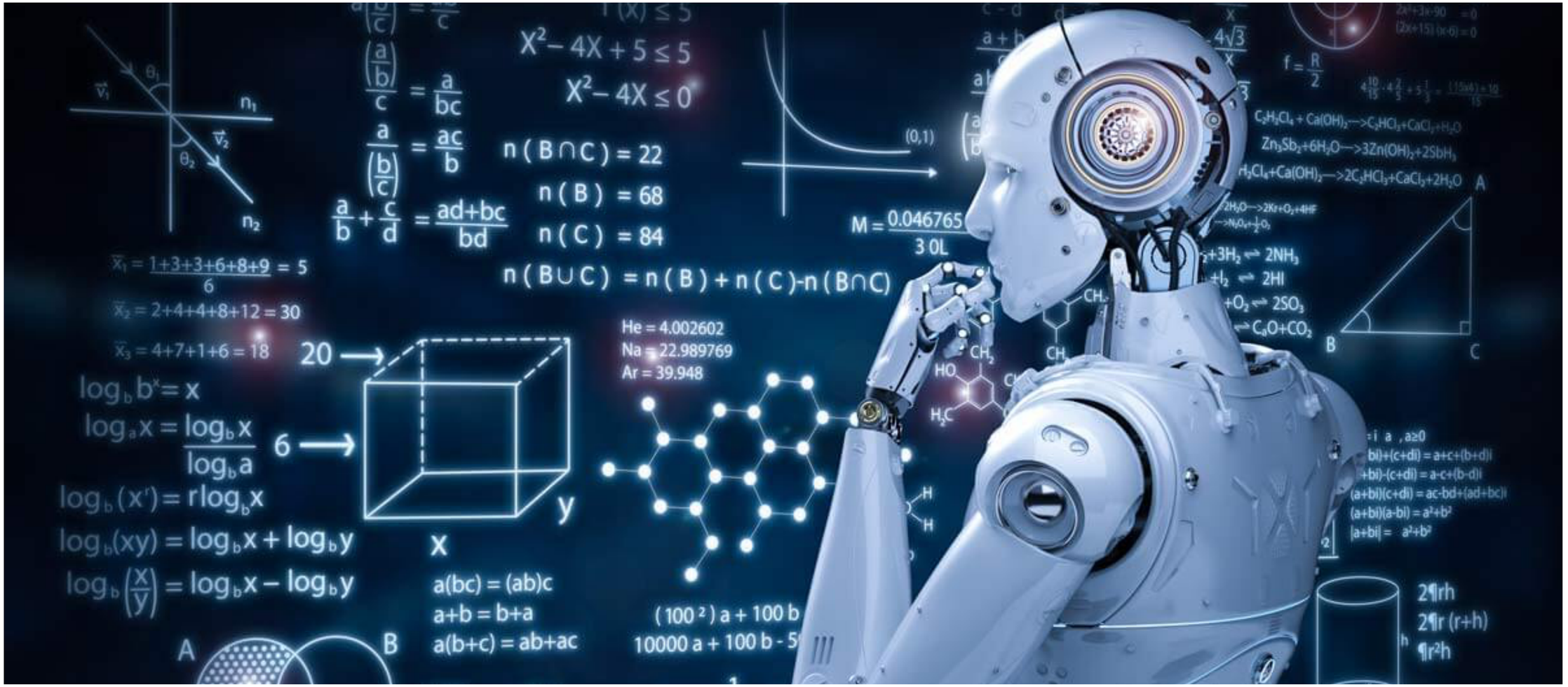
DEEP LEARNING

EPISODE I

Maurizio Pierini

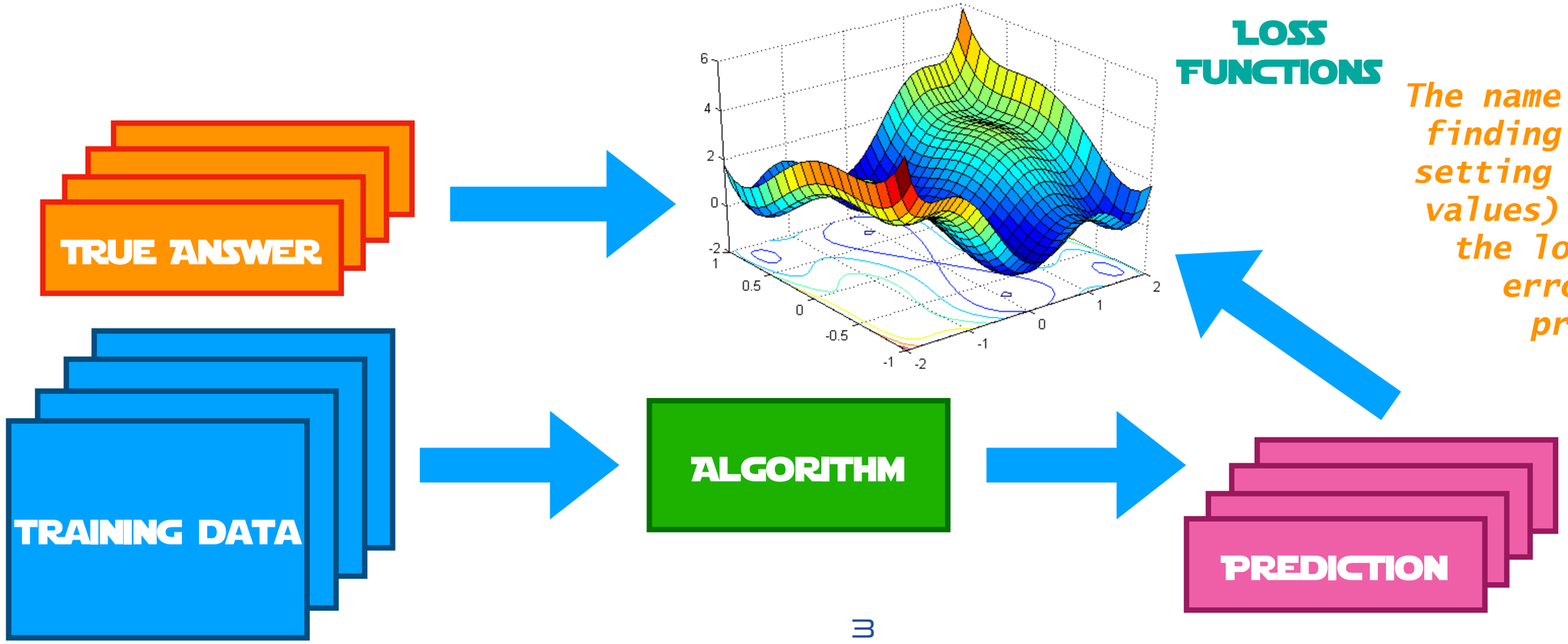


What is Machine Learning ?

European Research Council

A definition (Wikipedia)

Machine learning (ML) is the scientific study of algorithms and statistical models that computer systems use to progressively improve their performance on a specific task. Machine learning algorithms build a mathematical model of sample data, known as "training data", in order to make predictions or decisions without being explicitly programmed to perform the task.

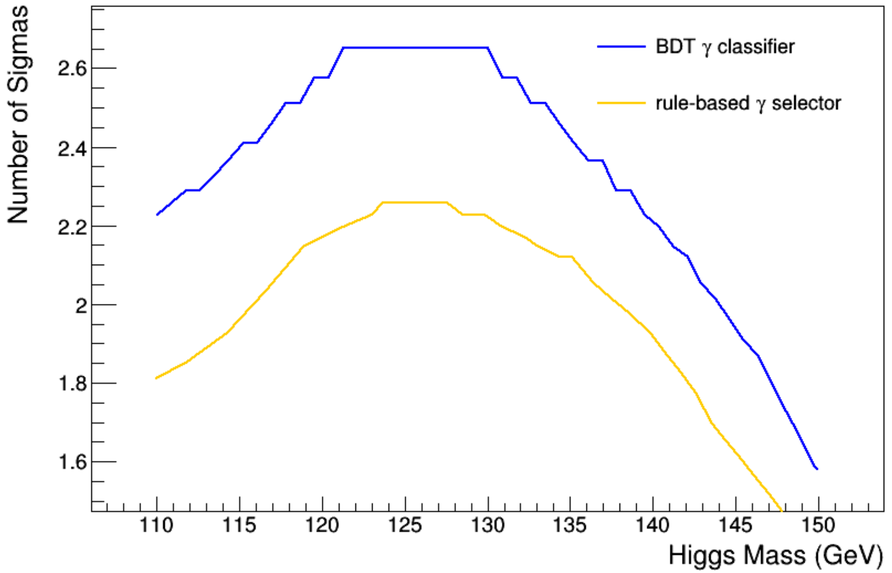


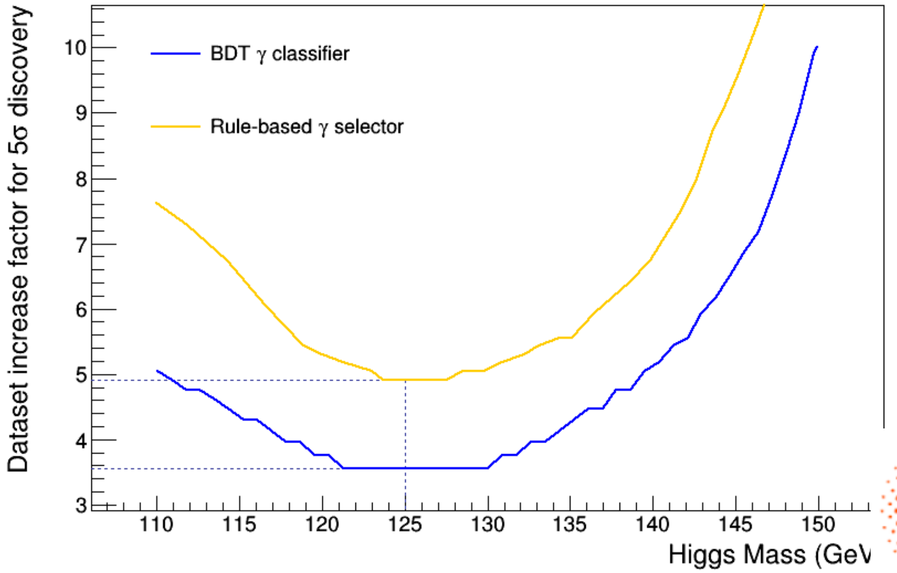
Typical problems

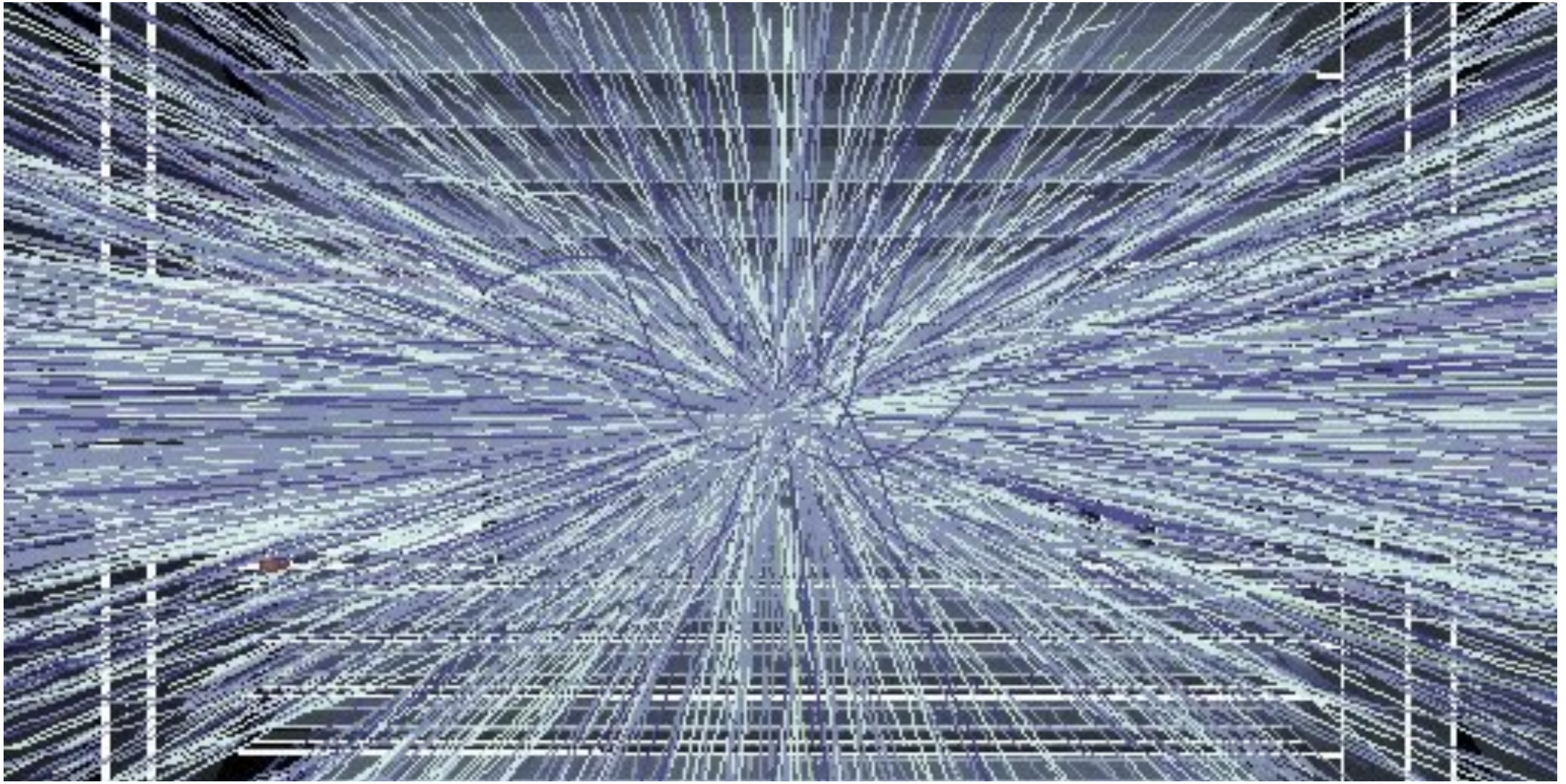
- *Classification:*
  - *given an image, identify the object represented*
  - *in particle physics, given a particle shower, identify the particle kind*
- *Regression:*
  - *given a set of quantities x, learn some function f(x)*
  - *in particle physics, given a particle shower, learn its energy*
- *Many more (anomaly detection, generation, denoising, etc), mainly thanks to new architectures (deep learning)*

Machine Learning in HEP

- *Long tradition*
  - *Neural networks used at LEP and the Tevatron*
  - *Boosted Decision Trees introduced by MiniNooNE and heavy used at BaBar*
  - ***BDTs ported to LHC and very useful on Higgs discovery***
  - *Now Deep Learning is opening up many new possibilities*

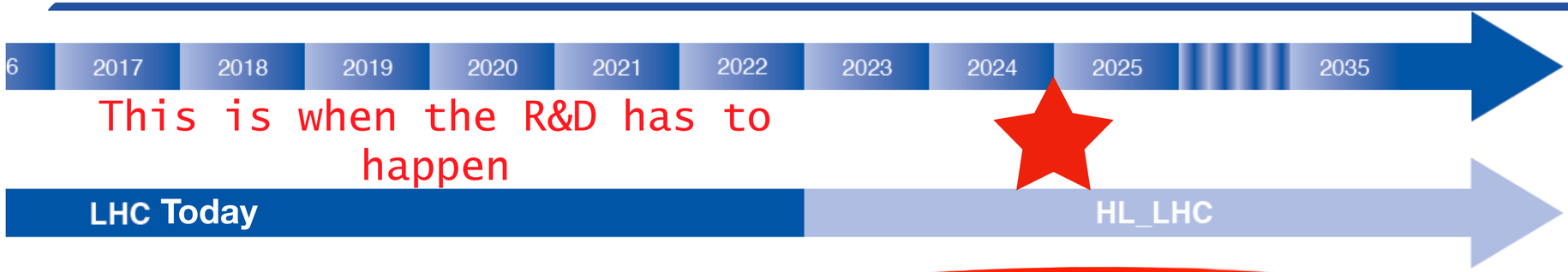






Why do we need Machine Learning?

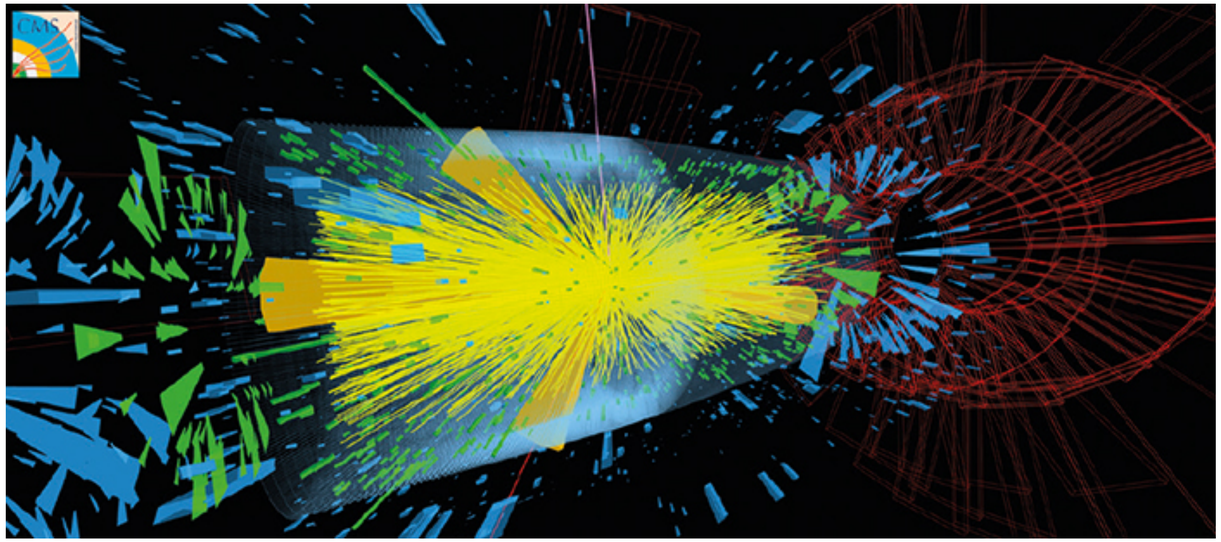
HL-LHC: elephant in the room



- ~40 collisions/event
- ~10 sec/event processing time
- (at best)Same computing resources as today

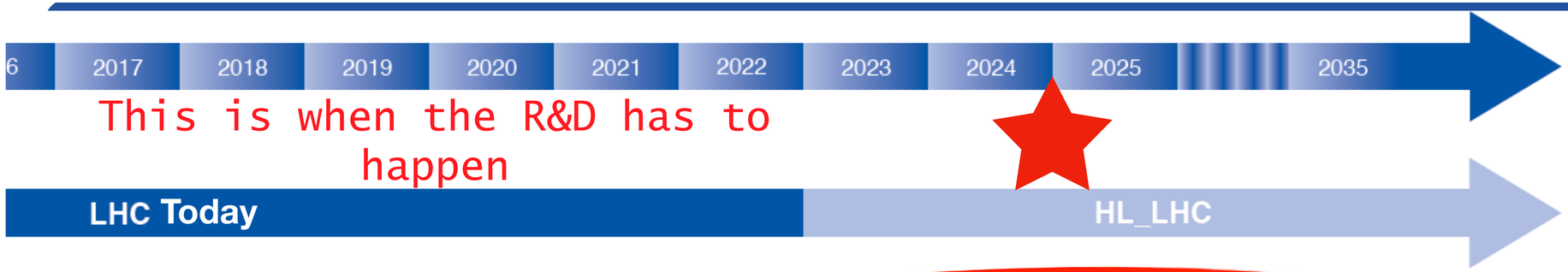
- ~200 collisions/event
- ~minute/event processing time(*)
- (at best)Same computing resources as today

- *Flat budget vs. more needs = **current rule-based reconstruction algorithms will not be sustainable***
- ***Adopted solution:** more granular and complex detectors → more computing resources needed → more problems*
- ***Modern Machine Learning might be the way out***



(*)With nowadays software development

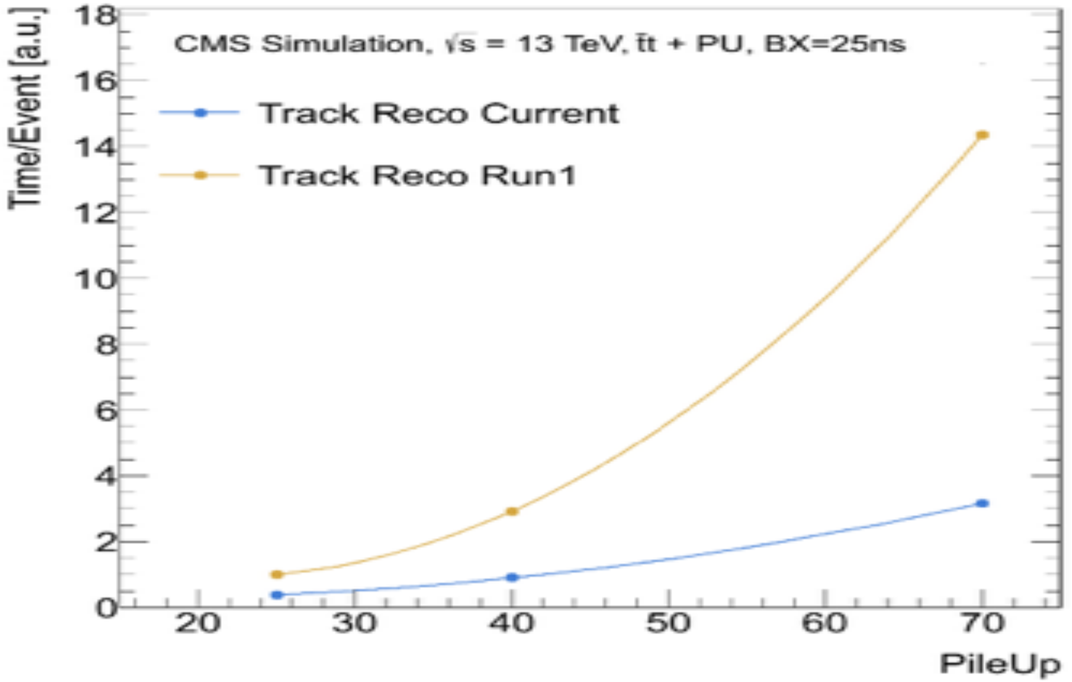
HL-LHC: elephant in the room



- ~40 collisions/event
- ~10 sec/event processing time
- (at best)Same computing resources as today

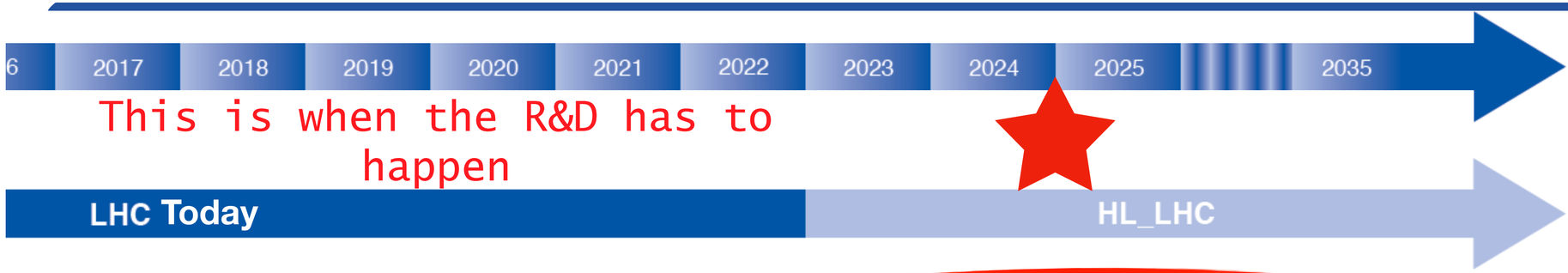
- ~200 collisions/event
- ~minute/event processing time(*)
- (at best)Same computing resources as

- *Flat budget vs. more needs = **current rule-based reconstruction algorithms will not be sustainable***
- ***Adopted solution:** more granular and complex detectors → more computing resources needed → more problems*
- ***Modern Machine Learning might be the way out***



(*)With nowadays software development

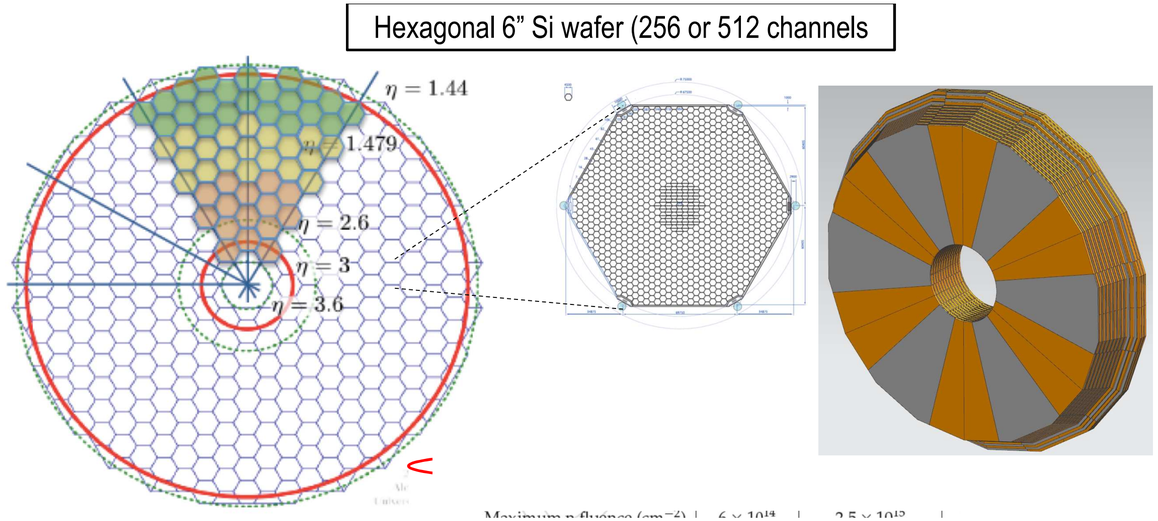
HL-LHC: elephant in the room



- ~40 collisions/event
- ~10 sec/event processing time
- (at best)Same computing resources as today

- ~200 collisions/event
- ~minute/event processing time(*)
- (at best)Same computing resources as today

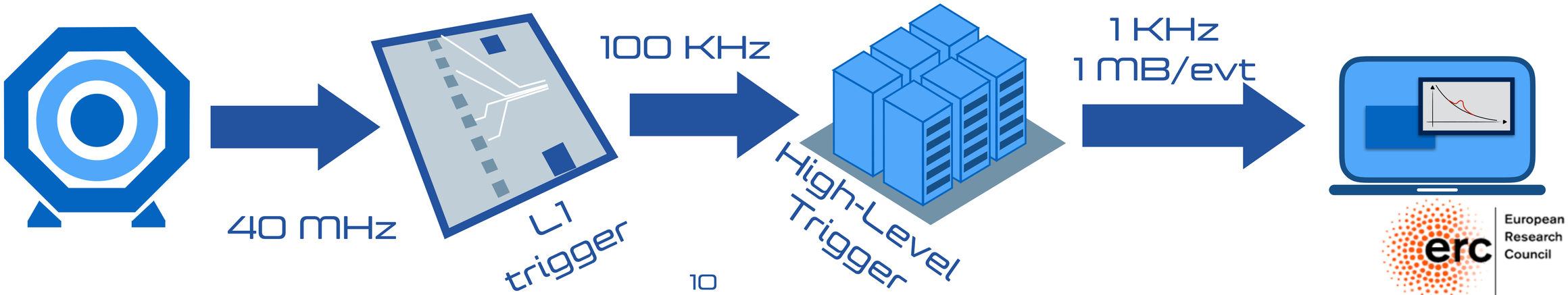
- *Flat budget vs. more needs =* ***current rule-based reconstruction algorithms will not be sustainable***
- ***Adopted solution:*** *more granular and complex detectors → more computing resources needed → more problems*
- ***Modern Machine Learning might be the way out***



(*)With nowadays software development

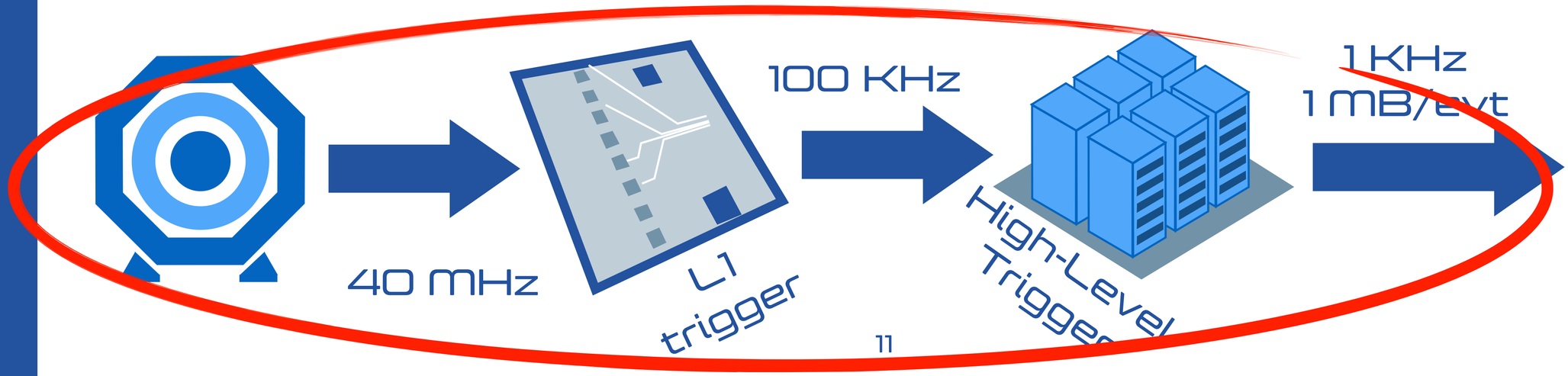
The LHC Big Data Problem

- *Too many data, too large data -> need to filter online*
- *Filters based on theoretical bias: **we might be loosing good events***
  - *L1 trigger: local, hardware based, on FPGA, @experiment site*
  - *HLT: local/global, software based, on CPU, @experiment site*
  - *Offline: global, software based, on CPU, @CERN T0*
  - *Analysis: user-specific applications running on the grid*

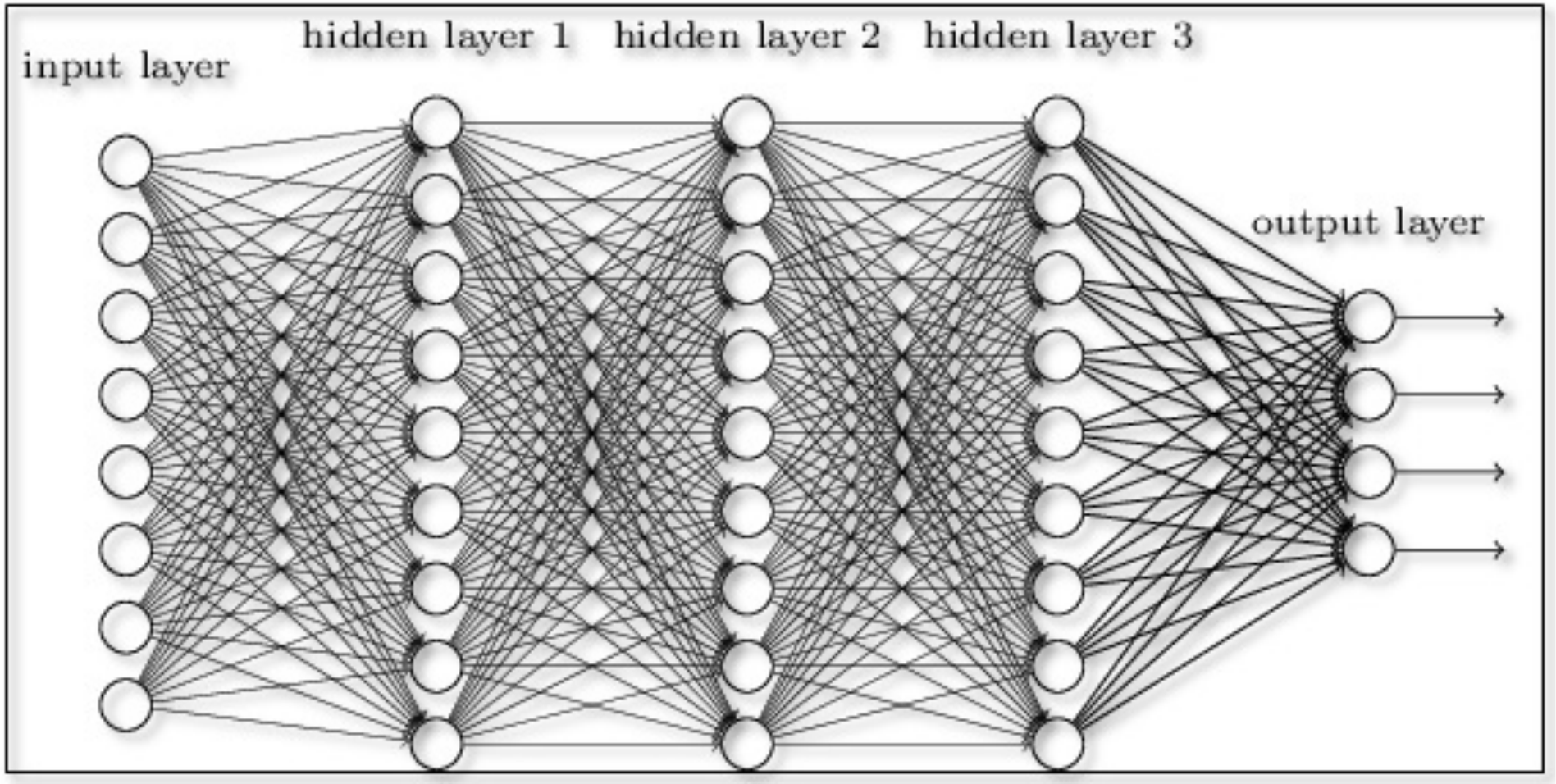


The LHC Big Data Problem

- *We are not seeing new physics: just re-doing what we do today with x10 more data WILL NOT be enough.*
- *The solution to the HL-LHC problem: modern Machine Learning as a fast shortcut between the data and the right answer (the outcome of our traditional & slow algorithms)*
- *This ML deployment need to happen in between collisions and data analysis (trigger, reconstruction, …), where freeing resources will make a difference*



European Research Council



Deep Learning

Deep Learning everywhere

Everything is a Recommendation

Over 75% of what people watch comes from our recommendations

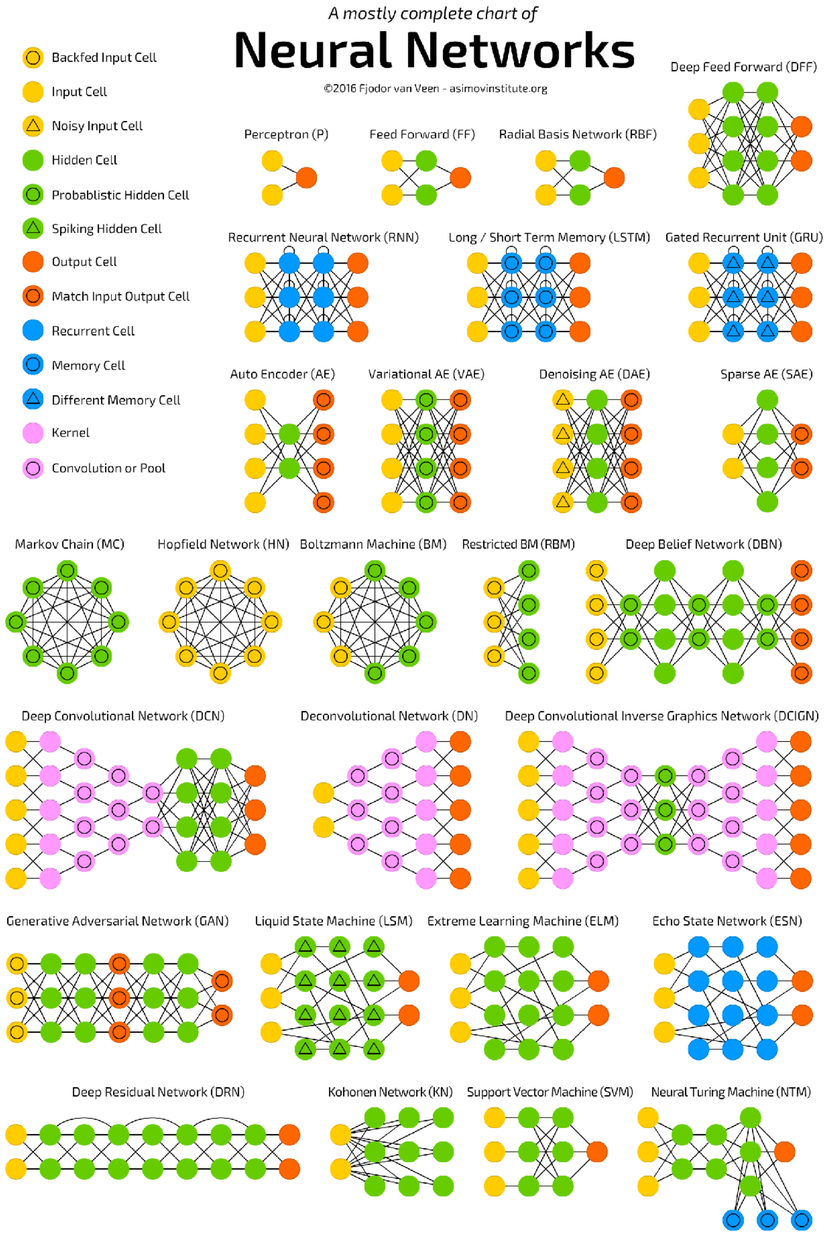
Recommendations are driven by **Machine Learning**

NETFLIX

European Research Council

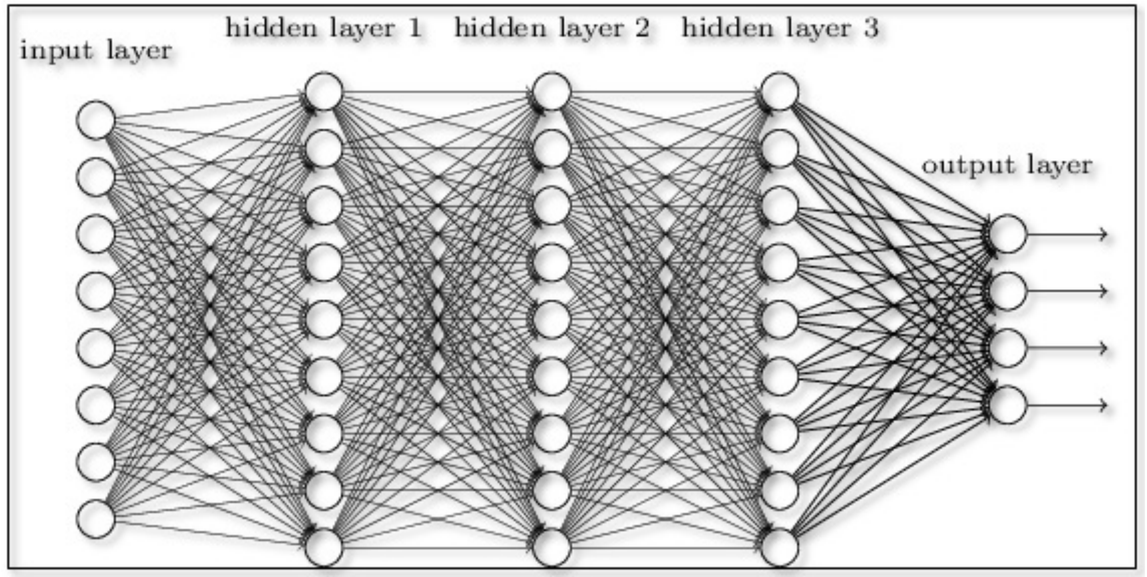
Neural Networks in a nutshell

- You have a task to accomplish, which can be represented as a smooth function from your inputs to the answer you want
  - You could program an algorithm to accomplish the task
  - Or you could train an algorithm to learn an approximation of the "solution function" (machine learning)
- NNs are (as of today) the best ML solution on the market
- NNs are usually structured in nodes connected by edges
  - each node performs a math operation on the input
  - edges determine the flow of neuron's inputs & outputs



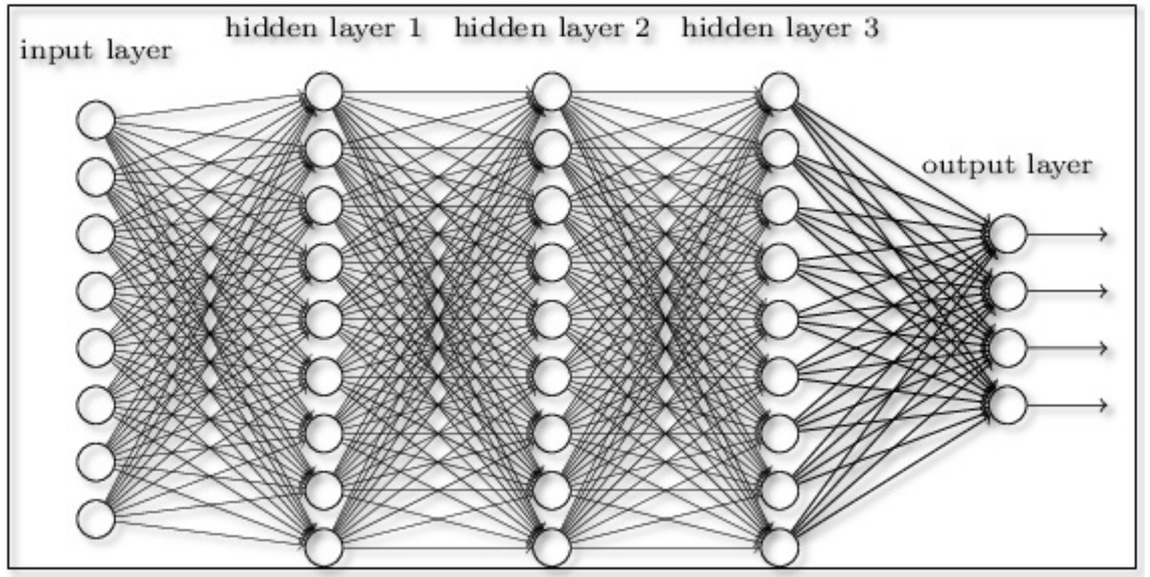
Feed-Forward NNs

- Feed-forward neural networks have hierarchical structures:
  - inputs enter from the left and flow to the right
  - no closed loops or circularities
- Deep neural networks are FF-NN with more than one hidden layer
- Out of this "classic idea, new architectures emerge, optimised for computing vision, language processing, etc



The role of a network node

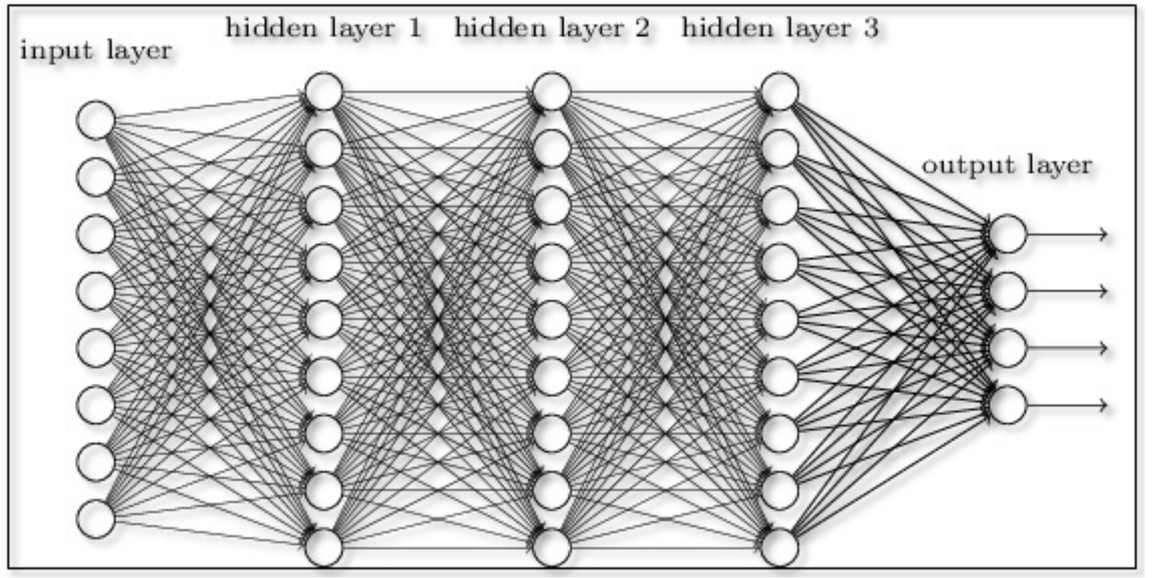
- **Each input is multiplied by a weight**
- The weighted values are summed
- A bias is added
- The result is passed to an activation function



$$w_{ij}x_j$$

The role of a network node

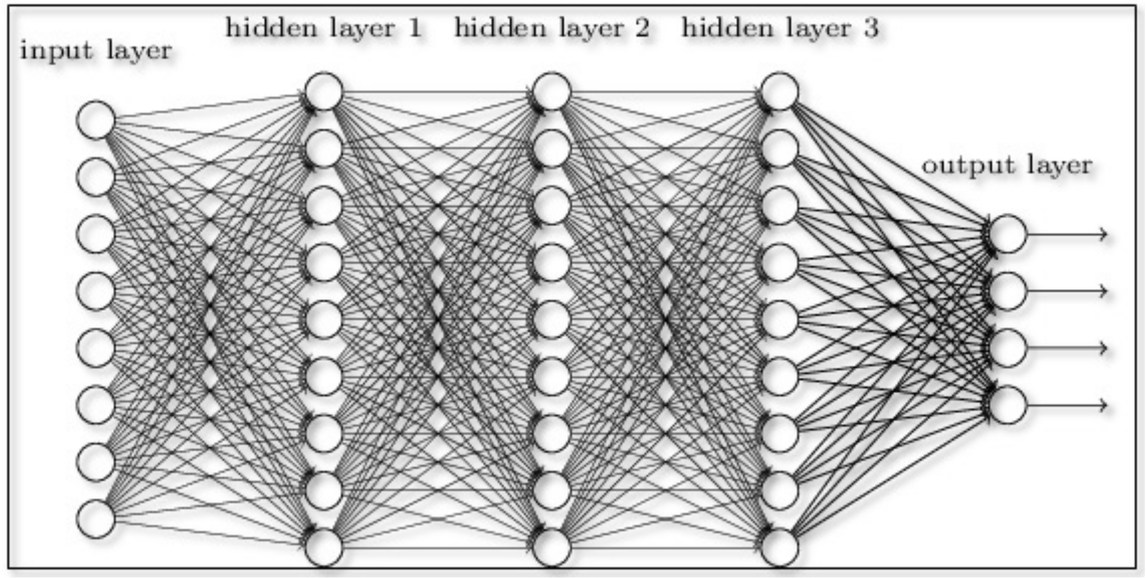
- Each input is multiplied by a weight
- **The weighted values are summed**
- A bias is added
- The result is passed to an activation function



$$\Sigma_j w_{ij} x_j$$

The role of a network node

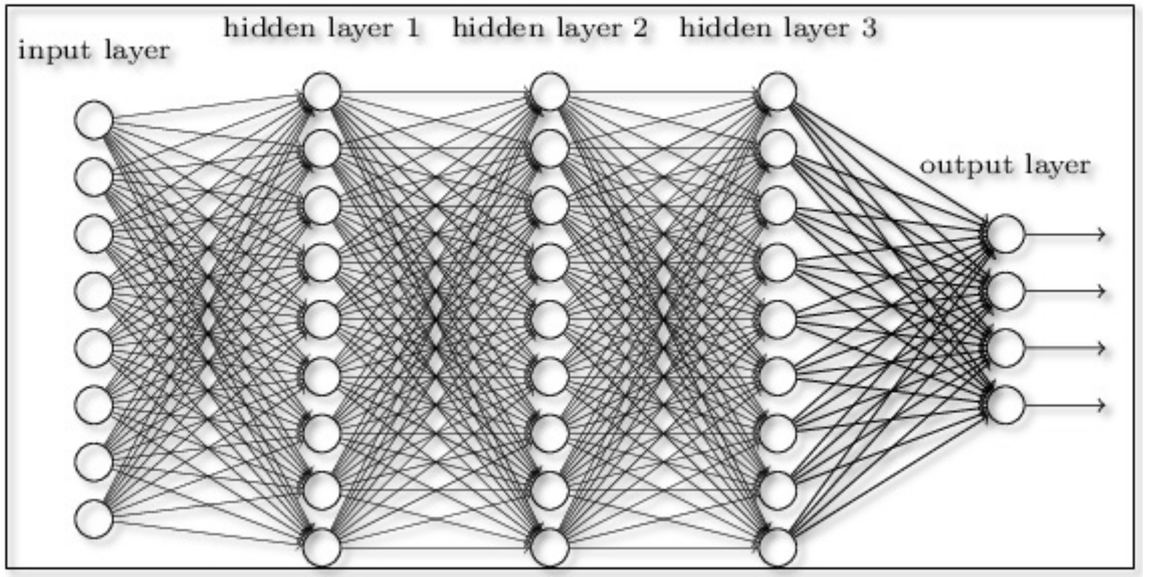
- Each input is multiplied by a weight
- The weighted values are summed
- **A bias is added**
- The result is passed to an activation function



$$\Sigma_j w_{ij} x_j + b_i$$

The role of a network node

- Each input is multiplied by a weight
- The weighted values are summed
- A bias is added
- **The result is passed to an activation function**

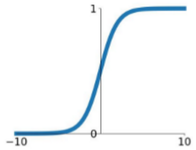


$$y_i = f(\Sigma_j w_{ij} x_j + b_i)$$

Activation Functions

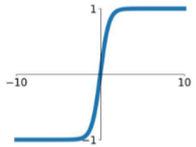
Sigmoid

$\sigma(x) = \frac{1}{1+e^{-x}}$



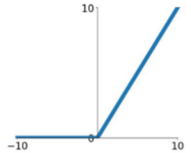
tanh

$\tanh(x)$



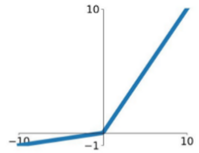
ReLU

$\max(0, x)$



Leaky ReLU

$\max(0.1x, x)$

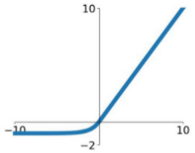


Maxout

$\max(w_1^T x + b_1, w_2^T x + b_2)$

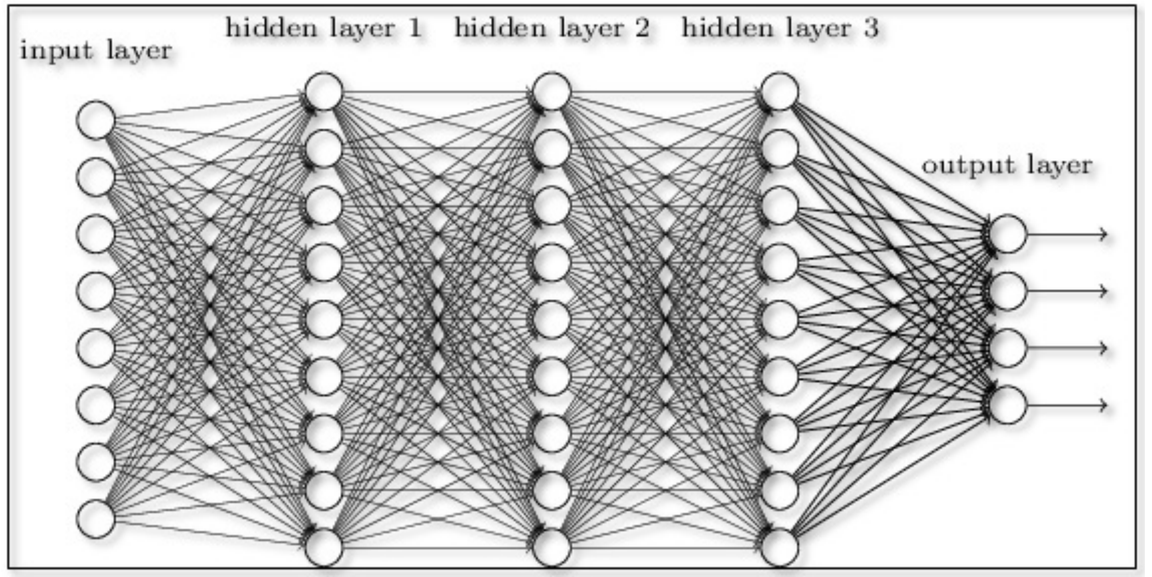
ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



The full picture

- *In a feed-forward chain, each node processes what comes from the previous layer*
- *The final result (depending on the network geometry) is K outputs, given N inputs*

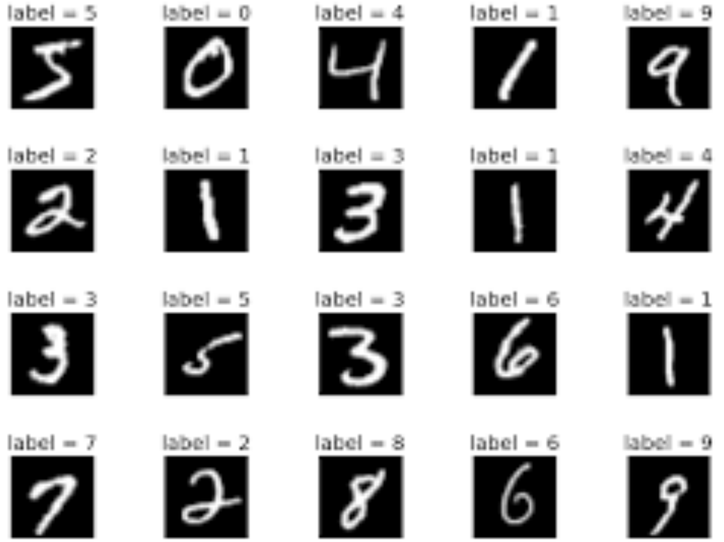


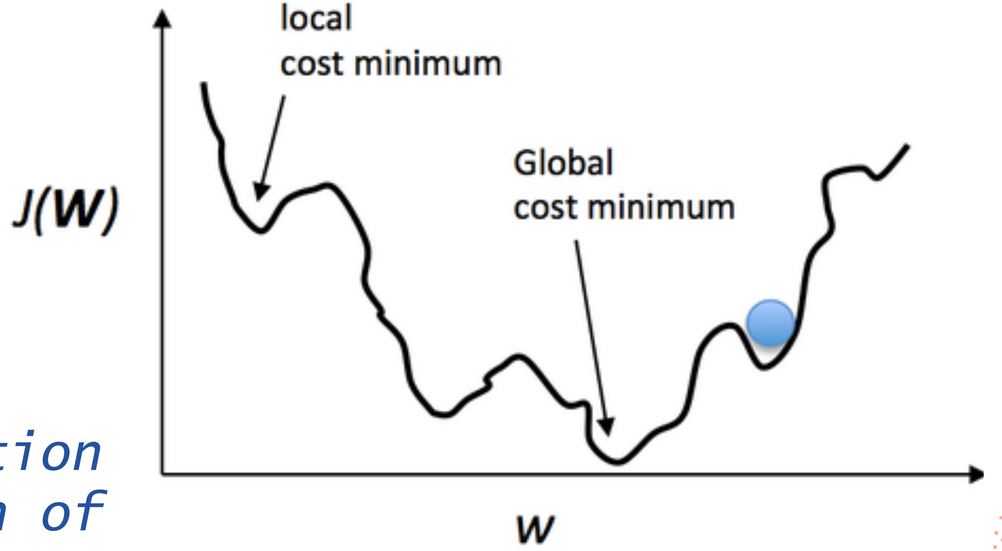
$$y_j = f^{(3)}(\Sigma_l w_{jl}^{(3)} f^{(2)}(\Sigma_k w_{lk}^{(2)} f^{(1)}(\Sigma_i w_{ki}^{(1)} x_i + b_k^{(k)}) + b_l^{(2)}) + b_j^{(3)})$$

- *One can show that such a mechanism allows to learn generic* $\mathbb{R}^N \to \mathbb{R}^K$ *functions*

Training

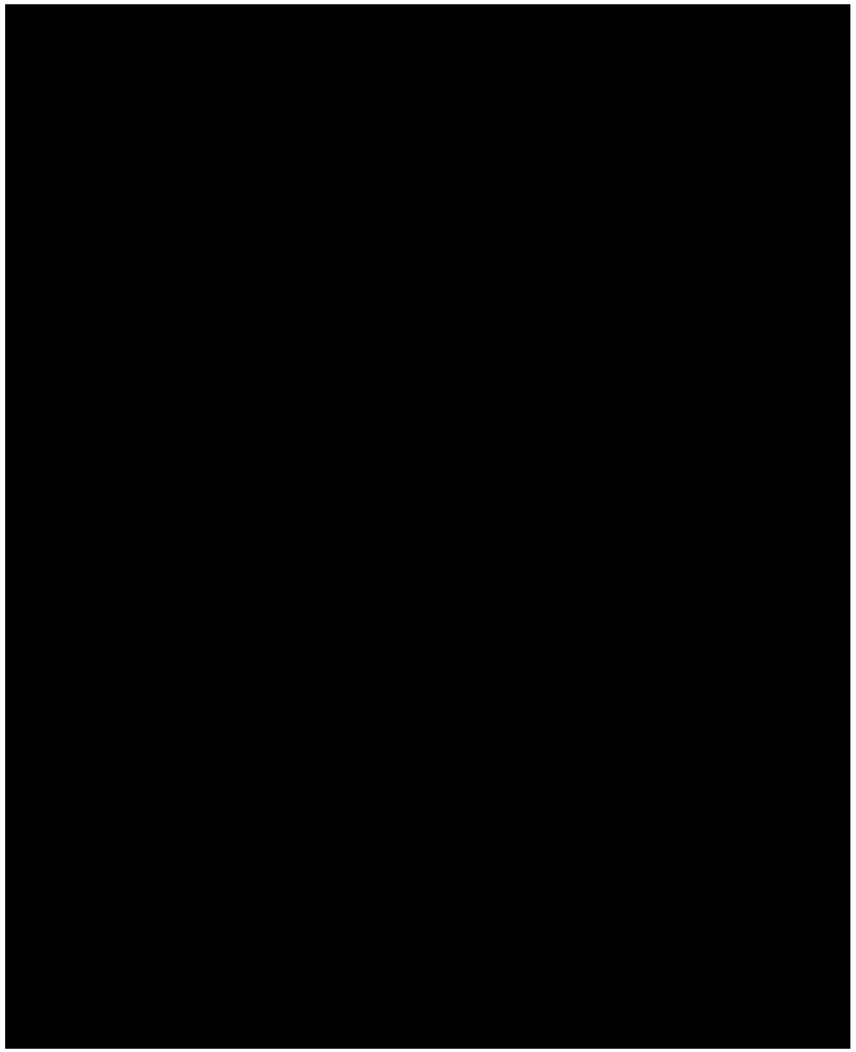
- A network is training specifying an input & a target dataset.
  - For each input example, the target is what the network should learn for that input
  - Target can be a different dataset (regression, classification, …) or the input itself (unsupervised methods)
- Also, a loss function is given:
  - quantification of how many mistakes the network makes
- Training process is the minimisation of the loss function, as function of the network parameters





Does it work? Why?

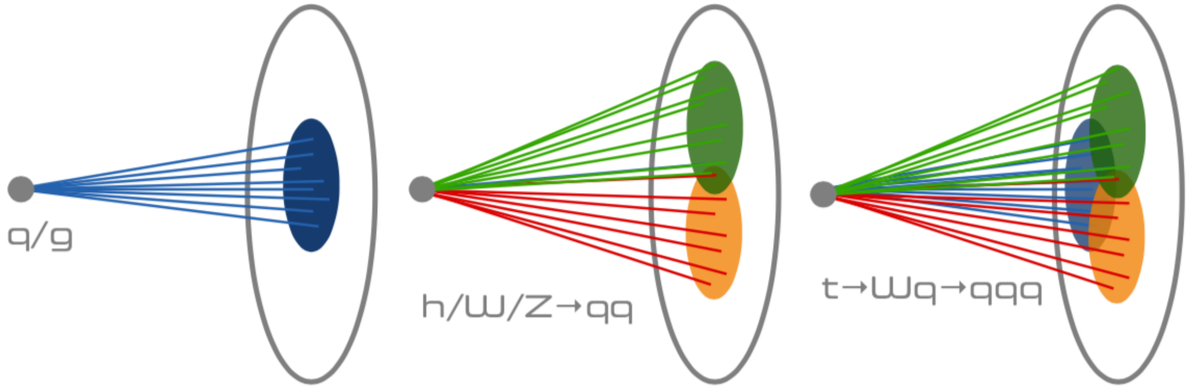
- *The non-linearity of the activation function allows to approximate non-linear functions y = f(x)*
- *The network learns by example to reproduce a result*
- *Can be used as a shortcut of complex algorithms when you have short time to get your answer*
- *The network is NOT intelligent in a human sense: don't ask the network to explain you how it gets the right answer*
- *Still, networks tend to get the right answers (i.e., they are intelligent in the most practical way)*

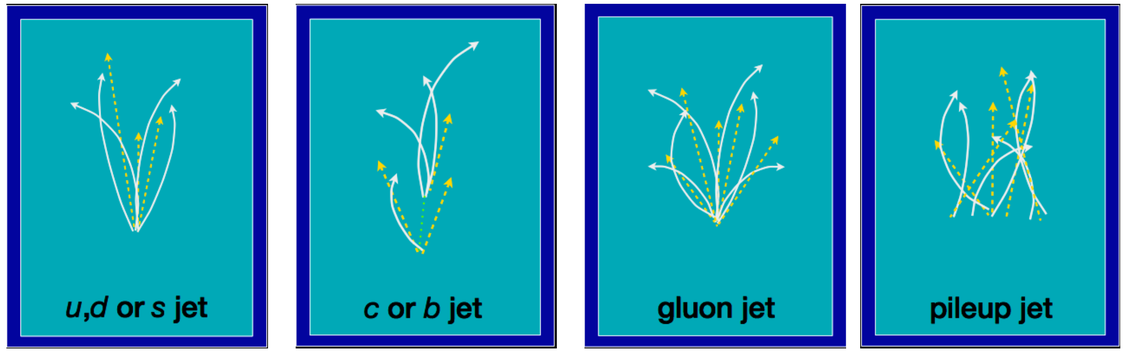


European Research Council

Example: jet tagging

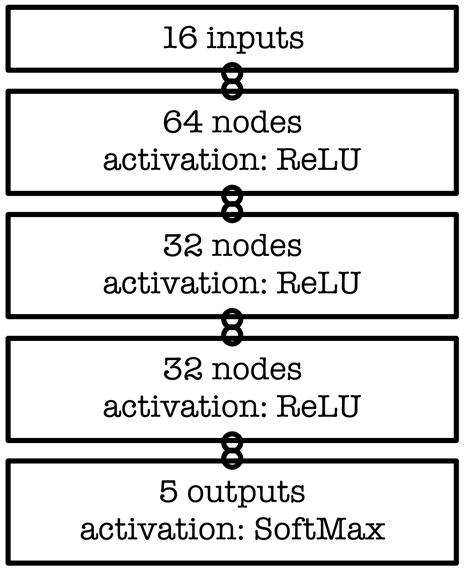
- *You have a jet at LHC: spray of hadrons coming from a "shower" initiated by a fundamental particle of some kind (quark, gluon, W/Z/H bosons, top quark)*
- *You have a set of jet features whose distribution depends on the nature of the initial particle*
- *You can train a network to start from the values of these quantities and guess the nature of your jet*
- *To do this you need a sample for which you know the answer*

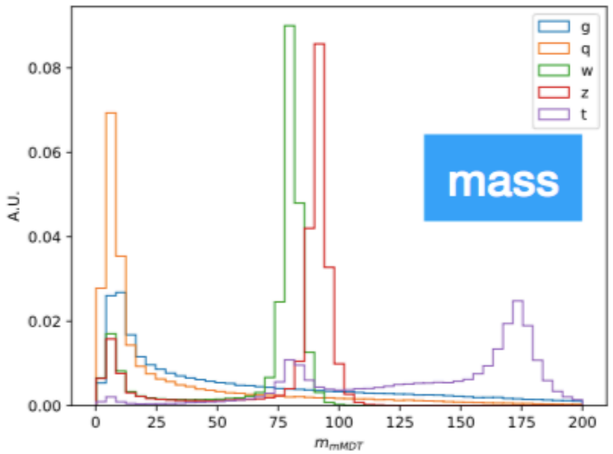


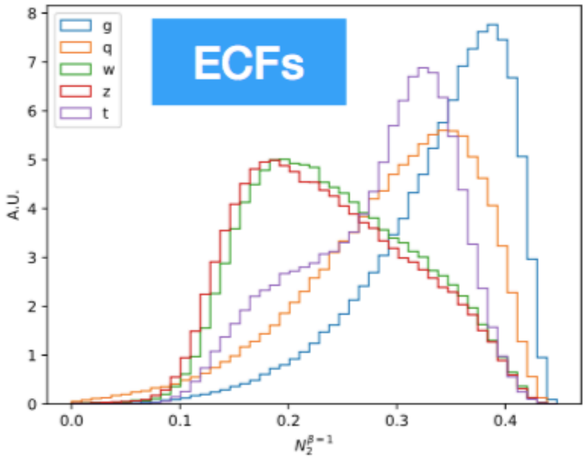


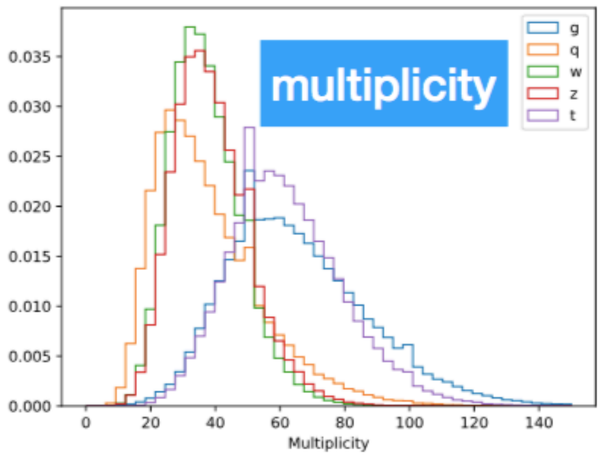
Example: jet tagging

- *You have a jet at LHC: spray of hadrons coming from a "shower" initiated by a fundamental particle of some kind (quark, gluon, W/Z/H bosons, top quark)*
- *You have a set of jet features whose distribution depends on the nature of the initial particle*
- *You can train a network to start from the values of these quantities and guess the nature of your jet*
- *To do this you need a sample for which you know the answer*

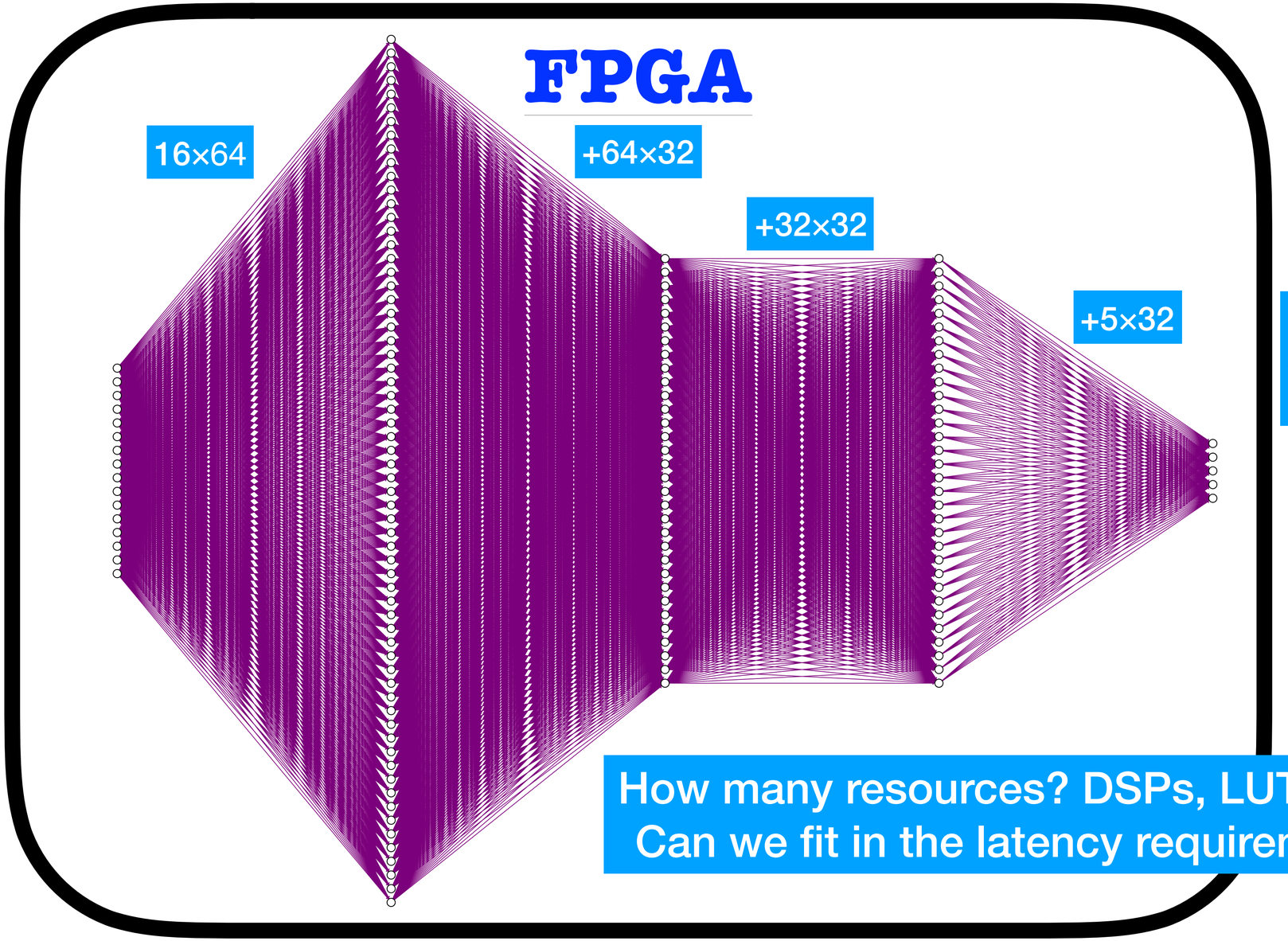




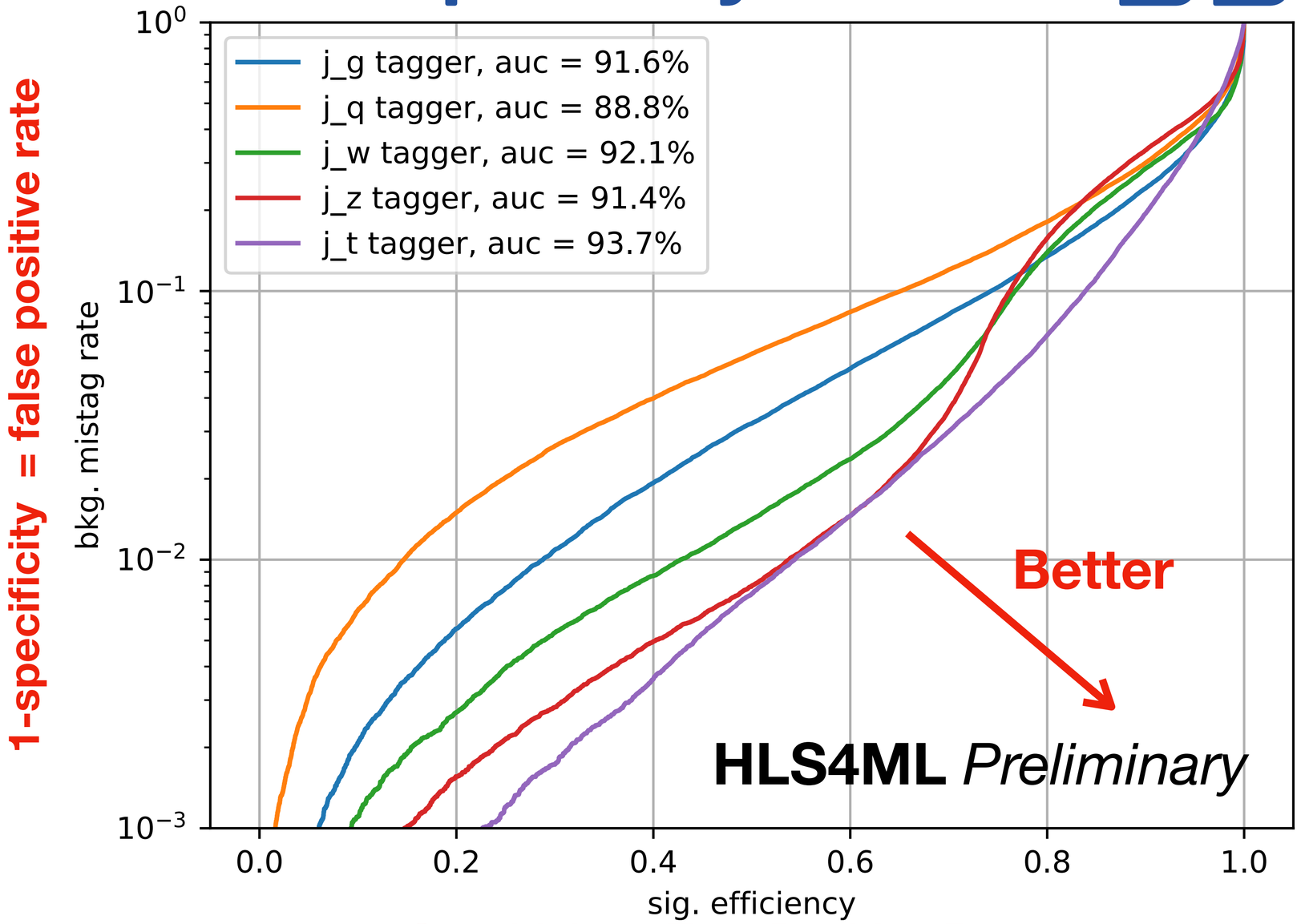




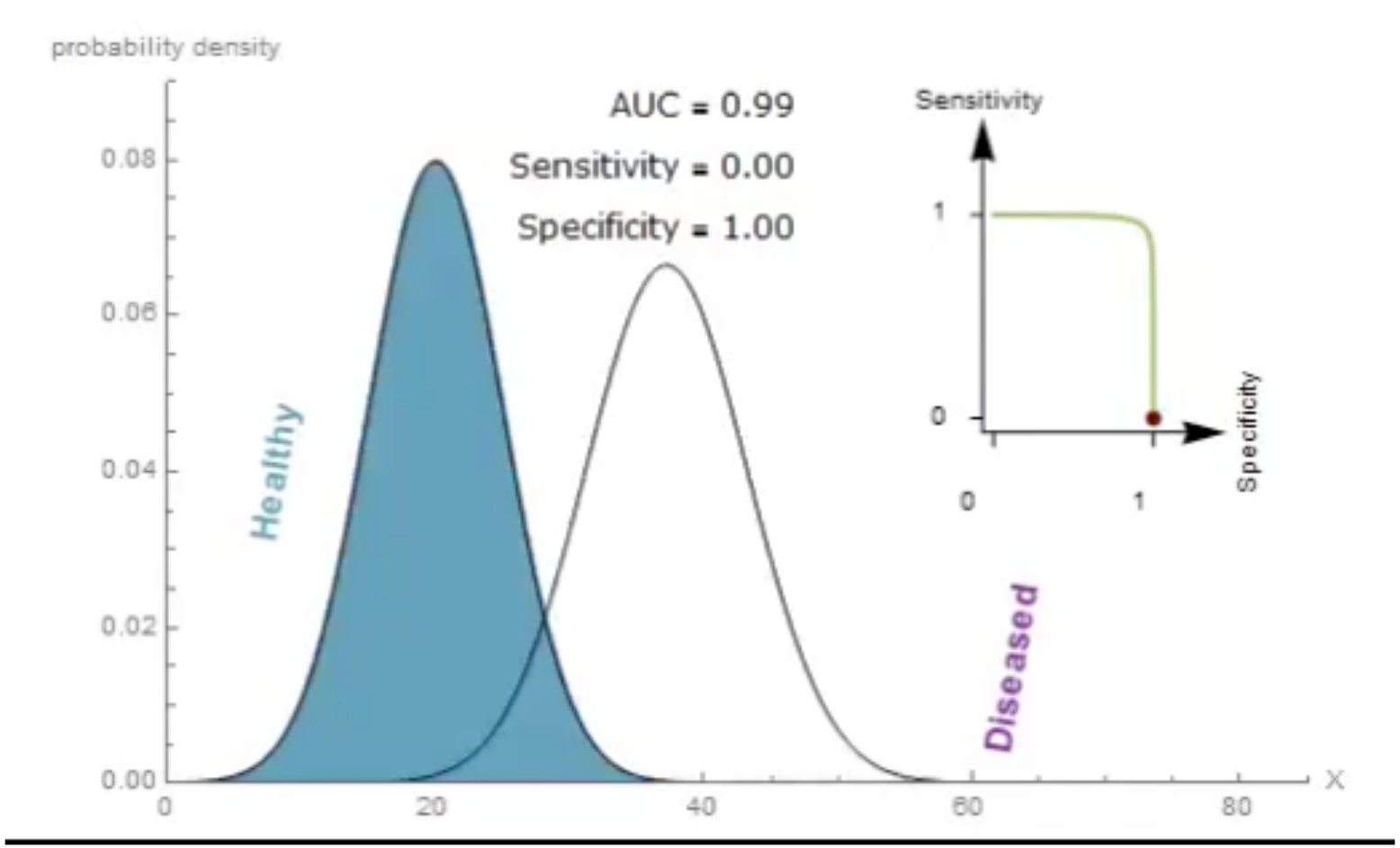
Example: jet tagging

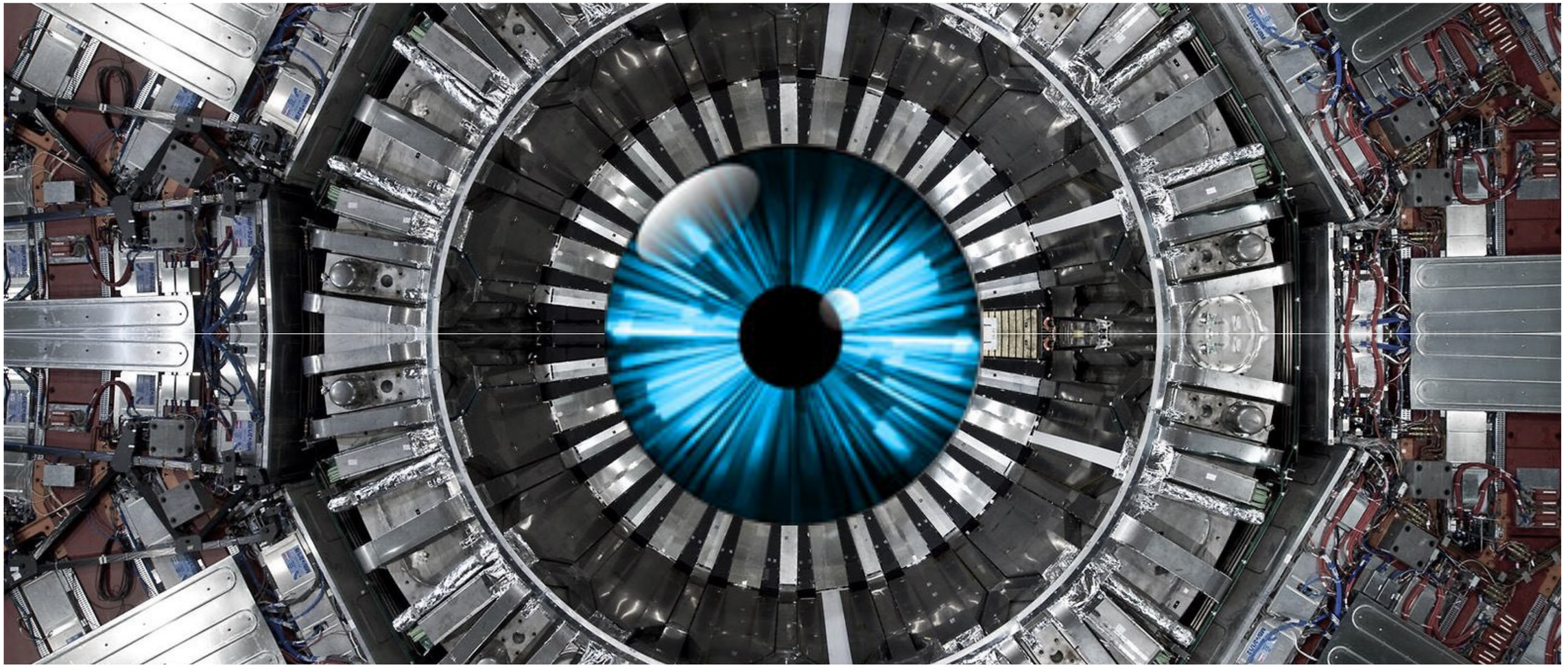


Example: jet tagging



What is a ROC curve?



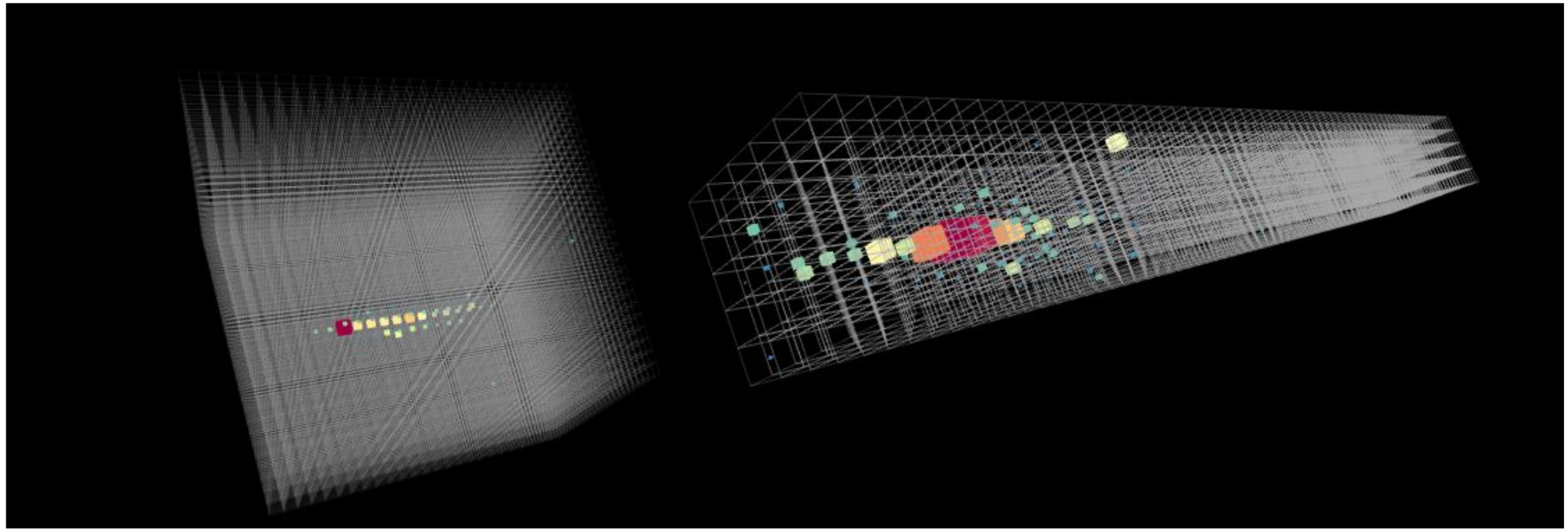


Particle Reconstruction & Computer Vision

European Research Council

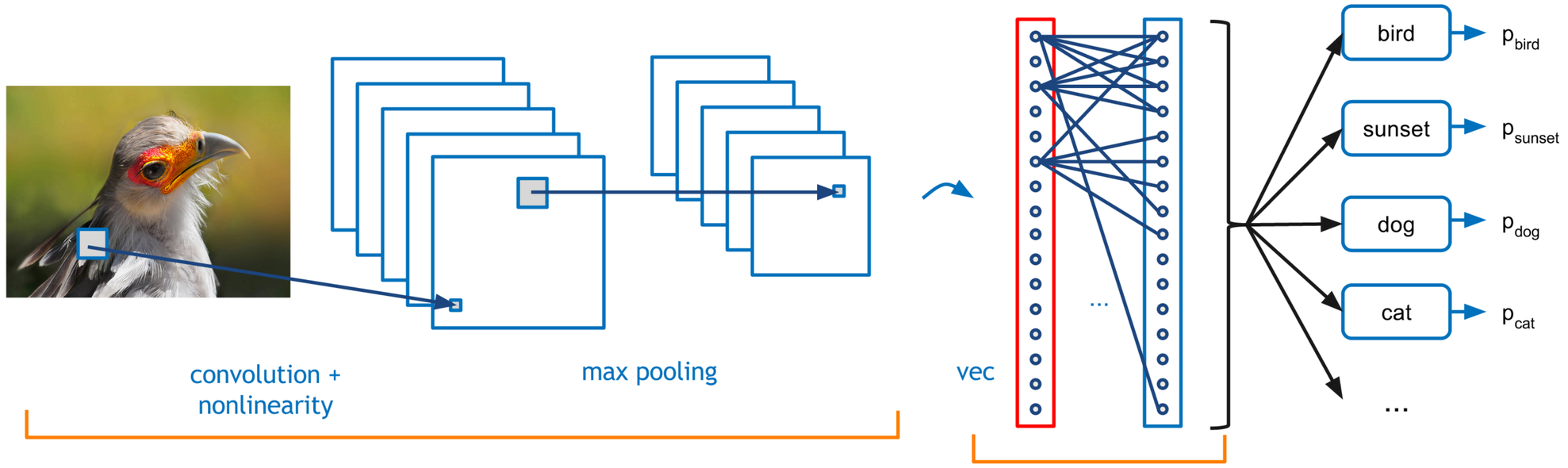
Calorimetry & Computer Vision

- *(next generation) digital calorimeters: 3D arrays of sensors with more regular geometry*
- *Ideal configuration to apply Convolutional Neural Network*
  - *speed up reconstruction at similar performances*
  - *and possibly improve performances*

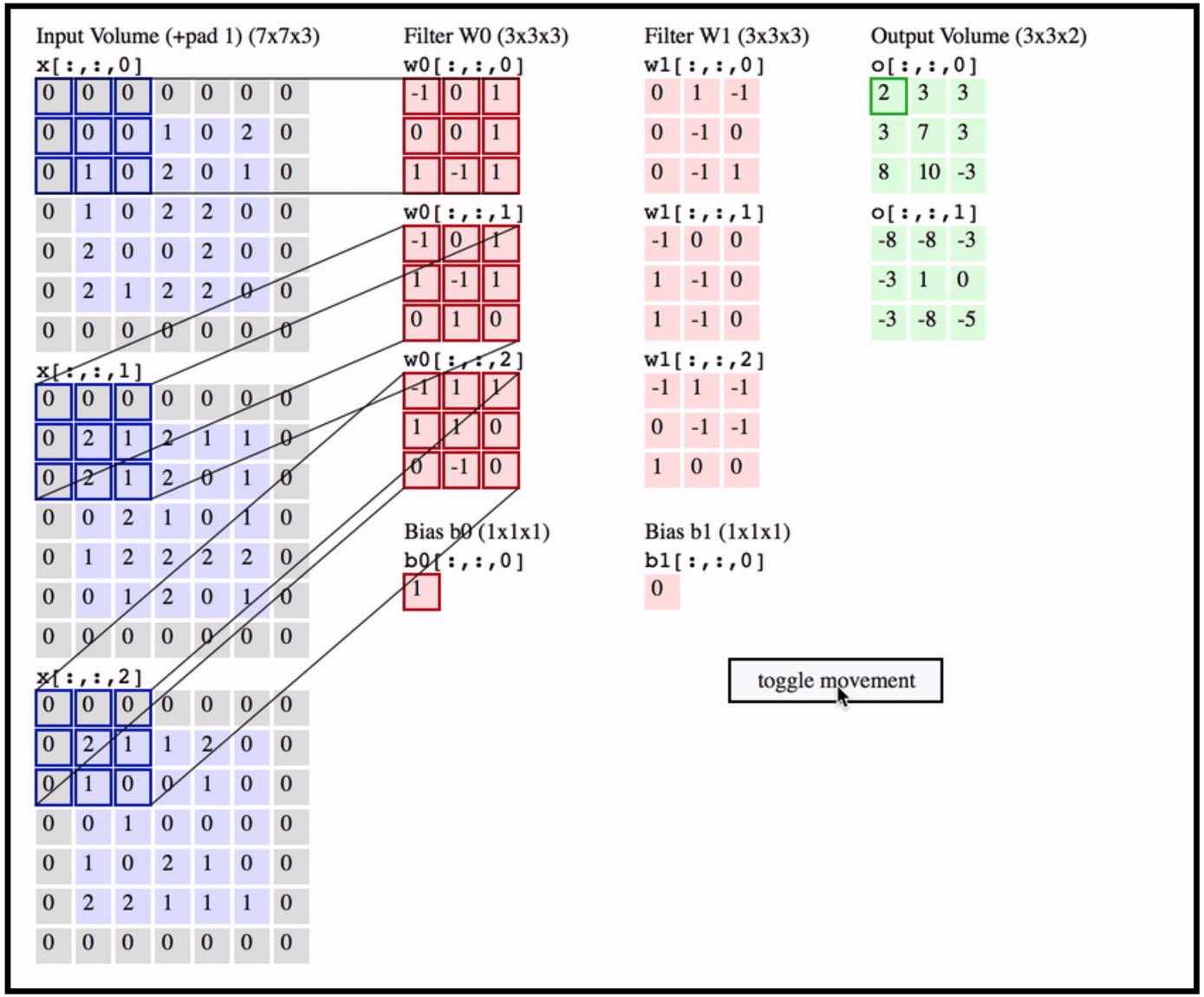


Convolutional Layer

- *Special architectures read the raw information (e.g., images) and convert them into "smart variables" (high-level features) to accomplish the task*
- *Typical example: convolutional neural networks for image processing & computing vision*

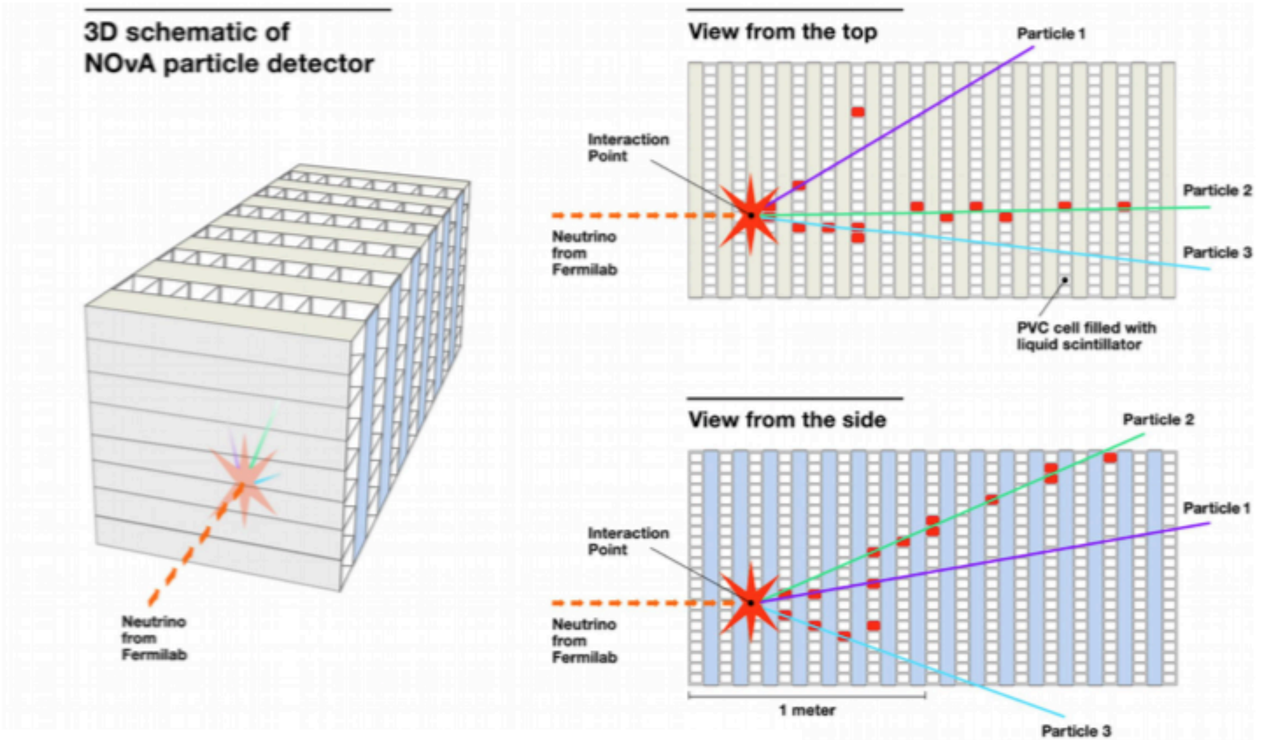


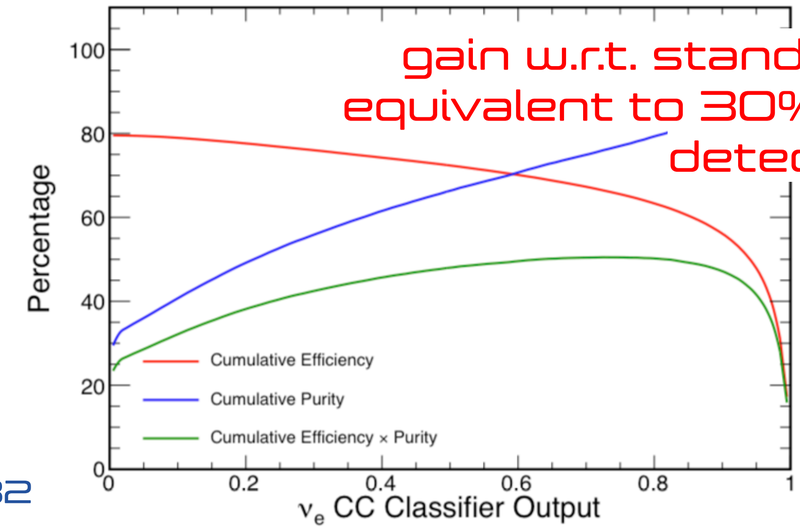
Convolutional Layer



Example: PID for ν experiments

- *Many HEP detectors (particularly underground) more and more structured as regular arrays of sensors*
- *Modern computer-vision techniques work with images as arrays of pixel sensor (in 1D, 2D, and 3D)*
- *These techniques were applied by Nova on electron and muon ID*
- *Impressive gain over traditional techniques (comparable to +30% detector == $$$ saved)*

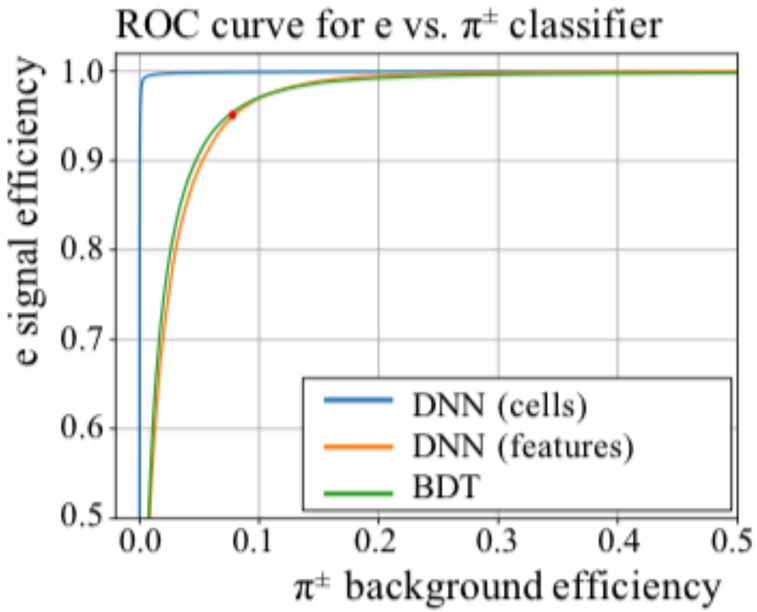


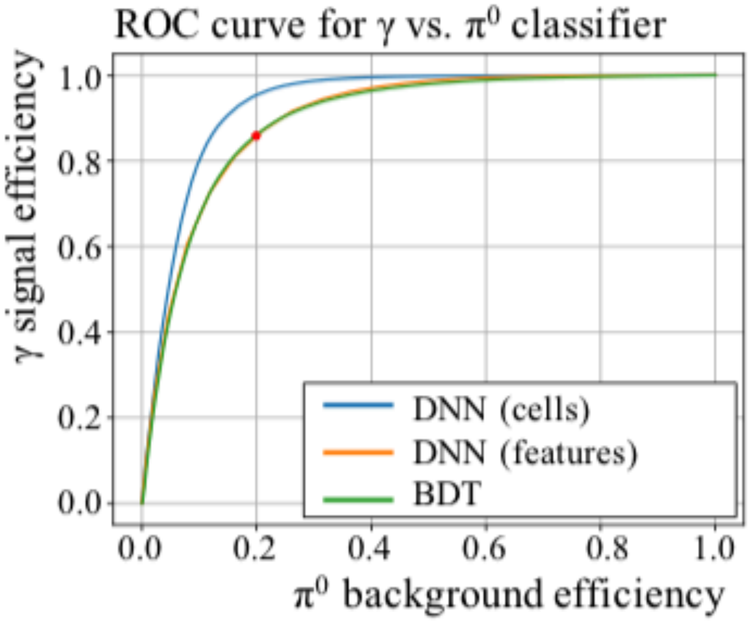


gain w.r.t. standard approach:
equivalent to 30% bigger mass of detector

Example: Particle ID

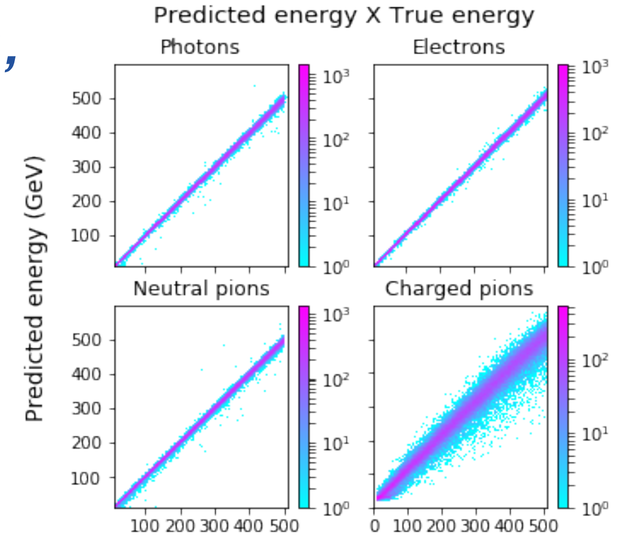
- *We tried particle ID on a sample of simulated events*
  - *one particle/event (e, γ, π^0, π)*
- *Different event representations*
  - *high-level features related to event shape (moments of X,Y, and Z projections, etc)*
  - *raw data (energy recorded in each cell)*
- *Pre-filtered pion events to select the nasty ones and make the problem harder*

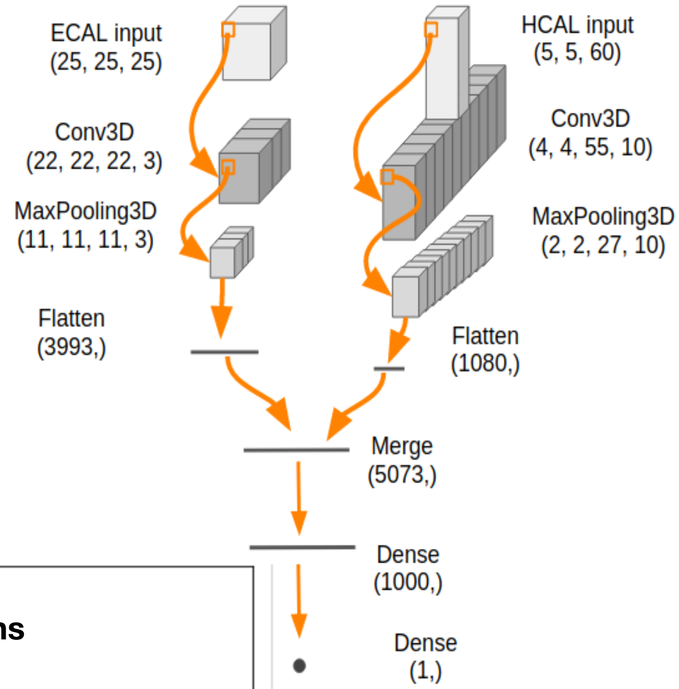


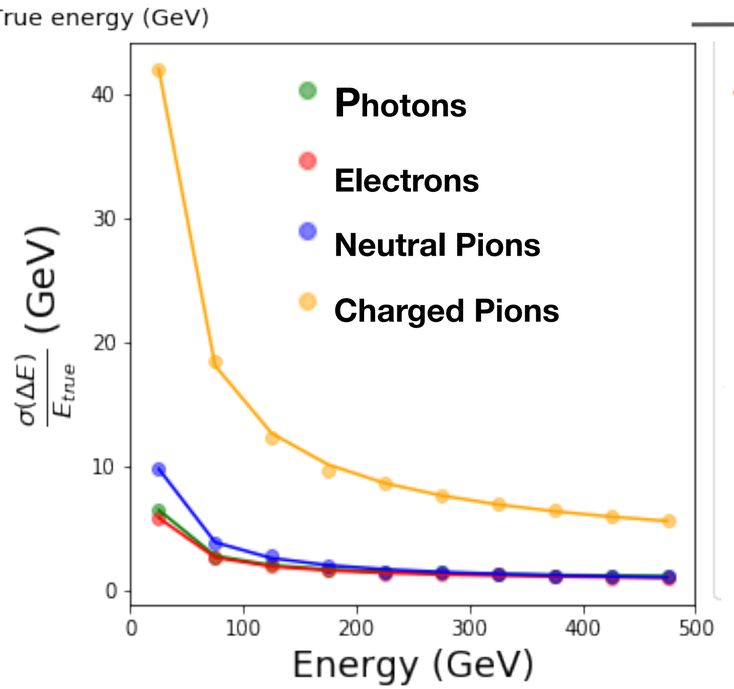


Example: Energy Regression

- *Correctly reconstruct energy, with physics meaningful performances*
  - *ECAL performances better than HCAL (as expected)*
  - *π^0 resolution ~ $\sqrt{2}$ γ resolution (as expected)*
- *FAST: used only RAW data as inputs -> no pre-processing*
  - *Processing time reduced by 10^3 wrt traditional approaches*
  - *Potentially usable both online and offline*

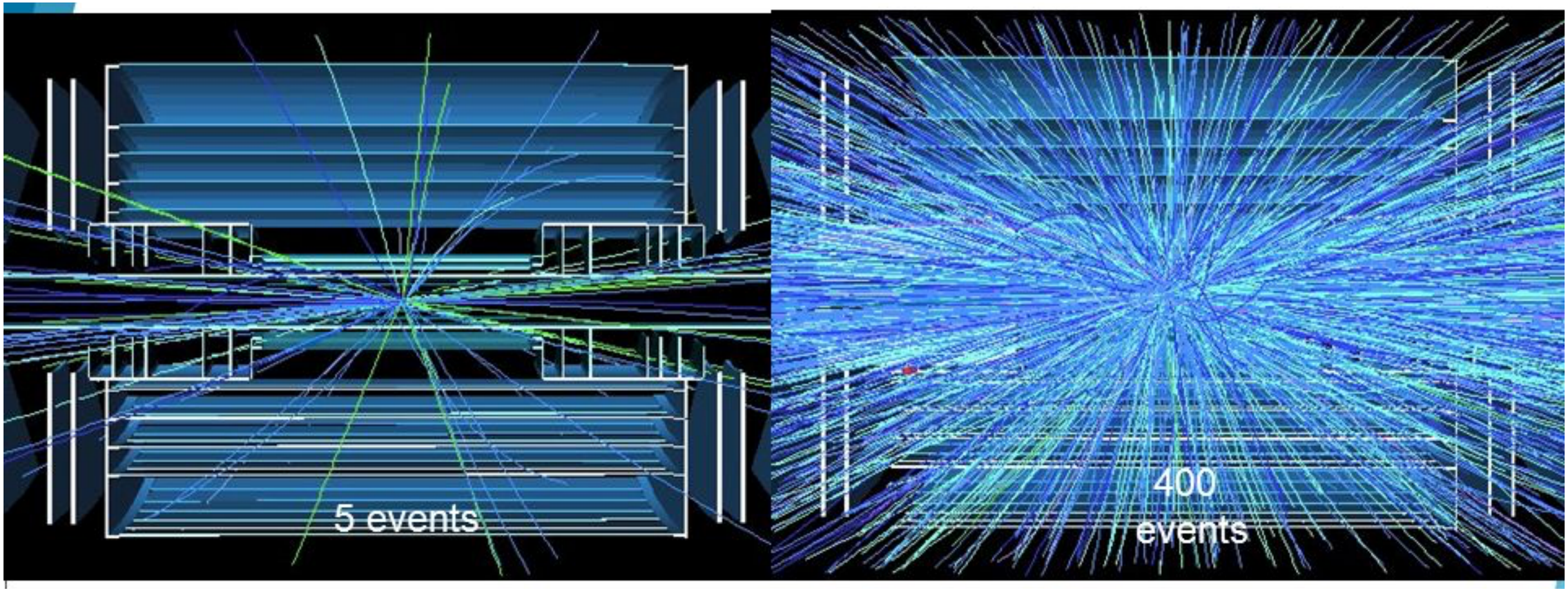






$\frac{\sigma(\Delta E)}{E_{true}}$ (GeV)

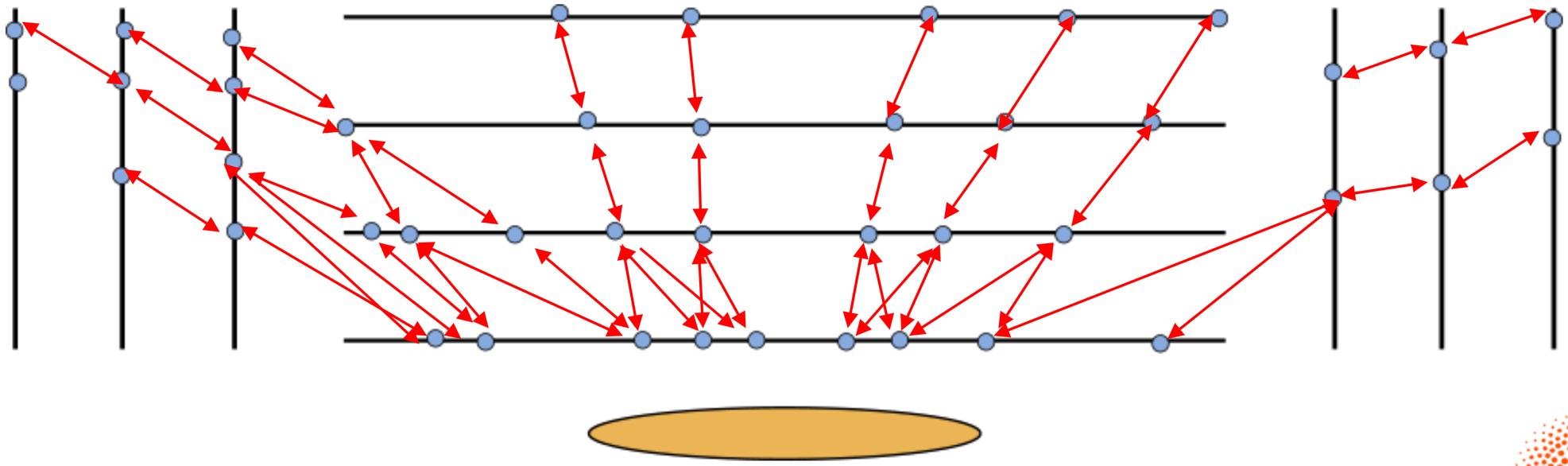
Combinatorics Reduction



- *Tracking is the most expensive workflow we have in RECO*
- *The more tracks we have, the more problematic it becomes (non-linearity due to combinatoric when connecting dots*

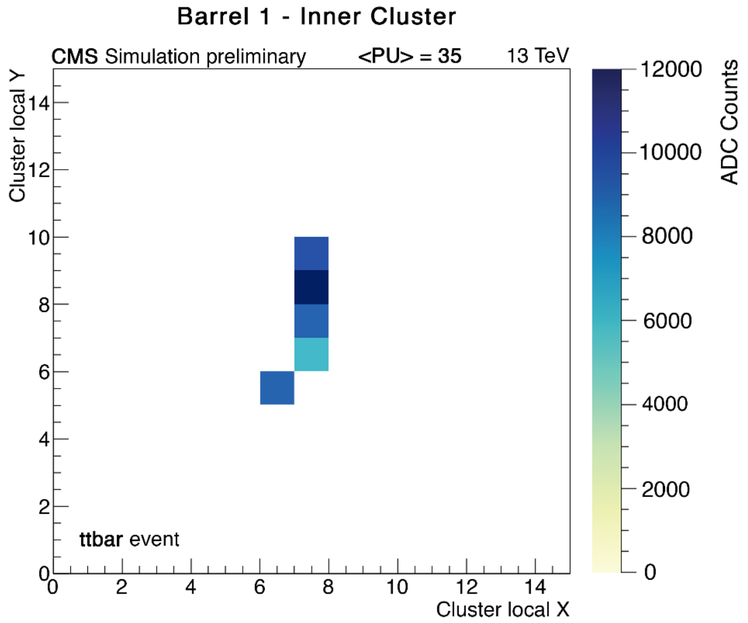
Combinatorics Reduction

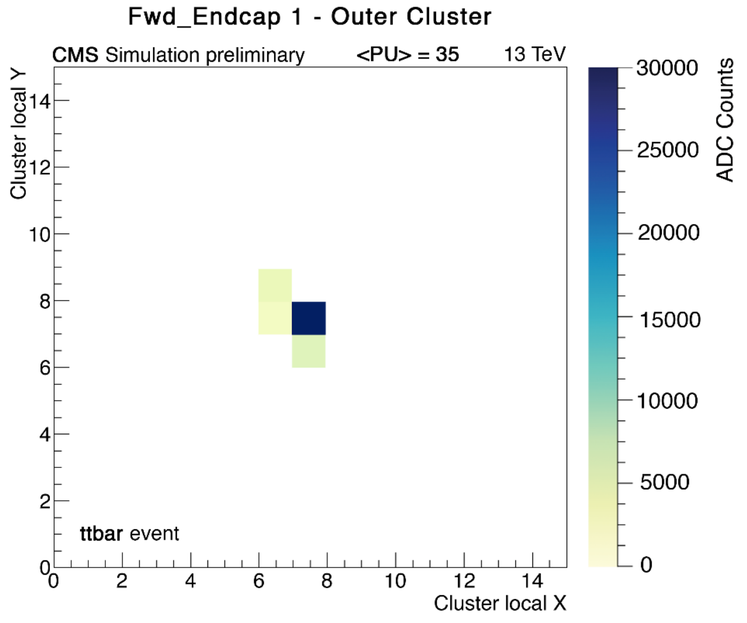
- *Track reco starts with a see in the inner layers (triplet of hits)*
- *Finding seeds is not the most CPU intensive aspect*
- *But its outcome determines the complexity of the following steps*
  - *one can speed up tracking by reducing the number of fake seeds*
- *We tried to solve this problem using a ConvNN*



Seeds as images

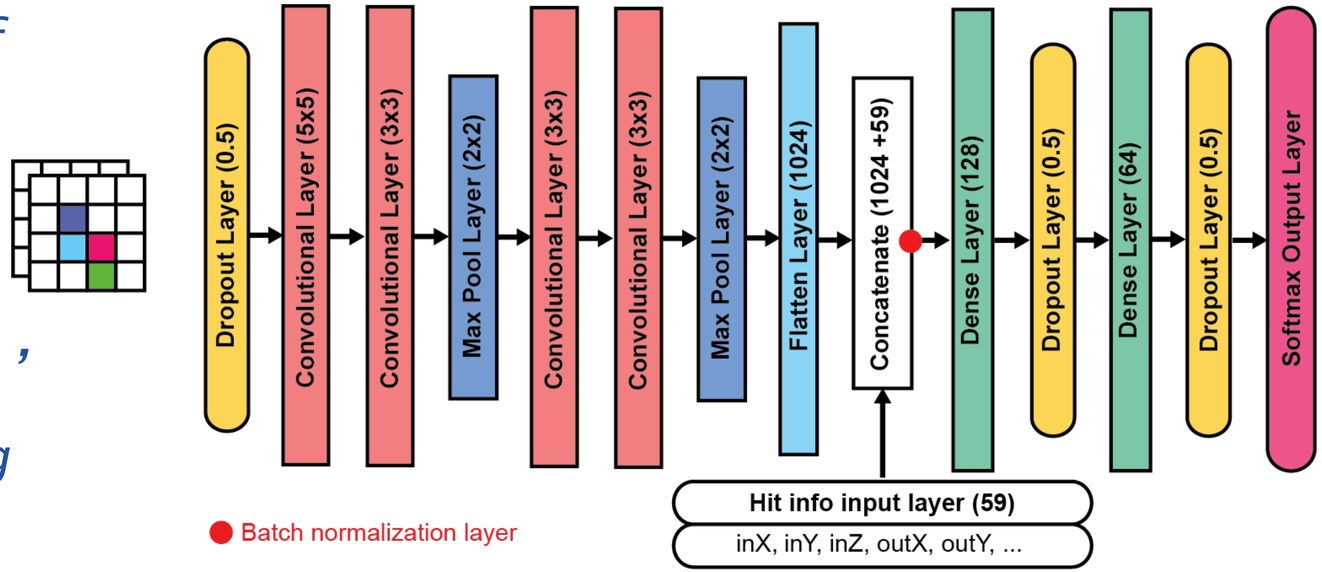
- *The detector sees the charge deposited by the crossing particle: a hit*
- *A hit is a window of sensors (16x16 here) with its deposited charge. This can be seen as a sparse digital image.*
- *Given two images, one can train a network to decide if a pair of hits is a good or bad match*



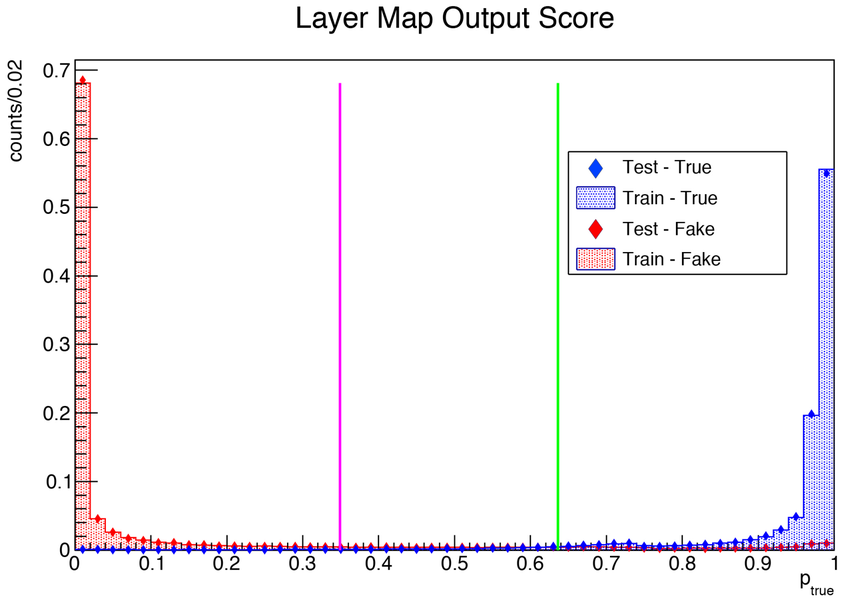


PixelSeed ConvNN

- *The final model uses two sets of inputs:*
  - *the hit images*
  - *a set of expert features (e.g., position of the hits in the detector) to help the learning process*

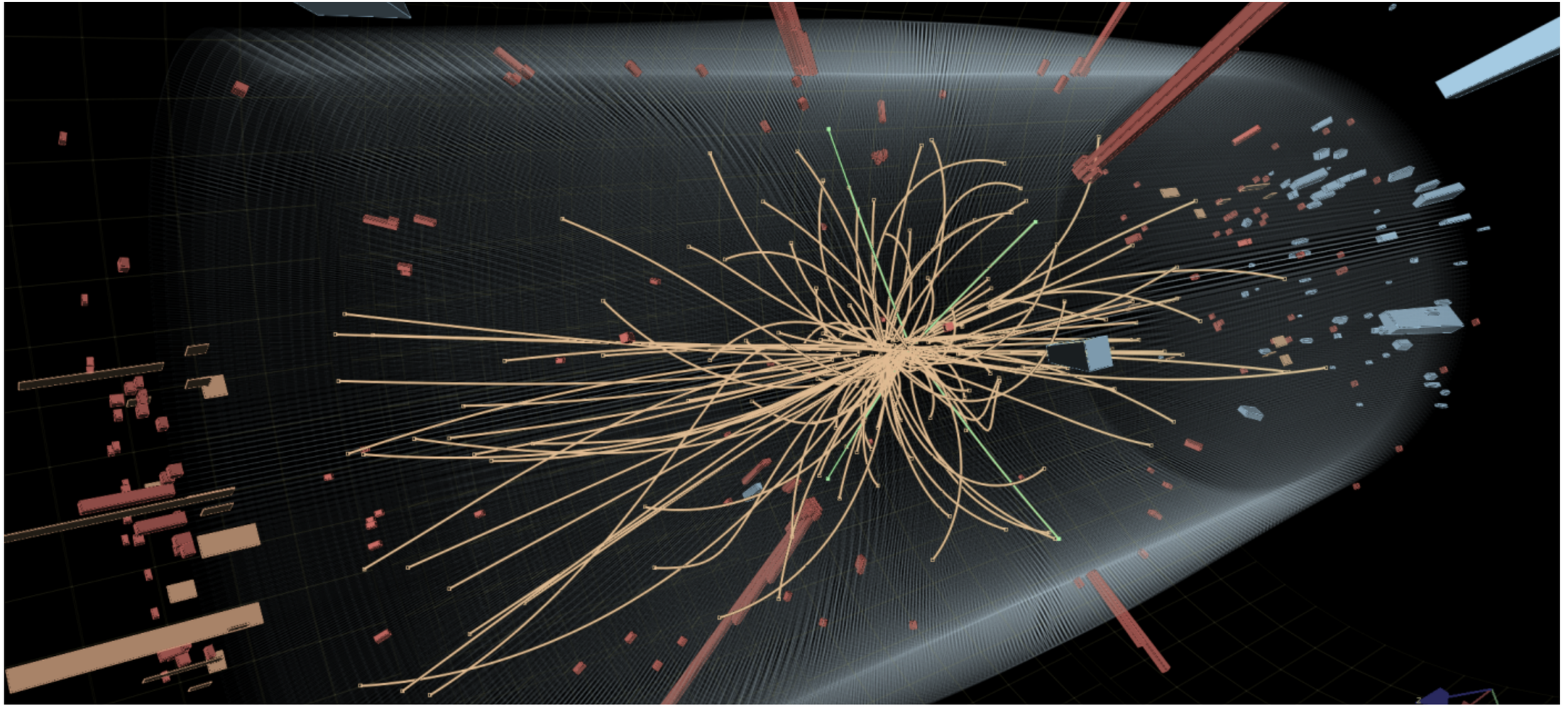


- *The trained model shows a good separation of true vs fake seeds*



- ***One can reduce the fake rate by one order of magnitude with a few % loss in efficiency***

```
Efficiency (tpr) @ fake rejection
tpr @ rej 50%: 0.998996700259
tpr @ rej 75%: 0.990524391331
tpr @ rej 90%: 0.922210826719
tpr @ rej 99%: 0.338669401587
```

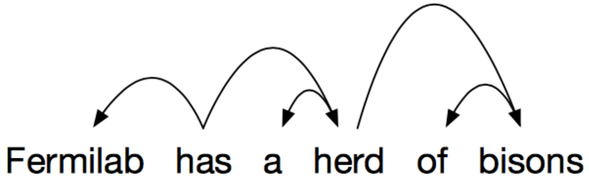


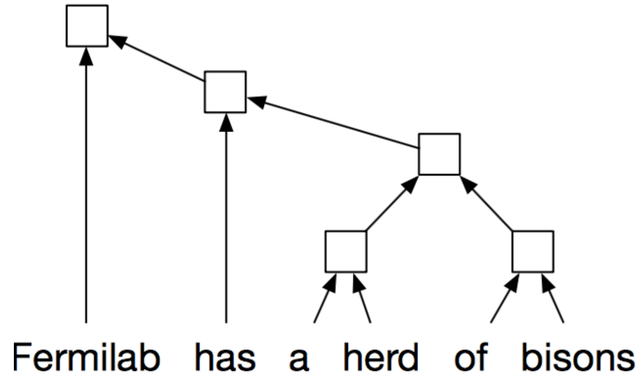
HEP & Language processing networks

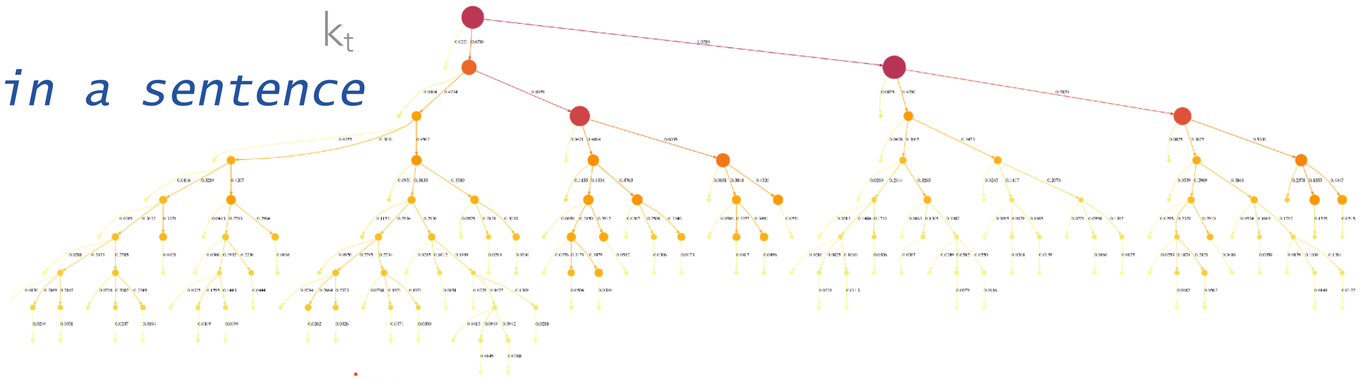
European Research Council

Particle Flow & language processing

- *CMS uses particle flow for event reconstruction:*
  - *At some point in the central processing, collision images are turned into a list of particles.*
  - *From these particles, complex objects (e.g., jets) are formed*
- *In this framework, Computing vision approaches are not necessarily ideal*
- *One can instead use language-processing approaches (e.g., recurrent neural networks*
  - *particles are words in a sentence*
  - *QCD is the grammar*

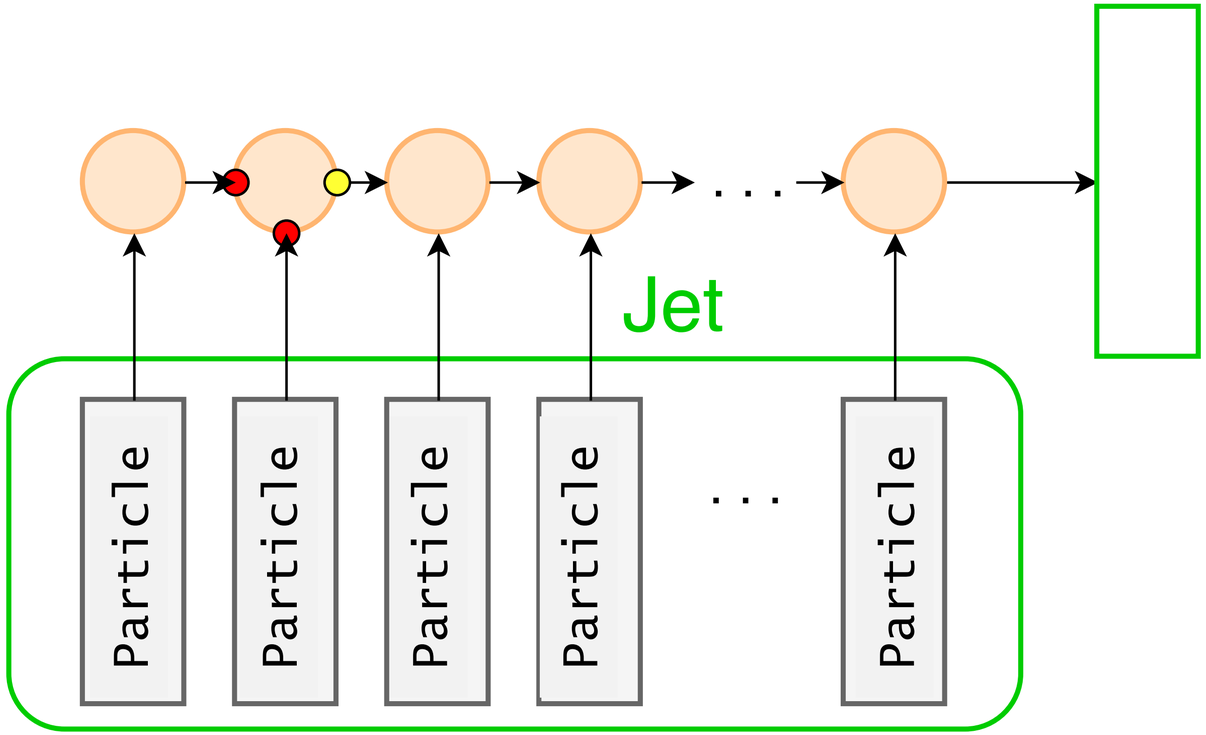






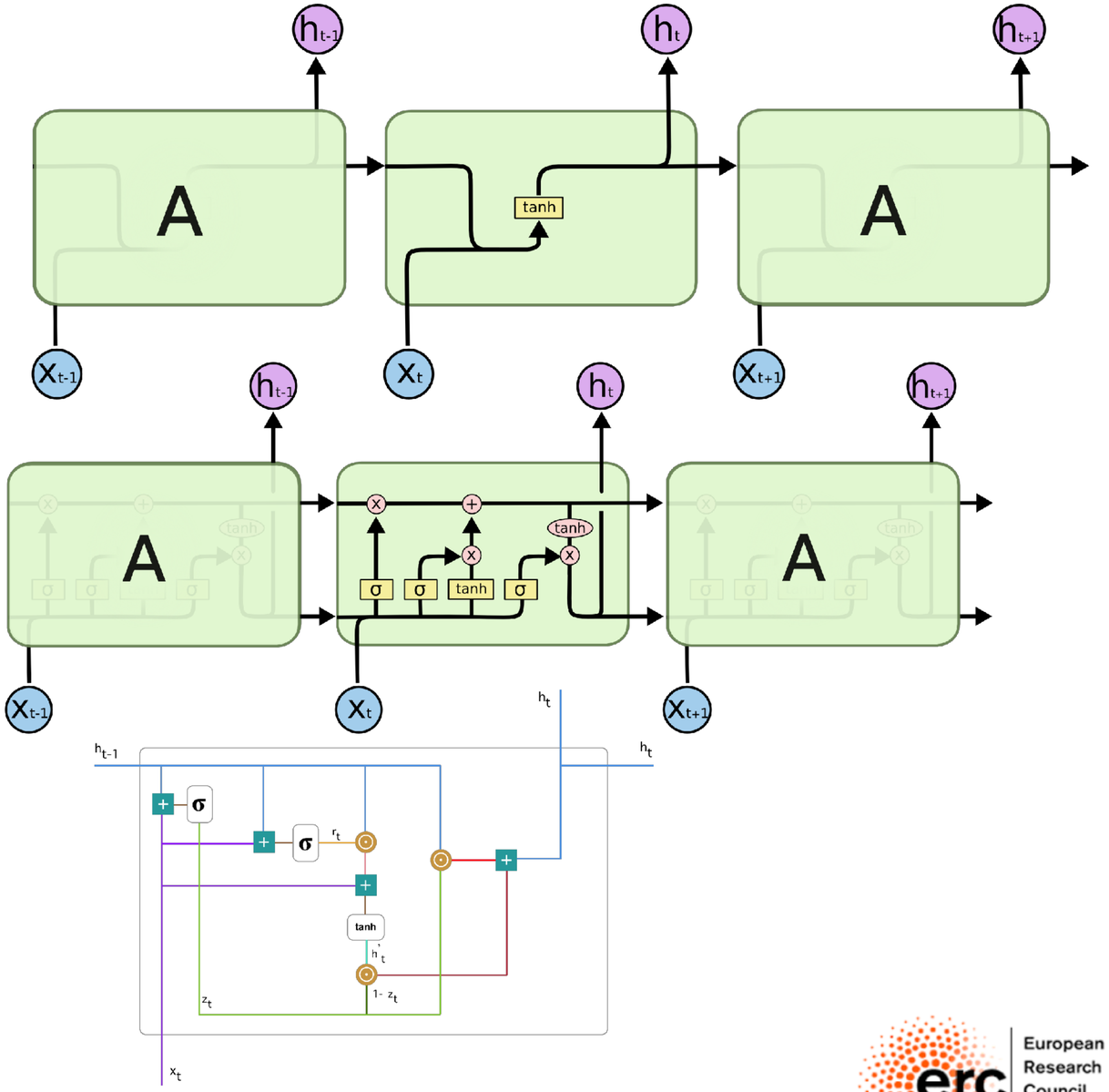
Recurrent Neural Networks

- A network architecture suitable to process an ordered sequence of inputs
  - words in text processing
  - a time series
  - particles in a list
- Could be used for a single jet or the full event
- Next step: graph networks (active research direction)



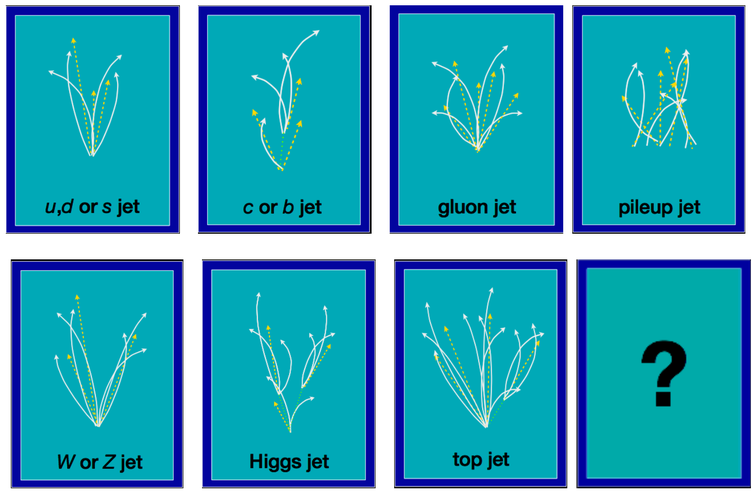
RNN architectures

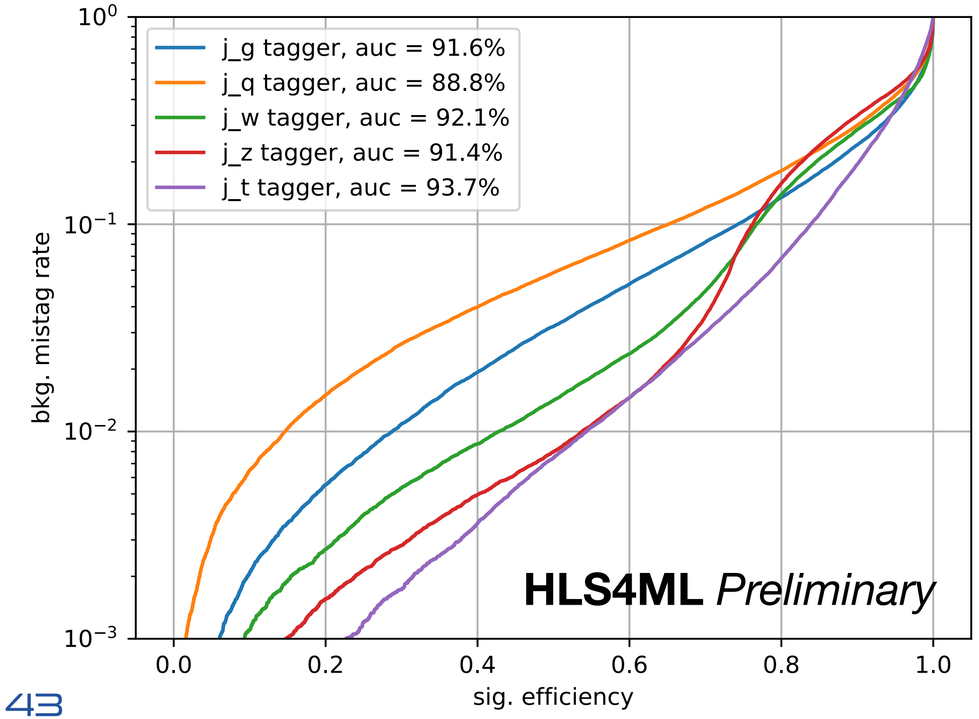
- *Plain RNNs have one-layer processing units*
- *More complicated architectures with multiple layer processing introduced to mimic long-term memory*
  - *Long-Short Term Memory (LSTM) networks*
  - *Gated Recurrent Unit (GRU) networks*
- *These complex architecture usually give better performances*



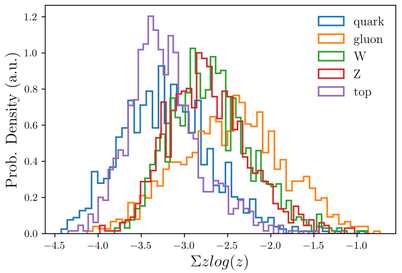
A comparison: jet tagging

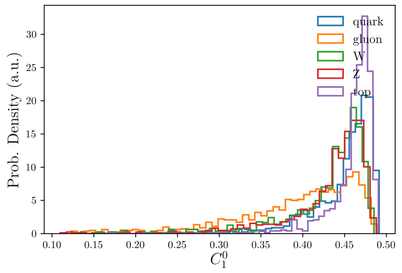
- *Back to the initial jet-tagging problem*
- *Now use different jet representations*
  - *High level features*
  - *Images: useful when your jets are made of calotowers*
  - *Sequences: useful when your jets are lists of particles (e.g., with PF)*

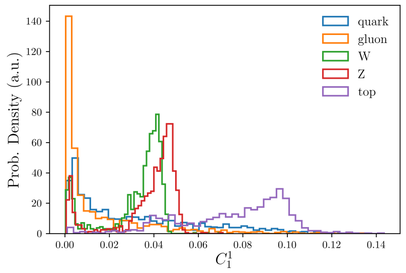


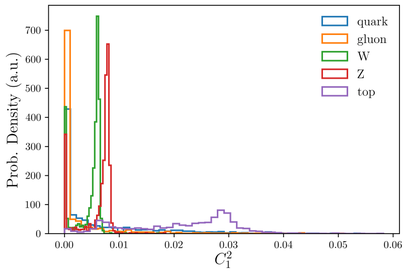


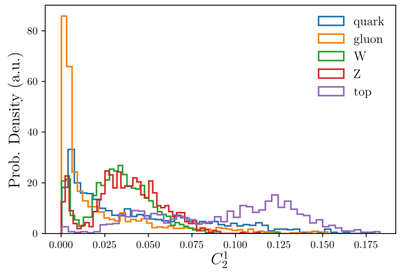
Jets features

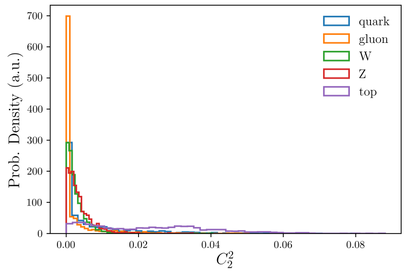


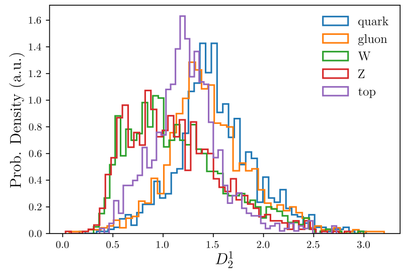


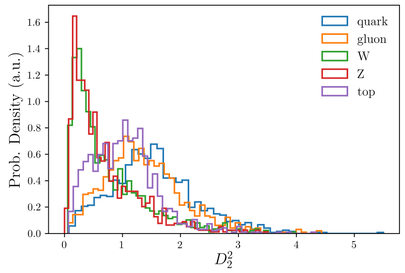


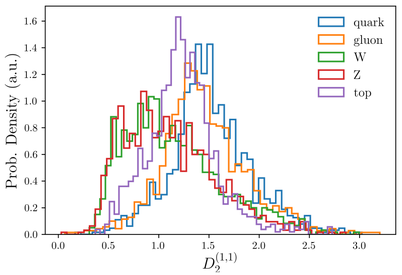


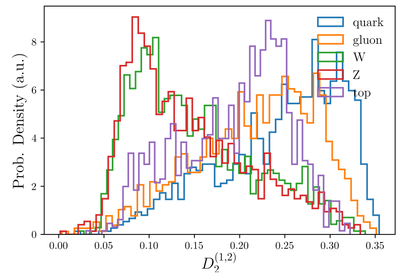


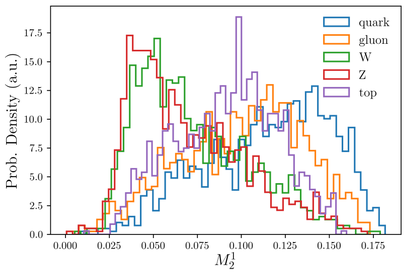


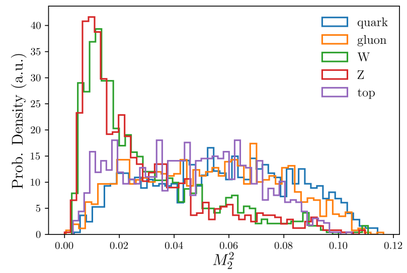


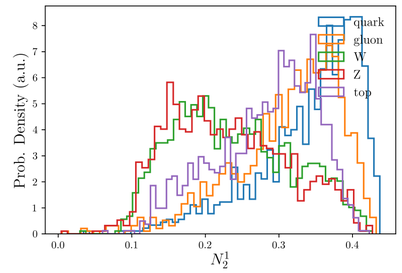


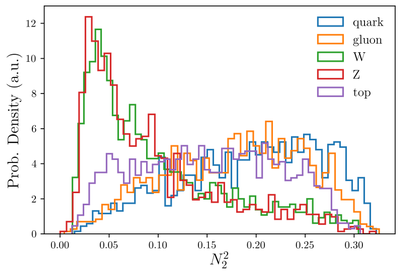


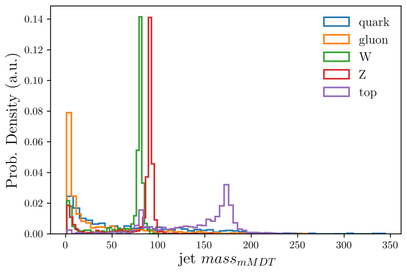


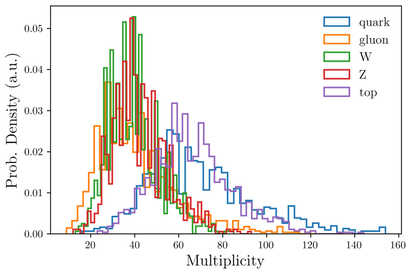




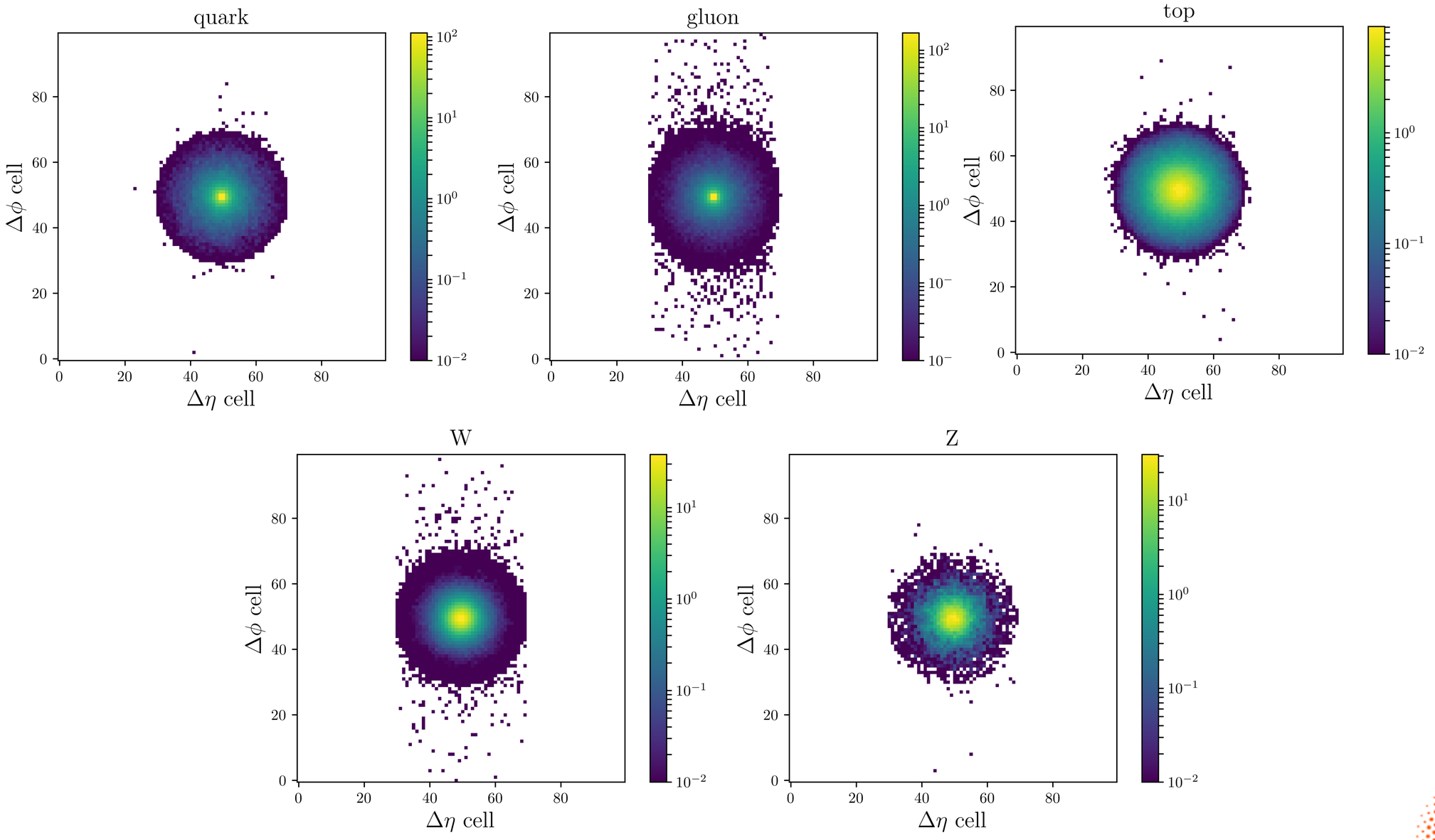




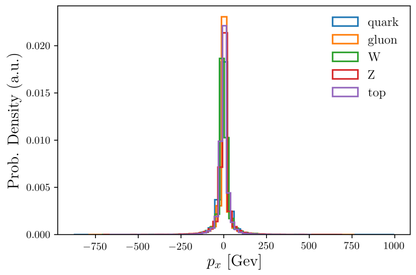


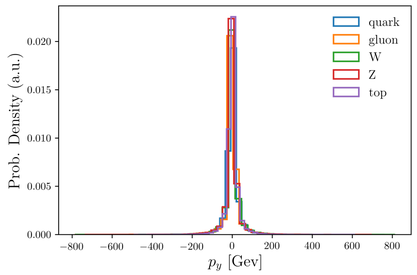


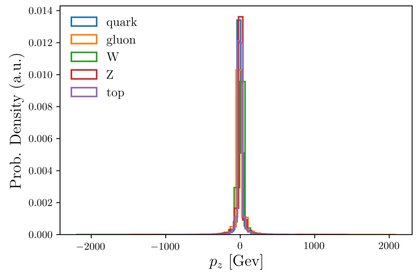
Jets as images

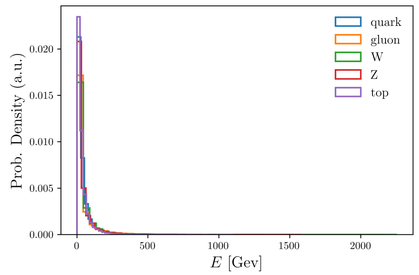


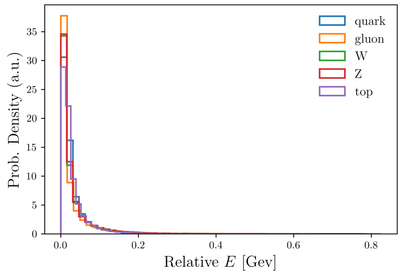
Jets as sequences

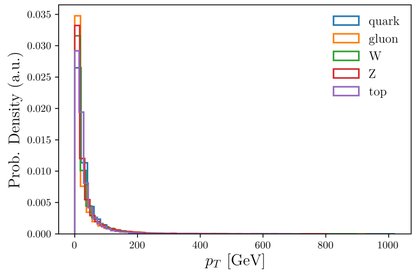


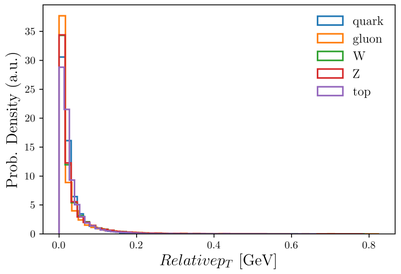


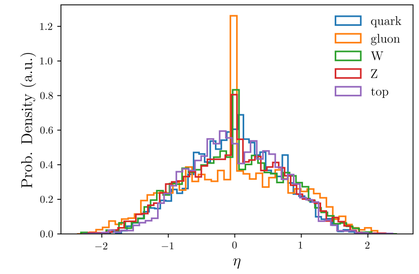


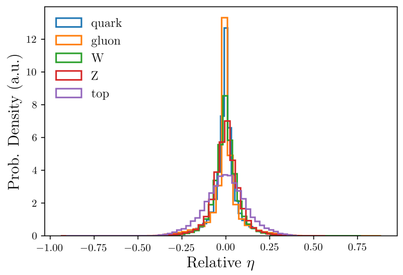


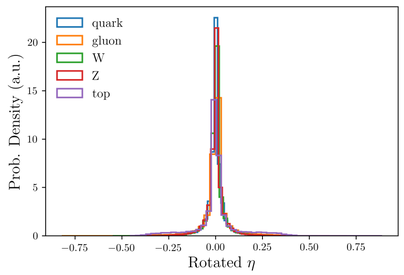


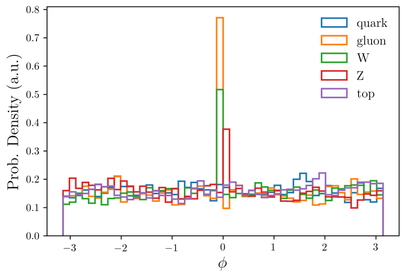


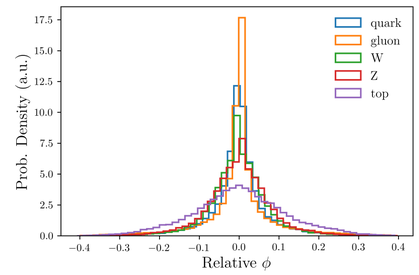


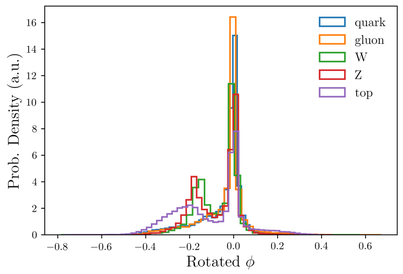


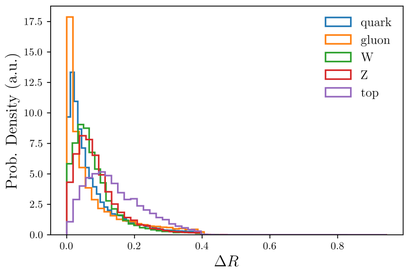


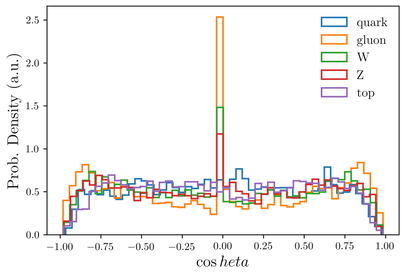


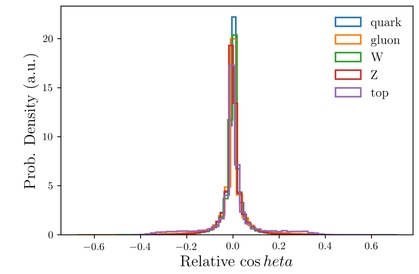




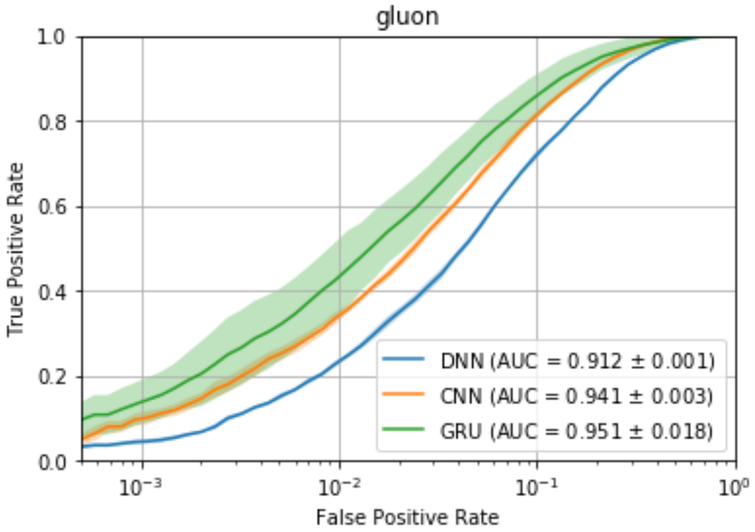


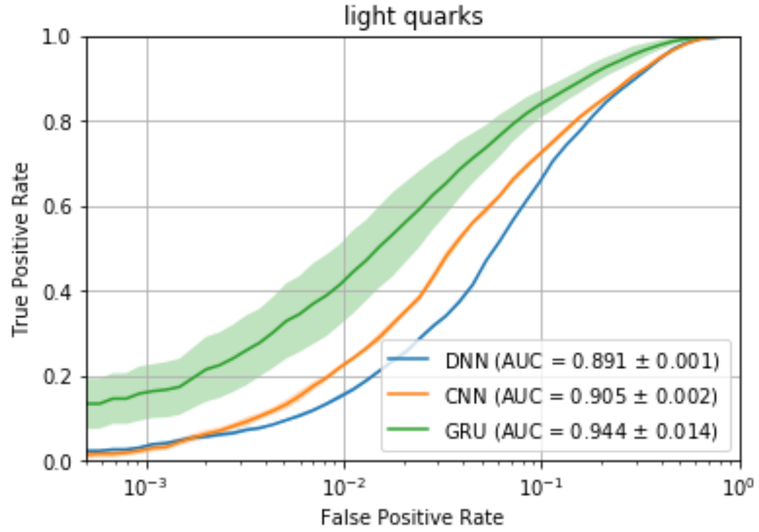


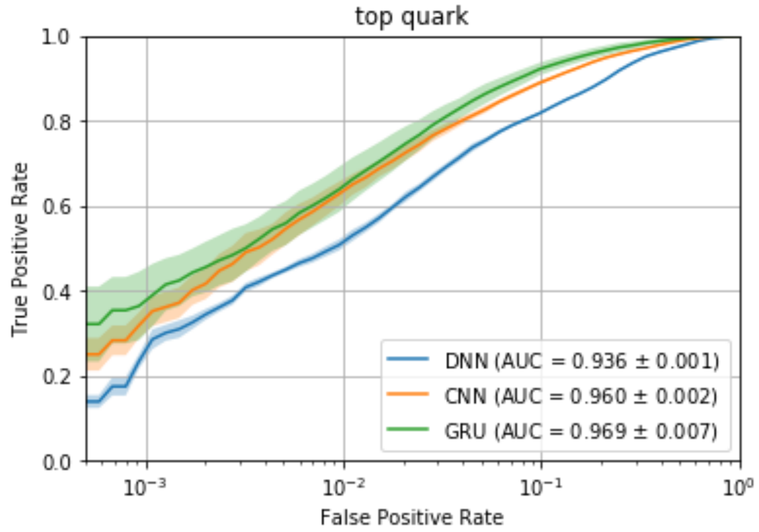


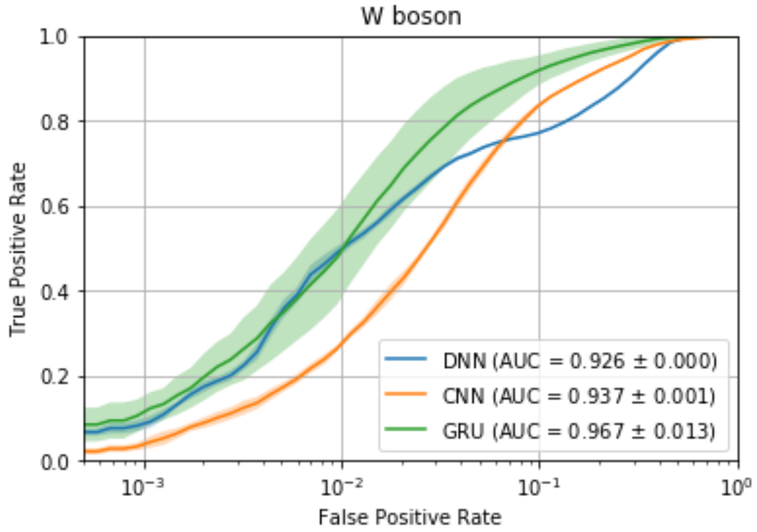


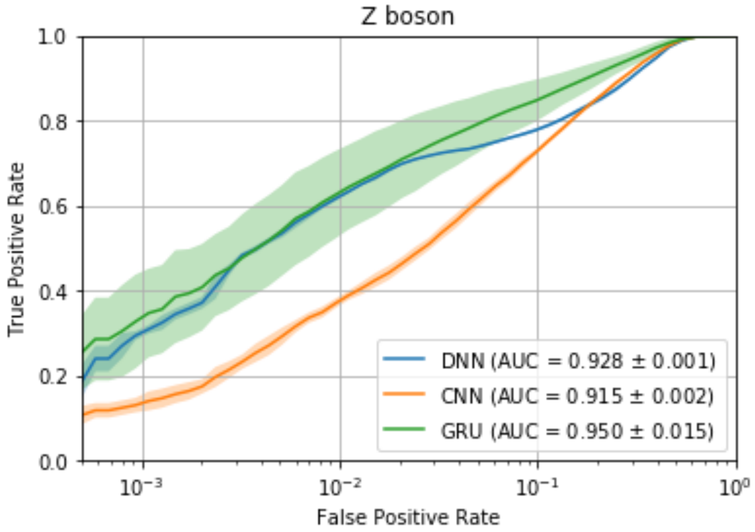
Comparison


gluon
True Positive Rate
False Positive Rate
DNN (AUC = 0.912 ± 0.001)
CNN (AUC = 0.941 ± 0.003)
GRU (AUC = 0.951 ± 0.018)


light quarks
True Positive Rate
False Positive Rate
DNN (AUC = 0.891 ± 0.001)
CNN (AUC = 0.905 ± 0.002)
GRU (AUC = 0.944 ± 0.014)


top quark
True Positive Rate
False Positive Rate
DNN (AUC = 0.936 ± 0.001)
CNN (AUC = 0.960 ± 0.002)
GRU (AUC = 0.969 ± 0.007)


W boson
True Positive Rate
False Positive Rate
DNN (AUC = 0.926 ± 0.000)
CNN (AUC = 0.937 ± 0.001)
GRU (AUC = 0.967 ± 0.013)


Z boson
True Positive Rate
False Positive Rate
DNN (AUC = 0.928 ± 0.001)
CNN (AUC = 0.915 ± 0.002)
GRU (AUC = 0.950 ± 0.015)

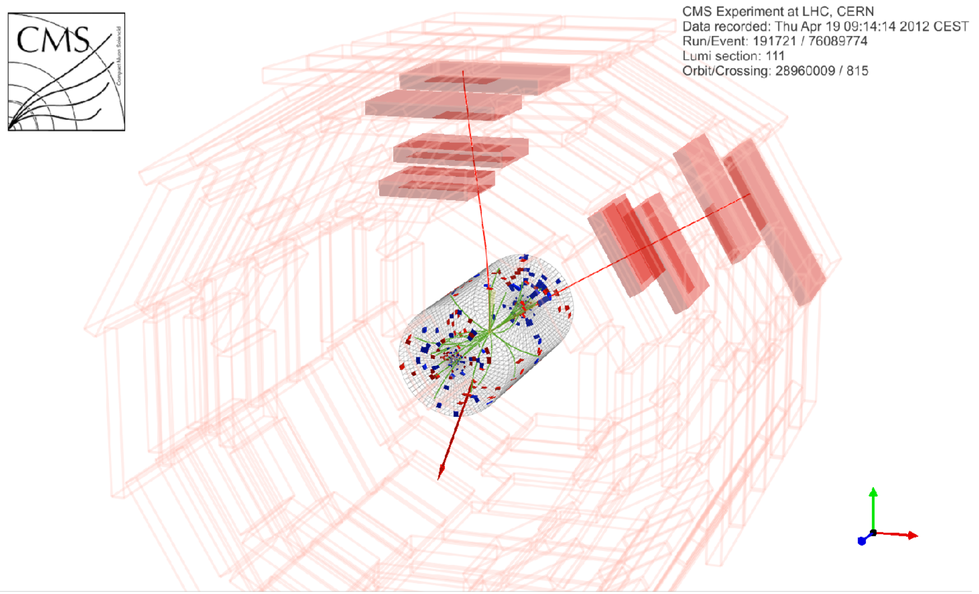
A few remarks

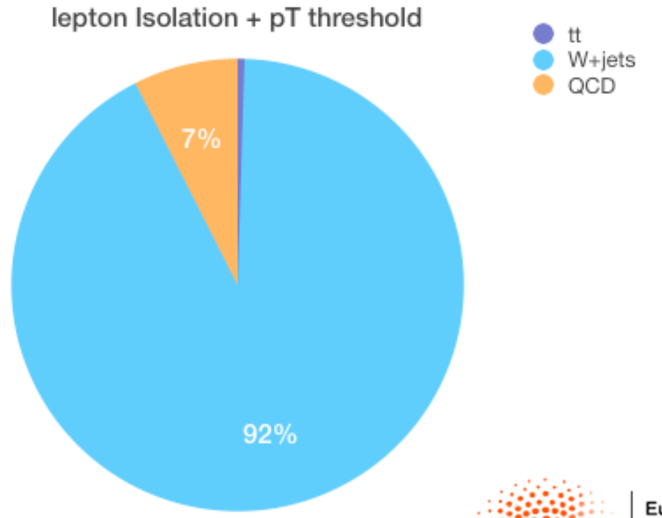
- *RNNs are friendly architectures if your reconstruction delivers particles*
- *CNNs come with a lot of issues*
  - *detector is often an irregular array of sensors*
  - *one could "fake" a regular array, at the cost of loosing in angular resolution*
  - *Also, sometimes this is not really possible (e.g., track angular resolution is pT dependent)*
- *GRU, instead, allows to abstract from geometry and sometimes gives better performances (win-win solution)*

A Topology Classifier

A typical example: leptonic triggers

- *at the LHC, producing an isolated electron or muon is very rare. Typical smoking gun that something interesting happened (Z,W,top,H production)-> **TAKE THEM!***
- *Triggers like those are very central to ATLAS/CMS physics*
- ***The sample selected is enriched in interesting events, but still contaminated by non-interesting ones***
- *Can we clean this up w/o biasing the physics? yes, with ML*

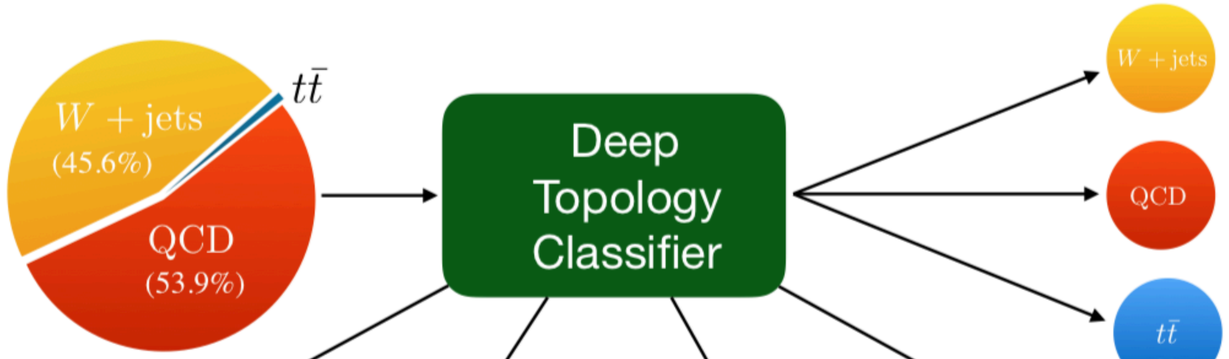




See contribution to NIPS workshop

European Research Council

A Topology Classifier



Deep Topology Classifier

Calo Image Classifier

Raw images of the calorimetry hits fed to a convolutional NN.

Particle Sequence Classifier

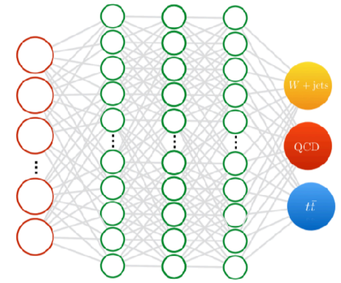
A sequence of particles taken as input to a recurrent NN.

Abstract Image Classifier

Based on an abstract representation of the reconstructed particles as an image to feed to a convolutional NN.

High-level Feature Classifier

Use high-level features as inputs to a fully connected NN.



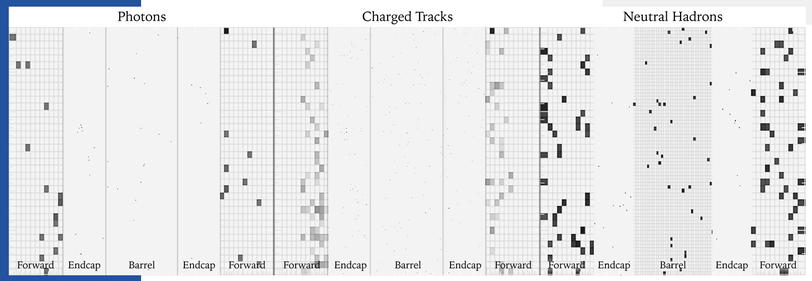
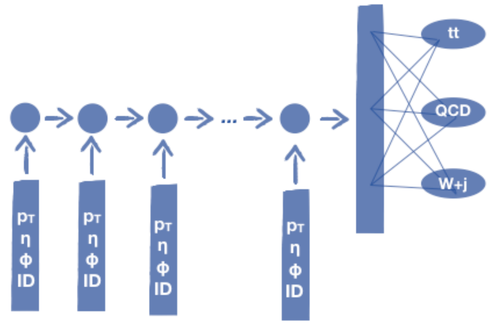
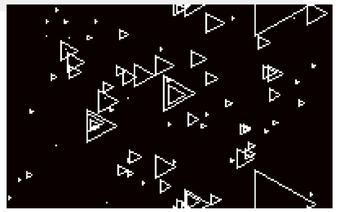
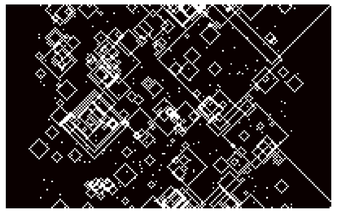


Figure 2: An example of a $t\bar{t}$ event as the input of the raw-image classifier.

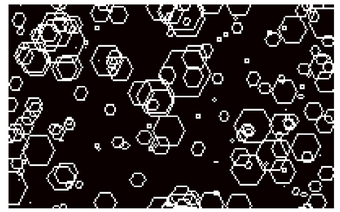




(a) Photons

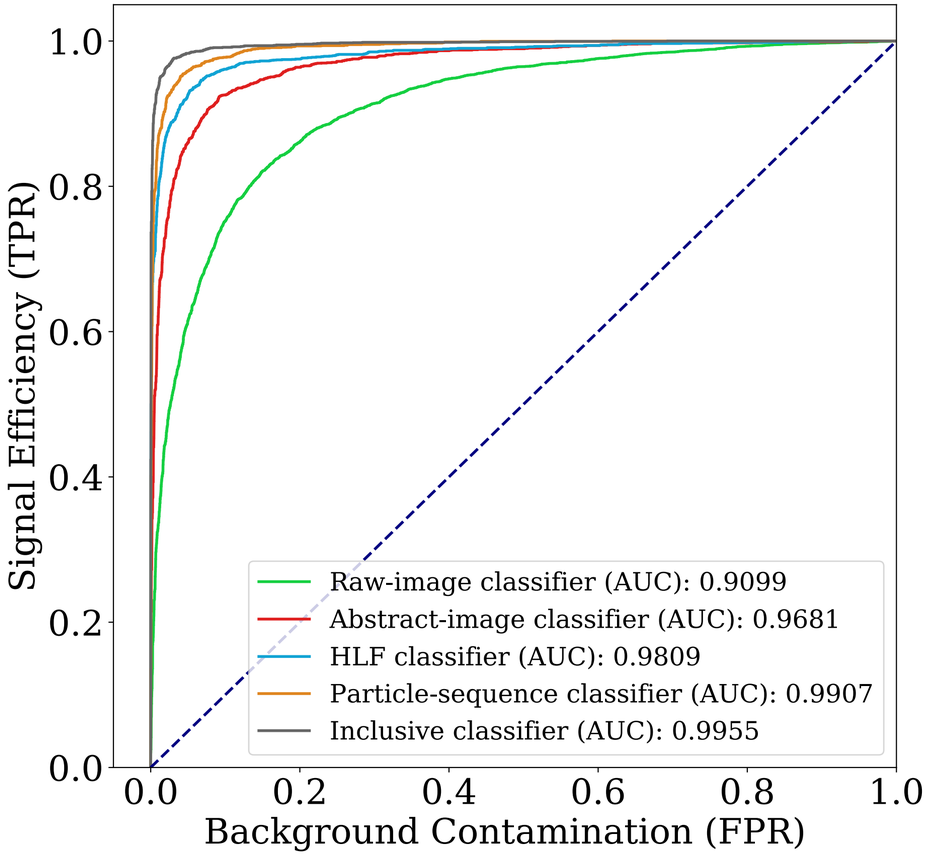


(b) Charged Particles

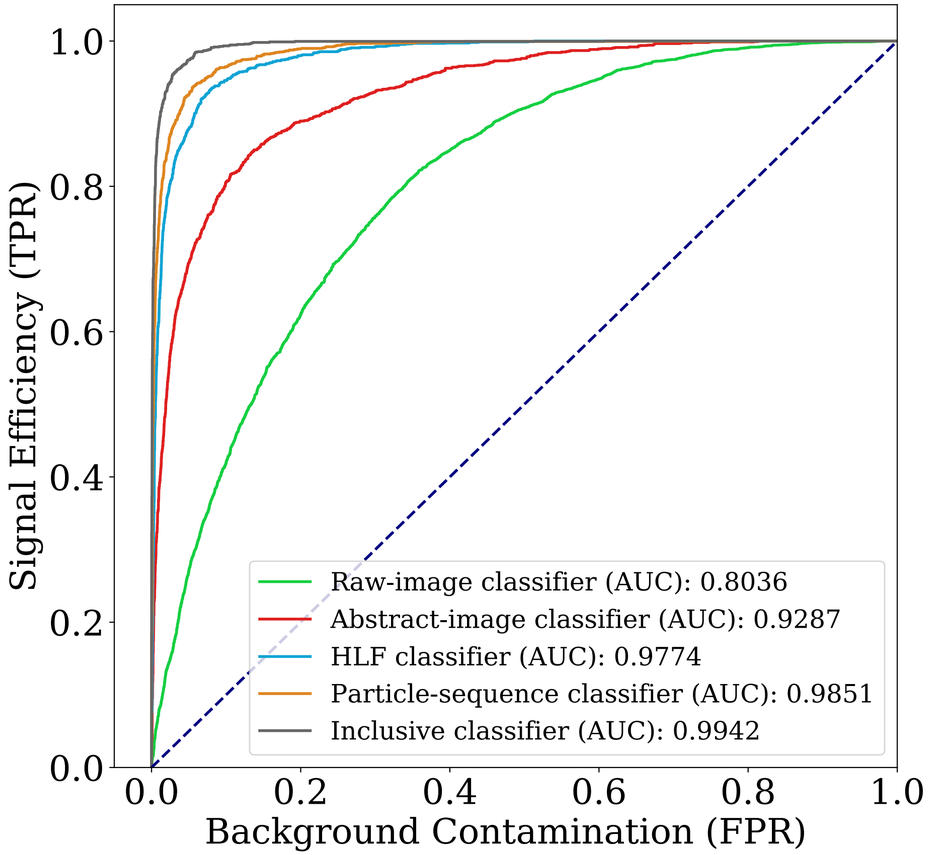


(c) Neutral Hadrons

Selection performances



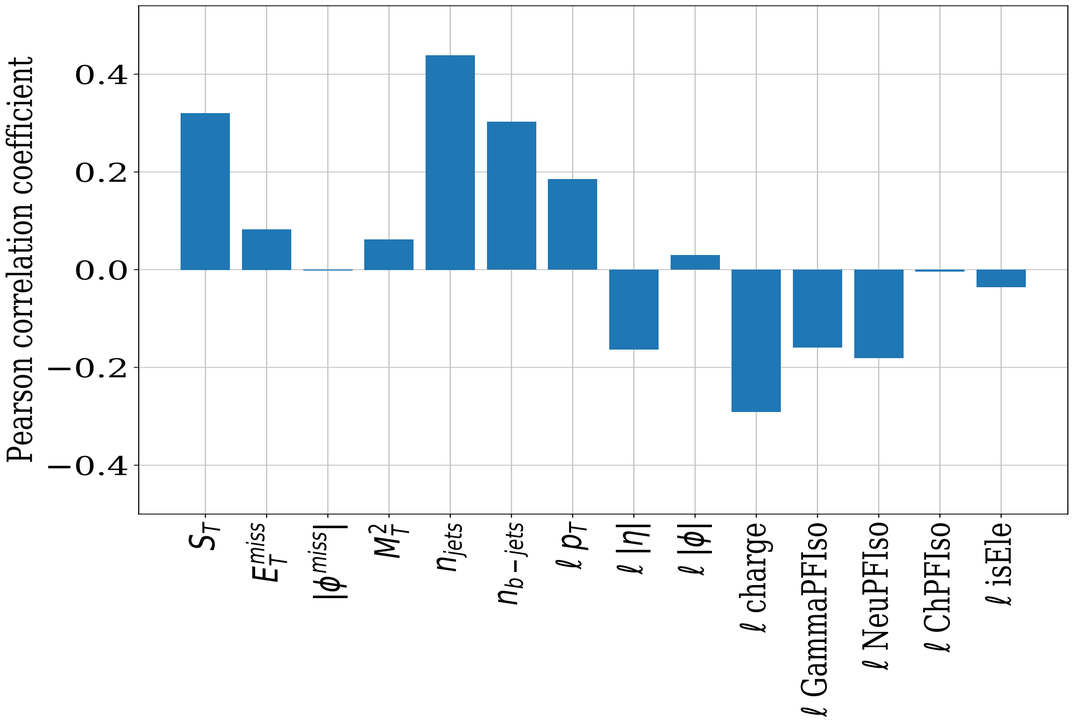
(a) $t\bar{t}$ selector

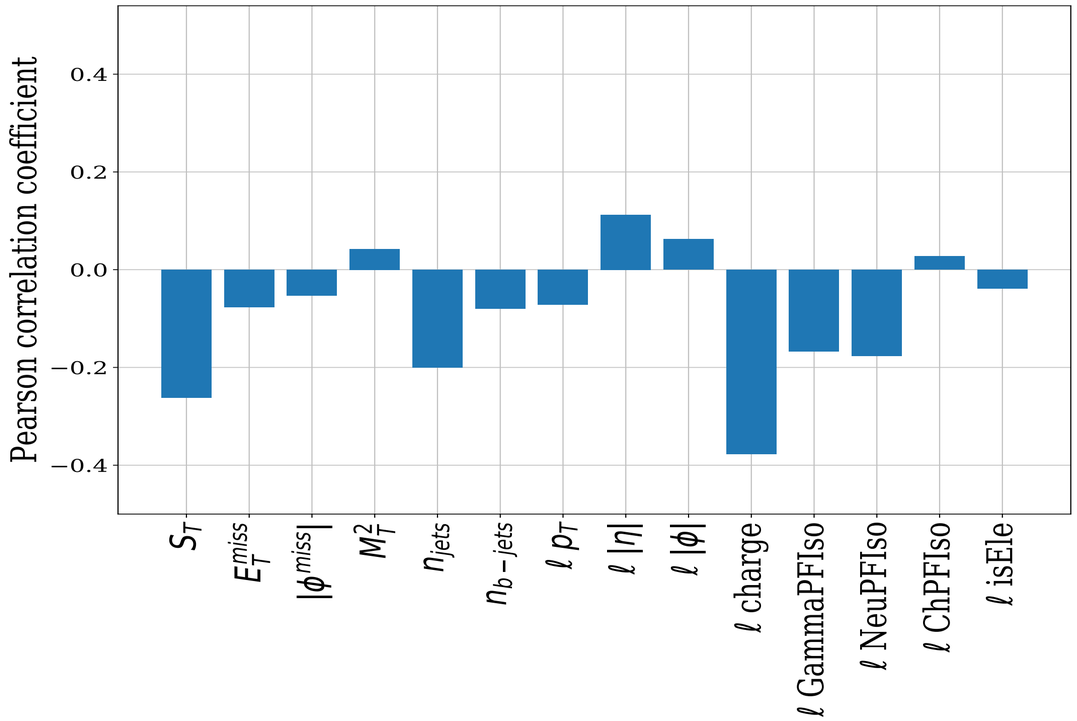


(b) W selector

Can select 99% of the top events and reduce the fraction of written events by a factor ~ 7

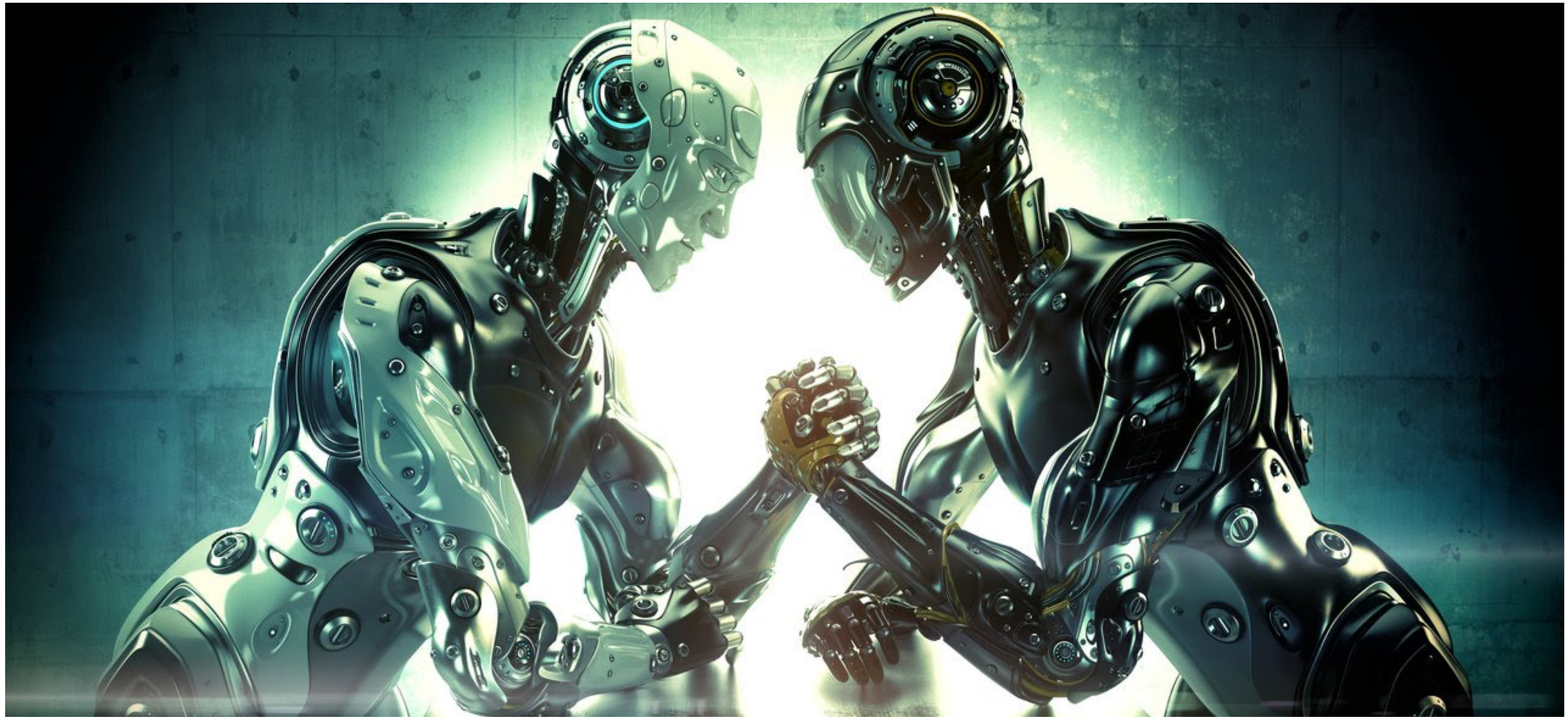
Selection performances





What is the network learning?

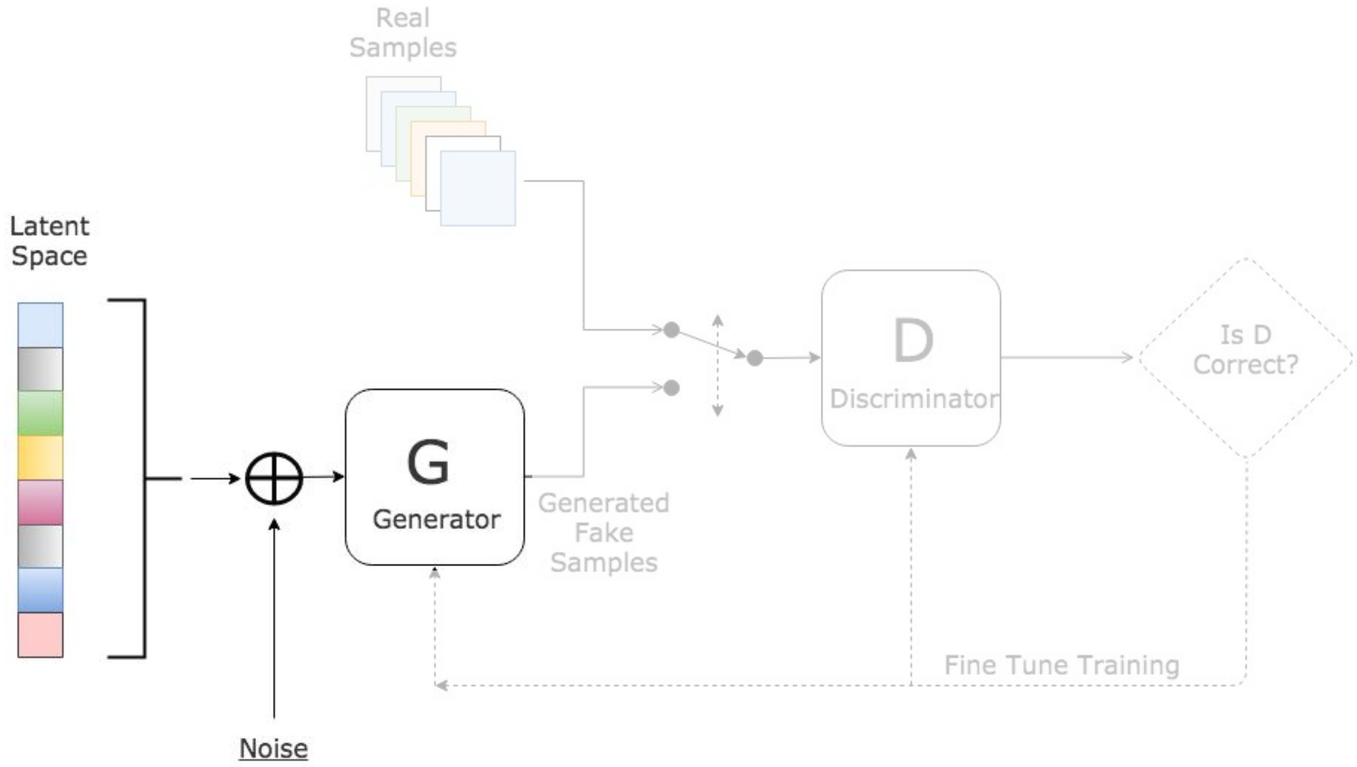
- tt events are more crowded that W events
- leptons in W and tt events are isolated from other particles



Generative Adversarial Networks

Generative Adversarial Training

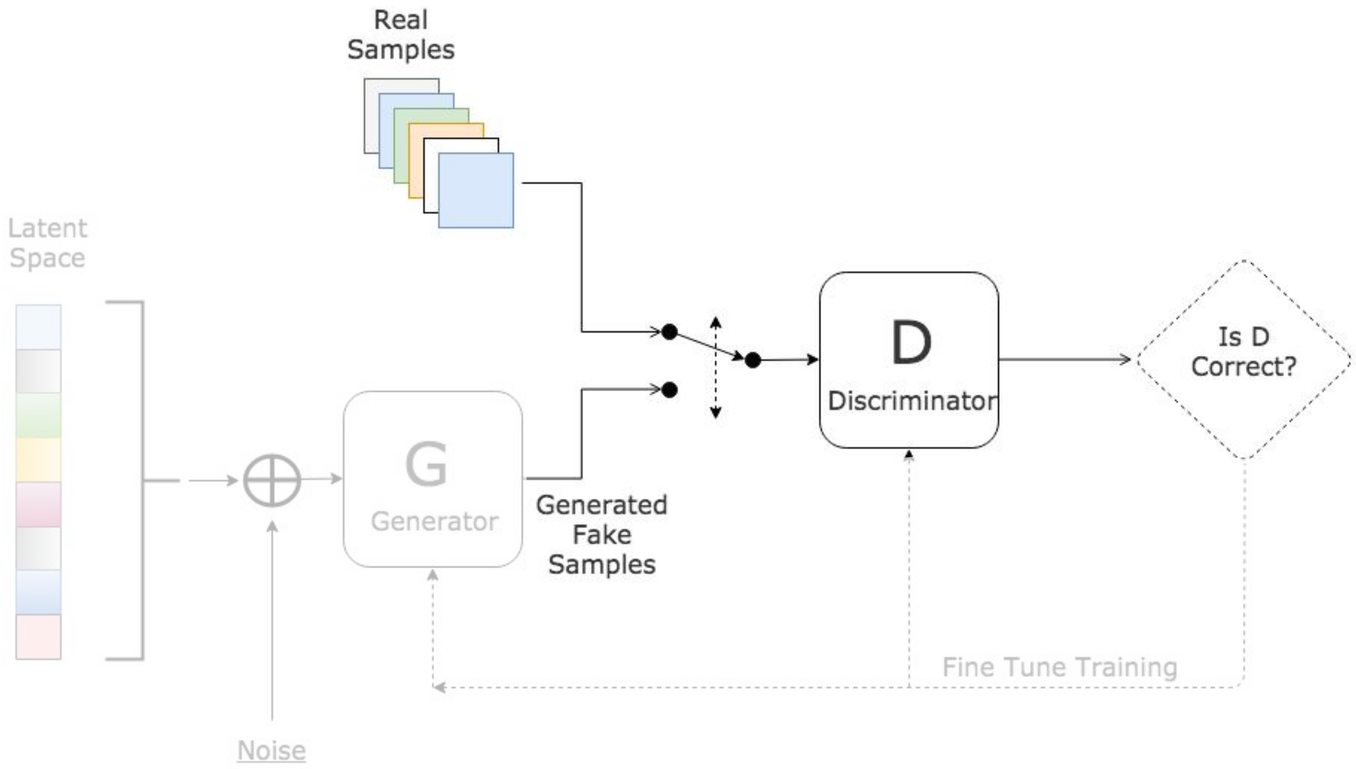
- *Two networks trained in competition*
  - ***Generator: creates images starting from random noise*** *(and optionally some other information to transform)*
  - *Discriminator: tries to distinguish true from generator-created images*



- *The loss function to minimise is written as Loss(Gen)-Loss(Disc)*
  - *Goes up is discriminator improves*
  - *Goes down if the generator improves*
- *The generator learns to make images like a given set it never sees, simply training itself to fool the generator*

Generative Adversarial Training

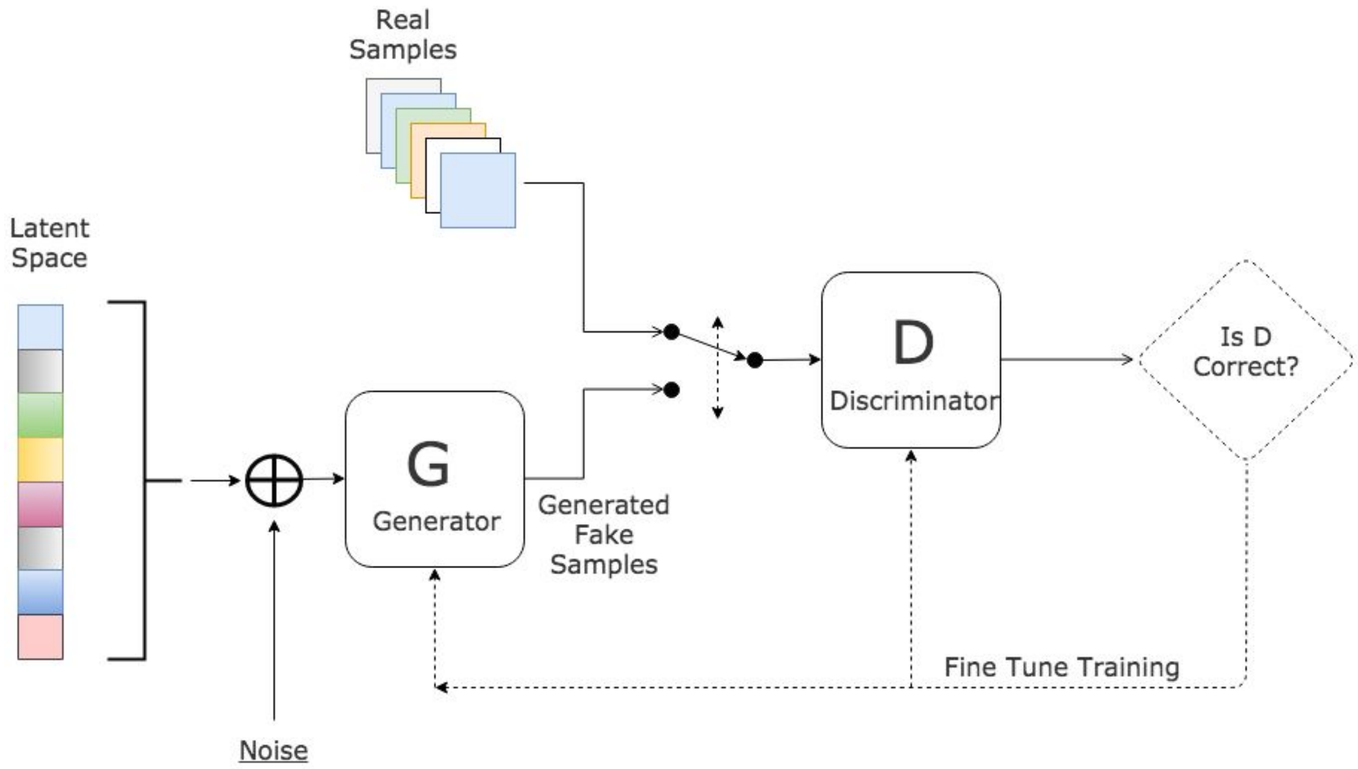
- *Two networks trained in competition*
  - *Generator: creates images starting from random noise (and optionally some other information to transform)*
  - ***Discriminator: tries to distinguish true from generator-created images***



- *The loss function to minimise is written as Loss(Gen)-Loss(Disc)*
  - *Goes up is discriminator improves*
  - *Goes down if the generator improves*
- *The generator learns to make images like a given set it never sees, simply training itself to fool the generator*

Generative Adversarial Training

- *Two networks trained in competition*
  - *Generator: creates images starting from random noise (and optionally some other information to transform)*
  - *Discriminator: tries to distinguish true from generator-created images*



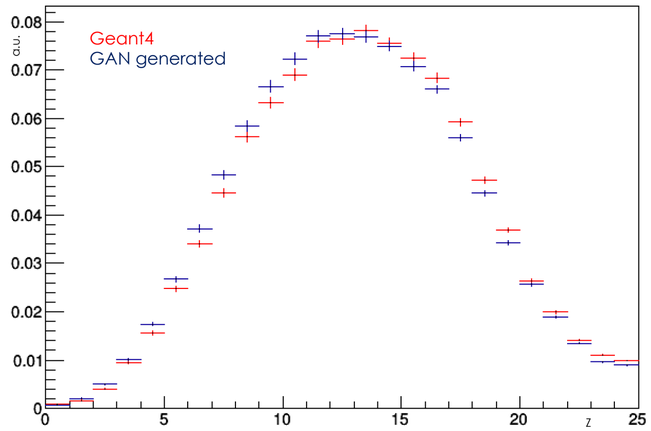
- ***The loss function to minimise is written as Loss(Gen)-Loss(Disc)***
  - ***Goes up is discriminator improves***
  - ***Goes down if the generator improves***
- ***The generator learns to make images like a given set it never sees, simply training itself to fool the generator***

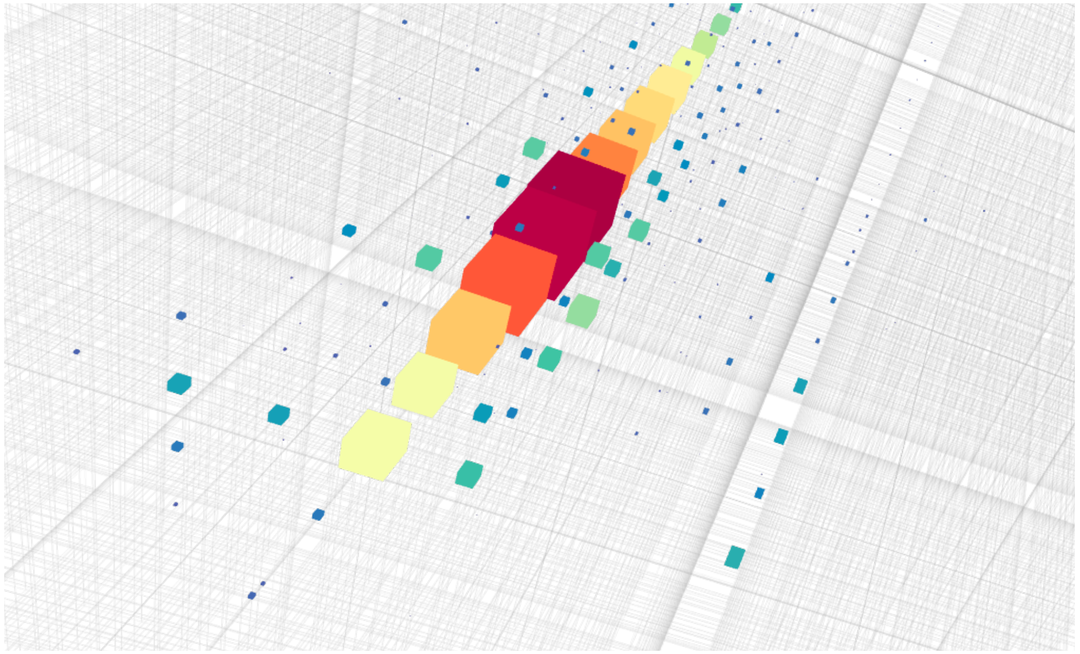
Adversarial training in azione

Image generation

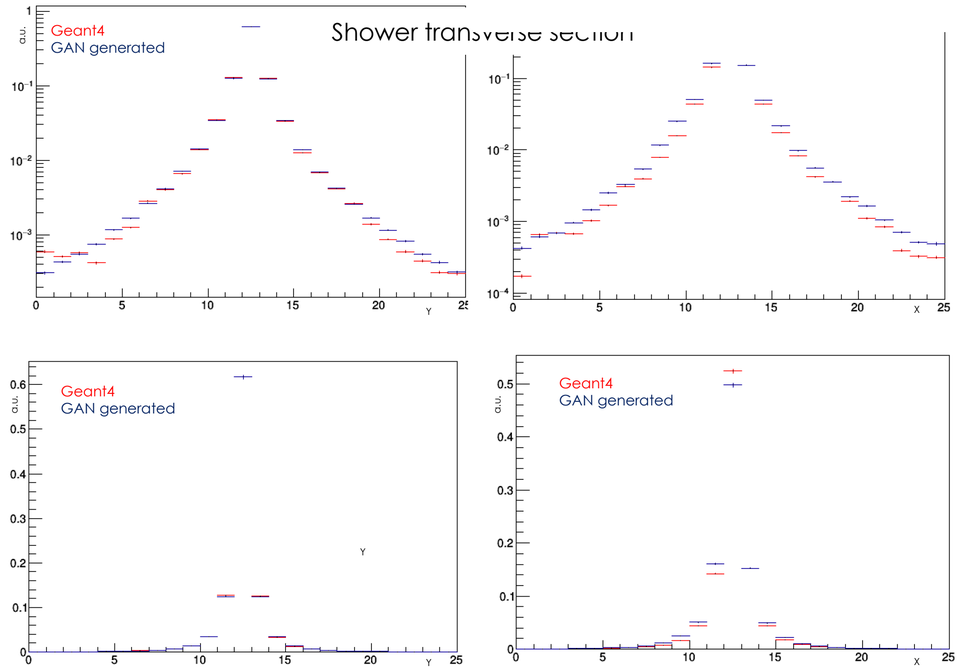
- Start from random noise
- Works very well with images
  - Applied to electron showers in digital calorimeters as a replacement of GEANT

Shower longitudinal section





Shower transverse section

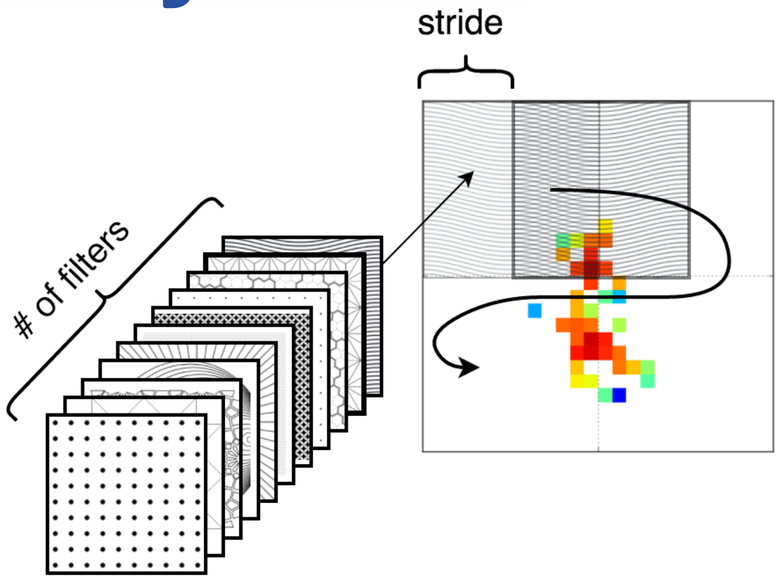


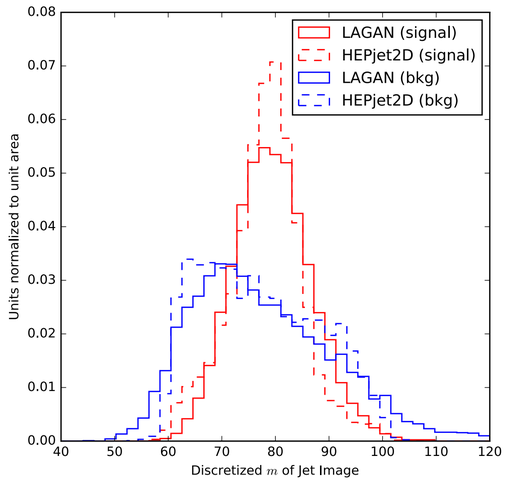
See contribution to NIPS workshop

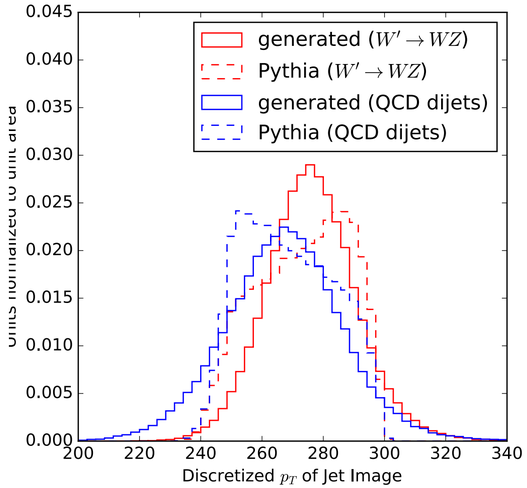
European Research Council

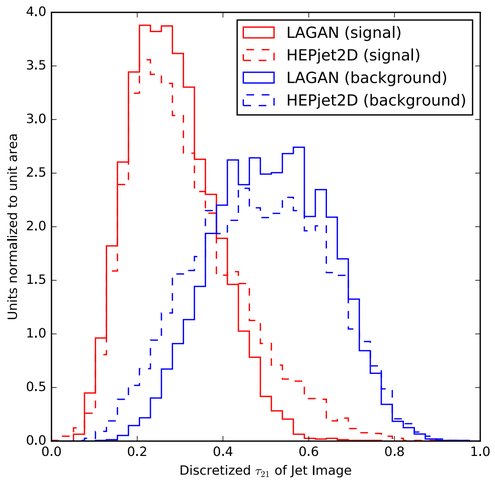
Generating full jets

- Start from random noise
- Works very well with images
  - Applied to electron showers in digital calorimeters as a replacement of GEANT







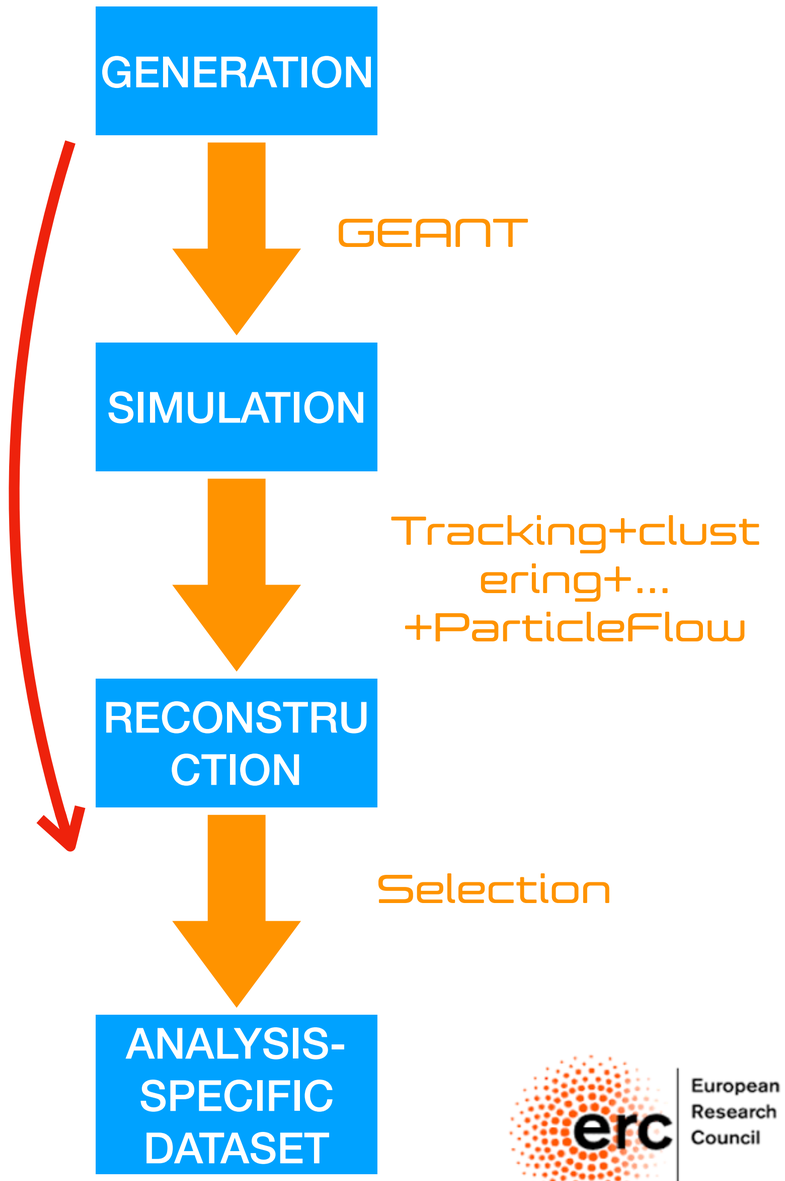


de Olivera, Paganini, and Nachman
https://arxiv.org/pdf/1701.05927.pdf

Figure 6: The distributions of image mass $m(I)$, transverse momentum $p_{\mathrm{T}}(I)$, and n-subjettiness $\tau_{21}(I)$. See the text for definitions.

Simulation is half of the problem

- *Reconstruction involves more than one detector (e.g., tracker + calorimeter) and produces (at least in CMS) a list of particles*
- *GANs were proved to be useful to emulate the SIM+RECO step in one goal*
  - *jets out of full particle reconstruction emulated in a GAN*
  - *trained on actual SIM+RECO synthetic data by CMS*



Simulation is half of the problem

- *Reconstruction involves more than one detector (e.g., tracker + calorimeter) and produces (at least in CMS) a list of particles*
- *GANs were proved to be useful to emulate the SIM+RECO step in one goal*
  - *jets out of full particle reconstruction emulated in a GAN*
  - *trained on actual SIM+RECO synthetic data by CMS*

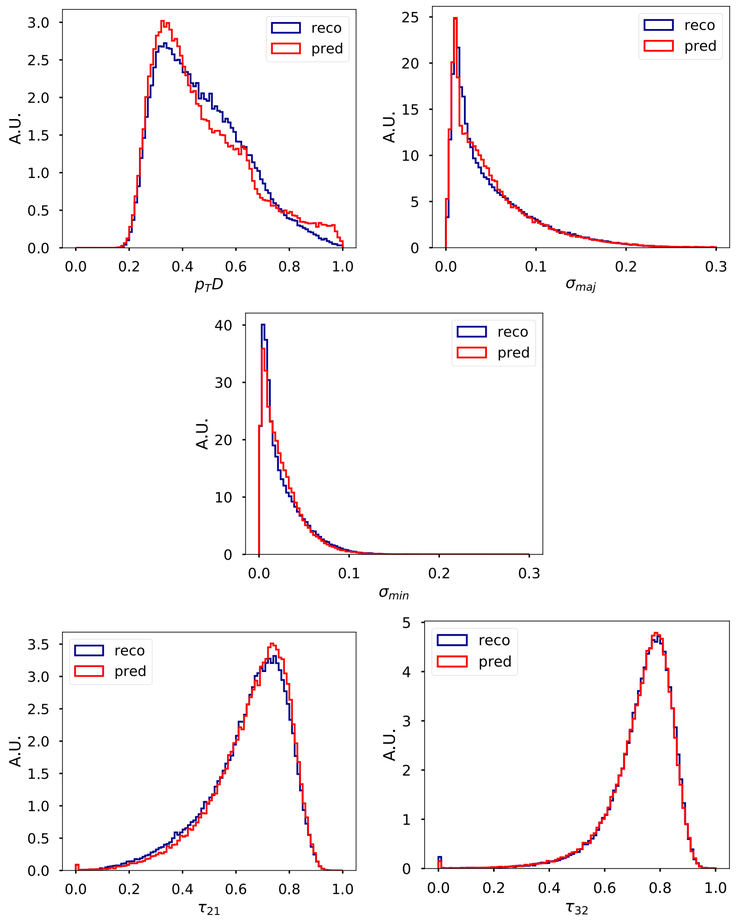


Fig. 6 Distribution of high level variables used for quark/gluon discrimination (first two rows) and merged jets tagging (last row). Blue histograms are obtained from the input data, while red ones are obtained using the generative model.

END OF EPISODE I

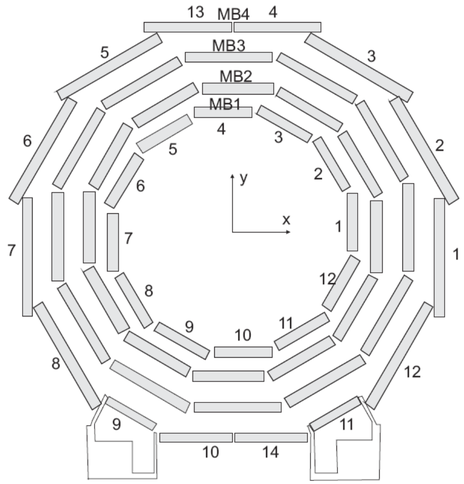
Backup

Detector Monitoring

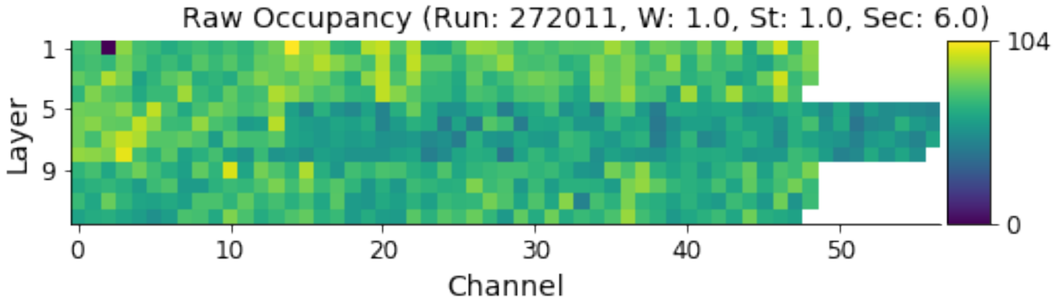
European Research Council

Data Quality Monitoring

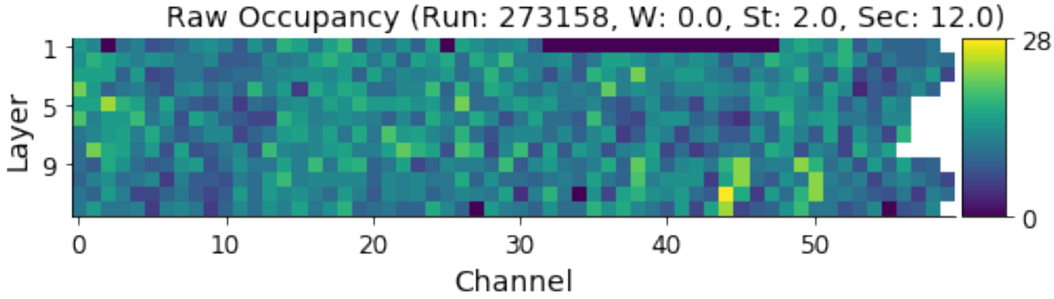
- *When taking data, >1 person watches for anomalies in the detector 24/7*
- *At this stage no global processing of the event*
- *Instead, local information from detector components available (e.g., detector occupancy in a certain time window)*



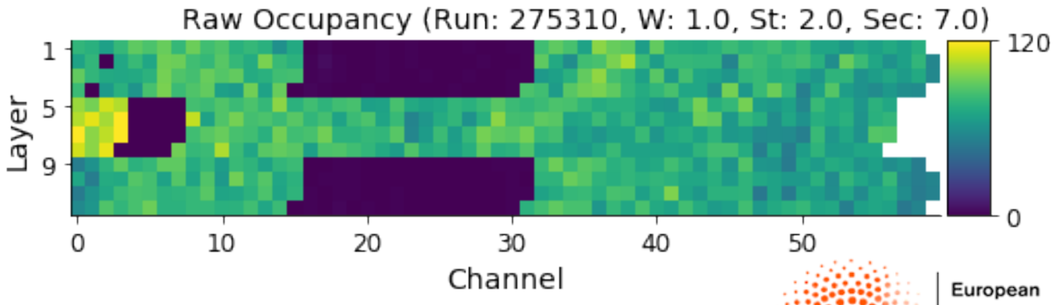
A



B



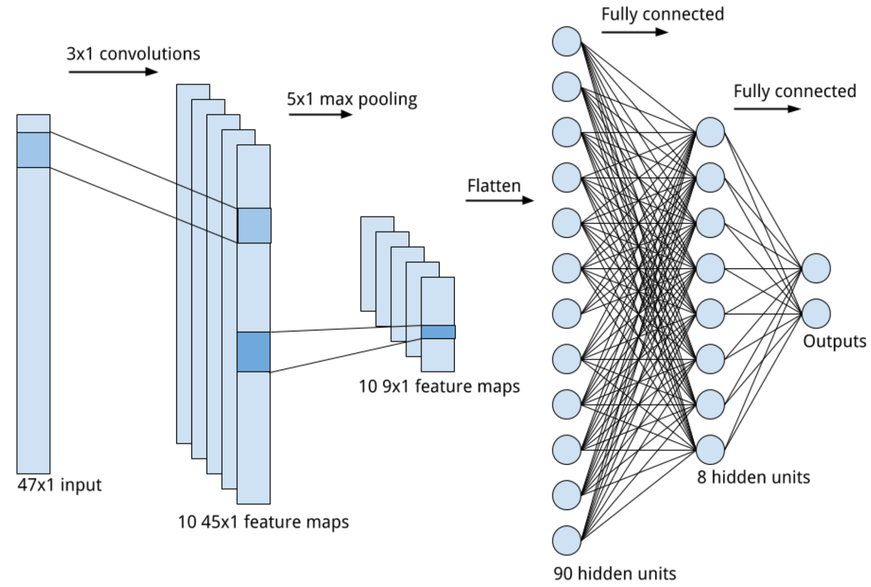
C

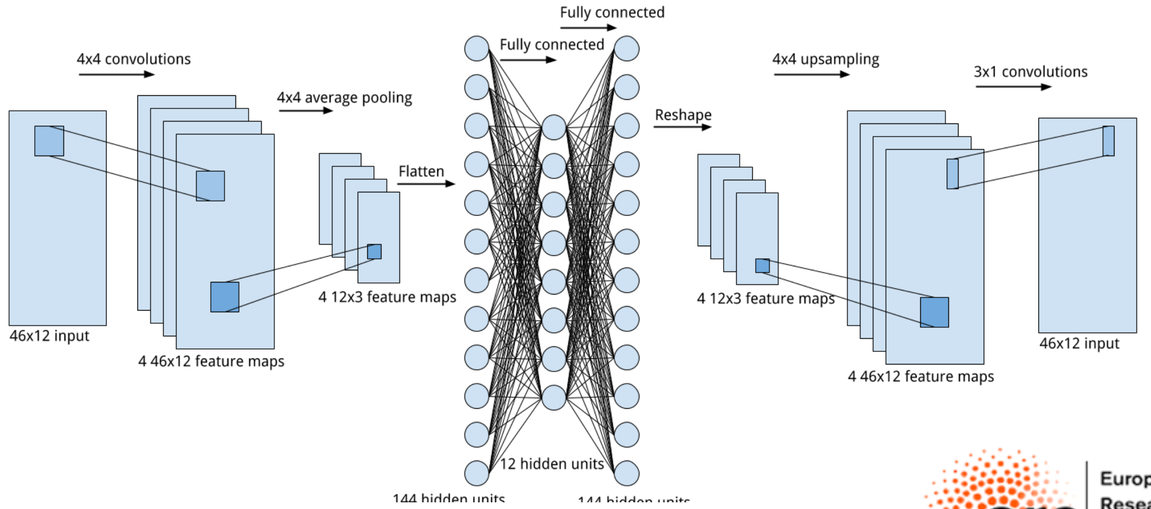


Two approaches

- *Given the nature of these data, ConvNN are a natural analysis tool. Two approaches pursued*
  - *Classify good vs bad data. Works if failure mode is known*
  - *Use autoencoders to assess data "typicality". Generalises to unknown failure modes*

A. Pol et al., to appear soon

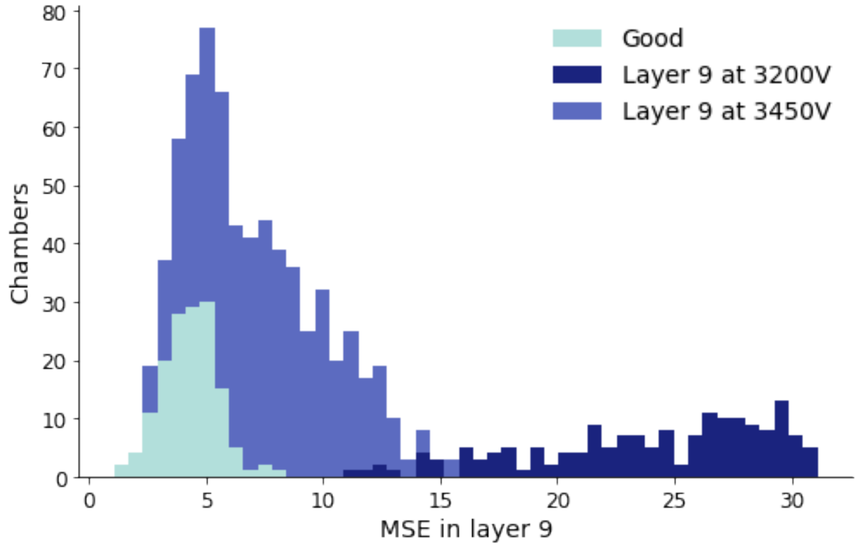


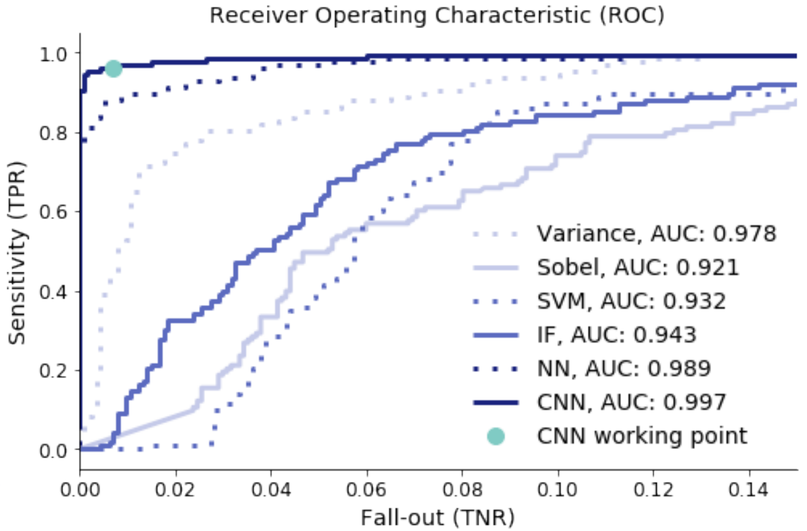


Two approaches

- *Given the nature of these data, ConvNN are a natural analysis tool. Two approaches pursued*
  - *Classify good vs bad data. Works if failure mode is known*
  - *Use autoencoders to assess data "typicality". Generalises to unknown failure modes*

A. Pol et al., to appear soon

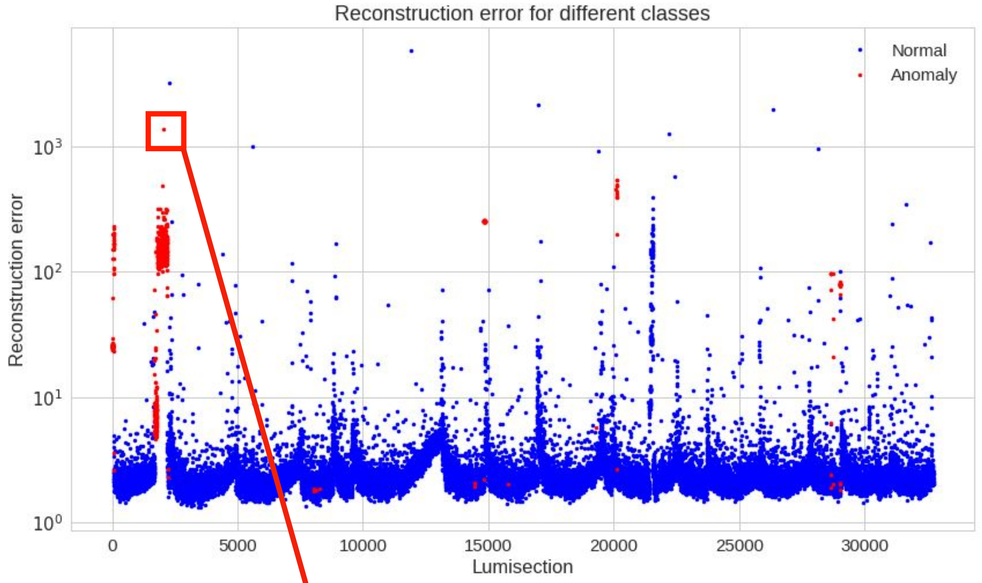


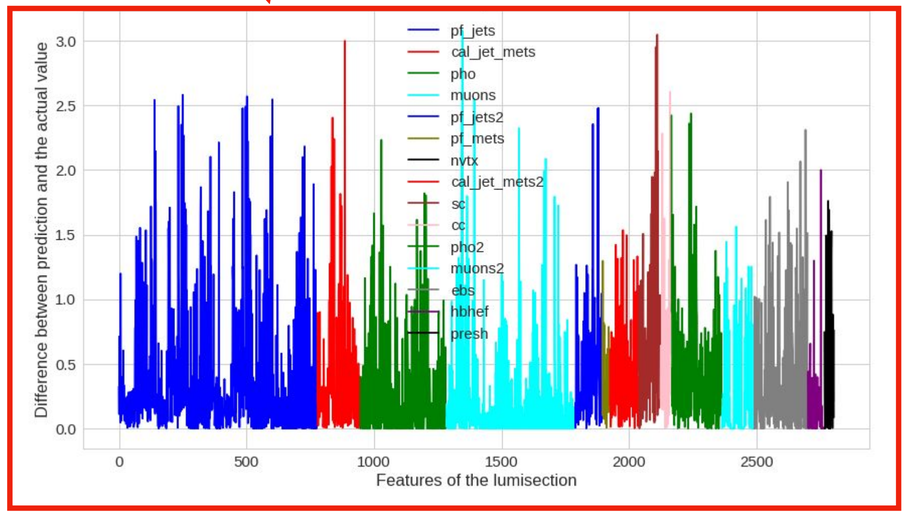


European Research Council

Data Quality Certification

- *Autoencoder-based 1-class approach generalises to later stages of quality assessment*
  - *after reconstruction of the events, event reconstruction allows a global assessment (w.g., looking at electrons, muons, etc rather than hits in the detector)*
  - *A global autoencoder can spot all these features*
  - *Monitoring individual contributions to loss function (e.g., MSE) one can track the problem back to a specific physics object/detector component*





F. Široký et al., to appear sooner or later

HL4ML: FPGA details

Xilinx Vivado 2017.2

Results are slightly different in other versions of Vivado

e.g. 2016.4 optimization is less performant for Xilinx ultrascale FPGAs

Clock frequency: 200 MHz

Latency results can vary (~10%) with different clock choices

FPGA: Xilinx Kintex Ultrascale (XCKU115-FLVB2104)

Results are slightly different in other FPGAs

e.g. Virtex-7 FPGAs are slightly differently optimized

European Research Council

Why Deep Learning

- *Neural network can model non linear functions*
  - *the more complex is the network, the more functions it can approximate*
- *Neural network are faster to evaluate (inference) than typical reco algorithm.*
  - *This is the speed up we need*
- *Neural Networks (unlike other kind of ML algorithms) are very good with raw (non-preprocessed) data (the recorded hits in the event)*
  - *could use them directly on the detector inputs*

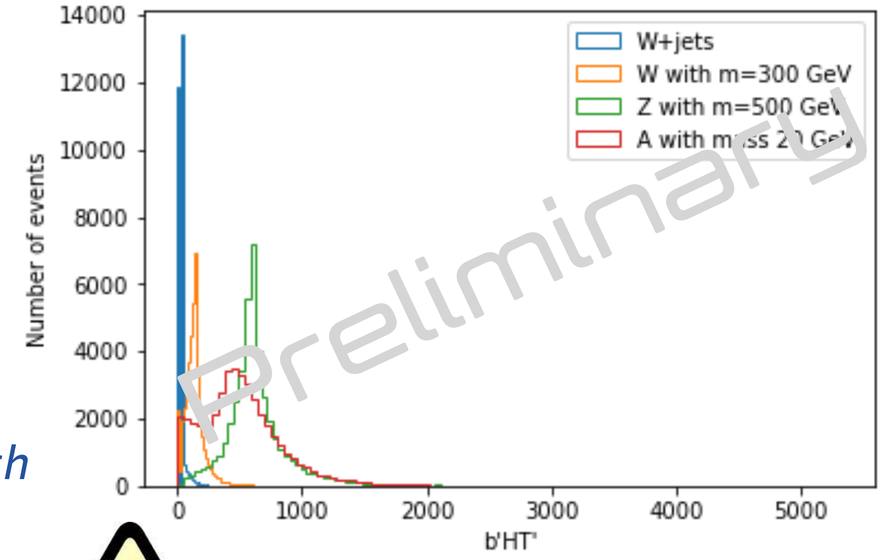
$$(pT, \eta, \phi, E)_{OFFLINE} = f\left((pT, \eta, \phi, E)_{ONLINE}\right)$$

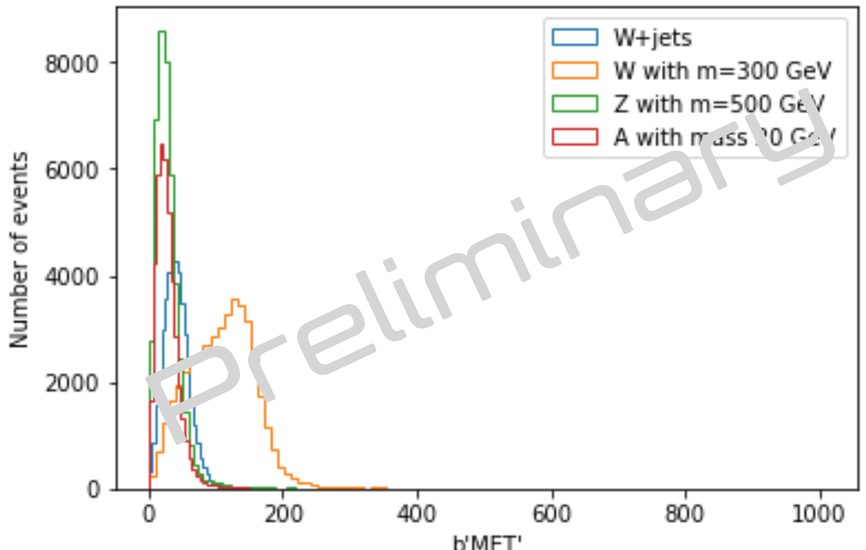
$$(pT, \eta, \phi, E)_{OFFLINE} = g\left(\text{Event hits}\right)$$

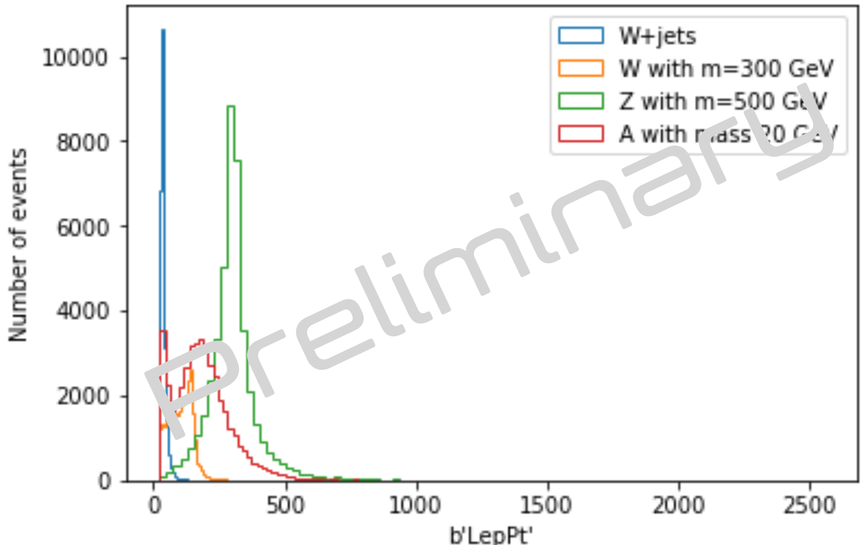
One would have to learn f and g to evaluate them at trigger. Online processing is replaced by offline training

Beyond the toy-model

- *Approach works in principle*
  - *Can identity easily 2 of the 3 models*
  - *With enough statistics, could see the third*
- *Might not work in absolute*
  - *encoder based on physics motivate quantities which are not model-agnostic*
  - *Use deep:learning: train on raw data directly.* ***To be done next***

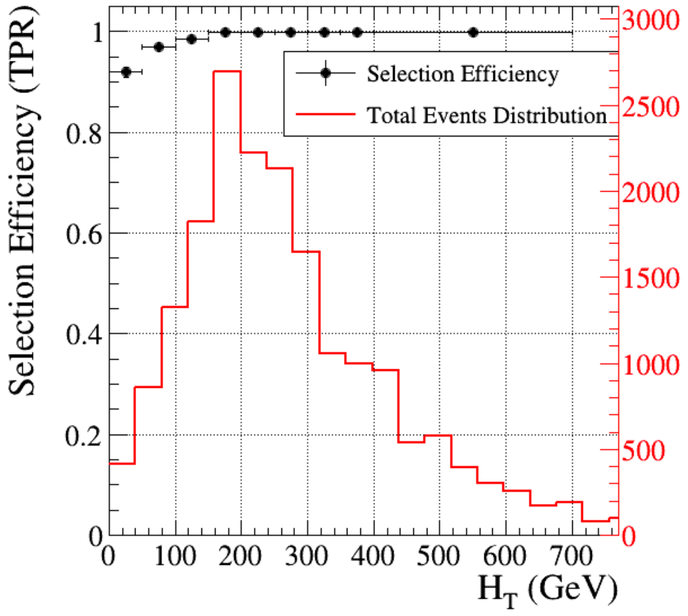


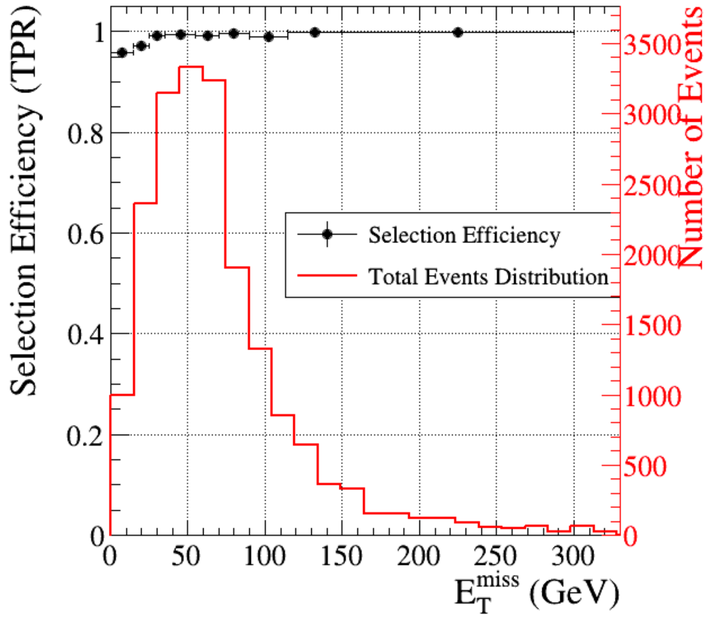


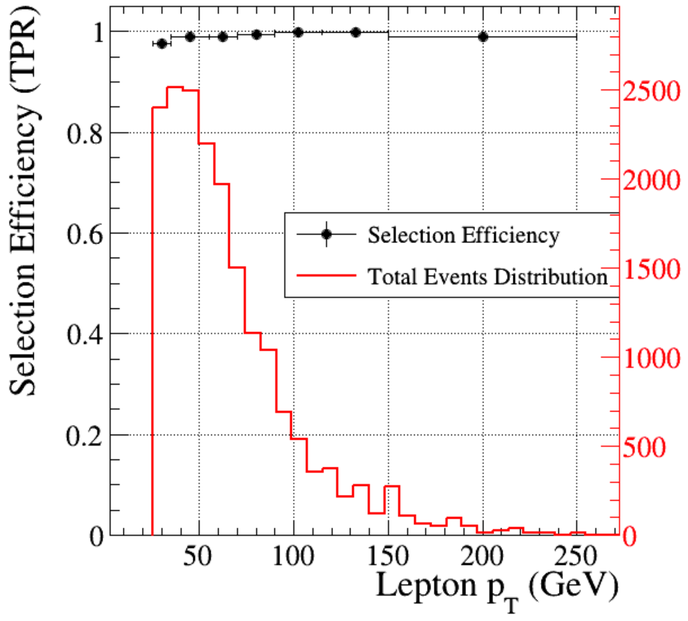


Kinematic Bias?

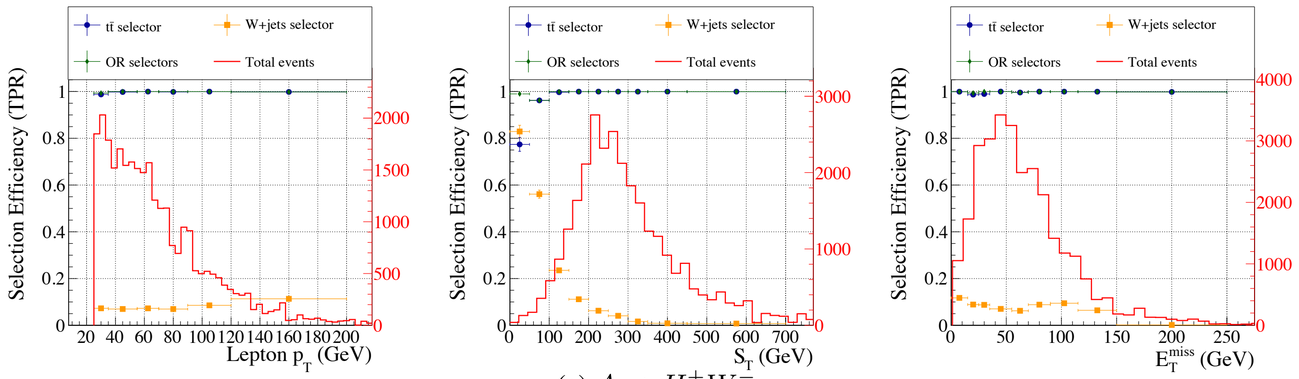
- *With 99% signal efficiency, bias on kinematic variables within the uncertainty of a trigger-efficiency measurement*



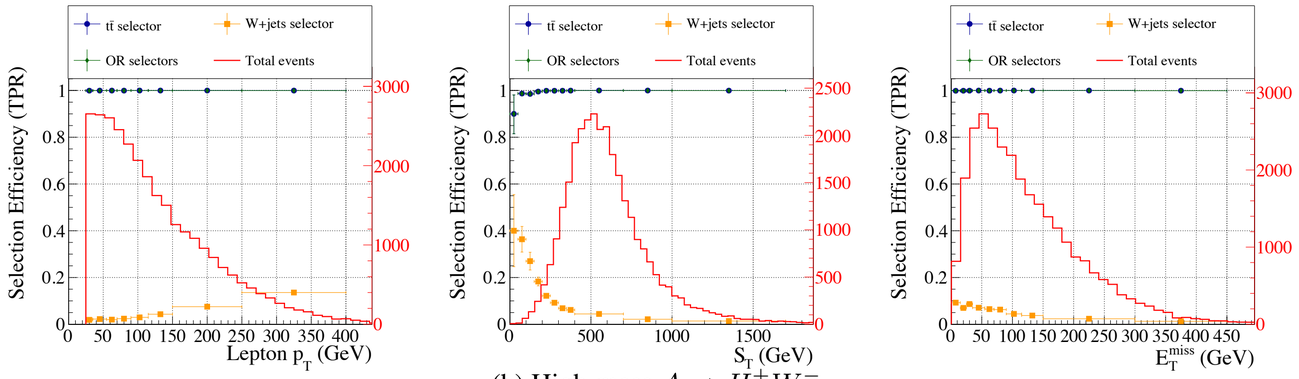




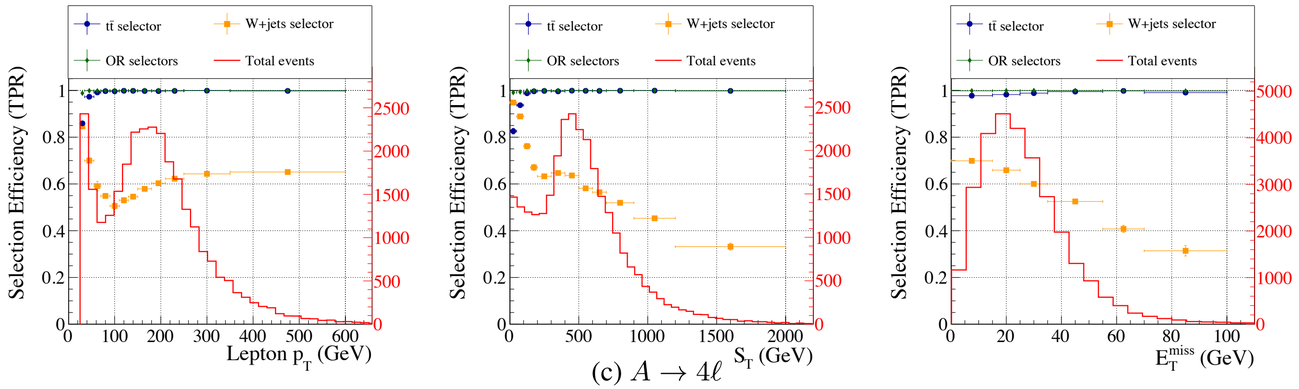
TOPCLASS: do we kill New Physics?



(a) $A \to H^+W^-$



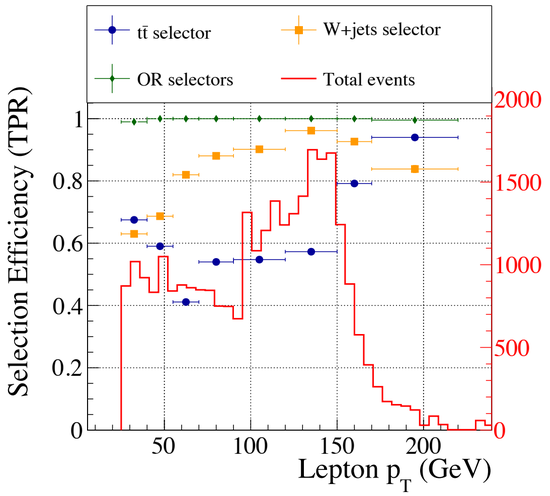
(b) High-mass $A \to H^+W^-$

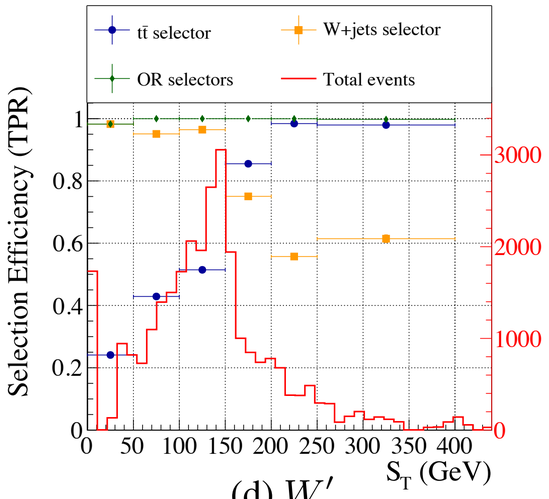


(c) $A \to 4\ell$

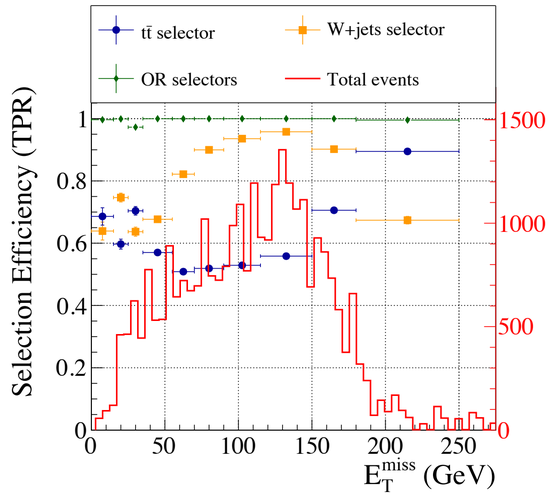
European Research Council

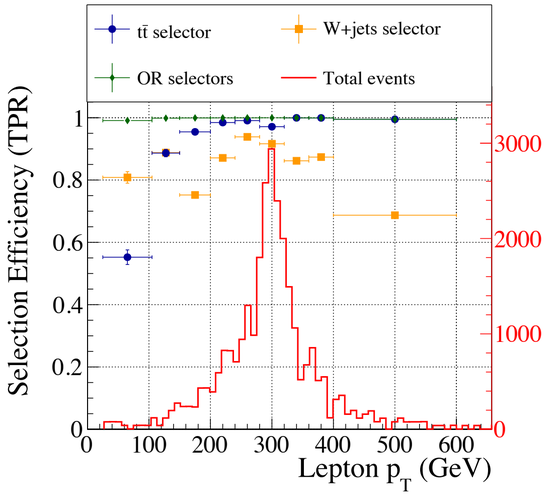
TOPCLASS: do we kill New Physics?

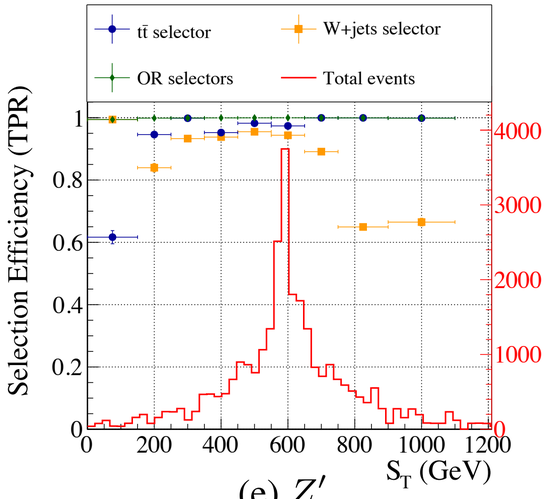




(d) W'







(e) Z'