CBM performance for K_S^0 meson measurement using Machine Learning

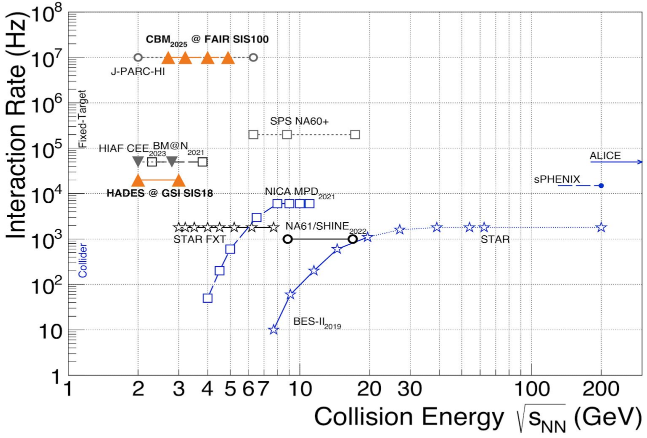
Olha Lavoryk (Taras Shevchenko National University of Kyiv),

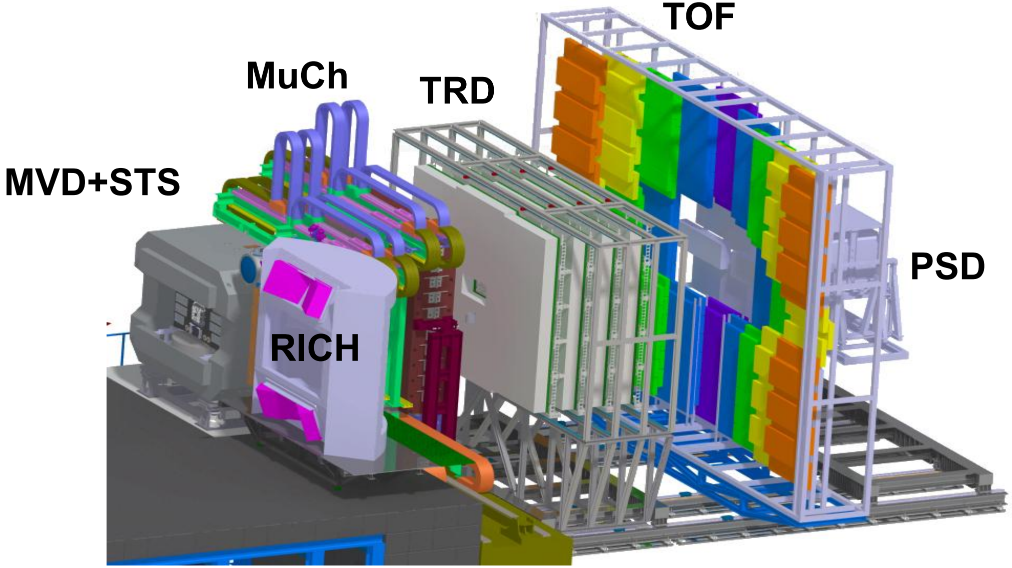
Andrea Dubla, Ilya Selyuzhenkov, Oleksii Lubynets, Shahid Khan, Viktor Klochkov

FAIRNESS
25 May 2022

CBM physics goals and experimental challenges

CBM Collaboration, EPJA 53 3 (2017) 60
T.Galatyuk, NPA982 (2019), update (2021)





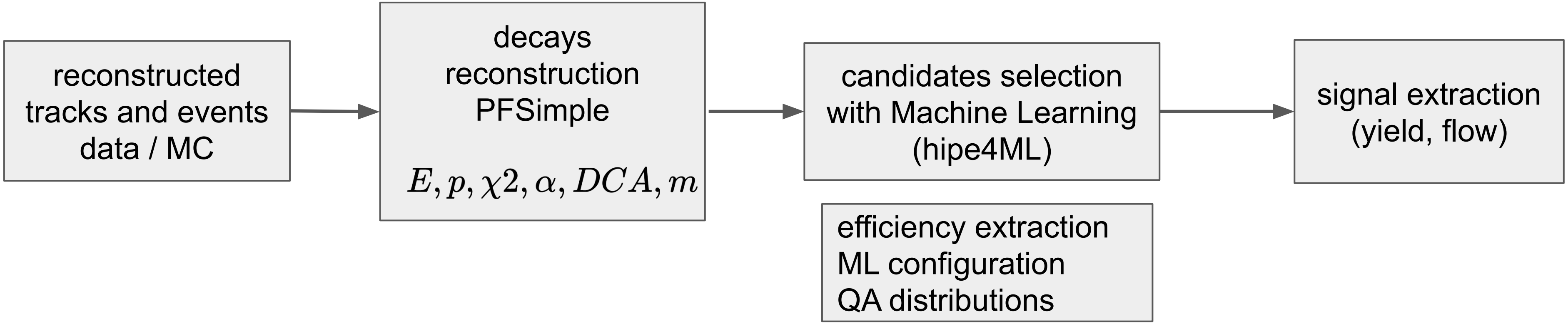
CBM physics program: study QCD matter in extreme conditions (high net-baryon densities, moderate temperatures), equation of state of nuclear matter at densities similar to the densities in the core of neutron stars

Major observables:

- Multi-strange hyperons and Hypernuclei
- Flows and fluctuations
- Dilepton spectra

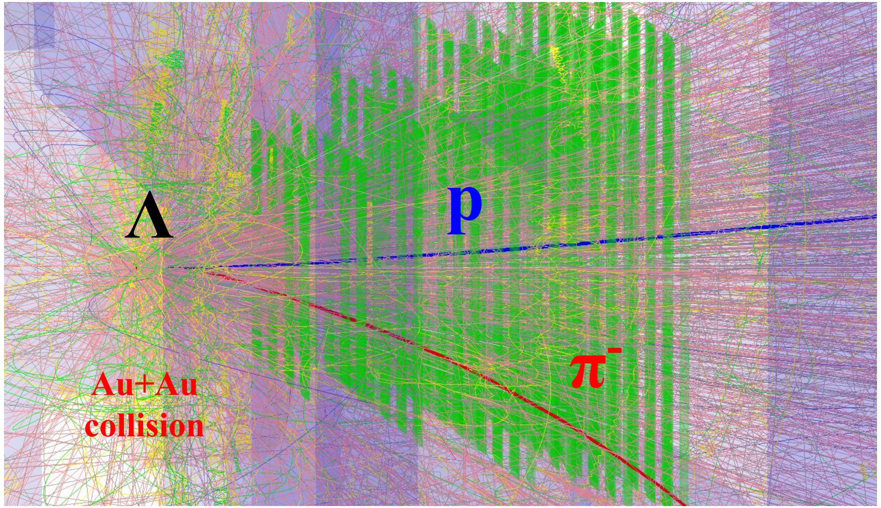
- Tracking: Micro-Vertex Detector (MVD)
  Silicon Tracking System (STS)
- Particle identification:
  Muon Chambers (MuCh)
  Ring Imaging Cherenkov (RICH) detector
  Transition Radiation Detector (TRD)
  Time of Flight (TOF) detector
- Collision geometry: Projectile Spectator Detector (PSD)

(Multi-)strange analysis workflow



(Multi-)strange reconstruction in CBM

CBM event display Au+Au @ p_{beam} = 12*A* GeV/*c* central DCM-QGSM-SMM



Reliable and efficient reconstruction of (multi-)strange hadrons, hypernuclei, and other decays is crucial for CBM physics analysis:

- **PFSimple** package is designed to be flexible and modular for systematic performance studies and physics analysis

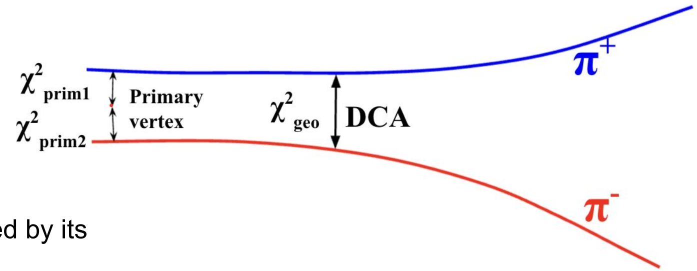
Manual optimization of the multi-dimensional parameter space of decay selection variables is inefficient and will require a lot of time:

- An automatic and efficient procedure, which can reject background and efficiently select signal is needed

Use Machine Learning (ML) algorithms for selection optimization

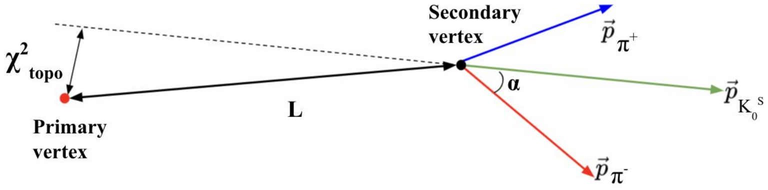
K_S^0 reconstruction

- Combine all pion tracks (MC PID is used)
- Pion pair coming from a K_S^0 decay is termed as signal (MC=1)
- Signal sample mass range: 0.43485 – 0.56135 Gev (5σ around the K_S^0 peak)
- Pion pair not originating from a K_S^0 decay is considered as background (MC=0)
- Background sample mass range: 0.3-1 GeV ($m_{\pi+}+m_{\pi-}$=0.996 GeV≈1 GeV)



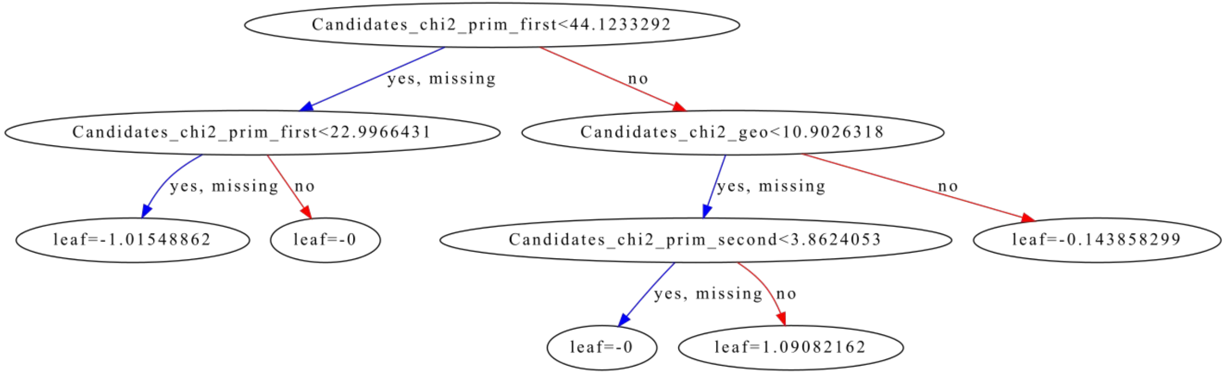
Variables :

- χ^2_{prim} - squared distance Δ**r** between the daughter track and the primary vertex divided by its error
- **DCA** - distance of closest approach between positive and negative & pion tracks
- χ^2_{geo} - squared distance Δ**r** between daughter tracks divided by its error C
- **cosinepos** - angle between proton and K_S^0 momenta
- **cosineneg** - angle between pion and K_S^0 momenta
- **L/ΔL** - distance between primary and secondary vertex divided over its error
- χ^2_{topo} - squared distance Δ**r** between V0-candidate trajectory and the primary vertex divided by its error C
- **cosine topological** - cosine of the angle between primary vertex and point of K_S^0 origin



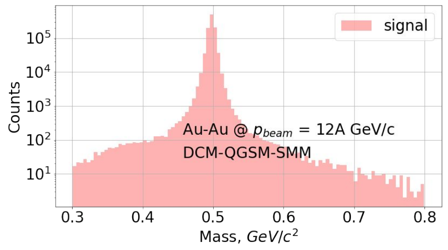
Machine learning via Boosted Decision Trees

- Boosting combines weak learners (error rate <50%)
  to make a strong learner (error rate <25%)
- Decision trees (weak learners) are combined together to make a GB algorithm
- In each step a new tree is used to improve the previous prediction
- XGB is an extension of GB with:
  - better control over overfitting
  - parallel processing
  - additional features

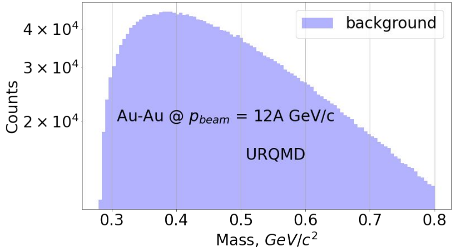


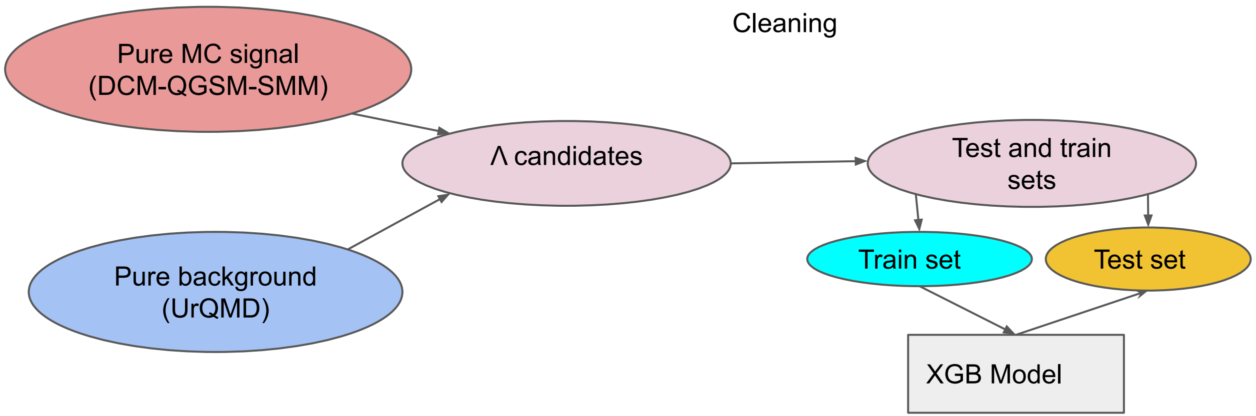
Following the implementation by the ALICE Collaboration in the hipe4ml package

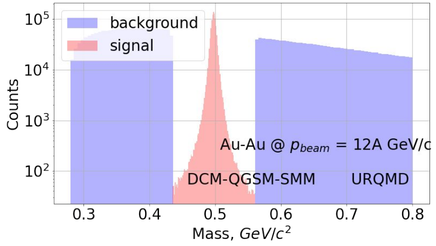
XGB implementation for K_S^0



- DCM-QGSM-SMM sample as simulated data (MC signal)
- UrQMD sample is treated as experimental data (MC background)
- K^0_S candidates are cleaned by removing nonphysical values
- K^0_S candidates are divided into train(80%) and test(20%) samples

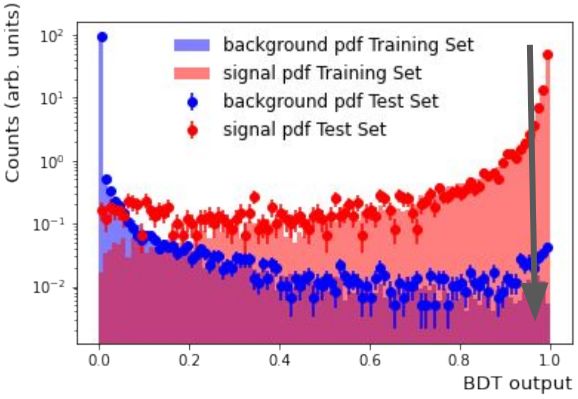


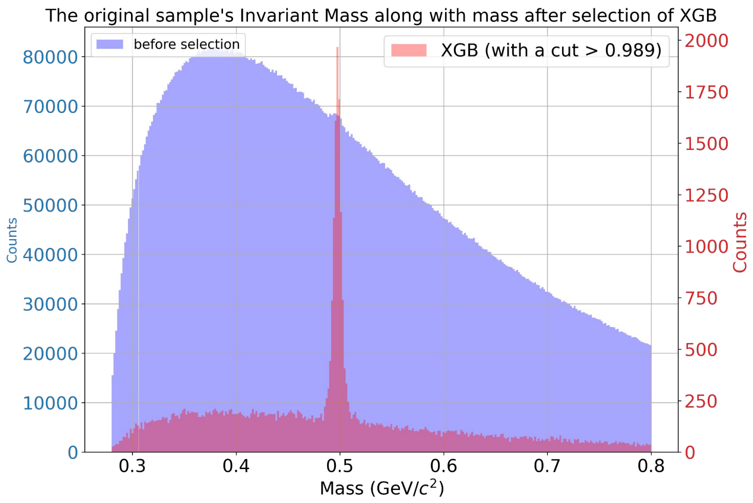


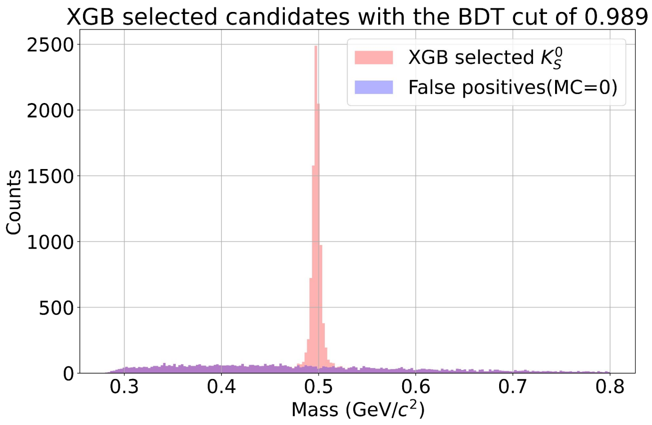


Background is selected ± 5σ away from the K^0_S peak mean

XGB performance for K^0_S candidates selection



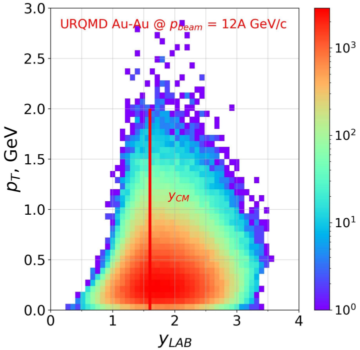




XGB trained and tested models are applied to a sample of 50k of DCM-QGSM-SMM and UrQMD events

Yield Extraction: fitting procedure

Divide (p_T,y) phase space into 6x6 bins

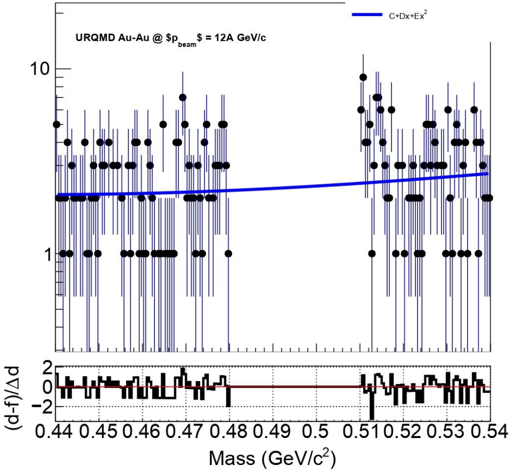


Double Gaussian function is used for signal and 2nd order polynomial for background

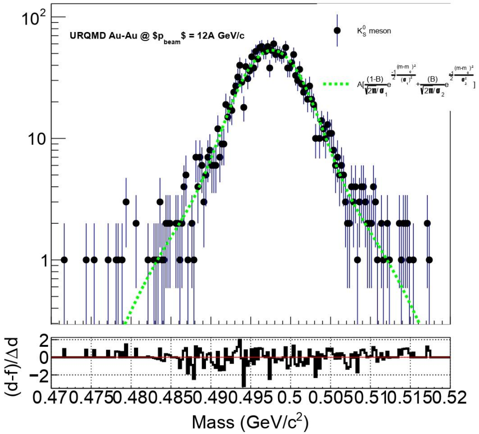
$$Fit(m) = Ae^{\frac{-1}{2}\frac{(m-m_0)^2}{\sigma_1^2}} + Be^{\frac{-1}{2}\frac{(m-m_0)^2}{\sigma_2^2}} + pol2(m)$$

1. Exclude signal region (m<0.43485 & m>0.56135) and fit background with *pol2(m)*
2. Use background fit parameters as initial values for next iteration, where signal (double Gaussian) fit function has fixed $m_0 = 0.4976$ GeV/c^2 and widths σ_1=0.004 GeV, σ_2=0.007 GeV
3. Use fit parameters as initial values for unconstrained fit to the whole inv. mass range

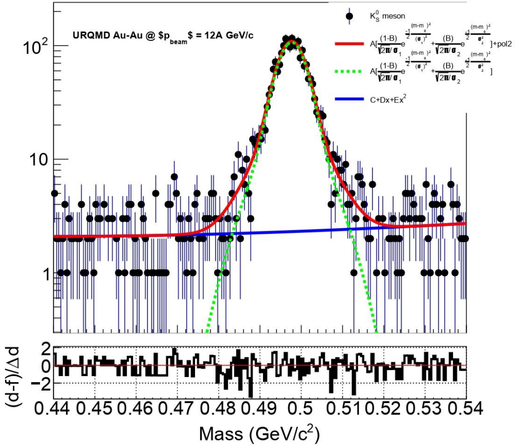
Step 1



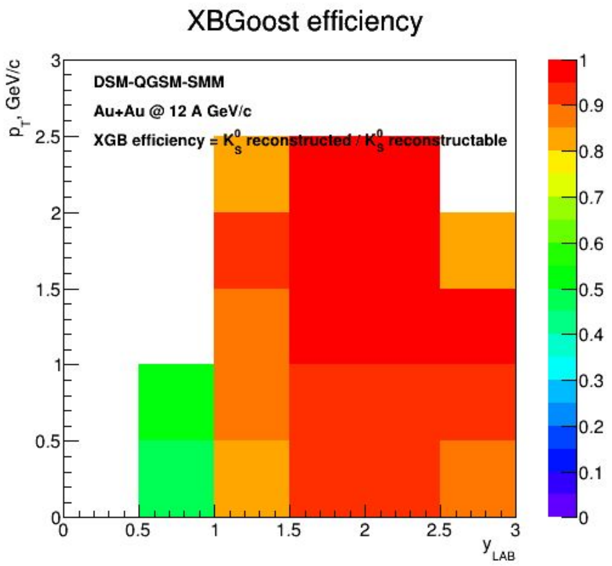
Step 2



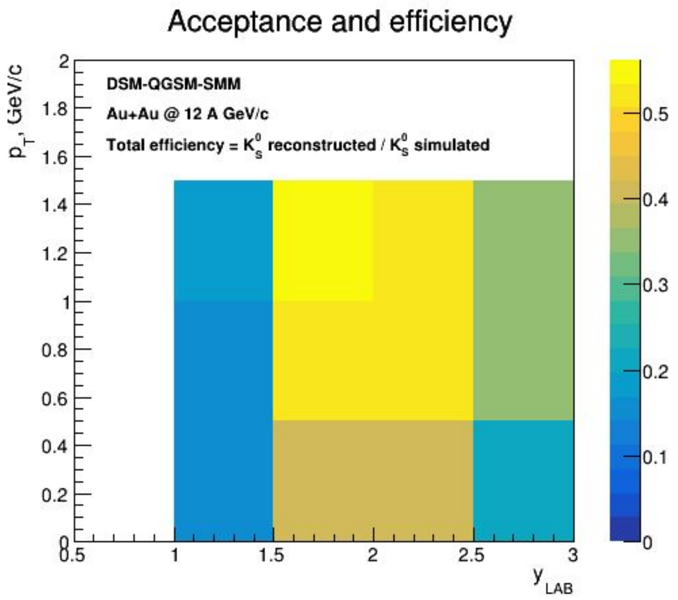
Step 3



Acceptance and efficiency of K_S^0 decays reconstruction



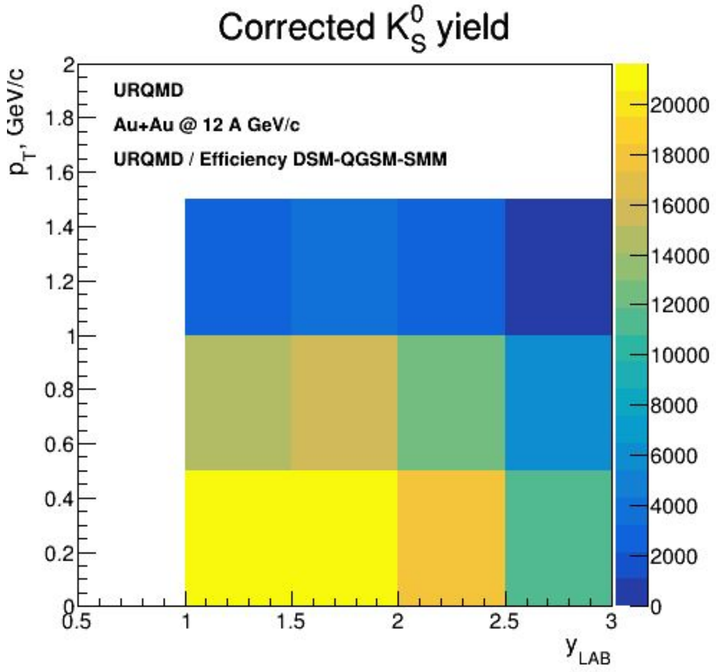
XGBoost efficiency ~ 95%

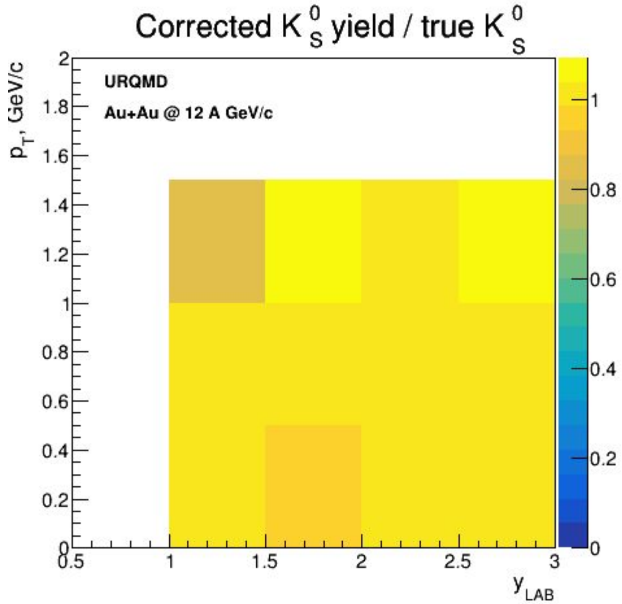


Total reconstruction (acc x efficiency) ~ up to 50%

Reconstructed = reconstructed + selected K^0_S
Reconstructable = both daughters are reconstructed

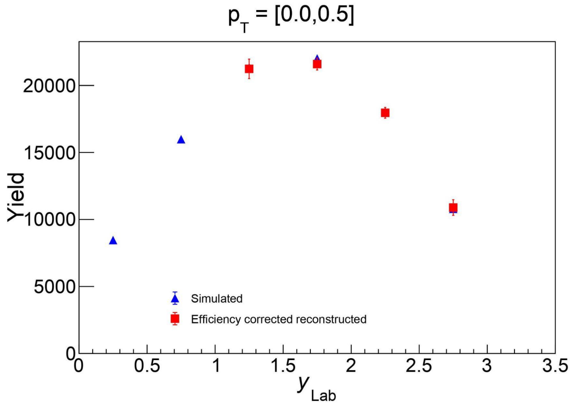
Efficiency and acceptance corrected K^0_S yield

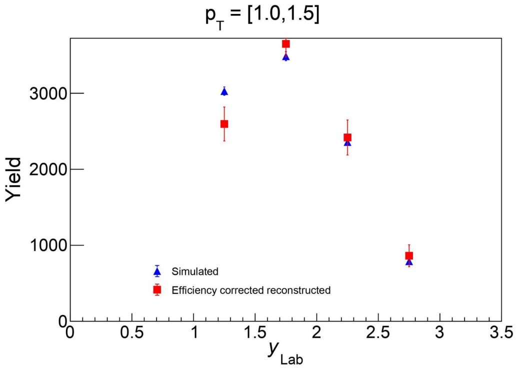


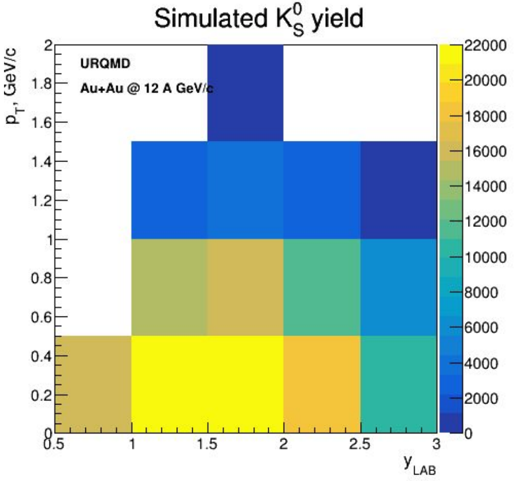


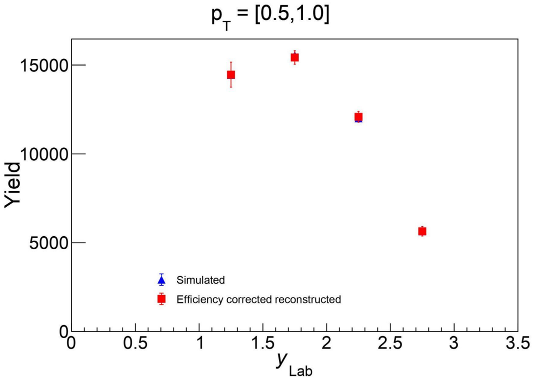
The corrected yield is in good agreement with the simulated yield

Efficiency and acceptance corrected yield









Summary

- K_S^0 meson decays are efficiently selected using ML techniques
- Machine learning framework for analysis of particle decays was used for K_S^0 extraction
- Optimization of selection criteria performed via XGB
- Yield, extracted after XGB selection and (acceptance x efficiency) corrected is compatible with initial model spectra

Thank you for the attention!

Back up

Machine learning implementation

- Input (root) files with signal and background
- Plot variables distributions and correlation matrices
    - QA
    - plot (non-)linear correlations
    - Select features for optimization
- Tune parameters by Bayesian optimization
- Train and test
- Save model as C++ library
- Apply model on data
- Check results after selection
    - confusion matrix
    - possibility to visualize the selection
    - p_T-rapidity distributions
    - variables distributions before and after ML cut (signal and background)

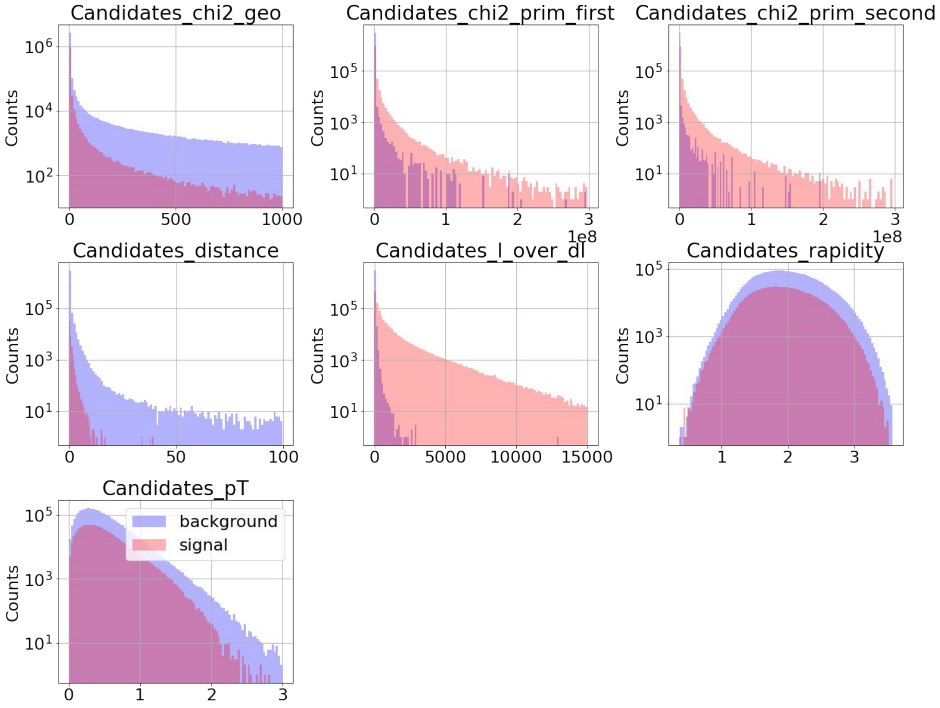
Selection optimization tool developed for ALICE Collaboration

https://hipe4ml.github.io; Tutorial

Integration with hipe4ML is in progress

Current version of the code: https://github.com/CBM-ML/CandidatesClassifier

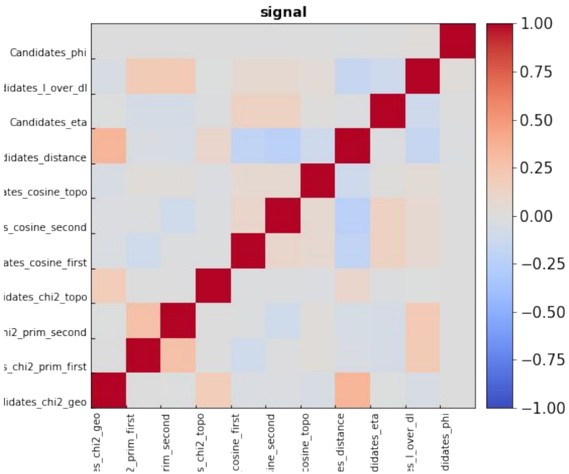
Variables distribution

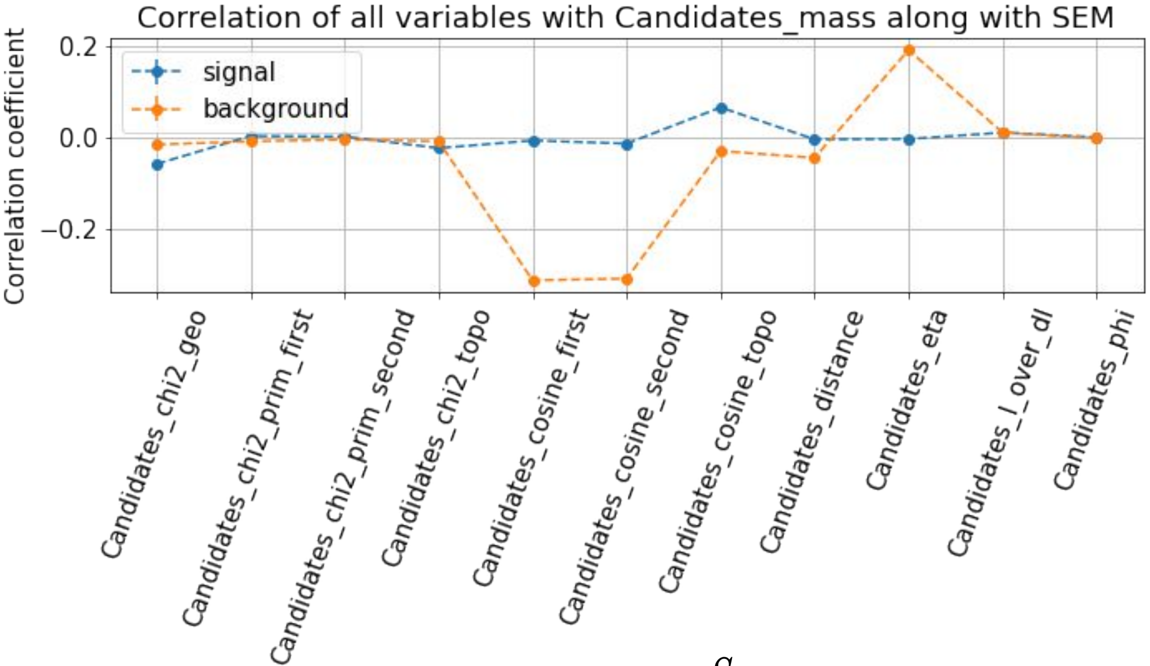


Quality cuts were applied:

- -50 < X < 50
- -50 < Y < 50
- -1 < Z < 80
- 0 < distance < 100
- 1 < eta < 6.5
- 0 < chi2_topo < 100000
- 0 < chi2_geo < 1000
- 0 < chi2_prim_first < 3e8
- 0 < chi2_prim_second < 3e8

Variables linear correlation plots





$$r = \frac{\sum(X-\bar{X})(Y-\bar{Y})}{\sqrt{\sum(X-\bar{X})^2(Y-\bar{Y})^2}}$$

$r - Pearson\ correlation\ coefficient,$

$\bar{X} - mean\ of\ X\ variable$

$\bar{Y} - mean\ of\ Y\ variable$

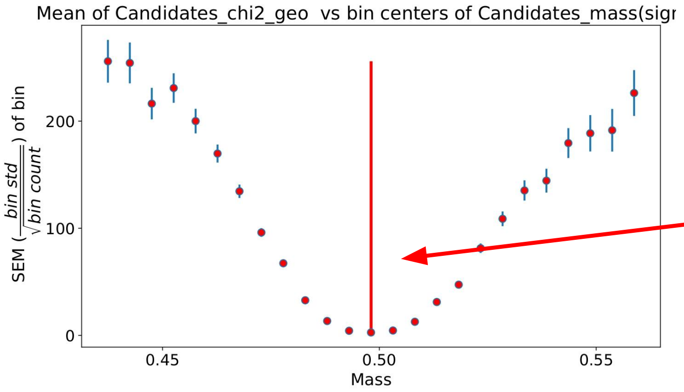
$$S_{\bar{X}} = \frac{S}{\sqrt{n}}$$

$S_{\bar{X}} - standard\ deviation\ of\ the\ mean,$

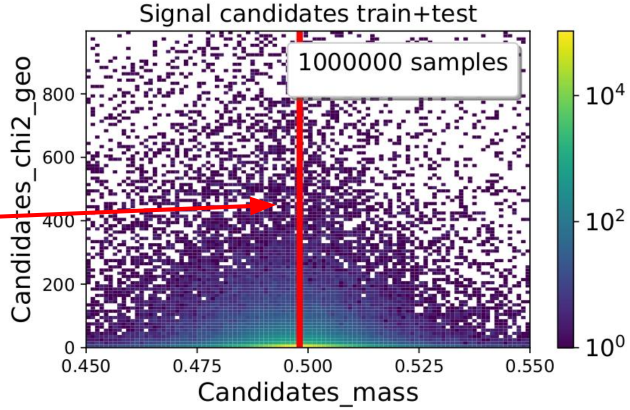
$S - standard\ deviation\ of\ sample$

$\sqrt{n} - sample\ size$

Non-linear correlations



The red line denotes the K_S^0 peak region



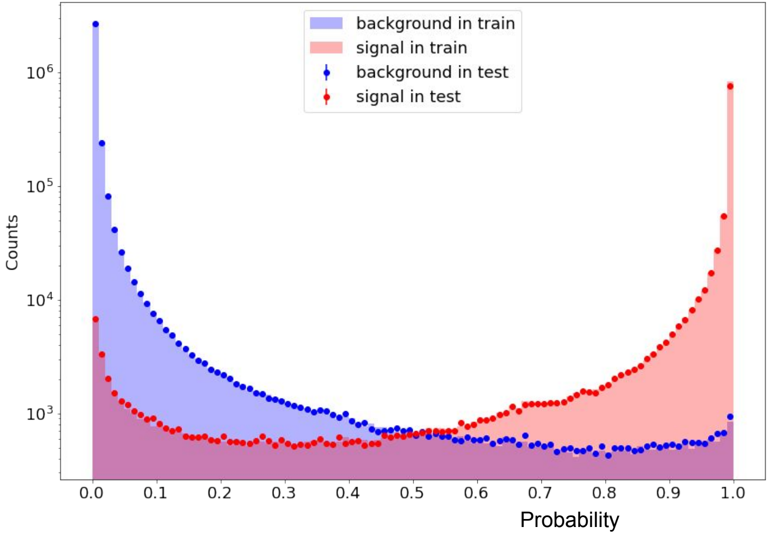
Signal is selected ± 5σ within the K_S^0 peak mean

2D distribution between variables and invariant mass

The mean of each bin of the variable, to be checked for correlation(Y axis), is plotted against the bin center of the mass variable(X axis). Also SEM is calculated for each bin and it is also shown in the same plot

XGB Model evaluation

Model trained on the train sample is applied to the train-test data sets

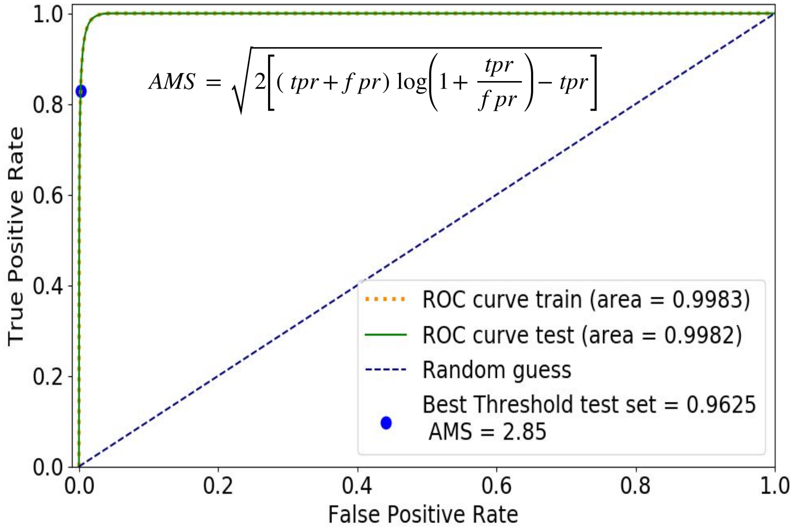


True positive rate = tpr; Signal = S ; Background = B

$$tpr = \frac{S\ classified\ as\ S}{S\ classified\ as\ S + S\ classified\ as\ B}$$

$$fpr = \frac{B\ classified\ as\ S}{B\ classified\ as\ B + B\ classified\ as\ S}$$

Optimize Λ candidates selection for significance

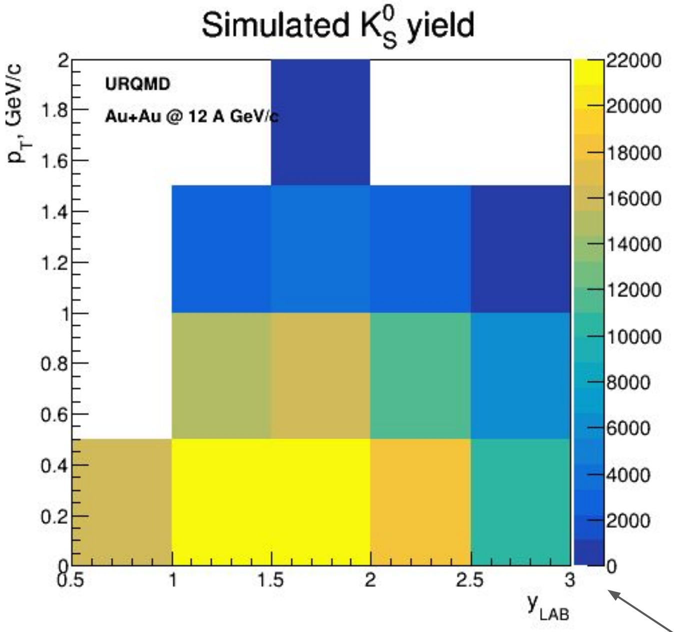


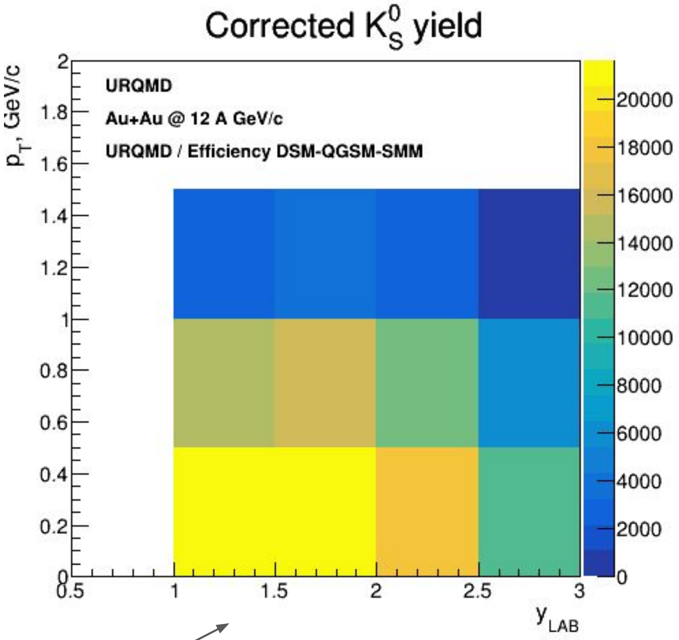
Threshold on the ROC (Receiver Operating Characteristic) curve which maximizes Approximate Median Significance (AMS) on the test sample is our Best Threshold

$$AMS_2 = \frac{s}{\sqrt{b}} \times \sqrt{1 + O\left[\left(\frac{s}{b}\right)^3\right]} \quad ; b >> s \quad \Rightarrow AMS_2 \approx \frac{s}{\sqrt{b}}$$

Shahid Khan et al.

Simulated and reconstructed K^0_S yield





The corrected yield is in good agreement with the simulated yield

ML framework configuration with TOML

TOML format (toml.io/en) is a minimal configuration file format that's easy to read due to obvious semantics.

- Designed to map unambiguously to a hash table
- Easy to parse into data structures in a wide variety of languages

Implemented for CBM:
User can specify parameters via configuration files

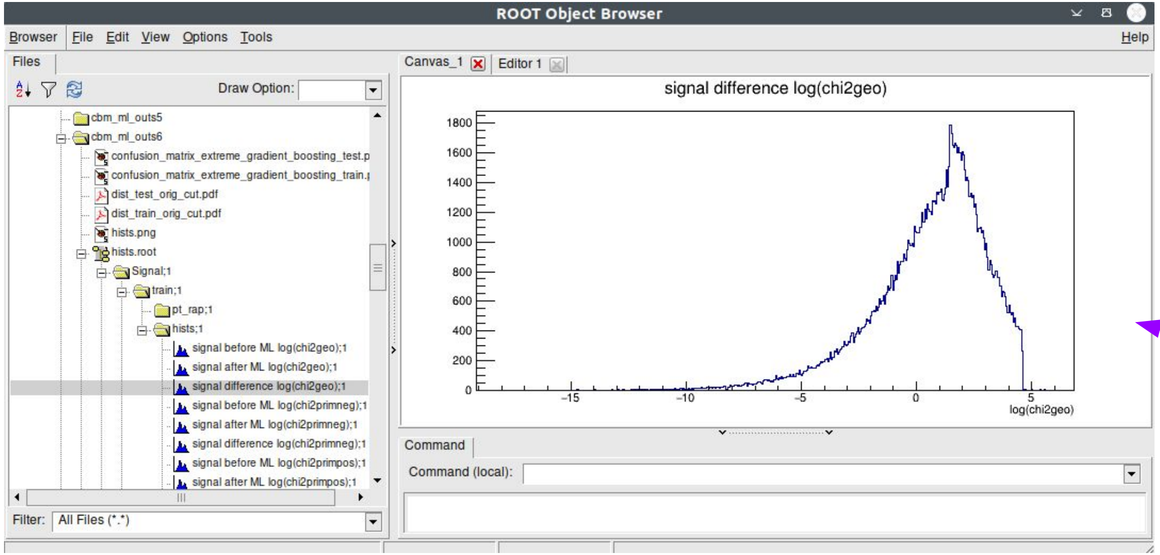
```
title = "Input path config for xgb classifier"

  Signal
path = "/home/olha/CBM/dataset10k_tree/dcm_1m_prim_signal.root"
tree = "PlainTree"

  Background
path = "/home/olha/CBM/dataset10k_tree/urqmd_100k_cleaned.root"
tree = "PlainTree"

  deploy
path = "/home/olha/CBM/dataset10k_tree/urqmd_100k_cleaned.root"
tree = "PlainTree"
```

Output file with QA information



possible retrieve variable distribution for further check

Output PDF or PNG plot doesn’t allow to do manipulations with histogram pictures (rebin, scaling), so we need to save object itself

Saved histograms as root objects for more precise comparison , could be found here