AI Tools for Plasma Diagnostics by X-ray Imaging and Spectroscopy in the PANDORA Project Frame

Bianca Peri on behalf of the PANDORA Collaboration

HIGH PRECISION X-RAY MEASUREMENTS 2025

JUN 16 – 20, 2025

LABORATORI NAZIONALI DI FRASCATI INFN

SAPIENZA UNIVERSITÀ DI ROMA

INFN LNS

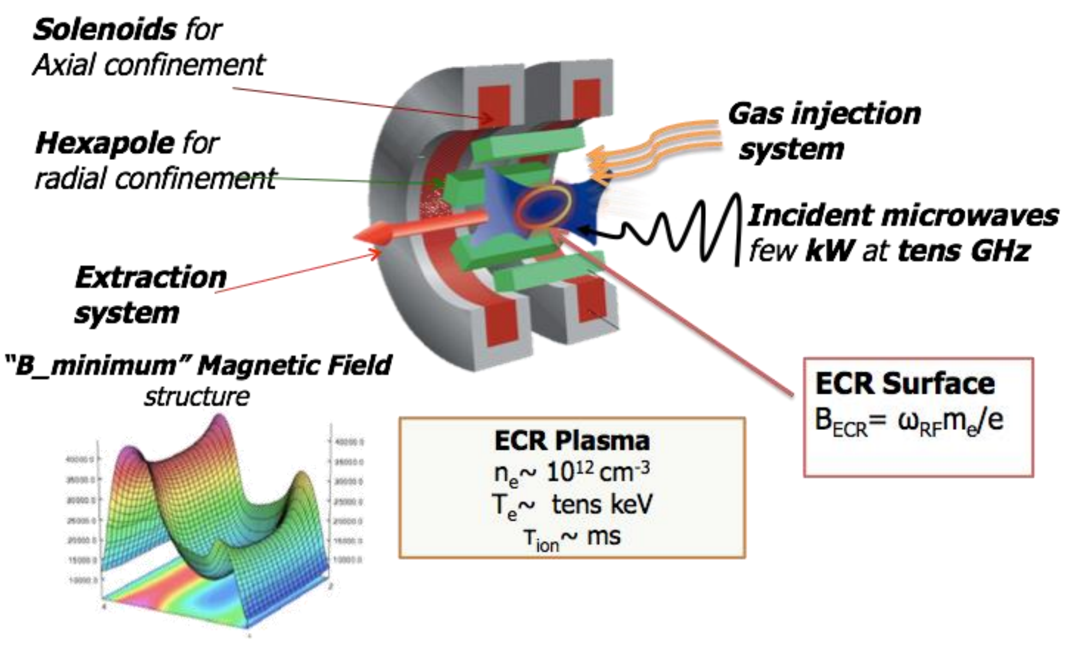
Plasmas for Astrophysics Nuclear Decay Observation and Radiation for Archaeometry

Outline

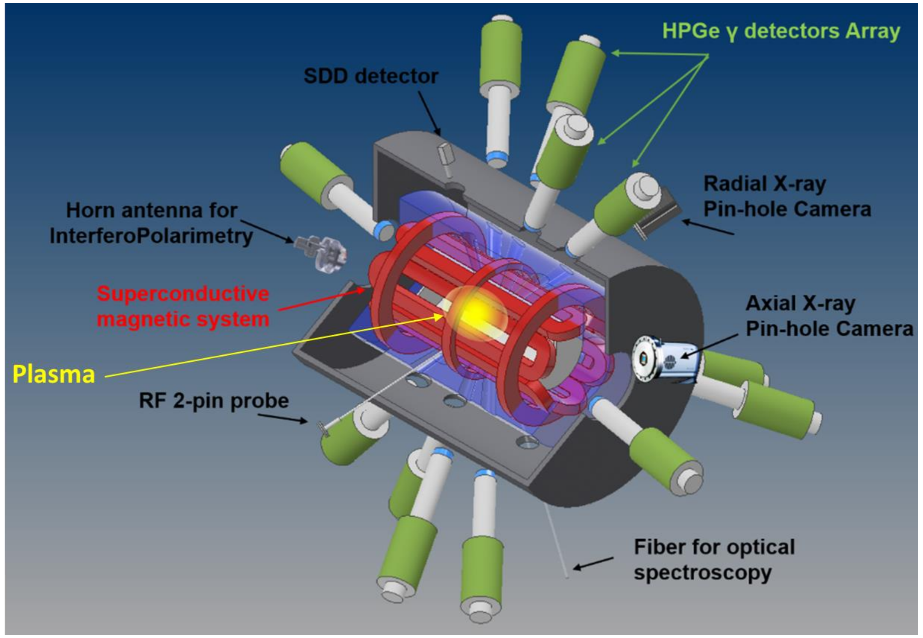
- **Introduction: CCD X-rays Imaging**
- **First steps: *Imaging & Clustering Tools***
- **Neural Networks**
- **Conclusions & Perspectives**

Introduction:
CCD X-rays Imaging

THE PANDORA PROJECT

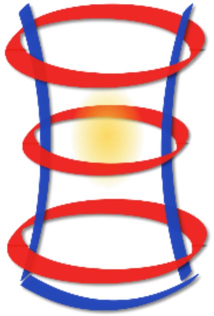


Plasma is excited by Electron-Cyctron-Resonance by microwaves and confined by magnetic fields



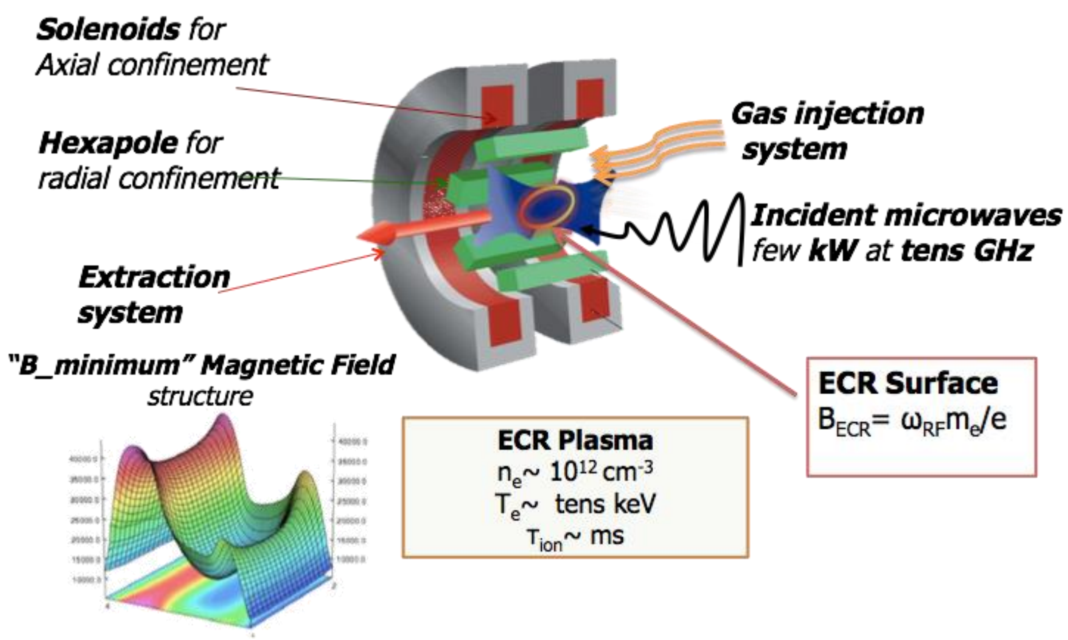
A multidiagnostic system surrounding the plasma chamber was developed to measure plasma parameters

Goal: investigate beta decay in stellar like Electron-Cyclotron-Resonance plasma in compact trap, for fundamental studies in Nuclear Astrophysics and its application.

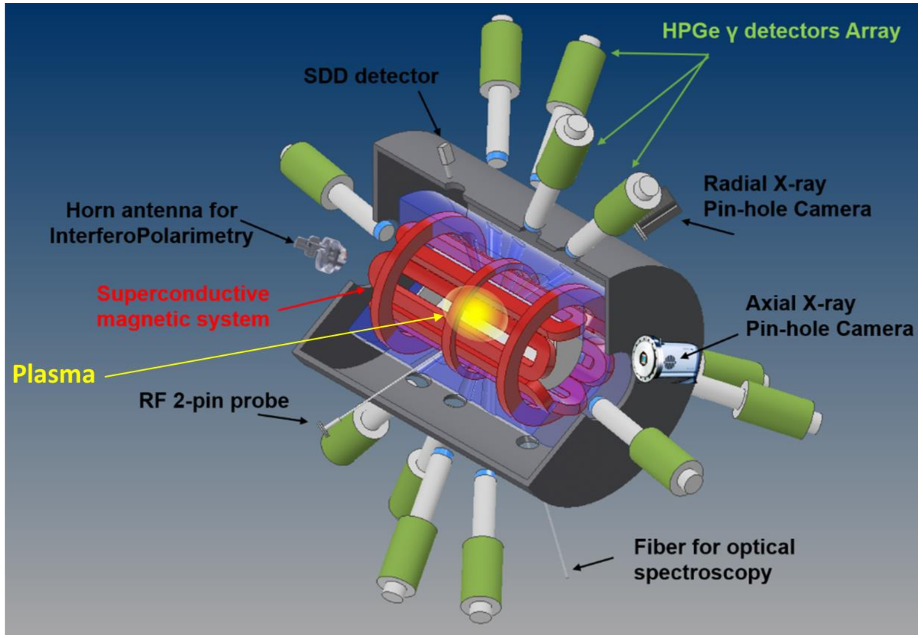


Plasmas for
Astrophysics
Nuclear
Decay
Observation and
Radiation for
Archaeometry

THE PANDORA PROJECT

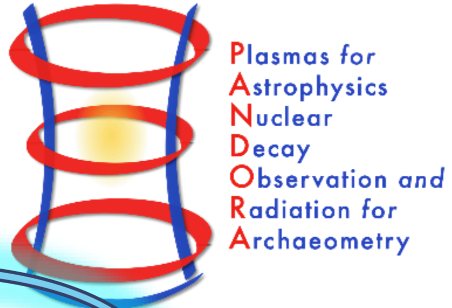


Plasma is excited by Electron-Cyctron-Resonance by microwaves and confined by magnetic fields



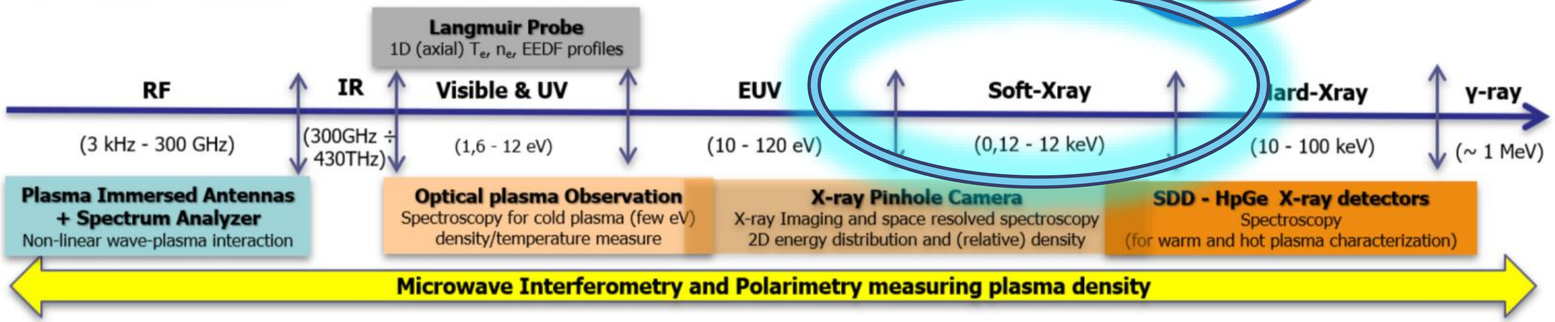
A multidiagnostic system surrounding the plasma chamber was developed to measure plasma parameters

Goal: investigate beta decay in stellar like Electron-Cyclotron-Resonance plasma in compact trap, for fundamental studies in Nuclear Astrophysics and its application.



Method: ECR plasmas emit radiation from microwave to hard X-rays and this radiation can be used to investigate plasma parameters in different regimes

Plasma Emitted Radiation



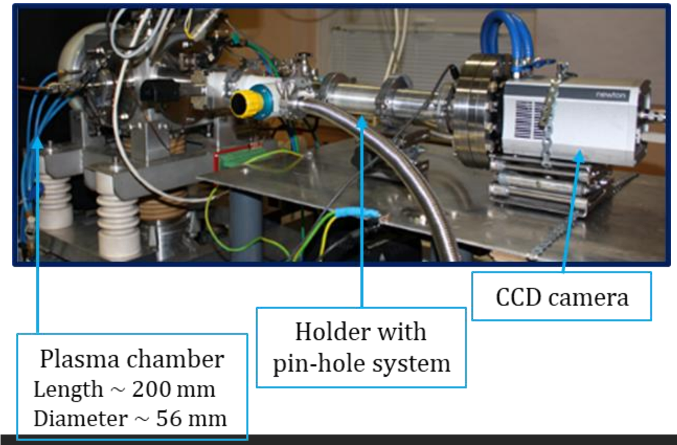
➢ Experimental Set-up

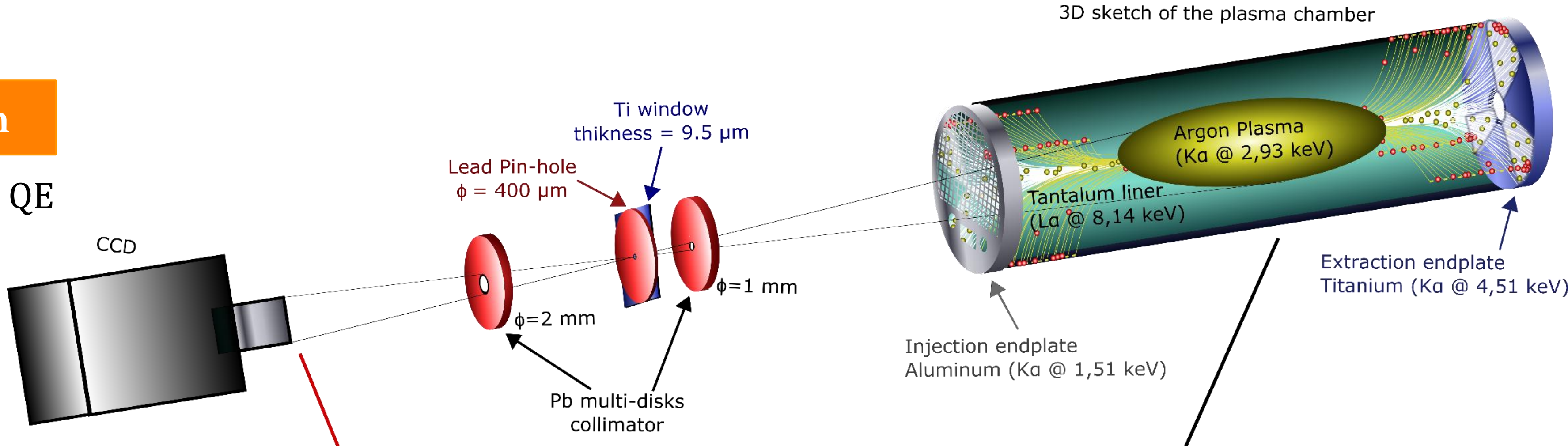
CCD Camera & Pin-hole System

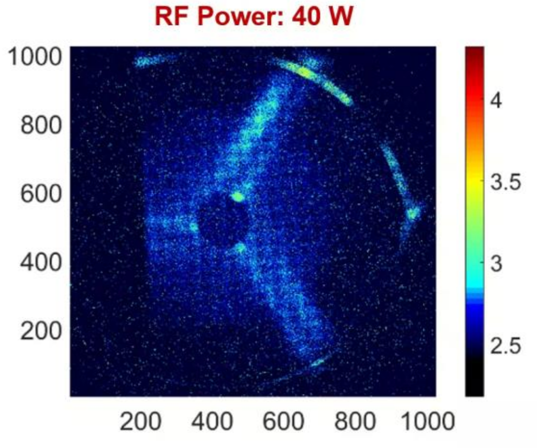
- Sensitivity range ~0,5 ÷ 30 keV (>95% QE within the range)
- Sensor Size: 2,76 cm x 2,76 cm(2048x2048 Pixels)
- Pixel size: 13.5 μm x 13.5 μm
- Lead Pin-hole (diameters 400 μm)
- Lead multi-disks collimator system to reduce the scattering noise, increasing resoltution and signal-to-noise ratio

Energy resolution: 236 eV @ 8 keV

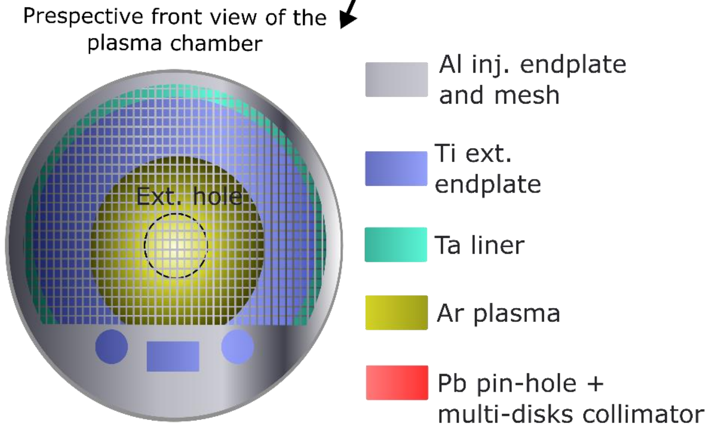
Space resolution: 500μm







"**Live**" **plasma structure and X-ray emission investigations**



Plasmas for
Astrophysics
Nuclear
Decay
Observation and
Radiation for
Archaeometry

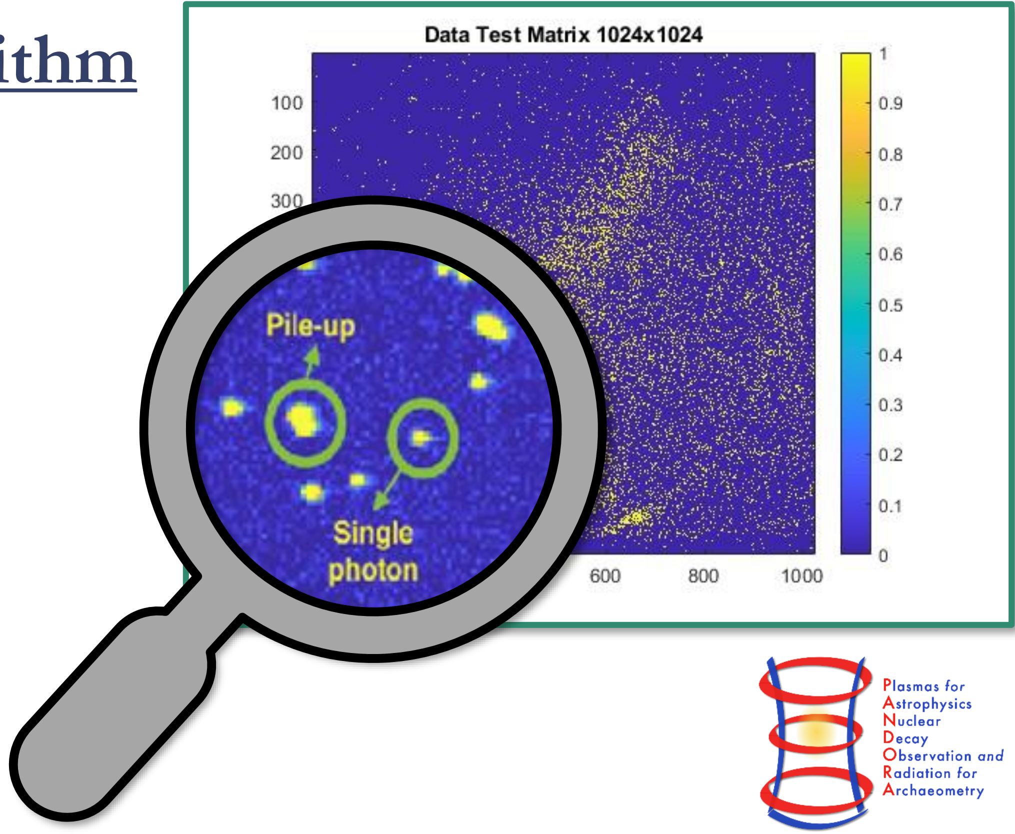
➤ State of the Art Single Photon Counting Algorithm

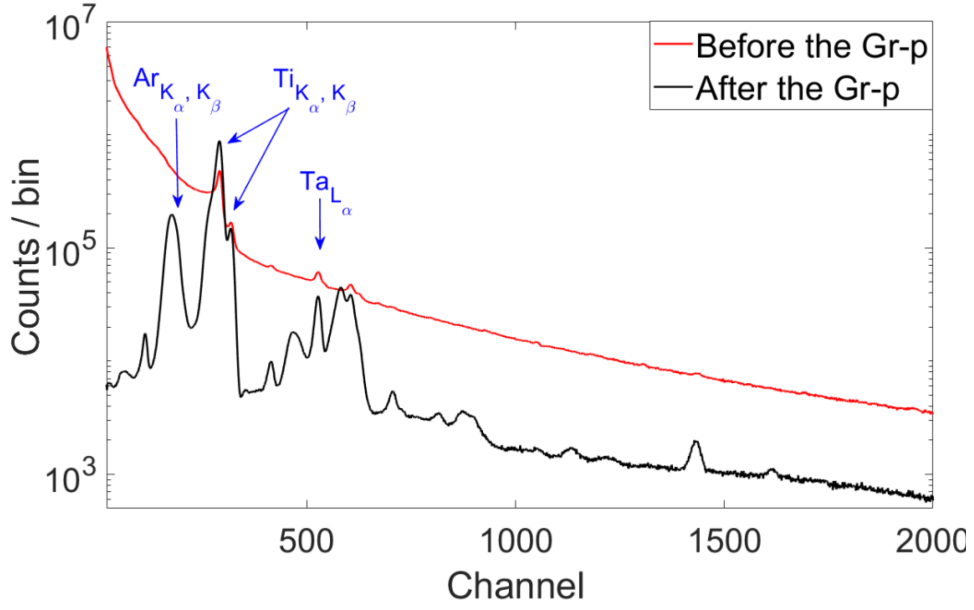
Each pixel becomes an indipendent spectrally-sensitive detector:
SPATIALLY-RESOLVED SPECTROSCOPY pixel-by-pixel

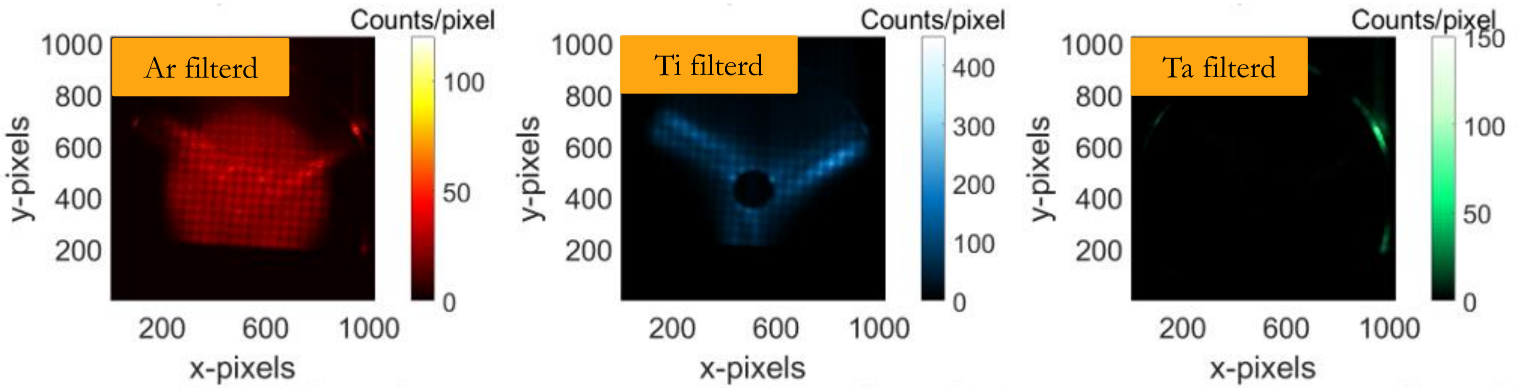
→ **Decoupling of photon number versus energy**
- **very short exposure-time (@ 50-500ms)**
- **thousands of SPhC frames**

→ **Minimize** the **pile-up probability**

→ **Cluster Size Filtering:**
- **Large size (> 5 pixels): multi-photons**
- **Small size (≤ 5 pixels): single-photon**







➢ State of the Art Single Photon Counting Algorithm

Each pixel becomes an indipendent spectrally-sensitive detector:
SPATIALLY-RESOLVED SPECTROSCOPY pixel-by-pixel

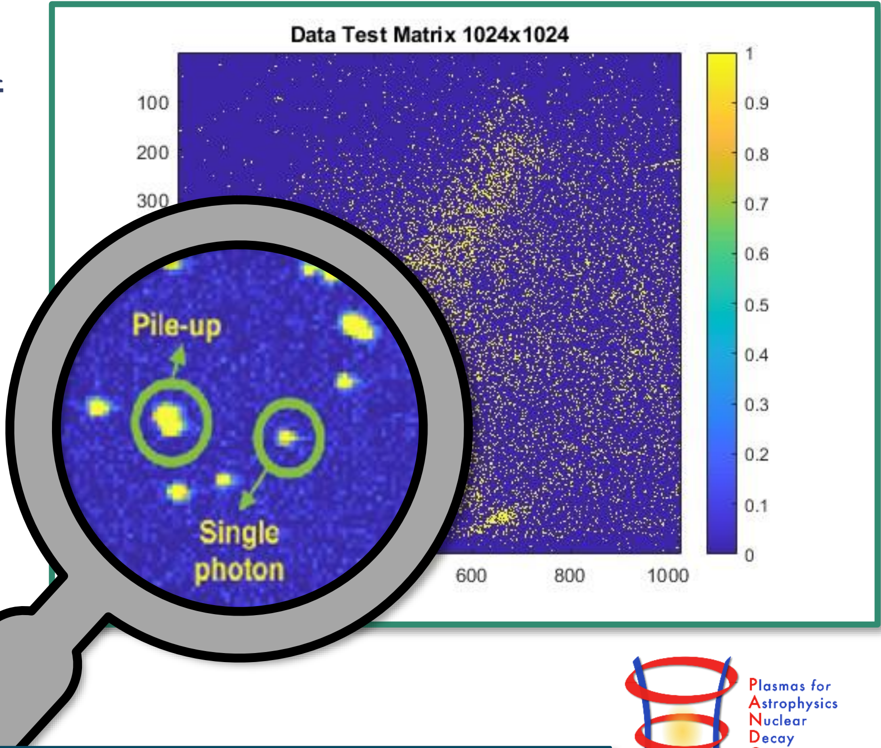
→ **Decoupling of photon number versus energy**

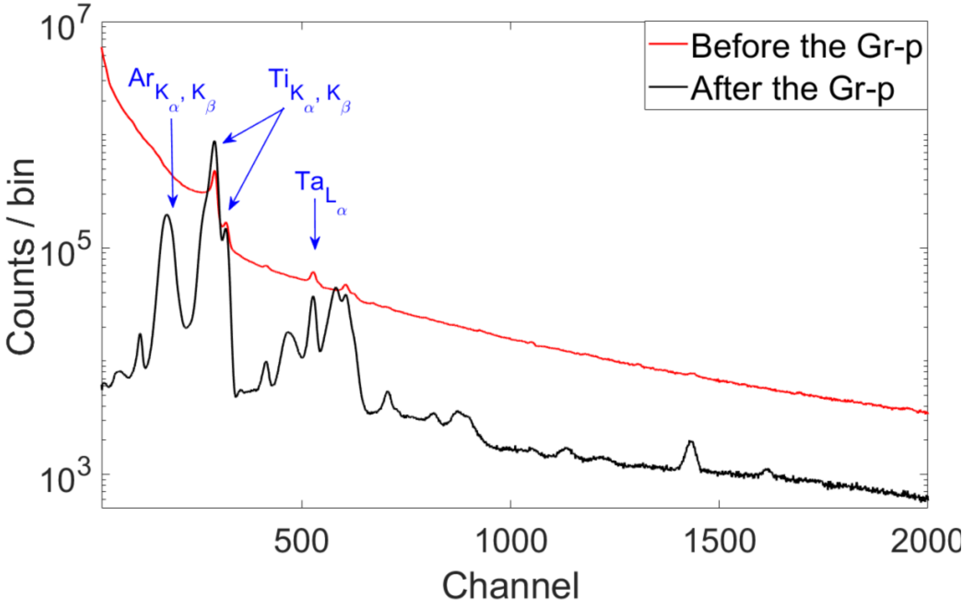
- **very short exposure-time (@ 50-500ms)**
- **thousands of SPhC frames**

→ **Minimize** the **pile-up probability**

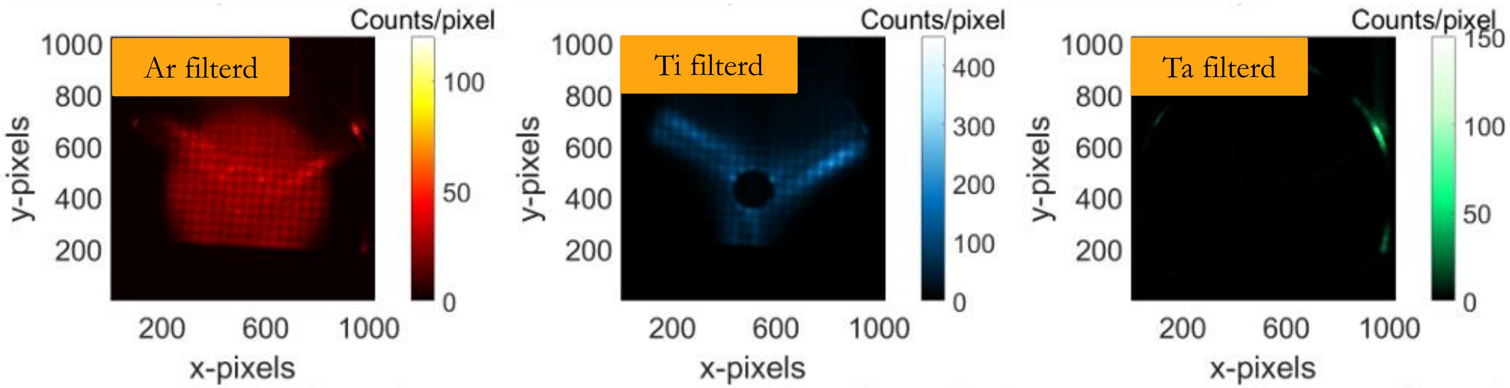
→ **Cluster Size Filtering:**

- **Large size (> 5 pixels): multi-photons**
- **Small size (≤ 5 pixels): single-photon**





Check on Giorgio Finocchiaro’s Talk: Space and time-resolved X-ray spectroscopy technique by CCD pinhole camera



Purposes and Goals

Employing a **Machine Learning Algorithm** allows to:

- **Improve the energy resolution**
- **Improve the signal to noise ratio**
- **Minimize the spurious effect contribution**

Furthermore, it will push towards different goals, such as:

- **Identification of specific patterns**
- **Online Analysis Options**

We decided to use an Hybrid Approach

Previous Algorithm	Un-supervised Model	Supervised Model
Bwconncomp: State of the art tool for image analysis	**Clustering Algorithms:** AI tool for clustering used for pattern recognition, and labelling	**Neural Net:** Supervised model that works with labelled data to learn identify and classify events

First steps:

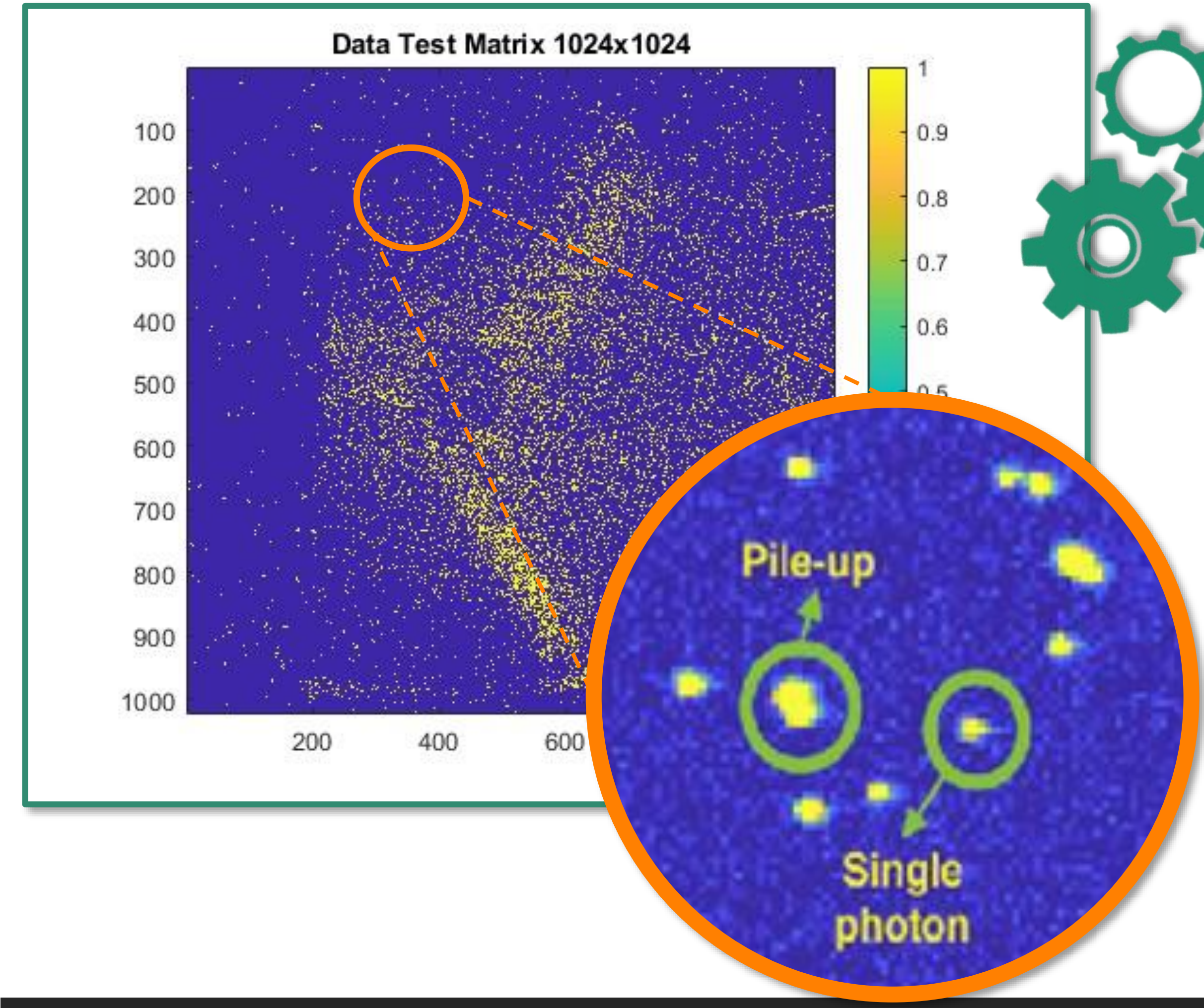
Imaging & Clustering

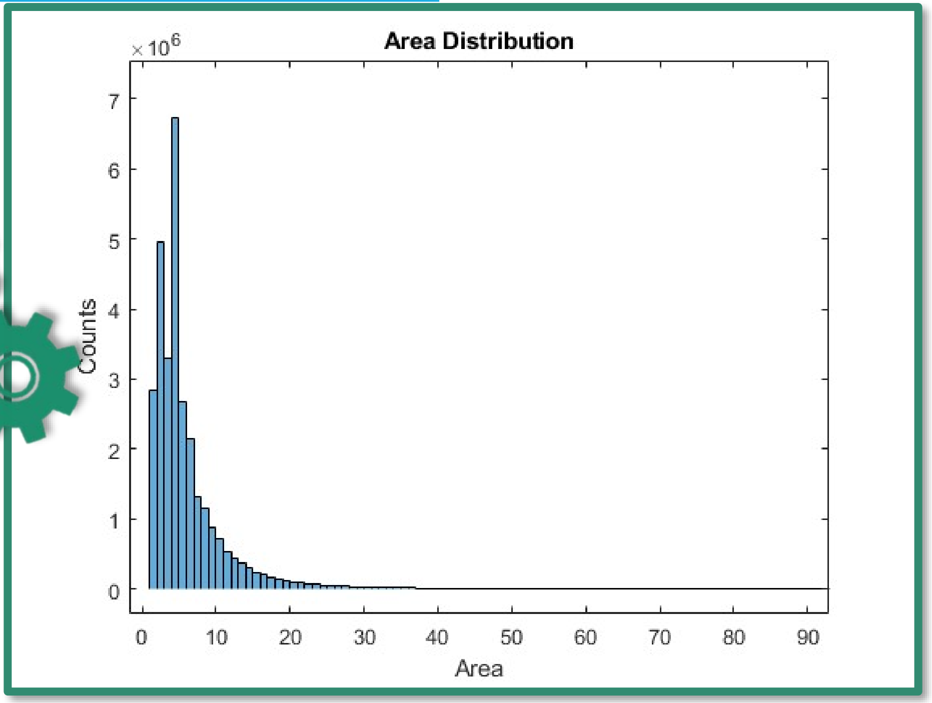
Tools

1° step

Bwconncomp

Connettivity based tool for imaging recognition.





MATLAB

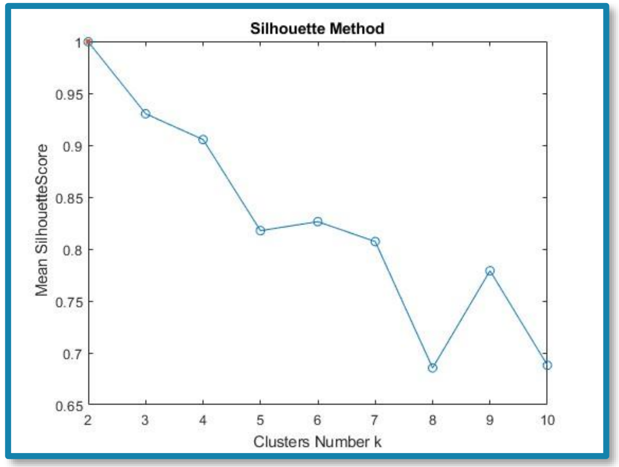
Events	Area	N Local Max	Eccentricity	….
1	20	1	0.3	----
2	4	3	1	----
-----	----	----	----	----

2° step

Clustering Algorithm: K-means

K-means is used to group features based on similarities.

❖ Euristic method for the evaluation of the cluster number in our dataset, based on the mean **Silhouette Score.**



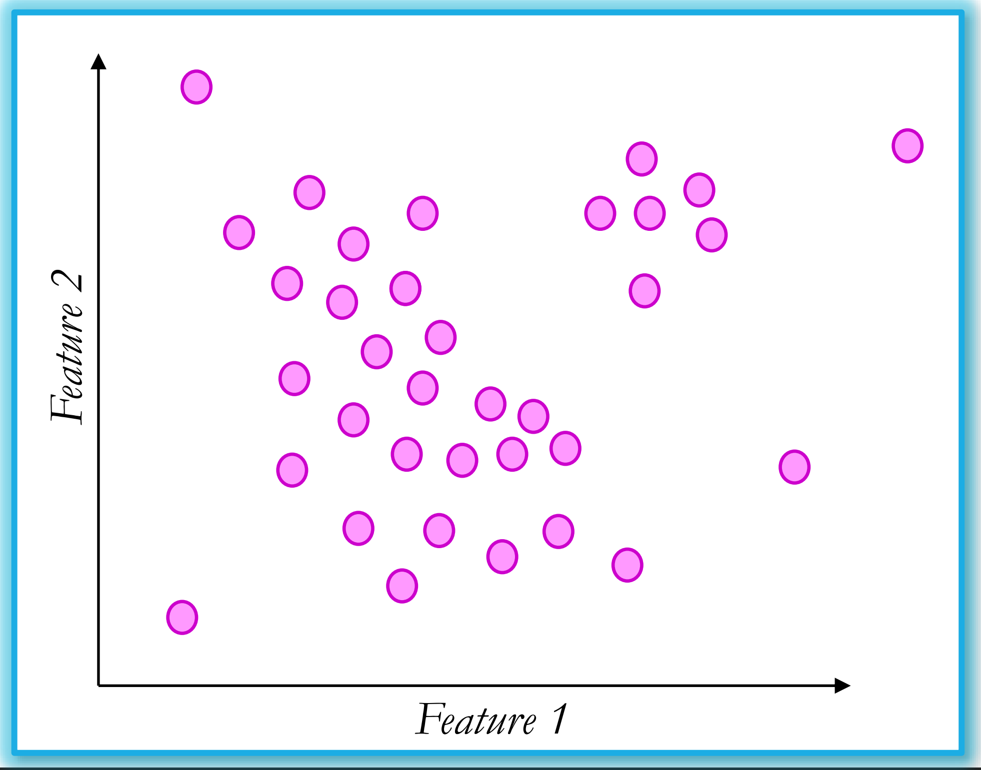
Spheroidal clusters

Distance between objects and centroids

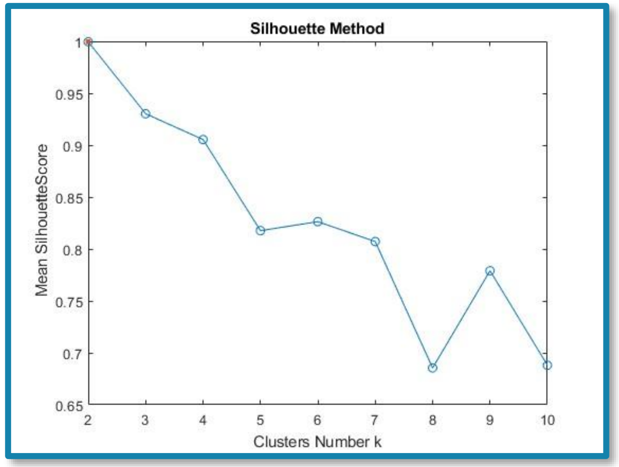
Excellent time efficiency and scalability

Prior Definition of Cluster Number

Strong Dependence on Initialization Conditions



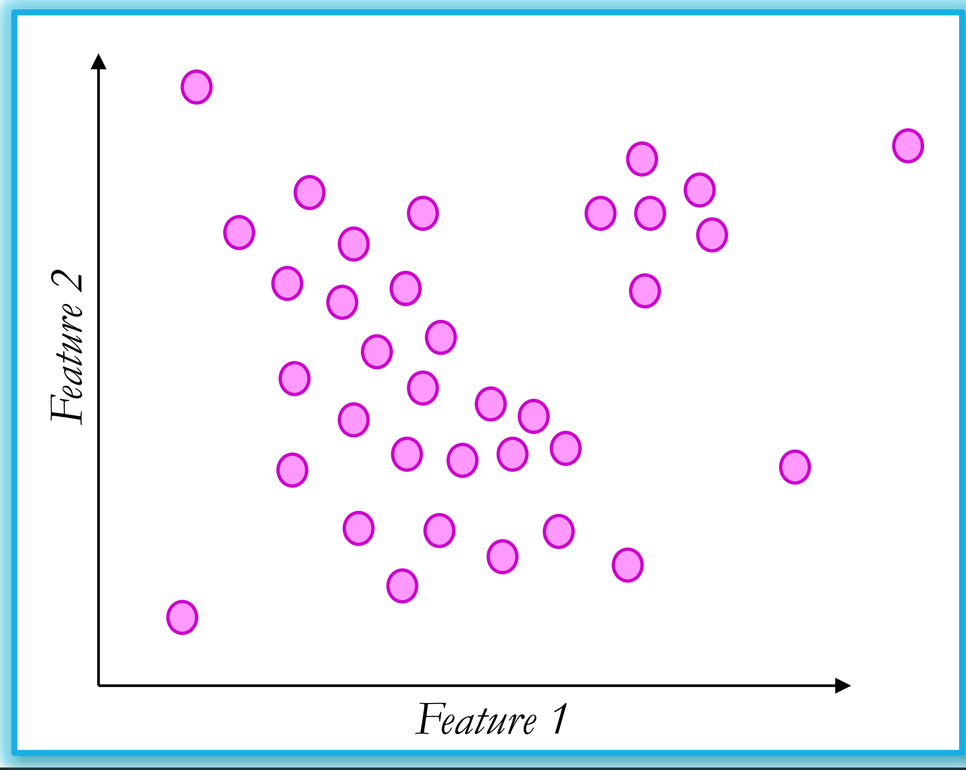
Clustering Algorithm: K-means

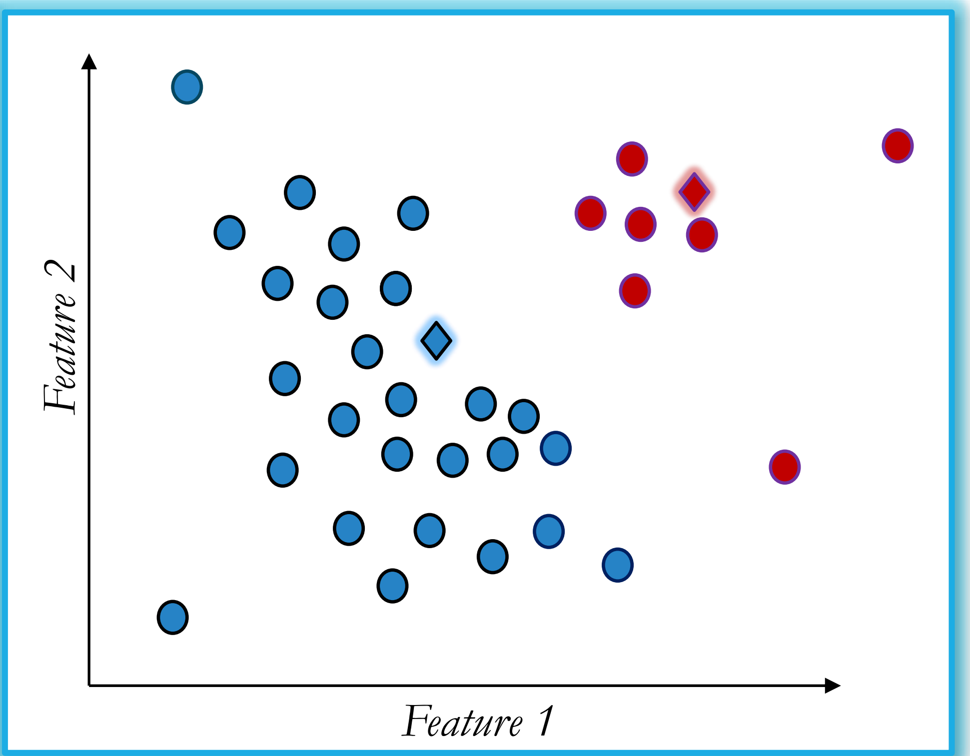


K-means is used to group features based on similarities.

❖ Euristic method for the evaluation of the cluster number in our dataset, based on the mean **Silhouette Score.**

- Spheroidal clusters
- Distance between objects and centroids
- Excellent time efficiency and scalability
- Prior Definition of Cluster Number
- Strong Dependence on Initialization Conditions



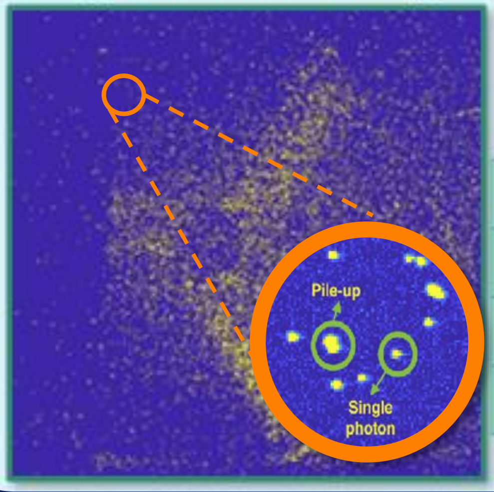


Neural Network

3° step

Neural Network

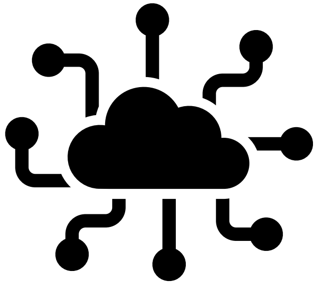
1°step: bwconncomp



Events	Area	N Local Max	....
1	20	1	----
2	4	3	----
-----	----	----	----

Supervised Algorithm

Are specific algortihm, such as the neural networks, both linear and convolutional, that has to be train to work as classifier with the help of a Labeled dataset.

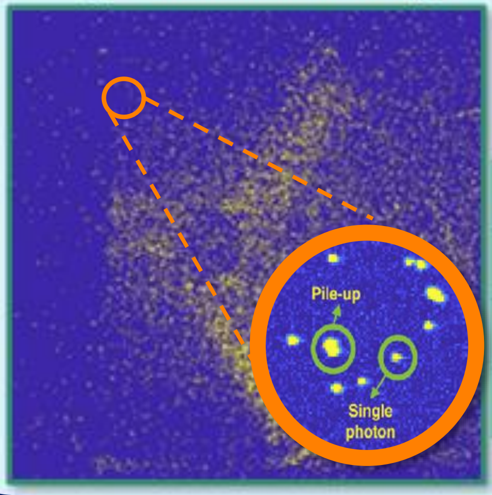


They learn how to make the wanted classification based on the dataset using the label as feedback for the training.

3° step

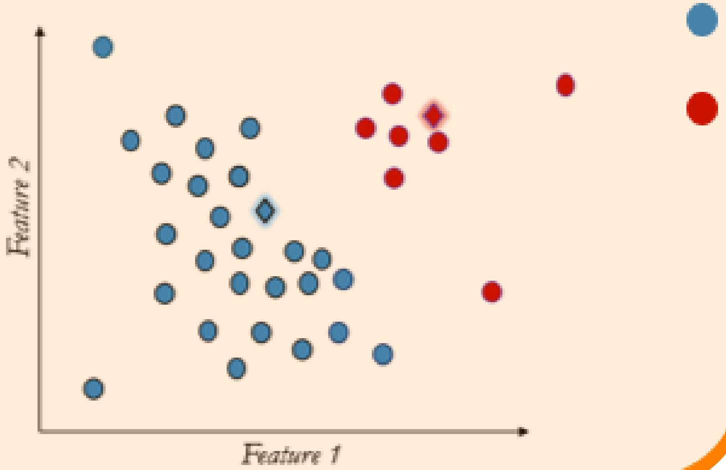
Neural Network

1°step: bwconncomp



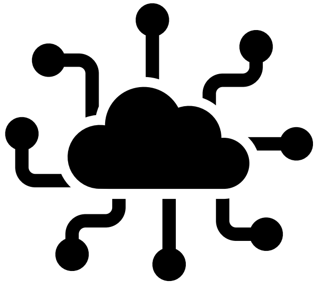
Events	Area	N Local Max	....
1	20	1	----
2	4	3	----
-----	----	----	----

2°step: K-means



Supervised Algorithm

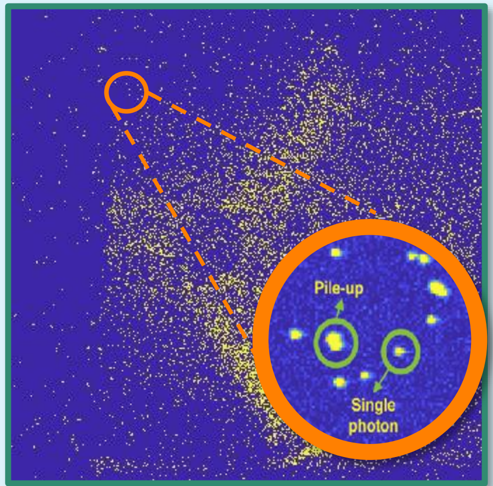
Are specific algortihm, such as the neural networks, both linear and convolutional, that has to be train to work as classifier with the help of a Labeled dataset.



They learn how to make the wanted classification based on the dataset using the label as feedback for the training.

Neural Network

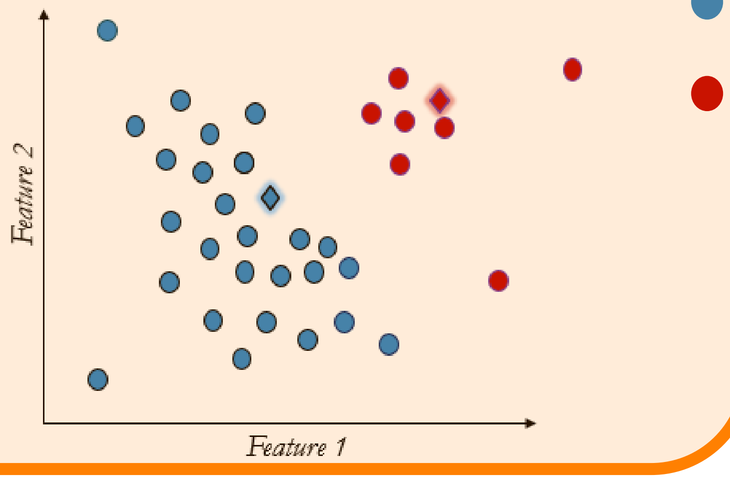
1°step: bwconncomp



Events	Area	N Local Max	….
1	20	1	----
2	4	3	----
-----	----	----	----

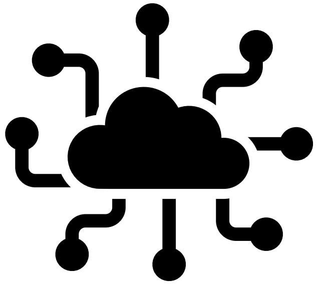
2°step: K-means

K-means



Supervised Algorithm

Are specific algortihm, such as the neural networks, both linear and convolutional, that has to be train to work as classifier with the help of a Labeled dataset.

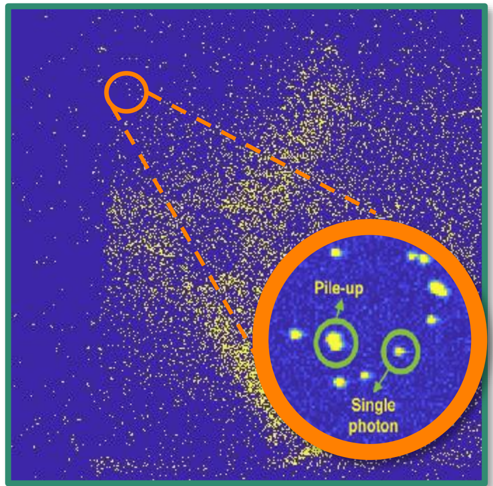


They learn how to make the wanted classification based on the dataset using the label as feedback for the training.

3° step

Neural Network

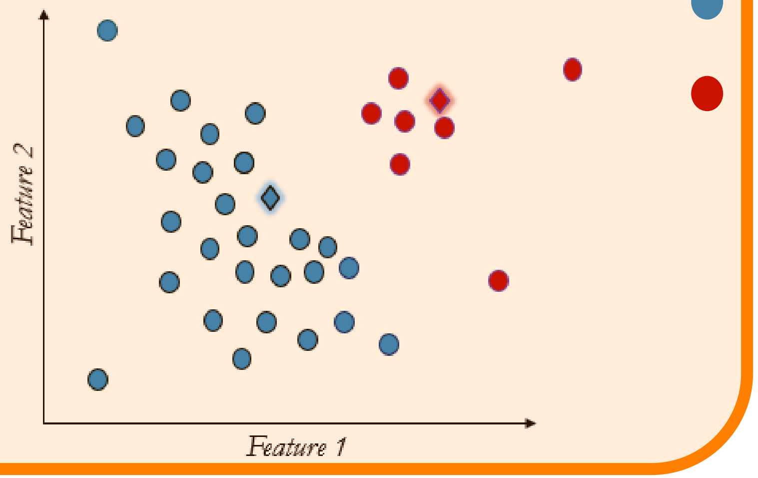
1°step: bwconncomp



Events	Area	N Local Max	….
1	20	1	----
2	4	3	----
-----	----	----	----

2°step: K-means

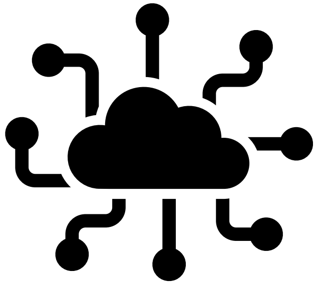
K-means



Supervised Algorithm

Are specific algortihm, such as the neural networks, both linear and convolutional, that has to be train to work as classifier with the help of a Labeled dataset.

Dataset

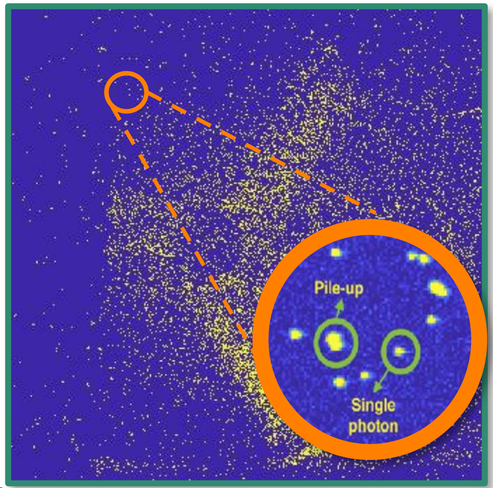


They learn how to make the wanted classification based on the dataset using the label as feedback for the training.

3° step

Neural Network

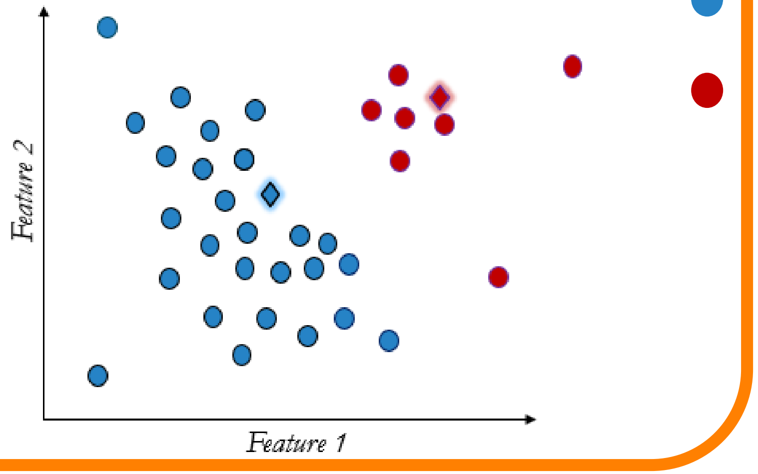
1°step: bwconncomp



Events	Area	N Local Max	….
1	20	1	----
2	4	3	----
-----	----	----	----

2°step: K-means

K-means

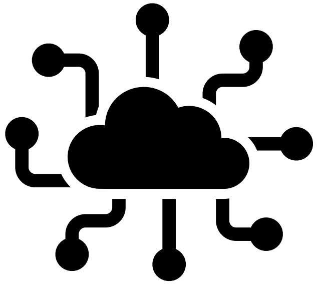


Supervised Algorithm

Are specific algortihm, such as the neural networks, both linear and convolutional, that has to be train to work as classifier with the help of a Labeled dataset.

Dataset

Label

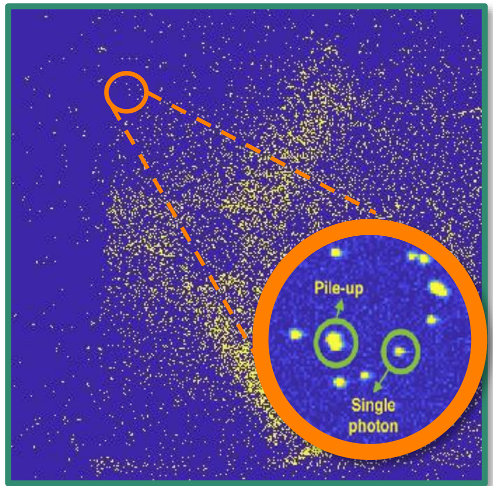


They learn how to make the wanted classification based on the dataset using the label as feedback for the training.

3° step

Neural Network

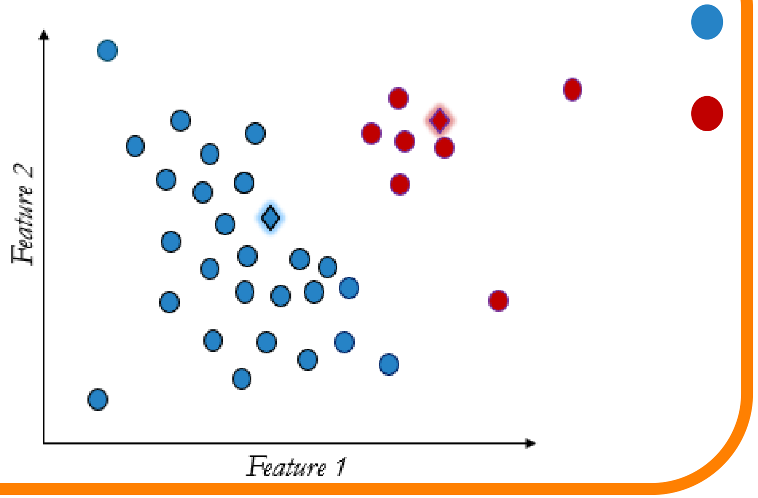
1°step: bwconncomp



Events	Area	N Local Max	….
1	20	1	----
2	4	3	----
-----	----	----	----

2°step: K-means

K-means

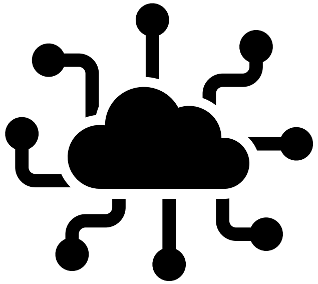


Supervised Algorithm

Are specific algortihm, such as the neural networks, both linear and convolutional, that has to be train to work as classifier with the help of a Labeled dataset.

Dataset

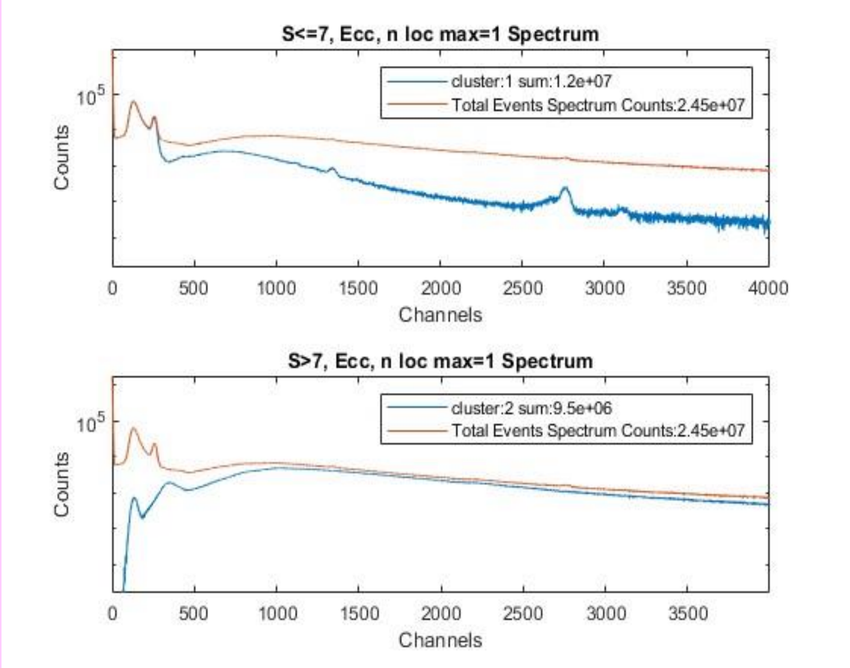
Label



Classification

They learn how to make the wanted classification based on the dataset using the label as feedback for the training.

Data Preparation and Labelling



Gas mixing configuration: only Kr
CCD_plasma_13_11_24
Atomki Lab- Debrecen Hungary

N Events

Dataset			Labels
Number Of Local Max	Clustersize	Eccentricity	K-means IDX
…	…	…	…
…	…	…	…

The most important parameter for the label determination

Main k-means parameter for events discrimination

Binary distinction preliminarily

Data **Preparation** and **Labelling**

Normalization needed

Normalization function optimized for the neural network called **mapminmax.**

Dataset

Number Of Local Max	Clustersize	Eccentricity
…	…	…
…	…	…

Labels

K-means IDX
…
…

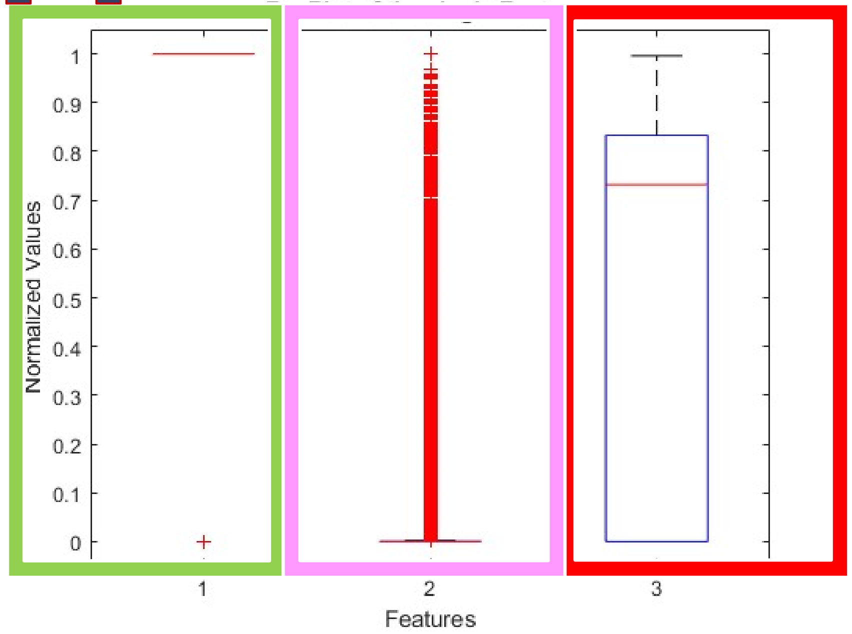
N Events

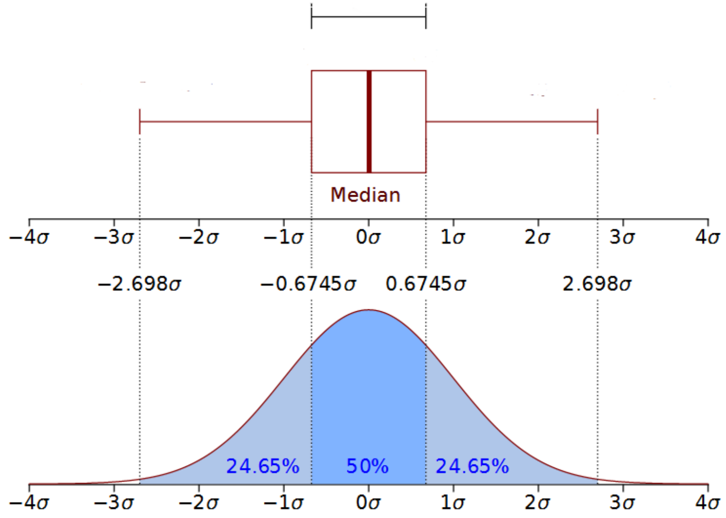
We made it into a binary values

Normalized between 0 and 1

Ranging from 0 to 1

Careful evaluation of the distribution





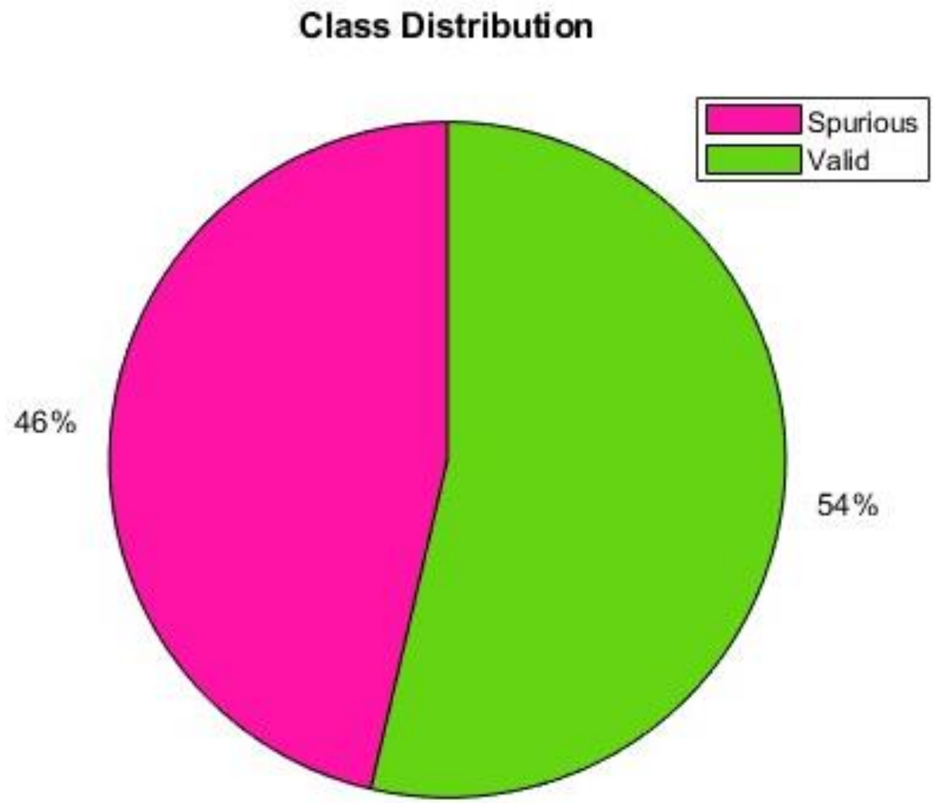
Data Preparation and Labelling

Normalization needed

Normalization function optimized for the neural network called **mapminmax.**

Balanced Dataset

A **binary variable** was created, employing the results coming from the k-means output, i.e. the **cluster index**

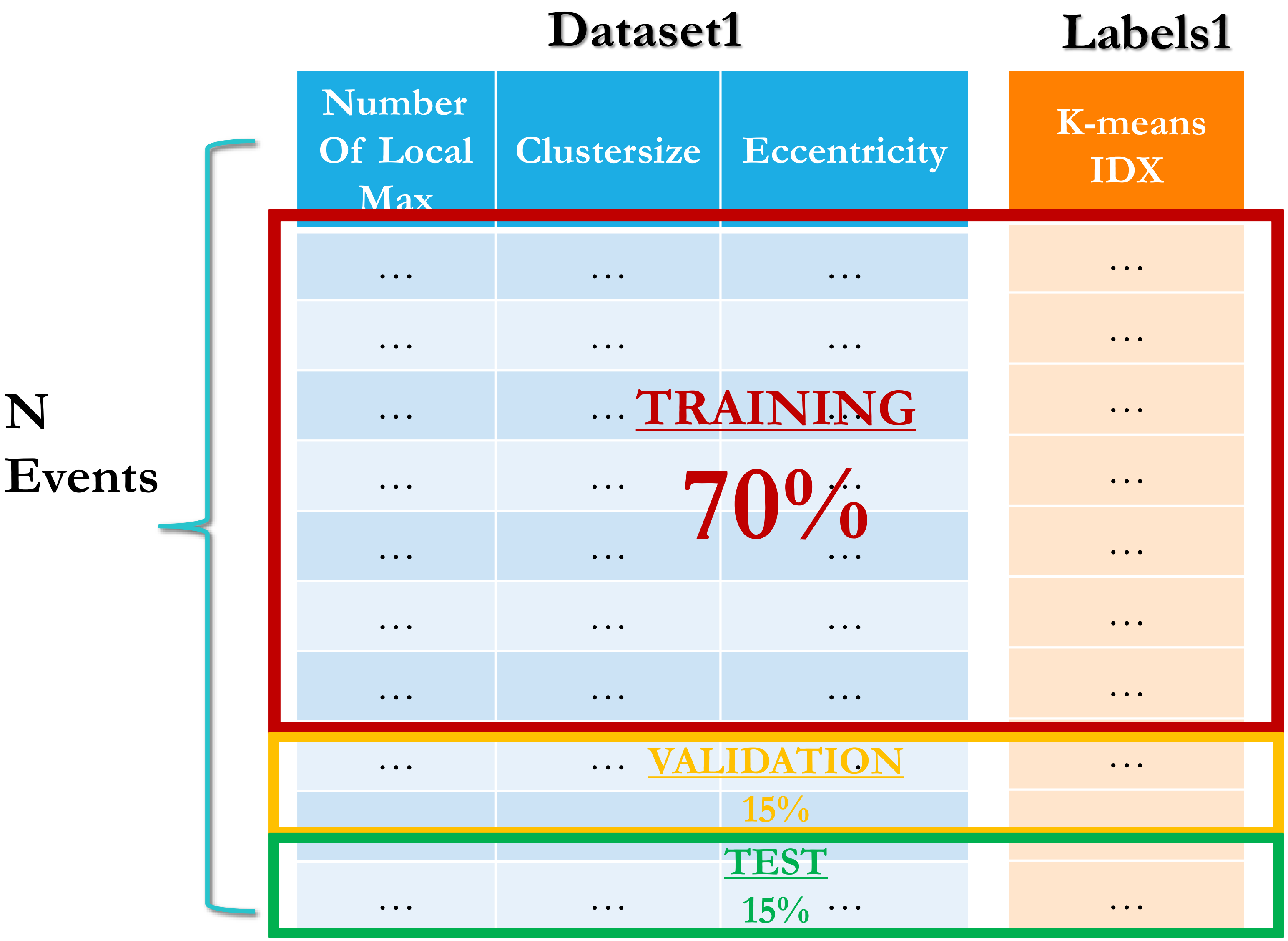


Dataset | **Labels**

N Events

Number Of Local Max	Clustersize	Eccentricity	K-means IDX
…	…	…	…
…	…	…	…
We made it into a binary values	Normalized between 0 and 1	Ranging from 0 to 1	

Data Organization



Used to **train** the net, updating the weight between the layers based on these data

Used **during training** to monitor and track the net performances: **overfitting, learning**

Used **after training** to evaluate the net performances: on new data.

➢ Data Organization

Dataset1 | **Labels1**

Number Of Local Max	Clustersize	Eccentricity	K-means IDX
…	…	…	…
…	…	…	…
…	…	…	…
…	…	…	…
…	…	…	…
…	…	…	…
…	…	…	…
…	…	…	…
…	…	…	…

N Events

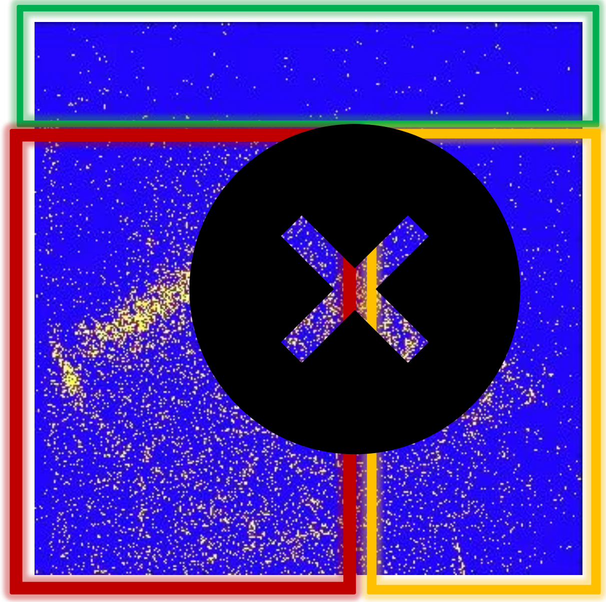
TRAINING 70%

VALIDATION 15%

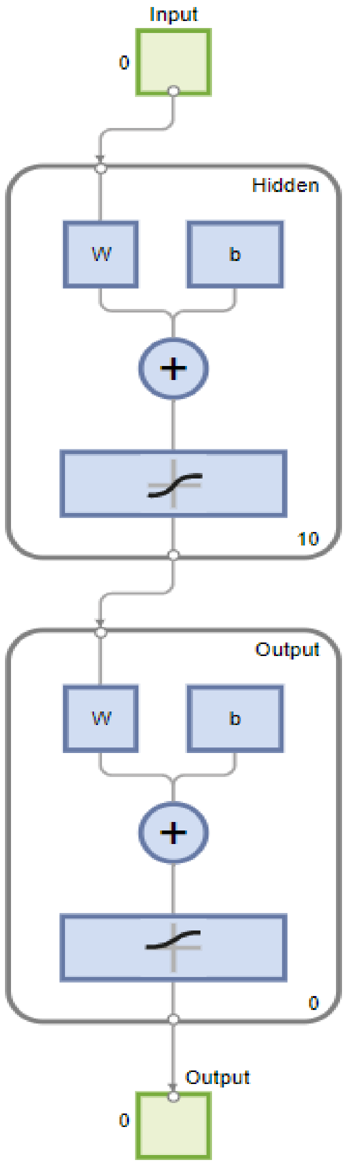
TEST 15%

Randomization Needed

Allows for a more efficient training.
It's a particularly usefull procedure to obtain a rappresentative dataset division.



Net Development



Feed-Forward Net [PATTERNET], which is the simplest neural net, easy to built and to train with specific charcateristic:

- It flows in one single direction, without any cycle or feedback connection
- Neuron Layers Connected via algorithm-evaluated weights

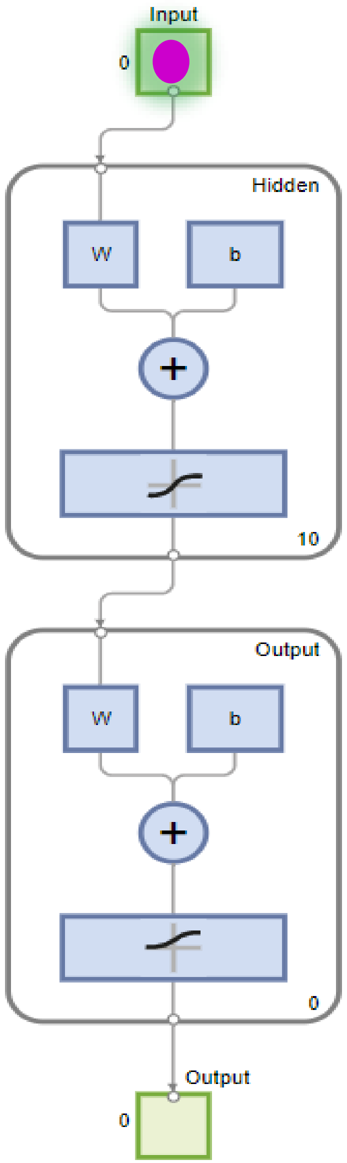
TYPICAL STRUCTURE

Input Layer

Hidden Layers

Output Layer

Net Development



Feed-Forward Net [PATTERNET], which is the simplest neural net, easy to built and to train with specific charcateristic:

- It flows in one single direction, without any cycle or feedback connection
- Neuron Layers Connected via algorithm-evaluated weights

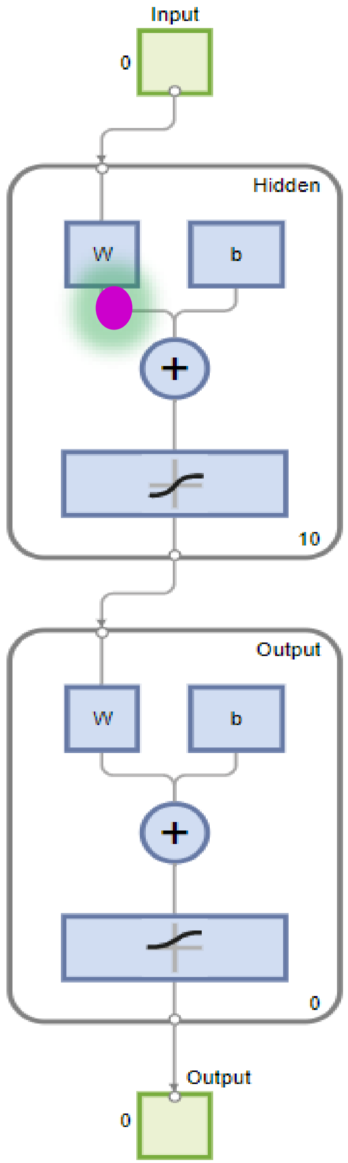
TYPICAL STRUCTURE

Input Layer → Input **data**, **organized** and classified in train, validation and test, **labeled**.
In our case we will have 3 neurons as we employed 3 features

Hidden Layers

Output Layer

➢ Net Development



Feed-Forward Net [PATTERNET], which is the simplest neural net, easy to built and to train with specific charcateristic:

- It flows in one single direction, without any cycle or feedback connection
- Neuron Layers Connected via algorithm-evaluated weights

TYPICAL STRUCTURE

Input Layer

Hidden Layers →

Output Layer

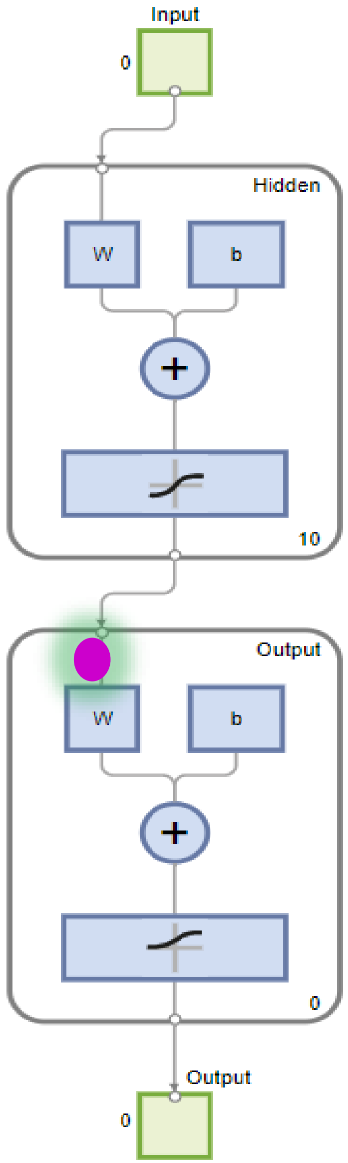
The hidden neurons layers act on the data weighting them based on specific **activation function** depending upon the training algorithm, modulating non linear relationship between data

The non-linearity of the model is introduced by two parameter the weight W and the bias b.

$$W_{\{t+1\}} = W_t + \lambda_t * p_t$$
$$b_{\{t+1\}} = b_t + \lambda_t * p_t$$

Where λ is the adaptive step, and p the conjugate direction

➢ Net Development



Feed-Forward Net [PATTERNET], which is the simplest neural net, easy to built and to train with specific charcateristic:

- It flows in one single direction, without any cycle or feedback connection
- Neuron Layers Connected via algorithm-evaluated weights

TYPICAL STRUCTURE

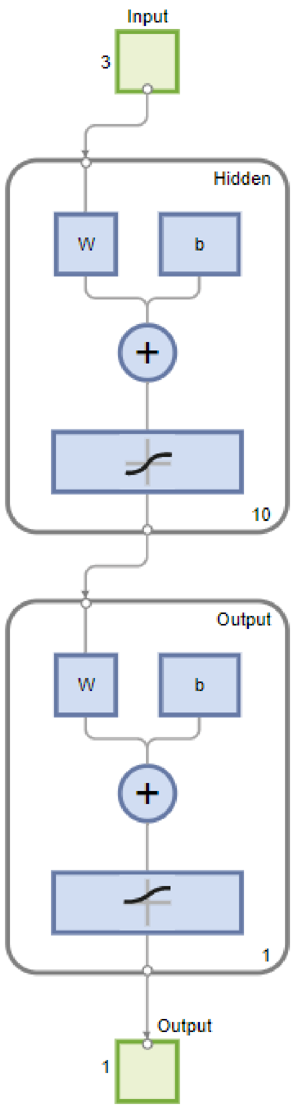
Input Layer

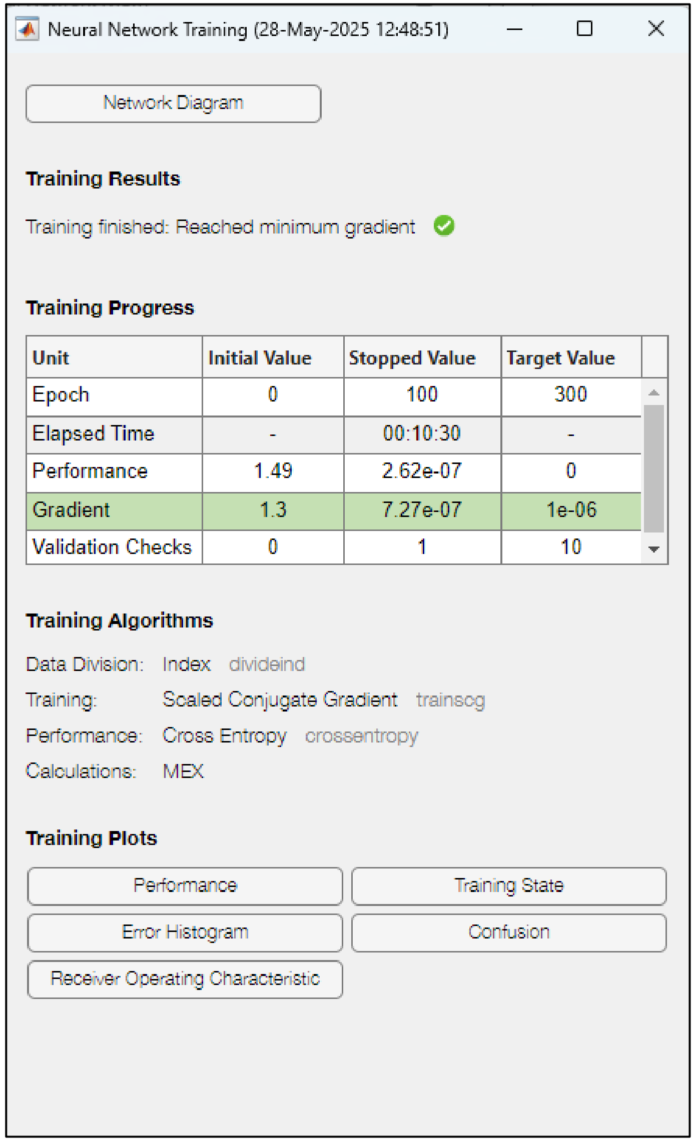
Hidden Layers

Output Layer →

Once in **the 1-neuron Output Layer**, the information have to be processed by the default sigmoid function into a interpretable results, such as the **probability** of belonging to a specific class (mainly used for binary classification)

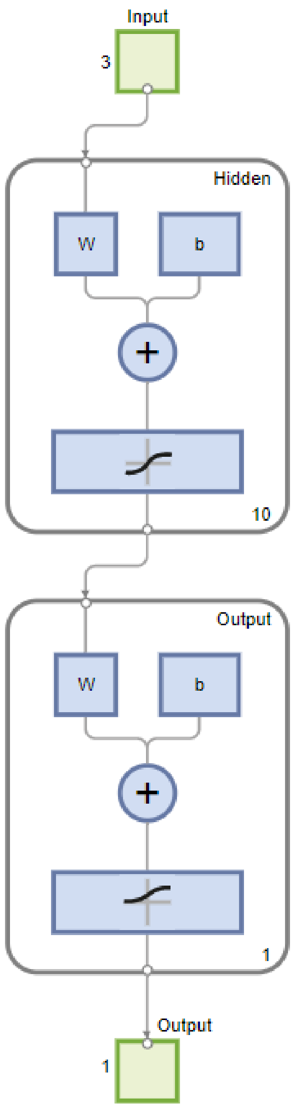
➢Net Training

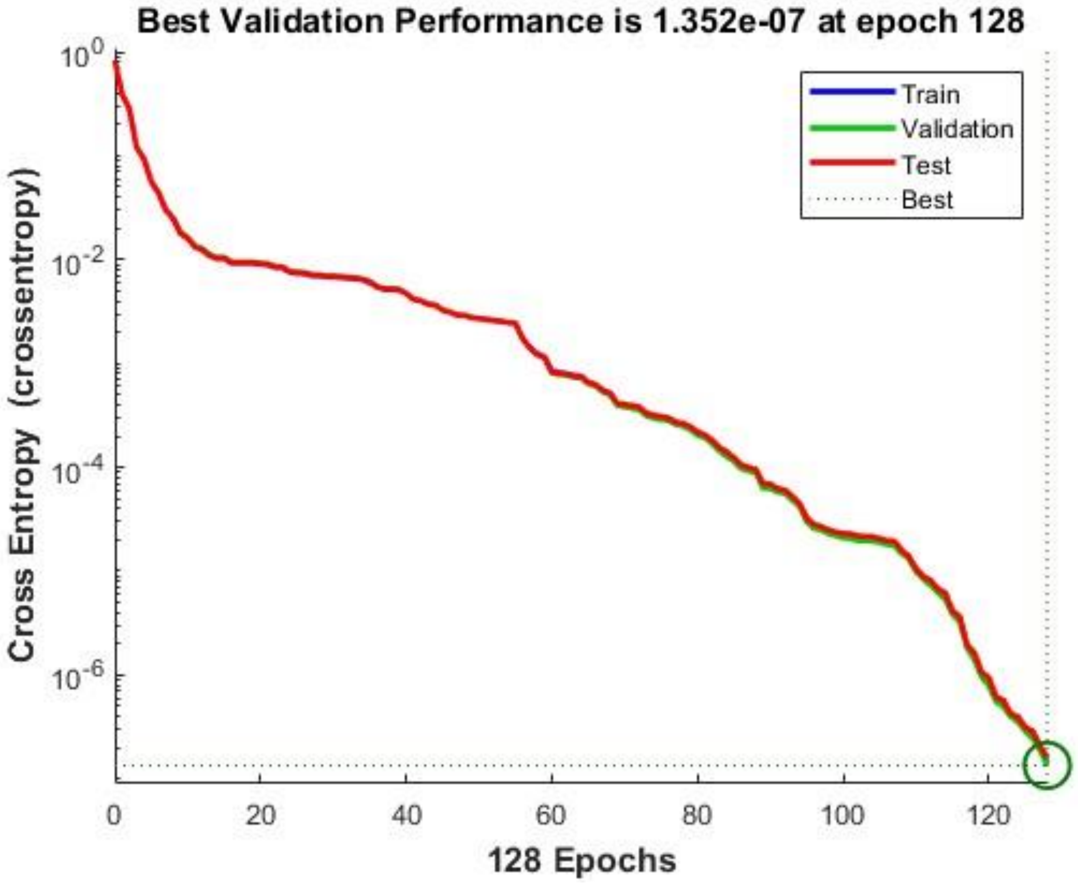


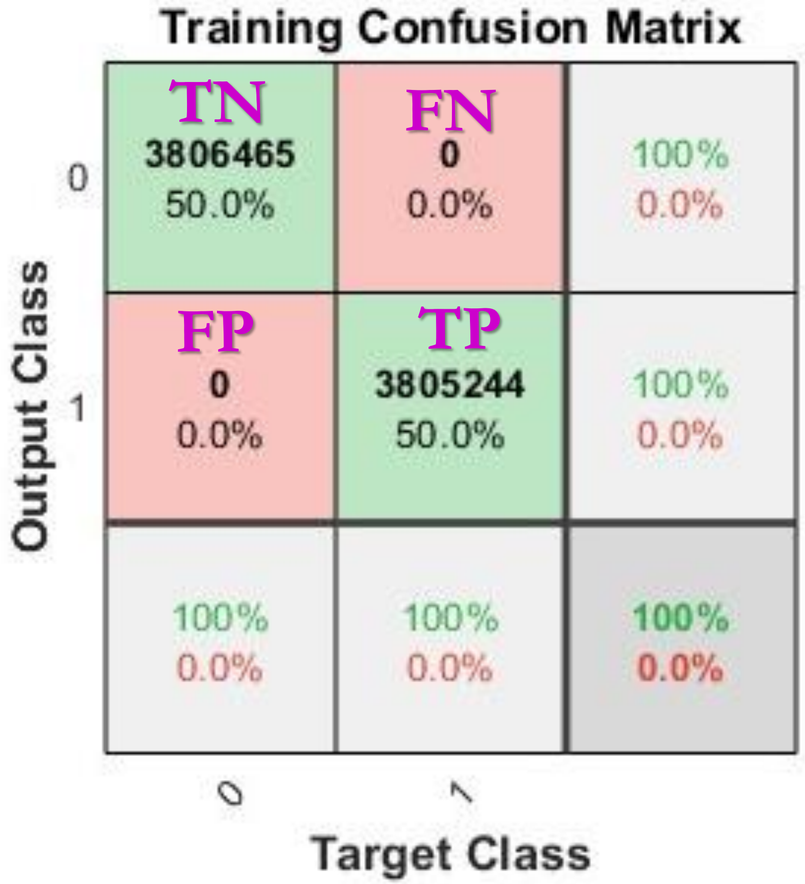


Parameters	Value
Neurons in the Hidden Layer	10
Training Algorithm	SCG
Number of Epoch	300
Number of Events	7767611
Activation Function	Tansig
Activation Function Output	Logsig
Minimum Gradient	1 e-6
Maximum Fails	10
Loss Function	Cross Entropy

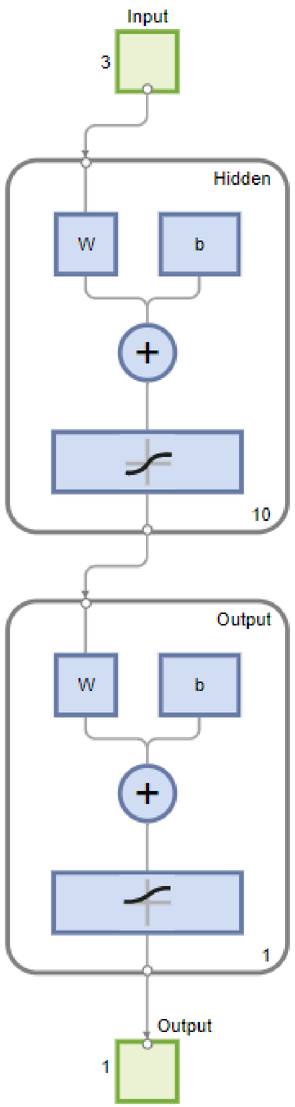
➢Net Training

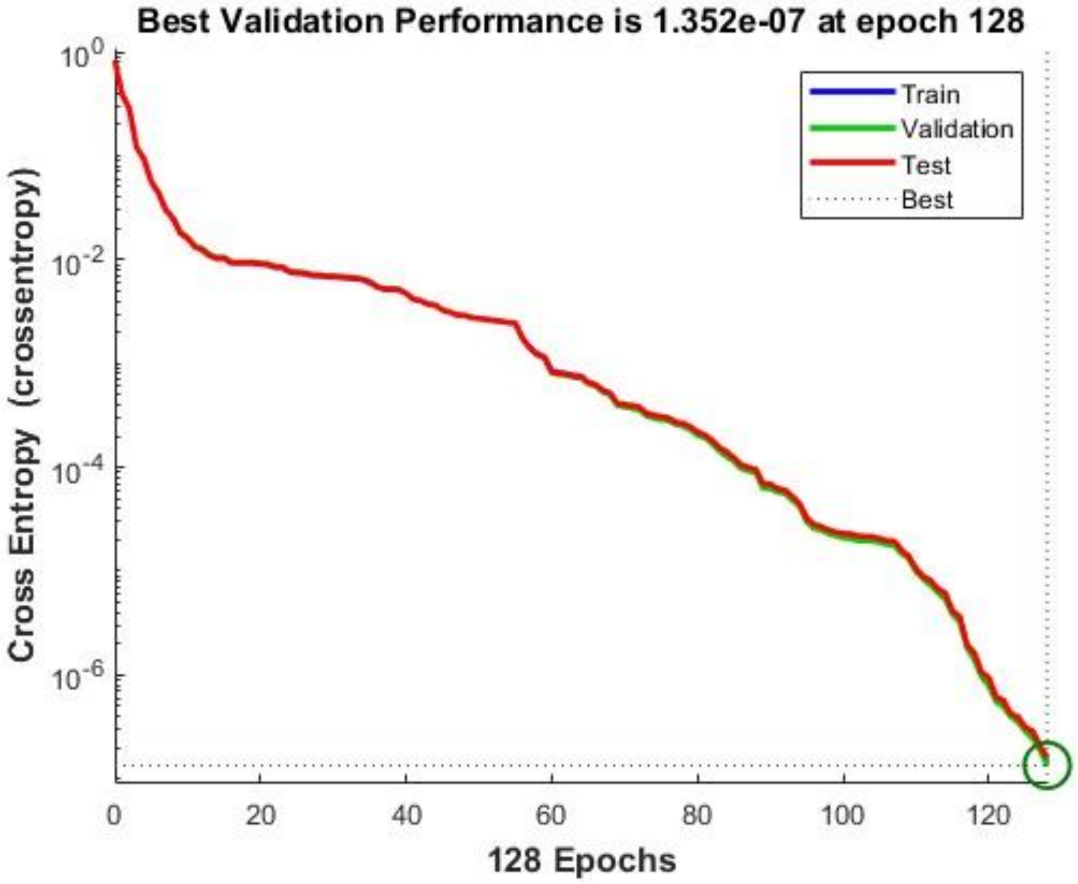




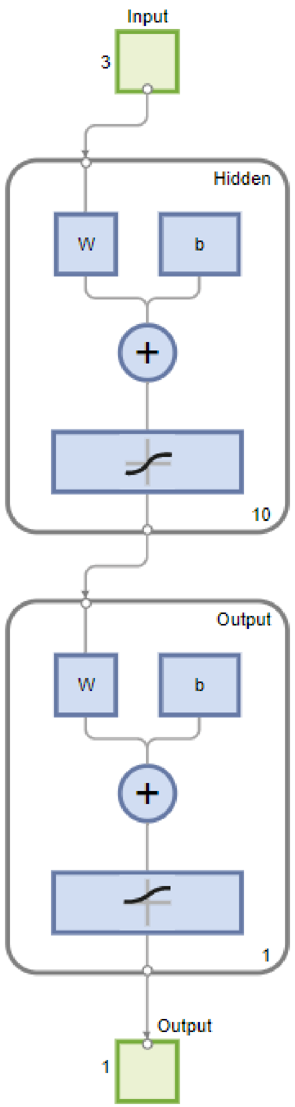


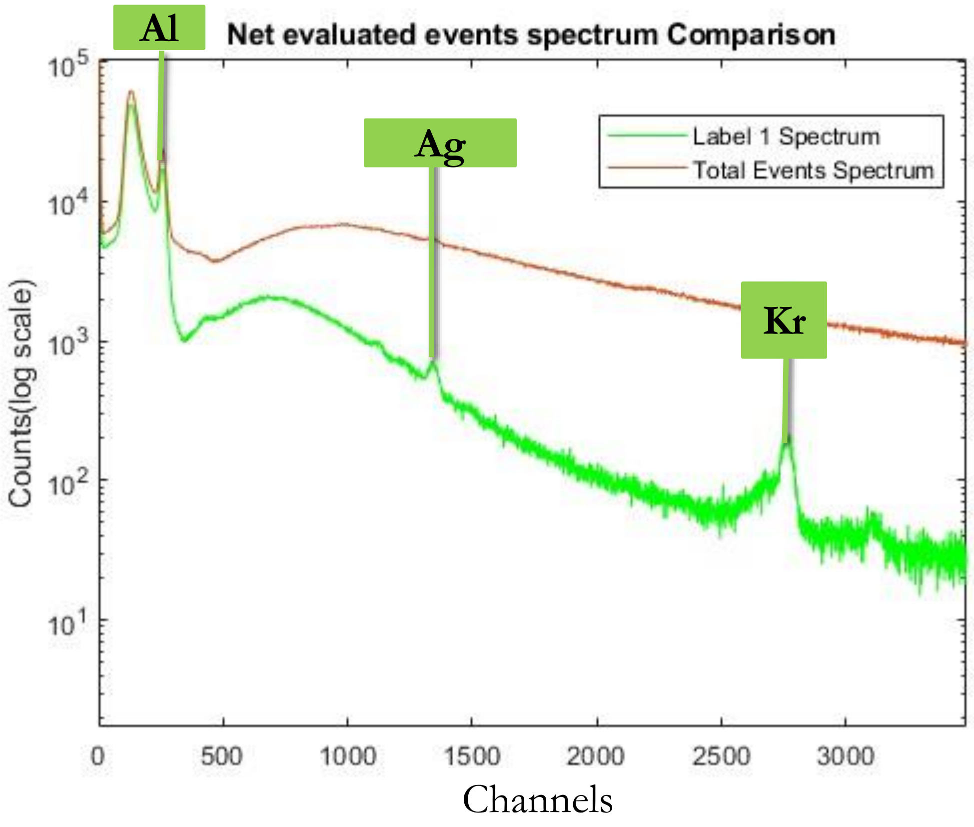
➤Net Training

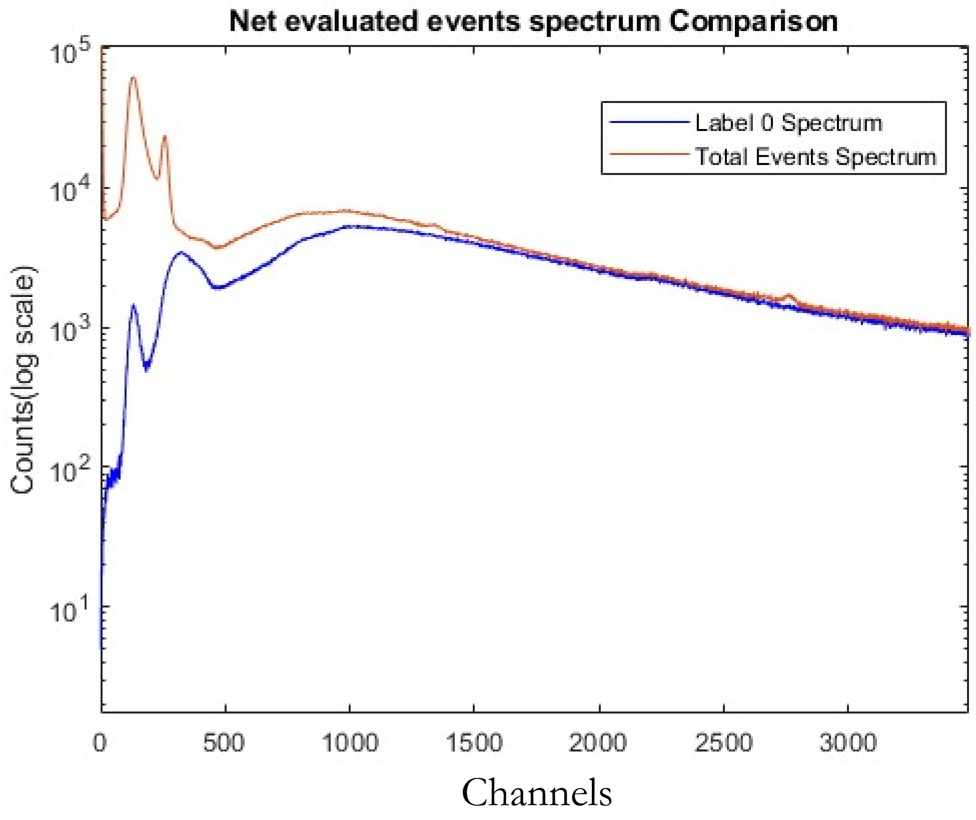




➤First Results







2% FWHM
@ Kr peak

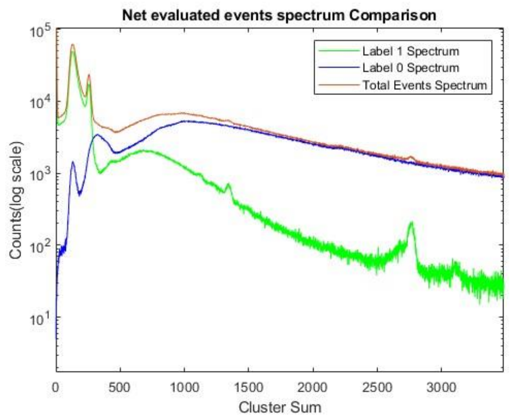
Conclusions &

Perspectives

➢Conclusions & Perspectives

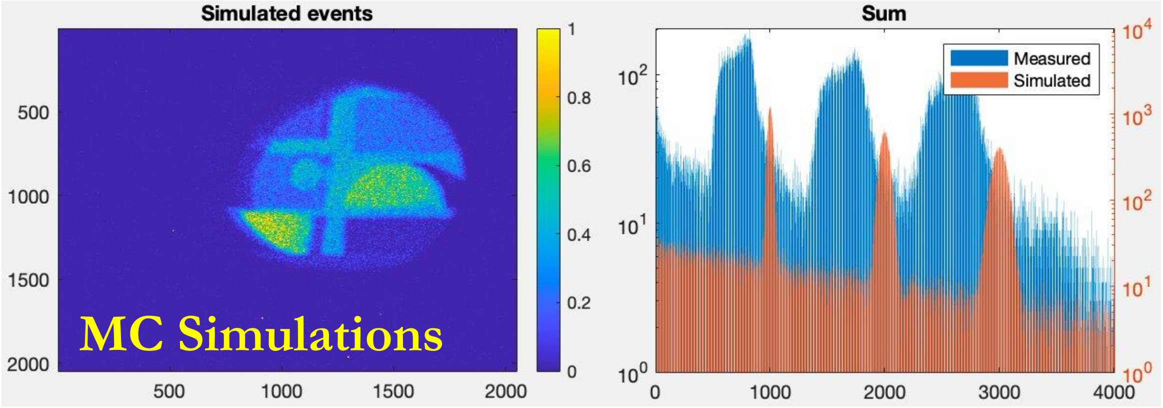
ACHIEVEMENTS

- **Validation** of the AI based unsupervised Algorithm K-means
- **Development and Training** of a Feed-Forward Neural Network
- **First Results** on the data coming from the last experimental Campaign



TO DO LIST

- **Test of the Methods performances** on a new dataset
- **Further training of the Net** on different dataset
- **Test of the Net Performance** on simulated data



Thanks for the Attention

HIGH PRECISION X-RAY MEASUREMENTS 2025

JUN 16 – 20, 2025

LABORATORI NAZIONALI DI FRASCATI INFN

SAPIENZA UNIVERSITÀ DI ROMA

INFN LNS

Plasmas for
Astrophysics
Nuclear
Decay
Observation and
Radiation for
Archaeometry

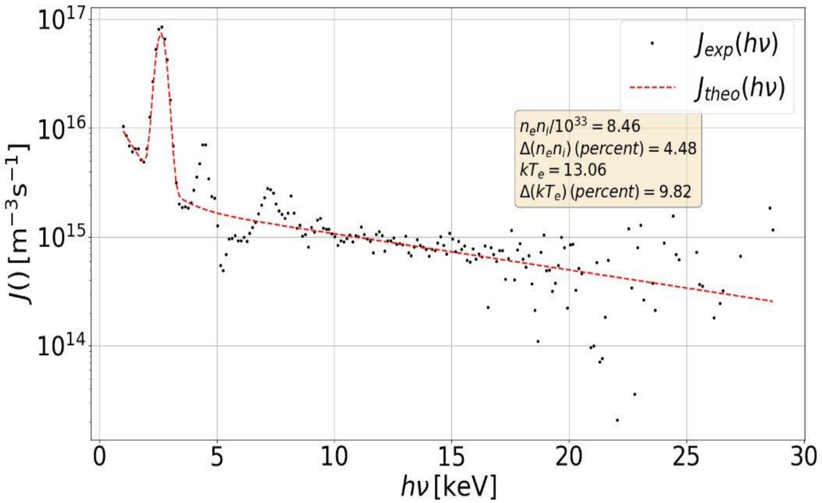
Back-up Slides

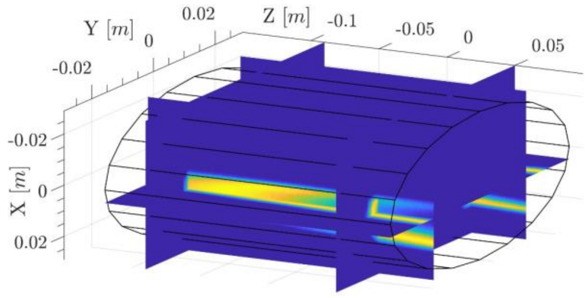
➢ Thermodynamic Parameters & Emissivity models

Soft X-ray spectrum can be converted into X-ray **emissivity density** ***J(hν)*** by calculating the emissivity volume V_P and geometrical efficiency Ω_g

$$J(h\nu) = h\nu \frac{N^p(h\nu)}{t} \frac{4\pi}{\Delta E V_P \Omega_g}$$

Total energy emitted per unit time, volume and energy, through photons of energy hν





Warm Electron population (0.5-30 keV)

Emissivity Model

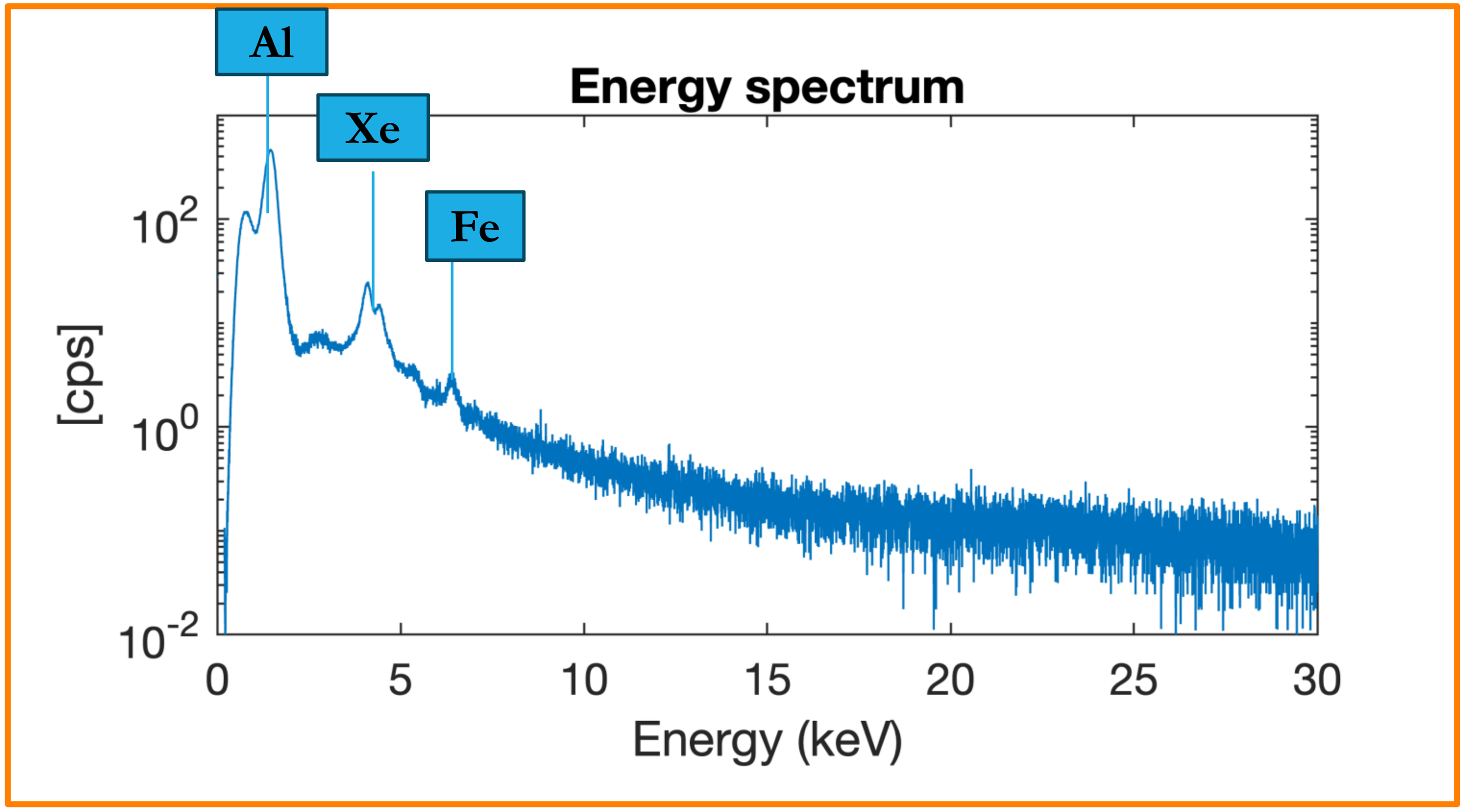
The plasma X-ray spectrum can be decomposed into continuous bremsstrahlung and discrete line emission, so de emissivity density: **$J_{theo}(h\nu) = J_{theo,brem}(h\nu) + J_{theo,line}(h\nu)$**

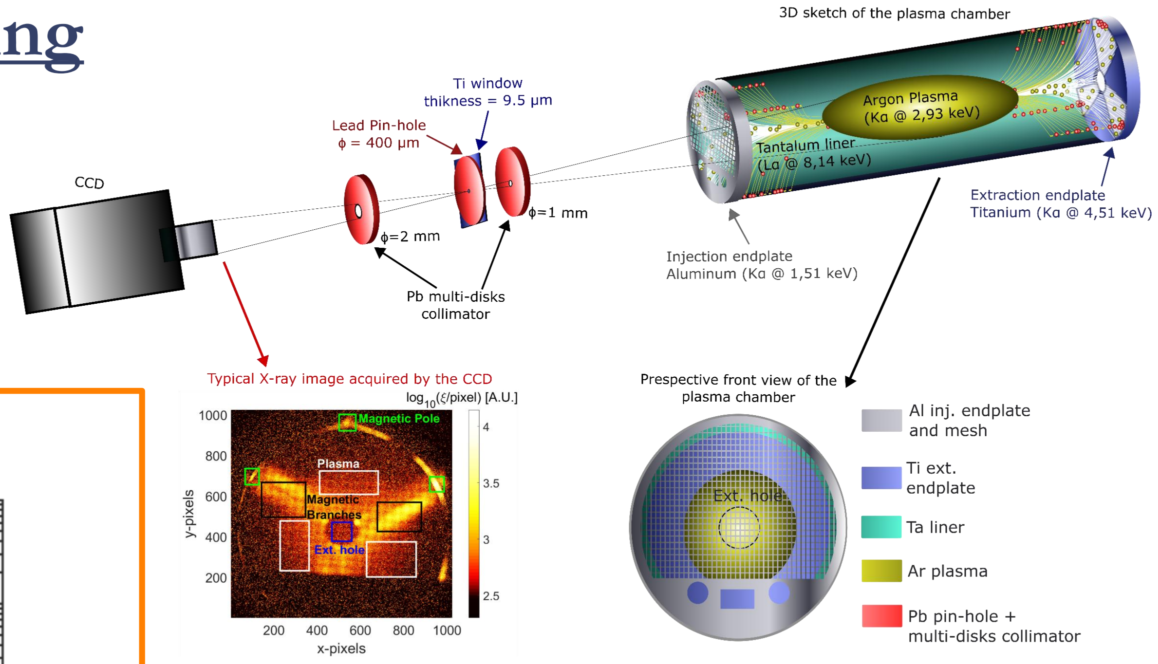
$$J_{theo,brem}(h\nu) = n_e n_i (Z\hbar)^2 \left(\frac{4\alpha}{\sqrt{6m_e}}\right)^3 \sqrt{\frac{\pi}{k_B T_e}} e^{-h\nu/k_B T_e}$$

$$J_{nl\to nl'} = \frac{h\nu_{nl\to nl'}}{\Delta E} n_e n_i \omega_{nl\to nl'} \int_I^\infty \sigma_{nl,ion}(E) v_e(E) f(E) \mathrm{d}E$$

Thermodynamic Parameters

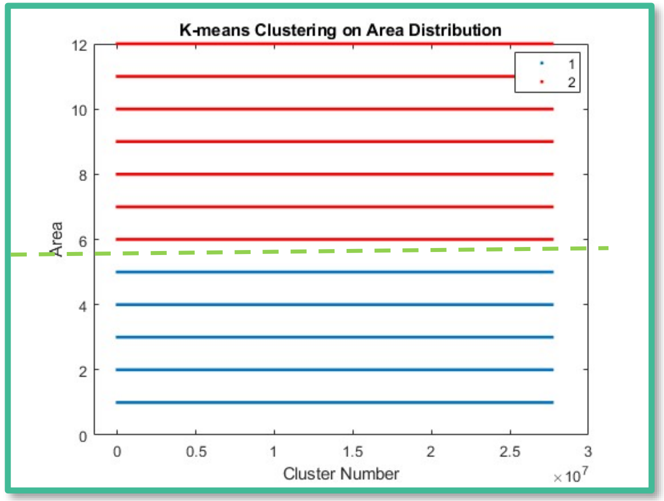
➢ Full-field Energy Spectrum: Gas Mixing

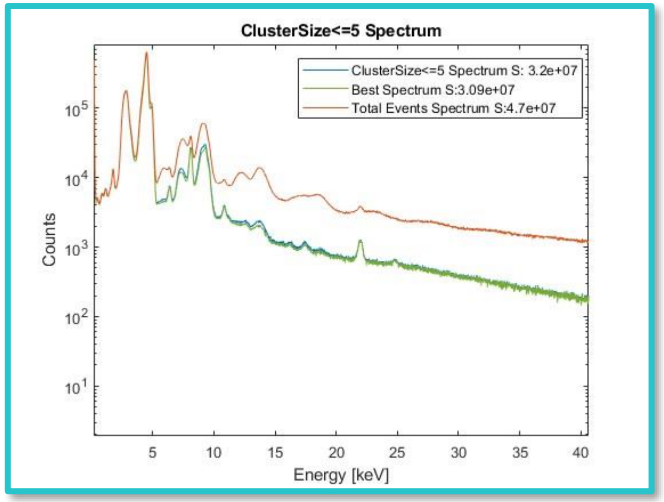




Fluorescence peaks and **Bremsstrahlung Continuum** for Gas Mixing Measurement in the last experimental campaing at **ATOMKI** Lab

➢Validation of the Method

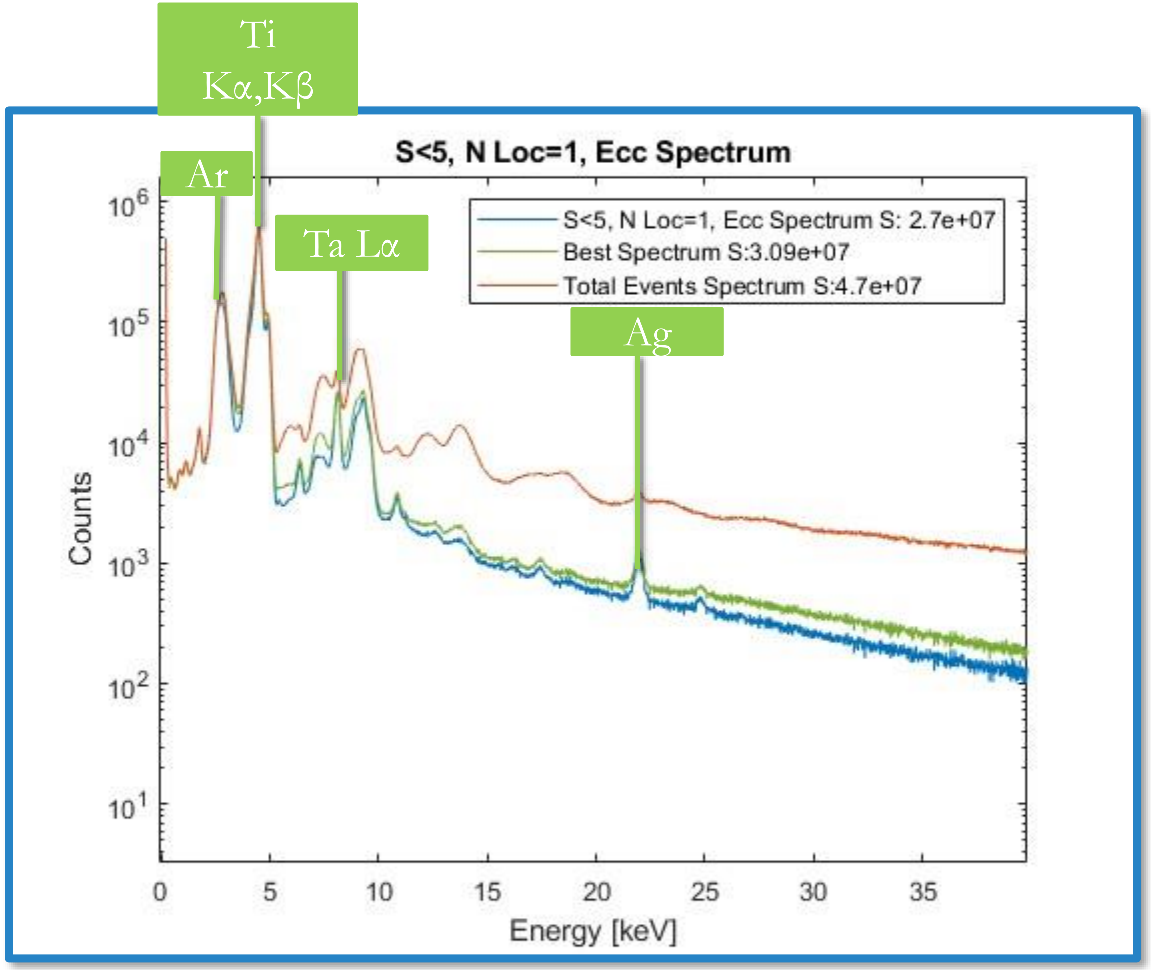




Multiple Features Spectrum:

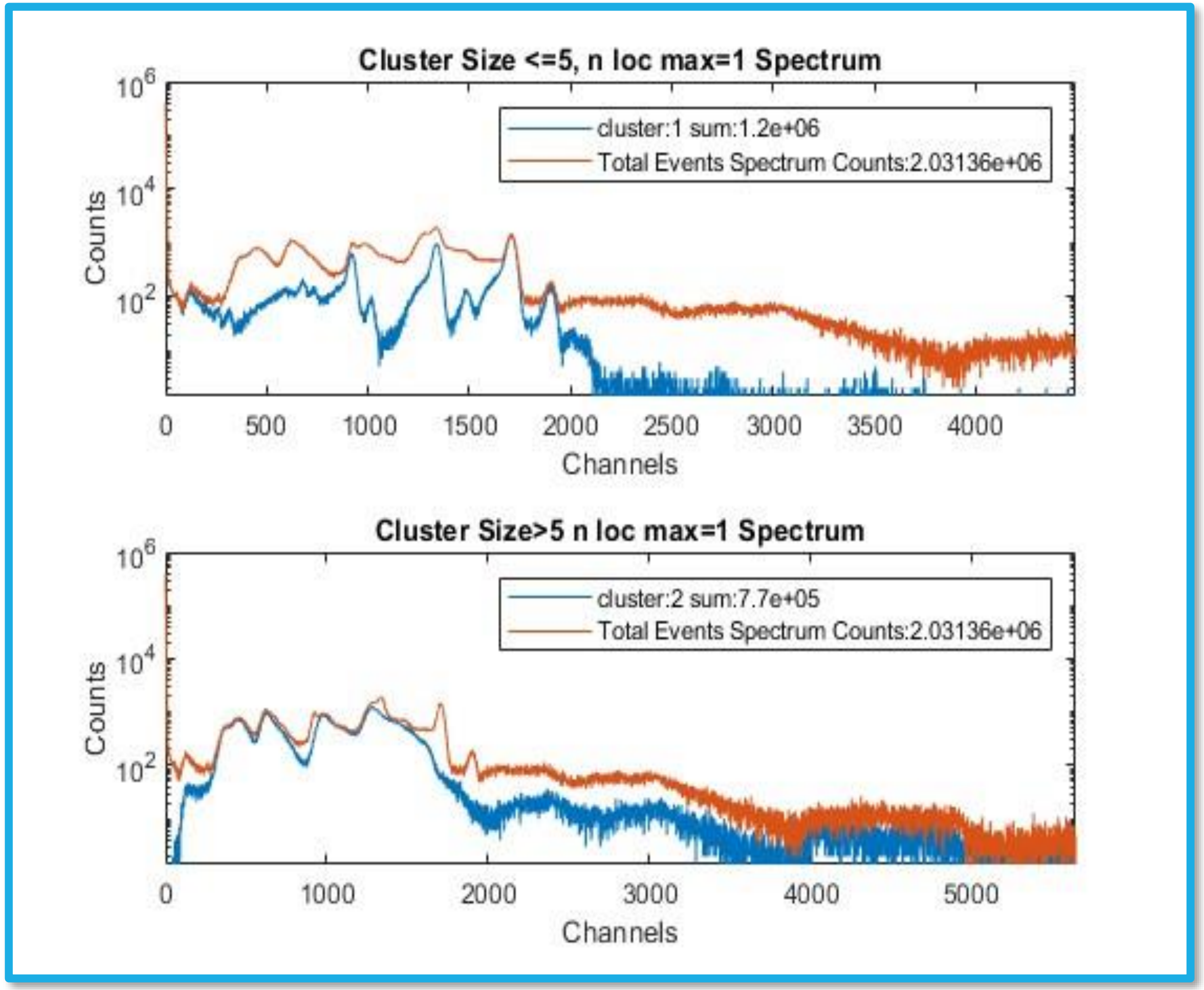
AI Algorithm

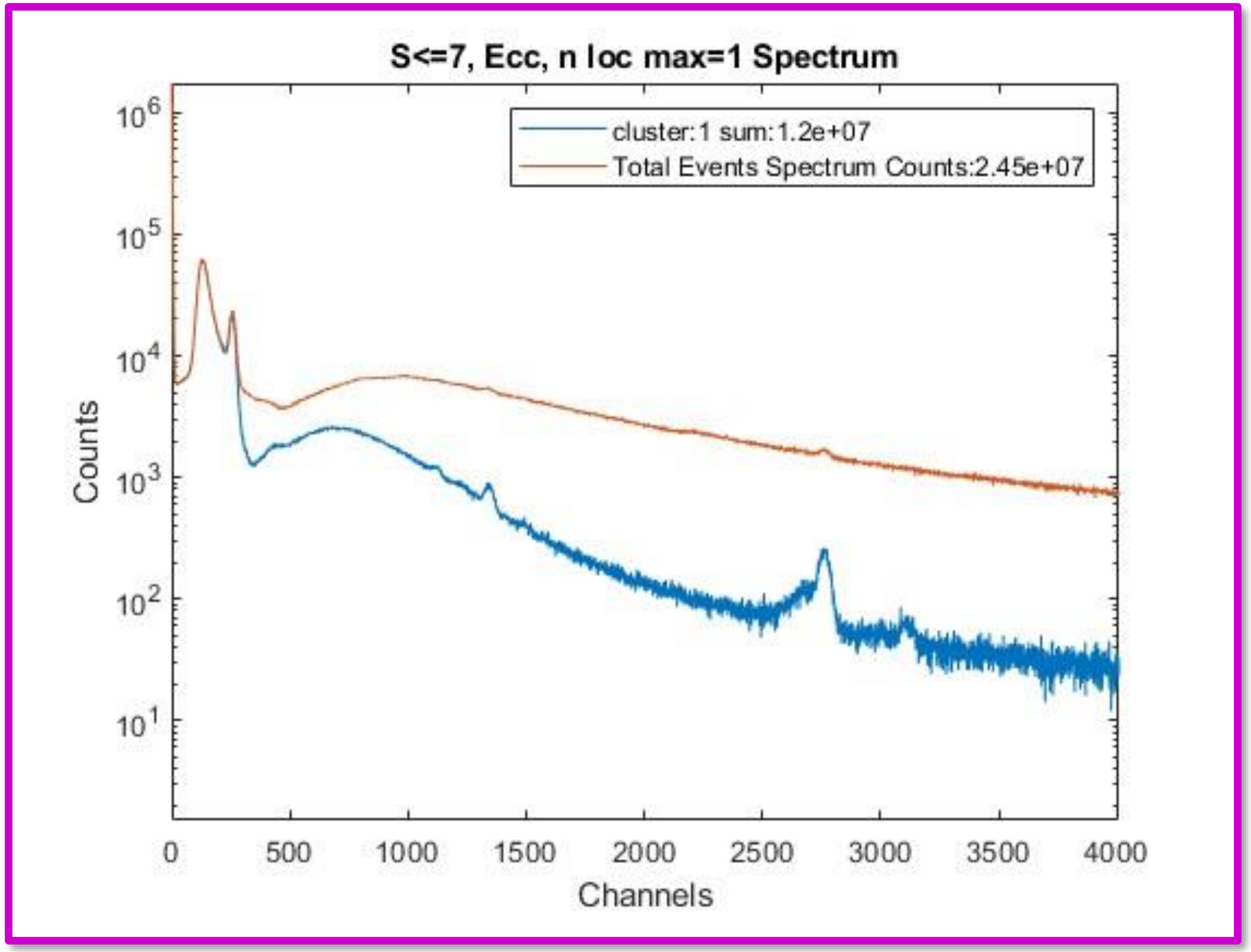
- **Cluster Size**
- **Number of Local Max =1**
- **Eccentricity**



FWHM Ta (keV)	Intensity Ti (Counts)	I_{MP}/I_{DP} Ti	$I_{MP}/_{EP}$ Ta	Counts
0,2535	17377326	10,78818	3,309524	3,1 x10^7
0,2531	15518357	12,05066	3,42512	2,7 x10^7

➢Results of K-means on New Data Set



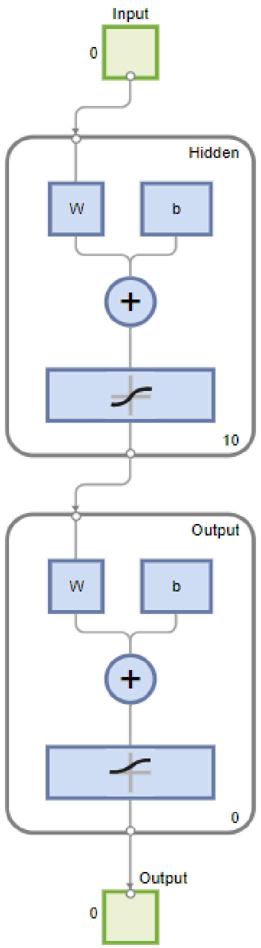


Multielemental Analysis. FPT 2024- INFN-LNS Catania, Italy

Gas mixing measures, Krypton Plasma. Esperimental Campain November 2024- Debrecen, Hungary

➢ Net Development

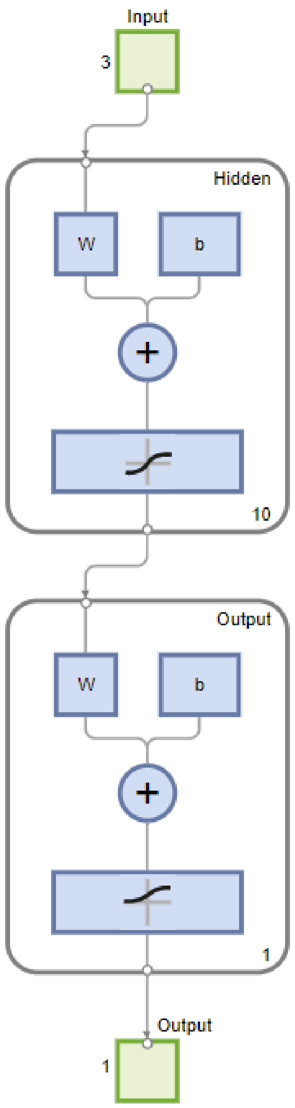
Among the **Feed-Forward Nets** in MATLAB we choose the already optimized net for structured data in classification problems: **PATTERNET**

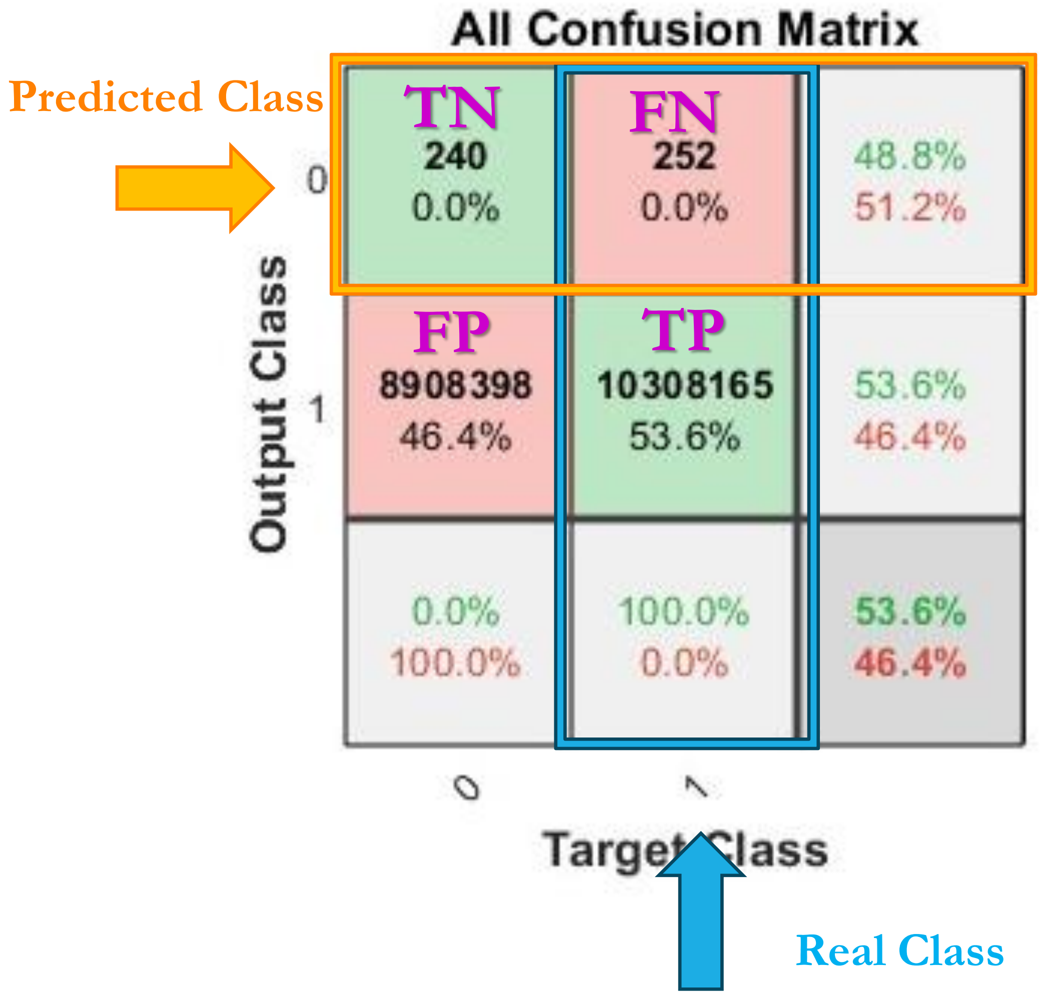


Parameters Definition

- **Neurons in the hidden layer:** we need to esperimentally determine the number trying 5, 10(default), 15 and 20
- **Loss Function used:** it measures the distance between the net prediction and the reality. We want to minimizing it. The **CrossEntropy Function** is the optimized one for binary classification so evaluated: $L = -\sum ylog(p) + (1-y)log(1-p)$
- **Number of Epochs:** it represents the number of time the data are read by the net, also to be test esperimentally (200-500)
- **Minimum Gradient:** to measure the gradient and if no usefull adjustment were made it kills the train (no time waste) 1 e-6
- **Maximum Failure:** maximum number of iteration without change generally 6-10 (avoid overfitting)
- **Training Algorithm: Scale Conjugated Gradient (SCG)** is optimized for fast convergence, empolying an adattive step learning rate.

➤1st Attempt to Train





Let's explore the confusion Matrix:

We have the Predicted Class in rows, and the Real Class in the coloumns. So we can easily highlight the

True Positive(True predicted Correclty),

False Positive(True predicted Wrong),

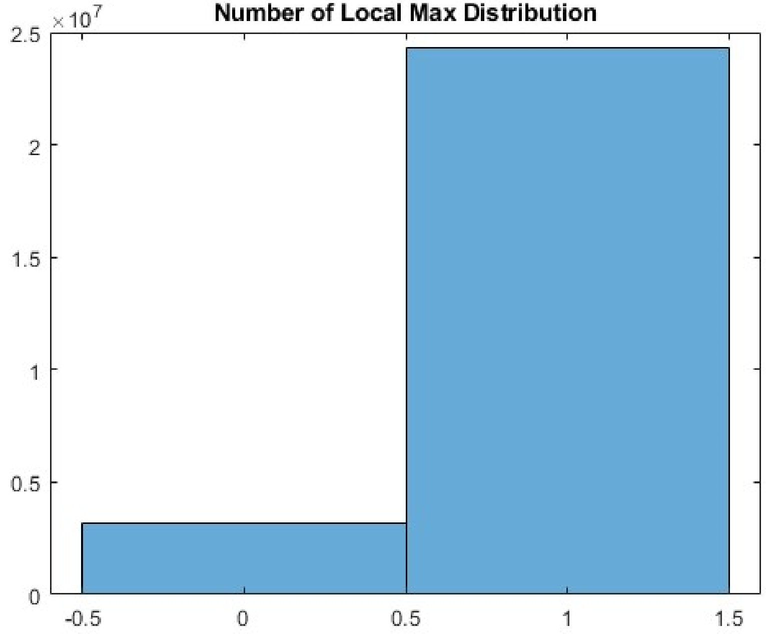
True Negative(False predicted Correctly) and

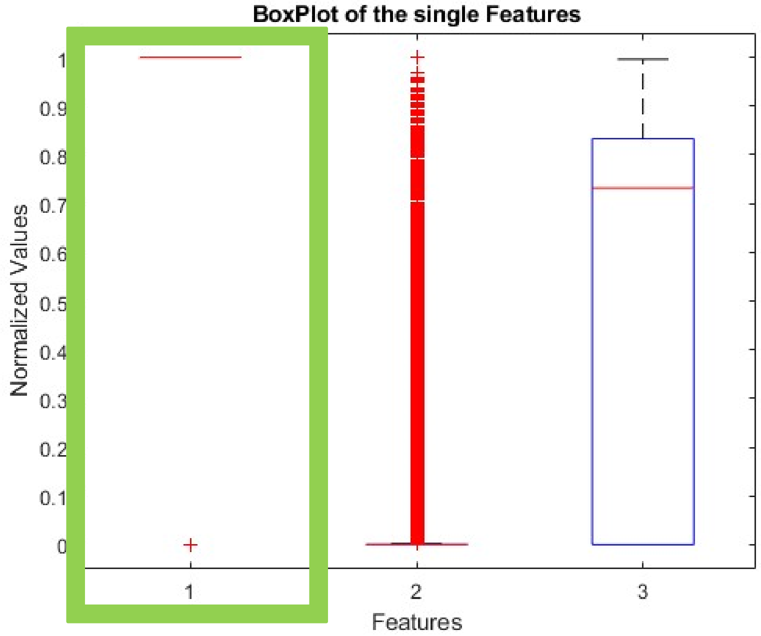
False Negative(False predicted Wrong) class.

➢ Data Check

Dataset1

Number Of Local Max	Clustersize	Eccentricity
…	…	…
…	…	…





In this case we're working with a binary feature (box plot not so informative): as we can see there's a significant imbalance. If the neural net has an hard time recognizing the zero pattern, the dataset can be balanced by different techniques:

- **Oversampling**, simply repeating the events multiple time to match the order of magnitude of the 1 class
- **Class Weighting**, setting different weights on the two classes, without modifing the dataset
- **A combination of the two techniques**

Of course we need to compare the performance of the neural net with and without the balancing processes.

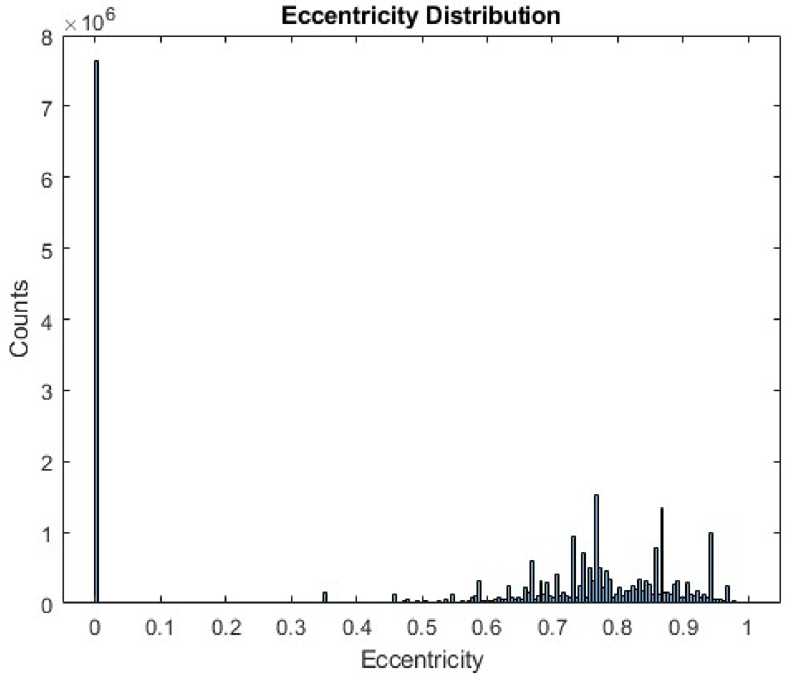
Strong correlation with the output!

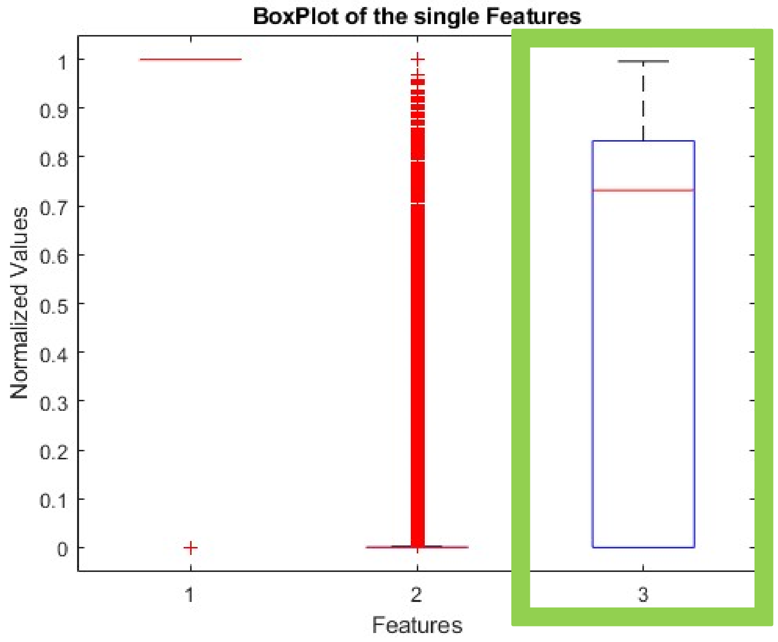
X^2 ≈ 98 e p≈ 0

➢ Data Check

Dataset1

Number Of Local Max	Clustersize	Eccentricity
…	…	…
…	…	…





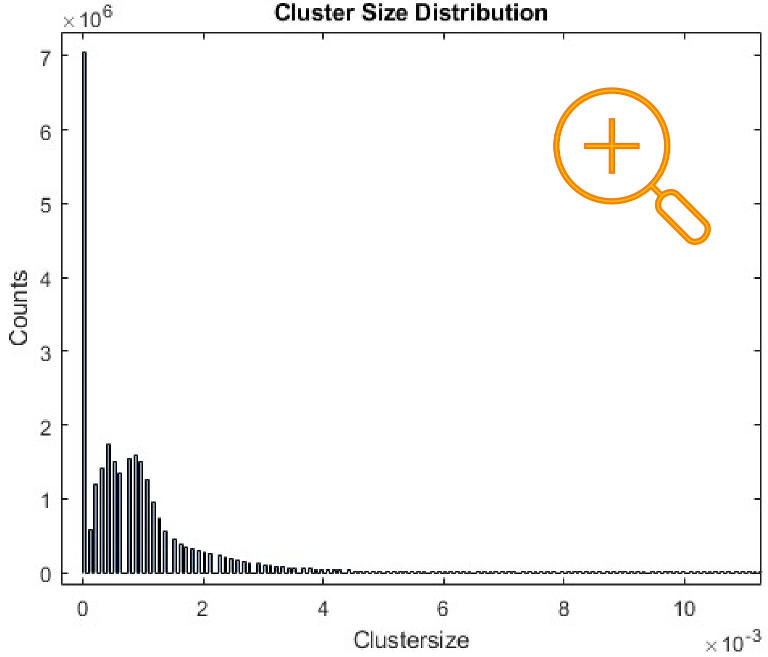
This feature is indeed **better distributed**, in fact it shows a wide box, with the median closer to the higher value, meaning more than half of the data is distributed on higher value.

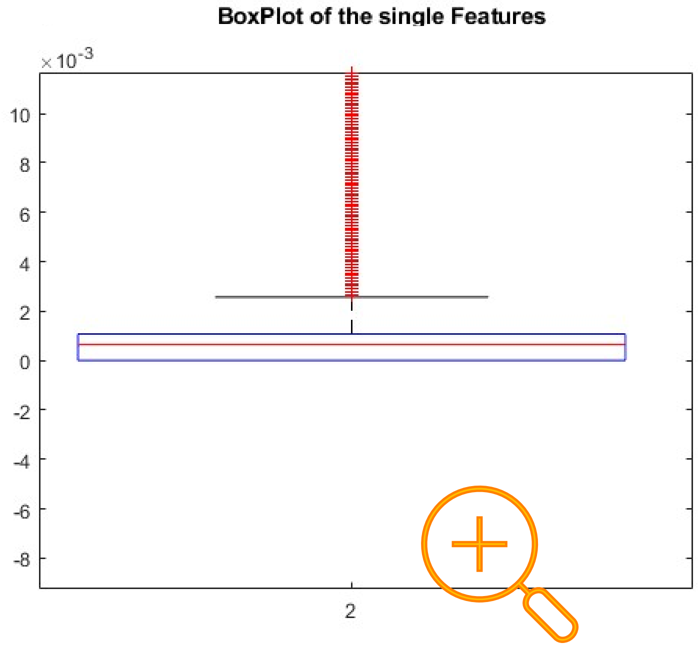
More over it's not necessary to balance it, and it also show a **smaller correlation** to the output in respect to the other two features, as expected.

Data Check

Dataset1

Number Of Local Max	Clustersize	Eccentricity
…	…	…
…	…	…



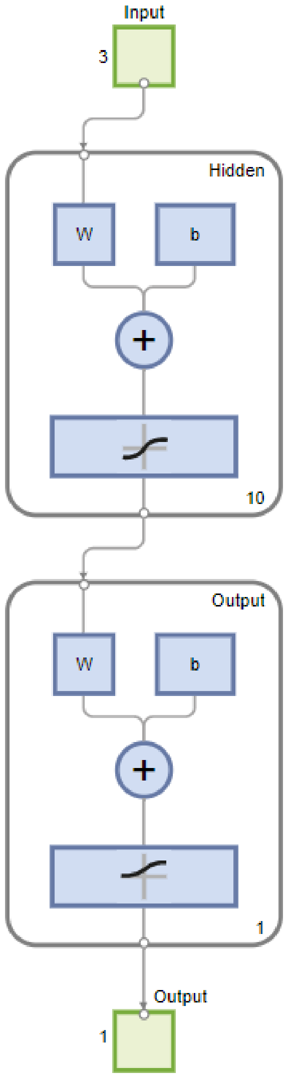


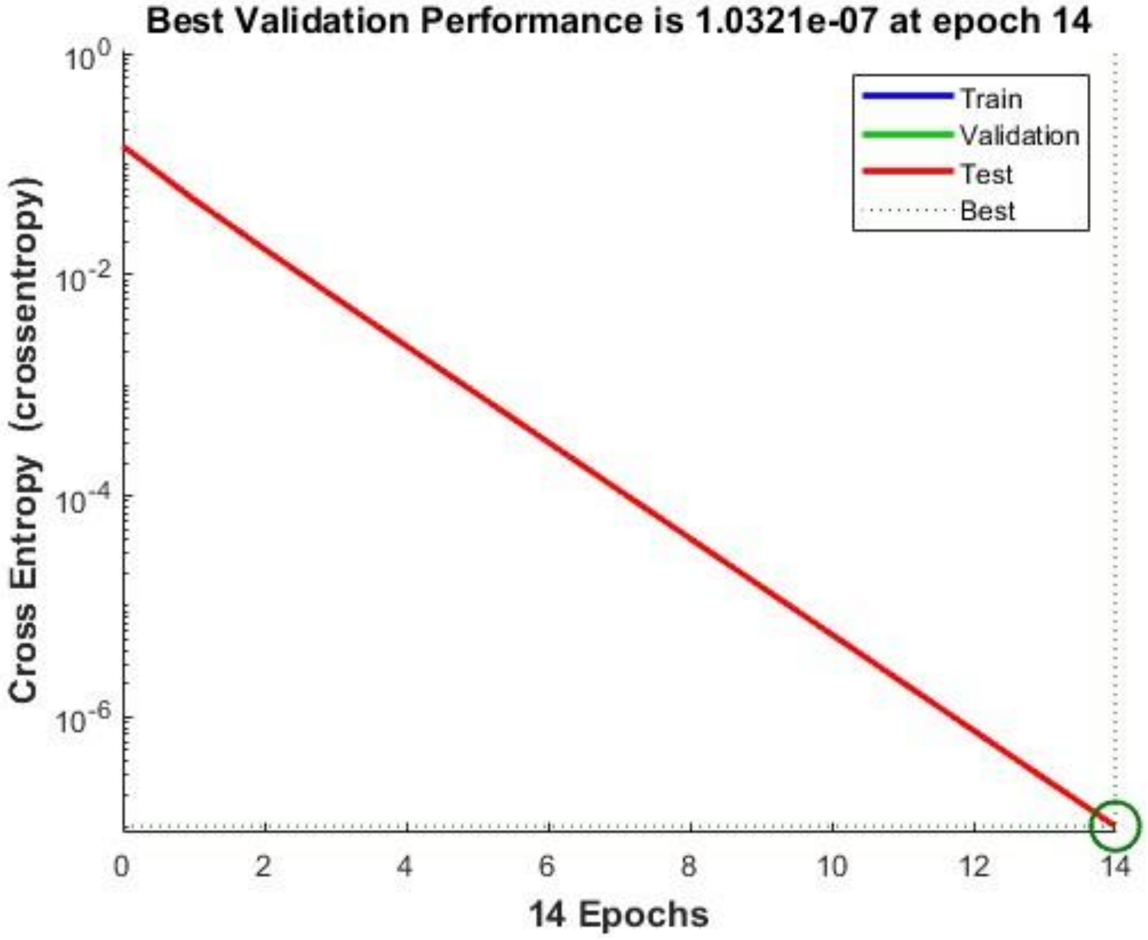
The box is really narrow and it's easy to tell that the distribution is peaked on small value. It shows a great number of higher value outliers. This means that the distribution is <u>**not balanced at all**</u>, but its correlation to the output is <u>**significant**</u>, we do have to modify it:

- <u>**Cut off the outliers**</u> brutally and make a new normalization
- Make a <u>**log-transform**</u> in order to make the model more sensible in the lower part of the distribution
- Make it into a <u>**categorical feature**</u> with a new binning

➢Validation Check 1: Number of Local Max

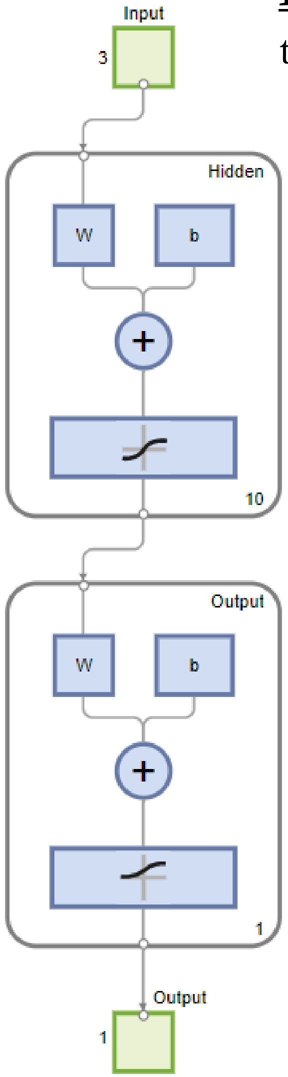
As expected the Confusion Matrix shows that we have a **100% accuracy**, so all the prediction were correct, furthermore the **Loss Function** was minimize sharply, with a strong coherence in behaviour between train, validation and test. Lastly the training stopped due to the gradient stop condition.

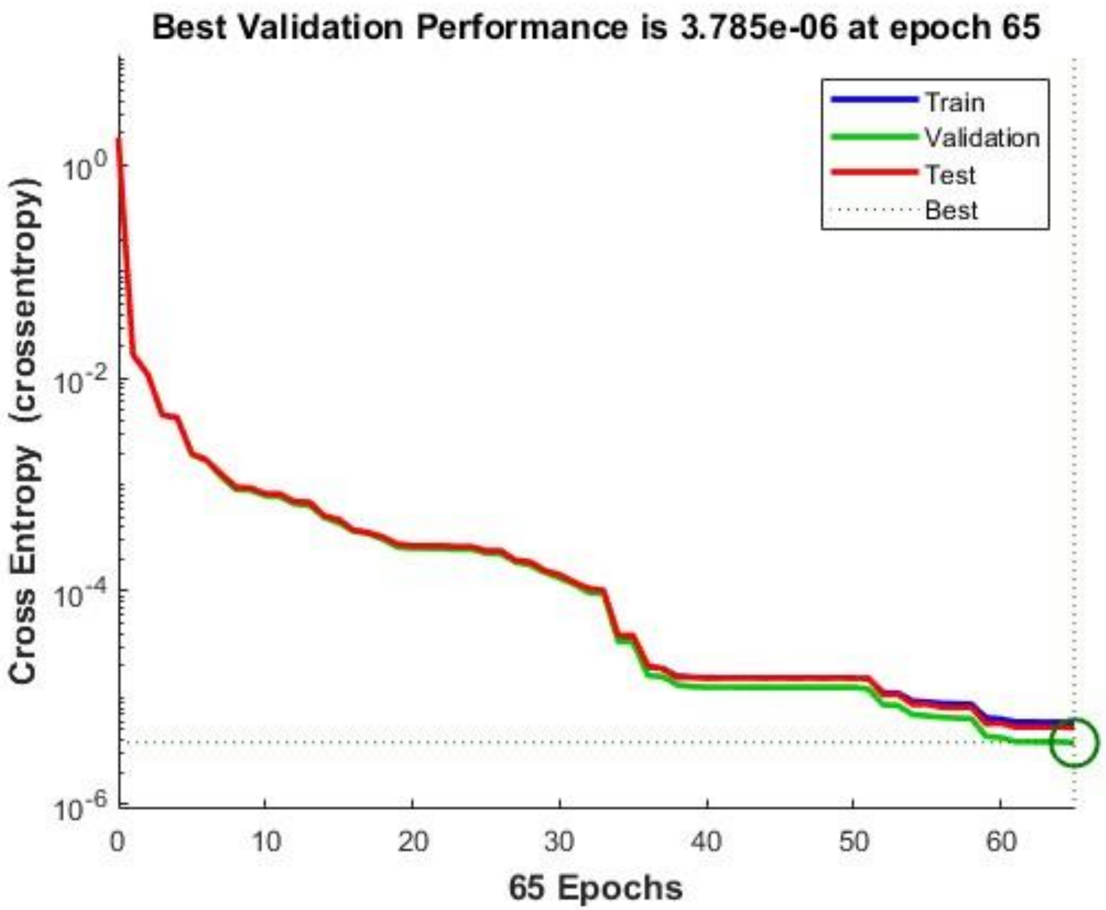




➢Validation Check 2: Eccentricity

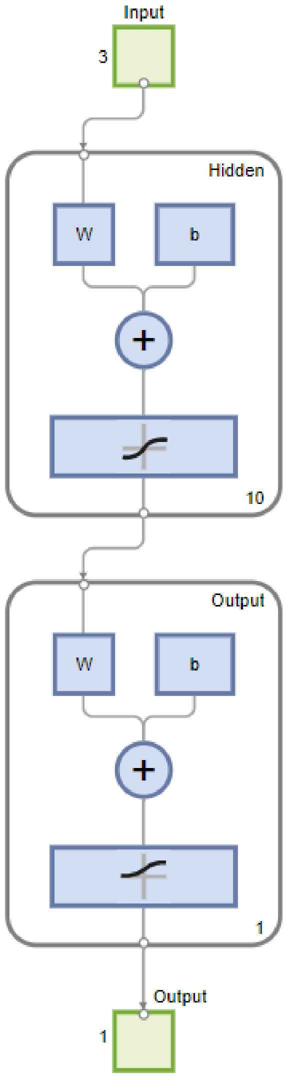
The Confusion Matrix shows that we have a **100% accuracy**, so all the prediction were correct, furthermore the **Loss Function** was minimize, with a strong coherence in behaviour between train, validation and test. Lastly the training stopped due to the gradient stop condition.

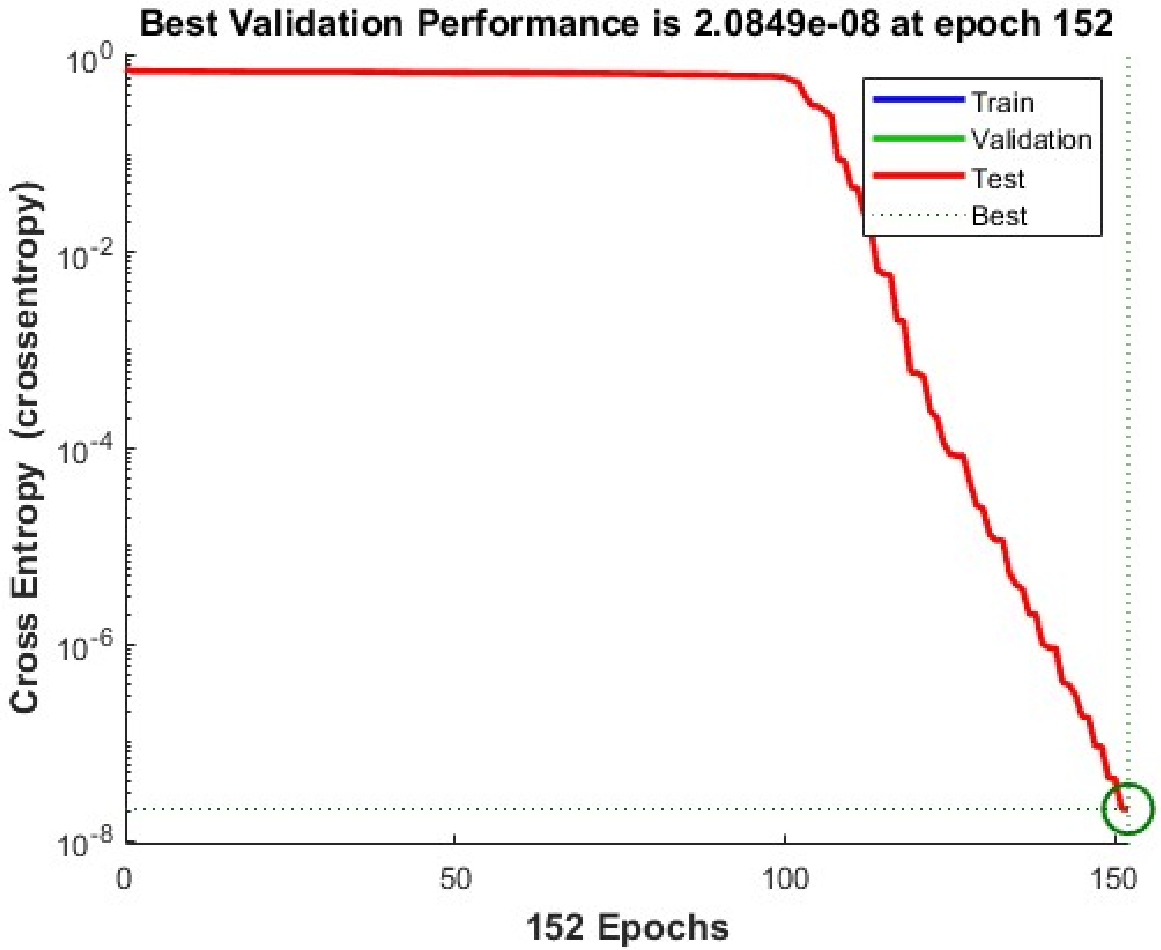




➢Validation Check 2.1: Cluster Size

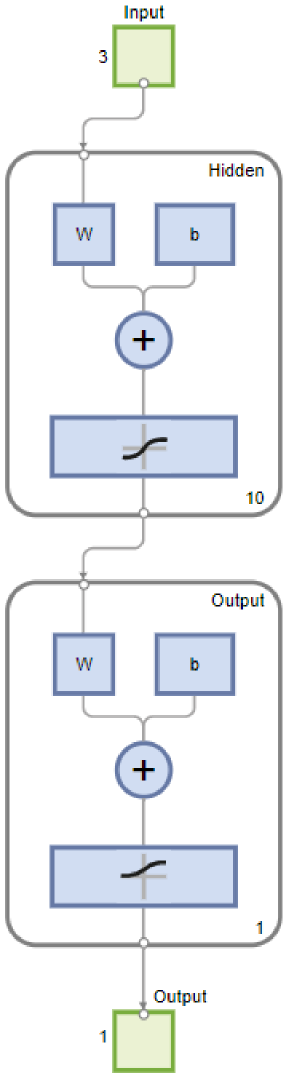
Normalized dataset

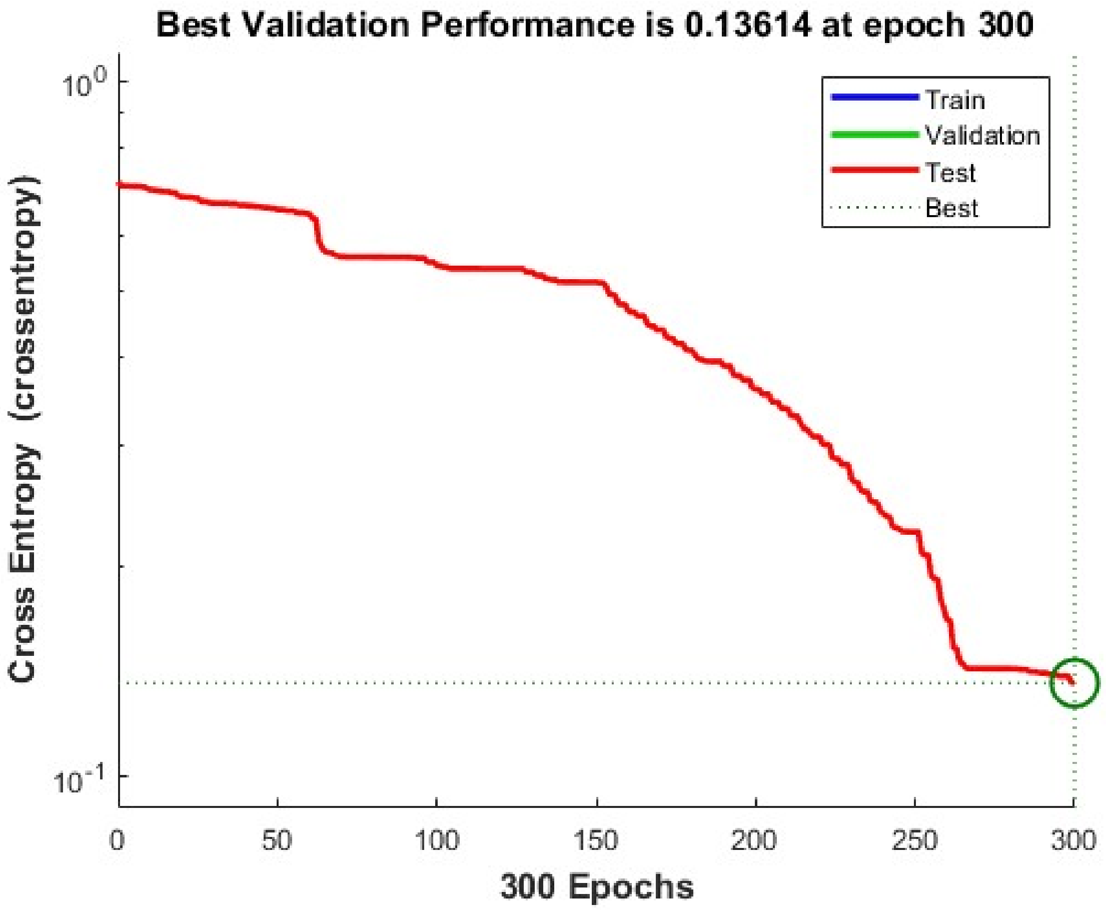




➢Validation Check 2.2 : Cluster Size

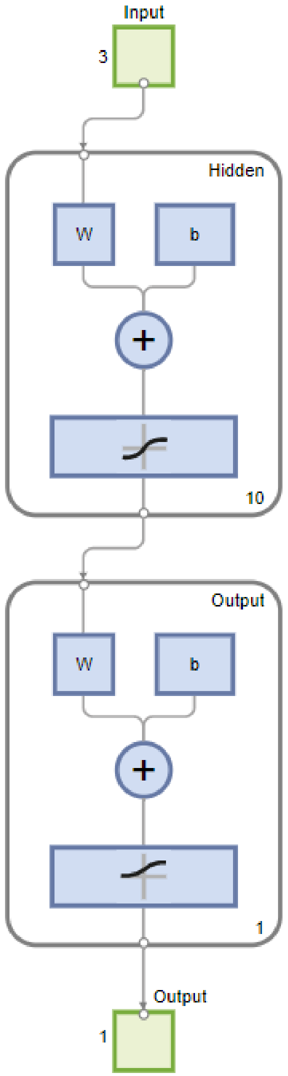
Non-normalized dataset

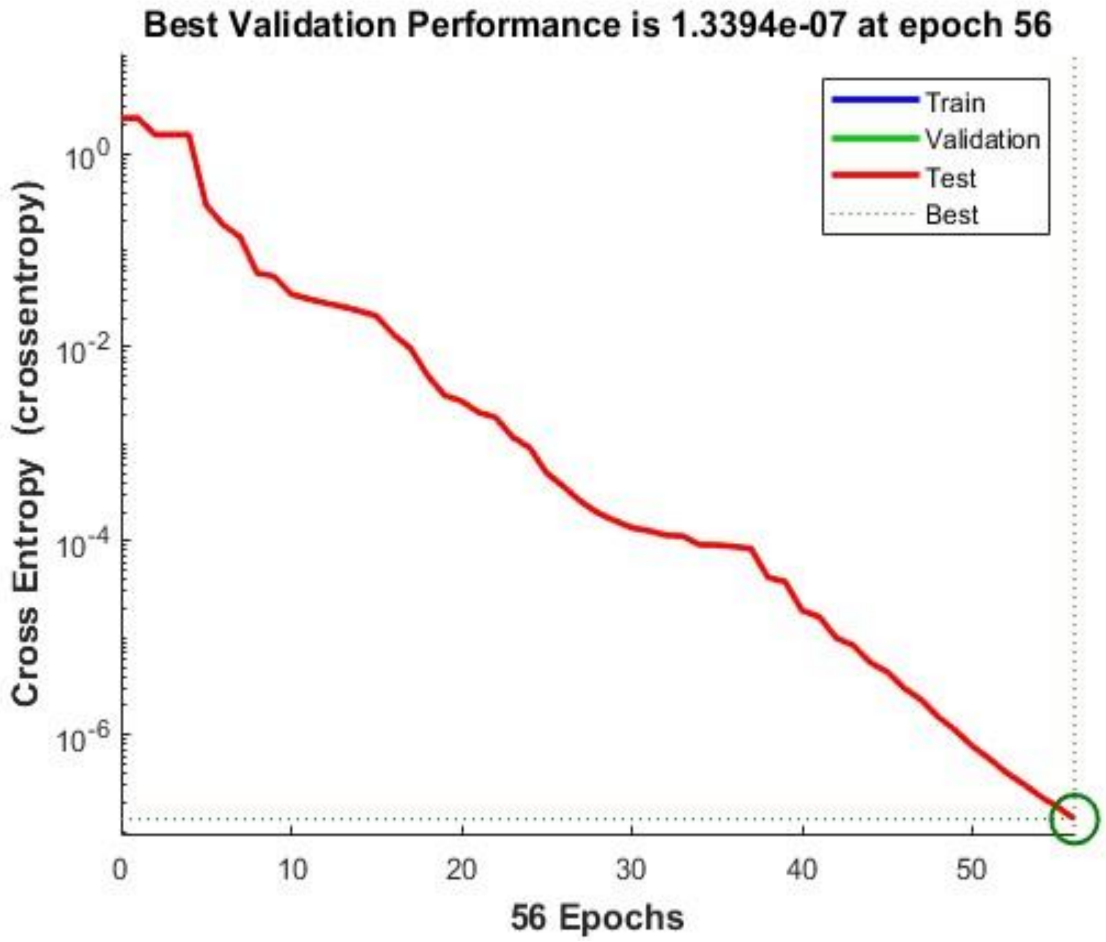




➢Validation Check 2.3.1 : Cluster Size

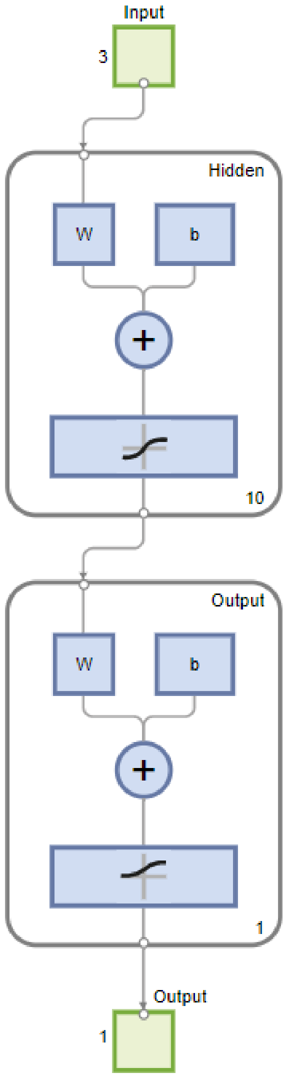
Normalized Cut dataset (@50)

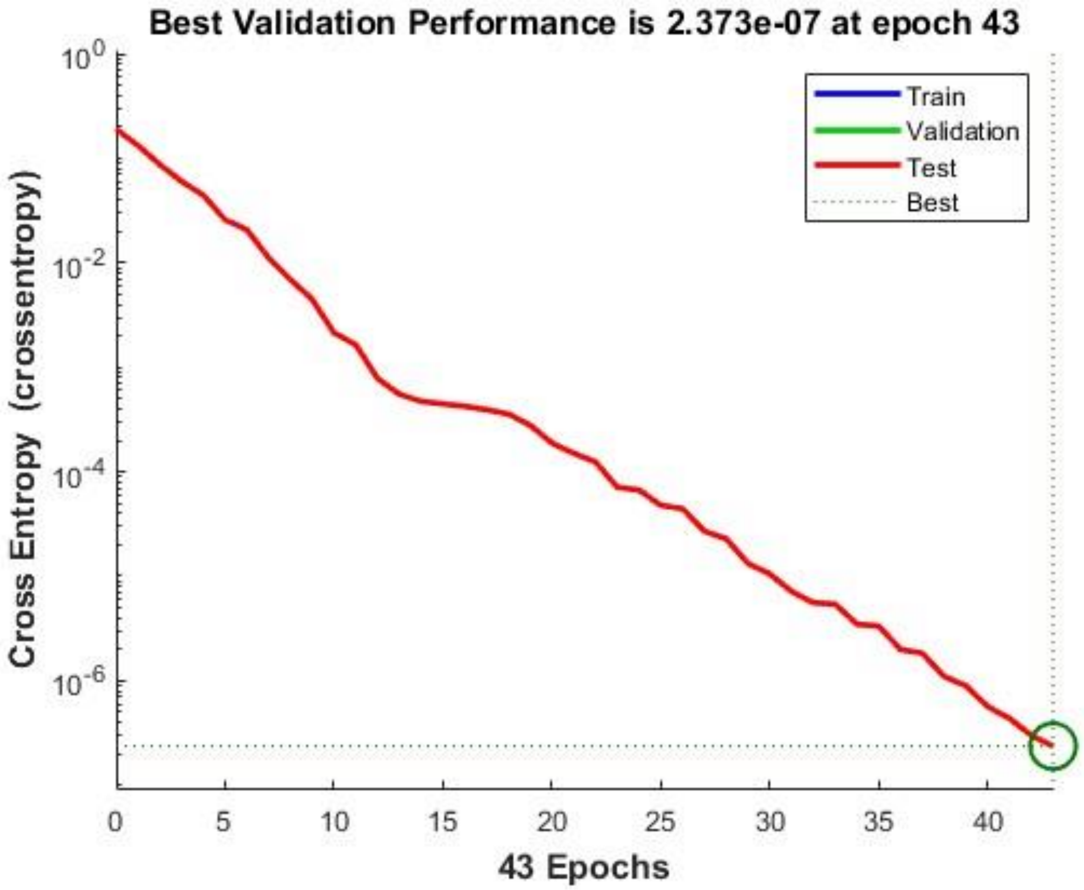




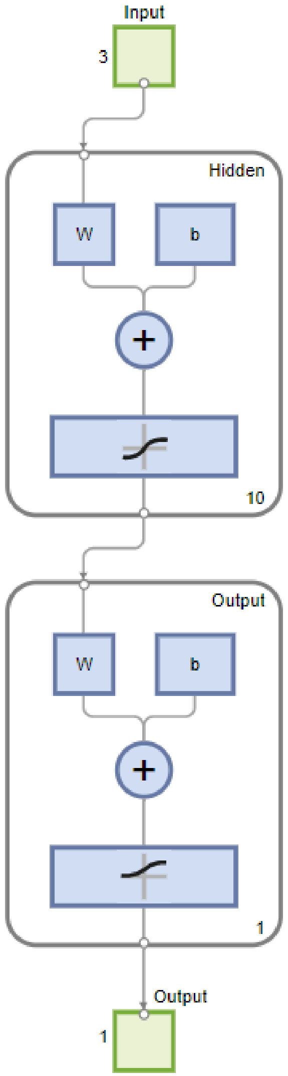
➢Validation Check 2.3.2 : Cluster Size

Normalized Cut dataset (@30)

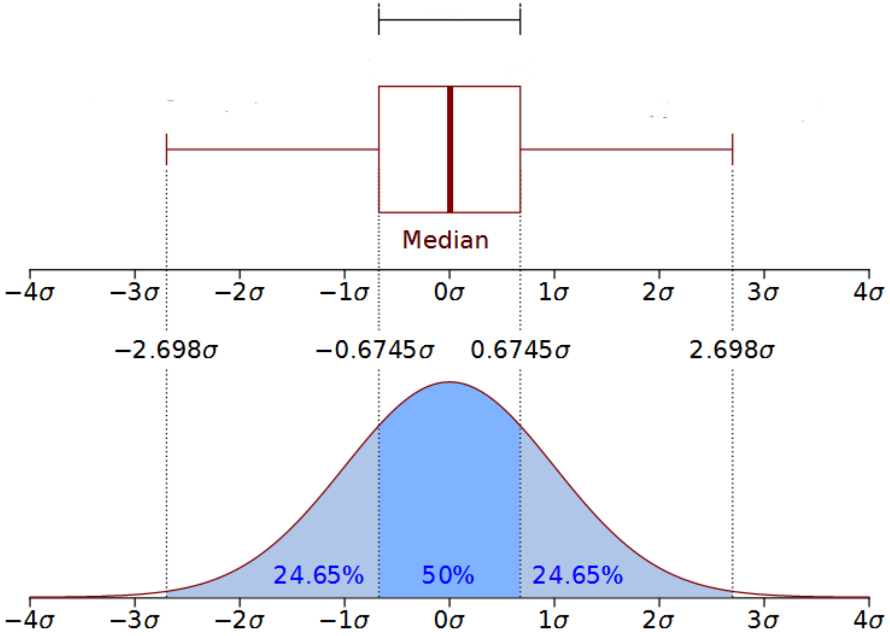


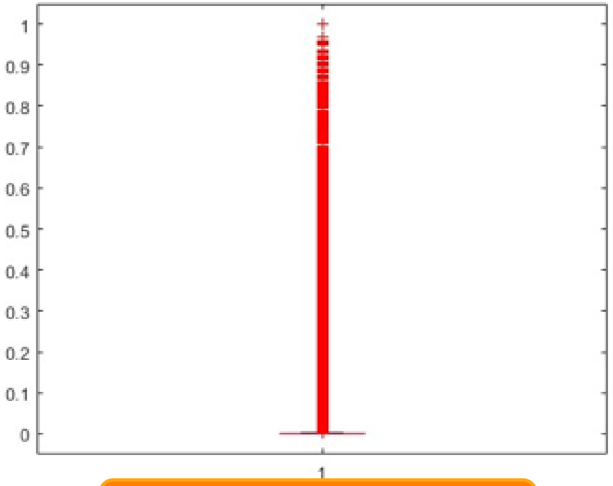


➢Validation Check 2.3 : Cluster Size

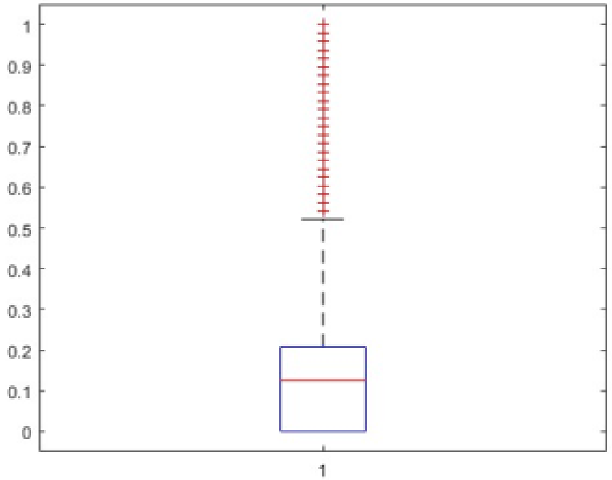


Let's compare different BoxPlot graph, obtained with the whole dataset, the cutted one at 50 and at 30.

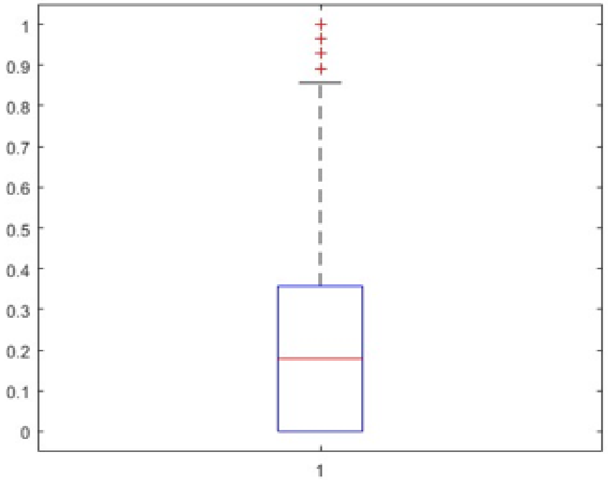




Whole Dataset



Cut at 50 Dataset



Cut at 30 Dataset