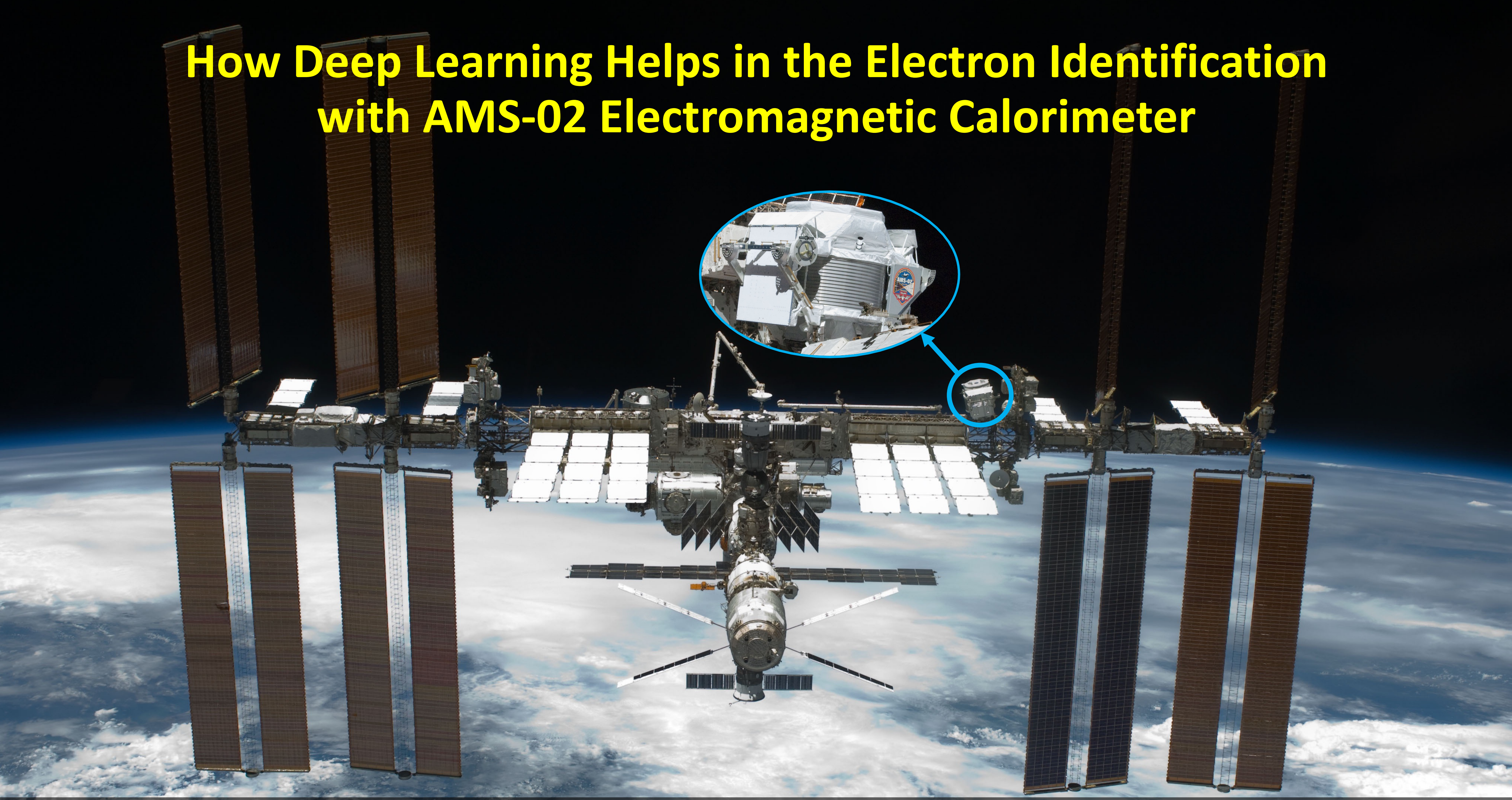
How Deep Learning Helps in the Electron Identification with AMS-02 Electromagnetic Calorimeter

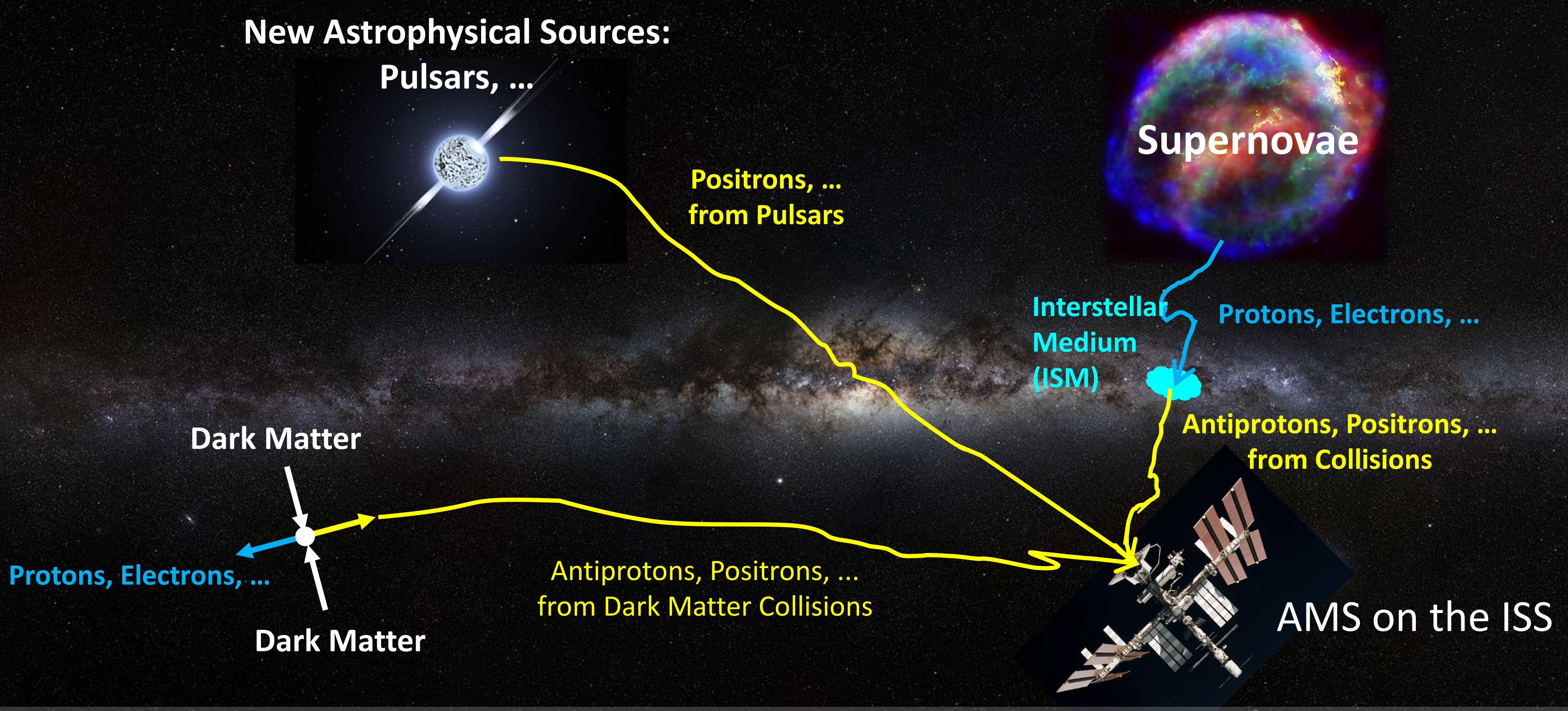


Hsin-Yi Chou

Institute of Physics, Academia Sinica

April 22, 2022

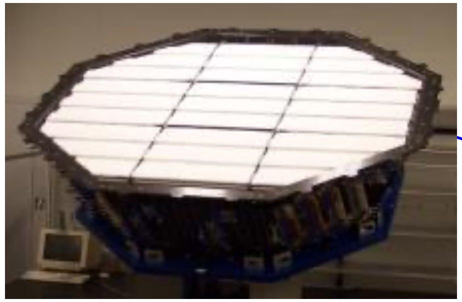
On the Origins of Cosmic Antimatters



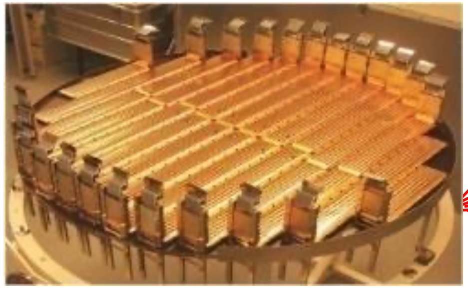
Four elementary particles (Protons Electrons Antiprotons Positrons)
are stable and could travel through the Galaxy.
They carry information of the origin and propagation history of cosmic rays.

AMS (A TeV precision, multipurpose magnetic spectrometer)

Transition Radiation Detector (TRD)
Identify $p^{\pm}/e^{\pm}$

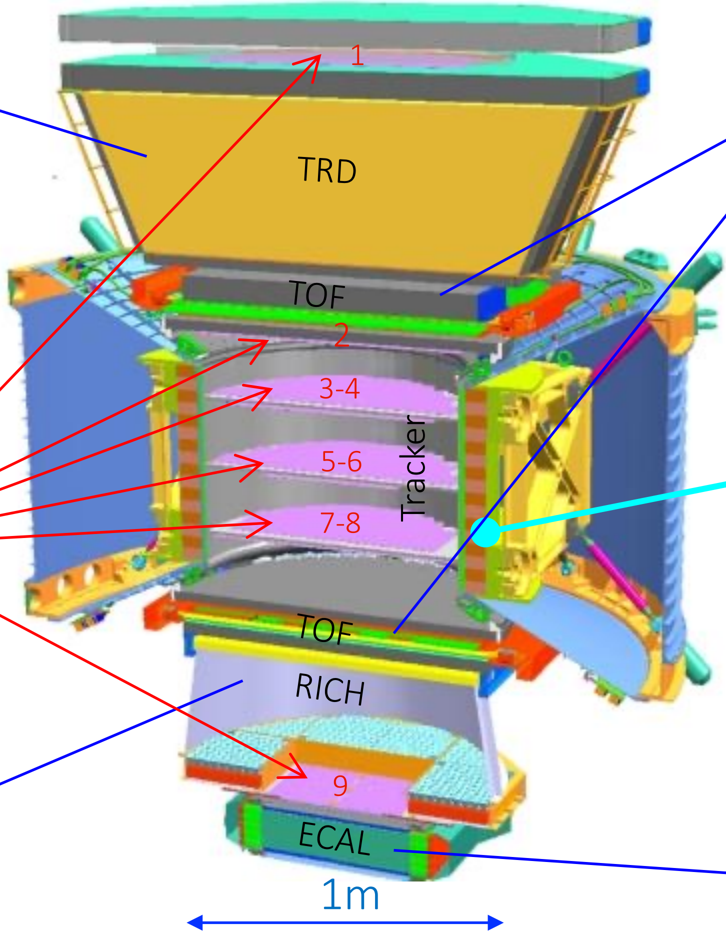


Silicon Tracker (Tracker)
Z, P



Ring Imaging Cherenkov (RICH)
Z, β

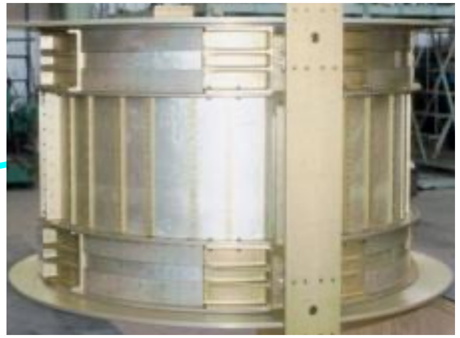
The charge (Z) and energy (E) or rigidity ($R \equiv P/Z$) are measurement independently by several detectors



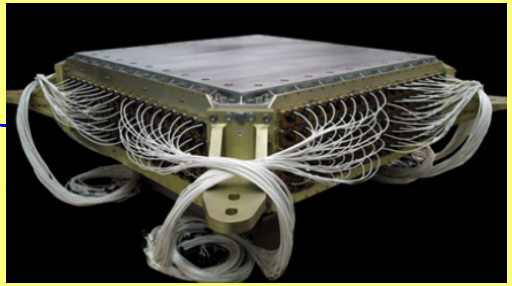
Volume: 5m x 4m x 3m
Weight: 7.5 tons

Time of Flight (TOF)
Z, β

Magnet (~0.14 Tesla)
$\pm Z$



Electromagnetic Calorimeter (ECAL)
E of $e^{\pm}$ and Identify $p^{\pm}/e^{\pm}$



Absolute Charge $|Z|$
Tracker & TOF

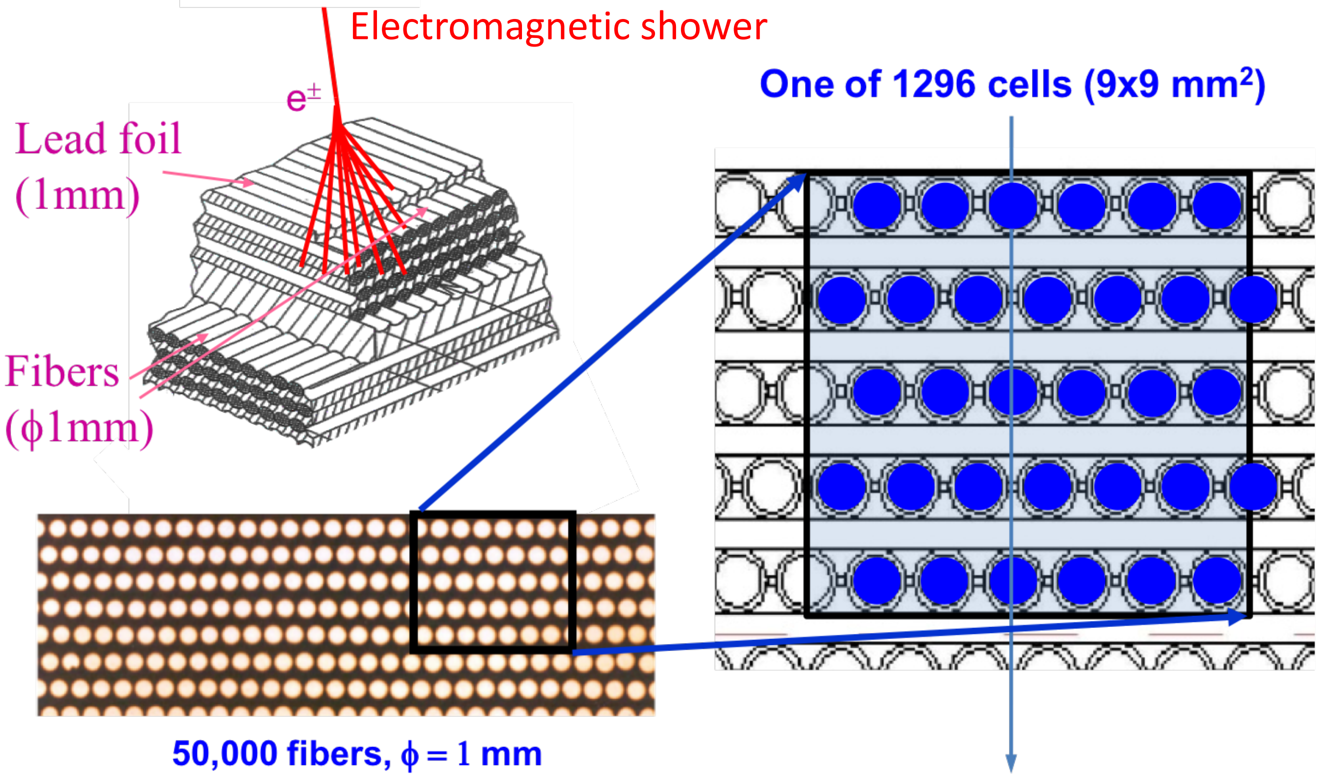
Energy of $e^{\pm}$
ECAL

Charge-Sign
Rigidity R
Tracker with Magnet

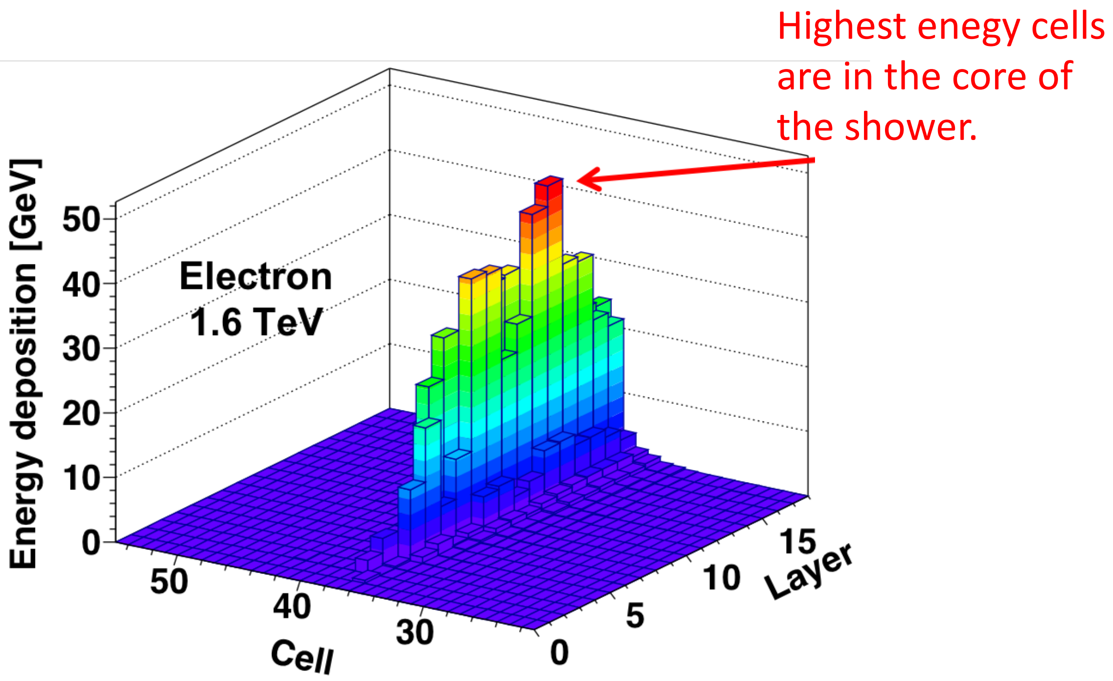
Velocity $\beta = v/c$
TOF & RICH

$p^{\pm}/e^{\pm}$ identification
TRD & ECAL

AMS Electromagnetic Calorimeter (ECAL)



50,000 fibers, ϕ = 1 mm
distributed uniformly inside 1,200 lb of lead

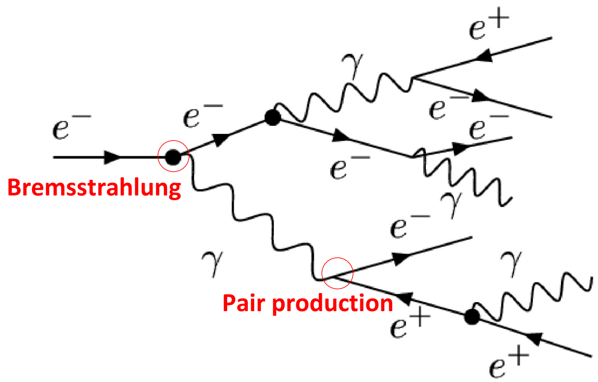


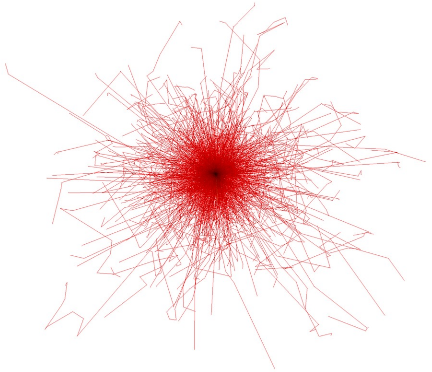
1296 cells = 72 cells/layer x 18 layers

Direction Projection of Layers (10 in Z-Y, 8 in Z-X)

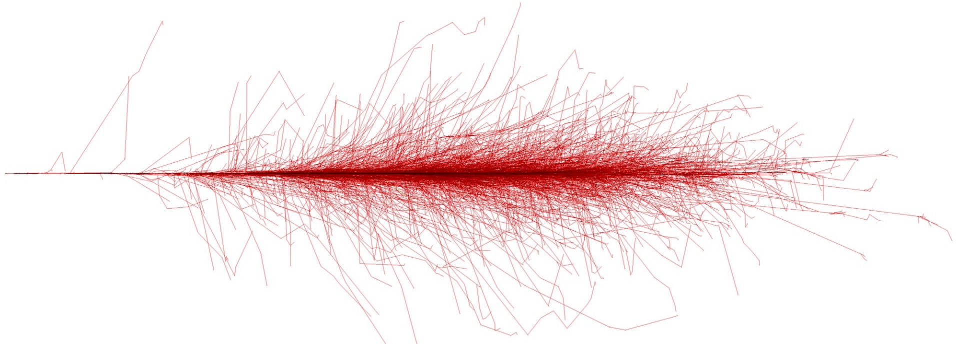
Y	Y	X	X	Y	Y	X	X	Y	Y	X	X	Y	Y	X	X	Y	Y

Electromagnetic (EM) Shower ($e^{\pm}$)



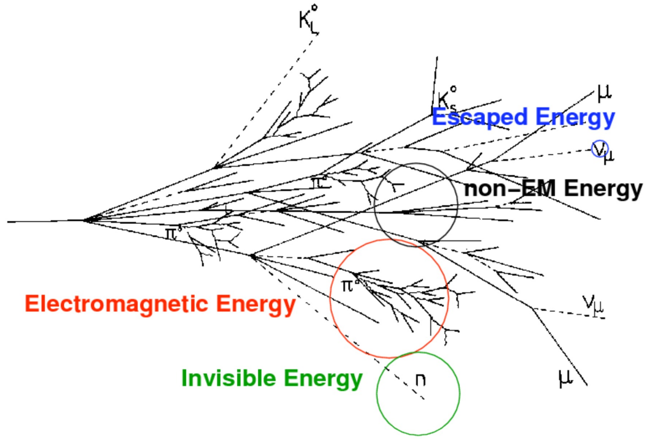


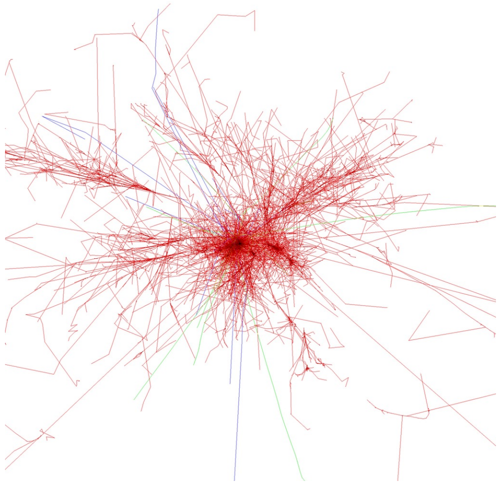
X-Y Projection



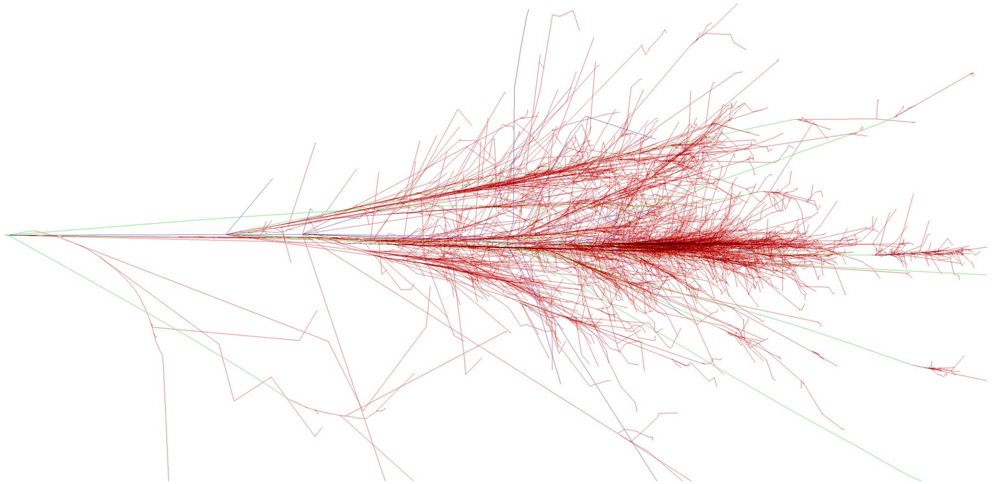
Z-X Projection

Hadron Shower ($p^{\pm}$)





X-Y Projection



Z-X Projection

Two-class Prediction Problem

Consider Two-class prediction problem (binary classification), in which the outcomes are labeled either as positive (P) or negative (N). There are four possible outcomes from a binary classifier.

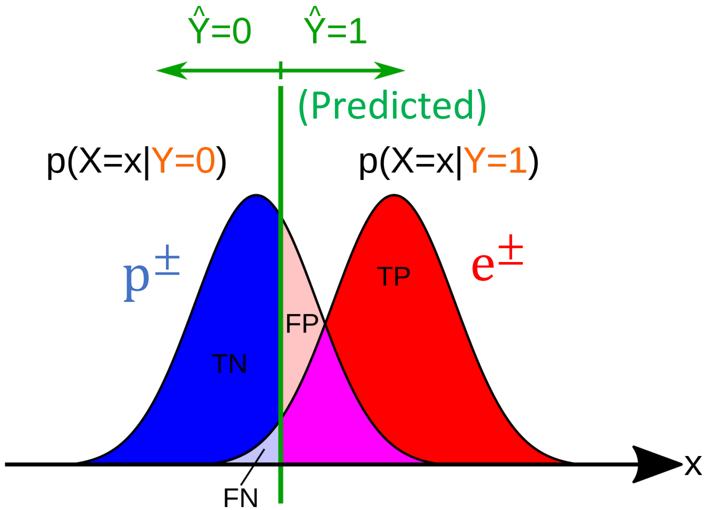
A shower image

Binary Classifier

(Model)

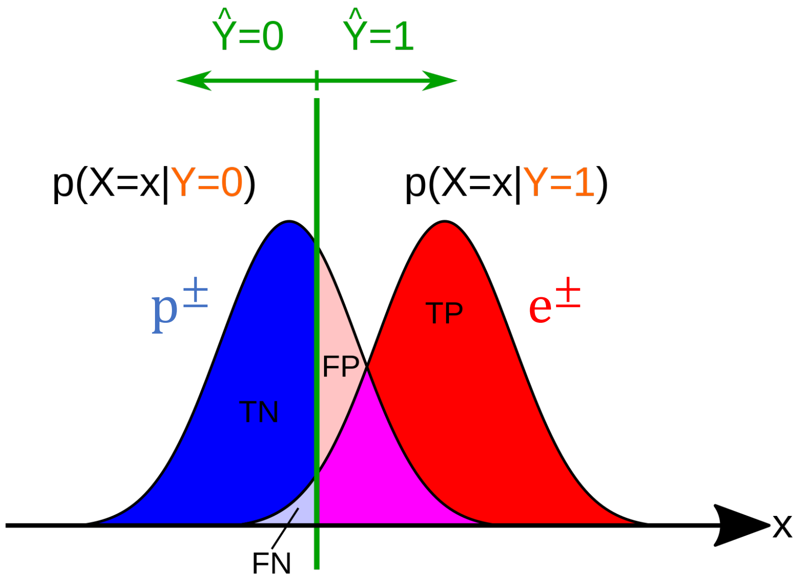
A predicted value (x)
(Is it like an electron or EM shower?)

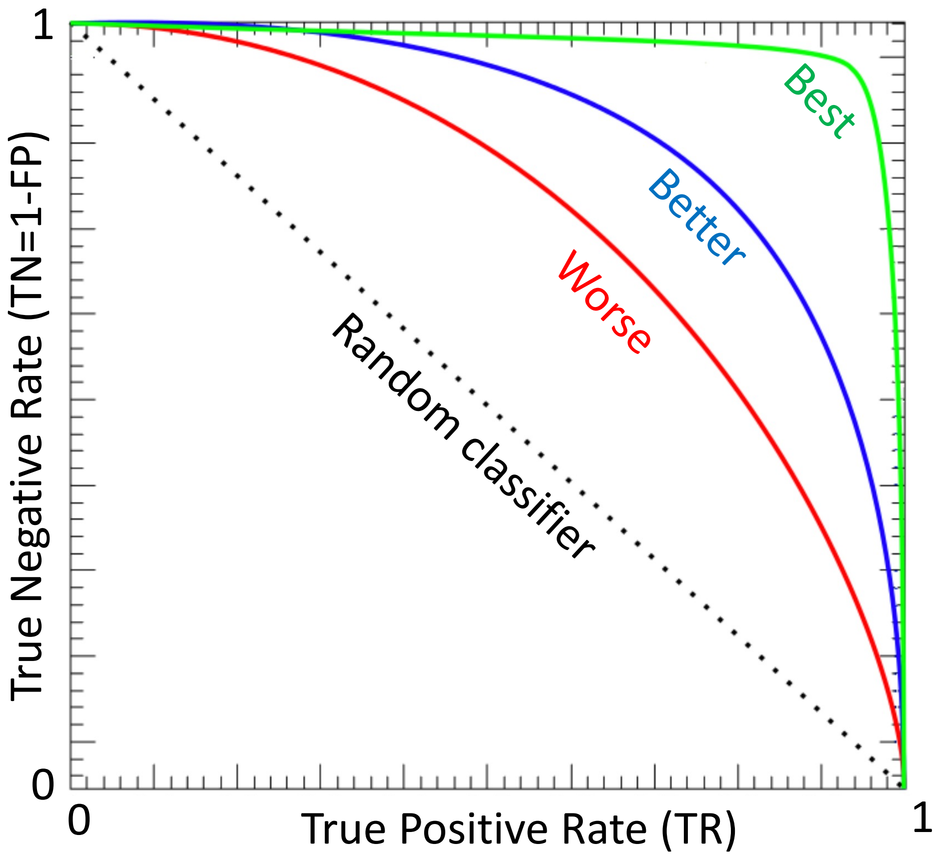
		Predicted condition	
	Total population = P + N	Positive (PP) $e^{\pm}$	Negative (PN) $p^{\pm}$
Actual condition	$e^{\pm}$ Positive (P) Electromagnetic Showers	**True positive** (TP), hit	**False negative** (FN), **type II error**, miss, underestimation
	$p^{\pm}$ Negative (N) Hadron Shower	**False positive** (FP), **type I error**, false alarm, overestimation	**True negative** (TN), correct rejection



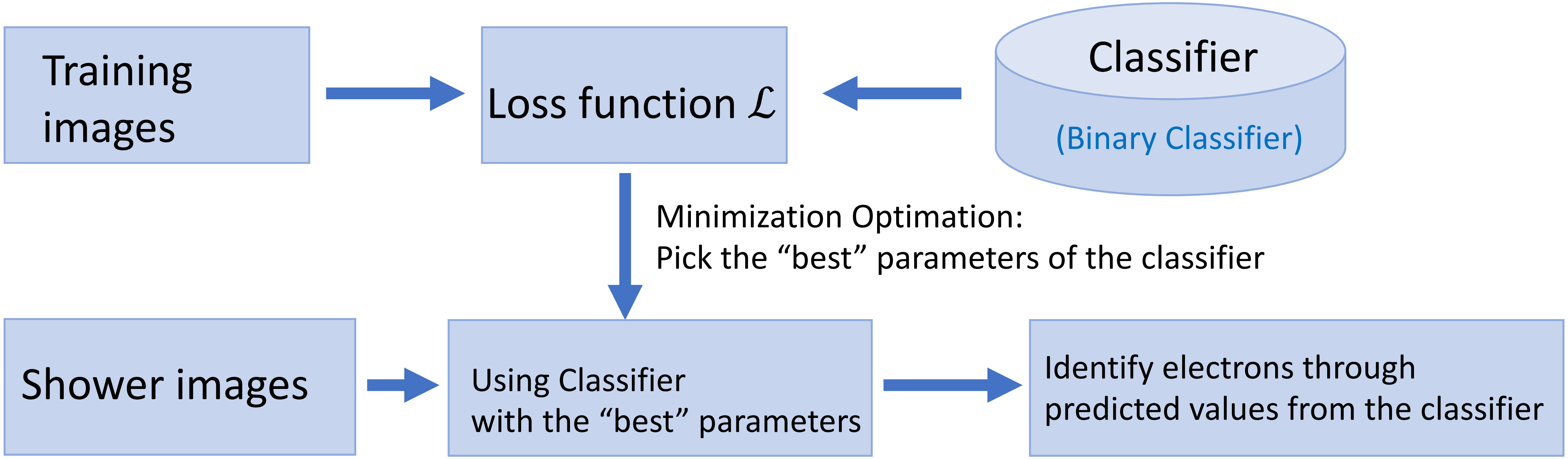
Receiver Operating Characteristic (ROC) Curves

- ROC curves are a good way to illustrate the performance of given classifier
- Shows the negative rejection over the positive rate of the remaning sample
- Best classifier can be identified by the largest area under curve (AUC)





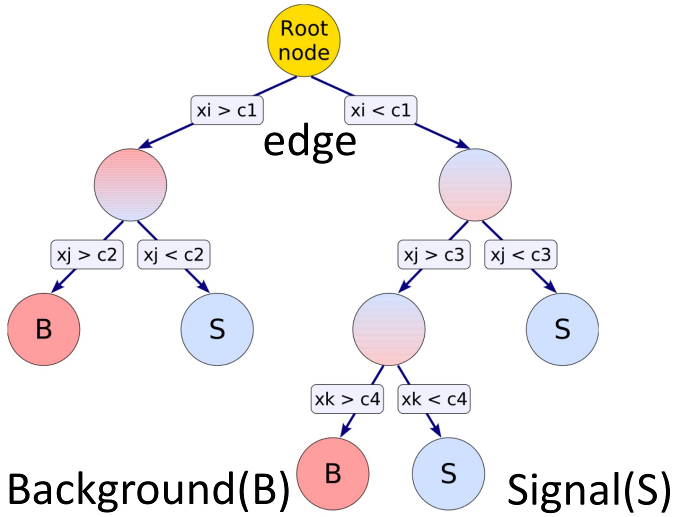
Electron Identification by Machine Learning

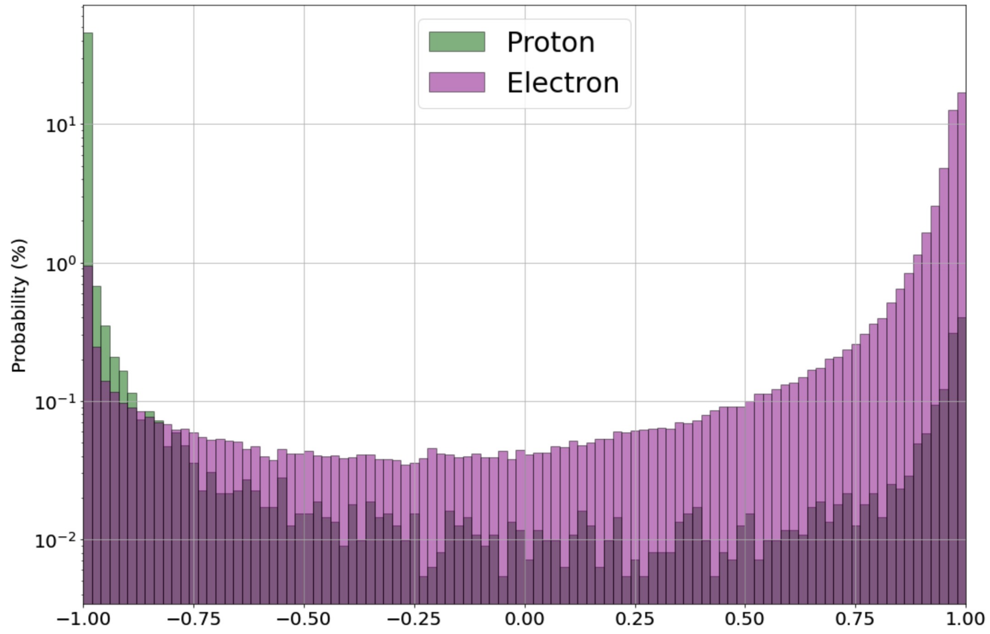


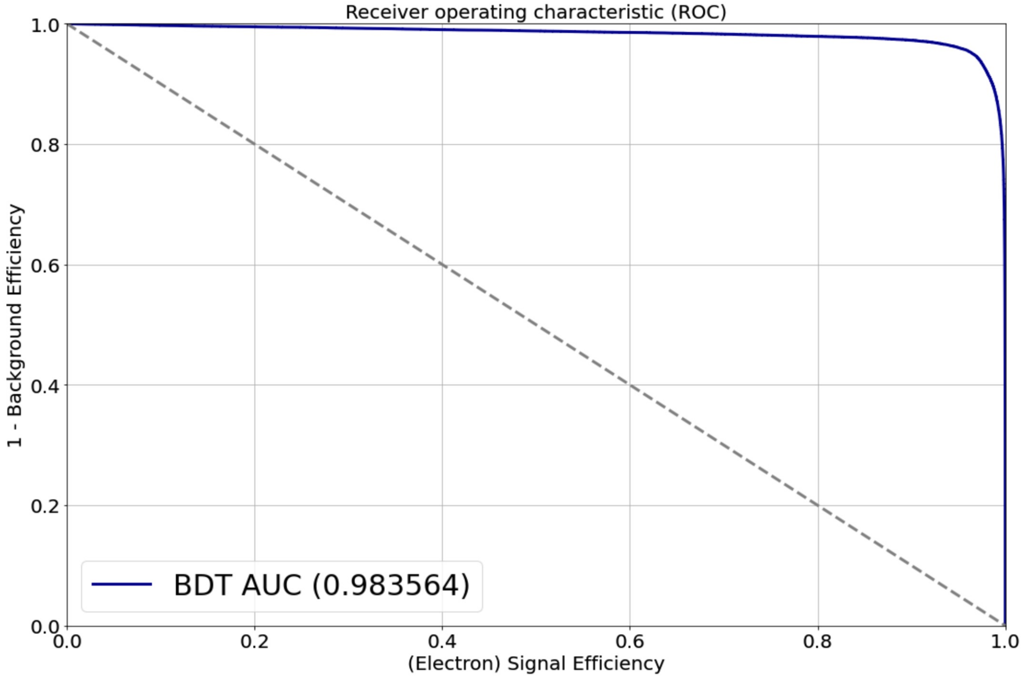
Binary Classifier: Boost Decision Tree (BDT)

Boost decision tree (BDT) is most popular machine learning method in the high energy physics experiment.

An example of a decision is composed of nodes, edges, and the leaves. The nodes are related to different features of the input, and the edges suggests the feature assignments. The nodes at the last level of this tree are called leaves, which give the answer to the classification. The final classifier would be the weighted summary of these decision trees.





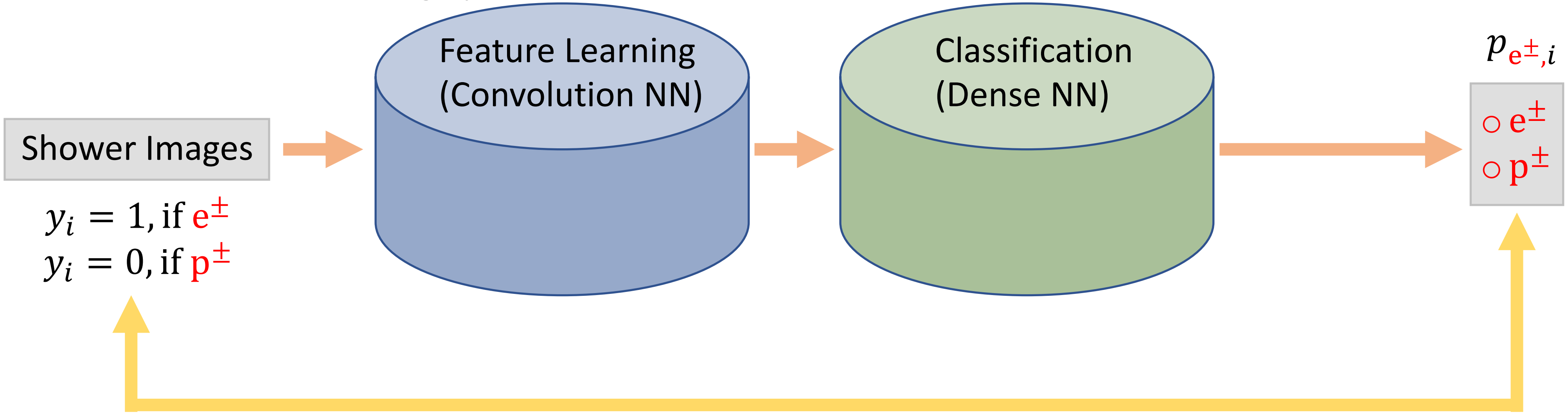


Binary Classifier: Deep Neural Network (DNN)

Learn abstract features and concepts from raw image pixels

Learn high-level features by increasing non-linear relations between features

$p_{\mathrm{e}^\pm,i} + p_{\mathrm{p}^\pm,i} = 1$



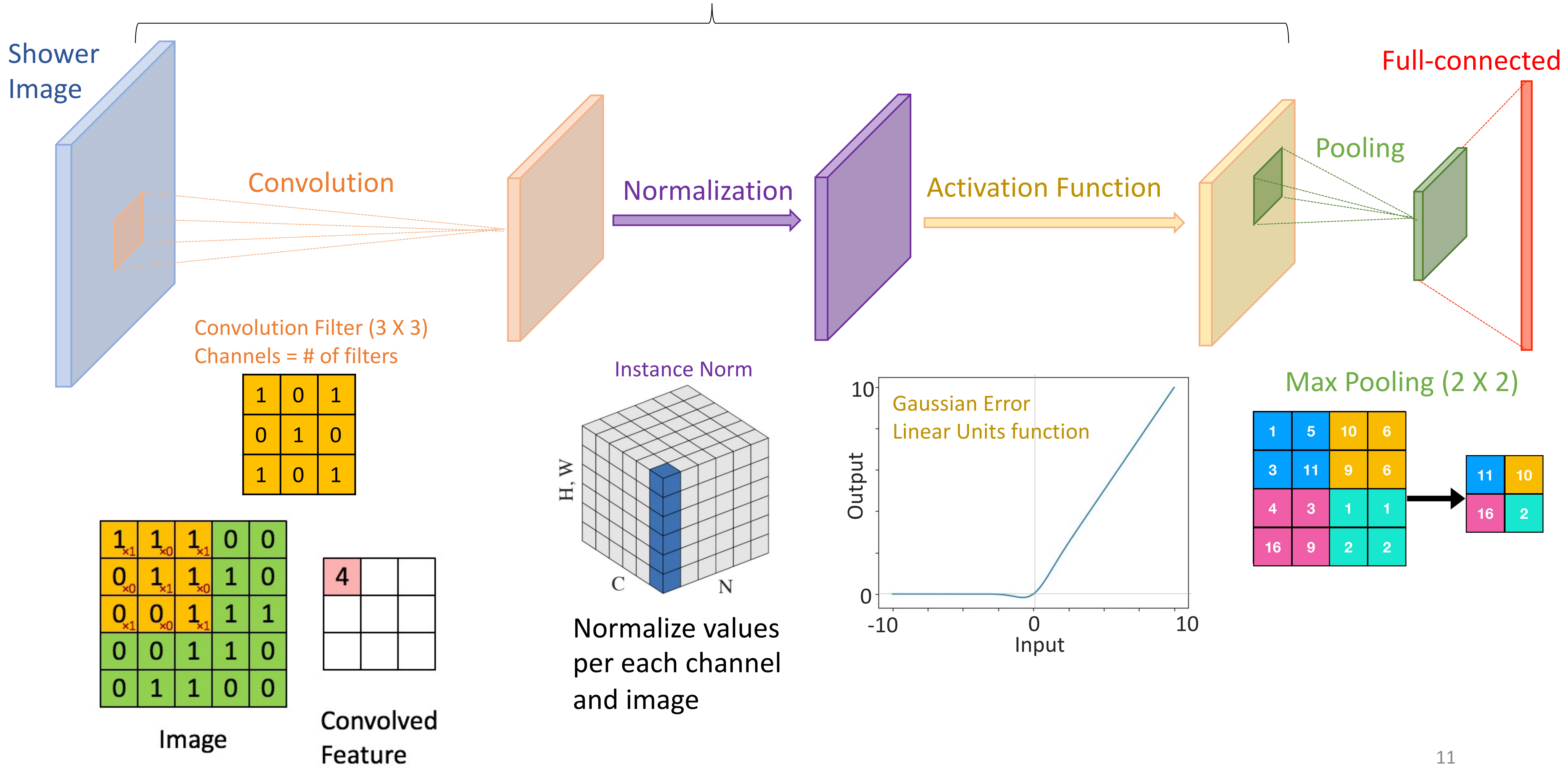
$y_i = 1$, if $\mathrm{e}^\pm$
$y_i = 0$, if $\mathrm{p}^\pm$

Loss function: Binary Cross Entropy

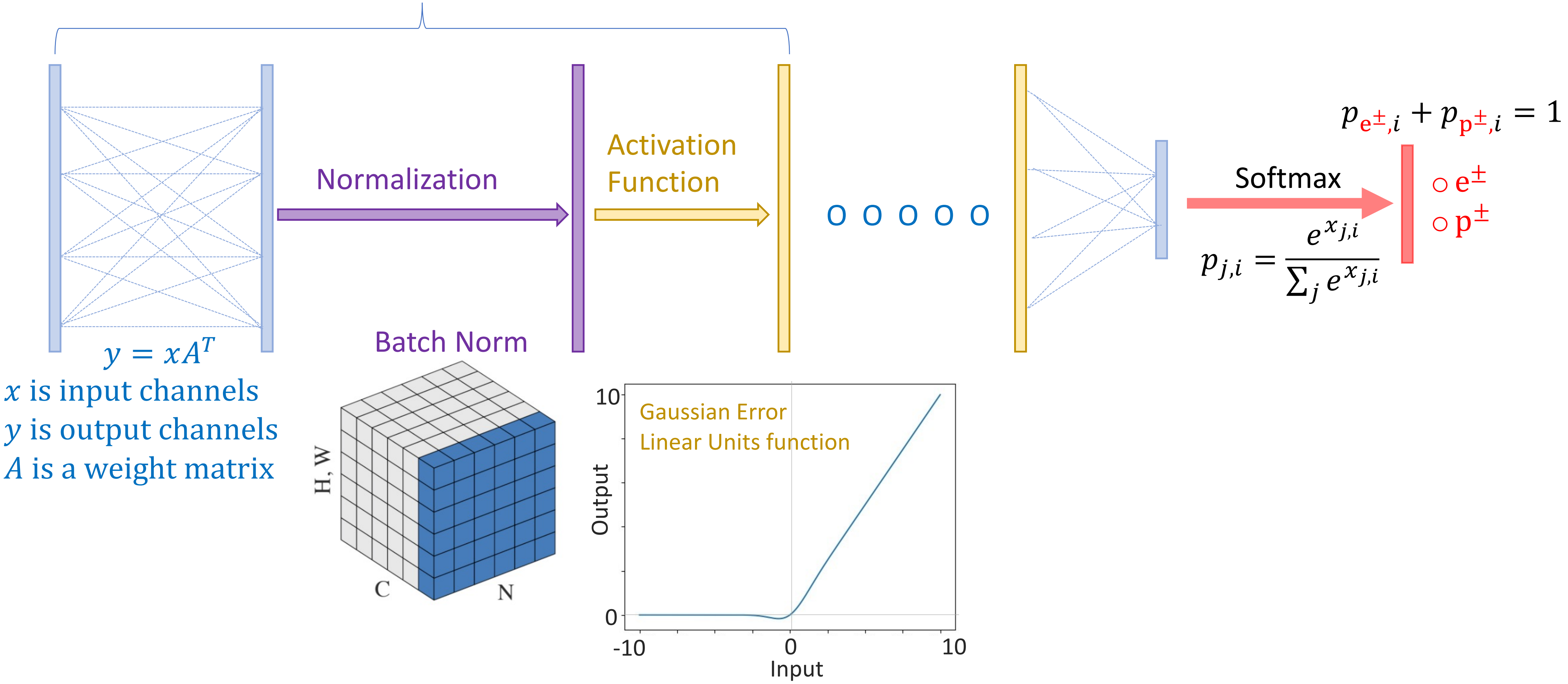
$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \cdot \ln\left(p_{\mathrm{e}^\pm,i}\right) + (1 - y_i) \cdot \ln\left(p_{\mathrm{p}^\pm,i}\right)\right]$$

Binary cross entropy compares each of the predicted probabilities to actual class output which can be either 0 or 1. It then calculates the score that penalizes the probabilities based on the distance from the expected value. That means how close or far from the actual value.

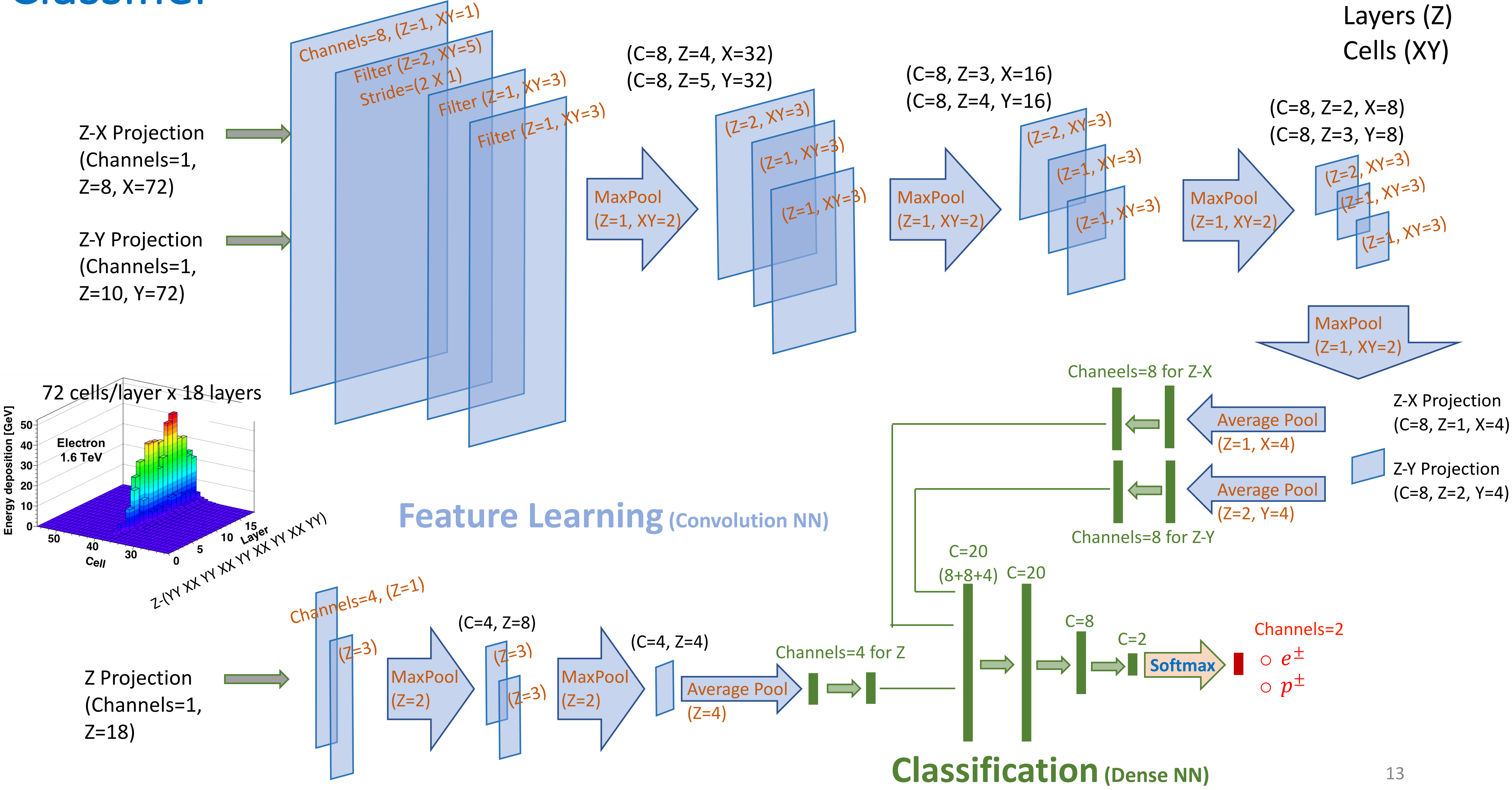
Feature Learning (Convolution Neural Network)



Classification (Dense Neural Network)



Classifier

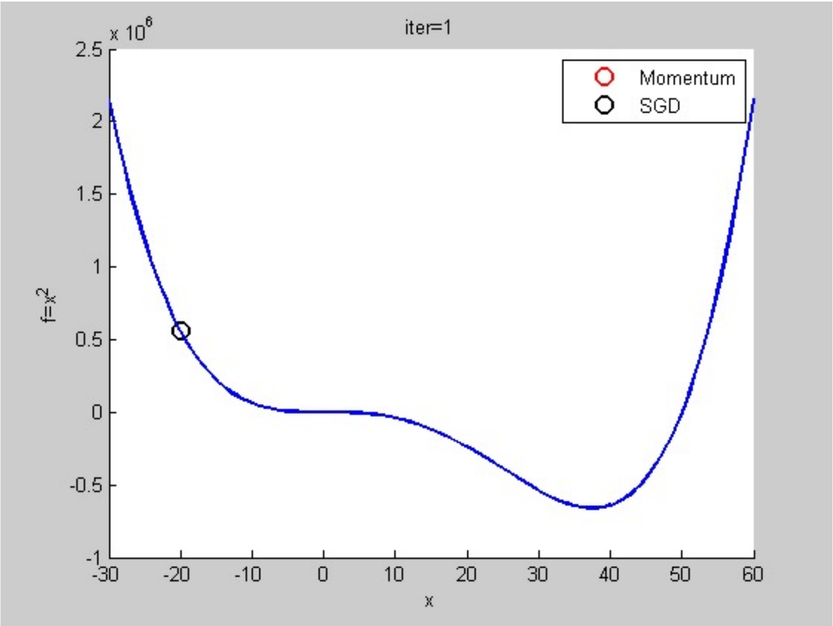


Channels (C)
Layers (Z)
Cells (XY)
Z-X Projection (Channels=1, Z=8, X=72)
Z-Y Projection (Channels=1, Z=10, Y=72)
Channels=8, (Z=1, XY=1)
Filter (Z=2, XY=5)
Stride=(2 X 1)
Filter (Z=1, XY=3)
Filter (Z=1, XY=3)
MaxPool (Z=1, XY=2)
(C=8, Z=4, X=32)
(C=8, Z=5, Y=32)
(Z=2, XY=3)
(Z=1, XY=3)
(Z=1, XY=3)
MaxPool (Z=1, XY=2)
(C=8, Z=3, X=16)
(C=8, Z=4, Y=16)
(Z=2, XY=3)
(Z=1, XY=3)
(Z=1, XY=3)
MaxPool (Z=1, XY=2)
(C=8, Z=2, X=8)
(C=8, Z=3, Y=8)
(Z=2, XY=3)
(Z=1, XY=3)
(Z=1, XY=3)
MaxPool (Z=1, XY=2)
72 cells/layer x 18 layers
Energy deposition [GeV]
Electron 1.6 TeV
Cell
Layer
Z-(YY XX YY XX YY XX YY XX YY)
Feature Learning (Convolution NN)
Chaneels=8 for Z-X
Average Pool (Z=1, X=4)
Average Pool (Z=2, Y=4)
Channels=8 for Z-Y
Z-X Projection (C=8, Z=1, X=4)
Z-Y Projection (C=8, Z=2, Y=4)
Z Projection (Channels=1, Z=18)
Channels=4, (Z=1)
(Z=3)
MaxPool (Z=2)
(C=4, Z=8)
(Z=3)
(Z=3)
MaxPool (Z=2)
(C=4, Z=4)
Average Pool (Z=4)
Channels=4 for Z
C=20 (8+8+4)
C=20
C=8
C=2
Softmax
Channels=2
○ $e^{\pm}$
○ $p^{\pm}$
Classification (Dense NN)

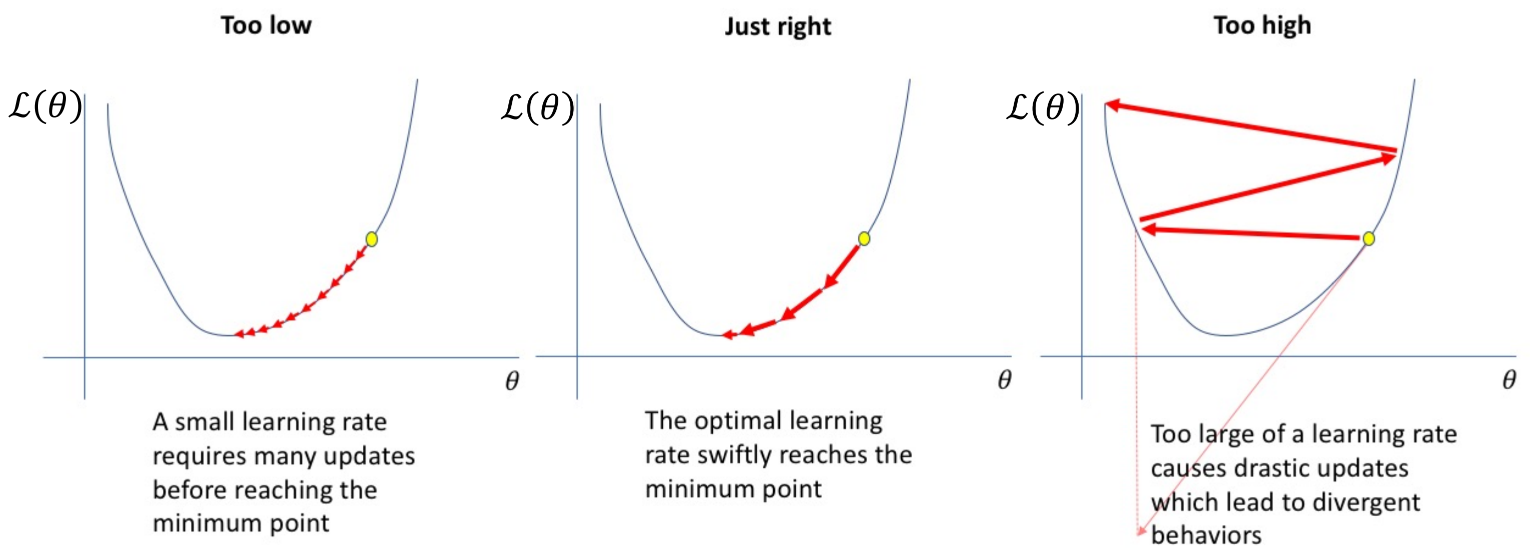
Optimization: Stochastic Gradient Descent (SGD)

$$\theta_{s+1} = \theta_s - (\eta_s \cdot g(\theta_s) + m \cdot \eta_{s-1} \cdot g(\theta_{s-1}))$$

$$g(\theta_s) = \frac{1}{N}\sum_{i=1}^{N} \nabla\mathcal{L}_i(\theta_s)$$



$\mathcal{L}_i(\theta_s)$: loss value of sample i at step s
$g(\theta_s)$: gradient at step s
θ_s: model parameters at step s
η_s: learning rate at step s
m: momentum
N: batch size

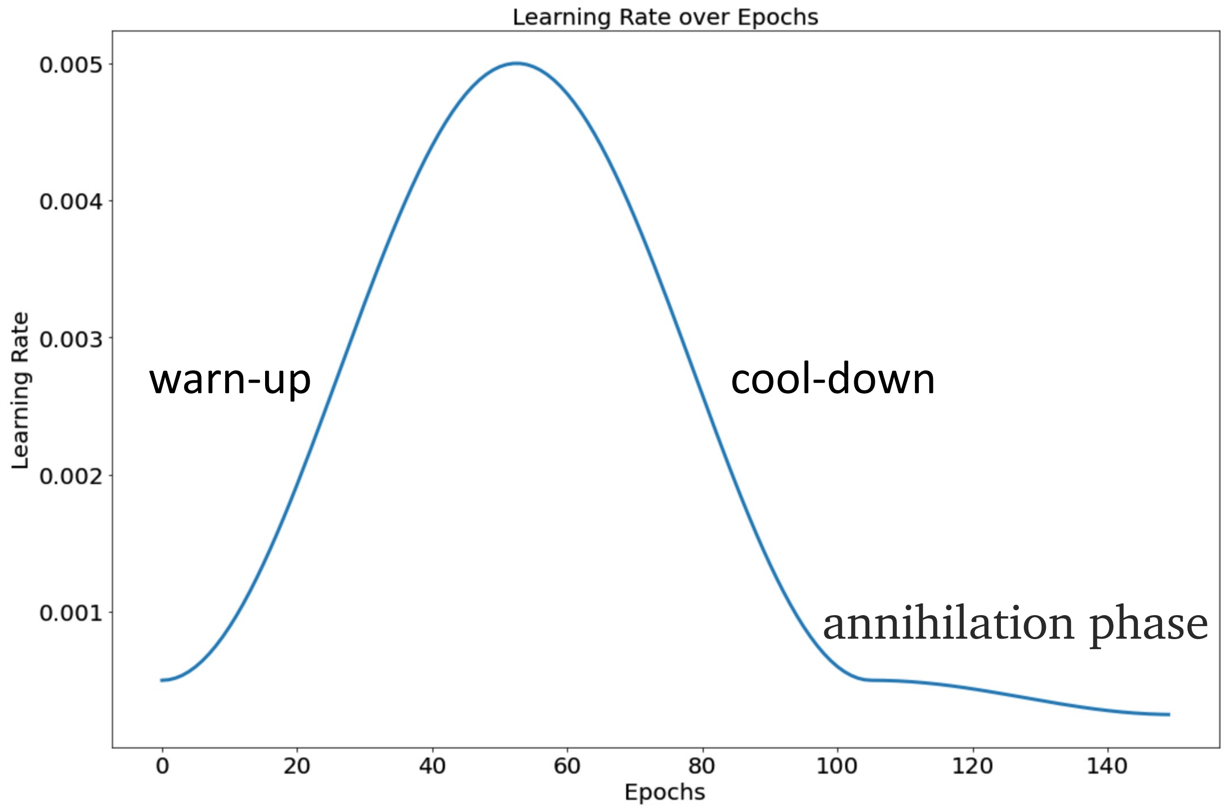


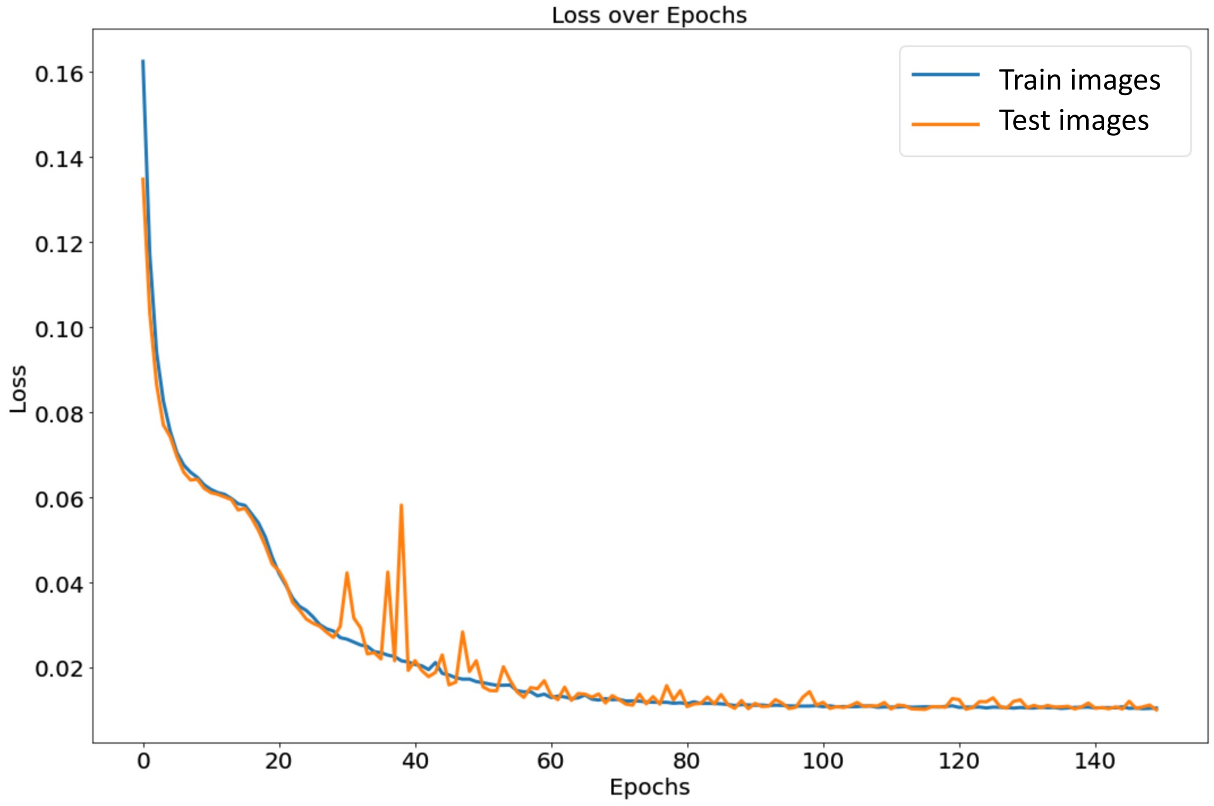
One Cycle Learning Rate Policy

Super-Convergence: Very Fast Training of Neural Networks Using Large Learning Rates (arXiv:1708.07120v3)

- Model parameters are initial to random numbers.
- The gradient is large in beginning iterations, which causes the gradient to be not smooth enough.
- The small learning rate is used in beginning iterations.
- The large learning rate is used to accelerate learning.
- The final small learning rate is used to converge the minimum loss.

$$g(\theta_s) \approx g(\theta_{s-1})$$





for large minibatches (e.g., 8k) the linear scaling rule

Results and Conclusion

- DNN shows 95% signal efficiency at 1% background remaining, and it is better than 40% from BDT.
- Deep Learning is useful in the electron identification with AMS-02 electromagnetic calorimeter.

Deep Neural Network (DNN)