Graph Neural Network for τ identification

Kirill Bukin

Supervisors:

Dirk Kruecker

Leonid Didukh

Isabell Melzer-Pellmann

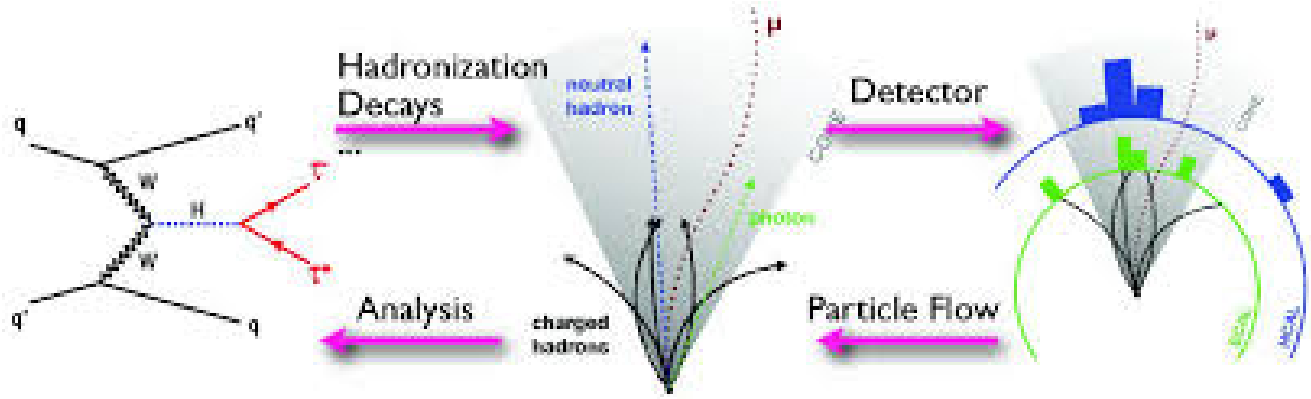
August 14, 2019

Introduction

		τ Decay Mode	Branching Fraction (%)
Leptonic		$\tau^{\pm} \to e^{\pm} + \overline{\nu}_e + \nu_\tau$	17.84 ± 0.04
		$\tau^{\pm} \to \mu^{\pm} + \overline{\nu}_\mu + \nu_\tau$	17.41 ± 0.04
Hadronic	One-prong	$\tau^{\pm} \to \pi^{\pm} + (\geq 0\ \pi^0) + \nu_\tau$	49.46 ± 0.10
		$\tau^{\pm} \to \pi^{\pm} + \nu_\tau$	10.83 ± 0.06
		$\tau^{\pm} \to \rho^{\pm}(\to \pi^{\pm} + \pi^0) + \nu_\tau$	25.52 ± 0.09
		$\tau^{\pm} \to a_1(\to \pi^{\pm} + 2\pi^0) + \nu_\tau$	9.30 ± 0.11
		$\tau^{\pm} \to \pi^{\pm} + 3\pi^0 + \nu_\tau$	1.05 ± 0.07
		$\tau^{\pm} \to h^{\pm} + 4\pi^0 + \nu_\tau$	0.11 ± 0.04
Hadronic	Three-prong	$\tau^{\pm} \to \pi^{\pm} + \pi^{\mp} + \pi^{\pm} + (\geq 0\pi^0) + \nu_\tau$	14.57 ± 0.07
		$\tau^{\pm} \to \pi^{\pm} + \pi^{\mp} + \pi^{\pm} + \nu_\tau$	8.99 ± 0.06
		$\tau^{\pm} \to \pi^{\pm} + \pi^{\mp} + \pi^{\pm} + \pi^0 + \nu_\tau$	2.70 ± 0.08

- Goal of our work is to discriminate hadronically decaying tau from quark and gluon jets using graph learning techniques
- As a baseline, DeepPF network is used

PF algorithm



The Particle Flow event reconstruction aims at reconstructing all stable particles arising from collisions, using combined response from all the CMS sub-detectors. This list of individual particles is then used to build jets, to determine the missing transverse energy, to reconstruct and identify taus from their decay products, etc.
https://cds.cern.ch/record/2029414/

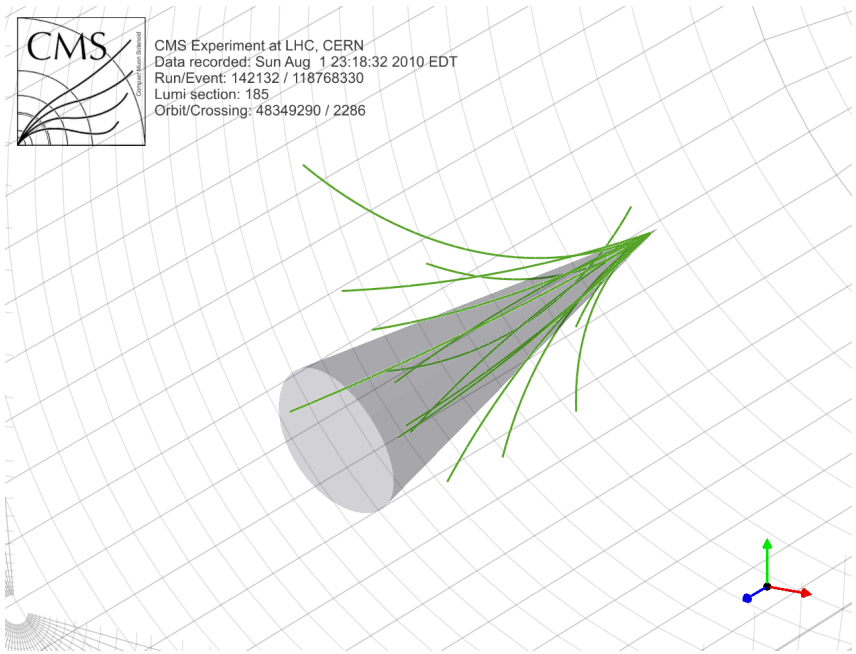
MC samples

Samples from 2016 MC are used. True tau are selected from DY MC samples, fake from W+jet MC samples. In both cases, only decays into one charged hadron (decay mode 0), one charged hadron and a neutral pion (decay mode 1), or three charged hadrons (decay mode 10) are selected

Reweighting is applied to flatten in tau p_T spectrum

DPF network (Author: Owen Colegrove)

- tau are reconstructed from particle flow particles
- Variables from particles contained in cone centered on the tau are formatted into a 2-D table and fed into a Deep Neural Network
- Each event are represented as matrix 60*47(number of particles - 60; number of features - 47)
- If number of particles in cone is less then 60, the last rows of table are filled with zeroes



Particles	feature1	...	featureN
p1	f1,p1		fN,p1
...			
pm	f1,pm		fN,pm

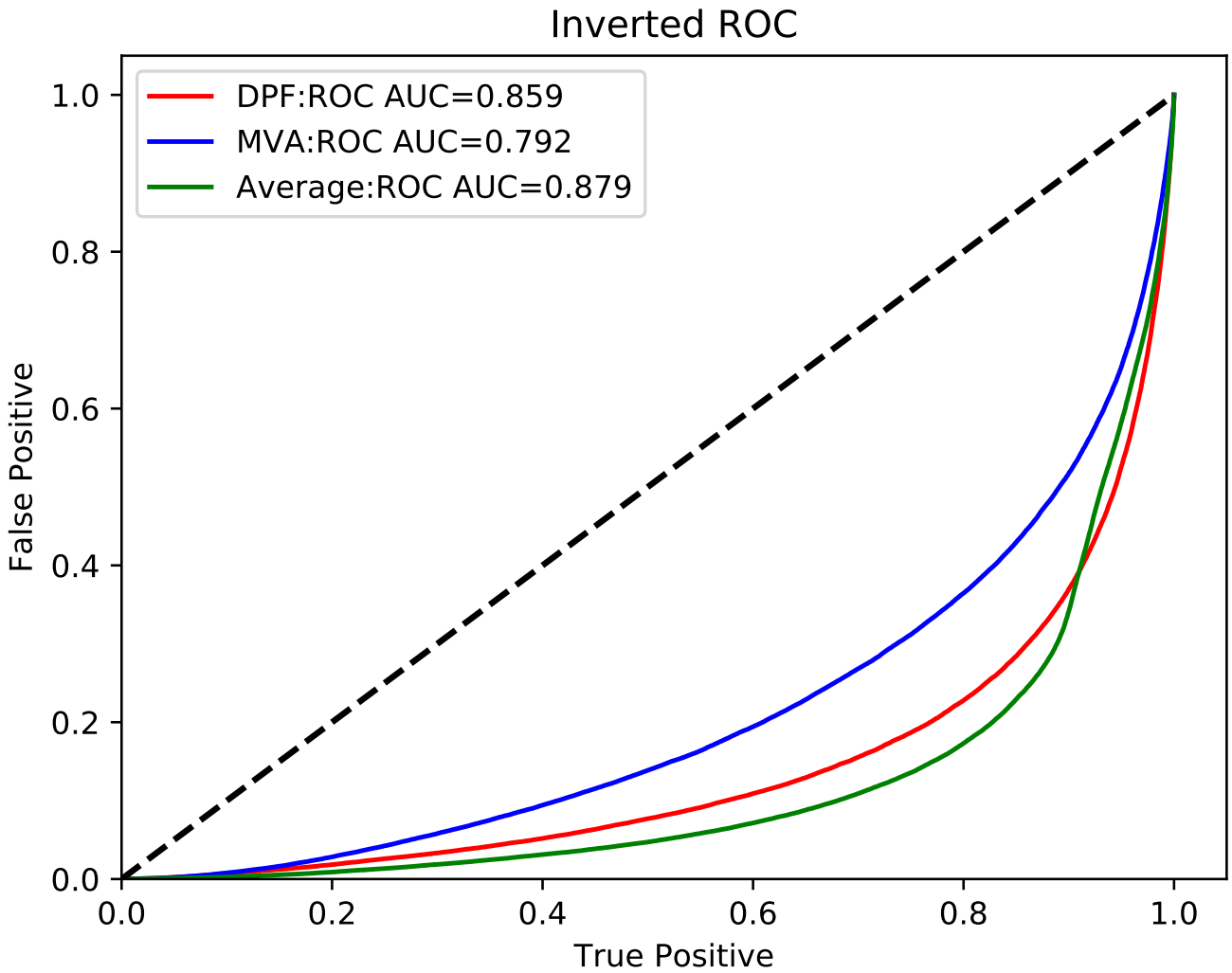
Loss function

As a loss function, binary cross entropy is used:

$$BCE = \frac{1}{N}\sum_{i=0}^{N} y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i)$$

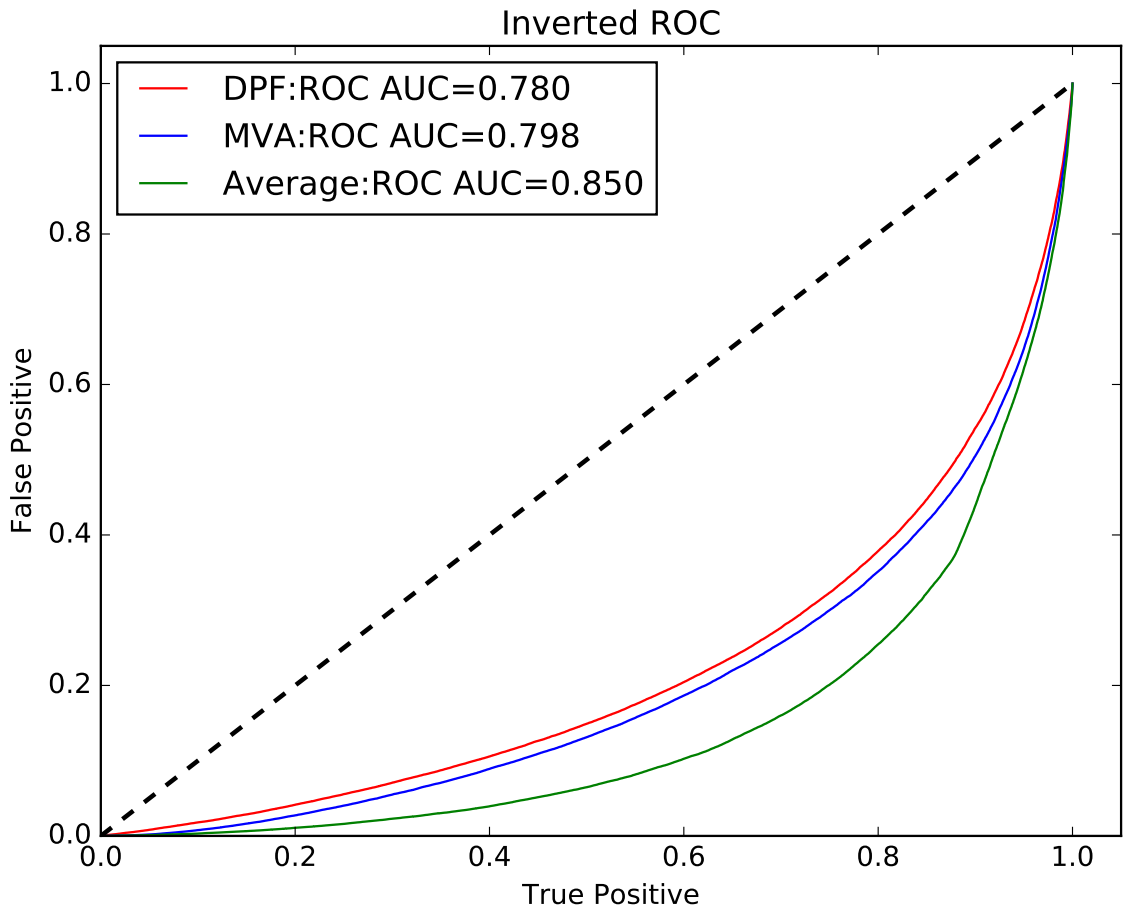
Loss is optimised by Adam optimiser

DPF performance



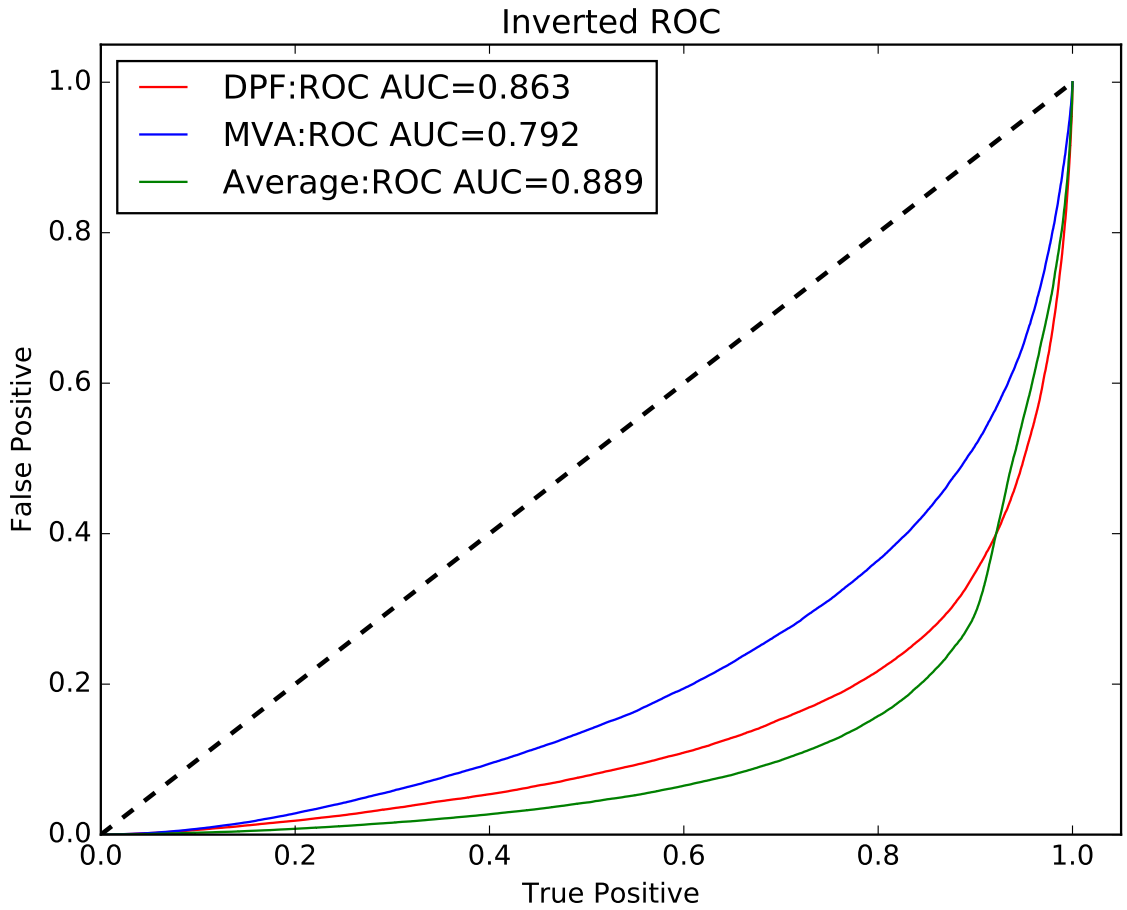
Modifications of DPF

DPF with transposed input matrix (convolution over the particles, for each feature) shows worse performance than DPF



Modifications of DPF

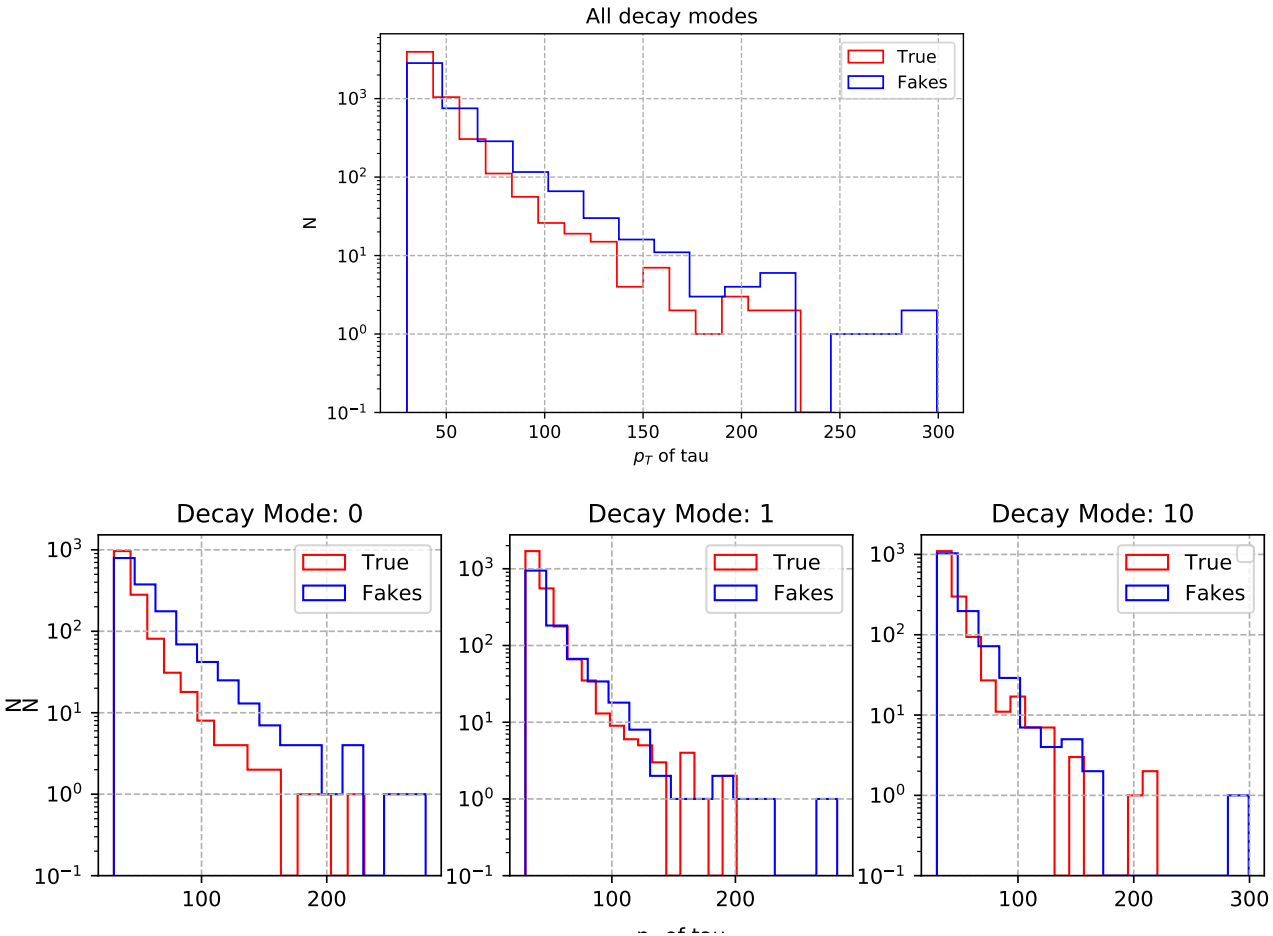
DPF with reordered features shows performance close to the one of DPF



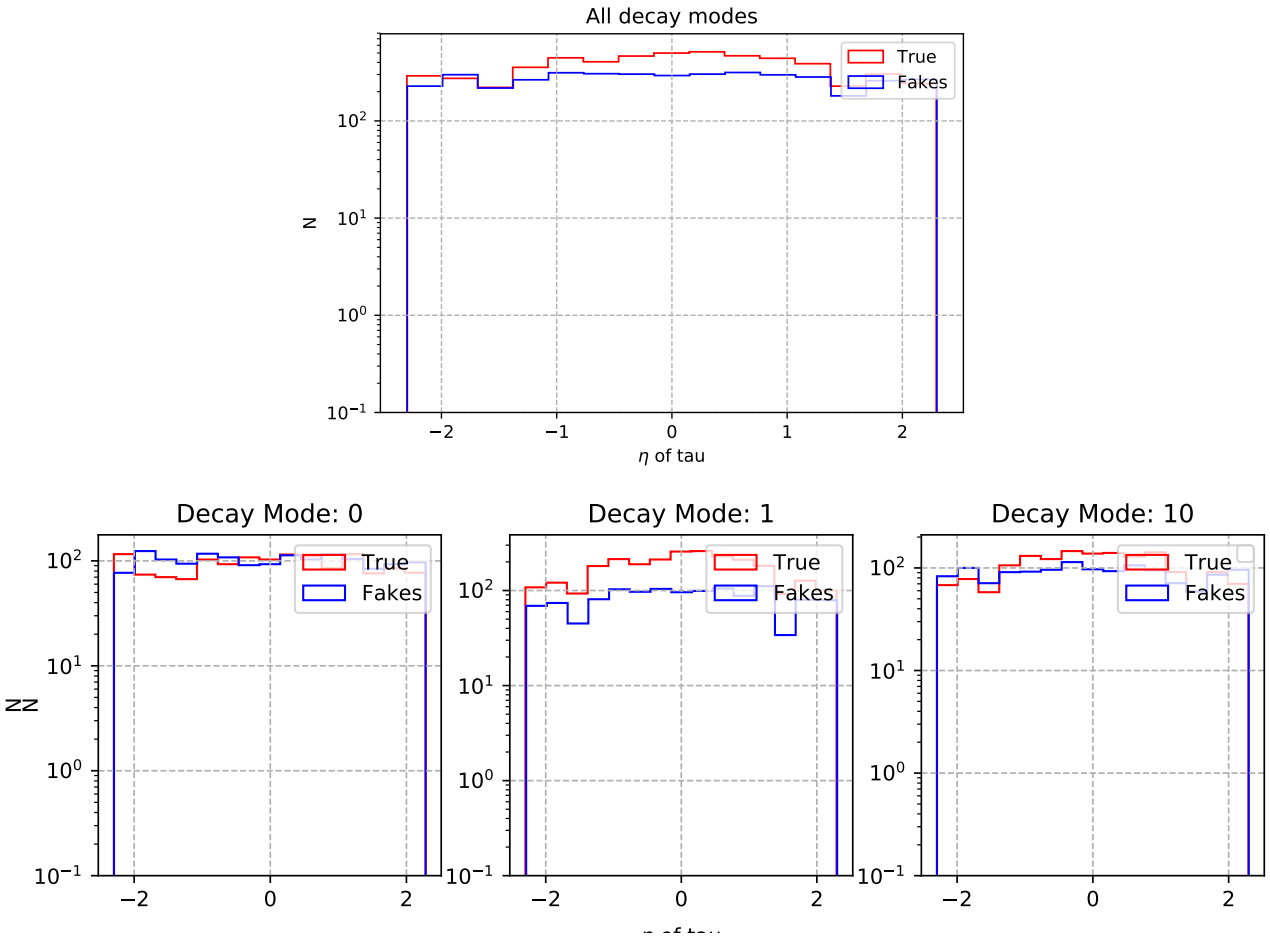
Graph Convolutional Network

- The same data and the same features as in DPF network are used
- Variables from particles contained in cone centered on the tau are represented as a point cloud (each particle represented as a one point)
- Graph is constructed from a particle cloud as k-nearest neighbour graph
- Graph fed into Graph Convolutional Neural Network, which consists of graph convolutional layers and FC layers
  - Features describing the entire event (p_T of reconstructed tau, number of PF candidates) are concatenated to feature vector of every particle in the event
  - Alternative: fed these features directly to FC layers with outputs of convolutional layers

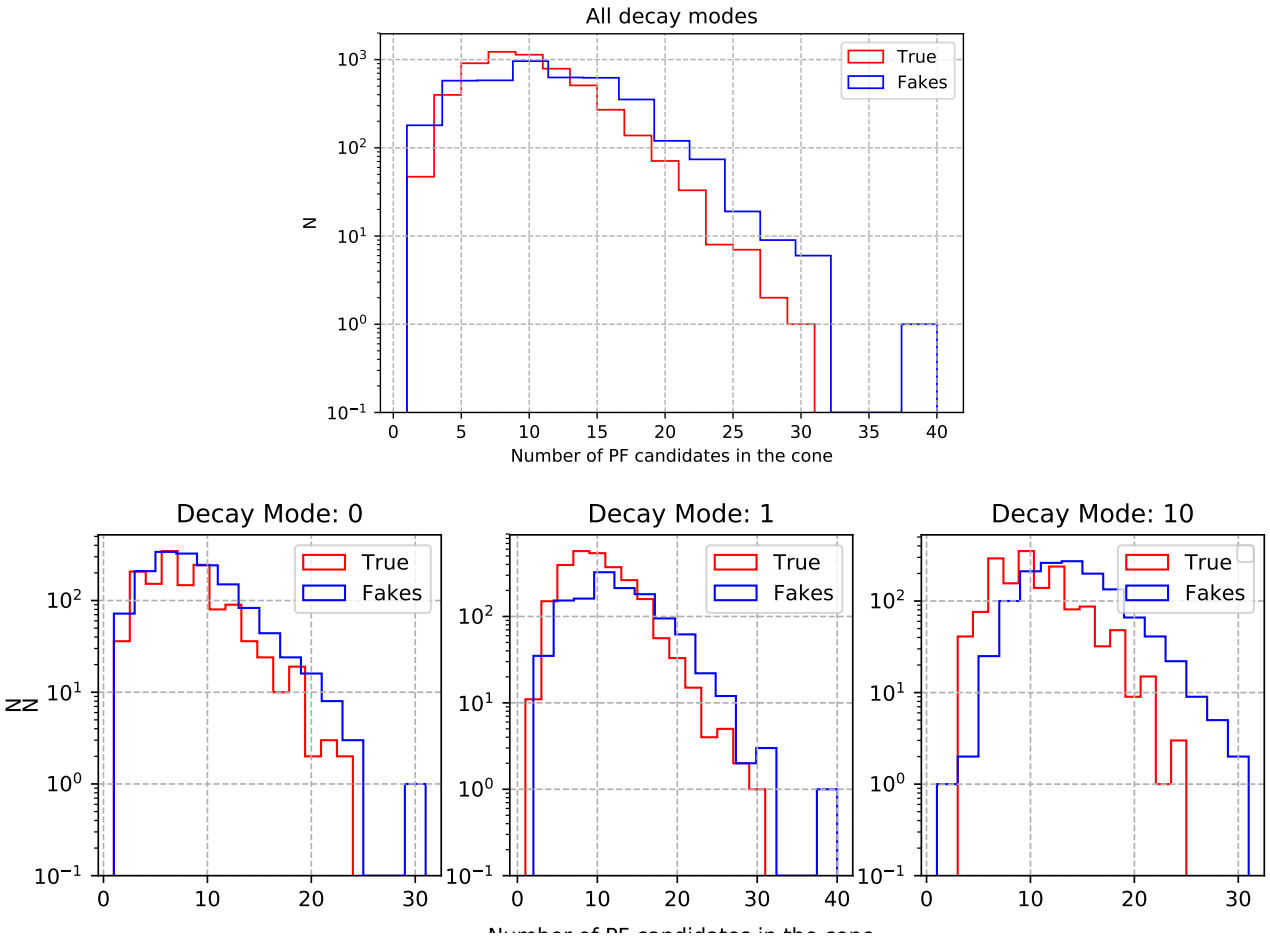
Feature: p_T of tau



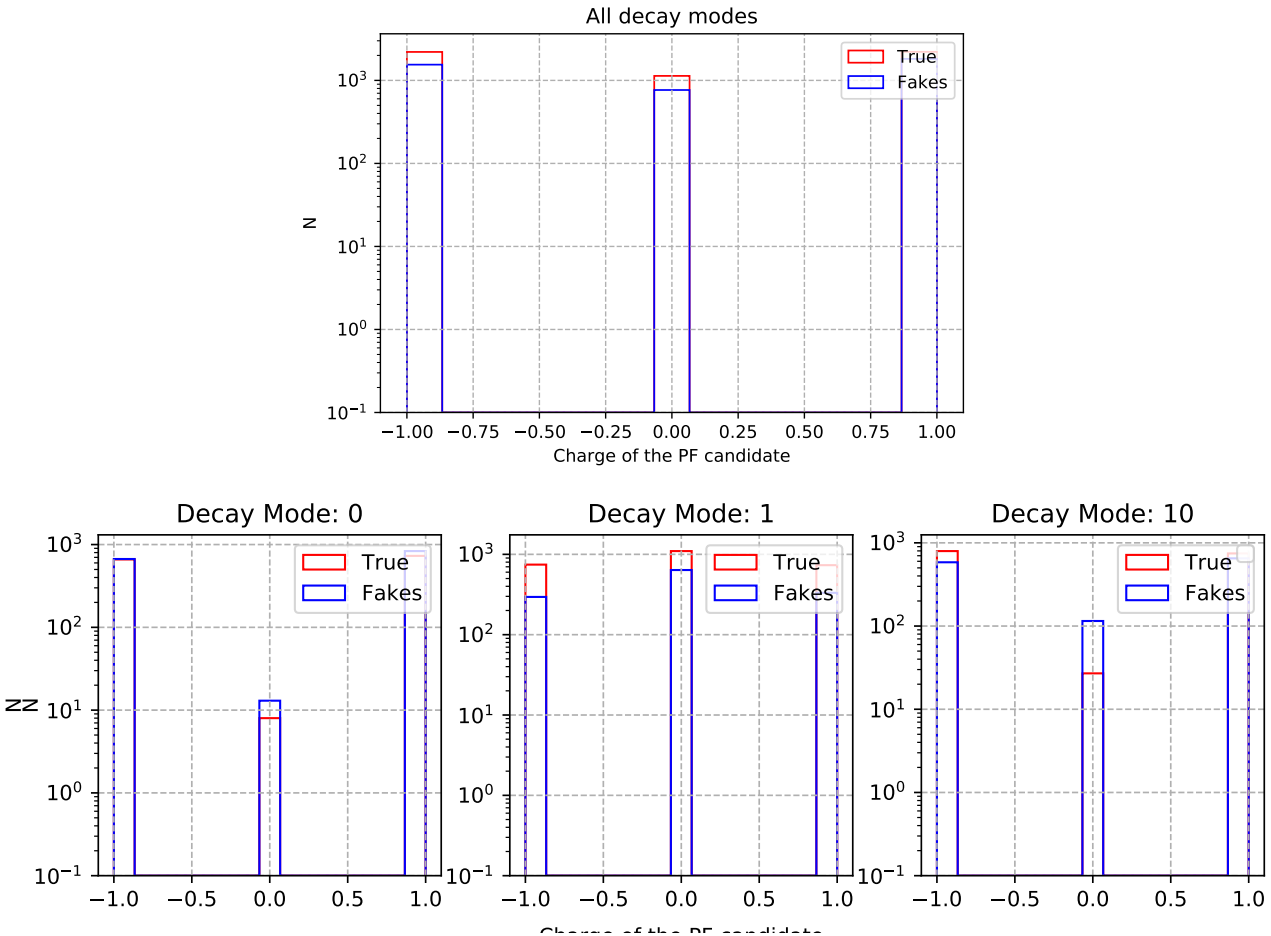
Feature: η of tau



Feature: Number of PF candidates in the cone

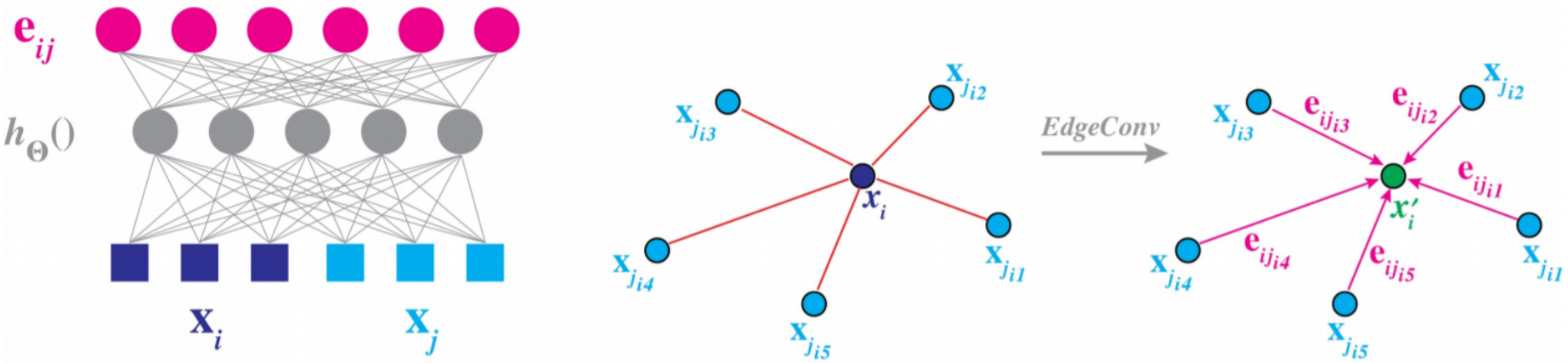


Feature: Charge of the PF candidate



EdgeConv Layer

https://arxiv.org/abs/1801.07829



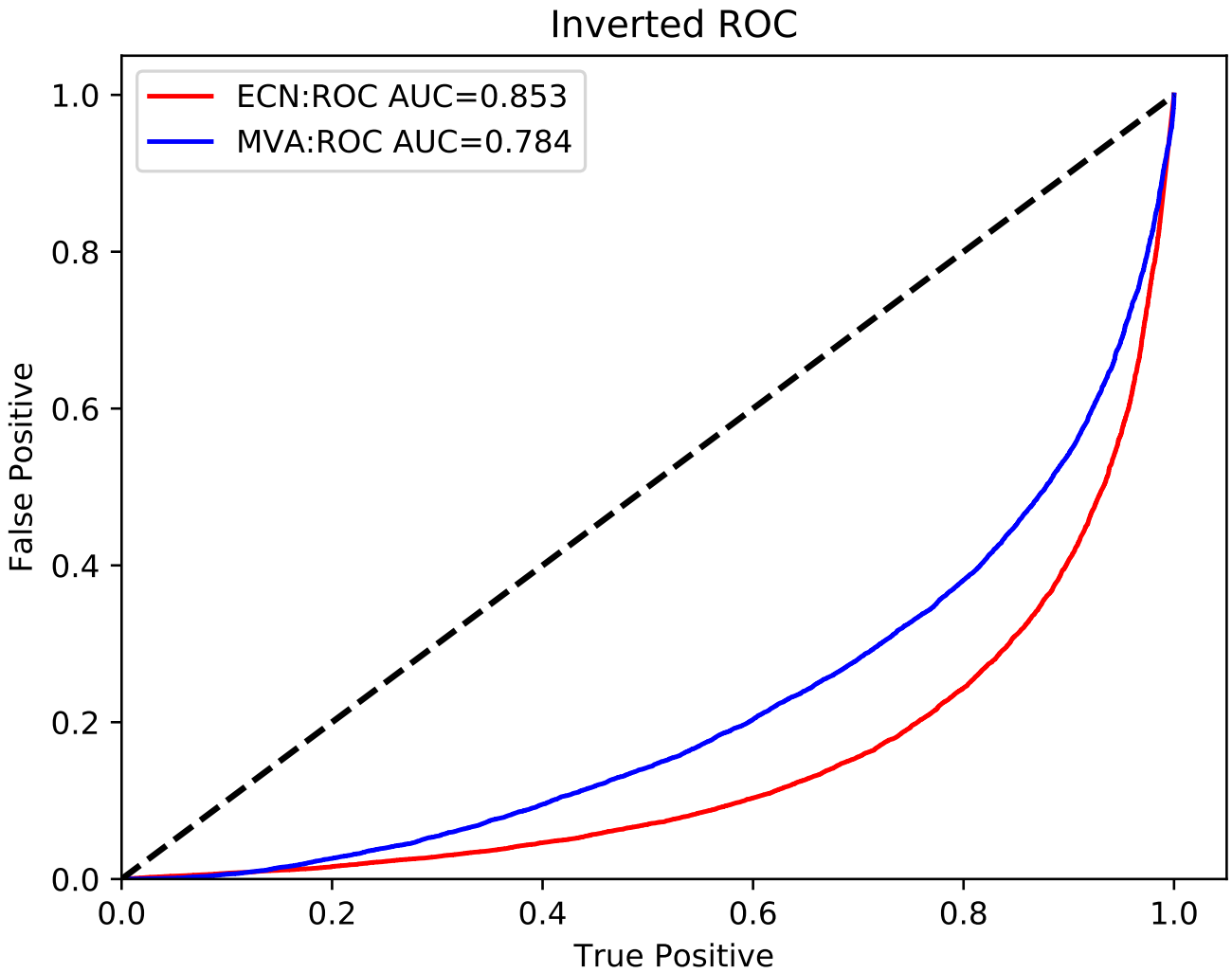
Edge features are defined as:

$$e_{ij} = h_\Theta(x_i, x_j)$$

EdgeConv operation is defined by applying a symmetric aggregation operation $\square$ (e.g., $\sum$ or max):

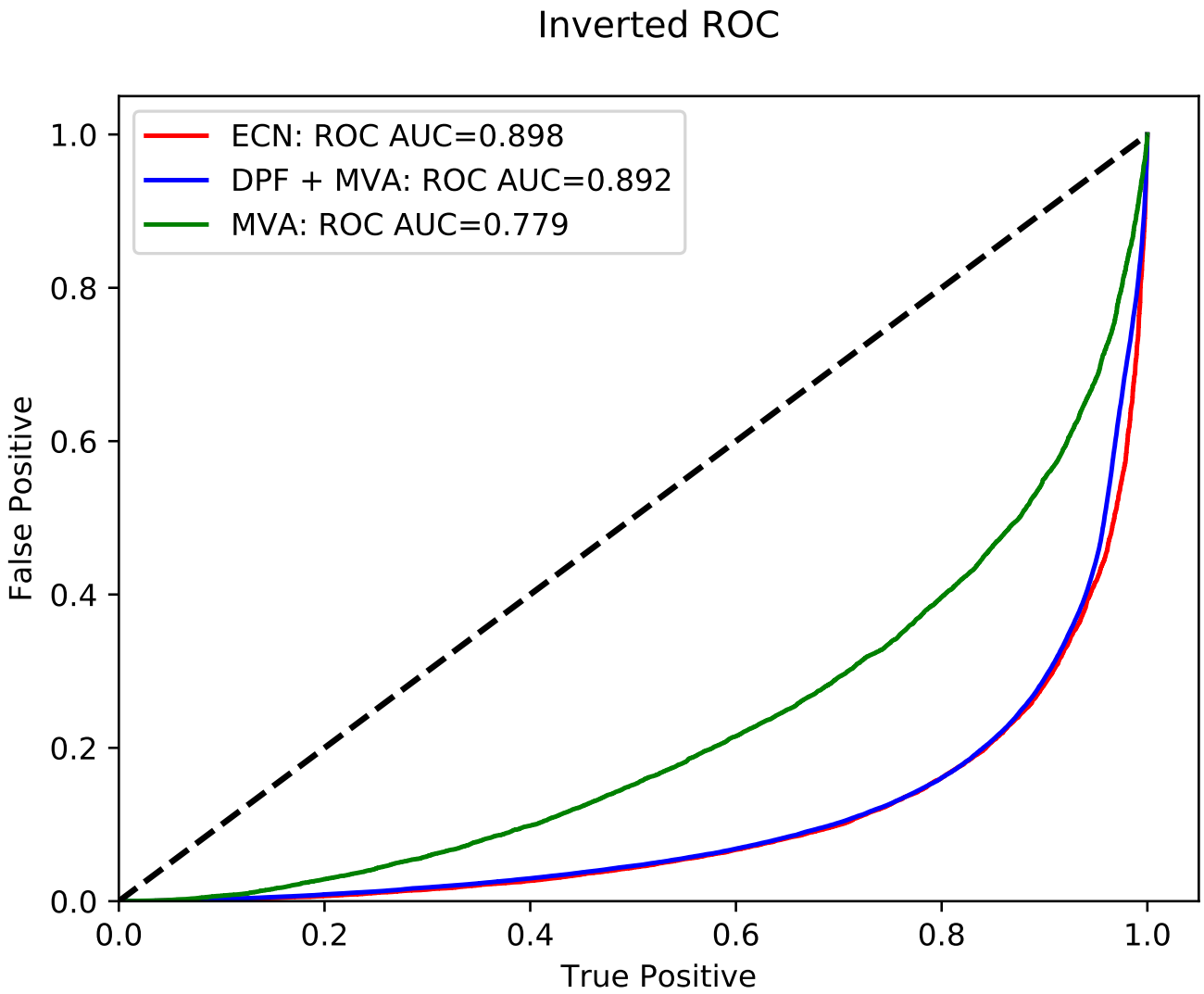
$$x'_i = \square h_\Theta(x_i, x_j)$$

ECN performance (1 Convolutional layer)

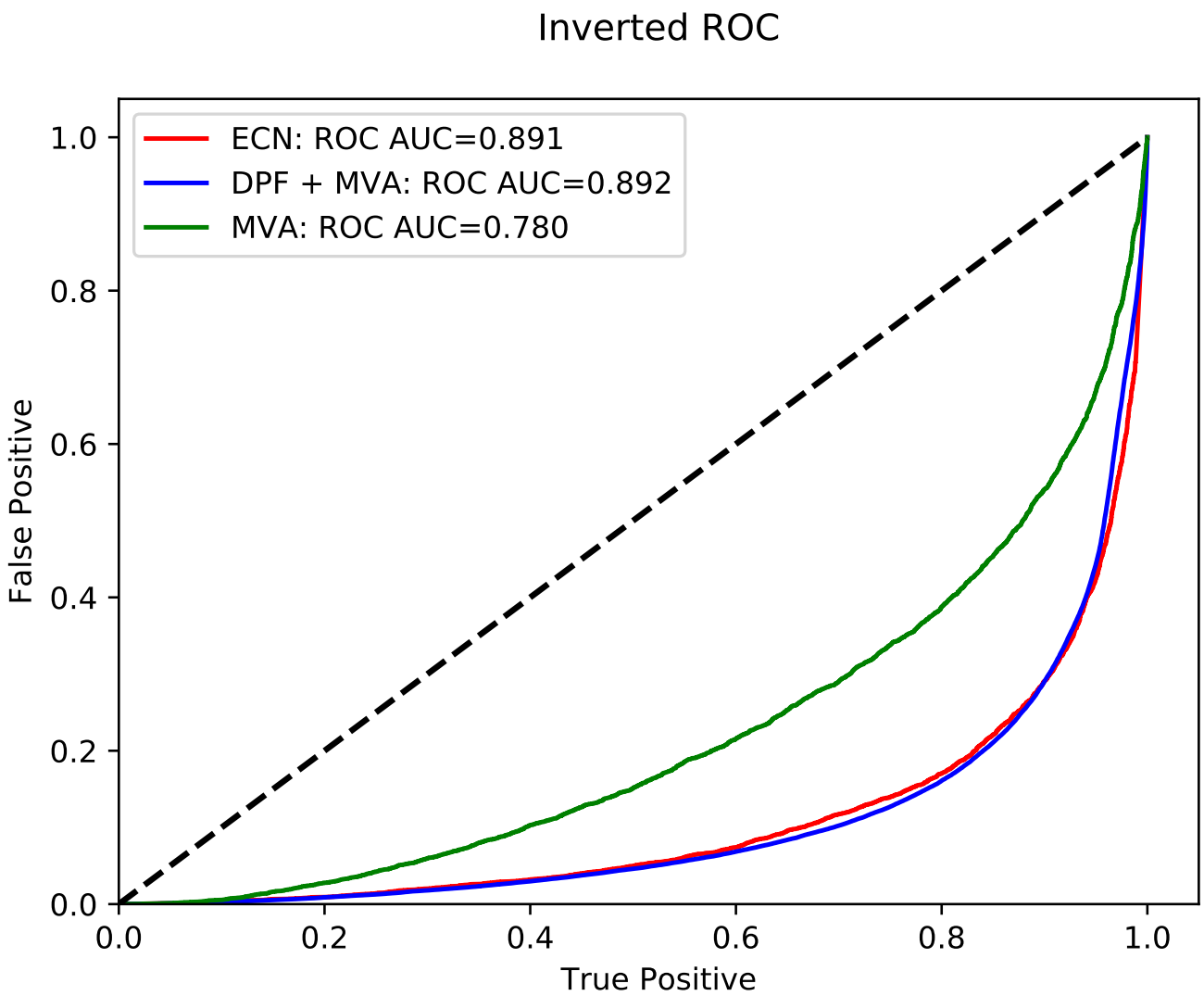


ECN performance (3 Convolutional layers)

Feature normalization fixed, added one hot encoding for categorical features

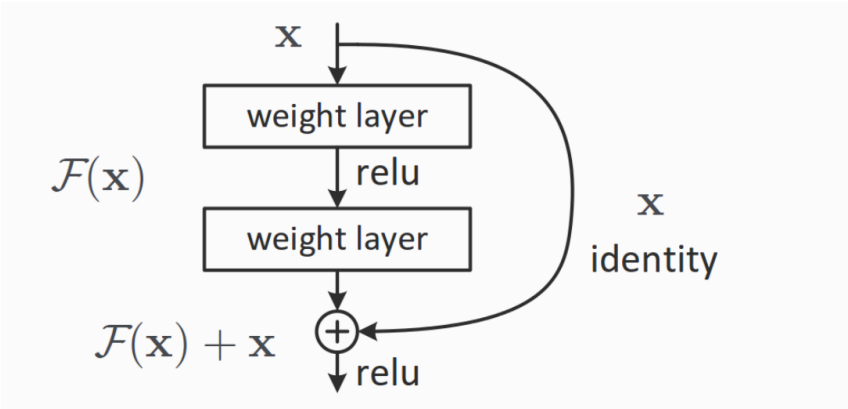


ECN performance (Separated features)



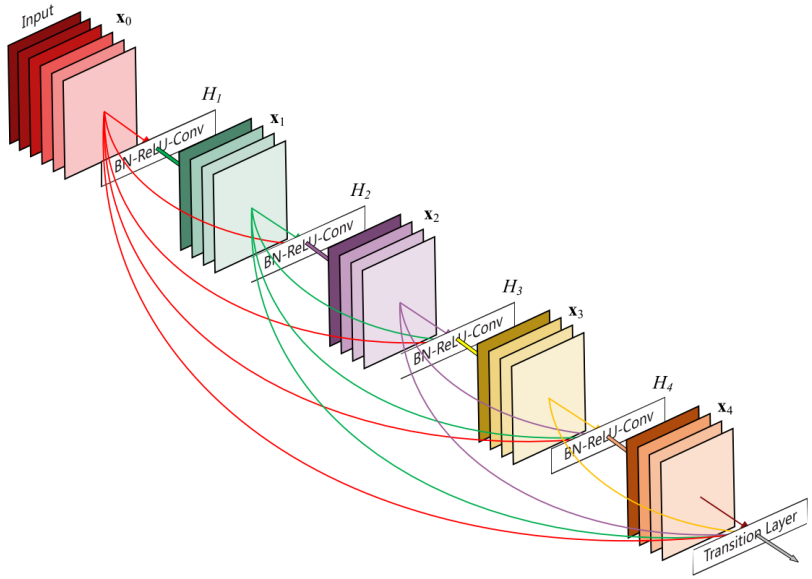
Residual and Dense Neural Networks

Residual Network block



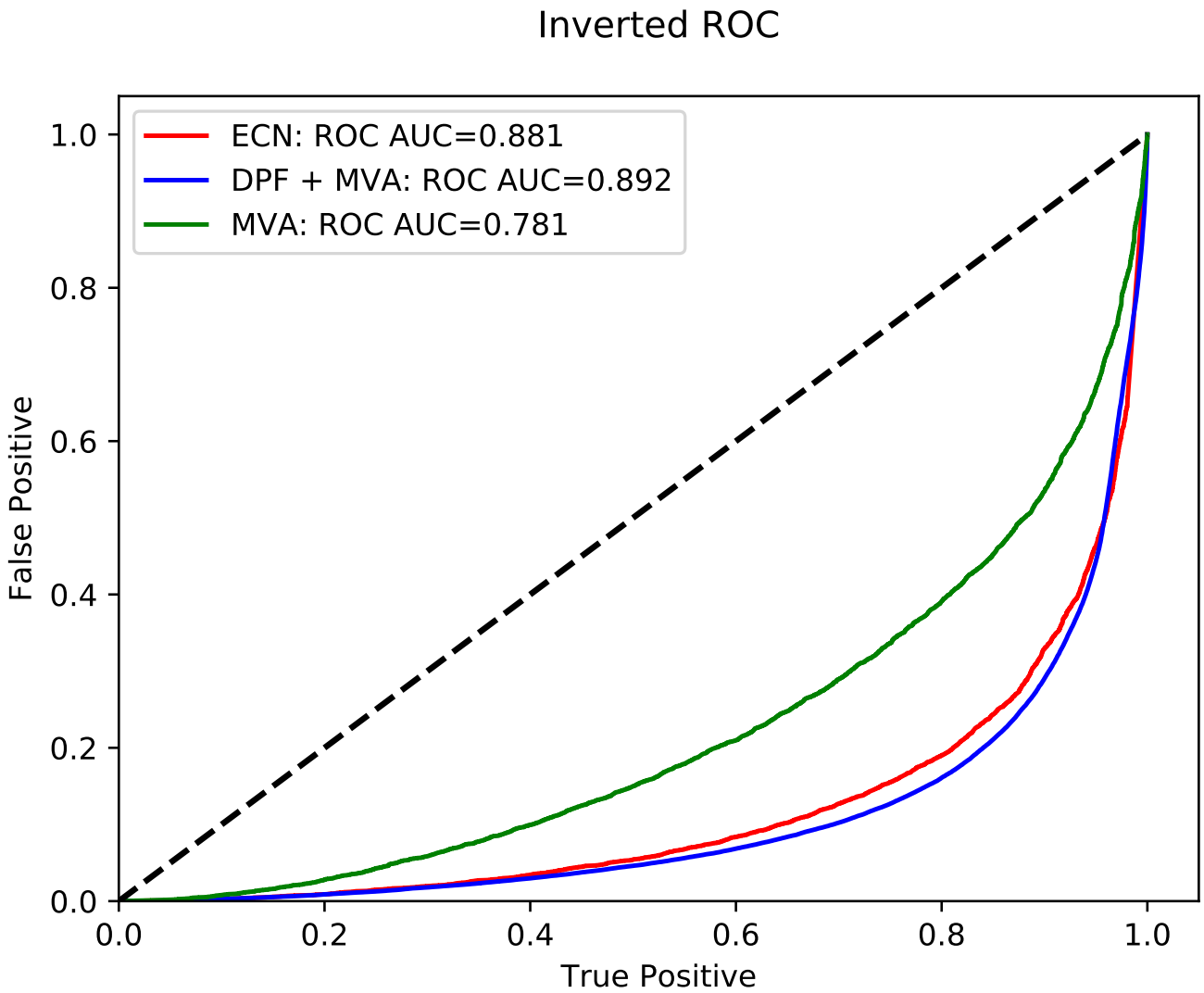
https://arxiv.org/abs/1512.03385

Dense Network block

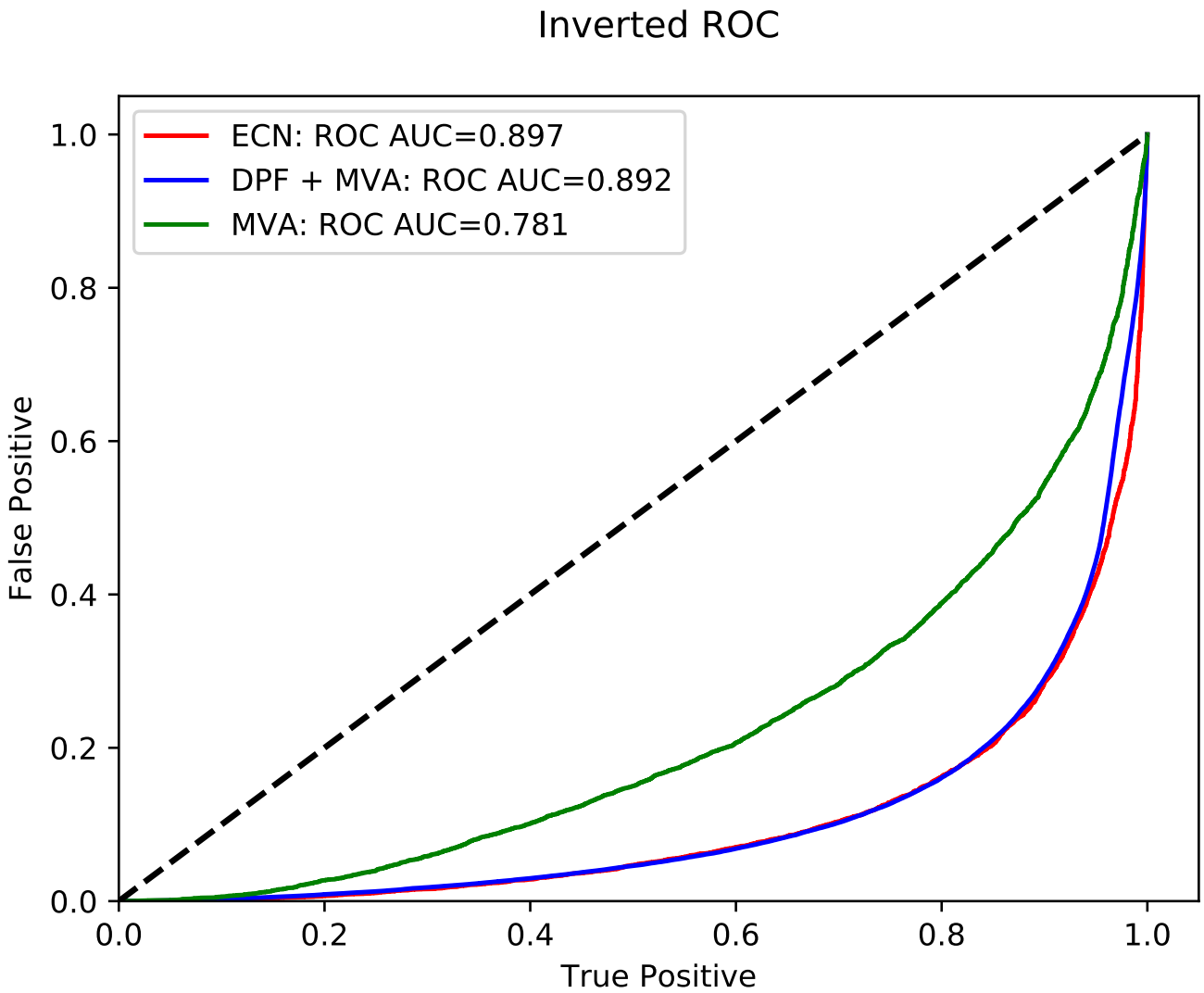


https://arxiv.org/abs/1608.06993

ECN performance (ResNet)



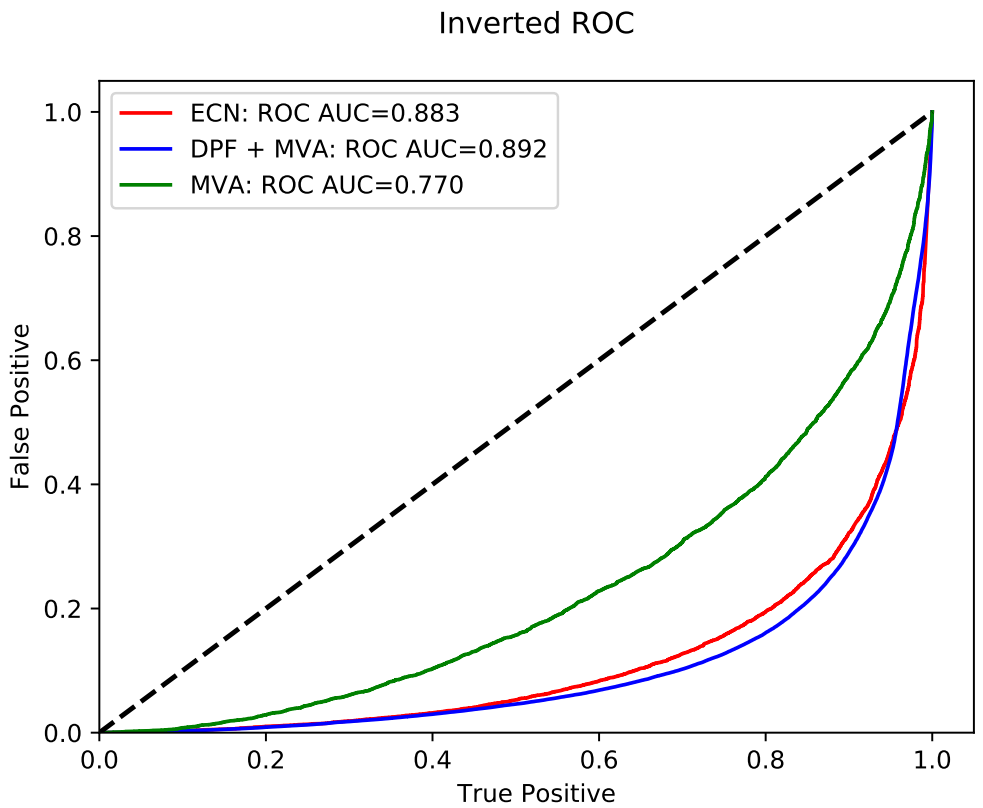
ECN performance (DenseNet)



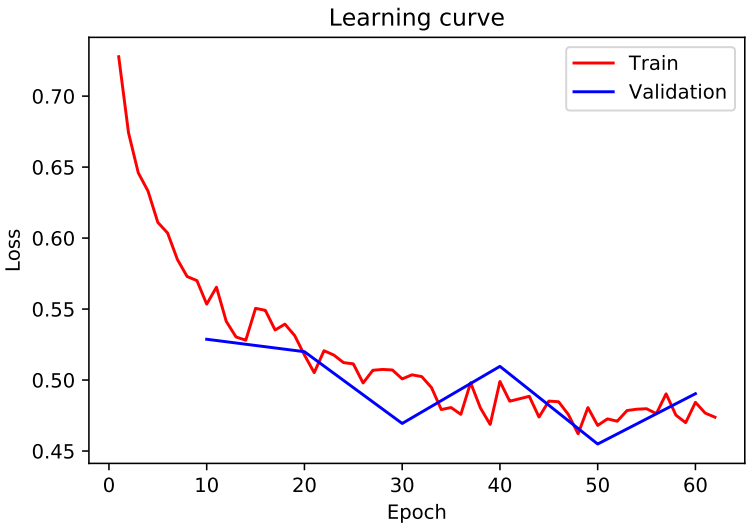
ECN performance (1 Conv layer, Cosine Annealing)

https://arxiv.org/abs/1608.03983

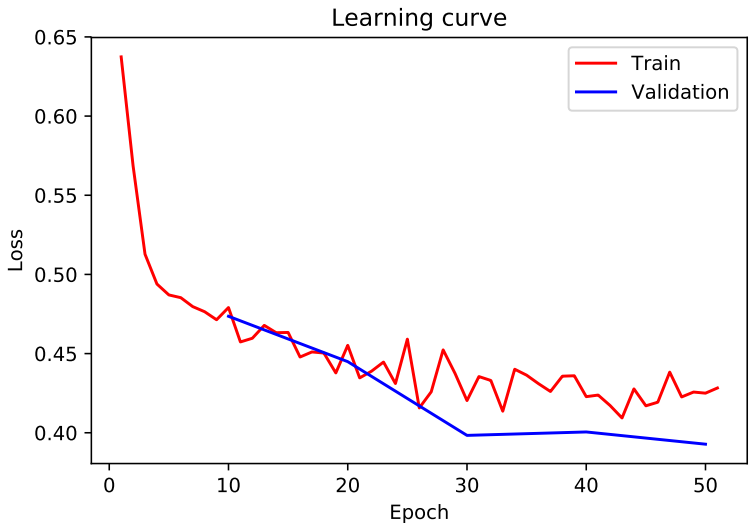
$$\eta_t = \eta_{min} + \frac{1}{2}(\eta_{max} - \eta_{min})(1 + \cos(\frac{T_{cur}}{T_i}\pi))$$



Constant LR vs. Cosine Annealing



Constant LR



Cosine Annealing

Results

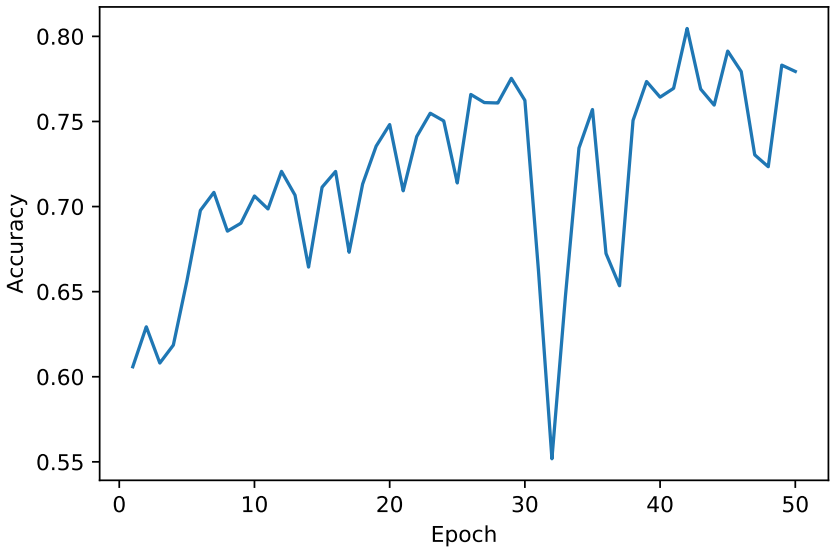
Architecture	ROC AUC score	Number of parameters
MVA	0.792	-
DPF	0.863	8838945
DPF + MVA	0.892	-
ECN (1 layer)	0.853	**44419**
ECN (3 layers)	**0.898**	223939
ECN (separated features)	0.891	227011
ECN-ResNet (3 layers)	0.881	279683
ECN-DenseNet (3 layers)	0.897	265475
ECN (1 layer)	0.883	49283

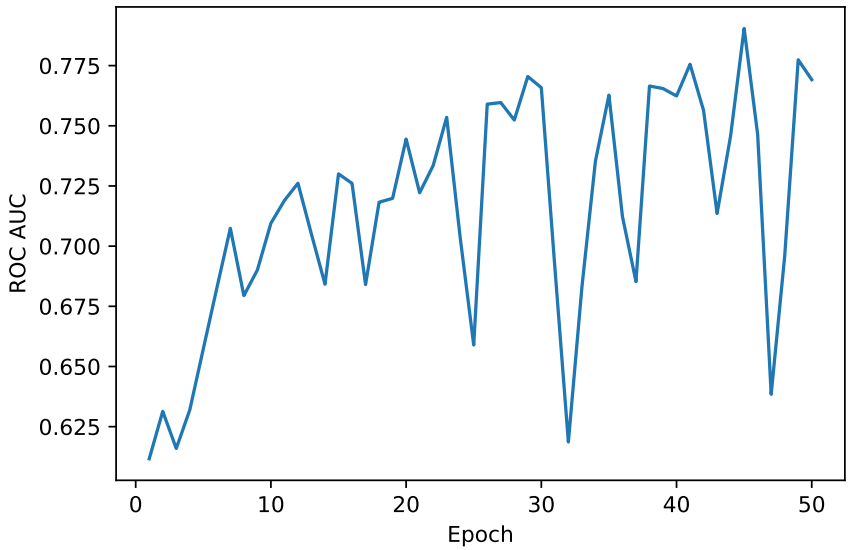
Conclusions and plans

- DPF outperforms MVA
- Modifications of DPF do not bring significant improvement of performance
- EdgeConv Network outperforms DPF, while having less number of parameters
- Next steps:
  - Train Graph Networks with different types of convolutions
  - Train Graph Networks with pooling layers
  - Optimize the hyperparameters

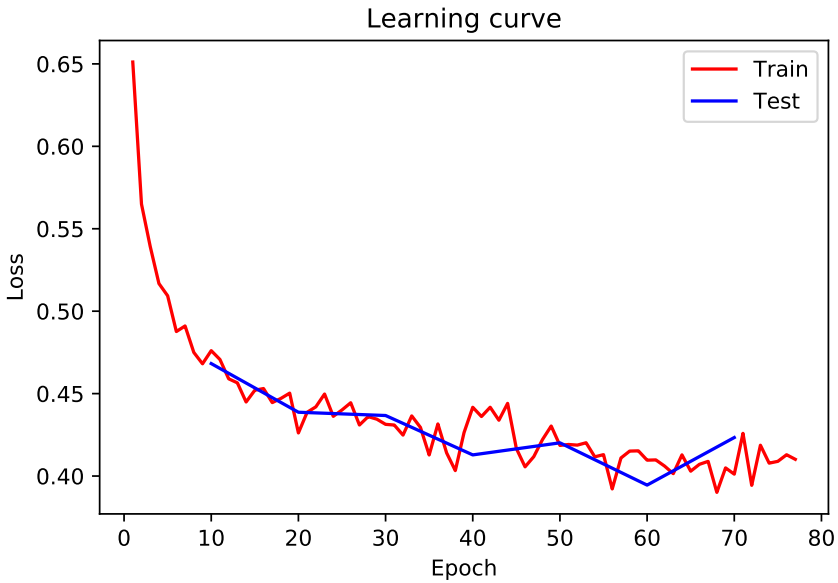
Backup

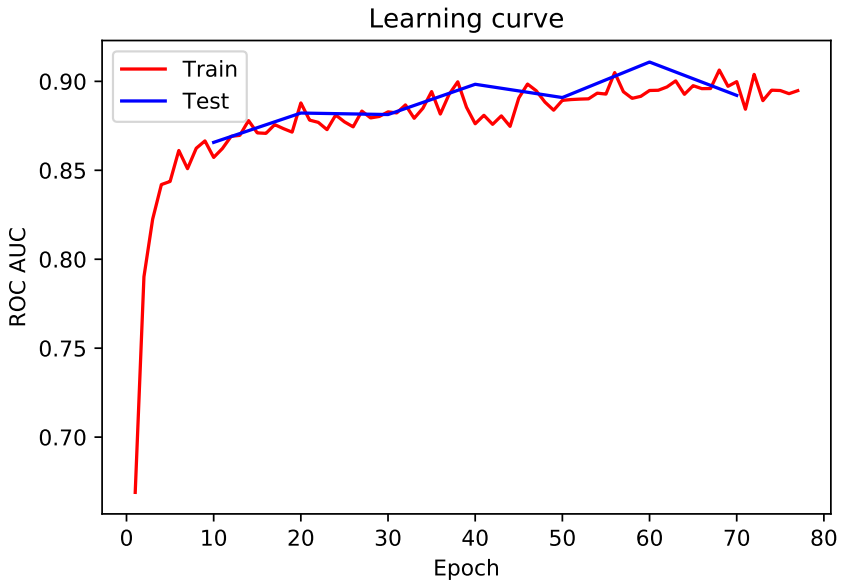
Learning curve for DPF (for training sample)





Learning curve for ECN with 3 layers





Output distributions

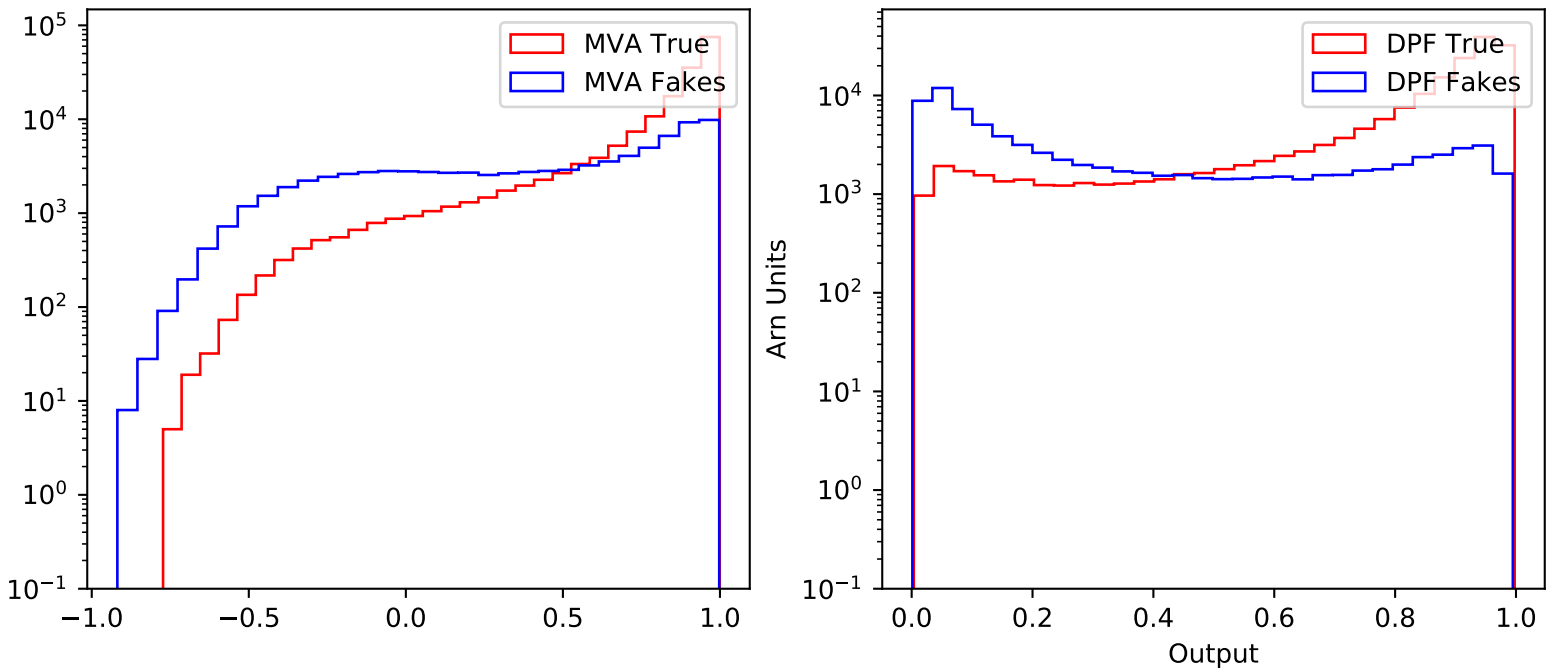


Figure: Output distribution for all decay modes

Output distributions for different decay modes

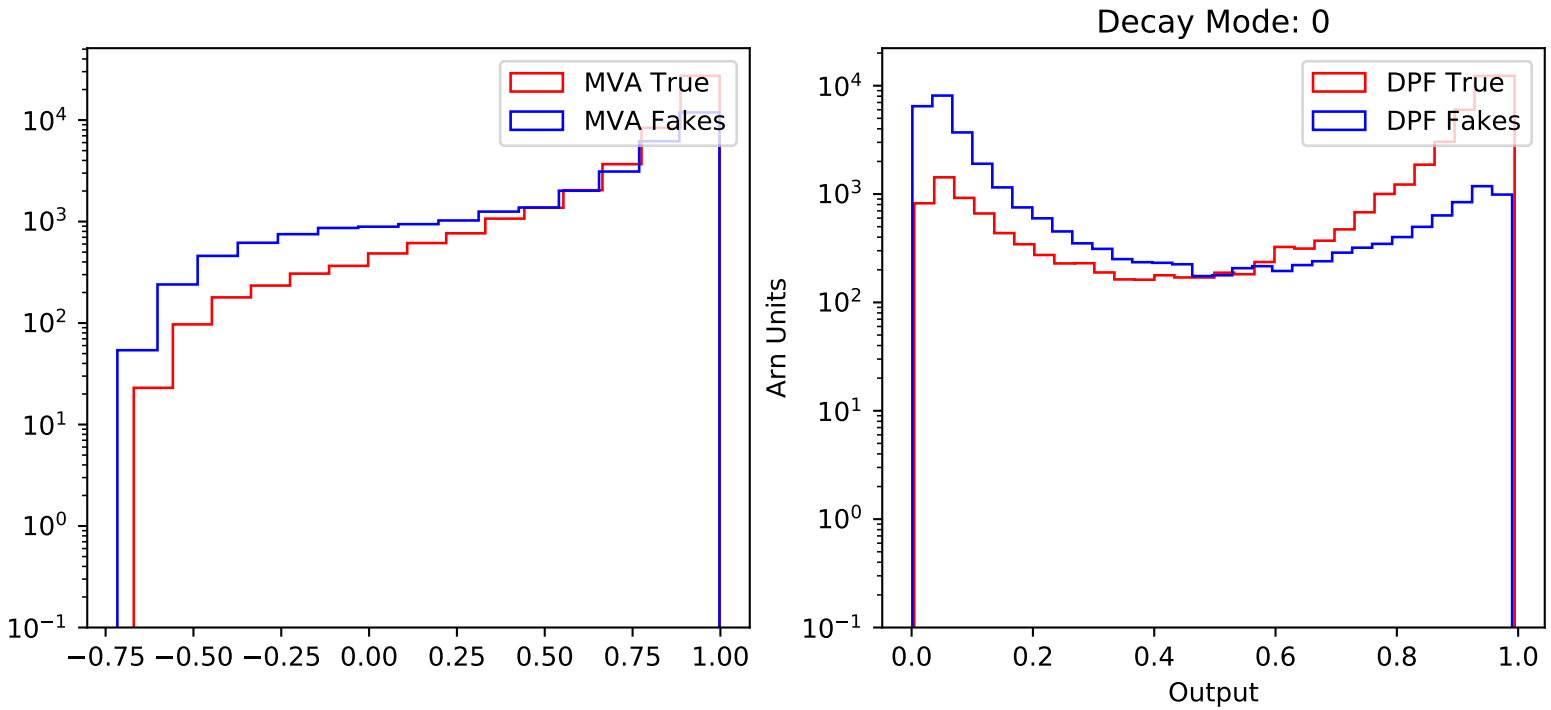


Figure: Output distribution for decay mode 0 (into one charged hadron)

Output distributions for different decay modes

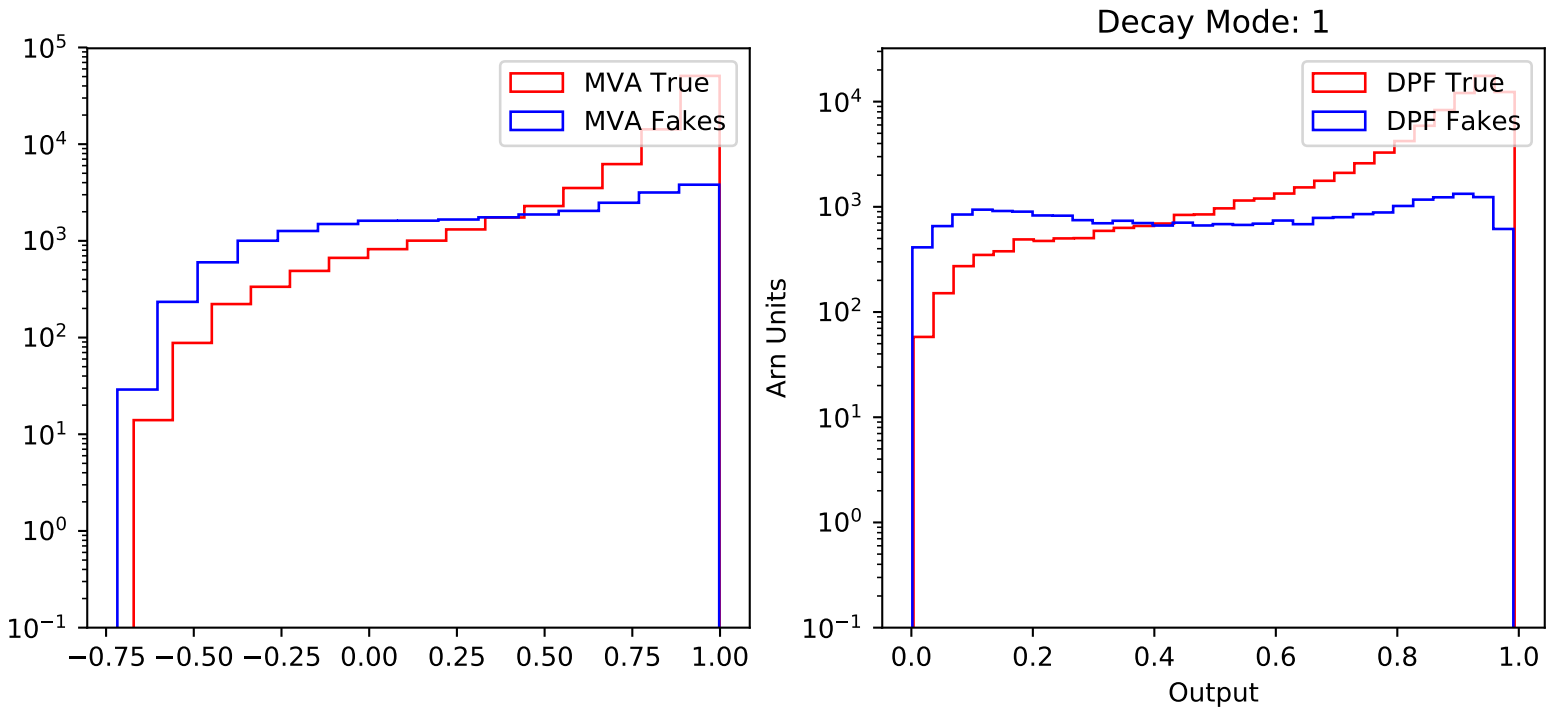


Figure: Output distribution for decay mode 1 (into one charged hadron and a neutral pion)

Output distributions for different decay modes

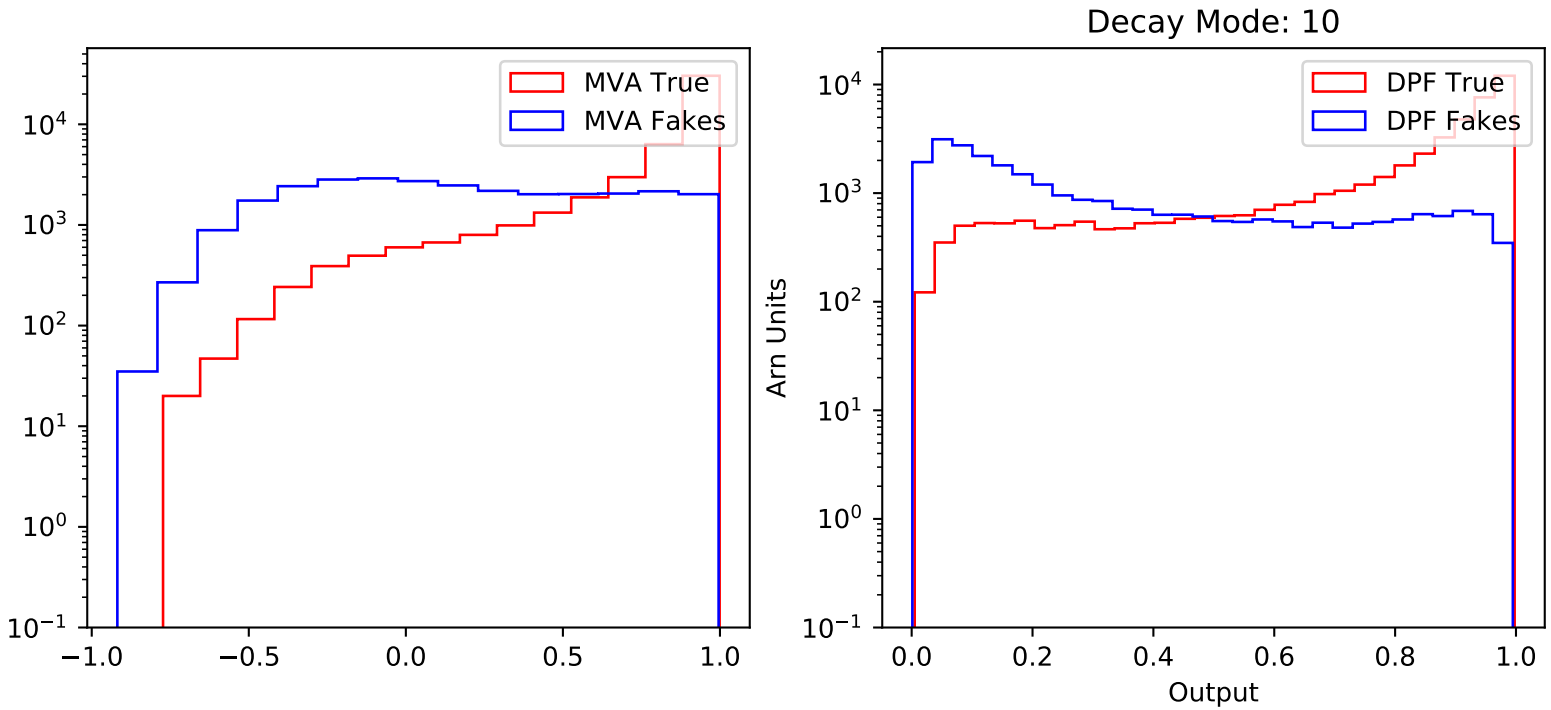
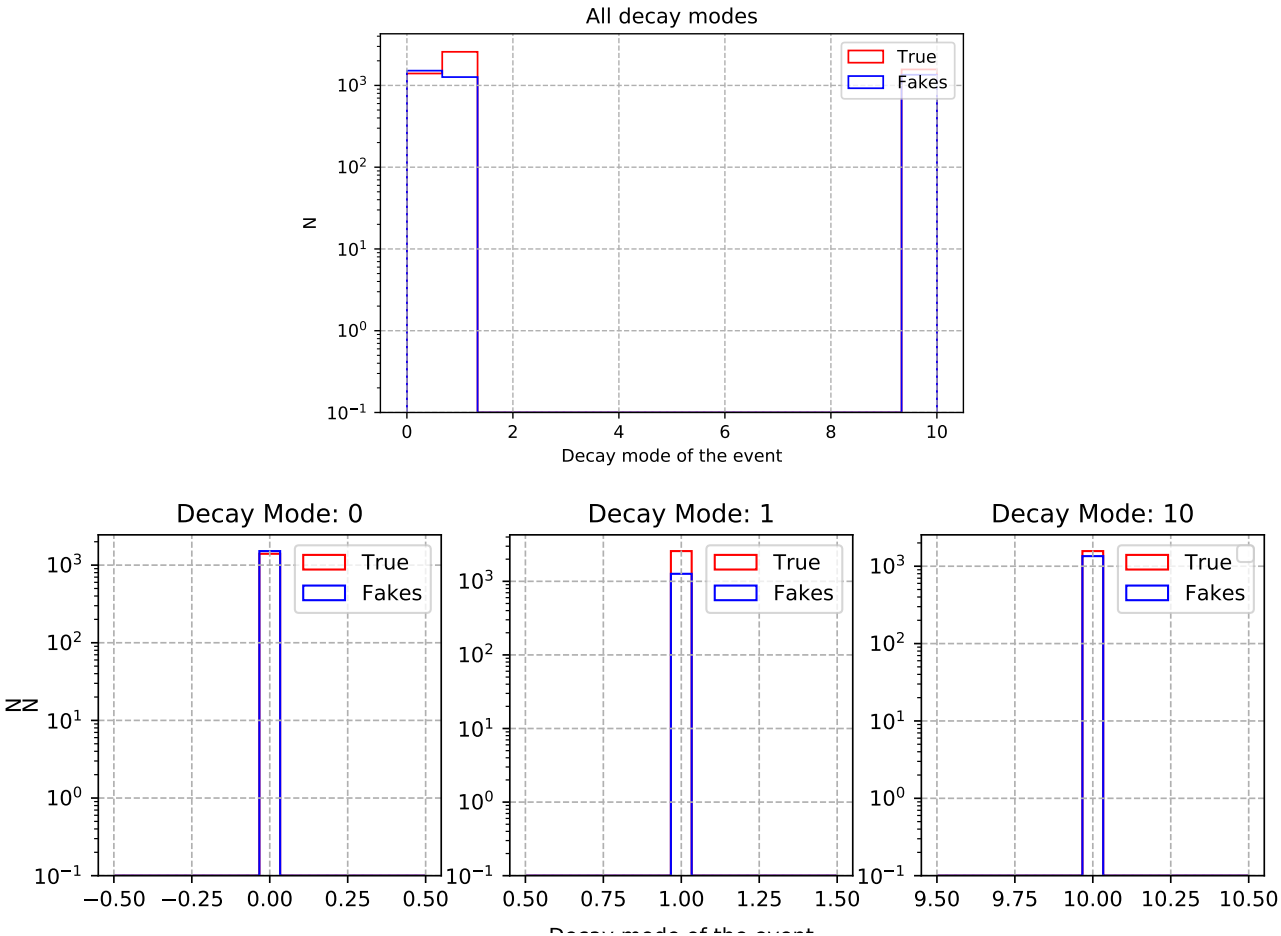
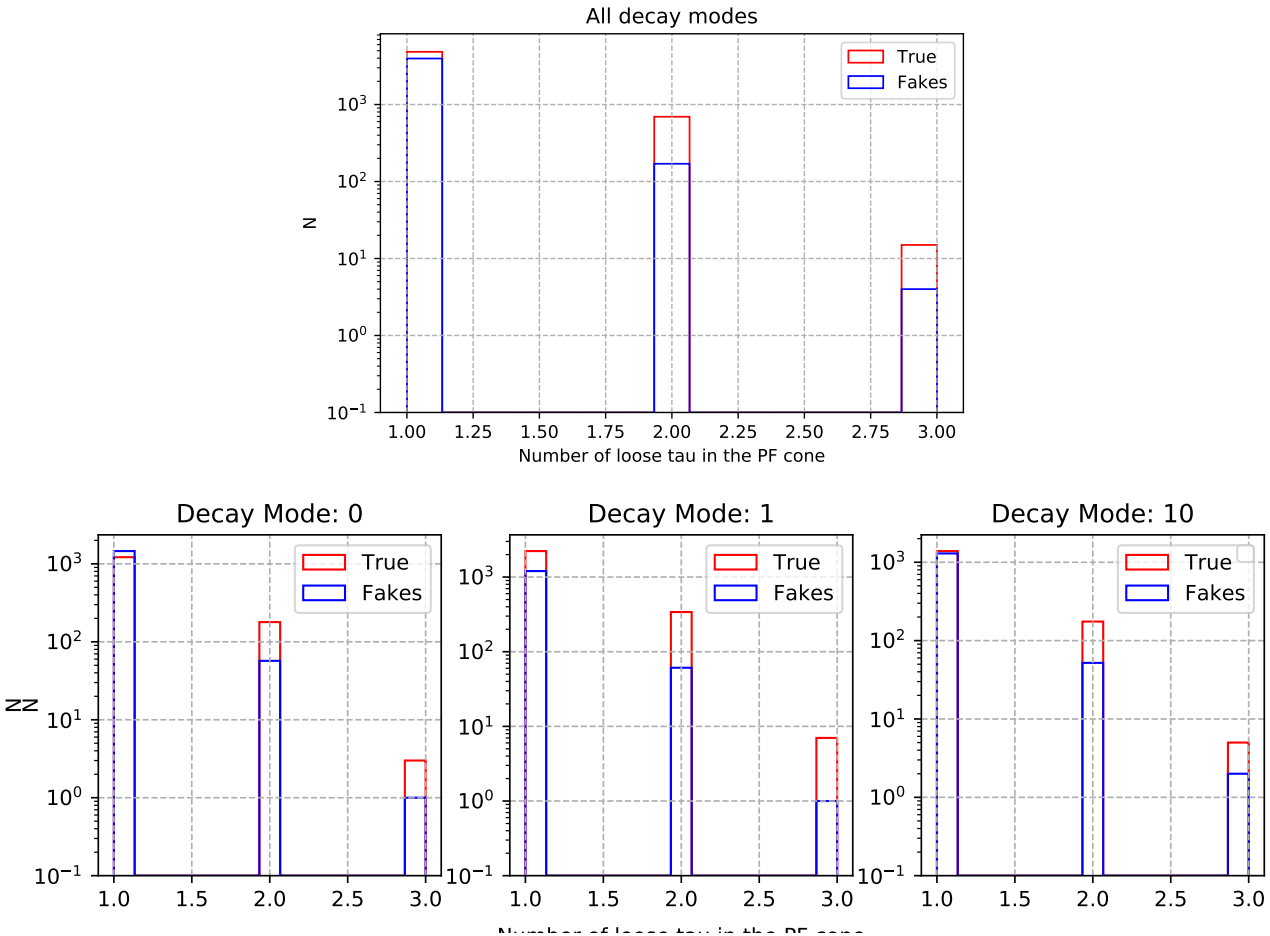


Figure: Output distribution for decay mode 10 (into three charged hadrons)

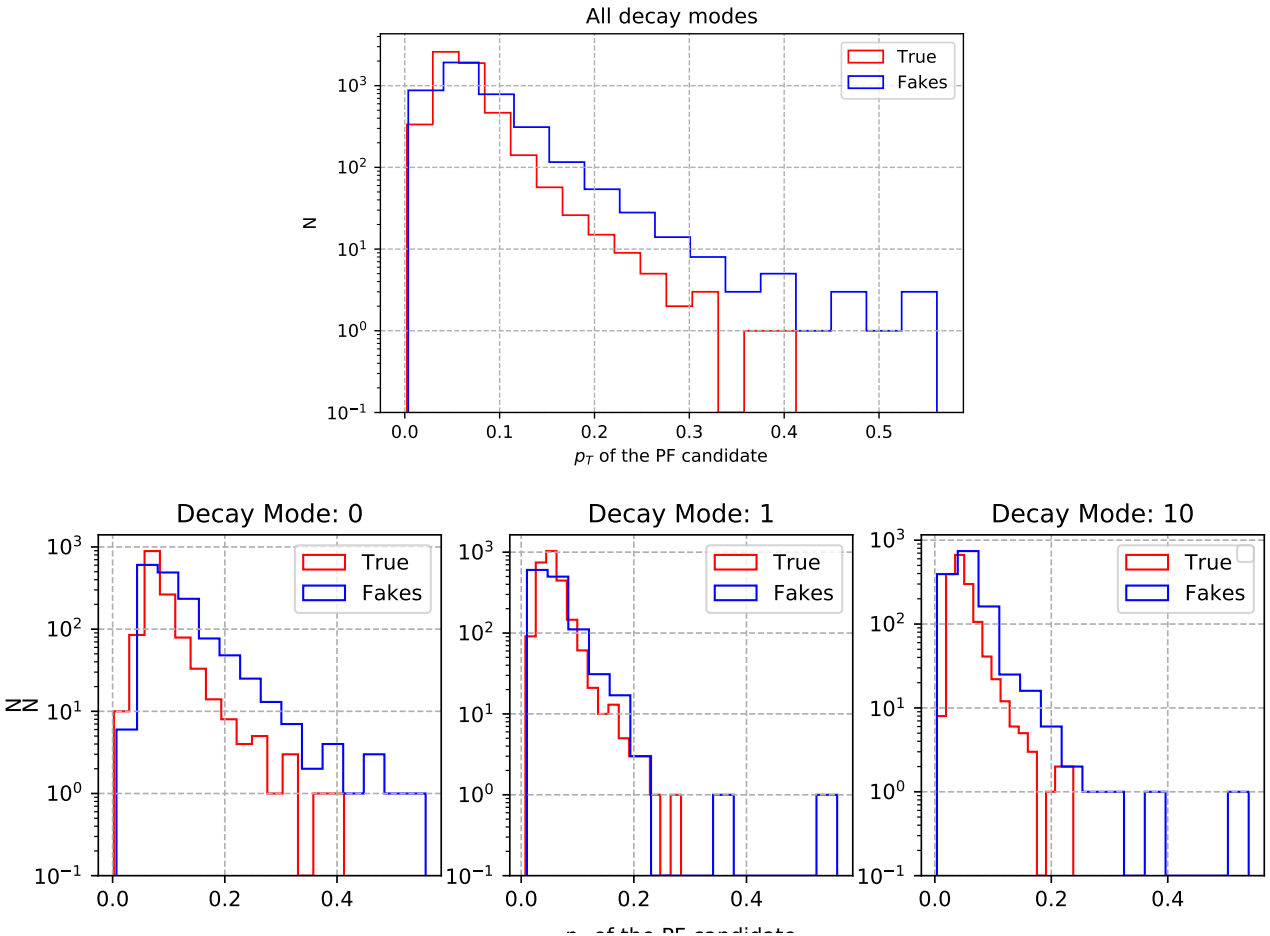
Feature: Decay mode of the event



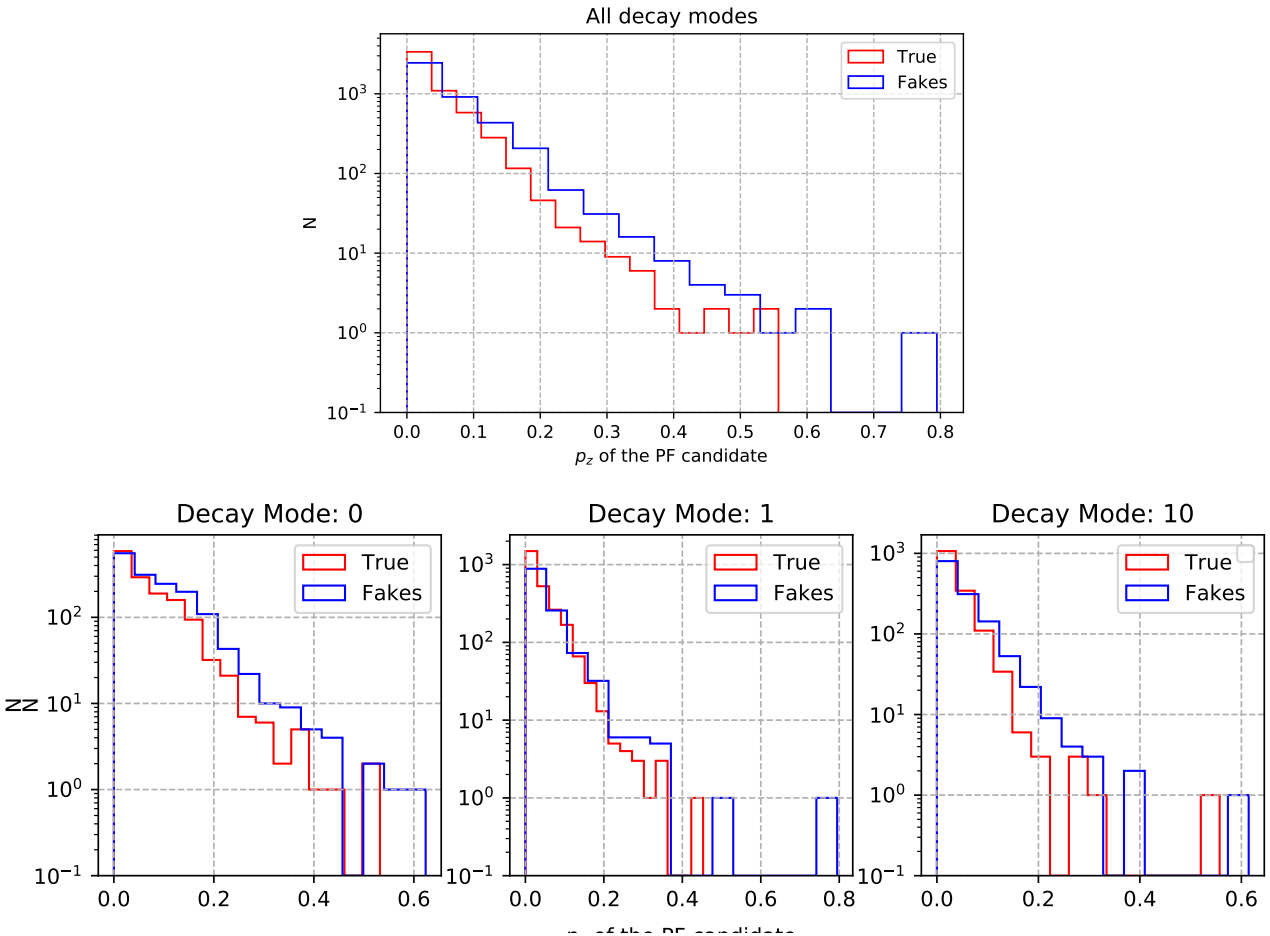
Feature: Number of loose tau in the PF cone



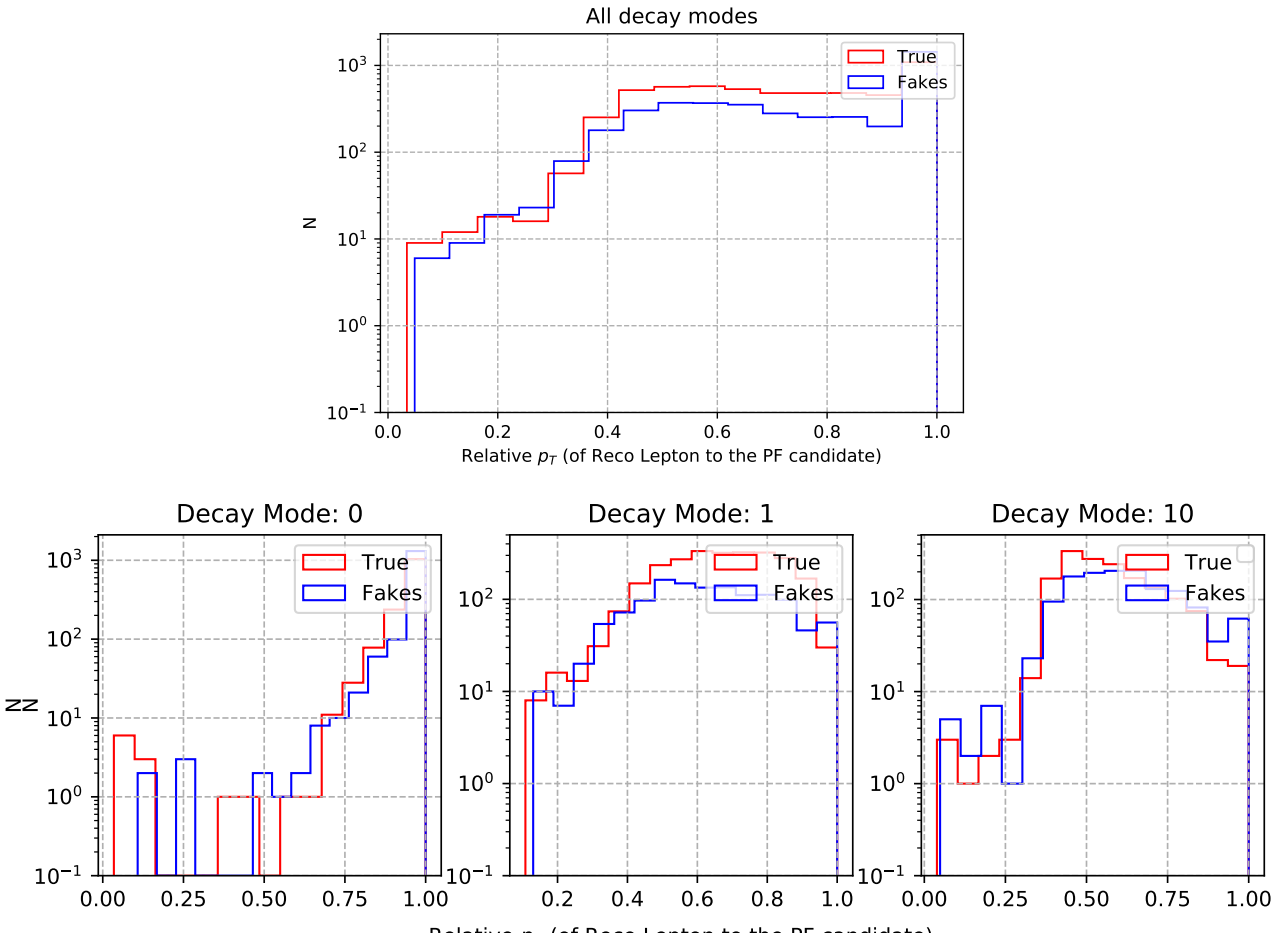
Feature: p_T of the PF candidate



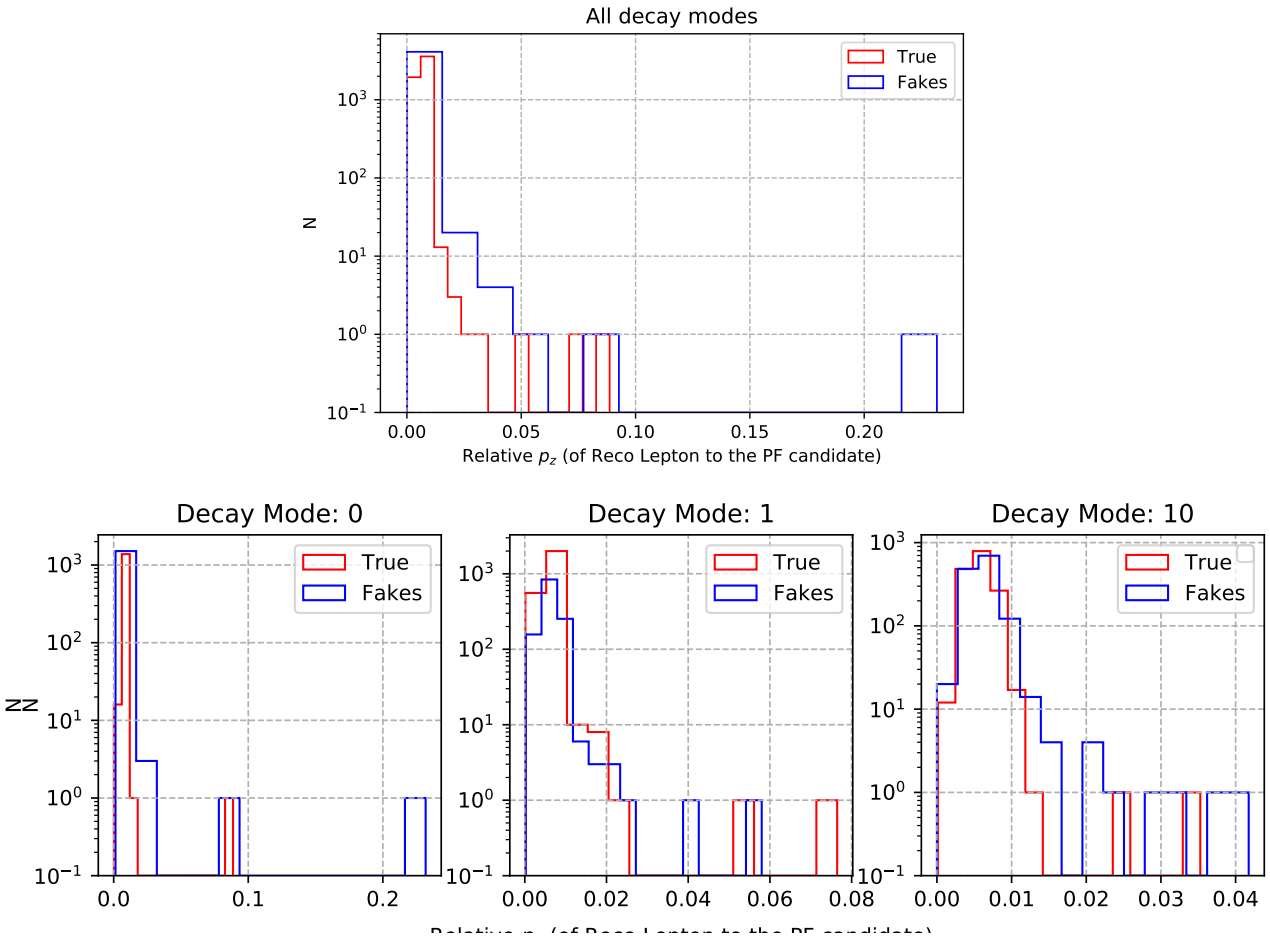
Feature: p_z of the PF candidate



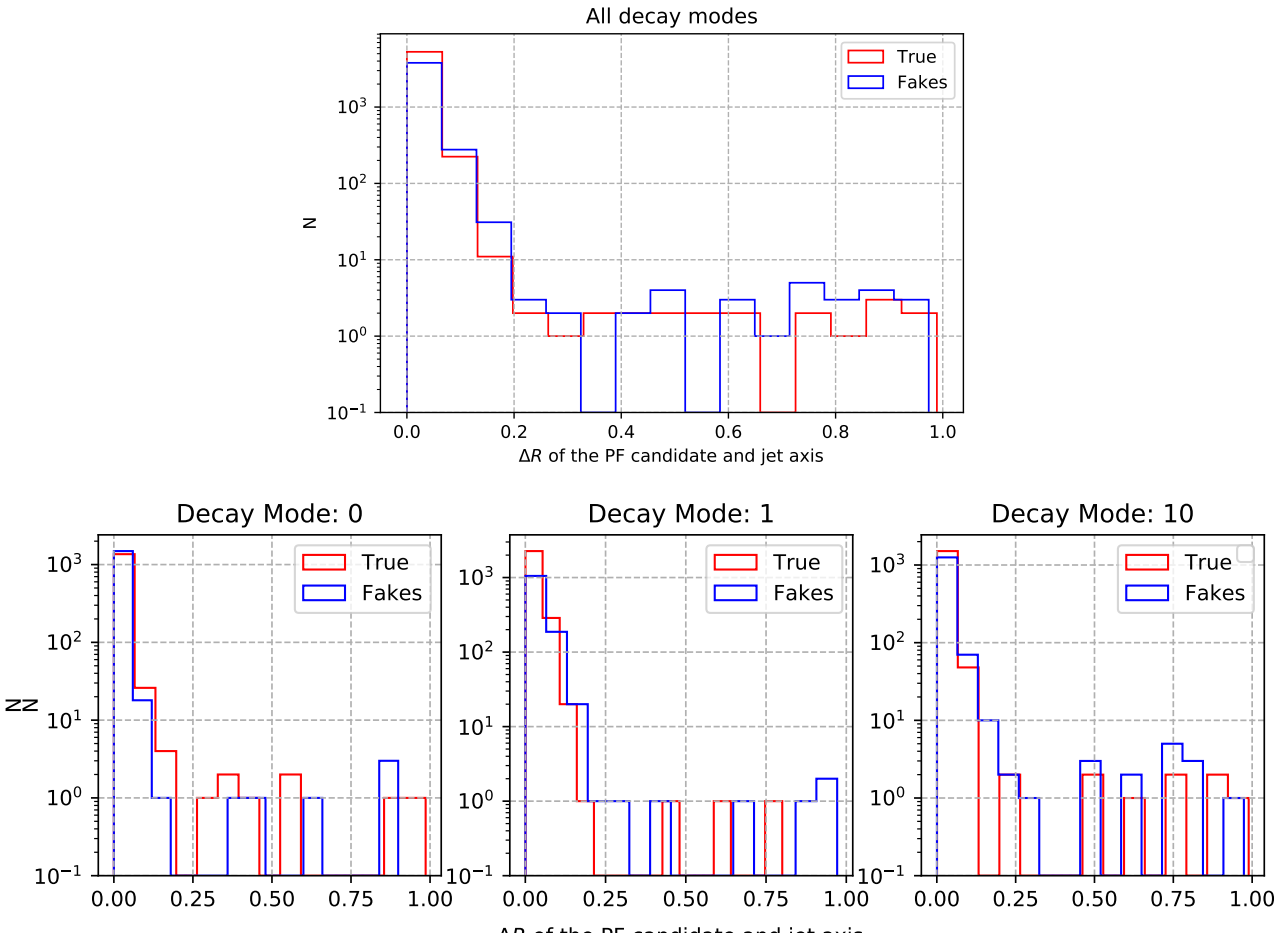
Feature: Relative p_T (of Reco Lepton to the PF candidate)



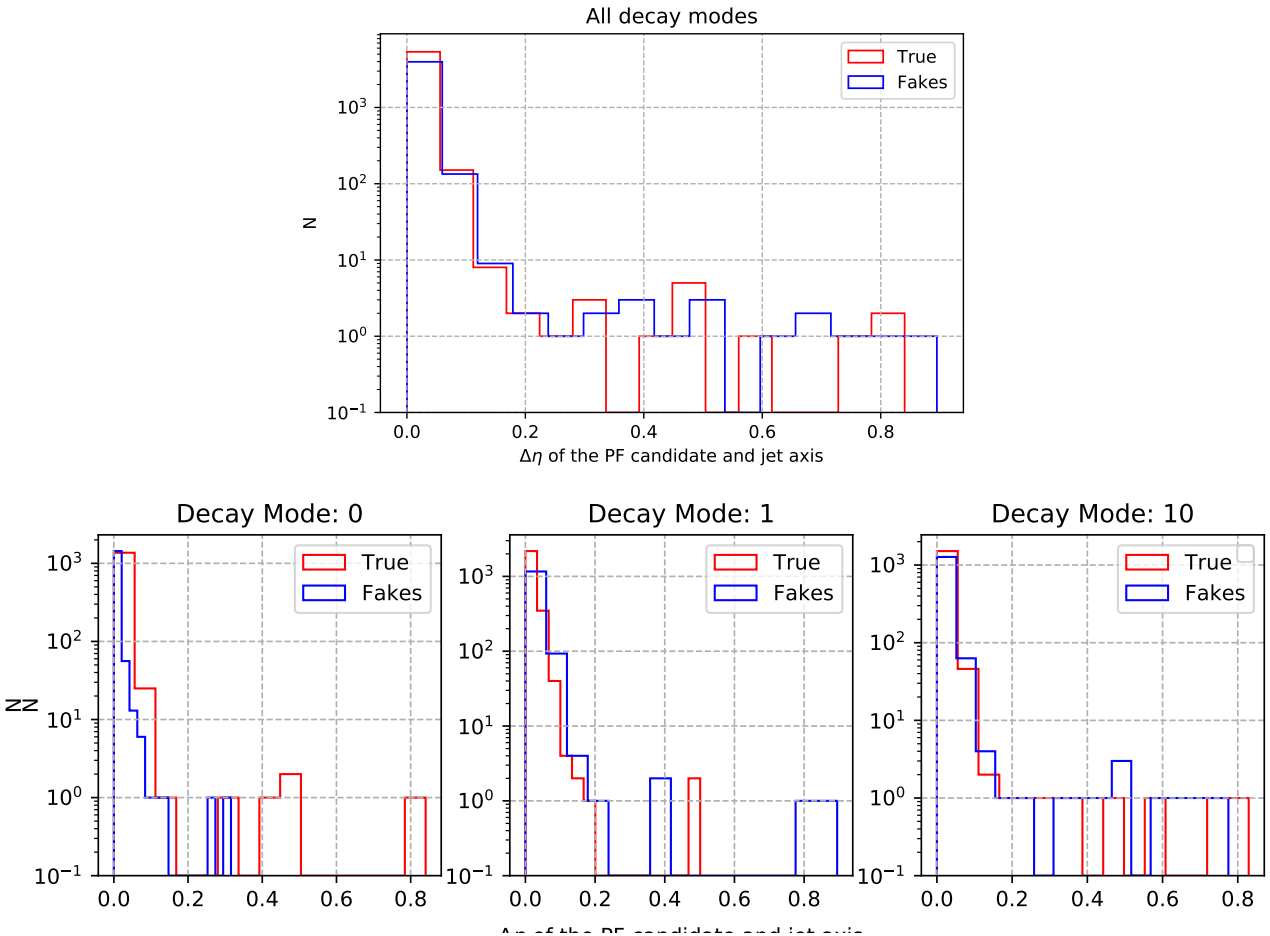
Feature: Relative p_z (of Reco Lepton to the PF candidate)



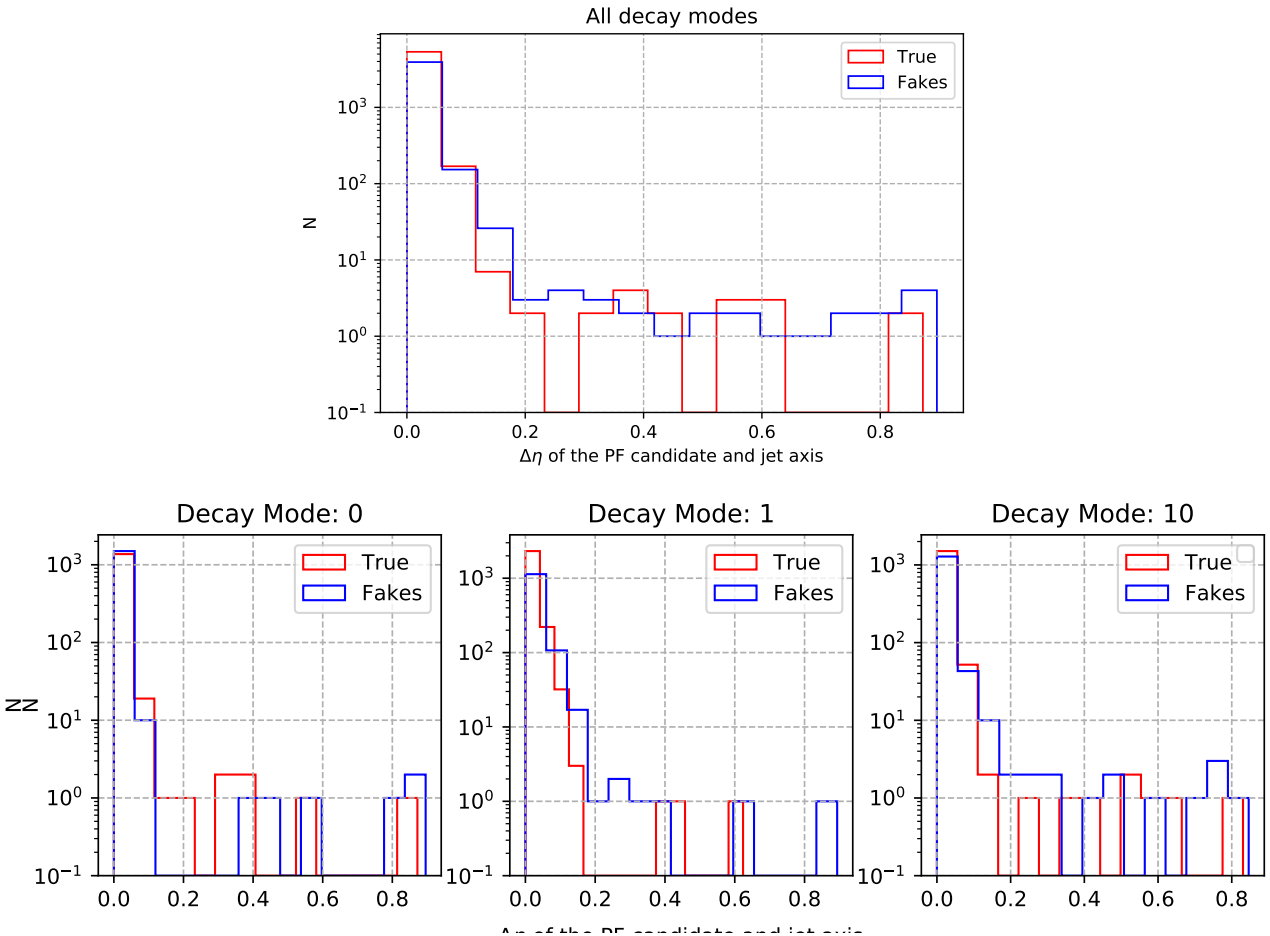
Feature: ΔR of the PF candidate and jet axis



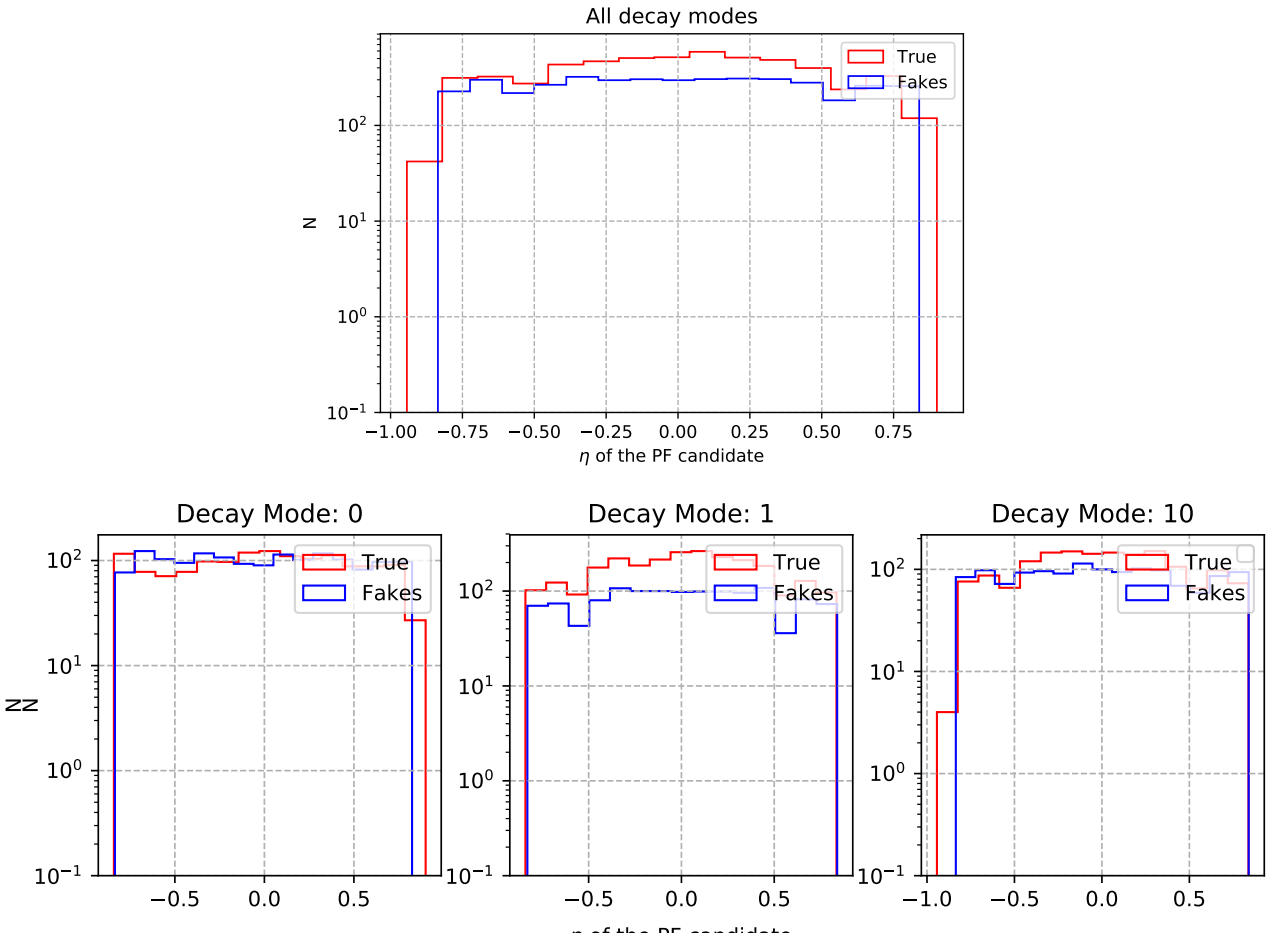
Feature: $\Delta\eta$ of the PF candidate and jet axis



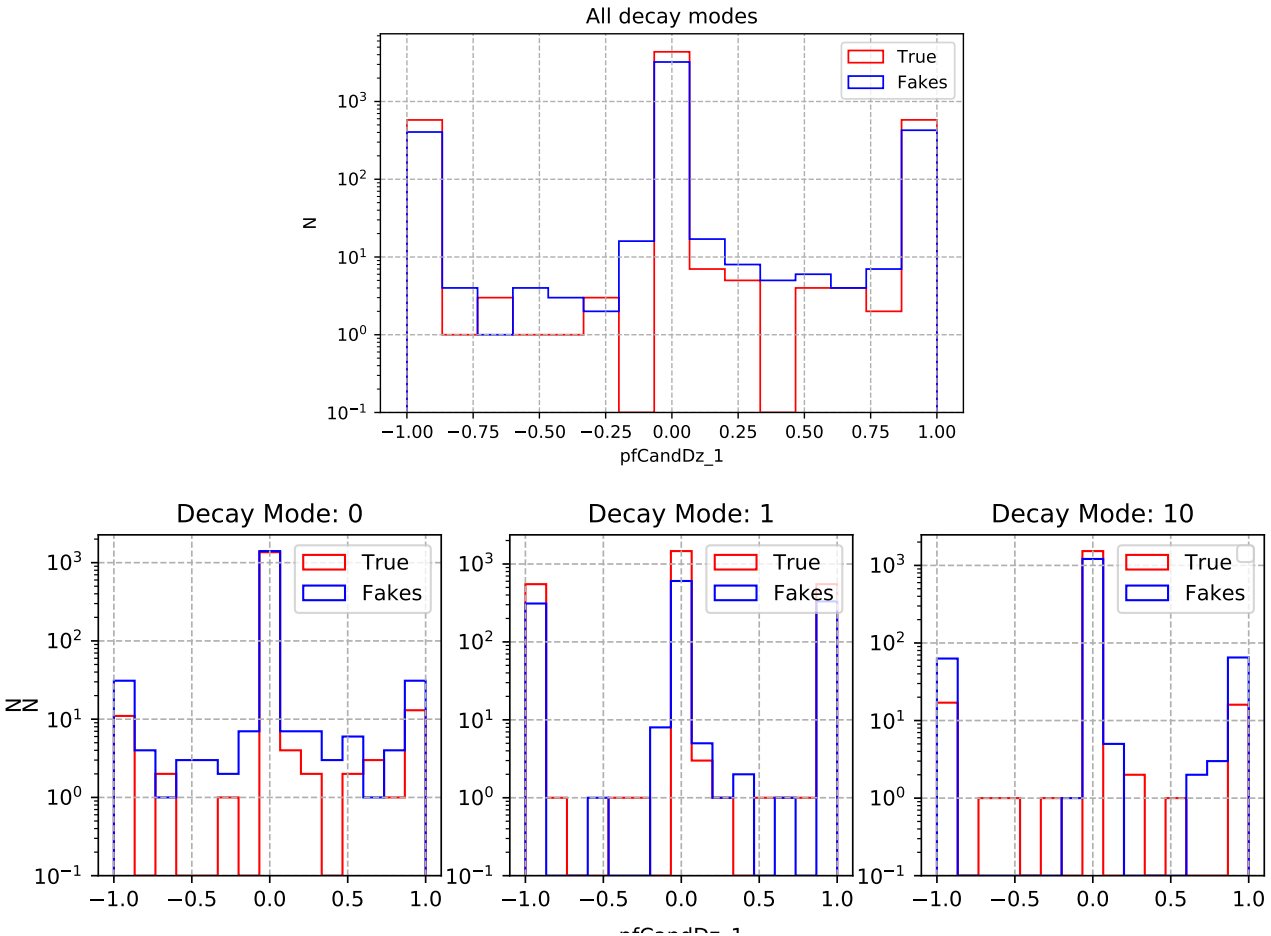
Feature: $\Delta\phi$ of the PF candidate and jet axis



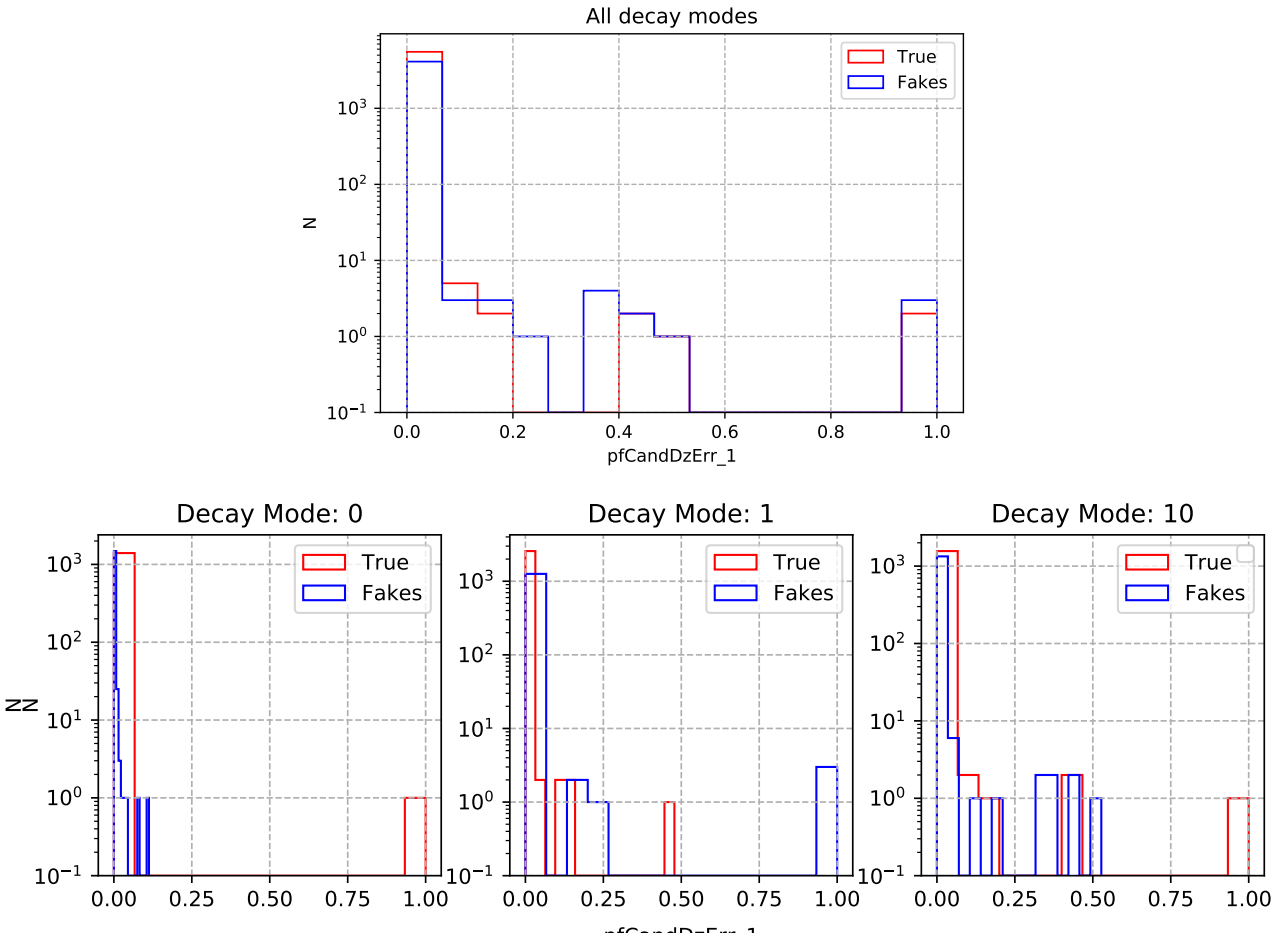
Feature: η of the PF candidate



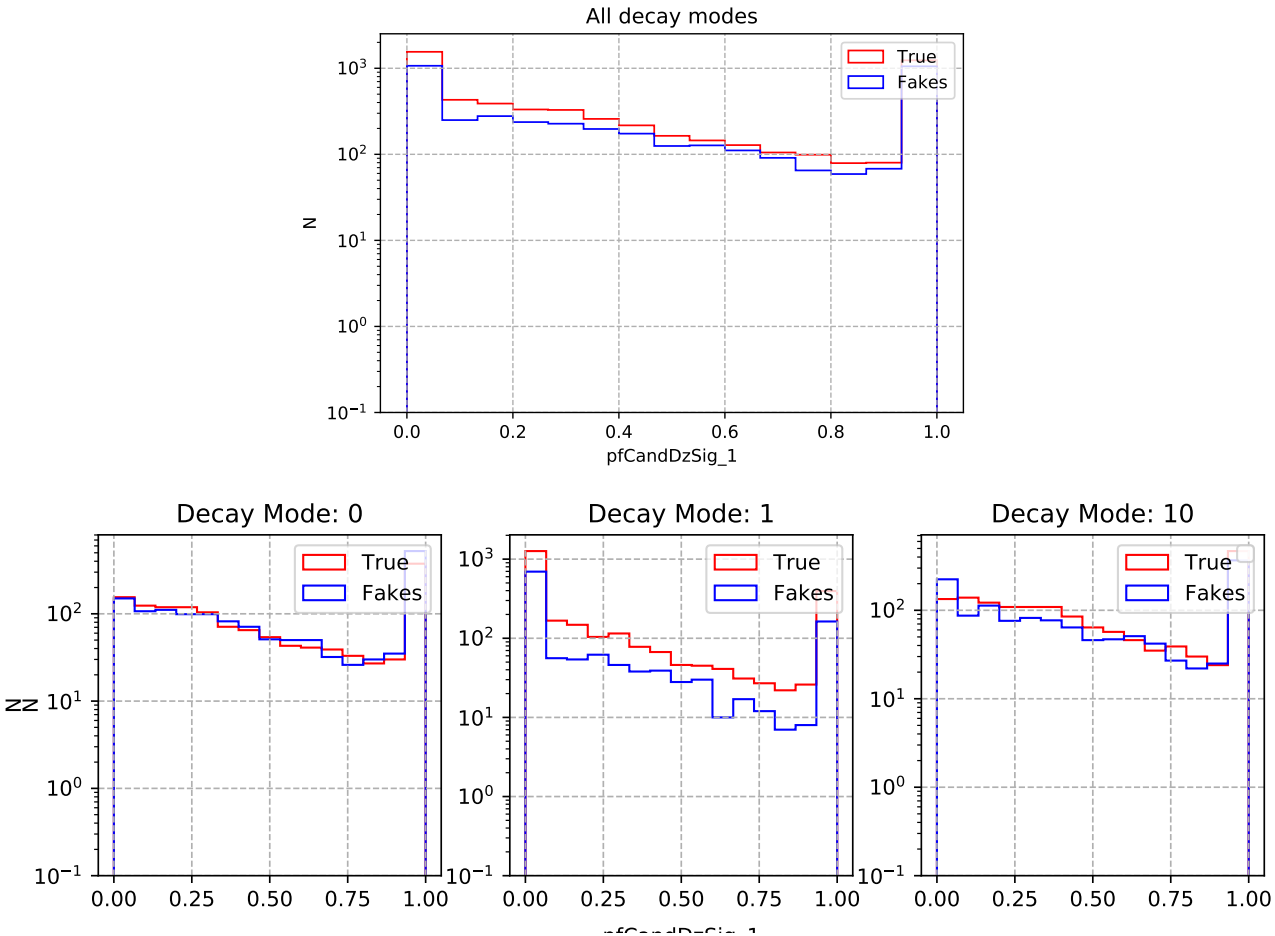
Feature: pfCandDz$_1$



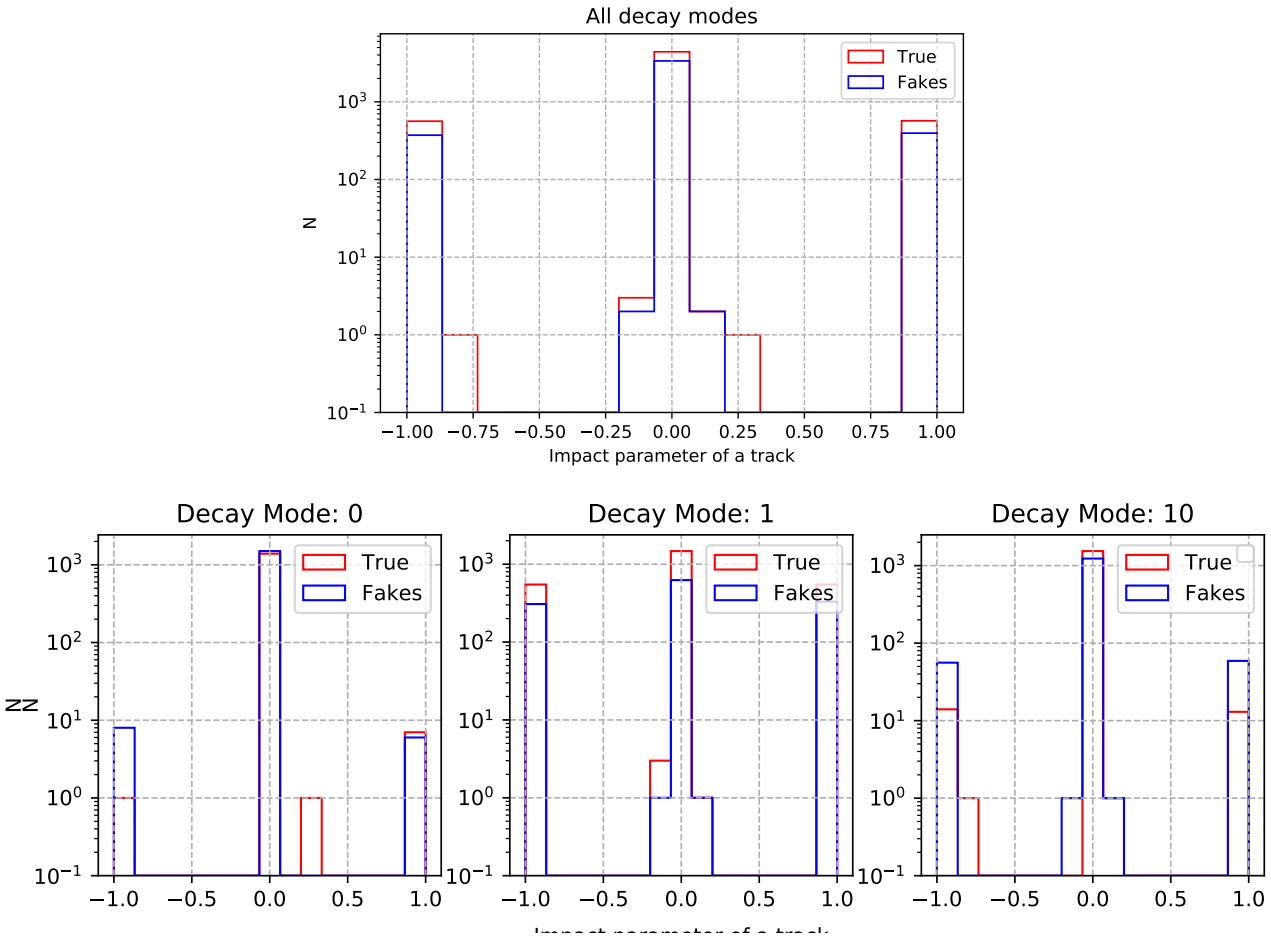
Feature: pfCandDzErr_1



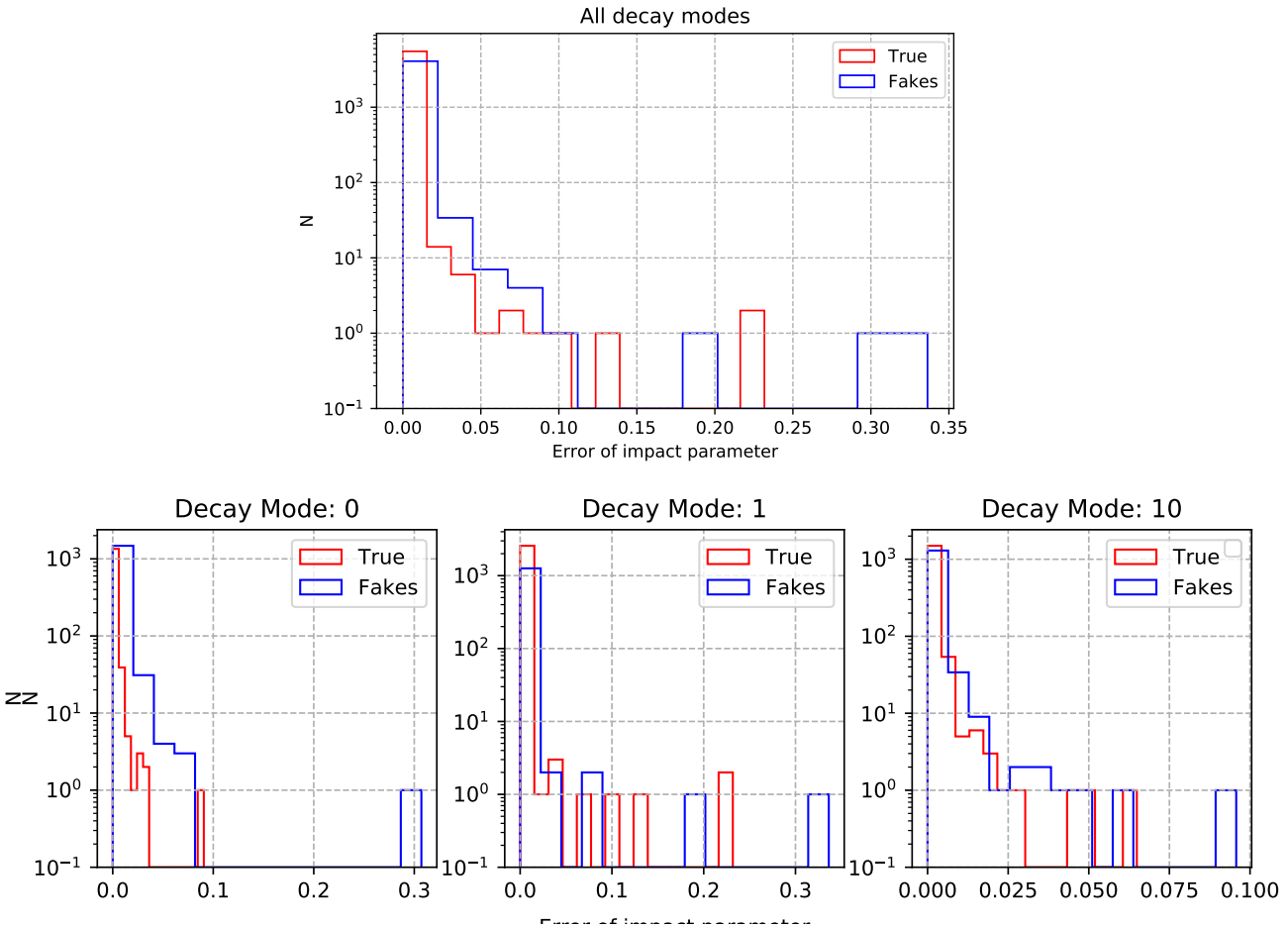
Feature: pfCandDzSig$_1$



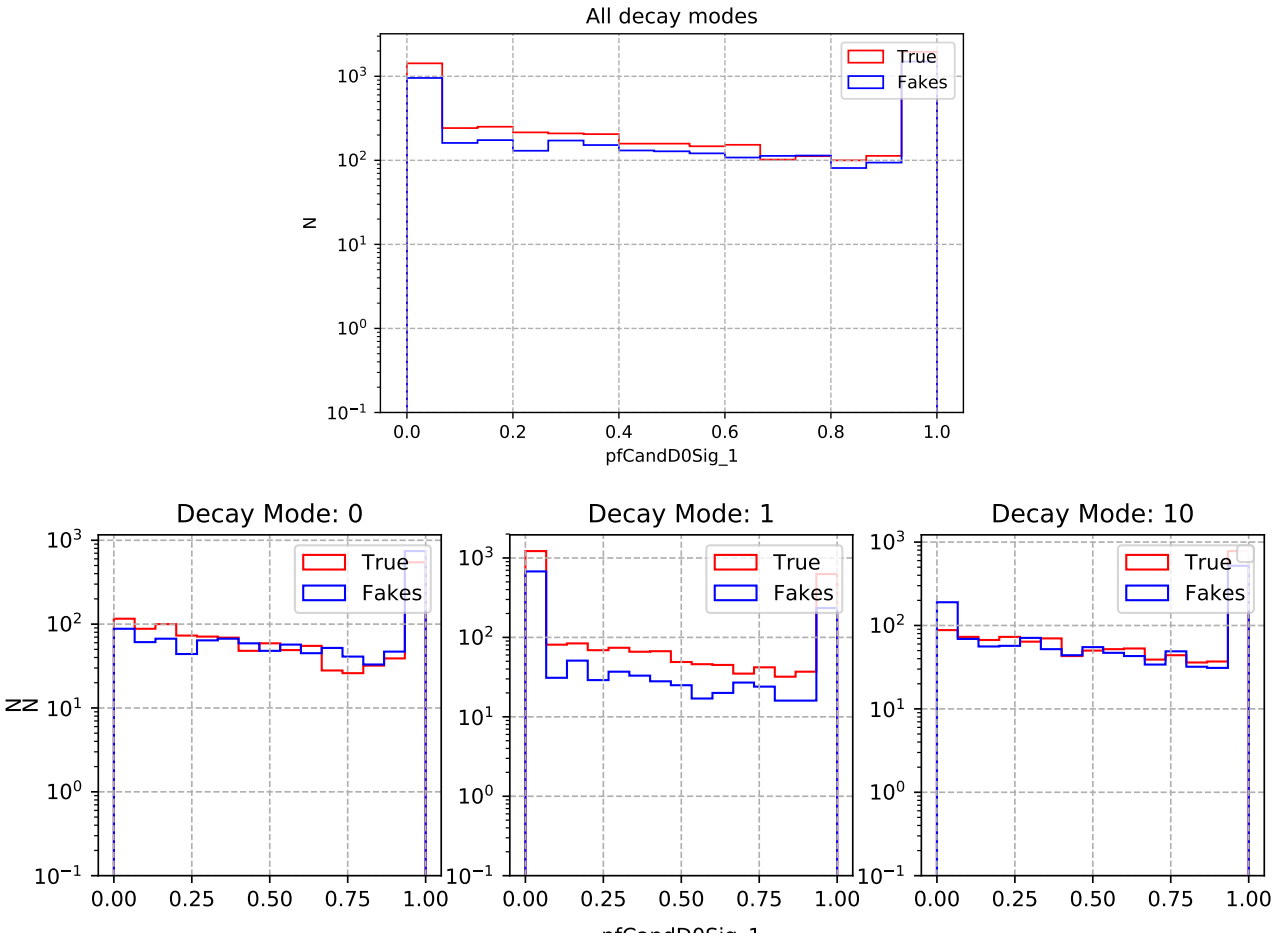
Feature: Impact parameter of a track



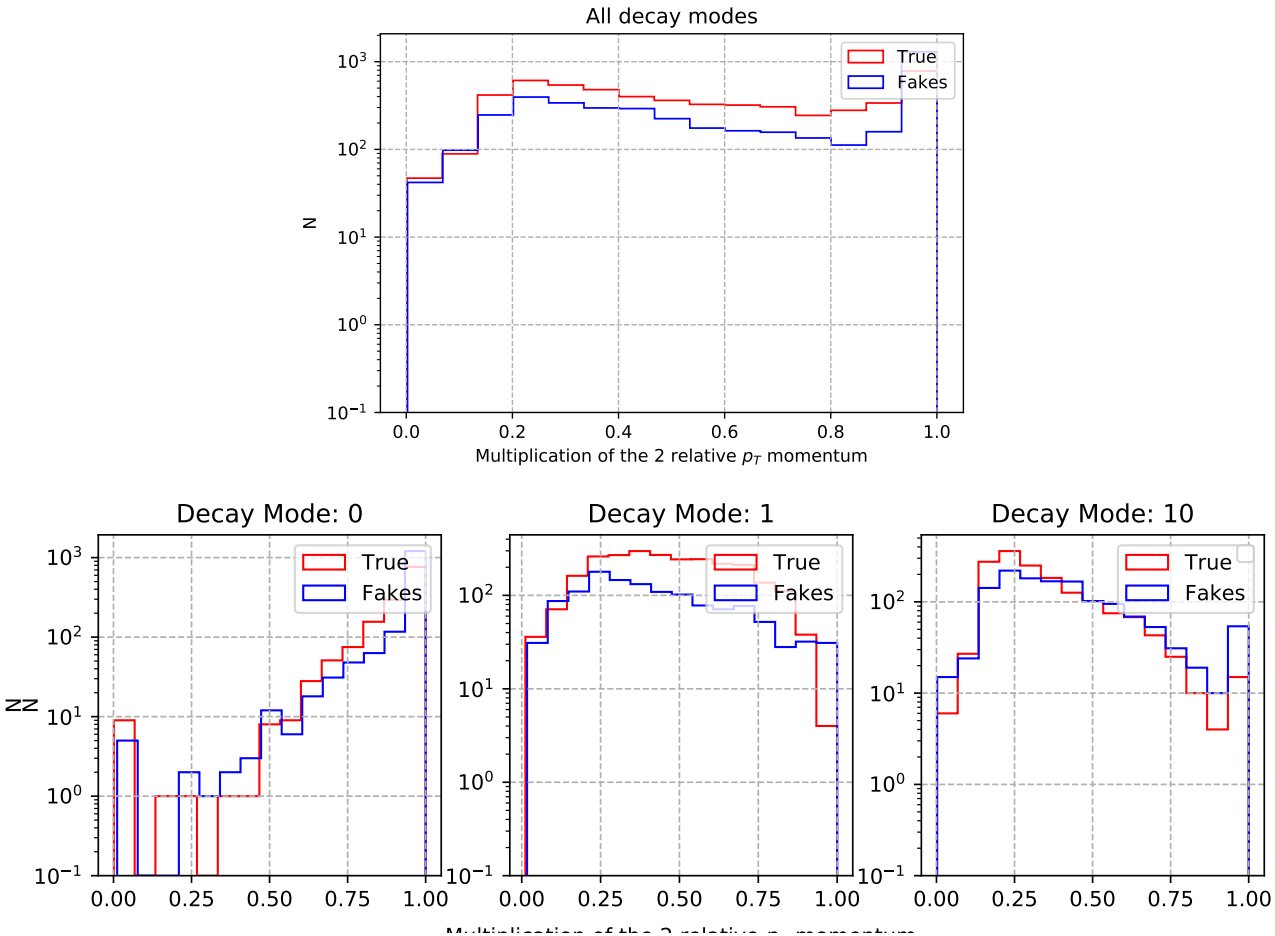
Feature: Error of impact parameter



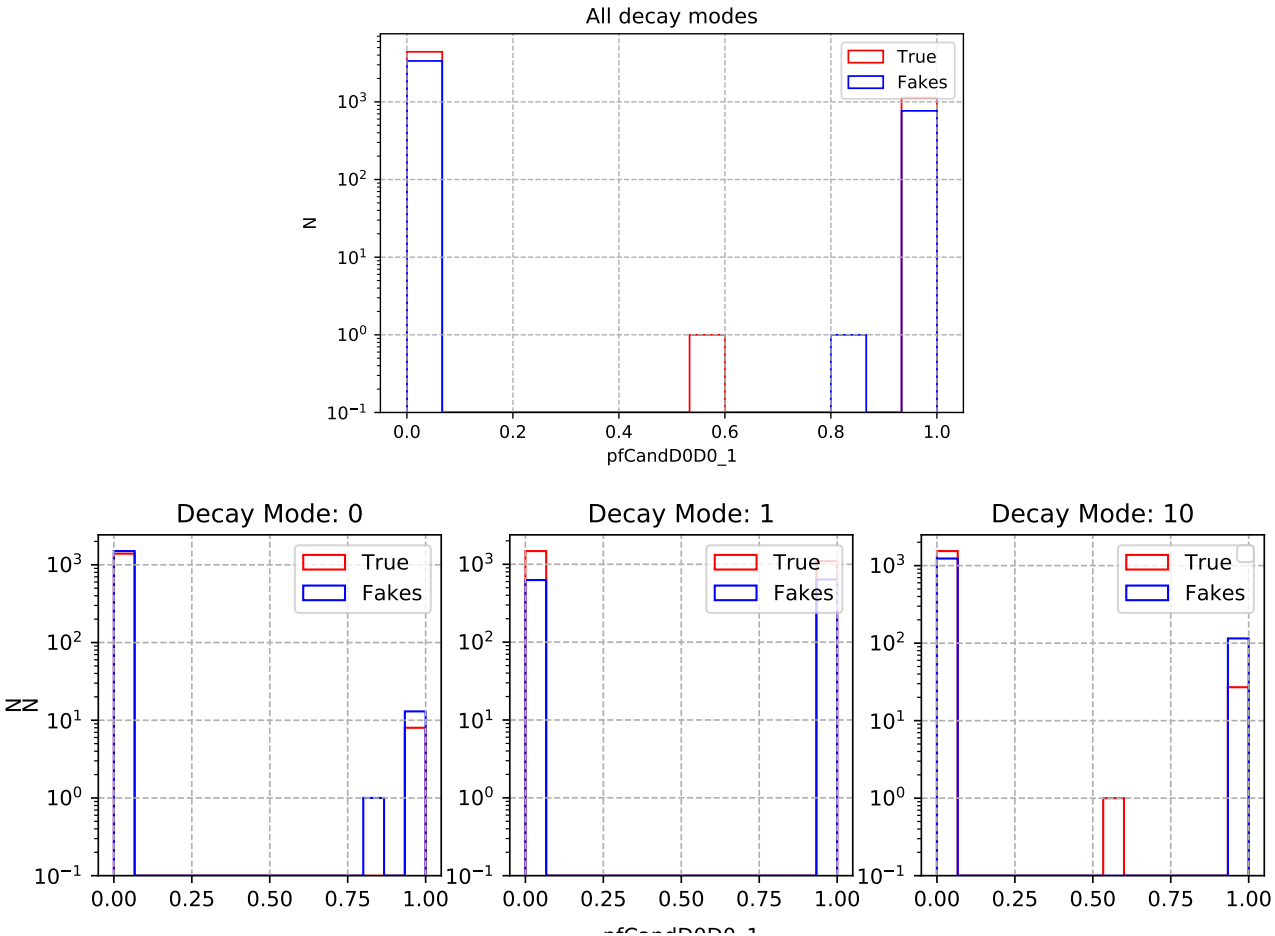
Feature: pfCandD0Sig$_1$



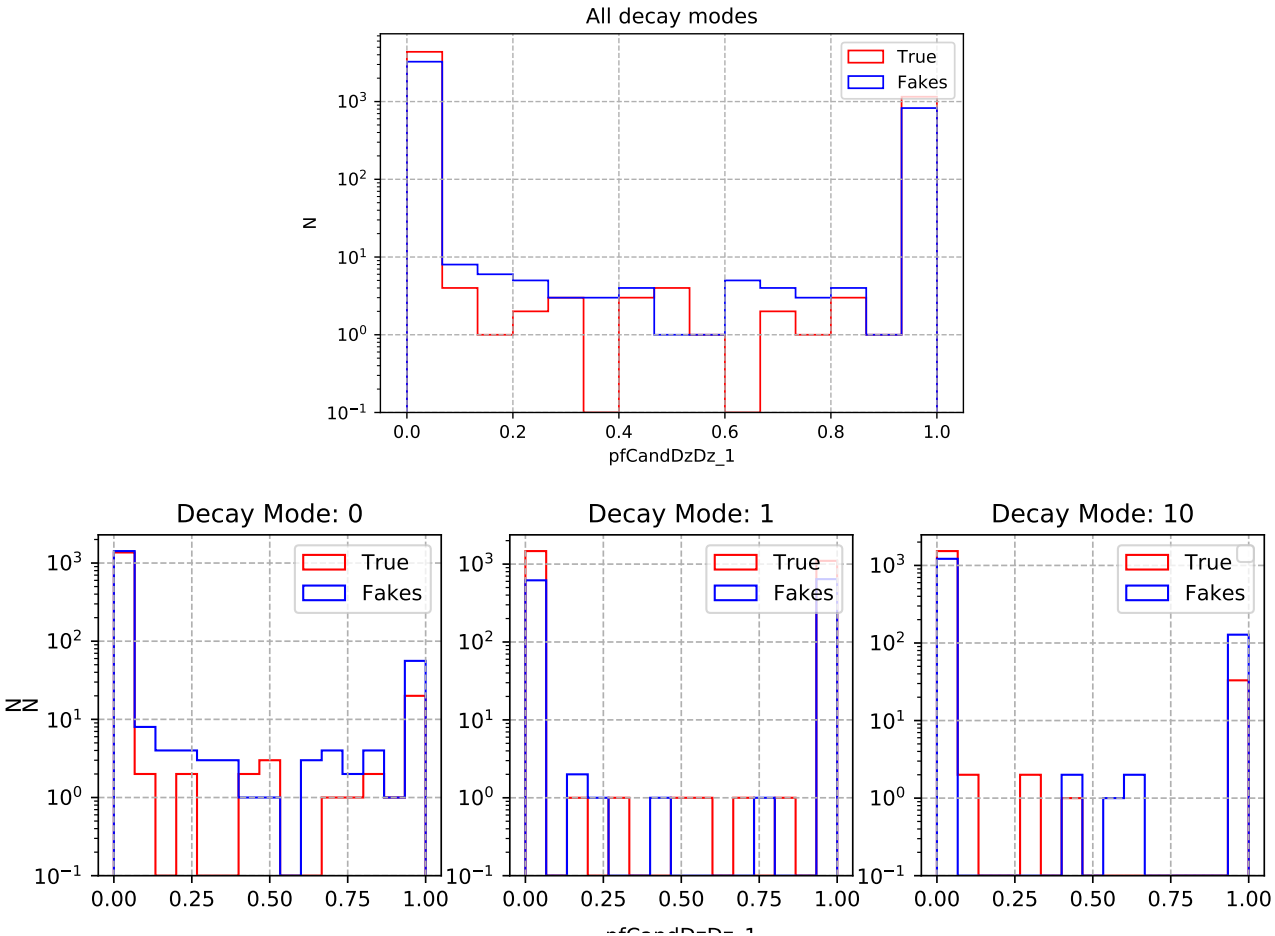
Feature: Multiplication of the 2 relative p_T mom



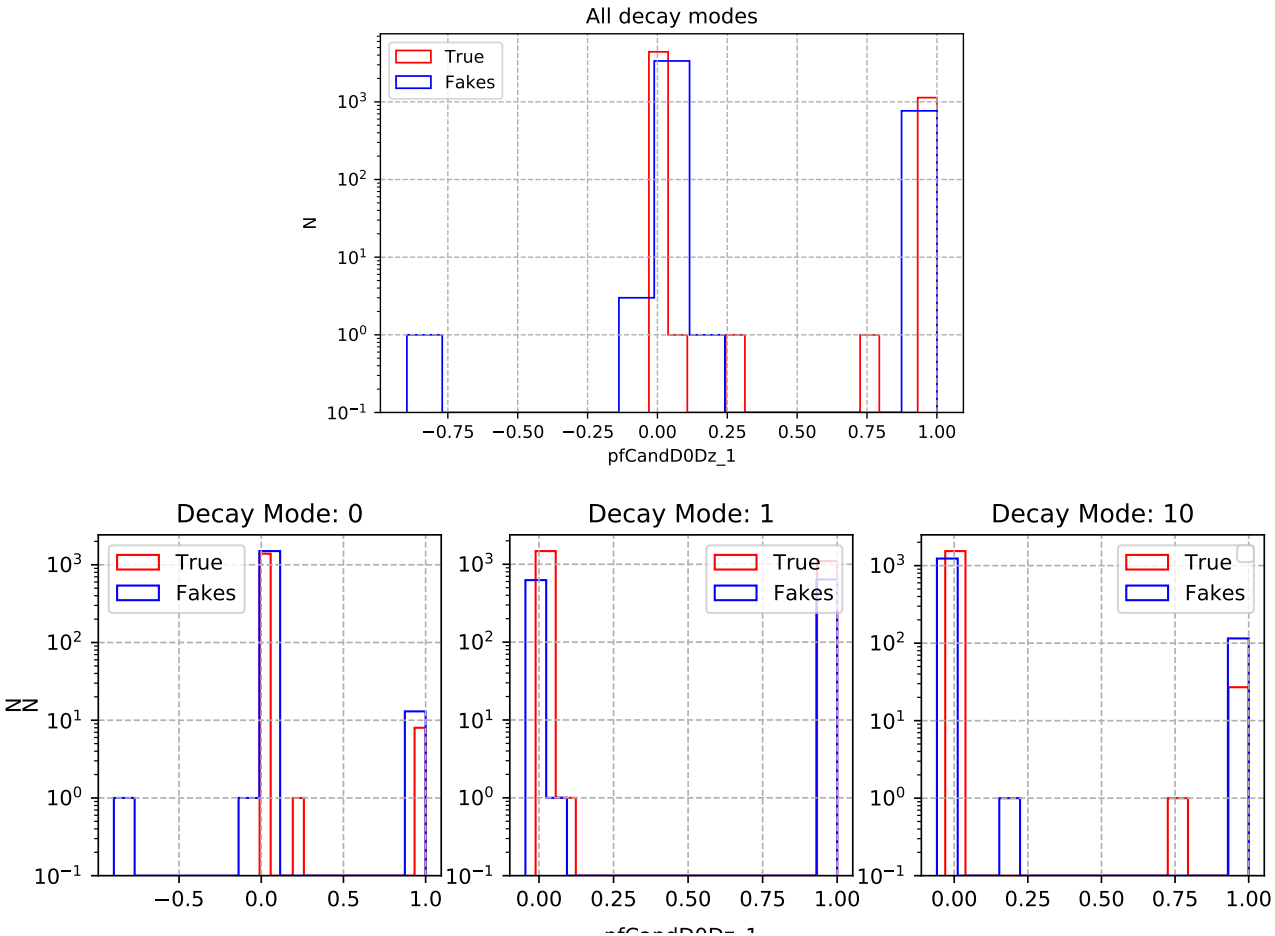
Feature: pfCandD0D0$_1$



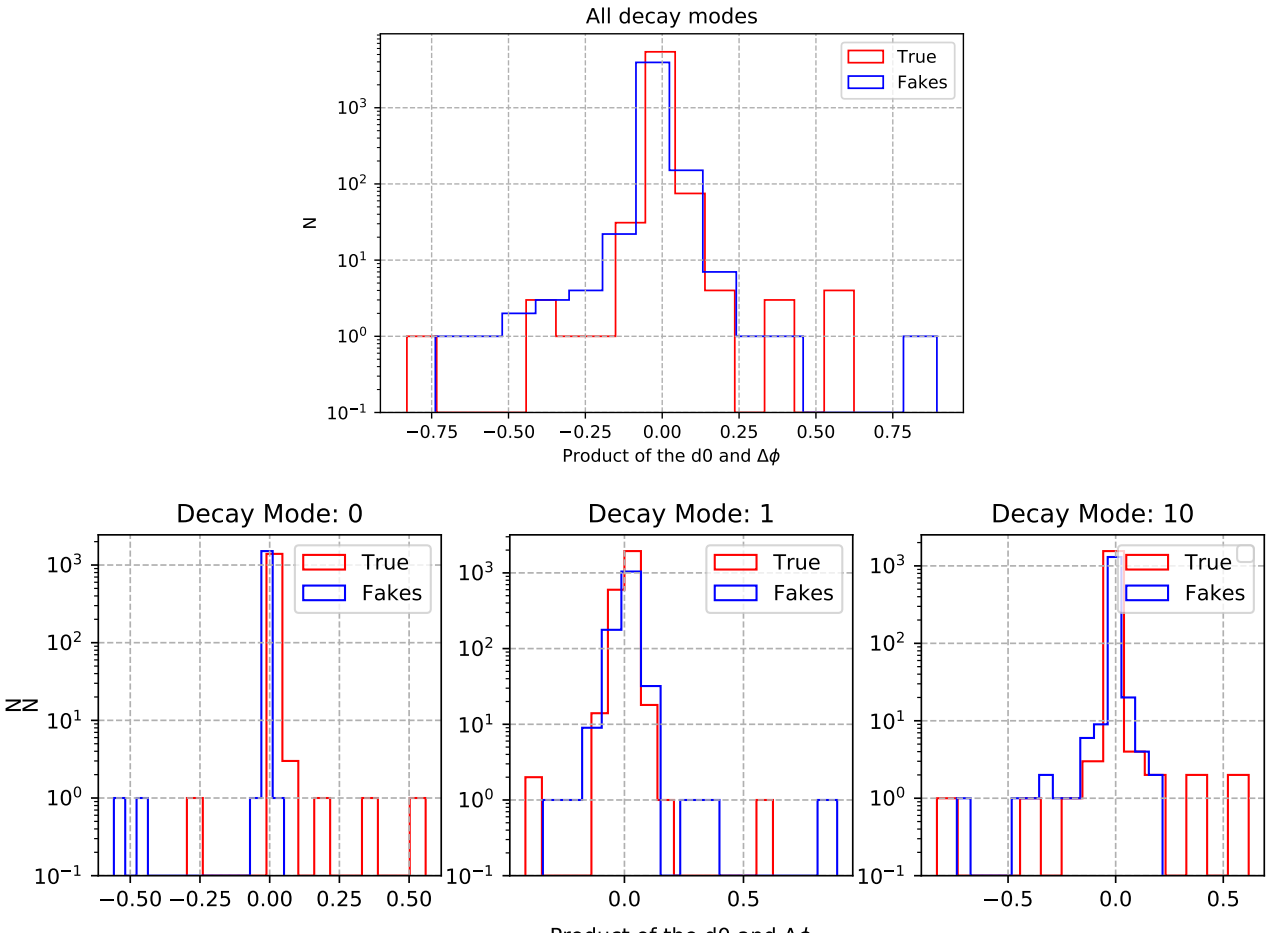
Feature: pfCandDzDz$_1$



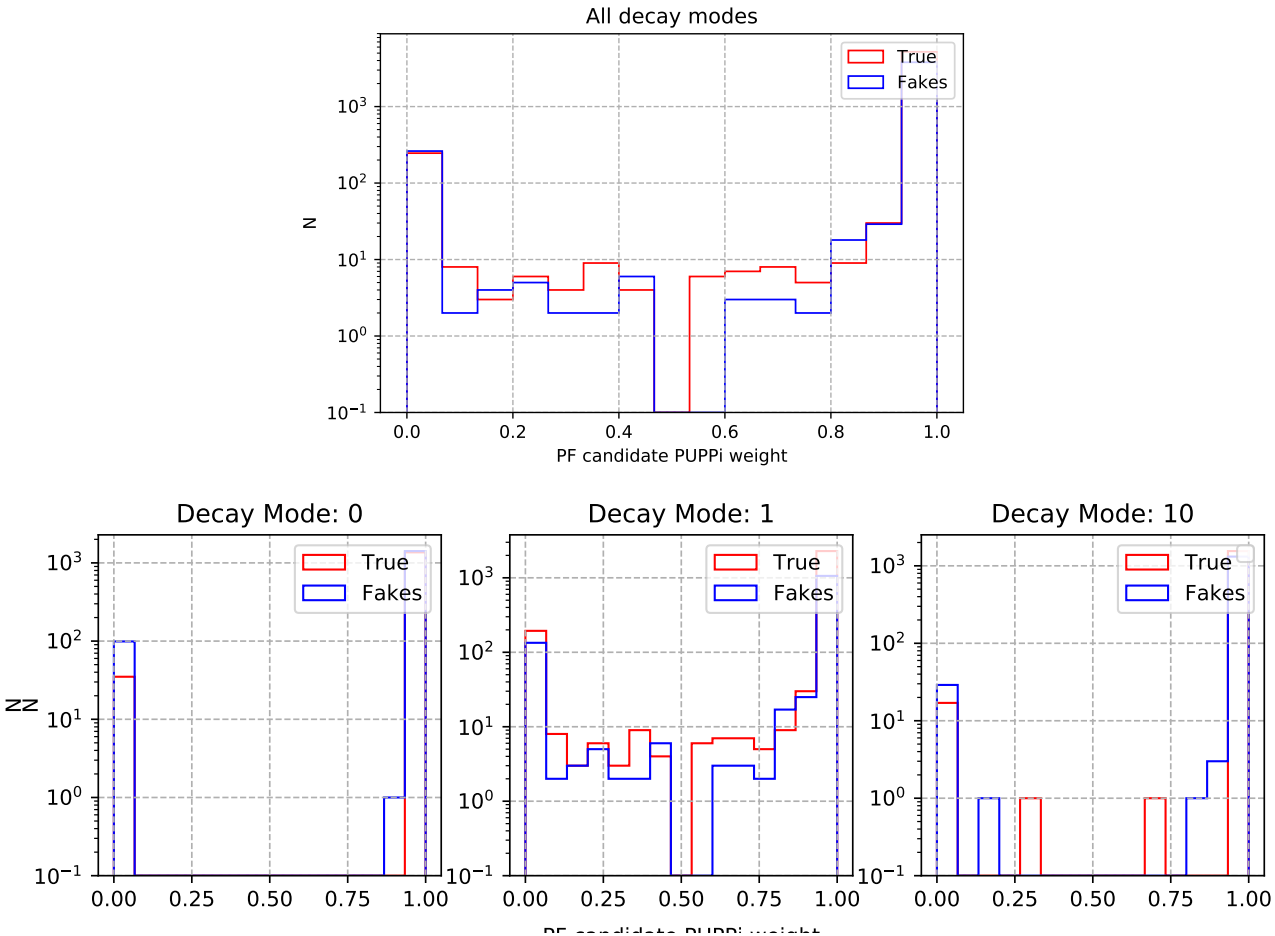
Feature: pfCandD0Dz$_1$



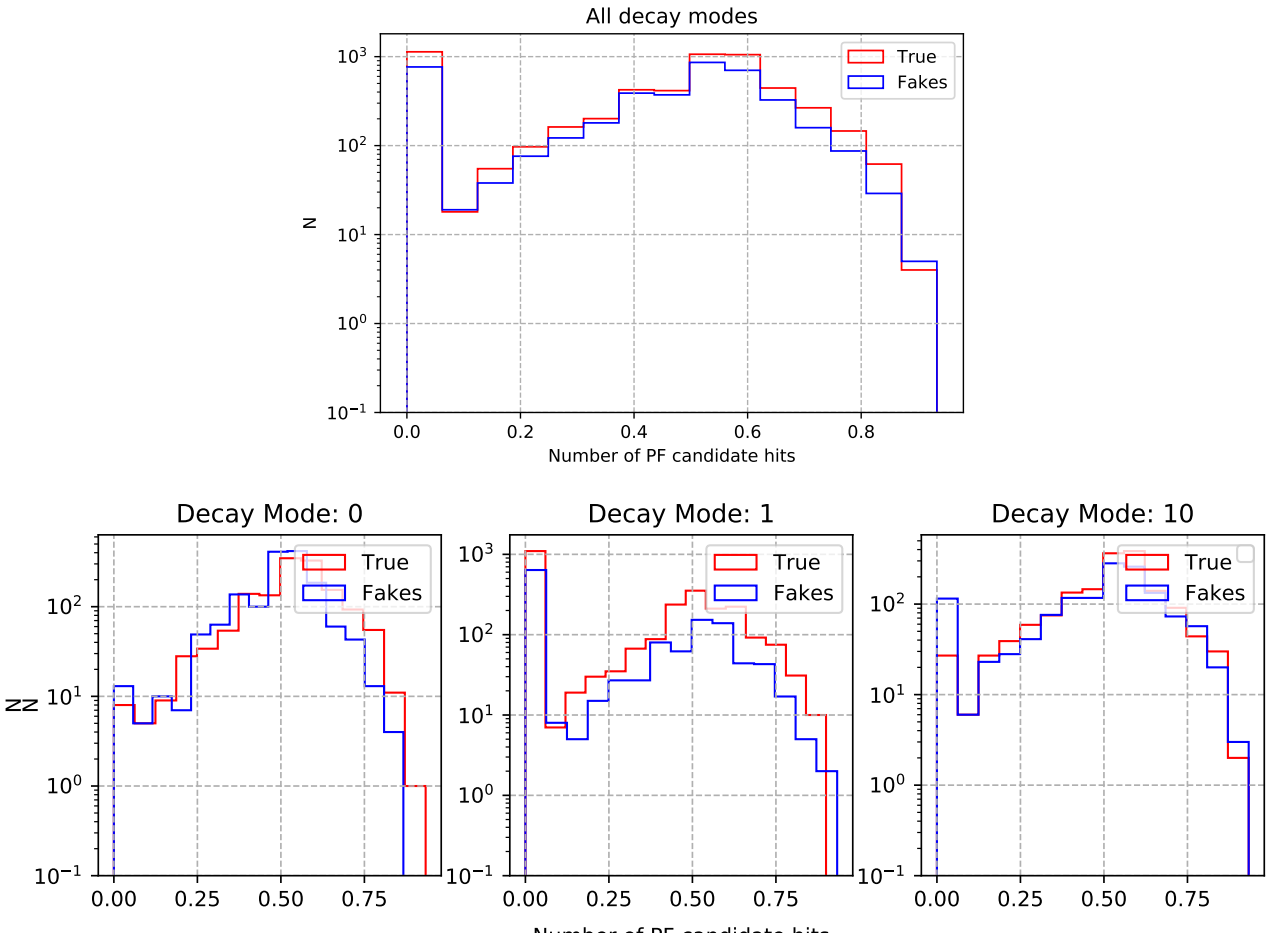
Feature: Product of the d0 and $\Delta\phi$



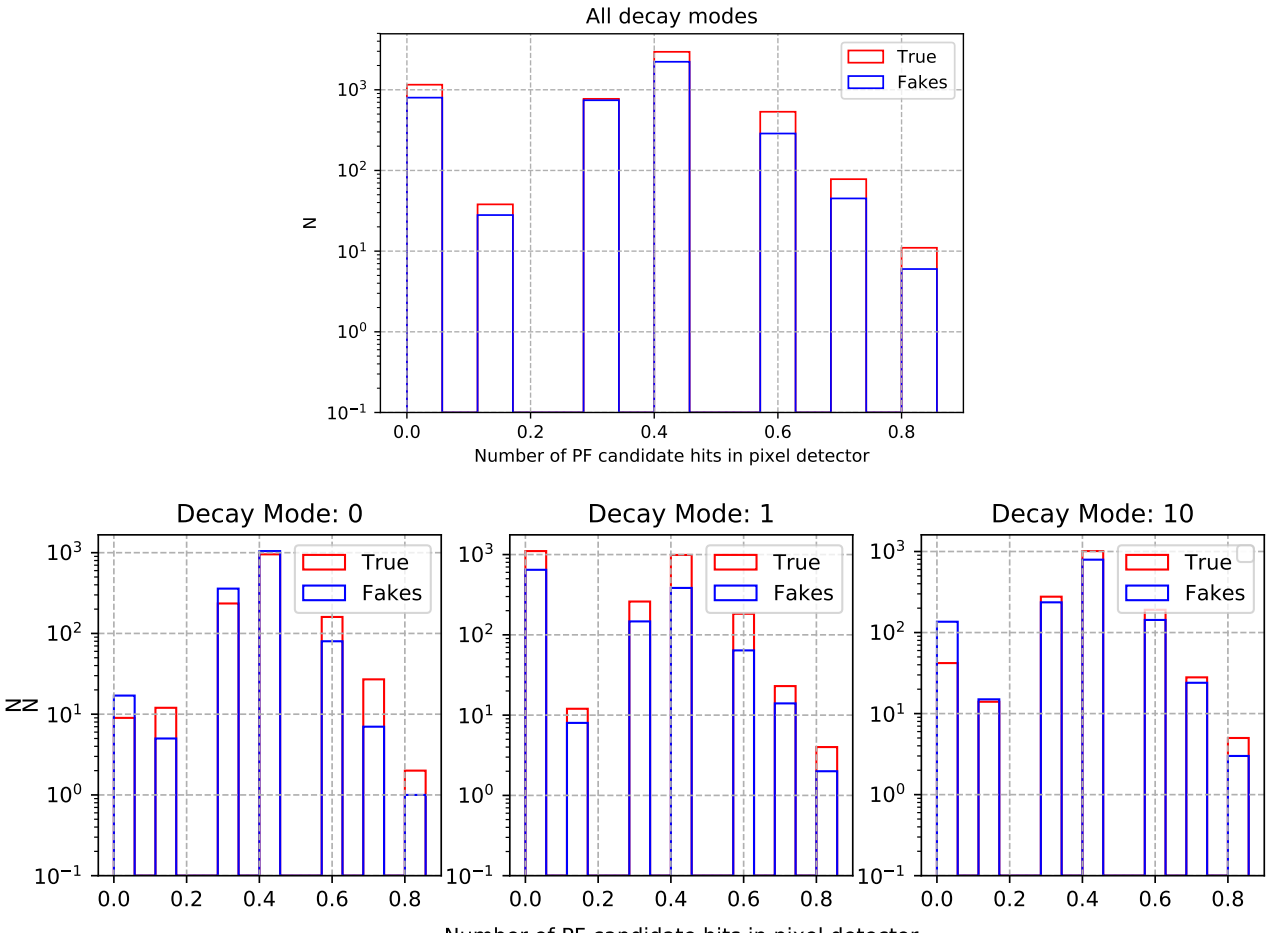
Feature: PF candidate PUPPi weight



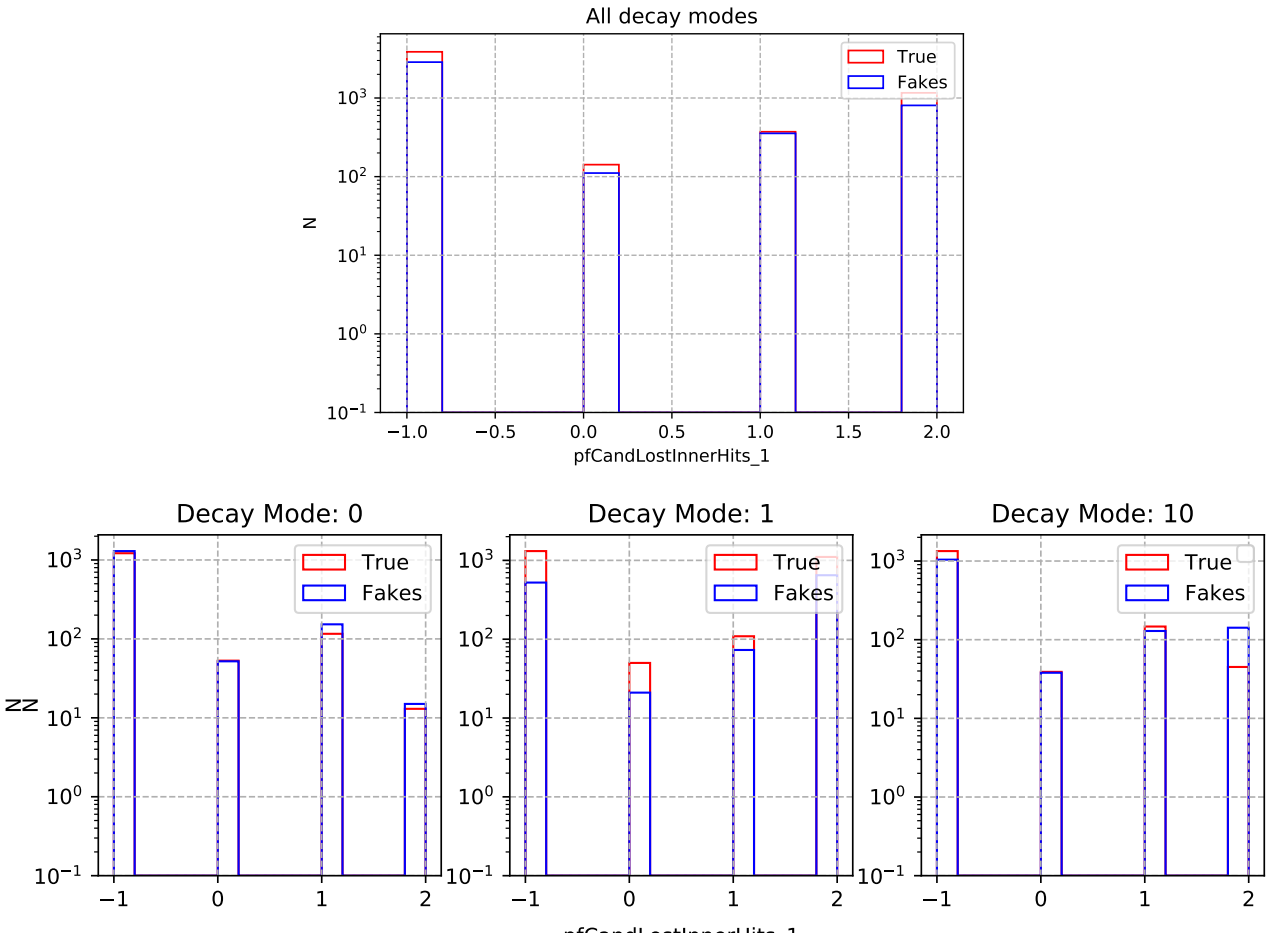
Feature: Number of PF candidate hits



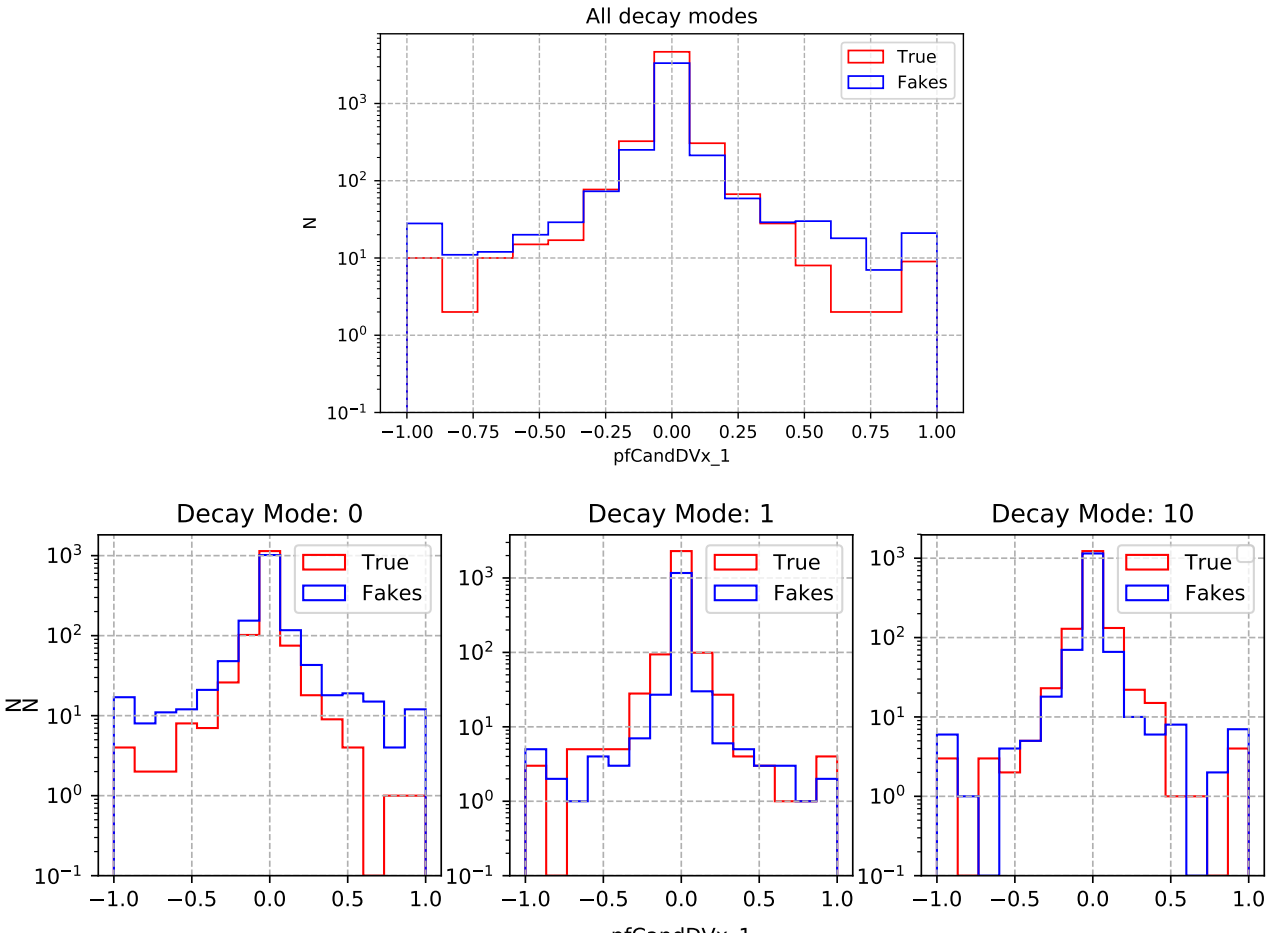
Feature: Number of PF candidate hits in pixel de



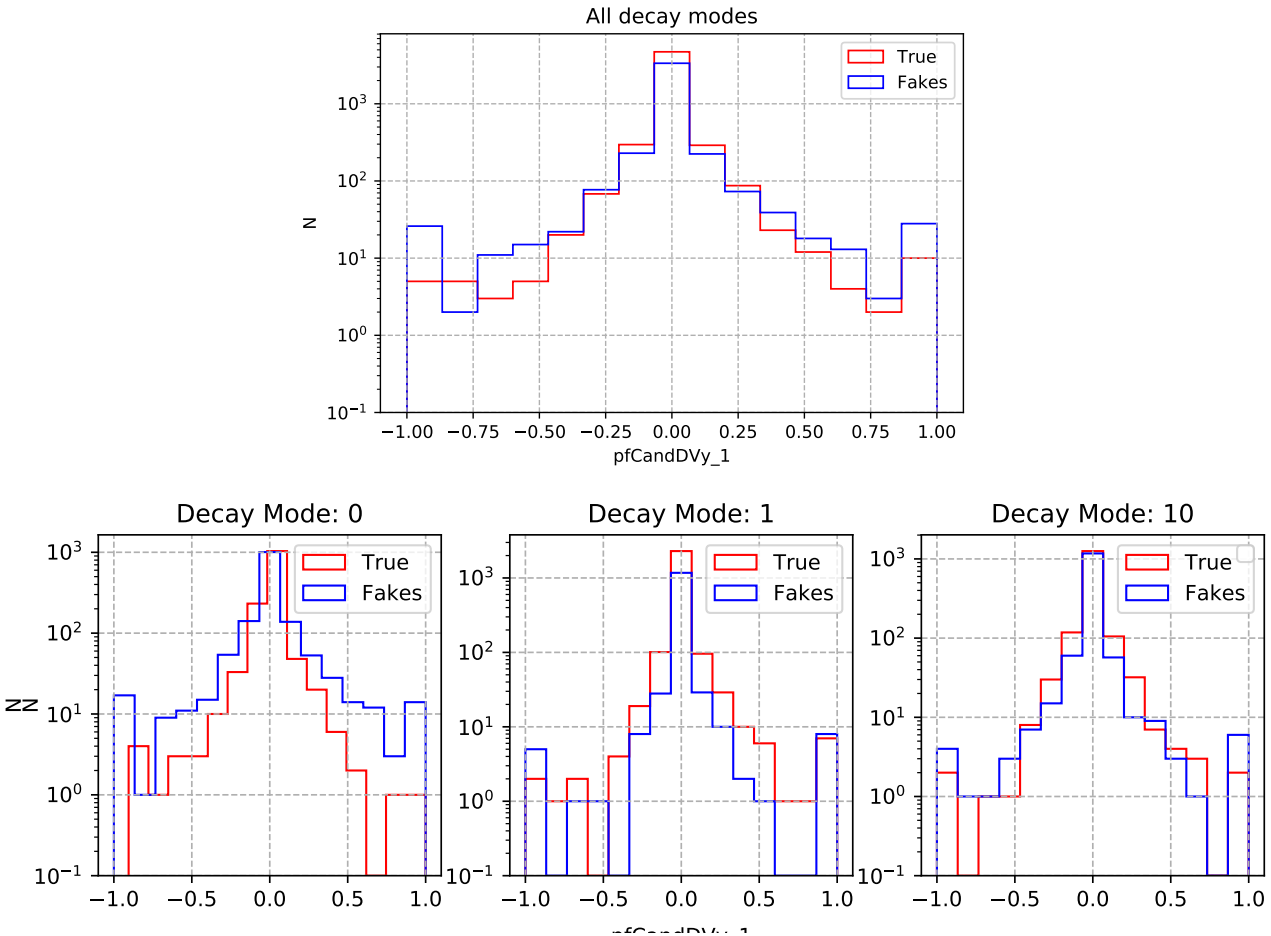
Feature: pfCandLostInnerHits$_1$



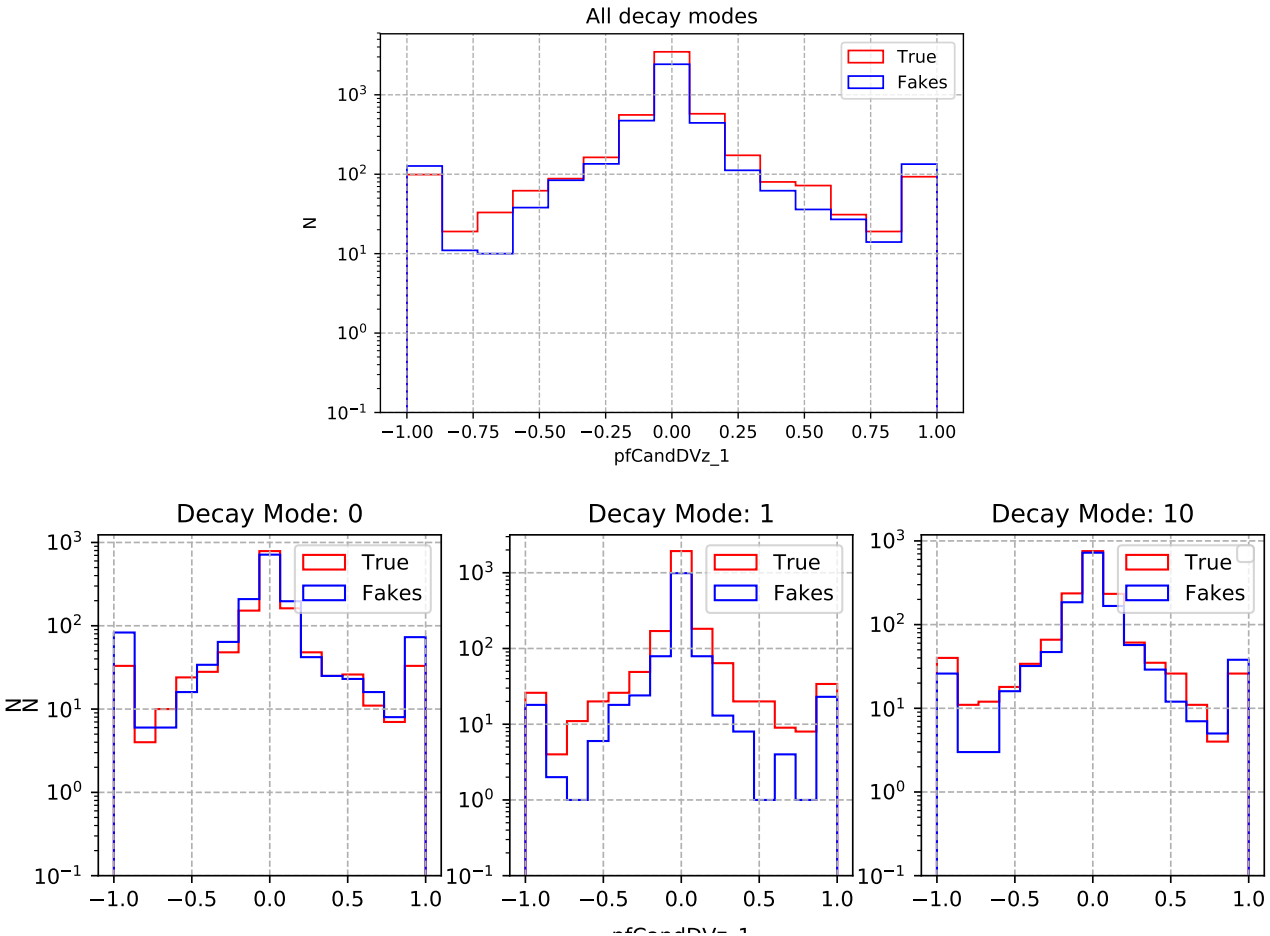
Feature: pfCandDVx$_1$



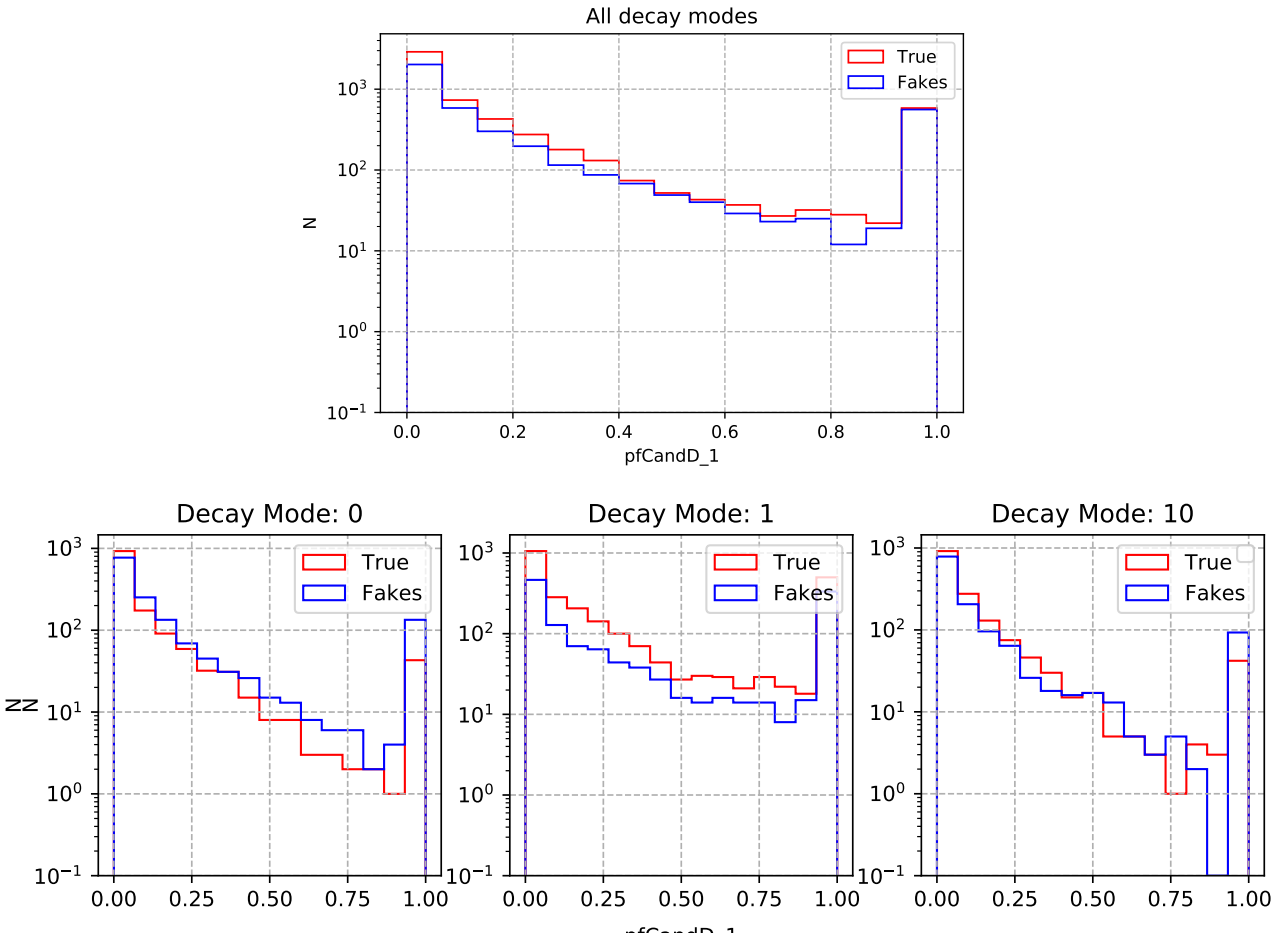
Feature: pfCandDVy_1



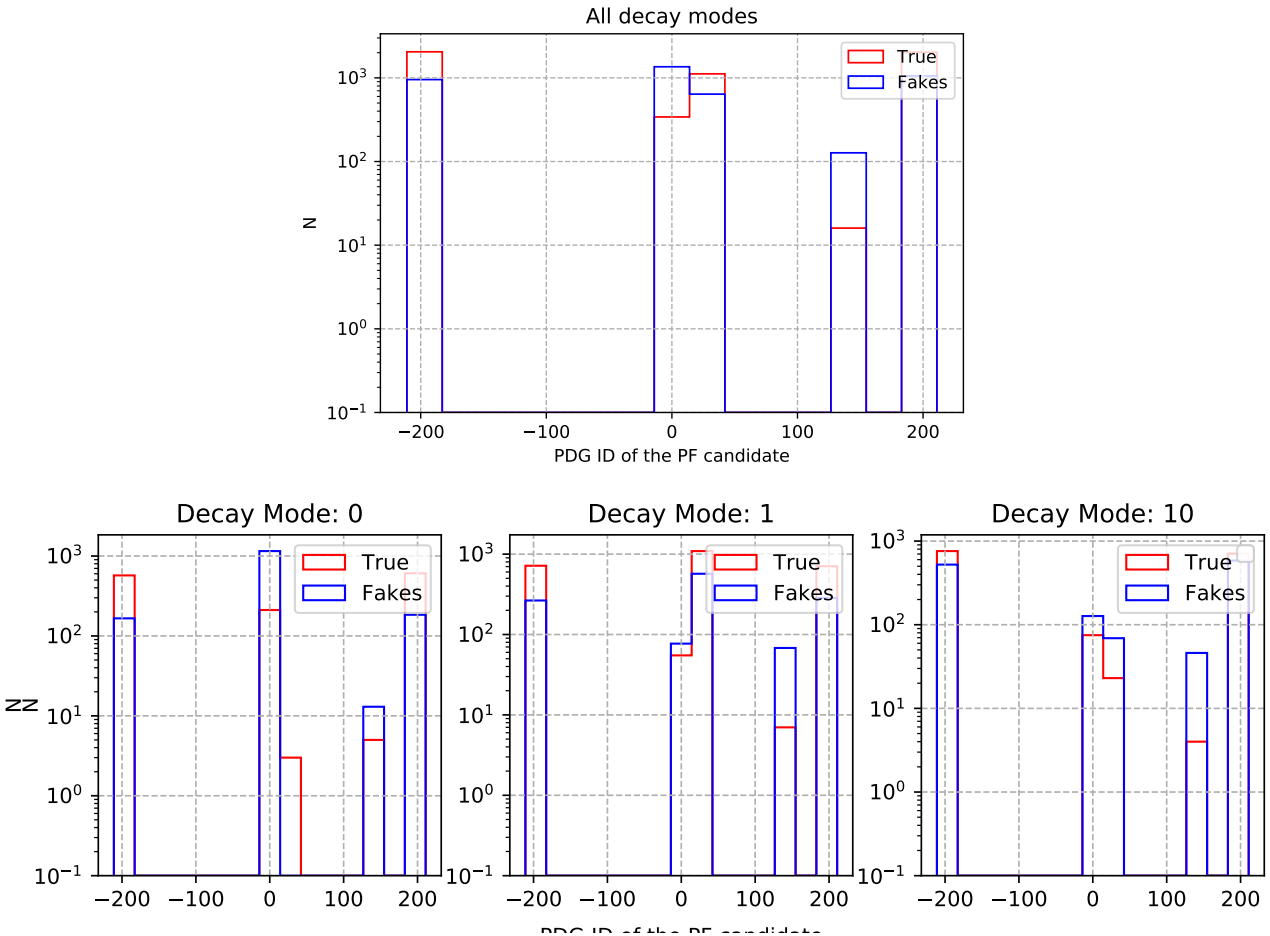
Feature: pfCandDVz$_1$



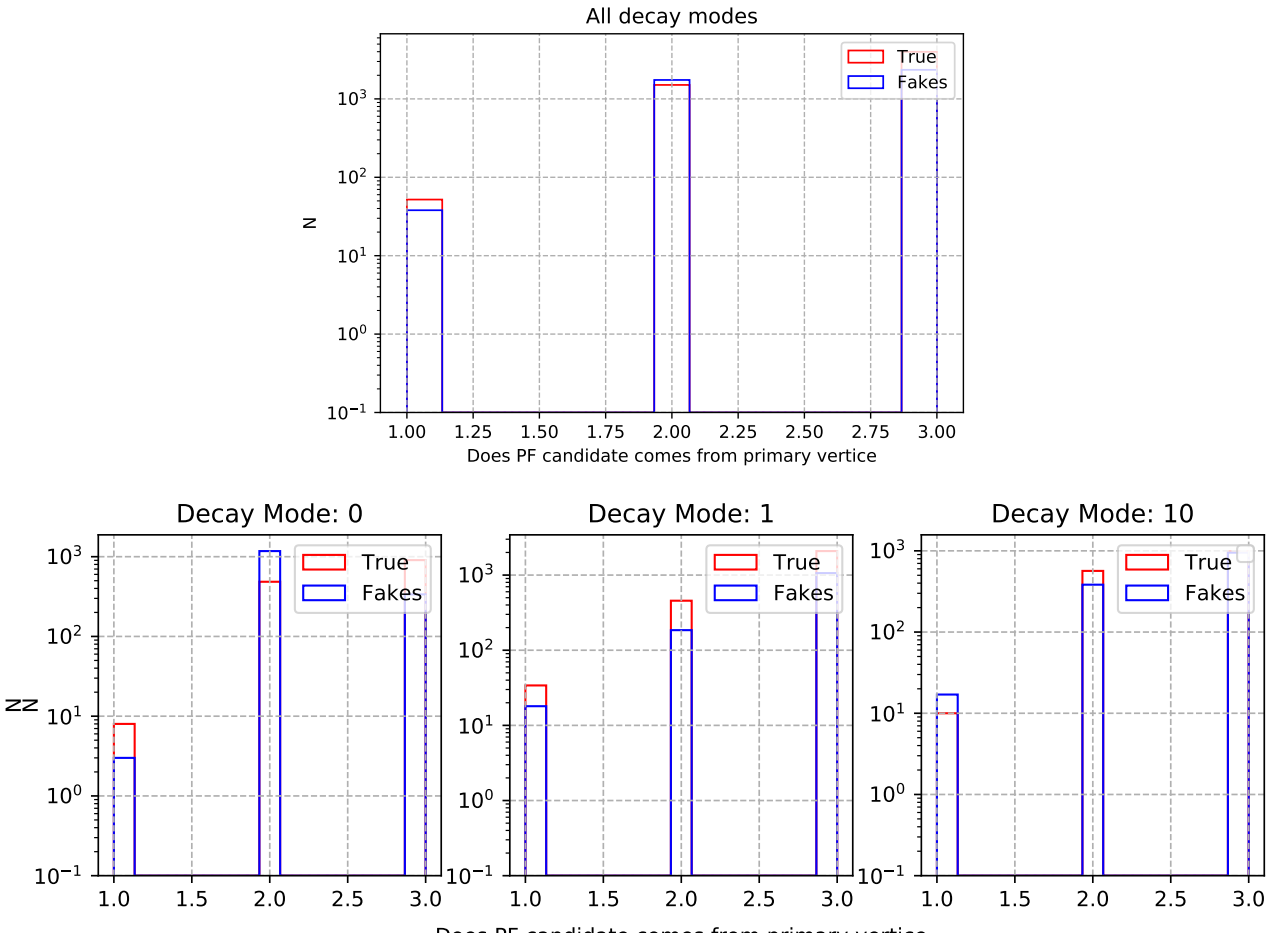
Feature: pfCandD$_1$



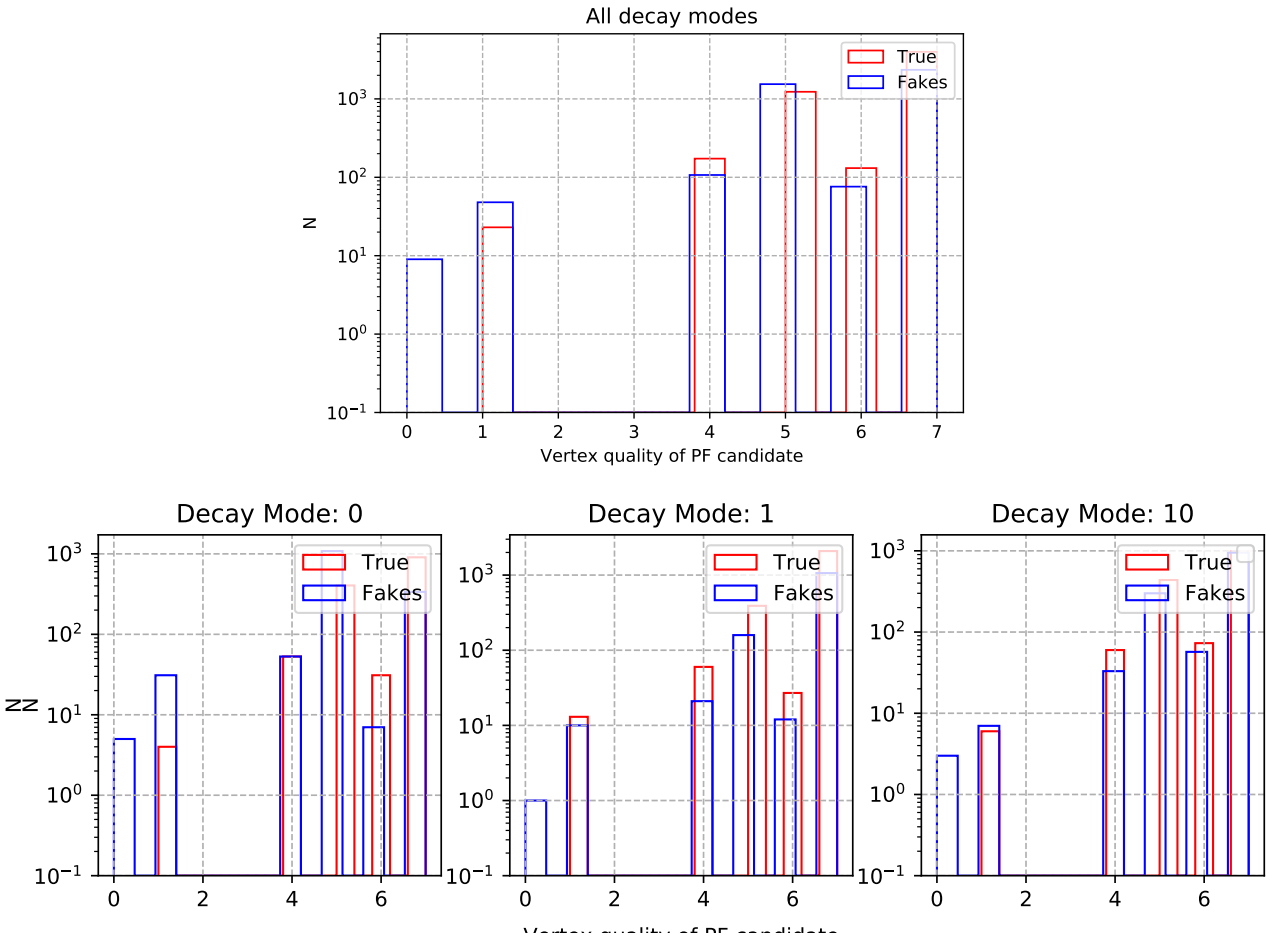
Feature: PDG ID of the PF candidate



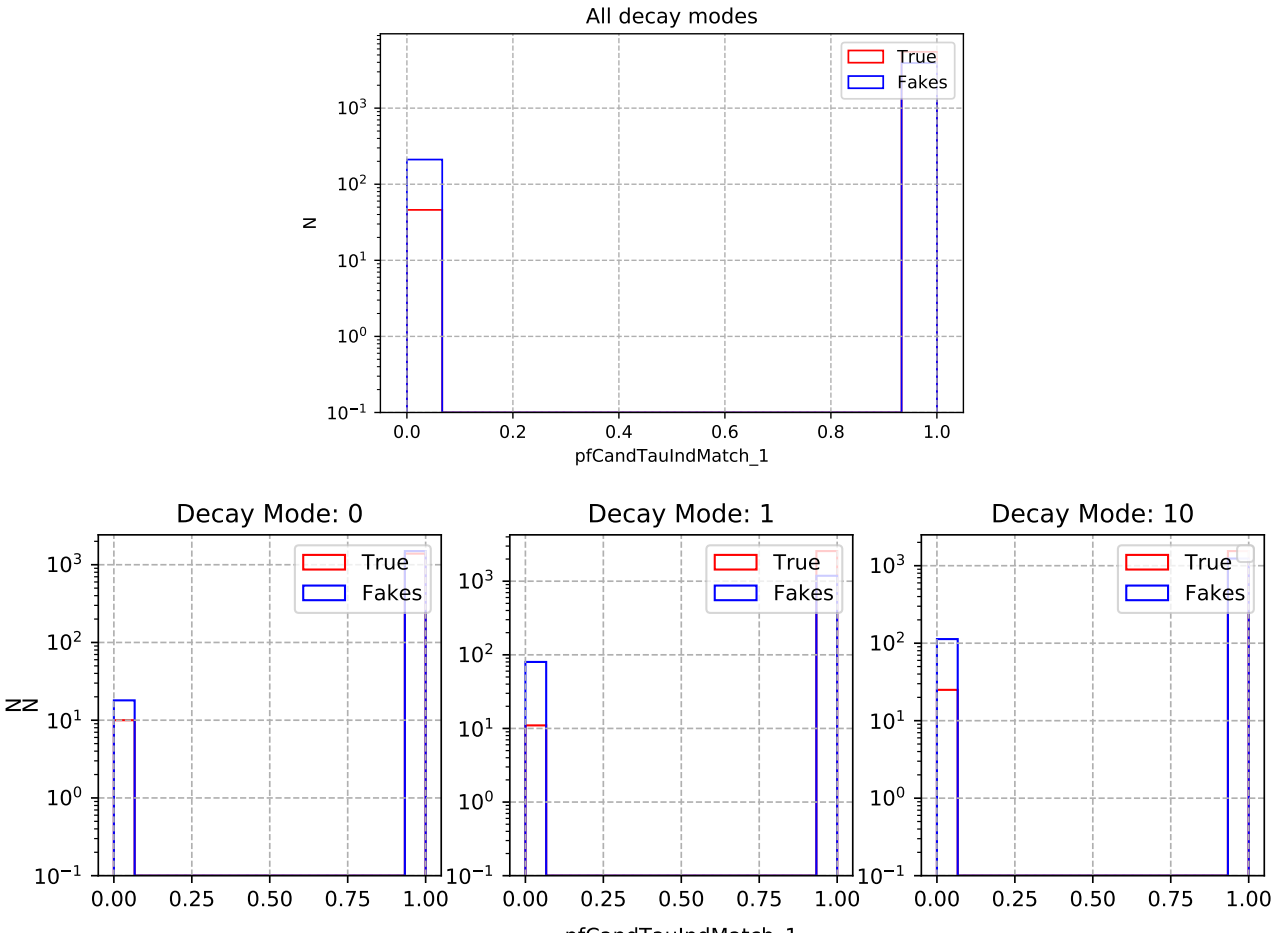
Feature: Does PF candidate comes from primary vertice



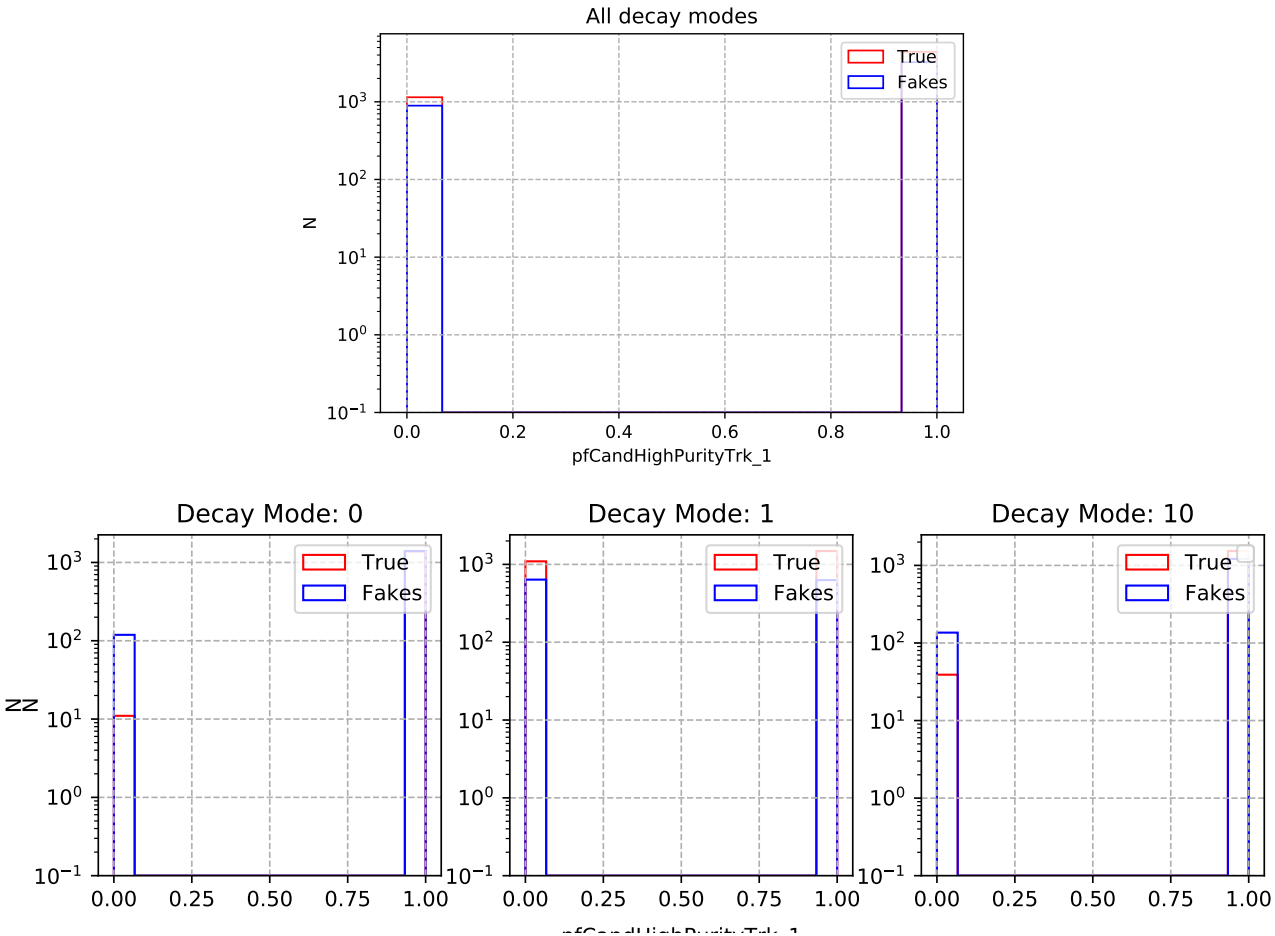
Feature: Vertex quality of PF candidate



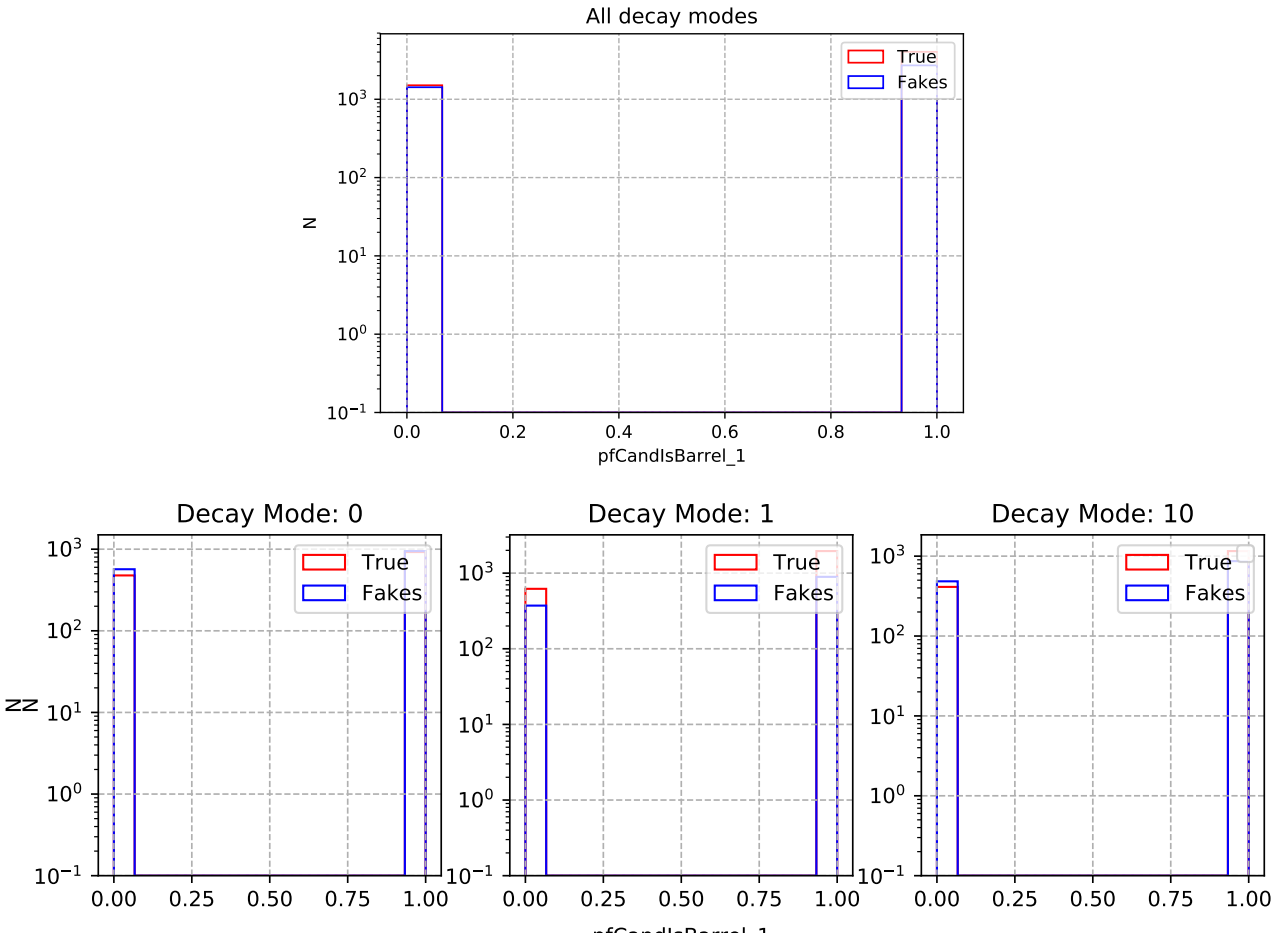
Feature: pfCandTauIndMatch$_1$



Feature: pfCandHighPurityTrk$_1$



Feature: pfCandIsBarrel$_1$



Feature: lepHasSV$_1$