Particle decay classifiers for the LHCb Run 3 first-level trigger

Presented by: Sean Condon - MIT

Mentors: Michael Williams, Thomas Boettcher, Daniel Craik - MIT

28 September 2020

Simulated LHCb Data allows Training of Classifiers

-**The task:** update the one-track and two-track high level triggers for run 3 of LHCb, implement optimal triggers into Allen pipeline

-**Training data:** 180,000 simulated LHCb events (~2 million decays), made up of 6 decay types and one 2018MinBias file for trigger rate calculation

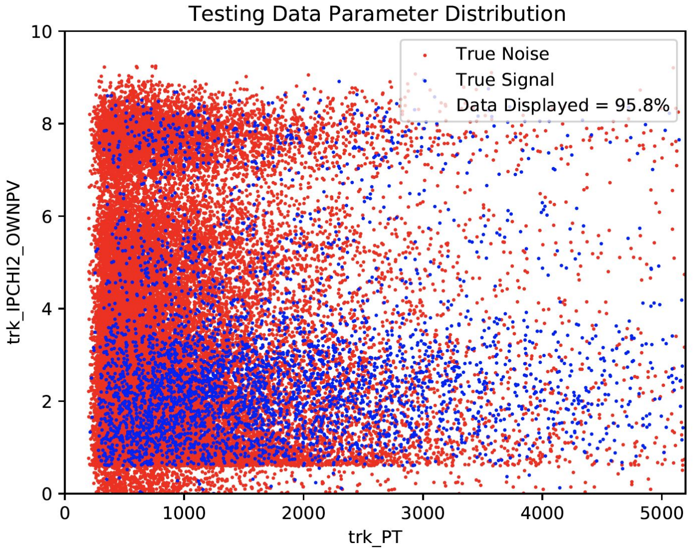
-Cut out signals with low momentum and lifetime: parent particle PT > 2 GeV, tau > 0.2 ps, $2 < \eta < 5$ (removes ~92% of data)

Training Data Cont.

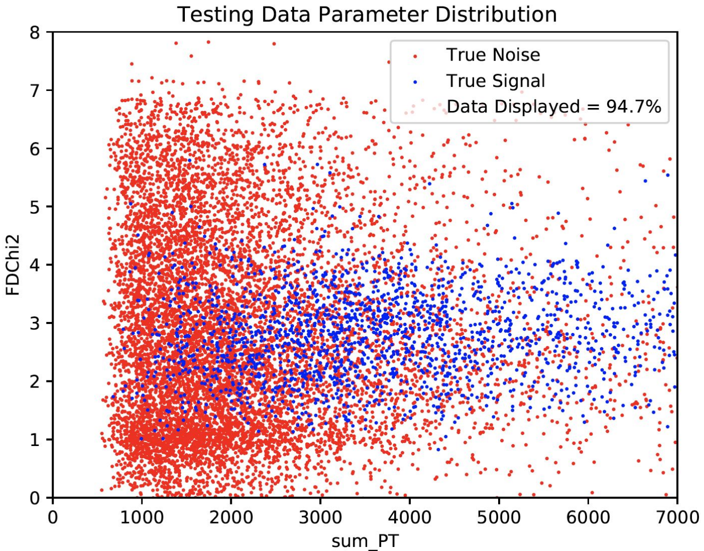
-Classifier inputs are taken from *LHCb Trigger Reoptimization* paper†

Classifier	Analysis Variables
1-Track Model	PT, IP_{χ^2}
2-Track Model	sum PT, vertex χ^2, FD χ^2, N (# tracks with $IP_{\chi^2} < 16$)

One-Track Data



Two-Track Data



Catboost Enables Easy Training of BDT Models

-Catboost is a python library for Boosted Decision Tree classifiers, enabling:

-A variety of training options for BDTs

-Saving classifiers in JSON format easily read into Allen pipeline

-Total simulated LHCb data is split into 75% training, 25% testing

-Training data 92% noise, so noise decays are weighted to balance training

-BDT model hyperparameters (learning rate, model depth, model iterations) optimized by exploring parameter space.

Notes on Classifier Evaluation

-Catboost Classifiers are trained on single decay examples, but they are evaluated on LHCb events (average ~10 decays) for efficiency and trigger rate

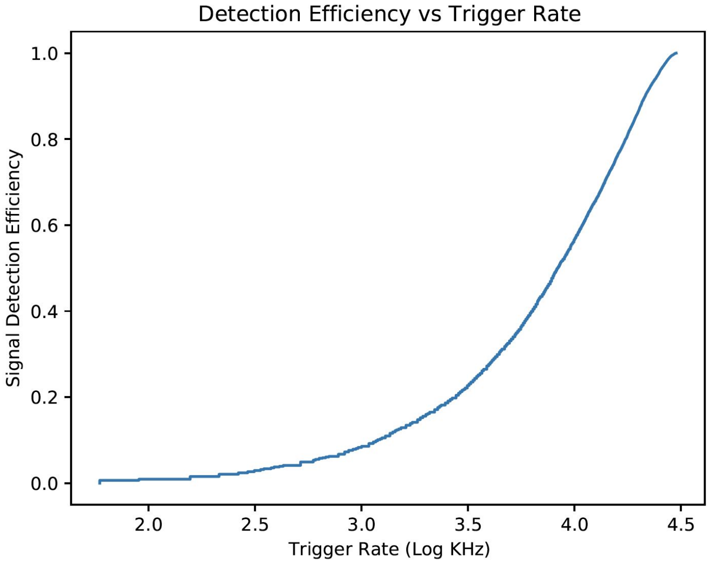
-For each decay, output = softmax-ed vector in the format of [$\text{Prob}_{\text{noise}}$, $\text{Prob}_{\text{signal}}$]

-If any decay in an evaluation event has $\text{Prob}_{\text{signal}}$ > threshold, the entire event is considered to be triggered on

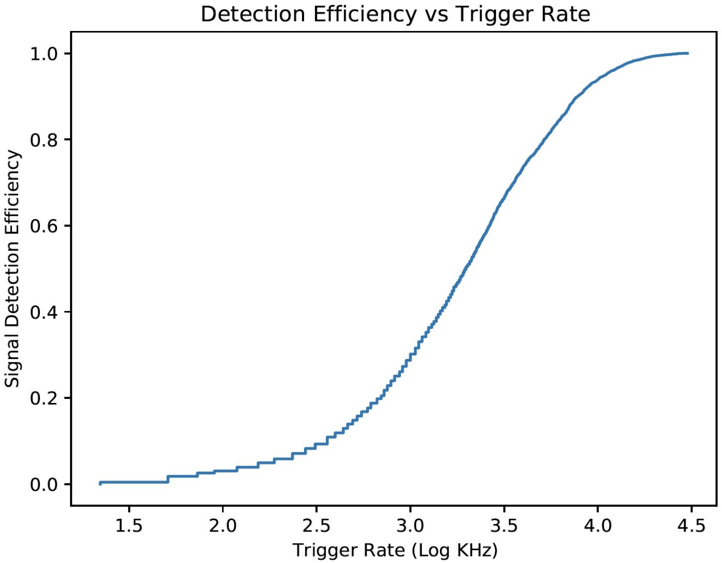
-True positives when event has signal, AND classifier triggers on signal decay

Performance of Preliminary Classifiers

One-Track: eff = 13.7% at 1 MHZ



Two-Track: eff = 40.7% at 1 MHZ



Addressing Decay Type Imbalance

-Training datasets are very imbalanced by decay type

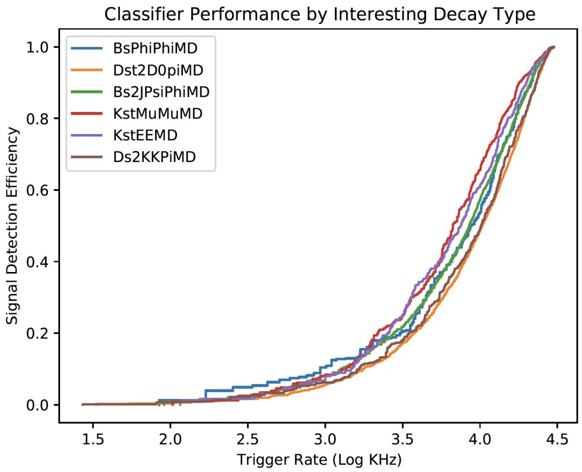
N_{large} = 95,000 (Bs2JPsiPhi); N_{small} = 2,470 (Ds2KKPi)

-Downsample all decay datasets to have N_{small} events

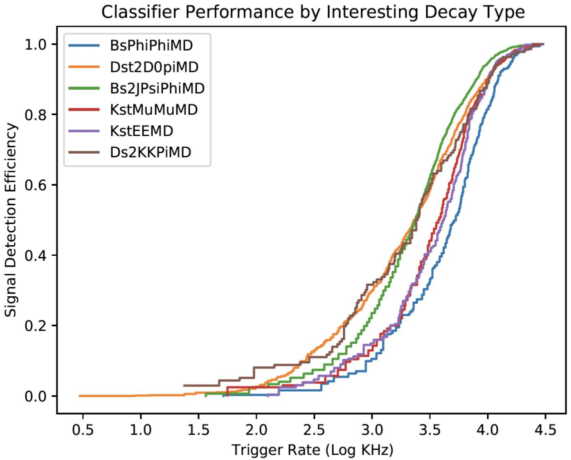
-To avoid low training data, instead downsample all decay datasets to upsamplingDegree*N_{small} events

-upsamplingDegree was varied between 1-10, optimal performance (AUC ROC) at 7

One-Track



Two-Track



Additional Improvements on Classifiers

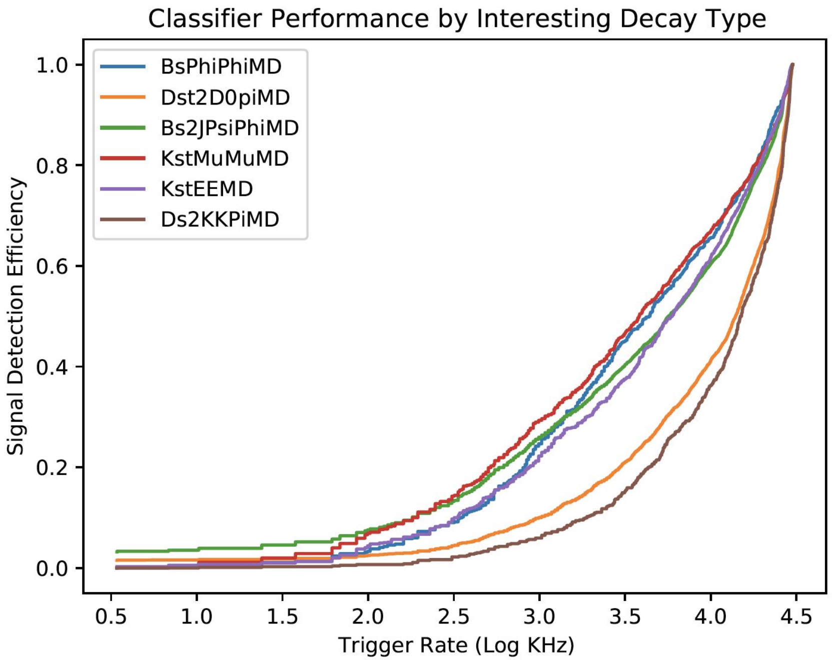
-Shuffling of training data helped to prevent overtraining on certain decay types

-Re-scaling parameters improved performance, namely putting all χ^2 input variables on a log scale before training

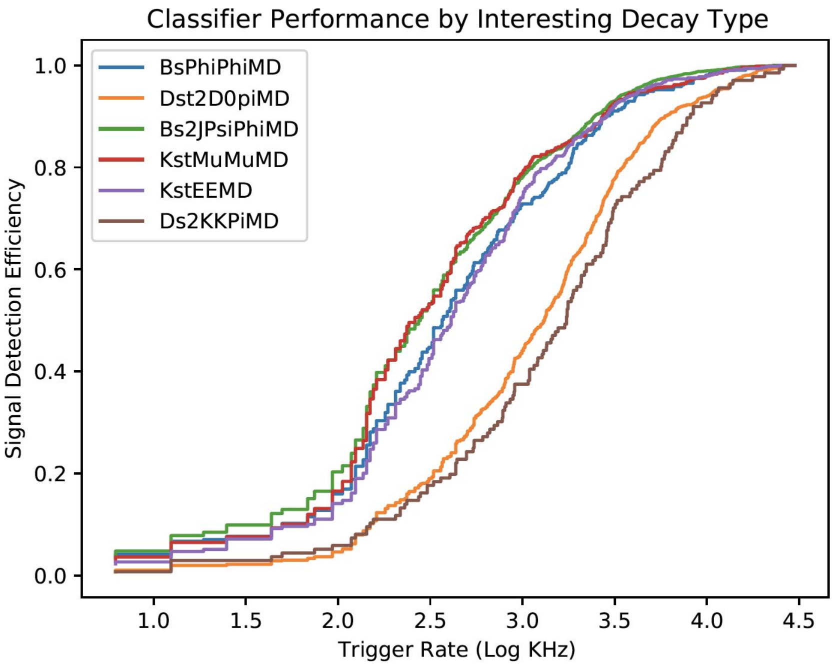
-Improvements increased efficiency several percentage points each at constant trigger rate

Performance of Improved Classifiers

OneTrack: eff = 28.1% at 1 MHZ



TwoTrack: eff = 70.7% at 1 MHZ



Note: lower efficiency modes are charm

Comparison with Cut-based classifiers

-Two-track classifier shows significant improvement over cut-based classifier

-One-track classifier shows slight improvement over cut-based classifier

-One-track cut-based classifier is a function fit to the response of an ML selection

Data Type	Classifier	Signal Detection Efficiency (at same FPR)
1-Track	Cut-Based	21.4%
	BDT	24.4%
2-Track	Cut-Based	60.4%
	BDT	75.1%

Implementation of Models into Allen

-Late July and early August were spent writing C++ code to implement one-track and two-track models in Allen Pipeline for LHCb run 3

-Two corresponding new algorithms, one new sequence

-Secured an LHCb computing account, merge request coming soon

-Infrastructure to make implementing additional Catboost classifiers easier

Experimenting with 3-Track and 4-Track Triggers

- Classifier inputs are taken from *LHCb Trigger Reoptimization* paper†

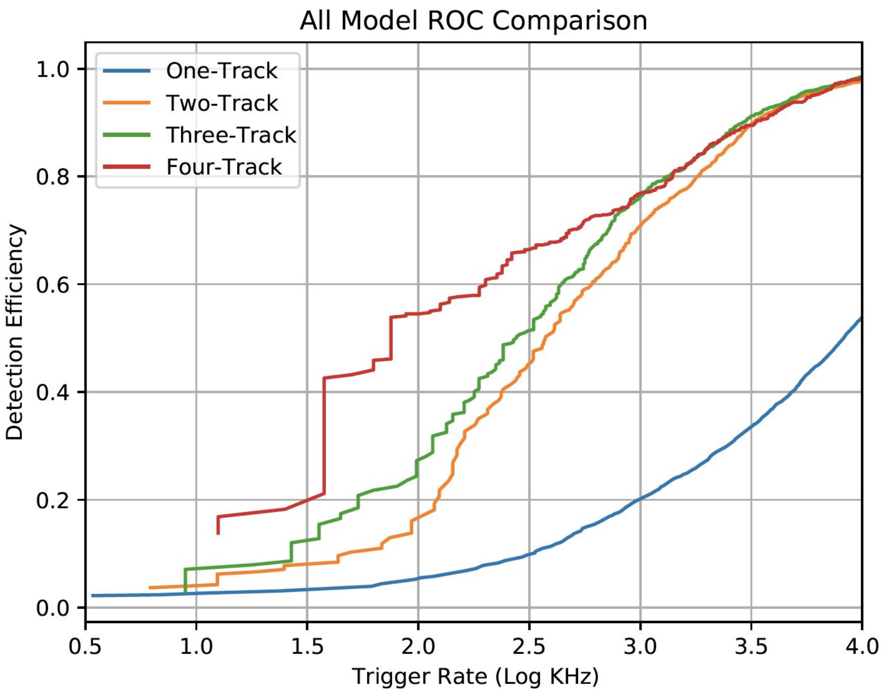
Analysis variables:
n, mcor, sum PT, $vertex_{\chi^2}$, η, $FD_{\chi 2}$, min PT, IP_{χ^2}, N(tracks), N(tracks with $IP_{\chi^2} < 16$)

-In run 2, these selections could only be done in HLT2. Allen tracking in run 3 is so fast that these selections can now be made in HLT1.

-3-Track, 4-Track models trained separately, all D decays excluded

-Model hyperparameters were optimized by exploring parameter space

Performance of All Four BDT Models



Note: in actual analysis, use an OR selection on all of these lines to improve efficiency.

Summary of Work Completed

1. Optimized 1-track and 2-track BDT classifiers
2. Implemented optimal 1-track and 2-track models into Allen
3. Demonstrated efficiency of 3-track and 4-track classifiers

Next steps: -Submit merge request for new Allen algorithms

-Implement 3-track / 4-track into Allen pipeline