Belle II

JUSTUS-LIEBIG-UNIVERSITÄT GIESSEN

# BELLE II PIXELDETECTOR CLUSTER ANALYSIS USING NEURAL NETWORK ALGORITHMS

Katharina Dort on behalf of the Belle II group at the University of Giessen

(katharina.dort@cern.ch)

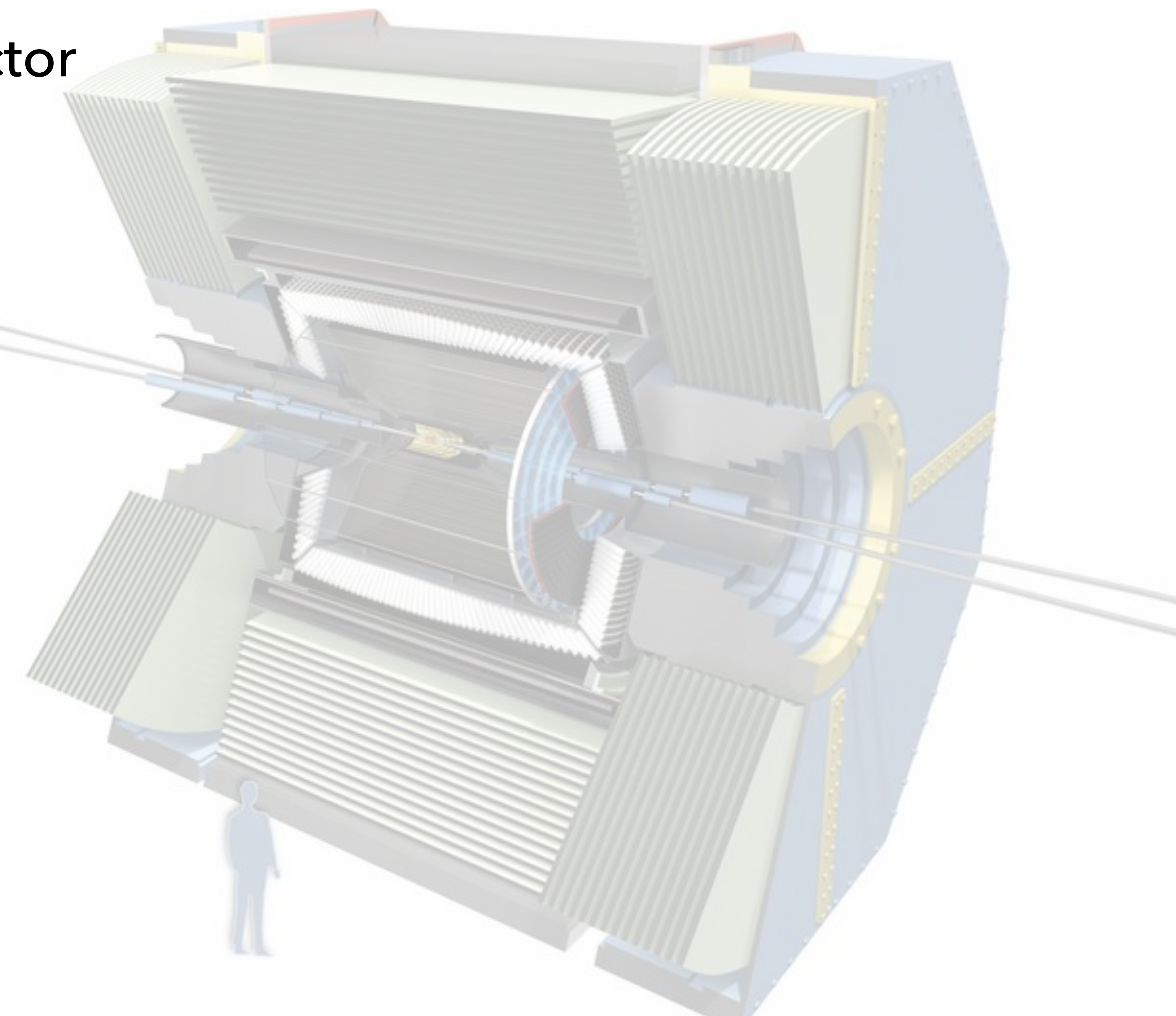
**Vienna 13/01/20**

SPONSORED BY THE
Federal Ministry of Education and Research

JENNIFER2
EU grant n.822070

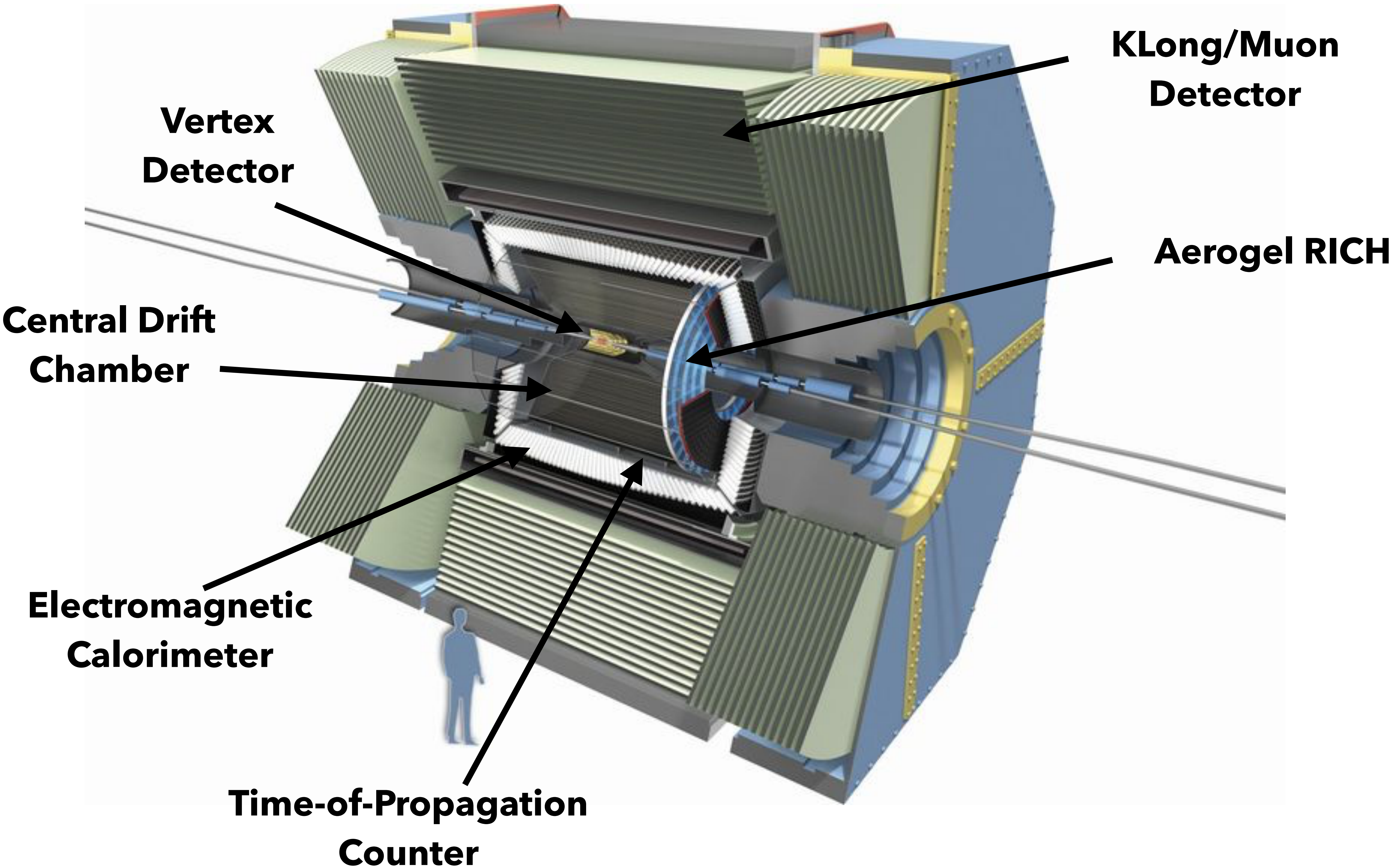
# OUTLINE

- Motivation
- Recap: The Belle II Pixel Detector
- Highly Ionizing Particles
- Analysis Strategy
    - Offline Analysis
    - Online Analysis
- Summary and Outlook

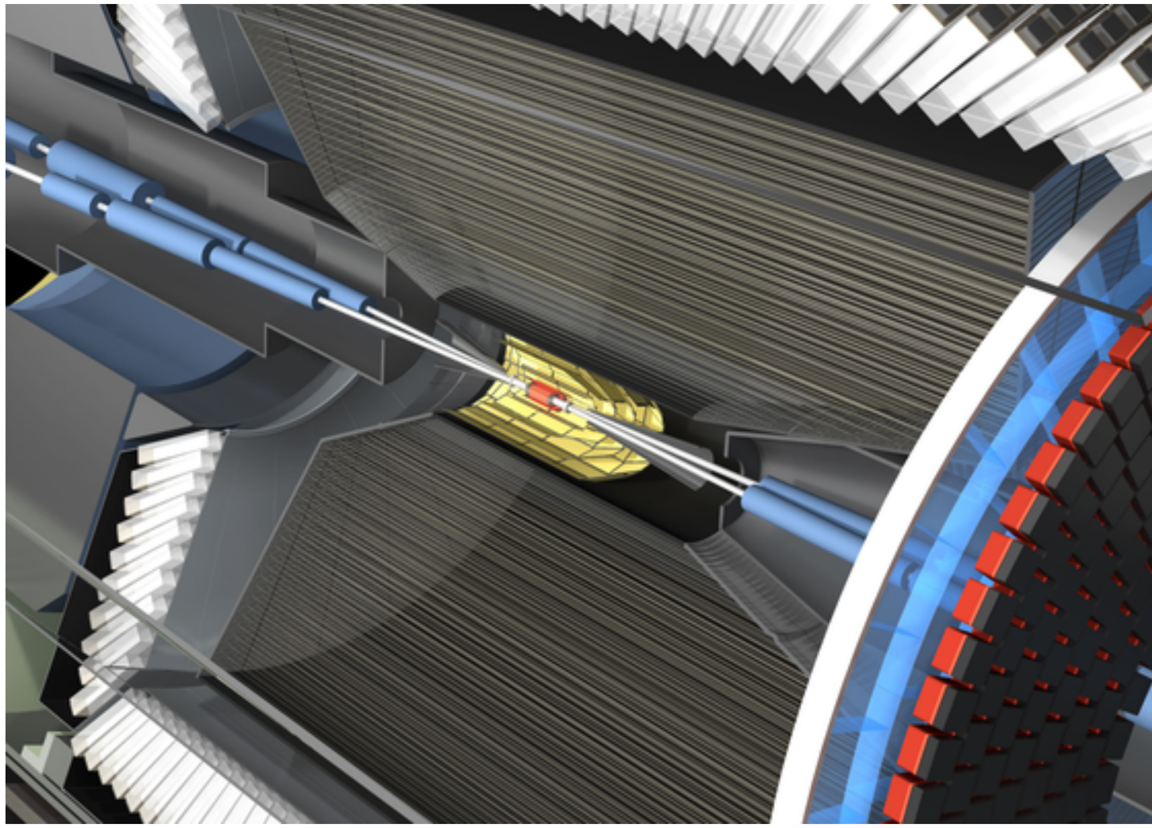


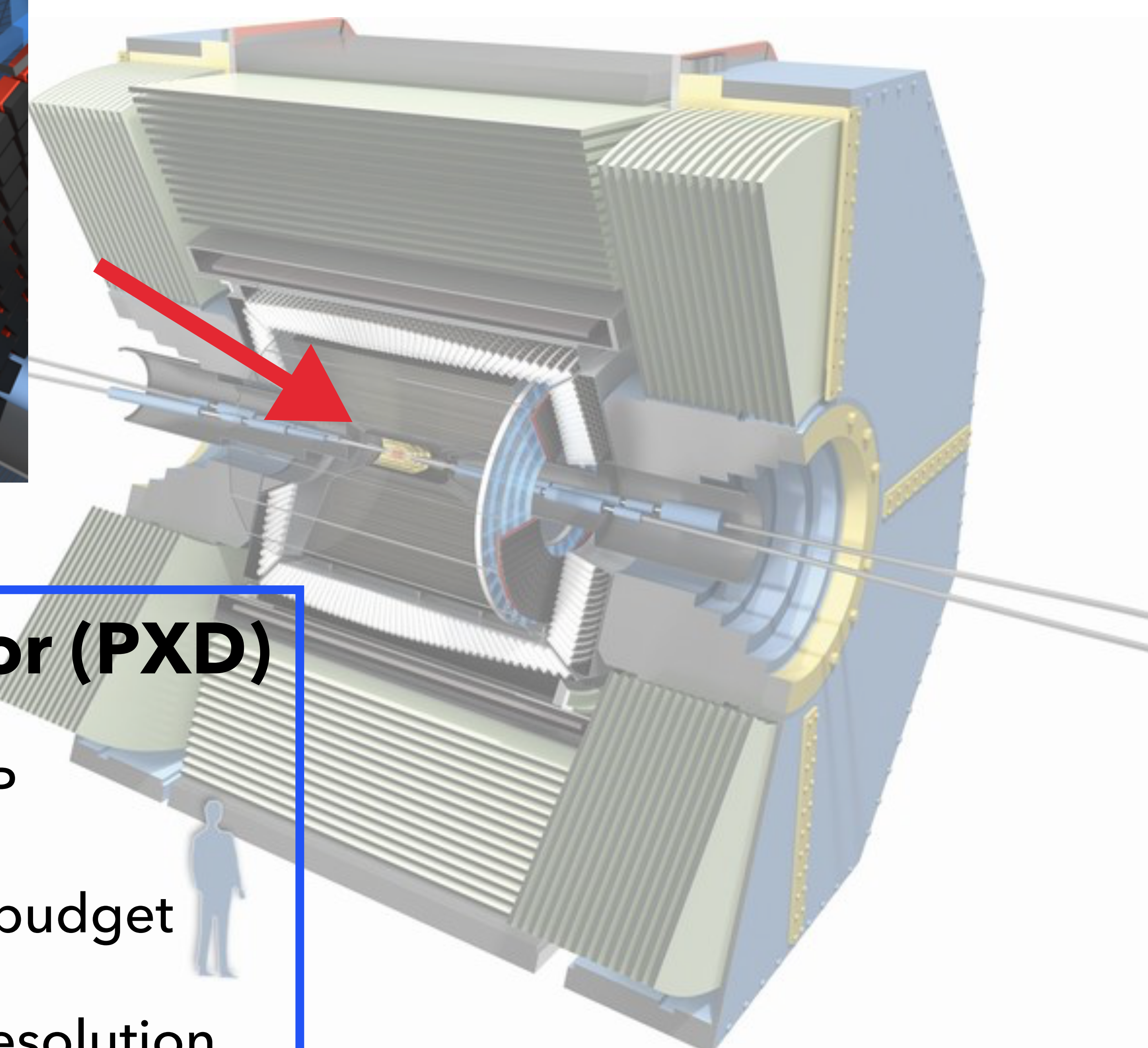
# THE BELLE II PIXEL DETECTOR

# BELLE II DETECTOR



# PIXEL DETECTOR





## 2 layer* DEPFET Pixel Detector (PXD)

- R = 1.4 cm / 2.2 cm
- Thickness: 75 μm
- Pixel size: 50 μm - 85 μm

√ Close to the IP

√ Low material budget

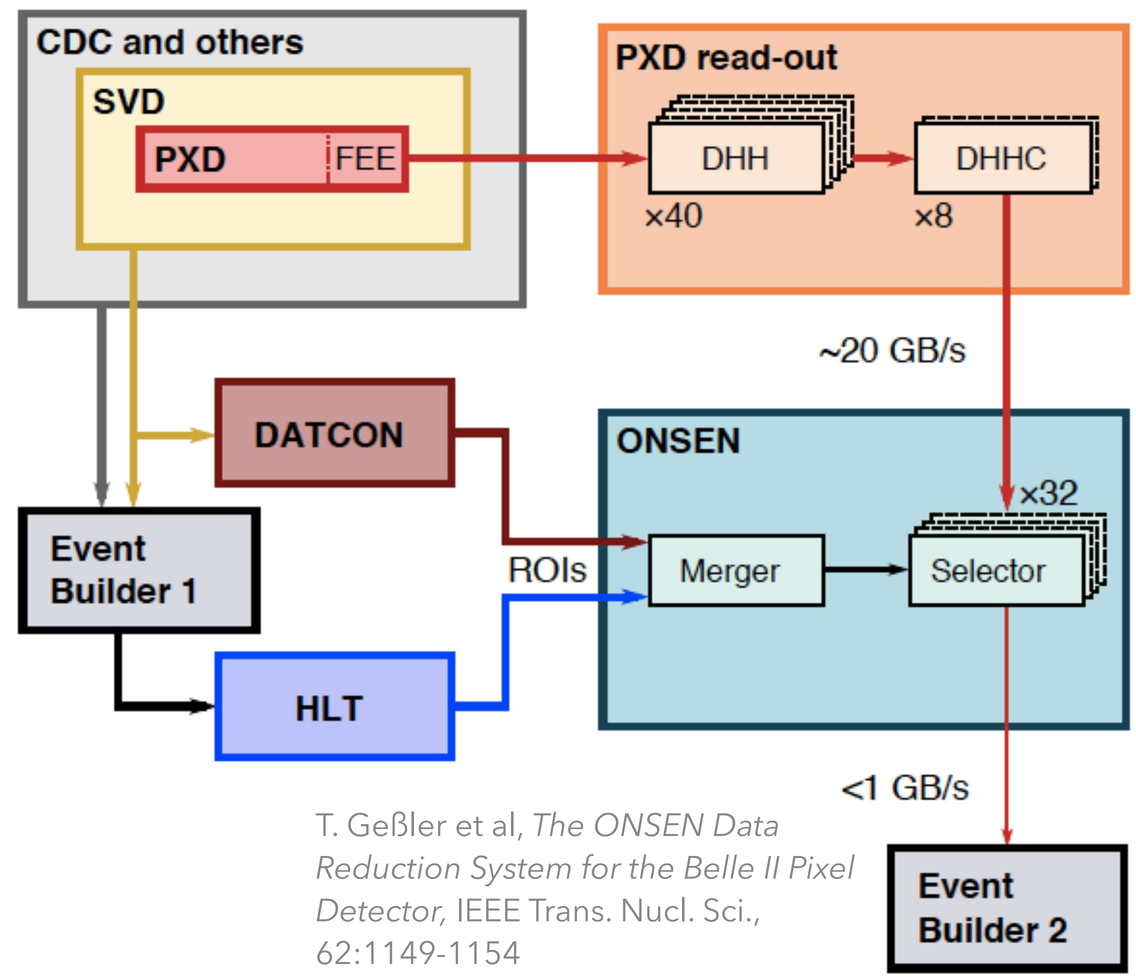
√ High spatial resolution

*Only one layer installed so far

# PIXEL DETECTOR READ-OUT

- Data rate from PXD is drastically higher than rate from all other sub-detectors combined

➡ Online data reduction is required




T. Geßler et al, *The ONSEN Data Reduction System for the Belle II Pixel Detector*, IEEE Trans. Nucl. Sci., 62:1149-1154

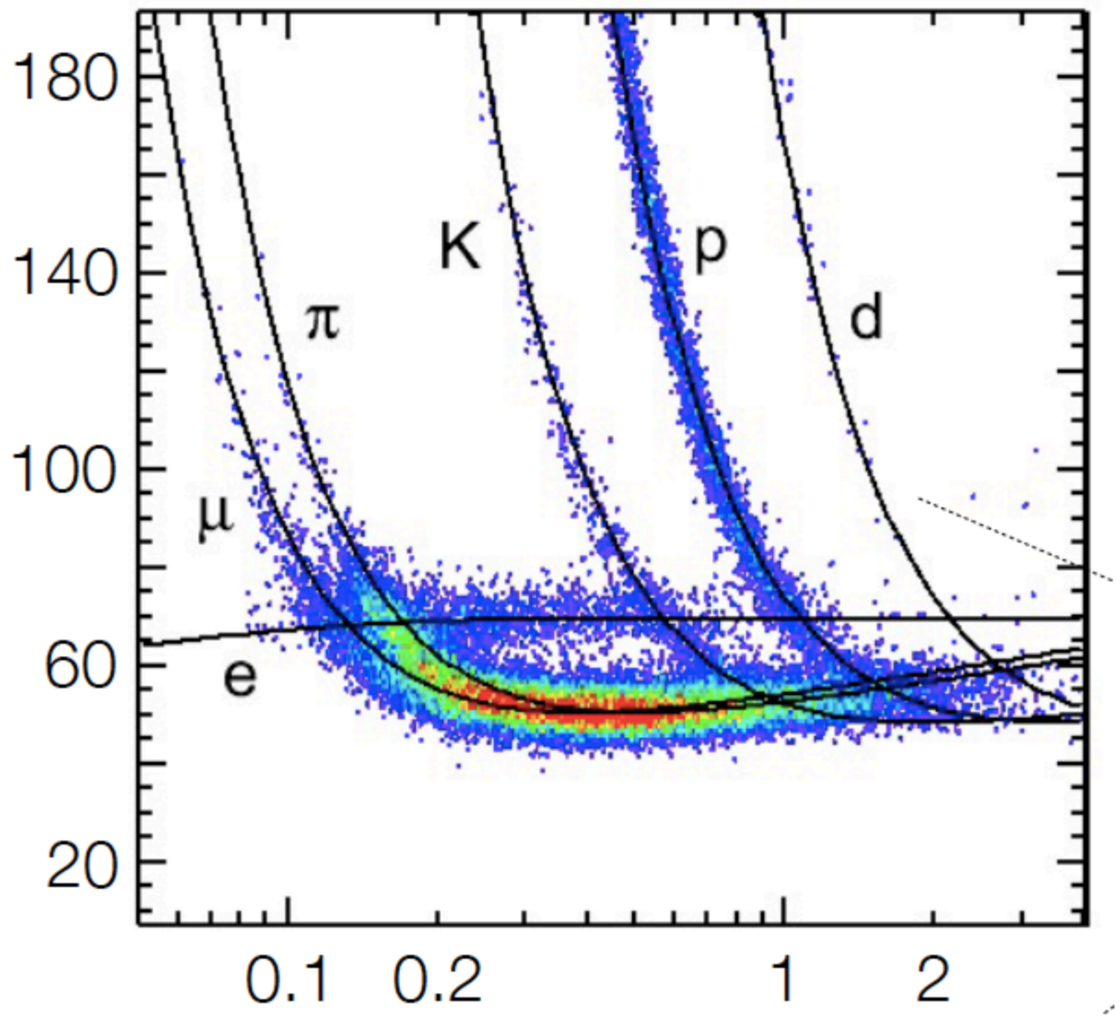
- Challenge: only particles leaving a reconstructable track in the outer tracking detectors are detectable

- Solution: a **cluster rescue system** deployed on the ONSEN would mitigate the loss of particles without a reconstructable track

# HIGHLY IONIZING PARTICLES

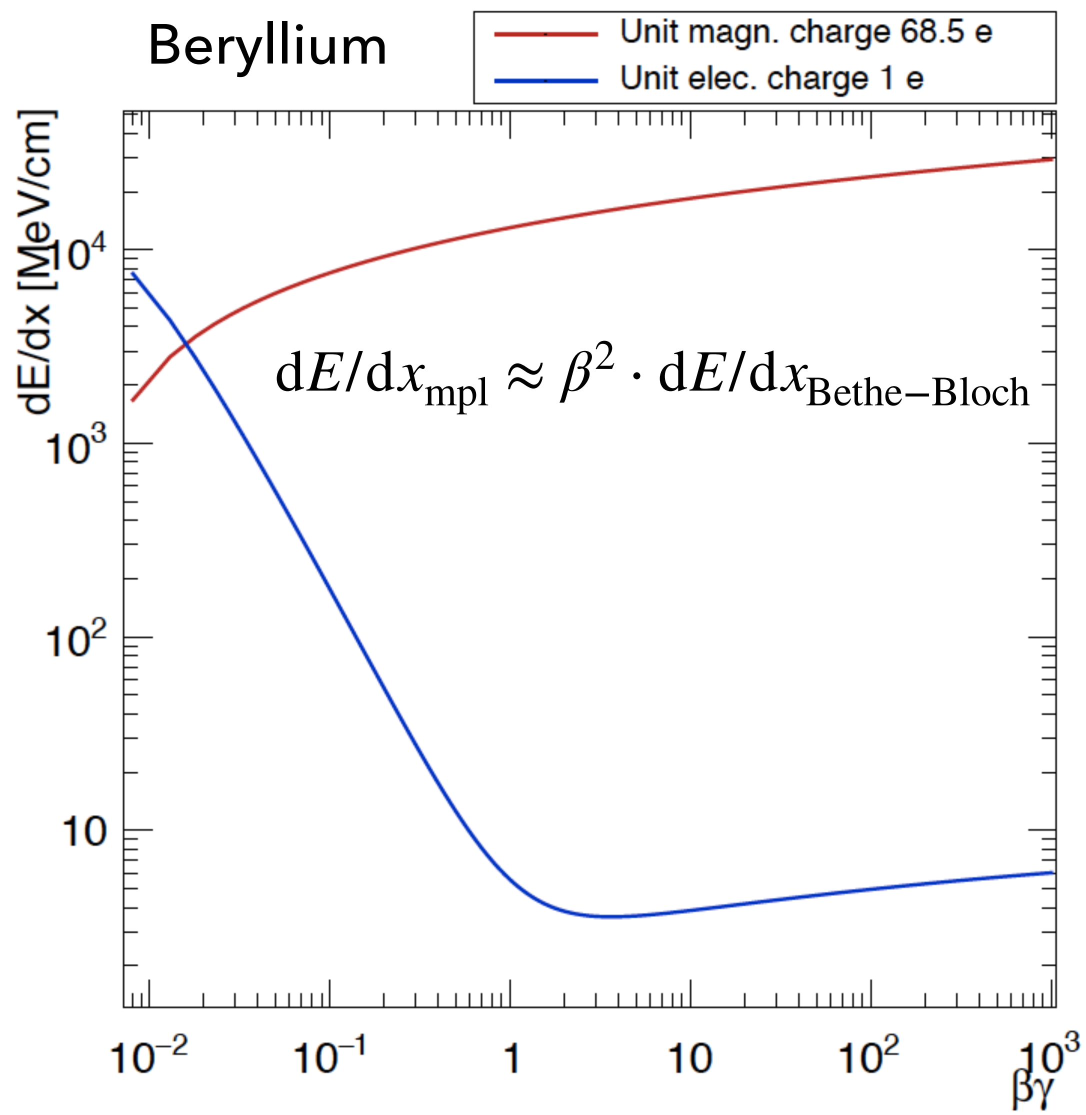
# WHAT ARE HIGHLY IONIZING PARTICLES?

- Energy deposition of particles in matter increases with
  - decreasing momentum of projectile
    **Example:** slow pions
  - increasing mass of projectile
    **Example:** deuterons, heavy exotic particles



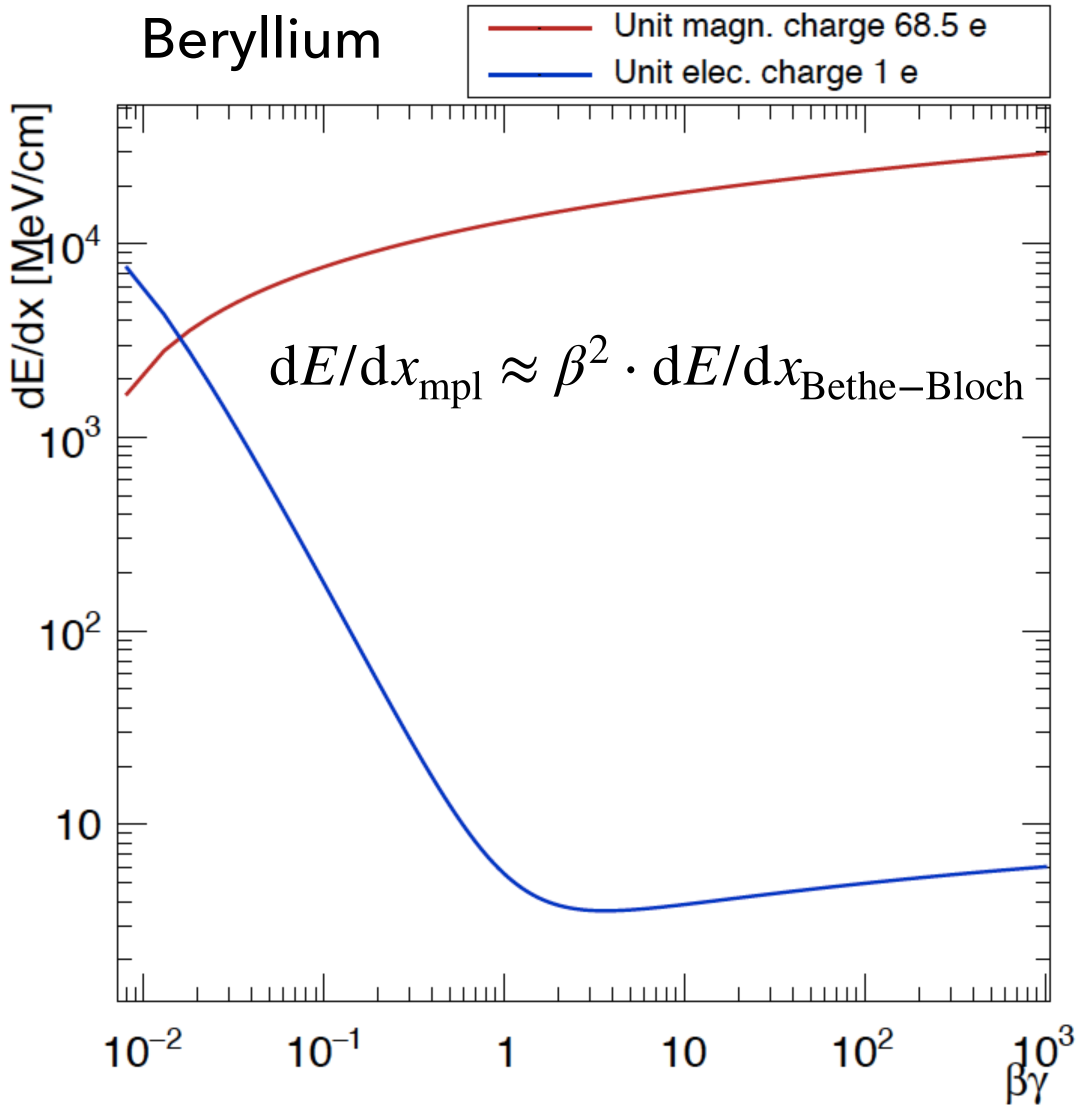
# WHAT ARE HIGHLY IONIZING PARTICLES?

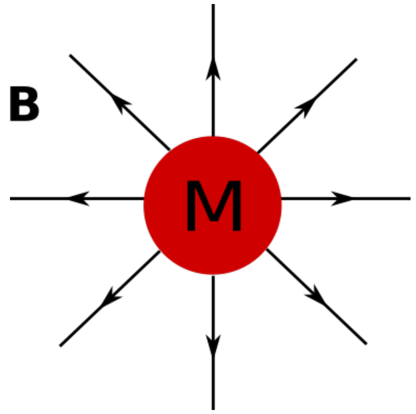
- Energy deposition of particles in matter increases with
  - decreasing momentum of projectile
    **Example:** slow pions
  - increasing mass of projectile
    **Example:** deuterons, heavy exotic particles
  - non-Bethe-Bloch energy loss
    **Example:** magnetically charged particles



# WHAT ARE HIGHLY IONIZING PARTICLES?

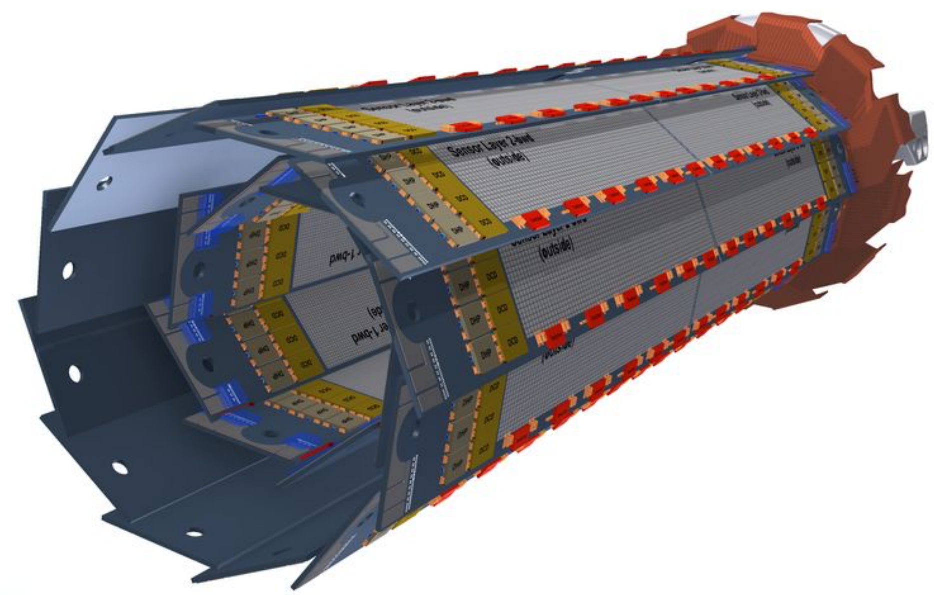
- Energy deposition in metal increases with
  - decreasing momentum of projectile
    **Example:** slow pions
  - increasing mass of projectile
    **Example:** deuterons, heavy exotic particles
  - non-Bethe-Bloch energy loss
    **Example:** magnetically charged particles

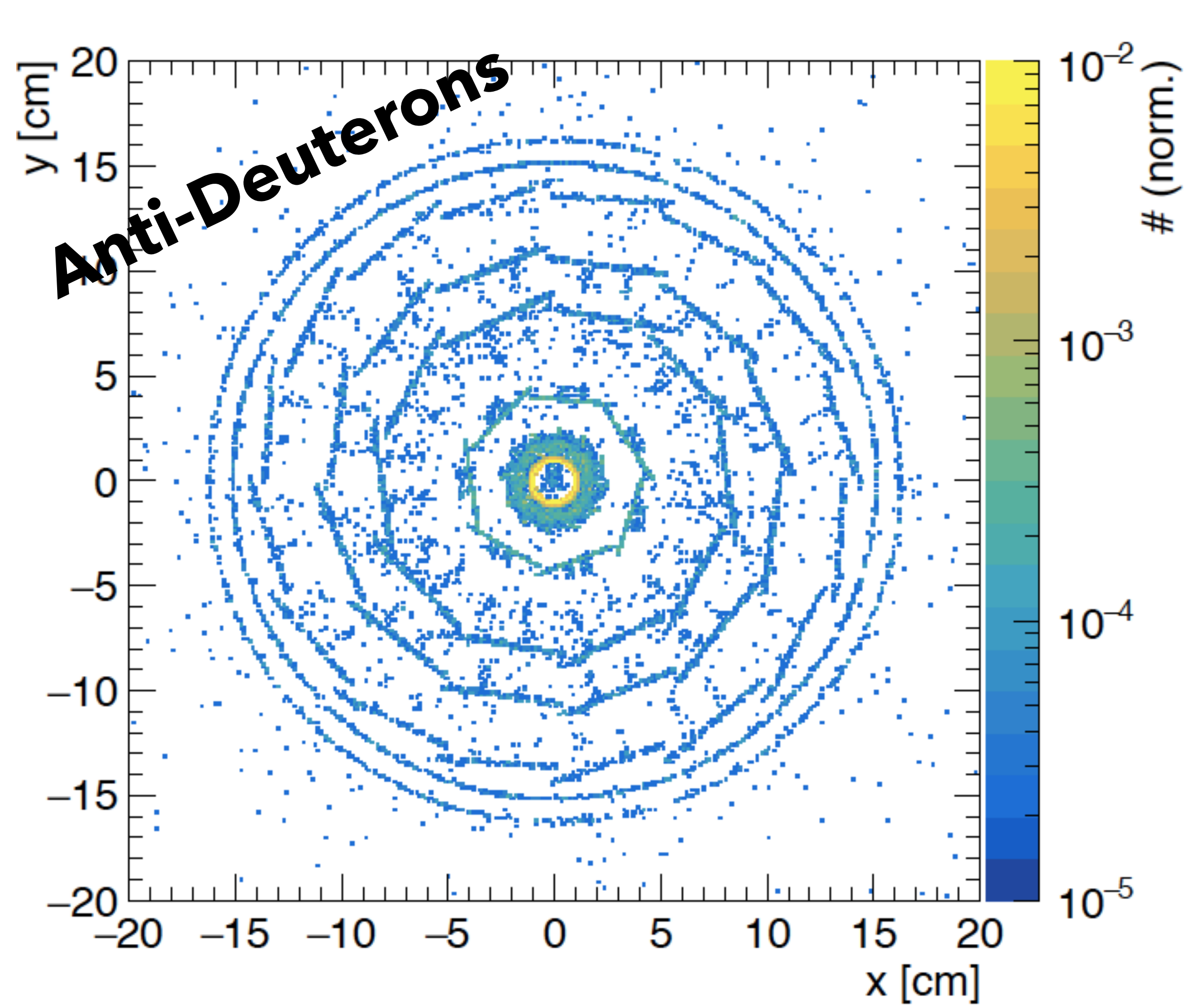


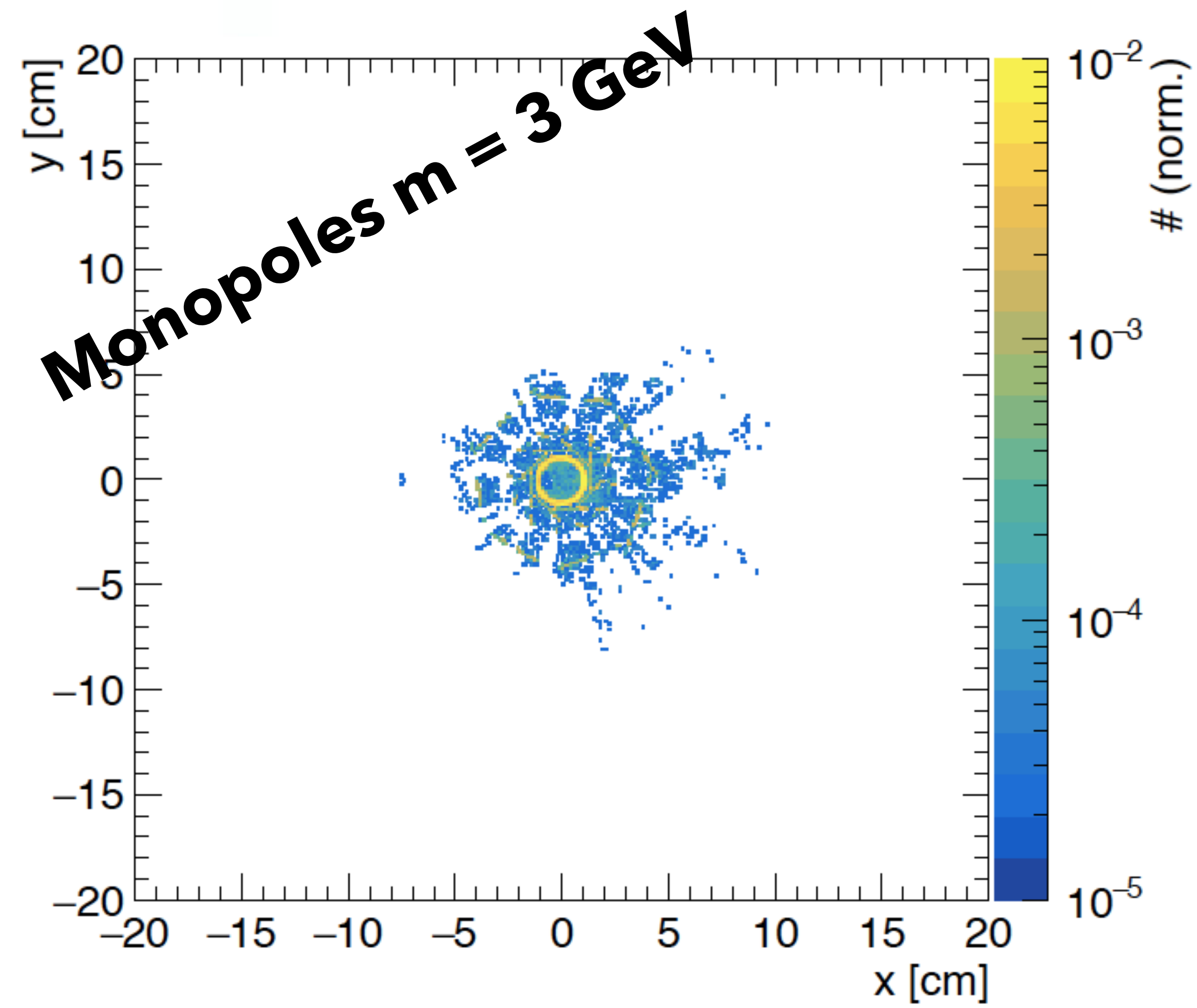



# RANGE IN VERTEX REGION

- Maximum range in vertex region illustrate strong confinement of highly ionizing particles
- Signal is discarded by ONSEN due to lack of reconstructed particle tracks





# CLUSTER RESCUE

Detection/identification of highly ionizing particles solely by PXD data

**Offline Analysis**

- Data recorded by using random triggers (mainly phase 2)
- Focus on **unsupervised learning techniques** to perform model-independent, unbiased analysis

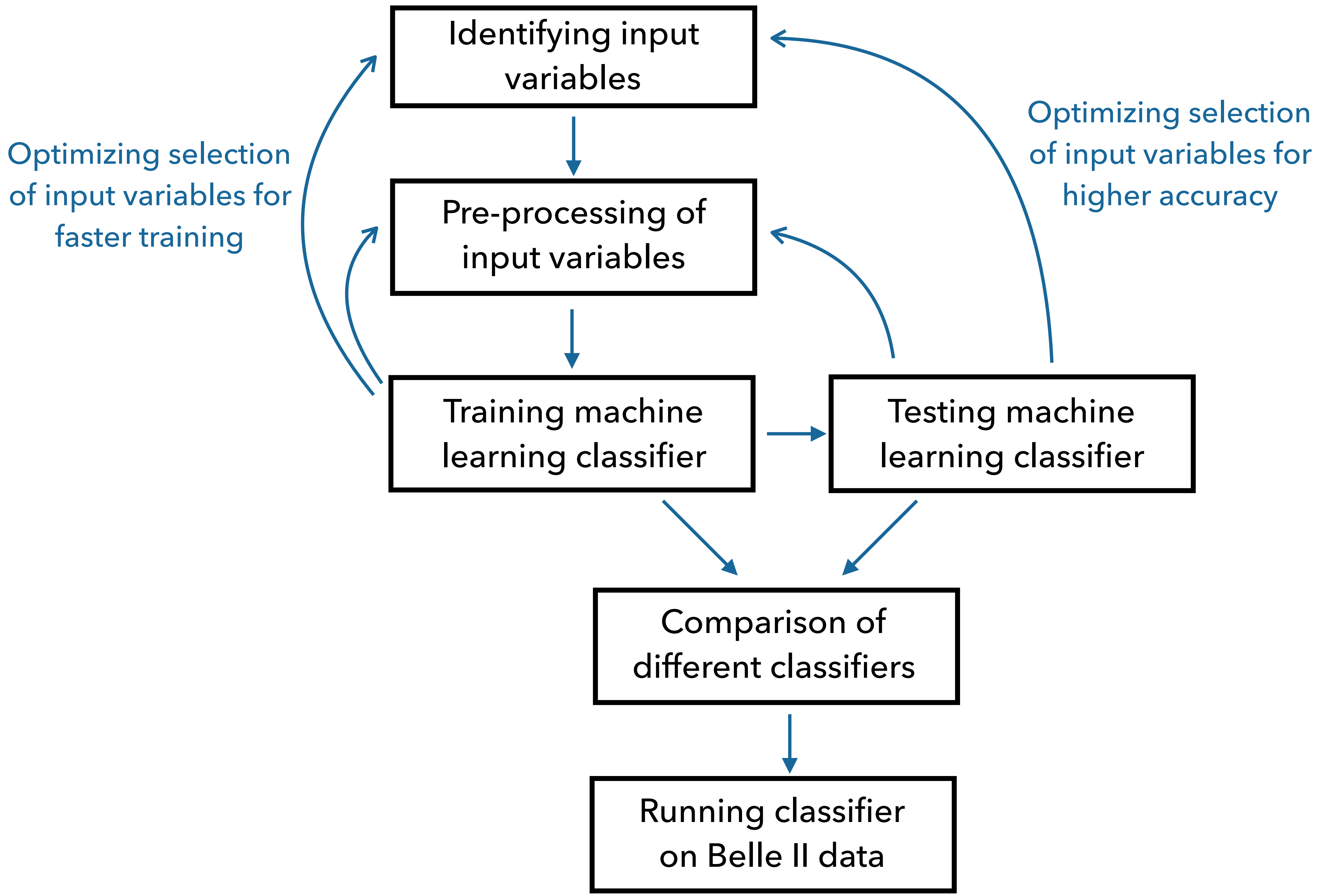
**Online Analysis**

- Implemented on and therefore optimized for FPGAs
- Extension to the ONSEN
- Similar efforts from Karlsruhe and Munich

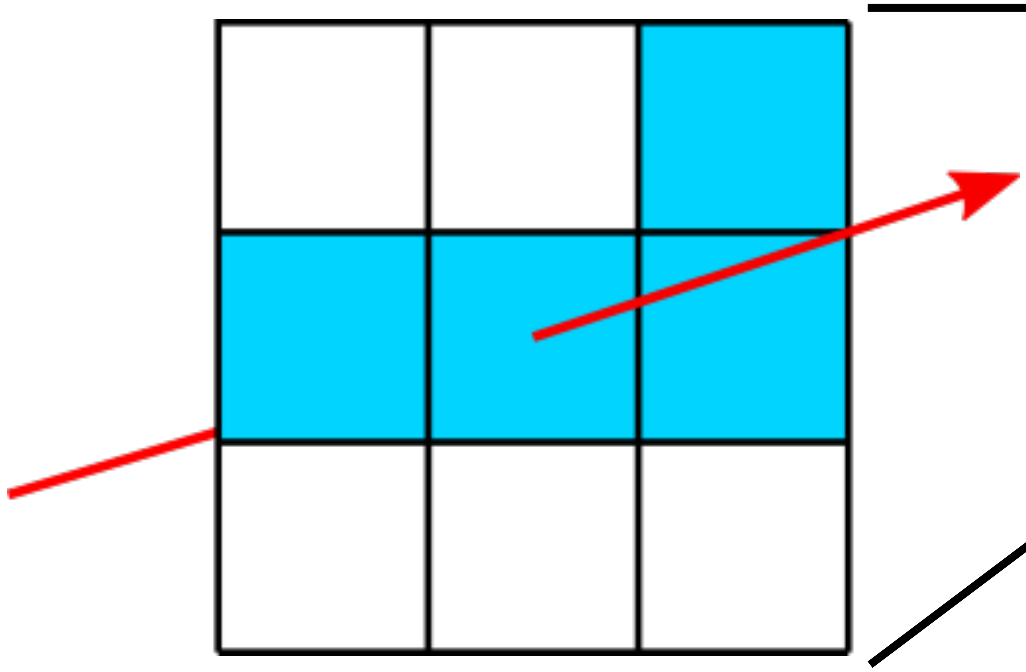
Taking advantage of state-of-the-art machine learning techniques

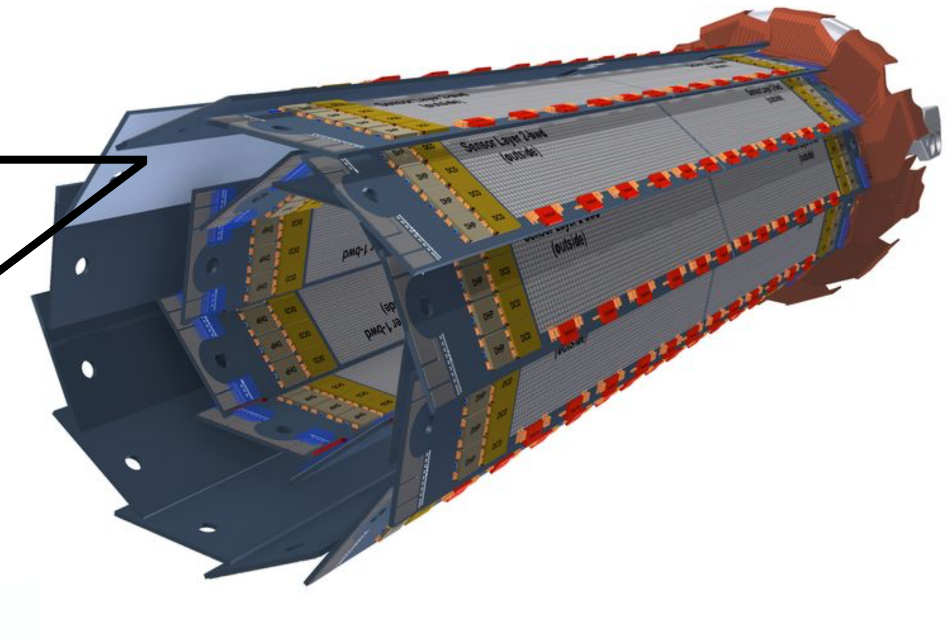
# ANALYSIS STRATEGY

# ANALYSIS STRATEGY



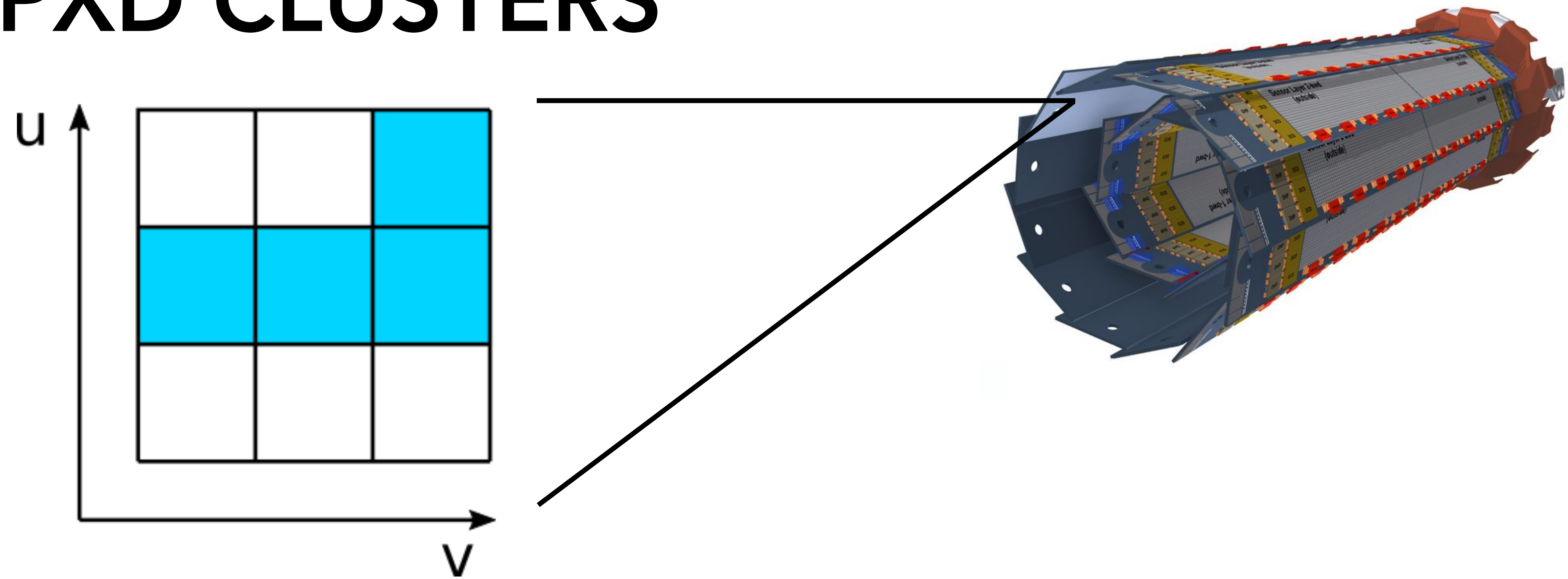
# FROM PARTICLES TO PXD CLUSTERS





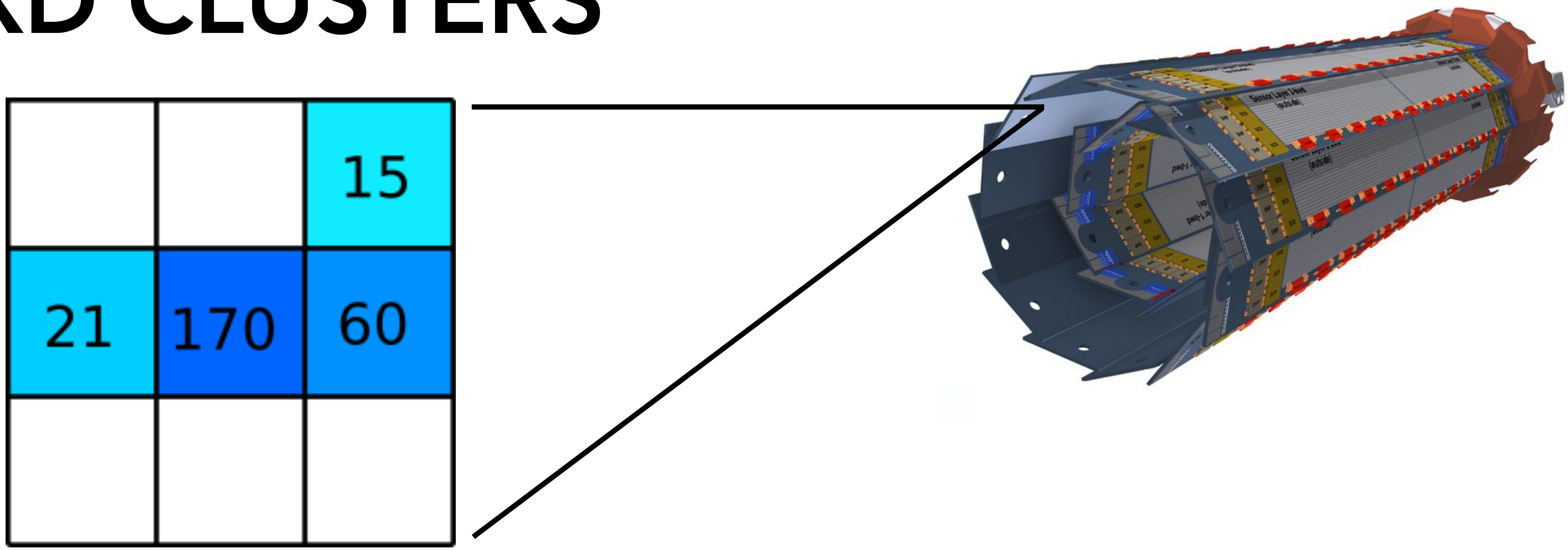
- Interaction of particle with PXD yields cluster

# FROM PARTICLES TO PXD CLUSTERS



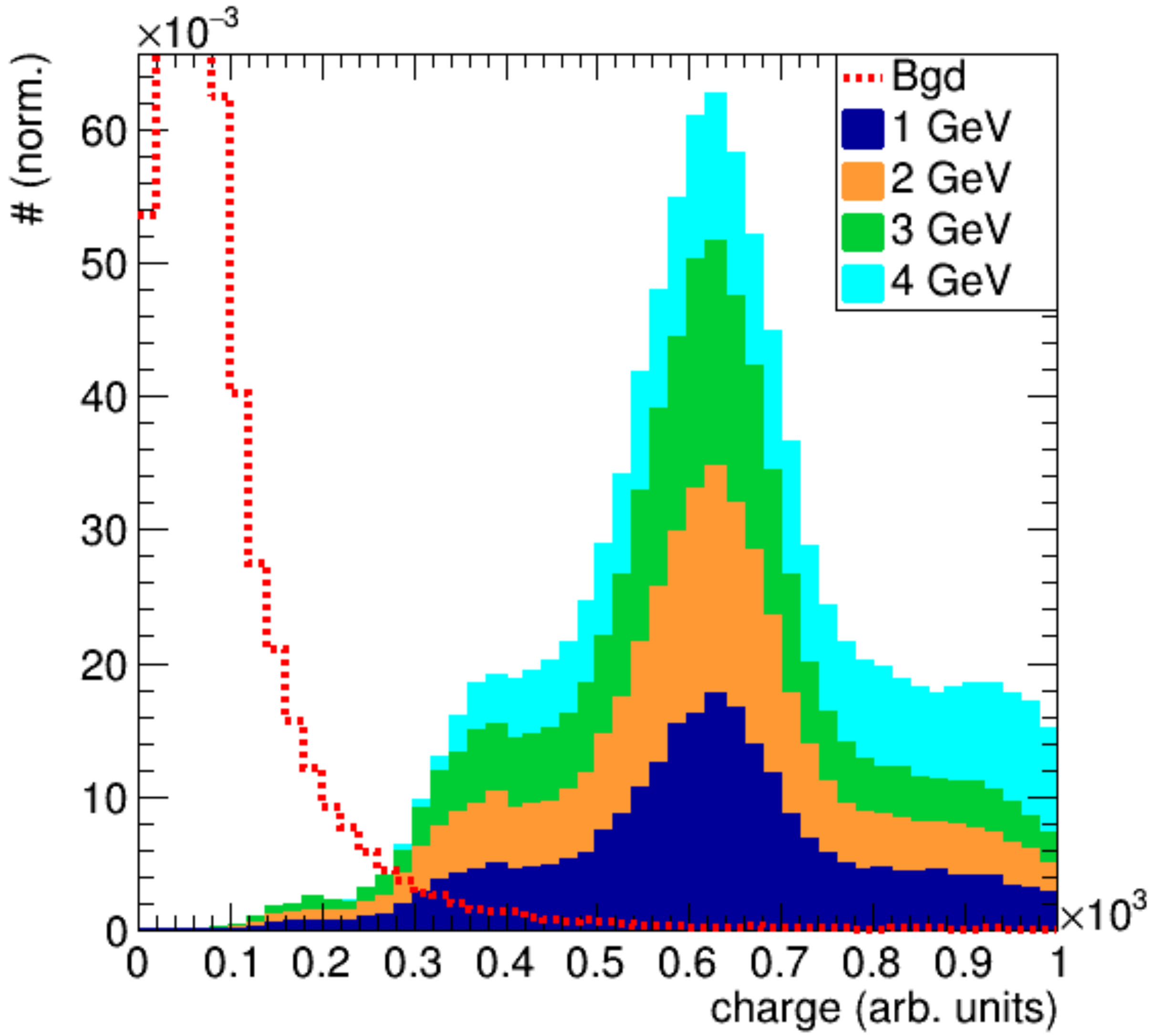

- Interaction of particle with PXD yields cluster
- <u>Basic</u> cluster properties used for PID:
  - Cluster size (+ size in u/v direction)
  - Cluster charge (+ maximum/minimum pixel charge )
- Considered but not used: cluster length, cluster angle

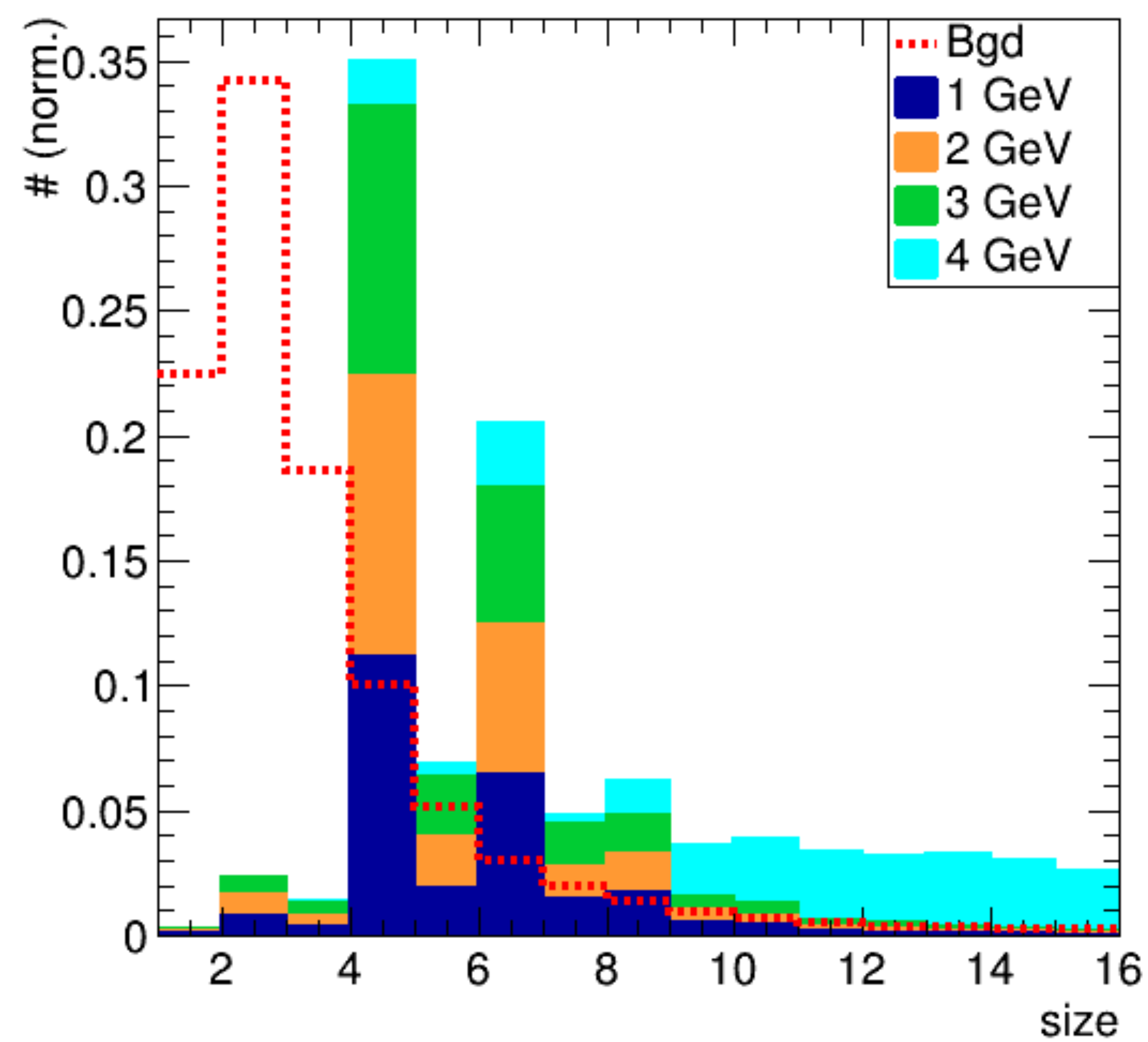
# FROM PARTICLES TO PXD CLUSTERS




- Interaction of particle with PXD yields cluster
- <u>Basic</u> cluster properties used for PID:
    - Cluster size (+ size in u/v direction)
    - Cluster charge (+ maximum/minimum pixel charge )
- Considered but not used: cluster length, cluster angle

# CLUSTER PROPERTIES EXAMPLES

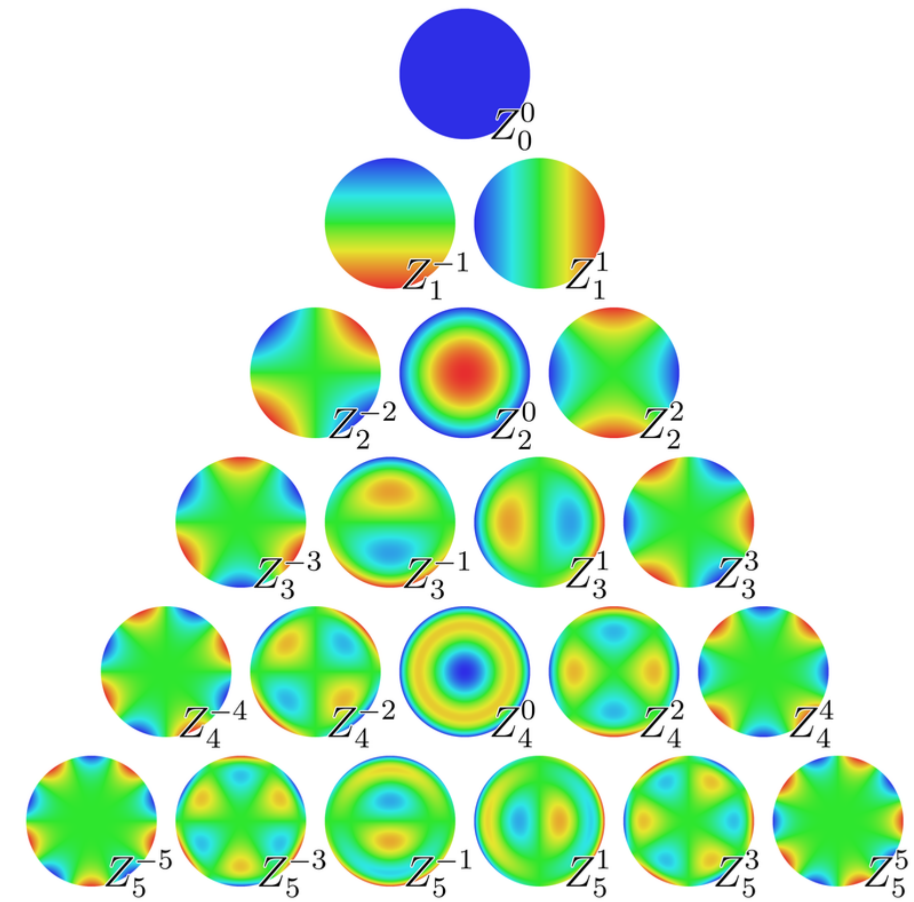


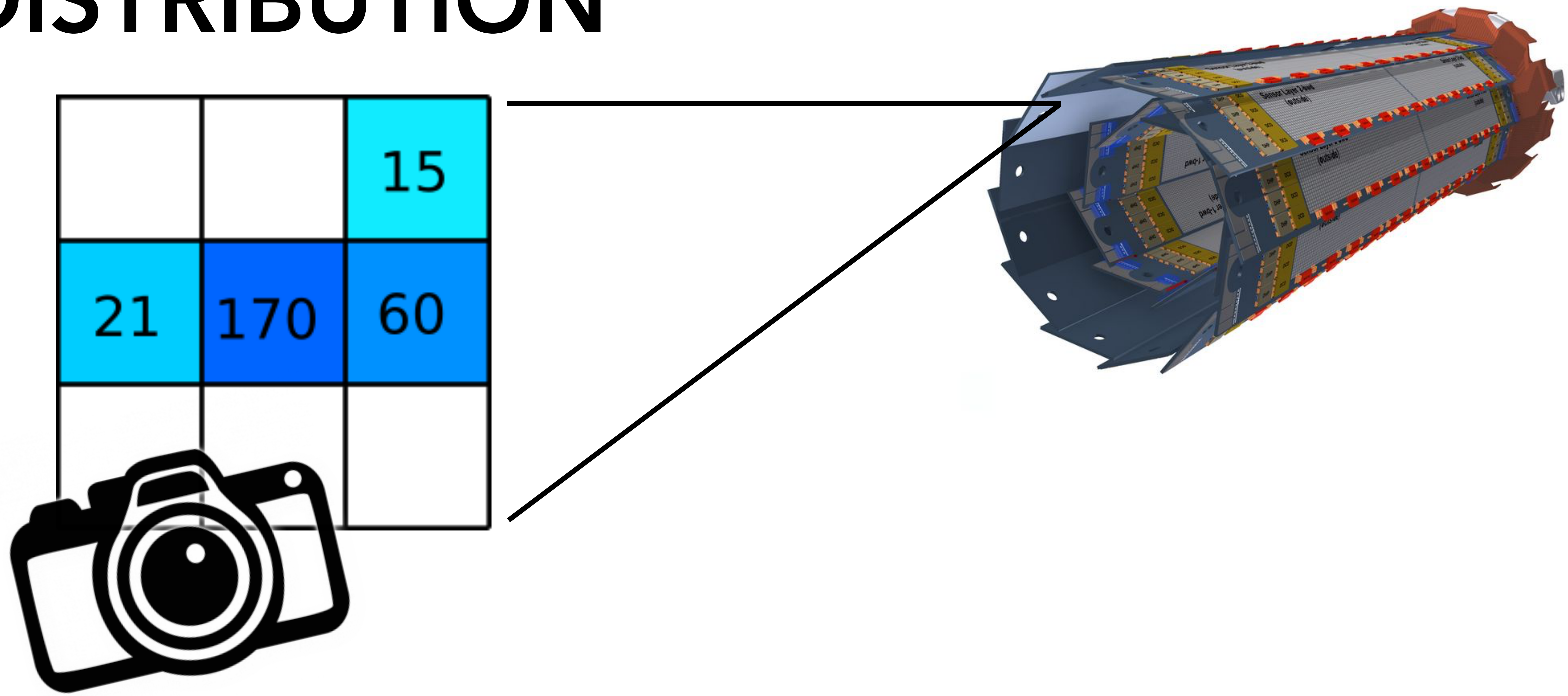

High energy deposition of magnetic monopoles yields
**large high-charge clusters**

# CLUSTER SHAPE ANALYSIS

- Objective: Generate *new* cluster observables which help in discriminating signal from background
- Expansion of pixel charge distribution in terms of orthogonal polynomials:
    - Image moments by Flusser: invariant under scaling, rotation, skewness, kurtosis
    - Zernike moments (up to 3rd order): invariant under rotation



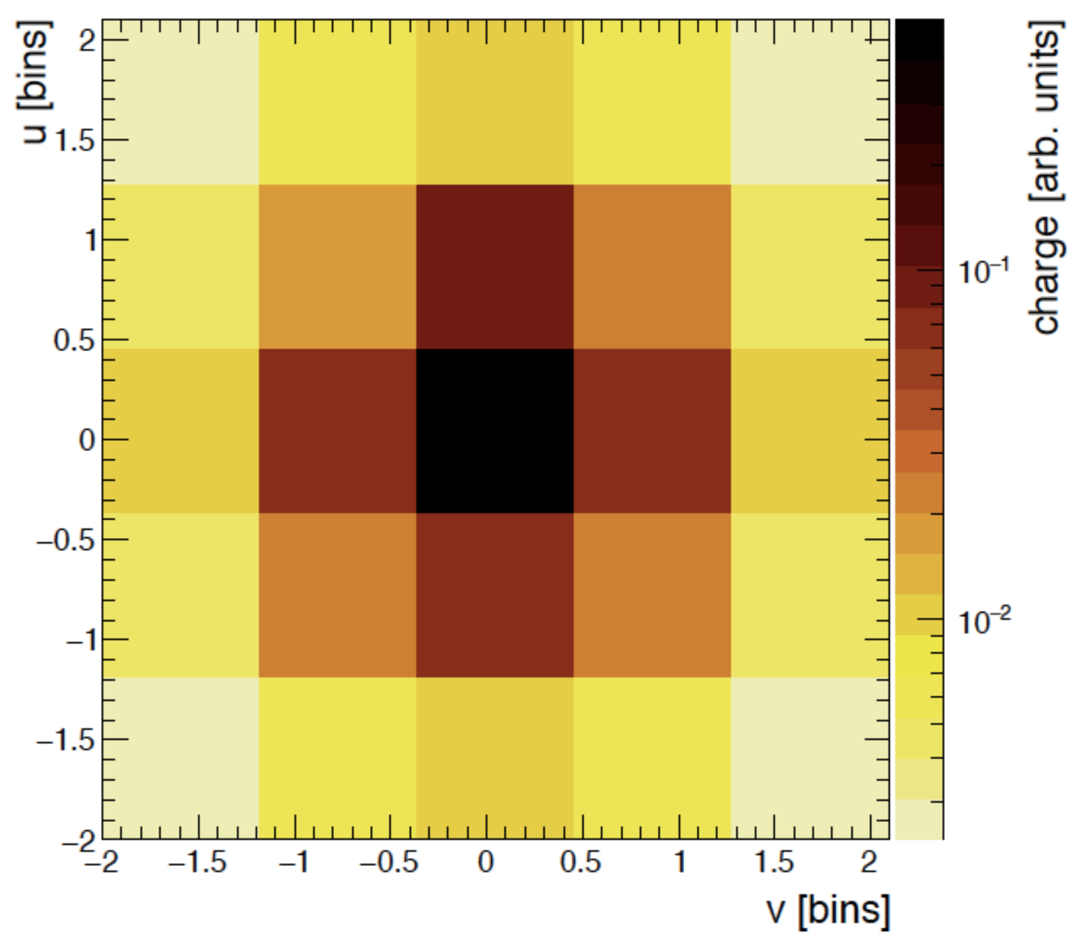
# PIXEL CHARGE DISTRIBUTION



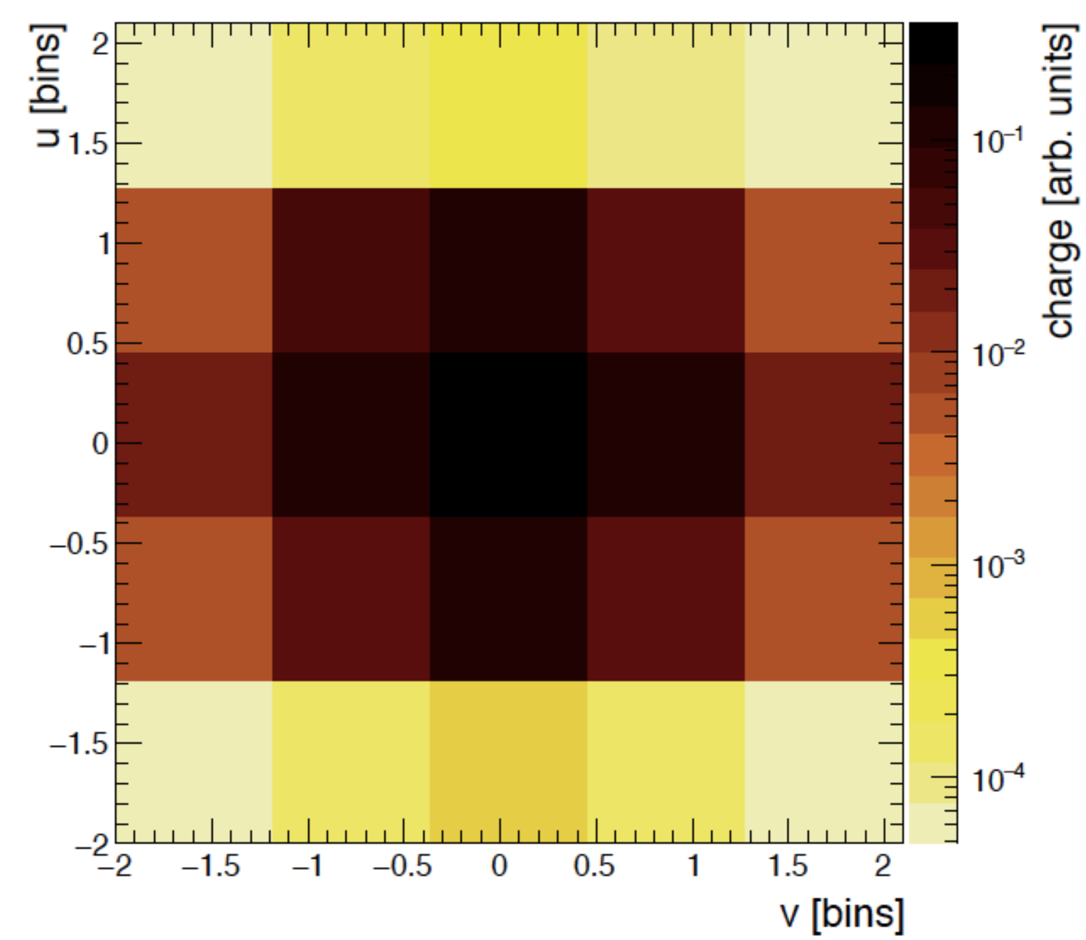

- Alternative to cluster properties: pixel charge distribution (*image of cluster*)
- Exploit existing image recognition techniques (most notably convolutional neural network)

# PIXEL CHARGE DISTRIBUTION EXAMPLES

**Background**




**Magnetic Monopoles m = 3 GeV**

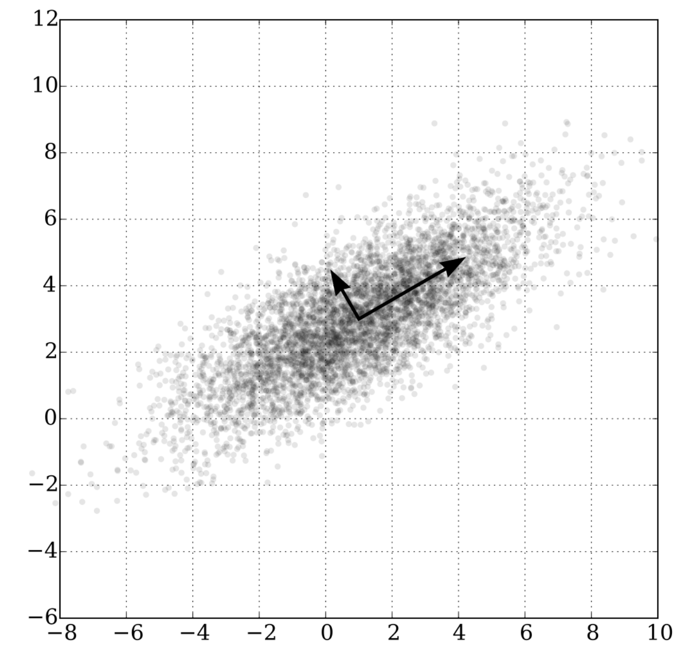


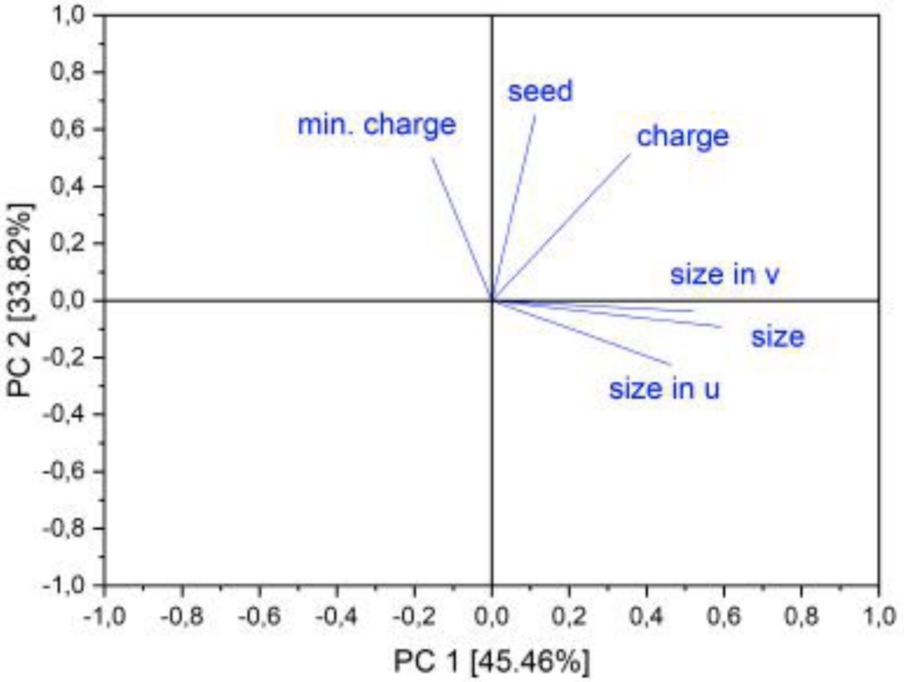

Average pixel charge distribution of background and magnetic monopoles

# PRE-PROCESSING: PRINCIPAL COMPONENT ANALYSIS

- Principal component transformation generates linear combination which maximizes variance of a given input data set

- Allows to filter out redundant information and generate variables with higher discriminative power






- Principal component transformation grouped basic input variables in *size-like* and *charge-like*

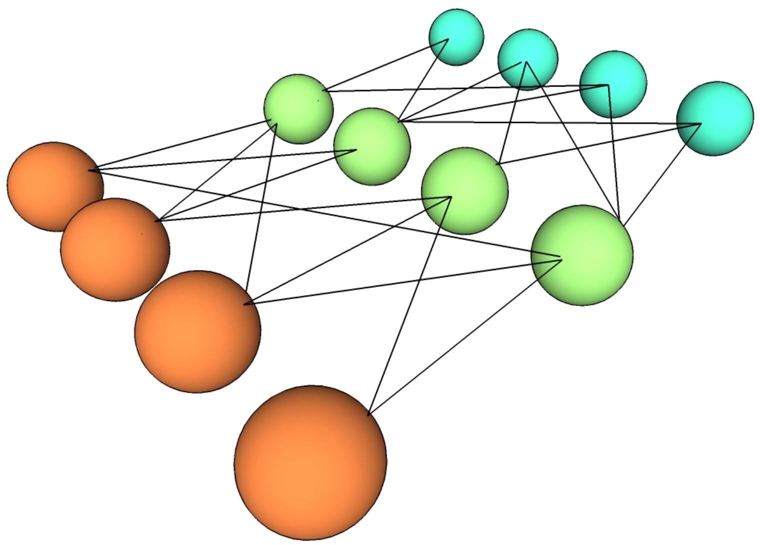
- Reduction from 6 to 4 input variables possible with information loss of only ~2.5%

# OFFLINE ANALYSIS

# SUPERVISED LEARNING

# FEED-FORWARD NEURAL NETWORKS

- Supervised learning in order to separate HIPs from beam background*
- Implemented with PyTorch and trained on CPU and GPU



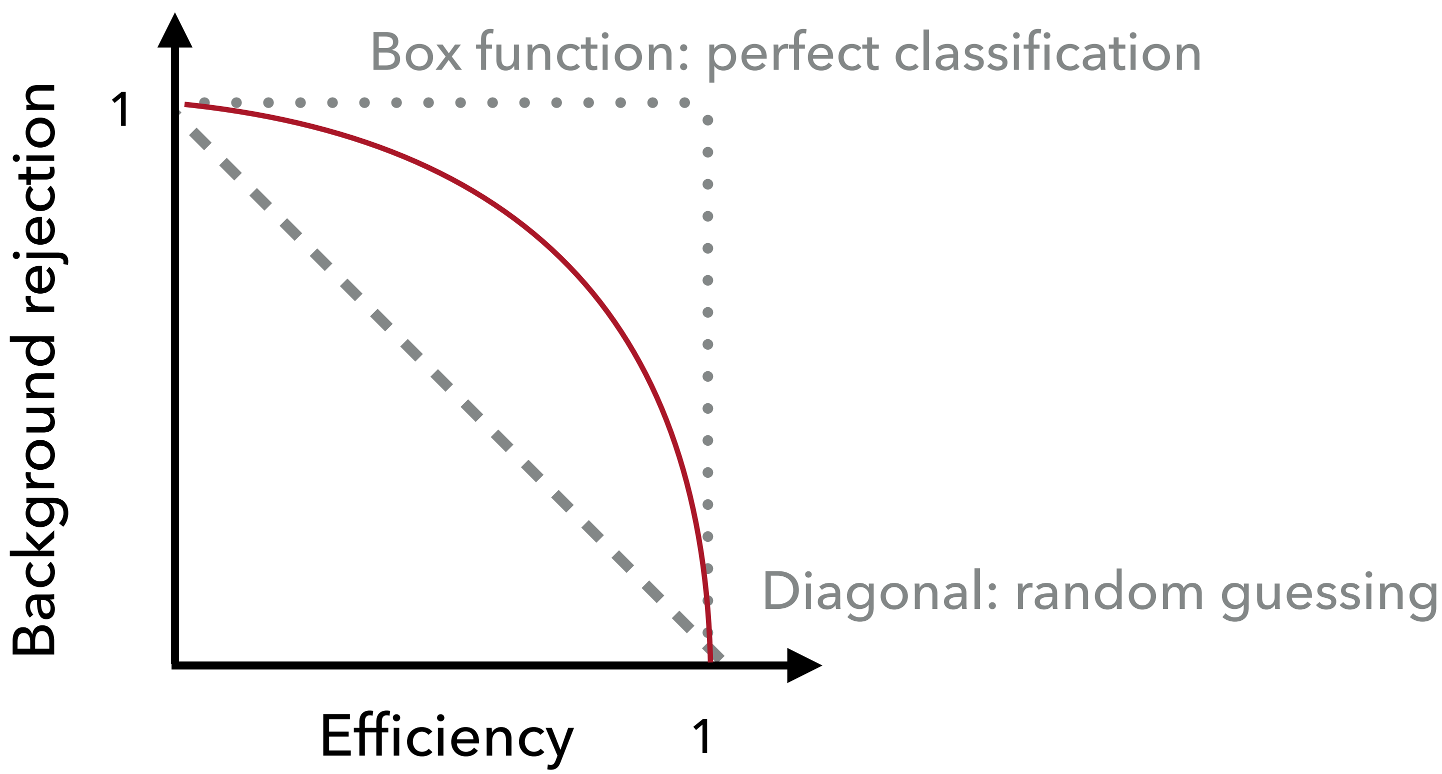
| **Feed-Forward Network Parameters** |
| --- |
| 4 layers / 2 hidden |
| > 50 nodes per layer |
| ReLu Activation Function |
| CrossEntropy Loss Function |
| Stochastic Gradient Descent (SGD) Optimizer |
| Batch Size: 256 |
| Learning Rate: 0.0001 |
| Momentum: 0.9 |

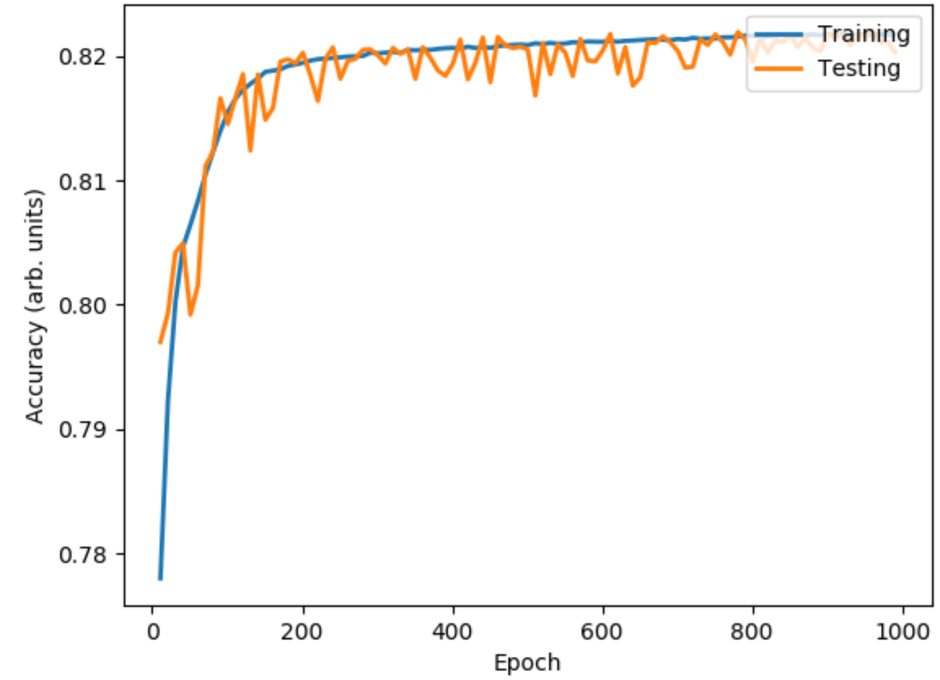
*beam background

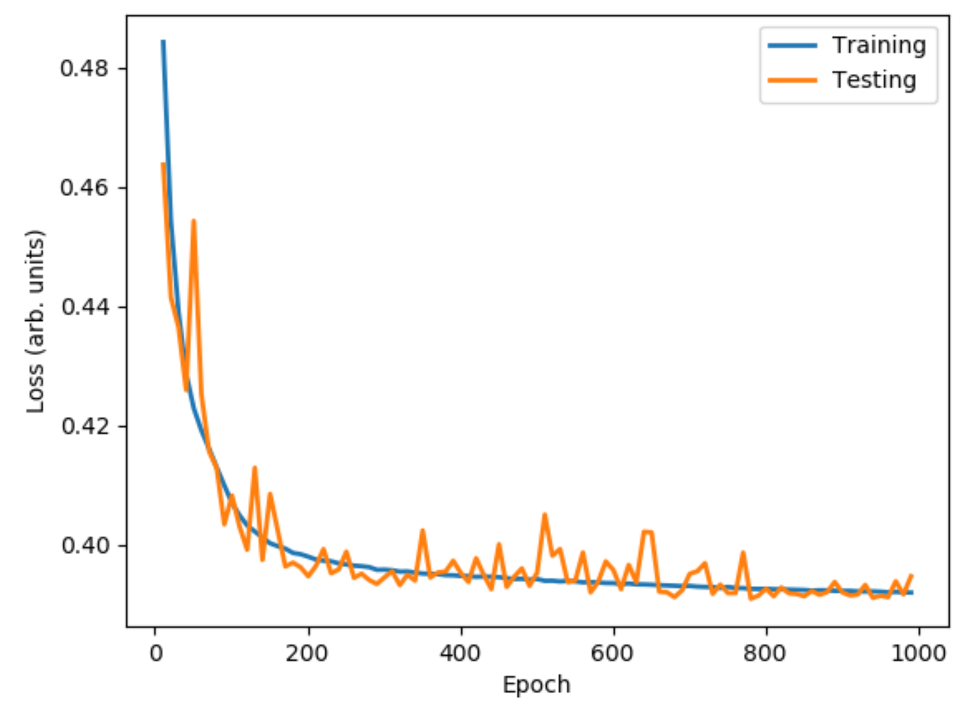
- Official mixed MC beam background
- Includes luminosity-dependent and beam-induced background

# FEED-FORWARD NEURAL NETWORKS

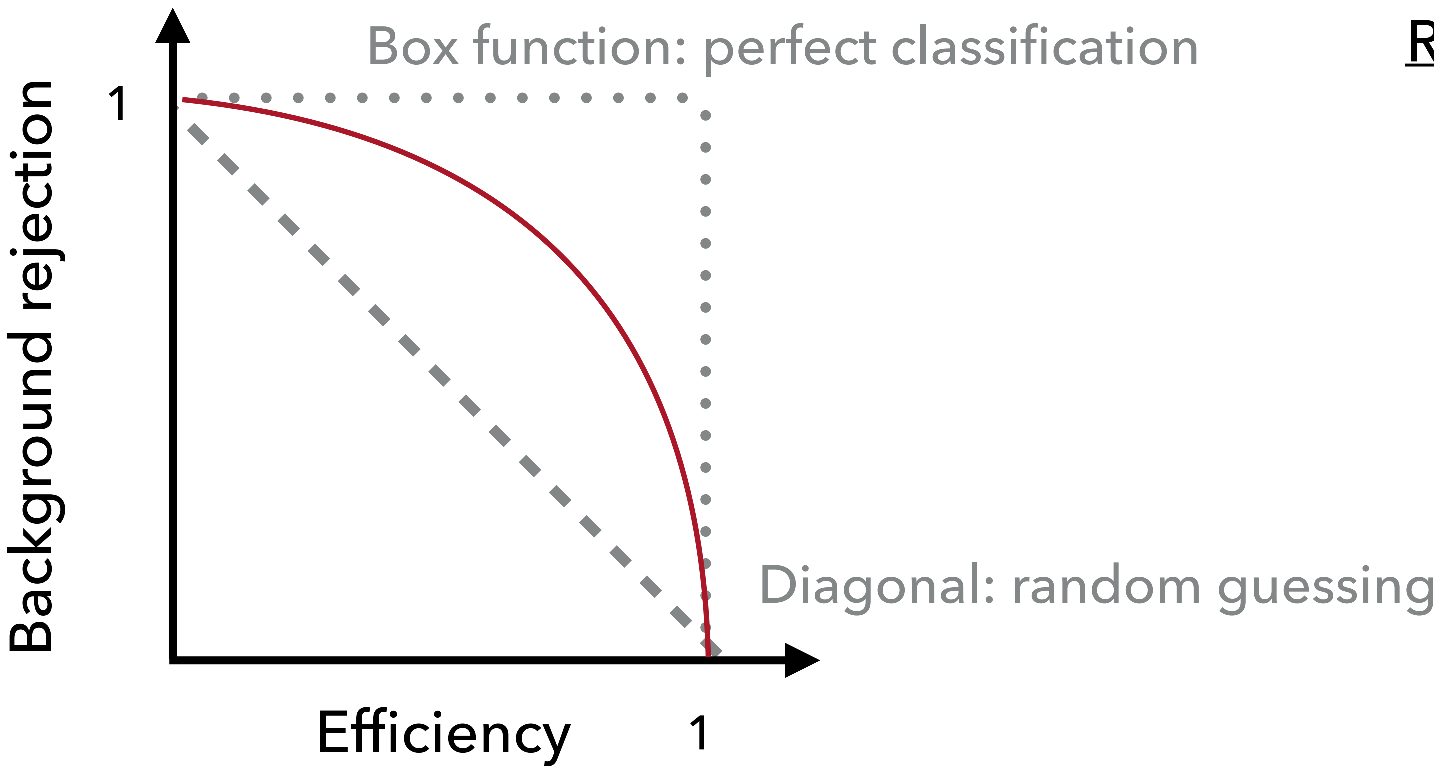
- Loss and accuracy is monitored during training (~8h)
- Cut on classification axis determines accuracy of neural network



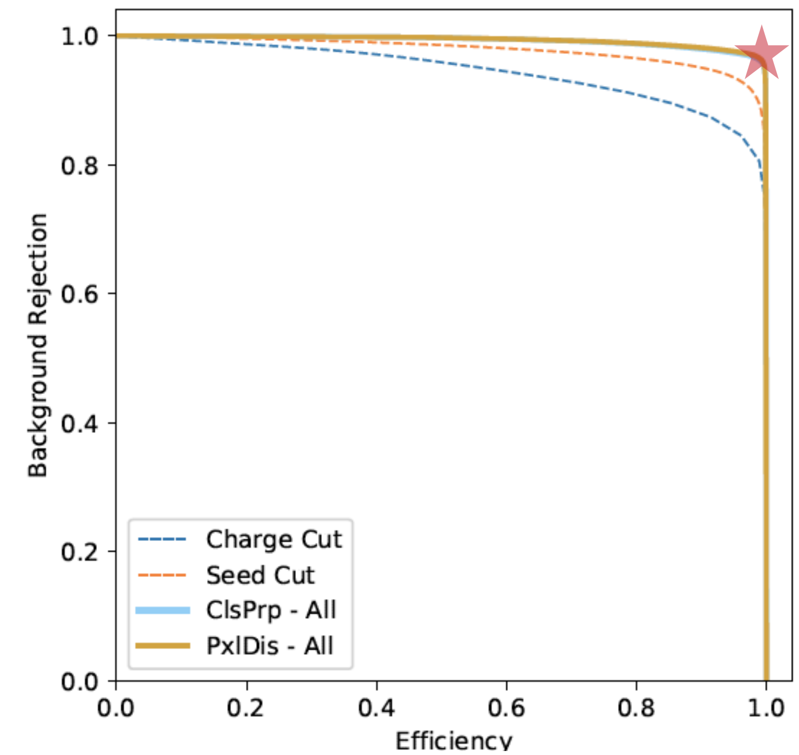



# ROC-CURVES

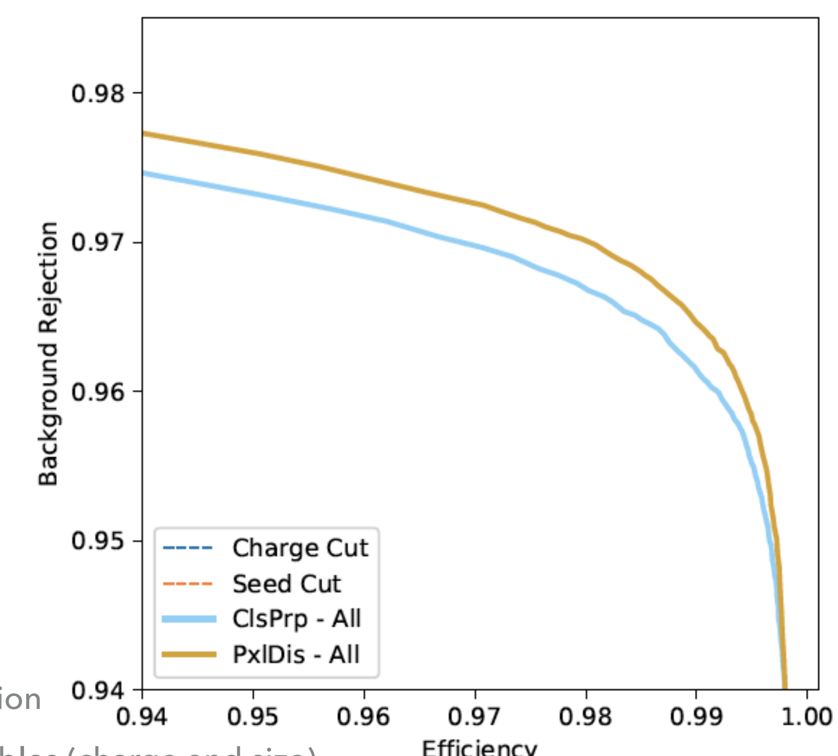



## ROC curves for anti-deuteron identification




- Neural networks perform better than linear cut

- Using pixel distribution yields better results compared to cluster properties




PxlDis = Pixel charge distribution

ClsPrp = Basic cluster observables (charge and size)

# FEED-FORWARD NEURAL NETWORKS

- Studied influence of charge, momentum, mass and cluster size
- Feed-Forward Networks are highly successful but they require supervised learning
    - Labeled data has to be available/ reliable
    - Unsuited for model-independent search for *new physics*
    - However: often accuracy is traded off in order to perform unsupervised learning

Results for magnetic monopoles

| Input set | Mass [GeV] | Training [%] | Testing [%] | AUC [%] |
|---|---|---|---|---|
| ClsPrp | 1 - 4 | 99.0 | 99.0 | 99.932 |
| PxlDis | 1 - 4 | 98.8 | 98.8 | 99.873 |
| ClsPrp | 1 | 98.6 | 98.6 | 99.877 |
| PxlDis | 1 | 98.8 | 98.7 | 99.890 |
| ClsPrp | 4 | 99.7 | 99.8 | 99.966 |
| PxlDis | 4 | 99.8 | 99.7 | 99.972 |

Training & Testing evaluated at point marked with a star on previous slide

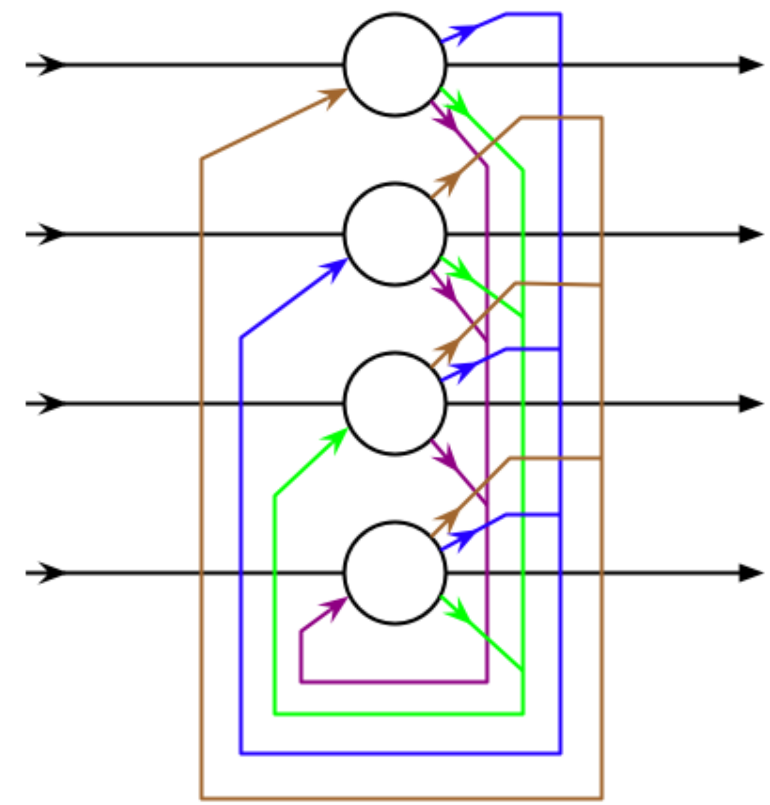
ClsPrp = Basic cluster observables (charge and size)
PxlDis = Pixel charge distribution

Results for anti-deuterons

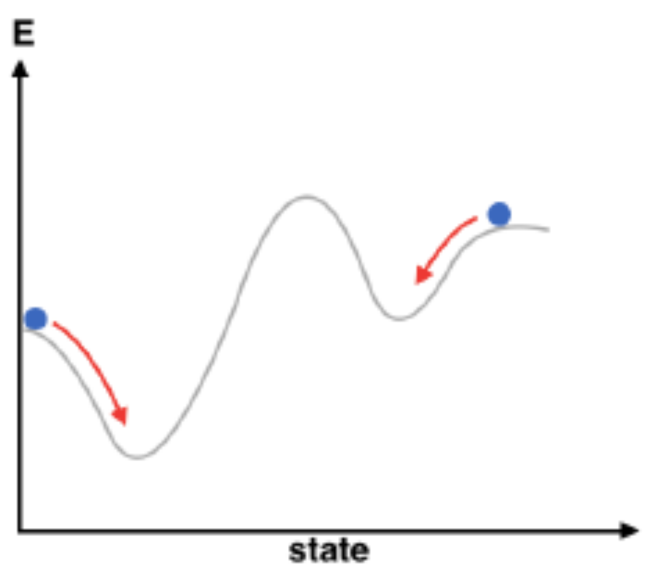
| $p_{\max}$ [GeV] | Input set | Cluster types | Training [%] | Testing [%] | AUC [%] |
|---|---|---|---|---|---|
| 1 | ClsPrp | all | 97.5 | 97.4 | 99.24 |
| 1 | ClsPrp | multi pixel | 97.3 | 97.4 | 99.26 |
| 1 | ClsPrp | single pixel | 97.7 | 97.7 | 98.82 |
| 3 | ClsPrp | all | 82.2 | 82.1 | 90.19 |
| 3 | ClsPrp | multi pixel | 83.1 | 83.2 | 91.34 |
| 3 | ClsPrp | single pixel | 78.7 | 78.7 | 85.94 |
| 1 | PxlDis | all | 97.7 | 97.7 | 99.27 |
| 1 | PxlDis | multi pixel | 97.7 | 97.7 | 99.32 |
| 1 | PxlDis | single pixel | 97.7 | 97.7 | 98.81 |
| 3 | PxlDis | all | 82.6 | 82.6 | 90.58 |
| 3 | PxlDis | multi pixel | 83.6 | 83.6 | 91.70 |
| 3 | PxlDis | single pixel | 78.7 | 78.7 | 86.09 |

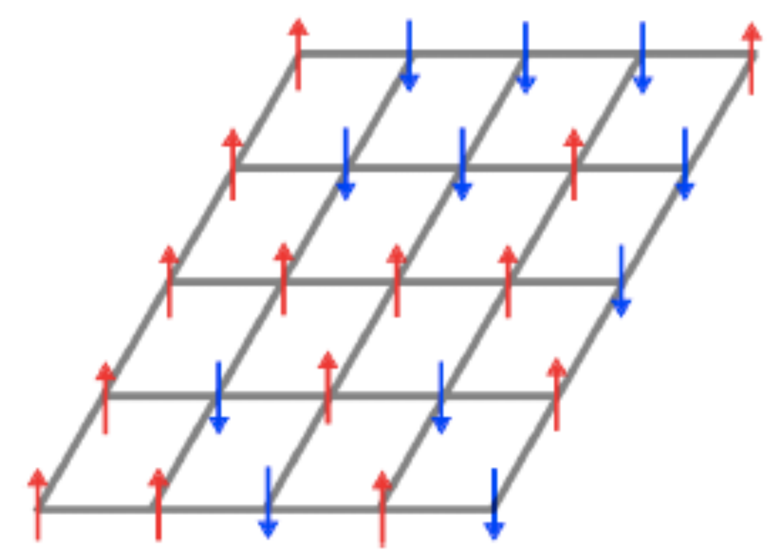
# UNSUPERVISED LEARNING

# HOPFIELD NETWORKS



- Recurrent binary neural network (*associative* memory)
- Network learns *pattern* associated with background/signal input vector -> stored in weight matrix
- Weight between neurons determines *energy* of the entire network
- Stable state is reached when energy of network is minimized (similar to spin-spin interaction in quantum mechanical many-body systems)
- Incomplete or distorted patterns are recognized



# HOPFIELD NETWORKS

Belle II
JUSTUS-LIEBIG-UNIVERSITÄT GIESSEN

- So far, Hopfield network is trained to store 4 patterns (2 background, 2 signal)

- Custom activation function (instead of binary activation) is used to feed additional information into the network

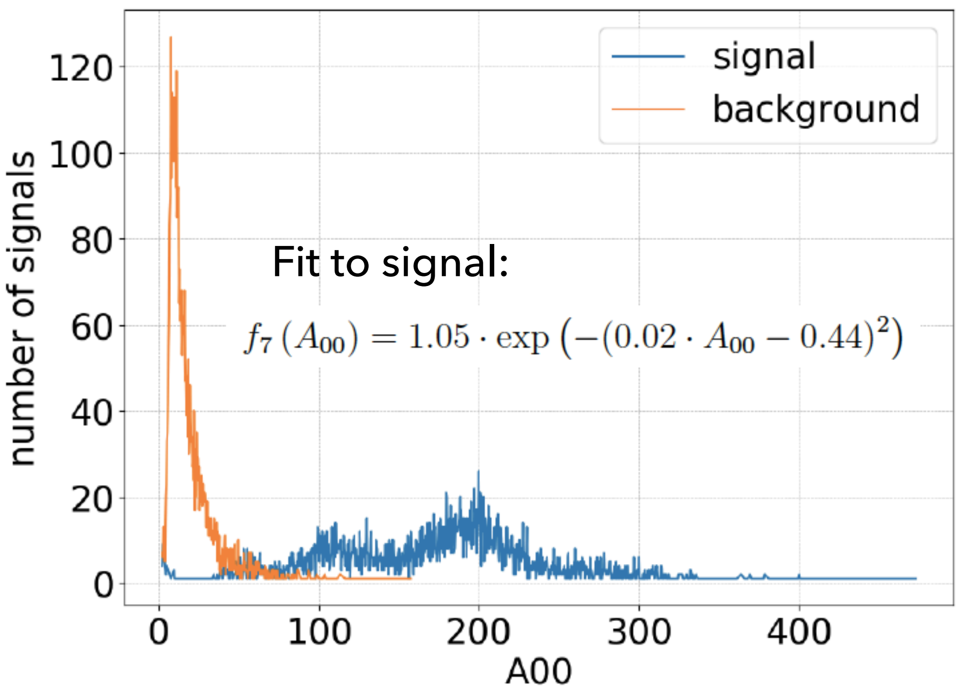
- Local, global properties and Zernike moments used in final analysis

$$\begin{pmatrix} w_{11}^{signal} & \cdots & w_{n1}^{signal} \\ \vdots & \ddots & \vdots \\ w_{1n}^{signal} & \cdots & w_{nn}^{signal} \end{pmatrix} \qquad \begin{pmatrix} w_{11}^{background} & \cdots & w_{n1}^{background} \\ \vdots & \ddots & \vdots \\ w_{1n}^{background} & \cdots & w_{nn}^{background} \end{pmatrix}$$

$f \qquad 1-f$

$$\begin{pmatrix} w_{11} & \cdots & w_{n1} \\ \vdots & \ddots & \vdots \\ w_{1n} & \cdots & w_{nn} \end{pmatrix}$$

Zernike Moment A00




Separation of magnetic monopoles from beam background

| 3D | local prop. | global prop. | Zernike mom. | accuracy |
|---|---|---|---|---|
| ✓ | ✗ | ✗ | ✗ | 63.0 % |
| ✓ | ✓ | ✗ | ✗ | 76.5 % |
| ✓ | ✓ | ✓ | ✗ | 95.7 % |
| ✓ | ✓ | ✓ | ✓ | 97.7 % |

Local prop - cluster size + cluster size in x/y + max. charge + cluster charge

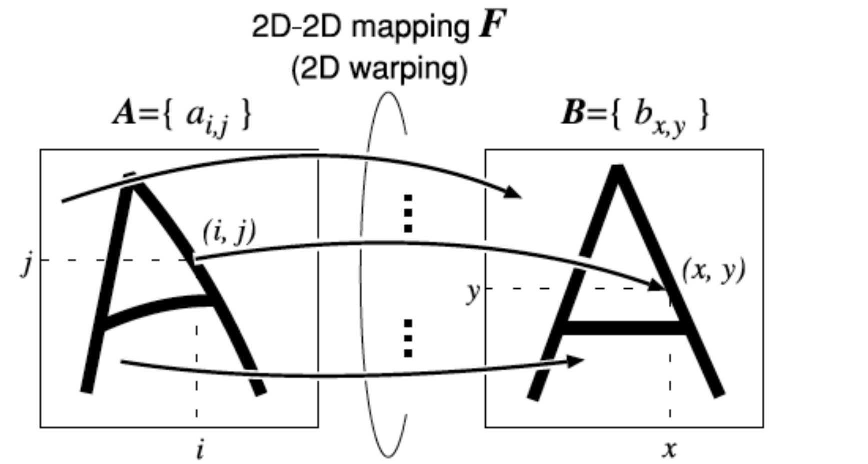
Global prop. - local prop + global position

Zernike mom. - global prop + Zernike moments

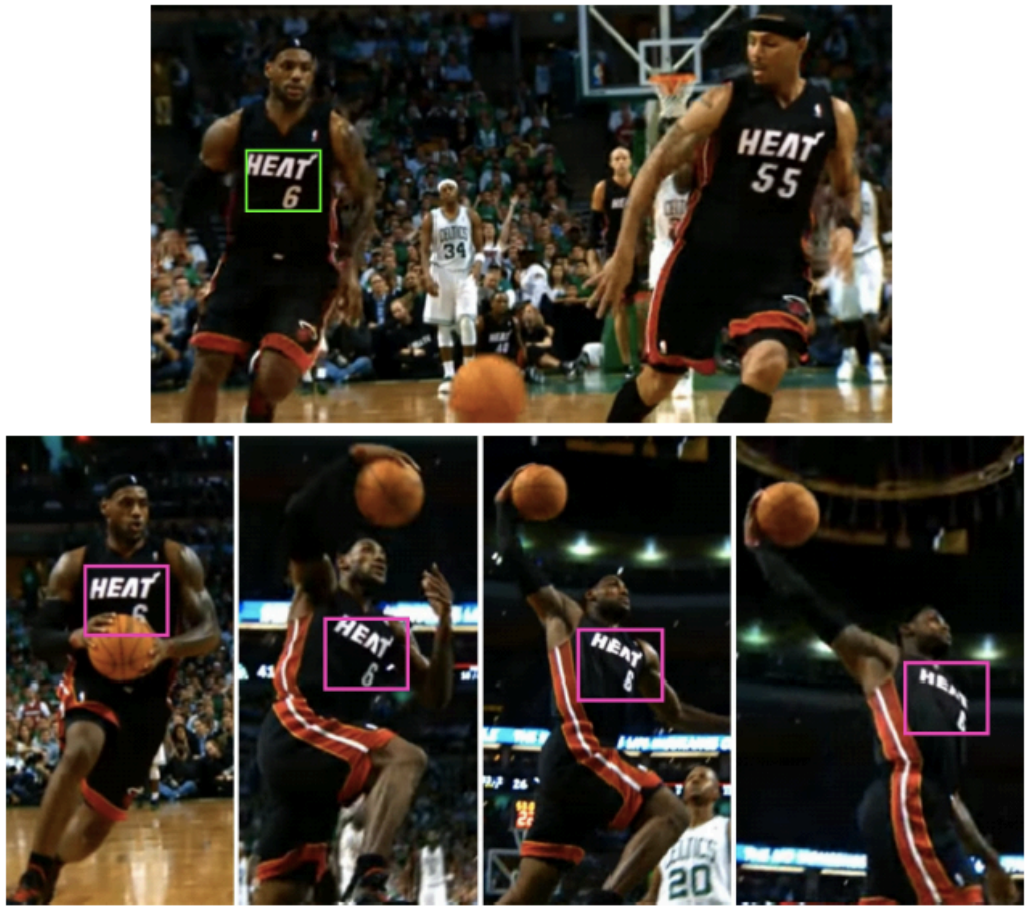
# ELASTIC MATCHING (EM) NEURAL NETWORK

- EMs are employed for pattern recognition problems (i.e. handwriting/gesture/face/… recognition)
- Focus on a subset of pixels displaying features which correspond between tested image and target (*template matching*)
- Crucial impact of distance measure

**No results yet - project currently under preparation**




*S. Uchida. "A survey of elastic matching techniques for handwritten character recognition." IEICE transactions on information and systems 88.8 (2005): 1781-1790.*



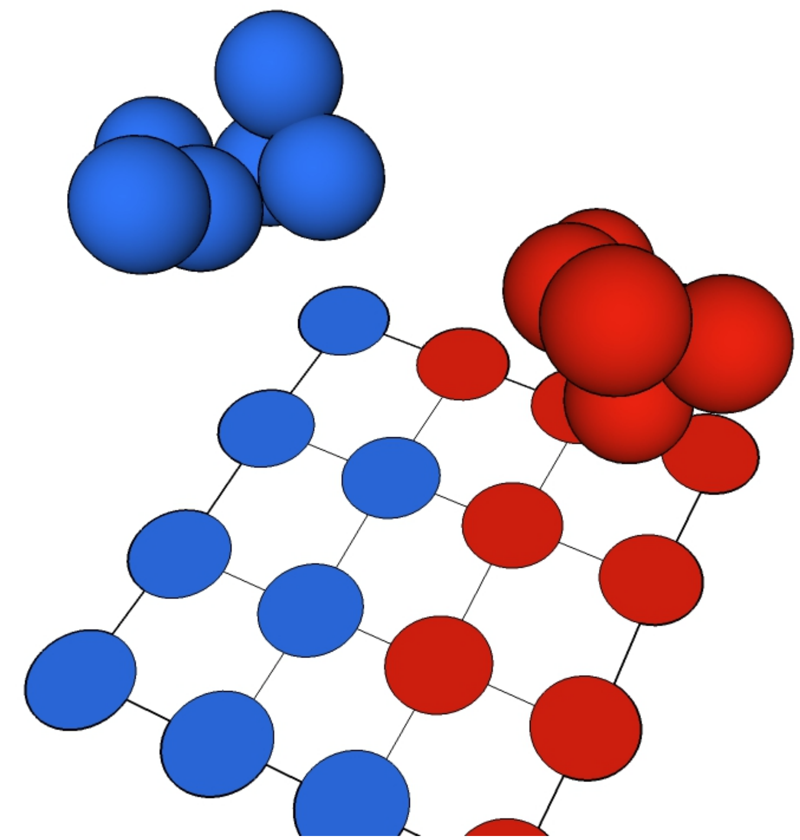
*I. Talmi et al. "Template matching with deformable diversity similarity." IEEE Conference on Computer Vision and Pattern Recognition. 2017.*

# SELF-ORGANIZING MAPS (SOMS)

- Unsupervised model-agnostic learning technique which requires no training with ground truth information

- Separation of HIPs and beam background in multi-dimensional space with subsequent transformation to two dimensions for evaluation purposes

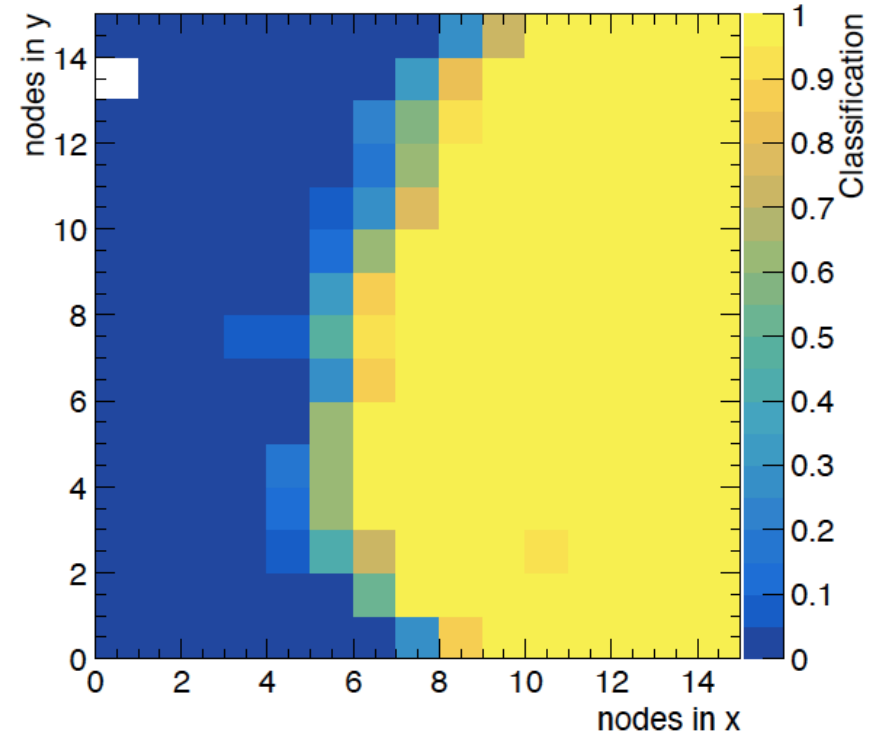
- Implemented on CPU

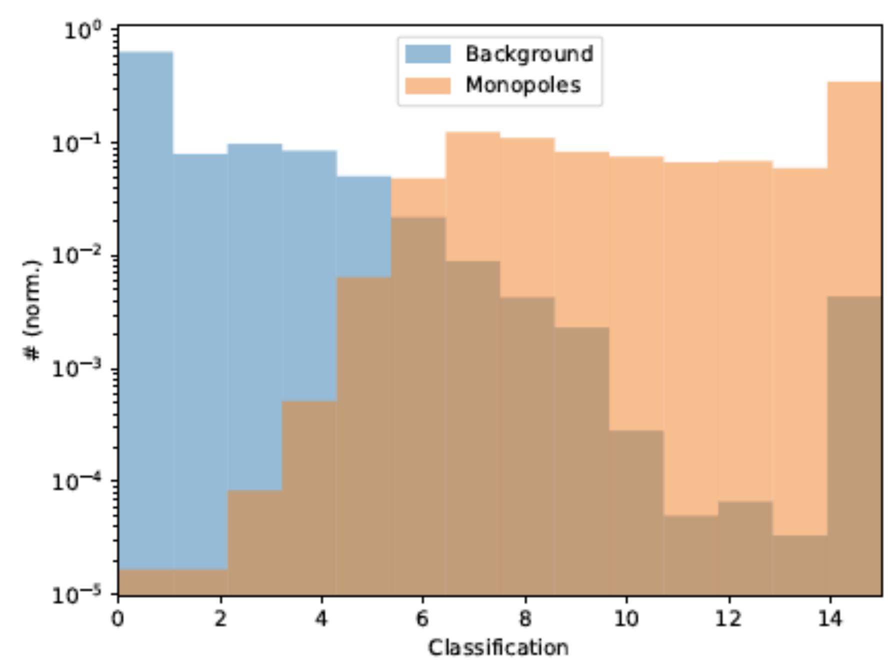


| **Self-Organizing Maps Parameters** |
|---|
| 15 x 15 Nodes |
| Neighborhood function: Gaussian |
| Width of Gaussian: 7 |
| Learning Rate: 0.01 |

# SELF-ORGANIZING MAPS (SOMS)

- Training time approx. 10x shorter compared to FFNs
- Monitoring of training process not trivial
- Trained map can be evaluated by a defining classification regions or on a bin-by-bin basis
- In our case: convergence more difficult to achieve compared to FFNs



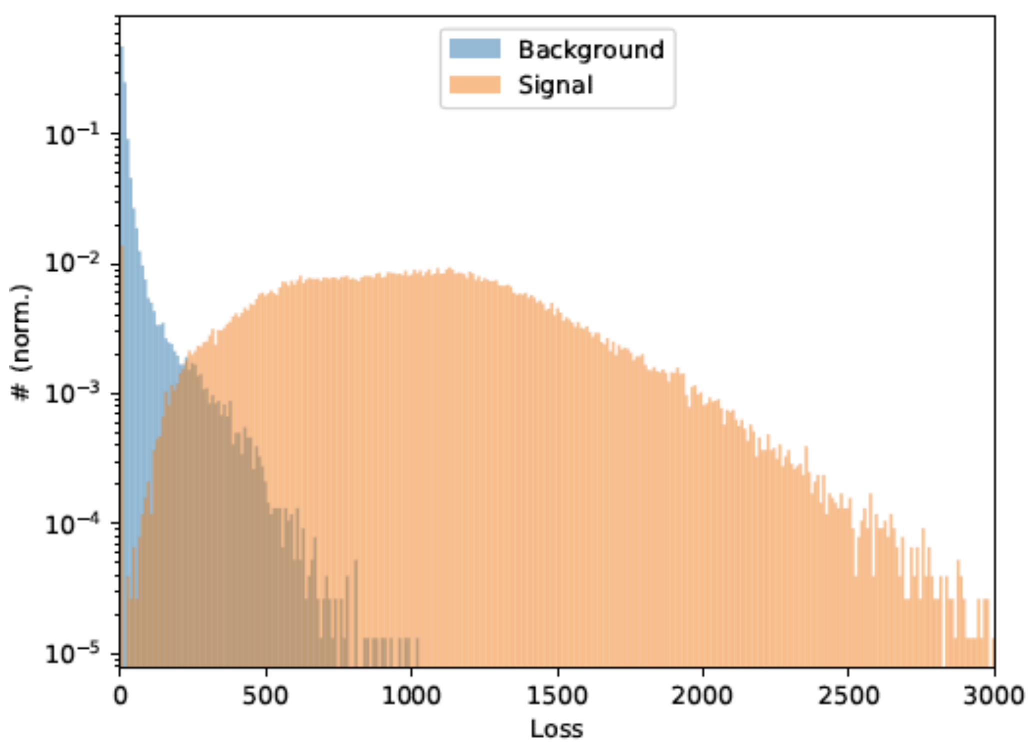

<u>Results for magnetic monopoles</u>

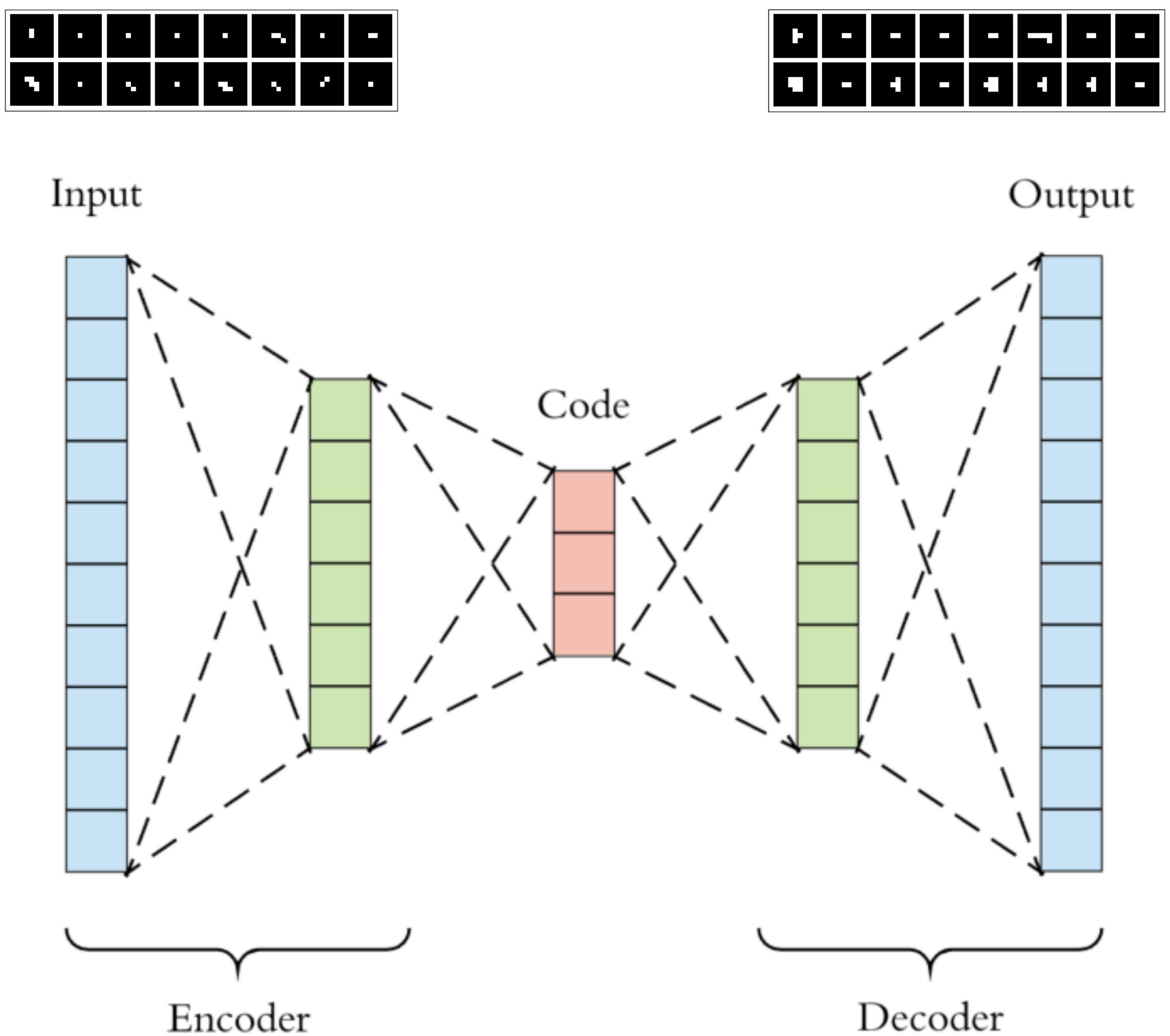
| Input set | Mass [GeV] | Training [%] | Testing [%] | AUC [%] |
|---|---|---|---|---|
| ClsPrp | 1 - 4 | 99.0 | 99.0 | 99.932 |
| PxlDis | 1 - 4 | 98.8 | 98.8 | 99.873 |
| ClsPrp | 1 | 98.6 | 98.6 | 99.877 |
| PxlDis | 1 | 98.8 | 98.7 | 99.890 |
| ClsPrp | 4 | 99.7 | 99.8 | 99.966 |
| PxlDis | 4 | 99.8 | 99.7 | 99.972 |

# ANOMALY DETECTION WITH AUTO-ENCODERS

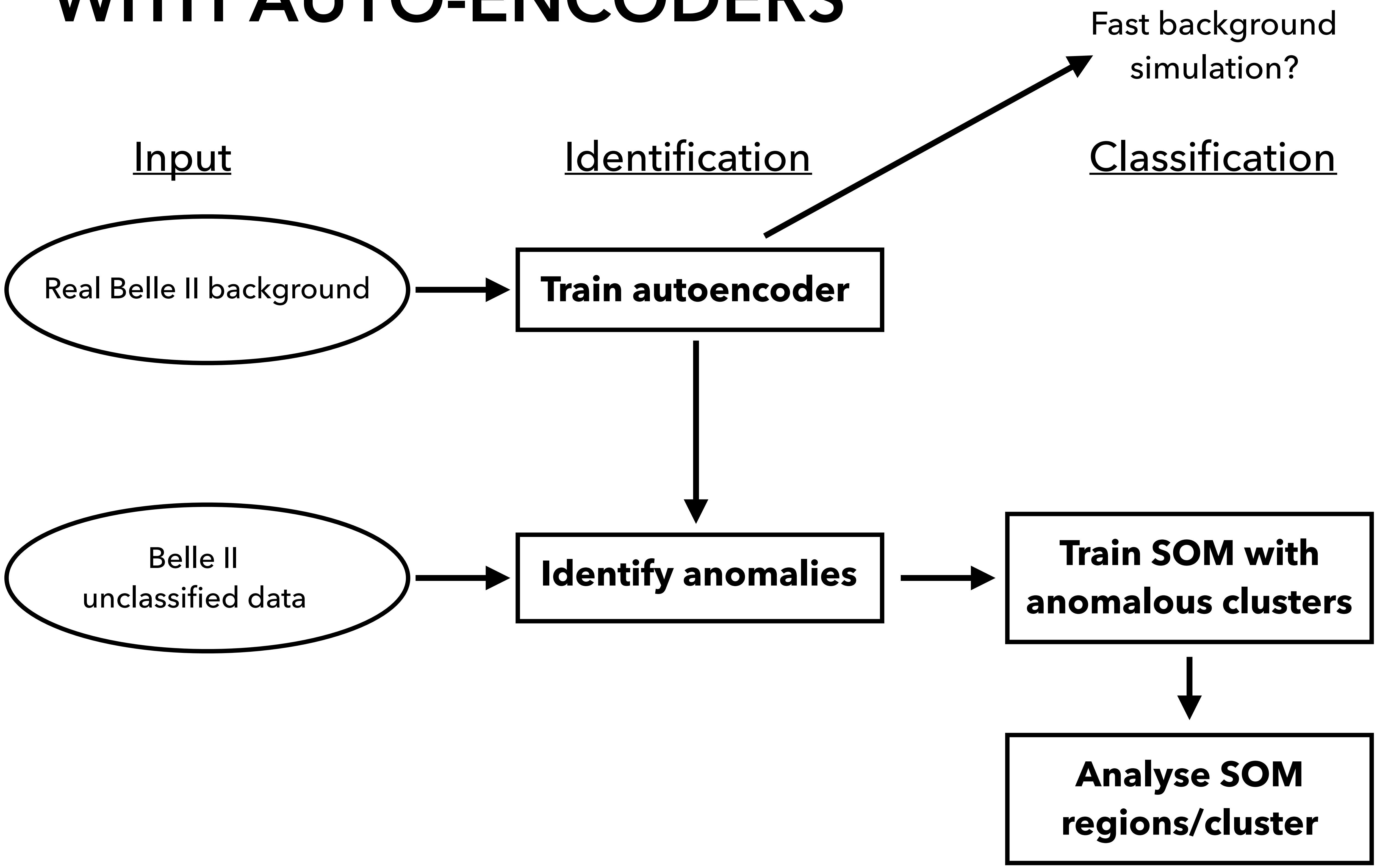
- Learn encoding + decoding of data in unsupervised manner

- Auto-encoder can only reconstruct what it has been trained on -> loss function increases for *unknown* input



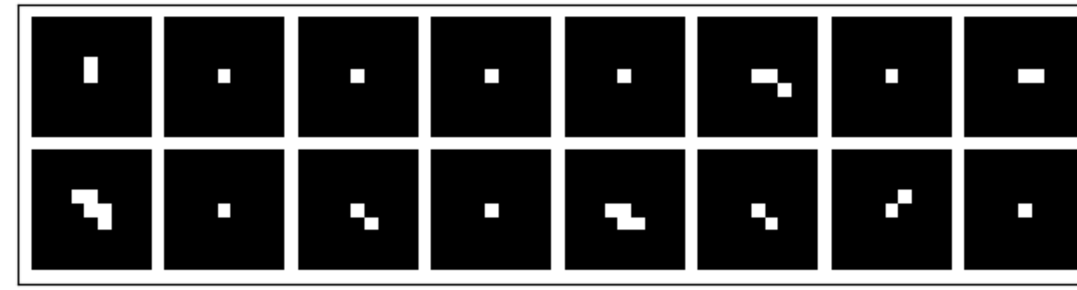



# ANOMALY DETECTION WITH AUTO-ENCODERS



# CHALLENGES

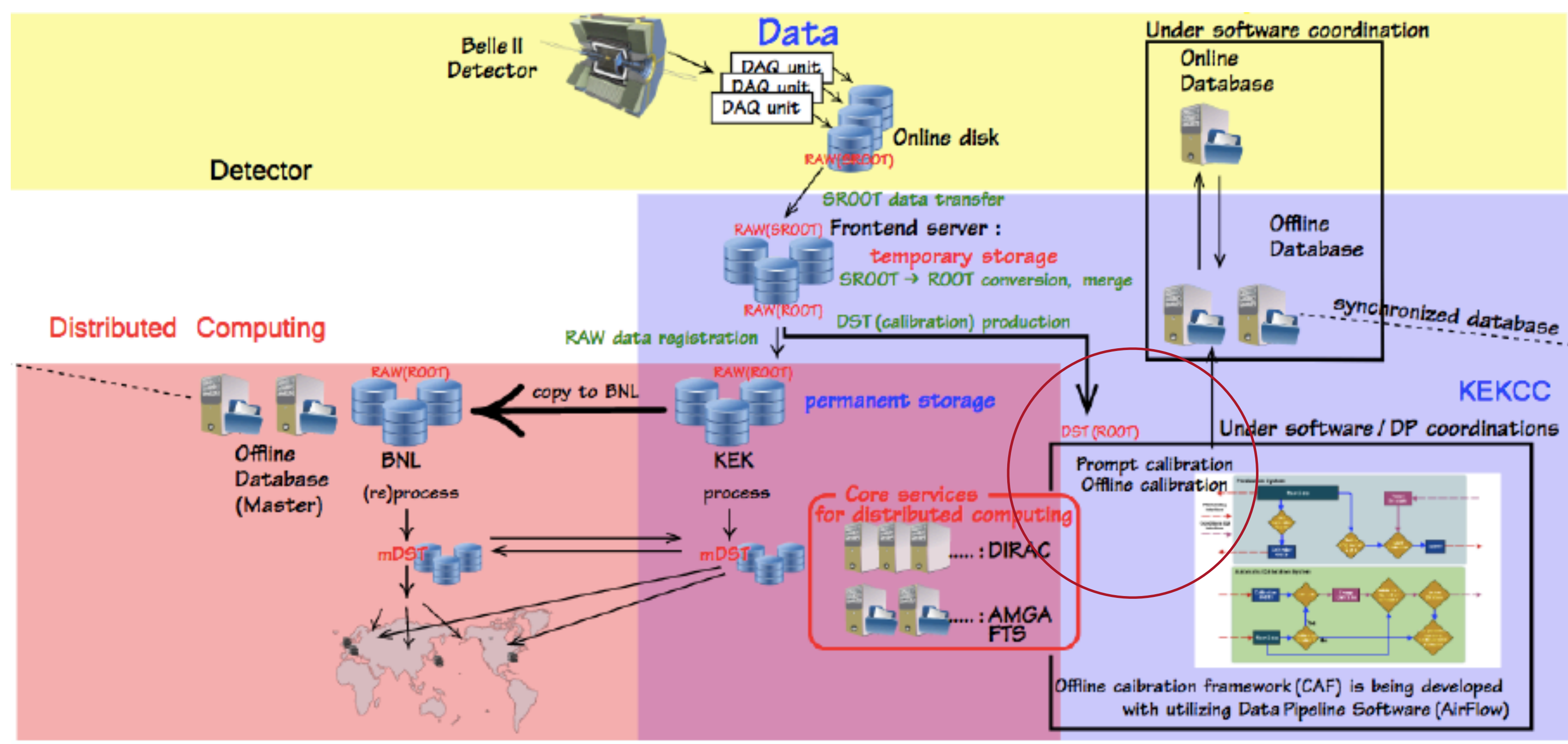
## Analysis-specific



- Sparsity of input matrix / limited amount of information
- Equivocal assignment of clusters to category
- Humongous imbalance of background to signal clusters in data
- Estimation of uncertainties

## Belle II-specific

- No cluster information in mdst/cdst files -> largely confined to Phase II data and data for background studies

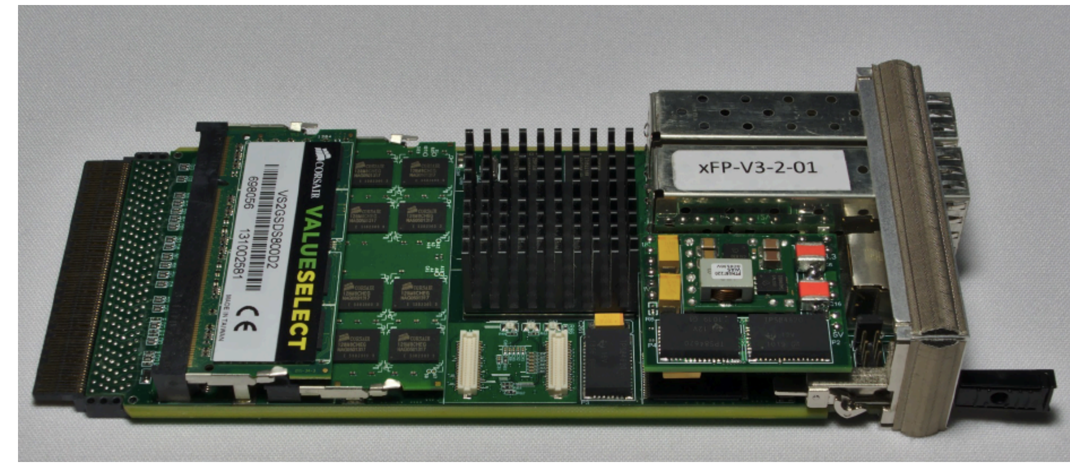



*From J. Bennett's talk at the Belle II summer school July 2019*

# ONLINE ANALYSIS

# NEURAL NETWORKS ON FPGA

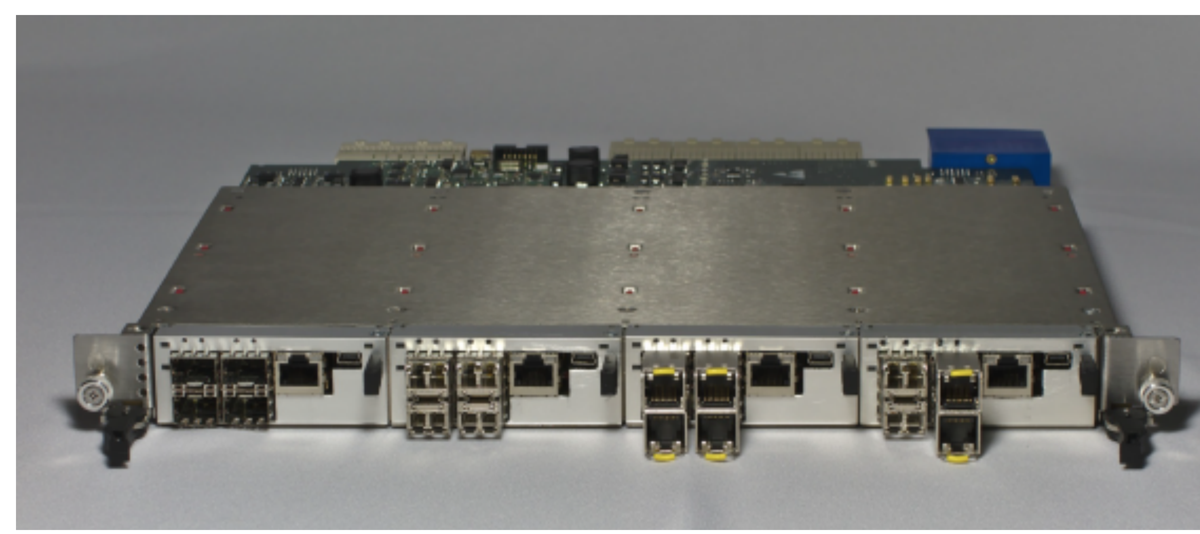
- Field programmable gate arrays (FPGAs) have prominent role in data acquisition for Belle II




- Highly parallel processing architecture make FPGAs ideal candidates for machine learning tasks
- DSP slices can be used for saving resources and speed up computation

*Parallel Computation Using DSP Slices in FPGA*, S. Unnikrishnan et al., Procedia Technology 24 ( 2016 ) 1127–1134

- Communication with DSP slices and adaptation of neural network to FPGA architecture currently tested
- In future, neural networks could be directly integrated into the ONSEN which would also require online cluster finding on DHH or ONSEN (not implemented yet)



# SUMMARY / OUTLOOK

- Identification of particles with high energy deposition at Belle II PXD has seen considerable progress in the last year
- Different data pre-processing and analysis techniques were/are explored and compared
- Focus primarily on model-agnostic unsupervised learning techniques which exploit pattern recognition/matching
- Networks are benchmarked by identifying magnetic monopoles and/or anti-deuterons : accuracies well above 97% are achieved

- Online applications of neural networks running on FPGAs (particularly for the ONSEN) are investigated

# OUTLOOK

- Exploiting hardware acceleration (GPUs, FPGAs) for faster processing and online applications

- Getting permission to analyze Belle II Phase II data and Phase III background data

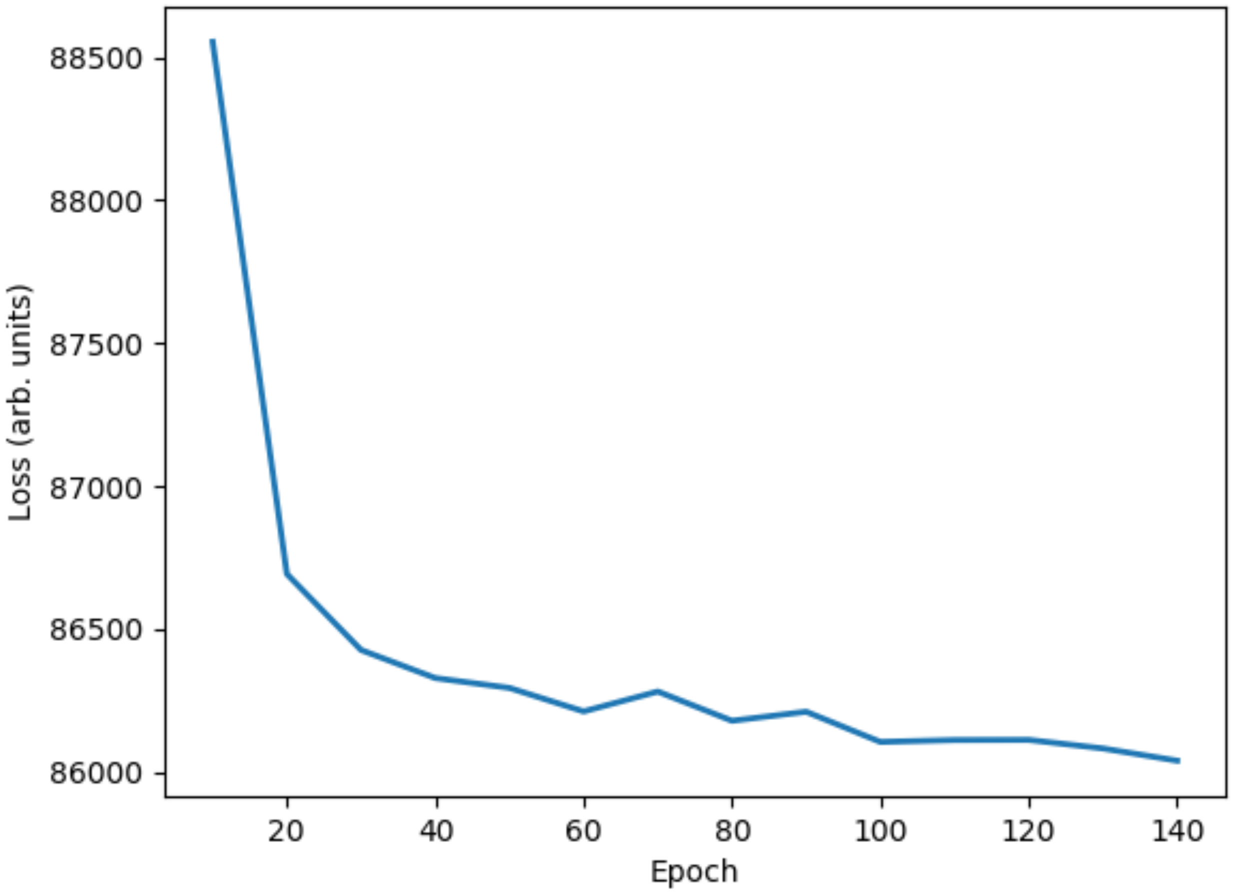
- Close contact to other groups working on ML (in particular theory department and groups from the Technische Hochschule Mittelhessen)

- ML seminar with external experts planned for end of February

# BACK - UP

# NETWORK ARCHITECTURE

- Input: 9x9 pixel matrix around seed pixel
- 3 layers in encoder/decoder with descending/ascending number of nodes per layer
- Bottleneck layer: 16 nodes
- Learning rate: 1E-5
- Batch size: 256
- Loss function: mean square error
- Optimizer: Adam

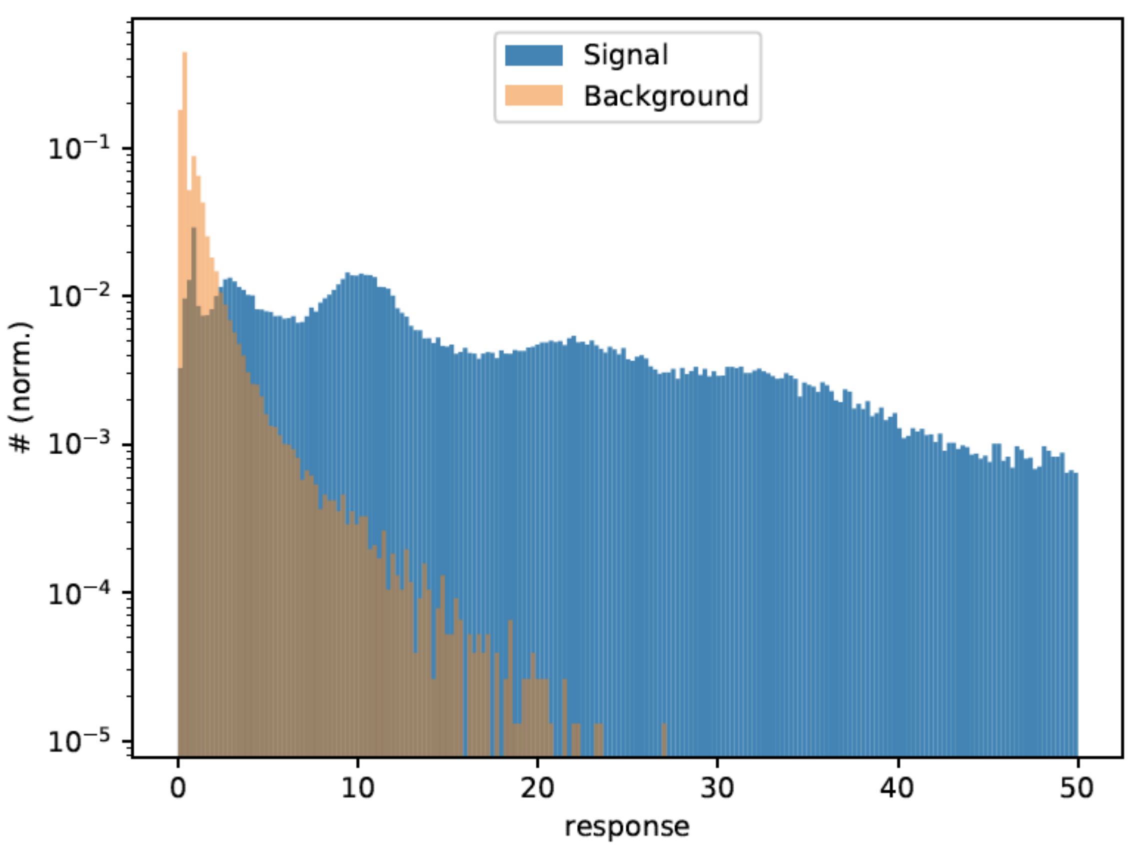


# SIGNAL VS BGD

JUSTUS-LIEBIG-UNIVERSITÄT GIESSEN

- Response of bottleneck layer can yield information about differences in background/signal clusters -> no black box

Node 4




Node 5