STAR Forward Tracking & Optimization with Pythia8 Events

Youqi Song

Yale University

Outline

- Pythia 8 simulation details
- Boosted-Decision Tree tracking optimization
- Plans and next steps
- Summary

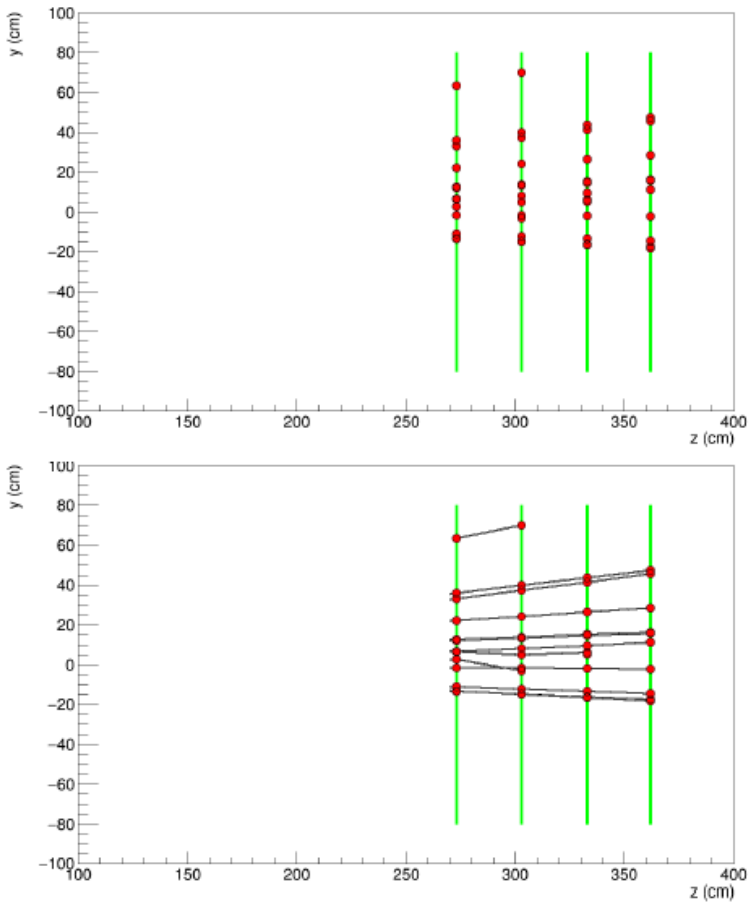
Pythia 8 Studies: Simulation Details

- Pythia 8 pp Drell-Yan events at $\sqrt{s}$ = 510 GeV
- Vertex: (x, y, z) = (0.1, -0.1, 0) cm, (σ_x, σ_y , σ_z)= (0.1, 0.1, 30) cm
- 50,000 events
- /gpfs01/star/pwg/youqi/runPythia/out

```
*-------  PYTHIA Flag + Mode + Parm + Word Settings (changes only)  --------
|
| Name                                          |                     Now |
|                                               |                         |
| BeamRemnants:reconnectRange                   |                 1.50000 |
| Beams:eCM                                     |               510.00000 |
| MultipartonInteractions:alphaSvalue           |                 0.13500 |
| MultipartonInteractions:bProfile              |                       3 |
| MultipartonInteractions:ecmPow                |                 0.19000 |
| MultipartonInteractions:expPow                |                 2.00000 |
| MultipartonInteractions:pT0Ref                |                 2.08500 |
| PDF:pSet                                      |                       8 |
| SigmaDiffractive:dampen                       |                      on |
| SigmaDiffractive:maxAX                        |                65.00000 |
| SigmaDiffractive:maxXB                        |                65.00000 |
| SigmaDiffractive:maxXX                        |                65.00000 |
| SigmaProcess:alphaSvalue                      |                 0.13500 |
| SpaceShower:rapidityOrder                     |                      on |
| WeakSingleBoson:ffbar2gmZ                     |                      on |
|
*-------  End PYTHIA Flag + Mode + Parm + Word Settings  -------------------
```

Goal: find tracks from hits on sTGC

- Generate events and obtain hits:
  - Event output is fed into GEANT + STAR detector model to record the hits
  - Tracking software smears hit locations

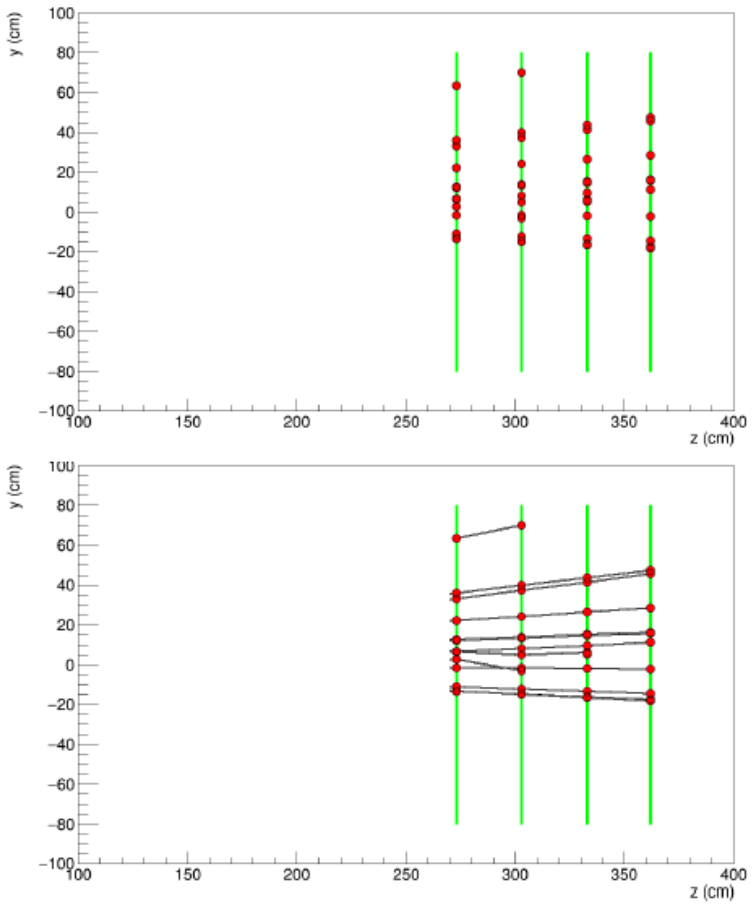


Track Finding

Plot from Daniel

Goal: find tracks from hits on sTGC

- Generate events and obtain hits:
  - Event output is fed into GEANT + STAR detector model to record the hits
  - Tracking software smears hit locations
- Calculate hit pair observables for hits
- Feed a training sample into a boosted decision tree
- Use the trained model to predict whether hit pairs from a testing sample are real or fake
- Evaluate the performance of the model



Track Finding

Plot from Daniel

Hit pair observables

- Hit pair (crit2) observables, calculated for each pair of hits that are:
  - On adjacent tracker layers (1-2, 2-3 or 3-4)

$$\Delta\phi = \phi_A - \phi_B$$

$$\rho = \sqrt{x^2 + y^2}$$

$$\Delta\rho = \rho_A - \rho_B$$

$$\left(\frac{\Delta R}{\Delta z}\right)^2 = \frac{(\Delta x)^2 + (\Delta y)^2 + (\Delta z)^2}{(\Delta z)^2}$$

$$\frac{\rho_A * z_B}{\rho_B * z_A}$$

Hit triplet observables

- Hit triplet (crit3) observables, calculated for each triplet of hits that are:
  - On adjacent tracker layers (1-2-3 or 2-3-4)

Criteria 2DAngle :

$$\Delta x_1 = x_B - x_A$$
$$\Delta y_1 = y_B - y_A$$

$$\Delta x_2 = x_C - x_B$$
$$\Delta y_2 = y_C - y_B$$

$$\cos^2(\theta) = \frac{(\Delta x_1 \Delta x_2 + \Delta y_1 \Delta y_2)^2}{(\Delta x_1^2 + \Delta y_1^2)(\Delta x_2^2 + \Delta y_2^2)}$$

Criteria 3DAngle:

$$\Delta x_1 = x_B - x_A$$
$$\Delta y_1 = y_B - y_A$$
$$\Delta z_1 = z_B - z_A$$

$$\Delta x_2 = x_C - x_B$$
$$\Delta y_2 = y_C - y_B$$
$$\Delta z_2 = z_c - z_B$$

$$\cos^2(\theta) = \frac{(\Delta x_1 \Delta x_2 + \Delta y_1 \Delta y_2 + \Delta z_1 \Delta z_2)^2}{(\Delta x_1^2 + \Delta y_1^2 + \Delta z_1^2)(\Delta x_2^2 + \Delta y_2^2 + \Delta z_2^2)}$$

Criteria ChangeRZRatio :

$$\Delta RZ = \left(\frac{\Delta R}{\Delta z}\right)_{BA}^2 - \left(\frac{\Delta R}{\Delta z}\right)_{BC}^2$$

Training for hit pairs

- For each pair of hits, BDT takes in:
  - Crit2 values

		Crit2_RZRatio	Crit2_DeltaRho	Crit2_DeltaPhi	Crit2_StraightTrackRatio
entry	**subentry**				
0	**0**	1.021419	-4.738697	0.801099	0.691984
	1	1.022302	-4.839012	0.601002	0.688314
	2	1.304533	-17.746450	19.210548	0.409136
	4	1.594602	-26.755651	22.318817	0.318865
	6	1.001245	0.990740	1.653889	0.973371
...	**...**	...	...	...	...
49499	**4**	1.007454	2.785313	0.434769	0.998189
	5	1.299637	-15.502758	12.340158	0.688496
	6	1.299396	-15.427925	12.452241	0.689373
	7	1.006945	2.691032	0.081525	0.996621
	8	1.006917	2.640335	0.675205	0.995380

  - 0 (fake, a mixture of hits from different tracks) or 1 (real, both hits from the same track)

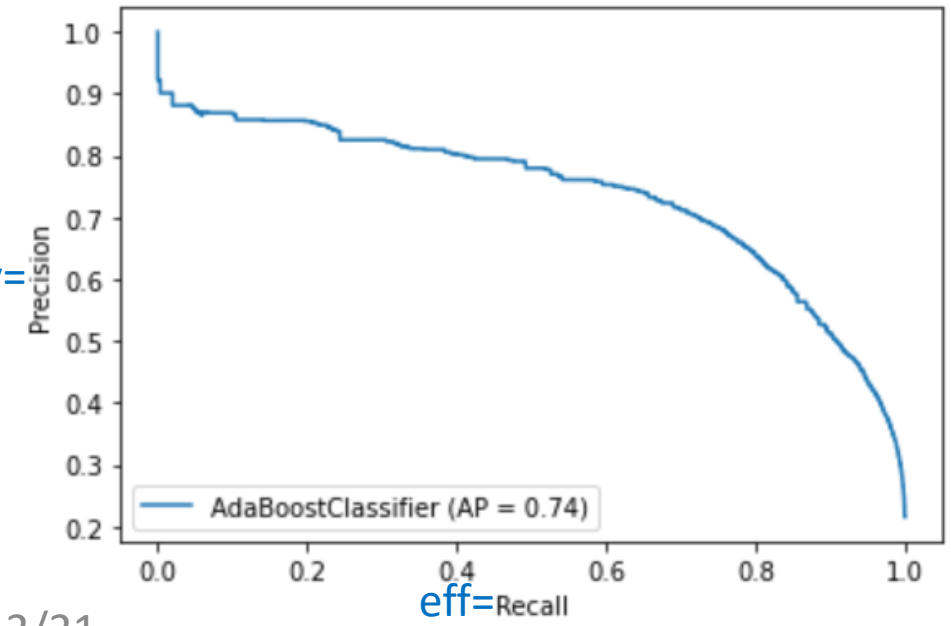
Testing for hit pairs

	precision	recall	f1-score	support
fake	0.91	0.94	0.92	231134
real	purity= 0.74	eff= 0.66	0.69	61367
accuracy			0.88	292501
macro avg	0.82	0.80	0.81	292501
weighted avg	0.88	0.88	0.88	292501

Area under ROC curve: 0.9199

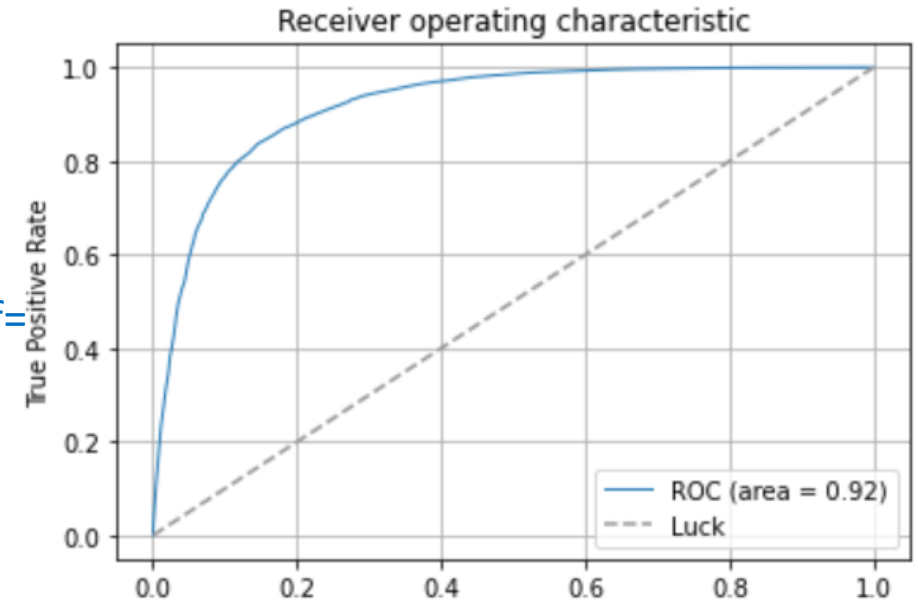
background rej = 1 / fpr
fpr = 1 – fake recall
→ fake recall = 1 – 1 / rej
→ rej = 1 / (1 – fake recall)

purity=



eff=

eff=

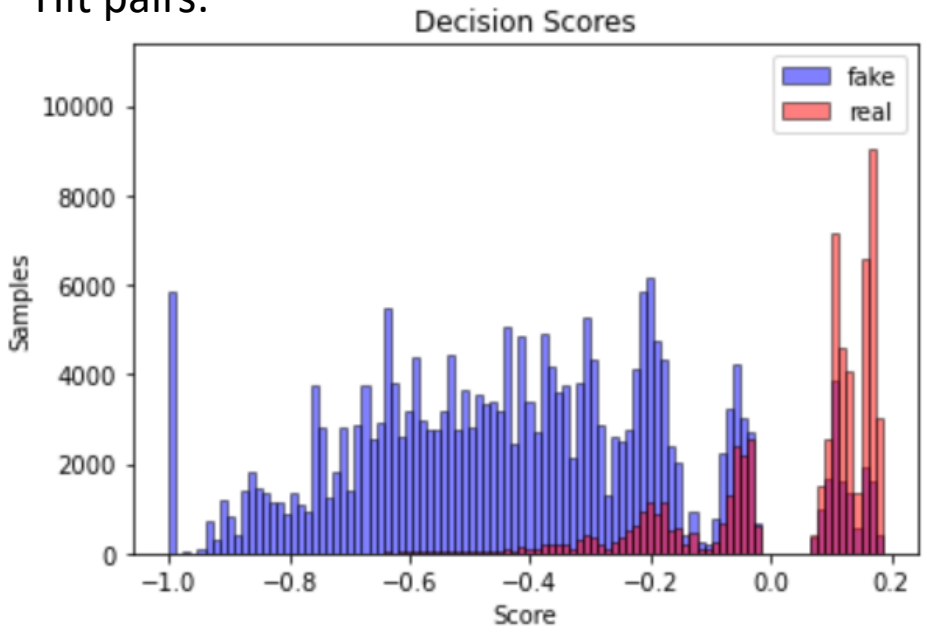


1/background rej =

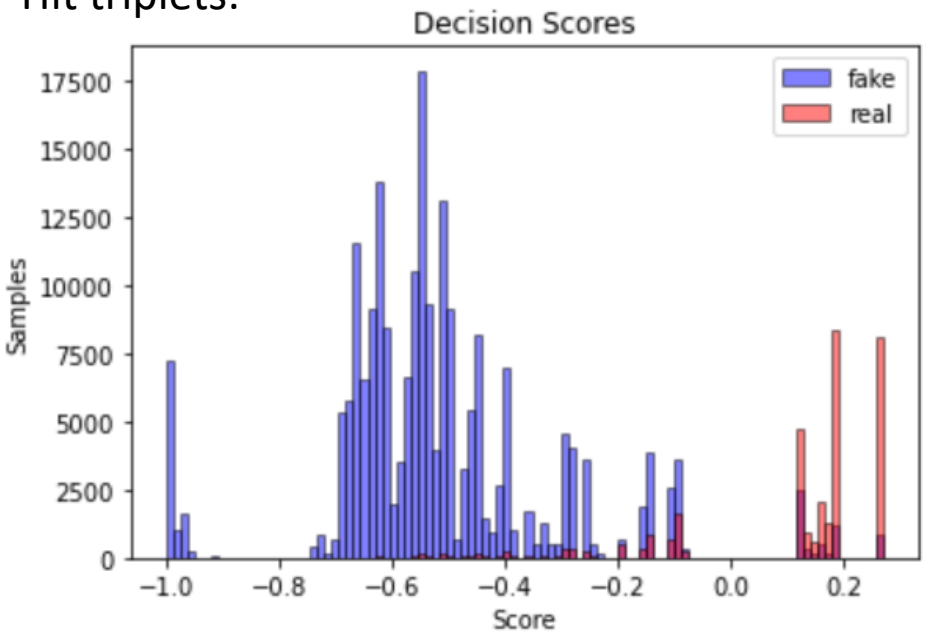
Testing

- For each pair, a decision score is calculated based on its observable values. If it passes a certain threshold (determined by the BDT), then it is identified as real.

Hit pairs:



Hit triplets:



Tracking efficiency

- Count reconstructed tracks
  - Look at a real track, and see if it is missed / reconstructed
  - Trained BDT predicts if a pair is real / fake, but doesn't predict the tracks
  - Criterion: if at least one pair of hits of a given track is identified as real, then we consider this track as reconstructed

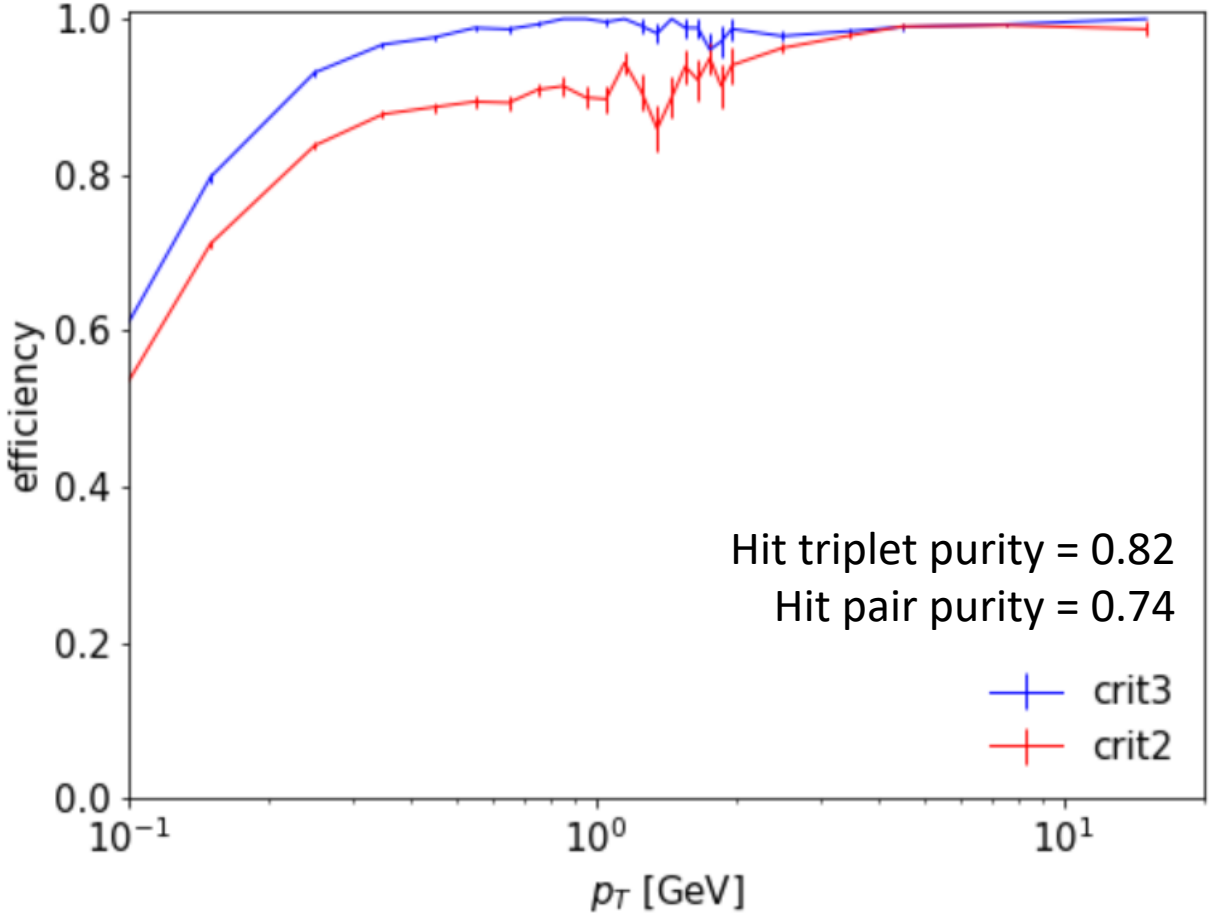
probability (identified by BDT) that the hit is real – probability fake

entry	Crit2_DeltaPhi_trackIds	proba_diff	y2_predicted	
1848.0	25.0	0.076483	1.0	
1848.0	27.0	0.080750	1.0	
1848.0	114.0	-0.022738	0.0	
1848.0	114.0	-0.024307	0.0	BDT identifies as fake
1848.0	114.0	0.046811	1.0	BDT identifies as real

Tracking efficiency

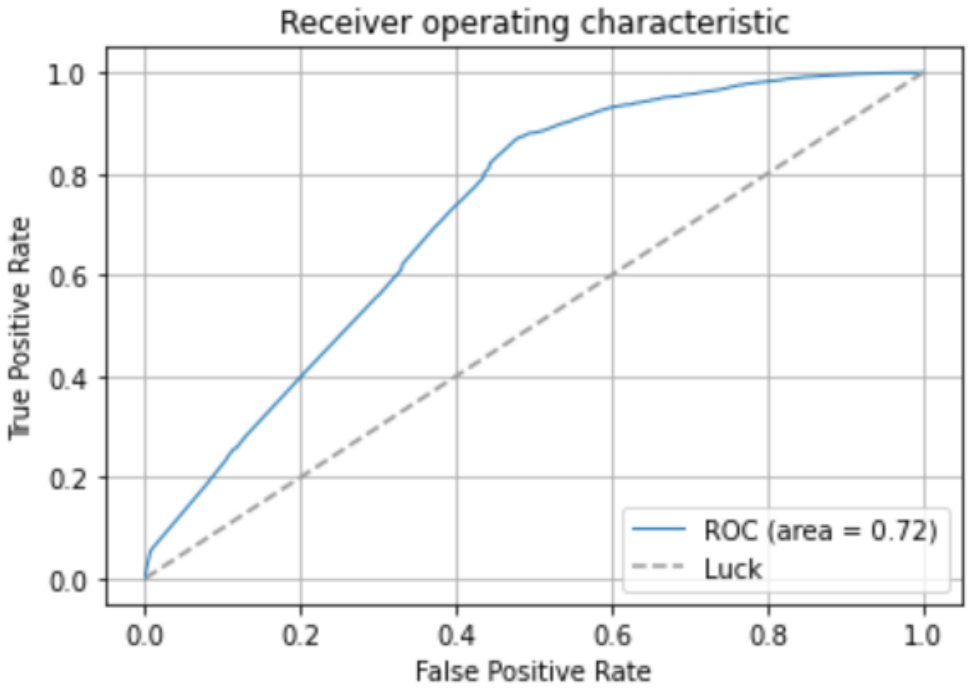
- Plotted as a function of track pT
- Different working points can be selected to achieve higher efficiency (with lower purity) etc.

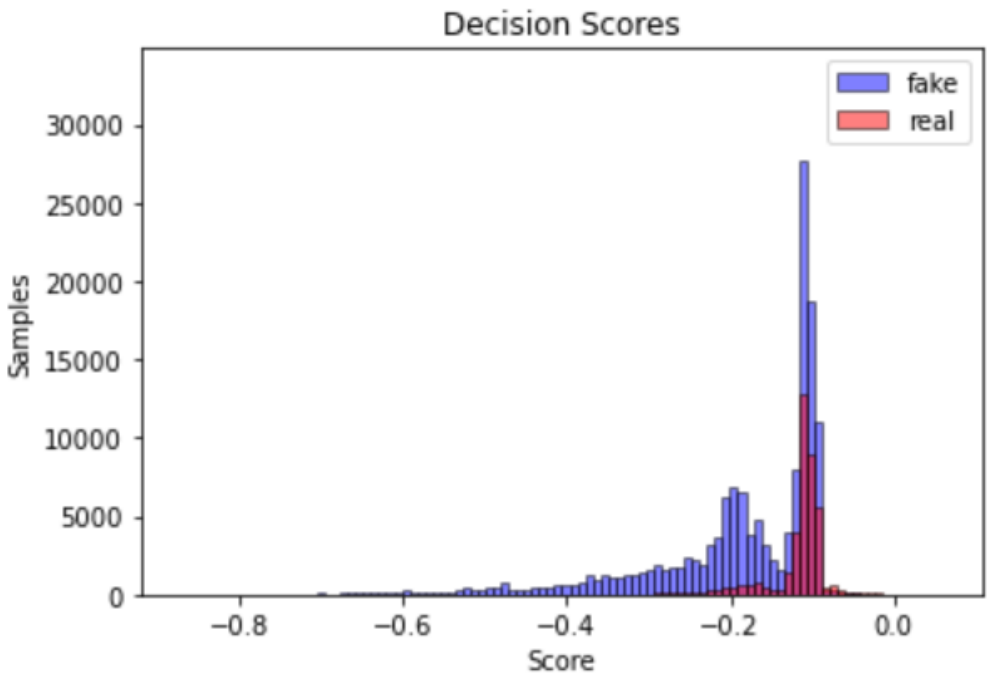
- In the end the tracks will be selected sequentially:
  - Using 2-hit criteria (Crit2), then
  - Using 3-hit criteria (Crit3)



Next steps

- Using Crit2 and Crit3 limits the information – maybe more powerful discrimination is possible
- Try training directly on “raw” info – no interpretation
- Directly train on hit pair locations: input (x_1,y_1 ,z_1,x_2,y_2 ,z_2)
- Results with BDT training are not so good (lack of separation between real and fake) – expected since BDT ill formed for this task





Next Steps

- Neural networks are more suited for learning abstract relationships
- Neural network classifier may be able to choose real pairs from raw data (x_1,y_1 ,z_1,x_2,y_2 ,z_2) tuples.
- Goal: train a NN for 2-hit, 3-hit and 4-hit tuples to improve track seed quality

Summary

- Help with first study of tracking in Pythia8 events with Daniel's FWD tracker
- Develop procedure for training Boosted-Decision Trees to optimize track seed finding
- BDT performance looks promising

- Compare BDT cut efficiency and purity with Daniel's default cuts
- Major issue right now:
  - Read trained model into ROOT for future analysis: having trouble with ROOT5 interface
  - ROOT6 is not available for STAR software (maybe in future)
  - HFT and other groups have used BDT before, need to find out how they read model parameters and implemented in C++
- More sophisticated techniques (neural networks) may provide more benefit

- Thank you!

Backup

- Event trigger

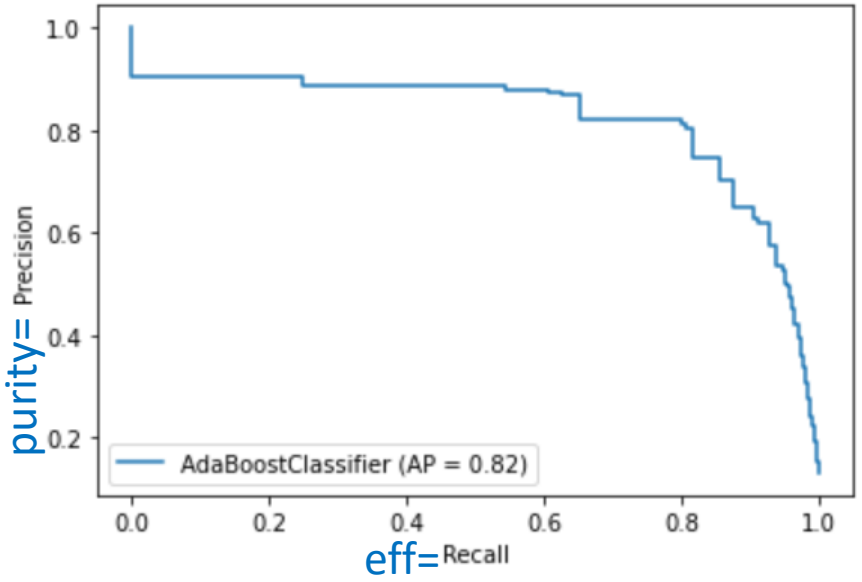
```
_filter = new StarParticleFilter();  _primary->AddFilter( _filter );
//                        pdg  ptmn ptmx  etamn etamx parent pdgid
_filter -> AddTrigger( +11, 1.4, -1.0, 2.25, 4.25,  23 );
_filter -> AddTrigger( -11, 1.4, -1.0, 2.25, 4.25,  23 );
```

- For crit3 (variables: 'Crit3_3DAngle', 'Crit3_ChangeRZRatio', 'Crit3_2DAngle')

1 – 1 / rej =

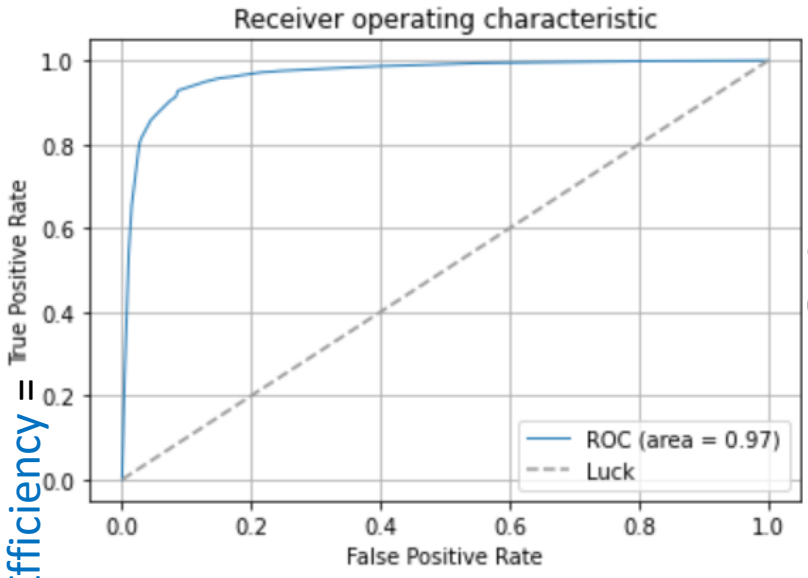
purity= eff=

	precision	recall	f1-score	support
fake	0.97	0.97	0.97	215656
real	0.82	0.80	0.81	32746
accuracy			0.95	248402
macro avg	0.89	0.89	0.89	248402
weighted avg	0.95	0.95	0.95	248402



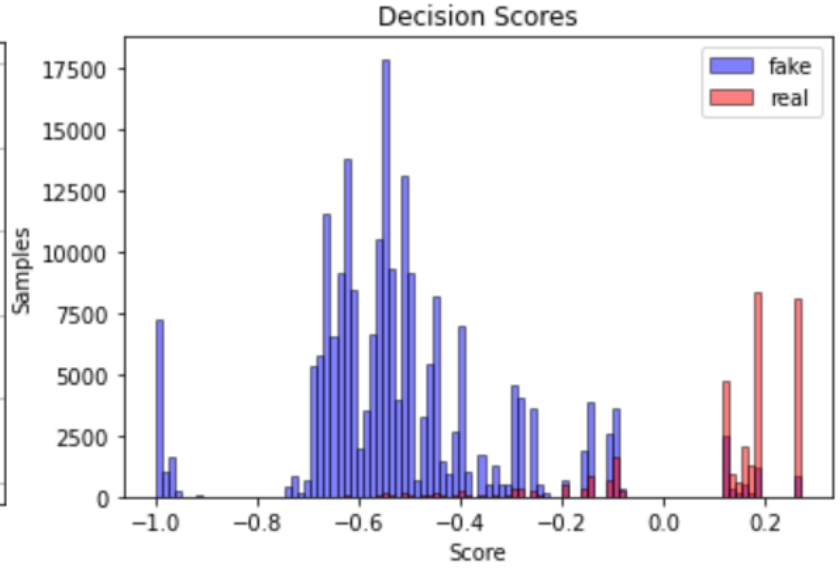
purity=

eff=

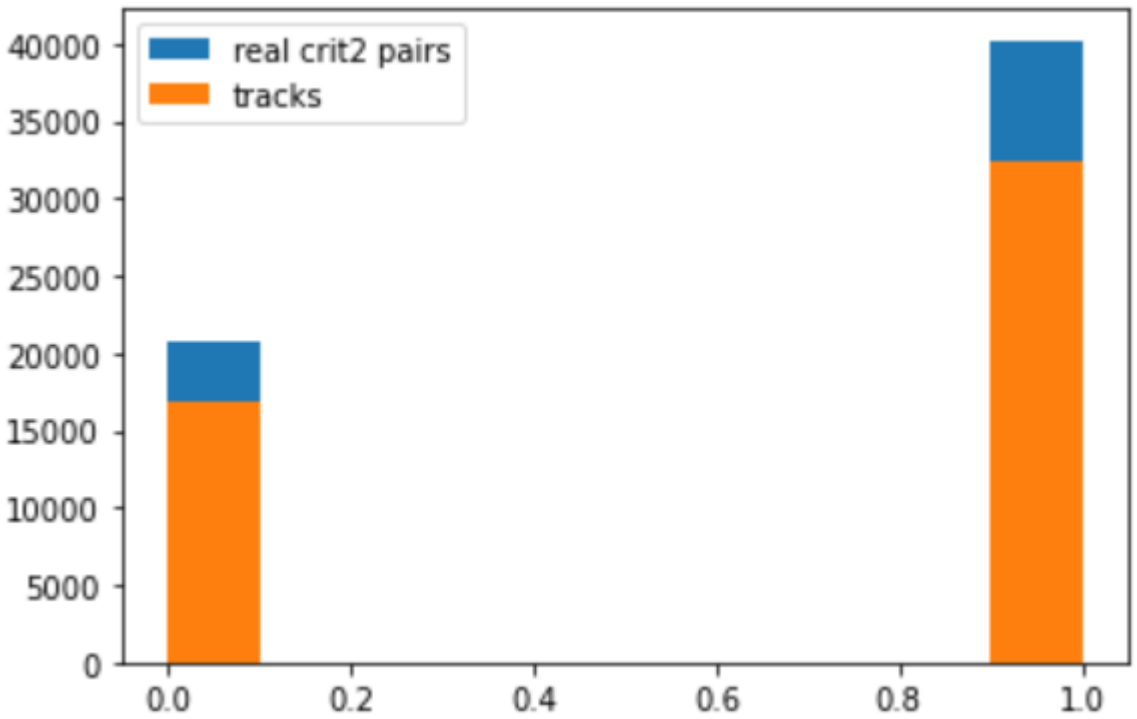


Efficiency =

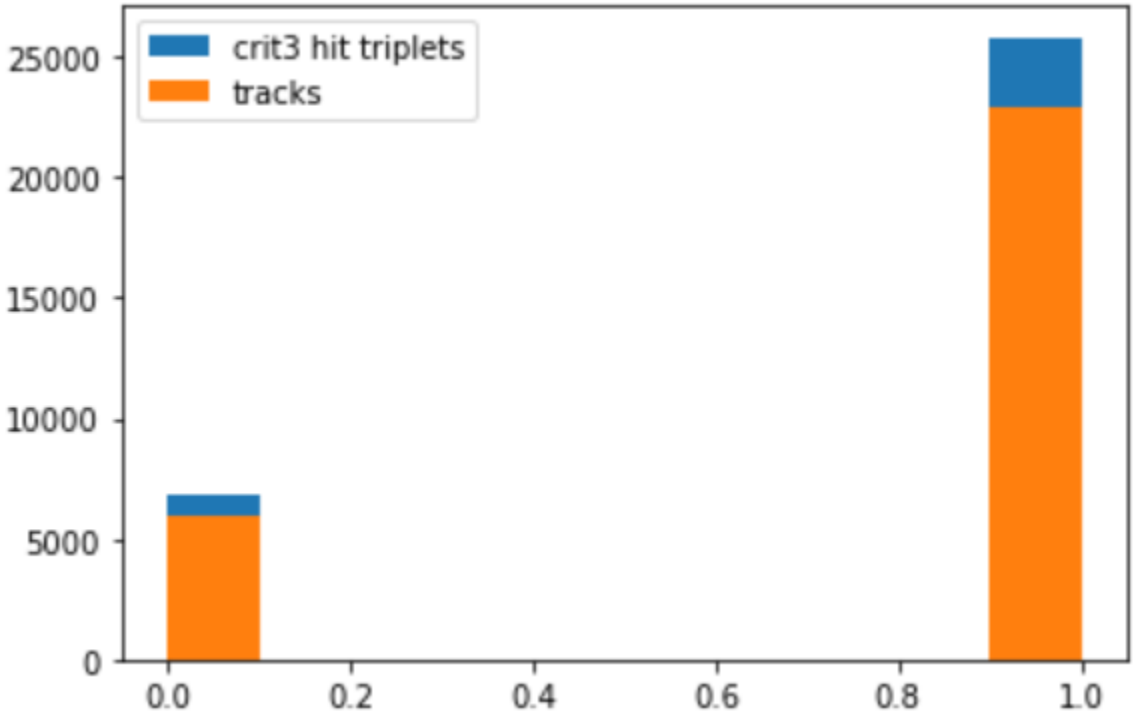
= 1/background rej

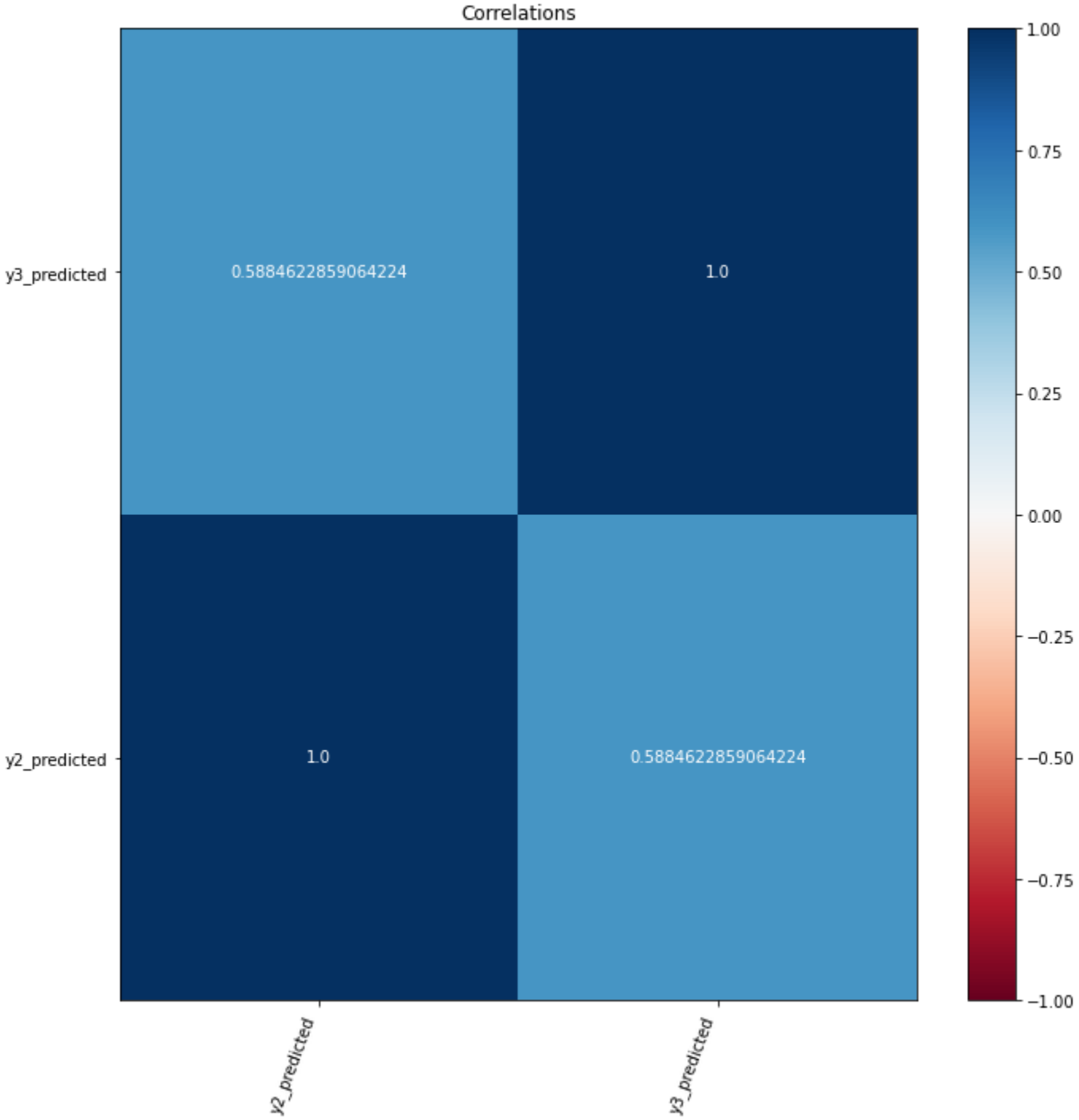


- Tracking efficiency vs pT
  - Over all pT intervals, tracking efficiency is similar to “hit pair efficiency”



missed reconstructed




Correlations
y3_predicted
y2_predicted
0.5884622859064224
1.0
1.0
0.5884622859064224
y2_predicted
y3_predicted
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00