



Classification tree Update

Toby Burnett

Frank Golf

The old way

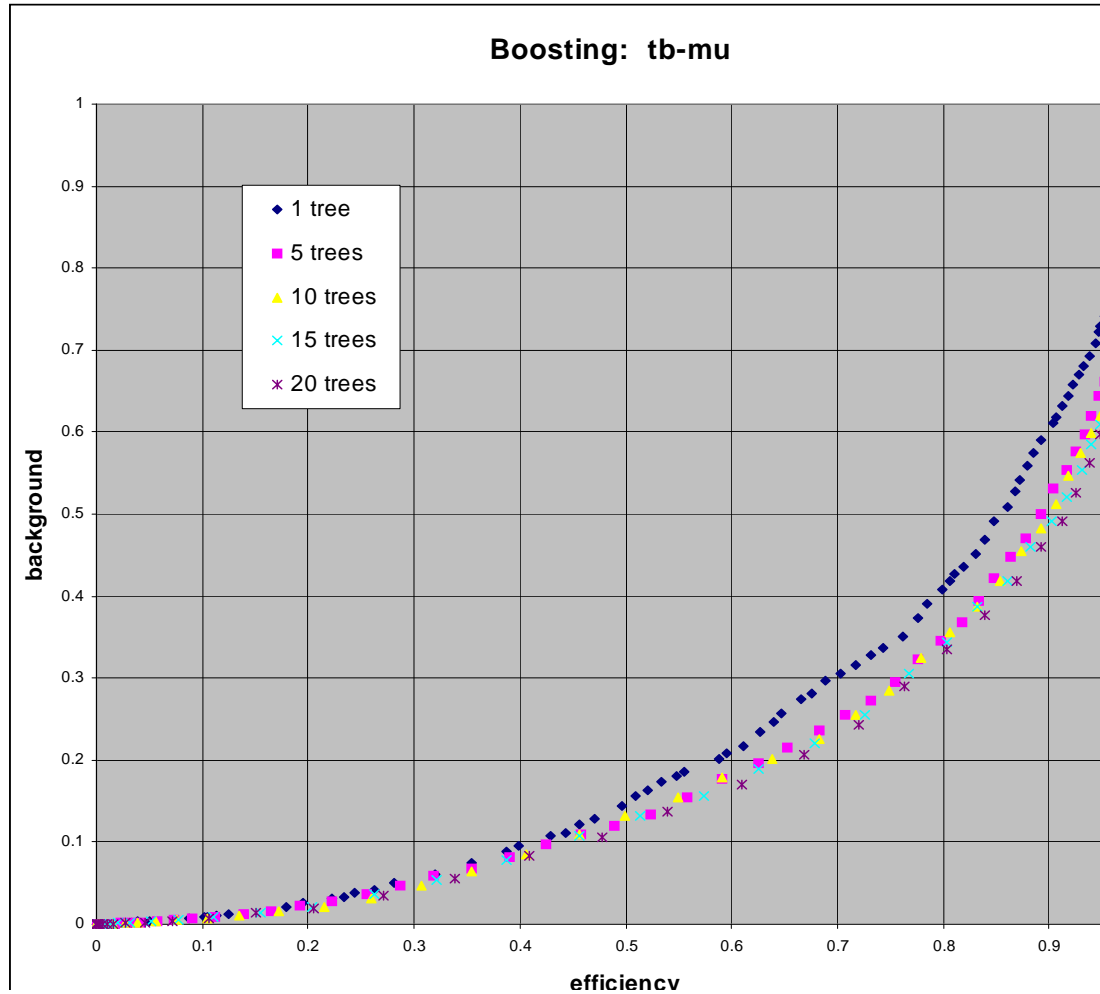
- Recall the current scheme we have to use the classification tree technology:
 - Bill creates a tree using Insightful miner, and makes the XML file containing the trees available
 - I import this file into the merit package, and write additional code to mimic the cuts he made that were not in the trees
 - The classification package is used to evaluate the trees from the AnalysisNTuple variables, with output (the IM variables) being added to the tuple.
 - I spend some more time verifying that I got it right.

The new way:

- We have a new package, *classifier*, which contains C++ classes to create trees, and evaluate them. It uses ROOT TTrees, and generates a simple ascii table with the tree definition. See the previous talk:
 - http://www-glast.slac.stanford.edu/software/anagroup/burnett/GLAST_Decision_Trees.pdf
- Advantages are:
 - Anyone can train trees, play with variables, etc.
 - Advanced techniques involving multiple trees, e.g. boosting, are available

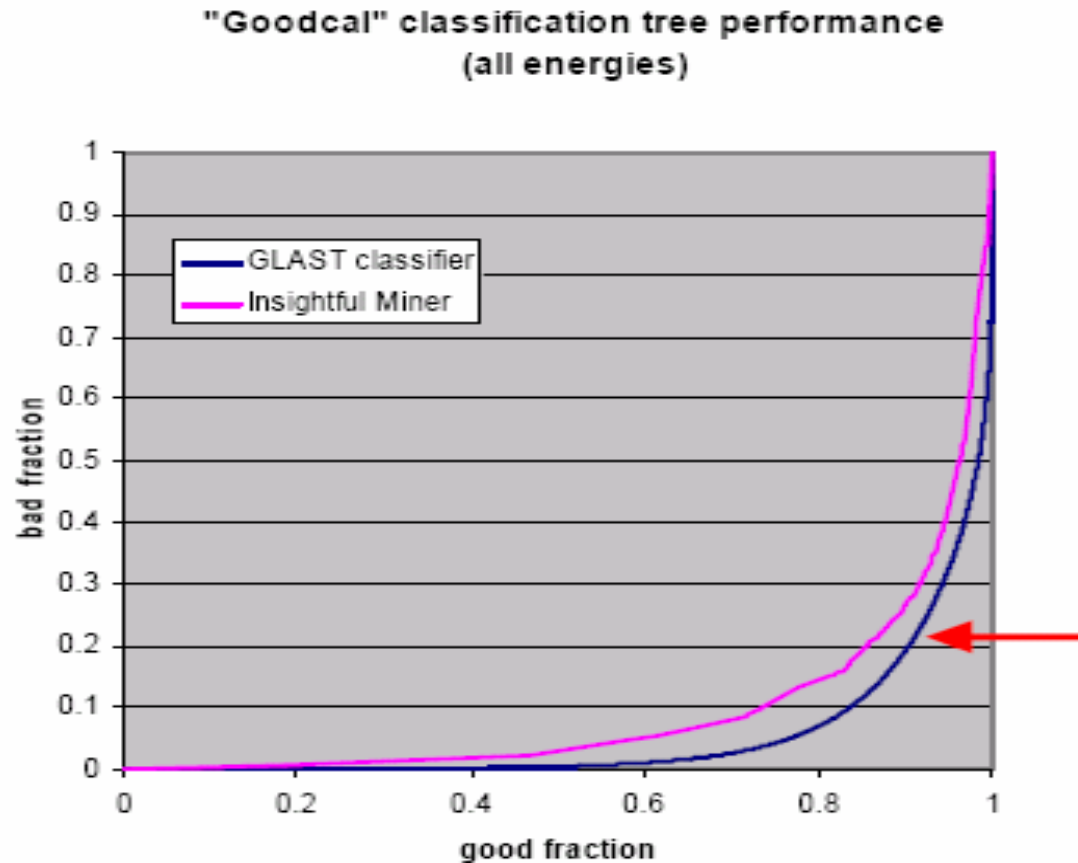
Current status: boosting works!

Example using D0 data



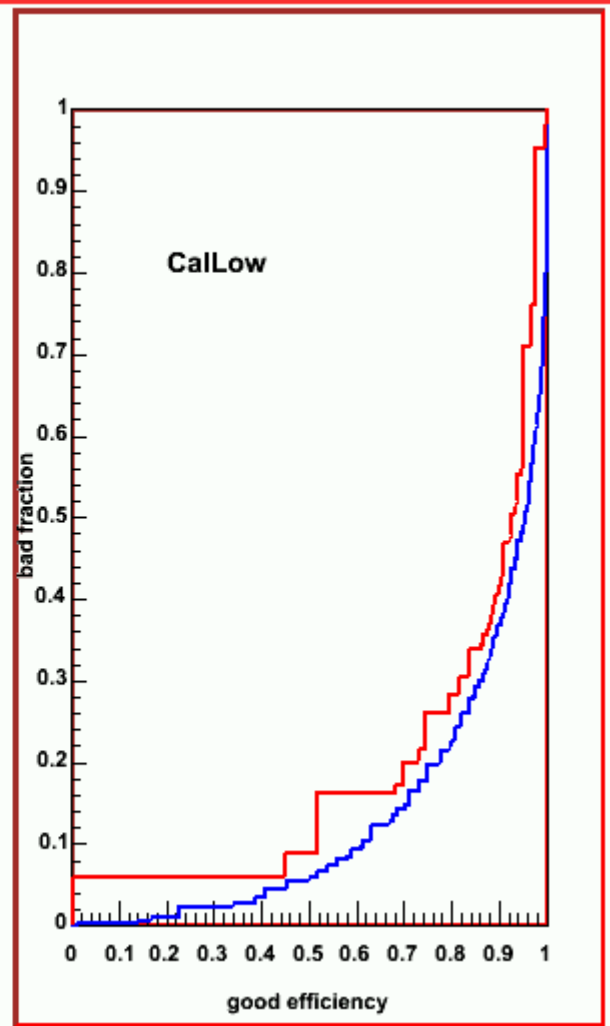
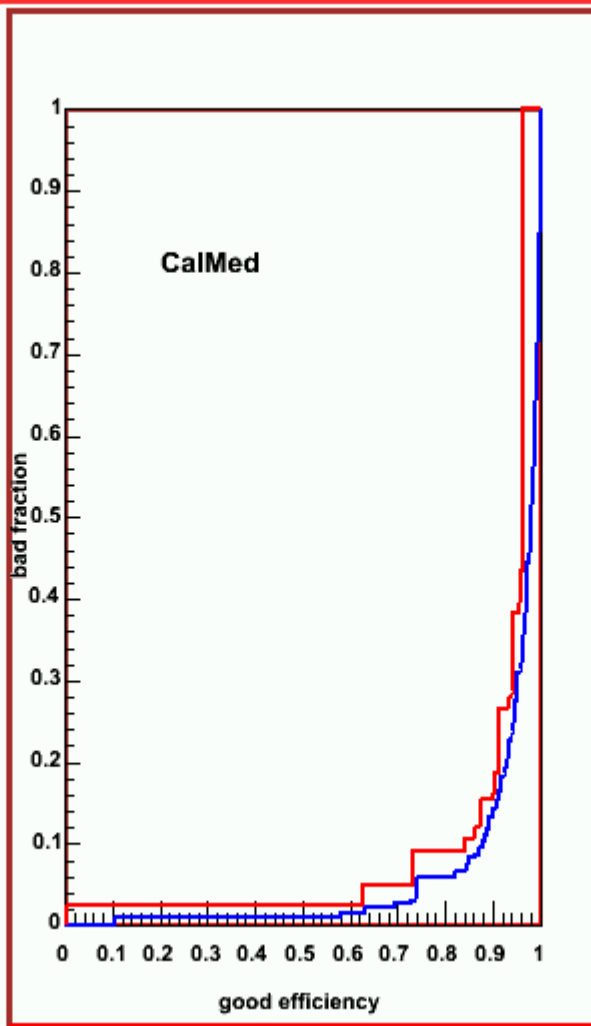
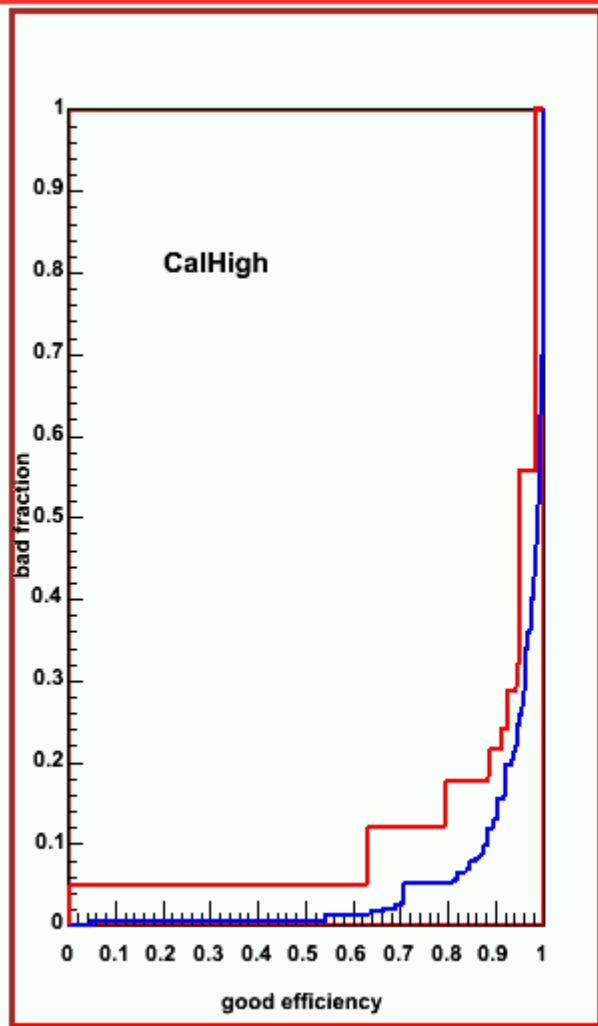
Back to GLAST...

We showed this plot before but had not compared our single tree with Bill's individual trees for low, medium, and high cal energies.



Separate tree comparison

— IM
— classifier



Conclusions, next steps

- The current classifier single-tree algorithm applied to all energies is slightly better than the three individual IM trees
- Boosting will certainly improve the result
- Still to do:
 - One-track vs. vertex: which estimate is better?
 - PSF tail suppression
 - Good-gamma prediction