Gradient Boosted Decision Trees for Particle Identification Problem at MPD Experiment

A. Ayriyan (JINR & AANL)

in cooperation with **Vladimir Papoyan**, A. Aparin, H. Grigorian, A. Korobitsin, A. Mudrokh

Conference on High Energy Physics

11-14.09.2023, Yerevan, Armenia

IDENTIFICATION PROBLEM OF CHARGED PARTICLES

In Machine Learning terms PID can be considered as **classification** task (**Supervised learning**).

Let

X - is the input space (particle characteristics such as: **dE/dx**, **m2**, **q**, **P,** etc)

Y - is the output space (particle species such as: **π**, **k**, **p**, etc.)

Unknown mapping exists

$$\mathbf{m : X \rightarrow Y},$$

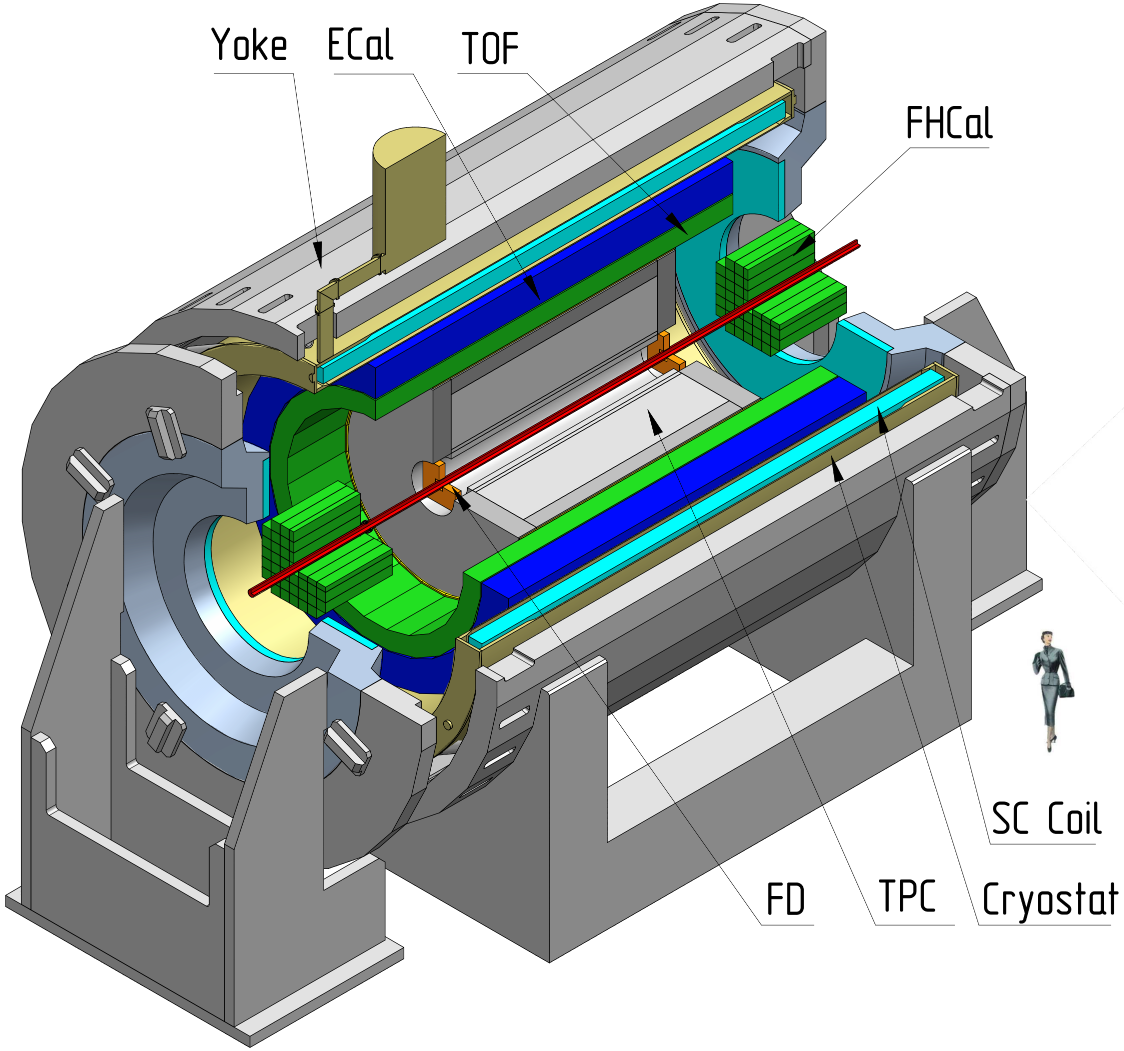
for values which known only on objects from the finite training set

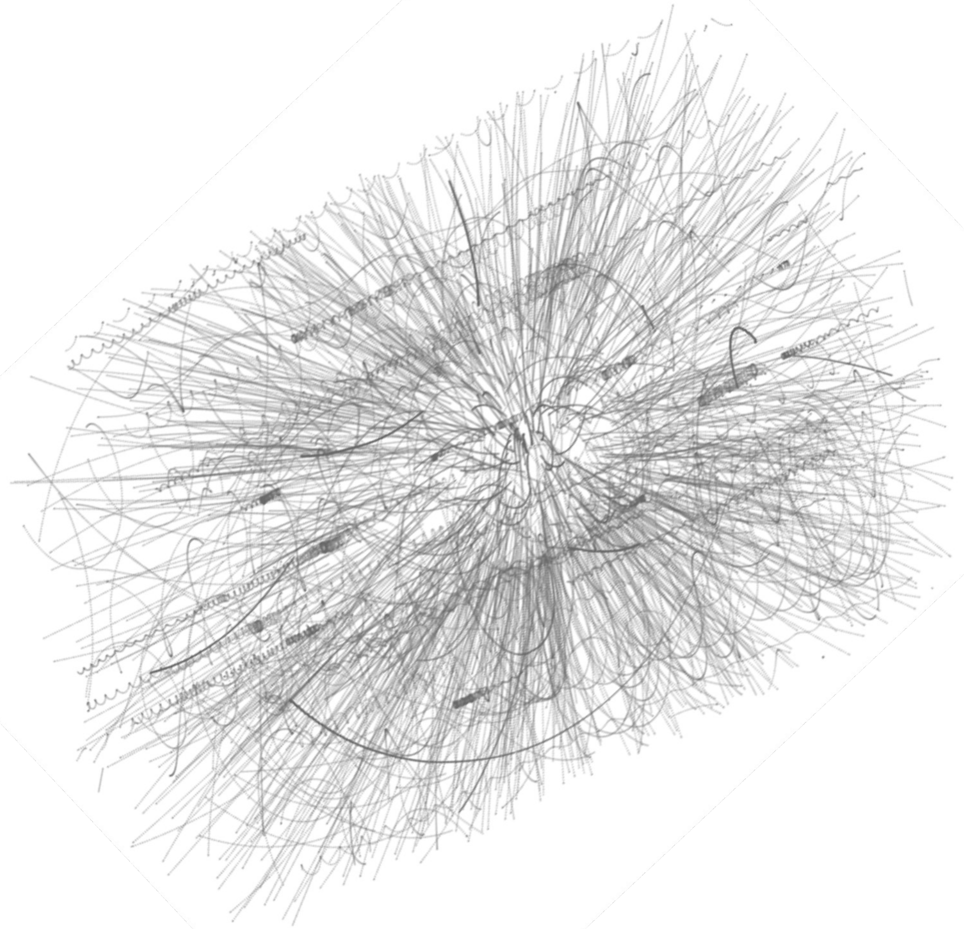
$$\mathbf{X^n = (x_1, y_1), ..., (x_n, y_n)},$$

Goal is to find an algorithm a that classifies an arbitrary new object $x \in X$

$$\mathbf{a : X \rightarrow Y}.$$

MPD apparatus and PID

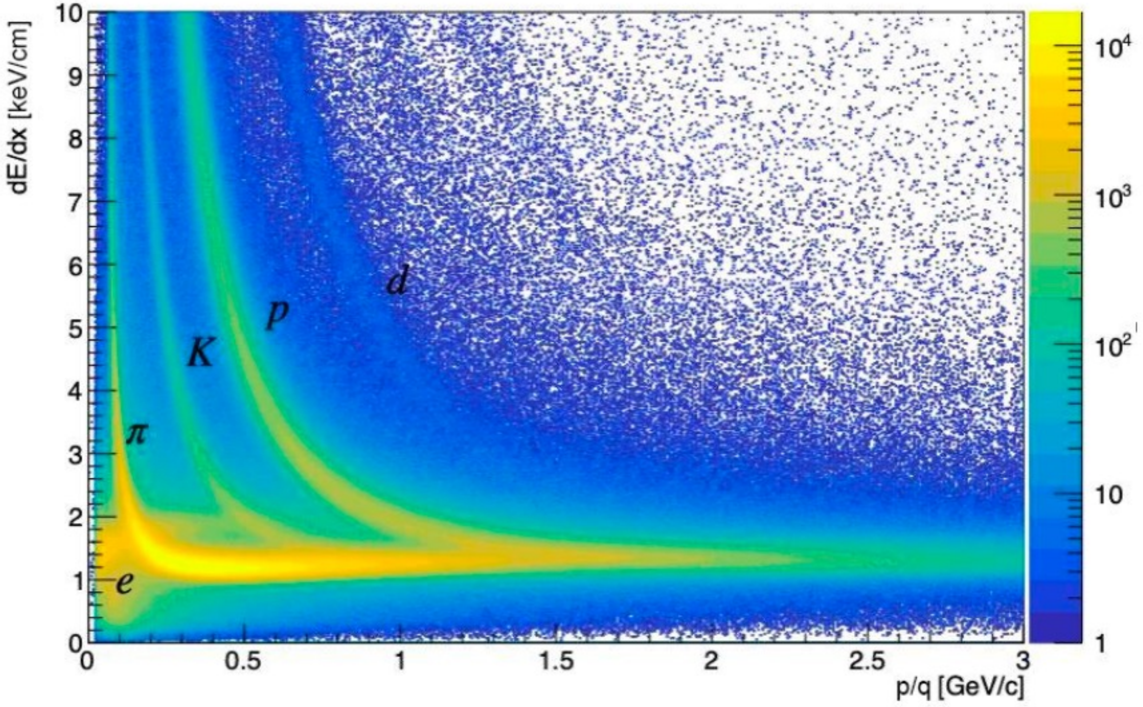


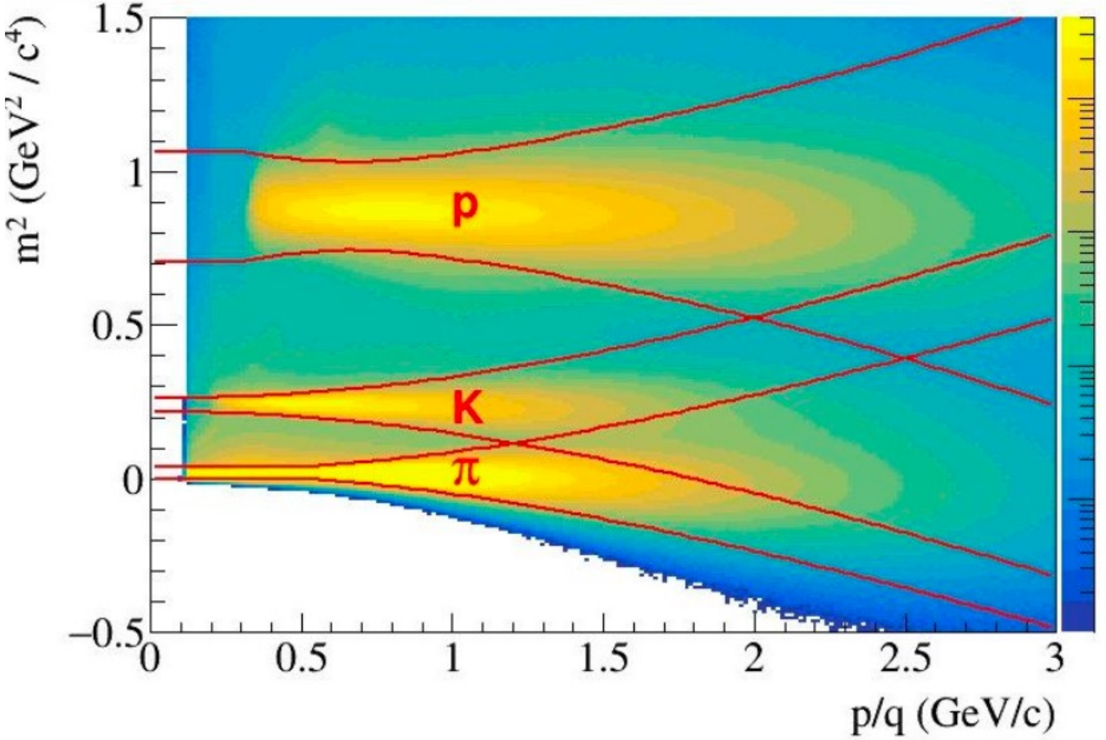


MPD particle identification (PID) based on **Time-Projection Chamber** (TPC) and **Time-of-Flight** (TOF).

Particle Identification in MPD experiment

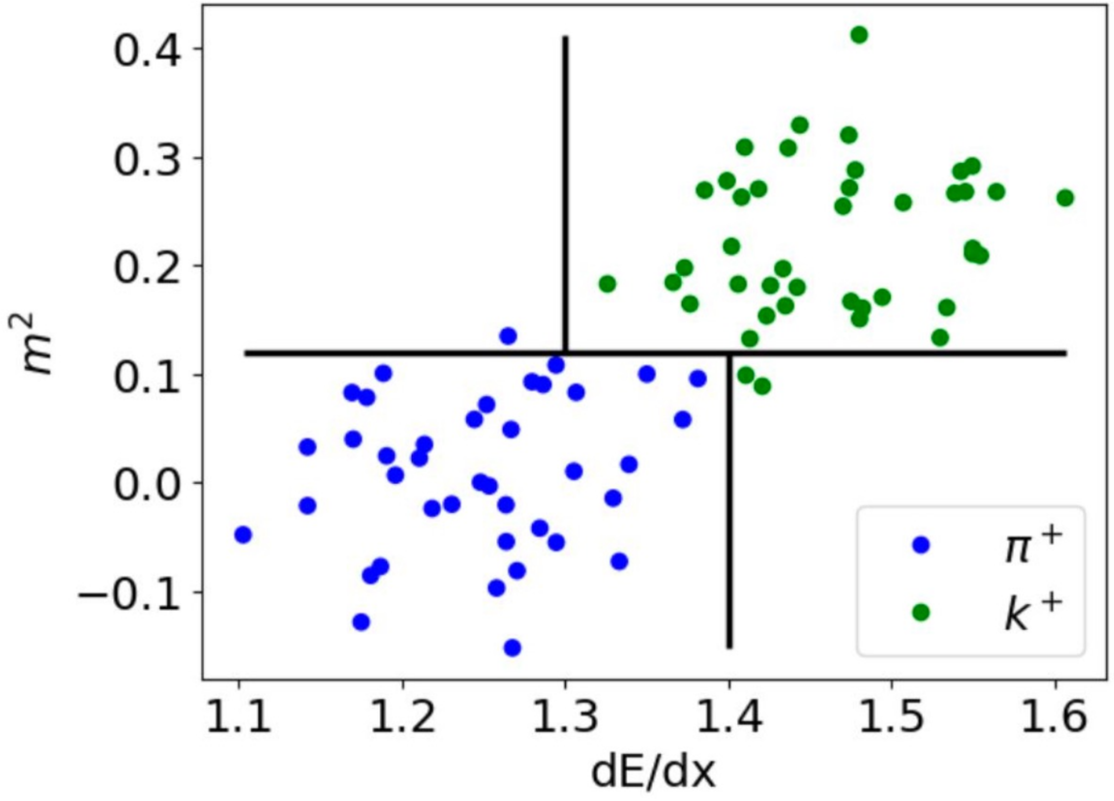
Particle identification can be achieved by using information about **momentum**, **charge**, **energy loss** (TPC) and **mass squared** (TPC + TOF).

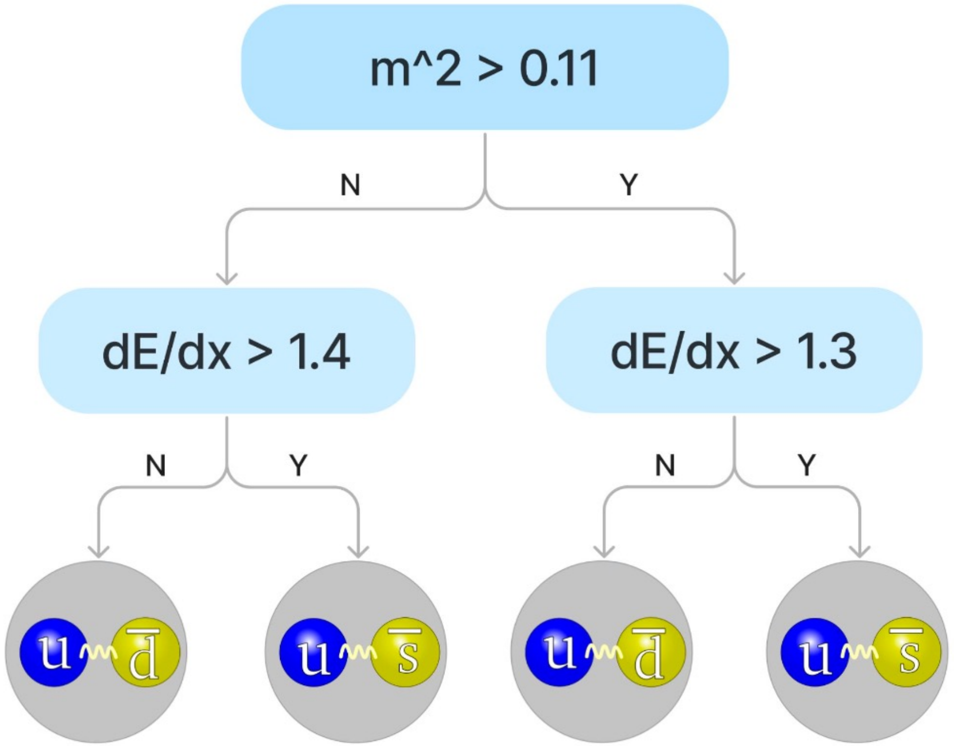




DECISION TREES FOR PID

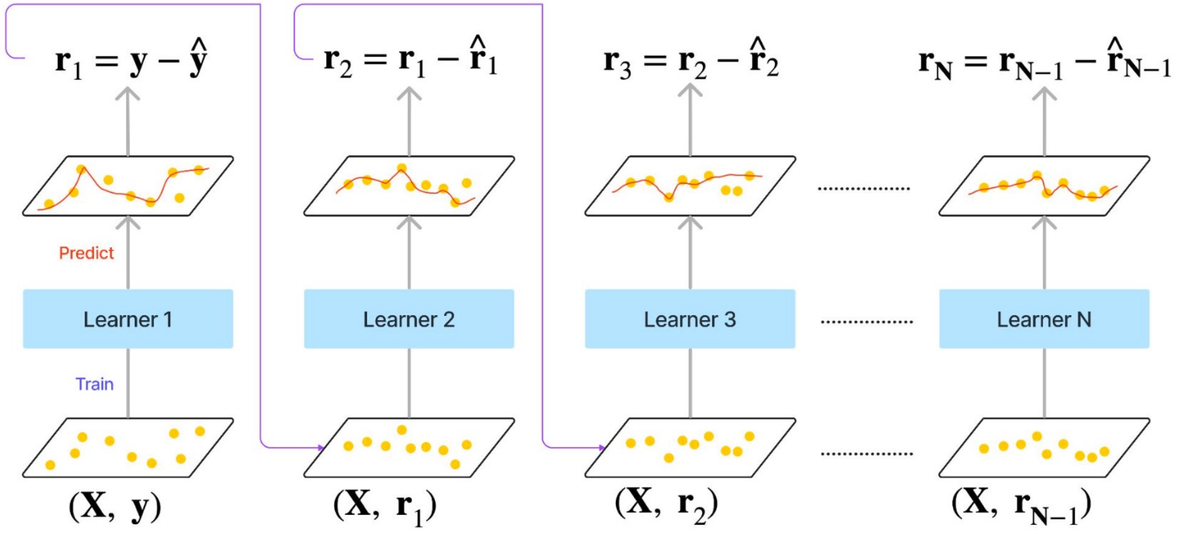
Gradient Boosted Decision Tree (GBDT) uses decision trees as weak learner. They can be considered as automated multilevel **cut-based** analysis.

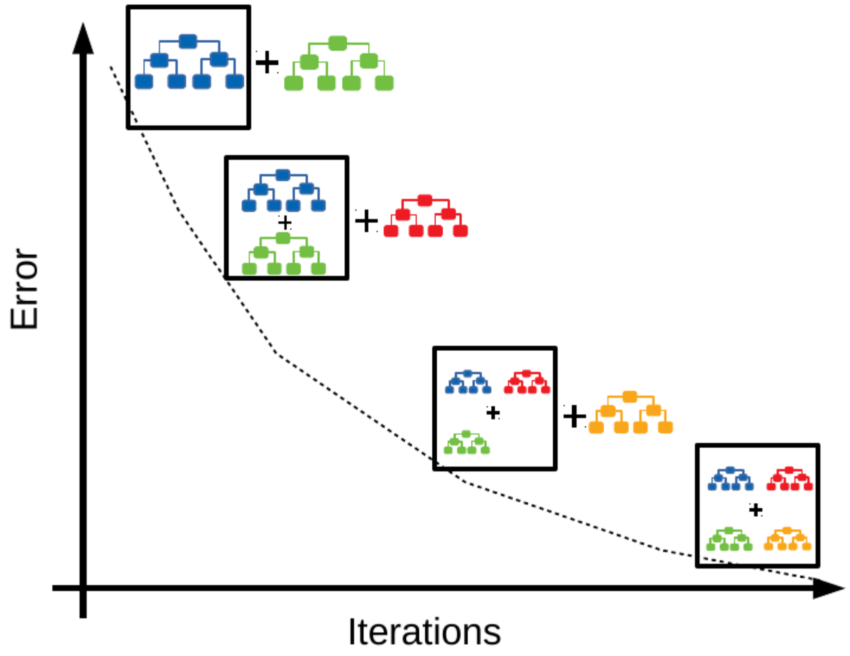




GRADIENT BOOSTING

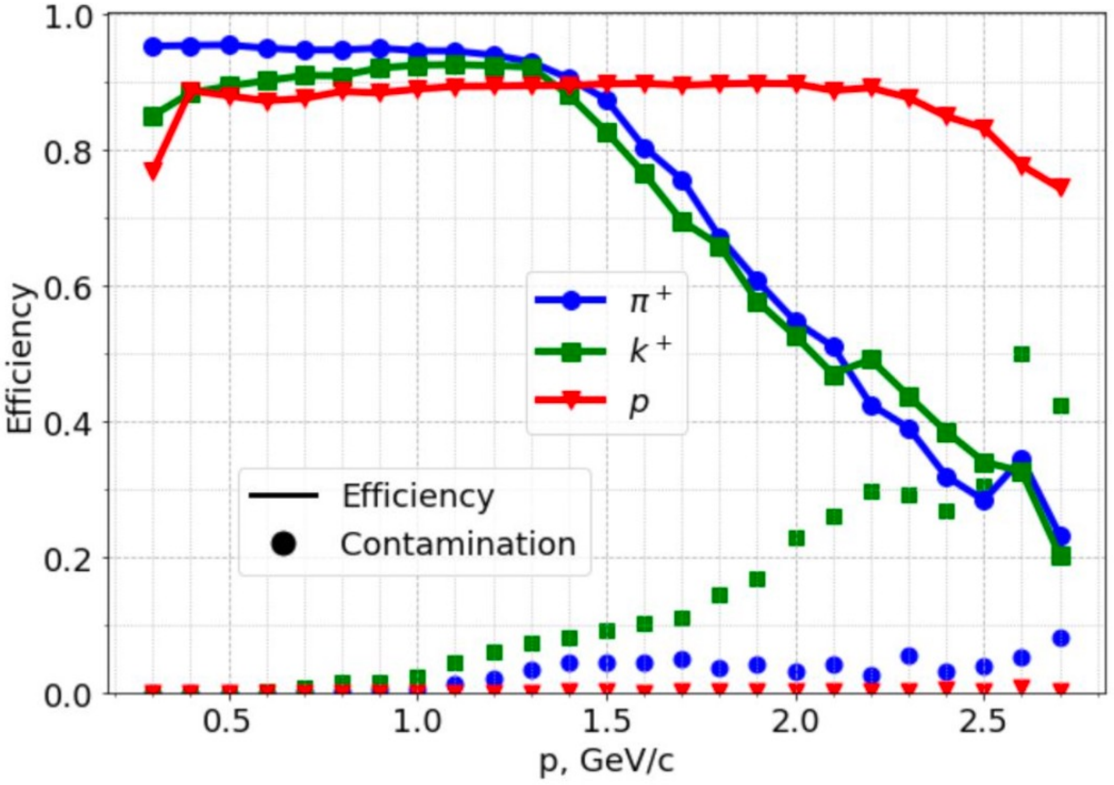
Gradient boosting is a machine learning technique which combines **weak learners** into a single strong learner in an iterative fashion.

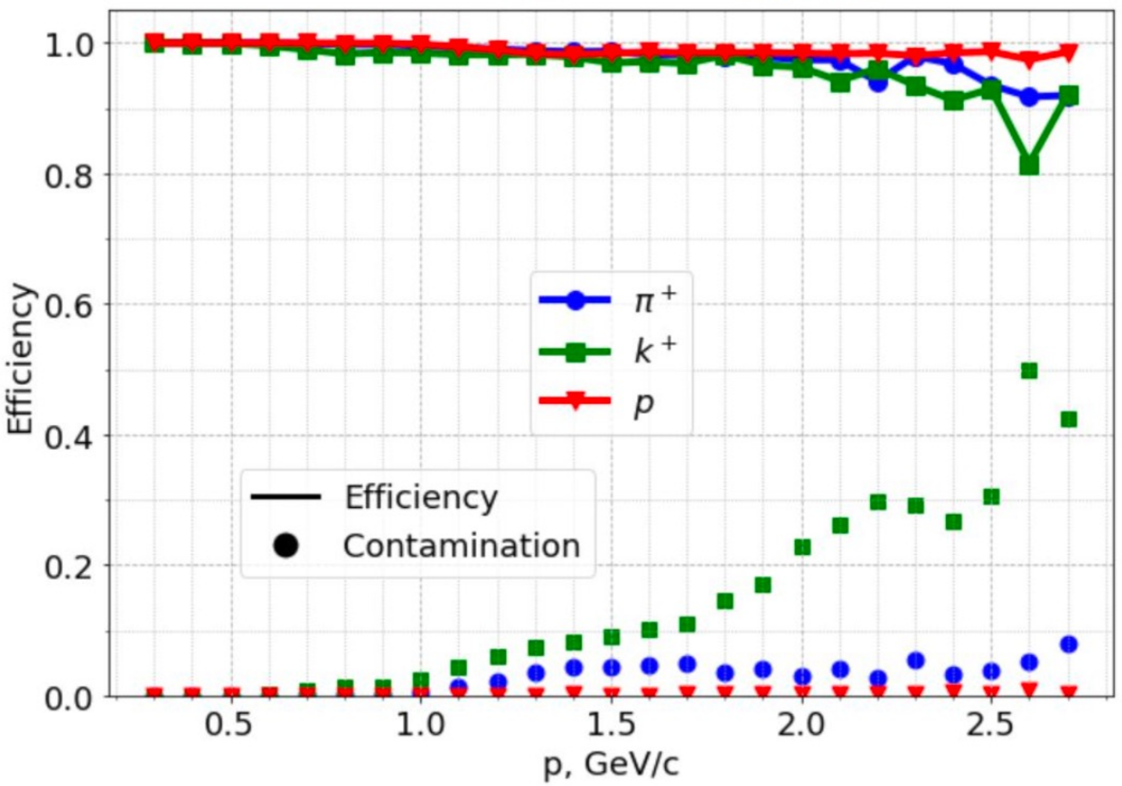




When **weak learnrs are decision tree**, the resulting algorithm is called **gradient-boosted decision trees**.

BASELINE PID IN MPD - N-SIGMA



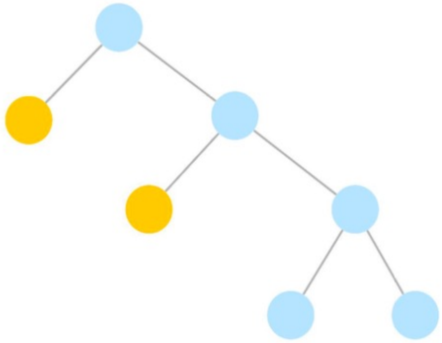


PID efficiency and contamination for all tracks (left) and only identified tracks (right) in Bi+Bi collisions at 9.2 GeV

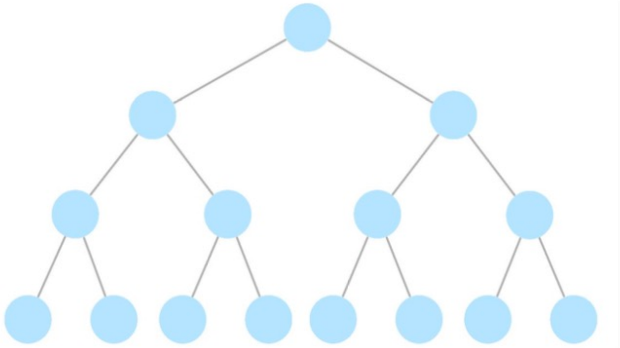
$$E^S = \frac{N^S{}_{corr}}{N^S{}_{true}} \qquad C^S = \frac{N^S{}_{incorr}}{N^S{}_{corr} + N^S{}_{incorr}}$$

XGBoost vs LightGBM vs CatBoost vs SketchBoost

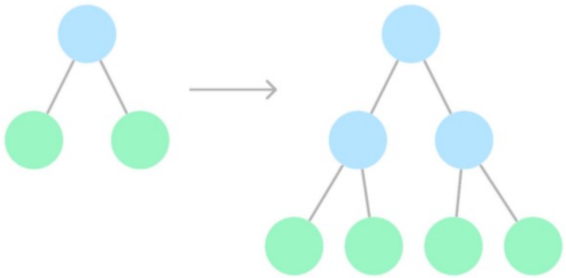
Asymmetric Tree (XGB, LGBM)



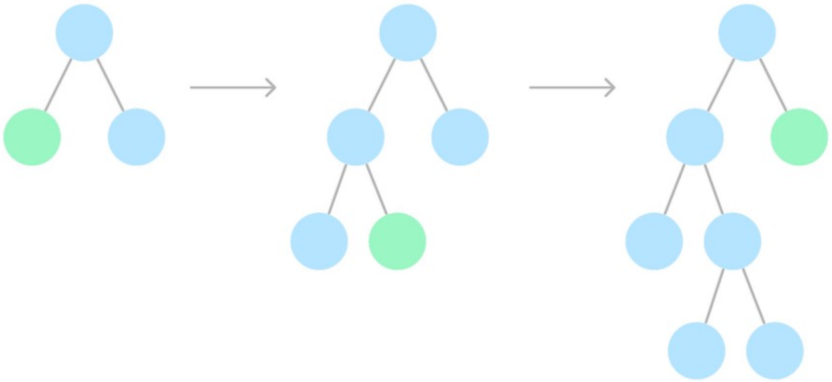
Symmetric Tree (CatBoost, SketchBoost)



Level-wise Tree Growth (XGB)



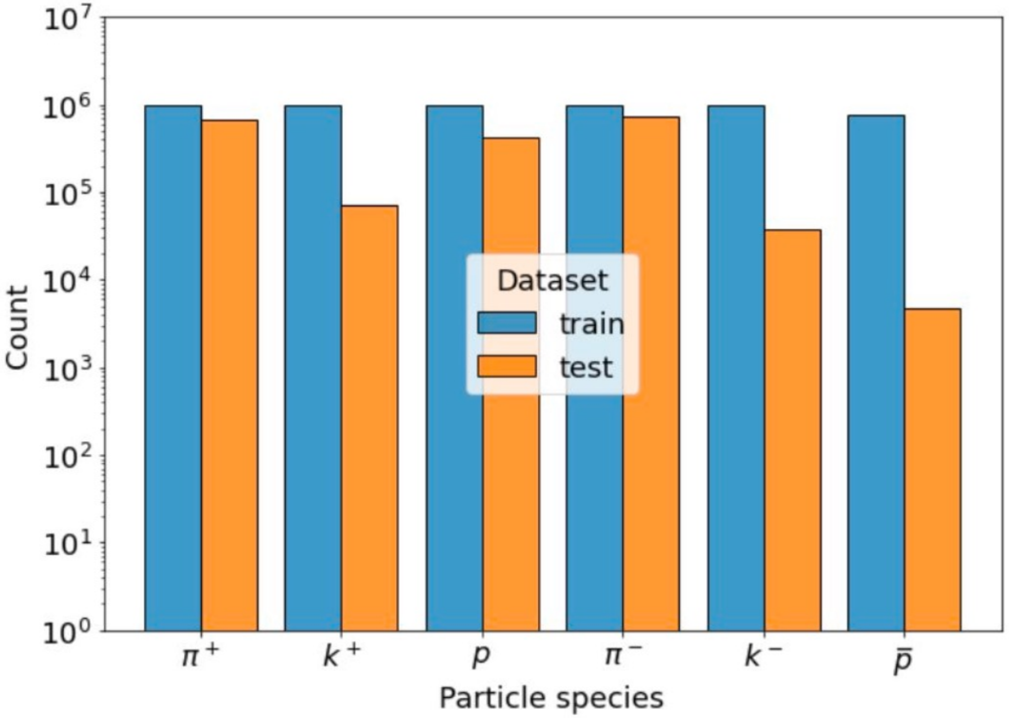
Leaf-wise Tree Growth (LGBM)



DATASETS

Subsamples of the two MPD Monte-Carlo productions have been used

	prod05	prod06
Event generator	UrQMD	PHQMD
Transport	Geant 4	Geant 4
Impact parameter ranges	0-16 fm (mb)	0-12 fm
Smear Vertex XY	0.1 cm	0.1 cm
Smear Vertex Z	50 cm	50 cm
Colliding system	Bi+Bi	Bi+Bi
Energy	9.2 GeV	9.2 GeV



track selection criteria: $(p < 100)$ & $(|m^2| < 100)$ & $(nHits > 15)$ & $(|eta|<1.5)$ & $(dca < 5)$ & $(|Vz| < 100)$

Two stages of the experiments

Some parameters for the tuning and model evaluation stages

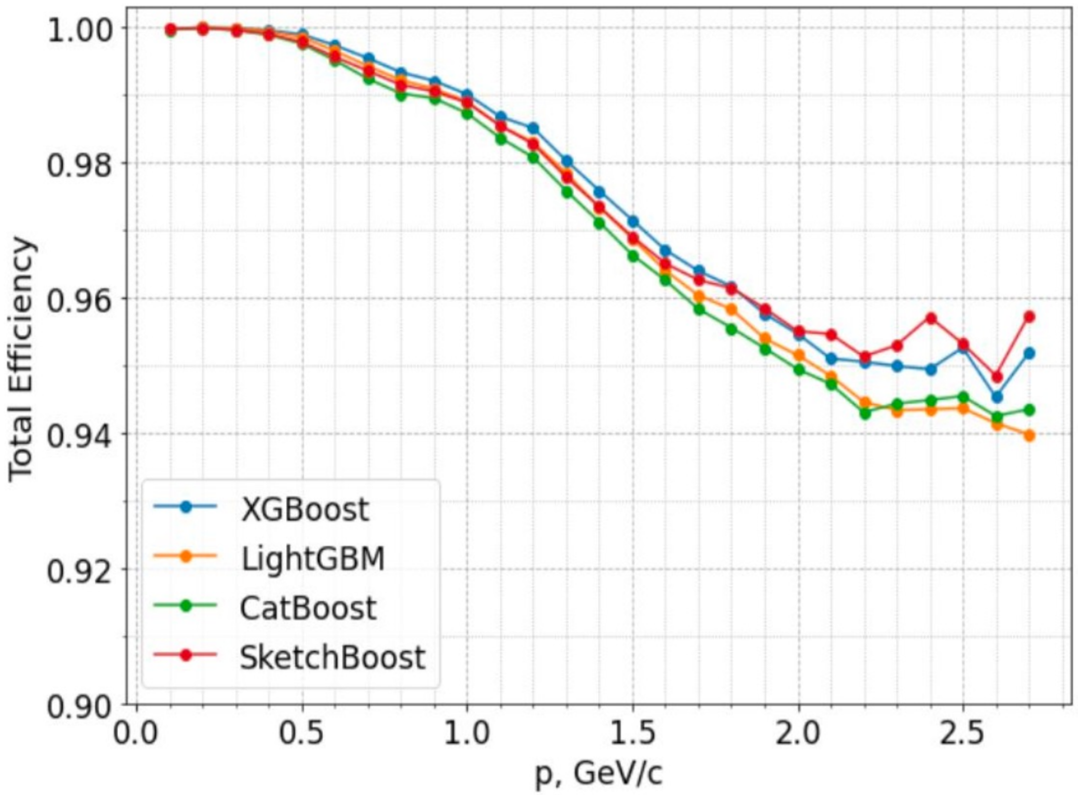
Stage	Learning Rate	Max Number of Iterations	Early Stopping
Tuning	0.05	5 000	200
Model Evaluation	0.015	20 000	500

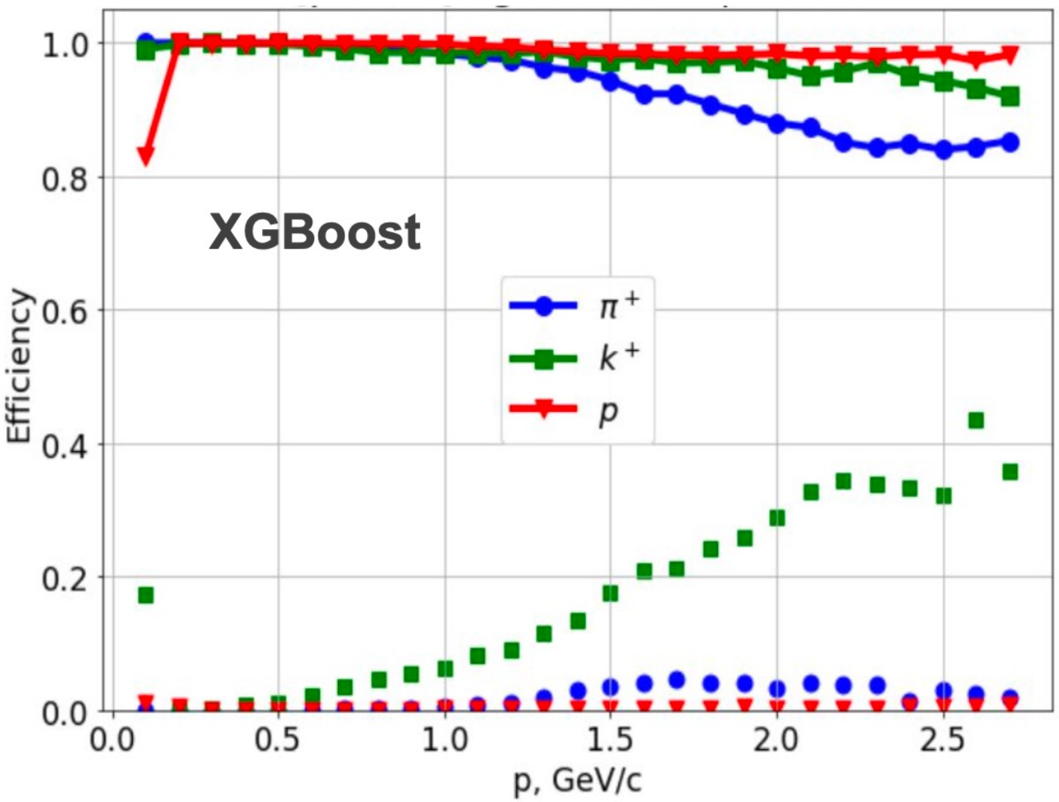
Results for hyperparameter tuning (after **30 iterations** of the TPE algorithm for each GBDT)

Framework	Max. Depth	L2 leaf reg.	Min. data in leaf size	Rows sampling rate
XGBoost	8	2.3	0.00234	0.942
LightGBM	12	0.1	4	0.981
CatBoost	8	3.0	5	0.99
SketchBoost	8	3.0	5	0.99

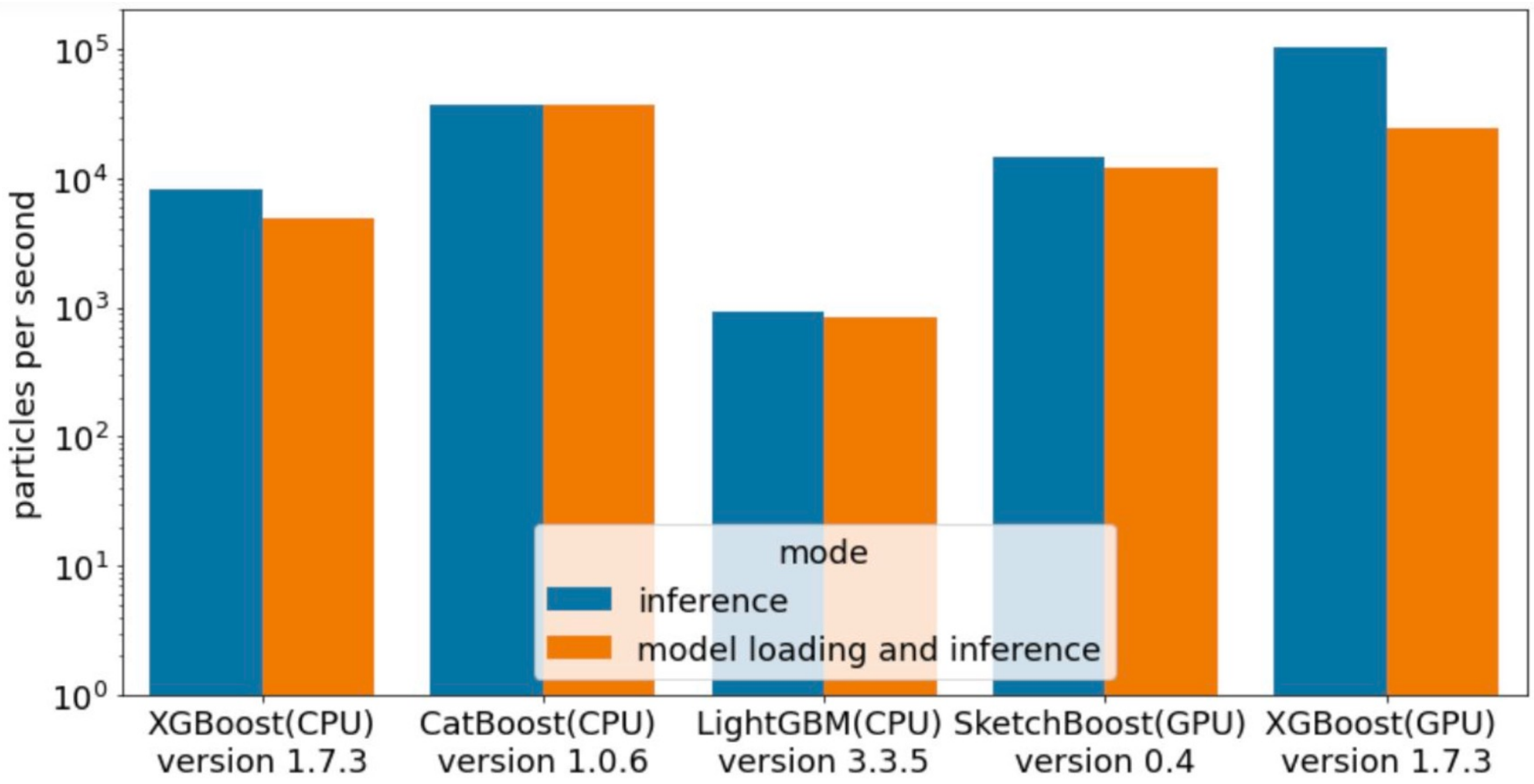
Comparative analysis of the algorithms. Efficiency

	XGBoost	LightGBM	CatBoost	SketchBoost
Total Efficiency	0.99327	0.99235	0.99138	0.99239





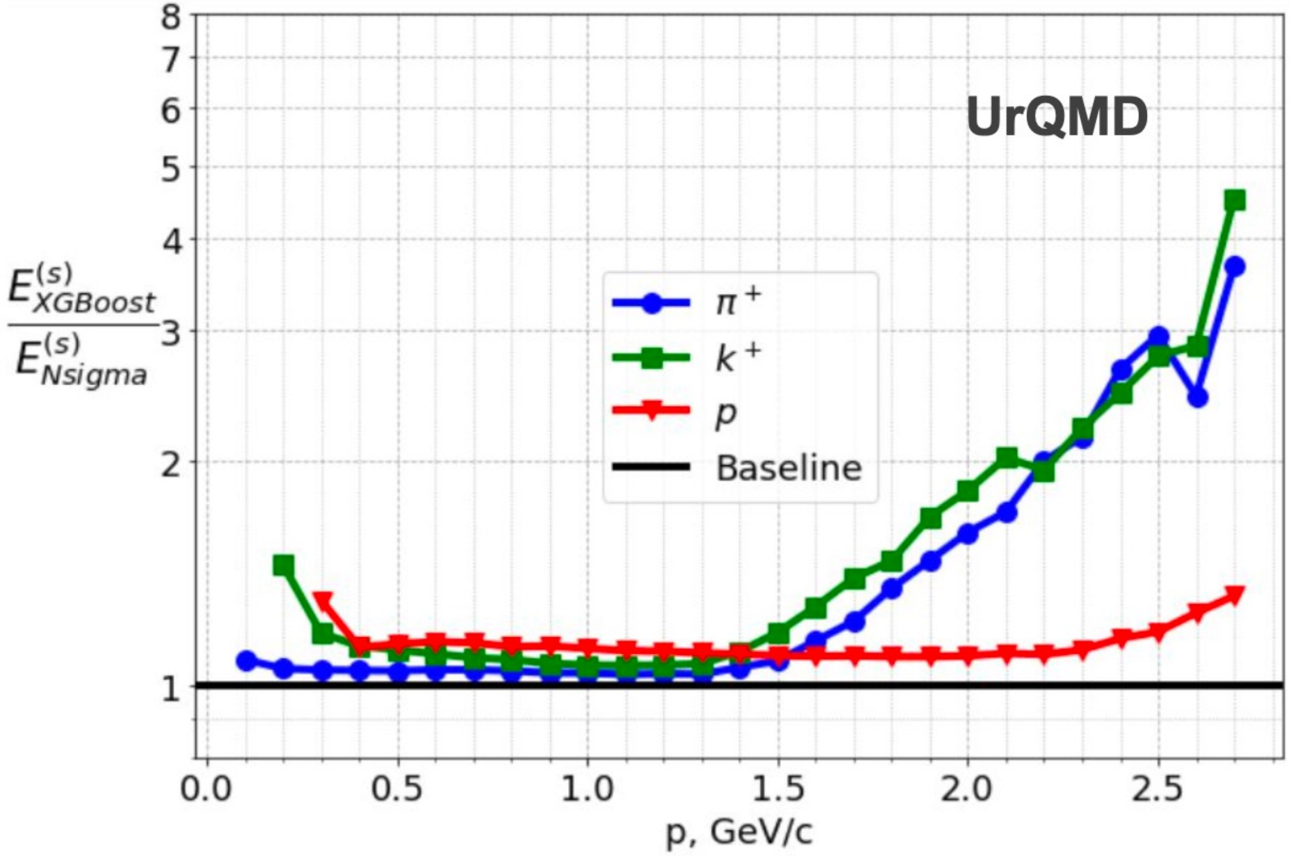
COMPARATIVE ANALYSIS OF THE ALGORITHMS. TIMING

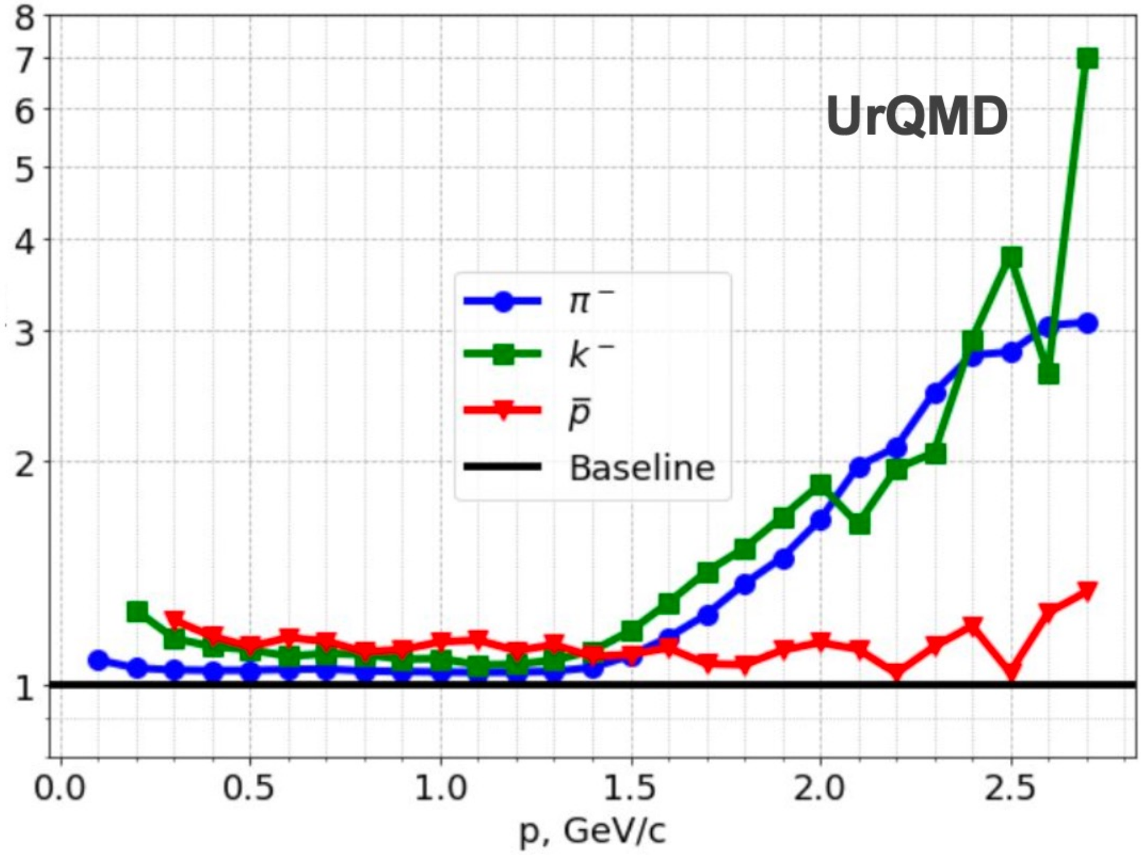


GPU: Nvidia Tesla V100-SXM2 NVLink 32GB HBM2

CPU: Intel Xeon Gold 6148 CPU @ 2.40 GHz 20 Cores / 40 Threads

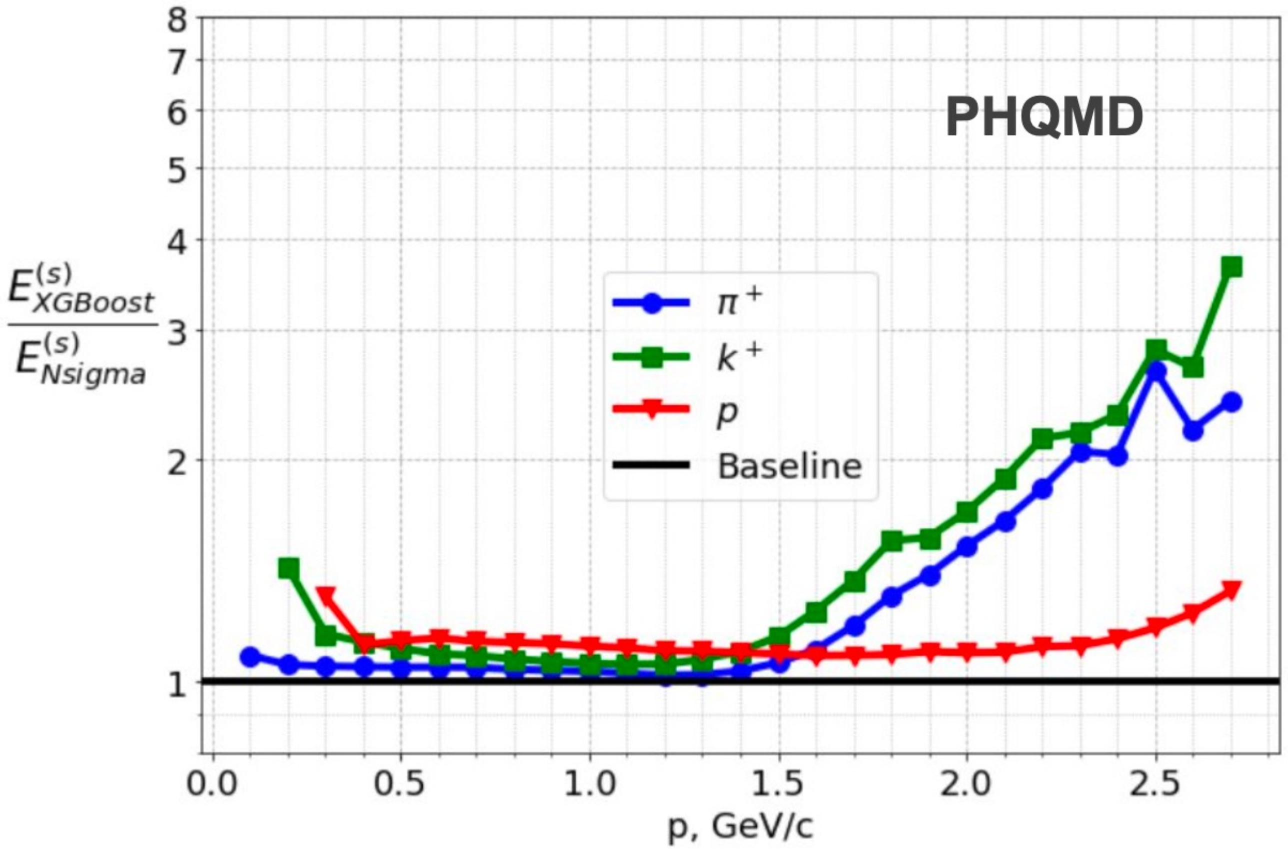
Comparison with N-sigma

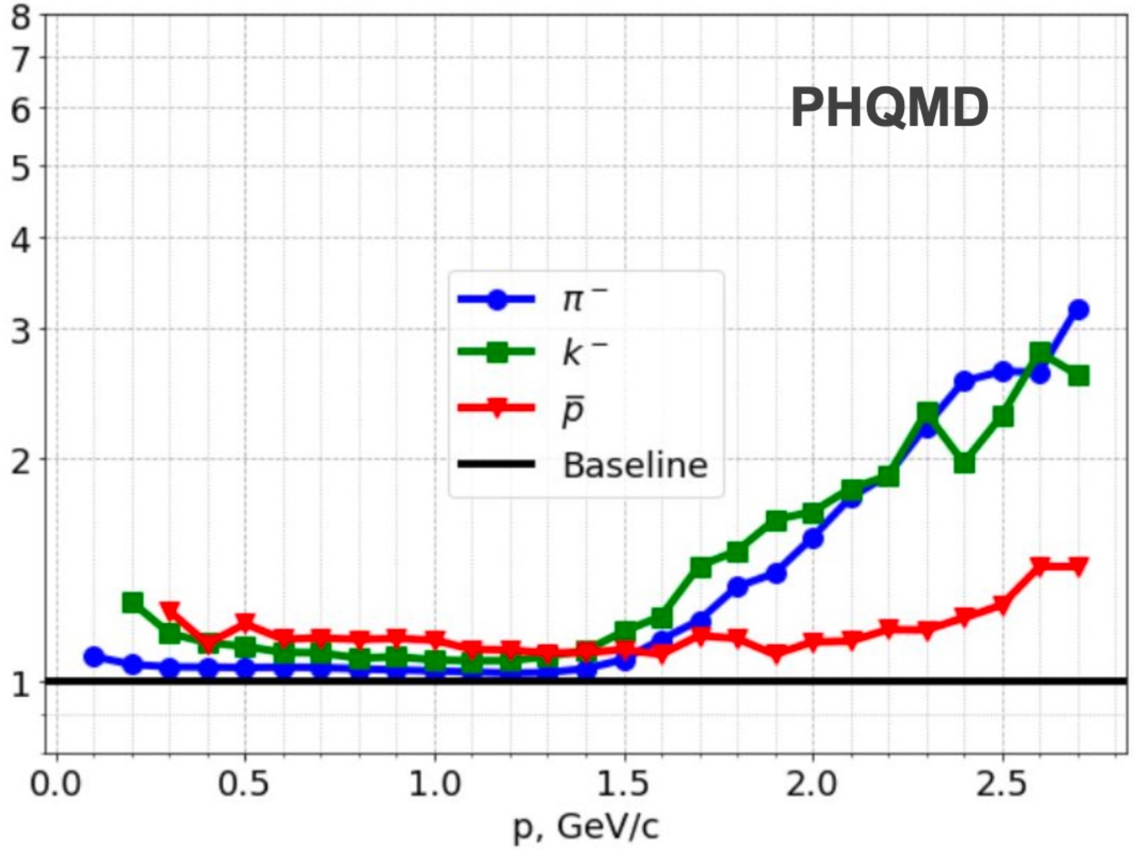




Efficiency ratio of XGBoost and n-sigma method

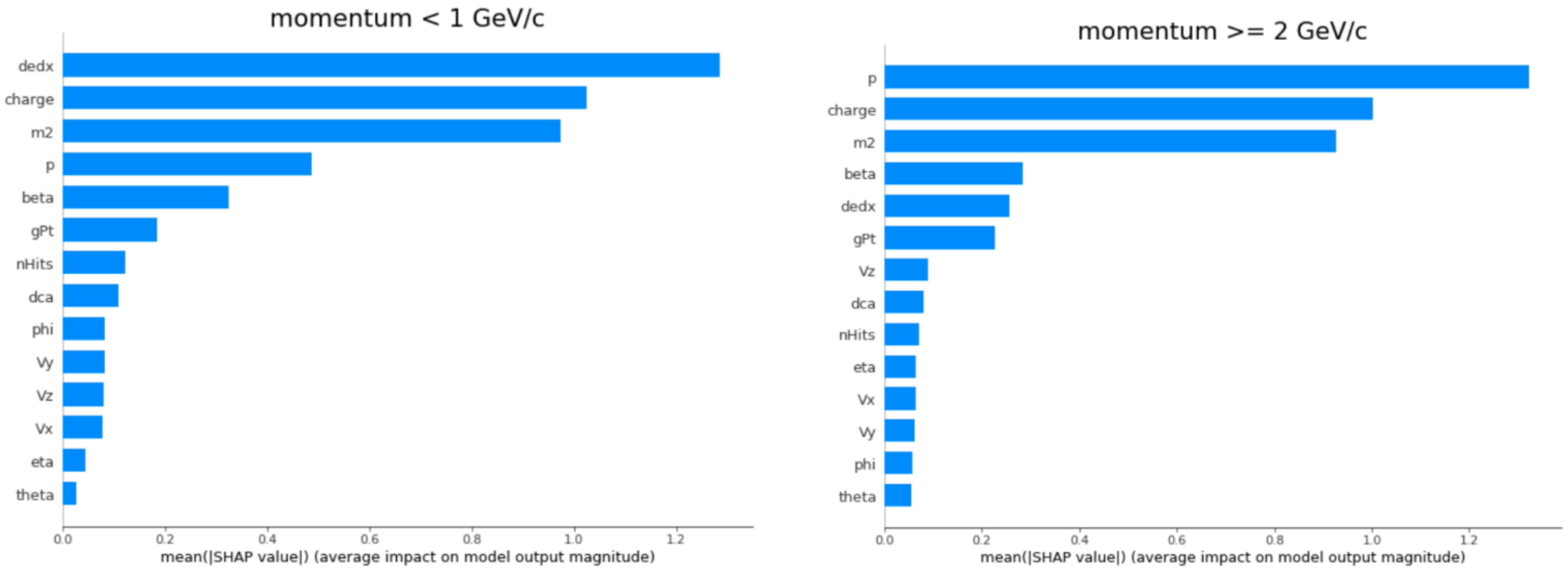
COMPARISON WITH N-SIGMA



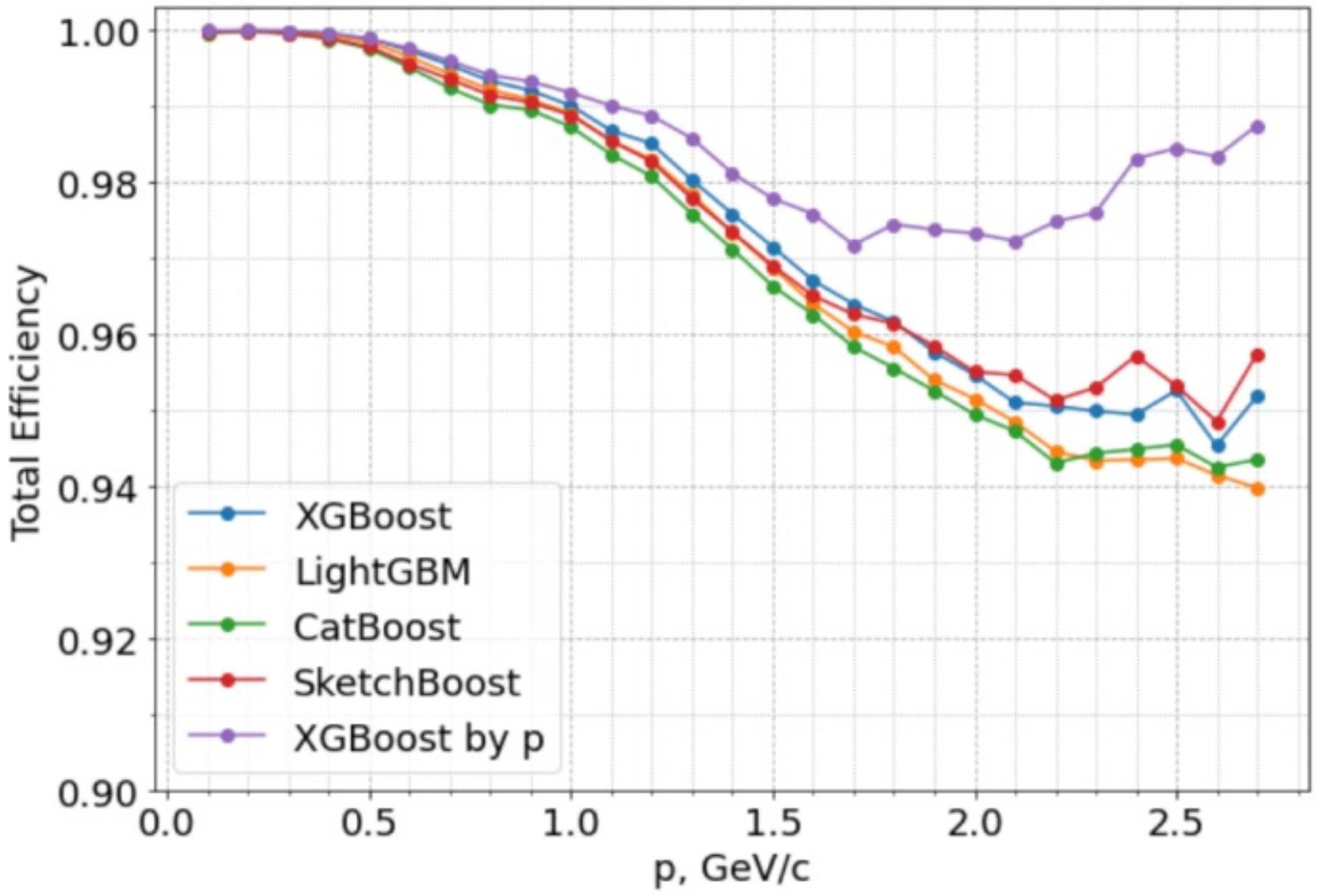


Efficiency ratio of XGBoost and n-sigma method

XGBoost Model Interpretation. Feature Importance



Final Efficiency of XGBoost



CONCLUSION AND OUTLOOKS

In general XGBoost has been demonstrated highest PID efficiency in comparison with considered algorithms of GBDT.

Next we are going to do additional testing to characterize identification stability of the model on data produced with different initial parameters of generated MC tracks at the MPD detector;

Also we are going to analyze the nature of the misclassifications and investigate the class imbalance problem.