

International Cosmic Day 2025:

Introduzione all'analisi dei dati

Pietro Monti-Guarnieri

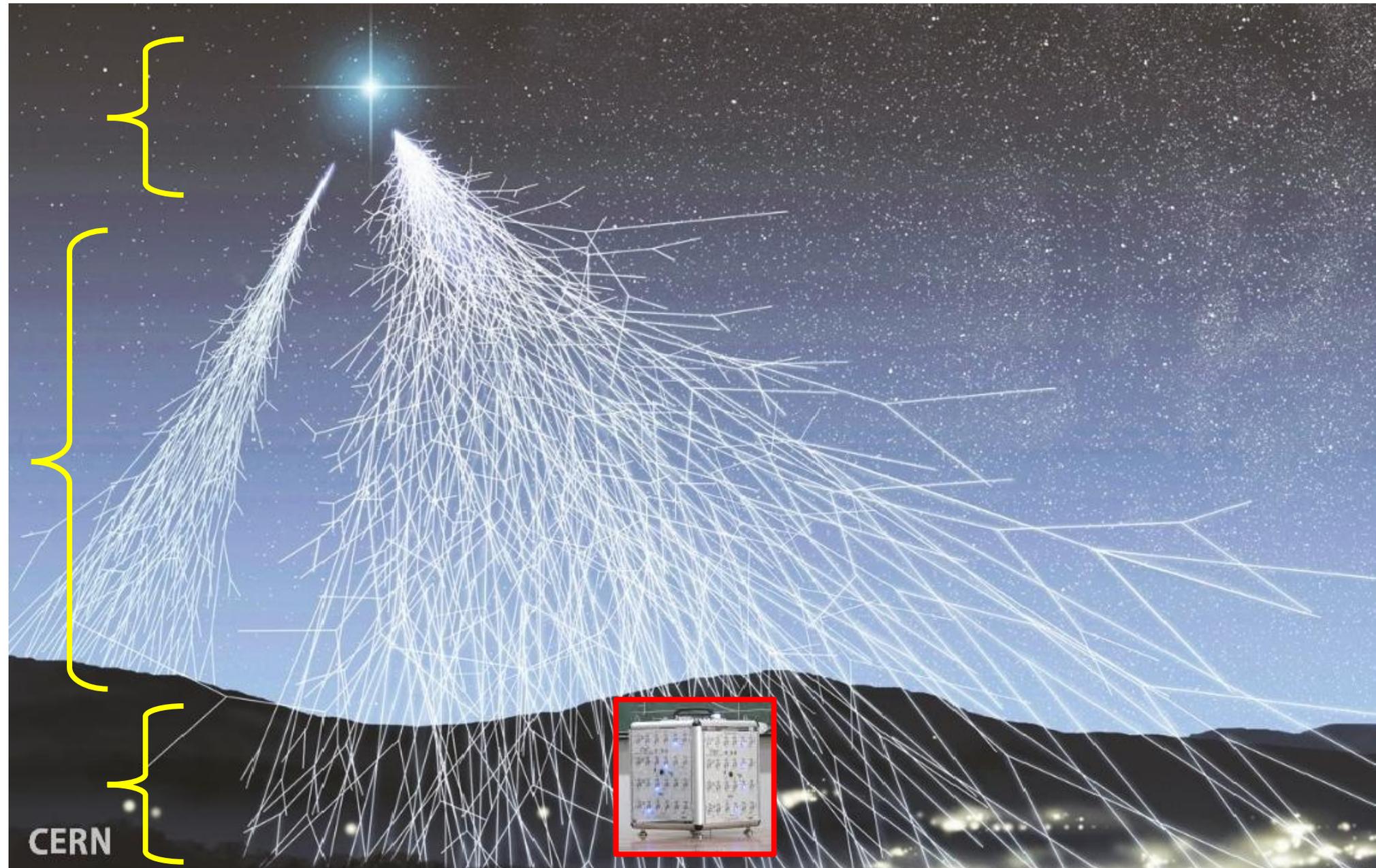
13 Novembre 2025



Raggi cosmici primari

Particelle prodotte nelle interazioni in alta atmosfera

Particelle residue a terra: **muoni**
($\sim 1 / \text{cm}^2 \cdot \text{min}$)



Promemoria: il Cosmic Ray Cube (CRC)

Il CRC misura la **frequenza di arrivo delle particelle** cariche che lo attraversano (detta anche *rate*):

$$f = \frac{\Delta N}{\Delta T} [s^{-1}] \cong \frac{\Delta N}{\Delta T} [Hz]$$

Dove

- ❑ ΔN = numero di particelle che attraversano il CRC
- ❑ ΔT = tempo di misura

Il CRC **non misura** alcuna altra proprietà interessante dei raggi cosmici (come energia, carica, composizione...)

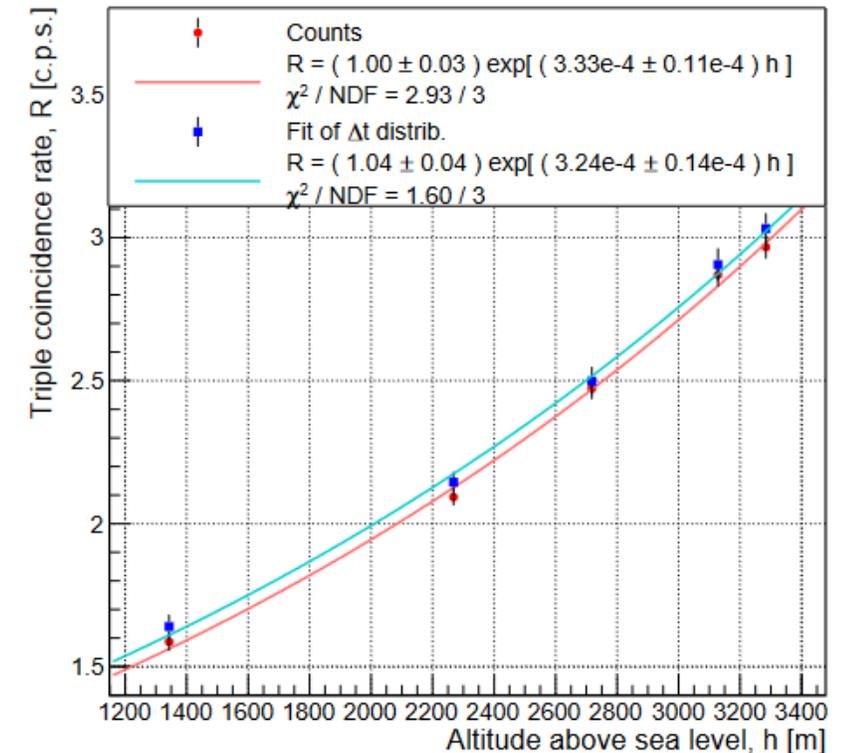


Scopo principale di oggi: misurare la **frequenza di arrivo dei raggi cosmici** in varie condizioni... ma perché?

In generale, la frequenza cambia in base a:

- Periodo dell'anno (→ modulazione solare)
- Altitudine (per i curiosi, vedete [qui](#))
- Angolo di osservazione

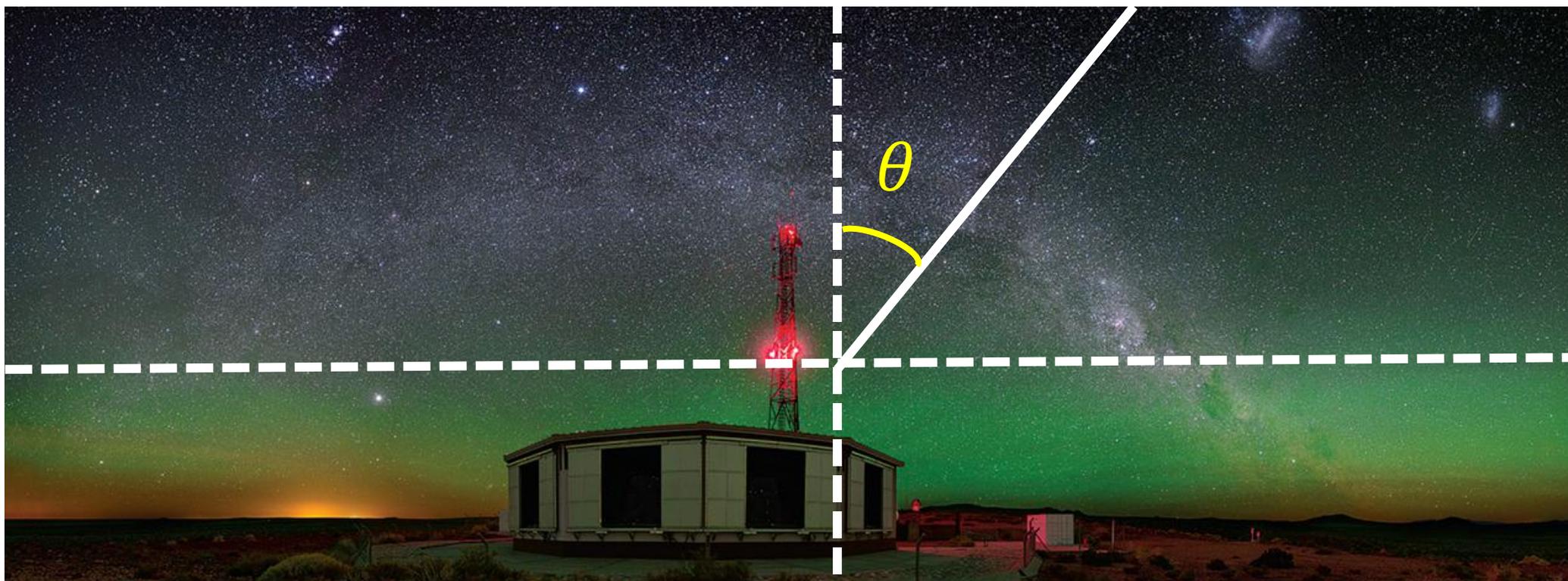
Noi lavoreremo sull'ultimo caso!



Sperimentalmente si è osservato che:

$$f(\theta) \propto \cos^2(\theta)$$

a causa dell'enorme differenza di lunghezza di cammino dei raggi cosmici in alta atmosfera ($L \cdot \rho = g/cm^2$), tra zenit e orizzonte



Sperimentalmente si è osservato che:

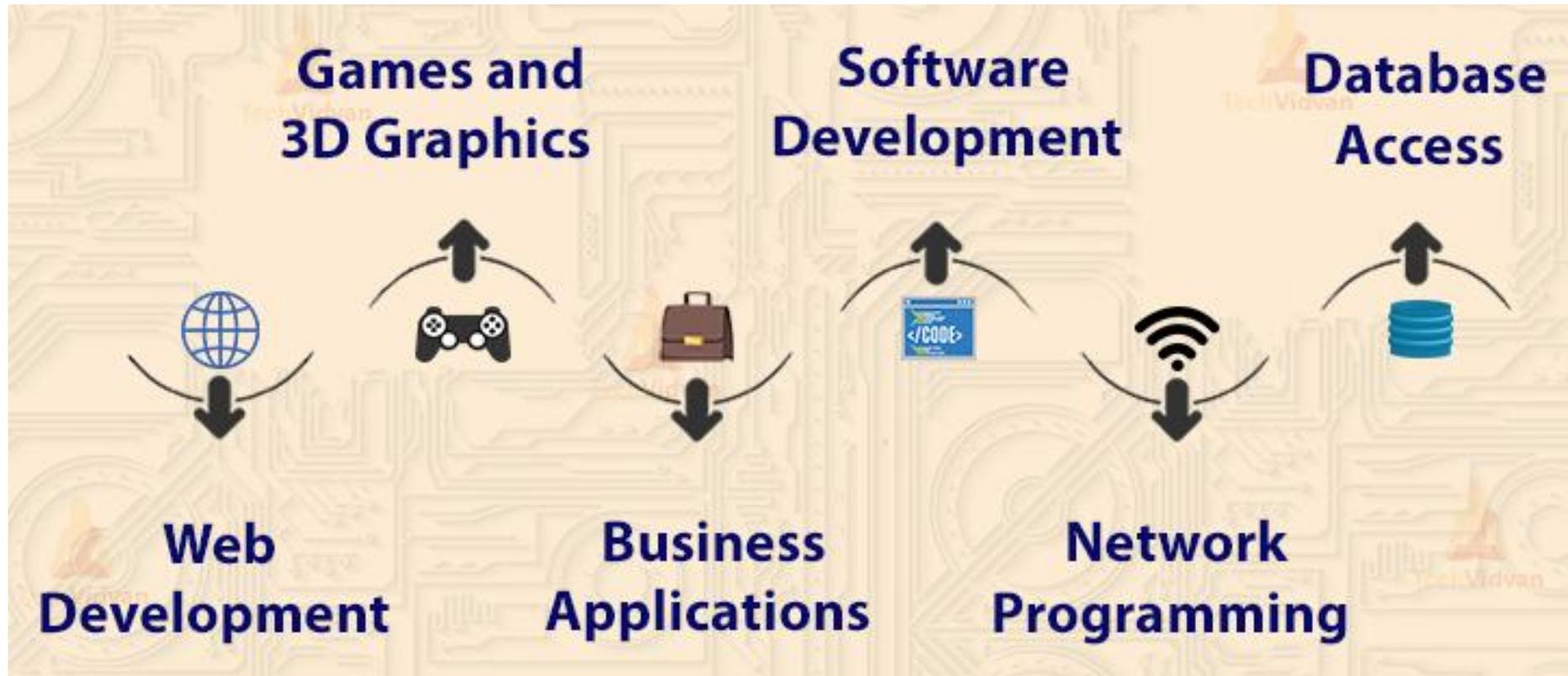
$$f(\theta) \propto \cos^2(\theta)$$

a causa dell'enorme differenza di lunghezza di cammino dei raggi cosmici in alta atmosfera ($L \cdot \rho = g/cm^2$), tra zenit e orizzonte



Lo strumento per fare analisi dati: Python

Python è un linguaggio di programmazione open-source, usato da più di 10 milioni di persone (2023) per molteplici scopi (tra cui sviluppo di algoritmi AI e **analisi dati**)



Istruzioni per l'analisi dati (almeno 1 PC / gruppo)

1. Aprire il sito dell'ICD alla voce **“Misura e analisi dati (tutti)”**:

<https://agenda.infn.it/event/48413/contributions/274348/>

2. Aprire [Jupyter Browser](#)

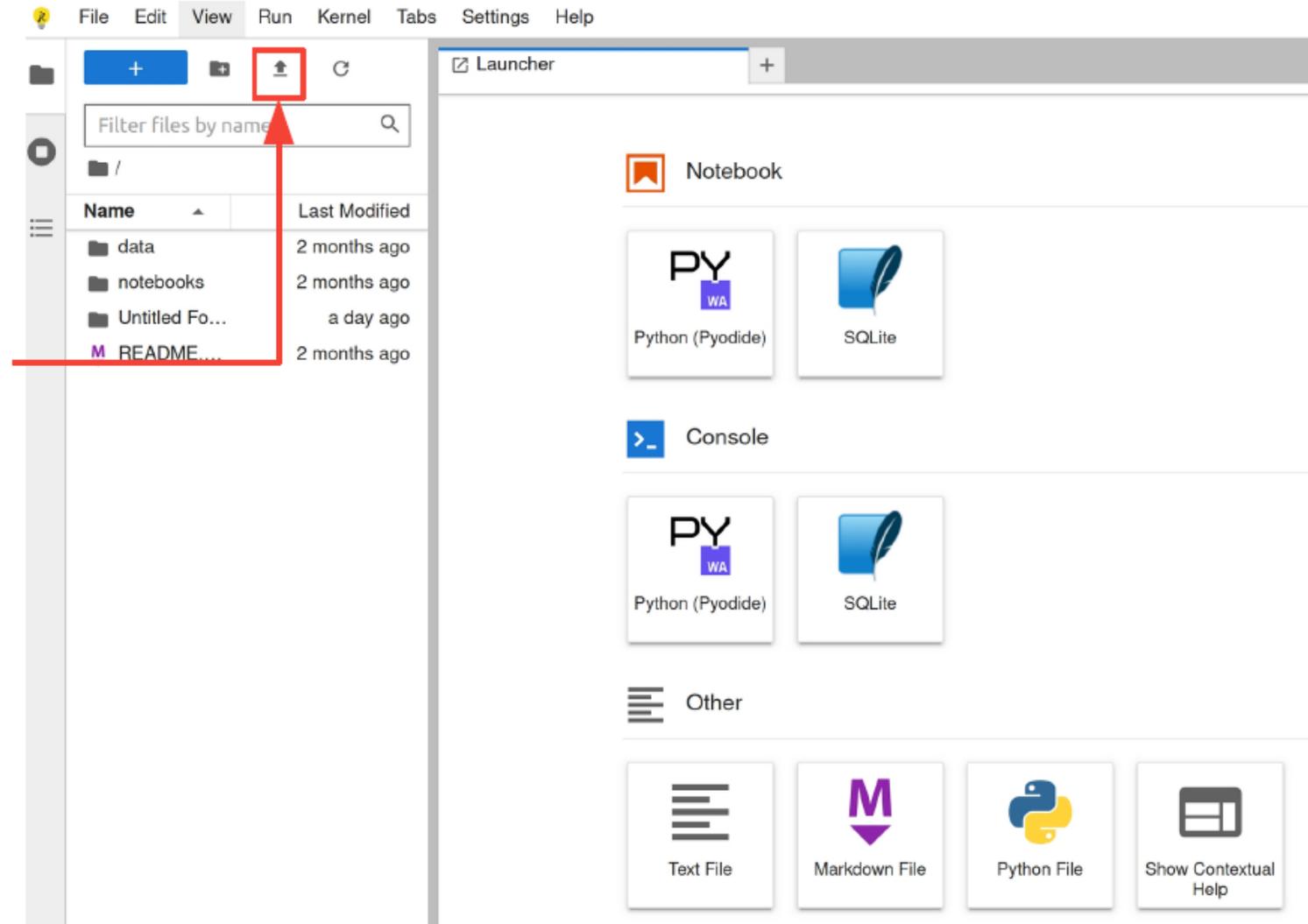
3. Scaricare la cartella **“Analisi_ICD2025”** sul computer ed estrarne il contenuto

The screenshot displays the event page for 'Misura e analisi dati (tutti)'. On the left, a sidebar menu includes 'Sintesi', 'Calendario', 'Registration', 'Informativa dati personali (DA SCARICARE)', and a button for 'Per maggiori informazioni' with the email 'scuola@ts.infn.it'. The main content area shows event details: '13 nov 2025, 14:00', '2h', and 'Auditorium (Area di Ricerca Padriciano)'. Below this, the 'Relatore' is identified as 'Pietro Monti-Guarnieri (Istituto Nazionale di Fisica Nucleare)'. The 'Materiali di presentazione' section contains three items: 'Analisi_ICD2025.zip' (highlighted with a green box), 'ICD25_analisi.pdf', and 'Jupyter Browser' (highlighted with a red box). A red line connects the 'Per maggiori informazioni' button to the 'Jupyter Browser' link, and a green arrow points from the text 'Analisi_ICD2025' in the instructions to the corresponding zip file.

4. Importare sulla home di Jupyter Notebook tutti i file, dentro una cartella apposita
5. Ogni gruppo si occuperà di acquisire e analizzare un file dati, corrispondente a un angolo X, da salvare con il nome:

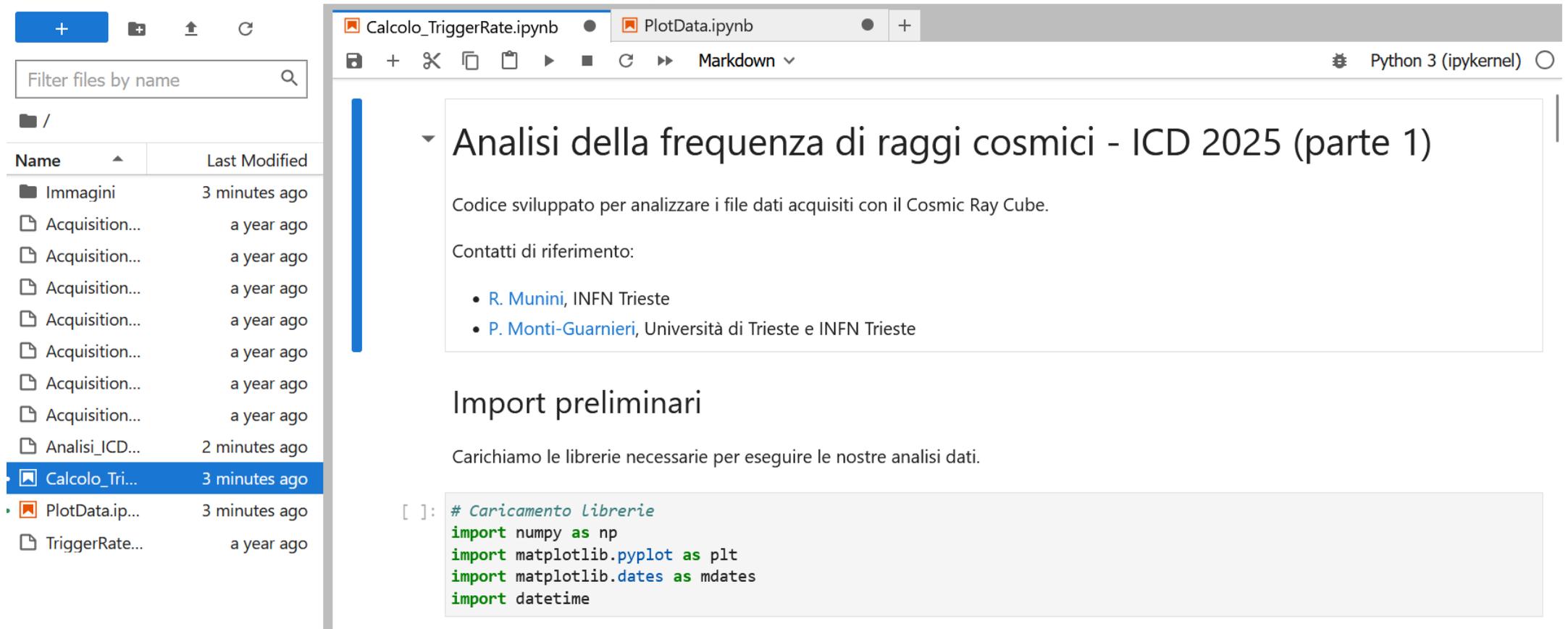
Acquisition_X_CRC_data.txt

e.g. Acquisition_0_CRC_data.txt



6. Cliccare sul primo script, *Calcolo_TriggerRate.ipynb*

A questo punto si aprirà un file contenente un codice Python, che può essere **eseguito cella per cella** (*Run* → *Run Selected Cells*, oppure Shift + Invio), visualizzando immediatamente l'output



The screenshot shows a Jupyter Notebook interface. On the left is a file browser with a search bar and a list of files. The file 'Calcolo_Tri...' is selected. On the right is a notebook cell with the following content:

Analisi della frequenza di raggi cosmici - ICD 2025 (parte 1)

Codice sviluppato per analizzare i file dati acquisiti con il Cosmic Ray Cube.

Contatti di riferimento:

- [R. Munini](#), INFN Trieste
- [P. Monti-Guarnieri](#), Università di Trieste e INFN Trieste

Import preliminari

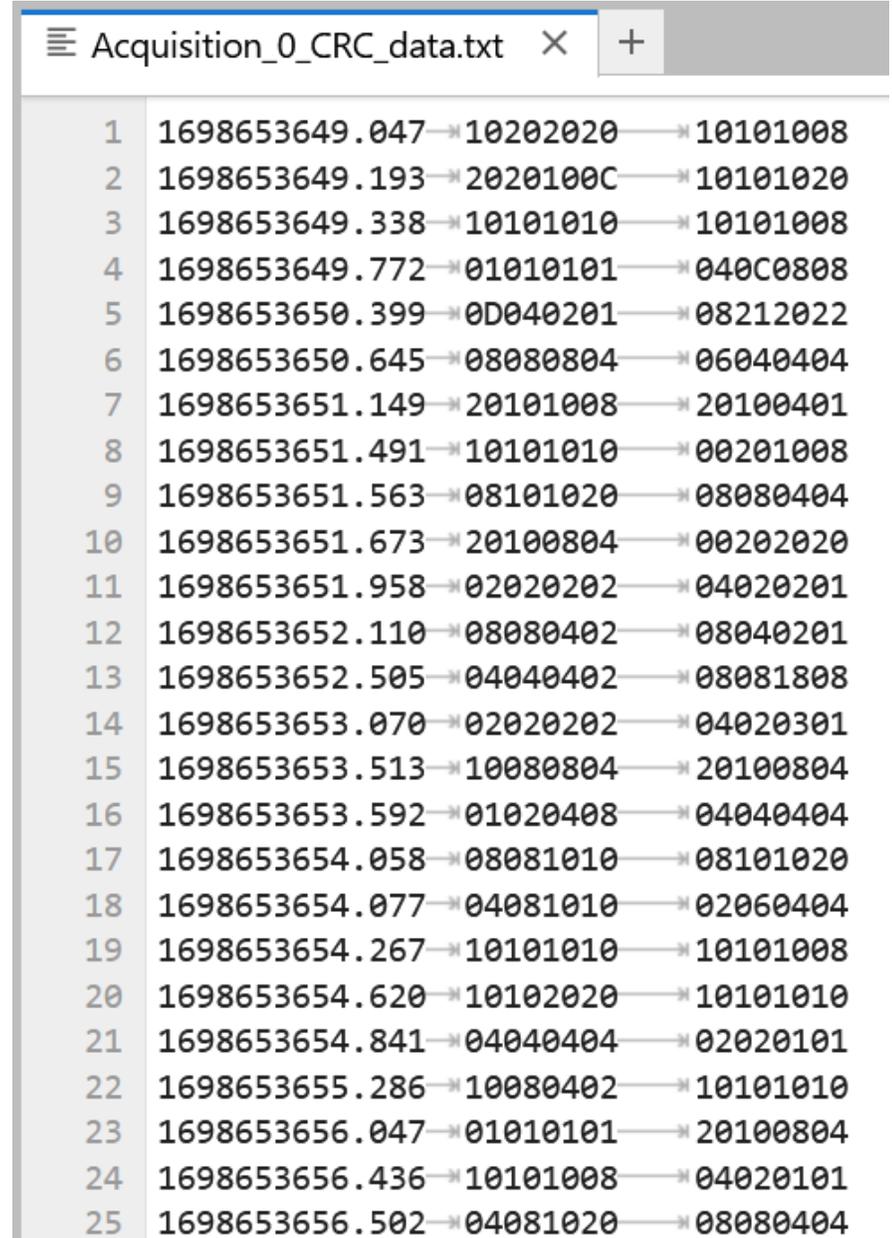
Carichiamo le librerie necessarie per eseguire le nostre analisi dati.

```
[ ]: # Caricamento librerie
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.dates as mdates
import datetime
```

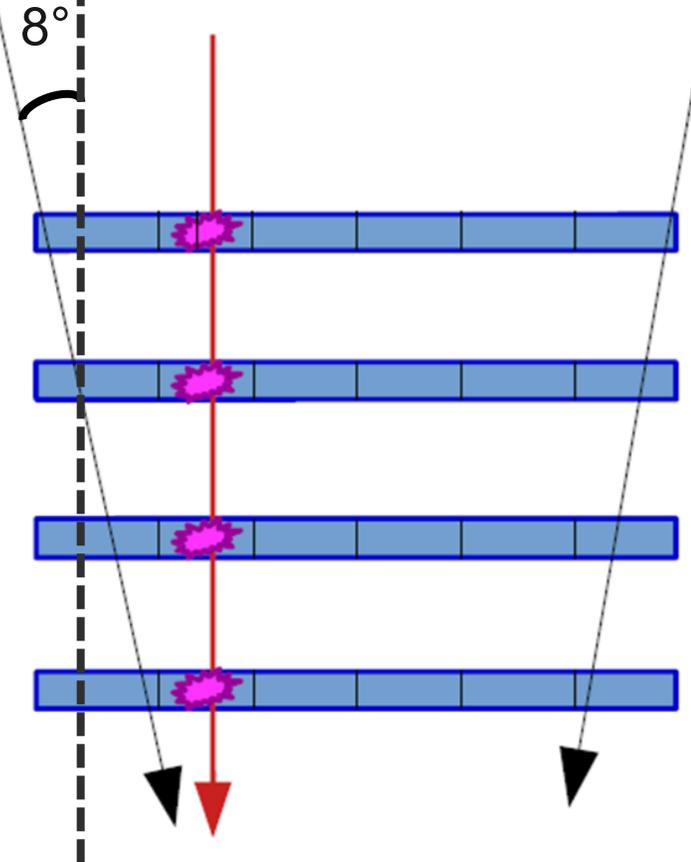
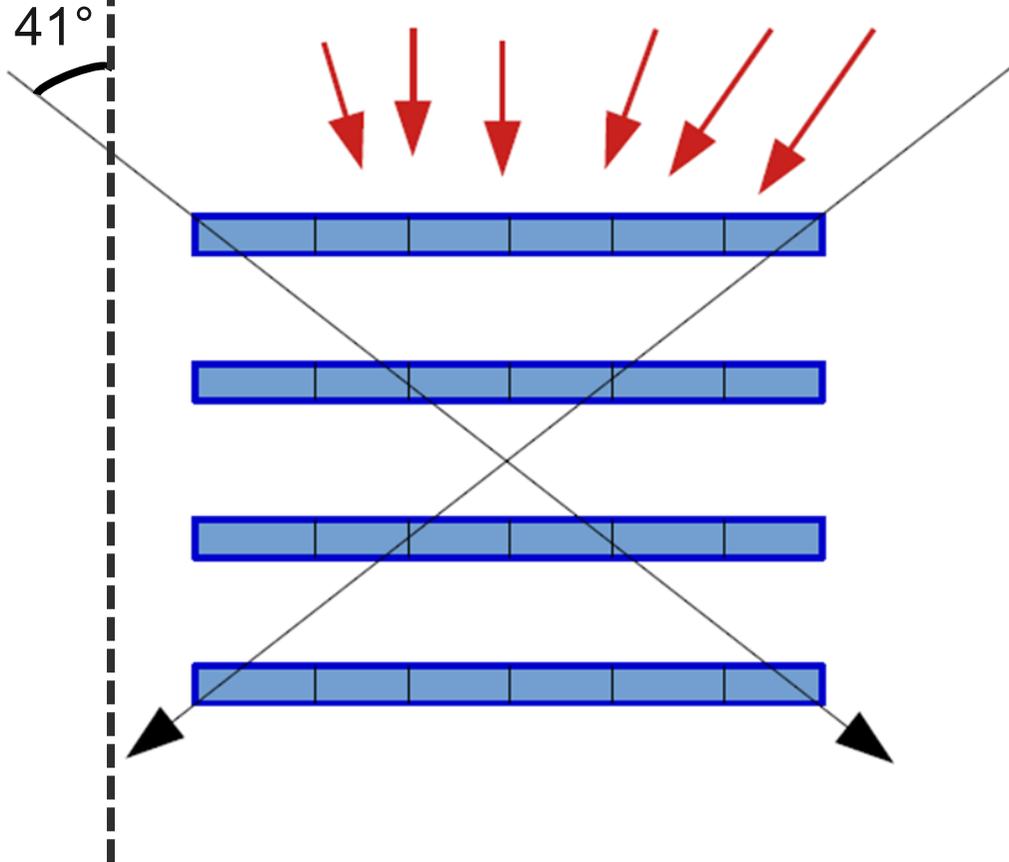
Lo script *Calcolo_TriggerRate*

Scopo dello script:

- ❑ Aprire un file dati (acquisito a un angolo θ)
- ❑ Misurare il numero di eventi ΔN_{tot} e il tempo di acquisizione ΔT
- ❑ Ricavare **la frequenza dei raggi cosmici** (f_{tot})
- ❑ Misurare il numero di eventi verticali ΔN_{vert} (vedi prossima slide)
- ❑ Ricavare **la frequenza dei raggi verticali** (f_{vert})



Line	Timestamp	Hex 1	Hex 2
1	1698653649.047	10202020	10101008
2	1698653649.193	2020100C	10101020
3	1698653649.338	10101010	10101008
4	1698653649.772	01010101	040C0808
5	1698653650.399	0D040201	08212022
6	1698653650.645	08080804	06040404
7	1698653651.149	20101008	20100401
8	1698653651.491	10101010	00201008
9	1698653651.563	08101020	08080404
10	1698653651.673	20100804	00202020
11	1698653651.958	02020202	04020201
12	1698653652.110	08080402	08040201
13	1698653652.505	04040402	08081808
14	1698653653.070	02020202	04020301
15	1698653653.513	10080804	20100804
16	1698653653.592	01020408	04040404
17	1698653654.058	08081010	08101020
18	1698653654.077	04081010	02060404
19	1698653654.267	10101010	10101008
20	1698653654.620	10102020	10101010
21	1698653654.841	04040404	02020101
22	1698653655.286	10080402	10101010
23	1698653656.047	01010101	20100804
24	1698653656.436	10101008	04020101
25	1698653656.502	04081020	08080404



- ❖ I raggi cosmici visti dal CRC possono **arrivare da molti angoli diversi**, per la sua estensione superficiale e il suo spessore (angolo massimo: 41° dalla verticale)
- ❖ Può essere interessante confrontare la frequenza di tutti gli eventi con quella calcolato sui soli **eventi verticali**, dove i raggi cosmici attraversano sempre la N-esima barra in ogni piano (angolo massimo: 8° dalla verticale).

Una volta acquisiti i dati a tutti gli angoli...

- ❑ Salvare un file di testo chiamato *TriggerRate_OneDayAcquisition.txt* contenente i seguenti dati, in ordine (una riga per ciascuna presa dati ad angolo diverso):
 - ❑ Angolo di misura (in radianti)
 - ❑ Frequenza misurata: verticale e totale
 - ❑ **Errore** associato alla frequenza verticale e totale

```
TriggerRate_OneDayAcquisiti X +
1 0.0*0.03558060724613263*2.580835209318318→0.0038367555516128373→0.03267665476846592
2 0.261666666666666666*0.032345166047961754→2.4171273044591417→0.003301214685962542→0.028537691454641967
3 0.523333333333333333→0.0260101195972789→2.0719550589832383→0.002682738997036567→0.023944039120580386
4 0.785→0.015691044417866926→1.5442151989169726→0.002060335070353568→0.02043929163904804
5 1.046666666666666666→0.00897740662962229*1.0040884030362163→0.0014375355495589671→0.015202993873630475
6 1.308333333333333333→0.0026468466636622814→0.5570833742696261→0.0006419546099466641→0.009313223585997574
7 1.57→0.0010695129972348818→0.3852385816040044→0.0004783007529273481→0.009077639231229818
8
```

Errore...?

- ❖ In fisica è **fondamentale** attribuire un **errore**, o **incertezza**, a ogni grandezza che viene misurata
- ❖ A volte è più importante (e difficile) capire come misurare l'errore rispetto alla grandezza stessa!
- ❖ Normalmente distinguiamo due tipi di errore:
 1. **Errore statistico**, dovuto cioè alla **durata limitata della misura** (*intuitivamente, ci aspettiamo che una misura più lunga dello stesso fenomeno sia più precisa, rispetto a una singola misura più breve*).

Negli esperimenti di conteggio si può dimostrare che l'errore associato a un numero di eventi misurati ΔN è $Err(\Delta N) \sim \sqrt{\Delta N}$ e di conseguenza, per la frequenza, vale:

$$Err(f) = \frac{\sqrt{\Delta N}}{\Delta T}$$

2. Errore sistematico, dovuto cioè a problematiche dell'apparato di misura o dello sperimentatore.

Esempi nel nostro caso:

- ✓ Difetti dei fotomoltiplicatori collegati alle barre del CRC, o dei circuiti elettronici ad essi collegati
- ✓ Variazioni del flusso di raggi cosmici primari (che noi stiamo assumendo costante, ma ci sono leggere variazioni dovute all'attività solare...)
- ✓ Geometria cubica del CRC

L'errore sistematico è molto più difficile da quantificare, e **non ce ne occuperemo in questa occasione** (*ma riflettete su come si potrebbe misurare...*)

Lo script *PlotData*

Scopo dello script:

- ❑ Aprire il file con le frequenze misurate a vari angoli
- ❑ Rappresentare la frequenza misurata in funzione dell'angolo di vista
- ❑ Calcolare i **residui**, ovvero la differenza relativa tra i dati misurati e il modello teorico (il modello è $f_0 \cdot \cos^2 \theta$, dove f_0 è la frequenza misurata a 0°):

$$\text{Residui}(\theta) = \frac{f(\theta) - f_0 \cdot \cos^2 \theta}{f_0 \cdot \cos^2 \theta}$$

Se i dati sono in buon accordo al modello, la differenza a numeratore dovrebbe essere piccola e quindi i **residui molto prossimi a zero**.

È così?

Si direbbe di no!

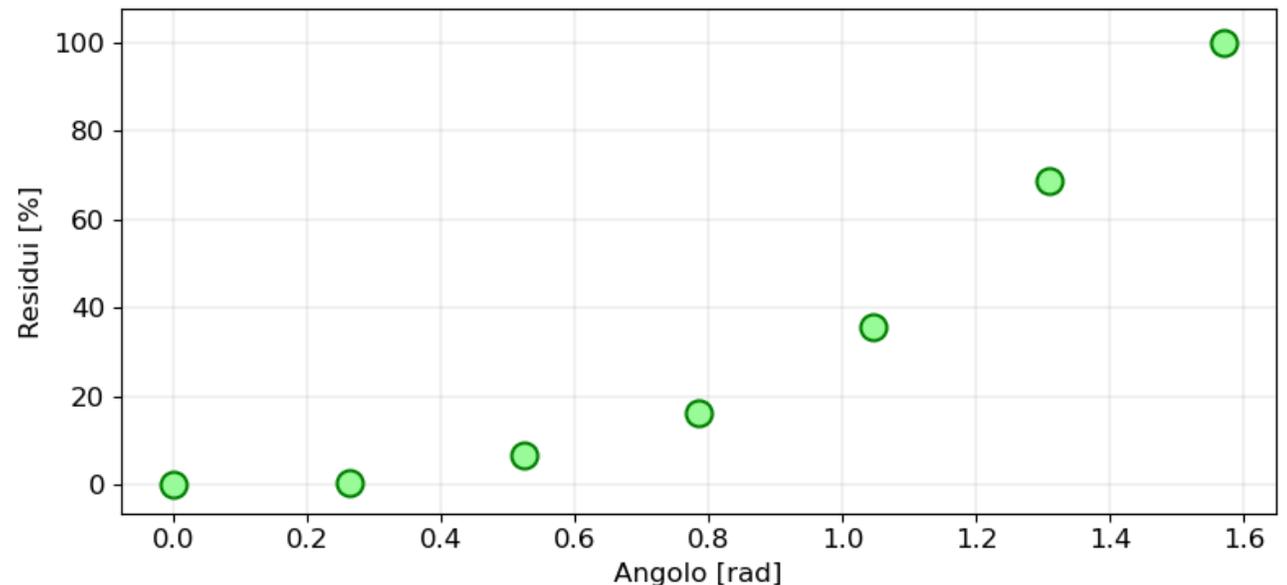
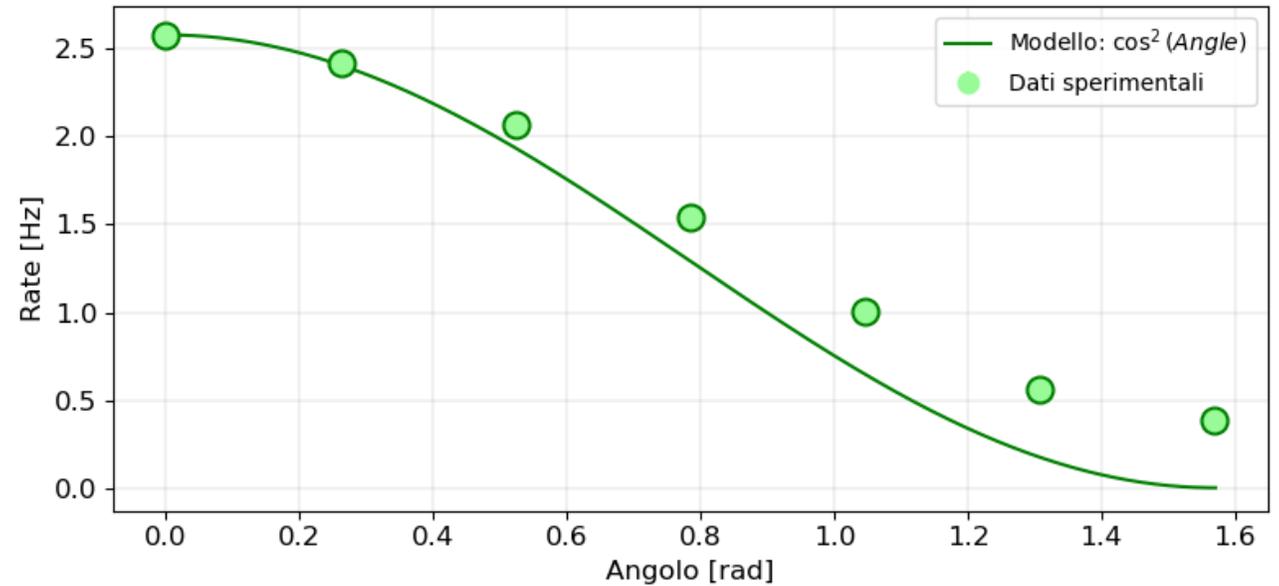
Questo può significare che:

- ❑ Il modello è giusto, ma i dati sono affetti da grossi errori e li stiamo sottostimando
- ❑ Il modello è sbagliato e va modificato

Proviamo a modificare l'esponente del coseno, o ad aggiungere una costante alla funzione:

$$f(\theta) = A \cdot \cos^B \theta + C$$

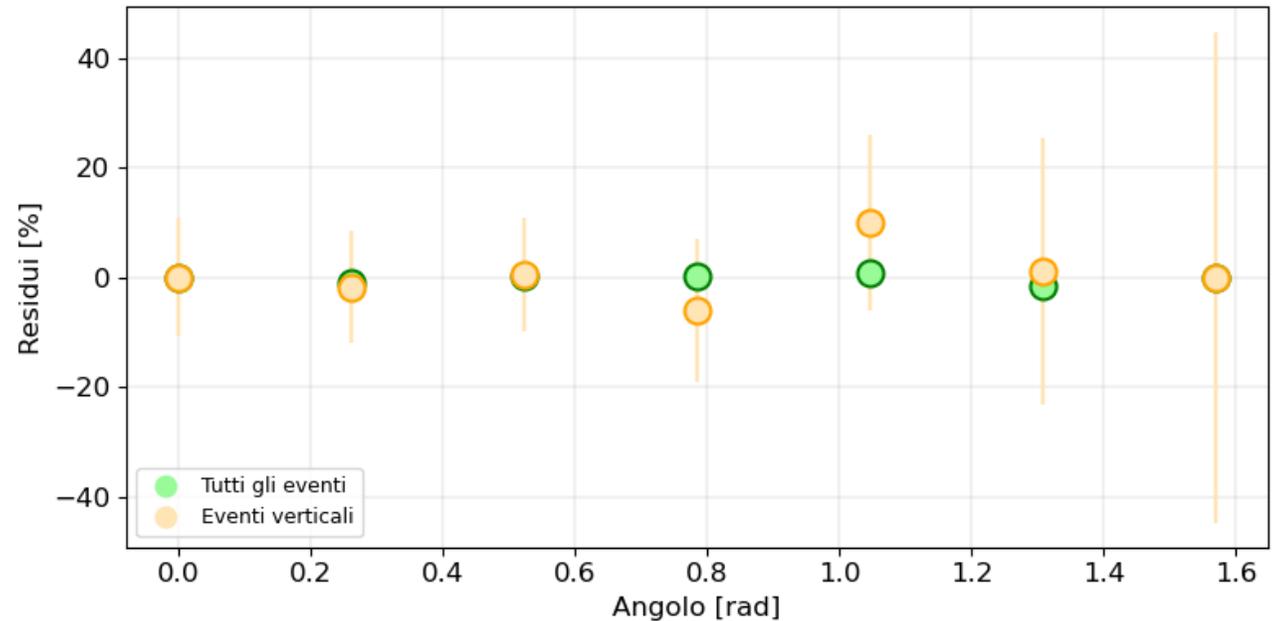
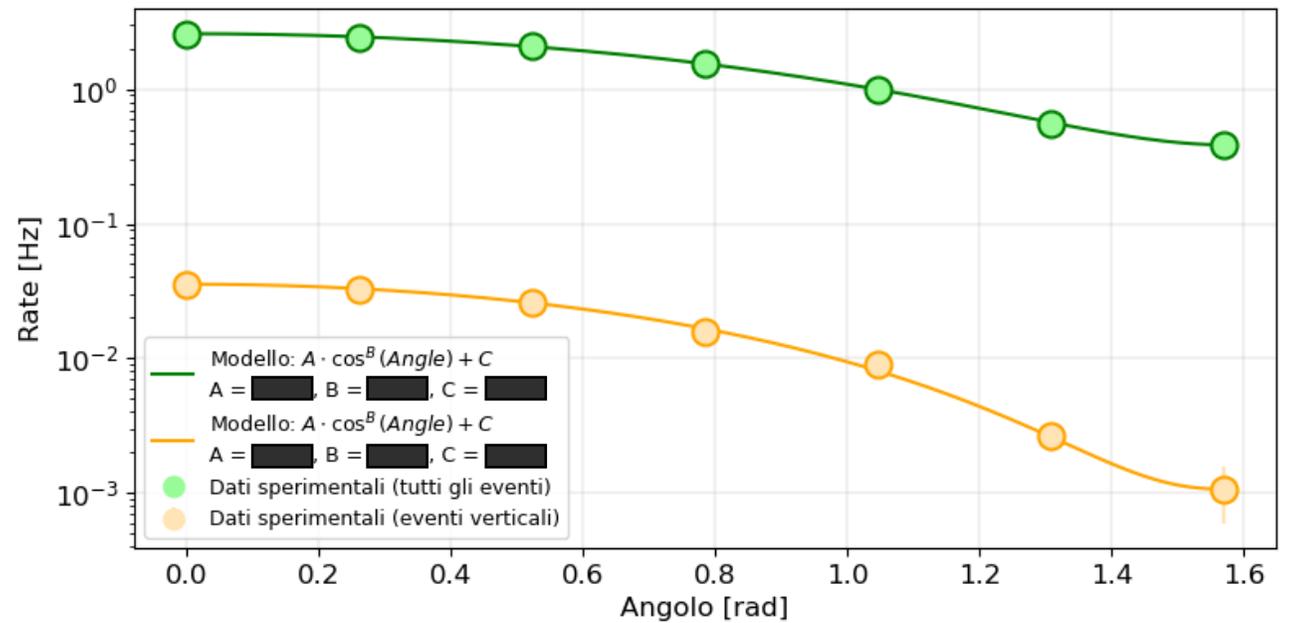
Rate (tutti gli eventi)



Molto meglio!

Ma per quale combinazione di parametri?

Provate a trovarla!



Istruzioni per la stesura della relazione finale

La relazione finale deve essere strutturata nel seguente modo:

- Scopo della misura
- Apparato sperimentale
- Presa dati (descrizione della procedura seguita)
- Analisi dati (descrizione del metodo usato e dei risultati ottenuti, includendo i valori dei parametri che permettono di riprodurre al meglio i dati)
- Se possibile, svolgere l'analisi usando sia i dati del 13 novembre sia i nostri
- Problematiche sperimentali che avete incontrato e come le avete affrontate

- ❖ Ogni relazione sarà fatta in collaborazione tra 2-3 studenti e dovrà avere una lunghezza di 2 o 3 pagine (Word o LaTeX), includendo anche il grafico finale, con le curve $f(\theta)$ per tutti gli eventi e per gli eventi verticali
- ❖ Le relazioni devono essere inviate entro il **30 gennaio 2026**

N.B. : **non stupitevi** se le misure fatte il 13 novembre danno risultati diversi da quelli che trovate con i dati presi in precedenza! Ci possono essere molte differenze tra prese dati differenti:

- Durata delle misure
- Data delle misure (→ il flusso di raggi cosmici primari cambia nel tempo)
- Possibili malfunzionamenti del rivelatore...

Non cercate di riprodurre grafici identici a quelli che vi abbiamo presentato, ma create i vostri e commentate esplicitamente le differenze che osservate!

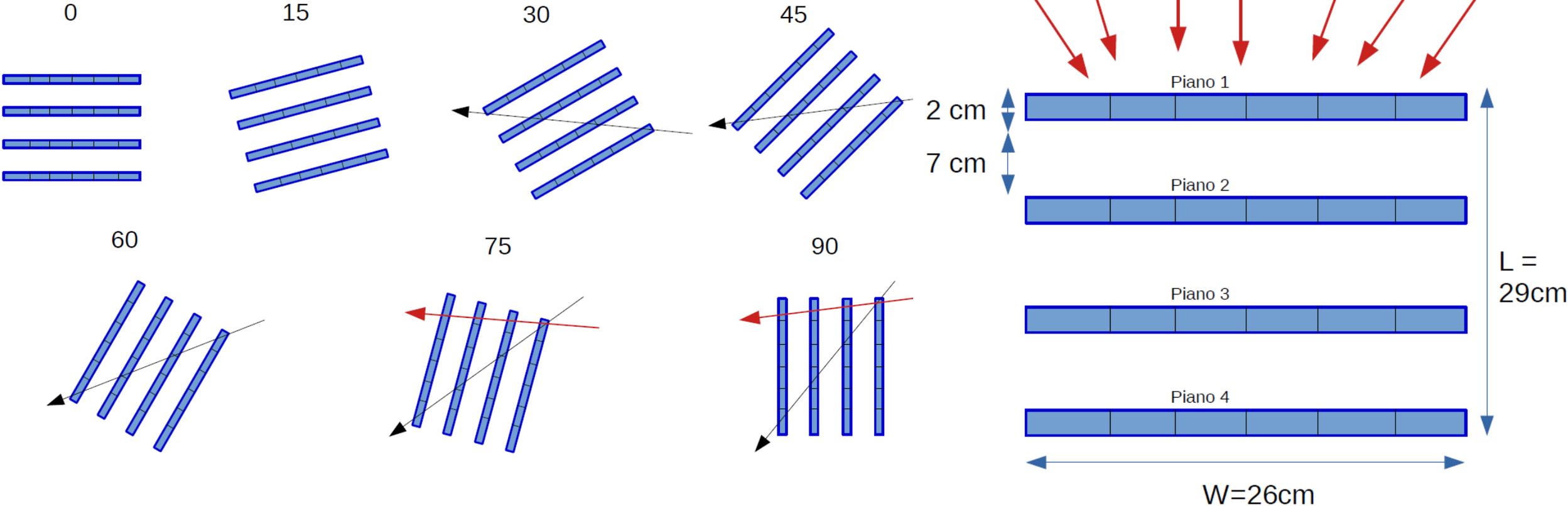
Negli script che vi abbiamo consegnato ci sono molti **spunti di riflessione** sull'analisi che state facendo. Eccone altri (non sono un compito, ma un suggerimento guida per relazioni):

1. Come si può aumentare la statistica di eventi acquisita?
2. Quali sono gli errori dominanti nella misura? Come li potremmo ridurre?
3. Come si può restringere l'angolo di apertura («angolo solido») del telescopio e quindi avere una stima più precisa del flusso di muoni ad un dato angolo?
4. Vi aspettate che il flusso di muoni sia maggiore o minore di quello che abbiamo misurato, se ripetessimo l'esperienza in cima al Monte Bianco (altitudine 4805 m)? E in fondo alla Fossa delle Marianne (profondità 10994 m)?

Grazie dell'attenzione!

Per domande, dubbi o curiosità contattare: pietro.monti-guarnieri@phd.units.it

Figure aggiuntive sul CRC



Per i più volenterosi: il fit

Fare un fit = cercare la migliore combinazione di parametri che permettono di descrivere un set di dati con una funzione analitica

In Python esiste un algoritmo per farlo in modo (quasi) automatico: **curve_fit**

```
scipy.optimize.
```

```
curve_fit
```

```
curve_fit(f, xdata, ydata, p0=None, sigma=None, absolute_sigma=False,
check_finite=None, bounds=(-inf, inf), method=None, jac=None, *,
full_output=False, nan_policy=None, **kwargs)
```

Use non-linear least squares to fit a function, f, to data.

Assumes `ydata = f(xdata, *params) + eps`.

```
from scipy.optimize import curve_fit

def cos2_final(x,A,B,C):
    return ( A*(np.cos(x)**B) + C )

p0 = [np.max(triggerRate_Total)-np.min(triggerRate_Total),2,np.min(triggerRate_Total)]

popt, pcov = curve_fit(cos2_final, angles, triggerRate_Total, p0=p0, sigma=errorY_Total, absolute_sigma=True,
                        bounds=(0,np.inf), full_output=False)
```