CAFAna (and CAFs): what's in the pipeline?

DUNE LBL workshop – CERN

Chris Backhouse

University College London

August 18, 2022

What are CAFs / CAFAna?

► Very briefly:

► **C**ommon **A**nalysis **F**iles are the high level representation of art (FD) or edepsim etc (ND) outputs useful for analysis

► CAFAna is a framework to
  1. Read those files and fill histograms in an efficient / ergonomic way
  2. Oscillation and cross-section analysis tools built on top of that
  3. Born Frequentist, but NOvA added two(!) Bayesian fitters

► Used for the FD TDR, PRISM analysis, and by other experiments

► Goal is to get the DUNE CAFs to the next level

► Please ask me to elaborate on any aspect of CAF-world!

CAF structure

- ► Klassic KAFs are an ad-hoc tree, and a mess
- ► NOvA and SBN use proper hierarchical structure
- ► Now exist for DUNE in the form of StandardRecord in `duneanaobj`
- ► IMO main goal of this workshop is to complete the switchover
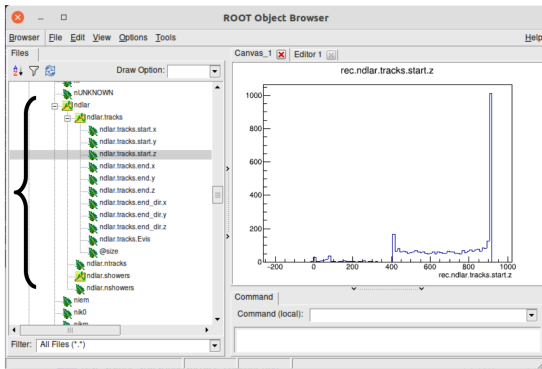
Reading new-style CAFs

- Some crude forward- / back-compatability because top-level fields left in place for now
- To explore non-top-level entries in hierarchy, and many gains in file reading efficiency, need `feature/srproxy` branch of `lblpwgtools`

- Main obstacle to merging this – systematics!
  - This all works in SBN so no fundamental issue
  - Some troubles filling syst weights (mixed up with switch to genie3?)
  - Most important customer right now is PRISM who have a very long-lived branch needing merging and testing
- Personal goal this workshop to get this in so we can move forward

FlatCAFs
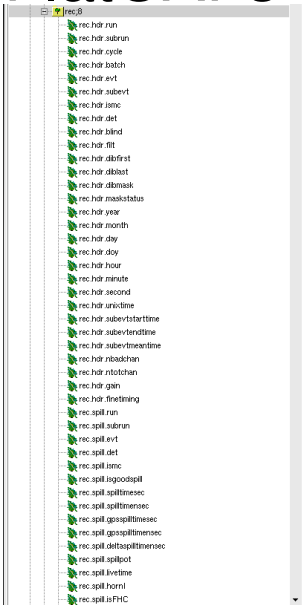
(NOVa flatcaf)

- ▶ Speed of `TTree::Draw()` comes from columnar layout and only touching the fields you need
- ▶ CAFAna with SRProxy does the same
- ▶ At some level of nesting ROOT insists you load the full object – trying to access individual TBranches goes wrong[1]
- ▶ FlatCAFs encode exactly the same StandardRecord structure, and look exactly the same from CAFAna, but are laid out explicitly in plain TTrees
- ▶ Can give dramatic file-reading speedup
- ▶ Enables slimming by dropping entire fields
- ▶ One day should consider RNtuple / HDF5
- ▶ Should be fairly easy to enable – workshop subgoal

[1]See ROOT-9543 for this being closed WONTFIX

5/9

RecordSource

- ► Currently you create a spectrum like
  Spectrum s(loader, axis, cut, shift);
- ► Cut is actually _Cut<StandardRecord> etc
- ► Exposes too much about experiment-specific records to core code

- ► New structure has Spectrum s(src, axis);
  e.g. Spectrum s(loader[cut], axis);
- ► Can get more elaborate *e.g.*
  loader[spillcut].Slices()[slicecut].Tracks()[trackcut]
- ► This fills one histogram entry per track

- ► This is strictly better, but requires adjustment of old code
- ► In progress on feature/cafanacore_v2
- ► Adventurous welcome to help me test

EnsembleSpectrum

- ▶ Currently osc analysis duplicated in different "universes"
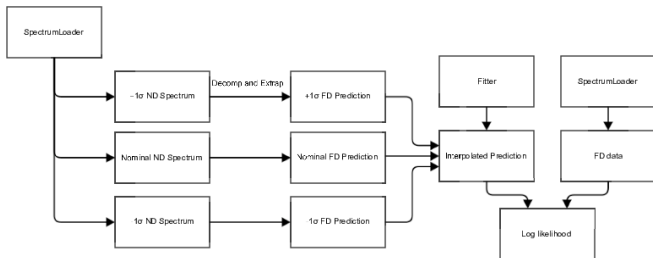
- ▶ In new scheme EnsembleSpectrum contains the universes, and there is a single Prediction etc
- ▶ Can trivially inspect uncertainty band at any point in process
- ▶ Individual Spectrum operations become "beefier", more suspectible to acceleration

```
EnsembleSpectrum(src.Ensemble(multiverse)[cut], axis);
```

EnsembleSpectrum (SBN plots)
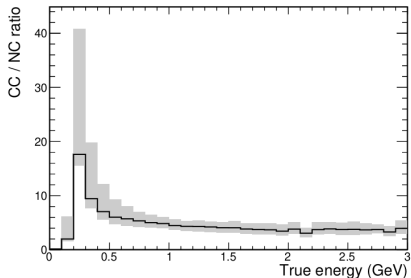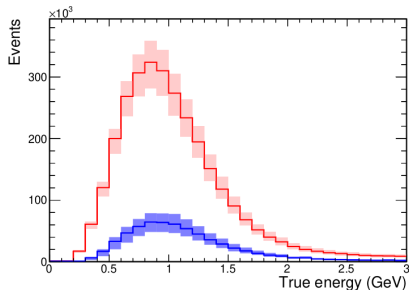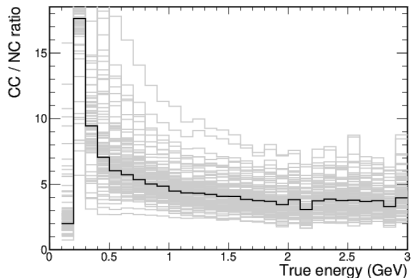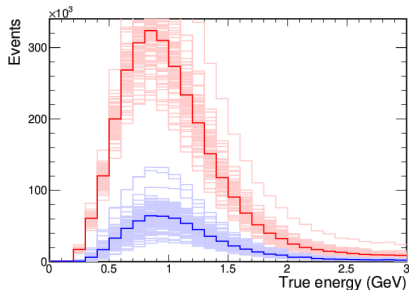
Other activity

PRISM

- ▶ Discussions of intrinsic vs accidental memory usage
- ▶ Get branch merged!

Steriles

- ▶ Under active development (Mike W et al)
- ▶ PISCES CovMx from NOvA (integrate with existing LBL covmx)
- ▶ Motivation to make proper `cafanafit`

MCMC

- ▶ Existing HMC implementation with STAN
  - ▶ Bring back up to date with NOvA
- ▶ New MH MCMC from NOvA/T2K work – "Aria"
- ▶ Pursue `cafanabayes`

- ▶ CAFs for NDGar
- ▶ CAFAna for atmospheric oscillations
- ▶ CAFAna for xsec analyses

  . . .