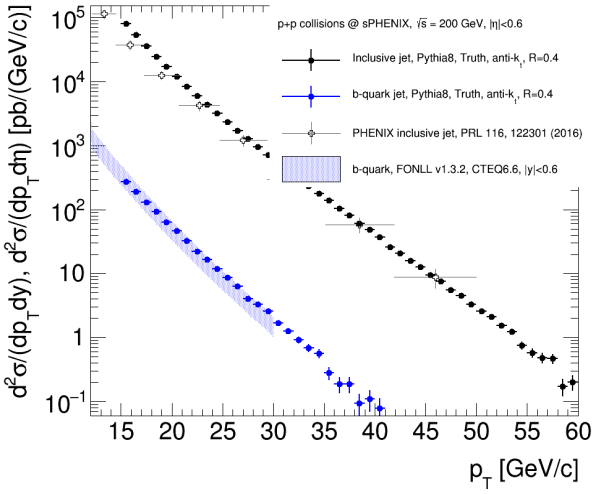
Brief review of b-jet tagging - high DCA track count

Initial b-jet fraction with Pythia 8 compared with FONLL



Initial b-jet fraction was estimated by Pythia 8

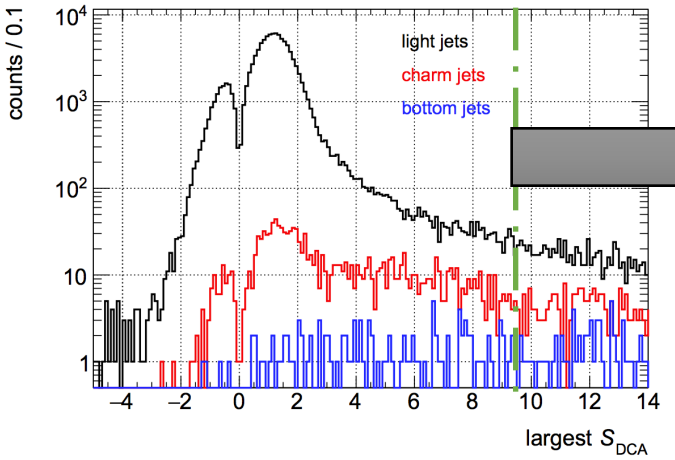
- consistent with FONLL calculation

Jet Reco: FastJet package, anti-kT method, from EMCal and HCal Towers

DCA and PCA (point of closest approach) position reported from tracking software
sign determined by PCA position, vertex position and jet vector

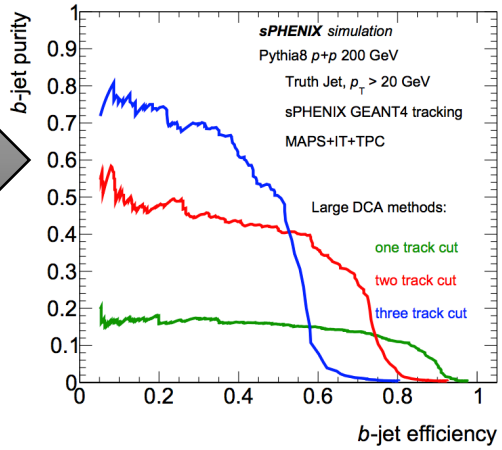
Eff. vs. cut for l/c/b jet + initial b-jet fraction ⟹
b-jet purity vs. b-jet efficiency

3D DCA distribution



Cut

Direct Cut



More sophisticated algorithm?

- Need to output one curve
- Using all DCA information comprehensively
- Using more features to help

Maximum Likelihood, Random Forest, Neural Network

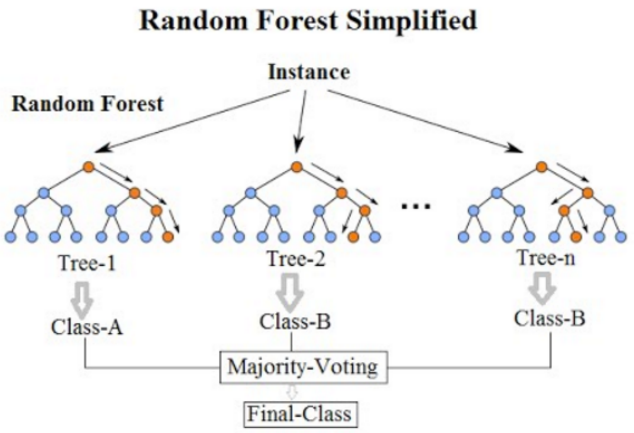
Tried Random Forest in this study

- effective, interpretable, fast, handles correlated data
- used 'randomForest' package in R
- https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm#overview
- At each split, 'mtry' features randomly chosen, use the one with best Gini gain

In the original paper on random forests, it was shown that the forest error rate depends on two things:

- The *correlation* between any two trees in the forest. Increasing the correlation increases the forest error rate.
- The *strength* of each individual tree in the forest. A tree with a low error rate is a strong classifier. Increasing the strength of the individual trees decreases the forest error rate.

Reducing m reduces both the correlation and the strength. Increasing it increases both. Somewhere in between is an "optimal" range of m - usually quite wide. Using the oob error rate (see below) a value of m in the range can quickly be found. This is the only adjustable parameter to which random forests is somewhat sensitive.



Candidate features

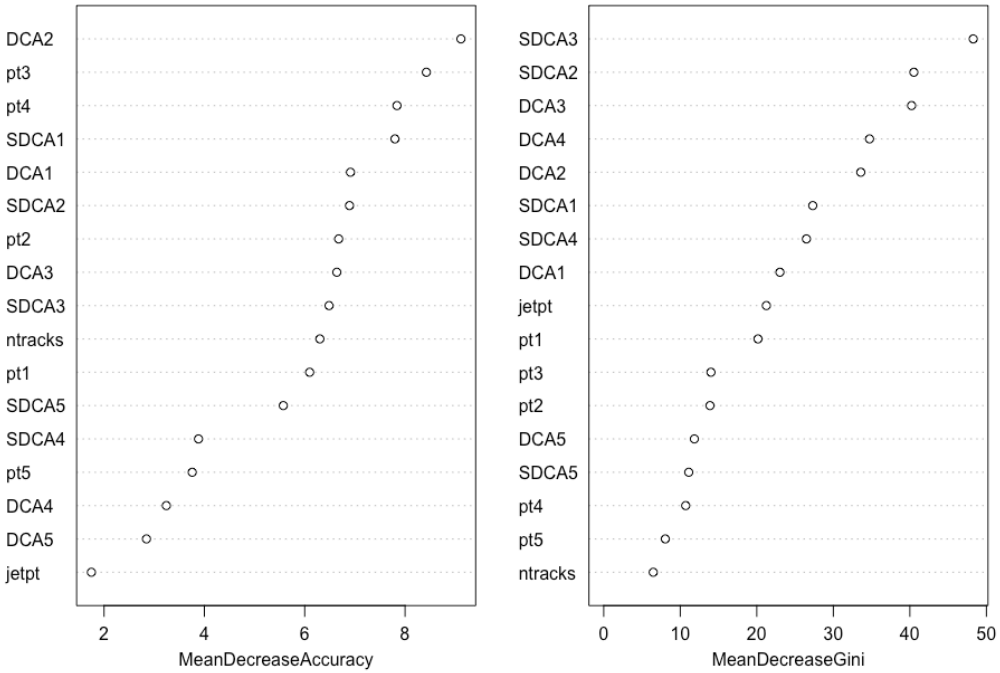
for a identified jet, find tracks within jet cone with

- Largest 3D DCA (DCA1), second largest 3D DCA (DCA2), ..., DCA5
- Largest 3D DCA/error (SDCA1), ..., SDCA5

pT for tracks with SDCA1, SDCA2... denoted as pT1, pT2...pT5

jet features as jet_pT, ntracks in jet cone

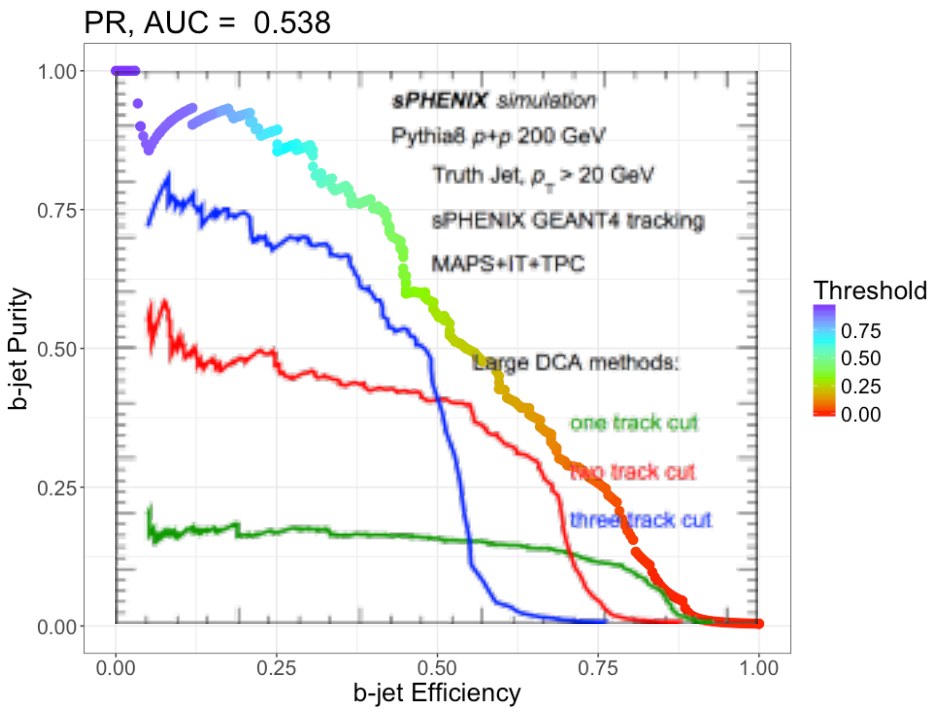
feature importance



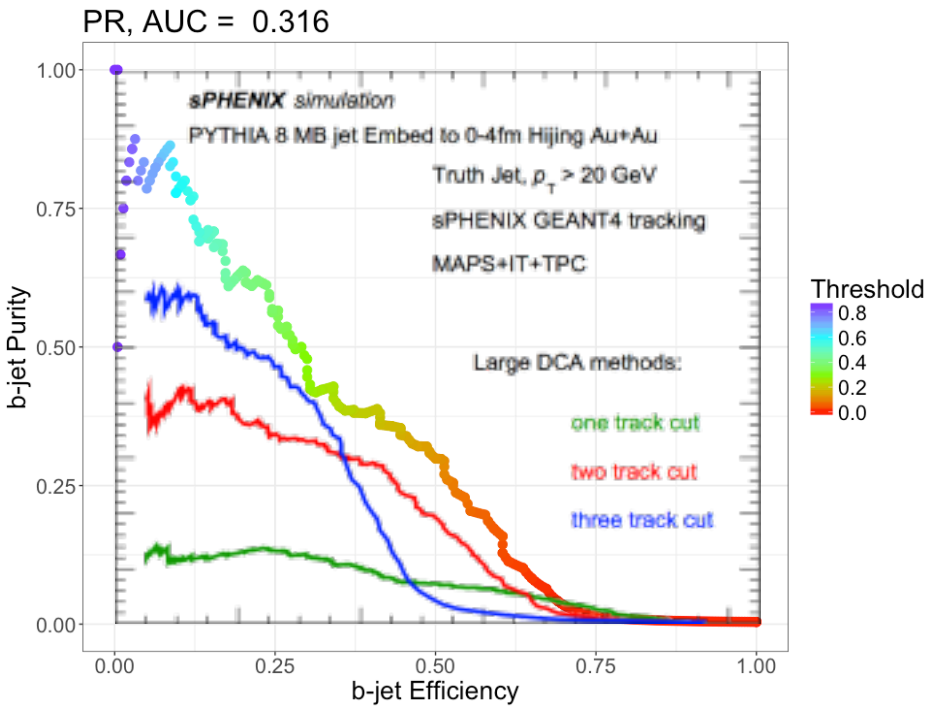
First batch results

- split data equally into training and testing
- mtry = 4, ntree = 200
- some improvement compared with direct cut method for both pp and hijing especially at high end

pp



hijing



Working on

- balancing data - weighting class error
- feature selection
- running more data
- some tuning of 'mtry'