“Usage of CUDA for improving HLT performance”

participant: Belytskyi Dmytro

Mentor: Andrii Usachov

Prerequisites

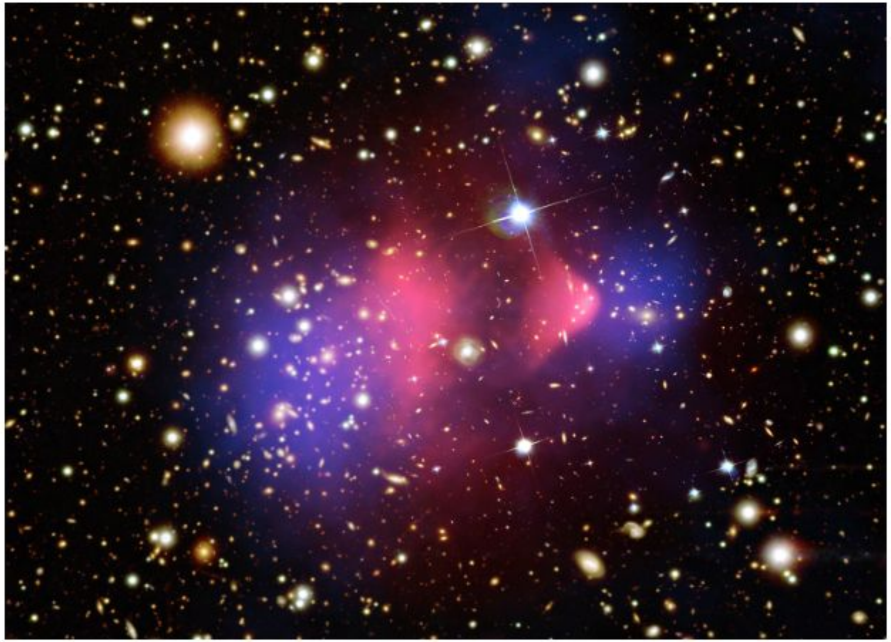


Figure 1: The "Bullet Cluster" (1E 0657-558). A combined image with data from the Chandra X-ray Observatory, the Magelland and Hubble space telescope, and gravitational lensing

So there we should attempt to expand of the Standard Model (SM), ensuring its alignment with observable data on cosmological scales

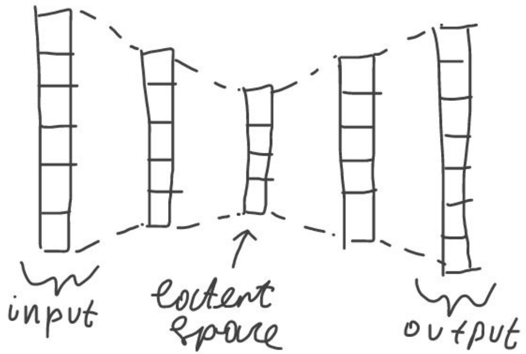
detecting anomalies in LHCb data with autoencoder.

Chain of thoughts

- The perspective place for search for possible candidates is the range of long lived particles.
- If “dark” particle is long-lived it should leave exotic trace at low level data.
- Thus the goal to search low-level data for anomalies

The general idea, how to detect anomalies, is simple : we will train autoencoder to reconstruct the background noise data.

Expectation is so that autoencoder will struggle to reconstruct the data with mixed in anomalies signal. Which appears in loss score

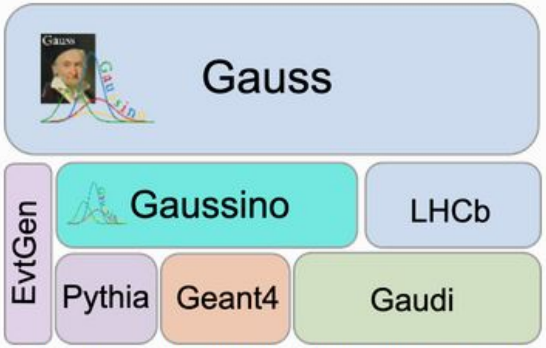


Data generation

Two samples where generated with gauss.

One is large one - background noise
And the other test sample - the one with mixed in higgs boson decay signal

The generated data stored into .xgen file format and further relevant parameters retrieved with Davinci



Gauss (-on-Gaussino) dependencies for Sim11 and beyond.

- Data consist from events
- Each event contain vertices(primary,secondary)
- Each vertices could be described by number of tracks, their angle, cartesian coordinates.

Data filtering

```
[array([[11.        , 44.27757402]]),
 array([[8.        , 1.17662957]]),
 array([[61.        , 71.11979942],
        [10.        , 19.23197502],
        [ 9.        , 32.10598961],
        [ 6.        ,  6.30813314]]),
 array([[21.        , 28.51872979],
        [16.        , 28.26696563]]),
 array([[25.        ,  2.25500172],
        [20.        , 20.91851244],
        [ 4.        , 40.17039385]]),
 array([[57.        , 71.00658684],
        [41.        , 24.55303106],
        [38.        , 19.27188189]]),
 array([[60.      ,  6.388345],
        [26.      , 25.521933]]),
 array([[22.        , 14.22079988],
        [12.        , 17.73504506],
        [11.        , 58.54394087]]),
```

So such cats were introduced to initial data vertices:

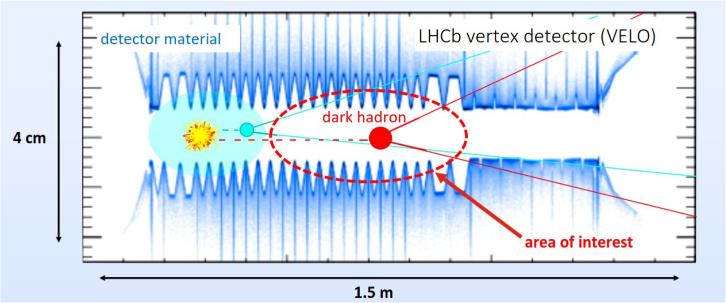
- cut by Velo detector geometry

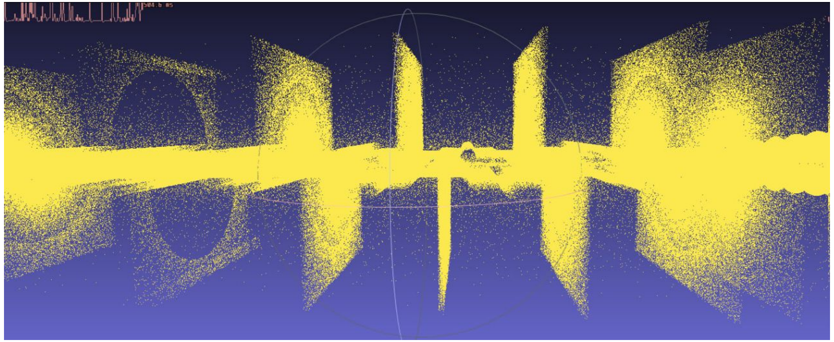
  $-400<Z<800$
  $X^2+Y^2<40^2$
  *all distances in mm

- Filter vertices which are related to interactions with detector meter

- Cut by number of tracks
  num_tracks>=3

Then all vertices in each event were sorted by num_tracks parameter in descending order.



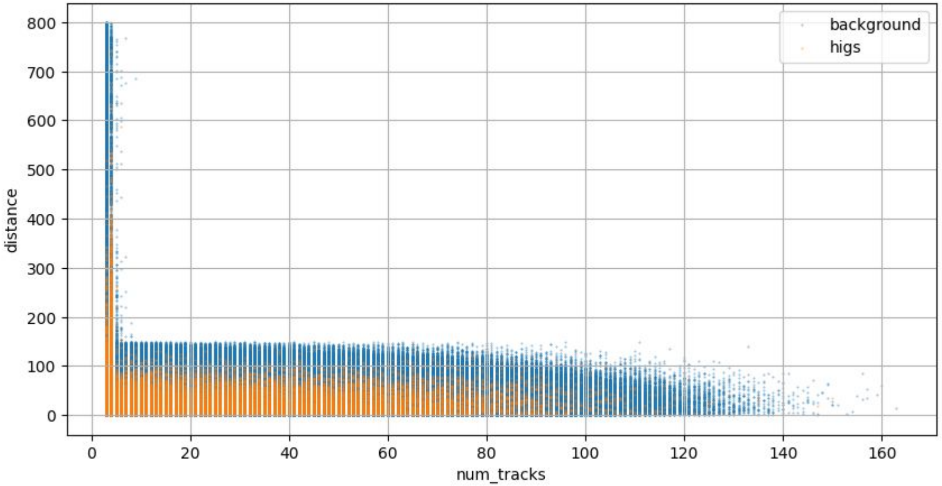


Vertices caused by interaction with detector

Autoencoder and data structure

The data shape was chosen from initial analysis.

Each event is described by the vector of length 12 of float32 (basically flattened array of 6 1x2 vectors num_tracks+distance from origine)



Data points distribution

Autoencoder consist form 5 fully connected layers 8 - 4 -3 - 4 -8

Autoencoder structure

training

I use keras with tensorflow backend and pytorch frameworks

As net has quite small size it is reasonable to train it on CPU,

Attempts to train of GPU were made and as a conclusion loading sparse random (which is totally valid description of training dataset) data in and out GPU memory from RAM prove to be a bottleneck.

So I train encoder for 100 epochs with batch size of 512 and MSE loss

Finally i do inference on the sample from train data which does not take part in train and test data which contain higgs boson decay signal