UNIVERSITÀ DEL SALENTO

INFN Istituto Nazionale di Fisica Nucleare SEZIONE DI LECCE

Sviluppo di tecniche di machine learning per lo studio della composizione di massa dei raggi cosmici di altissima energia e della componente muonica degli sciami atmosferici estesi con i dati dell'Osservatorio Pierre Auger

"Self Organizing Map algorithms on GPUs for faster Time Series Analysis clustering"

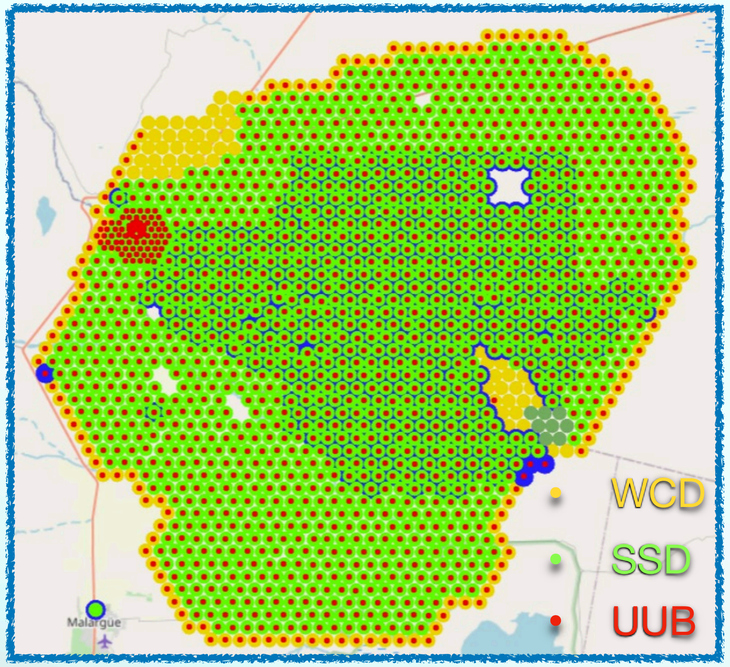
Matteo Conte, Università del Salento, INFN - sezione di Lecce

PIERRE AUGER OBSERVATORY

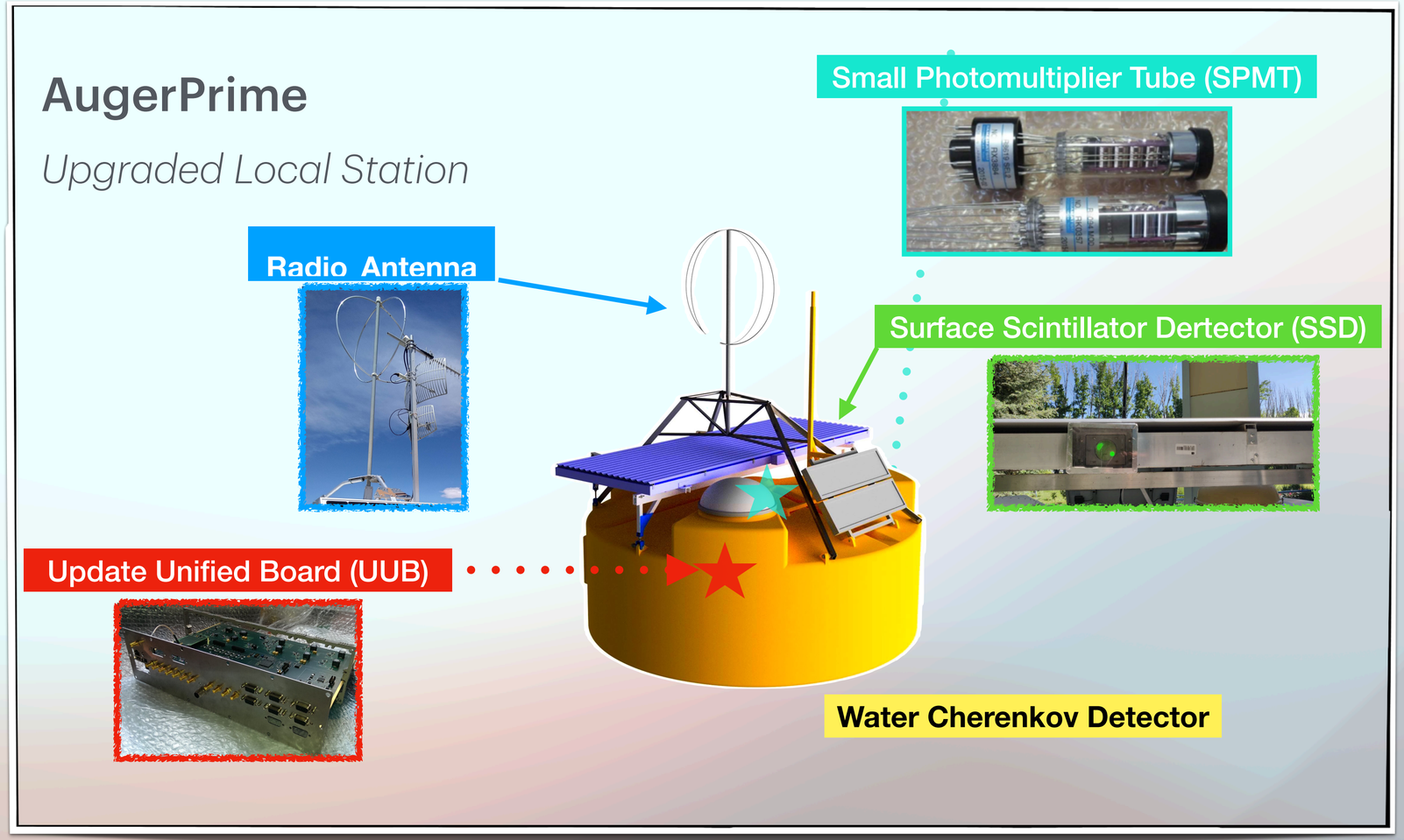
Punti chiave

- Osservatorio Pierre Auger e il suo upgrade
- Composizione di Massa dei raggi cosmici ad altissime energie
- Self Organizing Map
- Addestramento, monitoraggio e ottimizzazione
- Test preliminari per la caratterizzazione della frazione di segnale muonico f_μ:
  - Sui ‘meta’ dati di ricostruzione dello sciame atmosferico esteso
  - Sui segnali (temporali) dei PMT
- Step successivi

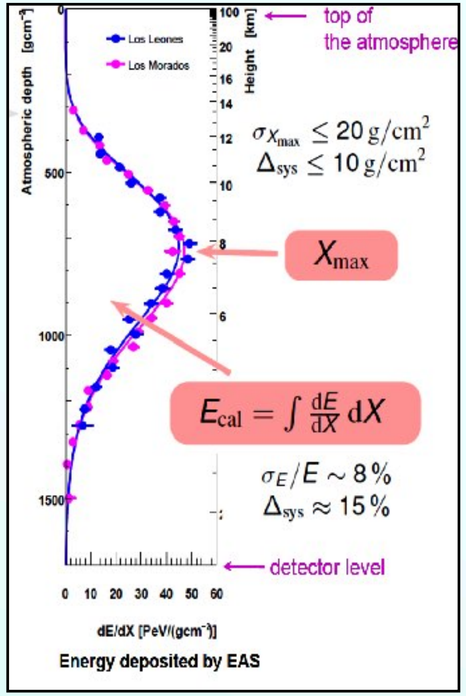
Il Rivelatore di Superficie dell'Osservatorio Pierre Auger

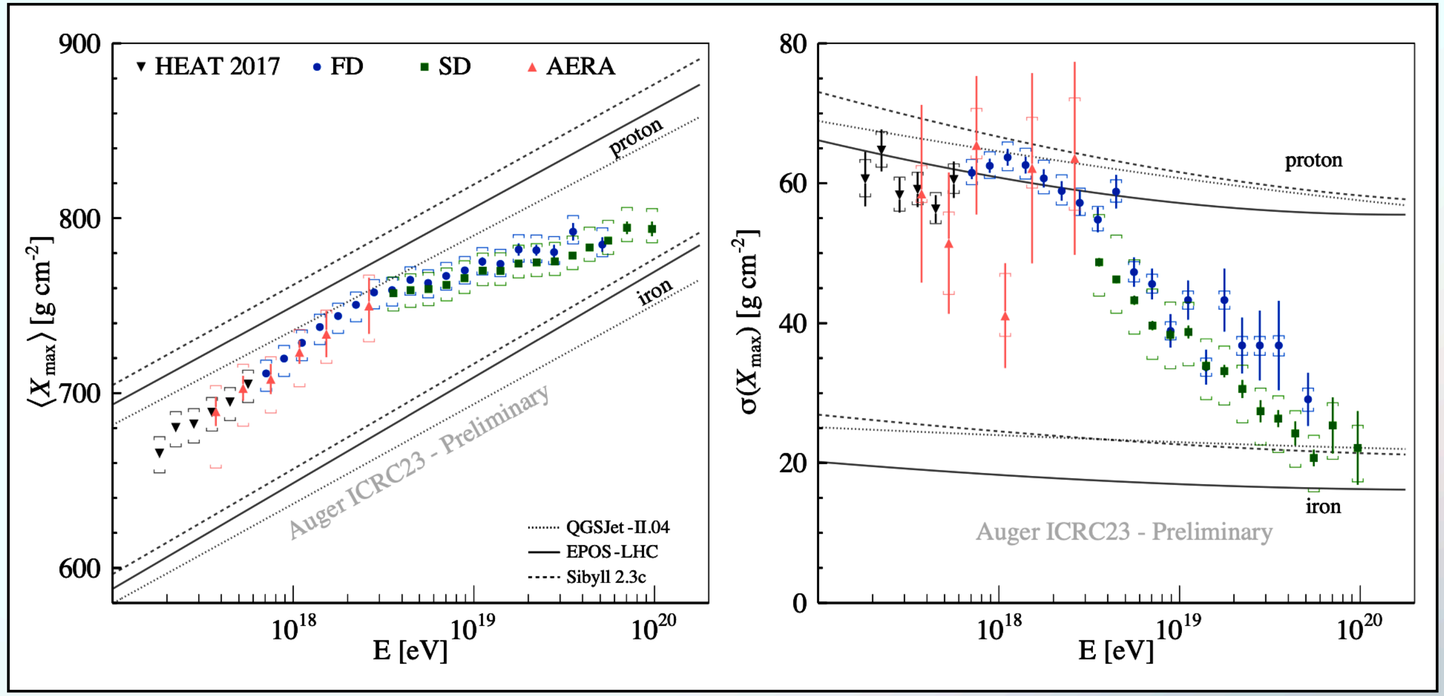


- Oltre 1600 Stazioni Water Cherenkov
- Oltre 1400 SSD



Misura della composizione di massa degli UHECRs





https://doi.org/10.22323/1.444.0365

X_{max} è un'osservabile sensibile alla massa

Evoluzione generale del trend di massa dalle misure dei primi due momenti di X_{max}

La composizione del flusso di UHECRs sopra i 100 PeV è descritta come:

- La grande maggioranza di UHECR primari sono **nuclei atomici ionizzati**.
- All'aumentare dell'energia, la massa media di questi nuclei prima decresce, raggiungendo il punto più leggero a 3 EeV, e dopo aumenta significativamente.

Misura della composizione di massa degli UHECRs

- Le misure ibride soffrono di bassa statistica (FD 15% duty cycle)

- Migliorare l'analisi alle più alte energie con i dati del Surface Detector (100% duty cycle).

- Metodi data-driven usando il tempo di salita medio dalle stazioni SD in un evento (Δ) collegato alla prossimità del massimo dello sciame da terra

Maximum Rigidity

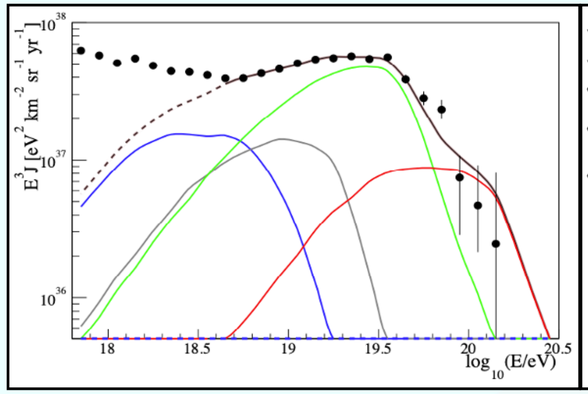
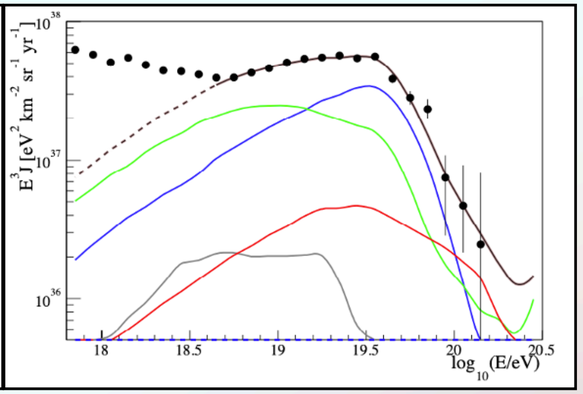
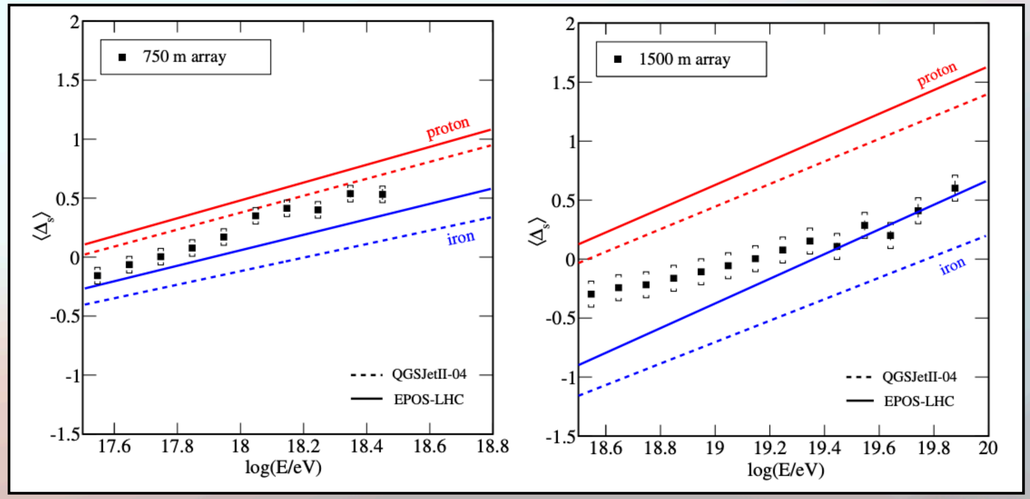


Photo-disintegration

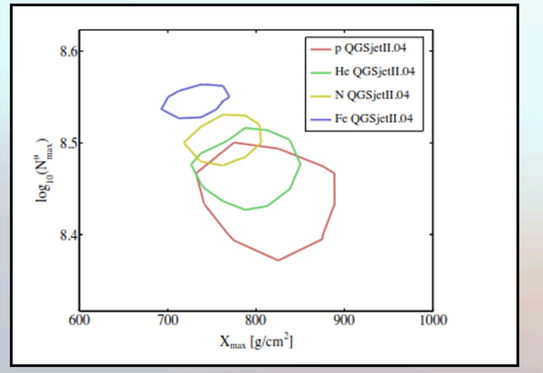




Estrazione delle componenti elettromagnetica e muonica in superficie

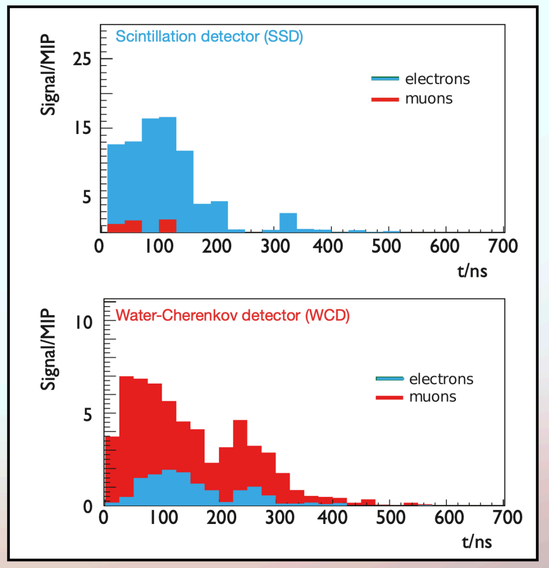
AugerPrime

- Due misure indipendenti (WCD + SSD)
- Algoritmi di Machine learning per estrarre il segnale muonico e stimare parametri correlati alla massa del primario

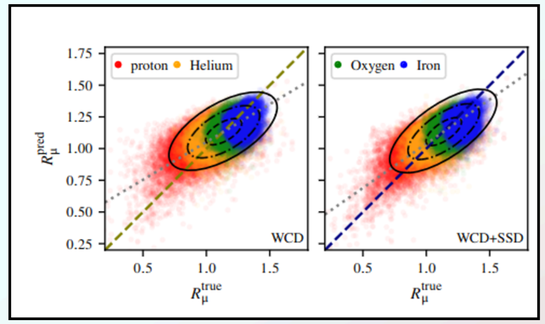


Matrix Inversion

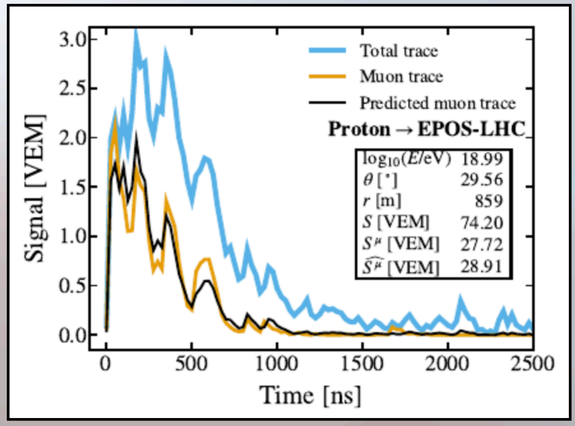
$$\mathcal{F}_{\rm em} = \frac{1}{\lambda-\beta}\left(\frac{S_{\rm SSD}}{\mathcal{A}_{\rm SSD}} - \frac{S_{\rm WCD}}{\mathcal{A}_{\rm WCD}}\right),$$
$$\mathcal{F}_{\mu} = \frac{1}{\lambda-\beta}\left(\lambda\frac{S_{\rm WCD}}{\mathcal{A}_{\rm WCD}} - \beta\frac{S_{\rm SSD}}{\mathcal{A}_{\rm SSD}}\right).$$



Supervised - DNN

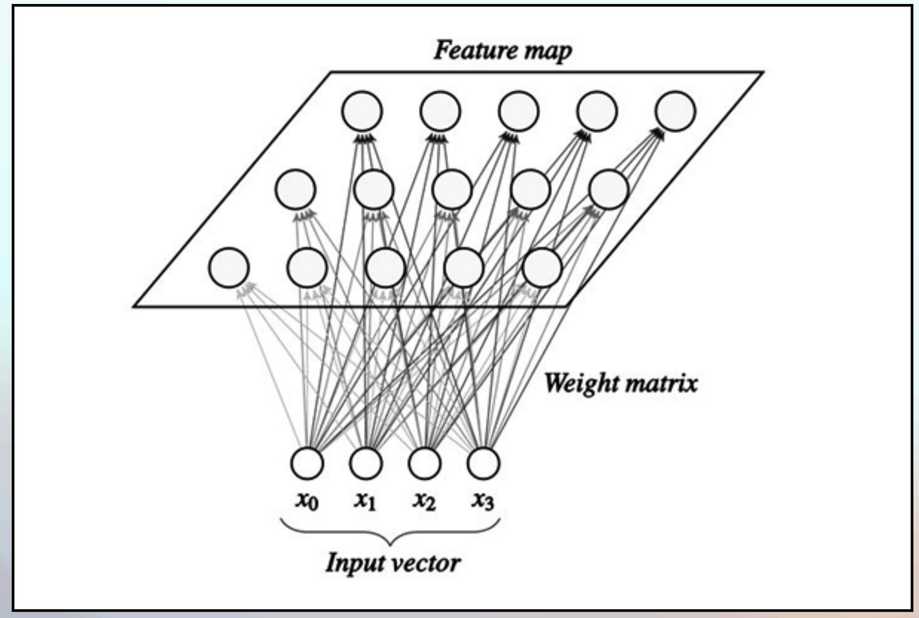


Supervised - RNN



Self Organizing Map

Una mappa auto-organizzante, o SOM, è un metodo di riduzione della dimensionali dei dati. Si tratta di una rete neurale ***non supervisionata*** per costruire una rappresentazione discretizzata a bassa dimensione dallo spazio di input dei campioni di addestramento



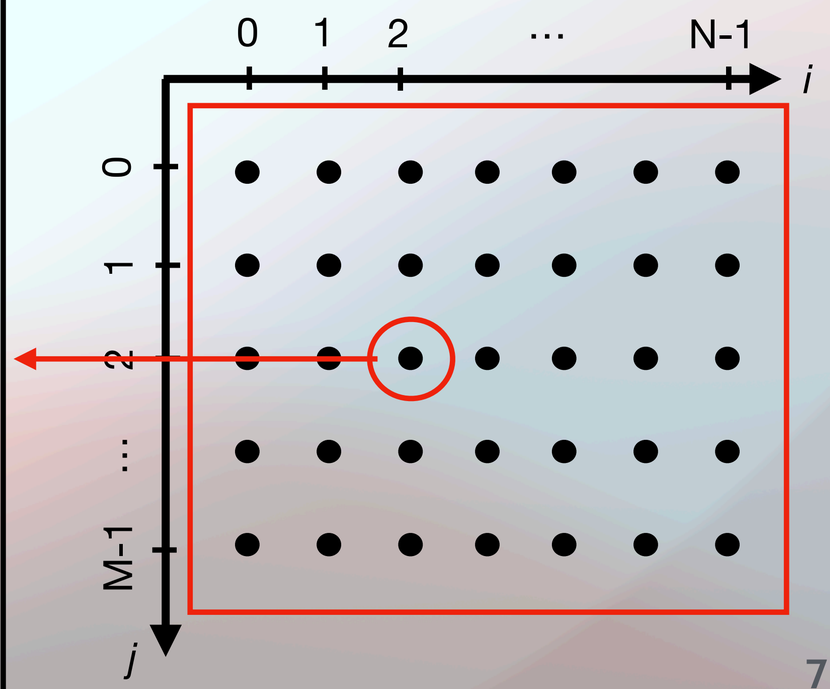
Q = N x M (neuroni)

Vettore di riferimento:

$$\vec{r}^{(i,j)} = [r_1^{(i,j)}, r_2^{(i,j)}, \ldots, r_k^{(i,j)}]$$

- *Neurone (i, j)*
- *Cardinalità k legata al vettore di input*

$$\vec{x} = [x_0, x_1, \ldots, x_k]$$



Addestramento

- I vettori di riferimento dei singoli neuroni vengono inizializzati con due possibili opzioni: '***pca***' o '***random***' initialization sui dati.

- Per ogni input $l \in [0, L-1]$ viene determinato un neurone vincitore, come il neurone (i,j) che minimizza la distanza euclidea: $D^l_{min} = \min_{(i,j)} \sum_{k=0}^{k=K-1} (r_k^{(i,j)} - x_k^l)^2$

- Definiamo una singola epoca t quando tutti gli input L sono stati elaborati dalla mappa

- Aggiornamento di tutti i vettori di riferimento: $\vec{r}^{(i,j)} \rightarrow \vec{r}'^{(i,j)}$ IPER-PAR: $[\alpha(\frac{t}{N_e}), \sigma(\frac{t}{N_e})]$

- Questo processo viene ripetuto per un numero selezionato di epoche (N_e) o opzionalmente fino al raggiungimento di una determinata condizione

Test preliminari sui MC per la caratterizzazione di f_μ

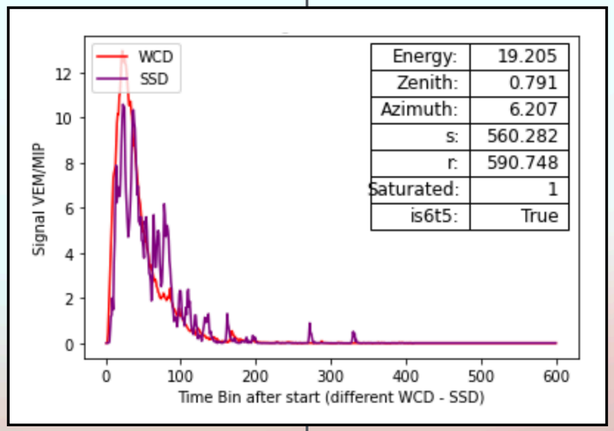
Sciami simulati con CORSIKA

- $\sim 5 \times 10^5$ segnali da eventi simulati
- Distribuiti uniformemente in logaritmo dell'energia (sopra $10^{18.5} eV$) direzione di arrivo e massa del primario (p, He, CNO, Fe)
- Ricostruzione con Offline

Test preliminari sui MC per la caratterizzazione di f_μ

Selezione Dataset 1 - Serie Temporali

- ***Media pesata dei segnali dei PMT nel WCD***
- Cardinalità:
  $k = 300$ bin ($\simeq 2.5\mu s$ time window) con la nuova e più veloce elettronica di acquisizione
- Normalizzazione dei segnali
- Cluster basati sulla forma del segnale



Selezione Dataset 2 - 'Meta'

Parametri ricostruiti dello sciame

- x_0 E : energia primario.
- x_1 r : distanza della stazione dall'asse dello sciame
- x_2 θ : zenith
- x_3 ψ : azimuth

Informazioni dai segnali della stazione

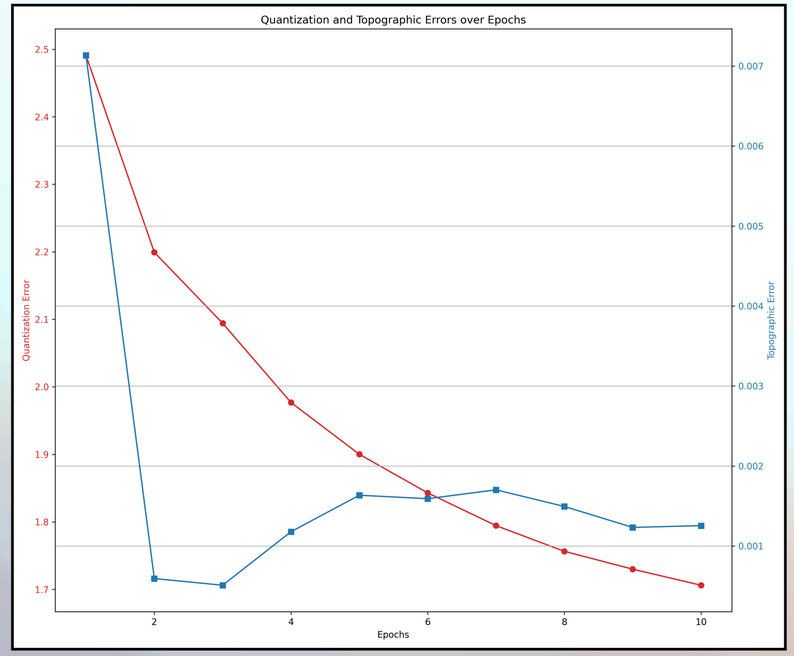
- x_4 S_{WCD} : segnale rilasciato in acqua [VEM].
- x_5 S_{SSD} : segnale rilasciato nello scintillatole [MIP]
- x_6 t_r : tempo di salita
- x_7 t_f : tempo di discesa.
- x_8 t_{50}: tempo al 50% del segnale totale
- x_9 A/peak: rapporto area-picco

Test preliminari sui MC per la caratterizzazione di f_μ

Dataset 1 - Serie Temporali

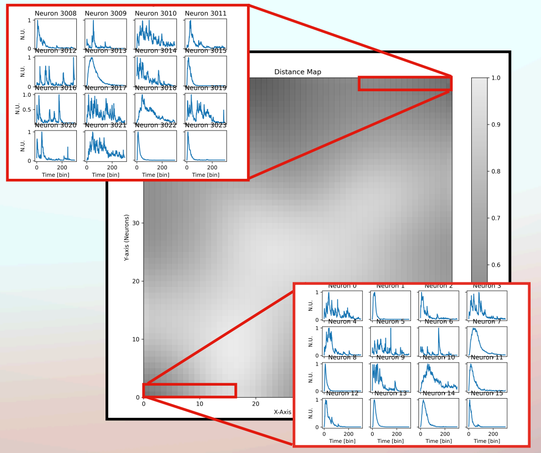
- Errori di **quantizzazione** e **topografico** in funzione del numero di epoche

- $L \simeq 3.7 \times 10^5$, $Q = \text{int}\ (5\sqrt{L}) = 3024$
- $\alpha_0 = 0.5, \sigma_0 = 10, N_e = 10$
- **Neuron Distance Map (grid:** 55×55**)**



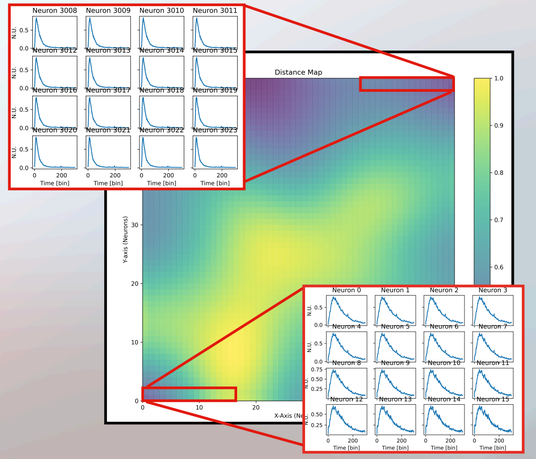
PRIMA DELL'ADDESTRAMENTO

Inizializzazione 'random'



DOPO L'ADDESTRAMENTO

Clustering di segnali simili in diverse zone della mappa

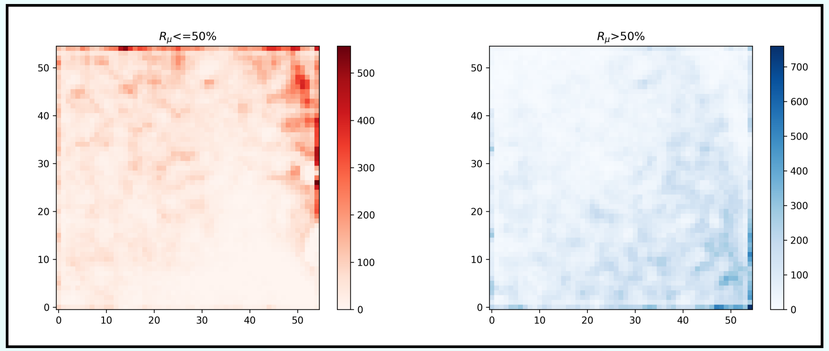


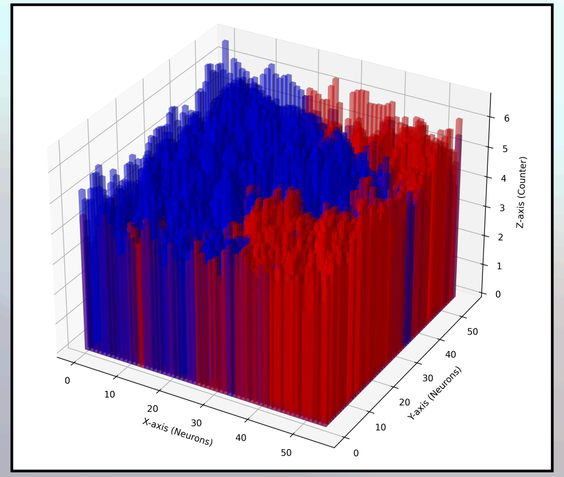
Test preliminari sui MC per la caratterizzazione di f_μ

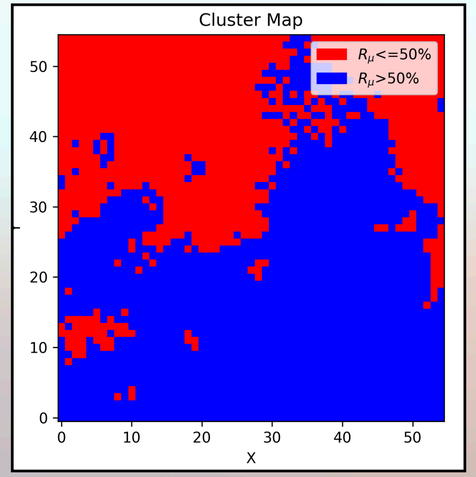
Dataset 1 - Serie Temporali

Clustering Binario

$\{f_\mu \leq 50\,\%\,, f_\mu > 50\%\}$

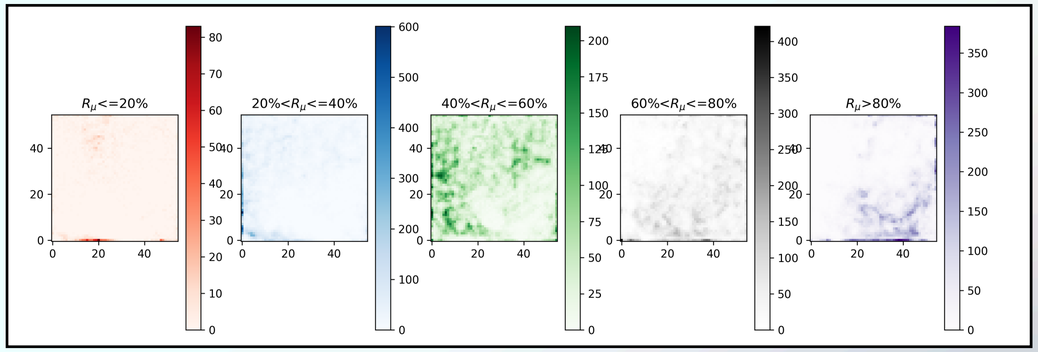


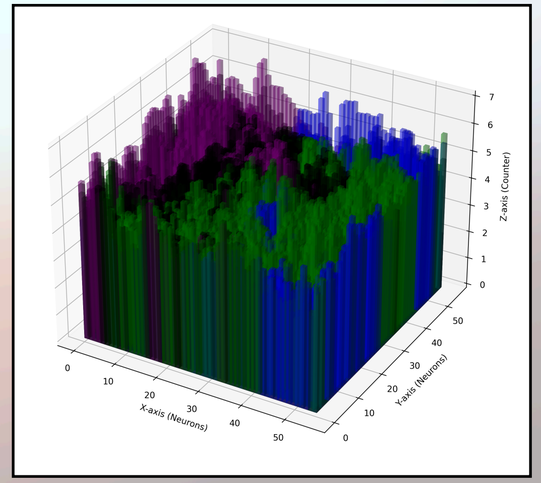


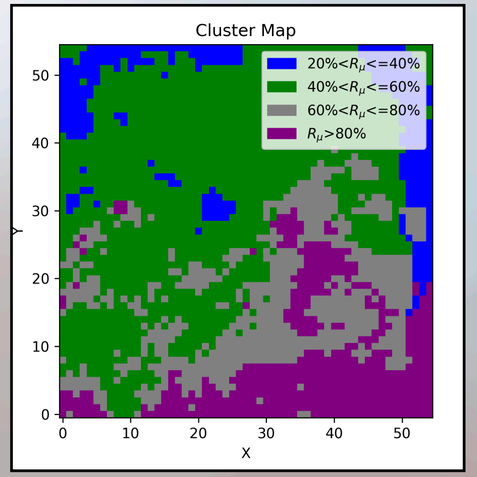


Clustering Multiclasse

$\{f_\mu \leq 20\,\%\,, 20\,\% < f_\mu \leq 40\,\%\,, 40\,\% < f_\mu \leq 60\,\%\,, 60\,\% < f_\mu \leq 80\,\%\,, f_\mu > 20\%\}$





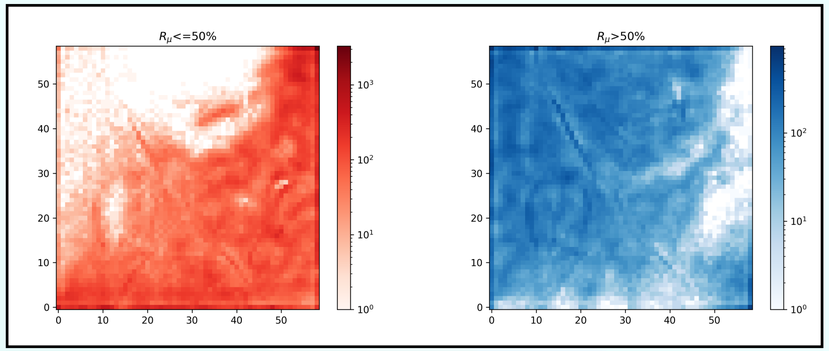


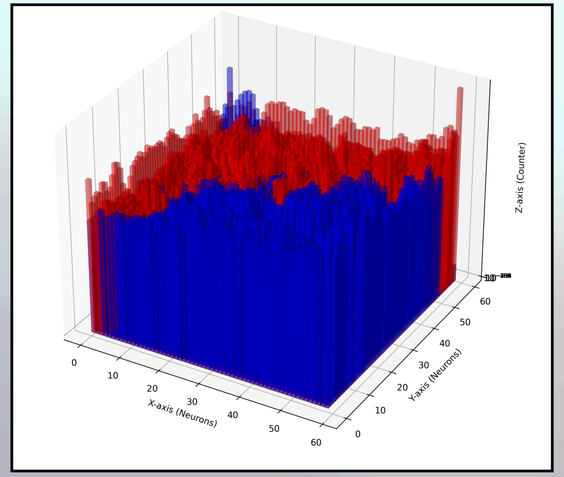
Test preliminari sui MC per la caratterizzazione di f_μ

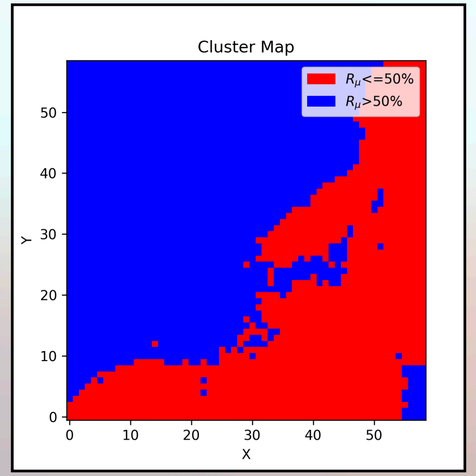
Dataset 2 - ‘Meta’

Clustering Binario

$\{f_\mu \leq 50\,\%\,, f_\mu > 50\%\}$

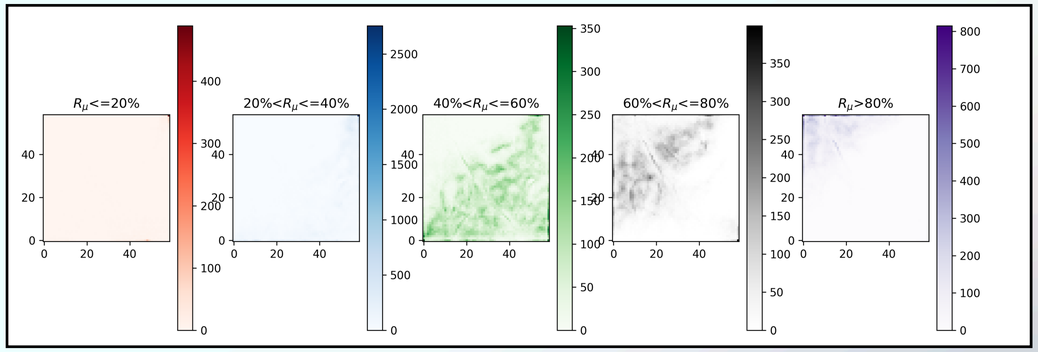


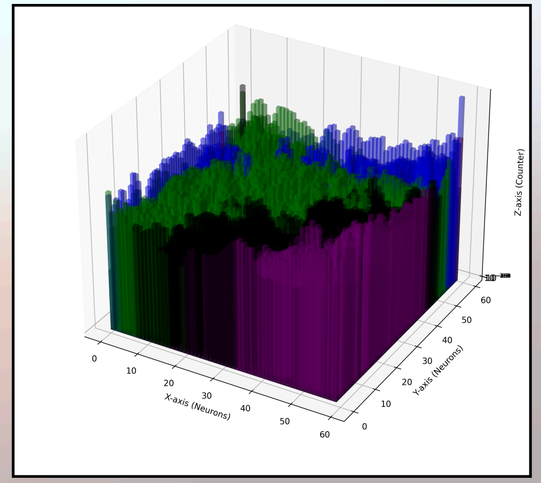


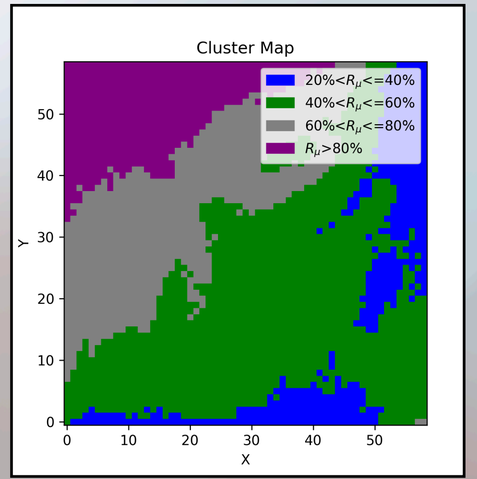


Clustering Multiclasse

$\{f_\mu \leq 20\,\%\,, 20\,\% < f_\mu \leq 40\,\%\,, 40\,\% < f_\mu \leq 60\,\%\,, 60\,\% < f_\mu \leq 80\,\%\,, f_\mu > 20\%\}$







Conclusioni

- Test preliminari dimostrano una maggiore efficacia per cluster di input ad alta dimensionalità rispetto ad algoritmi più semplici

- Risultati sui soli segnali WCD mancano delle informazioni aggiuntive legate ai parametri di ricostruzione dello sciame

Step Successivi

- Studio di input più adatti al clustering finalizzato allo studio della muon fraction

- Ricerca di strutture direttamente sui dati (simulations unbiased)

- Applicazione per classificazione ‘sciame per sciame’ della composizione di massa

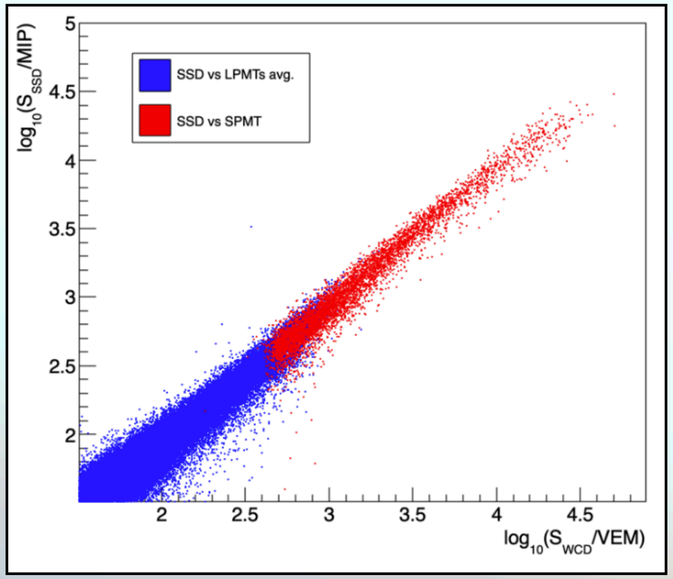
GRAZIE DELL'ATTENZIONE

BACKUP

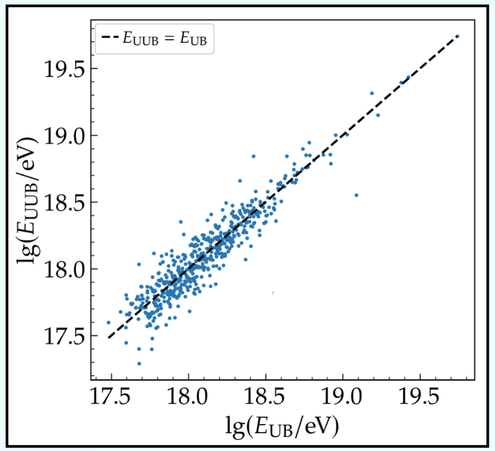
AugerPrime (Phase II)

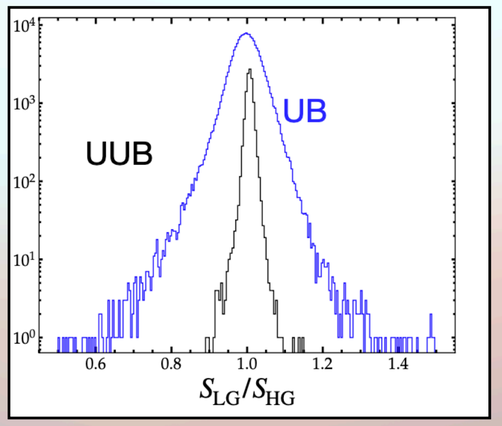
Performances

WCD dynamic range extension



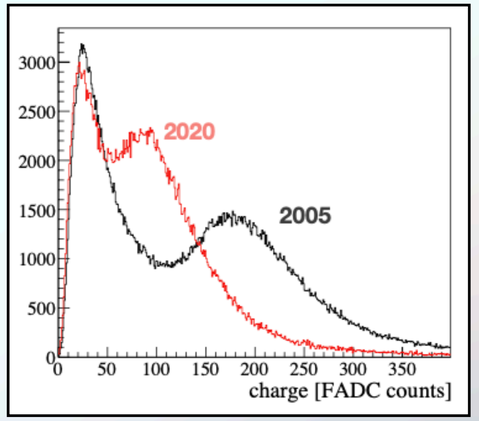
UB - UUB comparison

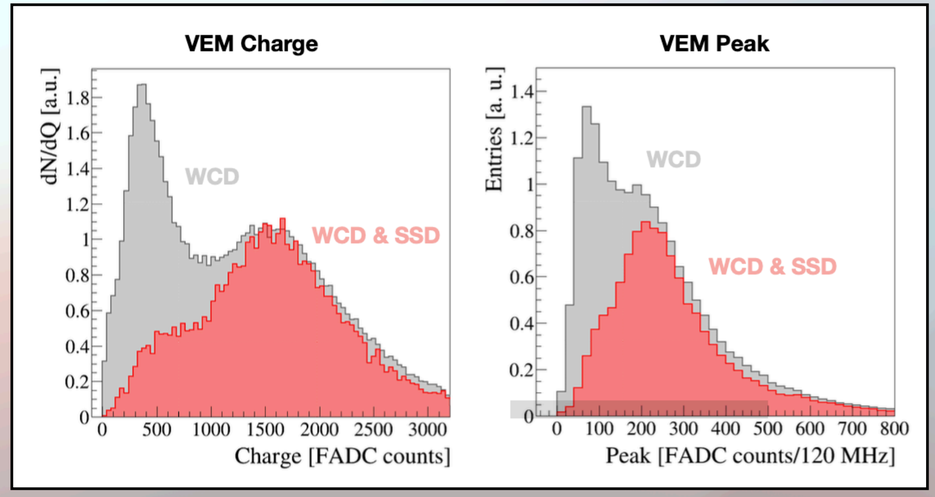




Calibration

L'aging del Water Cherenkov detector influisce sull'istogramma di calibrazione





Numero di neuroni

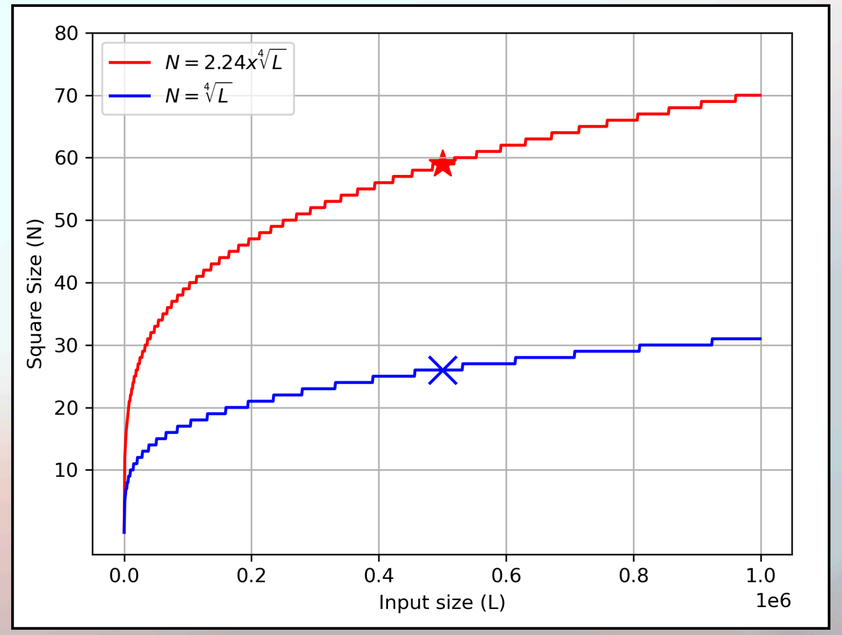
- **Q** non deve essere troppo grande, altrimenti la mappa risultante avrebbe un singolo neurone adattato per ogni dato di input

- **Q** non deve essere troppo piccolo, poiché una mappa troppo povera non riesce a cogliere un'adeguata organizzazione dei dati in classi separate

In genere: $Q = 5\sqrt{\text{len(dataset)}} = 5\sqrt{L}$

Per cui se si sceglie una griglia quadrata $N = M$

$$N = \sqrt{5\sqrt{L}} \simeq 2.24\sqrt[4]{L}$$



Aggiornamento pesi

Quando si definisce un neurone vincitore, tutti i pesi sono aggiornati come:

$$r_k^{\prime(i,j)} = r_k^{(i,j)} + \alpha(\frac{t}{N_e})H(\frac{t}{N_e}, \vec{d}_{min} - \vec{d}_{(i,j)})(x_k^l - r_k^{(i,j)})$$

$r_k^{\prime(i,j)}$: peso (i,j) Aggiornato all'input ***l-esimo***

$r_k^{(i,j)}$: peso (i,j) Aggiornato all'input ***(l-1)-esimo***

$(x_k^l - r_k^{(i,j)})$: *Differenza tra input **l-esimo** e **peso***

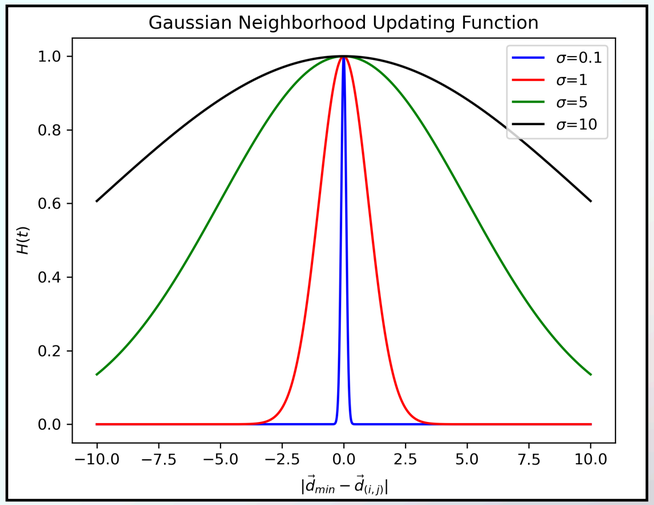
Aggiornamento pesi

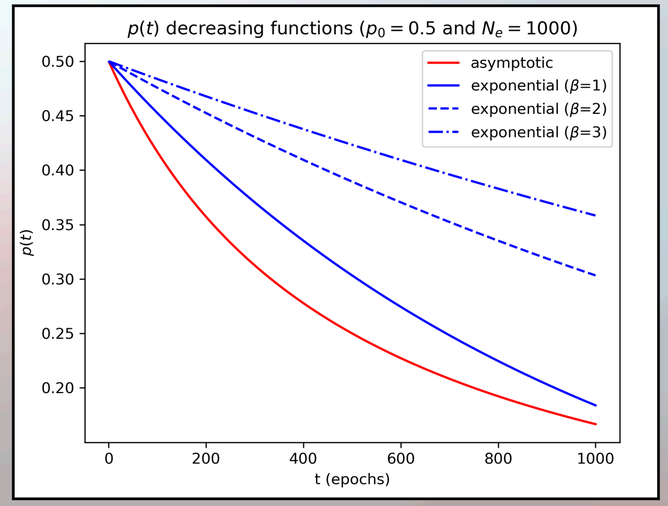
Quando si definisce un neurone vincitore, tutti i pesi sono aggiornati come:

$$r_k^{\prime(i,j)} = r_k^{(i,j)} + \alpha(\frac{t}{N_e})H(\frac{t}{N_e}, \vec{d}_{min} - \vec{d}_{(i,j)})(x_k^l - r_k^{(i,j)})$$

- **Learning Rate** α : iperparametro, funzione di N_e
- **Funzione di aggiornamento di vicinanza**:

$H(\frac{t}{N_e}, \vec{d}_{min} - \vec{d}_{(i,j)})$ in genere una *Gaussiana*: $\exp[-\frac{(\vec{d}_{min} - \vec{d}_{(i,j)})^2}{2\sigma^2(\frac{t}{N_e})}]$





Monitoraggio del training

Errori di quantizzazione e topografico

- A differenza dei modelli supervisionati, in cui possiamo minimizzare una funzione di perdita, qui possiamo monitorare il processo di "addestramento" tramite ***l'errore di quantizzazione*** definito come:

$$QE = \frac{\sum_{l=0}^{l=L-1} D_{min}^{l}}{L}$$

- Possiamo controllare anche ***l'errore topografico*** che definisce quanto è buona la topografia della mappa, per preservare la vicinanza tra neuroni simili:

$$TE = \frac{\text{N. casi in cui i primi due BMU non sono vicini}}{\text{N. casi totali}} \times 100\,\%$$

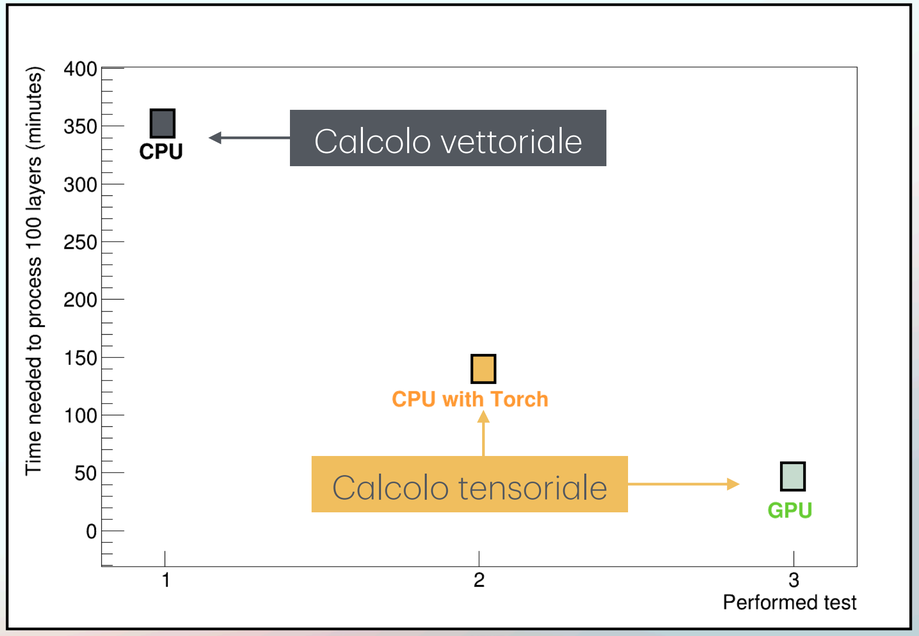
Implementazione di una libreria ottimizzata per lavoro su GPU

- Disponibili librerie basate su Numpy (calcolo vettoriale) delle Self Organizing Maps (SOM).

  https://github.com/JustGlowing/minisom.git

- Sviluppo di una libreria ottimizzata e più versatile per l'utilizzo di SOM in diversi contesti (fisici e non), basato sull'utilizzo di **pyTorch** per il calcolo tensoriale su GPU

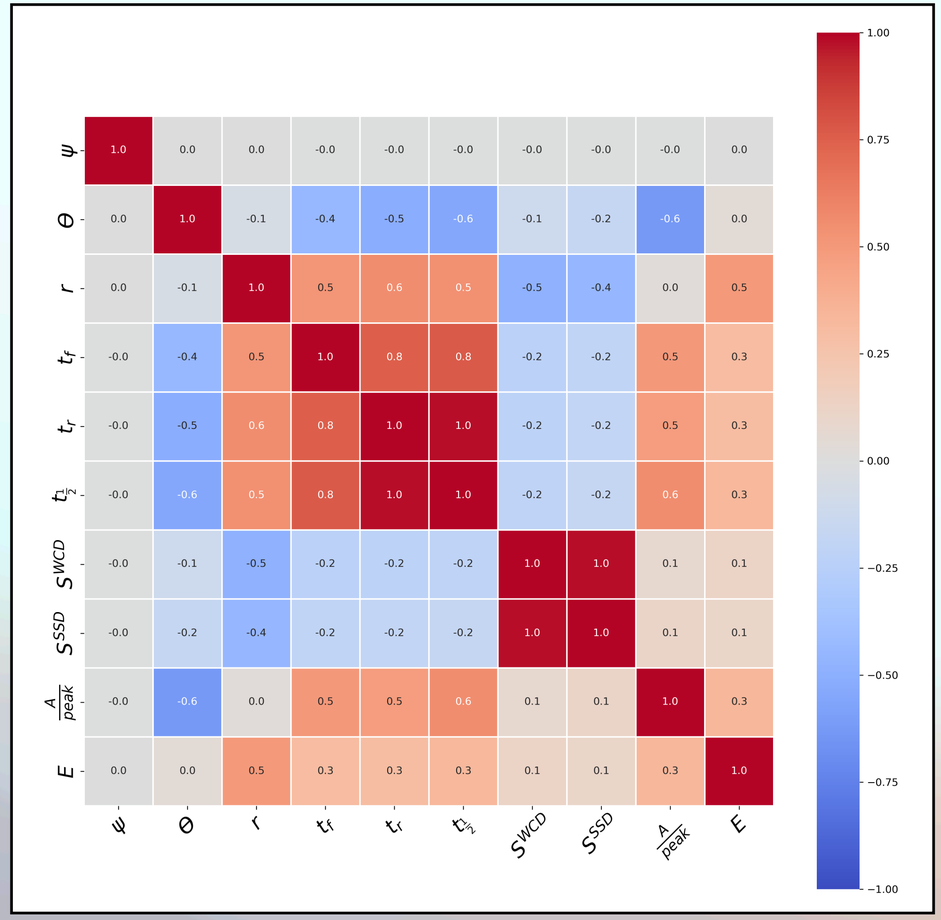
A disposizione:
2 GPU's TESLA v100 - 32 GB



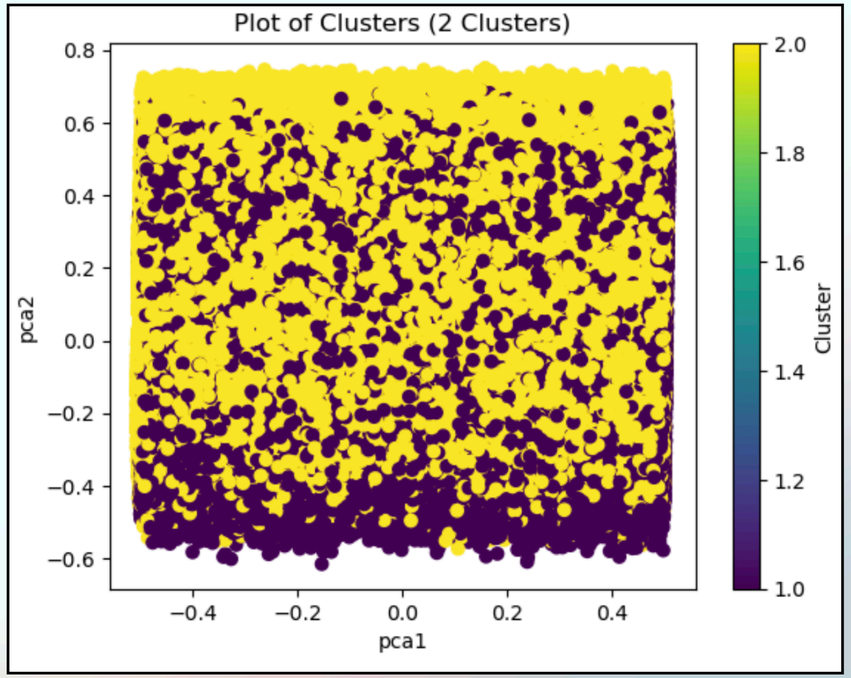
Fino a $\times 10$ volte più veloce con la nuova libreria su GPU

K-Means algorithm

Selezione dataset 2 - ‘Meta’



K=2 , 2 PCA Components



Score

```
Accuracy Score (Train): 0.19581108931292882
Accuracy Score (Test): 0.19556767714094825
```