This was a very interesting workshop on the fast developing field of ML. In the name of all attendants, I would like to express my **sincere thanks to Young-Kee, everyone who helped her, and the Chicago ATLAS group for organising this event**

Andreas Hoecker (CERN)
Andreas.Hoecker@cern.ch

We had …

- an extremely motivating talk by Matt Schwartz with lots of ideas for applications in particular on jet physics,
- a very pedagogical and thoughtful report by Ben Hooberman discussing the challenges of ML and emphasizing the importance of order-independent variable length representations of the data,
- fascinating talks by David Miller and Nhan Viet Tran on the usage of ML in custom hardware triggers,
- and Nhan and Walter Hopkins pointed out that ML can be used as a universal language for optimal use of modern computing hardware and accelerators (neuromorphic computing),
- Walter also gave examples of efficient use cases for ML for new physics searches, reconstruction parametrisation and unfolding, and emphasised as others the need for efficient use of current and forthcoming HPCs for machine learning and event generation.
- Jahred Adelman, Rob Gardner and Mark Neubauer discussed opportunities for training (of people and models) and for opening ML knowledge & usage to more people and applications.

A few general remarks

- Matt pointed out that ML is an exciting new field, and I may add here that — in addition to providing the MC generators we use for training — the creativity of theorists, often in collaboration with experimentalists, helps drive ML forward and broaden the spectrum of possible applications.
- HEP has a lot of experience with multivariate classification and regression and benefited from MVAs through its history, not only for the discovery of the Higgs boson. But modern ML goes beyond multivariate analysis: we are talking about a new qualitative stage allowing us to exploit low-level features of our data.
- Fortunately, the times where one had to justify the mere use of a neural network against strong voices in an experimental collaboration (the "black box" argument), or where one had to do a cut-based validation analysis ("which one could understand") in parallel, are behind us. Today ML is widely accepted as a useful tool to improve performance.
- This being said, we also have become more mature and realistic in the use of ML, and more critical with respect to hypes and false claims of miracle performance.

- So, given that we are witnessing a revolution of ML, no application should be excluded *a priori*. Everything we do should be creatively looked at with respect to whether ML can help.
- Certainly this workshop exhibited beautiful examples of this.

A few specific comments on ML use

- A problem with NNs used to be convergence in presence of many variables, which is by part the reason why robust BDTs were often preferred in spite of their large training output files. Convergence of NNs has been improved with the rise of deep learning, so there should — in principle — be no need to use BDTs (NN should always perform at least equivalently).
- HEP also has experience with MVA-based regression (such as the ATLAS Pixel cluster reconstruction, energy calibration, uncertainty estimation, pileup removal (PUMML) looks very promising, …), but much less than classification. There must still be low-hanging fruits in the field of ML-based regression (incl. multi-dimensional regression).
- As was said during the meeting, regression should allow to efficiently merge sequential steps of a complex calibration procedure into a single one (eg, jet energy calibration). Another application could be on the theory side improved phase space integration / sampling, or possibly making NLO MC generators more efficient (event generation is very CPU expensive nowadays).
- Similarly, the application of multi-class classification (separating in one go signal and several backgrounds), although equivalent to the training of several dedicated binary classifiers, is more efficient, elegant and ought to be tried more.

Deep ML

- We saw interesting ideas and avenues for the use of deep learning in HEP, but not so many applications have yet made it beyond the exploratory step. As emphasised by Ben, often much more work is needed to bring a tool to the quality needed for professional use in an experiment. We should invest this work!
- We should be critical: what is the best approach to solve a given problem (incl. non-ML applications), rather than: I want to use GANs whatever the application.
- HEP has a long tradition, and many specific solutions to its (specific) problems were developed (just think of Kalman fitters, Gaussian sum filters, etc). The propagation of a track in a magnetic field, its interaction with detector material, or the propagation of showers in a calorimeter follow physical laws that the specific techniques exploit, and ML algorithms have to learn. Replacing or substituting the specific techniques with ML-based tools should be possible but requires R&D.
- Example: vertex reconstruction from tracker hits ("end-to-end deep learning") is a hugely complex problem, not easily solvable with just throwing hit coordinates at a deep NN (of whatever architecture). I find these really complex applications extremely interesting for the new tools we have at hand, and hope we will see breakthroughs in the near future.

- Imaging techniques seem to be very promising for track reconstruction of neutrino scattering events in a LAr TPC, and this may also benefit from the fact that these detectors are new and there is no long tradition of specific track reconstruction. This is an opportunity.
- Fast simulation also is a very promising field of ML application via Generative Models. For example, Generative Adversarial Networks (GANs) and/or Variational Auto Encoders (VAE) are used in ATLAS to mimic multi-D calorimeter shower distributions. However, also here, we need to be critical: we need extremely precise reproduction of the G4 simulation to make it applicable to physics sample production. Exploratory studies are important but not enough if it is to become really useful.
- ML should (hopefully) be the death of lookup tables. It is much more efficient to learn the features of a multi-D parameters space than to fully map it. Could generic tools be developed for such an application?
- Certainly, ML application for substructure identification should be better than custom tools, imaging technology using CNNs seems promising. So, as emphasized by Matt, it is often better to throw all the information at an ML algorithm, rather than computing complicated variables and use (only) these. A similar remark applies to the resource hungry matrix element methods.

- Hardware (MPSoC & FPGA devices) are very promising applications for fast triggers (see David's talk on the ATLAS gFex), and may still provide interesting solutions for the Phase-2 upgrade of ATLAS and CMS, so we should not stop thinking about this now that the TDRs are done.

A personal word to conclude

- ML can only learn what we train it to. It is a technique. It (so far) has no inherent knowledge beyond what we teach it. We are physicists, and — while ML is highly interesting and powerful — it is not physics. As physicists we ask a physics question, invent an experiment that addresses it, build a detector and data acquisition system to realise the experiment, and analyse the data to (hopefully) answer the question. ML is useful in many of these steps but it cannot replace them. So, while exploring the fascinating world of ML, please continue to do think and be creative — in physics!