Extending the Physics Reach of LHCb in Run 3 Using Machine Learning in the Real-Time Data Ingestion and Reduction System

R. Fang [1], H. Schreiner [1,3], M. D. Sokoloff [1], M. Stahl [1], C. Weisser [2], M. Williams [2]

University of Cincinnati [1], MIT [2], Princeton University[3]

Abstract

- LHCb's High Level Trigger will process 5 TB/s of data. Machine learning algorithms have the potential to improve fidelity and execute very quickly.
- The first stage (Hlt1) will process approximately 30 MHz of events.
- The second stage (Hlt2) will process approximately 1 MHz of events.
- We are developing a hybrid deep learning algorithm to identify primary and secondary vertices in *pp* collisions.

Supported by NSF awards OAC-1740102 & OAC-1739772 and via sub-awards under Cooperative Agreement OAC-1836650.

The Run 3 LHCb Detector & Baseline Trigger

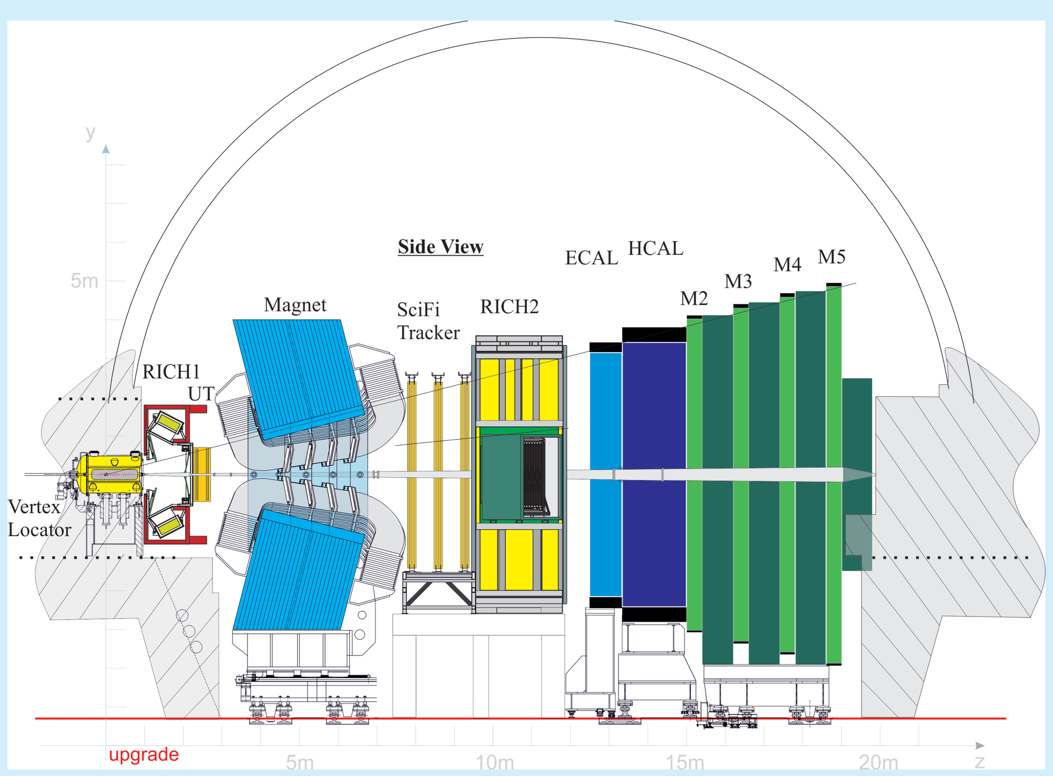


Figure 1: LHCb detector schematic. Charged tracks are reconstructed using data collected in the Vertex Locator (VELO) and 4 additional tracking stations (UT, T1–T3). LHCb is ~ 20 m long, 10 m high.

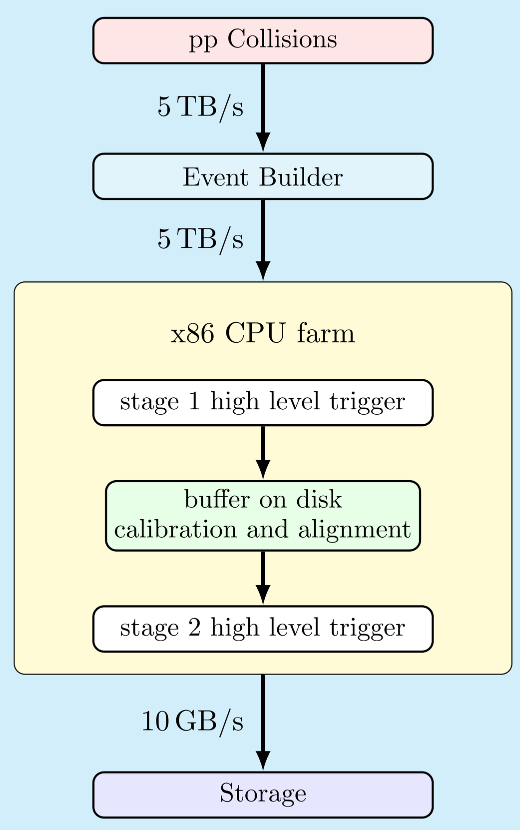
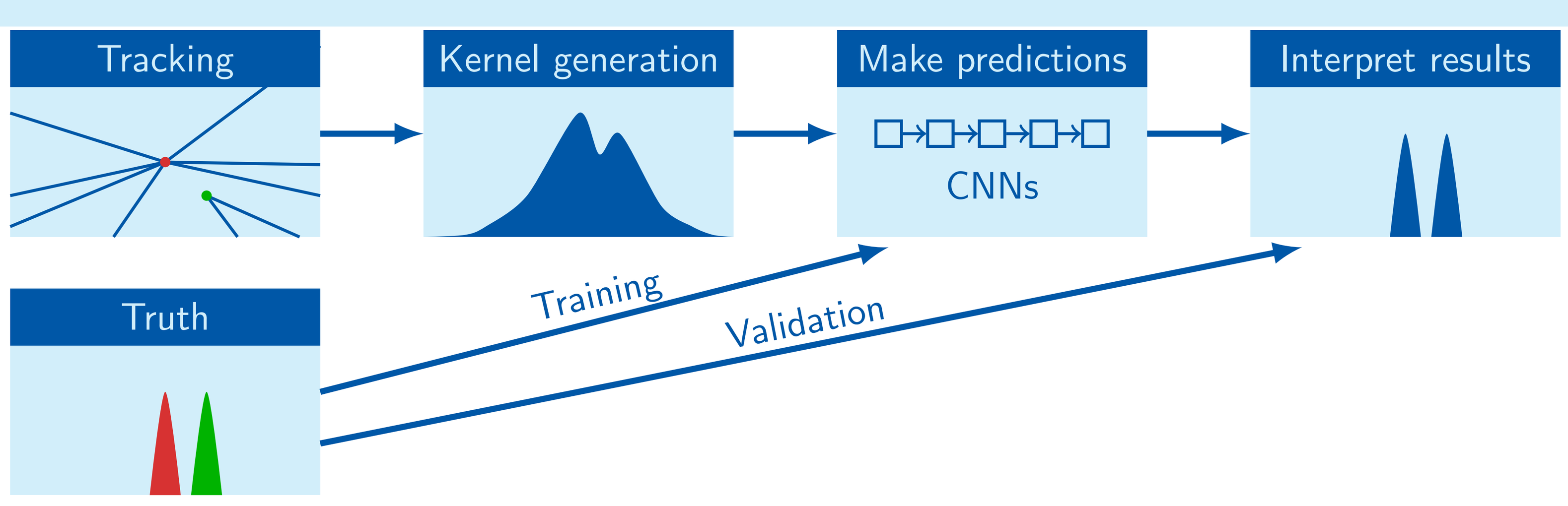


Figure 2: CPU Trigger Schematic. A GPU option for Hlt1 has been demonstrated, as well.

A hybrid ML approach to finding primary vertices



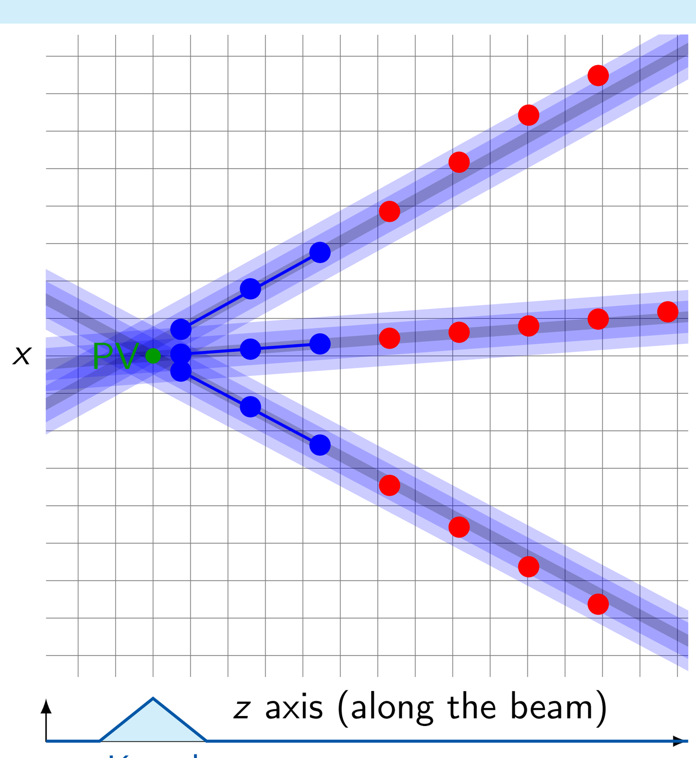
Machine learning features (so far)

- Prototracking converts sparse 3D dataset to feature-rich 1D dataset
- Easy and effective visualization due to 1D nature
- Even simple networks can provide interesting results

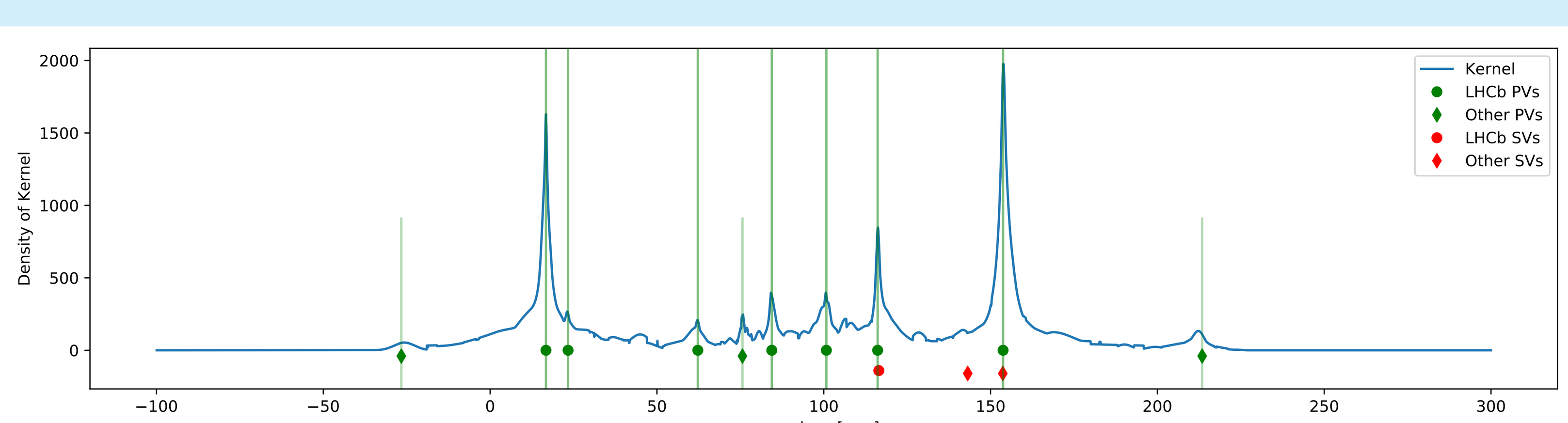
Kernel generation (41M pixels $\rightarrow$ 4K histogram entries)

Tracking procedure

- Hits lie on the 26 planes
- For simplicity, only 3 tracks shown
- Make a 3D grid of voxels (2D shown)
- Note: only z will be fully calculated and stored
- Tracking (full or partial)
- Fill in each voxel center with Gaussian PDF
- PDF for each (proto)track is combined
- Fill z "histogram" with xy value at max KDE



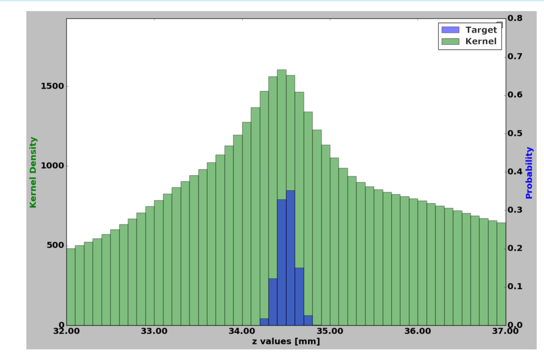
A typical kernel (4000 $\times$ 100 μm bins)

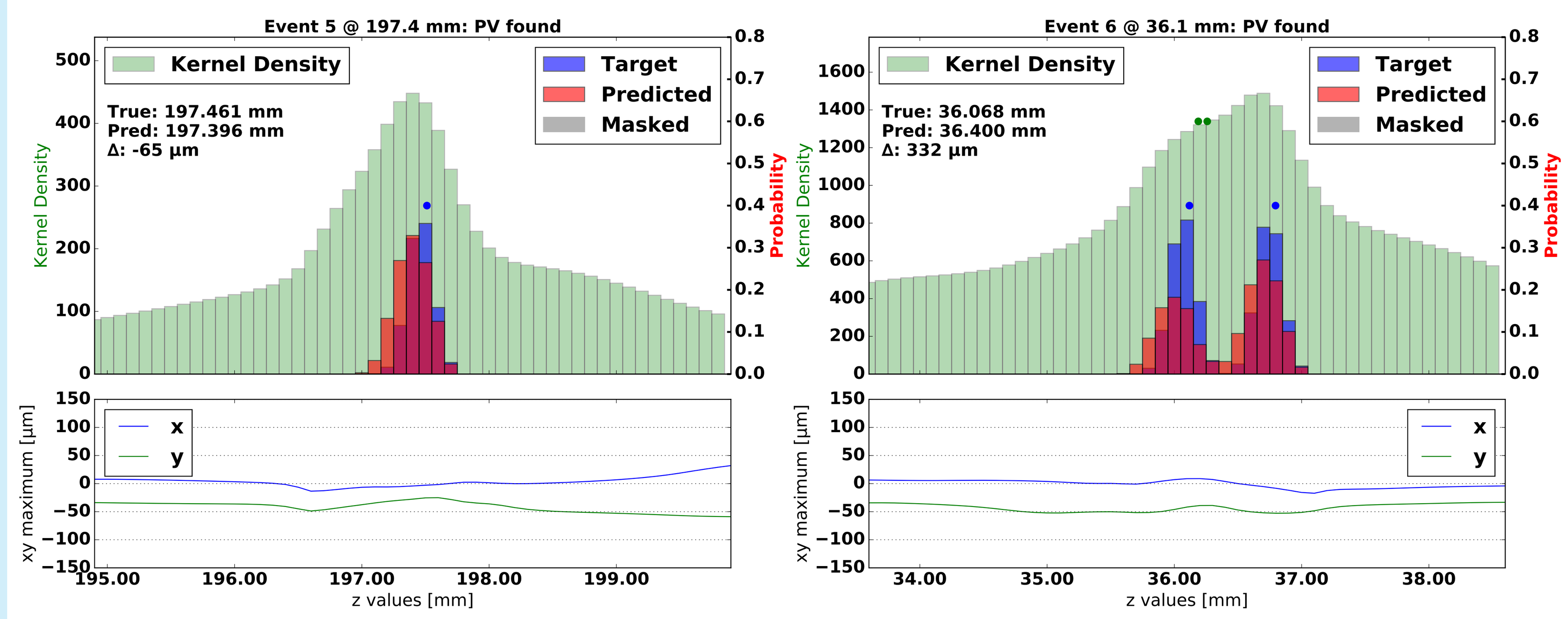


Target histograms as proxies to learn (circa early 2019)

Build target distribution

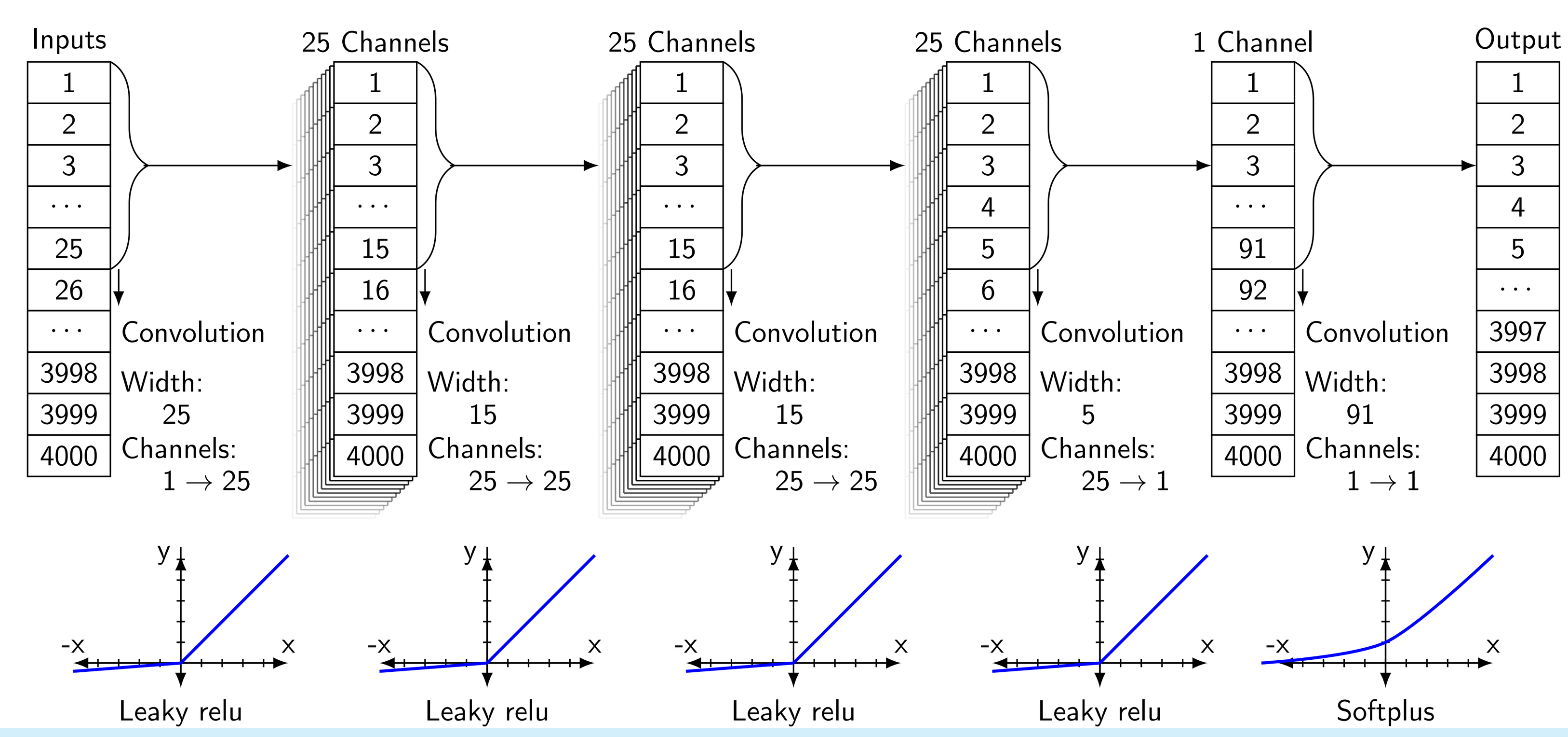
- True PV position as the mean of Gaussian
- σ (standard deviation) is 100 μm (simplification)
- Fill bins with integrated PDF within ± 3 bins ($\pm 300\,\mu$m)



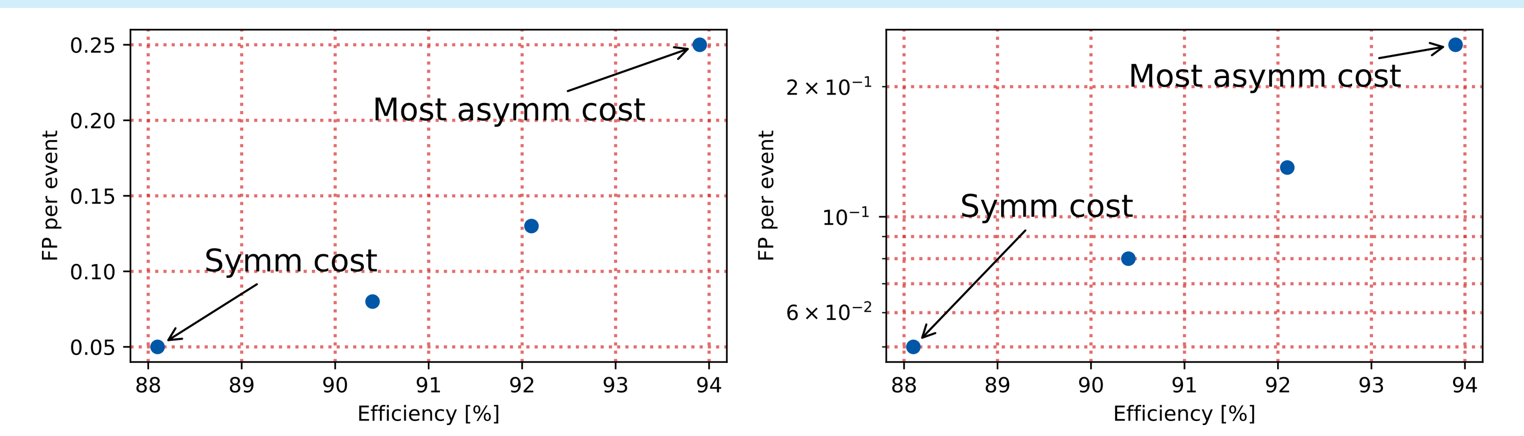


PV found example | PV found example

Architecture (circa early 2019) [implemented using PyTorch]



Tune efficiency vs. false positive (FP) rate using cost function



Search for PVs (handwritten, maybe not optimial)

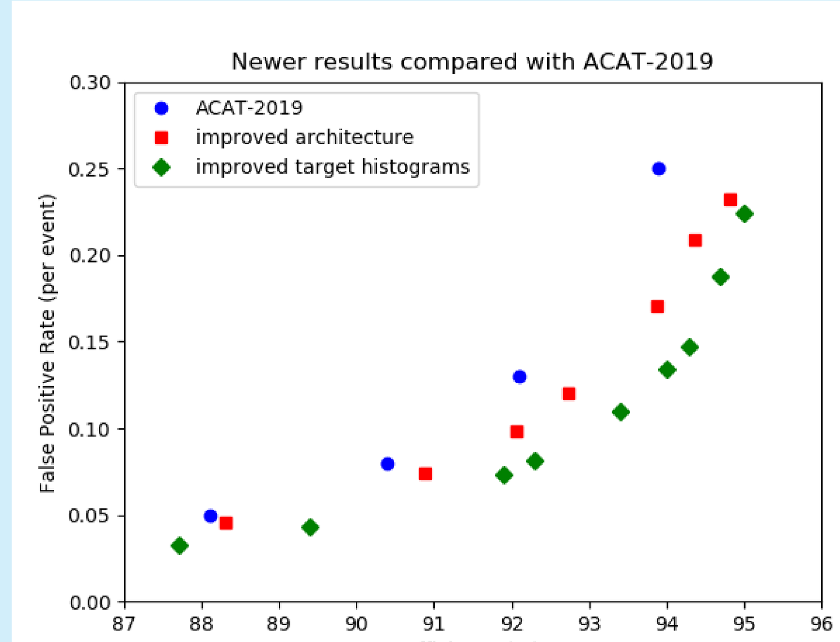
- Search ± 5 bins ($\pm 500\mu$m) around a true PV
- At least 3 bins with predicted probability $> 1\%$ and integrated probability $> 20\%$.

Tunable efficiency vs. FP

- The asymmetry parameter controls FP vs. efficiency

More recent progress and future plans

- added xy features perturbatively, using a parallel CNN;
- added layers to original CNN;
- modified target histograms (learning proxies);
- tested inference engine on LHCb full simulation data;
- deployed inference engine in LHCb software stack.



- For a fixed efficiency of 94%, the false positive rate is about **2$\times$ smaller** than a year ago for the same toy MC data.
- We will **re-train** the algorithm **using full LHCb simulation** in place of toy simulation to improve performance.
- We will develop another machine learning algorithm to **learn the KDE directly from the tracks**, then combine the two algorithms into one.
- We develop an algorithm to **assign tracks to PVs** probabilistically.