Advances in Machine Learning in experimental High Energy Physics

David Rousseau

LAL-Orsay

rousseau@lal.in2p3.fr

IPHC seminar

16th Oct 2017

Outline

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

Focus on applications rather than details of the techniques

ML in HEP

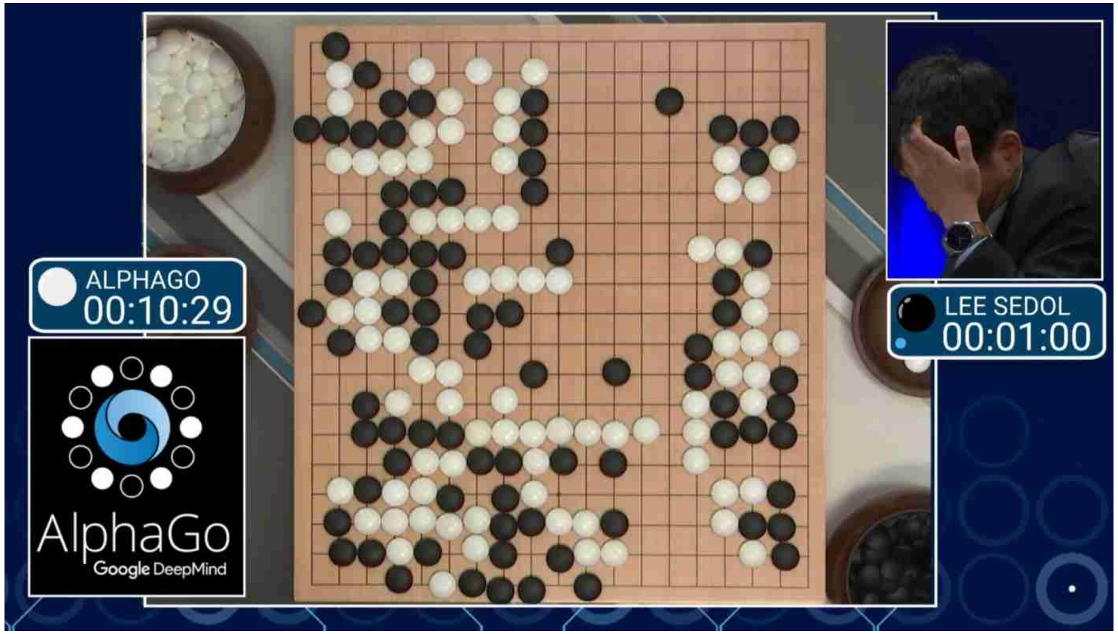
- Use of Machine Learning (a.k.a Multi Variate Analysis as we call it) already at LEP somewhat, much more at Tevatron (Trees)
- At LHC, Machine Learning used almost since first data taking (2010) for reconstruction and analysis
- In most cases, Boosted Decision Tree with Root-TMVA, on ~10 variables
- For example, impact on Higgs boson sensitivity at LHC:

analysis	data taking year	no ML sensitivity	ML sensitivity	ML data gain
ATLAS H $\to \gamma\gamma$ [16]	2011-2012	4.3	-	-
CMS H $\to \gamma\gamma$ [17]	2011-2012	?	2.7	?
ATLAS H $\to \tau^+\tau^-$ [18]	2012	2.5	3.4	85%
CMS H $\to \tau^+\tau^-$ [19]	2012	3.7	-	-
ATLAS VH $\to$ bb [20]	2012	1.9	2.5	73%
ATLAS VH $\to$ bb [21]	2015-2016	2.8	3.0	15%
CMS VH $\to$ bb [22]	2012	1.4	2.1	125%
CMS VH $\to$ bb [23]	2015-2016	-	2.8	-

➔~50% gain on LHC running

ML in HEP

- Meanwhile, in the outside world :



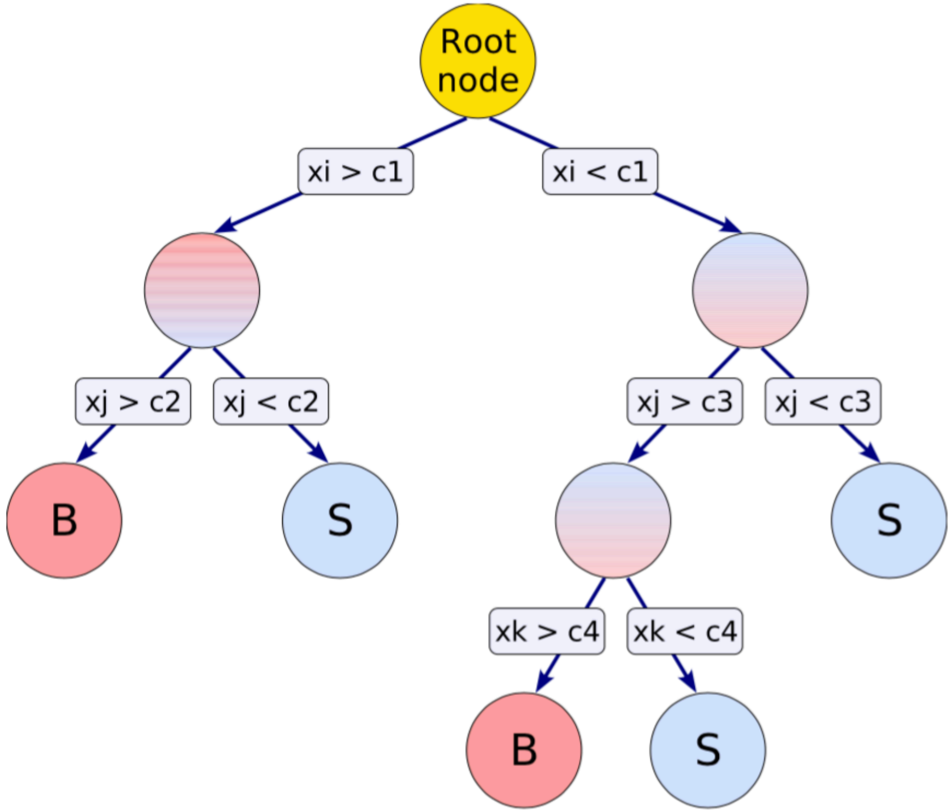
- "Artificial Intelligence" not a dirty word anymore!
- We've realised we're been left behind! Trying to catch up now…

Multitude of HEP-ML events

- HiggsML Challenge, summer 2014
  - ➔HEP ML NIPS satellite workshop, December 2014
- Connecting The Dots, Berkeley, January 2015
- Flavour of Physics Challenge, summer 2015
  - ➔HEP ML NIPS satellite workshop, December 2015
- DS@LHC workshop, 9-13 November 2015
- LHC Interexperiment Machine Learning group
  - Started informally September 2015, gaining speed
  - IML workshop @CERN 20-22 March 2017
- Moscou/Dubna ML workshop 7-9th Dec 2015
- Heavy Flavour Data Mining workshop, 18-21 Feb 2016
- Connecting The Dots, Vienna, 22-24 February 2016
- Hep Software Foundation workshop 2-4 May 2016 at Orsay, M
- Connecting The Dots, LAL-Orsay, 6-9 March 2017
- DS@HEP workshop @FNAL 8-12 May 2017
- ACAT conference Seattle, Sep 2017

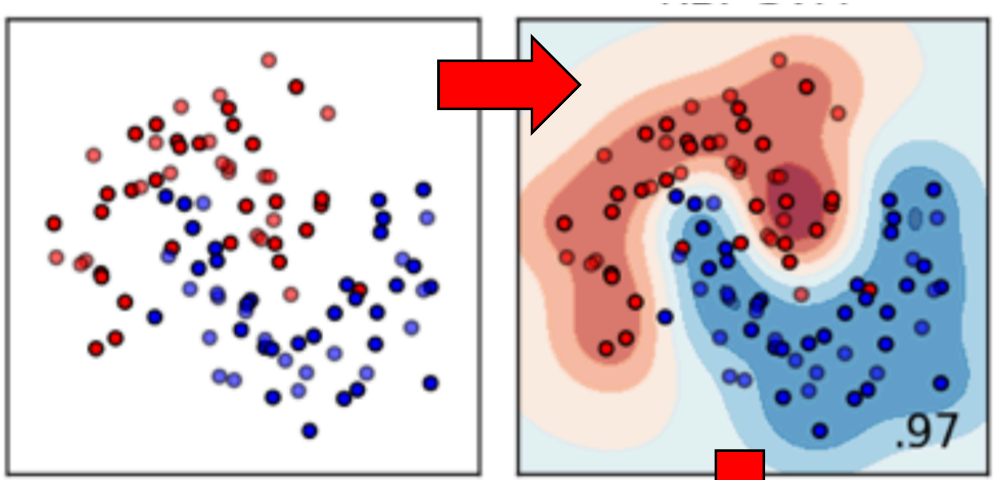
ML Basics

BDT in a nutshell

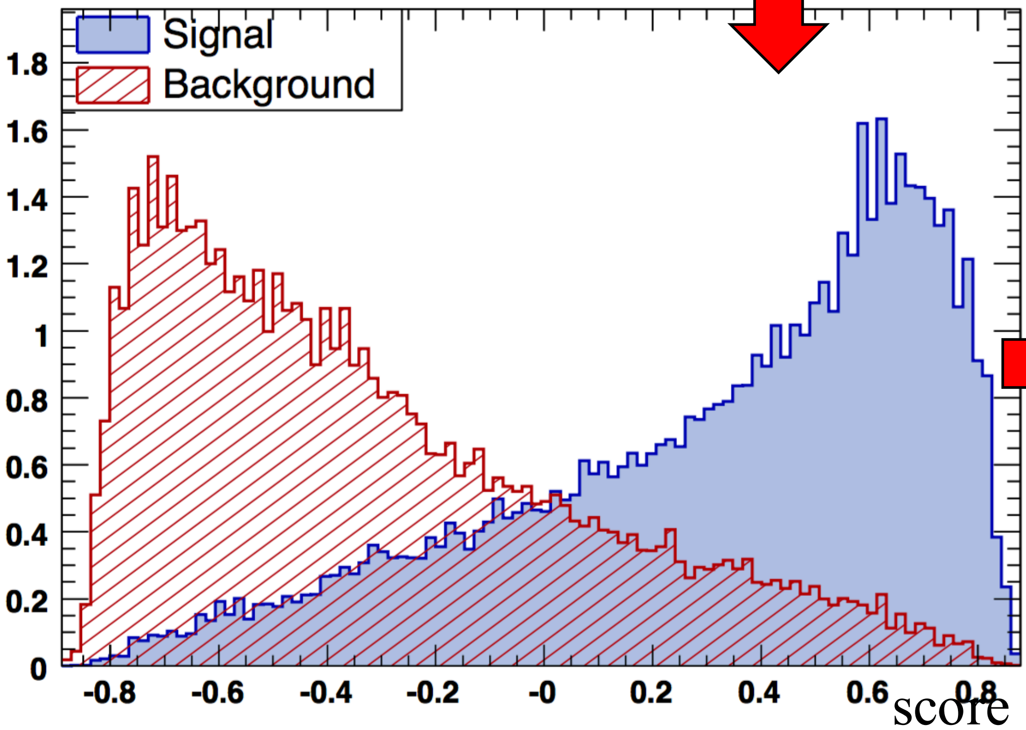


- Single tree (CART) <1980
- AdaBoost 1997 : rerun increasing the weight of misclassified entries ➔Boosted Decision Trees (Gradient BDT, random forest…)

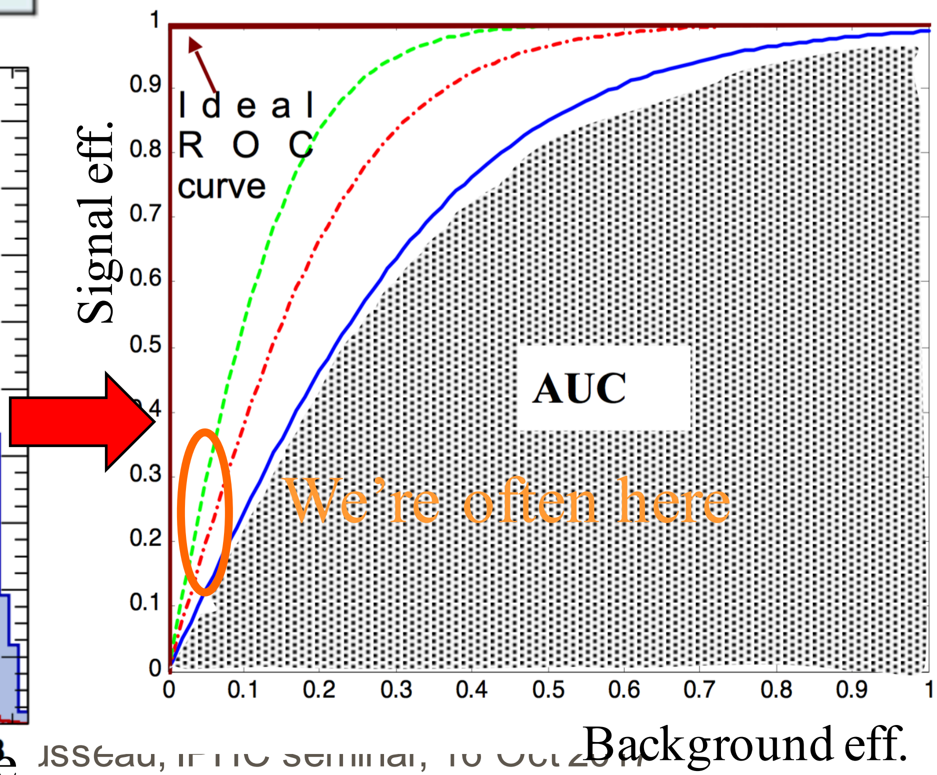
Classifier basics



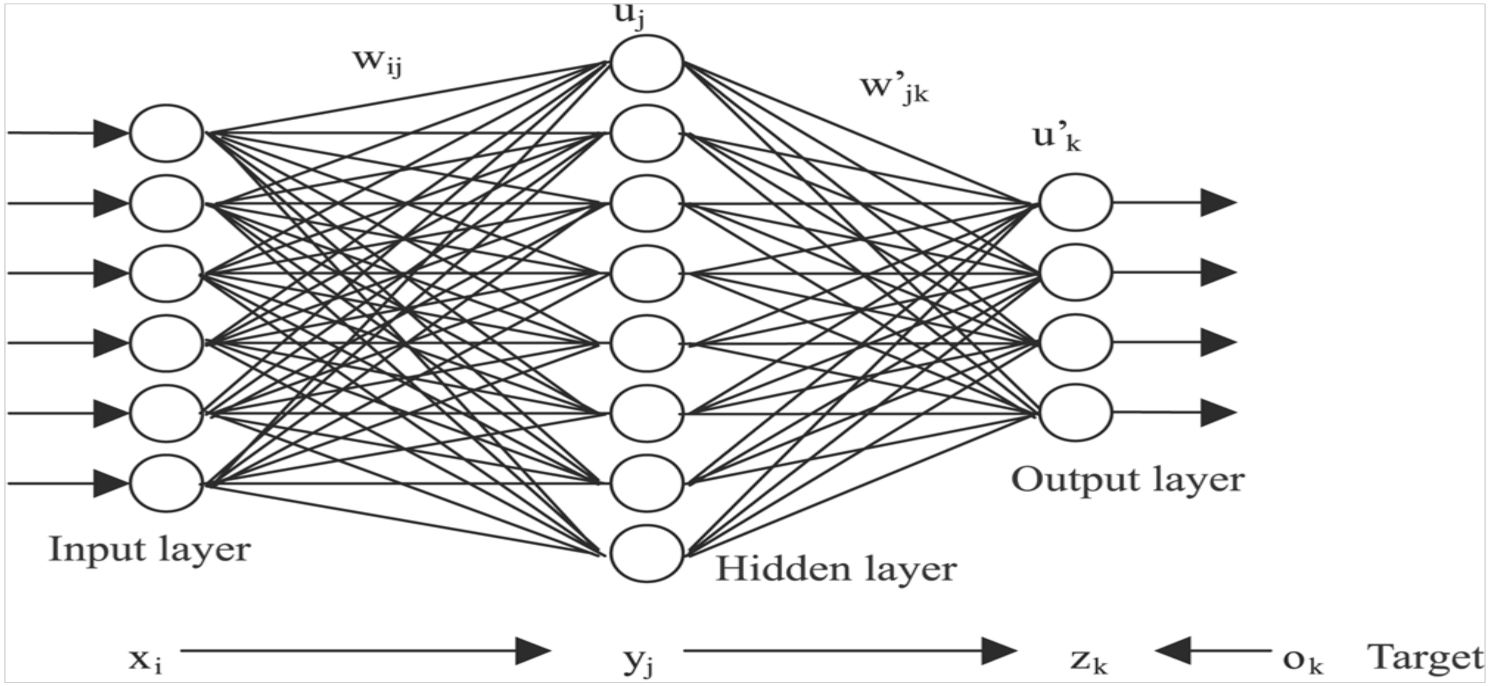
Classification : learn label 0 or 1
Regression : learn continuous variable



AUC : Area Under the (ROC) Curve

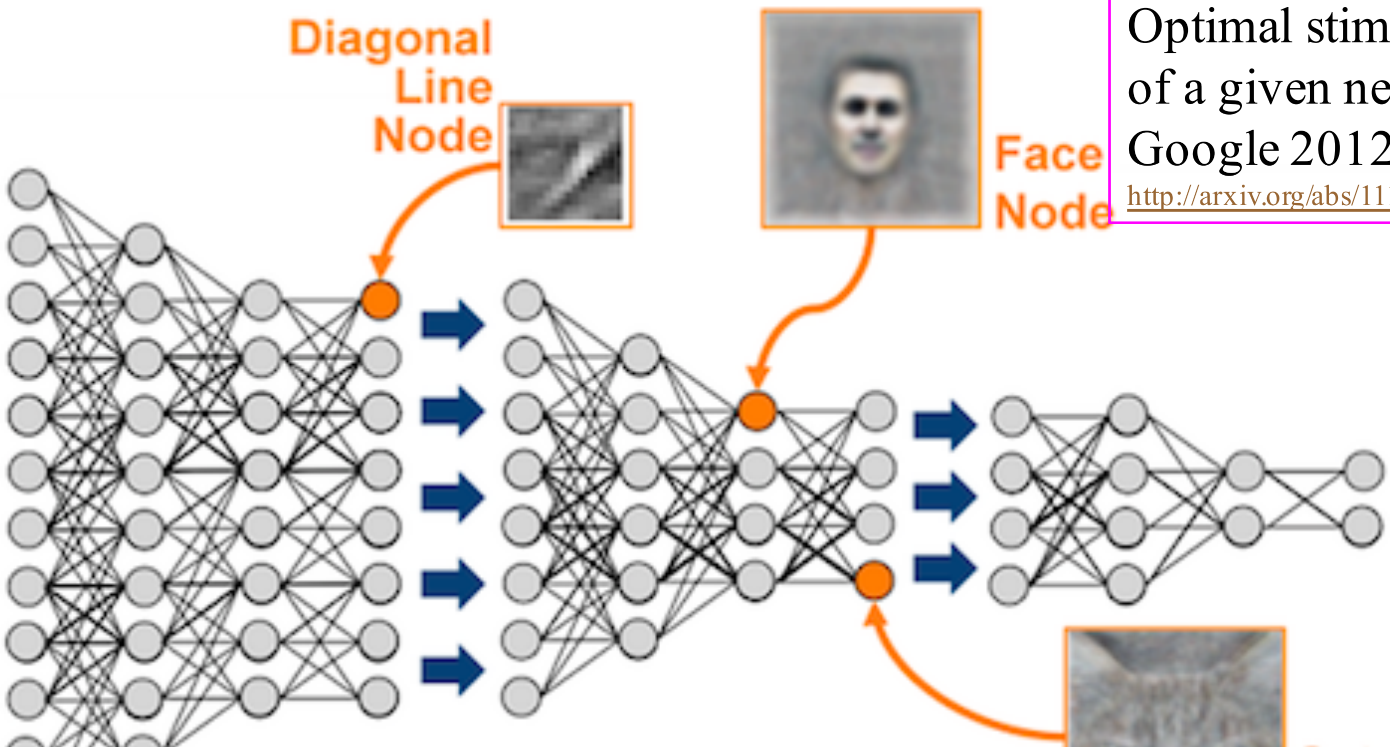


Neural Net in a nutshell

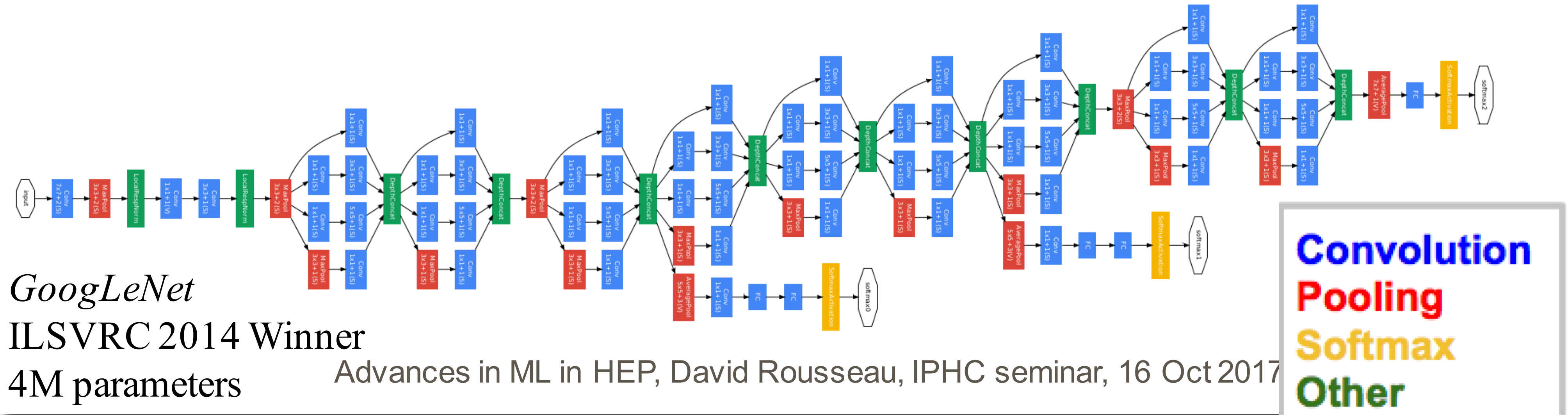


- Neural Net ~1950!
- But many many new tricks for learning, in particular if many layers (also ReLU instead of sigmoïd activation)
- "Deep Neural Net" up to 100 layers
- Computing power (DNN training can take days even on GPU)

Deep learning



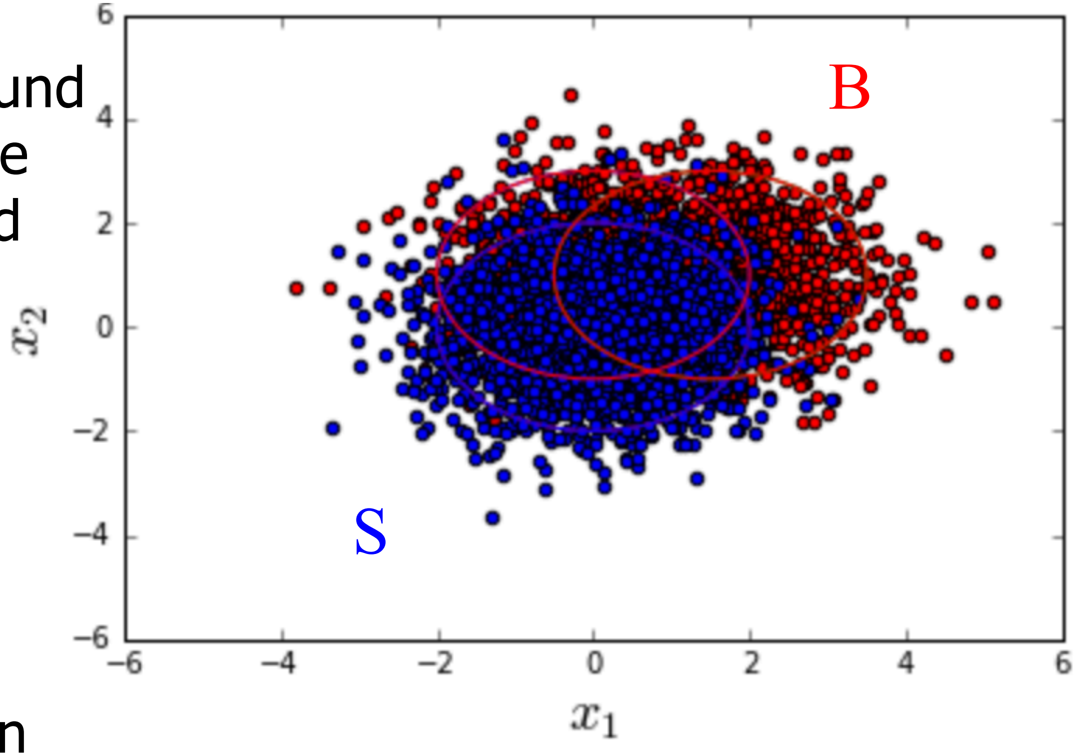
Optimal stimulus of a given neuron
Google 2012
http://arxiv.org/abs/1112.6209



GoogLeNet
ILSVRC 2014 Winner
4M parameters

No miracle

- ML (nor Artificial Intelligence) does not do any miracles
- For selecting Signal vs Background and underlying distributions are known, nothing beats Likelihood ratio! (often called "bayesian limit"):
  - $L_S(x)/L_B(x)$
- OK but quite often L_S L_B are unknown
  - + x is n-dimensional
- ML starts to be interesting when there is no proper formalism of the pdf
- ➔mixed approach, if you know something, tell your classifier instead of letting it guess

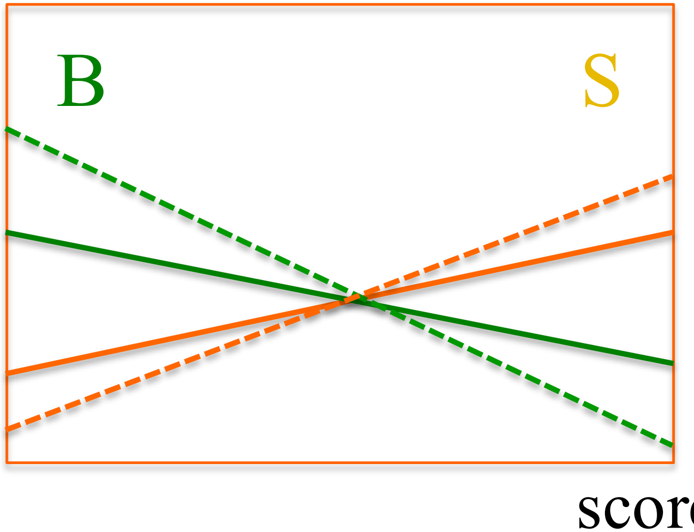


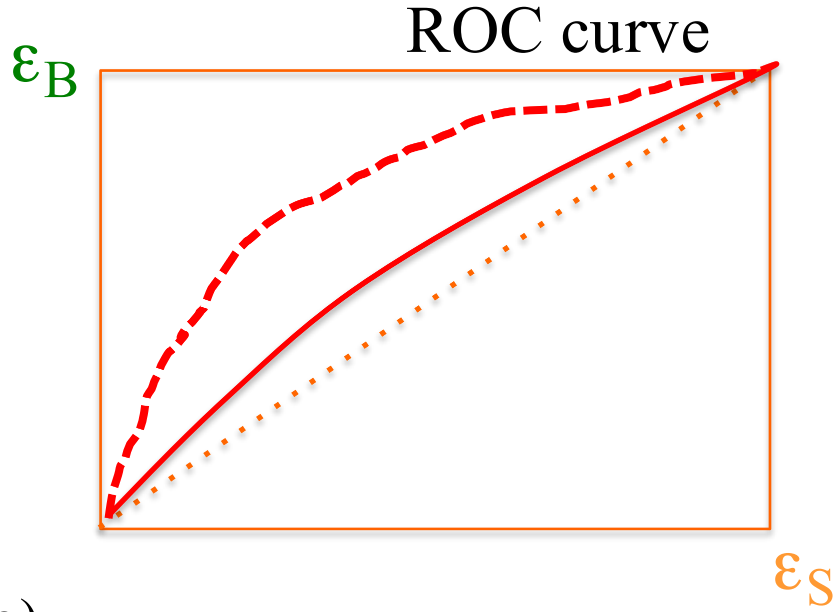
Some vocabulary

- "Features"
  - variables
- "Hyper-parameters":
  - These are all the "knobs" to optimize an algorithm, e.g.
    - number of leaves and depth of a tree
    - number of nodes and layers for NN
    - and much more
  - "Hyper-parameter tuning/fitting"== optimising the knobs for the best performance

ML Techniques

Overtraining





Evaluated on training dataset (wrong)

Evaluated on independent test dataset (correct)

Score distribution different on test dataset wrt training dataset

➔"Overtraining"== possibly excessive use of statistical fluctuation

under/over training

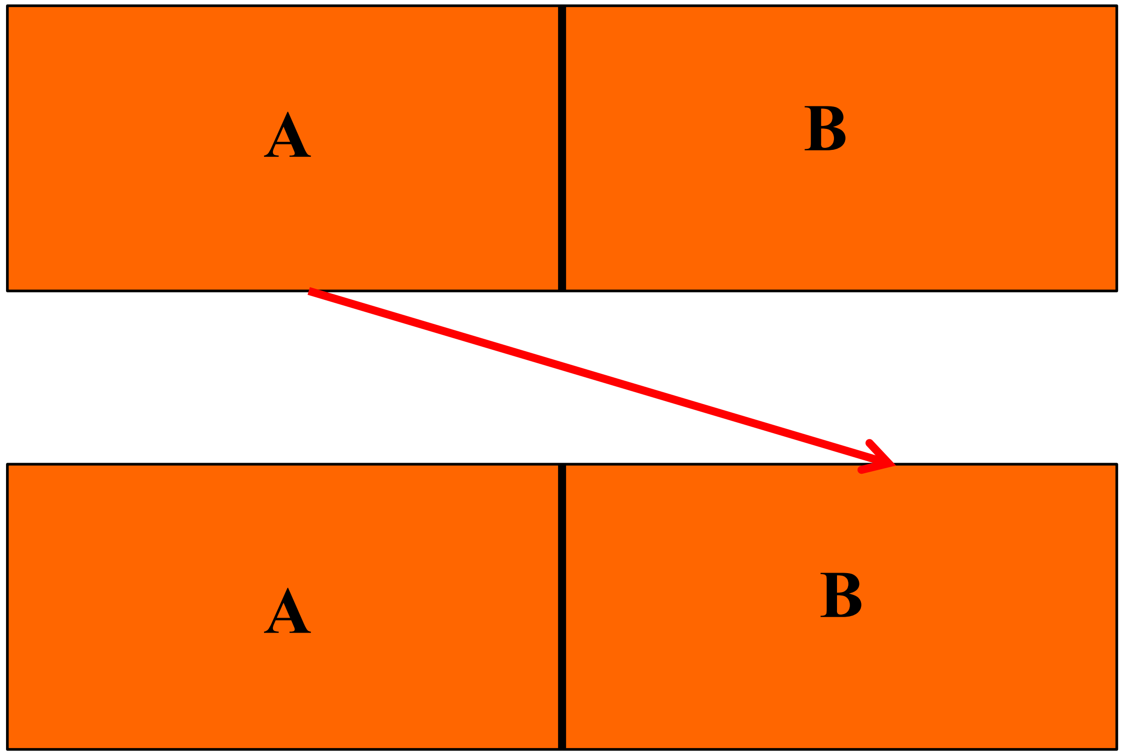
Gilles Louppe, github

Some overtraining is good!

Cross-Validation

One-fold Cross Validation

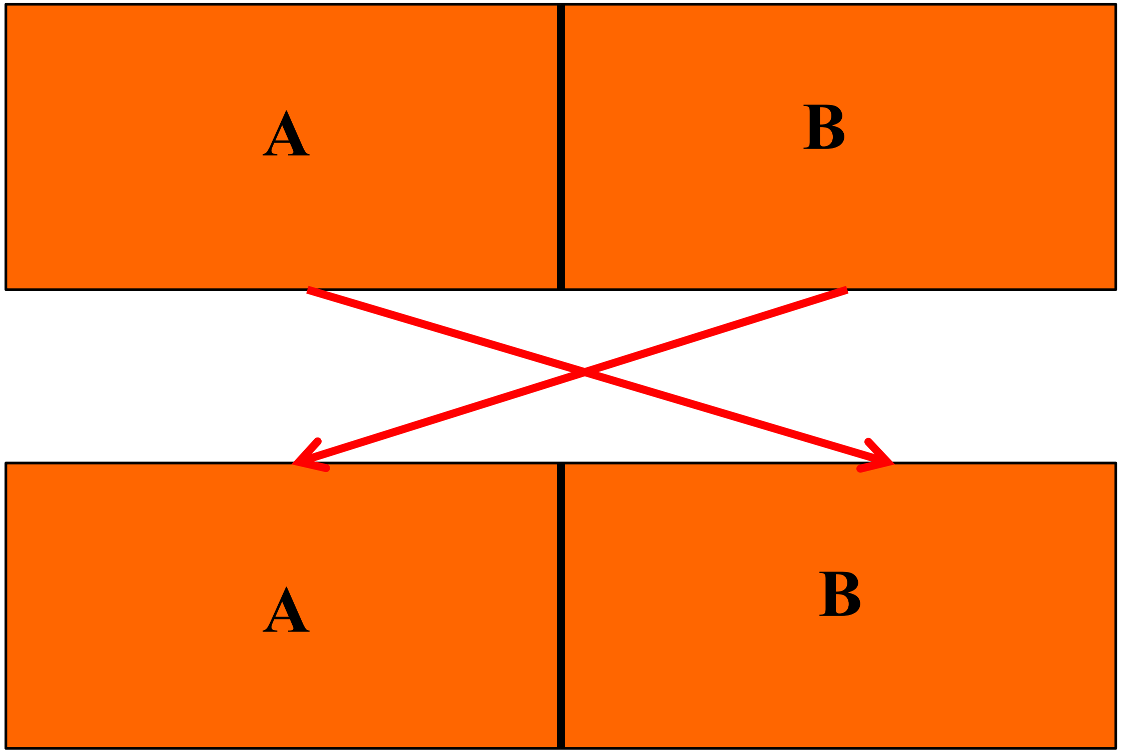
Goal of CV is to measure performance and optimise hyper-parameters



Standard basic way (default TMVA until recently)

Cross-Validation

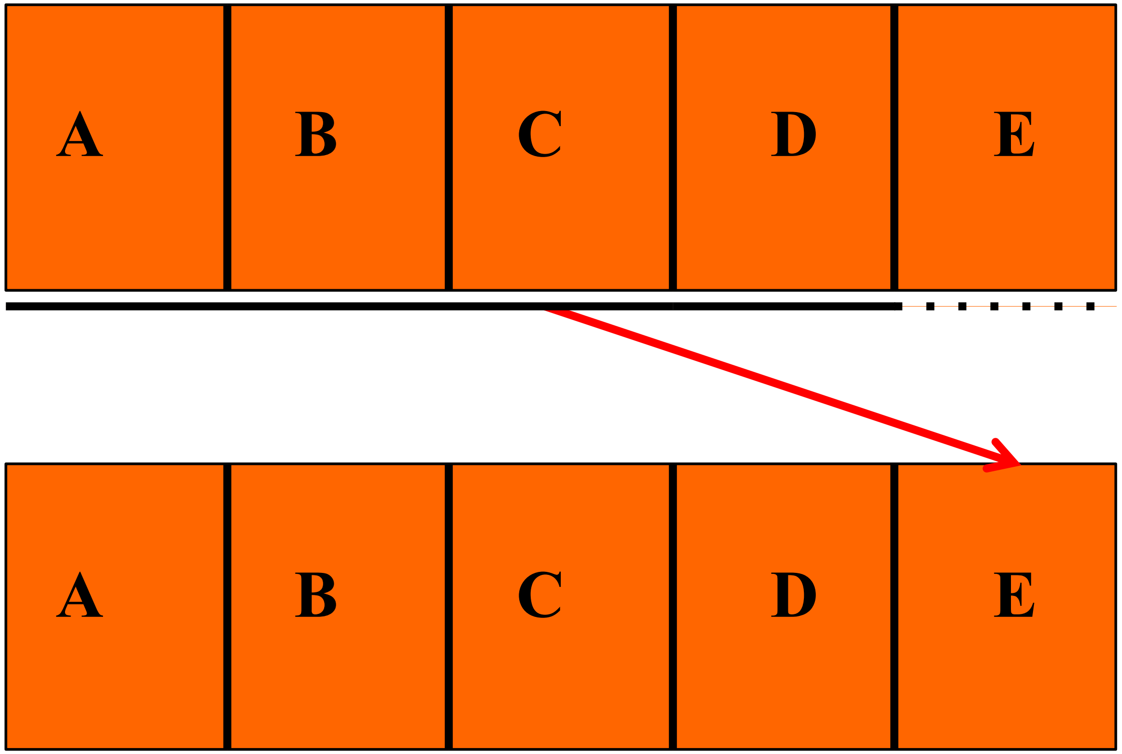
Two-fold Cross Validation



➔test statistics = total statistics
➔double test statistics wrt one fold CV
➔(double training time of course)

Cross-Validation

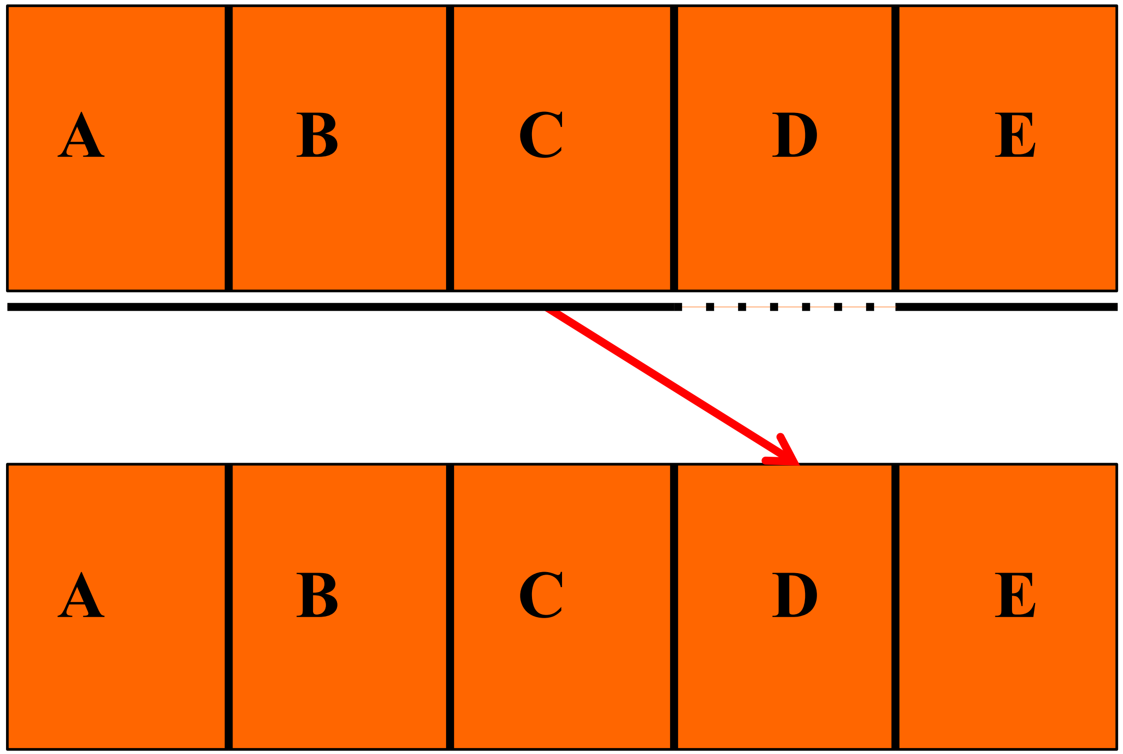
5-fold Cross Validation



same test statistics wrt two-fold CV,
larger training statistics 4/5 over ½ (larger training time as well)
bonus: variance of the samples an estimate of the statistical uncertair

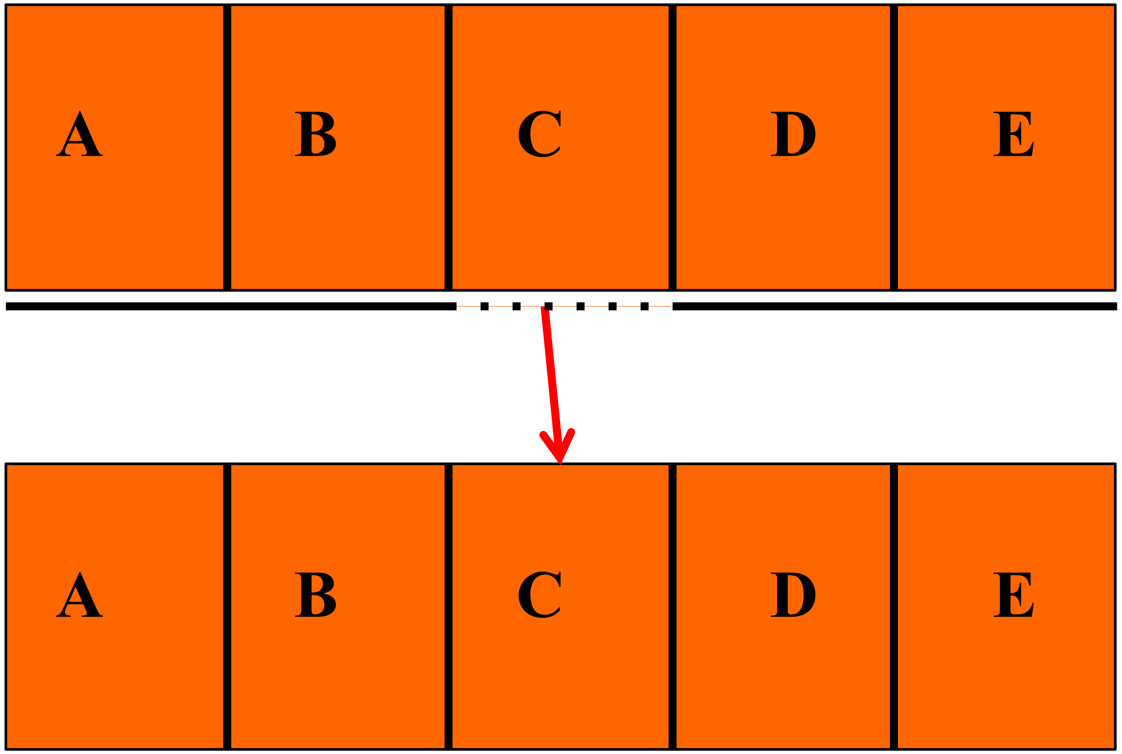
Cross-Validation

5-fold Cross Validation



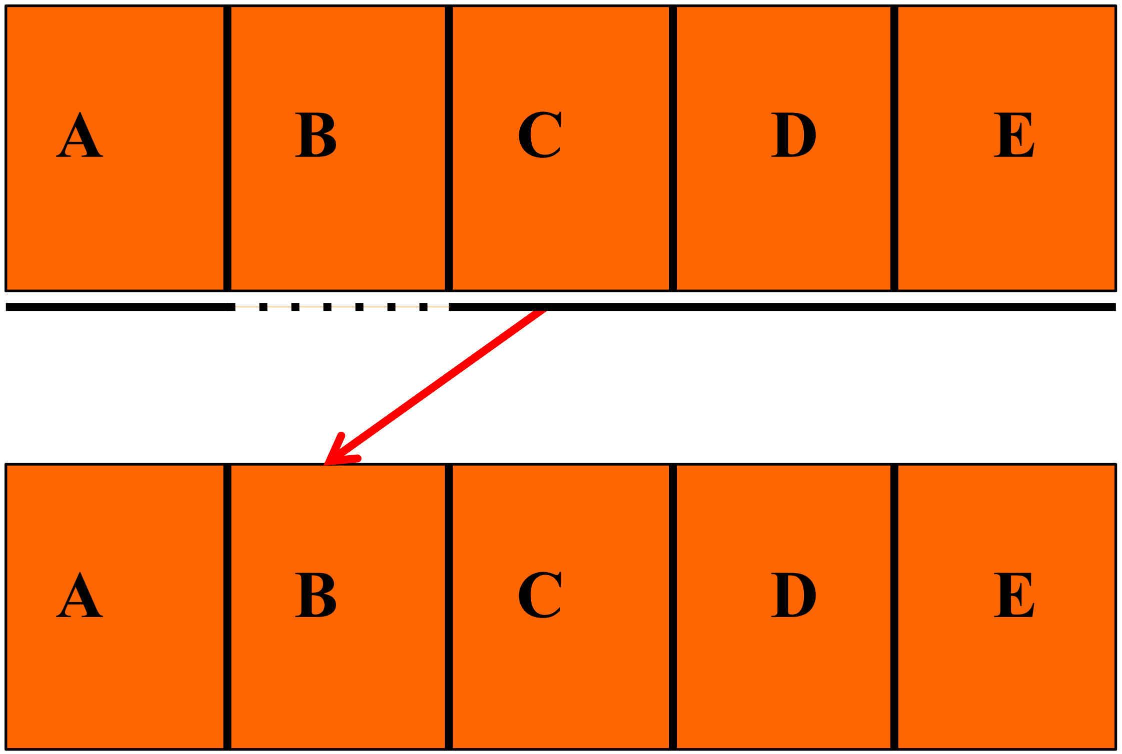
Cross-Validation

5-fold Cross Validation



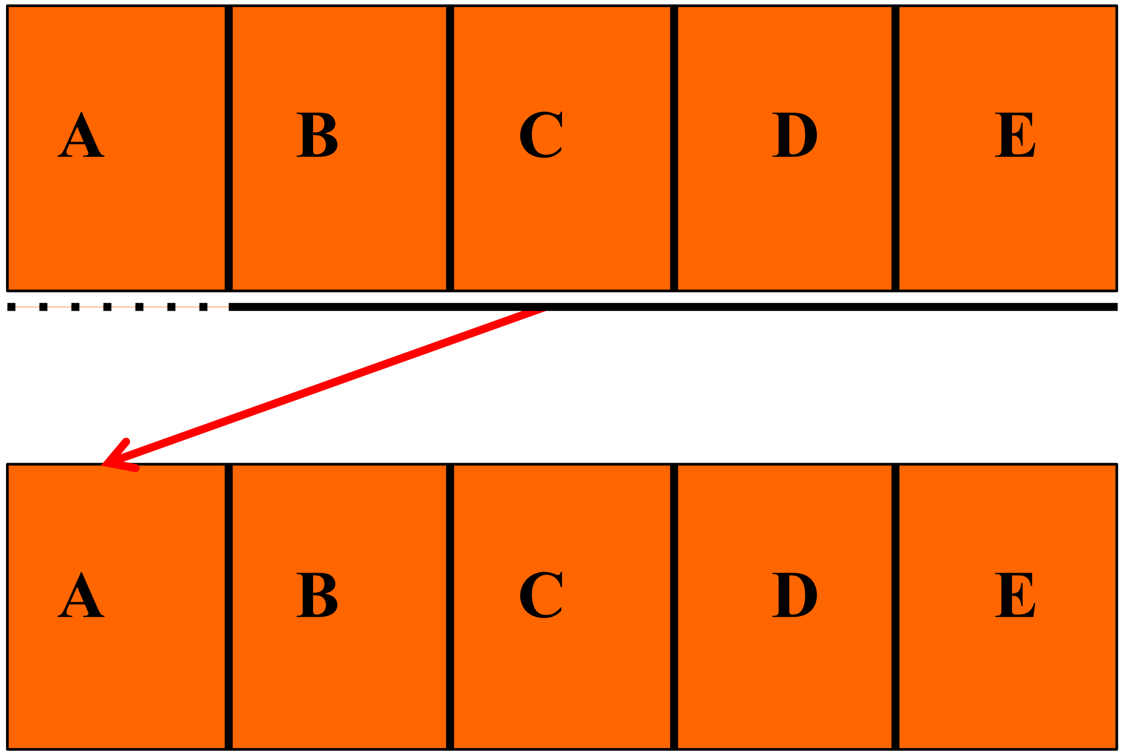
Cross-Validation

5-fold Cross Validation



Cross-Validation

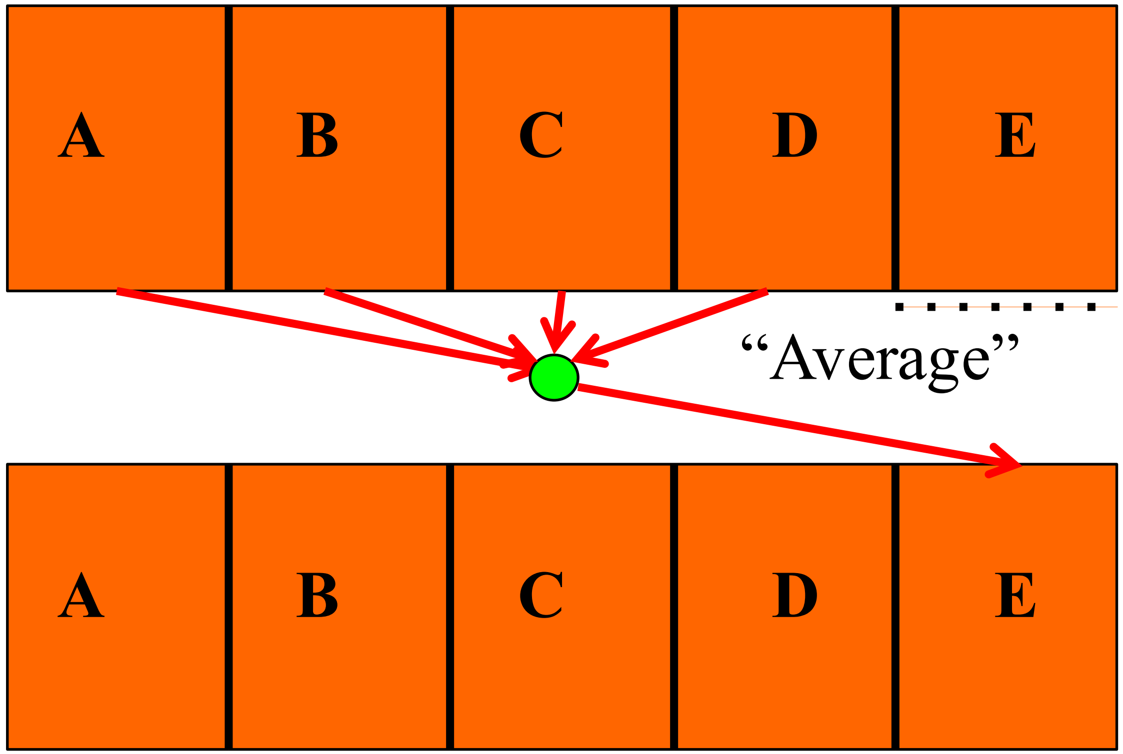
5-fold Cross Validation



Note : if hyper-parameter tuning, need a third level of independent sample "nested CV"

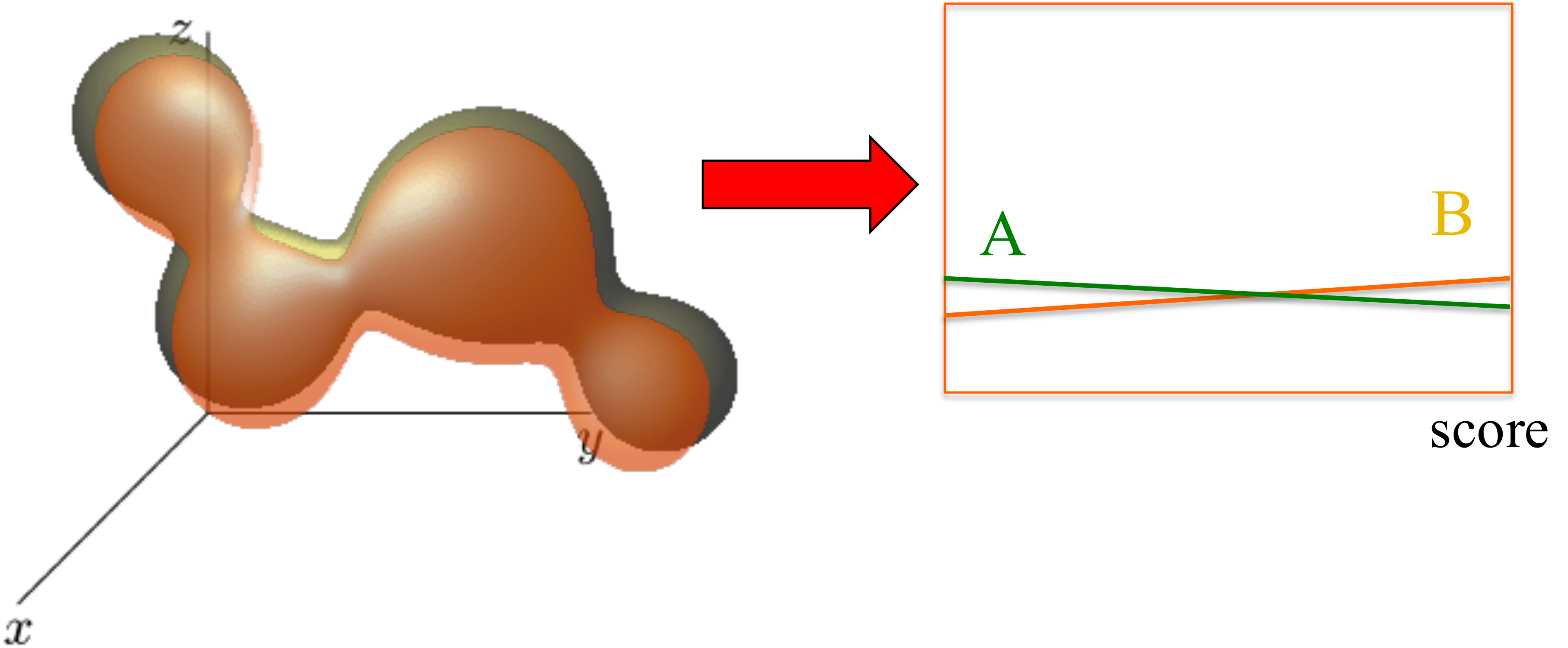
Cross-Validation

5-fold Cross Validation "à la Gabor"



Average of the scores on A B C D is
often better than the score of one training ABCD
(also save on training time)

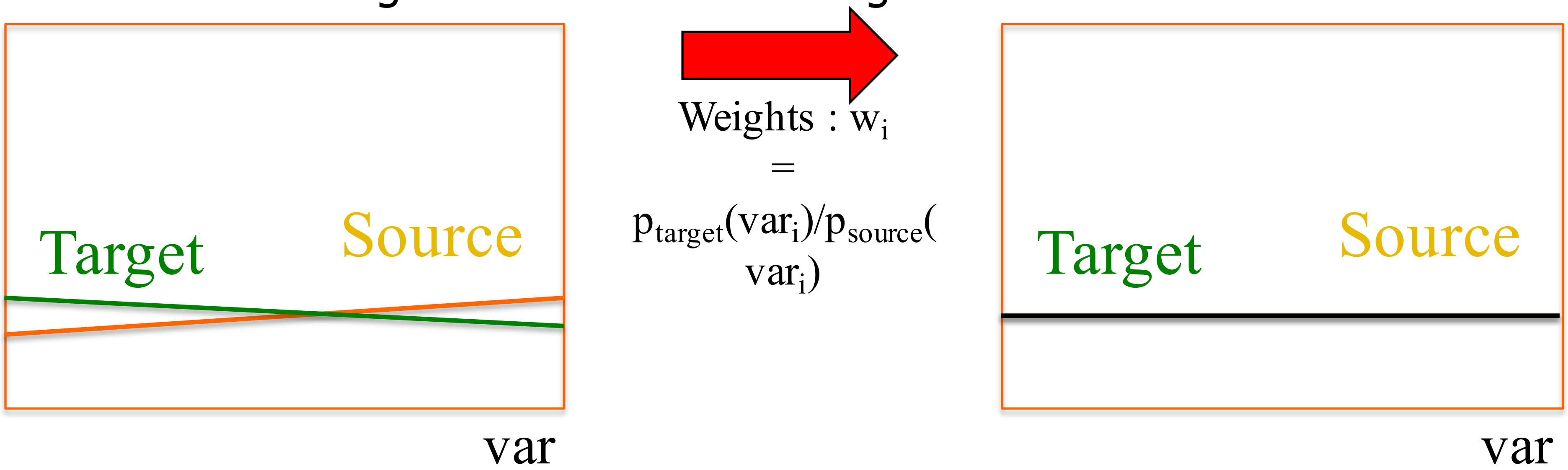
What does a classifier do?



- The classifier "projects" the two multidimensional "blobs" maximising the difference, without (ideally) any loss of information

Re-weighting

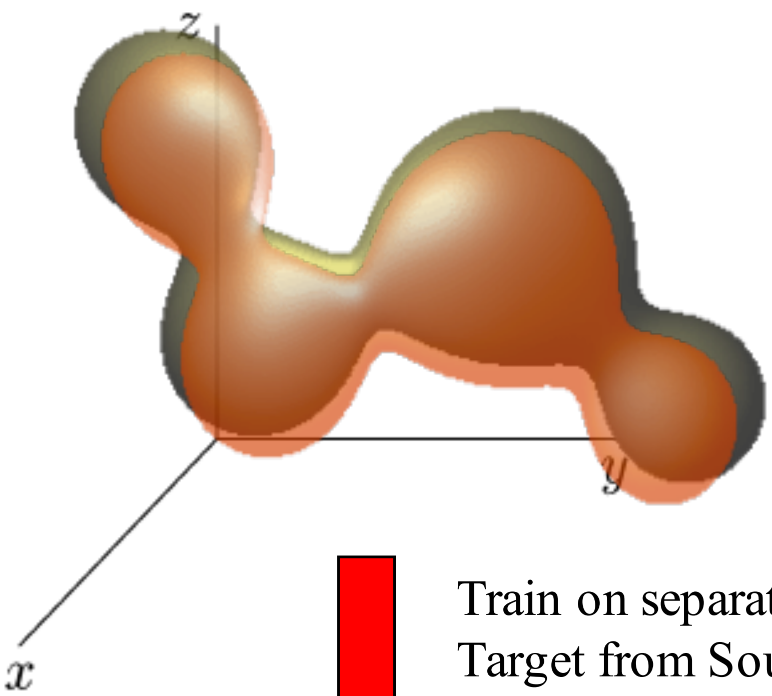
- Suppose a variable distribution is slightly different between a Source (e.g. Monte Carlo) and a Target (e.g. real data)
  - →reweight!   …then use reweighted events



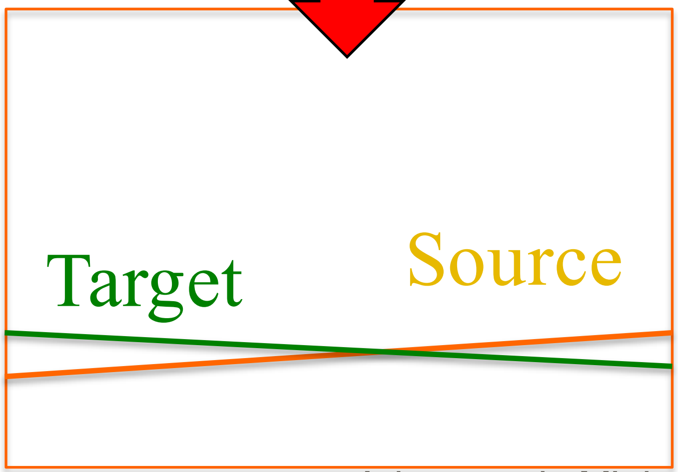
- What if multi-dimension ?
- Usually : reweight separately on 1D projections, at best 2D, because of quick lack of statistics
- Can we do better ?

Multidimension reweighting

[See demo on Andrei Rogozhnikov github]() and also [Kyle Cranmer's github]()



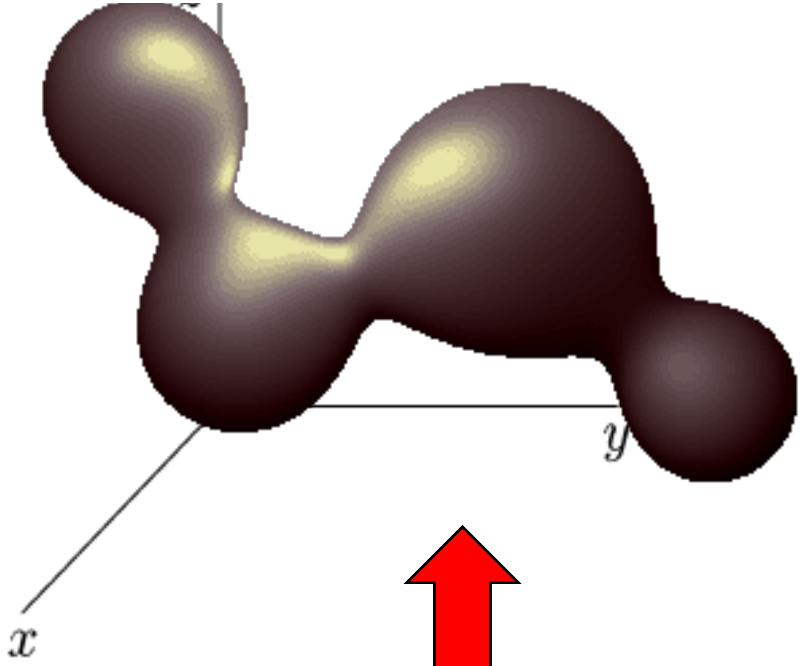
Train on separating
Target from Source

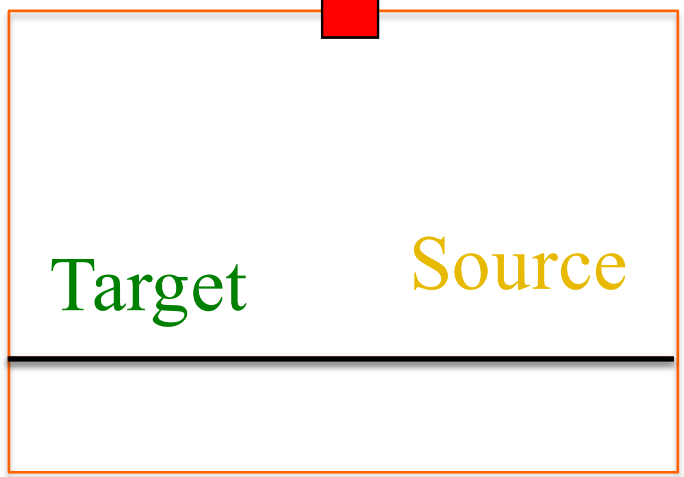


score

Weights : w_i

$$= P_{target}(score_i)/p_{source}(score_i)$$



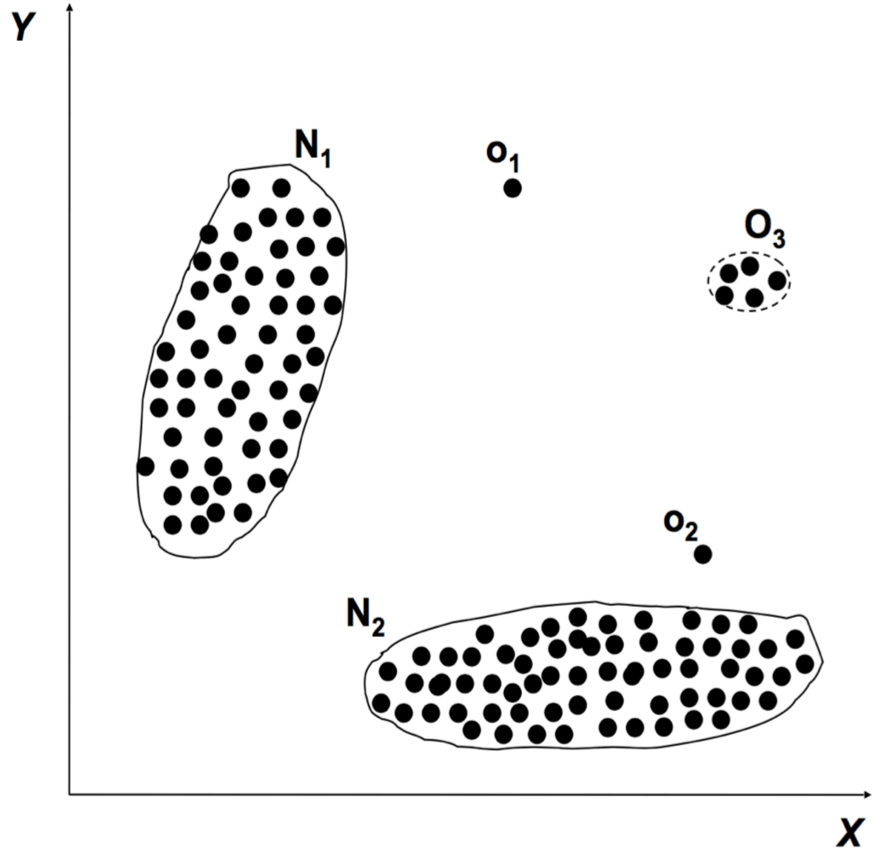


score

Multi dimensional reweighting (2)

- Reweighting the Source distribution on the score allows multidimensional reweighting without statistics problem
- Usual caveat still hold : Target support should be included in Source support, distributions should not be too different otherwise unmanageable very large or very small weights
- (Note : “reweighting” in HEP language <==> “importance sampling” in ML language)

Anomaly : point level

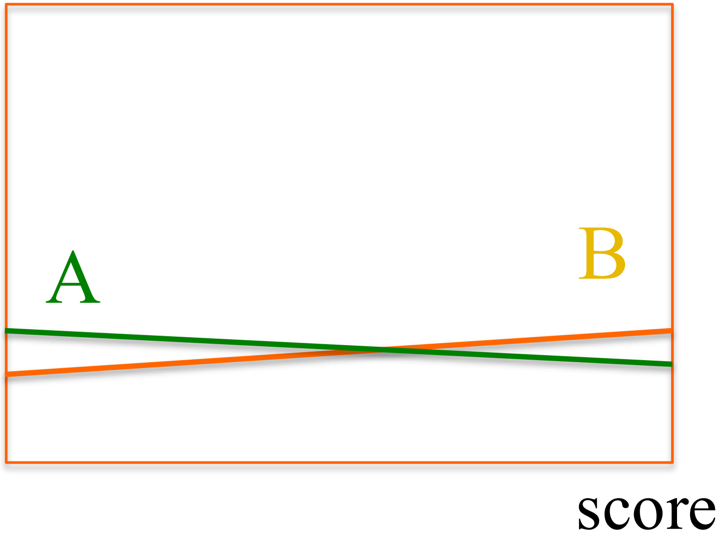


- Also called outlier detection
- Two approaches:
  - Unsupervised : give the full data, ask the algorithm to cluster and find the lone entries : o1, o2, O3
  - Supervised : we have a training "normal" data set with N1 and N2. Algorithm should then spot o1,o2, O3 as "abnormal" i.e. "unlike N1 and N2" (no a priori model for outliers)
- Application : detector malfunction, grid site malfunction, or even new physics discovery…

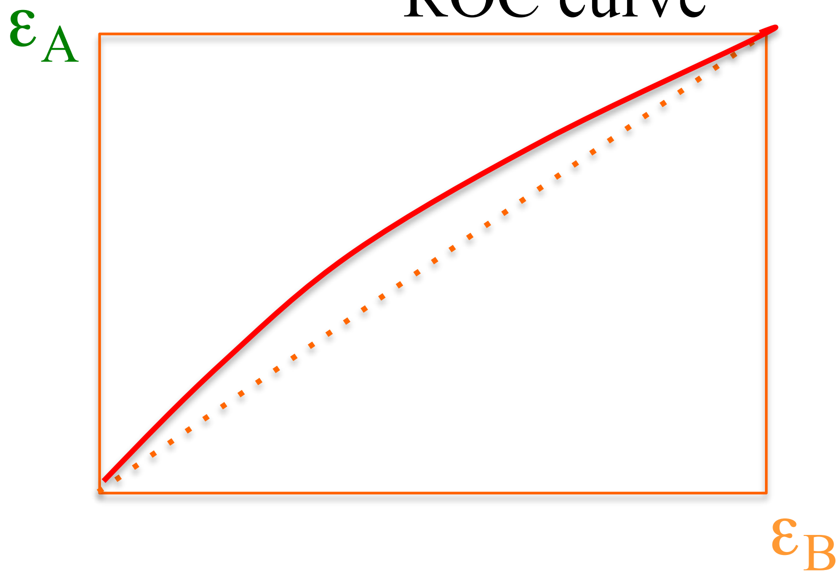
Anomaly : population level

- Also called collective anomalies
- Suppose you have two independent samples A and B, *supposedly* statistically identical. E.g. A and B could be:
  - MC prod 1, MC prod 2
  - MC generator 1, MC generator 2
  - Geant4 Release 20.X.Y, release 20.X.Z
  - Production at CERN, production at BNL
  - Data of yesterday, Data of today
- How to verify that A and B are indeed identical ?
- Standard approach : overlay histograms of many carefully chosen variables, check for differences (e.g. KS test)
- One ML approach (not the only one): ~~ask an artificial scientist~~, train your favorite classifier to distinguish A from B, histogram the score, check the difference (e.g. AUC or KS test)
  - ➔only one distribution to check

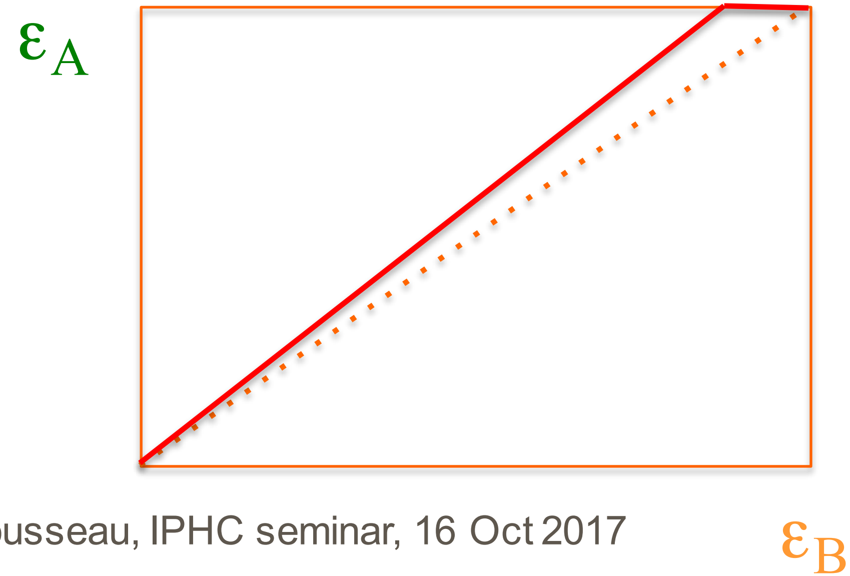
Small non-local difference



ROC curve



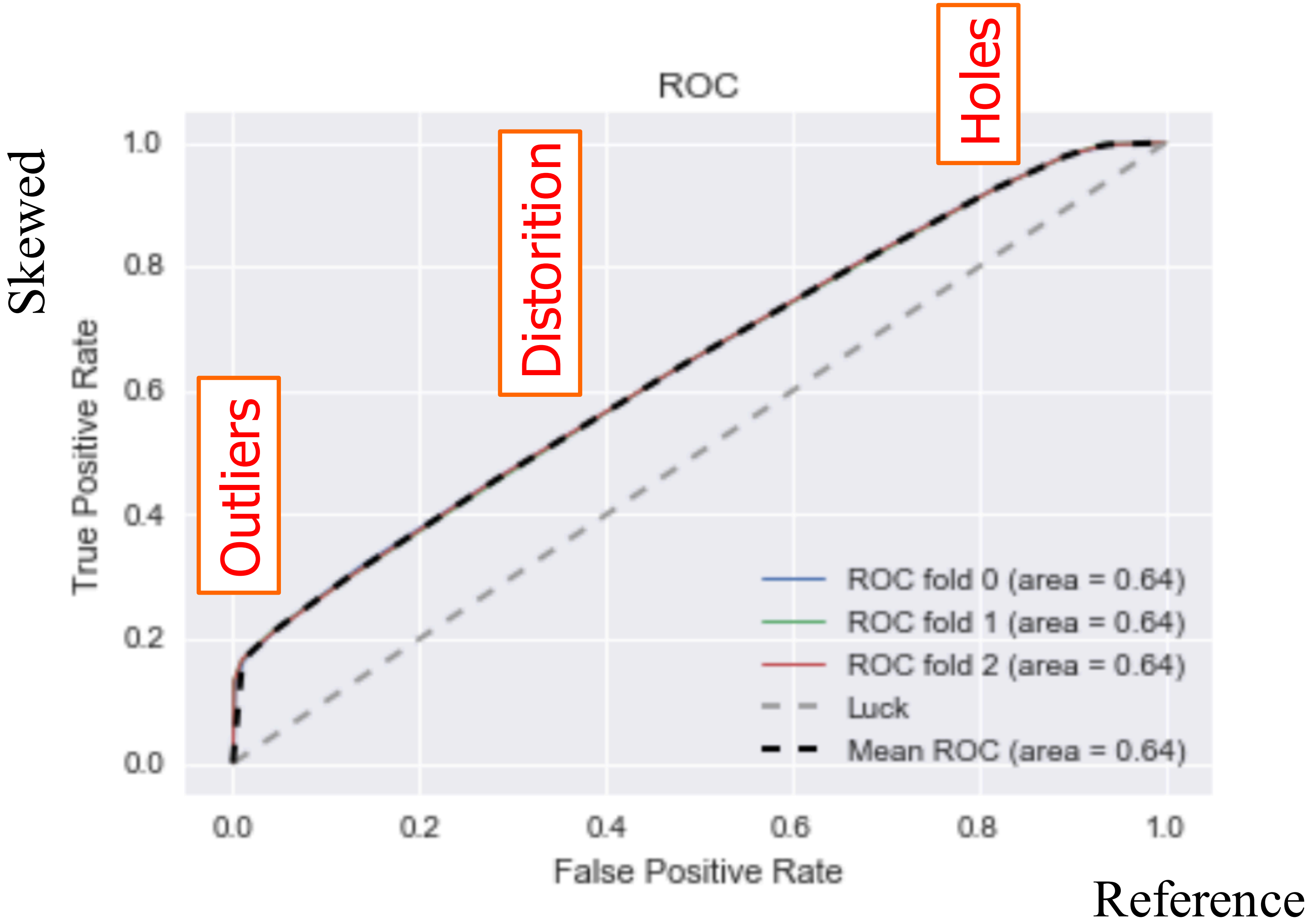
Local big difference (e.g. non overlapping distribution, hole)



HSF ML RAMP on anomaly

- Rapid Analysis Model Prototyping : collaborative competition around a dataset and a figure of merit. Organised in June 2016 by CDS Paris Saclay with HEP people. See agenda.
- Dataset built from the Higgs Machine Learning challenge dataset (on CERN Open Data Portal)
  - Lepton, and tau hadron 3 momentum, MET : PRImary variables
  - DERived variables e.g various invariant masses (computed from the above) from Htautau analysis
  - ➔reference dataset
- "Skewed" dataset built from the above, introducing small and big distortions:
  - Change of tau energy scale (Small scaling of Ptau)
  - Holes in eta phi efficiency map of lepton and tau hadron
  - Outliers introduced, each with 5% probability
    - Eta tau set to large non possible values
    - P lepton scaled by factor 10
    - Missing ET + 50 GeV
    - Phi tau and phi lepton swapped ➔ DERived variables inconsistent with PRImary one
  - ➔skewed dataset

HSF ML RAMP on anomaly (2)



HSF RAMP (2)

team	submission	accuracy
mcherti	adab2_mt1_calibrated	0.611
dhrou	adab2_mt1	0.611
kazeevn	GradientBoosting	0.596
glouppe	bags2	0.594
glouppe	boosting-duo	0.595
mcherti	adaboost2	0.594
glouppe	bags	0.593
mcherti	adaboost1	0.593
djabbz	beta tester	0.591
soobash	ExtraTreesClassifier	0.576
mcherti	extratrees1	0.562
dhrou	DRv0	0.553
calaf	starting_kit_paolo	0.526

Breakthrough : add new variable:
$\Delta m_T = \sqrt{(2 P_{lT} * MET * (1 - \cos(\phi_l - \phi_{MET})))} - m_T$
Non zero for some outliers
➔classifiers were unable to guess it

Classifier optimisation

➔what functional form classifiers can learn ?

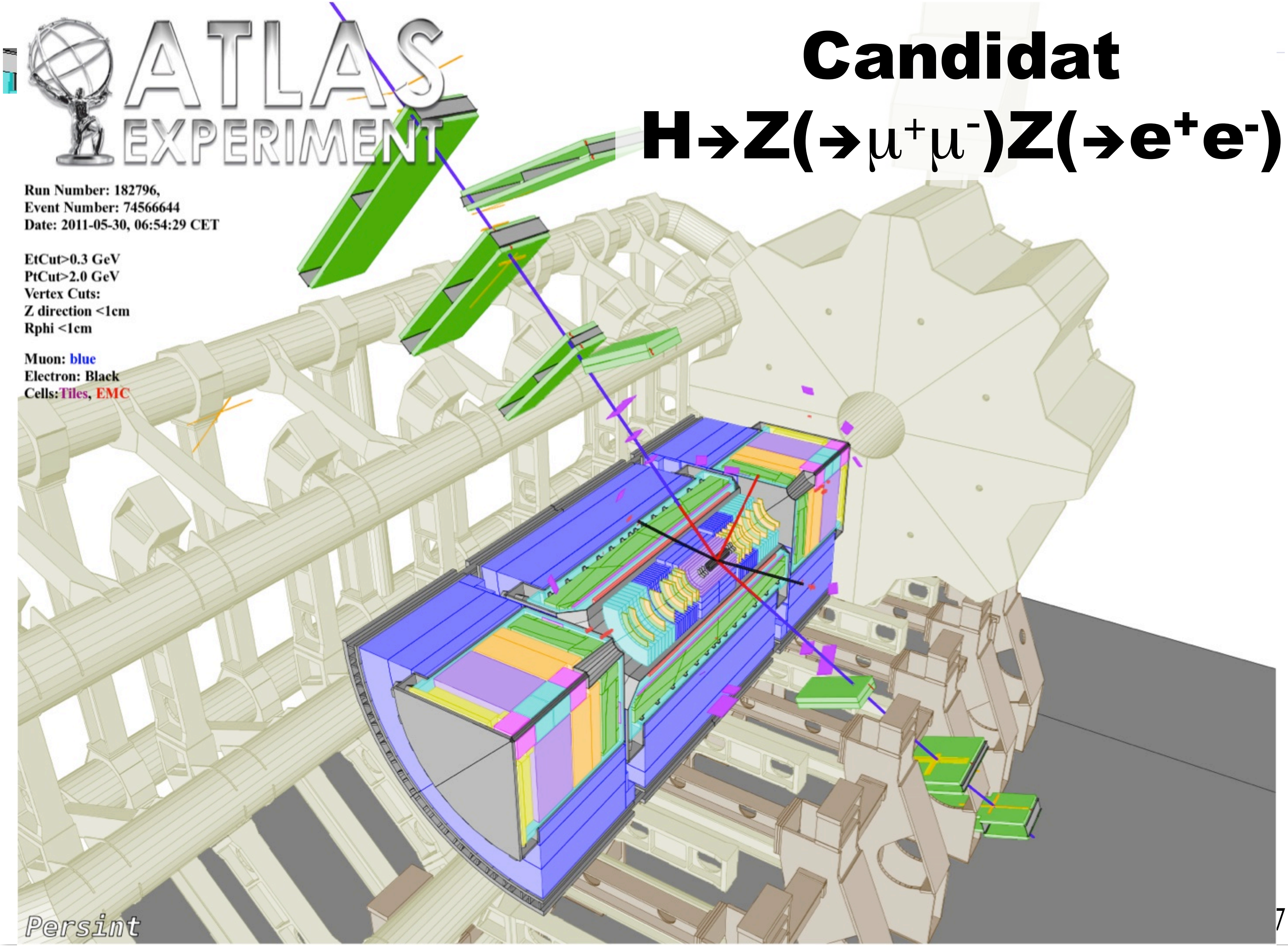
ML Tools

Modern Software and Tools

- New version of TMVA (root 6.0.8 on beyond) (see talk Lorenzo Moneta, Sergei Gleyzer IML workshop CERN March 2017)
  - Jupyter interface
  - Hyper-parameter optimisation
  - Cross-validation
  - (…unfortunately not so well documented yet)
- Non HEP software
  - Sci-kit learn : de facto standard toolbox ML (except Deep Learning) (python, but fast)
  - Theano + Keras : NN toolbox (build a NN in a few lines of python)
  - XGBoost best BDT on the market, both speed and performance (c++ with python interface
- Note : for ~10 variable classification/regression task BDT is still the tool of choice!
- Platforms
  - Your laptop is sufficient in many cases : install e.g. Anaconda https://docs.continuum.io/anaconda/install (demo)
  - If not, more and more platforms looking for users, maybe on your campus (with GPU DNN ==millions of parameter to optimise=>heavy duty linear algebra)
  - 50 GPU platform at Lyon CC-IN2P3, little used so far
- For CERN users:
  - SWAN interactive data analysis on the web see https://swan.web.cern.ch/content/machine-learning
  - CVMFS ML setup for any CVMFS enabled platform

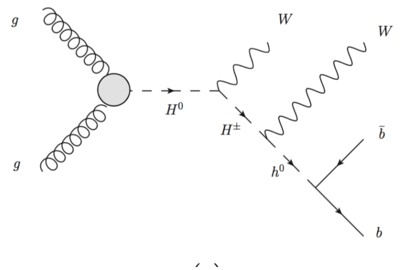
ML in analysis

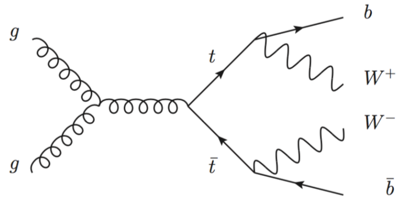
Candidat $H \rightarrow Z(\rightarrow \mu^+\mu^-)Z(\rightarrow e^+e^-)$

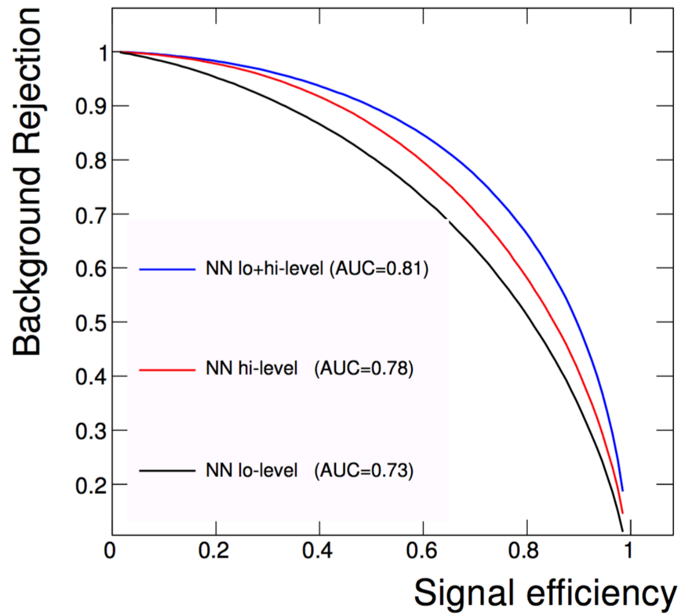


Deep learning for analysis

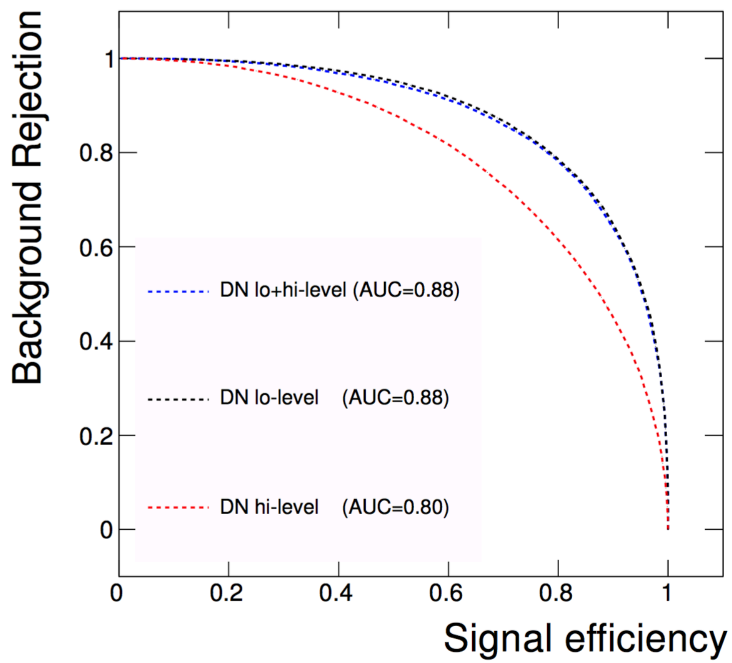
1402.4735 Baldi, Sadowski, Whiteson







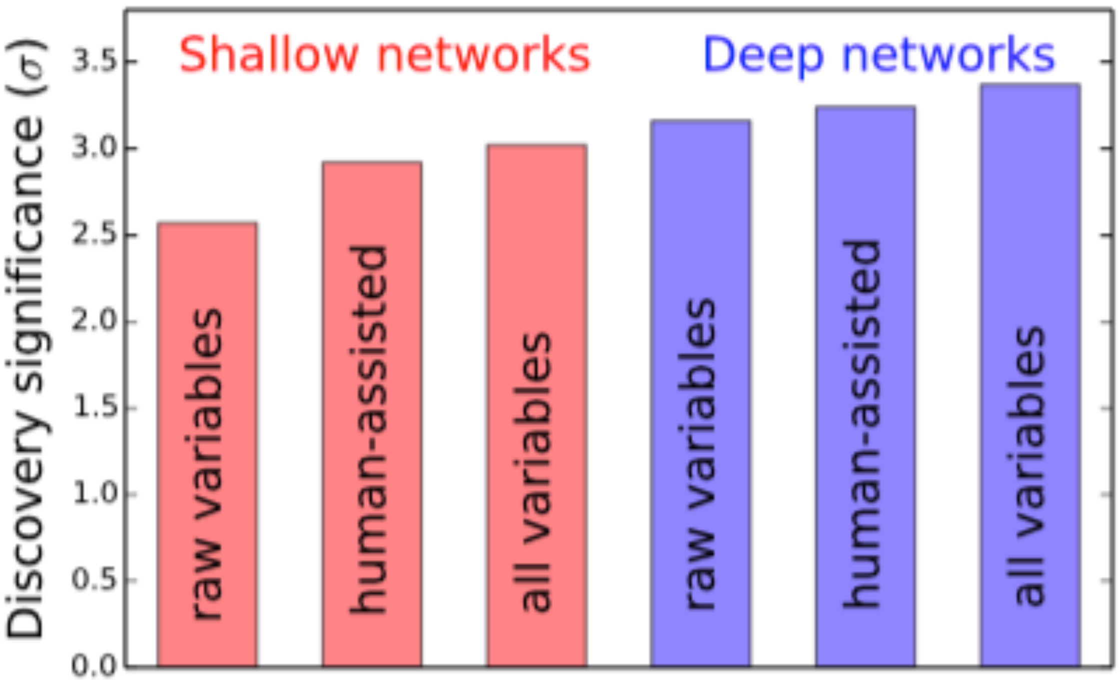
- MSSM at LHC : H^0➔WWbb vs tt➔WWbb
- Low level variables:
  - 4-momentum vector
- High level variables:
  - Pair-wise invariant masses
- Deep NN outperforms NN, and does not need high level variables
- DNN learns the physics ?



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
- ~10M events used for training (>>10* full G4 simulation in ATLAS)

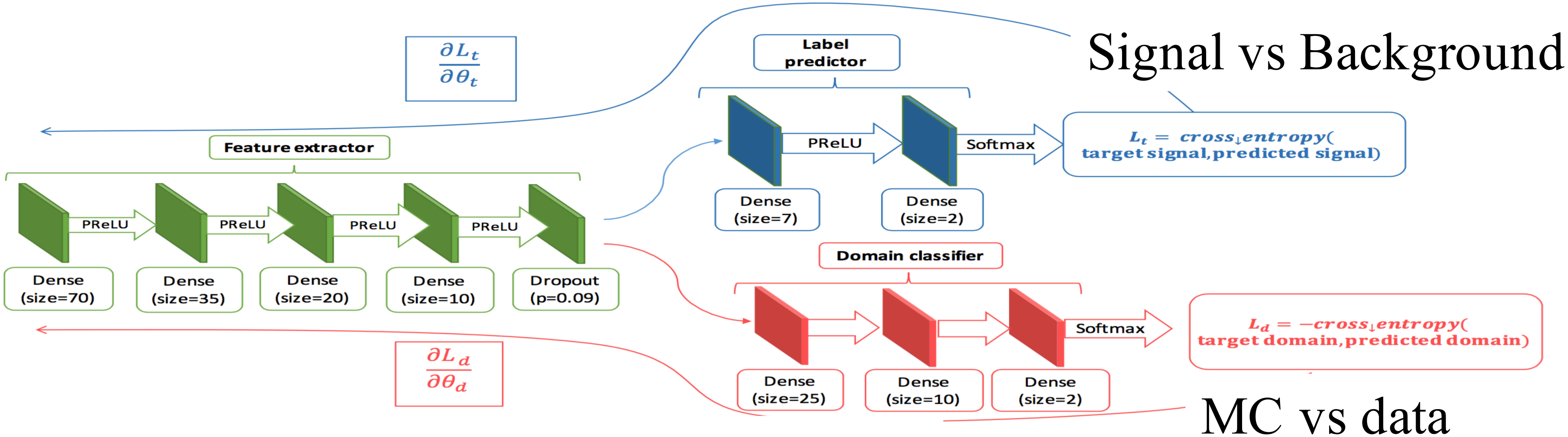
Systematics-aware training

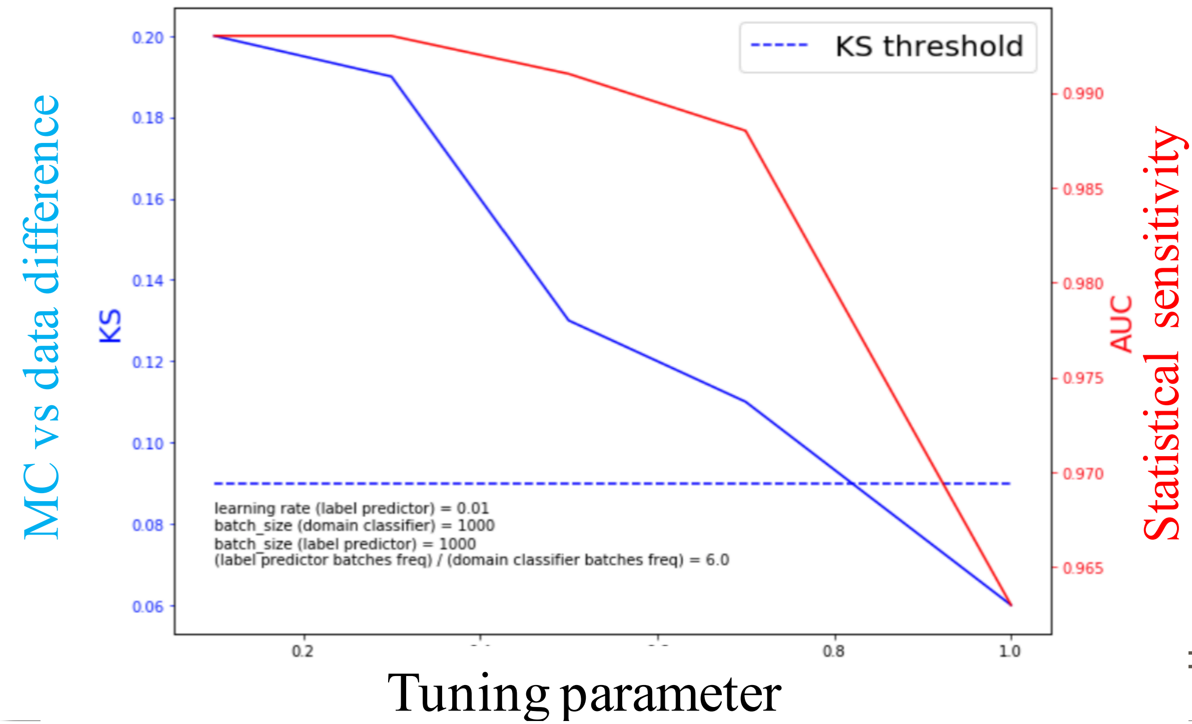
- Our experimental measurement papers typically ends with
  - measurement = m $\pm$ σ(stat) $\pm$ σ(syst)
  - σ(syst) systematic uncertainty : known unknowns, unknown unknowns…
- Name of the game is to minimize quadratic sum of :

  σ(stat) $\pm\sigma$(syst)
- ML techniques used so far to minimise σ(stat)
- Impact of ML on σ(syst) or even better global optimisation of σ(stat) $\pm$ σ(syst) is an open problem
- Worrying about σ(syst) untypical of ML in industry
- However, a hot topic in ML in industry: *transfer learning*
- E.g. : train image labelling on a image dataset, apply on new images (different luminosity, focus, angle etc…)
- For HEP : we train with Signal and Background which are not the real one (MC, control regions, etc...)➔source of systematics

Syst Aware Training: adversarial

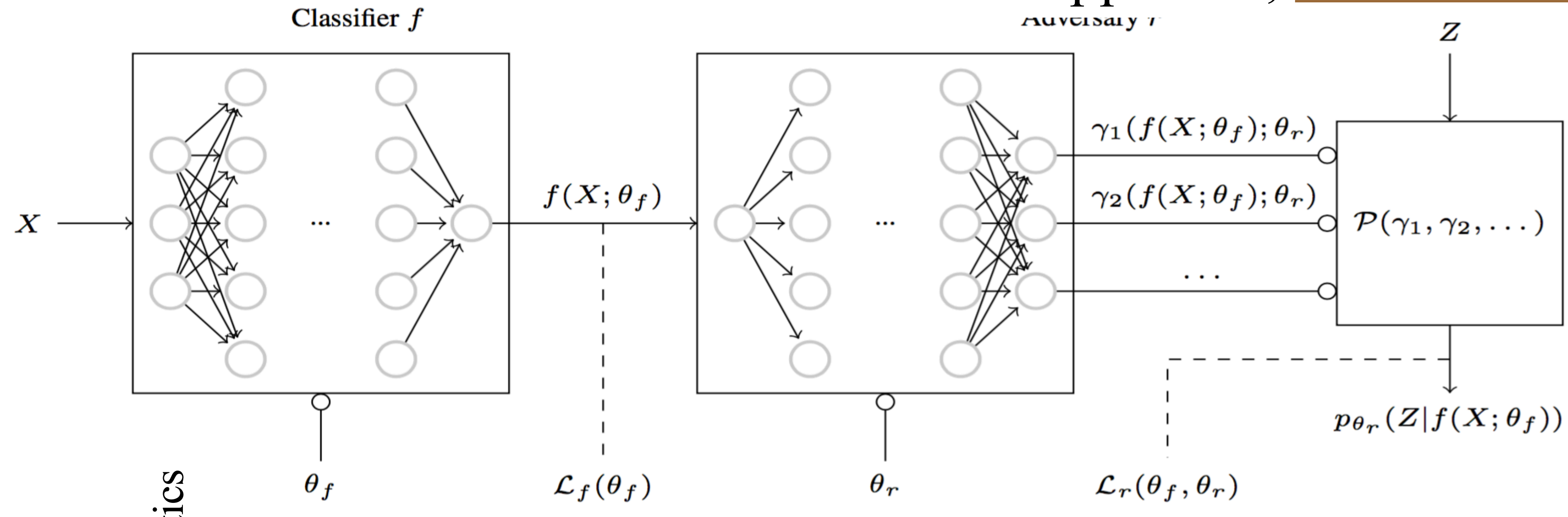
Inspired from 1505.07818 Ganin et al : ACAT 2017 Ryzhikov and Ustyuzhanin

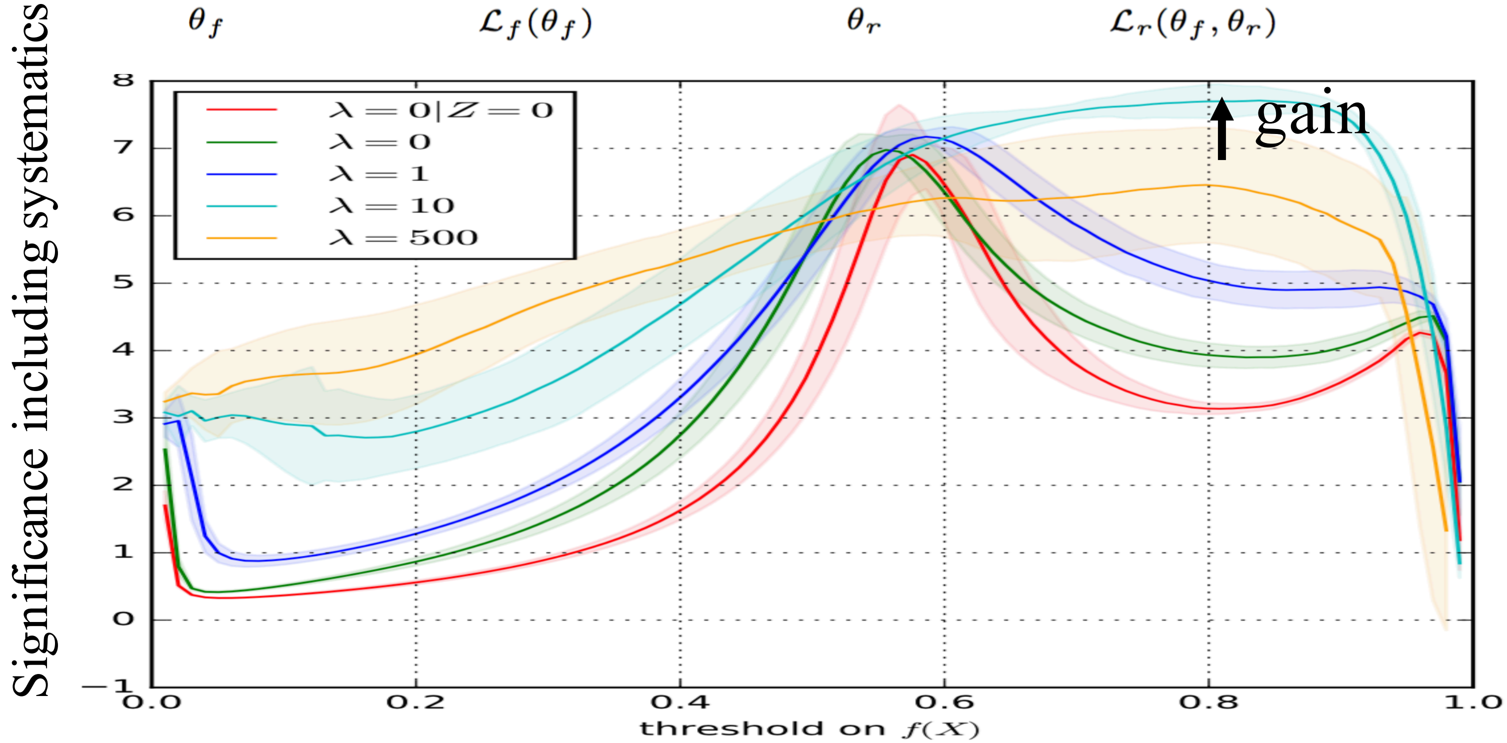




Syst Aware training: pivot

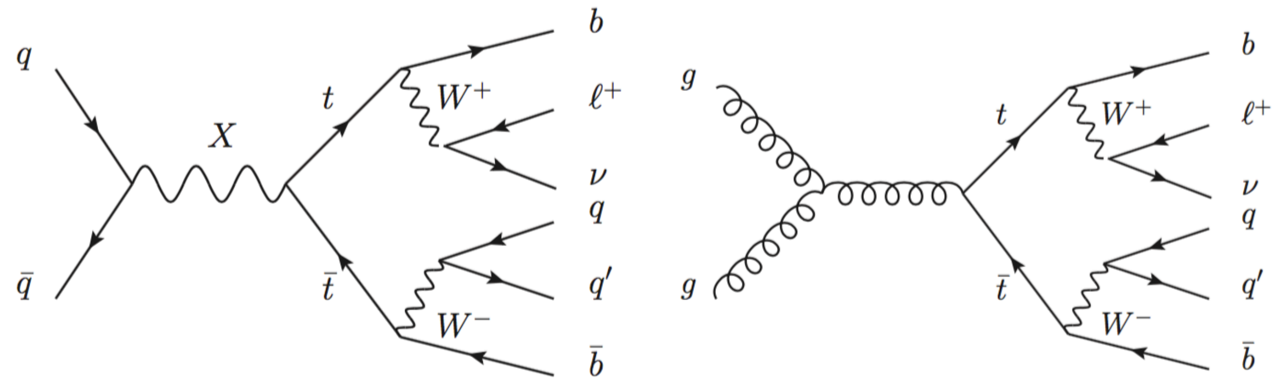
Louppe et al, 1611.01046

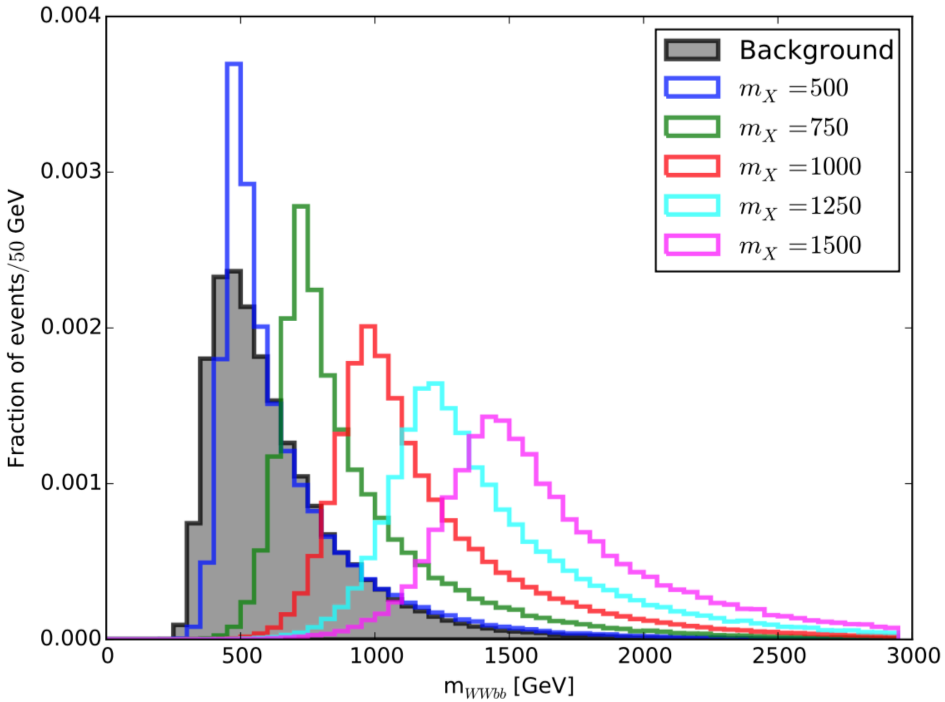




Parameterised learning

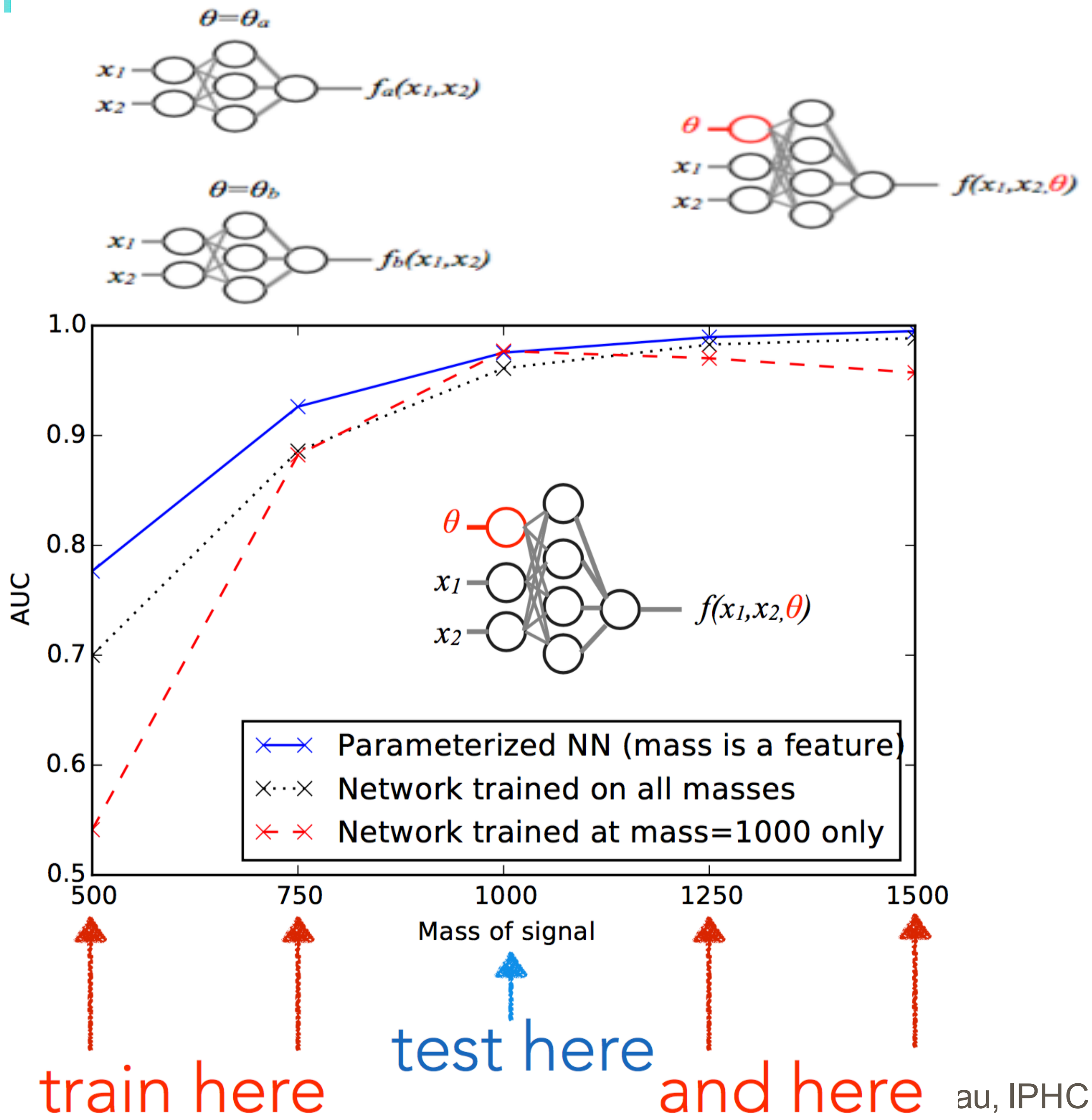
1601.07913 Baldi, Cranmer, Faucett, Sadowksi, Whiteson





- Typical case: looking for a particle of unknown mass
- E.g. here tt decay

Parameterised learning (2)



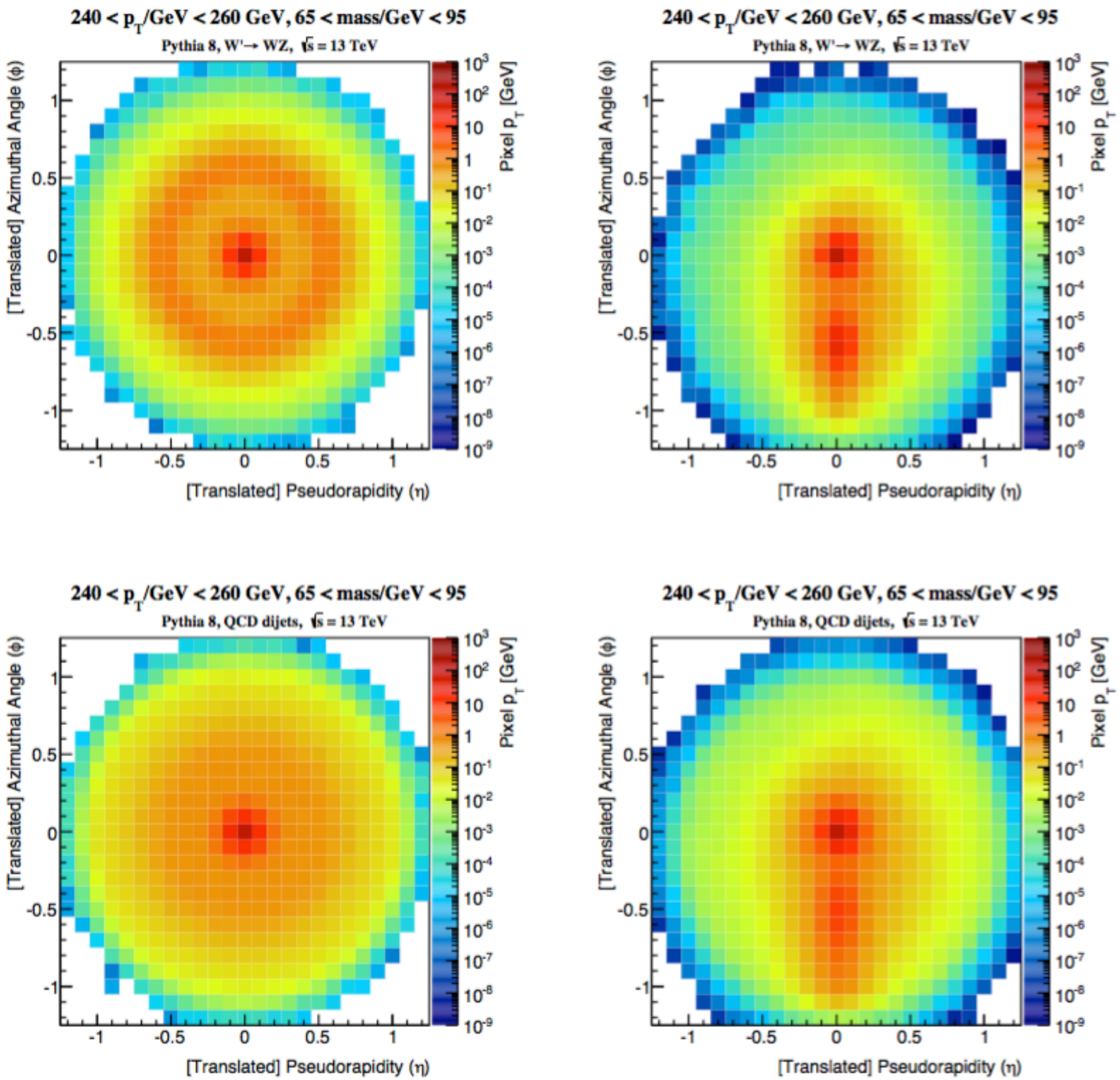
- Train on 28 features plus true mass
- Parameterised NN as good as single mass training
- ➔clean interpolation
- (mass just an example)
- Very recently used by CMS bblνl ν search https://arxiv.org/pdf/1708.04188.pdf

ML in reconstruction

Jet Images

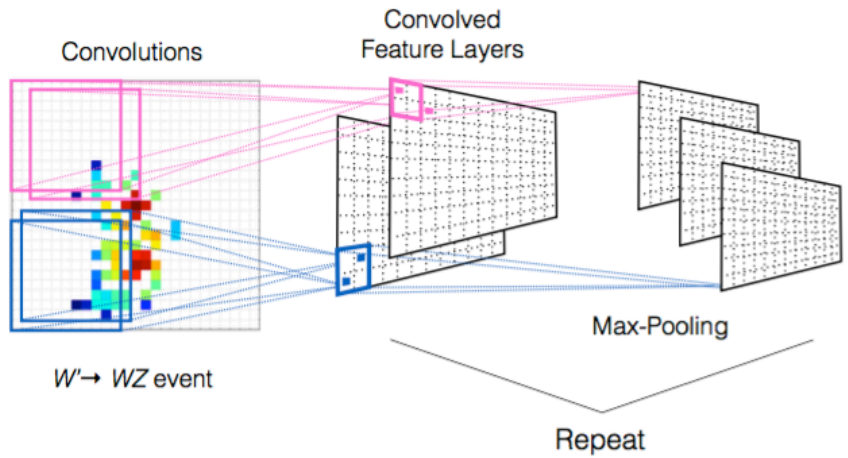
arXiv 1511.05190 de Oliveira, Kagan, Mackey, Nachman, Schwartzman

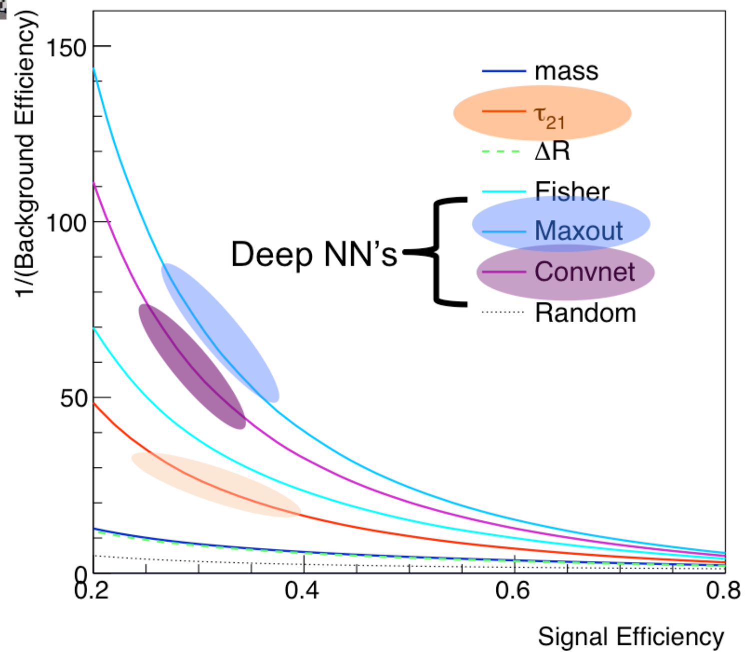
- Distinguish boosted W jets from QCD
- Particle level simulation
- Average images:



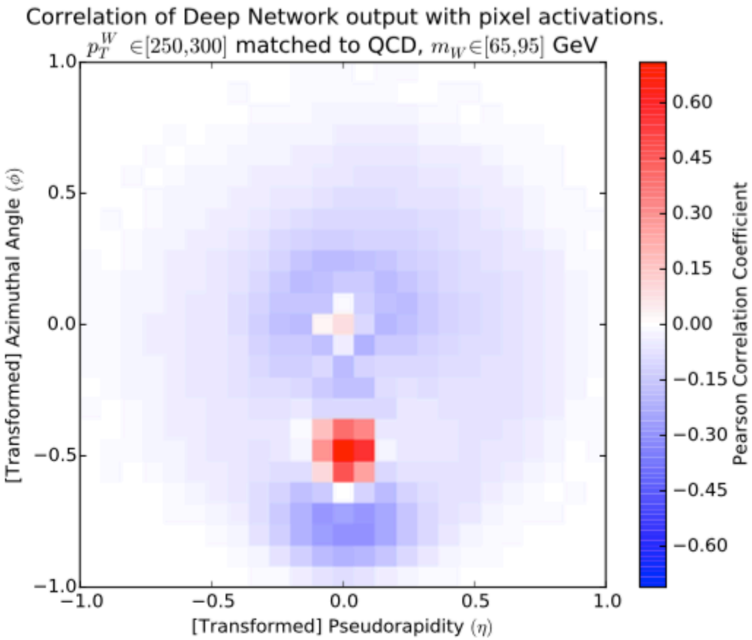
Jet Images : Convolution NN

arXiv:1511.05190



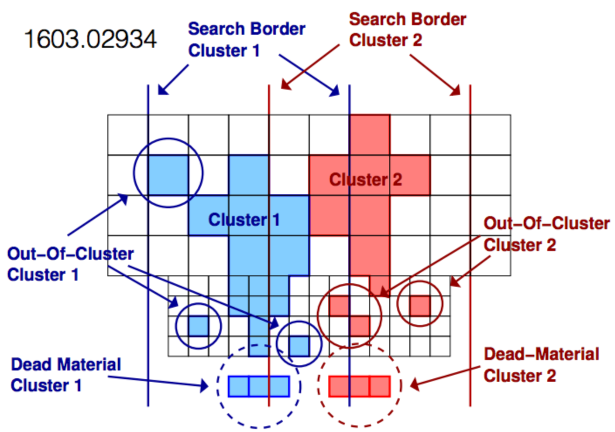


- Variables build from CNN outperform the more usual ones

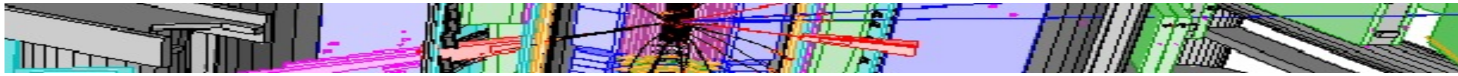


- What the CNN sees (the "cat" neurone")
- Now need proper detector and pileup simulation
- ➔3Dimension

1603.02934

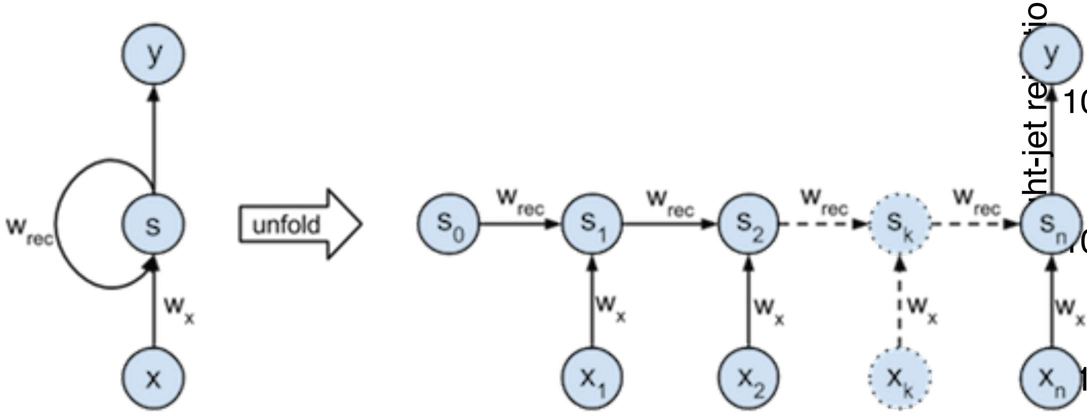


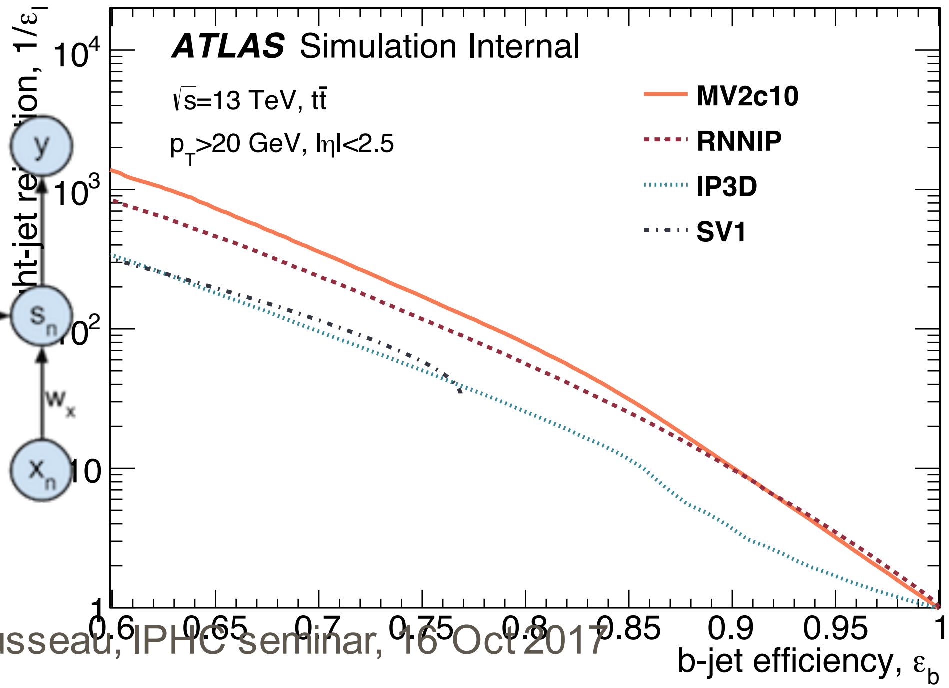
RNN for b tagging



ATL-PHYS-**PUB**-2017-003

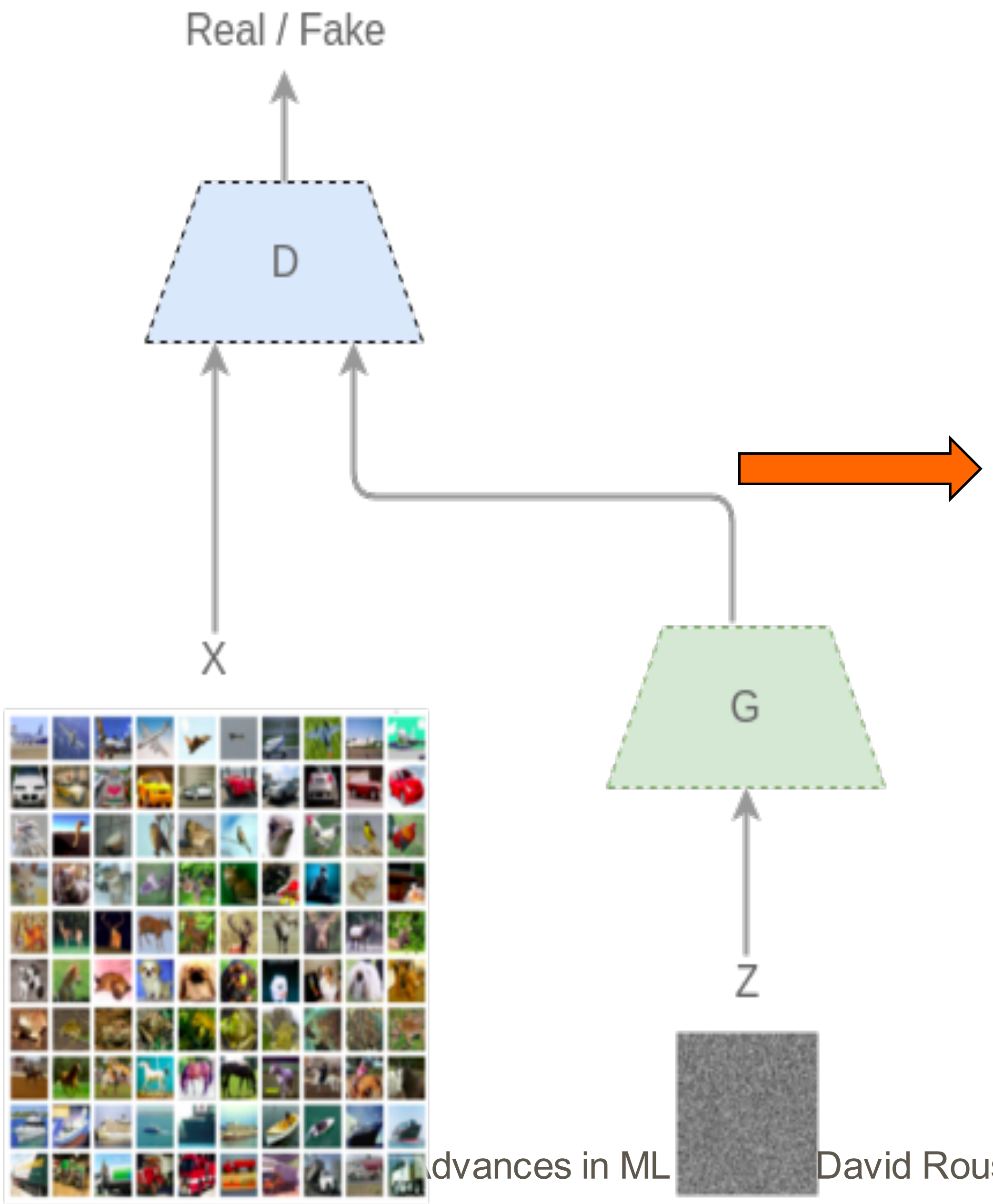
- BDT and usual NN expect a fix number of input. What to do when the number of inputs is not fixed like the tracks for b-quark jet tagging ?
- Recurrent neural networks have seen outstanding performance for processing sequence data
  - Take data at several "time-steps", and use previous time-step information 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 in other places!
  - Combining tracks with clusters? Track to vertex ma

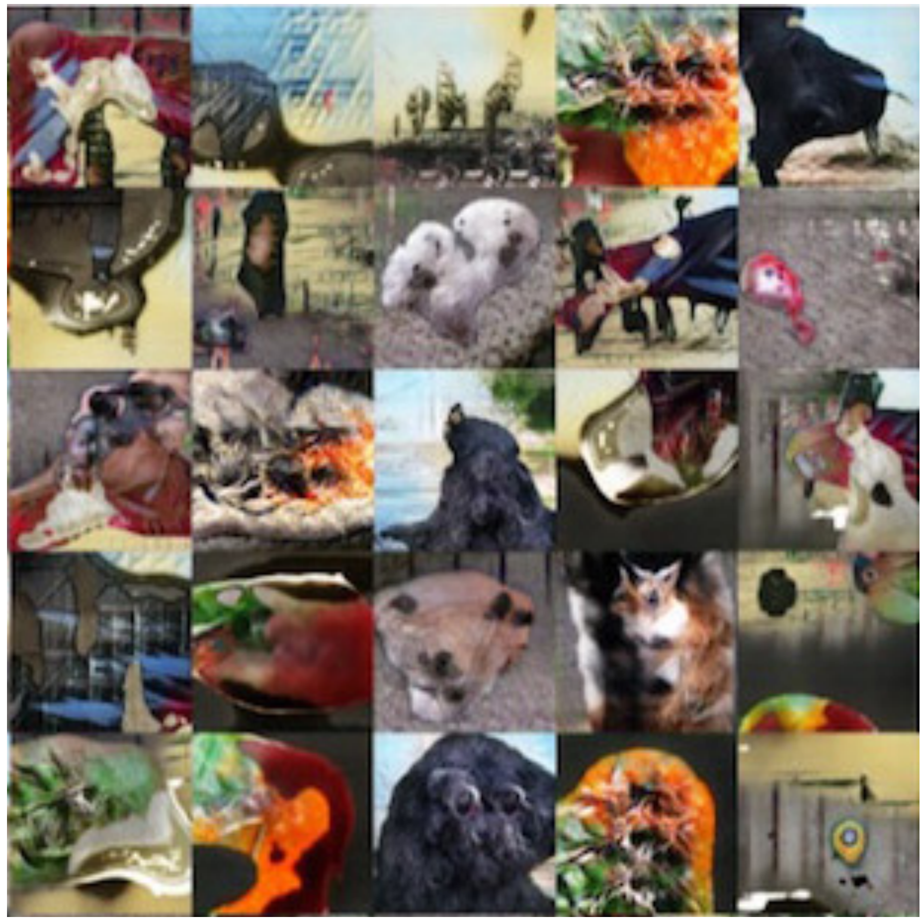




ML in simulation

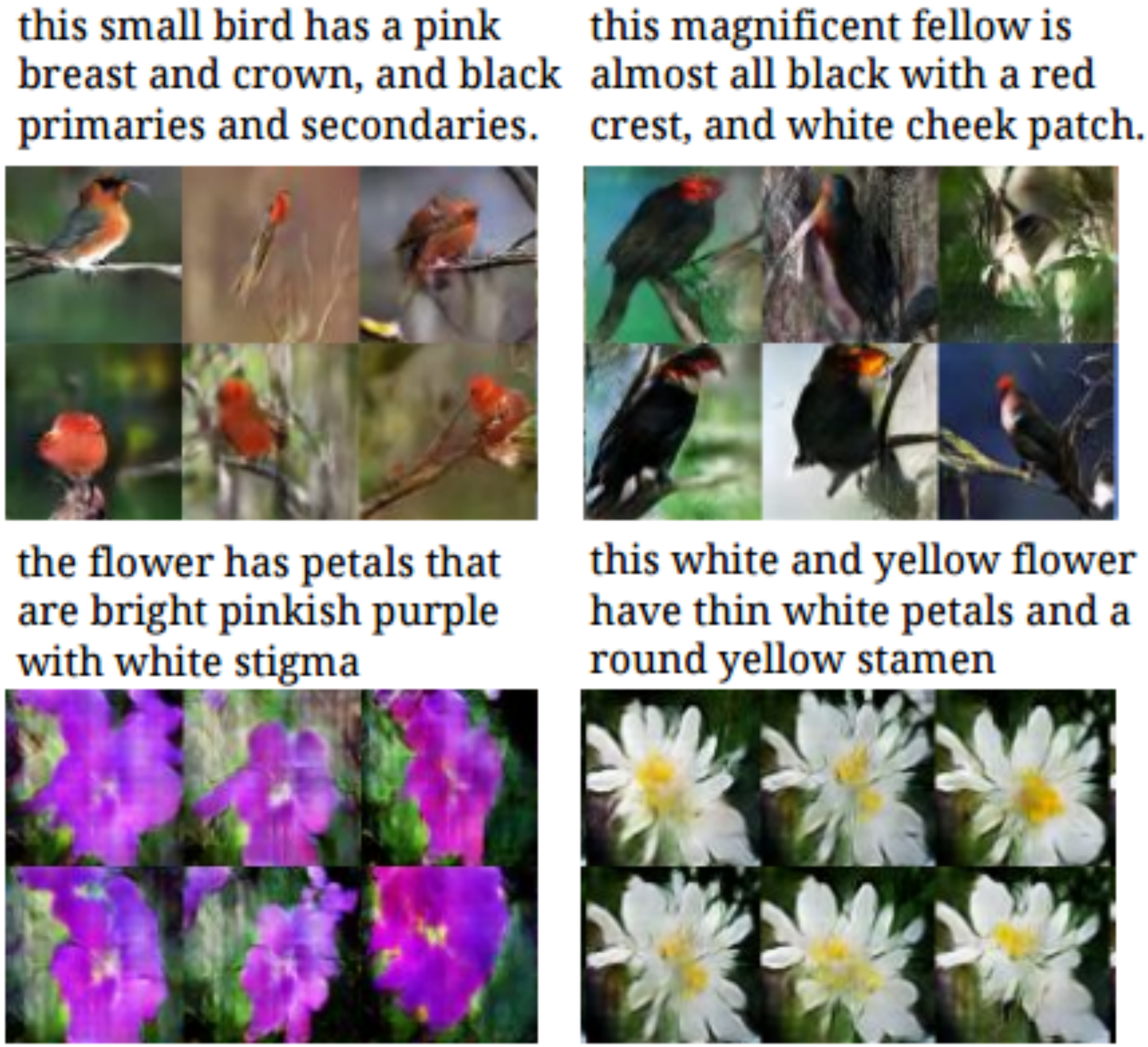
Generative Adversarial Network



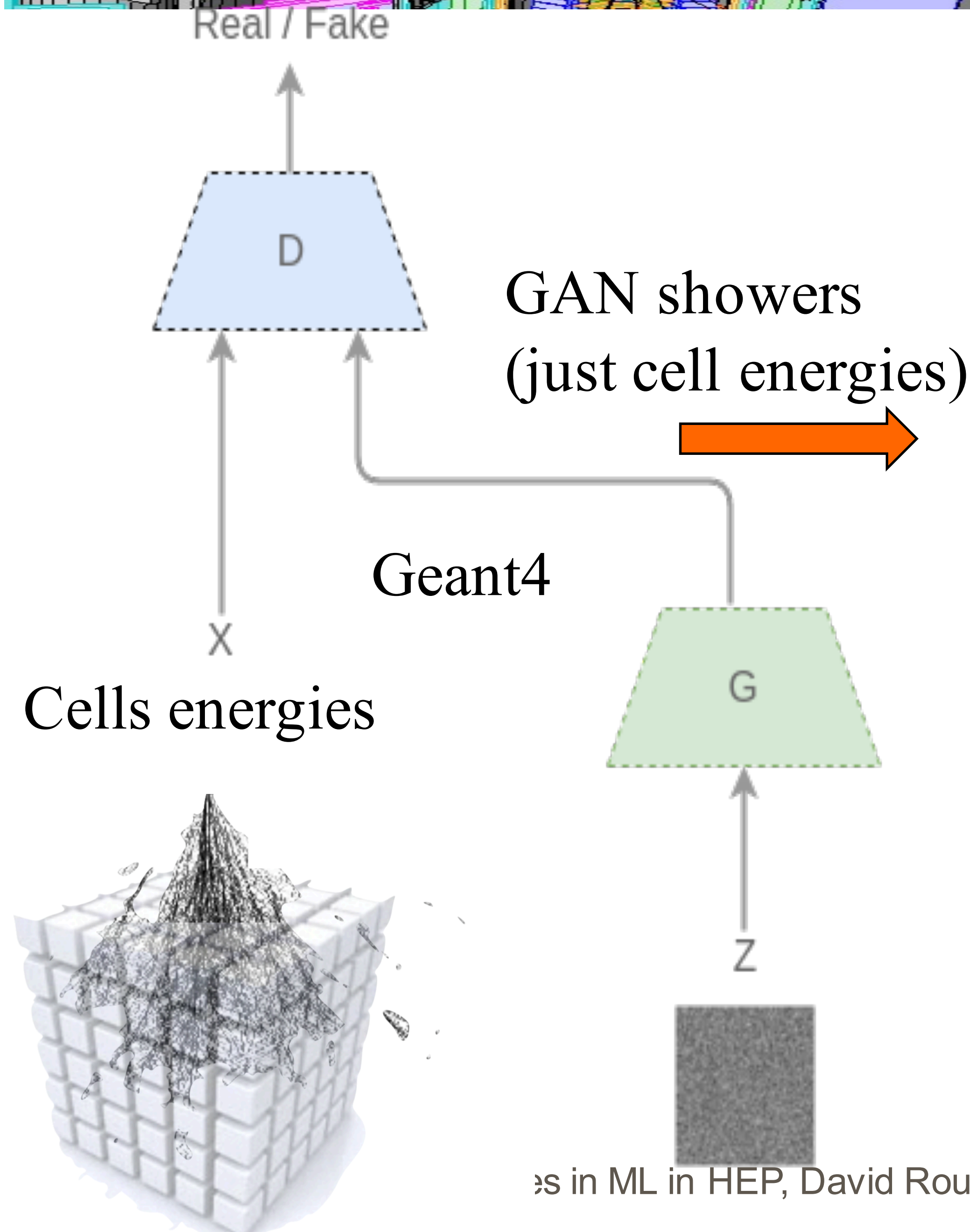


Condition GAN

Text to image



GAN for simulation

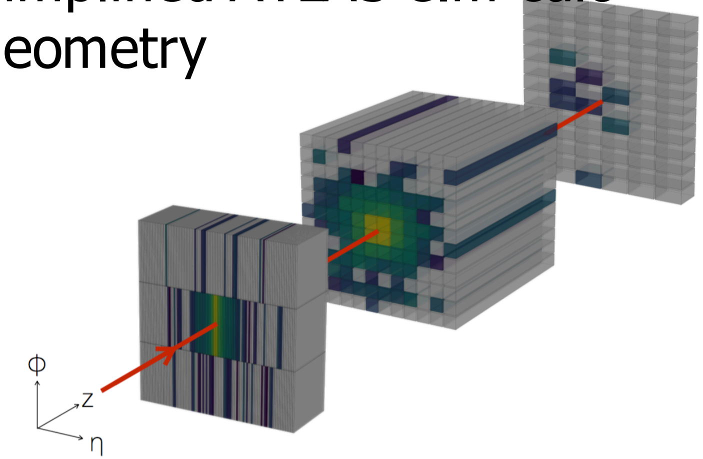


- Half of LHC grid computers (~300.000 cores) are crunching Geant4 simulation 24/24 365/365
- …while LHC experiments are collecting more and more events
- ➔reducing CPU consumption of simulation is very important

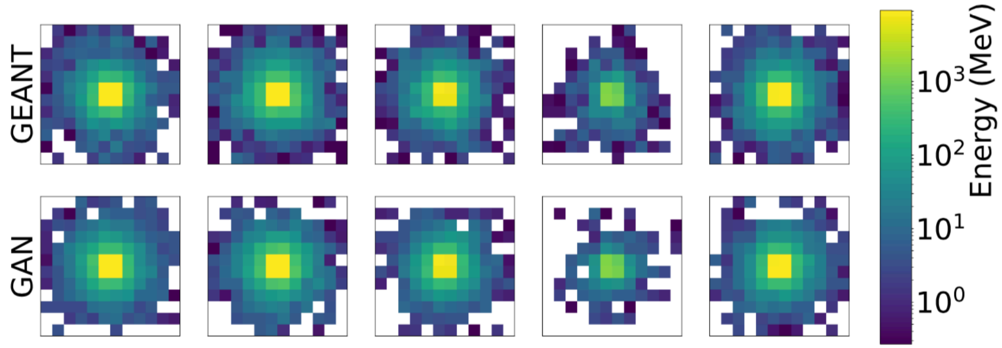
- Imagine training a GAN on single particle showers of all types and energies
- Then when an event is simulated it would ask for GAN showers on request (superfast by 3-4 order of magnitude)
- Would replace current fast simulation, frozen shower libraries….
- Just an idea until recently, but see https://arxiv.org/abs/1705.02355 ,also GeantV team is looking into this
- If/when it works, would require large GPU clusters

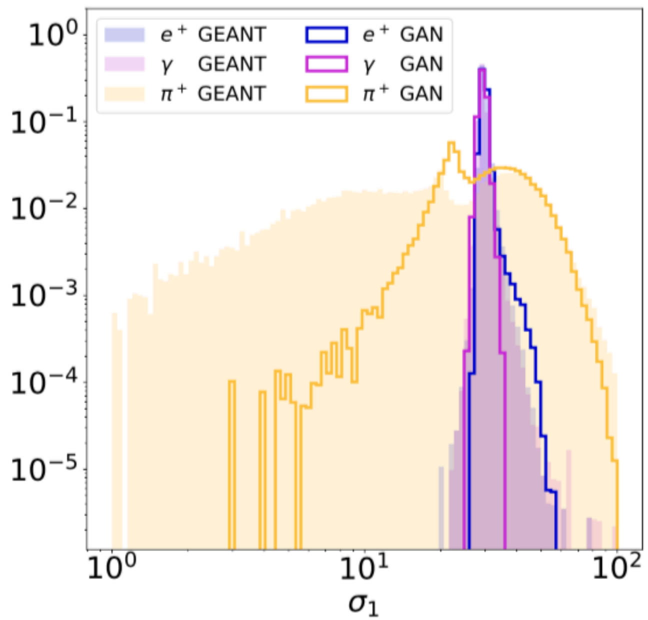
CaloGAN

Simplified ATLAS e.m calo geometry



[Paganini et al.](#)



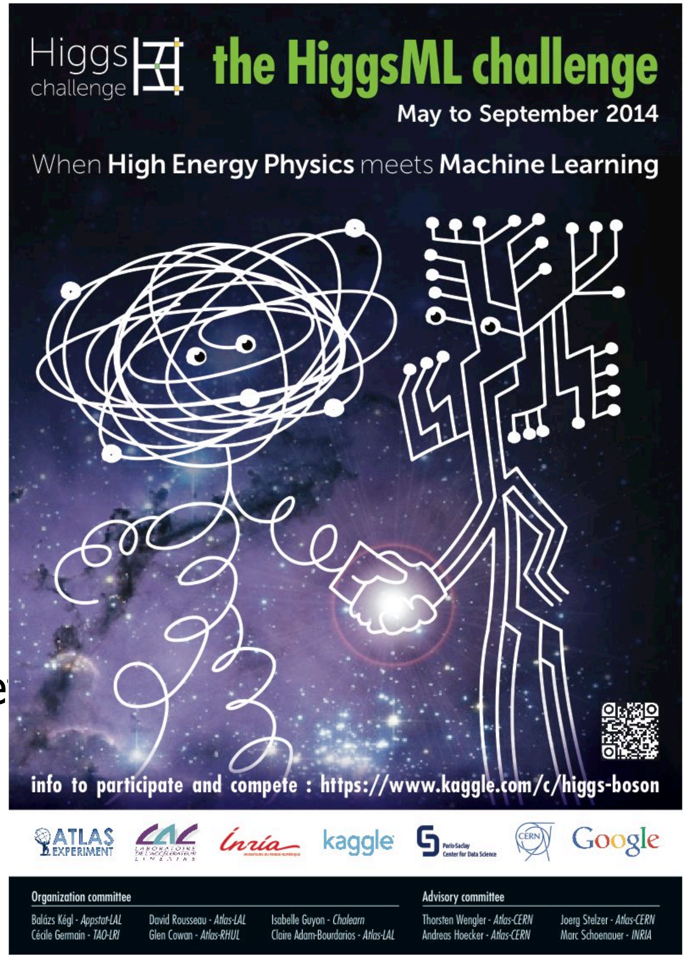


- σ_1:width in Middle layer
- One of many physics variable examined
- Pion more difficult
- ➔very promising

Data Challenges

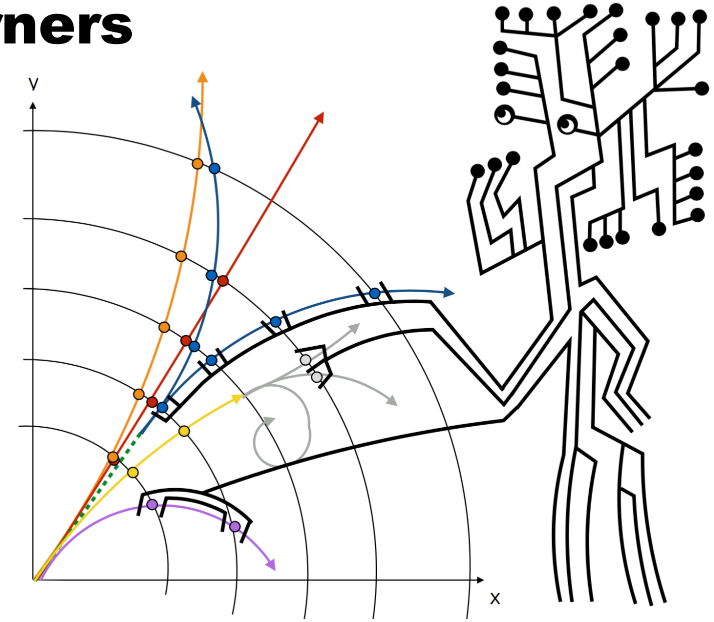
Higgs Machine learning challenge

- See talk DR CTD2015 Berkeley
- An ATLAS Higgs signal vs background classification problem, optimising statistical significance
- Ran in summer 2014
- 2000 participants (largest on Kaggle at that time)
- Outcome
  - Best significance 20% than with Root-TMVA
  - BDT algorithm of choice in this case where number variables and number of training events limited (NN very slightly better but much more difficult to tune)
  - XGBoost written for HiggsML, now best BDT on the marke
  - Wealth of ideas, documented in JMLR proceedings v42
  - Still working on what works in real life what does not
  - Raised awareness about ML in HEP
- Also:
  - Winner Gabor Melis hired by DeepMind
  - Tong He, co-developper of XGBoost, winner of special "HEP meets ML" price got a PhD grant and US visa



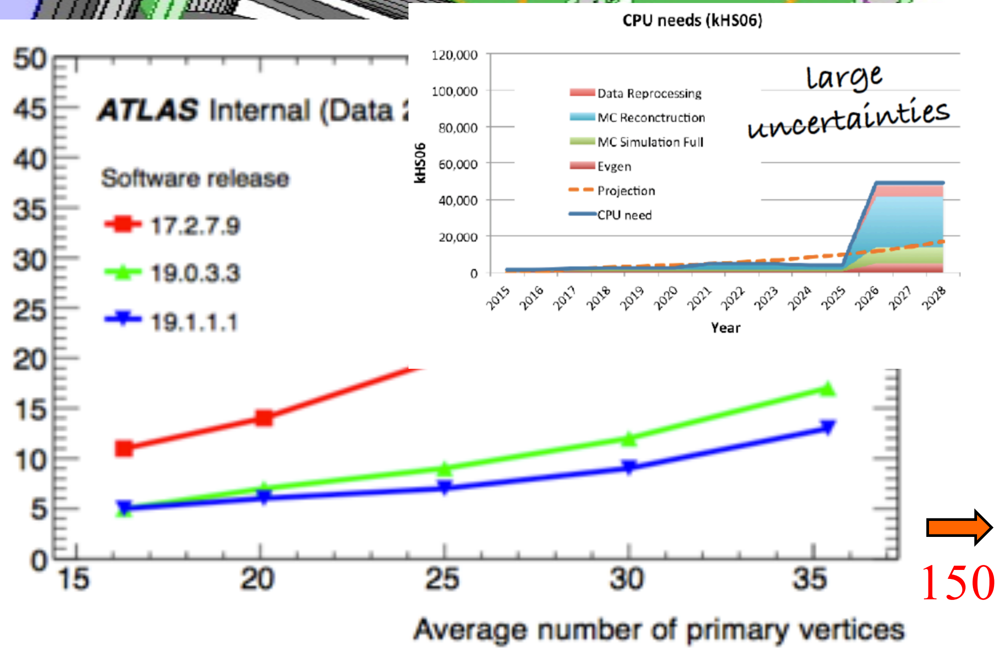
Towards a Future Tracking Machine Learning challenge

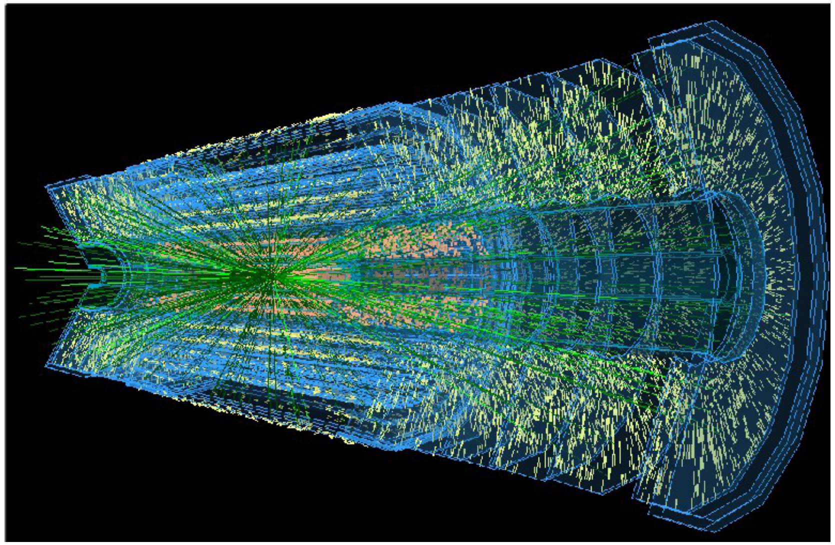
A collaboration between ATLAS and CMS physicists, and Machine Learners



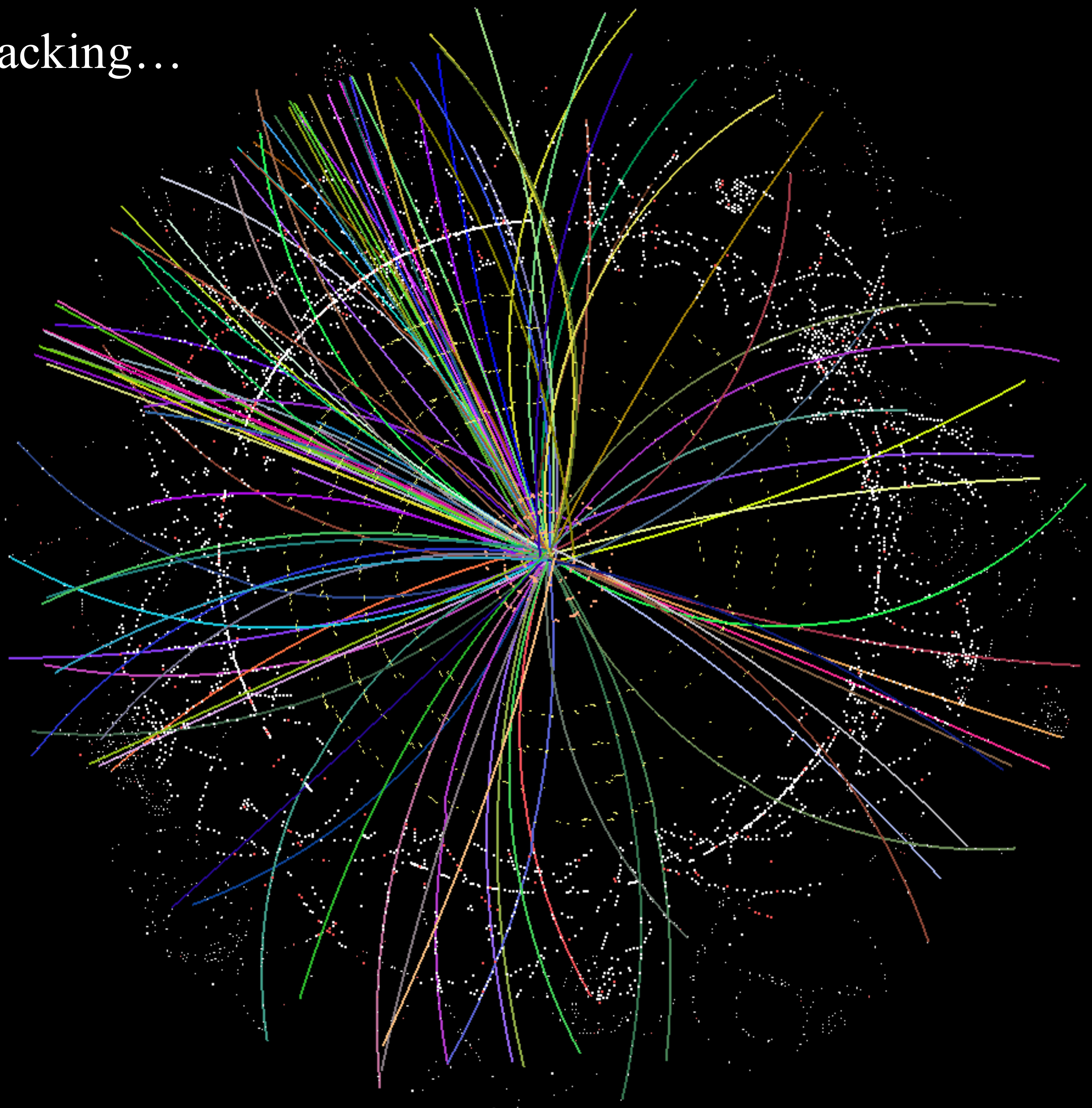
TrackML : Motivation

- See details DR talk at CTD/WIT 2017
- 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)

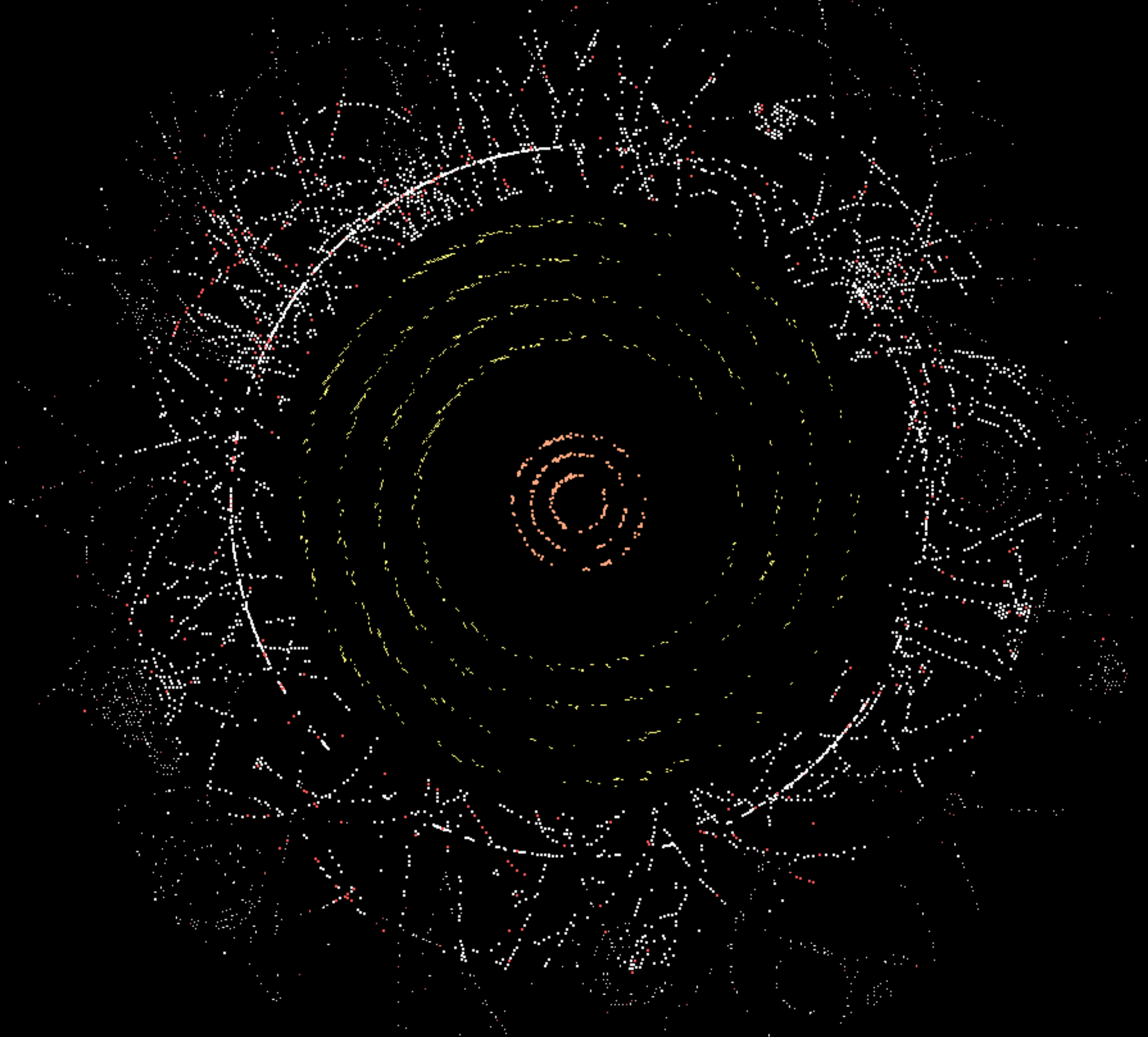




HEP tracking…



…fascinates ML experts

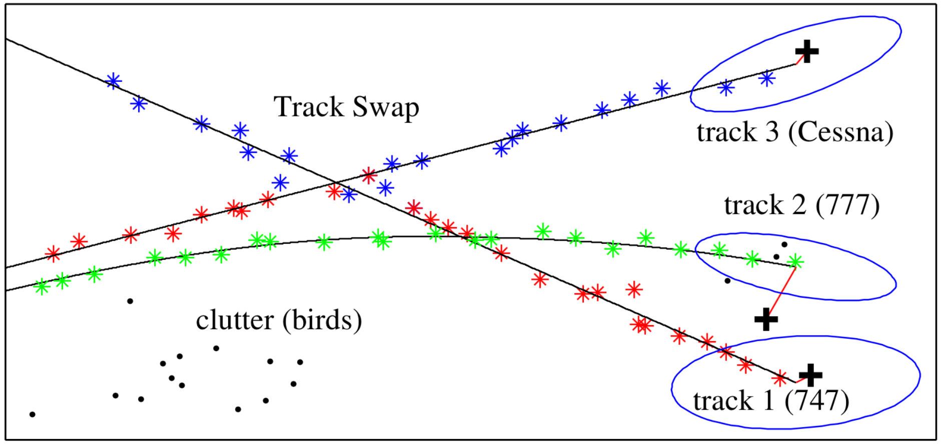


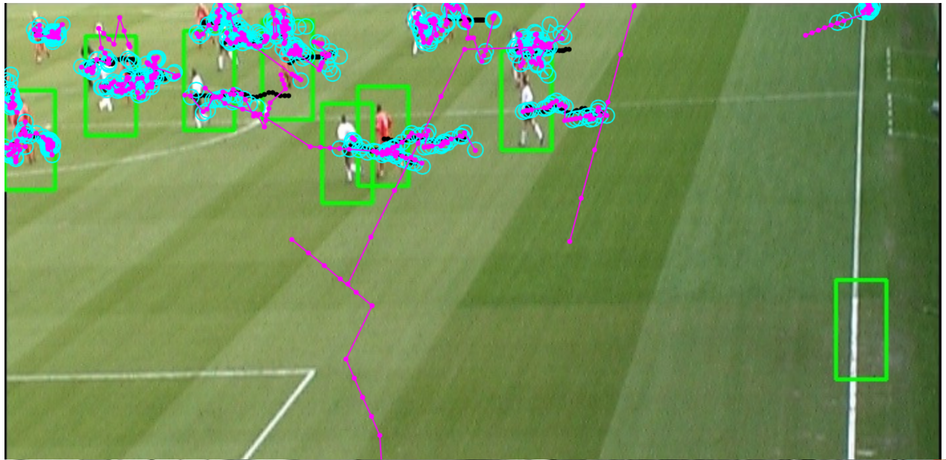
TrackML : engaging Machine Learners

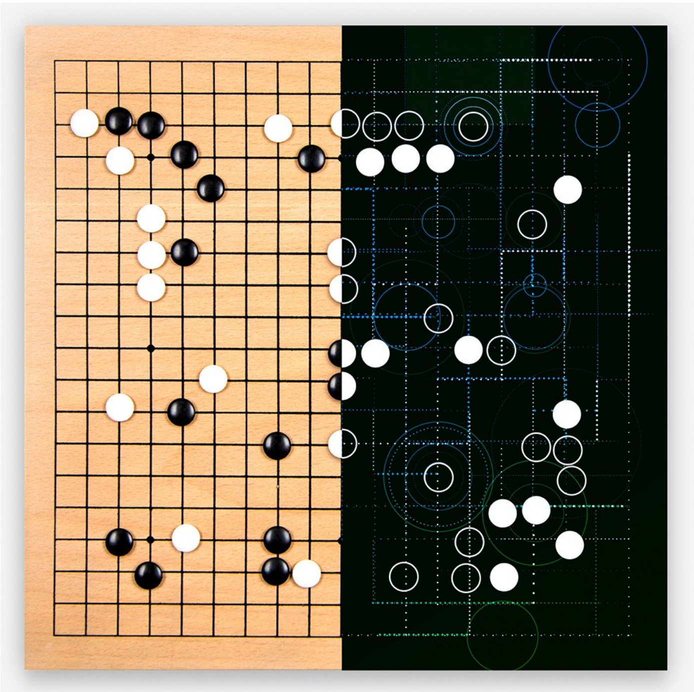
- Suppose we want to improve the tracking of our experiment
- We read the literature, go to workshops, hear/read about an interesting technique (e.g. ConvNets, MCTS…). Then:
  - Try to figure by ourself what can work, and start coding➔traditional way
  - Find an expert of the new technique, have regular coffee/beer, get confirmation that the new technique might work, and get implementation tips➔better
- …repeat with each technique...
- Much much better:
  - Release a data set, with a benchmark, and have the expert do the coding him/herself
  - ➔ he has the software and the know-how so he'll be (much) faster even if he does not know anything about our domain at the beginning
  - ➔engage multiple techniques and experts simultaneously (e.g. 2000 people participated to the Higgs Machine Learning challenge) in a comparable way
  - ➔even better if people can collaborate
  - ➔a challenge is a dataset with a benchmark and a buzz
  - Looking for long lasting collaborations beyond the challenge
- Focus on the pattern recognition : release list of 3D points, challenge is to associate them into tracks fast. Use public release of ATLAS tracking (ACTS) as a simulation engine and starting kit

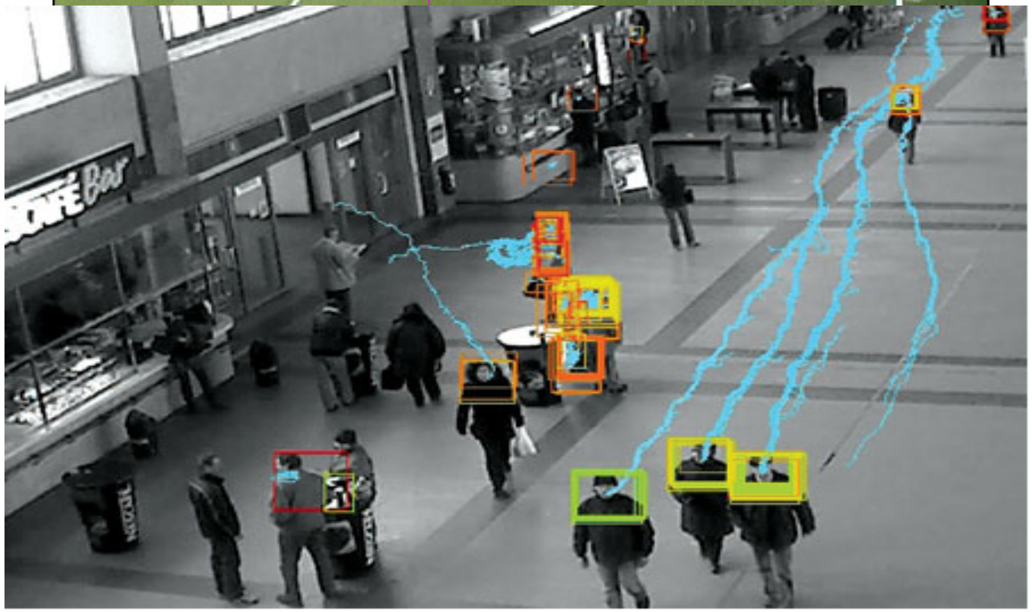
Pattern recognition

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



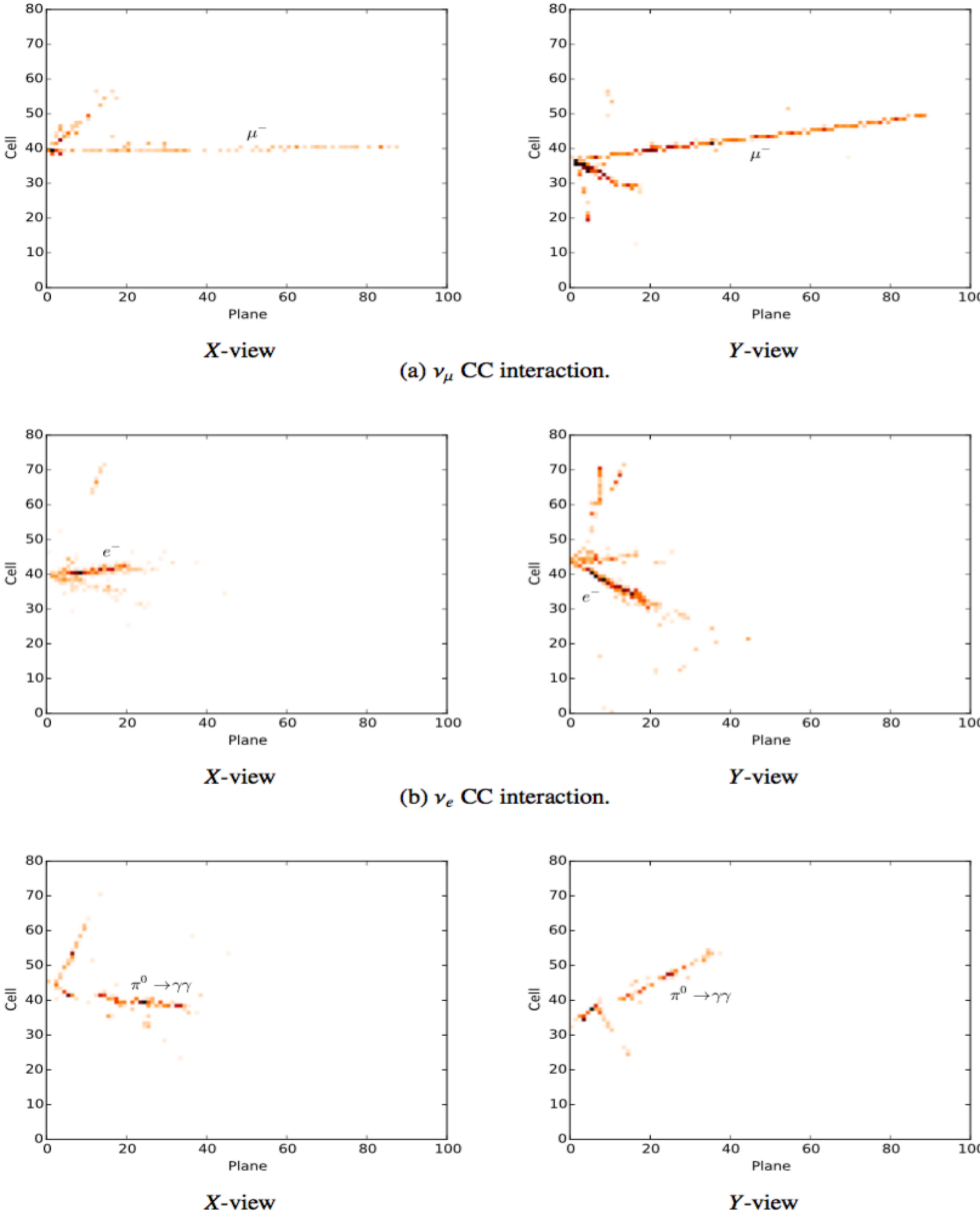






A recent attempt : NOVA

arXiv 1604.01444 Aurisano et al

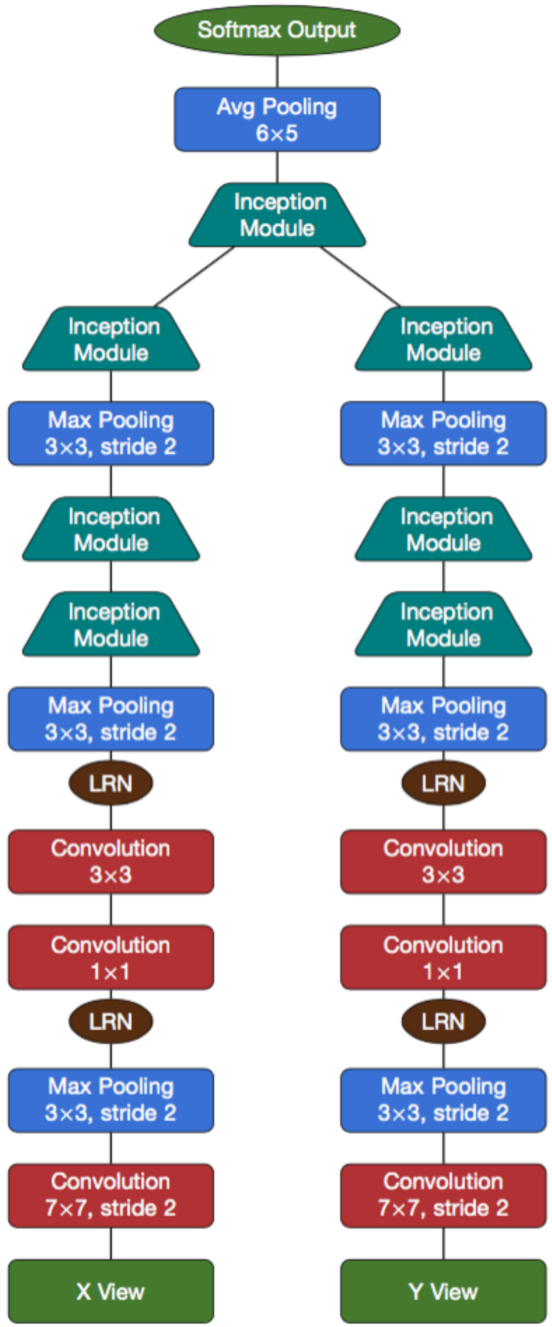


(a) ν_μ CC interaction.

(b) ν_e CC interaction.

(c) NC interaction.

Neutrino interaction classification
Using Convolutionnal Neural Network (GoogleNet)
Actually used for analysis



CTDWIT 2017 2D tracking Hackathon

CTDWIT 6-9th March 2017 LAL-Orsay

- Very simplified 2D simulation with HL-LHC ATLAS layout (circular detectors, multiple scattering, inefficiency, stopping tracks)

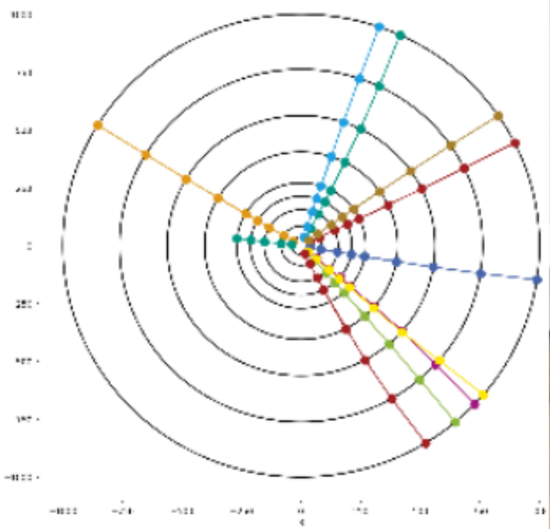
EPJ Web Conf., 150 (2017) 00015

- Run on RAMP platform
- 30 people (tracking experts mostly) for 2 hours in the same room, plus 36 hours till the end of the conference
- Winner is a Monte Carlo Tree Search algorithm (used in Go algorithms before and also by Alpha-Go)
- Runner-up a "real" ML algorithm : Long Short Term Memory

Belle II Experiment @belle2collab · 15 min

Congrats to four #Belle2 PhD students for winning the Tracking Challenge at this year's Connecting the DotsD Conference! #ctdwit #hackathon

À l'origine en anglais



1

David Rousseau
@dhpmrou

.@SteveAFarrell winner of #CTDWIT TrackMLRamp 2D #hackathon at @LALOrsay in the ML category. Congrats !

À l'origine en anglais

Wrapping-up

More on ML in HEP history

- Very first ML in HEP paper known
- ML for tracking and calo clustering
- B. Denby still active outside HEP: analysis of ultrasonic image of the tongue

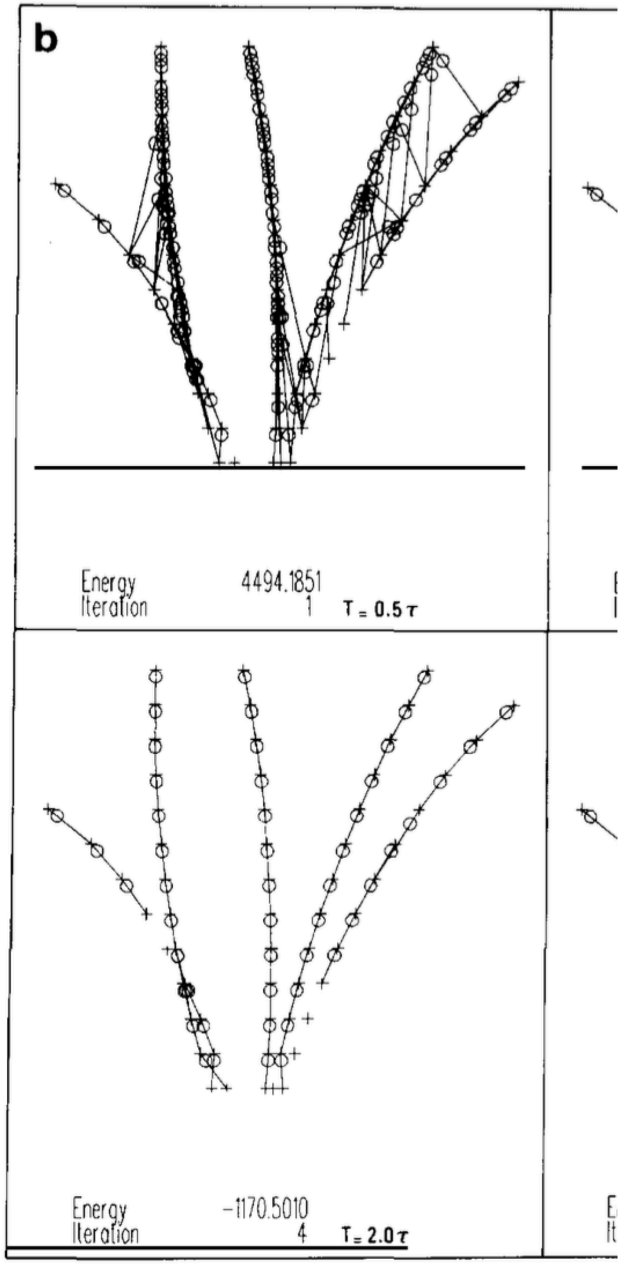
Computer Physics Communications 49 (1988) 429–448
North-Holland, Amsterdam

NEURAL NETWORKS AND CELLULAR AUTOMATA IN EXPERIMENTAL HIGH ENERGY PHYSICS

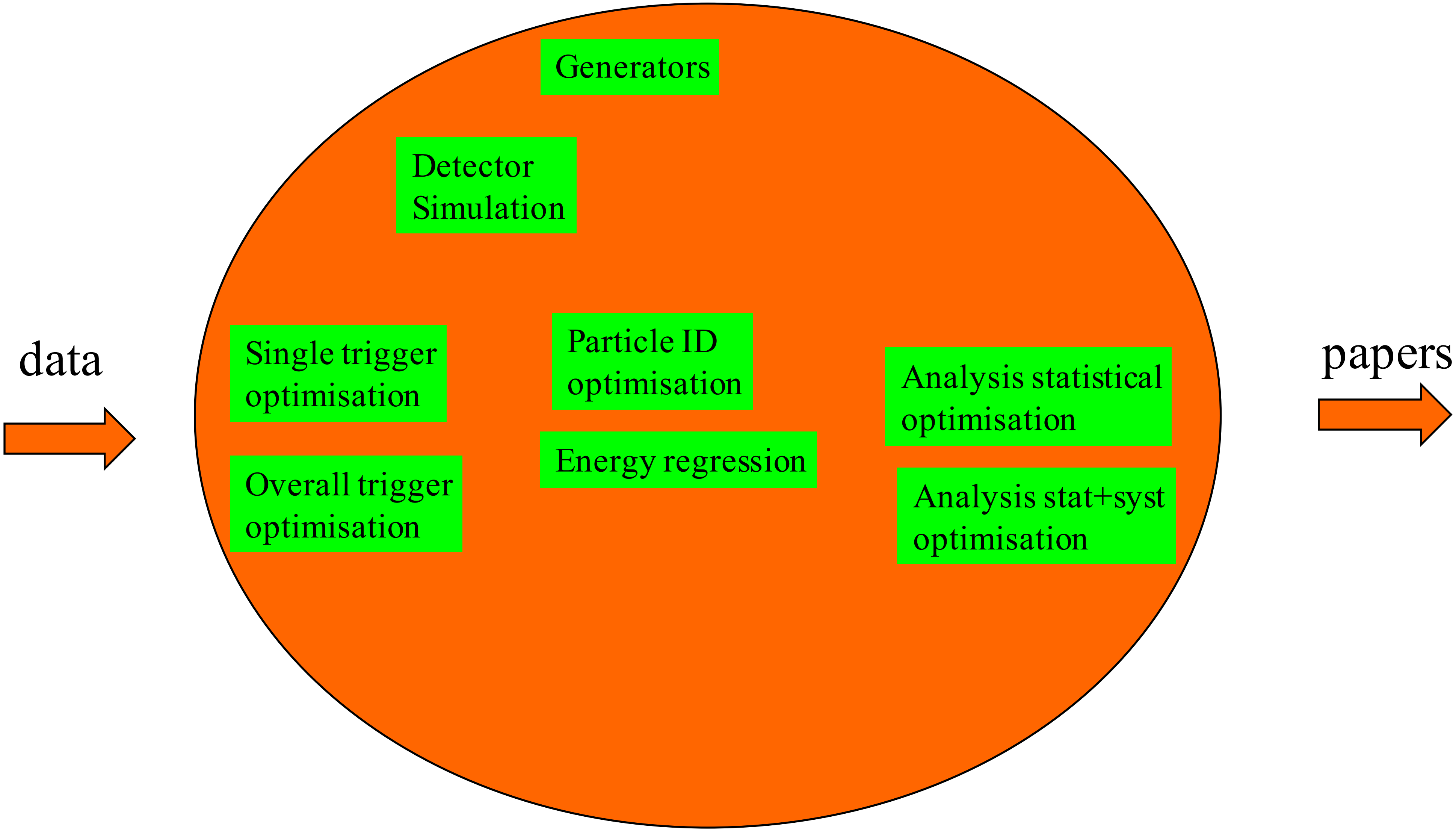
B. DENBY

Laboratoire de l'Accélérateur Linéaire, Orsay, France

Received 20 September 1987; in revised form 28 December 1987



ML playground



Collection of links

- In addition to workshops mentioned in the first transparencies, and references mentioned in the talks
- Interexperiment Machine Learning group (IML) is gathering speed (documentation, tutorials, etc…). Topical monthly meeting. Workshop 20-22 March :
- An internal ATLAS ML group has started in June 2016. In CMS in June 2017
- https://higgsml.lal.in2p3.fr
- http://opendata.cern.ch/collection/ATLAS-Higgs-Challenge-2014: permanent home of the challenge dataset
- NIPS 2014 workshop agenda and proceedings http://jmlr.org/proceedings/papers/v42/
- Mailing list opened to any one with an interest in both Data Science and High Energy Physics : HEP-data-science@googlegroups.com and lhc-machinelearning-wg@cern.ch
- IN2P3 project starting – http://listserv.in2p3.fr/cgi-bin/wa?A0=MACHINE-LEARNING-L open to anyone with some interest to ML (planning on 2 x 1day workshop per year)
- IN2P3 School of Statistics 28 May 1 June 2018 To be Confirmed (see SoS 2016)

ML Collaborations

- Many of the new ML techniques are complex→difficult for HEP physicists alone
- ML scientists (often) eager to collaborate with HEP physicists
  - prestige
  - new and interesting problems (which they can publish in ML proceedings)
- Takes time to learn common language
- Access to experiment internal data an issue, but there are ways out
- Note : Yandex Data School of Analysis (with ~10 ML scientists) now a bona fide institute of LHCb
- Very useful/essential to build HEP - ML collaborations : study on shared dataset, thesis (Computer Science or HEP)
- There is probably a friendly Machine Learner on your campus!

Open Data

- Public dataset are essential to collaborate (beyond talking over beer/coffee) on new ML techniques with ML experts (or even physicists in other experiments)
  - can share without experiments Non Disclosure policies
- Some collaborations built on just generator data (e.g. Pythia) or with simple detector simulation e.g. Delphes
  - good for a start, but inaccurate
- Effort to have better open simulation engine (e.g. Delphes 4-vector detector simulation, ACTS for tracking)
- UCI dataset repository has some HEP datasets
- Role of CERN Open Data portal:
  - We (ATLAS) initially saw its use for outreach purposes (CMS has been more open on releasing data)
  - But after all, ML collaboration is a kind of scientific outreach
  - ➔ATLAS uploaded there in 2015 the data from Higgs Machine Learning challenge (essentially 4-vectors from full G4 ATLAS simulation Higgs->tautau analysis)
  - ATLAS consider releasing more datasets dedicated to ML studies

Conclusion

- 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
- Some of these are ~easy, most are complex: open source software tools are ~easy to get, but still need (people) training, know-how
- More and more open datasets/simulators
- More and more HEP and ML workshops, forums, schools, challenges
- More and more direct collaboration between HEP researchers and ML researchers
- HEP will need more and more access to (GPU) training resources
- Never underestimate the time for :
  - (1) Great ML idea➔
  - (2) …demonstrated on toy dataset➔
  - (3) …demonstrated on real experiment analysis/dataset ➔
  - (4) …experiment publication using the great idea