OmniFold

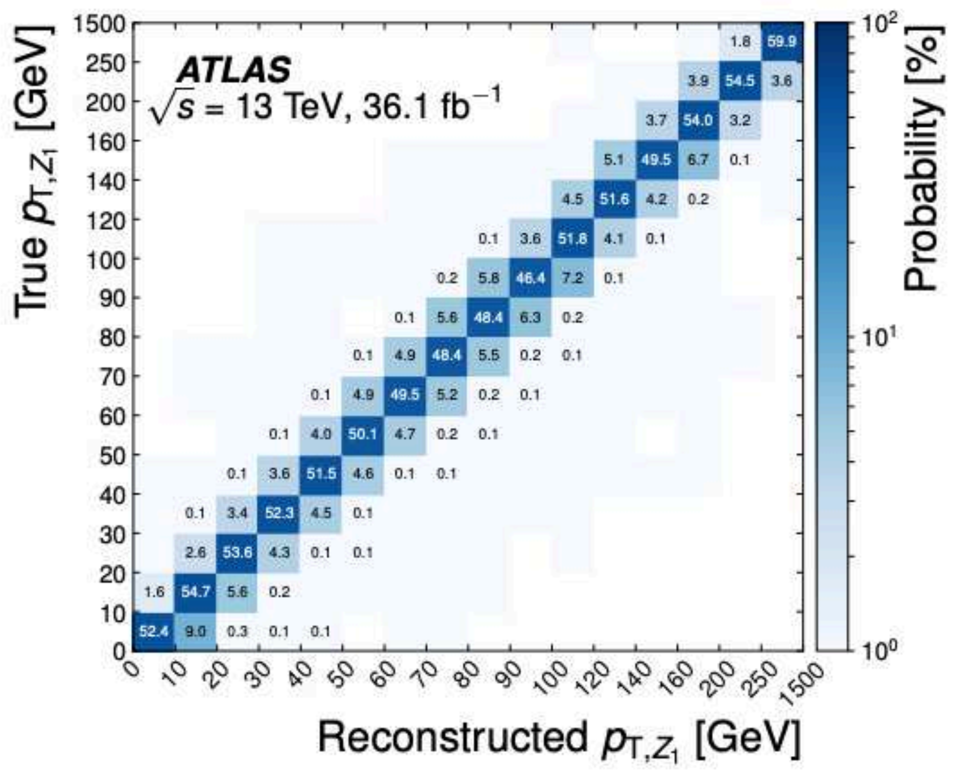
Simultaneously Unfolding All Observables with Deep Learning

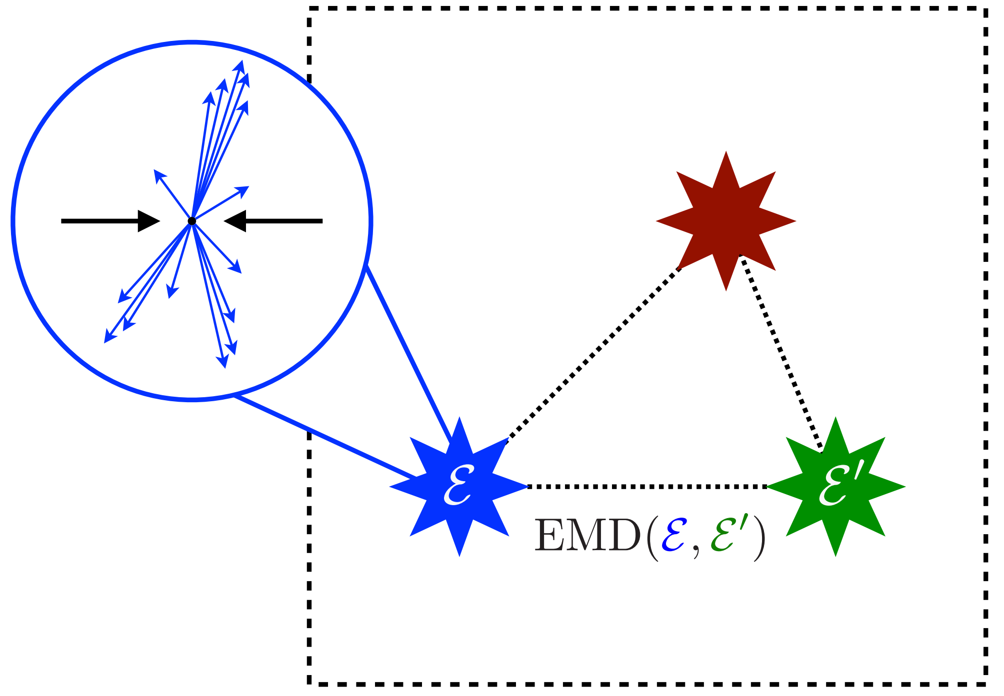
Patrick T. Komiske III
Massachusetts Institute of Technology
Center for Theoretical Physics

Based on work with Anders Andreassen, Eric Metodiev, Ben Nachman, and Jesse Thaler
1911.09107 (PRL)

CMS Machine Learning Forum

February 10, 2021

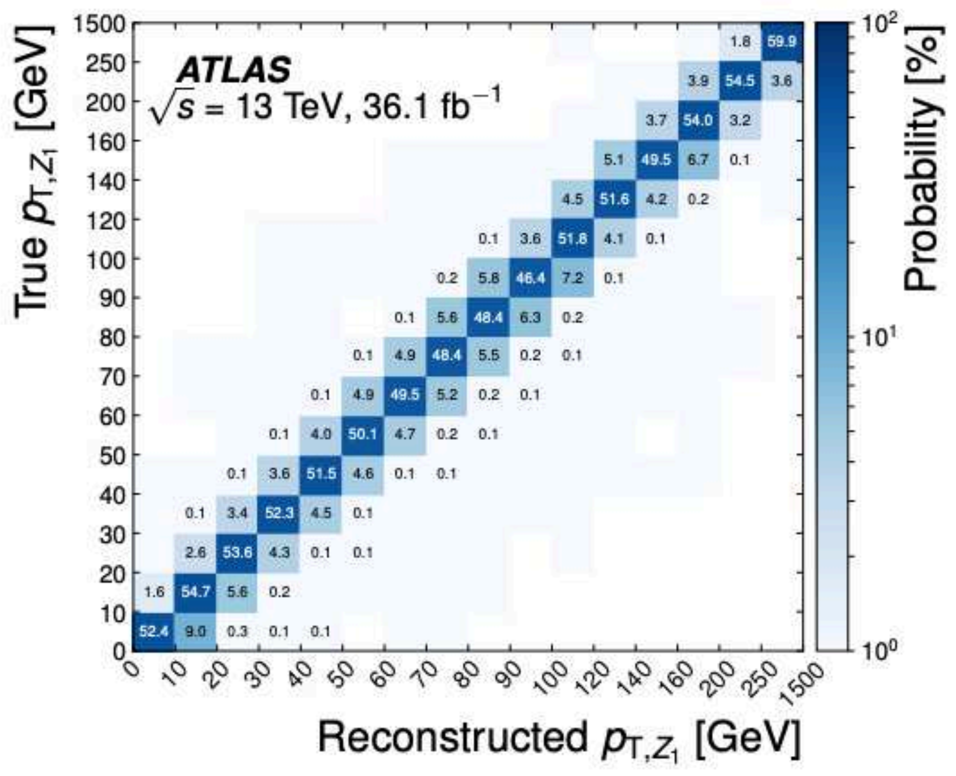




Unfolding Setup

OmniFold

Unfolding Beyond Observables



Unfolding Setup

OmniFold

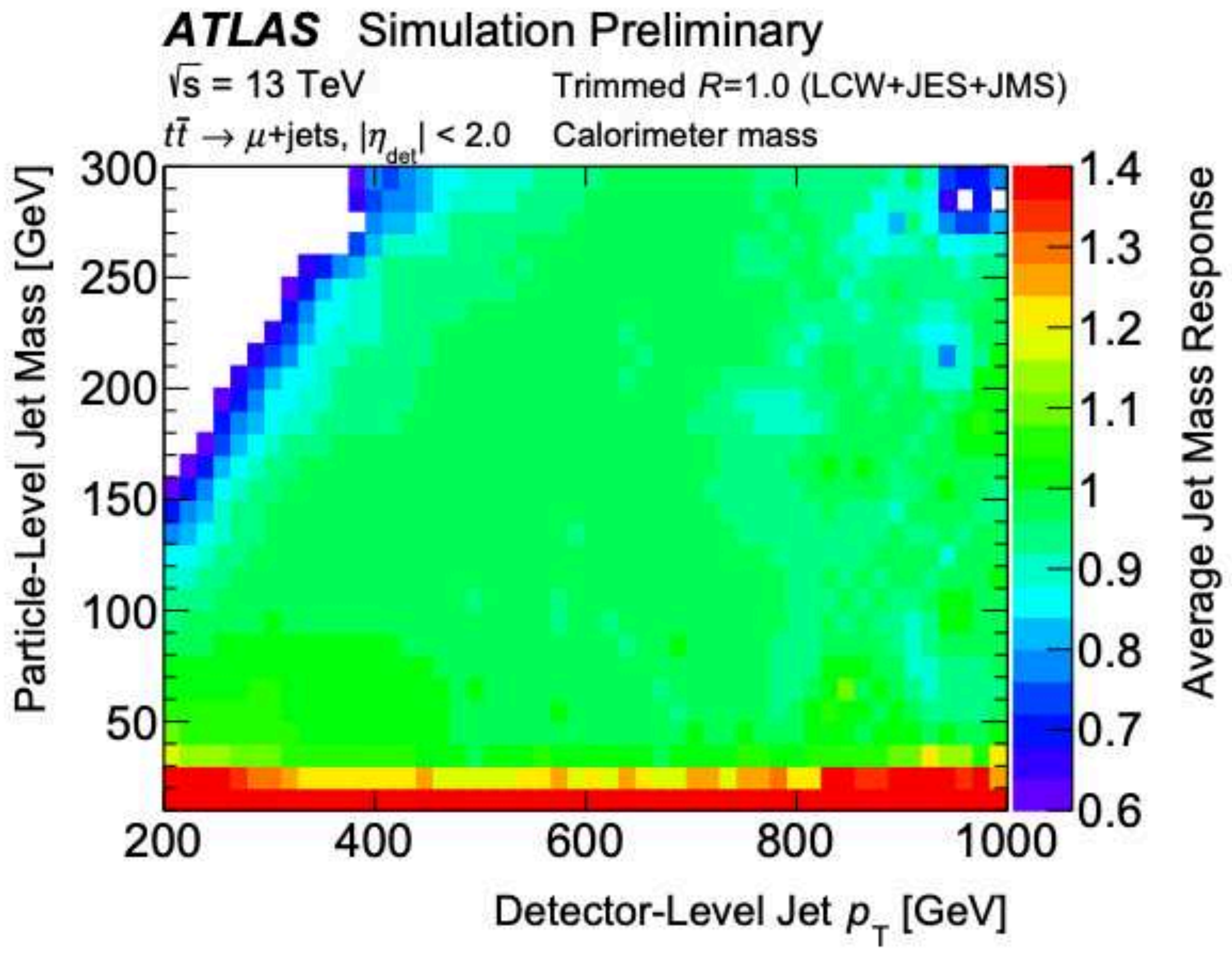
Unfolding Beyond Observables

Correcting for Detector Effects

- Detectors introduce (potentially correlated) smearing and biasing that must be corrected in any measurement
- Material interactions and detector geometry modeled with sophisticated (i.e. expensive) simulation software (e.g. GEANT4)

Forward folding simulates given truth-level events and calculates detector-level quantities

[ATLAS-CONF-2020-022]

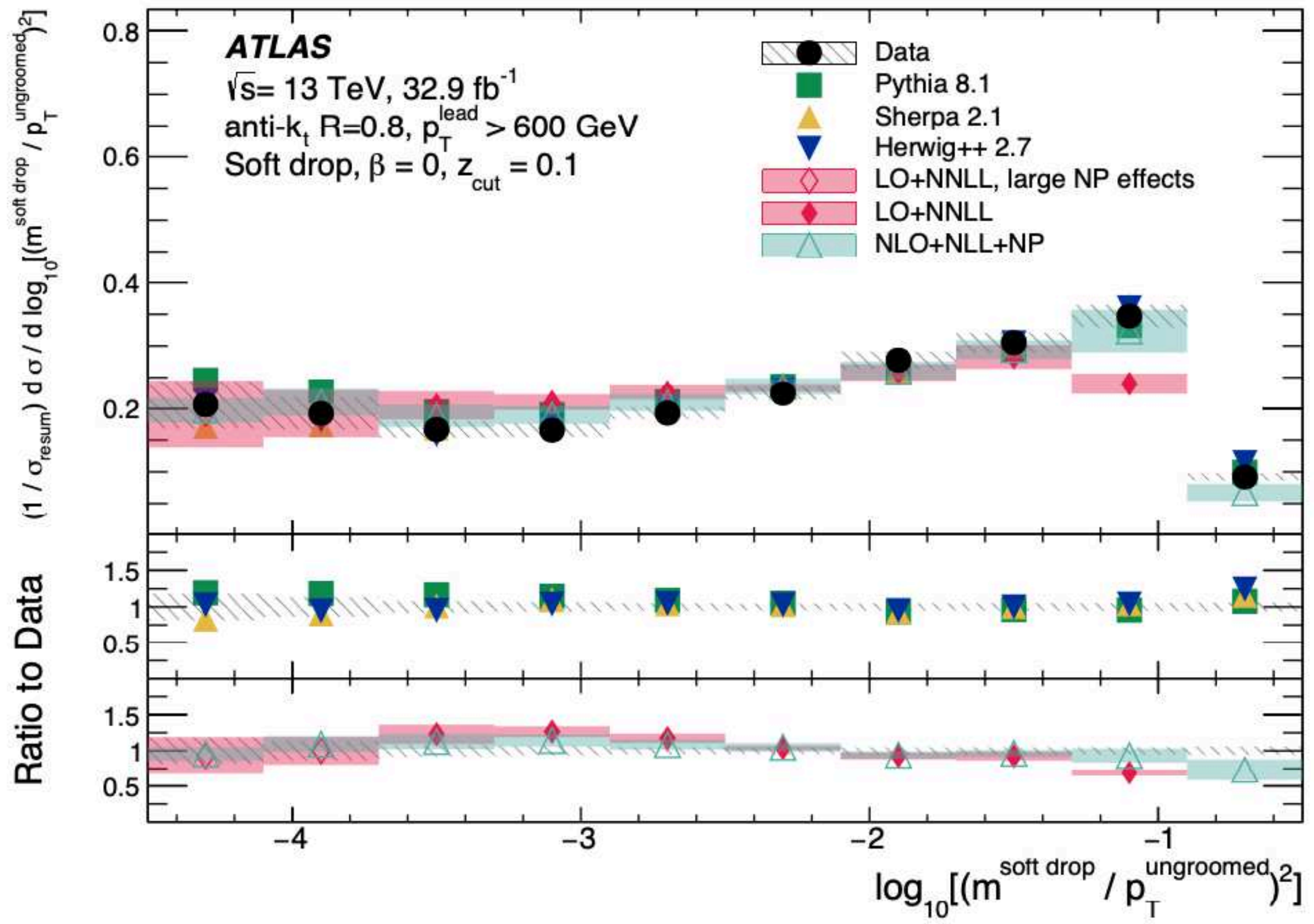


Detector response varies according to jet mass and pT
Explicitly depends on specific detector geometry

Unfolding estimates truth-level quantities given experimental data and information about detector response

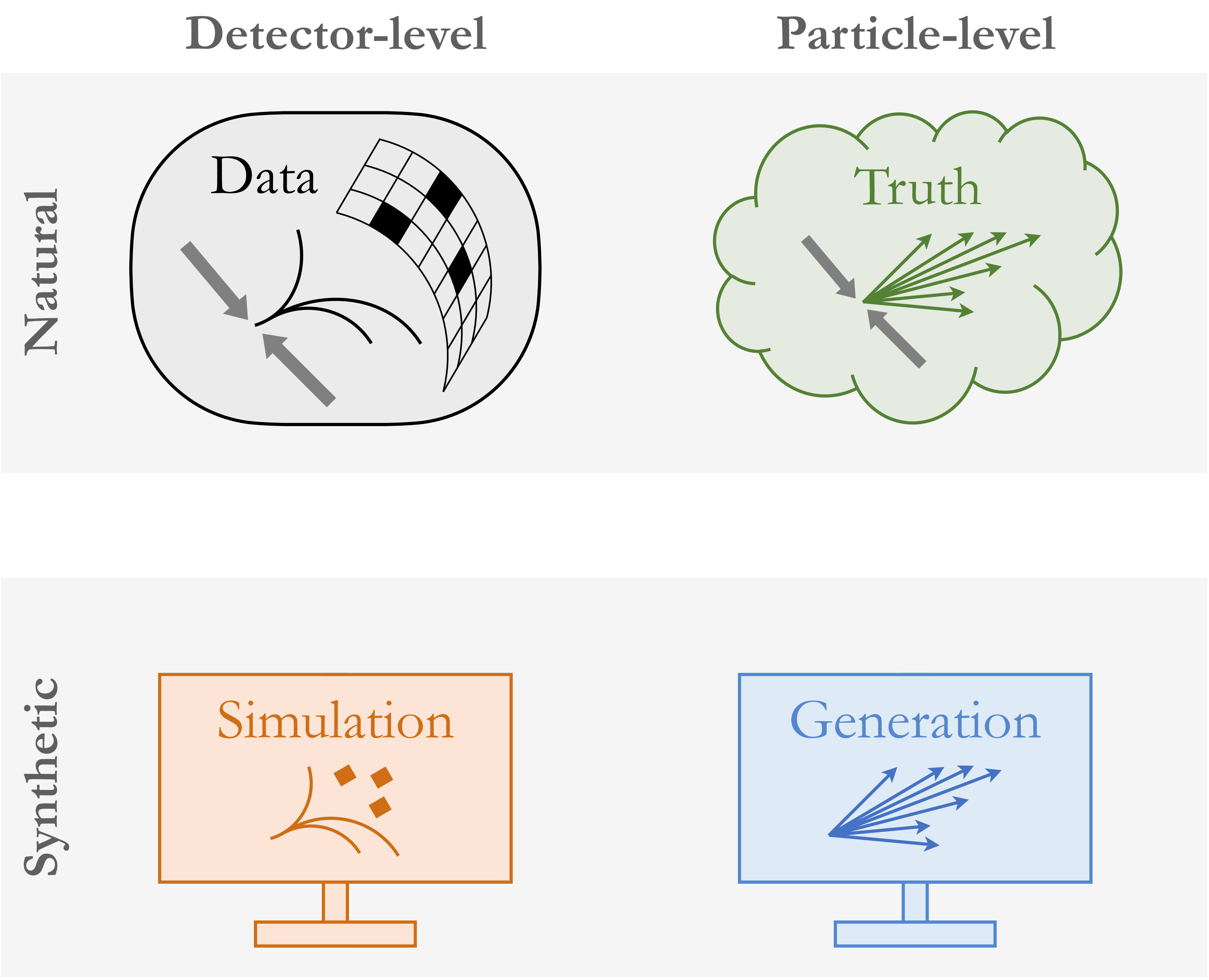
[ATLAS, PRL 2018]

Soft Drop Jet Mass Measurement

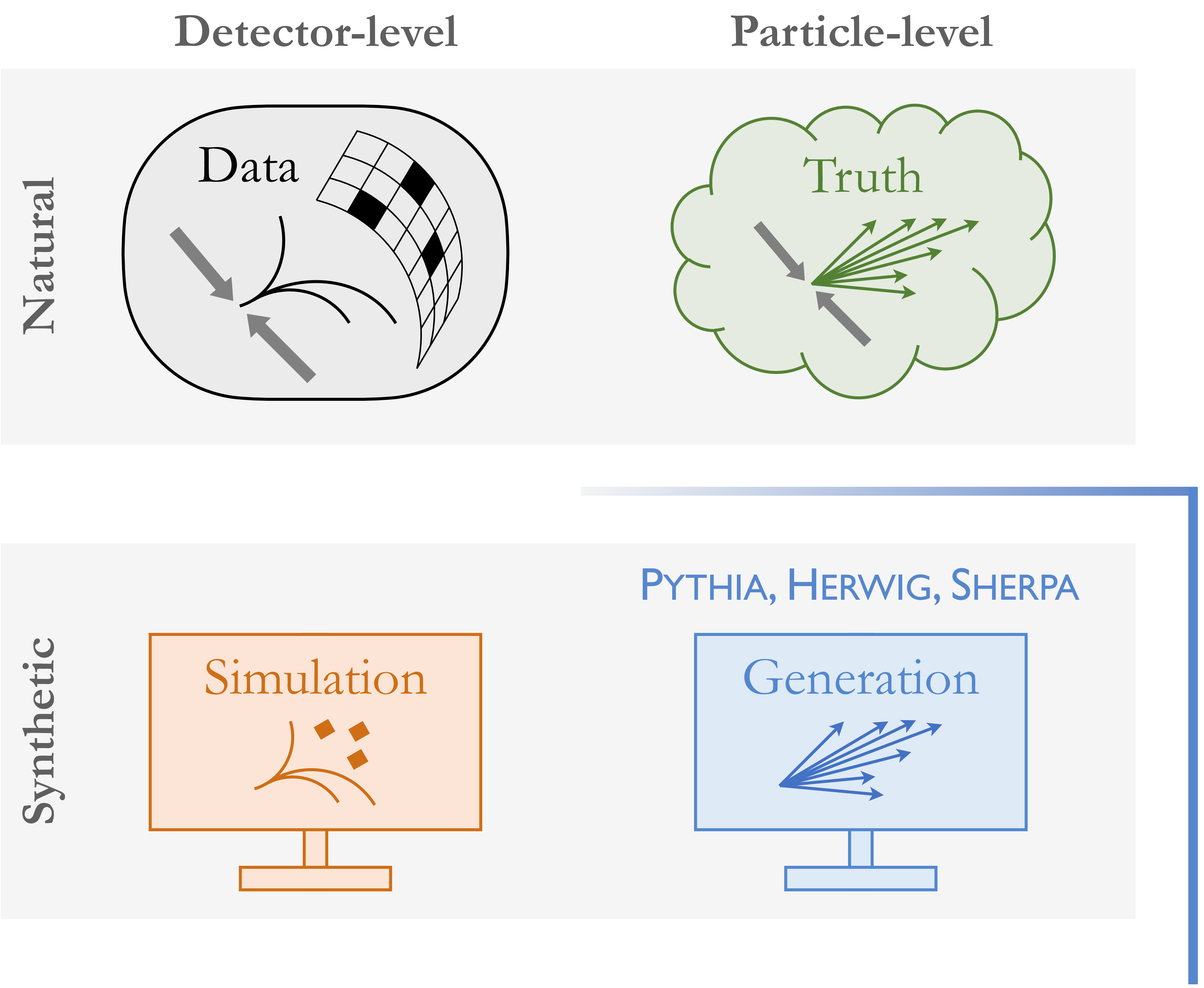


Comparison to precision theory possible in detector-independent manner
Measurement can be used by anyone, no need for detailed experimental information

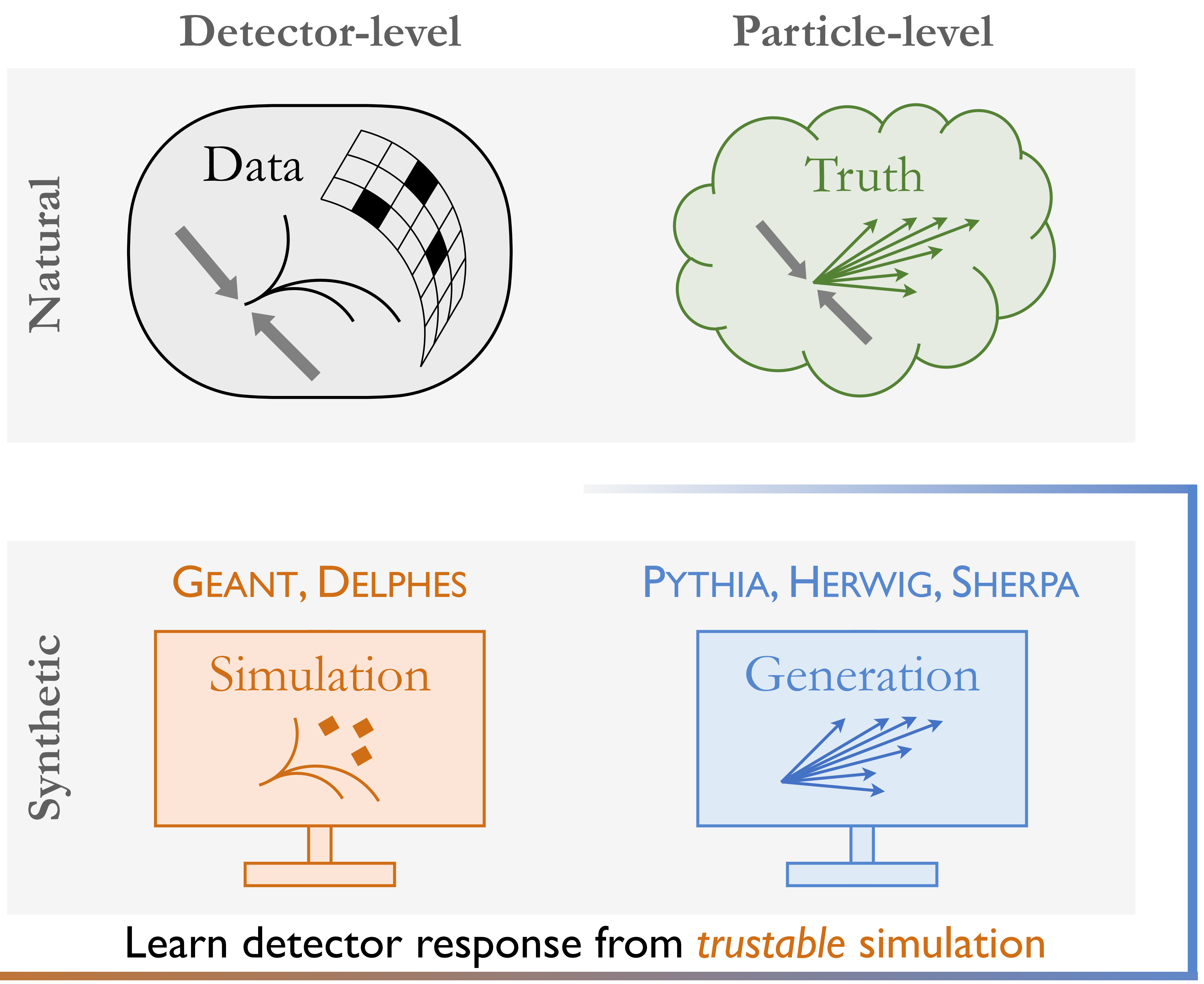
Unfolding Setup



Unfolding Setup

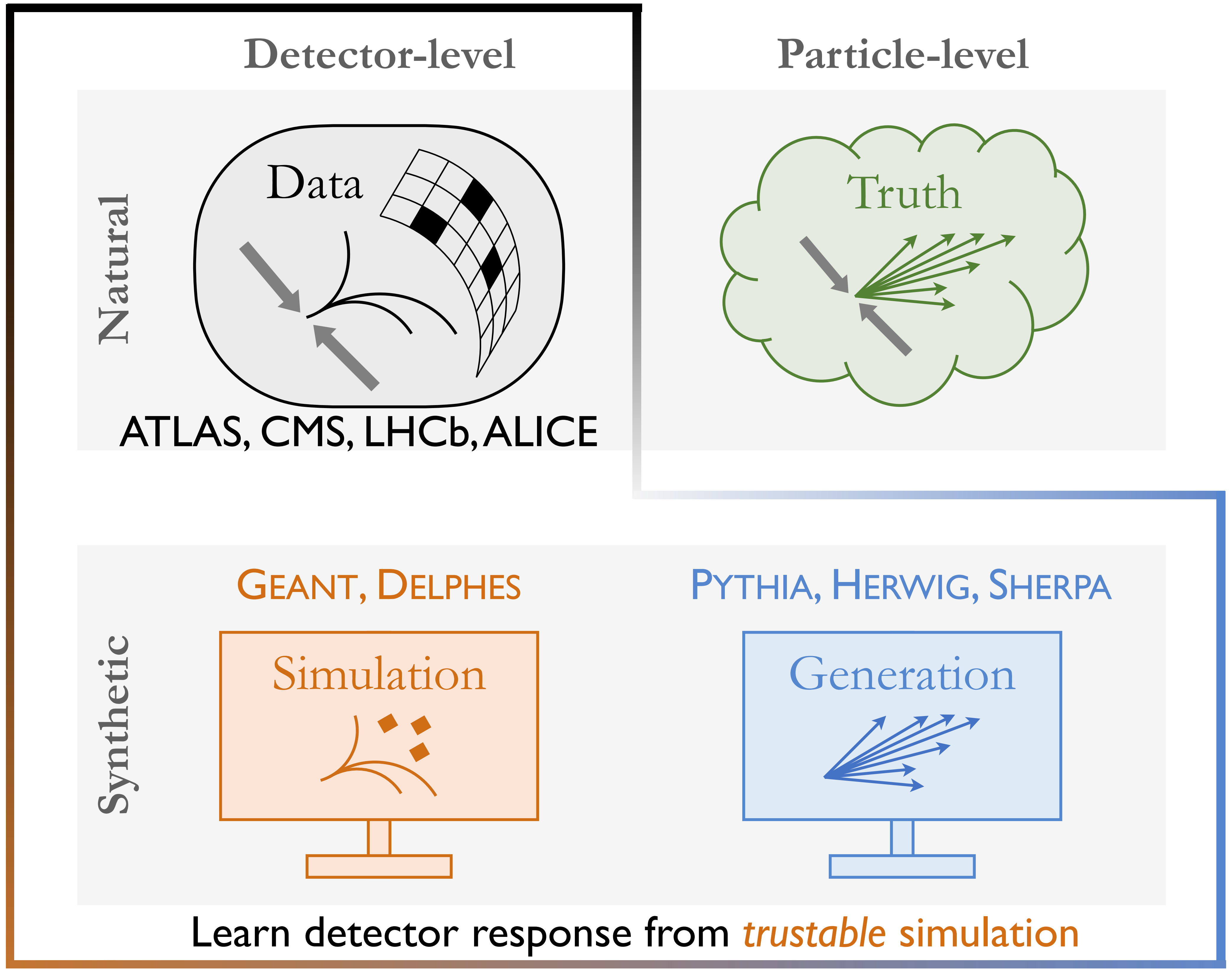


Unfolding Setup



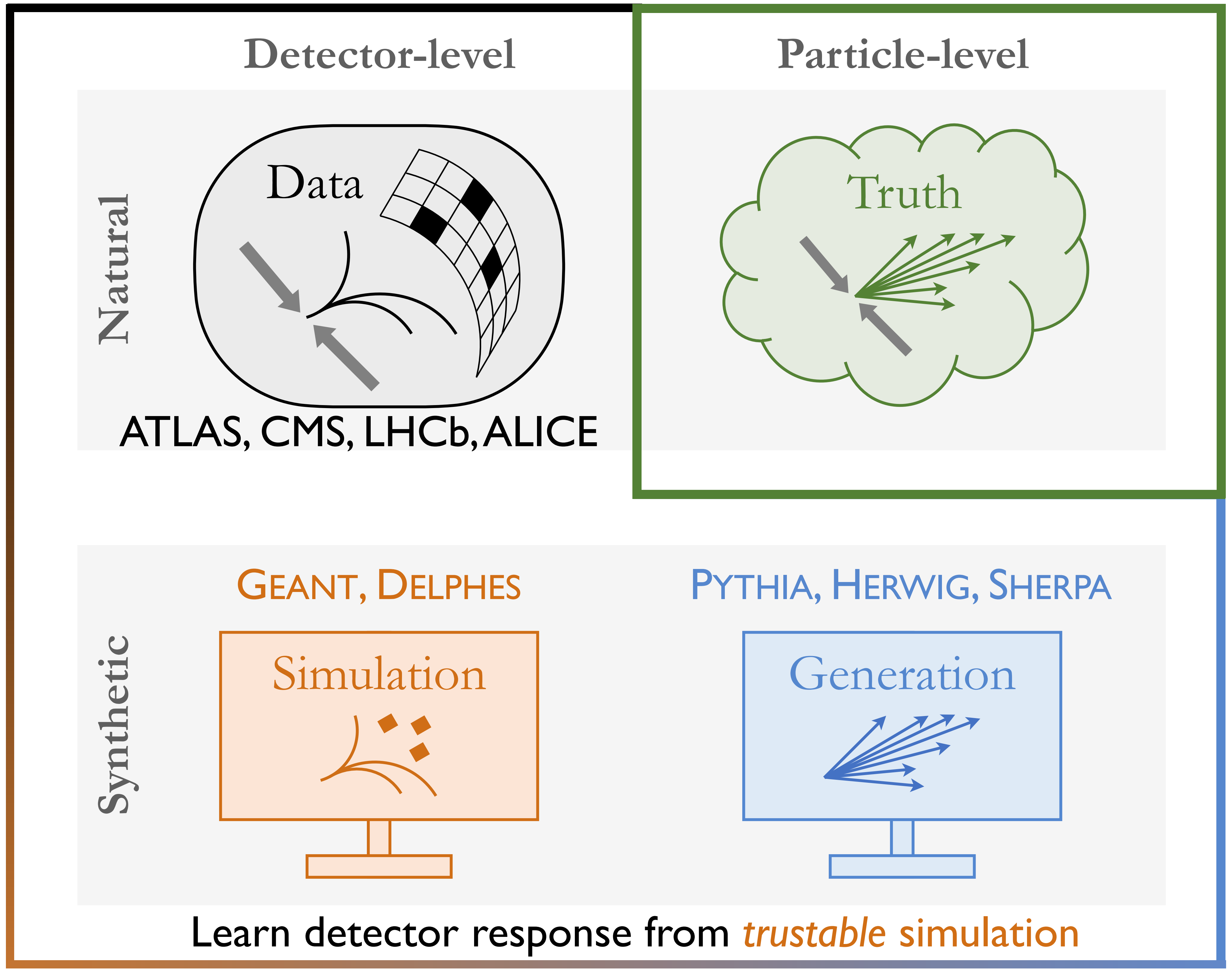
Learn detector response from *trustable* simulation

Unfolding Setup



Unfolding Setup

Truth-level measurements can be compared *across experiments* and to *theoretical calculations*



Goal of unfolding is to learn a generative particle-level model that reproduces the data

WLOG, the generative model can be MC generator + weighting function on particle-level phase space

Challenges with Traditional Unfolding

Previous methods are inherently binned

Binning fixed ahead of time, cannot be changed later

Performance of method sensitive to binning

Limited number of observables

Binning induces curse of dimensionality

Response matrix depends on auxiliary features

Detector-level quantity may not capture full detector effect

Challenges with Traditional Unfolding

Previous methods are inherently binned

Binning fixed ahead of time, cannot be changed later
Performance of method sensitive to binning

Limited number of observables

Binning induces curse of dimensionality

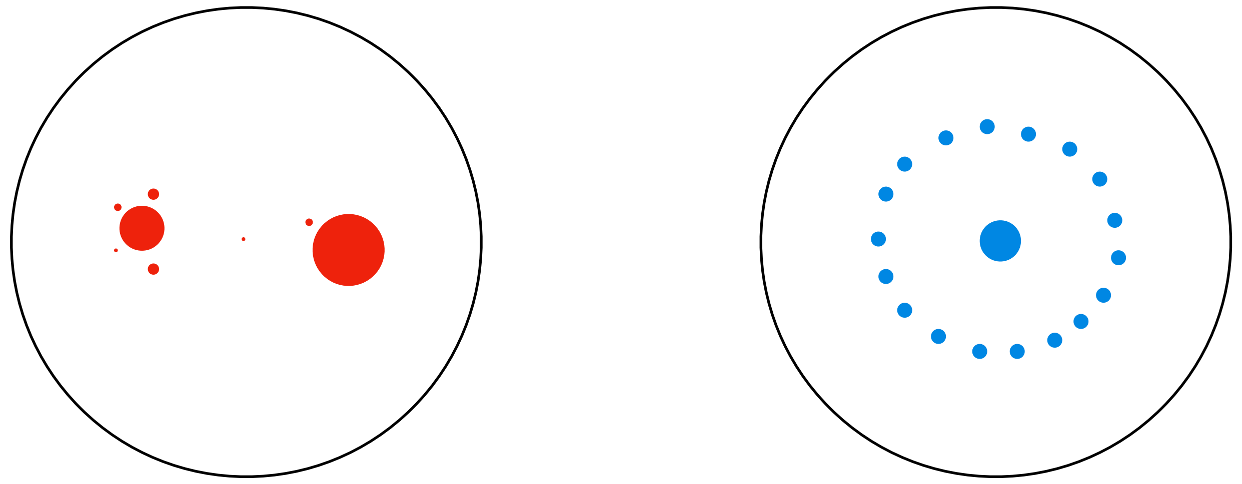
Response matrix depends on auxiliary features

Detector-level quantity may not capture full detector effect

Example – Two jets acquiring the same mass in different ways

Jet 1
Two hard prongs

Jet 2
Hard core, diffuse spray



Challenges with Traditional Unfolding

Previous methods are inherently binned

Binning fixed ahead of time, cannot be changed later

Performance of method sensitive to binning

Limited number of observables

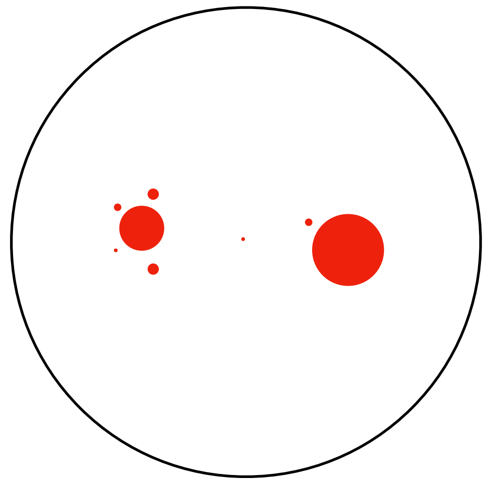
Binning induces curse of dimensionality

Response matrix depends on auxiliary features

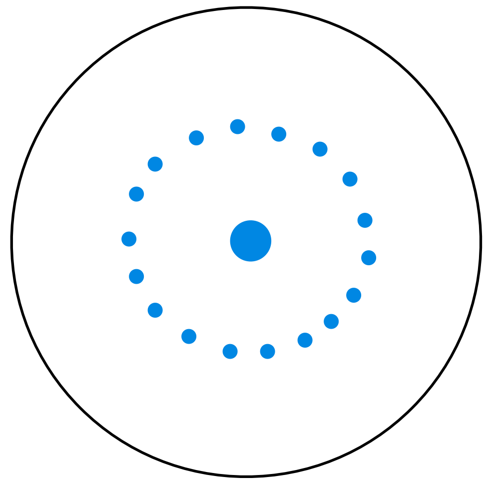
Detector-level quantity may not capture full detector effect

Example – Two jets acquiring the same mass in different ways

Jet 1
Two hard prongs



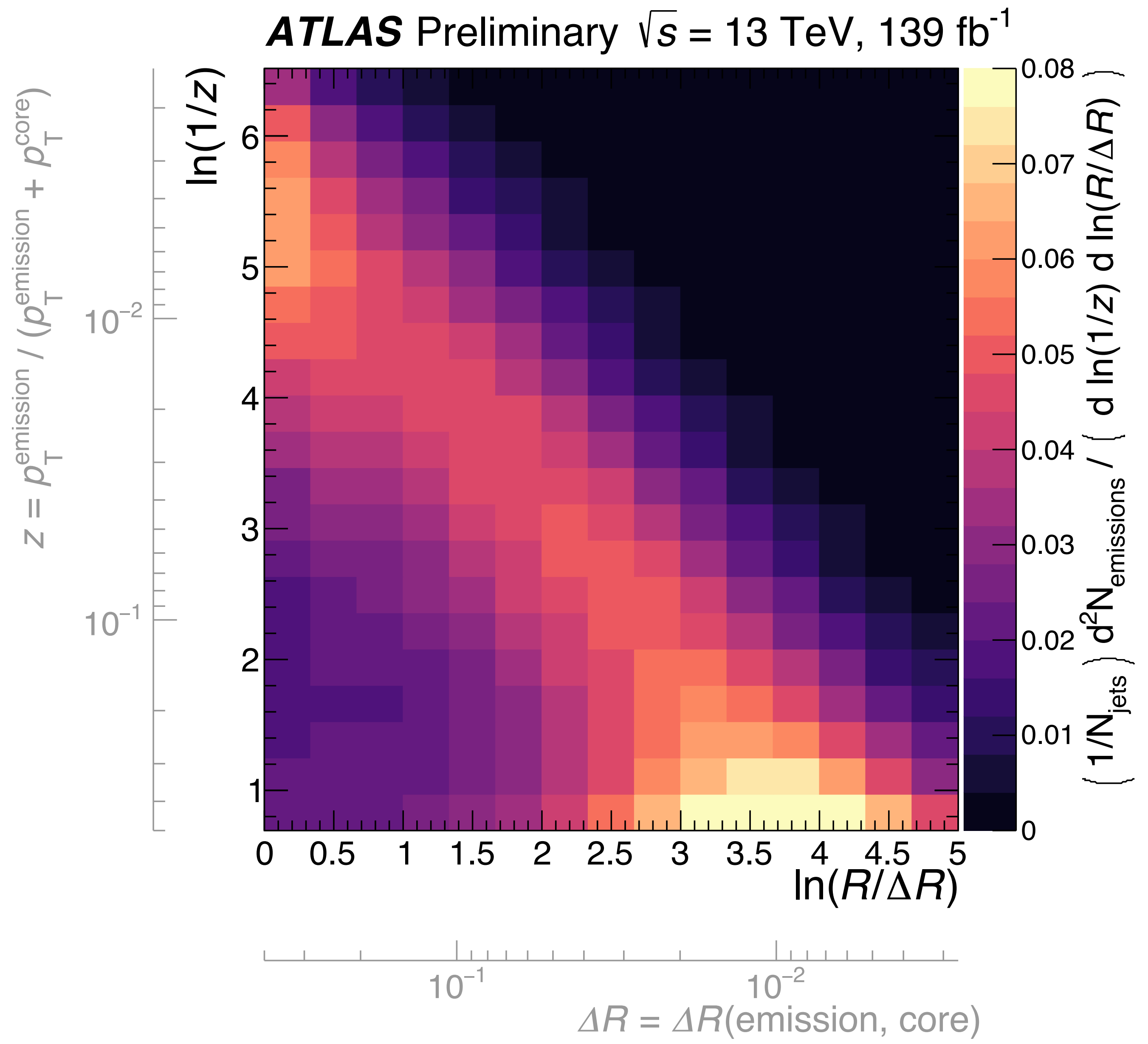
Jet 2
Hard core, diffuse spray



Example with IBU

ATLAS State-of-the-art Lund Plane Measurement

[PRL 2020]



21 x 15 bins in $\ln(1/z)$ x $\ln(R/\Delta R)$

- Must redo unfolding for other binnings e.g. finer/coarser, k_T (diagonal) binning, etc.

Limited to two observables

- 21^2 x 15^2 elements in response matrix R
- Going differential in n bins of p_T would multiply size of R by n^2

Traditional Unfolding

Iterated Bayesian Unfolding (IBU)

[Richardson, JOSA 1972; Lucy, AJ 1974; D'Agostini, NIMPA 1995]

Maximum likelihood, histogram-based unfolding method for a small number of observables

Choose observable(s) and binning at detector-level and particle-level

measured distribution: $m_i = \Pr(\text{measure } i)$

true distribution: $t_j^{(0)} = \Pr(\text{truth is } j)$

Calculate *response matrix* R_{ij} from generated/simulated pairs of events

$$R_{ij} = \Pr(\text{measure } i \mid \text{truth is } j)$$

Calculate new particle-level distribution using Bayes' theorem

$$t_j^{(n)} = \sum_i \Pr(\text{truth}_{n-1} \text{ is } j \mid \text{measure } i) \times \Pr(\text{measure } i)$$

$$= \sum_i \frac{R_{ij} t_j^{(n-1)}}{\sum_k R_{ik} t_k^{(n-1)}} \times m_i$$

Iterate procedure to remove dependence on prior

Consider a situation with two particle-level bins and two detector-level bins

$$t_j^{(0)} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_j$$

Uniform prior

$$m_i = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i$$

Bins are measured equally

$$R_{ij} = \begin{pmatrix} 1 & \frac{1}{2} \\ 0 & \frac{1}{2} \end{pmatrix}_{ij}$$

Bin 1 reconstructed perfectly
Bin 2 reconstructed equally

Iterated Bayesian Unfolding (IBU)

[Richardson, JOSA 1972; Lucy, AJ 1974; D'Agostini, NIMPA 1995]

Maximum likelihood, histogram-based unfolding method for a small number of observables

Choose observable(s) and binning at detector-level and particle-level

measured distribution: $m_i = \Pr(\text{measure } i)$

true distribution: $t_j^{(0)} = \Pr(\text{truth is } j)$

Calculate *response matrix* R_{ij} from generated/simulated pairs of events

$$R_{ij} = \Pr(\text{measure } i \mid \text{truth is } j)$$

Calculate new particle-level distribution using Bayes' theorem

$$t_j^{(n)} = \sum_i \Pr(\text{truth}_{n-1} \text{ is } j \mid \text{measure } i) \times \Pr(\text{measure } i)$$
$$= \sum_i \frac{R_{ij} t_j^{(n-1)}}{\sum_k R_{ik} t_k^{(n-1)}} \times m_i$$

Iterate procedure to remove dependence on prior

Consider a situation with two particle-level bins and two detector-level bins

$$t_j^{(0)} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_j \qquad m_i = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i \qquad R_{ij} = \begin{pmatrix} 1 & \frac{1}{2} \\ 0 & \frac{1}{2} \end{pmatrix}_{ij}$$

Uniform prior

Bins are measured equally

Bin 1 reconstructed perfectly
Bin 2 reconstructed equally

$$t_j^{(1)} = \sum_i \frac{\begin{pmatrix} \frac{1}{2} & \frac{1}{4} \\ 0 & \frac{1}{4} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{3}{4} & \frac{1}{4} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i$$

After one iteration

Iterated Bayesian Unfolding (IBU)

[Richardson, JOSA 1972; Lucy, AJ 1974; D'Agostini, NIMPA 1995]

Maximum likelihood, histogram-based unfolding method for a small number of observables

Choose observable(s) and binning at detector-level and particle-level

measured distribution: $m_i = \Pr(\text{measure } i)$

true distribution: $t_j^{(0)} = \Pr(\text{truth is } j)$

Calculate *response matrix* R_{ij} from generated/simulated pairs of events

$$R_{ij} = \Pr(\text{measure } i \mid \text{truth is } j)$$

Calculate new particle-level distribution using Bayes' theorem

$$t_j^{(n)} = \sum_i \Pr(\text{truth}_{n-1} \text{ is } j \mid \text{measure } i) \times \Pr(\text{measure } i) = \sum_i \frac{R_{ij} t_j^{(n-1)}}{\sum_k R_{ik} t_k^{(n-1)}} \times m_i$$

Iterate procedure to remove dependence on prior

Consider a situation with two particle-level bins and two detector-level bins

$$t_j^{(0)} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_j$$

Uniform prior

$$m_i = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i$$

Bins are measured equally

$$R_{ij} = \begin{pmatrix} 1 & \frac{1}{2} \\ 0 & \frac{1}{2} \end{pmatrix}_{ij}$$

Bin 1 reconstructed perfectly
Bin 2 reconstructed equally

$$t_j^{(1)} = \sum_i \frac{\begin{pmatrix} \frac{1}{2} & \frac{1}{4} \\ 0 & \frac{1}{4} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{3}{4} & \frac{1}{4} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i = \begin{pmatrix} \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{2}}_{1/3} + 0 \\ \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{4}}_{1/6} + \underbrace{\frac{1}{2} \times 4 \times \frac{1}{4}}_{1/2} \end{pmatrix} = \begin{pmatrix} \frac{1}{3} \\ \frac{2}{3} \end{pmatrix}_j$$

After one iteration

Iterated Bayesian Unfolding (IBU)

[Richardson, JOSA 1972; Lucy, AJ 1974; D'Agostini, NIMPA 1995]

Maximum likelihood, histogram-based unfolding method for a small number of observables

Choose observable(s) and binning at detector-level and particle-level

measured distribution: $m_i = \Pr(\text{measure } i)$

true distribution: $t_j^{(0)} = \Pr(\text{truth is } j)$

Calculate *response matrix* R_{ij} from generated/simulated pairs of events

$$R_{ij} = \Pr(\text{measure } i \mid \text{truth is } j)$$

Calculate new particle-level distribution using Bayes' theorem

$$t_j^{(n)} = \sum_i \Pr(\text{truth}_{n-1} \text{ is } j \mid \text{measure } i) \times \Pr(\text{measure } i)$$

$$= \sum_i \frac{R_{ij} t_j^{(n-1)}}{\sum_k R_{ik} t_k^{(n-1)}} \times m_i$$

Iterate procedure to remove dependence on prior

Consider a situation with two particle-level bins and two detector-level bins

$$t_j^{(0)} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_j \qquad m_i = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i \qquad R_{ij} = \begin{pmatrix} 1 & \frac{1}{2} \\ 0 & \frac{1}{2} \end{pmatrix}_{ij}$$

Uniform prior

Bins are measured equally

Bin 1 reconstructed perfectly
Bin 2 reconstructed equally

$$t_j^{(1)} = \sum_i \frac{\begin{pmatrix} \frac{1}{2} & \frac{1}{4} \\ 0 & \frac{1}{4} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{3}{4} & \frac{1}{4} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i = \begin{pmatrix} \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{2}}_{1/3} + 0 \\ \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{4}}_{1/6} + \underbrace{\frac{1}{2} \times 4 \times \frac{1}{4}}_{1/2} \end{pmatrix} = \begin{pmatrix} \frac{1}{3} \\ \frac{2}{3} \end{pmatrix}_j$$

After one iteration

$$\vdots$$

$$t_j^{(n)} = \sum_i \frac{\begin{pmatrix} \frac{1}{n+1} & \frac{n}{2(n+1)} \\ 0 & \frac{n}{2(n+1)} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{n+2}{2(n+1)} & \frac{n}{2(n+1)} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i = \begin{pmatrix} \frac{1}{n+2} \\ \frac{n+1}{n+2} \end{pmatrix}_j \to \begin{pmatrix} 0 \\ 1 \end{pmatrix}_j$$

At the nth iteration

Correct truth distribution obtained as $n \to \infty$

IBU as Reweighting

[Richardson, JOSA 1972; Lucy, AJ 1974; D'Agostini, NIMPA 1995]

Consider a situation with two particle-level bins and two detector-level bins

$$t_j^{(0)} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_j \qquad m_i = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i \qquad R_{ij} = \begin{pmatrix} 1 & \frac{1}{2} \\ 0 & \frac{1}{2} \end{pmatrix}_{ij}$$

Uniform prior

Bins are measured equally

Bin 1 reconstructed perfectly
Bin 2 reconstructed equally

$$t_j^{(1)} = \sum_i \frac{\begin{pmatrix} \frac{1}{2} & \frac{1}{4} \\ 0 & \frac{1}{4} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{3}{4} & \frac{1}{4} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i = \begin{pmatrix} \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{2}}_{1/3} + 0 \\ \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{4}}_{1/6} + \underbrace{\frac{1}{2} \times 4 \times \frac{1}{4}}_{1/2} \end{pmatrix} = \begin{pmatrix} \frac{1}{3} \\ \frac{2}{3} \end{pmatrix}_j$$

After one iteration

$$\vdots$$

Correct truth distribution obtained as $n \to \infty$

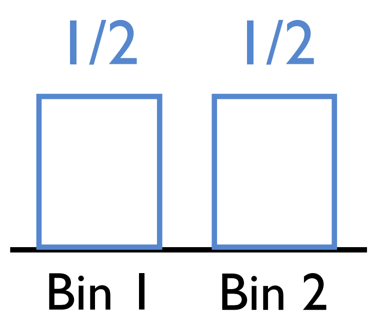
$$t_j^{(n)} = \sum_i \frac{\begin{pmatrix} \frac{1}{n+1} & \frac{n}{2(n+1)} \\ 0 & \frac{n}{2(n+1)} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{n+2}{2(n+1)} & \frac{n}{2(n+1)} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i = \begin{pmatrix} \frac{1}{n+2} \\ \frac{n+1}{n+2} \end{pmatrix}_j \to \begin{pmatrix} 0 \\ 1 \end{pmatrix}_j$$

At the n$^{\text{th}}$ iteration

IBU as Reweighting

[Richardson, JOSA 1972; Lucy, AJ 1974; D'Agostini, NIMPA 1995]

prior



Consider a situation with two particle-level bins and two detector-level bins

$$t_j^{(0)} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_j \qquad m_i = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i \qquad R_{ij} = \begin{pmatrix} 1 & \frac{1}{2} \\ 0 & \frac{1}{2} \end{pmatrix}_{ij}$$

Uniform prior

Bins are measured equally

Bin 1 reconstructed perfectly
Bin 2 reconstructed equally

$$t_j^{(1)} = \sum_i \frac{\begin{pmatrix} \frac{1}{2} & \frac{1}{4} \\ 0 & \frac{1}{4} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{3}{4} & \frac{1}{4} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i = \begin{pmatrix} \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{2} + 0}_{1/3} \\ \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{4}}_{1/6} + \underbrace{\frac{1}{2} \times 4 \times \frac{1}{4}}_{1/2} \end{pmatrix} = \begin{pmatrix} \frac{1}{3} \\ \frac{2}{3} \end{pmatrix}_j$$

After one iteration

$$\vdots$$

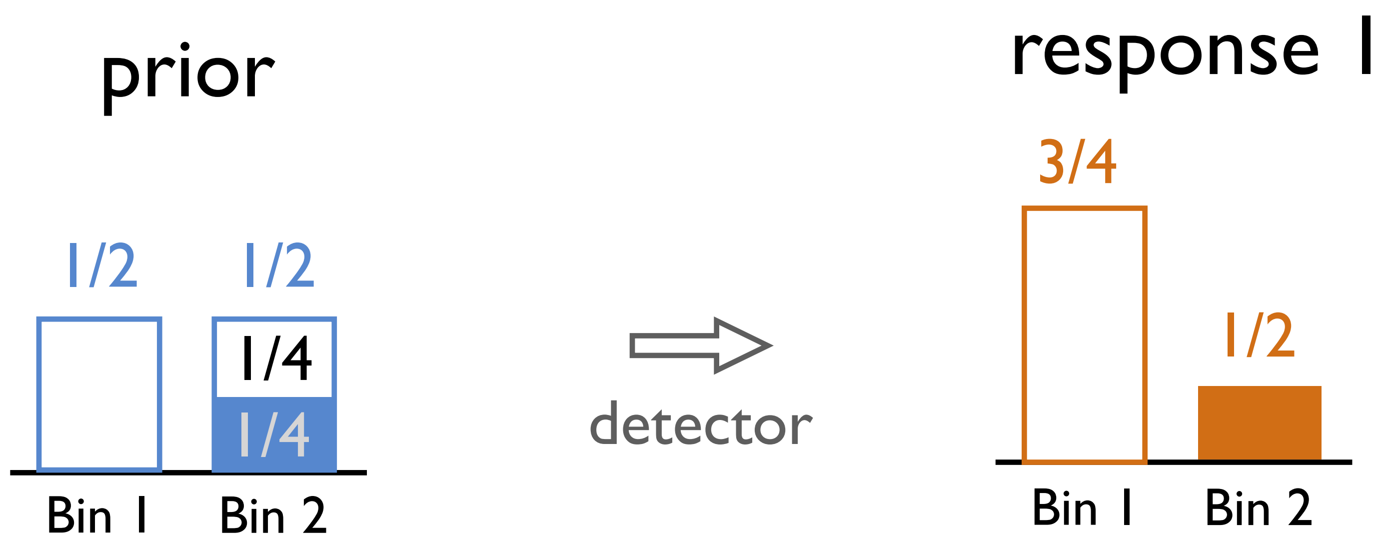
$$t_j^{(n)} = \sum_i \frac{\begin{pmatrix} \frac{1}{n+1} & \frac{n}{2(n+1)} \\ 0 & \frac{n}{2(n+1)} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{n+2}{2(n+1)} & \frac{n}{2(n+1)} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i = \begin{pmatrix} \frac{1}{n+2} \\ \frac{n+1}{n+2} \end{pmatrix}_j \to \begin{pmatrix} 0 \\ 1 \end{pmatrix}_j$$

At the n$^{\text{th}}$ iteration

Correct truth distribution obtained as $n \to \infty$

IBU as Reweighting

[Richardson, JOSA 1972; Lucy, AJ 1974; D'Agostini, NIMPA 1995]



Consider a situation with two particle-level bins and two detector-level bins

$$t_j^{(0)} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_j \qquad m_i = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i \qquad R_{ij} = \begin{pmatrix} 1 & \frac{1}{2} \\ 0 & \frac{1}{2} \end{pmatrix}_{ij}$$

Uniform prior

Bins are measured equally

Bin 1 reconstructed perfectly
Bin 2 reconstructed equally

$$t_j^{(1)} = \sum_i \frac{\begin{pmatrix} \frac{1}{2} & \frac{1}{4} \\ 0 & \frac{1}{4} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{3}{4} & \frac{1}{4} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i = \begin{pmatrix} \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{2} + 0}_{1/3} \\ \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{4}}_{1/6} + \underbrace{\frac{1}{2} \times 4 \times \frac{1}{4}}_{1/2} \end{pmatrix} = \begin{pmatrix} \frac{1}{3} \\ \frac{2}{3} \end{pmatrix}_j$$

After one iteration

$$\vdots$$

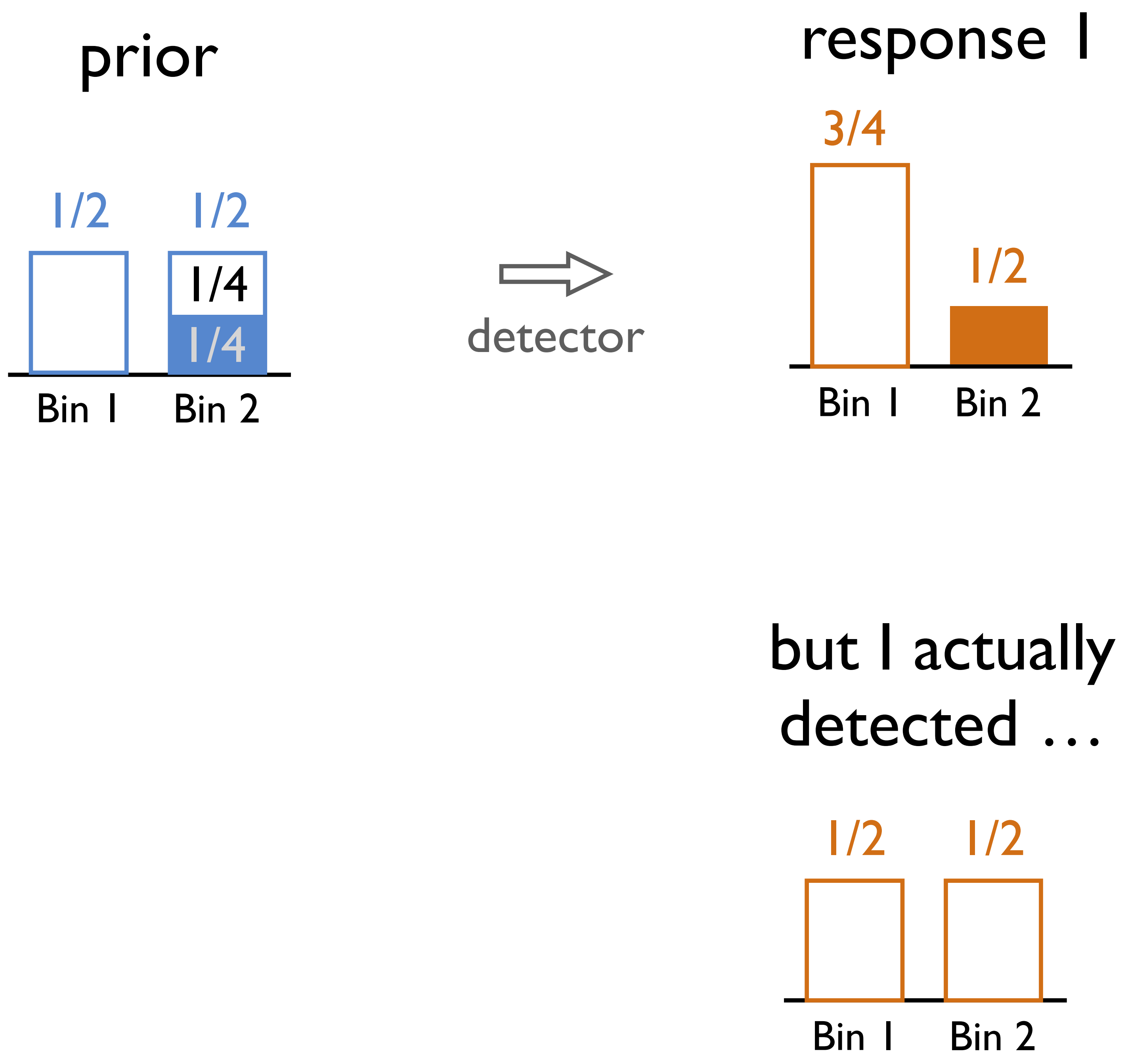
Correct truth distribution obtained as $n \to \infty$

$$t_j^{(n)} = \sum_i \frac{\begin{pmatrix} \frac{1}{n+1} & \frac{n}{2(n+1)} \\ 0 & \frac{n}{2(n+1)} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{n+2}{2(n+1)} & \frac{n}{2(n+1)} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i = \begin{pmatrix} \frac{1}{n+2} \\ \frac{n+1}{n+2} \end{pmatrix}_j \to \begin{pmatrix} 0 \\ 1 \end{pmatrix}_j$$

At the n^{th} iteration

IBU as Reweighting

[Richardson, JOSA 1972; Lucy, AJ 1974; D'Agostini, NIMPA 1995]



Consider a situation with two particle-level bins and two detector-level bins

$$t_j^{(0)} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_j \qquad m_i = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i \qquad R_{ij} = \begin{pmatrix} 1 & \frac{1}{2} \\ 0 & \frac{1}{2} \end{pmatrix}_{ij}$$

Uniform prior — Bins are measured equally — Bin 1 reconstructed perfectly, Bin 2 reconstructed equally

$$t_j^{(1)} = \sum_i \frac{\begin{pmatrix} \frac{1}{2} & \frac{1}{4} \\ 0 & \frac{1}{4} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{3}{4} & \frac{1}{4} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i = \begin{pmatrix} \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{2}}_{1/3} + 0 \\ \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{4}}_{1/6} + \underbrace{\frac{1}{2} \times 4 \times \frac{1}{4}}_{1/2} \end{pmatrix} = \begin{pmatrix} \frac{1}{3} \\ \frac{2}{3} \end{pmatrix}_j$$

After one iteration

⋮

Correct truth distribution obtained as $n \to \infty$

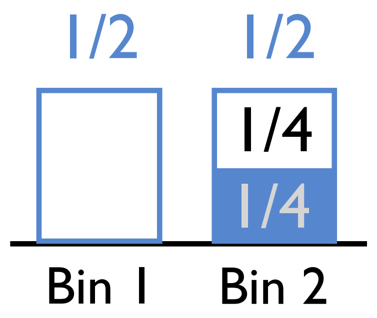
$$t_j^{(n)} = \sum_i \frac{\begin{pmatrix} \frac{1}{n+1} & \frac{n}{2(n+1)} \\ 0 & \frac{n}{2(n+1)} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{n+2}{2(n+1)} & \frac{n}{2(n+1)} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i = \begin{pmatrix} \frac{1}{n+2} \\ \frac{n+1}{n+2} \end{pmatrix}_j \to \begin{pmatrix} 0 \\ 1 \end{pmatrix}_j$$

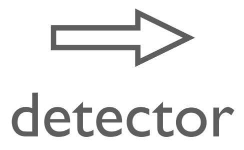
At the n^th iteration

IBU as Reweighting

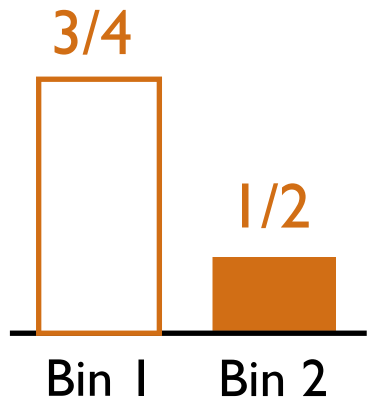
[Richardson, JOSA 1972; Lucy, AJ 1974; D'Agostini, NIMPA 1995]

prior

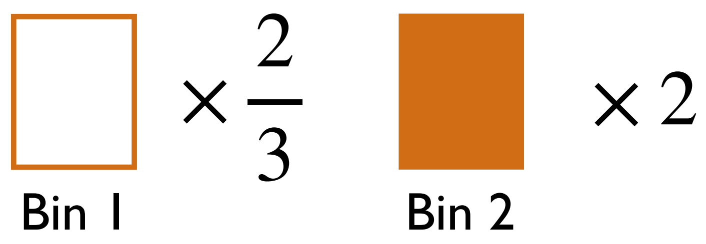




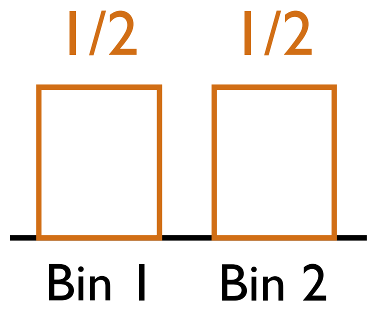
response 1



reweight bins to match



but I actually detected …



Consider a situation with two particle-level bins and two detector-level bins

$$t_j^{(0)} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_j \qquad m_i = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i \qquad R_{ij} = \begin{pmatrix} 1 & \frac{1}{2} \\ 0 & \frac{1}{2} \end{pmatrix}_{ij}$$

Uniform prior

Bins are measured equally

Bin 1 reconstructed perfectly
Bin 2 reconstructed equally

$$t_j^{(1)} = \sum_i \frac{\begin{pmatrix} \frac{1}{2} & \frac{1}{4} \\ 0 & \frac{1}{4} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{3}{4} & \frac{1}{4} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i = \begin{pmatrix} \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{2}}_{1/3} + 0 \\ \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{4}}_{1/6} + \underbrace{\frac{1}{2} \times 4 \times \frac{1}{4}}_{1/2} \end{pmatrix} = \begin{pmatrix} \frac{1}{3} \\ \frac{2}{3} \end{pmatrix}_j$$

After one iteration

$\vdots$

$$t_j^{(n)} = \sum_i \frac{\begin{pmatrix} \frac{1}{n+1} & \frac{n}{2(n+1)} \\ 0 & \frac{n}{2(n+1)} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{n+2}{2(n+1)} & \frac{n}{2(n+1)} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i = \begin{pmatrix} \frac{1}{n+2} \\ \frac{n+1}{n+2} \end{pmatrix}_j \rightarrow \begin{pmatrix} 0 \\ 1 \end{pmatrix}_j$$

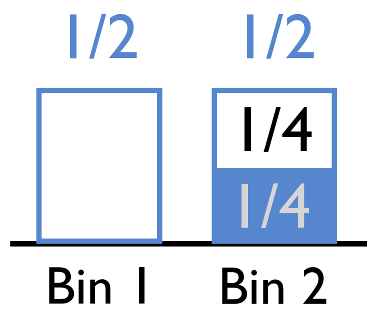
At the n^{th} iteration

Correct truth distribution obtained as $n \rightarrow \infty$

IBU as Reweighting

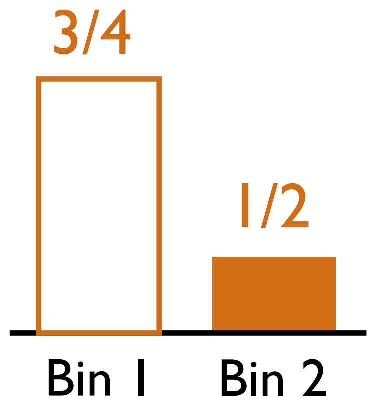
[Richardson, JOSA 1972; Lucy, AJ 1974; D'Agostini, NIMPA 1995]

prior

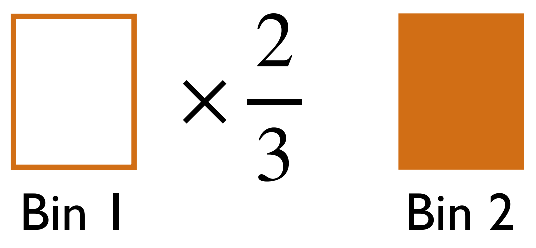


detector

response 1

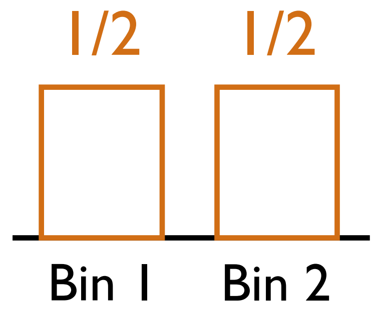


reweight bins to match

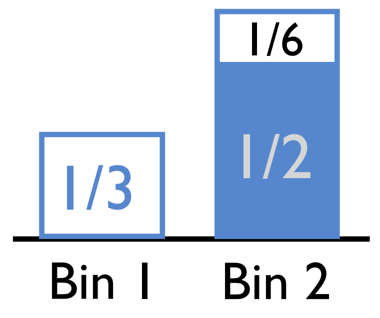


$\times \frac{2}{3}$ $\times 2$

but I actually detected …



pull reweighting back to truth level



Consider a situation with two particle-level bins and two detector-level bins

$$t_j^{(0)} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_j \qquad m_i = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i \qquad R_{ij} = \begin{pmatrix} 1 & \frac{1}{2} \\ 0 & \frac{1}{2} \end{pmatrix}_{ij}$$

Uniform prior

Bins are measured equally

Bin 1 reconstructed perfectly
Bin 2 reconstructed equally

$$t_j^{(1)} = \sum_i \frac{\begin{pmatrix} \frac{1}{2} & \frac{1}{4} \\ 0 & \frac{1}{4} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{3}{4} & \frac{1}{4} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i = \begin{pmatrix} \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{2}}_{1/3} + 0 \\ \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{4}}_{1/6} + \underbrace{\frac{1}{2} \times 4 \times \frac{1}{4}}_{1/2} \end{pmatrix} = \begin{pmatrix} \frac{1}{3} \\ \frac{2}{3} \end{pmatrix}_j$$

After one iteration

$\vdots$

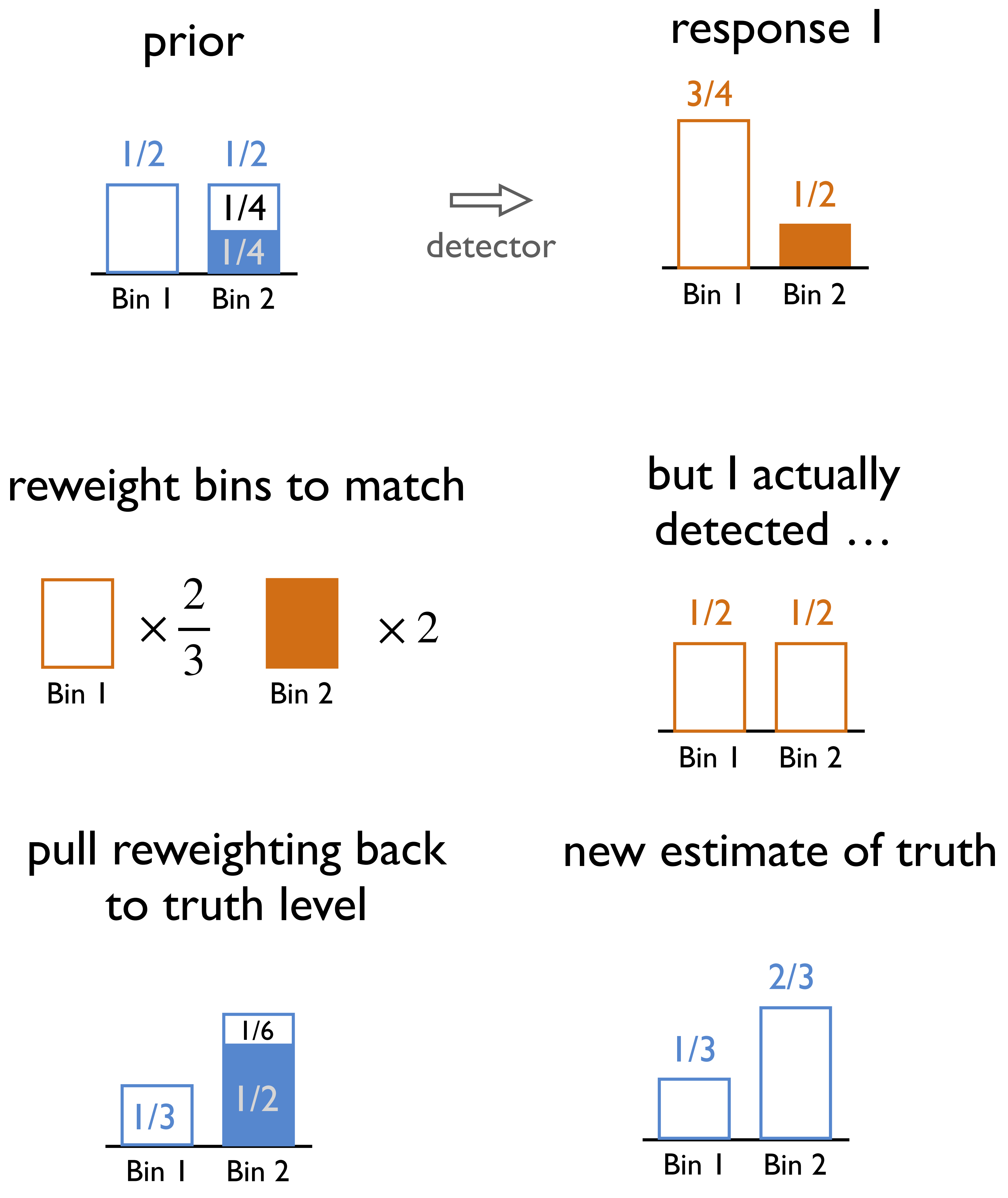
$$t_j^{(n)} = \sum_i \frac{\begin{pmatrix} \frac{1}{n+1} & \frac{n}{2(n+1)} \\ 0 & \frac{n}{2(n+1)} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{n+2}{2(n+1)} & \frac{n}{2(n+1)} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i = \begin{pmatrix} \frac{1}{n+2} \\ \frac{n+1}{n+2} \end{pmatrix}_j \rightarrow \begin{pmatrix} 0 \\ 1 \end{pmatrix}_j$$

Correct truth distribution obtained as $n \to \infty$

At the n[th] iteration

IBU as Reweighting

[Richardson, JOSA 1972; Lucy, AJ 1974; D'Agostini, NIMPA 1995]



Consider a situation with two particle-level bins and two detector-level bins

$$t_j^{(0)} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_j \qquad m_i = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i \qquad R_{ij} = \begin{pmatrix} 1 & \frac{1}{2} \\ 0 & \frac{1}{2} \end{pmatrix}_{ij}$$

Uniform prior

Bins are measured equally

Bin 1 reconstructed perfectly
Bin 2 reconstructed equally

$$t_j^{(1)} = \sum_i \frac{\begin{pmatrix} \frac{1}{2} & \frac{1}{4} \\ 0 & \frac{1}{4} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{3}{4} & \frac{1}{4} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i = \begin{pmatrix} \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{2} + 0}_{1/3} \\ \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{4}}_{1/6} + \underbrace{\frac{1}{2} \times 4 \times \frac{1}{4}}_{1/2} \end{pmatrix} = \begin{pmatrix} \frac{1}{3} \\ \frac{2}{3} \end{pmatrix}_j$$

After one iteration

⋮

$$t_j^{(n)} = \sum_i \frac{\begin{pmatrix} \frac{1}{n+1} & \frac{n}{2(n+1)} \\ 0 & \frac{n}{2(n+1)} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{n+2}{2(n+1)} & \frac{n}{2(n+1)} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i = \begin{pmatrix} \frac{1}{n+2} \\ \frac{n+1}{n+2} \end{pmatrix}_j \to \begin{pmatrix} 0 \\ 1 \end{pmatrix}_j$$

Correct truth distribution obtained as $n \to \infty$

At the n[th] iteration

Unfolding Setup

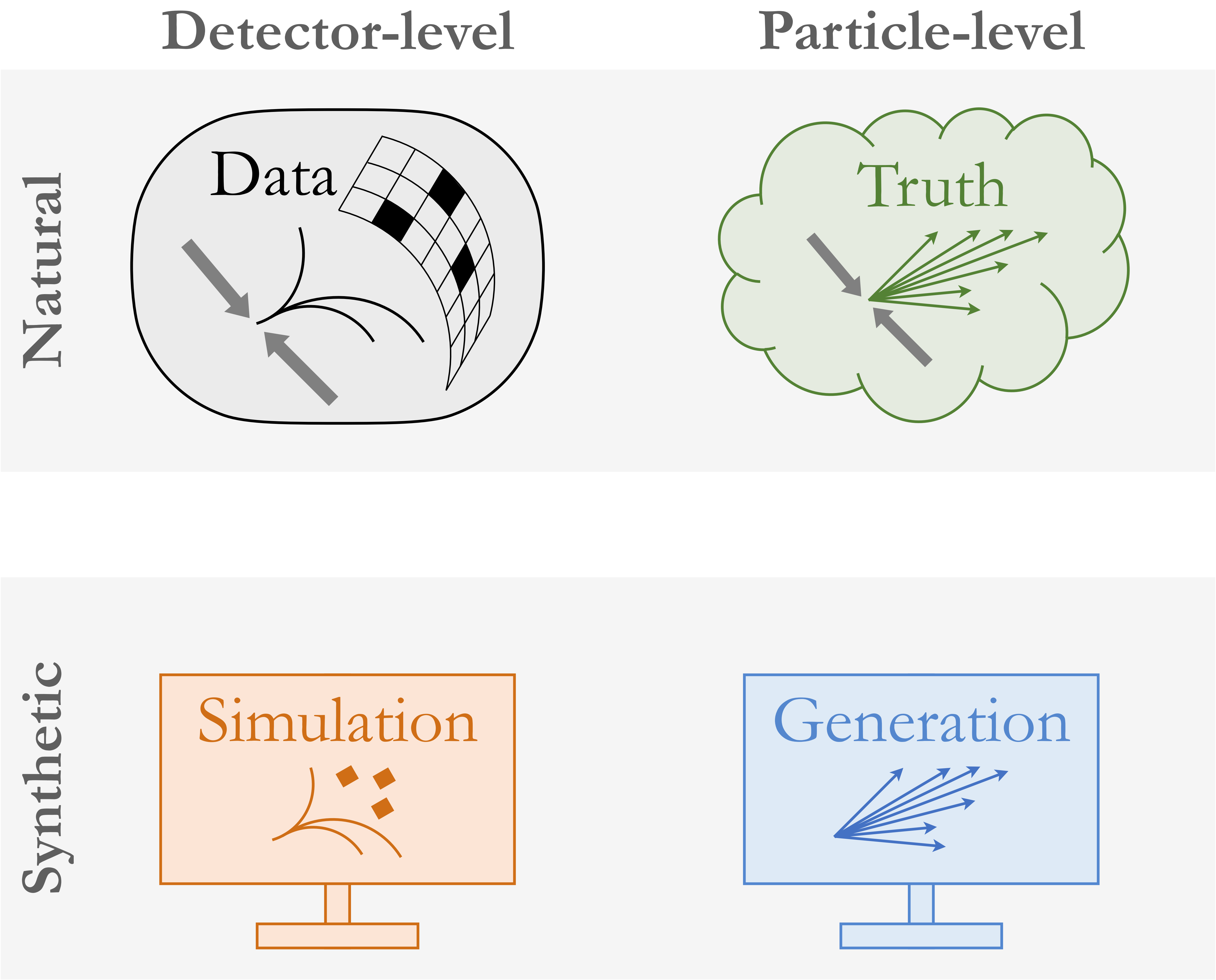
OmniFold

Unfolding Beyond Observables

OmniFold Algorithm – Schematic

[Andreassen, **PTK**, Metodiev, Nachman, Thaler, PRL 2020]

OmniFold weights particle-level Gen to be consistent with Data once passed through the detector



OmniFold Algorithm – Schematic

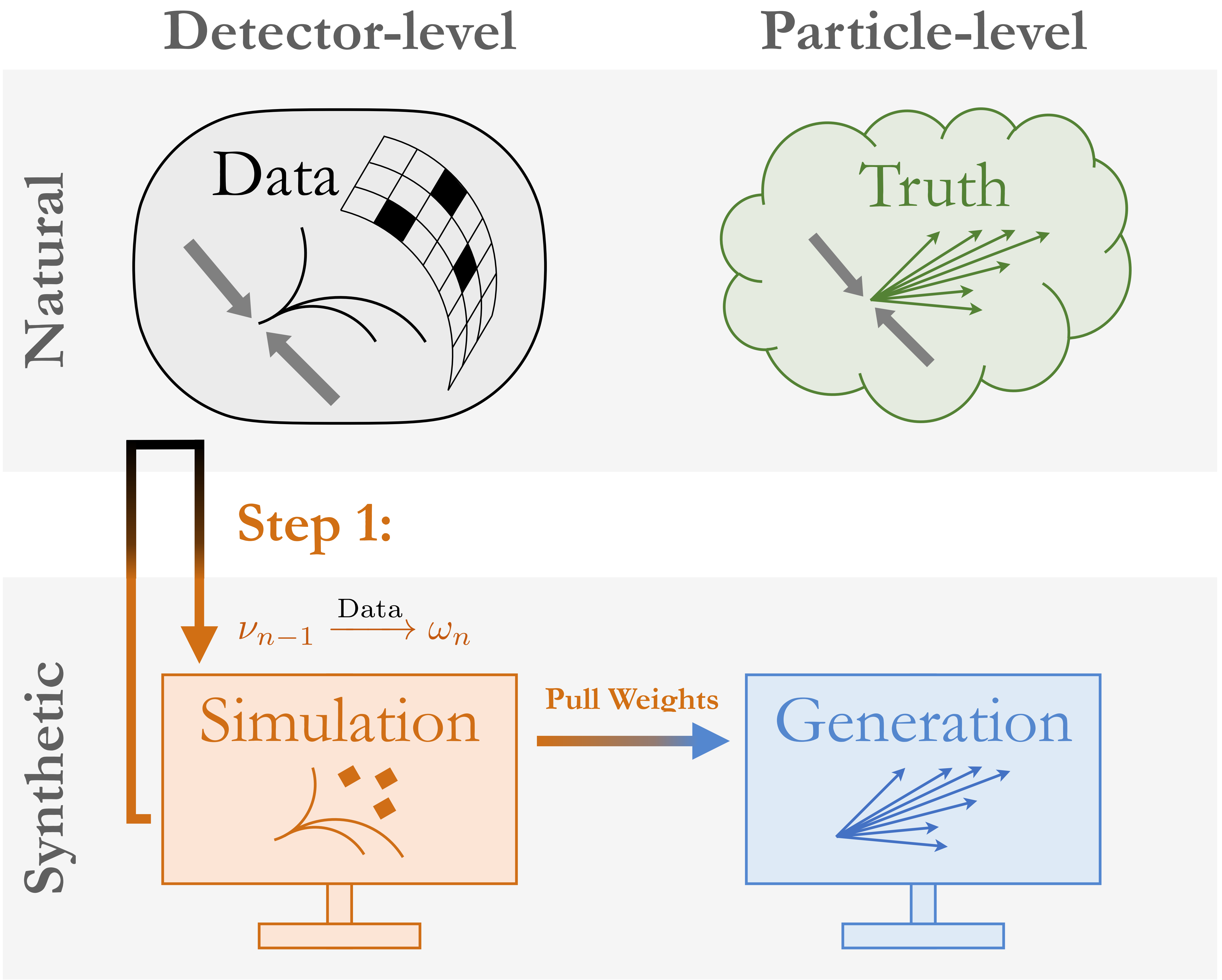
[Andreassen, **PTK**, Metodiev, Nachman, Thaler, PRL 2020]

OmniFold weights particle-level Gen to be consistent with Data once passed through the detector

Step 1

– Reweights Sim_{n-1} to data
– Pulls weights back to particle-level Gen_{n-1}

Incorporates the response matrix



OmniFold Algorithm – Schematic

[Andreassen, **PTK**, Metodiev, Nachman, Thaler, PRL 2020]

OmniFold weights particle-level Gen to be consistent with Data once passed through the detector

Step 1

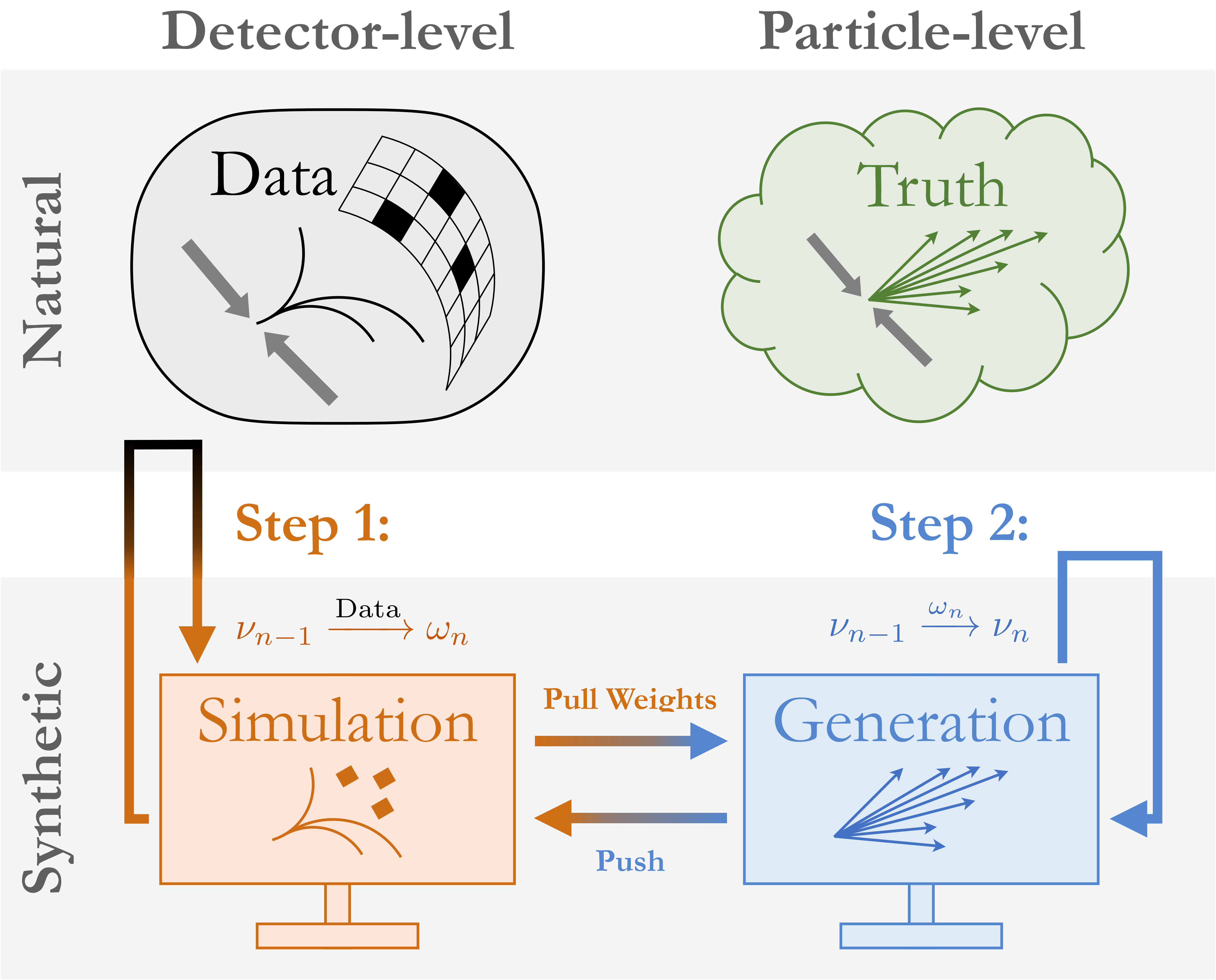
– Reweights Sim_{n-1} to data
– Pulls weights back to particle-level Gen_{n-1}

Incorporates the response matrix

Step 2

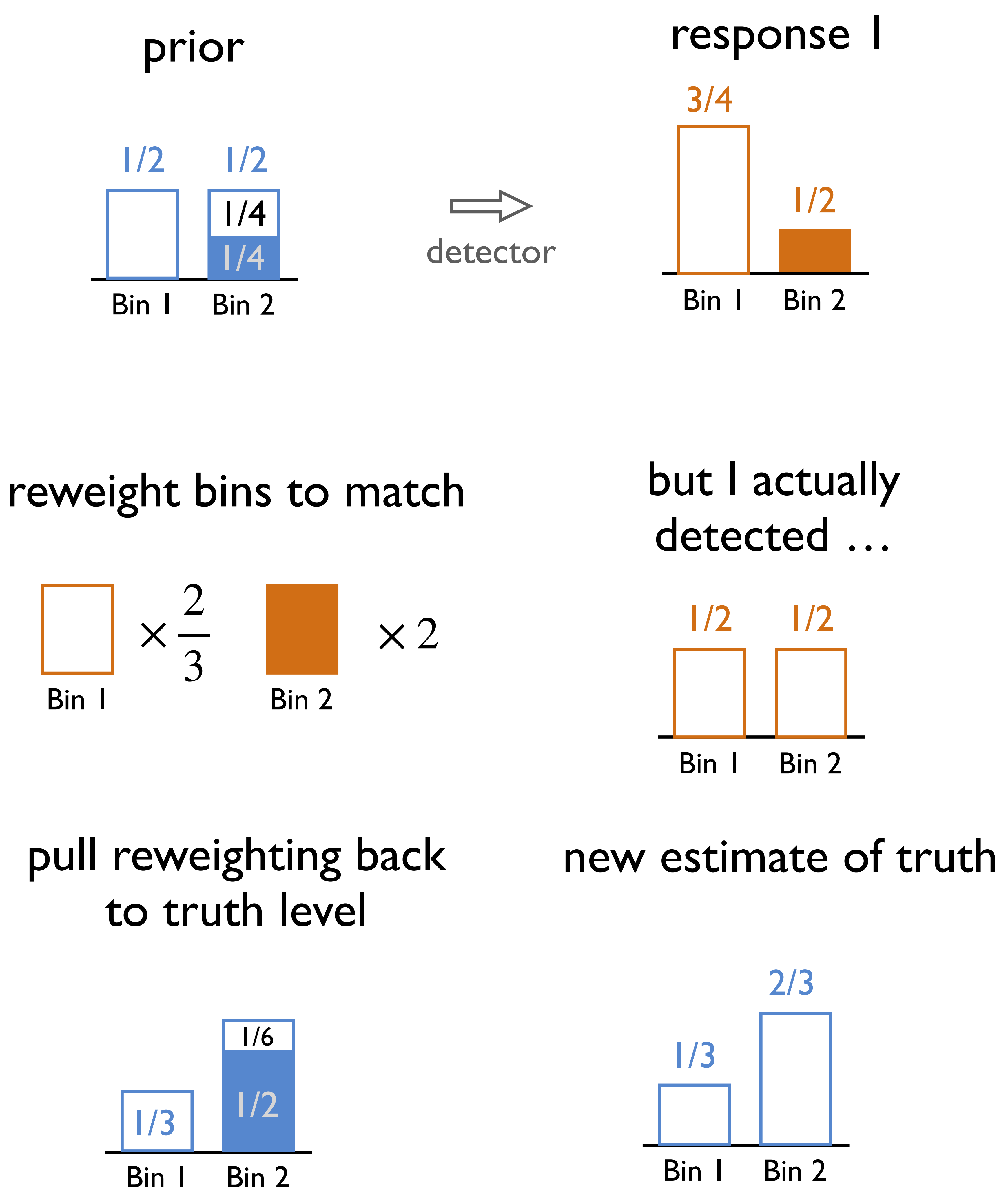
– Reweights Gen_{n-1} to (step 1)-weighted gen_{n-1}
– Pushes weights to detector-level Sim_n

Constructs valid particle-level function by averaging gen-level weights



IBU as Reweighting

[Richardson, JOSA 1972; Lucy, AJ 1974; D'Agostini, NIMPA 1995]



Consider a situation with two particle-level bins and two detector-level bins

$$t_j^{(0)} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_j \qquad m_i = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i \qquad R_{ij} = \begin{pmatrix} 1 & \frac{1}{2} \\ 0 & \frac{1}{2} \end{pmatrix}_{ij}$$

Uniform prior

Bins are measured equally

Bin 1 reconstructed perfectly
Bin 2 reconstructed equally

$$t_j^{(1)} = \sum_i \frac{\begin{pmatrix} \frac{1}{2} & \frac{1}{4} \\ 0 & \frac{1}{4} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{3}{4} & \frac{1}{4} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i = \begin{pmatrix} \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{2}}_{1/3} + 0 \\ \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{4}}_{1/6} + \underbrace{\frac{1}{2} \times 4 \times \frac{1}{4}}_{1/2} \end{pmatrix} = \begin{pmatrix} \frac{1}{3} \\ \frac{2}{3} \end{pmatrix}_j$$

After one iteration

$\vdots$

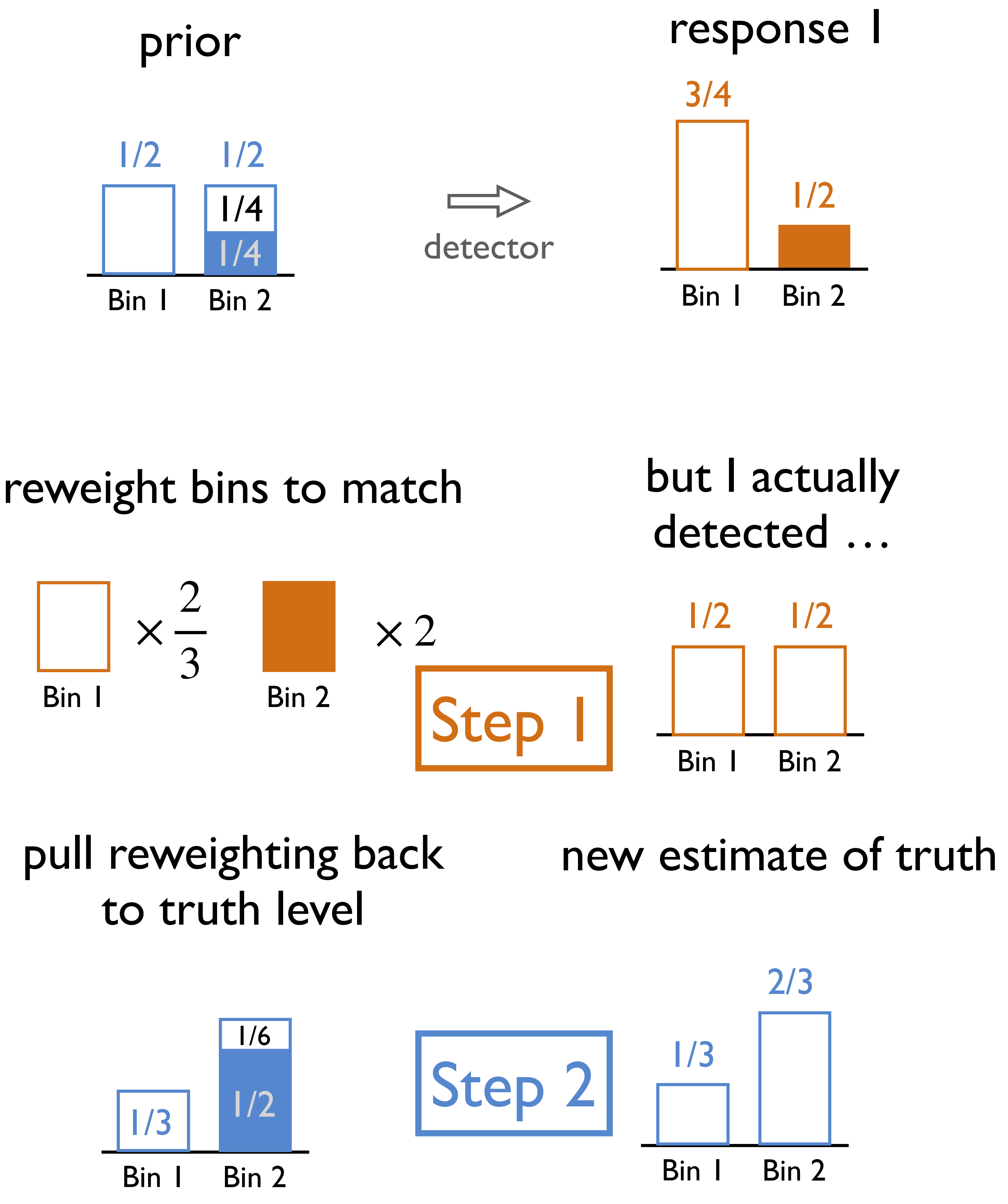
$$t_j^{(n)} = \sum_i \frac{\begin{pmatrix} \frac{1}{n+1} & \frac{n}{2(n+1)} \\ 0 & \frac{n}{2(n+1)} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{n+2}{2(n+1)} & \frac{n}{2(n+1)} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i = \begin{pmatrix} \frac{1}{n+2} \\ \frac{n+1}{n+2} \end{pmatrix}_j \rightarrow \begin{pmatrix} 0 \\ 1 \end{pmatrix}_j$$

Correct truth distribution obtained as $n \rightarrow \infty$

At the n^{th} iteration

IBU as Reweighting

[Richardson, JOSA 1972; Lucy, AJ 1974; D'Agostini, NIMPA 1995]



Consider a situation with two particle-level bins and two detector-level bins

$$t_j^{(0)} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_j \qquad m_i = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i \qquad R_{ij} = \begin{pmatrix} 1 & \frac{1}{2} \\ 0 & \frac{1}{2} \end{pmatrix}_{ij}$$

Uniform prior

Bins are measured equally

Bin 1 reconstructed perfectly
Bin 2 reconstructed equally

$$t_j^{(1)} = \sum_i \frac{\begin{pmatrix} \frac{1}{2} & \frac{1}{4} \\ 0 & \frac{1}{4} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{3}{4} & \frac{1}{4} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i = \begin{pmatrix} \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{2} + 0}_{1/3} \\ \underbrace{\frac{1}{2} \times \frac{4}{3} \times \frac{1}{4}}_{1/6} + \underbrace{\frac{1}{2} \times 4 \times \frac{1}{4}}_{1/2} \end{pmatrix} = \begin{pmatrix} \frac{1}{3} \\ \frac{2}{3} \end{pmatrix}_j$$

After one iteration

$\vdots$

$$t_j^{(n)} = \sum_i \frac{\begin{pmatrix} \frac{1}{n+1} & \frac{n}{2(n+1)} \\ 0 & \frac{n}{2(n+1)} \end{pmatrix}_{ij}}{\begin{pmatrix} \frac{n+2}{2(n+1)} & \frac{n}{2(n+1)} \end{pmatrix}_i} \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}_i = \begin{pmatrix} \frac{1}{n+2} \\ \frac{n+1}{n+2} \end{pmatrix}_j \to \begin{pmatrix} 0 \\ 1 \end{pmatrix}_j$$

At the n^th iteration

Correct truth distribution obtained as $n \to \infty$

Unfolding via Likelihood Reweighting

Likelihood ratio is optimal binary classifier by Neyman-Pearson lemma

$$L[(w, X), (w', X')](x) = \frac{p_{(w,X)}(x)}{p_{(w',X')}(x)}$$

L – likelihood ratio
w – weights
X – phase space
x – element of X
p – probability density

Unfolding via Likelihood Reweighting

Likelihood ratio is optimal binary classifier by Neyman-Pearson lemma

$$L[(w,X),(w',X')](x) = \frac{p_{(w,X)}(x)}{p_{(w',X')}(x)}$$

L – likelihood ratio
w – weights
X – phase space
x – element of X
p – probability density

Model output of a well-trained classifier accesses likelihood ratio

$$\text{Model}[(w,X),(w',X')](x) \simeq \frac{L[(w,X),(w',X')](x)}{1+L[(w,X),(w',X')](x)}$$

Assuming softmax output

[Cranmer, Pavez, Louppe, 1506.02169; Andreassen, Nachman, PRD 2020]

Unfolding via Likelihood Reweighting

Likelihood ratio is optimal binary classifier by Neyman-Pearson lemma

$$L[(w,X),(w',X')](x) = \frac{p_{(w,X)}(x)}{p_{(w',X')}(x)}$$

L – likelihood ratio
w – weights
X – phase space
x – element of X
p – probability density

Model output of a well-trained classifier accesses likelihood ratio

$$\text{Model}[(w,X),(w',X')](x) \simeq \frac{L[(w,X),(w',X')](x)}{1+L[(w,X),(w',X')](x)}$$

Assuming softmax output

[Cranmer, Pavez, Louppe, 1506.02169; Andreassen, Nachman, PRD 2020]

OmniFold repeatedly reweights one weighted sample (A) to another (B)

$$w_{A'}(x) = w_A(x) \times \frac{\text{Model}[(w_B,B),(w_A,A)](x)}{1-\text{Model}[(w_B,B),(w_A,A)](x)}$$

A' is statistically indistinguishable from B

Likelihood reweighting benefits from architectural improvements

OmniFold Algorithm – Equations

[Andreassen, **PTK**, Metodiev, Nachman, Thaler, PRL 2020]

Inputs

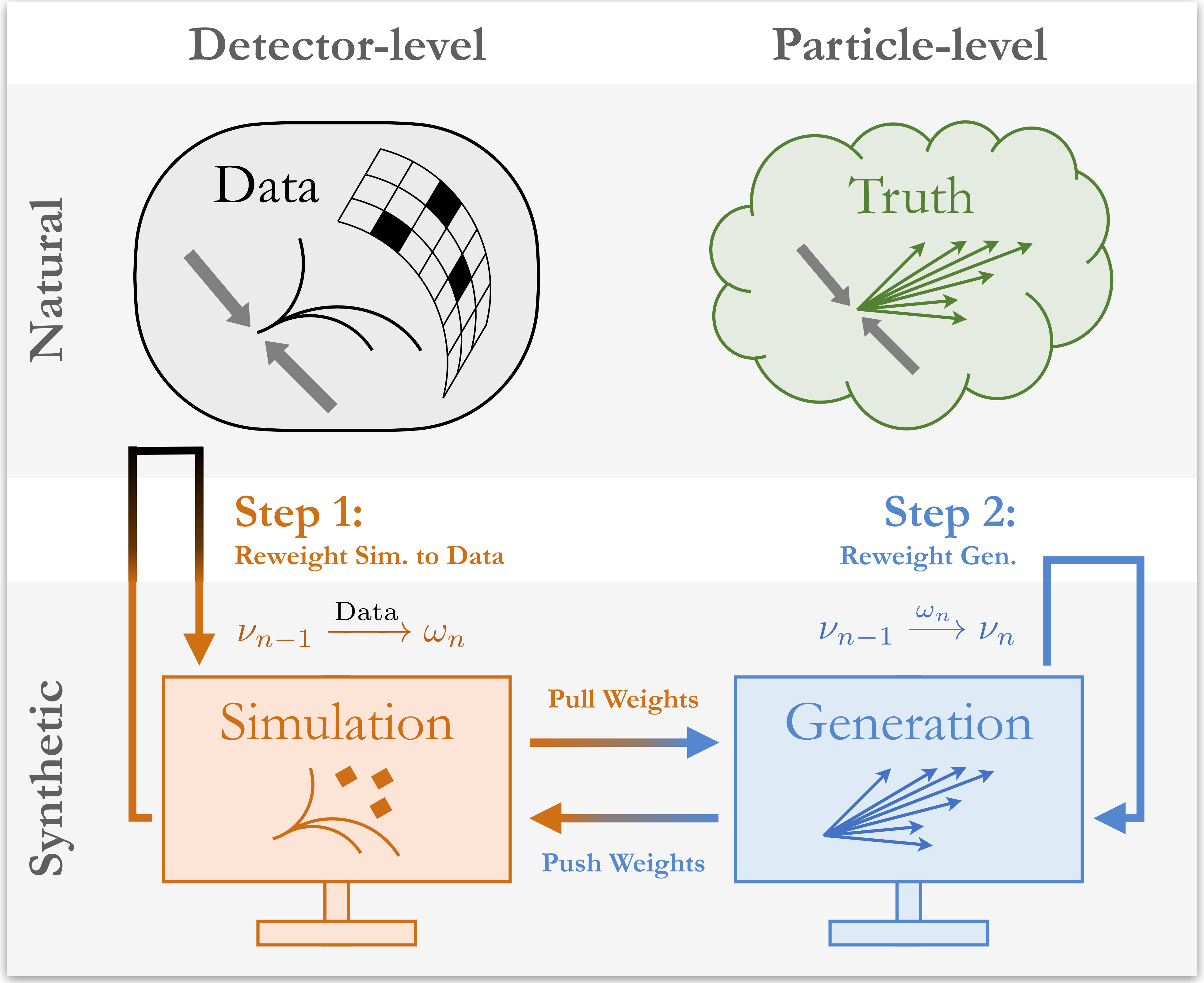
(t, m) – pairs of Gen and Sim events
$\nu_0(t)$ – initial particle-level weights for Gen
– Data

Results of Steps 1 and 2

$\nu_n(t)$ – particle-level weights for Gen, n$^{\text{th}}$ iteration
$\omega_n(m)$ – detector-level weights for Sim, n$^{\text{th}}$ iteration

Pulling/Pushing Weights

$\omega_n^{\text{pull}}(t) = \omega_n(m)$ – pulling ω_n back to particle-level
$\nu_n^{\text{push}}(m) = \nu_n(t)$ – pushing ν_n to detector-level



OmniFold Algorithm – Equations

[Andreassen, **PTK**, Metodiev, Nachman, Thaler, PRL 2020]

Inputs

(t, m) – pairs of Gen and Sim events
$\nu_0(t)$ – initial particle-level weights for Gen
– Data

Results of Steps 1 and 2

$\nu_n(t)$ – particle-level weights for Gen, n[th] iteration
$\omega_n(m)$ – detector-level weights for Sim, n[th] iteration

Pulling/Pushing Weights

$\omega_n^{\text{pull}}(t) = \omega_n(m)$ – pulling ω_n back to particle-level
$\nu_n^{\text{push}}(m) = \nu_n(t)$ – pushing ν_n to detector-level

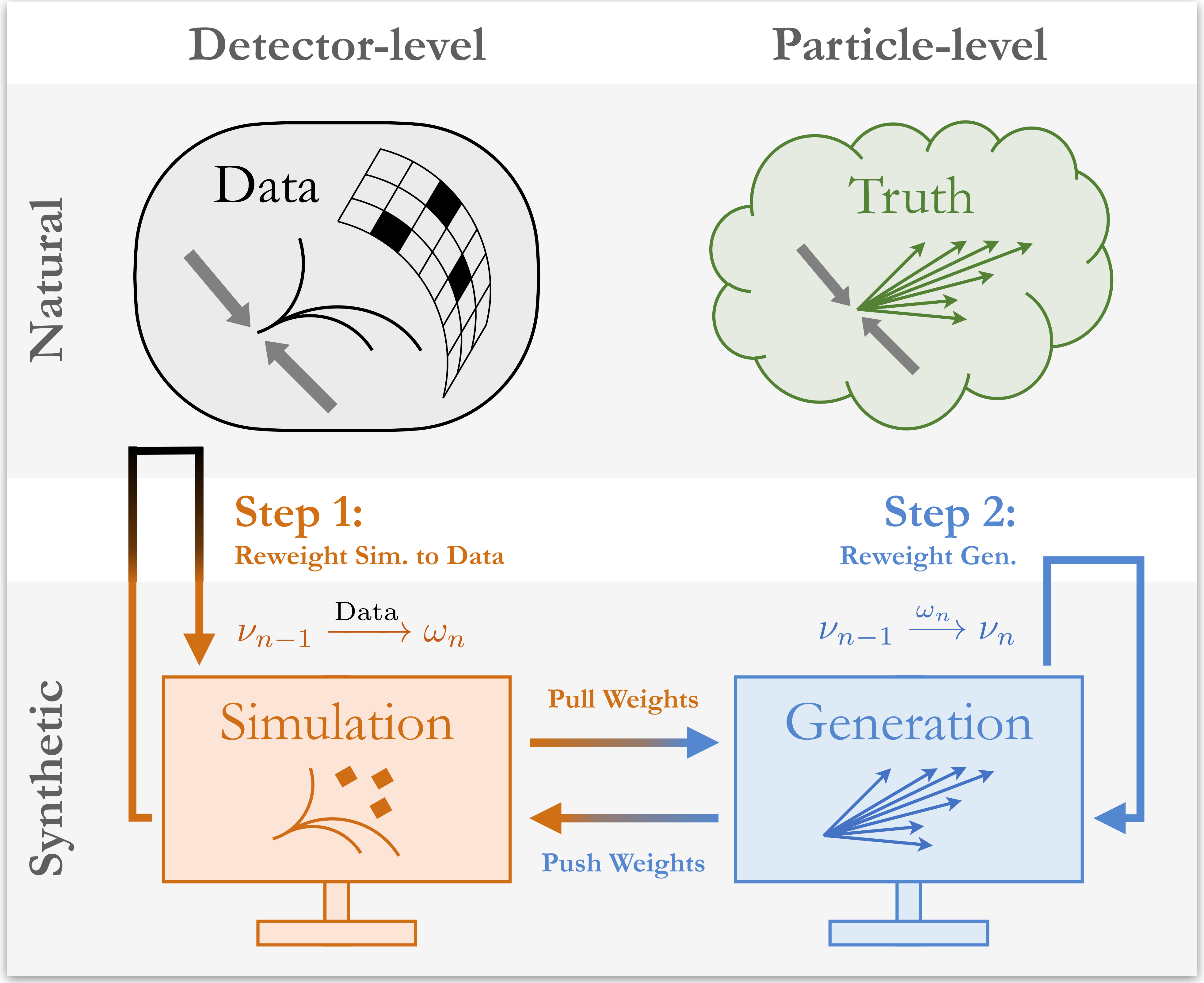
OmniFold

Step 1 – $\omega_n(m) = \nu_{n-1}^{\text{push}} \times L[(1, \text{Data}), (\nu_{n-1}^{\text{push}}, \text{Sim})](m)$

Step 2 – $\nu_n(t) = \nu_{n-1}(t) \times L[(\omega_n^{\text{pull}}, \text{Gen}), (\nu_{n-1}, \text{Gen})](t)$

Unfold *any** observable $p_{\text{Gen}}(t)$ using universal weights $\nu_n(t)$

$$p_{\text{unfolded}}^{(n)}(t) = \nu_n(t) \times p_{\text{Gen}}(t)$$



*Observables should be chosen responsibly

OmniFold Algorithm – Equations

[Andreassen, **PTK**, Metodiev, Nachman, Thaler, PRL 2020]

Inputs

(t, m) – pairs of Gen and Sim events
$\nu_0(t)$ – initial particle-level weights for Gen
– Data

Results of Steps 1 and 2

$\nu_n(t)$ – particle-level weights for Gen, n^th iteration
$\omega_n(m)$ – detector-level weights for Sim, n^th iteration

Pulling/Pushing Weights

$\omega_n^{\text{pull}}(t) = \omega_n(m)$ – pulling ω_n back to particle-level
$\nu_n^{\text{push}}(m) = \nu_n(t)$ – pushing ν_n to detector-level

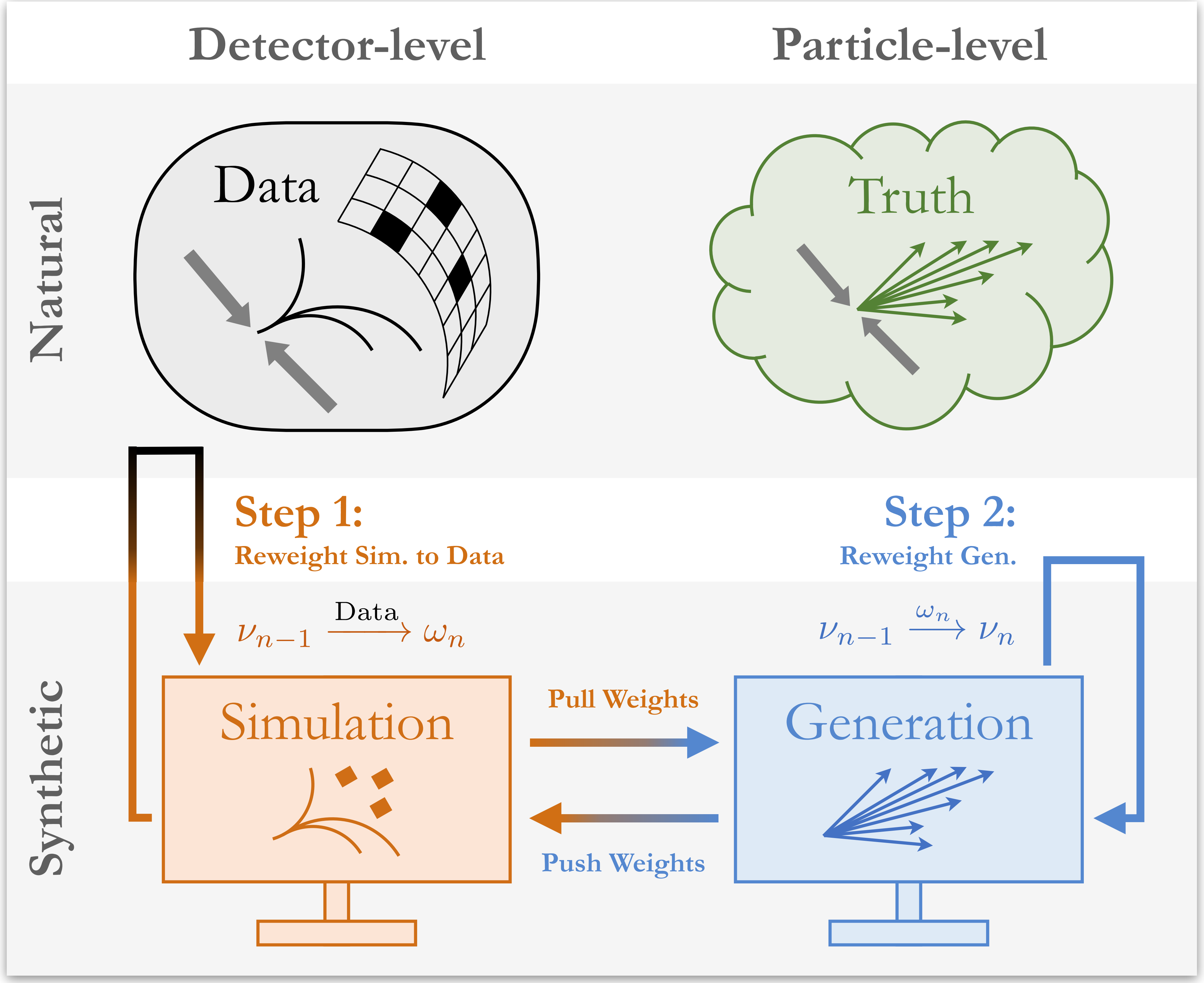
OmniFold

Step 1 – $\omega_n(m) = \nu_{n-1}^{\text{push}} \times L[(1, \text{Data}), (\nu_{n-1}^{\text{push}}, \text{Sim})](m)$

Step 2 – $\nu_n(t) = \nu_{n-1}(t) \times L[(\omega_n^{\text{pull}}, \text{Gen}), (\nu_{n-1}, \text{Gen})](t)$

Unfold *any** observable $p_{\text{Gen}}(t)$ using universal weights $\nu_n(t)$

$$\boxed{p_{\text{unfolded}}^{(n)}(t) = \nu_n(t) \times p_{\text{Gen}}(t)}$$



OmniFold is continuous IBU!

After first iteration, with $\nu_0(t) = 1$:

$$\nu_1(t) p_{\text{Gen}}(t) = \int dm\, p_{\text{Gen|Sim}}(t|m)\, p_{\text{Data}}(m)$$

*Observables should be chosen responsibly

Constructing High-Dimensional Classifiers

How to represent jets to a machine learning architecture?

An unordered, variable length collection of particles

Due to quantum-mechanical indistinguishability

Due to probabilistic nature of jet formation

$$J\left(\{p_1^\mu, \ldots, p_M^\mu\}\right) = J(\{p_{\pi(1)}^\mu, \ldots, p_{\pi(M)}^\mu\}), \qquad \underbrace{M \geq 1}_{\text{Multiplicity}} \;, \qquad \underbrace{\forall\, \pi \in S_M}_{\text{Permutations}}$$

p_i^μ represents *all* the particle properties:

- Four-momentum – $(E, p_x, p_y, p_z)_i^\mu$
- Other quantum numbers (e.g. particle id, charge)
- Experimental information (e.g. vertex info, quality criteria, tracking info)

Methods for processing point clouds/jets should respect the appropriate symmetries

Machine Learning for Point Clouds – Deep Sets

A general permutation-symmetric function is additive in a latent space

Deep Sets

[1703.06114]

Manzil Zaheer[1,2], **Satwik Kottur**[1], **Siamak Ravanbhakhsh**[1],
Barnabás Póczos[1], **Ruslan Salakhutdinov**[1], **Alexander J Smola**[1,2]
[1] Carnegie Mellon University [2] Amazon Web Services

Deep Sets Theorem [63]. *Let $\mathfrak{X} \subset \mathbb{R}^d$ be compact, $X \subset 2^{\mathfrak{X}}$ be the space of sets with bounded cardinality of elements in $\mathfrak{X}$, and $Y \subset \mathbb{R}$ be a bounded interval. Consider a continuous function $f : X \to Y$ that is invariant under permutations of its inputs, i.e. $f(x_1, \ldots, x_M) = f(x_{\pi(1)}, \ldots, x_{\pi(M)})$ for all $x_i \in \mathfrak{X}$ and $\pi \in S_M$. Then there exists a sufficiently large integer ℓ and continuous functions $\Phi : \mathfrak{X} \to \mathbb{R}^\ell$, $F : \mathbb{R}^\ell \to Y$ such that the following holds to an arbitrarily good approximation:*[1]

$$f(\{x_1, \ldots, x_M\}) = F\left(\sum_{i=1}^{M} \Phi(x_i)\right)$$

Machine Learning for Point Clouds – Deep Sets

A general permutation-symmetric function is additive in a latent space

Deep Sets

[1703.06114]

Manzil Zaheer[1,2], **Satwik Kottur**[1], **Siamak Ravanbhakhsh**[1],
Barnabás Póczos[1], **Ruslan Salakhutdinov**[1], **Alexander J Smola**[1,2]
[1] Carnegie Mellon University [2] Amazon Web Services

Feature space

Variable length

Deep Sets Theorem [63]. *Let $\boxed{\mathfrak{X} \subset \mathbb{R}^d}$ be compact, $\boxed{X \subset 2^{\mathfrak{X}} \text{ be the space of sets}}$ with bounded cardinality of elements in $\mathfrak{X}$, and $Y \subset \mathbb{R}$ be a bounded interval. Consider a continuous function $f : X \to Y$ that is invariant under permutations of its inputs, i.e. $f(x_1, \ldots, x_M) = \boxed{f(x_{\pi(1)}, \ldots, x_{\pi(M)}) \text{ for all } x_i \in \mathfrak{X} \text{ and } \pi \in S_M}$. Then there exists a sufficiently large integer ℓ and continuous functions $\Phi : \mathfrak{X} \to \boxed{\mathbb{R}^\ell}$, $F : \boxed{\mathbb{R}^\ell} \to Y$ such that the following holds to an arbitrarily good approximation:*[1]

Permutation invariance

Latent space

$$\boxed{f(\{x_1, \ldots, x_M\}) = F\left(\sum_{i=1}^{M} \Phi(x_i)\right)}$$

General parametrization for a function of sets

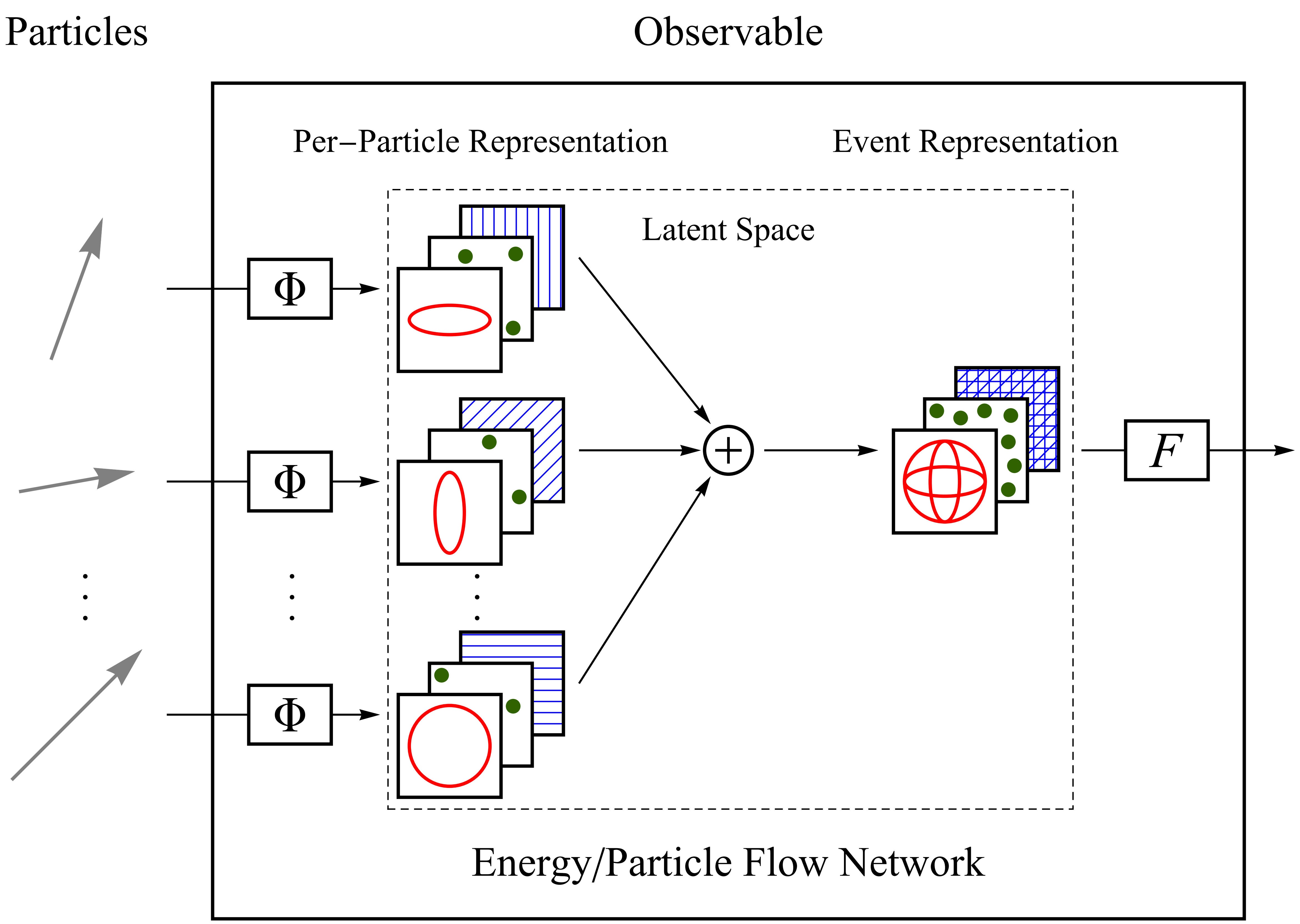
Approximating Φ and F with Neural Networks

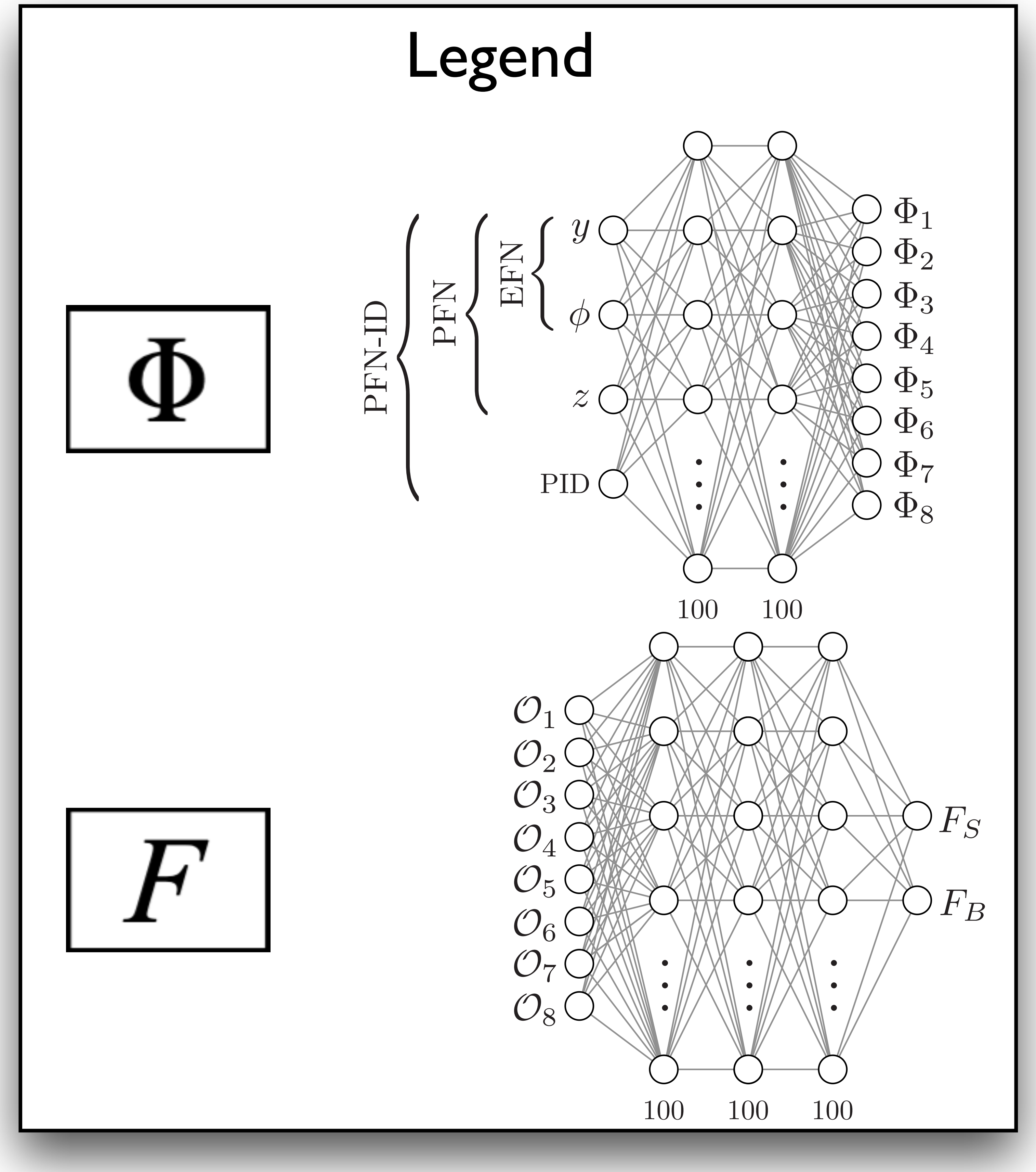
[**PTK**, Metodiev, Thaler, JHEP 2019]

Employ neural networks as arbitrary function approximators

Use fully-connected networks for simplicity

Default sizes – Φ: (100, 100, ℓ), F: (100, 100, 100)





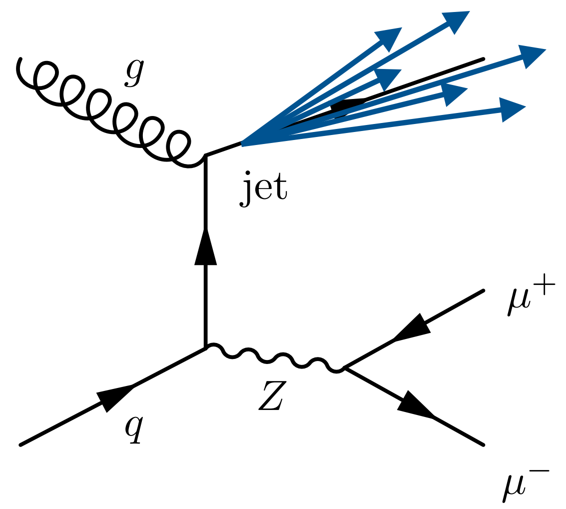
Particle Flow Network (PFN)

$$\text{PFN}(\{p_1^\mu, \ldots, p_M^\mu\}) = F\left(\sum_{i=1}^{M} \Phi(p_i^\mu)\right)$$

Fully general latent space

Testing OmniFold

Ingredients for Z + Jet Case Study



$Z(\to \mu^+\mu^-)$ + Jet Events

"Data" – HERWIG 7.1.5
MC – PYTHIA 8.243, tune 26
1.6 million events each after cuts

Detector Simulation

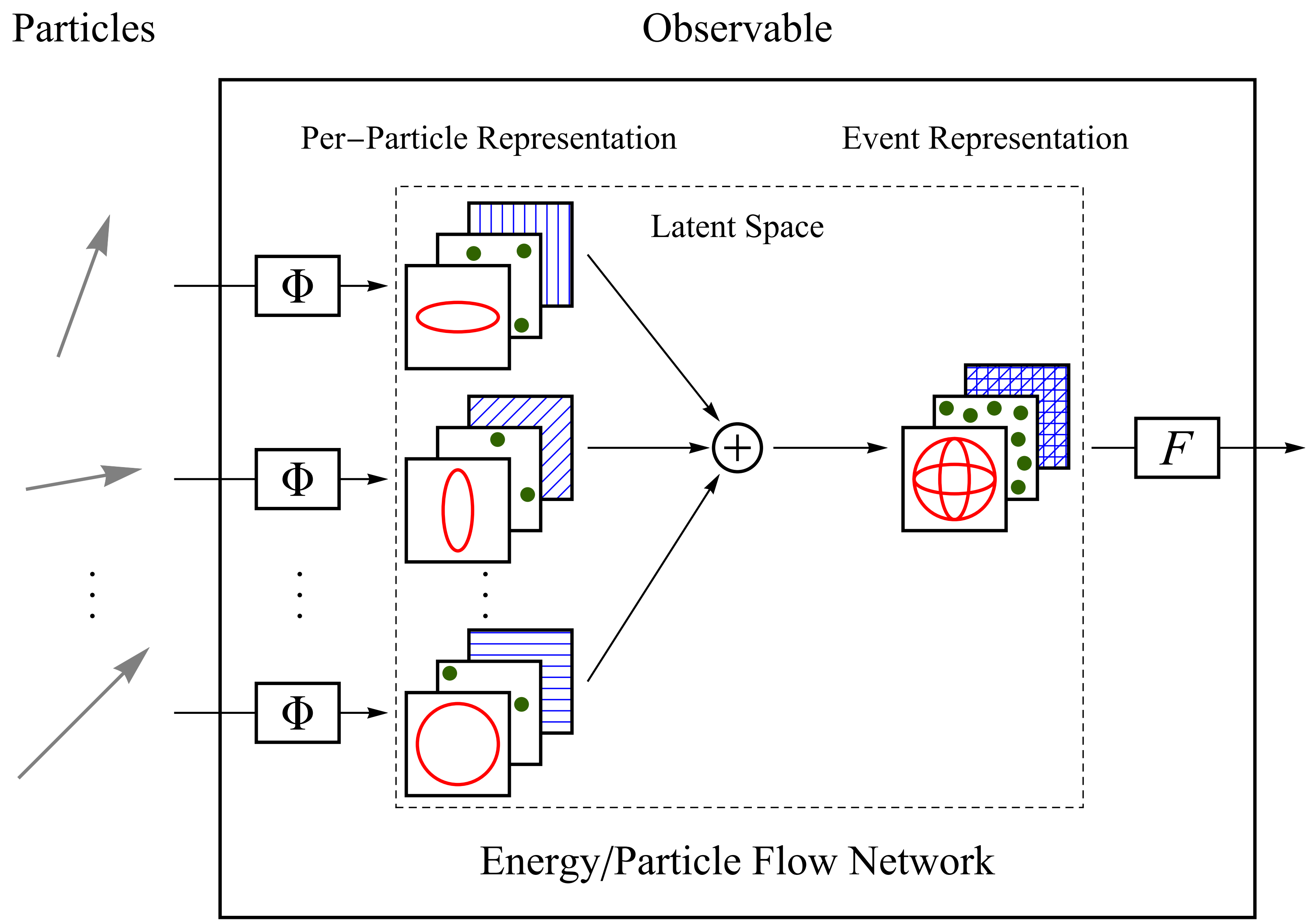
CMS-like detector – DELPHES 3.4.2

Jets

Anti-k_T, $R = 0.4$ – FASTJET 3.3.2
$p_T^Z > 200$ GeV, assume excellent muon detector resolution

Datasets publicly available
- *With two additional Pythia tunes*
- *Accessible via EnergyFlow*

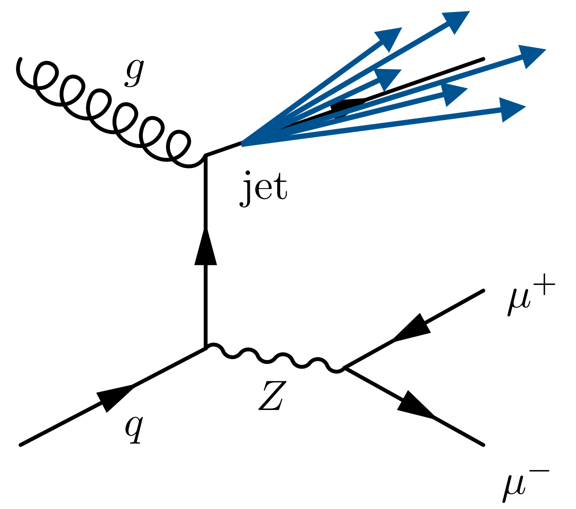
OmniFold Binder Demo



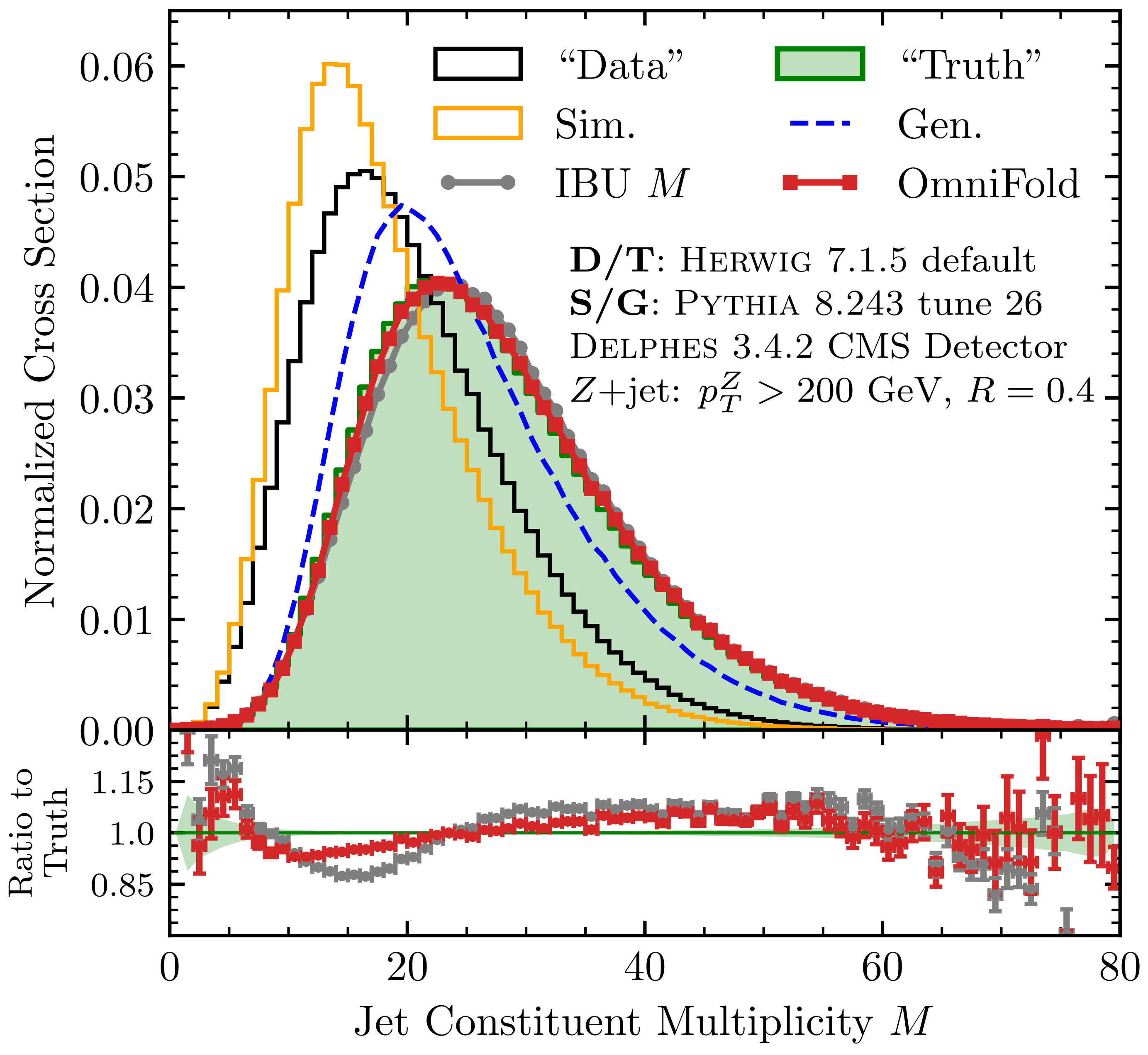
Particle Flow Network (PFN) architecture processes full radiation pattern of the event

- PFN-Ex: $(p_T, y, \phi, \mathrm{PID})$ input features
- Φ : $(100, 100, 256)$ dense layers
- F : $(100,100,100)$ dense layers
- ReLU activations, softmax output
- Categorical cross-entropy loss
- 20% validation sample
- 10 epoch patience

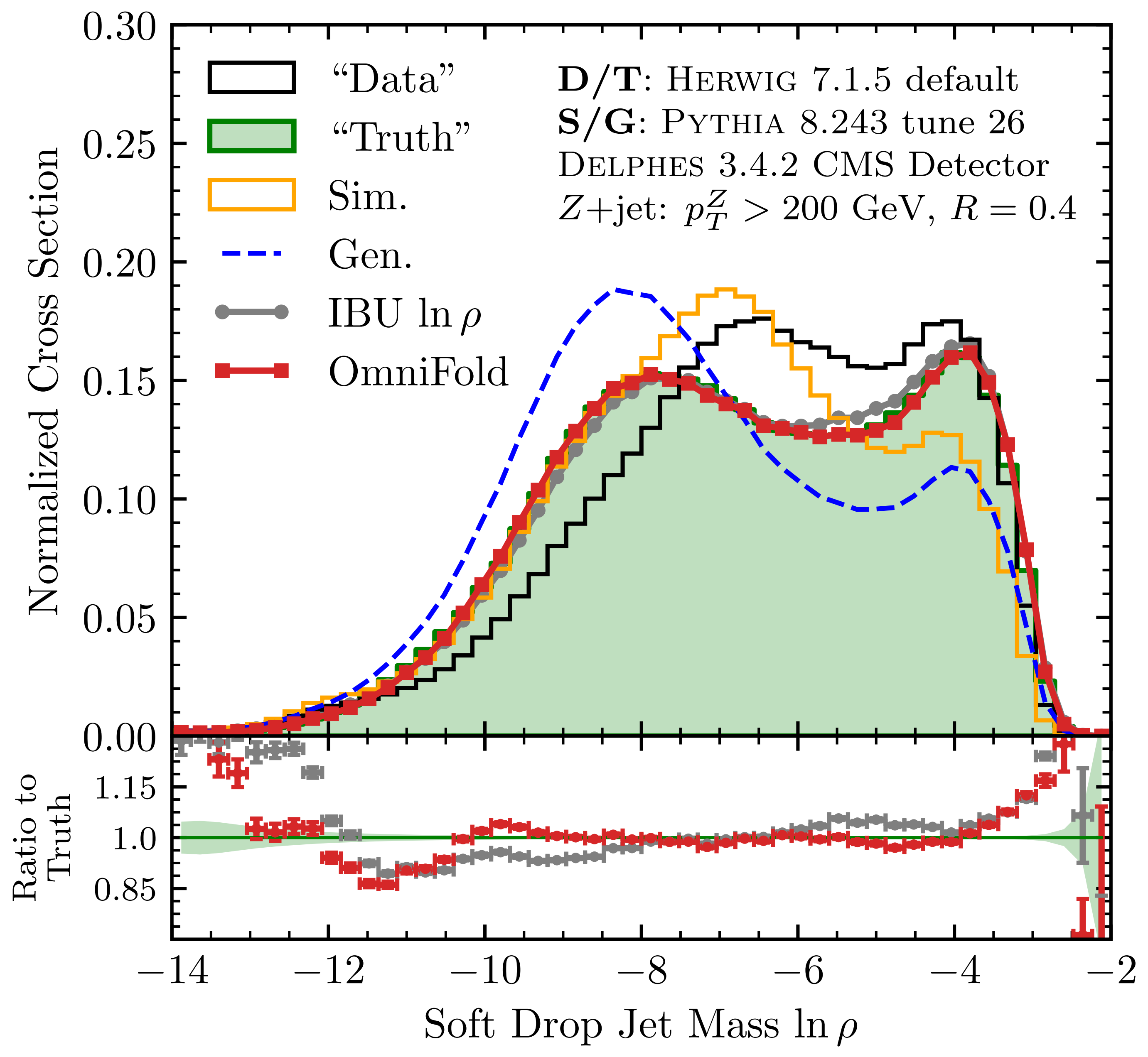
OmniFolding Jet Substructure Observables



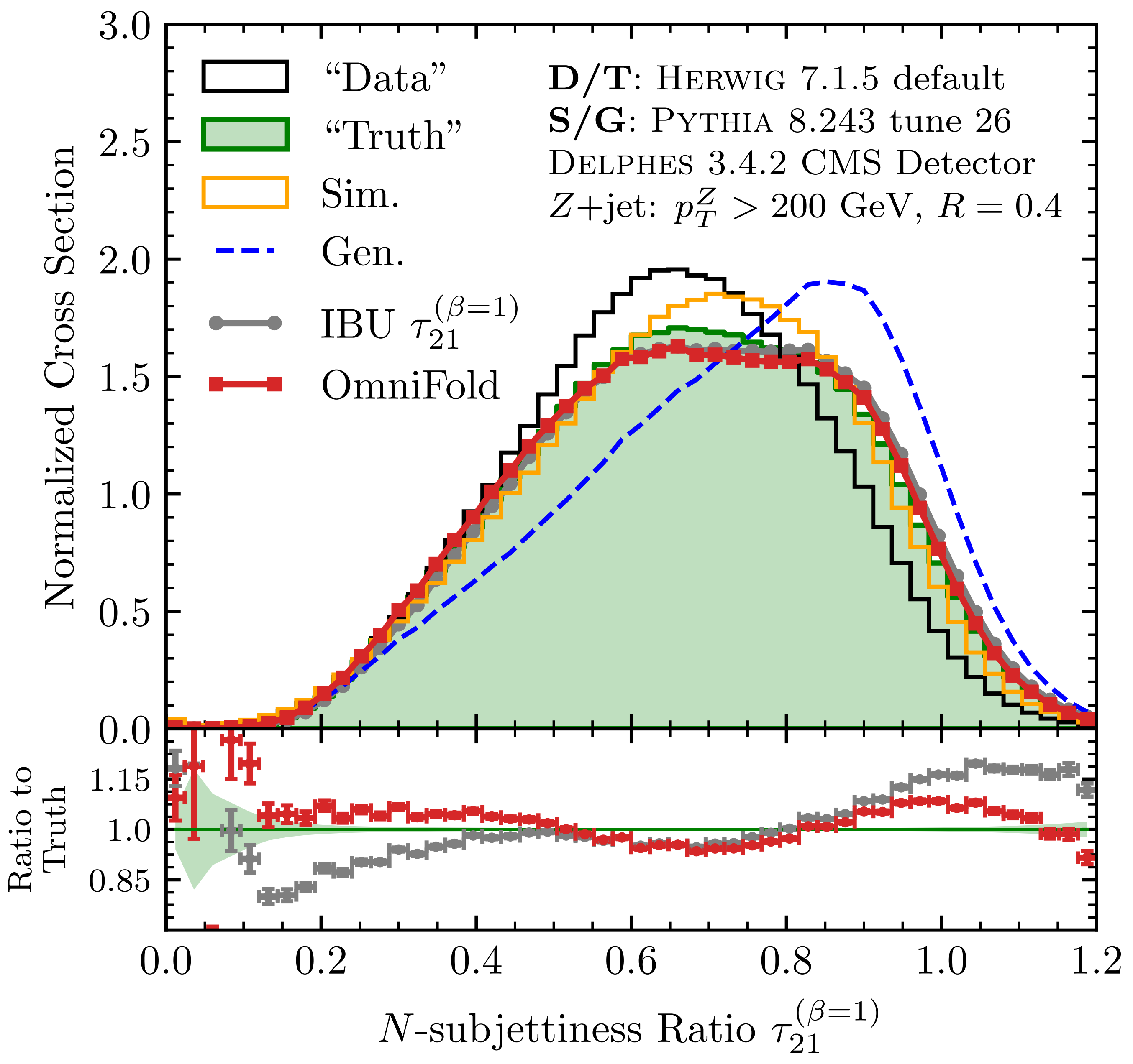
Single OmniFold instantiation vs. repeated applications of IBU



IRC unsafe



IRC safe



Sudakov safe

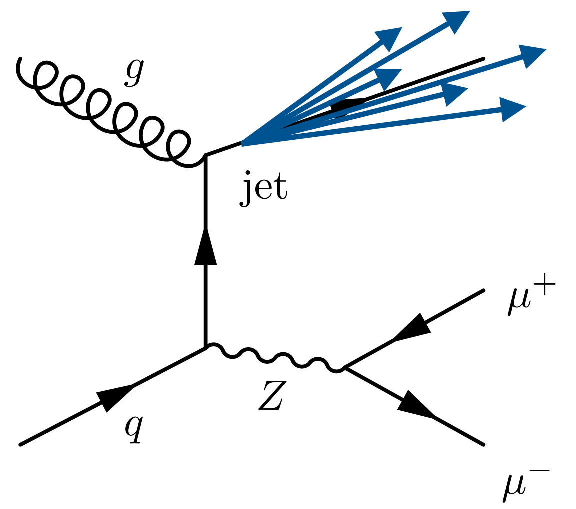
OmniFold equals or outperforms IBU

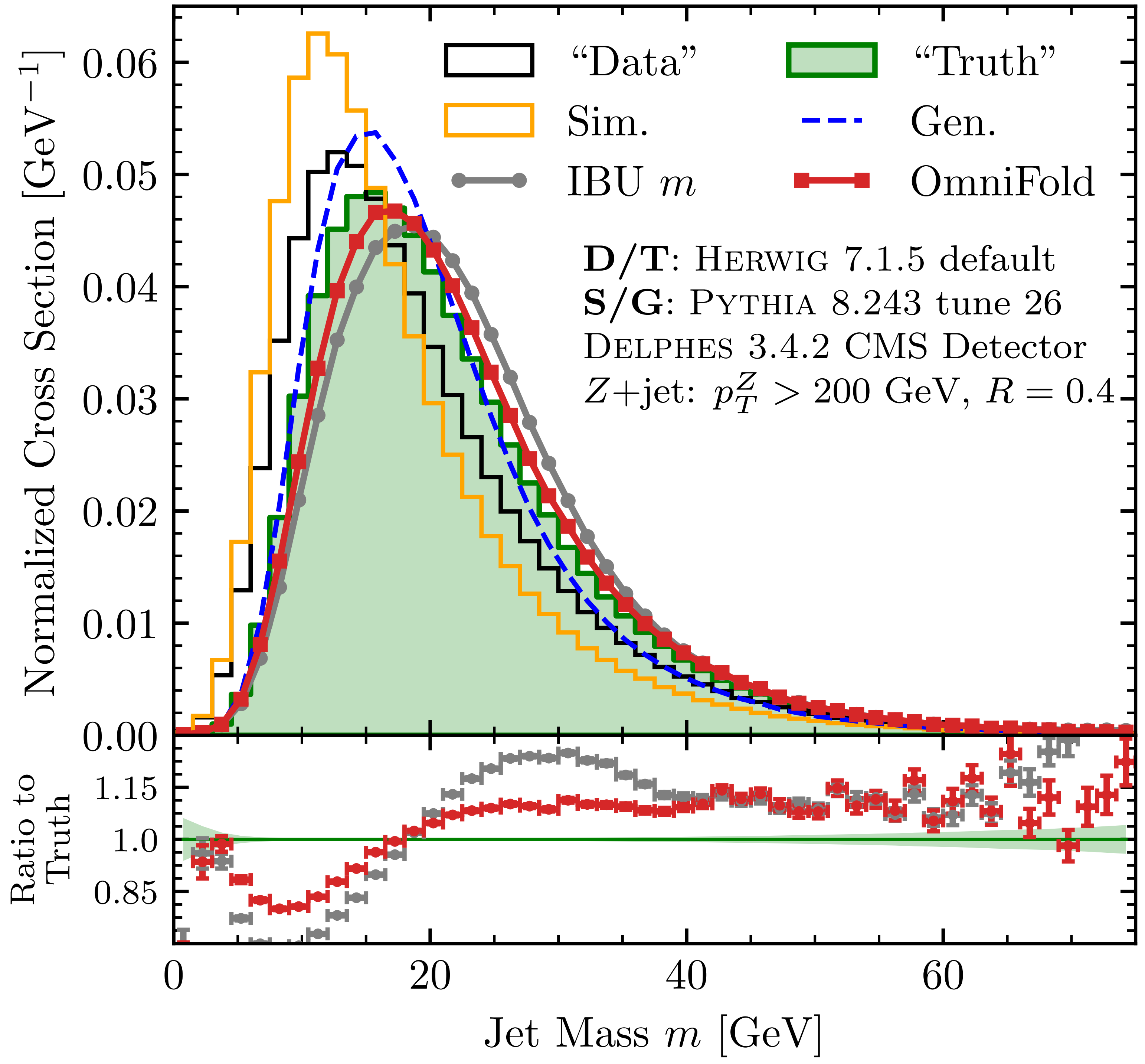
Five unfolding iterations in all cases

Statistical uncertainties on prior shown in ratio

(See backup for more on soft drop)

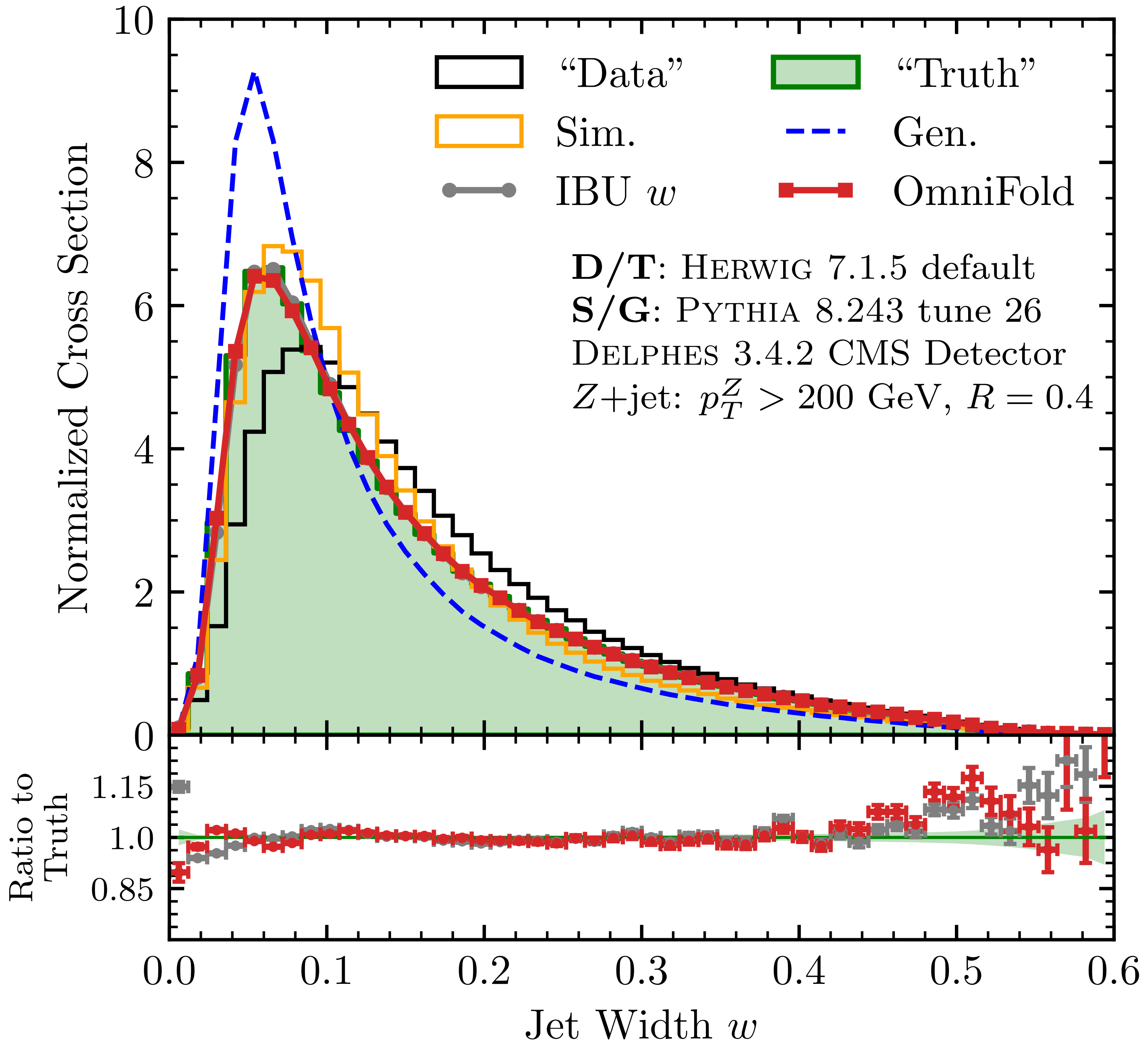
Additional OmniFolded Distributions





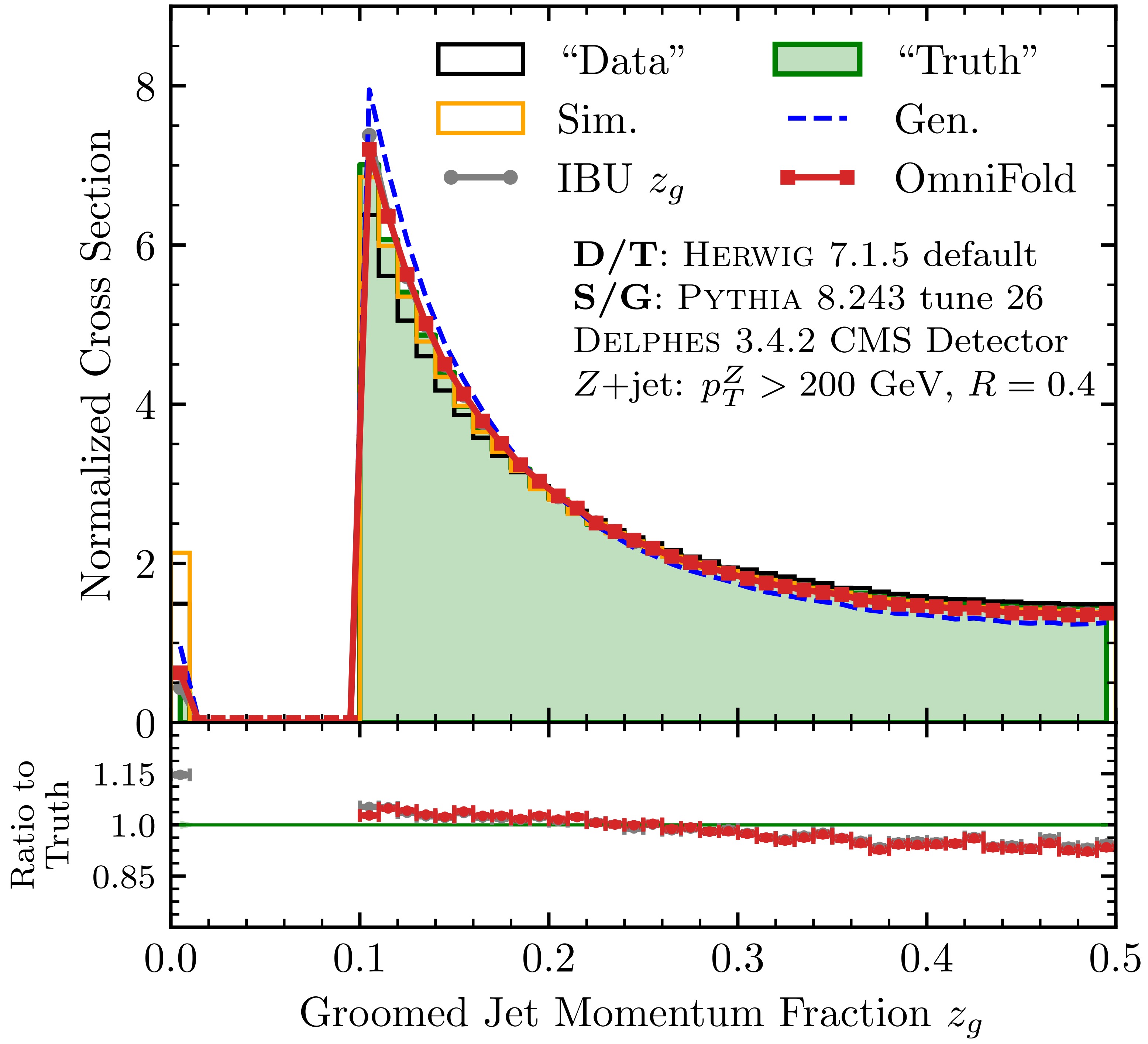
Jet mass affected by particle masses

$$m_J^2 = \left(\sum_{i \in \text{jet}} p_i^\mu\right)^2$$



IRC-safe observables easier to unfold

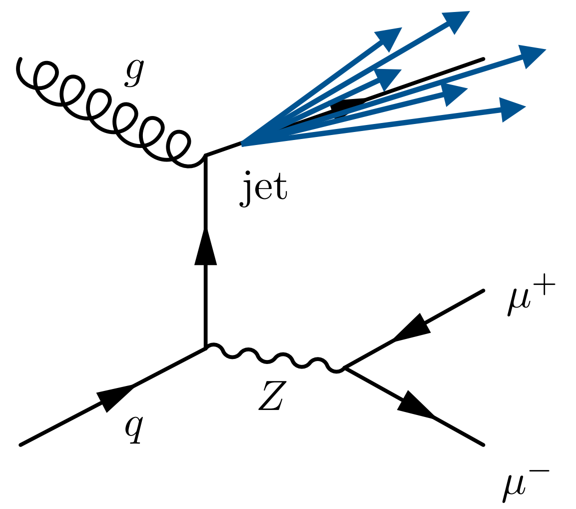
$$w = \frac{1}{\sum_j p_{Tj}} \sum_i p_{Ti} \sqrt{(y_i - y_J)^2 + (\phi_i - \phi_J)^2}$$



z_g remarkably stable under choice of method

z_g = p_T fraction of first splitting to pass soft drop

OmniFold Results by Event Representation



User is free to choose *event representation* in the OmniFold procedure

OmniFold – full phase space information

MultiFold – multiple observables

UniFold – single observable, essentially unbinned IBU

	Observable					
Method	m	M	w	$\ln\rho$	τ_{21}	z_g
OmniFold	**2.77**	**0.33**	0.10	**0.35**	0.53	0.68
MultiFold	3.80	0.89	**0.09**	0.37	**0.26**	**0.15**
UniFold	8.82	1.46	0.15	0.59	1.11	0.59
IBU	9.31	1.51	0.11	0.71	1.10	0.37
Data	24.6	130	15.7	14.2	11.1	3.76
Generation	3.62	15	22.4	19	20.8	3.84
	mass	mult.	width	groomed mass	N-subj. ratio	

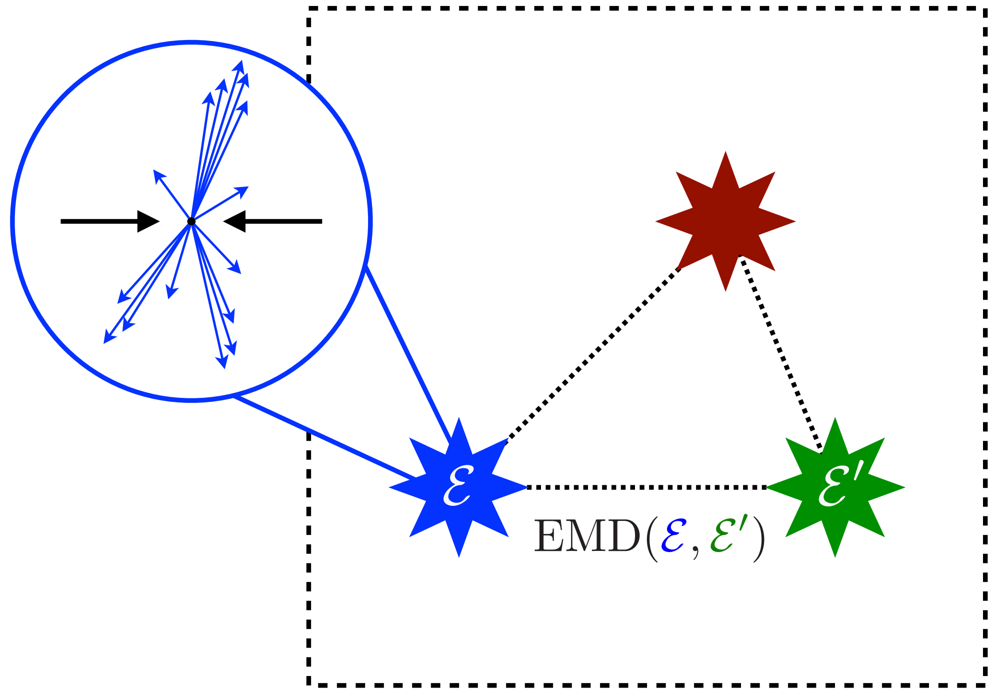
Evaluate performance using triangular discriminator

$$\Delta(p,q) = \frac{1}{2}\int d\lambda\, \frac{(p(\lambda)-q(\lambda))^2}{p(\lambda)+q(\lambda)} (\times 10^3)$$

OmniFold/MultiFold *outperforms IBU on all observables!*

Single MultiFold training based on all six observables

UniFold is similar to or outperforms IBU



Unfolding Setup

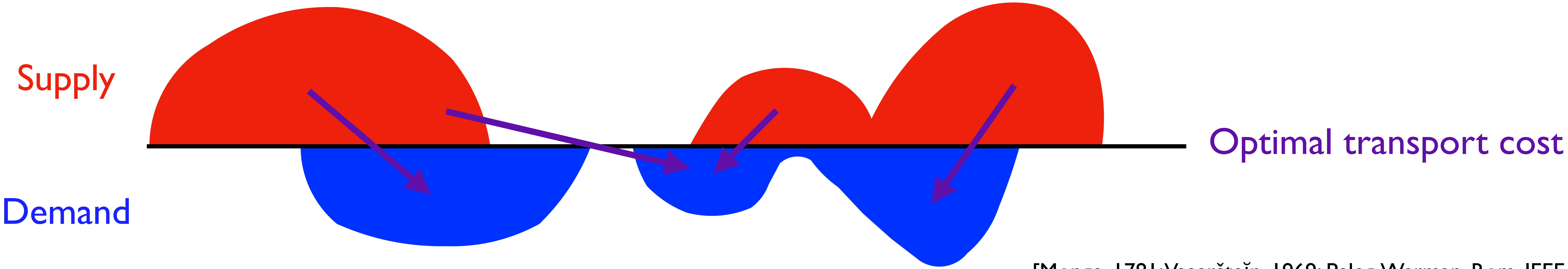
OmniFold

Unfolding Beyond Observables

Optimal Transport in Particle Physics

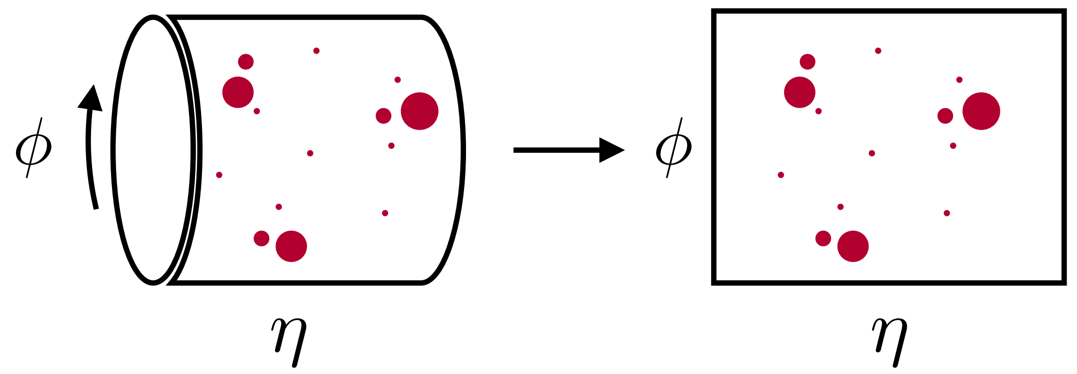
When are Two Distributions Similar?

Optimal transport minimizes the “work" (stuff x distance) required to transport supply to demand



[Monge, 1781; Vaseršteĭn, 1969; Peleg, Werman, Rom, IEEE 1989; Rubner, Tomasi, Guibas, ICCV 1998, ICJV 2000; Pele, Werman, ECCV 2008; Pele, Taskar, GSI 2013]

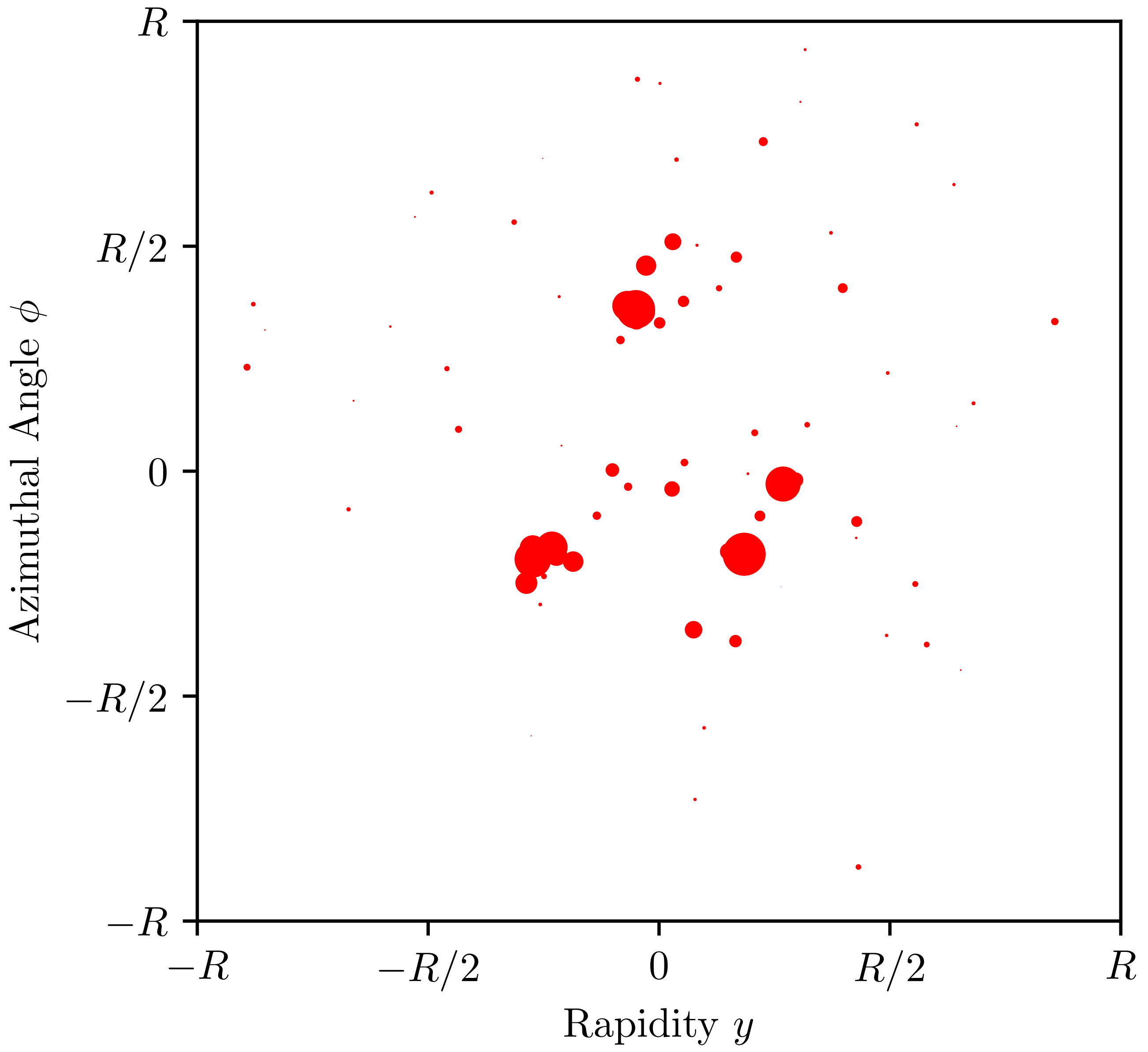
When are Two Events similar?

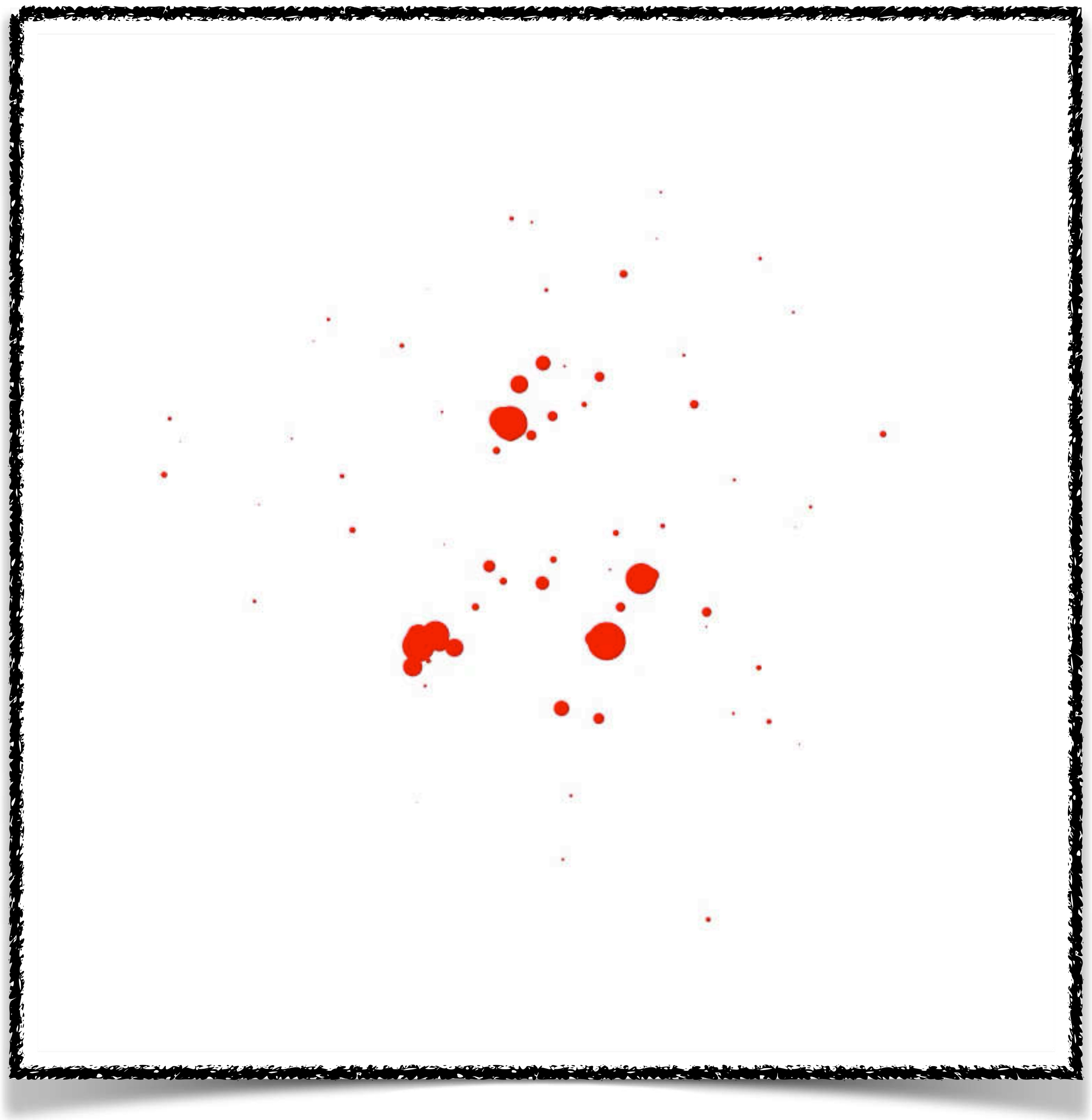


[**PTK**, Metodiev, Thaler, PRL 2019]

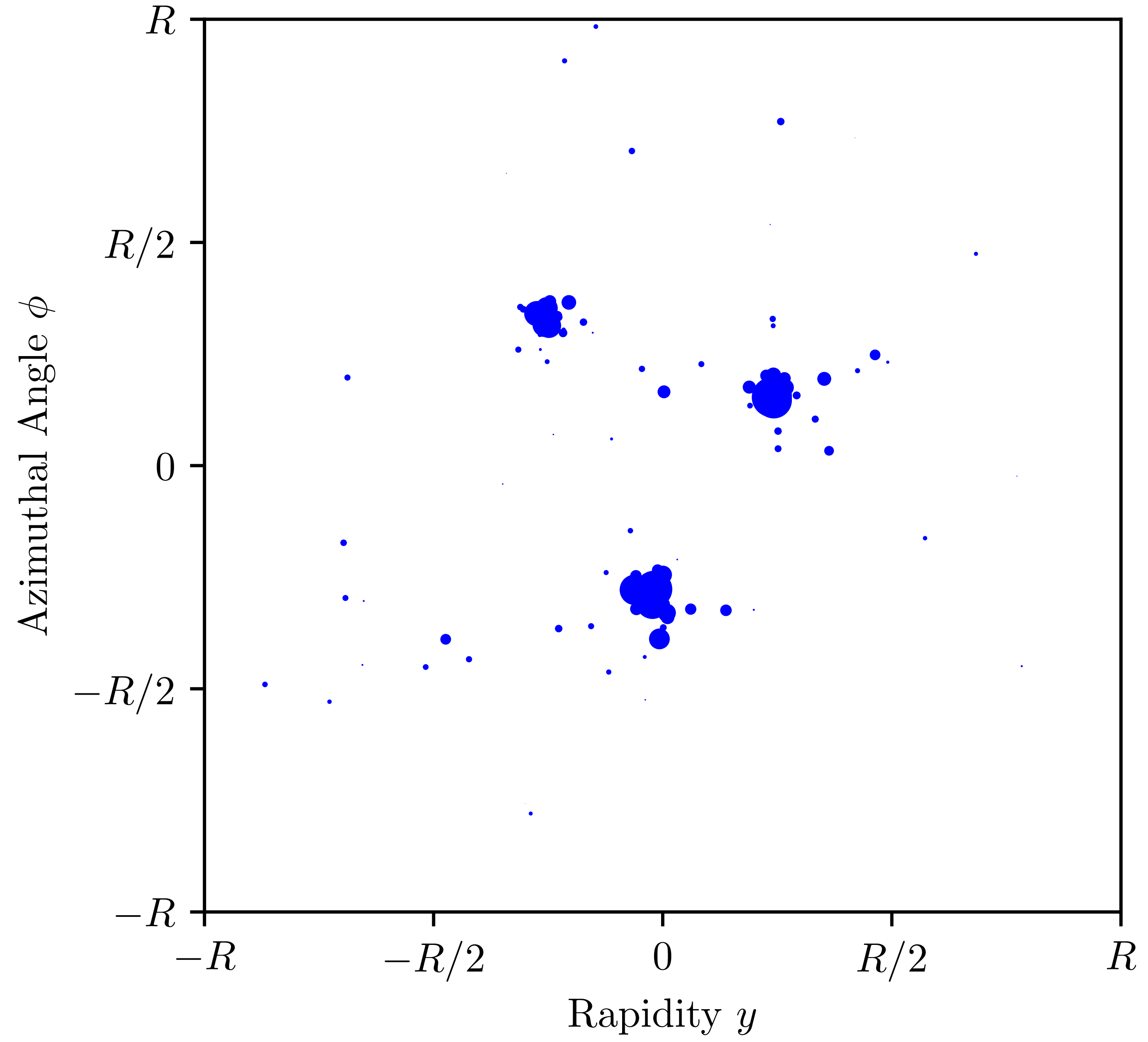
Optimal transport minimizes the "work" (stuff x distance) required to transport supply to demand

Top Jet 1





Top Jet 2



$$\mathcal{E}(\hat{n}) = \sum_{i=1}^{M} E_i\, \delta(\hat{n} - \hat{n}_i)$$

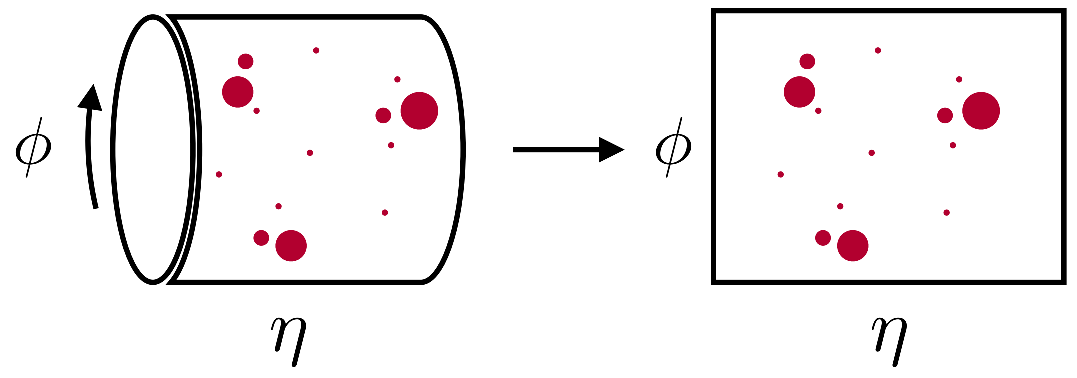
Provides a metric on normalized distributions in a space with a ground distance measure

↳ symmetric, non-negative, triangle inequality, zero iff identical

$$\theta_{ij} = \sqrt{\Delta y_{ij}^2 + \Delta \phi_{ij}^2}$$

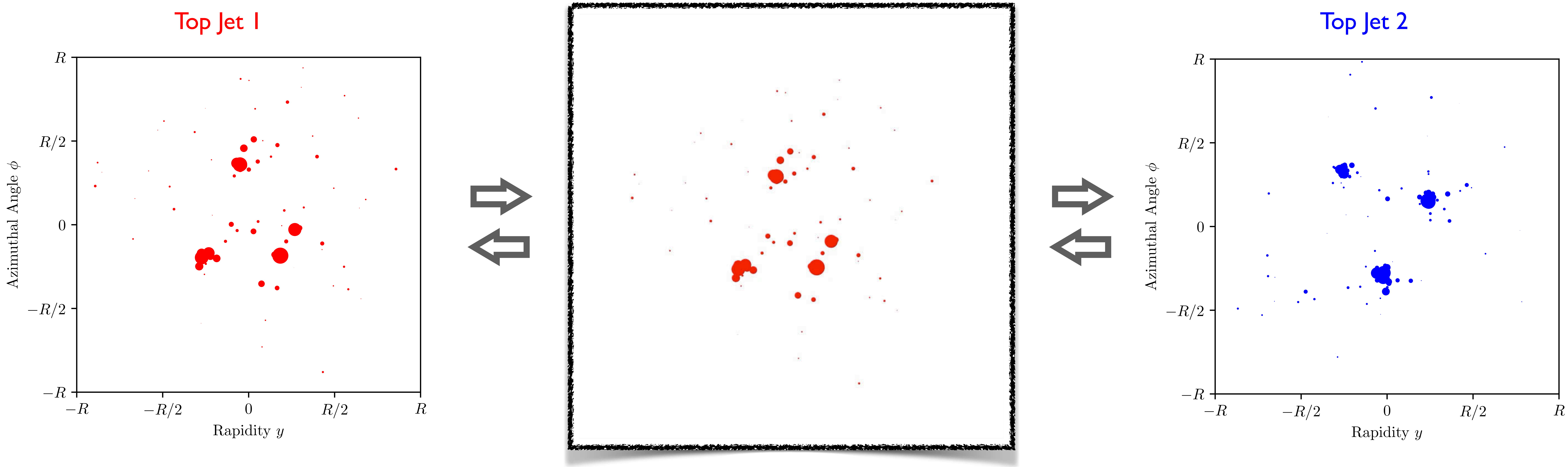
[Peleg, Werman, Rom, IEEE 1989; Rubner, Tomasi, Guibas, ICCV 1998, ICJV 2000; Pele, Werman, ECCV 2008; Pele, Taskar, GSI 2013]

When are Two Events similar?



[**PTK**, Metodiev, Thaler, PRL 2019]

Optimal transport minimizes the "work" (stuff x distance) required to transport supply to demand



$$\mathcal{E}(\hat{n}) = \sum_{i=1}^{M} E_i\, \delta(\hat{n} - \hat{n}_i)$$

Provides a metric on normalized distributions in a space with a ground distance measure

↳ symmetric, non-negative, triangle inequality, zero iff identical

$$\theta_{ij} = \sqrt{\Delta y_{ij}^2 + \Delta\phi_{ij}^2}$$

[Peleg, Werman, Rom, IEEE 1989; Rubner, Tomasi, Guibas, ICCV 1998, ICJV 2000; Pele, Werman, ECCV 2008; Pele, Taskar, GSI 2013]

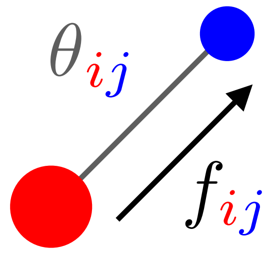
The Energy Mover's Distance (EMD)

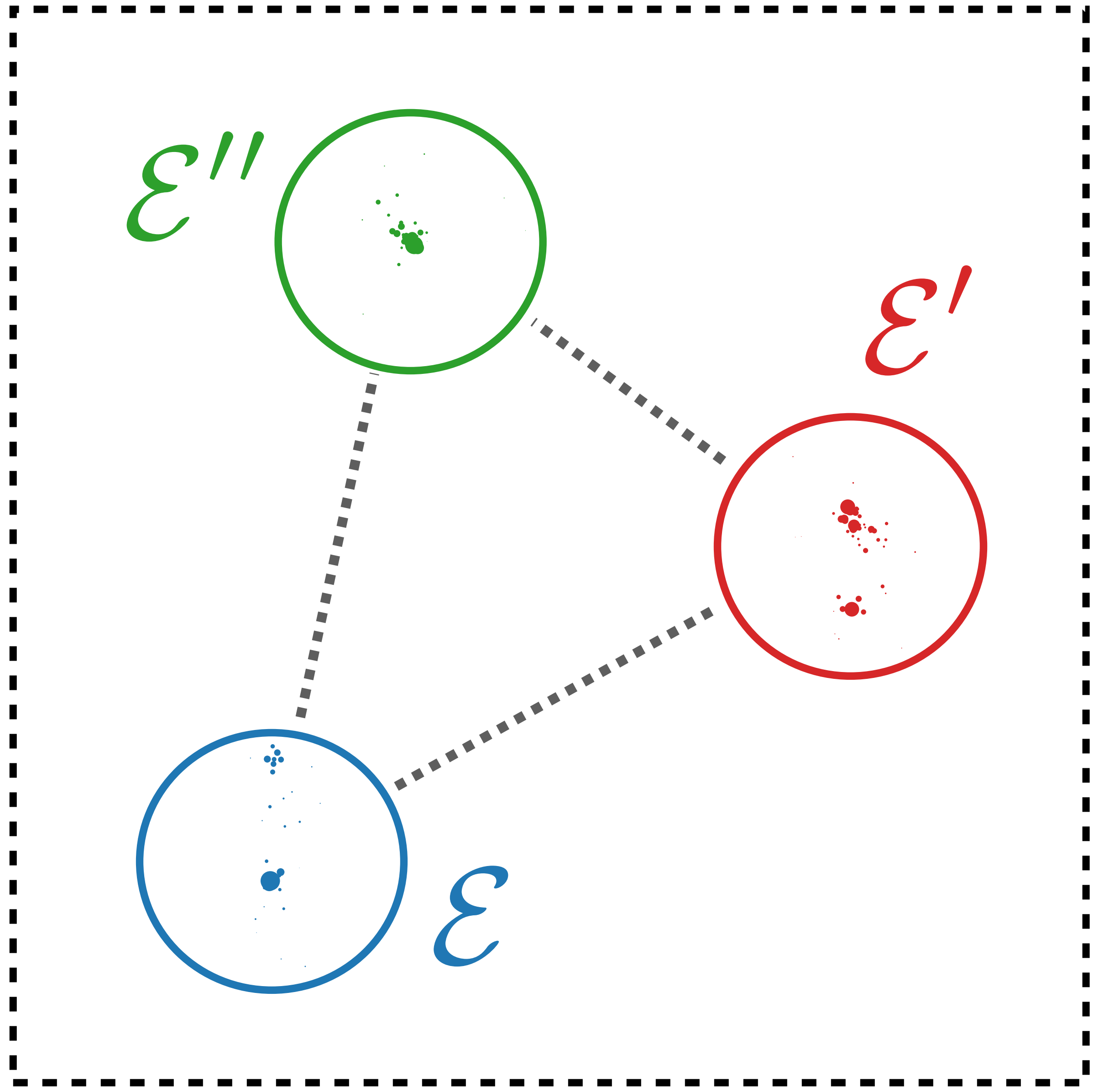
[**PTK**, Metodiev, Thaler, PRL 2019]

EMD between energy flows defines a metric on the space of events

$$\mathrm{EMD}_{\beta,R}(\mathcal{E},\mathcal{E}') = \underbrace{\min_{\{f_{ij}\geq 0\}} \sum_i \sum_j f_{ij} \left(\frac{\theta_{ij}}{R}\right)^{\beta}}_{\text{Cost of optimal transport}} + \underbrace{\left| \sum_i E_i - \sum_j E'_j \right|}_{\text{Cost of energy creation}}$$

$$\underbrace{\sum_j f_{ij} \leq E_i, \quad \sum_i f_{ij} \leq E'_j, \quad \sum_{ij} f_{ij} = \min\left(\sum_i E_i, \sum_j E'_j\right)}_{\text{Capacity constraints to ensure proper transport}}$$





R: controls cost of transporting energy vs. destroying/creating it

β: angular weighting exponent

Triangle inequality satisfied for $R \geq d_{\max}/2$

$$0 \leq \mathrm{EMD}(\mathcal{E},\mathcal{E}') \leq \mathrm{EMD}(\mathcal{E},\mathcal{E}'') + \mathrm{EMD}(\mathcal{E}'',\mathcal{E}')$$

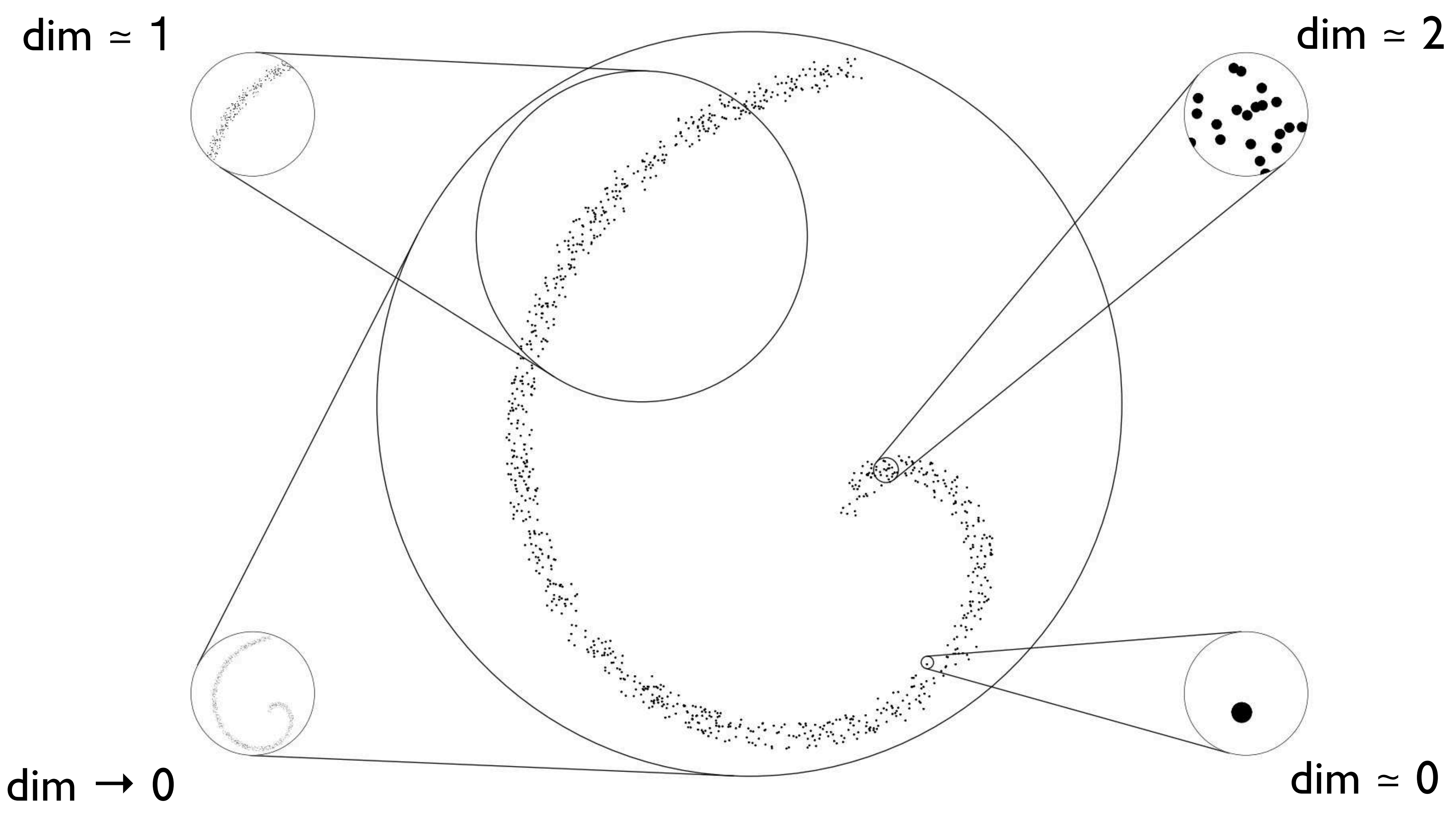
i.e. $R \geq$ jet radius for conical jets

Quantifying Event-Space Manifolds

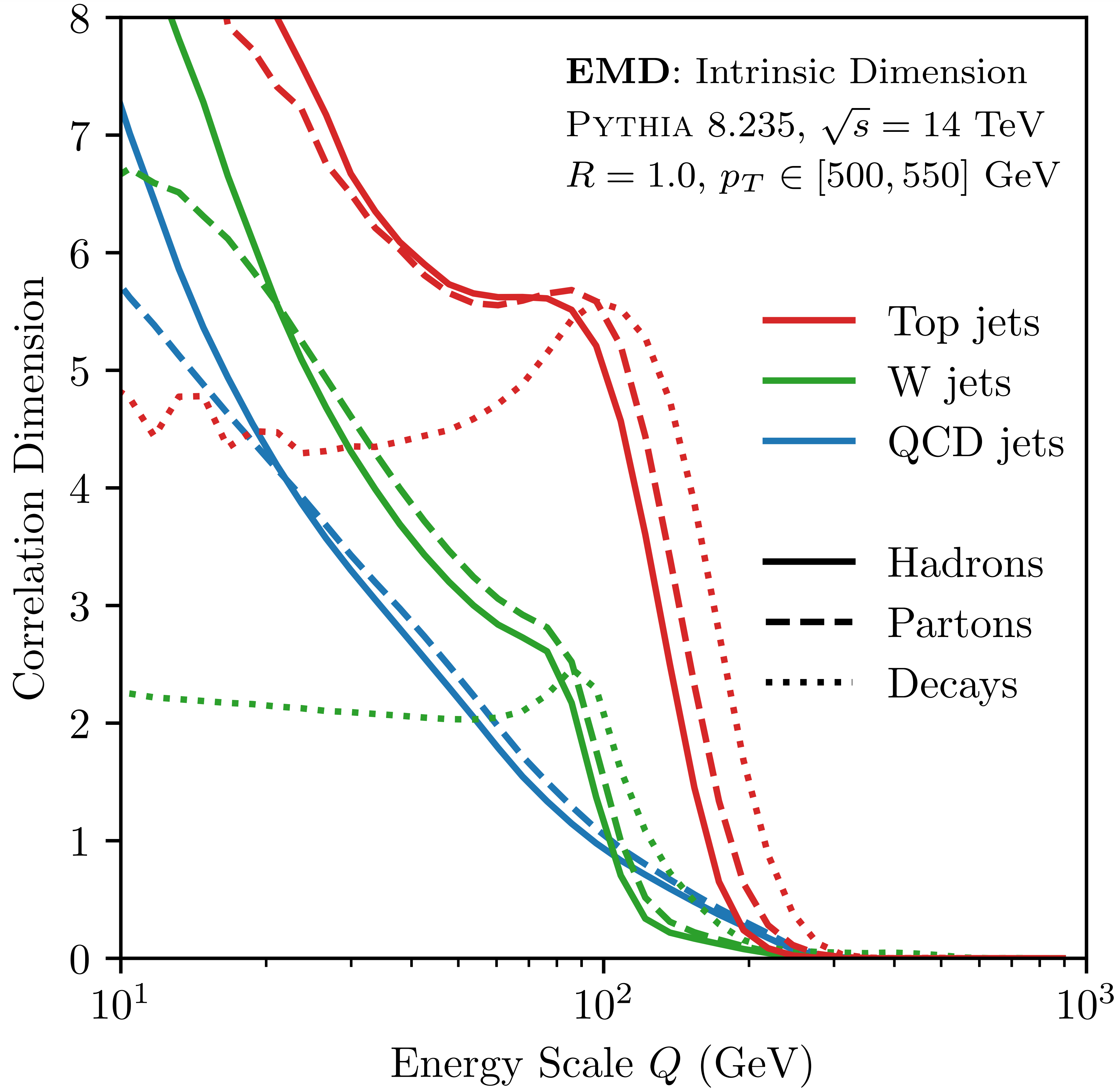
[PTK, Metodiev, Thaler, PRL 2019]

$$\dim(Q) = Q\frac{\partial}{\partial Q}\ln\sum_i\sum_j\Theta(\text{EMD}(\mathcal{E}_i,\mathcal{E}'_j) < Q)$$

Correlation dimension: *how does the # of elements within a ball of size Q change?*



$$N_{\text{neigh.}}(Q) \propto Q^{\dim} \implies \dim(Q) = Q\frac{d}{dQ}\ln N_{\text{neigh.}}(Q)$$

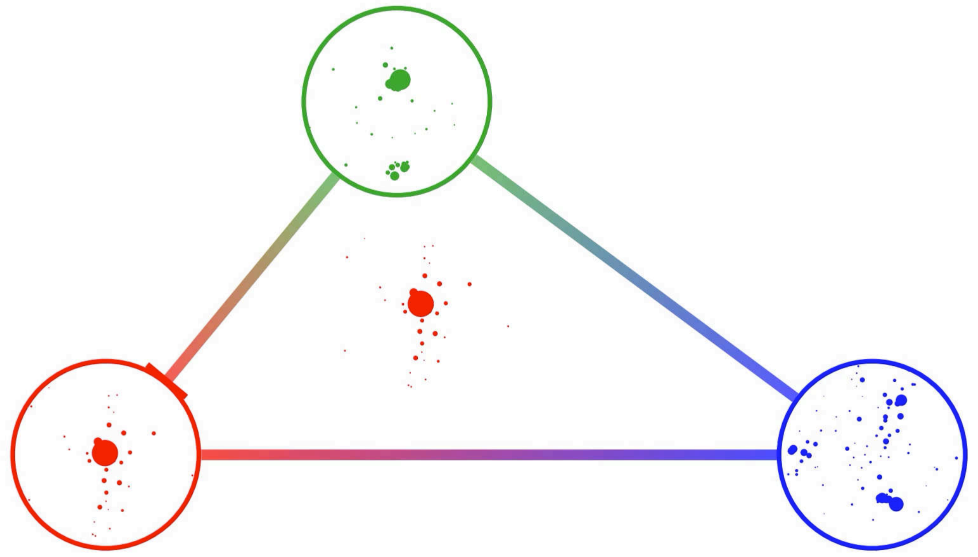


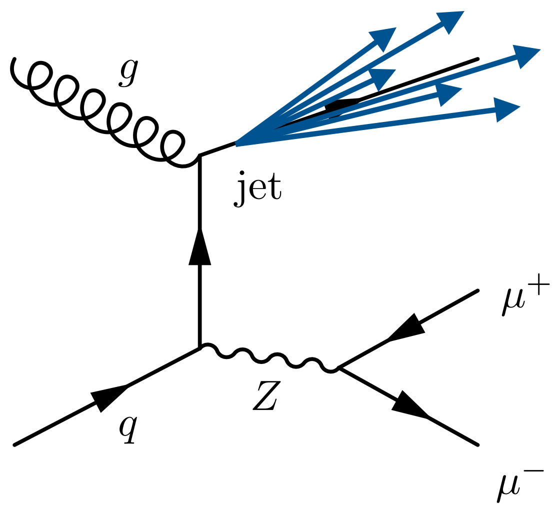
Correlation dimension lessons:
- Decays are "constant" dim. at low Q
- Complexity hierarchy: QCD < W < Top
- Fragmentation increases dim. at smaller scales
- Hadronization important around 20-30 GeV

[Grassberger, Procaccia, PRL 1983]

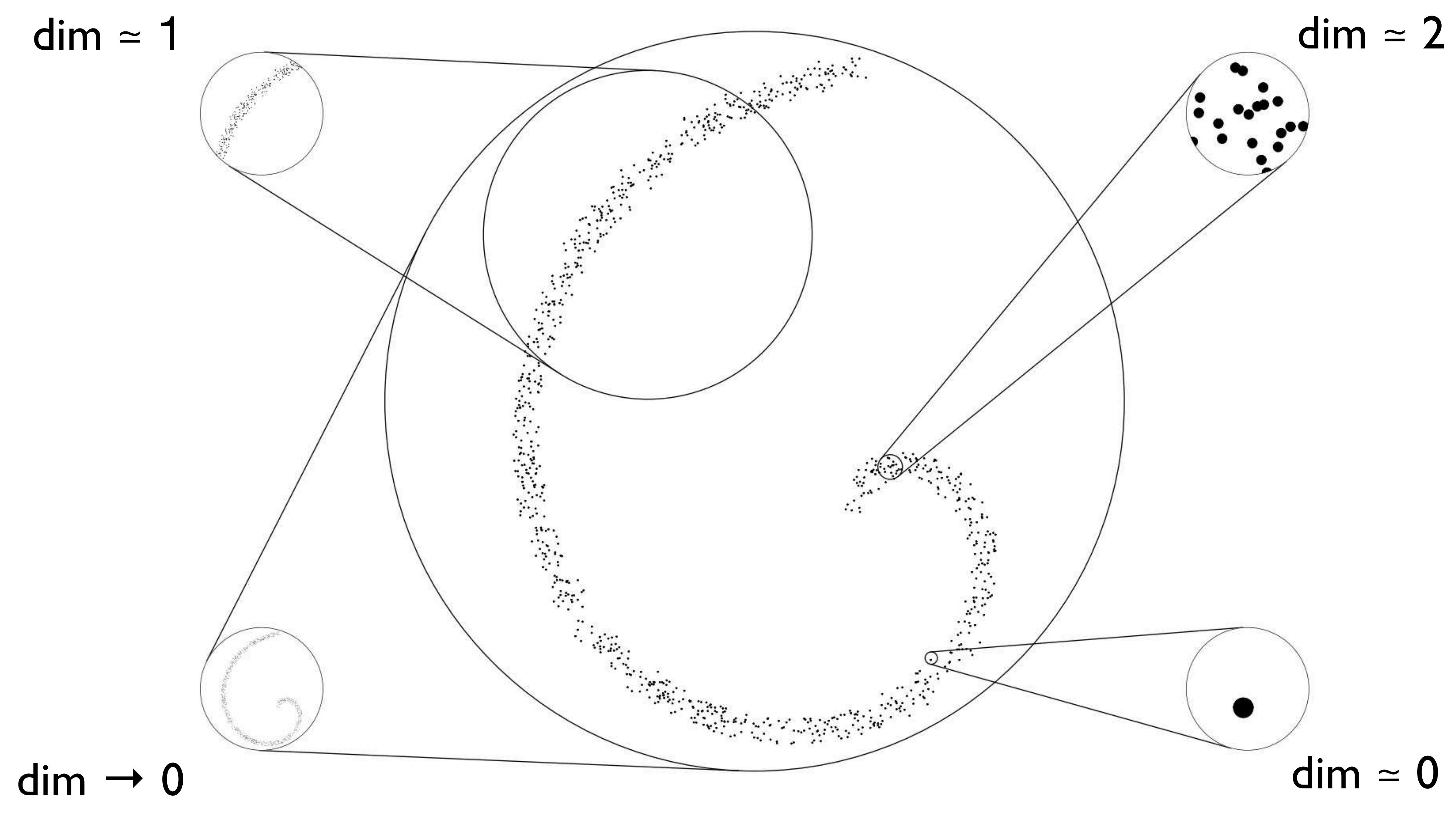
*Preliminary LL calculation for QG jets in backup

Unfolding Beyond Observables





Correlation dimension: *how does the # of elements within a ball of size Q change?*

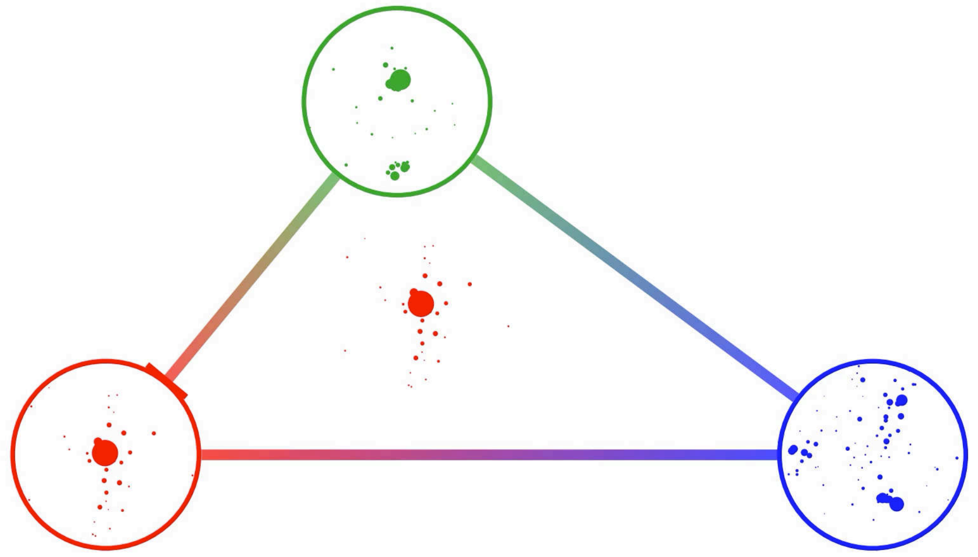


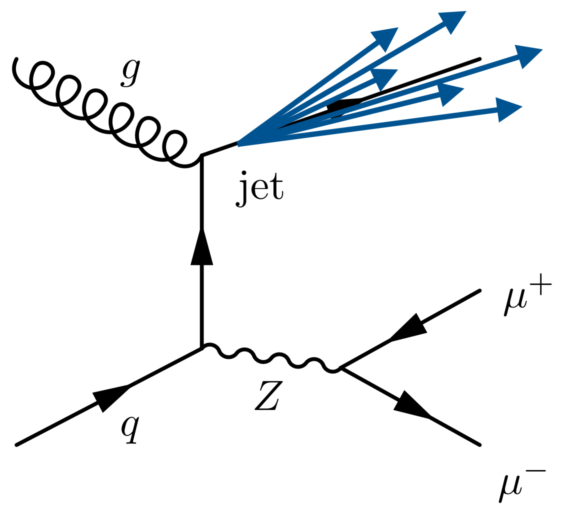
$$N_{\text{neigh.}}(Q) \propto Q^{\text{dim}} \implies \dim(Q) = Q\frac{d}{dQ}\ln N_{\text{neigh.}}(Q)$$

$$\dim(Q) = Q\frac{\partial}{\partial Q}\ln \sum_i \sum_j \boxed{w_i w'_j}\,\Theta(\text{EMD}(\mathcal{E}_i, \mathcal{E}'_j) < Q)$$

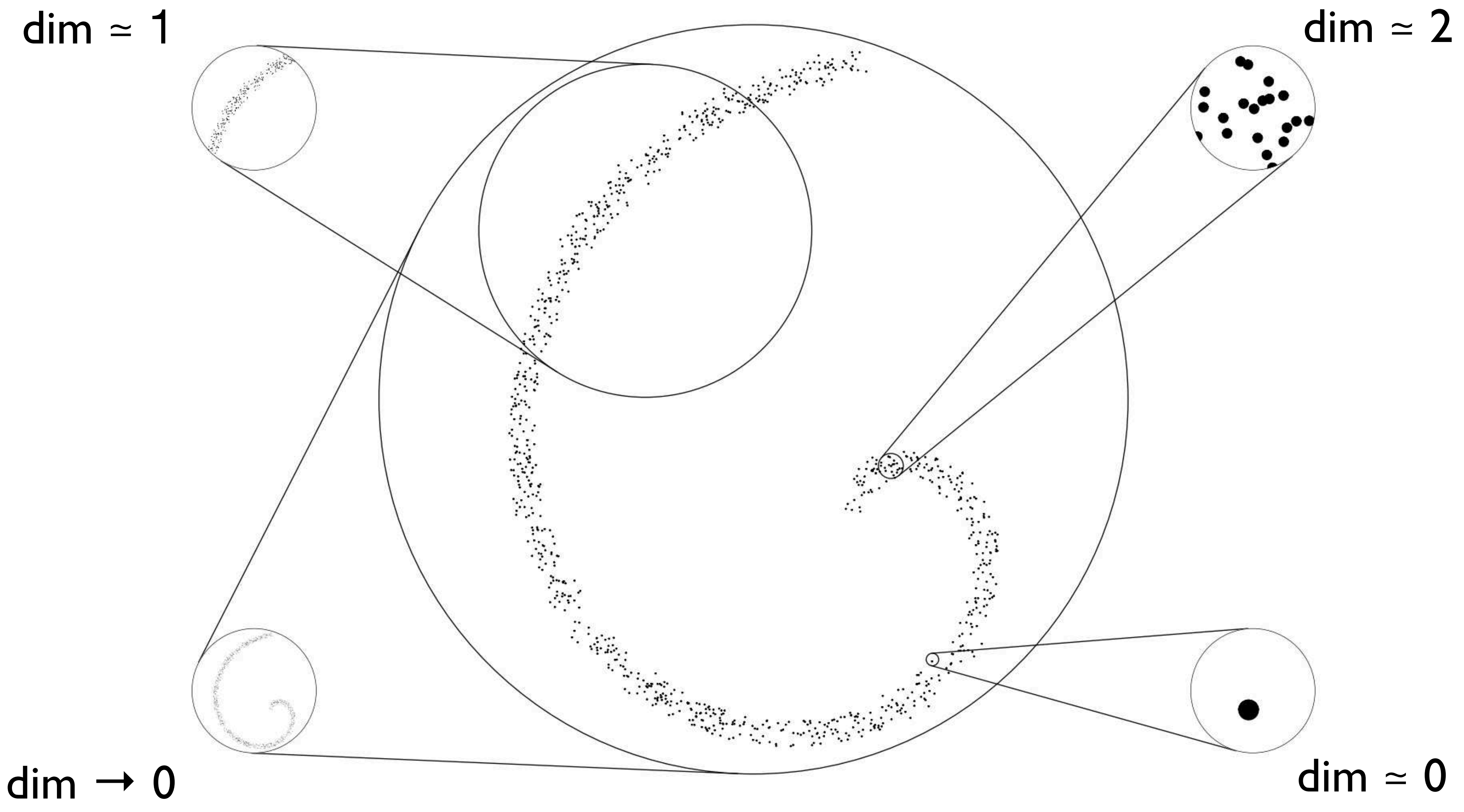
Weighted events naturally accommodated

Unfolding Beyond Observables





Correlation dimension: *how does the # of elements within a ball of size Q change?*



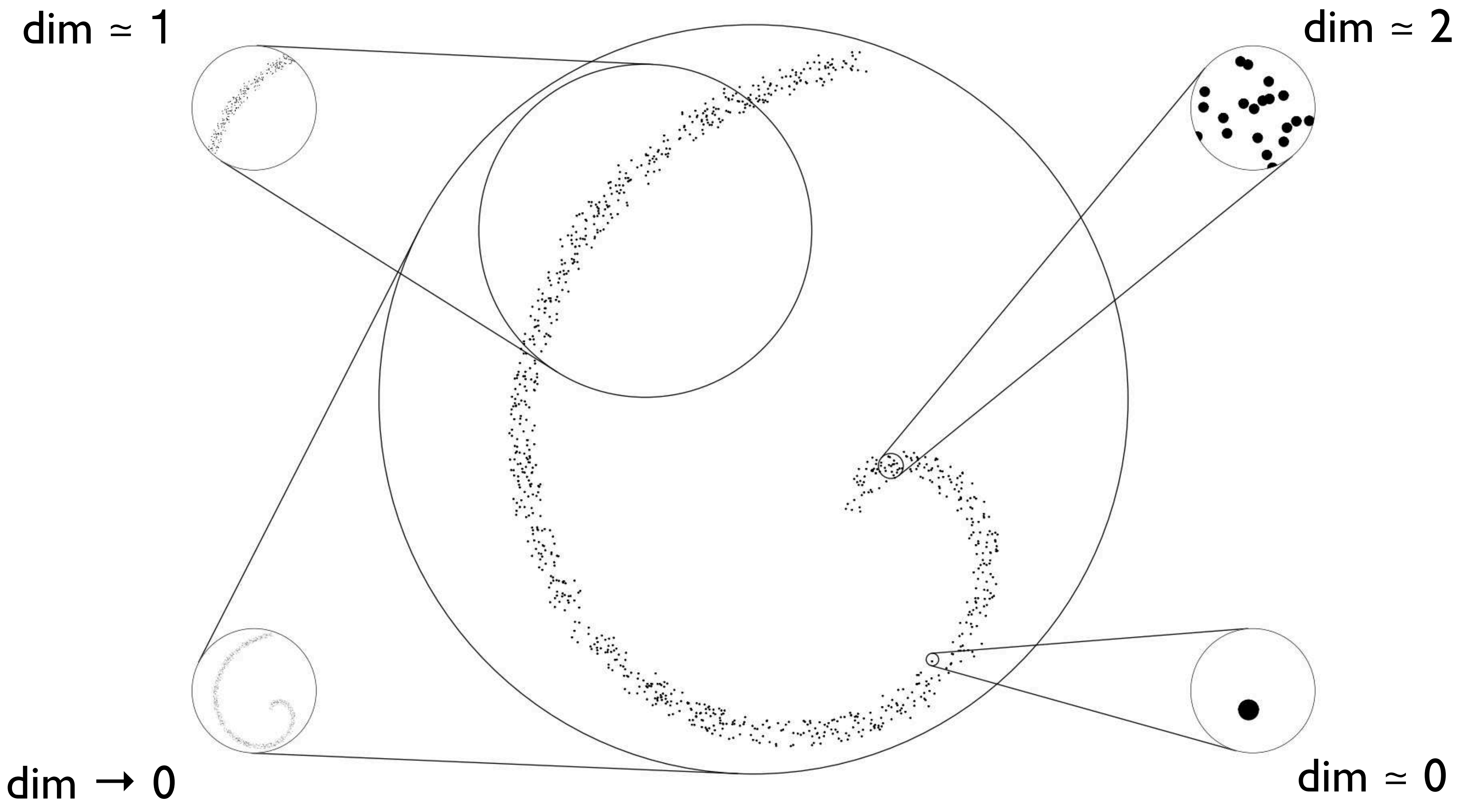
$$N_{\text{neigh.}}(Q) \propto Q^{\text{dim}} \implies \dim(Q) = Q \frac{d}{dQ} \ln N_{\text{neigh.}}(Q)$$

$$\dim(Q) = Q \frac{\partial}{\partial Q} \ln \sum_i \sum_j \boxed{w_i w'_j}\, \Theta(\text{EMD}(\mathcal{E}_i, \mathcal{E}'_j) < Q)$$

Weighted events naturally accommodated

Unfolding Beyond Observables

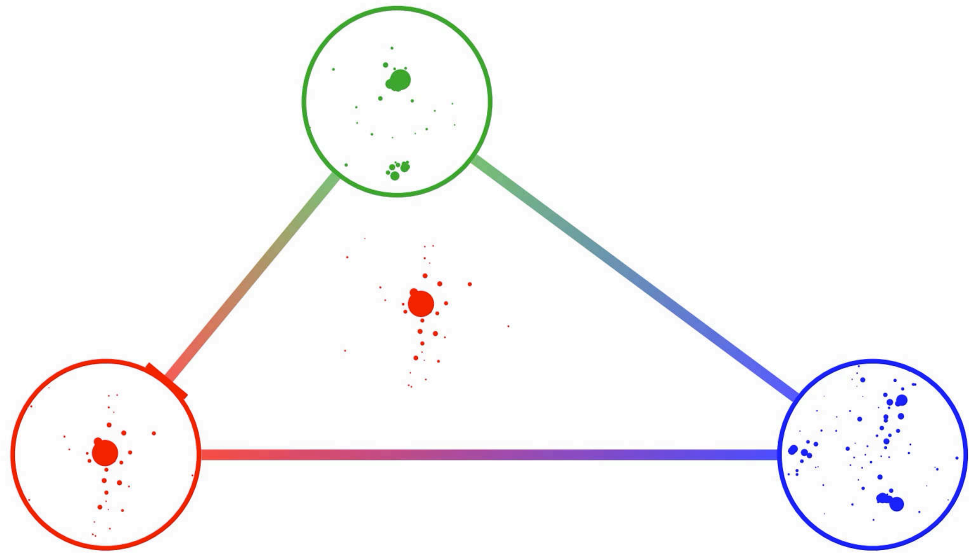
Correlation dimension: *how does the # of elements within a ball of size Q change?*

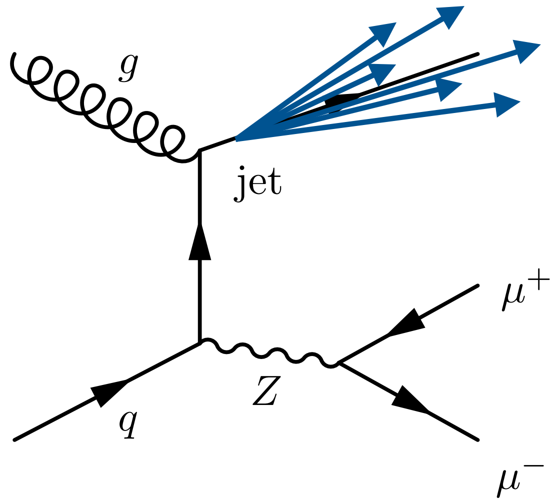


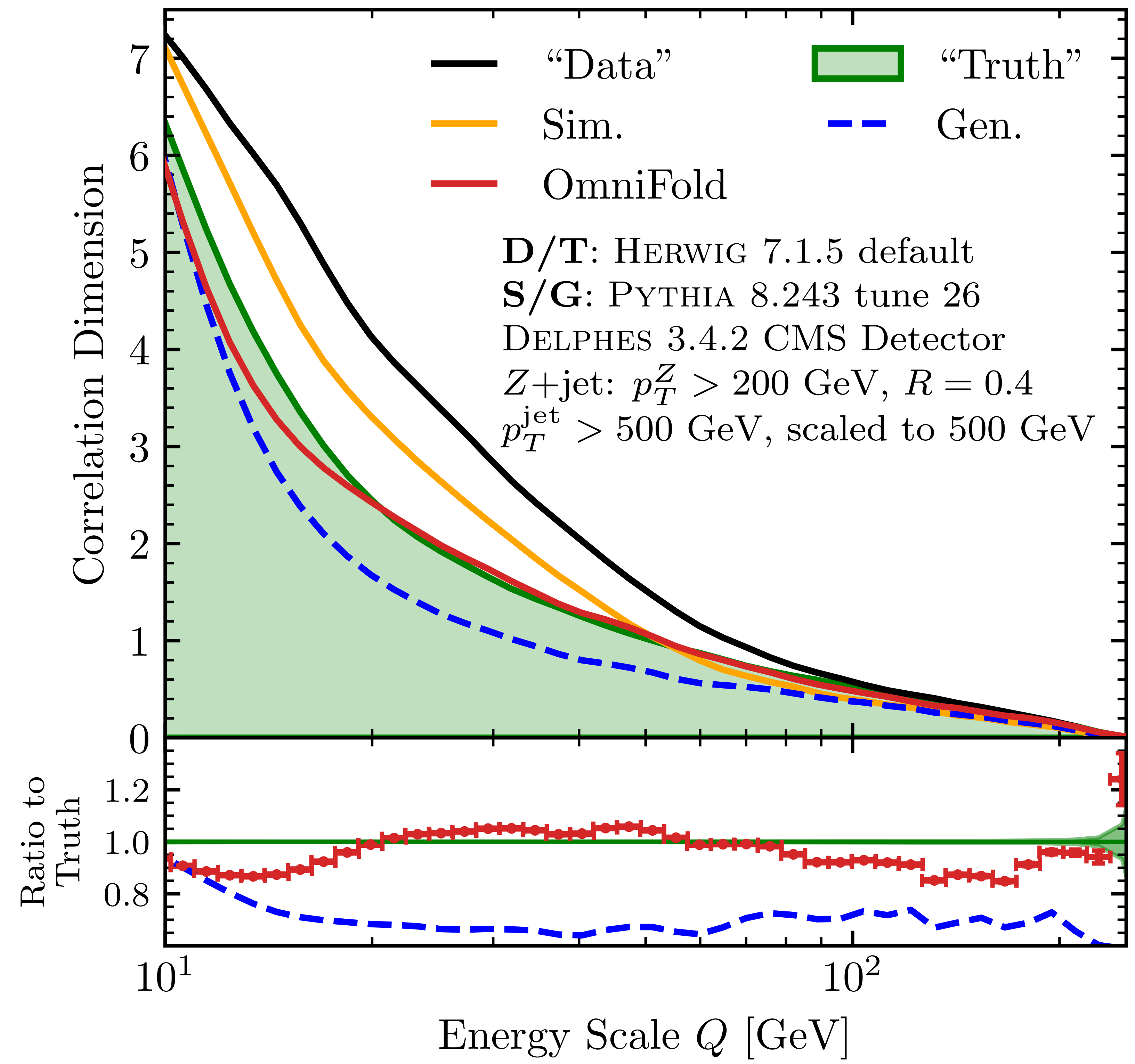
$$N_{\text{neigh.}}(Q) \propto Q^{\text{dim}} \implies \dim(Q) = Q\frac{d}{dQ}\ln N_{\text{neigh.}}(Q)$$

$$\dim(Q) = Q\frac{\partial}{\partial Q}\ln\sum_i\sum_j \boxed{w_i w'_j}\,\Theta(\mathrm{EMD}(\mathcal{E}_i,\mathcal{E}'_j) < Q)$$

Weighted events naturally accommodated







Same OmniFold training can unfold a complicated function of pairs of events!

Larger detector effects and loss of stats seen at low Q

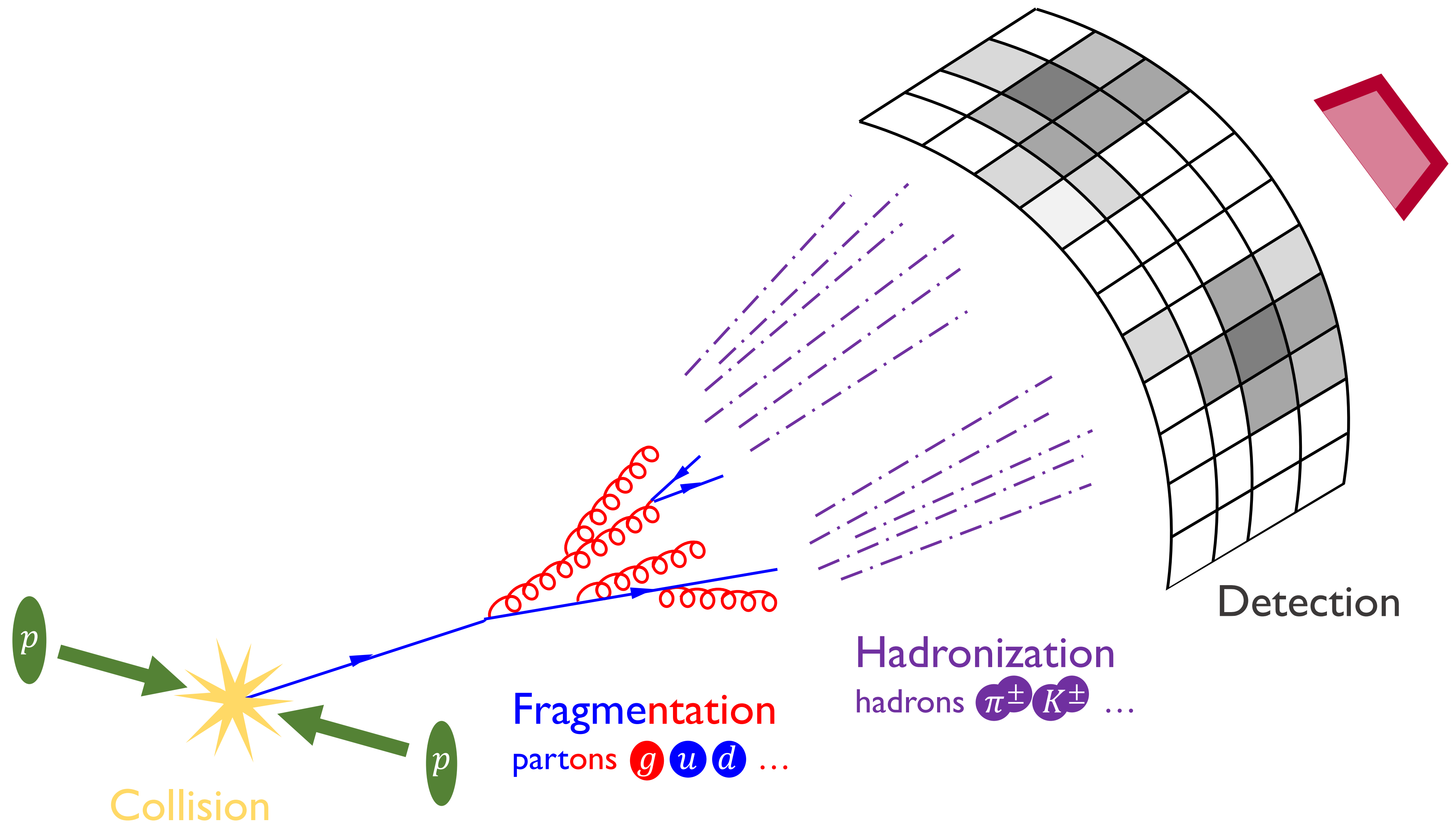
Beyond Observables via Weighted Cross Sections

Standard observables

Calculates a single number for each jet/event and study the distribution of values

Weighted cross section

Calculate a distributional quantity per event and study the mean distribution

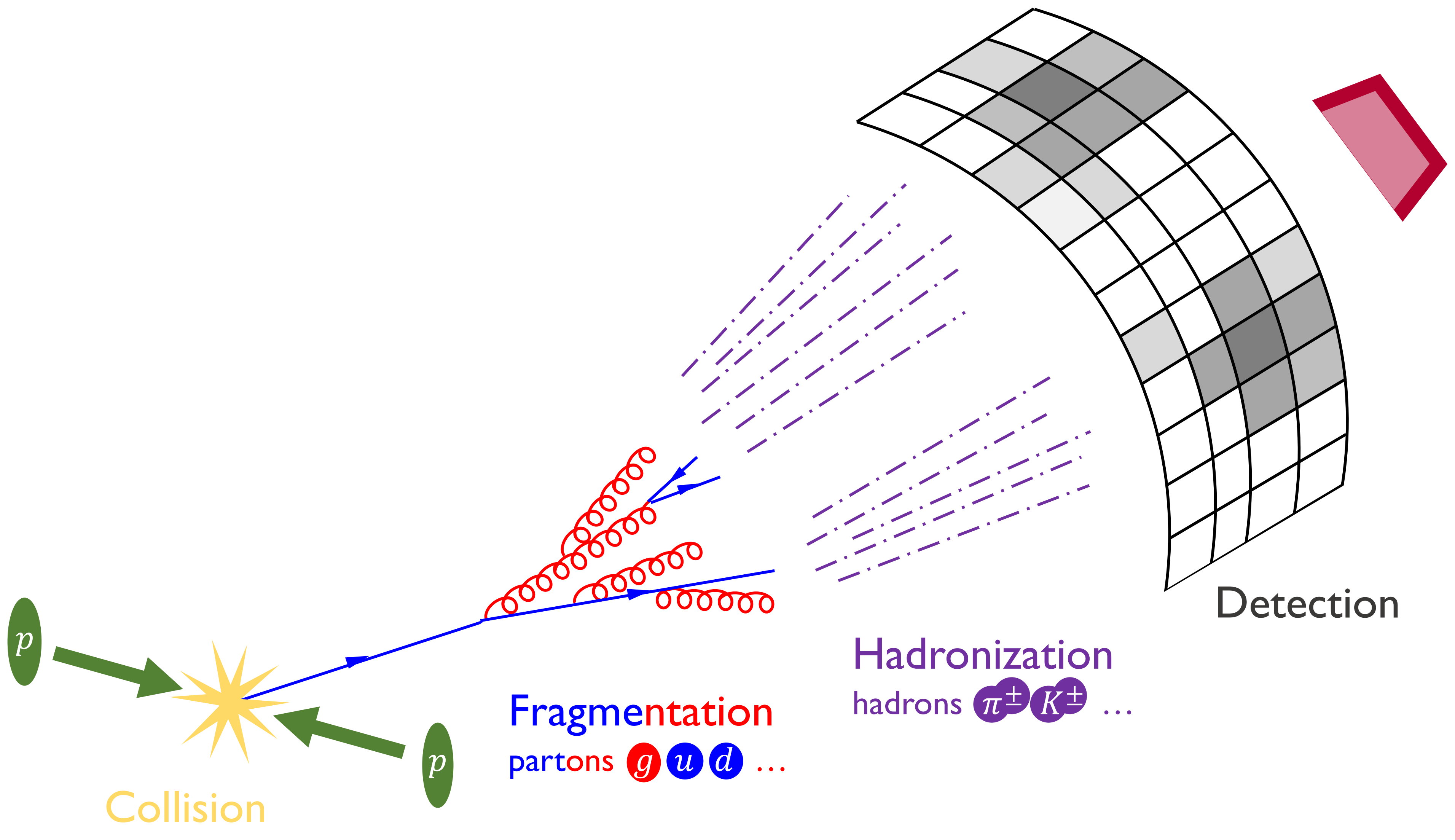


$$\mathcal{E}(\hat{n}) = \int_0^\infty dt \lim_{r\to\infty} r^2 n^i T_{0i}(t, r\hat{n})$$

Stress-energy flow

[PTK, Moult, Thaler, Zhu, *to appear soon*]

Beyond Observables via Weighted Cross Sections



$$\mathcal{E}(\hat{n}) = \int_0^\infty dt \lim_{r\to\infty} r^2 n^i T_{0i}(t, r\hat{n})$$

Stress-energy flow

Standard observables

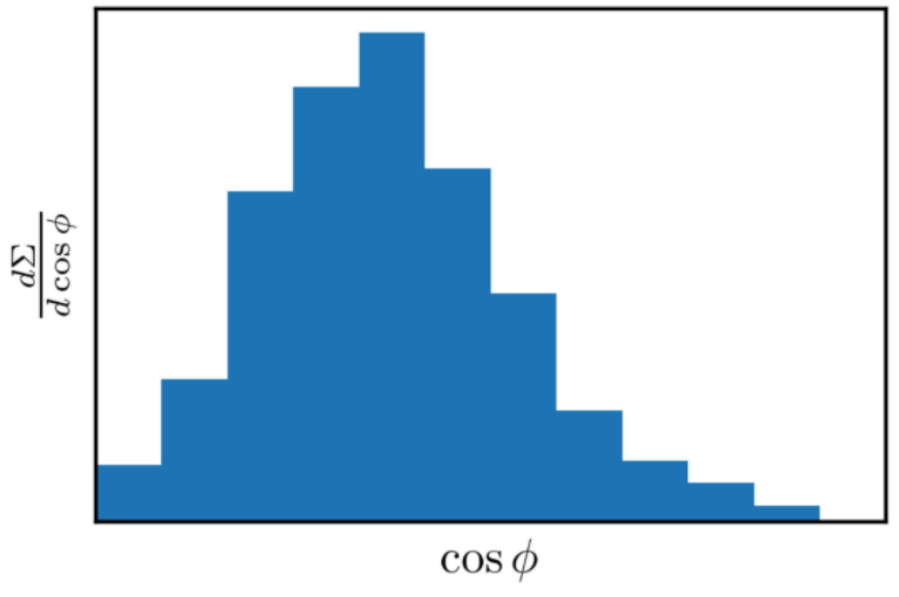
Calculates a single number for each jet/event and study the distribution of values

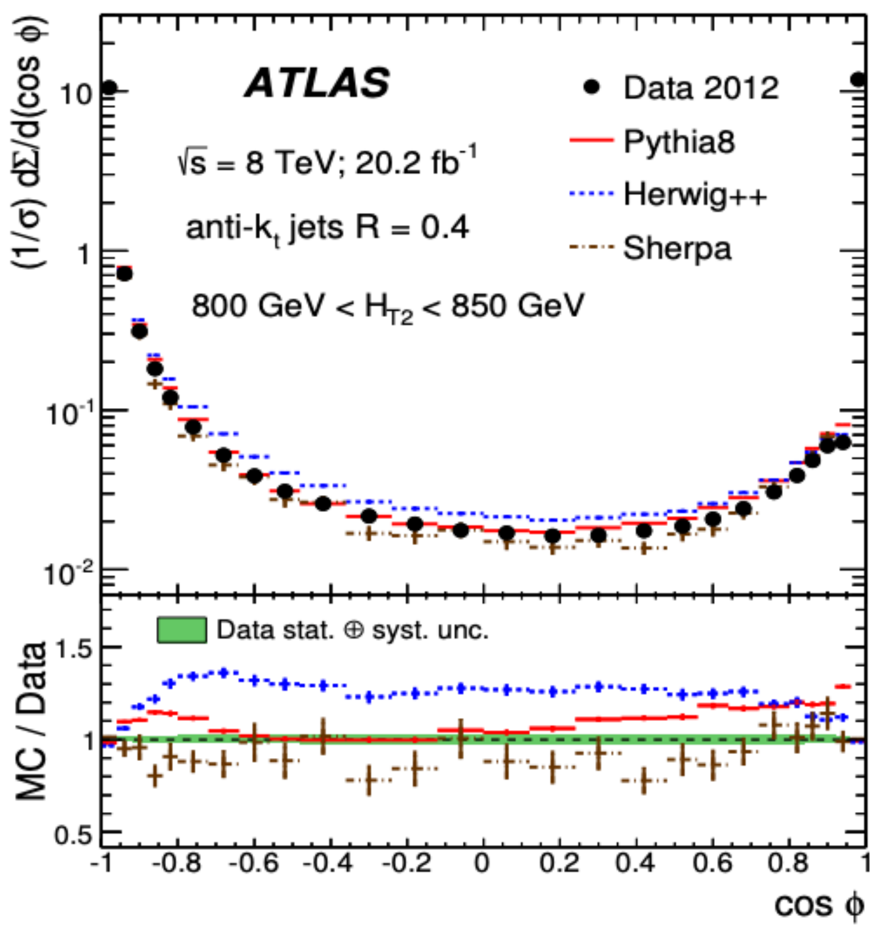
Weighted cross section

Calculate a distributional quantity per event and study the mean distribution

e.g. energy-energy correlator (EEC)

$$\frac{d\Sigma}{d\cos\phi} = \sum_{i,j} \int d\sigma \, \frac{E_i E_j}{Q^2} \delta(\cos\theta_{ij} - \cos\phi)$$



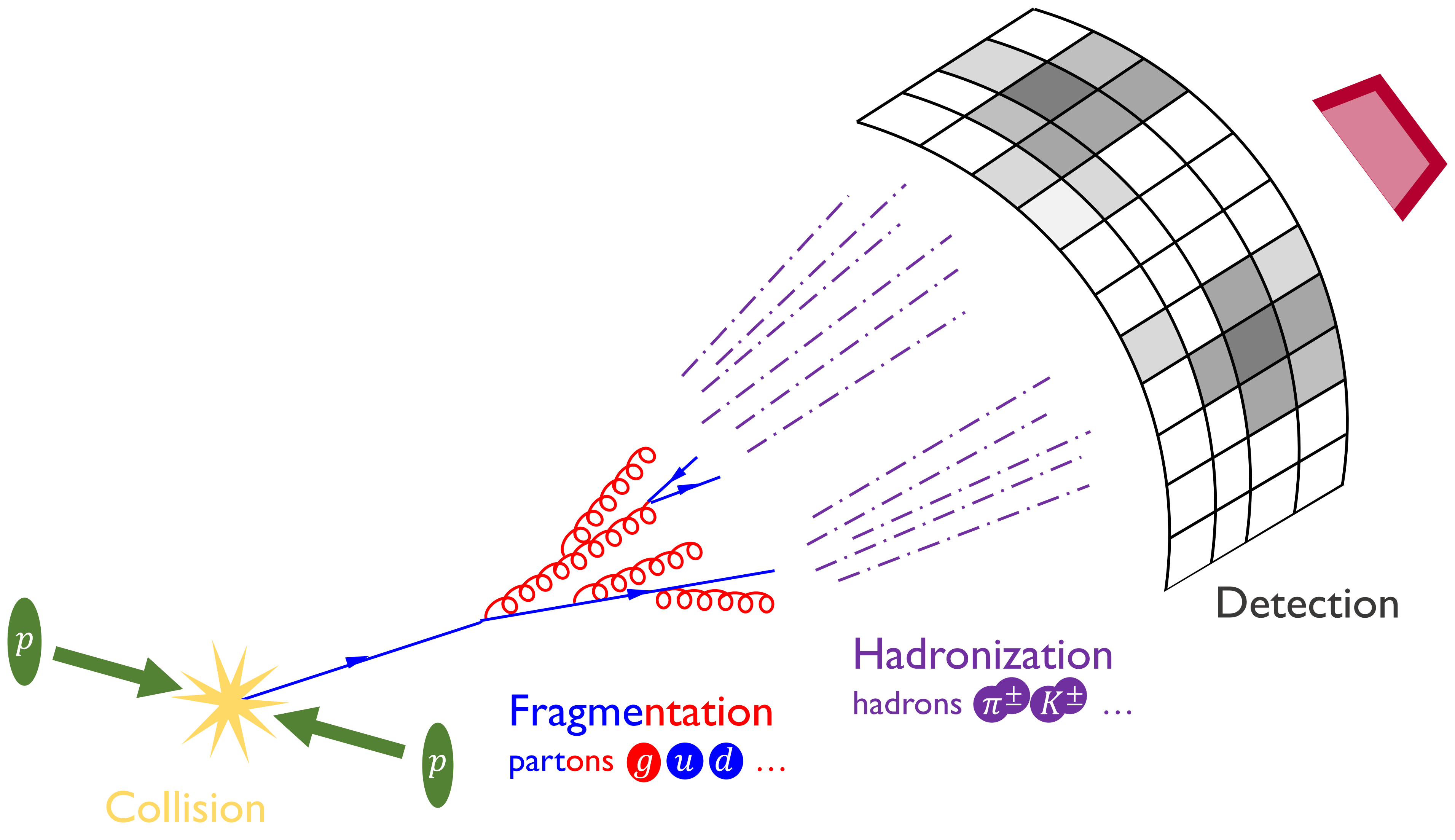


[PTK, Moult, Thaler, Zhu, *to appear soon*]

Beyond Observables via Weighted Cross Sections

Standard observables

Calculates a single number for each jet/event and study the distribution of values



$$\mathcal{E}(\hat{n}) = \int_0^\infty dt \lim_{r\to\infty} r^2 n^i T_{0i}(t, r\hat{n})$$

Stress-energy flow

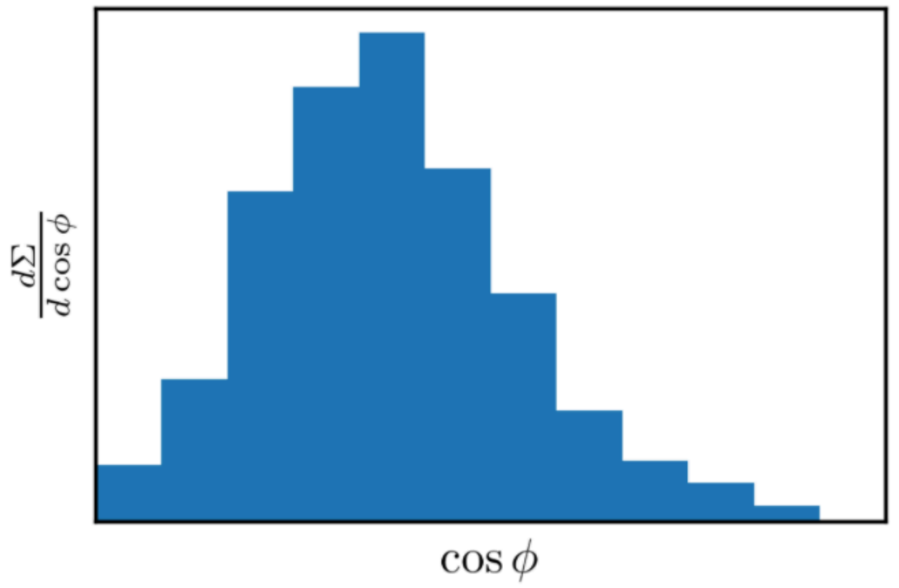
Weighted cross section

Calculate a distributional quantity per event and study the mean distribution

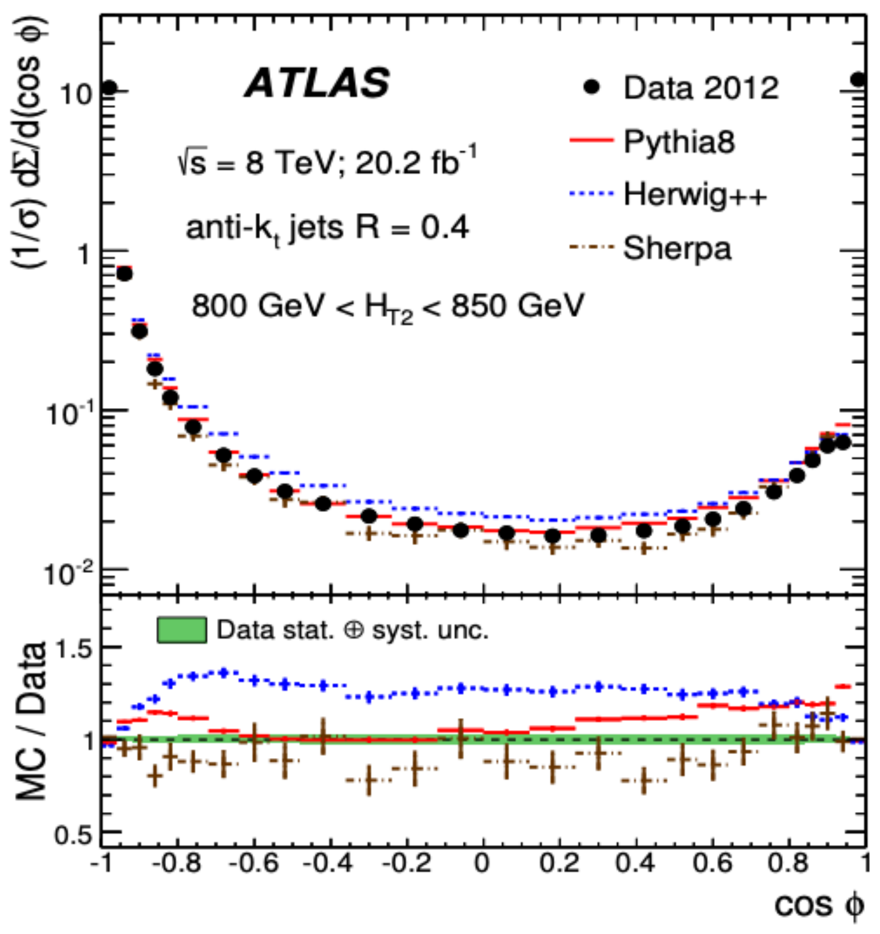
e.g. energy-energy correlator (EEC)

$$\frac{d\Sigma}{d\cos\phi} = \sum_{i,j} \int d\sigma \frac{E_i E_j}{Q^2} \delta(\cos\theta_{ij} - \cos\phi)$$

$$\Downarrow$$



$\Rightarrow$



$$\frac{1}{\sigma_{\text{tot}}} \frac{d\sigma}{d\hat{n}_1 \cdots d\hat{n}_N} = \frac{\langle \mathcal{O}\mathcal{E}(\hat{n}_1)\cdots\mathcal{E}(\hat{n}_N)\mathcal{O}^\dagger \rangle}{\langle \mathcal{O}\mathcal{O}^\dagger \rangle}$$

Correlations of energy flow operators can be directly studied!

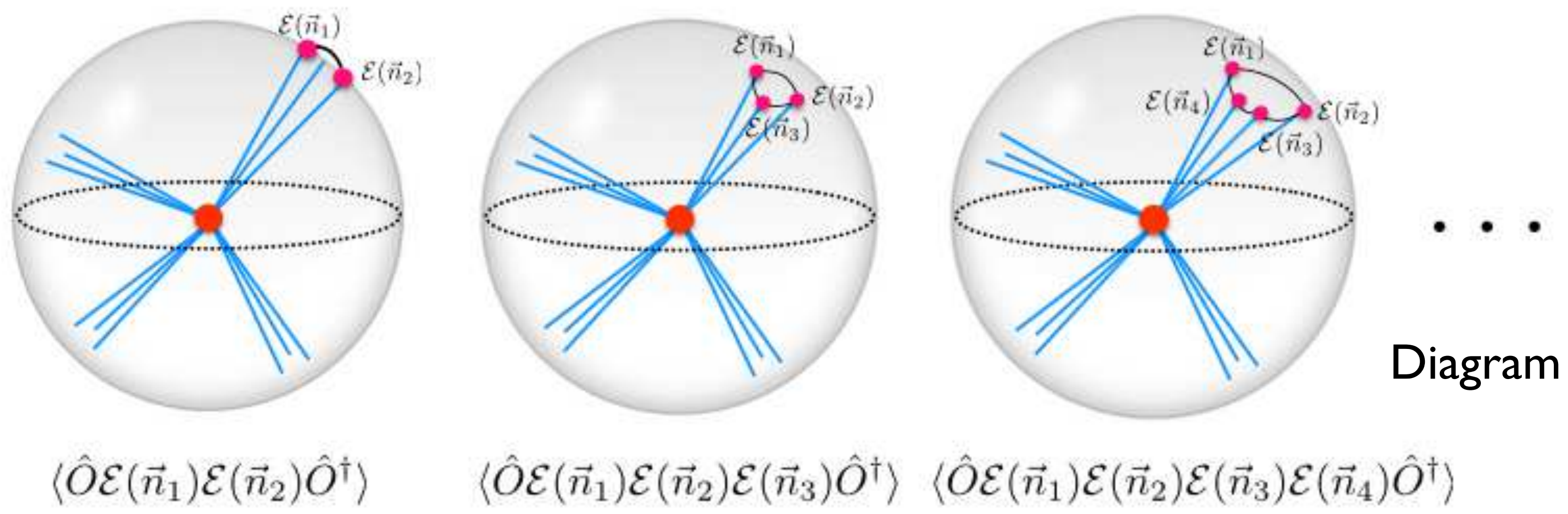
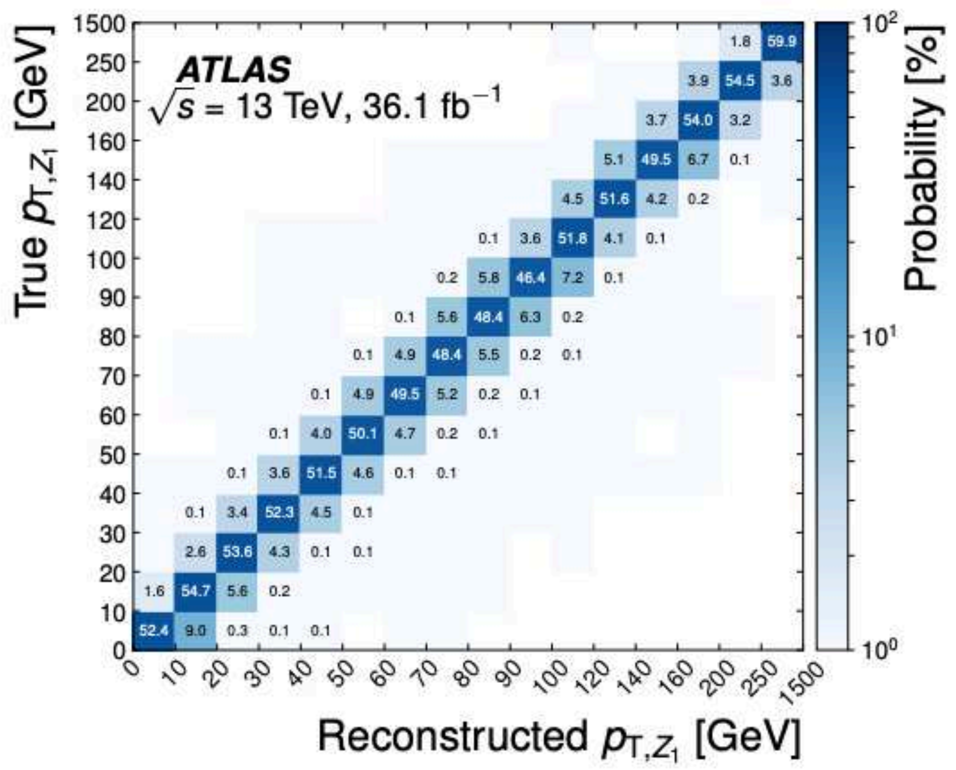
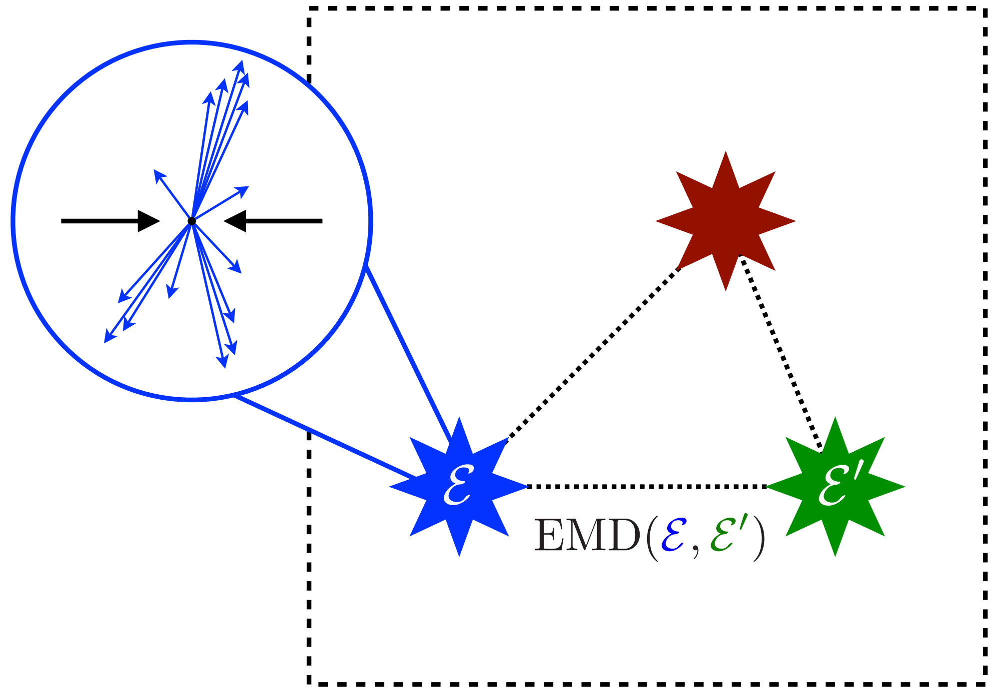


Diagram by Ian Moult

[PTK, Moult, Thaler, Zhu, *to appear soon*]





Unfolding Setup

- Unfolding corrects distributions for detector effects using detector simulation in a *prior independent* manner
- Traditional unfolding (e.g. IBU) is based on histograms and limited to a few observables due to the curse of dimensionality

OmniFold

- Maximum likelihood solution to the unfolding problem (like IBU)
- Likelihood-free inference uses high-dim. classifiers to capture the full phase-space
- Learns a single particle-level weighting function that unfolds *all* observables

Unfolding Beyond Observables

- Non-per-event quantities can be of broad interest in particle physics
- Traditional unfolding is challenged due to reliance on low-dimensional histograms
- Can be unfolded with OmniFold as easily and naturally as any other quantity

Additional Slides

OmniFold Etymology

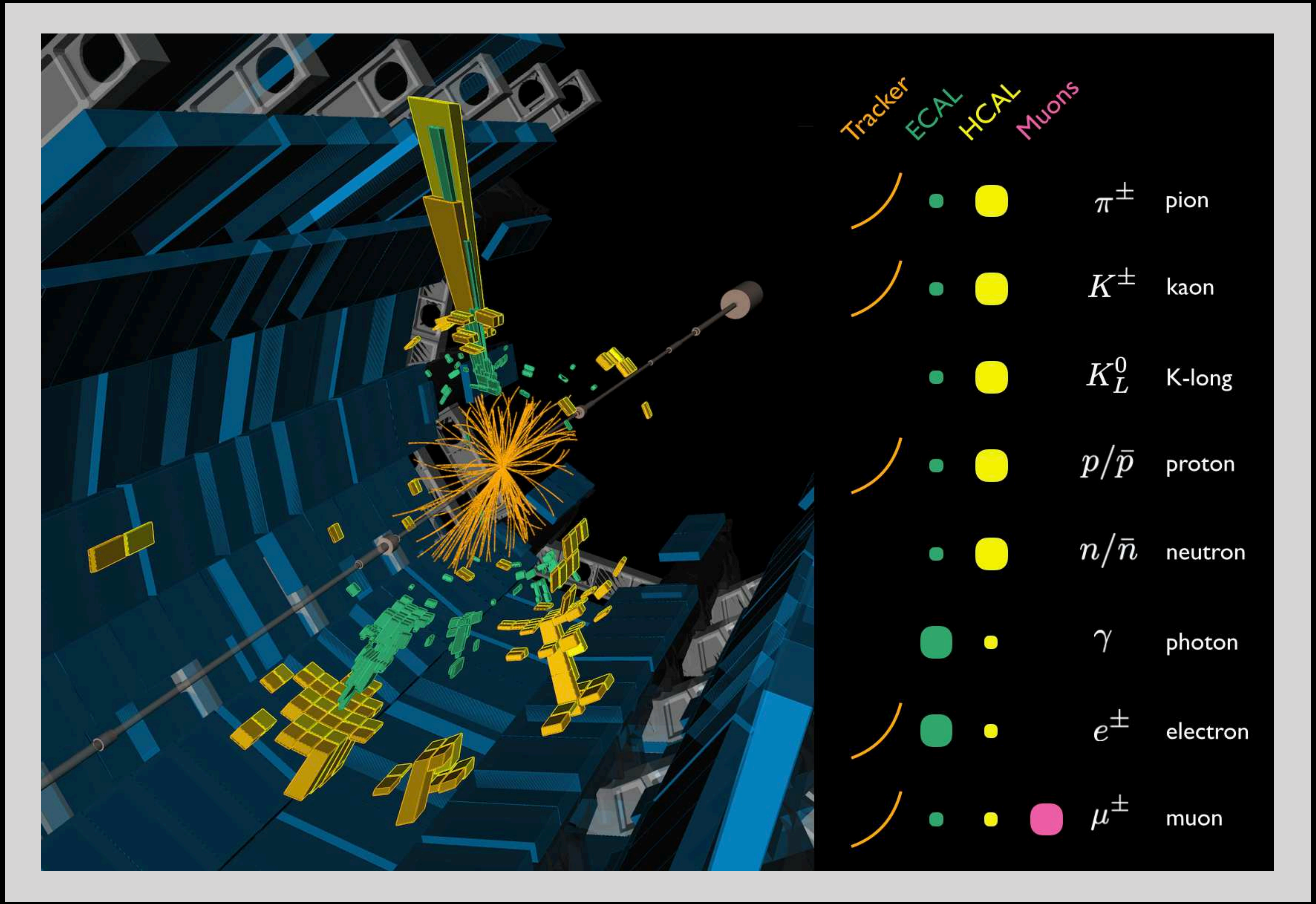
The Mountain sat upon the Plain
In his tremendous Chair –
His observation **omnifold**,
His inquest, everywhere –

The Seasons played around his knees
Like Children round a sire –
Grandfather of the Days is He
Of Dawn, the Ancestor –

Emily Dickinson, #975

Particle Collider Events

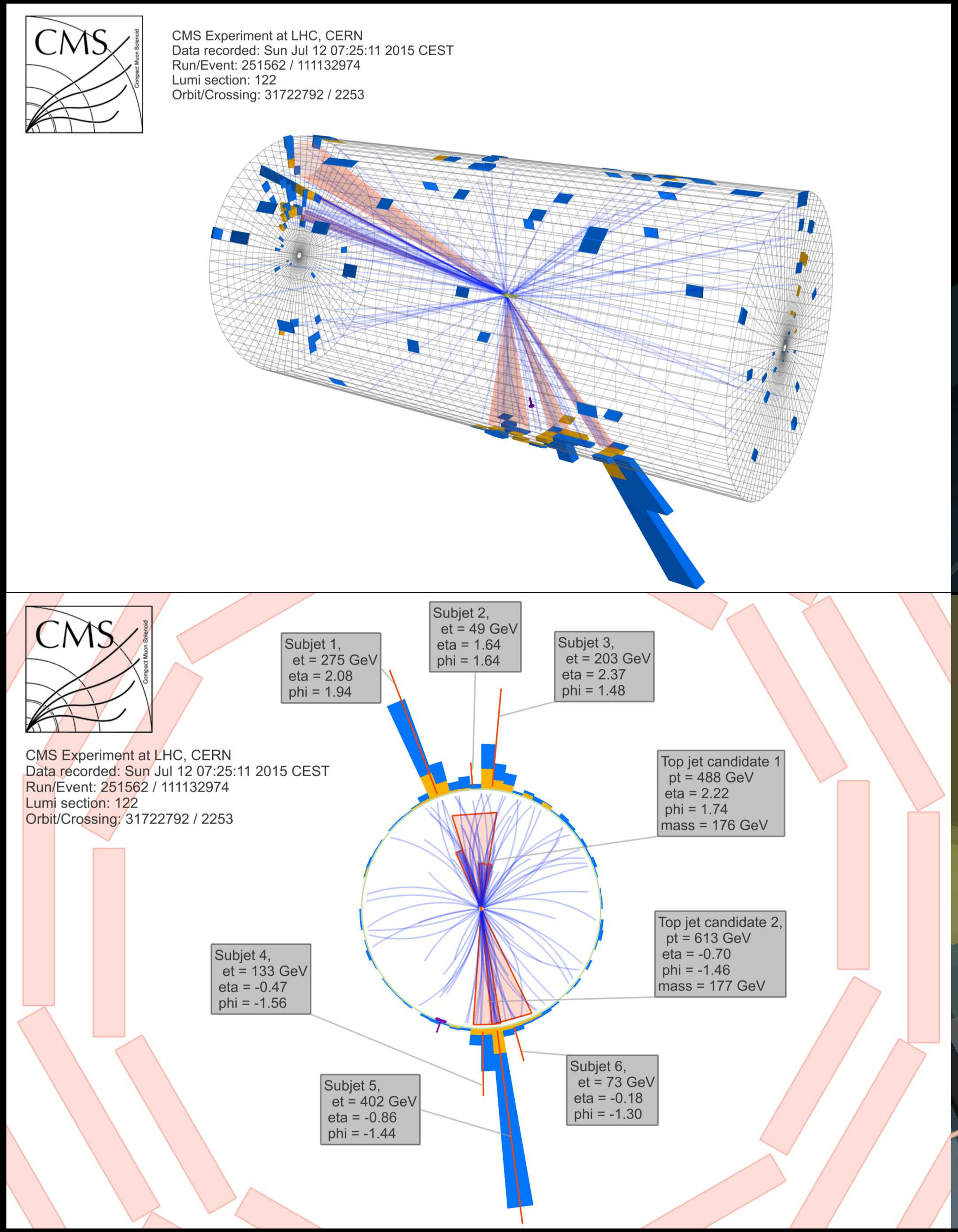
High-energy collisions produce final state particles with
energy, *direction*, *charge*, *flavor*, and *other quantum numbers*



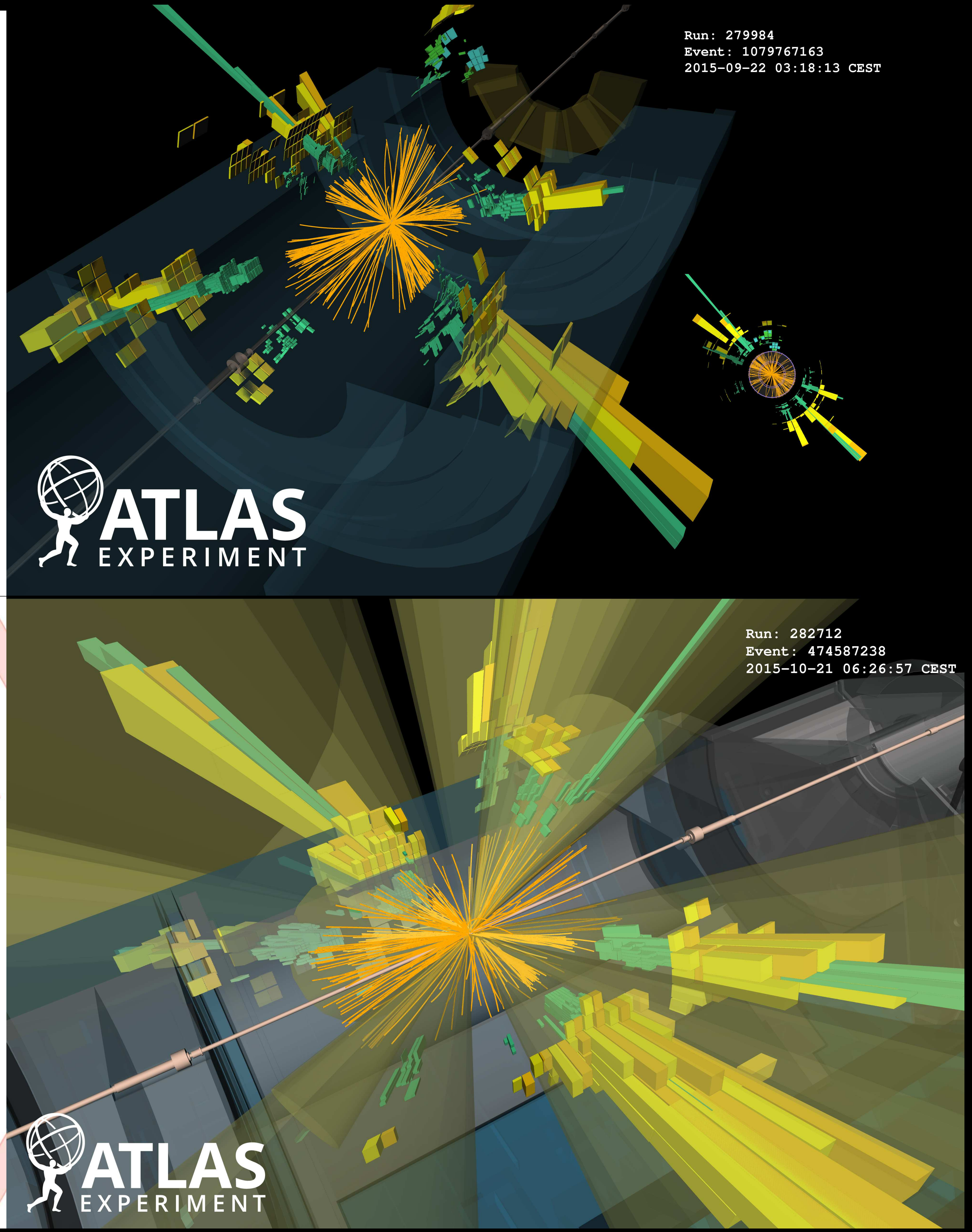
e.g. the ATLAS detector

Jets at the Large Hadron Collider

CMS hadronic $t\bar{t}$ event

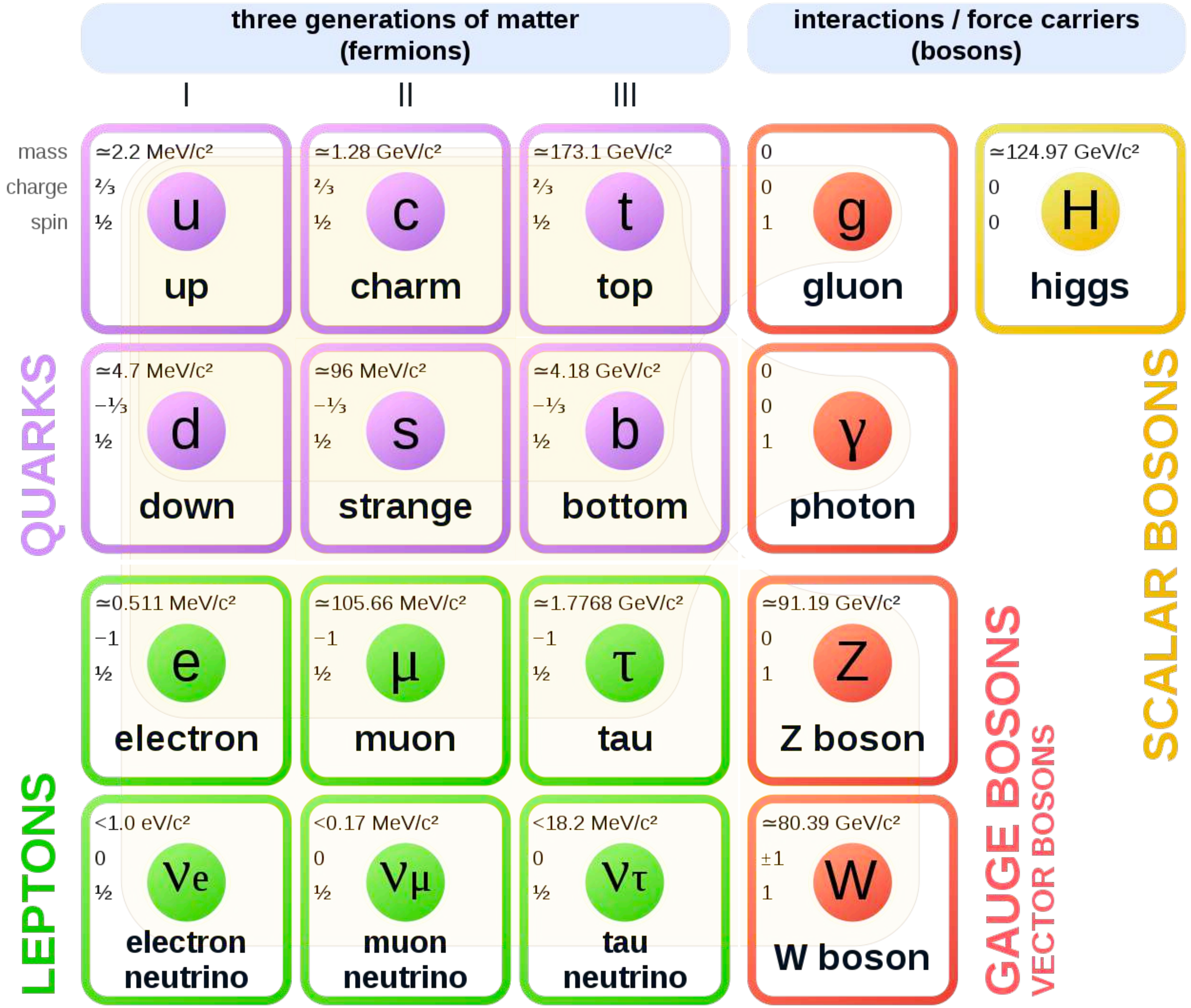


ATLAS high jet multiplicity events

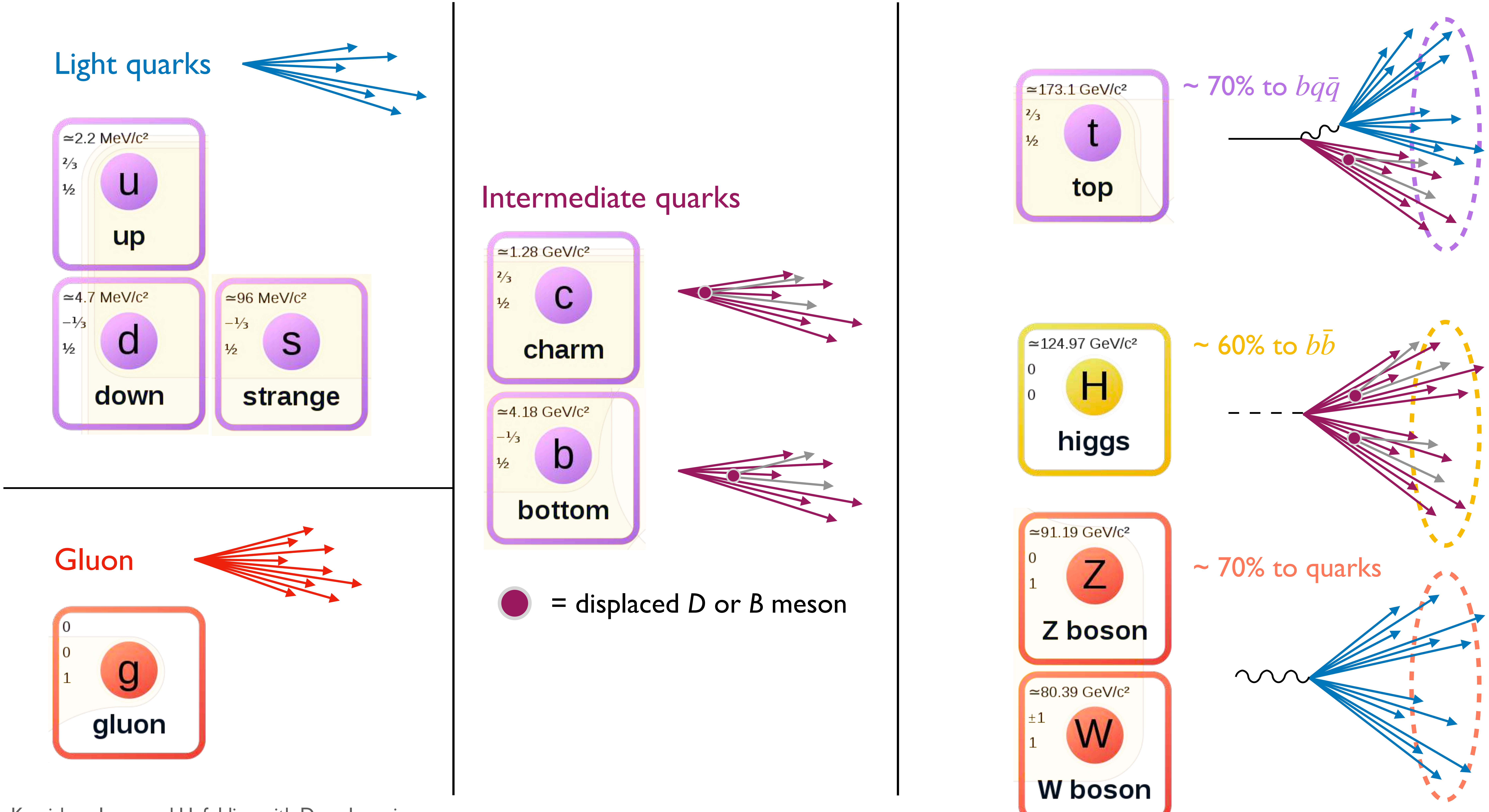


Jet substructure techniques enabled by fantastic detector resolution and reconstruction

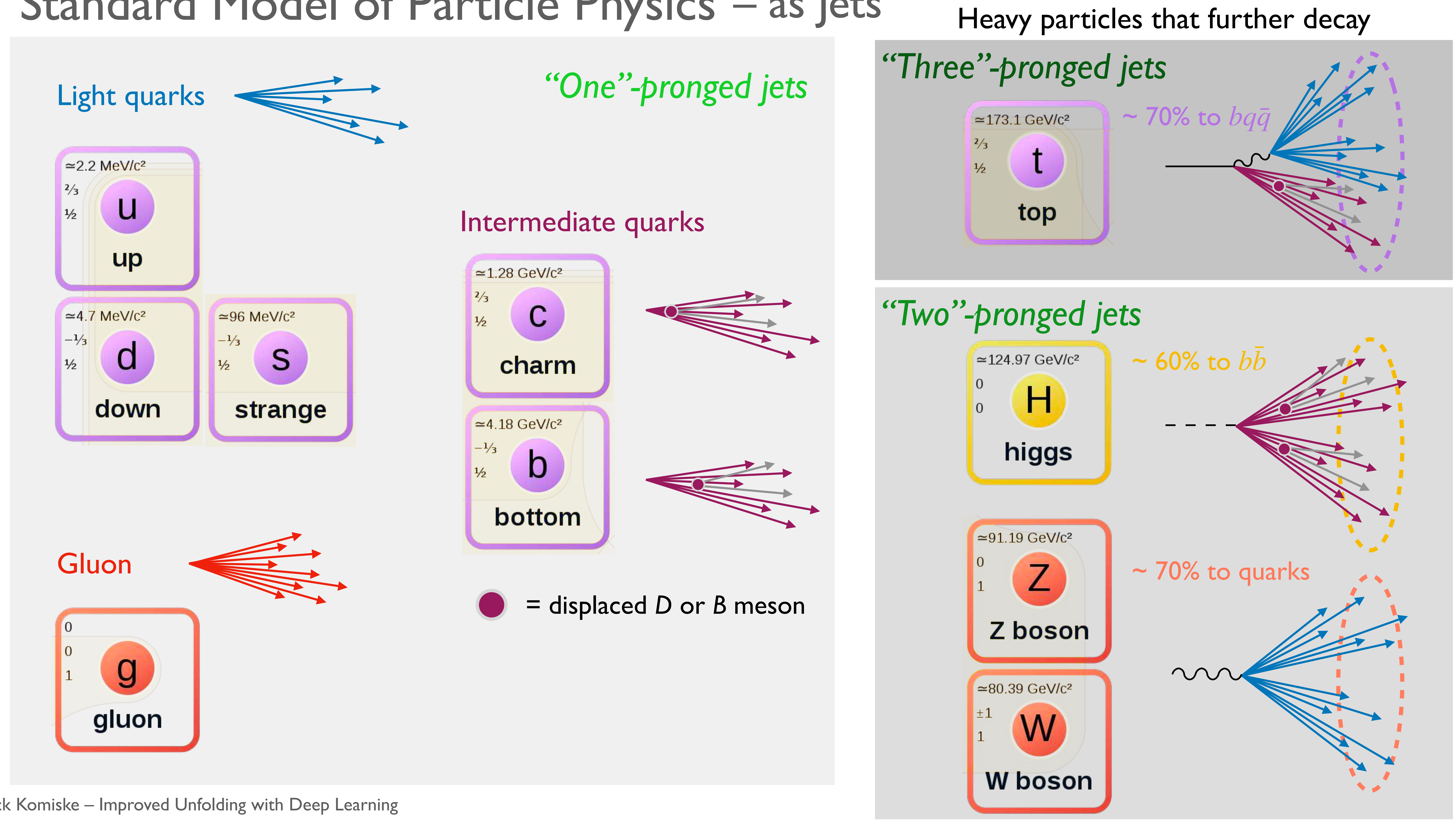
Standard Model of Particle Physics



Standard Model of Particle Physics – as Jets

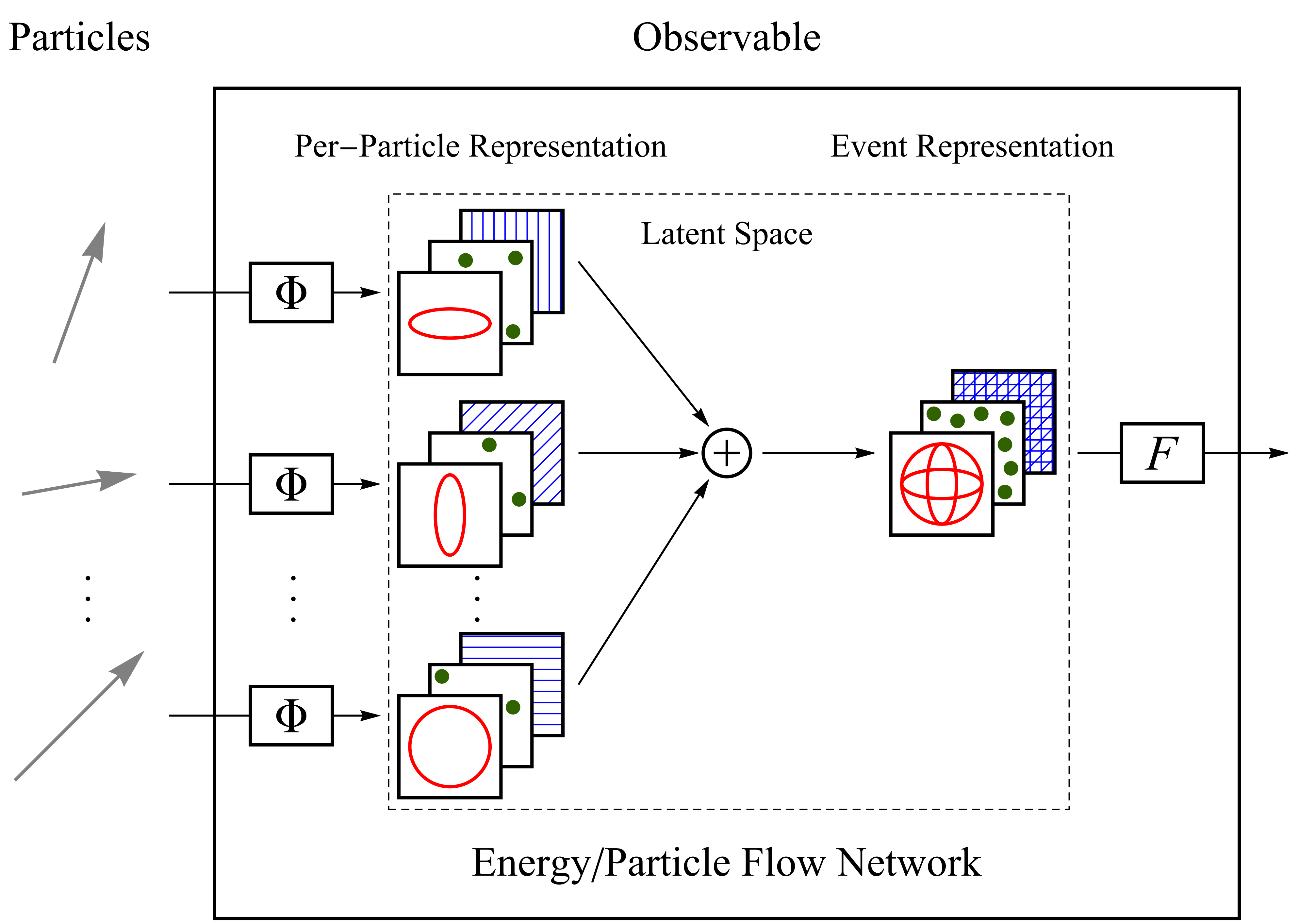


Standard Model of Particle Physics – as Jets



Energy/Particle Flow Networks (EFNs/PFNs)

[Zaheer, Kottur, Ravanbhakhsh, Póczos, Salakhutdinov, Smola, 1703.06114;
PTK, Metodiev, Thaler, 1810.05165;
EnergyFlow Python Package]



Particle Flow Network (PFN)

$$\mathrm{PFN}(\{p_1^\mu, \ldots, p_M^\mu\}) = F\left(\sum_{i=1}^{M} \Phi(p_i^\mu)\right)$$

Fully general latent space

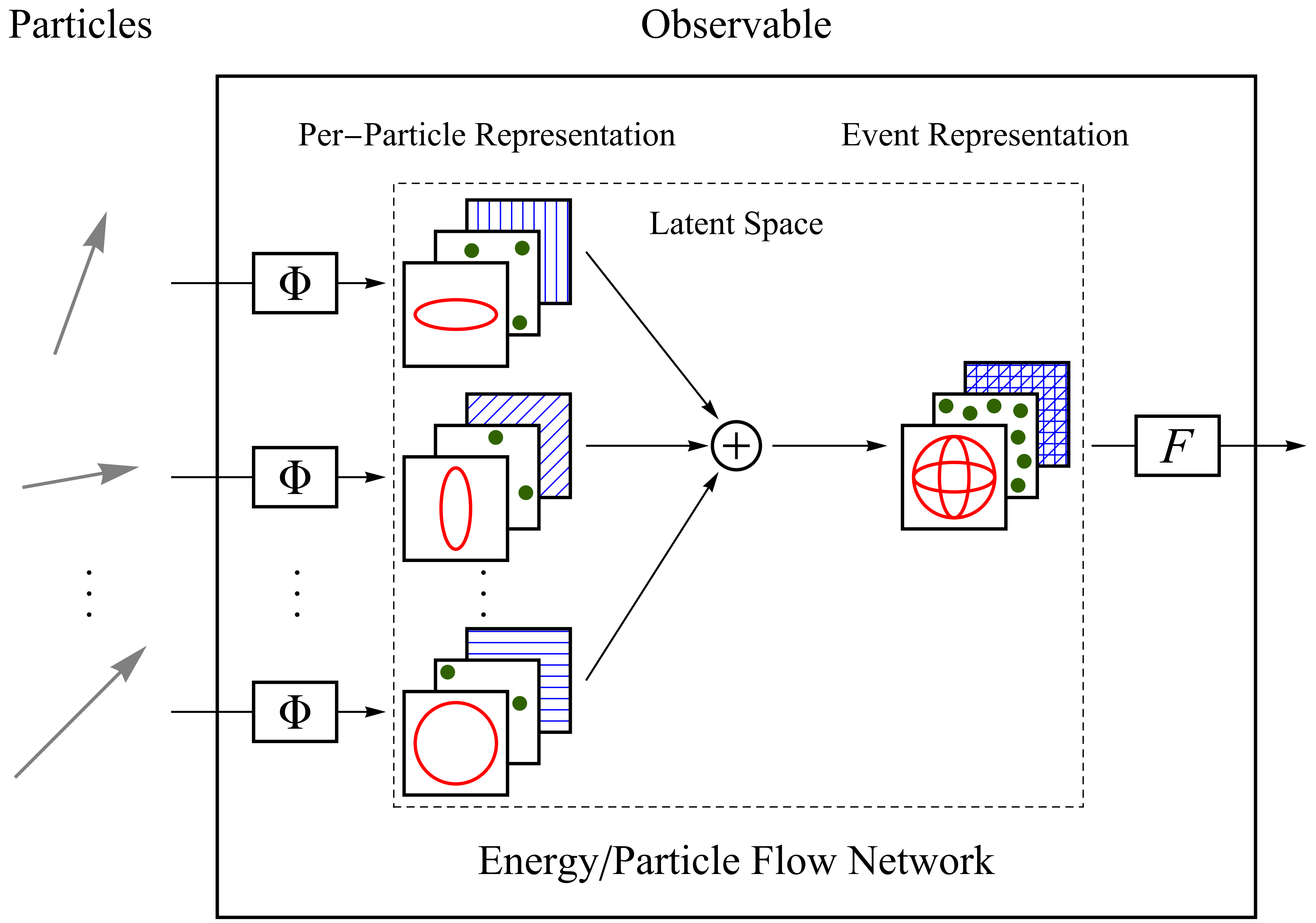
Energy Flow Network (EFN)

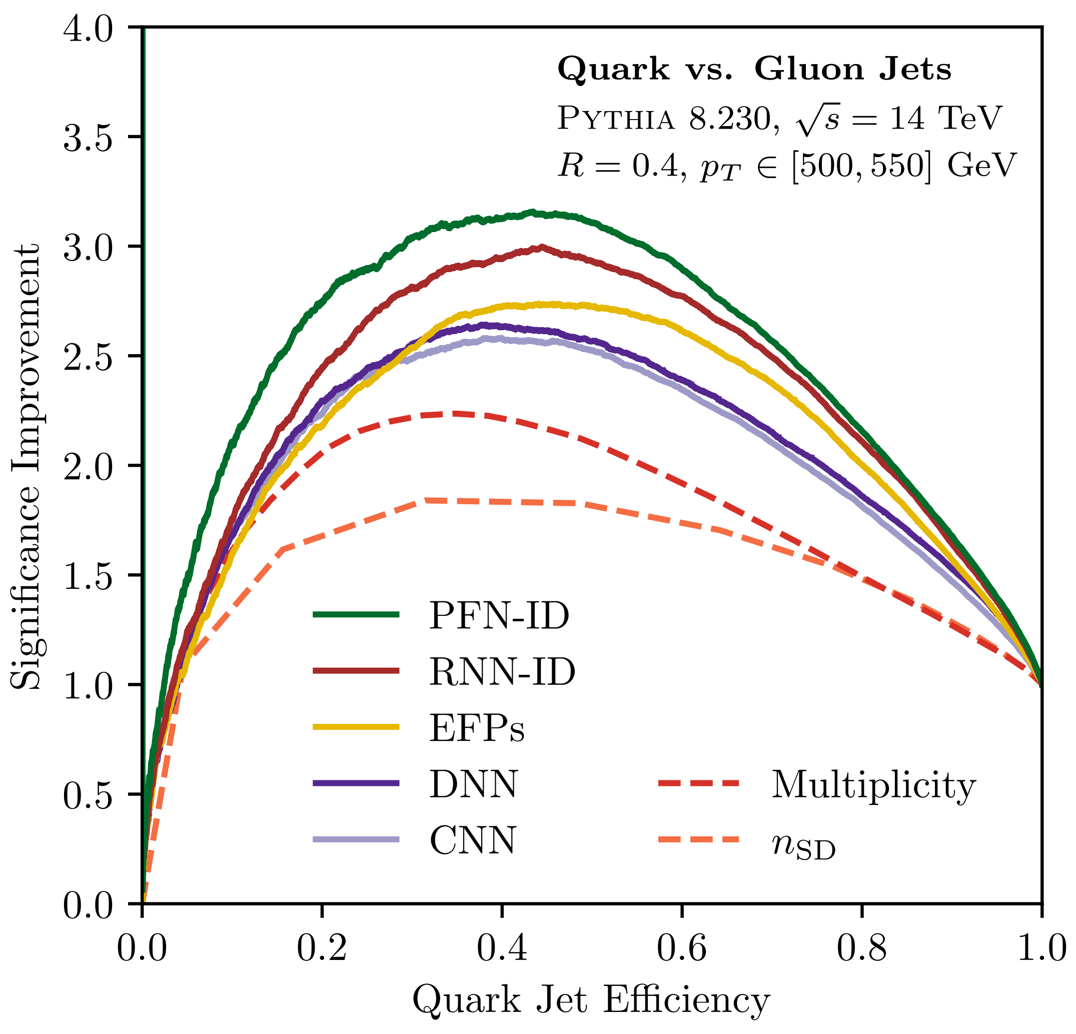
$$\mathrm{EFN}(\{p_1^\mu, \ldots, p_M^\mu\}) = F\left(\sum_{i=1}^{M} z_i \Phi(\hat{p}_i)\right)$$

IRC-safe latent space

Energy/Particle Flow Networks (EFNs/PFNs)

[Zaheer, Kottur, Ravanbhakhsh, Póczos, Salakhutdinov, Smola, 1703.06114;
PTK, Metodiev, Thaler, 1810.05165;
EnergyFlow Python Package]





Improved performance (and training) compared to RNN and CNN

Particle Flow Network (PFN)

$$\mathrm{PFN}(\{p_1^\mu, \ldots, p_M^\mu\}) = F\left(\sum_{i=1}^{M} \Phi(p_i^\mu)\right)$$

Fully general latent space

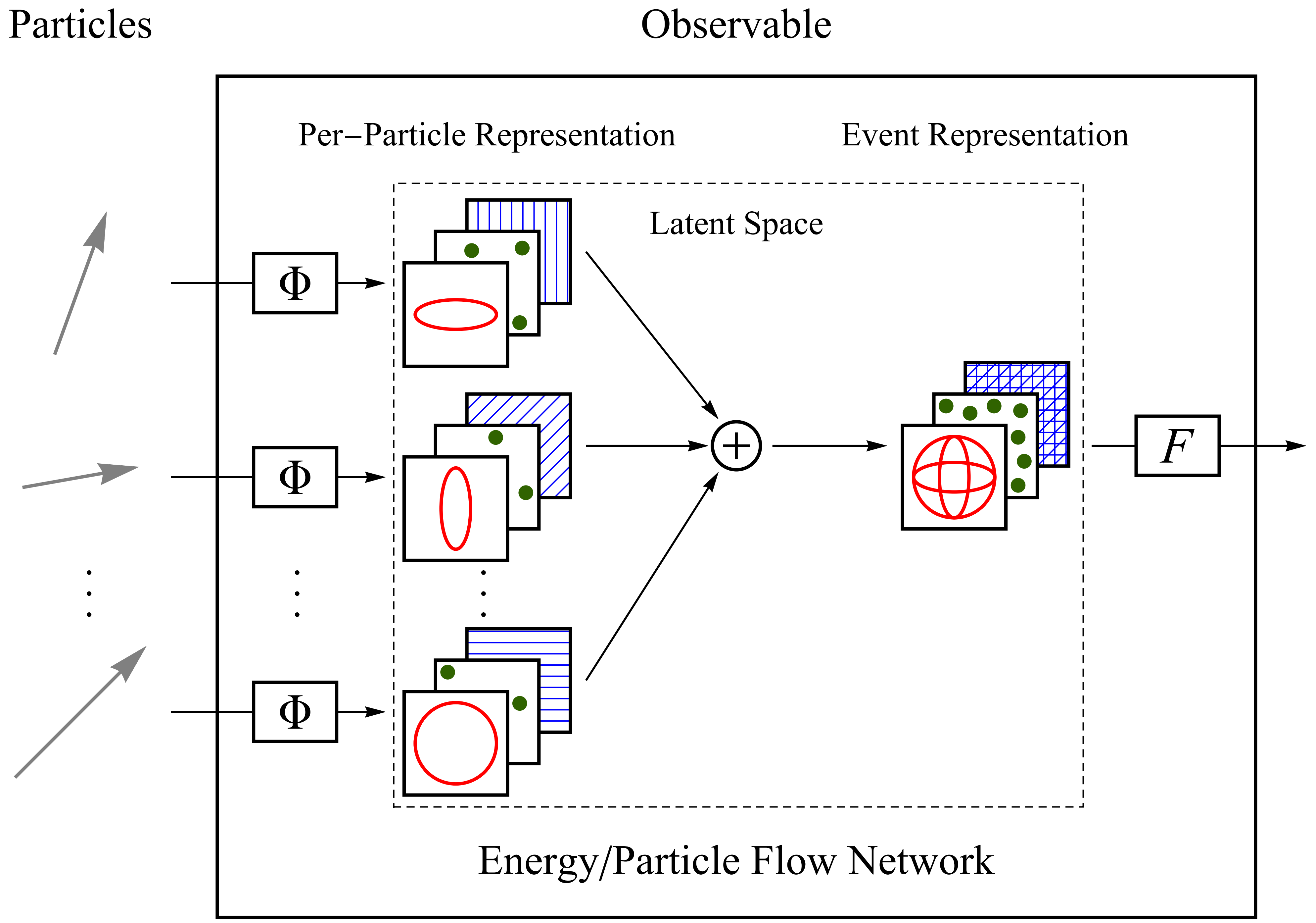
Energy Flow Network (EFN)

$$\mathrm{EFN}(\{p_1^\mu, \ldots, p_M^\mu\}) = F\left(\sum_{i=1}^{M} z_i \Phi(\hat{p}_i)\right)$$

IRC-safe latent space

Energy/Particle Flow Networks (EFNs/PFNs)

[Zaheer, Kottur, Ravanbhakhsh, Póczos, Salakhutdinov, Smola, 1703.06114;
PTK, Metodiev, Thaler, 1810.05165;
EnergyFlow Python Package]



Particle Flow Network (PFN)

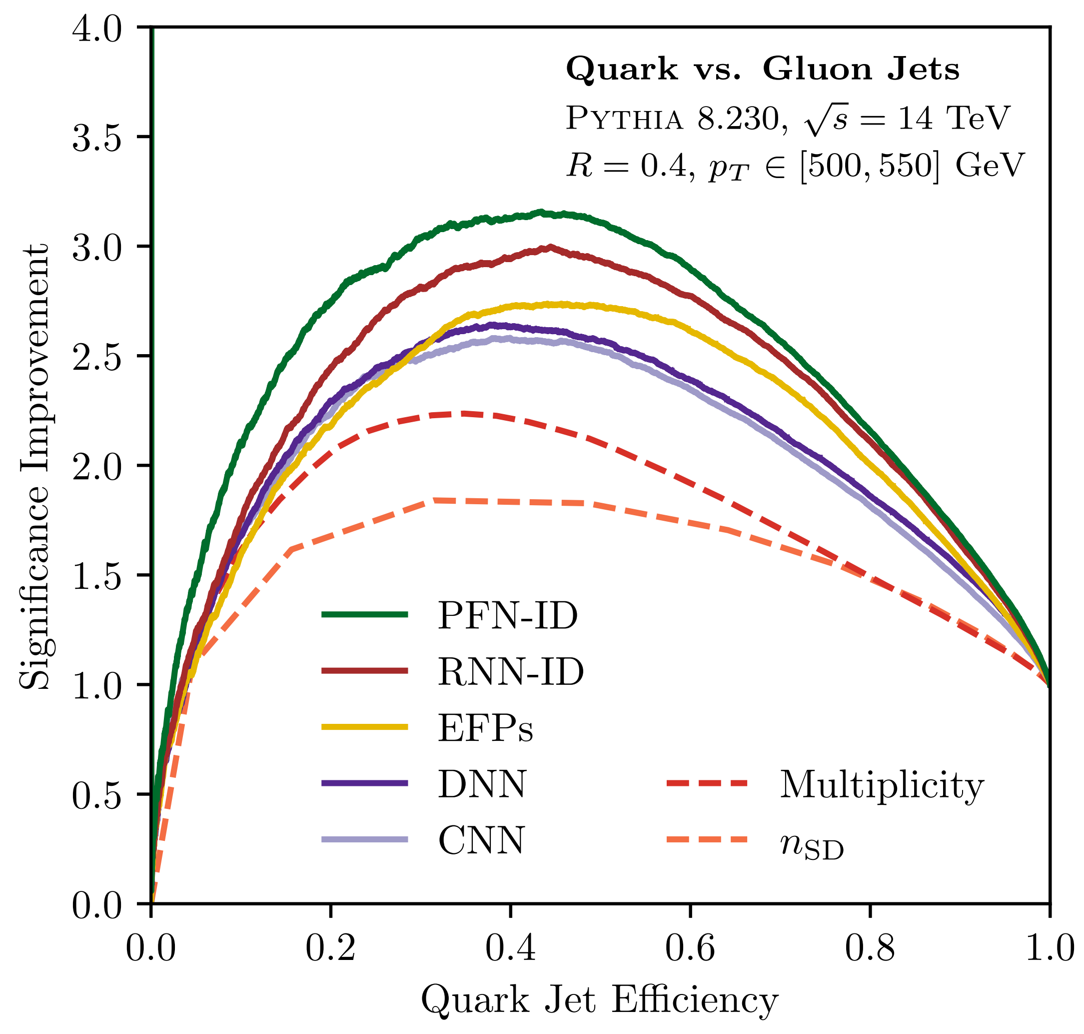
$$\mathrm{PFN}(\{p_1^\mu,\ldots,p_M^\mu\}) = F\left(\sum_{i=1}^{M}\Phi(p_i^\mu)\right)$$

Fully general latent space

Energy Flow Network (EFN)

$$\mathrm{EFN}(\{p_1^\mu,\ldots,p_M^\mu\}) = F\left(\sum_{i=1}^{M} z_i\,\Phi(\hat{p}_i)\right)$$

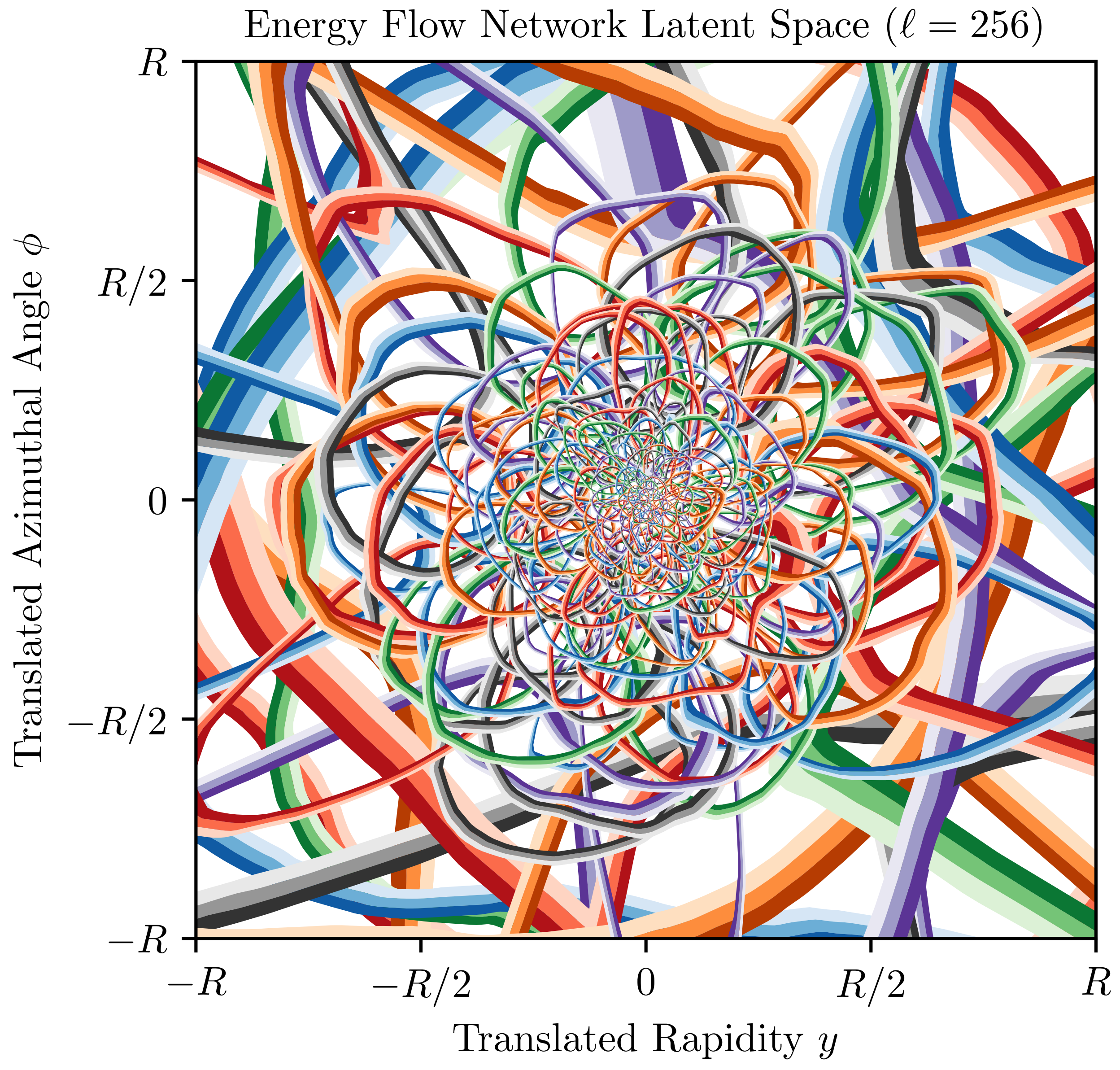
IRC-safe latent space



Improved performance (and training) compared to RNN and CNN

Latent space visualization reveals what the network has learned

Dynamic pixel sizing related to collinear singularity of QCD!



Quark vs. Gluon: Classification Performance

PFN-ID: Full particle flavor info

$(\gamma, \pi^\pm, K^\pm, K_L, p, \bar{p}, n, \bar{n}, e^\pm, \mu^\pm)$

PFN-Ex: Experimentally accessible info

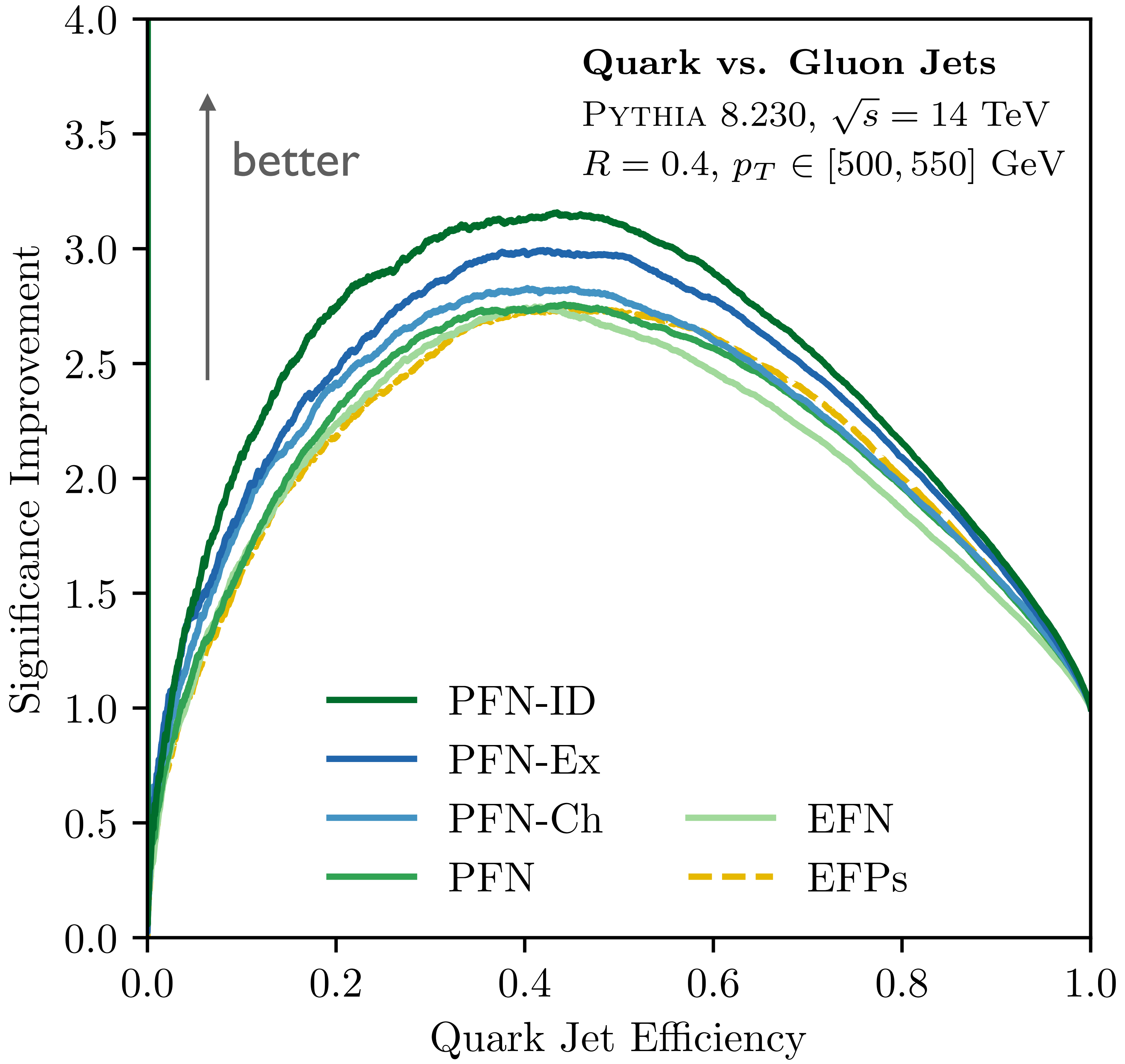
$(\gamma, h^{\pm,0}, e^\pm, \mu^\pm)$

PFN-Ch: Particle charge info

$(+,0,-)$

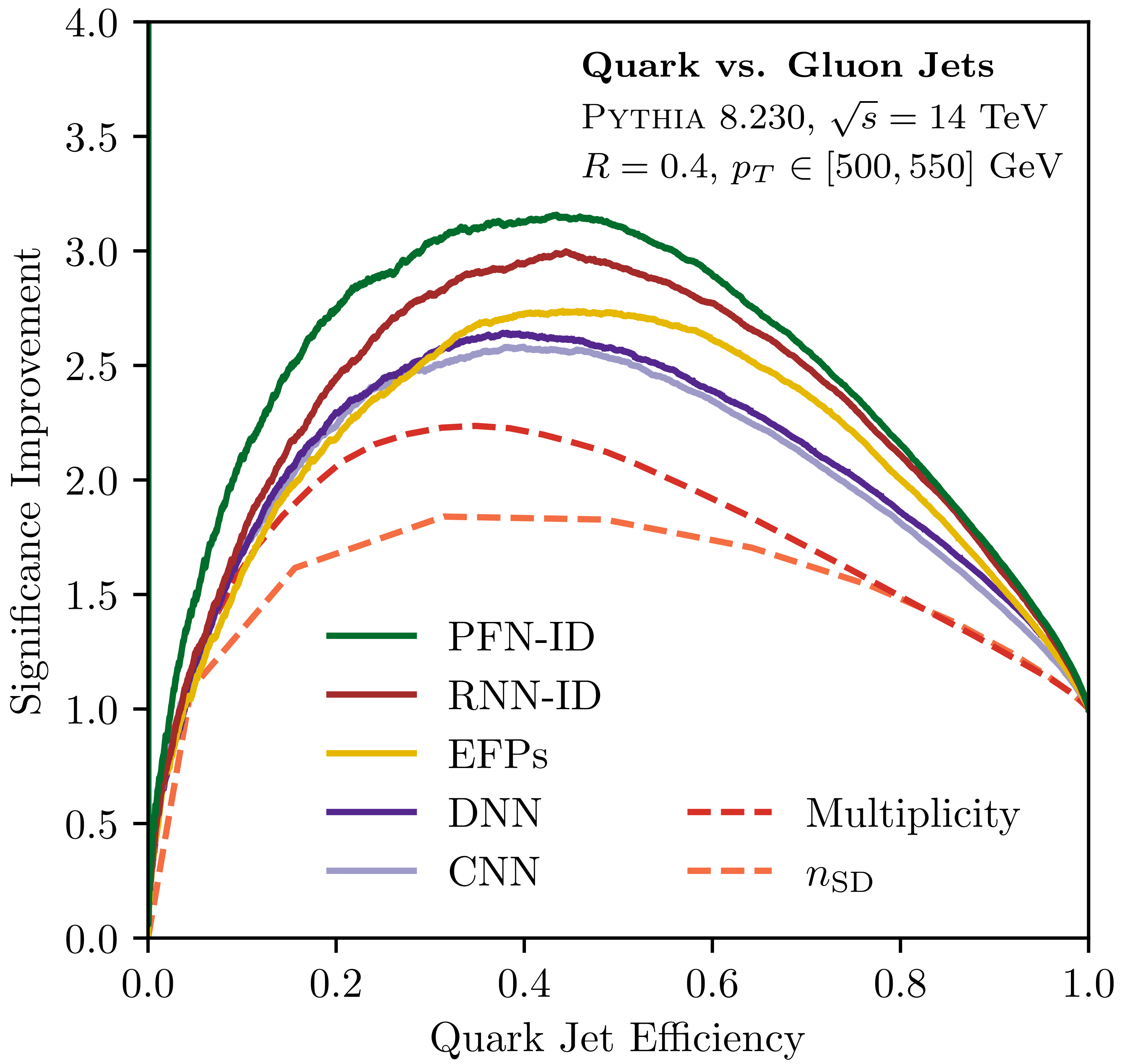
PFN: No particle type info, arbitrary energy dependence

EFN: IRC-safe latent space



Latent space dimension ℓ = 256

EFPs are comparable to EFN



PFN-ID better than RNN-ID

Quark vs. Gluon: Latent Dimension Sweep

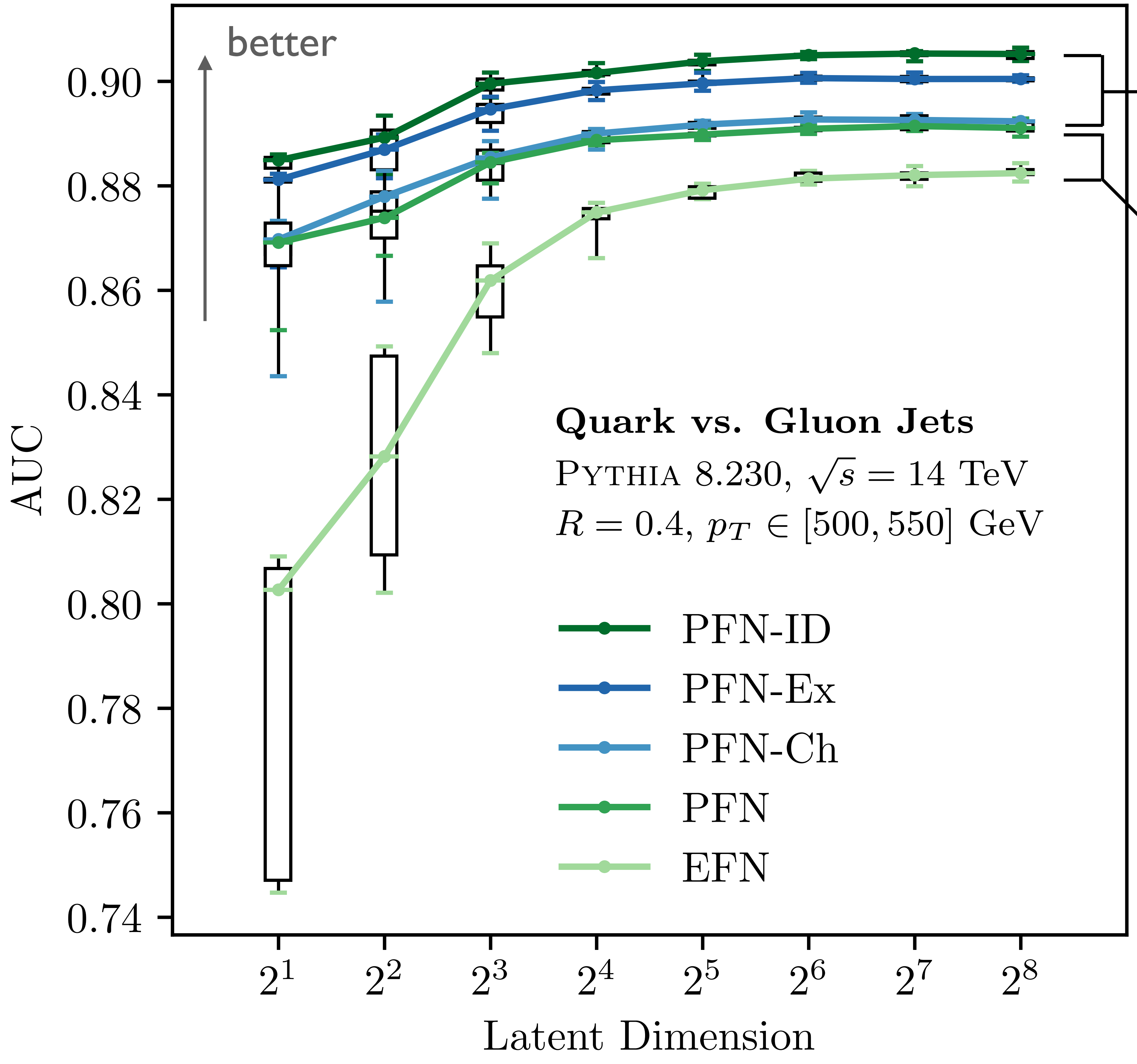
PFN-ID: Full particle flavor info
$(\gamma, \pi^\pm, K^\pm, K_L, p, \bar{p}, n, \bar{n}, e^\pm, \mu^\pm)$

PFN-Ex: Experimentally accessible info
$(\gamma, h^{\pm,0}, e^\pm, \mu^\pm)$

PFN-Ch: Particle charge info
(+,0,−)

PFN: No particle type info, arbitrary energy dependence

EFN: IRC-safe latent space



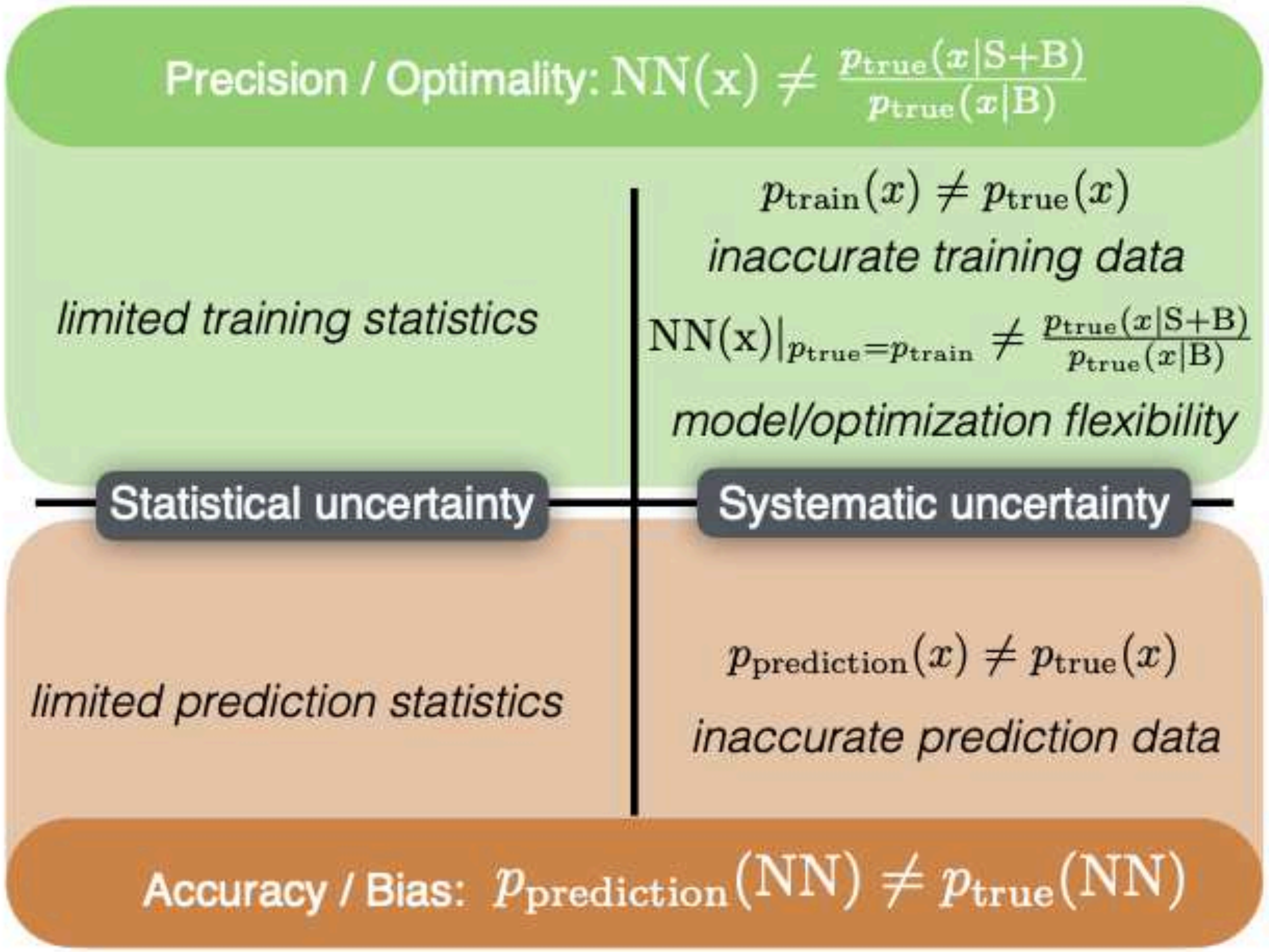
Particle type info helpful

IRC unsafe information clearly helpful

Performance saturates as dimension increases

Dealing with Uncertainties

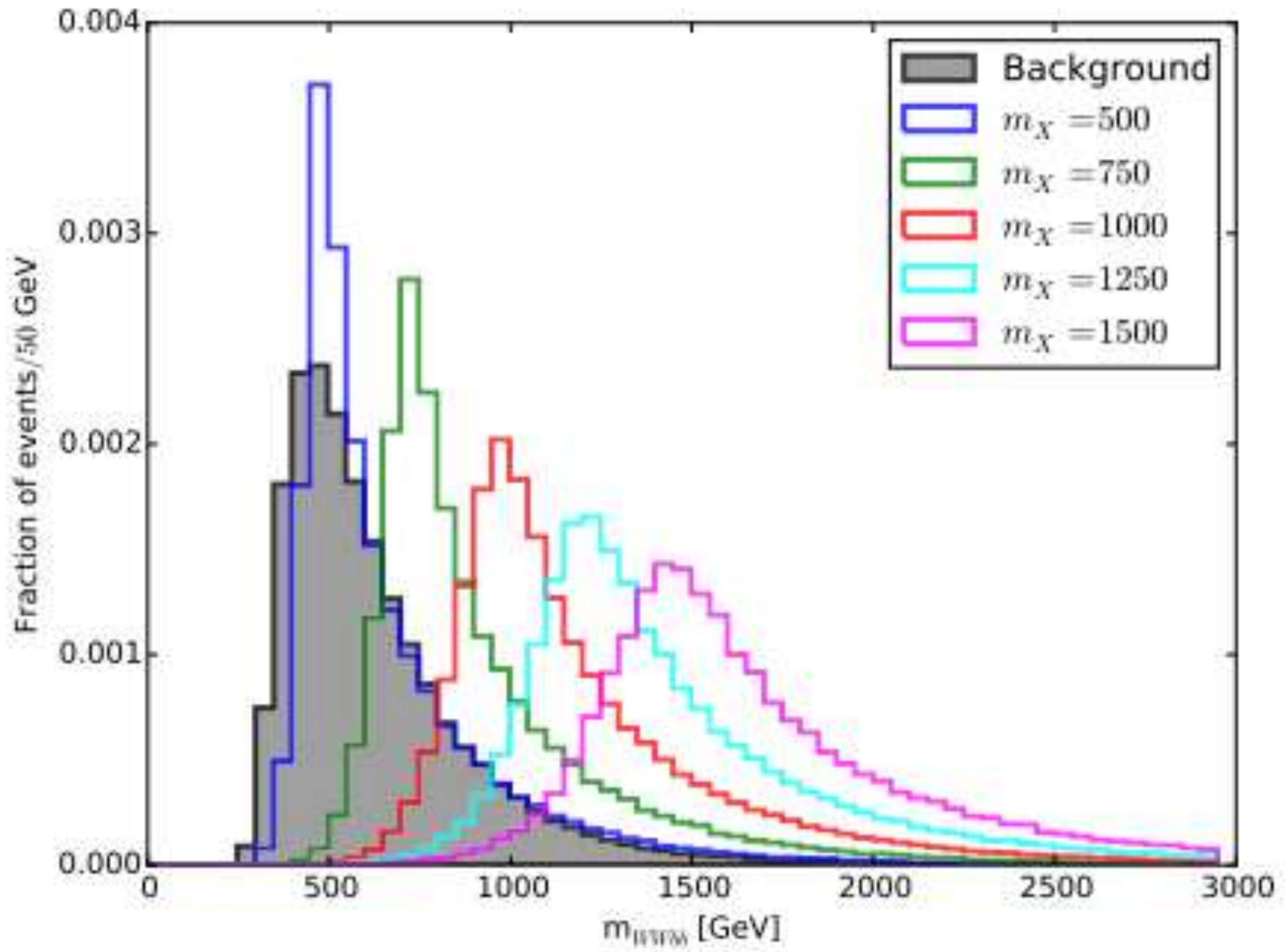
Sources of uncertainty in a statistical analysis



[Nachman, 1909.03081]

Parametrized models could enable efficient profiling to handle systematic uncertainties

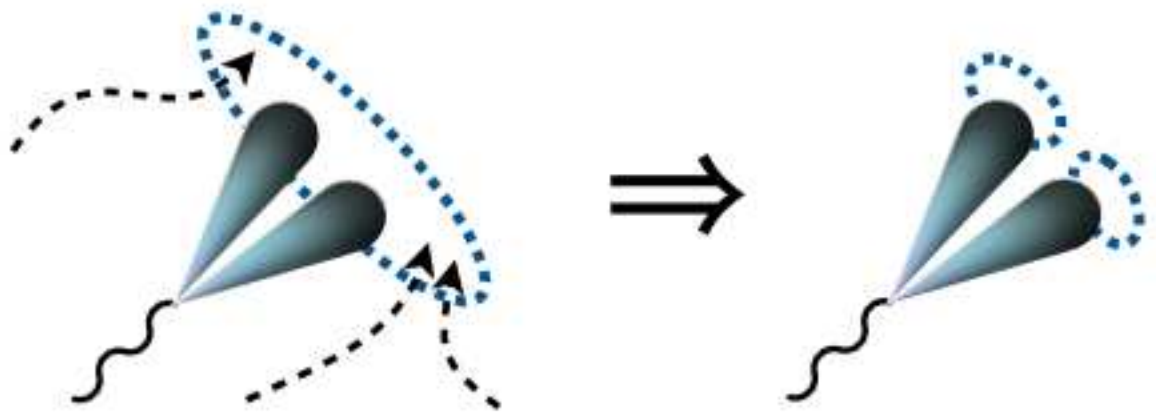
[similar to Baldi, Cranmer, Faucett, Sadowski, Whiteson, 1601.07913]



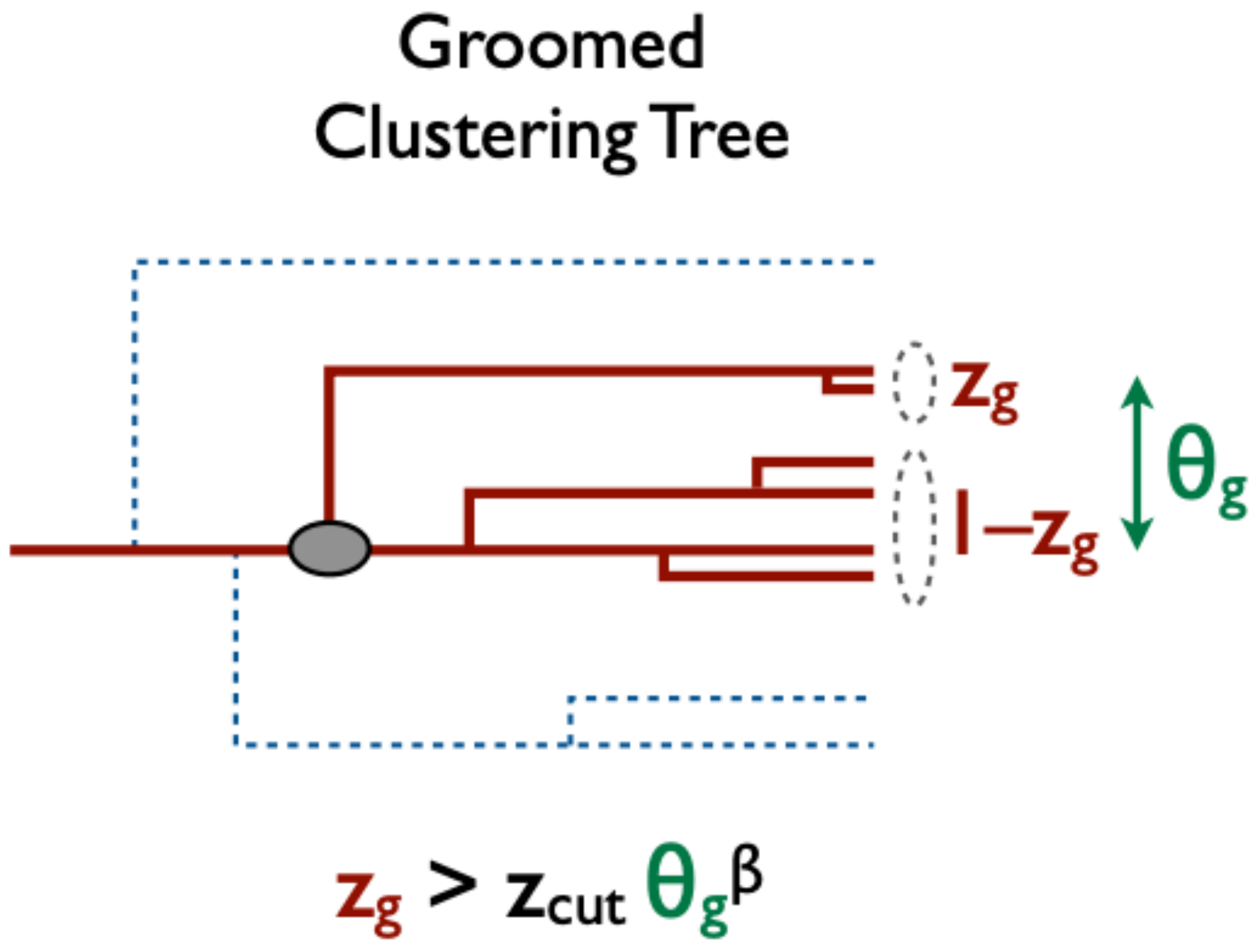
Training a neural network on several different signal masses and allowing it to interpolate between them

Soft Drop

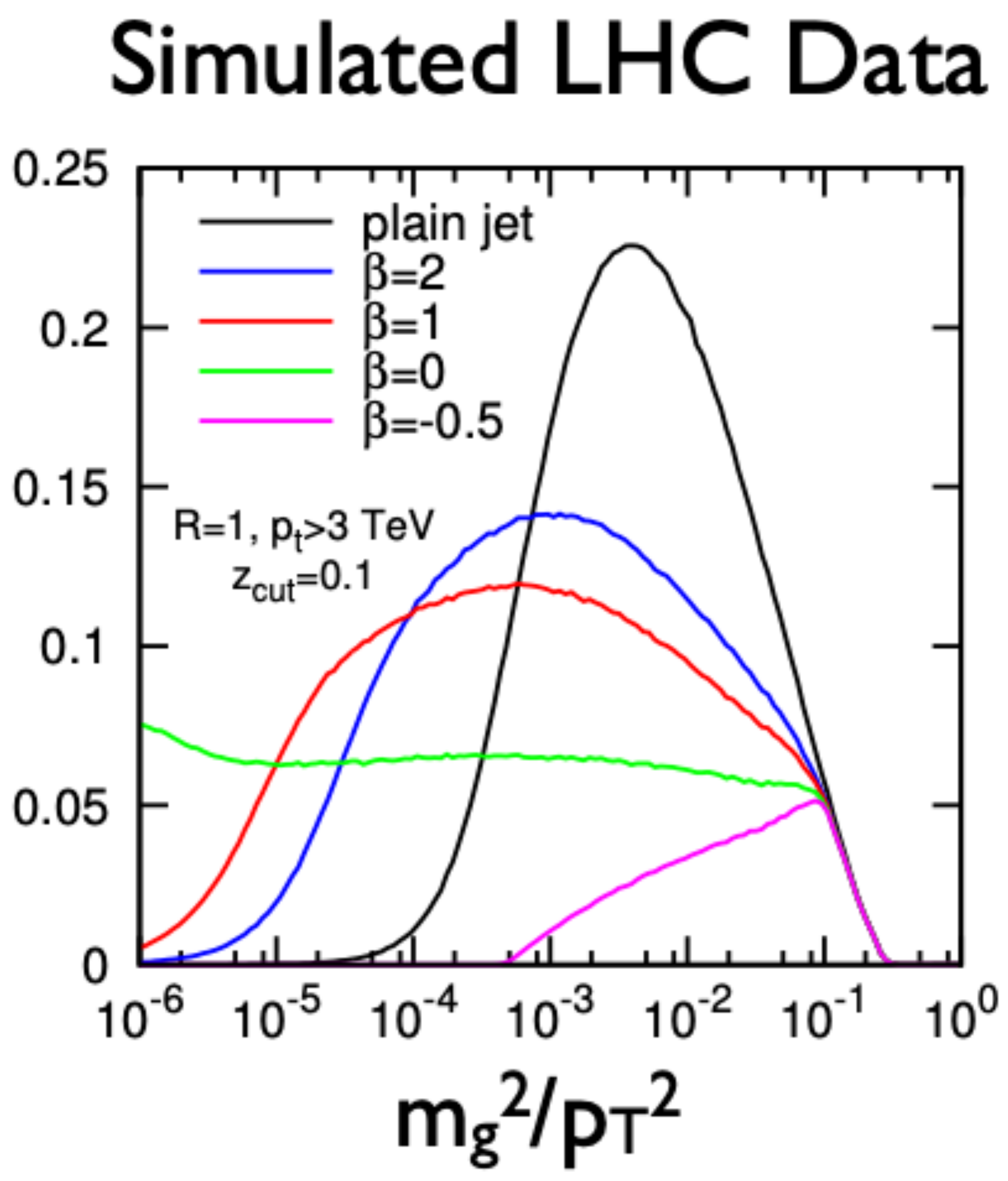
Contaminating radiation in jets necessitates grooming



Soft Drop algorithm



Calculating mass on SD jets

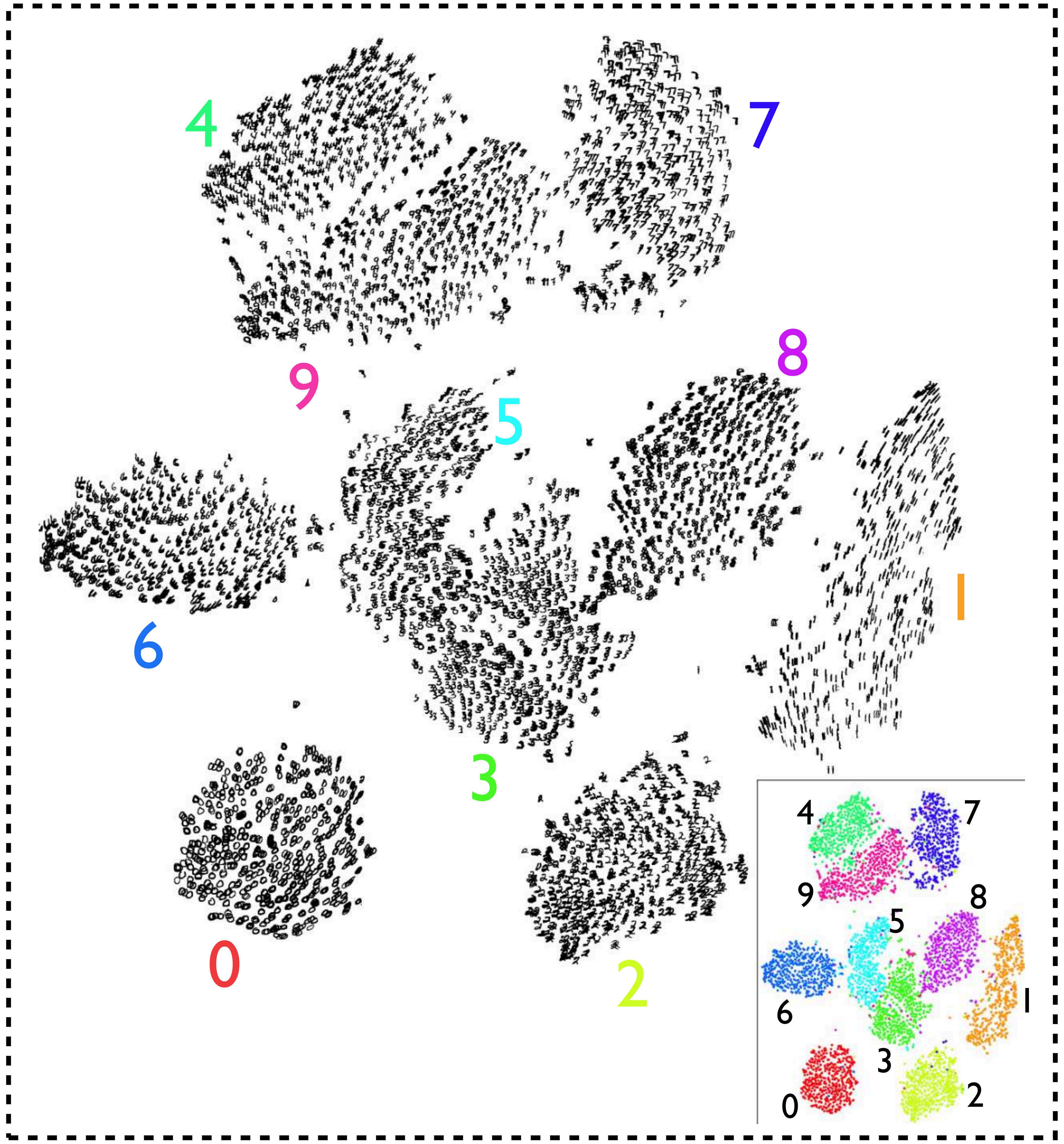


Diagrams by Jesse Thaler

Visualizing Geometry in the Space of Events

t-Distributed Stochastic Neighbor Embedding (t-SNE)

MNIST handwritten digits

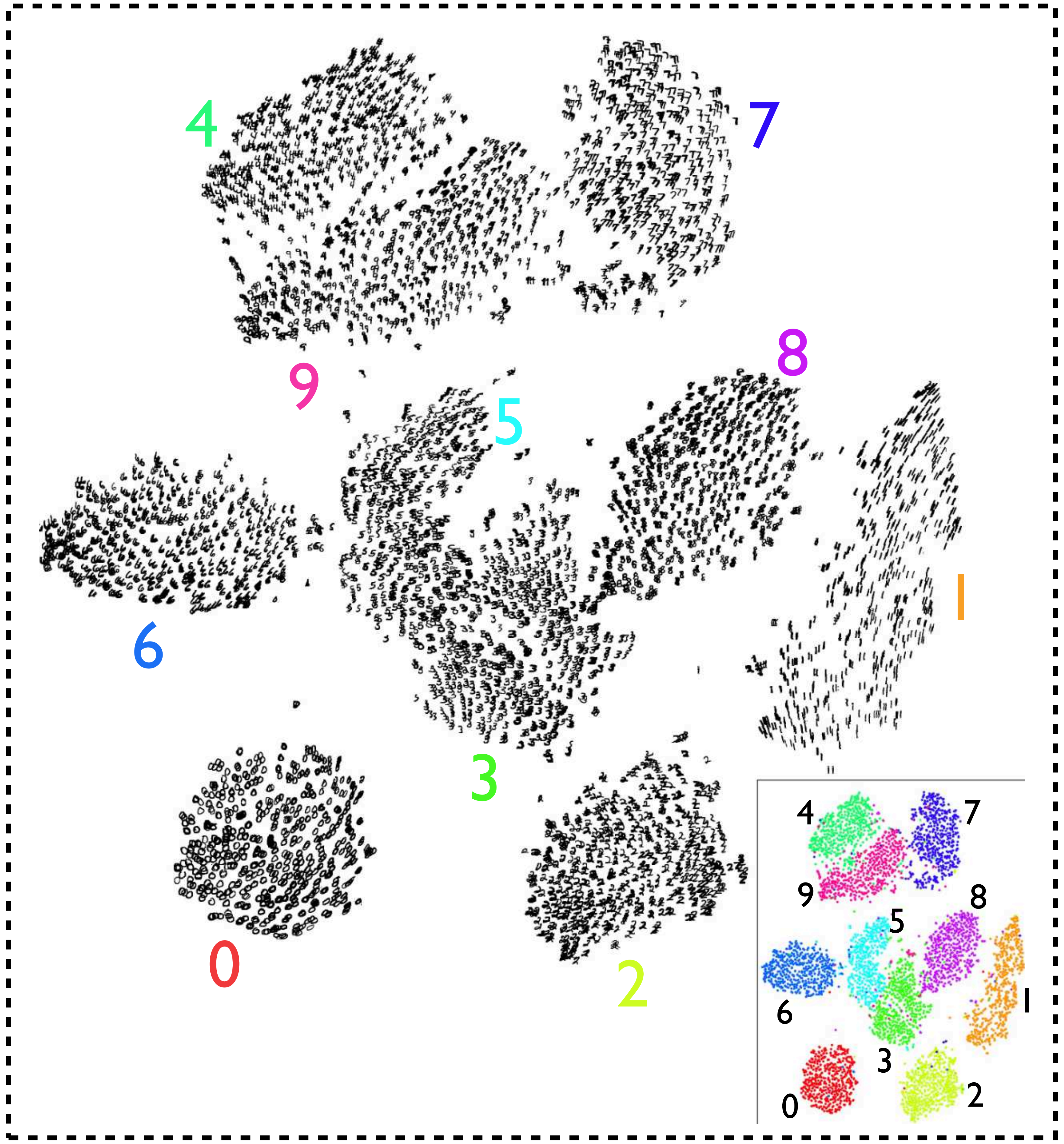


[L. van der Maaten, G. Hinton, JMLR 2008]

Visualizing Geometry in the Space of Events

t-Distributed Stochastic Neighbor Embedding (t-SNE)

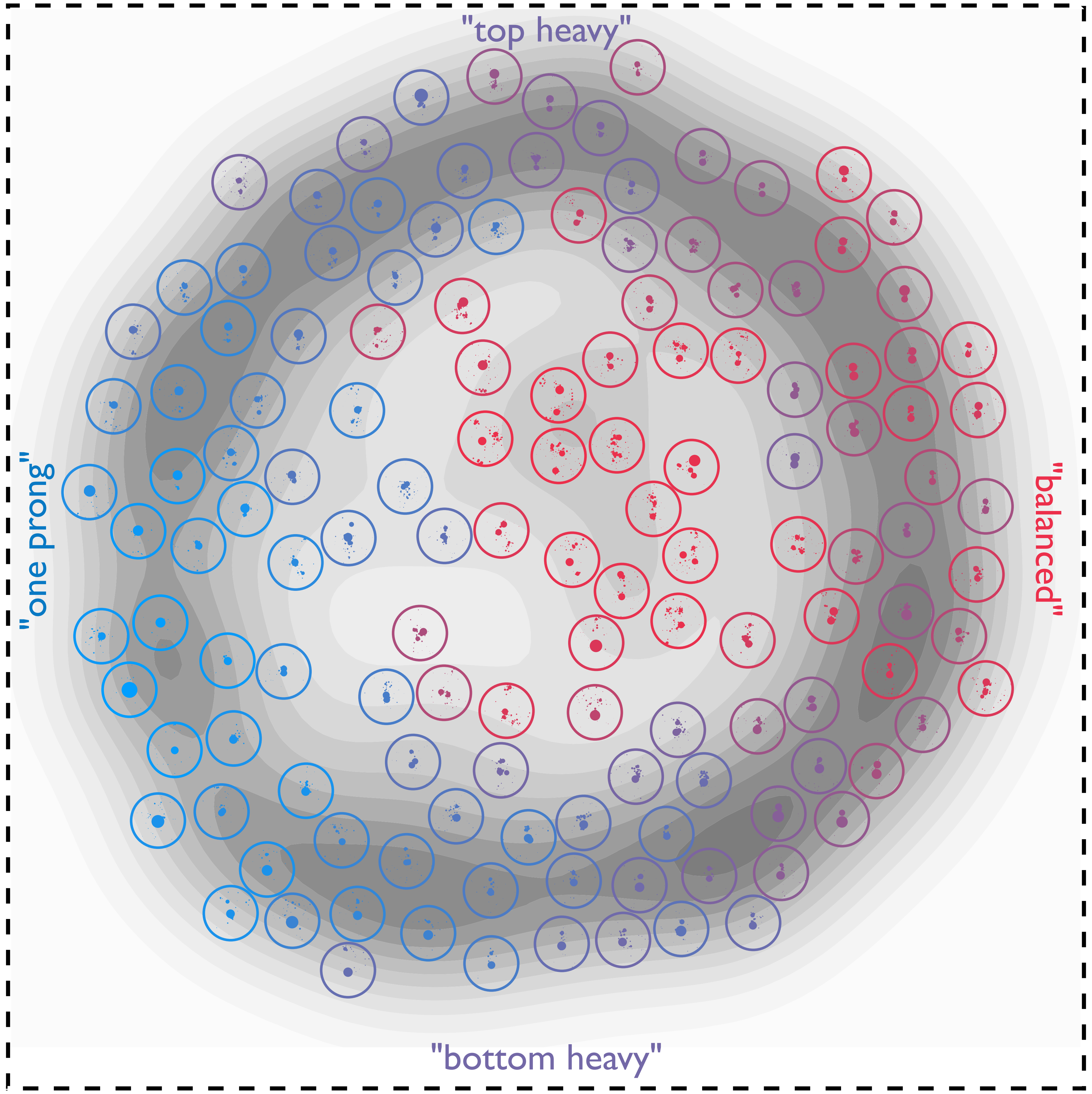
MNIST handwritten digits

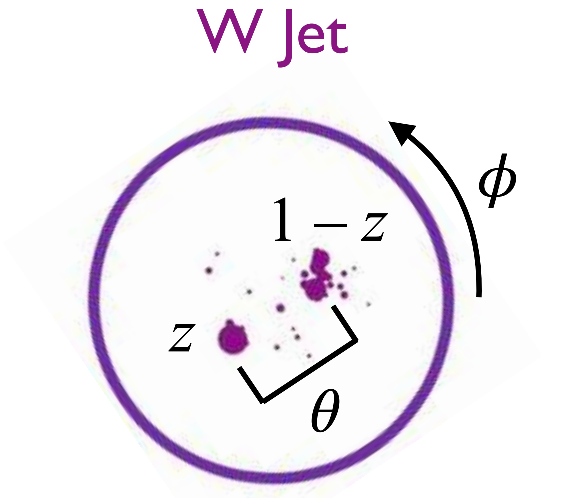


[L. van der Maaten, G. Hinton, JMLR 2008]

[PTK, Metodiev, Thaler, PRL 2019]

Geometric space of W jets



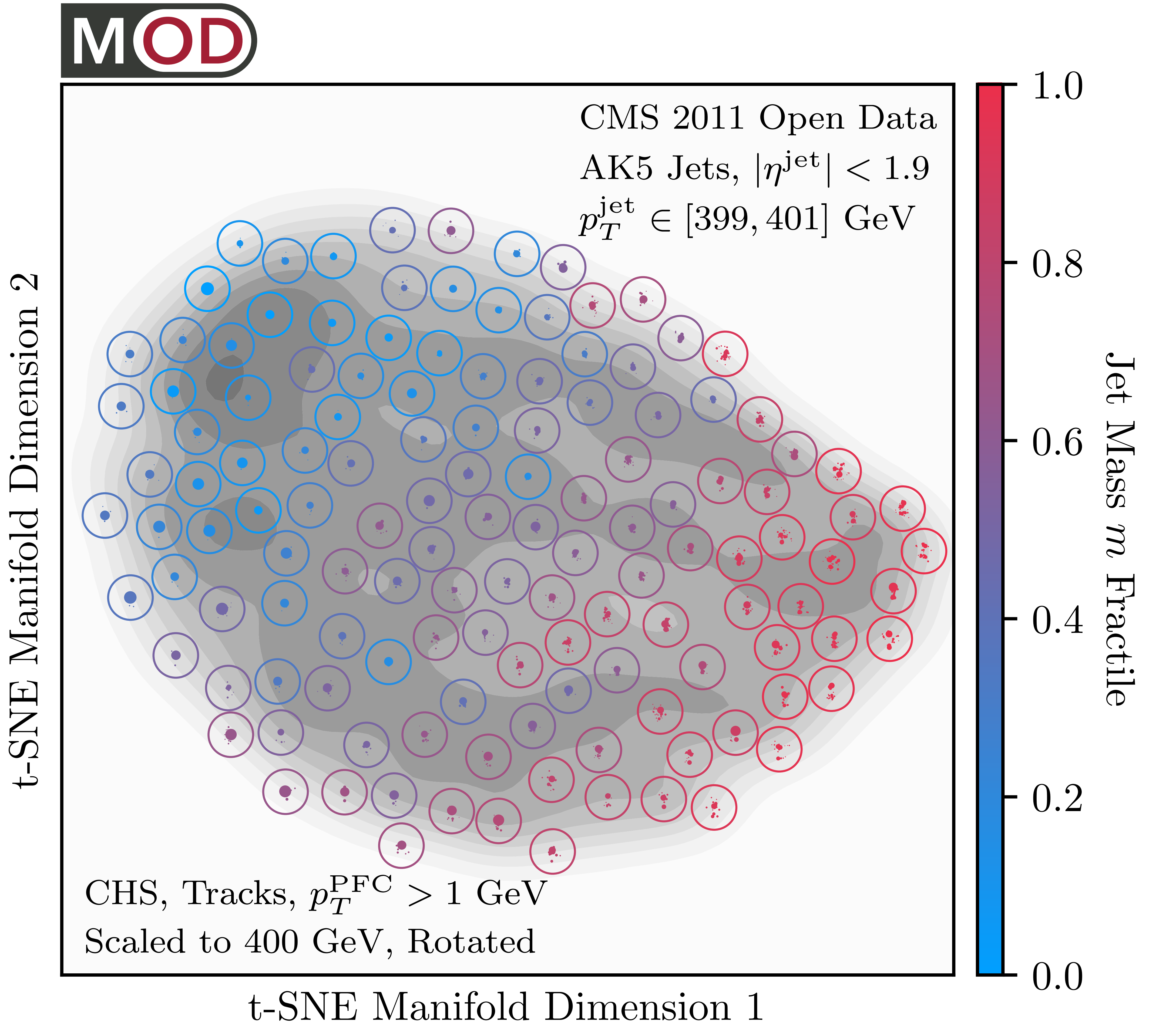


Constraints: W Mass and $\phi = 0$ preprocessing

Gray contours represent the density of jets
Each circle is a particular W jet

Visualizing Geometry in CMS Open Data

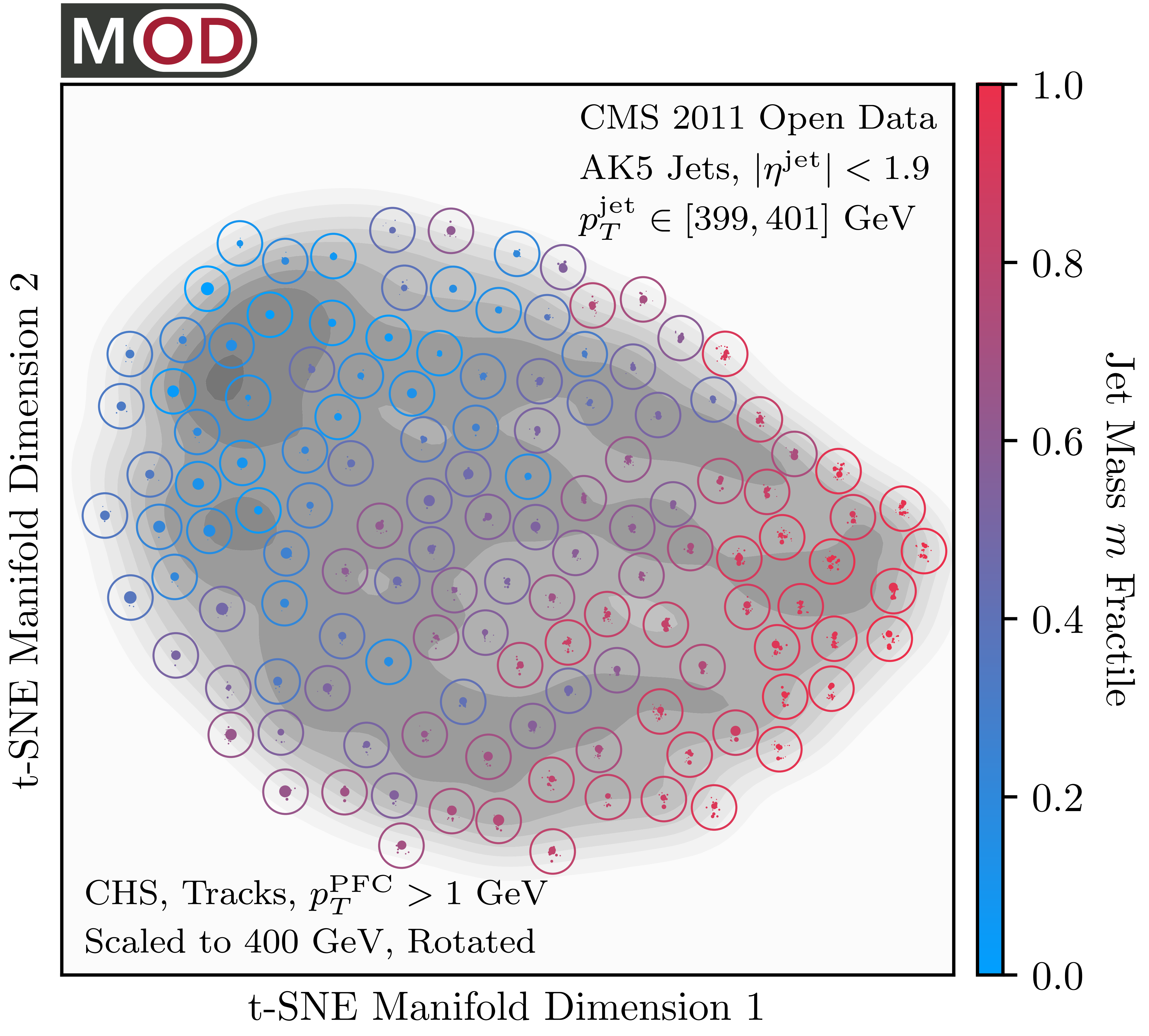
[PTK, Mastandrea, Metodiev, Naik, Thaler, PRD 2019; code and datasets at energyflow.network]



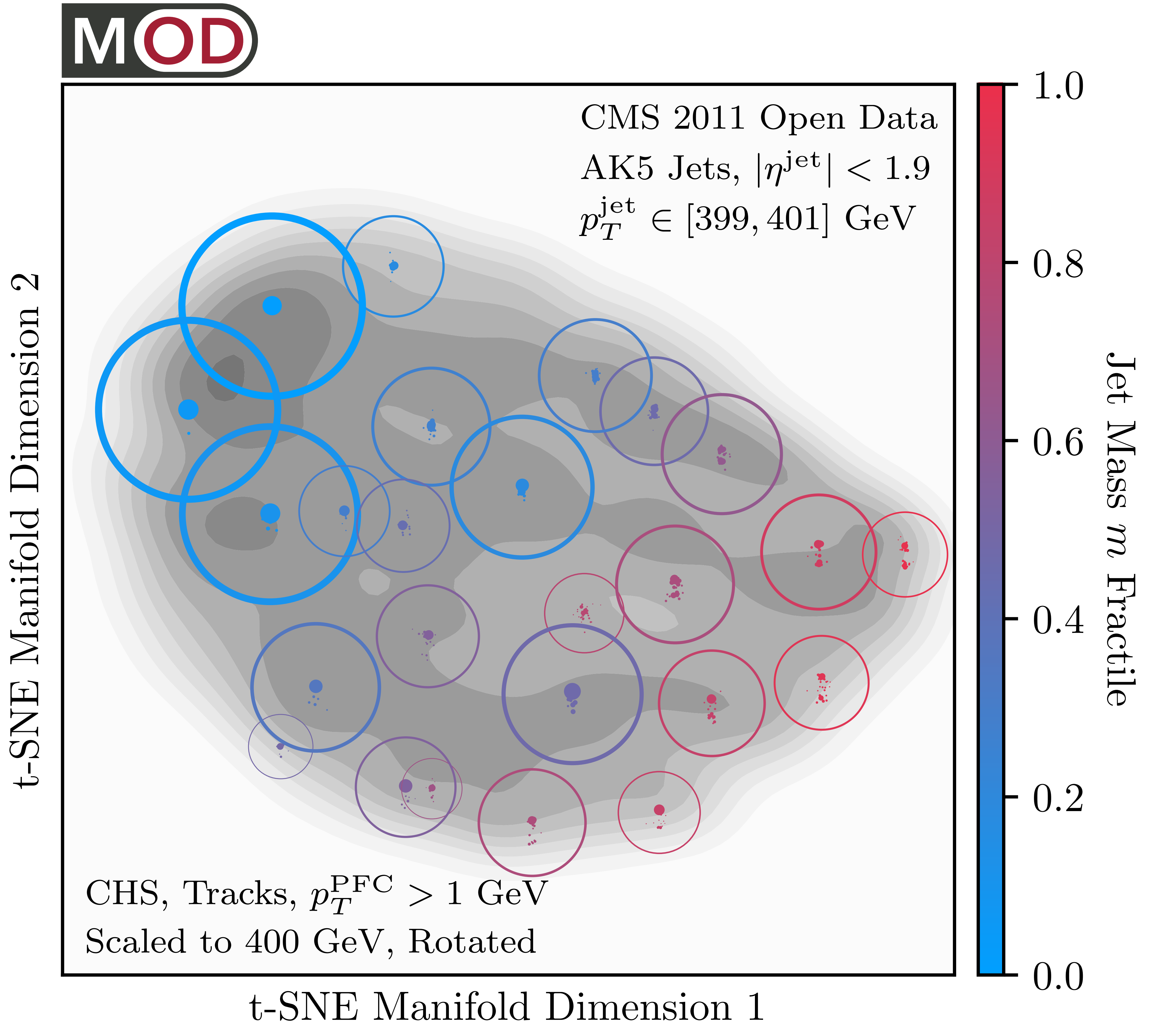
Example jets sprinkled throughout

Visualizing Geometry in CMS Open Data

[PTK, Mastandrea, Metodiev, Naik, Thaler, PRD 2019; code and datasets at energyflow.network]



Example jets sprinkled throughout

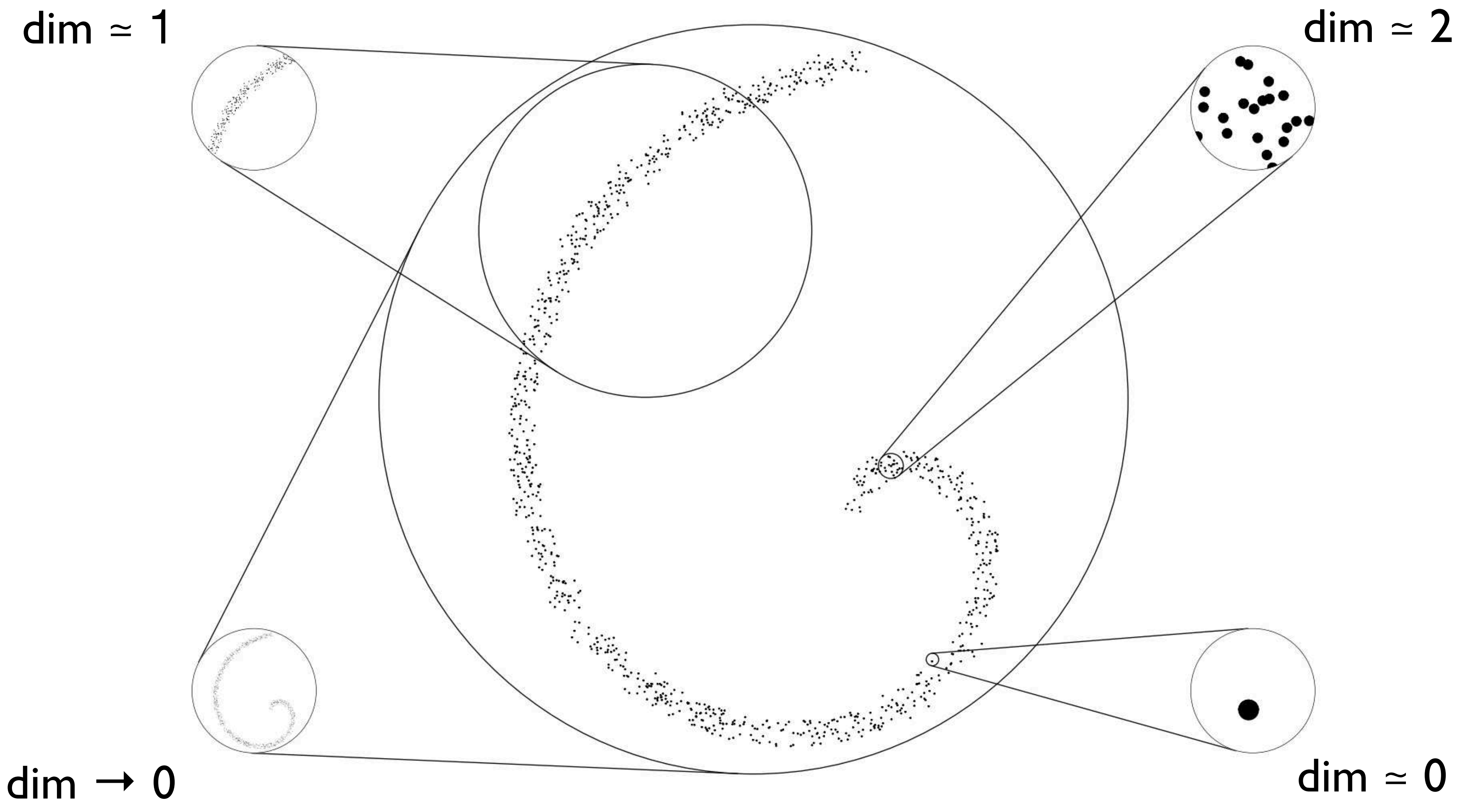


25 most representative jets ("medoids")
Size is proportional to number of jets associated to that medoid

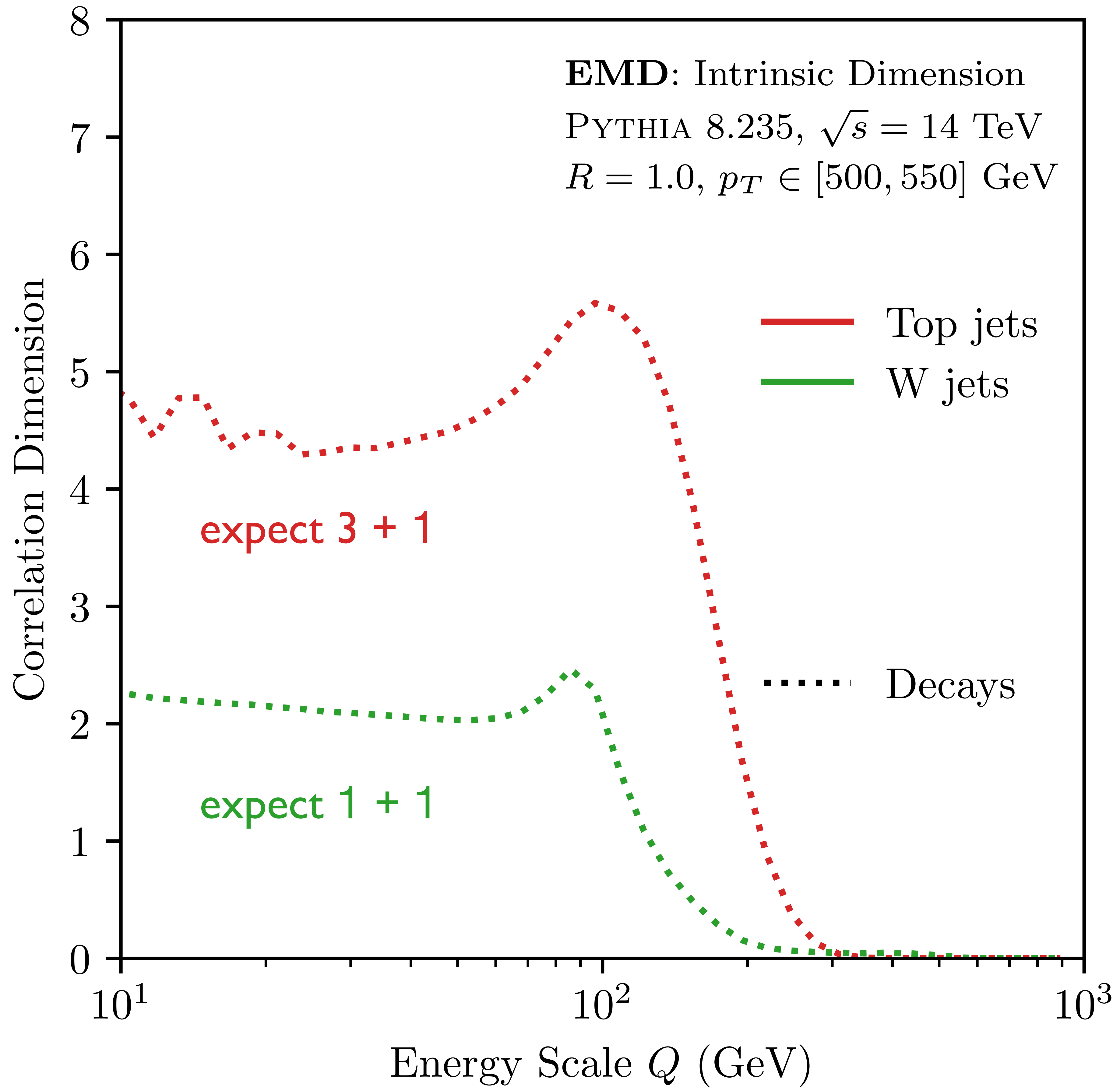
Quantifying Event-Space Manifolds

$$\dim(Q) = Q\frac{\partial}{\partial Q}\ln\sum_i\sum_j\Theta(\mathrm{EMD}(\mathcal{E}_i,\mathcal{E}'_j) < Q)$$

Correlation dimension: *how does the # of elements within a ball of size Q change?*



$$N_{\mathrm{neigh.}}(Q) \propto Q^{\dim} \implies \dim(Q) = Q\frac{d}{dQ}\ln N_{\mathrm{neigh.}}(Q)$$



Correlation dimension lessons:

Decays are "constant" dim. at low Q

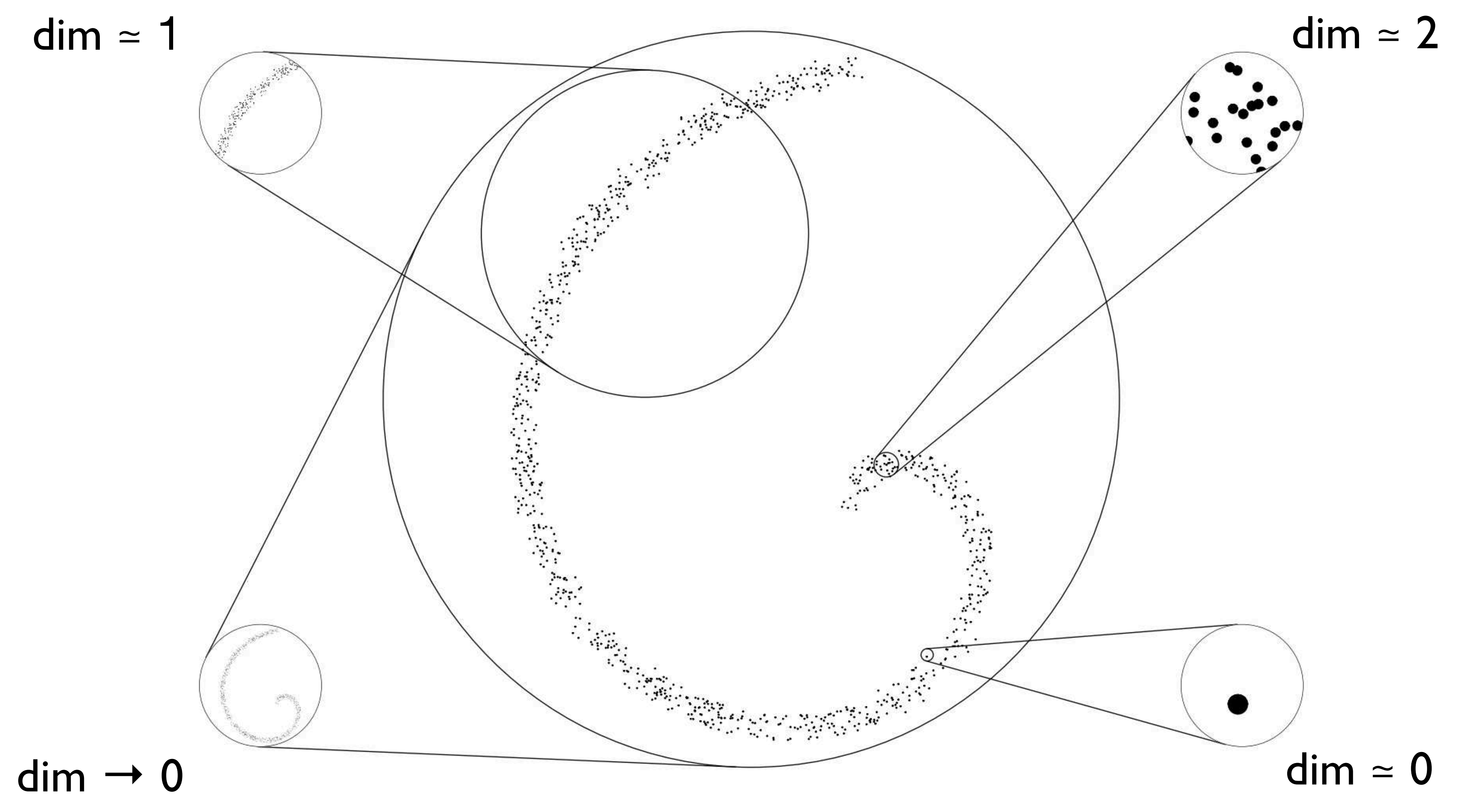
[Grassberger, Procaccia, PRL 1983; PTK, Metodiev, Thaler, PRL 2019]

*Preliminary LL calculation for QG jets in backup

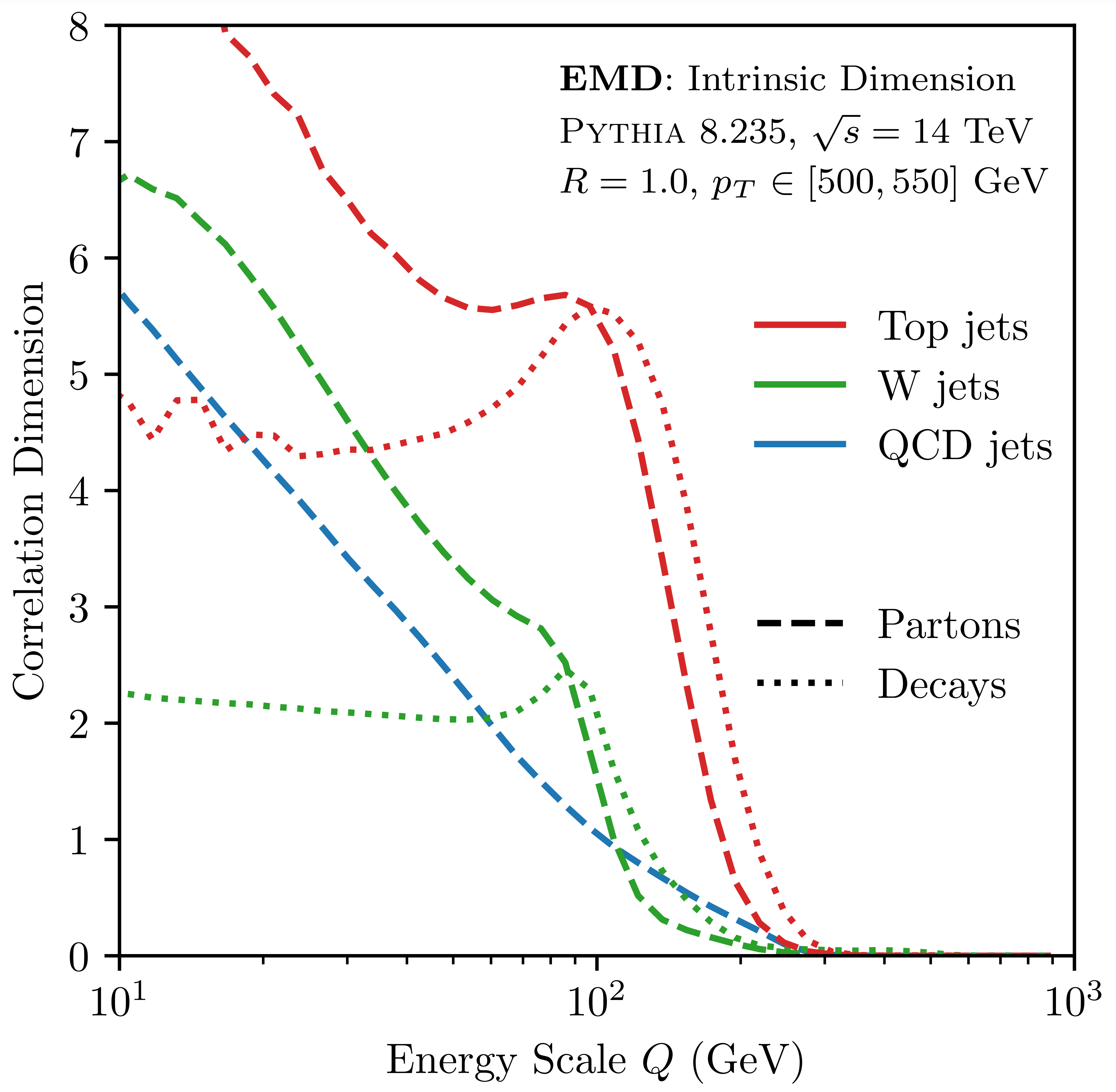
Quantifying Event-Space Manifolds

$$\dim(Q) = Q\frac{\partial}{\partial Q}\ln\sum_i\sum_j\Theta(\mathrm{EMD}(\mathcal{E}_i,\mathcal{E}'_j) < Q)$$

Correlation dimension: *how does the # of elements within a ball of size Q change?*



$$N_{\text{neigh.}}(Q) \propto Q^{\dim} \implies \dim(Q) = Q\frac{d}{dQ}\ln N_{\text{neigh.}}(Q)$$



Correlation dimension lessons:

- Decays are "constant" dim. at low Q
- Complexity hierarchy: QCD < W < Top
- Fragmentation increases dim. at smaller scales

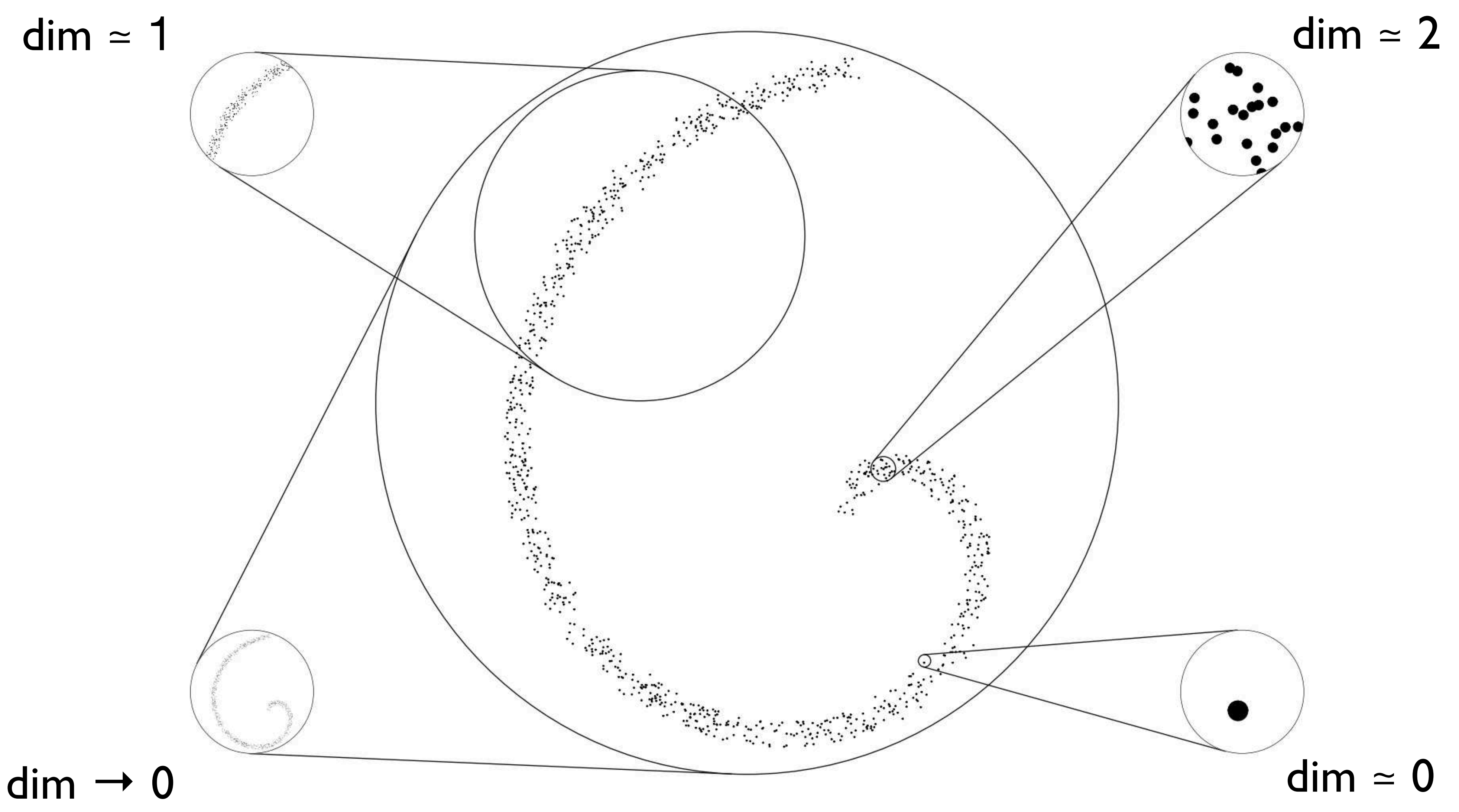
[Grassberger, Procaccia, PRL 1983; PTK, Metodiev, Thaler, PRL 2019]

*Preliminary LL calculation for QG jets in backup

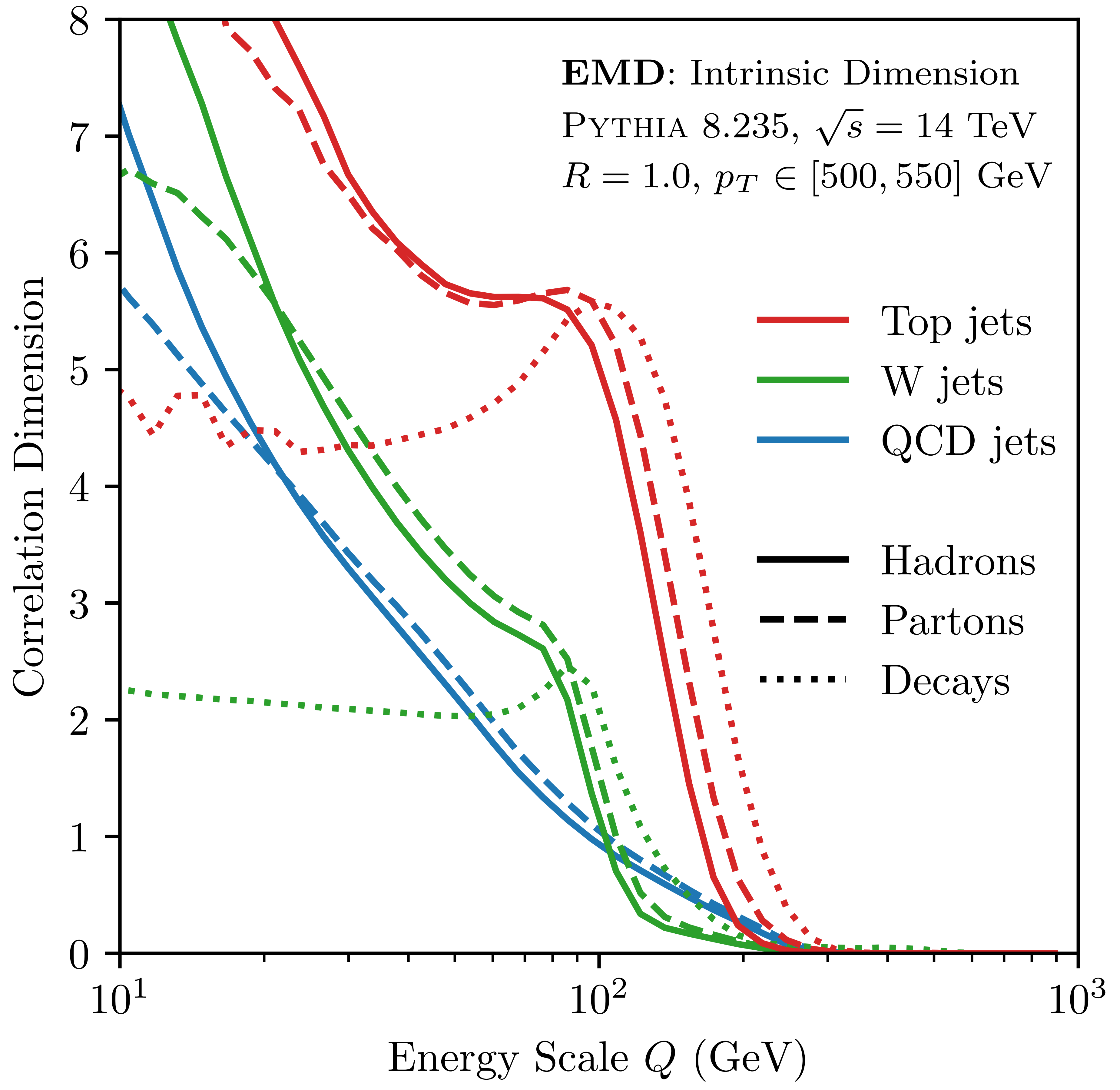
Quantifying Event-Space Manifolds

$$\dim(Q) = Q\frac{\partial}{\partial Q}\ln \sum_i \sum_j \Theta(\text{EMD}(\mathcal{E}_i, \mathcal{E}'_j) < Q)$$

Correlation dimension: *how does the # of elements within a ball of size Q change?*



$$N_{\text{neigh.}}(Q) \propto Q^{\dim} \implies \dim(Q) = Q\frac{d}{dQ}\ln N_{\text{neigh.}}(Q)$$



Correlation dimension lessons:

- Decays are "constant" dim. at low Q
- Complexity hierarchy: QCD < W < Top
- Fragmentation increases dim. at smaller scales
- Hadronization important around 20-30 GeV

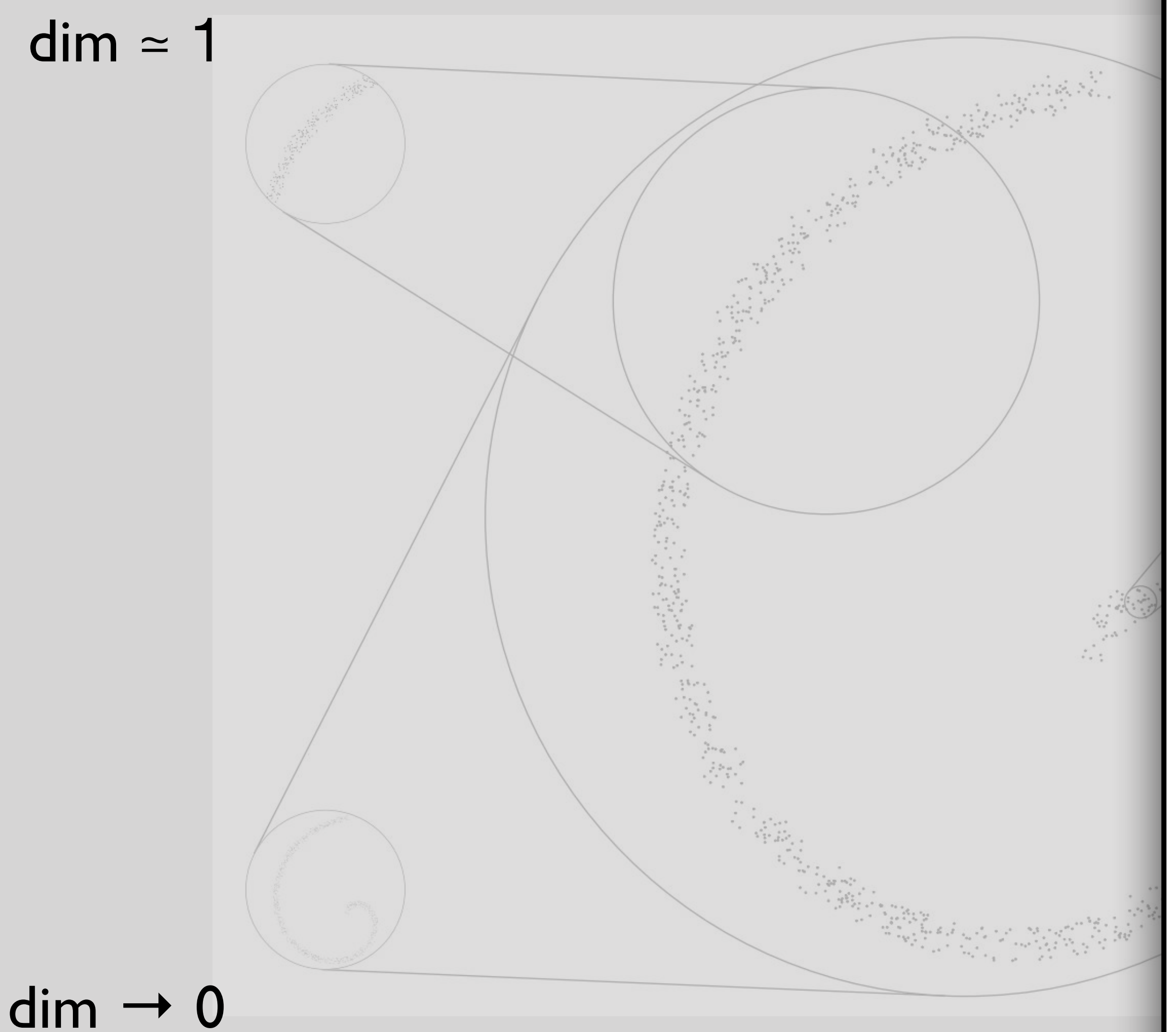
[Grassberger, Procaccia, PRL 1983; PTK, Metodiev, Thaler, PRL 2019]

*Preliminary LL calculation for QG jets in backup

Quantifying Event-Space Manifolds

$$\dots \sum_j \Theta(\mathrm{EMD}(\mathcal{E}_i, \mathcal{E}'_j) < Q)$$

Correlation dimension: *how ... elements within a ball of si...*



$N_{\mathrm{neigh.}}(Q) \propto Q^{\mathrm{dim}} \implies \mathrm{dim}(Q$

Correlation dimension

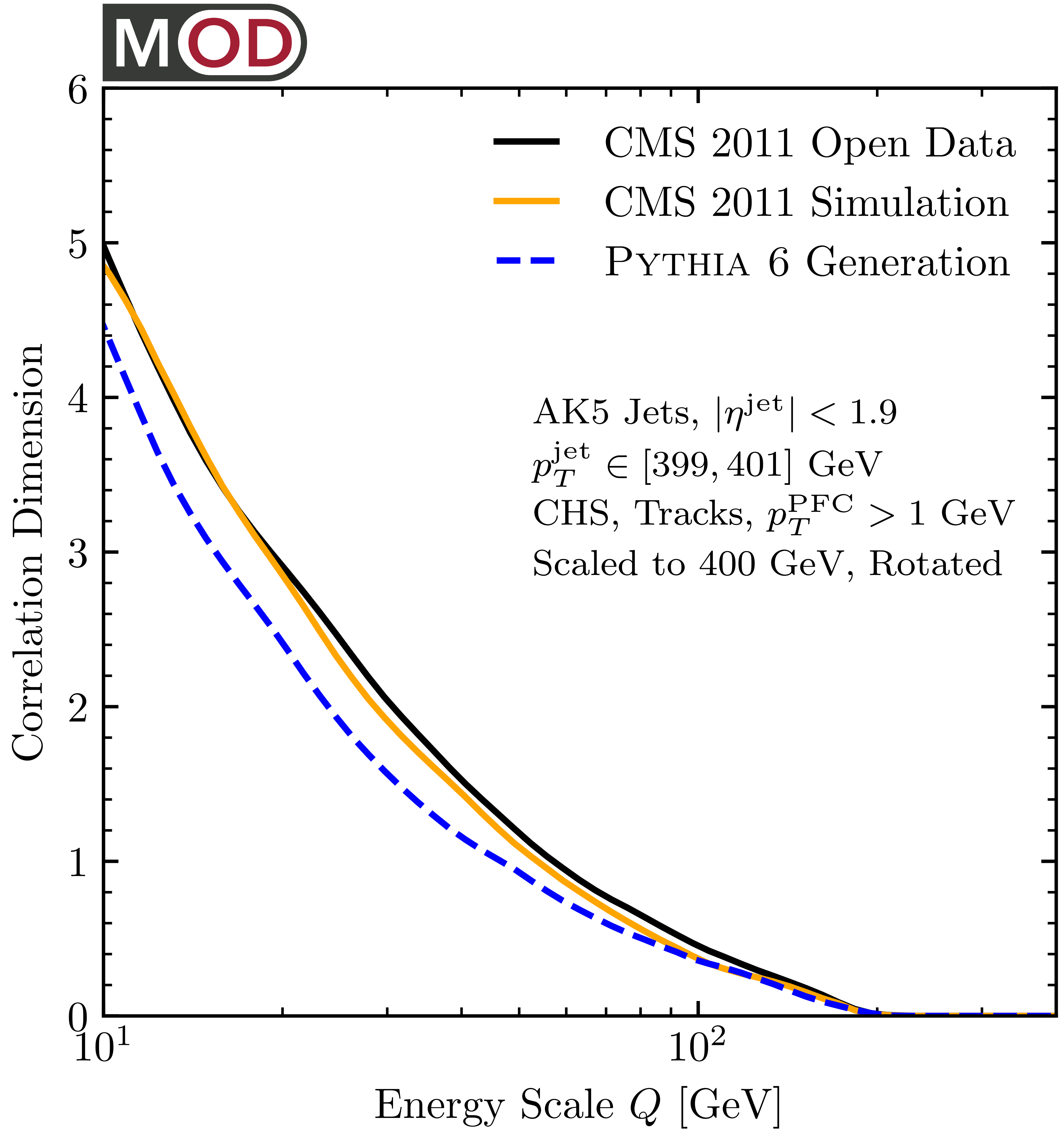
Decays are "constant" di...

Complexity hierarchy: Q...

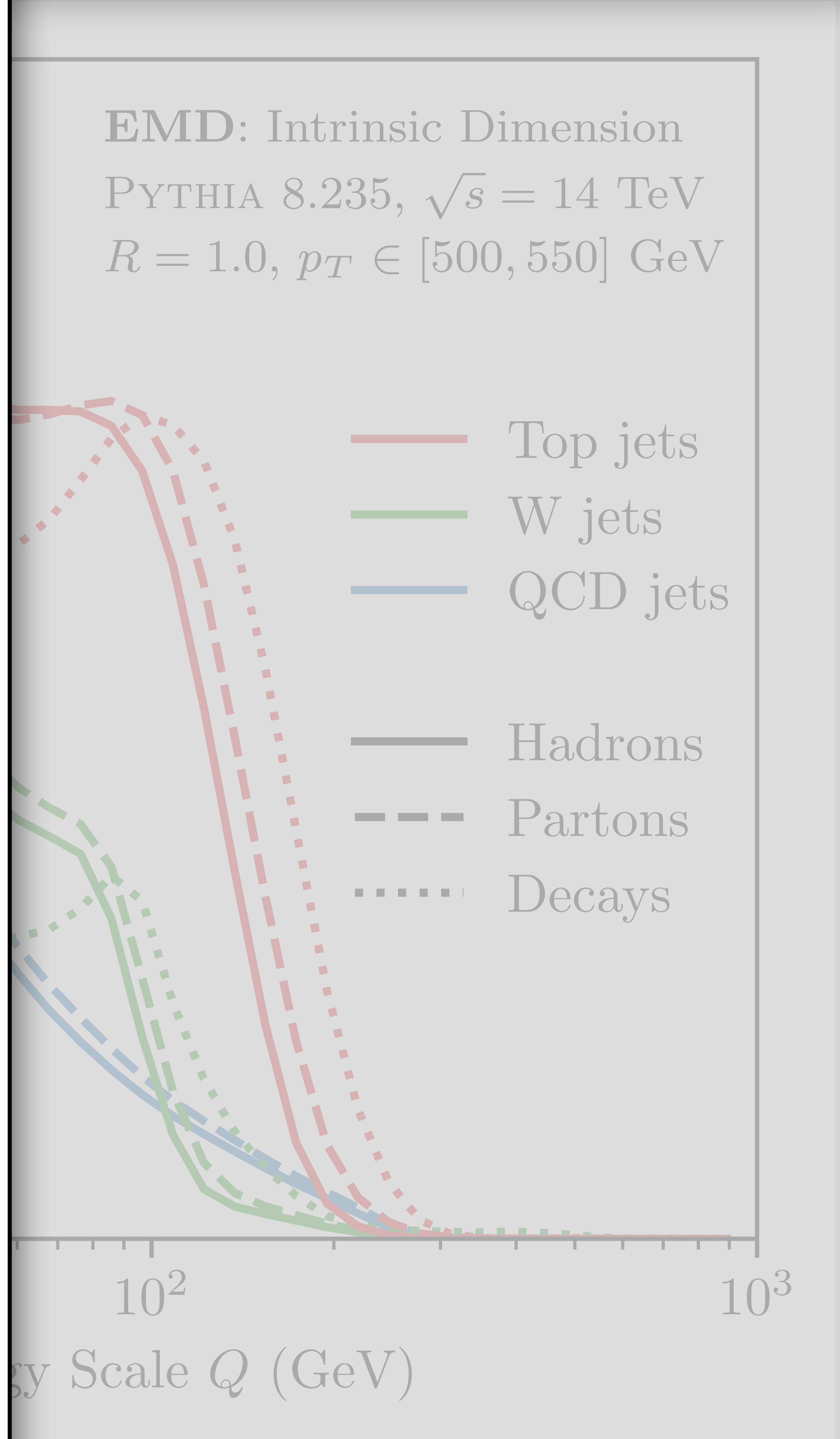
Fragmentation increases...

Hadronization important...

... in CMS Open Data



*More in backup



[... Procaccia, PRL 1983; PTK, Metodiev, Thaler, PRL 2019]

*Preliminary LL calculation for QG jets in backup

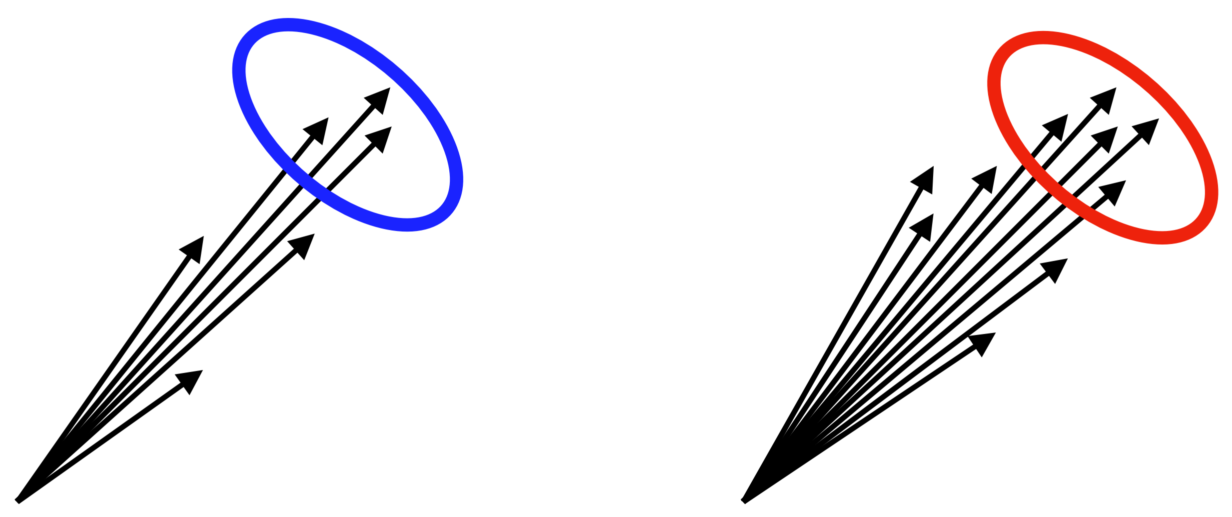
Quark and Gluon Correlation Dimensions

$$\dim(Q) = Q\frac{\partial}{\partial Q}\ln\sum_i\sum_j\Theta(\mathrm{EMD}(\mathcal{E}_i,\mathcal{E}'_j) < Q)$$

Leading log (single emission) calculation:

$$\dim_i(Q) \simeq -\frac{8\alpha_s}{\pi}C_i\ln\frac{Q}{p_T/2}$$

color factor

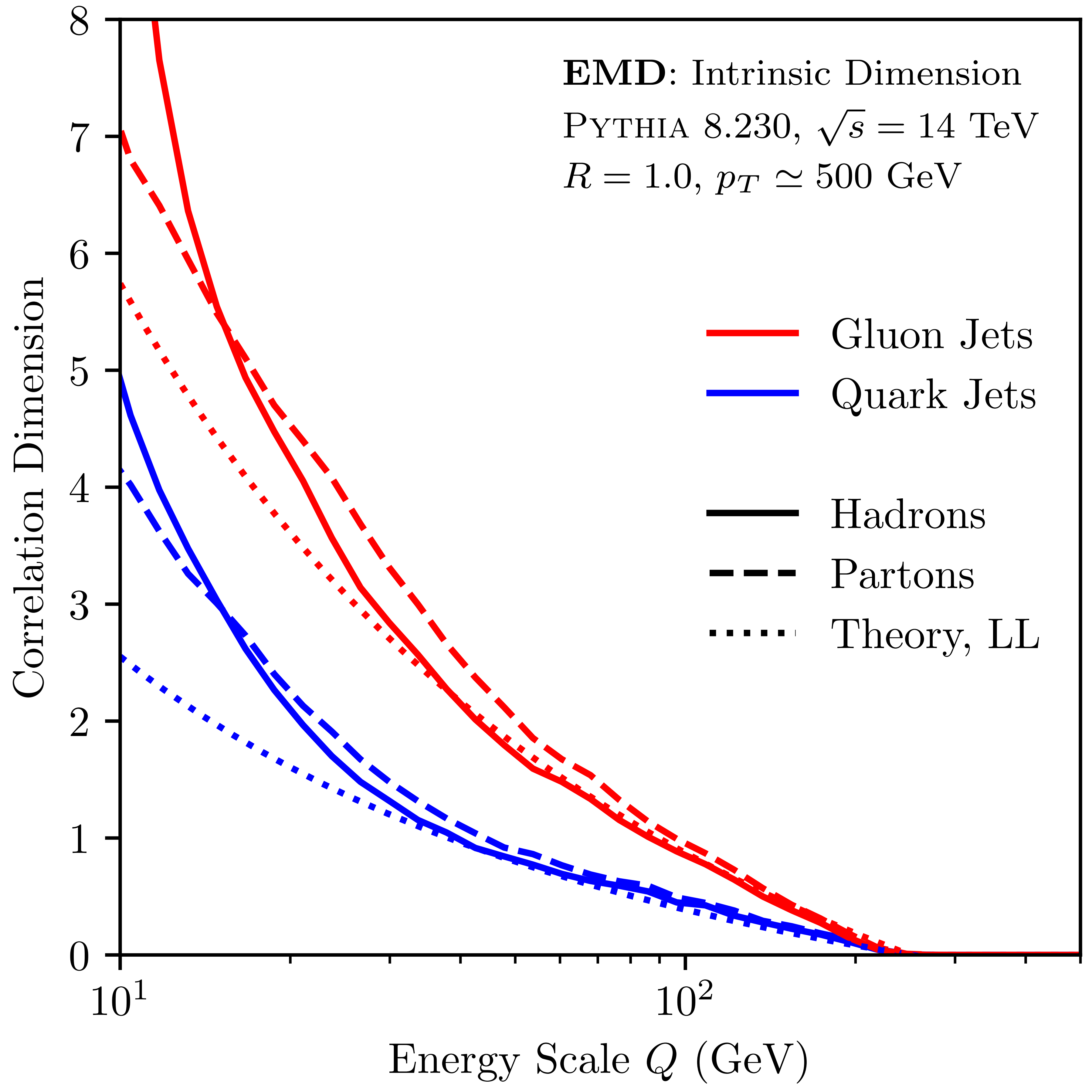


Quark

Gluon

$C_F = 4/3$

$C_A = 3$



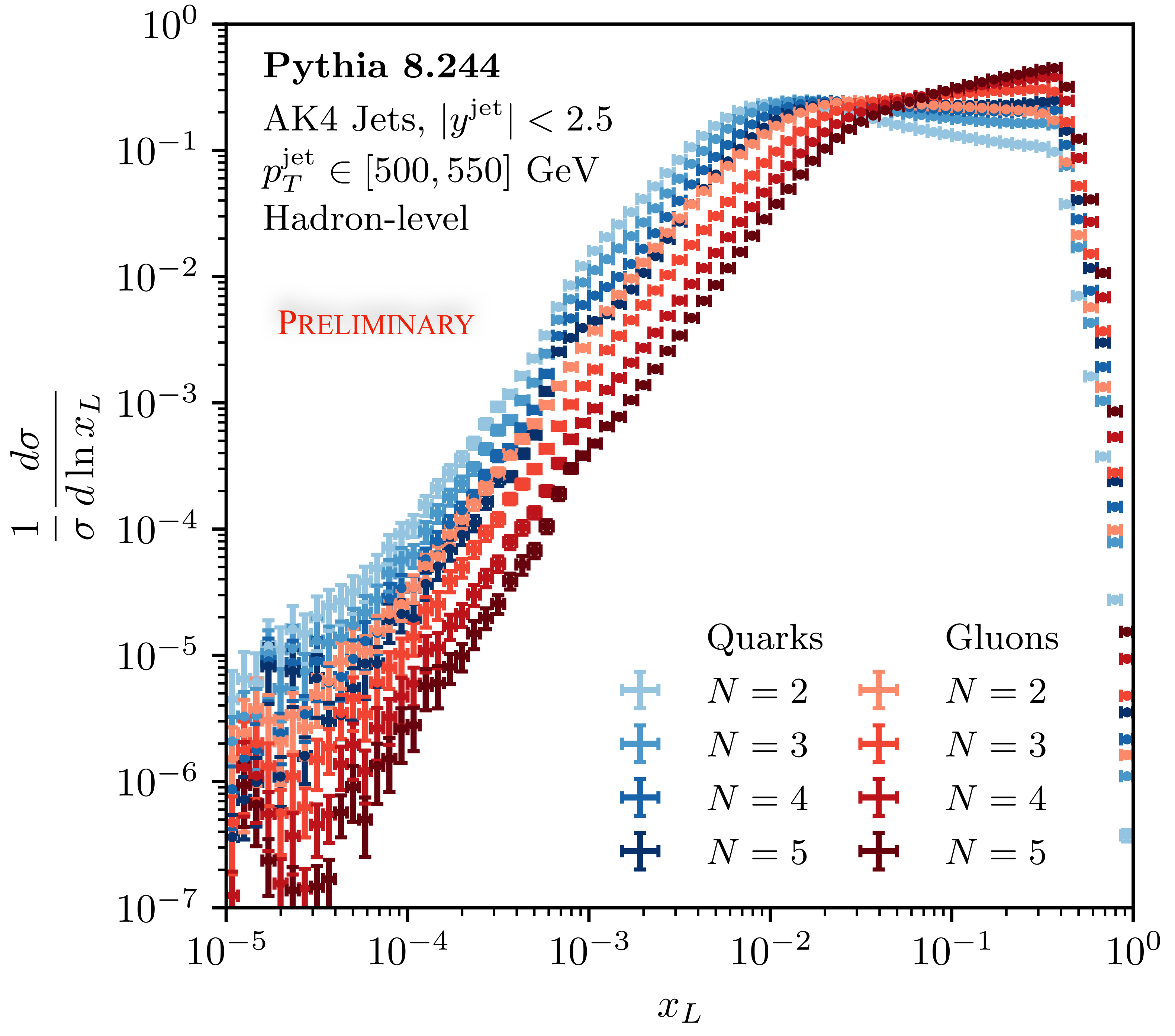
[PTK, Metodiev, Thaler, *to appear soon*]

Energy-Energy Correlators – Projection to Longest Side

[PTK, Moult, Thaler, Zhu, *to appear soon*]

Integrate out shape dependence but keep overall size dependence

$$\frac{d\Sigma^{[N]}}{dx_L} = \sum_n \sum_{1\leq i_1\leq\ldots\leq i_N\leq n} \int d\sigma_n \frac{E_{i_1}\cdots E_{i_N}}{Q^N}\delta(x_L - \max_{1\leq j<k\leq N}\{\theta_{i_j i_k}\})$$

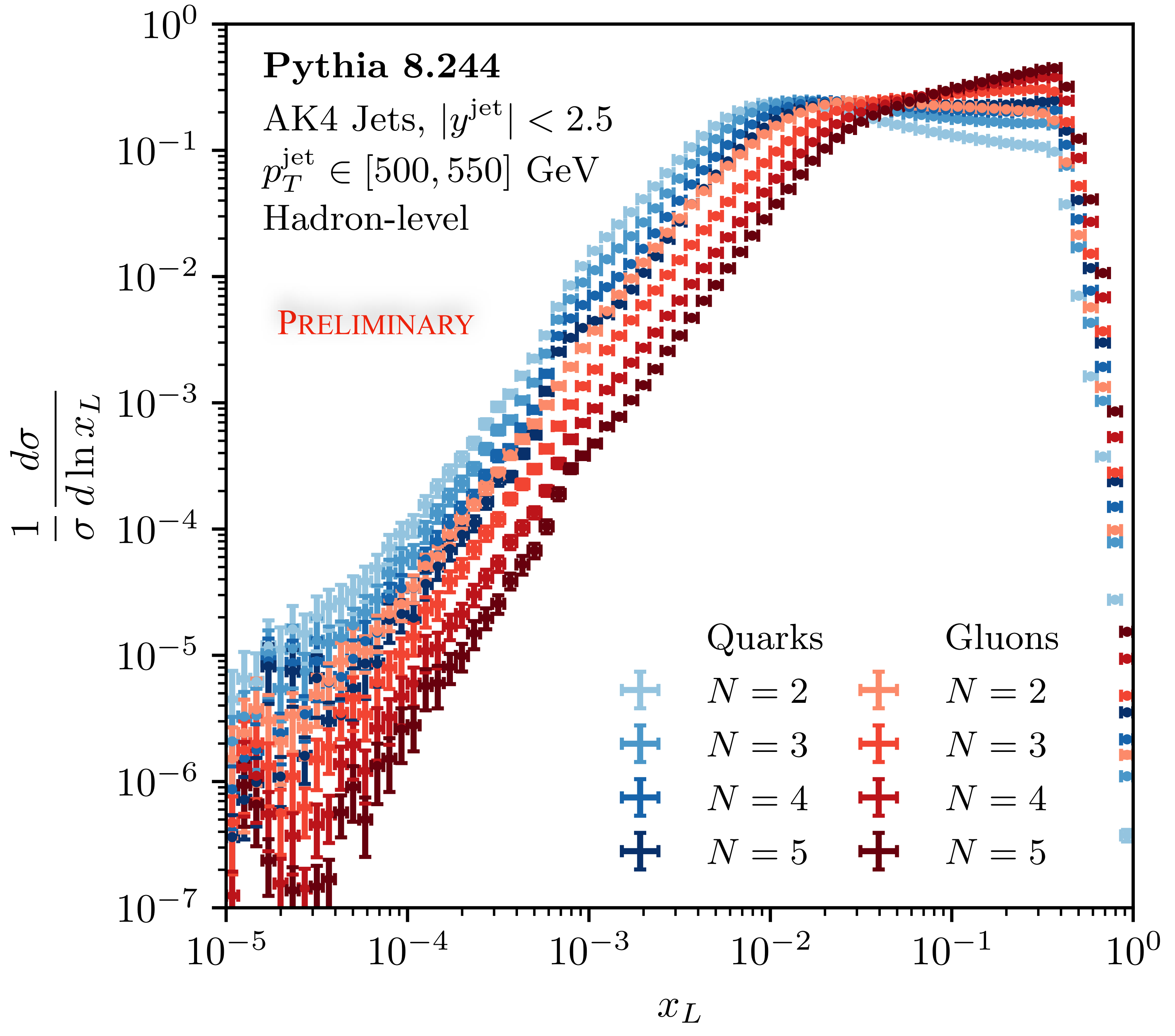


Energy-Energy Correlators – Projection to Longest Side

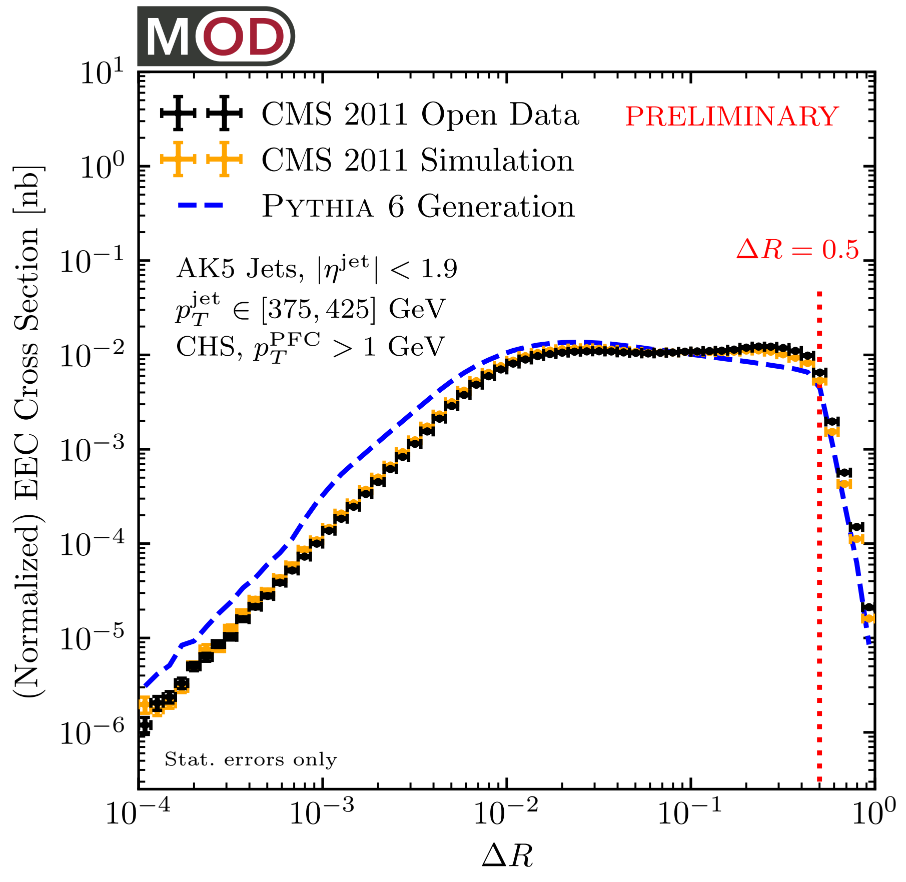
[PTK, Moult, Thaler, Zhu, *to appear soon*]

Integrate out shape dependence but keep overall size dependence

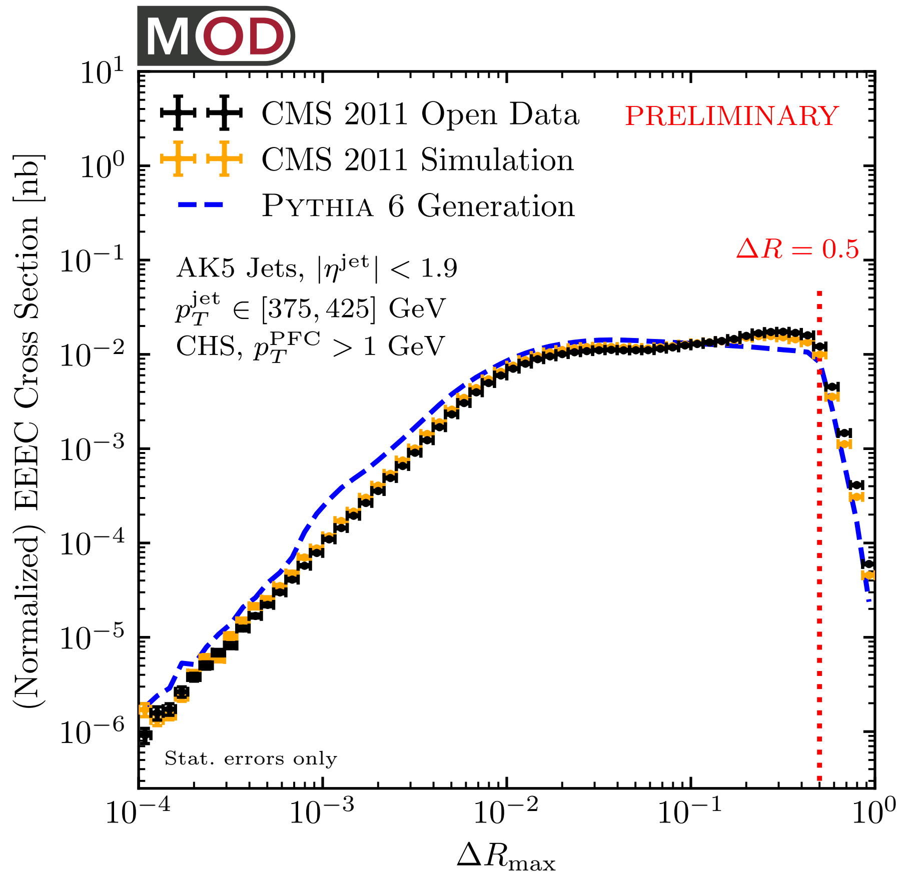
$$\frac{d\Sigma^{[N]}}{dx_L} = \sum_n \sum_{1\le i_1 \le \ldots \le i_N \le n} \int d\sigma_n \frac{E_{i_1}\cdots E_{i_N}}{Q^N}\delta(x_L - \max_{1\le j<k\le N}\{\theta_{i_j i_k}\})$$



$N = 2$



$N = 3$

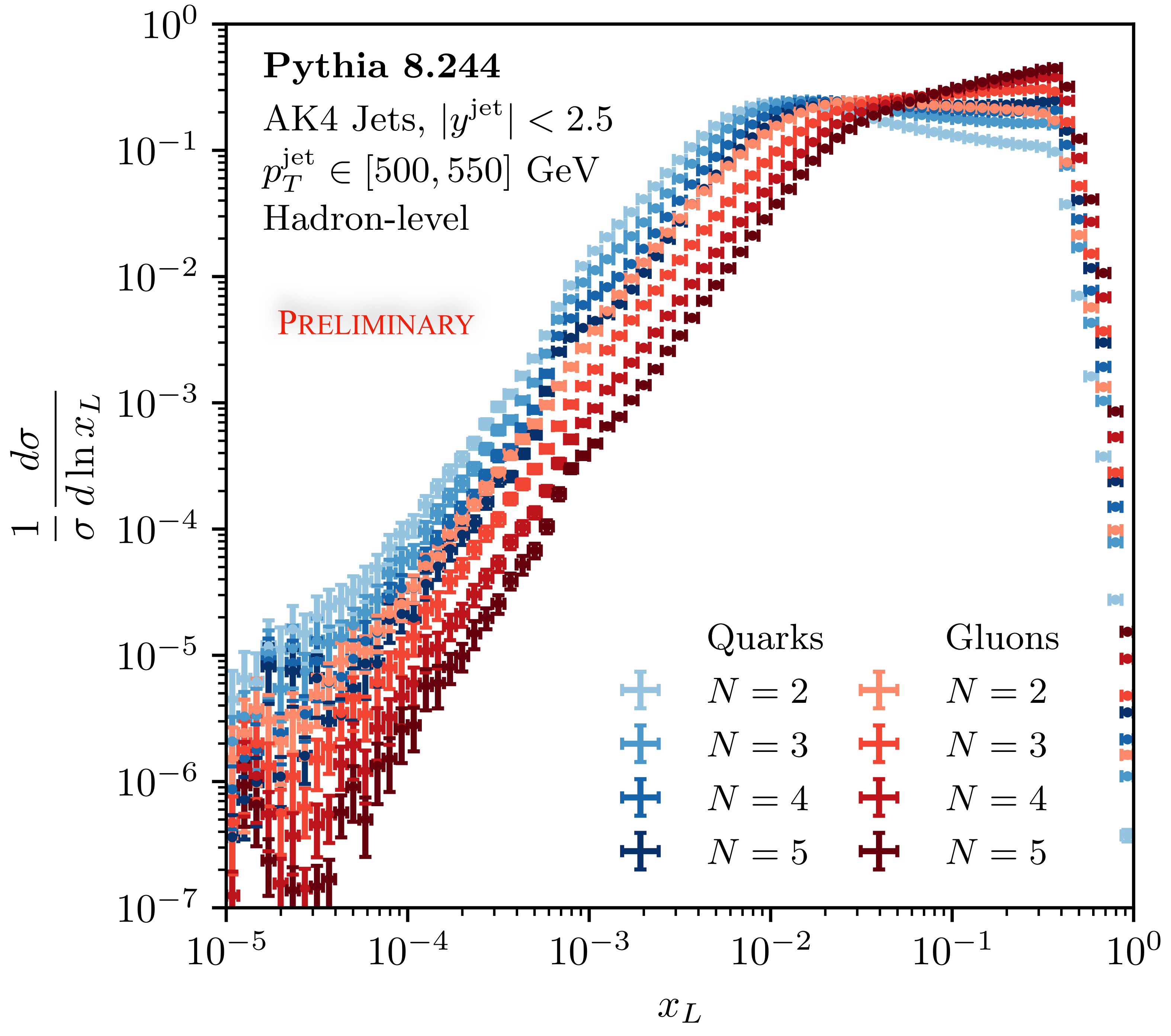


Energy-Energy Correlators – Projection to Longest Side

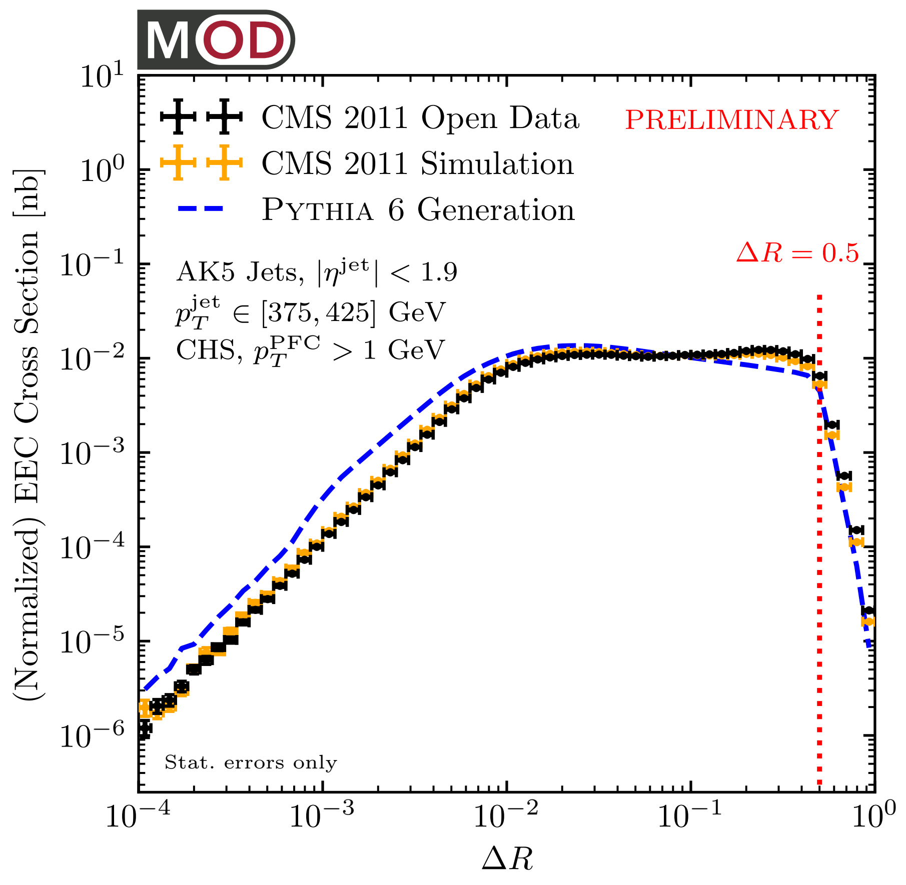
[PTK, Moult, Thaler, Zhu, *to appear soon*]

Integrate out shape dependence but keep overall size dependence

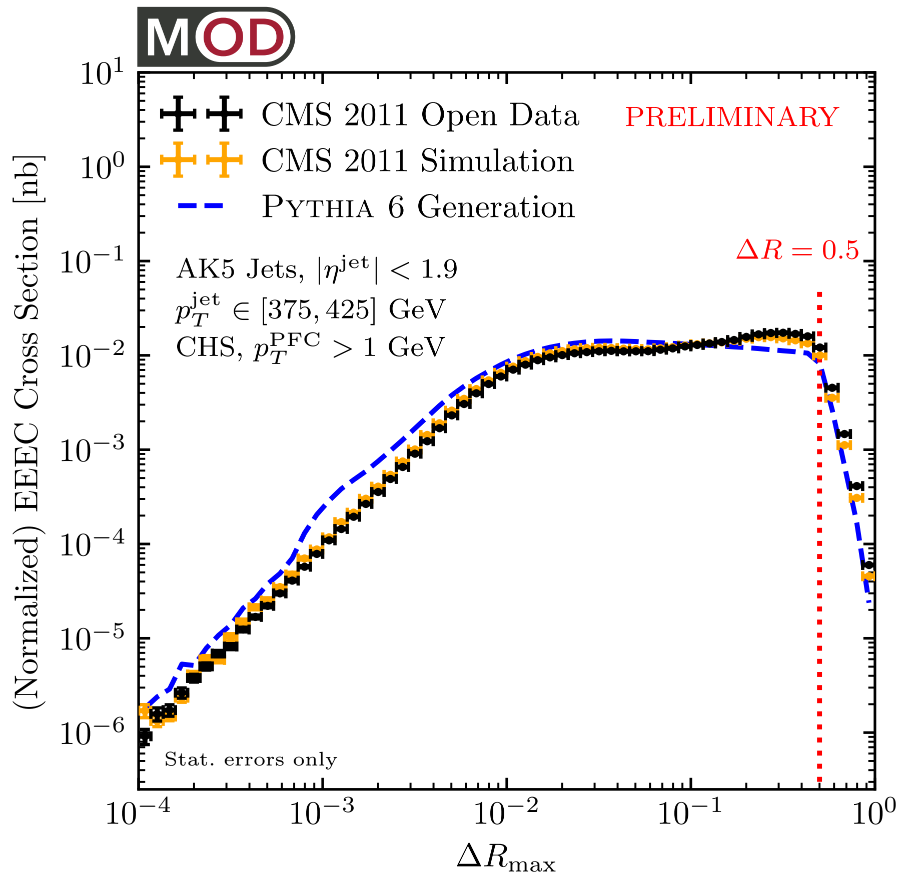
$$\frac{d\Sigma^{[N]}}{dx_L} = \sum_n \sum_{1\le i_1 \le \ldots \le i_N \le n} \int d\sigma_n \frac{E_{i_1}\cdots E_{i_N}}{Q^N}\delta(x_L - \max_{1\le j<k\le N}\{\theta_{i_j i_k}\})$$



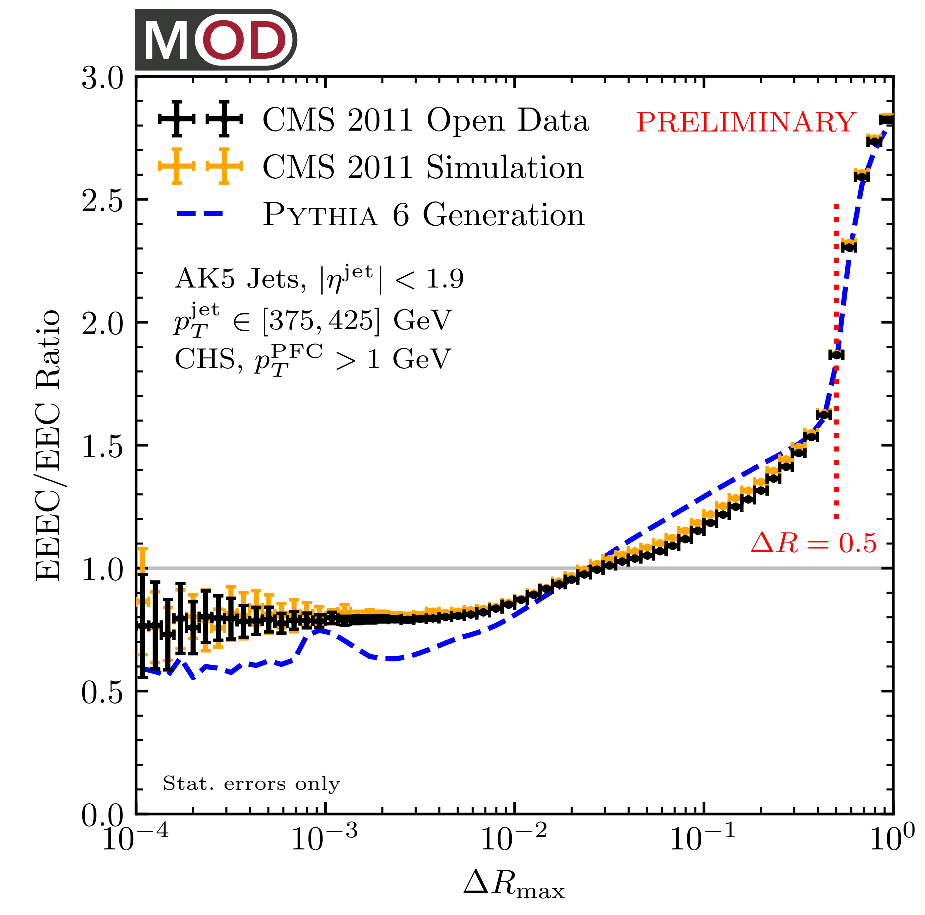
$N = 2$



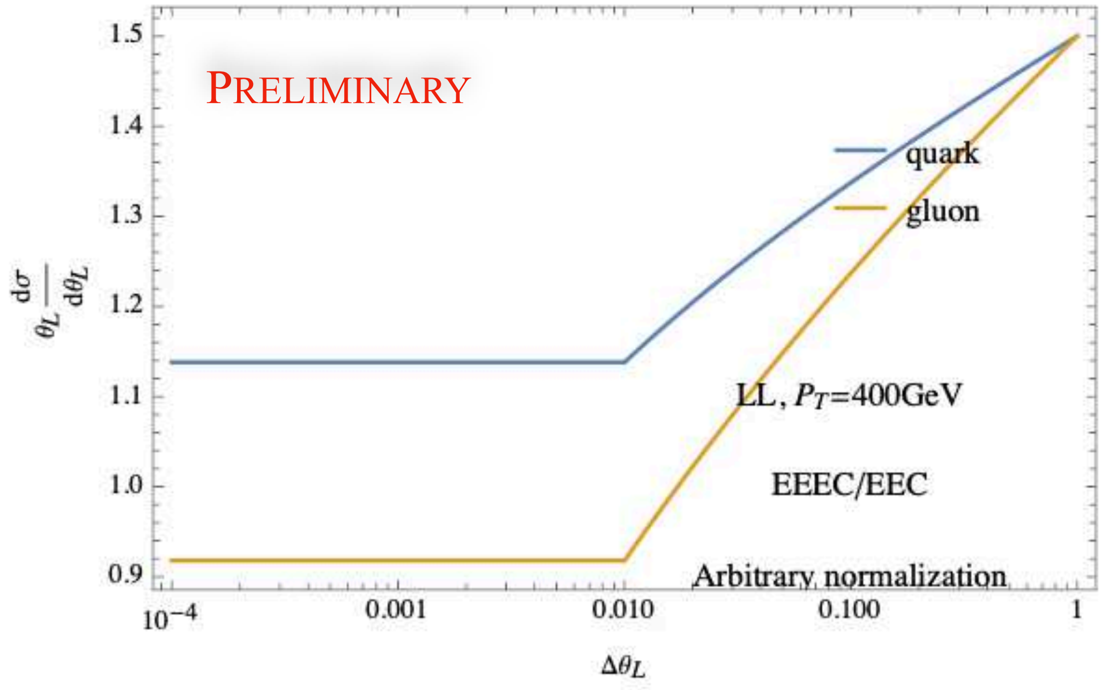
$N = 3$



EEEC/EEC Ratio



LL prediction of ratio



EEEC – Full Shape Dependence

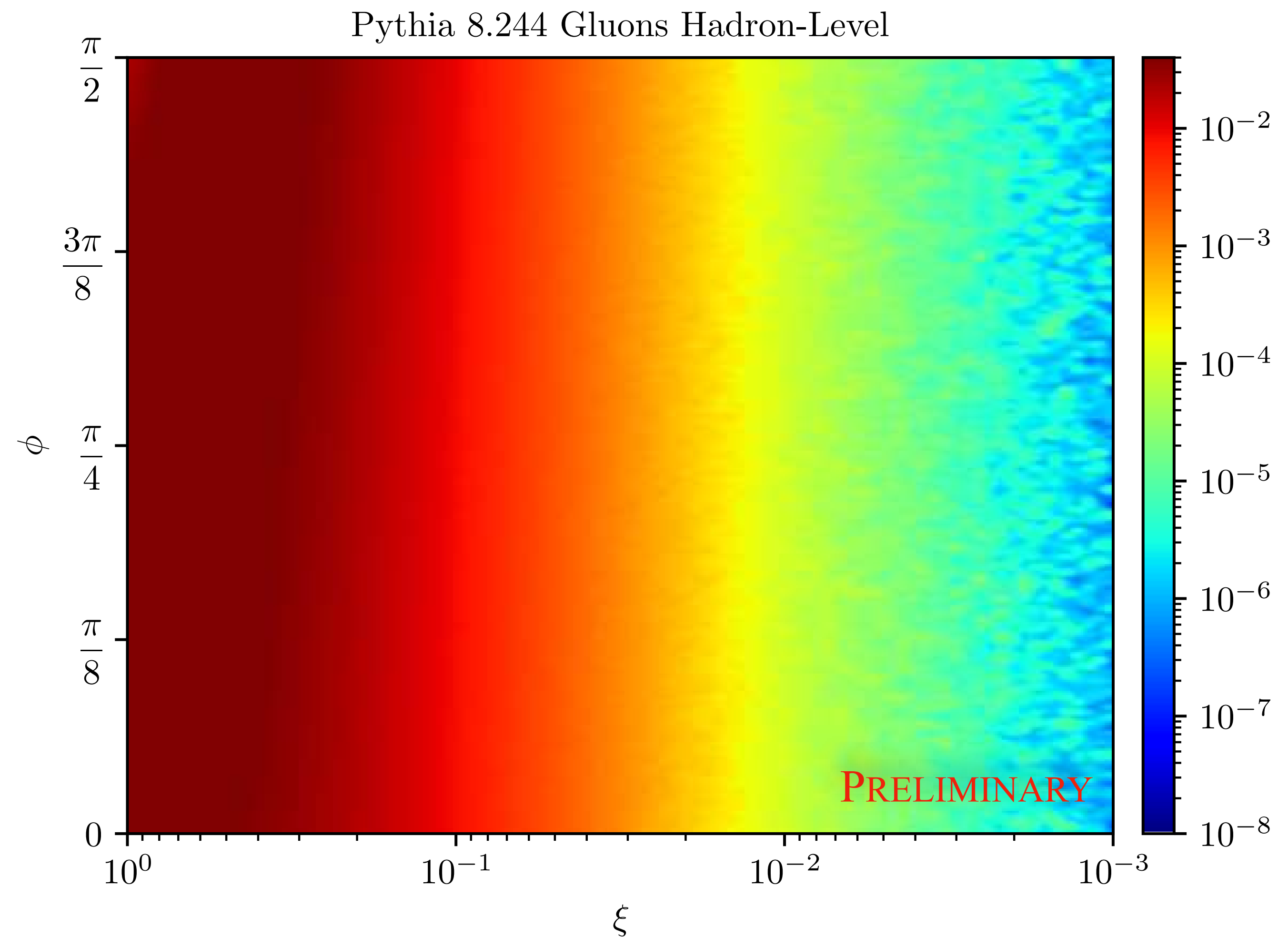
[PTK, Moult, Thaler, Zhu, *to appear soon*]

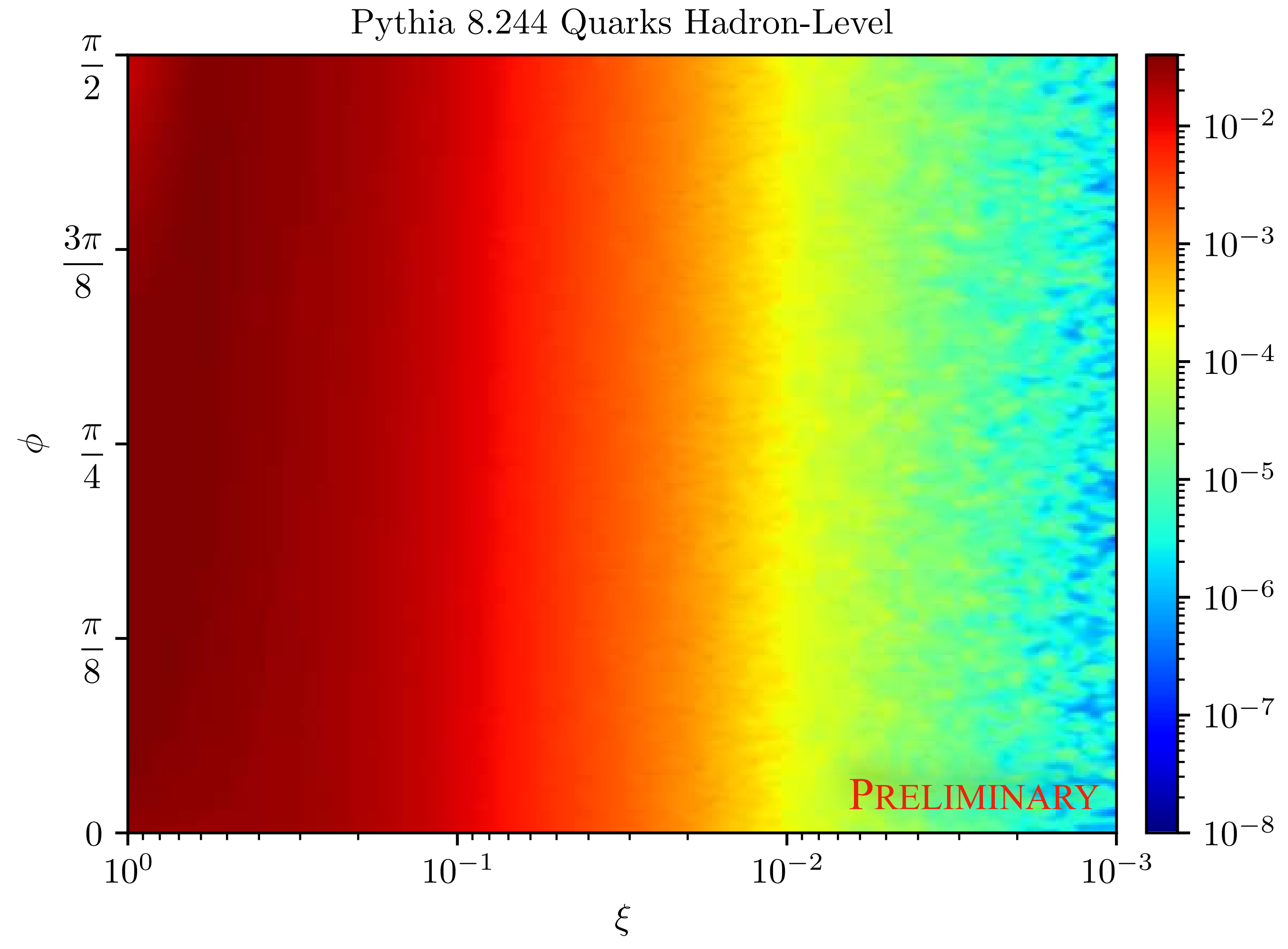
For $x_L \sim 0.01$

less aligned

ϕ

more aligned





less collinear ⟷ more collinear

EEEC – Full Shape Dependence

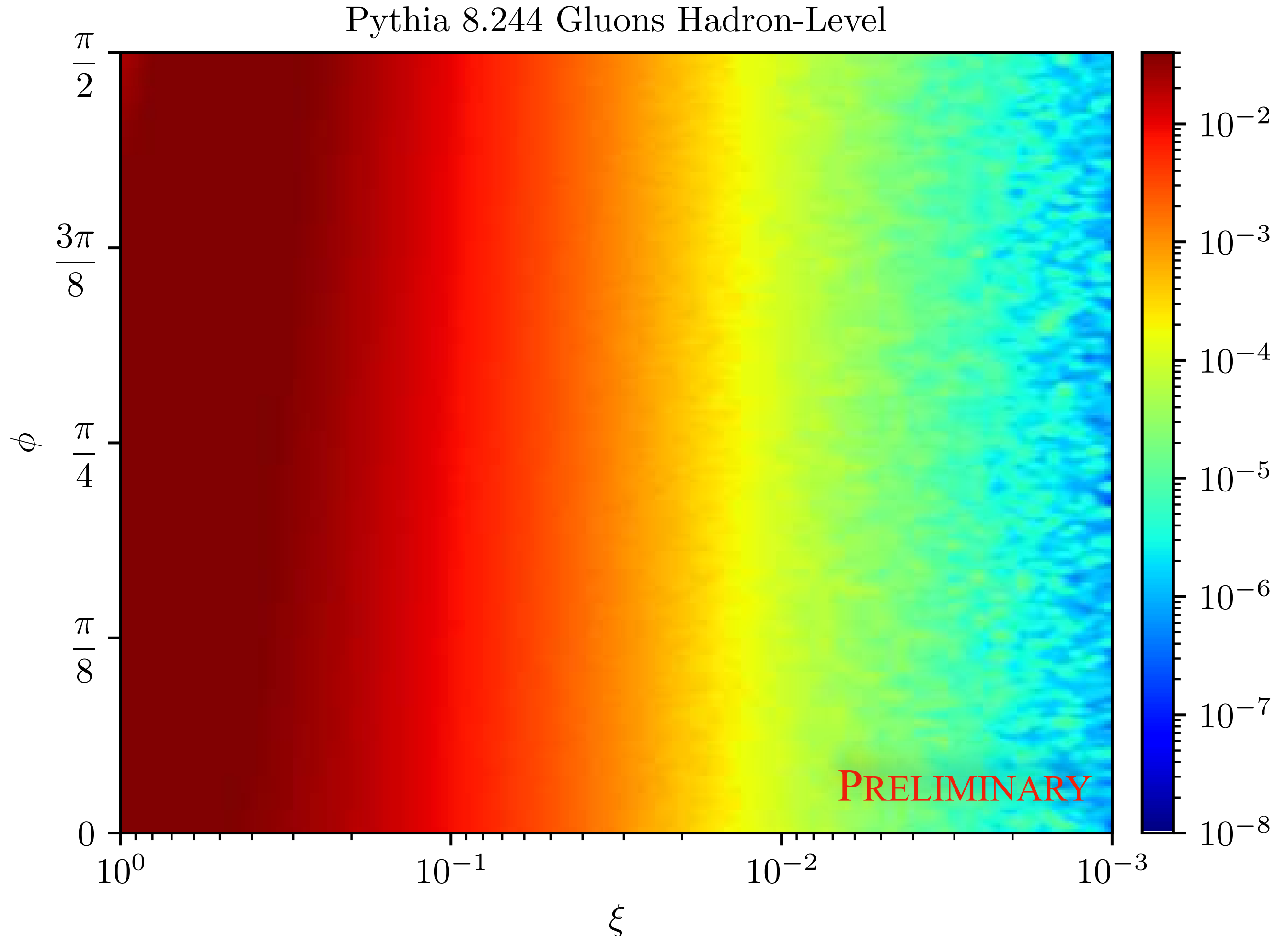
[PTK, Moult, Thaler, Zhu, *to appear soon*]

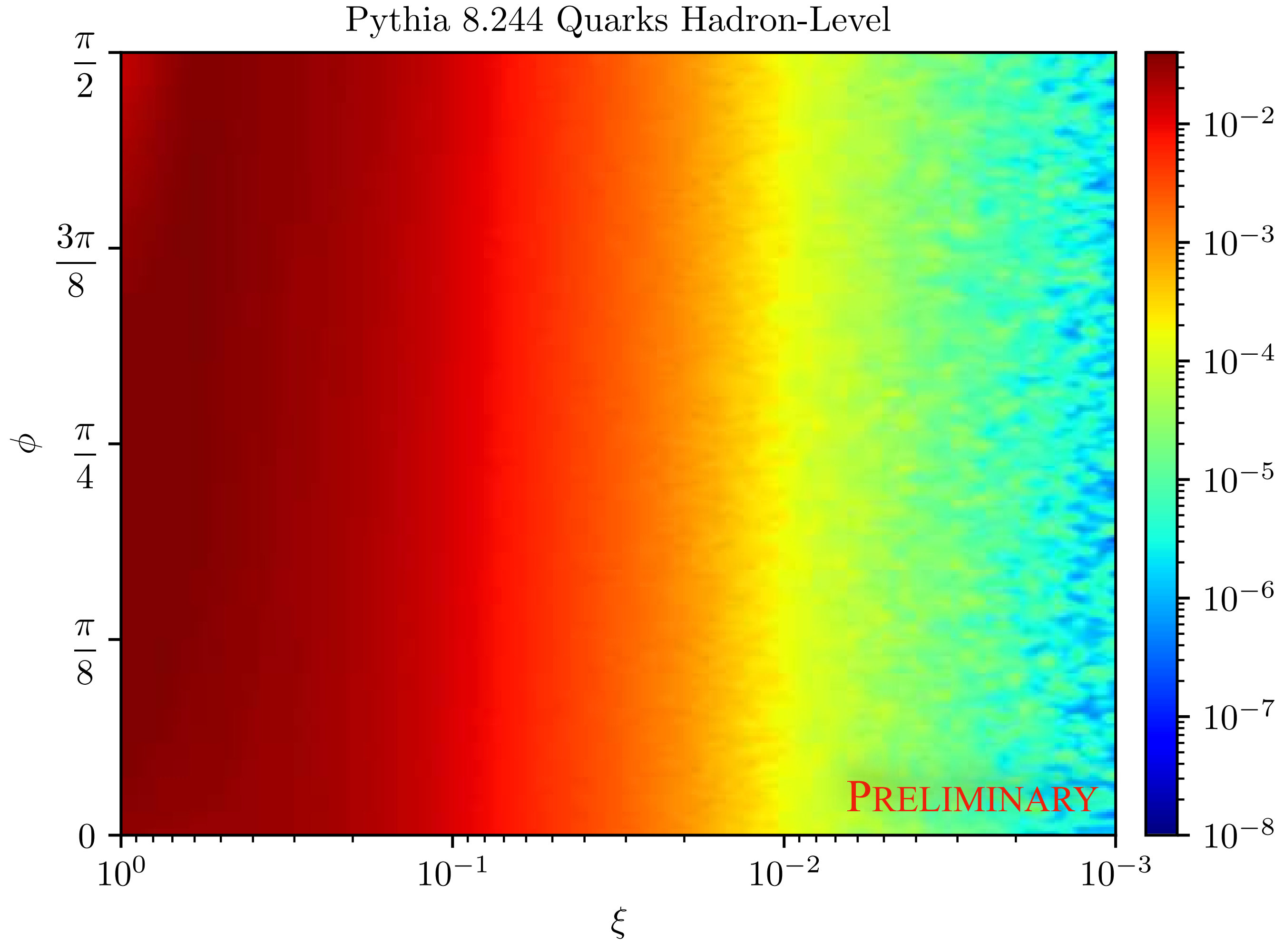
For $x_L \sim 0.01$

less aligned

ϕ

more aligned





less collinear ⟵⟶ more collinear