



Related Collaborations

*IN2P3/IRFU Machine Learning workshop, 22-23 January 2020*

---

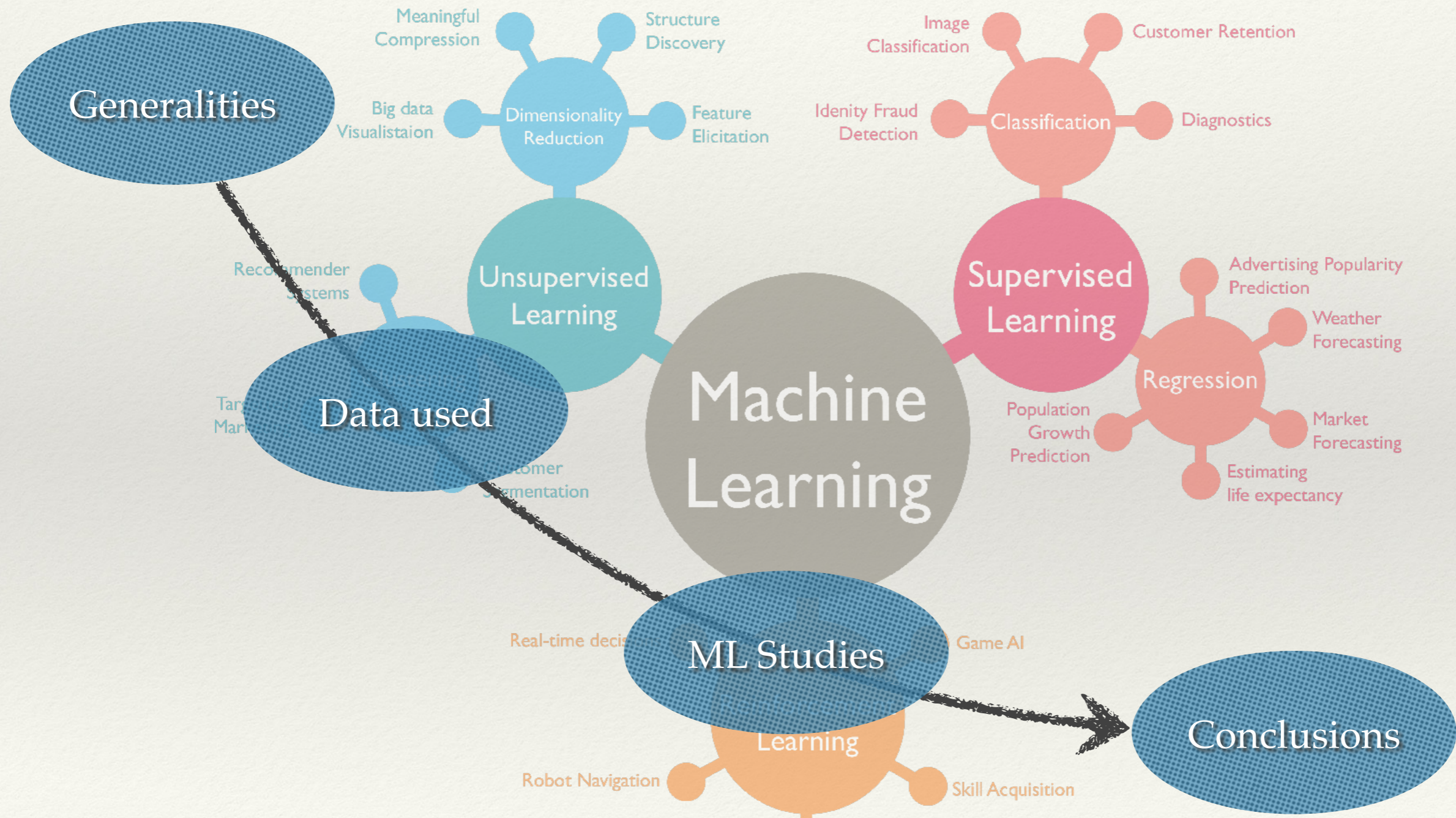
# Waveform Processing using Artificial Neural Networks



---

G. Baulieu, L. Ducroux, J. Dudouet, X. Fabian, O. Stézowski, K. Zougagh (stage M1)

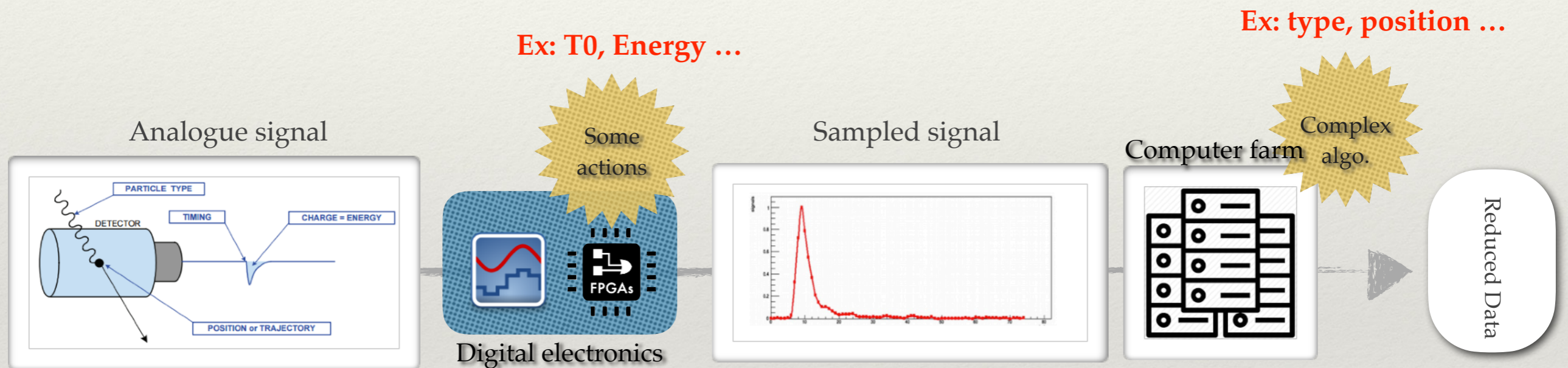
# Outline



# One R&D trend in Nuclear Physics

Developments in :

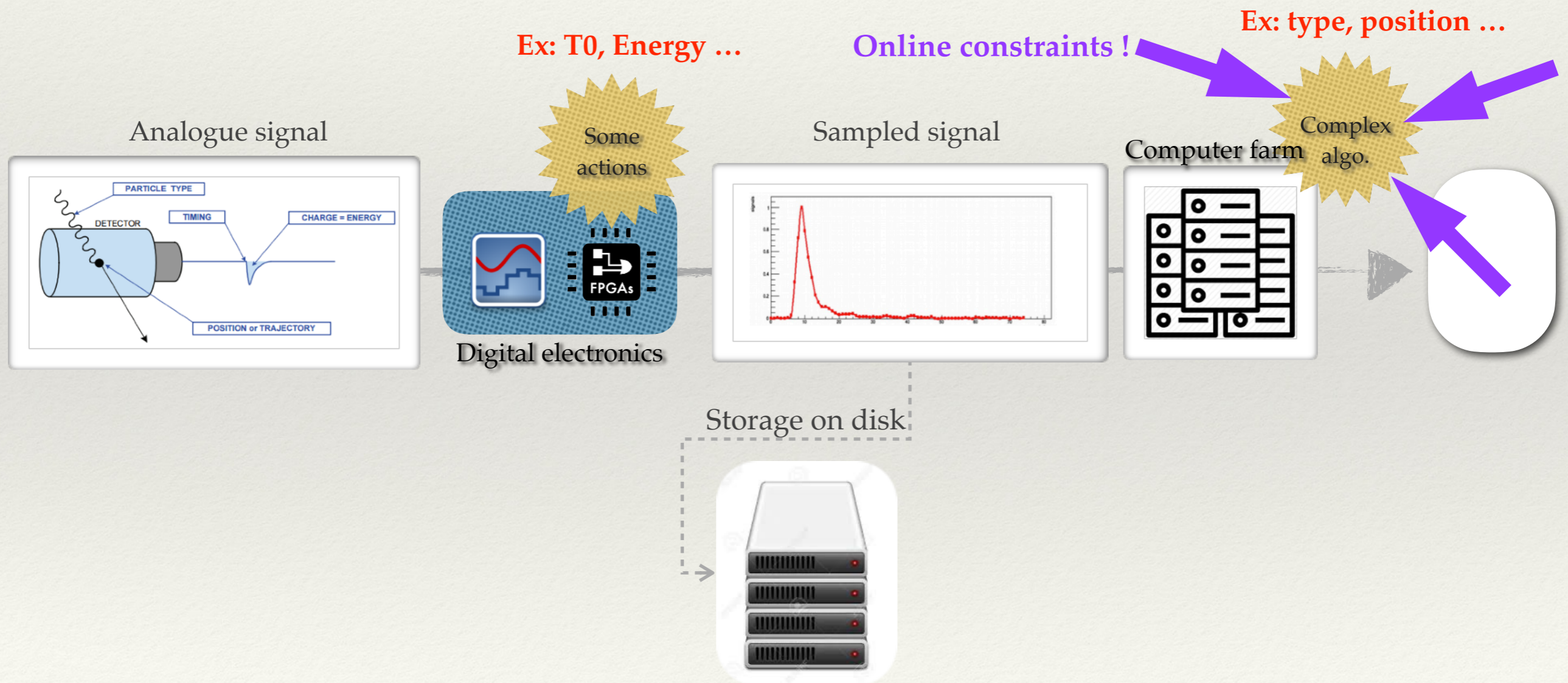
Digital electronics + high data transfer (optical fiber, ethernet) + disk array + computers  
+ ... ?



# One R&D trend in Nuclear Physics

Developments in :

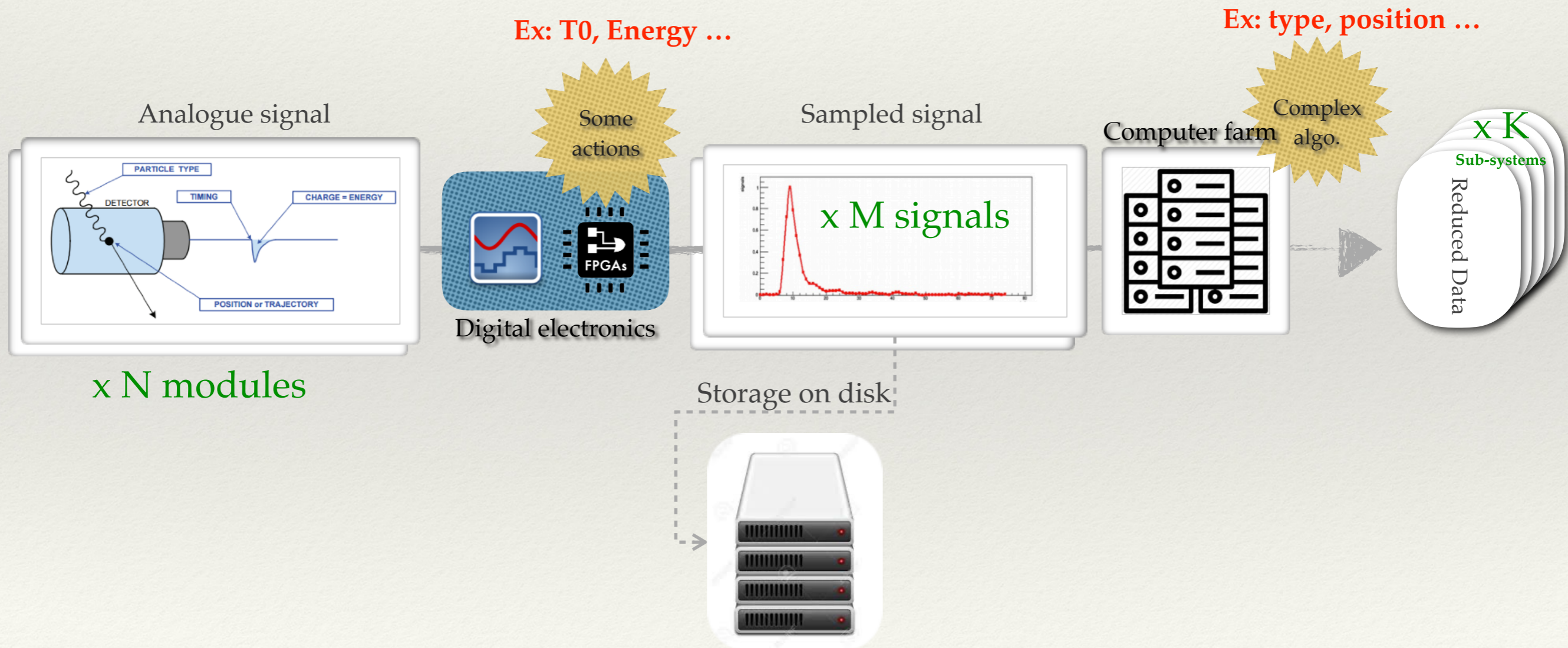
Digital electronics + high data transfer (optical fiber, ethernet) + disk array + computers  
+ ... ?



# One R&D trend in Nuclear Physics

Developments in :

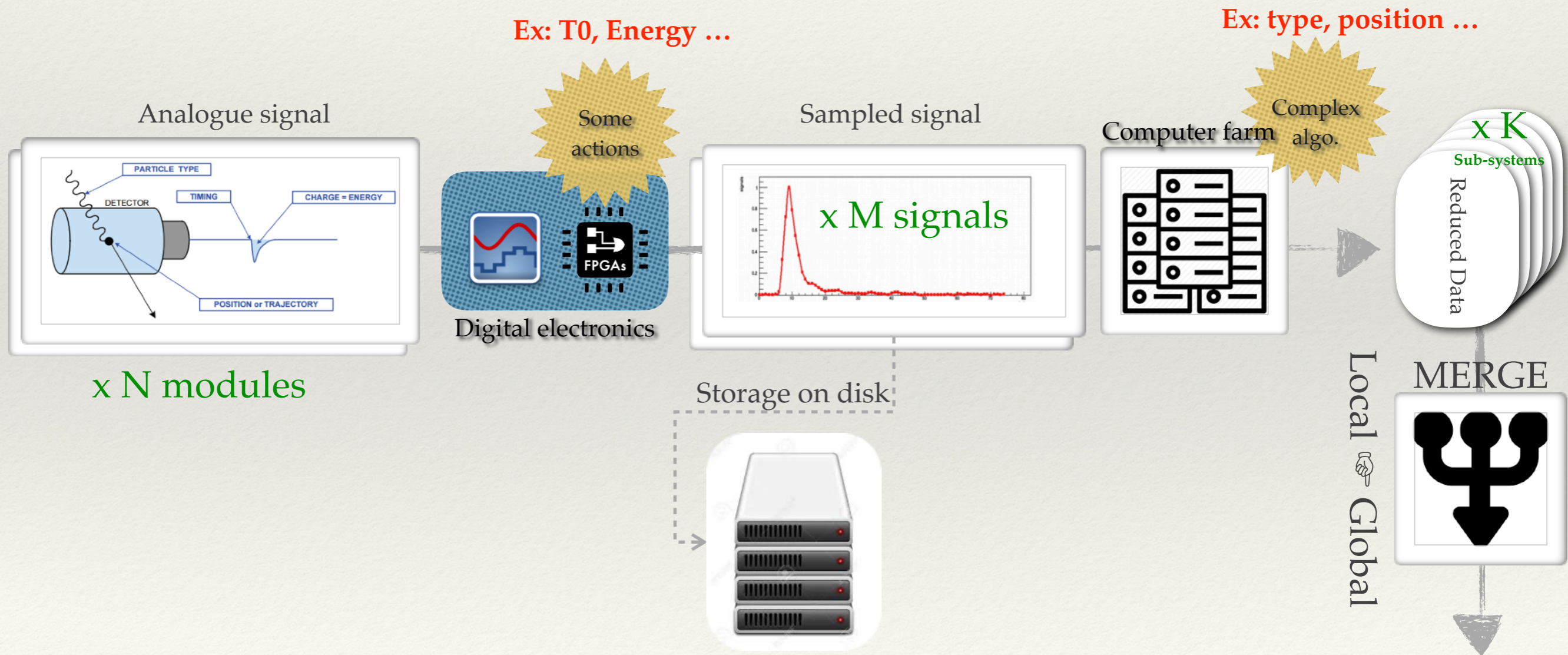
Digital electronics + high data transfer (optical fiber, ethernet) + disk array + computers  
+ ... ?



# One R&D trend in Nuclear Physics

Developments in :

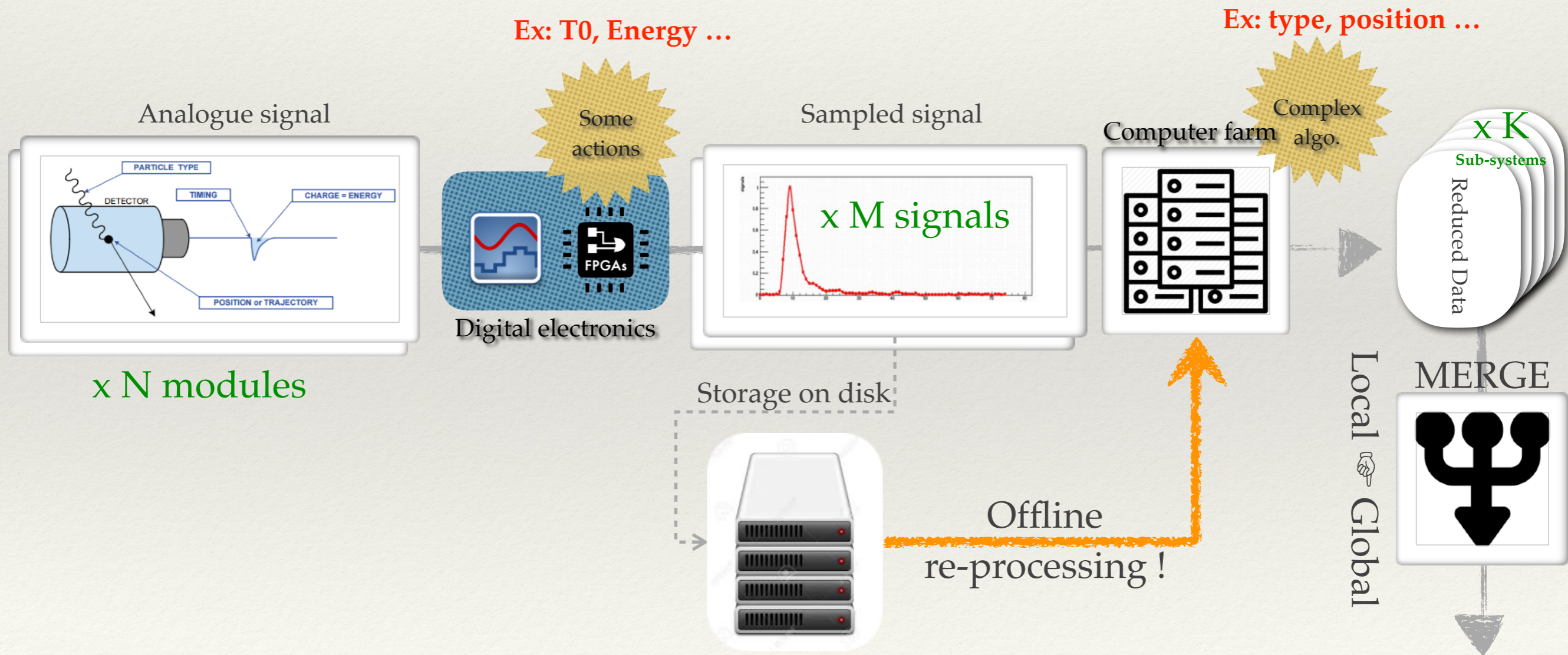
Digital electronics + high data transfer (optical fiber, ethernet) + disk array + computers  
+ ... ?



# One R&D trend in Nuclear Physics

Developments in :

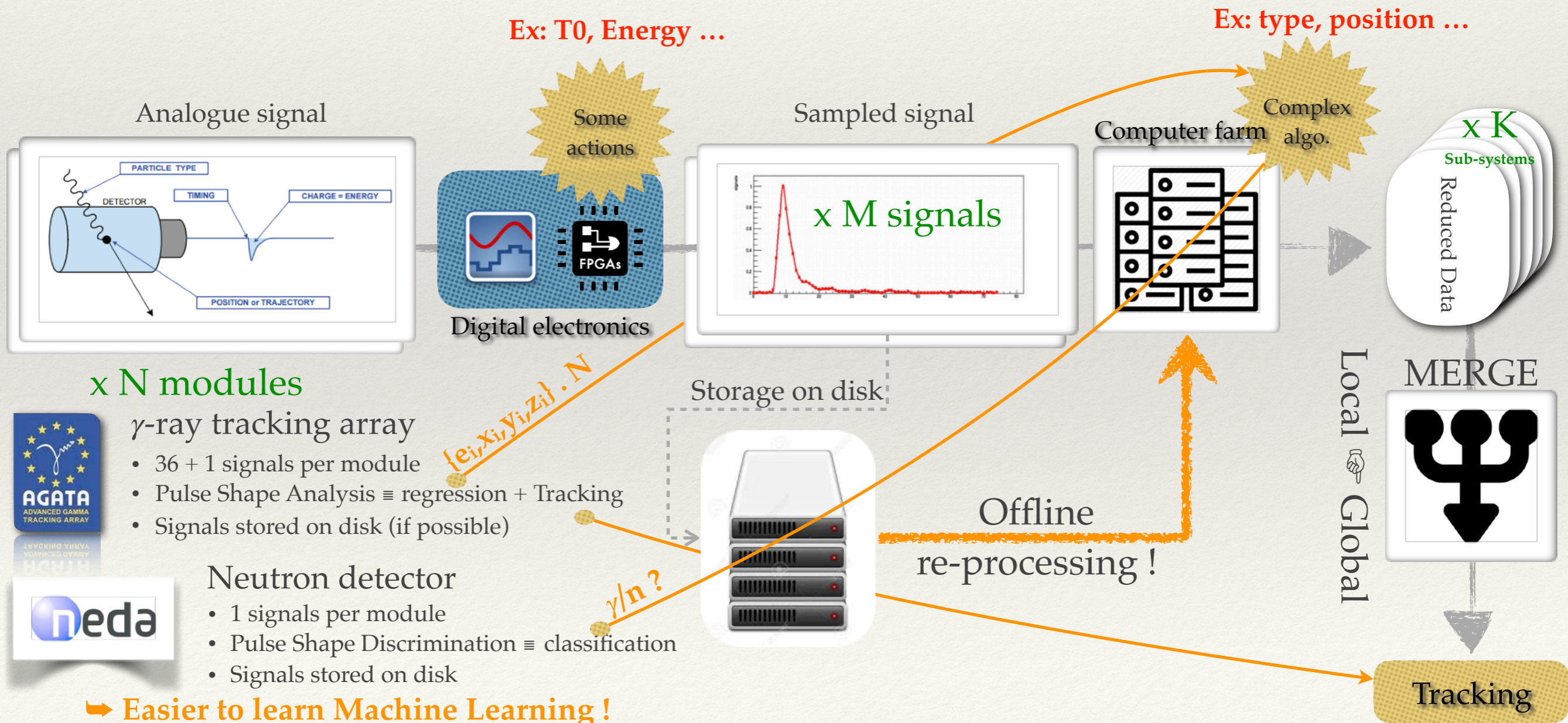
Digital electronics + high data transfer (optical fiber, ethernet) + disk array + computers  
+ ... ? + Machine Learning !



# One R&D trend in Nuclear Physics

Developments in :

Digital electronics + high data transfer (optical fiber, ethernet) + disk array + computers  
+ ... ? + Machine Learning !





# Pulse Shape Discrimination in NEDA

Experiment NEDA + AGATA in coincidence [GANIL 2018]



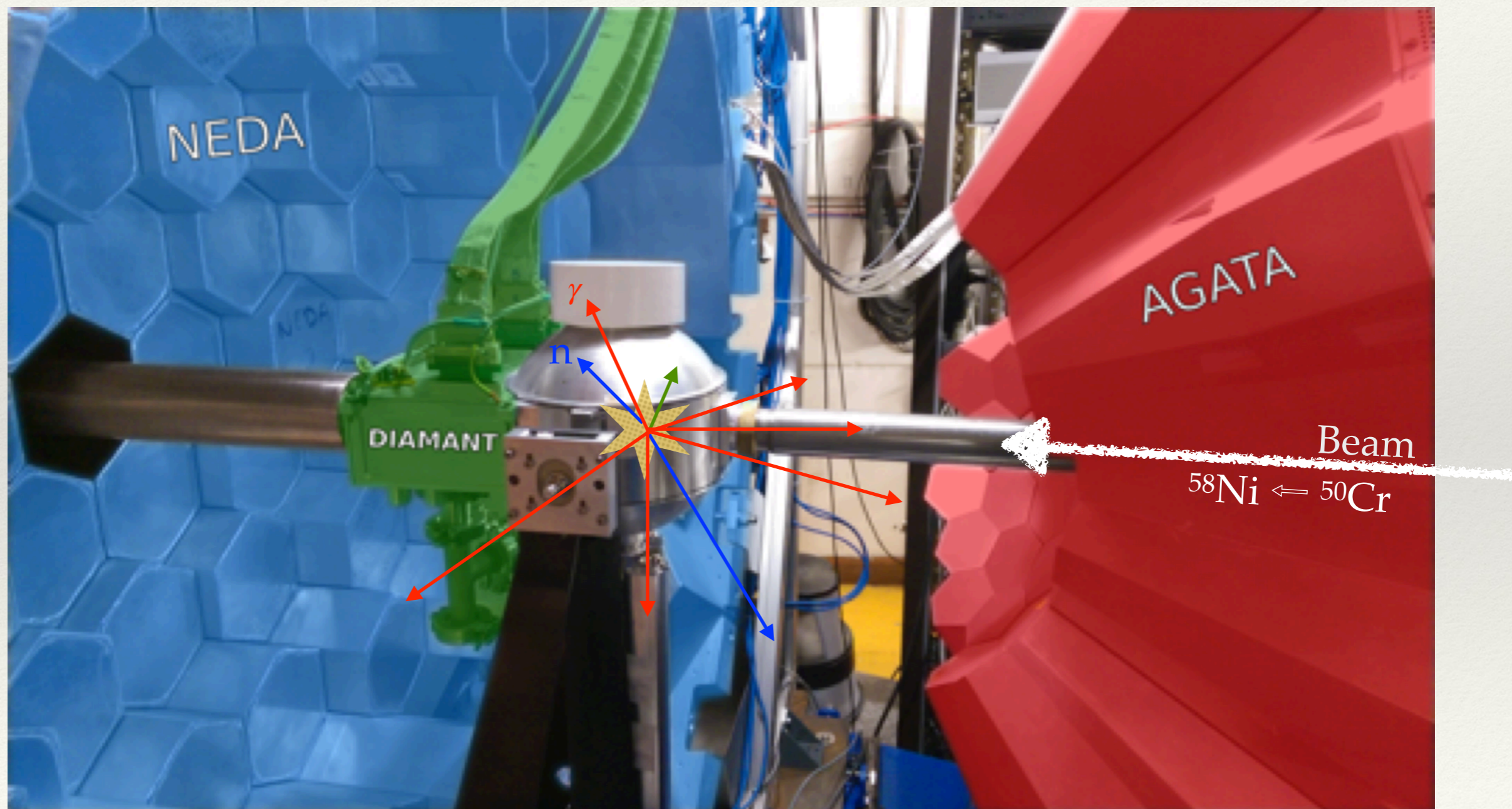
# Pulse Shape Discrimination in NEDA

Data from an experiment AGATA + NEDA + DIAMANT in coincidence [GANIL 2018]



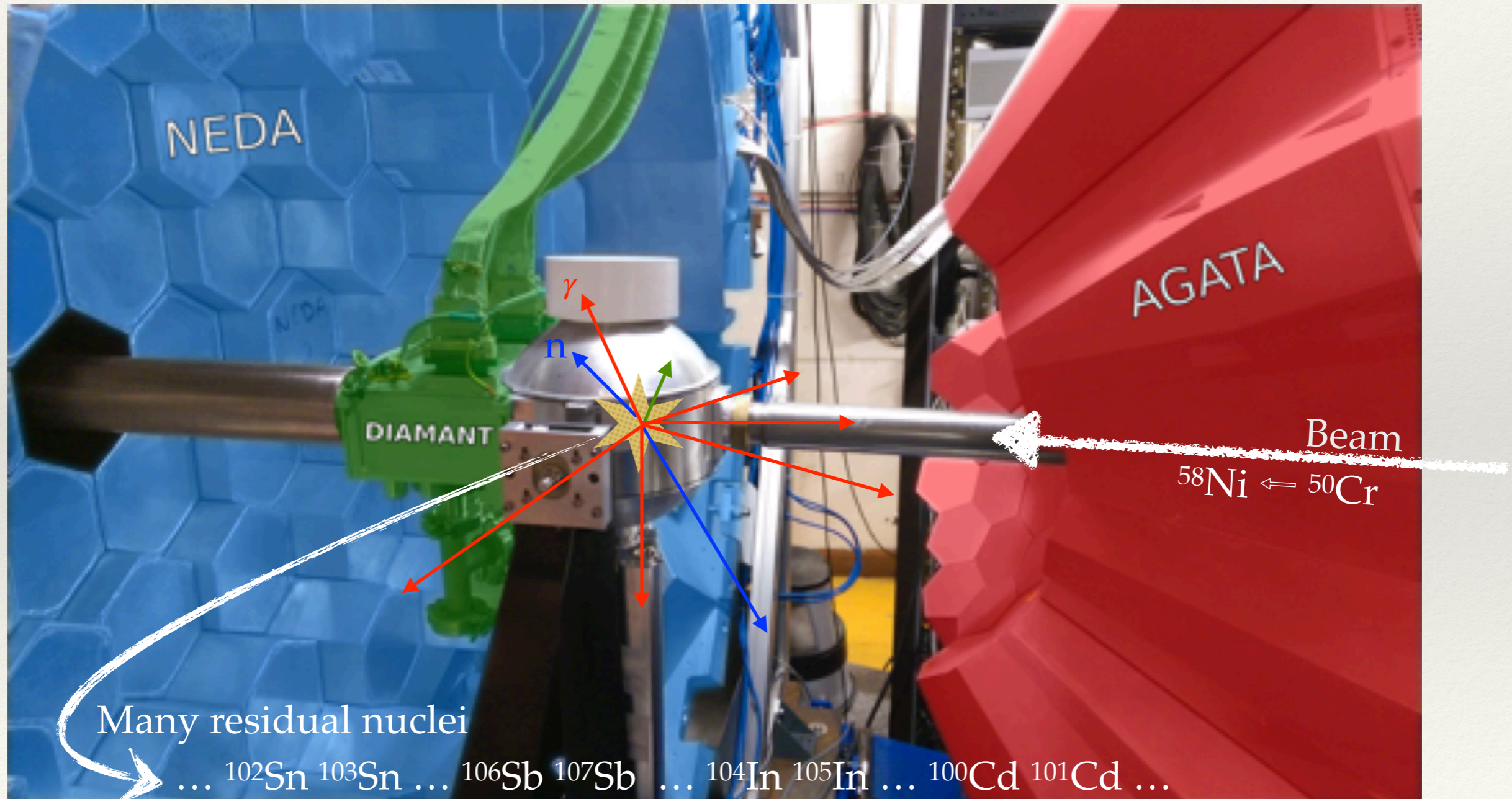
# Pulse Shape Discrimination in NEDA

Data from an experiment AGATA + NEDA + DIAMANT in coincidence [GANIL 2018]



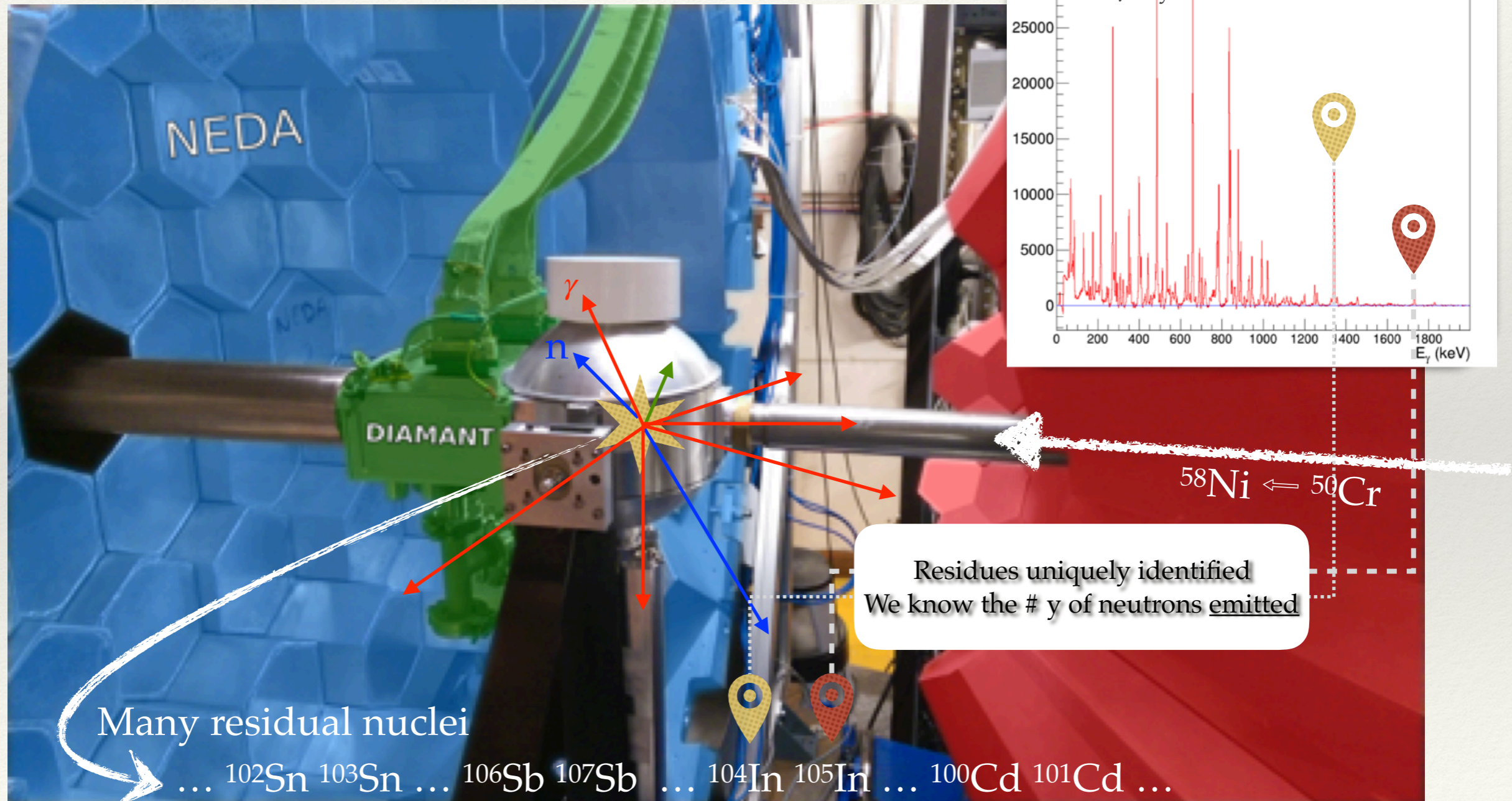
# Pulse Shape Discrimination in NEDA

Data from an experiment AGATA + NEDA + DIAMANT in coincidence [GANIL 2018]



# Pulse Shape Discrimination in NEDA

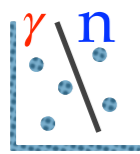
Data from an experiment AGATA + NEDA + DIAMANT in coincidence [GANIL 2018]



# Pulse Shape Discrimination in NEDA

Data from an experiment AGATA + NEDA + DIAMANT in coincidence [GANIL 2018]

Modules fired in NEDA



PSD

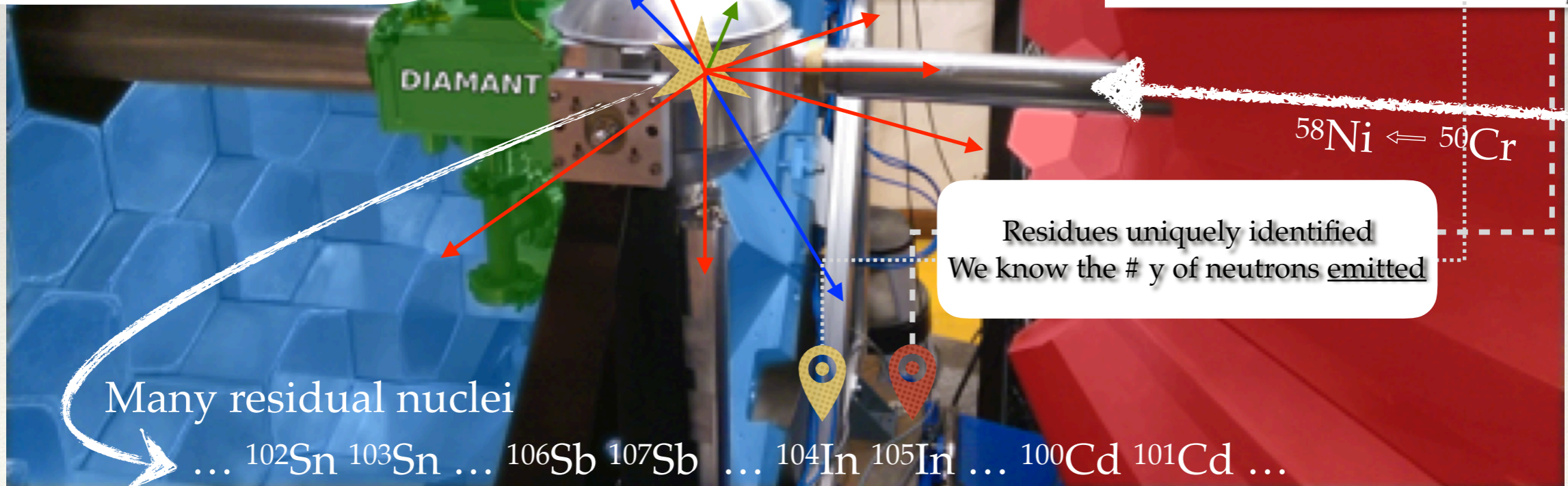
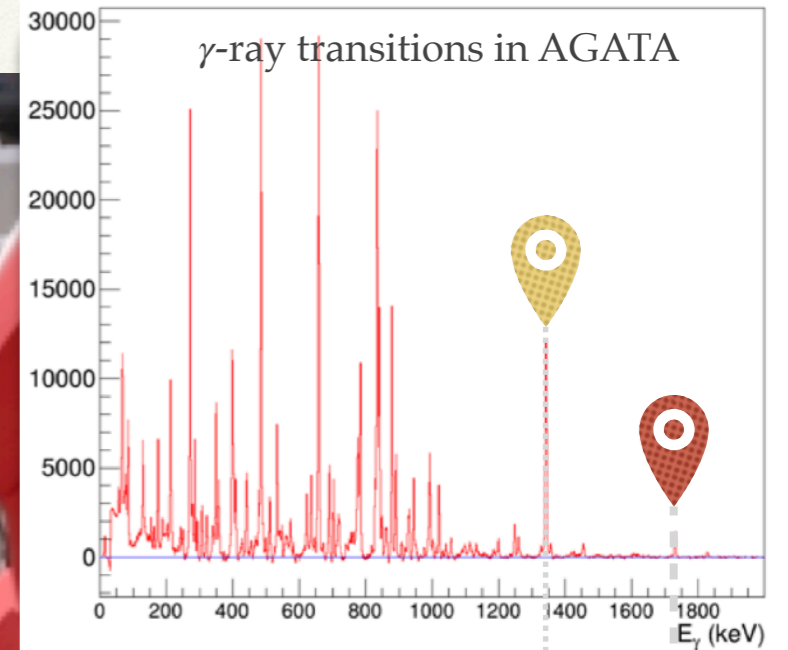
Artificial Neural Networks

We know the #  $x$  of neutrons detected

We can get the probability to be wrong

$$P(x_{\text{detected}} | y_{\text{emitted}})$$

**Mislabel Probability**  $\equiv P(1n | 0n)$



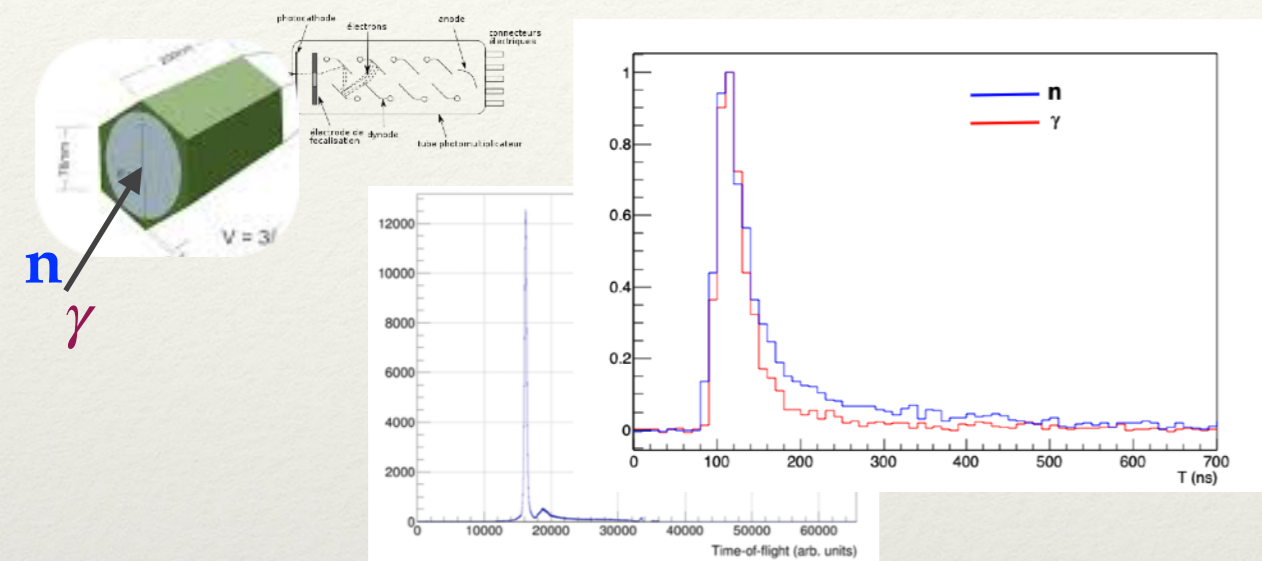
Many residual nuclei

...  $^{102}\text{Sn}$   $^{103}\text{Sn}$  ...  $^{106}\text{Sb}$   $^{107}\text{Sb}$  ...  $^{104}\text{In}$   $^{105}\text{In}$  ...  $^{100}\text{Cd}$   $^{101}\text{Cd}$  ...

Residues uniquely identified  
We know the #  $y$  of neutrons emitted



# ML for Pulse Shape Discrimination



Inputs used for the **Discrimination** :  
the waveform - the amplitude - the time of flight

Common parametrisation of the signal

$$s(t) = A [\exp(-t/\tau d1) - \exp(-t/\tau r) + R^*(\exp(-t/\tau d2) - \exp(-t/\tau r))] \text{ if } t > T0$$

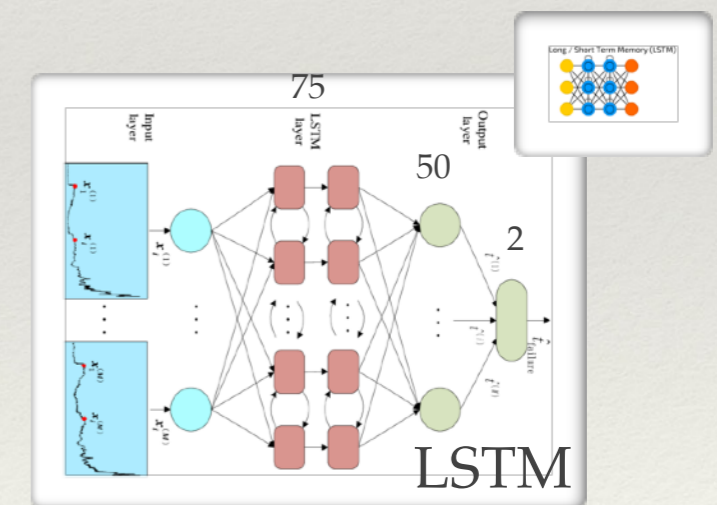
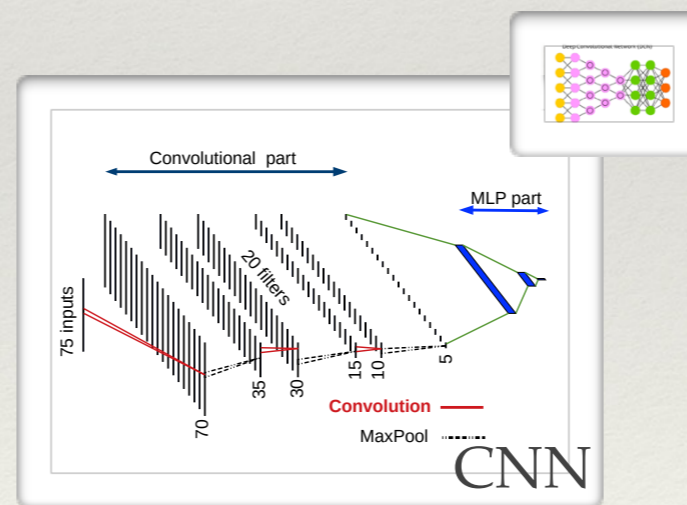
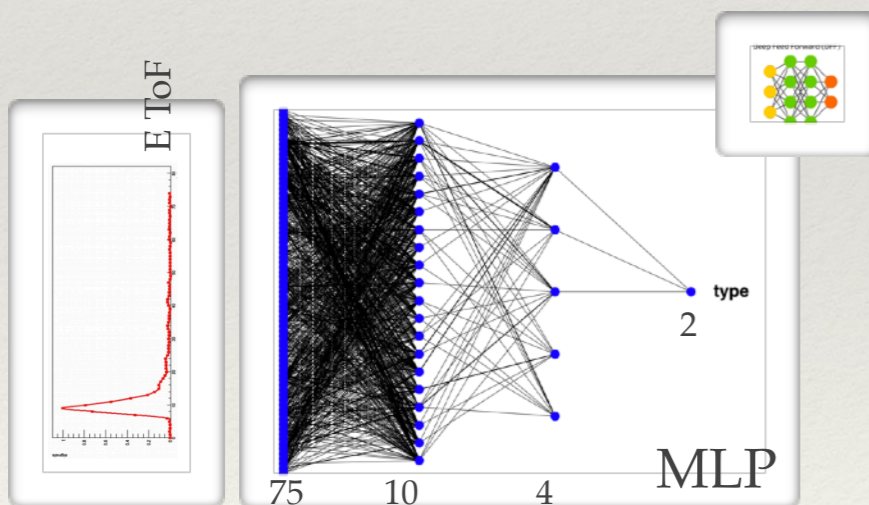
**A** amplitude = energy

**T0** relies on how the signal is captured

**td1, td2, tr** independent of  $\gamma$  and  $n$

**R** depends of the type of the particle

Three different Artificial Neural Network architectures tested : MLP / LSTM / CNN



👍 **pattern**

👍 **time series**

R&D NEDA

discrimination for low energy better than classical methods \*

Implementation with ROOT - mono thread / CPU

➡ **Tensorflow / multi CPU / GPU**



Number of parameters

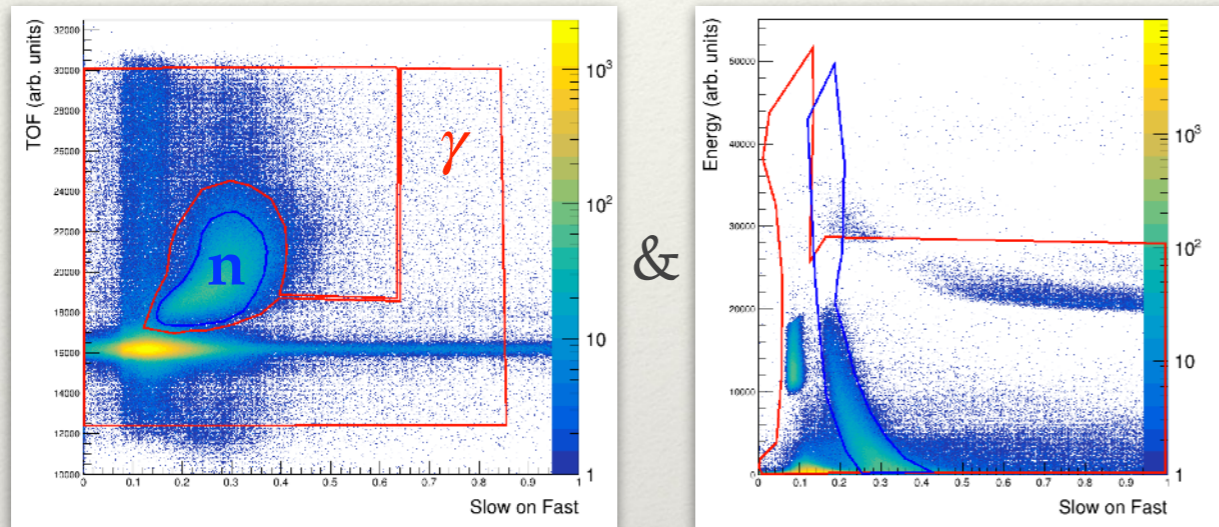
MLP: 814, LSTM: 10502, CNN: 7042

\* Ronchi et al., NIMA 610 (2009) 534-539

# ML for Pulse Shape Discrimination

● Training of the networks, Keras, python interface (Tensorflow board), recent GPU card  
 Typical time : 0,5 to 1 hour

To build the training data set, combinaisons of 2D cuts (uniformly distributed)  
 + selection of clean\* waveforms



Relu + softmax + Cross entropy loss function

Optimizers:

- Stochastic Gradient Descent
- ADAM

MLP / CNN

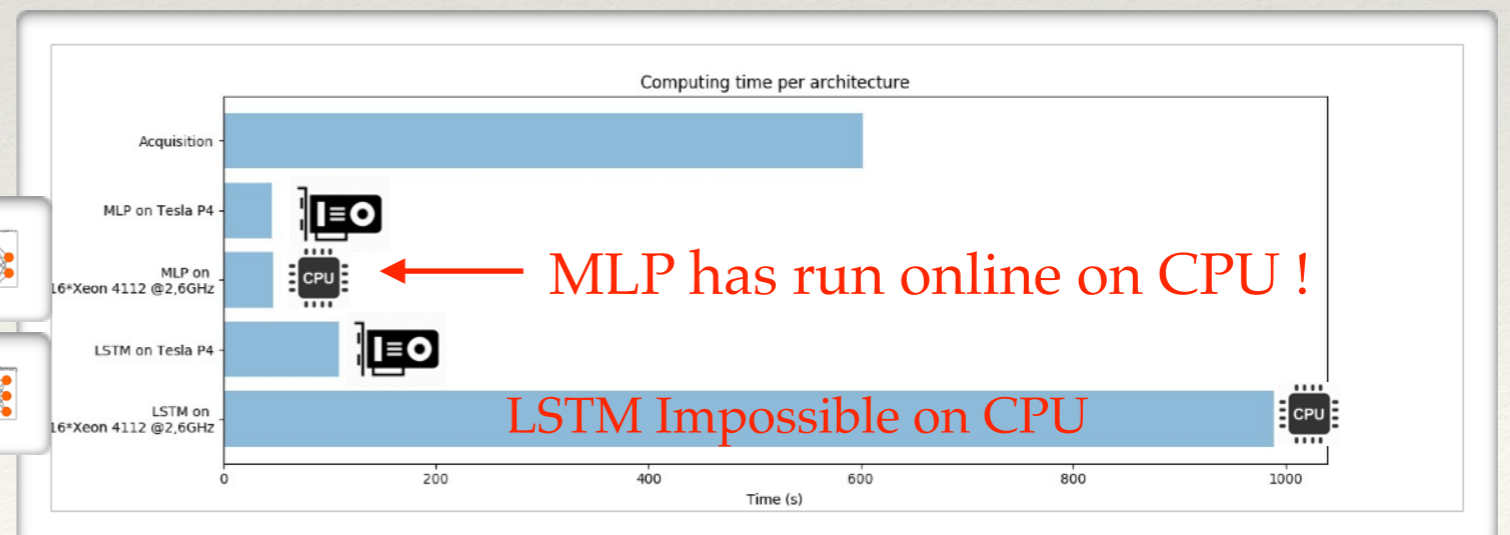
LSTM

● Inference Tensorflow, C++ interface

➔ Batch of signals for efficiency

Batch Size	MLP		CNN		LSTM	
	CPU	GPU	CPU	GPU	CPU	GPU
5000	2μs	1μs	10μs	3μs	50μs	13μs
20000	1μs	1μs	12μs	2μs	60μs	10μs
80000	1μs	1μs	12μs	4μs	75μs	9μs

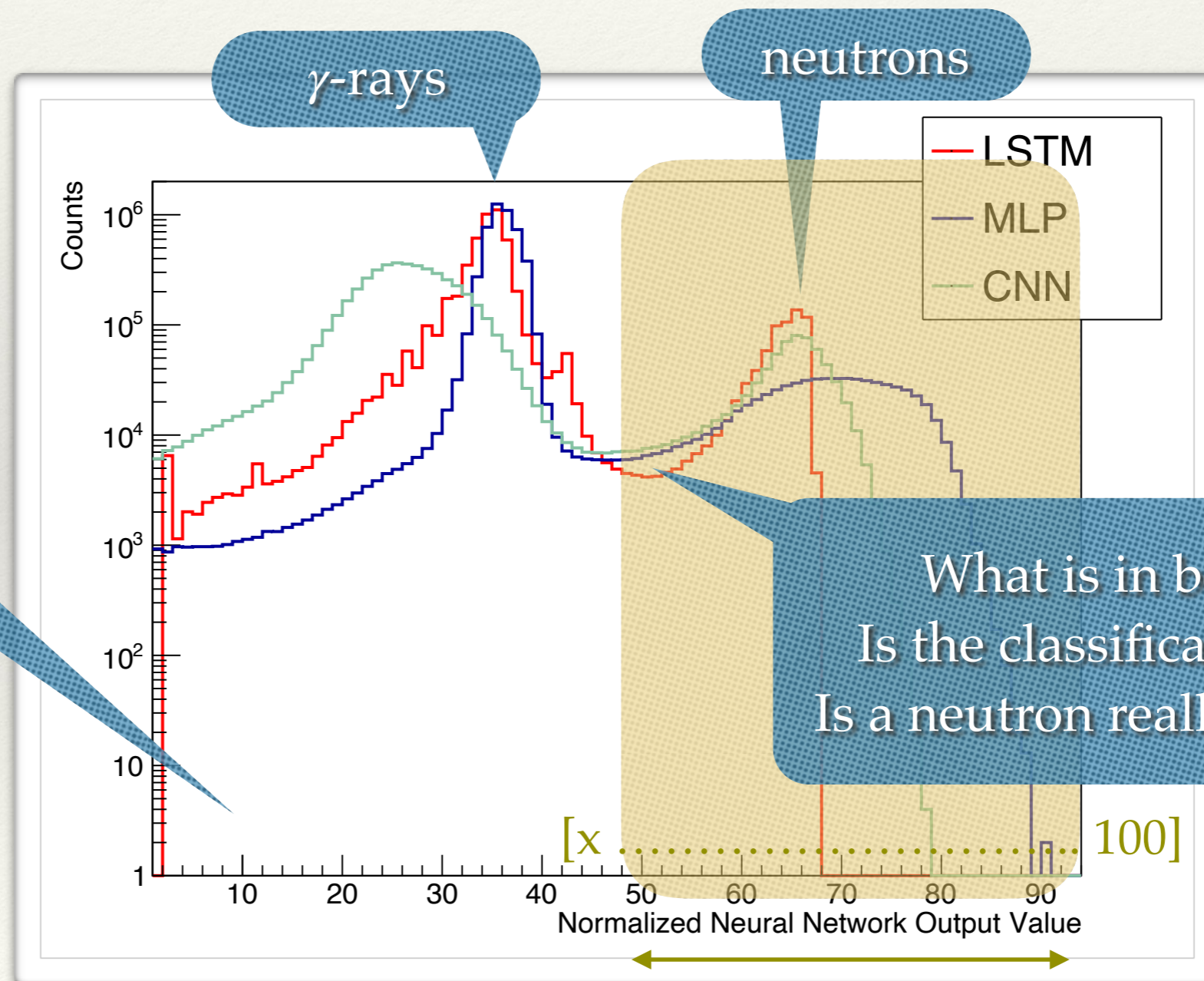
Typical time for one signal





# ML for Pulse Shape Discrimination

Distribution of the output value of the three different networks



Resolutions  
are different

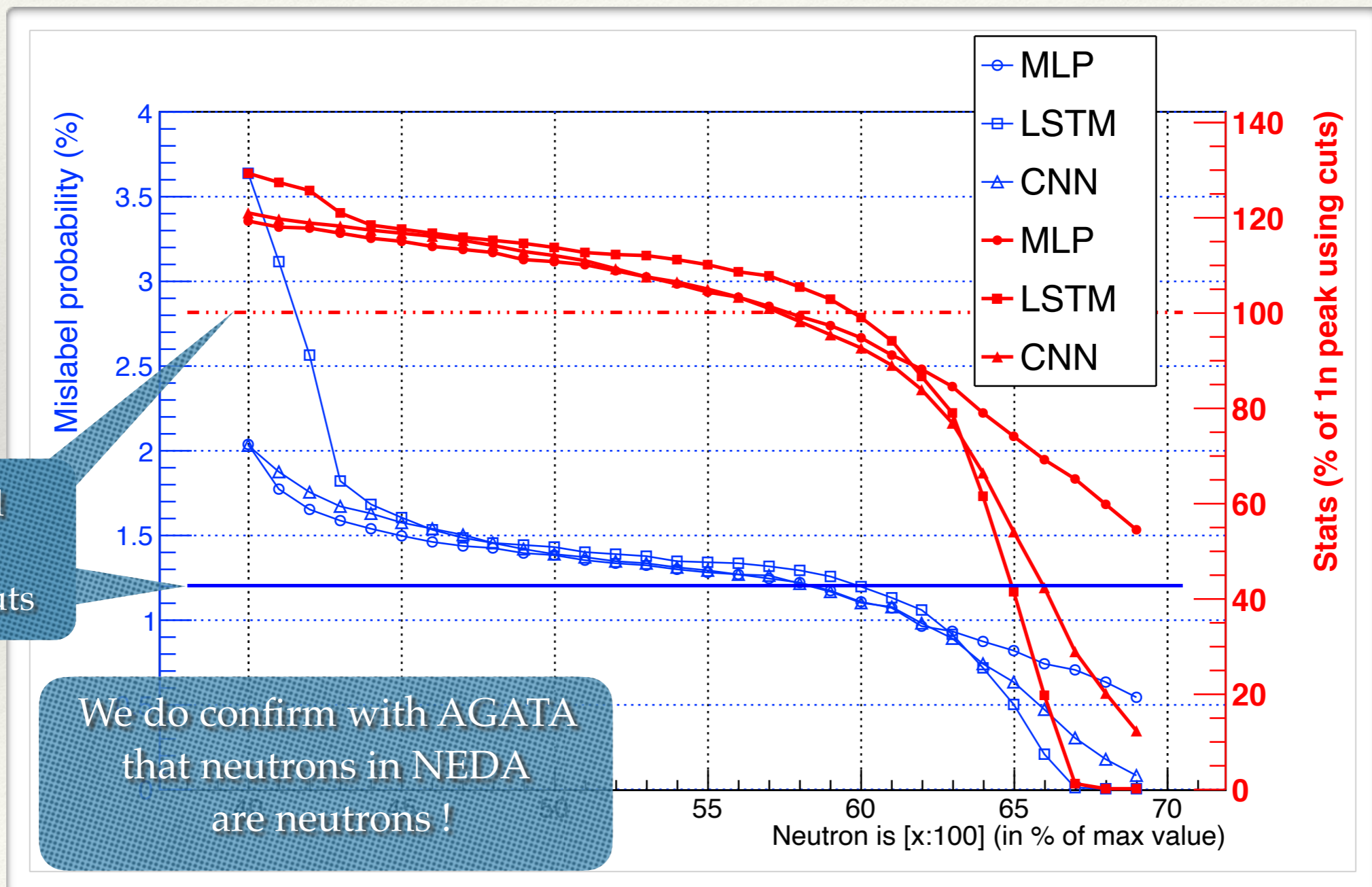
LSTM best one ?

What is in between ?  
Is the classification good ?  
Is a neutron really a neutron ?

Continuous parameter to play with growing cut in a multi-dimensional space !

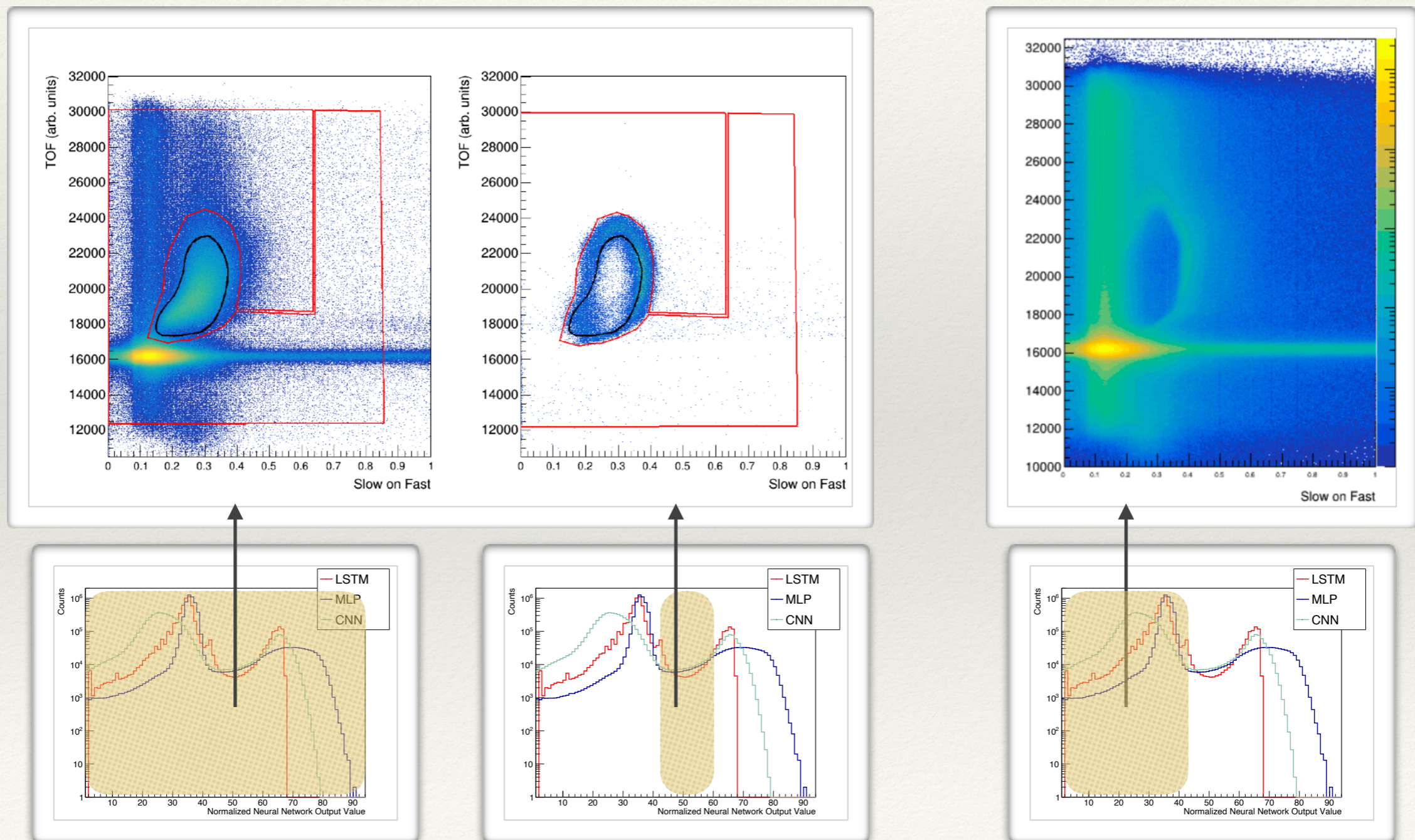
# ML for Pulse Shape Discrimination

Mislabel probability and impact on statistics in good events



# ML for Pulse Shape Discrimination

## Interpolation capabilities



# ML for Pulse Shape Discrimination

We had the feeling that LSTM seems a 'better' classifier. Why should it be ?

Other studies using synthetic signals generated from

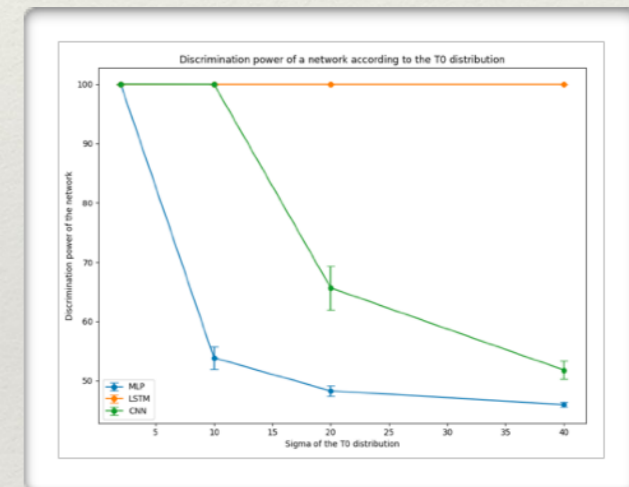
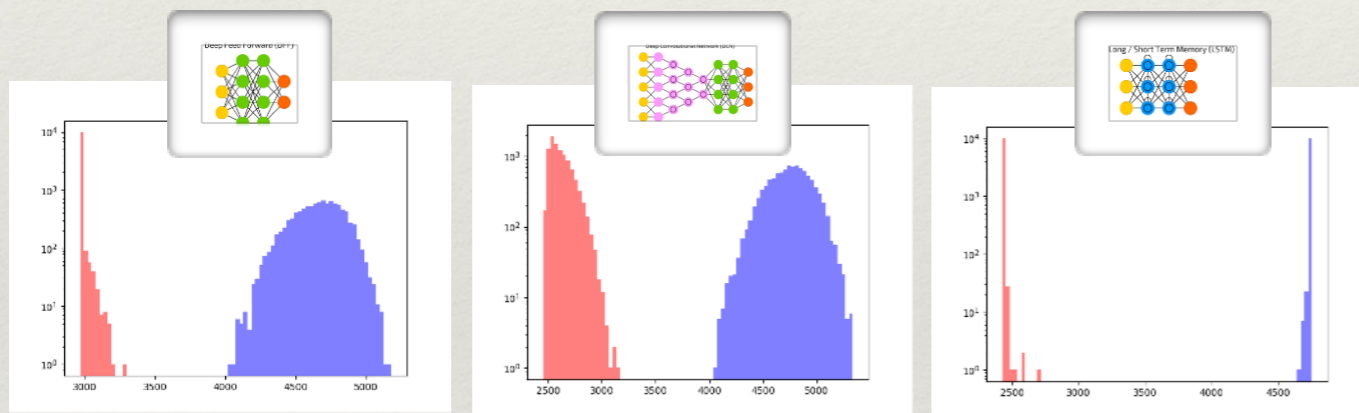
$$s(t) = \mathbf{A} [\exp(-t/\mathbf{td1}) - \exp(-t/\mathbf{tr}) + \mathbf{R}^*(\exp(-t/\mathbf{td2}) - \exp(-t/\mathbf{tr}))] \text{ if } t > \mathbf{T0}$$

Training with Gaussian distribution  $\sigma = 2$  of  $\mathbf{T0}$ , other parameters are constants

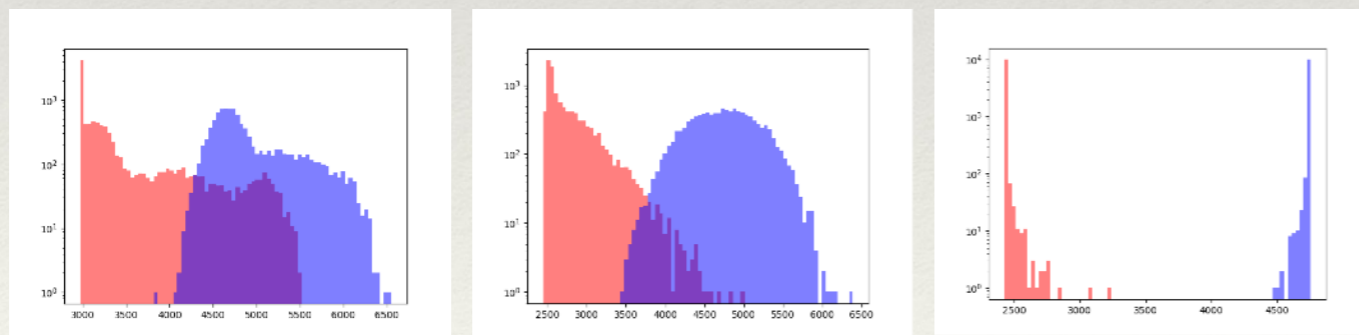
Test with Gaussian distribution  $\sigma = 20$  of  $\mathbf{T0}$

**LSTM the most robuste !**

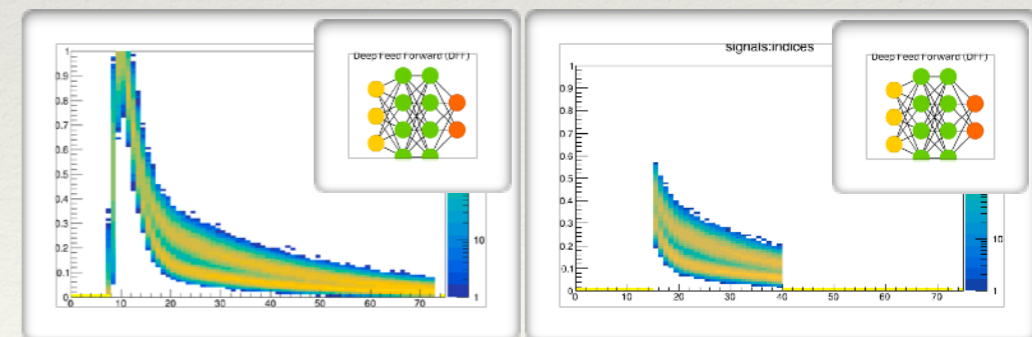
$\sigma = 2$



$\sigma = 20$



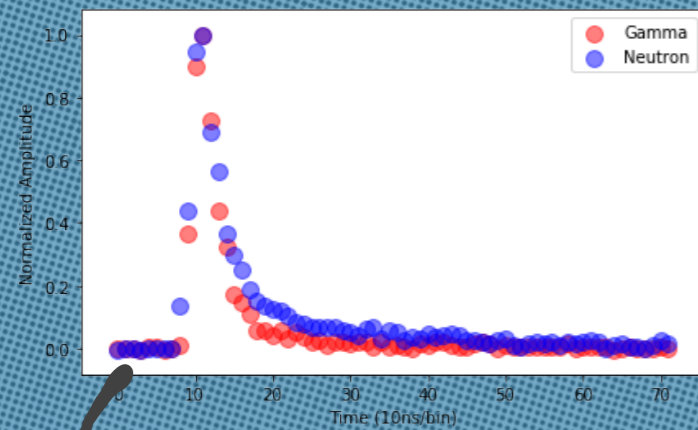
Neural network output value



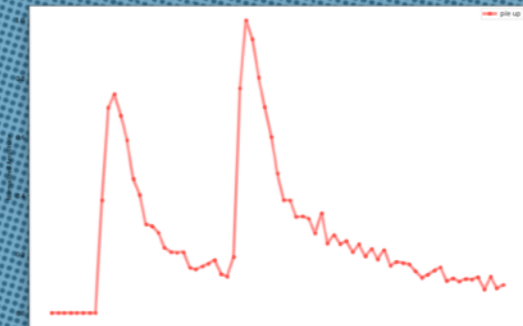
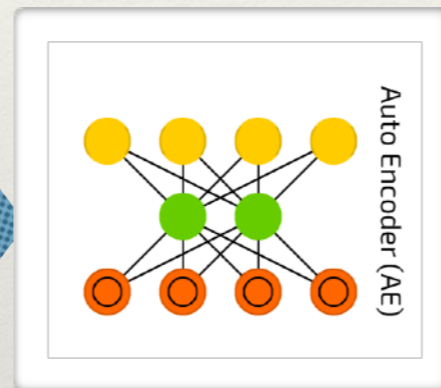
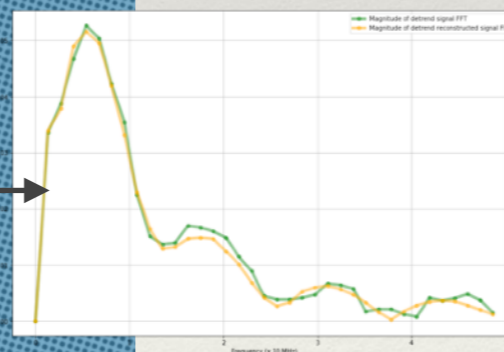
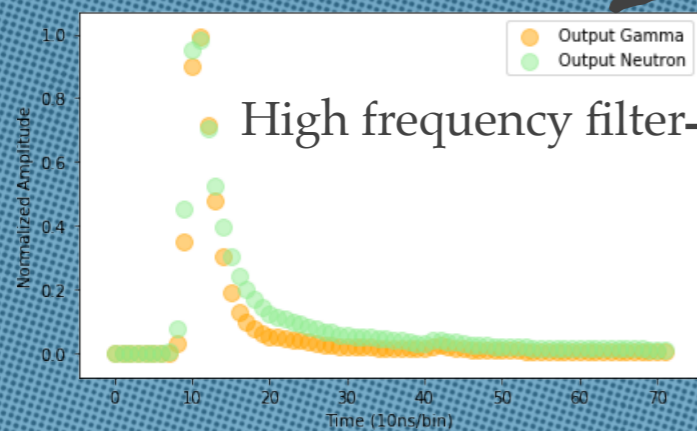
Truncated signals give the same output ...

**Feature extraction  $\equiv$  calibration really does matter !**

# Signals and auto encoders

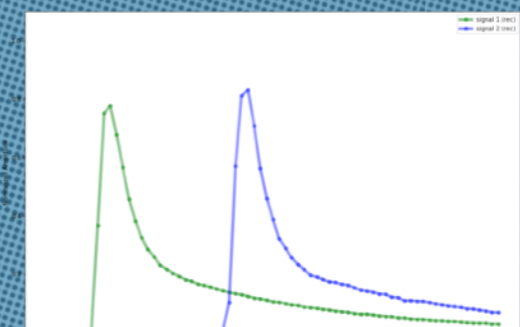
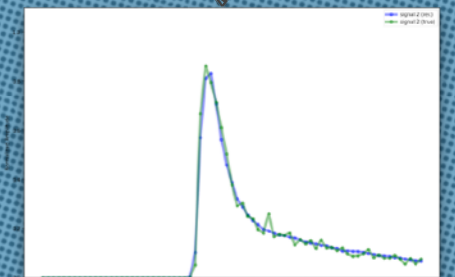
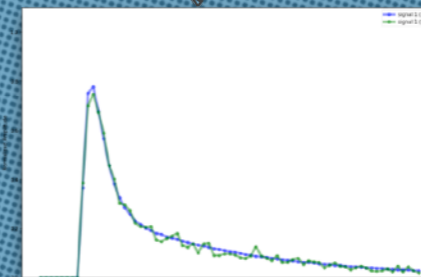


4  
neurons  
in bottleneck



Auto-encoder 1

Auto-encoder 2



Data compression !

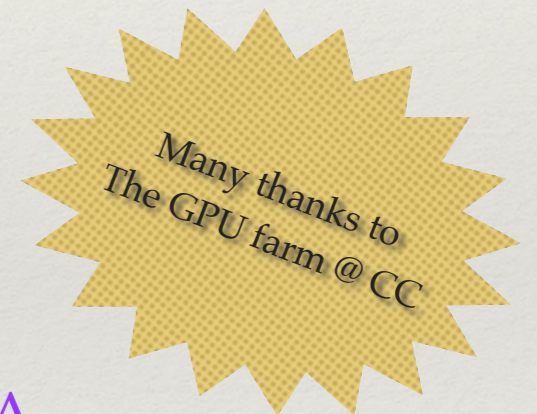
Denoising : Pile-up deconvolution

# Conclusions / Perspectives



## Our first steps in using Machine Learning for data processing, 3 ANN architectures studied

- MLP has run online in CPU farms !
- AGATA to qualify / quantify the  $\gamma/n$  discrimination in NEDA
- LSTM has some advantages, less sensitive to bad alignment of modules (time)
- Auto encoders into the game for compression / de-noising



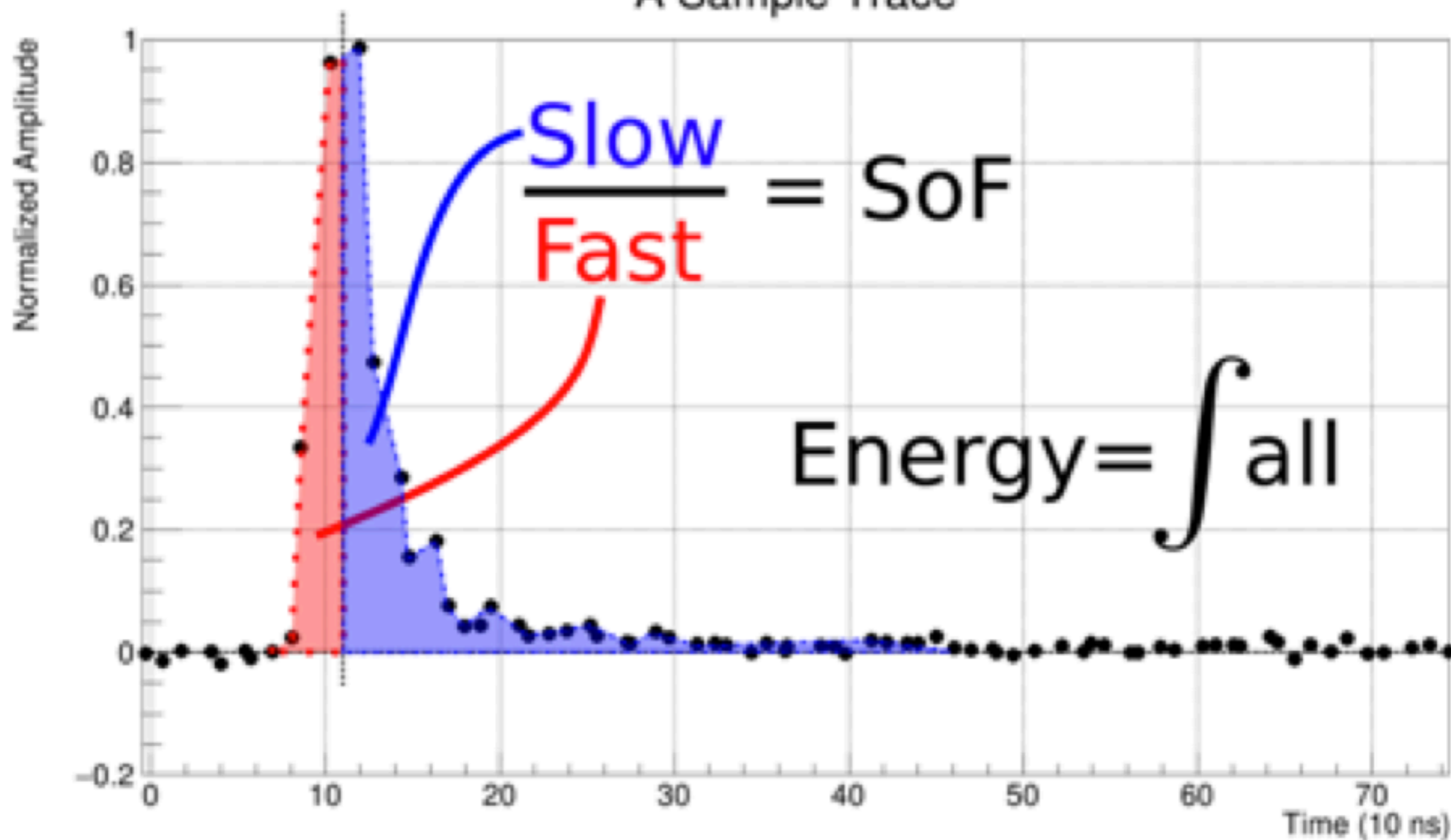
## Our future steps in using Machine Learning for data processing - NEDA

- Move to production -online- for data compression / de-noising / anomalies
- ANNs at lower stages ? FPGA ?

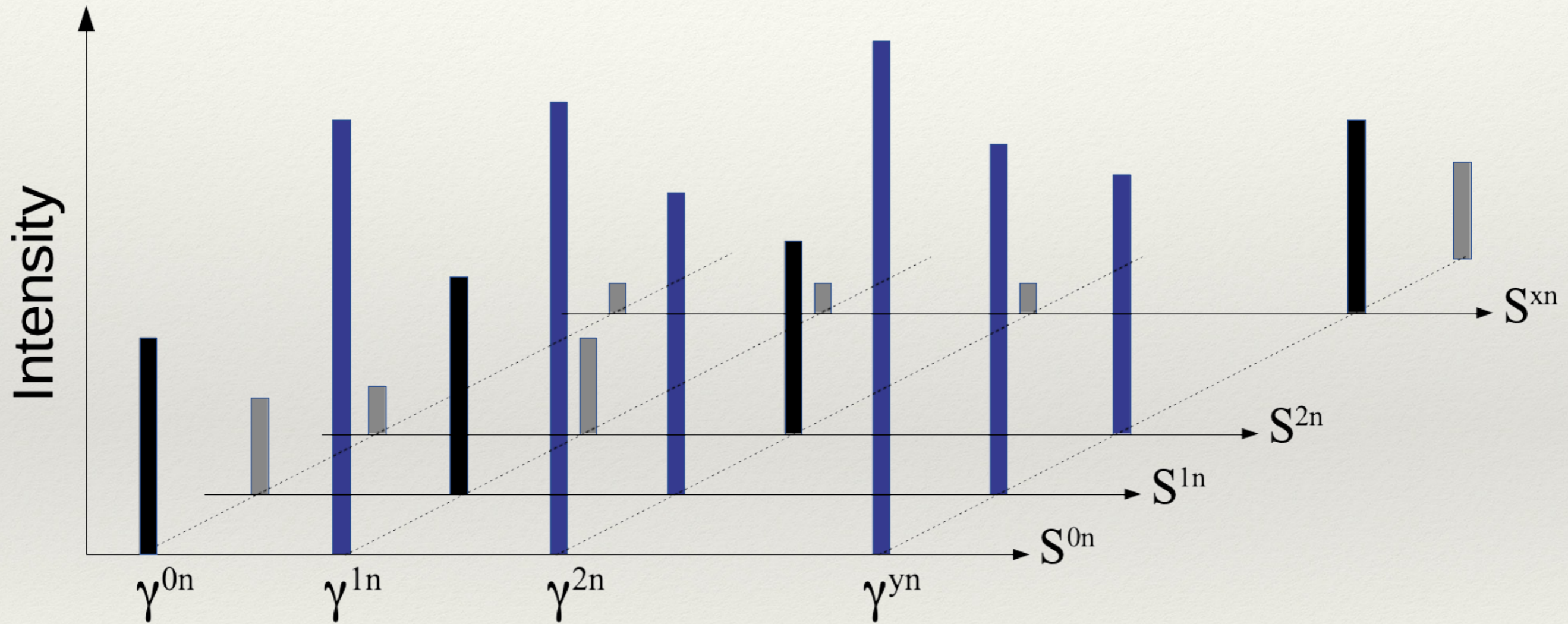
## Our future steps in using Machine Learning for data processing - AGATA

- Much more complex : Pulse Shape Analysis  $\equiv$  regression + Tracking (cluster stage)
- No model for the shapes of the pulses. It relies on complex simulations / scanning

# A Sample Trace



# Classification & Mislabeled

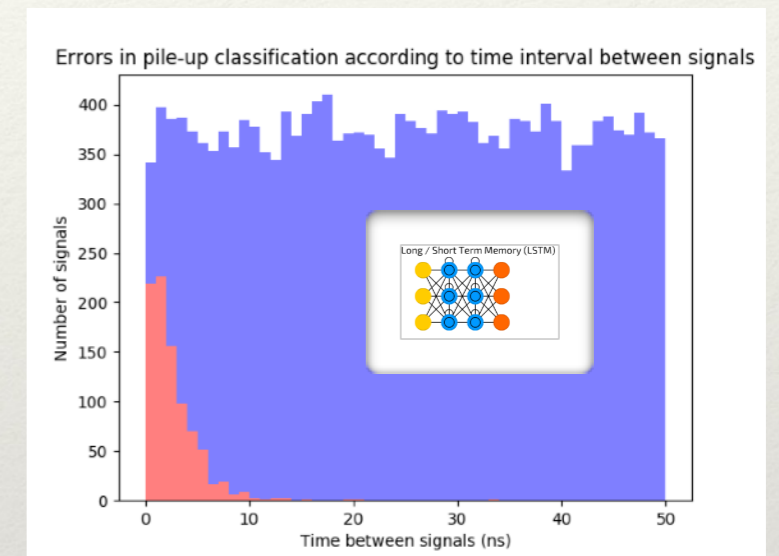
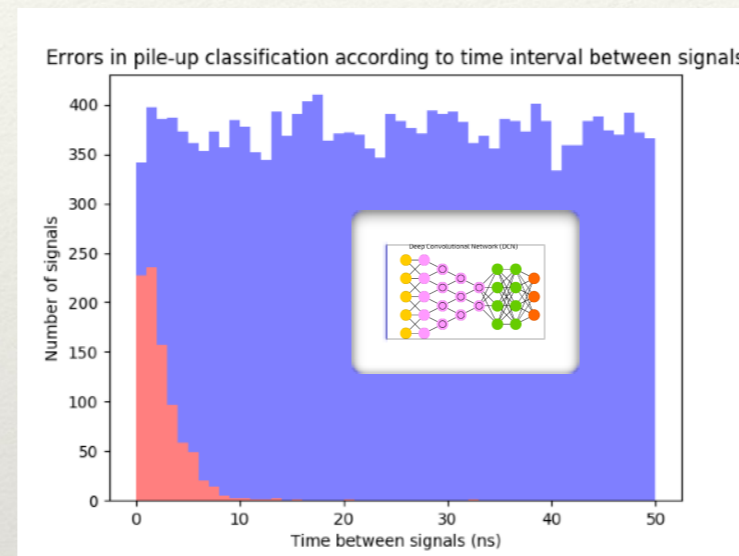
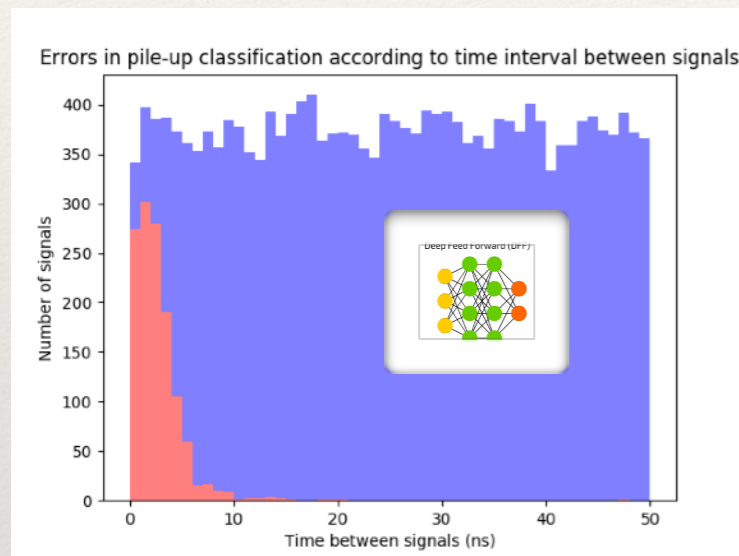




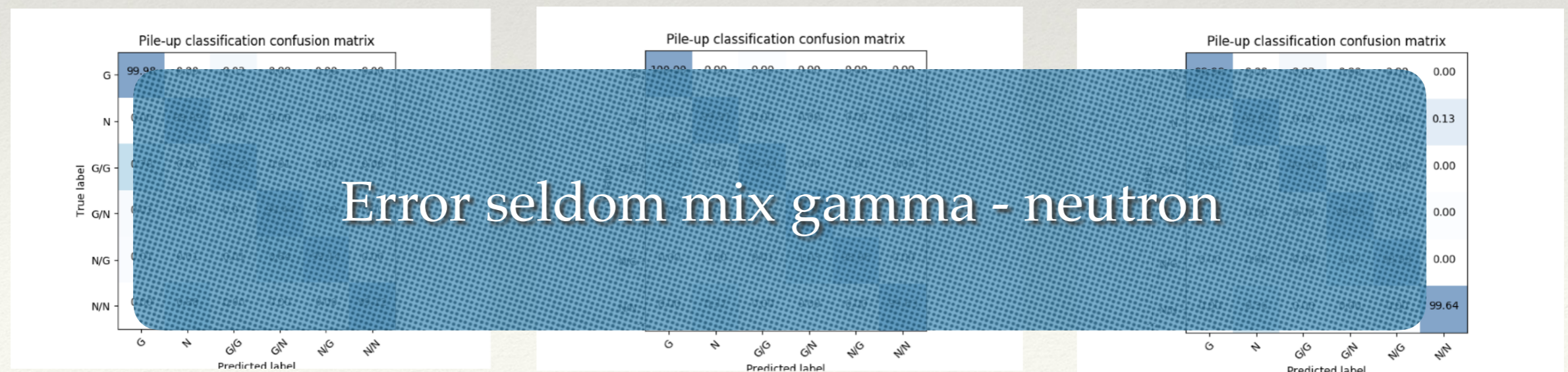
# Machine Learning on signals

## Study 2: Pileup identification

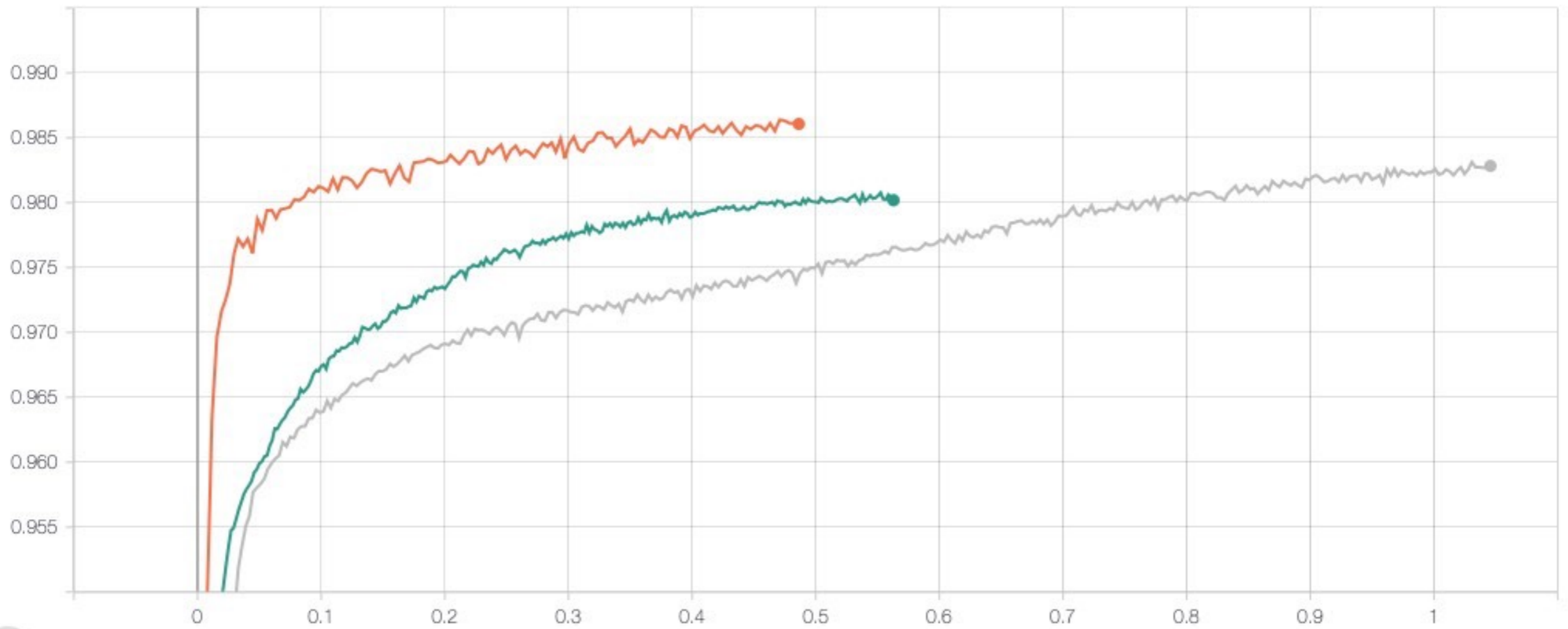
Error as fonction of the time between signals



## Confusion matrix



### epoch\_categorical\_accuracy



Name	Smoothed Value	Value	Step	Time	Relative
CNN 2020-01-08 17:41:37.482816	0.9828	0.9828	350.0	Wed Jan 8, 18:44:34	1h 2m 44s
LSTM 2020-01-08 18:44:50.793384	0.9861	0.9861	130.0	Wed Jan 8, 19:14:56	29m 10s
MLP 2020-01-08 17:07:32.840584	0.9802	0.9802	270.0	Wed Jan 8, 17:41:27	33m 46s