Study of lepton ID

Changhua Hao, Ligang Xia

Nanjing University

Update on MC samples

- Latest mater (**yesterday**) + Chenguang's TofRecAlg package and the CyberPFA algorithm
- particles: electron, muon and pion
- momentum: 1~ 10 GeV and 1000 events for each momentum (7GeV job is not done yet!)
- angle: 45 degrees
- Is most recent update from Fangyi merged in the master?

Update in CyberPFA

- **A new MR !153 in CEPCSW: harmonize with standalone analysis.**
  - ECAL and HCAL digitization: add some effects from electroncs.
    - Energy resolution may go worse.
  - Tracking: improve the precision in G4 simulation [MR !159 by Zhihao Li and Chengdong Fu]
    - Track momentum bias is fixed.
  - PFA:
    - Add a BDT-based track cleaning.
    - Add separate scale constant for charged/neutral ECAL and HCAL constant.

```
+ CyberPFAlg.TrackIDFile = "/cvmfs/cepcsw.ihep.ac.cn/prototype/releases/data/latest/CEPCSWData/offline-data/Reconstruction/CyberPFA_trackID/TrkID_BDT_BDTG.weights.xml"
+ CyberPFAlg.TrackIDMethod = "BDTG"
+ CyberPFAlg.EcalChargedCalib = 1.26
+ CyberPFAlg.HcalChargedCalib = 4.0
+ CyberPFAlg.EcalNeutralCalib = 1.0
+ CyberPFAlg.HcalNeutralCalib = 4.0
```

BDT weight file might not in /cvmfs/ yet, can be found in /afs/ihep.ac.cn/users/g/guofy/TrkID_BDT_BDTG.weights.xml

  - Jet clustering in GenMatch:
    - add a branch: barrelRatio = $N_{MC}^{|\cos\theta|<0.85} / N_{MC}^{all}$.
  - A set of scripts in Reconstruction/RecPFACyber/script/
  - *Detailed performance will be updated this week.*

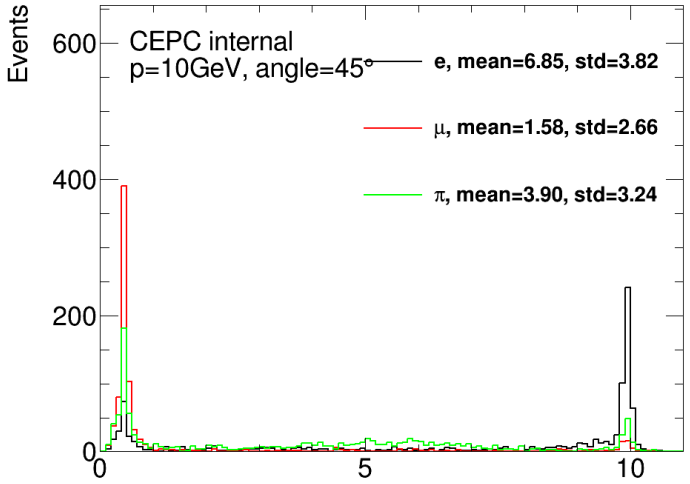
Issue1: No track momentum for 1 GeV particles

```
root [3] RecClusters -> Scan("EcalClus_Escale:EcalClus_ptrk")
*******************************************
*    Row   * Instance * EcalClus_ * EcalClus_ *
*******************************************
*        0 *        0 * 0.0337420 *       -99 *
*        0 *        1 * 0.1244335 *       -99 *
*        1 *        0 * 0.0018956 *       -99 *
*        2 *        0 * 0.7157256 *       -99 *
*        3 *        0 *           *           *
*        4 *        0 *           *           *
*        5 *        0 *           *           *
*        6 *        0 * 0.2401485 *       -99 *
*        7 *        0 * 0.9447402 *       -99 *
*        8 *        0 *           *           *
*        9 *        0 *           *           *
*       10 *        0 *           *           *
*       11 *        0 * 0.0054692 *       -99 *
*       11 *        1 * 0.0045299 *       -99 *
*       12 *        0 *           *           *
*       13 *        0 *           *           *
*       14 *        0 *           *           *
*       15 *        0 *           *           *
*       16 *        0 *           *           *
*       17 *        0 *           *           *
*       18 *        0 * 0.1946679 *       -99 *
*       18 *        1 * 0.0680412 *       -99 *
*       19 *        0 * 0.5789478 *       -99 *
*       20 *        0 *           *           *
*       21 *        0 *           *           *
```

- We find “EcalClus_ptrk” is -99 in all 1000 events.
- There are two possibilities.
  - **No associated track for the cluster at all**
  - More than 1 associated tracks
- where are the codes about track association?

```
if(m_EcalClusterCol[icl]->getAssociatedTracks().size()==1){
  const Track* trk = m_EcalClusterCol[icl]->getAssociatedTracks()[0];
  m_EcalClus_pTrk.push_back(trk->getMomentum());

  std::vector<TrackState> AllTrackStates = trk->getAllTrackStates();
  for(int istate=0; istate<AllTrackStates.size(); istate++){
    m_EcalClus_trk_tag.push_back(icl);
    m_EcalClus_trk_d0.push_back(AllTrackStates[istate].D0);
    m_EcalClus_trk_z0.push_back(AllTrackStates[istate].Z0);
    m_EcalClus_trk_phi.push_back(AllTrackStates[istate].phi0);
    m_EcalClus_trk_tanL.push_back( AllTrackStates[istate].tanLambda );
    m_EcalClus_trk_kappa.push_back( AllTrackStates[istate].Kappa);
    m_EcalClus_trk_omega.push_back( AllTrackStates[istate].Omega );
    m_EcalClus_trk_location.push_back( AllTrackStates[istate].location );
  }

}
else
  m_EcalClus_pTrk.push_back(-99);
```

Issue2: Truth Matching




```
if(m_EcalClusterCol[icl]->getAssociatedTracks().size()==1){
  const Track* trk = m_EcalClusterCol[icl]->getAssociatedTracks()[0];
  m_EcalClus_pTrk.push_back(trk->getMomentum());

  std::vector<TrackState> AllTrackStates = trk->getAllTrackStates();
  for(int istate=0; istate<AllTrackStates.size(); istate++){
    m_EcalClus_trk_tag.push_back(icl);
    m_EcalClus_trk_d0.push_back(AllTrackStates[istate].D0);
    m_EcalClus_trk_z0.push_back(AllTrackStates[istate].Z0);
    m_EcalClus_trk_phi.push_back(AllTrackStates[istate].phi0);
    m_EcalClus_trk_tanL.push_back( AllTrackStates[istate].tanLambda );
    m_EcalClus_trk_kappa.push_back( AllTrackStates[istate].Kappa);
    m_EcalClus_trk_omega.push_back( AllTrackStates[istate].Omega );
    m_EcalClus_trk_location.push_back( AllTrackStates[istate].location );
  }

}
else
  m_EcalClus_pTrk.push_back(-99);
```

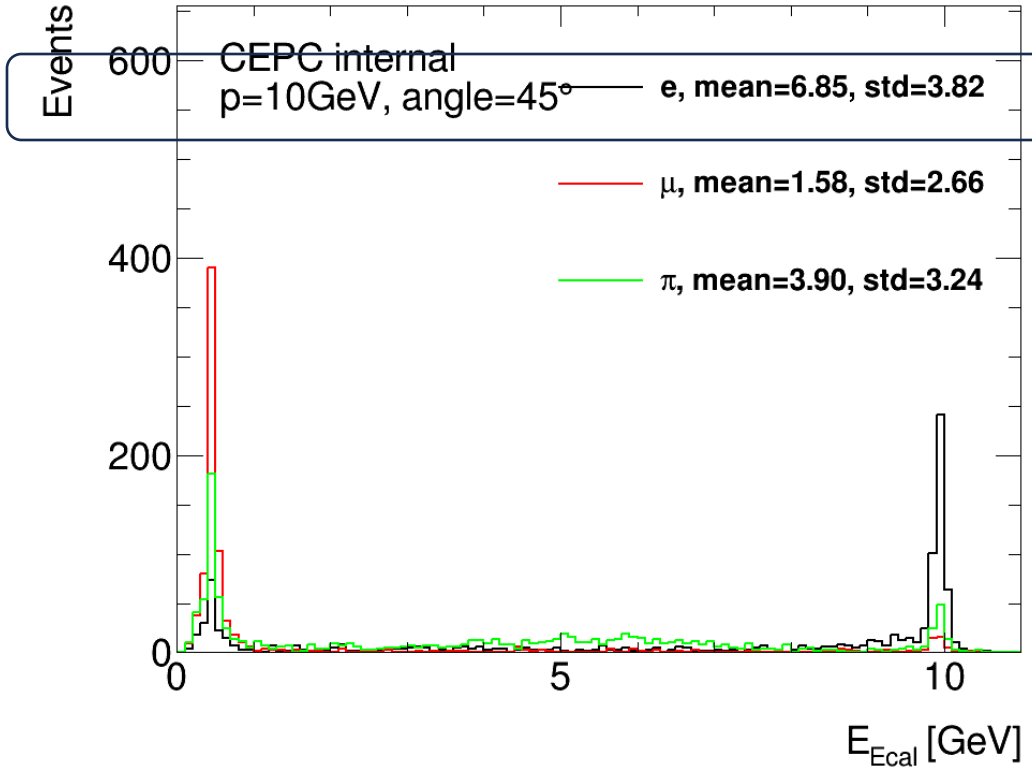
Issue:
- some electrons have very small energy deposit in Ecal
- some pions lose all energy in Ecal.
- wrong truth matching???
- Propose to save reconstructed track px/py/pz for truth matching

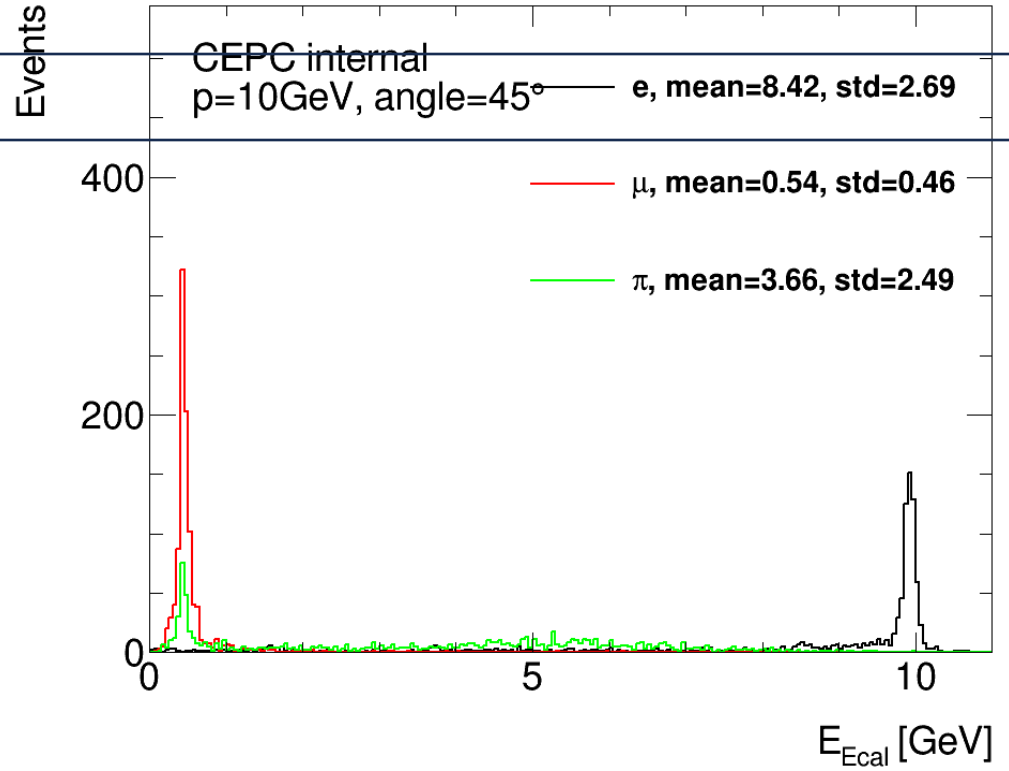
```
if(m_EcalClusterCol[icl]->getAssociatedTracks().size()==1){
  const Track* trk = m_EcalClusterCol[icl]->getAssociatedTracks()[0];
  m_EcalClus_pTrk.push_back(trk->getMomentum());
  m_EcalClus_Trk_px.push_back(trk->getP3()[0]);
  m_EcalClus_Trk_py.push_back(trk->getP3()[1]);
  m_EcalClus_Trk_pz.push_back(trk->getP3()[2]);

  std::vector<TrackState> AllTrackStates = trk->getAllTrackStates();
  for(int istate=0; istate<AllTrackStates.size(); istate++){
    m_EcalClus_trk_tag.push_back(icl);
    m_EcalClus_trk_d0.push_back(AllTrackStates[istate].D0);
    m_EcalClus_trk_z0.push_back(AllTrackStates[istate].Z0);
    m_EcalClus_trk_phi.push_back(AllTrackStates[istate].phi0);
    m_EcalClus_trk_tanL.push_back( AllTrackStates[istate].tanLambda );
    m_EcalClus_trk_kappa.push_back( AllTrackStates[istate].Kappa);
    m_EcalClus_trk_omega.push_back( AllTrackStates[istate].Omega );
    m_EcalClus_trk_location.push_back( AllTrackStates[istate].location );
  }
```

Issue2: Truth Matching

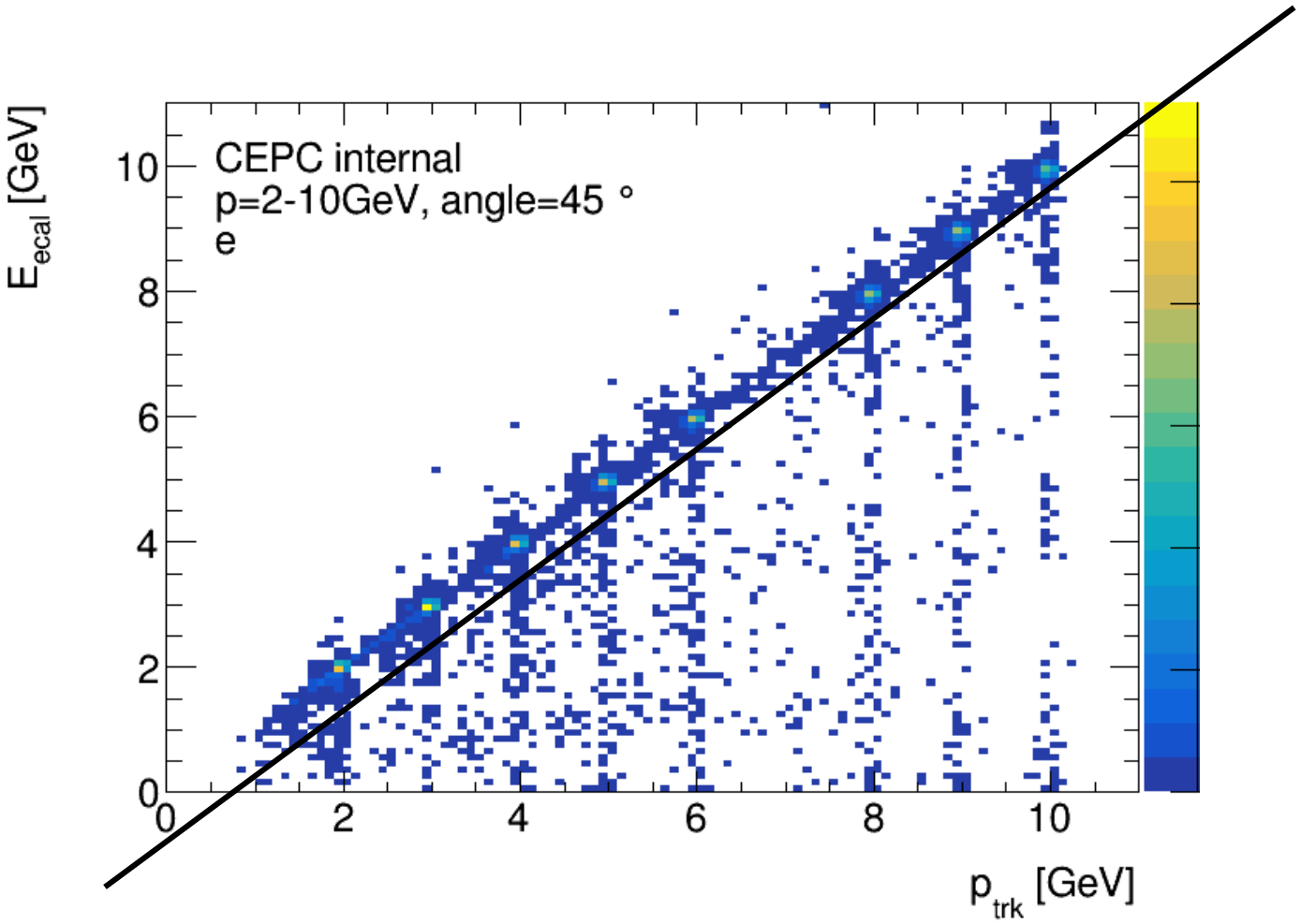
- truth matching: min(reco direction, truth direction)
- with truth matching, the issue is fixed.

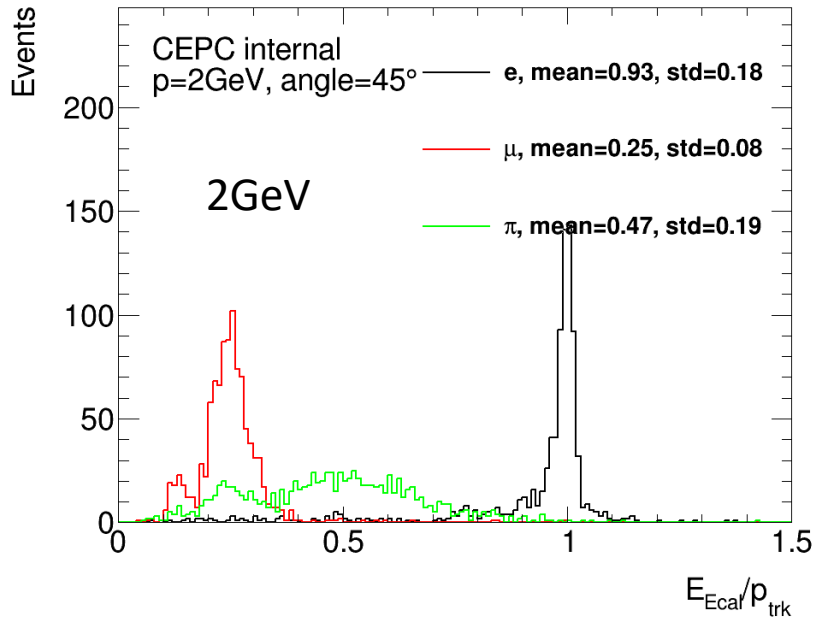
Issue:
- some electrons have very small energy deposit in Ecal
- some pions lose all energy in Ecal.
- wrong truth matching???
- Propose to save reconstructed track px/py/pz for truth matching

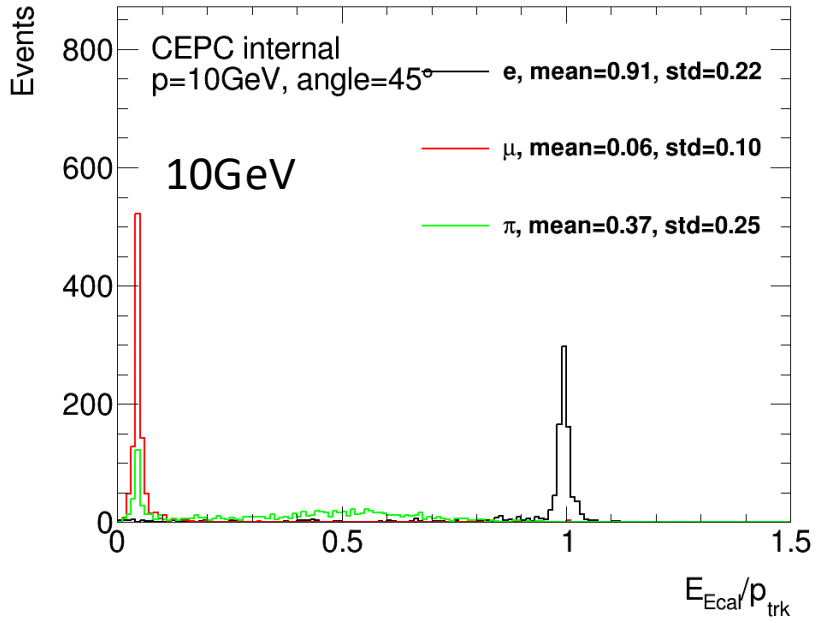




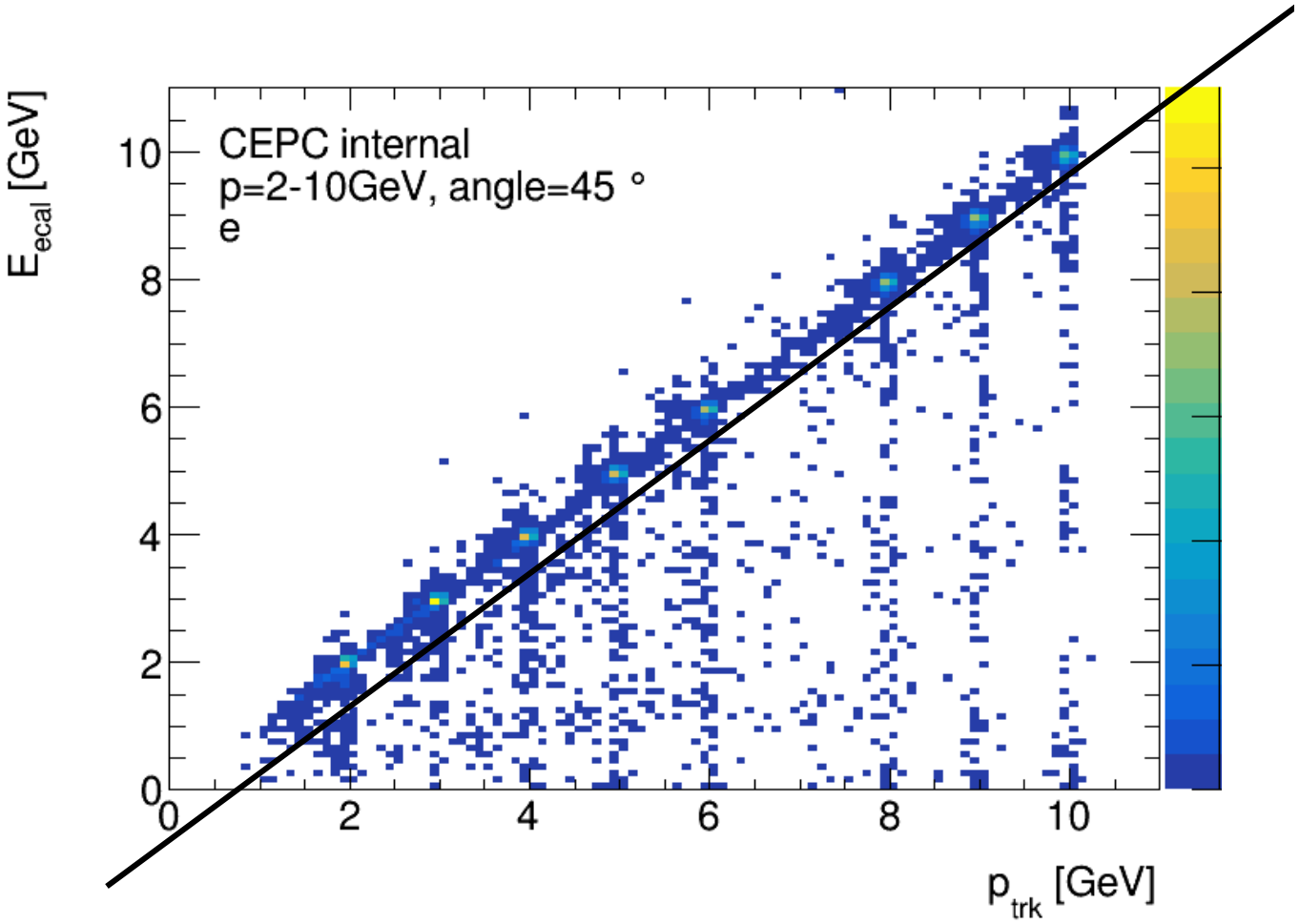
Electron Identification

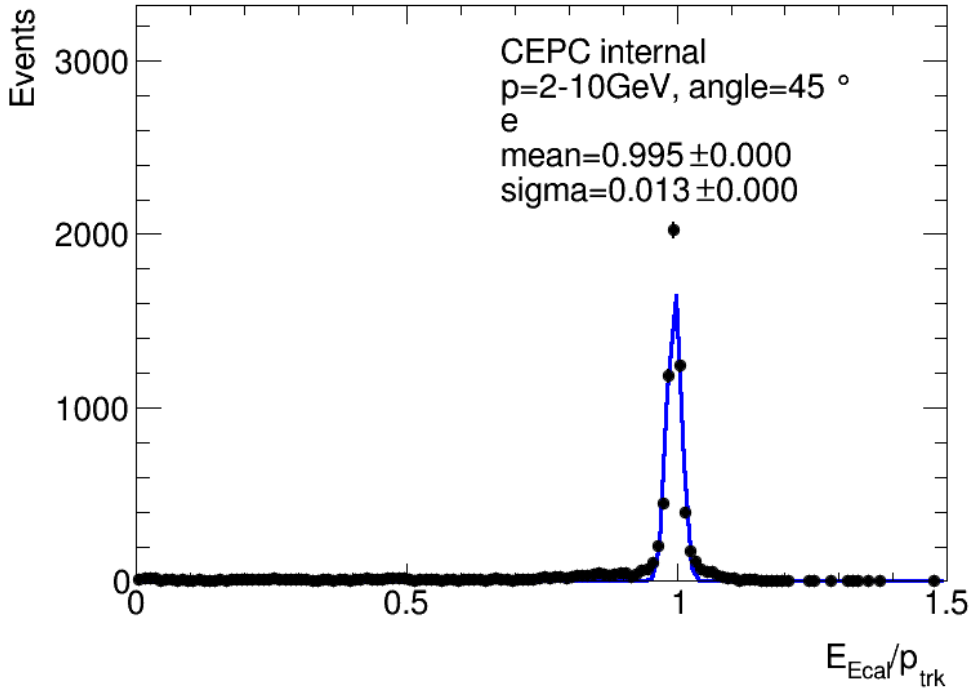






Electron Identification

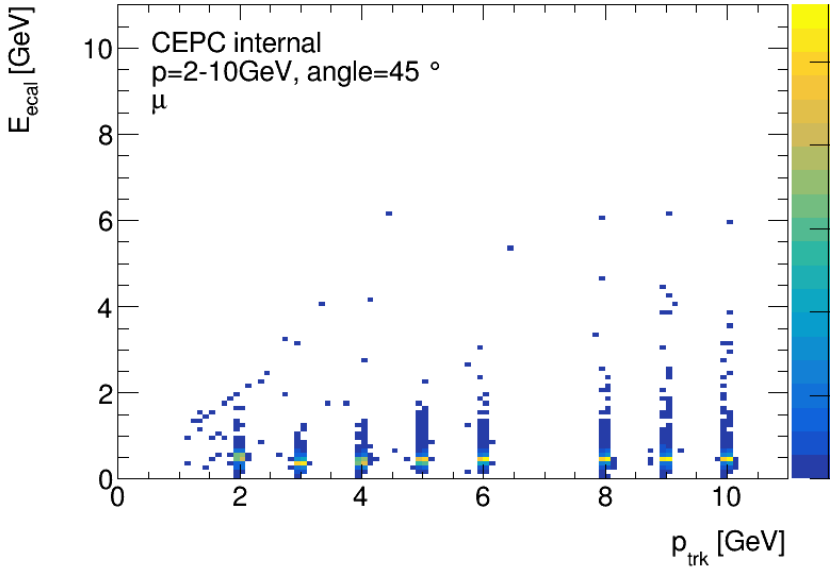


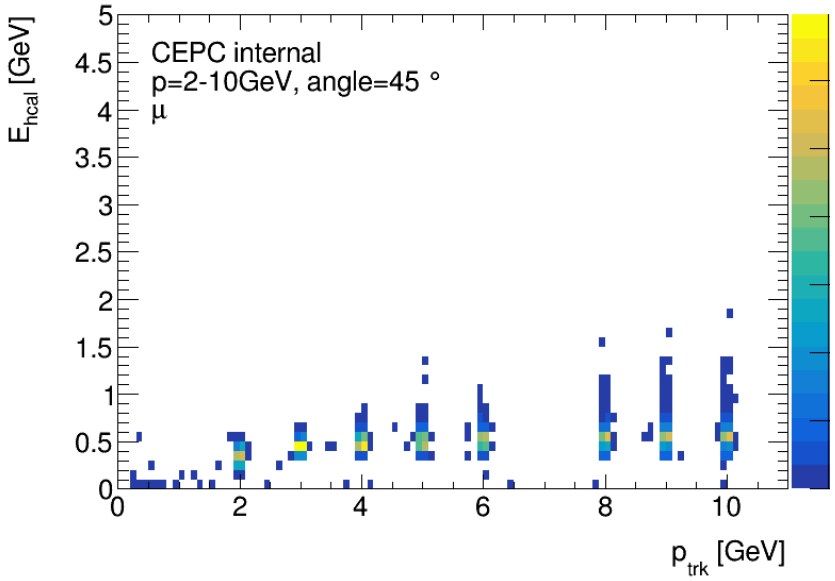


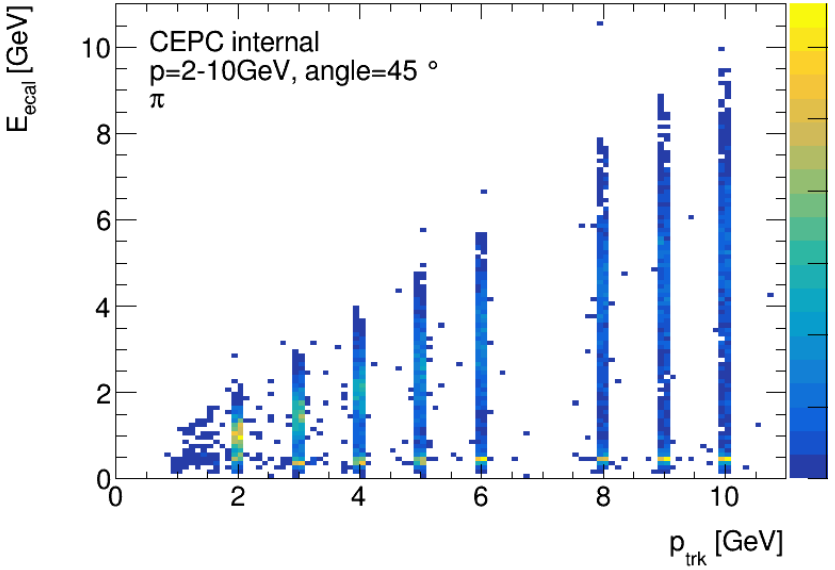
Proposed Electron ID working point:

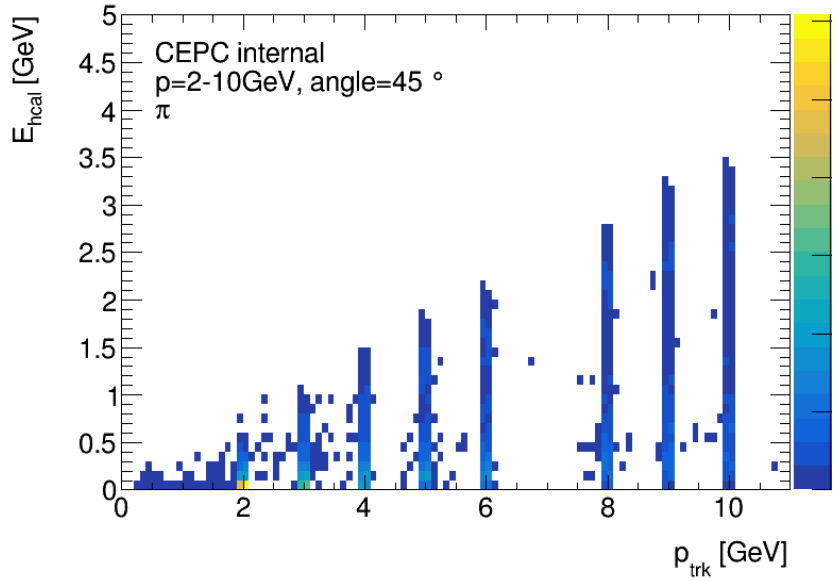
- definition: E(ecal)/p(trk) > 0.9
- efficiency: 81.8% at 2GeV and 82.7% at 10 GeV

Muon Identification





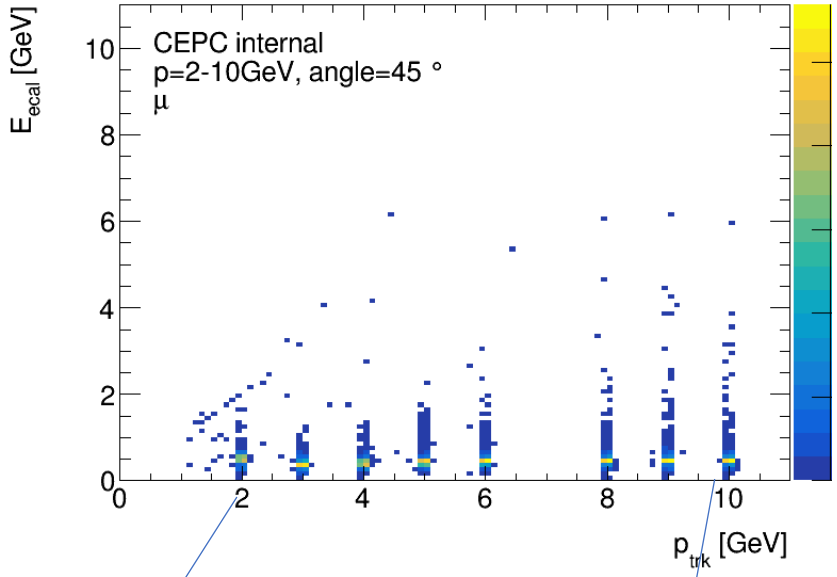


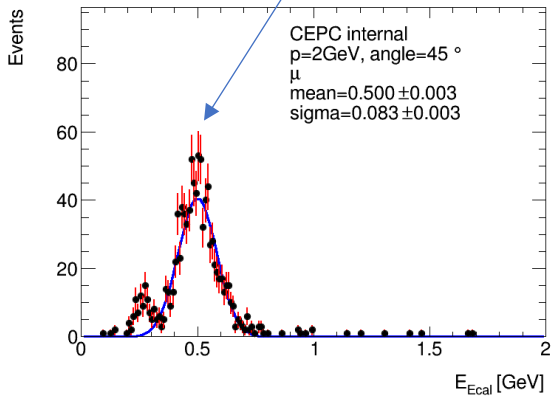


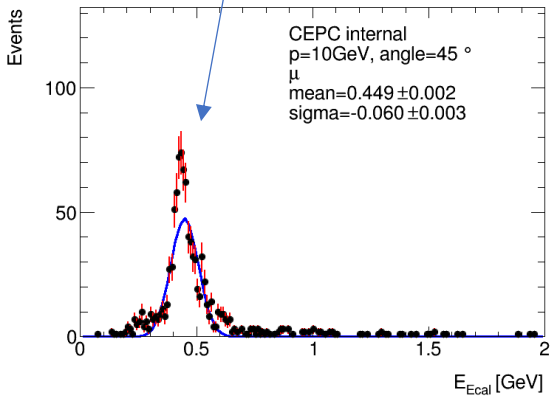
For a single-momentum muon:
- deposit a single-amount of energy in Ecal and Hcal

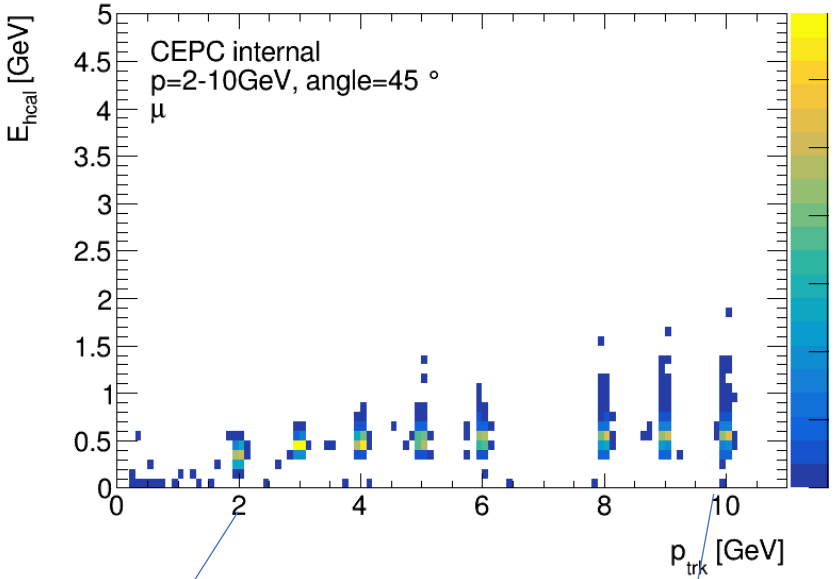
For pions, the energy distribution in Ecal and Hcal is much more broad.

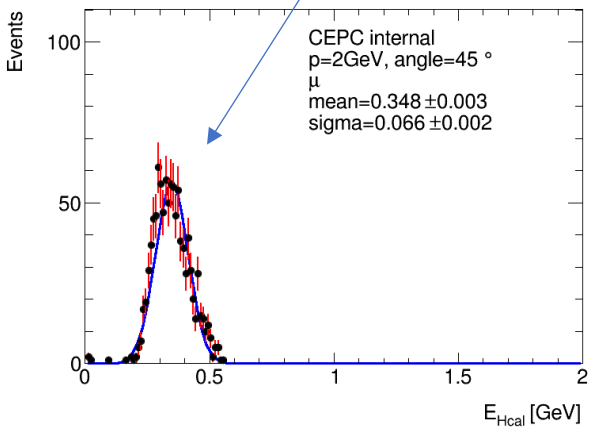
Muon Identification

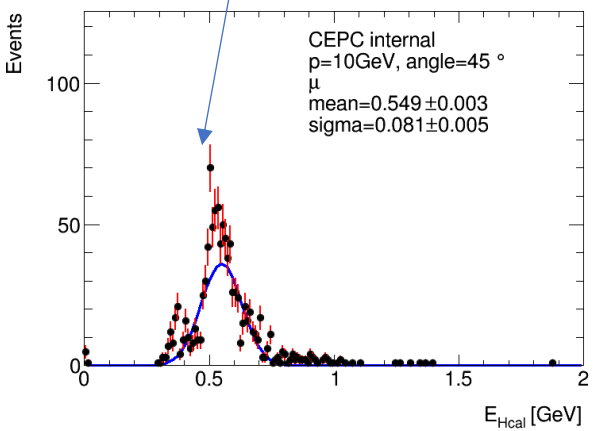










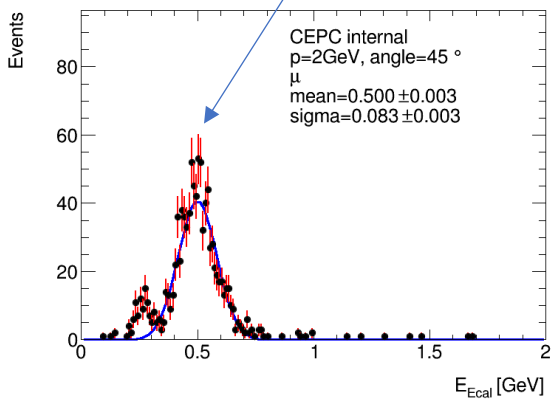


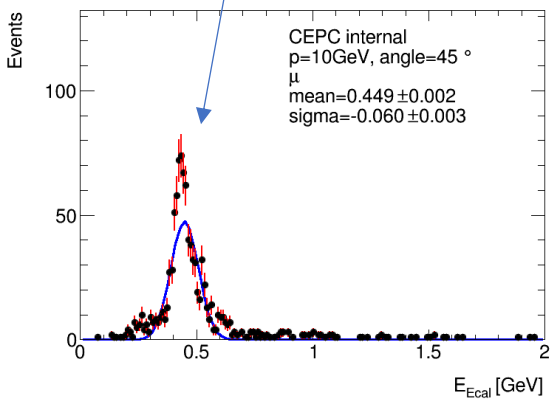
Muon Identification

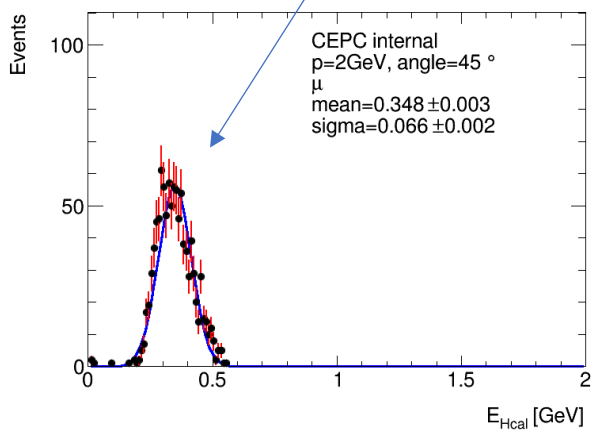
Proposed muon ID working point:
- definition: chi2_Ecal <3 and chi2_Hcal < 3 (~3sigma region)
- efficiency (2GeV): 74% (81% and 91% respectively)
- efficiency (10GeV): 57% (73% and 79% respectively)

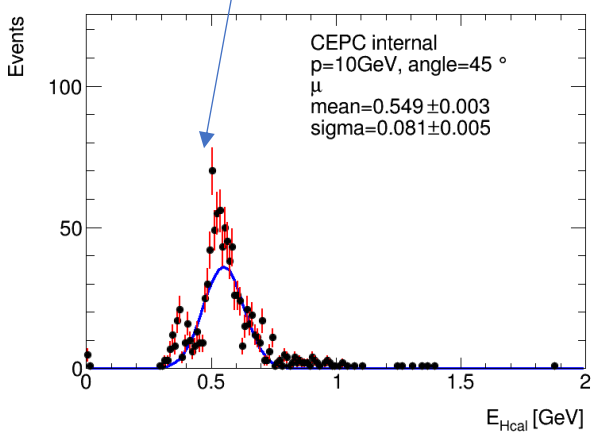
$$\chi^2_{Ecal}(2GeV) = \left(\frac{E_{Ecal} - 0.50}{0.083}\right)^2$$

$$\chi^2_{Hcal}(2GeV) = \left(\frac{E_{Hcal} - 0.348}{0.066}\right)^2$$









Summary

- Preliminary lepton ID working point.

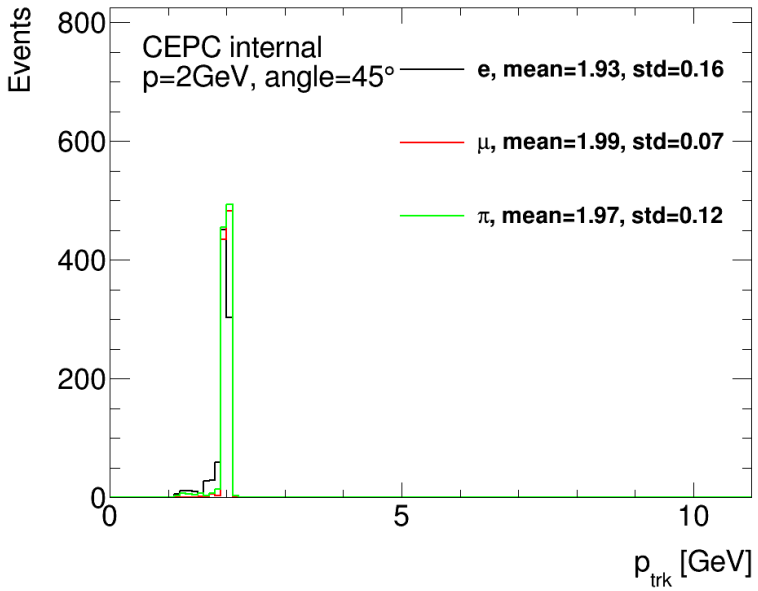
Proposed Electron ID working point:
- definition: E(ecal)/p(trk) > 0.9
- efficiency: 81.8% at 2GeV and 82.7% at 10 GeV

Proposed muon ID working point:
- definition: chi2_Ecal <3 and chi2_Hcal < 3 (~3sigma region)
- efficiency (2GeV): 74% (81% and 91% respectively)
- efficiency (10GeV): 57% (73% and 79% respectively)

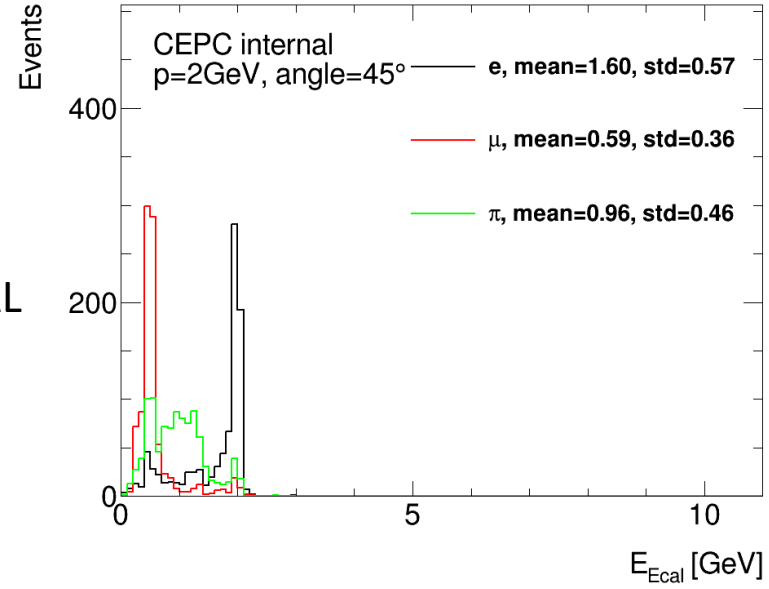
Work to do:
- improve the fits
- implement the leptonID into the CEPCSW
- study high-energy lepton ID (Z → ee/mumu)
- Machine Learning for lepton ID
- Muon counter (when available)

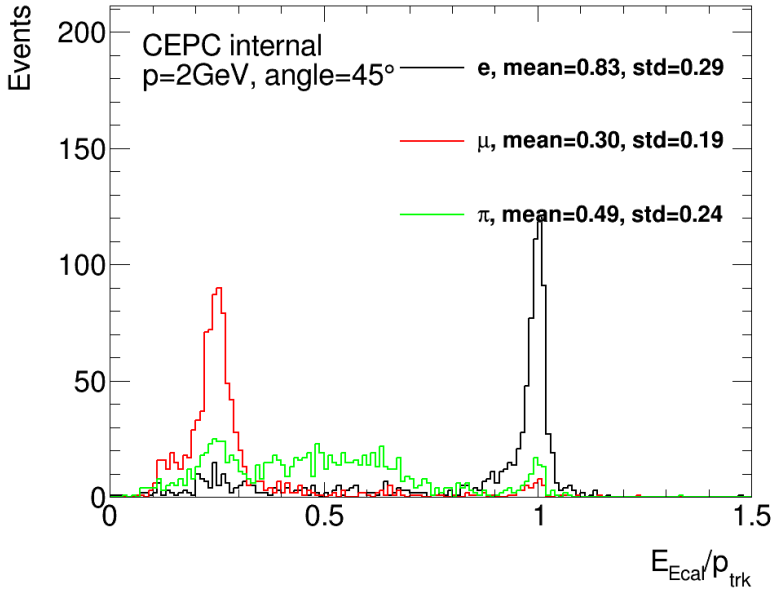
- BACK UP

Example: 2 GeV



ECAL

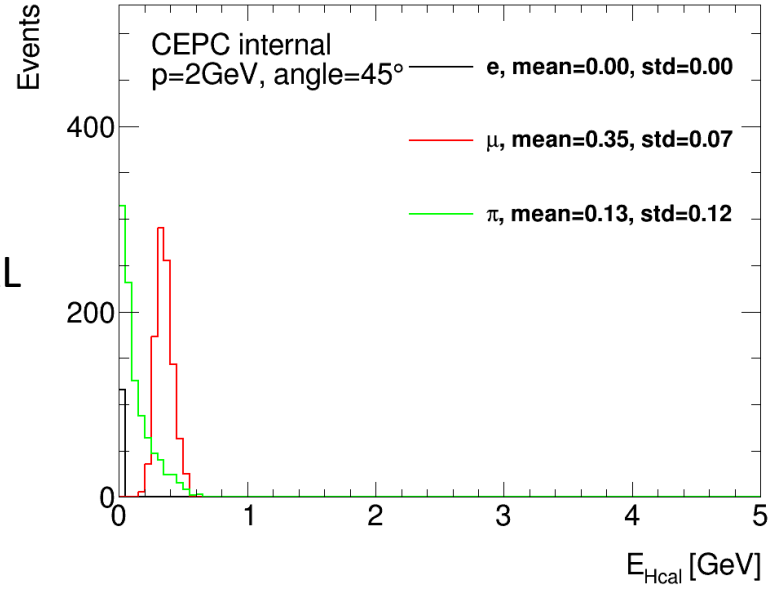


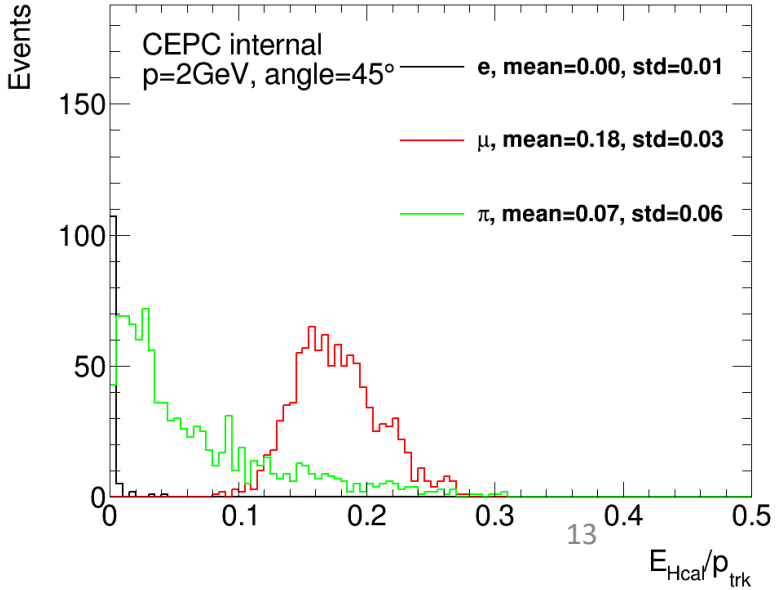


Features:

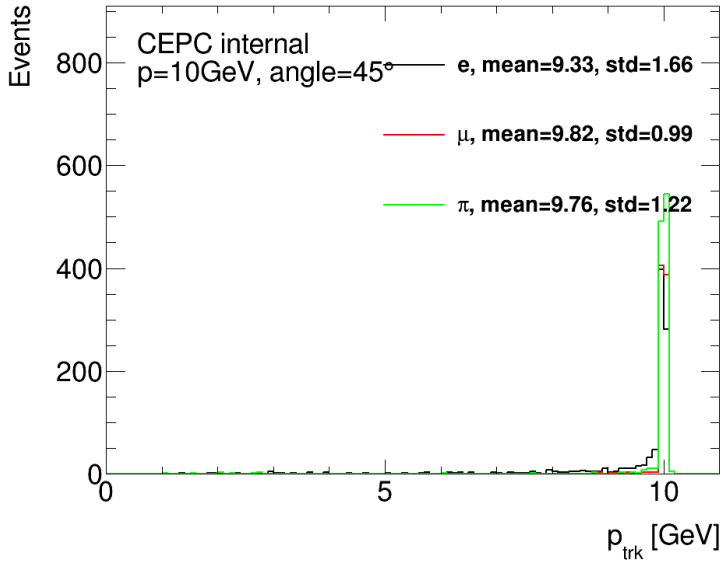
- electrons have high E/p in ECAL and won't reach HCAL.
- muons have gaussian-like energy deposit in ECAL and HCAL for a single momentum
- pions' energy deposits in ECAL/HCAL: hard to describe and clearly different from muon/electron (the best place to use ML!!).

HCAL

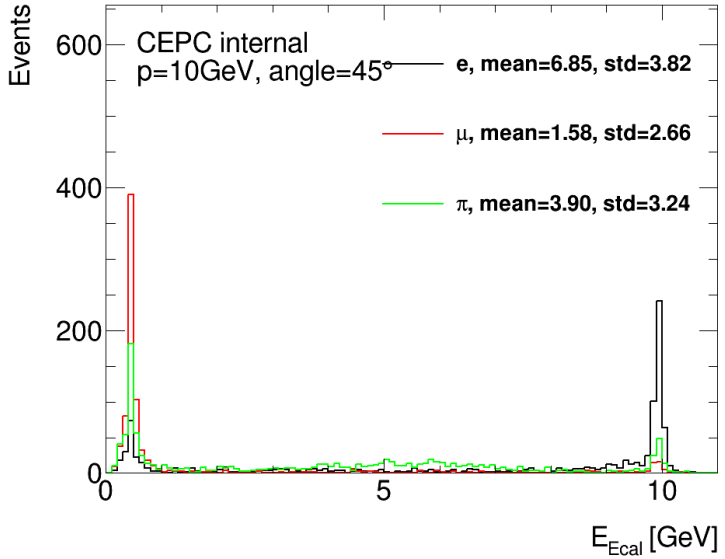


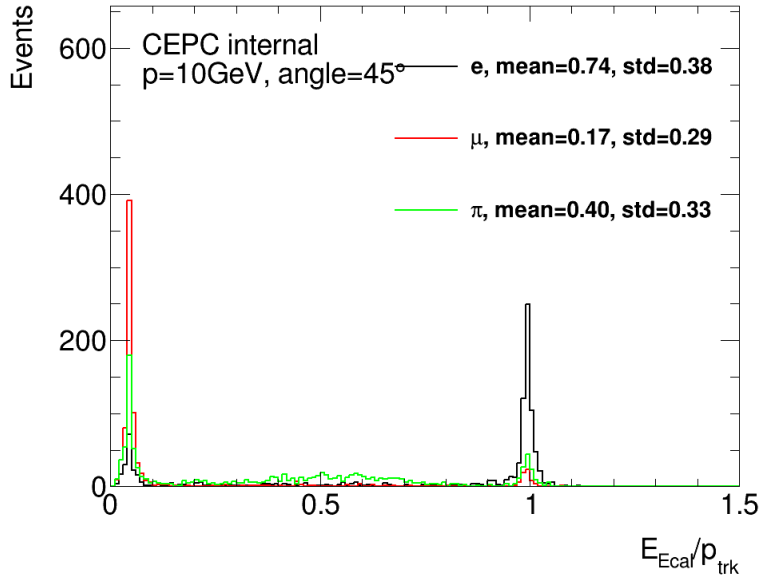


Example: 10 GeV



ECAL

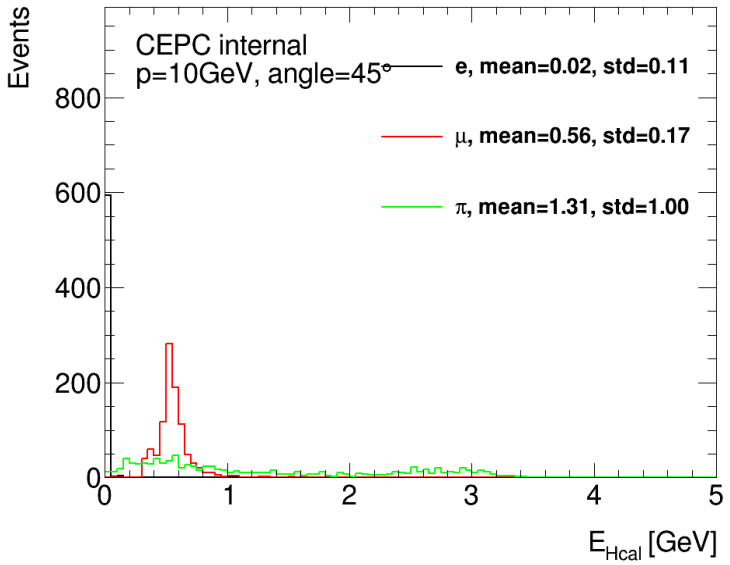


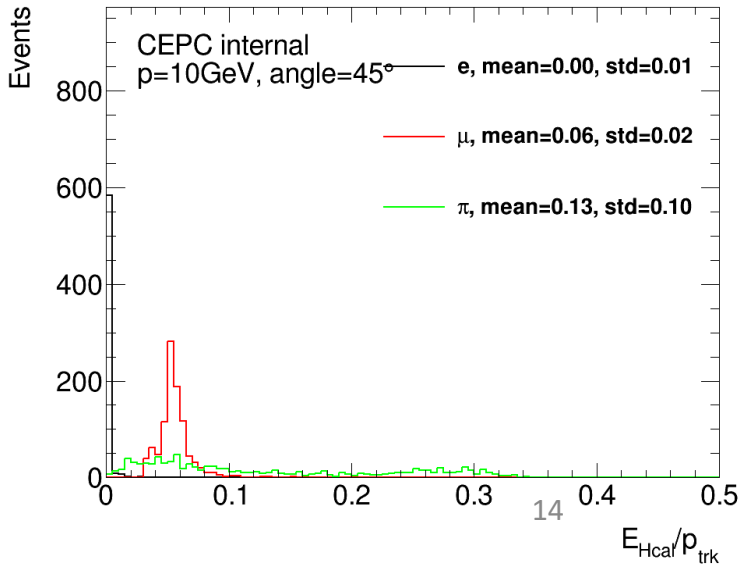


Features:

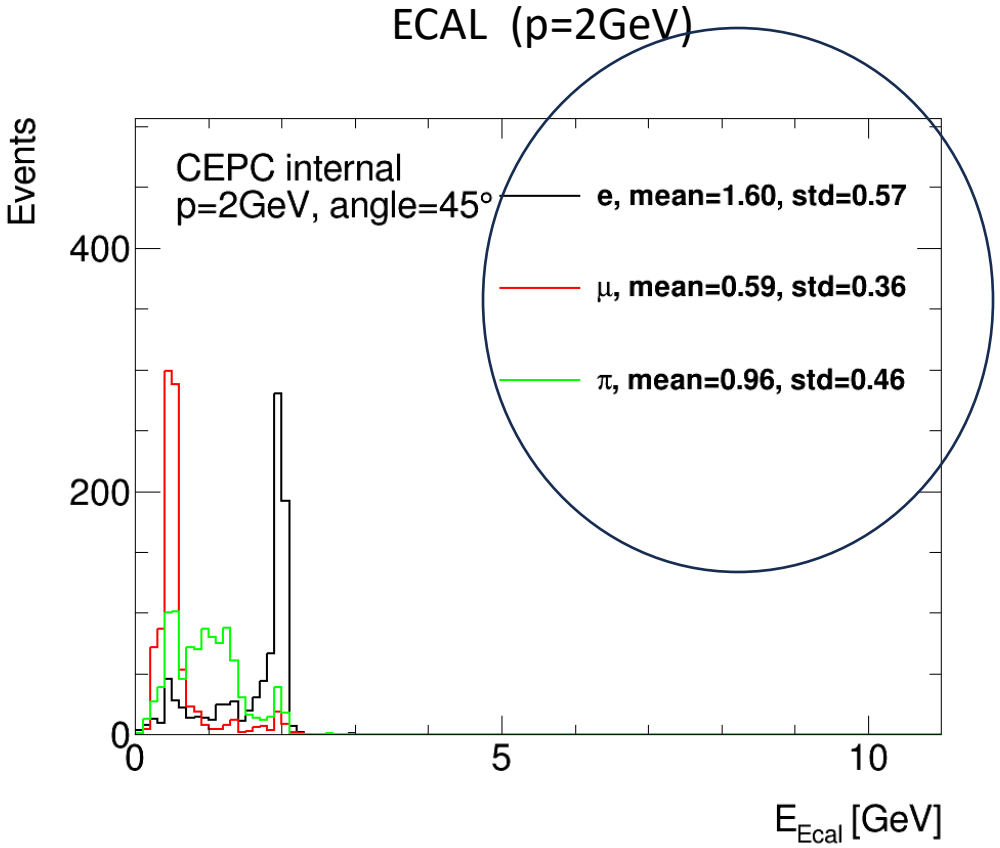
- electrons have high E/p in ECAL and won't reach HCAL.
- muons have gaussian-like energy deposit in ECAL and HCAL for a single momentum
- pions' energy deposits in ECAL/HCAL: hard to describe and clearly different from muon/electron (the best place to use ML!!).

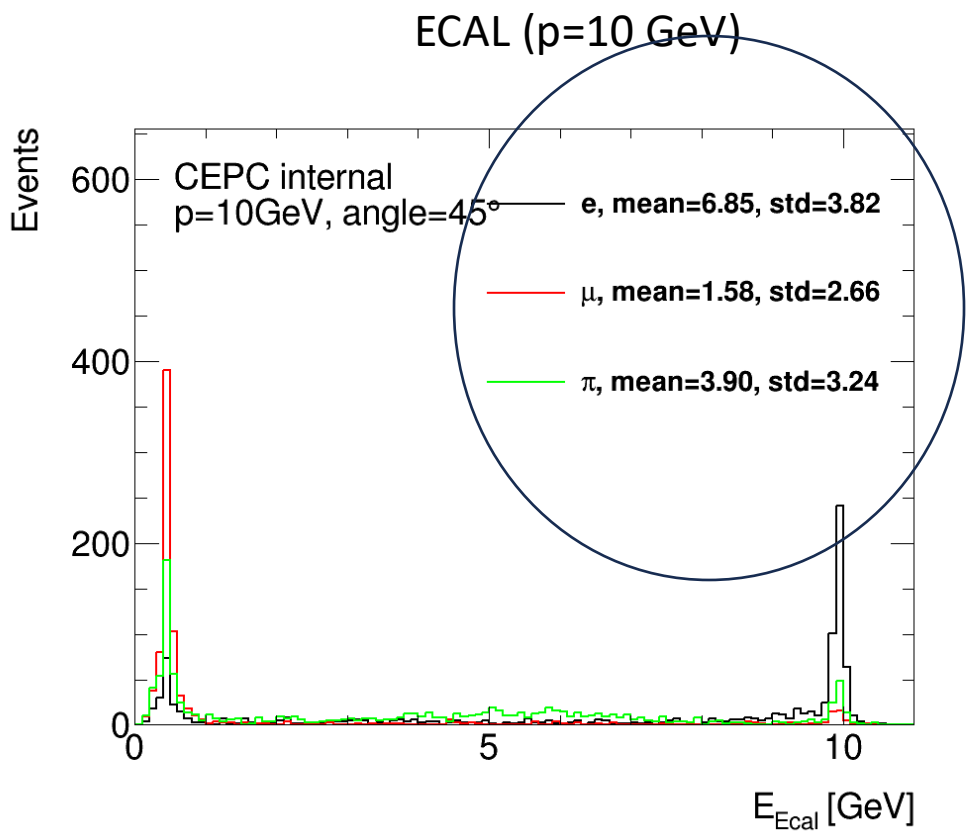
HCAL





Comparison: 2 GeV v.s. 10 GeV

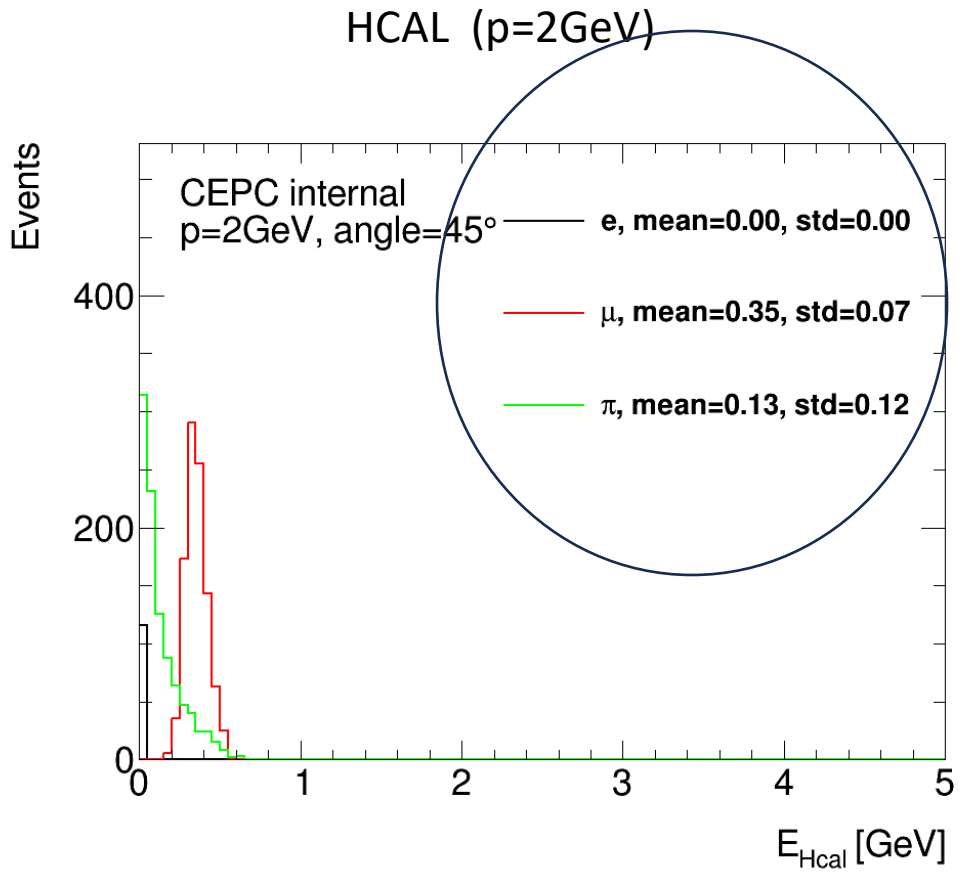


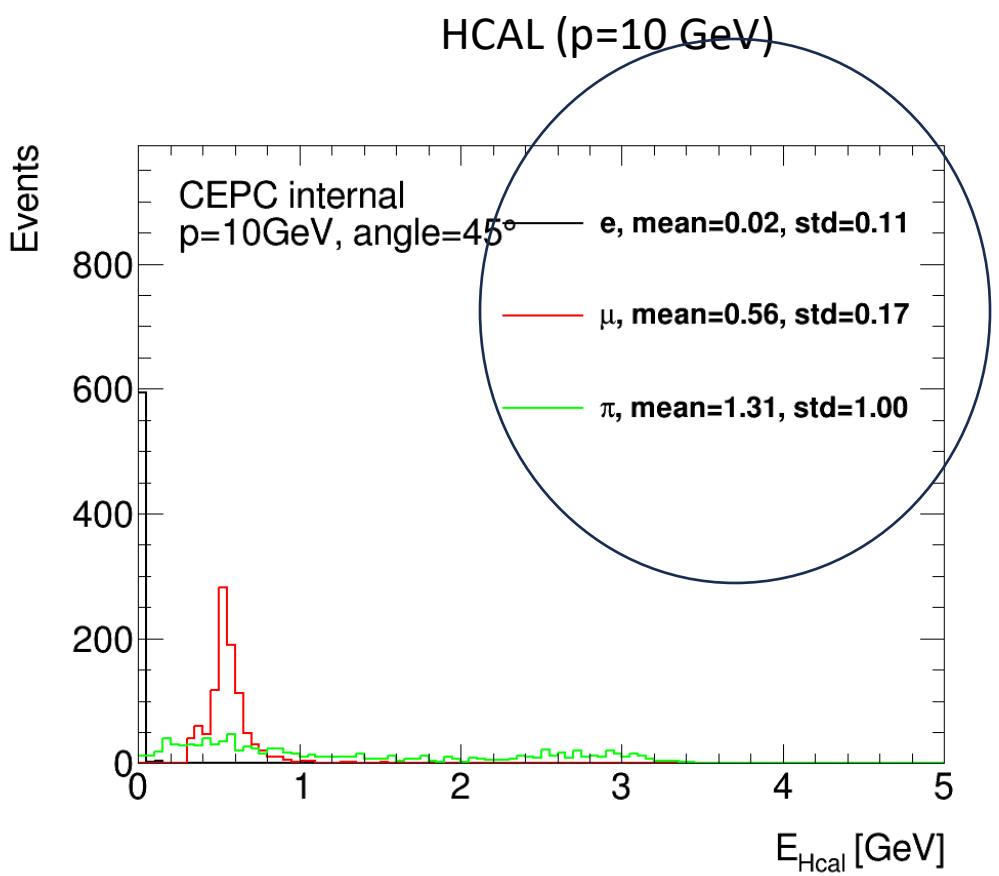


Features:

- the bigger momentum, the more energy deposit in ECAL.
- Similar Ecal energy increasement rate: 0.59 → 1.58 for muon, 0.96 → 3.90 for pion

Comparison: 2 GeV v.s. 10 GeV

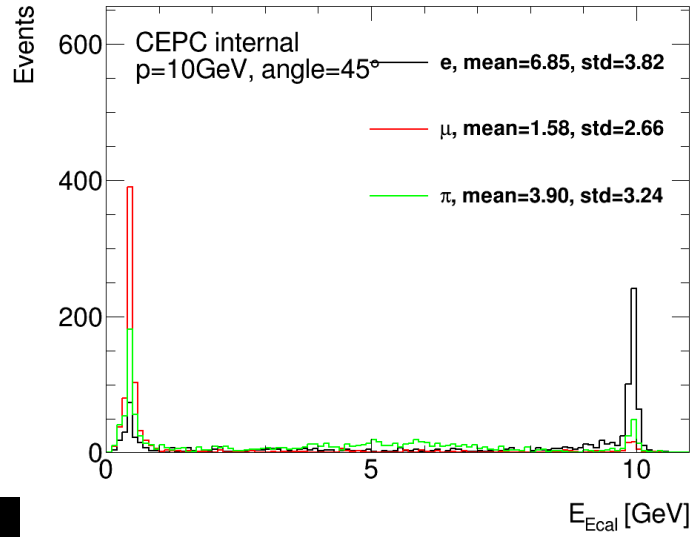




Features:

- the bigger momentum, the more energy deposit in ECAL.
- Very different Hcal energy increasement rate: 0.35 → 0.56 for muon, 0.13 → 1.31 for pion

Truth Matching



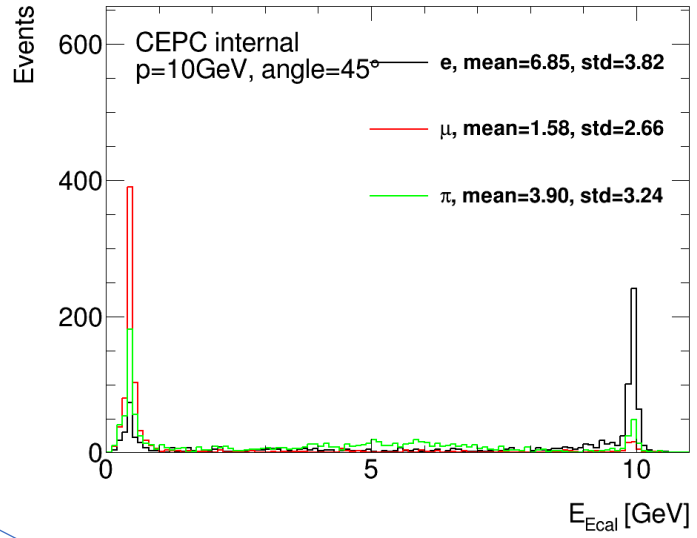
Issue:
- some electrons have very small energy deposit in Ecal
- some pions lose all energy in Ecal.
- wrong truth matching???
- Propose to save reconstructed track px/py/pz for truth matching

```
root [8] MCParticle -> Scan("mcPx:mcPy:mcPz")
************************************************************
*    Row   * Instance *      mcPx *      mcPy *      mcPz *
************************************************************
*        0 *        0 * 1.4141644 * 0.0117841 * 1.4142135 *
*        0 *        1 * -1.139660 * -0.837361 * 1.4142135 *
*        0 *        2 * 0.3934340 * 1.3583849 * 1.4142135 *
*        1 *        0 * 1.3731187 * -0.338444 * 1.4142135 *
*        1 *        1 * 1.1648671 * 0.8019254 * 1.4142135 *
*        1 *        2 * 0.1600361 * -1.405129 * 1.4142135 *
*        2 *        0 * 0.0461763 * 1.4134595 * 1.4142135 *
*        2 *        1 * -0.338782 * -1.373035 * 1.4142135 *
*        2 *        2 * 0.8967785 *  1.093521 * 1.4142135 *
*        3 *        0 * 0.9950917 * 1.0048842 * 1.4142135 *
*        3 *        1 * 0.7206547 * -1.216822 * 1.4142135 *
*        3 *        2 * -1.128017 * 0.8529807 * 1.4142135 *
*        4 *        0 * 0.8517094 * -1.128977 * 1.4142135 *
*        4 *        1 * -1.344648 * 0.4380868 * 1.4142135 *
*        4 *        2 * -1.388419 * -0.268870 * 1.4142135 *
*        5 *        0 * -1.220502 * 0.7144048 * 1.4142135 *
*        5 *        1 * -0.471672 * -1.333238 * 1.4142135 *
*        5 *        2 * 0.6957339 * -1.231240 * 1.4142135 *
*        6 *        0 * 1.2633993 * 0.6354698 * 1.4142135 *
*        6 *        1 * 0.1717654 * 1.4037437 * 1.4142135 *
*        6 *        2 * -0.798826 * -1.166994 * 1.4142135 *
*        7 *        0 *  1.336604 * -0.462049 * 1.4142135 *
*        7 *        1 * 0.6832138 * 1.2382321 * 1.4142135 *
*        7 *        2 * -0.738398 * -1.206137 * 1.4142135 *
*        8 *        0 * 1.3998915 * 0.2007579 * 1.4142135 *
```

```
root [9] RecClusters -> Scan("EcalClus_truthMC_pid:EcalClus_Escale:EcalClus_ptrk:EcalClus_truthMC_weight:EcalClus_truthMC_px:EcalClus_truthMC_py:EcalClus_truthMC_pz")
*****************************************************************************************************************
*    Row   * Instance * EcalClus_ * EcalClus_ * EcalClus_ * EcalClus_ * EcalClus_ * EcalClus_ * EcalClus_ *
*****************************************************************************************************************
*        0 *        0 *      -211 * 1.4472522 * 2.0014236 *         1 * 0.3934340 * 1.3583849 * 1.4142135 *
*        0 *        1 *        13 * 0.7885690 * 2.0063059 * 0.9999999 * -1.139660 * -0.837361 * 1.4142135 *
*        0 *        2 *        11 * 1.7394524 * 1.8885551 * 1.0000001 * 1.4141644 * 0.0117841 * 1.4142135 *
*        1 *        0 *        13 * 0.0246001 *       -99 *         1 * 1.1648671 * 0.8019254 * 1.4142135 *
*        1 *        1 *        13 * 0.4542815 * 1.9947865 * 0.9999999 * 1.1648671 * 0.8019254 * 1.4142135 *
*        1 *        2 *        11 * 2.0022621 * 2.0035669 * 1.0000001 * 1.3731187 * -0.338444 * 1.4142135 *
*        1 *        3 *      -211 * 1.4630663 * 1.9876115 * 0.999
*        1 *        4 *        13 * 0.1924284 * 2.0423650 *
*        2 *        0 *      -211 * 0.5237308 * 1.9973602 *
*        2 *        1 *        11 * 2.0535900 * 2.0026135 * 0.999
*        2 *        2 *        13 * 0.4954095 * 1.9859817 * 0.999
*        3 *        0 *        13 * 0.4955662 * 2.0048124 *
*        3 *        1 *        11 * 2.0085377 * 1.9962967 * 1.000
*        3 *        2 *      -211 * 0.8898755 * 2.0096585 *
*        4 *        0 *        11 * 2.0282018 * 1.9962714 *
*        4 *        1 *      -211 * 0.3638647 * 2.0013227 *
*        4 *        2 *        13 * 0.4583229 * 2.0000076 *
*        5 *        0 *        11 * 1.8551514 *       -99 *
*        5 *        1 *      -211 * 0.7934654 * 1.9935165 *
*        5 *        2 *        13 * 0.2550188 * 2.0047075 *
*        6 *        0 *        13 * 0.5617442 * 1.9993442 *
*        6 *        1 *        11 * 1.9143279 * 1.9908685 * 0.999
*        6 *        2 *      -211 * 0.4418135 * 2.0033662 * 0.999
*        7 *        0 *      -211 * 0.6981110 * 1.9944497 *
*        7 *        1 *        13 * 0.5945575 * 2.0036141 * 0.997
```

```
    m_totE_Ecal += m_EcalClusterCol[icl]->getLongiE();
    auto truthMap = m_EcalClusterCol[icl]->getLinkedMCP();
    for(auto iter: truthMap){
      m_EcalClus_truthMC_tag.push_back(icl);
      m_EcalClus_truthMC_pid.push_back(iter.first.getPDG() );
      m_EcalClus_truthMC_px.push_back(iter.first.getMomentum().x);
      m_EcalClus_truthMC_py.push_back(iter.first.getMomentum().y);
      m_EcalClus_truthMC_pz.push_back(iter.first.getMomentum().z);
      m_EcalClus_truthMC_E.push_back(iter.first.getEnergy());
      m_EcalClus_truthMC_EPx.push_back(iter.first.getEndpoint().x);
      m_EcalClus_truthMC_EPy.push_back(iter.first.getEndpoint().y);
      m_EcalClus_truthMC_EPz.push_back(iter.first.getEndpoint().z);
      m_EcalClus_truthMC_weight.push_back(iter.second);
    }
  }
```

Truth Matching



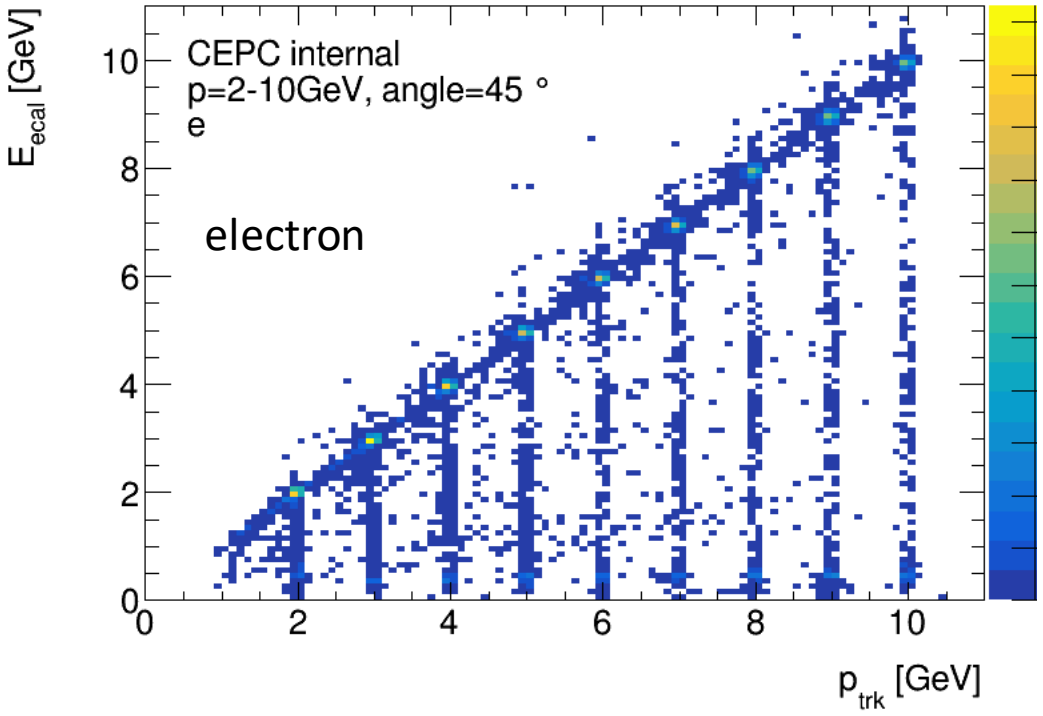
Issue:
- some electrons have very small energy deposit in Ecal
- some pions lose all energy in Ecal.
- wrong truth matching???
- Propose to save reconstructed track px/py/pz for truth matching

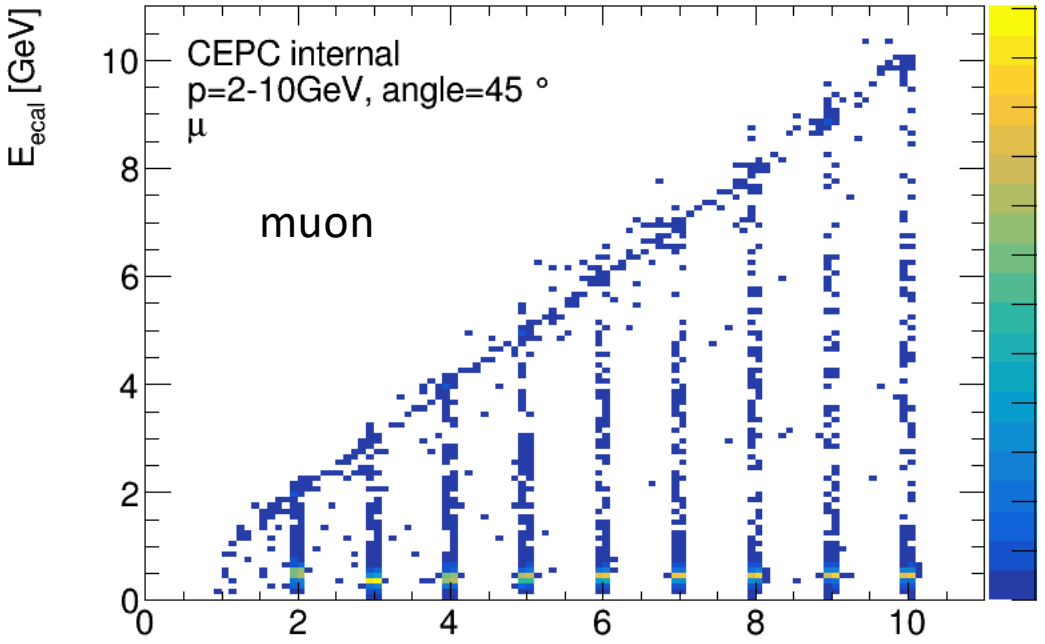
If you think it is OK, I can prepare a MR to add more variables.

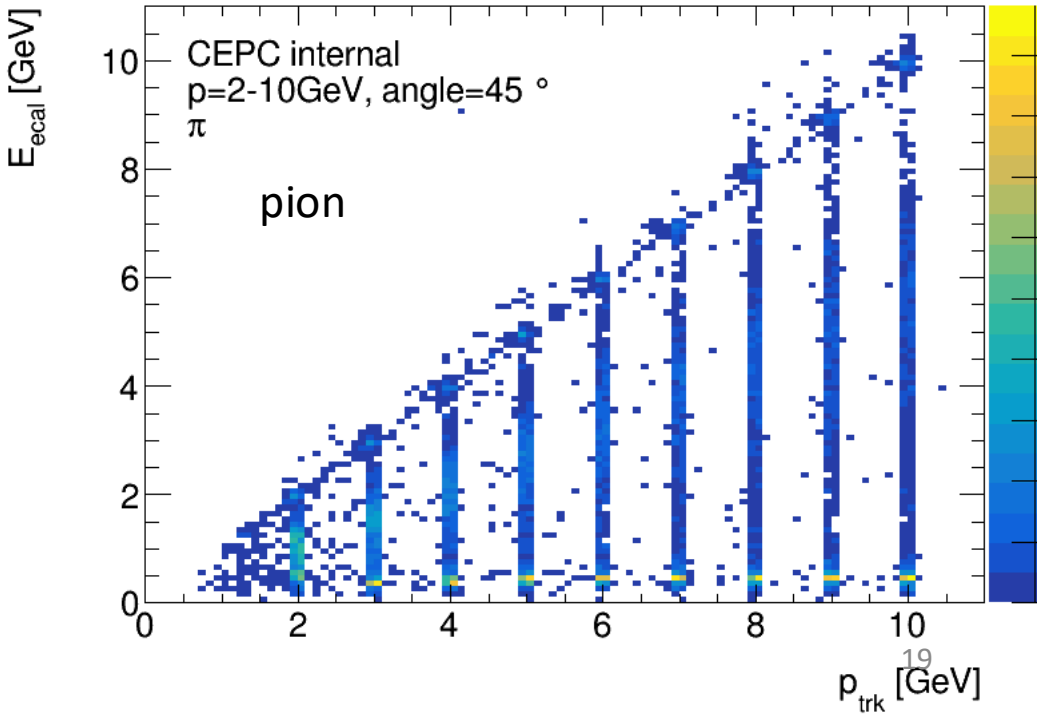
```
  m_totE_Ecal += m_EcalClusterCol[icl]->getLongiE();
  auto truthMap = m_EcalClusterCol[icl]->getLinkedMCP();
  for(auto iter: truthMap){
    m_EcalClus_truthMC_tag.push_back(icl);
    m_EcalClus_truthMC_pid.push_back(iter.first.getPDG() );
    m_EcalClus_truthMC_px.push_back(iter.first.getMomentum().x);
    m_EcalClus_truthMC_py.push_back(iter.first.getMomentum().y);
    m_EcalClus_truthMC_pz.push_back(iter.first.getMomentum().z);
    m_EcalClus_truthMC_E.push_back(iter.first.getEnergy());
    m_EcalClus_truthMC_EPx.push_back(iter.first.getEndpoint().x);
    m_EcalClus_truthMC_EPy.push_back(iter.first.getEndpoint().y);
    m_EcalClus_truthMC_EPz.push_back(iter.first.getEndpoint().z);
    m_EcalClus_truthMC_weight.push_back(iter.second);
  }
}
```

```
if(m_EcalClusterCol[icl]->getAssociatedTracks().size()==1){
  const Track* trk = m_EcalClusterCol[icl]->getAssociatedTracks()[0];
  m_EcalClus_pTrk.push_back(trk->getMomentum());

  std::vector<TrackState> AllTrackStates = trk->getAllTrackStates();
  for(int istate=0; istate<AllTrackStates.size(); istate++){
    m_EcalClus_trk_tag.push_back(icl);
    m_EcalClus_trk_d0.push_back(AllTrackStates[istate].D0);
    m_EcalClus_trk_z0.push_back(AllTrackStates[istate].Z0);
    m_EcalClus_trk_phi.push_back(AllTrackStates[istate].phi0);
    m_EcalClus_trk_tanL.push_back( AllTrackStates[istate].tanLambda );
    m_EcalClus_trk_kappa.push_back( AllTrackStates[istate].Kappa);
    m_EcalClus_trk_omega.push_back( AllTrackStates[istate].Omega );
    m_EcalClus_trk_location.push_back( AllTrackStates[istate].location );
  }

}
else
  m_EcalClus_pTrk.push_back(-99);
```

Ecal Energy:p(trk)



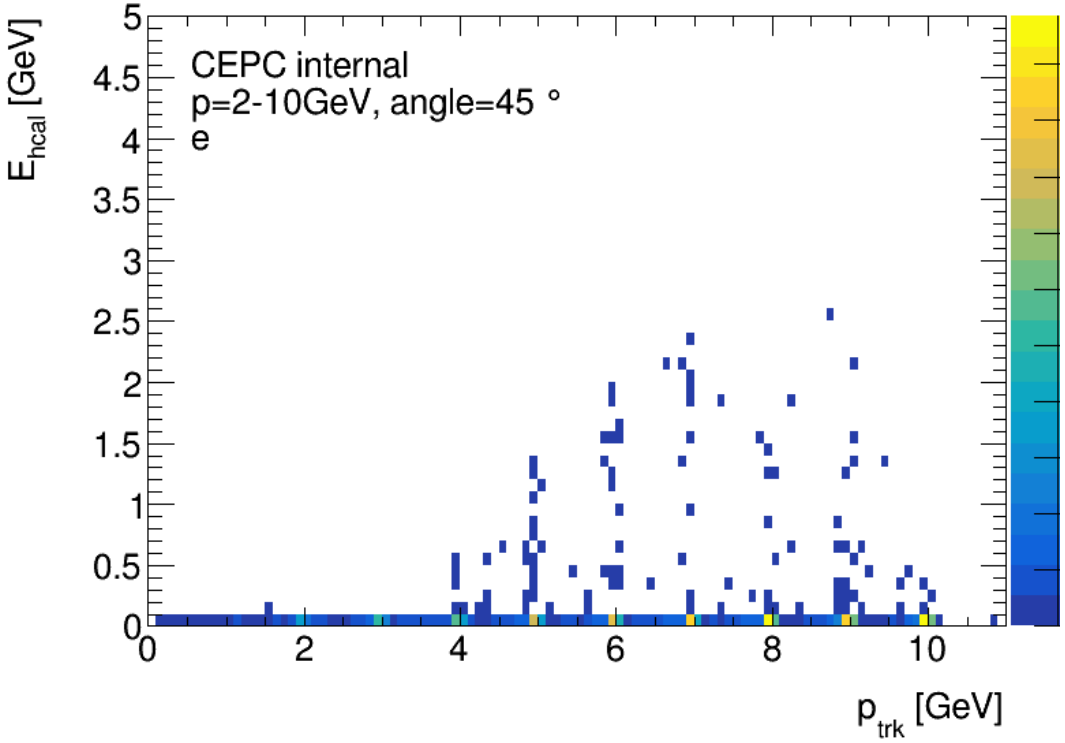


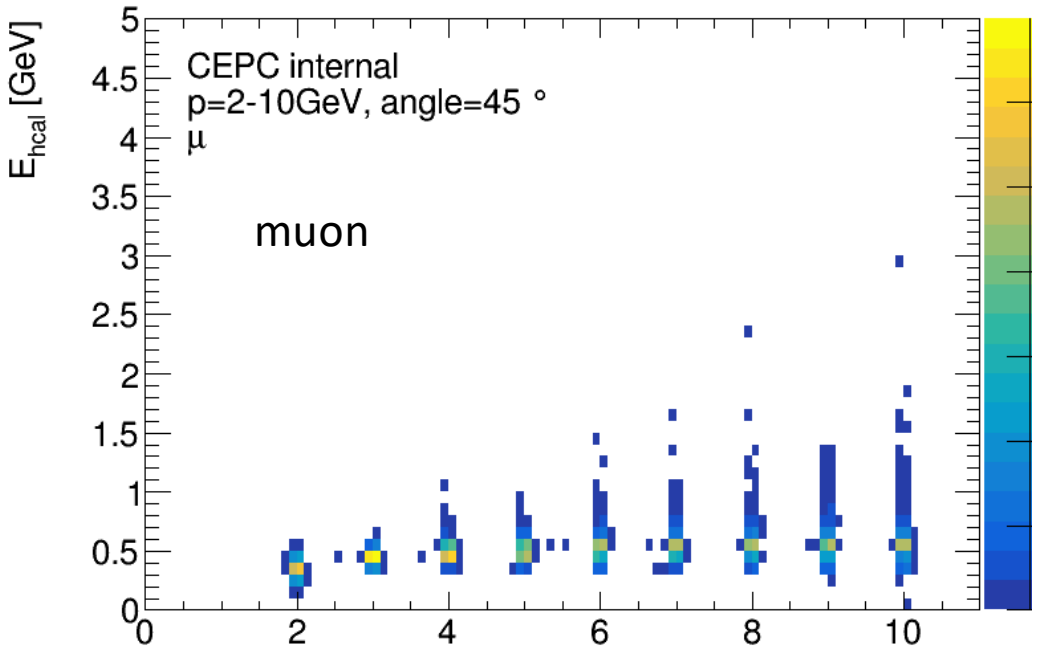


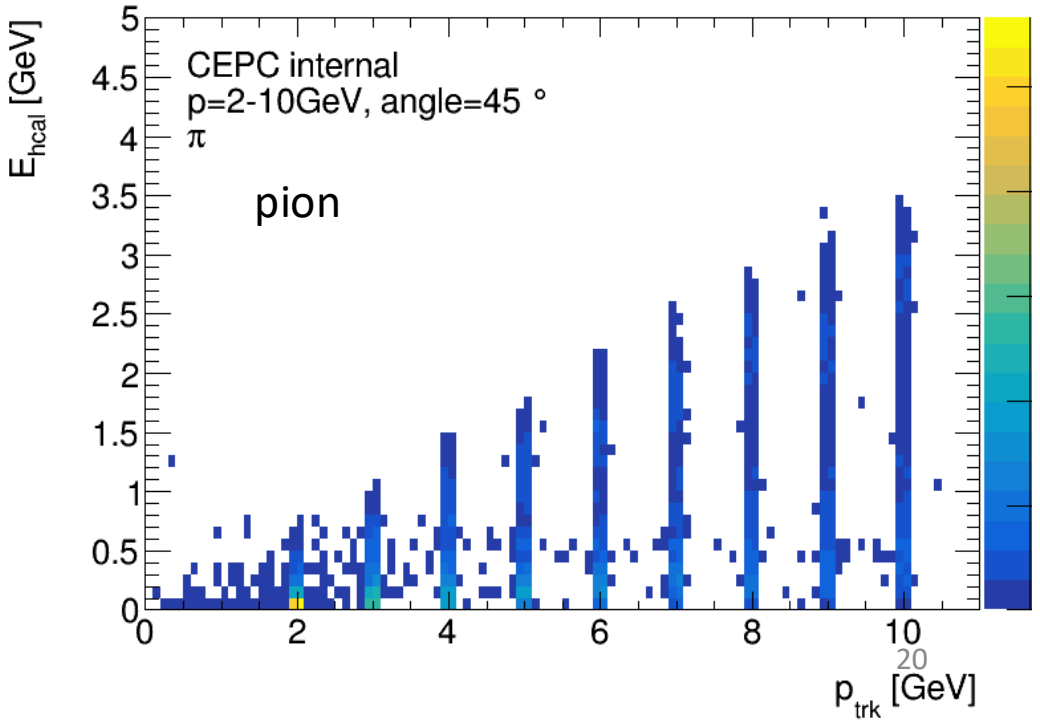
Features on Ecal energy:

- electrons will lose all energy
- muon lose the same energy
- pion lose more energy at low momentum and the same energy as muon at high momentum

Hcal Energy:p(trk)







Features on Hcal energy:

- Most of electrons will not reach Hcal.
- muons lose the same energy
- pions lose more energy than muon and lose more energy if their momentum is bigger.

Summary

- Had a look at the energy deposits in Ecal and Hcal for e/mu/pi with different momenta 1-10GeV

- A couple of issues:
  - No track momentum for 1 GeV particles
  - maybe wrong truth matching (e→pi, pi-->e…)

- Work to do (hope to report on next Monday):
  - design identification criteria for electrons and muons
  - calculate the ID efficiency
  - meanwhile, fix the issues above
  - update with including Fangyi's updates