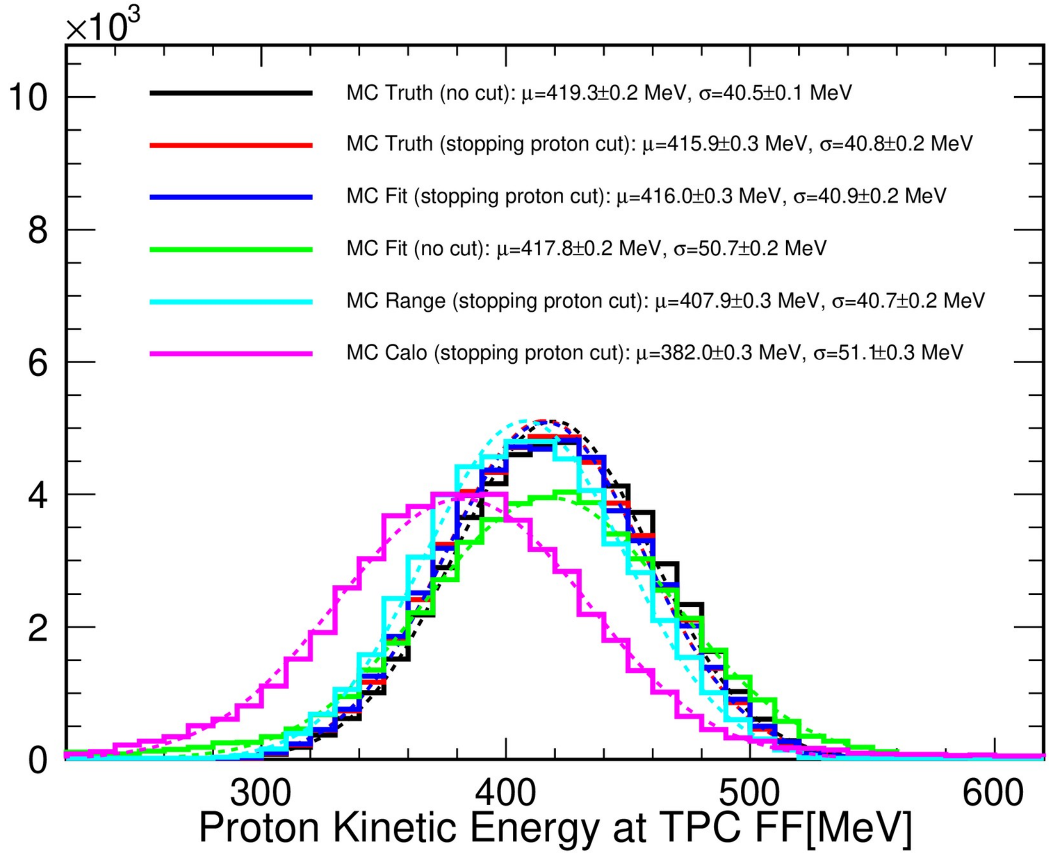
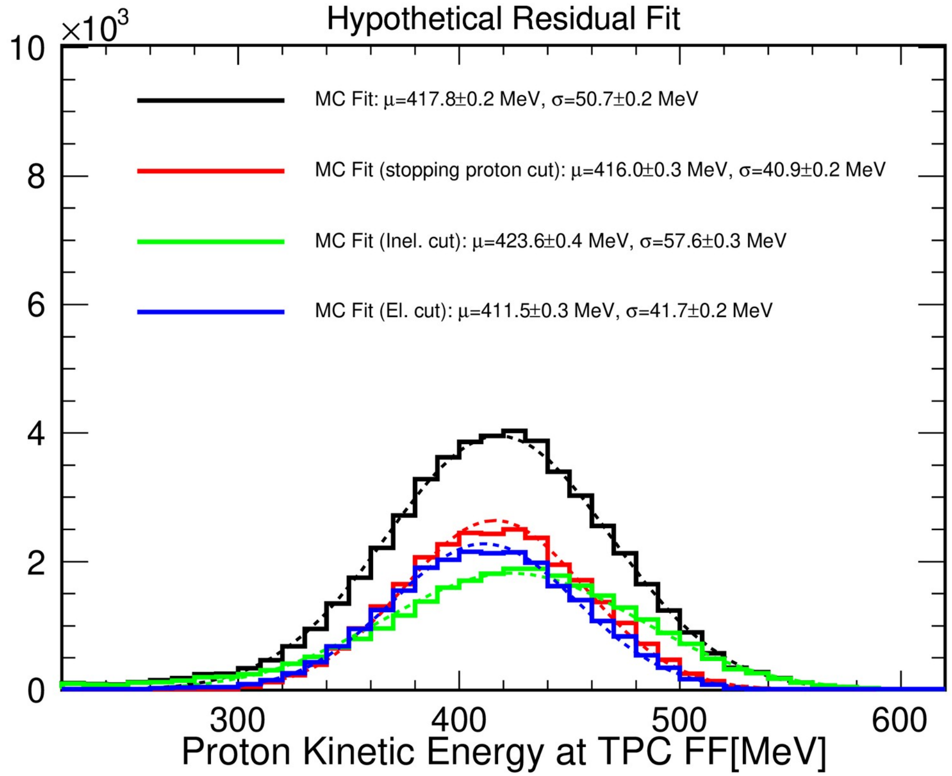
1 GeV/c Proton-argon Inelastic Cross-section Update

- Update on KE systematics
- Study of improving inelastic event selection

Heng-Ye Liao
ProtoDUNE hadron-argon XS measurements
August 18, 2022

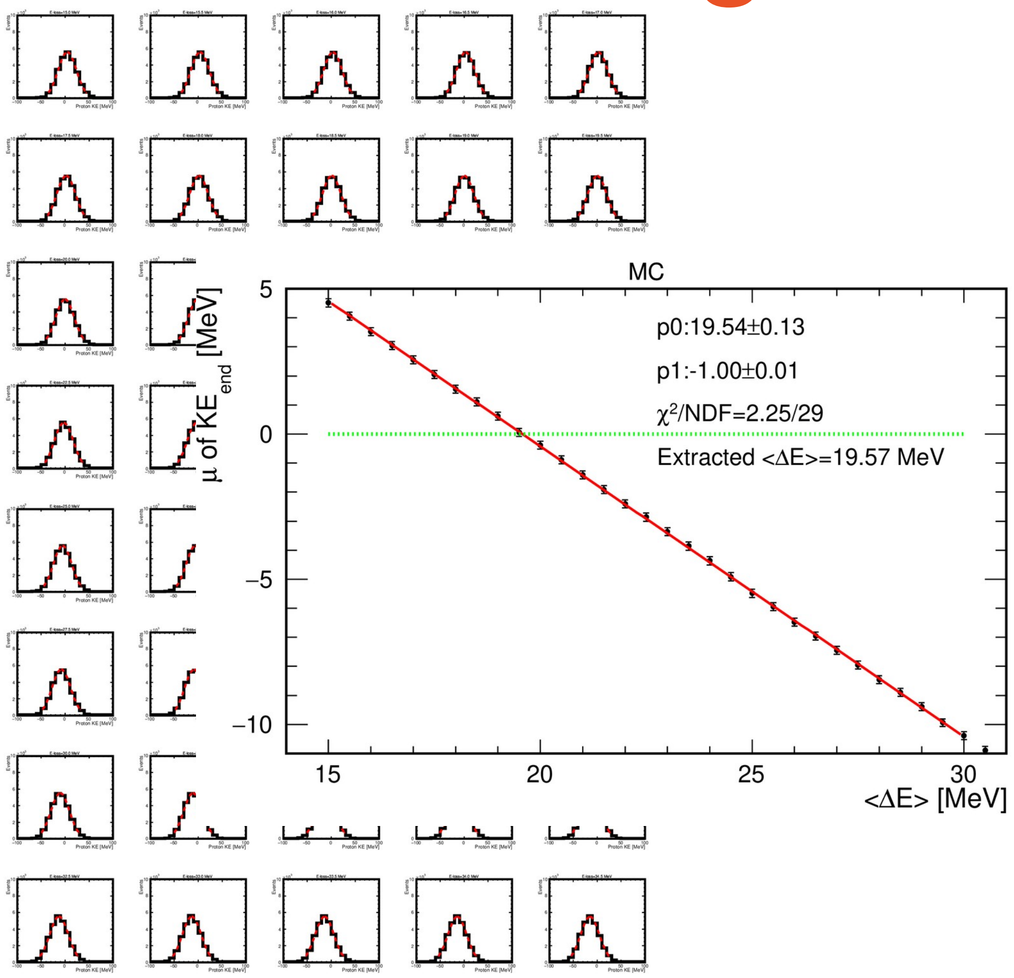
KEff: Summary





- KE(truth) is the same before/after stopping proton cut
- Compare fit, range, calo method: Fit method is the best that can represent truth energy
- Wider distribution of fit method without stopping proton cut (because of inel. component)

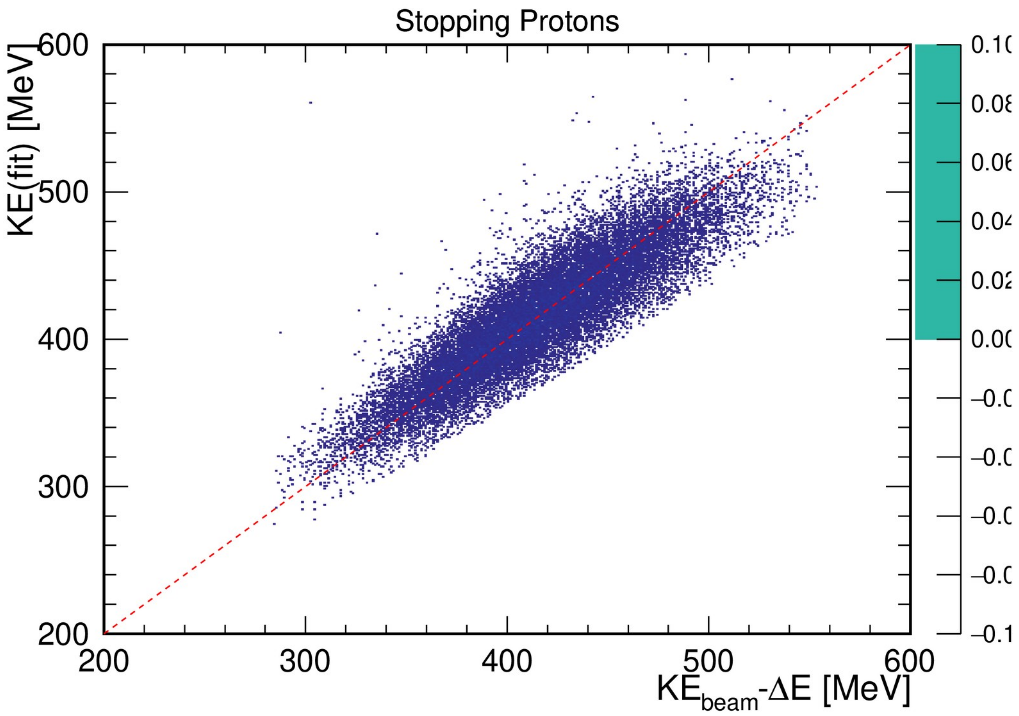
E-loss using scanning method: Summary

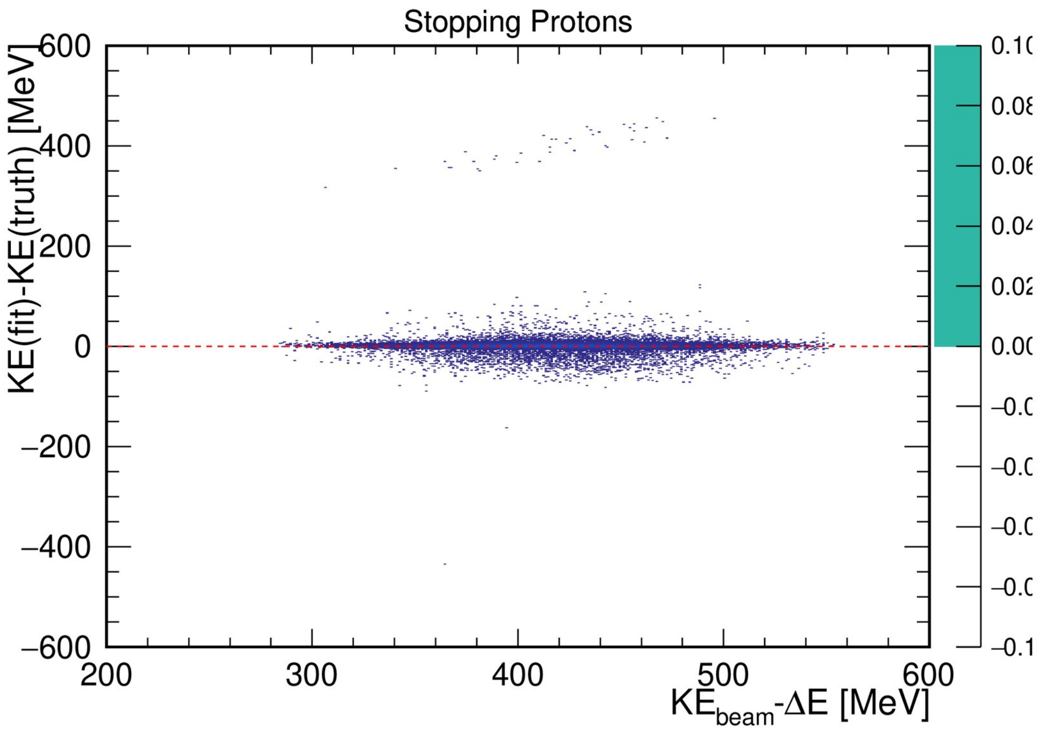


Method	E-loss [MeV]	
	Data	MC
Fit	24.6	17.2
Fit (stop)	25.2	19.6
Range	29.8	27.1
Range (stop)	29.7	26.7
Calo	45.6	49.3
Calo (stop)	45.4	48.7

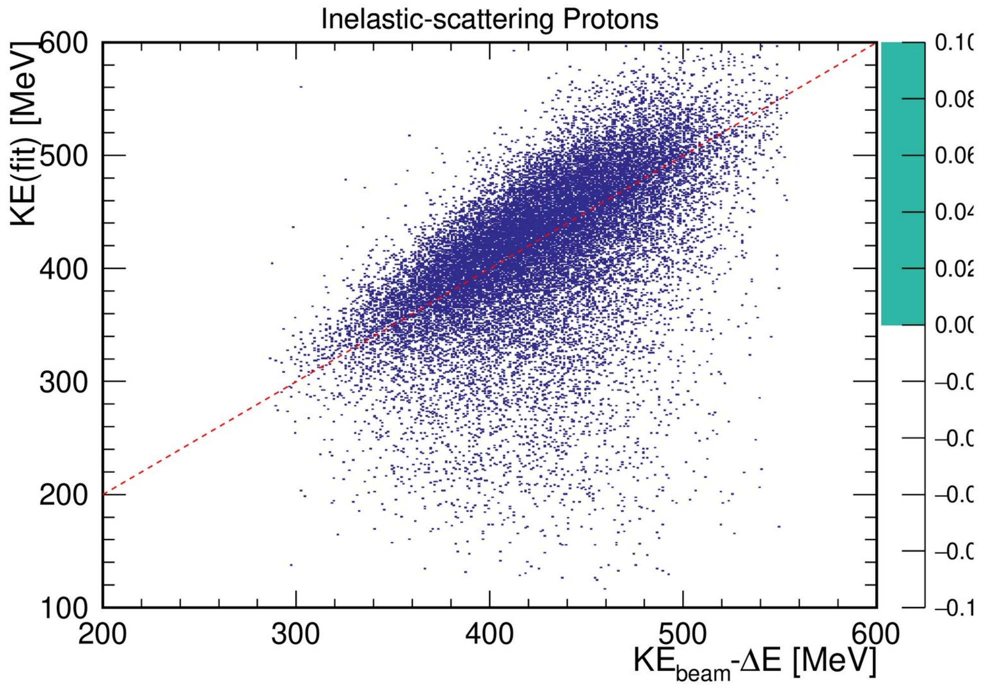
- Use fit (stop) to determine E-loss

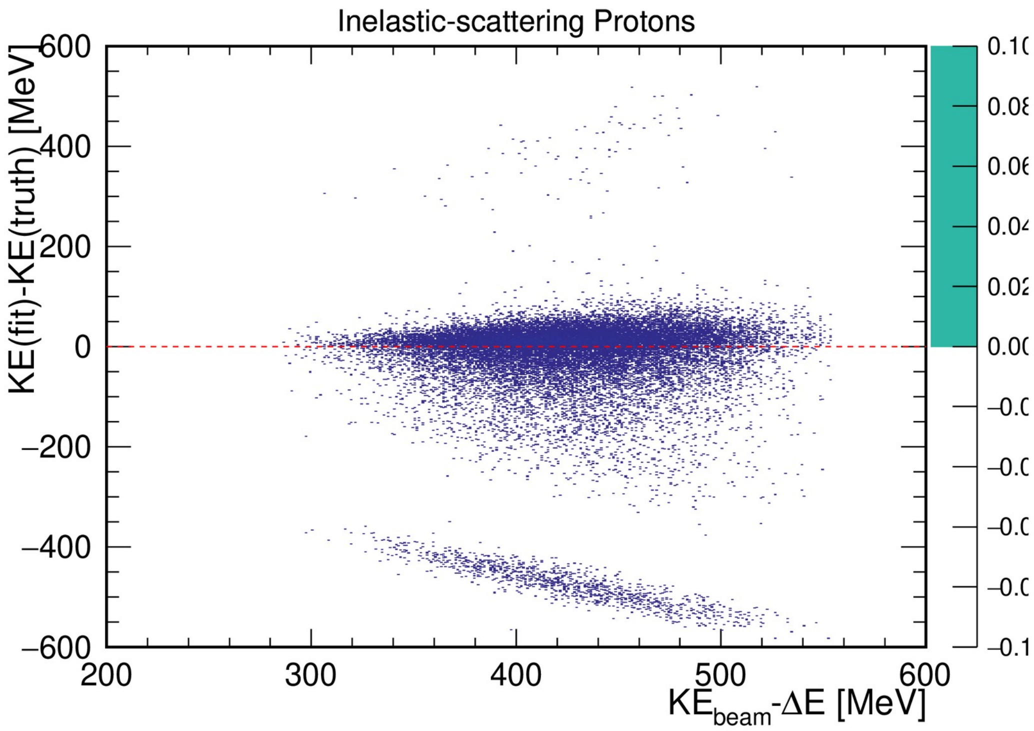
KE_{ff}(reco) v.s. KE_{ff}(truth): Stopping Protons





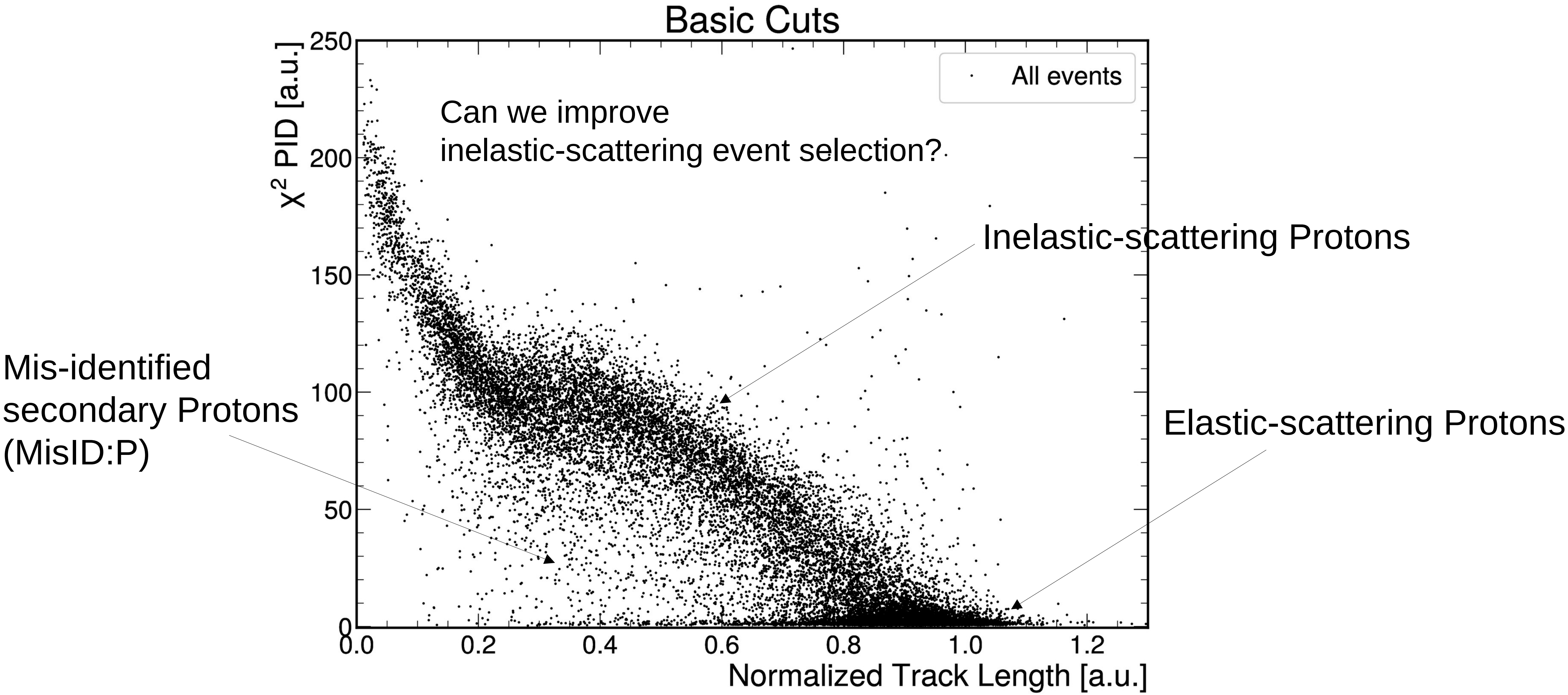
KEff(reco) v.s. KEff(truth): Inelastic-scattering Protons





- Can we make event-by-event correction at KEff, instead of reweighting?

Inelastic-scattering Proton Event Selection



Feature Observables

- 9 features used in total:
  (1) PID: Chi^2 PID
  (2) ntrklen: Normalized track length
  (3) B: Impact parameter
  (3D distance between endpoint to the projected line fitted using the first 3 hits)
  (4) trklen: track length
  (5) calo: Σ(dE/dx*dx)
  (6) mediandedx: Median dE/dx
  (7) avcalo: Σ(dE/dx*dx)/track length (energy loss per distance)
  (8) endpointdedx: Averaged dE/dx of the last 3 hits
  (9) costheta: Angle between beam and TPC track

Inelastic Event Selection using XGBoost

- XGBoost: eXtreme Gradient Boosted trees (2016)
- Software package: https://xgboost.readthedocs.io/en/stable/

XGBoost: A Scalable Tree Boosting System

Tianqi Chen
University of Washington
tqchen@cs.washington.edu

Carlos Guestrin
University of Washington
guestrin@cs.washington.edu

ABSTRACT

Tree boosting is a highly effective and widely used machine learning method. In this paper, we describe a scalable end-to-end tree boosting system called XGBoost, which is used widely by data scientists to achieve state-of-the-art results on many machine learning challenges. We propose a novel sparsity-aware algorithm for sparse data and weighted quantile sketch for approximate tree learning. More importantly, we provide insights on cache access patterns, data compression and sharding to build a scalable tree boosting system. By combining these insights, XGBoost scales beyond billions of examples using far fewer resources than existing systems.

Keywords

Large-scale Machine Learning

problems. Besides being used as a stand-alone predictor, it is also incorporated into real-world production pipelines for ad click through rate prediction [15]. Finally, it is the de-facto choice of ensemble method and is used in challenges such as the Netflix prize [3].

In this paper, we describe XGBoost, a scalable machine learning system for tree boosting. The system is available as an open source package[2]. The impact of the system has been widely recognized in a number of machine learning and data mining challenges. Take the challenges hosted by the machine learning competition site Kaggle for example. Among the 29 challenge winning solutions [3] published at Kaggle's blog during 2015, 17 solutions used XGBoost. Among these solutions, eight solely used XGBoost to train the model, while most others combined XGBoost with neural nets in ensembles. For comparison, the second most popular

https://dl.acm.org/doi/pdf/10.1145/2939672.2939785

Question: Does the person like computer games?
Inputs: age, gender, occupation (i.e. features)

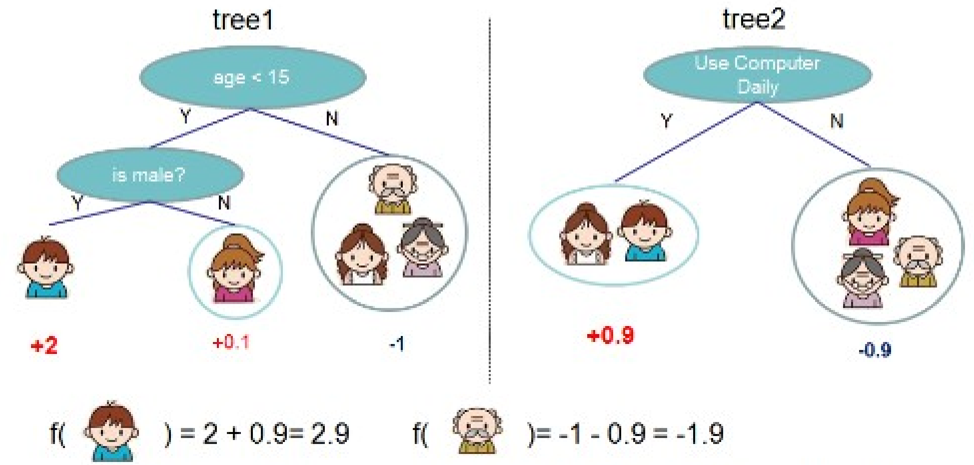
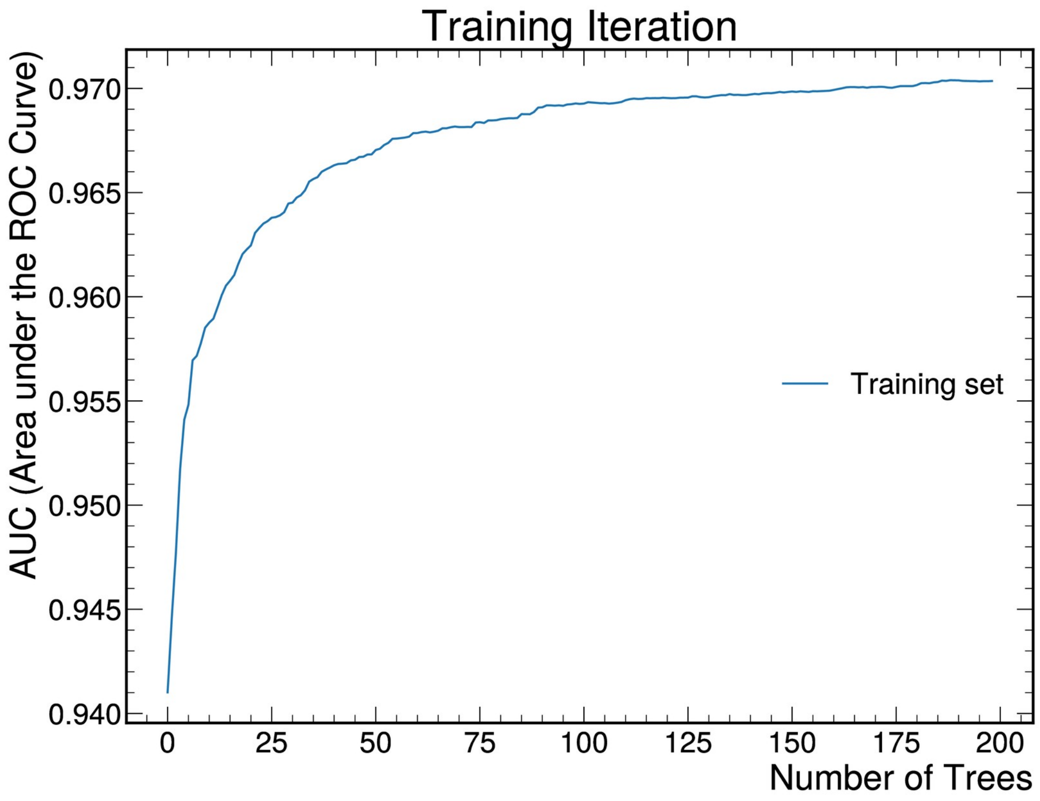


Figure 1: Tree Ensemble Model. The final prediction for a given example is the sum of predictions from each tree.

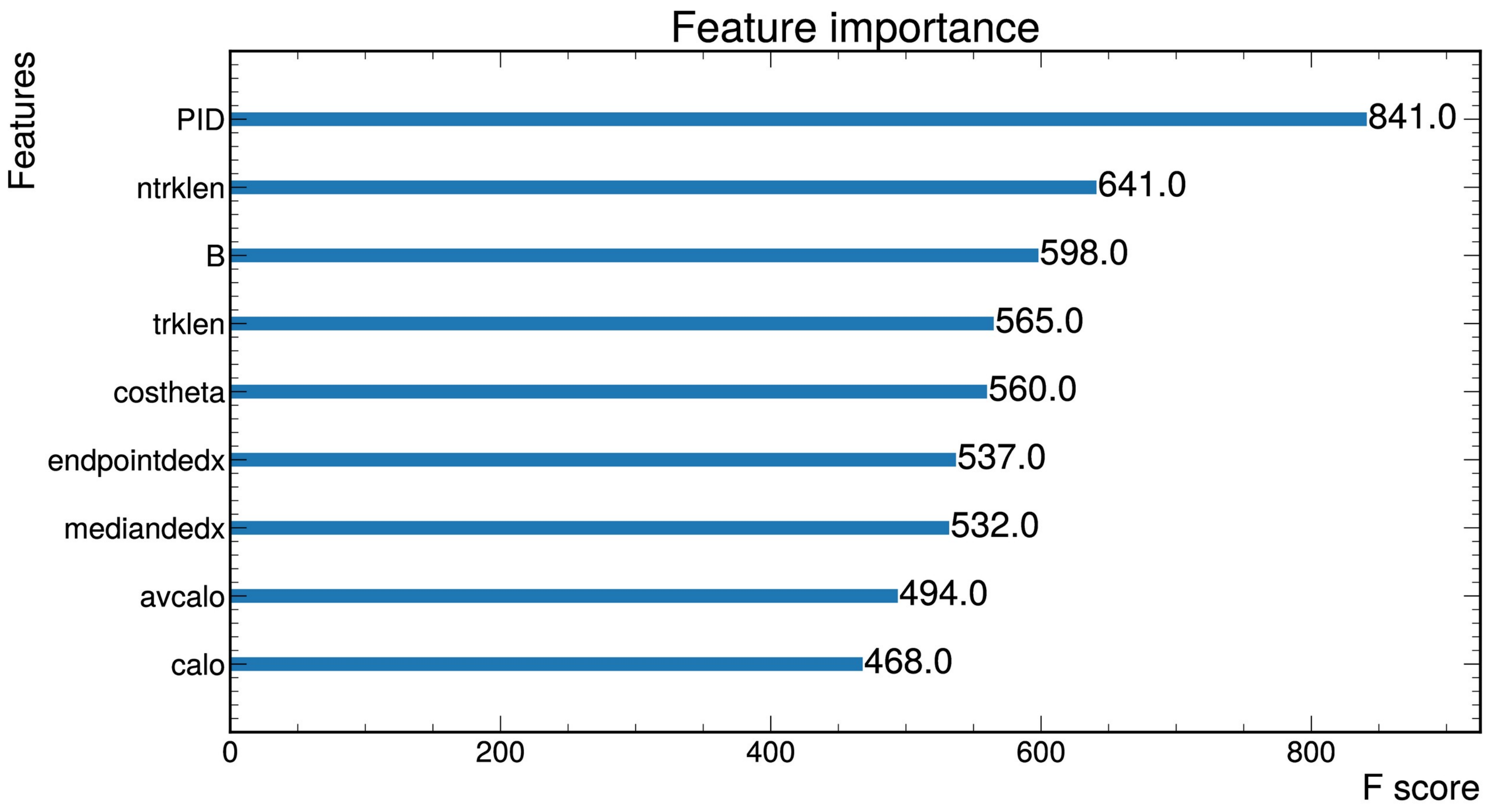
XGBoost: Training Process



Single XGBoost Decision Tree

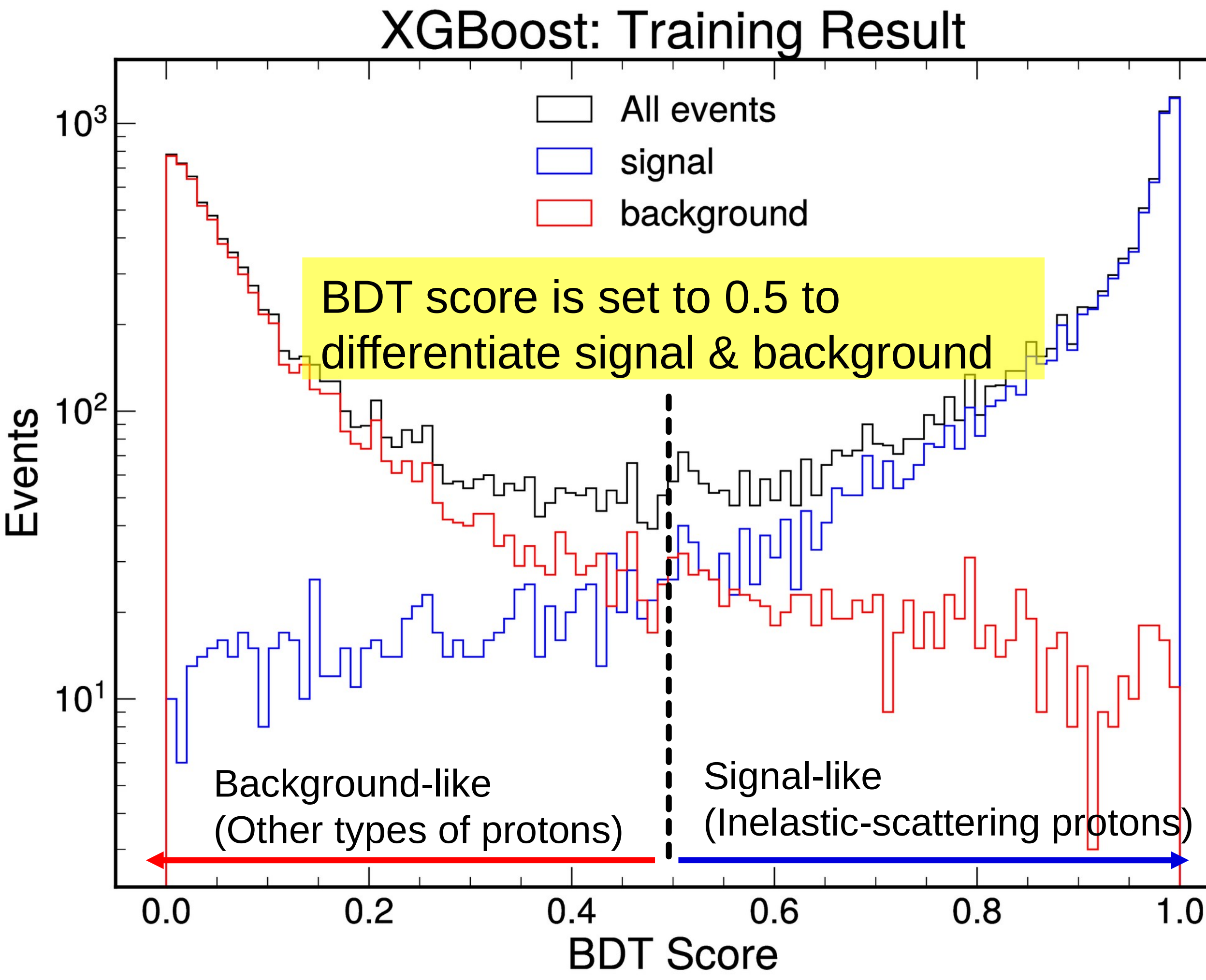
- MC: 60% used for training; 40% for cross-validation
- AUC(Area under ROC) is used for evaluation of “distance” between reco and truth
- Less than 40 sec processing time using prebuilt model

Feature Importance



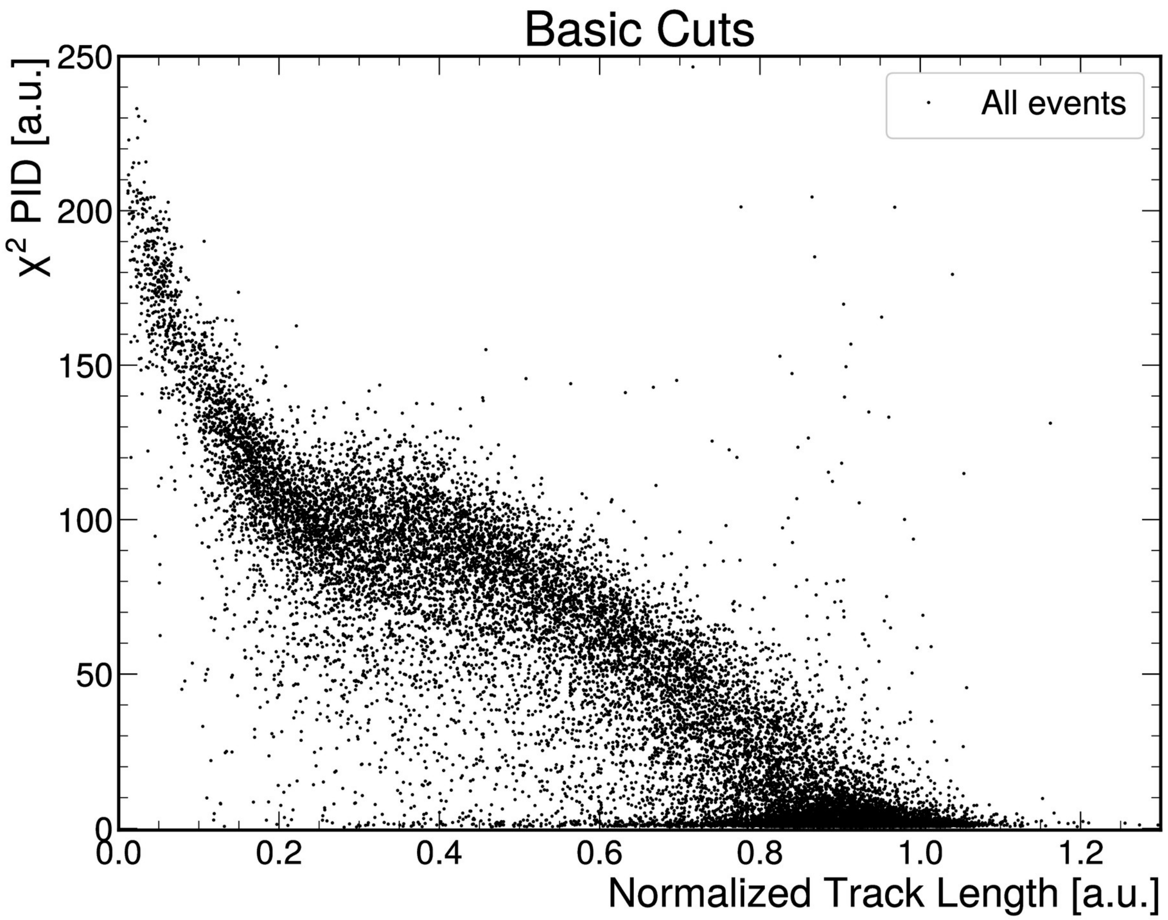
- F-score: A metric that sums up number of times each feature is split on
- Not surprised to see that PID is the most important feature

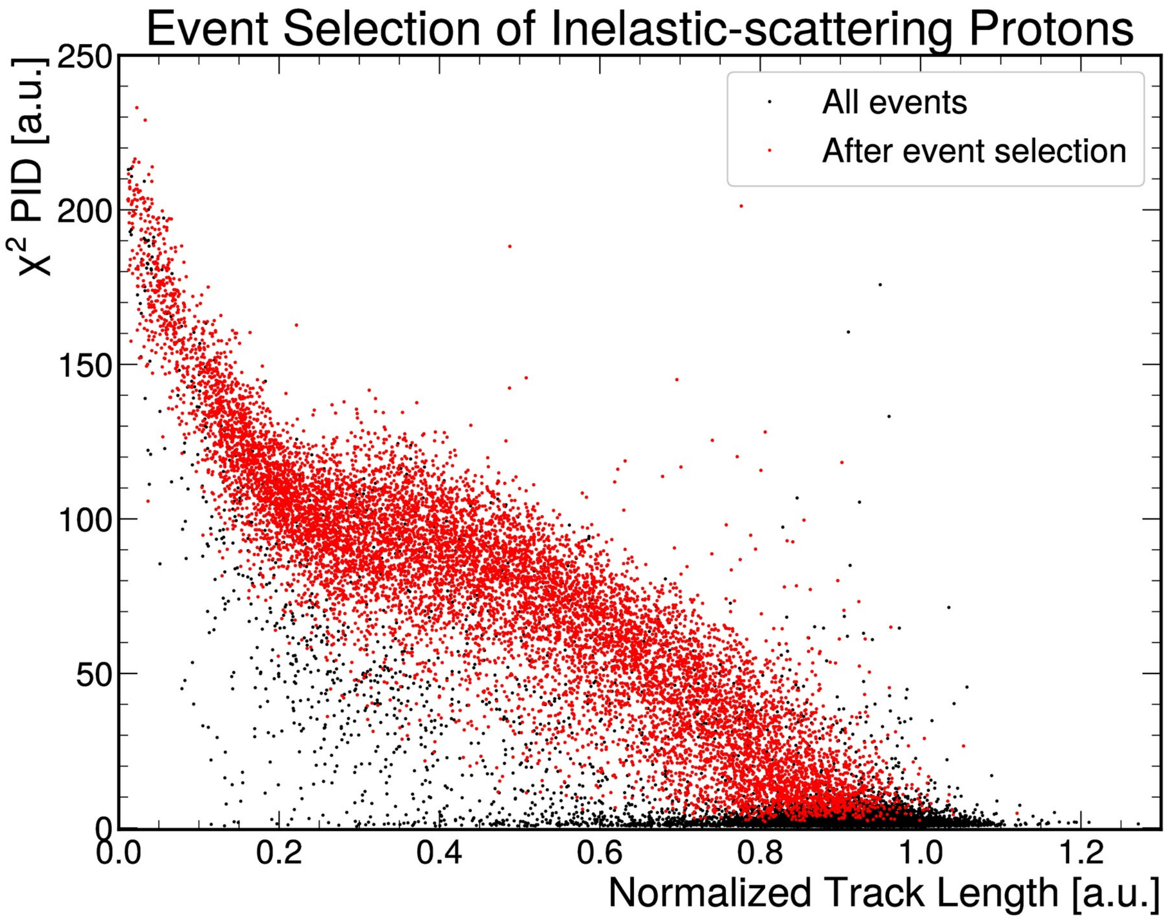
Training Result & Selection Cut



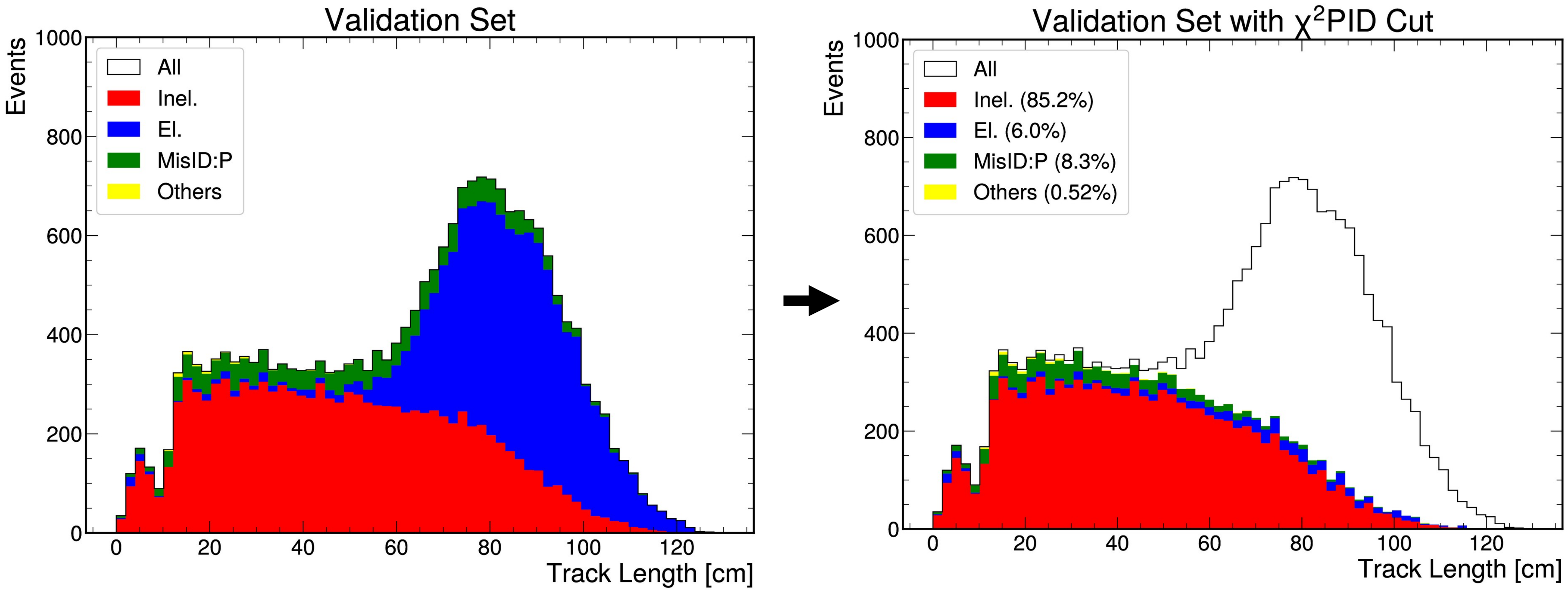
- Good separation between signal and background

Before/After BDT Cut

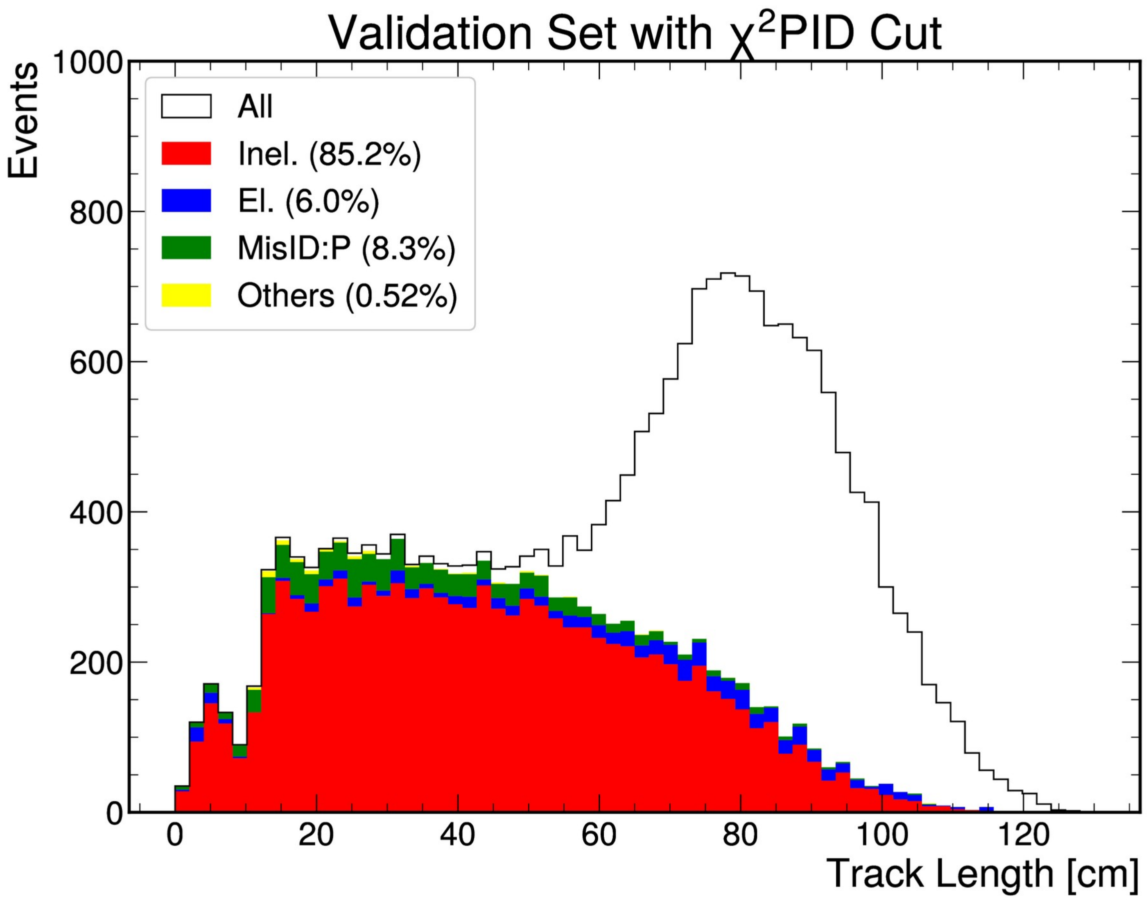


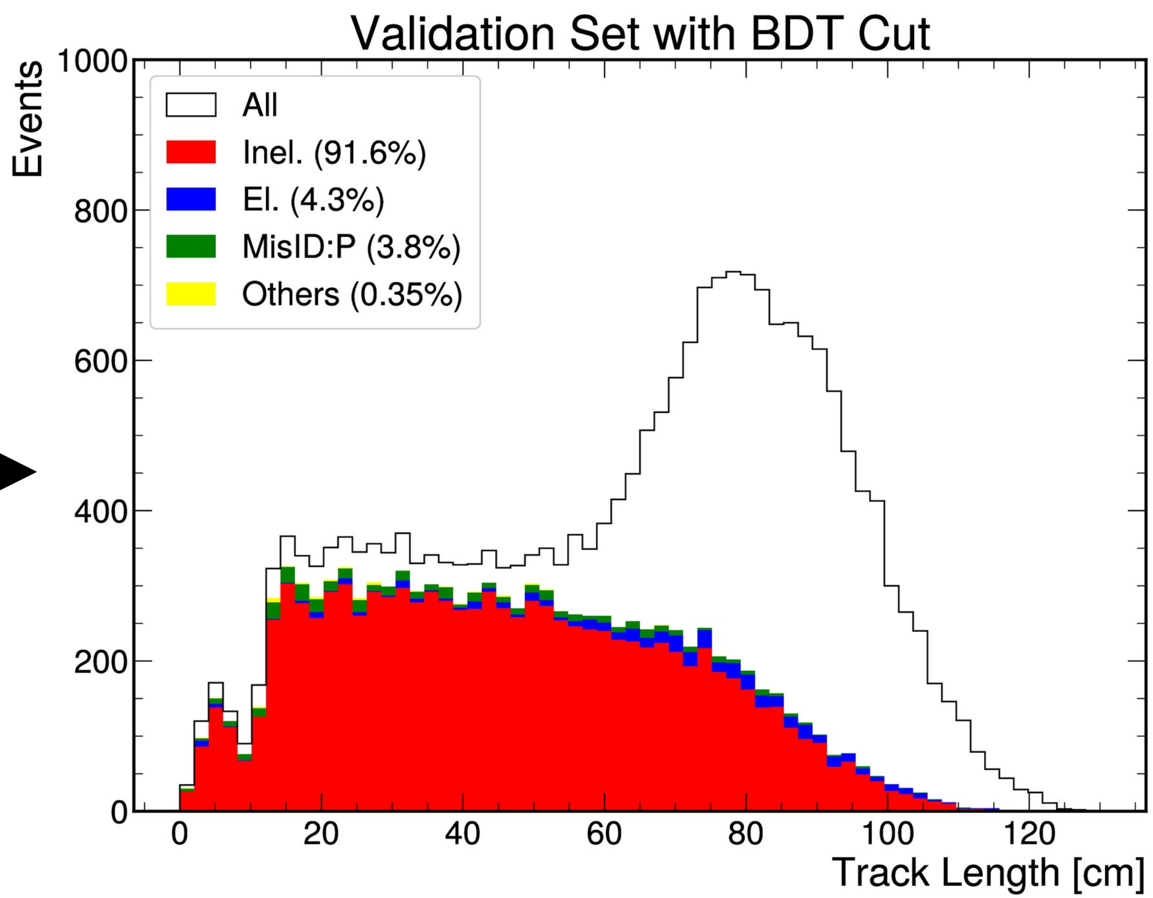


No Cut/Chi2 Cut



Chi2 Cut/BDT Cut





- Inel.: 6% improvement (91 % purity obtained)
  (4% MisID:P + 2 % El. background)

Backup

AUC Using TMVA