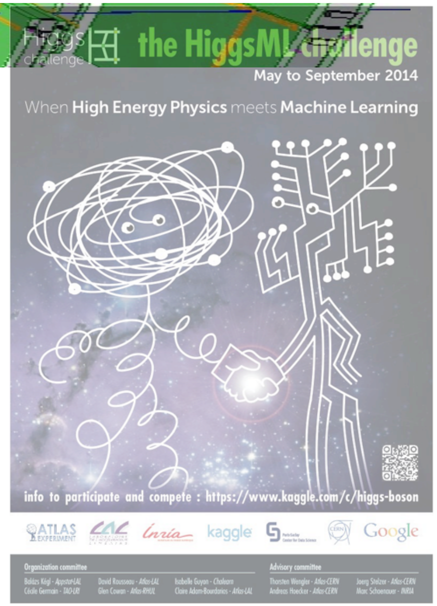
Machine Learning in ATLAS: activities and future challenges

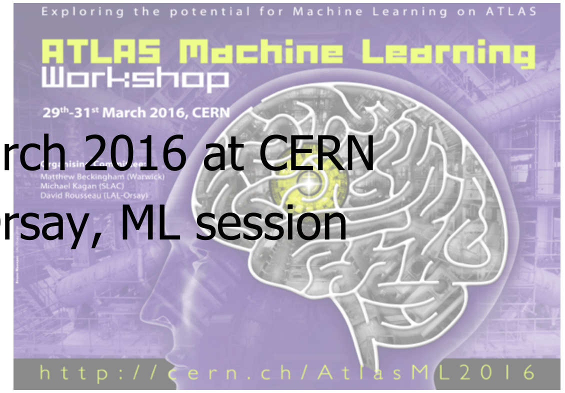
Matthew Beckingham, Michael Kagan, David Rousseau
for the ATLAS collaboration

OpenLab ML and Analytics workshop, 29th April 2016

ML events (with ATLAS participation)

- HiggsML Challenge, summer 2014
  - → HEP ML NIPS satellite workshop, December 2014
- Connecting The Dots, Berkeley, January 2015
- DS@LHC workshop, 9-13 November 2015
  - → future DS@HEP workshop
- LHC Interexperiment Machine Learning group
  - Started informally September 2015, gaining speed
- 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
- (internal) ATLAS Machine Learning workshop 29-31 March 2016 at CERN
- Hep Software Foundation workshop 2-4 May 2016 at Orsay, ML session
- TrackML Challenge, fall 2016

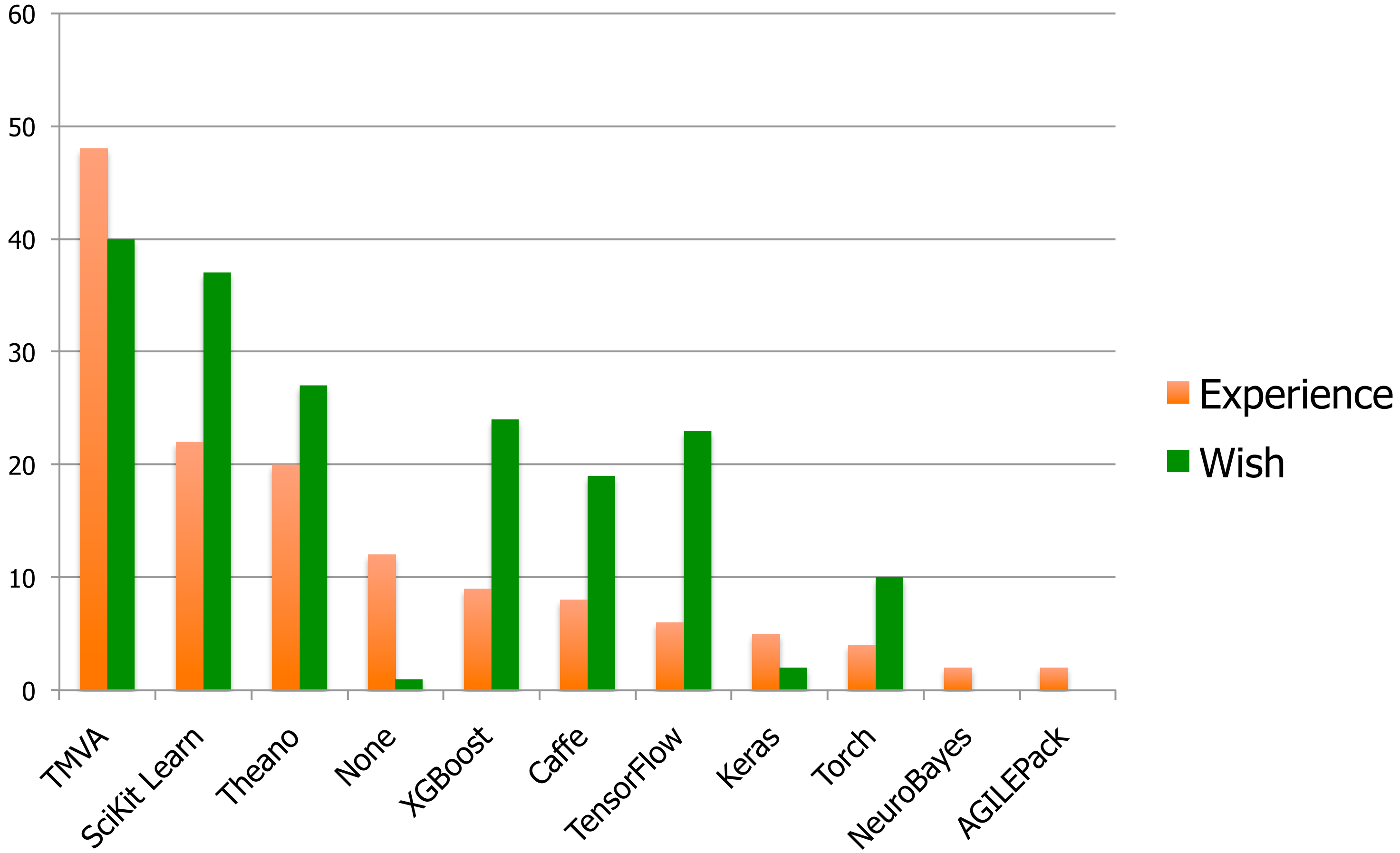




ML in Atlas

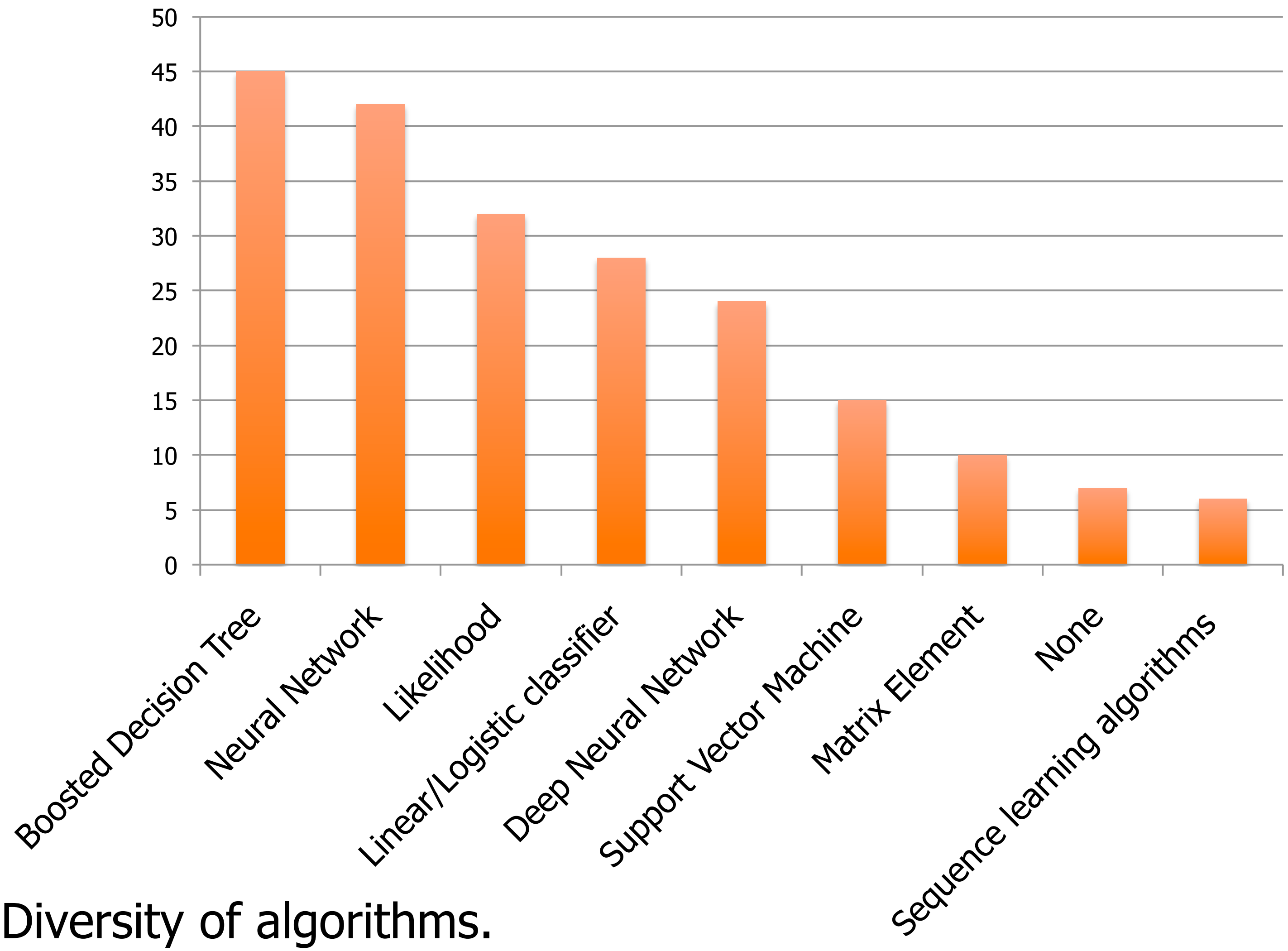
- Machine Learning (or rather Multi Variate Analysis as we used to call it) used almost since first data taking (2010) for reconstruction and analysis
- In most cases, Boosted Decision Tree with Root-TMVA, but recent explosion of usage and studies (see later)
- Recent Atlas ML workshop organised (by MB, MK, DR) to assess current usage, spot opportunities, and favour collaboration with ML experts
  - 200 participants
  - Survey of ML usage (see later)
  - Most of the material shown today gathered there (but limited to published material)
  - ATLAS ML forum being instantiated, will serve as a forum of discussions within ATLAS and with the outside world

ATLAS ML Survey 1



- Already experience beyond TMVA. Plan to use new tools.

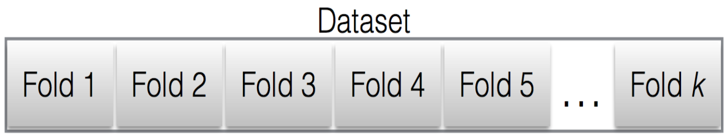
ATLAS ML Survey 2



- Diversity of algorithms.

Validation Techniques

- K-fold cross validation allows the estimation of the generalization error
  - Not overly dependent on the exact training / testing split
  - Average / RMS of k-fold errors gives estimate of true error rate
  - Very standard in the non HEP world. Little used in HEP. Being integrated in Root-TMVA, good!



- Split the dataset into k randomly sampled independent subsets (folds).
- Train classifier with k-1 folds and test with remaining fold.
- Repeat k times.

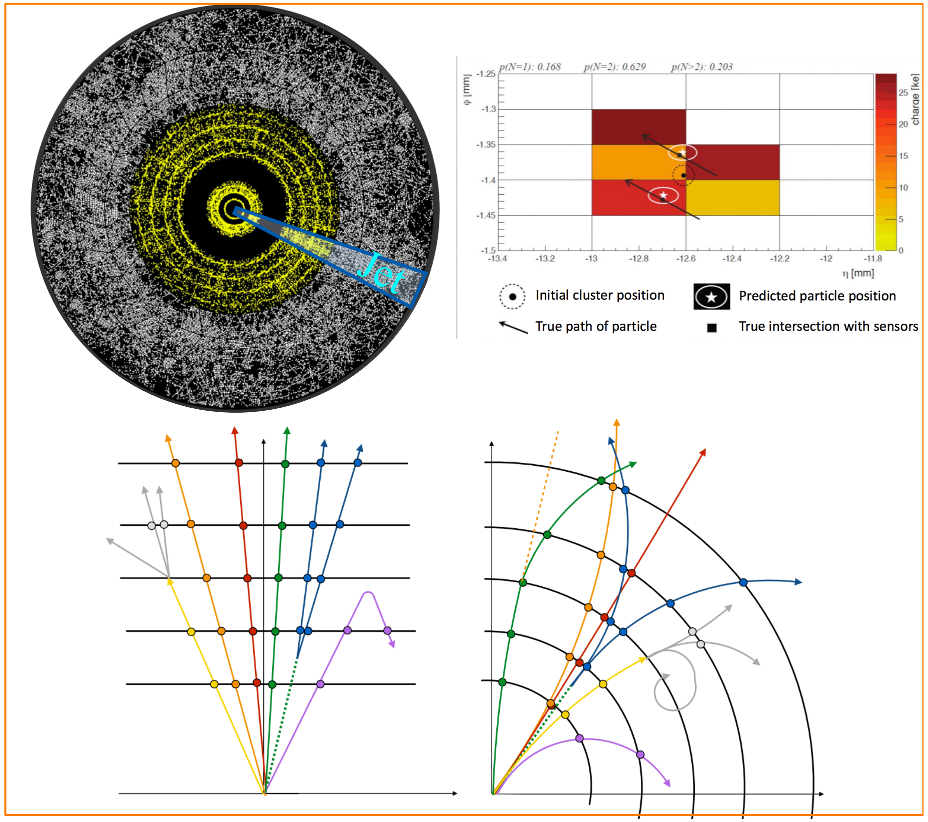
$$E = \frac{1}{k}\sum_{i=1}^{k} E_i.$$

Reconstruction

- Clear upcoming challenges as we approach HL-LHC

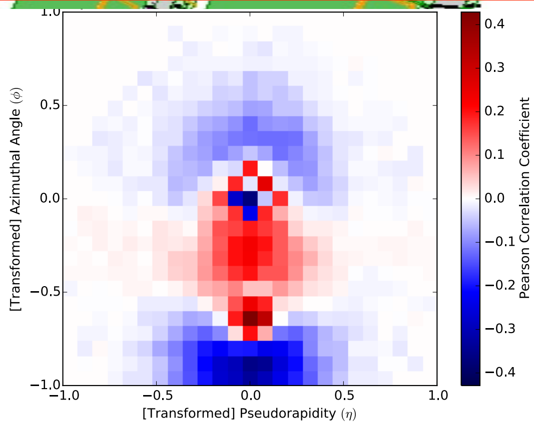
- Generally, making everything robust to increased pileup, and resource usage will be vital
  - New techniques needed

  (e.g. TrackML challenge, end of this talk)

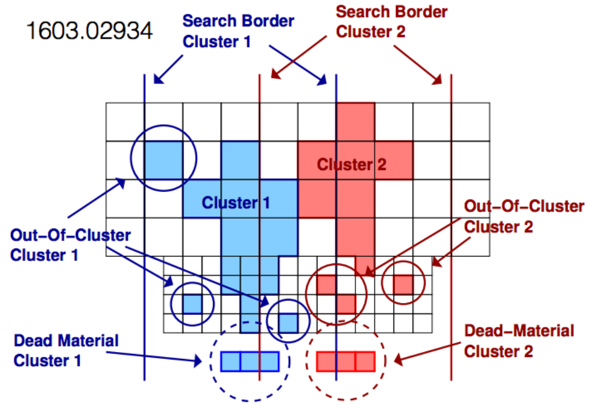


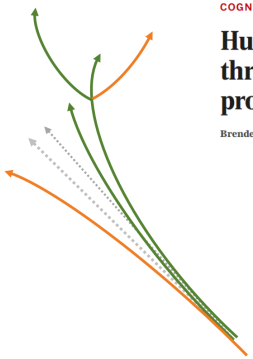
Looking at Data in New ways, using low level info

- Look at data in new ways, potentially a lot of exciting tools available! Just 2 examples:
  - Computer vision and Imaging
    - Already being studied for jet physics
    - Potential for use for e/gamma cluster ID, topocluster ID / calibration, tau clusters?
      - Similar ideas being investigated for ID in LArTPC's
  - Sequence learning / Machine translation
    - Process variable length sequences with recurrent neural networks
    - Fist studies for processing tracks for b-tagging



1603.02934

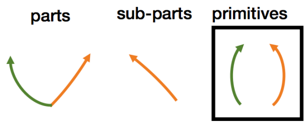




COGNITIVE SCIENCE

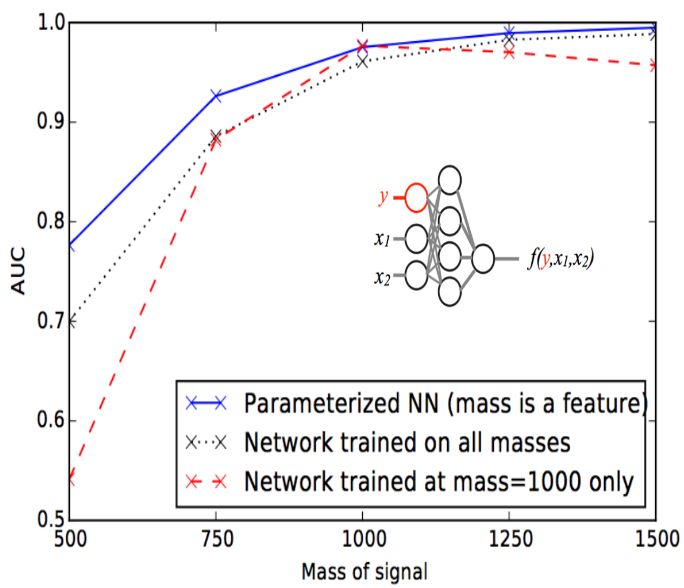
Human-level concept learning through probabilistic program induction

Brenden M. Lake,[1*] Ruslan Salakhutdinov,[2] Joshua B. Tenenbaum[3]



ML in Analysis

- ML is vital for some analysis (e.g. ttH, Single top, $B_s \rightarrow \mu\mu$)
- For most analyses, ML can help push sensitivity
  - We are very good at doing physics already!
- Tend to replace cut based selection by BDT
  - "We run a lot of BDT's" → break down complex analyses into classification problems
- New ideas to ease examination of many signal regions
- ML may help improve statistical analysis techniques
- Take step back: Investigate new ways to look at data,
  - Reconstructing complex final states
  - Interplay between ML and Matrix Element techniques
  - Different approaches to handle multiple classifications
  - Improved inputs to MVA can lead to significant gains

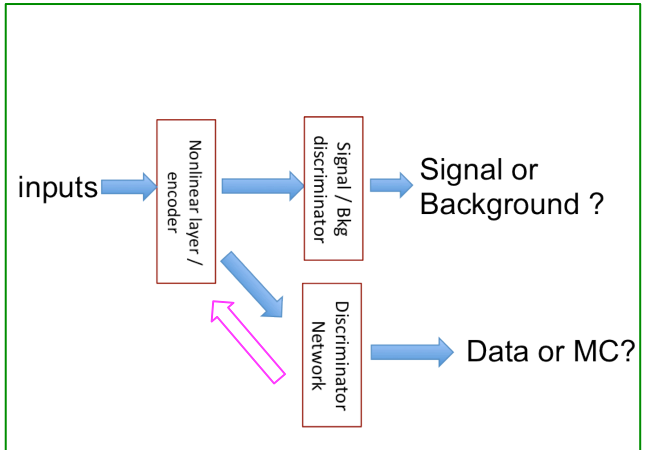


D. Whiteson et al.

ML in Analysis : systematics

Q. Buat et al.

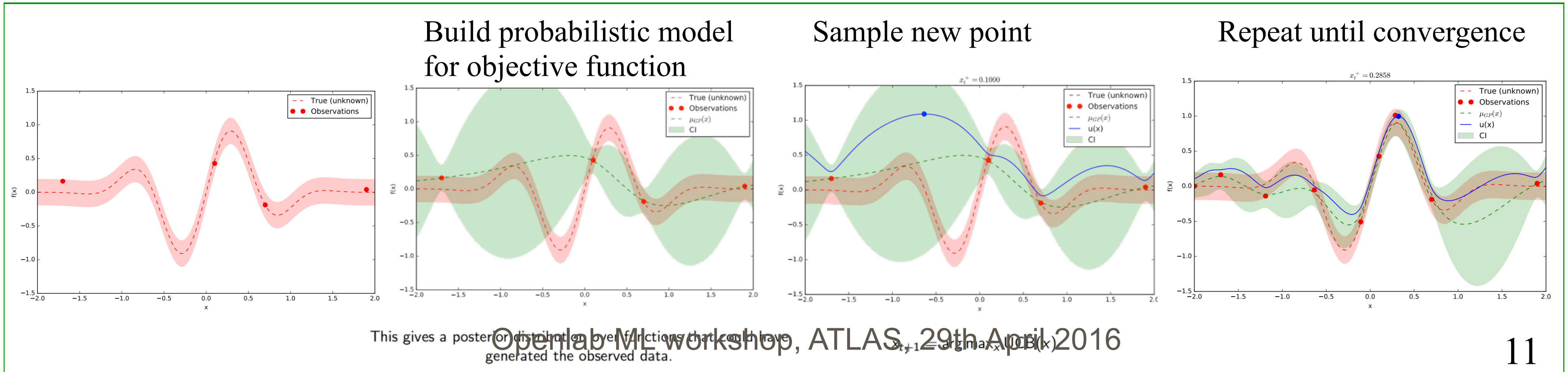
- Our experimental papers typically ends with
  - measurement = m ± σ(stat) ± σ(syst)
  - σ(syst) systematic uncertainty : known unknowns, unknown unknowns…
- Name of the game is to minimize quadratic sum of : σ(stat) ±σ(syst)
- ML techniques used so far to minimise σ(stat)
- Impact of ML on σ(syst) or even better global optimisation of σ(stat) ± σ(syst) is an open problem
- Worrying about σ(syst) untypical of ML in industry



ML in Simulation

- We invest a lot of resources (CPU: ~100k cores *year, human) on very fine tuned simulations:
  - so far very manual optimisation by super experts
  - optimisation in many dimensions parameter space, with costly evaluation

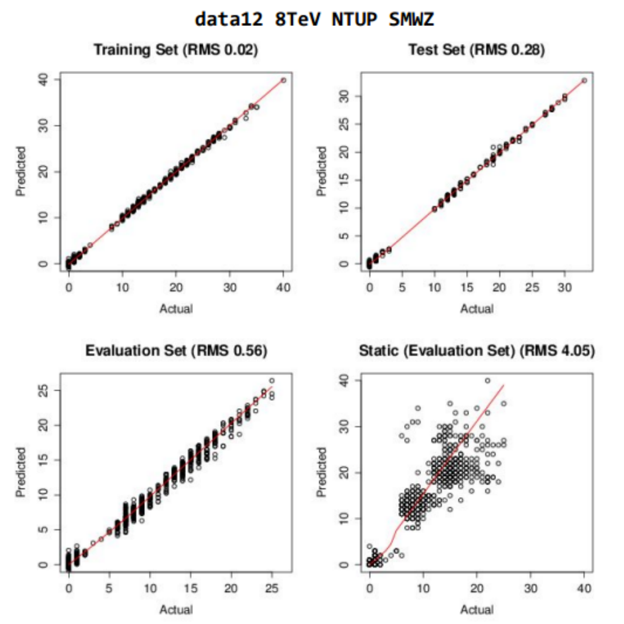
- Now turning to more modern techniques e.g.:
  - Bayesian Optimization and Gaussian Processes



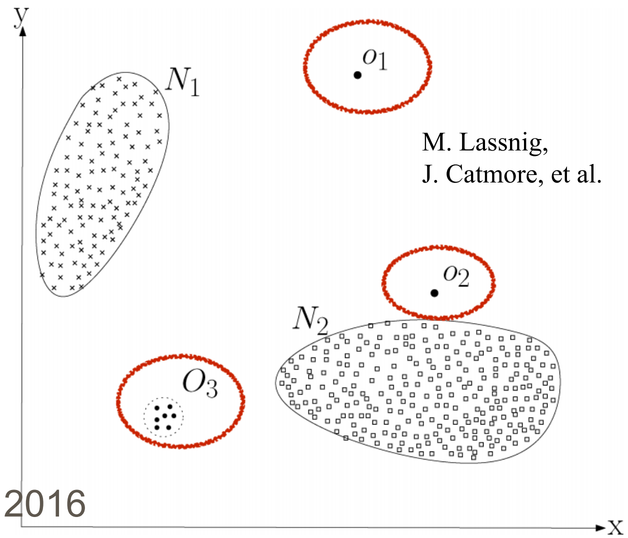
ML in Software/Computing

- Workflow and request completion optimization
  - Under dynamic task constraints
- Data brokering
  - Limited CPU and world-wide distributed storage resources for an increasing volume of data
  - How best to place data around different sites for optimal access? Where best to send jobs?
- Anomaly detection and prevention
  - Sometimes, things don't work as expected
  - Automated preventive measures
  - Monitor computing infrastructure? Aid Data Quality? MC / Data production validation?

  - Could also be useful for a generic search for new physics, when we don't know exactly what we are looking for?



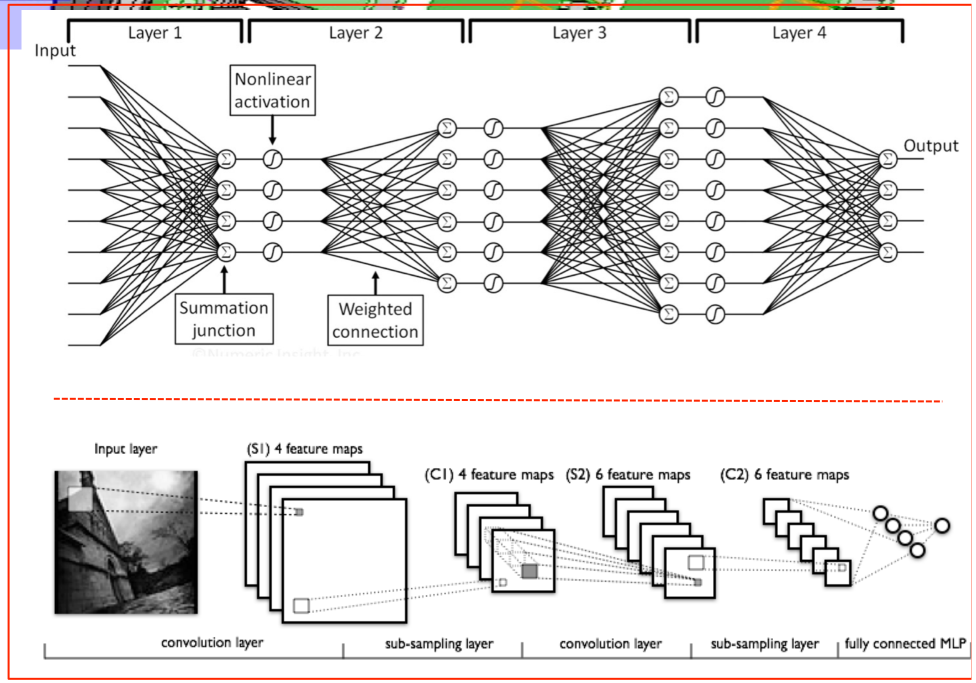
M. Lassnig et al.

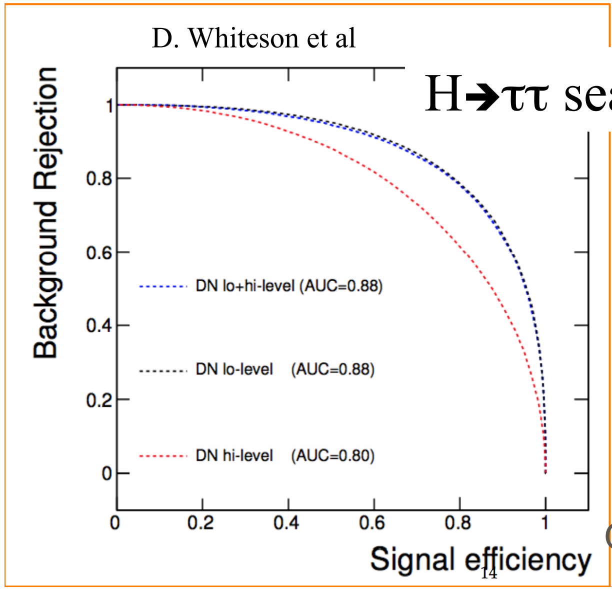


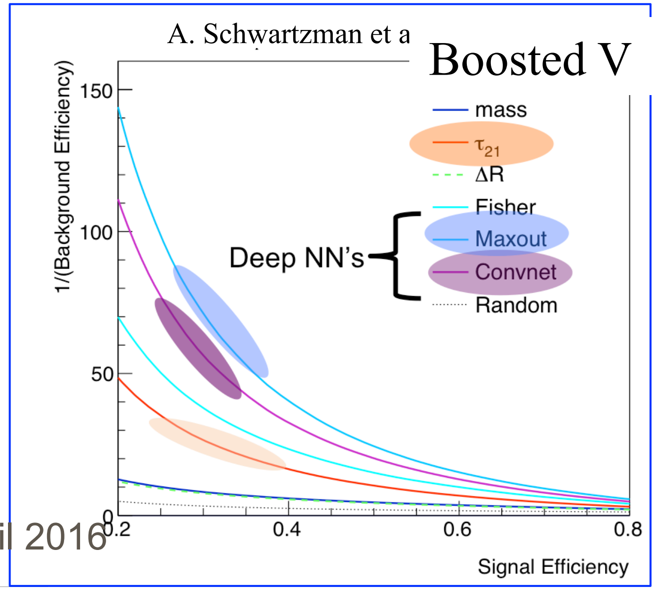
M. Lassnig, J. Catmore, et al.

Potential for Deep Learning

- Variety of studies inside and outside ATLAS show large potential for using DNN’s
  - Improving reconstruction (particle identification)
  - Improving Searches and Analyses
  - …
- These have been seen to be very powerful algorithms in ML, definitely worth more exploration on ATLAS







Tools

- Root-TMVA heavily used, but it is limited
  - Easy to use on lxplus and local batch systems, nicely integrated to Root
  - But does not currently have many modern algorithms / validation techniques
  - Can lead to large usage of memory
  - We understand new development efforts underway, very welcome
- Many people trying new algorithms / techniques are using common data science tools (Scikit-learn, Xgboost, Theano, TensorFlow,…)
  - Such tools not available on lxplus
  - Quite some “plumbing” to integrate these tools in typical HEP workflow
  - Common installations for popular tool would lower barrier to use in ATLAS

ML platforms

- Training time can become prohibitive (days), especially Deep Learning, especially with large datasets
- With hyper-parameter optimisation, cross-validation, number of trainings for a particular application large ~100
- We're exploring ML platforms :
  - Dedicated cluster (with GPUs)
  - Relevant software preinstalled (VM)
  - Possibility to load large datasets (GB to TB)

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 NDA
- 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. ACTS for tracking, see later)
- Role of CERN Open Data portal:
  - We (in ATLAS) initially saw its use for outreach purposes
  - But after all ML collaboration is a kind of scientific outreach
  - ➔We've uploaded there in 2015 the data from Higgs Machine Learning challenge (essentially 4-vectors from full G4 ATLAS simulation Higgs->tautau analysis)
  - We consider releasing more datasets dedicated to ML studies

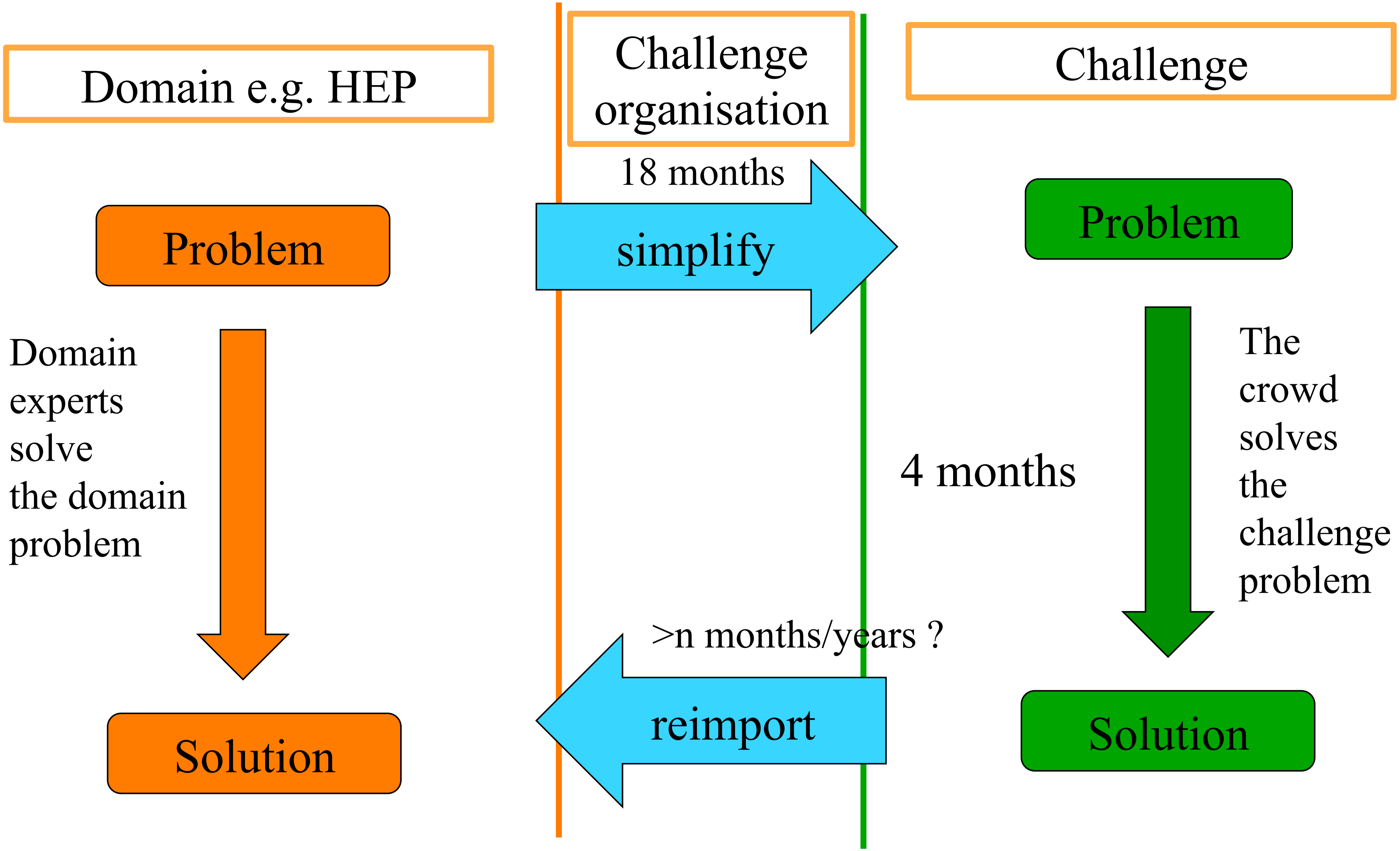
Challenges (competition)

- Challenges are essentially a way to create a buzz around an open dataset dressed with a benchmark
  - HiggsML (ATLAS) 2014
  - FlavourML (LHCb) 2015
  - future TrackML (ATLAS+CMS) 2016?
- Buzz in non-HEP world to get the attention of ML specialists
- More on this now

HiggsML in a nutshell

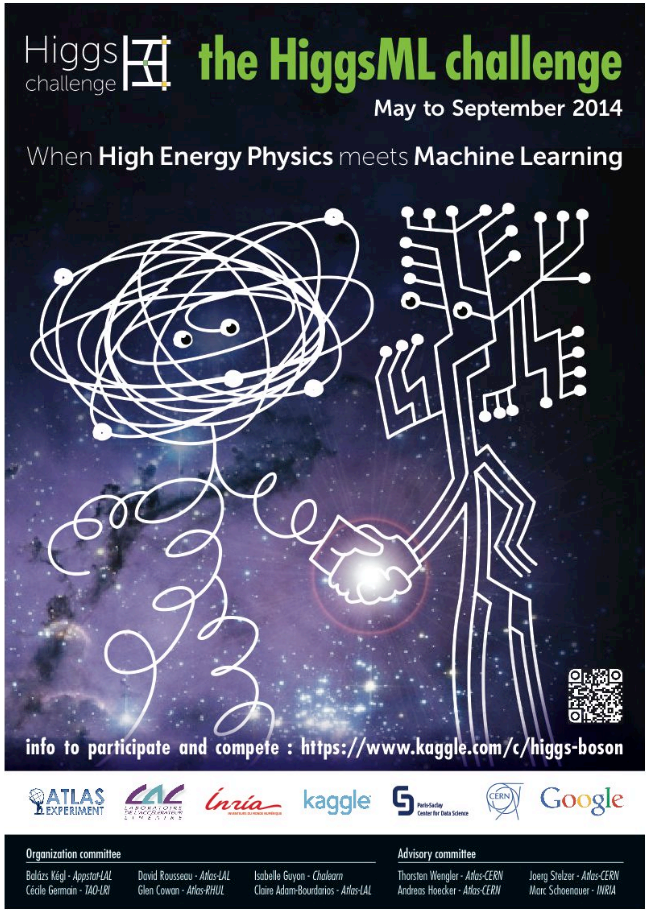
- Why not put some ATLAS simulated data on the web and ask data scientists to find the best machine learning algorithm to find the Higgs ?
  - Instead of HEP people browsing machine learning papers, coding or downloading possibly interesting algorithm, trying and seeing whether it can work for our problems
- Challenge for us : make a full ATLAS Higgs analysis simple for non physicists, but not too simple so that it remains useful
- Also try to foster long term collaborations between HEP and ML
- Do not underestimate the time to learn common languages (e.g. hand waving explanation of S/sqrt(B) not enough)
- Do not underestimate the percolation time :
  - 1) New ML ideas ➔ 2) Demo on toy data set ➔ 3) Demo in real ATLAS analysis➔4) published ATLAS analysis ==> we're still between 1 and 2 for most new ideas

From domain to challenge and back



Higgs Machine learning challenge

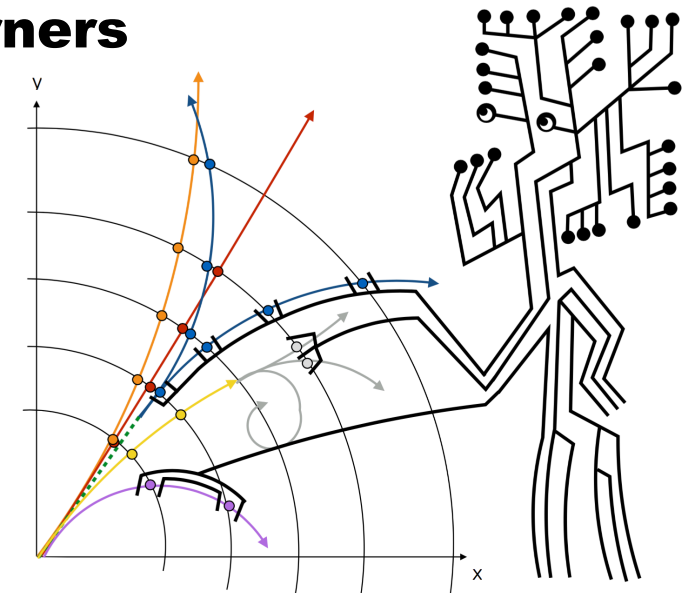
- See talk DR CTD2015 Berkeley
- An ATLAS Higgs signal vs background classification problem, optimising statistical significance
- Ran in summer 2014
- 2000 participants (largest on Kaggle at that time)
- Outcome
  - Best significance 20% than with Root-TMVA
  - BDT algorithm of choice in this case where number variables and number of training events limited (NN very slightly better but much more difficult to tune)
  - XGBoost best BDT on the market (quite wide spread nowadays)
  - 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

Future Tracking Machine Learning challenge

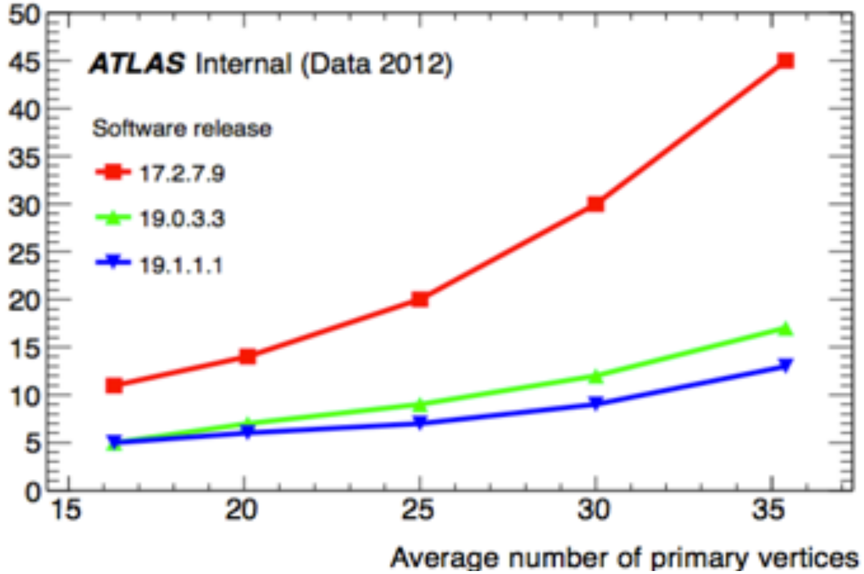
A collaboration between ATLAS and CMS physicists, and Machine Learners



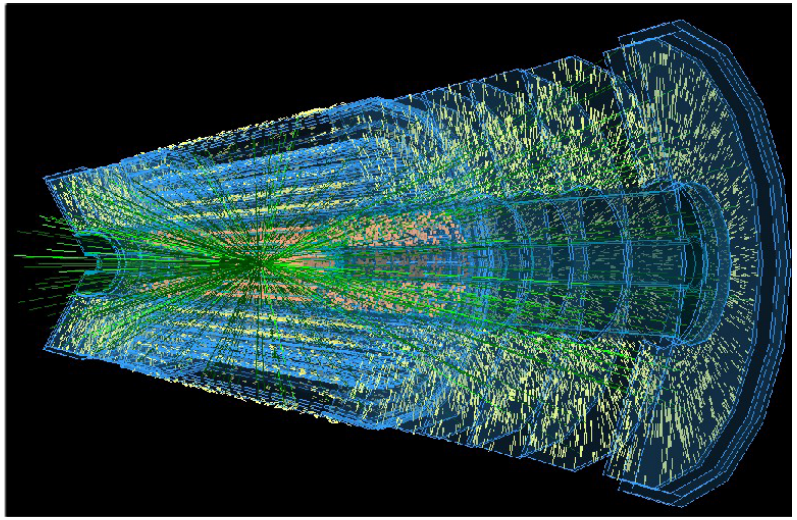
TrackML : Motivation 1

Graeme Stewart ECFA HL-LHC workshop 2014

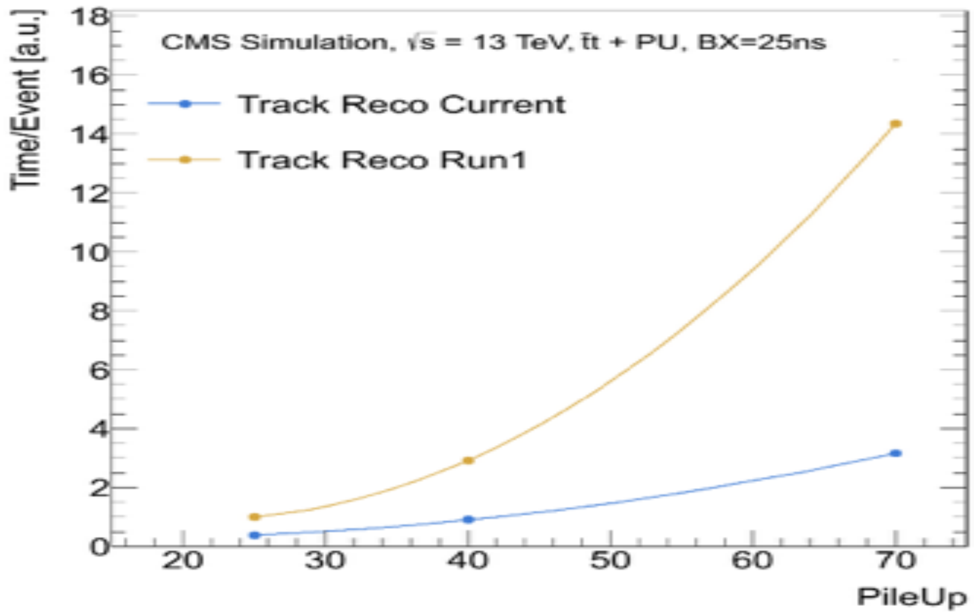
- Tracking (in particular pattern recognition) dominates reconstruction CPU time at LHC
- HL-LHC (phase 2) perspective : increased pileup :
  - Run 1 (2012): <>~20
  - Run 2 (2015): <>~30
  - Phase 2 (2025): <>~150
- CPU time quadratic/exponential extrapolation (difficult to quote any number)



150



Openlab ML work



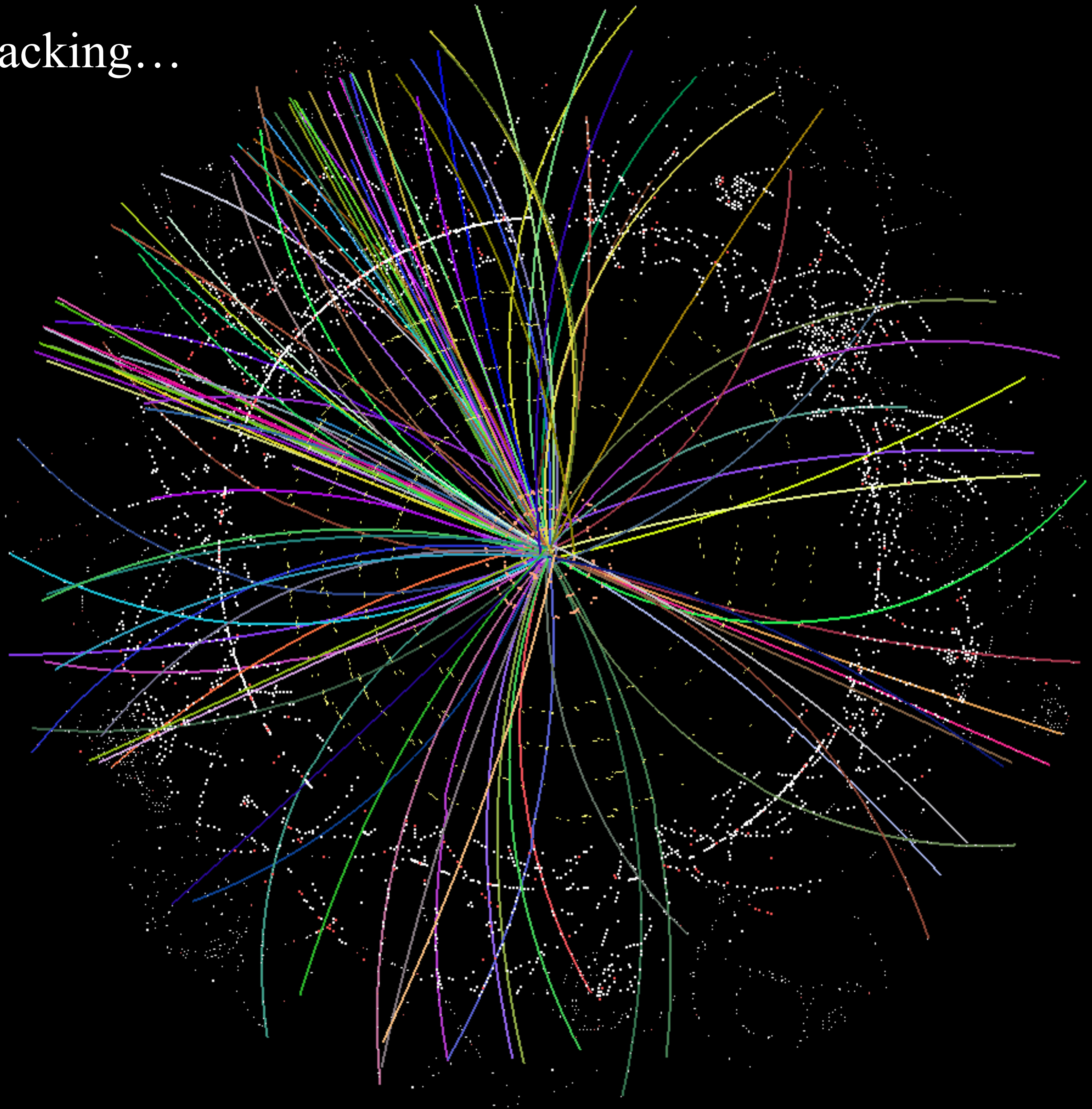
TrackML : Motivation 2

- LHC experiments future computing budget flat (at best)
- Installed CPU power per \$==€==CHF expected increase factor ~10 in 10 years
- Experiments plan on increase of data taking rate ~10 as well (~1kHz to 10kHz)
- ➔HL reconstruction at mu=150 need to be as fast as Run1 reconstruction at mu=20
- ➔requires very significant software improvement, factor 10-100
- 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)
- Need to engage a wide community to tackle this problem

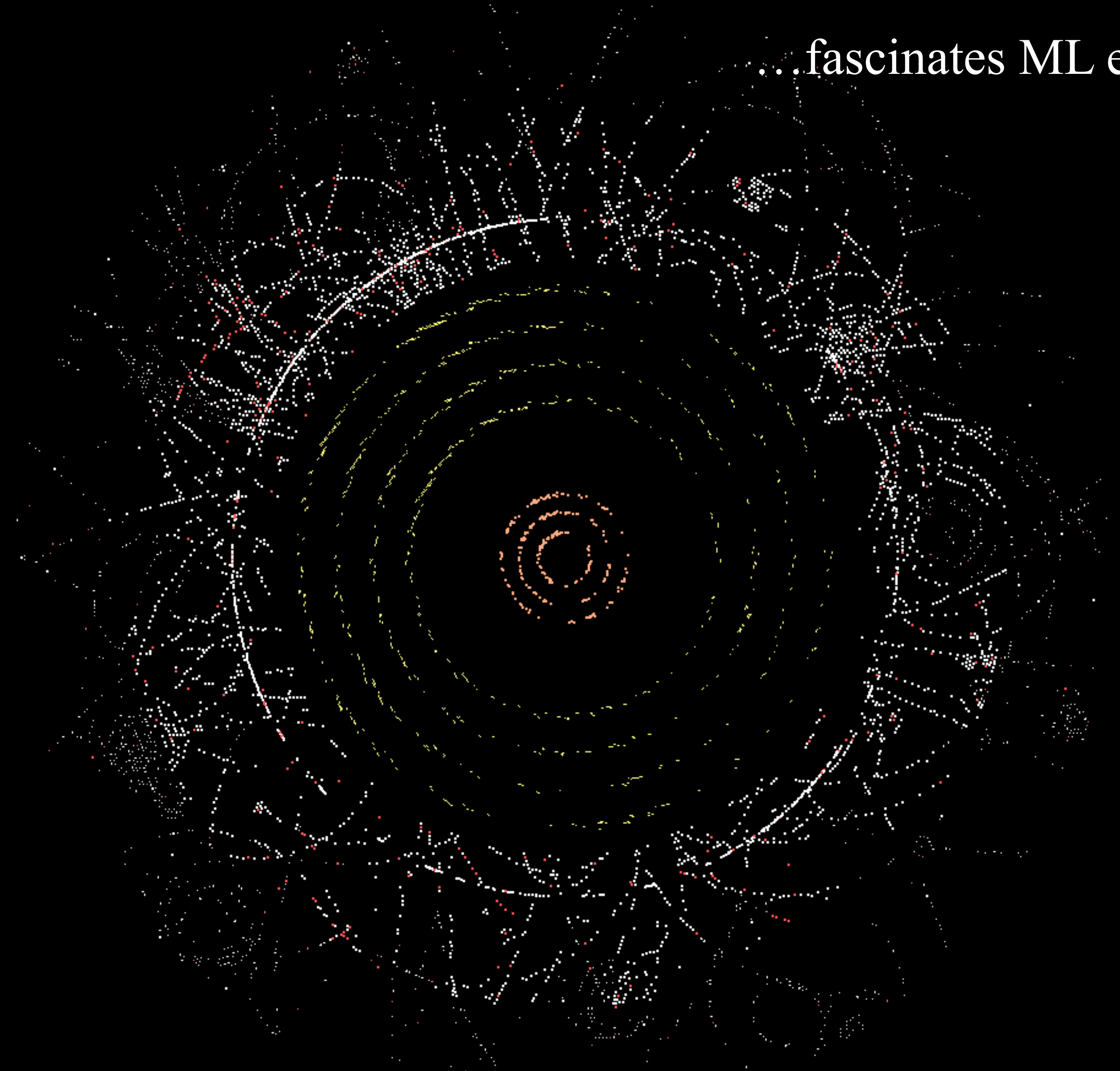
TrackML : engaging Machine Learners

- 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

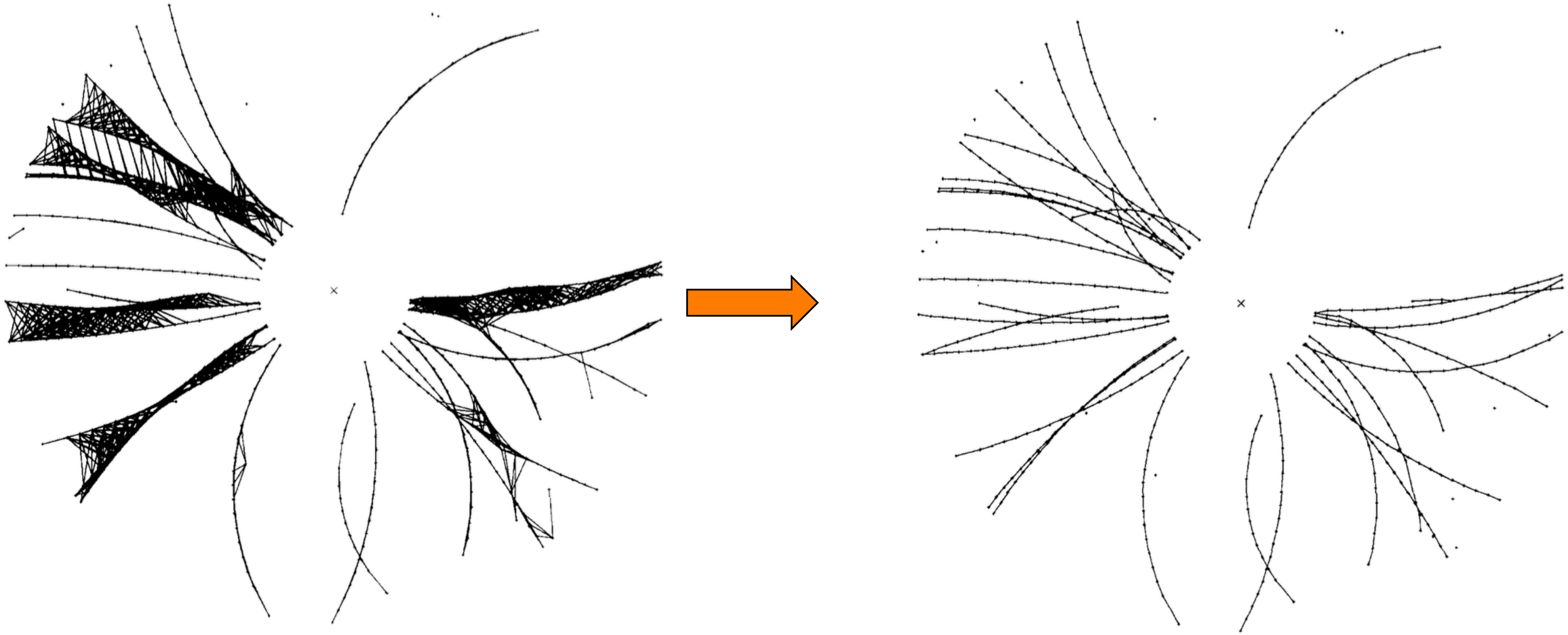
HEP tracking…



…fascinates ML experts



TrackML : An early attempt



- Stimpfl-Abele and Garrido (1990) (ALEPH)
- All posssible neighbor connections are built, the correct ones selected by the NN (not used in production)
- Also PhD Vicens Gaitan 1993, winner of Flavour of Physics challenge

TrackML :CPU measurement

- Contrary to HiggsML or Flavour of Physic challenge need to evaluate CPU time
  - We know already how to solve the problem, but not quick enough (by factors)
  - CPU time to find the tracks
  - Cap on memory used (e.g. one x86-64 core with 2GB)
  - Training time unlimited
- Some platforms (see AutoML, Codalab, Topcoder) now allow to automatically upload, compile and run software
  - ➔well defined hardware (CPU and memory available)
  - ➔uniform comparison
  - Could also use an Amazon instance
- Positive side-effect : limit diversity of software languages and libraries
- Also the training dataset is very large (~1TB), better left on the platform
- We're more interested in the detailed algorithm (as it would be explained in a technical paper) rather than the software itself (but we do want to see the software)
- We're more interested in new approaches than in super-optimised version of old approaches
- We're looking for industry involvement there : a powerful platform to handle 100-1000 training per day on 1TB dataset

Conclusion

- Machine Learning techniques widely used on ATLAS
- Recent explosion of novel (for HEP) ML techniques, novel applications for Analysis, Reconstruction, Simulation, Trigger, and Computing
- ATLAS Machine Learning group being set up by June 2016
- Building collaborations with ML scientists on a variety of issues
- Looking for powerful hardware+software platforms for:
  - Deep NN training
  - Tracking ML challenge