Optimal pi0 Photon Cuts – v2

Shyam Chauhan (Graduate student at OU)

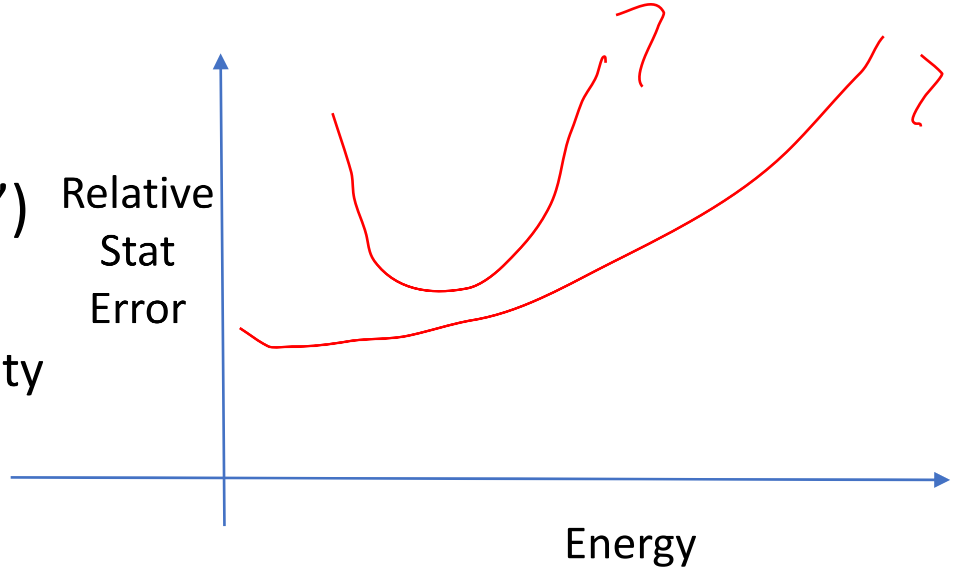
Advisor: Justin Frantz

Note:this presentation assumes you have seen our previous presentation entitled “Optimal pi0 Photon Cuts v1” in the the Aug 9 calo calibrations mtg

Optimizing energy cuts

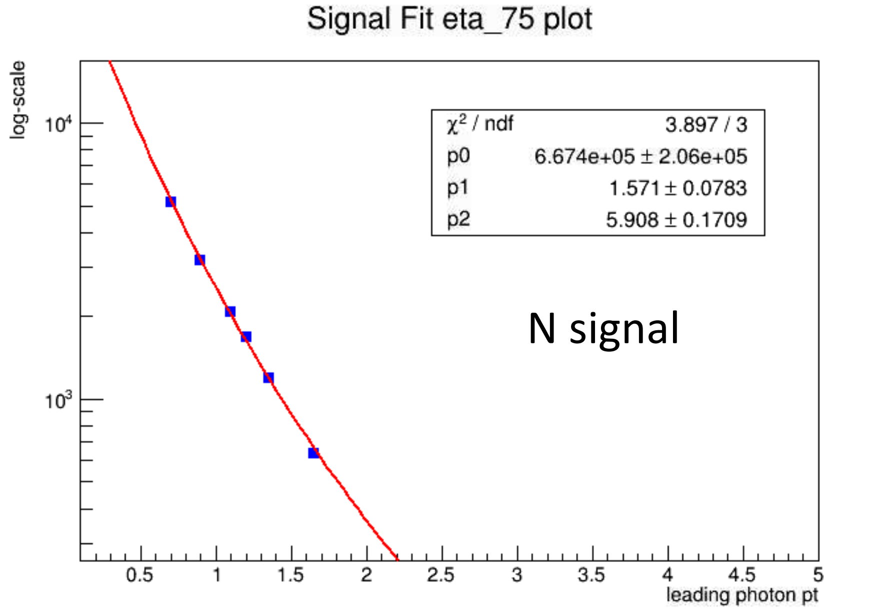
1) not addressed today but rough ideas from current fit method developments

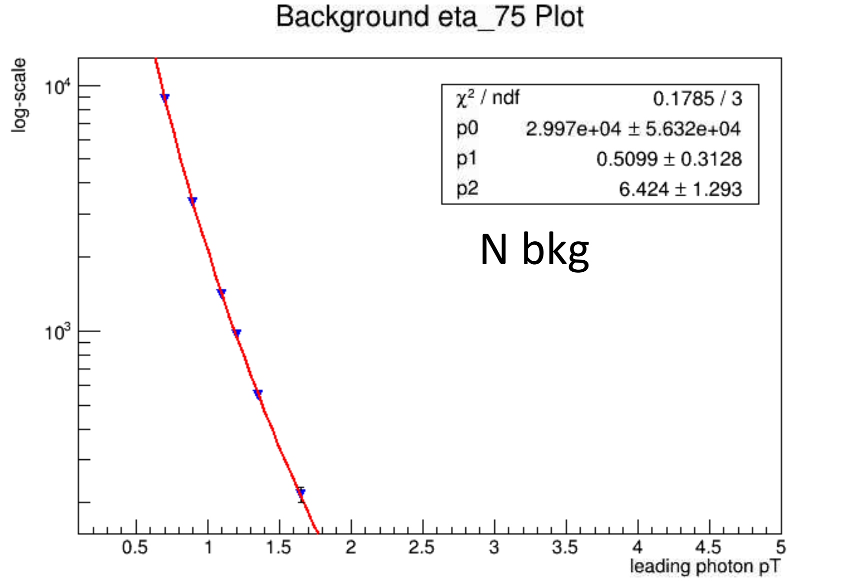
- Two goals:
- 1) Peak fitting robustness to eliminate/minimize human interaction each iteration
  - Most important consideration is S/B (B is mostly combinatoric background)
  - Drives cuts towards high energy decay photons to get good S/B
- 2) Maximize statistical precision per event to minimize number of events needed
  - Drives cuts towards lower energy … e.g. lower than fit considerations
  - statistical precision of fit: S/B: if Comb. Bkg increases fast enough, faster than Signal, as energy is lowered, there may be optimal energy cut values for optimizing statistical precision
- Address 2): want a fairly simple estimator of relative statistical precision/error vs energy cut values without relying on *actual* fitting
  - too slow to do actual analyses for many cut values
- With MDC2 pass used so far, minimum photon E cut for either photon is ~1.1 GeV—call this photon 2 E ("E2") cut (new pass 0.6 GeV cut being analyzed)
  - There are also energy asymm α cuts to vary (α= |E1-E2|/(E1+E2)) but leave this as const < 0.5 for simplicity
- **First look: Vary only photon1 E ("E1") cuts**
  - Pi0 Pt/energy ~ E1+1.1 e.g. for 1.3, pi0 pt/E ~= 2.4 GeV



Fit Signal and Background as fn of energy

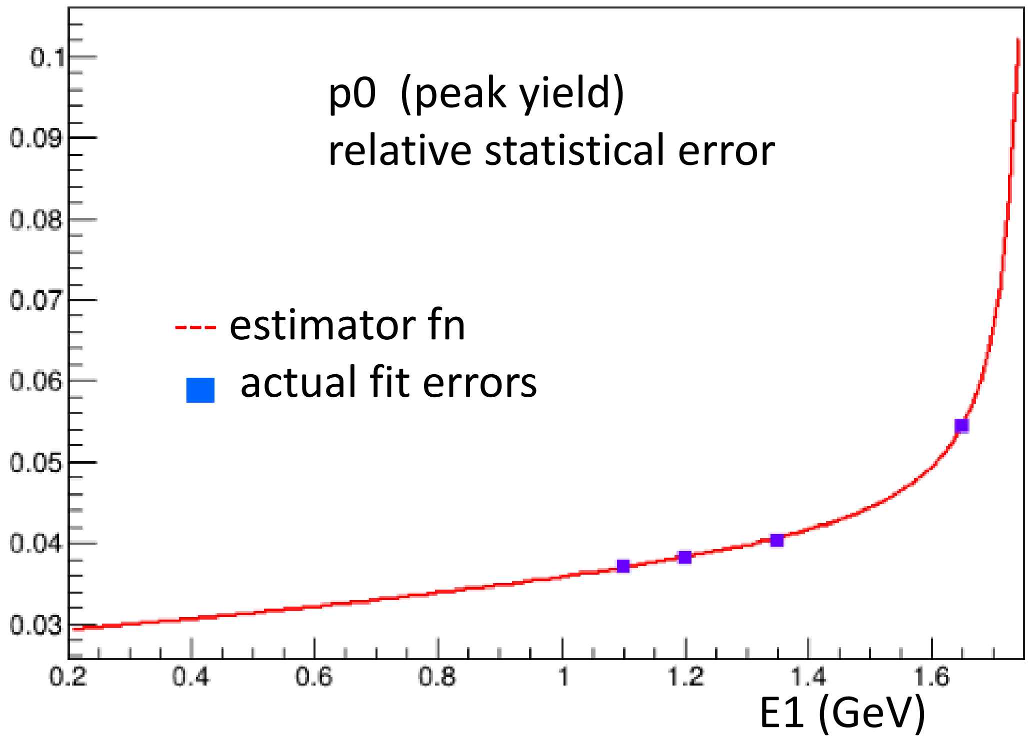
- For Six E1 cut values, estimate S and B at all cut values with simple sideband estimates
  - Avg low and high band = Bkg
  - Peak integral = S+B
  - S = Peak integral - Bkg
- Fit dependence of N signal and N bkg vs E1 (actually Pt1)
  - Previous to do (now done): now we use more realistic e.g. (modified) power law **dN/dE = p0/(p1+E1)^p2**
- Extract Fit Fns of S(E1) and B(E1) at all E1.
- We want to minimize stat error of peak mean (gaus “p1” param—but need formula to avoid actual fit)
  - not directly minimizing S/B
  - Need estimator fn f(S,B) of peak mean error

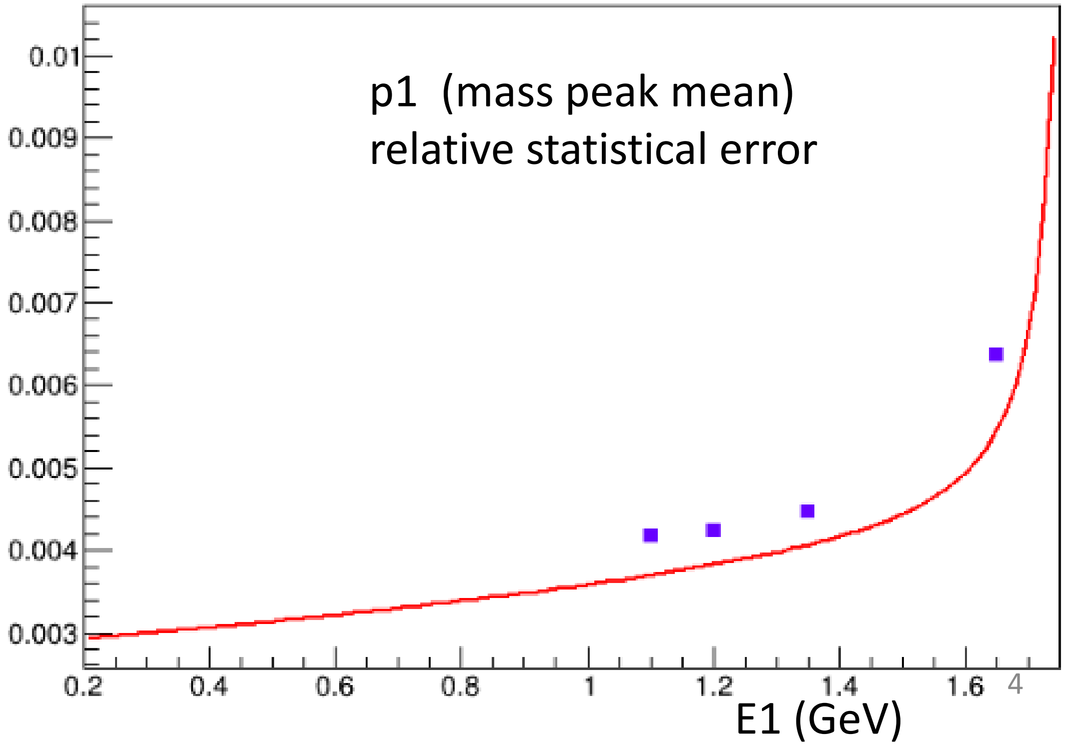




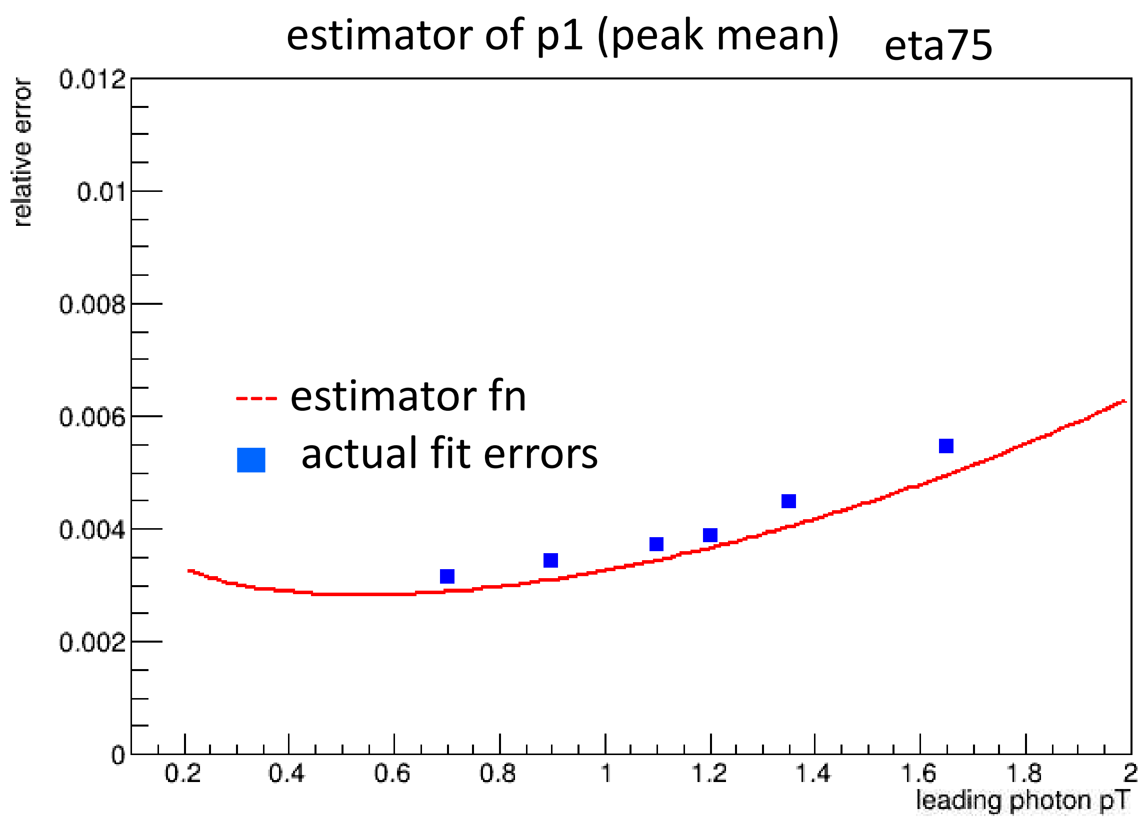
First results (From previous presentation)

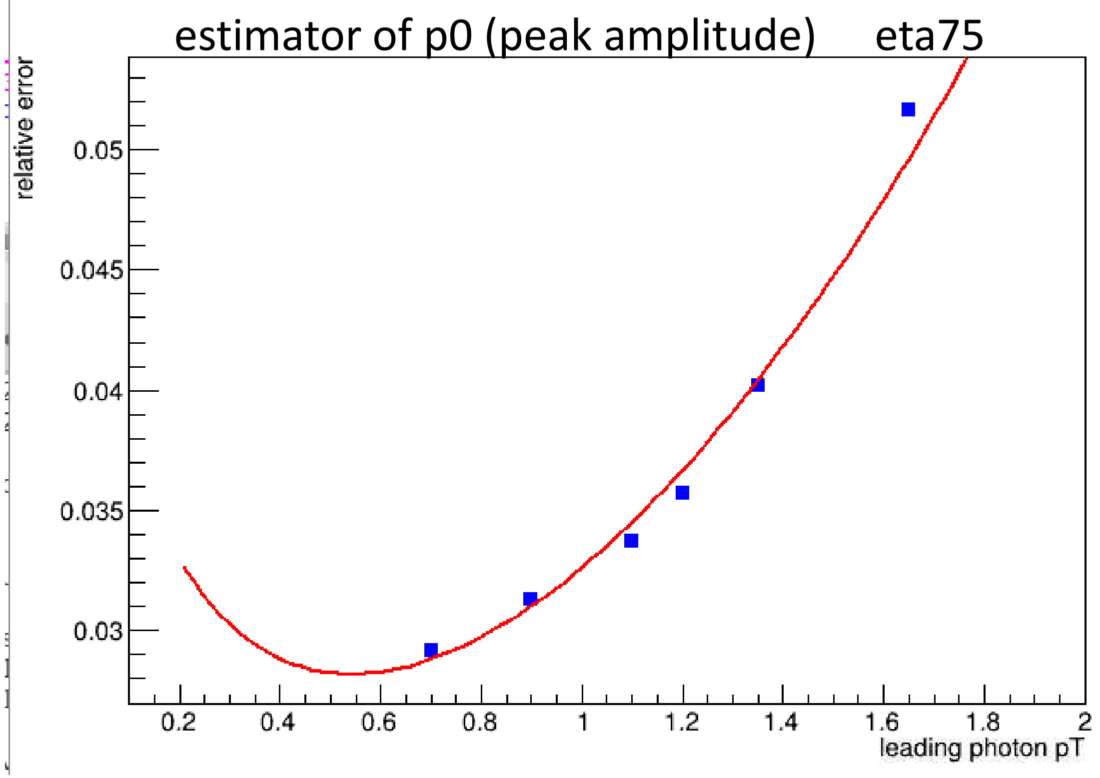
- **Ignoring energy dependence of peak sigma p2** (expected from energy resolution) by setting p2/p1 = const = 0.1 (true for all blue points), test minimal error from energy dependence of S and combinatoric B alone:
- First Results: No local minimum in E1, but error flattens as E1 decreases**—gain in statistical precision from going lower in E1 small, outweighed by fit robustness**
  - sqrt(S+2B)/S very good estimator of p0 error values
  - p2*sqrt(S+2B)/S*p1: ~correct shape for p1 error
- Lower than ~0.9 not trustable…B.C. THESE WERE DONE WITH POLYNOMIAL FITS we are studying effects of cuts (new pass with much lower min photon E2 cut), S/B functional forms, and energy resolution to push lower, see if local min





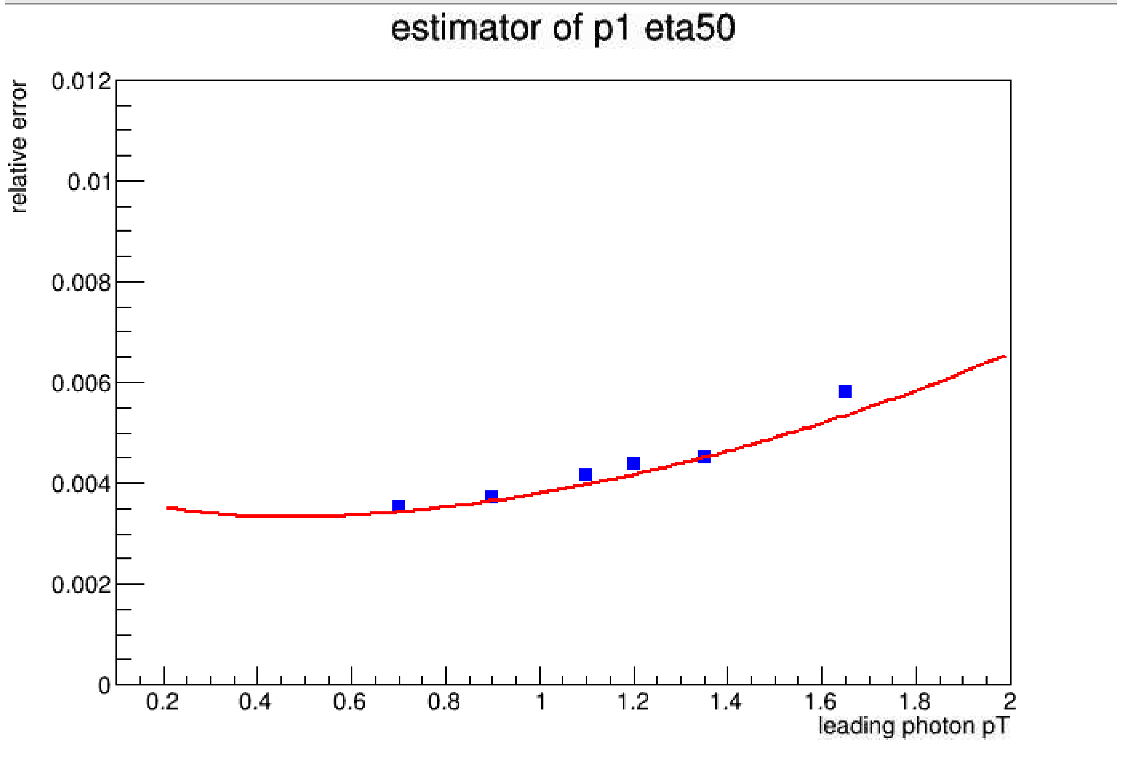
Results from this time

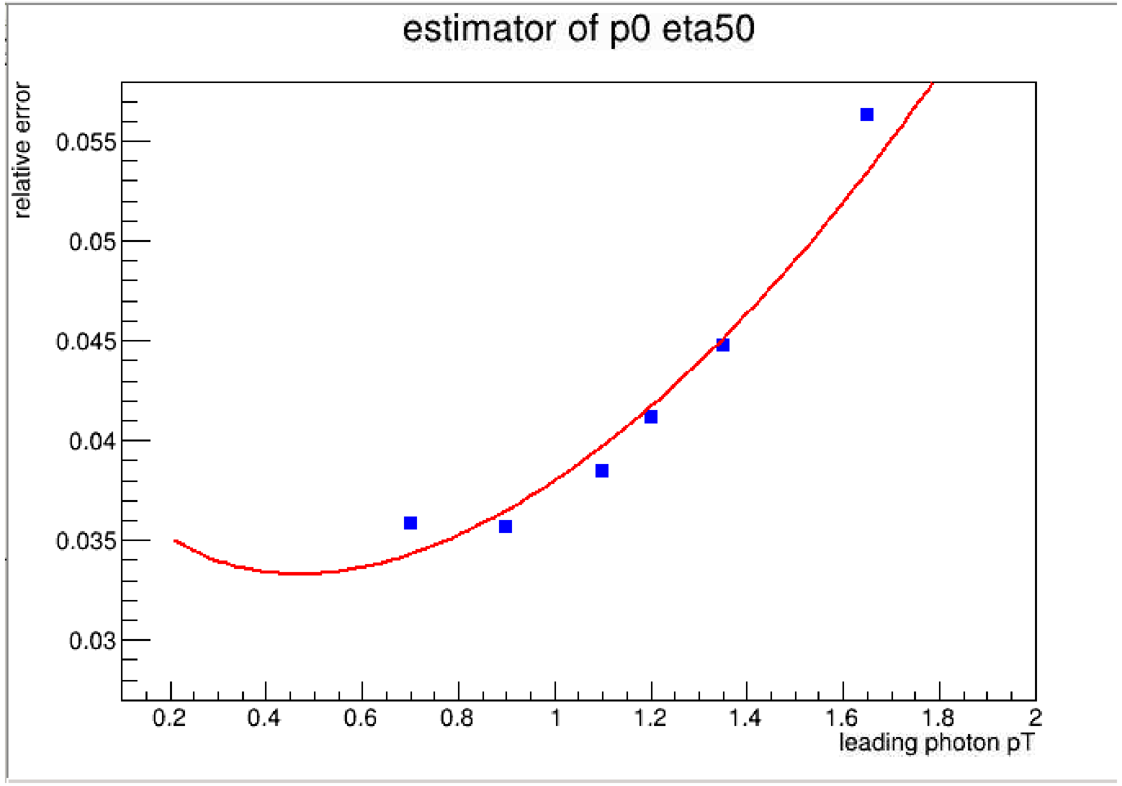




- use new pass of MDC2 with 0.6 GeV cuts (could go lower in energies for this study)
- Effect of better S, B, fit functions and more points allows reliable extrapolation over ~whole range shown

Another eta slice: eta 50 ~ η = 0 (mid-rapidity)

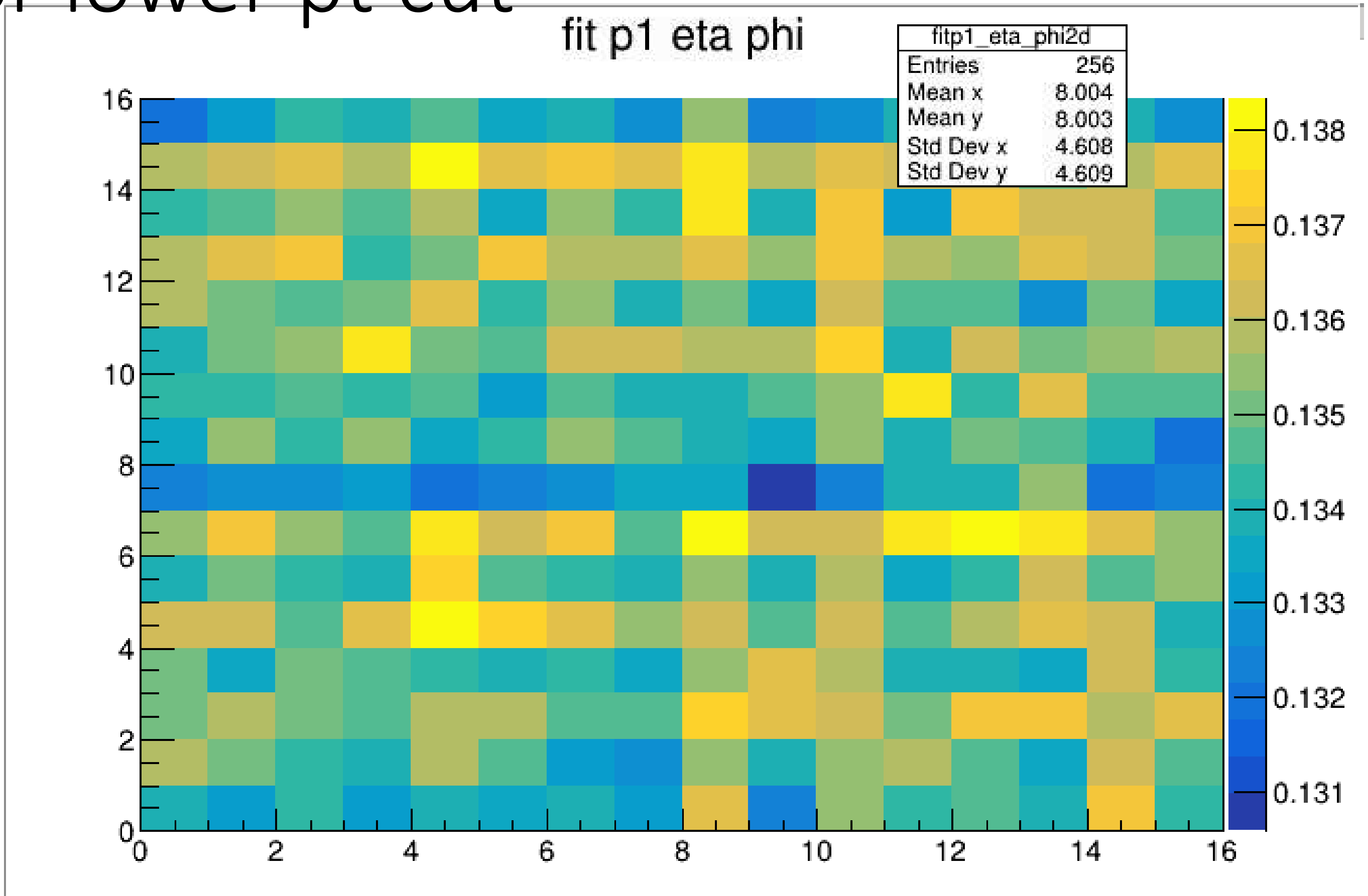




- sqrt(S+2B)/S very good estimator of p0 error values
- p2*sqrt(S+2B)/S*p1: ~correct shape for p1 error
- We do see the minimum relative error (we want to be within 10-15% away from it for our other analysis), keeping in mind we don't include lots of unwanted events that will simply won't increase our statistical precision
- maybe use ~0.9 GeV-1 GeV ?

Fit reliability for lower pt cut

- New pass also allows to test fitting reliability at lower cut value of 0.9
- New pass uses ~40,000 mdc2 – 4 files double stats as before
- 256 "tower groups" inv mass distributions: not failing any fits with 0.9 GeV photon cuts
- TODO: explore reliabilty with 24k fits need embedding:
  - Plan: put this in large scale testing with Chris Pinkenburg tests?



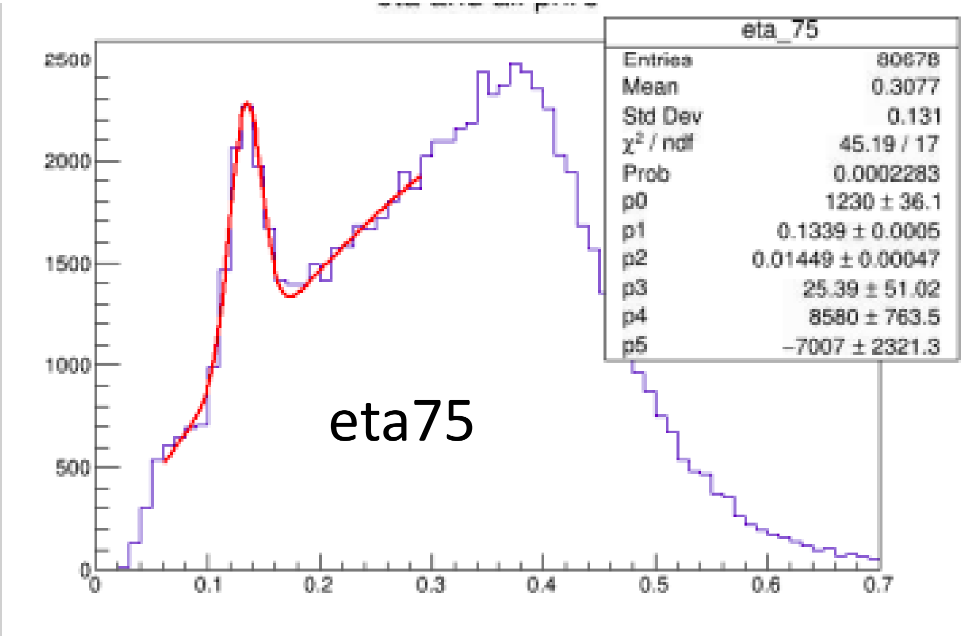
Each square in this 16 x 16 (=256) grid is the pi0 peak mean fit value from a combined invariant mass distribution combined from independent groups of 96 towers each, combined from across entire EMC in a way shown in Aug 9 presentation and in the backup - with decalibration factors varying randomly in each

Backup

- First some requested mass plots added after the presentation
- Then some backup re-explaining some points from Aug 9 presentation

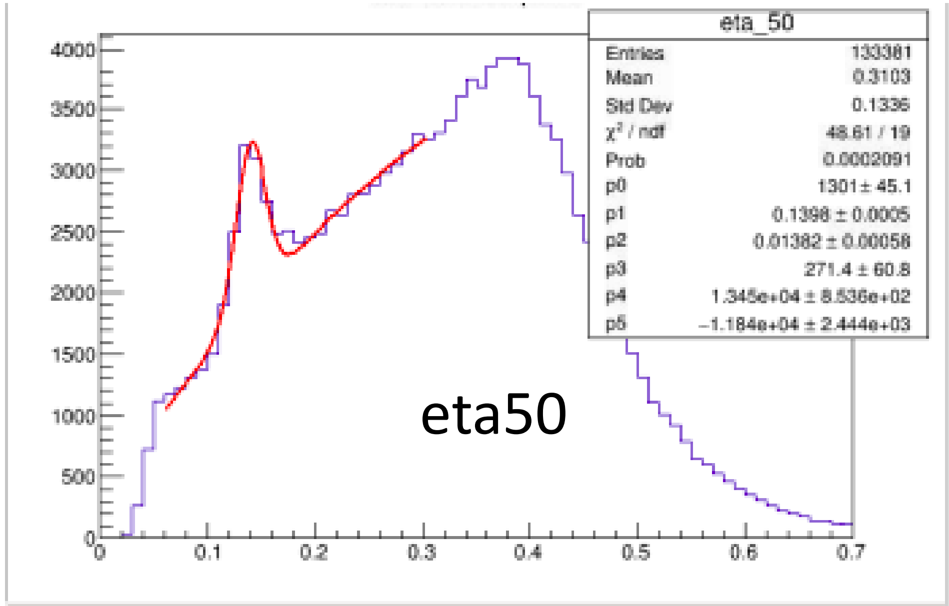
Mass plots photon pt1 cut = 0.7 GeV/c, eta75 (eta~0.5) eta50 (eta ~0)

Except for 2D plot at end, all results in presentation are done in eta slices ie intregated over 256 phi bins (x256 stats) for each eta bin out of 96 (0-96)




```
N1 counts: 3637
N2 counts: 13932
N3 counts: 13905
p0: 1230.28
p0 error: 36.0887
p1: 0.133944
p1 error: 0.000452193
p2: 0.0144931
p2 error: 0.000465436
==================
Signal: 5161
Background: 8771
root [1]
```

photon cut: 0.7

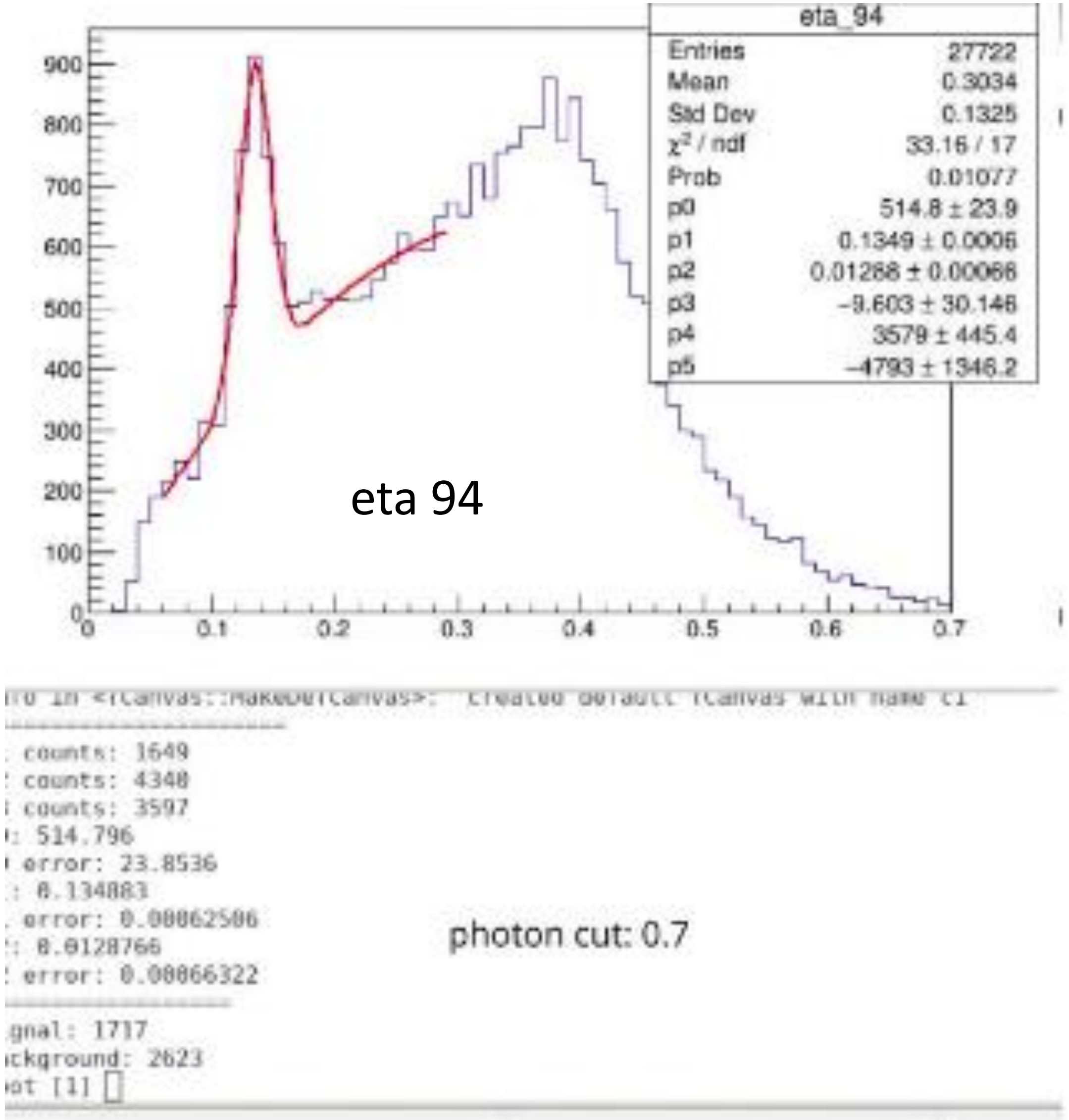



```
Processing macro_to_get_data_for_google_sheet.C("corr_v14_0.7.root", 50)...
Info in <TCanvas::MakeDefCanvas>:  created default TCanvas with name c1
==================
N1 counts: 8695
N2 counts: 19929
N3 counts: 20763
p0: 1300.99
p0 error: 45.1052
p1: 0.139826
p1 error: 0.000528371
p2: 0.013824
p2 error: 0.000582144
==================
Signal: 5200
Background: 14729
root [1]
```

photon cut: 0.7

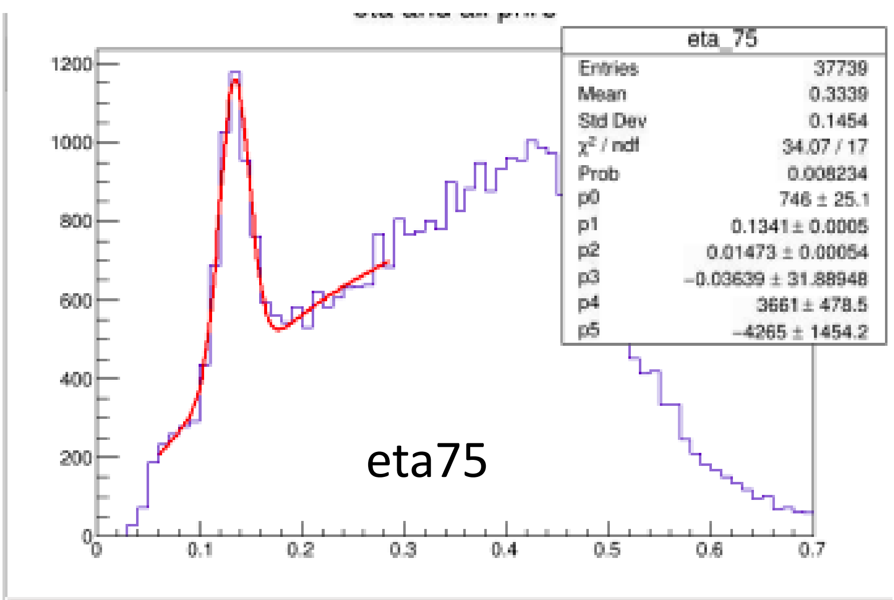
As discussed during the talk cutting pt1 rather than E1 reduces the eta dependence, not very different fitting systematics but there still is some small difference in S/B

Eta 94 (eta =~+1.0), pt1 cut = 0.7 GeV/c



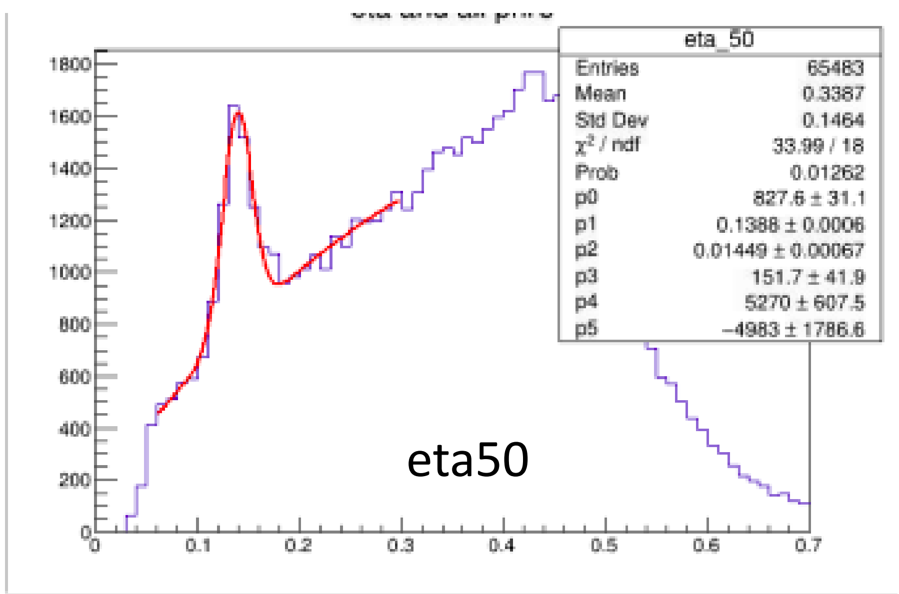
part of apparent better S/B is probably due to better energy/mass resolution at high eta

pt1 cut= 0.9 mass distributions




```
Processing macro_to_get_data_for_google_sheet.C("corr_v14_0.9.root", 75)...
Info in <TCanvas::MakeDefCanvas>:  created default TCanvas with name c1
======================
N1 counts: 1355
N2 counts: 6493
N3 counts: 5295
p0: 746.007
p0 error: 25.1044
p1: 0.134139
p1 error: 0.000507345
p2: 0.0147345
p2 error: 0.00053771
================
Signal: 3168
Background: 3325
root [1]
```

photon cut: 0.9

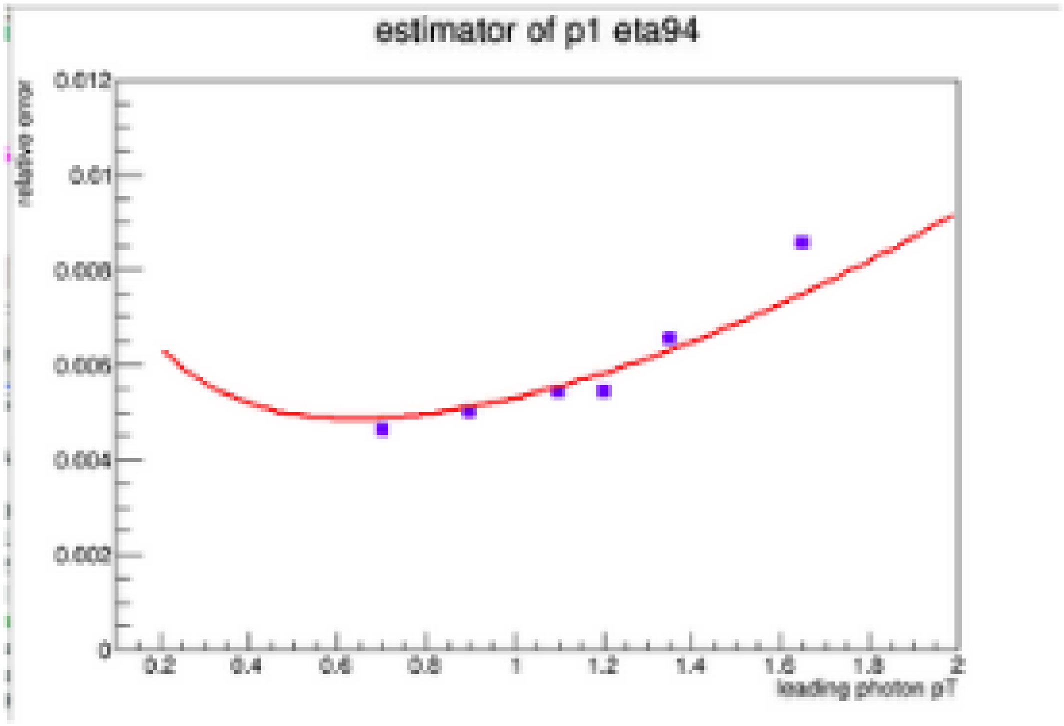


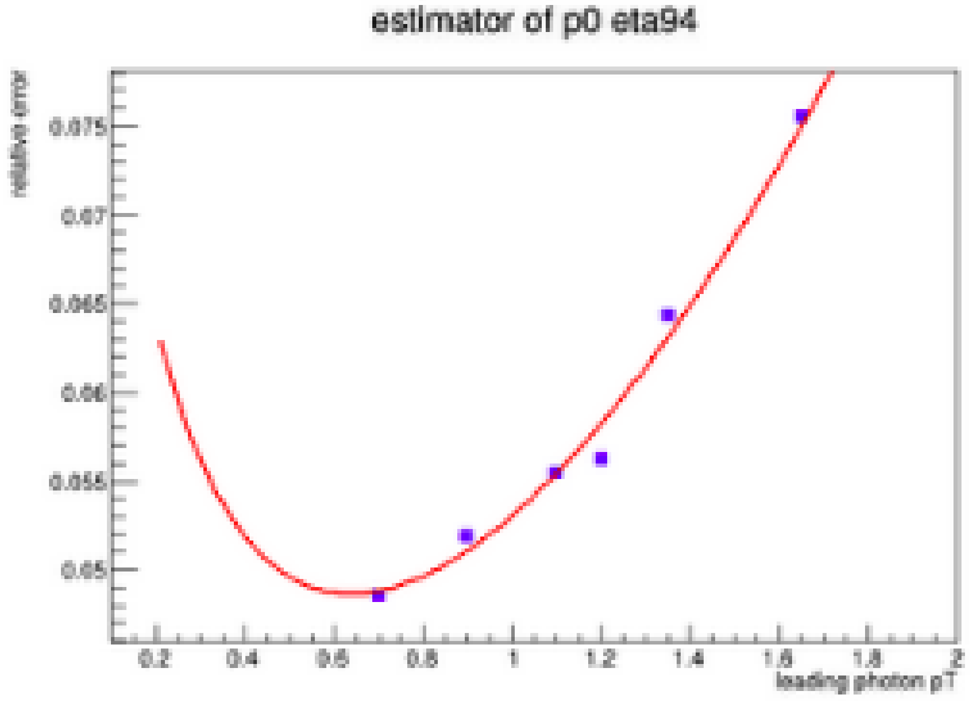

```
Processing macro_to_get_data_for_google_sheet.C("corr_v14_0.9.root", 50)...
Info in <TCanvas::MakeDefCanvas>:  created default TCanvas with name c1
======================
N1 counts: 3585
N2 counts: 9398
N3 counts: 8348
p0: 827.623
p0 error: 31.088
p1: 0.138813
p1 error: 0.000574499
p2: 0.0144878
p2 error: 0.000667771
================
Signal: 3471.5
Background: 5926.5
root [1]
```

photon cut: 0.9

Whole S/B optimization plot for eta 94

Very similar behavior as expected





From last presentations

Estimator fn

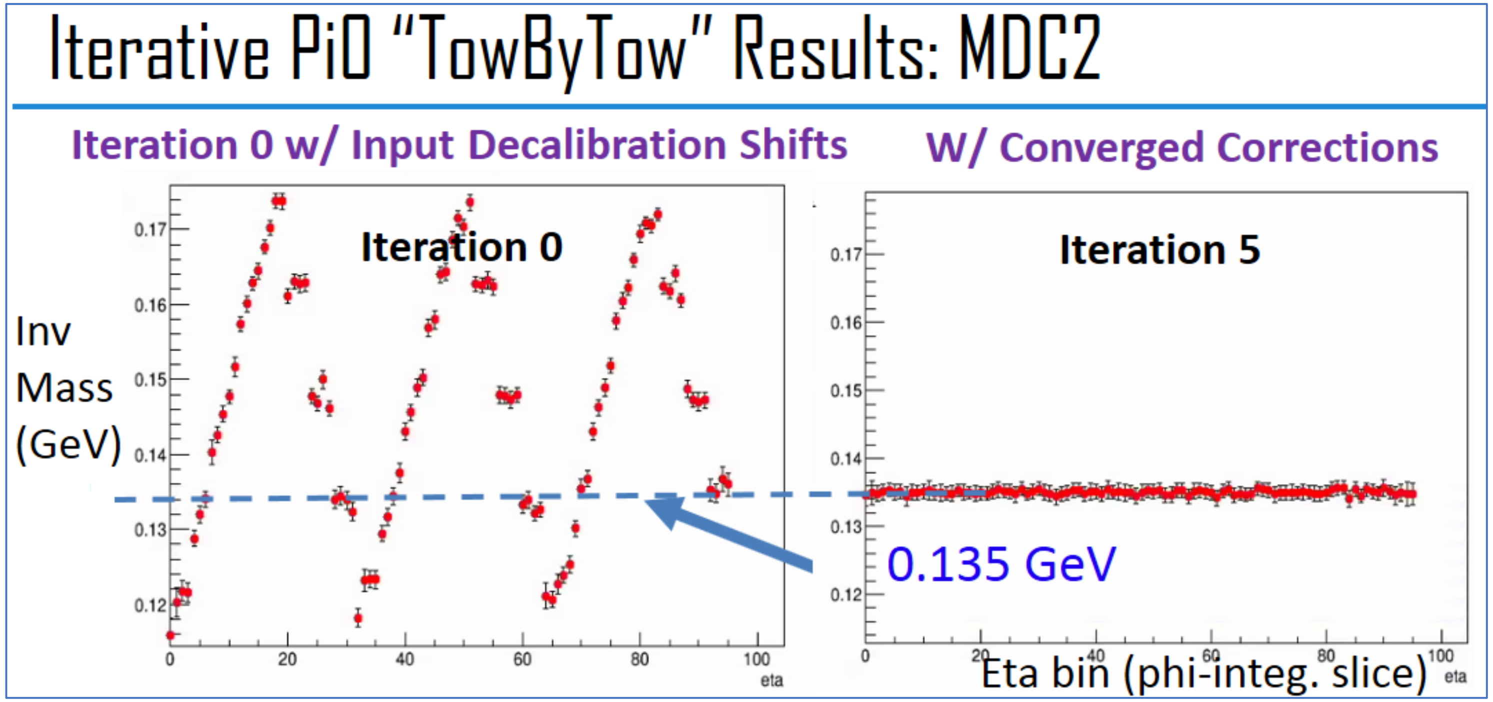
- Naïve relative error of peak **yield** assuming poisson sqrt(S+B)/S
- Better formula known (e.g. CKB thesis*) to be ∝ sqrt(S+2B)
- Found that sqrt(S+2B)/S best estimator of peak **yield** relative error = 1/ sqrt("N_measments")
- For peak **mean** error, use std error of mean formula : std_dev / sqrt("N_measurements")
- peak mean relative error → divided by ~0.135 gaus fit parameters : f(S,B) = **[p2* sqrt(S+2B)/S]/p1**
  - comparison to p1 error from fit, relatively good, ~10% below actual errors– across several cut, eta values

	rison of p0			Comparison of p1					
histo number	relerr3 = sqrt(S+B)/(S+B)	relerr4 = sqrt(S+2B) / S	relerr5 = sqrt(S+2B) / (S+2B)	relerr1_p1 = (p2/p1) * (1/sqrt(S))	rel_errfit = (fit error on p1) / p1	relerr2_p1 = (p2/p1)*(sqrt(S+B)/S)	relerr3_p1 = (p2/p1) * (sqrt(S+B)/(S+B))	relerr4_p1 = (p2/p1) * (sqrt(S+2B)/S)	relerr5_p1 = (p2/p1) * (sqrt(S+2B)/(S+2B))
eta_25 (1.1, 1.1)	0.009970	0.043895	0.007638	0.0019744	0.0567241	0.0036263059	0.00107499	0.004733070	0.00082362
eta_25 (1.2, 1.2)	0.011793	0.045553	0.009137	0.0021144	0.0056165	0.0036578294	0.00122224	0.004721088	0.00094698
eta_25 (1.1, 1.3) ??	0.009970	0.043895	0.007638	0.0019744	0.0056724	0.0036263059	0.00107499	0.004733070	0.00082362
eta_25 (1.35, 1.35)	0.014904	0.046890	0.011789	0.0024238	0.0056957	0.0038237317	0.00153646	0.004833927	0.00121537
eta_25 (agg4)	0.033352	0.061947	0.028496	0.0043933	0.0075036	0.0055344146	0.00348748	0.006477524	0.00297971
eta_50 (1.1, 1.1)	0.007549	0.050817	0.005626	0.0017593	0.0067539	0.0039405244	0.00078550	0.005287739	0.00058537
eta_50 (1.2, 1.2)	0.008849	0.049880	0.006662	0.0018941	0.0064750	0.0039019886	0.00091944	0.005182992	0.00069219
eta_50 (1.1, 1.3) ??	0.007549	0.050817	0.005626	0.0017593	0.0067539	0.0039405244	0.00078550	0.005287739	0.00058537
eta_50 (1.35, 1.35)	0.011171	0.052735	0.008513	0.0021244	0.0067229	0.0040292016	0.00112006	0.005287341	0.00085354
eta_50 (agg4)	0.029617	0.062865	0.024720	0.0042102	0.0076105	0.0056038257	0.00316321	0.006714137	0.00264011
eta_75 (1.1, 1.1)	0.009381	0.037073	0.007253	0.0018497	0.0053252	0.0032333813	0.00105816	0.004181874	0.00081816
eta_75 (1.2, 1.2)	0.011078	0.038195	0.008678	0.0020205	0.0054065	0.0033204617	0.00122948	0.004238924	0.00096309
eta_75 (1.1, 1.3) ??	0.009381	0.037073	0.007253	0.0018497	0.0053252	0.0032333813	0.00105816	0.004181874	0.00081816
eta_75 (1.35, 1.35)	0.014028	0.040179	0.011219	0.0023581	0.0053513	0.0035689605	0.00155800	0.004462570	0.00124601
eta_75 (agg4)	0.034021	0.054416	0.029954	0.0047251	0.0069596	0.0056074301	0.00398166	0.006368645	0.00350575
eta_90 (1.1, 1.1)	0.013238	0.040645	0.010501	0.0022450	0.0048840	0.0035034480	0.00143859	0.004416818	0.00114110
eta_90 (1.2, 1.2)	0.014807	0.039247	0.011960	0.0023939	0.0049712	0.0035025978	0.00163611	0.004336561	0.00132147
eta_90 (1.1, 1.3) ??	0.013238	0.040645	0.010501	0.0022450	0.0048840	0.0035034480	0.00143859	0.004416818	0.00114110
eta_90 (1.35, 1.35)	0.018206	0.040847	0.015061	0.0028361	0.0054051	0.0038638335	0.00208176	0.004670638	0.00172216
eta_90 (agg4)	0.042258	0.064684	0.037569	0.0068702	0.0099632	0.0080144949	0.00588922	0.009014716	0.00523579

*Christian K. Boesing PHENIX thesis https://zenodo.org/record/4267413

Review previous MDC2 test on Eta Slices

- ~10-20 M (MDC2) events is not enough statistics to test full pi0 method on all 24 K towers tower by tower
  - Previous test : pseudorapidity (eta) slices - test sensitivity to common decalibration shift value across each eta bin (same for 256) phi bins. Allows 256* more statistics when fitting invariant mass histogram
- Eta slices only share and unfold energy (and decalibration) in "1-D" eta-direction – converges faster than 2-D (eta-phi) unfolding

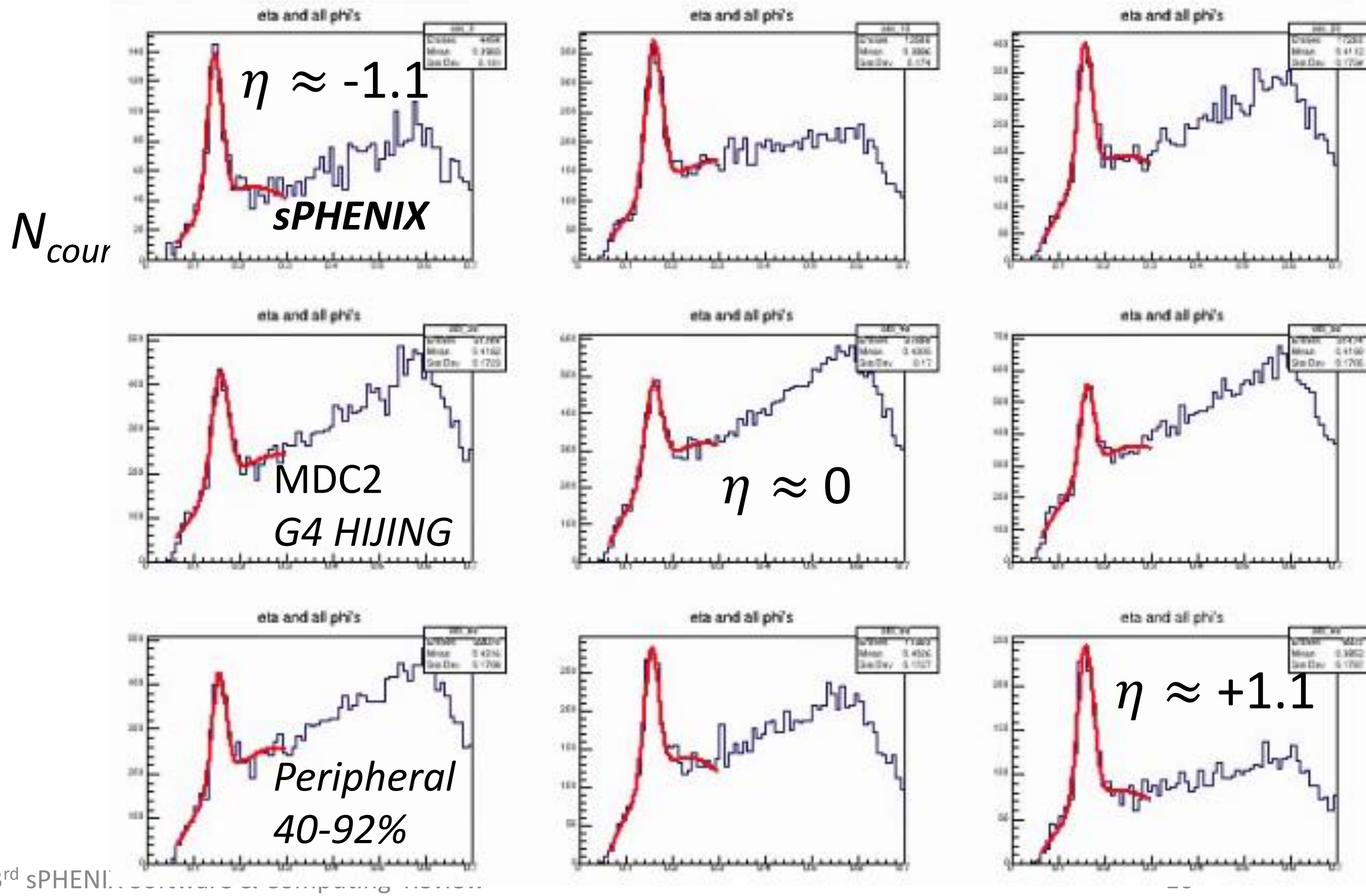


From 2022 S&C review

Iterative Pi0 “TowByTow” Results: MDC2

from 2022 S&C review

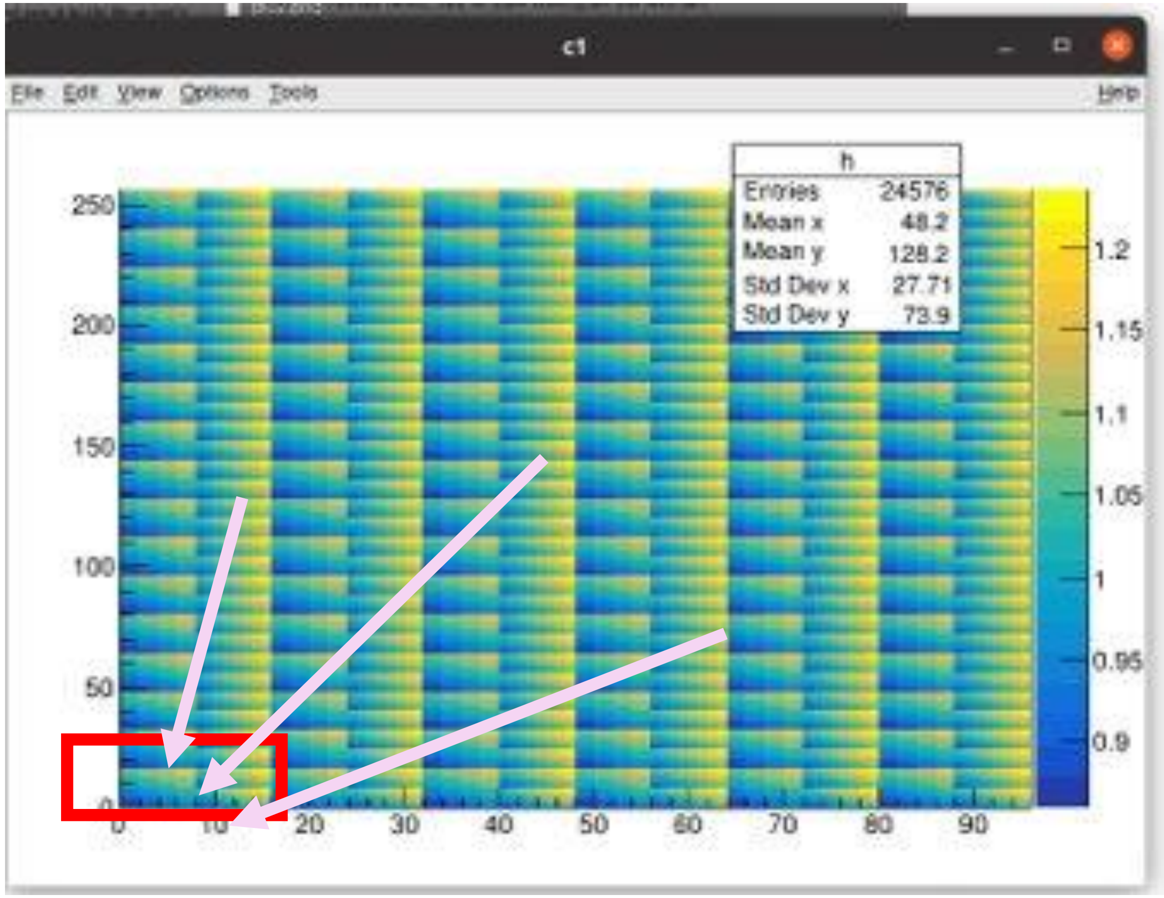
- PHENIX method fully implemented in sPHENIX tested over MDC2
- MDC2: 20M MB ev   This ana using >40% Peripheral.
- MDC2 “full test” over 96 eta slices (phi-integrated) instead of full 24k Towers
- Will also use this procedure for monitoring the eta-flattening on similar size event samples in the online calibration



2D Pattern And “2D” Combined Tower stats

- New test: make a single decalibration shift pattern for 2-D eta-phi 16x16 tower-unit, repeat same pattern across detector
- perform iterations as usual but to detect shift, fit invariant mass histograms with extra statistics from combining all instances of pattern for each tower in the 16x16 unit

256/16 =
16 pattern
instances
along phi



96/16 = 6 pattern instances along eta

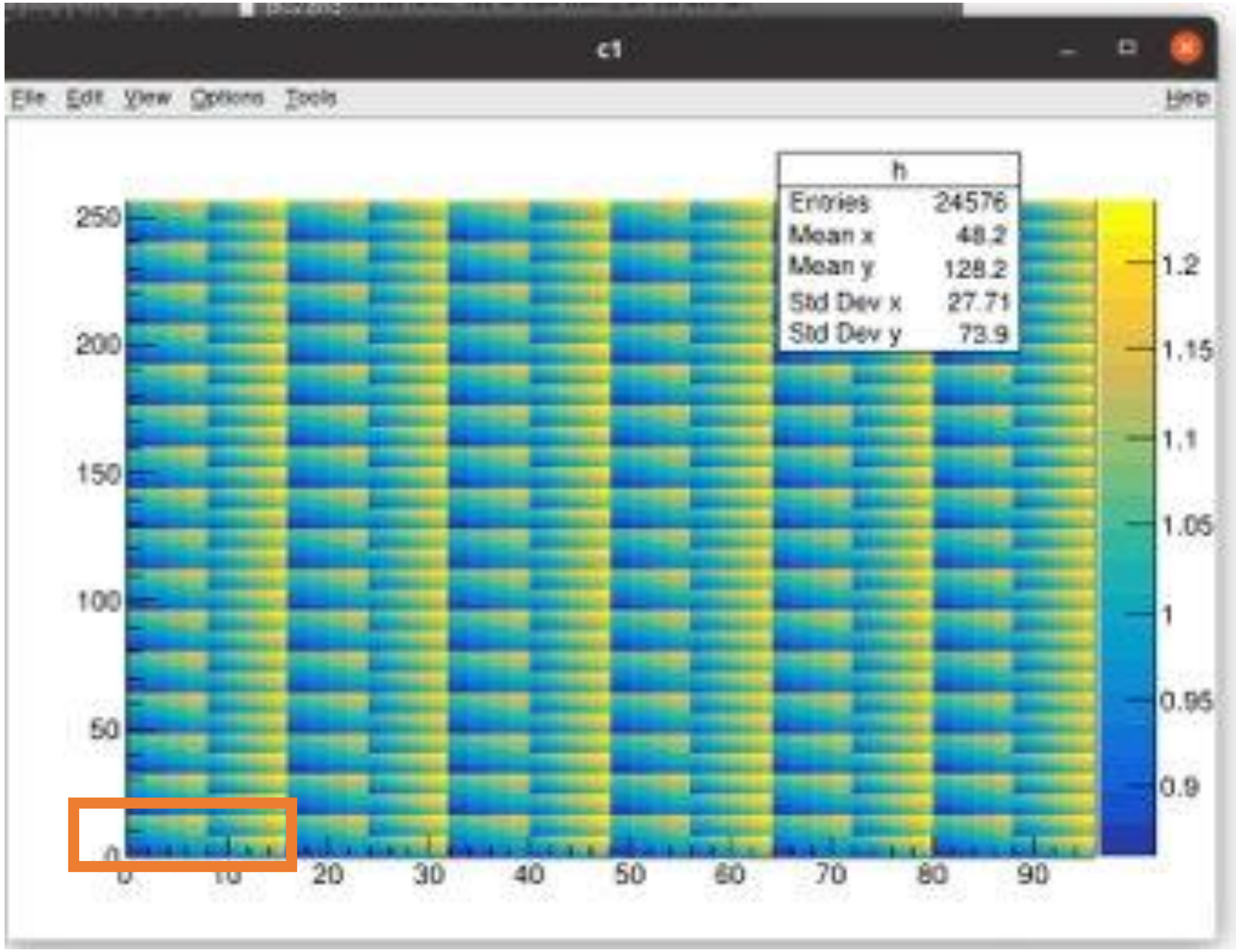
Total statistical increase for each of the 16x16 tower unit
 = times number of instances -->
6x16 = 96 times

Decalibration Shift pattern generation

For reference

Used "worst-case" decalibration shifts ranging from ~0.86 (14% down decalibration) to 1.3 (30% upward shift)

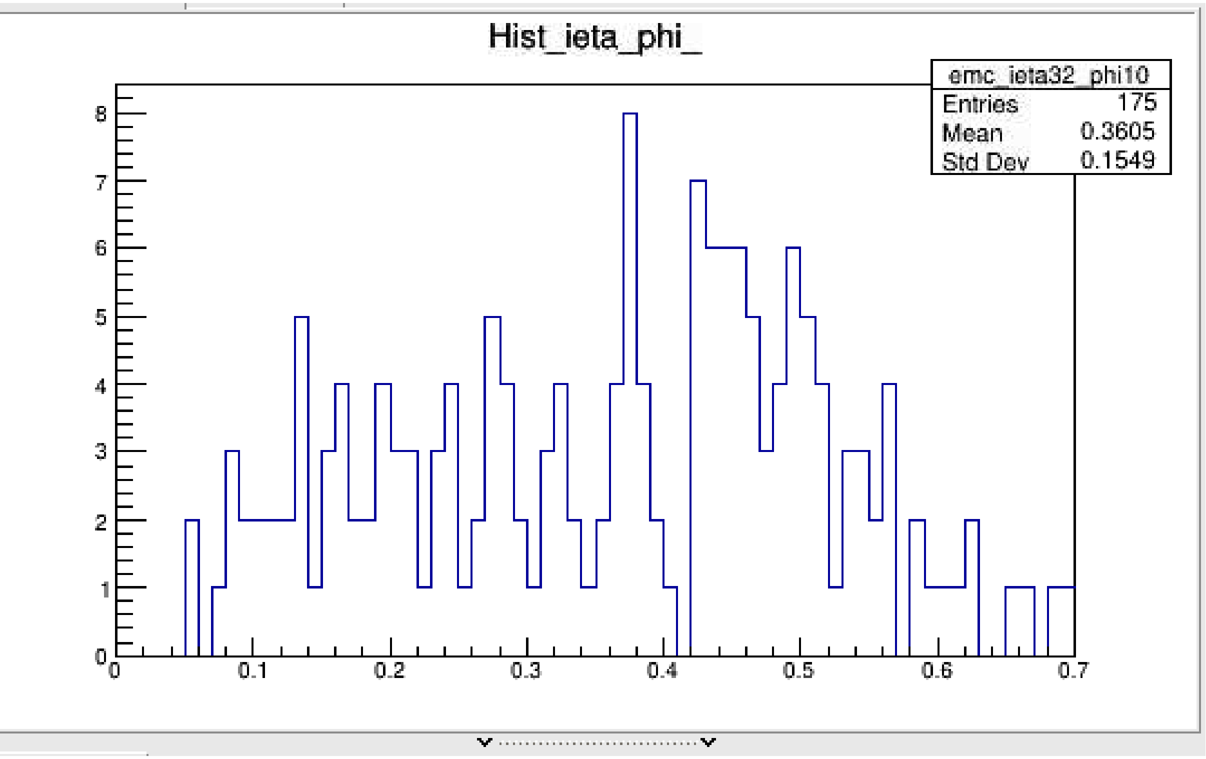
```
void pattern_jf()
{
    TH2F *hist = new TH2F("h", "", 96,0,96, 256,0,256);

    for (int i=0; i<96; i++)
    {
        for (int j=0; j<256; j++)
        {
            float e = 1.00;
            int ir = -999;
            int jr = -999;
            if ( (i>=8 && i<16) || (i>=24 && i<32) || (i>=40 && i<48) ||
              (i>=56 && i<64) || (i>=72 && i<80) || (i>=88 && i<96) )
            {
                ir = i%8;
                jr = j%8;

                e *= 0.885+ir*0.025+jr*0.025;

            }
            else
            {
                int ib2 = i/2;
                ir = ib2%4;
                int jb2 = j/2;
                jr = jb2%8;

                e *= 0.86+ir*0.030+jr*0.030;
            }

          // e *= 0.86+ir*0.03+jr*0.03;
            hist->SetBinContent(i+1, j+1, e);
        }
    }
    hist->Draw("colz");
    //    hist->Draw("text,same");
}
```

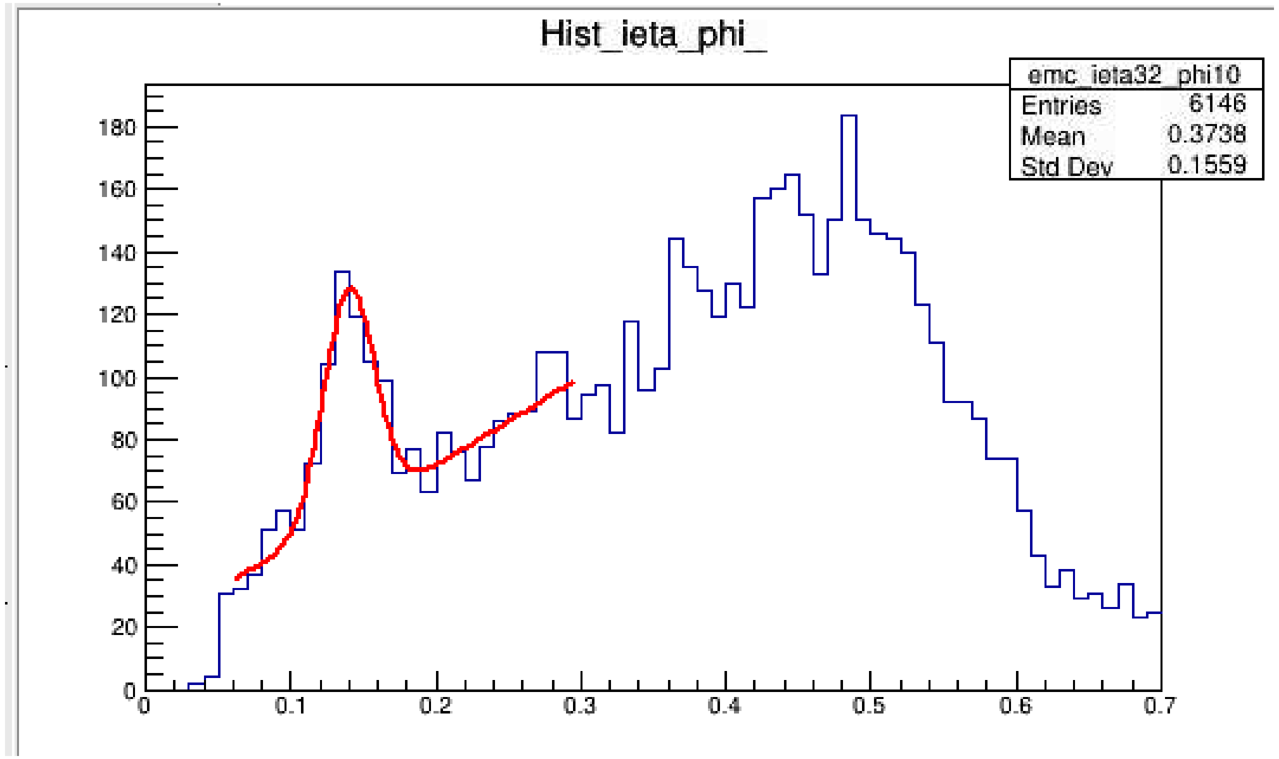


Apply this pattern of decalibration, make invariant mass histograms for each iteration then combine pattern instances

Example of combining statistics

Example: eta0_phi10 tower   ~8M total events



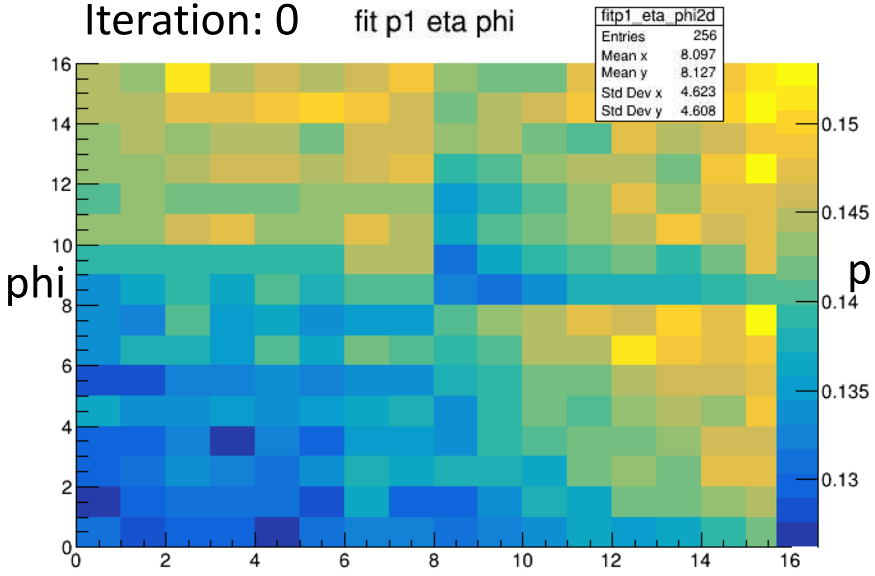


Single tower from all MDC 2 events statistics

Same tower location with combined 96 times statistics

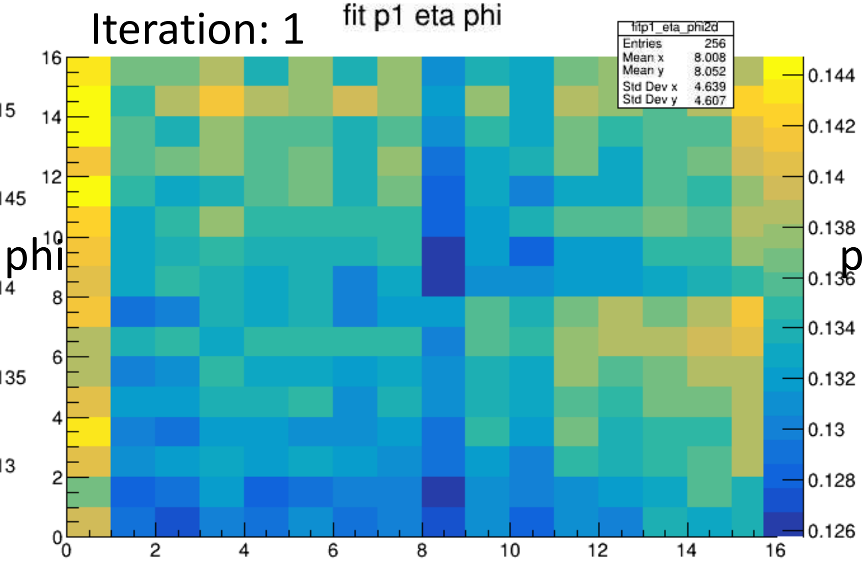
RESULTS   Pi0 Peak Fit Mass Value Distribution   (z-axis / color is mass fit value)

Iteration: 0



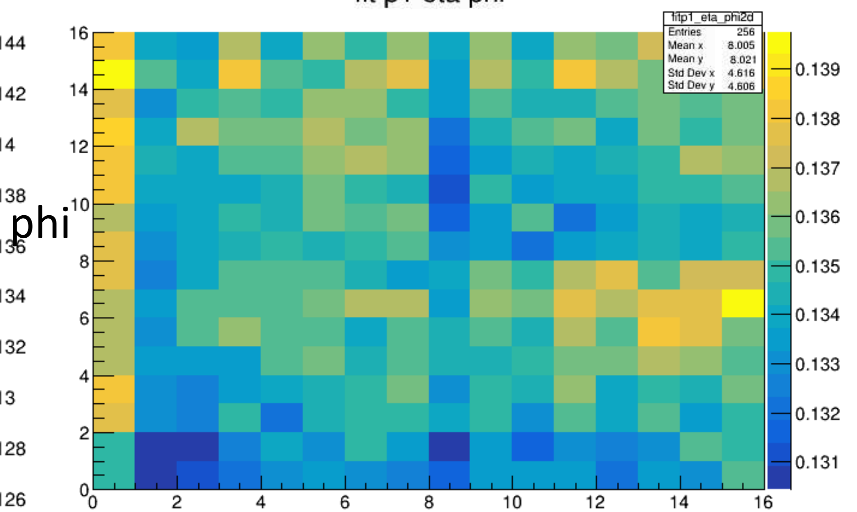
eta

Iteration: 1



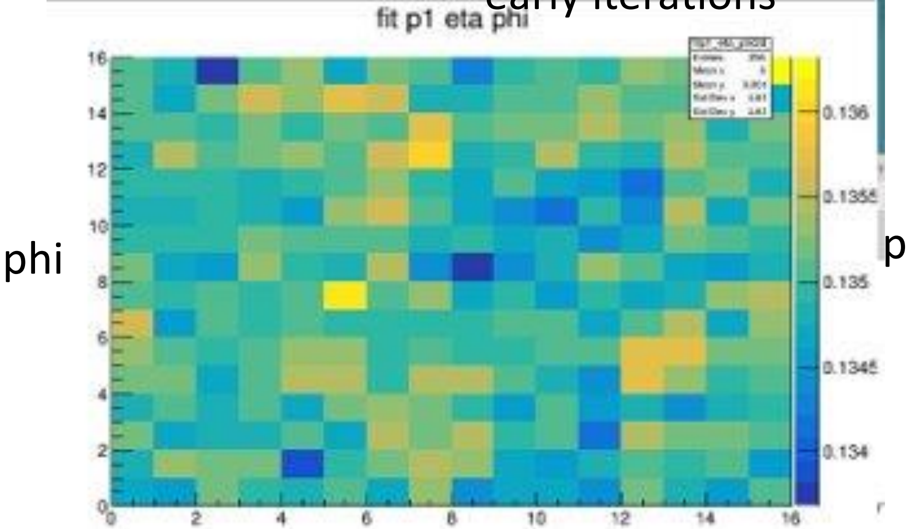
eta

Iteration: 2



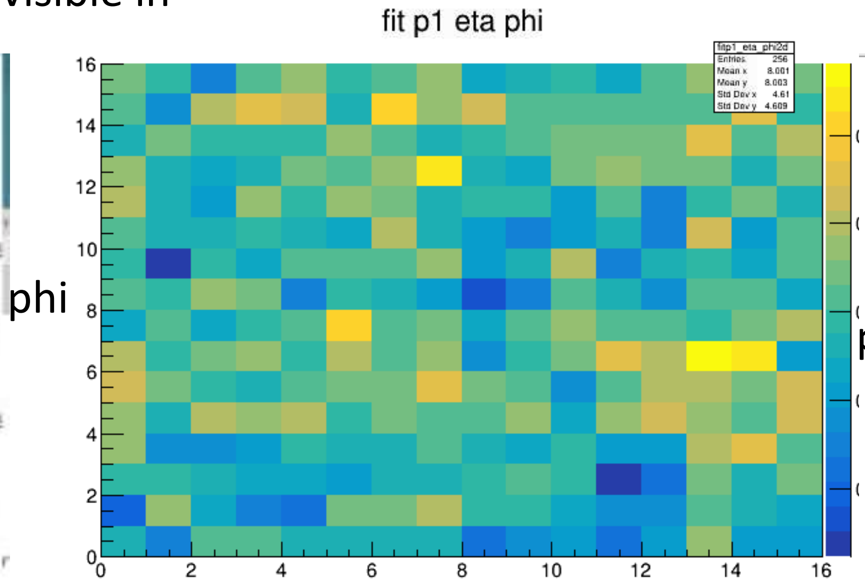
eta

Input Pattern effect visible in early iterations



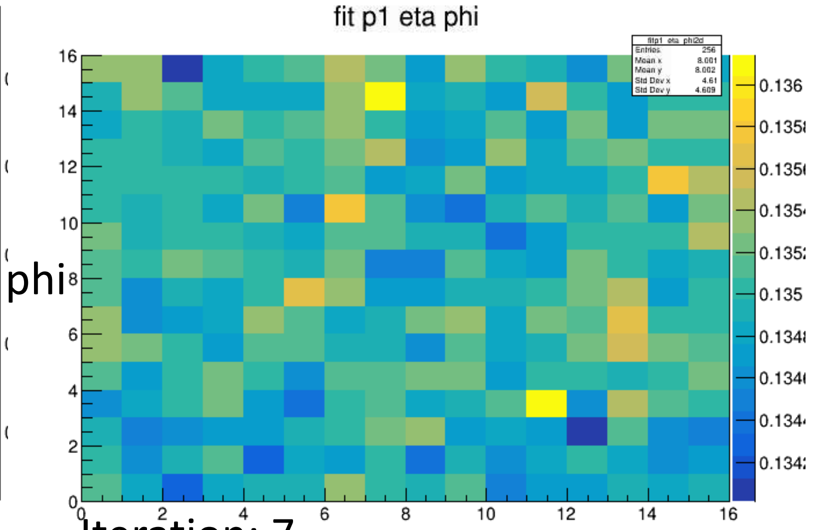
Iteration: 5

eta



Iteration: 6

eta



Iteration: 7

eta

1-Phi Bin Projection: ProjectionX of previous histograms for phi=10: