

---

Subject: [FIXED] Mc Truth Match

Posted by [Klaus Götzen](#) on Mon, 19 Aug 2013 14:20:55 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Hi,

I just did a quick MC truth check with the tutorial macro. Therefore I filled three lists 'piall', 'mcpj', and 'pimatch' with

```
pimatch.Cleanup();
theAnalysis->FillList(mcpj,"PionLoose","PidAlgoIdealCharged");
theAnalysis->FillList(piall,"PionAll");
```

```
for ( int ii=0 ; ii<piall.GetLength() ; ++ii )
    if ( theAnalysis->McTruthMatch(piall[ii] )
        pimatch.Add( piall[ii] );
```

In 'mcpj' selection is based on PidAlgoIdealCharge, 'piall' contains all charged tracks with pion PID assigned, and 'pimatch' is selected from this with the McTruthMatch methode.

Then I counted the number of candidates in 'pimatch' and 'mcpj', both with and without McTruth object assigned. The result looks like this for 1000 events of the tutorial channel:

pi:1443 (with McTruth:1443) mcpj:1521 (with McTruth:1521)

As you can see, in my test all selected candidates had an attached McTruth object. Second thing is, that there is a discrepancy in the number of candidates in both methods. I think I found the reason for this.

Let's therefore take a look to the candidates in all lists for one case of mismatch, where for each candidate it is shown [track number; PDG; charge; PID probs; PDG of truth object]:

```
pimatch(1) : [#2; 211; C=1; P=(0;0;1;0;0;); True=211]
mcpj(2)   : [#1; 211; C=1; P=(0;0;1;0;0;); True=-211] [#2; 211; C=1; P=(0;0;1;0;0;); True=211]
```

```
piall(4)  : [#0; -211; C=-1; P=(0;1;0;0;0;); True=13] [#1; 211; C=1; P=(0;0;1;0;0;); True=-211]
           [#2; 211; C=1; P=(0;0;1;0;0;); True=211] [#3; 211; C=1; P=(0;1;0;0;0;); True=-13]
```

As one can see, in 'pimatch' the PDG code of the selected object is exactly the same (including charge!) as of the corresponding McTruth object, whereas in 'mcpj' for track #1 PDG and true PDG have opposite sign.

So the reason for the difference is, that tracks with (for whatever reason) mis-reconstructed charge are accepted by ideal PidAlgo, since this one only requires certain pion PID probability without knowledge of charge, whereas the McTruthMatch performs an exact match including correct charge.

Therefore I guess, one should use the McTruthMatch methode or compare the reco PDG code to McTruth PDG code in order to do a proper counting and efficiency measurement.

Best,  
Klaus

---