Machine Learning in HEP : trends and successes

David Rousseau

LAL-Orsay

rousseau@lal.in2p3.fr @dhpmrou

CosmoStat, CEA-Saclay, 24th Jan 2019

Outline

- ML basics
- ML in reconstruction
- ML in analysis
- ML in simulation
- Wrapping up

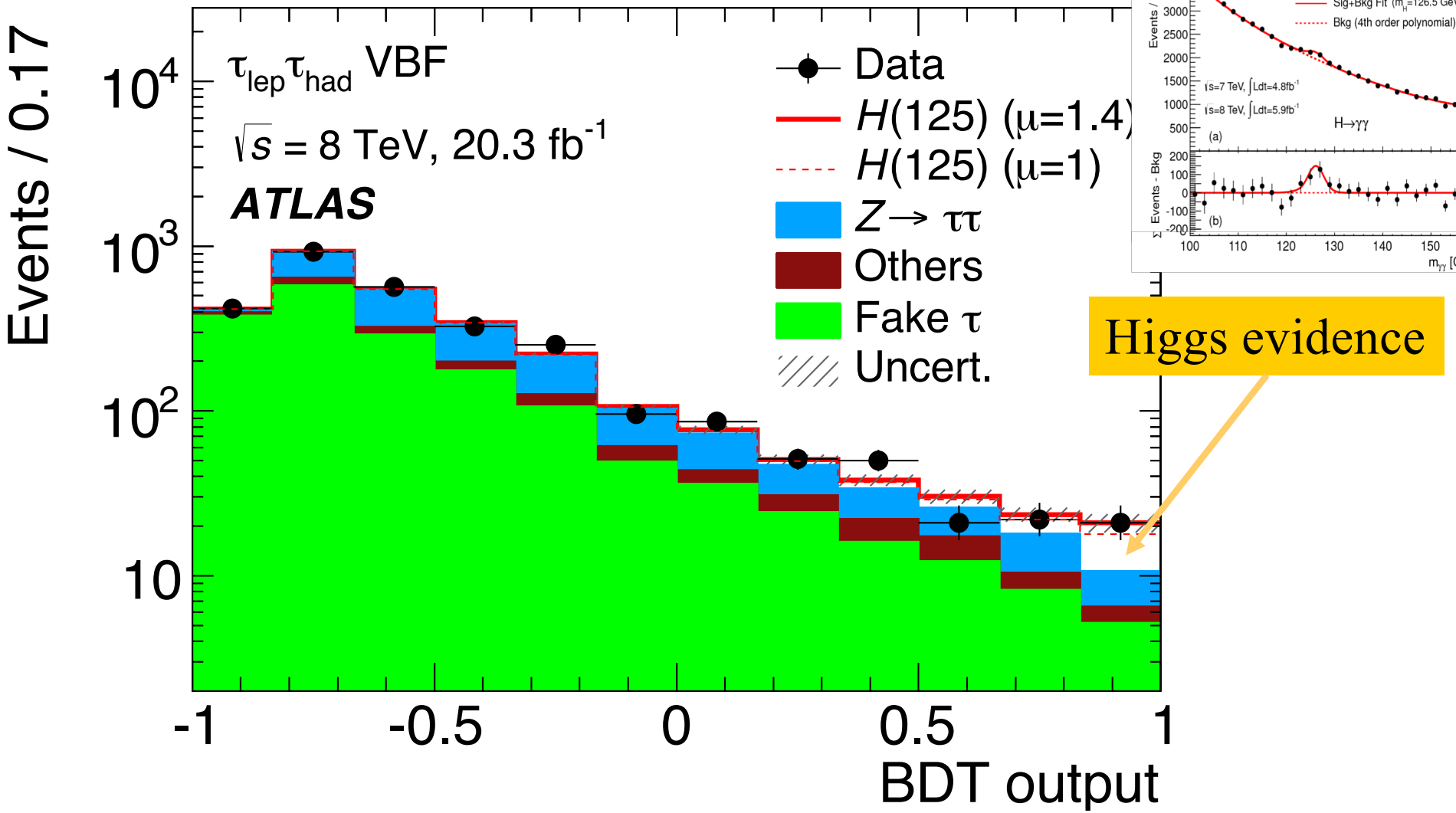
- Focus on applications rather than details of the techniques
- Deliberately incomplete (sorry…)
- No likelihood free inference, no classification without labels, no review on ML software, no application to distributed analysis, no GAN to uniformity, no Bayes optimisation, no reinforcement learning, no adversarial example, no probablilistic programming, no learning with quantum computing….

ML in Higgs Physics

Classifier

JHEP 04, 117 (2015) 1501.04943

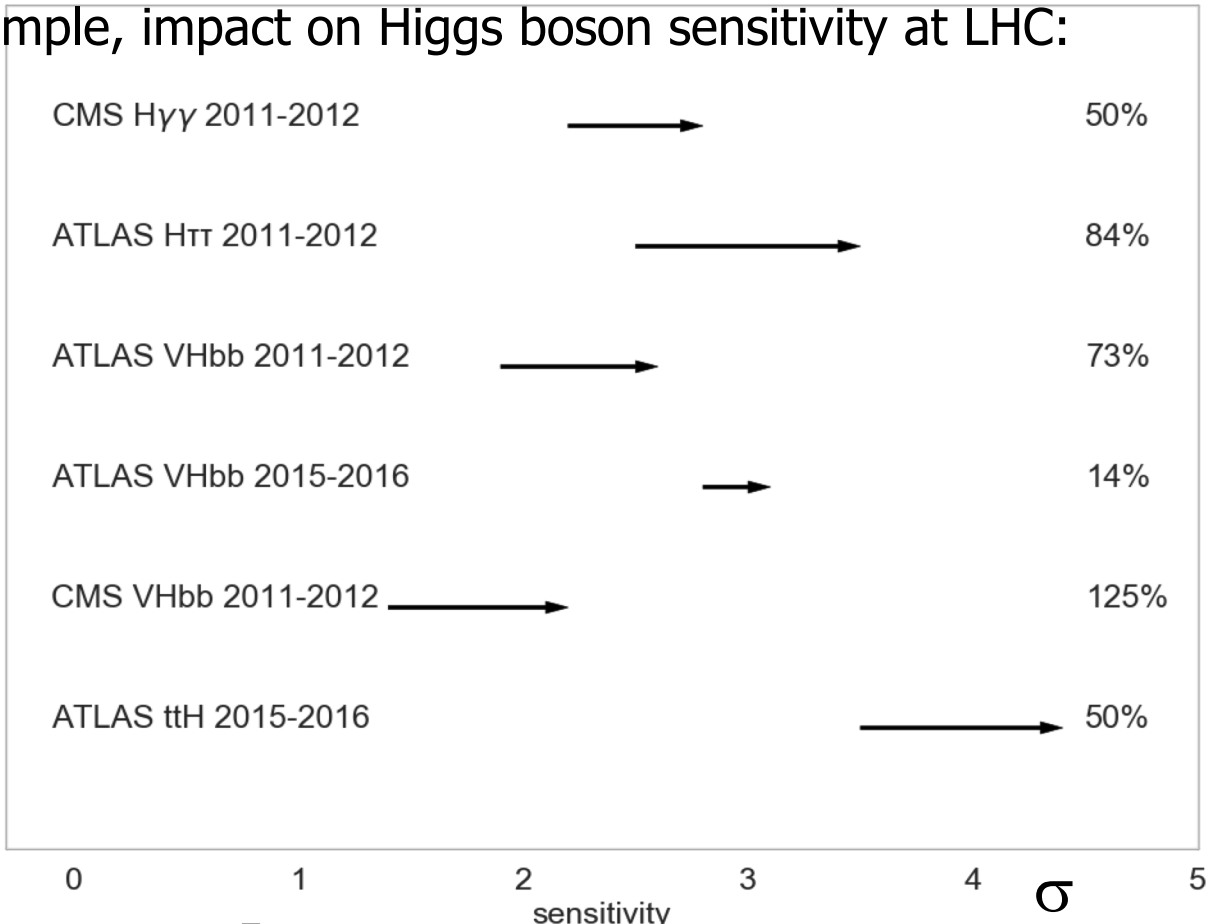
Boosted Decision Tree using ~a dozen of high level variables



Higgs evidence

ML on Higgs Physics

- At LHC, Machine Learning used almost since first data taking (2010) for reconstruction and analysis
- In most cases, Boosted Decision Tree with Root-TMVA, on ~10 variables
- For example, impact on Higgs boson sensitivity at LHC:

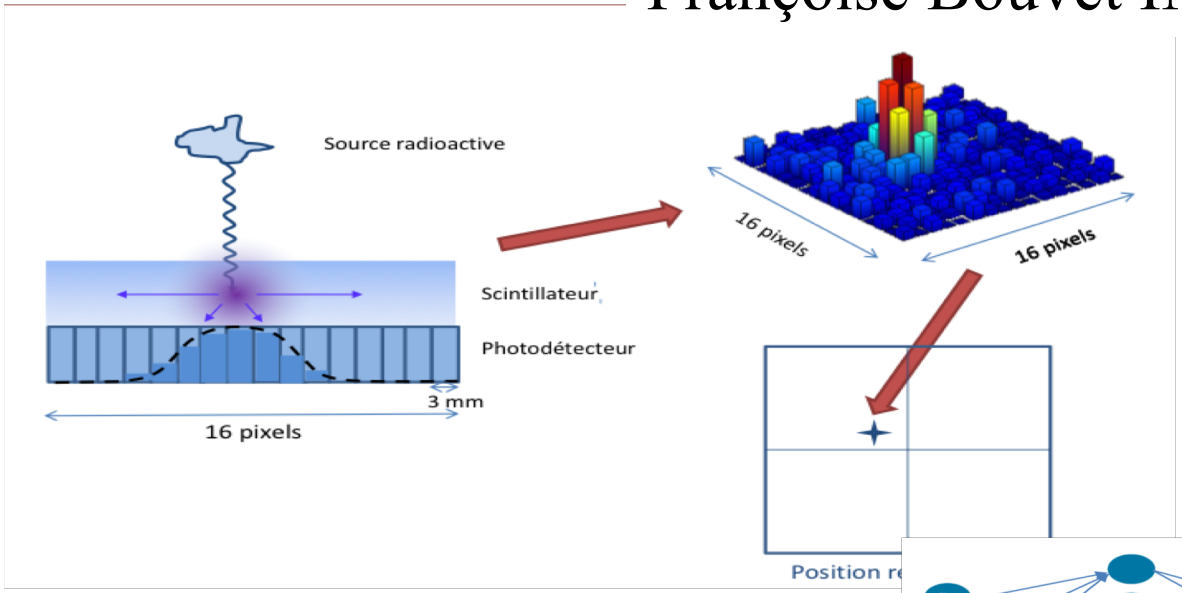


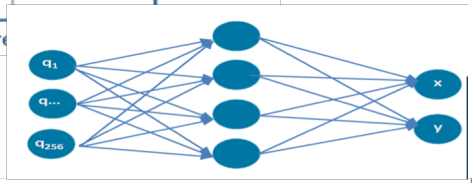
➔~50% gain on LHC running

ML in reconstruction

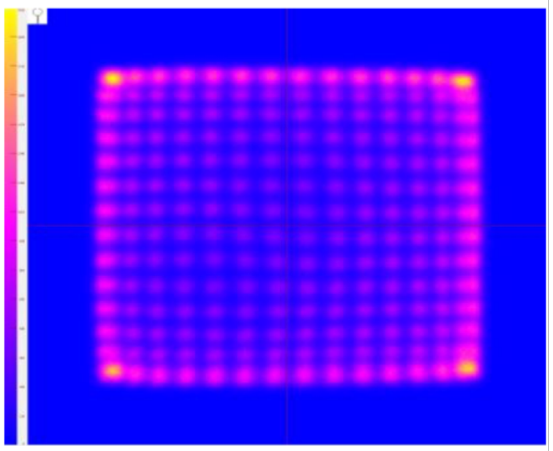
β,γ camera for medical application

Françoise Bouvet IMNC

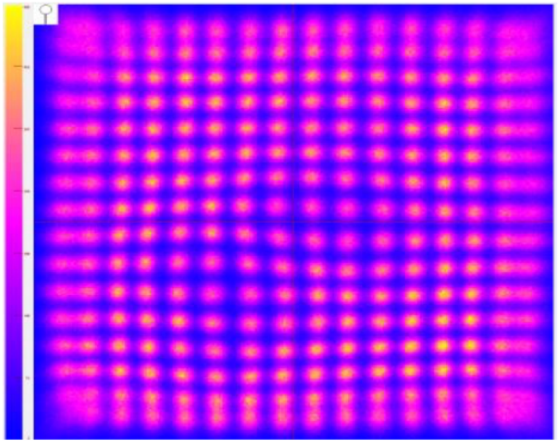




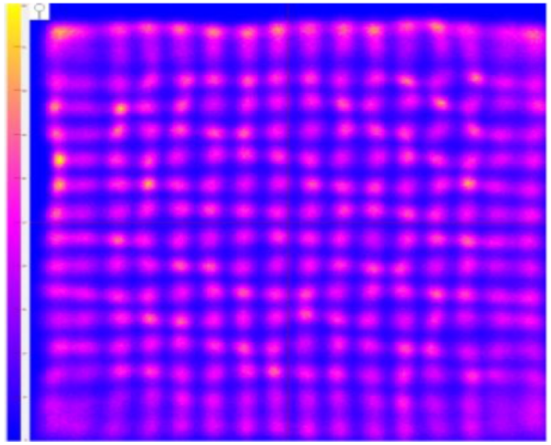
Barycentre Inaccurate & fast



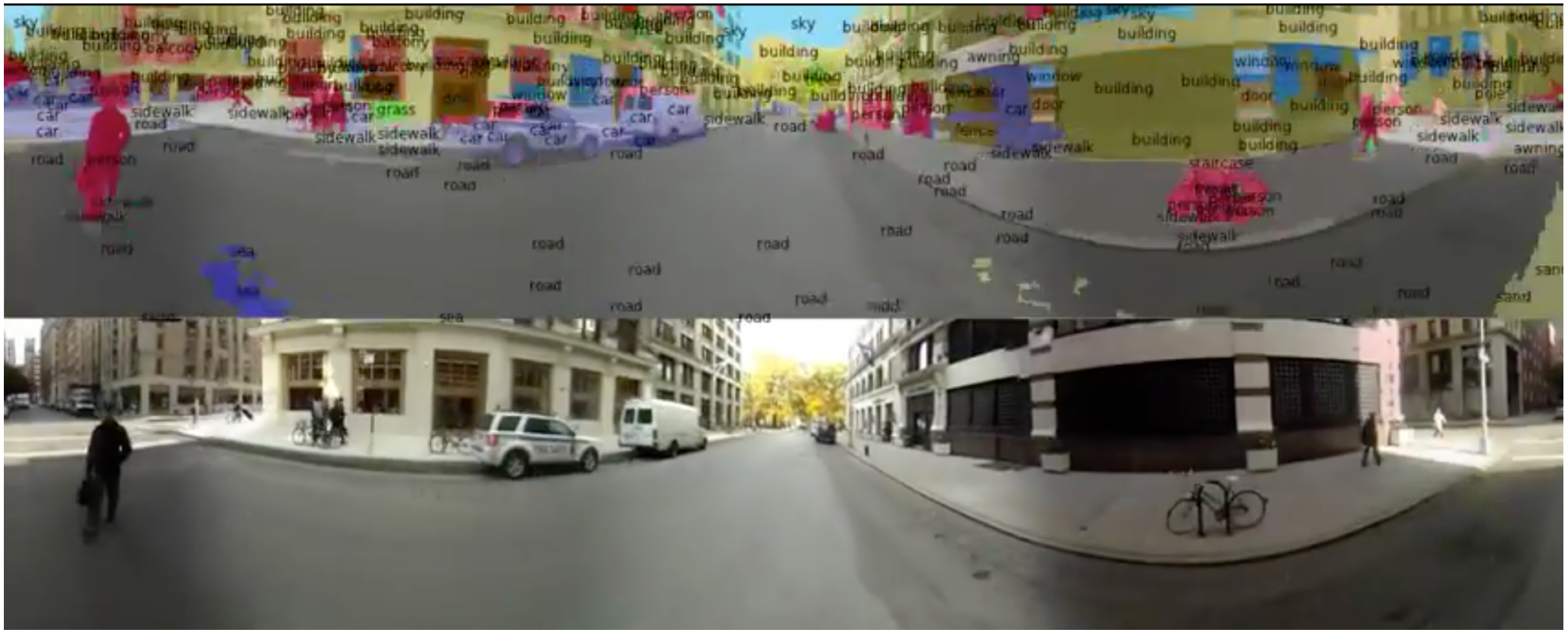
Levenberg Marquart accurate & slow



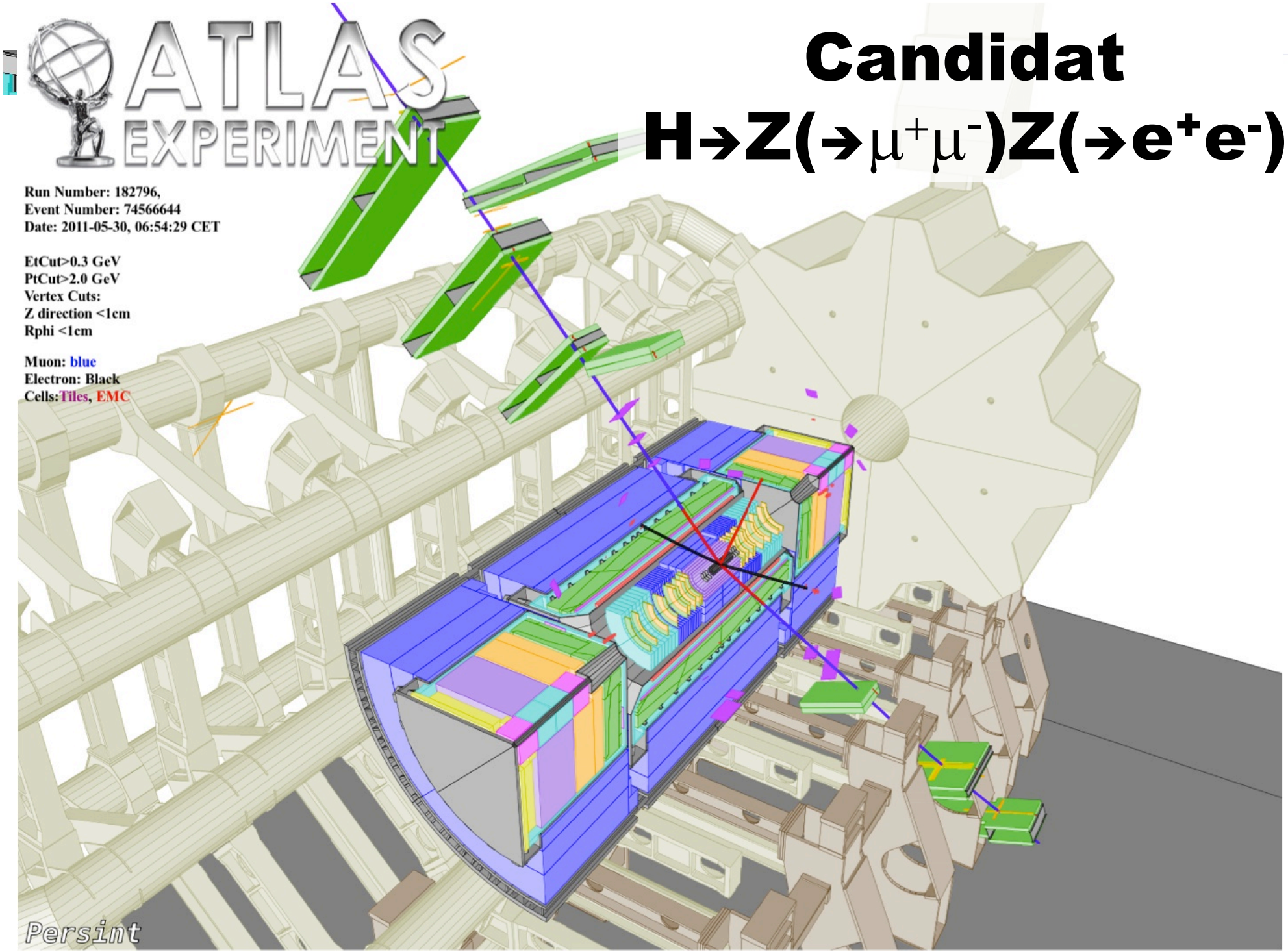
Keras – 256-40-2 accurate & fast



Typical Deep Learning application



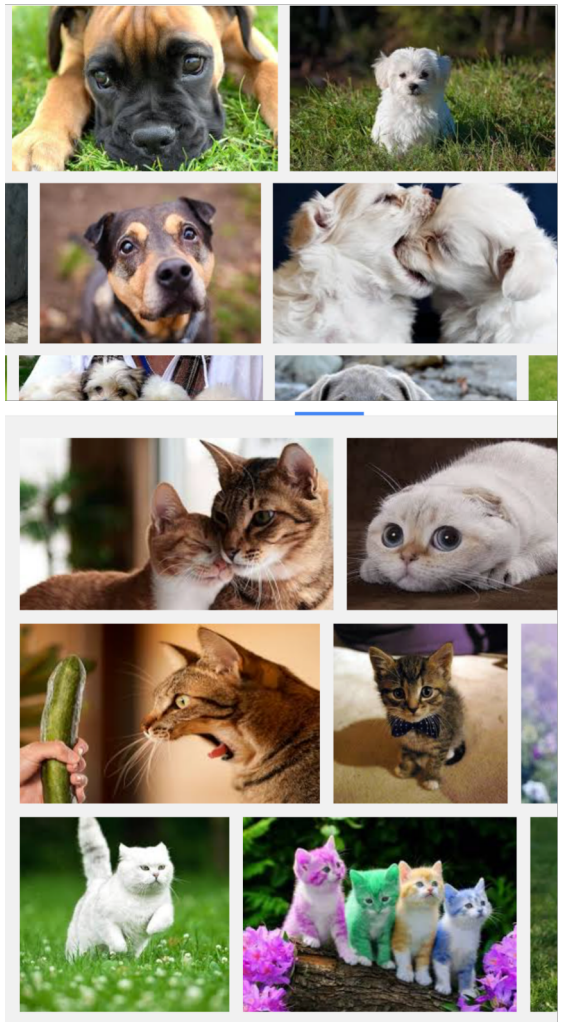
Candidat
$H \rightarrow Z(\rightarrow \mu^+\mu^-)Z(\rightarrow e^+e^-)$



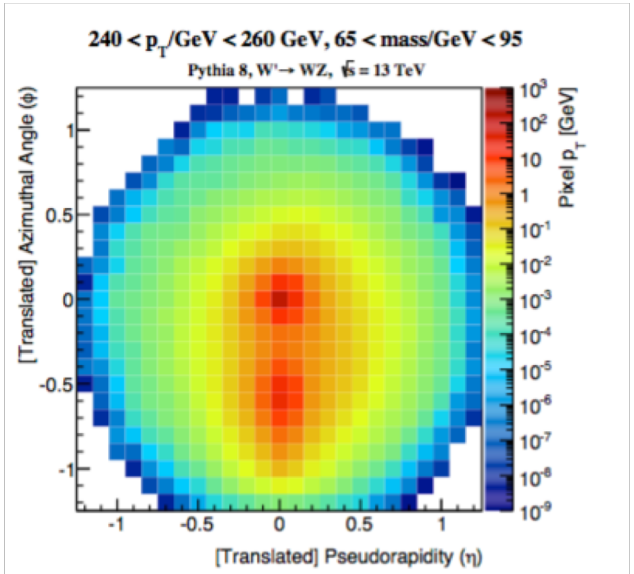
Jet Images

arXiv 1511.05190 de Oliveira, Kagan, Mackey, Nachman, Schwartzman

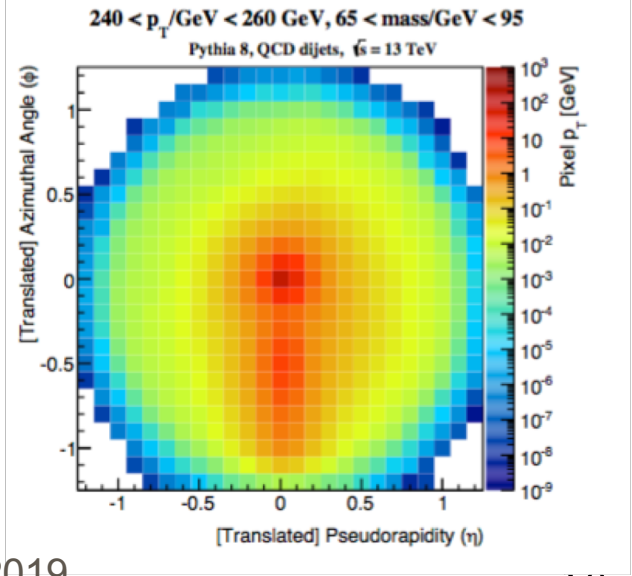
- Distinguish boosted W jets from QCD
- Particle level simulation
- Average images:



Boosted W➔qq jet

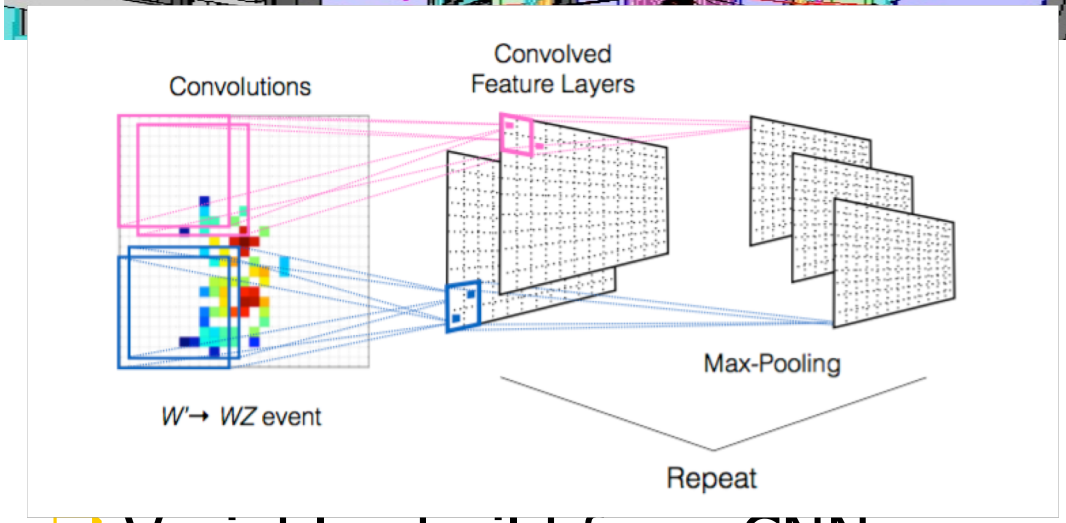


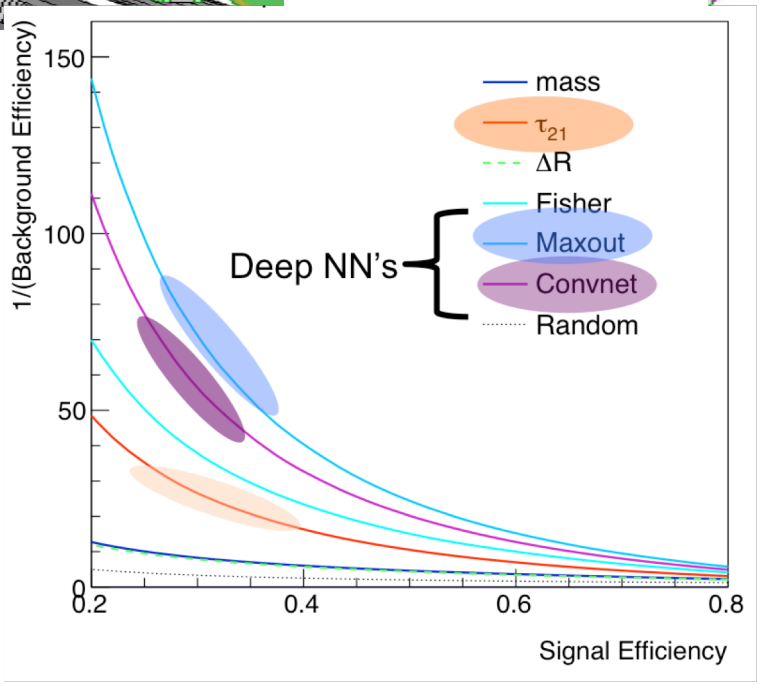
QCD



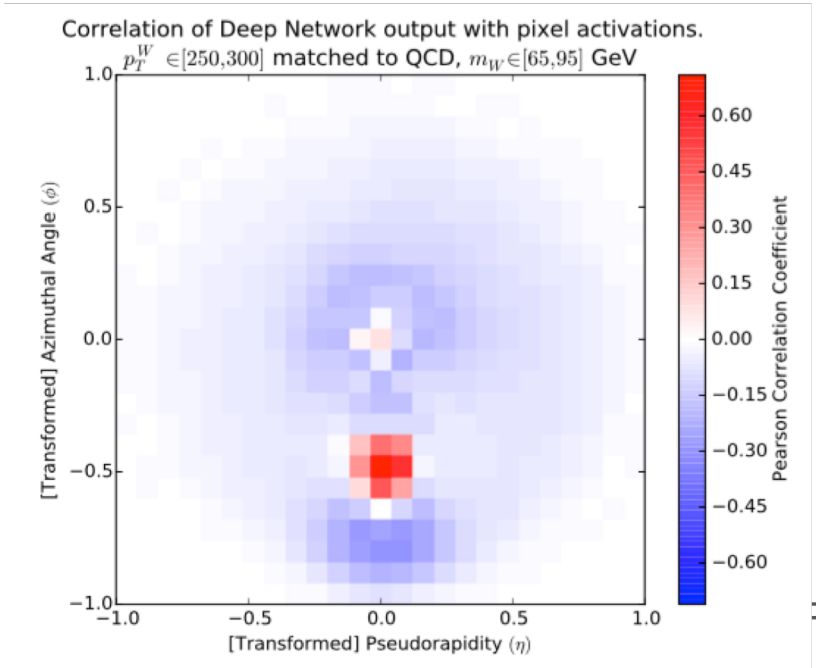
Jet Images : Convolution NN

arXiv:1511.05190

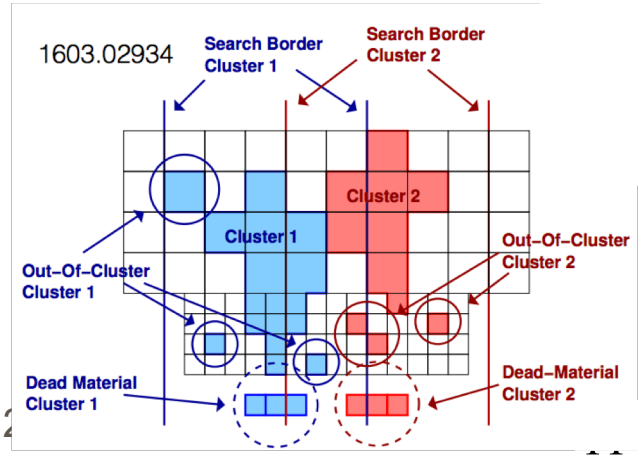




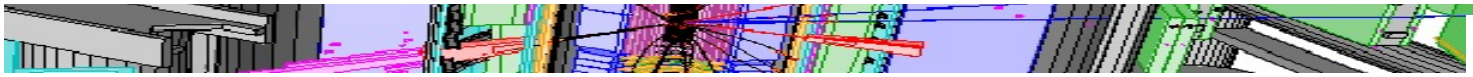
- Variables build from CNN outperform the more usual ones



- What the CNN sees (the "cat" neurone")
- Now need proper detector and pileup simulation ATL-PHYS-PUB-2017-017
- ➔3Dimension ?

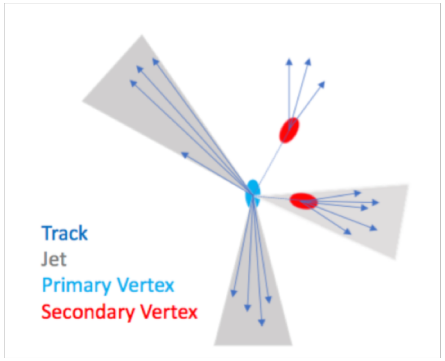


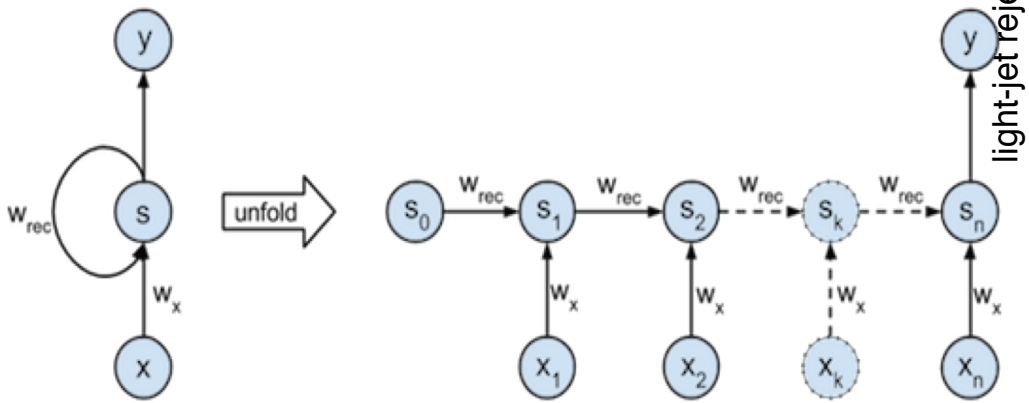
RNN for b tagging

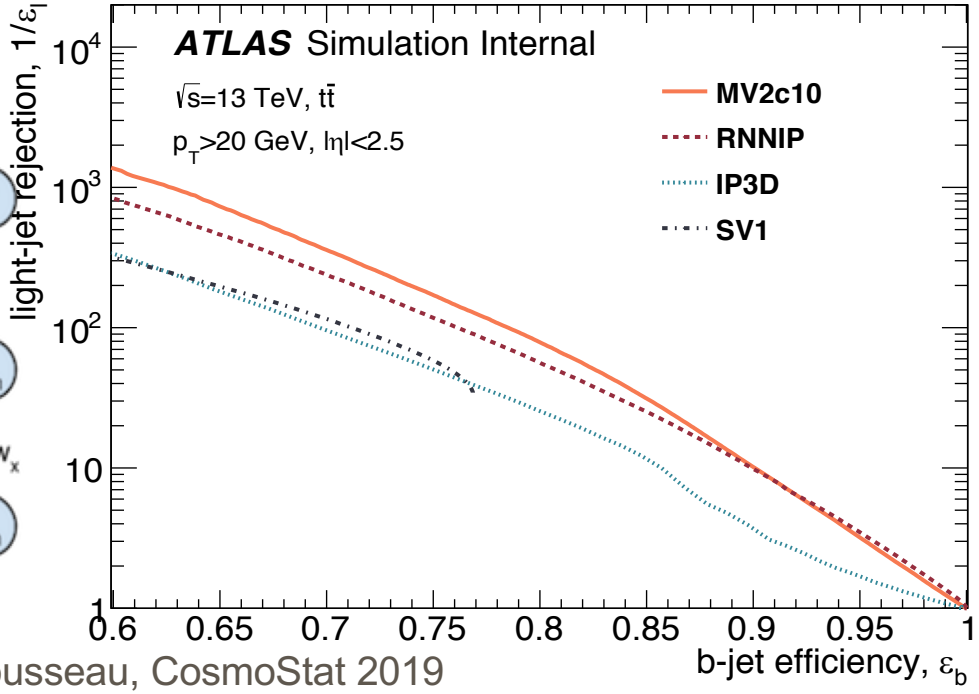


ATL-PHYS-PUB-2017-003

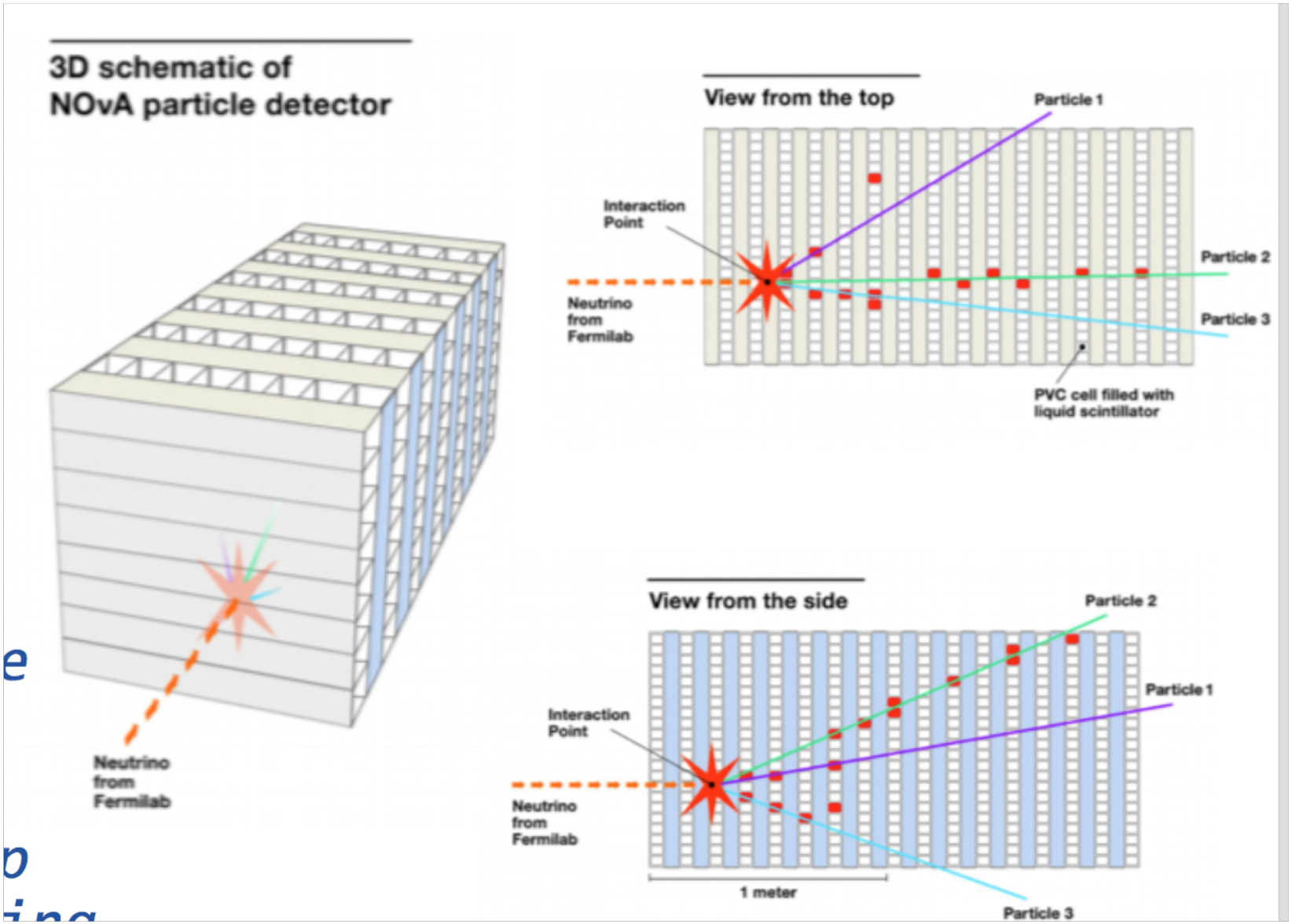
- BDT and usual NN expect a fix number of input. What to do when the number of inputs is not fixed like the tracks for b-quark jet tagging ?
- Recurrent Neural Networks (RNN) have seen outstanding performance for processing sequence data
  - Take data at several "time-steps", and use previous time-step information in processing next time-steps data
- For b-tagging, take list of tracks in jet and feed into RNN
  - Basic track information like d0, z0, pt-Fraction of jet, …
  - Physics inspired ordering by d0-significance
- RNN outperforms other IP algorithms
  - No explicit vertexing, still excellent performance
  - First combinations with other algorithms in progress
- Learning on sequence data may be important ir
  - Combining tracks with clusters? Track to vertex m





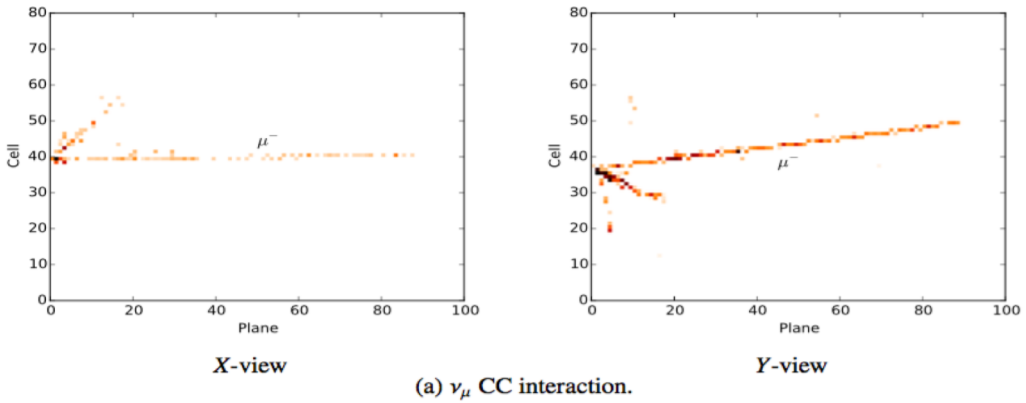


Deep Learning success : NOVA

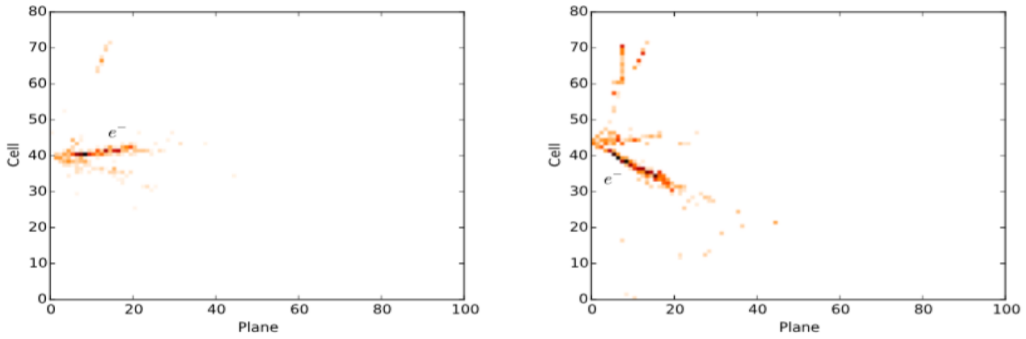


Nova (2)

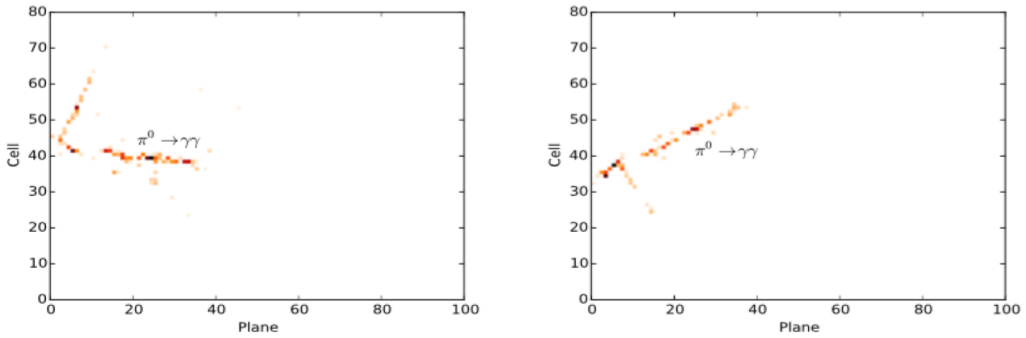
arXiv 1604.01444 Aurisano et al



(a) ν_μ CC interaction.



(b) ν_e CC interaction.



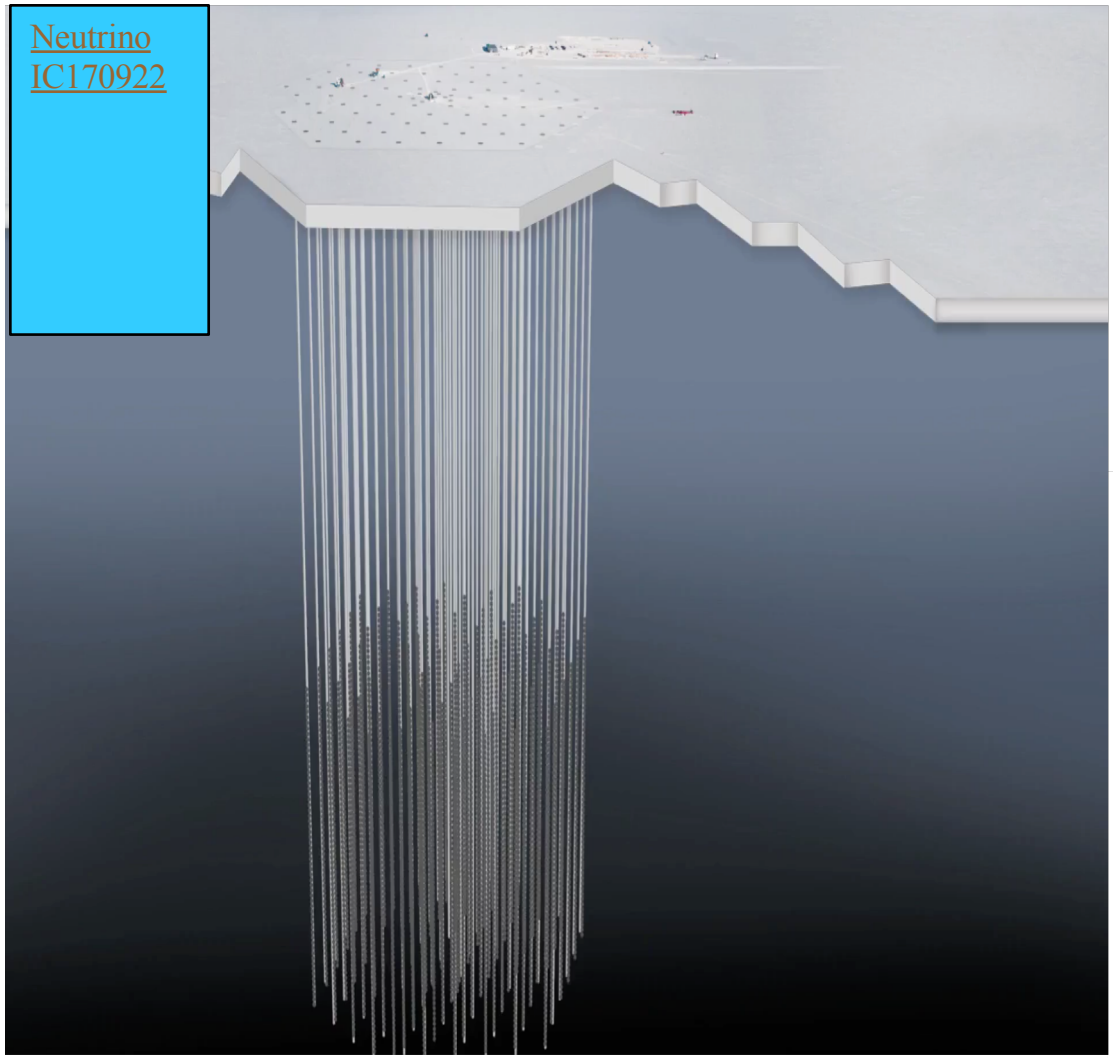
(c) NC interaction.

40% ε improvement

Neutrino interaction classification
Using Convolutionnal Neural Network (GoogleNet)
Actually used in physics results 1703.03328 and 1706.04592

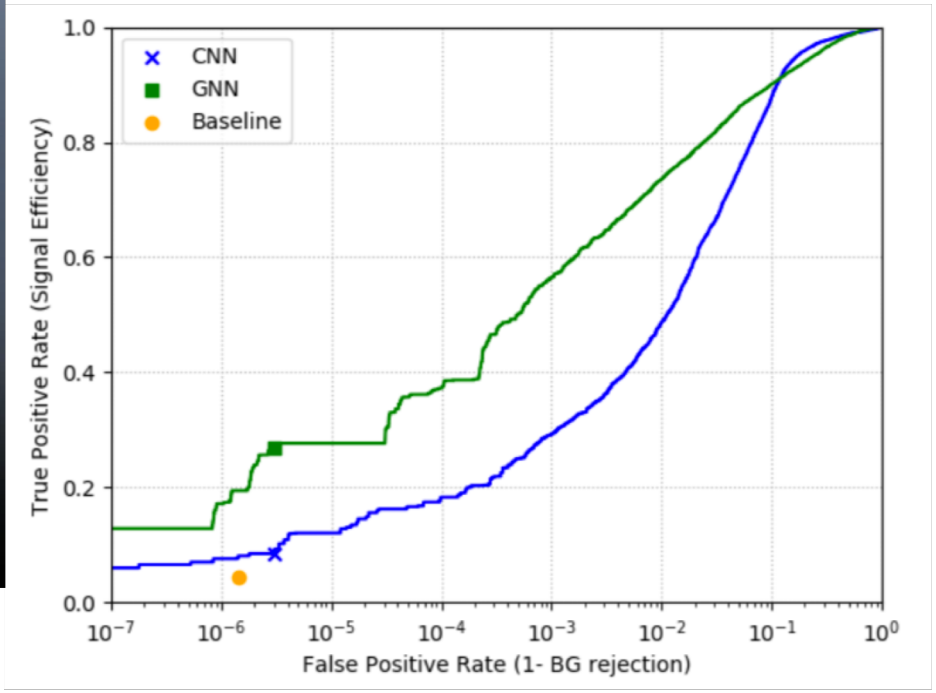
Ice Cube

1809.06166



Using Graph Convolution Network to separate neutrino from background

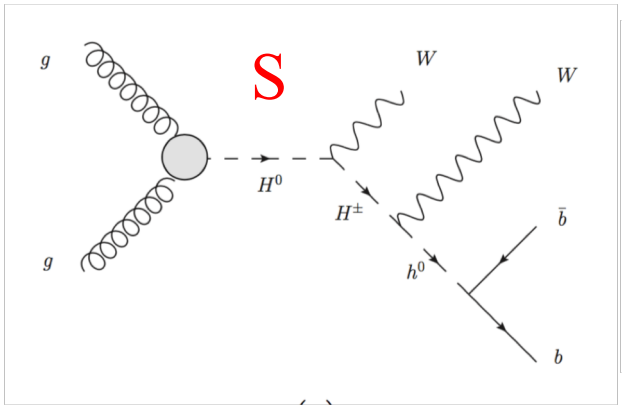
Promising but no physics result yet

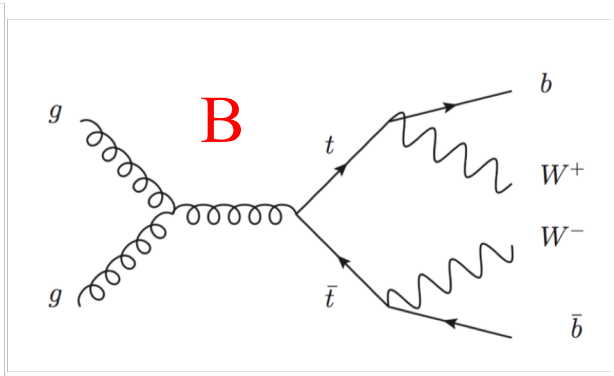


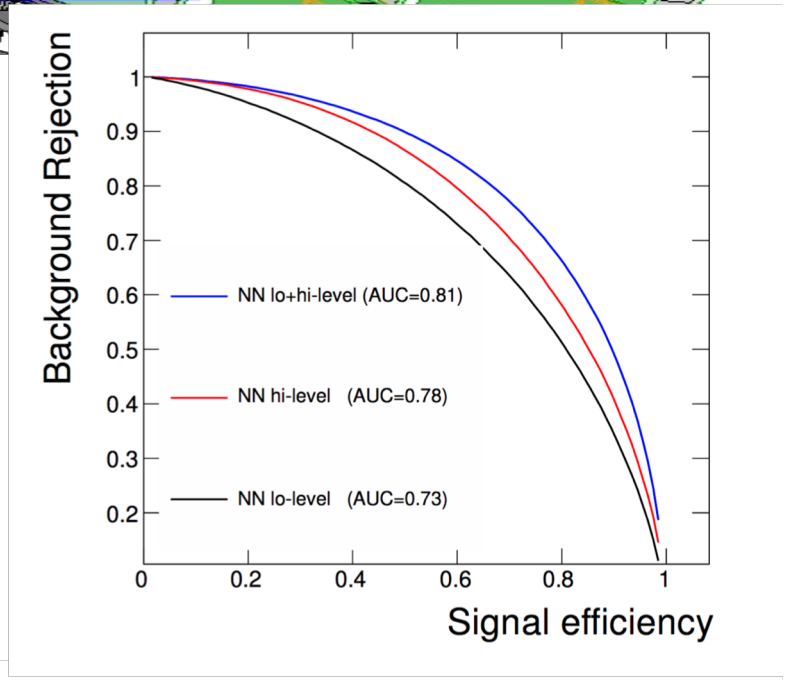
ML in Analysis

Deep learning for analysis

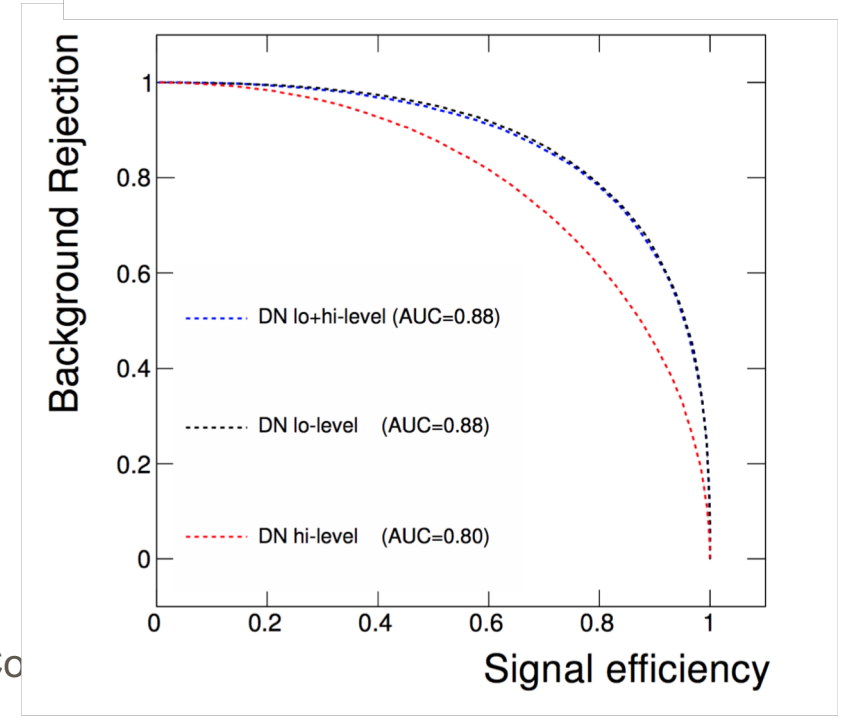
1402.4735 Baldi, Sadowski, Whiteson





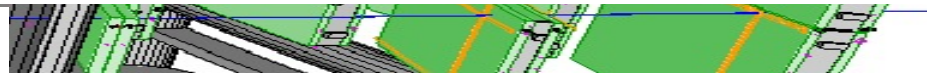


- MSSM at LHC : H^0➔WWbb vs tt➔WWbb
- Low level variables:
  - 3-momentum vectors
- High level variables:
  - Pair-wise invariant masses
- Deep NN outperforms NN, and does not need high level variables
- DNN learns the physics ???

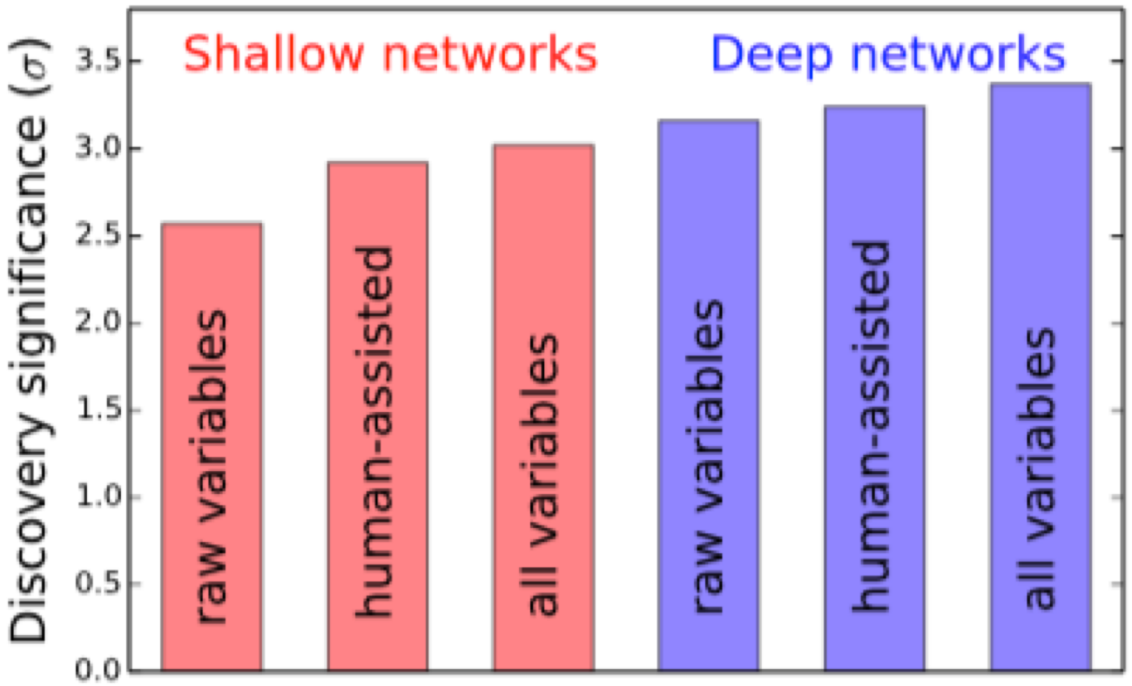


Deep learning for analysis (2)

1410.3469 Baldi Sadowski Whiteson



- H tautau analysis at LHC: H➔tautau vs Z➔tautau
  - Low level variables (4-momenta)
  - High level variables (transverse mass, delta R, centrality, jet variables, etc…)



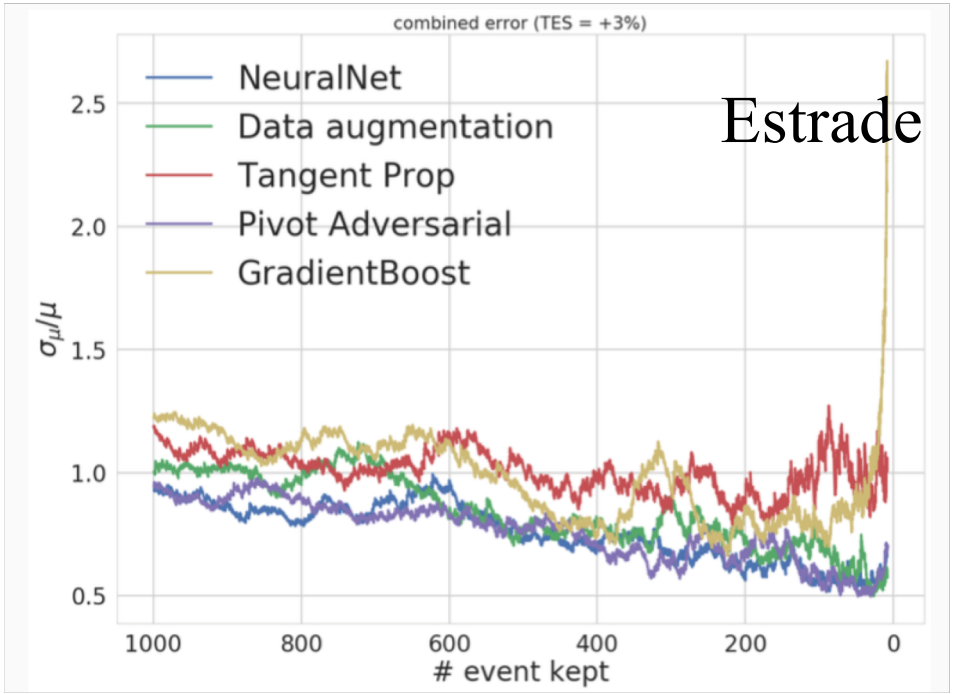
- Here, the DNN improved on NN but still needed high level features
- Both analyses with Delphes fast simulation
- ~100M events used for training (>>100* full G4 simulation in ATLAS)

DNN for analysis (3)

- No published LHC analyses using DL (CMS 2018 ttH « DNN » just two layers)
- Recent trend is to feed more (up to 20) variables to classifiers, even low level ones (3-vectors of particles) (see recent ATLAS/CMS ttH papers)
- A few NN in top and Higgs physics but no clear advantage wrt BDT
- Not completely clear why: most likely hypothesis : lack of training MC (Baldi et al papers use >>10^6 events, while a typical LHC analysis has at most 100K, even less, after all preselection)
- ➔DNN, not a drop-in replacement/improvement on BDT

SystML : syst aware training

- Pitch : typical ML classifier (BDT, NN) training is minimising the *statistical* uncertainty. However *systematic* uncertainty is an import aspect of an analysis (!)
  - =>how can an ML classifier take into account a model of systematics at training time, to optimise the *total* uncertainty ?
- Several studies done using HiggsML H tautau public sample
- No clear recommendation yet

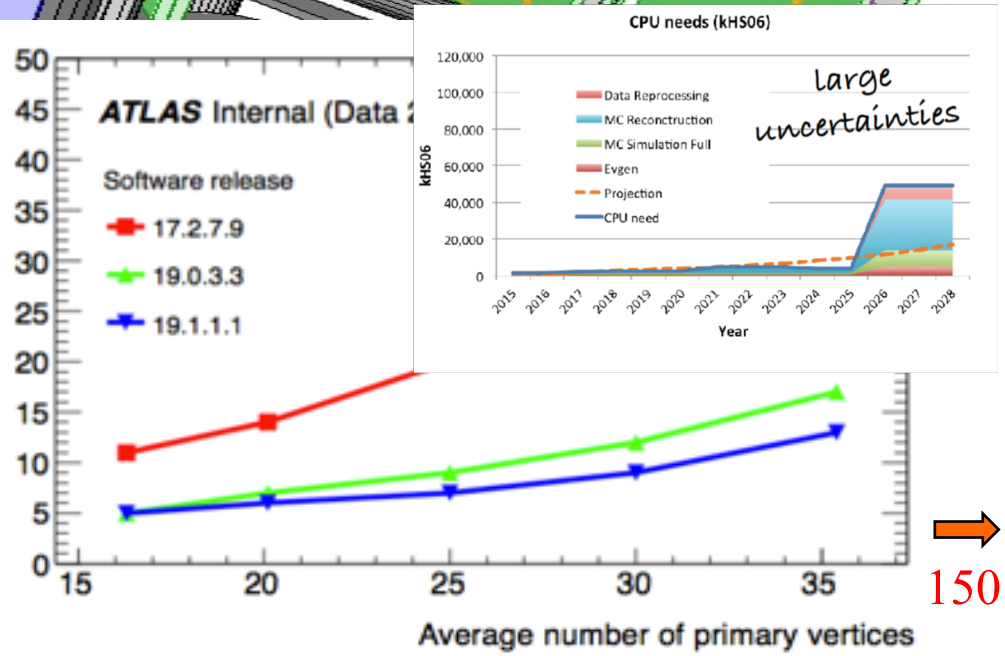


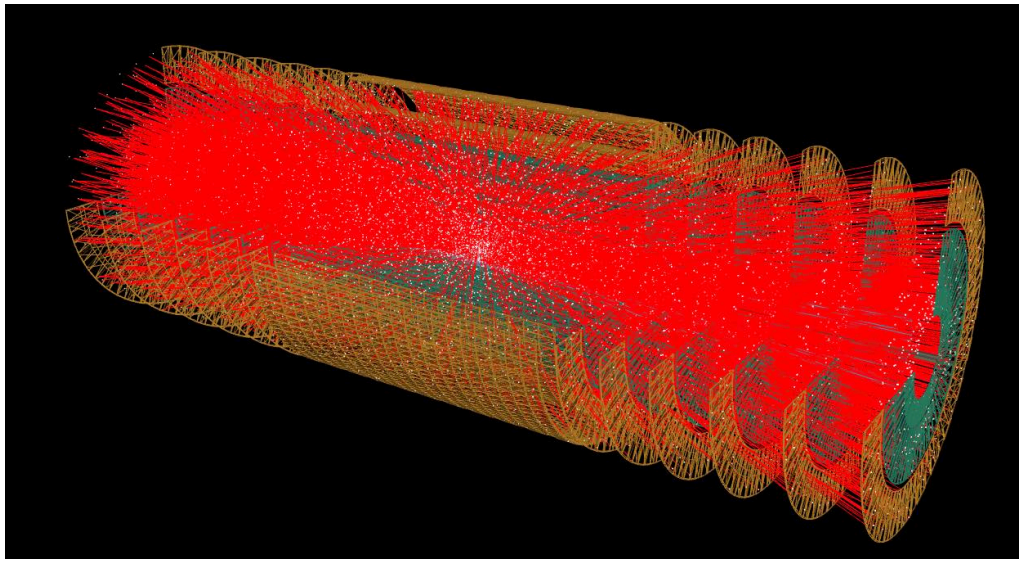
- Estrade Germain Guyon Rousseau Systematics aware learning: a case study in High Energy Physics In ESANN 2018 - 26th European Symposium on Artificial Neural Networks (2018) 1611.01046
- Louppe Kagan Cranmer, Learning to Pivot with Adversarial Networks, in Advances in Neural Information Processing Systems 30, 981-990 (2017)
- Elwood and Krucker Direct optimisation of the discovery significance when training neural networks to search for new physics in particle colliders 1806.00322
- Pablo de Castro, Tommaso Dorigo INFERNO: Inference-Aware Neural Optimisation 1806.04743
- Li-Gang Xi, QBDT, a new boosting decision tree method with systematic uncertainties into training for High Energy Physics 1810.08387

TrackML tracking challenge

Tracking competition

- Tracking (in particular pattern recognition) dominates reconstruction CPU time at LHC
- HL-LHC (phase 2) perspective : increased pileup :Run 1 (2012): <>~20, Run 2 (2015): <>~30,Phase 2 (2025): <>~150
- CPU time quadratic/exponential extrapolation (difficult to quote any number)
- Large effort within HEP to optimise software and tackle micro and macro parallelism. Sufficient gains for Run 2 but still a long way for HL-LHC.
- >20 years of LHC tracking development. Everything has been tried?
  - Maybe yes, but maybe algorithm slower at low lumi but with a better scaling have been dismissed ?
  - Maybe no, brand new ideas from ML (i.e. Convolutional NN)
- ➔Tracking challenge launched May-Aug 2018 on Kaggle : just accuracy
- ➔Throughput phase launched on Codalab : Sep-Mar 2019 : accuracy AND speed
- 125 events x (10’000 tracks / 100’000 points)
- Follow us on twitter @trackmllhc !
- Details on : https://sites.google.com/site/trackmlparticle/

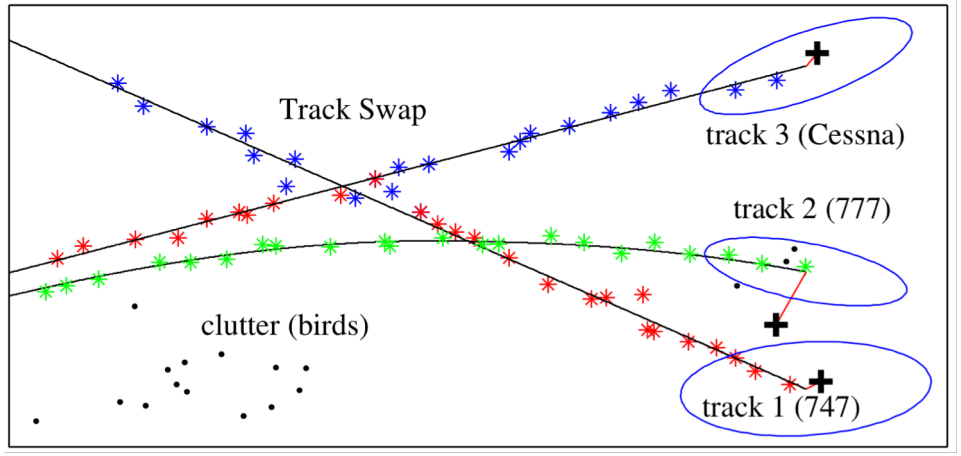


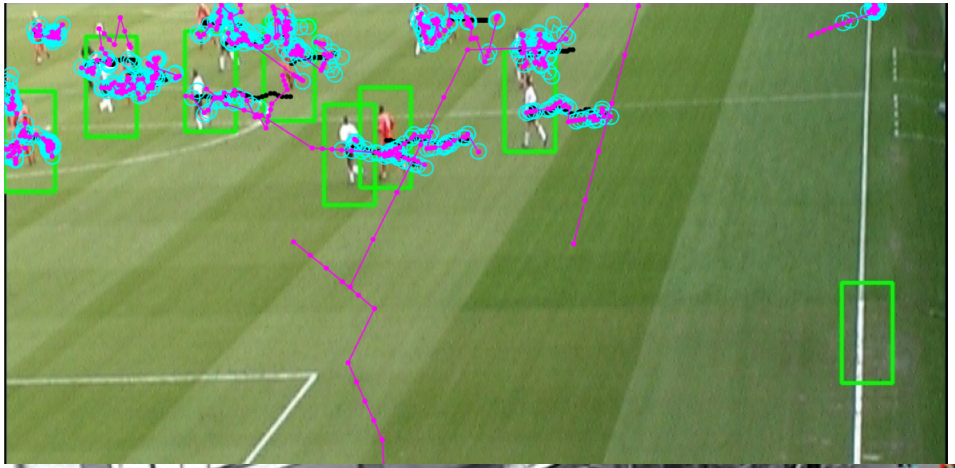


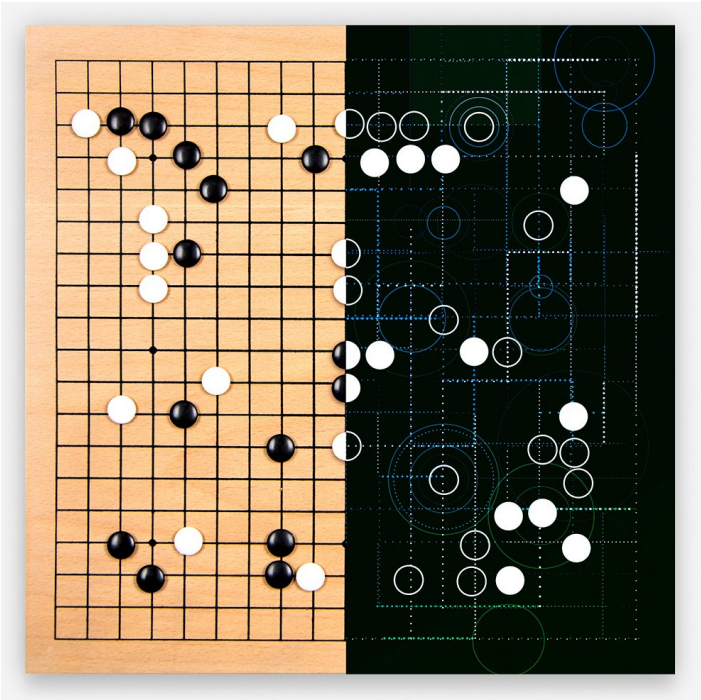

TrackML
sponsors
kaggle
NVIDIA
CERN
UNIVERSITÉ
DE GENÈVE
Paris-Saclay
Center for
Data Science
INSTITUTE
DATAIA
Data Science, Intelligence & Society
CHA
LEARN
IRIS-HEP
CERN
openlab
Inria
PyTorch
cnrs
erc
Fermilab
COMMON
GROUND
inqnet
UNIVERSITE
PARIS
SUD
université
PARIS-SACLAY

Pattern Recognition/Tracking

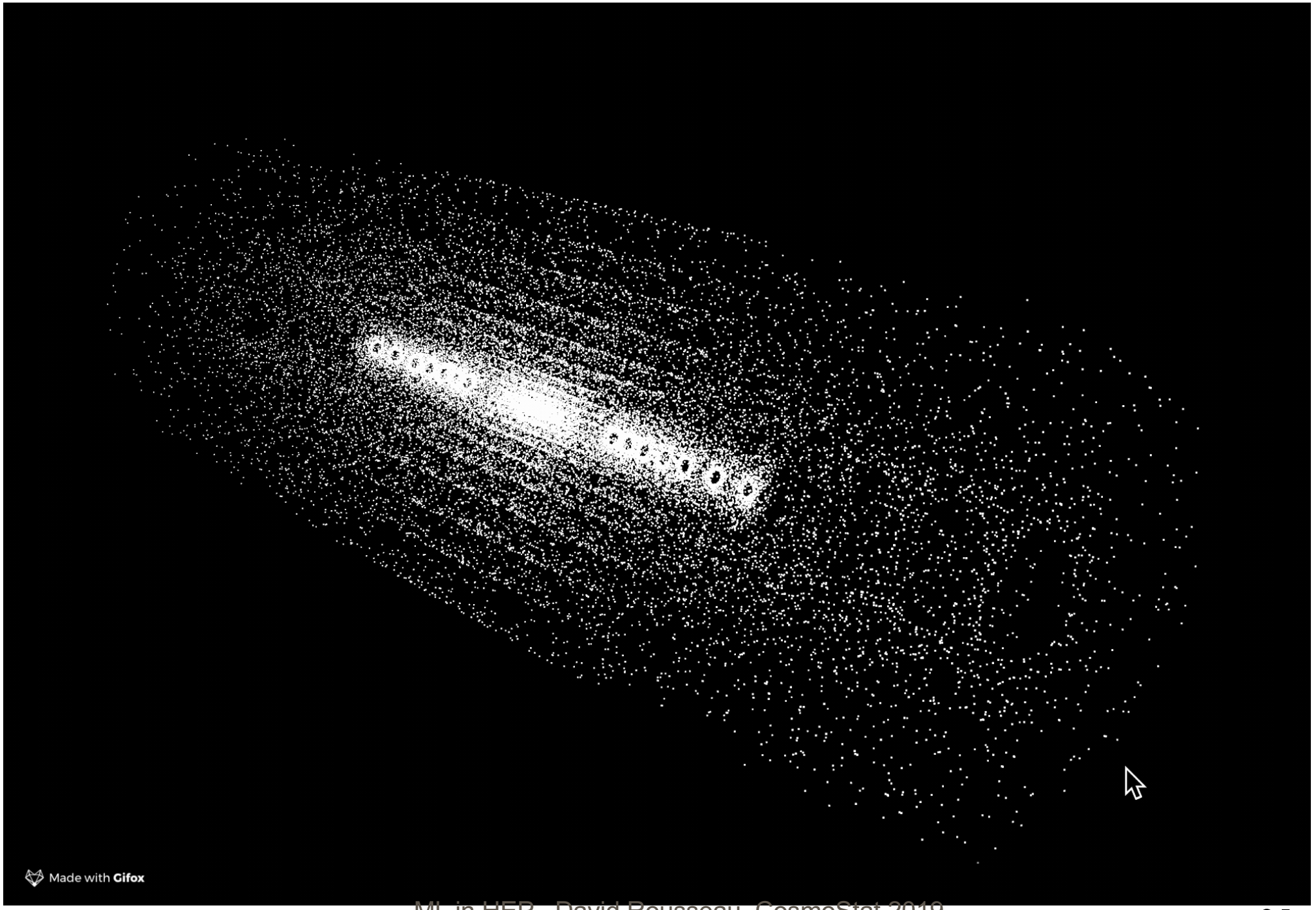
- Pattern recognition/tracking is a very old, very hot topic in Artificial Intelligence, but very varied
- Note that these are real-time applications, with CPU constraints

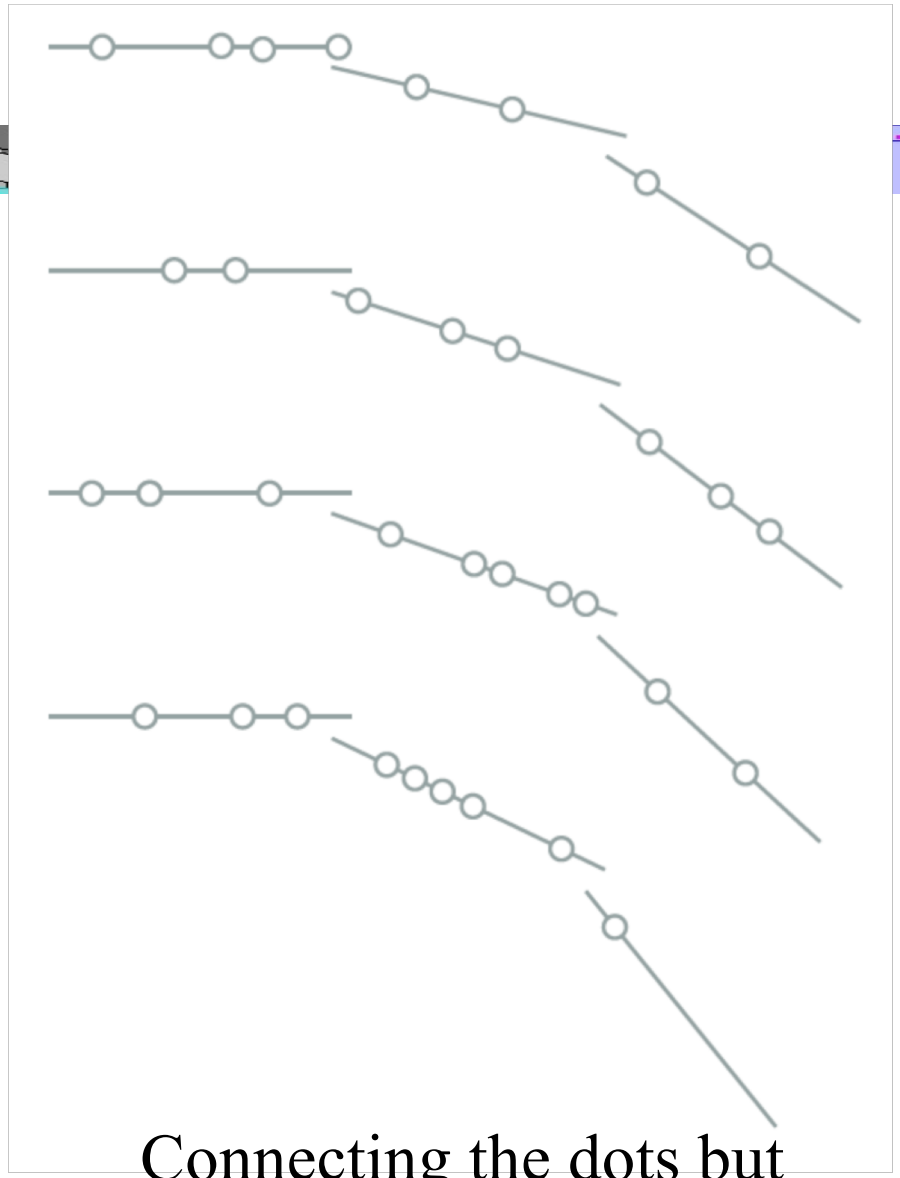


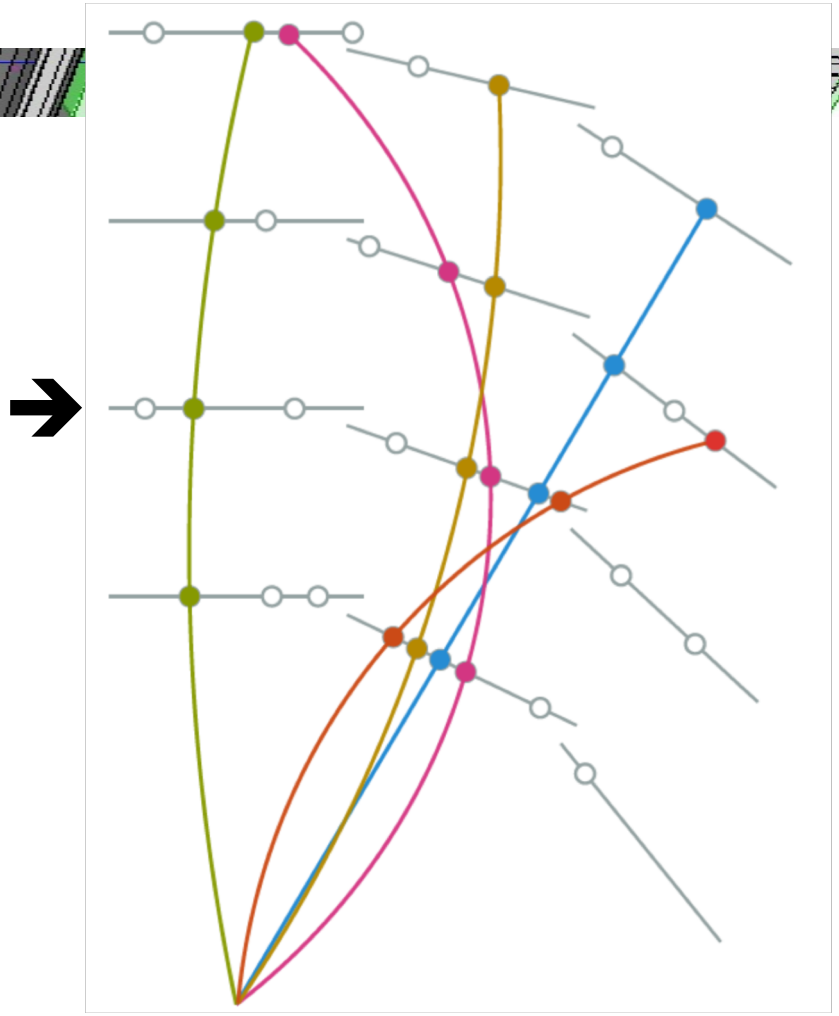




in HEP , David Rousseau, CosmoStat 2019


Made with Gifox





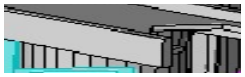
Connecting the dots but

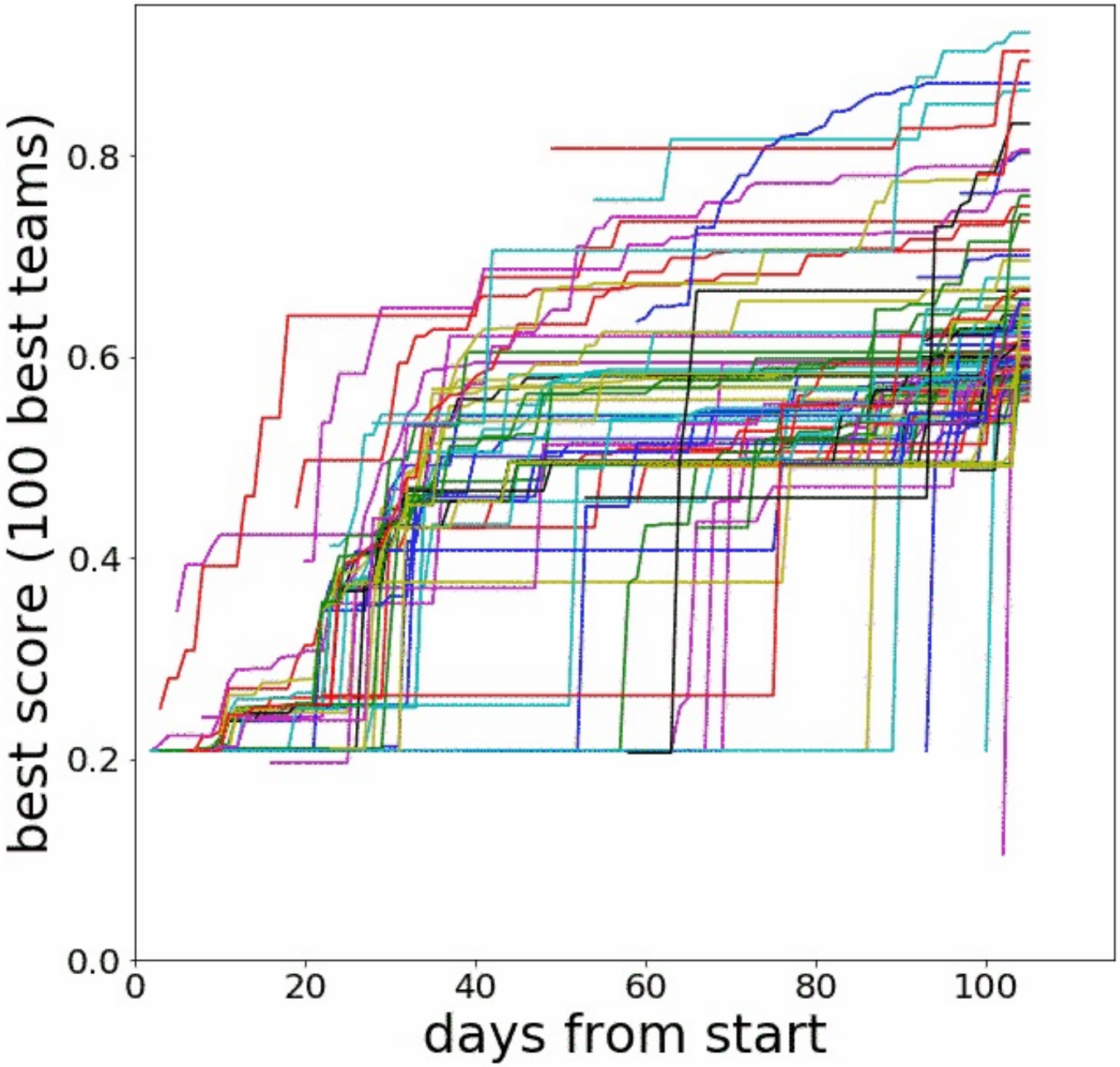
- 3 dimensions
- 10’000 tracks x 10 points

Why is it difficult?

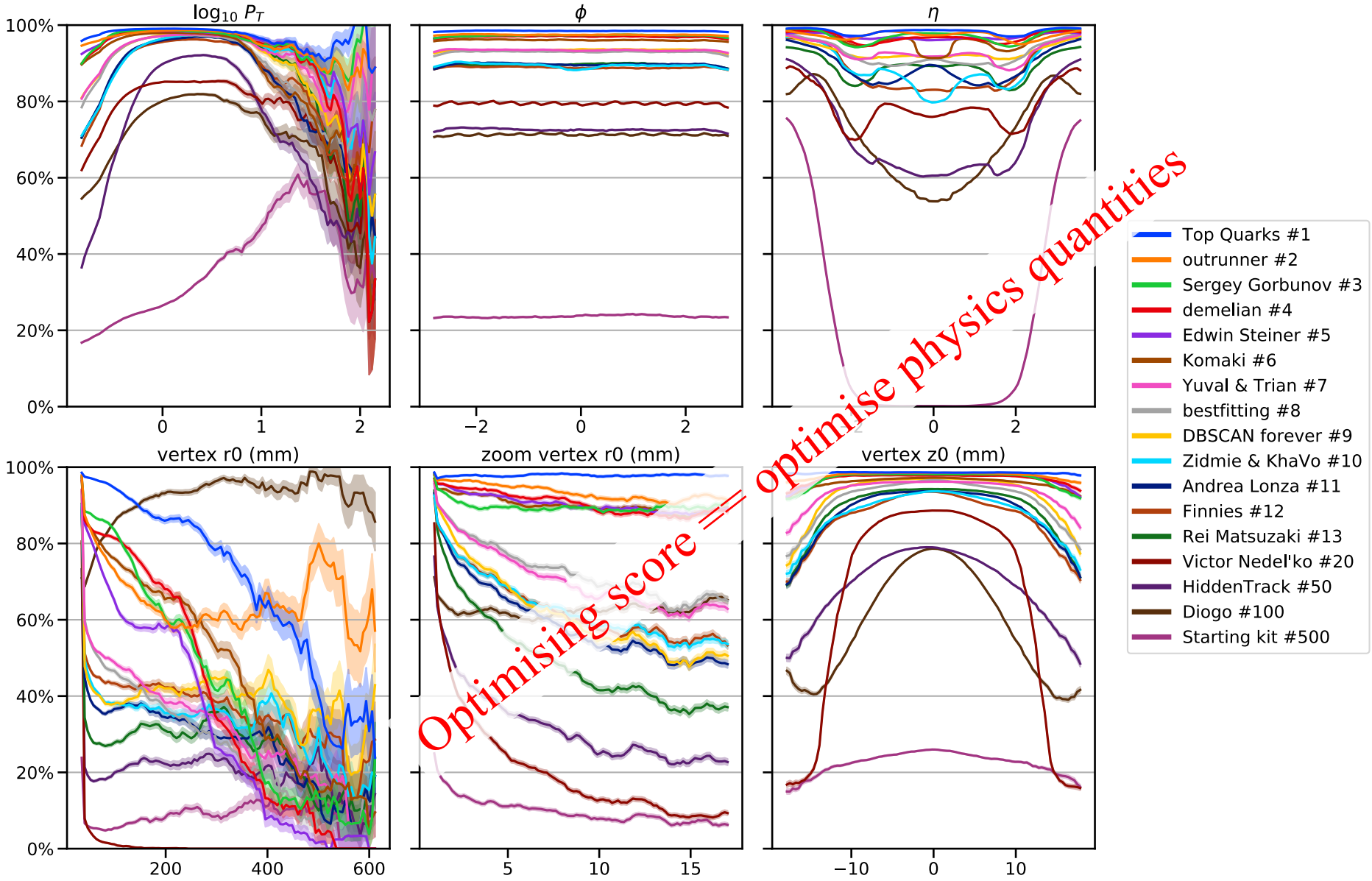
- 100’000 to group into 10’000 tracks of 10 points
  - ➔~$10^{450'000}$ combinations
  - ⇒brute force has (really) no chance
- Precision of the points : ~50μm on a volume ~40 m^3
  - ➔3 10^{14} voxels!
  - 2D projection ➔2 10^{9} pixels !
  - ⇒ image recognition algorithm have (really) no chance
- Not a classical problem

Evolution of leaderboard





Efficiency all



A few competitors

- icecube #1 92.2 % (norvegian CS master student) : combinatorial approach, with a bit of ML
- outrunner #2 90.3% (taïwanese software engineer) Deep Learning approach
  - Very innovative!
  - But brute force : takes one full day per event !
  - However code is using naïve python nested loops
- Sergey Gorbunov #3 89.4% demelian #4 87.1% : (HEP tracking trigger experts in HEP labs) parameterised local helix fitting
- Yuval & Trian #7 80.4% : (greek and israeli computing engineer) innovative clustering
- CPMP #9 80.1% : (french computing engineer) DBSCAN unsupervised clustering algorithm
  - we gave DBSCAN in starting kit, with a 20% score, because in only required a few lines
- Nicole and Liam Finnies #12 74.8% : (german data scientists) use LSTM
- We are currently writing a chapter for NeurIPS2018 Competition Workshop proceedings

Codalab Schematic

https://competitions.codalab.org/competitions/20112

CodaLab

	hit_id	x	y	z	volume_id	layer_id	module_id
0	1	-64.409897	-7.163700	-1502.5	7	2	1
1	2	-55.336102	0.635342	-1502.5	7	2	1
2	3	-83.830498	-1.143010	-1502.5	7	2	1
3	4	-96.109100	-8.241030	-1502.5	7	2	1

event(s) are loaded in memory

start

API to call python/C++ API

User executable

stop

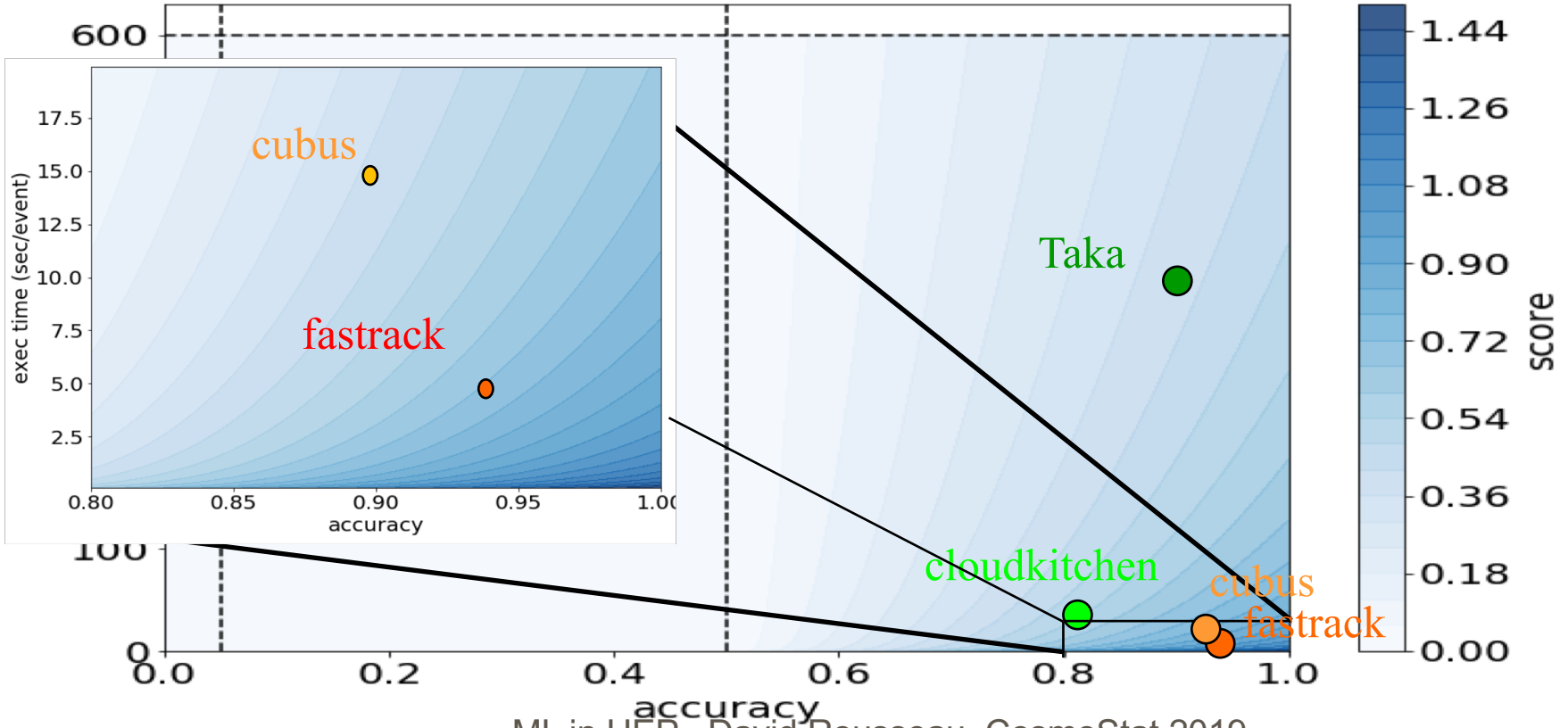
solution

Measure both accuracy score and evaluation time

VM 2 cores, 4 Gb memory

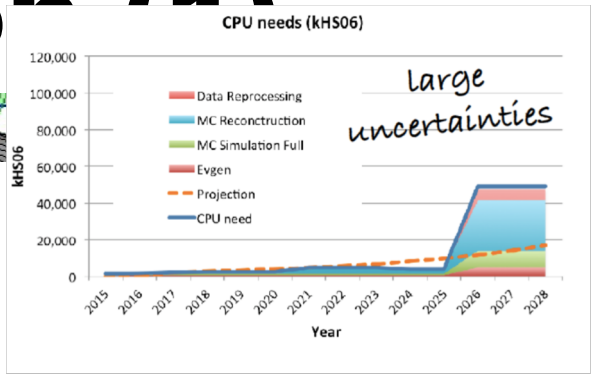
Throughput on-going results

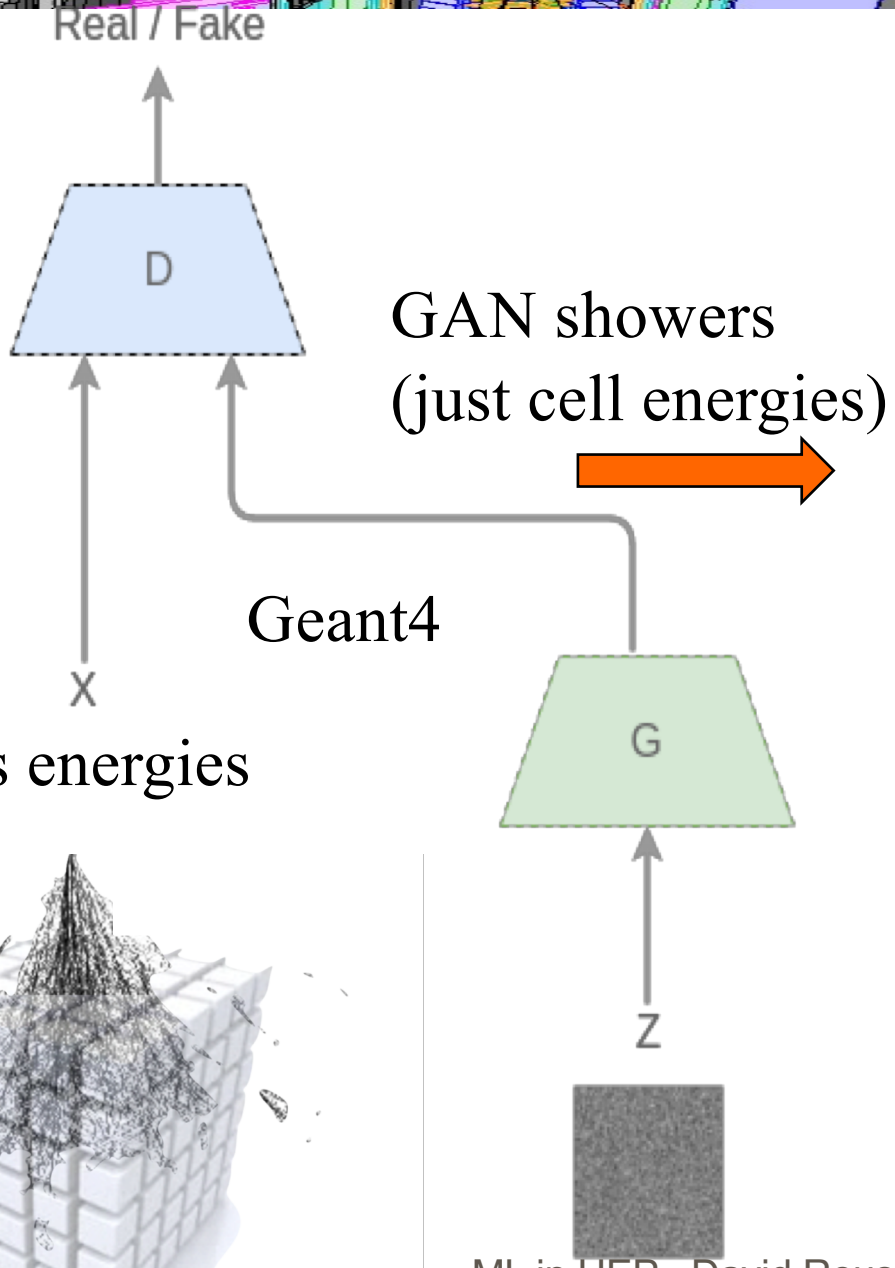
- *Ranking score* :
  - 0 if time >600 s or accuracy <50%
  - $\sqrt{log}(1 + 600/time) * (accuracy - 0.5)^2$
- Documented software of first phase #1 #2 #3 #7 #9 #11 #12 released
  - Can be used as starting point but need retuning
- ➔a couple of very fast participants at high accuracy



ML in simulation

GAN for simulation (1)

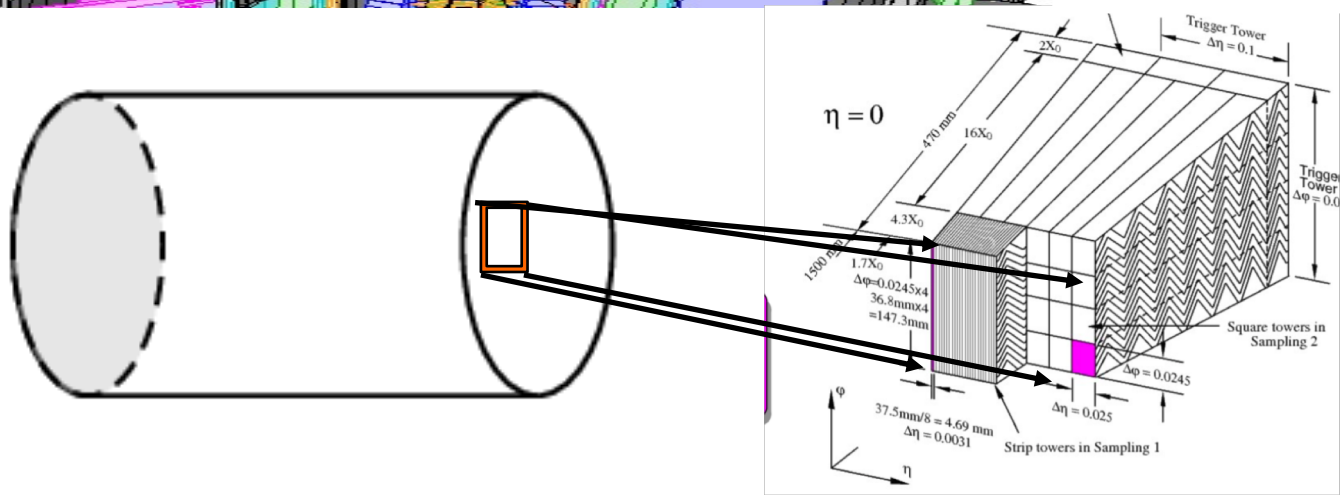




- Half of LHC grid computers (~500.000 cores) are crunching Geant4 simulation 24/24 365/365
- …while LHC experiments are collecting more and more events
- ➔reducing CPU consumption of simulation is very important
- Imagine training a GAN on single particle showers of all types and energies
- Then when an event is simulated it would ask for GAN showers on request (superfast by 3-4 order of magnitude)
- Would replace current fast simulation, frozen shower libraries….
- If/when it works, would require large GPU clusters

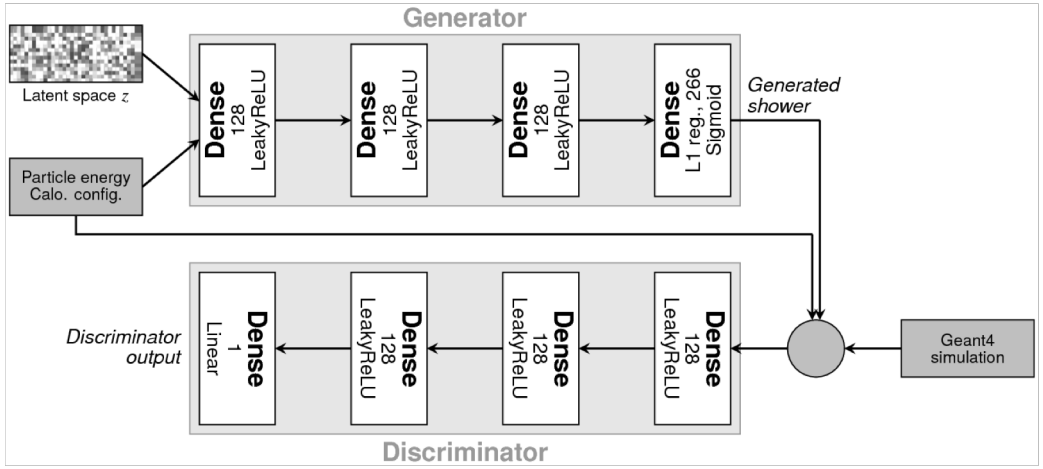
ATLAS calo simulation

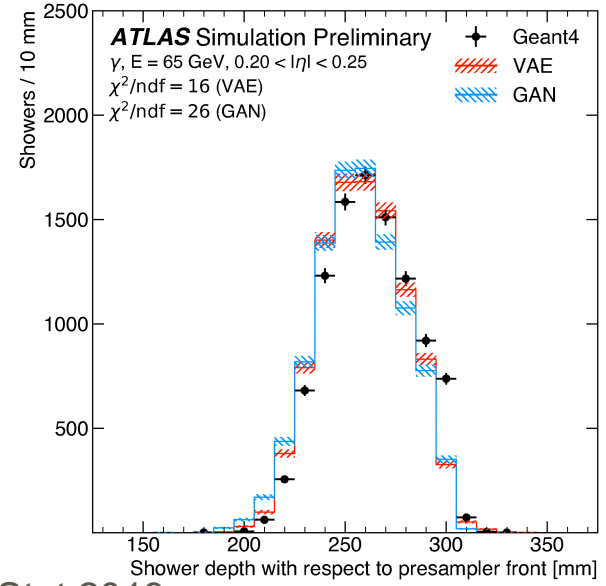
ATL-SOFT-PUB-2018-001



+ η, ϕ translation

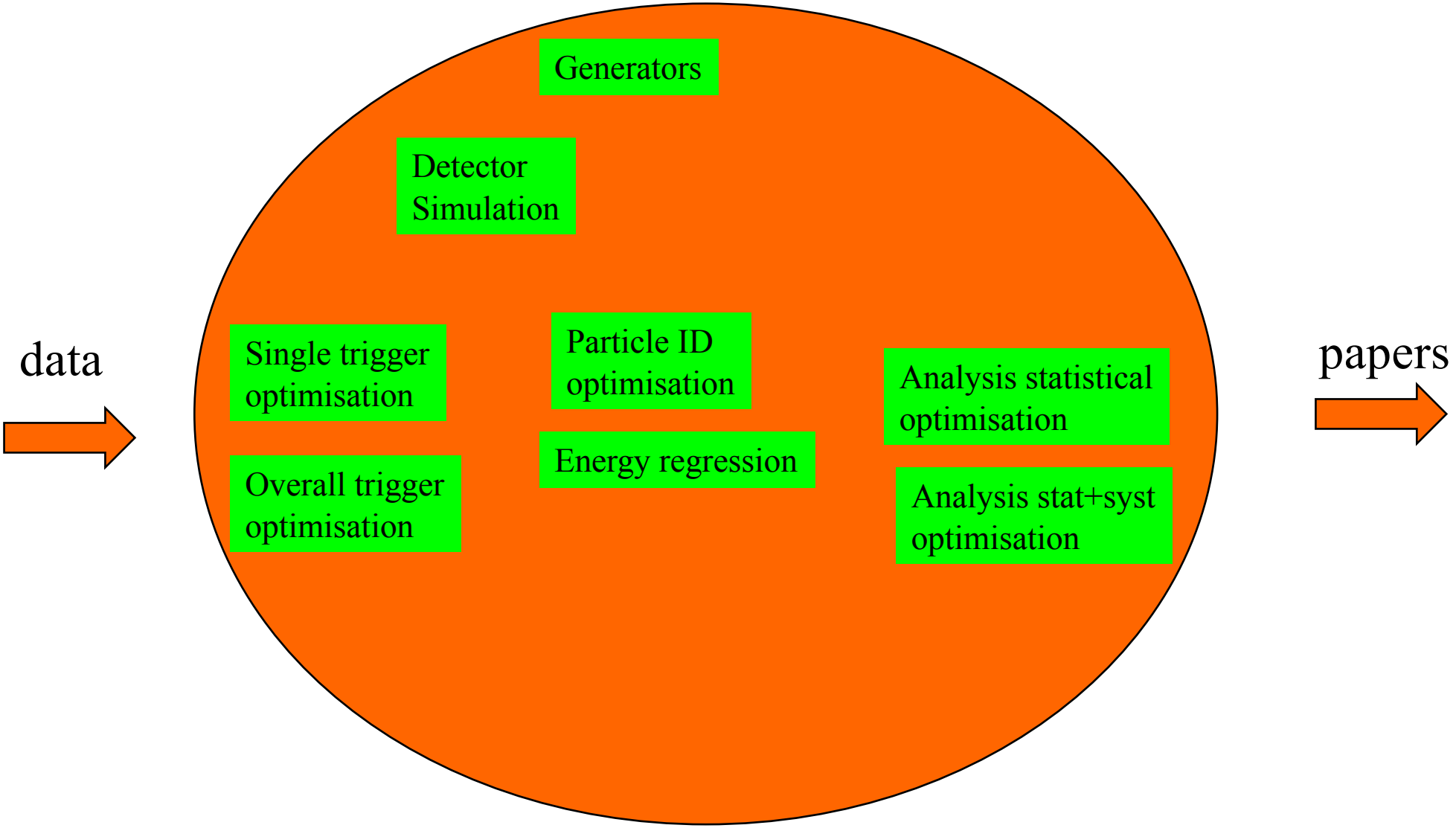
177000 cells ➔266 cells





Wrapping-up

ML playground



Conclusion (1)

- We (in HEP) are analysing data from multi-billion € projects➔should make the most out of it!
- Recent explosion of novel (for HEP) ML techniques, novel applications for Analysis, Reconstruction, Simulation, Trigger, and Computing
  - Do the same thing faster
  - Do better
- Some of these are ~easy, most are complex: open source software tools are ~easy to get, but still need (people) training, know-how
- Never underestimate the time for :
  - (1) Great ML idea➔
  - (2) …demonstrated on toy dataset➔
  - (3) …demonstrated on semi-realistic simulation ➔
  - (4) …demonstrated on real experiment analysis/dataset ➔
  - (5) …experiment publication using the great idea

Faster ML to production

- Training of HEP students and post-docs
  - … and senior scientists
- Campus-level sustained HEP ML collaborations
  - … not just workshops or challenges
- Public datasets
  - …not just toys but also real experimental ones
- Release software with papers
  - …matching "reproducibility" movement in ML
- Computing resources
  - …although (not yet) the limiting factor