The CTA pipepline prototype

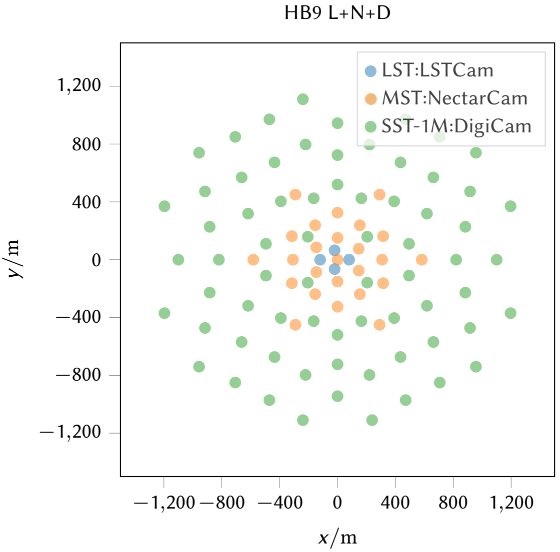
Tino Michael
CEA Paris Saclay, Irfu/DAp

video conference
CEA, 2018-01-15

the whole pipeline is up and running; cover here:

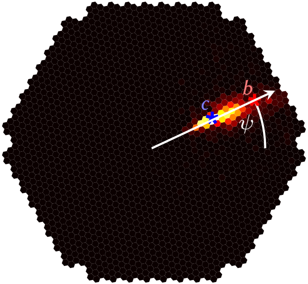
- image cleaning (tailcuts vs. wavelets)
- shower direction and impact reconstruction (purely geometric approach)
- h_{max} estimate (numerical minimisation in 3D)
- energy reconstruction (machine learning)
- event discrimination (machine learning)
- point-source sensitivity

- my github: `https://github.com/tino-michael/tino_cta`
- still needs external libraries for wavelet cleaning



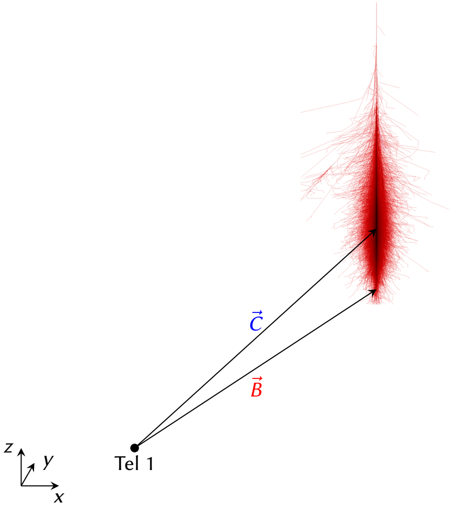
- until recently, only running on ASTRI mini-array
- now on full Paranal array:
- HB9 layout (LST + Nectar + DigiCam)
- pointing north at 20°
- on-axis gammas, diffuse protons and electrons

- Hillas parametrisation provides several parameters:
  - position of image core c
  - tilt of ellipsis ψ
- $\rightarrow$ "arbitrary" point b on shower image axis
- note that every pixel position on the camera can be assosiated to a direction vector in the horizontal frame (e.g. $c \rightarrow \vec{C}$ and $b \rightarrow \vec{B}$)
- $\vec{B}$ and $\vec{C}$ form a plane in which the shower lies



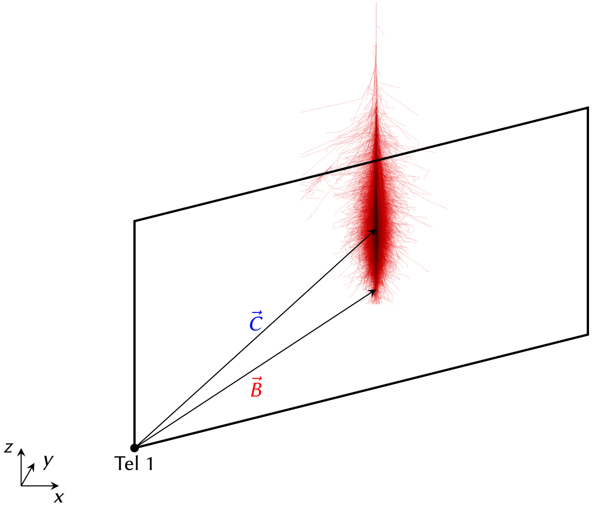
simple shower-reco – direction

horizontal frame



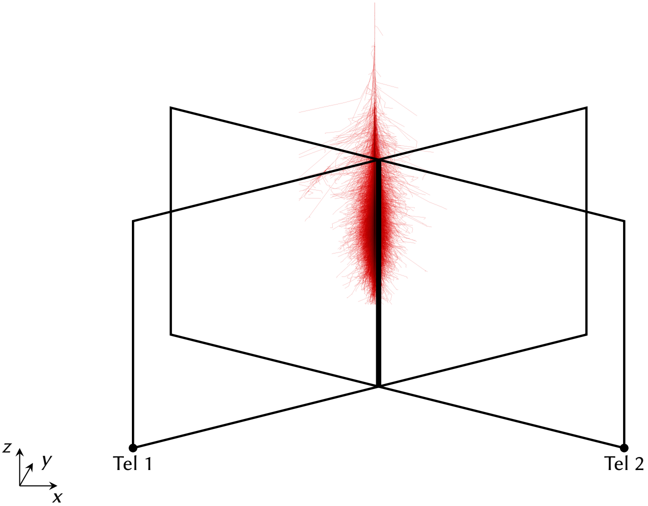
simple shower-reco – direction

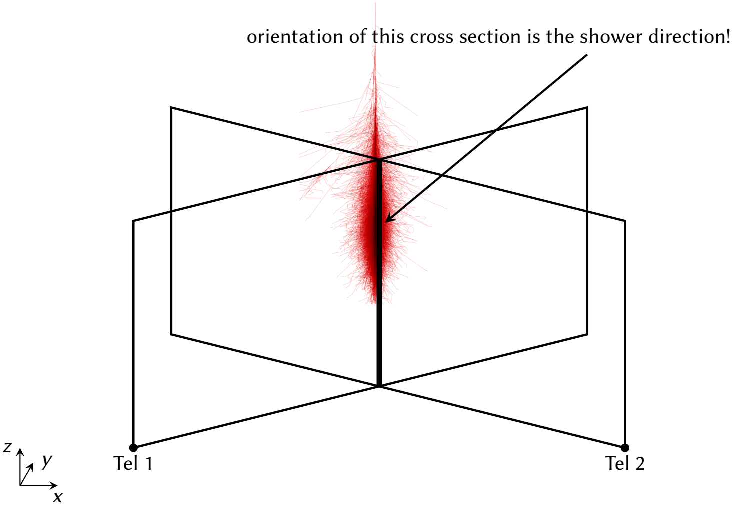
horizontal frame



simple shower-reco – direction

horizontal frame

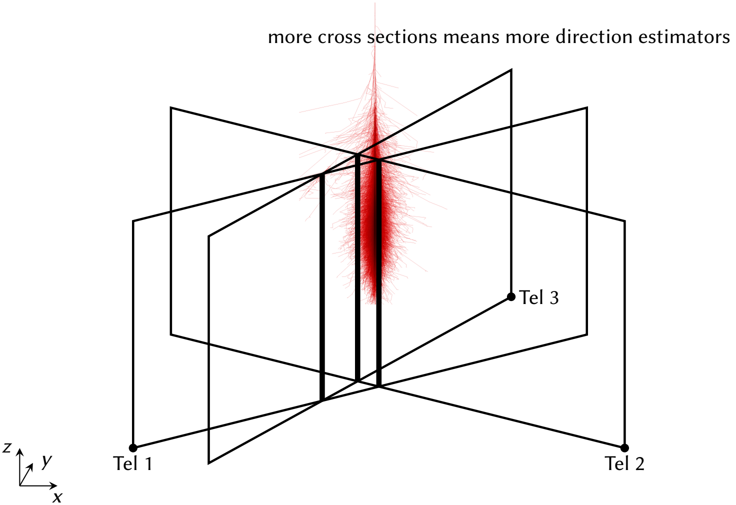




orientation of this cross section is the shower direction!
Tel 1
Tel 2
z
y
x

simple shower-reco – direction

horizontal frame



- this cross section is perpendicular to the normal direction of both intersecting planes ($\vec{n} = \vec{b} \times \vec{c}$)
- $\rightarrow$ shower direction is $\vec{n}_1 \times \vec{n}_2$
- add up all cross products for weighted mean direction:

$$\vec{d}_\gamma = \sum_{i=1}^{N_\text{Tels}} \sum_{j=i+1}^{N_\text{Tels}} w_{ij} \cdot \vec{n}_i \times \vec{n}_j$$

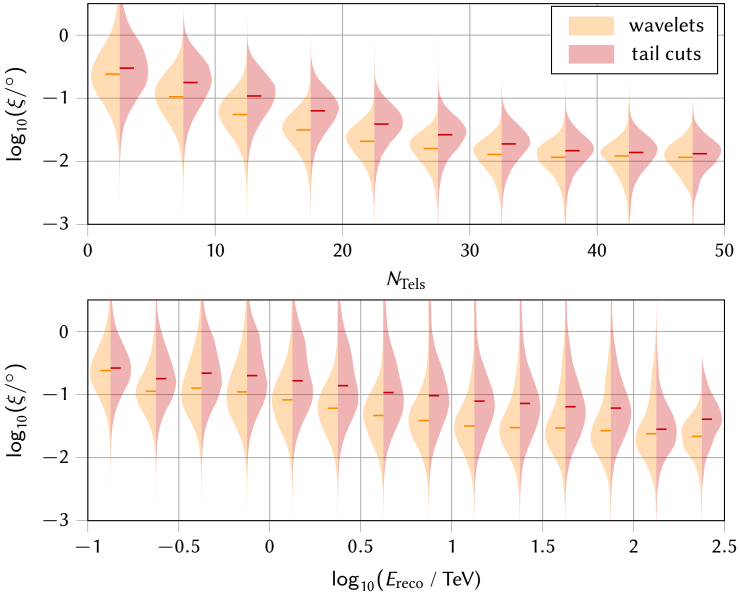
- w_{ij}: weight containing the total image intensity and eccentricity of the Hillas ellipsis
- note: $|\vec{n}_i \times \vec{n}_j| = |\vec{n}_i| \cdot |\vec{n}_j| \cdot \sin[\measuredangle(\vec{n}_i, \vec{n}_j)]$

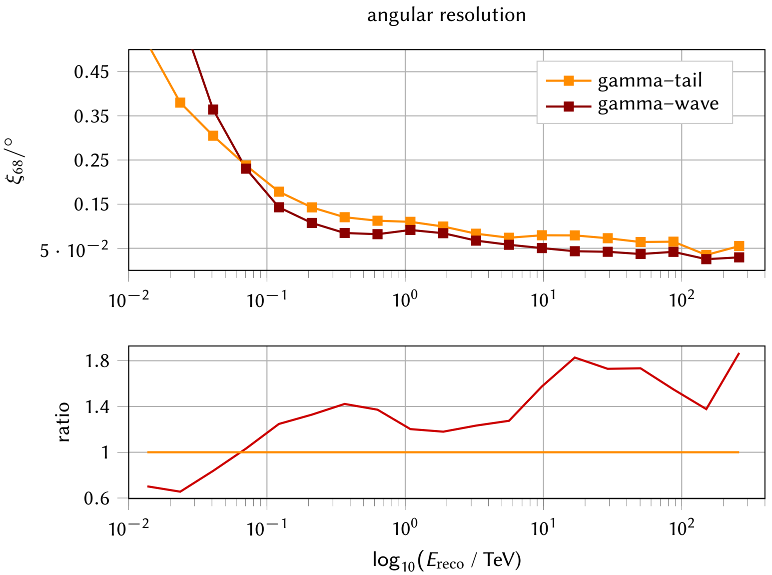
  $\rightarrow$ automatically weights contributions according to the angle between intersecting planes

  $\rightarrow$ planes pairs crossing with more acute angle are weighted less

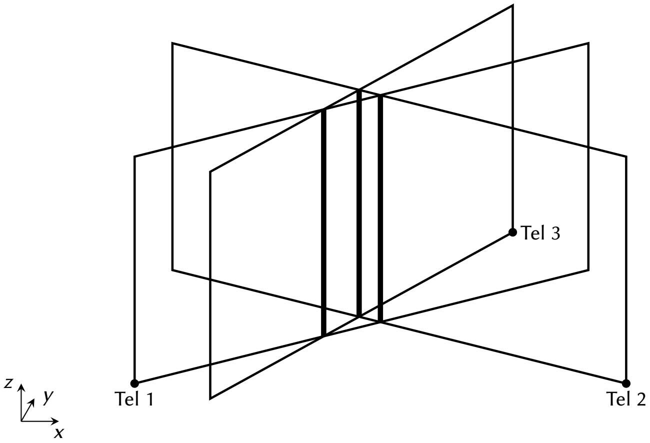
Shower Reconstruction – Direction

Angle between reconstructed and simulated direction

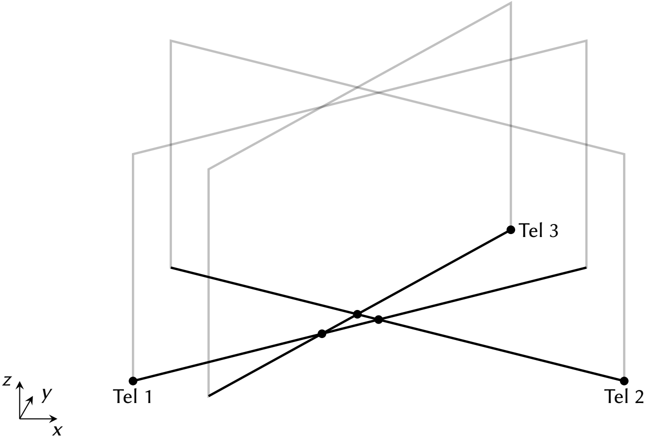


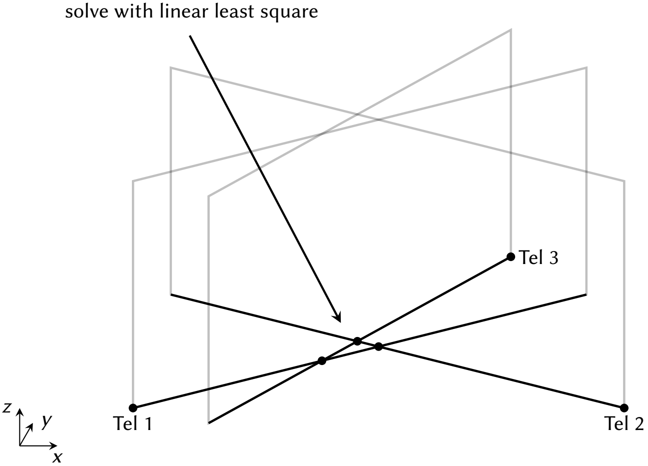


angular resolution
gamma–tail
gamma–wave
$\xi_{68}/°$
0.45
0.35
0.25
0.15
$5 \cdot 10^{-2}$
10^{-2}
10^{-1}
10^{0}
10^{1}
10^{2}
ratio
1.8
1.4
1
0.6
$\log_{10}(E_{\mathrm{reco}}$ / TeV)


Tel 3
Tel 1
Tel 2
z
y
x

simple shower-reco – impact position
horizontal frame





solve with linear least square
Tel 3
Tel 1
Tel 2
z
y
x

For any point $\vec{r}$ on a line i holds

$$\vec{n}_i \cdot \vec{r} = \vec{n}_i \cdot \vec{p}_i = d_i$$

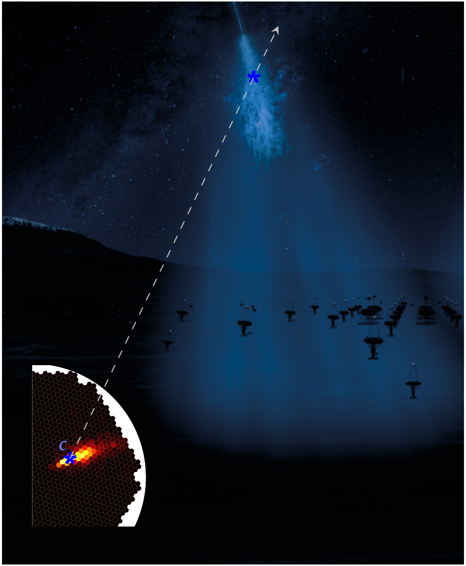
with n_i, the lines normal vector and $\vec{p}_i$, a fixed point on the line (e.g. the telescope position)
If $\vec{r}$ lies on several lines simultaneously, we can write:

$$\begin{pmatrix} n_1^x & n_1^y \\ \vdots & \vdots \\ n_m^x & n_m^y \end{pmatrix} \cdot \vec{r} = \begin{pmatrix} \vec{n}_1 \cdot \vec{p}_i \\ \vdots \\ \vec{n}_m \cdot \vec{p}_m \end{pmatrix} = \begin{pmatrix} d_1 \\ \vdots \\ d_m \end{pmatrix}$$

or $\mathbf{A} \cdot \vec{r} = \vec{d}$.
If all lines i do not cross in one single point $\vec{r}$, there won't be a solution for this equation system.
The "optimal" solution can still be found with least linear squares:

$$\vec{r}_{\chi^2} = (\mathbf{A}^\mathrm{T} \cdot \mathbf{A})^{-1} \cdot \mathbf{A}^\mathrm{T} \cdot \vec{d}$$

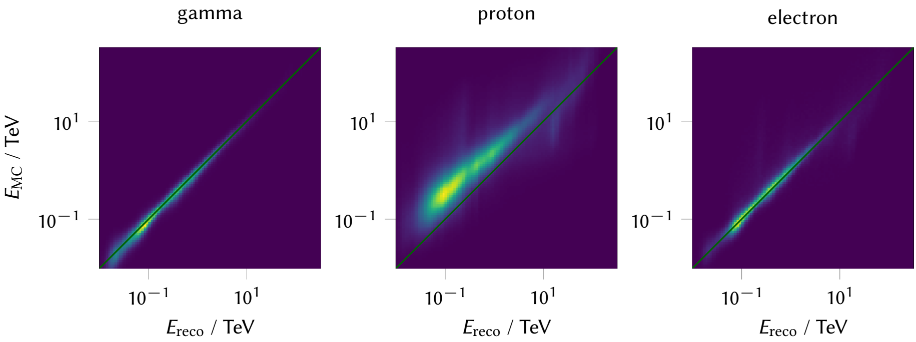
simple shower-reco – h_{max}



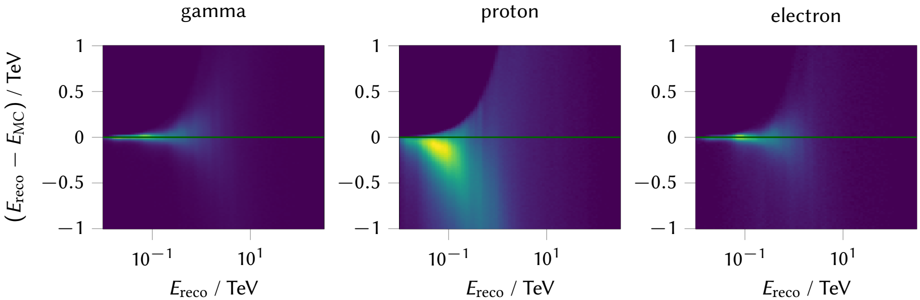
- project the core position of the Hillas ellipsis (vector $\vec{c}$ from direction reco) as a line into the sky
- find the point that minimises the average distance to lines from all telescopes
- no unambiguous normal parametrisation of a line in $\mathbb{R}^3$ ☹
- need to use numeric minimiser after all

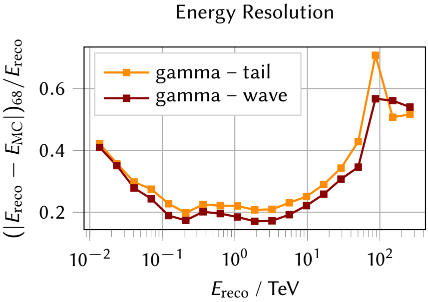
- train 1 Random Decision Forest for each telescope type
- follow a telescope-by-telescope approach
- training features include: Hillas length/width/higher moments, number of telescopes (per size), signal on telescope, summed signal on all triggered telescopes, distance between telescope, shower impact, error estimators, ...

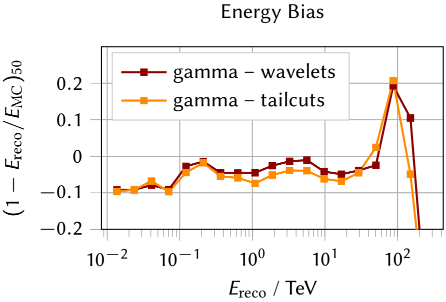
- then, for a given shower event, let the Forest estimate the energy from every telescope separately and combine them into a single energy estimator

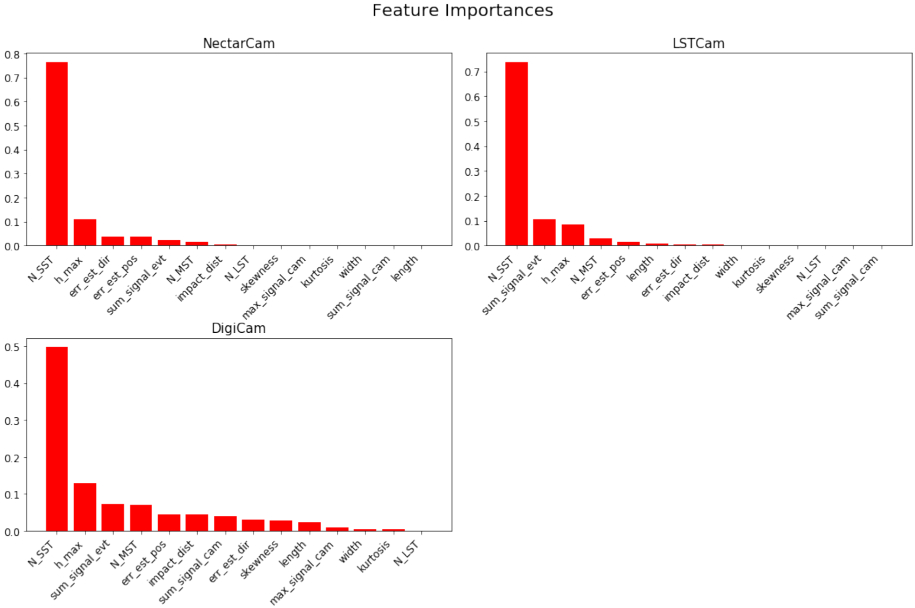


Shower Reconstruction – Shower Energy



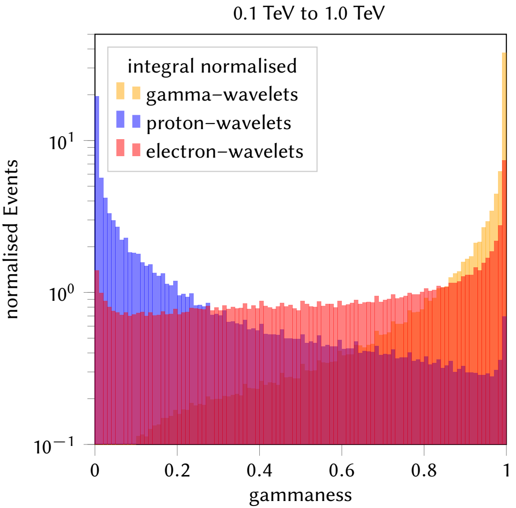






Feature Importances
NectarCam
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
N_SST
h_max
err_est_dir
err_est_pos
sum_signal_evt
N_MST
impact_dist
N_LST
skewness
max_signal_cam
kurtosis
width
sum_signal_cam
length
LSTCam
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
N_SST
sum_signal_evt
h_max
N_MST
err_est_pos
length
err_est_dir
impact_dist
width
kurtosis
skewness
N_LST
max_signal_cam
sum_signal_cam
DigiCam
0.5
0.4
0.3
0.2
0.1
0.0
N_SST
h_max
sum_signal_evt
N_MST
err_est_pos
impact_dist
sum_signal_cam
err_est_dir
skewness
length
max_signal_cam
width
kurtosis
N_LST

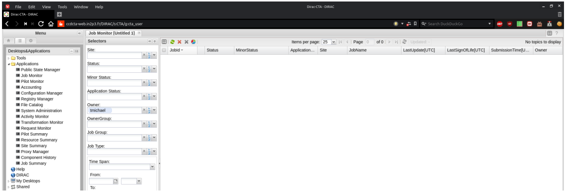
- Protons pose major background
- Event rate about 10^5 times above Photons
- Training Random Forest Classifier
- (virtually identical to energy estimation)

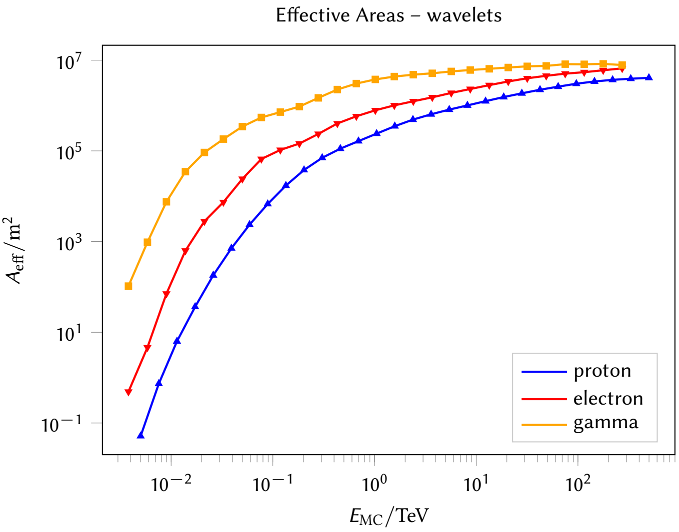


- all this runs with DIRAC on the GRID (including wavelet cleaning)
- lots of sweat, blood and tears to get it going
- Big thanks to Johan and Luisa to take care of the many tickets that I open all the time!

- processing a single setup (5k gamma and electron files, 40k proton files) takes about a week per cleaning mode

- take a look at my **dirac_submit.py** submit-script (handles also some book keeping)





Effective Areas – wavelets
$A_{\text{eff}}/\text{m}^2$
10^{-1}
10^{1}
10^{3}
10^{5}
10^{7}
10^{-2}
10^{-1}
10^{0}
10^{1}
10^{2}
E_{MC}/TeV
proton
electron
gamma

Summary

- data pipeline from reading of the MC files to IRFs and differential sensitivity fully implemented
- (almost) completely written in python
- many things unmentioned (pyhessio, camera calibration, ImPACT, muon reconstruction, ...)

Backup

Full List of Features for RandomForest

- impact_dist - distance between telescope and reconstructed impact position
- sum_signal_evt - total signal on all selected telescopes in the event
- sum_signal_cam - total signal on the current camera
- max_signal_cam - signal of the highest intensity pixel in the camera
- N_LST - number of selected LSTs in the event
- N_MST - number of selected MSTs in the event
- N_SST - number of selected SSTs in the event
- Hillas width
- Hillas length
- Hillas skewness
- Hillas kurtosis
- h_max - reconstructed height of shower maximum
- err_est_pos - error estimator of the reconstructed impact position
- err_est_dir - error estimator of the reconstructed shower direction

Event Weights

- reweighting of MC events to correspond to expected physical flux (e.g. Crab nebula for gammas, CR for protons)
- event-by-event weight that considers the generator spectrum:
- $w(E) = A_{\mathrm{gen}} \times I_{\Theta} \times E^{\gamma} \times I_E \times T_{\mathrm{obs}} / N_{\mathrm{gen}}$
  with:
  - A_{gen}: MC generator Area
  - $I_{\Theta} = 2\pi(1 - \cos\vartheta)$: angular phase space factor for diffuse flux
  - E^{γ}: considers that MC events have been drawn with an E^{-2} spectrum
  - γ: spectral index of the MC generator (here equal 2)
  - $I_E = (E_{\mathrm{max}}^{(1-\gamma)} - E_{\mathrm{min}}^{(1-\gamma)})/(1-\gamma)$: energy phase space factor
  - T_{obs}: assumed observation time
  - N_{gen}: number of generated MC events
- $w(E) \times \Phi(E)$ gives weight for every MC event so that their energy distribution looks like the selected events from the assumed flux Φ