Classifiers for centrality determination in proton-nucleus and nucleus-nucleus collisions

Igor Altsybeev, Vladimir Kovalenko
Saint-Petersburg State University

Quark Confinement and the Hadron Spectrum XII
September 1, 2016

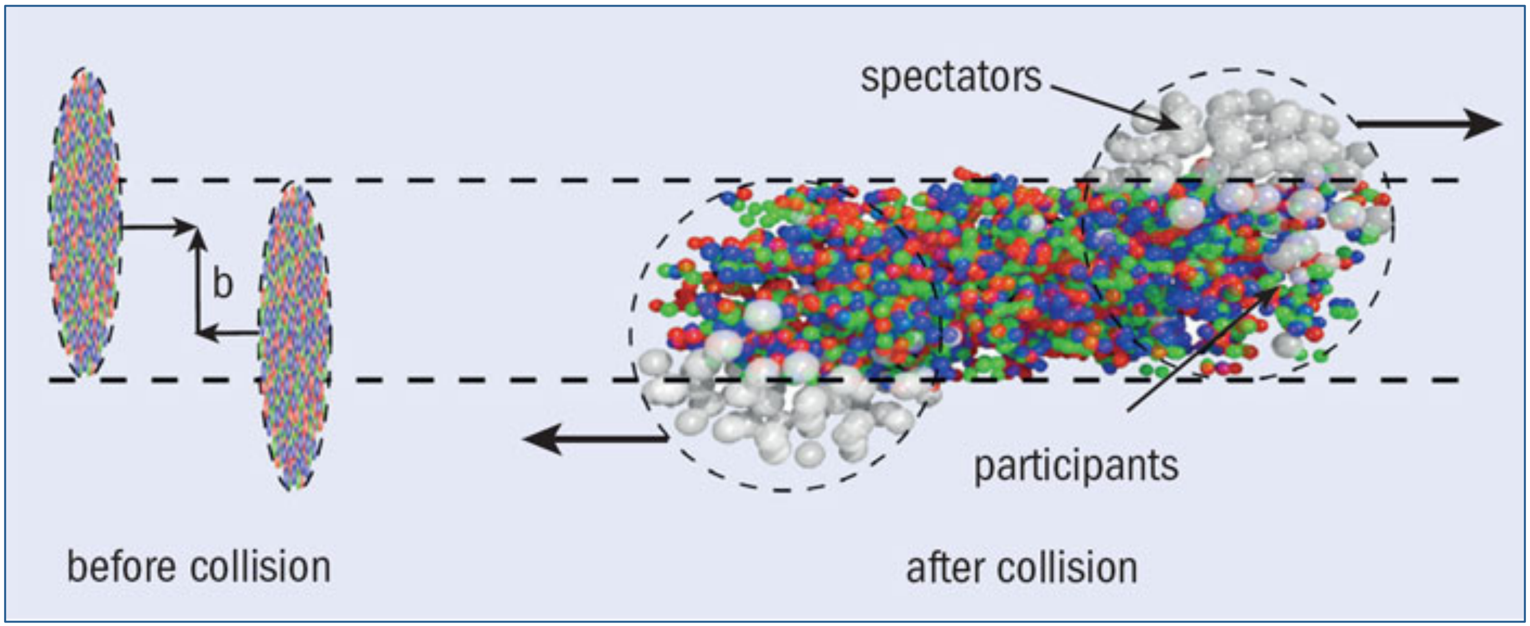
This work is supported by the Russian Science Foundation, GRANT 16-12-10176.

Prologue

- ML usage in HEP so far:
  search for rare decays, detector response optimization
- Interesting to find new applications
  - try to address centrality determination in heavy ion events

The collision centrality

The centrality is a key parameter in the study QCD matter at extreme energy densities, because it is directly related to the initial overlap region of the colliding nuclei.



The *impact parameter (b)* is the distance between the centers of the colliding nuclei.

Centrality determination in experiment

Usual receipt:

- use distribution of a signal in some detector
- fit with some geometry-based model
- split into *centrality classes* (0-100%)

In ALICE for Pb-Pb:

use *multiplicity distribution* in (semi-central) **VZERO detector** + Glauber fit

(-3.7<η<-1.7 and 2.8<η<5.1)

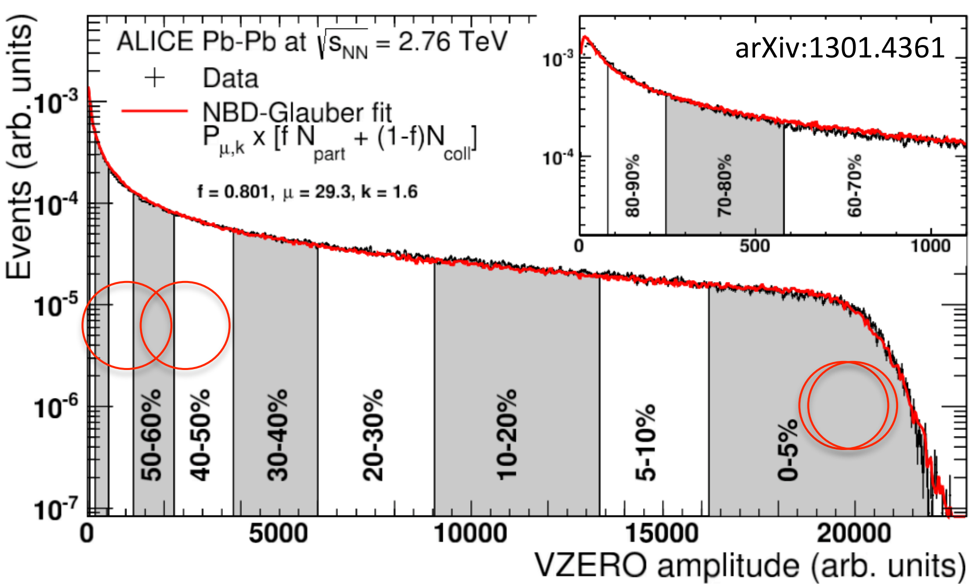
Centrality determination in experiment

Usual receipt:

- use distribution of a signal in some detector
- fit with some geometry-based model
- split into *centrality classes* (0-100%)

In ALICE for Pb-Pb:

use *multiplicity distribution* in (semi-central) **VZERO detector** + Glauber fit



(-3.7<η<-1.7 and 2.8<η<5.1)

close to 0% → most **central** events

closer to 100% → **peripheral** events

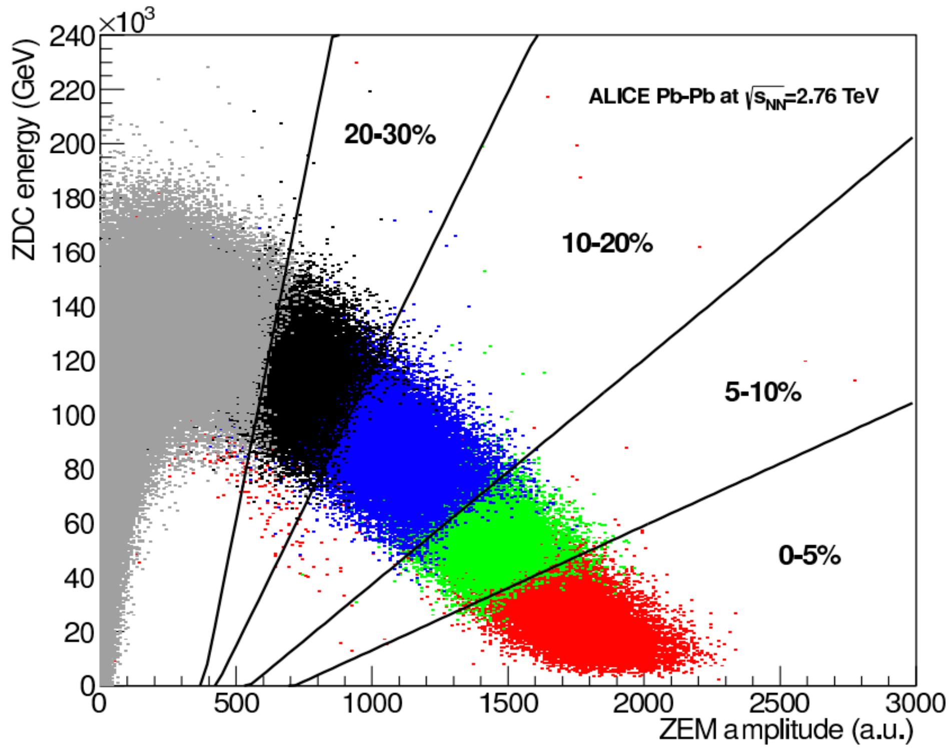
→ b_{impact}, N_{part}, N_{coll}, N_{spec} are not directly measurable and are deduced from the Glauber model.

Alternative method in ALICE

Based on 2D distribution of signals from two detectors:

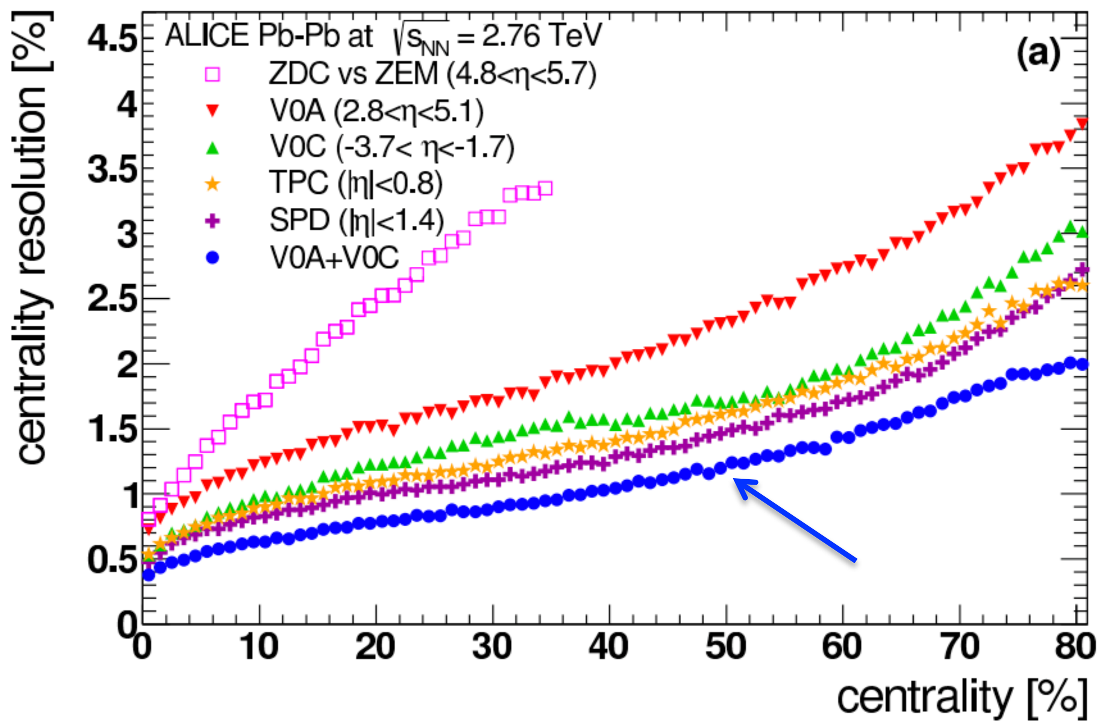
energy from neutrons-spectators in Zero-degree calorimeters (**ZDC**) && signals in electromagnetic calorimeters (**ZEM**)

(4.8<η<5.7)



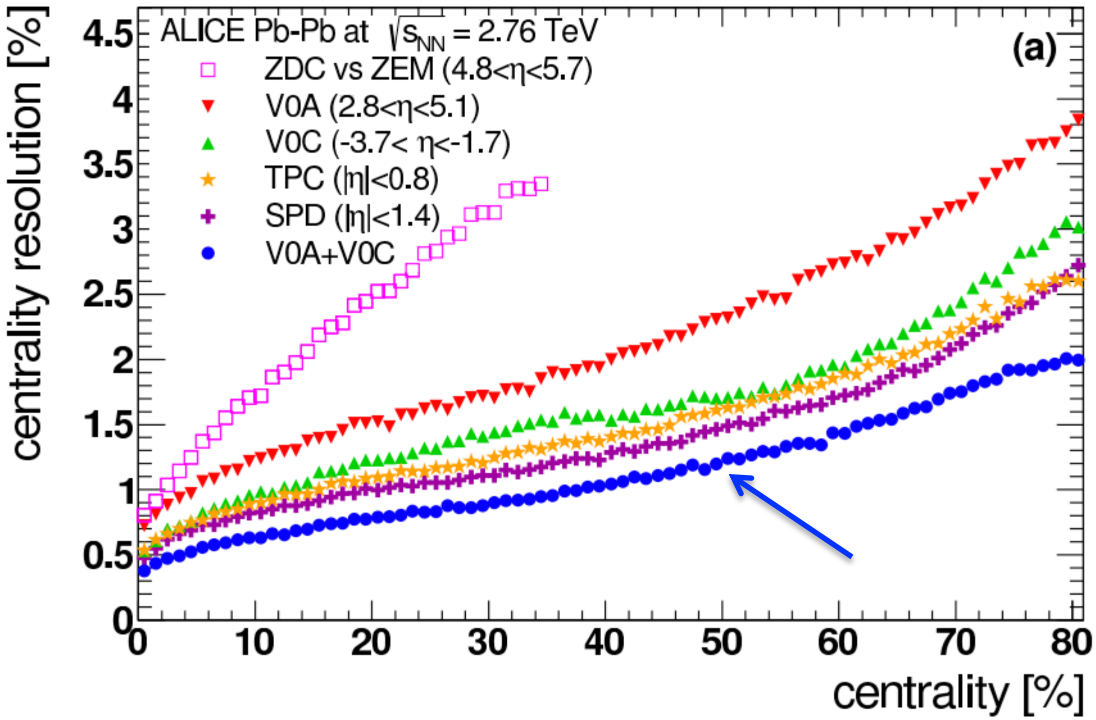
Splitting in centrality classes is done by drawing (arbitrary) lines.

Centrality resolution in experiment



Best centrality resolution is achieved by usage of the VZERO estimator.

Centrality resolution in experiment



Best centrality resolution is achieved by usage of the VZERO estimator.

Can we perform better using multiple detectors simultaneously?
Try machine-learning techniques for that.

Centrality determination as a Machine-Learning task

- Use AMPT monte-carlo generator to simulate Pb-Pb events at 2.76 TeV (without detector response) (400k events)
- 5 features are selected in correspondence with the subsystems of the ALICE detector:

ZDC-C
N neutrons

VZERO-C
multiplicity
$-3.7 < \eta < -1.7$

TPC
multiplicity
$|\eta| < 0.8$

VZERO-A
multiplicity
$2.8 < \eta < 5.1$

ZDC-A
N neutrons

Centrality determination as a Machine-Learning task

- Use AMPT monte-carlo generator to simulate Pb-Pb events at 2.76 TeV (without detector response) (400k events)
- 5 features are selected in correspondence with the subsystems of the ALICE detector:

VZERO-C
multiplicity
$-3.7 < \eta < -1.7$

TPC
multiplicity
$|\eta| < 0.8$

VZERO-A
multiplicity
$2.8 < \eta < 5.1$

ZDC-C
N neutrons

ZDC-A
N neutrons

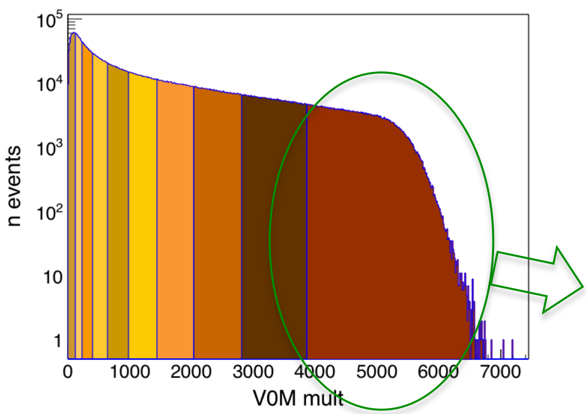
+3 additional features:
- average p_T
- kaon/pion ratio
- proton/pion ratio

Target for regression: **impact parameter *b***

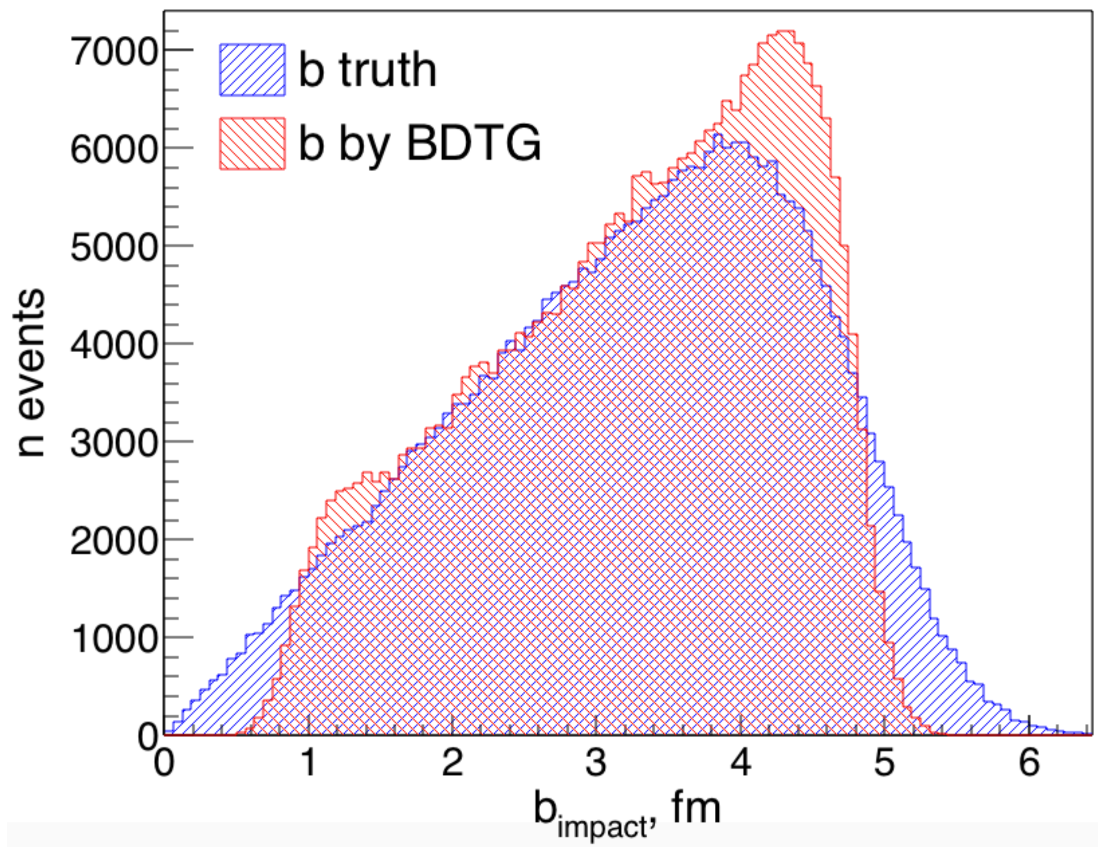
(using TMVA Version 4.2.0)

Target for regression: **impact parameter *b***

pre-selection:
events class **0-10%** (V0M)

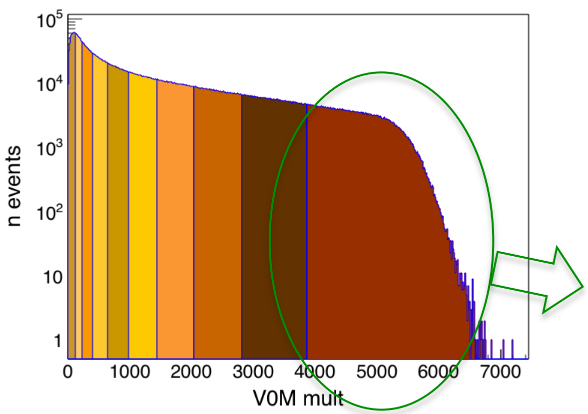


True distribution *VS* estimated by Boosted Decision Tree:

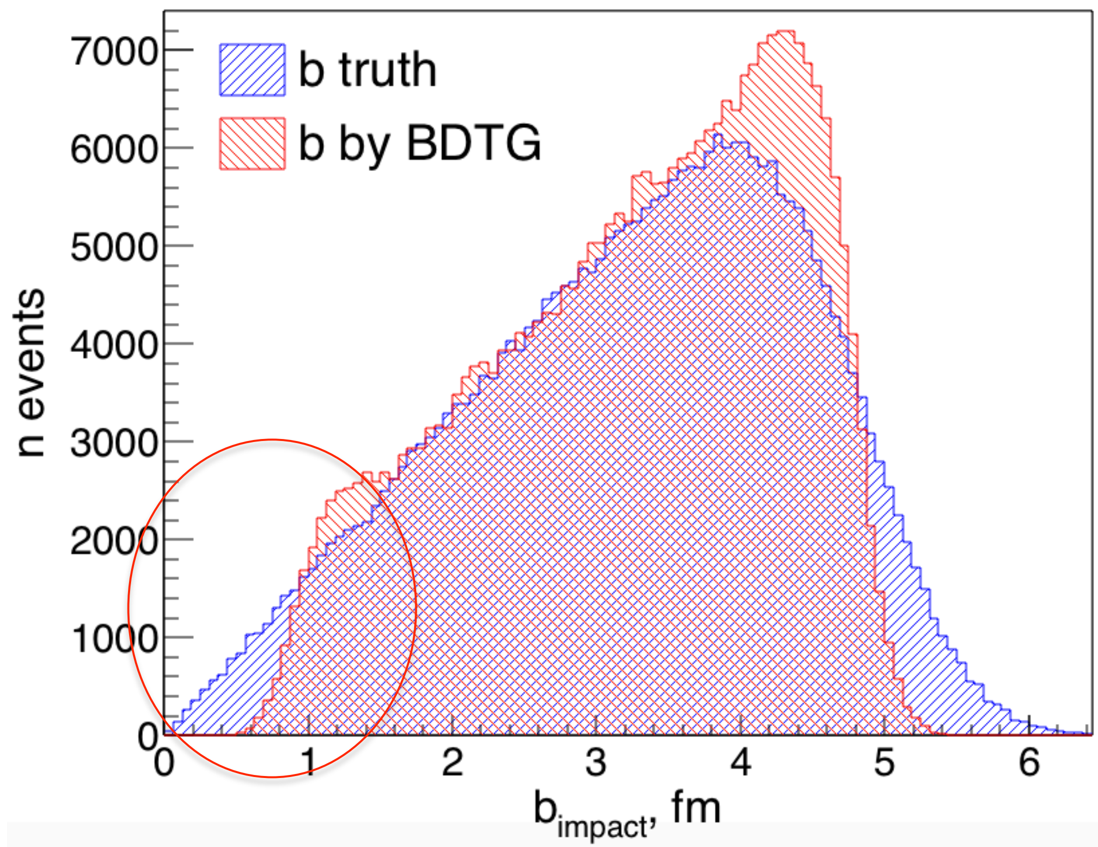


Target for regression: **impact parameter *b***

pre-selection:
events class **0-10%** (V0M)



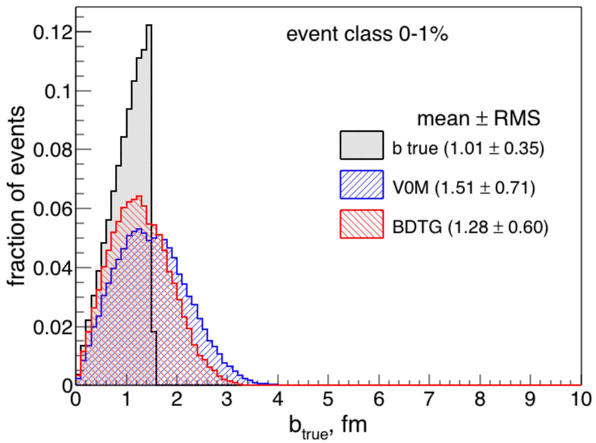
True distribution *VS* estimated by Boosted Decision Tree:

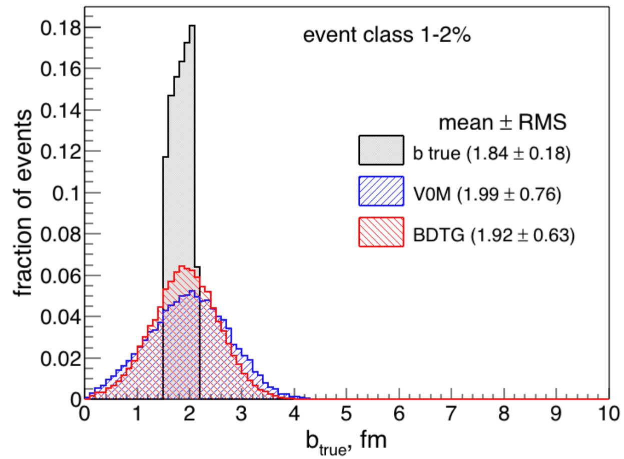


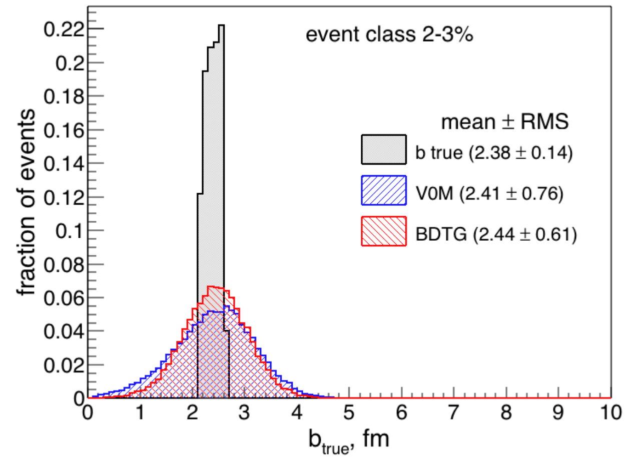
Regression by ML estimator: unavoidable "deformation" of the *b*-distribution at the edges.

ML estimator *vs* V0M-based selection

- Split the regression output of the estimator into centrality classes of 1% width
- Look at *b impact* distributions:

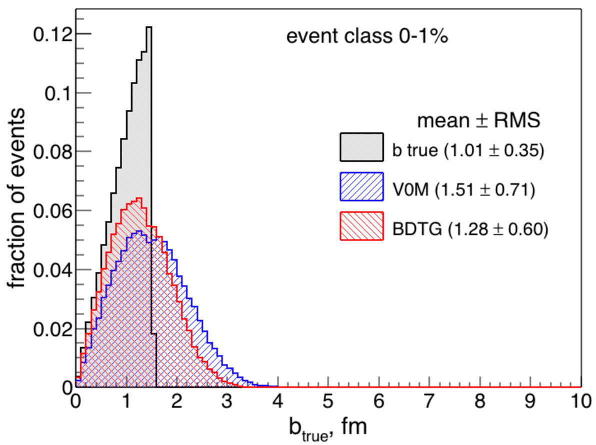


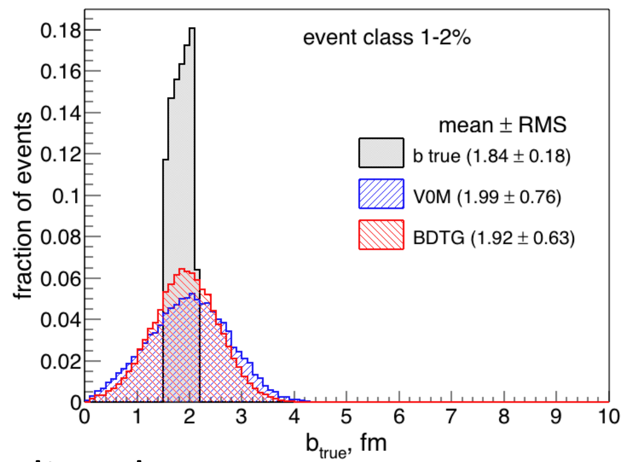


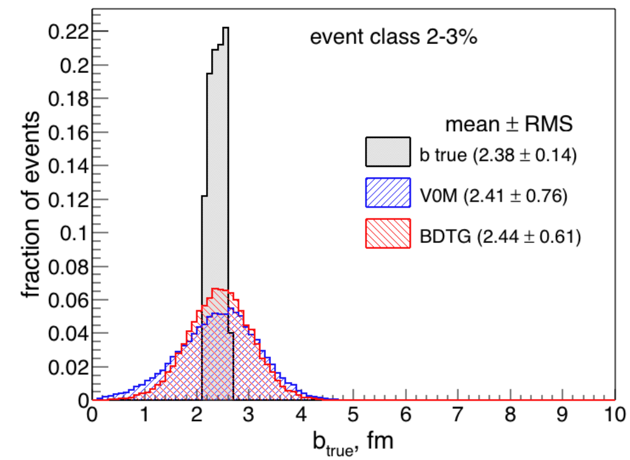


ML estimator *vs* V0M-based selection

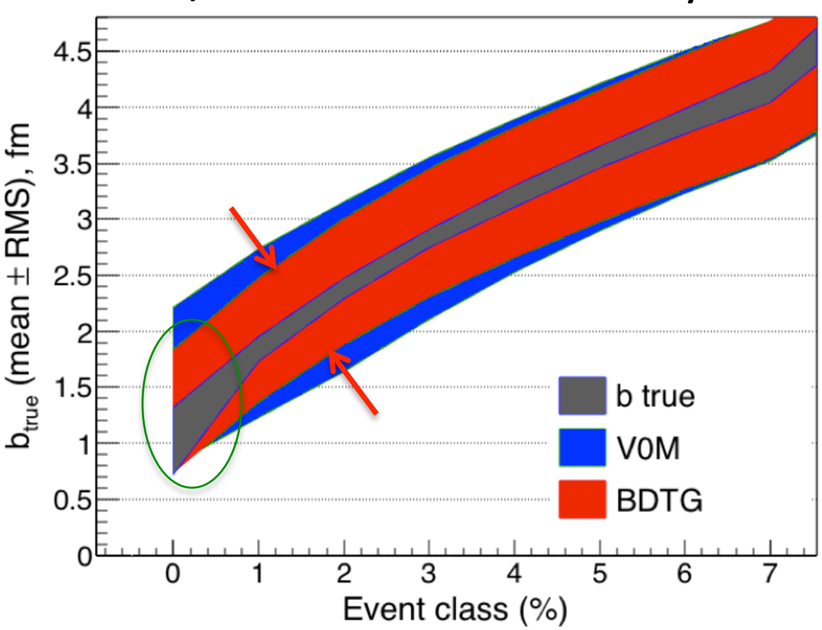
- Split the regression output of the estimator into centrality classes of 1% width
- Look at *b impact* distributions:







Mean +/- RMS within centrality classes:



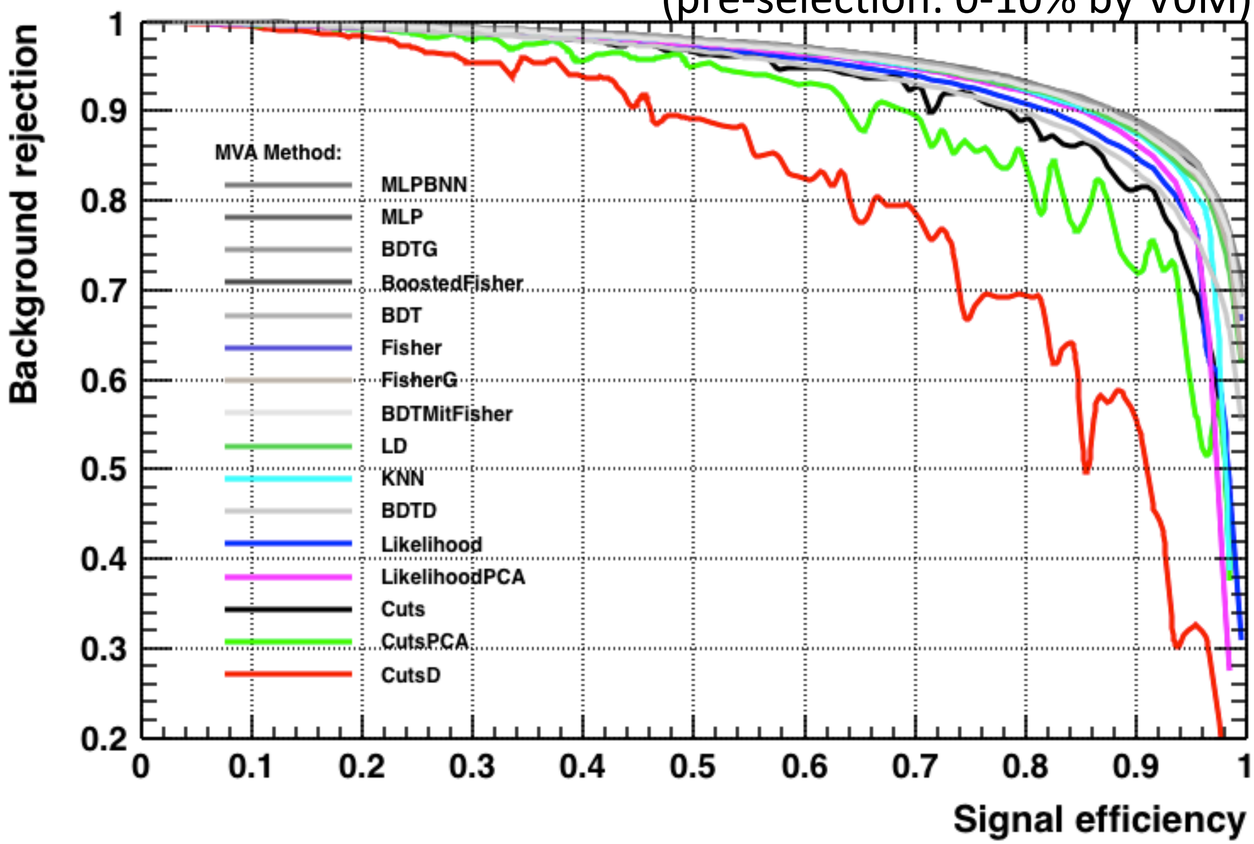
- *narrower* RMS by ML estimator than by V0M
- most central 0-1% events by estimator are *closer to truth* b

Study in terms of $\mathbf{N_{part}}$ should go even closer to truth!

Now try <u>classification</u> task: find *most central* events

signal = 0-1% most central events (b_{impact} < 1.5 fm)
background = other events

(pre-selection: 0-10% by V0M)



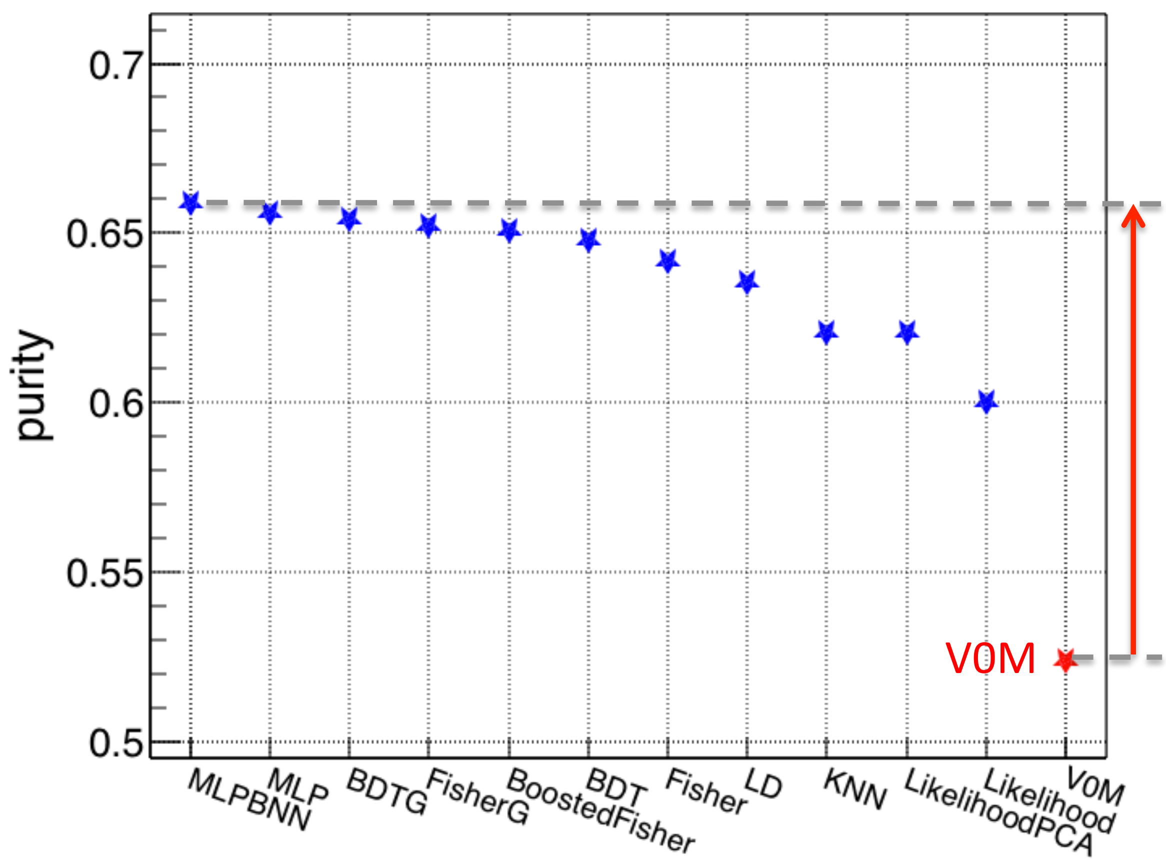
90k events

- Similar performance for the most of estimators
- Cut-based classifiers perform worse than others

Fraction of "signal" events selected by classifiers output

(="purity")

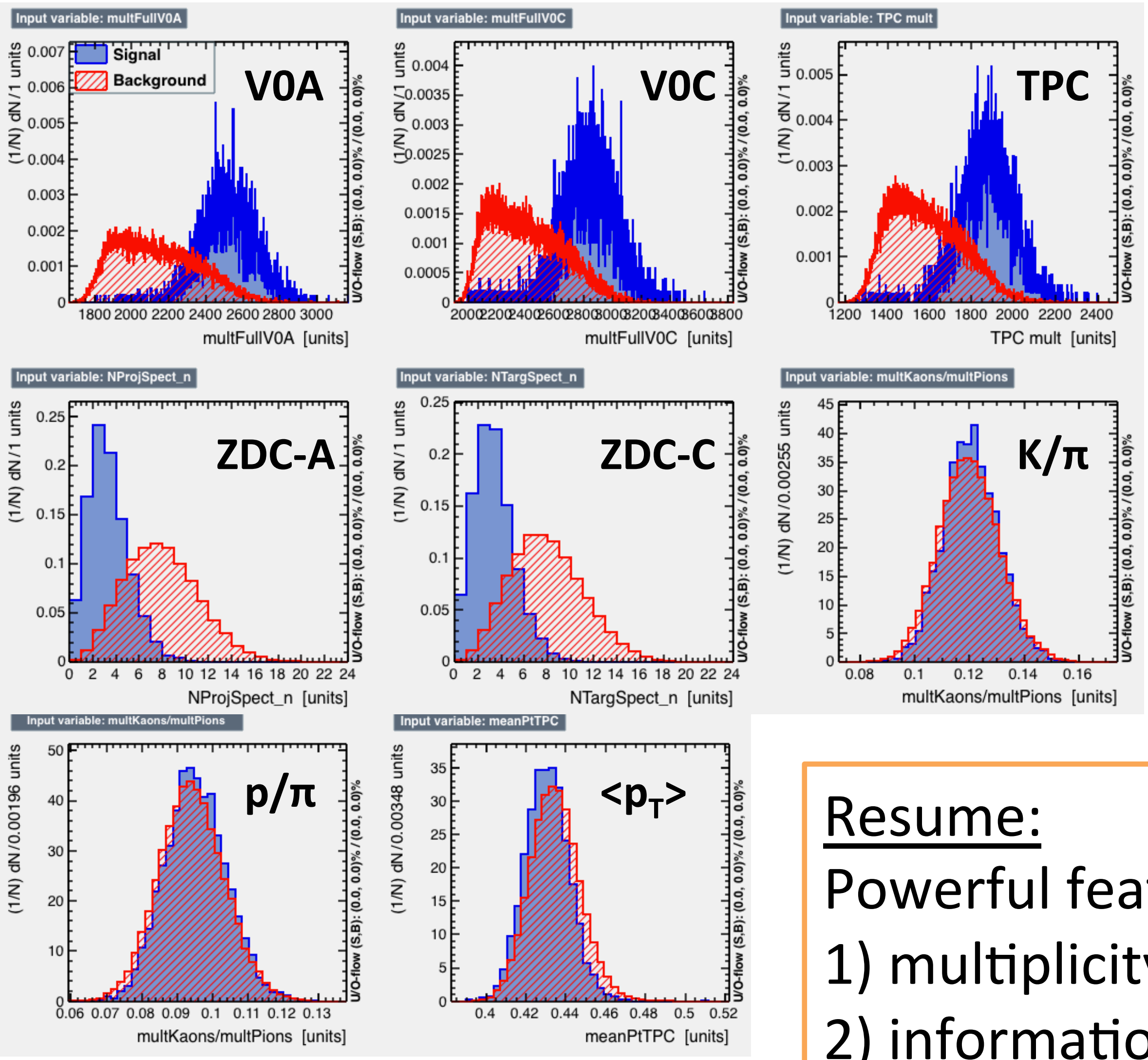
signal = 0-1% most central events (b_{impact} < 1.5 fm)

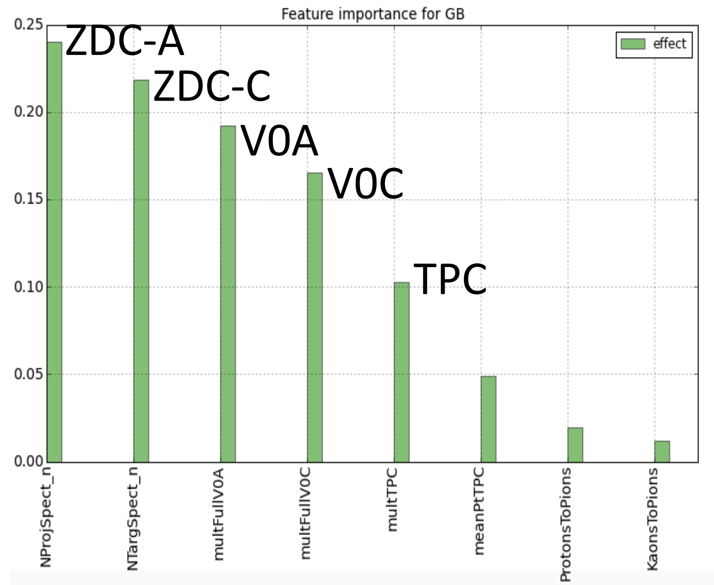


Compare to V0M estimator: purity is increased by ~13%.

Significant gain of combined usage of information from several sub-detectors.

Which features are the most relevant?





<u>Resume:</u>

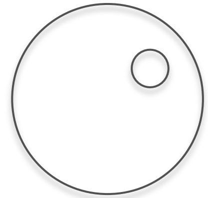
Powerful features are:

1) multiplicity in large rapidity ranges

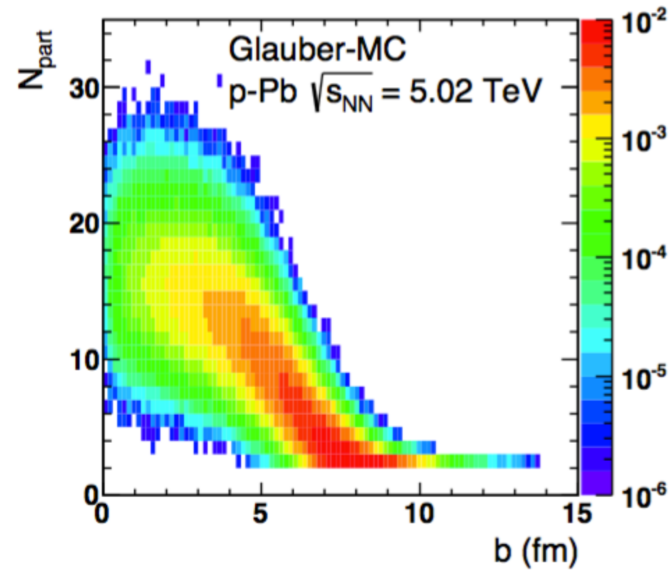
2) information about spectators

Additional features do not help.

From Pb-Pb to p-Pb collisions



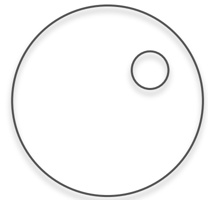
centrality in p-Pb (ALICE)
arXiv:1412.6828



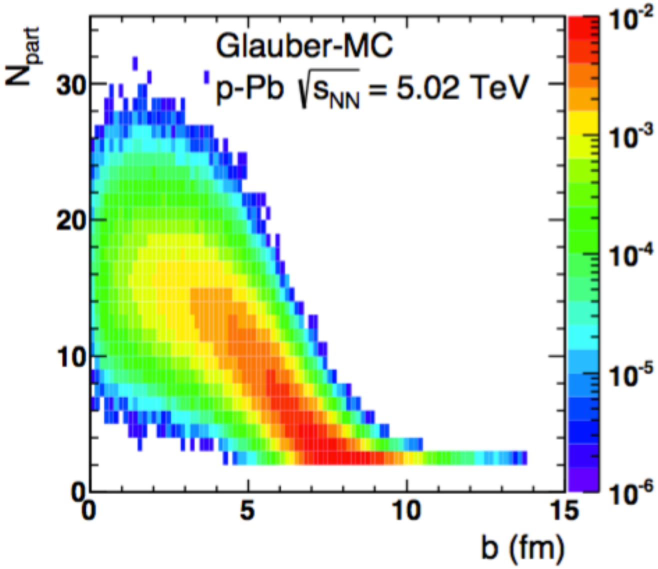
p-Pb collisions:

- larger fluctuations in Npart
- More bias from multiplicity-based estimators
- N_{part} is more reliable *target* than b_{impact}

From Pb-Pb to p-Pb collisions



centrality in p-Pb (ALICE)
arXiv:1412.6828



p-Pb collisions:

- larger fluctuations in Npart
- More bias from multiplicity-based estimators
- N_{part} is more reliable *target* than b_{impact}

Try ML classification task with N_{part} as a target:

split all events into 5 classes
0-20, 20-40, 40-60, 60-80, 80-100%,
0-20% = most central
How often do we get true N_{part} in 0-20% using estimator output?

Use for that 5 *mln* AMPT p-Pb events

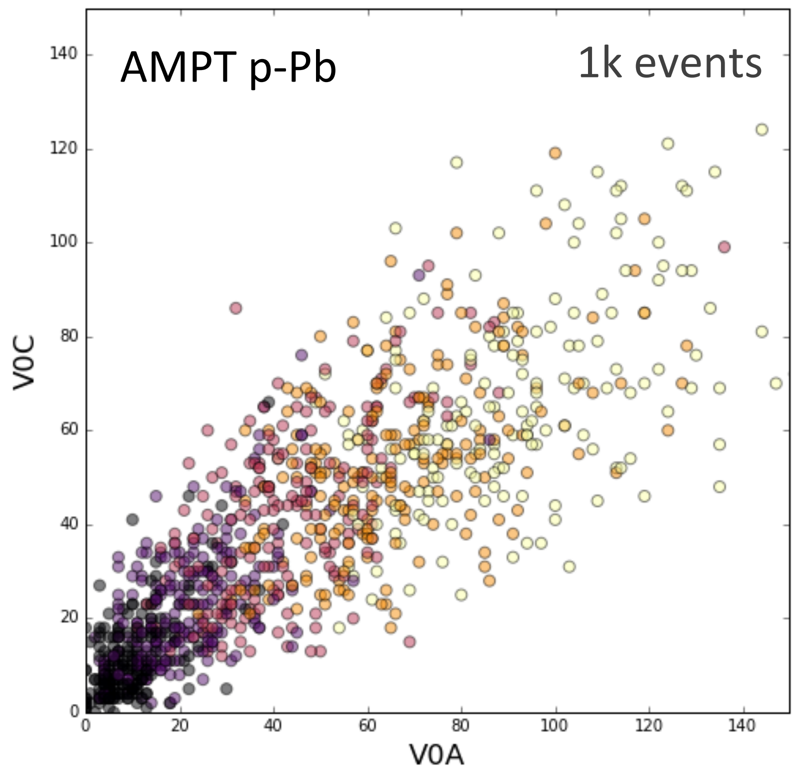
Visualize decision boundaries

p-Pb

Two features: mult in VZERO-C && mult in VZERO-A

$-3.7 < \eta < -1.7$ $\quad$ $2.8 < \eta < 5.1$

Target: N_{part}



Colors:
5 true centrality classes

Visualize decision boundaries

p-Pb

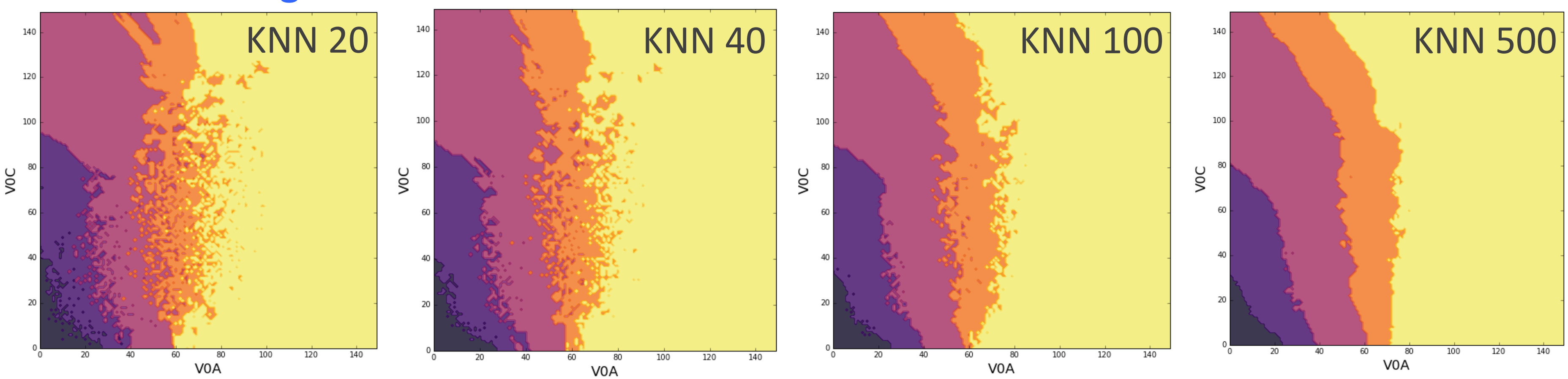
Two features: mult in VZERO-C && mult in VZERO-A

$-3.7 < \eta < -1.7$ $\quad$ $2.8 < \eta < 5.1$

Target: N_{part}

(using scikit-learn library)

K-nearest neighbors:



Visualize decision boundaries

p-Pb

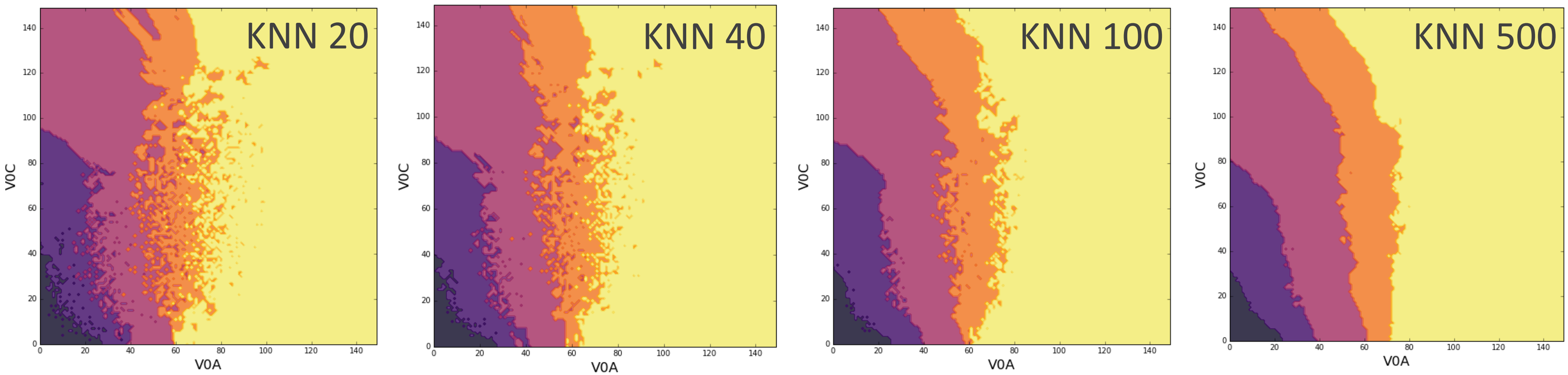
Two features: mult in VZERO-C && mult in VZERO-A

$-3.7 < \eta < -1.7$ $\quad$ $2.8 < \eta < 5.1$

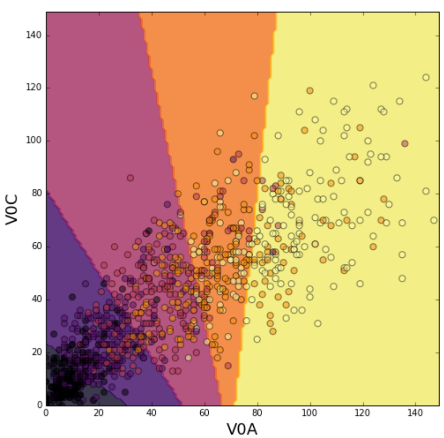
Target: N_{part}

(using scikit-learn library)

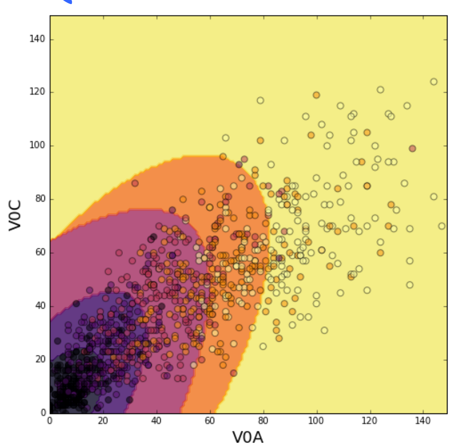
K-nearest neighbors:



Linear Discriminant:



Quadratic Discriminant:



Visualize decision boundaries (other features)

p-Pb

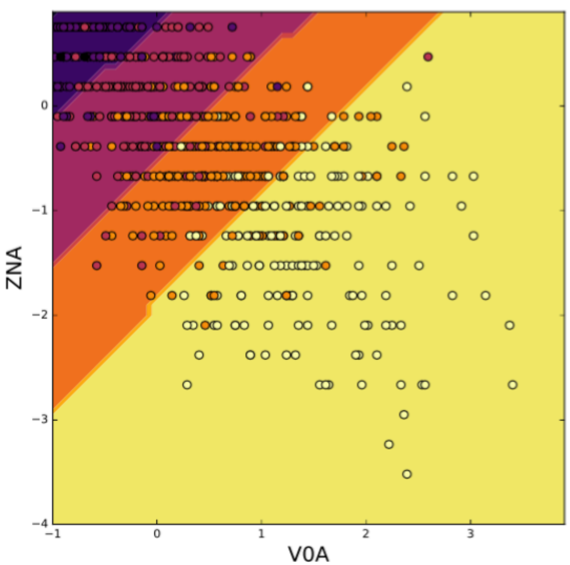
Two other features: mult in VZERO-A && ZDC-neutrons

Target: N_{part}

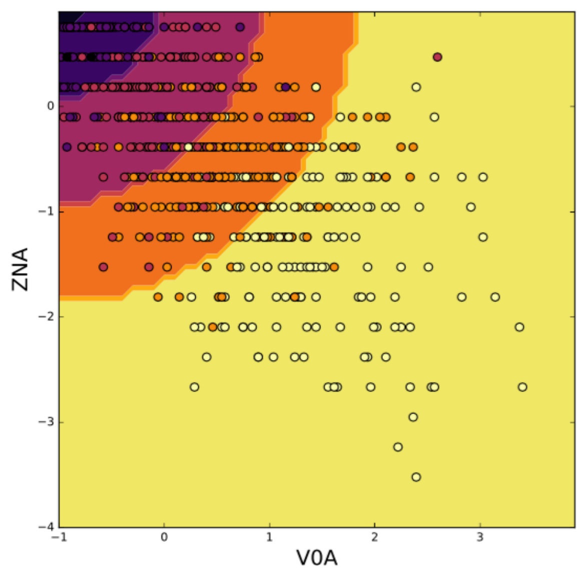
$2.8 < \eta < 5.1$

(using scikit-learn library)

Linear Discriminant:



Quadratic Discriminant:



Visualize decision boundaries (other features)

p-Pb

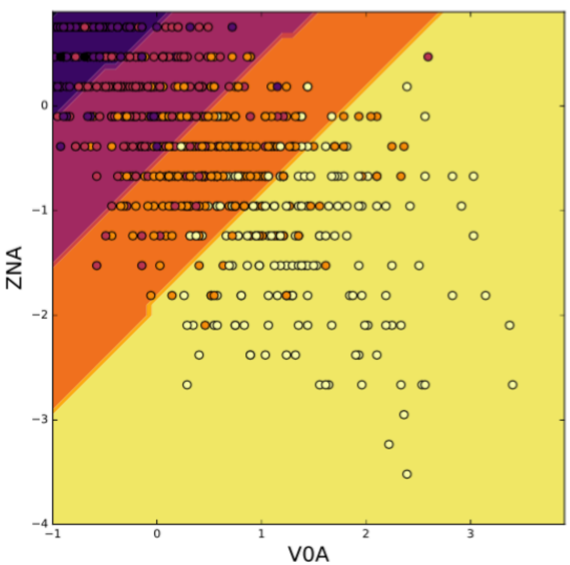
Two other features: mult in VZERO-A && ZDC-neutrons

Target: N_{part}

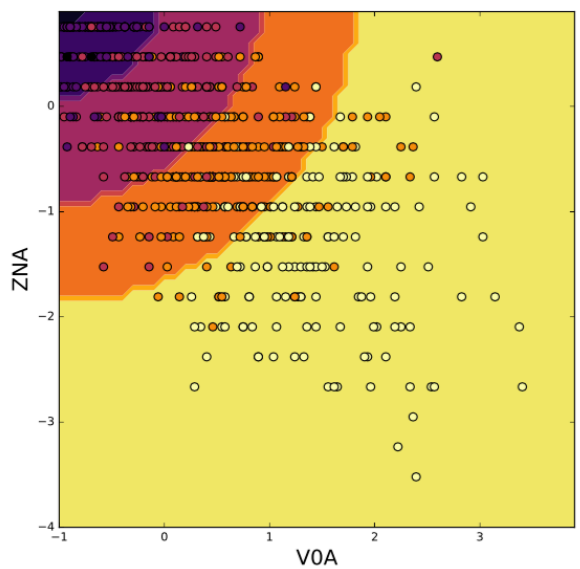
$2.8 < \eta < 5.1$

(using scikit-learn library)

Linear Discriminant:

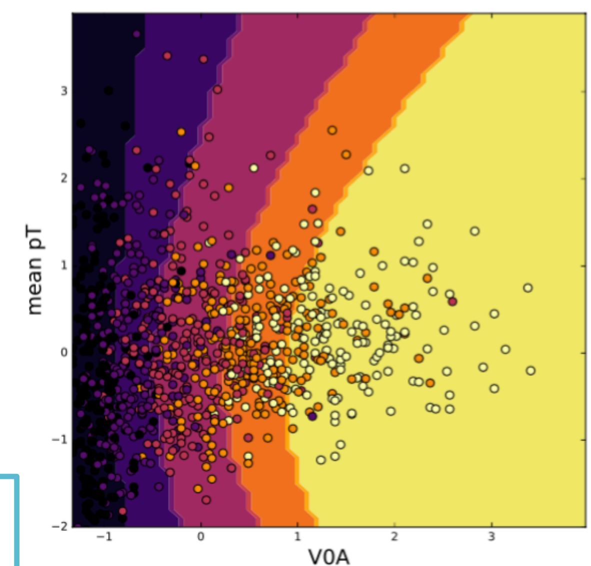


Quadratic Discriminant:



Try another feature: mean p_T (vs V0A)

Quadratic Discriminant:



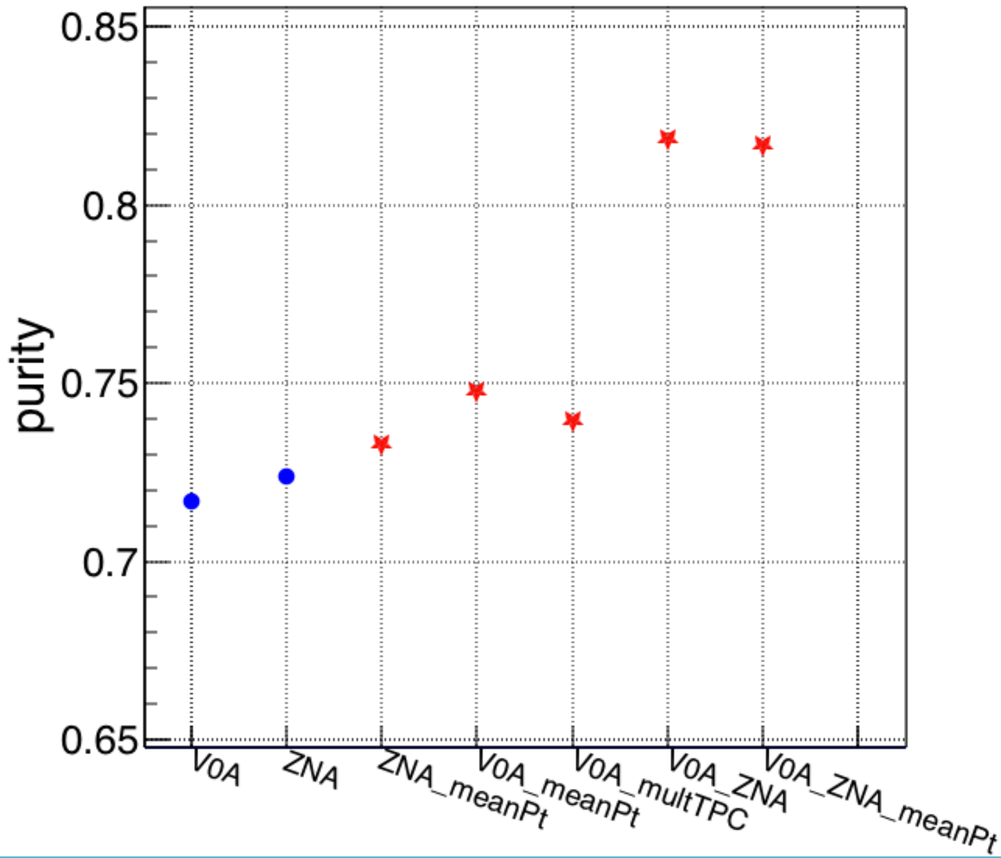
Task:
how often do we get true N_{part} in 0-20% using estimator output?

"Purity" of selection events in 0-20% by N_{part}

p-Pb

Compare different feature sets
basing on Quadratic Discriminant classification:

(target: N_{part})

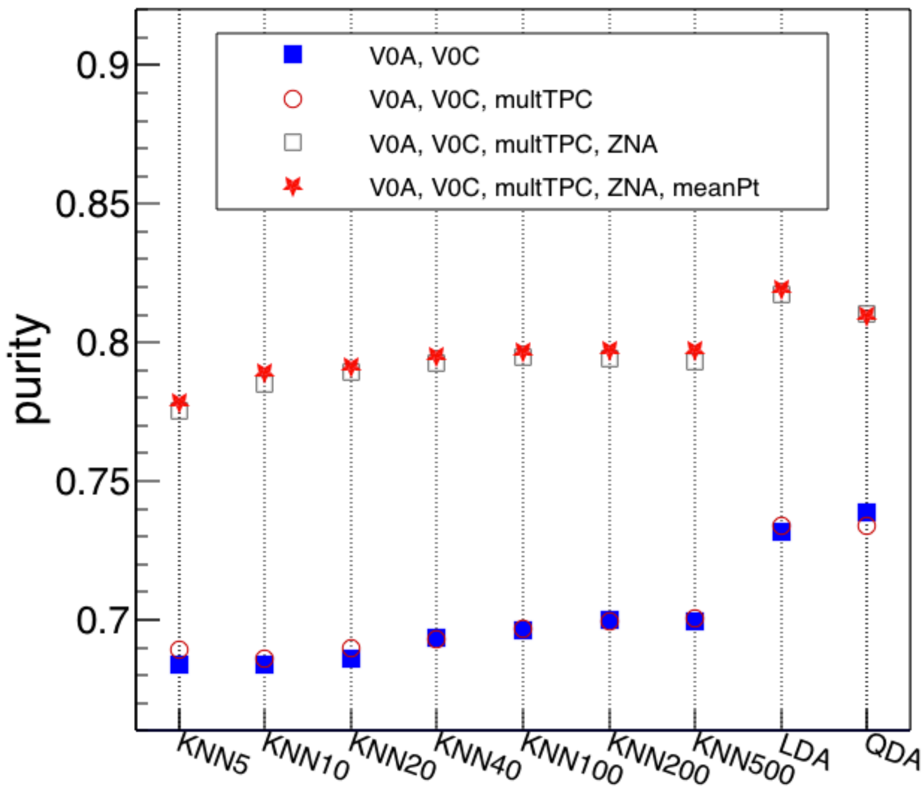


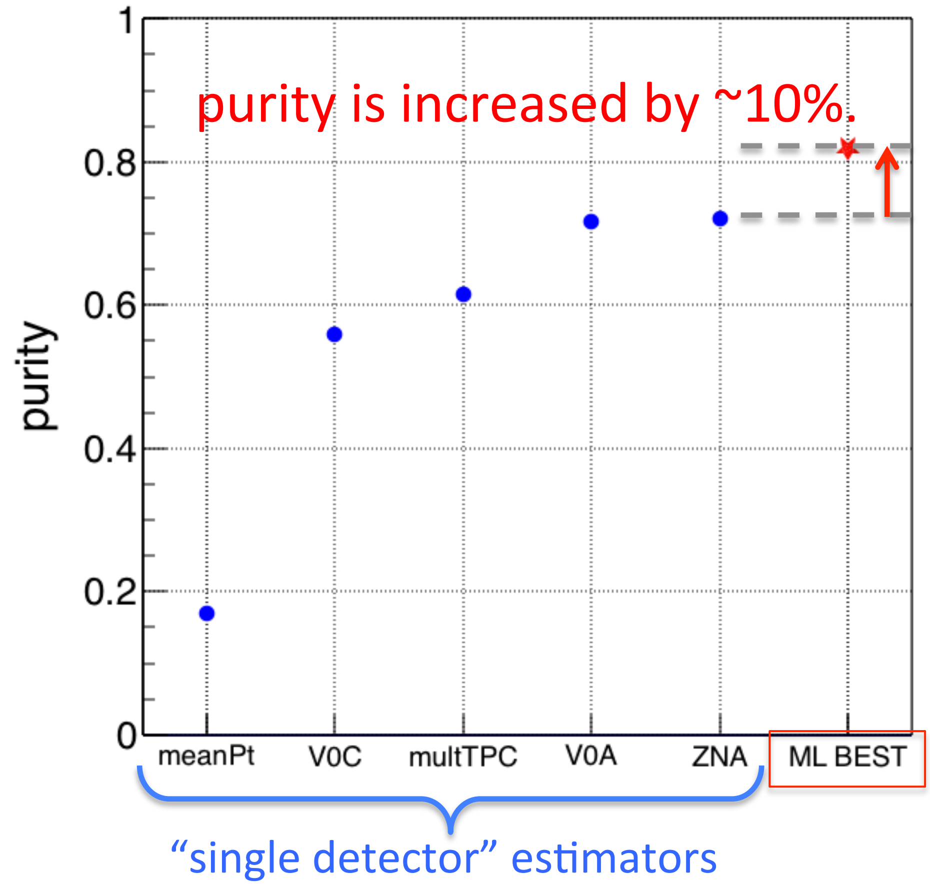
- Combination V0A && ZNA provides best results
  - Other features are almost irrelevant

"Purity" of selection events in 0-20% by N_{part}

p-Pb

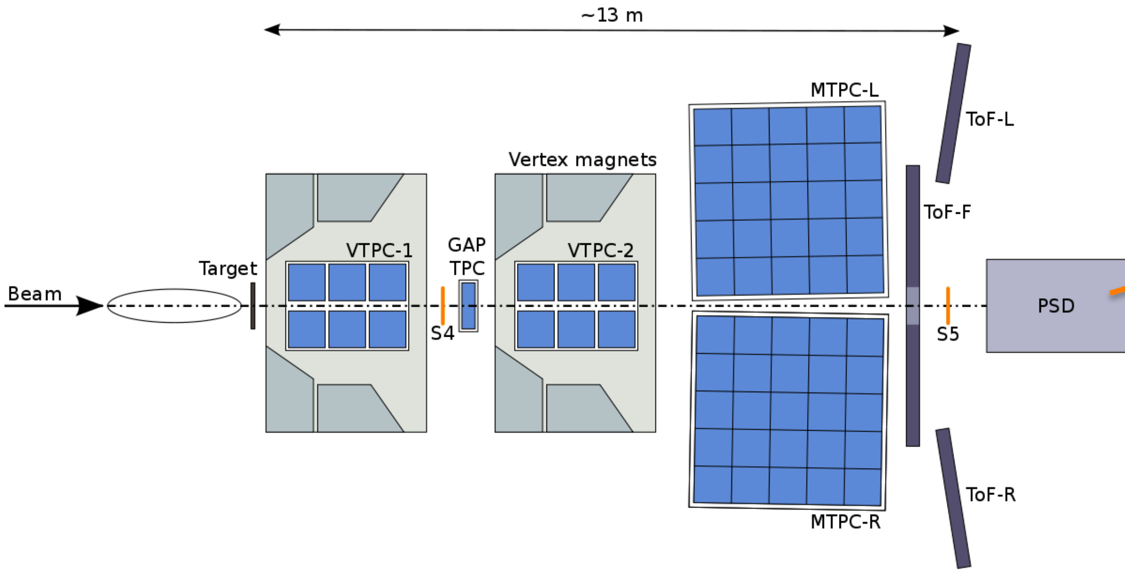
Different feature sets and several ML algorithms:



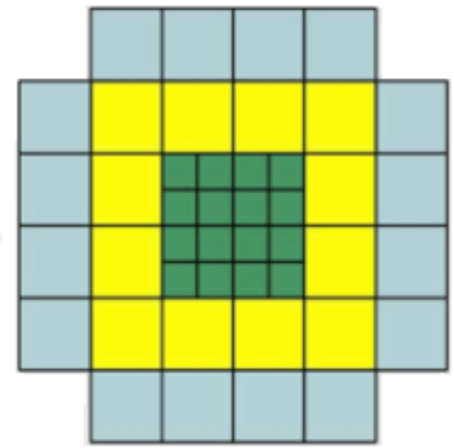


ML-based classification allowed to increase purity of 0-20% class in terms of N_{part} by ~10%.

Possible ML task in fixed target experiment NA61/SHINE



Projectile Spectator Detector (PSD)



http://shine.web.cern.ch

- Centrality in AA collisions in NA61 experiment is determined by energy in modules of PSD (possibly in combination with data from TPC's).
- Modules in PSD are fired by spectators **and** particles born in collision.

→ Possibilities to use ML classifiers to cope with these conditions and increase resolution of centrality selection?

Summary

Accurate centrality is a baseline for many physics analysis (crucial, for example, in fluctuations and correlations studies).

Presented work is exploratory to demonstrate possible usage of ML techniques for classification of events in terms of centrality.

Summary

Accurate centrality is a baseline for many physics analysis (crucial, for example, in fluctuations and correlations studies).

Presented work is exploratory to demonstrate possible usage of ML techniques for classification of events in terms of centrality.

- We want to go back to native definition of the notion of “centrality”: through the *impact parameter and/or N nucleons-participants.*
- ML algorithms are able to select most central events with higher “purity” without loss of statistics using info from several detectors.
- In this exploratory work, increase in purity is ~10-13%
  → possible to gain more?.
- Combination of several strong features is enough, “weak” features are irrelevant.
- If use in real life → need to tune MC to match real detector data.

Summary

Accurate centrality is a baseline for many physics analysis (crucial, for example, in fluctuations and correlations studies).

Presented work is exploratory to demonstrate possible usage of ML techniques for classification of events in terms of centrality.

- We want to go back to native definition of the notion of "centrality": through the *impact parameter and/or N nucleons-participants.*
- ML algorithms are able to select most central events with higher "purity" without loss of statistics using info from several detectors.
- In this exploratory work, increase in purity is ~10-13%
  → possible to gain more?.
- Combination of several strong features is enough, "weak" features are irrelevant.
- If use in real life → need to tune MC to match real detector data.

Thanks for your attention!