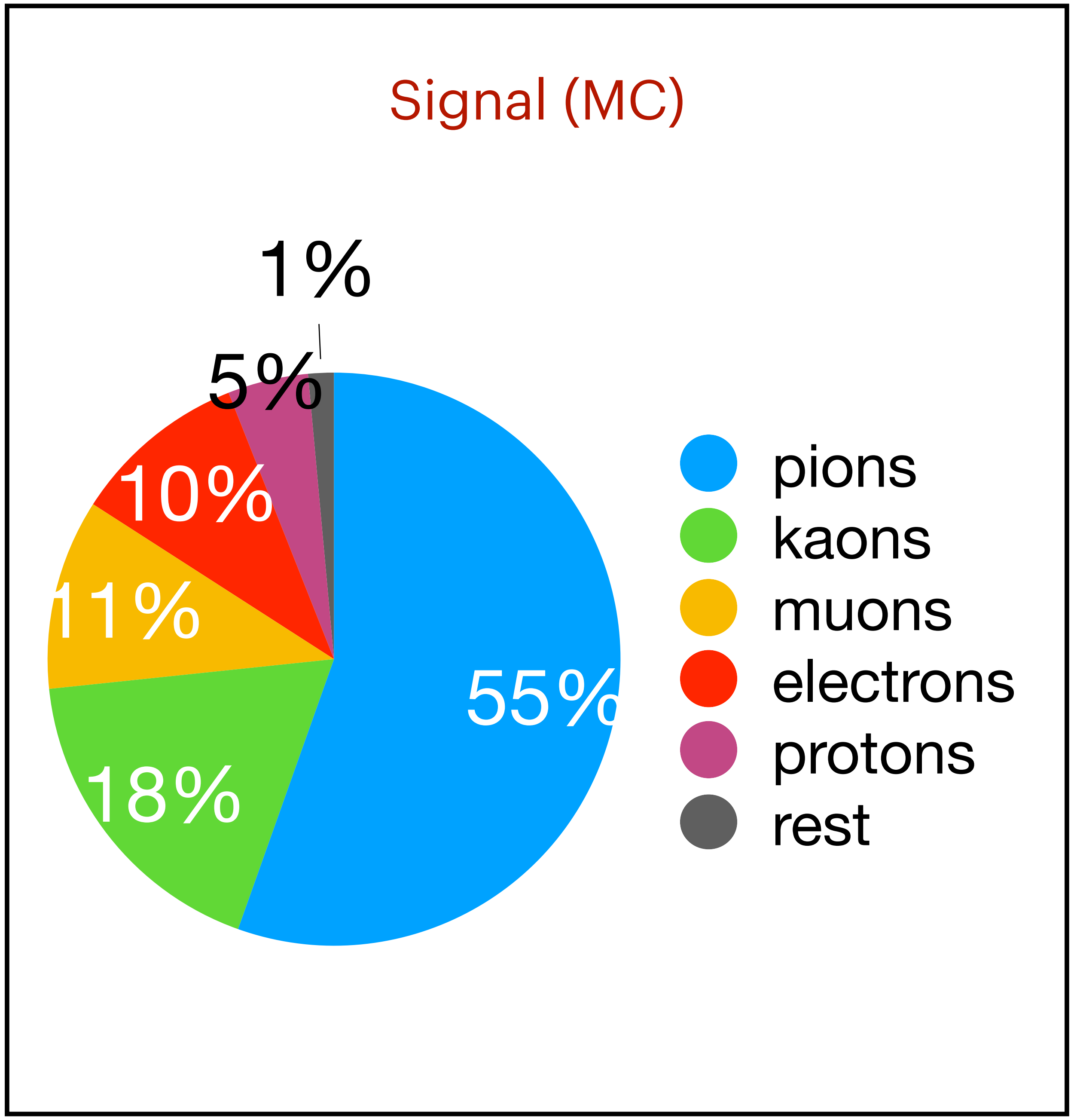
$B^0 \to \pi^0\pi^0$ updates

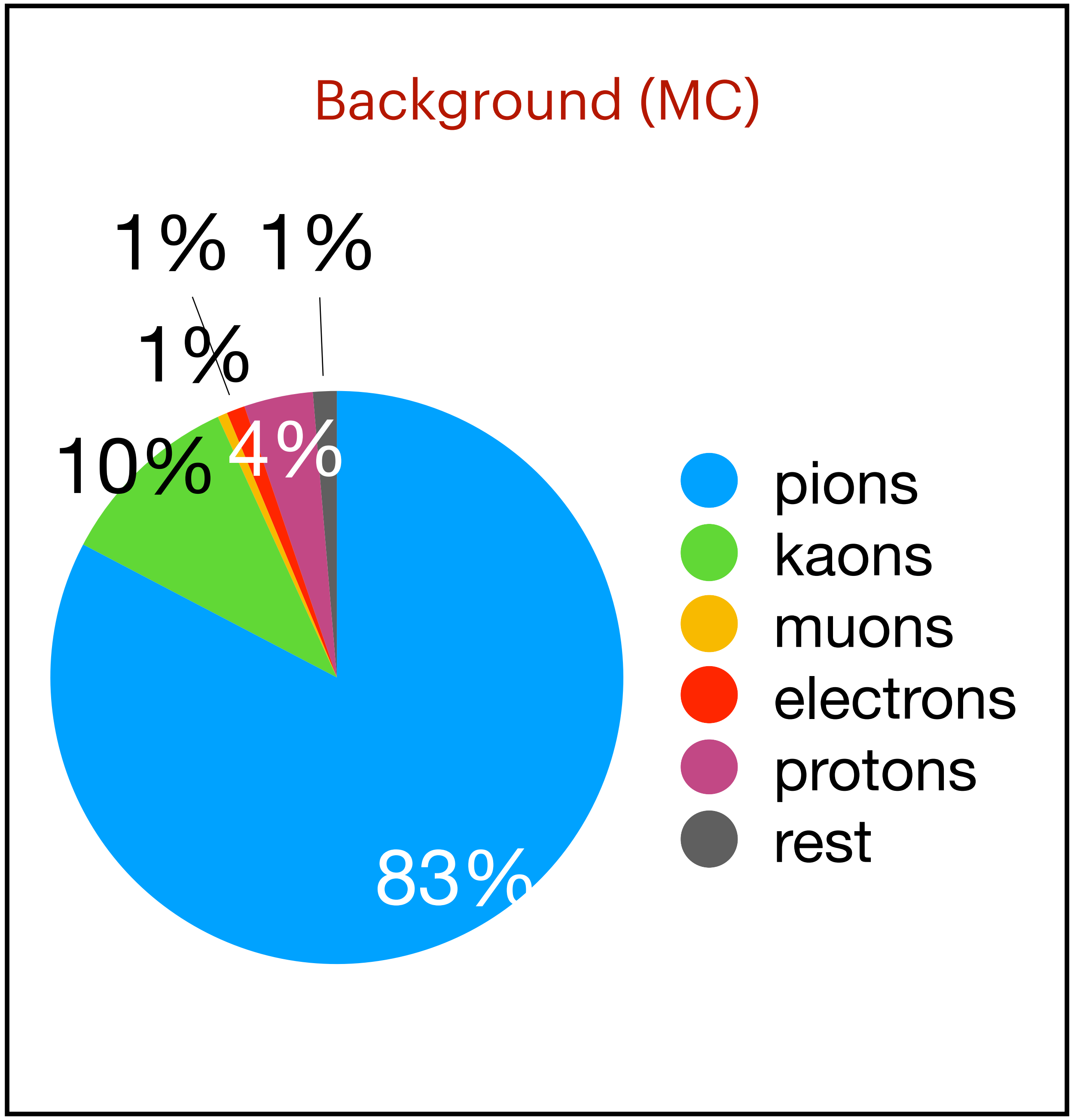
Sebastiano

New possible variables for my CSBDT

PID of ROE tracks

Composition of ROE most energetic track in reconstructed $B^0 \to \pi^0\pi^0$ events:





PID can be a powerful discriminator.

PID of ROE tracks

Use PID as input in CSBDT. But, must correct PID in simulation.

Use syst correction framework to fit PID pdf and correct signalMC PID (for bkg I already use off-res data). I cannot use correction tables (I don't have a specific PID selection).

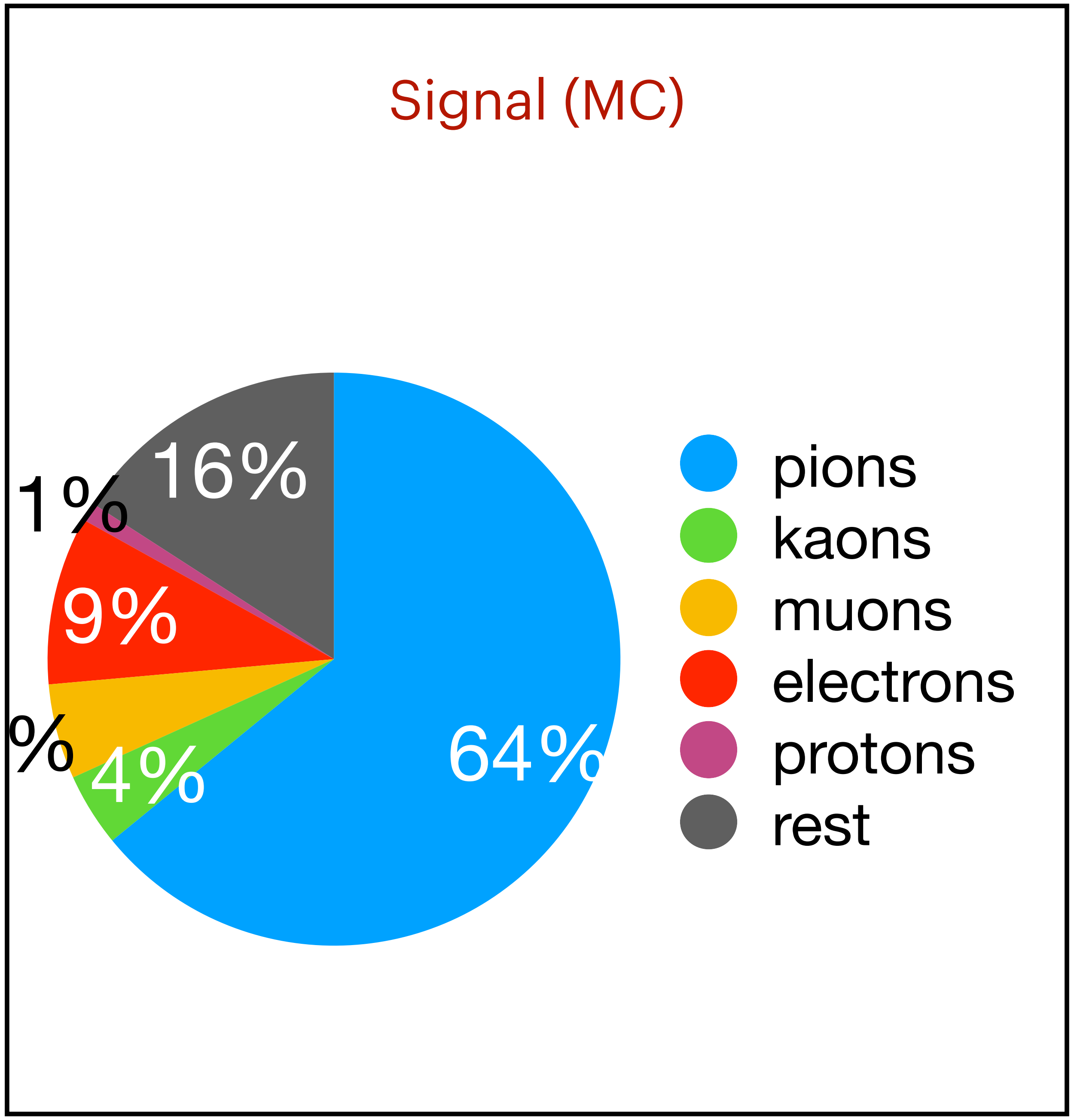
Input variables used in CSBDT:
- pionID
- kaonID
- muonID
- electronID

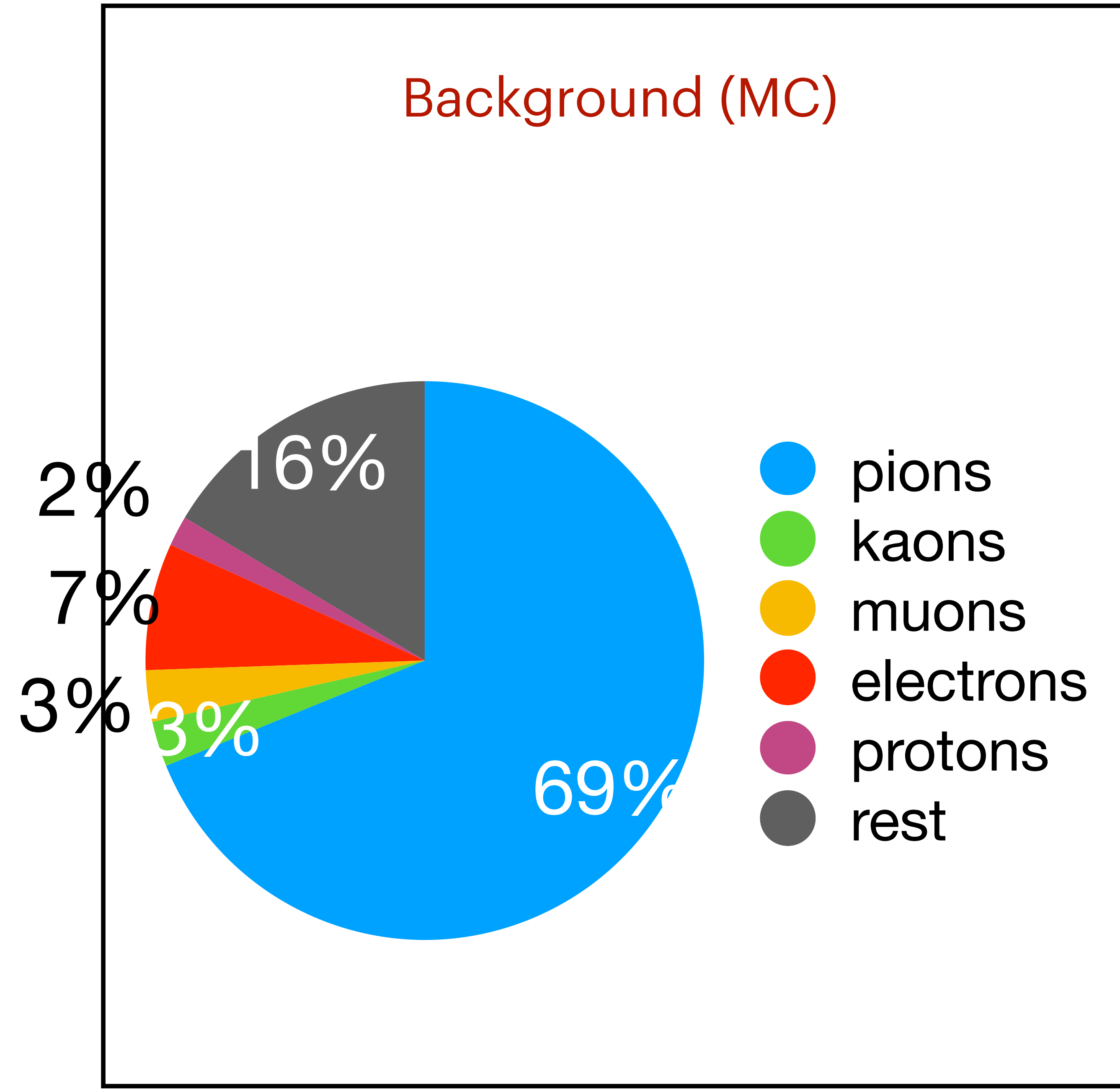
For each particle species, need to correct all the PIDs: 16 combinations!

Lepton corrections not ready in the framework yet.

Less energetic ROE track (soft pion?)

Composition of ROE less energetic track in reconstructed $B^0 \rightarrow \pi^0\pi^0$ events:





PID less powerful, but here momentum, dr, and dz distributions can discriminate.

FEI

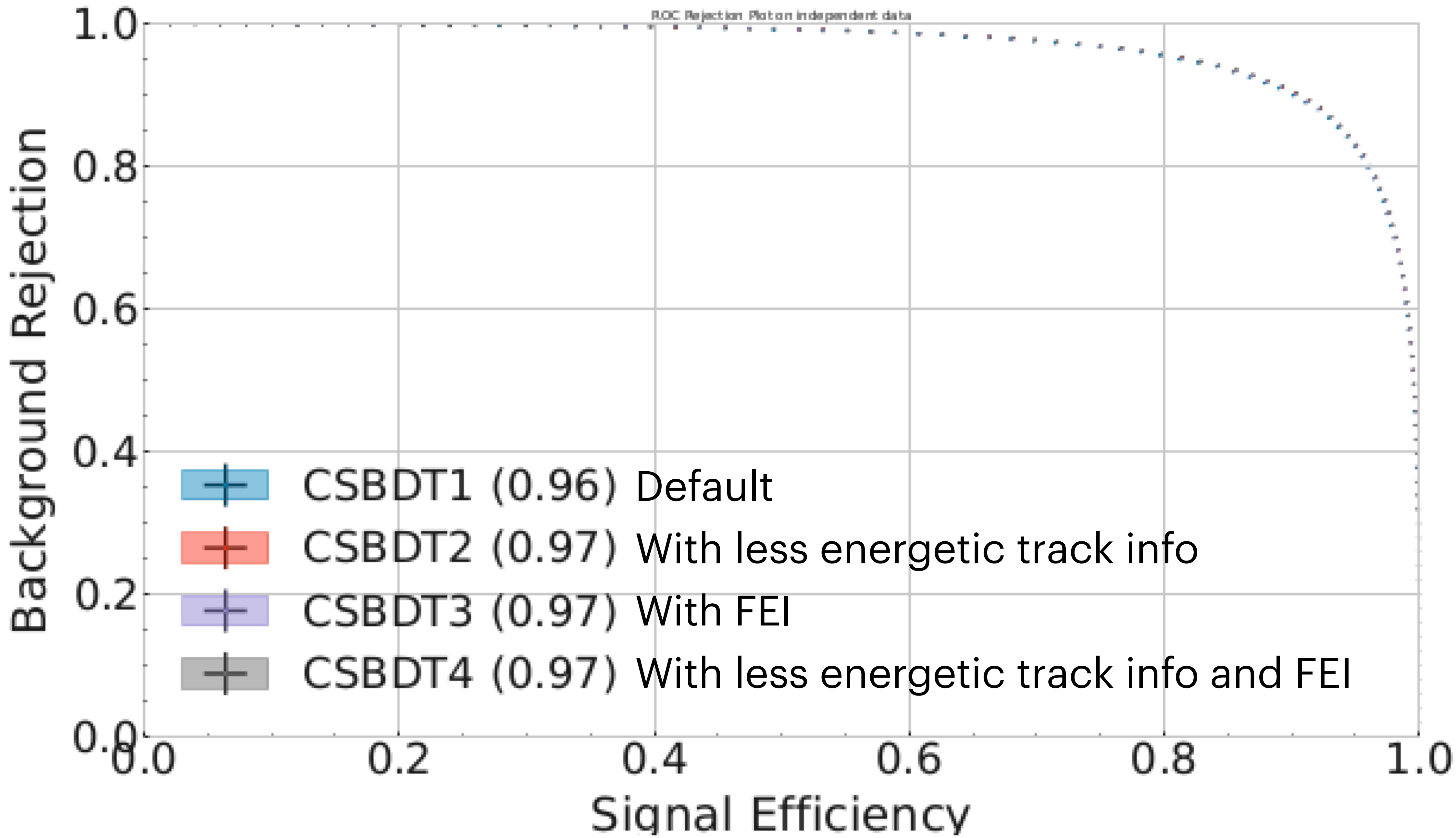
Look at semileptonic FEI of the B^0.

Signal (MC)	Background (MC)
Fraction (SigProb != NaN) = 76.57%	Fraction (SigProb != NaN) = 62.37%
Fraction (SigProb>0.00001) = 68.11%	Fraction (SigProb>0.00001) = 52.15%
Fraction (SigProb>0.0001) = 54.65%	Fraction (SigProb>0.0001) = 36.04%
Fraction (SigProb>0.001) = 30.88%	Fraction (SigProb>0.01) = 18.15%

Cut on NaN values could already help.
Apply selection, or use as input in BDT.

ROC comparison

Use all new variables as inputs in CSBDT. NaN values are considered having no info at all.

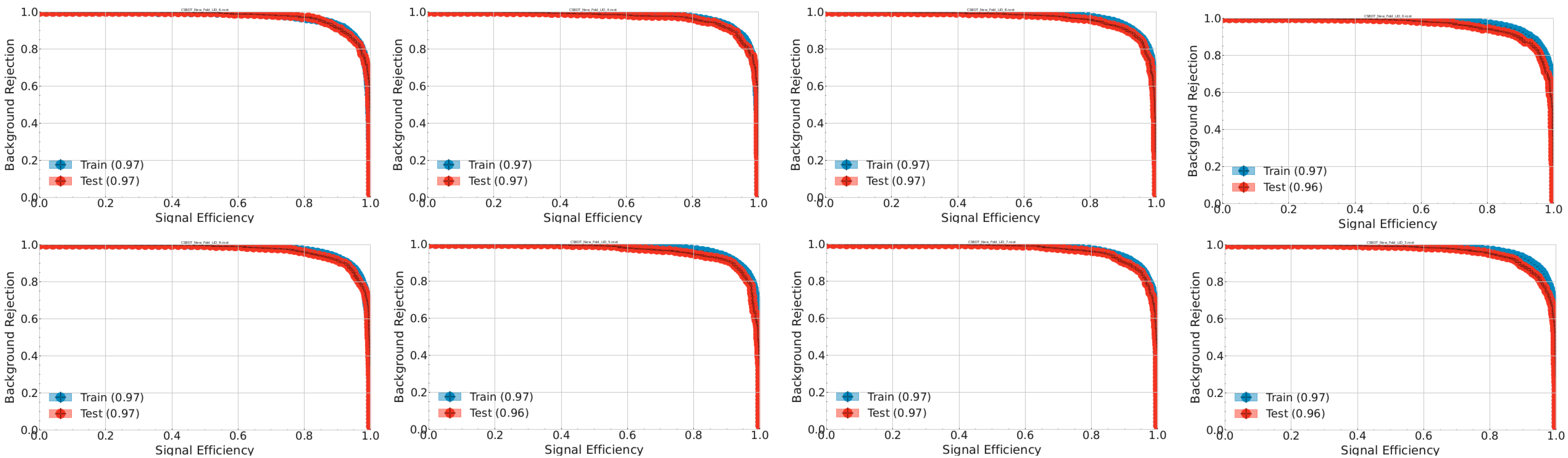


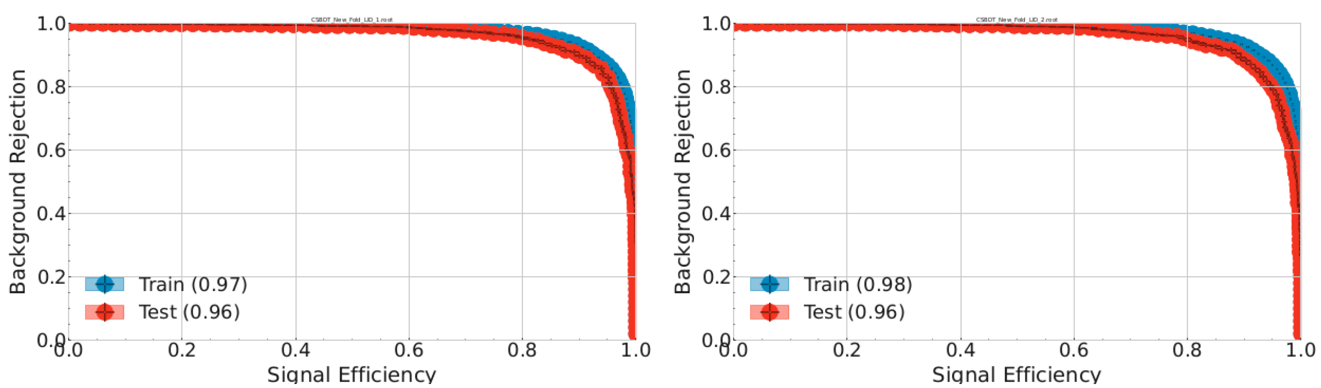
No improvement.

Am I using the NaN info of the FEI?
Is a simple cut better?

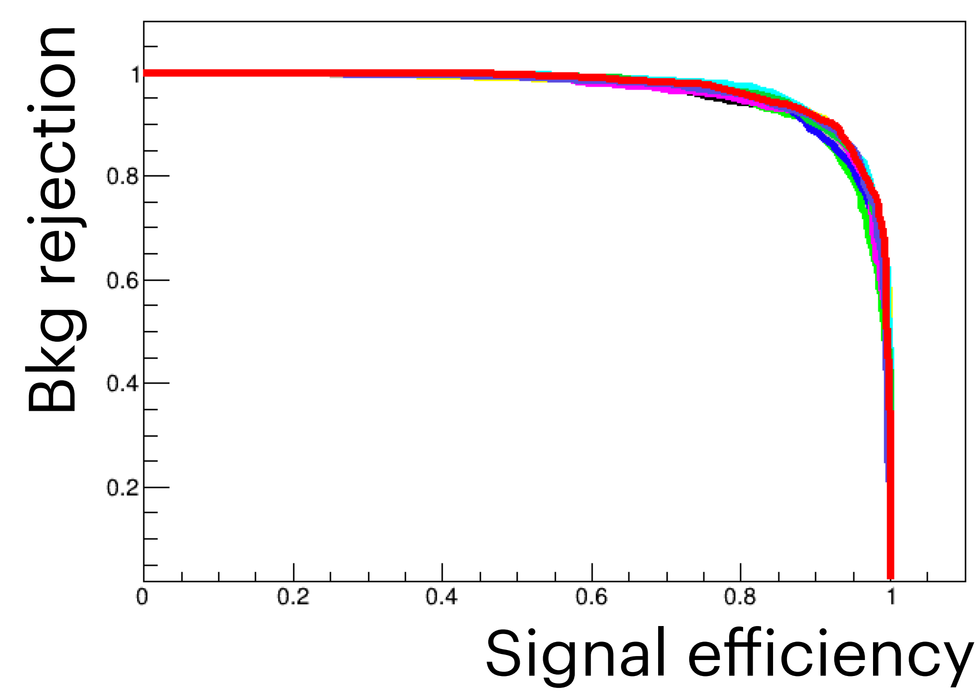
Back to the old CSBDT

Go back to default CSBDT, and use all off-res data (possible final configuration).
Perform k-fold cross evaluation.





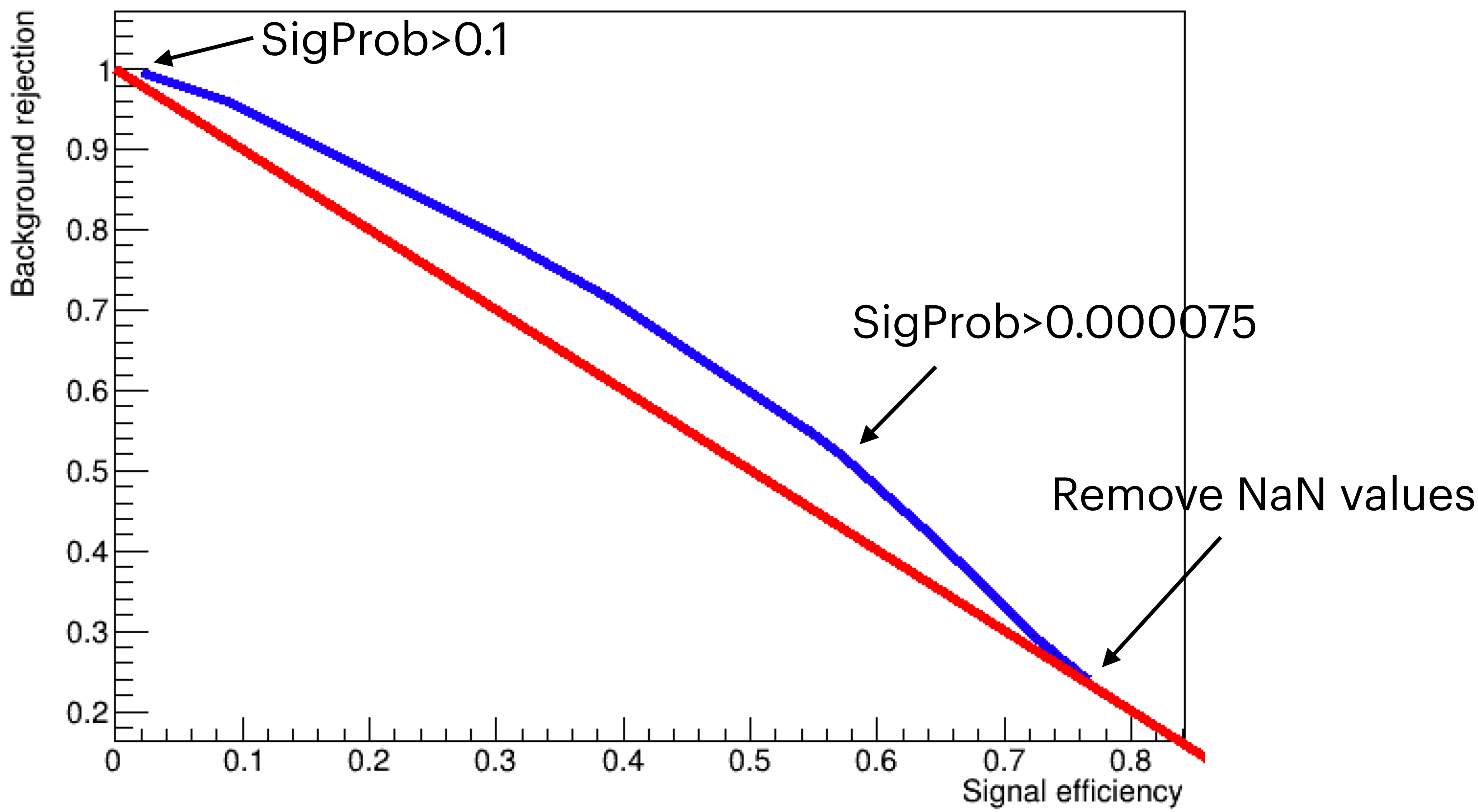
No bias observed. BDT is stable.
Use all data for the training.



ROC comparison

Remove less energetic track and FEI info from CSBDT.
After k-fold cross evaluation, obtain "final" CSBTD using all off-res data.
After a selection on CSBDT > 0.7, obtain ROC curve for FEI selection.

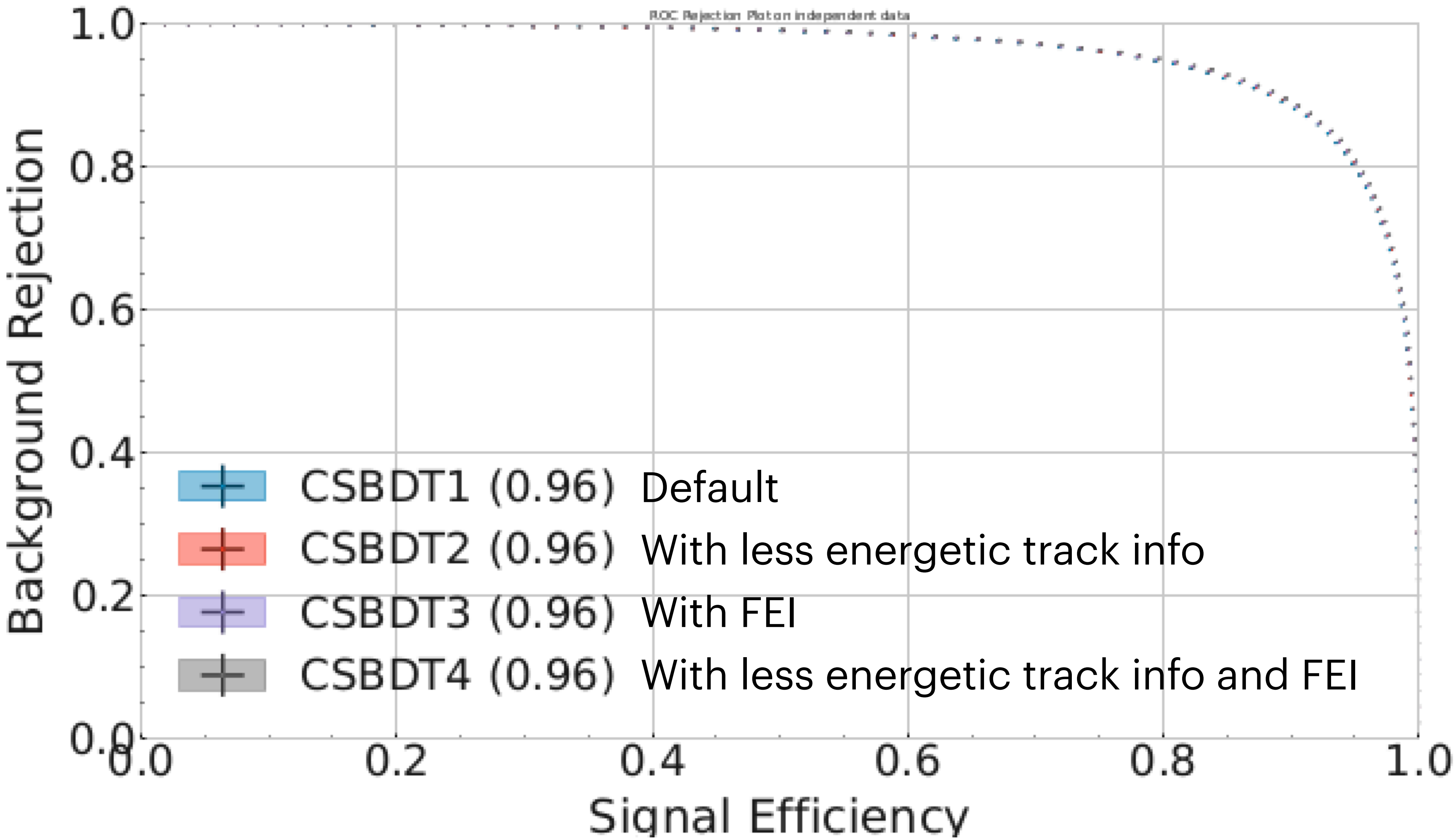
NB: "wrong" extreme values are due to NaNs.



We lose 43% of the signal and 52% of the background.

ROC comparison (w/o FEI NaN values)

Let's instead remove first all the FEI NaN values, and repeat the ROC comparison.



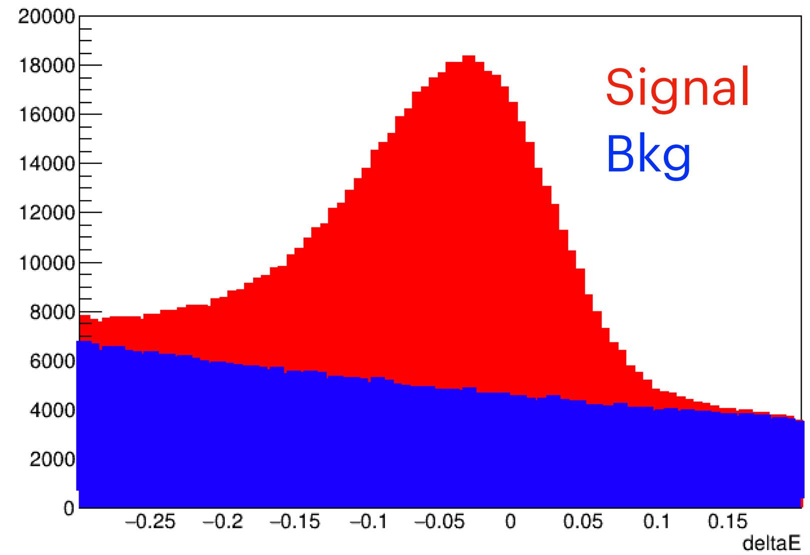
No improvement (even lower AUC).
Probably there's some correlation btw FEI and other variables.

Do we need the FEI NaN information?

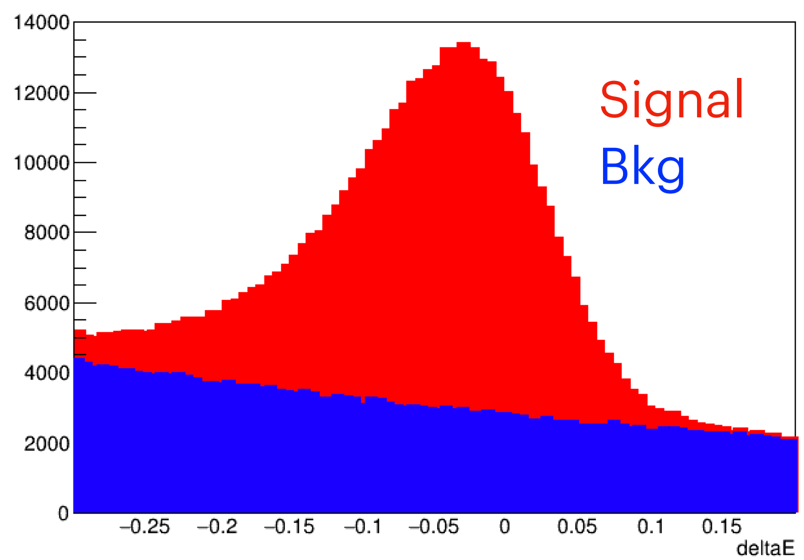
NaN values in FEI

Take sample with signal:bkg = 1:1.

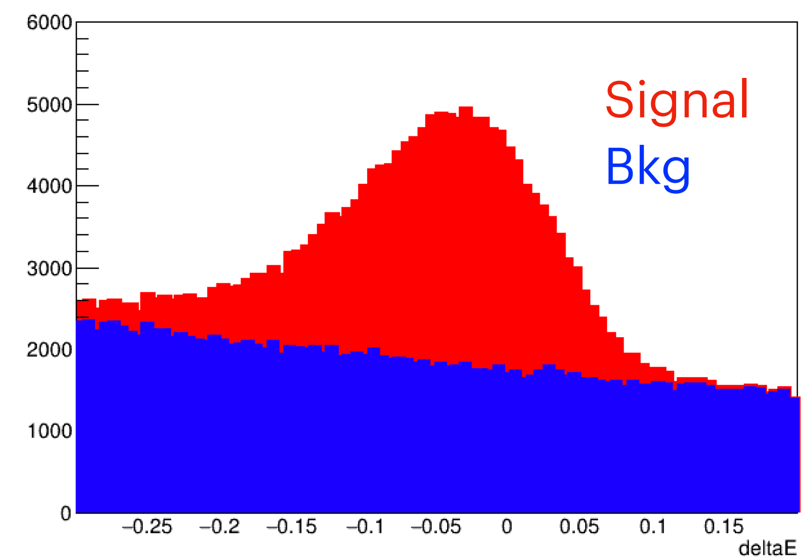
Total



FEI!=NaN



FEI==NaN



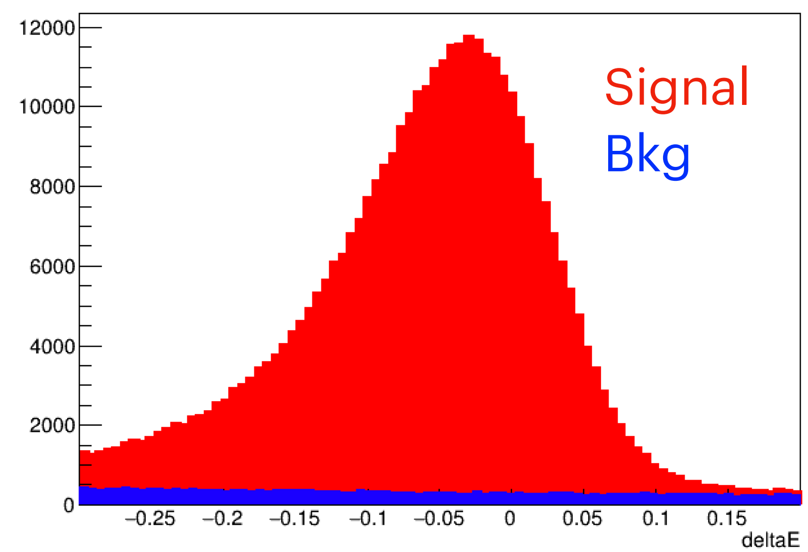
FEI has NaN value more frequently in background wrt signal

→ signal/bkg separation.

NaN values in FEI

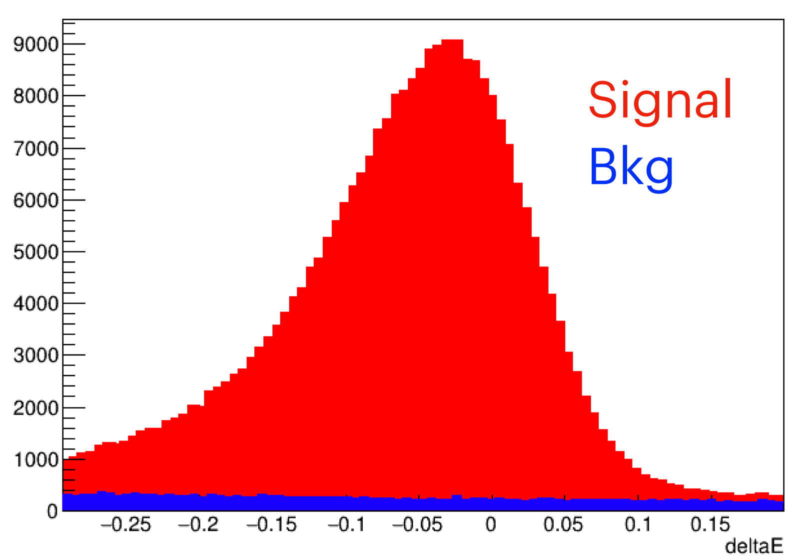
Take sample with signal:bkg = 1:1. Apply selection CSBDT>0.7 (default CSBDT w/o FEI).

Total

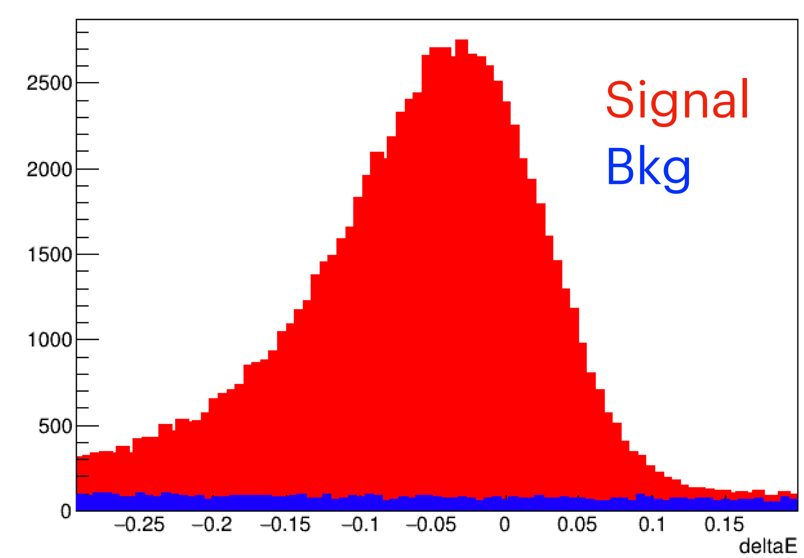


After CSBDT selection

FEI!=NaN



FEI==NaN



No separation.

There’s correlation between FEI and CSBDT.

NaN values in BDT

Low priority

Effect will be very small, not sure if we want to pursue this.

How can I use the NaN information of the FEI in the BDT?

2

FastBDT will ignore the NaN values during training. When finding the best cut these entries are simply not counted for that particular feature. During application, if the tree tries to cut on a feature with a NaN value it will treat the node as the terminal node. See section 4.2: Support for Missing Values here.

answered Dec 27 '1
Marcel Hohmann
1207 •12 •27 •41

updated Dec 27 '1

I asked Marcel if there's a way to circumvent this, still waiting for his reply.

New variables explored bring no large improvement.

Fitter

First steps

Take fitter from Riccardo and Benigno.

Change some small elements to make it work for my case:

- in decoder.cc, add my variables

- in plotter.cc, allow plotting of charge integrated sample with charge=0 ($B^0\bar{B}^0$)

- in generator.cc, modify variables to be generated

- when fitting a toy, remember to add the variable “charge” in the config file (or in the generated sample) even if charge=0 and it’s not used in the fit. Otherwise, fit will fail

- add new functions in functions.cc

- change configFile

Main problems encountered

Fitter is developed for a charged channel, and relies on division btw B^+ and B^-.

Passing to charge integrated fit of B^0 sample was ~easy, but now need to pass from 2 charge bins to 7 qr bins. Not trivial. Many parts need to be changed.

Best would be to obtain an object independent from number of bins (1,2, … , n) using cycles (?) instead of if()…else() for each bin.

Also, now the pdf shapes must be the same in the 2 charge bins. I'll need different shapes for different bins → need to generalise.

Need to modify fitter.cc, plotter.cc, … I'll ask help to Benigno.

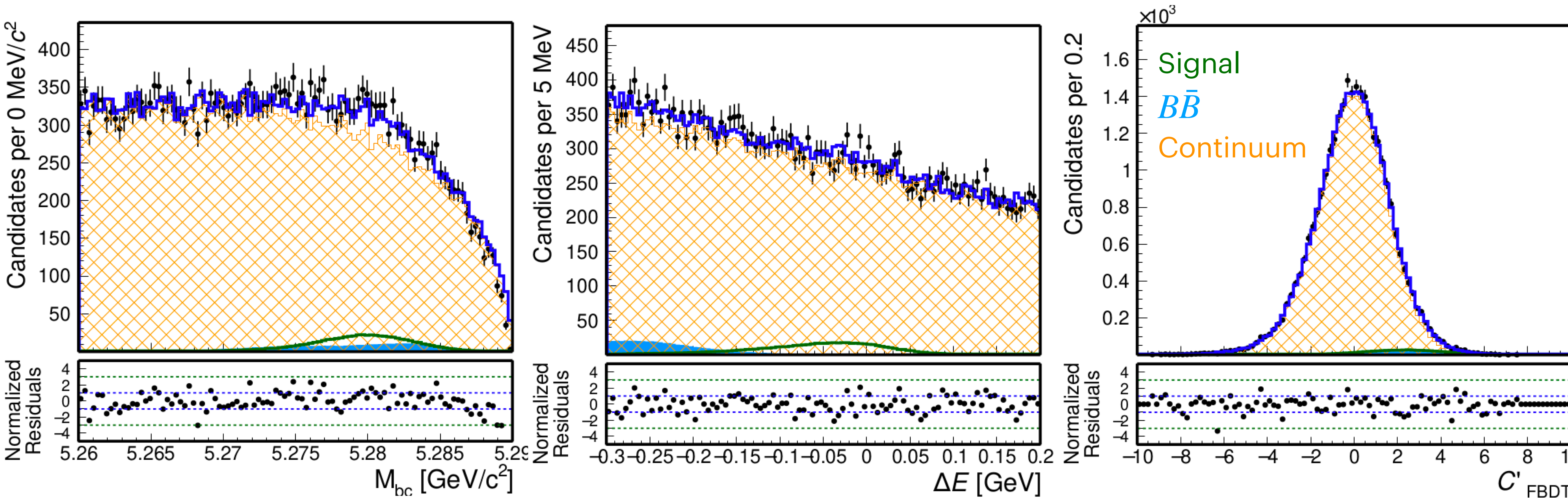
Improvements I need:
- fitter independent from number of charge (or wrong tag, or whatever) bins
- shapes can be different from bin to bin

First results

Non extended ML fit.

Fitted signalMC, fitted $B\bar{B}$ background from MC.

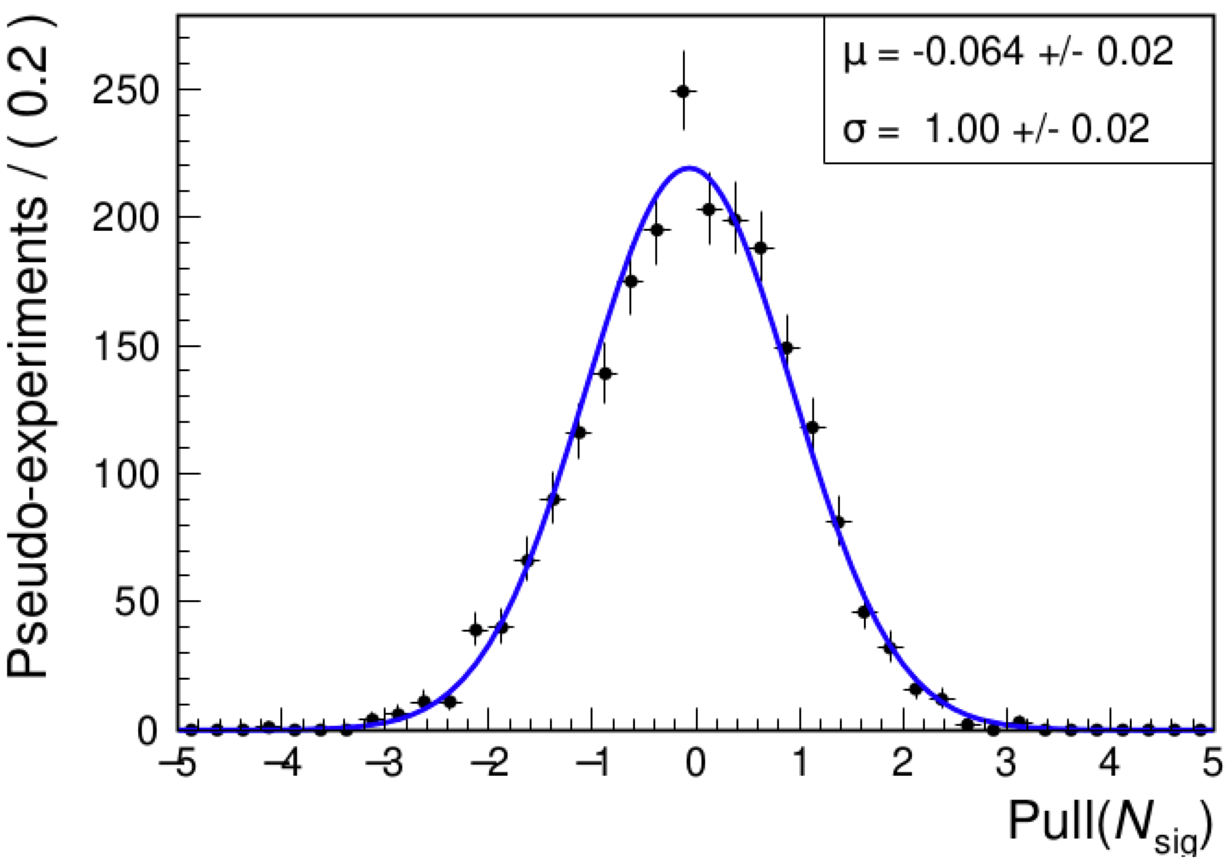
Fix signal and $B\bar{B}$ parameters, and fit realistic $B^0 \to \pi^0\pi^0$ sample of 1ab^{-1} (qr-integrated).

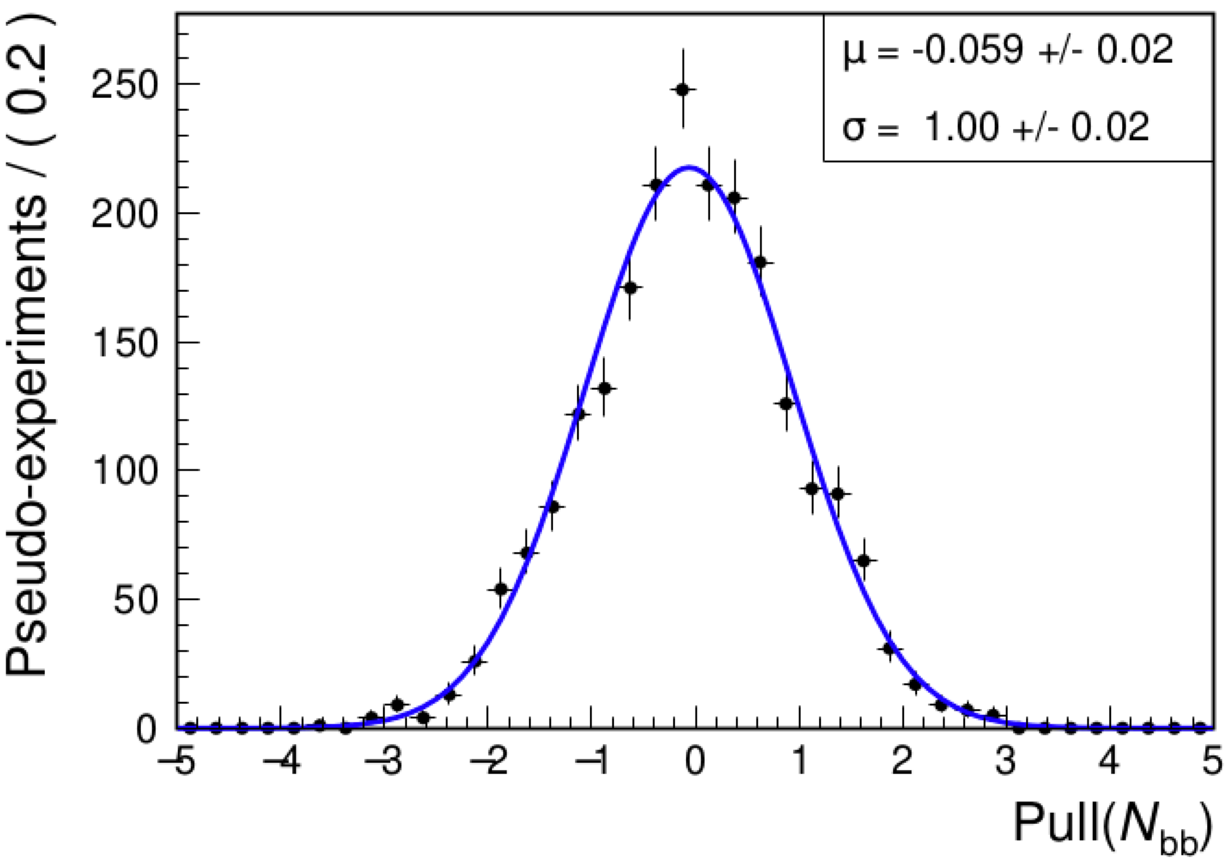


Looks good.

Pulls

Float values of signal yield and $B\bar{B}$ yield (1 ab^{-1}) multinomially (total number of events is fixed). Repeat 2000 times.

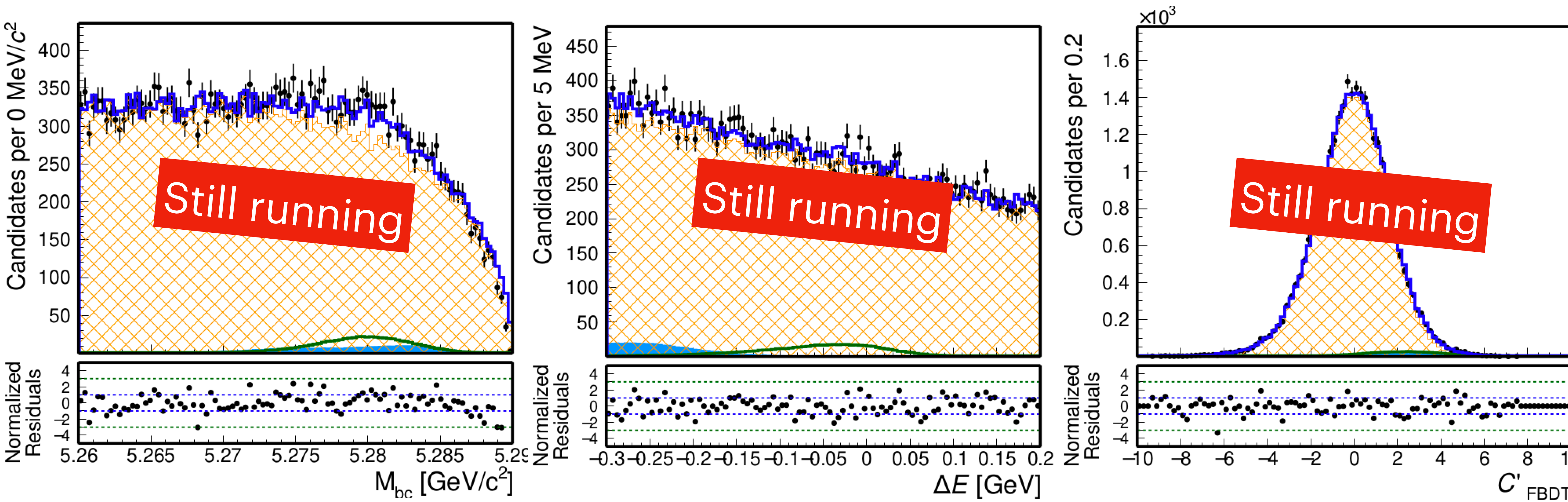




Small under-evaluation of yields.

First results

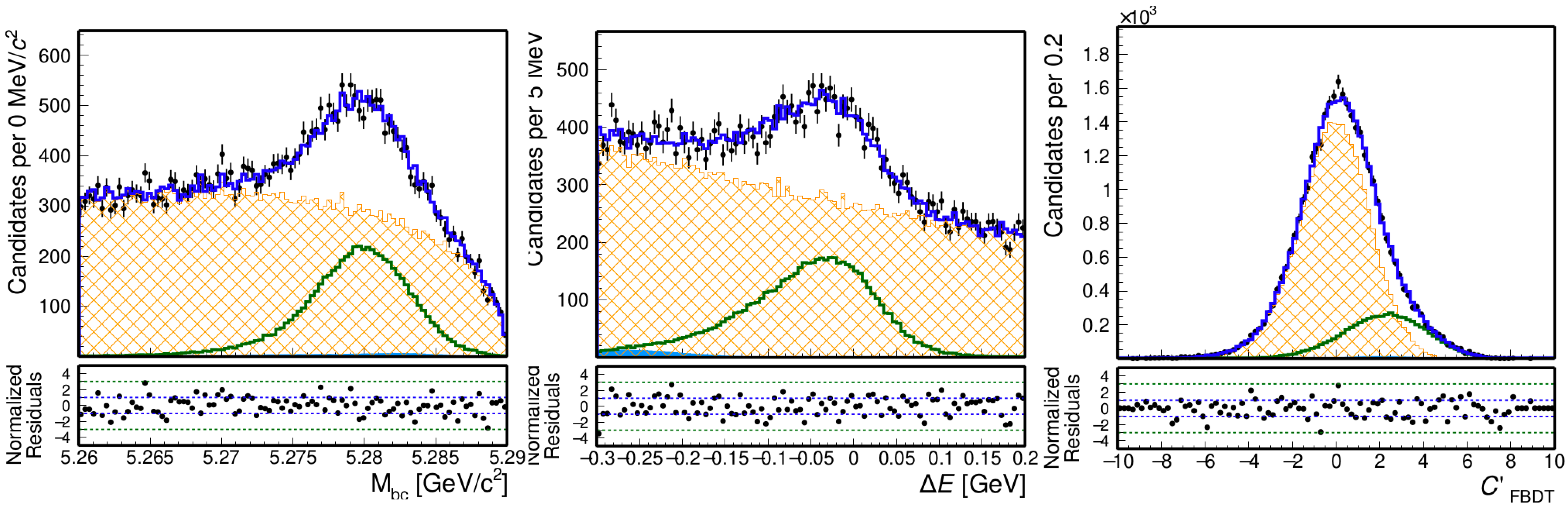
Check on pdfs: generate high-statistics sample (100ab^{-1}) and check results.



Still running

First results

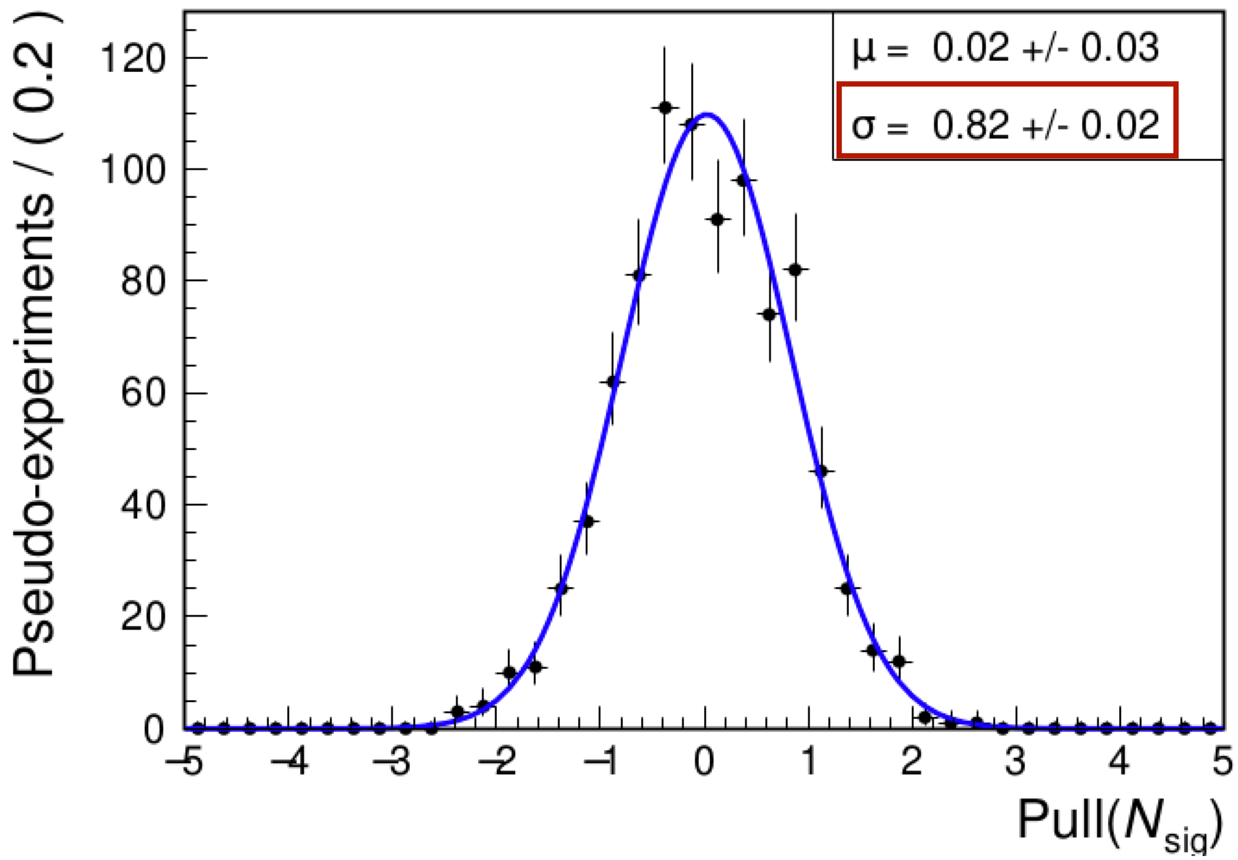
Check: generate sample with 10x signal and check results.

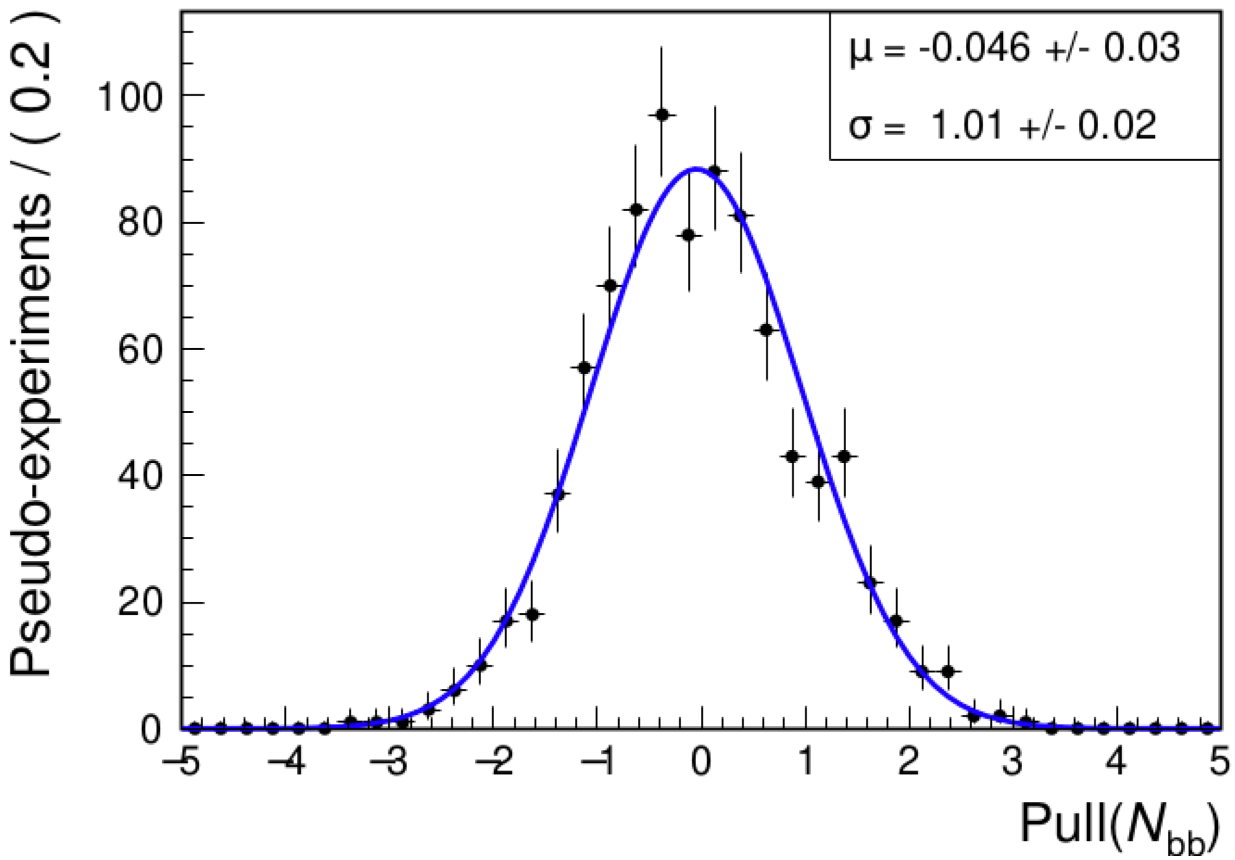


Looks good.

Pulls

Float values of signal yield and $B\bar{B}$ yield (1 ab^{-1}) multinomially (total number of events is fixed). Repeat 1000 times.

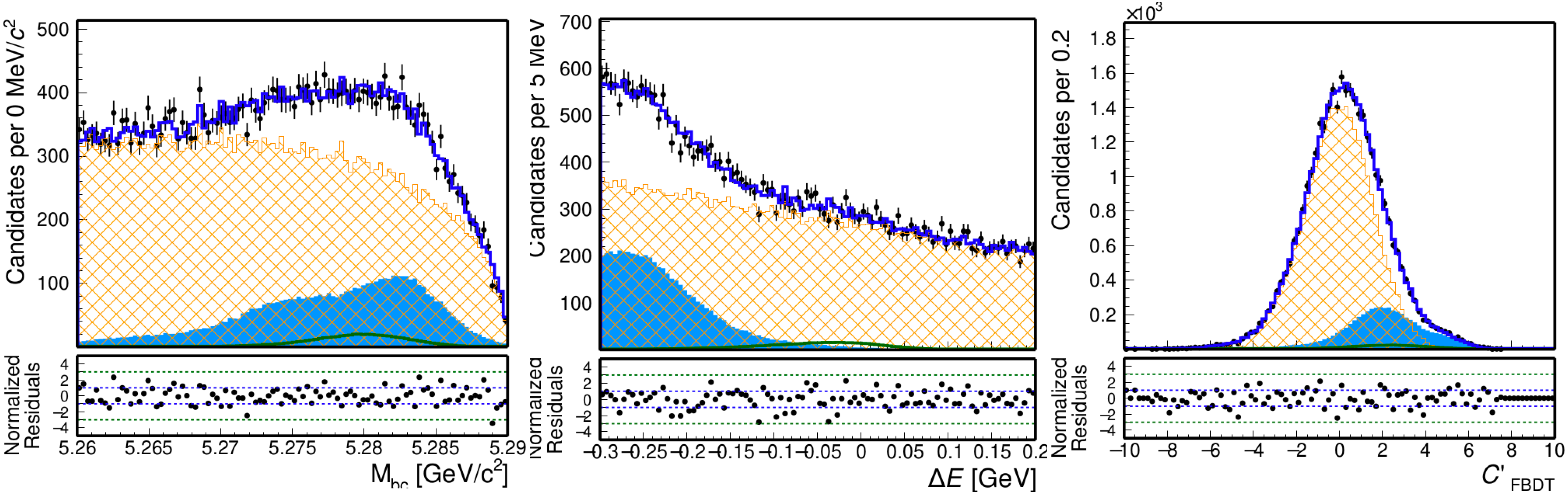




Overestimation of signal uncertainty.

First results

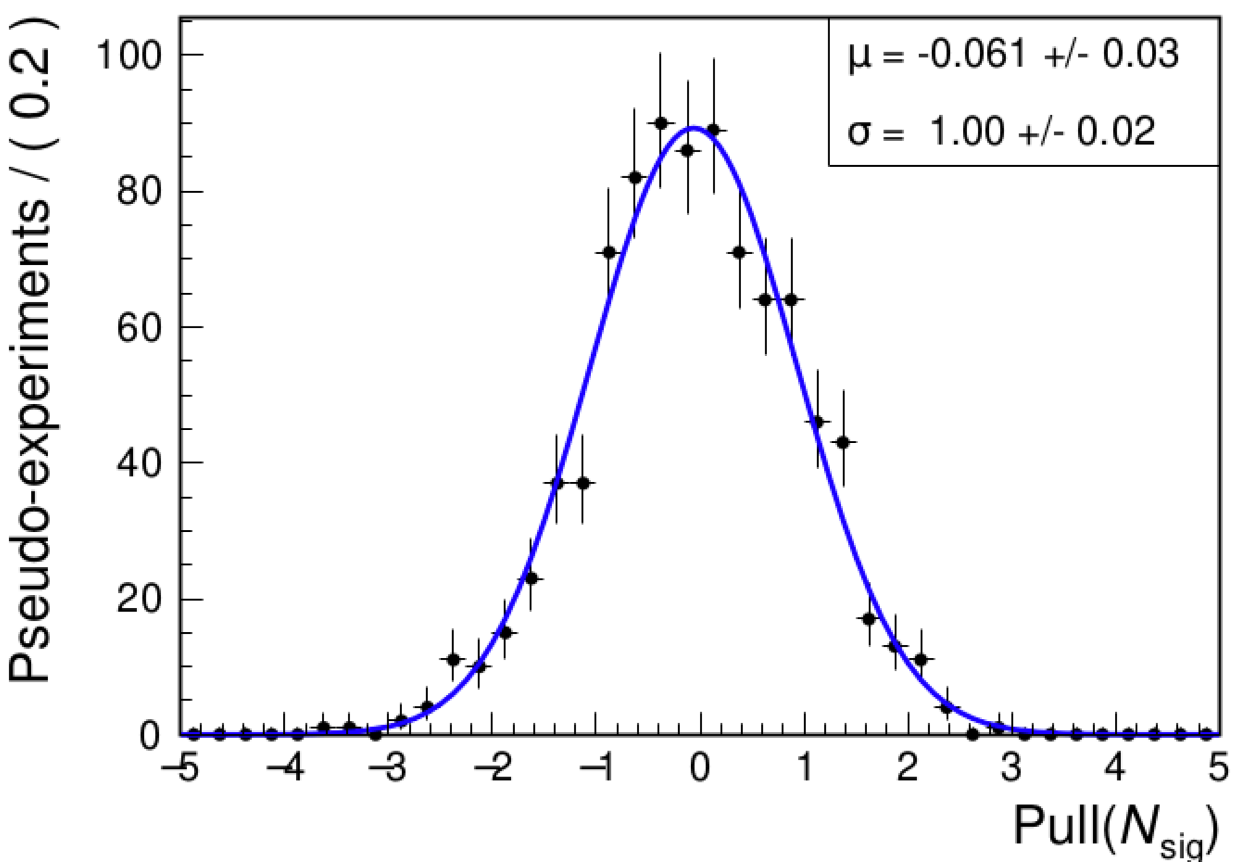
Check: generate sample with 10x $B\bar{B}$ and check results.

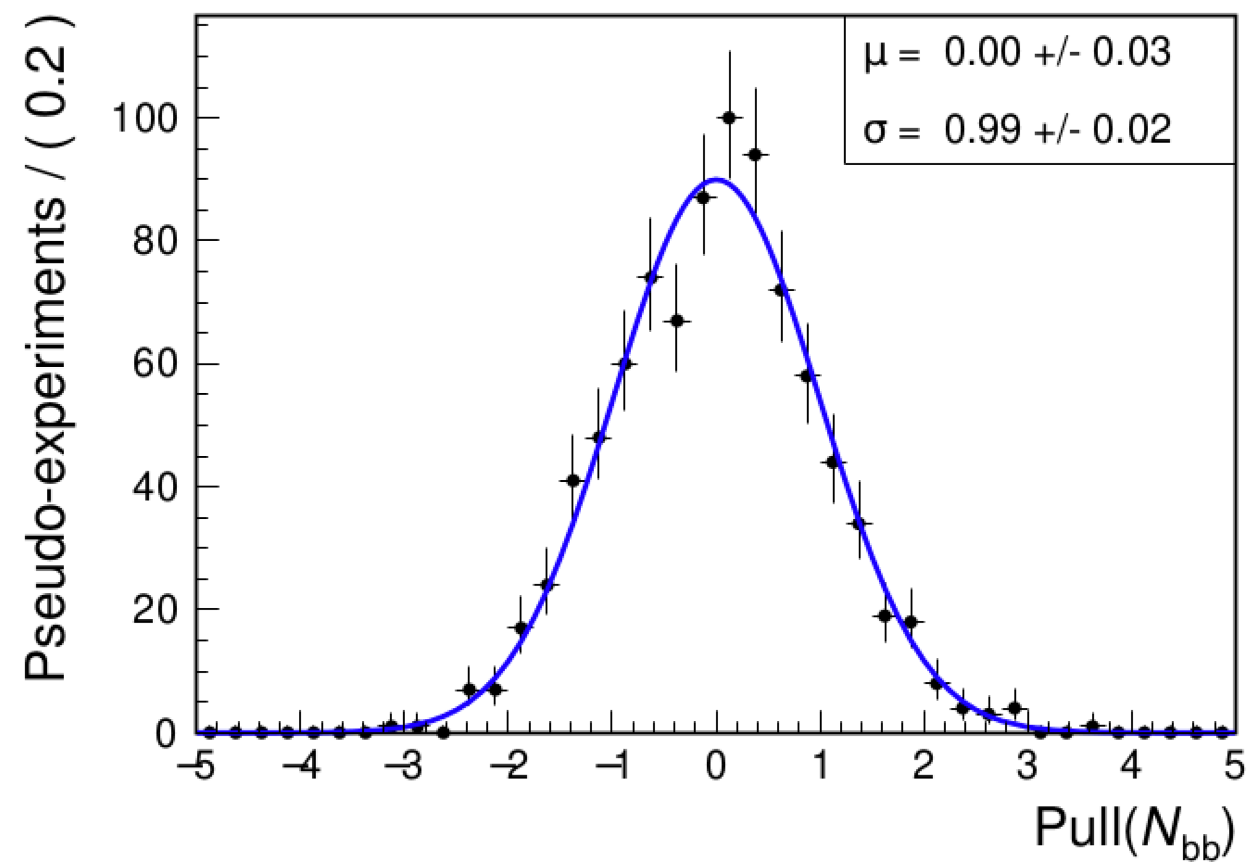


Looks good.

Pulls

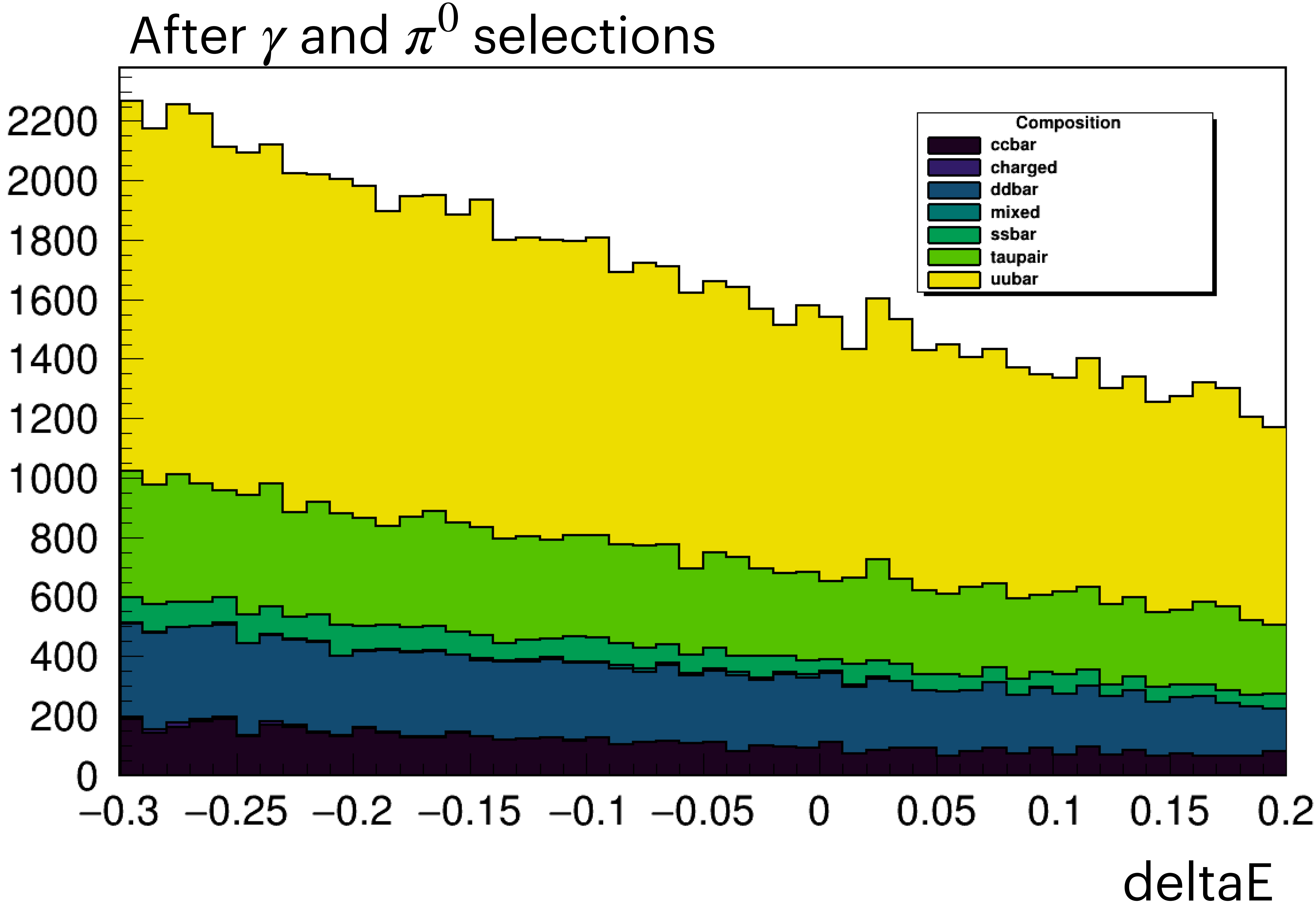
Float values of signal yield and $B\bar{B}$ yield (1 ab^{-1}) multinomially (total number of events is fixed). Repeat 1000 times.





Looks good.

Next step (after completing validation): pass to 7 bins of qr, or directly include w as variable in fit.



After γ and π^0 selections
Composition
ccbar
charged
ddbar
mixed
ssbar
taupair
uubar
0
200
400
600
800
1000
1200
1400
1600
1800
2000
2200
−0.3
−0.25
−0.2
−0.15
−0.1
−0.05
0
0.05
0.1
0.15
0.2
deltaE

FEI

Look at hadronic FEI of the B^0.

Signal (MC)

Fraction (SigProb != NaN) = 10.9%

Fraction (SigProb>0.00001) = 8.4%

Background (MC)

Fraction (SigProb != NaN) = 17.1%

Fraction (SigProb>0.00001) = 12.99%

No