Machine Learning & Real-Time Analysis @ LHCb

Mike Williams

Department of Physics & Laboratory for Nuclear Science
Massachusetts Institute of Technology

May 4, 2017

The Large Hadron Collider

LHCb

- 70 institutes
- 16 countries
- 700 physicists
- Almost 400 papers!

The Short-Short Version

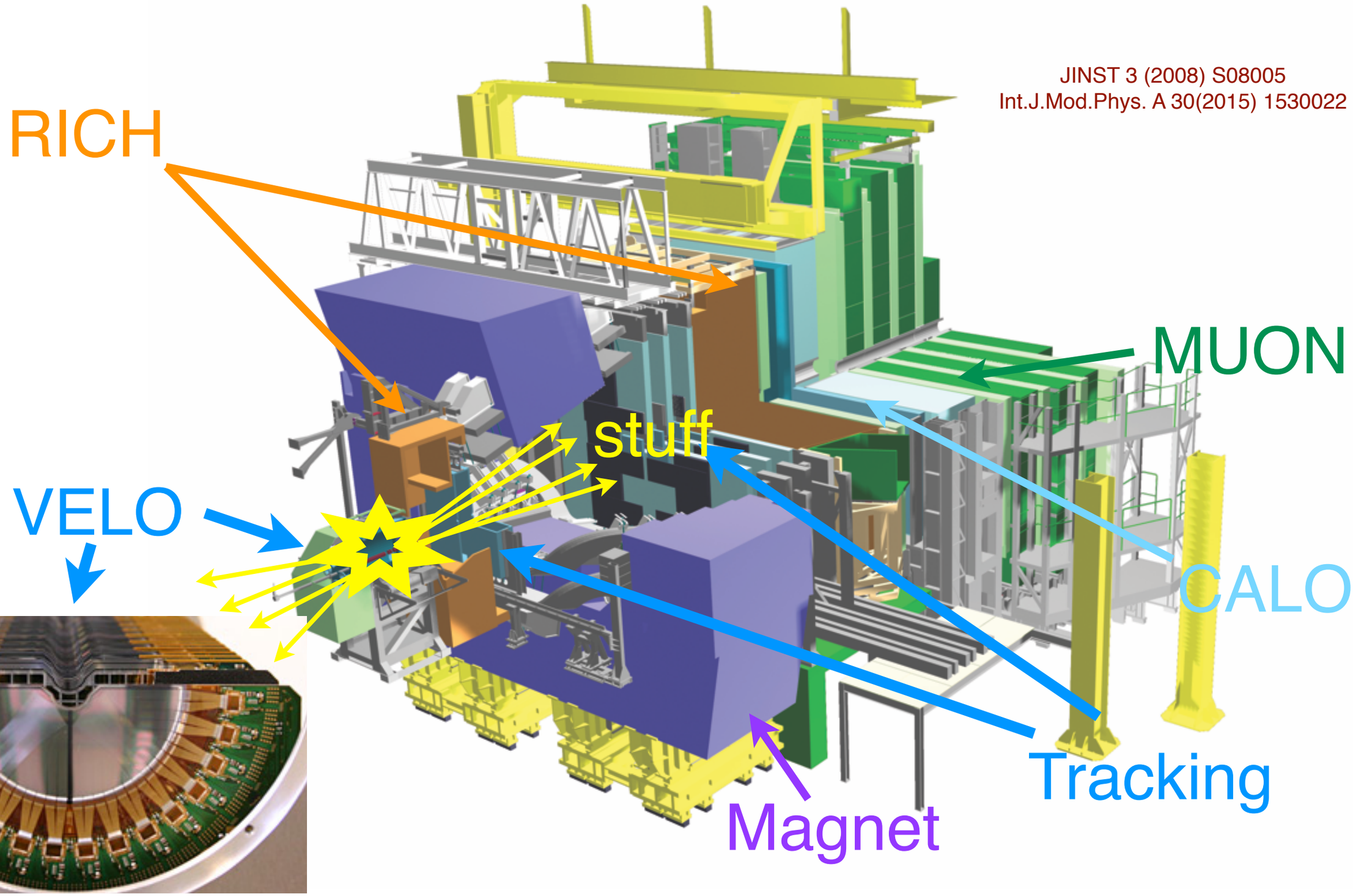
We use ML almost everywhere, and we've moved to a real-time calibration system putting much "analysis" online—to enable great science!

LHCb Detector

LHCb is a forward Spectrometer ($2 < \eta < 5$)

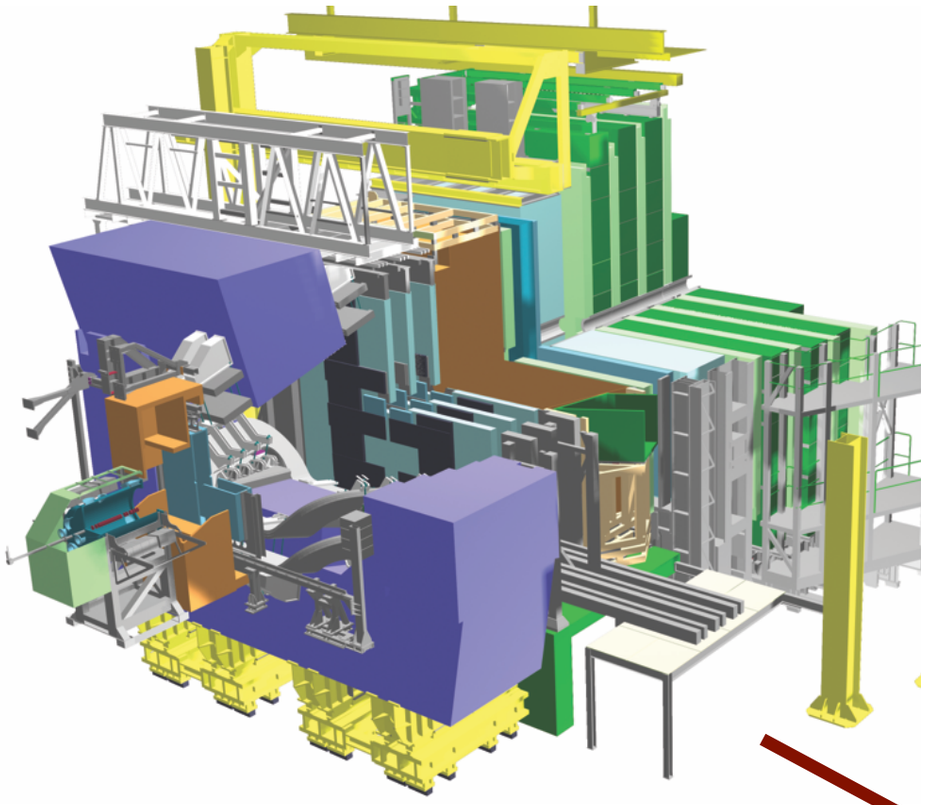
(roughly 1-15^o)

JINST 3 (2008) S08005
Int.J.Mod.Phys. A 30(2015) 1530022



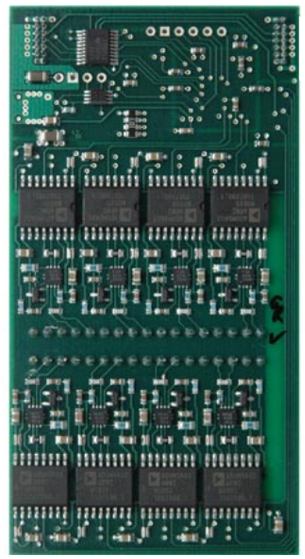
JINST 8 (2013) P04022

Real-Time Processing



Simple feature-building in custom electronics (e.g. FPGAs) required to reduce the data volume to a transferable rate.

TB/s
(post zero suppression)

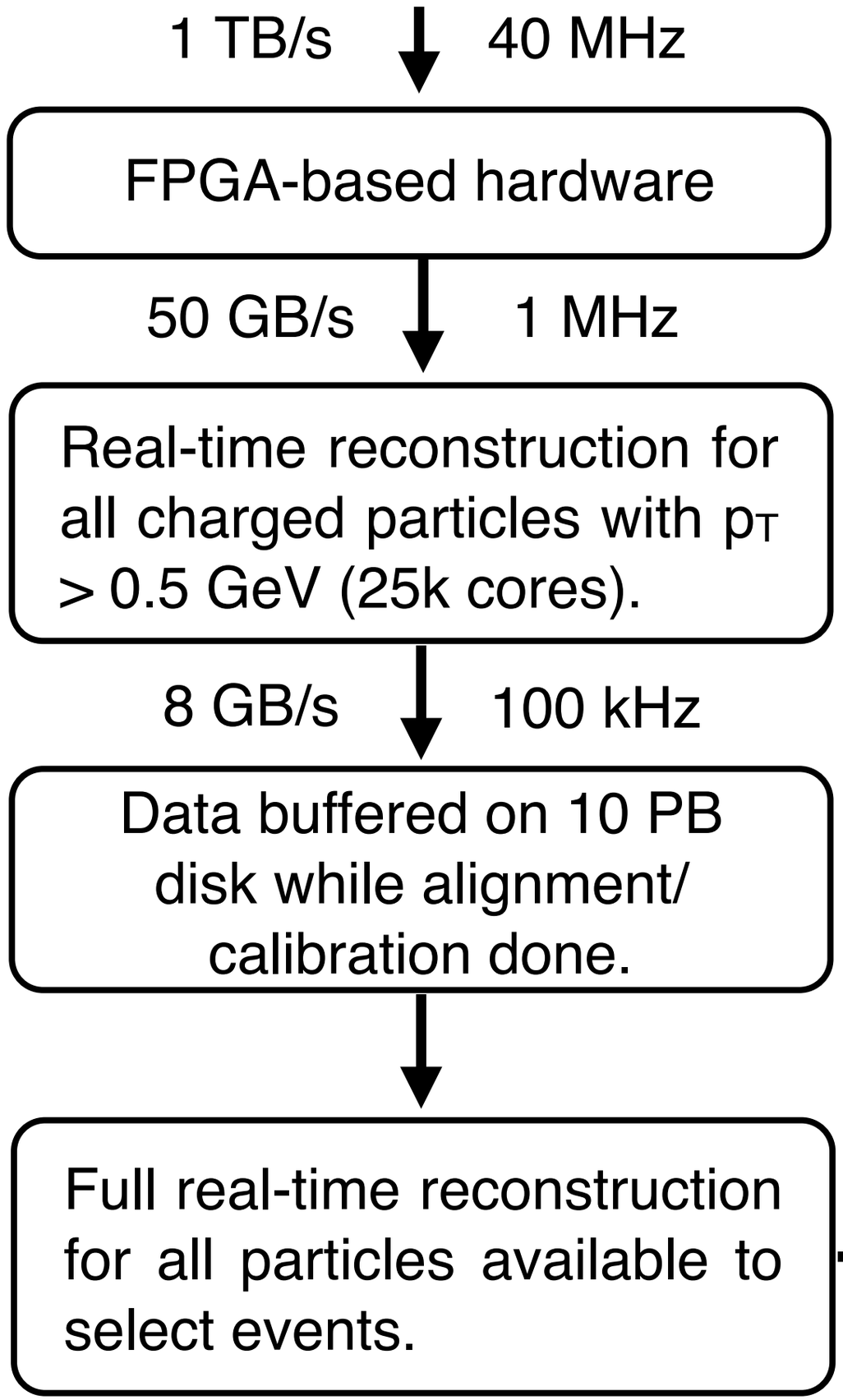


Online computing farm processes 250 PB / year, can only persist 1% of this.

50 GB/s

LHCb will move to a triggerless-readout system for LHC Run 3 (2021-2023), and process 5 TB/s in real time on the CPU farm.

Real-Time Processing (Run 2)



Precision measurements benefit greatly from using the final (best) reconstruction in the online event selection—need real-time calibration!

Final event selection done with access to best-quality data (mostly done during down time between fills), removing the need (but perhaps not the desire) to retain the ability to re-reconstruct the data offline.

Real-Time Processing (Run 2)

1 TB/s ↓ 40 MHz

FPGA-based hardware

50 GB/s ↓ 1 MHz

Real-time reconstruction for all charged particles with p_T > 0.5 GeV (25k cores).

Heavy use of machine learning algorithms throughout the Run 1 and Run 2 trigger.

V.Gligorov, MW, JINST 8 (2012) P02013.

70% of output events here classified using ML algorithms.

8 GB/s ↓ 100 kHz

Data buffered on 10 PB disk while alignment/ calibration done.

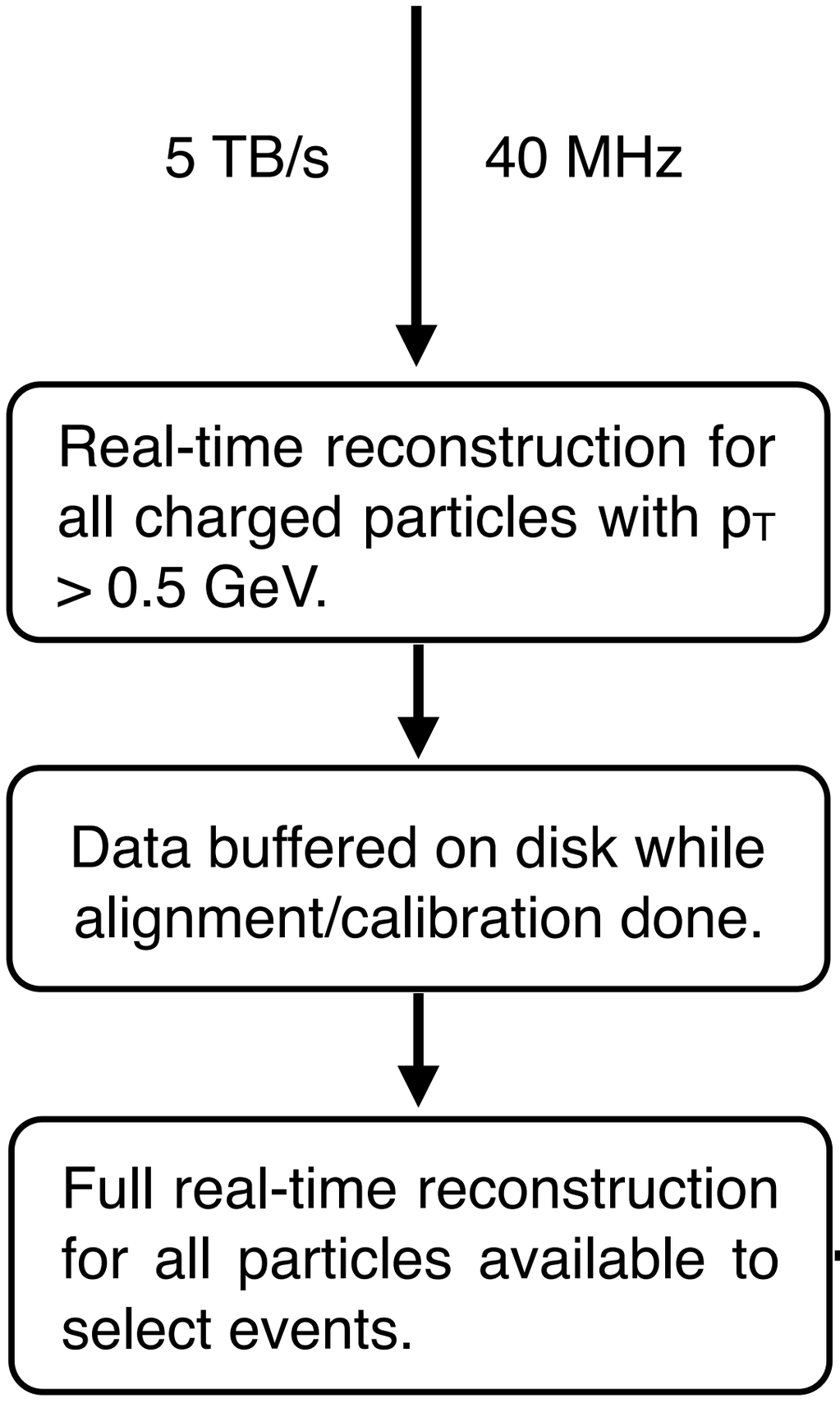
40% of output events here classified using ML algorithms.

ML also used online in tracking, particle ID, etc. (more on this later).

Full real-time reconstruction for all particles available to select events.

→ 5 PB/year (mix of full events & ones where only high-level info kept)

Real-Time Processing (Run 3)



Performing the charged-particle reconstruction on 5 TB/s of data in real time will be a challenge. Investigating ALL options here — use ML to speed it up? (Indeed, we already do some of this.)

Keeping the vast wealth of physics data will also be a challenge. Plan to migrate most of remaining classification to ML-based algorithms. Autoencoder-based data compression?

We are also working on ML-based anomaly detection.

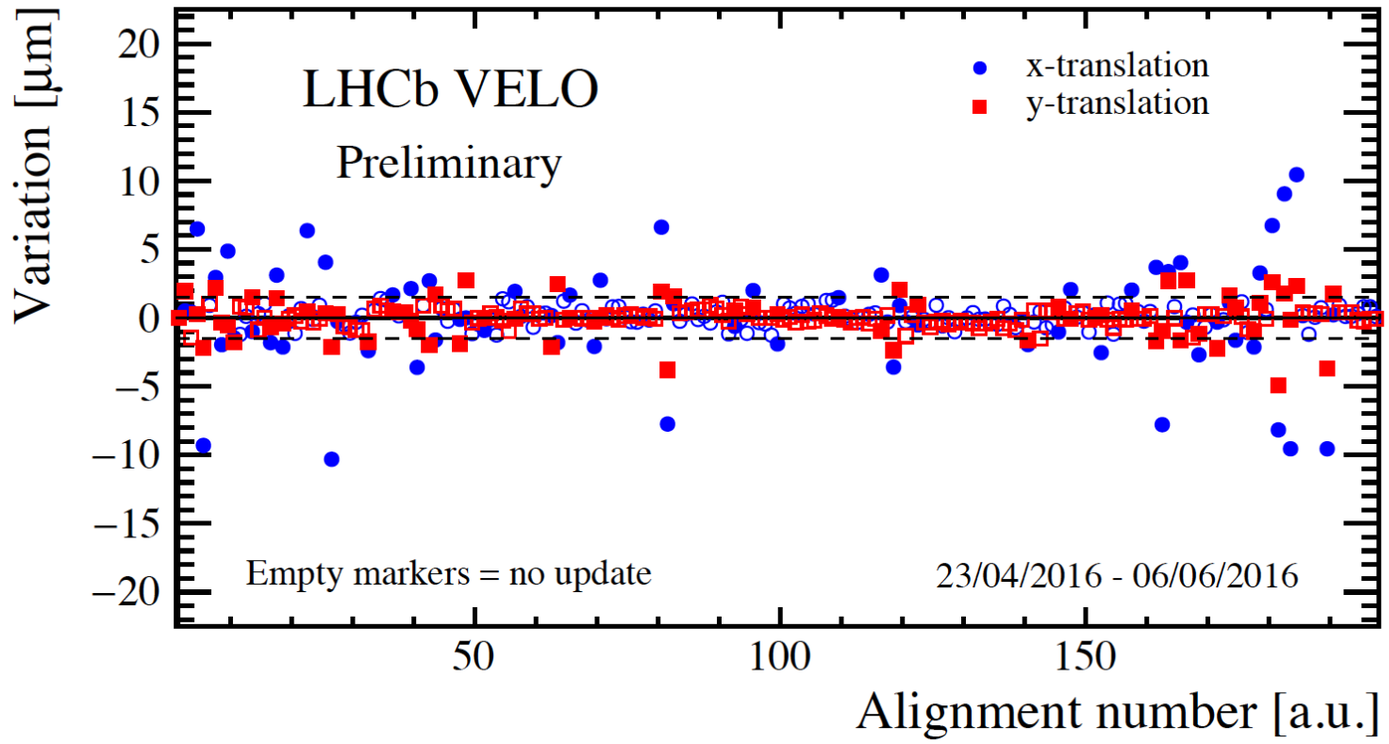
N.b., real-time alignment and calibration is NOT required to use ML in an online system.

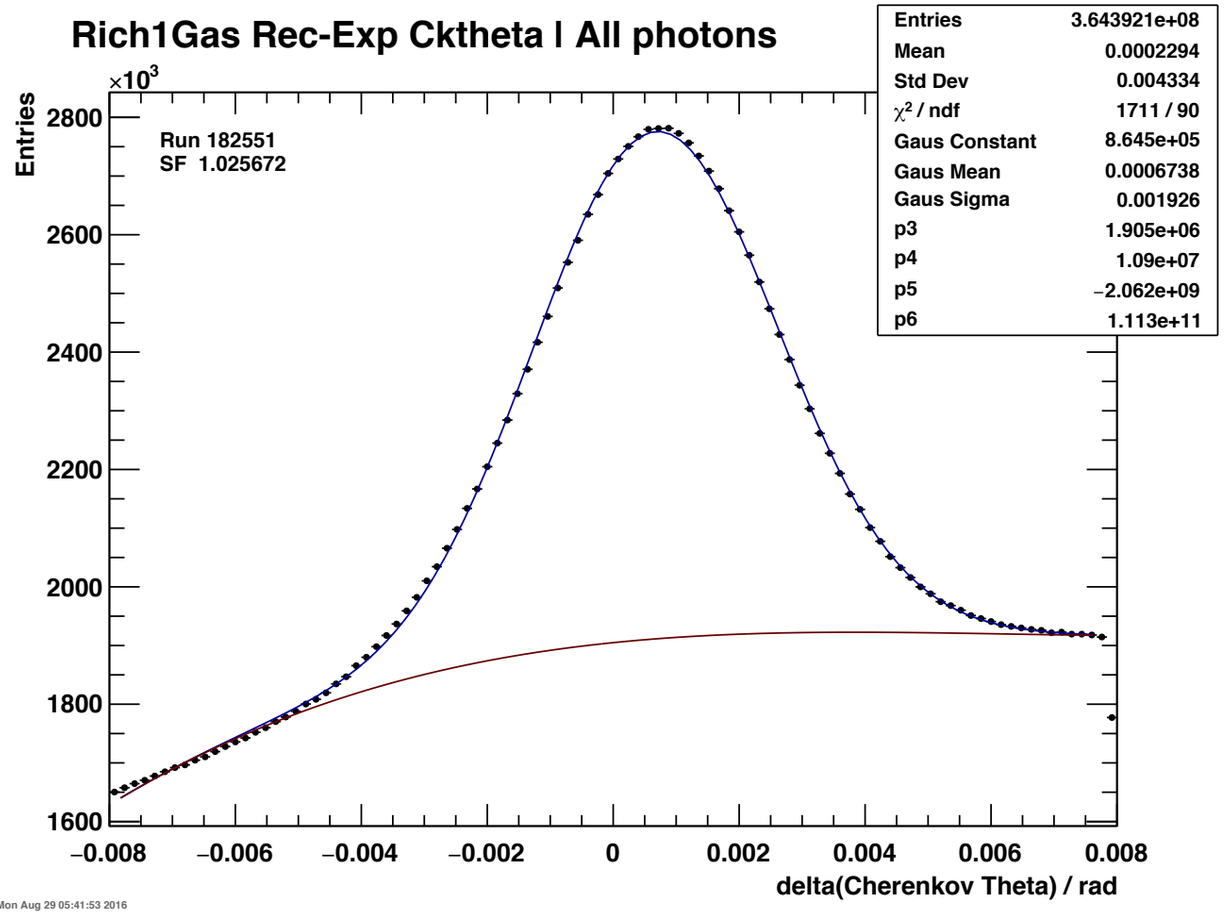
We first introduced ML into our primary event-classification algorithm at the start of 2011 data taking, but real-time calibrations were not implemented until 2015.

Our Run 1 ML-based trigger algorithm collected the data used in about 200 papers to date — and it was run on imperfect data (but designed to be robust against run-time instabilities).

Real-Time Calibration

VELO opens/closes every fill, expect updates every few fills. Rest of tracking stations only need updated every few weeks.





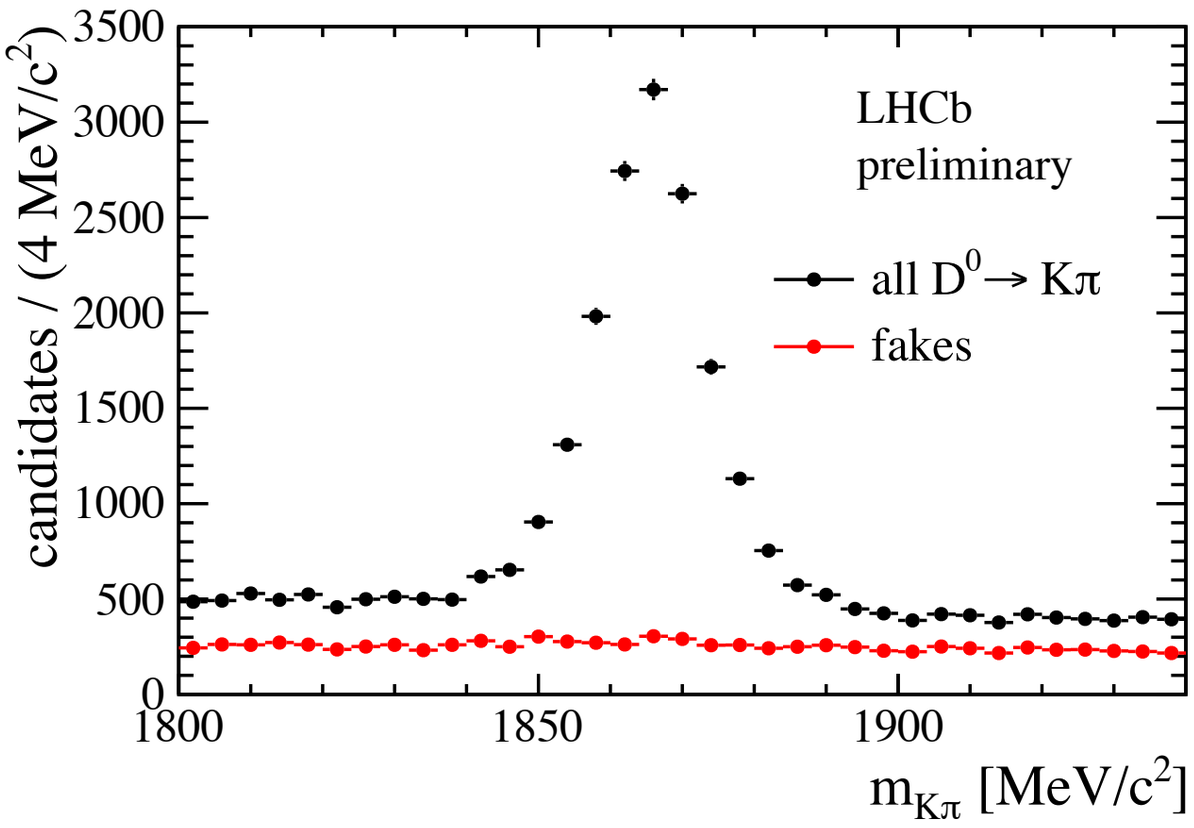
RICH gases indices of refraction must be calibrated in real time; requires ~1 min to run, and new calibrations are required for each run.

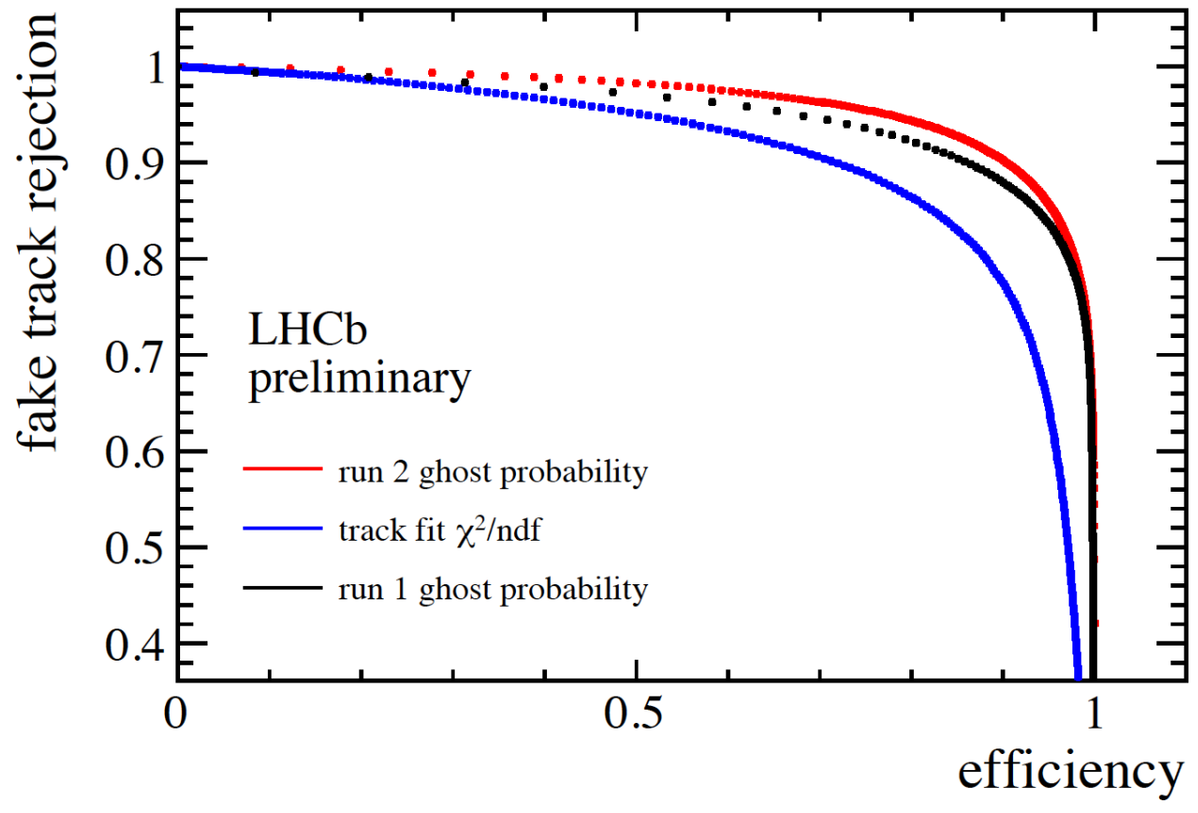
Calibration data is sent to a separate “stream” from the physics data after the first software-trigger stage. This permits running the calibrations on the online farm simultaneously with running the trigger.

Fake-Track Killer

Fake-track-killing neural network, most important features are hit multiplicities and track-segment chi2 values from tracking subsystems.

LHCb-PUB-2017-011



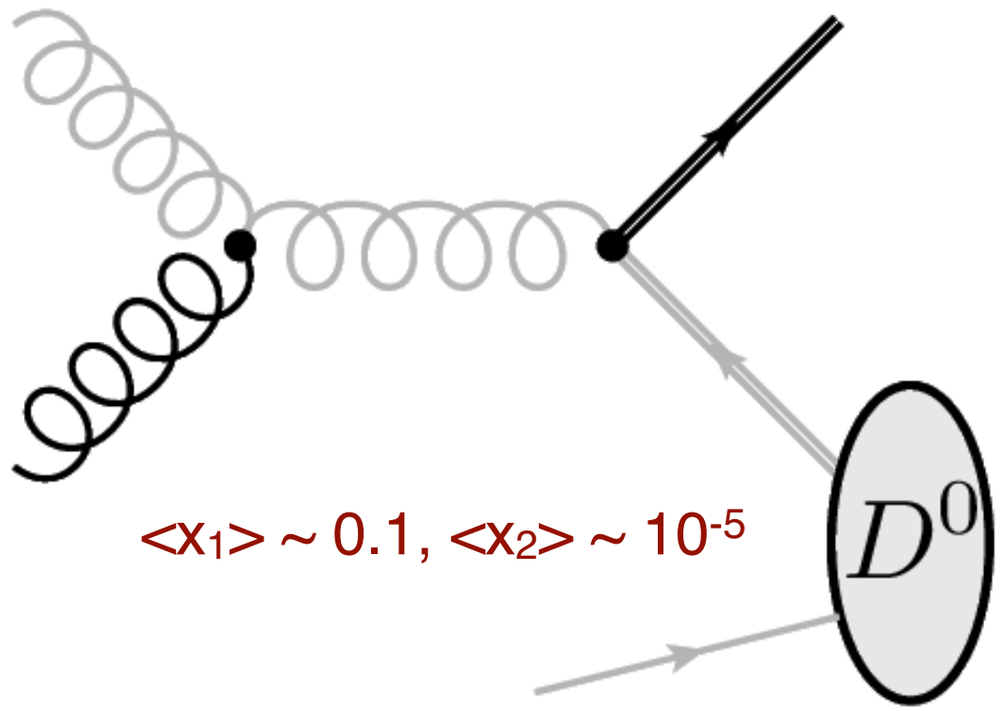


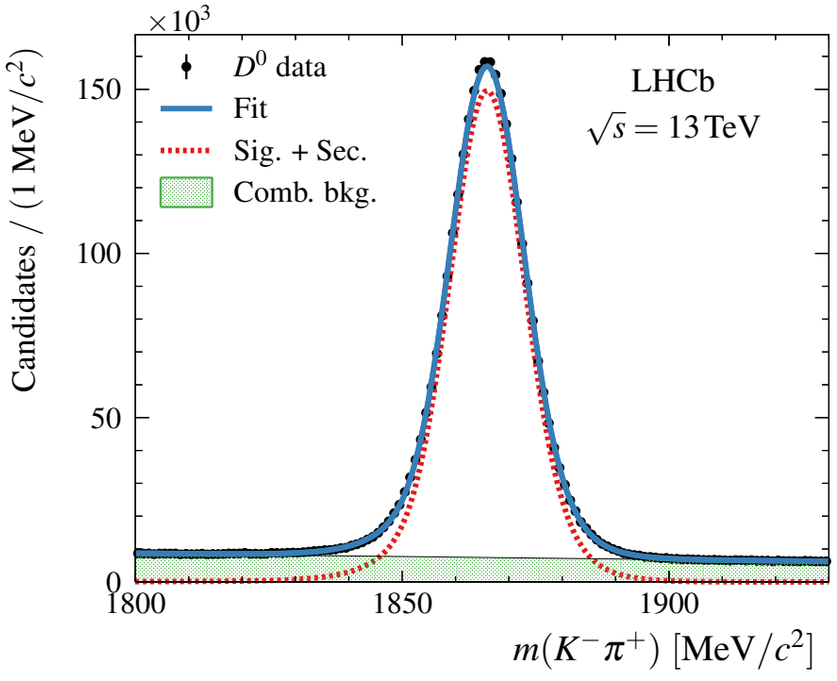
Run in the trigger on all tracks, so must be super fast. Use of custom activation function and highly-optimized C++ implementation.

Open Charm

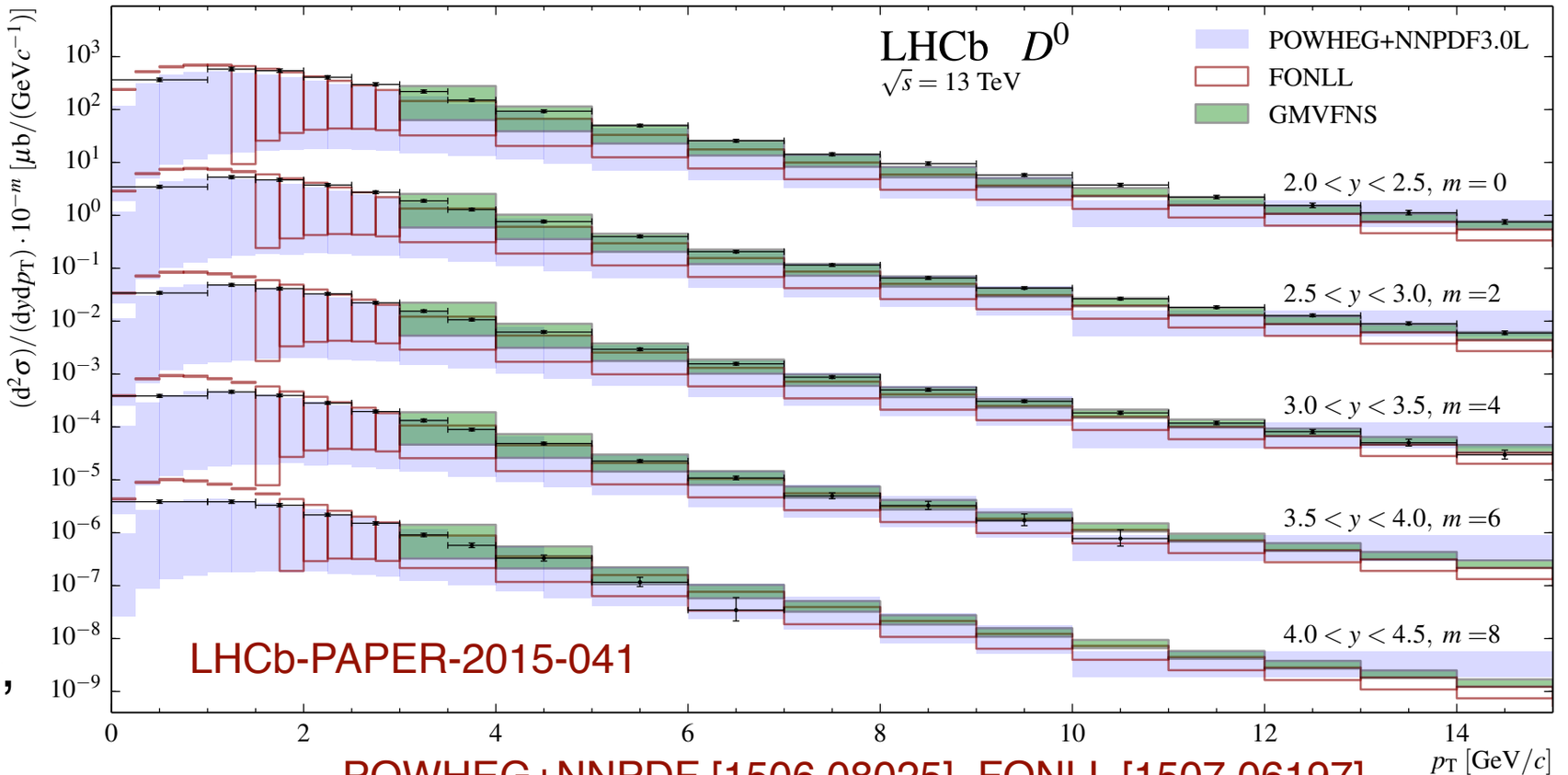
σ(cc)[13TeV] shown @ EPS (2015) within a week of recording the data; it was measured using online-reconstructed data. We achieved better mass and lifetime resolution online than we had offline in Run 1.

Excellent probe of the small-x gluon PDF.





Results also published for D^+, D_s, D^* at both 7 and 13 TeV.



POWHEG+NNPDF [1506.08025], FONLL [1507.06197], GMVFNS [1202.0439]

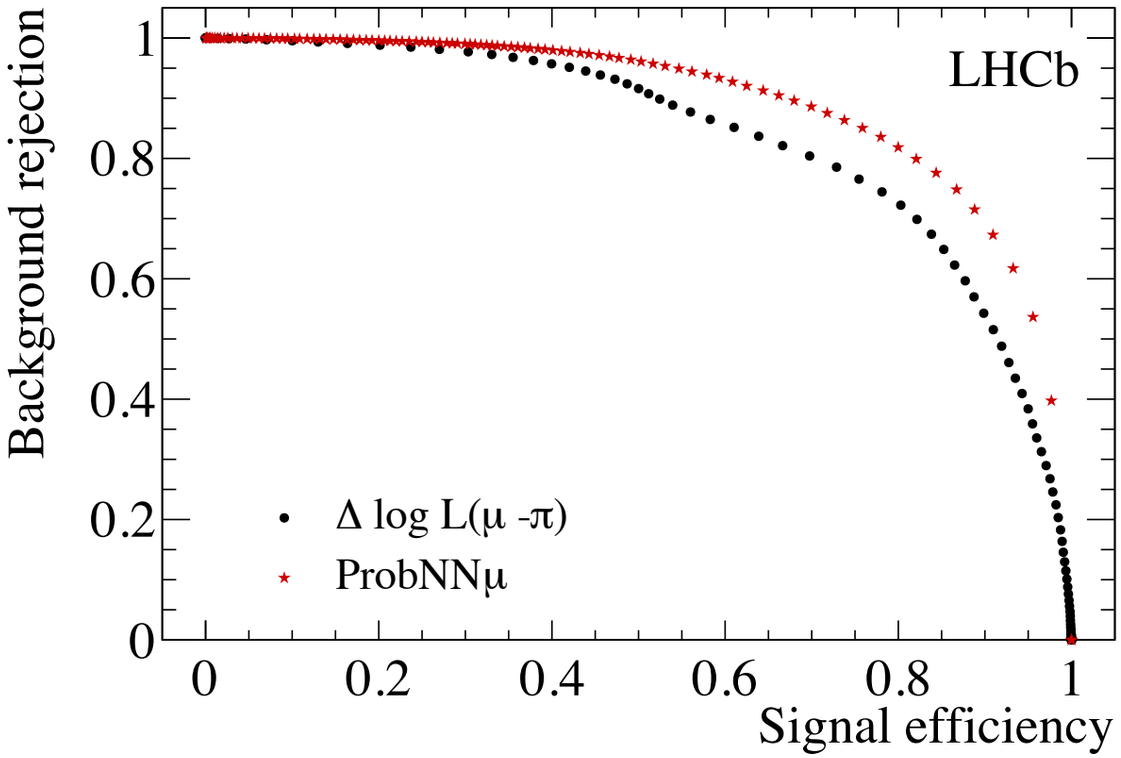
Charged PID

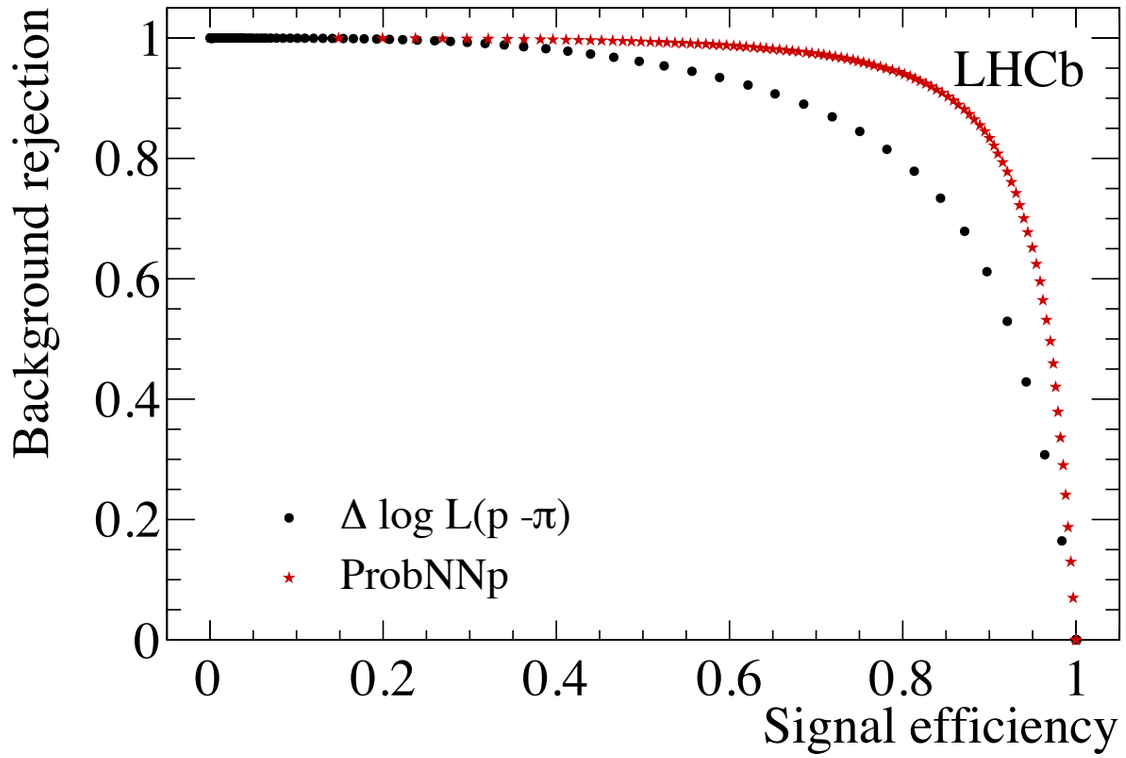
Charged PID: determining whether a track originates from an e, μ, π, K, p, or fake.

Info from the tracking, calorimeter, RICH, and muon systems all play an important role here—and are correlated.

PID NNs

Single-hidden-layer NN trained on 32 features from all subsystems. Each is trained to identify a specific type of particle (or fake track).



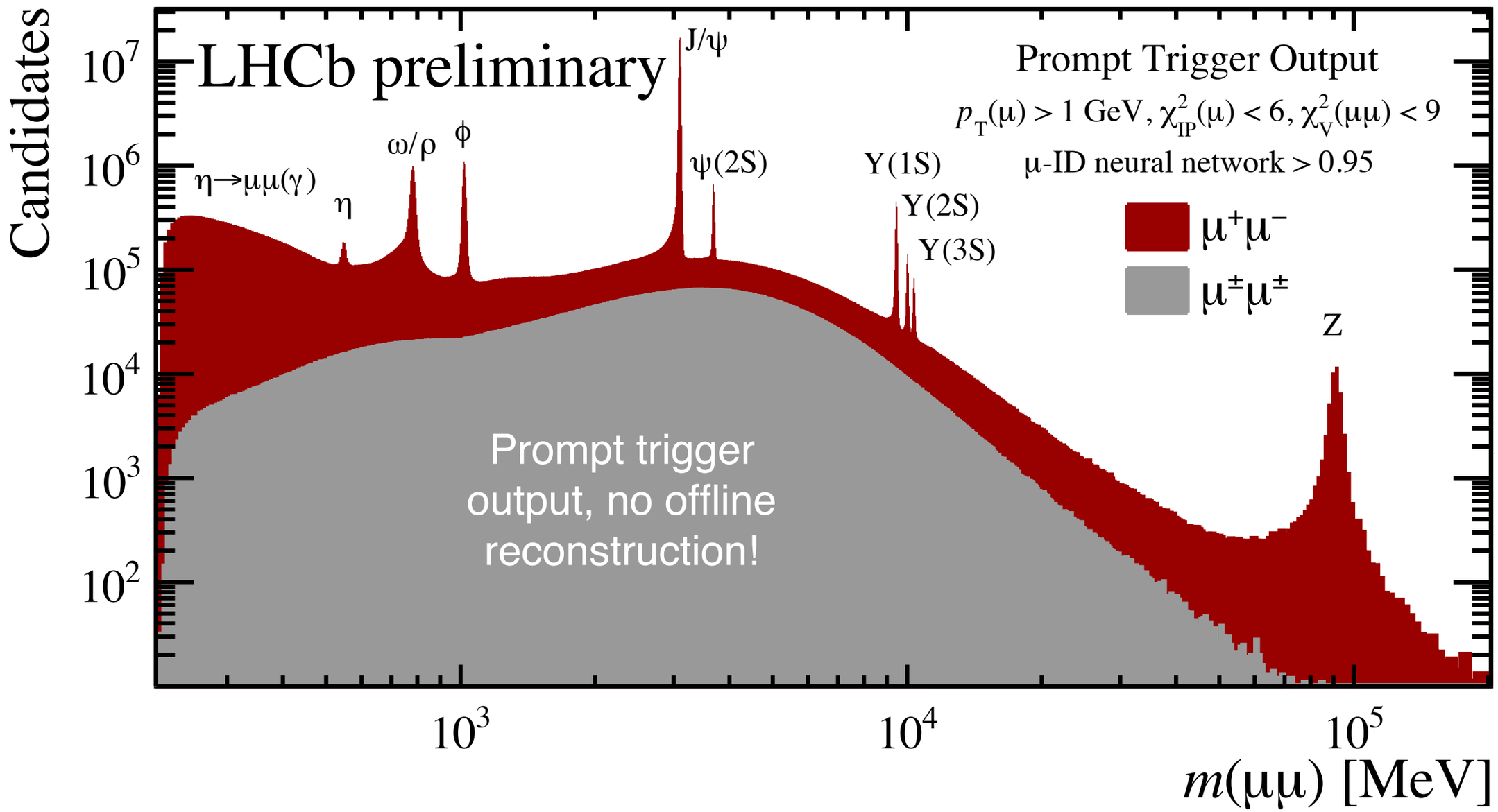


Typically get a factor of 3x less pion contamination in a muon sample than using the CombDLL approach — 10x less in a dimuon sample!

Currently exploring state-of-the-art: XGBoost ~ Deep NN ~ 50% less BKGD than basic BDT or ANN, which again give 2-3x less BKGD than DLLs.

Dark Photons?

New triggers in 2016 for both prompt and displaced dark-photon searches (rely heavily on advances to the LHCb online system in Run 2).

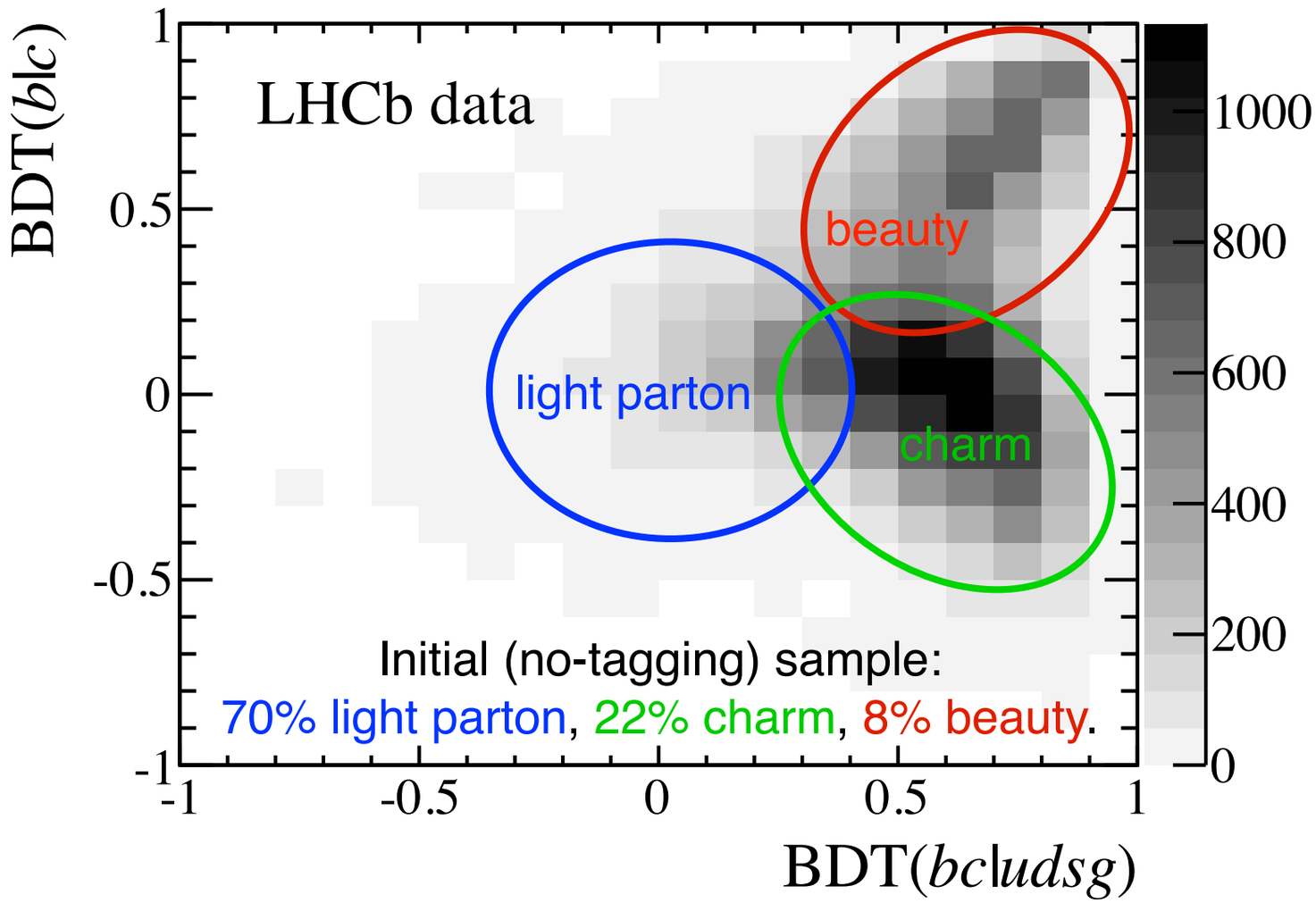


See proposed search in Ilten, Soreq, Thaler, MW, Xue, PRL 116, 251803 (2016) [1603.08926].

ML Jet Tagging

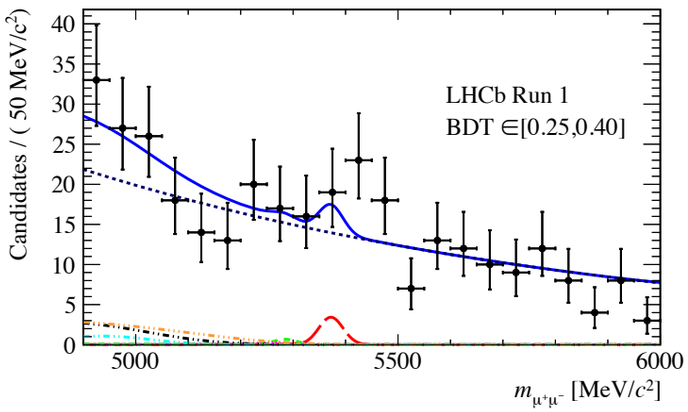
JINST 10 (2015) P06013
LHCb-PAPER-2015-016

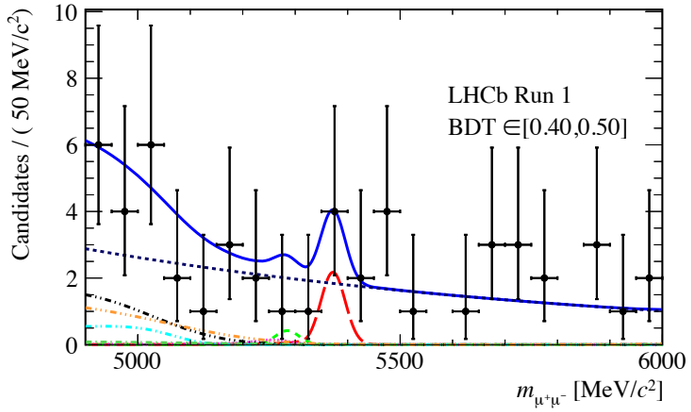
2-D BDT plane (nearly) optimally utilizes 10-D info to ID b, c, and light jets.

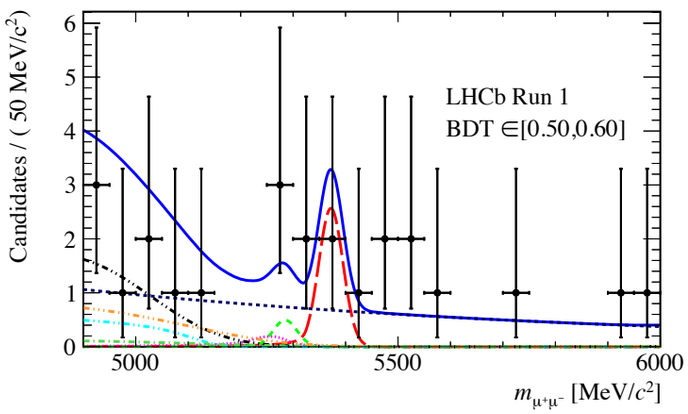


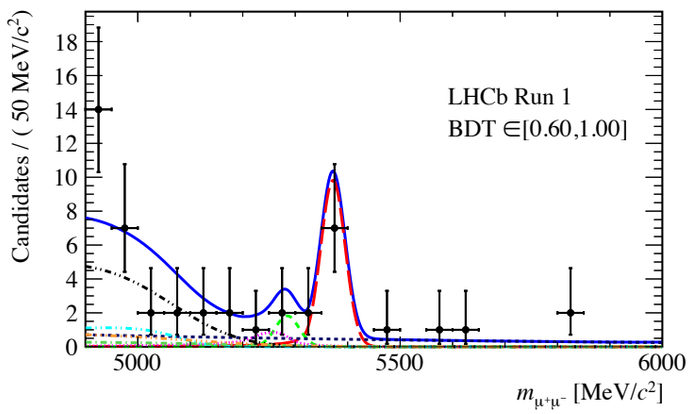
Performance validated & calibrated using large heavy-flavor-enriched jet data samples (2-D data validation much easier than 10-D!). Some analyses cut on these BDT responses, others fit the 2-D distributions to extract b,c,l yields.

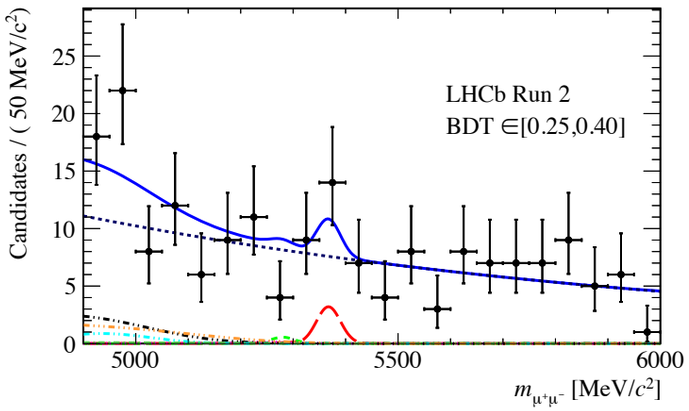
ML in Analysis

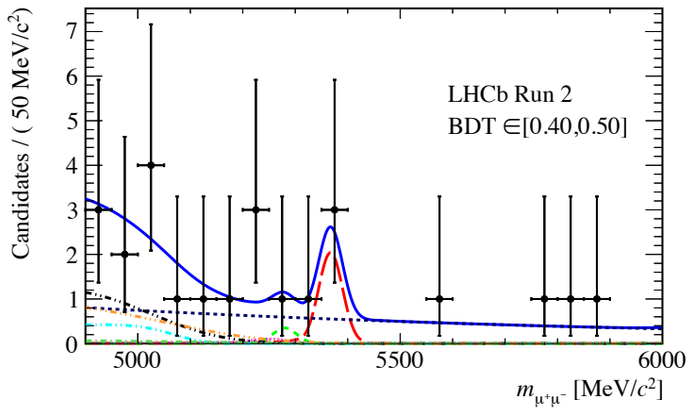


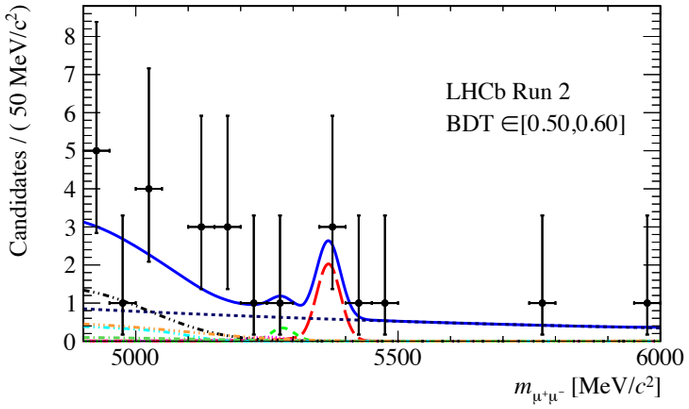


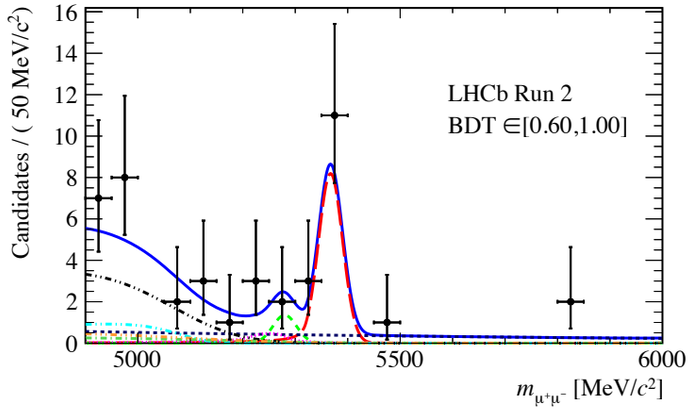












LHCb-PAPER-2017-001

Continuing to move beyond just cutting on response … Calibrate BDT to have uniform response on $B_s \to \mu\mu$ signal, bin data in BDT response and analyze all dimuon mass distributions simultaneously.

Constraints added to the likelihood for relationships between yields and shapes of the various components from bin to bin.

Details

•We typically train our ML algorithms on MC, then characterize their performance using data control samples (same way we characterize our hardware). In principle, data samples could also be used in the training, but then one would need to deal with BKGD in those samples (and wait for data to be taken to do the training).

•Dimensional reduction achieved by ML makes it possible to maximize performance without complicating data-driven validation. There are many standard candles at the LHC to use for data-driven validation.

•It's vital to collect the data samples required for calibration in the trigger! Typically this means some tag-and-probe control modes, where the response is calculated and stored but not cut on.

•As an aside, systematics tend to scale with inefficiency, so a highly-performant black box often incurs a smaller systematic than a simple, less performant algorithm — and also is easier to deal with than hardware.

•Bottom line: We use ML because it enables great science. It greatly improves performance in many areas, even converting some measurements from infeasible to simple & precise.

Tools, etc.

•LHCb uses a python API to configure C++ objects, i.e. everybody writes in python, experts write the C++. The code is versioned in git, and managed in gitlab.

•ML algorithms used to be mostly ROOT's TMVA , but are now migrating more and more to scikit-learn, Keras, etc.; i.e., we are moving away from physics-specific software and towards the tools used by the wider ML community. Hyper-parameter tuning using spearmint, hyperopt, etc. (see also Ilten, MW, Yang [1610.08328]).

•Custom loss functions, e.g., response is de-correlated from some set of features (Stevens, MW [1305.7248]; Rogozhnikova, Bukva, Gligorov, Ustyuzhanin, MW [1410.4140]). Already used in several papers (e.g. LHCb, PRL 115 (2015) 161802), and currently being used in many papers to appear soon.

•Many useful tools provided in the HEP-ML package pypi.python.org/pypi/hep_ml/0.2.0, which is basically a wrapper around sklearn, and in REP https://github.com/yandex/rep (both produced by our colleagues at Yandex).

Third Machine Learning in High Energy Physics Summer School 2017

17-23 July 2017
Reading
Europe/London timezone

Search...

Support

The Third Machine Learning summer school organized by Yandex School of Data Analysis, Laboratory of Methods for Big Data Analysis of National Research University Higher School of Economics and Imperial College London will be held in Reading, UK from 17 to 23 July 2017.

The school is intended to cover the relatively young area of data analysis and computational research that has started to emerge in High Energy Physics (HEP). It is known by several names including “Multivariate Analysis”, “Neural Networks”, “Classification/Clusterization techniques”. In more generic terms, these techniques belong to the field of “Machine Learning”, which is an area that is based on research performed in Statistics and has received a lot of attention from the Data Science community.

There are plenty of essential problems in High energy Physics that can be solved using Machine Learning methods. These vary from online data filtering and reconstruction to offline data analysis.

Students of the school will receive a theoretical and practical introduction to this new field and will be able to apply acquired knowledge to solve their own problems. Topics ranging from decision trees to deep learning and hyperparameter optimization will be covered with concrete examples and hands-on tutorials. A special data-science competition will be organized within the school to allow participants to get better feeling of real-life ML applications scenarios.

Expected number of students for the school is 50-60 people

Summary

Real-time calibration works, moving to a triggerless readout will provide even bigger gains, ML usage is ubiquitous — all of these enable great science!