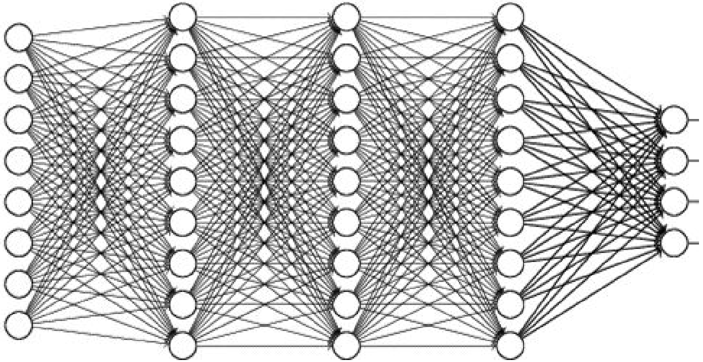
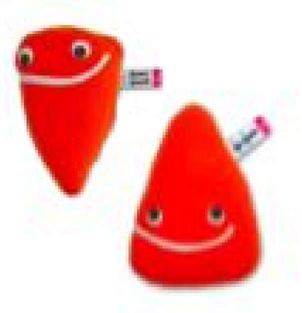
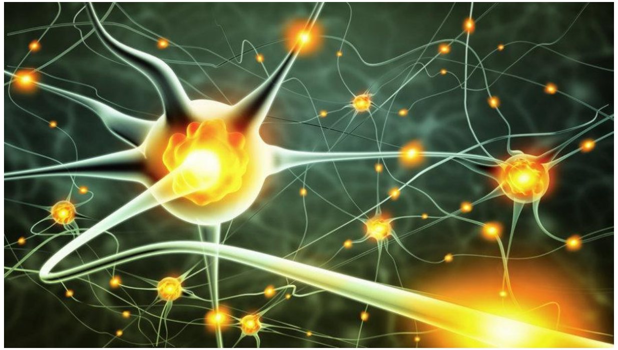




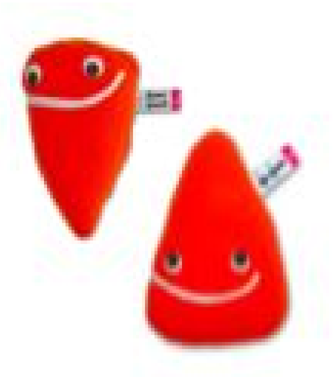
Deep Learning Photon Identification in a Super-Granular Calorimeter

Nikolaus Howe @ Williams College
Maurizio Pierini @ CERN
Jean-Roch Vlimant @ Caltech



Outline

- Introduction to the problem
- What is Machine Learning
- Our work so far
- Limitations and next steps

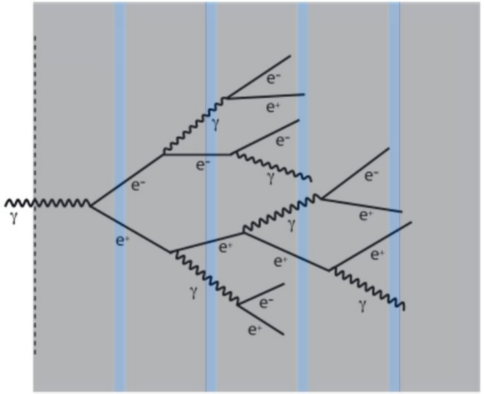


The Problem

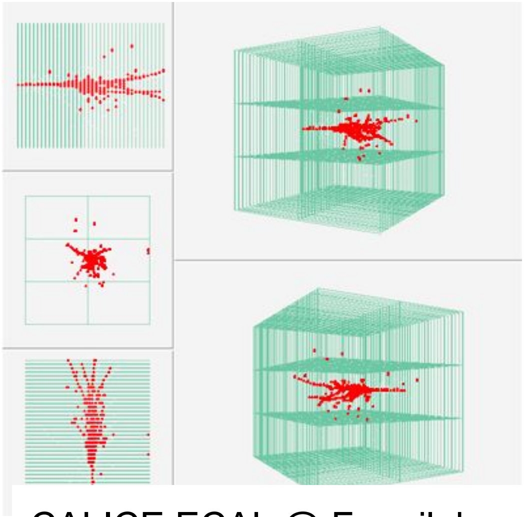
Introduction
- Problem -

Want to take advantage of high granularity of future HEP detectors...

...can contemporary Machine Learning (ML) techniques play a significant role?

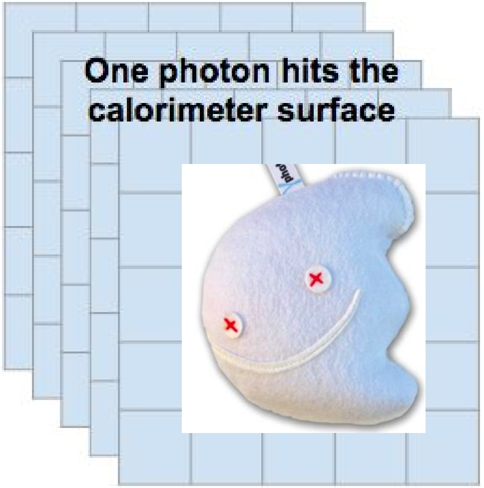


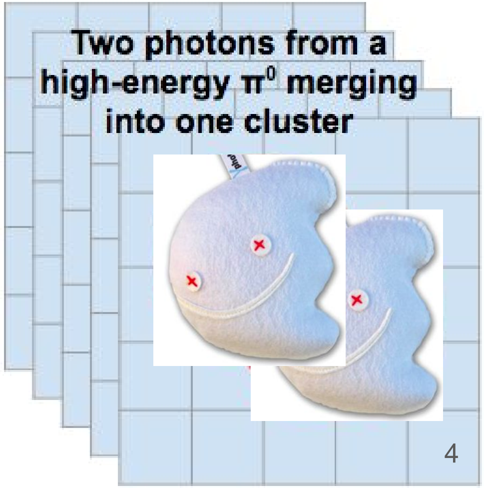
Photon shower



CALICE ECAL @ Fermilab

Define ideal problem:
Photon vs. Pi0
showers





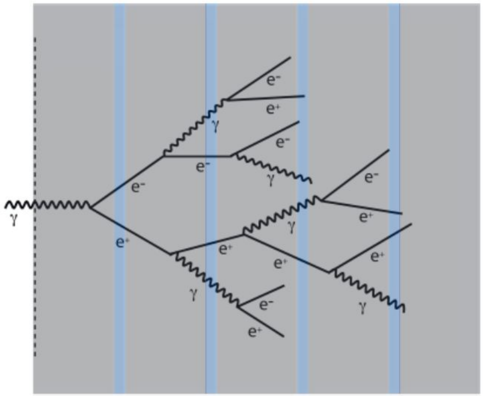
Focus on current cutting edge:

- Super-granular detector geometry
- Deep Neural Networks

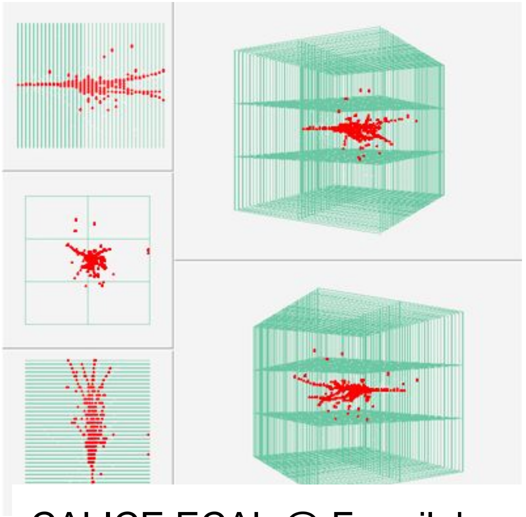
Introduction
- Problem -

Want to take advantage of high granularity of future HEP detectors...

...can contemporary Machine Learning (ML) techniques play a significant role?



Photon shower

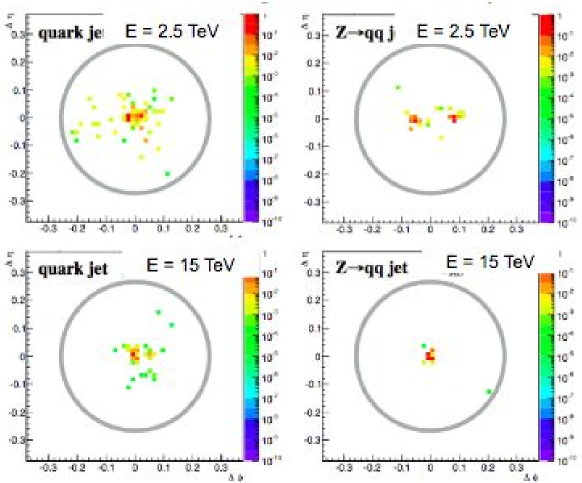


CALICE ECAL @ Fermilab

Define ideal problem:

Photon vs. Pi0

showers



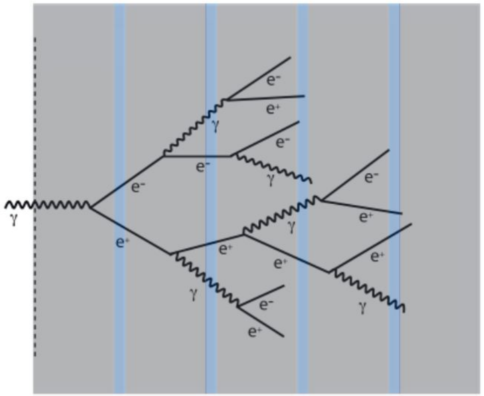
Focus on current cutting edge:

- Super-granular detector geometry
- Deep Neural Networks

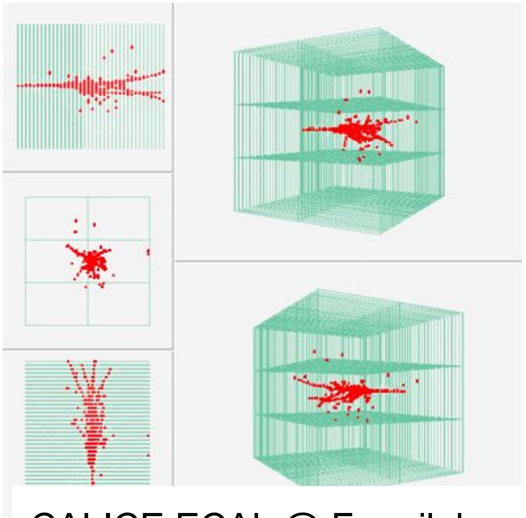
Introduction
- Detector -

Photon vs. **Pi0** showers
in LCD ECAL

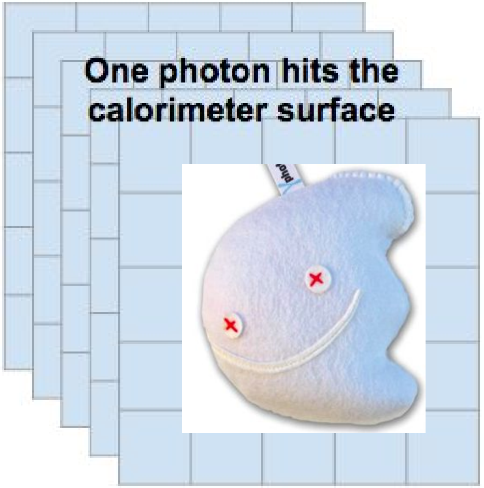
- Easy to simulate samples
- State of the art of the next-generation detector
- The ultimate granular calorimeter in (x,y) as well as z (3D problem)
- Very similar to classic ML problem (MNIST dataset, coming up later!)

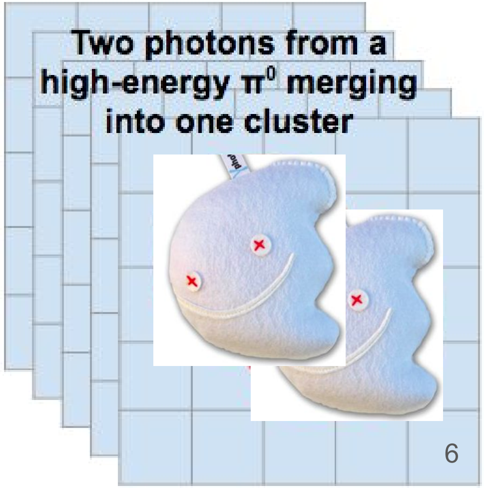


Photon shower



CALICE ECAL @ Fermilab

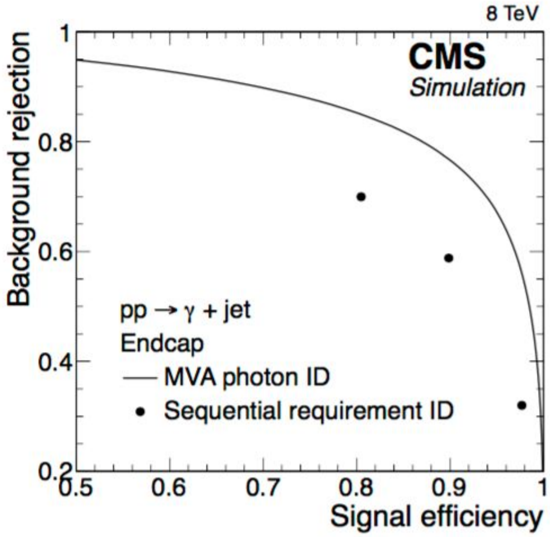




Introduction
- Particle ID -

ML in use in past & current experiments!

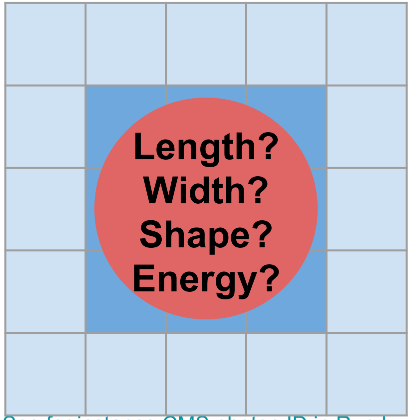
- Neural Networks @ LEP
- Boosted Decision Trees @ BaBar/Belle and the LHC



PID - *supervised classification problem:*

Current procedure:

- Get samples from simulation or detector
- Manually identify discriminating physics features (signal vs. background)
- Train algorithm (cut-based selection/BDT/NN/etc) to maximize separation
- Extract features and apply algorithm to classify new data

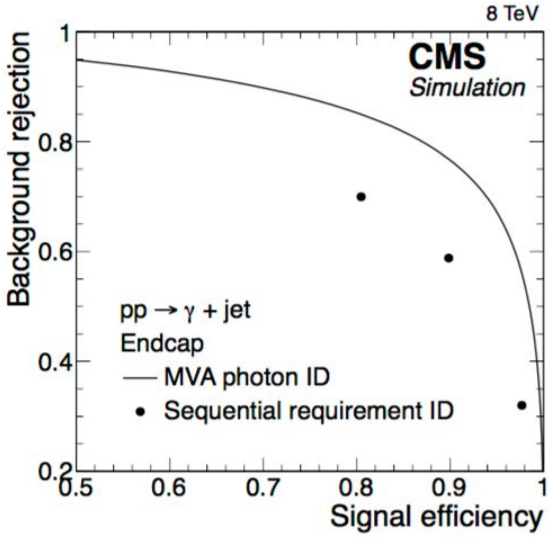


See for instance CMS photon ID in Run I

Introduction
- Particle ID -

ML in use in past & current experiments!

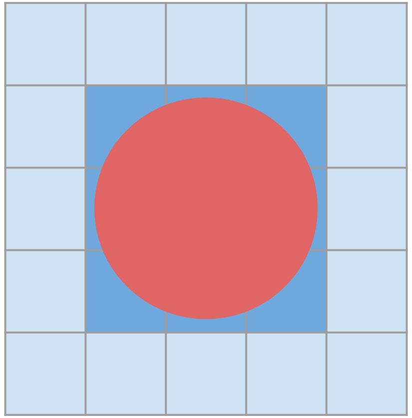
- Neural Networks @ LEP
- Boosted Decision Trees @ BaBar/Belle and the LHC



PID - *supervised classification problem:*

New Approach:

- Get samples from simulation or detector
- No feature identification—use only raw data
- Train DNN to *look at* clusters (imaging problem!) for PID
- Apply DNN to classify new data



Introduction
- Particle ID -

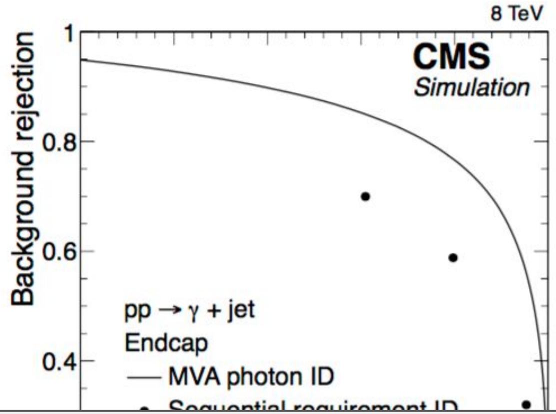
ML in use in past & current experiments!

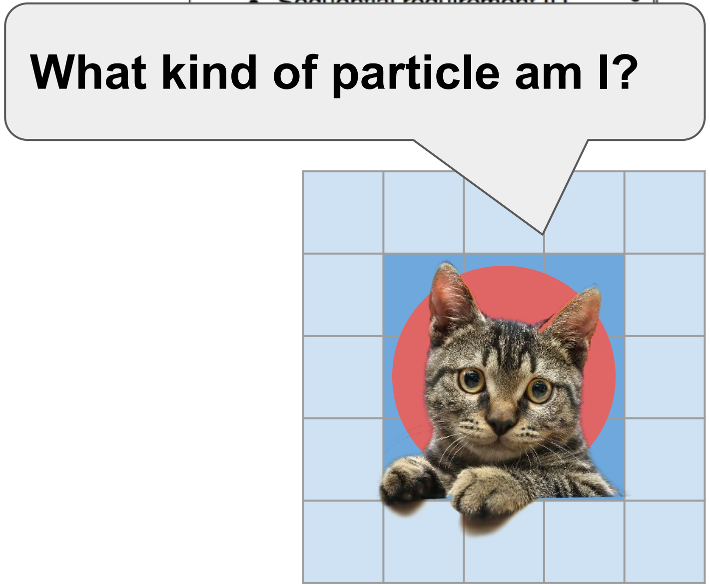
- Neural Networks @ LEP
- Boosted Decision Trees @ BaBar/Belle and the LHC

PID - *supervised classification problem:*

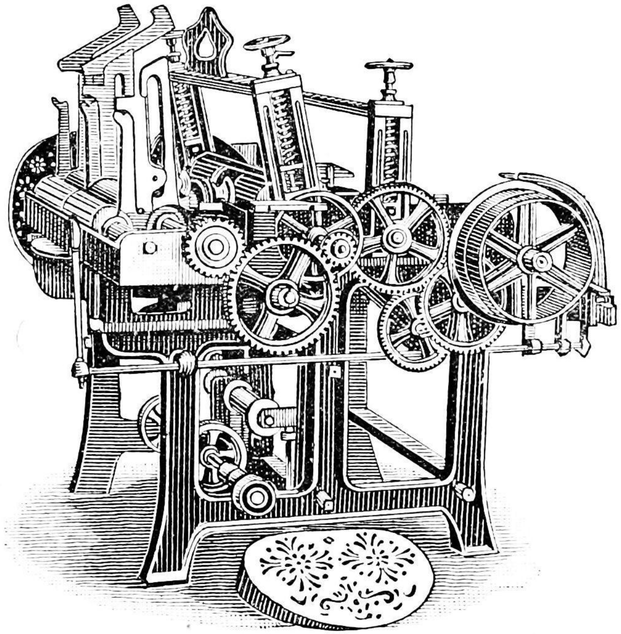
New Approach:

- Get samples from simulation or detector
- No feature identification—use only raw data
- Train DNN to *look at* clusters (imaging problem!) for PID
- Apply DNN to classify new data

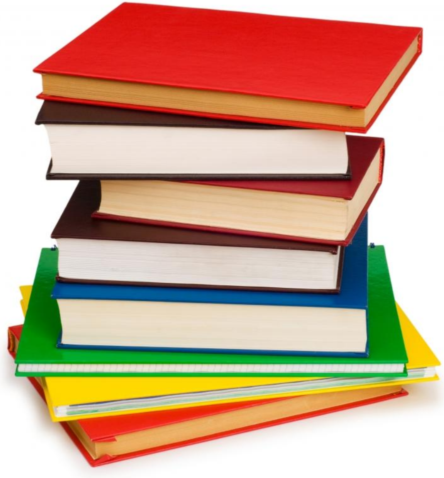




Machine Learning



+



= ?

Machine Learning
- MNIST -

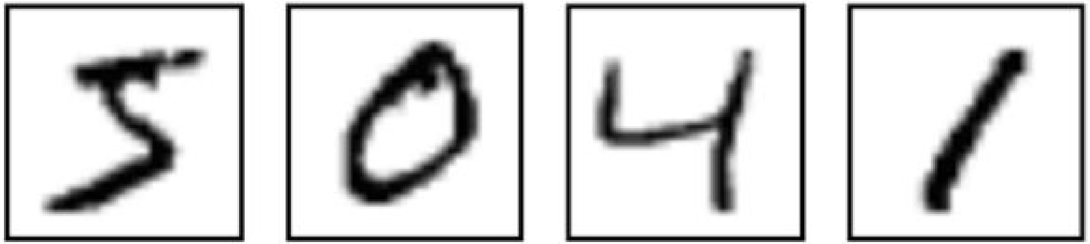
Machine Learning:

"teaching computers to classify data"

Classic Example:

MNIST handwritten digit dataset

70,000 28x28 centered arrays of labeled, handwritten digits



Machine Learning
- Procedure -

Collect tons of labeled data

↓

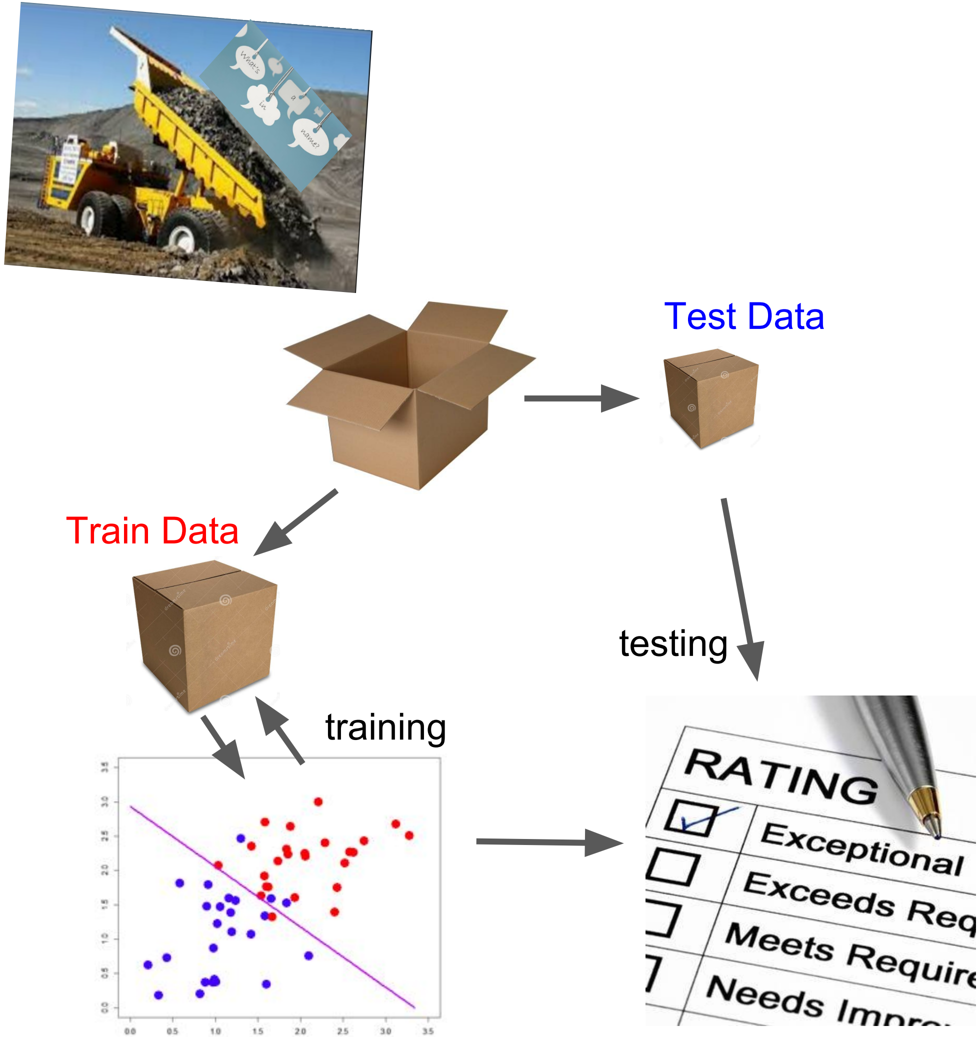
Divide into train and test sets

↓

Use train set to learn to separate classes based on attributes
(train for multiple ***epochs*** until ***loss*** is small)

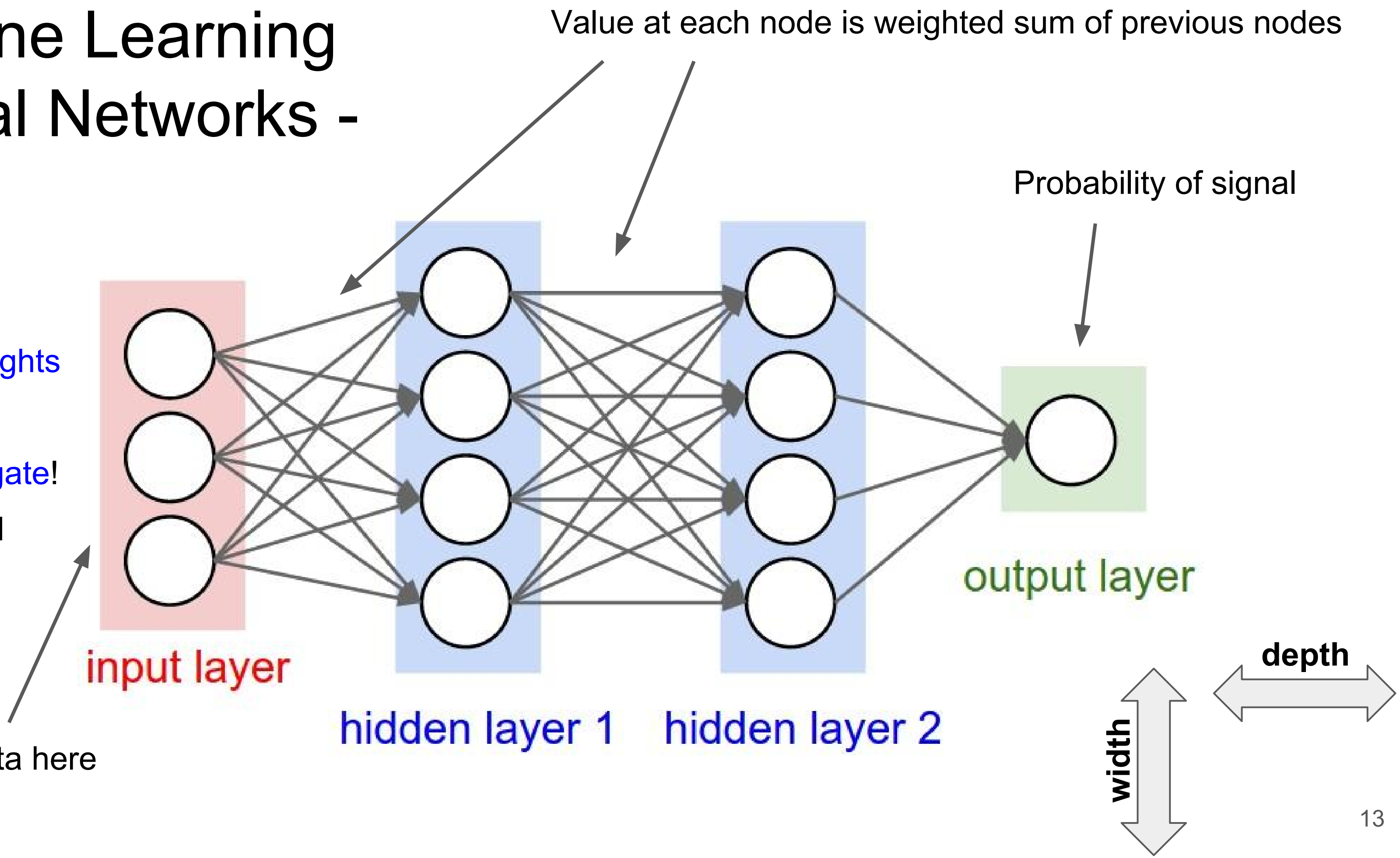
↓

Use test set to check model



Machine Learning
- Neural Networks -

- Start with random weights
- Get output
- Backpropagate!
- Repeat until small error



Machine Learning
- Hardware -

High dimensionality leads to highly parallelizable problem

↓

Use GPUs for training

Currently using two NVIDIA GTX TITANs @ Caltech...

...because CERN has no GPUs :(

Machine Learning
- Model Comparison -

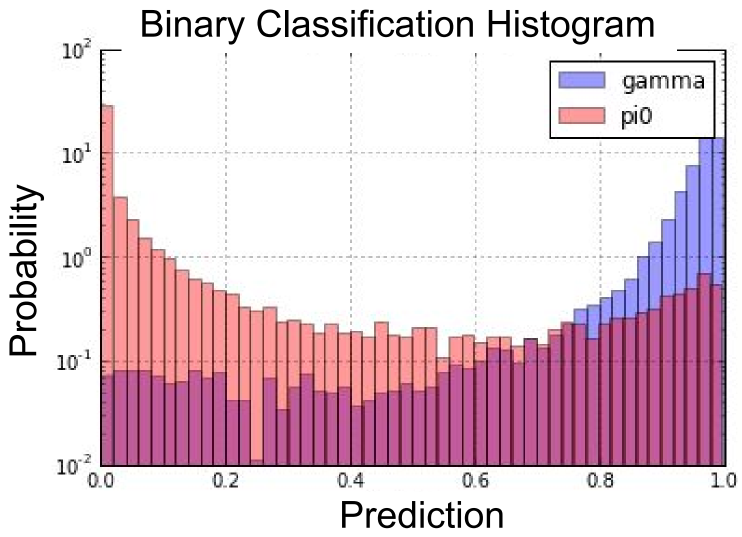
How to compare techniques?

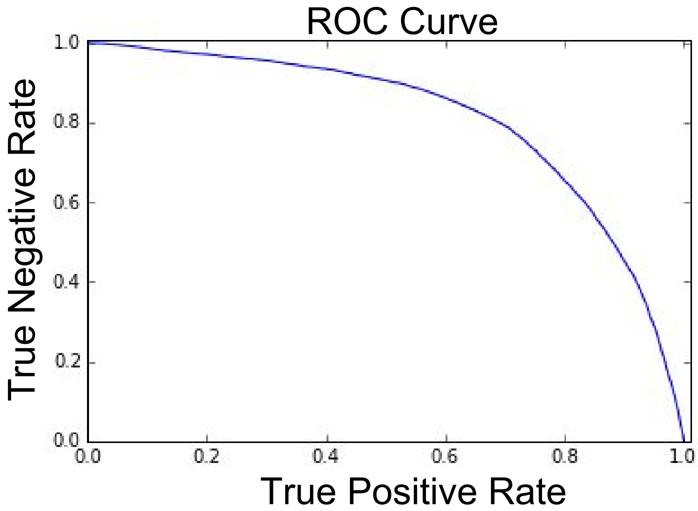
- Complexity / understandability
- Time to train
- Performance

ROC curves to compare performance

Give visual representation of Sensitivity vs. Specificity

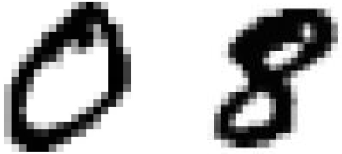
(aka True Positive Rate vs True Negative Rate)

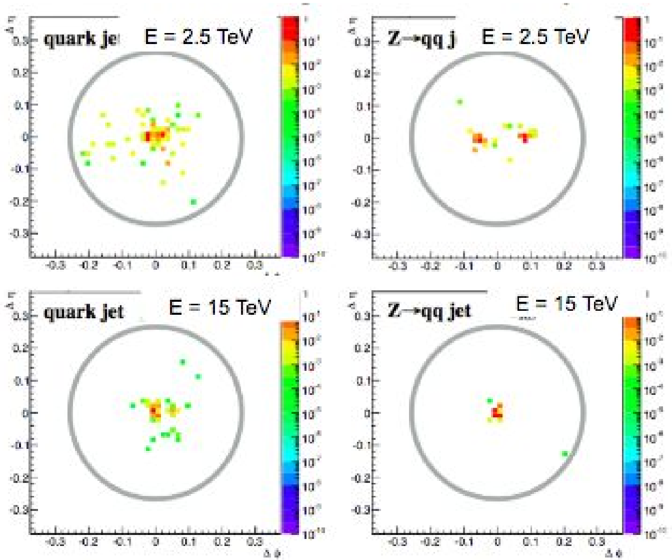


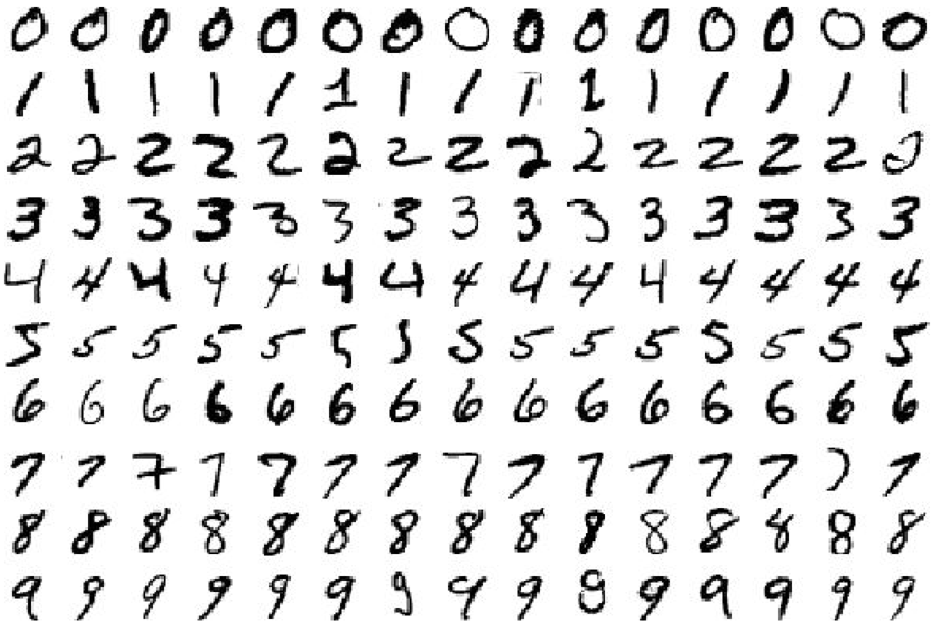


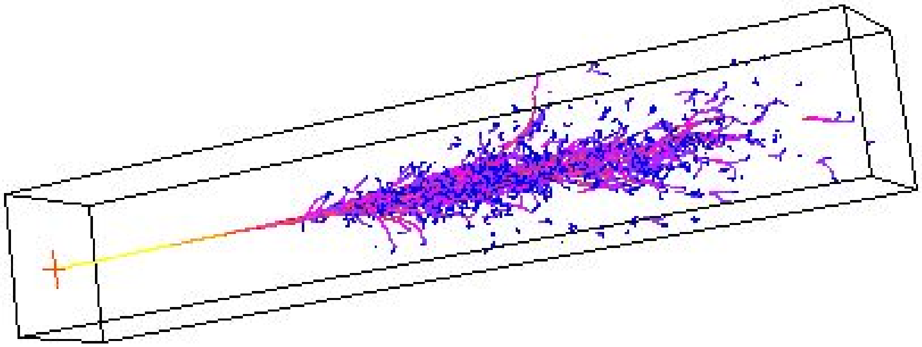
Particle ID
- // to MNIST -

- Imaging problem
- 2D vs 3D clusters
- “0” vs “8” analogy with “gamma” vs “pi0”









Our work

Project
- Tools -

python™

Software

jupyter

K

theano

Hardware

Project
- Data Preparation -

Use ROOT to generate 100-GeV single-particle events in Geant4 LCD calorimeter (@ CERN)

→

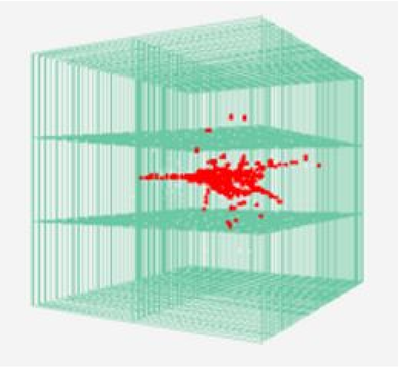
Extract ix, iy, iz, and E for each event; save in long python list (@ CERN)

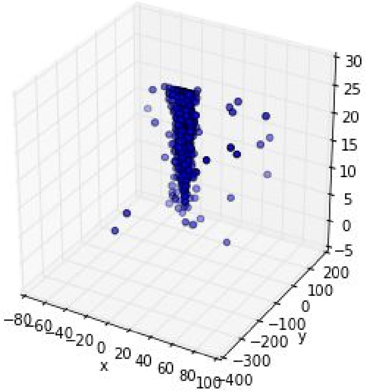
→ Send to Caltech!

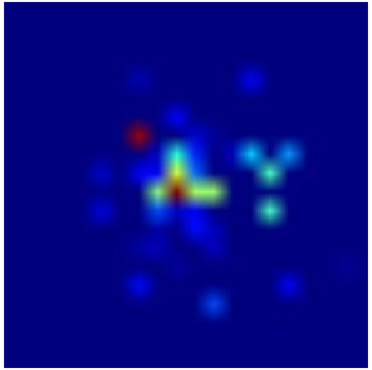
Parse into NumPy arrays; make 20x20x25 centered event arrays (@ Caltech)

→

Store ~40,000 event gamma and pi0 datasets (@ Caltech)





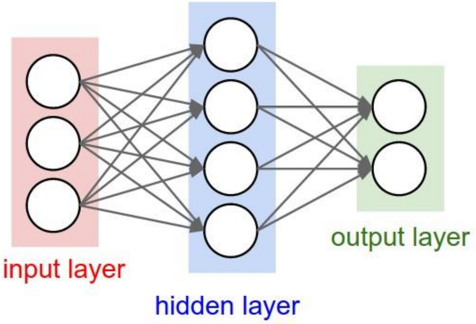


Project
- Network Topologies -

Three main types:

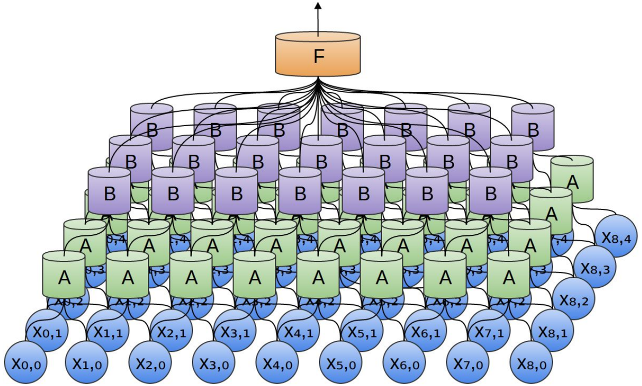
Fully connected

- Simple
- Generally fast
  (~1s per epoch)



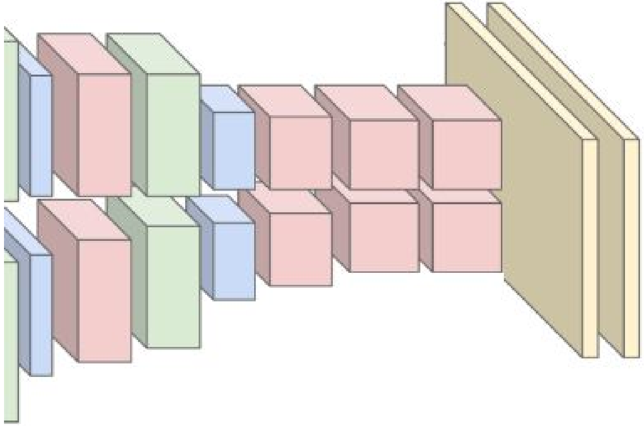
Convolutional

- Preserves Structure
- Slower
  (~60s per epoch)

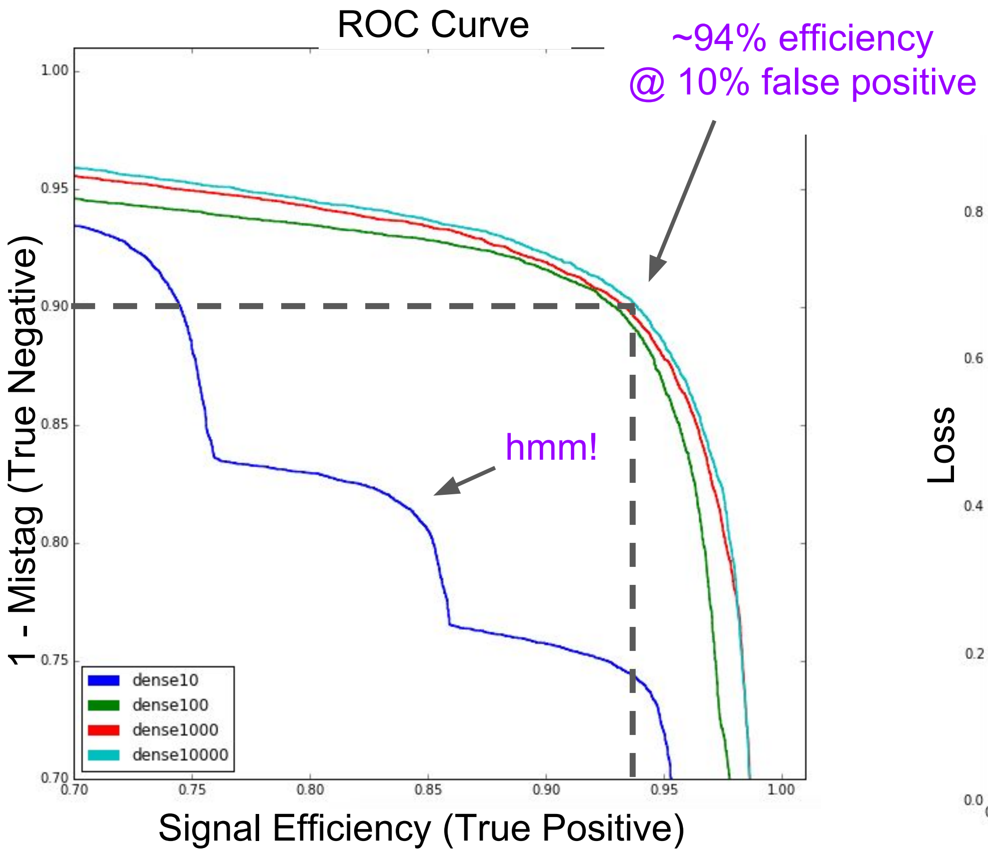


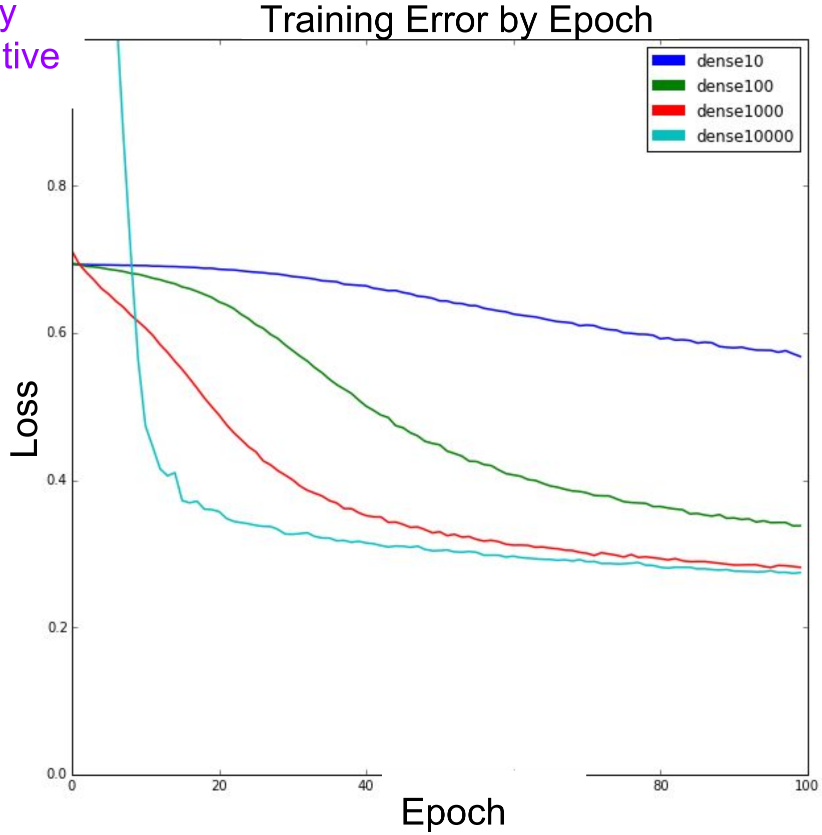
Branched and Multilayer Convolutional

- Even better feature recognition
- Takes long time to train
  (~180s per epoch)

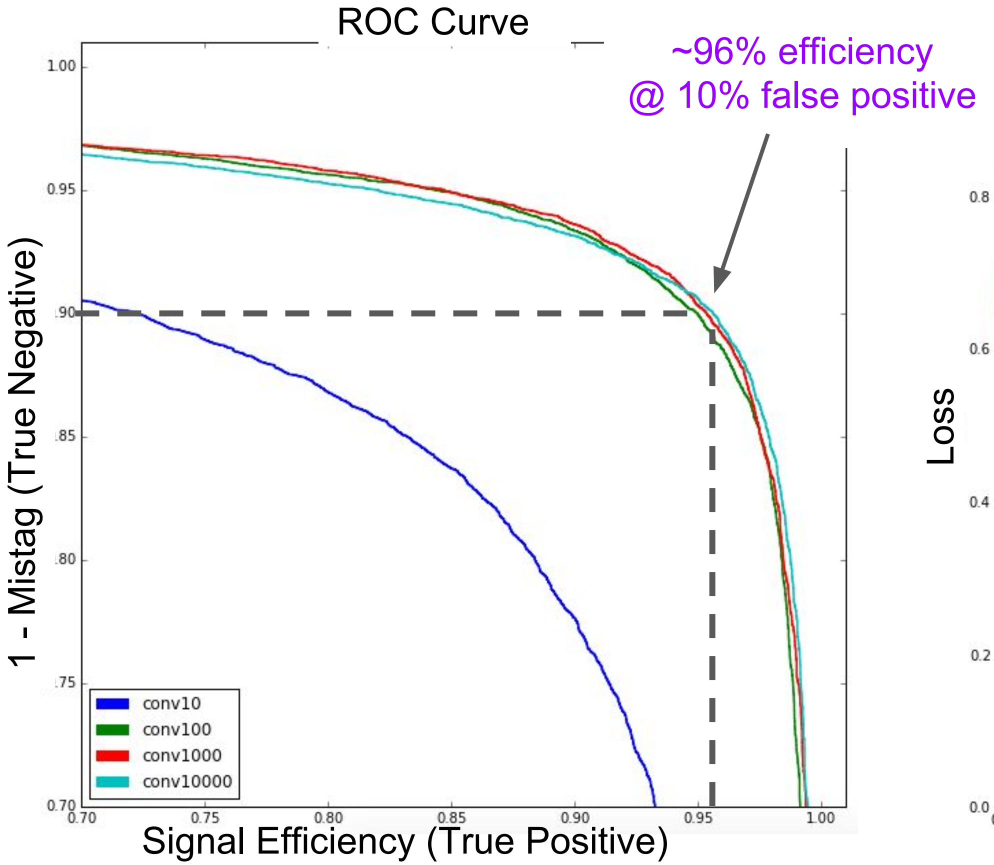


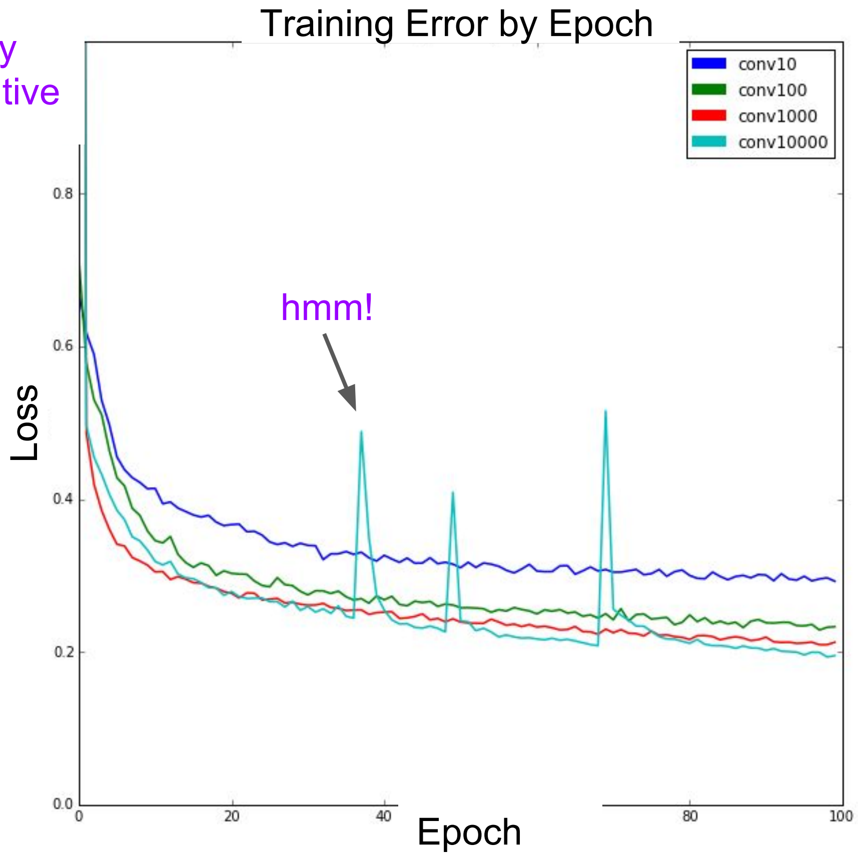
Project - Network Topologies - Fully Connected



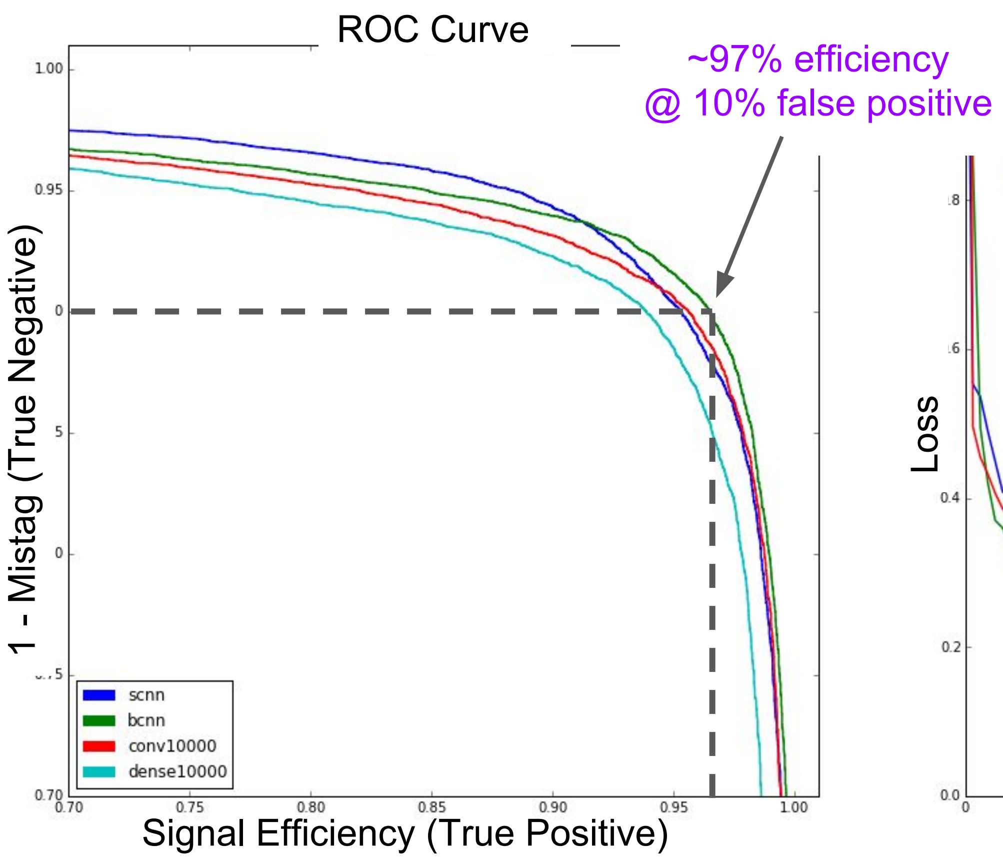


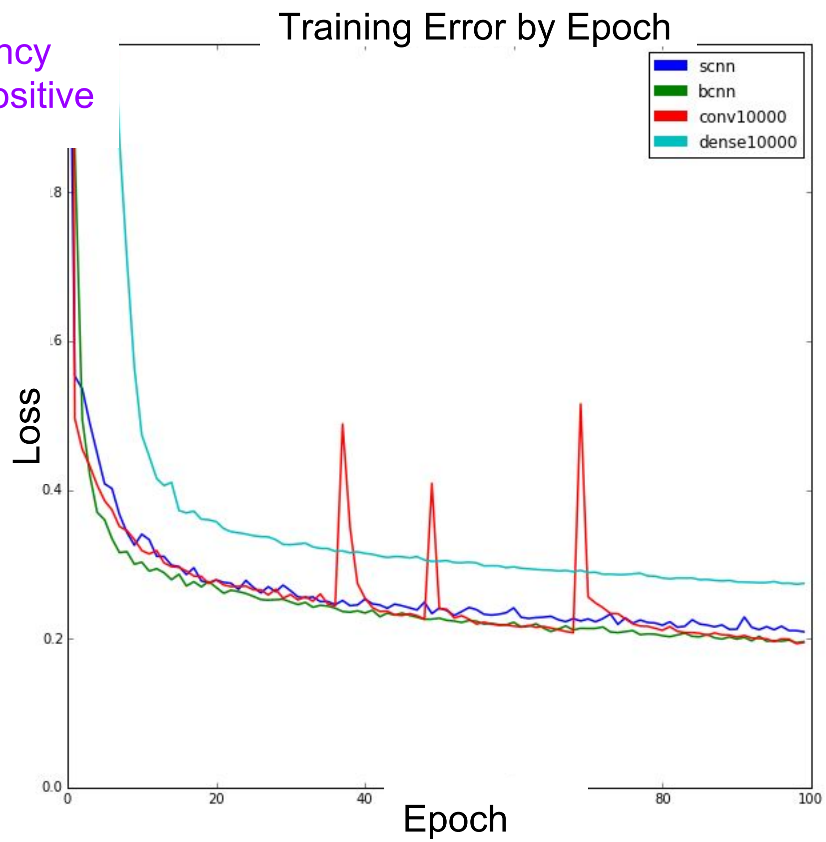
Project - Network Topologies - Convolutional



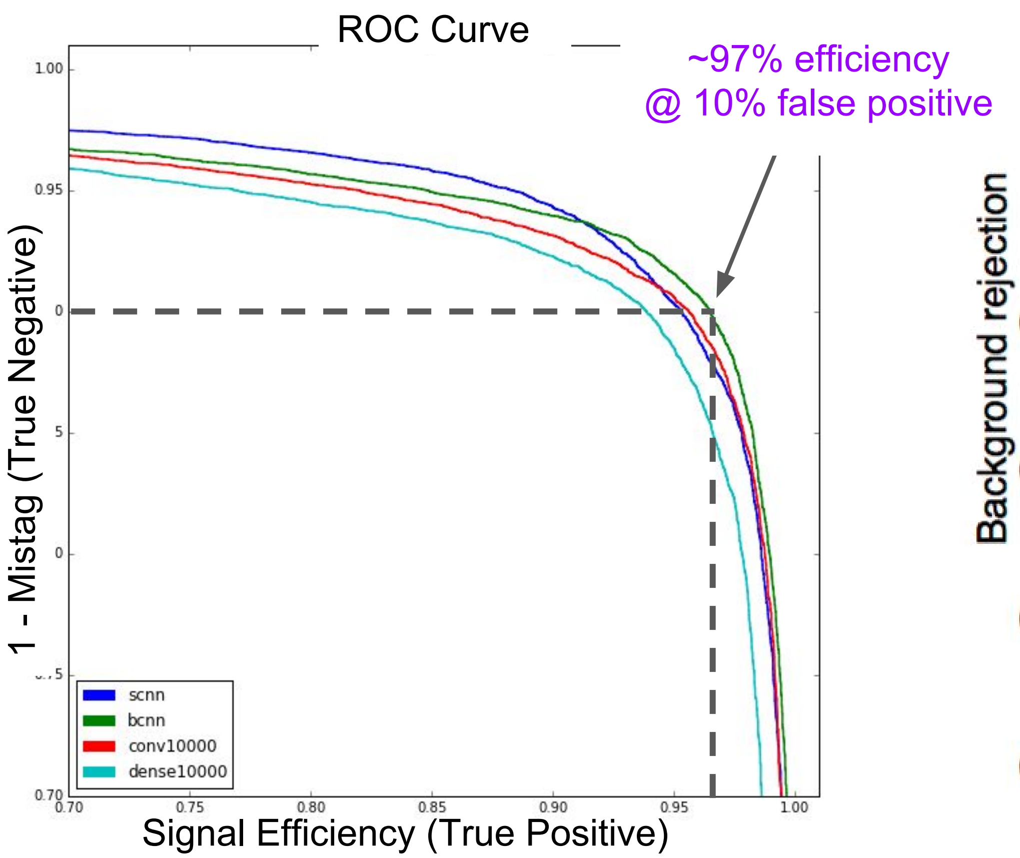


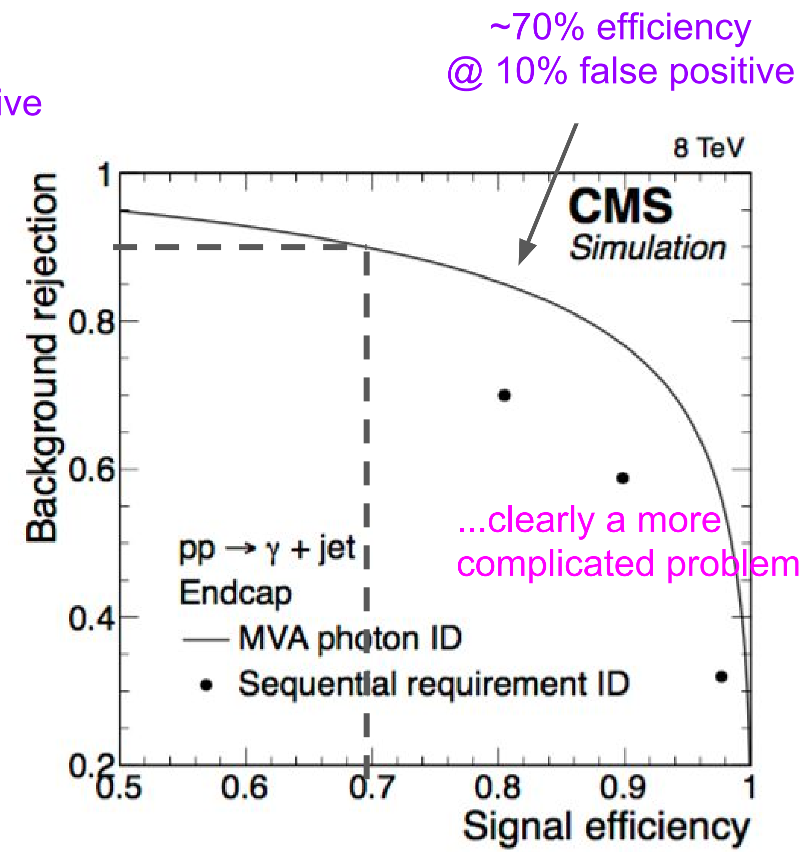
Project - Network Topologies - Convolutional II





Project - Network Topologies - Convolutional II





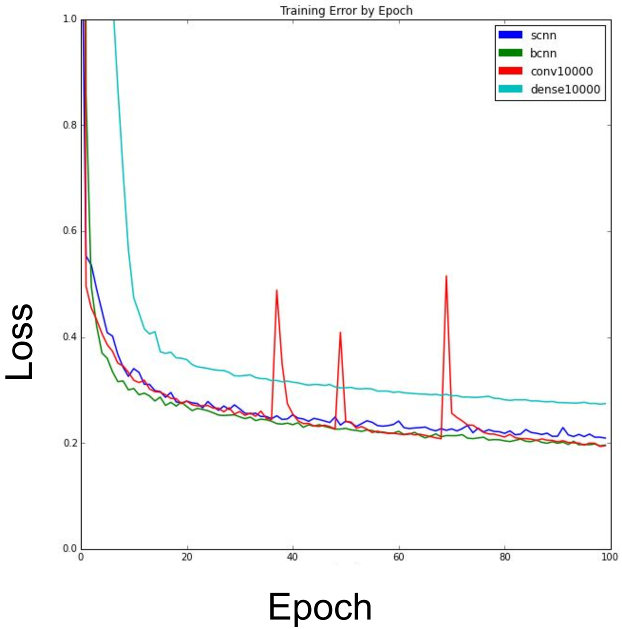
Limitations and Moving On

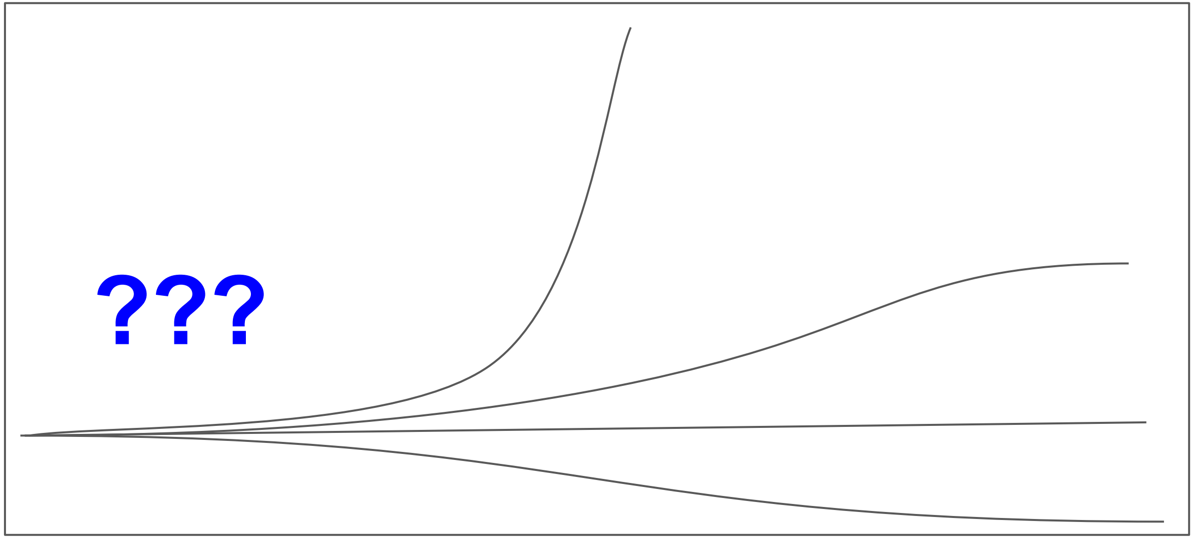
Project - Current Limitations

How long to train?

“Best” model?

- Loss keeps decreasing well after 100 epochs
- Don’t have enough GPUs for quick training!
    - Looking to get time on CSCS GPU cluster





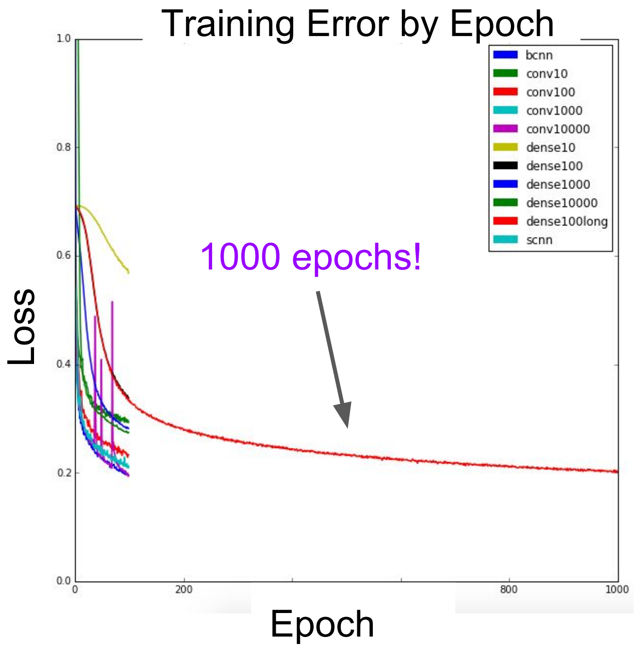
Project - Current Limitations

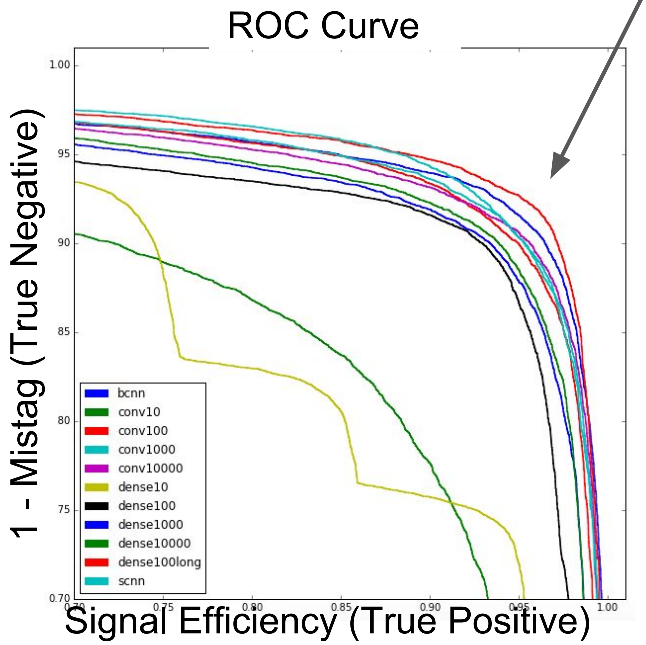
How long to train?

“Best” model?

- Loss keeps decreasing well after 100 epochs
- Don’t have enough GPUs for quick training!
    - Looking to get time on CSCS GPU cluster

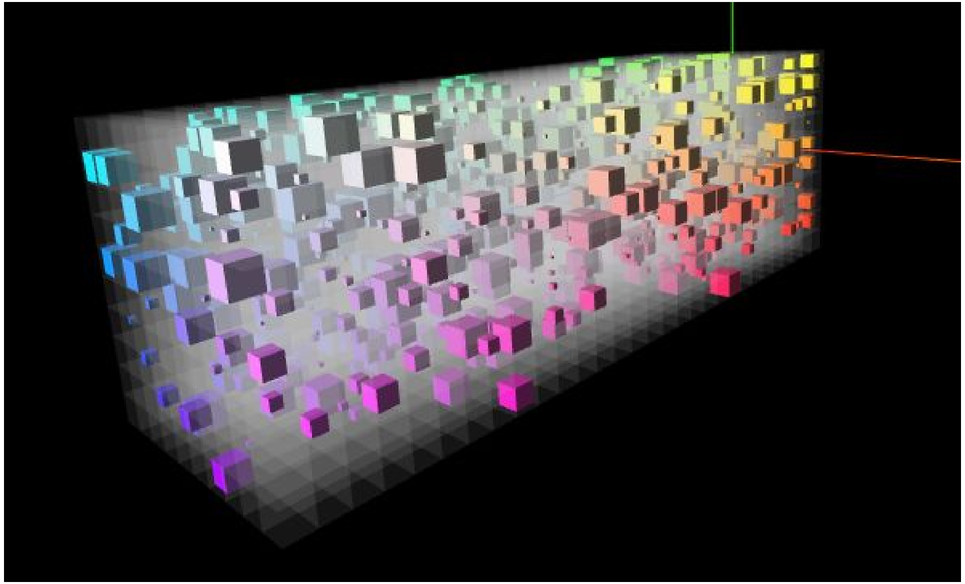
~98% efficiency
@ 10% false positive





Project
- Next weeks -

- Identify 'ideal' network topology for binary classification
  - Scan over hyperparameters
- Regression on energy



- Future -

Beyond 1-1 classification

- Jet ID
- Gluon vs quark vs boosted Higgs vs boosted Z vs boosted top

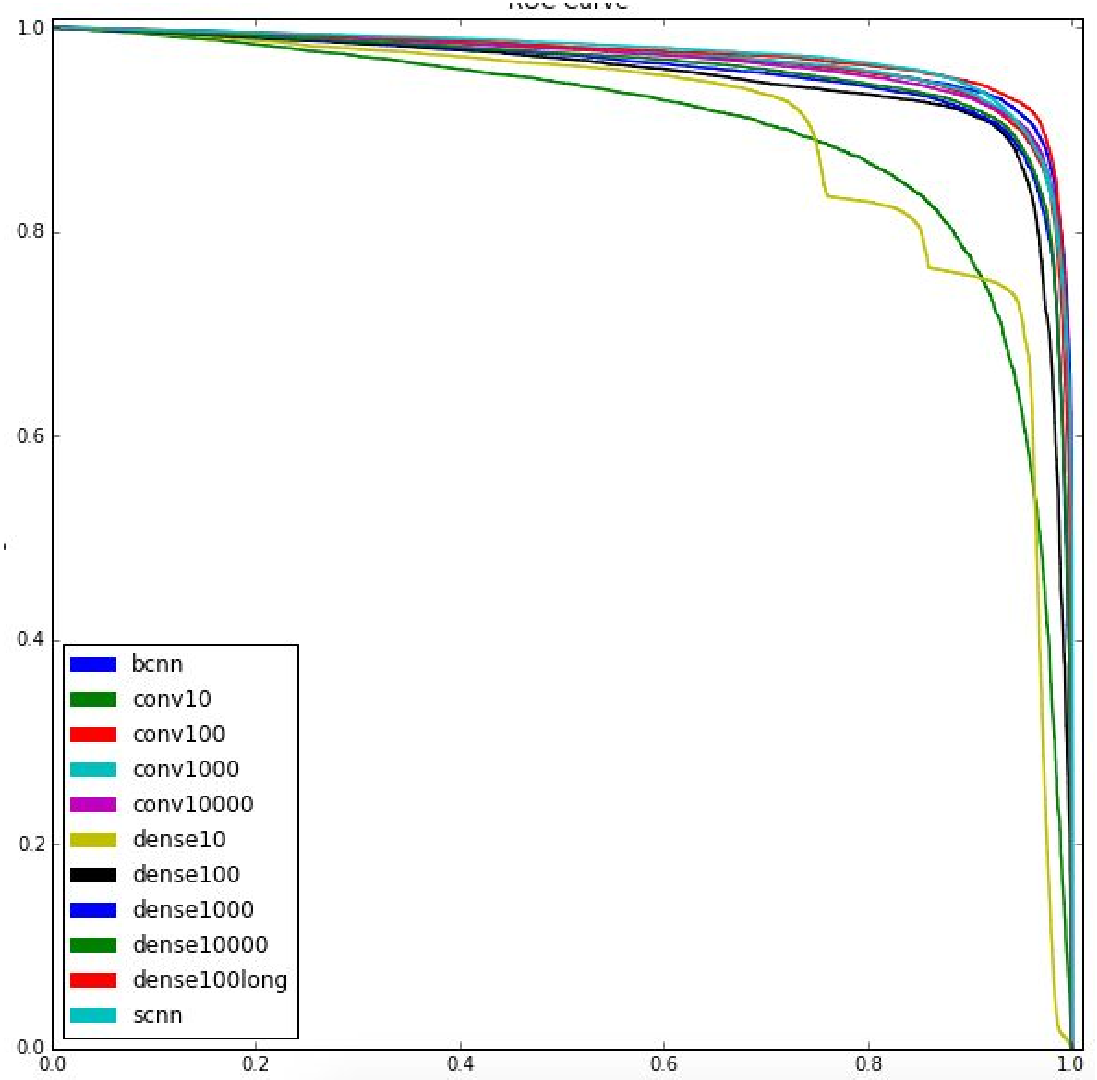
Provide general resource

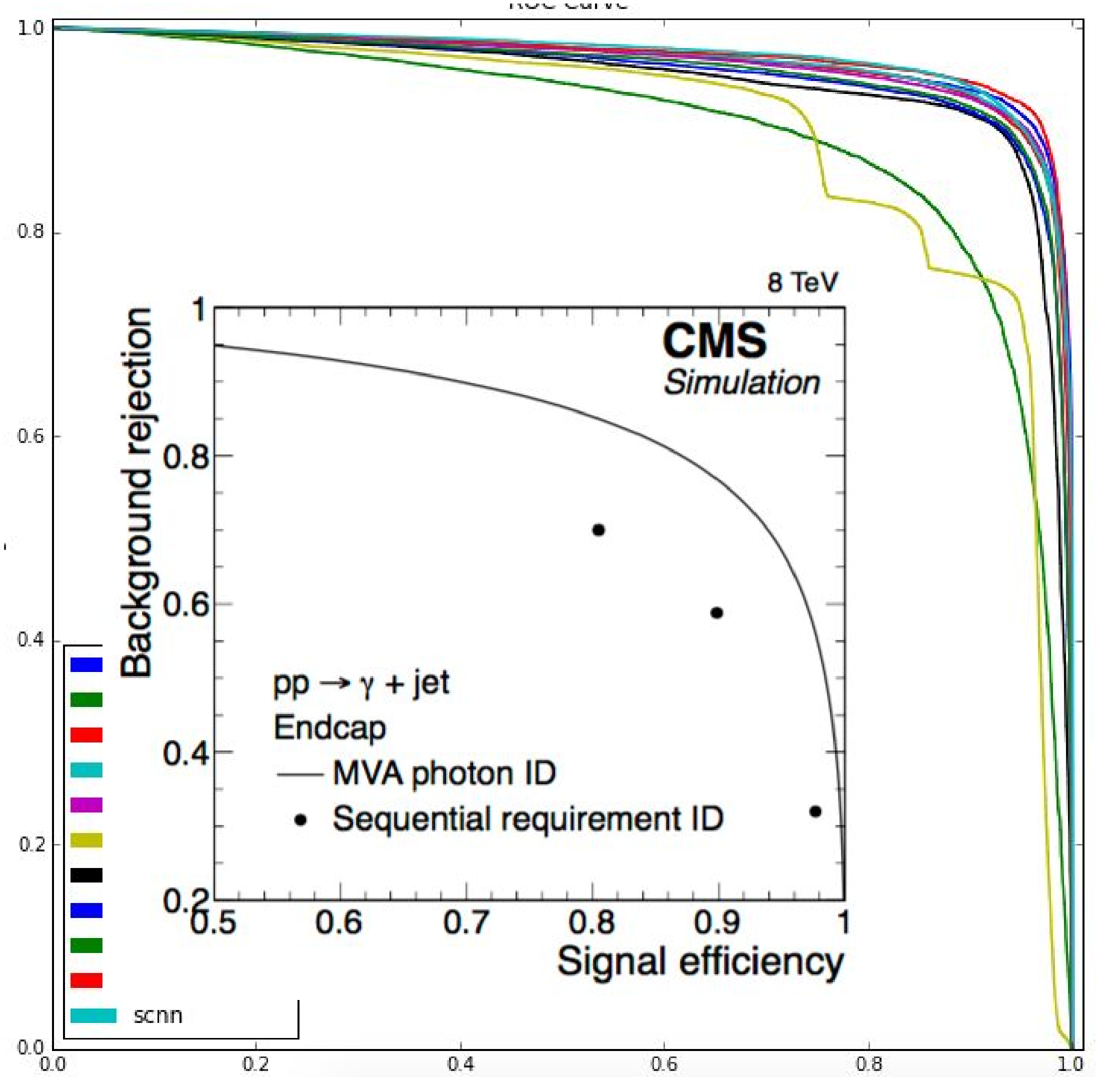
- Benchmark for professionals
- Publish on Open data as a ML standard (the MNIST of Particle Physics)

Big thank you to BU, Augusto, Maurizio & Jean-Roch!

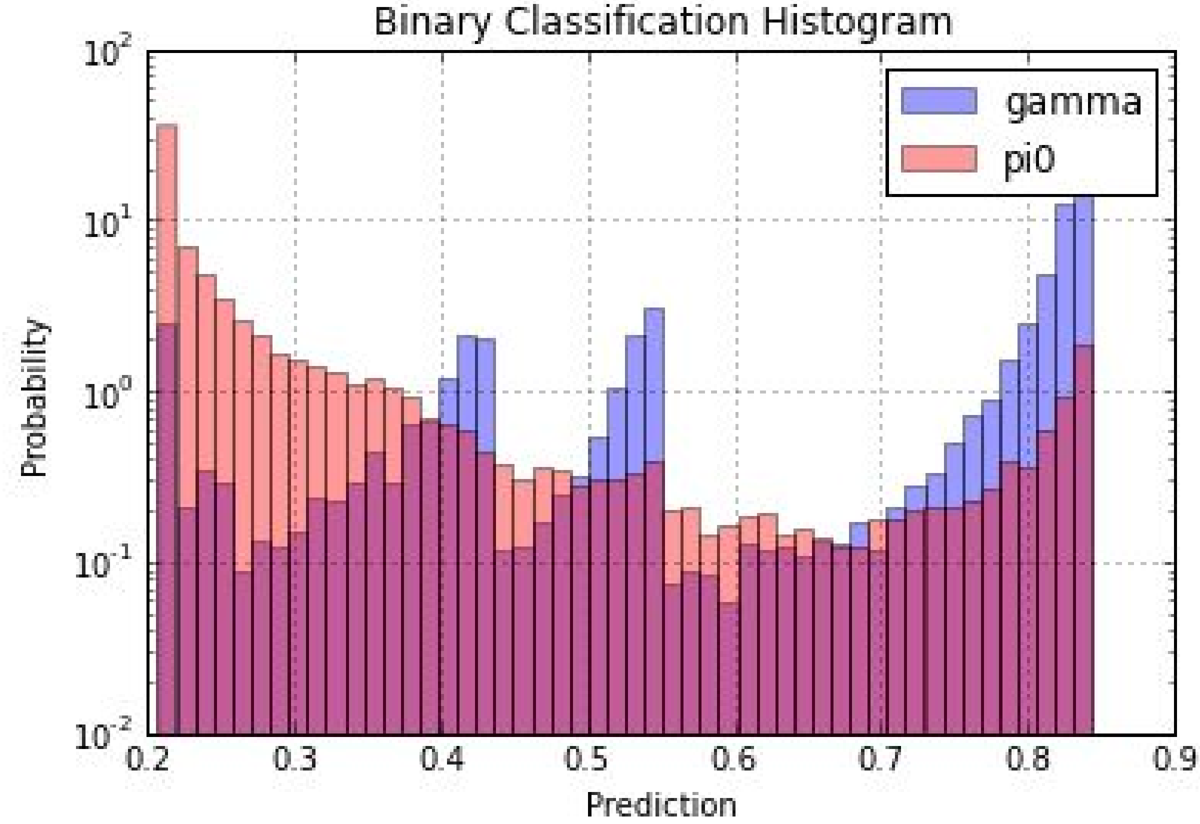
...any questions?

Backup Slides

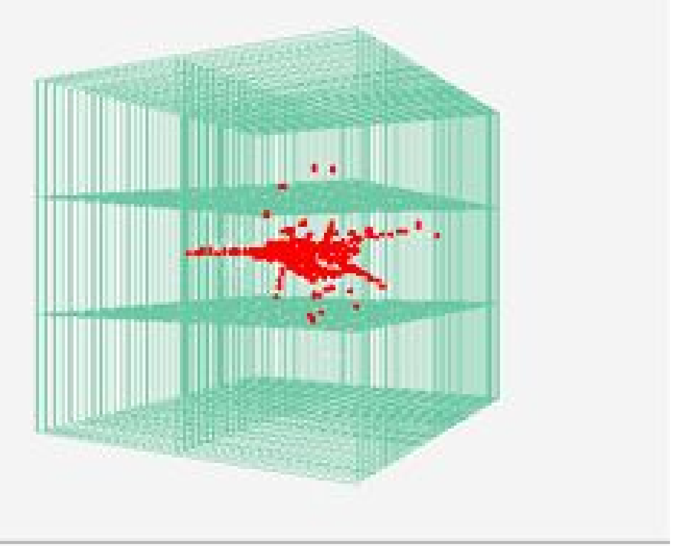

1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
bcnn
conv10
conv100
conv1000
conv10000
dense10
dense100
dense1000
dense10000
dense100long
scnn

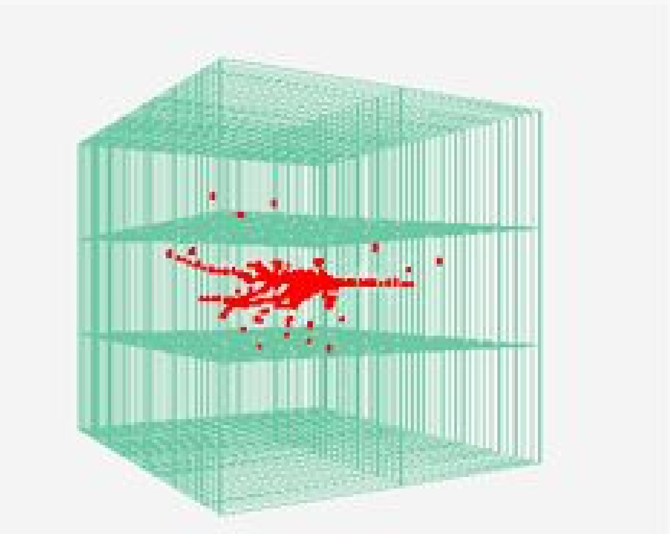


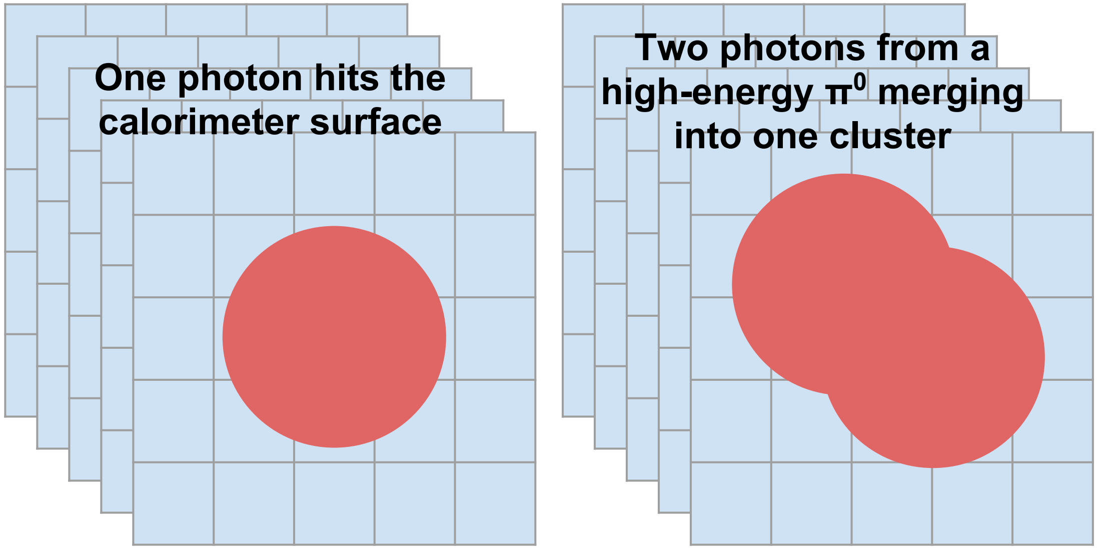
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
scnn
8 TeV
CMS
Simulation
Background rejection
Signal efficiency
pp → γ + jet
Endcap
MVA photon ID
Sequential requirement ID
1
0.8
0.6
0.4
0.2
0.5
0.6
0.7
0.8
0.9
1



https://titans.hep.caltech.edu:8182/notebooks/Niki/Plot_Models.ipynb



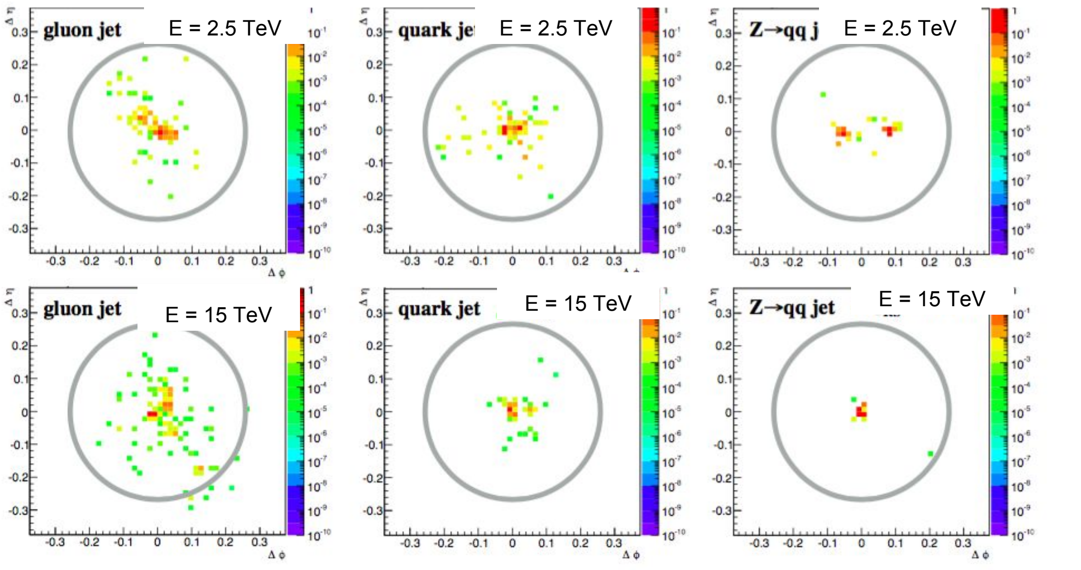


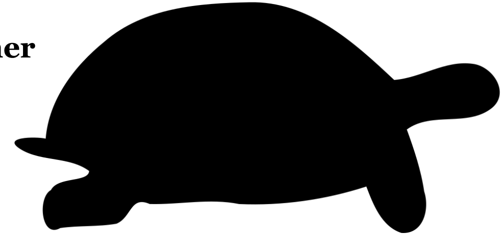

One photon hits the calorimeter surface
Two photons from a high-energy π^0 merging into one cluster

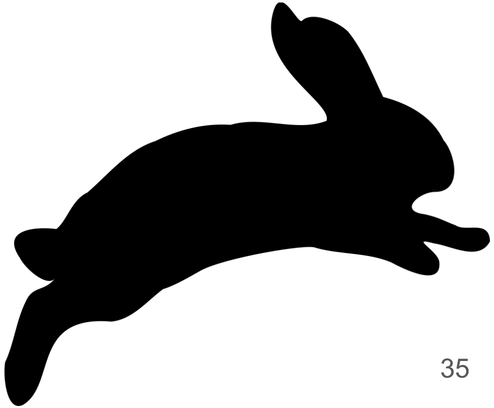
Particle ID
- Importance -

1) Fast yes/No answer based on what a cluster look like
   a) Problem reduced to reading the detector and evaluating the function
   b) No complex reconstruction involved
   c) Faster than actual reconstruction good for granular-calorimeter triggers at future colliders? (e.g. FCC, High-Luminosity LHC, etc)
2) Potentially, could reach similar performances as traditional techniques
   a) Energy reconstruction from regression (already done offline with BDTs @LHC)
   b) Particle identification from classification
3) **Could be extended to more complex problems (identify jets of particles, rather than single particles)**

Jet-constituent energy fraction as a function of the angular distance from the jet axis







Introduction
- ML for Particle ID -

Machine learning successfully used for Particle Identification in HEP

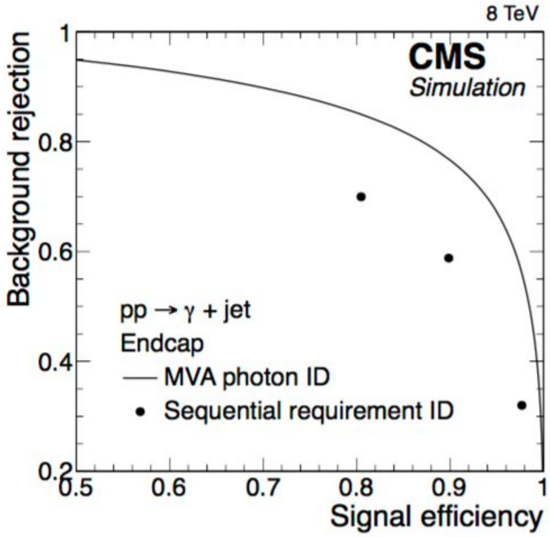
- Neural Networks @ LEP
- Boosted Decision Trees @ BaBar/Belle and the LHC

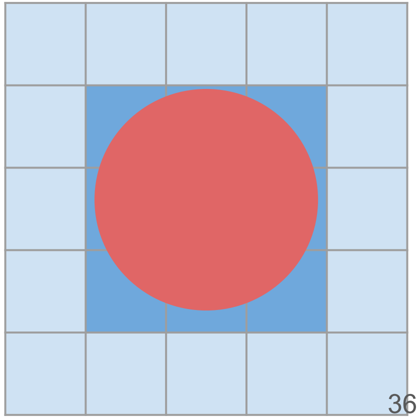
Particle Identification is a classical *supervised classification problem:*

- A training sample for signal and background available from real collisions and/or simulation
- A set of features identified, having discriminating power between signal and background. RAW data already “manipulated” to select an optimal set of physics-motivated variables. Good for the physicist, but potentially limiting for the algorithm.
- A selection algorith (cut-based selection/BDT/NN/etc) trained to maximize the separation

As a next step in this development line, we try to apply Deep Neural Networks

- Go back to RAW data
- Train a convolutional Neural Network to LOOK AT the clusters and see the pions [THE IMAGING CALORIMETRY]





An example of “high-level” feature (i.e., not RAW data)
R9 = (energy 3x3 cluster)/(energy 5x5 cluster)

See for instance CMS photon ID in Run I

Moving on...

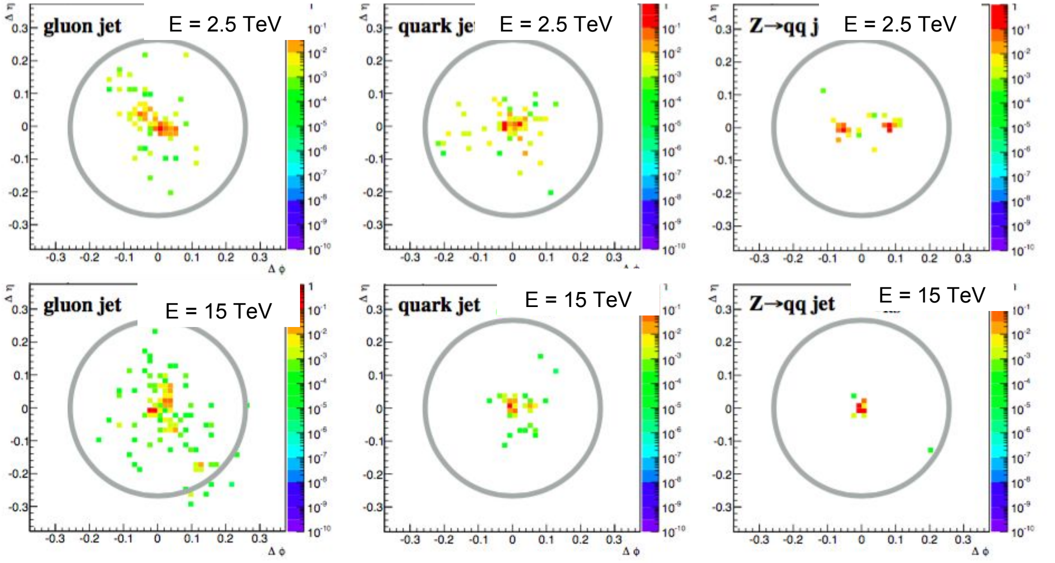
1) In this study:
   a) Regression on energy as well as pid
2) Potentially:
   a) Beyond 1-to-1 classification: Jet ID
      i) Gluon vs quark vs boosted Higgs vs boosted Z vs boosted top
3) In the future:
   a) More complex algorithms: provide a benchmark for professionals
      i) Publish dataset on Open data as a standard (the MNIST of Particle Physics) ← how exactly does this work?

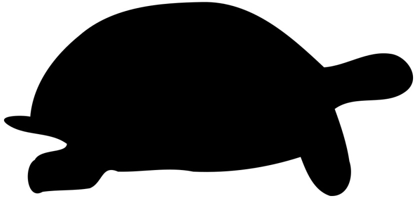
Show the event display they made for us

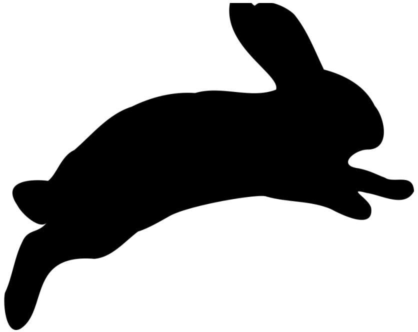
Particle ID
- Importance -

- Yes/No answer based on what a cluster looks like - no complex reconstruction
- Faster than actual reconstruction in triggers at future colliders? (e.g. FCC, High-Luminosity LHC, etc)
- Similar performance to traditional techniques for particle ID?
  - Energy reconstruction from regression (already done offline with BDTs @LHC)
- More complex problems?
  - Jet ID?

Jet-constituent energy fraction as a function of the angular distance from the jet axis







Ok, so what have we done so far?