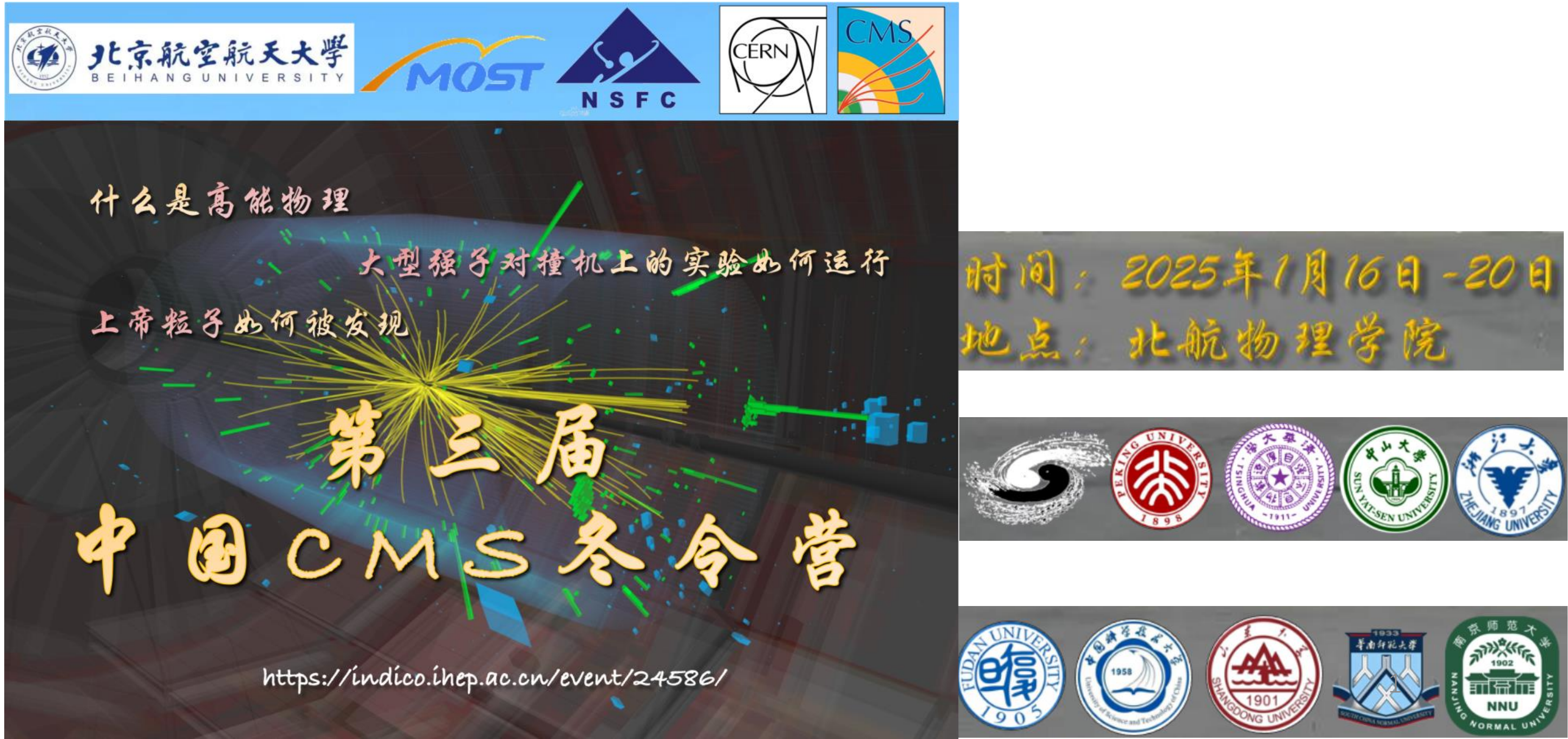
Electrons and Photons

Junquan Tao (IHEP/CAS, Beijing)

陶军全(中科院高能所)

CMS

Compact Muon Solenoid



中国科学院高能物理研究所

Institute of High Energy Physics

Chinese Academy of Sciences

Welcome to e/γ + Hands-on

- ➢ **Physics motivations**
- ➢ **e/γ reconstruction and identification**
- ➢ **Exercises**

Goal

- Learn basics of e/γ reconstruction and ID
  - ✓ Variables used in ID/discrimination
- Efficiency studies with $Z \rightarrow ee$ TnP
- Make comparison and performance plots

Some useful links

CMS Offline WorkBook : https://twiki.cern.ch/twiki/bin/view/CMSPublic/WorkBook

CMS DAS: https://lpc.fnal.gov/programs/schools-workshops/cmsdas.shtml

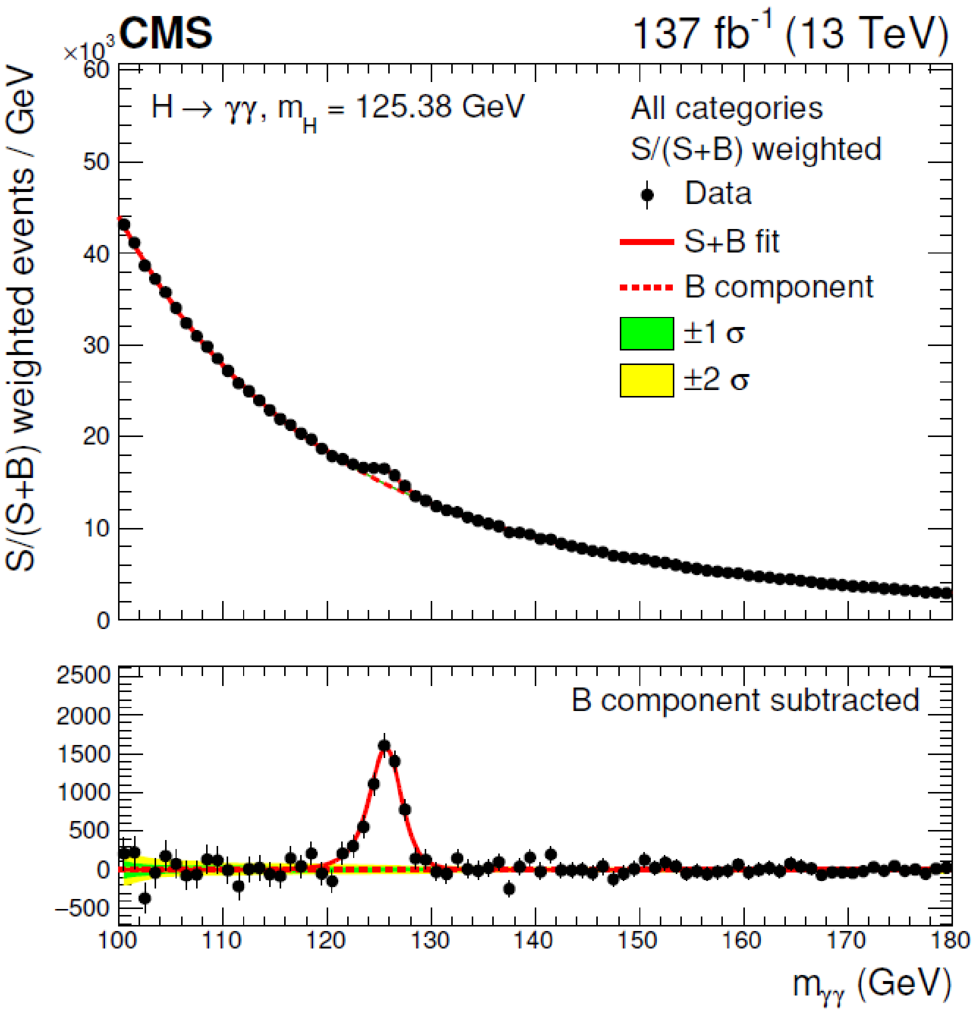
2025 CMSDAS

- LPC, *IN PERSON at FERMILAB* (January 13-17, 2025) - Indico Agenda

E/gamma POG twiki : https://twiki.cern.ch/twiki/bin/view/CMS/EgammaPOG

Importance of e and γ in CMS

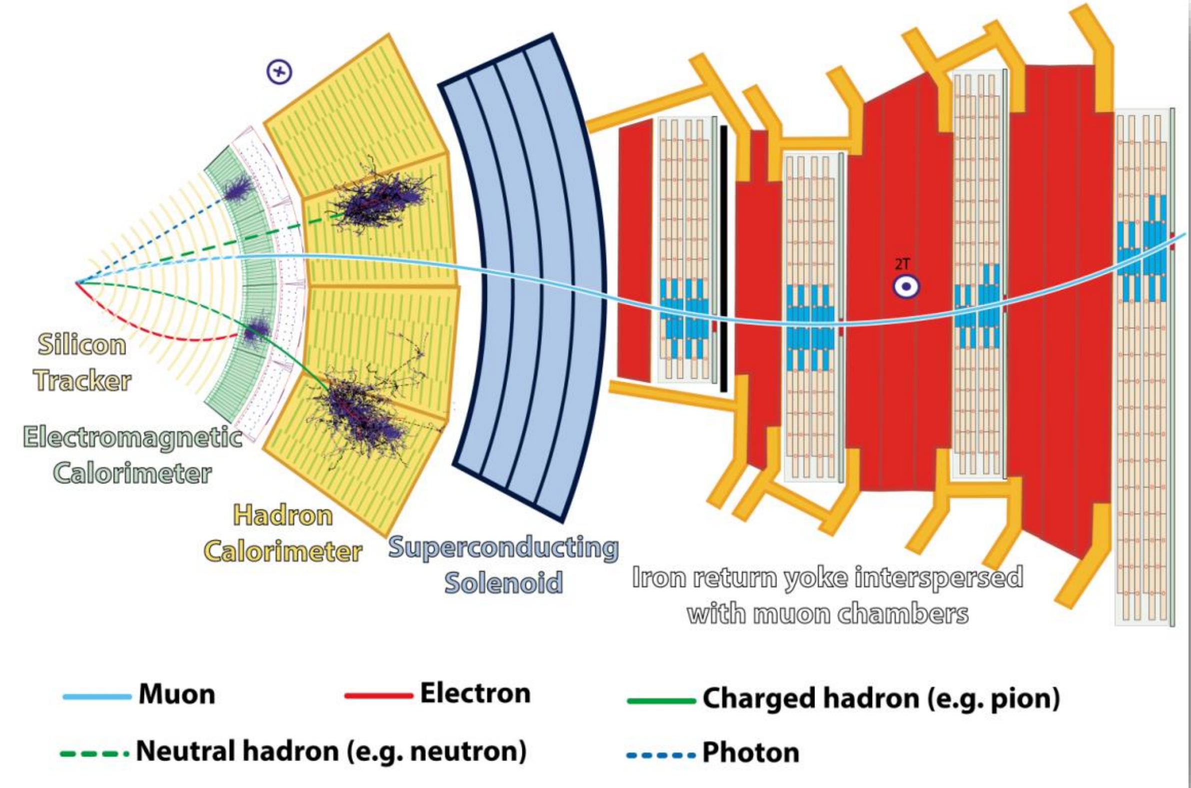
- **Standard Model (SM) physics**
  - ✓ $H \rightarrow \gamma\gamma$
  - ✓ $W \rightarrow e\nu$
  - ✓ $Z \rightarrow ee$
    - Standard candle of particle physics
    - Detector calibration
- **Beyond the Standard Model (BSM) physics**
  - ✓ **Z' and W'** decay directly the SM leptons (i.e. electrons)
    - **Any BSM with W/Z in final state**
  - ✓ Photons important to BSM searches: Dark Matter and SUSY etc.
    - Low-mass or high-mass $X \rightarrow \gamma\gamma$
    - Anything with a Higgs in the final state : $X \rightarrow HH/HY$, with $H/Y \rightarrow \gamma\gamma$
- **Trigger**
  - ✓ Electron/Photons used to trigger events like those listed above
  - ✓ Their standard model production rate is significantly lower than hadrons produced from QCD

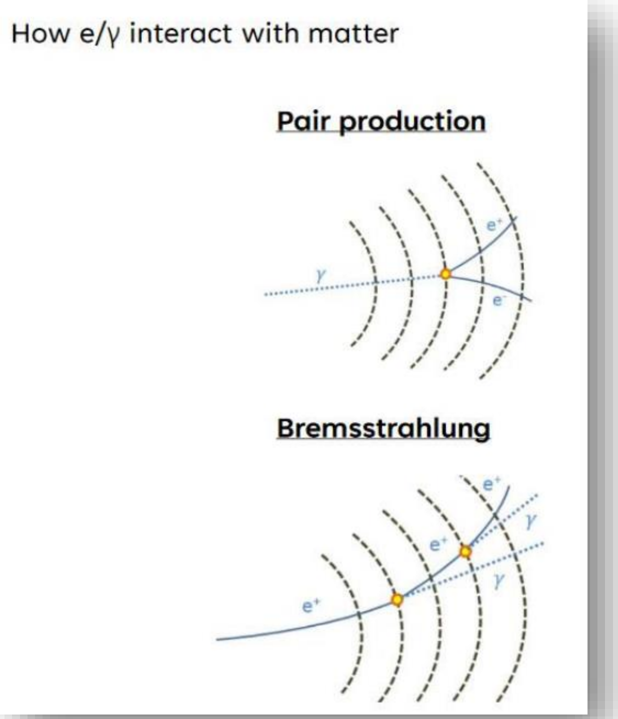


JHEP07(2021)027

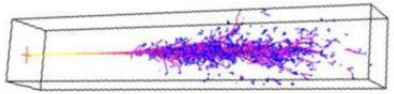
e and γ interactions in CMS

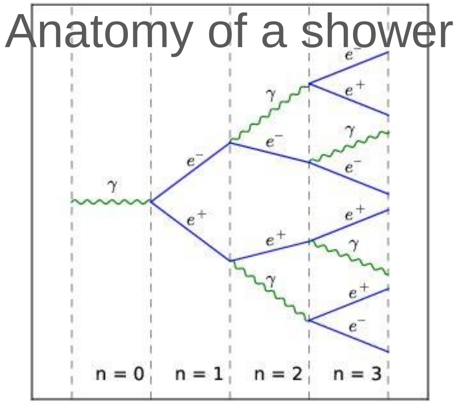
Both create electromagnetic showers in electromagnetic calorimeter (ECAL)



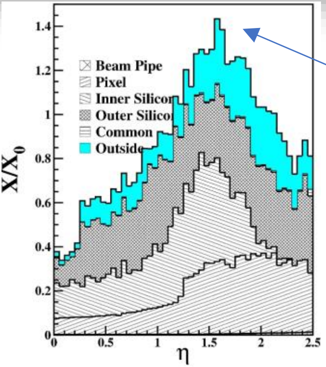


These two processes form the basis of the "electromagnetic shower"





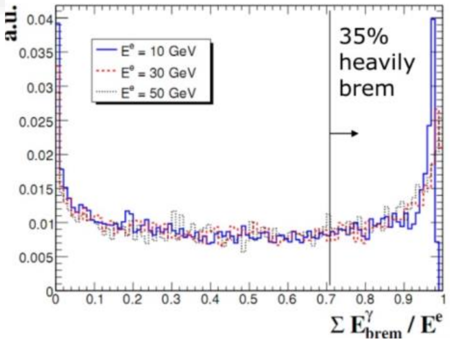
- ✓ **e**: Tracker + **ECAL**
- ✓ **γ : ECAL + Tracker** for converted photons + **HCAL** for fake rejection

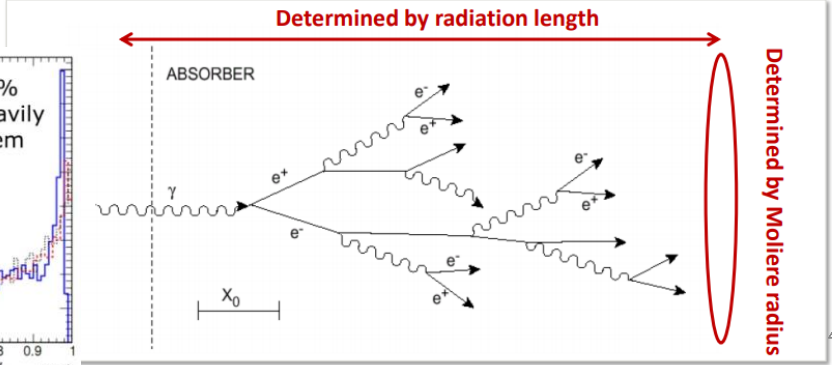


Conversion as high as ~70%

Material before ECAL radiation length X0

$$P_{conv} = 1 - e^{-\frac{7x}{9x_0}}$$



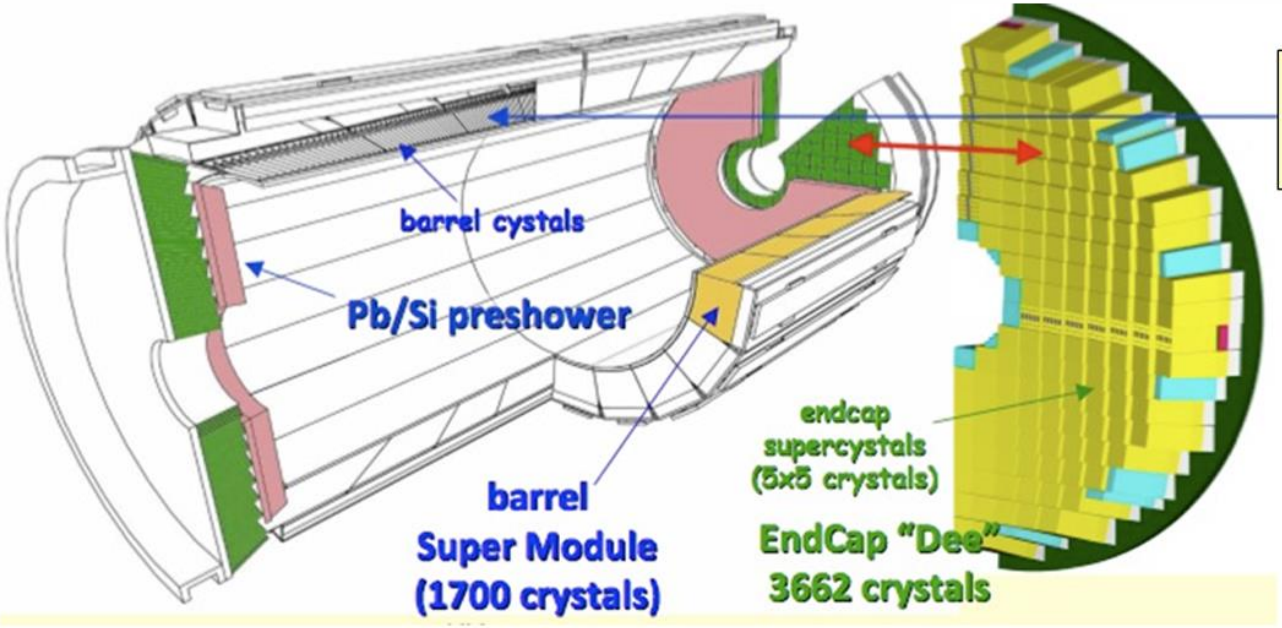


90% (95%) of its energy in 1 (2) M_R

Longitudinal extension ~ln(E_0/**Ec**)

Lateral extension : Moliere radius (M_R)

CMS ECAL

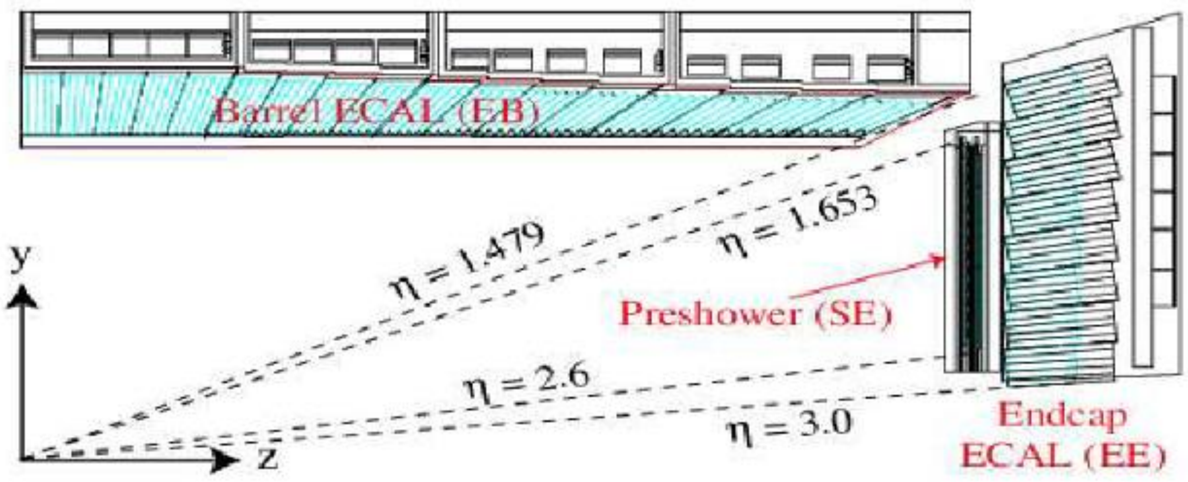


Tapered crystals to provide off-pointing of ~ 3° from vertex

Lead Tungstate ($PbWO_4$) scintillator

- 80% of light released in 25ns
- Has a density of 8.28 g/cm^3
- **A radiation length of 0.89cm** and a **Moliére radius of 2.2cm**
- Total weight of the ECAL is 88.7 tonnes

Two crystal producers: BTCP (Russia) and SIC (China)

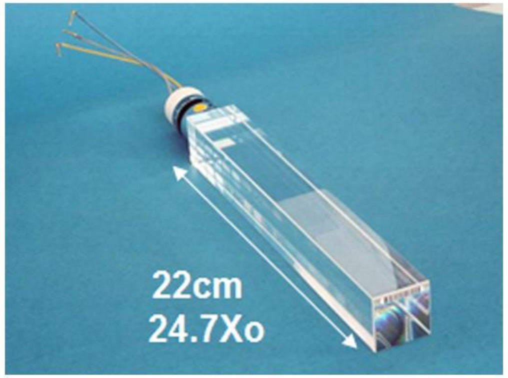


Barrel:
36 Supermodules
(18 per half barrel);
61200 crystals;
Total crystal mass 67.4t;
Avalanche PhotoDiode readout;
coverage: $|\eta| < 1.48$, $\sim 26X_0$.

Barrel crystal, tapered
34 types, $\sim 2.6x2.6\ cm^2$ at rear

Endcaps:
4 Dees (2 per endcap);
14648 crystals;
Total crystal mass 22.9t;
Vacuum PhotoTriode readout;
coverage: $1.48 < |\eta| < 3$, $\sim 25X_0$.

Endcap Preshower:
Pb ($2X_0$, $1X_0$)/Si;
4 Dees (2 per endcap);
4300 Si strips;
1.8 mm x 63 mm;
coverage: $1.65 < |\eta| < 2.6$.

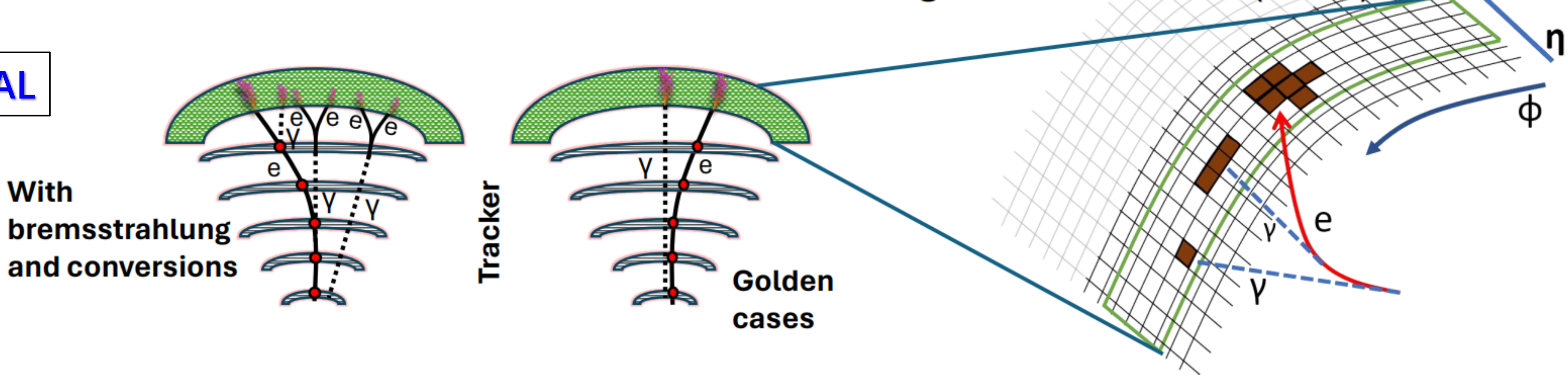


Endcap crystal, tapered
1 type, $3x3\ cm^2$ at rear

e and γ reconstruction

At CMS, **Reconstruction** & **Identification** of e **&** γ is done primarily using information from silicon tracker & electromagnetic calorimeter (ECAL)

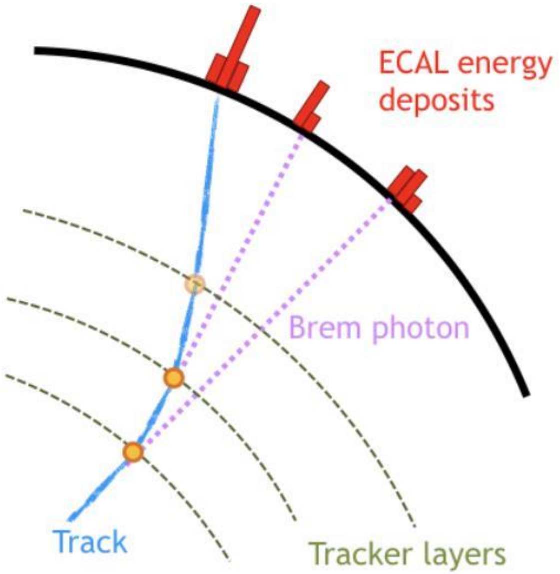
Tracker + ECAL

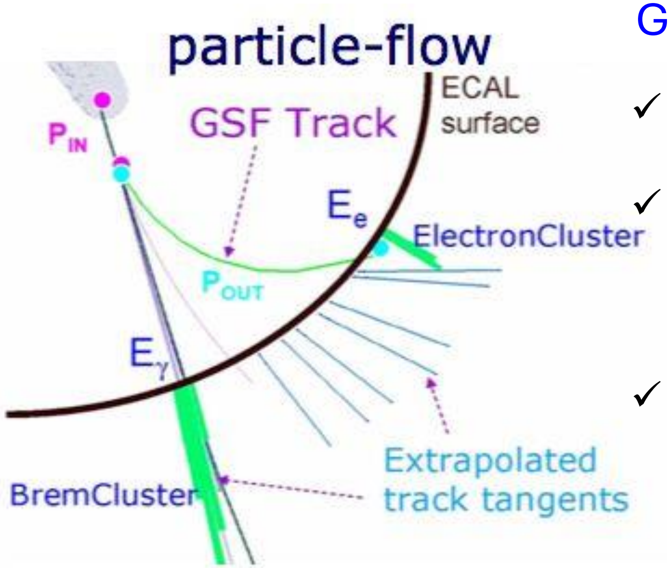


What does an electron look like?

Electron emits bremsstrahlung photons

- ✓ **multiple energy deposits in ECAL :** take into account energy of brems
- ✓ **Electron track can have “kinks”:** special tracking needed for electrons to take care of that





Gaussian Sum Filter (GSF)

- ✓ For all other charged particle tracking, **Kalman Filter** is used
- ✓ Electrons use a dedicated tracking algorithm known as **Gaussian Sum Filter (GSF).** It is a non-linear extension of KF
- ✓ GSF track connected to superclusters via the PF algorithm

ECAL/Energy Reconstruction

- ECAL **RecHit** energy is calculated by a multifit from ECAL Digis

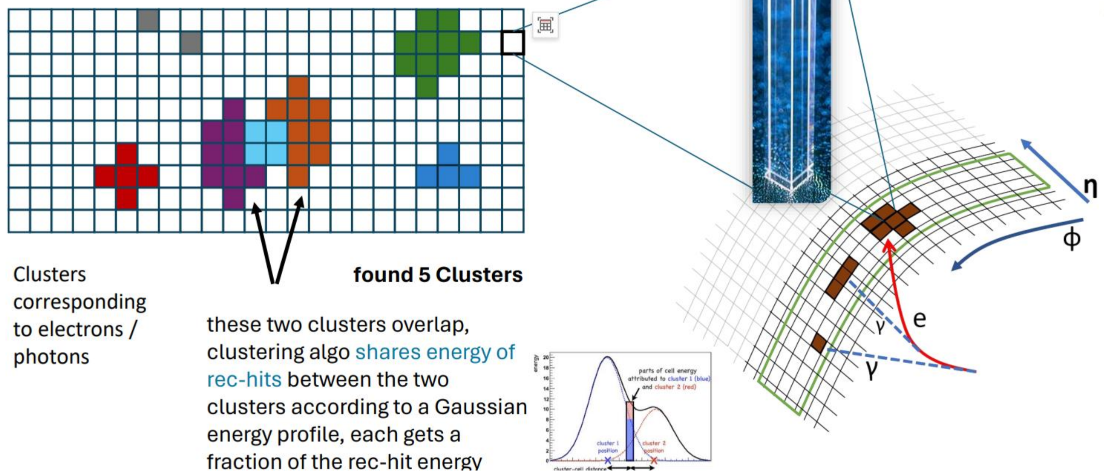
- RecHits are clustered into **Particle Flow (PF) clusters**

- **PF Mustache Algorithm** creates **superclusters**
  - ✓ Groups clusters wide in ϕ and narrow in η
  - ✓ This is to catch bremsstrahlung

- **Refined supercluster** adds soft brem and accounts for photon conversions using tracker info

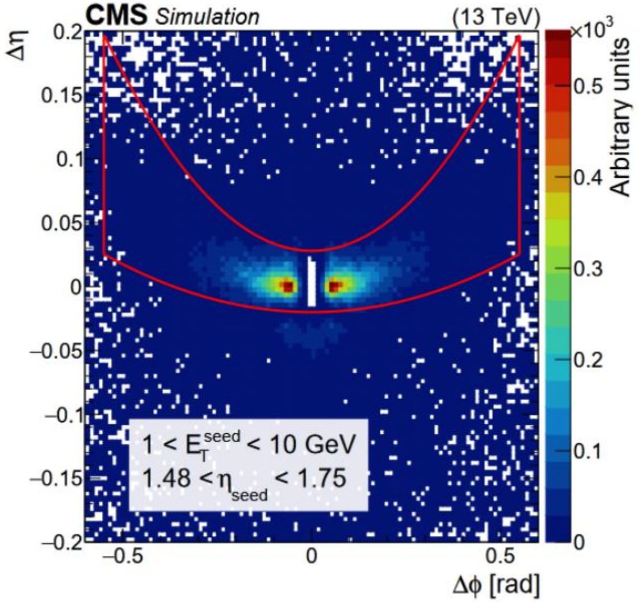
Clustering

- Clustering of ECAL clusters

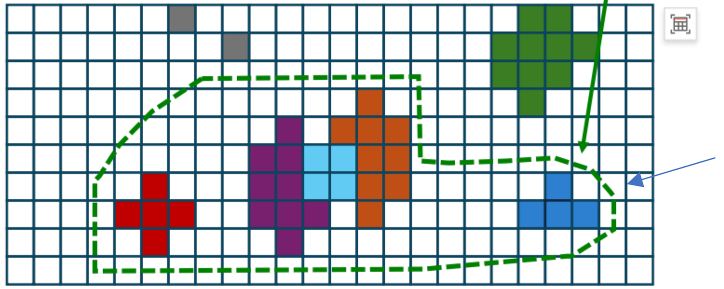


Moustache supercluster

a cluster of clusters

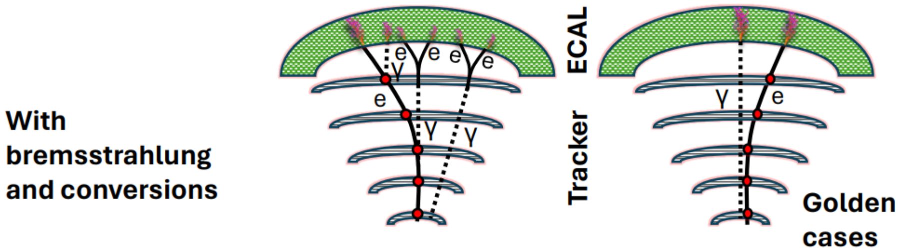


Refined Supercluster



Refined superclusters use the **information from the tracker**, to be able to link bremsstrahlung emissions to missed ECAL deposits

There is also dedicated photon conversion recovery algorithm

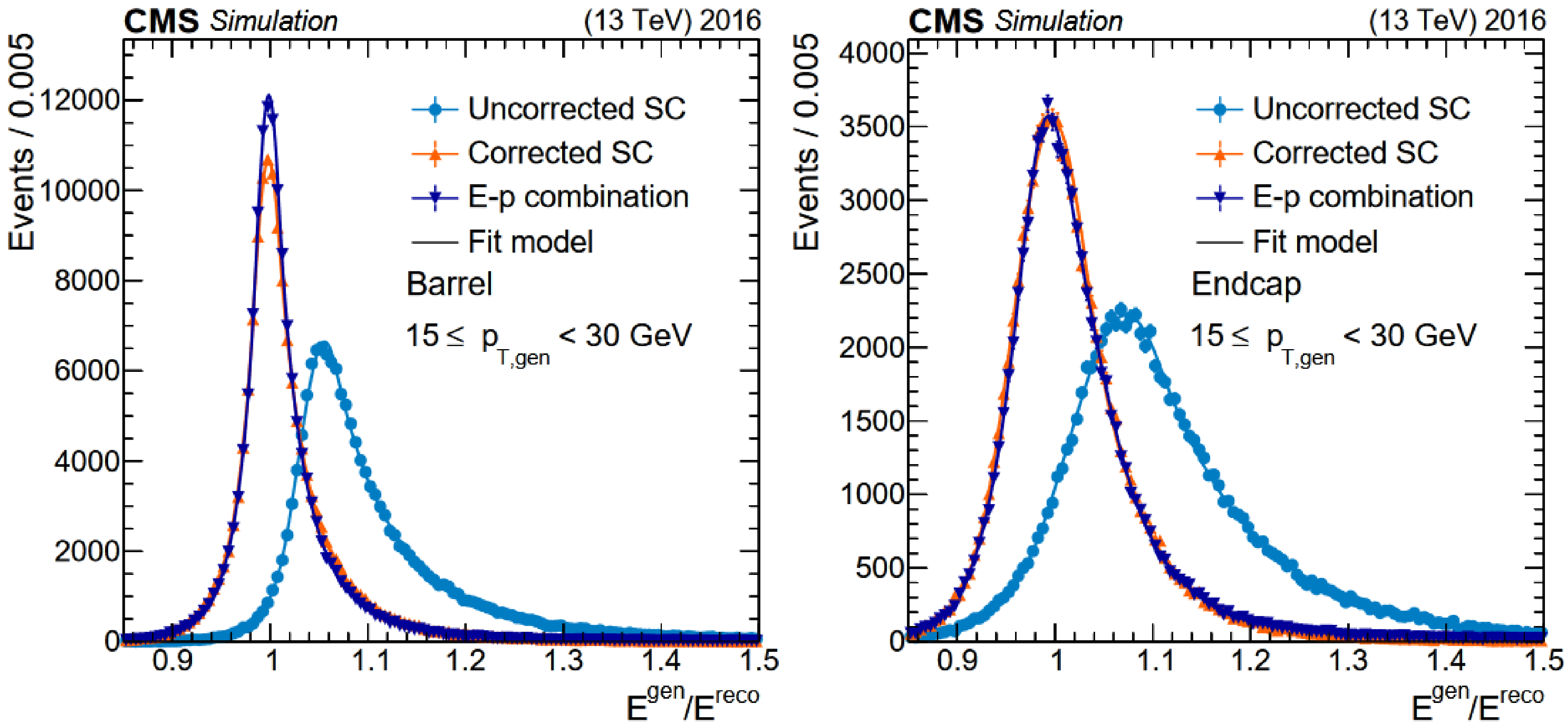


Conversion reconstruction is challenging:
- Displaced tracks to be reconstructed (issues with resolution and combinatorics)
- Trailing conversion leg may be very soft
- Conversion legs can radiate photon

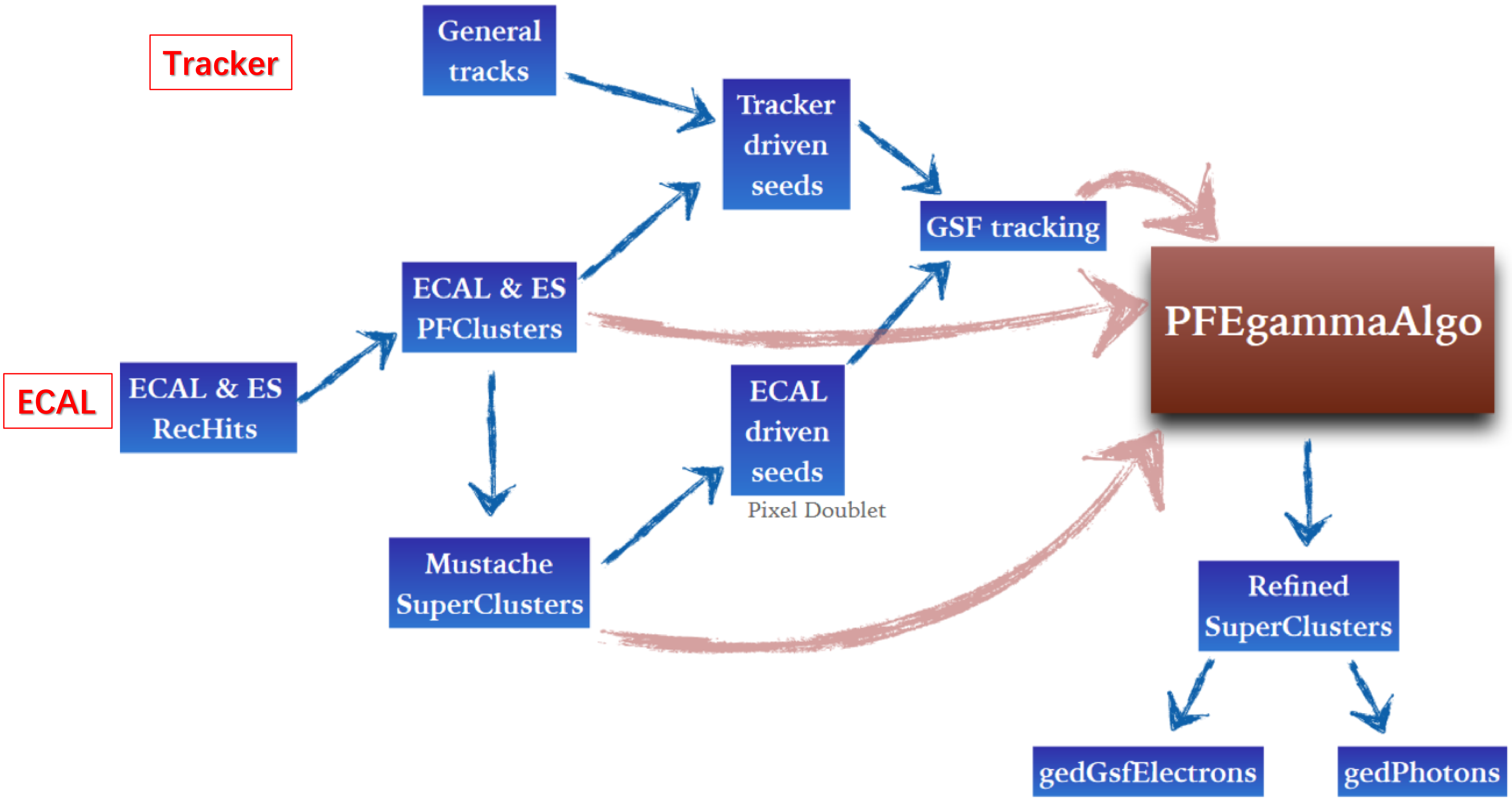
Energy corrections

Several **losses** occur before electrons and photons deposit energy in the ECAL

- We **calibrate the reconstructed energy** back to the expected original energy using **correction** procedures
- Employ **machine learning (Regression)** in tandem with algorithmic approaches
- **Tracker information** used for E-p combination



e and γ reconstruction



Electron and photon identifications

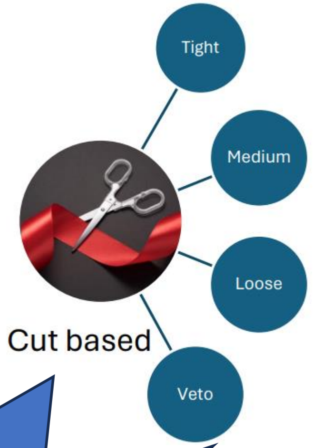
Two schemes are primarily used for identification:

- **Cut-based** selections on various high-level properties
- **Machine learning** based classifiers trained on these high level properties

CutBasedElectronIdentificationRun3

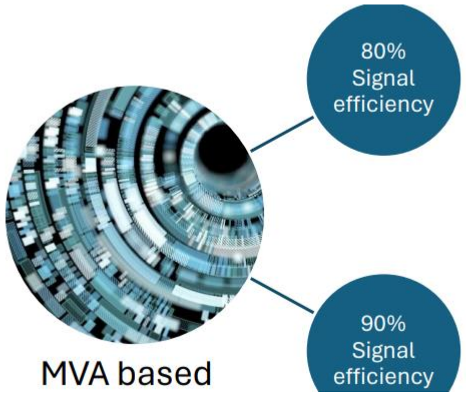
122X-tuned selection, barrel cuts (|eta supercluster| <= 1.479)

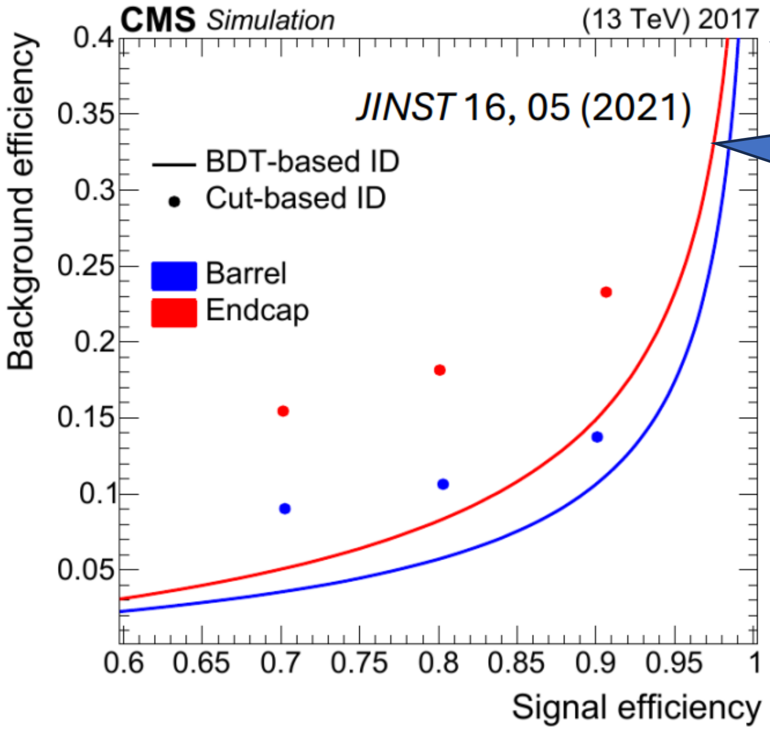
	Veto (122X)	Loose (122X)	Medium (122X)	Tight (122X)	Access in
full5x5_sigmaIetaIeta <	0.0117	0.0107	0.0103	0.0101	link
abs(dEtaSeed) <	0.0071	0.00691	0.00481	0.00411	link
abs(dPhiIn) <	0.208	0.175	0.127	0.116	link
H/E <	$0.05+1.28/E_{SC}+0.0422\rho/E_{SC}$	$0.05+1.28/E_{SC}+0.0422\rho/E_{SC}$	$0.0241+1.28/E_{SC}+0.0422\rho/E_{SC}$	$0.02+1.16/E_{SC}+0.0422\rho/E_{SC}$	link
relIsoWithEA <	$0.406+0.535/p_T$	$0.194+0.535/p_T$	$0.0837+0.535/p_T$	$0.0388+0.535/p_T$	link
abs(1/E-1/p) <	0.178	0.138	0.0966	0.023	link
expected missing inner hits <=	2	1	1	1	link
pass conversion veto	yes	yes	yes	yes	link

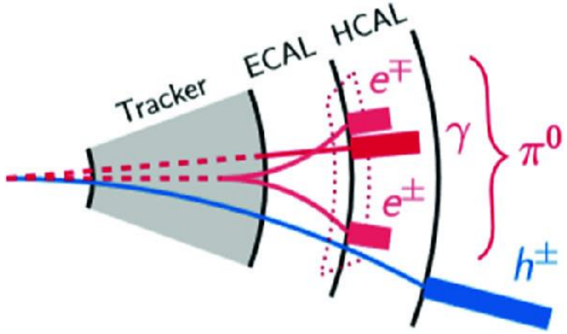


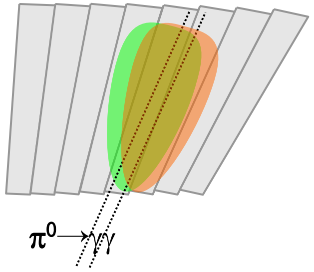
Exercise 1 : photon identifications with machine learning (TMVA BDT)

Exercise 2 : selection efficiencies and data/MC scale factors with $Z\to ee$ tag-and-probe (TnP)



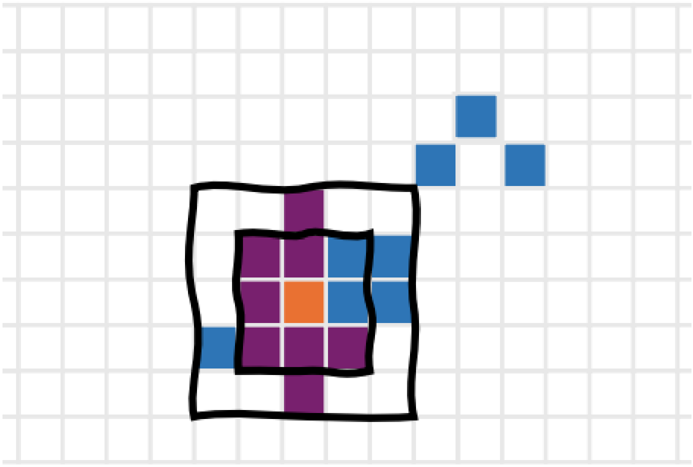






Variables in e/γ id : **EM shower**

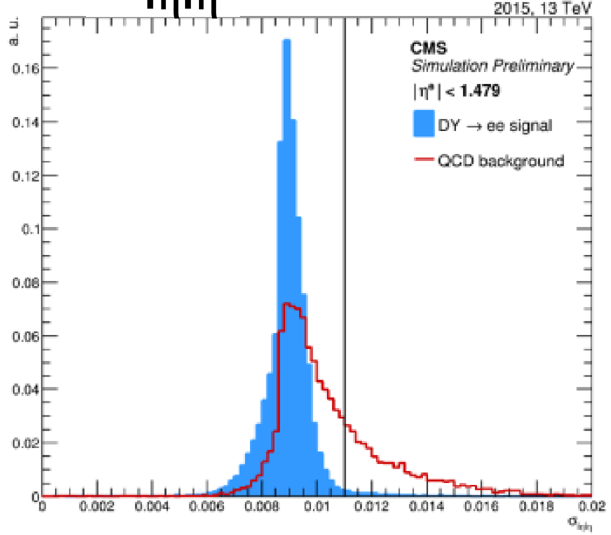
- Description of the **electromagnetic shower** (energy deposit pattern, lateral and longitudinal spread etc.)
- **Tracking and clustering matching parameters** (momentum trajectory extrapolated to ECAL considering the magnetic field etc.)
- Quantification of **isolation** of these objects (Energy sums of crystals in ECAL in a defined area, leakage in HCAL etc.)



E_{sc} : Energy of supercluster
E_{2x2} : Energy contained in 2X2 crystals
E_{3x3} : Energy contained in 3X3 crystals
E_{5x5} : Energy contained in 5X5 crystals
E_{1x5} : Energy contained in 1X5 crystals
$d\eta_{sc}$: η width of supercluster
$d\Phi_{sc}$: Φ width of supercluster

Description of the EM shower shape

$\sigma_{i\eta i\eta}$

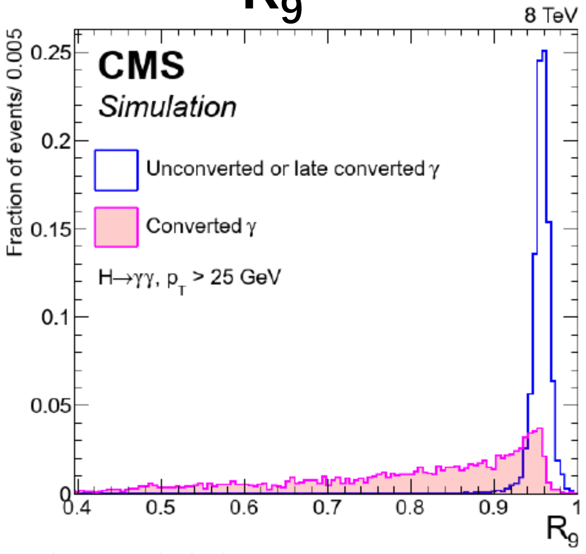


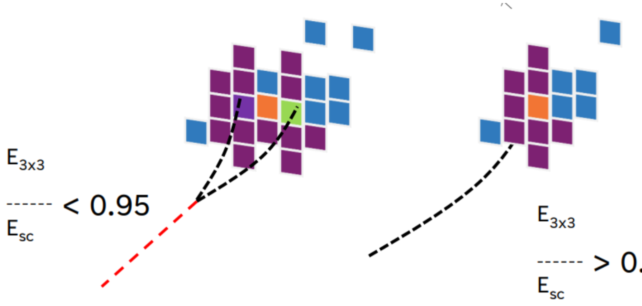
Lateral extension in terms of crystal cells

$$\sigma_{i\eta i\eta} = \sqrt{\left(\frac{\Sigma_i^{5\times5} w_i(\eta_i - \bar{\eta}_{5\times5})^2}{\Sigma_i^{5\times5} w_i}\right)} \quad w_i = 4.7 + \ln\frac{E_i}{E_{5\times5}}$$

Used as template in for example Vγ/VVγ measurements for yields extraction

R_9



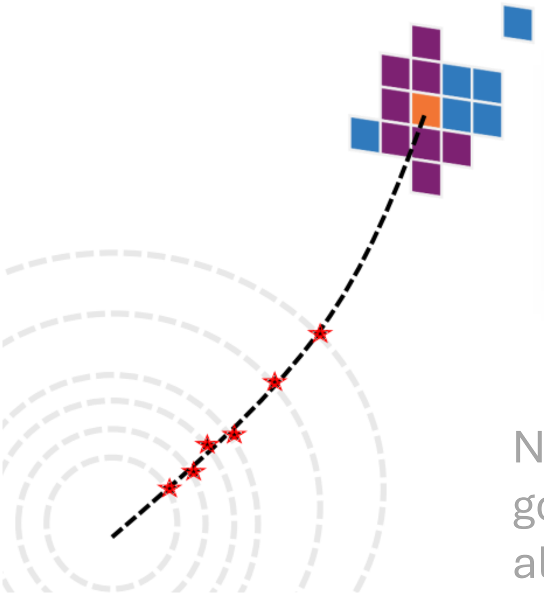


5x5 matrix contains ~96% (~97%) of unconverted photon energy in EB (EE)

$$S_4 = E_{2x2}/E_{5x5}$$

Variables in e/γ id : **tracking + clustering**

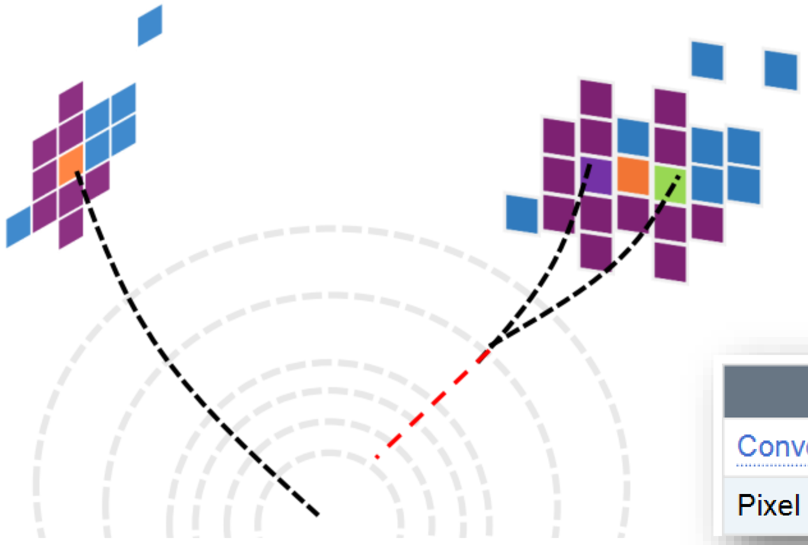
- Description of the **electromagnetic shower** (energy deposit pattern, lateral and longitudinal spread etc.)
- **Tracking and clustering matching parameters** (momentum trajectory extrapolated to ECAL considering the magnetic field etc.)
- Quantification of **isolation** of these objects (Energy sums of crystals in ECAL in a defined area, leakage in HCAL etc.)



N^{gsf}_{hits} : Hits in the "gsf" track
N^{kf}_{hits} : Hits in the "kf" track
E/p : Energy of supercluster/ momentum
χ^2: Track quality

$\Delta\phi$ between track and supercluster

Not just for electrons, even for photons: good ones should not have a "track at all" or would have converted



Conversion ID variables

Conversion Safe Electron Veto
If a secondary vertex is found, this is not an electron!

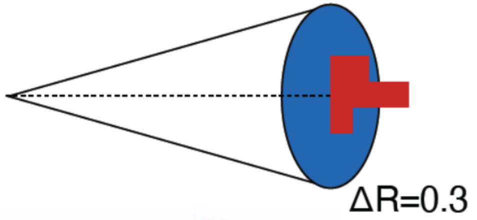
Pixel Veto
Track in pixel detector, this is not a photon

Electron veto	Use case	How to access the variable
Conversion safe electron veto	analysis not sensitive to electron -> photon fake rate	pat::photon->passElectronVeto()
Pixel seed veto	analysis sensitive to electron -> photon fake rate	pat::photon->hasPixelSeed()

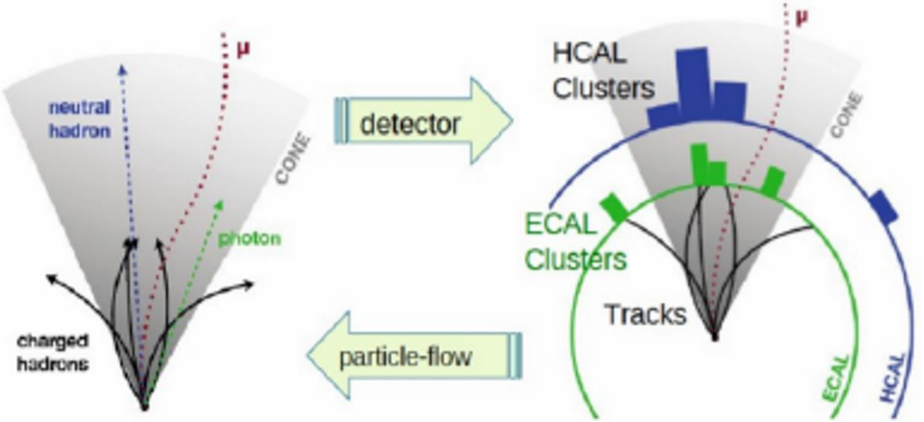
Variables in e/γ id : **isolations**

- Description of the **electromagnetic shower** (energy deposit pattern, lateral and longitudinal spread etc.)
- **Tracking and clustering matching parameters** (momentum trajectory extrapolated to ECAL considering the magnetic field etc.)
- Quantification of **isolation** of these objects (Energy sums of crystals in ECAL in a defined area, leakage in HCAL etc.)

Isolation cone around e/γ

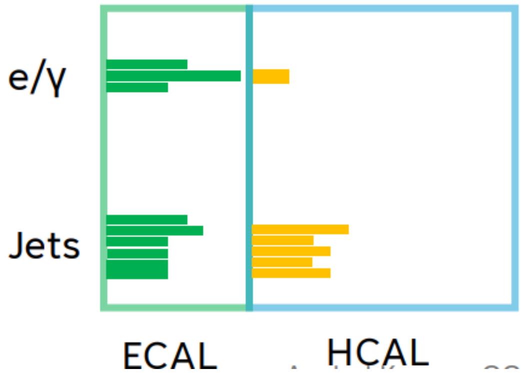


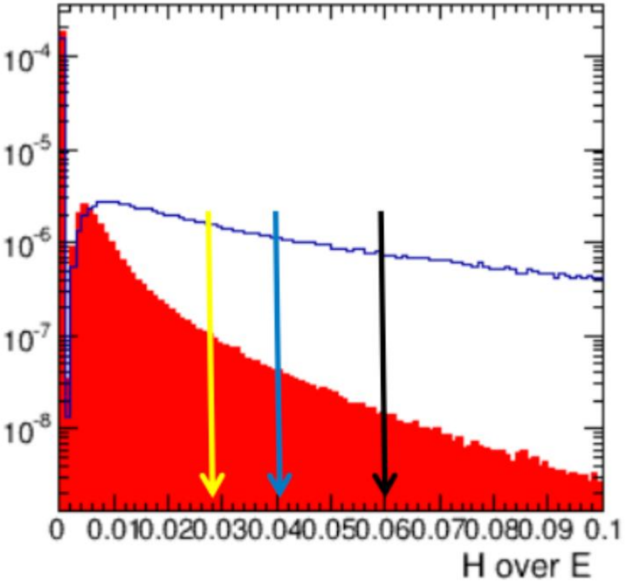
$\Delta R = \mathrm{sqrt}(\Delta\eta^2+\Delta\phi^2) = 0.3/0.4$



H/E : Energy leaked into HCAL / Energy in ECAL

-- HCAL energy within ΔR = 0.15 from the e/γ supercluster



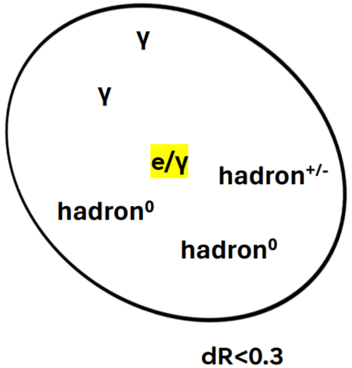


PF Particle Isolations

Charged hadrons

Neutral hadrons

Photons

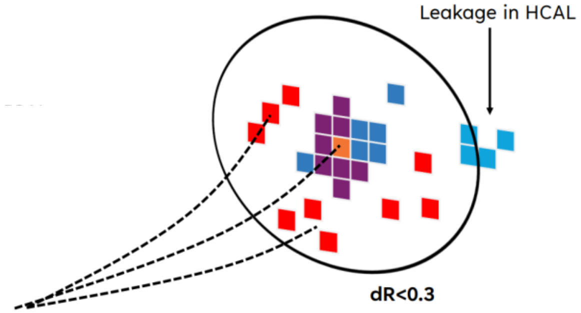


PF Cluster+Tracker isolations

HCAL clusters

ECAL clusters

Tracks



Two exercises

- Exercise 1 (~15min) -- **Photon identifications with machine learning (TMVA BDT)** with photon candidates in EB and EE separately
  - ✓ /publicfs/cms/user/taojq/README_PhotonIDMVA_ForRun2Hgg
  - ✓ ML techniques: MVA, Xgboost, … (more ML in *Congqiao Li*)
  - ✓ Run3: ID MVA trained and provided by EGamma POG

- Exercise 2 (~15min) – e/γ **selection efficiencies and data/MC scale factors with $Z \to ee$ tag-and-probe (TnP) technique**
  - ✓ /publicfs/cms/user/taojq/README_Zee_TnP
  - ✓ Example: selection efficiency with a cut on the photon ID MVA score（>-0.7）
  - ✓ Efficiency of any selections on e/γ

Exercise 1 : photo ID with TMVA

1) Produce comparison plots of the discriminating variables for prompt and fake photons

2) Photon identification training with Machine Learning “TMVA BDTG”

Toolkit for Multivariate Analysis (TMVA)

3) Application of photon ID MVA : evaluate the photon id MVA scores of the photon candidates in any samples (data, $H\rightarrow\gamma\gamma$ MC, ⋯)

Environment setup

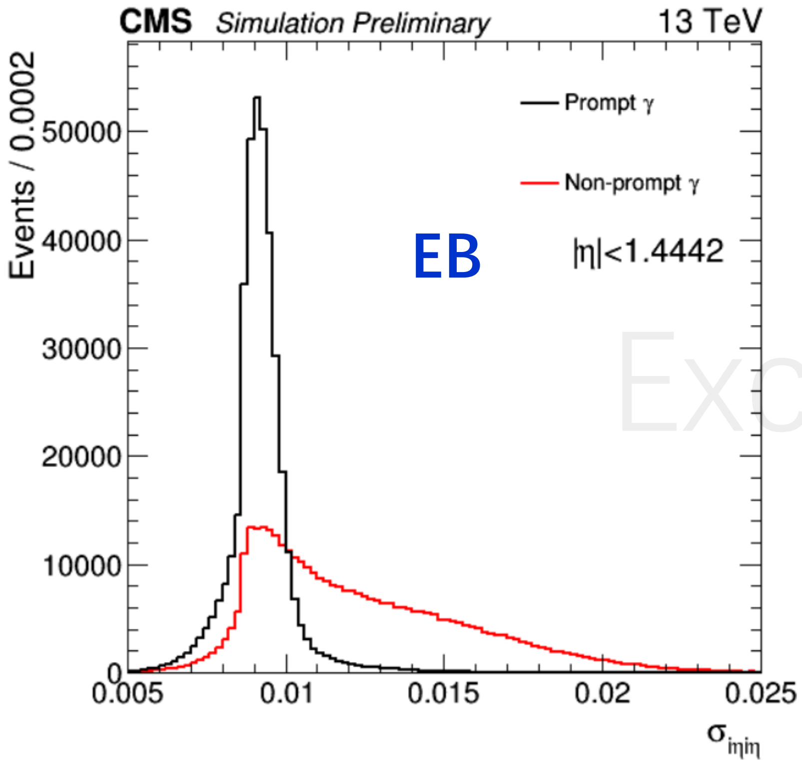
```
cd /publicfs/cms/user/taojq/ (or your home dir)   ← Your working area
#setenv SCRAM_ARCH slc7_amd64_gcc700 (Bash: export SCRAM_ARCH=slc7_amd64_gcc700) (echo $SHELL)
Bash: export SCRAM_ARCH=el9_amd64_gcc12    Csh/TCsh: setenv SCRAM_ARCH el9_amd64_gcc12  (echo $SHELL)
Bash: source /cvmfs/cms.cern.ch/cmsset_default.sh  Csh/TCsh: source /cvmfs/cms.cern.ch/cmsset_default.csh
cmsrel CMSSW_14_1_0
cd CMSSW_14_1_0/src/   ← Can test with any CMSSW version you have checked out (OS9)
cmsenv
mkdir PhotonIDMVA_ForRun2Hgg
cd PhotonIDMVA_ForRun2Hgg
cp /publicfs/cms/user/taojq/PhotonIDMVA_ForRun2Hgg/* .
#just put a soft link to the input file, since it's a large file
ln -s /publicfs/cms/user/taojq/PhotonIDMVA_ForRun2Hgg/InputFile .
```

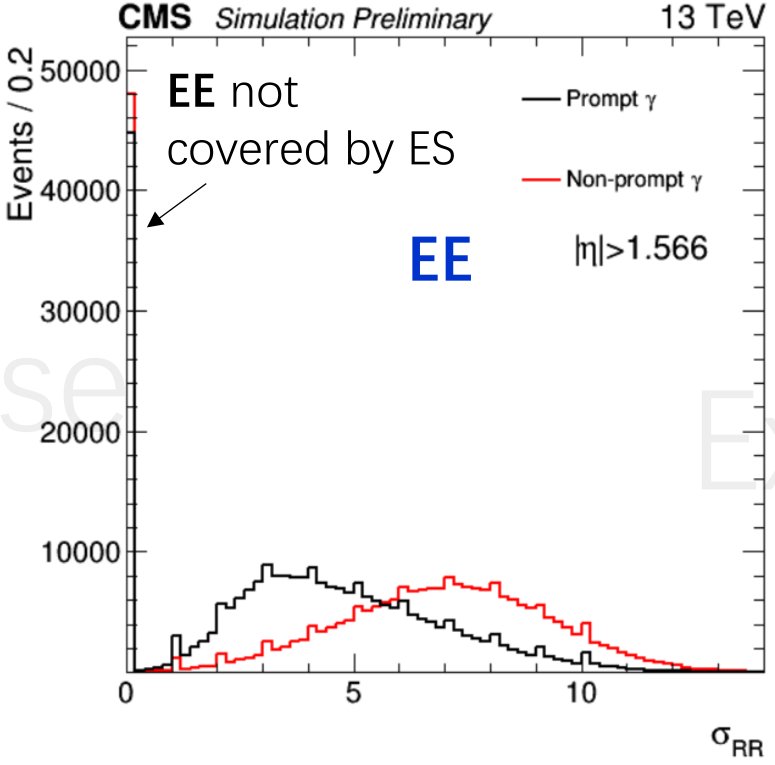
ROOT (+TMVA)

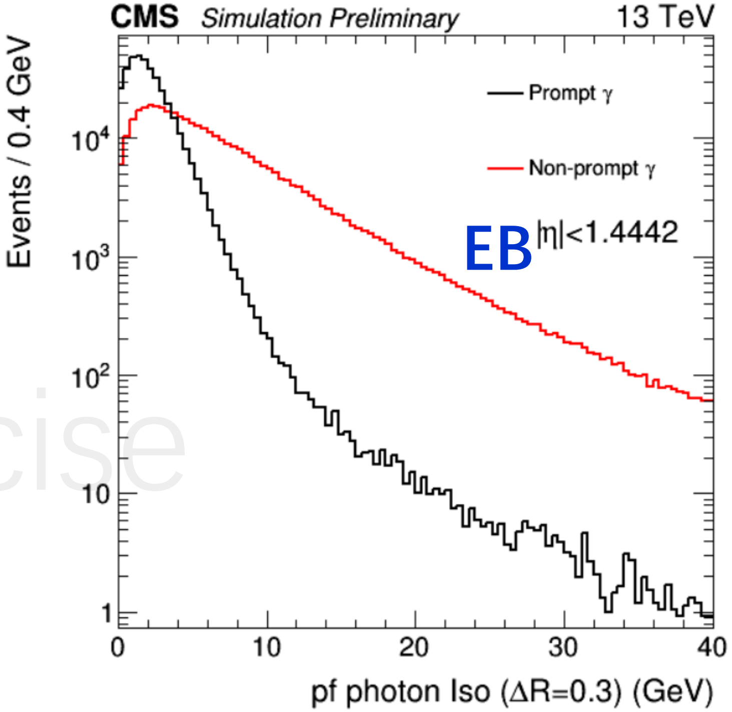
Excise 1 : Produce plots of the identification variables for prompt and fake photons

```
> mkdir SigBkgComparisonsPlots
> root -l -b -q DrawComparisonPlots.C
--> output plots under SigBkgComparisonsPlots/
```

Both prompt and non-prompt photons selected from GJet MC







$\sigma_{i\eta i\eta}$: can be directly used in for example Vγ/VVγ cross section measurements for yields extraction

Preshower σ_{RR}: the standard deviation of the shower spread in the x and y planes of the preshower (only defined in the endcap)

$$\sigma^2_{xx} \equiv \left| \frac{\sum E_i (X_i - \bar{X})^2}{\sum E_i} \right| \qquad \underset{\text{Eff.}}{\sigma_{RR}} \equiv \sqrt{\underset{\text{Eff.}}{\sigma^2_{XX}} + \underset{\text{Eff.}}{\sigma^2_{YY}}}$$

link

Also **PF photon iso** in ΔR=0.4 used in γγ XS measurement (*IHEP…Eur. Phys. J. C (2014) 74:3129*)

Excise 1 : Photon identification with TMVA "BDTG"

- TMVA – BDTG (Gradient Boosting Decision Trees) : could try more ML methods (TMVAClassification.C)

- Train γ-ID MVA for EB and EE separately
  - ✓ Shower shape variables: E_{2x2}/E_{5x5}, $cov_{i\eta i\phi}$, $\sigma_{i\eta i\eta}$, R_9, σ_η, σ_ϕ, Preshower σ_{RR} and E_{ES}/E_{SC} (EE)
  - ✓ Isolation variables: PF Photon ISO, PF Charged ISO (selected vertex), PF Charged ISO (worst vertex)
  - ✓ Other variables: ρ, Supercluster η, Supercluster E_{RAW}

```
> root -l -b -q TMVAClassification_EB.C
> root -l -b -q TMVAClassification_EE.C
```

&

>& t_tmva_eb.log &

```
                       : Ranking input variables (method unspecific)...
IdTransformation       : Ranking result (top variable is best ranked)
                       : --------------------------------------------
                       : Rank : Variable                 : Separation
                       : --------------------------------------------
                       :    1 : pho_PFChWorstIso         : 2.605e-01
                       :    2 : pho_SigmaIEtaIEtaFull5x5 : 2.066e-01
                       :    3 : pho_PFPhoIso             : 2.056e-01
                       :    4 : s4                       : 1.737e-01
                       :    5 : pho_SigmaIEtaIPhiFull5x5 : 1.688e-01
                       :    6 : pho_SCEtaWidth           : 1.077e-01
                       :    7 : pho_PFChIso              : 1.045e-01
                       :    8 : pho_R9Full5x5            : 8.418e-02
                       :    9 : pho_SCPhiWidth           : 4.673e-02
                       :   10 : pho_SCRawE               : 7.673e-03
                       :   11 : pho_SCEta                : 6.271e-03
                       :   12 : evt_rho                  : 1.460e-03
                       : --------------------------------------------
Factory                : Train method: BDTG for Classification
                       :
                       : Events with weight == 0 are going to be simply ignored
BDTG                   : #events: (reweighted) sig: 513110 bkg: 513110
                       : #events: (unweighted) sig: 534514 bkg: 491705
                       : Training 2000 Decision Trees ... patience please
                       : [...............] (1%, time left: 54 mins) min)
```

Half for training and the rest half for testing (homework)

10k of each sig/bkg for **training** (~2min) and 100k of each sig/bkg for testing (~2min)

```
  dataloader->PrepareTrainingAndTestTree( mycuts, mycutb,
//                                    "nTrain_Signal=0:nTrain_Background=0:SplitMode=Random:NormMode=EqualNumEvents:!V" ); //NumEvents
                                   "nTrain_Signal=10000:nTrain_Background=10000:nTest_Signal=100000:nTest_Background=100000:SplitMode=Random:NormMode=EqualNumEvents:!V" ); //test only 10000 events forbkg/sig
```

→ **weight files**: PhotonID_Weight/weights/PhoID_*_Train_GJetMC_BDTG.weights.xml (*=barrel/endcap) (for later application)

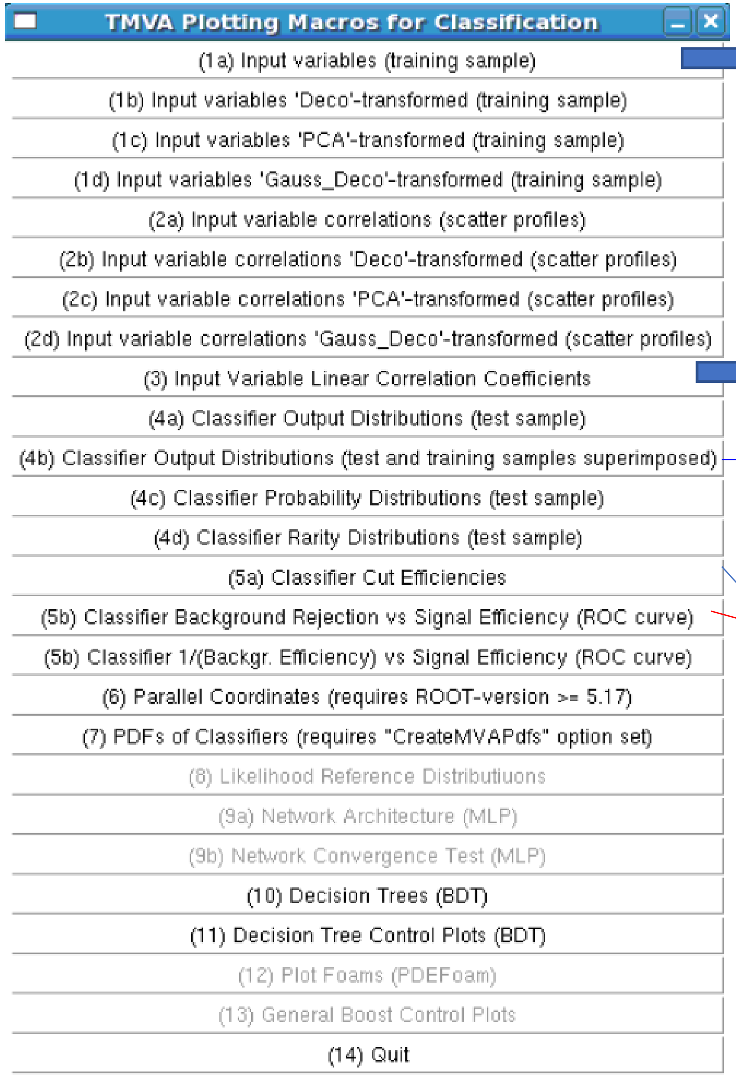
→ **output root files**: PhoID_*_Train_GJetMC.root (*= barrel/endcap)

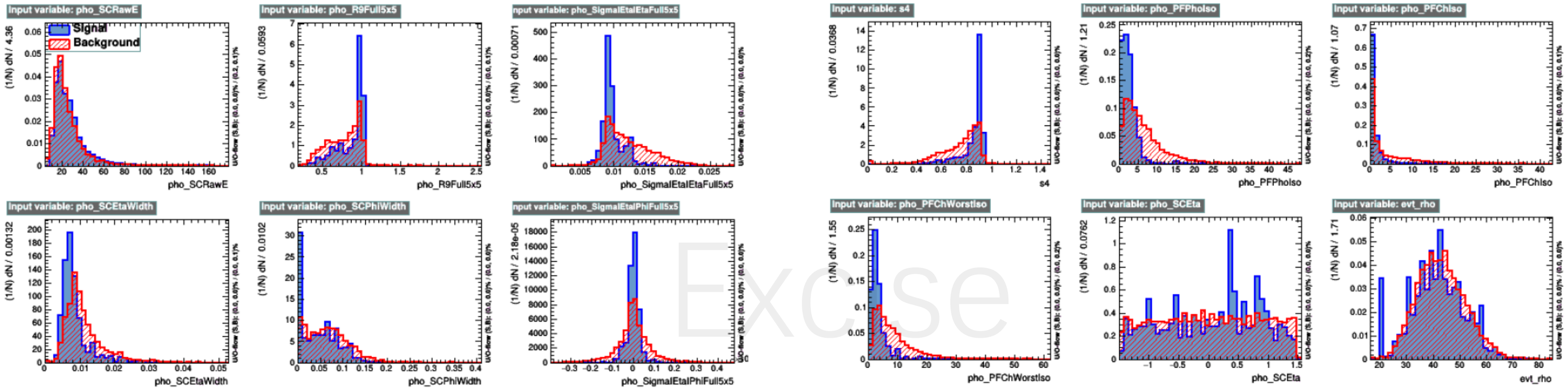
- Once training done, "TMVAGui"

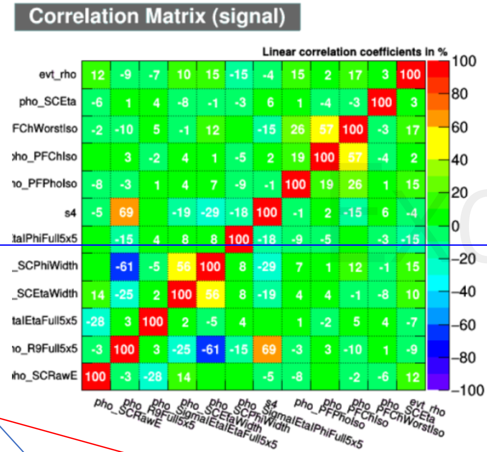
```
>root -l
>TMVA::TMVAGui("PhoID_barrel_Train_GJetMC.root")
--> comparisons of inputs, correlation matrix, output, eff., ....
```

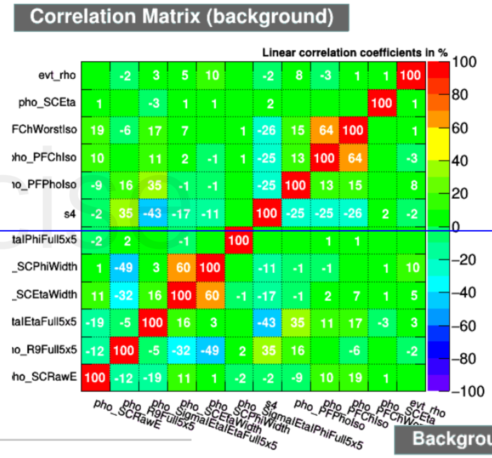
Information in the output root

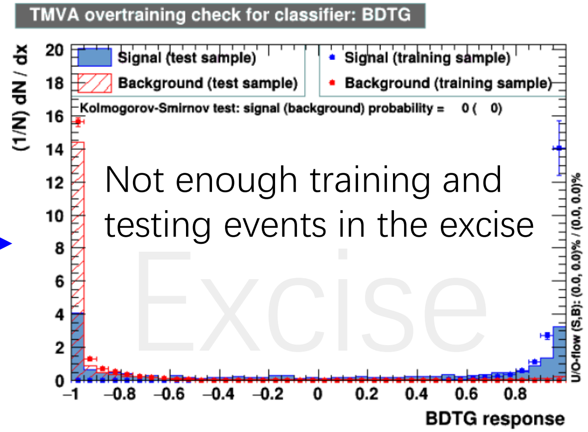
TMVA::TMVAGui("PhoID_barrel_Train_GJetMC.root")



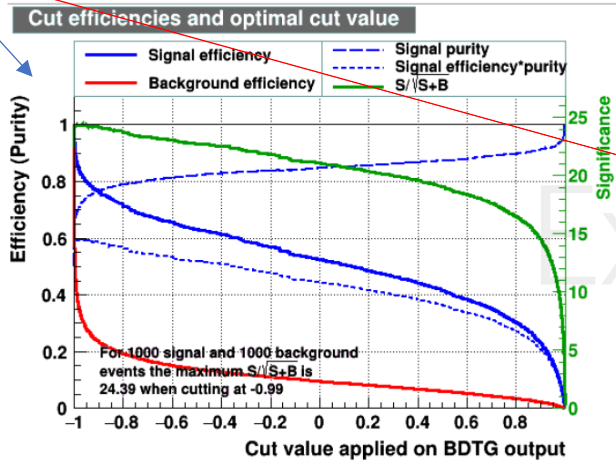


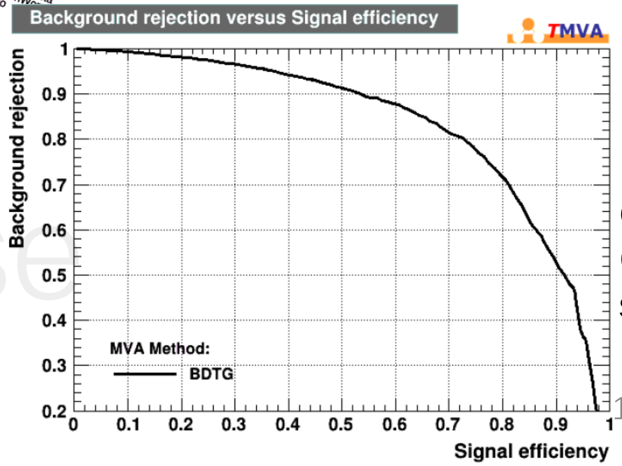






Not enough training and testing events in the excise





ROC: Reciever Operations Characterics

Excise 1 : Application of photon ID MVA

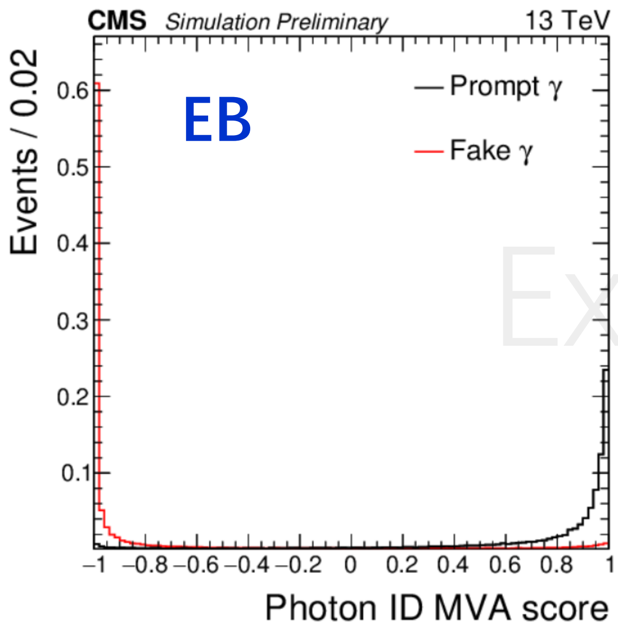
- **Evaluate the photon id MVA scores** of the photon candidates in any samples (Z→ee data/MC, Z→μμγ data/MC, diphoton data/MC for **validations**, data and H→γγ MC for **analysis**)

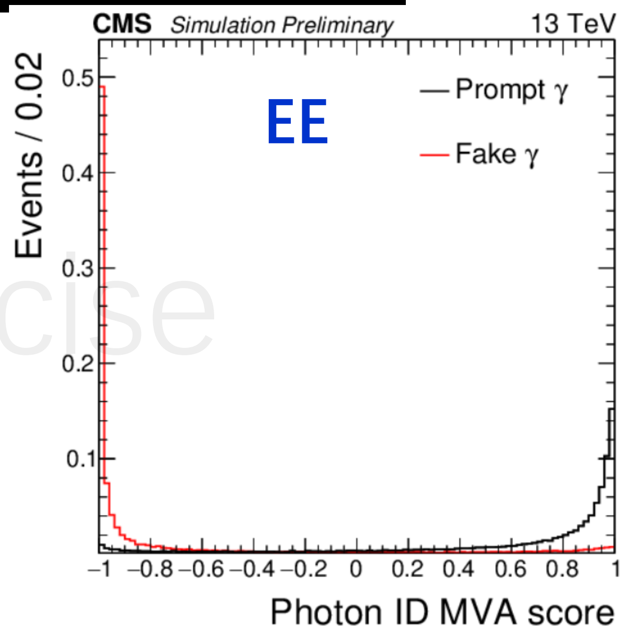
```
> root -l -b -q evaluateMVA_AnySample.C
--> Output_mvares_included.root
    --> Cut on "idmva_res" to suppress bkg on analyses
```

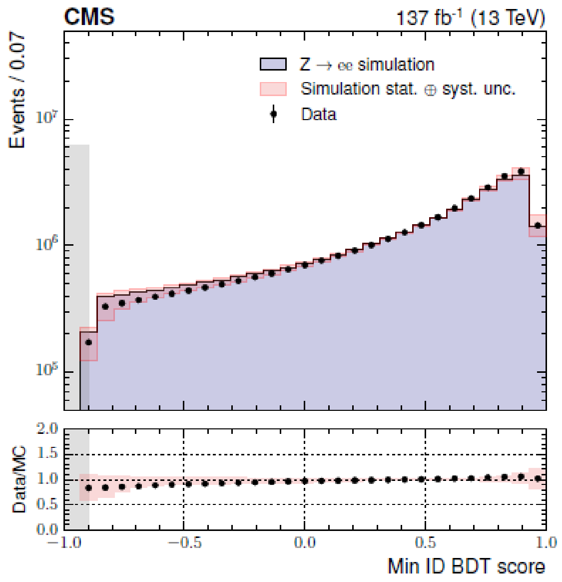
- To obtain **comparison plots** of photon ID MVA between signal photons and fake photons, in EB and EE separately

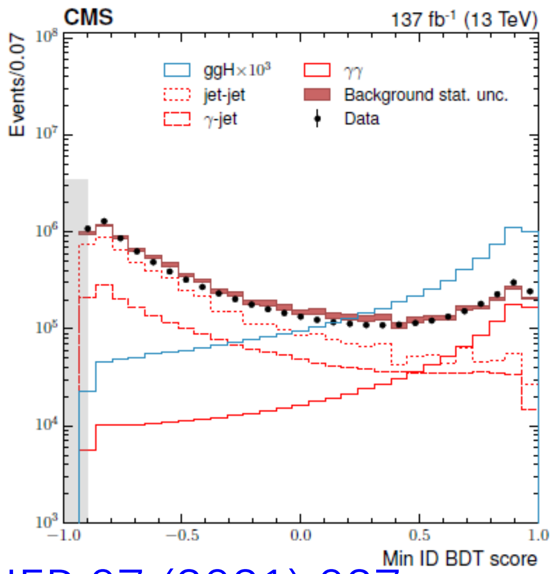
```
> root -l -b -q DrawComparisonPlots_IDMVA.C
#together with plotting style hggPaperStyle.C

--> *.png / *.pdf plots
```









JHEP 07 (2021) 027

Applications in **H→γγ analyses**:

- After **looser cut (>-0.9),** photon ID MVA scores of leading and subleading photons are used as two of inputs of **diphoton BDT** (for event categorization targeting untagged event classes with signal mainly from ggH production mode)
- **Tighter cut** in some dedicated event classes, such as >-0.2 in VBF tagged events

EGmma POG γ-ID MVA : Run2 vs Run3

- Some update on the input features

Run2 EGamma γ ID MVA	Variables	NanoAOD
For both EB and EE	R_9	Photon_r9 (full5x5)
	E_{2x2}/ E_{5x5}	Photon_s4 (full5x5)
	Covinjiφ	Photon_sieip
	$\sigma_{i\eta i\eta}$	Photon_sieie
	η_{width}	Photon_etaWidth
	ϕ_{width}	Photon_phiWidth
	PF Photon Isolation	Photon_pfPhoIso03
	PF charged Isolation wrt choosen vertex	Photon_pfChargedIsoPFPV
	PF Charged Isolation wrt Worst vertex	Photon_pfChargedIsoWorstVtx
	Super cluster raw energy	Photon_energyRaw
	Super cluster eta	Photon_scEta
	Energy density ρ	fixedGridRhoAll (event-level)
EE-only	ES effective sigma	Photon_esEnergyOverRawE
	ES energy/ SC raw energy	Photon_esEffSigmaRR

Same inputs as Run2 H→γγ photon ID MVA: dedicated MVA for vertex determination in H→γγ → "pf Charged Iso wrt choosen vertex"

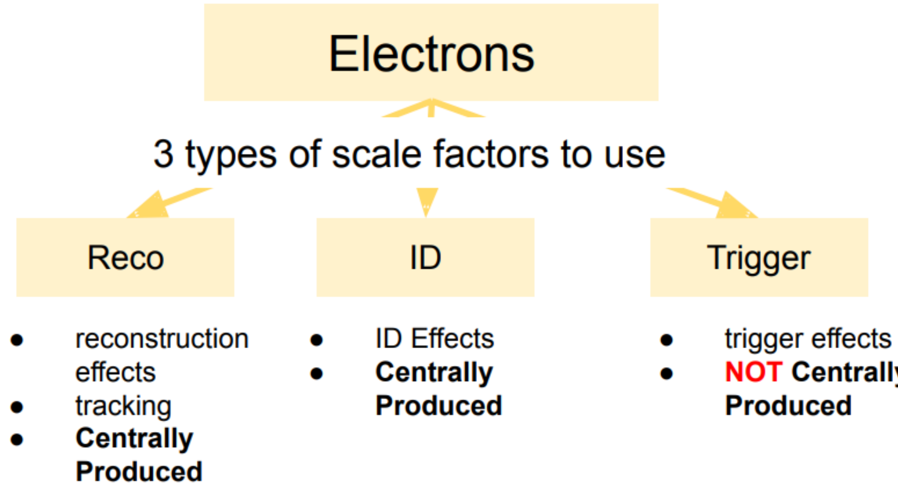
Current Run3 EGamma γ-ID MVA: similar as Run2

- ✓ "PF photon isolation" removed
- ✓ Adding **H/E, Track Iso** (hollow cones 0.3) and Track Iso (solid cone 0.4), **PF ECAL cluster Iso**, **PF HCAL Cluster Iso** (EE-only)
- ✓ Keeping the rest the same as Run2

Exercise 2： selection eff with Z→ee TnP

- **Selection efficiencies and trigger efficiencies** can differ in data and MC
  - ✓ If using MC (signal), want the MC to behave like data
- Scale Factors (SF) are made to match MC efficiencies to Data efficiencies
  - ✓ Defined as the ratio of data efficiencies to MC efficiencies
  - ✓ usually applied as an event weight
- Efficiency-type SF measured via Tag And Probe* OR a different dataset
  - ✓ High pT: $Z\rightarrow\mu^+\mu^-$, $Z\rightarrow e^+e^-$
  - ✓ Low-pT: $J/\psi\rightarrow\mu^+\mu^-$, $J/\psi\rightarrow e^+e^-$

One leg as Tag with tighter selections (ID, HLT ..), the other leg as Probe with looser selections to study the selection efficiencies : peaks in ee/$\mu\mu$ invariant mass spectrum



UL SF documentation

- If you have a custom ID or trigger scheme
  - May need to make your own SFs
    - Present your results to the EGamma POG for approval
  - Reach out to the EGamma contact in your Physics Analysis Group (PAG)
    - EXO, B2G, HIG, etc should have folks
    - The POG keeps a list of approved custom scale factors to prevent duplicate work
      - Often has frameworks already in place

- ☐ Exercise: selection with a cut on the customized photon ID MVA score （>-0.7）: /publicfs/cms/user/taojq/README_Zee_TnP
- ☐ Efficiency of any selections you plan to study

Environment setup

Useful links: https://github.com/cms-egamma/egm_tnp_analysis
https://twiki.cern.ch/twiki/bin/view/CMS/ElectronScaleFactorsRun2

```
#To a container with OS7
/cvmfs/container.ihep.ac.cn/bin/hep_container  shell CentOS7
cd /publicfs/cms/user/taojq/
export SCRAM_ARCH=slc7_amd64_gcc700
source /cvmfs/cms.cern.ch/cmsset_default.sh
cmsrel CMSSW_10_6_29
```

OS7 (OS9 in testing)

Zee TnP package

```
cd /publicfs/cms/user/taojq/CMSSW_10_6_29/src/
cmsenv
#git clone git@github.com:cms-egamma/egm_tnp_analysis.git
cp -r /publicfs/cms/user/taojq/Zee_TnP/egm_tnp_analysis .
cd egm_tnp_analysis
make
#Note: if you modify anything in histUtils.pyx then you need to run make cython-build before make
```

Configurations and modifications

Skip this step, all modifications included in /publicfs/cms/user/taojq/Zee_TnP/egm_tnp_analysis

1) **settings.py**: etc/config/settings_Tao2017.py
   --> **example with UL2017 Zee TnP**
2) **Input samples**: etc/inputs/tnpSampleDef.py
3) Tree name, mass range and binning defined in **tnpEGM_fitter.py**
4) Other modifications: "minPtForSwitch" related in libPython/fitUtils.py, initial values and ranges of "nBkgF" and "nSigF" in libCpp/histFitter.C,...

etc/inputs/tnpSampleDef.py

```
##
TaoEos2017 = '/publicfs/cms/data/Publics/taojq/ZeeTnP/UL17/' #'/eos/cms/store/user/jtao/ZeeTnP/UL17/'
TaoInp2017 = {
'DY_MC_NLO'                : tnpSample('DY_MC_NLO',
                                        TaoEos2017 + 'TnPTree_DYJetsToLL_M-50_TuneCP5_13TeV-amcatnloFXFX-pythia8_UL2017_DY_NLO_EvalIDMVA_Wgt.root',
                                        isMC = True, nEvts =  -1 ),
'DY_MC_LO'                : tnpSample('DY_MC_LO',
                                        TaoEos2017 + 'TnPTree_DYJetsToLL_M-50_TuneCP5_13TeV-madgraphMLM-pythia8_UL2017_DY_LO_EvalIDMVA.root',
                                        isMC = True, nEvts =  -1 ),
'Data_2017' : tnpSample('Data_2017' , TaoEos2017 + 'TnPTree_SingleElectron_UL2017_EvalIDMVA_Wgt.root' , lumi = 41.5 ),
}
```

etc/config/settings_Tao2017.py

Any selections/cuts you plan to study

```
# flag to be Tested
flags = {
    'passingIDMVACut'    : '(ph_mvaID_Mod > -0.7)'
    }
baseOutDir = 'results_UL2017/'
```

Data and MC samples

```
### not: you can setup another sampleDef File in inputs
import etc.inputs.tnpSampleDef as tnpSamples
tnpTreeDir = ''

samplesDef = {
    'data'   : tnpSamples.TaoInp2017['Data_2017'].clone(),
    'mcNom'  : tnpSamples.TaoInp2017['DY_MC_NLO'].clone(),
    'mcAlt'  : None,
    'tagSel' :  None
}
```

pT and η bins

```
########## bining definition  [can be nD bining]
#############################################################
biningDef = [
#    { 'var' : 'ph_sc_eta' , 'type': 'float', 'bins': [-2.5,-2.0,-1.566,-1.4442, -0.8, 0.0, 0.8, 1.4442, 1.566, 2.0, 2.5] },
#    { 'var' : 'ph_et' , 'type': 'float', 'bins': [20.0,35.0,50.0,100.0,200.,1000.0] },
    { 'var' : 'ph_sc_eta' , 'type': 'float', 'bins': [-2.5, -1.566, 0.0, 1.566, 2.5] },
    { 'var' : 'ph_et' , 'type': 'float', 'bins': [20.0, 50.0, 1000.0] },
]
```

Homework: test with more bins

Some preselections

```
#############################################################
########## Cuts definition for all samples
#############################################################
### cut
#cutBase   = 'mass>60. && mass<120. && tag_Ele_et>35. && ph_et>20. && ph_hoe<0.08'
cutBase   = 'mass>60. && mass<120. && tag_Ele_et>35. && ph_et>20. && ph_hoe<0.08 && mcTrue>0.5' ##data mcTrue always ==1
#cutBase   = 'mass>60. && mass<120. && tag_Ele_et>35. && ph_et>20. && mcTrue>0.5' ##data mcTrue always ==1
```

Nominal fits, skip the systematics (homework)

```
###############################################################
########## fitting params to tune fit by hand if necessary
###############################################################
tnpParNomFit = [
    "meanP[-1.0,-2.0,0.1]","sigmaP[0.8,0.1,2.0]",
    "meanF[-1.0,-2.0,0.1]","sigmaF[1.0,0.1,3.0]",
    "acmsP[60.,40.,80.]","betaP[0.03,0.01,0.5]","gammaP[0.05, -0.5, 0.2]","peakP[90.0]",
    "acmsF[60.,40.,80.]","betaF[0.09,0.01,0.5]","gammaF[0.04, -0.5, 0.2]","peakF[90.0]",
    ]

tnpParAltSigFit = [
    "meanP[-2.0,-4.0,0.1]","sigmaP[0.5,0.1,3.0]","alphaP[2.0,0.5,3.5]" ,'nP[3,-5,5]',"sigmaP_2[1.5
,0.5,5.0]","sosP[1.5,0.5,3.0]",
    "meanF[-1.0,-4.0,0.1]","sigmaF[0.5,0.1,3.0]","alphaF[2.0,0.5,3.5]",'nF[3,-5,5]',"sigmaF_2[2.5,
0.5,5.0]","sosF[2.0,0.5,3.5]",
    "acmsP[60.,40.,90.]","betaP[0.2,0.01,0.5]","gammaP[0.05, -0.3, 0.5]","peakP[90.0]",
    "acmsF[60.,40.,100.]","betaF[0.1,0.01,0.3]","gammaF[-0.03, -0.3,0.5]","peakF[90.0]",
    ]

tnpParAltSigFit_addGaus = [
    "meanP[-0.0,-5.0,5.0]","sigmaP[1,0.7,6.0]","alphaP[2.0,1.2,3.5]" ,'nP[3,-5,5]',"sigmaP_2[1.5,0
.5,6.0]","sosP[1,0.5,5.0]",
    "meanF[-0.0,-5.0,5.0]","sigmaF[2,0.7,6.0]","alphaF[2.0,1.2,3.5]",'nF[3,-5,5]',"sigmaF_2[2.0,0.
5,6.0]","sosF[1,0.5,5.0]",
    "meanGF[90.0,70.0,100.0]","sigmaGF[10,1.0,20.0]",
    "acmsP[60.,40.,80.]","betaP[0.03,0.01,0.5]","gammaP[0.05, -1, 1]","peakP[90.0]",
    "acmsF[60.,40.,80.]","betaF[0.03,0.01,0.5]","gammaF[0.03, -1, 1]","peakF[90.0]",
    ]

tnpParAltBkgFit = [
    "meanP[0.3,0.01,1.0]","sigmaP[0.3,0.01,1.0]",
    "meanF[0.3,0.01,1.0]","sigmaF[0.3,0.01,1.0]",
    "alphaP[-0.5,-1.,0.01]",
    "alphaF[-0.5,-1.,0.01]",
    ]
```

S and B parameters

etc/config/settings_Tao2017.py

Fitting

https://twiki.cern.ch/twiki/bin/view/CMS/ElectronScaleFactorsRun2

- Signal
  - Template from MC and smear with Gaussian
- Background
  - RooCMSShape
- Mass window : 60-120 GeV
- MC efficiencies : MC truth matching and tau rejection and then evaluated by counting

Systematics

- Variation tag selection criteria: tight/medium WP, change of tag pT (e.g. 25 to 35)
- Signal model: use a fully parametric model (Double Crystall Ball convoluted with Breit-Wigner + exponential for low pT bins). The usage of the Double CB allows to properly fit also the high pT bins.
- Background modelling: RooCMSShape vs Exponential (keep signal model as the nominal one)
- Different generator: use aMC@NLO (see MC table above)
- Additional sources may be considered in case of photon ID e.g. electron/photon R9 difference, Higgs to Z pT spectrum difference...

Remarks:

- Generator systematics affect only MC so they are derived from a simple counting method.
- In most of the analysis a PU reweigh is already applied thus it is not needed to account for an additional systematic contribution in the scale factor(i.e. double counting).If data/MC corrections for different

than the nominal PU reweighing are needed it is possible to recompute the scale factors with three simple steps:

- recompute the target PU distribution varying the minbias xsec
- update the puTarget in the configuration
- rerun the MC counting fit

Exercise 2： binned histograms

➢ **Check bins**

python tnpEGM_fitter.py etc/config/settings_Tao2017.py --flag passingIDMVACut --checkBins

```
===>  Output directory:
results_UL2017//passingIDMVACut/
bin00_ph_sc_eta_m2p50Tom1p57_ph_et_20p00To50p00
('  - cut: ', 'mass>60. && mass<120. && tag_Ele_et>35. && ph_et>20. && ph_hoe<0.08 && mcTrue>0.5 &&
ph_sc_eta >= -2.500000 && ph_sc_eta < -1.566000 && ph_et >= 20.000000 && ph_et < 50.000000')
bin01_ph_sc_eta_m1p57To0p00_ph_et_20p00To50p00
('  - cut: ', 'mass>60. && mass<120. && tag_Ele_et>35. && ph_et>20. && ph_hoe<0.08 && mcTrue>0.5 &&
ph_sc_eta >= -1.566000 && ph_sc_eta < 0.000000 && ph_et >= 20.000000 && ph_et < 50.000000')
bin02_ph_sc_eta_0p00To1p57_ph_et_20p00To50p00
('  - cut: ', 'mass>60. && mass<120. && tag_Ele_et>35. && ph_et>20. && ph_hoe<0.08 && mcTrue>0.5 &&
ph_sc_eta >= 0.000000 && ph_sc_eta < 1.566000 && ph_et >= 20.000000 && ph_et < 50.000000')
bin03_ph_sc_eta_1p57To2p50_ph_et_20p00To50p00
('  - cut: ', 'mass>60. && mass<120. && tag_Ele_et>35. && ph_et>20. && ph_hoe<0.08 && mcTrue>0.5 &&
ph_sc_eta >= 1.566000 && ph_sc_eta < 2.500000 && ph_et >= 20.000000 && ph_et < 50.000000')
bin04_ph_sc_eta_m2p50Tom1p57_ph_et_50p00To1000p00
('  - cut: ', 'mass>60. && mass<120. && tag_Ele_et>35. && ph_et>20. && ph_hoe<0.08 && mcTrue>0.5 &&
ph_sc_eta >= -2.500000 && ph_sc_eta < -1.566000 && ph_et >= 50.000000 && ph_et < 1000.000000')
bin05_ph_sc_eta_m1p57To0p00_ph_et_50p00To1000p00
('  - cut: ', 'mass>60. && mass<120. && tag_Ele_et>35. && ph_et>20. && ph_hoe<0.08 && mcTrue>0.5 &&
ph_sc_eta >= -1.566000 && ph_sc_eta < 0.000000 && ph_et >= 50.000000 && ph_et < 1000.000000')
bin06_ph_sc_eta_0p00To1p57_ph_et_50p00To1000p00
('  - cut: ', 'mass>60. && mass<120. && tag_Ele_et>35. && ph_et>20. && ph_hoe<0.08 && mcTrue>0.5 &&
ph_sc_eta >= 0.000000 && ph_sc_eta < 1.566000 && ph_et >= 50.000000 && ph_et < 1000.000000')
bin07_ph_sc_eta_1p57To2p50_ph_et_50p00To1000p00
('  - cut: ', 'mass>60. && mass<120. && tag_Ele_et>35. && ph_et>20. && ph_hoe<0.08 && mcTrue>0.5 &&
ph_sc_eta >= 1.566000 && ph_sc_eta < 2.500000 && ph_et >= 50.000000 && ph_et < 1000.000000')
```

➢ **Create the binning**

python tnpEGM_fitter.py etc/config/settings_Tao2017.py --flag passingIDMVACut --createBins

--> results_UL2017//passingIDMVACut//bining.pkl

➢ Create the **histograms** of Zee invariant mass

python tnpEGM_fitter.py etc/config/settings_Tao2017.py --flag passingIDMVACut --createHists

--> results_UL2017/passingIDMVACut/*.root

--> N_eta*N_pt "Pass" and "Fail" TH1D hists

Time-consuming step, depending on the binning (N_eta*N_pt) and input root files

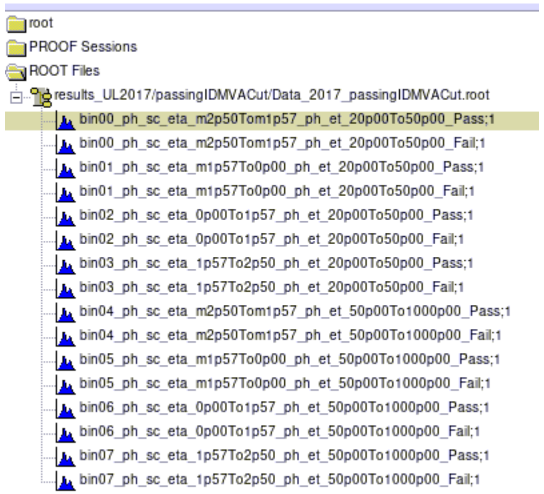
```
Starting event loop to fill histograms..
0 % Data_2017
5 % Data_2017
10 % Data_2017
15 % Data_2017
20 % Data_2017
25 % Data_2017
30 % Data_2017
35 % Data_2017
40 % Data_2017
45 % Data_2017
50 % Data_2017
55 % Data_2017
60 % Data_2017
65 % Data_2017
70 % Data_2017
75 % Data_2017
80 % Data_2017
85 % Data_2017
90 % Data_2017
95 % Data_2017
100 % Data_2017
```

results_UL2017/passingIDMVACut/Data_2017_passingIDMVACut.root and DY_MC_NLO_passingIDMVACut.root

```
TFile*          results_UL2017/passingIDMVACut/Data_2017_passingIDMVACut.root
 KEY: TH1D      bin00_ph_sc_eta_m2p50Tom1p57_ph_et_20p00To50p00_Pass;1
 KEY: TH1D      bin00_ph_sc_eta_m2p50Tom1p57_ph_et_20p00To50p00_Fail;1
 KEY: TH1D      bin01_ph_sc_eta_m1p57To0p00_ph_et_20p00To50p00_Pass;1
 KEY: TH1D      bin01_ph_sc_eta_m1p57To0p00_ph_et_20p00To50p00_Fail;1
 KEY: TH1D      bin02_ph_sc_eta_0p00To1p57_ph_et_20p00To50p00_Pass;1
 KEY: TH1D      bin02_ph_sc_eta_0p00To1p57_ph_et_20p00To50p00_Fail;1
 KEY: TH1D      bin03_ph_sc_eta_1p57To2p50_ph_et_20p00To50p00_Pass;1
 KEY: TH1D      bin03_ph_sc_eta_1p57To2p50_ph_et_20p00To50p00_Fail;1
 KEY: TH1D      bin04_ph_sc_eta_m2p50Tom1p57_ph_et_50p00To1000p00_Pass;1
 KEY: TH1D      bin04_ph_sc_eta_m2p50Tom1p57_ph_et_50p00To1000p00_Fail;1
 KEY: TH1D      bin05_ph_sc_eta_m1p57To0p00_ph_et_50p00To1000p00_Pass;1
 KEY: TH1D      bin05_ph_sc_eta_m1p57To0p00_ph_et_50p00To1000p00_Fail;1
 KEY: TH1D      bin06_ph_sc_eta_0p00To1p57_ph_et_50p00To1000p00_Pass;1
 KEY: TH1D      bin06_ph_sc_eta_0p00To1p57_ph_et_50p00To1000p00_Fail;1
 KEY: TH1D      bin07_ph_sc_eta_1p57To2p50_ph_et_50p00To1000p00_Pass;1
 KEY: TH1D      bin07_ph_sc_eta_1p57To2p50_ph_et_50p00To1000p00_Fail;1
```

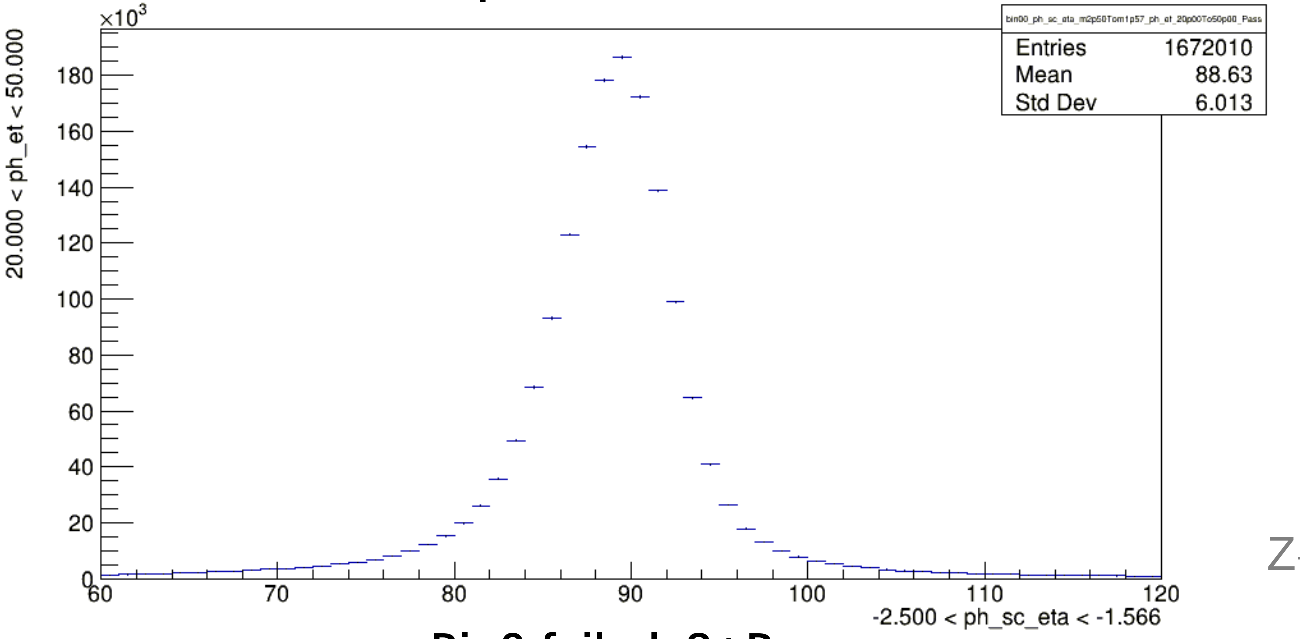
root -l
results_UL2017/passingIDMVACut/Data_2017_passingIDMVACut.root

c=new TBrowser()



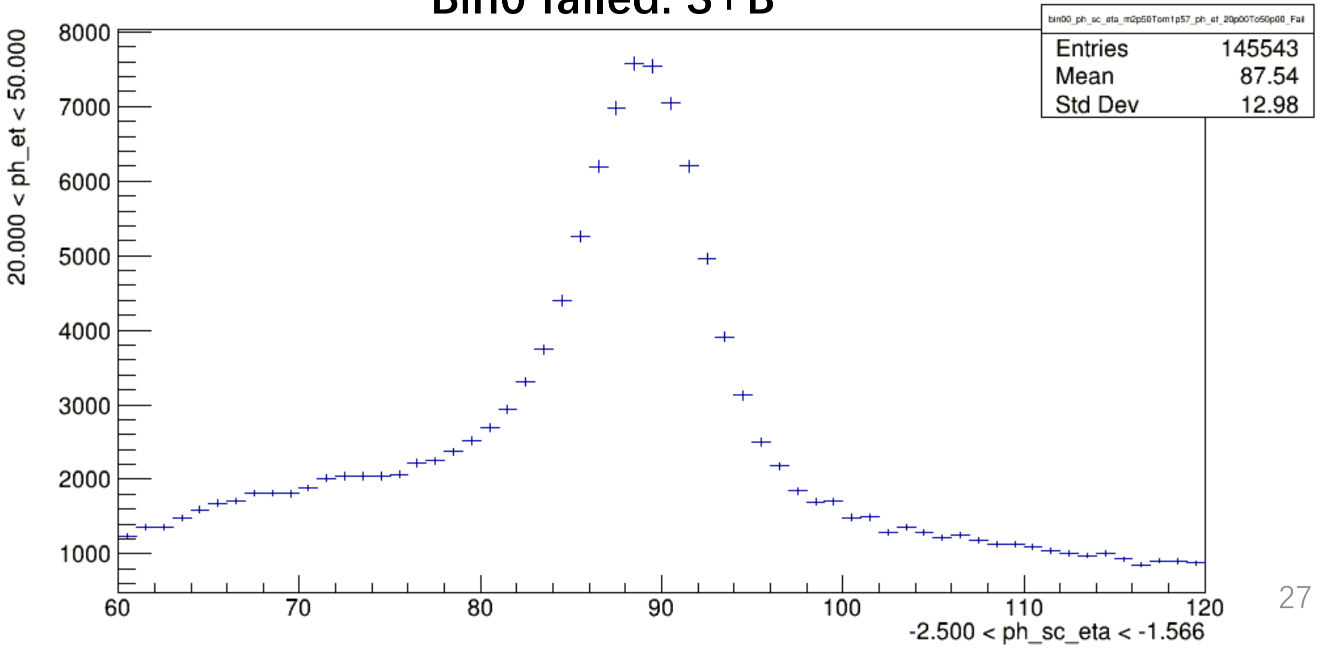
Invariant mass with tag + passed probe, tag + failed probe

Bin0 passed: S+B



$Z \rightarrow e^+e^-$

Bin0 failed: S+B

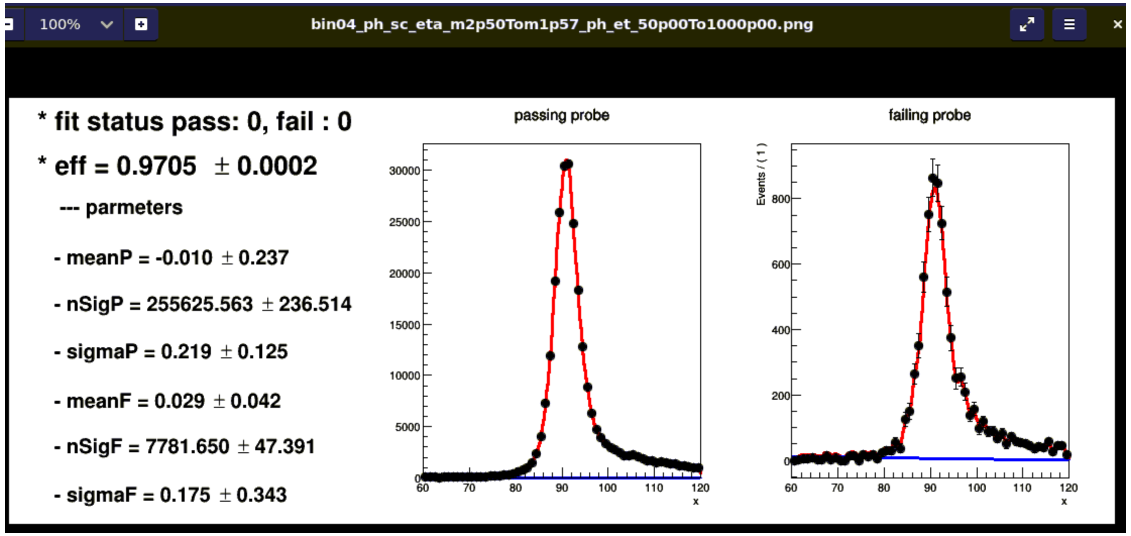


Exercise 2： nominal fitting

$$\text{eff} = \frac{N\_S_{passed}}{N\_S_{passed} + N\_S_{failed}}$$

MC: *python tnpEGM_fitter.py etc/config/settings_Tao2017.py --flag passingIDMVACut --doFit –mcSig*
--> plots: results_UL2017/passingIDMVACut/plots/DY_MC_NLO/nominalFit/*.png
--> rootuples: results_UL2017/passingIDMVACut/DY_MC_NLO_passingIDMVACut.nominalFit-bin*.root

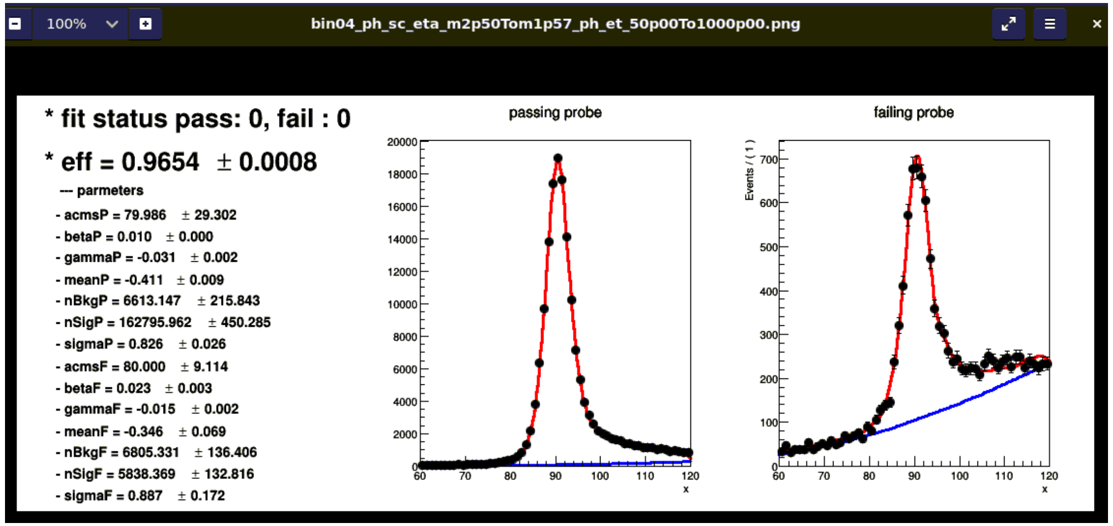
results_UL2017/passingIDMVACut/plots/DY_MC_NLO/nominalFit/



DY MC: only Sig

Data: *python tnpEGM_fitter.py etc/config/settings_Tao2017.py --flag passingIDMVACut --doFit*
--> plots: results_UL2017/passingIDMVACut/plots/Data_2017/nominalFit/*.png
--> rootuples: results_UL2017/passingIDMVACut/Data_2017_passingIDMVACut.nominalFit-bin*.root

results_UL2017/passingIDMVACut/plots/Data_2017/nominalFit/



Data: S+B

Note: may need to tune parameters for some bins

Exercise 2： data/MC scale factors

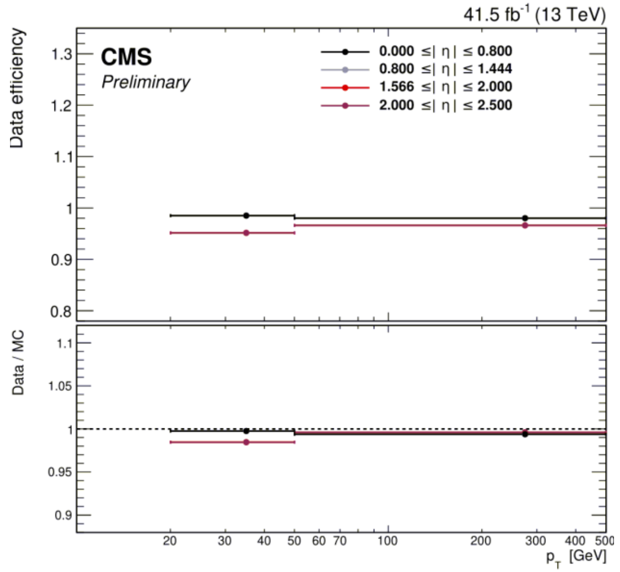
Once all fits are fine, put everything in the egm format txt file, and get the plots of eff and SFs with unc.

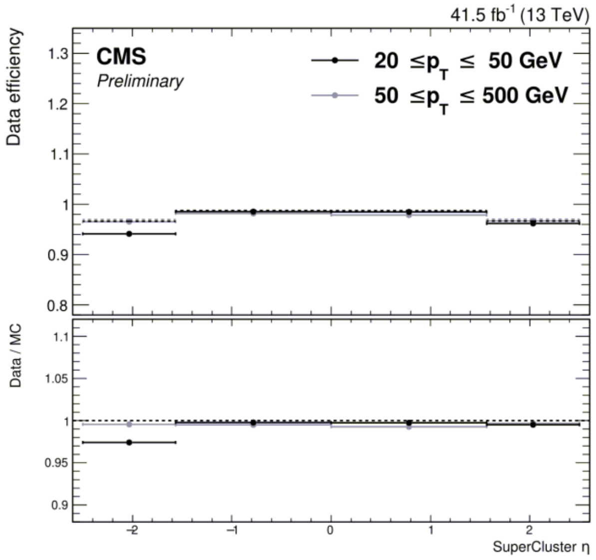
python tnpEGM_fitter.py etc/config/settings_Tao2017.py --flag passingIDMVACut --sumUp

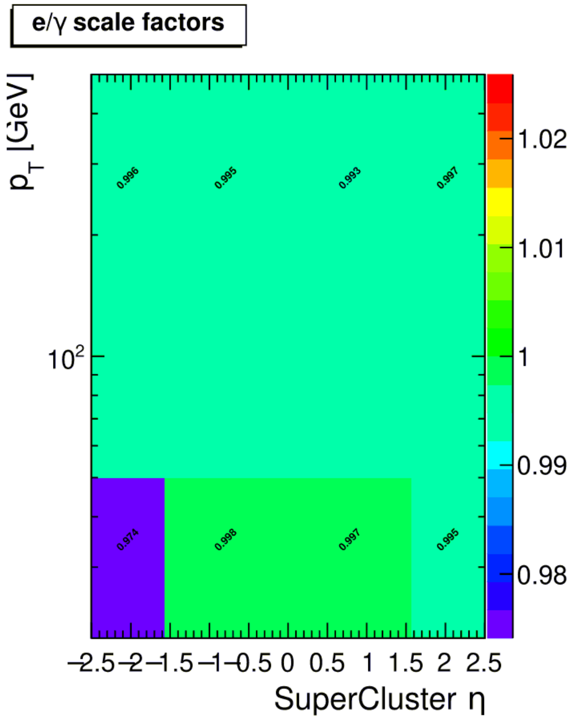
-->
results_UL2017/passingIDMVACut//egammaEffi.txt
--> results_UL2017//passingIDMVACut/*.png
--> results_UL2017//passingIDMVACut/*.pdf

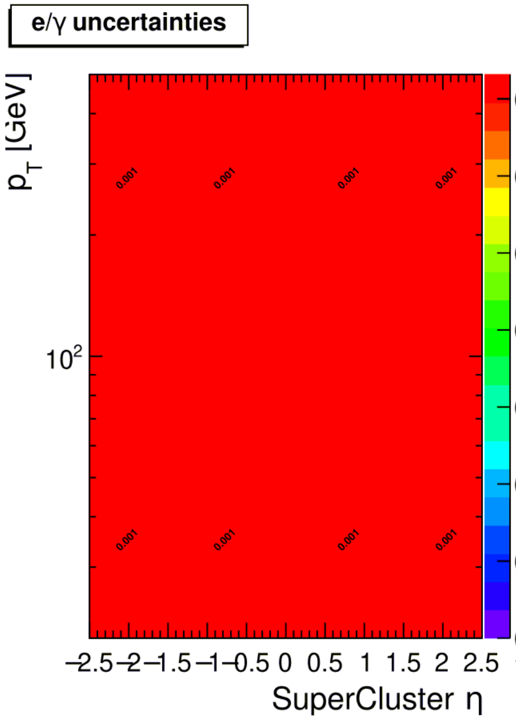
Note: systematics plots are empty (skip the fitting, homework)

SFs are usually applied as an event weight, to correct your MC signal with the cut/selections employed in your analysis

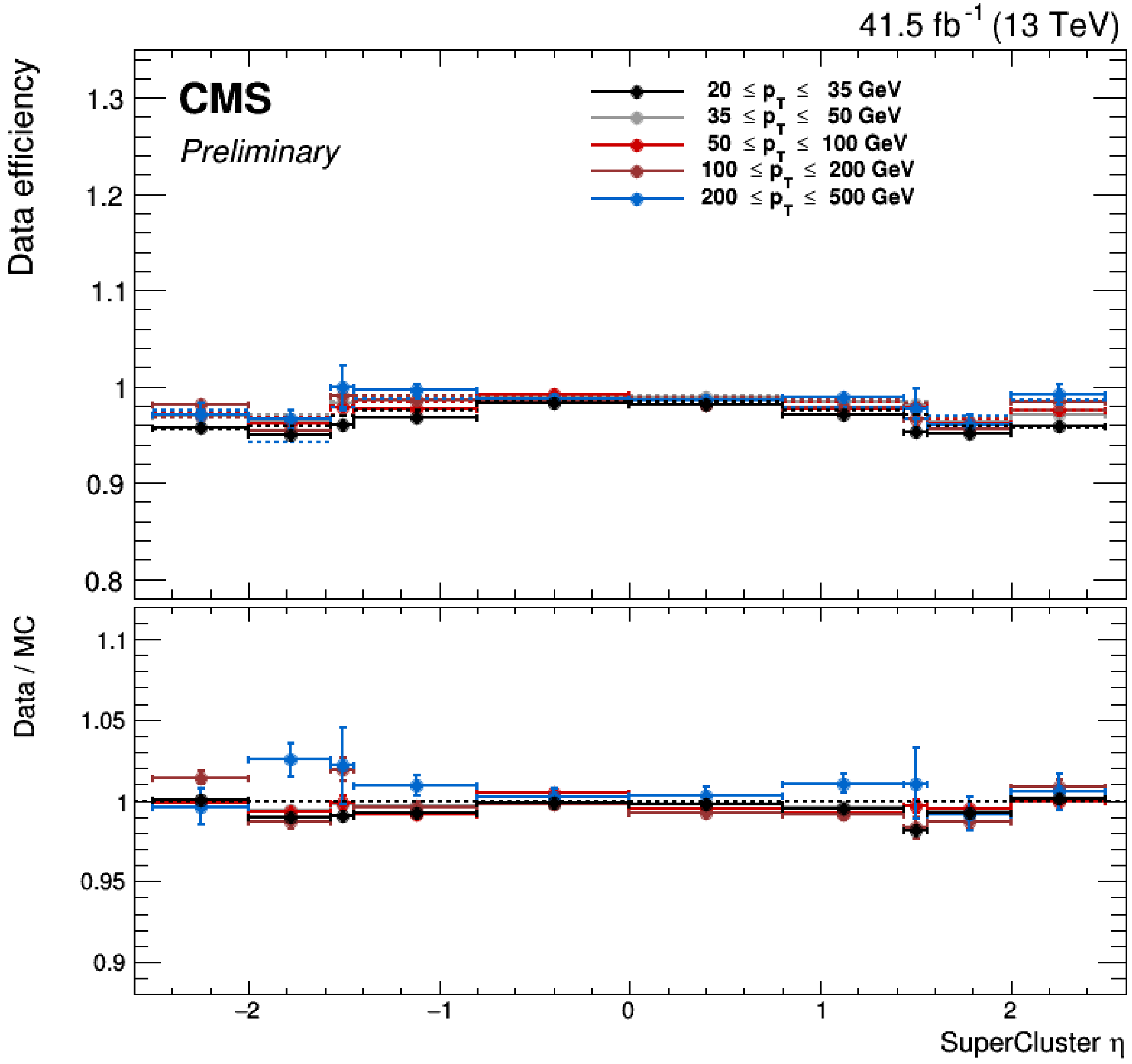


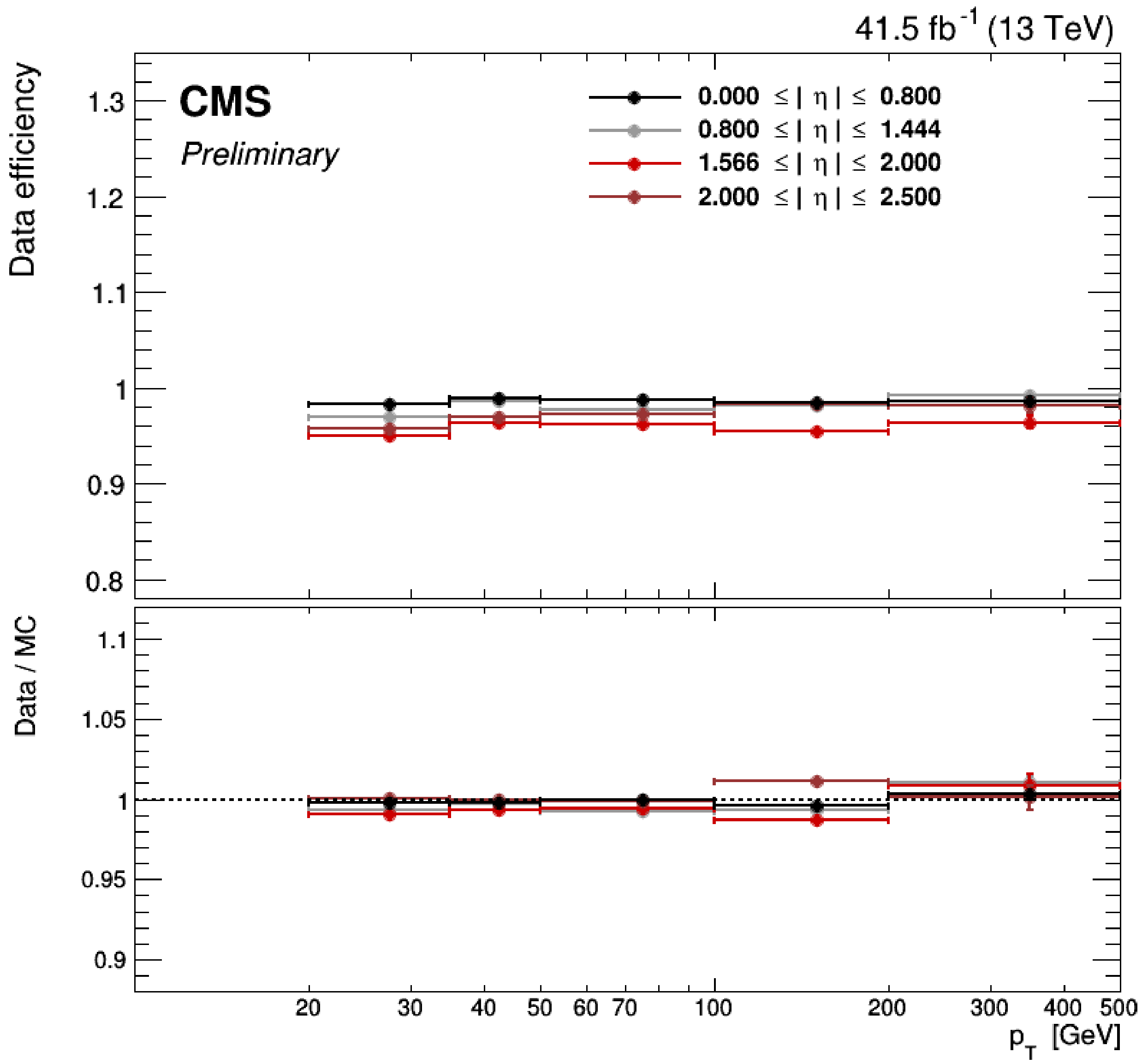






UL2017 data eff. and data/MC SF (stat. unc.)

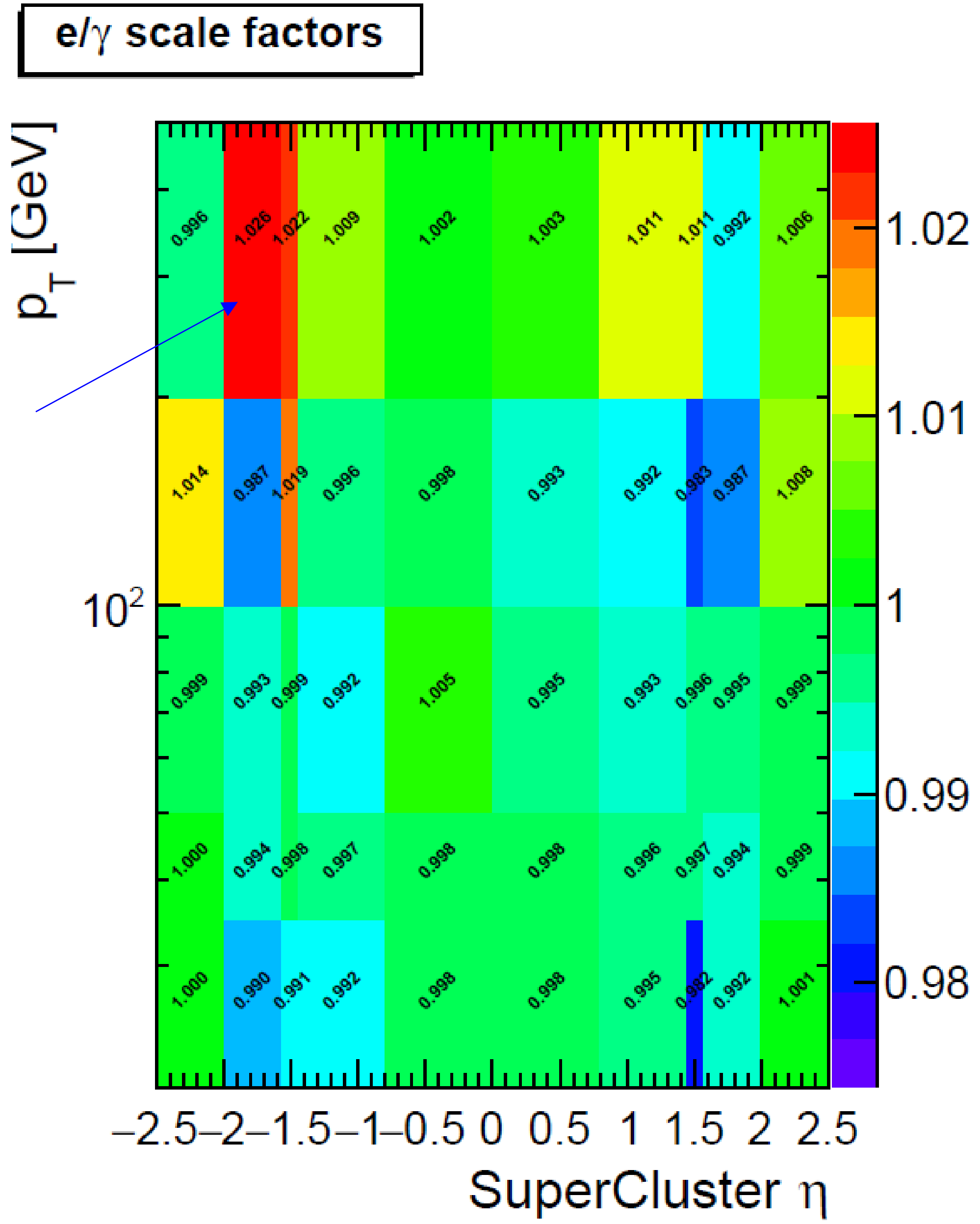


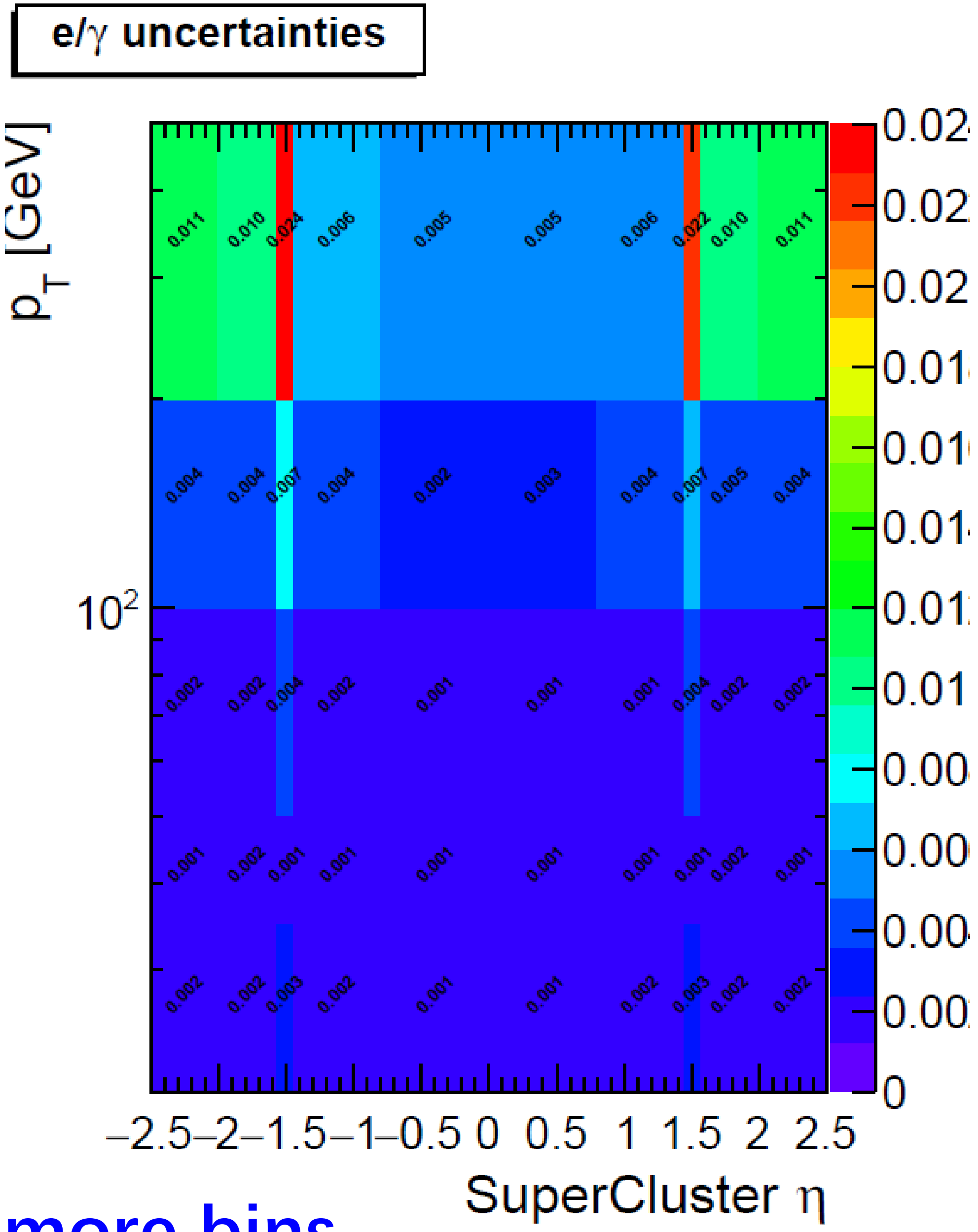


My presentation at EGamma meeting on 15 Mar 2024

more bins

UL2017 SF and stat. unc. : 2D





more bins

SF : <3% different from 1

Except EB/EE gaps, typically stat. unc. <~1%

My [presentation](#) at EGamma meeting on 15 Mar 2024

The End ...Many thanks!