Status of HighLAND and ProtoDUNE analysis

A. Cervera
(IFIC-Valencia)

IFIC e-noval meeting
07/03/2019

Introduction

- Trying to produce highland MiniTrees from ProtoDUNE LArSoft files
- Able to do it at Fermilab SL7 machines and my laptop.
    - Problems on SL6 and also at IFIC
- Immediate goal is to provide input for Fran's analysis
    - Miguel is also starting to do Analysis
- Next goal is to convince people that HighLAND is the way to go
    - reproduce several ongoing ProtoDUNE analyses

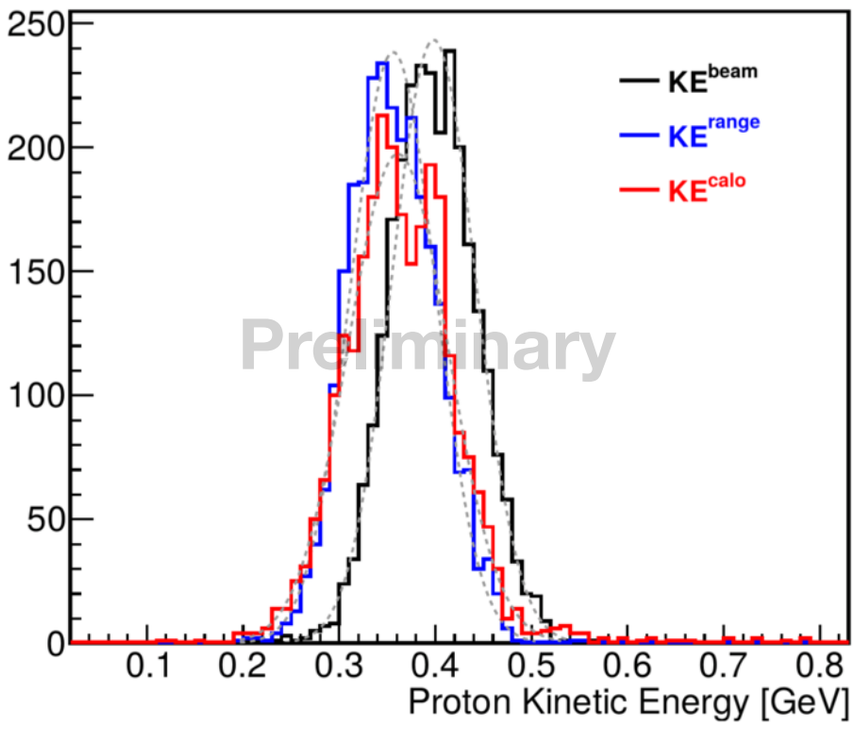
Data reduction

ejemplo para 230 sucesos

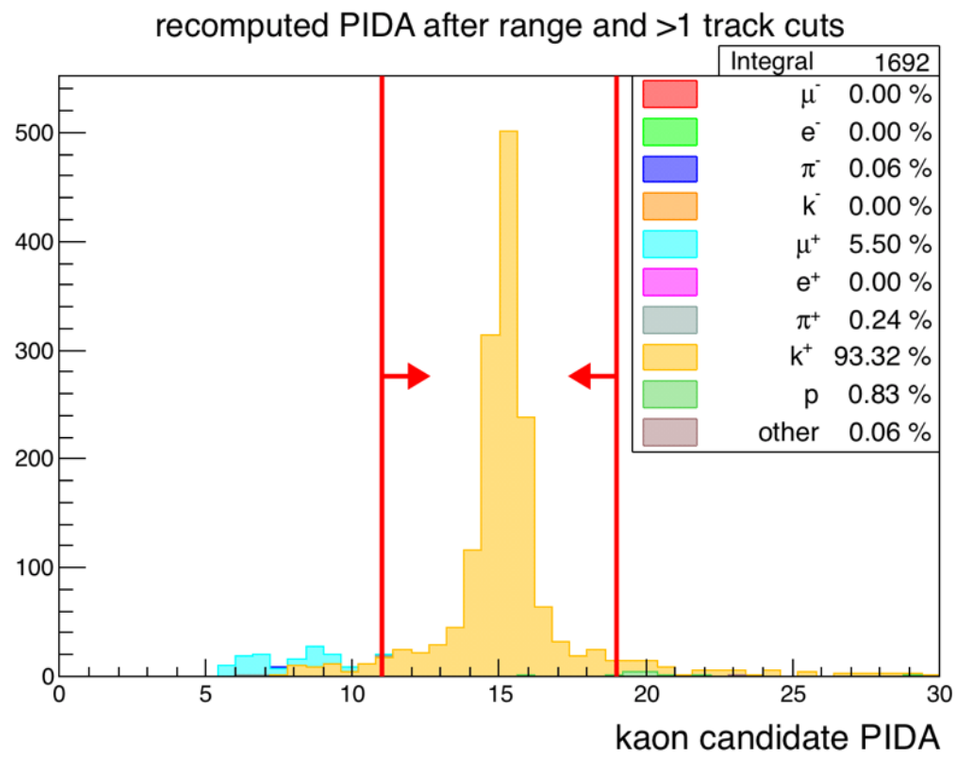
DUNE official recon output

50 GB **LArSoft file** → (**HighLAND** 2000'') → 20 MB **MiniTree** → (**HighLAND** 2'') → 1 MB **MicroTree**

2200''



linea de comando de root usando las utilidades gráficas de **HighLAND**



- the package highland/LArSoftReader does the work
- **LArSoftTreeConverter**: I've taken the code that creates the AnalTree in LArSoft (by Georgios) and tried to fill the same variables in the event model.
- The code works and produces MiniTrees, but I'm not sure everything is correctly filled.
- Trying to understand the output now

- Files in srcs/dunetpc/dune/Protodune/Analysis/

```
BeamExample
CMakeLists.txt
ProtoDUNEAnalCosmicTree.fcl
ProtoDUNEAnalCosmicTree_module.cc
ProtoDUNEAnalTree.fcl
ProtoDUNEAnalTree_module.cc
ProtoDUNEAnalTree_variables.C
ProtoDUNEBeamlineFilter.fcl
ProtoDUNEBeamlineFilter_module.cc
ProtoDUNEBeamlineReco.fcl
ProtoDUNEBeamlineReco_module.cc
ProtoDUNEBeamlineUtils.cxx
ProtoDUNEBeamlineUtils.fcl
ProtoDUNEBeamlineUtils.h
ProtoDUNEDataUtils.cxx
ProtoDUNEDataUtils.fcl
ProtoDUNEDataUtils.h
ProtoDUNEPFParticleUtils.cxx
ProtoDUNEPFParticleUtils.h
ProtoDUNEShowerUtils.cxx
ProtoDUNEShowerUtils.h
ProtoDUNESliceUtils.cxx
ProtoDUNESliceUtils.h
ProtoDUNETrackUtils.cxx
ProtoDUNETrackUtils.h
ProtoDUNETruthBeamFilter.fcl
ProtoDUNETruthBeamFilter_module.cc
ProtoDUNETruthBeamParticle.fcl
ProtoDUNETruthBeamParticle_module.cc
ProtoDUNETruthUtils.cxx
ProtoDUNETruthUtils.h
ProtoFlash_module.cc
UtilityExample.fcl
UtilityExample_module.cc
beamreco_job.fcl
protoflash.fcl
runProtoDUNEAnalCosmicTree.fcl
runProtoDUNEAnalTree.fcl
runProtoDUNEBeamlineFilter_Electron_1GeV.fcl
runProtoDUNEBeamlineFilter_Kaon_7GeV.fcl
runProtoDUNEBeamlineFilter_Pion_1GeV.fcl
runProtoDUNEBeamlineFilter_Pion_2GeV.fcl
runProtoDUNEBeamlineFilter_Pion_3GeV.fcl
runProtoDUNEBeamlineFilter_Pion_7GeV.fcl
runProtoDUNEBeamlineFilter_Proton_7GeV.fcl
runProtoDUNETruthBeamFilter.fcl
runProtoUtilExample.fcl
```

```
ProtoDUNEAnalCosmicTree_module.cc
ProtoDUNEAnalTree_module.cc
ProtoDUNEBeamlineFilter_module.cc
ProtoDUNEBeamlineReco_module.cc
ProtoDUNETruthBeamFilter_module.cc
ProtoDUNETruthBeamParticle_module.cc
ProtoFlash_module.cc
UtilityExample_module.cc
```

Utilities used by AnalTree

```
ProtoDUNEBeamlineUtils.cxx
ProtoDUNEDataUtils.cxx
ProtoDUNEPFParticleUtils.cxx
ProtoDUNEShowerUtils.cxx
ProtoDUNESliceUtils.cxx
ProtoDUNETrackUtils.cxx
ProtoDUNETruthUtils.cxx
```

ProtoDUNEAnalTree_module.cc

```
/*
// Now we want to access the output from Pandora. This comes in the form of particle flow objects (recob::PFParticle).
// The primary PFParticles are those we want to consider and these PFParticles then have a hierarchy of daughters that
// describe the whole interaction of a given primary particle
//
//                      / daughter track    / grand-daughter track
//                     /                   /
//  primary track     /                   /
//  ----------------- ---- daughter track ----- grand-daughter track
//                    \
//                    /\-
//                    /\\-- daughter shower
//
// The above primary PFParticle will have links to three daughter particles, two track-like and one shower-like
*/
```

```
fPandoraBeam = tfs->make<TTree>("PandoraBeam", "Beam events reconstructed with Pandora");
fPandoraBeam->Branch("run",                          &fRun,                          "run/I");
fPandoraBeam->Branch("subrun",                       &fSubRun,                       "subrun/I");
fPandoraBeam->Branch("event",                        &fevent,                        "event/I");
fPandoraBeam->Branch("timestamp",                    &fTimeStamp,                    "timestamp/D");
fPandoraBeam->Branch("Nactivefembs",                 &fNactivefembs,                 "Nactivefembs[5]/I");

fPandoraBeam->Branch("beamtrigger",                  &fbeamtrigger,                  "beamtrigger/I");
fPandoraBeam->Branch("tof",                          &ftof,                          "tof/D");
fPandoraBeam->Branch("cerenkovStatus",               &fcerenkovStatus,               "cerenkovStatus[2]/I");
fPandoraBeam->Branch("cerenkovTime",                 &fcerenkovTime,                 "cerenkovTime[2]/D");
fPandoraBeam->Branch("cerenkovPressure",             &fcerenkovPressure,             "cerenkovPressure[2]/D");
fPandoraBeam->Branch("beamtrackMomentum",            &fbeamtrackMomentum,            "beamtrackMomentum/D");
fPandoraBeam->Branch("beamtrackEnergy",              &fbeamtrackEnergy,              "beamtrackEnergy/D");
fPandoraBeam->Branch("beamtrackPos",                 &fbeamtrackPos,                 "beamtrackPos[3]/D");
fPandoraBeam->Branch("beamtrackDir",                 &fbeamtrackDir,                 "beamtrackDir[3]/D");
fPandoraBeam->Branch("beamtrackTime",                &fbeamtrackTime,                "beamtrackTime/D");
fPandoraBeam->Branch("beamtrackPdg",                 &fbeamtrackPdg,                 "beamtrackPdg/I");
fPandoraBeam->Branch("beamtrackID",                  &fbeamtrackID,                  "beamtrackID/I");

fPandoraBeam->Branch("primaryVertex",                &fprimaryVertex,                "primaryVertex[3]/D");
fPandoraBeam->Branch("primaryIstrack",               &fprimaryIstrack,               "primaryIstrack/I");
fPandoraBeam->Branch("primaryIsshower",              &fprimaryIsshower,              "primaryIsshower/I");
fPandoraBeam->Branch("primaryBDTScore",              &fprimaryBDTScore,              "primaryBDTScore/D");
fPandoraBeam->Branch("primaryNHits",                 &fprimaryNHits,                 "fprimaryNHits/I");
fPandoraBeam->Branch("primaryTheta",                 &fprimaryTheta,                 "primaryTheta/D");
fPandoraBeam->Branch("primaryPhi",                   &fprimaryPhi,                   "primaryPhi/D");
fPandoraBeam->Branch("primaryLength",                &fprimaryLength,                "primaryLength/D");
fPandoraBeam->Branch("primaryMomentum",              &fprimaryMomentum,              "primaryMomentum/D");
fPandoraBeam->Branch("primaryEndMomentum",           &fprimaryEndMomentum,           "primaryEndMomentum/D");
fPandoraBeam->Branch("primaryEndPosition",           &fprimaryEndPosition,           "primaryEndPosition[3]/D");
fPandoraBeam->Branch("primaryStartPosition",         &fprimaryStartPosition,         "primaryStartPosition[3]/D");
fPandoraBeam->Branch("primaryEndDirection",          &fprimaryEndDirection,          "primaryEndDirection[3]/D");
fPandoraBeam->Branch("primaryStartDirection",        &fprimaryStartDirection,        "primaryStartDirection[3]/D");
fPandoraBeam->Branch("primaryOpeningAngle",          &fprimaryOpeningAngle,          "primaryOpeningAngle/D");
fPandoraBeam->Branch("primaryID",                    &fprimaryID,                    "primaryID/I");
fPandoraBeam->Branch("primaryShowerBestPlane",       &fprimaryShowerBestPlane,       "primaryShowerBestPlane/I");
fPandoraBeam->Branch("primaryShowerEnergy",          &fprimaryShowerEnergy,          "primaryShowerEnergy/I");
fPandoraBeam->Branch("primaryShowerMIPEnergy",       &fprimaryShowerMIPEnergy,       "primaryShowerMIPEnergy/I");
fPandoraBeam->Branch("primaryShowerdEdx",            &fprimaryShowerdEdx,            "primaryShowerdEdx/I");
fPandoraBeam->Branch("primaryMomentumByRangeProton", &fprimaryMomentumByRangeProton, "primaryMomentumByRangeProton/D");
fPandoraBeam->Branch("primaryMomentumByRangeMuon",   &fprimaryMomentumByRangeMuon,   "primaryMomentumByRangeMuon/D");
fPandoraBeam->Branch("primaryKineticEnergy",         &fprimaryKineticEnergy,         "primaryKineticEnergy[3]/D");
fPandoraBeam->Branch("primaryRange",                 &fprimaryRange,                 "primaryRange[3]/D");
fPandoraBeam->Branch("primaryTrkPitchC",             &fprimaryTrkPitchC,             "primaryTrkPitchC[3]/D");
fPandoraBeam->Branch("primaryT0",                    &fprimaryT0,                    "primaryT0/D");
fPandoraBeam->Branch("primaryPID_Pdg",               &fprimaryPID_Pdg,               "primaryPID_Pdg[3]/I");
fPandoraBeam->Branch("primaryPID_Ndf",               &fprimaryPID_Ndf,               "primaryPID_Ndf[3]/I");
fPandoraBeam->Branch("primaryPID_MinChi2",           &fprimaryPID_MinChi2,           "primaryPID_MinChi2[3]/D");
fPandoraBeam->Branch("primaryPID_DeltaChi2",         &fprimaryPID_DeltaChi2,         "primaryPID_DeltaChi2[3]/D");
fPandoraBeam->Branch("primaryPID_Chi2Proton",        &fprimaryPID_Chi2Proton,        "primaryPID_Chi2Proton[3]/D");
fPandoraBeam->Branch("primaryPID_Chi2Kaon",          &fprimaryPID_Chi2Kaon,          "primaryPID_Chi2Kaon[3]/D");
fPandoraBeam->Branch("primaryPID_Chi2Pion",          &fprimaryPID_Chi2Pion,          "primaryPID_Chi2Pion[3]/D");
```

beam detectors info

primary track

+ more variables ….

secondary tracks

```
fPandoraBeam->Branch("NDAUGHTERS",                  &fNDAUGHTERS,                  "NDAUGHTERS/I");
fPandoraBeam->Branch("daughterVertex",              &fdaughterVertex,              "daughterVertex[3]/D");
fPandoraBeam->Branch("daughterIstrack",             &fdaughterIstrack,             "daughterIstrack[NDAUGHTERS]/I");
fPandoraBeam->Branch("daughterIsshower",            &fdaughterIsshower,            "daughterIsshower[NDAUGHTERS]/I");
fPandoraBeam->Branch("daughterNHits",               &fdaughterNHits,               "daughterNHits[NDAUGHTERS]/I");
fPandoraBeam->Branch("daughterTheta",               &fdaughterTheta,               "daughterTheta[NDAUGHTERS]/D");
fPandoraBeam->Branch("daughterPhi",                 &fdaughterPhi,                 "daughterPhi[NDAUGHTERS]/D");
fPandoraBeam->Branch("daughterLength",              &fdaughterLength,              "daughterLength[NDAUGHTERS]/D");
fPandoraBeam->Branch("daughterMomentum",            &fdaughterMomentum,            "daughterMomentum[NDAUGHTERS]/D");
fPandoraBeam->Branch("daughterEndMomentum",         &fdaughterEndMomentum,         "daughterEndMomentum[NDAUGHTERS]/D");
fPandoraBeam->Branch("daughterEndPosition",         &fdaughterEndPosition,         "daughterEndPosition[NDAUGHTERS][3]/D");
fPandoraBeam->Branch("daughterStartPosition",       &fdaughterStartPosition,       "daughterStartPosition[NDAUGHTERS][3]/D");
fPandoraBeam->Branch("daughterStartDirection",      &fdaughterStartDirection,      "daughterStartDirection[NDAUGHTERS][3]/D");
fPandoraBeam->Branch("daughterEndDirection",        &fdaughterEndDirection,        "daughterEndDirection[NDAUGHTERS][3]/D");
fPandoraBeam->Branch("daughterOpeningAngle",        &fdaughterOpeningAngle,        "daughterOpeningAngle[NDAUGHTERS]/D");
fPandoraBeam->Branch("daughterShowerBestPlane",     &fdaughterShowerBestPlane,     "daughterShowerBestPlane[NDAUGHTERS]/D");
fPandoraBeam->Branch("daughterShowerEnergy",        &fdaughterShowerEnergy,        "daughterShowerEnergy[NDAUGHTERS]/D");
fPandoraBeam->Branch("daughterShowerMIPEnergy",     &fdaughterShowerMIPEnergy,     "daughterShowerMIPEnergy[NDAUGHTERS]/D");
fPandoraBeam->Branch("daughterShowerdEdx",          &fdaughterShowerdEdx,          "daughterShowerdEdx[NDAUGHTERS]/D");
fPandoraBeam->Branch("daughterMomentumByRangeProton", &fdaughterMomentumByRangeProton, "daughterMomentumByRangeProton[NDAUGHTERS]/D");
fPandoraBeam->Branch("daughterMomentumByRangeMuon", &fdaughterMomentumByRangeMuon, "daughterMomentumByRangeMuon[NDAUGHTERS]/D");
fPandoraBeam->Branch("daughterKineticEnergy",       &fdaughterKineticEnergy,       "daughterKineticEnergy[NDAUGHTERS][3]/D");
fPandoraBeam->Branch("daughterRange",               &fdaughterRange,               "daughterRange[NDAUGHTERS][3]/D");
fPandoraBeam->Branch("daughterTrkPitchC",           &fdaughterTrkPitchC,           "daughterTrkPitchC[NDAUGHTERS][3]/D");
fPandoraBeam->Branch("daughterID",                  &fdaughterID,                  "daughterID[NDAUGHTERS]/I");
fPandoraBeam->Branch("daughterT0",                  &fdaughterT0,                  "daughterT0[NDAUGHTERS]/D");
```

+ more variables ….

tertiary tracks

```
fPandoraBeam->Branch("granddaughterVertex",                &fgranddaughterVertex,                "granddaughterVertex[NDAUGHTERS][3]/D");
fPandoraBeam->Branch("NGRANDDAUGHTERS",                    &fNGRANDDAUGHTERS,                    "NGRANDDAUGHTERS/I");
fPandoraBeam->Branch("granddaughterIstrack",                    &fgranddaughterIstrack,                    "granddaughterIstrack[NGRANDDAUGHTERS]/I");
fPandoraBeam->Branch("granddaughterIsshower",                   &fgranddaughterIsshower,                   "granddaughterIsshower[NGRANDDAUGHTERS]/I");
fPandoraBeam->Branch("granddaughterNHits",                      &fgranddaughterNHits,                      "granddaughterNHits[NGRANDDAUGHTERS]/I");
fPandoraBeam->Branch("granddaughterTheta",                      &fgranddaughterTheta,                      "granddaughterTheta[NGRANDDAUGHTERS]/D");
fPandoraBeam->Branch("granddaughterPhi",                        &fgranddaughterPhi,                        "granddaughterPhi[NGRANDDAUGHTERS]/D");
fPandoraBeam->Branch("granddaughterLength",                     &fgranddaughterLength,                     "granddaughterLength[NGRANDDAUGHTERS]/D");
fPandoraBeam->Branch("granddaughterMomentum",                   &fgranddaughterMomentum,                   "granddaughterMomentum[NGRANDDAUGHTERS]/D");
fPandoraBeam->Branch("granddaughterEndMomentum",                &fgranddaughterEndMomentum,                "granddaughterEndMomentum[NGRANDDAUGHTERS]/D");
fPandoraBeam->Branch("granddaughterEndPosition",                &fgranddaughterEndPosition,                "granddaughterEndPosition[NGRANDDAUGHTERS][3]/D");
fPandoraBeam->Branch("granddaughterStartPosition",              &fgranddaughterStartPosition,              "granddaughterStartPosition[NGRANDDAUGHTERS][3]/D");
fPandoraBeam->Branch("granddaughterStartDirection",             &fgranddaughterStartDirection,             "granddaughterStartDirection[NGRANDDAUGHTERS][3]/D");
fPandoraBeam->Branch("granddaughterEndDirection",               &fgranddaughterEndDirection,               "granddaughterEndDirection[NGRANDDAUGHTERS][3]/D");
fPandoraBeam->Branch("granddaughterOpeningAngle",               &fgranddaughterOpeningAngle,               "granddaughterOpeningAngle[NGRANDDAUGHTERS]/D");
fPandoraBeam->Branch("granddaughterShowerBestPlane",            &fgranddaughterShowerBestPlane,            "granddaughterShowerBestPlane[NGRANDDAUGHTERS]/D");
fPandoraBeam->Branch("granddaughterShowerEnergy",               &fgranddaughterShowerEnergy,               "granddaughterShowerEnergy[NGRANDDAUGHTERS]/D");
fPandoraBeam->Branch("granddaughterShowerMIPEnergy",            &fgranddaughterShowerMIPEnergy,            "granddaughterShowerMIPEnergy[NGRANDDAUGHTERS]/D");
fPandoraBeam->Branch("granddaughterShowerdEdx",                 &fgranddaughterShowerdEdx,                 "granddaughterShowerdEdx[NGRANDDAUGHTERS]/D");
fPandoraBeam->Branch("granddaughterMomentumByRangeProton",      &fgranddaughterMomentumByRangeProton,      "granddaughterMomentumByRangeProton[NGRANDDAUGHTERS]/D");
fPandoraBeam->Branch("granddaughterMomentumByRangeMuon",        &fgranddaughterMomentumByRangeMuon,        "granddaughterMomentumByRangeMuon[NGRANDDAUGHTERS]/D");
fPandoraBeam->Branch("granddaughterKineticEnergy",              &fgranddaughterKineticEnergy,              "granddaughterKineticEnergy[NGRANDDAUGHTERS][3]/D");
fPandoraBeam->Branch("granddaughterRange",                      &fgranddaughterRange,                      "granddaughterRange[NGRANDDAUGHTERS][3]/D");
fPandoraBeam->Branch("granddaughterTrkPitchC",                  &fgranddaughterTrkPitchC,                  "granddaughterTrkPitchC[NGRANDDAUGHTERS][3]/D");
fPandoraBeam->Branch("granddaughterID",                         &fgranddaughterID,                         "granddaughterID[NGRANDDAUGHTERS]/I");
fPandoraBeam->Branch("granddaughterMotherID",                   &fgranddaughterMotherID,                   "granddaughterMotherID[NGRANDDAUGHTERS]/I");
fPandoraBeam->Branch("granddaughterT0",                         &fgranddaughterT0,                         "granddaughterT0[NGRANDDAUGHTERS]/D");
```

+ more variables ….

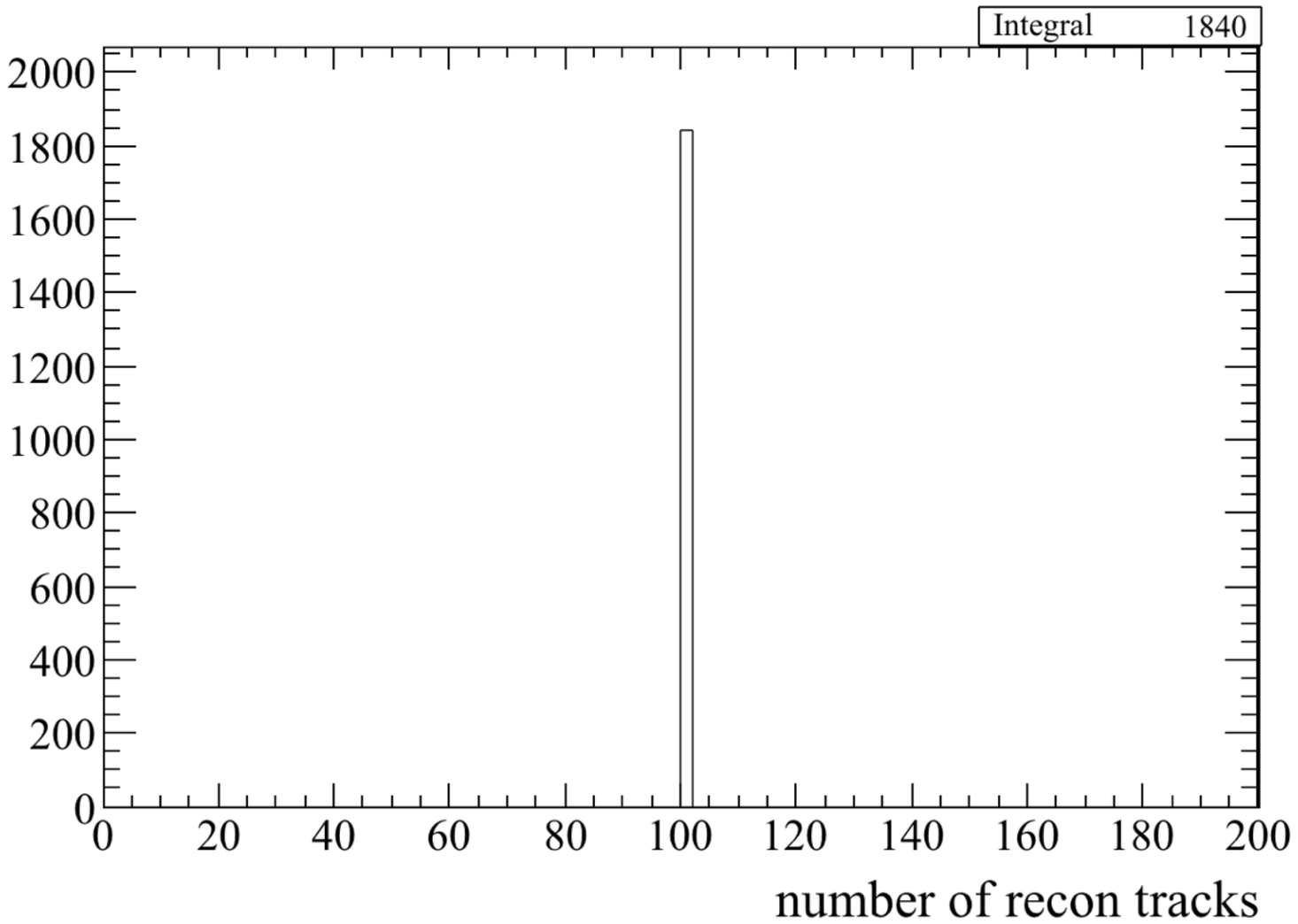
Creating MiniTrees

- I’ve managed to create MiniTrees for 1840 events
  - A MiniTree for each LArSoft file (with 10 events) because in jobs of 10 most jobs are killed
  - I need to figure out how to submit batch jobs at Fermilab
    - I learned how to use the **jobsub** command, but the code needs to be in a special place. Any clue ?

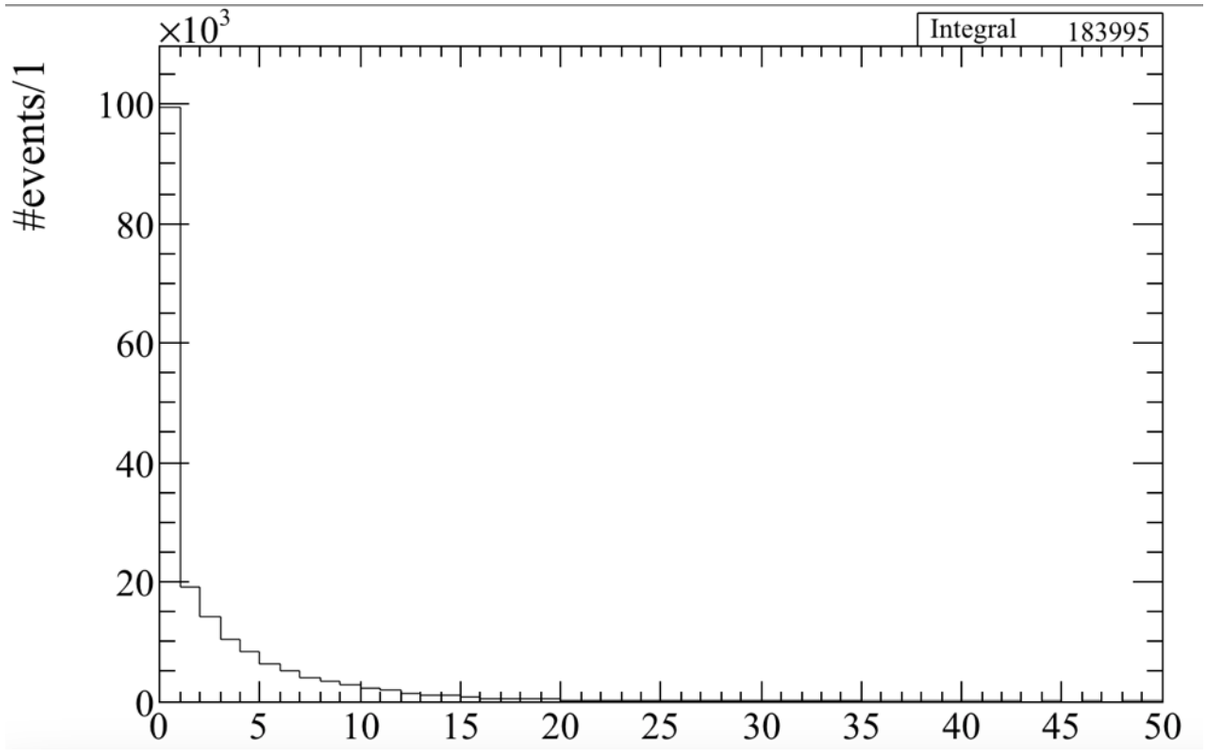
http://dune-data.fnal.gov/mc/mcc11/jhugondatasets.html

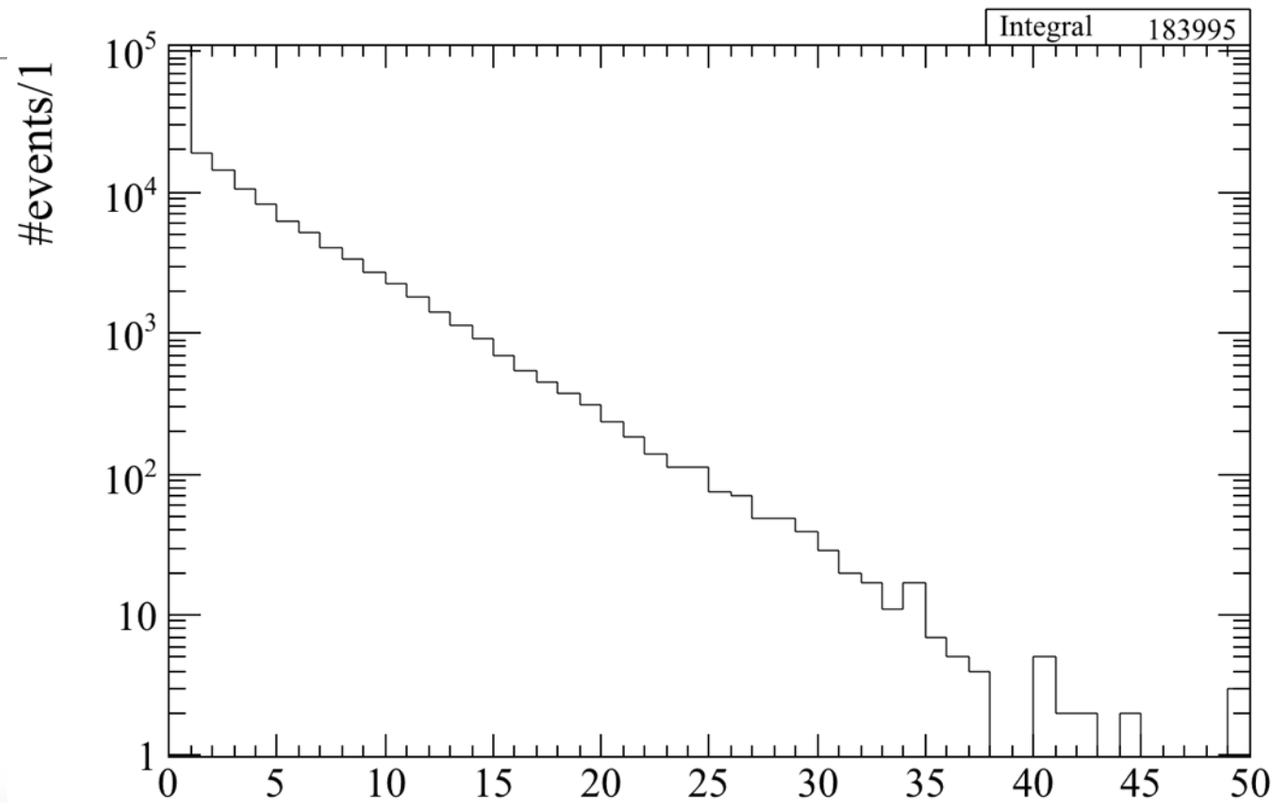
Multiplicity

- The number of recon tracks is always above 100. I'm only saving the first 100. Must use a better criterium

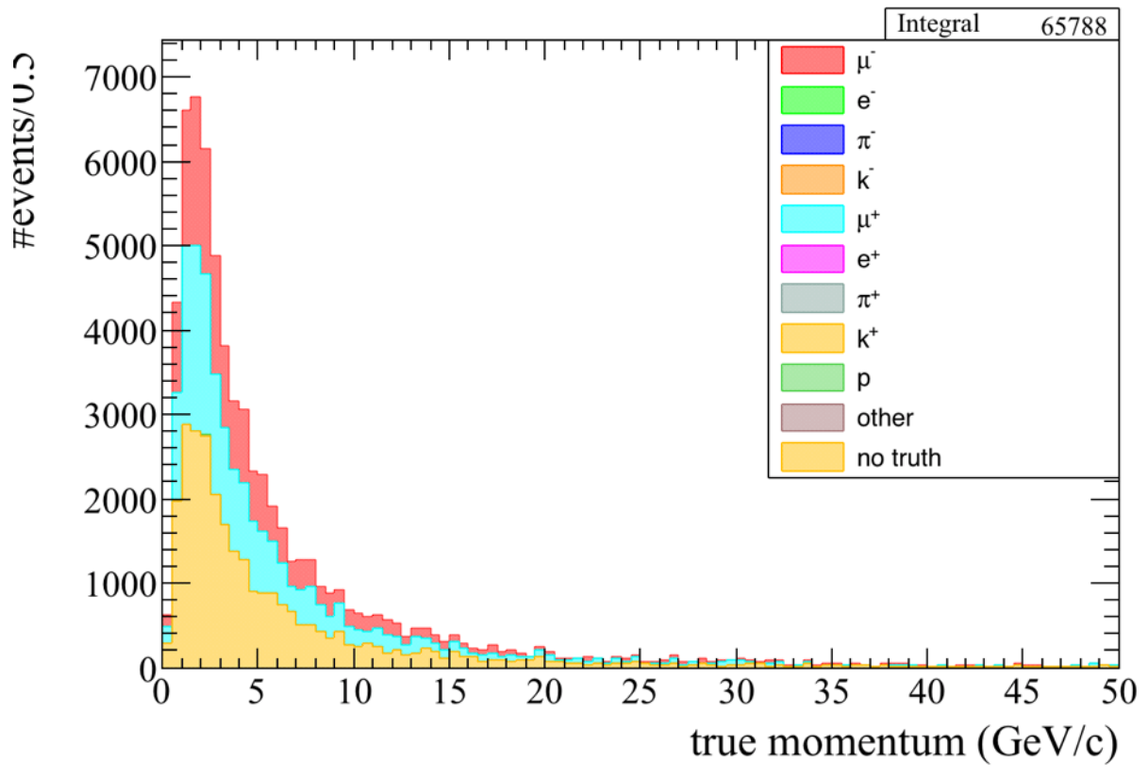


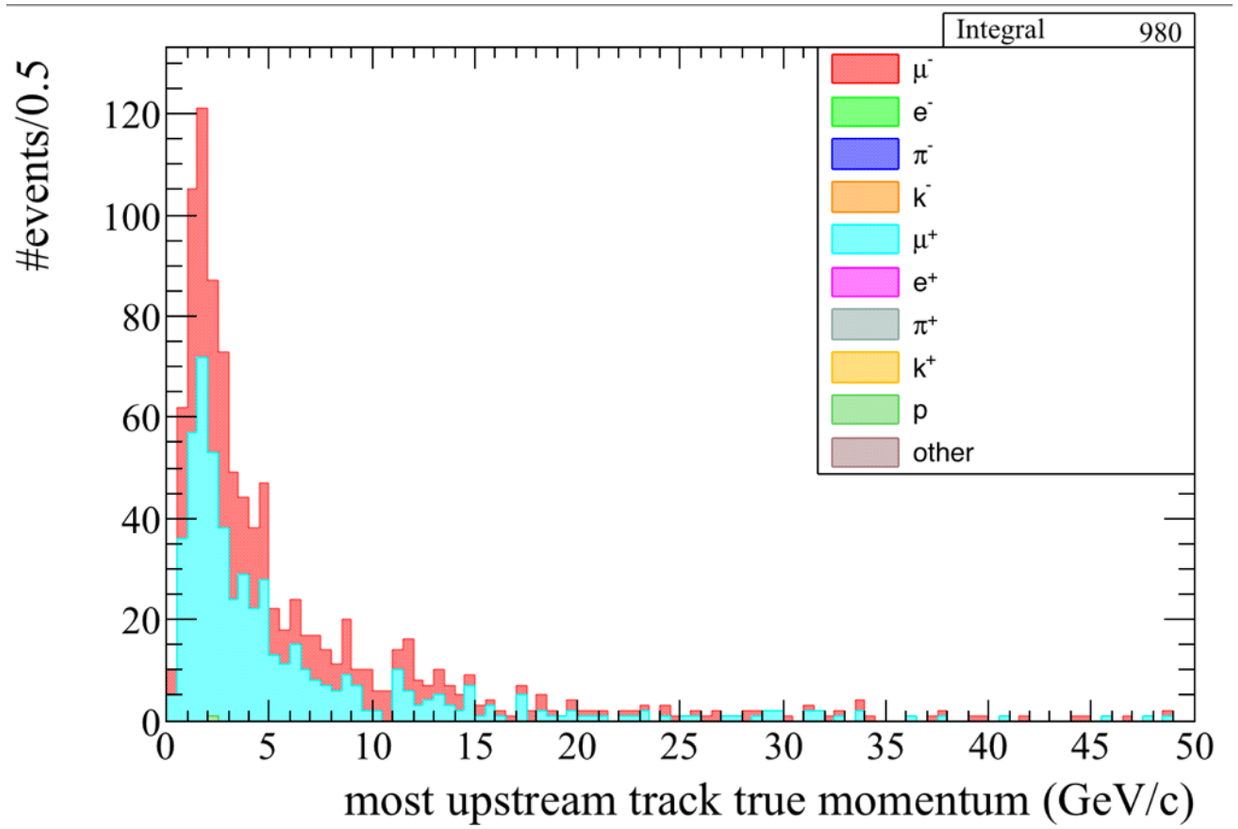
Number of daughters

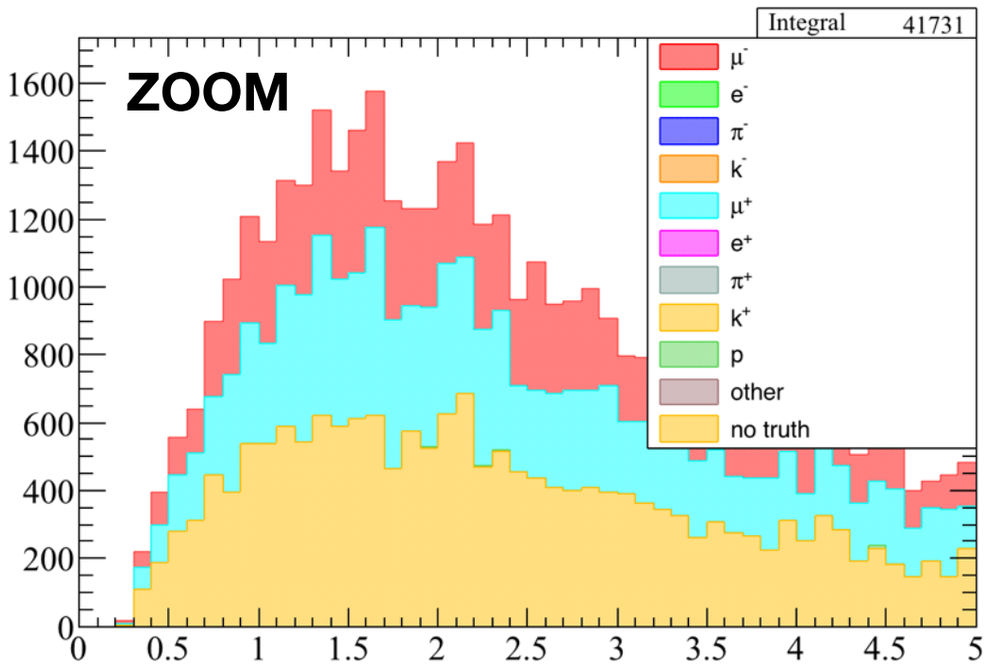




True momentum





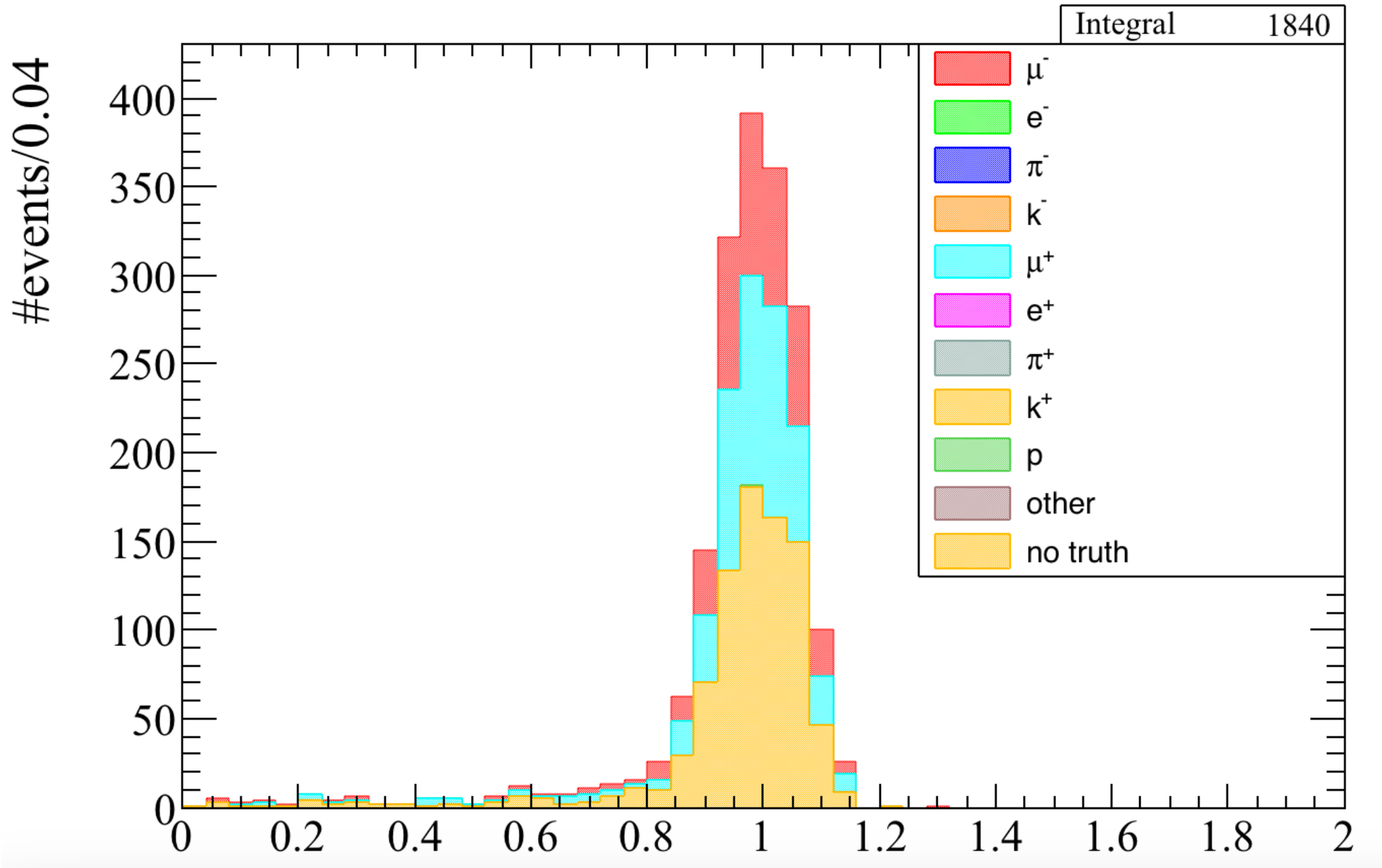


The beam particle is not there !!!!

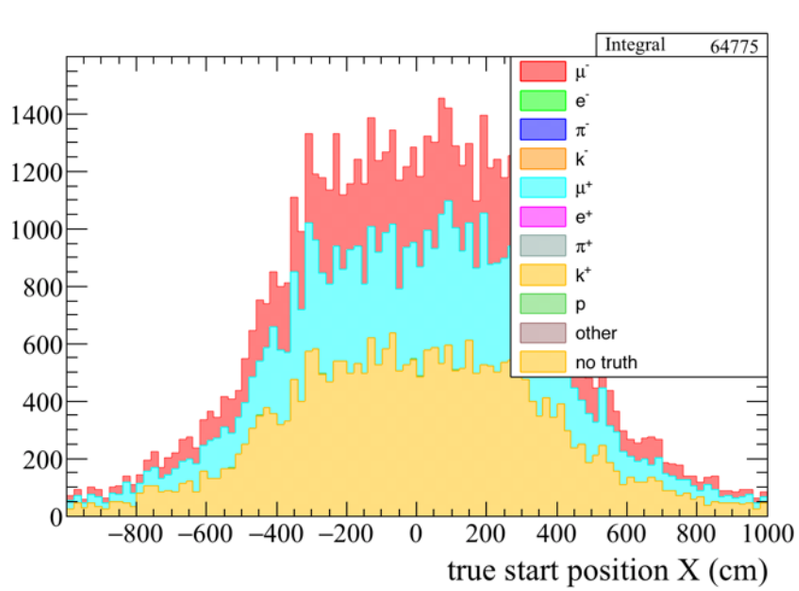
I think this is because it is not on the first 100 that I'm saving

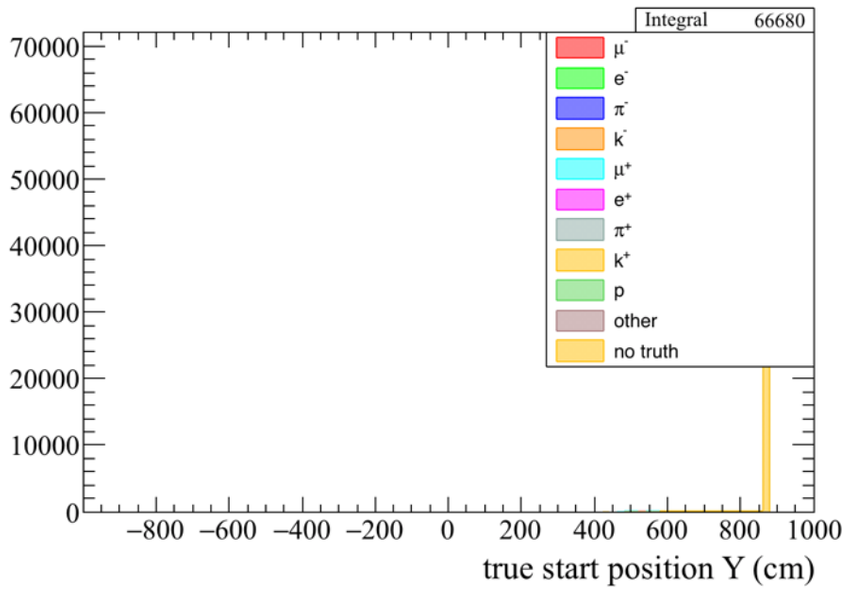
Beam particle true momentum

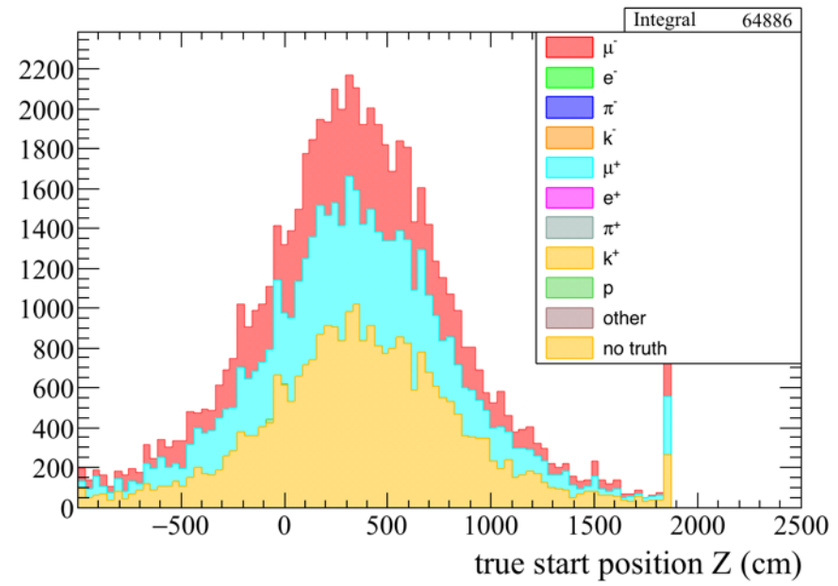
- Save as a property of AnaBeam inside AnaSpill

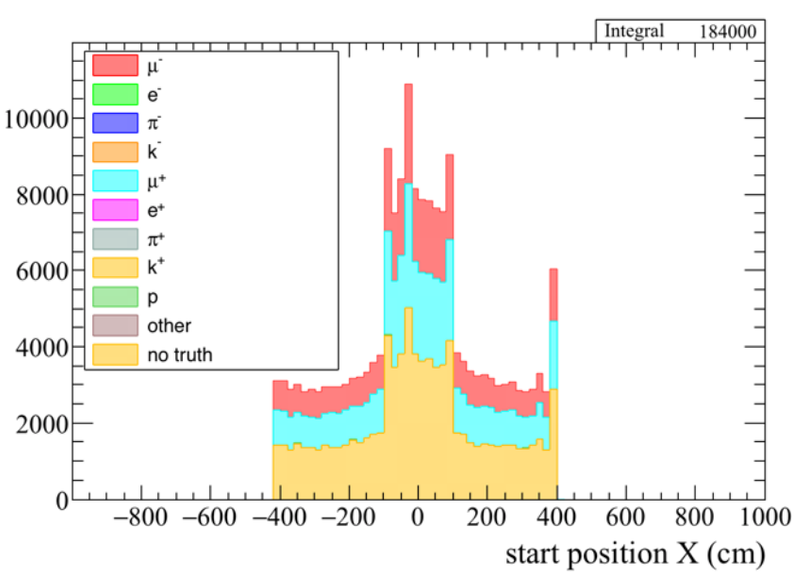


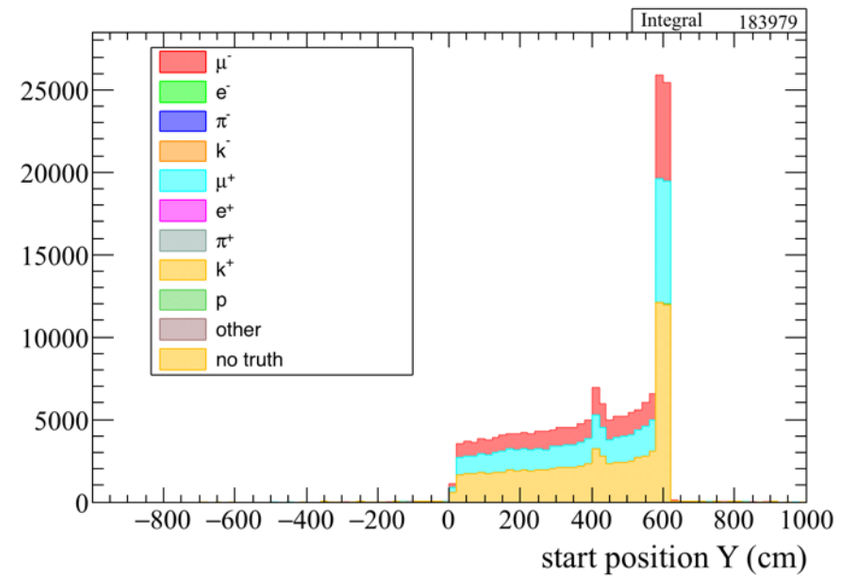
Position

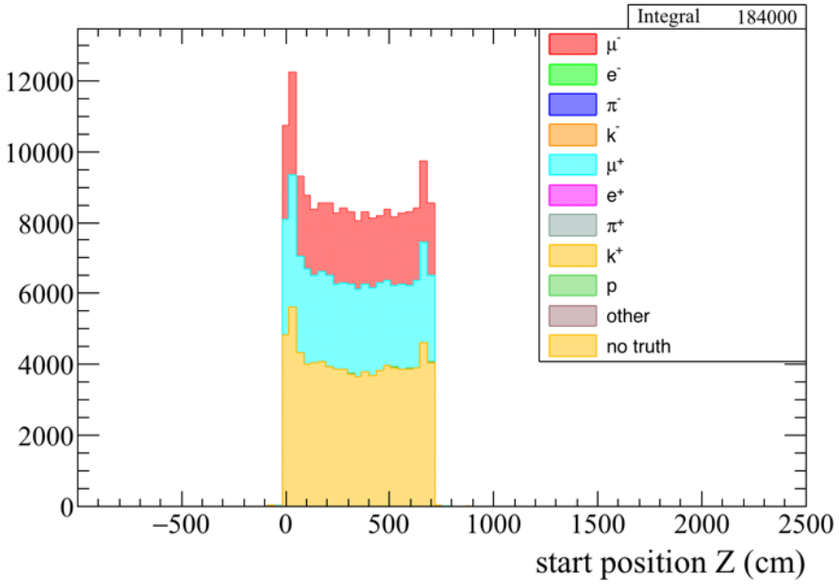

Integral 64775
μ⁻
e⁻
π⁻
k⁻
μ⁺
e⁺
π⁺
k⁺
p
other
no truth
true start position X (cm)


Integral 66680
μ⁻
e⁻
π⁻
k⁻
μ⁺
e⁺
π⁺
k⁺
p
other
no truth
true start position Y (cm)


Integral 64886
μ⁻
e⁻
π⁻
k⁻
μ⁺
e⁺
π⁺
k⁺
p
other
no truth
true start position Z (cm)

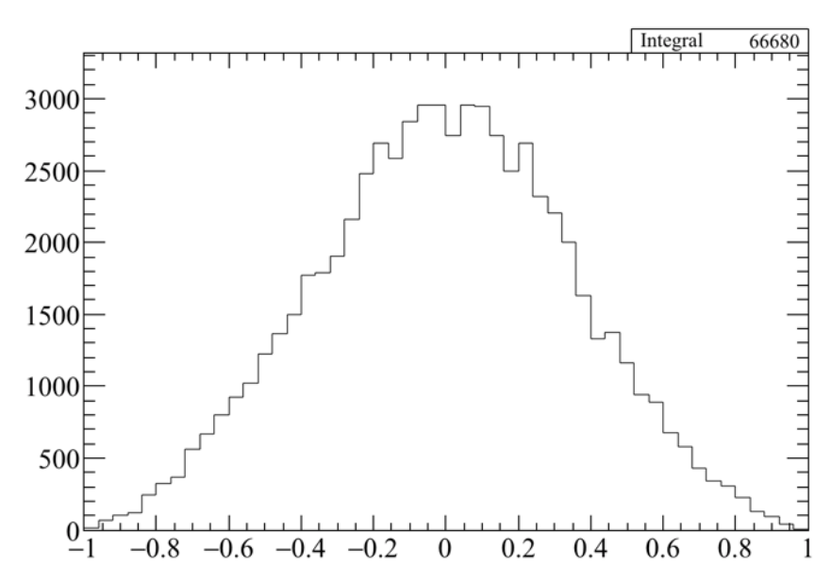

Integral 184000
μ⁻
e⁻
π⁻
k⁻
μ⁺
e⁺
π⁺
k⁺
p
other
no truth
start position X (cm)

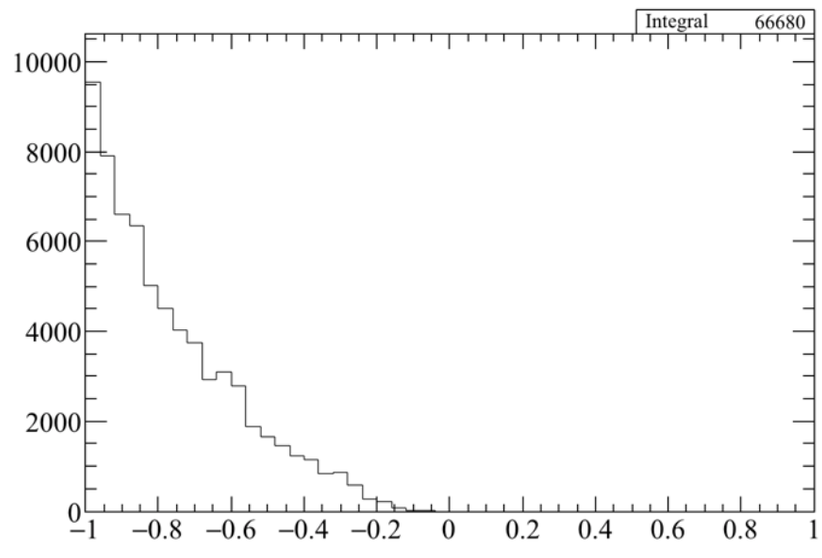

Integral 183979
μ⁻
e⁻
π⁻
k⁻
μ⁺
e⁺
π⁺
k⁺
p
other
no truth
start position Y (cm)

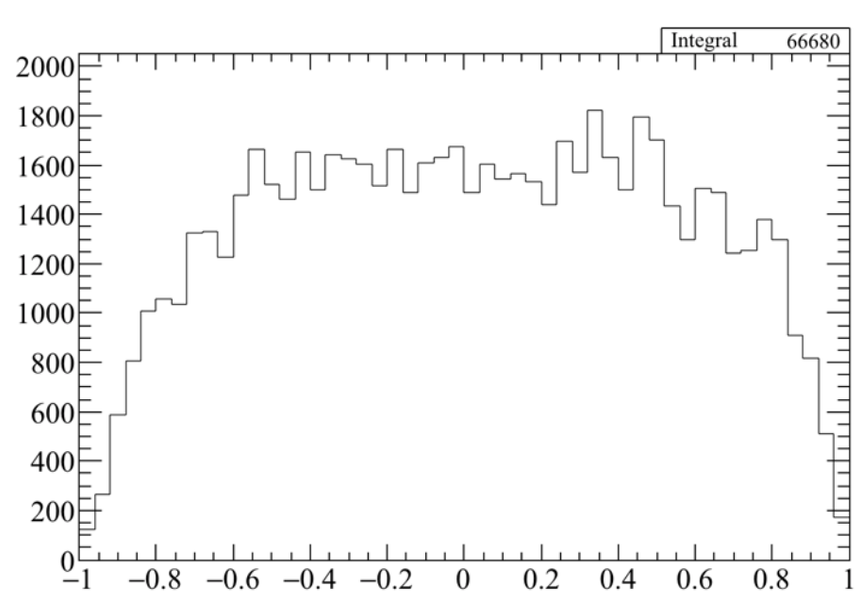
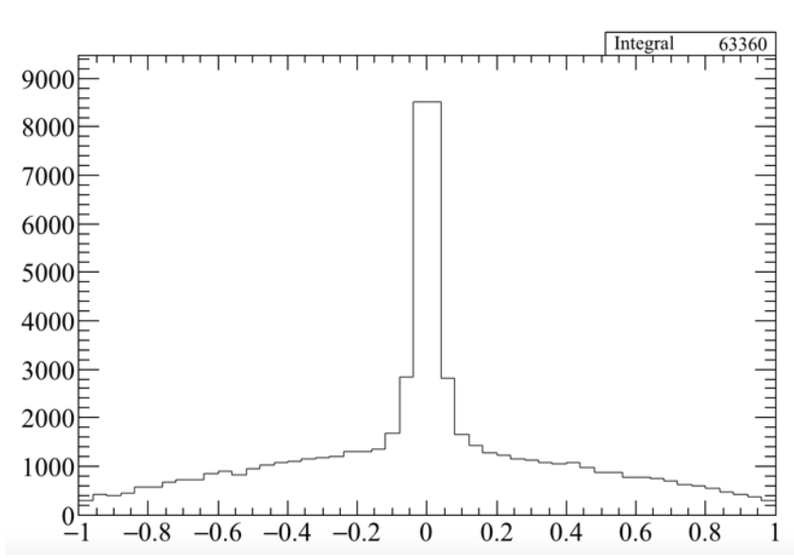
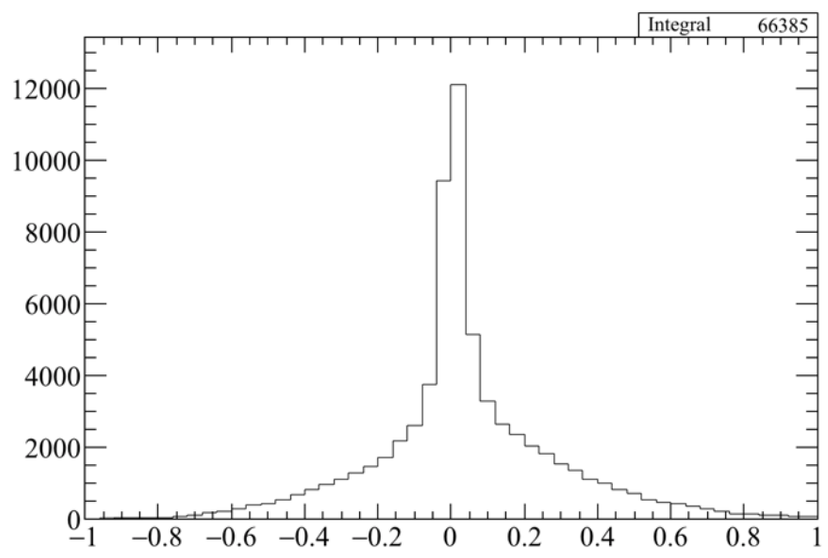
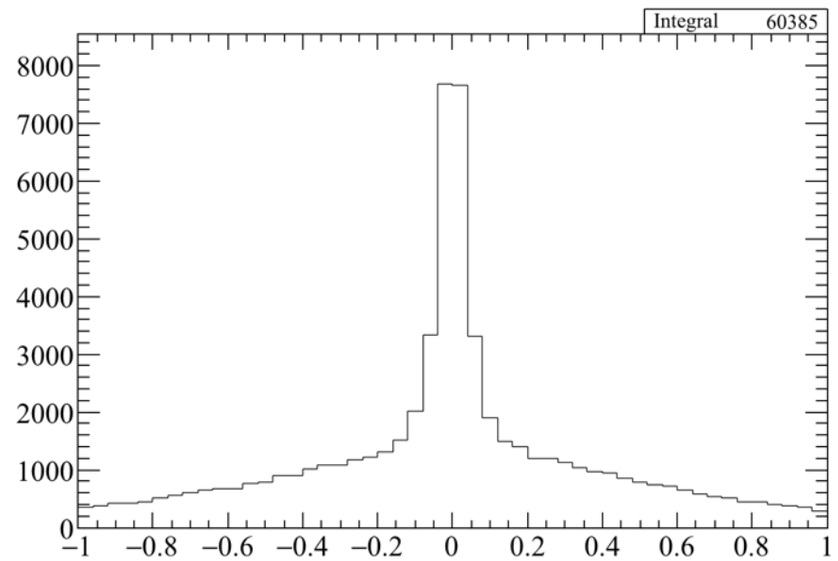

Integral 184000
μ⁻
e⁻
π⁻
k⁻
μ⁺
e⁺
π⁺
k⁺
p
other
no truth
start position Z (cm)

Direction

True start direction







Rec-True direction