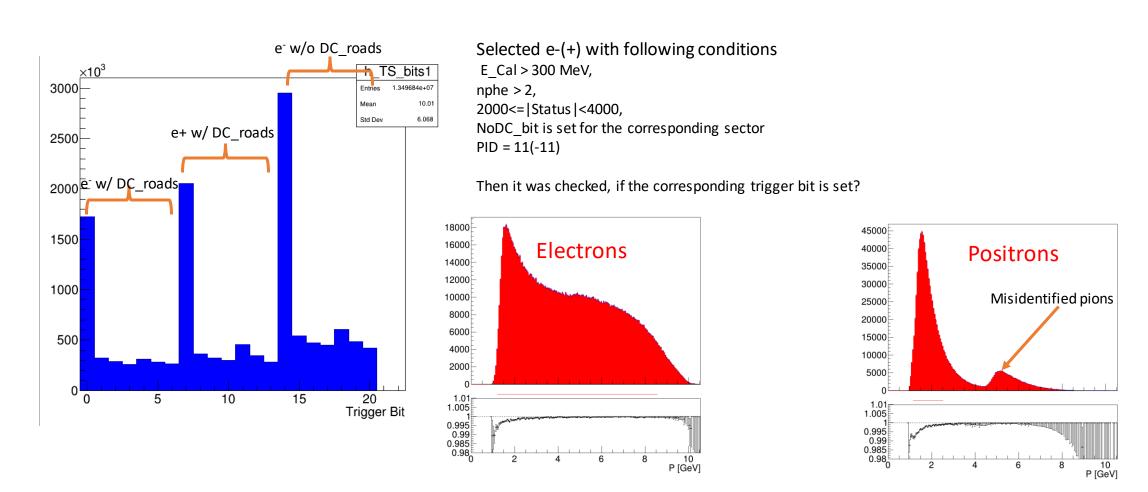
Trigger validation

- Electron Trigger without DC_Road: Doesn't distinguish between electrons and positrons
 - E_PCal > 60 MeV AND E_ECal > 300 MeV AND HTCC signal in the same sector
- Electron Trigger with DC_Roads
 - Same as Electron trigger AND An In-bending Road in the same sector
- Positron Trigger with DC_Roads
 - Same as Electron trigger AND An Out-bending Road in the same sector
- Electron without DC_Roads is validated using the random trigger run:
 - Offline events were reconstructed, and checked whether the trigger bit for the electron in the given sector is set
- Electron/Positron with DC_Roads is validated using a run with "Electron without DC_Roads" trigger:
 - Electrons/Positrons are reconstructed in offline, Made sure the "e- Without C_Road" bit is set, and then it is checked whether the "e-/e+ with DC_Road" bit is set for the given sector.

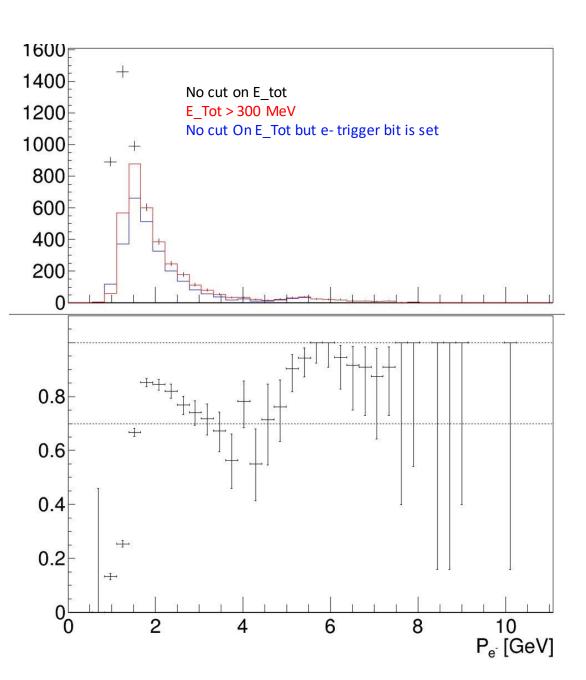
Validation of e-/e+ With DC_roads

Run 16122



The DC_Road efficiency is 99%+ across the whole momentum range At smaller momenta (below 2 GeV), it gets lower, but still higher than 99% Equally good for both: electrons and positrons

Electron-trigger without DC_Roads: Using positrons



Runs: 16123, 16124 and 16125

Initial selection:

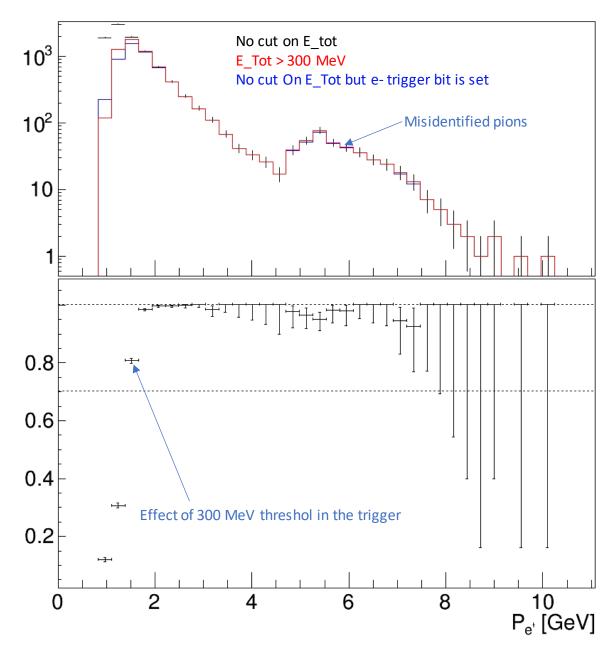
- * Pid == -11
- * basic fiducial cuts
- * 2000 <= | Status | < 4000

20% - 30% inefficiency in between 2GeV to 5 GeV.

Thanks to Ben!, who browsing those failed events, first noticed that HTCC sector is different from PCal/ECal sectors.

In the trigger all those should be in the same sector, so there is a good reason why the trigger bit is not set for those events.

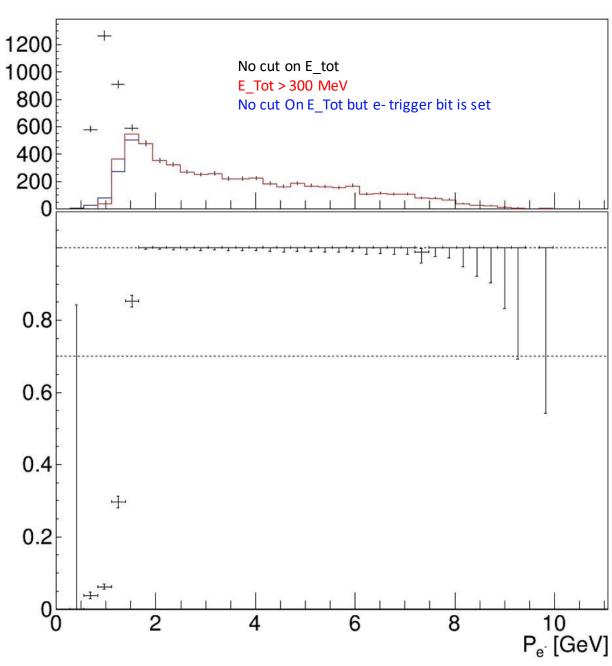
Requiring same sectors for HTCC,PCal, EC_in



Same thing, but with an additional condition that HTCC sectot, PCal sector and EC_inner sector of the reconstructed electron should be the same.

At 1.75 GeV the efficiency is already very close to 100%

Electron-trigger without DC_Roads: Using electrons



Initial selection:

- * Pid == -11
- * basic fiducial cuts
- * 2000 <= | Status | < 4000
- * HTCC, PCal and EC_in have the same sector