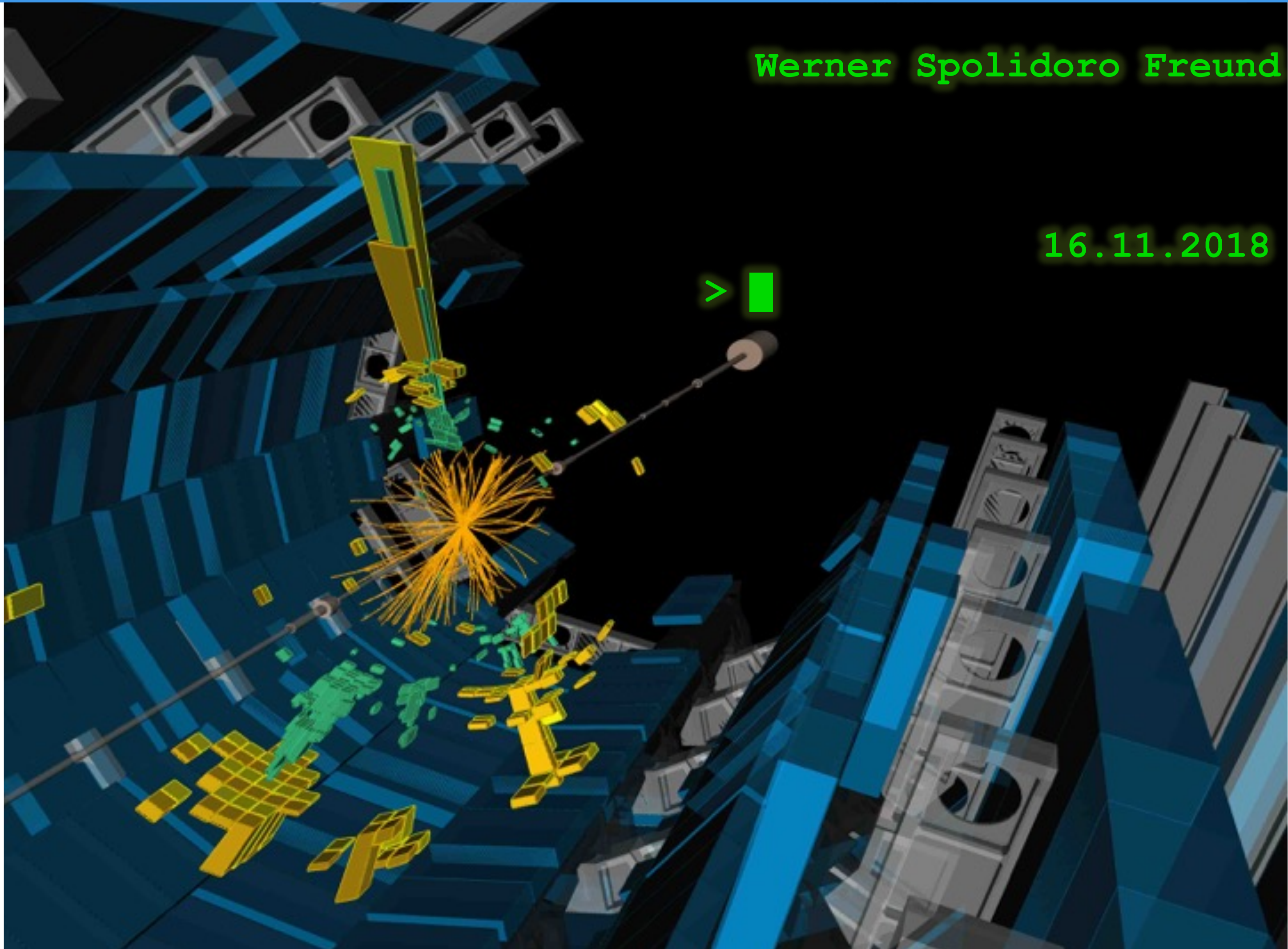
Machine Learning for calorimetric reconstruction in ATLAS

- with additional interesting stuff…



Werner Spolidoro Freund

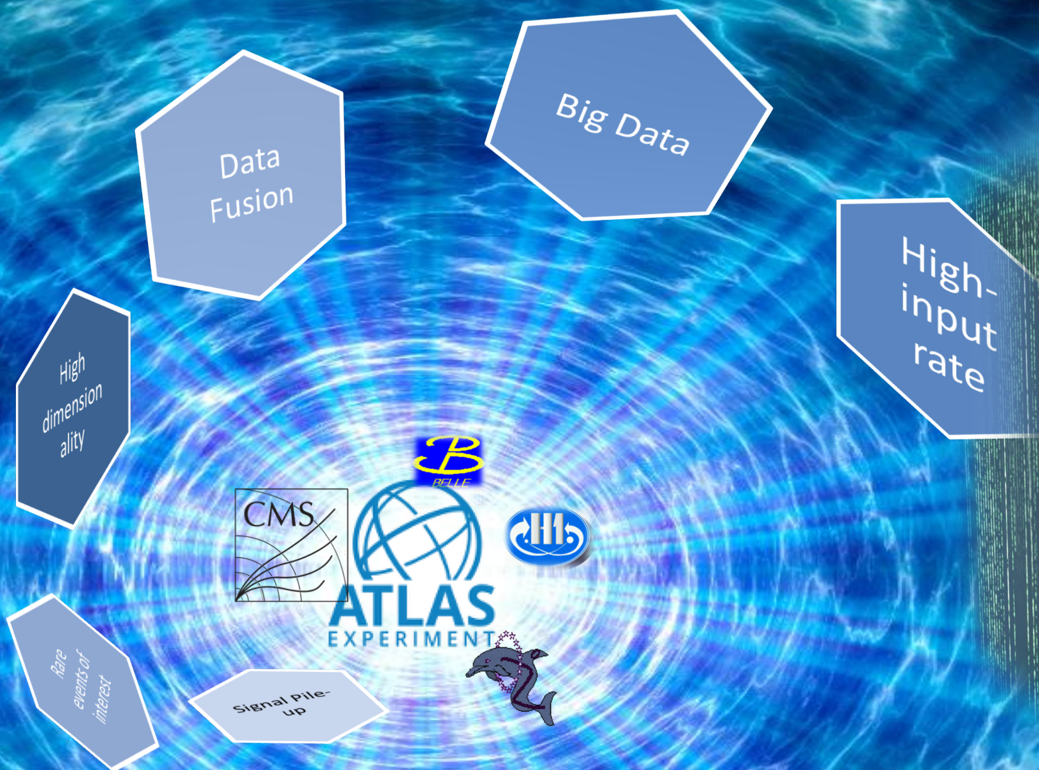
16.11.2018

>

Outline

- Introduction
  - Common challenges;
  - Solutions;
  - Domain Knowledge (ATLAS Calorimeters);
  - Reconstruction of Physics Objects.
- Data fusion and deep learning:
  - Approaching the Problem as a Data Scientist;
  - Domain Knowledge Matters;
- Some ATLAS Calorimeter Related ML Applications and Possibilities:
  - Cell Energy Estimation with ML;
  - Electron Energy Calibration;
  - NeuralRinger;
  - Pre-processing considerations;
  - Mitigating statistical dependencies in Likelihood;
  - Data fusion using Expert Neural Networks;
  - Increasing TileCal Granularity with ML;
  - ML in Code Assertion for ATLAS TileCal;
  - Detector Anomalies;

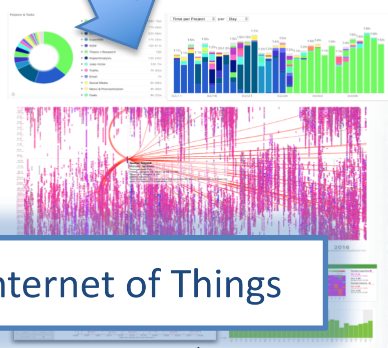
- Conclusions.



About a half a century ago, in a not so far away galaxy, large HEP experiments were precursors of some Digital Era challenges...

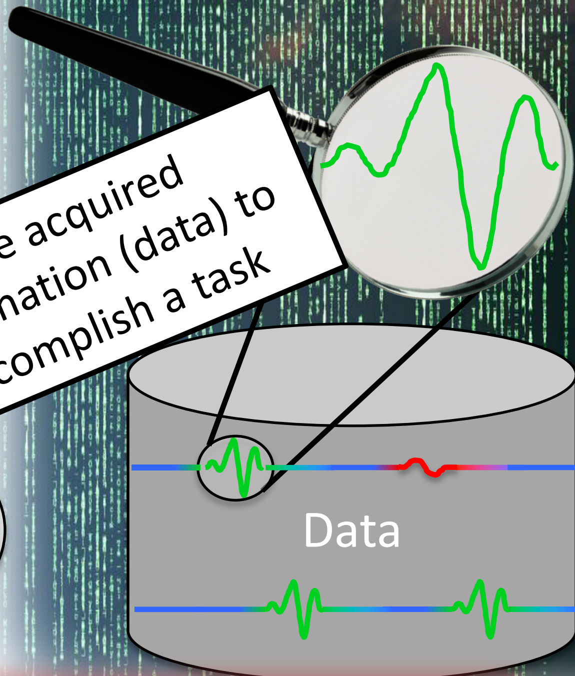
HEP Challenges

Services



Internet of Things

Digital Era = Information Era!



Some challenges in HEP experiments

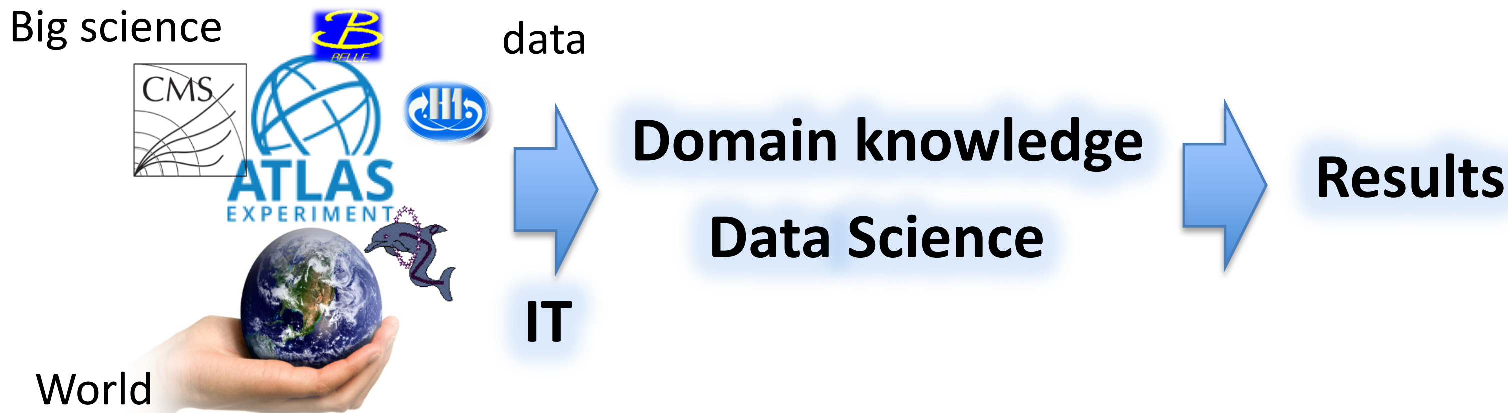
- High dimensionality
- Big Data
- Large amount of sensors
- Data Fusion
- Signal pile-up
- High event rate
- Rare Events of Interest

Medical screening of pathology risk

i.e.: seek events of interests

Electrical Network Transitories, Faults or Anomaly Detection

Solutions

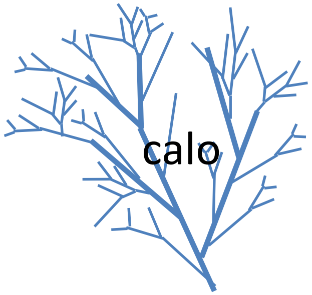


a movement in the direction of intensifying the exchange of data solutions between different fields (or services, products)!

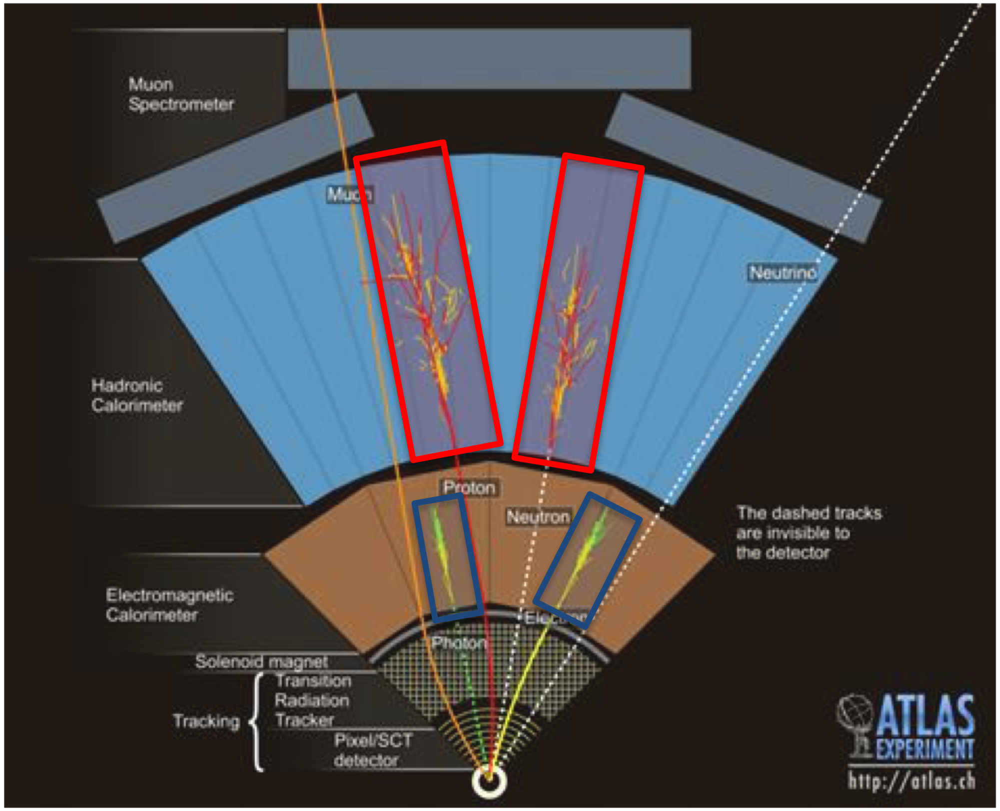
- Information on different domains can be represented in similar ways (time-series, images, distributions, fuzzy functions);
- Similar tasks (examples for calorimeter readings):
  - Classify: physics object type;
  - Regress: physics object energy;
  - Transcript: "a shower with symmetric and narrow energy deposition up to the second calorimeter layer. Mainly electromagnetic.".
- Common set of tools to manipulate/model/visualize data, i.e.:
  - Statistics (Inference, ICA, NMF etc.);
  - Machine Learning (SVM, NN, SOM).

Tools capable of doing that are not new, what changed?

Domain Knowledge: Calorimetry

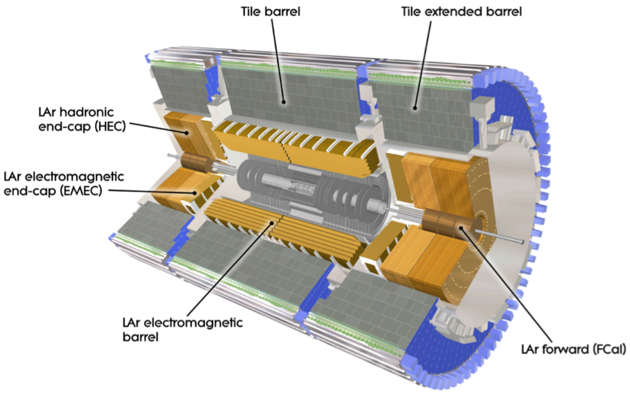


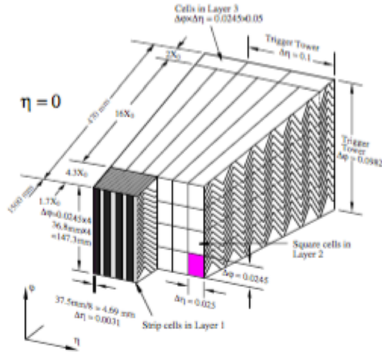
- **Particle shower**: process resulting from the interaction of some **physics objects** with the calorimeter;
- Culminates in a successive multiplication of the number of particles with lower energy in an **approximately conic geometry**;
- The calorimeter is instrumented to collect signals that are, as much as possible, directly proportional to the energy lost by the physics object in sensitive regions (cells);
- These signals are collected in pulses (time-series) that are used to perform many tasks, mainly:
    - Regression of the physics object 4-moments;
    - Classification of the physics object type.



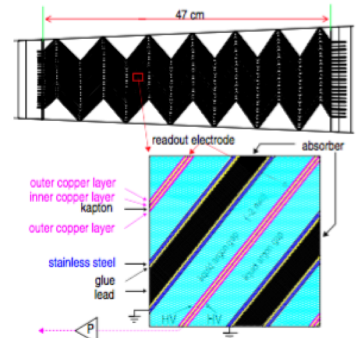
https://cds.cern.ch/record/1096081

Domain Knowledge: ATLAS Cal. Instrumentation

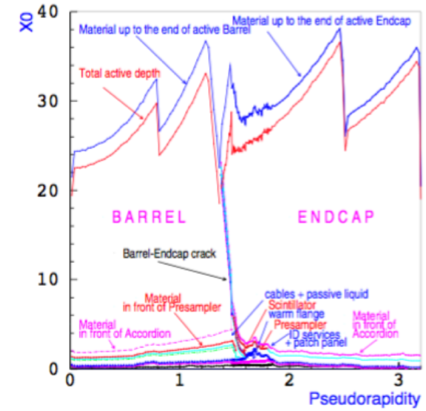




EM Instrumentation



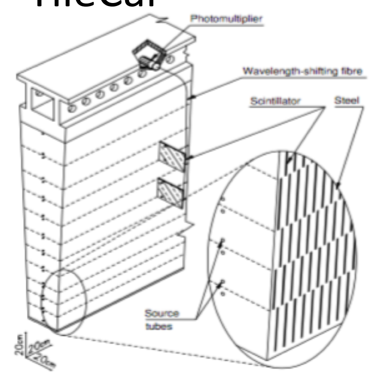
LAr Accordeon



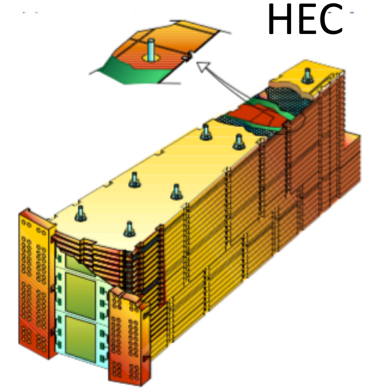
- Up to 7 longitudinal (physics object travel direction) samplings: 4 electromagnetic (EM, with one pre-sampler: not always considered as EM) + 3 hadronic (HAD);
- Different technologies/materials employed in the instrumentations (precision calorimeters);
- Different granularity (cell size)/samplings;
- The calorimeter instrumentation is symmetric (but with some non-uniformities) in any plane slicing ATLAS in a normal direction to the beam-axis and passing through it;
- But not if we slice it in planes parallel to the end-caps:
    - Changes in granularity;
    - Amount of material in the instrumentation.

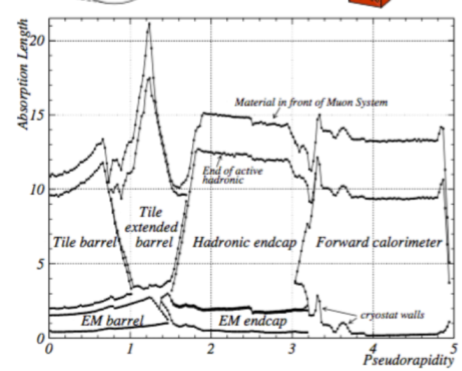
HAD Instrumentation

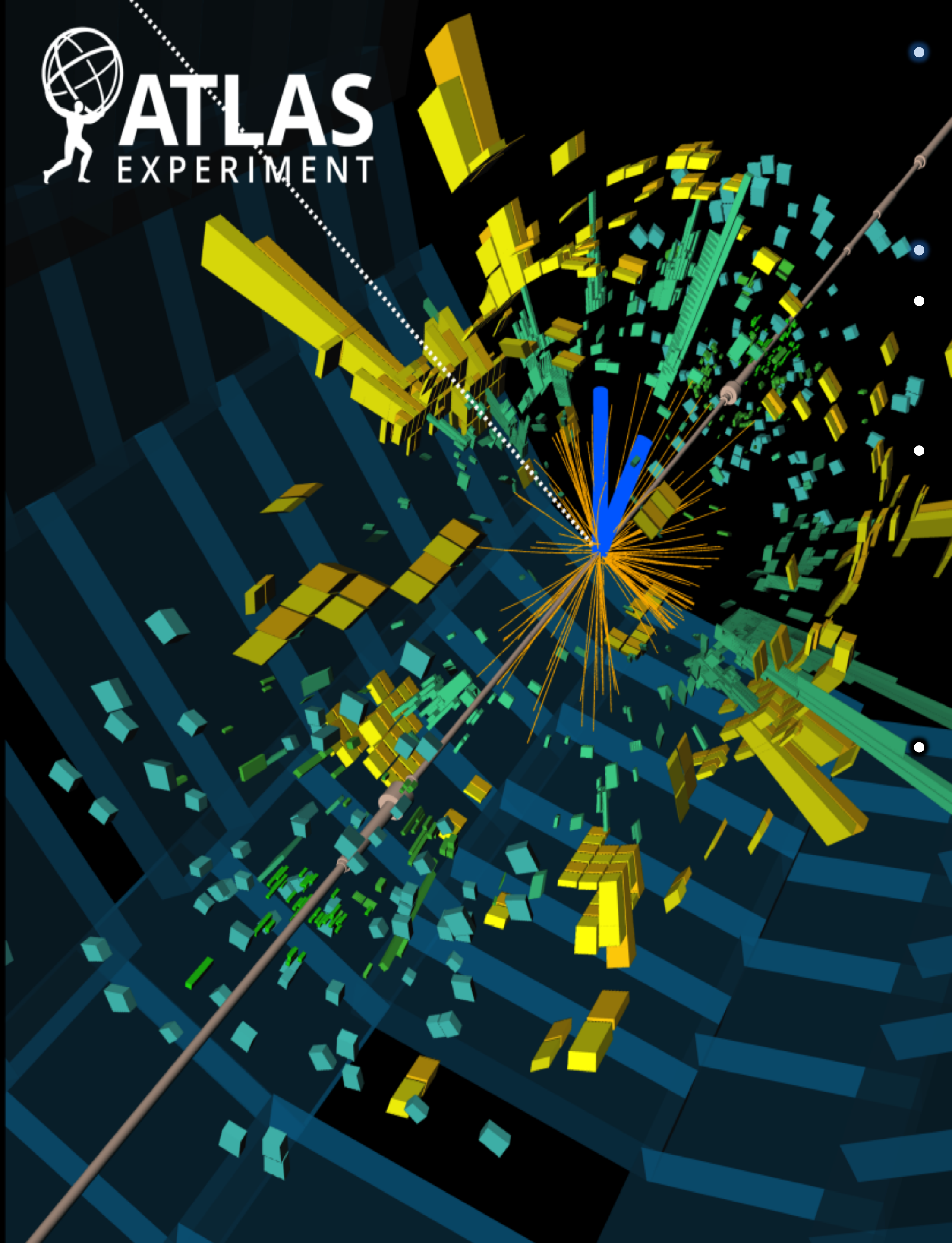
TileCal



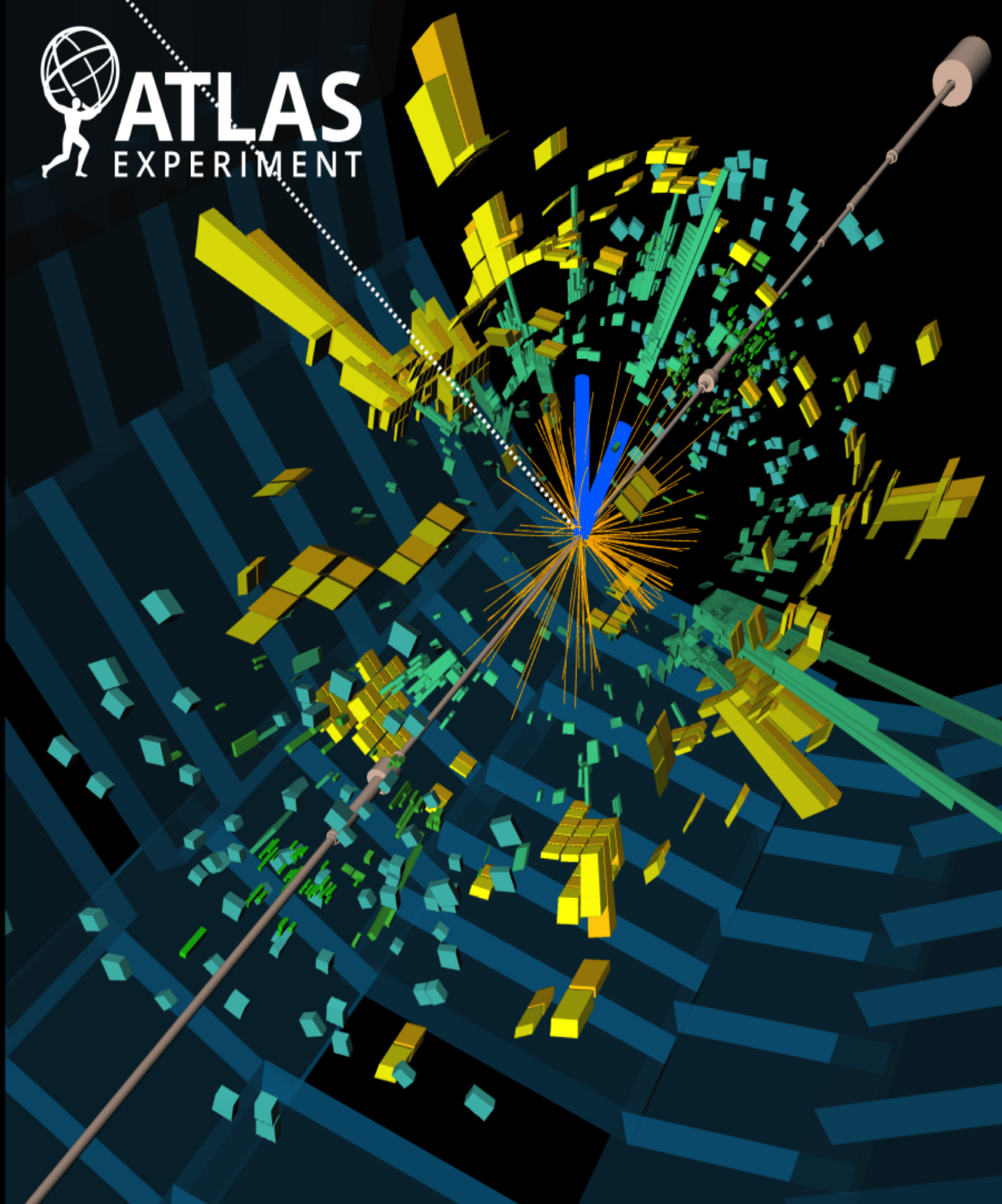
HEC





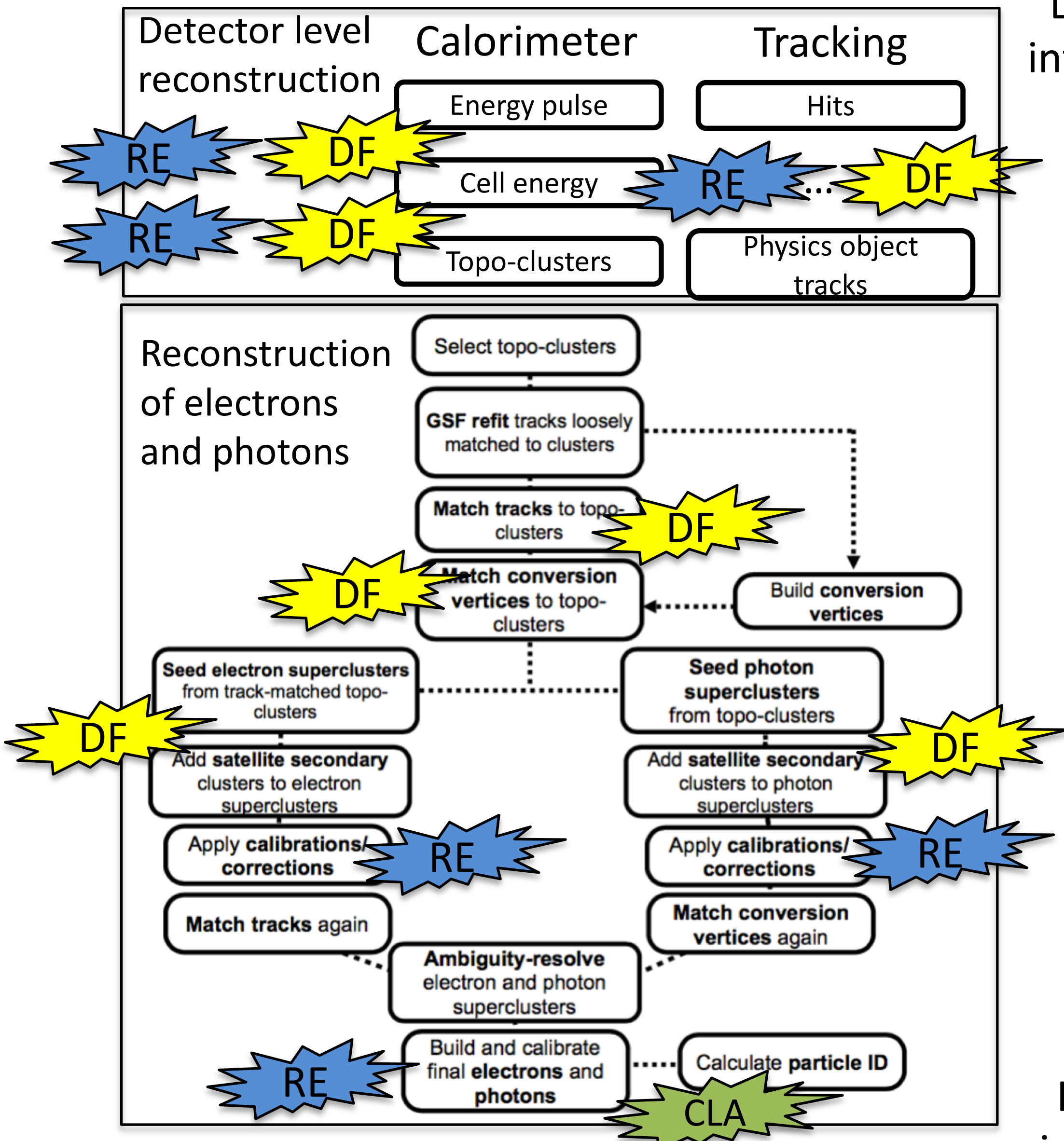


- Assume we have some expert knowledge about all the details on how we collected data in an HEP experiment;
- And we want to search for new physics;
- ATLAS calorimeter systems have 200k readout cells + information from many other systems (data fusion);
- One could process all readout information directly to search for a physics process:
  - I.e. search for the Higgs boson processing the pulses from 200k readout cells…
- Would require a lot of resources :
  - High-dimensional representation;
  - Rare observation of the Higgs boson decays:
    - ATLAS estimated 200 H->gg observations in its discovery.
  - High-input rate: recorded data contained 2.2 trillion bunch-crossing events;
  - Big data (2.6 PB).

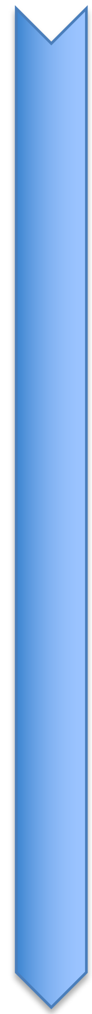


- However, we can consider:
- Readings are sparse;
- We have a set of physics object (electrons, photons, jets, taus, muons) that we know beforehand how they should behave;
- These physics objects can be part of our interesting new physics;
- Signatures left by physics process can be grouped together;
- Make our analysis easier: we reconstruct physics in steps, generating objects of gradual higher level of abstraction!

Domain Knowledge: Reconstruction of Physics Objects

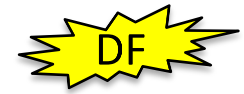


Low-level information

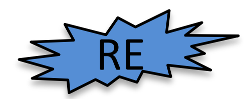


High-level information

Legend:

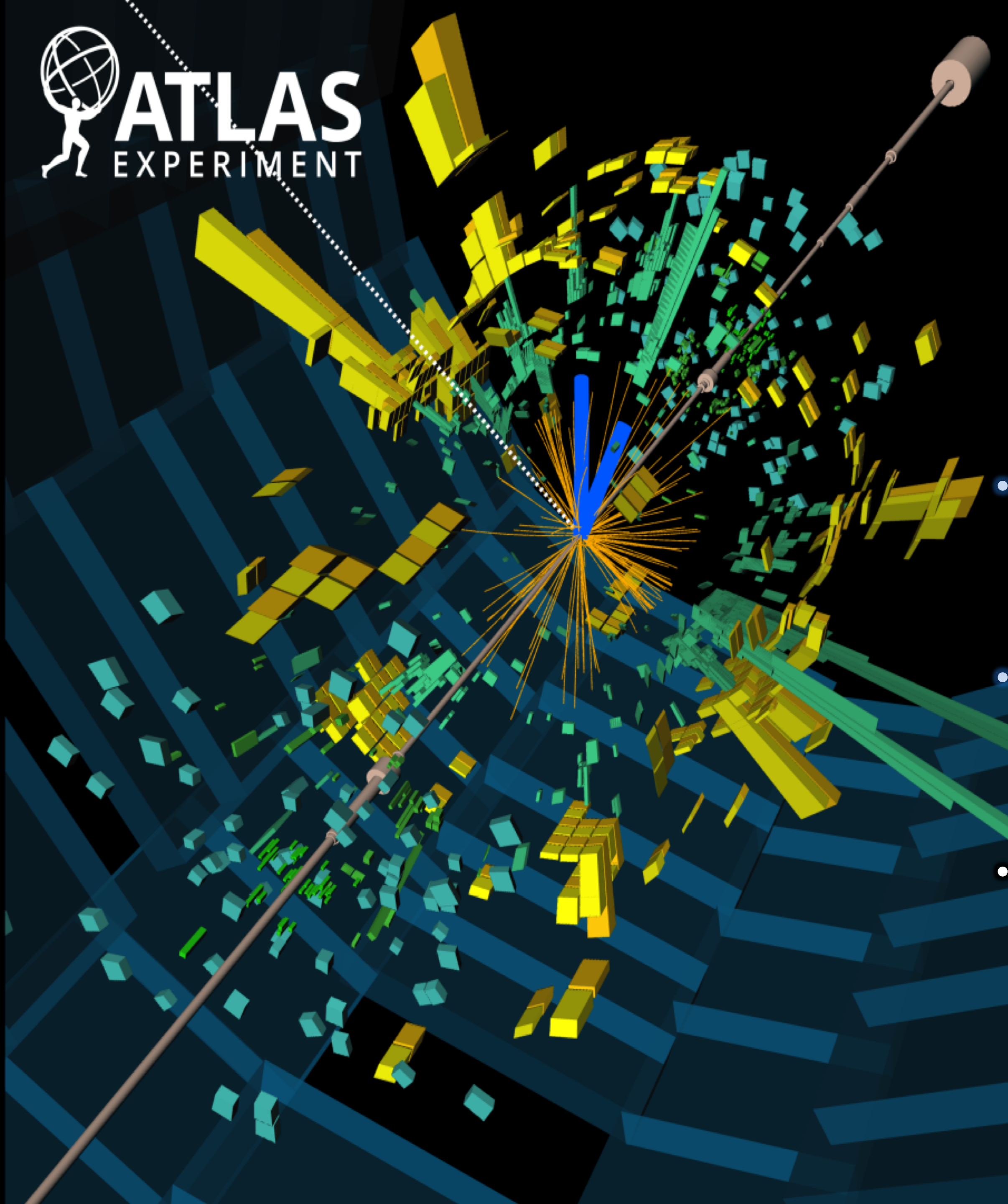


Data fusion: combining data to estimate or forecast a state of an entity.

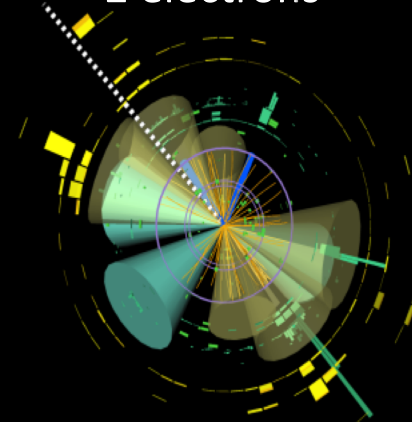


Regression: predict a continuous numerical value given input;

Classification: predict which of k categories the input belongs.



3 b-tag jets, 6 non b-tag jets, 2 electrons



- Make our analysis easier: reconstruct physics in steps, generating objects of gradual higher level of abstraction;
- Analysis can be performed using expert solution at each one of the tasks without needing to do it yourself!
- No need to be a complete expert in every task;

Deep Learning: New set of tools

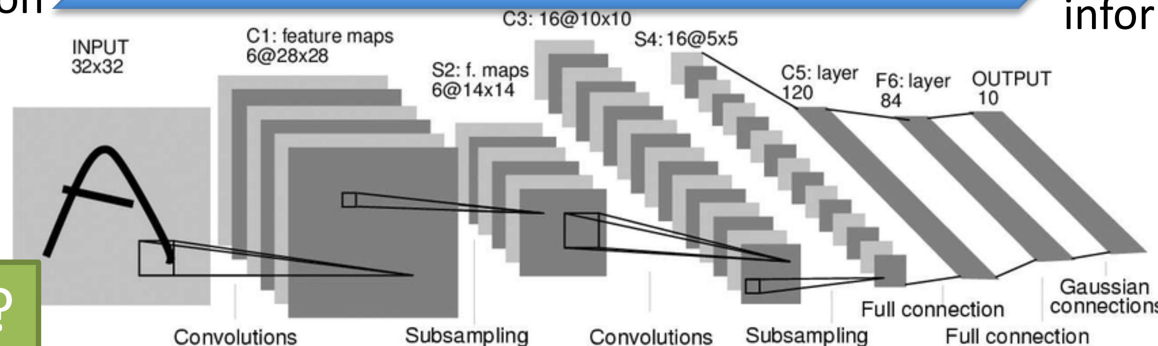
Energy pulse | Cell energy | Topo-clusters | Super-clusters

Patterns: Shower shapes | CaloRings

Models: Cut-based | Likelihood | MLP | BDT

Tag | Energy

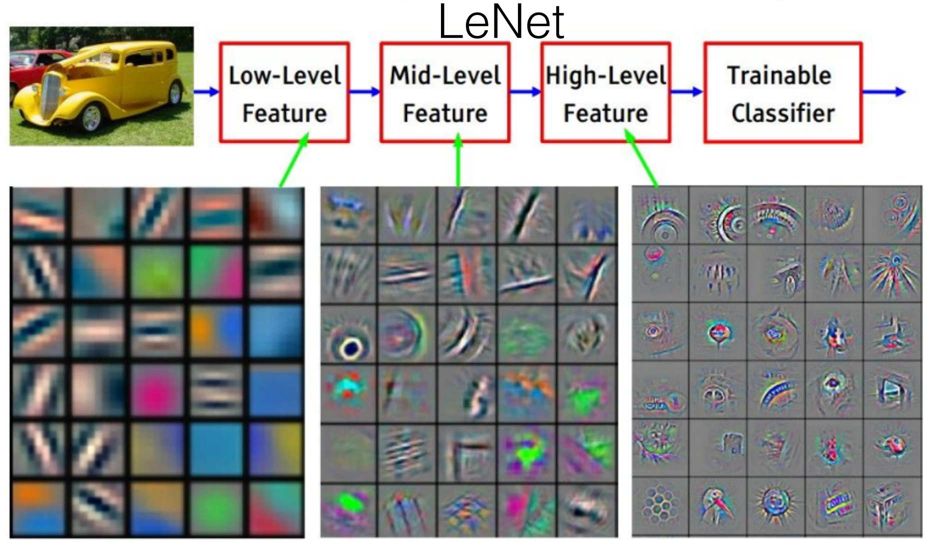
Low-level information → High-level information



LeNet

What changed?

- Found ways to improve learning process;
- Extract subsequently higher level of information abstraction.

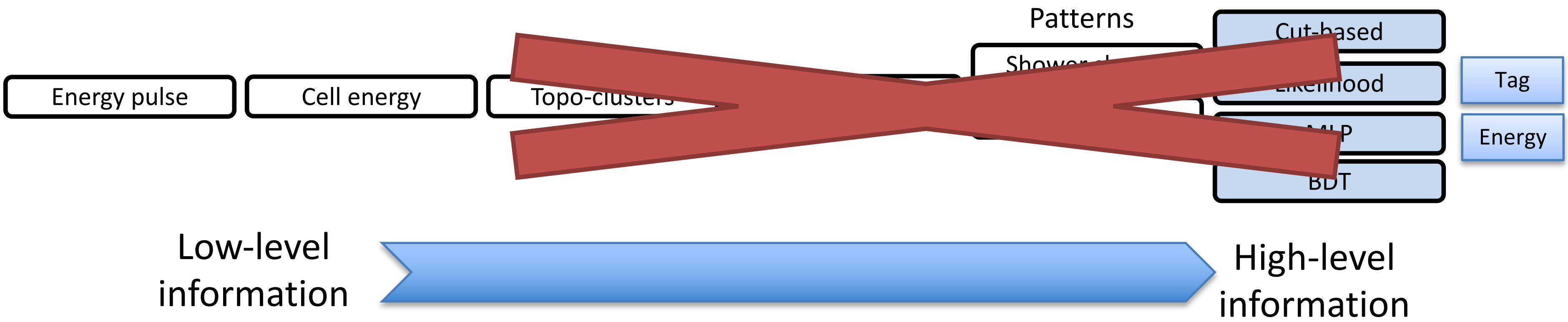


ImageNet (Zeiler & Fergus, 2013)

I.e.: In convolution neural networks (CNNs):

- **Parameter sharing**: use same parameters for several inputs;
- **Sparse representation**: consider only a small set of the inputs at a time;
- Also: 1D, 3D etc.

Approaching the Problem as a Data Scientist



How would a data scientist (with no feedback from domain knowledge) probably approach the problem:

If performance is better, then set to operate and publish!

But, what are the drawbacks?

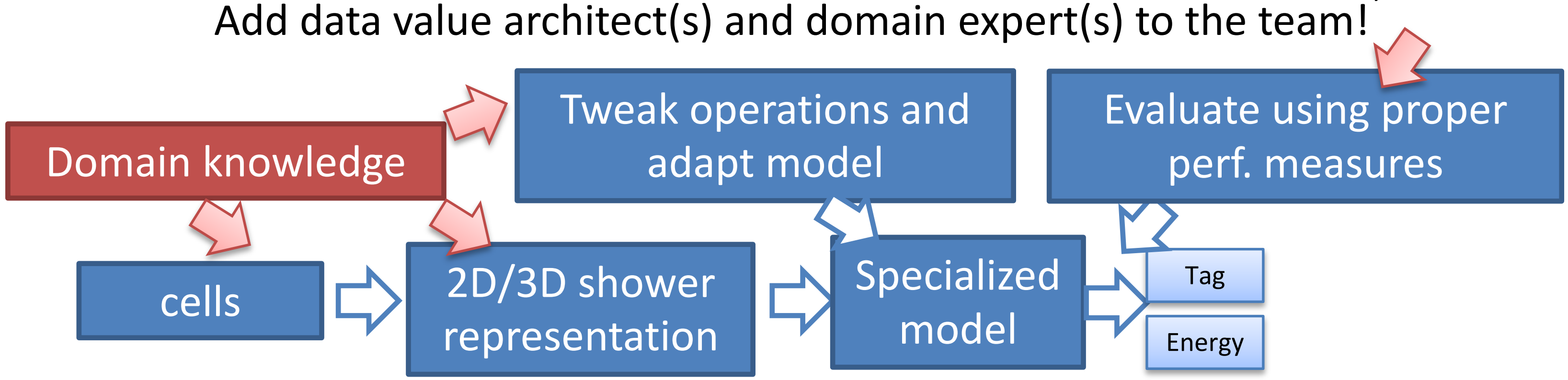
Domain Knowledge Matters

But, what are the drawbacks? (and many more)

- Domain knowledge is lost:
  - Probably the main reason for the feeling of using a black box solution!
- Fully counting on the operations computed by the model and the set of rules used for it to update its parameters;
- The rules are updated using measurements that may not consider all nuances involved in the final goal (physics analysis);
- Complex models are more plastic w.r.t. simpler ones:
  - Cross-validation can help to mitigate overfitting at where we have plenty of data;
- But, in HEP, models can be (and are) set to operate where we lack data…
- Risky to extrapolate model with no domain knowledge (operate it on other conditions then it was trained/evaluated data), specially plastic models.

How to approach the problem? (or adventure through other fields)

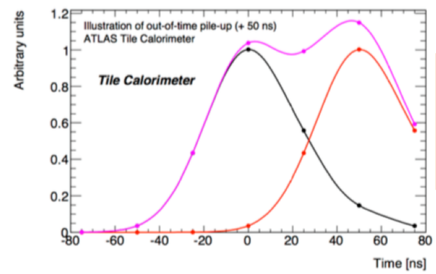
Add data value architect(s) and domain expert(s) to the team!



Deep Learning is not a One fit All Solution

- A lot of data is required to be able to successfully build the patterns with Deep learning;
- Learning process requires a lot of computational power;
- Demand specific hardware.

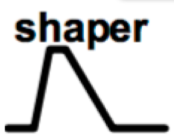
- I.e.: regression task with 7-dimensional input space for the ATLAS TileCal;
- Domain knowledge: task does not require fusing too much information.



Out-of-time pile-up @ +50 ns

Electronic Noise

Signal pile-up



Gaussian perturbation

Non-Gaussian perturbation

Energy pulse

Cell Energy

If only electronic noise:

- Shaper: makes the signal to have fixed size length;
- Remaining task: estimate amplitude;
- Optimum filter: Define a weighed sum that corresponds to the minimum variance under electronic noise only conditions;

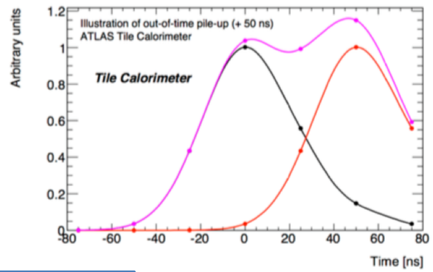
Current approach:

- Considers pile-up contribution as additional noise using the same approach;
- However: non-linear contributions (need to access high-order stats.)

Cell Energy Estimation with ML

"Nonlinear Correction for an Energy Estimator Operating at Severe Pile-Up Conditions" (2017)

- I.e.: regression task with 7-dimensional input space for the ATLAS TileCal;



Out-of-time pile-up @ +50 ns

Electronic Noise

Signal pile-up

Gaussian perturbation

Non-Gaussian perturbation

Energy pulse

Use Maximum Likelihood Estimation (MLE);

Estimate non-linearities with machine learning methods (shallow learning MLP)

Linear estimation

Cell Energy

\+

Cell Energy

- Allows access to linear estimation.

- ML **only learns to correct non-linearities** and high-order stats;
- MLP does not need to be dense: allows online operation (faster) and possibly near optimal perf.;

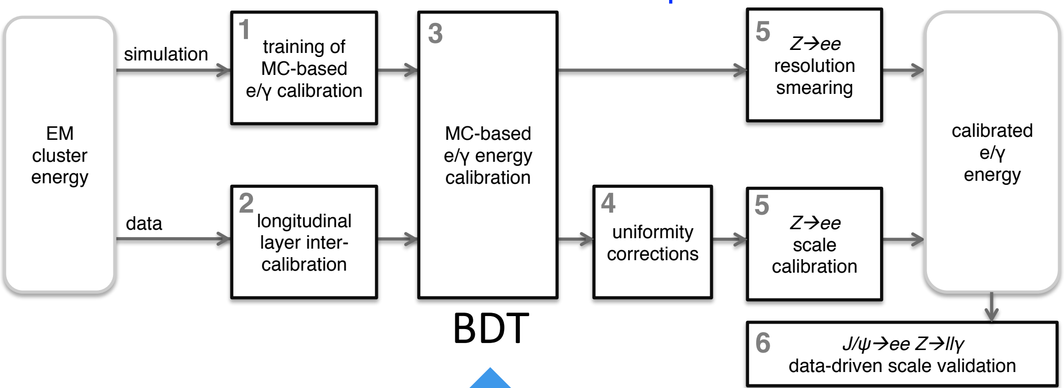
Further improvements could be achieved if proper handling data fusion strategy

Cell energy estimation may consider neighboring cells to improve estimation

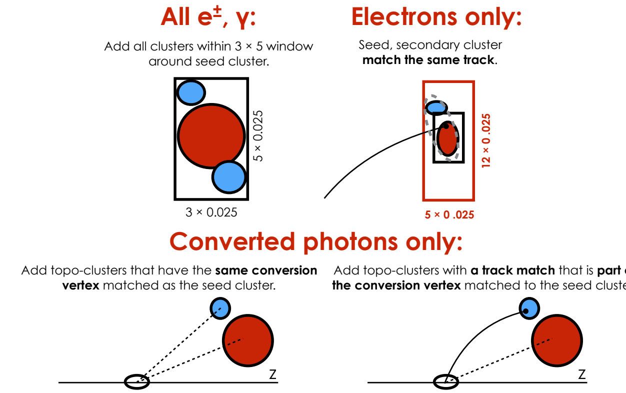
Electron Energy Calibration

ATL-PHYS-PUB-2017-022

Run 1
Calibration of EM particles:

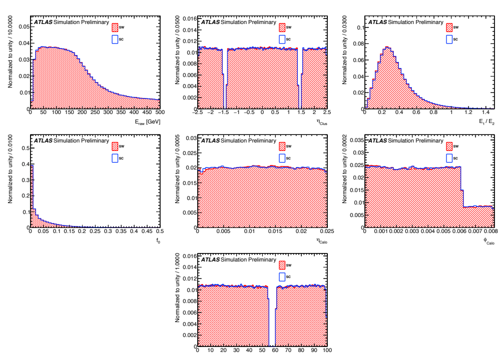


Run 2
Super-clusters

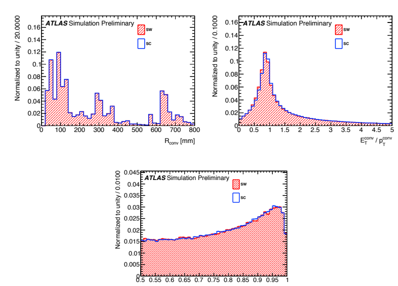


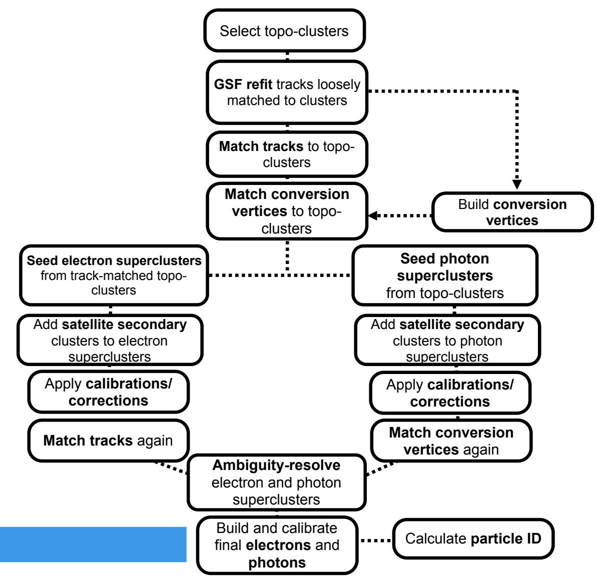
Expert knowledge to handle asymmetric shower development

Calo-based



Track-based





7 high-level* quantities

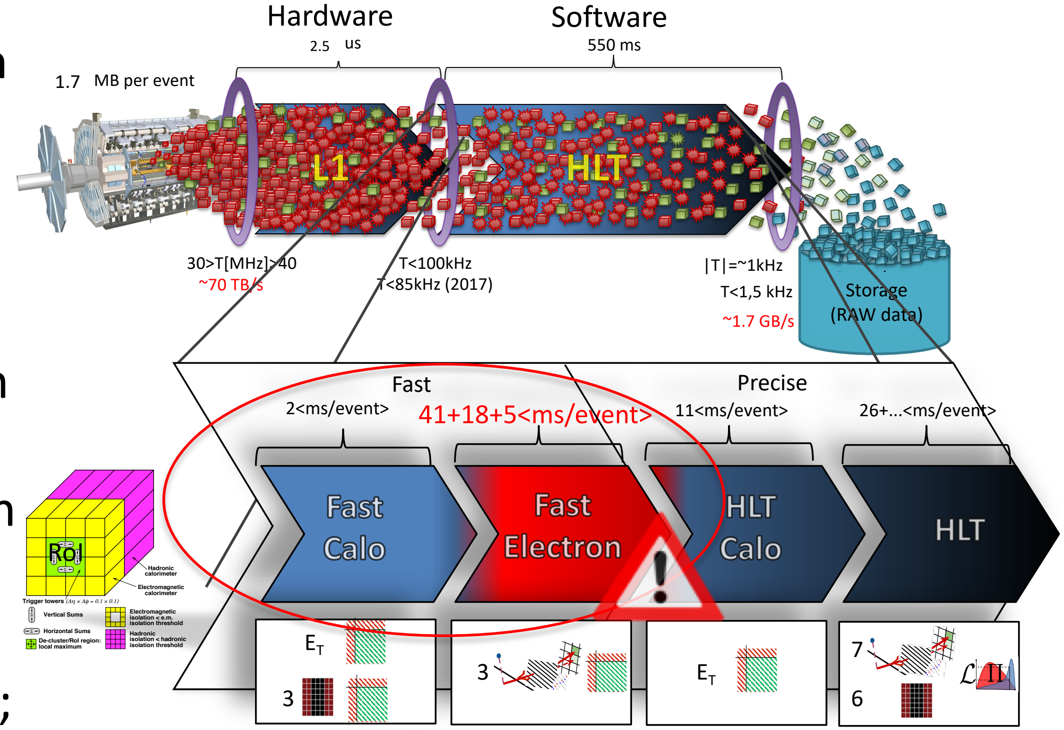
Super-cluster asymmetric shower improves quantities used on calibration

* High-level information in a context of data fusion, i.e. require many sensors to be obtained.

ATLAS Trigger System: 2017 Challenges

"Neural second-level trigger system based on calorimetry" (1996)

- Calorimetric information plays an important role for online reconstruction:
    - Fast readout and data manipulation (w.r.t. computing vision methods in tracking);
- In 2017, reprocessings showed an urgent need to reduce HLT processing demands (otherwise great impact to physics analyses);



Early 2017 nominal electron HLT

Electron chain:

- One of most relevant opportunities was at the FastCalo step (first HLT decision, calo-only):
    - NeuralRinger: Use of Machine Learning to reduce its processing rate!

Electron Identification (Fast step)

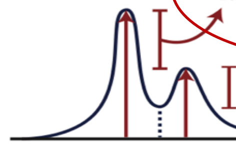
Old

Variables and Position

	Strips	2nd	Had.
Ratios	f_1, f_{side}	R_η*, R_ϕ	$R_{Had.}$*
Widths	$w_{s,3}$, $w_{s,tot}$	$w_{\eta,2}$*	-
Shapes	ΔE, E_{ratio}		

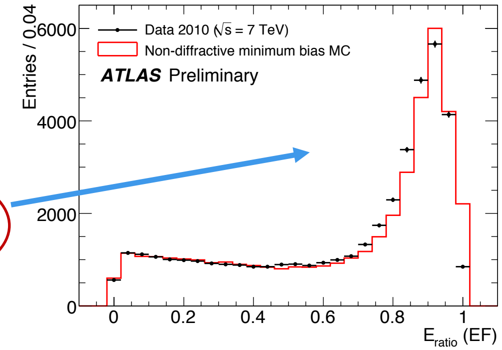
* Used in PhotonLoose.

Shower Shapes



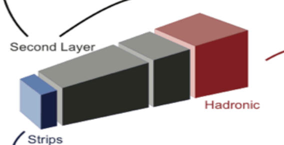
$$E_{\text{ratio}} = \frac{E^{S1}_{\max,1} - E^{S1}_{\max,2}}{E^{S1}_{\max,1} + E^{S1}_{\max,2}}$$

$$\Delta E = E^{S1}_{\max,2} - E^{S1}_{\min}$$



Energy Ratios

$$R_\eta = \frac{E^{S2}_{3\times7}}{E^{S2}_{7\times7}} \qquad R_\phi = \frac{E^{S2}_{3\times3}}{E^{S2}_{3\times7}}$$



$$R_{\text{Had}} = \frac{E^{\text{Had}}_T}{E_T}$$

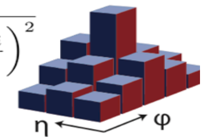
$$f_1 = \frac{E_{S1}}{E_{\text{Tot.}}}$$

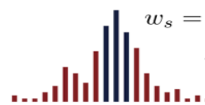
$$f_{\text{side}} = \frac{E^{S1}_7 - E^{S1}_3}{E^{S1}_3}$$

Widths

$$w_{\eta,2} = \sqrt{\frac{\sum E_i \eta_i^2}{\sum E_i} - \left(\frac{\sum E_i \eta_i}{\sum E_i}\right)^2}$$

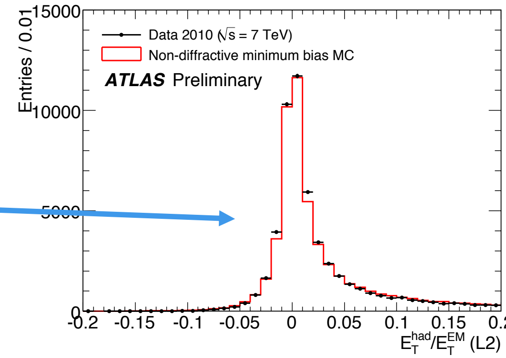
Width in a 3×5 (Δη×Δφ) region of cells in the second layer.



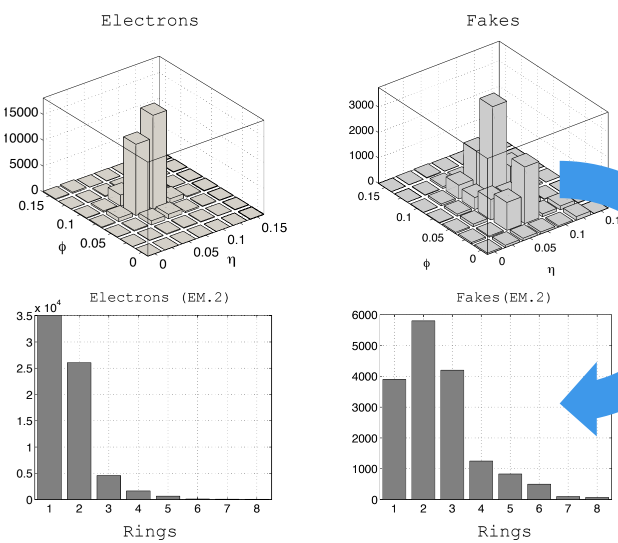


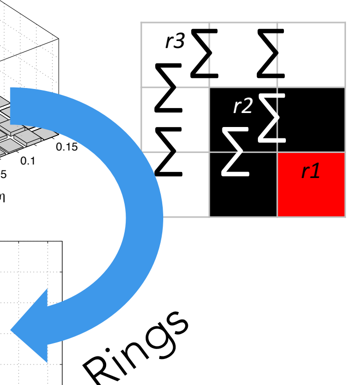
$$w_s = \sqrt{\frac{\sum E_i (i - i_{\max})^2}{\sum E_i}}$$

ws3 = w₁ uses 3 strips in η;
wstot is defined similarly, but uses 20 strips.



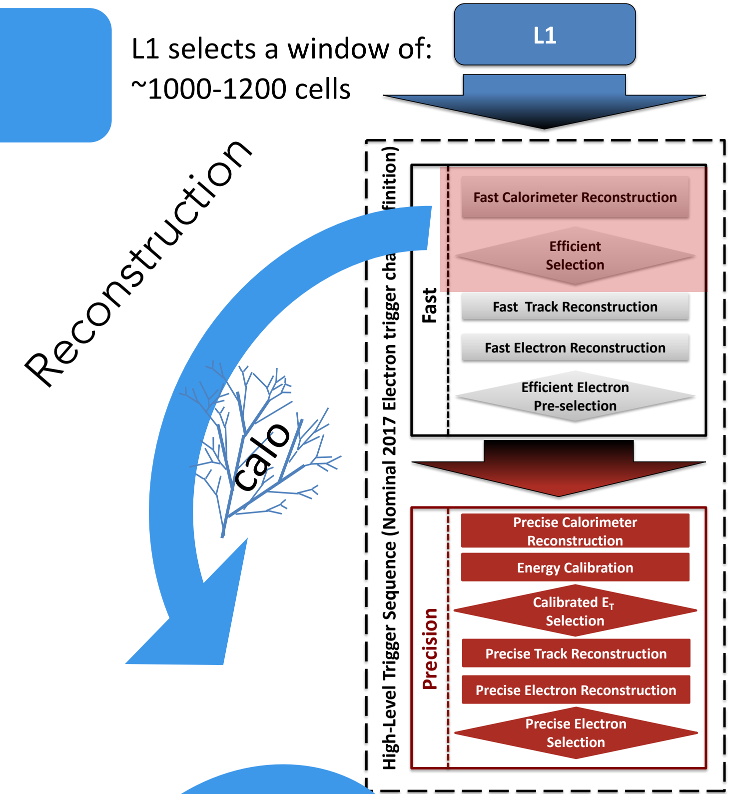
New





- **Ringer Shape:**
  - Concentric rings are built for all layers;
  - Compact cell information used to describe the event throughout of the calorimeter

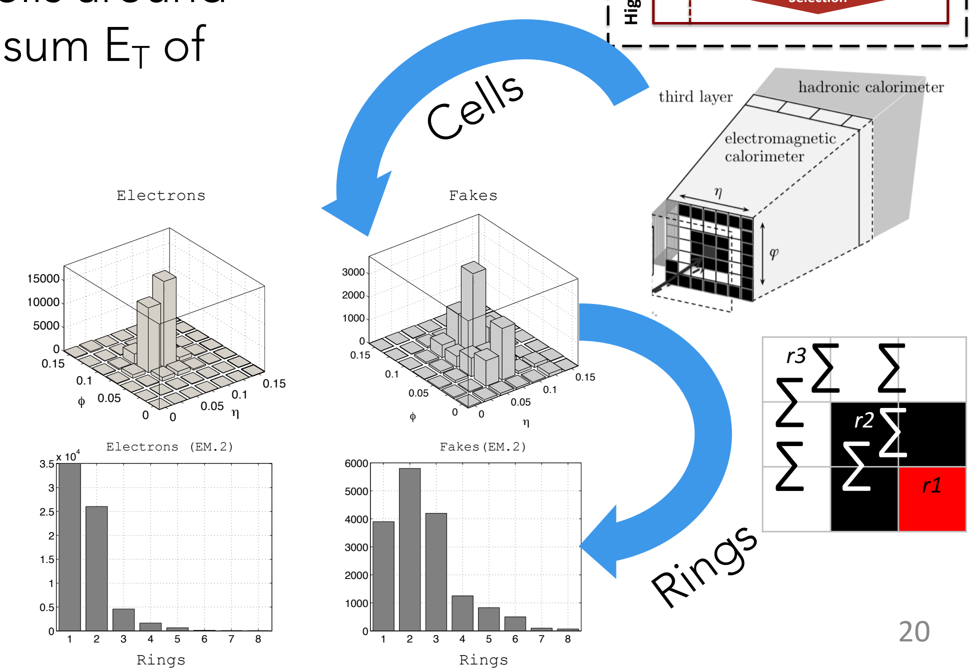
Ringer Reconstruction



- Explore approximately conic structure of the shower;
- Ringer reconstruction setup in the Fast Calorimeter Reconstruction:
  - Built from all calorimeter layers, centered in a window from the cluster barycenter;
  - First ring in each layer is the cell closest to cluster barycenter;
  - The next ring is the collection of cells around the previous one; ring value is the sum E_T of all cells composing the ring.

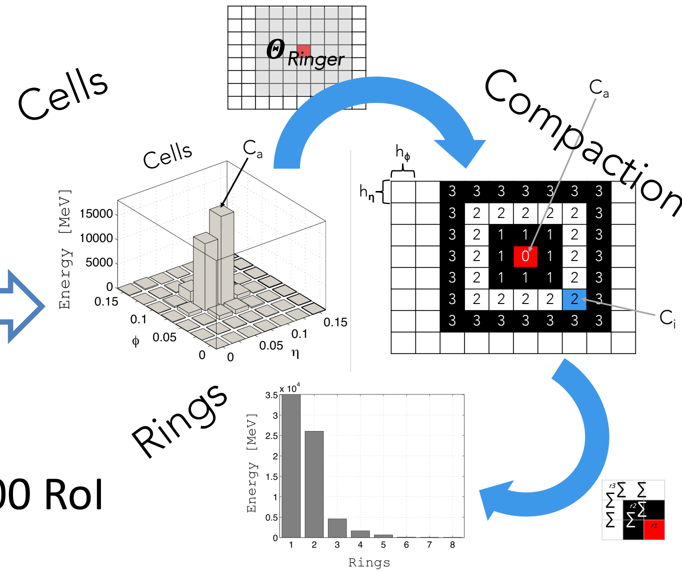
Total number of Rings per layer (covering 0.4 x 0.4 region in η x φ)						
PS	EM1	EM2	EM3	HAD1	HAD2	HAD3
8	64	8	8	4	4	4

Total: 100 rings



Rings: Machine Learning Point of View

cells → Approx. conic shower development: Specialist knowledge →



→ Hypothesis testing: Model → Tag

Advantages:

- ✓ Data compaction: typical 1000-1200 RoI cells to 100 rings;
- ✓ Dimension reduction using specialist knowledge: middle-term patterns are understood (when compared to pattern engineering obtained from deep learning):
  - ❑ May be helpful to understand deep learning patterns/model behavior;
- ✓ Keeps the physics interpretation: explores lateral/longitudinal information of shower developments, as standard shower shape quantities;
- ✓ Really fast computation (~100us).

Limitations:

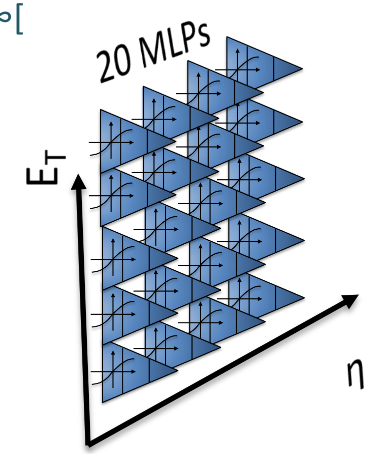
- ❑ Currently, it does not account for granularity changes w.r.t eta;
- ❑ Compaction comes with a cost: asymmetries in the shower development and other unknown discriminating features that could be built through deep learning are lost.

NeuralRinger Ensemble

- Domain knowledge (physics analysis): deal with changes in the detector response according to energy/position of the incident physics object by folding them in bins;
  - Pseudorapidity: Detector granularity, X0 etc;
  - Energy: Affects shower development;
- Rings are also subject to distortions as the standard shower shapes:
  - Use an ensemble of neural networks: natural way for approaching the problem (offline ID);
  - Fast decision computation: (~10us);
  - Other motivation: deal with big data.

Ensemble Composition

- Single-layer (tanh) fully connected MLP models
- E_T [GeV]= [15, 20, 30, 40, 50, ∞[
- η = [0, 0.8, 1.37, 1.54, 2.5]
- Event outside bins use nearest MLP to extrapolate;
- Single Output node (tanh):
  - Electrons: +1;
  - Background: -1.



MLP Training

- Use of shallow learning MLP (1 single layer)
- Trigger operation without an offline counterpart using similar method;
- Reduce problems when extrapolating to higher energy operation;
- More difficult to evaluate systematic effects on rejected samples.

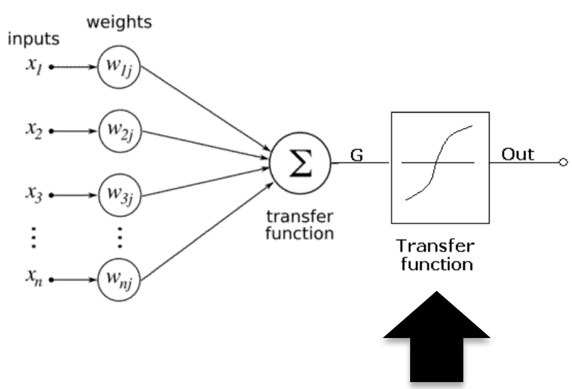
Tuning Procedure/(Hyper)Params

Pre-processing

- Use specialist knowledge: normalization by the absolute sum of the rings energy;
- Simple, efficient and aparametric approach;
- Keeps shower shape lateral/longitudinal profile easy to understand;

Decision Making

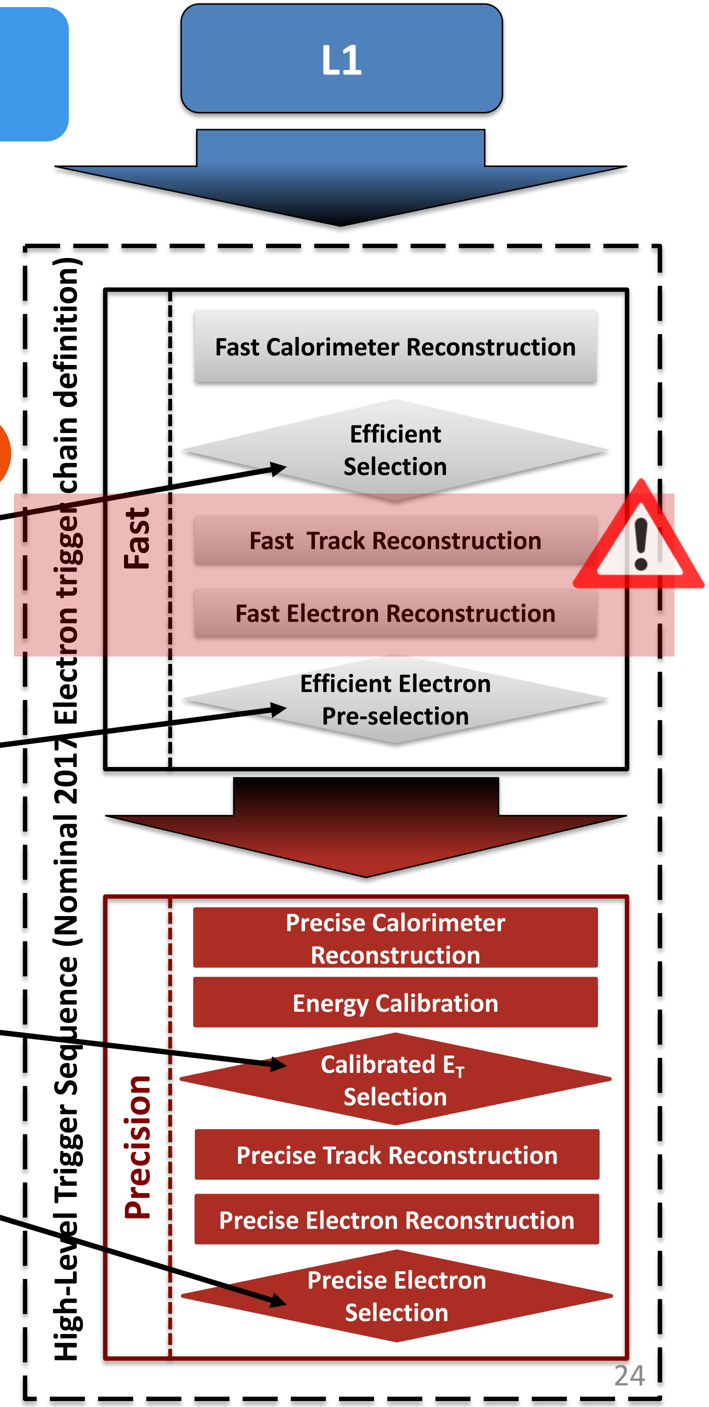
- *Set the non-linear transfer function (tanh) in the output neuron as linear for operation.*
- Then: MLP can be used to apply pile-up correction by computing the threshold as a linear function of a global pile-up estimation (mean number of collisions);



This was set to a linear function

HLT e/g Workflow

Old	New
Fast Shower Shapes reconstruction	Ringer Reconstruction
Cut-Based Selection	Ensemble of neural networks + Pileup correction
Track reconstruction	
Track cuts	
Precise Shower Shapes Reconstruction	
Energy Calibration	
Precise E_T cut	
Precise track reconstruction	
Electron Identification based on the Likelihood at relevant quantities (calo+track) + Pileup correction	

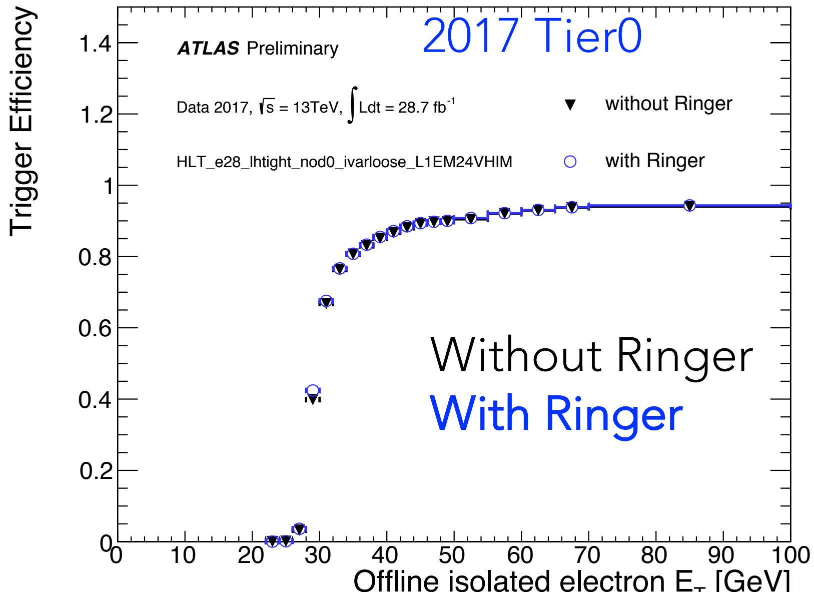


Trigger Efficiency

Operation (2017 collision data):

e28_lhtight_nod0_noringer
- Used a backup trigger with the previous cut-based selection to assess:
  - Efficiencies changes;
  - Impact in the offline (T&P systematics).

- Clean unbiased samples given by the **tag & probe** method;
- Kept HLT signal efficiency unchanged after the switch in early 2017:
  - Estimated primary chain latency reduction: ~**200 ms to ~100 ms**;
  - Higher rejection power (~2-3X);
- Estimated **electron + photon slice: ~1/4 latency reduction**;



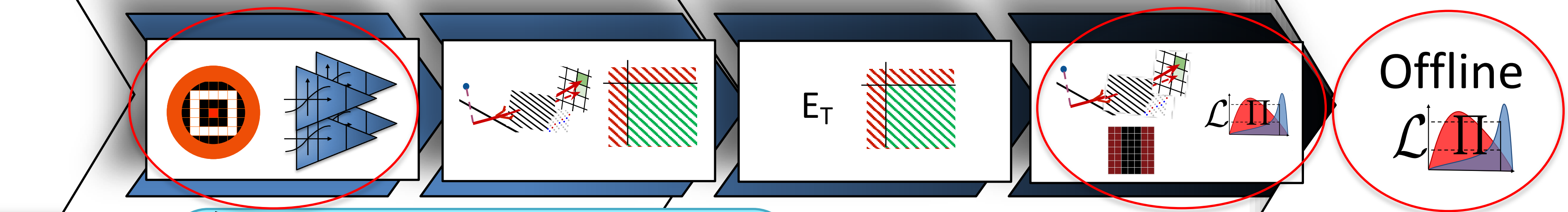
https://twiki.cern.ch/twiki/bin/view/AtlasPublic/EgammaTriggerPublicResults#Performance_of_Ringer_in_Trigger

Trigger as a Hybrid Method

Fast

Precise

All algorithms must be consistent: offline is the reference for analysis



NeuralRinger

Ensemble of Neural Networks (2017 for E_T > 15 GeV): reduces HLT farm processing demands

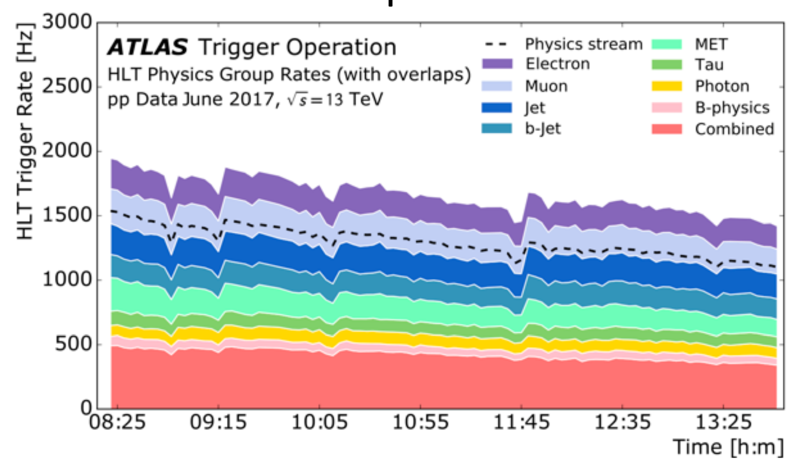
Likelihood

Final HLT decision (2015): major determinant for the output rate

HLT Farm @ **electron** + photon slice

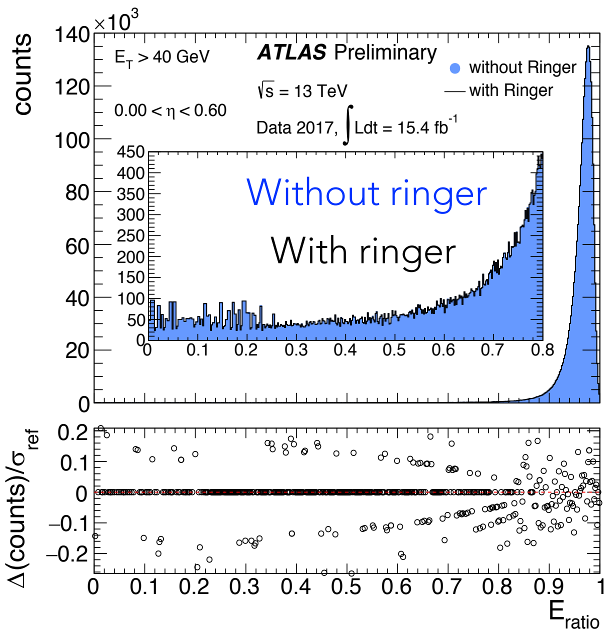
-25%!

Output rate:



Offline Impact

Probe profile due to changes in Z Tag systematics



$E_{ratio} = (E^1_{max1} - E^1_{max2})/(E^1_{max1} + E^1_{max2})$

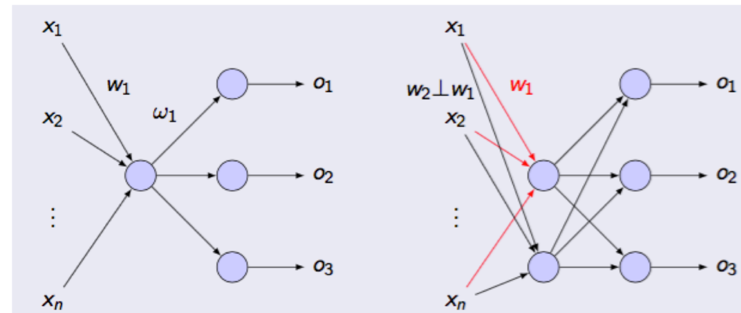
- ❑ To verify impact on the offline systematics after the introduction of the ringer in the trigger sequence:
  - ✓ We assess the Δ(counts)/σ (~chi residuals in black markers) where the ringer histogram is used as a model to the baseline histogram (experimental outcome).
- ☛ Residuals are small and oscillate freely around zero, which suggests absence of bias.

https://cds.cern.ch/record/2629408/files/ATL-COM-DAQ-2018-120.pdf

Pre-processing

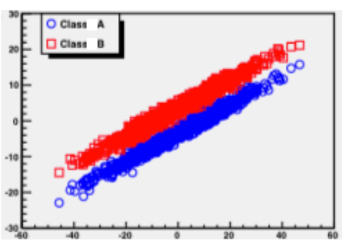
- Rings are statistically dependent: we could further compact our dimensional input through pre-processing:
  - Pre-processings can make the feature space more discriminant.
- Well-known procedure is to use **Principal Component Analysis (PCA)** to decorrelate information and keep only a fraction of the original variance;
  - Optimal for Gaussian processes;
- **Principal Components of Discrimination (PCD)** concentrate information based on their discrimination power;
  - More suitable for classification tasks;

One way of obtaining PCDs using MLPs

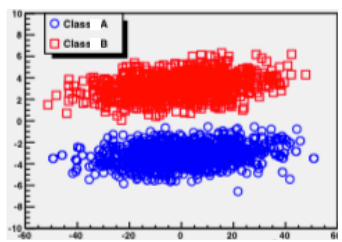


Toy with two 2D gaussians

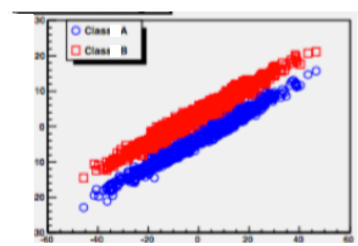
Original base



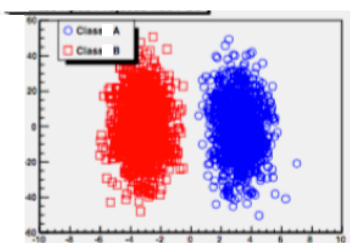
All PCAs projection

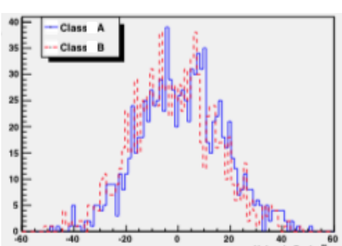


Original base

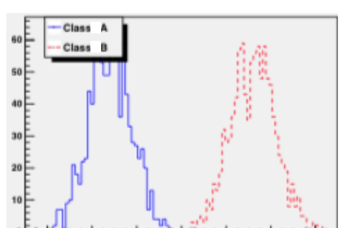


All PCDs projection



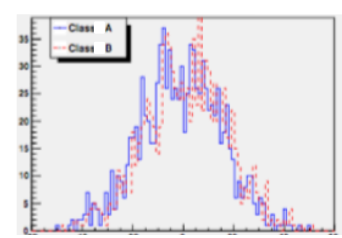


1st PCA projection



2nd PCA projection

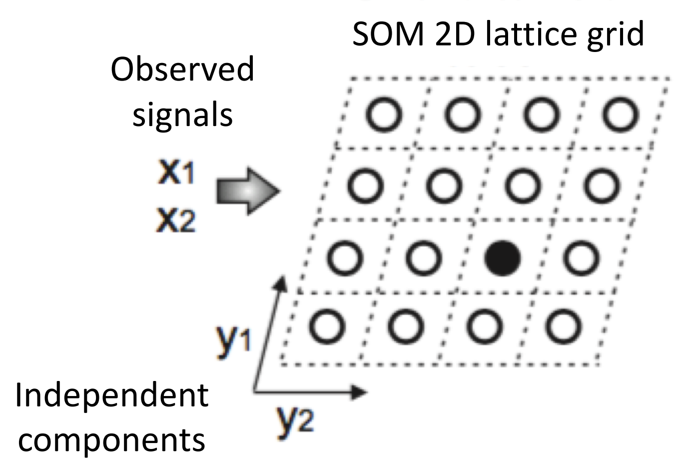
1st PCD projection



2nd PCD projection

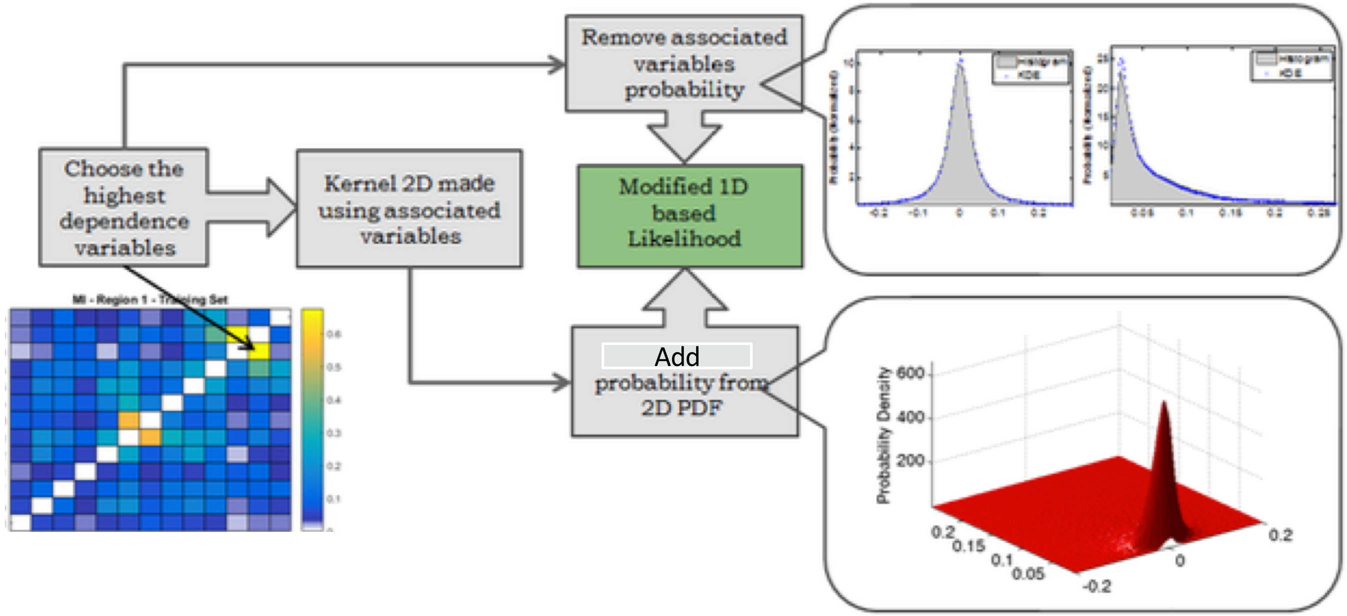
Pre-processing

- **Independent Component Analysis (ICA)** seeks statistical independence rather than data uncorrelation:
  - I.e.: retrieve the independent sources that can generate the rings (or shower shapes) by mixing them;
- Considers higher order statistics, being more suitable than PCA for non-gaussian processes;
- Variants consider the separation of **independent sources non-linearly (NLICA)** mixed, i.e.: use self-organizing maps (SOM) lattice positions;
- Another approach **NMF (Non-negative Matrix Factorization)** is to optimize new basis divergence value at some non-negative measure.



Can use SOM lattice GRIDs > 2D if suitable;

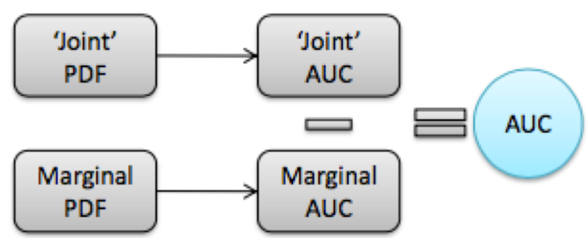
Mitigating Dependencies in Likelihood



- How to evaluate dependency:
  - Using mutual information

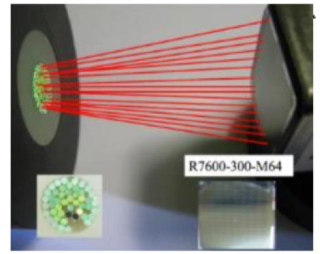
$$I(X;Y)=\underbrace{H(X)}_{\text{Entropy}}-\underbrace{H(X/Y)}_{\text{Conditional Entropy}}$$

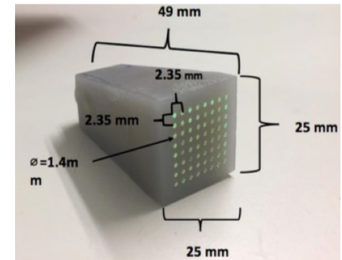
  - Considering effects on AUC

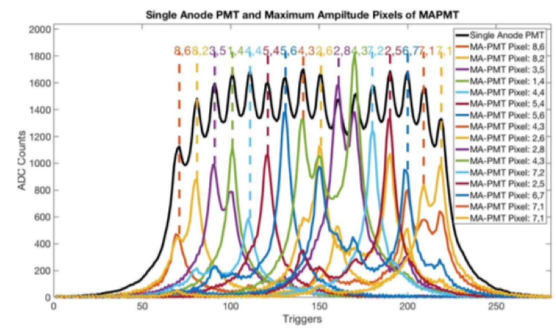


Increasing TileCal Granularity with ML

- Under severe pile-up conditions, detector granularity can be essential to successfully perform the required tasks;
- Increase granularity without changing the mechanical structure of the detector;
- Use a multianode 8x8 signals;
- NMF+MLP and CNN with very near efficiency (dataset with ~120 samples);
- Use GAN to increase statistics;
- Results (evaluated CNN only) suggest that a 2x granularity is feasible:
    - 4x in the barrel? To be investigated.



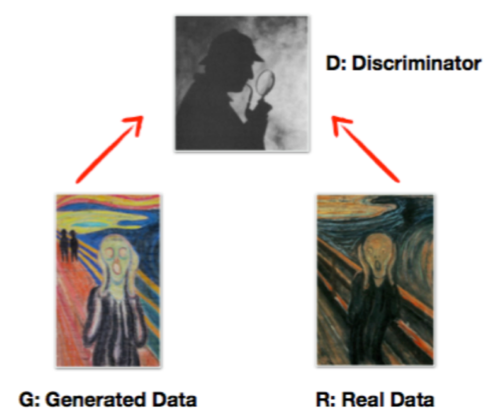




MinMax Game

$$J^{(D)} = -\frac{1}{2}\mathbb{E}_{x\sim p_{data}} \log D(x) - \frac{1}{2}\mathbb{E}_z \log(1 - D(G(z)))$$

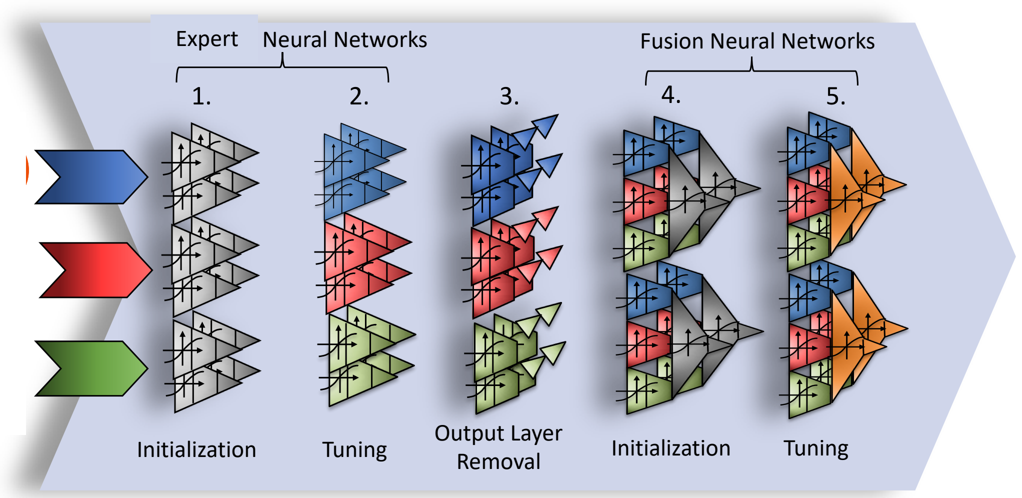
$$J^{(G)} = -J^{(D)}$$



Data fusion using Expert Neural Networks

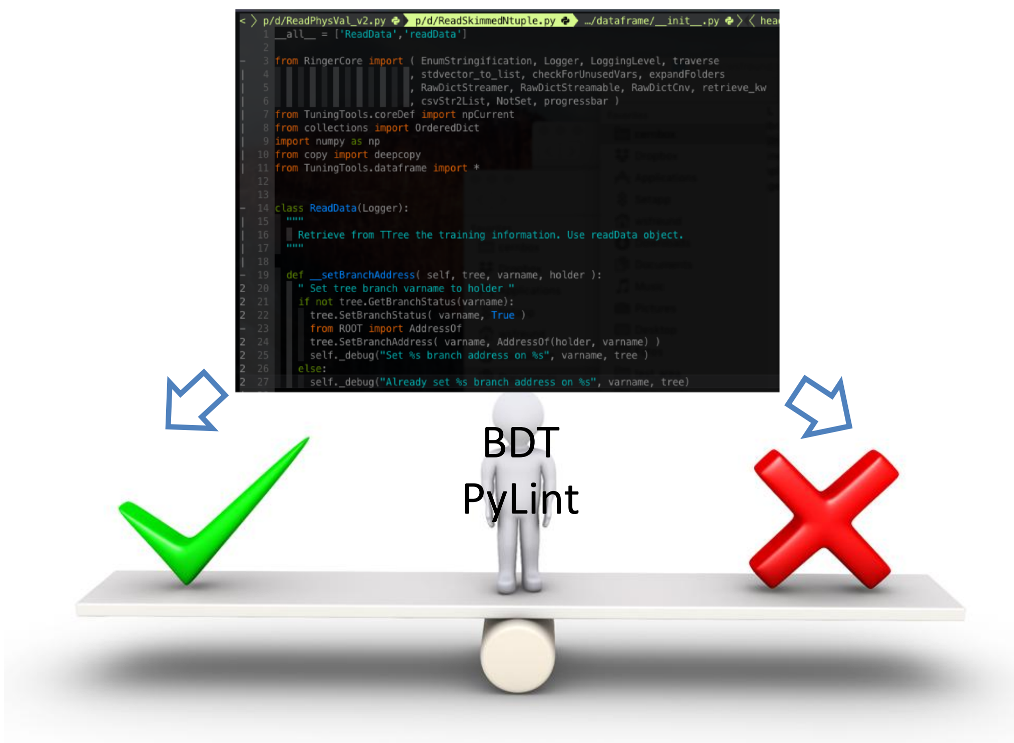
"Particle Discrimination using Matched Filters and Expert Neural Networks" (1999)

- The concept of considering sparse connections may take advantage of construction frontiers;
- ATLAS TileCal discrimination capabilities was considered using a prototype;
- Prototype had 2 longitudinal samplings:
  - Input space are the cells of the prototypes;
  - Use one expert MLP to each sampling (input dim: 200, 46);
  - Fuse their information using the hidden layers outputs with another MLP;
- Objective is to extract expert features and then to make classification upon this features.



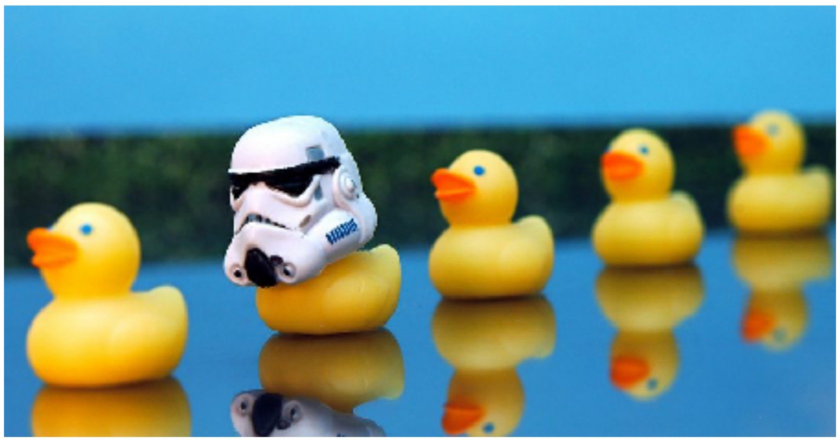
ML in Code Assertion for ATLAS TileCal

- TileCal has its own web-based collaboration development tool:
  - Integrates data-quality, calibration, developments;
- To avoid overloading the main server, the jobs are sent to other servers;
- These jobs can contain flaws that could waste resources:
  - i.e. make serves unavailable for essential services;
- The system uses a BDT upon a set of code quality factors to indicate whether the code is executable or not executable;
- Improve results w.r.t. static code assertion (pylint).



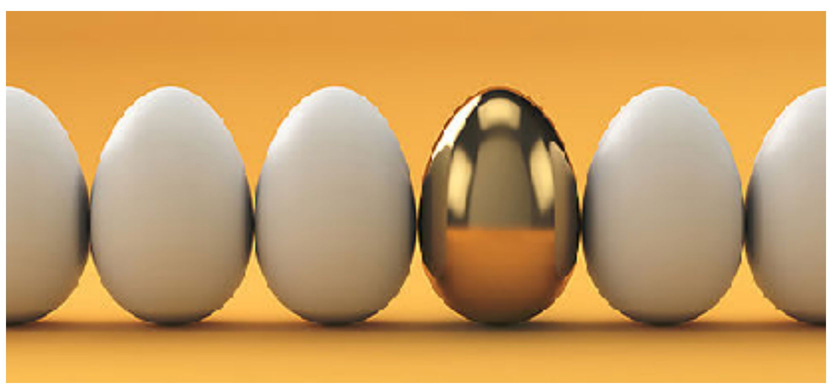
Anomaly Detector

- Aonamly dteetcion;
- Japanese ancient proverb: 出る 杭 は 打た れる;
- Or: Nails that stand out are hammered;
- Examples of special nails (subjective):



“Pile-up”: same class

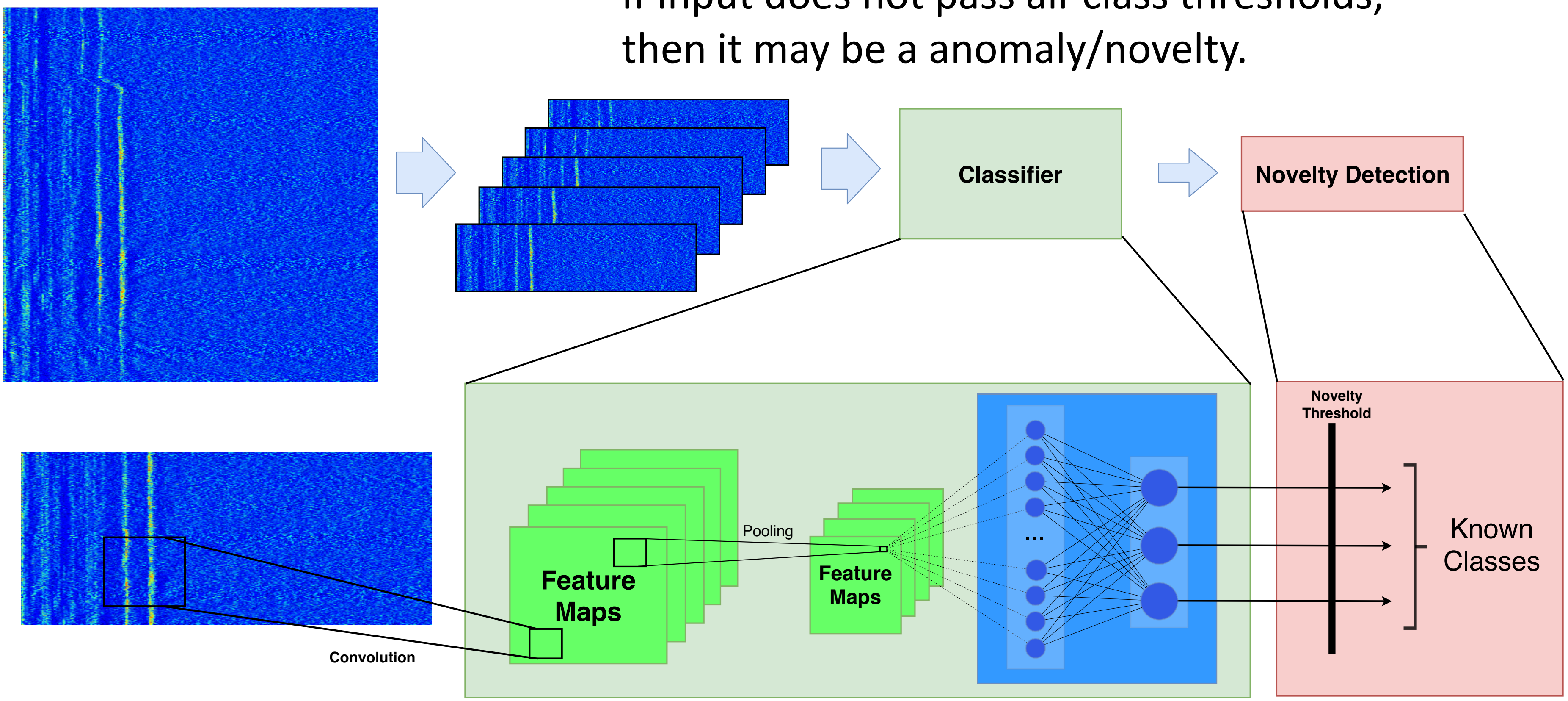
Anomaly detector: same class



Novelty: new class

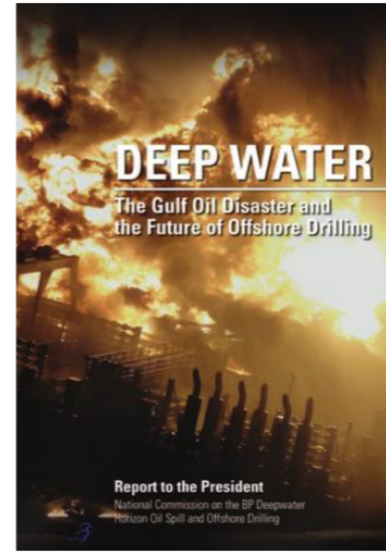
Anomaly Detector

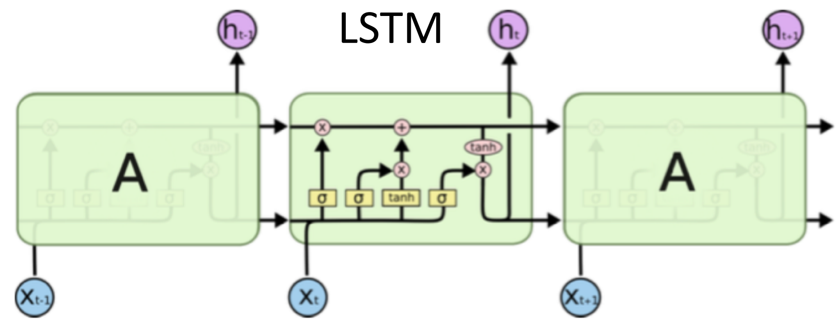
- Simple but efficient approach: add a threshold to every known class;
- If input does not pass all-class thresholds, then it may be a anomaly/novelty.



Fault Detection

- Failures produce a negative impact on the safety of any process plant;
- The consequences of a gross accident are even more serious (Himmelblau, 1978);
- I.e. (Sutherland, 2016):
    - Eleven workers lost their lives.
    - Sinking of the Deepwater Horizon rig.
    - Massive marine and coastal damage.
    - One of the largest environmental disasters in US history;
- **Problem**: traditional Fault detection and diagnosis (FDD) methods are unable to consume the huge amount of the data available nowadays;
- **Common phenomenon**: Large volumes of data with very little information (Dai, 2013);
- **Opportunity**: Investigate the use of Long Short-Term Memory (LSTM) recurrent neural network in FDD applications.





- Input is considered as sequential data (usually causal relationship);
- **Parameter sharing** (same weights applied to every input);
- Add paths to make **learning gradients to flow for long durations** by using a dedicated unit: allow **time-scale (range)** integration to be **adjusted dynamically**.

Conclusions

- HEP Experiments, Data fusion and Deep Learning:
  - Physics reconstruction builds objects of gradual higher level of abstraction by sequentially fusing more information together:
    - No need to handle all information at once, makes process easier;
  - Deep learning models usually proceeds in similar fashion, making learning process easier and more likely to be successful:
    - One key element is parameter sharing (but beware, not always a reasonable priori);
  - LHC upgrades with severe signal pile-up conditions may require to revisit how we perform data fusion.

Conclusions

- Use of Machine Learning:
  - Domain knowledge cannot (or shouldn't) be replaced:
    - Never forget model extrapolation;
    - A more powerful approach can be formulated if considering particular information already known (no need to approximate it);
    - Machine learning should approximate what is needed: think of ways to add prior information to the model learning process and you will probably end up with a less complex model (more reliable) and eventually more efficient;
  - Build a multidisciplinary team and collaborate -> benefit of collective knowledge:
    - Other field solutions may be pretty useful for the problem and eventually domain experts might not even know that it exists (that's what the data scientist and data value architect are there for);
    - But usually other field solutions will need to be adapted to your needs, that's where a good teamwork takes place.

That's all for today

Backup