Advanced Machine-Learning Solutions in LHCb Operations and Data Analysis

Fedor Ratnikov

National Research University High School of Economics,
Yandex School of Data Analysis,

on behalf of LHCb Collaboration

Fermilab, July 31 - August 4, 2017
Content

- Introduction
- ML in Trigger
- ML in PID
- ML in Monitoring
LHCb Detector @LHC

◊ One of four big LHC detectors

◊ Primary physics program: charm and beauty physics

◊ need excellent track reconstruction and particle ID

ML in LHCb Trigger

◇ Use real time alignment, calibration and reconstruction in Run II
   ◇ need RECO online ⇔ RECO offline

◇ Use BDT and/or NN in
  ◇ track reconstruction
  ◇ fake tracks suppression
  ◇ particle ID
  ◇ High Level Trigger decisions
BDT Speedup

◊ Strong requirements to the speed of the code which runs in real time

◊ Out of the box ML solution usually produce heavy classifiers
  ◊ need speed up / simplify

◊ Two main approaches to speed up BDT
  ◊ post-pruning
    ◊ remove less important branches/trees
    ◊ reduce number from thousands to hundred
    ◊ trading speed for precision
  ◊ Bonsai BDT (BBDT)
    ◊ discretisation of input features
      ◊ converts trees into Look Up Table
      ◊ trading speed for memory
      ◊ limited by ~1Gb

arXiv:1210.6861 [physics.ins-det]
Long Tracks Reconstruction

- In Run II apply NN-based clones and fakes rejection online
- Use two deep NN in forward tracking
  - for track candidate selection after stereo fit
    - 16 Input nodes, 17,9,5 HL nodes
  - for rejection of bad 4-layer clusters
    - 9 Input nodes, 16,10 HL nodes
Downstream Tracking

- Downstream tracking contains two Multivariate classifiers
  - reject as much fake T-seeds as possible: avoiding unnecessary reconstruction
    - BBDT
  - final accepting tracks: further reducing fake tracks
    - MLP

- Improved both fake tracks reduction and signal efficiency gain by 3-5%
  - implemented for 2017 operation
Topological Trigger

- HLT-1 track is looking for either one super high $p_T$ or high displaced track
- HLT-1 2-body SV classifier is looking for two tracks making a vertex
- HLT-2 improved topological classifier uses full reconstructed event to look for 2, 3, 4 and more tracks making a vertex
  - it’s all about finding Secondary Vertices
- most of LHCb Run I papers were made using this trigger
- We aggressively re-optimised this trigger for Run II operation

![Diagram showing HLT-1 and HLT-2 triggers with event vertices and rates: HLT-1 track: 100 kHZ, HLT-1 2-body SV: 50 kHZ, HLT-2 Topo: 2-4 kHZ]
Compare Different Classifiers

- NN: neural net
- MN: Yandex’s MatrixNet (a BDT)
- Logistic Regression

Compare efficiency at given rate

NN and MN are close, but MN usually wins

arXiv:1510.00572 [physics.ins-det]
Aggregated Metric

- Need to have aggregative metric to measure quality
- Weight signal events in such a way that decays have the same sum of weights
- Optimise ROC curve in a region with small FPR
Speed up and Improvements

- Need to speed up trees
  - compare BBDT with post-pruning
    - post-pruning is slightly better

- Gain 10%..70% efficiency for different channels

\[ \varepsilon_{\text{HLT}}(\text{Run II}) / \varepsilon_{\text{HLT}}(\text{Run I}) \]

<table>
<thead>
<tr>
<th>mode</th>
<th>2.5 kHz</th>
<th>4.0 kHz</th>
</tr>
</thead>
<tbody>
<tr>
<td>( B^0 \to K^*[K^+\pi^-]\mu^+\mu^- )</td>
<td>1.64</td>
<td>1.72</td>
</tr>
<tr>
<td>( B^+ \to \pi^+K^-K^+ )</td>
<td>1.59</td>
<td>1.65</td>
</tr>
<tr>
<td>( B_s^0 \to D_s^-[K^+K^-\pi^-]\mu^+\nu_{\mu} )</td>
<td>1.14</td>
<td>1.47</td>
</tr>
<tr>
<td>( B_s^0 \to \psi(1S)[\mu^+\mu^-]K^+K^-\pi^+\pi^- )</td>
<td>1.62</td>
<td>1.71</td>
</tr>
<tr>
<td>( B_s^0 \to D_s^-[K^+K^-\pi^-]\pi^+ )</td>
<td>1.46</td>
<td>1.52</td>
</tr>
<tr>
<td>( B^0 \to D^+[K^-\pi^+\pi^+]D^-[K^+\pi^-\pi^-] )</td>
<td>1.40</td>
<td>1.86</td>
</tr>
</tbody>
</table>
Multi-Class Charge Particles ID

- Problem: identify charged particle associated with a track (multiclass classification problem)

- particle types: Ghost, Electron, Muon, Pion, Kaon, Proton.

- LHCb detector provides diverse information, collected by subdetectors: CALO, RICH, Muon and Track observables

- this information can be efficiently combined using ML

- Monte Carlo simulated samples for various decays are available (6 millions tracks in training and 6 millions in test)
Particle ID Approaches: $\Delta \log L$

- Separate classification by subsystems (no MVA)
  - convert particle response in the subsystem for likelihood of particular ID hypothesis
  - two RICH detectors
  - calorimeters: pre-shower, ECAL, HCAL
  - muon systems
- combine likelihood of different subsystems into global likelihoods of different ID hypothesis
  - $L(id) = L_{\text{RICH}}(id) \times L_{\text{Calo}}(id) \times L_{\text{Muon}}(id)$

Combined Particle ID, ProbNN

- Combined one-class classification
  - information from all subsystems is analysed together using neural network with one hidden layer (TMVA MLP)
  - classifiers “this ID” vs “any other ID” are optimised for every hypothesis
  - best established approach, mostly used by current physics analyses

Further Approaches: Baseline

- Work on further improvement of the particle identification quality
- Consider One-class approach (ProbNN) as a baseline
  - consists of 6 binary classification models: One-vs-Rest
  - good separation quality
    - ROC AUC \(\sim 0.91 \ldots 0.99\) for different hypothesis
    - statistical AUC uncertainties for used simulated samples are \(\sim 0.0002\)
- Consider \((1 - \text{ROC AUC})\) for the same ID as a metric for quality comparison
Combined Multi-class Classification

- Binary classification is widespread
  - major ML use-case in HEP
  - binary classification is very simple to implement
    - some algorithms are defined only for binary classification
- XGBoost, CatBoost, sklearn and most neural networks implementations support multi-class implementations
- Combined multi-class PID
  - information from all subsystems is analysed together using MVA
  - classifier gives weight to each hypothesis, $\sum w(id) = 1$
  - this is new approaches being tested on MC
The number of weights (parameters) in Neural Network is similar for binary classification and multi-class classification.

Computationally multi-class classification has (almost) the same complexity as binary classification (unless there are hundreds of classes and more).

<table>
<thead>
<tr>
<th></th>
<th>Ghost</th>
<th>Electron</th>
<th>Muon</th>
<th>Pion</th>
<th>Kaon</th>
<th>Proton</th>
</tr>
</thead>
<tbody>
<tr>
<td>baseline</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>deep NN</td>
<td>-29 %</td>
<td>-41 %</td>
<td>-52 %</td>
<td>-37 %</td>
<td>-20 %</td>
<td>-17 %</td>
</tr>
</tbody>
</table>

LHCb Simulation, preliminary
BDT

- Multi-class with XGBoost
- Multi-class with CatBoost

<table>
<thead>
<tr>
<th></th>
<th>Ghost</th>
<th>Electron</th>
<th>Muon</th>
<th>Pion</th>
<th>Kaon</th>
<th>Proton</th>
</tr>
</thead>
<tbody>
<tr>
<td>baseline</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>deep NN</td>
<td>-29%</td>
<td>-41%</td>
<td>-52%</td>
<td>-37%</td>
<td>-20%</td>
<td>-17%</td>
</tr>
<tr>
<td>XGBoost</td>
<td>-24%</td>
<td>-37%</td>
<td>-50%</td>
<td>-34%</td>
<td>-18%</td>
<td>-15%</td>
</tr>
<tr>
<td>CatBoost</td>
<td>-30%</td>
<td>-43%</td>
<td>-54%</td>
<td>-37%</td>
<td>-20%</td>
<td>-18%</td>
</tr>
</tbody>
</table>

LHCb Simulation, preliminary

\[
\frac{(1 - \text{AUC})}{(1 - \text{AUC}_{\text{baseline}})}
\]
Flatness

- The whole PID information strongly depends on particle momentum, that leads to strong dependency between PID efficiency and momentum.

- In some analyses we need to have flat PID along signal particle momentum.

- Requirement for flatness over a set of variables may be added into loss function for training.
  
  - by price of some quality degradation
  
Flat Models

Uniform boosting provides flatness along 4 variables at once

- $p$
- $p_T$
- $\eta$
- $n$ Tracks

ML in LHCb
Flat Models Quality

- Flatness is a very strong restriction, holding this restriction leads to quality decreasing.

- Still, flat model looks preferable comparing to baseline in ROC AUCs

<table>
<thead>
<tr>
<th></th>
<th>Ghost</th>
<th>Electron</th>
<th>Muon</th>
<th>Pion</th>
<th>Kaon</th>
<th>Proton</th>
</tr>
</thead>
<tbody>
<tr>
<td>baseline</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td>I</td>
</tr>
<tr>
<td>p + p_T flatness</td>
<td>-23 %</td>
<td>-20 %</td>
<td>-27 %</td>
<td>-26 %</td>
<td>+2 %</td>
<td>+5 %</td>
</tr>
<tr>
<td>2d(p, p_T) flatness</td>
<td>-21 %</td>
<td>-9 %</td>
<td>-13 %</td>
<td>-23 %</td>
<td>+12 %</td>
<td>+23 %</td>
</tr>
<tr>
<td>p + p_T + η + nTracks flatness</td>
<td>-22 %</td>
<td>-13 %</td>
<td>-26 %</td>
<td>-24 %</td>
<td>+2 %</td>
<td>+6 %</td>
</tr>
<tr>
<td>4d(p, p_T, η, nTracks) flatness</td>
<td>-21 %</td>
<td>-4 %</td>
<td>-13 %</td>
<td>-20 %</td>
<td>+10 %</td>
<td>+25 %</td>
</tr>
</tbody>
</table>

LHCb Simulation, preliminary

(1-AUC)/(1-AUC_{baseline})
Monitoring Robo-shifter

- Robo-shifter is machine-learning based system designed to assists the DQ shifter
- Given run data it can predict probability of run being good or bad
- Hint for potential problem sources is extracted from decision trees
- In process of commissioning for Data Quality Monitoring
Conclusions

◊ These are few highlights of successful or encouraging use of ML approaches in crucial points of the LHCb detector operation and data processing

◊ There are many other uses

◊ boosting physics analyses

◊ optimise data processing and data handling

◊ neutral particles identification

◊ ...

◊ Different ML techniques are on the rise

◊ LHCb follows this development and uses the best to ultimately improve our physics results
### Backup: Forward Tracking

<table>
<thead>
<tr>
<th>MC performance</th>
<th>$\nu = 1.6$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2016 w.r.t. 2015</td>
<td>w/ RL</td>
</tr>
<tr>
<td>timing HLT1</td>
<td>$\pm 0%$</td>
</tr>
<tr>
<td>timing HLT2</td>
<td>$+4%$</td>
</tr>
<tr>
<td>fake rate</td>
<td>$-27%$</td>
</tr>
<tr>
<td>fake rate HLT1</td>
<td>$-15%$</td>
</tr>
<tr>
<td>$\varepsilon$ long</td>
<td>$+0.5%$</td>
</tr>
<tr>
<td>$\varepsilon$ long from B</td>
<td>$+0.2%$</td>
</tr>
<tr>
<td>$\varepsilon_{HLT1}$ long from B $p &gt; 3, p_T &gt; 0.5$ GeV</td>
<td>$+0.1%$</td>
</tr>
</tbody>
</table>

[Diagram showing VELO, TT, magnet, T stations, downstream track, upstream track, and long track.]