Machine Learning for Tagging and Removing Pileup

Aviv Cukierman[1,2] and Max Zimet[1]

December 7, 2015

[1]SLAC; [2]Stanford Physics Department

U.S. DEPARTMENT OF ENERGY
Office of Science

SLAC NATIONAL ACCELERATOR LABORATORY

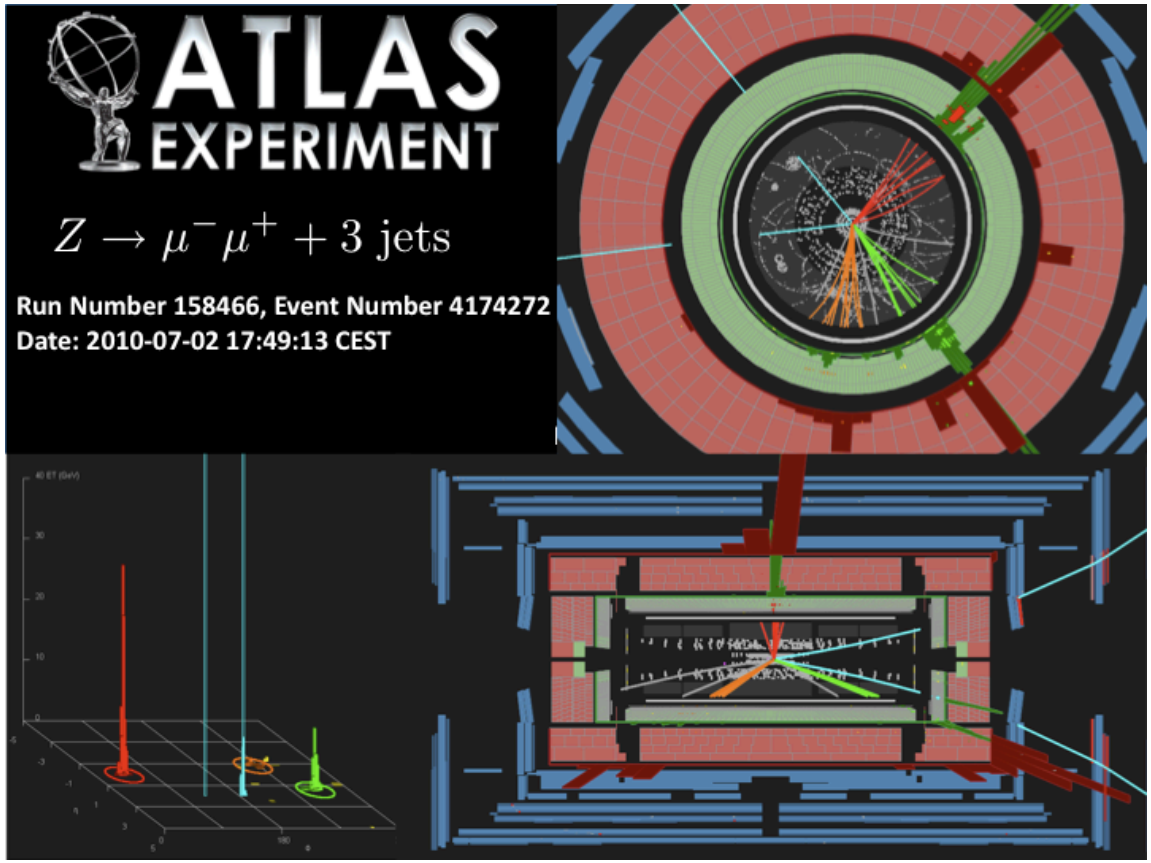
Outline

- Motivation and Background
- Techniques
- Figures of Merit
- Results
- Conclusions

Motivation and Background

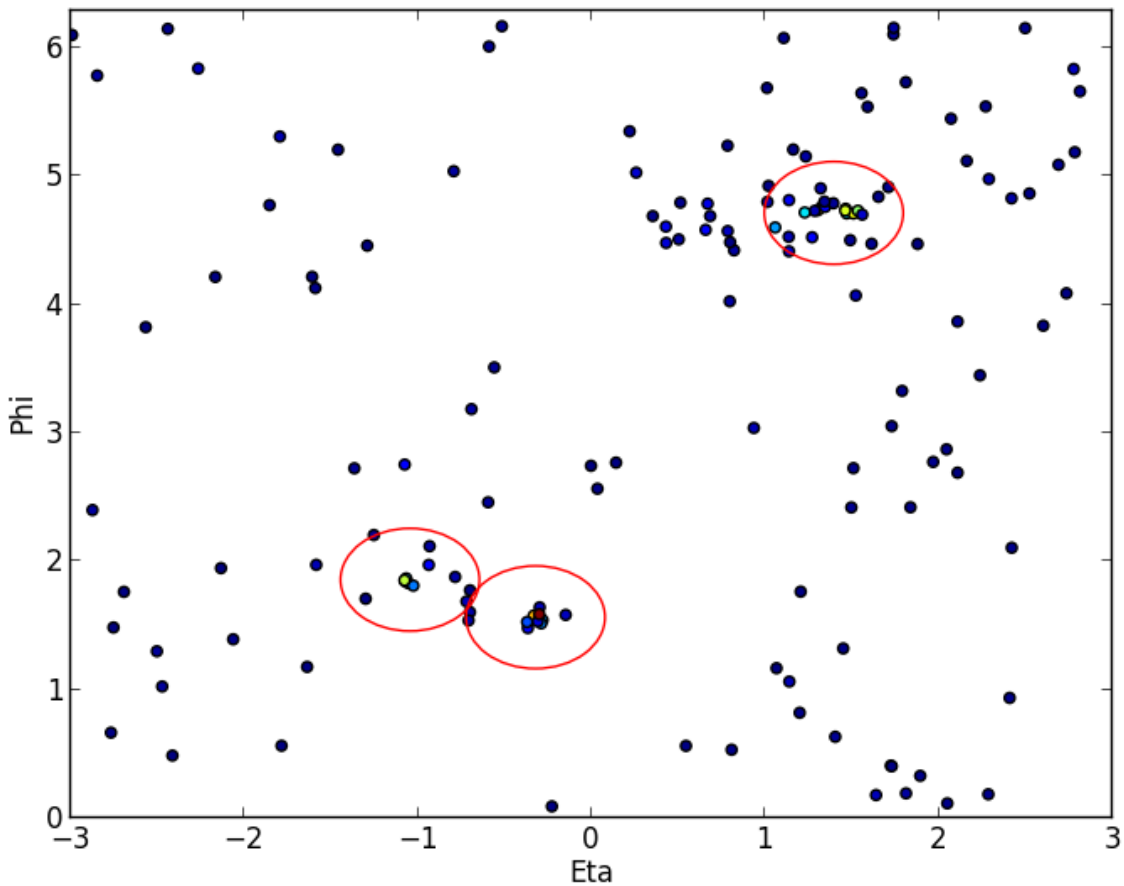
Jets

- Large number of particles produced in high-energy collisions
- Hadronizing quarks and gluons result in collimated sprays of energetic particles – clusters of particles in detector

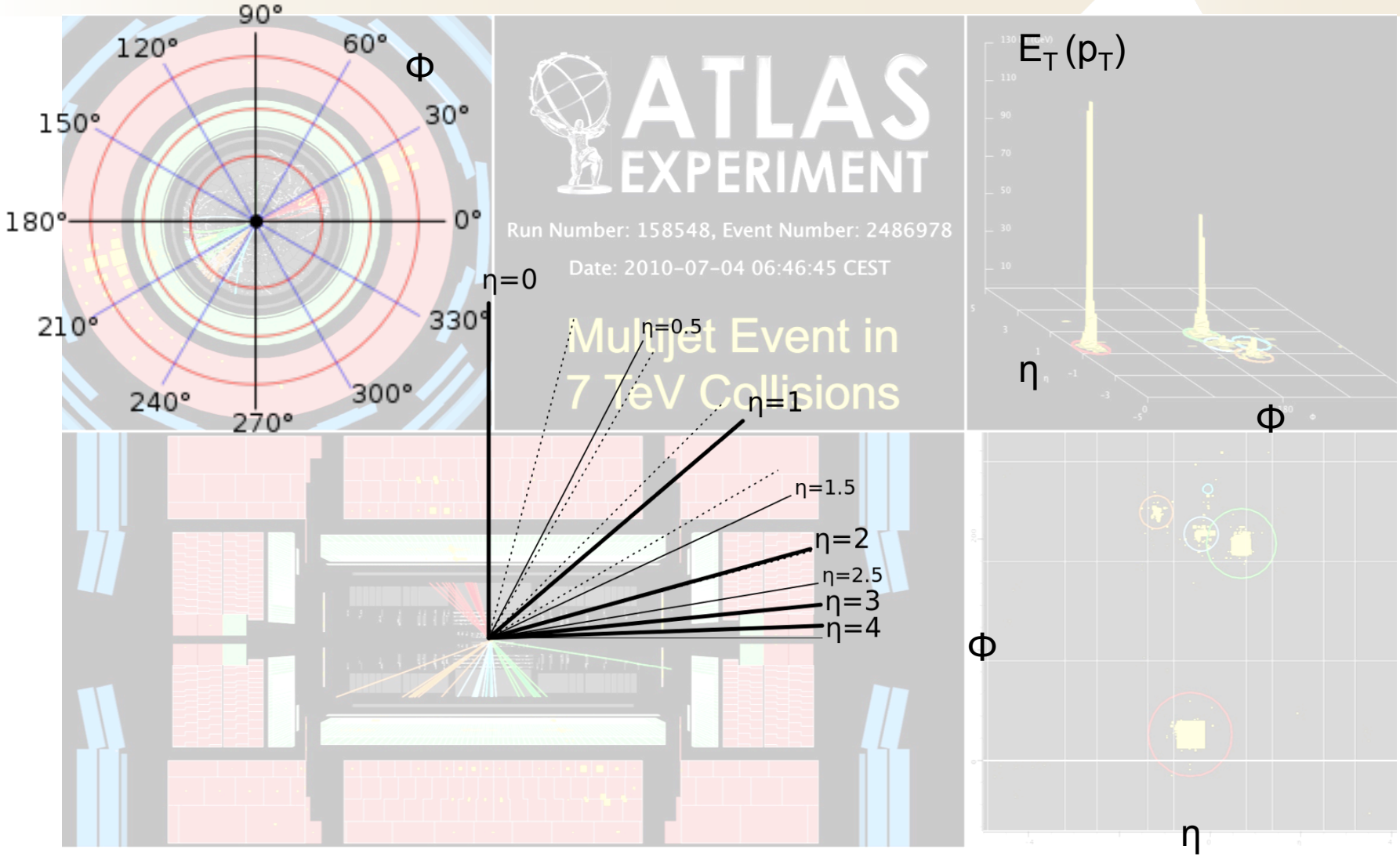


Jets

- Well-developed clustering algorithms for particles

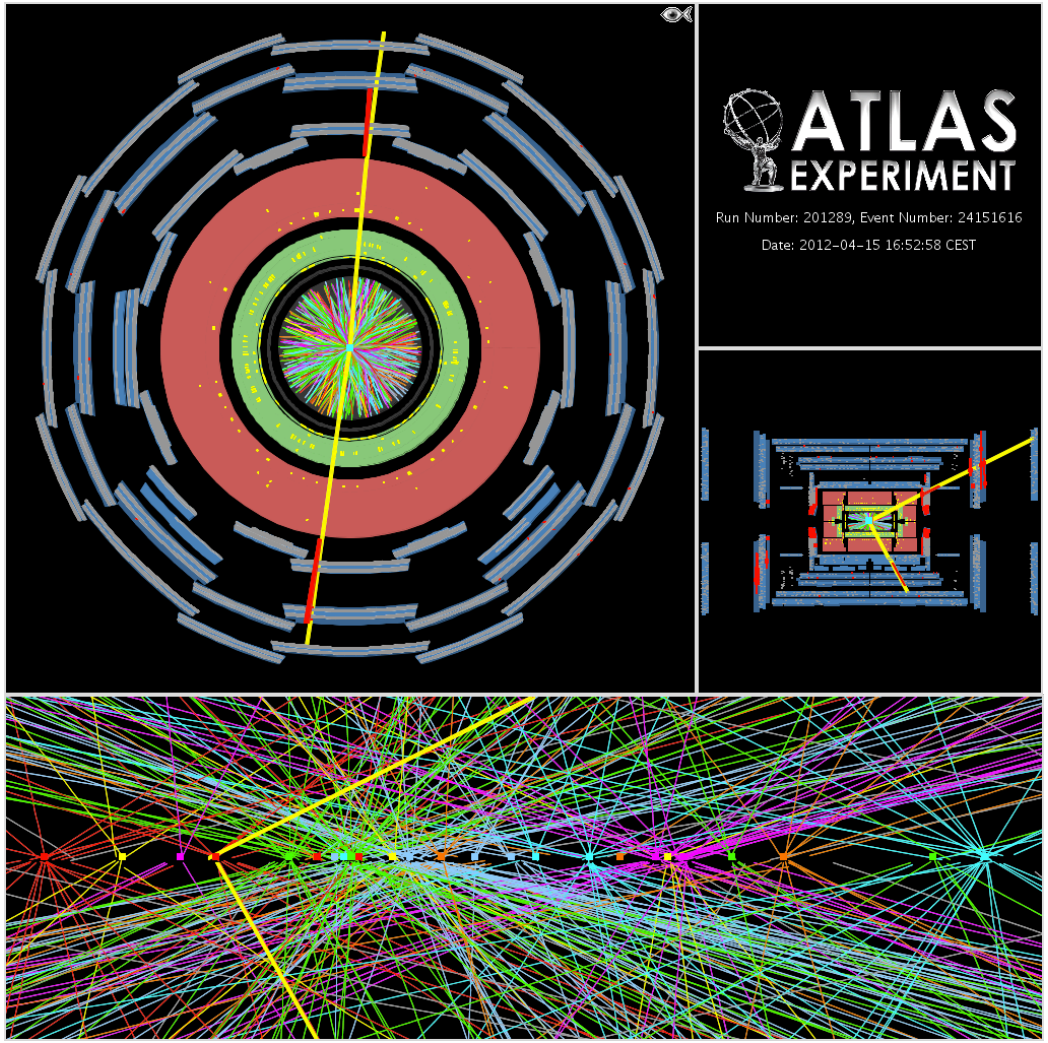


Definitions



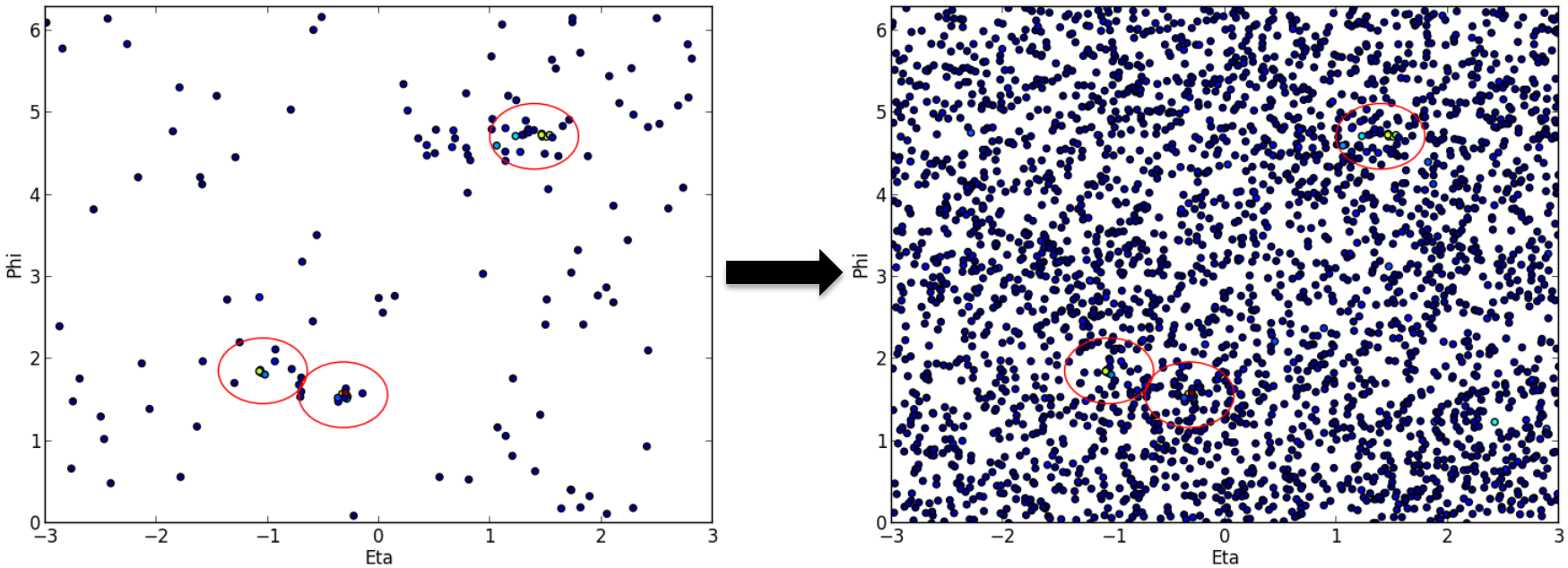
Pileup

- Many protons collided at once
- Multiple interaction points
- In a given event, one interesting interaction, the rest uninteresting physics
- Pileup superposed on top of jets



Pileup

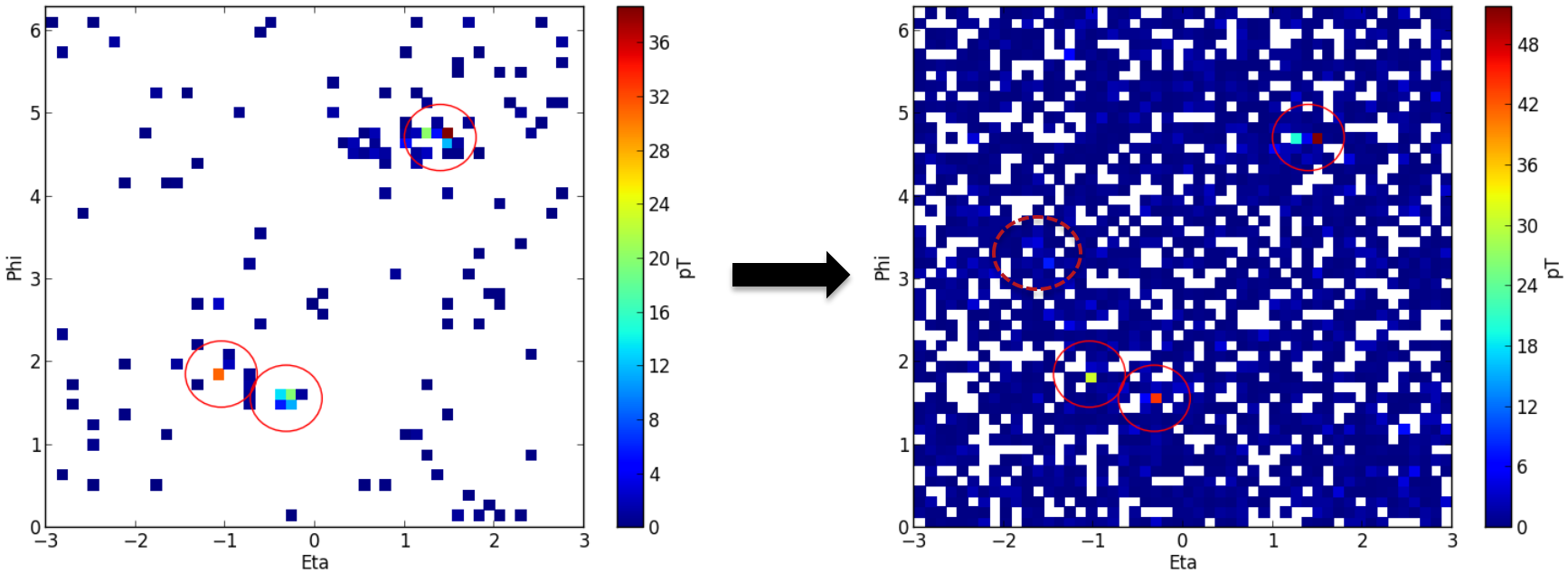
- Pileup makes clustering harder



- Challenge: Reconstruct jets while mitigating pileup contribution

Pileup

- Pileup makes clustering harder

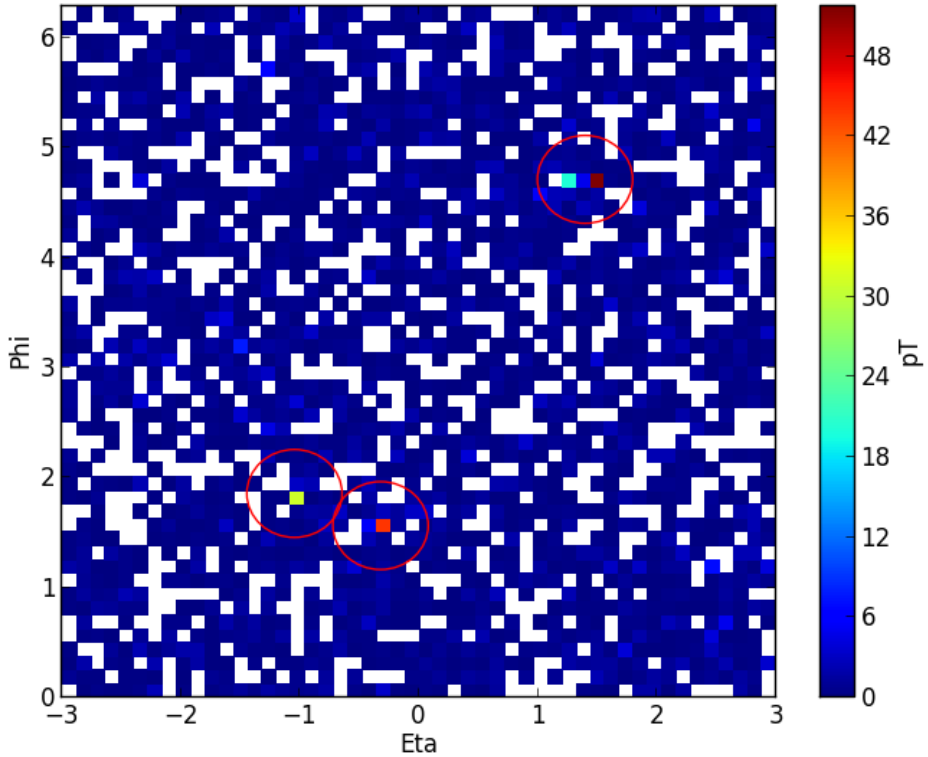


- Challenge: Reconstruct jets while mitigating pileup contribution

Techniques

Area Subtraction

- Estimate mean density of pileup energy deposit in event
- Estimate area of jet
- Subtract area * pileup energy density
- Current state of the art in ATLAS



Pileup Tagging

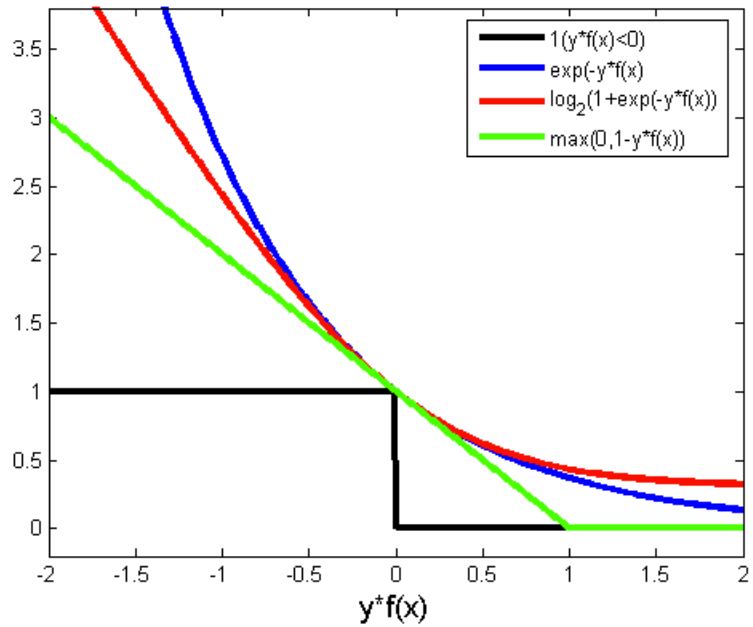
- Use machine learning techniques to tag particles as pileup
- Remove tagged particles
- Run clustering on remaining particles

Pileup Tagging

- Linear classifier (use simulations for training)
- Features
  - Sum p_T in cones around particle
  - Sum p_T/(distance from particle) in cones around particle
    - PUPPI (http://arxiv.org/abs/1407.6013)
  - Tracking information in cones around particle
    - CVF (Francesco Rubbo, Aviv Cukierman)
  - Eta, phi of particle
  - ~~p_T of particle~~
  - Combinations, kernels, indicators, etc. of above features
- Note: only care about classifying particles within jets
  - Assume all particles outside of jets are pileup

Pileup Tagging

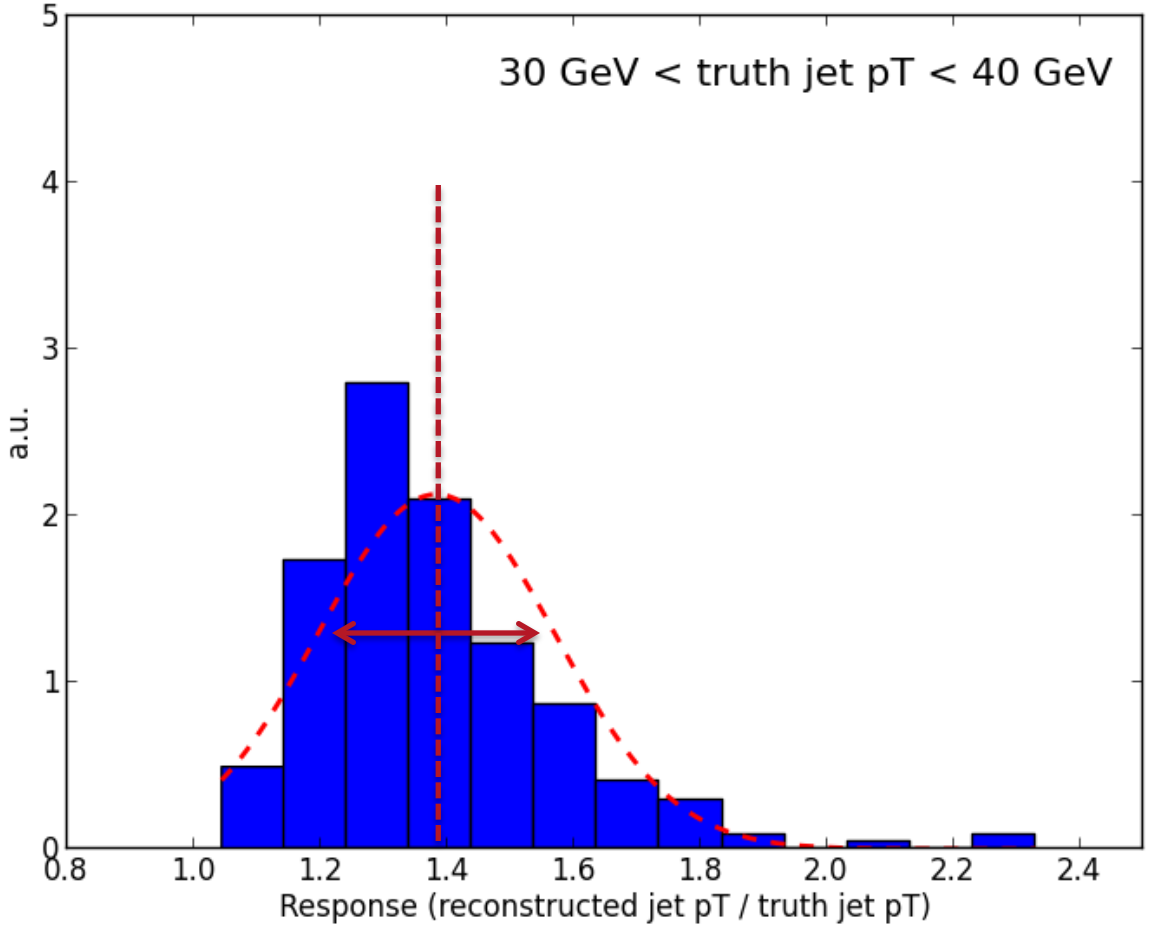
- Feature vector Φ(particle); Weight vector w; Classification score f = $\Phi^T w$; Prediction: sign(f)
- Truth value y = {-1,1}
- Loss: hinge loss (y*f)
- Problem – to first order, everything is pileup!
- Fix: penalize mislabeling non-pileup more
  - For pileup, hinge loss (y*f)
  - For not pileup, x * hinge loss (y*f)
    - x = {2,5,10,…}
- Remove pileup particles
- Weight pileup particles by score



Figures of Merit

Response

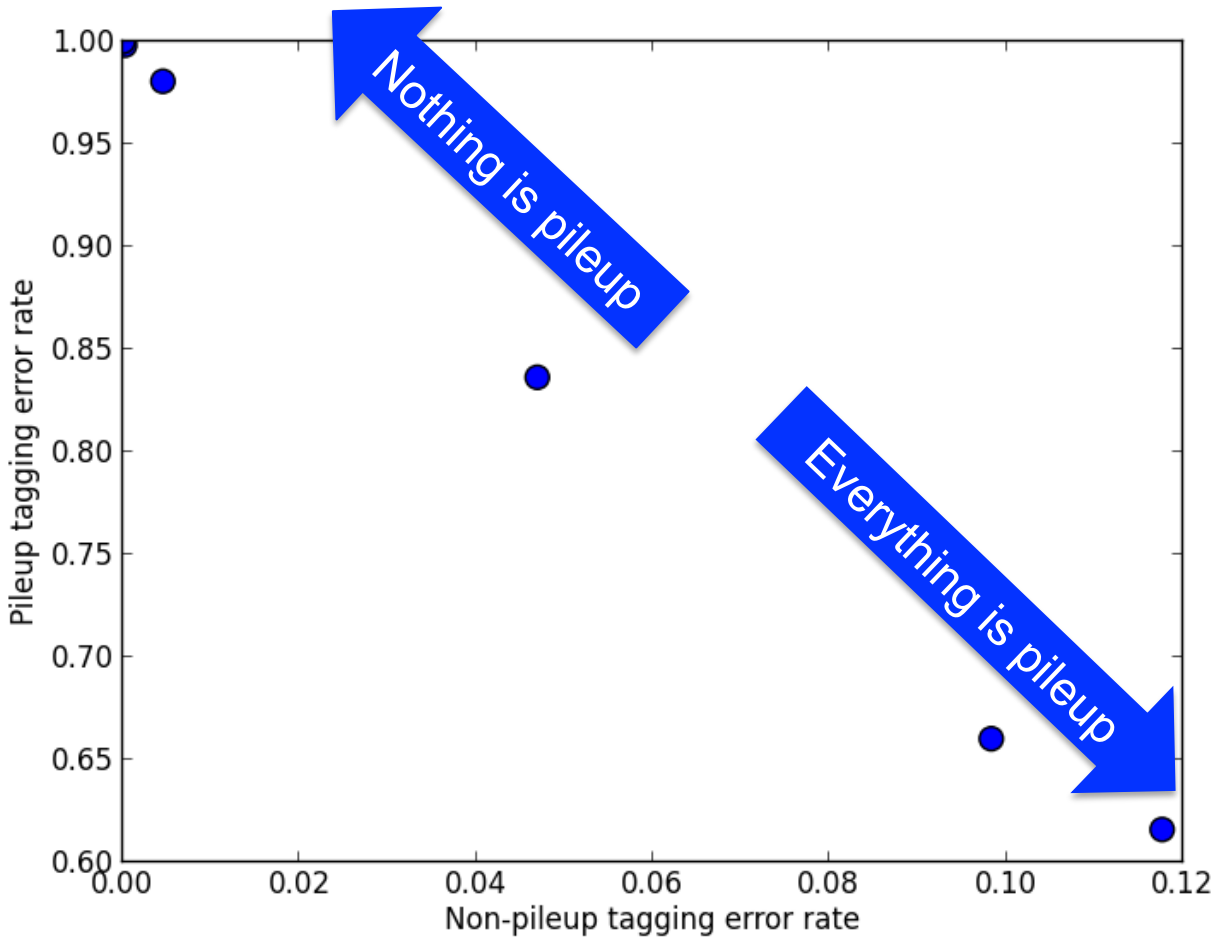
- Difference between reconstructed jet p_T and true value
  - Get true value by running jet clustering on only non-pileup particles



Results

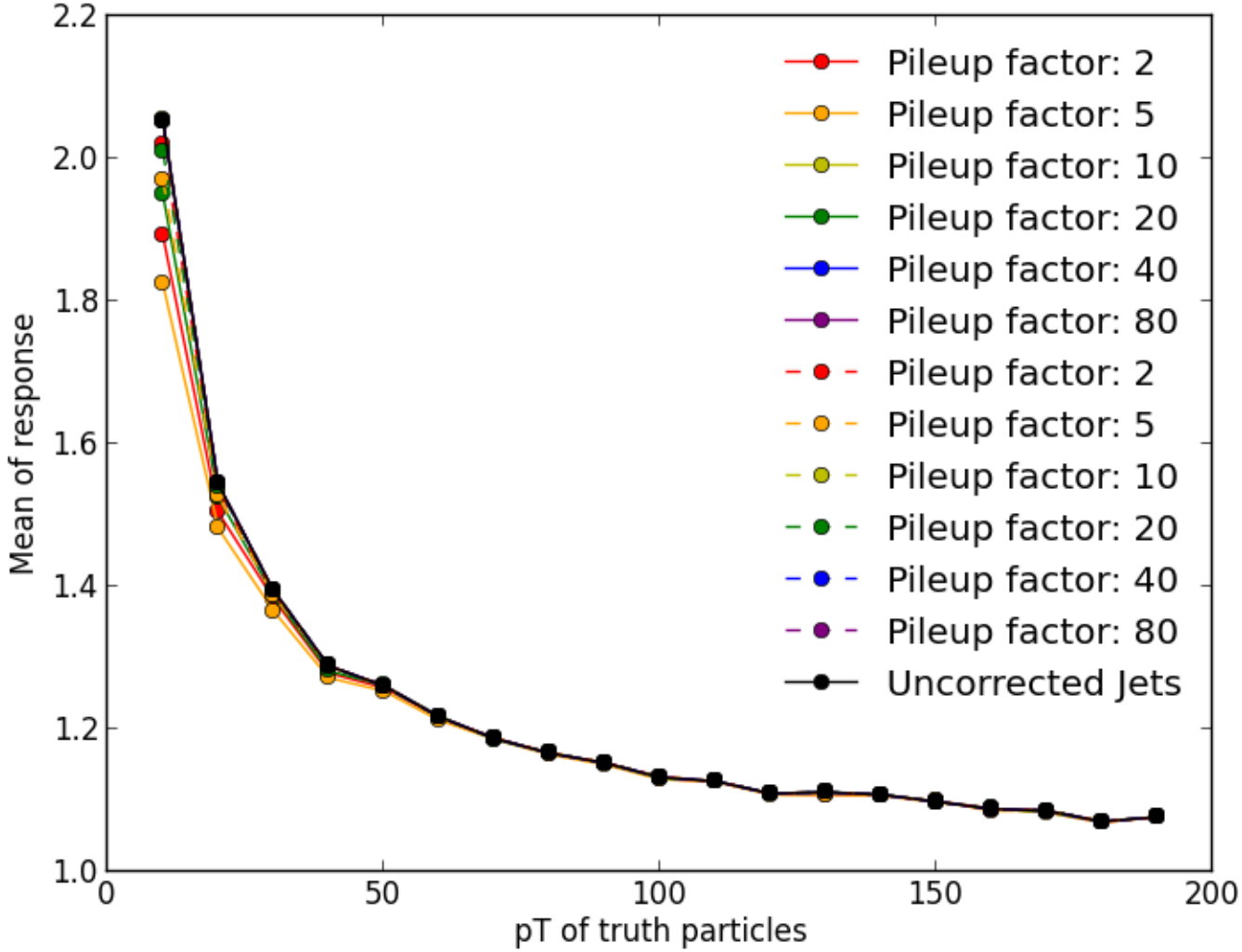
ROC Curve

- Balance between false positives and false negatives



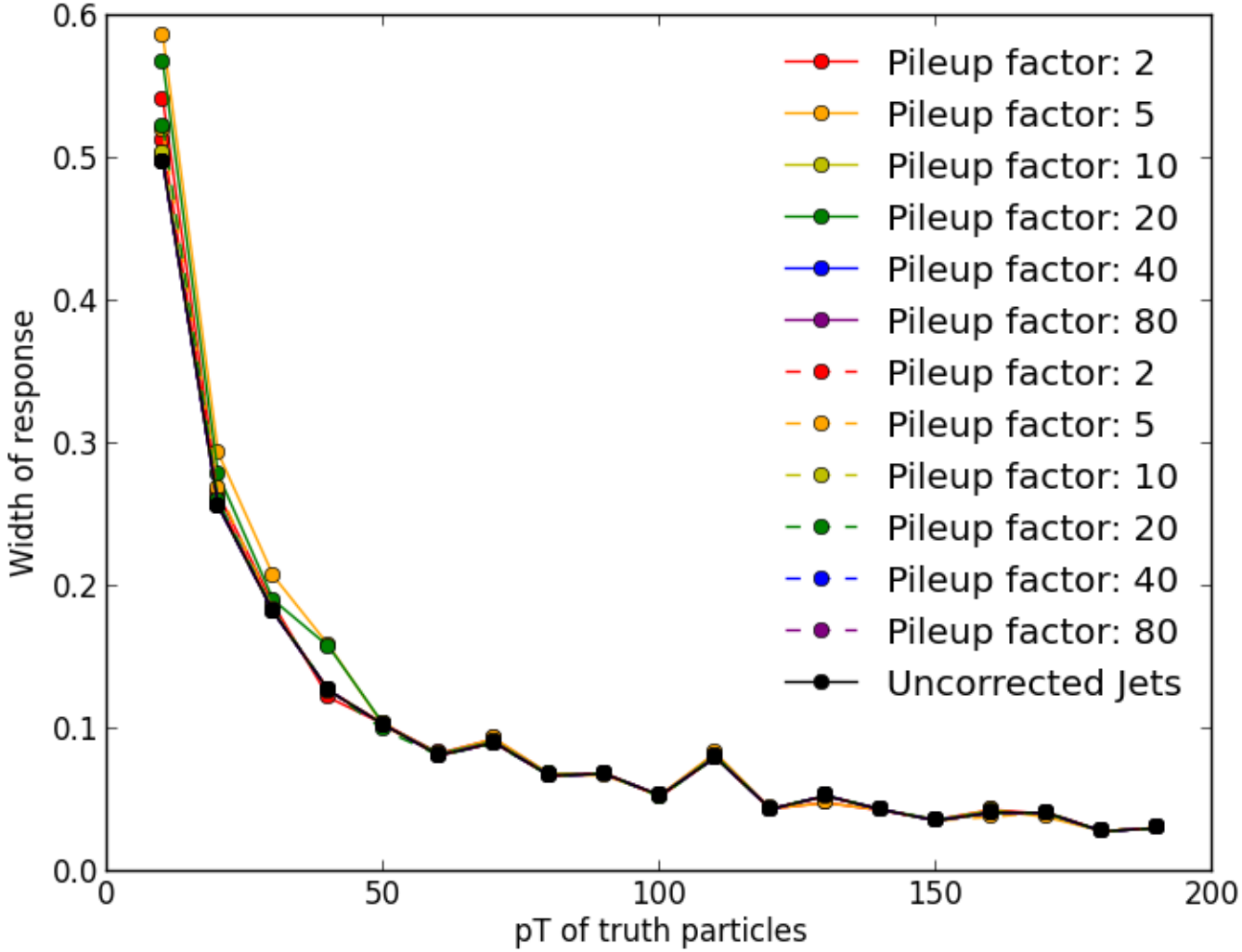
Mean of Response

- Not much improvement yet



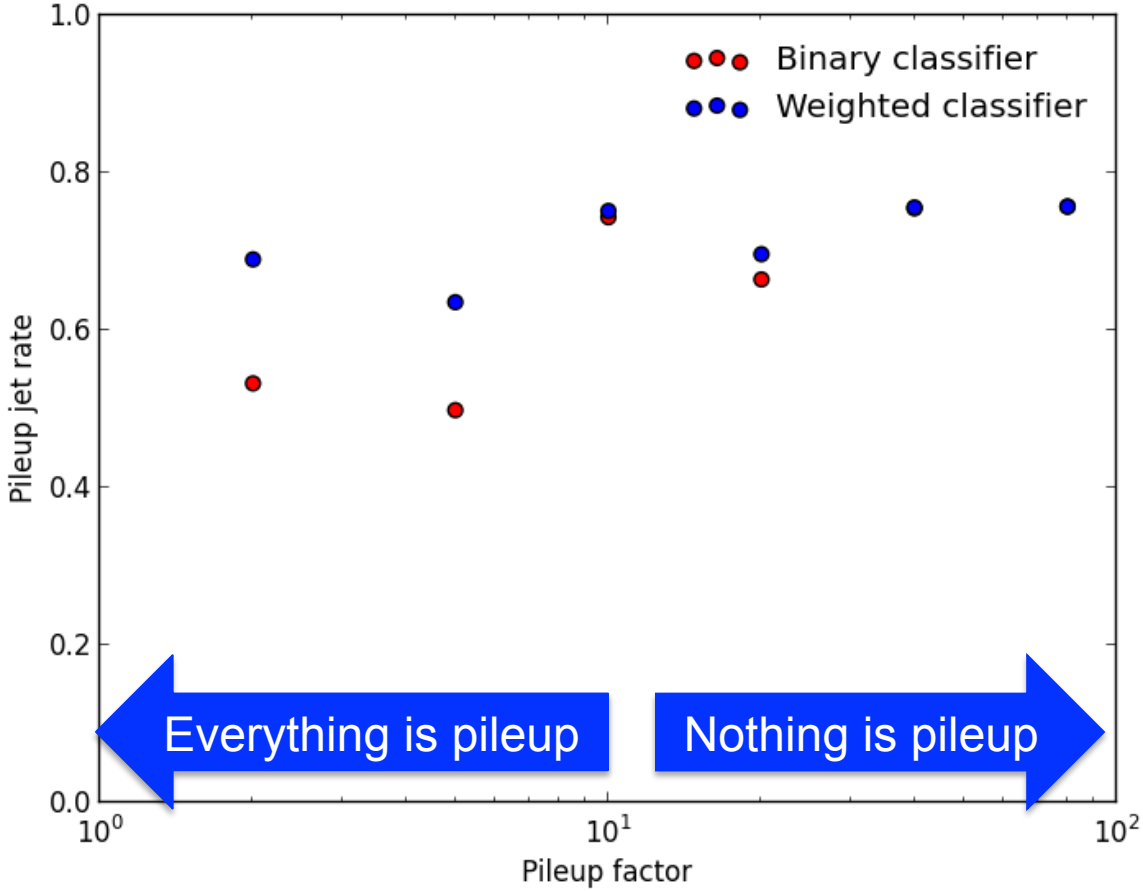
Width of Response

- Not much improvement yet



Pileup jet rate

- Trained on particles in all jets, not just jets we know are not from pileup
- We learned how to suppress pileup jets!



Conclusions and Future Work

Conclusions and Future Work

SLAC

- Learned how to suppress pileup jets
  - Not the original goal
- Future work:
  - Choose training set more carefully
    - Try training only on particles in high-pT jets assigned to non-pileup truth jets
  - Use neural networks to learn features
  - Incorporate area subtraction after classification
  - Use more advanced techniques for classification
  - Examine features for discrimination power
- Other ideas (from AI):
  - Treat anti-kT as search through space of possible jets, optimize over anti-kT distance metric cost
  - Treat event as factor graph – variables are particles, value is which jet it's in; weight between factors is higher if in truth they are in the same jet
    - Include pileup jets