone, dotato di applicazioni gratuite come phyphox, diventa un potente strumento di misura per svariate grandezze fisiche. Riporta le tue misure nel programma e traccia il grafico.

Un'ultima possibilità, utile in casi come quello del pendolo semplice, è quella di rappresentare graficamente grandezze indirette, cioè derivate dalle grandezze misurate. Nel caso del pendolo semplice vale la legge:

$$T^2 = k l \quad [P!]$$

dove k è una costante, a condizione che l'ampiezza di oscillazione sia piccola rispetto alla lunghezza l del pendolo.

La legge [P!] afferma che il quadrato del periodo del pendolo è direttamente proporzionale

alla sua lunghezza. La proporzionalità diretta si manifesta in un nuovo grafico in cui la grandezza rappresentata sull'asse y sia T^2 invece di T , poiché i punti misurati si dispongono lungo una retta passante per l'origine.

Il programma `misure_grafico_derivare.py` crea questo nuovo grafico. Rispetto al codice del programma precedente abbiamo aggiunto questo blocco di istruzioni.

```
12 #=====
13 #GRANDEZZE DERIVATE
14 misure_y_2 = misure_y**2
15 errori_y_2 = 2*errori_y*misure_y
16 #=====
```

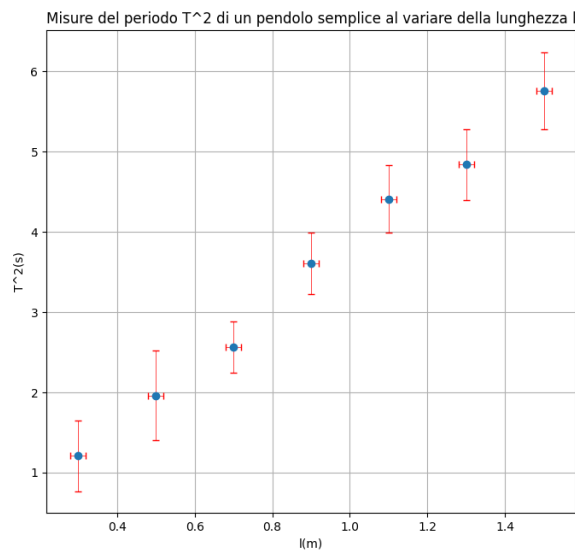
La riga `%e` leva al quadrato le misure dell'array `misure_y`, per ottenere T^2 da T , mentre la riga `%c` allarga le incertezze su T^2 .

Suggerimento Ricordiamo che data una misura $\bar{a} \pm \Delta \bar{a}$ in cui \bar{a} è il valore della grandezza e $\Delta \bar{a}$ l'incertezza, si ha che $\Delta \bar{a}^2 = \Delta \bar{a} \frac{d\bar{a}^2}{d\bar{a}} = \frac{d\bar{a}^2}{d\bar{a}} \Delta \bar{a} = 2\bar{a} \Delta \bar{a}$

Dovendo rappresentare sull'asse y la grandezza indiretta, i parametri passati a `ax.errorbar` cambiano come nel codice sottostante.

```
22 ax.errorbar(misure_x, misure_y_2, xerr=errori_x, yerr=errori_y_2, ecolor="red", elinewidth=0.5,  
23             capsize=3, marker="o", linestyle="none")
```

Dopo l'esecuzione del programma otteniamo il grafico seguente.



Osserviamo che in questo nuovo grafico i punti giacciono approssimativamente su una retta, proprio come ci aspettavamo.