Preliminary application of unsupervised Self-Organizing Maps for muon fraction characterization

Meeting of the Auger Italian Collaboration 3-5 February 2025

Matteo Conte[1,2], Daniele Martello[1,2], Gabriella Cataldi[2], Ugo Giaccari[2], Achille Nucita[1,2,3], Antonio Franco[2,3]

1 – Università del Salento, Lecce
2 – Istituto Nazionale di Fisica Nucleare, Lecce
3 – INAF, Lecce

UNIVERSITÀ DEL SALENTO

INFN Istituto Nazionale di Fisica Nucleare SEZIONE DI LECCE

PIERRE AUGER OBSERVATORY

OVERVIEW

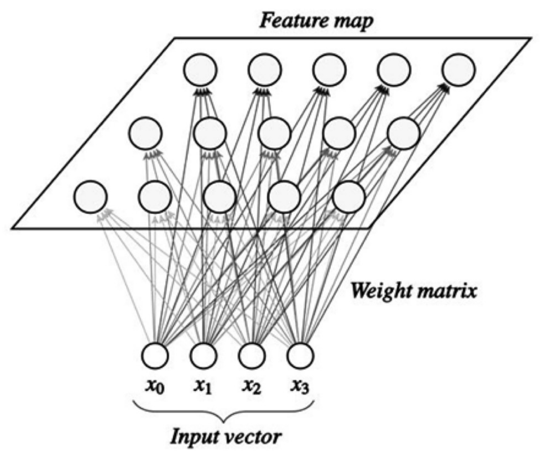
- Application
- Method
- Training
- Preliminary results
- Future Work

Application - f_μ clustering using Self-Organizing Map

- Find clusters and correlations from single station data:
  - WCD essential features from time traces $(signal, AoP, risetime, \Delta t_{50})$
  - SSD essential features from time traces $(signal, \Delta t_{integration}, peak)$ → under study
  - Local station geometry (r_{sh})
  - Global (reconstructed) feature from shower (E, θ, φ)
- Extract information on feature not used in training:
  - $f_\mu \equiv \frac{S_\mu}{S_{tot}}\,(\mathrm{WCD})$ → Know from MC simulations (**semi-supervised method**)
- Using a data dimensionality reduction method, applied on a large-scale dataset:
  - Specific training on GPU's
  - Application on test dataset and performance

Method

A **Self-Organizing Map**, or **SOM**, is a method of data dimensionality reduction. It involves an unsupervised neural network to construct a discretized low-dimensional representation from the input space of training samples.



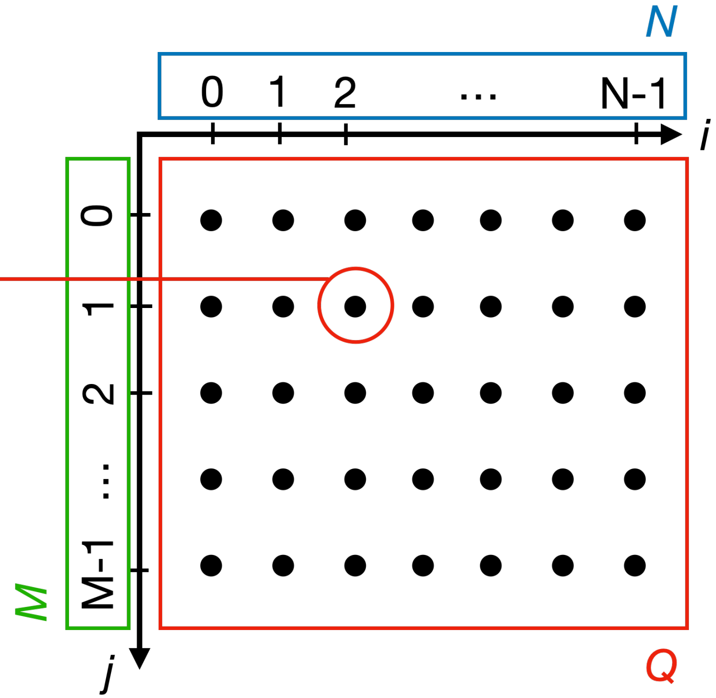
The SOM consists of a set of:

$Q = N \times M$ *neurons*

- Each neuron is identified by its position (*i, j*)
- Each neuron is characterized by a set of *k* values which make it unique and different from all the others and define the:

Reference (weight) VECTOR

$$\vec{r}_{(i,j)} = [r_0^{(i,j)}, r_1^{(i,j)}, \; \ldots \; r_{k-1}^{(i,j)}]$$



- Q must not be too large, otherwise the resulting map would have one single neurons adapted per each input data
- On the other hand a too poor map fails to catch an adequate organization of the data into separate classes

Training – 1/4

Determining the Winning Neuron

The coordinates of the *winning neuron* (i^l_{win}, j^l_{win}) for one of the input (say the l^{th} input) with $l \in [0, L-1]$ is found by minimizing the distance:

$$D^l_{min} = \min_{(i,j)} \sqrt{\sum_{k=0}^{K-1} m^l_k (r^{(i,j)}_k - w^l_k)}$$

Where:

- $\vec{w}_l = [w^l_0, w^l_1, \ldots w^l_{k-1}]$ is the l^{th} input
- $\vec{r}_{(i,j)} = [r^{(i,j)}_0, r^{(i,j)}_1, \ldots r^{(i,j)}_{k-1}]$ (*i,j*) neuron reference vector
- k is the vectors cardinality (both $\vec{w}_l$ and $\vec{r}_{(i,j)}$)
- $m^l_k = \begin{cases} 0 & \text{if } w^l_k = \text{NaN} \\ 1 & \text{elsewhere} \end{cases}$ is a mask to handle missing or corrupted data

Training – 2/4

Weight Vectors Update

When a winning neuron is determined, the neurons weights are updated:

$$r_k'^{(i,j)} = r_k^{(i,j)} + \alpha(\frac{t}{N_e})H(\frac{t}{N_e}, \vec{d}_{min} - \vec{d}_{(i,j)})(w_k^l - r_k^{(i,j)})$$

where:

- t is the current epoch
- N_e is the total number of epochs
- $\vec{d}_{min}$ is the position of the winning neuron in the map
- $\vec{d}_{(i,j)}$ is the position of the current (*i,j*) neuron

Training – 3/4

Neighborhood updating function

When a winning neuron is determined, the neurons weights are updated:

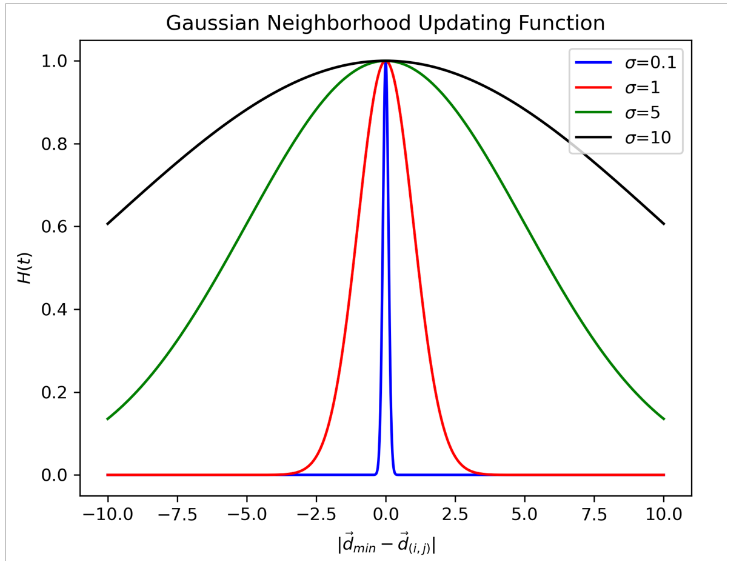
$$r'^{(i,j)}_k = r^{(i,j)}_k + \boxed{\alpha(\frac{t}{N_e})} H(\frac{t}{N_e}, \vec{d}_{min} - \vec{d}_{(i,j)})(w^l_k - r^{(i,j)}_k)$$

where:

- $H(\frac{t}{N_e}, \vec{d}_{min} - \vec{d}_{(i,j)})$ is the *neighborhood updating function*

  usually chosen to be a Gaussian:

$$H(\frac{t}{N_e}, \vec{d}_{min} - \vec{d}_{(i,j)}) = \exp[-\frac{(\vec{d}_{min} - \vec{d}_{(i,j)})^2}{\boxed{2\sigma^2(\frac{t}{N_e})}}]$$



Training – 4/4

Decreasing Function – Hyperparameters time evolution

Both $\sigma(\frac{t}{N_e})$ and $\alpha(\frac{t}{N_e})$ are decreasing functions with the number of epochs

Asymptotic decay (default)

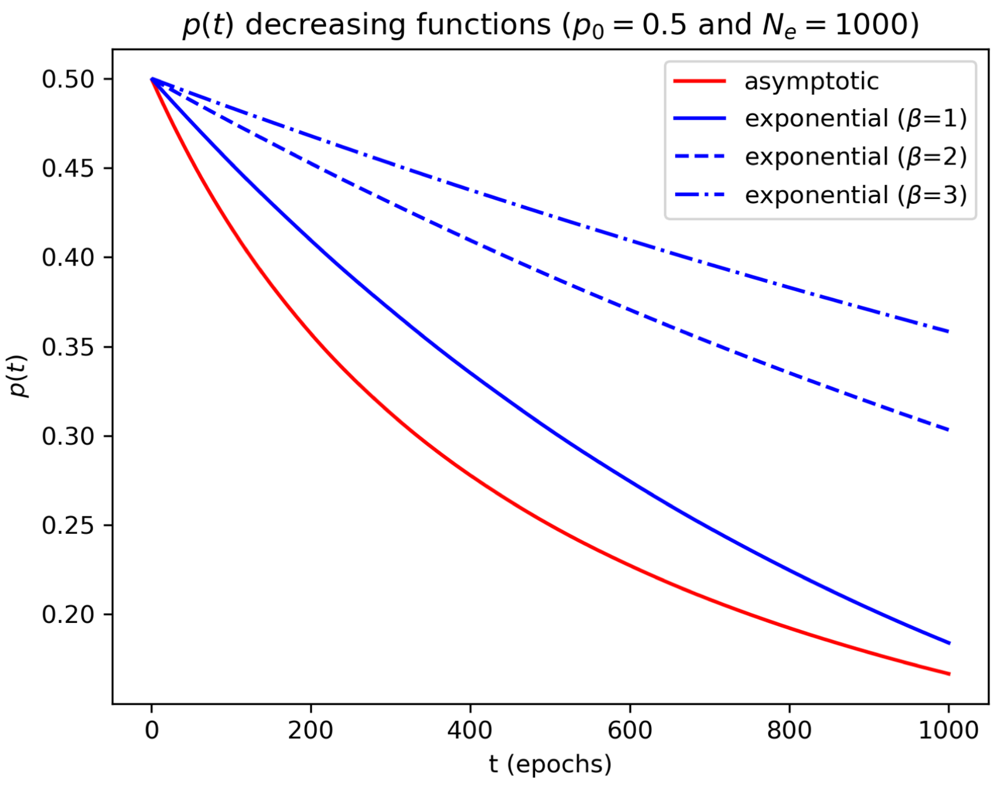
$$\alpha(\frac{t}{N_e}) = \frac{\alpha_0}{(1 + \frac{2t}{N_e})}$$

$$\sigma(\frac{t}{N_e}) = \frac{\sigma_0}{(1 + \frac{2t}{N_e})}$$

Negative Exponential

$$\sigma(\frac{t}{N_e}) = \sigma_0 \times exp(-\frac{t}{\beta N_e})$$

$$\alpha(\frac{t}{N_e}) = \alpha_0 \times exp(-\frac{t}{\beta N_e})$$

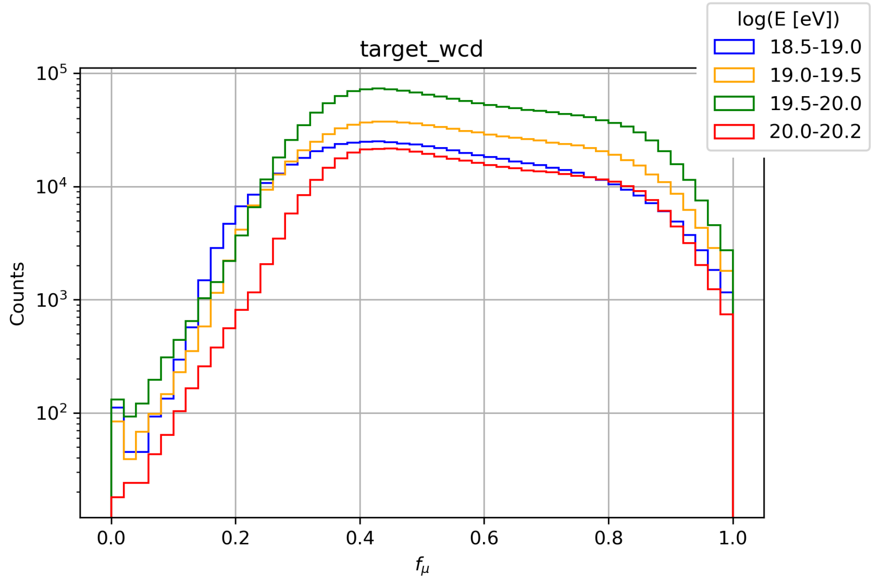


Data Selection – MC simulations

icrc-2023 / EPOS-LHC interaction model

SIMULATED SHOWERS:
- *p, He, O, Fe*
- $\log(E\ /eV)$ in [18.5, 20.2]
- minRecLevel 3, 6T5
- θ up to 60°
- Candidate stations:
  - $s_{WCD}/VEM > 5$
  - $s_{SSD}/MIP > 10$
- Excluding LG saturation

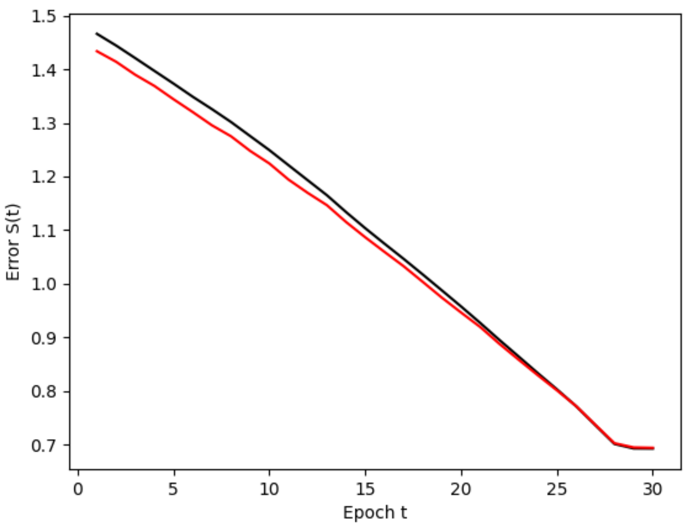
DATASET:
- Lenght: Over 3.6 million of inputs
- Standardization: $(x - \mu)/\sigma$
- Subset splits:
  - TRAINING: $\sim 42\%$
  - VALIDATION: $\sim 5\% \rightarrow$ *Early Stopping*
  - PROBABILITY MAPS: $\sim 27\%$
  - TEST: $\sim 26\%$



The distribution of the target variable is not uniform across the phase space but exhibits an intrinsic deficit at the edges. This affects the network's predictive ability in accurately characterizing this class of data.

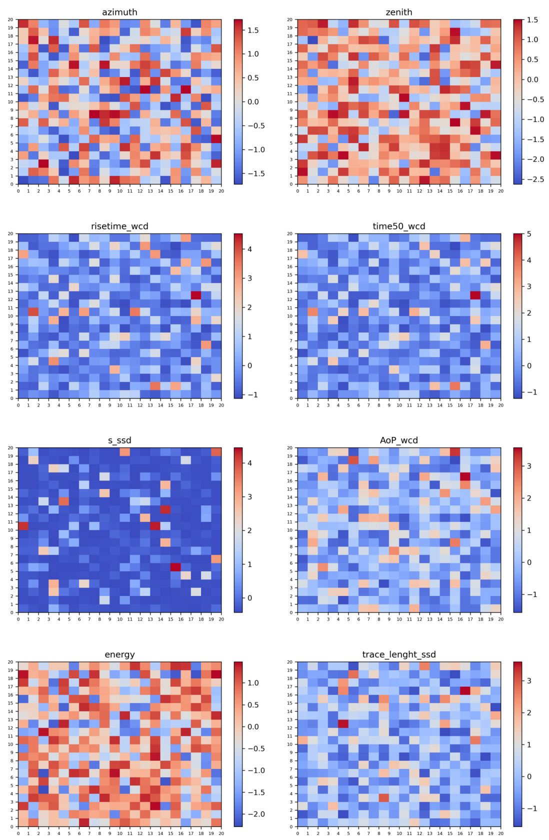
Training Results

Quantization Error vs Epoch

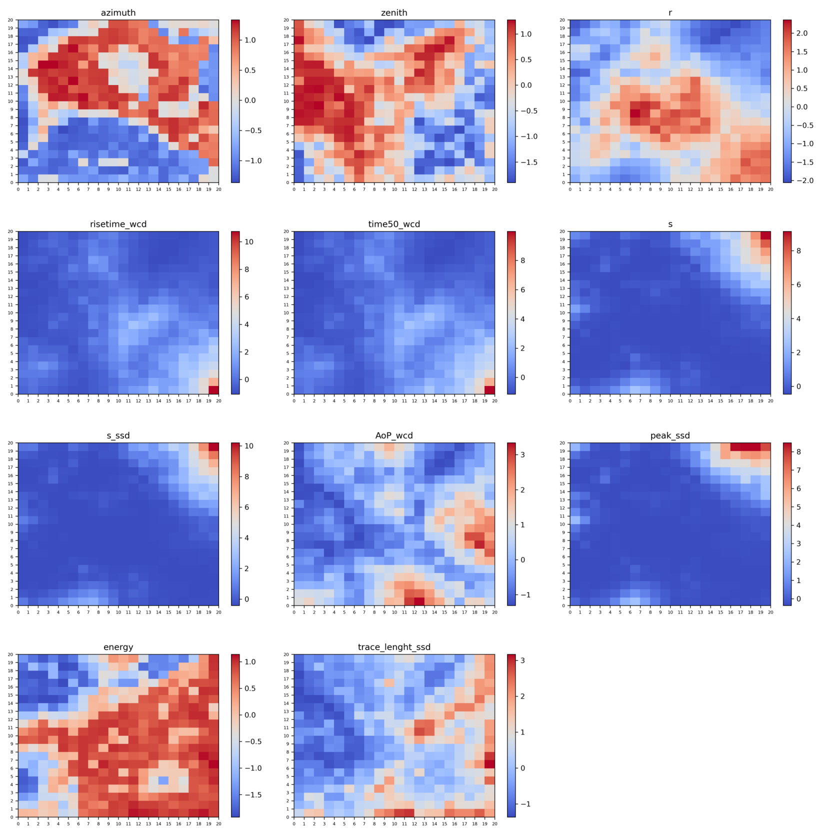


- ➢ $Q.E. = \frac{\sum_{l=0}^{L-1} D_{min}^{l}}{L}$
- ➢ Early stopping when convergence reached

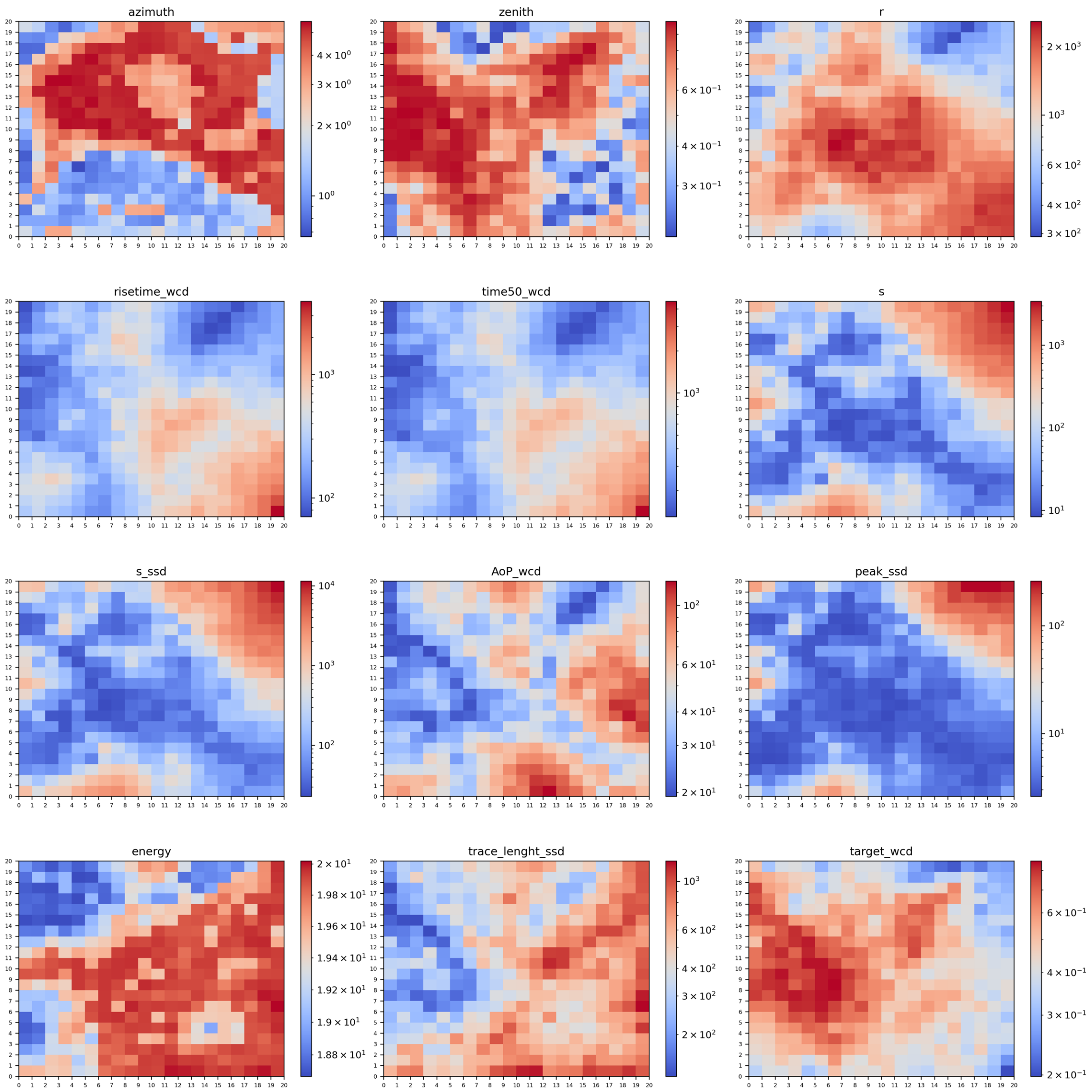
WEIGHTS MAPS before training



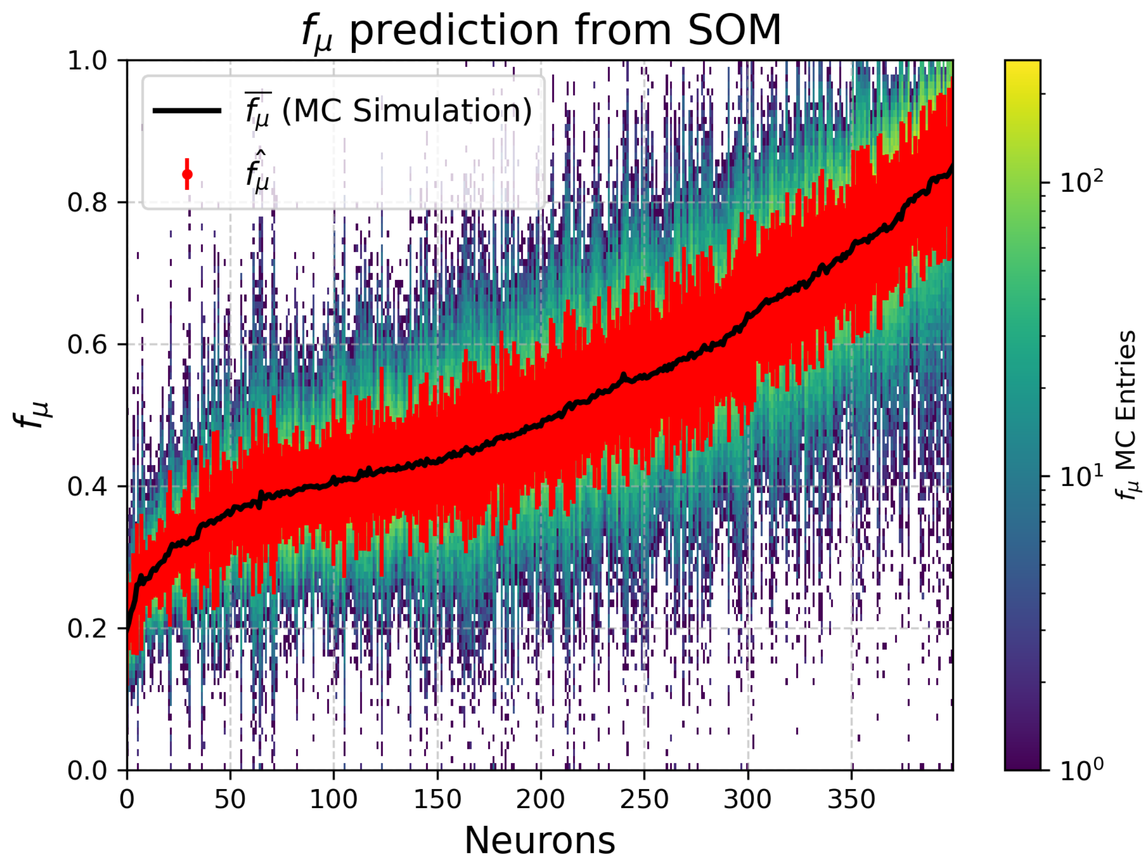
WEIGHTS MAPS after training



Features MAPS / Predictions on new data



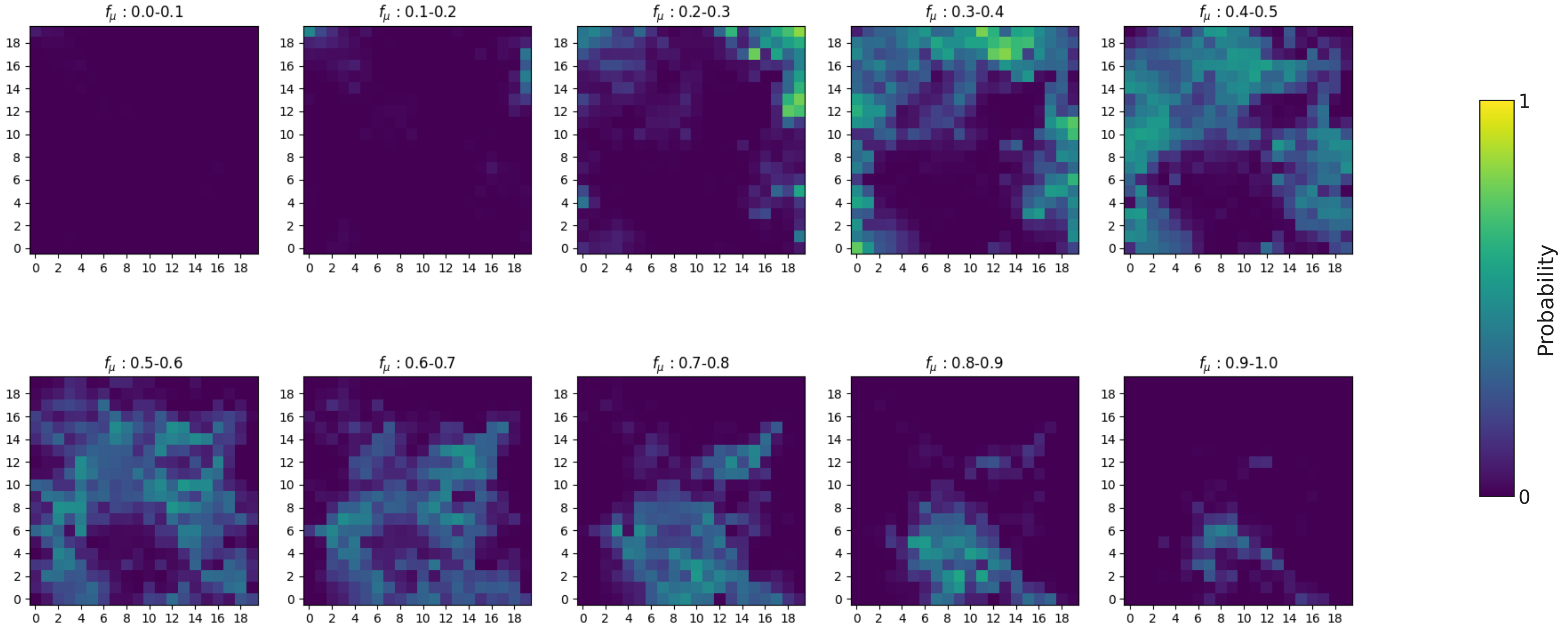
Performance on a new set of data (~1 milion)



Probability MAPS

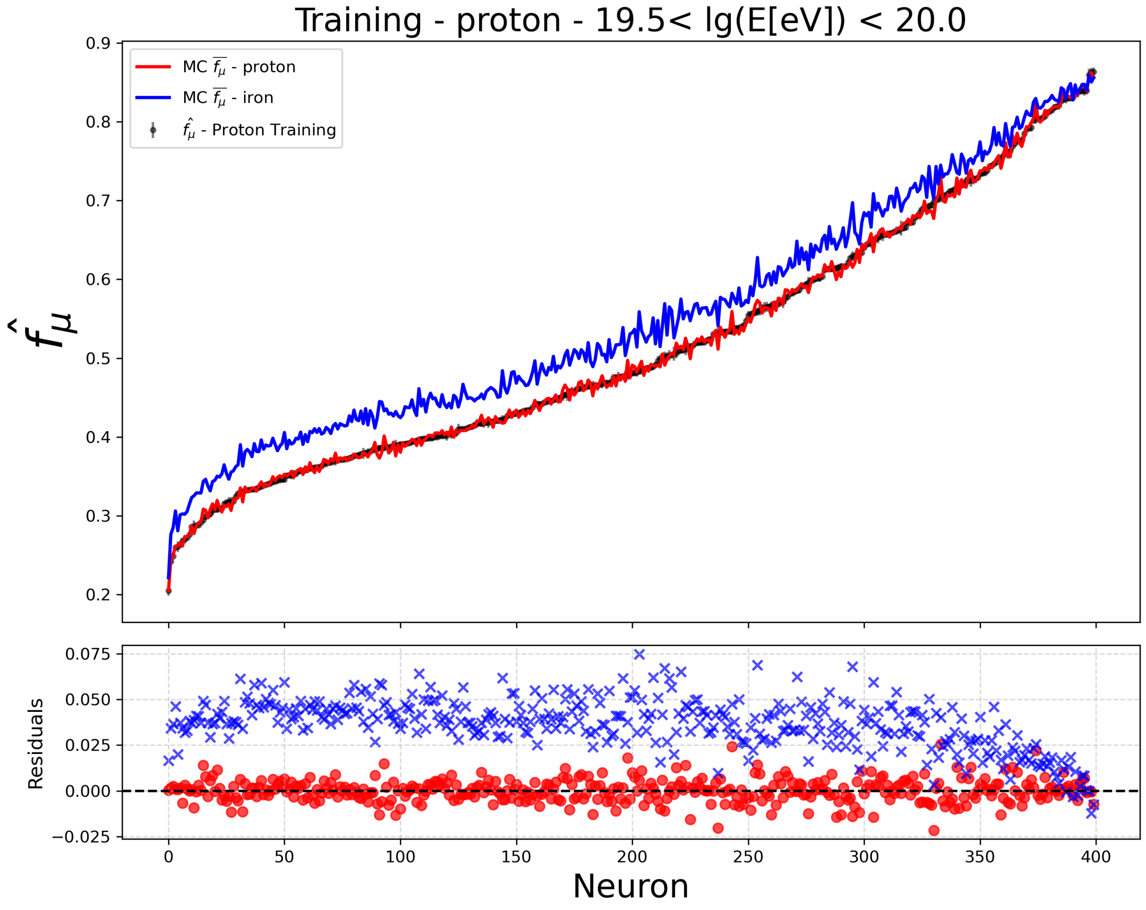
Probability Maps $f_\mu \subset [0.0, 1.0]$ binned in 10 classes

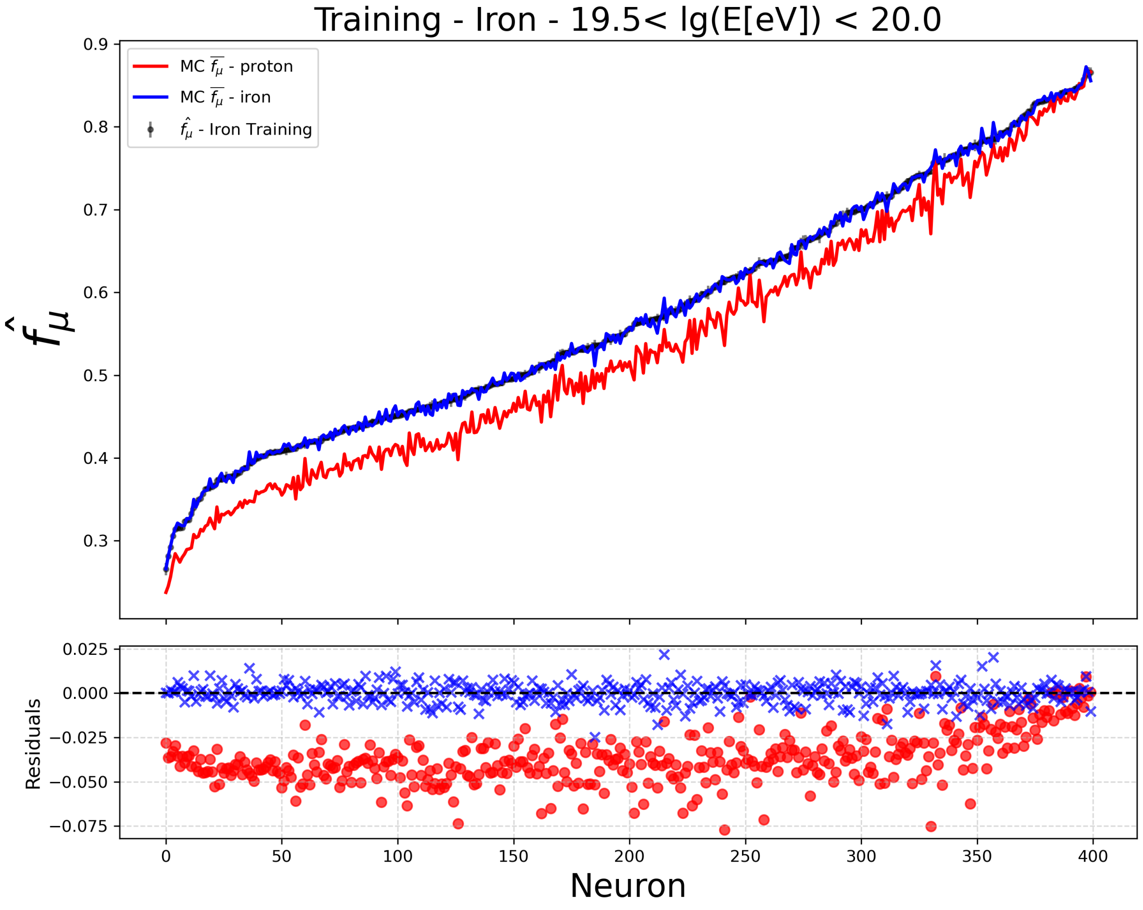
(a discrete pdf per single neuron)



Further tests – Energy and mass dependence

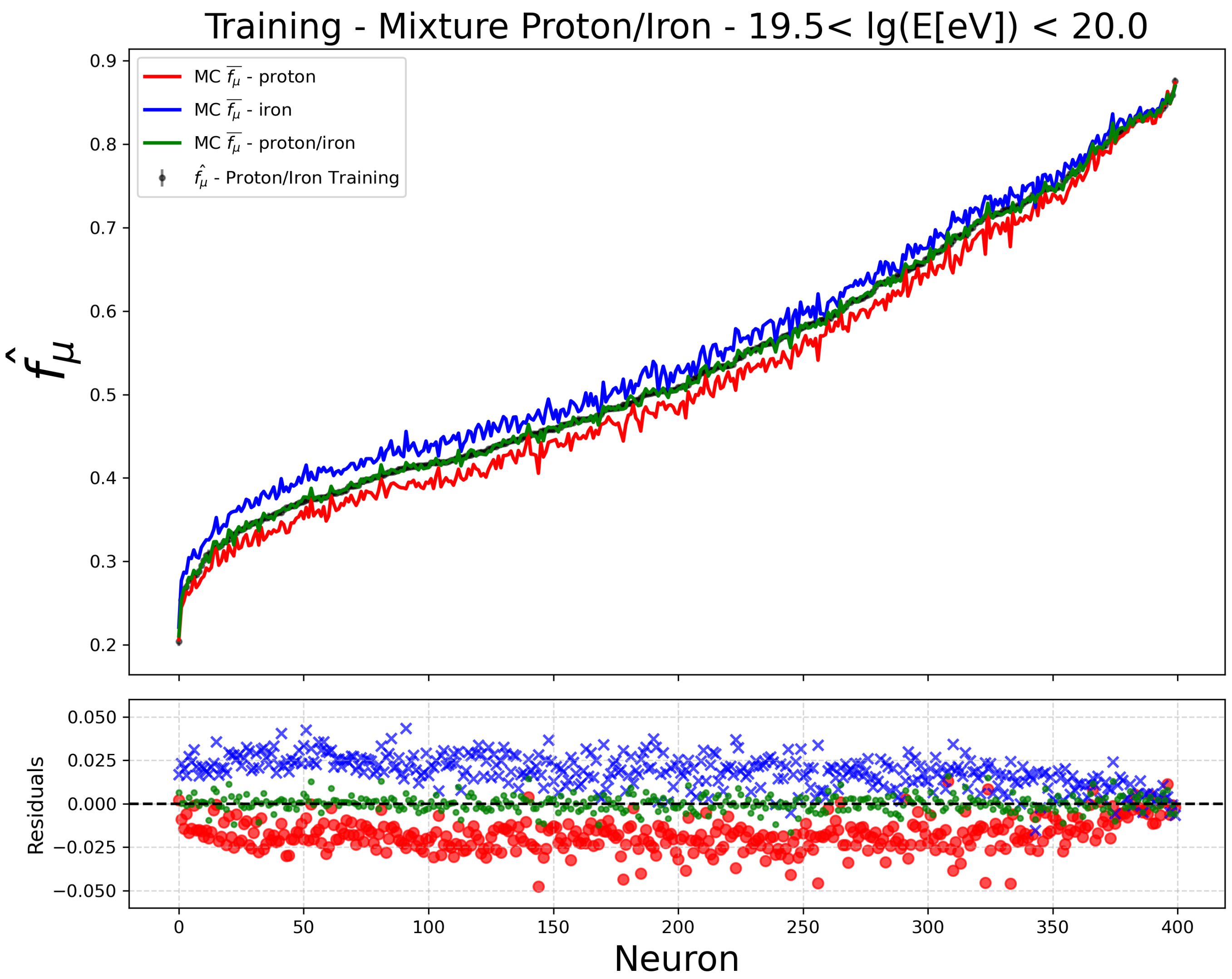
- $19.5 < \log(E/eV) < 20.0$
- Two different training on *proton* and *iron* subsamples





Further tests – Energy and mass dependence

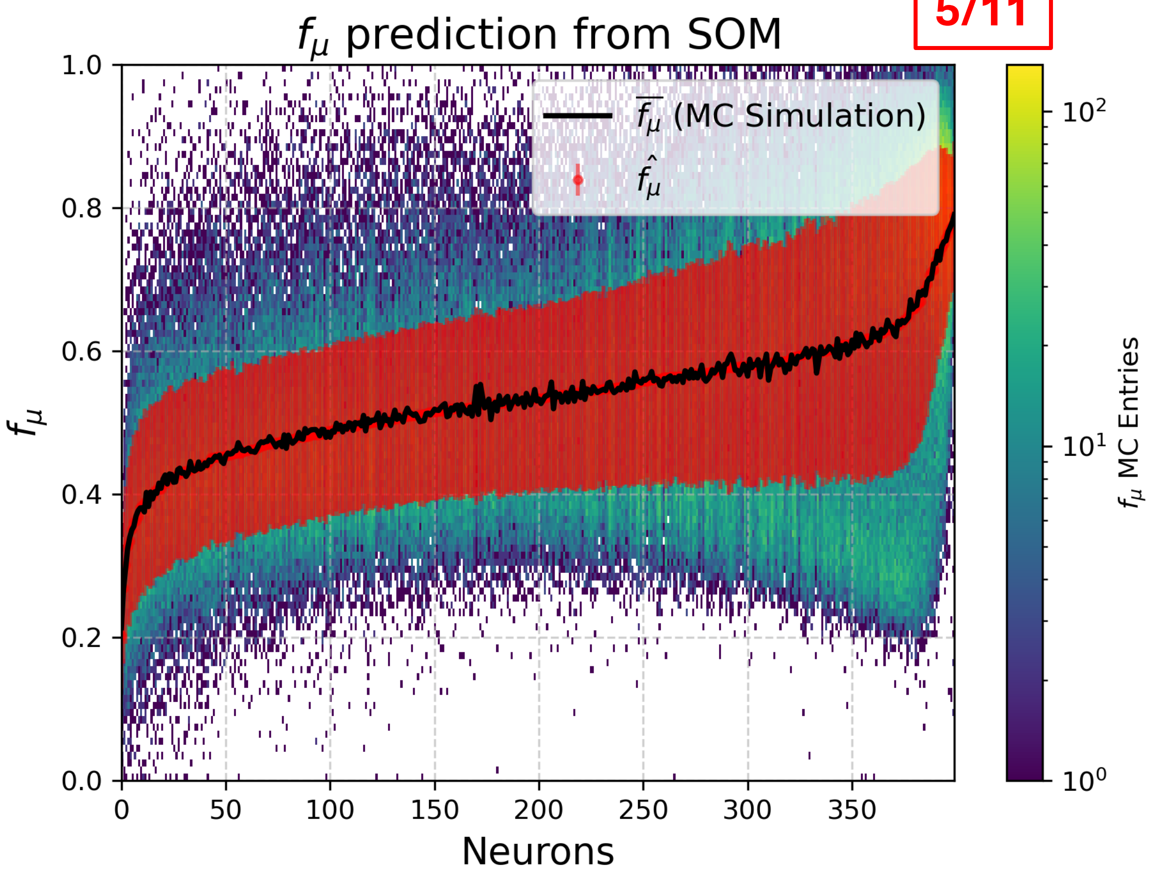
- $19.5 < \log(E/eV) < 20.0$
- Training on a *mixture* of *proton* and *iron* subsamples
- Fixed bias:
  - around -0.018 for proton-induced showers
  - around $+0.019$ for iron-induced showers
- Effect mitigated at the edge of phase space



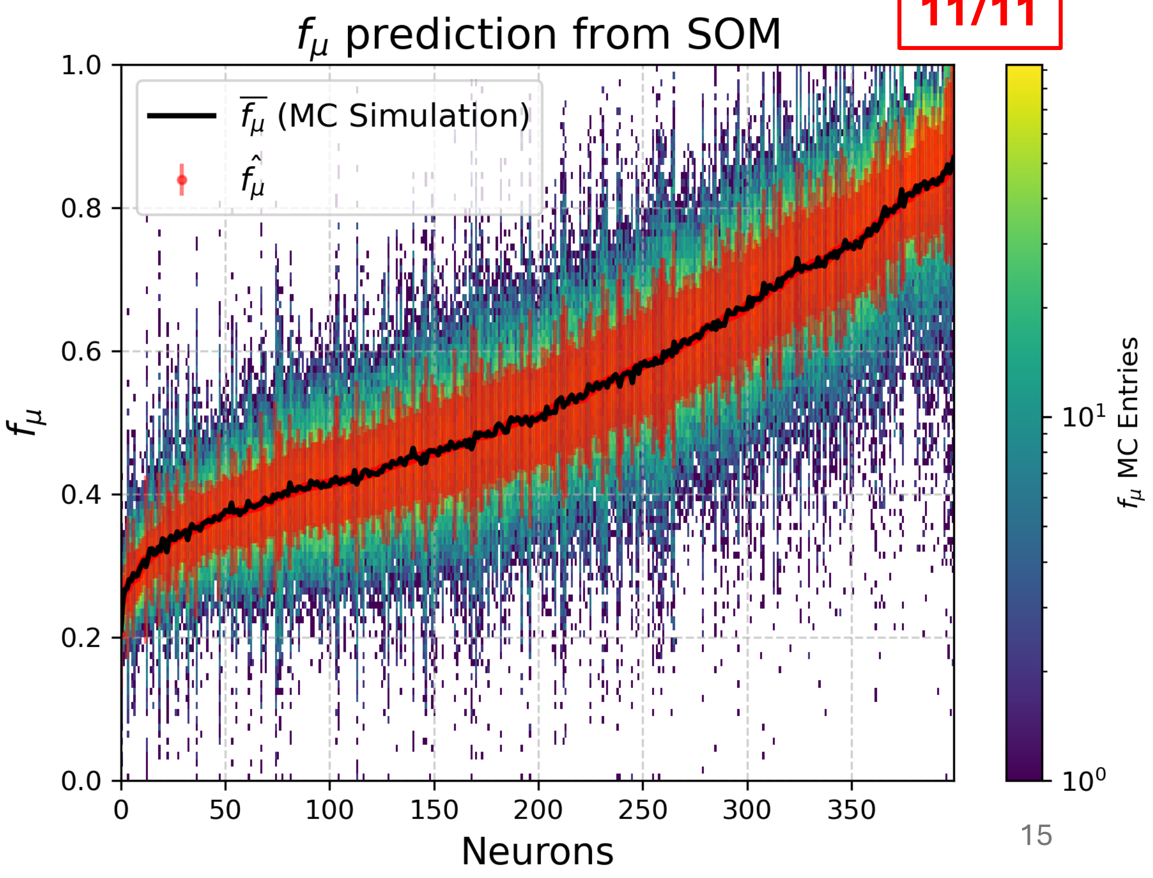
Dependence on Feature selection

To illustrate the importance of feature selection when passing inputs to the network for the characterization of the muonic fraction, we show the difference in training results on the proton/iron mix when using only 5 out of the 11 features compared to using all of them.
In the first case, we lose predictive capability, with a significant increase in statistical uncertainty and an almost uniform distribution of the dataset.

5/11



11/11



Summary

- The use of Self-Organizing Maps (SOM) enables training a network that effectively captures features and correlations within the data.
- The trained SOM accurately reproduces these features on previously unseen data, demonstrating strong generalization capabilities.
- The model allows for the estimation of a feature that was not included during training with a certain degree of precision.
- The selection of input data significantly influences the overall performance and accuracy of the model.

Next Step

- Conduct new tests to further investigate the dependence of the method on energy, primary mass and interaction model.
- Explore different combinations in the hyperparameter space
- Perform additional tests by evaluating extra **SSD features to improve precision and reduce the model's statistical uncertainty**.
- Apply the method by leveraging the full signal from both the WCD and SSD, and potentially the entire Auger SD event, similar to image classification approaches.
- Application and comparison with SD-Phase II data

Backup slides

Data Selection – MC simulations

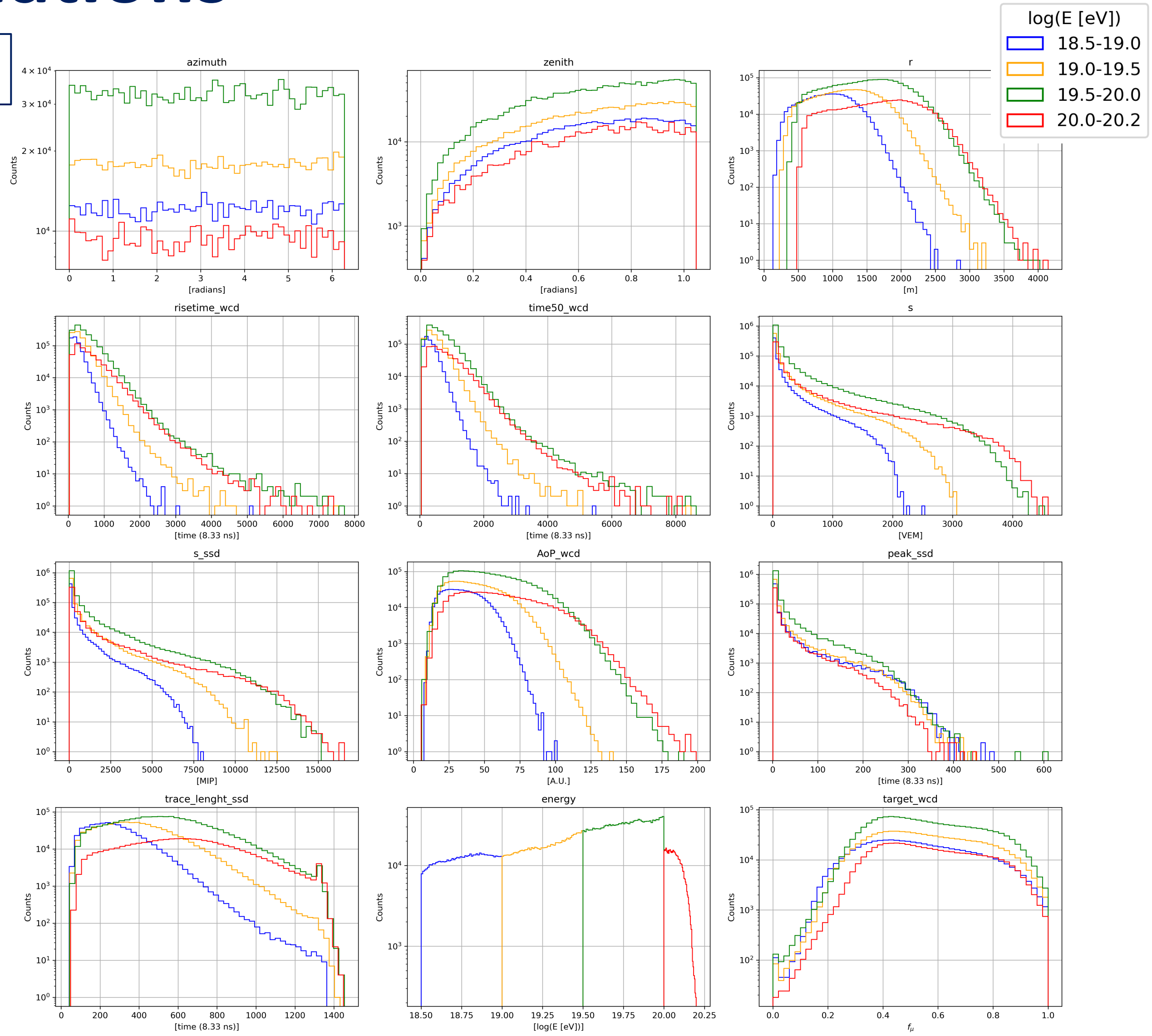
icrc-2023 / EPOS-LHC interaction model

SIMULATED SHOWERS:

- *p, He, O, Fe*
- $\log(E\ /eV)$ in [18.5, 20.2]
- minRecLevel 3, 6T5
- θ up to 60°
- Candidate stations:
  - $s_{WCD}/VEM > 5$
  - $s_{SSD}/MIP > 10$
- Excluding LG saturation

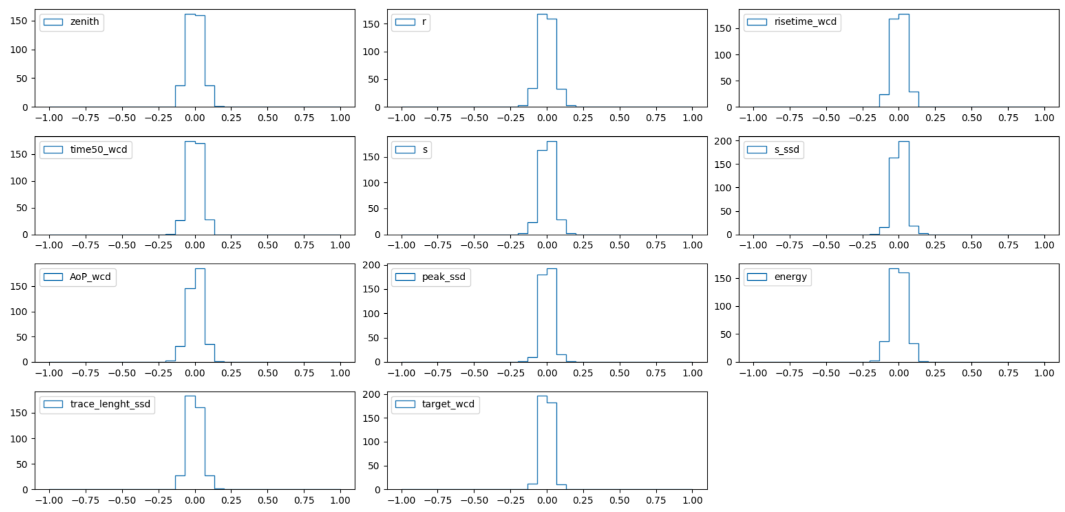
DATASET:

- Lenght: Over 3.6 million of inputs
- Standardization: $(x - \mu)/\sigma$
- Subset splits:
  - TRAINING: $\sim 42\%$
  - VALIDATION: $\sim 5\% \rightarrow$ *Early Stopping*
  - PROBABILITY MAPS: $\sim 27\%$
  - TEST: $\sim 26\%$

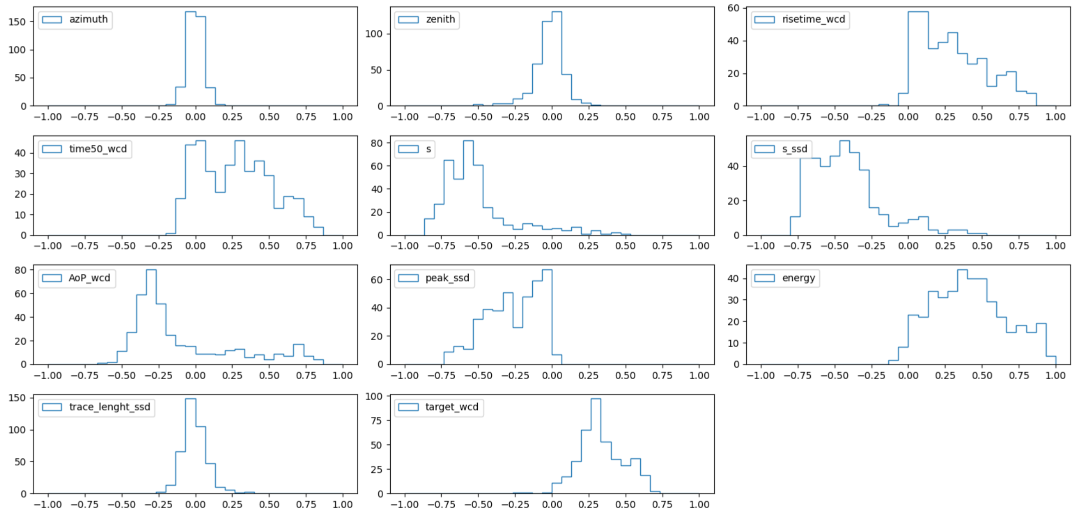


Correlations in the features MAP

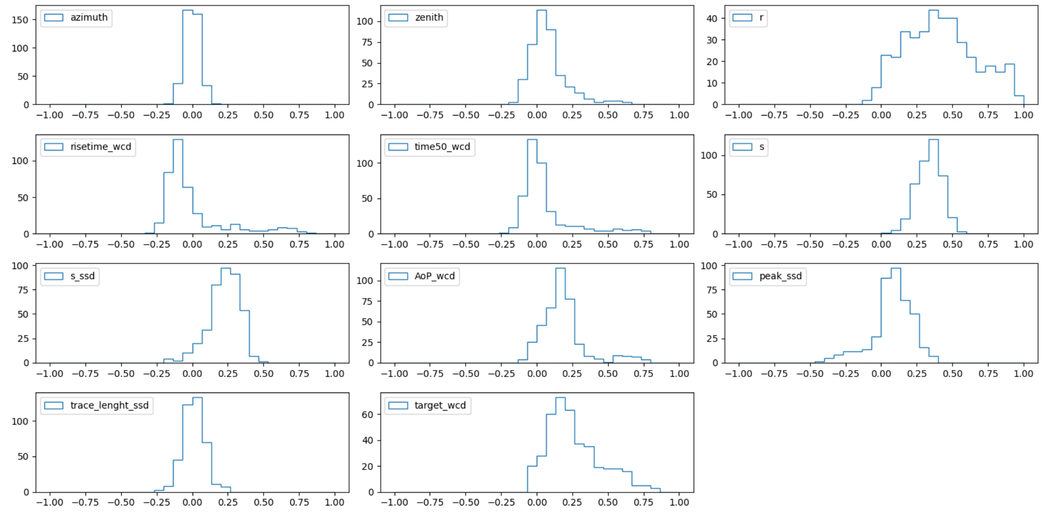
Correlations feature - azimuth

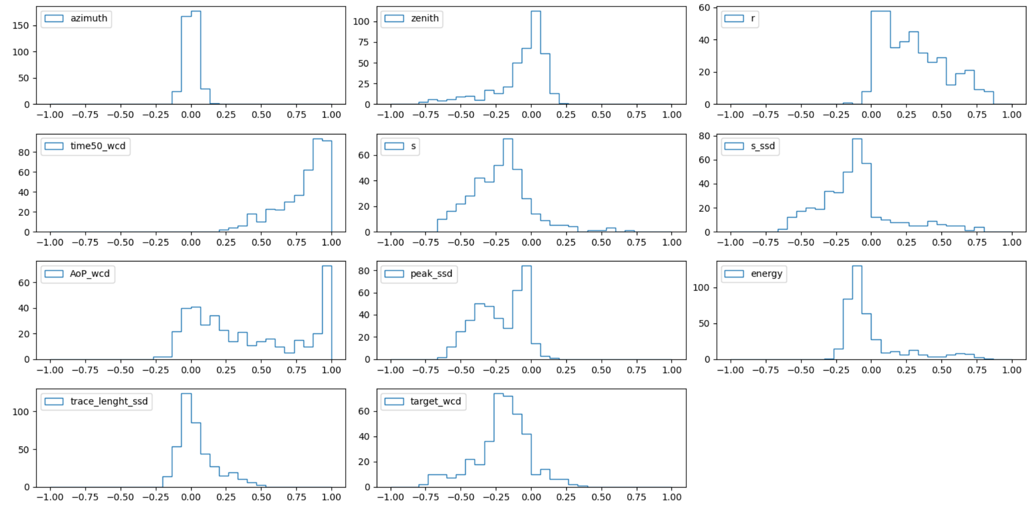


Correlations feature - r



Correlations feature - energy

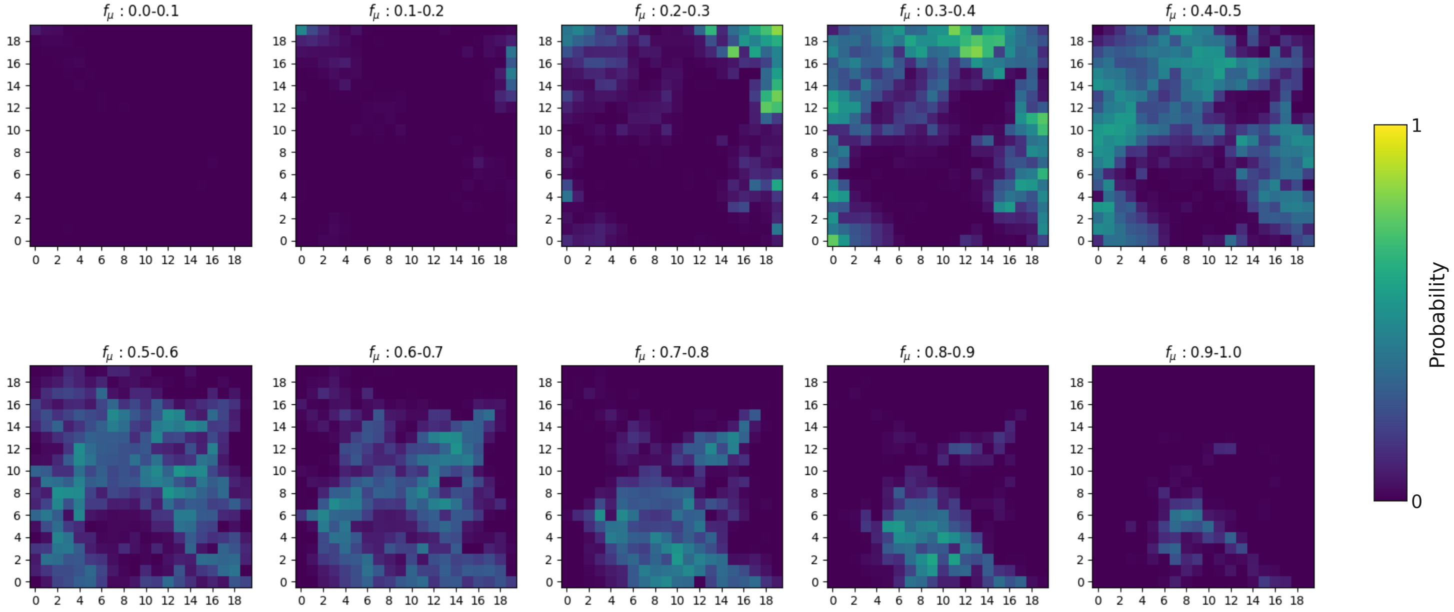


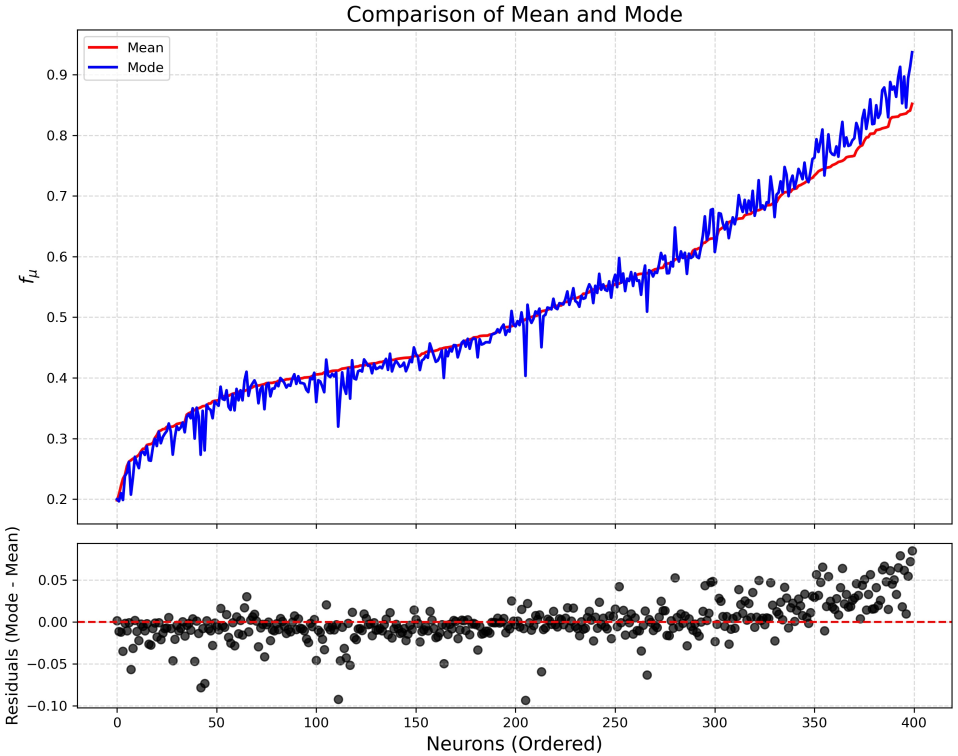


Probability MAPS

Probability Maps $f_\mu \subset [0.0, 1.0]$ binned in 10 classes

(a discrete pdf per single neuron)





The input data distributions in each of the 400 neurons show good agreement between the mean and the mode, making them interchangeable in the model's prediction, except at the edges of the phase space.

Statistics in trained SOM