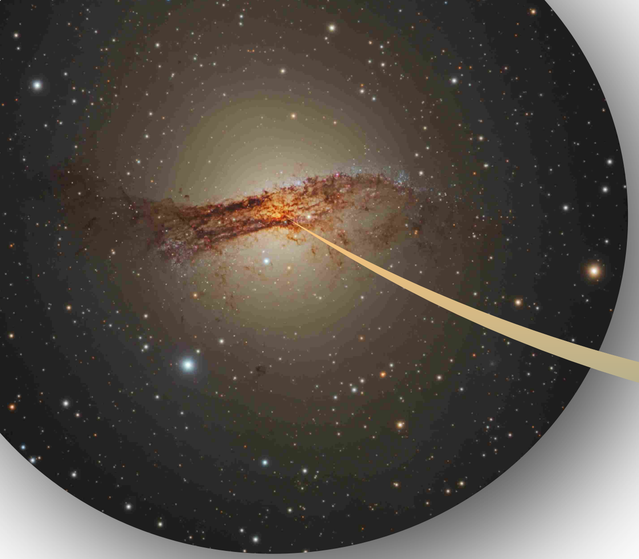
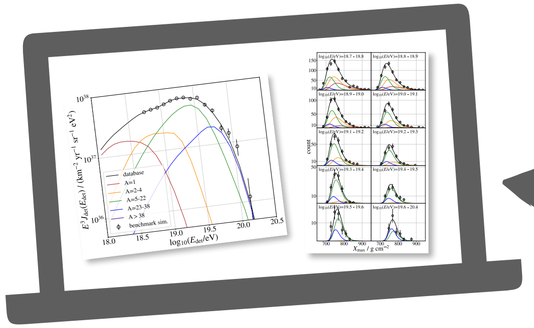


CRPropa Workshop

RWTH AACHEN UNIVERSITY

PIERRE AUGER OBSERVATORY

Inverting UHECR propagation with a normalizing flow



SPONSORED BY THE

Federal Ministry of Education and Research

Teresa Bister, Martin Erdmann, *Josina Schulte*

Overview

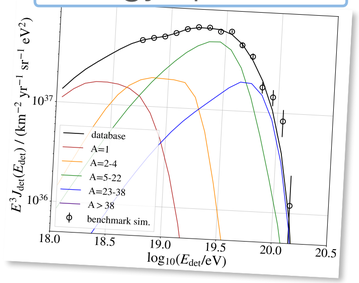
Problem

Explicit UHECR sources currently unknown

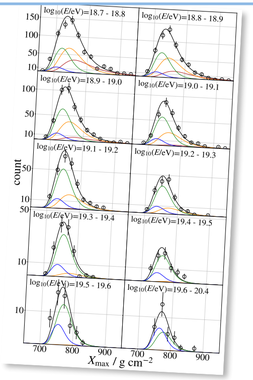
Measurements

Pierre Auger Observatory

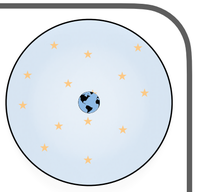
energy spectrum



shower maximum distribution



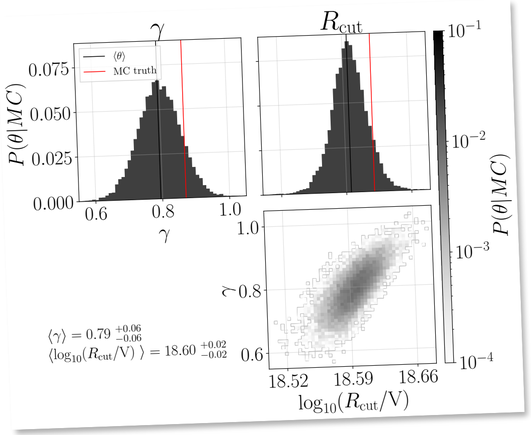
Goal



Constrain **source model parameters** with **posterior distributions**

→ uncertainties, correlations

$$J_{\rm inj}(E) \propto E^{-\gamma} \cdot f_{\rm cut}(E, Z \cdot R_{\rm cut})$$



Combined fit of energy spectrum and X_{max} distributions

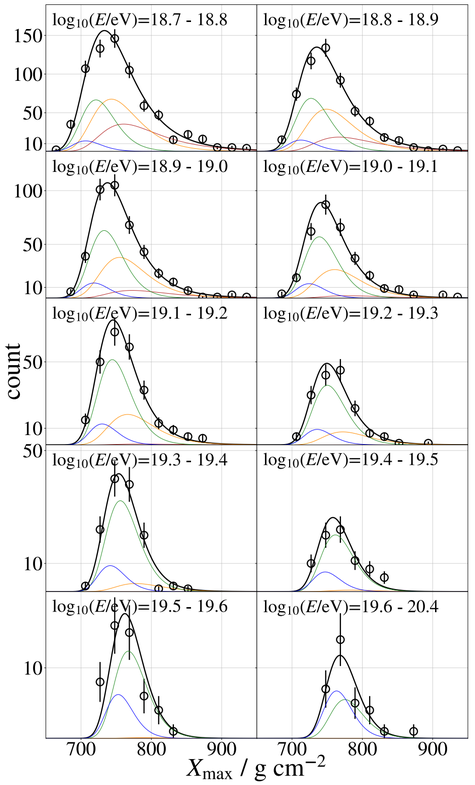
→ *what can we learn about the sources of UHECRs from our measurements?*

A. Aab et al JCAP04(2017)038

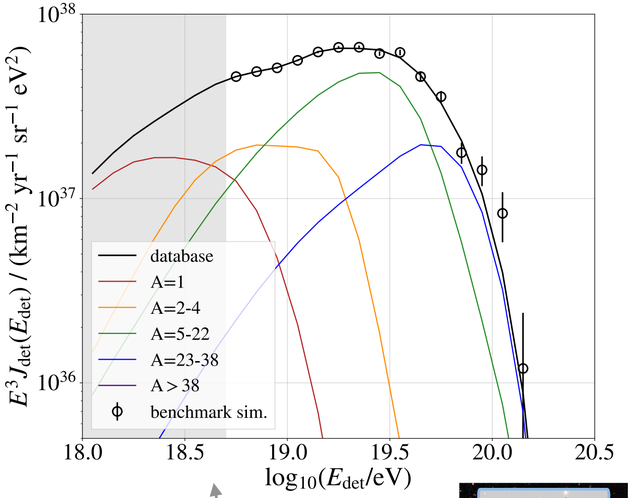
Measurements

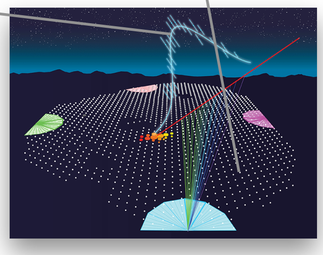
Detected **energy spectrum** & **shower maximum distribution** contain imprints from the source emission

X_{max} distributions



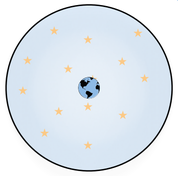
energy spectrum





CRPropa 3

Source model



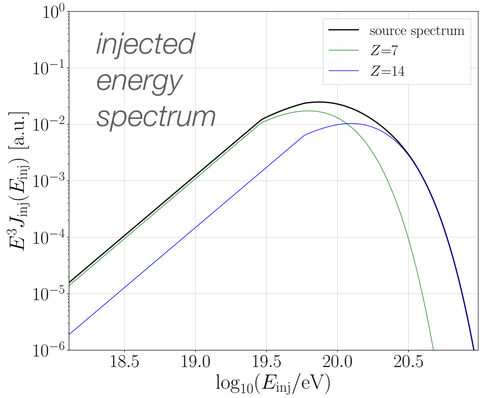
homogeneously distributed sources, each emitting 5 different nuclei with contribution $a(\mathrm{A}_i)$:

power-law energy spectrum with maximum rigidity:

$$J_{\mathrm{inj}}(E) \propto E^{-\gamma} \cdot f_{\mathrm{cut}}(E, Z \cdot R_{\mathrm{cut}}) \qquad E_{\mathrm{max}} = R_{\mathrm{cut}} \cdot Z$$

free parameters of source emission:

- spectral index γ, highest rigidity R_{cut}, relative elemental contribution

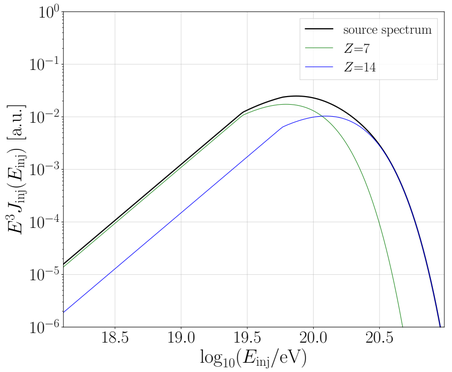


CRPropa database

1d CRPropa 3 simulations with 10^4 CRs per bin

- ⚛ mass bins: 5 (H, He, N, Si, Fe)
- ★ distance bins: $d/\mathrm{Mpc} = 1 - 5670$ ($z_{\mathrm{max}} \approx 2$) in 118 logarithmic bins
- ‣ energy bins: $\log_{10}(E_{\mathrm{inj}}/\mathrm{eV}) = 18 - 21$ with width $\log_{10}(E_{\mathrm{inj}}/\mathrm{eV}) = 0.02$

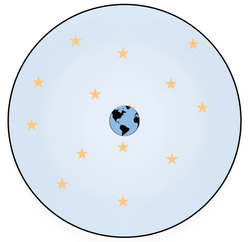
example injected energy spectrum

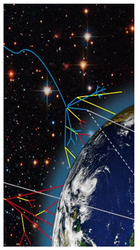


considered interactions

- ‣ nuclear decay
- ‣ electron pair production
- ‣ photopion production
- ‣ photodisintegration

} with CMB & EBL *(Gilmore model)*



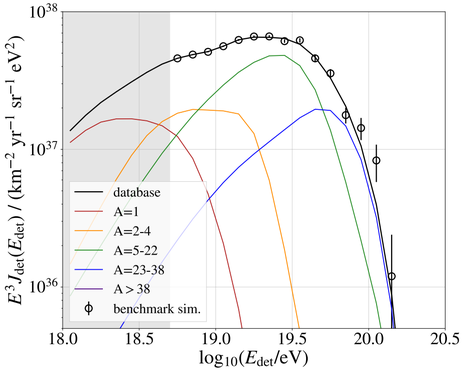


binning at detection

- ⚛ mass bins: 5 $(1, 2-4, 5-22, 23-38, >38)$
- ‣ energy bins: $\log_{10}(E_{\mathrm{det}}/\mathrm{eV}) = 18 - 21$ with width $\log_{10}(E_{\mathrm{det}}/\mathrm{eV}) = 0.02$

→ reweight database according to wanted source emission

example detected energy spectrum



Markov Chain Monte Carlo *(MCMC)*

Bayes

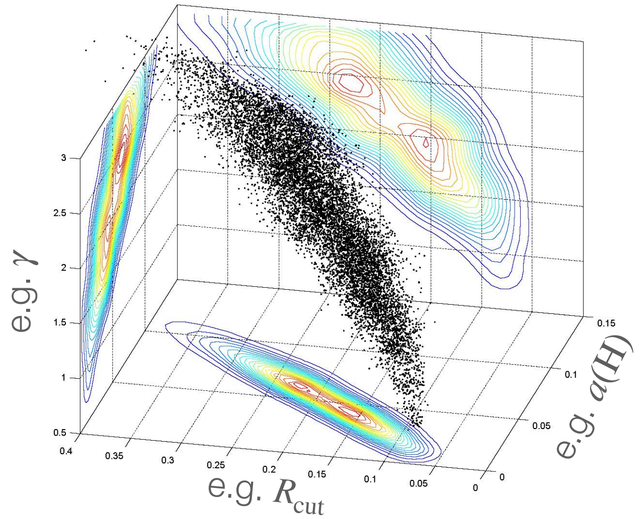
posterior probability of physics parameters $p(\theta \mid y) \propto L(y \mid \theta)\, p(\theta)$

prior

likelihood *(has to be explicitly formulated)*

Markov Chain Monte Carlo (MCMC) method

- Markov chains *(series of samples)* of parameters θ
  - target distribution: $p(\theta \mid y)$
- efficient sampling in multidimensional parameter space
- different sampling algorithms available *(here: sequential Monte Carlo)*
- convergence:
  - enough chains *(→ different starting points)* with enough samples
  - high effective sample size
- computationally expensive



https://www.bgc-jena.mpg.de/bgi/uploads/People/MaartenBraakhekke/correlatedSample.jpg

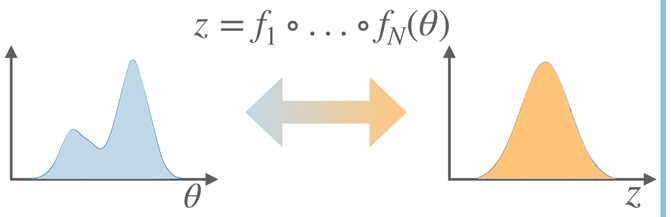
→ *try alternative method*

Normalizing flow for posterior estimation

idea

normalizing flow:

- approximation of complex probability distributions given samples from that distribution
- transformation of a *simple probability distribution* into a more *complex distribution* by a composition of differentiable & invertible mappings



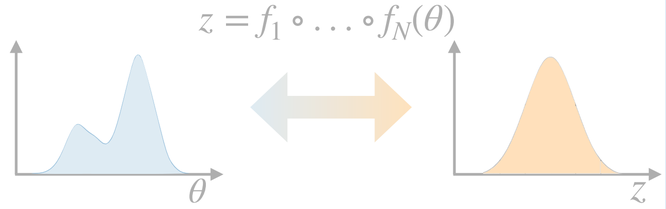
arXiv:1908.09257

Normalizing flow for posterior estimation

idea

normalizing flow:

- approximation of complex probability distributions given samples from that distribution
- transformation of a *simple probability distribution* into a more *complex distribution* by a composition of differentiable & invertible mappings

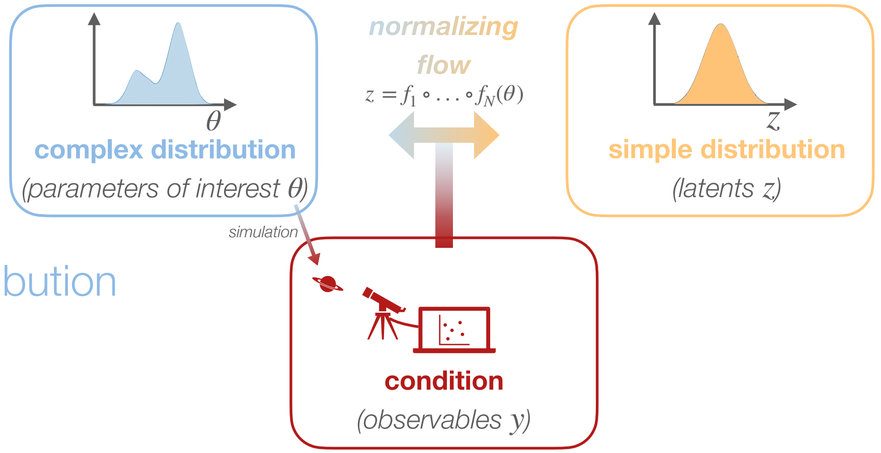


arXiv:1908.09257

approach

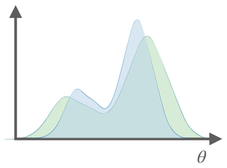
conditional normalizing flow: reconstruct conditional probability distribution, like posterior distribution $p(\theta \mid y)$

- mapping between physics parameters θ & latents z *under the* ***condition*** of observables y
- training process: use many samples θ, $y(\theta)$ from known forward model
- inverse pass: posterior distributions $p(\theta \mid y)$ for specific observed y



suitable loss function: Kullback-Leibler divergence

- minimizes difference: true underlying posterior distribution & cNF posterior distribution
- ensures convergence to correct posterior distributions
- enforce normal distribution for latents z



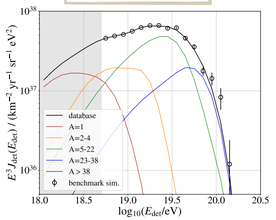
Methodical differences: MCMC vs cNF

both methods give **posterior distributions given an observation**

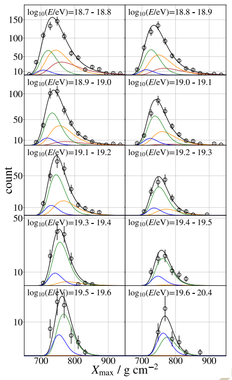
MCMC

- **likelihood** approach
  - engineered according to the experimental statistics of the observables measured at Earth
- maximize likelihood
  - minimize difference between predicted & measured **observables**
- convergence of sampled space to posteriors in limit of infinite samples

Poisson

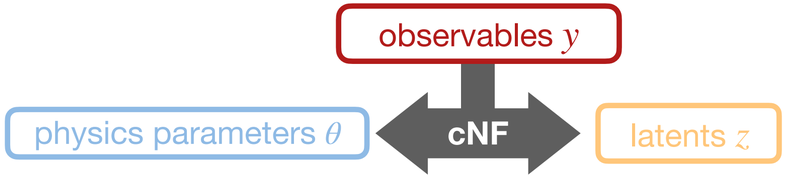


multinomial



cNF

- likelihood-free/simulation-based inference using **loss** function
  - directly minimize difference between true **posterior** distributions of the source parameters & the posterior distributions of the network
- agreement of observables only implicitly achieved by agreement of source parameter posteriors



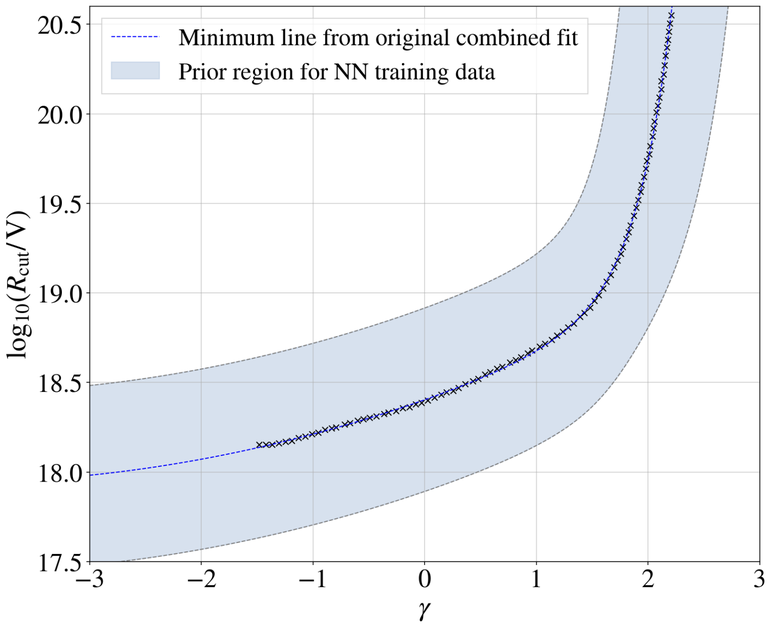
Training data for the cNF for the combined fit

Source

Prior *(on source parameters)* for **training data set** *(size: 1.000.000)*

- spectral parameters γ, R_{cut} region

A. Aab et al JCAP04(2017)038



- 5 elements (H, He, N, Si, Fe) with side conditon: sum = 1

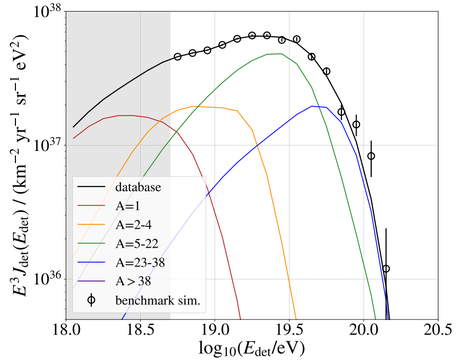
reweighting using CRPropa 3 database

Observation

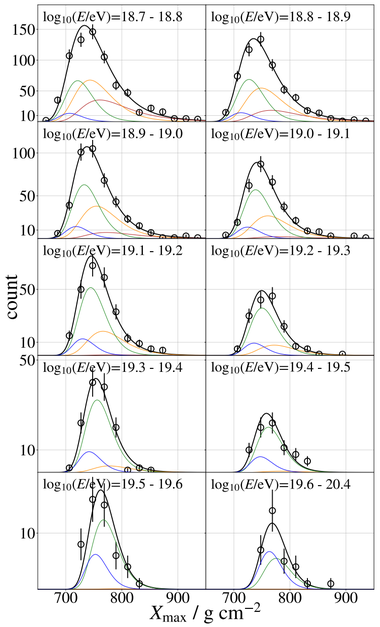
Energy spectrum & **shower maximum distribution** as observation

Example observation

17 bins



10 x 24 bins

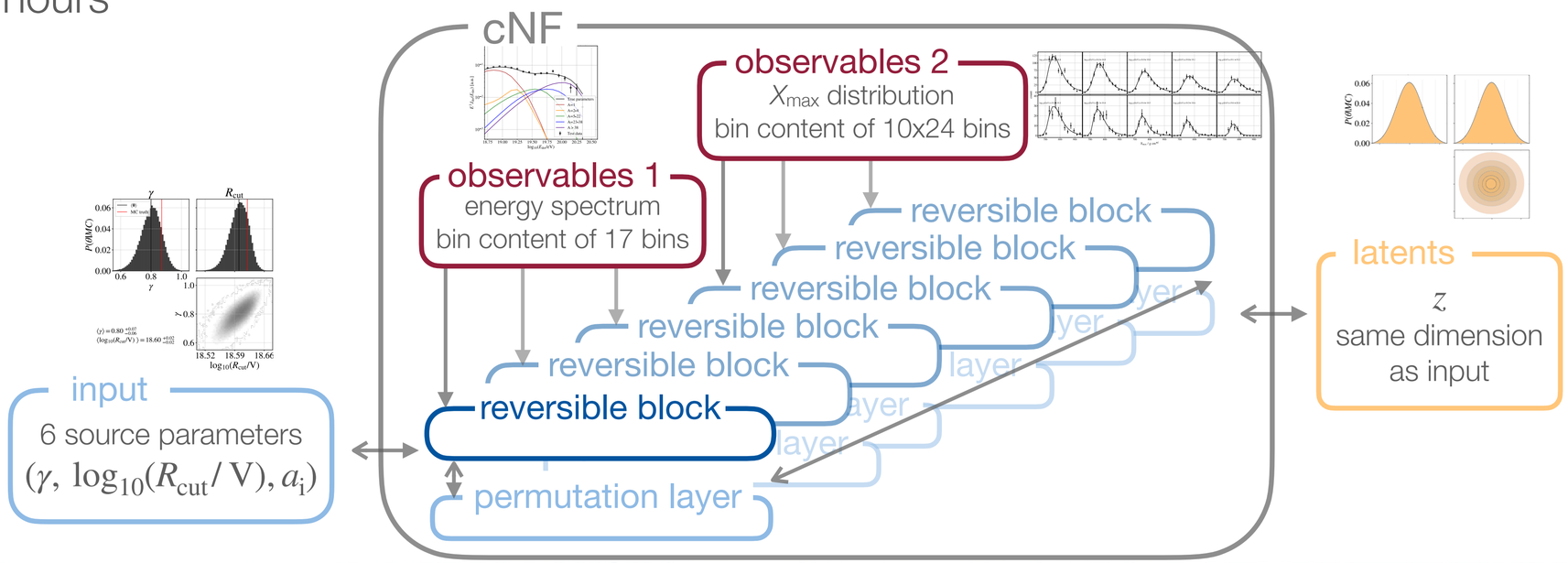


During training:

- fluctuate every bin according to Poisson statistics with event numbers:
  ~ SD statistics for energy spectrum
  ~ FD statistics for depth of shower maximum distributions
- normalize X_{max} distributions in every energy bin to remove energy spectrum information
- rescale energy spectrum with E^3 to flatten spectrum & improve reconstruction quality

Network structure

- **6 reversible blocks**
- reversible block gets *either* energy spectrum *or* depth of shower maximum distributions
  - **energy spectrum**: bin content of 17 energy bins above $10^{18.7}\,\mathrm{eV}$
  - **shower maximum distribution**: bin content of 10 energy x 24 X_{max} bins
- permutation layers randomly permute the input & support *more stable results*
- 10^6 training samples
- training time ~13 hours

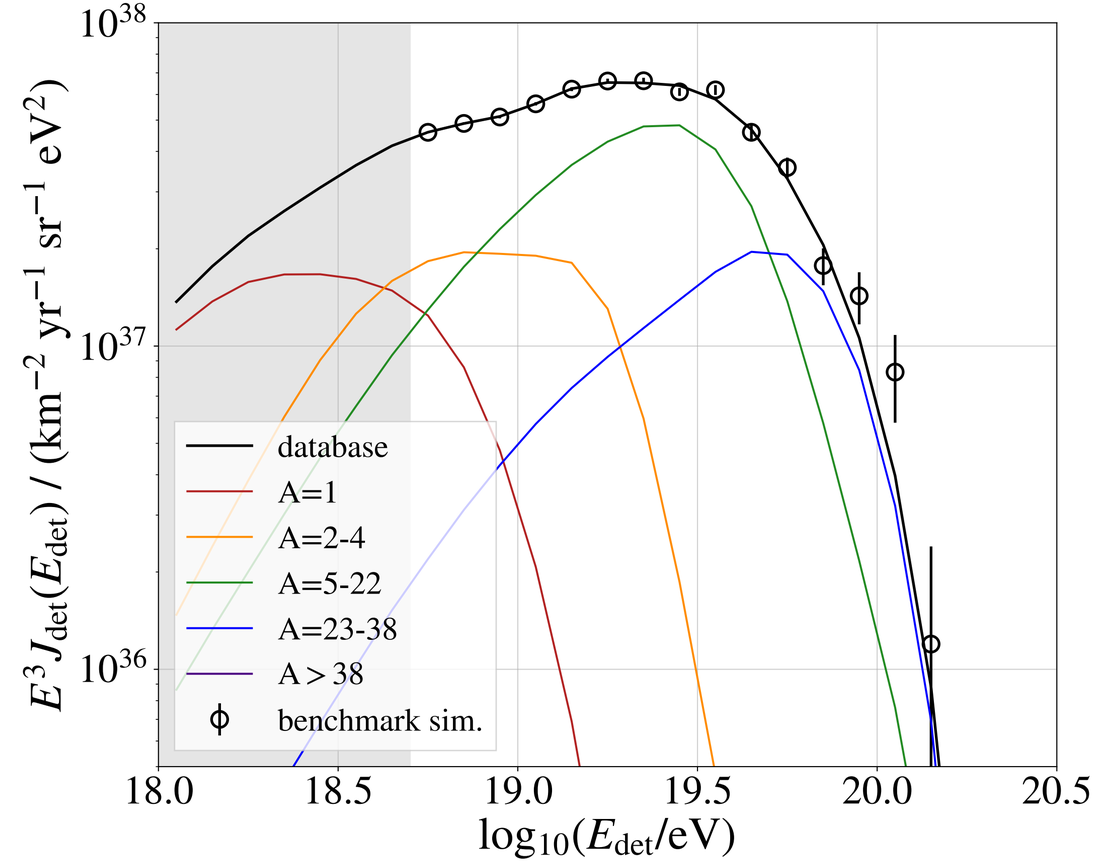


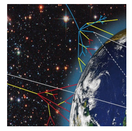
Benchmark simulation: observables

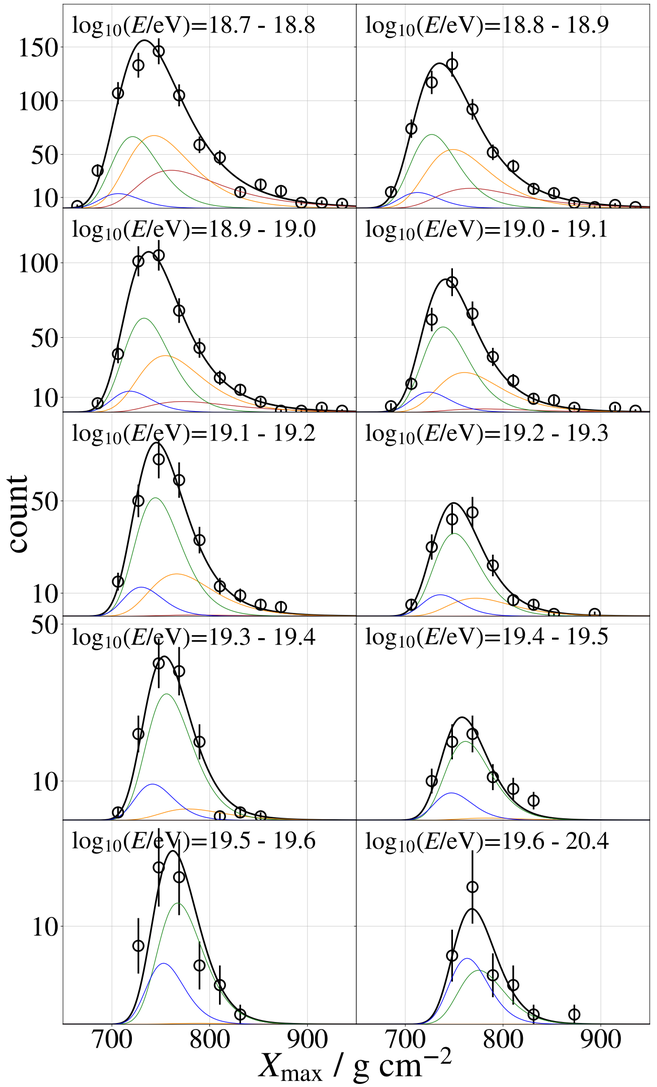
benchmark simulation using source parameters from previous analysis on data from the PAO

$$\gamma = 0.87,\ \log_{10}(R_{\rm cut}/\,{\rm V}) = 18.62, a({\rm N}) = 0.88, a({\rm Si}) = 0.12$$

energy spectrum





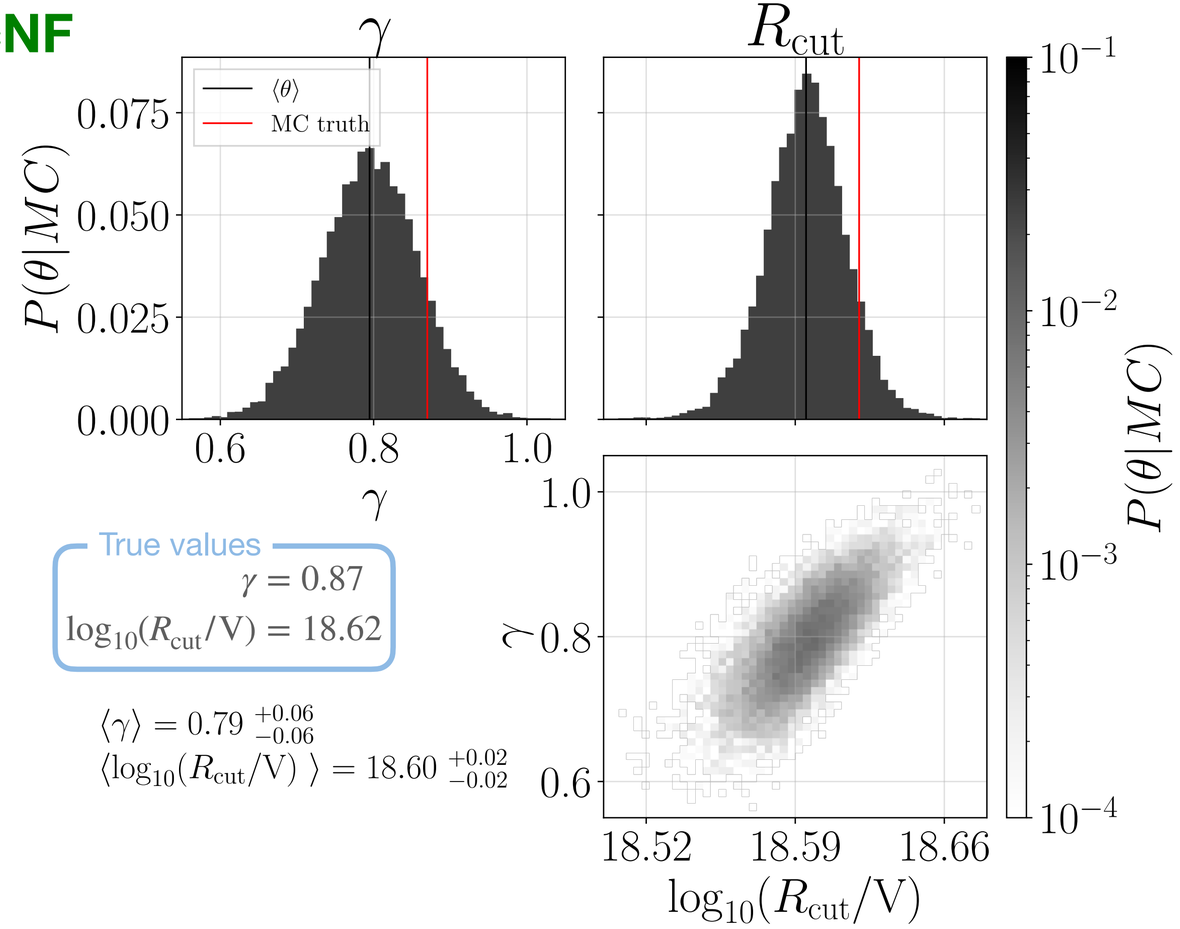


depth of shower maximum distributions

→ evaluate with new method *(cNF)* and compare to MCMC

Benchmark simulation: posterior distributions

cNF



True values

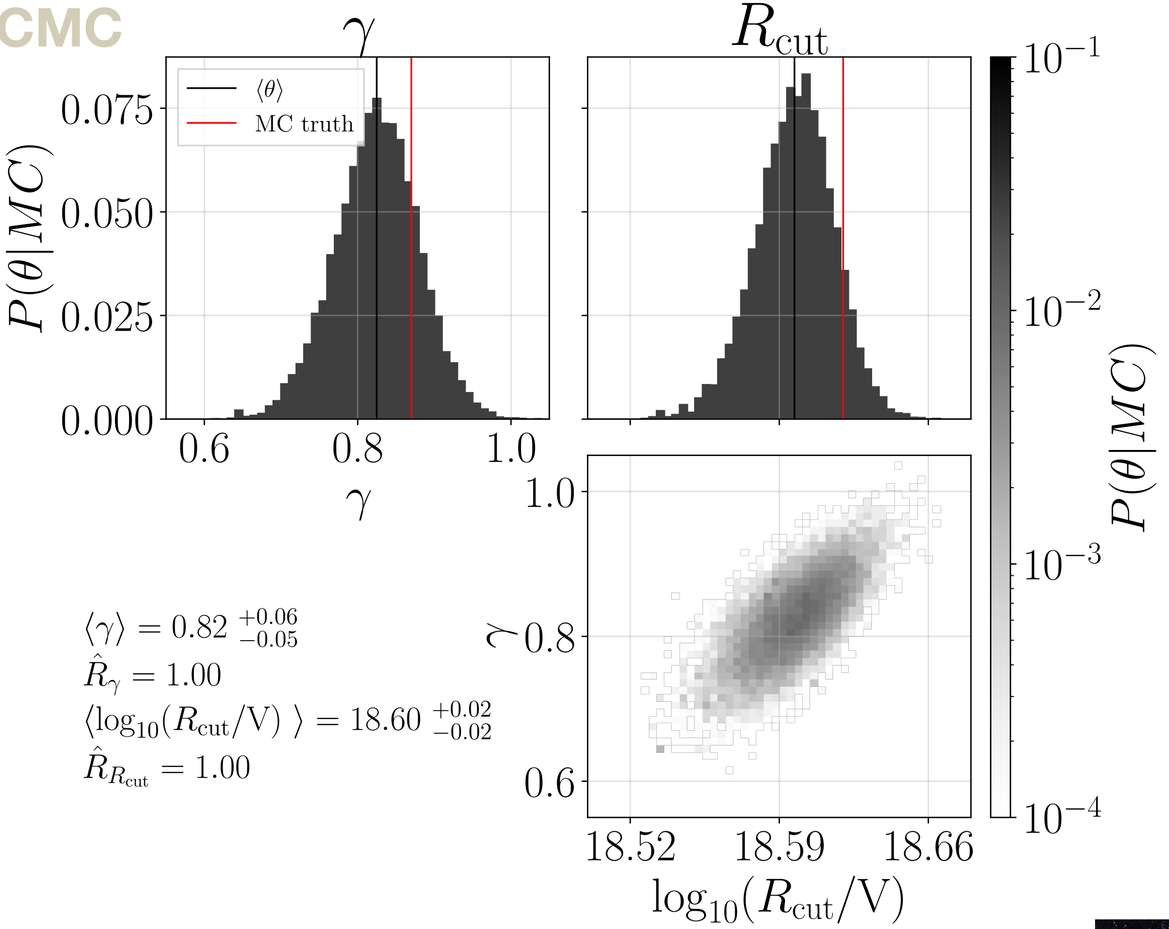
$\gamma = 0.87$

$\log_{10}(R_{\rm cut}/{\rm V}) = 18.62$

$\langle\gamma\rangle = 0.79\ ^{+0.06}_{-0.06}$

$\langle\log_{10}(R_{\rm cut}/{\rm V})\ \rangle = 18.60\ ^{+0.02}_{-0.02}$

MCMC



$\langle\gamma\rangle = 0.82\ ^{+0.06}_{-0.05}$

$\hat{R}_{\gamma} = 1.00$

$\langle\log_{10}(R_{\rm cut}/{\rm V})\ \rangle = 18.60\ ^{+0.02}_{-0.02}$

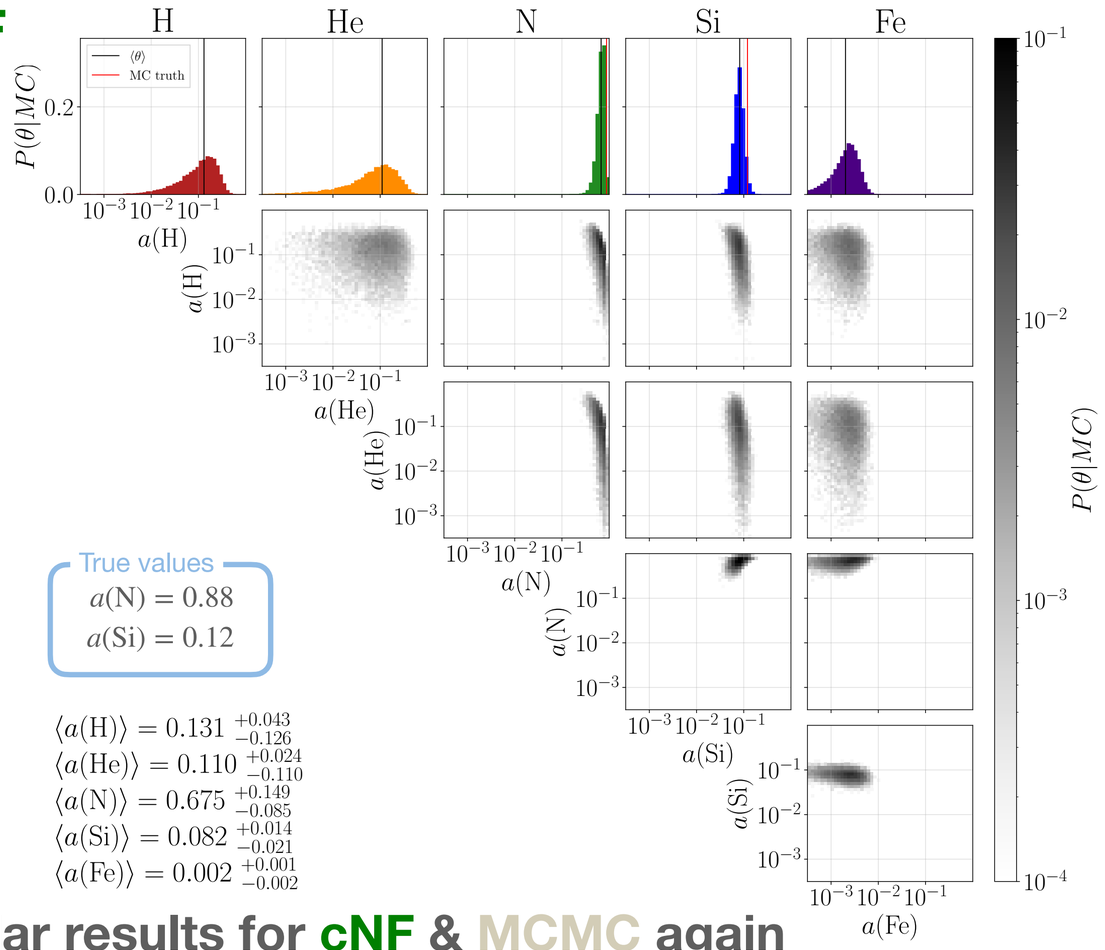
$\hat{R}_{R_{\rm cut}} = 1.00$

similar posterior distributions

- symmetric, true value within 1σ, positive correlation between $\gamma, \log_{10}(R_{\rm cut}/{\rm V})$

Benchmark simulation: posterior distributions

cNF



True values

$a(\mathrm{N}) = 0.88$

$a(\mathrm{Si}) = 0.12$

$\langle a(\mathrm{H})\rangle = 0.131\ ^{+0.043}_{-0.126}$

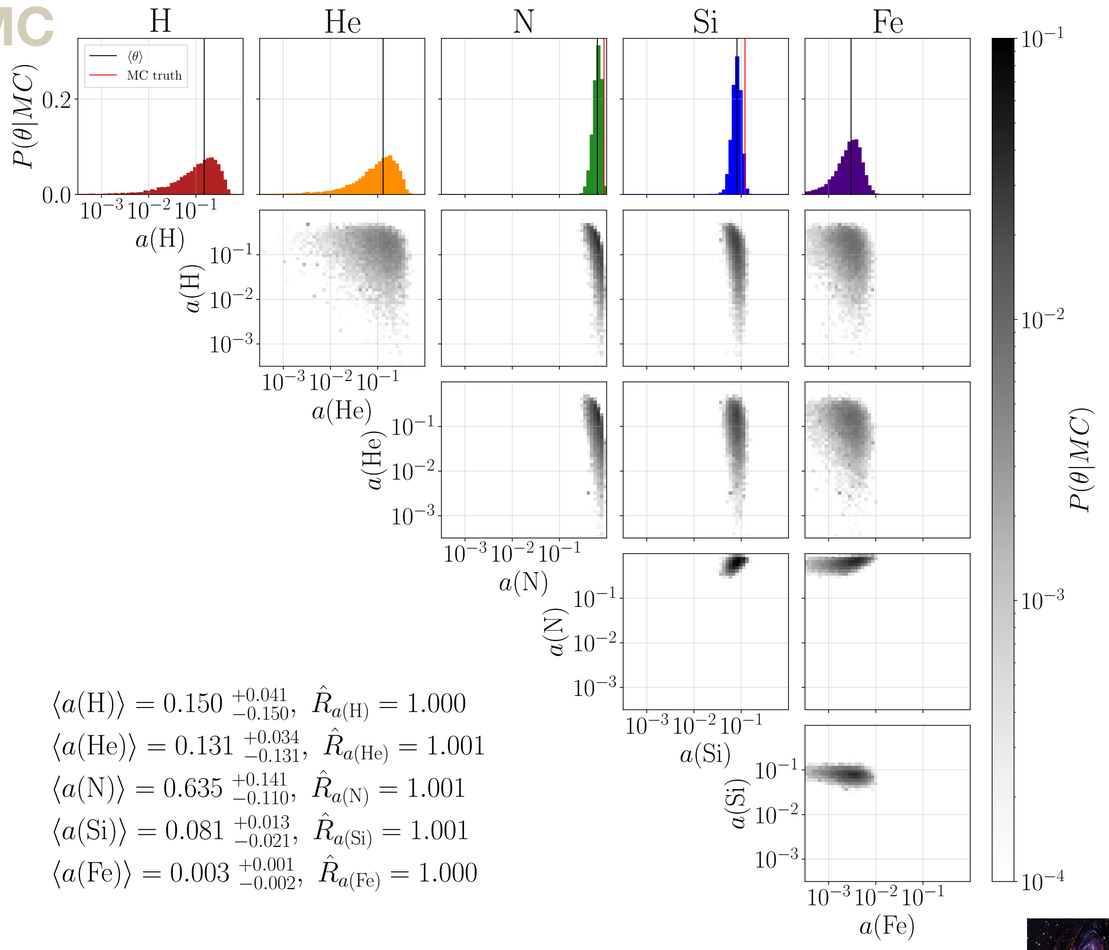
$\langle a(\mathrm{He})\rangle = 0.110\ ^{+0.024}_{-0.110}$

$\langle a(\mathrm{N})\rangle = 0.675\ ^{+0.149}_{-0.085}$

$\langle a(\mathrm{Si})\rangle = 0.082\ ^{+0.014}_{-0.021}$

$\langle a(\mathrm{Fe})\rangle = 0.002\ ^{+0.001}_{-0.002}$

MCMC



$\langle a(\mathrm{H})\rangle = 0.150\ ^{+0.041}_{-0.150},\ \hat{R}_{a(\mathrm{H})} = 1.000$

$\langle a(\mathrm{He})\rangle = 0.131\ ^{+0.034}_{-0.131},\ \hat{R}_{a(\mathrm{He})} = 1.001$

$\langle a(\mathrm{N})\rangle = 0.635\ ^{+0.141}_{-0.110},\ \hat{R}_{a(\mathrm{N})} = 1.001$

$\langle a(\mathrm{Si})\rangle = 0.081\ ^{+0.013}_{-0.021},\ \hat{R}_{a(\mathrm{Si})} = 1.001$

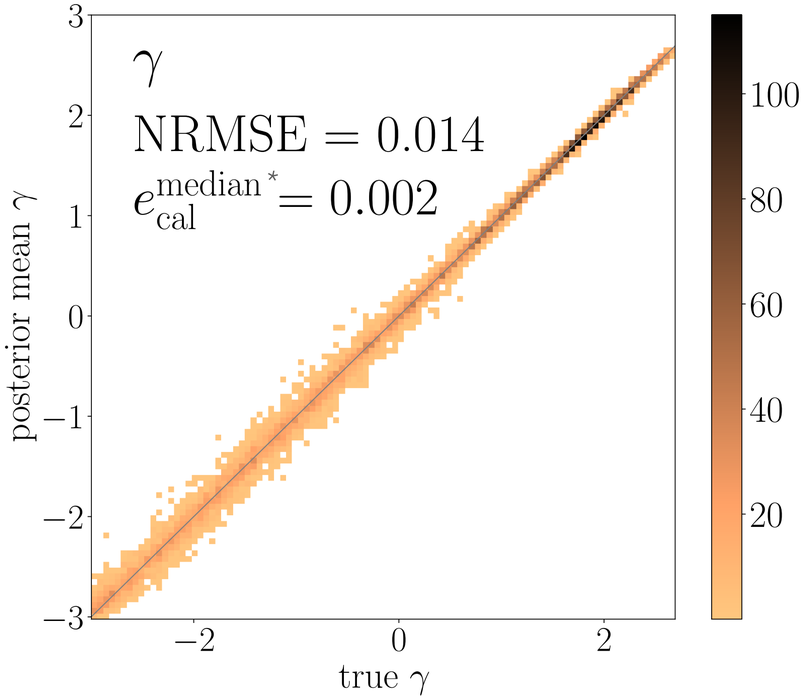
$\langle a(\mathrm{Fe})\rangle = 0.003\ ^{+0.001}_{-0.002},\ \hat{R}_{a(\mathrm{Fe})} = 1.000$

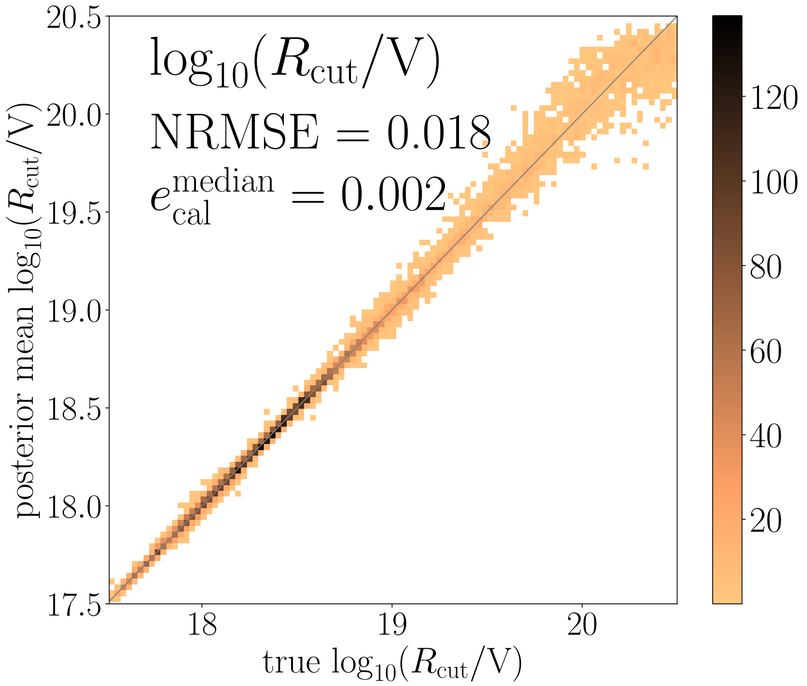
similar results for cNF & MCMC again

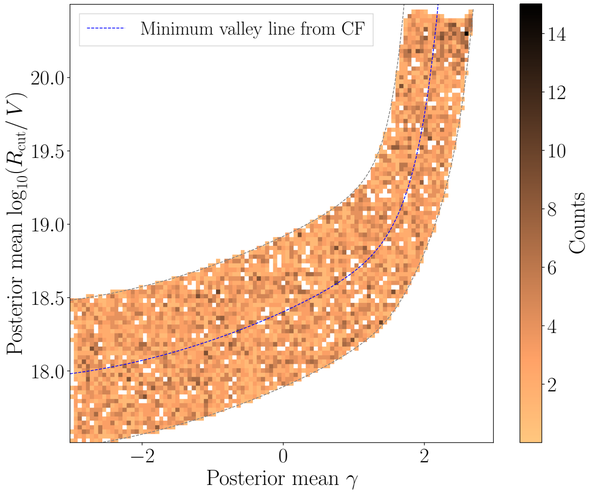
- largest contribution by nitrogen & silicon, at most tiny iron fraction
- broad posteriors for lighter fractions including 0
  - almost no light elements present in observations $E > 10^{18.7}\,\mathrm{eV}$ for rigidity cut-off $\log_{10}(R_{\mathrm{cut}}/\mathrm{V}) = 18.62$ at sources

Stability of cNF results

Evaluate performance on test dataset *(size 10.000)*





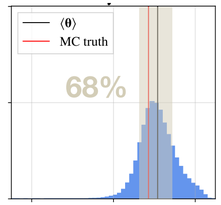


**median over absolute values in range of confidence intervals* $(0.01, 0.99)$ *in* 0.01 *steps*

→ **Good reconstruction quality** of cNF for many test datasets

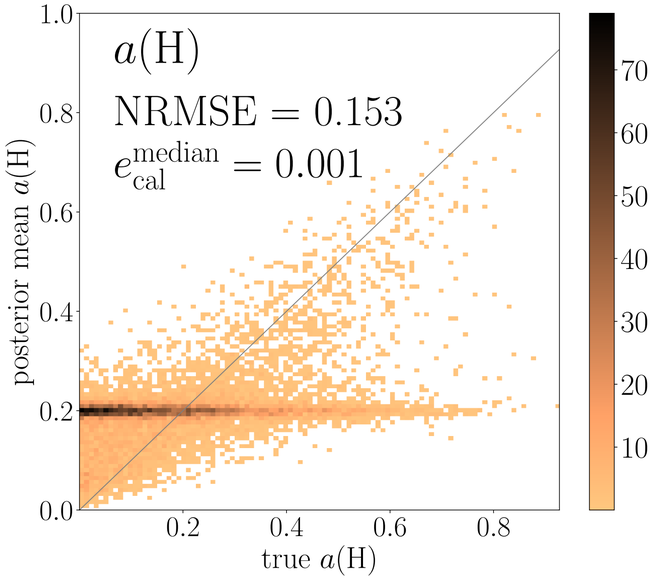
Calibration error: $e_{\mathrm{cal}} = q_{\mathrm{inliers}} - q$ for estimation of correctness of the widths of the posterior distributions

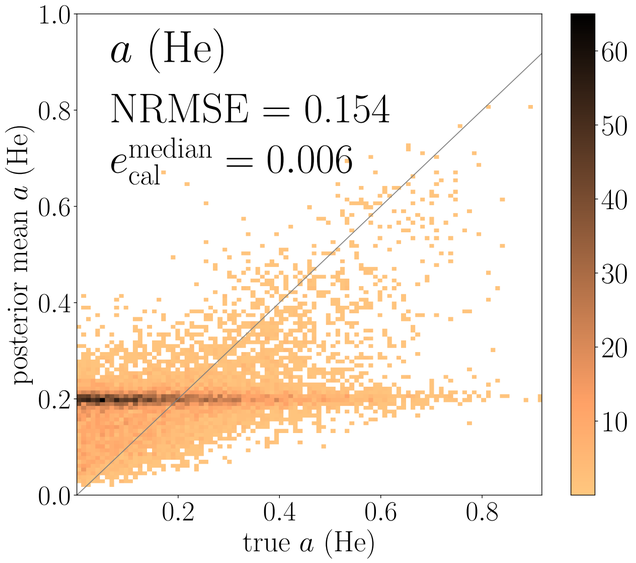
- confidence interval q & fraction of observations $q_{\mathrm{inliers}} = \dfrac{N_{\mathrm{inliers}}}{N}$ with true value in q-confidence interval

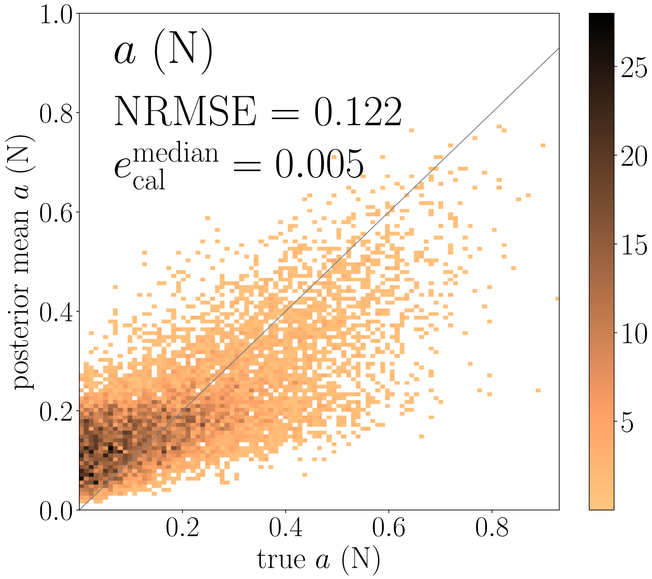


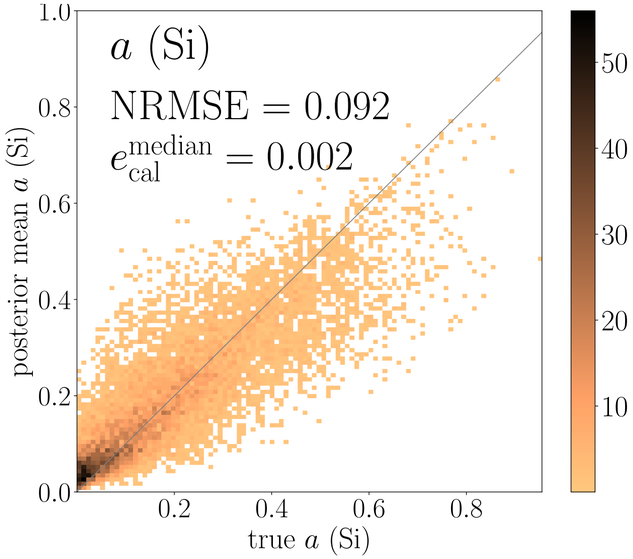
✓ $e_{\mathrm{cal}}^{\mathrm{median}}$ close to 0 → suitable widths of posterior distributions → **appropriate uncertainty estimation** possible

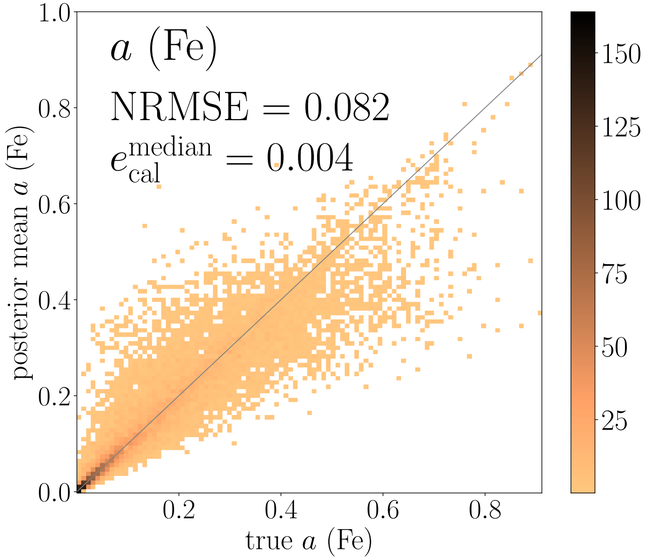
Stability of cNF results: elemental fractions











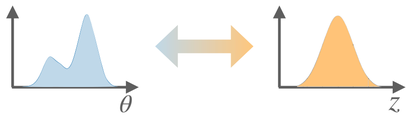
- **Reconstruction quality** of elemental fractions **depends on mass**
  - light elements cannot be reconstructed, prediction converges to average value for five elements
  - heavier fractions can be better constrained

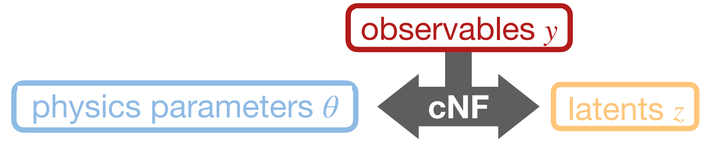
✓ small calibration error: suitably large uncertainty for unrecoverable light fractions

Conclusion

conditional normalizing flow

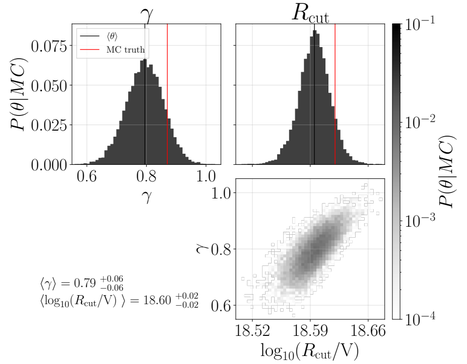
- method based on transformation of distributions



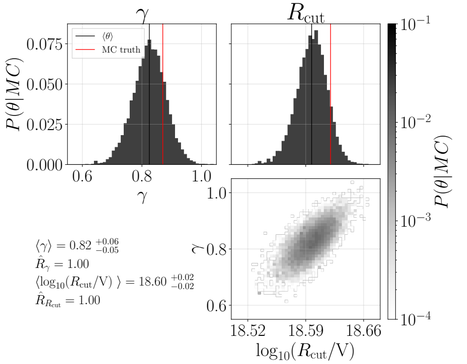


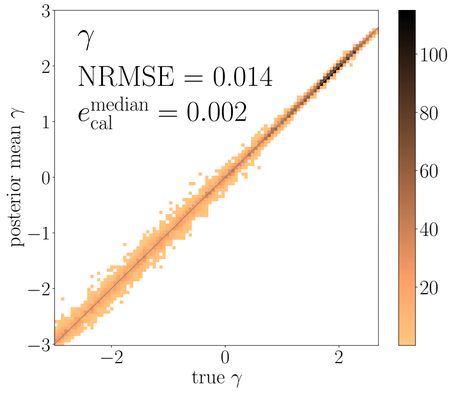
- enables **creation of posterior distributions** *(like MCMC)*
  - very similar posterior distribution to MCMC
    although techniques work inherently different
- provides **fast evaluation** of many test datasets
  - rigorous testing
  - possible extension to more/different observables
- learns mapping between observables & source parameters

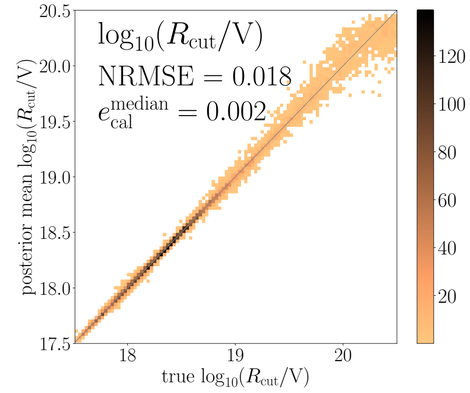
cNF



MCMC







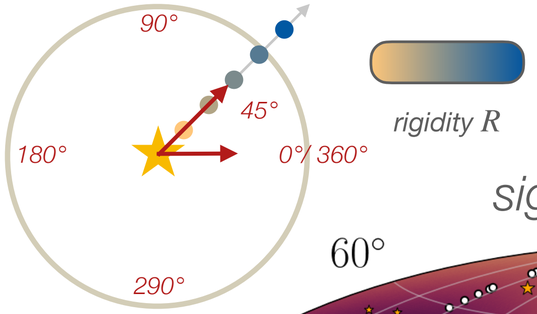
possible alternative applications

- NF as generator → create training data for classification/…
- NF as fast simulator → use learned mapping in bigger models

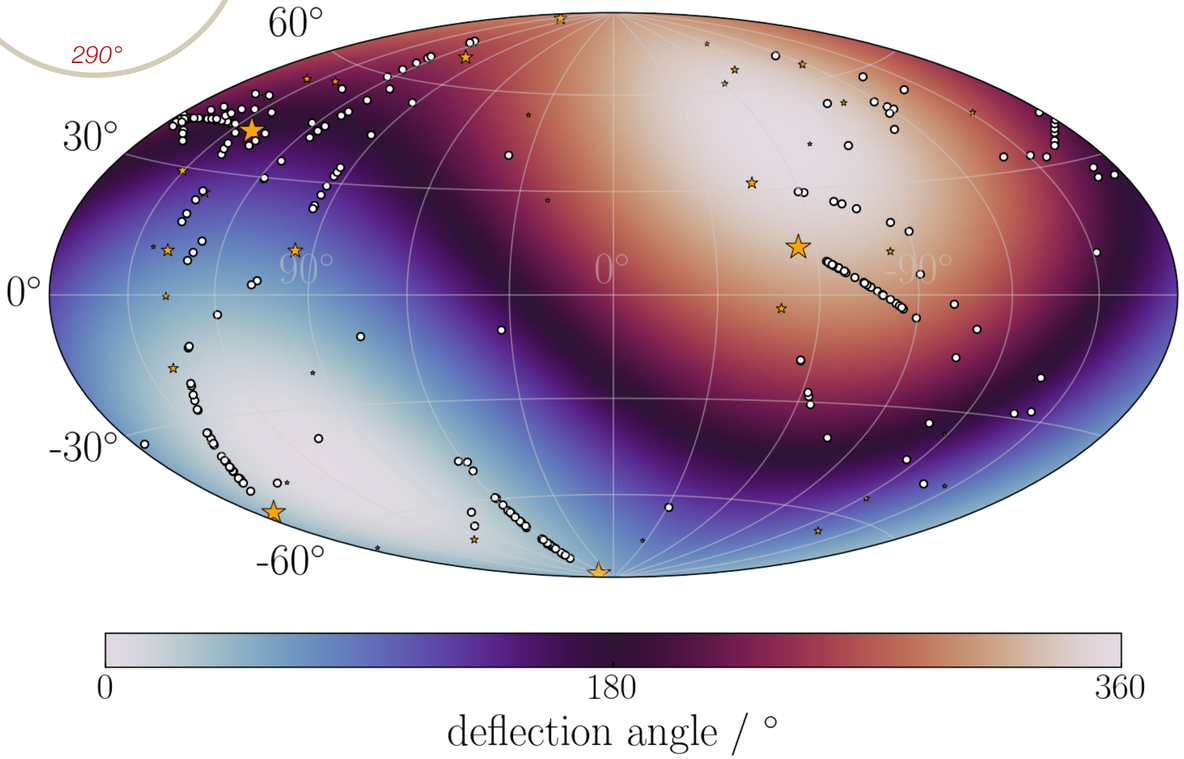
Outlook: arrival directions as observables

use **arrival directions** as observables & reconstruct parameters of „*GMF*“ = simple deflection map

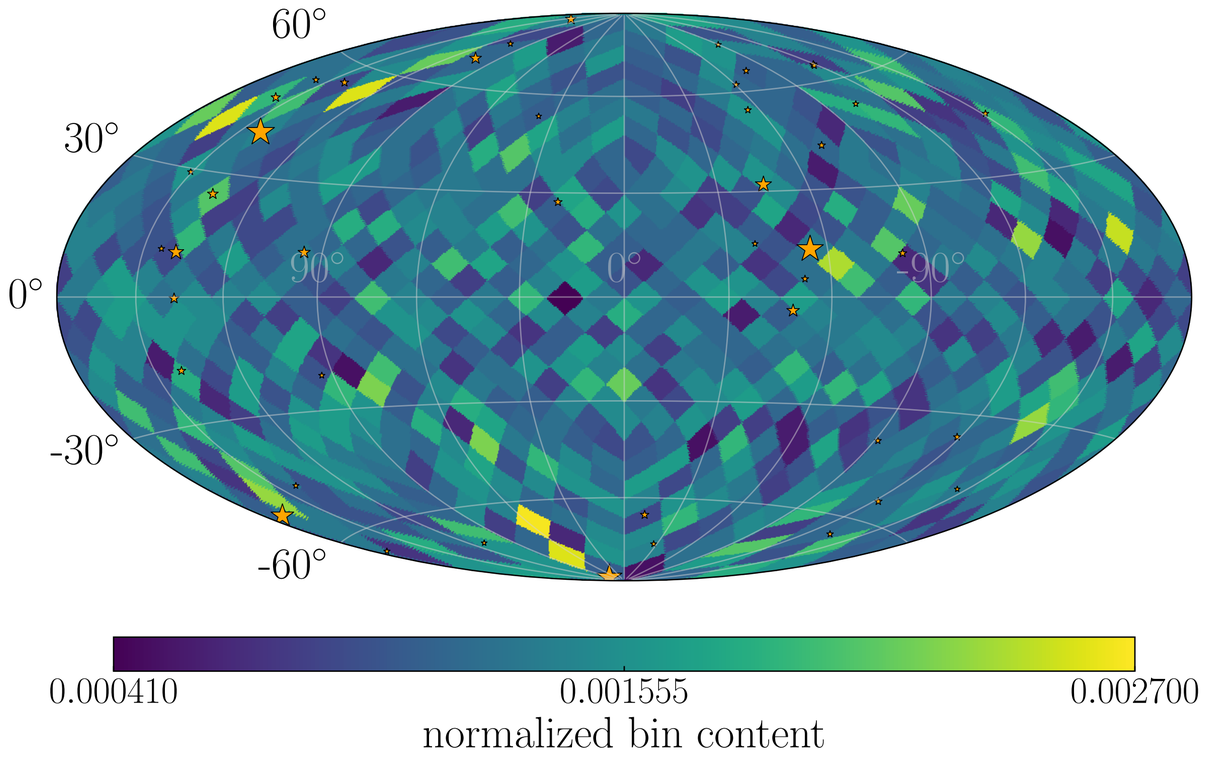
deflection map: dipole → value at different positions describe coherent deflection direction



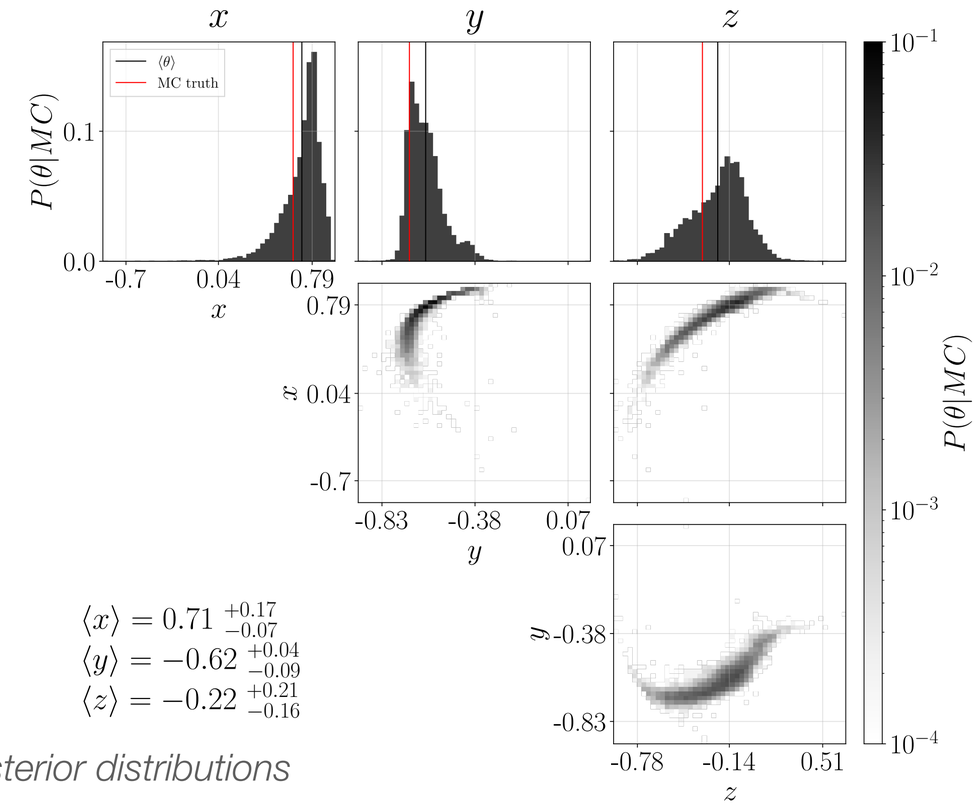
signal contribution



total = 2% signal + 98% background

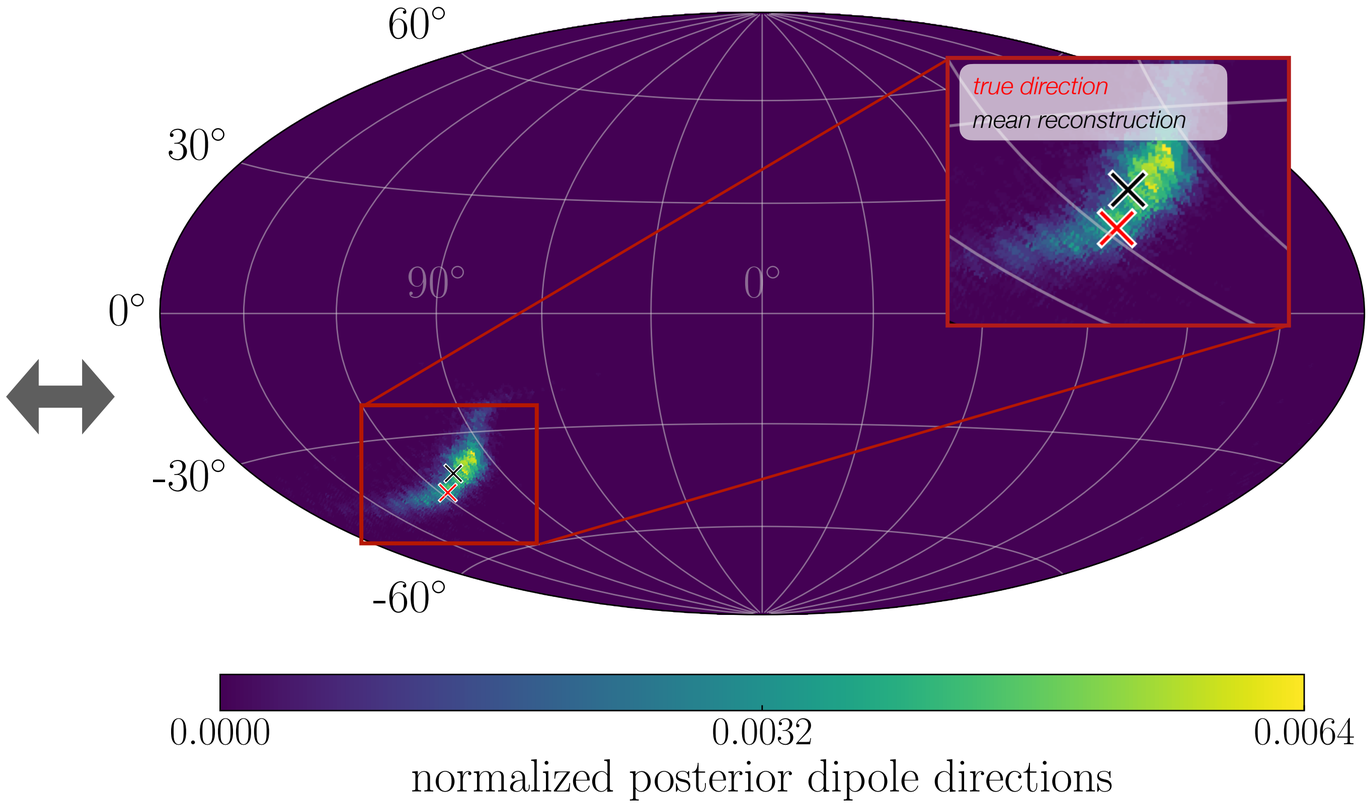


Outlook: arrival directions as observables



$\langle x \rangle = 0.71\ ^{+0.17}_{-0.07}$
$\langle y \rangle = -0.62\ ^{+0.04}_{-0.09}$
$\langle z \rangle = -0.22\ ^{+0.21}_{-0.16}$

posterior distributions



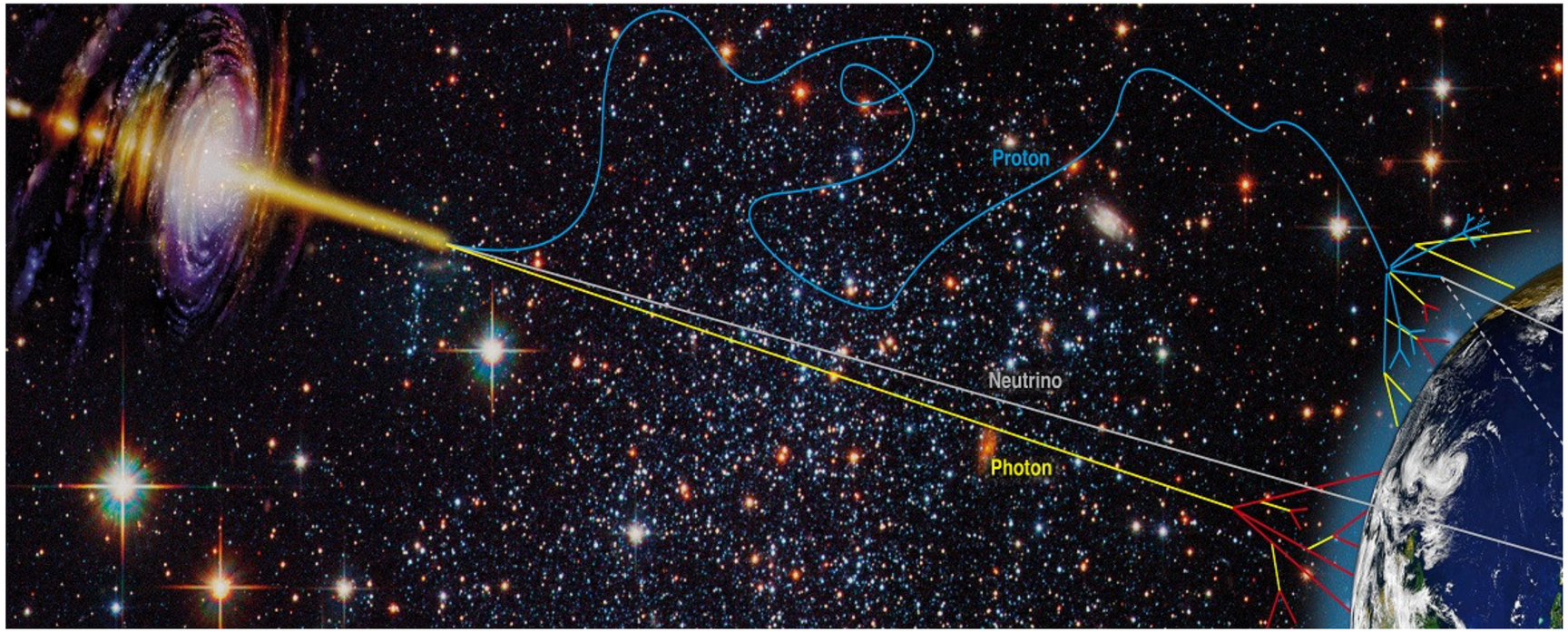
✓ dipole direction can be reconstructed with cNF despite quite large background contamination

future plans:

- more detailed deflection maps *(higher multipole orders)*, different source scenarios
- add other observables (energy, X_{max}) *(similar to Teresa)*

Thank you for your attention!

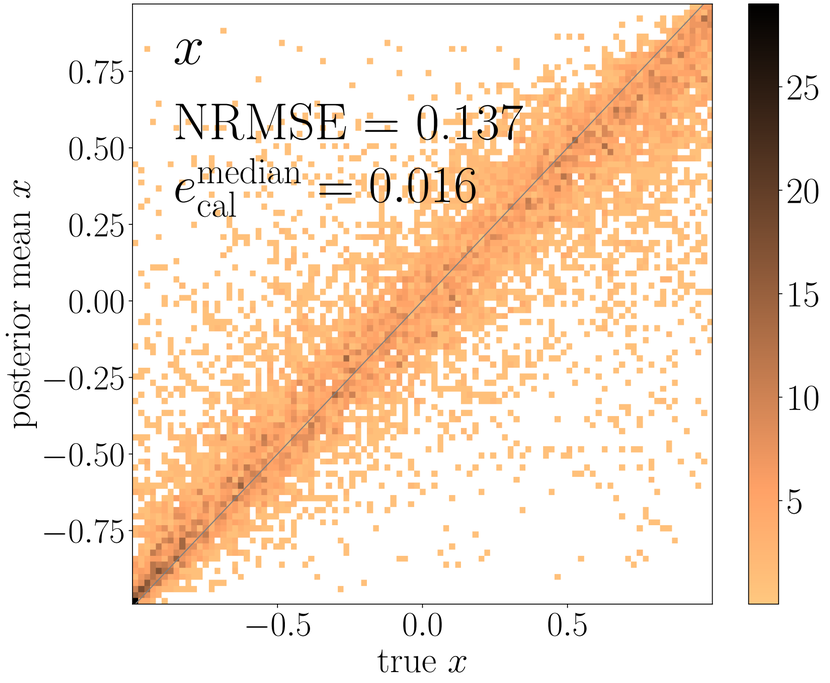
Backup

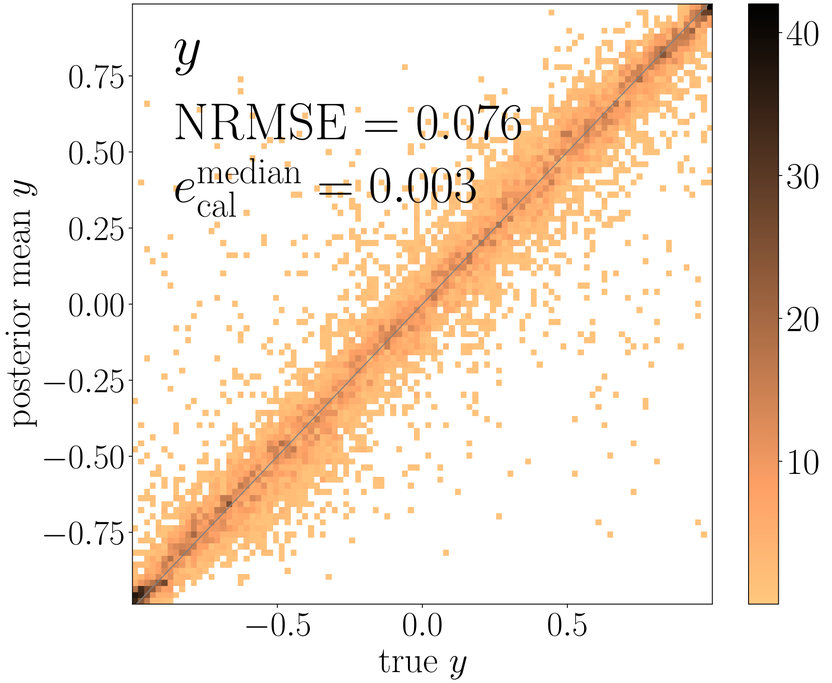


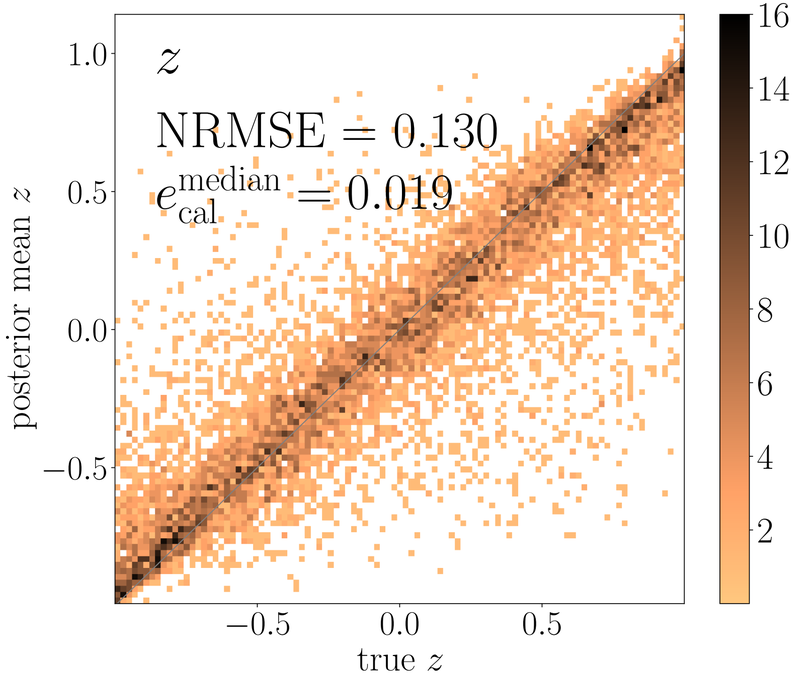
https://www.uni-potsdam.de/typo3temp/assets/_processed_/b/c/csm_kosm_Strahlung2_b7bead59a7.jpg

Outlook: arrival directions as observables

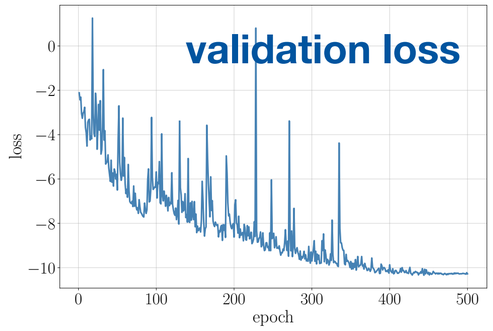
network setup: 5 blocks, internal size 256 (# NN parameters: 3,296,030)

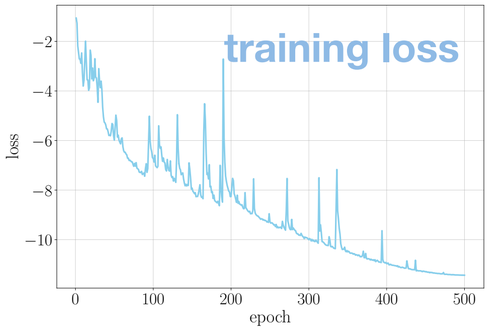






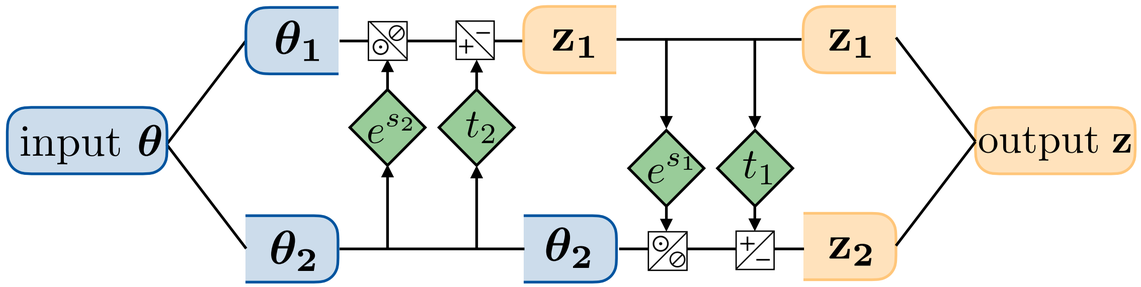
→ very good calibration error, mean reconstruction improvable





Underlying method: reversible blocks

reversible block



- split input vector θ in 2 halves: $\theta = [\theta_1, \theta_2]$
- transform by affine function:
  use mappings s_i & t_i, $i \in \{1,2\}$
  (arbitrarily complicated, not invertible)
  output $\mathbf{z} = [\mathbf{z_1}, \mathbf{z_2}]$

$$\mathbf{z}_1 = \theta_1 \odot \exp(s_2(\theta_2)) + t_2(\theta_2)$$
$$\mathbf{z}_2 = \theta_2 \odot \exp(s_1(\mathbf{z}_1)) + t_1(\mathbf{z}_1)$$

- Invertibility

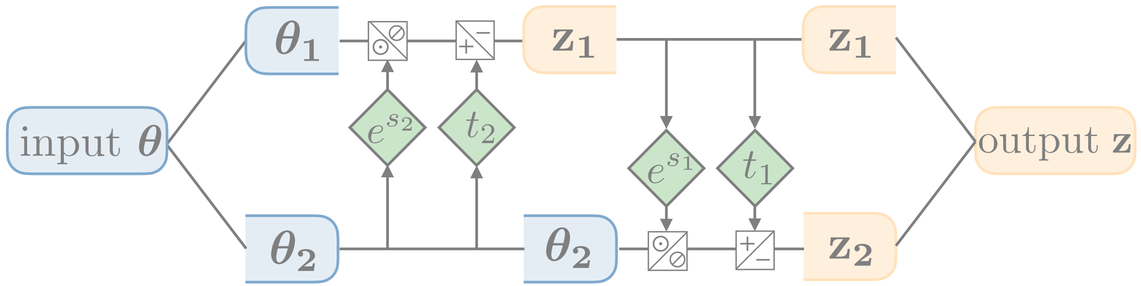
element-wise multiplication

$$\theta_2 = (\mathbf{z}_2 - t_1(\mathbf{z}_1)) \odot \exp(-s_1(\mathbf{z}_1))$$
$$\theta_1 = (\mathbf{z}_1 - t_2(\theta_2)) \odot \exp(-s_2(\theta_2))$$

based on *real-valued non-volume preserving* (**Real NVP**) architecture (*arXiv:1605.08803*)

Underlying method: reversible blocks

reversible block



- split input vector θ in 2 halves: $\theta = [\theta_1, \theta_2]$
- transform by affine function:
  use mappings s_i & t_i $i \in \{1,2\}$
  (arbitrarily complicated, not invertible)
  output $\mathbf{z} = [\mathbf{z_1}, \mathbf{z_2}]$

$$\mathbf{z}_1 = \theta_1 \odot \exp(s_2(\theta_2)) + t_2(\theta_2)$$
$$\mathbf{z}_2 = \theta_2 \odot \exp(s_1(\mathbf{z}_1)) + t_1(\mathbf{z}_1)$$

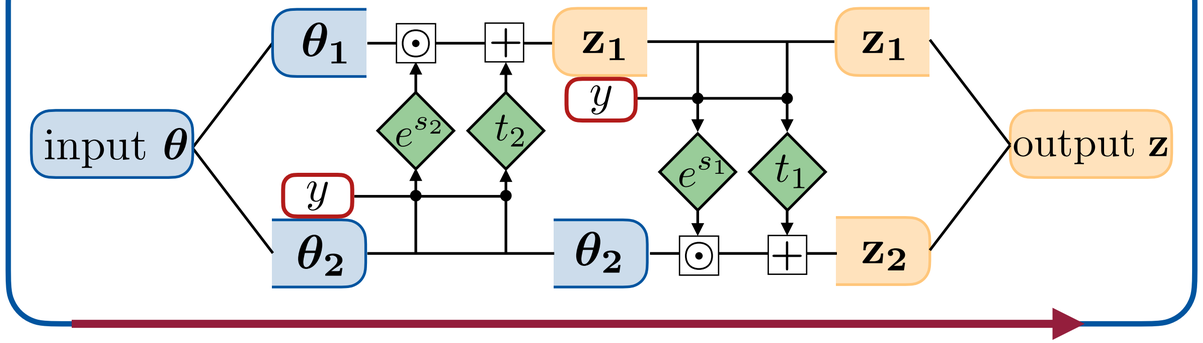
- Invertibility (*element-wise multiplication* $\odot$)

$$\theta_2 = (\mathbf{z}_2 - t_1(\mathbf{z}_1)) \odot \exp(-s_1(\mathbf{z}_1))$$
$$\theta_1 = (\mathbf{z}_1 - t_2(\theta_2)) \odot \exp(-s_2(\theta_2))$$

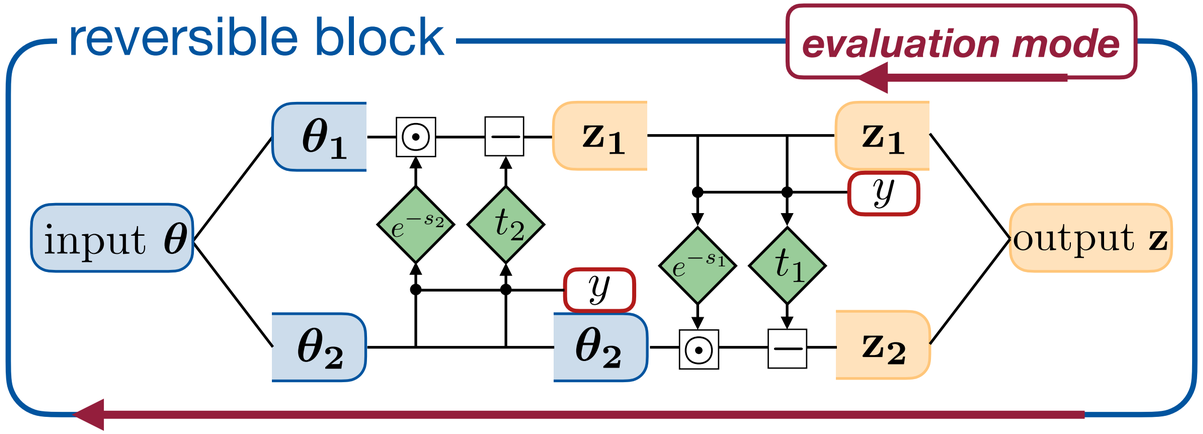
based on *real-valued non-volume preserving* (**Real NVP**) architecture (*arXiv:1605.08803*)

Using the observation y as *condition*

reversible block — ***training mode***

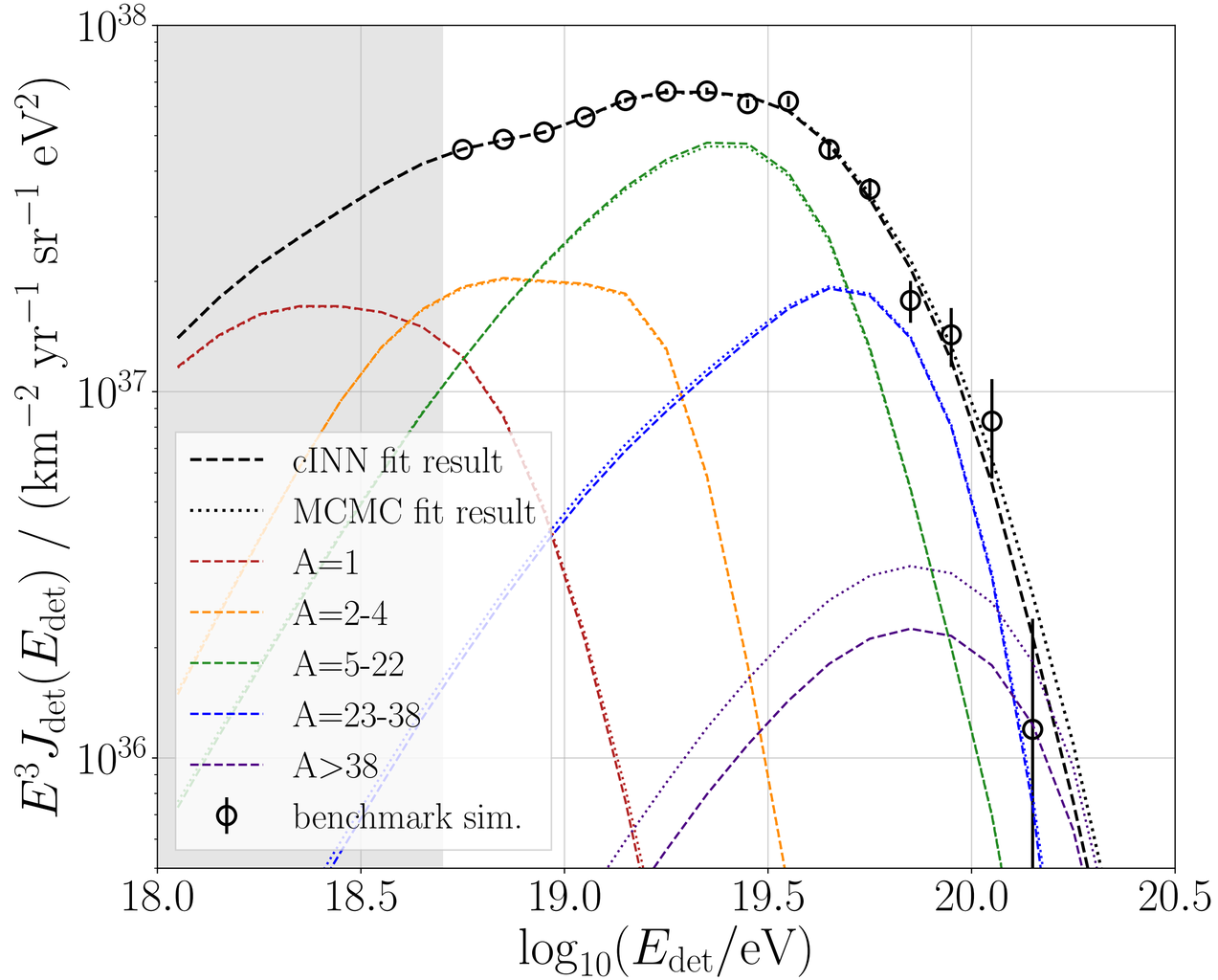


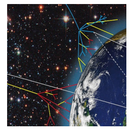
reversible block — ***evaluation mode***



Benchmark simulation: energy spectrum fit

Prediction *(using mean values of both methods)* for energy spectrum





- ✓ agreement of prediction for energy spectrum with benchmark simulation for both methods

Deviance = -2 · (likelihood of fitted model - likelihood of saturated model)

cNF:

$$D^{\mathrm{cNF}} = D_E + D_{X_{\mathrm{max}}} = 14.3 + 124.7 = 139.0$$

MCMC:

$$D^{\mathrm{MCMC}} = D_E + D_{X_{\mathrm{max}}} = 14.7 + 123.7 = 138.4$$

→ small deviance achieved with both methods

Network & training setup

Network settings

Reversible blocks: 6

GLOW subnetwork structure

fully connected transformation as subnet constructor (width 256)

Training settings

Initial learning rate: 10^{-3}

Learning rate decay: cosine

Final learning rate: 10^{-5}

Adam optimizer betas: $(0.9, 0.999)$

Batch size: 1000

Epochs: 1000

Training data

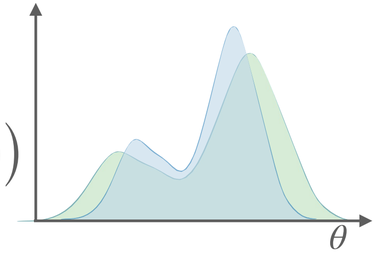
- **energy spectrum**: binned between *18.7 & 20.4* with width *0.1* in $\log_{10}(E/\,\mathrm{eV})$
- **shower maximum distribution**: binned in energy between *18.7 & 19.6* with width *0.1 + 1 high energy bin,* binned in X_{max} between 550 & 1050 g/cm^2 with width 20 g/cm^2 *(normalize energy bins to remove spectrum information)*

subnetworks

- **fully connected** transformation with internal size (here: 256)
- 3 layers with **ReLU activation** of **linear transformation**
- last layer: linear transformation only

Suitable loss function

Kullback-Leibler divergence: true posterior $p(\theta\,|\,\mathbf{y})$ & cNF posterior $p_\phi(\theta\,|\,\mathbf{y})$



$$\mathscr{L} = \mathbb{KL}\big(p(\theta\,|\,\mathbf{y}) \parallel p_\phi(\theta\,|\,\mathbf{y})\big) = \mathbb{E}_{\theta\sim p(\theta|\mathbf{y})}\big(\log p(\theta\,|\,\mathbf{y}) - \log p_\phi(\theta\,|\,\mathbf{y})\big) = \text{const.} + \mathbb{E}_{\theta\sim p(\theta|\mathbf{y})}\big(-\log p_\phi(\theta\,|\,\mathbf{y})\big)$$

probability conservation: $p_\phi(\theta\,|\,\mathbf{y})\,\mathrm{d}\theta = p(z)\,\mathrm{d}z$

$$\mathscr{L} = \mathbb{E}_{\theta\sim p(\theta|\mathbf{y})}\big(-\log p_\phi(\theta\,|\,\mathbf{y})\big) = \mathbb{E}_{\theta\sim p(\theta|\mathbf{y})}\Big(-\log\big(p(\mathbf{z})\cdot|\det\Big(\frac{\partial\,\mathbf{z}}{\partial\theta}\Big)|\big)\Big) = \mathbb{E}_{\theta\sim p(\theta|\mathbf{y})}\Big(-\log\big(p(\mathbf{z})\big) - \log\big(|\det\Big(\frac{\partial\,\mathbf{z}}{\partial\theta}\Big)|\big)\Big)$$

Using Gaussian distribution for latents z & Jacobian of reversible blocks = triangular matrix, where $\mathbf{z} = f(\theta)$

$$\mathscr{L} = \frac{1}{m}\sum_{i=1}^{m}\Big(\frac{1}{2}\|f(\theta_i)\|^2 - \sum_{l=1}^{2}\sum_{j} s_{l,j}\Big)$$ with m training data sets

Jacobian

- example: $f_1(\theta) = \begin{cases}\mathbf{z}_1 = \theta_1 \odot \exp(s_2(\theta_2)) + t_2(\theta_2)\\ \theta_2 = \theta_2\end{cases}$, $\dfrac{\partial f_1(\theta)}{\partial\theta} = \begin{pmatrix}\frac{\partial\mathbf{z}_1}{\partial\theta_1} & \frac{\partial\mathbf{z}_1}{\partial\theta_2}\\ \frac{\partial\theta_2}{\partial\theta_1} & \frac{\partial\theta_2}{\partial\theta_2}\end{pmatrix} = \begin{pmatrix}\mathrm{diag}(\exp\big(s_2(\theta_2)\big)) & \frac{\partial\mathbf{z}_1}{\partial\theta_2}\\ 0 & \mathbb{I}\end{pmatrix}$
- $|\det\dfrac{\partial\mathbf{z}}{\partial\theta}| = \dfrac{\partial f_1(\theta)}{\partial\theta}\cdot\dfrac{\partial f_2(\tilde{\theta}=f_1(\theta))}{\partial\tilde{\theta}} = \prod_j \exp(s_2(\theta_2))_j\cdot\exp(s_1(\mathbf{z}_1))_j = \exp\Big(\sum_j s_2(\theta_2)_j + \sum_j s_1(\mathbf{z}_1)_j\Big)$

(equivalently for $\mathbf{z}_2$ with $f_2(\theta)$)

Likelihood for MCMC

Total likelihood: $\mathscr{L} = \mathscr{L}_E \cdot \mathscr{L}_{X_{\max}}$

energy spectrum: $\mathscr{L}_E = \prod_e \frac{(p^e)^{k^e}}{k^e!} \exp(-p^e)$

shower maximum distributions: $\mathscr{L}_{X_{\max}} = \prod_{\tilde{e}} k^{\tilde{e}}! \prod_x \frac{(p^{\tilde{e},x})^{k^{\tilde{e},x}}}{k^{\tilde{e},x}!}$

with

- the predicted spectrum p calculated from the simulation database, the event counts of the measurement/benchmark simulation k, each in energy bins e
- the predicted number of events $p^{\tilde{e},x}$ and the measured number of events $k^{\tilde{e},x}$, each in energy bin $\tilde{e}$ and $X_{\max}$ bin x

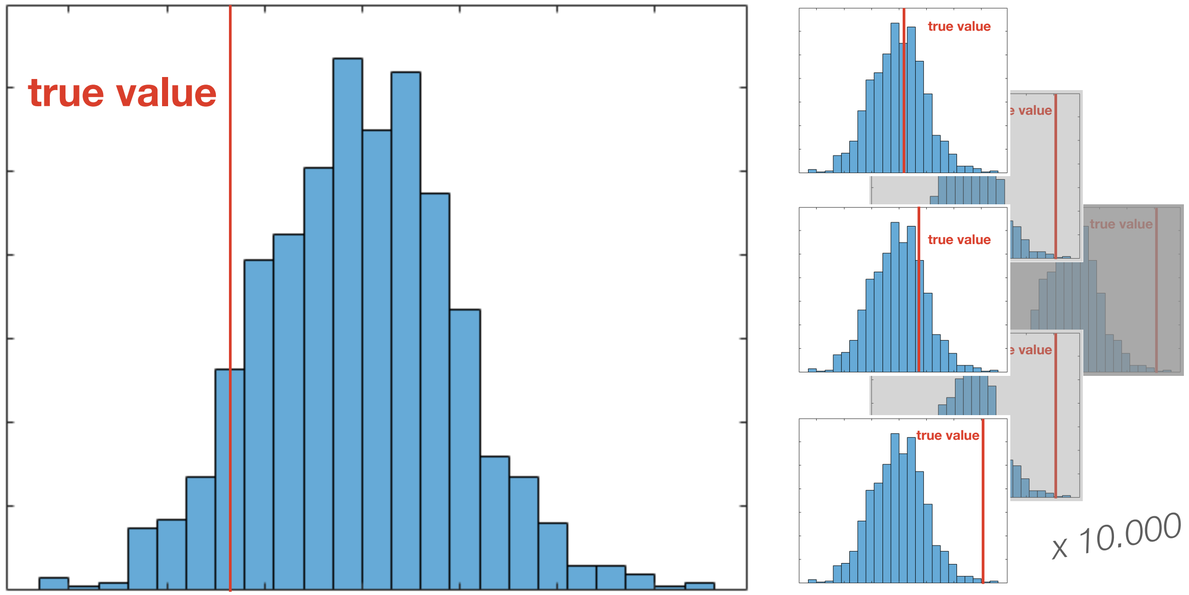
MCMC sampling: 3 chains with 5,000 steps each

Calibration error

For estimation of correctness of the widths of the posterior distributions

- $e_{\text{cal}} = q_{\text{inliers}} - q$
  - confidence interval q & fraction of observations $q_{\text{inliers}} = \frac{N_{\text{inliers}}}{N}$ with true value in q-confidence interval
- $e_{\text{cal}}^{\text{med}}$: median over range of confidence intervals $(0.01, 0.99)$ in 0.01 steps

For each free parameter:
Evaluate posteriors for all test data sets (10.000)



Count for how many test sets the true value lies in the q-confidence interval